

Regis University

ePublications at Regis University

Regis University Student Publications
(comprehensive collection)

Regis University Student Publications

Fall 2007

Interactive Web Portal Application for Ambalta School for Children with Autism

Cathy Talbot
Regis University

Follow this and additional works at: <https://epublications.regis.edu/theses>



Part of the [Computer Sciences Commons](#)

Recommended Citation

Talbot, Cathy, "Interactive Web Portal Application for Ambalta School for Children with Autism" (2007).
Regis University Student Publications (comprehensive collection). 449.
<https://epublications.regis.edu/theses/449>

This Thesis - Open Access is brought to you for free and open access by the Regis University Student Publications at ePublications at Regis University. It has been accepted for inclusion in Regis University Student Publications (comprehensive collection) by an authorized administrator of ePublications at Regis University. For more information, please contact epublications@regis.edu.

Regis University
College for Professional Studies Graduate Programs
Final Project/Thesis

Disclaimer

Use of the materials available in the Regis University Thesis Collection ("Collection") is limited and restricted to those users who agree to comply with the following terms of use. Regis University reserves the right to deny access to the Collection to any person who violates these terms of use or who seeks to or does alter, avoid or supersede the functional conditions, restrictions and limitations of the Collection.

The site may be used only for lawful purposes. The user is solely responsible for knowing and adhering to any and all applicable laws, rules, and regulations relating or pertaining to use of the Collection.

All content in this Collection is owned by and subject to the exclusive control of Regis University and the authors of the materials. It is available only for research purposes and may not be used in violation of copyright laws or for unlawful purposes. The materials may not be downloaded in whole or in part without permission of the copyright holder or as otherwise authorized in the "fair use" standards of the U.S. copyright laws and regulations.

Interactive Web Portal Application for Ábalta School For Children with Autism

by

Cathy Talbot

cathytalbot@gmail.com

MSc in Software and Information Systems

School for Professional Studies

Regis University

Denver, Colorado

&

National University of Ireland

Galway

July 2007

Abstract

Ábalta is a school, based in Galway, designed to educate children who suffer from autism. The school was borne out of the identified need for children with autism to get a specific focused education. In July, 2001, four sets of parents came together, using their own personal funds, to establish the school and had it up and running by September, 2001. It took months of hard work, political negotiating, and lobbying, culminating with a high court case to attain funding from the state.

In today's Internet-oriented world, a web application is essential for businesses, organizations and individuals to optimize their impact on the world: to reach more people, be more efficient, learn more and achieve their goals.

As Ábalta School is ever expanding, there is an extensive need for a web portal for the school. A web portal will provide the school with a means of sharing information on autism and answering questions that people may have. Currently, when a person requires information, they contact the school and the school principle is the person who answers questions. The principle can spend at any one time, up to an hour answering questions on the telephone, sending brochures to people and explaining the methodologies used by the school. Having a web portal would enable the staff and employees of Ábalta to save time and effort by directing people to the web portal where questions can be answered and sought-after information can be found.

Ábalta also wishes to provide support for the people who are affected, either directly or indirectly, by autism. Support can be achieved using a number of different methods including communication. In a web portal, a forum would provide an opportunity for the public to discuss any topic and support each other. A forum would provide a method for communication between parents, caregivers and interested parties, to exchange encouragement and support and the answering of questions by people who have the experience to answer them.

Although Ábalta is a government recognised school, it is primarily a charity and obtains most of its funding through donations from the public. It is estimated that there is over one billion internet users globally so it is logical that one approach for raising funds would be to utilise the internet through Ábalta's web portal application.

It is also necessary that the administration team in Ábalta be able to update this web portal themselves. As there is no dedicated Information Technology specialist employed by the charity, it is important that updates to the web portal be easy to achieve and maintain. Therefore the web portal will also provide a content administration management system that will allow an administrator to update content on the web portal.

Currently, Ábalta has a manual system for recording student details. A web based application will allow for storage of student records in a database making the job of updates, insertions, deletions and queries on existing records, easy for users.

Acknowledgements

Thank you to Regis University and the National University of Ireland, Galway for providing me with the opportunity to further my education. A special thank you to Mark James who has provided me with support, guidance and encouragement throughout this year.

To the directors and staff of Ábalta School, for their help and contribution in the development of the web portal.

And lastly to the children and students of Ábalta School; they made this project worthwhile.

Table of Contents

Abstract.....	2
Acknowledgements	4
Table of Contents	5
List of Figures	7
List of Tables	9
Chapter I Introduction	10
Statement of Problem and Goals to be Achieved.....	10
Thesis Statement	12
Project Need.....	12
Organization of Chapters	14
Barriers / Limitation	16
Scope of the Project.....	17
Chapter II Review of Literature and Research	18
Definition of Terms	18
Overview of Literature and Research Sources.....	20
Research Methods	20
Literature and Research Specific to Project.....	22
Methodology / Process Models	22
Design Patterns.....	24
Platform Architecture	28
Application Server.....	34
Databases	36
Donations	38
HTML Editor.....	41
Build Framework.....	42
The Known and Unknowns about the Project	43
Contribution of the Project to the Field	44
Chapter III Methodology	45
Formats for Presenting Results / Deliverables	45
Life Cycle Models / Methodology	45
Agile Process Model.....	46
Resource Requirements	66
Review of Deliverables.....	68
Outcomes	70
Chapter IV Frameworks Architecture	73
Model-View-Controller (MVC) framework	73
Security Framework	83
Build Framework.....	87
Summary	90
Chapter V Third Party Integration	92
Payment Service Provider.....	92
Realex Payments	93
Development and Integration.....	97

Summary	107
Chapter VI Ábaltas' Web Portal Application	108
Informational Content	108
Registration	109
Login / Logout	110
Forum	112
New Post / Reply	113
Search Forum	114
Donations	115
Donation Gauge – Thermometer	118
Content Management System	119
Gallery	121
Chapter VII Project History	123
How the Project Began	123
Project Management	124
Significant Events / Milestones in the Project	125
Changes to the Project Plan	128
Chapter VIII Findings and Analysis	131
Analysis of Results	131
Evaluation of Project Goals Met	136
What went Right / What went Wrong	138
Findings / Analysis	139
Conclusion	141
Lessons Learned	141
What could be done differently	142
Summary of the Project	143
Did the Project Meet Expectations?	146
Next Stage of Evolution	148
Conclusions / Recommendations	149
Appendix A The View Component Code	153
Appendix B The Controller Component Code	181
Appendix C The Model Component Code	187
Actions: ie\edu\galway\abaltaabaschool\actions	187
Beans: ie\edu\galway\abaltaabaschool\beans	224
BusinessObjects: ie\edu\galway\abaltaabaschool\businessobjects	232
Database: ie\edu\galway\abaltaabaschool\database	238
Stats: ie\edu\galway\abaltaabaschool\stats	242
Util: ie\edu\galway\abaltaabaschool\util	244
Appendix D – The Build Code	251
Appendix E – The Database Code	255
References:	263

List of Figures

Figure 1 Model-View-Controller Structure (eNode Inc. (2002))	25
Figure 2 Agile Process Model (Software Engineering Methodology (2003)).....	47
Figure 3 Extreme Programming (XP)(Don Wells (2006)).....	49
Figure 4 Release Schedule.....	56
Figure 5 Iteration Schedule.....	57
Figure 6 MVC Design Model (JSP Architecture Overview, n.d.).....	58
Figure 7 Class File system for Informational Content for Ábalta.....	62
Figure 8 Class File system for Forum for Ábalta.....	63
Figure 9 Class File system for Donations for Ábalta.....	64
Figure 10 Class File system for Content Management System for Ábalta	65
Figure 11 ActionMapping from struts-config.xml for Ábaltas' forum search facility	75
Figure 12 UI for forum search	75
Figure 13 ActionFormBean from struts-config.xml for Ábaltas' user registration facility	76
Figure 14 Invalid username submitted and subsequent error message returned from Action.....	77
Figure 15 JavaBean for accessing and manipulating Ábaltas' forum messages	79
Figure 16 Validation for user form input (Corresponding to Controllers ActionFormBean)	79
Figure 17 Two-tier model for database access (Java Sun JDBC Introduction (1995-2007))	80
Figure 18 MySQL database configuration	81
Figure 19 JDBC initialising new data source	81
Figure 20 SQL prepared statements for inserting into MySQL database	81
Figure 21 Web portal accessing Bean for data for output to screen	82
Figure 22 Login facility	84
Figure 23 Security Configuration in web.xml file	85
Figure 24 Login form action	85
Figure 25 Authorisation security-constraints for the forum facility in Ábalta	86
Figure 26 Security Roles for Ábalta.....	86
Figure 27 Authorization unsuccessful	87
Figure 28 Unauthorised access	87
Figure 29 Ábaltas' build.xml file - compile and clear targets	89
Figure 30 build.xml - Build process.....	89
Figure 31 Ábaltas' File Structure.....	90
Figure 32 RealAuth Redirect Solution (Realauth Developer's Guide).....	95
Figure 33 template.html.....	97
Figure 34 donations_frame.jsp.....	98
Figure 35 struts-config.xml file	99
Figure 36 Error Handling – DonationAction.java.....	100
Figure 37 Hashing algorithm SHA-1	101
Figure 38 Initialising the JavaBean with required data.....	102
Figure 39 Redirection file donations_fram2.jsp sends required information	102

Figure 40 Secured server provided by Realex Payments	103
Figure 41 struts-config.jsp code for Realex's response	103
Figure 42 Response data from Realex Payments (Reauth Developer's Guide, 2007).	104
Figure 43 Response Codes (Reauth Developer's Guide, 2007).....	104
Figure 44 Verify that donation was successful	105
Figure 45 MailUtil.java invokes the java mail API	106
Figure 46 Sample Receipt Email	106
Figure 47 Home Page of Ábaltas' web portal application.....	109
Figure 48 Registration Process	110
Figure 49 Login.....	111
Figure 50 Logout.....	111
Figure 51 Forum Discussions	112
Figure 53 View / Reply to Message	113
Figure 55 Forum Search	115
Figure 56 Register for Donation Feature.....	116
Figure 57 Secured Server for Card Details Entry	117
Figure 58 Unsuccessful Donation	117
Figure 59 Email for Successful Donation.....	118
Figure 60 Donation Gauge - Thermometer	119
Figure 61 Edit Content	120
Figure 62 HTML Editor for Content Management System.....	121

List of Tables

Table 1 Cost of MyCharity.ie	38
Table 2 Cost of Realex	40
Table 3 User Stories	54
Table 4 Significant Milestones in this project	128

Chapter I Introduction

Ábalta is a school, based in Galway, designed to educate children who suffer from autism. The school promotes the teaching of applying an education methodology called Applied Behavioural Analysis (ABA). This methodology breaks down educational tasks into tiny building blocks which a child with autism can be taught.

In addition to being a school, Ábalta is an Irish registered charity who relies on the generosity of the public to help fund and operate the school. This chapter serves to introduce Ábalta, establish why this charity was chosen as a thesis subject, convey the subsequent chapters and layout of this thesis project and discuss and outline the limitations and barriers that confine the research, design and development of the project.

Statement of Problem and Goals to be Achieved

Ábalta School was born out of the identified need for children with autism to get a specific focused education. In July, 2001, four sets of parents came together, using their own personal funds, to establish the school and had it up and running by September, 2001. It took months of hard work, political negotiating, and lobbying, culminating with a high court case to attain funding from the state.

Ábalta opened in September, 2001, with six students. Four of these children had, prior to attending Ábalta, been educated in home-based programmes. The other two children had been receiving other types of autism services.

Ábalta School expanded in 2003 to cater for 12 children at a site in Cuan Glas in Galway. In 2005, Ábalta School opened an additional facility in Newcastle in Galway to cater for an additional six children. Ábalta currently caters for 18 children and has had four of its students transfer into mainstream services where they can access an appropriate curriculum.

A web portal application will provide the school with a central repository of information, which will be available for any parent or caregiver in need of information regarding autism and the resources and support available for children with autism at any time. It will offer a means for people to communicate and support each other by providing a forum for discussion.

A web-portal application will provide a method for fundraising online, to utilise the success of the internet to help its cause, and to provide support and communication for people who are affected by autism.

It is necessary that the administration team in Ábalta be able to update this web portal themselves. As there is no dedicated Information Technology specialist employed by the charity, it is important that updates to the web portal be easy to achieve and maintain. Therefore the web portal will also provide a content administration management system that will allow an administrator to update content on the web portal.

Currently, Ábalta has a manual system for recording student details. A web based application will allow for storage of student records in a database making the job of updates, insertions, deletions and queries on existing records, easy for users.

Thesis Statement

The internet serves as an excellent channel for promoting information and communication to billions of people across the world. For charities, like Ábalta, the internet offers a means for communicating, raising awareness, generating funds and providing support for people afflicted with disability and illness. Today, many technologies exist for designing and developing web applications; albeit certain technologies may be superior for engineering web based applications than others. Exploration of these technologies is necessary to determine the most efficient and effective technology route to design and develop Ábalta's web portal application. This project will research and analyse web based technologies for the optimised design and development of a web portal application for the Ábalta School for Children with Autism. Using these technologies, a web portal application will be designed, developed and implemented and will be used to communicate, raise awareness, generate funds and provide support for people who are affected by autism.

Project Need

In today's Internet-oriented world, a web application is essential for businesses, organizations and individuals to optimize their impact on the world: to reach more people, be more efficient, learn more and achieve their goals.

As Ábalta School is ever expanding, there is an extensive need for a web portal for the school. A web portal will provide the school with a means of sharing information on autism and answering questions that people may have. Currently, when a person requires information, they contact the school and the school principle is the person who answers questions. The principle can spend at any one time, up to an hour answering questions on the telephone, sending brochures to people and explaining the methodologies used by the school. Having a web portal would enable the staff and employees of Ábalta to save time and effort by directing people to the web portal where questions can be answered and sought-after information can be found.

A web-portal application will provide a method for fundraising online, to utilise the success of the internet to help its cause, and to provide support and communication for people who are affected by autism.

It is necessary that the administration team in Ábalta be able to update this web portal themselves. Therefore the web portal will also provide a content administration management system that will allow an administrator to update content on the web portal.

Currently, Ábalta has a manual system for recording student details. A web based application will allow for storage of student records in a database making the job of updates, insertions, deletions and queries on existing records, easy for users.

Organization of Chapters

Chapter 1 serves as an introduction to Ábalta School for Children with Autism and attempts to justify the necessity of a web portal application for the school and the relevance of such an application in today's world. The chapter discusses and outlines the limitations and barriers that confine the research, design and development of the project.

Chapter 2 offers a review of all the literature and sources that were researched and investigated throughout the life cycle of the project. The chapter will detail the research methods used in investigating the problem and provide a thorough and comprehensive description and account of technologies that are available for web based applications. Researched technologies are compared and contrasted to determine the most effective and efficient technology for the project solution. Details of what is known and unknown about the project will be stated and the contribution that this project will make to the field will be acknowledged.

Chapter 3 discusses the design of this project. It portrays the design methodology used during the life cycle of the project and outlines the deliverables to be presented upon completion.

Chapter 4 offers an insight into the framework architecture of the project. There were a number of frameworks manipulated during the design and development of Ábaltas' web portal application. These frameworks incorporate the design pattern framework, the security framework and the system operations / build framework.

Chapter 5 serves to portray the integration with third party companies. In today's technologically advanced world, it is estimated that there is over one billion internet users globally so it was logical that one approach for generating funds would be to utilise the internet through Ábaltas' web portal application. This prompted the integration with third party organisation who would provide the secured server required for the safe and successful acceptance and processing of credit and debit cards over the internet.

Chapter 6 will provide an overview of Ábaltas' web portal application, as it is delivered and deployed. It will detail the features of the web portal and what it offers to Ábalta.

Chapter 7 serves to provide an insight into the evolution of the project from the initial idea of developing a web portal application for Ábalta School to its present state.

Chapter 8 attempts to accumulate the findings and analysis of the project, evaluate whether the project goals were met and discuss what sections of the project went according to plan and what sections of the project did not go according to plan.

Finally the conclusion will detail what was learned from the experience of this project; what may be done differently. It will offer a summary of the project and whether the project met the initial expectations. It will document the next stage of evolution of the project and offer recommendations for the future.

Barriers / Limitation

As Ábalta School for Children with Autism is a charity based organization, it is imperative that costs be kept to a minimum; however, development of a web portal application necessitates that expenses will be incurred. There are several areas that expenditure will be essential including internet hosting fees and domain name purchase. Other possible technologies, including security, may incur charges so it is important that cost is considered when researching, investigating and determining which technologies to utilize.

There are a vast range of technologies available for the design and development of web applications. These technologies must be appropriately explored and examined to determine the most suitable for this project. The schedule of this project ensures that time will be of the utmost importance and it will be crucial that time be spent effectively in researching and implementing this project.

Ábalta currently has a manual system for information sharing and communication, providing support, generating funds / donations and raising awareness in general. Ábalta directors, its principle, employees and teachers will need to be trained and educated on the new web portal application.

Scope of the Project

The web portal application will provide a hub of information on the Ábalta School, its practices and autism in general. It will provide information and facilities, which include:

- “About ABA” – the education methodology that is used in the school to teach students afflicted with autism
- “Mission Statements” – which explains the core purpose of the charity and what they are striving to achieve
- “Testimonials” – information from the parents of the children being educated in Ábalta School and how the methodologies used helps and benefits their children
- “Literature” – further information and readings on autism
- “Fundraising / Events” – information on current fundraising and events to promote the awareness of Ábalta and to generate funds for the school
- “Make a Donation” – a method for providing the ability for internet / web users to donate money to the school from the internet
- “Forum” - to enable discussions / questions between parents, teachers and interested parties
- “FAQ” – prepared frequently asked questions
- “Gallery” – provide for the presentation of photographs from events and the school onto the internet
- “Login / Logout” – allowing for the logging in and out of users of the web portal application
- “Content Administration / Management System” – allowing an administrator to dynamically update content on the web portal

Chapter II Review of Literature and Research

This chapter offers a review of all the literature and sources that were researched and investigated throughout the life cycle of the project. It will detail the research methods used in investigating the problem and provide a thorough and comprehensive description and account of technologies that are available for web based applications. Researched technologies are compared and contrasted to determine the most effective and efficient technology for the project solution. Details of what is known and unknown about the project will be stated and the contribution that this project will make to the field will be acknowledged.

Definition of Terms

MVC	Model View Controller
HTML	Hypertext Mark-up Language
XML	Extensible Mark-up Language
XSL	Extensible Style-sheet Language
JSP	Java Server Pages
EJB	Enterprise Java Beans
COM+	Component Object Model
CLR	Common Language Runtime
GUI	Graphical User Interface
COBOL	Common Business Oriented Language
IL	Intermediary Language
VS	Visual Studio

J2EE	Java 2 Platform, Enterprise Edition
API	Application Programming Interface
CPU	Central Processing Unit
JVM	Java Virtual Machine
JRE	Java Runtime Environment
ASP	Active Server Pages
JDBC	Java Database Connection
RMI	Remote Method Invocation
JNDI	Java Naming and Directory Interface
JMS	Java Message Service
JTA	Java Transaction API
JAAS	Java Authentication and Authorization Service
CMP	Container-Managed Persistence
IIS	Internet Information Services
MySQL	My Structured Query Language
ANSI	American National Standards Institute
ISO	International Organization for Standardization
GPL	General Public License
GNU	GNU's Not Unix
XP	Extreme Programming
WAR	Web Application Archive
SSL	Secure Sockets Layer
LGPL	Lesser General Public License

Overview of Literature and Research Sources

Many technologies had been proposed for the design and development of this project and it was crucial that intensive research be carried out to determine which technological route was best suited to the needs of the school. This chapter outlines and investigates the various technologies and methods that are available for implementing a web portal application for Ábalta.

Numerous formats have been used for researching and investigating the solution. Literature from library sources has been analyzed and appraised; meetings with the directors and the principle of Ábalta were setup and attended to transfer information; online tutorials, articles and forums were investigated and undertaken for knowledge and technical transfer.

Investigation and research of literature and technologies provided a means for objectively making a choice on the best approach for achieving the solution of a web portal application for Ábalta.

Research Methods

The investigation and research started with a meeting with the board of directors of Ábalta. This meeting was used to gather information regarding the charity, where it was currently lacking and what they would like to achieve from a technological approach. Meetings were ongoing throughout the analysis phase of the project as it was important to continually gauge the most appropriate direction and functionality of the web portal.

Online articles and forums were explored as it was invaluable to investigate developers and experts' opinions and judgments on the various technologies available. These also helped in determining which technologies would be most capable and appropriate for what was to be achieved. Online tutorials in numerous software technologies also proved invaluable as they provided a means of software knowledge transfer and were easily accessible.

Software engineering, design and development books and manuals were utilized to provide knowledge and understanding in numerous specialized areas. These books and manuals were important as they provided education on the different technologies used in the development of this project. These were sourced from the college library through off-campus access and from the literature that was used and studied throughout the college year. Course modules, studied throughout the year, were applicable to the technologies used in the project and were fundamental during the design and development of the web portal application. Literature from these modules proved invaluable to educating and teaching these specialized technologies.

Finally a meeting was set up with the teachers and experts of Ábalta and was attended by some of the students of the school. This helped achieve a greater understanding of autism and its effects on family, friends and the people who work with children affected with autism.

Literature and Research Specific to Project

The literature and research section offers a review of all the literature and sources regarding technologies available that were researched and investigated throughout the life cycle of the project. This section will provide a thorough and comprehensive description and account of technologies that are available and suitable for the development of Ábaltas' web portal application. Researched technologies are compared and contrasted, when applicable, to determine the most effective and efficient technology for the project solution.

Methodology / Process Models

According to Roger S. Pressman (2005), a process model defines a distinct set of activities, actions, tasks, milestones and work products that are required in order to engineer high-quality software. It is important to implement these models as they provide stability, control and organization to an activity that has the potential to become quite chaotic. The process guides an engineer or a team of engineers through a set of framework activities that are organized into a process flow. The life cycle of Ábaltas' web portal application would benefit greatly from following a methodology as it is a proven method for addressing customer requirements and delivering software products on time.

Agile Process

According to Roger S. Pressman (2005), agile software engineering combines a philosophy and a set of development guidelines. The philosophy encourages customer satisfaction and early incremental delivery of software; small, highly motivated project

teams; informal methods; minimal software engineering work products and overall development simplicity. Agile software processes are characterized by three key assumptions about the majority of software projects:

- The difficulty in predicting which software requirements will persist and which will be modified.
- Design and construction are interleaved, for many types of software. It is recommended that both these activities be performed in tandem so that design models are proven as they are created.
- The activities analysis, design, construction and testing are not as predictable (from a planning point of view).

The fundamental responsibility of an agile process is to be adaptable. For progression, an agile software process must adapt incrementally. To adapt incrementally, an agile team requires customer feedback and an effective channel for customer feedback is either an operational prototype or a portion of an operational system. Software increments (executable prototypes or an operational system portion) should be delivered in short time periods so that adaptation keeps pace with change. This iterative approach enables regular customer evaluation of the software increment, providing necessary feedback to the software team, and influencing the process adaptations that are made to accommodate the feedback. (Roger S. Pressman, 2005)

Extreme Programming (XP)

Ábaltas' web portal application would benefit immensely from the extreme programming (XP) agile methodology. According to Don Wells (2006), XP is a proven deliberate and

disciplined approach to software design and development. It is successful because it emphasizes customer satisfaction. The methodology is designed to deliver the software customers need when it is needed. XP empowers developers to confidently respond to changing customer requirements, even when changes are required late in the life cycle.

Design Patterns

According to Steve Masover (June 2004), it has become widely accepted amongst software engineers and architects that designing software applications in explicit conformity to patterns facilitates the re-use of insight, knowledge and experience assembled by the best and brightest amongst software engineers over the course of thousands of real-world software development efforts. Design patterns can represent solutions to problems that arise when developing software within a particular context. The design and development of Ábaltas' web portal application would greatly benefit from following a design pattern as they provide a proven solution to solving software development problems and issues, resulting in highly cohesive modules. The design pattern applicable researched during the analysis stage for designing and developing Ábaltas' web portal application was the model-view-controller.

According to Steve Masover (June, 2004), the Model-View-Controller (MVC) is a design pattern intended for the architecture of web applications. It has been widely adopted, across many languages and implementation frameworks, as a pattern whose purpose is to achieve a clean separation between three components of almost any web application. These components can be defined as the model, the view and the controller and they each handle a discreet set of tasks.

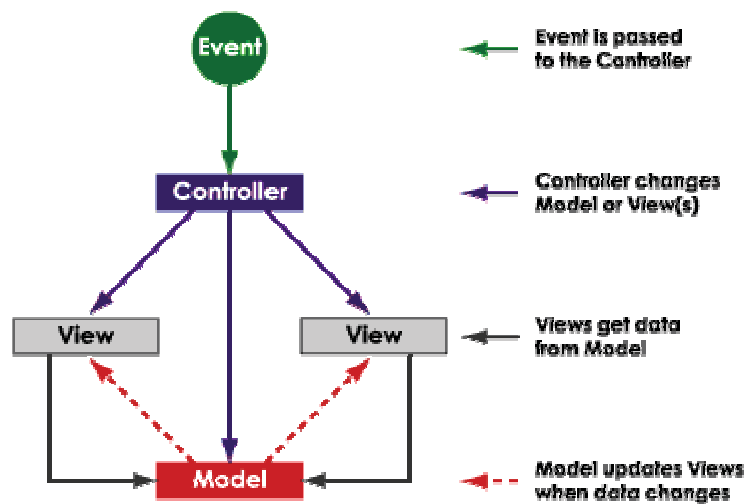


Figure 1 Model-View-Controller Structure (eNode Inc. (2002))

The view component is the interface that the user can see and interact with. It is what the end user will use to interact with Ábaltas' web portal. In a web application, such as Ábaltas', this layer is composed of web pages through a mark-up language interface such as HTML, XML/XSL and more recently web services. The MVC architecture handles the use of different views for an application through view technologies rendering it in a way that can dynamically link it to the application's business-logic and data layer (i.e. the model). (Steve Masover (June 2004))

A widely utilized technology for constructing dynamic web pages is Java Server Pages (JSP). They combine static markup and JSP-specific coding that reference or execute Java. (Steve Masover (June 2004))

According to Brian Kotec (Oct 2002), the model component represents enterprise data and business logic / rules and is where most of the processing occurs when utilizing the MVC design pattern. Databases and object components such as Java Beans fall under the model component. The existence of a database offers data persistence providing long-term storage of data – beyond the scope and time-frame of the user’s session in the application. The data returned by the model is display-neutral and therefore a single model can provide data for any number of display interfaces, shielding other layers (such as the view component) from the model facade.

The controller interprets requests from the user and determines the appropriate component (model or view) to send the request for fulfillment. The sequence of requests to the model component or the view component and the required input from the user, defines the applications’ workflow. Therefore, the controller defines the workflow of the application.

The controller can use numerous frameworks in the MVC architecture. These include the Struts Framework.

Struts Framework

Apache Struts is an open-source framework that is used for building web applications in Java Server Pages. The technology provides the Controller portion of the application. Struts focus on receiving requests from the client (typically a web browser), deciding what business logic function is to be performed, and then delegating responsibility for

producing the next phase of the graphical user interface to an appropriate View component. (Apache Foundation, n.d.)

According to IBM (2004), Struts contributes to the controller component in the MVC design pattern as follows:

- The Struts action servlet handles run-time events as per an agreement with a set of rules that are provided at deployment time. These rules are specified in a Struts configuration file and specify how the servlet responds to every outcome received from the business logic. This provides modularity as modifications to the flow of control require changes only to the configuration file.
- Struts provide the Java class `org.apache.struts.action.Action`, which Java developers can subclass to create an "action class". During run time, the action servlet is said to "execute actions," allowing the servlet to invoke the `execute` method of each of the instantiated action classes. The object returned from the `execute` method manages the direction of the action servlet as to what action or JSP file to access next.

These features greatly enhance the separation the control logic (what to do) with the view logic (how it's rendered). The Struts framework would greatly benefit Ábaltas' web application as it supports presentation implementations such as JSP and model implementations such as a MySQL database and JavaBeans.

The Model-View-Controller enforces the separation of the presentation layer from the data resulting in an application that is easier maintained, structurally organized and self-contained.

Platform Architecture

The platform is the underlying system on which applications can be executed. For Ábalta, it is important that the platform be of minimal cost, platform-independent (it can be executed anywhere) and be relatively stable. For this project, the platforms that were researched and investigated were the .Net platform and the Java platform. This section documents the main features of each platform and measure how they compare for what Ábalta requires.

.Net Platform

According to Roger Sessions (March 2001), the most significant part of the .Net platform is the .Net framework. The framework is a general runtime environment that is closely associated with the underlying operating system. It includes a component-oriented middle-tier infrastructure (COM+), the Common Language Runtime (CLR) environment, a just-in-time compiler and a set of operating system libraries packaged using the .NET component model.

The integrated development environment tool for .NET is Visual Studio.NET. GUI programmers, for the presentation tier, which would include Ábaltas' user interface, use Visual Studio.NET to define the logic that delivers web pages to thin client systems. The

business tier is developed using Visual Studio.NET to implement business logic in a wide variety of languages and package that business logic as COM+ components.

Visual Studio.NET is language neutral. A variety of languages can be plugged into its platform. "Standard" Microsoft languages that will come with VS.NET are VisualBasic, VisualC++, and VisualC#. Additional languages are available through third parties, including COBOL from Fujitsu and Eiffel from Interactive Software Engineering.

VS.NET's language neutrality is critical to the .NET platform strategy and is achieved by translating all Visual Studio.NET languages into a common language called Intermediary Language (IL). Through the creation of an IL translator, language vendors can make their languages compatible with VS.NET. Compatible languages are referred to as a .NET enabled language. (Roger Sessions, March 2001)

J2EE Platform

Roger Sessions (March 2001) found that the J2EE architecture can be divided into five parts:

1. Java language system
2. The client programming model
3. The middle tier infrastructure
4. The programmer enterprise API
5. Non programmer visible API

Java Language System

Java programming language is the language used to write Java applications (including JavaBeans component which would equate to the model component in the MVC architecture). During compilation of a Java program, it is converted to byte codes that are the portable machine language of a CPU architecture known as the Java Virtual Machine (JVM). Although the JVM can be implemented directly in hardware, it is usually implemented as a software program, which interprets and executes byte codes.

The Java platform includes the predefined set of Java classes, which exist on every Java installation and are available for all Java programs to use. The Java platform can also be referred to as the Java runtime environment (JRE) or the core Java APIs (application programming interfaces). The Java platform allows for optional standard extensions using extension APIs, which exist in some but not all Java installations. (Roger Sessions, March 2001)

The Client Programming Model

The J2EE client-programming model focuses on interacting with the browser and has three pieces: Java Applets, Java Servlets, and Java Server Pages. Ábaltá will be designed as a browser based application.

The technologies that manage HTTP requests and HTML responses are Java Servlets and Java Server Pages, which are analogous to ASP.NET (Active Server Pages) in the Microsoft space. (Roger Sessions, March 2001)

Java Server Pages

According to Scott McPherson (2000), Java Server Pages (JSP) technology provide for the creation of dynamic web pages and simplification of the task for building web applications that work with a wide variety of web servers, application servers, browsers and development tools. They enable the creation of dynamic content by reusing predefined components and by interacting with components using server-side scripting.

A JSP page is an HTML web page that contains additional bits of code that execute application logic to generate dynamic content. This application logic may include JavaBeans, JDBC objects, Enterprise Java Beans (EJB), and Remote Method Invocation (RMI) objects, all of which can be easily accessible from a JSP page.

JSP and HTML would equate to the View component in the MVC architecture and would serve to present Ábaltas' application as a browser based web application.

Java Beans

JavaBeans, written in the Java programming language, are a portable, platform-independent component model in the MVC architecture. They enable reusable components to be written once and run anywhere - benefiting from the platform-independent power of Java technology. JavaBeans act as a bridge between proprietary component models and provide a seamless and powerful process for developers to build components that run in ActiveX container applications. (JavaBeans FAQ, n.d.)

The Middle Tier Infrastructure

J2EE's middle tier infrastructure is Enterprise Java Beans (EJB) which is equivalent to the .NET COM+. The EJB specification defines the architecture for both the development and deployment of transactional, distributed object applications-based, server-side software components. (Roger Sessions (March 2001))

The Programmer Enterprise API

The most important parts of the Java Enterprise API are the following:

- Java Database Connection (JDBC) 2.0 - The JDBC API is the industry standard for database-independent connectivity between the Java language and an extensive range of databases – SQL databases and other tabular data sources, such as spreadsheets or flat files. The API provides a call-level API for SQL-based database access, enabling java programs to execute SQL statements and defines how a java programmer can access a database in tabular format from Java code using a set of standard interfaces and classes written in the Java language. (What is JDBC?, n.d.) Ábalta will require a database to act as a persistent store for data. JDBC would be used to provide the connectivity between the model and the database.
 - Java Naming and Directory Interface (JNDI) - The API used to access information in enterprise name and directory services from Java.
 - Java Message Service (JMS) 1.0 - The Java API used for asynchronous workflow.
- (Roger Sessions (March 2001))

Non programmer visible API

The non programmer visible API includes APIs that define how other products can plug into J2EE, such as the Connector API, and APIs that have been effectively superseded by later advancements in the J2EE model, such as the JTA (Java Transaction API). (Roger Sessions (March 2001))

Security

Ábalta will require security for the login / logout, forum and content management system in the web portal application. For this project, the security software researched was Java Authentication and Authorization Service.

According to Sun Developer Network (n.d.), Java Authentication and Authorization Service (JAAS) is a framework, supplemented with the Java platform, through which users who execute their code, can be required to have explicit permission to perform certain operations. JAAS provides a set of APIs that enable services to authenticate and enforce access controls upon users. This typically means that a JAAS-enabled application requires a user to log into it similar to a user logging into their computer (in fact, JAAS often uses the operating system to authenticate the user directly). Additionally JAAS provides a set of classes that authorize users to perform certain operations.

According to BEA (2007), authentication using JAAS and the Security Framework is achieved in the following manner:

1. The client application creates a callback handler (a JAAS standard that allows a number of arguments to be passed as complex objects to a method to retrieve

- specific authentication data, such as usernames) containing a callback that allows a provider to request authentication information from the application.
2. The callback handler is passed by the client application through the authentication service of the Java API into the Security Framework.
 3. The Security Framework presents the callback handler to the LoginModule (responsible for authenticating users within the policy domain and for populating a subject with the necessary principals (users and groups)) for the appropriate authentication provider.
 4. The callback handler is used by the LoginModule to request specific authentication information (e.g. username or password).
 5. The client application is responsible for collecting the appropriate information to respond to the authentication callback, e.g. which may include prompting for a username or password.
 6. When the LoginModule collects all of the required information, it performs one of the following:
 - Authentication success which returns a valid subject
 - Authentication failure which throws a exception (LoginException) (BEA, 2007)

Application Server

An application server is a software platform that provides dynamic data or content to the World Wide Web. It is required by this project as Ábaltas' web portal application will

deliver dynamic content. The application server studied in this project is the Resin Application Server.

Resin Application Server

According to An Overview of Resin (2003), Resin is a high-performing XML application server used with JSPs, servlets, JavaBeans, XML, and a host of other technologies. Over the past number of years, Resin servers have evolved with the needs of its users. As of this writing there are two Resin application servers available. There is a core Resin server and a Resin Enterprise server, which enhances the core product with Enterprise JavaBeans (EJB) container-managed persistence (CMP) and database caching.

Resin is a self-contained application server in addition to a Web server and therefore there is no requirement for any additional server software. The internal Web server is very fast, efficient and, generally, outperforms the competition. Resin also supports external Web servers including: Apache, Netscape, IIS, and O'Reilly WebSite.

Resin application server supports the Java JSP specification. JSP pages can use Java to embed code within the HTML tags and text on the page. The Resin server allows for the creation of custom tag libraries to add better internal documentation to Web pages. Full support is provided for Web applications pulling together JSP pages, scripts, and beans.

Resin supports the use of a hardware load balancer for higher cost situations and, using the Resin server itself, for low-cost solutions. Load balancing works by executing the

LoadBalanceServlet servlet on one front-end machine, and distributing user requests to numerous back-end servers. For system efficiency, the Resin server includes an HTTP proxy cache. (An Overview of Resin (2003))

The Resin server provides security mechanisms that may be used in applications. The security method that may be used with Ábaltá is the security that provides authentication through JdbcAuthenticator, which will ask the MySQL database for the (input) password matching the (input) user's name. (An Overview of Resin (2003))

Databases

A database is a structured collection of data and can be anything ranging from a simple shopping list to a picture gallery or the vast amounts of information in a corporate network. To manipulate data stored in a computer database, a database management system such as MySQL Server is required. Database management systems play a key role in computing, either as standalone utilities, or as part of other applications.

Ábaltá will be utilizing a database for the application as it requires the use of a persistent store of data for its web portal application. The database management system researched in this project was MySQL.

MySQL

According to What is MySQL? (n.d.), MySQL is a relational database management system. A relational database works by storing data in separate tables rather than putting all the data in one big repository, adding speed and flexibility. MySQL (SQL being an

acronym for Structured Query Language), is defined by the ANSI/ISO SQL Standard, and is the most common standardized language used to access databases. Evolving since 1986, the SQL standard has several existing versions.

MySQL is open source meaning that it is possible for anyone to use and modify the software. The MySQL software is available for download from the Internet and can be used for free. The source code may be changed as required. MySQL software utilises the GPL (GNU General Public License), <http://www.fsf.org/licenses/>, to define what may and may not be done with the software in different situations. (What is MySQL? (n.d).)

MySQL database server is an efficient, reliable, and easy to use server. It was originally developed to handle extensive databases much faster than existing solutions and has been successfully used in highly demanding production environments for several years.

Although constantly developing, MySQL server offers a rich and useful set of functions. Its connectivity, speed, and security make MySQL Server highly suited for accessing databases on and over the Internet. (What is MySQL?, n.d.)

MySQL Server performs in client/server or embedded systems. MySQL database software is a client/server system that comprises a multi-threaded SQL server that supports different backends, several different client programs and libraries, administrative tools, and a vast range of APIs. MySQL Server can be provided as an embedded multi-threaded library that can link into an application to get a smaller, faster, easier-to-manage standalone product. (What is MySQL?, n.d.)

Donations

One of the most important features of the web portal application will be to generate funds for Ábalta. There are a number of methods that will allow us to effectively raise funds and are detailed in this section.

www.MyCharity.ie

www.Mycharity.ie is a web application that provides services to charities and fundraisers alike, to make it as relatively easy, simple and safe as possible to raise money for charity.

According to mycharity (n.d.), through their website, charities are offered an online presence and the ability to take donations direct from the public online. In addition charities are offered the ability for fundraisers to create their own unique fundraising page for any event, and to raise money for charity.

Once registered with MyCharity.ie, fundraisers are provided with their own fundraising page that can be customised to their own liking and their page allows them to fundraise for charity without the need for keeping lists of sponsors, collecting and checking money, writing cheques etc. Everything happens online in a safe and secure environment.

	<i>Cost</i>
Setup Fee	€500
Per Donation	3% per Donation

Table 1 Cost of MyCharity.ie

Realex

According to Realex Payments (n.d.), Realex Payments is one of Europe's leading payment service providers. Based in Dublin, Ireland, the company has processed millions of euros worth of donations for dozens of Ireland's leading charities over the last six years. Using Realex Payments allows charities to accept donations from web users at any time and anywhere.

Using Realex Payments, donors can easily access client websites / web portals and donate, by credit card or debit card in real time. Before charities can process credit card payments in real time, they are required to have a merchant services agreement with their bank. A redirect integration option is available that will link a charity's web portal with Realex's Payment Gateway.

Redirect Integration

The charity does not require a secure server and can be hosted anywhere and on any platform with the only requirement being that CGI scripts are enabled. The charity is not responsible for collecting the card details and will not have access to them. This ensures that the charity is not liable for any card fraud that may occur.

When a user donates, the web portal page is redirected to a RealAuth application on a secure Realex Payments server by a script that is running on the charity's web portal. Realex is responsible for hosting the secure, charity-branded web page. Realex collects the card details, processes the payment and the transaction results are returned to the charity's web portal via a script on the web portal. (Realex Payments, n.d.)

	<i>Cost</i>
Setup Fee	€150
Monthly Service Fee	€25
Per Transaction	0 if transaction < 100 per month 25 cent thereafter

Table 2 Cost of Realex

JFreeChart

It would be optimal to provide a method of determining how much has been raised in fundraising through the web portal application. One method is to provide a chart that will display the funds raised.

JFreeChart is an open-source Java chart library. It supports many forms of charts including pie charts, bar charts (horizontal and vertical, regular and stacked), line charts, Gantt charts, and thermometers. JFreeChart can be used in applications, applets, servlets and JSP to generate charts. (JFreeChart 2005-2007)

A thermometer plot displays a single value (from a ValueDataset) in a thermometer type display. “This plot supports a number of options:

1. Three sub-ranges which could be viewed as 'Normal', 'Warning' and 'Critical' ranges.
2. The thermometer can be run in two modes:

- fixed range, or
 - range adjusts to current sub-range.
3. Settable units to be displayed.
 4. Settable display location for the value text.” (Class ThermometerPlot, n.d., <http://www.jfree.org/jfreechart/api/javadoc/org/jfree/chart/plot/ThermometerPlot.html>)

HTML Editor

The forum feature and content management system feature in Ábaltas’ web portal application will require the use of a HTML editor for the correct formatting of content.

TinyMCE

According to TinyMCE (n.d.), TinyMCE is a platform-independent web-based Javascript, HTML, What You See Is What You Get, editor control. It is released as open source under LGPL by Moxiecode Systems AB. It has the ability to convert HTML text-area fields to editor instances. TinyMCE can be easily integrated into features such as content management systems and forums.

To incorporate TinyMCE , the browser must be compatible with browsers which include Internet Explorer, Mozilla, and FireFox. As it is a JavaScript application, JavaScript needs to be turned on.

TinyMCE is a zipped file and can be downloaded from the TinyMCE website. Using WinZip, the files can be extracted into the desired location. Once the archive has been

extracted, the web page will need be to edited to include the configuration and javascript for TinyMCE. The following code will need to be included in the application.

```
<script language="javascript" type="text/javascript"  
src="../jscripts/tiny_mce/tiny_mce.js"></script>
```

```
<script language="javascript" type="text/javascript">
```

```
tinyMCE.init({  
    mode : "textareas"  
}); (TinyMCE:Installation, August 2007)
```

TinyMCE will hugely enhance the users' experience with Ábaltas' web portal application as it will provide an easy-to-use editor with a powerful editor control, for the forum and content management system.

Build Framework

This project will need to be deployed by building all the source files into a web application archive (WAR) file and uploading this file to the host server. One method for building the source files into a WAR file is achieved by ANT (Another Neat Tool).

ANT

ANT is an open source Java-based build tool available from Apache. It works to build and automate the building of a software project in anticipation of deployment. ANT utilizes a build.xml file to “build” the project. The build.xml file is a basic, well-formed XML file that contains one (or more) project and at the very least one default target – a target is a set of tasks that are to be executed. (A Little Ant Can Lighten the Load, 2007)

The Known and Unknowns about the Project

There were many unknowns about design and development of the web portal for Ábalta during the initial stages. One of the more concerning unknowns was regarding the technology and software to use and develop with and it was fundamental that various technologies be researched and investigated to find the most appropriate.

The functionality of the web portal was indefinite. It was unclear on what the web portal would provide to the end user. Was a login to the portal necessary? Need it be secured?

What was known about the project was the fact that the content should be updateable through the web portal. This meant that the web portal should provide a means of a content management system in order for an end user to be able to add, edit and delete content in the web portal.

It was also necessary that the web portal provide essential information and support regarding autism to the end user while at the same time, generating awareness for the disability. An added bonus would be the functionality of a donation system whereby people could donate money to the charity online.

Finally as it is a charity school, it was imperative that costs for the design and development of the web portal be kept to a minimum.

Contribution of the Project to the Field

This project will endorse and utilize modern technologies in achieving a web portal application for Ábalta. It will provide a detailed and comprehensive account of different and corresponding technologies and compare and contrast them. It will attempt to provide an unbiased recommendation for the best and most applicable technologies to be used when developing a web portal.

This project will offer as a model for any other charity to implement a web portal to provide information and support of its services. It will be designed in such a way that it can be easily utilized by any other charity in a “plug and play” mode.

It will provide Ábalta with a modern and more sophisticated system for distributing information and support on autism while, simultaneously, raising money and awareness for the charity.

Chapter III Methodology

The design and development of Ábaltas' web portal application was accomplished adhering to an agile methodology which combines a philosophy and a set of development guidelines. The agile methodology used was extreme programming which is one of the more prominent ones. This chapter serves to portray the design methodology used during the life cycle of the project and outline the deliverables to be presented upon completion.

Formats for Presenting Results / Deliverables

The deliverables for this project will include a fully functional and operational web portal application for Ábalta School for Children with Autism. This web portal will provide information on the services that Ábalta provide a support system for people and interested parties affected by autism, a facility for donating funds over the internet, a content management system and a subsystem for the storage and manipulation of student details and records.

The deliverables for this system will be broken into features and releases. These releases will be delivered incrementally throughout the schedule of the project life cycle.

Life Cycle Models / Methodology

According to Roger S. Pressman (2005), a process model defines a distinct set of activities, actions, tasks, milestones and work products that are required in order to engineer high-quality software. It is important to implement these models as they provide stability, control and organization to an activity that has the potential to become quite

chaotic. The process guides an engineer or a team of engineers through a set of framework activities that are organized into a process flow.

Agile Process Model

During the design and development of Ábaltas' web portal application, the agile software development methodology was followed providing the project with a proven set of life cycle guidelines. According to Roger S. Pressman (2005), agile software engineering combines a philosophy and a set of development guidelines. The philosophy encourages customer satisfaction and early incremental delivery of software; small, highly motivated project teams; informal methods; minimal software engineering work products and overall development simplicity. Agile software processes are characterised by three key assumptions about the majority of software projects:

- The difficulty in predicting which software requirements will persist and which will be modified. Similarly, it is difficult to predict how customer priorities will change as a project progresses.
- Design and construction are interleaved, for many types of software. It is recommended that both these activities be performed in tandem so that design models are proven as they are created. It is often difficult to predict how much design is necessary, before construction is used to prove the design.
- The activities analysis, design, construction and testing are not as predictable (from a planning point of view).

The fundamental responsibility of an agile process is to be adaptable. This was fundamental during the life cycle of the project as feature creep became an issue. For progression, an agile software process must adapt incrementally. To adapt incrementally,

an agile team requires customer feedback and an effective channel for customer feedback is either an operational prototype or a portion of an operational system. Software increments (executable prototypes or an operational system portion) should be delivered in short time periods so that adaptation keeps pace with change. This iterative approach enables regular customer evaluation of the software increment, providing necessary feedback to the software team, and influencing the process adaptations that are made to accommodate the feedback.

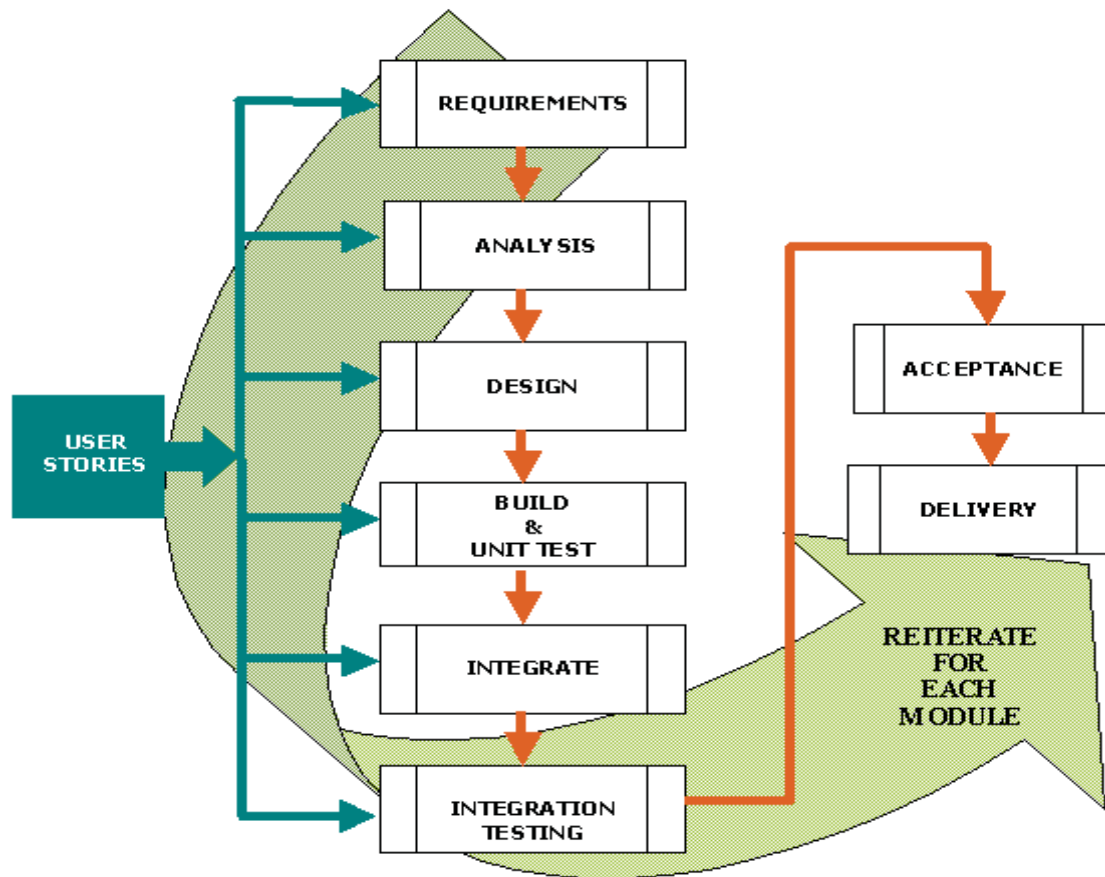


Figure 2 Agile Process Model (Software Engineering Methodology (2003))

Extreme Programming (XP)

Ábaltas' web portal application utilized the extreme programming (XP) methodology, which is one of the most prominent existing agile methodologies. According to Don Wells (2006), XP is a proven deliberate and disciplined approach to software design and development. It is successful because it emphasizes customer satisfaction. The methodology is designed to deliver the software customers need when it is needed. XP empowers developers to confidently respond to changing customer requirements, even when changes are required late in the life cycle.

XP improves a software project in four important ways; communication, simplicity, feedback, and courage. XP engineers communicate with their customers and fellow engineers. Their design is kept simple and clean. They get feedback by testing the software starting from day one. The system is delivered to the customers as early as possible and changes are implemented as suggested. With this foundation XP engineers can courageously respond to changing requirements and technology.

XP is similar to a jigsaw puzzle. There are many small pieces which individually make no sense, but when combined together a complete picture can be seen.

XP uses an object oriented approach as its development paradigm and encompasses a defined set of rules and practices that occur within the context of four framework activities: planning, design, coding and testing.



Extreme Programming Project

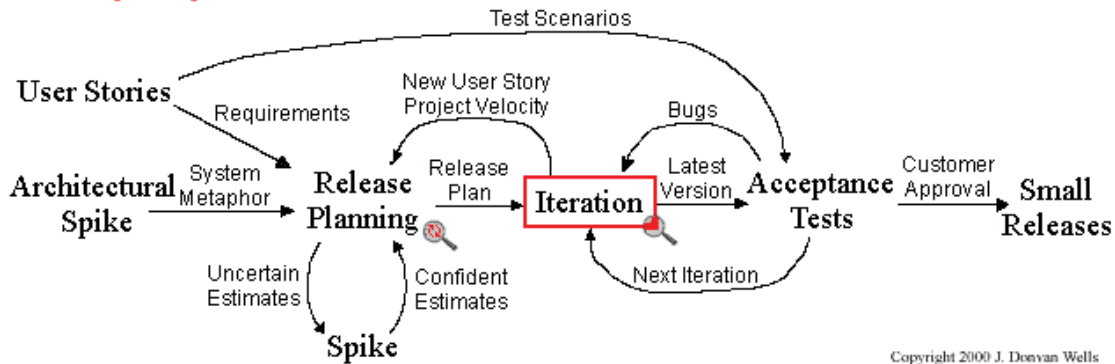


Figure 3 Extreme Programming (XP)(Don Wells (2006))

Planning

The planning activity begins with the creation of a set of stories (user stories) that describe features and functionality for software to be built. These user stories serve the same purpose as use cases but are quite different. User Stories are written by the customers as requirements that the system needs to have. They are similar to usage scenarios, except that they are not just limited to describing a graphical user interface. They are in the format of approximately three sentences of text written by the customer in their own terminology without “techno-syntax”. User stories should only provide enough detail to make a reasonably low risk estimate of the time required to implement the story. When implementing the story, developers will meet with the customer and receive a detailed description of the requirements face-to-face. (Don Wells, 2006)

Customers are extremely involved in each aspect of the project. They and the XP team of developers work together to determine how to group stories into the next release (increment) to be developed. As development work and the project proceeds, the customer can add stories, modify an existing story, split stories or eliminate them completely. The XP team then reconsiders all remaining releases and changes its plans accordingly.

User Stories

During the initial meeting with the directors and principle of Ábalta School, the following user stories were identified and submitted.

- Internet

Ábalta school information must be available on the web for public access. The domain name of the site must be obtained from the department of the higher education authority, and Ábalta must be recognized as an educational facility

- Ábalta School Online

The web portal must distribute information regarding Ábalta School, its history and its mission statements. It is important that Ábalta conveys the importance of providing the highest quality of education to its students with autism.

- Autism Information and ABA Methodology

The web portal must contain and distribute information and support regarding autism.

Ábalta uses Applied Behaviour Analysis (ABA) methodology for educating students and it is important that information regarding this practise is documented for public access.

Links should be provided to other autism organizations.

- Email

The web portal must provide email accounts for the directors, principle, teachers and employees of Ábalta School.

- Forum

It is important that there is a forum where discussions and communication can occur.

This will enable a support network to be built up, documenting questions, real-life experiences and offering comfort and support between people.

- Donations

Ideally, the web portal will be able to accept donations from people wishing to do so. It is important that funds will be raised for the school and the services it provides.

- Content Management System

The web portal must be updatable. Optimally, the web portal will be able to be edited, dynamically, by an administrator / principle of the school. The administrator / principle will not have any IT / Software experience so therefore updating must be low maintenance.

- Students

Student records and details should be stored in the database with access to them through the web portal. They are currently held on file in the offices but this is manual system, of which is time consuming. The web portal should provide a system for storing student records in a secure environment which can be updated and modified on demand. These records must be password protected.

- Gallery

The web portal should have a gallery of photographs that will present photographs from fundraisers and events that have occurred.

Release Plan

When user stories have been written, a release planning meeting is setup in order to create a release plan. The release plan specifies which user stories will be implemented for each system release and dates for those releases.

The user stories for Ábalta were reviewed and the following time estimates had been calculated. These calculations are based on the fact that there is one software developer designing and developing the web portal.

User Stories	Work Involved	Time Estimate
Internet	<ul style="list-style-type: none">• Research web hosting companies and acquire hosting for the web portal• Contact the Dept. of Higher Education Authority and seek domain name.• Register domain name	1 week
Ábalta School Online	<ul style="list-style-type: none">• Research appropriate technologies for web portal implementation – database, web application server, architecture framework• Design and develop framework	3 weeks

	<p>architecture</p> <ul style="list-style-type: none"> • Design and create database for data storage • Graphical user interface design for web portal application • Design and implement the Gallery • Create prototype for approval 	
Autism Information and ABA Methodology	<ul style="list-style-type: none"> • Obtain information regarding autism and ABA for upload • Design and create database tables • Develop code for obtaining data from the database 	2 weeks
Email	<ul style="list-style-type: none"> • Obtain list of proposed email addresses from Ábalta • Through the hosting company, create email account for each user 	3 days
Forum	<ul style="list-style-type: none"> • Research best solution for forum • Design and create database tables • Design and implement Login / Logout security feature • Design and implement forum architecture 	4 weeks

	<ul style="list-style-type: none"> • Design and implement GUI for forum 	
Donations	<ul style="list-style-type: none"> • Research most appropriate method for submitting donations • Design and implement donation architecture • Design and implement visual chart for displaying funds raised through online donations 	4 weeks
Content Management System	<ul style="list-style-type: none"> • Research existing content management systems available • Design and create database tables • Design and implement system architecture 	4 weeks
Students System	<ul style="list-style-type: none"> • Design and create database tables for storage of student records • Design and implement GUI for student system • Design and implement system architecture 	4 weeks
Testing	<ul style="list-style-type: none"> • Testing of system at all stages of development 	4 weeks

Table 3 User Stories

The user stories are grouped into releases as follows.

Release 1:

- Internet
- Ábalta School Online
- Autism Information and ABA Methodology
- Email
- Gallery

Release 2:

- Forum

Release 3:

- Donations
- Donation Chart Gauge

Release 4:

- Content Management System

Release 5:

- Student System

Release 6:

- Testing

time for implementing that functionality when it becomes the most important story in the release plan.

Each of the releases for Ábalta was seen as an iteration. The iterations were classed and delivered in the following order:

[illegible]

Figure 5 Iteration Schedule

During the iterations, meetings were continuously setup and attended to communicate problems, solutions and promote focus.

Design

The design phase meticulously abides by the “keep it simple” principle. A simple design and representation is always preferred over a more complex representation. (Roger S. Pressman, 2005) The design is viewed as a transient artifact that can and should be continually modified and changed as development proceeds.

MVC Design Model

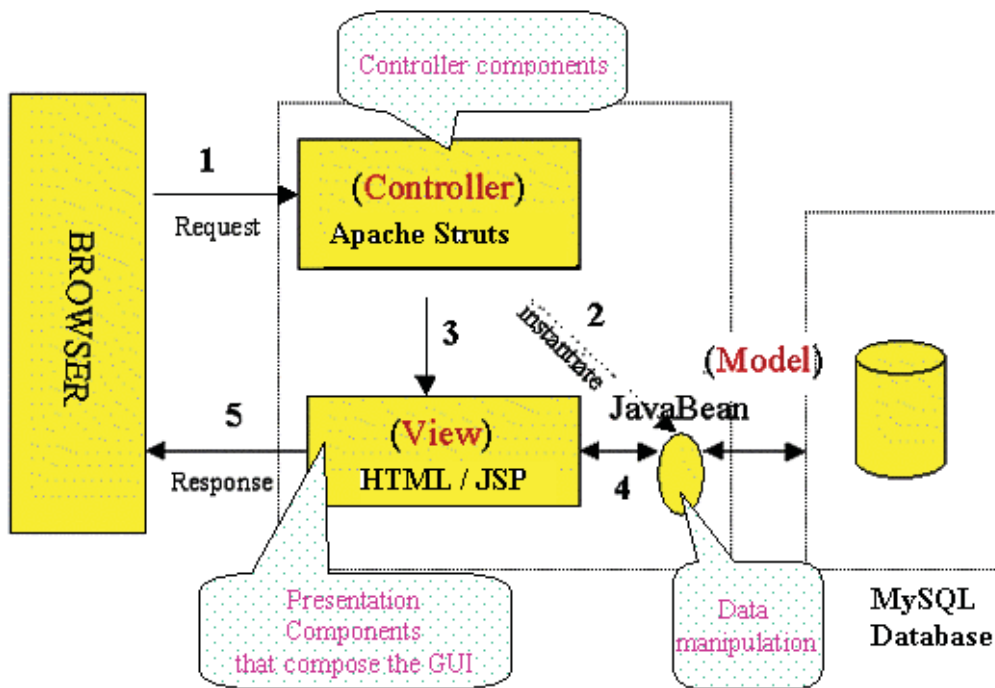


Figure 6 MVC Design Model (JSP Architecture Overview, n.d.)

View

The View renders the model into a form suitable for interaction, most typically a user interface element. (Model-View-Controller (MVC) with JavaScript (2006)). Ábalta's web portal user interface is through a web browser where the user can request an action. The View model will be designed in HTML and JSP allowing for dynamic web pages to be created.

Controller

The Controller processes and responds to events and actions, most typically user actions, and invokes changes on the model and sometimes the view component. (Model-View-

Controller (MVC) with JavaScript (2006)) Ábalta's web portal controller is implemented with Apache Struts. According to Struts and model-view-controller design pattern (2004), the Apache Struts provides double contribution to the view component:

- Struts provide the Java class `org.apache.struts.action.ActionForm`, in which Java developers can subclass to create a form bean. During run time, the bean is used in two ways:
 - As a JSP page prepares the related HTML page/form for display, the JSP page accesses the java bean, which holds values to be placed into the page / form. Such values are provided from either the business logic or from previous user input.
 - When user input is returned from a Web browser, the java bean can validate and hold that input either for use by business logic or (if validation failed) for subsequent redisplay or error messages.
- Struts provides numerous, custom JSP tags that are simple to use but are powerful in the sense that they hide information thus ensuring modularity. The GUI designer does not need to know much about form beans, for example, beyond the bean names and the names of each field in a given bean.

Struts contribute to the controller component as follows:

- The Struts action servlet handles run-time events as per an agreement with a set of rules that are provided at deployment time. These rules are specified in a Struts configuration file and specify how the servlet responds to every outcome received from the business logic. This provides modularity as modifications to the flow of control require changes only to the configuration file.

- Struts provide the Java class `org.apache.struts.action.Action`, which Java developers can subclass to create an "action class". During run time, the action servlet is said to "execute actions," allowing the servlet to invoke the `execute` method of each of the instantiated action classes. The object returned from the `execute` method manages the direction of the action servlet as to what action or JSP file to access next.

Model

According to the Apache Software Foundation (2007), the model component represents enterprise data and business logic / rules and is where most of the processing occurs when utilizing the MVC design pattern. It can usually be divided into two major subsystems – the internal state of the application/ system and the actions that can be taken to modify that state. Databases and object components such as Java Beans fall under the model component. Ábalta's web portal model is implemented with Java Beans and a MySQL database.

Many web applications represent the internal state of the system as a set of one or more Java Beans. The bean properties represent the features and details of the systems' state. Depending on an applications' complexity, these beans may be self contained (and therefore know how to persist their own state), or they may be facades that can retrieve the system's state from another component such as a database, a search engine, an Entity Enterprise JavaBean, or something else entirely.

MySQL is a Structured Query Language-based relational DBMS that runs under a vast array of operating systems making it portable. It is also open-source making it cost effective for a charity. This database provides data persistence for long term storage of data – beyond the scope and time-frame of the user’s session in the web portal. (What is MySQL, n.d.)

Coding

According to Don Wells (2006) a requirement of XP is that the customer be readily available, not only to help the development team, but to be a part of it too. It is recommended that all phases of an XP project require regular communication with the customer, preferably face to face.

Developers must integrate and release code into the code repository whenever possible. It is not recommended to hold onto changes for more than a day. The process of continuous integration helps to avoid diverging or fragmented development efforts, where developers are not communicating with each other about what code can be re-used, or what could be shared. Continuous integration ensures that everyone works with the latest version.

During the coding phases of Ábalta’s web portal, there were constant and continuous weekly meetings with the principle of the school to ensure that the direction of the application was correct. During these meetings, any release made would be tested and verified. Feedback was given and user stories were revised and modified as appropriate.

Coding File System

Using the XP methodology, the coding phase was broken into iterations and released as soon as it was tested and verified.

Iteration 1: File and Class System for Informational Content for Ábalta Online

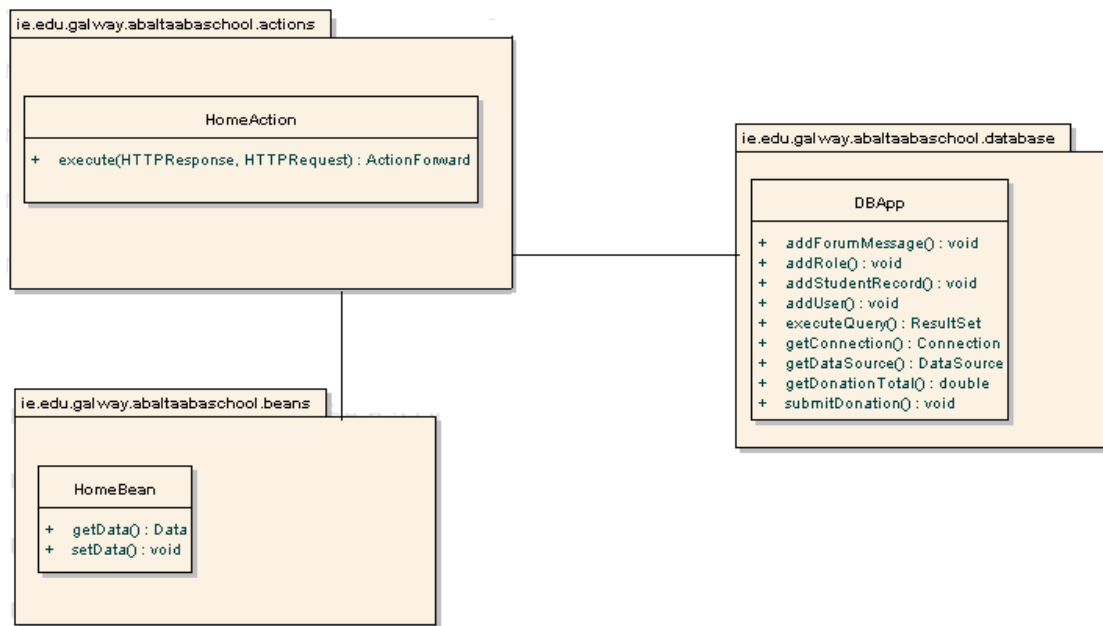


Figure 7 Class File system for Informational Content for Ábalta

Iteration 2: File and Class System for Forum

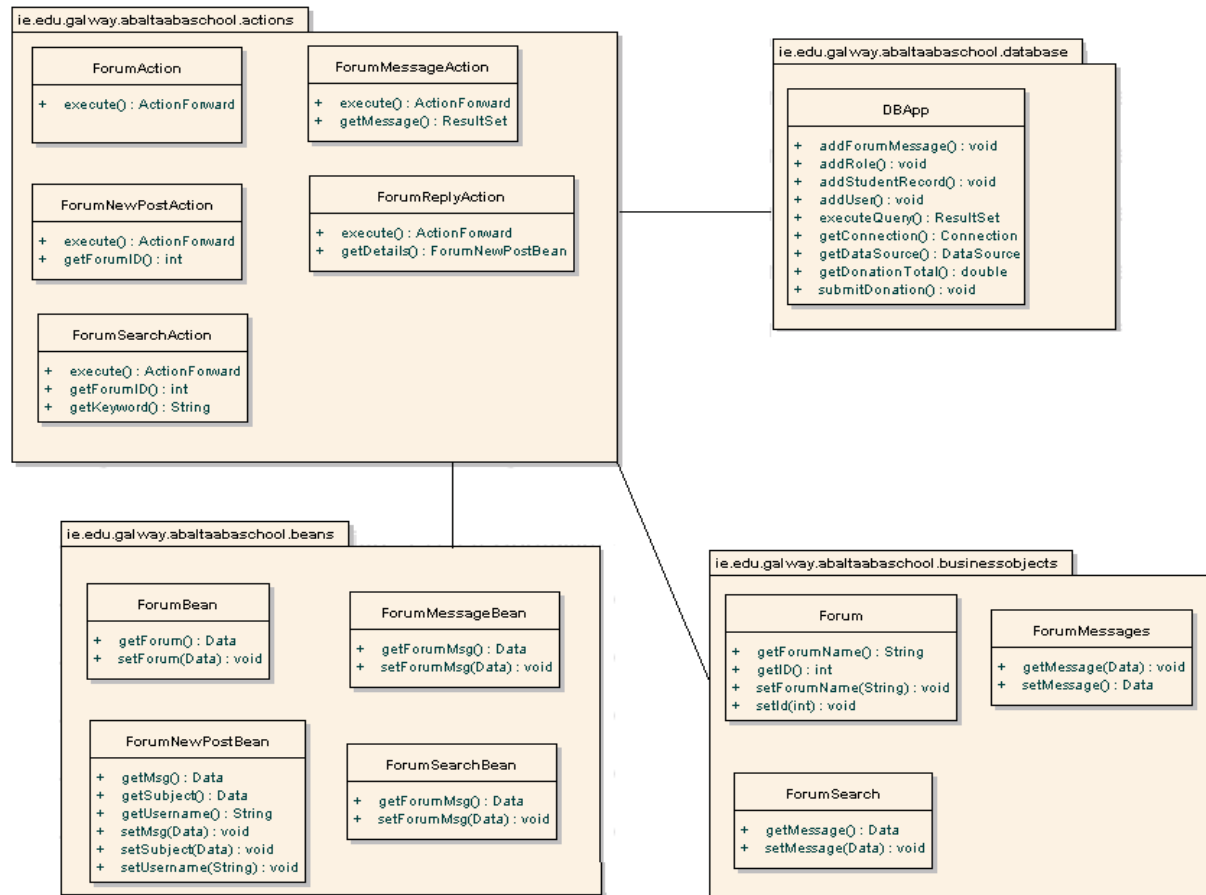


Figure 8 Class File system for Forum for Ábalta

Iteration 3: File and Class System for Donations for Ábalta

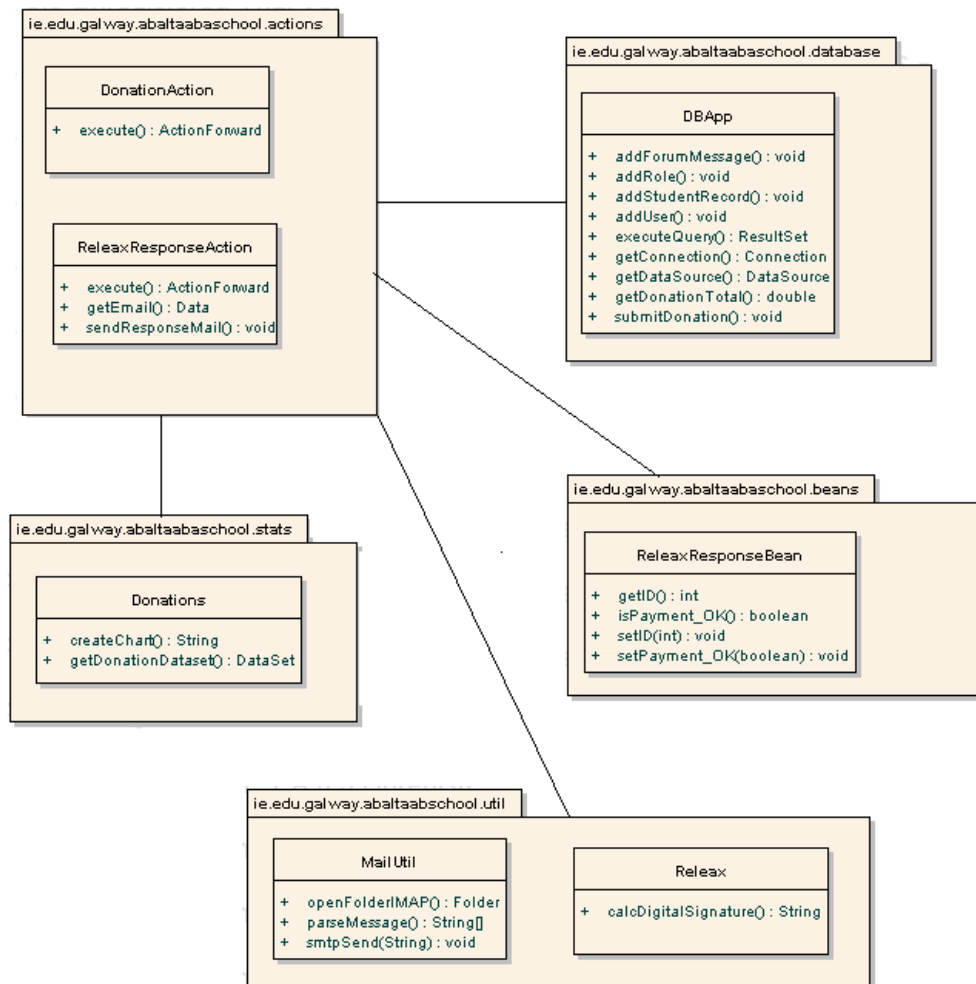


Figure 9 Class File system for Donations for Ábalta

Iteration 4: File and Class System for Content Management System

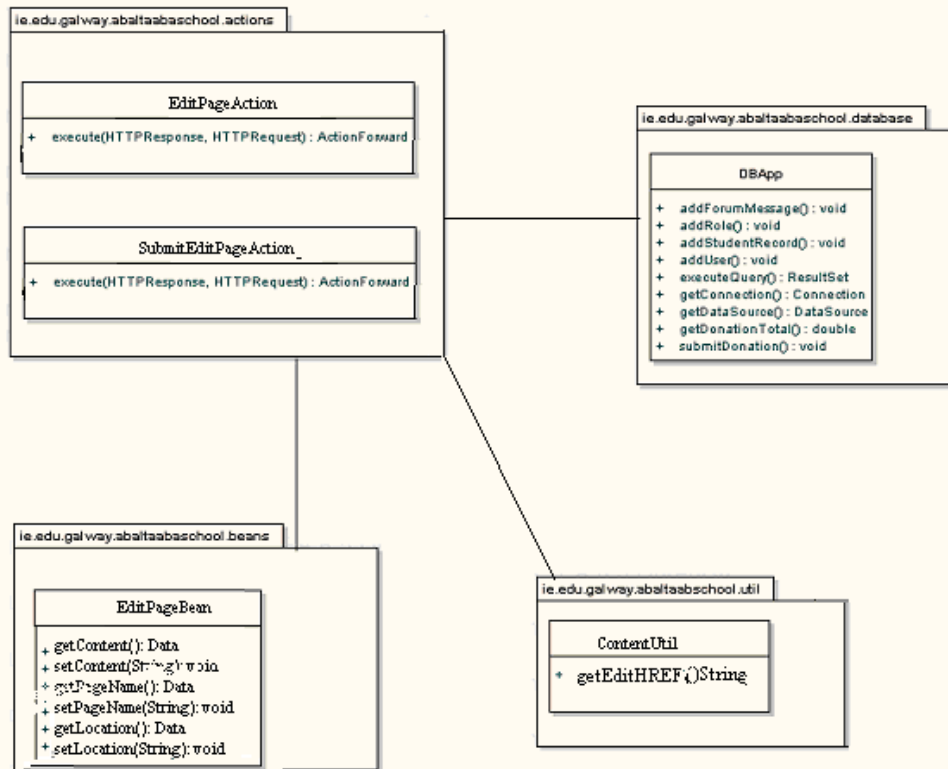


Figure 10 Class File system for Content Management System for Ábalta

Testing

XP acceptance tests are specified by the customer and focus on the overall system features and functionality that are visible and reviewed by the customer. They are created from user stories, and during an iteration the user stories selected during the iteration-planning meeting will be translated into acceptance tests. The customer, along with the developer specifies scenarios to test when a user story has been correctly implemented. A user story may have one or many acceptance tests, in order to ensure the correct functionality.

Acceptance tests are black box system tests and each acceptance test represents an expected result from the system or application. Customers are responsible for verifying the correctness and suitability of the acceptance tests and reviewing test scores to decide which failed tests are of highest priority. Acceptance tests can also be used as regression tests prior to a production release. (Don Wells (2006))

During and at the end of each iteration in the project, acceptance tests were completed by the directors and principle of Ábalta School. These acceptance tests involved working through the functionality of the web portal and verifying that everything works as it is supposed to. When the directors and principle were satisfied that functionality was correct, the iteration would be prepared for release. This involved building all the code into a WAR (Web Application Archive) file and uploading the file unto the hosting server, where it is deployed.

Resource Requirements

The following are resources that are required for the successful implementation and deployment of Ábalta's web portal.

- Web Hosting

The World Wide Web is a massive collection of web sites / portals all hosted on web servers i.e. computers. All HTML files, graphics, etc. reside on a web server known as

the web host. Web hosting clients upload their web files to a shared (or dedicated) web-server, which makes it publicly available.

Ábalta uses Hosting Ireland for their web hosting. Hosting Ireland provided a number of technologies that were needed for the successful implementation and deployment of Ábaltas' web portal. These technologies include J2EE technologies, a MySQL database, Resin application server and mailboxes that will allow for the creation of email addresses. The company worked with the Dept of Higher Education Authority (Ireland) to set up the domain name for the web portal www.abaltaabaschool.galway.edu.ie. In addition, Hosting Ireland provided an alias domain name for Ábalta that would make it easier to find in web searches www.abaltaschool.ie. It provided this alias name free of charge.

Total cost of web hosting resource including tax: €181.50 per annum

- Application Server

Hosting Ireland provides a Resin Application Server inclusive in its fee. The Resin application server provides an environment for J2EE technologies to run in cooperation with a web server.

- Database

Hosting Ireland provides a MySQL database. MySQL is a proven, open source database that will provide more than enough space to store data.

- WAR file

J2EE web applications can be deployed onto an application server and are comprised of all related technologies. They are built into a web application archive (WAR) file to enable the management of it as one unit. Ábaltas' WAR file includes the following technologies:

- HTML
- JSP
- Apache Struts
- Java Code
- JavaBeans

The WAR file is build using the ANT build tool and is then uploaded to Hosting Ireland for deployment on their web application server.

Review of Deliverables

Deliverables for this project are as follows:

- Informational Content for Ábalta School
 - “About ABA” – the education methodology they use to teach students afflicted with autism
 - “Mission Statements” – which explains the core purpose of the charity and what they are trying to achieve

- “Testimonials” – information from the parents of the children being educated in Ábalta School and how the methodologies it uses helps and benefits their children
- “Literature” – further information and readings on autism
- “Fundraising / Events” – information on current fundraising and events to promote the awareness of Ábalta and to generate funds for the school
- “FAQ” – prepared frequently asked questions
- “Forum” - to enable discussions / questions between parents, teachers and interested parties
- “Make a Donation” – a method for providing the ability for internet / web users to donate money to the school from the internet
- “Donation Gauge” – to visually provide a means to measure the amount of funds raised through online donations
- “Content Administration / Management System” – allowing an administrator to dynamically update content on the web portal.
- “Student System” – for the storage and manipulation of student data and records
- “Login / Logout” – allowing for the logging in and out of users of the web portal application
- “Gallery” – provide for the displaying of photographs from fundraising events and the school onto the internet.

Outcomes

Development for Ábaltas' web portal began September 2006. The first task undertaken was to meet with the directors and principle of the school. This meeting served to analyze their user stories and gauge the requirements of the system. A feasibility analysis then ensued to ensure that all user stories could be developed and implemented.

The second task involved investigating web hosting companies and deciding on a domain name. It became apparent that Hosting Ireland was the best choice as it provided the technology that was required at a cost effective price. The domain name was provided by the Dept. of Higher Education Authority. We sought that the domain name include ábalta, aba, and school and the Dept. of Higher Education Authority gave us www.abaltaabaschool.galway.edu.ie. One stipulation was that the area that the school resided be included in the domain name thus the inclusion of “galway” in the domain name.

Once the hosting was set up, email addresses for all interested parties were created, verified and tested. These became immediately operational.

A graphical user interface was designed and prototyped for the directors to review and communicate feedback. This was developed using HTML and JSP. Appropriate feedback was given, acted upon and the design of the web portal evolved until all parties were happy with the outcome.

The MySQL database was designed and implemented. This allowed for the development of dynamic content for the web portal, and kept with the design of the model-view-controller design pattern.

The architecture framework was designed, prototyped, tested, and implemented. This provided the structure for the entire web portal application. Once the model-view-controller framework was in place, coding for the packages, classes, and beans began.

The forum was designed, tested and implemented. This involved much research and prototyping, including utilizing a HTML editor. The forum required that in order to post a message, it was necessary to login to the system. A registration form was designed and a login / logout facility was implemented to enable the public to register and actively participate in the forum. In order to post a message, a HTML editor was implemented. The editor implemented is an open-source editor called TinyMCE. This editor provides the ability to convert the text / post, entered by the user into the text-area, into html code for saving in the database. The forum was a major milestone in the project.

The donations feature involved working with a third party company called Realex Payments to provide a secure server for accepting credit card details over the internet. It also required that a merchant bank account be set up where all monies donated would be transferred to this special account. Working closely with Realex Payments and the bank, along with the treasurer of the Ábalta charity, this feature was designed, developed, tested rigorously and implemented enabling fund generation, online, for the charity. In addition

to this, a donation gauge was implemented in the form of a thermometer. The thermometer chart was developed using JFreeChart, which is an open-source chart library. This displays a visual measure of the amount of funds that have been donated to Ábaltas through the web portal.

The content management system was a feature that was very important to the daily running of the web portal. As there is no dedicated Information Technology specialist employed by the charity, it was important that updates to the web portal be easy to achieve and maintain. A content management system allowed for the simple updating of the web portal with relative ease and minimum knowledge of web development. The content management system is only available to the administrator of the system and requires that the administrator login to the web portal before use. Upon successful login, an edit facility for the content of the web portal is available to the administrator. The TinyMCE HTML editor was also implemented in the content management system to convert the text, entered by the user, into HTML for saving in the database.

Currently there is a manual system for storing student records. A computerized student system would help alleviate the manual effort and automate the storage of student details. A student system would provide for a more efficient method of manipulating student documentation, while being less time consuming and leaving the process less open to human error. This feature will be released in the next iteration of the web portal application.

Chapter IV Frameworks Architecture

There were a number of frameworks manipulated during the design and development of Ábaltas' web portal application. These frameworks incorporate the design pattern framework, the security framework and the system operations / build framework.

The design pattern framework implemented the Model-View-Controller architecture. The security framework employed the Java Authentication and Authorization Service (JAAS) architecture and the build framework utilized the ANT tool.

Model-View-Controller (MVC) framework

The MVC architecture is a web application infrastructure model that decouples the graphical user interface from the web application functionality and informational content.

Controller

According to the Apache Software Foundation (2007), the controller controls access to the view and the model components and co-ordinates the flow of data between them. For Ábaltas, the Struts framework was used as the controller of the web application. Struts focus on receiving requests from the client (web browser), deciding what business logic function is to be performed, and then delegating responsibility to an appropriate View component for producing the next phase of the user interface. The primary component of the Struts framework is a servlet of class `ActionServlet` and is configured by defining a set of `ActionMappings`. An `ActionMapping` defines a path that corresponds with the URI of the incoming request and usually specifies the fully qualified class name of an `Action` class. All `Actions` are subclassed / extended from the package

org.apache.struts.action.Action. Actions encapsulate calls to classes known as business logic classes, they interpret the outcome, and ultimately dispatch control to the appropriate View component to generate the response. While Struts dispatch to a View, actually rendering the View is outside its scope.

When initialized, the controller parses a configuration file (known as the struts-config.xml) and utilizes it to deploy other control layer objects. Together, these objects form what is known as the Struts Configuration.

Struts consult the ActionMappings (defined in the struts-config.xml) as it routes HTTP requests to other components in the framework. Requests may be forwarded to JavaServer Pages or Action subclasses available. Typically, a request is first forwarded to an Action and afterward to a JSP. The action mappings help the controller turn HTTP requests into application actions.

An individual ActionMapping usually contains a number of properties including:

- a request path (or "URI")
- the object type (Action subclass) to act upon the request
- any other properties required

```

<forward name="failure" path="/forum/forumNewPost" />
<forward name="cancel" path="/forum/forum.do" />
</action>

<action
  path="/forum/forumSearch"
  forward="/forum/forumSearch.jsp"/>

<action path="/submitForumSearch"
  type="ie.edu.galway.abaltaabaschool.actions.ForumSearchAction"
  name="forumSearch"
  scope="request"
  validate="true"
  input="/forum/forumSearch.jsp">
  <set-property property="cancellable" value="true"/>
  <forward name="success" path="/forum/forumSearchResults.jsp"/>
  <forward name="invalid" path="/forum/forumSearch.jsp"/>
  <forward name="cancel" path="/forum/forum.do" />
</action>

```

If action successful, forward to the Search Results page

View component that provides facility for user to enter search keyword

Model component i.e. JavaBean to forward the request for processing

If unsuccessful forward to search input page again for re-entry of keyword

Figure 11 ActionMapping from struts-config.xml for Ábaltas' forum search facility

School for Children with Autism

Home
About ABA
Mission Statements
Testimonials
Literature
FAQ
Fundraising/Events
Make A Donation
Gallery
Contact Us

Search Forum

Search for Keyword

Refine Search (Optional)

After Date ☐ 1 Jan 2007
Before Date ☐ 1 Jan 2007

Subject Only ☐
Message Only ☐
User Only ☐

Donations

2,000
1,500
1,000
500
0

Figure 12 UI for forum search

Ábaltas required that users register themselves in the system when they would like to actively participate in the forum. This registration would be carried out through input

forms in the web portal. The Struts framework provides a component called `ActionFormBeans` [`org.apache.struts.action.ActionFormBeans`] for the purpose of input forms. These `ActionFormBeans` are a collection of descriptor objects that are used to create instances of the `ActionForm` objects at runtime, making it easy to store and validate the data for input forms. When a mapping requires an `ActionForm`, the servlet looks up the form-bean descriptor by its name and then uses it to create an `ActionForm` instance of the specified type. (Struts Introduction, 2007)

The following is the sequence of events that occur when a request calls for a mapping that uses an `ActionForm`:

- The Struts servlet either retrieves or creates the `ActionForm` bean instance.
- The Struts servlet passes the bean to the Action object.
- If the request is being used to submit an input page, the Action object in the Model component can check the data. If necessary (maybe because of errors on input), the data can be returned to the input form along with a list of (error) messages to display on the page. Otherwise the data can be passed along to the Model component. (Struts Introduction, 2007)

```
<form-bean name="RegisterForm"
  type="org.apache.struts.action.DynaActionForm">
  <form-property name="name" type="java.lang.String" />
  <form-property name="address" type="java.lang.String" />
  <form-property name="phone" type="java.lang.String" />
  <form-property name="email" type="java.lang.String" />
  <form-property name="username" type="java.lang.String" />
  <form-property name="password" type="java.lang.String" />
  <form-property name="cpassword" type="java.lang.String" />
</form-bean>
```

Ábaltas' input form will hold text boxes for all properties specified here. The type indicates what is expected from the user. This allows for validation in the Model component

Figure 13 `ActionFormBean` from `struts-config.xml` for Ábaltas' user registration facility

School for Children with Autism

Registration

Please enter details

- Please enter a valid forum username

Name: Cathy Talbot

Address: Galway

Phone No.: 0977778264

Email: cathytalbot@gmail.com

Username:

Password: ●●●

Confirm Password: ●●●

Donation

2,000
1,500
1,000
500

Figure 14 Invalid username submitted and subsequent error message returned from Action

Model

The model contains all application-specific content and processing logic, including all content objects, access to external data / information sources such as databases and all processing functionality that are application specific. The model can sometimes be divided into two separate subsystems – the internal state of the system and the actions that can be taken to change that state. (The Apache Software Foundation, 2007)

According to Robert Englander (1997), JavaBeans is an architecture for both using and building components in Java and this architecture supports software reuse, and object orientation. The state components of the model define the current set of values of the

model and include methods to change those values. These methods capture some of the business logic.

Attributes of a JavaBean are called properties and are referenced by name. These properties can be read and written by calling methods on the JavaBean specifically created for that purpose. These methods are known as set and get method calls and they are used to obtain or change the value of a property. These methods are Java methods exposed by the class that implements the Bean and they represent the interface used to access and manipulate the component. Typically, the set of public methods which are defined by the class will map directly to the supported methods for the JavaBean, although it's possible that only a subset of the public methods be exposed. (Robert Englander, 1997)

During the course of the project, JavaBeans were used extensively. The JavaBeans classes have “get” and “set” methods. These methods provided an interface for property manipulation. The methods are accessed through actions, which defined the allowable changes and modifications to the state of the properties in response to an event occurring.

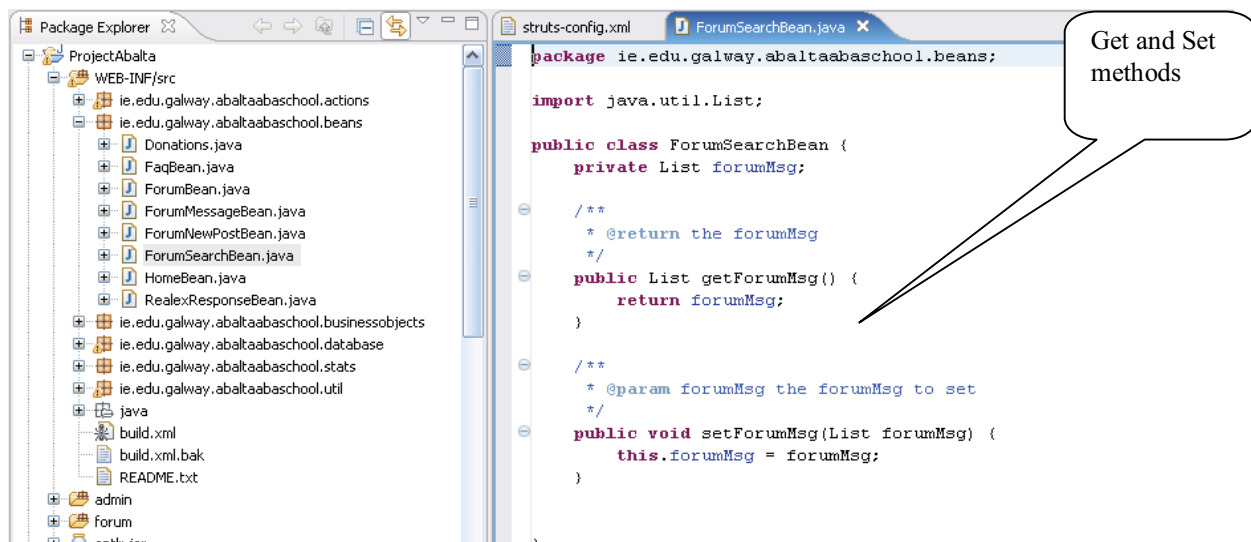


Figure 15 JavaBean for accessing and manipulating Ábaltas' forum messages

For ActionFormBeans that the controller provides, it generally assumes that there is an ActionForm bean defined in the application. The model for the beans helps the controller action form to validate the input from the user. Ábaltas' web portal utilises these beans for validation for user input in forms.



Figure 16 Validation for user form input (Corresponding to Controllers ActionFormBean)

In addition to the JavaBeans, Ábaltas' model also incorporates a MySQL database. The database offers persistent storage of Ábaltas' data and the model encapsulates the Java

Database Connectivity (JDBC) connection to the database and the queries used to manipulate data from the database.

JDBC

The JDBC API can access any kind of tabular data, including data in a MySQL database.

For Ábaltas' web portal, the JDBC API supports a two-tier model for database access.

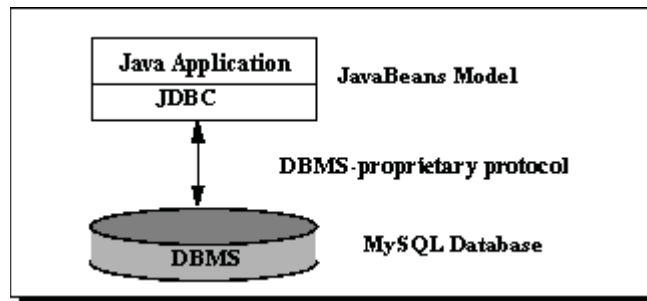


Figure 17 Two-tier model for database access (Java Sun JDBC Introduction (1995-2007))

The configuration for the database for this project is achieved through the J2EE deployment descriptor file web.xml. This file is used to define components and operating parameters for a web application and is packaged in the web application.

The database configuration utilizes the Java Naming and Directory Interface (JNDI) which is an API which allows the access of different naming and directory services.



Figure 18 MySQL database configuration

The JDBC driver communicates with a data source that will access the data. Commands can then be delivered to the MySQL database, and the results of those statements are returned.

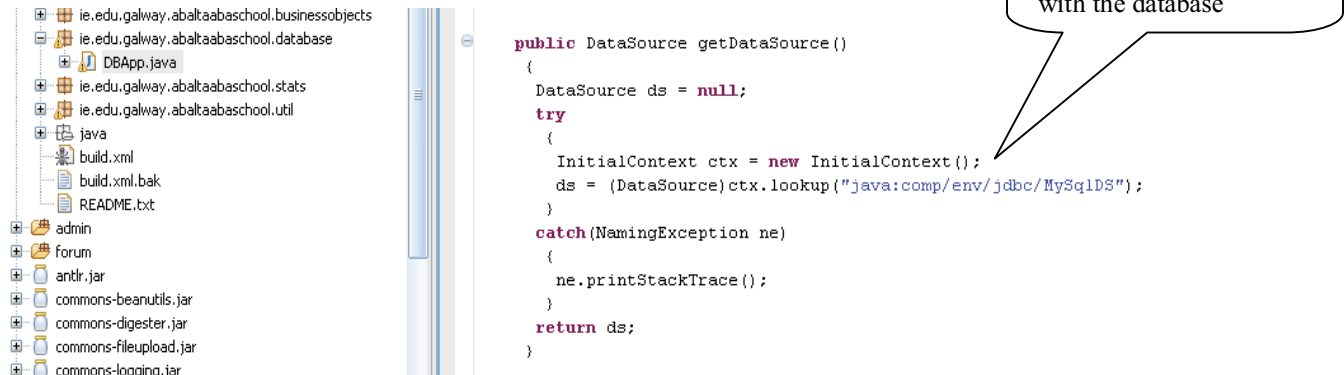


Figure 19 JDBC initialising new data source

```
public void addRole(Connection conn, String uType, String uName) throws SQLException
{
    String sql = "insert into roles values(null, ?,?)";
    PreparedStatement ps = conn.prepareStatement(sql);
    ps.setString(1, uType);
    ps.setString(2, uName);
    ps.executeUpdate();
    ps.close();
}
```

Figure 20 SQL prepared statements for inserting into MySQL database

View

The view framework represents the presentation logic of the application. The view components obtain the current state of the system from the model and provide the graphical user interface for the specific protocol involved.

This project uses JavaServer Pages and HTML for implementing the view component. When the JSP page prepares the related HTML for displaying data, the JSP page can access the model bean, which can store values required for presentation. These values are provided from the business logic layer or from previous user input. This functionality proved particularly useful for presenting data onto the user interface and was used extensively throughout the project.



Figure 21 Web portal accessing Bean for data for output to screen

Developing the view component of Ábaltas' Model-View-Controller architecture in JSP and HTML ensured adherence to and facilitated the standards of the user interface not containing any business logic, instead obtaining the data logic from the model when necessary.

Security Framework

This projects' security framework was developed using Java Authentication and Authorization Service (JAAS). This is a set of packages that enable authentication and enforce access controls upon users.

JAAS

According to JAAS Sun Developer Network (n.d.), JAAS services provide for:

- Authentication of users – reliably and securely determine who is currently executing the application, regardless of which part of the application is executing.
- Authorization of users – ensure users have the correct access control permissions to do the actions performed

The JAAS authentication framework is performed in a pluggable fashion permitting applications to remain independent from underlying authentication technologies. This facilitates new or updated technologies being plugged in without requiring modifications to the application itself. (Java Authentication and Authorization Service, Sun Developer Network, n.d.)

Integration with the Java 2 SDK allows the JAAS authorization to utilize the `java.security.Policy` API to handle Principal-based queries. The default policy implementation supports Principal-based grant entries. Consequently, access control is based not just on what code is running, but also on who is running it.

The configuration for JAAS is achieved through the web.xml file, similar to the database configuration, and the Resin web server. Resin provides a security layer which this project utilizes to provide authentication and authorization.

Authentication

Authentication in Ábaltas' web application is achieved via a login facility.



Figure 22 Login facility

Resin utilizes its own authenticator class called JdbcAuthenticator

(com.caucho.http.security.JdbcAuthenticator), which will ask the MySQL database for

Location of data source where username and password are stored

ut) password matching the (input) user's name.

Authenticator type

```
<!-- Resin-specific JdbcAuthenticator -->
<authenticator type='com.caucho.server.security.JdbcAuthenticator'>
  <init>
    <data-source>jdbc/MySQLDS</data-source>
    <password-query>
      SELECT password FROM users WHERE uname=?
    </password-query>
    <role-query>
      SELECT role_type role FROM roles WHERE uname=?
    </role-query>
    <password-digest>none</password-digest>
  </init>
</authenticator>
```

SQL query to select the role of the user as specified in database

SQL query to select password from database

Figure 23 Security Configuration in web.xml file

When the user logs into the application through the login facility, it executes code in a form, in the header.jsp file. This form must conform to certain conditions to implement the JAAS code in the web.xml. These are as follows:

- The form action must be named j_security_check. This will tell Resin to proceed with the authentication
- The form must have the username input name as j_username and the password name as j_password
- The form optionally has a j_uri which will specify the location to proceed to when successfully logged in



```
<form action="j_security_check" method="POST">
<strong><bean:message key="menu.label.uname"/></strong><input name="j_username" type="text" size="16"/>
<strong><bean:message key="menu.label.pword"/></strong><input name="j_password" type="password" size="16"/><br />
<input name="j_uri" type="hidden" value="/Home.do" />
<br/><input type="submit" <if(isLoggedIn) out.println("disabled");</if> value="Login" />
<nbsp;<nbsp;   <nbsp;   <nbsp;   <nbsp;   <a href="/info/register.do"><bean:message key="menu.label.reg"/></a>
</form>
```

Username and password

Proceed with authentication

On successful login go to Home.do page

If user logged in, disable login button

Figure 24 Login form action

Authorization

Authorization is used to protect selected areas of the web application. According to Java Authentication and Authorization Service, Sun Developer Network (n.d.), JAAS uses security constraints. The security constraint is an element that defines the access privileges to a collection of resources or pages within the web application.

Within Ábalta, there is a security restraint for the forum, whereby any person can view data in the forum, but only logged in users can participate in the forum.

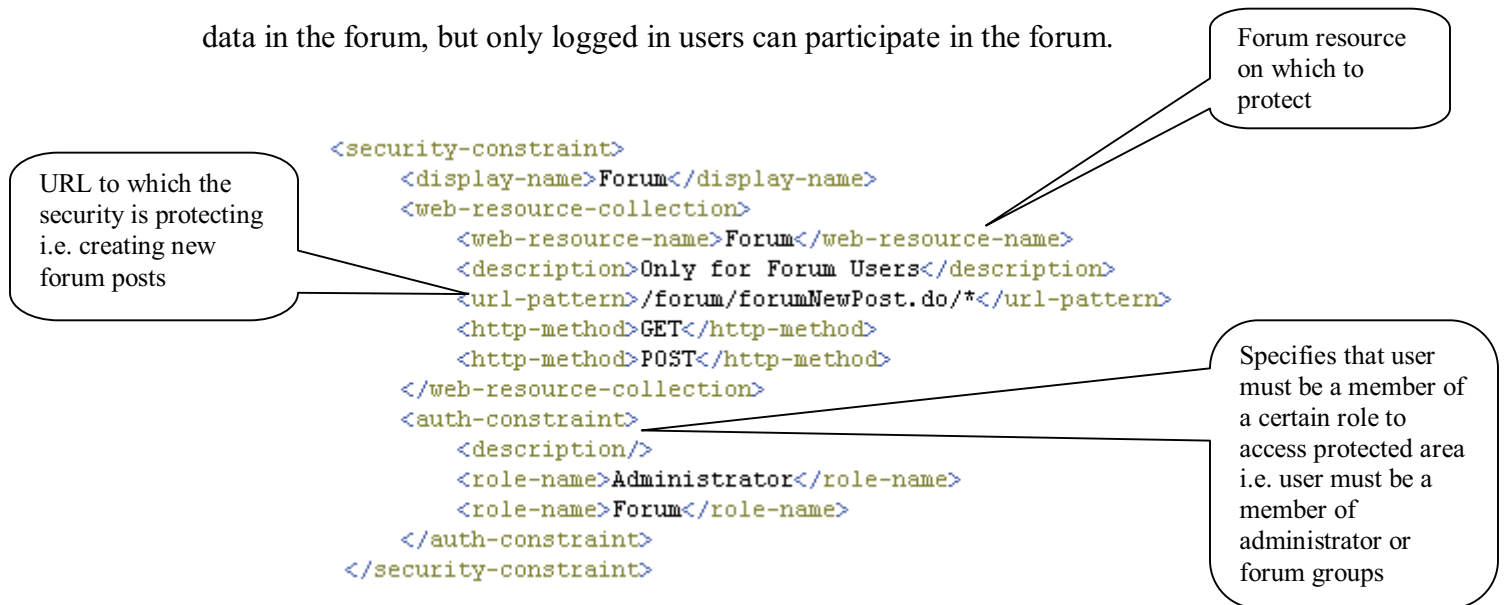


Figure 25 Authorisation security-constraints for the forum facility in Ábalta

The roles that are referenced must be setup in the web.xml file also. These roles are defined using a security-role.

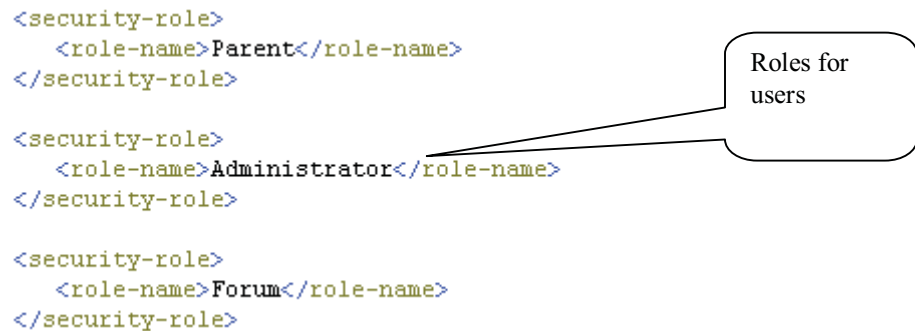


Figure 26 Security Roles for Ábalta

When a user attempts to login but does not have the correct privileges or an error occurs, they can be forwarded to a specified location. These scenarios are configurable.

If user is not authorised for access, forward to specified location

```
<login-config>
  <auth-method>FORM</auth-method>
  <form-login-config>
    <form-login-page>/info/unauthorised.jsp</form-login-page>
    <form-error-page>/info/loginerror.jsp</form-error-page>
  </form-login-config>
</login-config>
```

Figure 27 Authorization unsuccessful



Figure 28 Unauthorised access

JAAS offers this project huge advantages in authentication and authorisation for web applications, in an efficient, effective and configurable method.

Build Framework

This project is deployed by building all the source files into a web application archive (WAR) file and uploading this file to the host server. Building the source files into a WAR file is achieved by ANT (Another Neat Tool).

ANT

ANT is an open source Java-based build tool available from Apache. It works to build and automate the building of a software project in anticipation of deployment. ANT utilizes a build.xml file to “build” the project. The build.xml file is a basic, well-formed XML file that contains one (or more) project and at the very least one default target – a target is a set of tasks that are to be executed.

The build.xml file also contains properties. These are name/value pairs that are essentially variables and allow for the setting of values that can easily be changed later or overridden at the command line. (A Little Ant Can Lighten the Load, 2007)

This project involved the creation of a build.xml file for the successful building of all source code into a WAR file for deployment onto the web server. This projects’ build.xml file utilizes the targets and properties to successfully create a build framework for the web portal.

```
61
62
63     <!-- Normal build of application -->
64     <target name="compile" depends="prepare,resources">
65         <javac srcdir="src" destdir="classes">
66             <classpath refid="compile.classpath"/>
67         </javac>
68     </target>
69
70
71     <!-- Remove classes directory for clean build -->
72     <target name="clean"
73         description="Prepare for clean build">
74         <delete dir="classes"/>
75         <mkdir dir="classes"/>
76     </target>
77
78
```

The “compile” target compiled all the source code from the source directory. It outputs the WAR file to the destination directory as defined in the properties.

The “clean” target deletes the build and dist subdirectories. Cleaning is important as it forces a fresh compilation the next time the build is executed.

Figure 29 Ábaltas' build.xml file - compile and clear targets

This project instructs the build process to build the WAR file using the build.xml file

```
101
102
103     <property name="distpath.project" value="/projects/lib"/>
104
105     <!-- Build entire project -->
106     <target name="project" depends="clean,prepare,compile"/>
107
108     <!-- Create binary distribution -->
109     <target name="dist"
110           description="Create binary distribution">
111
112         <mkdir dir="${distpath.project}"/>
113
114         <jar
115             jarfile="${distpath.project}/${project.distname}.jar"
116             basedir="./classes"/>
117
118         <copy
119             file="${distpath.project}/${project.distname}.jar"
120             todir="${distpath.project}"/>
121
122         <war
123             basedir=".."
124             warfile="${distpath.project}/${project.distname}.war"
125             webxml="web.xml">
126             <exclude name="${distpath.project}/${project.distname}.war"/>
127         </war>
128     </target>
129
```

Create the
"projects/lib"
directory

Location of
where to output
the generated
file

Create a jar file
of all classes in
"classes"
directory

Make WAR file
consisting of
everything in base
directory and save
it.

Figure 30 build.xml - Build process

Source Code Directory Structure

For the successful implementation of ANT and automated builds, a key element is the directory structure employed for the source code. A standard directory structure for all Web applications is essential, promoting consistency and code re-use, as well as avoiding

administrative tedium and confusion. (A Little Ant Can Lighten the Load, 2007)

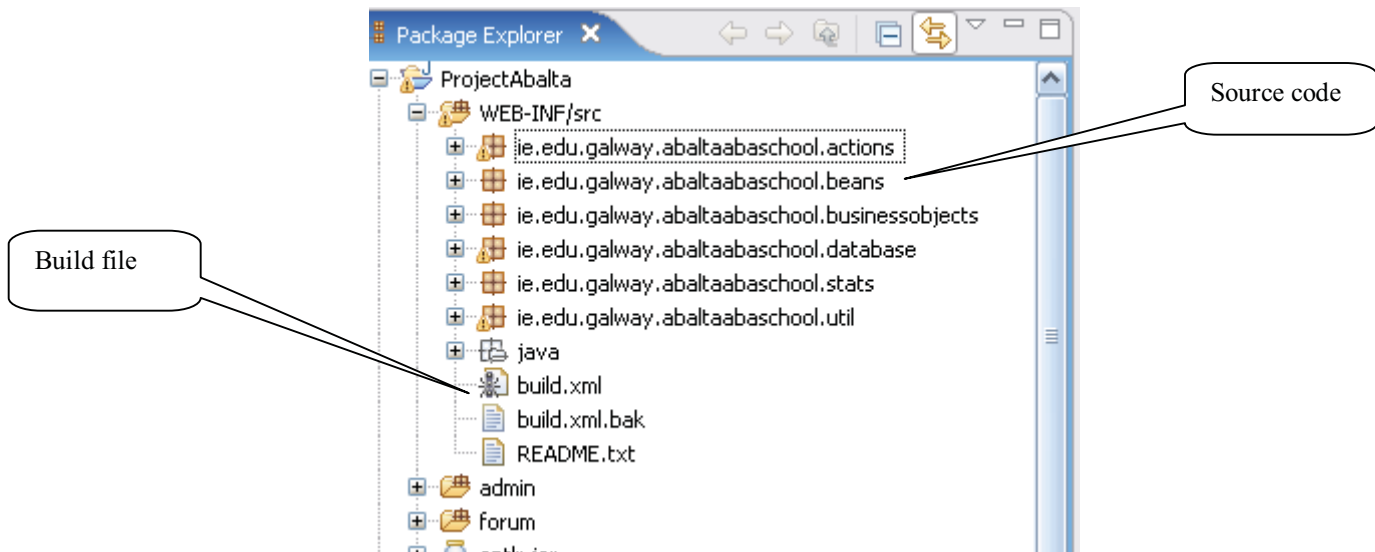


Figure 31 Ábaltas' File Structure

An important part of the file structure was separating the build files from the source code. This separation proved beneficial as it is easier to manage, move, and backup the source code, it was easier to identify what source files constituted the installed application, and it simplified distribution.

Summary

The frameworks employed in the development of this project worked to improve the overall design of the web portal. These frameworks helped to organize complex code in the project, provided standard interfaces that allowed model components to be combined

and reused, and offered standard methods to deal with common functions such as error handling and automatically building the project for deployment.

Chapter V Third Party Integration

Although Ábalta is a government recognised school, it is primarily a charity and obtains most of its funding through donations from the public. It is important that there is a facility for generating and obtaining monetary gifts in an easy and straight-forward method. In today's technologically advanced world, it is estimated that there is over one billion internet users globally so it was logical that one approach for generating funds would be to utilise the internet through Ábaltas' web portal application.

Payment Service Provider

A payment service provider is a provider who specializes in e-payments and offers a full or particular payment processing in aspects such as contract, billing, credit card authorization, and payer authentication for internet service providers or merchants such as businesses or charities.

Integration with a payment service provider is an ideal solution for Ábaltas' online donation feature. It overcomes issues such as the requirement for a secured server through the hosting company, provides a safe and secure method for processing credit card donations and ensures that the charity will not be liable in the unlikely event of credit card fraud.

From researching and investigating various payment service providers, Realex Payments, based in Ireland, presented an experienced, proven record of service for dealing with charities and offering a donation facility.

Realex Payments

Realex Payments' core business is the creation and running of payment exchanges that link backs with customers via the internet. They offer a range of payment and financial services including the following which this project will utilise for the donation facility:

- Real time credit and debit card authorisation services
- Real time fraud scoring and pattern checking services
- A credit processing service (Realex Payments, n.d.)

Realauth

Realauth is a Realex Payment-hosted, multi channel, bank, account and currency payment solution. It allows for the complete credit and debit card authorisations at the point of donations. The primary purpose of Realauth is to provide a real time authorisation and suitability scoring of a credit card transaction at the point of online sale / donation for a merchant. The authorisation and score request is split into two processes:

- Authorisation request is submitted directly to the merchant's acquiring bank
- Realex independently computes the suitability score using predefined merchant criteria

The bank's response is then combined with the suitability score and returned to the merchant – all in a matter of seconds. (Realex Payments, n.d.)

Ábalta School does not have a secure server for the safe processing of credit and debit cards. Realex will provide a secure server for Ábalta and donors will be re-directed from Ábaltas' web portal at the card processing stage to this secured server. All these transactions occur over secure SSL connections

The Reauth redirect ensures that Ábalta will not be required to collect and store credit or debit card details, in fact, it ensures that Ábalta does not have any access to them whatsoever. The donor is redirected to the Reauth application on a secure Realex Payments server by a script running on Ábaltas' web application. This Realex server hosts an Ábalta-branded web page where credit card details are collected and the donation is processed. The donor and the results of the transaction are sent back to Ábalta via another script on the Ábaltas' web application.

This project will integrate with Realex on two levels:

1. Ábaltas' web portal submits requests for authorisations
2. Ábaltas' web portal accepts the response from the Realex application to the request.

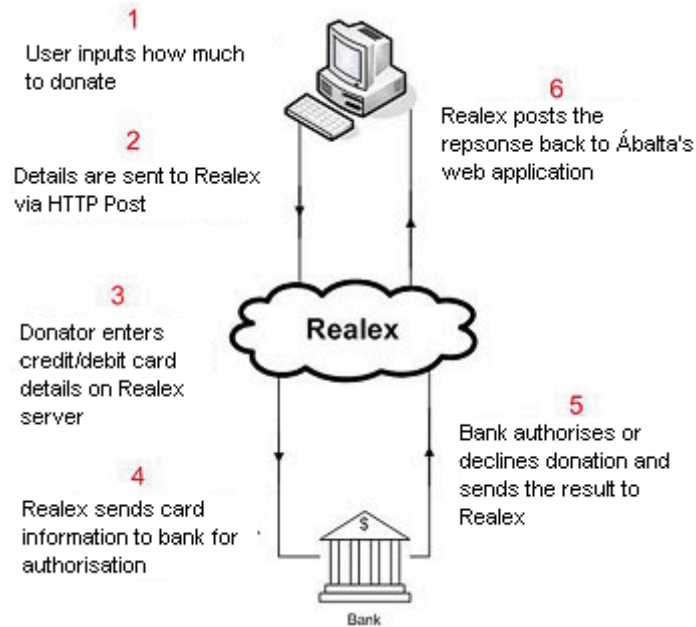


Figure 32 RealAuth Redirect Solution (Realauth Developer's Guide)

Prerequisites

In order for Ábalta to integrate with Realex Payments, there were some fundamental requirements that needed to be addressed. These were as follows:

- Ábalta must have a web application built for commerce
- Ábalta must have a merchant account with an approved bank, that enables the acceptance of credit cards over appropriate channels
- The web application must be amended to link to Realex Payments secure server

Web Application

This project involved the design and development of a web application for Ábalta (www.abaltaabaschool.galway.edu.ie) and incorporated the design of an e-commerce system for donations and fundraising.

Bank

As Ábalta did not have a merchant account prior to this, it was required that one be set up. Ábalta had a charity account set up with Allied International Bank (AIB) Ireland so a decision was made to also setup the merchant account with this bank. A meeting was arranged with a member of Ábaltas' board of directors, their treasurer and the merchant account manager of AIB and within one week a merchant account with AIB had been set up for Ábalta.

Contract with Realex

It was necessary for Ábalta to sign a contract with Realex. This contract incorporated the setup fees and monthly service fees, which were discounted because of Ábaltas' charity status, and the terms and conditions as of the service agreement.

Initial Setup Fee: €150.00

Monthly Service Fee: €25 : this included and allowed for 100 donations per month

Per Transaction: 0.25 cent : on all donations above the 100 donation mark

Development and Integration

The design, development and integration of the donation functionality into the web application involved working closely with the Realex Payments developer. There were a number of technologies that need to be utilised including the MySQL database, the JavaMail API and creating a digital signature using a hash of certain elements. The following serves to detail the design, development and integration of the donation utility provided in the web portal application.

Template

Realex Payments require a template which will contain the table for the input of credit / debit card details. Ideally the table should resemble the “look and feel” of the Ábaltas’ web application so as the donator will not realise that they have been redirected and the experience should be seamless.

Ábaltas’ template.html page was forwarded to Realex Payments for use.

```
23
24 </HEAD>
25 <BODY>
26     <br><br>
27     <fieldset>
28     <legend>Donation</legend>
29
30     <DIV ALIGN="CENTER">
31         <!--E-PAGE TABLE HERE-->
32     </DIV>
33 </fieldset>
34 </BODY>
35 </HTML>
36
```

Releax will place the table for collecting credit / debit card details here

Figure 33 template.html

Donor Details

Functionality needed to be provided for the donor to input their details. A form was developed (donations frame.jsp) that allowed donators to input their contact details and the amount of money they wished to donate. These details were the responsibility of the web portal and would require storage in the MySQL database.

Thank you for choosing to donate to Ábalta

Contact Details

Name

Joe Symth

Address

Bohermore

Galway

Phone No.

091-732832

Email

joesymth@gmail.com

Note: Your receipt will be sent to this email address

Amount (Eur)

100.00

Continue

All fields are required

Privacy Policy

Ábalta takes the confidentiality of your information seriously.

We guarantee that every transaction you make at our Web Site will be 100% safe. Credit card payment are made through Realex. Security is central to the credit card industry and Realex Payments operate security policies and controls that are industry standards.

Ábalta is a registered charity. Charity Reg No: CHY14741

For any queries, please contact support@abaltaabaschool.galway.edu.ie

Figure 34 donations frame.jsp

Donation Details

When the user clicks on the “Continue” button, the following sequence of events occur.

(Realth Developer’s Guide, 2007)

1. The struts-config.xml file invokes the DonationAction class

```
239
240
241     <action
242         path="/submitDonation"
243         type="ie.edu.galway.abaltaabaschool.actions.DonationAction"
244         name="DonationForm"
245         scope="request"
246         validate="true"
247         input="/donate/donations.jsp">
248         <set-property property="cancellable" value="true"/>
249         <forward name="success" path="/donate/donations_frame2.jsp"/>
250         <forward name="failure" path="/donate/donations_frame.jsp"/>
251         <forward name="cancel" path="/donate/donations_frame.jsp" />
252     </action>
253
```

Invoke class
DonationAction.java

Figure 35 struts-config.xml file

2. The DonationAction.java class checks the form to ensure that all fields are correctly input and manages all the error handling for the donation amount.

```
if(((String)postForm.get("donation")).equals("")) {
    errors.add("donation",new ActionMessage("error.donation.required"));
}
else{
    try{
        amount = Double.parseDouble(postForm.get("donation").toString());
        if ((amount == 0) || (amount < 0)){
            throw new Exception();
        }
    }catch(NumberFormatException e){
        errors.add("donation",new ActionMessage("error.donation.parse"));
    }
    catch(Exception ex){
        errors.add("donation",new ActionMessage("error.donation.parse"));
    }
}
}
```

Ensure its
valid amount

Ensure that
the input is a
number

Figure 36 Error Handling – DonationAction.java

3. A digital signature is created

In order to ensure that the request comes from Ábalta, Realex requires that a digital signature is created. This digital signature is a combination of a number of elements.

These elements are as follows:

- Timestamp in the form of YYYYMMDDhhmmss
- Merchant ID that was provided by Realex Payments
- A unique donation ID generated for the donation
- The amount being donated
- The currency code used which is EUR
- A shared secret provided by Realex.

The digital signature is created using a secure hash algorithm known as SHA-1. This algorithm takes a string as an input and produces a 160-bit fixed size number. This number is a hash of the input and a small change in the input results in a significant change in the output. Ábalta has been reassured that there is no way to decrypt a secure hash, thus allowing for maximum security for their donors.

The algorithm utilises Java security class MessageDigest to assist in the hashing of data. The MessageDigest class provides applications with the functionality of a message digest algorithm, such as MD5 or SHA-1. These message digests are

secure one-way hash functions that take arbitrary-sized data and output a fixed-length hash value.

```
MessageDigest md;
```

```
try {
```

```
md = MessageDigest.getInstance("SHA");
```

```
md.update(toBeHashed.getBytes());
```

```
digestValue = md.digest();
```

Tell MessageDigest to use the SHA algorithm

Creates byte array from the digest

Iterate through byte array creating the digital signature

```
for (int i = 0; i < digestValue.length; i++) {  
    String c = Integer.toHexString(digestValue[i]);  
    if (digestValue[i] < 0) c = c.substring(6);  
    if (c.length() < 2) c = "0" + c;  
    sb.append(c);  
}
```

```
} catch (Exception e) {
```

Figure 37 Hashing algorithm SHA-1

4. The JavaBean (Donations.java) for the donations feature is then invoked and initialised. The properties that “set” are the fields that Realex requires upon redirection. These fields are:

- Merchant ID
- OrderID (DonationID)
- Amount to be donated
- Currency code
- Timestamp generated
- Hash Value

```

Donations donations = new Donations();
donations.setRedirect(true);
donations.setCurrency(Realax.CURRENCY);
donations.setTimeStamp(timestamp);
donations.setMerchant_id(Realax.MERCHANT_ID);
donations.setOrder_id(orderId);
donations.setAmount(amt);
donations.setShalHash(shalhash);

request.setAttribute("donationsBean", donations);

```

Figure 38 Initialising the JavaBean with required data

5. The donations frame2.jsp file is executed. This file redirects the donations to the Realex Payments secure server and sends the data required.

Link to Realex's secure server denoted by https

```

<form name="realexForm" action=https://epage.payandshop.com/epage.cgi method=post>

<input type=hidden name="MERCHANT_ID" value="<%=donationsBean.getMerchant_id()%>">
<input type=hidden name="ORDER_ID" value="<%=donationsBean.getOrder_id()%>">
<input type=hidden name="CURRENCY" value="<%=donationsBean.getCurrency()%>">
<input type=hidden name="AMOUNT" value="<%=donationsBean.getAmount()%>">
<input type=hidden name="TIMESTAMP" value="<%=donationsBean.getTimeStamp()%>">
<input type=hidden name="SHALHASH" value="<%=donationsBean.getShalHash()%>">
<input type=hidden name="AUTO_SETTLE_FLAG" value="1">

```

Access the data from the JavaBean

Figure 39 Redirection file donations_fram2.jsp sends required information

6. The user is then redirected to the secure server on Realex Payments server. This is where the user enters their credit card details. The user is oblivious to the fact that they have been re-directed; the user interface is the same as the previous screen. Ábalta does not have any access to the credit / debit card details.

Releax secured server processing the credit/debit cards

Donation

Card Type:

Card Number:

Security Code: ([about security code](#))

Expiration Date:

Cardholder's Name:

payment processed by **realex**
The real-time payments exchange

Links

- [Autism Society](#)
- [Autism Ireland](#)

Figure 40 Secured server provided by Realex Payments

- When the user has entered in their details, and clicks on the “Pay Now” button, the web application awaits a response from Realex Payments.
- When a response comes, the struts-config.xml file instructs the application what to do.

```
<action
  path="/realexResponse"
  type="ie.edu.galway.abaltaabaschool.actions.RealexResponseAction"
  name="RealexResponseForm"
  scope="request"
  validate="true">
  <forward name="success" path="/donate/realexResponse.jsp"/>
</action>
```

When class has been successful executed, run realexResponse.jsp

Execute the RealexResponsAction class

Figure 41 struts-config.jsp code for Realex’s response

8. The response incorporates the following data:

MERCHANT_ID	This is the merchant id that Realex Payments assign to you.
ORDER_ID	The unique order id that you sent to us.
AUTHCODE	Will contain a valid authcode if the transaction was successful. Will be empty otherwise.
RESULT	The outcome of the transaction. Will contain "00" if the transaction was a success or another value (depending on the error) if not. See the result codes table in Appendix A.
MESSAGE	Will contain a text message that describes the result code above.
PASREF	A unique reference that Realex Payments assign to your transaction.
SHA1HASH	A SHA-1 digital signature created using the above fields and your shared secret. Needs to be sent in lowercase.
TSS	The Transaction Suitability Score for the transaction.

Figure 42 Response data from Realex Payments (Realth Developer's Guide, 2007)

7. RealexResponsAction.java ensures that the response SHA1HASH algorithm is checked and verified against the hash algorithm that the web application had originally sent to Realex. This will validate that all is secured in the donation transaction.
8. If the hash algorithm were successfully matched, then a Boolean called "hashesMatch" in the JavaBeans class RealexResponseBean.java is set to true. Otherwise it is set to false. Then the result response is checked to verify if the donation transaction was successful.
- There are a number of codes that indicate whether the result of the transaction was successful or not. These are as follows:

00	Successful
101	Declined by Bank
102	Referral by Bank (treat as decline in automated system such as internet)
103	Card reported lost or stolen
2xx	Error with bank systems
3xx	Error with Realex Payments systems
5xx	Incorrect XML message formation or content
666	Client deactivated.

Figure 43 Response Codes (Realth Developer's Guide, 2007)

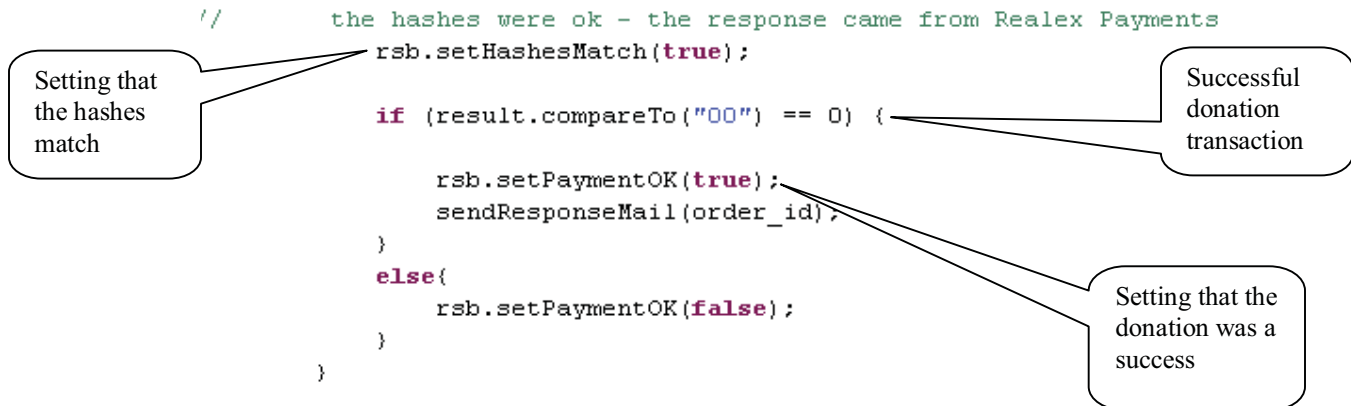


Figure 44 Verify that donation was successful

9. If the donation was successful, a setPaymentOK Boolean in the JavaBeans class RealexResponseBean.java is set to true, otherwise its set to false and an email is sent to the donor. The email will be sent (to the email address provided in the initial user details screen) as a receipt and thanking them for their donation. MailUtil.java invokes the java mail API to provide this functionality.

```

Properties props = System.getProperties();
props.put("mail.smtp.host", mailhost);
Session session = Session.getInstance(props, null);
Message msg = new MimeMessage(session);
if (from != null)
    msg.setFrom(new InternetAddress(from));
else
    msg.setFrom();

msg.setRecipients(Message.RecipientType.TO,
    InternetAddress.parse(to, false));
if (cc != null)
    msg.setRecipients(Message.RecipientType.CC,
        InternetAddress.parse(cc, false));
if (bcc != null)
    msg.setRecipients(Message.RecipientType.BCC,
        InternetAddress.parse(bcc, false));

msg.setSubject(subject);

msg.setText(message);

msg.setHeader("X-Mailer", "Abalta");
msg.setSentDate(new Date());

// send the thing off
Transport.send(msg);

```

Setup all the email details for the message

Send the email

Figure 45 MailUtil.java invokes the java mail API

Abalta Donation - Thank you. [Inbox](#)

☆ "support@abaltaabaschool.galway.edu.ie"

Dear Joe Smyth

Thank you very much for supporting Abalta. Your donation of EUR 20.0 is greatly appreciated.

Abalta's Charity Number is CHY14741.

For additional information or questions, please contact the support on support@abaltaabaschool.galway.edu.ie.

Thank you again for your generous support.

Sincerely,

Cairde Abalta

Figure 46 Sample Receipt Email

10. Confirmation will also be output onto the web application's user interface to verify the success or failure of the donation transaction. `realexResponse.jsp` will check the JavaBeans class `RealexResponseBean.java` to check the success and failure status of the donation. The user will be duly notified of the status.

Summary

Integration with Realex Payments provides Ábalta and its' donors with a secure and protected method of generating funds over the internet, offering opportunities for people globally to donate to this rewarding cause

Chapter VI Ábaltas' Web Portal Application

Ábaltas' web portal application is deployed and is available at www.AbaltaAbaSchool.galway.edu.ie and www.AbaltaSchool.ie. This chapter will provide an overview of the web portal application, its' features and the services it offers to Ábalta.

Informational Content

Ábaltas' web portal provides information about Ábalta, what the school hopes to achieve, the educational methodology it provides to its students, autism and offers testimonial statements from some the parents of the students who attend the school. It also provides information regarding any upcoming fundraising events or any recent events that have occurred.

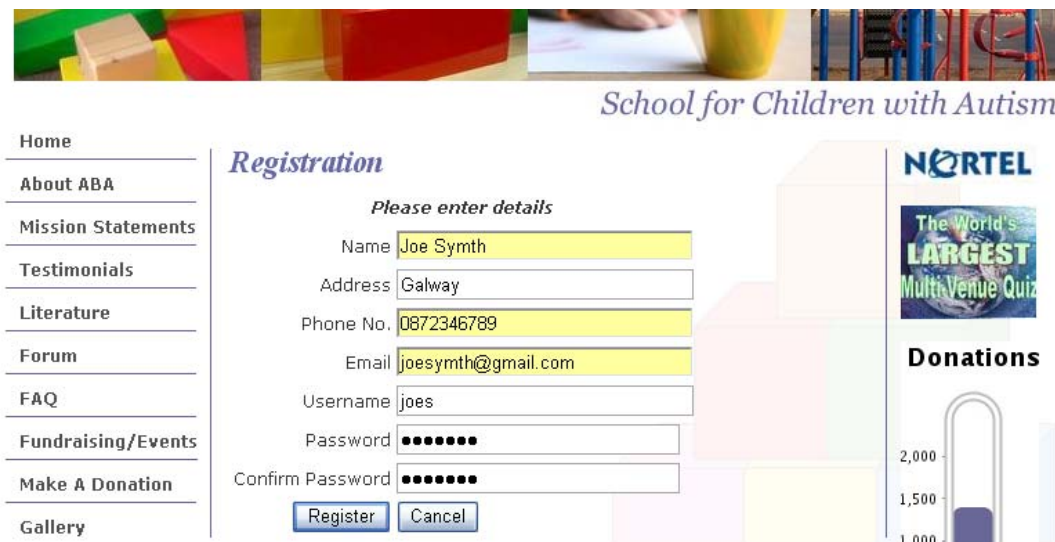


Figure 47 Home Page of Ábaltas' web portal application

Registration

It was necessary for security reasons that users register for the system in order to participate in the forum and to use the content management system. It is possible for

anyone to view the forum; however a user must be logged into the system in order to participate. The content management system requires that only a user with administrative privileges can manipulate the content of the web portal application. The registration process involved the completion of a form that would add / insert that user to the database.



The screenshot displays the 'Registration' page of the 'School for Children with Autism' website. At the top, there is a banner with three images: colorful blocks, a hand writing on a notepad, and a child on a playground. The site's name, 'School for Children with Autism', is written in a purple script font. On the left, a vertical menu lists various site sections: Home, About ABA, Mission Statements, Testimonials, Literature, Forum, FAQ, Fundraising/Events, Make A Donation, and Gallery. The main content area is titled 'Registration' and contains the instruction 'Please enter details'. Below this, there are input fields for Name (filled with 'Joe Symth'), Address (filled with 'Galway'), Phone No. (filled with '0872346789'), Email (filled with 'joesymth@gmail.com'), Username (filled with 'joes'), Password (masked with dots), and Confirm Password (masked with dots). 'Register' and 'Cancel' buttons are positioned at the bottom of the form. To the right of the form, there is a 'NORTEL' logo, a graphic for 'The World's LARGEST Multi-Venue Quiz', and a 'Donations' section featuring a bar chart with a scale from 1,000 to 2,000.

Figure 48 Registration Process

Login / Logout

The user may login to the web portal application through the login facility provided in the menu frame.

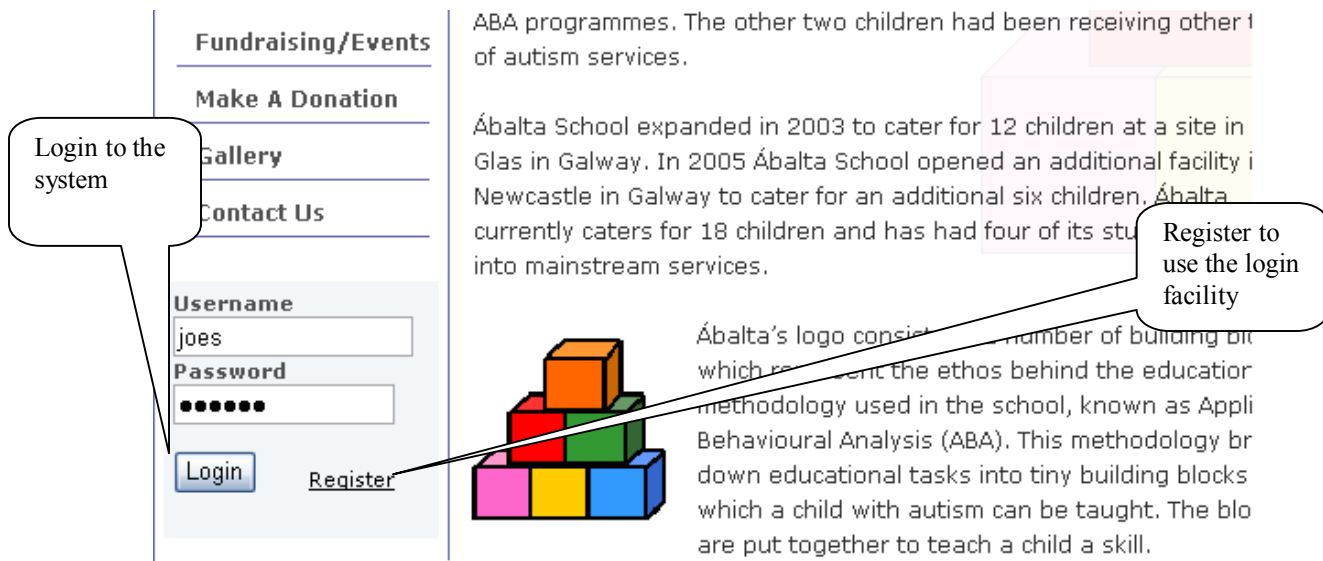


Figure 49 Login

When the user has successfully logged in, they may choose to logout at any time. A logout facility is provided to logging out of the system.



Figure 50 Logout

Forum

A forum was implemented that enables discussions and communication to occur. This enables a support network to be built up, documenting questions, real-life experiences and offering comfort and support between people.

There may be different discussions that are currently ongoing. In the forum area, there will be a list of discussions that are available for viewing or participating in.

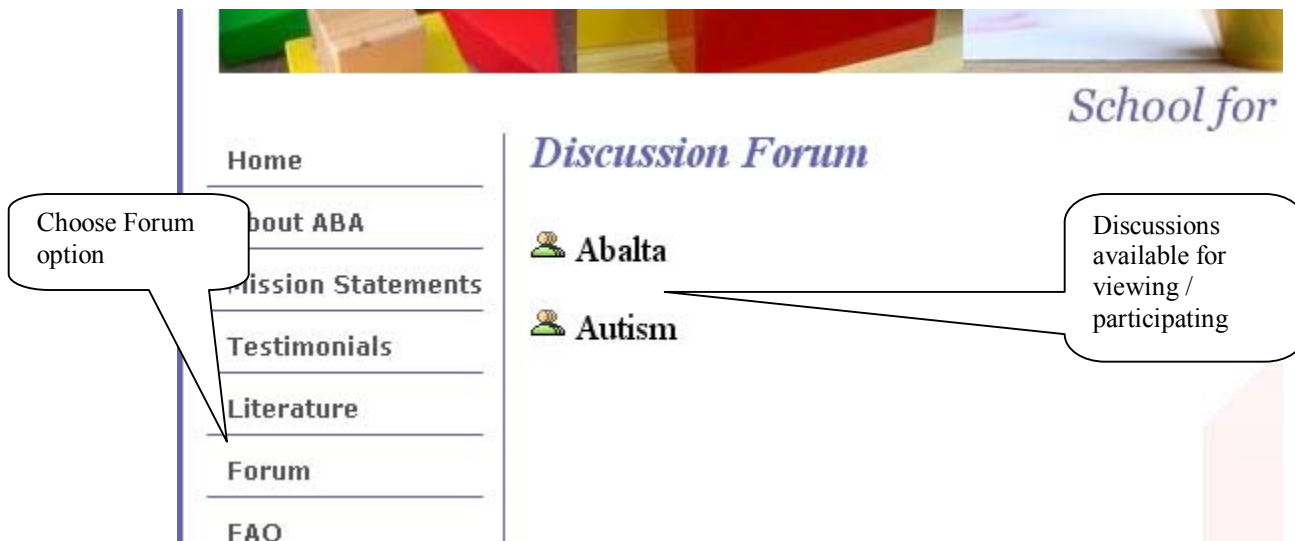


Figure 51 Forum Discussions

To view a discussion, the user can click on the discussion topic. This will bring the user to a dedicated forum area.

From this area, the user can choose to view any message or choose to submit a new post to the forum.



Figure 52 Forum Messages

Each of the messages are links. When the user clicks on the message link, the message details will be displayed. The user can then view the message and has the option of replying to the message.

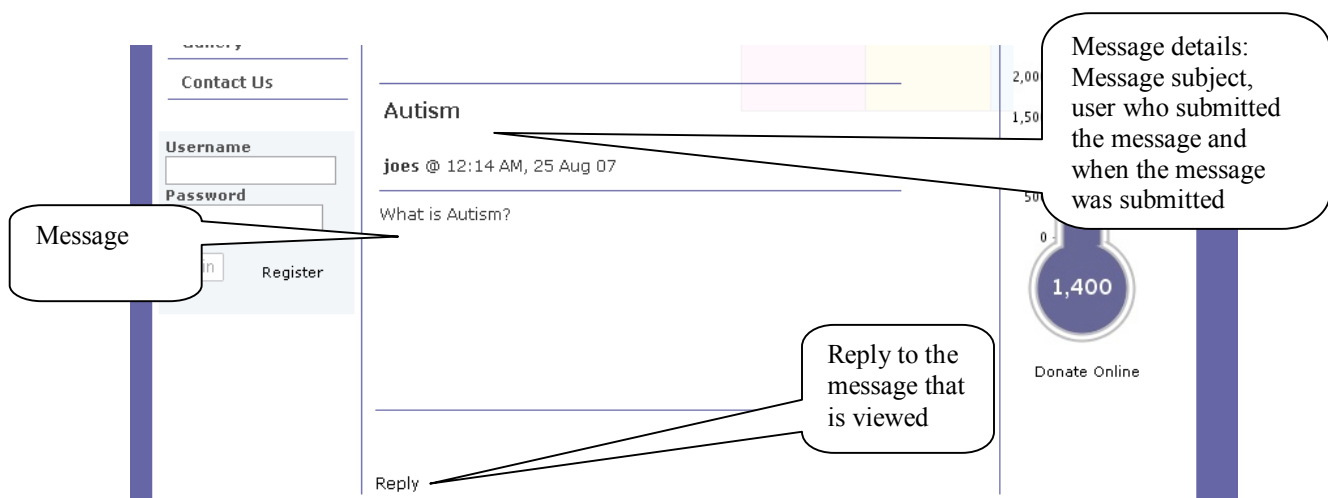


Figure 53 View / Reply to Message

New Post / Reply

The user may participate in the forum by submitting a new post. This is achieved by clicking on the “New Post” link in the forum area. When the user does so, they are presented to a dedicated posting area with a HTML editor for submitted the message.

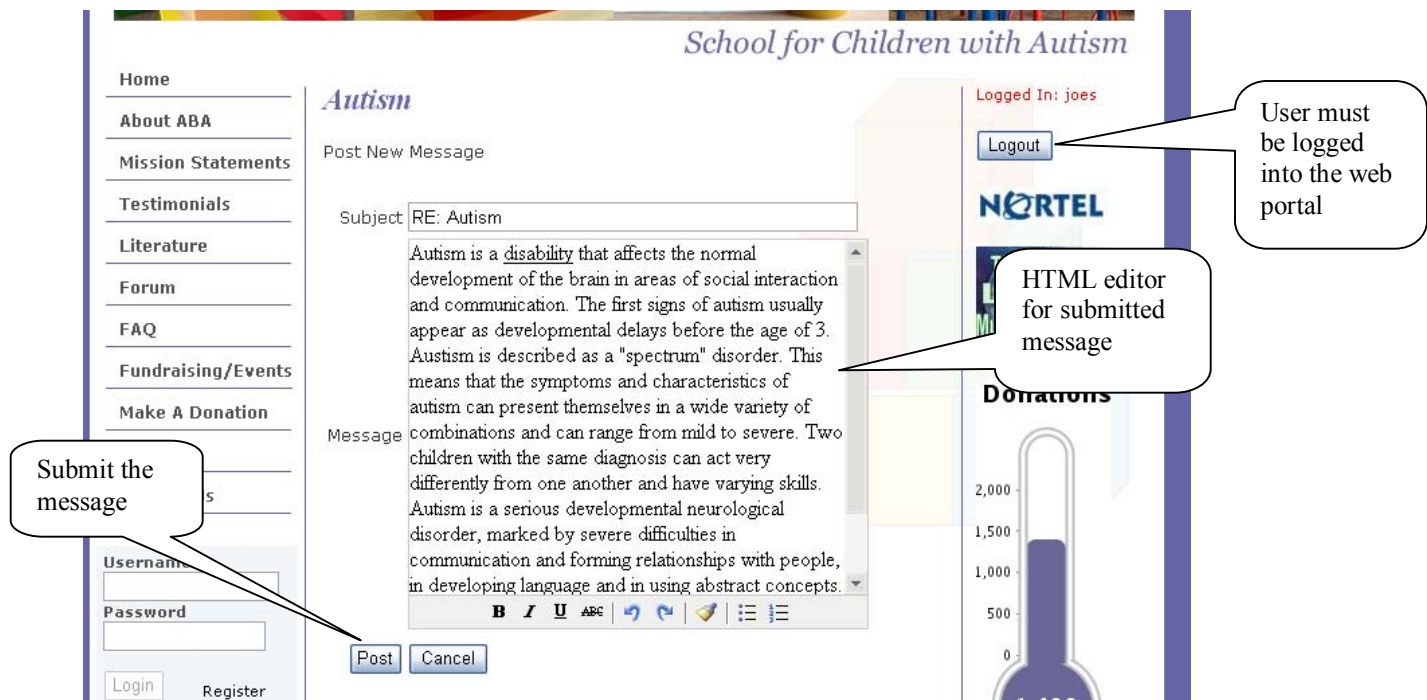


Figure 54 New Post

Search Forum

The user may choose to search the forum using a particular keyword. The search can be specified to search for the keyword across all messages that are in forum or there is an option to refine the search to make it more specialized. A specialized search may involve searching prior or post a certain date, or to search users, subjects or messages only.

School for Children with Autism

- Home
- About ABA
- Mission Statements
- Testimonials
- Literature
- Forum
- FAQ
- Fundraising/Events
- Make A Donation
- Gallery
- Contact Us

Search Forum

Search for Keyword

Refine Search (Optional)

After Date ☐ 1 Jan 2007

Before Date ☐ 1 Jan 2007

Subject Only ☐

Message Only ☐

User Only ☐

Logged In: mary

NORTEL

The World LARG

Donations

2,000

1,500

1,000

Specialised forum Search

Figure 55 Forum Search

Donations

As a charity, it is important that there is a facility for generating and obtaining monetary gifts in an easy and straight-forward method. The donations feature enabled the use of the internet for raising funds for Ábalta, reaching billions of people. The donations feature was designed and integrated with a third party organisation called Realex Payments who provided the secure server for the safe processing of credit cards.

Home

About ABA

Mission Statements

FAQ

Fundraising/Events

Make A Donation

Gallery

Contact Us

Username

Donation

Donation

Card Type:

Card Number:

Security Code: ([about security code](#))

Expiration Date:

Cardholder's Name:

payment processed by **realex**
The real time payment exchange

AIB Bank

NORTEL

The World's **LARGEST** Multi-Venue Quiz

Donat

2,000

1,500

1,000

500

0

Figure 57 Secured Server for Card Details Entry

When the user clicks on the “Pay Now” button, the server processes the donation.

The user is informed immediately of the result of the donation. A message, declaring whether the donation has been successful or unsuccessful, is displayed on screen.

Home

About ABA

Mission Statements

Testimonials

Literature

Forum

FAQ

Donation

Donation

We're sorry - there has been a problem processing your donation.

Please contact us at support@abaltaabaschool.galway.edu.ie and quote your donation number as: 148

Please note: Your credit card has not been charged.

NORTEL

The World's **LARGEST** Multi-Venue Quiz

Donations

Figure 58 Unsuccessful Donation

If the donation has been successful, the system sends an email to the donor. This email acts as a “thank you” to the donor and can be saved for the donor’s own records.



Figure 59 Email for Successful Donation

Donation Gauge – Thermometer

A donation gauge is available for viewing in the web portal. This is implemented as a thermometer. The thermometer gives an indication of how much has been donated to Ábalta through the web portal.

This is a dynamic thermometer that will automatically update when a web portal page is refreshed. It updates by retrieving from the database the sum of all donations that have been processed.

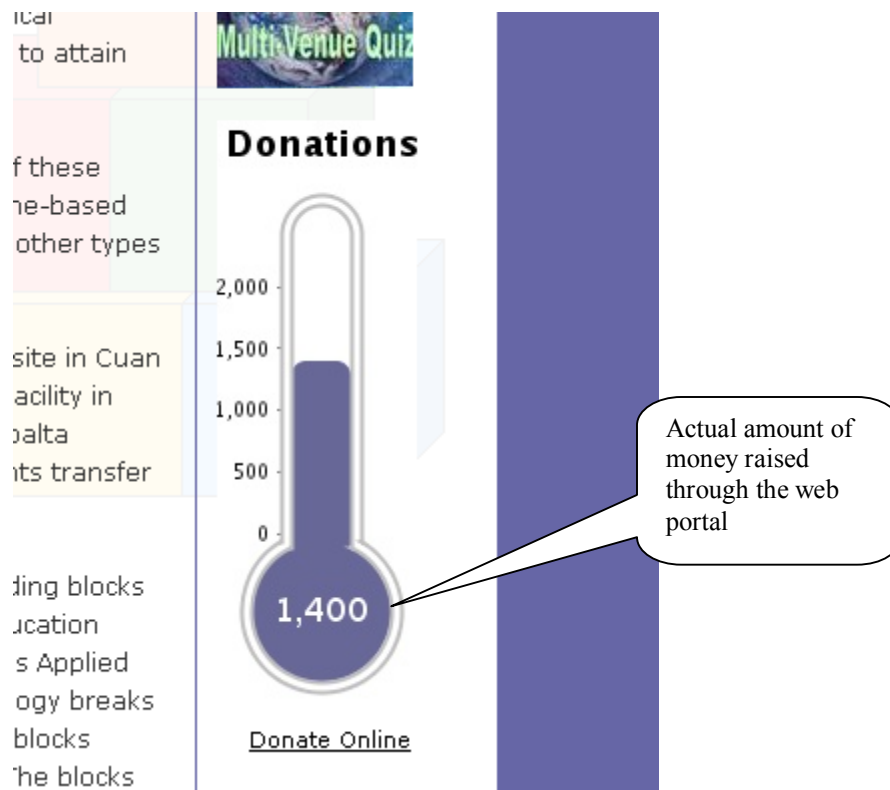


Figure 60 Donation Gauge - Thermometer

Content Management System

A content management system has been provided to Ábalta that will enable the easy maintenance of the content for the web portal application. A user must be logged into the system with administrative privileges in order to update any content in the web portal.

aabaschool.galway.edu.ie/info/gen.do?page_name=aba.jsp

Go Bookmarks 22 blocked Check AutoLink AutoFill Send to

Abalta

"I believe Abalta is one of the best centres for intervention for autistic children in the country."
- Nicola and James McNabb



Home
About ABA
Mission Statements
Testimonials
Literature
Forum

Applied Behaviour Analysis [Edit](#)

Applied Behaviour Analysis, or ABA, refers to a category of interventions that uses positive reinforcement to increase and decrease behaviours so that a person's quality of life improves. There are several teaching modalities that employ ABA; at Abalta School, we use a form of ABA called Discrete Trial Therapy. This methodology was studied extensively, and research found that DTT/ABA provided intensively resulted in 47% of the children studied reaching a level of skills so high as to make them indistinguishable from their typically developing peers.

DTT/ABA breaks down difficult concepts into small, manageable parts that can then be taught successfully. As children's skills grow, the concepts are re-combined into gradually more difficult concepts.

Each child has a curricular program designed specifically for them, according to their needs. Programmes are constantly updated to reflect children's progress and changing needs. All areas of development are considered when programs and IEPs are developed, including communication skills, social skills, self-help skills, and academic skills.

Parental input is essential in a quality DTT/ABA programme. Parents provide information not otherwise attainable, including whether skills are generalising from the school setting to home, what self-help skills and social skills need to be addressed, etc. Thus, parents must be involved in their children's programmes from the start, by attending IEP meetings, attending programme review meetings, and actively generalising skills learned in school to the child's other environments.

[Edit](#)

Logged In: webadmin
[Logout](#)

NORTEL
The World's LARGEST Multi Venue Quiz

Donations

2,000
1,500
1,000
500
0

1,400

[Donate Online](#)

Contact Us
Users:
Password:
[Login](#) [Register](#)

Events
Donation

User must be logged in with administrative privileges

Edit button allows user to edit content

Figure 61 Edit Content

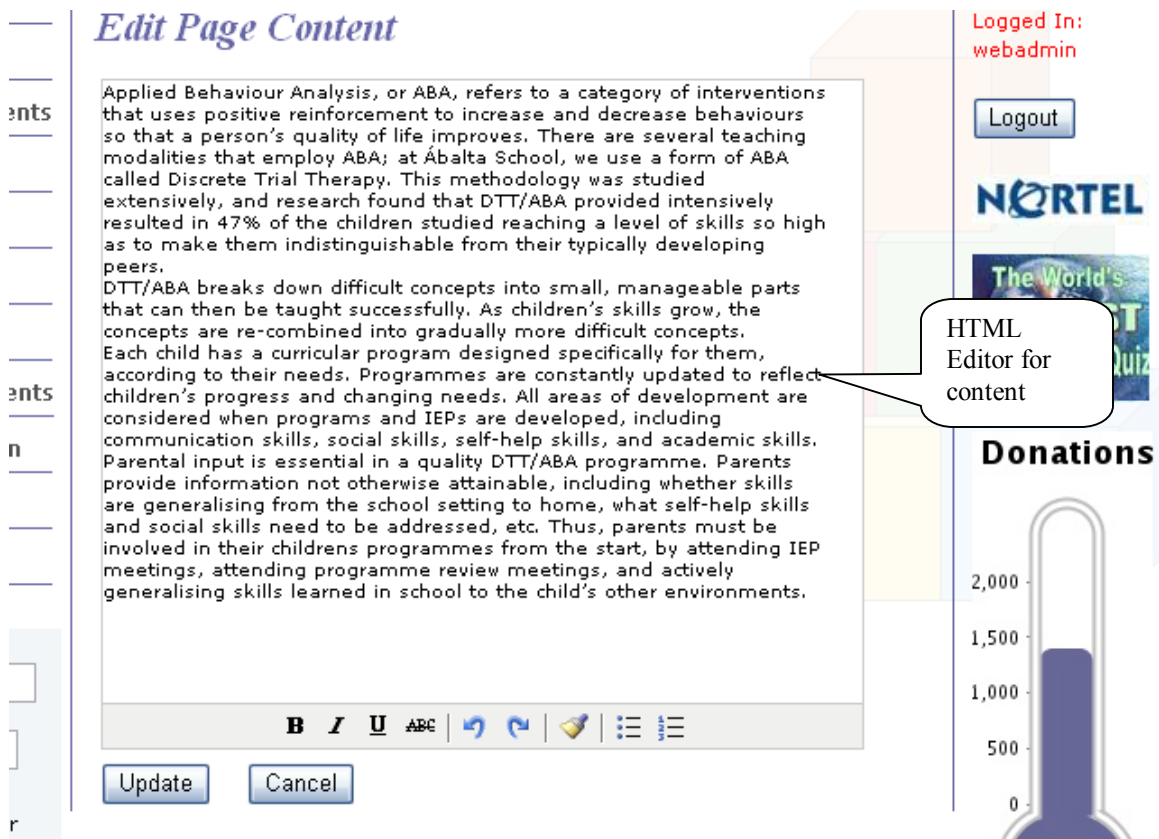


Figure 62 HTML Editor for Content Management System

Gallery

The gallery is a 3-D, interactive collection of photographs from recent fundraisers. The gallery is an open source product in which photographs can be uploaded to.

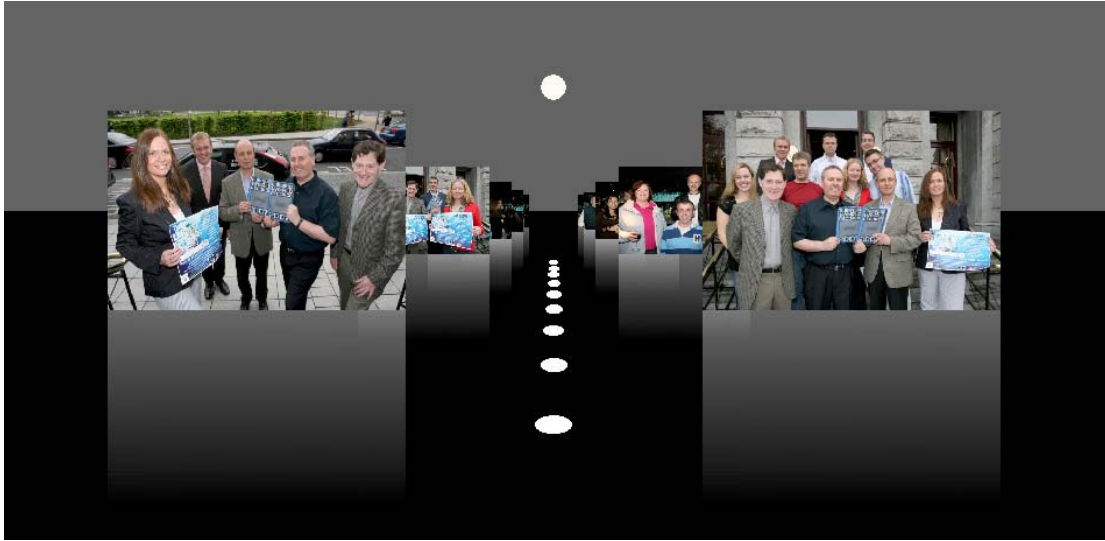


Figure 63 Gallery

Chapter VII Project History

This chapter serves to provide an insight into the evolution of the project from the initial idea of developing a web portal application for Ábalta School to its present state.

How the Project Began

Autism is a disability that affects the normal development of the brain in areas of communication and social interaction. It is a serious developmental neurological disorder, which is marked by severe difficulties in communication and forming relationships with people, in developing language and in using abstract concepts. Characteristics of autism include repetitive and limited patterns of behavior and obsessive resistance to tiny changes in familiar surroundings or routines. In 1998 autism was recognized by the government as a distinct special educational need and there have been significant developments in services, structures and legislative provisions since then. (Irish Autism Action, 2006)

Ábalta School was born out of the identified need for children with autism to get a specific focused education. In July, 2001, four sets of parents came together to establish the school and had it up and running by September, 2001. Initially the school was self funded by the parents of these special children and it took months of hard work, political negotiating, and lobbying, culminating with a high court case to attain funding from the state.

In aiming to raise awareness about autism and the Ábalta school, the parents of the students were asked to give talks / information sessions in their workplace. At these sessions, it became apparent that the only method of information about autism and Ábalta was through brochures, leaflets and word of mouth. However people were looking for more information about the charity, the education they provide in the school and what could be done to help. It was proposed that a web portal application would be set up to assist the charity. This web portal would serve to provide information, support, and awareness about Ábalta and autism and it would help in fundraising for the charity. As the organisation is a charity, it was crucial that web portal application be designed and developed free of charge and any third party resources used be with minimum cost. Being designed and developed as a Masters' thesis project, it met the criteria from a cost perspective and also benefited from researched, up-to-date, modern technology.

Project Management

This project was managed, designed and developed by one person. All activities, including research, design, development, implementation, and testing were performed on a part-time basis during out-of-work hours.

One area of the project, the donations online, required integration with a third party organisation. Communication and correspondence with the third party organisation was achieved through email and by telephone, and code integration was accomplished through email.

Using the extreme programming agile methodology, meetings were setup with the charity committee after every iteration was completed and before a new iteration began. These meetings occurred on the school premises and took place quite regularly to encourage feedback and direction on what the charity wanted and hoped to achieve with the web portal application.

Time management was also a factor. Continuous assessment of the schedule was required during the life cycle of the project. This was to ensure features were prioritised and given appropriate time to design and implement. Due to feature creep, this was a crucial part of managing the project and required focus on many occasions throughout the project life cycle. Time constraints became a big factor in what was delivered in the final project.

Significant Events / Milestones in the Project

Using the extreme programming agile methodology, this project was divided into iterations. Each iteration accounted for a significant milestone in the project and in effect each milestone was delivered and employed as it was completed. The following is the list of milestones reached in this project.

Milestone	Delivered	Duration	Date Delivered
1	Ábalta Online: <ul style="list-style-type: none"> Research appropriate technologies for web portal implementation – database, 	7 weeks	23 rd October 2006

	<p>web application server, architecture framework</p> <ul style="list-style-type: none"> • Design and develop framework architecture • Design and create database for data storage • Graphical user interface design for web portal application • Create prototype for approval 		
2	<p>Autism Information and ABA</p> <p>Methodology:</p> <ul style="list-style-type: none"> • Obtain information regarding autism and ABA for upload • Design and create database tables • Develop code for obtaining data from the database <p>Email:</p> <ul style="list-style-type: none"> • Obtain list of proposed email addresses from Ábalta • Using web portal, create email account for each user <p>Gallery:</p>	3.3 weeks	14 th November 2006

	<ul style="list-style-type: none"> • Research open source galleries for uploading photographs • Integrate with Ábaltas' web portal 		
3	<p>Forum:</p> <ul style="list-style-type: none"> • Research best solution for forum • Design and create database tables • Design and implement Login / Logout feature • Design and implement forum architecture • Design and implement GUI for forum 	4 Weeks	14 th January 2007
4	<p>Donations:</p> <ul style="list-style-type: none"> • Research most appropriate method for submitting donations • Design and implement donation architecture 	7 weeks	3 rd March 2007
5	<p>Content Management System</p> <ul style="list-style-type: none"> • Research existing content 	7 weeks	24 th April 2007

	management systems available <ul style="list-style-type: none"> • Design and create database tables • Design and implement system architecture 		
6	Report on Thesis	5 weeks	End of August 2007

Table 4 Significant Milestones in this project

From the milestone chart, it is evident that initial time estimates on each feature / iteration were not met and were deemed to be underestimated. Due to feature creep, deliverables were constantly revised and prioritised and resulted in certain features, such as the student system, being pushed out of the time scope of this project. The student system will be delivered in the next iteration of the project.

As we were using the extreme programming agile methodology, testing was completed after every iteration was delivered and therefore the schedule was adjusted to emphasise this. One week was scheduled at the end of each milestone to perform appropriate unit, and regression testing.

Changes to the Project Plan

As we were using the extreme programming agile methodology in the design and development of this project, it was relatively uncomplicated to introduce change to the project features and the project schedule. This methodology is designed to deliver the

software customers need when it is needed and empowers developers to confidently respond to changing customer requirements, even when changes are required late in the life cycle. (Don Wells, 2006)

The first change that was incorporated was the donation system. Initially, this was not required and was not part of the project plan. However, during the analysis stage, it became apparent that this feature would greatly benefit the charity and was therefore introduced. As it was introduced early in the life cycle of the project, it was reasonably comfortable to schedule. The feature was prioritized and deemed to be more crucial to the organization than the student system and the schedule was rearranged to reflect this. The student system deliverable dates were extended to allow for the design, development and delivery of the donations system.

In working with third party companies for the donation system feature, there were times when the progression of the project lay with other organizations. It was necessary to set up a merchant account which required charity committee members to visit the bank, sign documents and wait for processing. This resulted in schedule delays and subsequently the project plan required changing to reflect this.

Due to the methodology used, the schedule for each milestone / iteration was also extended to allow for testing to commence upon completion of each milestone deliverable. Initially, it was proposed that testing would instigate at the end of the project when all deliverables were met but due to the web portal features / iterations going "live"

when they had been completed, it was necessary that testing for each feature occur when the feature was completed. The project schedule was updated to reflect this.

Due to feature creep, time constraints and schedule revisions, the student system did not get completed during this life cycle. It is therefore proposed that it be designed, developed and delivered during the next evolution of the web portal application.

Chapter VIII Findings and Analysis

This chapter offers an analysis into the findings and results of this project. The project goals are evaluated and determined whether they have been met or not. The project is analysed to establish what sections or areas went right and according to plan and what sections or areas didn't go particularly well. The findings and analysis of this project are documented and examined.

Analysis of Results

Ábalta School now has a functional web portal application that works to provide information, support, educational methodologies used and to generate funds for the school. From the research, design and development of the application, it was discovered that a custom-made web portal will serve the needs of this charity organisation long after this thesis has been completed, helping to improve awareness of autism and promoting the disability. It was also ascertained that there were many technological routes for achieving this web portal, the research conducted provided an insight into these technologies, and the ones best suited to the tasks were selected for implementation.

Technologies Employed

The technologies used in this project consisted of Java 2 Platform, Enterprise Edition (J2EE) technologies. J2EE is a set of coordinated specifications and practices that together enable solutions for developing, deploying, and managing multi-tier server-centric applications such as Ábaltas' web portal application. J2EE incorporates a wide range of web technologies which facilitated the development web applications in a

consistent and efficient method. Technologies employed include the Model-View-Controller architecture which utilized Apache Struts as the controller providing the structure for the application, JSP and HTML for the view component and JavaBeans, and a MySQL database as the model.

The J2EE platform provides the capabilities necessary for a complete, stable, secure, and fast platform. The portable platform and open source aspect of the technologies made it an ideal choice for building the web portal application. (Java 2 Platform, Enterprise Edition (J2EE) FAQ, 1994-2007)

Java Authentication and Authorization Service (JAAS) is implemented to provide the security required for the login / logout facility. According to Sun Developer Network (n.d.), JAAS is a framework, supplemented with the Java platform, through which users who execute their code, can be required to have explicit permission to perform certain operations. JAAS provides a set of APIs that enable services to authenticate and enforce access controls upon users.

J2EE provides value to an application by significantly reducing the cost, being an open source technology, and complexity of developing and deploying multi-tier solutions, resulting in applications that can be rapidly deployed and easily enhanced. (Java 2 Platform, Enterprise Edition (J2EE) FAQ, 1994-2007)

Resin was implemented as the application server that provides dynamic data or content to the World Wide Web. Resin is a high-performing XML application server used with JSPs, servlets, JavaBeans, XML, and a host of other technologies. (An Overview of Resin (2003))

Feature Prioritization

It was necessary to prioritize features in this project as there was a time constraint on the project life. The time constraint imposed was twelve months which included the writing of this report and any vacation taken.

The prioritizing of features was a constant work-in-progress achieved with the Ábalta committee. Using the extreme programming agile methodology, there were frequent meetings with the committee to determine the course of action and direction of the project. Deliverables were achieved as iterations, and the iterations were extremely modular allowing for changes to be made with minimum impact on the overall project.

The extreme programming agile methodology served as a great model for prioritizing and re-prioritizing features.

Third Party Organization

When it was proposed that the web portal would work as a fundraising application as well as an informational one, it became apparent that due to lack of resources and time constraints, a third party organisation would be required to achieve this.

There were many organisations available for handling and managing online donations and these were investigated to determine the most appropriate for Ábalta. These organisations included utilising a charity website www.mycharity.ie, or implementing with an internet payment solution such as Realex Payments.

MyCharity.ie offers a web page on which an organisation could publicize their charity and fundraise for donations. It requires no software development. The charity organisation provides MyCharity with details and information about their charity which subsequently MyCharity would display on a web page with credit card processing facilities. MyCharity costs €500 for setup and then 3% of every transaction / donation thereafter.

Realex Payments is an Irish based company that offers a secure server with card processing abilities. On investigation it was found that it provided the services most suited to Ábalta by allowing their services to be integrated with Ábaltas' web portal. As Ábalta was in the process of attaining a web portal, it was logical that this application could be utilised for the donations service and providing a one-stop-shop for Ábalta information, support and fundraising.

Cost of Ábalta Web Portal Application

The web portal application incorporated a number of costs during its development. These costs including the hosting of the application, the domain name purchase and the service payments provider costs.

The first cost incorporated was the hosting of the application. For a J2EE application, Hosting Ireland charged a rate of €150 excluding VAT (value added tax). When the VAT was added, the total cost was €181.50 per year.

There were a number of domain names that could have been purchased for the school. The school committee were adamant that the domain name be recognised as an educational authority. The higher education authority donated the www.abaltaabaschool.galway.edu.ie to the school. As this was difficult to find in web searches, Hosting Ireland also donated an alias domain name www.AbaltaSchool.ie. This saved the organisation €140.

Realex Payments typically charges €250 setup fee excluding VAT. For Ábalta, as it is a charity organisation, it charged €150 excluding VAT. For a monthly service fee of €25, Ábalta could receive a maximum of 100 donations. After 100 donations, 25 cents will be charged on all transactions / donations. It was projected that donations would not exceed 100 per month. Total cost of Realex including VAT is €181.50 plus €25 per month.

Evaluation of Project Goals Met

The first and foremost goal of this project was to raise awareness of autism and the Ábalta School in Galway. This objective was indeed the reason that this project was undertaken. Although the school had been operating in Galway since 2001, there was a real lack of information regarding the education and services it provides to children with autism. Does this web portal provide information on autism and Ábalta School and the education it provides? Yes, the first goal achieved was to provide an informational system which imparts information on autism, Ábalta School and the education available for children. This informational system also provides the teachers, administration staff and committee members with email addresses for quick and easy correspondence with other parties.

The second goal was to provide support to the people that are affected by this disability. Support can be achieved in a number of different methods including talking and communicating with people who are in a similar situation, asking questions, providing an opportunity for discussion regarding education, services and the help that is available. Has this been achieved? Yes, this project provides a forum, an opportunity for discussion, for the people who are affected by autism. This forum provides a method of communication between parents, caregivers and interested parties, to exchange encouragement and support and the answering of questions by people who have the experience to answer them.

The third goal of this project was to generate funds and donations for the school.

Although Ábalta is a government recognised school, it is primarily a charity and obtains most of its funding through donations from the public. It is important that there is a facility for generating and obtaining monetary gifts in an easy and straight-forward method. Does the web portal application provide a means for fundraising? Yes, this goal has been achieved two fold. Firstly, there is an area in the web portal that provides information and communication on any fundraising events that may be coming up. People can check in at any time and see if there are any pending events. Secondly, the web portal provides a method of donating money over the internet, online in a safe and secure manner. This enables the global public to donate and contribute to Ábalta at any time, regardless of where they are in the world.

The fourth goal was to ensure that this project was relatively easy to maintain. As there is not a specialized Information Technology person employed by Ábalta, it was optimal that content management be an easy, efficient job for the staff. Is the content of the web portal application easy to maintain? Yes, the web portal incorporates a content management system whereby, upon successful login to the application, an administrator may update content in the web portal.

The fifth goal of this project was to provide a student system for the storing of student records. Due to time constraints this goal has not been met but will be achieved in the next evolution of the system.

What went Right / What went Wrong

Overall the project offered great experience for designing and developing a web portal application. There were some areas of the project that worked very well while other areas didn't.

The extreme programming (XP) agile methodology, which was adhered to, worked very well in determining what the Ábalta committee wanted from the project and the direction in which the project took. Agile software engineering combines a philosophy and a set of development guidelines and the philosophy encourages customer satisfaction and early incremental delivery of software; small, highly motivated project teams; informal methods; minimal software engineering work products and overall development simplicity. (Pressman, 2005) The XP methodology encourages communication with customers which was fundamental to this project and feedback which drove the projects' direction.

The technology used was perfect for the type of project that was being developed. The Model-View-Controller architecture coupled with J2EE technologies provided an ideal platform on which to design and develop this project. The open-source, platform independent features of J2EE made it an ideal choice for the development of a charity web application.

One part of the project that didn't work too well was the time estimates. The time estimates for each feature in the project were miscalculated. The features overran their

planned development time and the schedule suffered. The XP methodology used also requested for testing to be after each feature was delivered and this was not accounted for in the original schedule either.

Findings / Analysis

It has been found that in today's Internet-oriented world, a web portal is essential for businesses, organizations and individuals to optimize their impact on the world: to reach more people, be more efficient, learn more and achieve their goals. The web portal application has provided Ábalta with a means of reaching a global audience of billions of people, in a way that would not be possible without an online application. The web application has provided Ábalta with an international platform on which to raise awareness and support for autism.

This project established that there are many technologies available for the design and development of web applications. These technologies provide different routes on which to achieve the same thing – a web portal application. Upon further investigation, it was found that J2EE offers efficient and effective models, patterns and architectures that were ideal for what this project wanted to achieve. The open source, platform independent features made it a good route for a charity organisation to use. The J2EE community, which included tutorials, forums and examples, was excellent in providing advice, information and experiences to help in the design and development of this project.

The project was completed with minimum cost to the charity and has indeed raised funds for the school in the process. The donations feature has proved successful as it has

enabled people from all over the world to donate and support this worthy charity organisation. It is found that it is important that there be a facility for generating and obtaining monetary gifts in an easy and straight-forward method. In today's technologically advanced world, it is estimated that there is over one billion internet users globally so it was logical that one approach for generating funds would be to utilise the internet through Ábaltas' web portal application.

Conclusion

The conclusion will attempt to define and explain what was learned from the experience of designing and developing Ábaltas' web portal application. It will include what, in hindsight, would have been done or approached differently to ensure a more optimised project experience. The project will be summarised and the initial project expectations will be analysed to determine whether they were met. The next evolution of the project will be discussed and recommendations for the future will be documented.

Lessons Learned

Throughout the duration of this project, there were many lessons learned about the concepts of designing and developing software applications, project management and project architecture. These lessons serve to provide experience for the life cycle of software projects.

One of the most fundamental lessons learned was regarding the timing estimates in the initial plan. Originally, in the project plan, these time estimates had been grossly underestimated and they left no scope for the possibility of the project features being delivered late, or the project direction changing. Due to the time constraint and features being delivered late, there was one feature (student system) of this project that did not get delivered. Had the schedule been correctly estimated, this may not have happened.

In this project, new features were introduced and because of this, priorities changed. It is perhaps almost inevitable, especially when working to a customer specification, that

priorities will change and features will evolve, during the life cycle of the project. Scope should be added to the project plan for this.

Adhering to a design methodology, such as the extreme programming (XP) methodology adhered to in this project, helps to structure the project. A design methodology is a proven solution for projects and is used by experienced project teams globally. The XP methodology encourages communication with customers which was fundamental to this project and customer feedback which drove the projects' direction.

The technologies used in this project proved to be the most suitable when designing web applications. The J2EE platform provides the capabilities necessary for a complete, stable, secure, and fast platform and the portable platform and open source aspect of the technologies made it an ideal choice for building the web portal application. J2EE provides value to an application by significantly reducing the cost, being an open source technology, and complexity of developing and deploying multi-tier solutions, resulting in applications that can be rapidly deployed and easily enhanced. (Java 2 Platform, Enterprise Edition (J2EE) FAQ, 1994-2007) As the technology is open source, there was found to be a vast community of developers available online that could provide tutorials, advice and experience in developing web applications and were very helpful to J2EE novice developers.

What could be done differently

There are two key areas that should be focused on if the project was to be attempted again. These areas are the project schedule and the security of the project.

The project schedule would need to be more realistic to the project plan. The schedule must reflect the project size, the amount of people who will be working on it (and to what extent), and any vacation that people may need to take. Another factor to be accounted for, in the schedule, is to have enough scope for schedule flexibility when working and integrating with third party organizations, especially where the processing of data needs to occur. This project involved integration with a third party organization, however there was not much scope in the schedule to account for this and thus, due to unforeseen circumstances, the project plan was delayed and the schedule suffered.

The JAAS feature provides apt security for this project. However, if there were no time constraints, a suitable encryption algorithm would be imposed on passwords, in the login facility, to provide that little bit of extra security in the project. This will be integrated in the next evolution of the project.

Summary of the Project

In today's Internet-oriented world, a web application is essential for businesses, organizations and individuals to optimize their impact on the world: to reach more people, be more efficient, learn more and achieve their goals. Ábalta is a school, based in Galway, designed to educate children who suffer from autism. As Ábalta School is ever expanding, there is an extensive need for a web portal for the school.

A web portal application will provide the school with a central repository of information, which will be available for any parent or caregiver in need of information regarding autism and the resources and support available for children with autism at any time.

It is necessary that the administration team in Ábalta be able to update this web portal themselves. Therefore the web portal will also provide a content administration management system that will allow an administrator to update content on the web portal.

A web-portal application will provide a method for fundraising online, to utilize the success of the internet to help its cause, and to provide support and communication for people who are affected by autism.

For Ábaltas' web portal to be successful, it was necessary to gauge a complete set of requirements for the application from the people who would be utilizing it – the directors and staff of Ábalta. One of the first tasks undertaken was to meet with the charity committee to obtain the requirements for what they would like the system to do and achieve. One of the priorities of the system was that it provides information on autism and support to people who are affected by the disability.

Appropriate technologies were researched in order to determine the most suitable technologies in which to develop this application. These technologies covered a wide range of project development including the architecture in which the project was

designed, the frameworks utilized, the security of the application and the building and deployment of the project.

It was necessary to integrate this project with a third party organization for the donations / fundraising feature. The donation feature requires the processing of credit and debit cards and a third party organization could provide a secured server in order to process the cards in a secure and safe method. The third party organization – Realex Service Providers – enabled Ábalta to successfully raise funds on the internet efficiently and securely.

A content management system was required to allow an administrator of Ábalta to update the web portal on demand. It was necessary that this be low maintenance, easy to use and efficient as there is no full time staff member in Ábalta who is technically apt at maintaining web portals.

Throughout the duration of the project, the Ábalta charity committee were constantly met with and updated as to the process of the web portal. This maintained that the project was going in the right direction i.e. the direction that the Ábalta committee wanted it to take. These meetings encouraged ideas and feedback and provided scenarios for testing the project releases.

The project has been deployed in iterations. These iterations have added to the web portal application with each release. Using the web statistics supplied by the hosting company –

Hosting Ireland – it has been estimated that there are, on average, 174 “hits” to the Ábalta web portal daily and there has already been €1400.00 donated to the charity online. This web portal application has proved successful for Ábalta School for Children with Autism.

Did the Project Meet Expectations?

The primary expectation of this project was to raise awareness and knowledge of autism and the Ábalta School in Galway. Although the school had been operating in Galway since 2001, there was a real lack of knowledge and understanding of the education and services that it provides to children affected with autism. This expectation has been achieved as the web portal, which has been subsequently developed for this project, provides an informational system which communicates information on autism, Ábalta School and the education and its methodology available for children in Galway. This informational system also provides the teachers, administration staff and committee members with email addresses for quick and easy correspondence with other parties.

The second expectation of this project was to provide support to the community of people that are affected, either directly or indirectly, by this disability. Support can be achieved in a number of different methods including talking and communicating with people who are in a similar situation, asking questions, providing an opportunity for discussion regarding education, services and the help that is available. This expectation has been realized as the web portal application provides a forum, an opportunity for discussion, for the people who are affected by autism. This forum provides a method of communication between parents, caregivers and interested parties, to exchange encouragement and

support and the answering of questions by people who have the experience to answer them.

The third expectation of this project was to help to generate funds and donations for the school. Although Ábalta is a government recognised school, it is primarily a charity and obtains most of its funding through donations from the public. It is important that there is a facility for generating and obtaining monetary gifts in an easy and straight-forward method. This expectation has been met on two levels. Firstly, there is an area in the web portal that provides information and communication on any fundraising events that may be coming up. People can view this area at any time and see if there are any pending events. Secondly, the web portal provides a method of donating money over the internet, online in a safe and secure manner. This enables the global public to donate and contribute to Ábalta at any time, regardless of where they are in the world. At the time of publication, this facility has accepted donations of €1400.00 for Ábalta School for Children with Autism deeming that this expectation has been significantly surpassed.

The fourth expectation was to ensure that this web portal was relatively easy to maintain. As there is not a specialized Information Technology person employed by Ábalta, it was optimal that content management be an easy, efficient job for the staff. The web portal incorporates a content management system whereby, upon successful login to the application, an administrator may update content in the web portal.

The fifth expectation of this project was to find the most appropriate and suitable technologies for building the web portal application. Web portal technologies were researched and investigated, prototyped and analysed to determine which technological route to take for the development of the software. This expectation was achieved as the project has been designed and developed in an efficient and architecturally proven way with up-to-date technology.

The sixth expectation of this project was to provide a student system for the storing of student records. Due to time constraints, scheduling problems and a delay in some deliverables, this expectation has not been met but will be achieved in the next evolution of the system.

Next Stage of Evolution

The next iteration of the software will include the release of the Student System. This will be a system that will facilitate the adding, updating and deleting of students from the web portal application database. This system will allow the teachers and educators the facility to record in the web portal, weekly analysis of students' behaviour, achievements and difficulties incurred. It is hoped that this data could be automatically merged into a report and sent to the parents / caregivers to allow for weekly statements on the progress of their children.

Due to the sensitive nature of these reports, it is hoped that the next iteration will also include an encryption algorithm for the login facility of the web portal. An encryption

algorithm will further enhance the security that is already apart of the web portal application and protect against unauthorised access into the system.

Conclusions / Recommendations

This project has provided Ábalta School for Children with Autism with an opportunity for acquiring a web portal application that will benefit the charity immensely. This web portal application serves to provide the global public with information and knowledge of autism, provide support for people affected by the disability and detail the education methodology they follow in the school for autistic children.

Although Ábalta is a government recognised school, it is primarily a charity and obtains most of its funding through donations from the public. It was important that there was a facility developed for generating and obtaining monetary gifts in an easy and straight-forward method. In today's technologically advanced world, it is estimated that there is over one billion internet users globally so it was logical that one approach for generating funds would be to utilise the internet through Ábaltas' web portal application, allowing for online donations which could never be achieved prior to having a web portal application.

A content management system offers the staff of Ábalta the opportunity of maintaining the web portal and becoming responsible for all the data that may be displayed. This instils a confidence into staff members of using and modifying the web application as appropriate.

The web portal application has provided Ábalta School for Children with Autism with a channel for promoting the school, the education it provides and information about autism to billions of people across the world.

This project ascertained that there were many technological routes for achieving a web portal, research was conducted which provided an insight into these technologies, and the ones best suited to the tasks were selected for implementation.

The extreme programming (XP) agile methodology, which was adhered to throughout the life cycle of this project, worked very well in determining what the Ábalta committee wanted from the project and the direction in which the project took. Agile software engineering combines a philosophy and a set of development guidelines and the philosophy encourages customer satisfaction and early incremental delivery of software; small, highly motivated project teams; informal methods; minimal software engineering work products and overall development simplicity. (Pressman, 2005) The XP methodology encourages communication with customers which was fundamental to this project and customer feedback which drove the projects' direction.

The technologies used in this project consisted of Java 2 Platform, Enterprise Edition (J2EE) technologies, which are a set of coordinated specifications and practices that together enable solutions for developing, deploying, and managing multi-tier server-centric applications such as Ábalta's web portal application. J2EE incorporates a wide range of web technologies which facilitated the development of web applications in a

consistent and efficient method. Technologies employed include the Model-View-Controller architecture which utilized Apache Struts as the controller providing the structure for the application, JSP and HTML for the view component and JavaBeans, and a MySQL database as the model.

J2EE provided value to Ábaltas' web portal application by significantly reducing the cost, being an open source technology, and complexity of developing and deploying multi-tier solutions, resulting in applications that can be rapidly deployed and easily enhanced. (Java 2 Platform, Enterprise Edition (J2EE) FAQ, 1994-2007)

In reviewing and analysing the findings and the literature, the design and development of Ábaltas' web portal application has provided this author with an in-depth knowledge and understanding of web technology. The project covered a vast range of technologies, including database design and development, J2EE technologies, application servers and build frameworks. The project also encompassed process methodologies which are internationally respected.

The project management provided experience in multitasking. Not only had this author to manage the design and development of the web portal, it was necessary to control the project, integrate with third party organizations and continuously meet with the charity committee.

This author also gained skills in researching appropriate technologies / solutions for a given problem. Investigation and research of literature and technologies provided a means for objectively making a choice on the best approach for achieving the solution of a web portal application for Ábalta. Online tutorials, articles and forums were investigated and undertaken for knowledge and technical transfer.

This project has also provided an insight into the huge amount of work that charities accomplish and the determined volunteers that work in them.

Appendix A The View Component Code

Header.jsp

```
<%
    ie.edu.galway.abaltaabaschool.stats.Donations d = new
ie.edu.galway.abaltaabaschool.stats.Donations();
    String csrGoalsFileName = d.createChart(d.getDonationDataset(),
session);
    String csrGoalsGraphURL = request.getContextPath() +
"/servlet/DisplayChart?filename=" + csrGoalsFileName;
%>

<@ taglib uri="/tags/struts-logic" prefix="logic" %>
<@ taglib uri="/tags/struts-bean" prefix="bean"%>
<@ taglib uri="/taglib37.tld" prefix="set" %>

<%
    boolean isLoggedIn = false;

    if (request.getUserPrincipal() != null)
        isLoggedIn = true;
%>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<head>
<meta http-equiv="content-type" content="text/html; charset=utf-8" />
<meta name="description" content="" />
<meta name="keywords" content="Abalta, autism, autistic, children,
galway, school" />
<meta name="author" content="" />
<link rel="stylesheet" type="text/css" href="/css/layout.css"
media="screen" title="" />
<title>Abalta School</title>

<style type="text/css">
#thermometer{
background: url(<%=csrGoalsGraphURL%>);
width: 110px;
height: 300px;
}
</style>

<script language="javascript" type="text/javascript"
src="../../jscripsts/tiny_mice/tiny_mce.js"></script>
<script language="javascript" type="text/javascript">
```

```

tinyMCE.init({
    mode : "textareas",
    theme : "simple",
    force_br_newlines : true
});
</script>
</head>

<div id="top">
</div>

<body><div id="wrap">

<div id="logo">
</div>

<div id="banner">
</div>

<div id="logo2">
</div>

<div id="avmenu">
<h2 class="hide">Menu:</h2>
<ul>
<li><a href="/index.jsp"><bean:message key="menu.label.home"/></a></li>
<li><a href="/info/gen.do?page_name=aba.jsp"><bean:message
key="menu.label.aba"/></a></li>
<li><a href="/info/gen.do?page_name=mission.jsp"><bean:message
key="menu.label.mission"/></a></li>
<li><a href="/info/gen.do?page_name=testimonials.jsp"><bean:message
key="menu.label.testimonials"/></a></li>
<li><a href="/info/gen.do?page_name=literature.jsp"><bean:message
key="menu.label.literature"/></a></li>
<!--<li><a href="/forum/forum.do"><bean:message
key="menu.label.forum"/></a></li>-->
<li><a href="/info/faq.do"><bean:message
key="menu.label.faq"/></a></li>
<li><a href="/fundraising/fundraiser.do"><bean:message
key="menu.label.fundraiser"/></a></li>
<li><a href="/donate/donations.do"><bean:message
key="menu.label.donations"/></a></li>
<li><a href="/gallery/gallery.html">Gallery</a></li>
<li><a href="/info/gen.do?page_name=contact.jsp"><bean:message
key="menu.label.contact"/></a></li>
</ul>

<!--<div class="announce">

<form action="j_security_check" method="POST">
<strong><bean:message key="menu.label.uname"/></strong><input
name="j_username" type="text" size="16"/>
<strong><bean:message key="menu.label.pword"/></strong><input
name="j_password" type="password" size="16"/><br />
<input name="j_uri" type="hidden" value="/Home.do" />
<br/><input type="submit" disabled <%if(isLoggedIn)
out.println("disabled");%> value="Login" />

```

[illegible]

```
</div>
```

```
<div id="content">
```

```
Footer.jsp
```

```
</div>
```

```
<div id="footer">
```

```
</div>
```

```
<div id="top">
```

```
</div>
```

```
</div>
```

```
</body>
```

```
</html>
```

```
Index.jsp
```

```
<%@ taglib uri="/tags/struts-logic" prefix="logic" %>
```

```
<logic:redirect forward="home"/>
```

```
construction.jsp
```

```
<%@ page import="javax.naming.InitialContext,
    javax.naming.Context,
    java.util.Properties,
    java.sql.*,
    java.util.*,
    ie.edu.galway.abaltaabaschool.database.DBApp"%>
```

```
<%@ include file="../../header.jsp" %>
```

```
<h4>
```

```
Sorry, this page is currently under construction!
```

```
<div id="construction">
```

```
</div>
```

```
</h4>
```

```
<%@ include file="../../footer.jsp" %>
```

FAQ.jsp

```
<%@ page import="javax.naming.InitialContext,
                javax.naming.Context,
                ie.edu.galway.abaltaabaschool.beans.FaqBean,
                ie.edu.galway.abaltaabaschool.businessobjects.Faq,
                java.util.List,
                java.util.ArrayList,
                java.util.Iterator"%>

<%
    FaqBean fB = (FaqBean) request.getAttribute("FaqBean");
%>

<%@ include file="../header.jsp" %>

<h3><bean:message key="faq.label.title"/></h3>

<%
    List faqList = fB.getFaq();
    Iterator it = faqList.iterator();
    Faq faq;
    while(it != null && it.hasNext())
    {
        faq = (Faq) it.next();
    }
%>
    <a href="#<%out.println(faq.getId());%>"><% out.println(faq.getQuestion());
%></a><br>
    <%
    }
%>

<br><br>
<%
    it = faqList.iterator();
    while(it != null && it.hasNext())
    {
        faq = (Faq) it.next();

        <a name=
        <%
            out.println(faq.getId());
        %>>
        <%
            out.println(faq.getQuestion());
        %>
        </a>
        <%
            out.println(faq.getAnswer());
        %>
    }
%>
```

```

        %>
        <br><br><br>
    <%
    }
    %>

<%@ include file="../footer.jsp" %>

-----

gen.jsp

-----

<%@ page import="javax.naming.InitialContext,
                javax.naming.Context,
                ie.edu.galway.abaltaabaschool.beans.HomeBean"%>

<%
    HomeBean hB = (HomeBean) request.getAttribute("HomeBean");
%>

<%@ include file="../header.jsp" %>

<h3>
<%
    out.println(hB.getTitle());
%>
</h3>

<%
    out.println(hB.getData());
%>

<%@ include file="../footer.jsp" %>

-----

home.jsp

-----

<%@ page import="javax.naming.InitialContext,
                javax.naming.Context,
                ie.edu.galway.abaltaabaschool.beans.HomeBean"%>

<%
    HomeBean hB = (HomeBean) request.getAttribute("HomeBean");
%>

<%@ include file="../header.jsp" %>

<h3>
<%
    out.println(hB.getTitle());
//    System.out.println("user roles ===== "+request.getUserRoles());
%>

```



```
</h3>
```

```
<%  
    out.println(hB.getData());  
%>
```

```
<%@ include file="../footer.jsp" %>
```

loginerror.jsp

```
<%@ taglib uri="/tags/struts-bean" prefix="bean"%>  
<%@ include file="../header.jsp" %>  
  
<h3><bean:message key="login.label.incorrect"/></h3>  
  
<bean:message key="login.label.error"/>  
<br/>  
<bean:message key="login.label.retry"/>  
  
<%@ include file="../footer.jsp" %>
```

register.jsp

```
<%@ taglib uri="/tags/struts-bean" prefix="bean"%>  
<%@ taglib uri="/tags/struts-html" prefix="html"%>  
  
<%@ include file="../header.jsp"%>  
  
<html:form action="/submitRegister" method="post">  
  
<h3><bean:message key="register.label.title"/></h3>  
  
<table>  
    <tr>  
        <td align="center" colspan="3">  
            <h4.5><i><b><bean:message  
key="register.label.details"/></b></i></h4.5>  
        </tr>  
        <tr>  
            <td align="left" colspan="2">  
                <font color="red"><html:errors/></font>  
            </tr>  
        <tr>  
            <td align="right">  
                <bean:message key="register.label.name"/>  
            </td>  
            <td align="left">  
                <html:text property="name" size="30" maxlength="30"/>  
            </td>  
        </tr>  
    </table>
```

```

        </td>
    </tr>
    <tr>
        <td align="right">
            <bean:message key="register.label.address"/>
        </td>
        <td align="left">
            <html:text property="address" size="30" maxlength="30"/>
        </td>
    </tr>
    <tr>
        <td align="right">
            <bean:message key="register.label.phone"/>
        </td>
        <td align="left">
            <html:text property="phone" size="30" maxlength="30"/>
        </td>
    </tr>
    <tr>
        <td align="right">
            <bean:message key="register.label.email"/>
        </td>
        <td align="left">
            <html:text property="email" size="30" maxlength="30"/>
        </td>
    </tr>
    <tr>
        <td align="right">
            <bean:message key="register.label.uname"/>
        </td>
        <td align="left">
            <html:text property="username" size="30" maxlength="30"/>
        </td>
    </tr>
    <tr>
        <td align="right">
            <bean:message key="register.label.password"/>
        </td>
        <td align="left">
            <html:password property="password" size="30" maxlength="30"/>
        </td>
    </tr>
    <tr>
        <td align="right">
            <bean:message key="register.label.confirm.password"/>
        </td>
        <td align="left">
            <html:password property="cpassword" size="30" maxlength="30"/>
        </td>
    </tr>
    <tr>
        <td align="right">
            <html:submit><bean:message
key="register.label.reg"/></html:submit>
        </td>
        <td align="left">

```

```

        <html:cancel><bean:message
key="register.label.cancel"/></html:cancel>
    </td>
</tr>
</table>
</html:form>

```

```

<%@ include file="../footer.jsp"%>

```

success.jsp

```

<%@ taglib uri="/tags/struts-bean" prefix="bean"%>
<%@ include file="../header.jsp"%>
<h3><bean:message key="login.label.success"/></h3>
<bean:message key="login.label.success.successful"/>
<%@ include file="../footer.jsp"%>

```

authorised.jsp

```

<%@ taglib uri="/tags/struts-bean" prefix="bean"%>
<%@ include file="../header.jsp"%>
<h3><bean:message key="login.label.unauth"/></h3>
<bean:message key="login.label.unauth.access"/>
<%@ include file="../footer.jsp" %>

```

fundraiser.jsp

```

<%@ page import="ie.edu.galway.abaltaabaschool.stats.*,
java.io.PrintWriter"%>

```

```

<%@ include file="../header.jsp"%>

```

[illegible]

	PJ Flaherty in Salthill
	Walshes in Dunmore
	Richardsons in Eyre Square
	Red-Gap Barndaragh
	Reapy's in Tuam
	Kinsella's in Loughrea
	Keane's in Oranmore
	Menlo Park Hotel
	Sherry's in Clarenbridge
	The Twelve in Barna
	Rafferty's Rest in Kilcolgan
	Lohan's in Salthill

<@ include file="../../footer.jsp" %>

forum.jsp

```
<%@ page import="javax.naming.InitialContext,
               javax.naming.Context,
               ie.edu.galway.abaltaabaschool.beans.ForumBean,
               ie.edu.galway.abaltaabaschool.businessobjects.Forum,
               java.util.List,
               java.util.ArrayList,
               java.util.Iterator"%>

<%
    ForumBean fB = (ForumBean) request.getAttribute("ForumBean");
%>

<%@ include file="../header.jsp" %>
<link rel="stylesheet" type="text/css" href="/css/forum.css"
media="screen" title="" />

<h3><bean:message key="forum.label.title"/></h3>

<br>
<%
    List forumList = fB.getForum();
    Iterator it = forumList.iterator();
    Forum forum;
    while(it != null && it.hasNext())
    {
        forum = (Forum) it.next();
%>
        <div id = forumTitle>
             <a
href="/forum/forumMessages.do?forumid=<%=forum.getId()%>"><%=forum.getF
orumName()%></a><br><br>
        </div>
<%
    }
%>

<br><br><br><br><br><br><br><br><br><br><br><br><br><br>
<%@ include file="../footer.jsp" %>
```

forumMessages.jsp

```
<%@ page import="javax.naming.InitialContext,
               javax.naming.Context,
               ie.edu.galway.abaltaabaschool.beans.ForumMessageBean,
               ie.edu.galway.abaltaabaschool.businessobjects.ForumMessages,
```

```

        java.util.List,
        java.util.ArrayList,
        java.util.Iterator,
        java.text.*,
        java.util.*"%>

<%@ include file="../header.jsp" %>

<%
    ForumMessageBean fMB = (ForumMessageBean)
request.getAttribute("ForumMessageBean");
    String visible = "hidden";

    List forumList = fMB.getForumMsg();
    Iterator it = forumList.iterator();
    ForumMessages msg;
    String details;
    int selectedMsg = 0;

%>

<link rel="stylesheet" type="text/css" href="/css/forum.css"
media="screen" title="" />

<h3>
<%=session.getAttribute("ForumName")%>
</h3>

<div id=links>
    <a href=/forum/forumNewPost.do?newpost=true><bean:message
key="forum.new.message"/></a> |
    <a href=/forum/forumSearch.do><bean:message
key="forum.search.message"/></a> |
    <a href=/forum/forum.do><bean:message
key="forum.exit.message"/></a>
</div>
<br>

<div style="overflow:auto; height:250px; width:460px;">

<%
    String bgcolor = "FFFFFF";
    while(it != null && it.hasNext())
    {
        msg = (ForumMessages) it.next();

        if(bgcolor.equals("FFFFFF")){
            bgcolor="F5F5F5";
        }else{
            bgcolor = "FFFFFF";
        }
    }
%>

<table width=100% cellpadding=0 cellspacing=0 border=0>
    <tr>
        <td width=40% bgcolor="<%=bgcolor%>">

```

```

        <%=msg.getIndent()%><a
href="/forum/forumMessages.do?msgId=<%=msg.getMsg_id()%>&forumid=<%=msg
.getF_id()%>">
        <%=msg.getSubject()%></a>
        </td>
        <td width=20% bgcolor="<%=bgcolor%>">
            <%=msg.getUser_name()%>
        </td>
        <td width=40% bgcolor="<%=bgcolor%>">
<%
String dt = msg.getDate_time();
String ww = "";
SimpleDateFormat shortTime = new SimpleDateFormat("yyyy-MM-dd
hh:mm:ss.S");
Date dd = shortTime.parse(dt);

shortTime = new SimpleDateFormat("hh:mm a, dd MMM yy");

out.println(shortTime.format(dd));
%>
        </td>
        <tr>
</table>
<%
    }
%>

</div>

<div style="overflow:auto; height:250px; width:400px;">
<%
    it = forumList.iterator();
    while(it != null && it.hasNext()){
        msg = (ForumMessages) it.next();
        if
(String.valueOf(msg.getMsg_id()).equals(session.getAttribute("msgId")))
    {
%>
        <hr noshade size=1 width=100%>
        <table>
            <tr>
                <td width=50%>
                    <h4><B><%out.println(msg.getSubject());%></B><h4>
                </td>
            </tr>
            <tr>
                <td width=100%>
                    <B><%out.println(msg.getUser_name());%></B>
                    @
                <%
                    String dt = msg.getDate_time();
                    String ww = "";
                    SimpleDateFormat shortTime = new
SimpleDateFormat("yyyy-MM-dd hh:mm:ss.S");
                    Date dd = shortTime.parse(dt);

```



```

shortTime = new SimpleDateFormat("hh:mm
a, dd MMM yy");
out.println(shortTime.format(dd));
%>

</td>
</tr>
</table>
<hr noshade size=1 width=100%>
<%=msg.getMessage()%>
<%
    session.setAttribute("p_msg_id",
String.valueOf(msg.getMsg_id()));
    visible="visible";
}
}

session.setAttribute("msgId", null);
%>

</div>
<div style="width:400px; visibility:<%=visible%>;">
    <hr noshade size=1 width=100%>
    <br><br>
    <a href=/forum/forumNewPost.do><bean:message
key="forum.reply.message"/></a>
    <!--<a href=/forum.jsp><bean:message key="forum.next.message"/></a> |
    <a href=/forum.jsp><bean:message key="forum.next.previous"/></a>-->
</div>

<%@ include file="../footer.jsp" %>

-----

forumMsg.jsp

-----

<%@ taglib uri="/tags/struts-bean" prefix="bean"%>

<link rel="stylesheet" type="text/css" href="/css/forum.css"
media="screen" title="" />

<div id="messages">
    <a href=/forum.jsp><bean:message key="forum.reply.message"/></a>
    <a href=/forum.jsp><bean:message key="forum.next.message"/></a>
</div>

-----

forumNewPost.jsp

-----

```

```

<%@ page import="javax.naming.InitialContext,
                javax.naming.Context,
                ie.edu.galway.abaltaabaschool.beans.ForumNewPostBean,
                java.util.List,
                java.util.ArrayList,
                java.util.Iterator"%>

<%@ taglib uri="/tags/struts-html" prefix="html" %>

<%
    System.out.println("New pst jsp " + request.getParameter("newpost")+
    " " + request.getAttribute("newpost"));

    String reply = "";
    ForumNewPostBean fNPB = (ForumNewPostBean)
request.getAttribute("ForumNewPostBean");
    if (fNPB != null && fNPB.getSSubject() != null ) {
        if(fNPB.getSSubject().indexOf("RE:") != -1){
            reply = fNPB.getSSubject();
        }else{
            reply = "RE: "+fNPB.getSSubject();
        }
    }
%>

<%@ include file="../header.jsp"%>

<html:form action="/submitForumPost" method="post">

<link rel="stylesheet" type="text/css" href="/css/forum.css"
media="screen" title="" />

<h3>
    <%=session.getAttribute("ForumName")%>
</h3>
<p><bean:message key="forum.post.new"/></p>

<input type="hidden" name="fName"
value="<%=request.getParameter("fName")%>"/>

<table>
<tr></tr><tr></tr>
    <tr>
        <td align="right">
            <bean:message key="forum.post.subject"/>
        </td>
        <td align="left">
            <html:text property="subject" value="<%=reply%>"
size="50" maxlength="150"/>
        </td>
    </tr>

    <tr>

```

```

        <td align="right">
            <bean:message key="forum.post.msg"/>
        </td>
        <td align="left">
            <html:textarea property="message" rows="15"
cols="40"/>
        </td>
    </tr></tr><td align="left"></td>
    <!--<td align="right">
        <html:submit><bean:message
key="forum.label.attach"/></html:submit>
    </td>-->
</tr>

<tr>
    <td align="right">
        <html:submit><bean:message
key="forum.label.post"/></html:submit>
    </td>
    <td align="left">
        <!--<html:cancel><bean:message
key="forum.label.preview"/></html:cancel>-->
        <html:cancel><bean:message
key="forum.label.cancel"/></html:cancel>
    </td>
</tr>
</table>

</html:form>

<%
    if (fNPB != null){
        out.println("<hr noshade size=1 width=\"100%\">");
    %>
    <h3><%out.println(fNPB.getSSubject());%></h3>
    <%out.println(fNPB.getSUsername());%>
    <hr noshade size=1 width=100%>
    <%
        out.println(fNPB.getSMsg());
    %>
    %>
<%@ include file="../../footer.jsp" %>

```

forumSearch.jsp

```

<%@page import="java.util.Vector"%>
<%@ taglib uri="/tags/struts-bean" prefix="bean"%>
<%@ taglib uri="/tags/struts-html" prefix="html" %>

<%@ include file="../../header.jsp" %>

```

```

<%
    Vector mon = new Vector();
    mon.add("Jan"); mon.add("Feb"); mon.add("Mar"); mon.add("Apr");
mon.add("May"); mon.add("Jun");
    mon.add("Jul"); mon.add("Aug"); mon.add("Sept"); mon.add("Oct");
mon.add("Nov"); mon.add("Dec");
    pageContext.setAttribute("mon", mon);

    Vector day = new Vector();
    day.add("1"); day.add("2"); day.add("3"); day.add("4"); day.add("5");
day.add("6"); day.add("7");
    day.add("8"); day.add("9"); day.add("10"); day.add("11");
day.add("12"); day.add("13"); day.add("14");
    day.add("15"); day.add("16"); day.add("17"); day.add("18");
day.add("19"); day.add("20"); day.add("21");
    day.add("22"); day.add("23"); day.add("24"); day.add("25");
day.add("26"); day.add("27"); day.add("28");
    day.add("29"); day.add("30"); day.add("31");
    pageContext.setAttribute("day", day);

    Vector year = new Vector();
    year.add("2007"); year.add("2008"); year.add("2009");
    pageContext.setAttribute("year", year);
%>

<html:form action="/submitForumSearch" method="post">

<link rel="stylesheet" type="text/css" href="/css/forum.css"
media="screen" title="" />

<h3><bean:message key="forum.search.title"/></h3>

<table>
<tr></tr><tr></tr>
    <tr>
        <td align="left" colspan="2">
            <font color="red"><html:errors/></font>
        </tr>
        <tr>
            <td width="30%">
                <bean:message key="forum.search.for"/>
            </td>
            <td width="70%">
                <html:text property="keyword" size="40"
maxlength="70"/>
            </td>
        </tr>
    </tr></tr><tr></tr>
</table>
<table>
    <BR>
    <tr><h5><bean:message key="forum.label.refine"/><h5></tr>
    <tr><td><hr width=300%></td></tr>
    <tr>
        <td width="25%">
            <bean:message key="forum.label.after"/>
        </td>

```

```

        <td width="10%">
            <html:checkbox property="checkAfter"
value="yes"/>
        </td>
        <td width="10%">
            <html:select property="afterDay">
                <html:options name="day"/>
            </html:select>
        </td>
        <td width="10%">
            <html:select property="afterMon">
                <html:options name="mon"/>
            </html:select>
        </td>

        <td width="10%">
            <html:select property="afterYear">
                <html:options name="year"/>
            </html:select>
        </td>
    </tr>
    <tr>

        <td width="25%">
            <bean:message key="forum.label.before"/>
        </td>
        <td width="10%">
            <html:checkbox property="checkBefore"
value="yes"/>
        </td>
        <td width="10%">
            <html:select property="beforeDay">
                <html:options name="day"/>
            </html:select>
        </td>
        <td width="10%">
            <html:select property="beforeMon">
                <html:options name="mon"/>
            </html:select>
        </td>

        <td width="10%">
            <html:select property="beforeYear">
                <html:options name="year"/>
            </html:select>
        </td>
    </tr>

    <tr><td><hr width=300%></td></tr>
    <tr>

        <td width="25%">
            <bean:message key="forum.label.subject"/>
        </td>
        <td width="10%">
            <html:checkbox property="checkSubOnly"
value="yes"/>
        </td>
    </tr>

```

```

        </tr>
        <tr>
            <td width="25%">
                <bean:message key="forum.label.message" />
            </td>
            <td width="10%">
                <html:checkbox property="checkMsgOnly"
value="yes" />
            </td>
        </tr>
        <tr>
            <td width="25%">
                <bean:message key="forum.label.user" />
            </td>
            <td width="10%">
                <html:checkbox property="checkUserOnly"
value="yes" />
            </td>
        </tr>
        <tr><td><hr width=300%></td></tr>
    </table>
    <BR>
    <table>
        <tr>
            <td></td>
            <td align="right">
                <html:submit><bean:message
key="forum.label.find" /></html:submit>
            </td>
            <td align="left">
                <html:cancel><bean:message
key="forum.label.cancel" /></html:cancel>
            </td>
        </tr>
    </table>

    <%@ include file="../../footer.jsp" %>

    -----

    forumSearchResult.jsp

    -----

    <%@ page import="javax.naming.InitialContext,
        javax.naming.Context,
        ie.edu.galway.abaltaabaschool.beans.ForumSearchBean,

        ie.edu.galway.abaltaabaschool.businessobjects.ForumSearch,
        java.util.List,
        java.util.ArrayList,
        java.util.Iterator,
        java.text.*,
        java.util.*"%>

```

```

<%@ include file="../header.jsp" %>

<%
    ForumSearchBean fSB = (ForumSearchBean)
request.getAttribute("ForumSearchBean");

%>
<link rel="stylesheet" type="text/css" href="/css/forum.css"
media="screen" title="" />
<h3><%=session.getAttribute("ForumName")%></h3>
<table width = 95% cellspacing=0 cellpadding=0 border=0>

<%

    List forumList = fSB.getForumMsg();
    Iterator it = forumList.iterator();
    ForumSearch msg;

    while(it != null && it.hasNext())
    {
        System.out.println("In iterator ~~~~~");
        msg = (ForumSearch) it.next();%>

        <tr>
            <a
href="/forum/forumMessages.do?msgId=<%=msg.getMsg_id()%>&forumid=<%=msg
.getF_id()%>">
                <%=msg.getSubject()%></a>
            </tr>

            <tr>
                <%=msg.getMsgText()%>
            </tr>

            <tr><FONT SIZE="-2">
                Date Posted:
                <%
                    String dt = msg.getDate_time();
                    String ww = "";
                    SimpleDateFormat shortTime = new
SimpleDateFormat("yyyy-MM-dd hh:mm:ss.S");
                    Date dd = shortTime.parse(dt);
                    shortTime = new SimpleDateFormat("hh:mm a, dd MMM
yy");
                    out.println(shortTime.format(dd));
                %>

                </FONT>
            </tr>
            <br>

        <%    }
    %>
</table>

```

```
<%@ include file="../../../footer.jsp" %>
```

donations.jsp

```
<%@ taglib uri="/tags/struts-bean" prefix="bean"%>
```

```
<%@ taglib uri="/tags/struts-html" prefix="html"%>
```

```
<%@ include file="../../../header.jsp" %>
```

```
<h3><bean:message key="donations.label.title"/></h3>
```

```
<IFRAME width="462px" frameborder="0" height="550px"
src="/donate/donations_frame.do">
```

```
</IFRAME>
```

```
<%@ include file="../../../footer.jsp" %>
```

donations_frame.jsp

```
<%@ taglib uri="/tags/struts-bean" prefix="bean"%>
```

```
<%@ taglib uri="/tags/struts-html" prefix="html"%>
```

```
<jsp:useBean id="donationsBean" scope="request"
class="ie.edu.galway.abaltaabaschool.beans.Donations" />
```

```
<HTML>
```

```
<HEAD>
```

```
<style>
```

```
fieldset{
```

```
border: 1px solid #781351;
```

```
font-size: 10pt;
```

```
}
```

```
legend{
```

```
color:#000000;
```

```
font-weight: bold;
```

```
font-size: 10pt;
```

```
}
```

```
body {
```

```
margin: 0 auto;
```

```
padding: 0;
```

```
font: 76% Verdana,Tahoma,Arial,sans-serif;
```

```
}
```

```
</style>
```



```

</HEAD>
<BODY>

<html:form action="/submitDonation" method="post">

<bean:message key="donations.label.thank"/>
<br><br>
<fieldset>
<legend>Contact Details</legend>
<table style="font: 9pt Verdana,Tahoma,Arial,sans-serif;">
<tr>

<tr>
  <td align="left" colspan="2">
    <font color="red"><html:errors/></font>
  </tr>
<tr>
  <td width="130px" align="left">
    &nbsp;&nbsp;&nbsp;<bean:message key="donations.label.name"/>
  </td>
  <td width="100px" align="left">
    <html:text property="name" size="30" maxlength="30"/>
  </td>
</tr>
<tr>
  <td align="left">
    &nbsp;&nbsp;&nbsp;<bean:message key="donations.label.address"/>
  </td>
  <td align="left">
    <html:text property="address1" size="30" maxlength="30"/>
  </td>
</tr>
<tr>
  <td align="left">
    &nbsp;&nbsp;&nbsp;<bean:message key="donations.label.space"/>
  </td>
  <td align="left">
    <html:text property="address2" size="30" maxlength="30"/>
  </td>
</tr>
<tr>
  <td align="left">
    &nbsp;&nbsp;&nbsp;<bean:message key="donations.label.space"/>
  </td>
  <td align="left">
    <html:text property="address3" size="30" maxlength="30"/>
  </td>
</tr>
<tr>
  <td align="left">
    &nbsp;&nbsp;&nbsp;<bean:message key="donations.label.phone"/>
  </td>
  <td align="left">
    <html:text property="phone" size="30" maxlength="30"/>
  </td>
</tr>

```

```
|  |
| --- |
| &nbsp;&nbsp;&nbsp;&nbsp;&nbsp;<bean:message key="donations.label.email"/> |
| <html:text property="email" size="30" maxlength="30"/> |
| <bean:message key="donations.label.space"/> |
| <font style="font-size: 7pt"><i><bean:message key="donations.label.note"/></i></font> |
| &nbsp;&nbsp;&nbsp;&nbsp;&nbsp;<bean:message key="donations.label.donation"/> |
| <html:text property="donation" size="30" maxlength="30"/> |
| &nbsp;&nbsp;&nbsp;&nbsp;&nbsp;</td> |
| <html:submit><bean:message key="donations.label.proceed"/></html:submit> |
| &nbsp;&nbsp;&nbsp;&nbsp;&nbsp;</td> |
| <br/>&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;<bean:message key="donations.label.required"/><br/></td> |



</fieldset>
</html:form>

<br>
<h4.5><B><bean:message key="donations.label.policy"/></B></h4.5>
<br><br>
<bean:message key="donations.label.privacy"/>
<br><br>
<bean:message key="donations.label.realex"/>

```

```

<br><br>
<B><bean:message key="donations.label.charity"/></B>

<br><br>
<font style="font-size: 8pt">
<bean:message key="donations.label.support"/>
<a
href="mailto:support@abaltaabaschool.galway.edu.ie">support@abaltaabasc
hool.galway.edu.ie</a>
</font>
</BODY>
</HTML>

```

donations_frame2.jsp

```

<%@ taglib uri="/tags/struts-bean" prefix="bean"%>
<%@ taglib uri="/tags/struts-html" prefix="html"%>

<jsp:useBean id="donationsBean" scope="request"
class="ie.edu.galway.abaltaabaschool.beans.Donations" />

<HTML>
<HEAD>
<script language="javascript" type="text/javascript">
function submitForm(){
    document.realexForm.submit();
}
</script>
</HEAD>
<BODY>
<%(if(donationsBean.isRedirect()){out.println("onload=\"submitForm();\"")
});%>>

<form name="realexForm" action=https://epage.payandshop.com/epage.cgi
method=post>

<input type=hidden name="MERCHANT_ID"
value="<%=donationsBean.getMerchant_id()%>">
<input type=hidden name="ORDER_ID"
value="<%=donationsBean.getOrder_id()%>">
<input type=hidden name="CURRENCY"
value="<%=donationsBean.getCurrency()%>">
<input type=hidden name="AMOUNT"
value="<%=donationsBean.getAmount()%>">
<input type=hidden name="TIMESTAMP"
value="<%=donationsBean.getTimeStamp()%>">
<input type=hidden name="SHA1HASH"
value="<%=donationsBean.getShalHash()%>">
<input type=hidden name="AUTO_SETTLE_FLAG" value="1">

```

```
</BODY>
</HTML>
```

```
<jsp:useBean id="realexResponseBean" scope="request"
class="ie.edu.galway.abaltaabaschool.beans.RealexResponseBean" />
```

178

```

<%
} else {
// there was a problem processing the payment. Generally it's not
// recommended that you say what the problem was - just say there's
// been a problem and give them a customer services number or email to
// contact.

%>
<br>
We're sorry - there has been a problem processing your donation.
<br><br>
Please contact
us at <a
href="mailto:support@abaltaabaschool.galway.edu.ie">support@abaltaabasc
hool.galway.edu.ie</a>
and quote your donation number as: <%=realexResponseBean.getOrder_id()
%>
<br>
<br>
Please note: Your credit card has not been charged.
<br><br>
<%
} // end of the result if-else

} // end of the hashes if-else

%>

```

template.jsp

```

<HTML>
<HEAD>
  <TITLE>Abalta</TITLE>
  <style>
    fieldset{

border: 1px solid #781351;
font-size: 10pt;
}

    legend{
color:#000000;
font-weight: bold;
font-size: 10pt;
}

    body {
margin: 0 auto;
padding: 0;
font: 76% Verdana,Tahoma,Arial,sans-serif;
}
  </style>

</HEAD>

```

```
<BODY>
  <br><br>
  <fieldset>
    <legend>Donation</legend>

    <DIV ALIGN="CENTER">
      <!--E-PAGE TABLE HERE-->
    </DIV>
  </fieldset>
</BODY>
</HTML>
```

Appendix B The Controller Component Code

struts-config.xml

```
<?xml version="1.0" encoding="ISO-8859-1" ?>

<!DOCTYPE struts-config PUBLIC
    "-//Apache Software Foundation//DTD Struts Configuration 1.2//EN"
    "http://jakarta.apache.org/struts/dtds/struts-config_1_2.dtd">

<struts-config>

    <form-beans>
        <form-bean name="RegisterForm"
            type="org.apache.struts.action.DynaActionForm">
            <form-property name="name" type="java.lang.String"/>
            <form-property name="address" type="java.lang.String"/>
            <form-property name="phone" type="java.lang.String" />
                <form-property name="email" type="java.lang.String" />
                <form-property name="username" type="java.lang.String" />
                <form-property name="password" type="java.lang.String" />
                <form-property name="cpassword" type="java.lang.String" />
        </form-bean>

        <form-bean name="DonationForm"
            type="org.apache.struts.action.DynaActionForm">
            <form-property name="donation" type="java.lang.String"/>
            <form-property name="name" type="java.lang.String"/>
            <form-property name="address1" type="java.lang.String"/>
            <form-property name="address2" type="java.lang.String"/>
            <form-property name="address3" type="java.lang.String"/>
            <form-property name="phone" type="java.lang.String" />
                <form-property name="email" type="java.lang.String" />
        </form-bean>

        <form-bean name="RealexResponseForm"
            type="org.apache.struts.action.DynaActionForm">
            <form-property name="TIMESTAMP" type="java.lang.String"/>
                <form-property name="MERCHANT_ID" type="java.lang.String"/>
        </form-bean>
    </form-beans>
</struts-config>
```

```

        <form-property name="RESULT" type="java.lang.String"/>
        <form-property name="ORDER_ID" type="java.lang.String"/>
        <form-property name="MESSAGE" type="java.lang.String"/>
        <form-property name="AUTHCODE" type="java.lang.String"/>
        <form-property name="PASREF" type="java.lang.String"/>
        <form-property name="SHA1HASH" type="java.lang.String"/>
    </form-bean>

    <form-bean name="ForumPost"
type="org.apache.struts.action.DynaActionForm">
    <form-property name="subject" type="java.lang.String"/>
    <form-property name="message" type="java.lang.String"/>
    </form-bean>

    <form-bean name="forumSearch"
type="org.apache.struts.action.DynaActionForm">
    <form-property name="keyword" type="java.lang.String"/>
    <form-property name="checkAfter" type="java.lang.String"/>
    <form-property name="afterDay" type="java.lang.String"/>
    <form-property name="afterMon" type="java.lang.String"/>
    <form-property name="afterYear" type="java.lang.String"/>
    <form-property name="checkBefore" type="java.lang.String"/>
    <form-property name="beforeDate" type="java.lang.String"/>
    <form-property name="beforeDay" type="java.lang.String"/>
    <form-property name="beforeMon" type="java.lang.String"/>
    <form-property name="beforeYear" type="java.lang.String"/>
    <form-property name="checkSubOnly" type="java.lang.String"/>
    <form-property name="checkMsgOnly" type="java.lang.String"/>
    <form-property name="checkUserOnly" type="java.lang.String"/>
    </form-bean>

</form-beans>

<!-- ===== Global Exception
Definitions -->

<global-exceptions>
    <!-- sample exception handler
    <exception
        key="expired.password"
        type="app.ExpiredPasswordException"
        path="/changePassword.jsp"/>
    end sample -->
</global-exceptions>

```


<!-- ===== Global Forward
Definitions -->

```
<global-forwards>
  <!-- Default forward to "Welcome" action -->
  <!-- Demonstrates using index.jsp to forward -->
  <forward
    name="welcome"
    path="/Welcome.do"/>

  <forward
    name="home"
    path="/Home.do"/>

    <forward
      name="admin"
      path="/admin/home.jsp"/>

  <forward
    name="failure"
    path="/info/failure.jsp"/>

    <forward
      name="logout"
      path="/logout.do"/>

</global-forwards>
```

<!-- ===== Action Mapping
Definitions -->

```
<action-mappings>
  <!-- Default "Welcome" action -->
  <!-- Forwards to Welcome.jsp -->
  <action
    path="/Welcome"
    forward="/pages/Welcome.jsp"/>

    <action
      path="/pages/Hello"
      forward="/pages/Hello.jsp"/>

    <action
```

```

    path="/Home"
        type="ie.edu.galway.abaltaabaschool.actions.HomeAction">
    <forward name="success" path="/info/home.jsp"/>
    </action>

    <action
    path="/logout"
        type="ie.edu.galway.abaltaabaschool.actions.LogoutAction">
    <forward name="success" path="/Home.do"/>
    </action>

    <action
    path="/info/gen"
        type="ie.edu.galway.abaltaabaschool.actions.GenAction">
    <forward name="success" path="/info/gen.jsp"/>
    </action>

    <action
    path="/info/faq"

    type="ie.edu.galway.abaltaabaschool.actions.FAQQuestionsAction">
    <forward name="success" path="/info/faq.jsp"/>
    </action>

    <action
    path="/donate/donations"
    forward="/donate/donations.jsp"/>

    <action
    path="/donate/donations_frame"
    forward="/donate/donations_frame.jsp"/>

    <action
    path="/fundraising/fundraiser"
    forward="/fundraising/fundraiser.jsp"/>

    <action
    path="/realexResponse"
    type="ie.edu.galway.abaltaabaschool.actions.RealexResponseAction"
        name="RealexResponseForm"
        scope="request"
        validate="true">
        <forward name="success" path="/donate/realexResponse.jsp"/>
    </action>

    <action

```

```

    path="/submitDonation"
        type="ie.edu.galway.abaltaabaschool.actions.DonationAction"
        name="DonationForm"
        scope="request"
        validate="true"
        input="/donate/donations.jsp">
    <set-property property="cancellable" value="true"/>
    <forward name="success" path="/donate/donations_frame2.jsp"/>
    <forward name="failure" path="/donate/donations_frame.jsp"/>
        <forward name="cancel" path="/donate/donations_frame.jsp" />
    </action>

    <action
    path="/forum/forum"
        type="ie.edu.galway.abaltaabaschool.actions.ForumAction">
        <forward name="success" path="/forum/forum.jsp"/>
    </action>

<action
    path="/forum/forumMessages"
    type="ie.edu.galway.abaltaabaschool.actions.ForumMessagesAction">
    <forward name="success" path="/forum/forumMessages.jsp"/>
</action>

    <action
    path="/info/register"
    forward="/info/register.jsp"/>

        <action path="/submitRegister"
            type="ie.edu.galway.abaltaabaschool.actions.RegisterFormAction"
            name="RegisterForm"
            scope="request"
            validate="true"
            input="/info/register.jsp">
            <set-property property="cancellable" value="true"/>
            <forward name="success" path="/info/success.jsp"/>
            <forward name="invalid" path="/info/register.jsp" />
            <forward name="cancel" path="/Home.do" />
        </action>

    <action path="/forum/forumNewPost"
type="ie.edu.galway.abaltaabaschool.actions.ForumReplyAction">
    <forward name="success" path="/forum/forumNewPost.jsp" />
</action>

```

```

        <action path="/submitForumPost"
type="ie.edu.galway.abaltaabaschool.actions.ForumNewPostAction"
        name="ForumPost"
        scope="request"
        validate="true"
        input="/forum/forumNewPost.jsp">
        <set-property property="cancellable" value="true"/>
        <forward name="success" path="/forum/forumMessages.do"/>
        <forward name="failure" path="/forum/forumNewPost.jsp" />
        <forward name="cancel" path="/forum/forum.do" />
        </action>

    <action
path="/forum/forumSearch"
forward="/forum/forumSearch.jsp"/>

        <action path="/submitForumSearch"
type="ie.edu.galway.abaltaabaschool.actions.ForumSearchAction"
        name="forumSearch"
        scope="request"
        validate="true"
        input="/forum/forumSearch.jsp">
        <set-property property="cancellable" value="true"/>
        <forward name="success"
path="/forum/forumSearchResults.jsp"/>
        <forward name="invalid" path="/forum/forumSearch.jsp"/>
        <forward name="cancel" path="/forum/forum.do" />
        </action>

</action-mappings>

<message-resources parameter="MessageResources"/>

</struts-config>

```

Appendix C The Model Component Code

Actions: ie\edu\galway\abaltaabaschool\actions

DonationAction.java

```
package ie.edu.galway.abaltaabaschool.actions;

import java.sql.Connection;
import java.sql.SQLException;
import org.apache.struts.action.Action;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

import org.apache.struts.action.ActionForm;
import org.apache.struts.action.ActionForward;
import org.apache.struts.action.ActionMapping;
import org.apache.struts.action.ActionMessage;
import org.apache.struts.action.ActionMessages;
import org.apache.struts.action.ActionRedirect;
import org.apache.struts.action.DynaActionForm;

import ie.edu.galway.abaltaabaschool.beans.Donations;
import ie.edu.galway.abaltaabaschool.database.DBApp;
import ie.edu.galway.abaltaabaschool.util.Realax;

public class DonationAction extends Action{
    public ActionForward execute(ActionMapping mapping, ActionForm
form, HttpServletRequest request, HttpServletResponse response) throws
Exception
    {

        if (isCancelled(request)){
            return mapping.findForward("cancel");
        }

        DynaActionForm postForm = (DynaActionForm)form;
        DBApp dba = new DBApp();

        double amount = 0;

        //Create object of ActionMesssages
        ActionMessages errors = new ActionMessages();
        //Check and collect errors
        if(((String)postForm.get("name")).equals("")) {
            errors.add("name",new
ActionMessage("error.name.required"));
        }
        if(((String)postForm.get("address1")).equals("")){
```

```

        errors.add("address", new
ActionMessage("error.address.required"));
    }
    if(((String)postForm.get("phone")).equals("")) {
        errors.add("phone", new
ActionMessage("error.phone.required"));
    }
    if(((String)postForm.get("email")).equals("")) {
        errors.add("email", new
ActionMessage("error.emailaddress.required"));
    }
    if(((String)postForm.get("email")).indexOf("@")==-1 ||
((String)postForm.get("email")).indexOf(".") ==-1 ||
((String)postForm.get("email")).lastIndexOf(".") <
((String)postForm.get("email")).indexOf("@") ) {
        errors.add("email", new
ActionMessage("error.emailaddress.invalid"));
    }

    if(((String)postForm.get("donation")).equals("")) {
        errors.add("donation", new
ActionMessage("error.donation.required"));
    }
    else{

        try{
            amount =
Double.parseDouble(postForm.get("donation").toString());
            if ((amount == 0) || (amount < 0)){
                throw new Exception();
            }
        }catch(NumberFormatException e){
            errors.add("donation", new
ActionMessage("error.donation.parse"));
        }
        catch(Exception ex){
            errors.add("donation", new
ActionMessage("error.donation.parse"));
        }
    }

    saveErrors(request, errors);
    if(errors.isEmpty()){
        int amt = (int) (amount * 100);

        String address = (String)postForm.get("address1")
+ (String)postForm.get("address2") + (String)postForm.get("address3");

        String orderId = "-1";
        Connection con = dba.getConnection();
        if(con!=null)
        {
            dba.submitDonation(con,
(String)postForm.get("name"), address, (String)postForm.get("phone"),
(String)postForm.get("email"), amt);
            orderId = String.valueOf(dba.getOrderID(con));
        }
    }

```

```

        else
            return mapping.findForward("failure");

        if("-1".equals(orderId)){
            return mapping.findForward("failure");
        }

        Realax realexUtil = new Realax();

        String timeStamp = realexUtil.getTimeStamp();

        // These four values need to be corrected

        String testString = timeStamp + "." +
Realax.MERCHANT_ID + "." + orderId + "." +
        amt + "." + Realax.CURRENCY;

        String testStringHashed =
realexUtil.calcSHA1(testString);
        testString = testStringHashed + "." +
Realax.SECRET;

        String shalhash =
realexUtil.calcSHA1(testString);

        Donations donations = new Donations();
        donations.setRedirect(true);
        donations.setCurrency(Realax.CURRENCY);

        donations.setTimeStamp(timeStamp);
        donations.setMerchant_id(Realax.MERCHANT_ID);
        donations.setOrder_id(orderId);
        donations.setAmount(amt);
        donations.setShalHash(shalhash);

        request.setAttribute("donationsBean", donations);

        con.close();
        con = null;

        return mapping.findForward("success");
    }
    return mapping.findForward("failure");
}
}
}

```

FAQQuestionsAction.java

```
/*
 * Created on 11-Nov-2006
 *
 * TODO To change the template for this generated file go to
 * Window - Preferences - Java - Code Style - Code Templates
 */
package ie.edu.galway.abaltaabaschool.actions;

/**
 * @author Cathy
 *
 * TODO To change the template for this generated type comment go to
 * Window - Preferences - Java - Code Style - Code Templates
 */
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

import org.apache.struts.action.Action;
import org.apache.struts.action.ActionForm;
import org.apache.struts.action.ActionForward;
import org.apache.struts.action.ActionMapping;

import java.sql.*;
import java.util.ArrayList;
import java.util.List;

import ie.edu.galway.abaltaabaschool.database.DBApp;
import ie.edu.galway.abaltaabaschool.beans.FaqBean;
import ie.edu.galway.abaltaabaschool.businessobjects.Faq;

public class FAQQuestionsAction extends Action {
    public ActionForward execute(ActionMapping mapping, ActionForm
form, HttpServletRequest request, HttpServletResponse response) throws
Exception
    {
        System.out.println("Big issues here          1");

        FaqBean fB = new FaqBean();
        DBApp dba = new DBApp();

        List faqList = new ArrayList();

        Connection con = dba.getConnection();
        ResultSet rs = null;

        if(con!=null)
        {
            rs = dba.executeQuery(con, "select * from faq order
by 'id' asc");
        }
        else
            return mapping.findForward("failure");
    }
}
```



```

        System.out.println("Big issues here                2");

        //get the question
        while(rs.next())
        {
            Faq faq = new Faq();
            faq.setId(rs.getInt("id"));
            faq.setQuestion(rs.getString("question"));
            faq.setAnswer(rs.getString("answer"));
            faqList.add(faq);
        }

        System.out.println("Big issues here                3");
        fB.setFaq(faqList);

        con.close();
        con = null;

        System.out.println("Big issues here                4");
        request.setAttribute("FaqBean", fB);
        return mapping.findForward("success");
    }
}

```

ForumAction.java

```

/*
 * Created on 13-Jan-2007
 *
 * @author Cathy
 */
package ie.edu.galway.abaltaabaschool.actions;

import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

import org.apache.struts.action.Action;
import org.apache.struts.action.ActionForm;
import org.apache.struts.action.ActionForward;
import org.apache.struts.action.ActionMapping;

import java.sql.*;
import java.util.ArrayList;
import java.util.List;

import ie.edu.galway.abaltaabaschool.database.DBApp;
import ie.edu.galway.abaltaabaschool.beans.ForumBean;
import ie.edu.galway.abaltaabaschool.businessobjects.Forum;

```

```

public class ForumAction extends Action {
    public ActionForward execute(ActionMapping mapping, ActionForm
form, HttpServletRequest request, HttpServletResponse response) throws
Exception
    {

        List forumList = new ArrayList();

        ForumBean fB = new ForumBean();
        DBApp dba = new DBApp();

        Connection con = dba.getConnection();
        ResultSet rs = null;

        if(con!=null)
        {
            rs = dba.executeQuery(con, "select * from forum where
active='yes' order by forum_name asc");
        }
        else
            return mapping.findForward("failure");

        //get the question
        while (rs.next())
        {
            Forum forum = new Forum();
            forum.setId(rs.getInt("f_id"));
            forum.setForumName(rs.getString("forum_name"));
            forumList.add(forum);
        }

        fB.setForum(forumList);

        con.close();
        con = null;

        request.setAttribute("ForumBean", fB);
        return mapping.findForward("success");
    }
}

```

ForumMessageAction.java

```

/*
 * Created on 13-Jan-2007
 *
 * @author Cathy

```

```

    */
package ie.edu.galway.abaltaabaschool.actions;

import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;

import org.apache.struts.action.Action;
import org.apache.struts.action.ActionForm;
import org.apache.struts.action.ActionForward;
import org.apache.struts.action.ActionMapping;

import java.sql.*;
import java.util.ArrayList;
import java.util.List;

import ie.edu.galway.abaltaabaschool.database.DBApp;
import ie.edu.galway.abaltaabaschool.beans.ForumMessageBean;
import ie.edu.galway.abaltaabaschool.businessobjects.ForumMessages;

public class ForumMessagesAction extends Action{

    DBApp dba = new DBApp();
    List msgList= new ArrayList();
    ForumMessageBean fMB = new ForumMessageBean();

    public ActionForward execute(ActionMapping mapping, ActionForm
form, HttpServletRequest request, HttpServletResponse response) throws
Exception
    {

        HttpSession session = request.getSession();
        Connection con = dba.getConnection();

        msgList.clear();
        String msg = "";
        String sFid = "";

        if(request.getParameter("forumid")!=null){
            sFid = request.getParameter("forumid");
            session.setAttribute("forumid",sFid);
        }else{
            sFid = (String) session.getAttribute("forumid");
        }

        if(request.getParameter("msgId")!=null){
            msg =
getMessageById(request.getParameter("msgId"),con);
            session.setAttribute("msgId",
request.getParameter("msgId"));
        }

        getMessage(0, con, "##", sFid);

        request.setAttribute("ForumMessageBean", fMB);
        session.setAttribute("ForumName", getForumName(con,
sFid));
    }
}

```

```

        //to delete
        request.setAttribute("ForumName", getForumName(con,
sFid));

        con.close();
        con = null;
        return mapping.findForward("success");
    }

    /*
    Function: getMessage
    Purpose: Recursive Function that selects all the forum messages
    from the database messages table
            It determines if the message is a root-parent
    message, parent message or a child.
            If it is a root parent, it is the first in the
    tree.
            If it is a parent or child, it indents to the
    appropriate place
    Output: A list of objects is created that holds all the
    messages to be displayed
    */

    private String getMessageById(String msgId, Connection con){
        ResultSet rs =null;

        try{
            rs = dba.executeQuery(con, "select * from
messages where msg_id=" + msgId);

            if(rs.next()){
                return rs.getString("message");
            }

        }catch (SQLException se){
            se.printStackTrace();
        }
        return "";
    }

    private ResultSet getMessage(int parentId, Connection con,
String indent, String sForumId)
    {
        ResultSet rs =null;

        try{

            rs = dba.executeQuery(con, "select * from
messages where parent_msg_id="+parentId+ " AND f_id="+ sForumId +"
order by date_time desc");

            while(rs.next()){
                //create the business object
                ForumMessages fM = new ForumMessages();

```



```

    }
    return "";
}
}

```

ForumNewPostAction.java

```

package ie.edu.galway.abaltaabaschool.actions;

import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;

import org.apache.struts.action.Action;
import org.apache.struts.action.ActionForm;
import org.apache.struts.action.ActionForward;
import org.apache.struts.action.ActionMapping;
import org.apache.struts.action.DynaActionForm;
import org.apache.struts.action.ActionMessages;
import org.apache.struts.action.ActionMessage;

import ie.edu.galway.abaltaabaschool.database.DBApp;

import java.sql.Connection;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.util.*;
import java.lang.Integer;

import ie.edu.galway.abaltaabaschool.beans.ForumNewPostBean;

public class ForumNewPostAction extends Action{

    DBApp dba = new DBApp();

    public ActionForward execute(ActionMapping mapping, ActionForm
form, HttpServletRequest request, HttpServletResponse response) throws
Exception
    {
        if (isCancelled(request)){
            return mapping.findForward("cancel");
        }

        DynaActionForm postForm = (DynaActionForm)form;
        Connection con = dba.getConnection();

        HttpSession session = request.getSession();

        String sForumName = "";
        int iForumId = 0;
        String sPID = "";
    }
}

```

```

sForumName = session.getAttribute("ForumName").toString();
iForumId = getForumId(con, sForumName);

if (session.getAttribute("p_msg_id") != null)
    sPID = session.getAttribute("p_msg_id").toString();
else
    sPID = "0";

//Create object of ActionMesssages
ActionMessages errors = new ActionMessages();

//Check and collect errors
if(((String)postForm.get("subject")).equals("")) {
    errors.add("subject", new
ActionMessage("error.subject.required"));
}
//Check and collect errors
if(((String)postForm.get("message")).equals("")) {
    errors.add("message", new
ActionMessage("error.message.required"));
}

saveErrors(request, errors);

//Forward the page
if(errors.isEmpty()){

    if(con!=null)
    {
        dba.addForumMessage(con, iForumId,
Integer.parseInt(sPID), request.getUserPrincipal().toString(),
(String)postForm.get("subject"), (String)postForm.get("message") );
    }
    else{
        con.close();
        return mapping.findForward("failure");
    }

    con.close();
    con = null;

    return mapping.findForward("success");
}
else{
    con.close();
    con = null;
    return mapping.findForward("invalid");
}
}

private int getForumId(Connection con, String sFName){
    ResultSet rs =null;
    int sFID = 0;

    try{

```

```

        rs = dba.executeQuery(con, "select * from forum
where forum_name = '" + sFName + "'");
        while(rs.next()){
            sFID = rs.getInt("f_id");
        }
        return sFID;

    } catch (SQLException se){
        se.printStackTrace();
    }
    return 0;
}
}

```

ForumReplyAction.java

```

package ie.edu.galway.abaltaabaschool.actions;

import java.sql.Connection;

import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

import javax.servlet.http.HttpSession;

import org.apache.struts.action.Action;
import org.apache.struts.action.ActionForm;
import org.apache.struts.action.ActionForward;
import org.apache.struts.action.ActionMapping;

import ie.edu.galway.abaltaabaschool.beans.ForumNewPostBean;
import ie.edu.galway.abaltaabaschool.database.DBApp;

import java.sql.ResultSet;
import java.sql.SQLException;

public class ForumReplyAction extends Action{

    DBApp dba = new DBApp();

    public ActionForward execute(ActionMapping mapping, ActionForm
form, HttpServletRequest request, HttpServletResponse response) throws
Exception
    {
        if (isCancelled(request)){
            return mapping.findForward("cancel");
        }

        Connection con = dba.getConnection();
        HttpSession session = request.getSession();
    }
}

```



```

        String sPID = "";
        //boolean isReply = false;

        if (session.getAttribute("p_msg_id") != null)
            sPID = session.getAttribute("p_msg_id").toString();

        if (request.getParameter("newpost") == null){
            request.setAttribute("ForumNewPostBean",getDetails(con,
sPID));
        }else{
            request.setAttribute("ForumNewPostBean",null);
            session.setAttribute("p_msg_id", null);
        }

        return mapping.findForward("success");
    }
    private ForumNewPostBean getDetails(Connection con, String
sParentID){
        ResultSet rs =null;
        ForumNewPostBean fPB = new ForumNewPostBean();

        try{
            rs = dba.executeQuery(con, "select * from
messages where msg_id = '" + sParentID + "'");
            while(rs.next()){

                fPB.setSUsername(rs.getString("user_name"));
                fPB.setSSubject(rs.getString("subject"));
                fPB.setSMsg(rs.getString("message"));
            }

        }catch (SQLException se){
            se.printStackTrace();
        }
        return fPB;
    }
}

```

ForumSearchAction.java

```

package ie.edu.galway.abaltaabaschool.actions;

import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;

import org.apache.struts.action.Action;
import org.apache.struts.action.ActionForm;
import org.apache.struts.action.ActionForward;

```

```

import org.apache.struts.action.ActionMapping;
import org.apache.struts.action.DynaActionForm;
import org.apache.struts.action.ActionMessages;
import org.apache.struts.action.ActionMessage;

import ie.edu.galway.abaltaabaschool.database.DBApp;
import ie.edu.galway.abaltaabaschool.beans.ForumSearchBean;
import ie.edu.galway.abaltaabaschool.businessobjects.ForumSearch;

import java.sql.Connection;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.util.ArrayList;
import java.util.List;
import java.text.*;
import java.util.*;

public class ForumSearchAction extends Action{

    DBApp dba = new DBApp();
    List msgList= new ArrayList();
    String searchKeyword = "";

    public ActionForward execute(ActionMapping mapping, ActionForm
form, HttpServletRequest request, HttpServletResponse response) throws
Exception
    {
        if (isCancelled(request)){
            return mapping.findForward("cancel");
        }

        DynaActionForm pForm = (DynaActionForm)form;
        Connection con = dba.getConnection();

        HttpSession session = request.getSession();
        ForumSearchBean fSB = new ForumSearchBean();
        msgList.clear();
        int forumID = 0;
        int sqlStatementType = 0;

        ResultSet rs;
        String orSubKeyword = "";
        String orMsgKeyword = "";
        String orUserKeyword = "";
        int multiKeyword = 0;
        String dayBefore, monBefore, yearBefore, dateBefore = "";
        String dayAfter, monAfter, yearAfter, dateAfter = "";
        SimpleDateFormat sT = new SimpleDateFormat("yyyy-MM-dd
hh:mm:ss:S");
        Date dT = null;
        Date dTB = null;
        String[] words = null;
        forumID = getForumID(con,
session.getAttribute("ForumName").toString());

        //Create object of ActionMesssages

```

```

        ActionMessages errors = new ActionMessages();

        if(((String)pForm.get("keyword")).equals("")) {

            errors.add("keyword",new
ActionMessage("error.search.required"));

        }

        saveErrors(request,errors);

        //Forward the page
        if(errors.isEmpty()){

            if(!((String)pForm.get("keyword")).equals("")) {
                words = ((String)pForm.get("keyword")).split ( "
");

                if (words.length > 1)
                {
                    for (int i=0; i < words.length; i++){

                        if (i != words.length -1)
                        {
                            orSubKeyword =
orSubKeyword + "subject LIKE '%" + words[i] + "%' OR ";
                            orMsgKeyword =
orMsgKeyword + "message LIKE '%" + words[i] + "%' OR ";
                            orUserKeyword =
orUserKeyword + "user_name LIKE '%" + words[i] + "%' OR ";
                        }

                        else
                        {
                            orSubKeyword =
orSubKeyword + "subject LIKE '%" + words[i] + "%'";
                            orMsgKeyword =
orMsgKeyword + "message LIKE '%" + words[i] + "%'";
                            orUserKeyword =
orUserKeyword + "user_name LIKE '%" + words[i] + "%'";
                        }
                    }
                    multiKeyword = 1;
                }

                searchKeyword = (String)pForm.get("keyword");
                sqlStatementType = 1;
                //errors.add("subject",new
ActionMessage("error.subject.required"));
            }

            if(!((String)pForm.get("checkAfter")).equals("")) {

                dayAfter = ((String)pForm.get("afterDay"));
                if (dayAfter.length() < 2)
                    dayAfter = "0" + dayAfter;
                monAfter = ((String)pForm.get("afterMon"));
                yearAfter = ((String)pForm.get("afterYear"));
            }
        }
    }

```

```

        dateAfter = yearAfter + "-" + monAfter + "-" +
dayAfter + " 00:00:00:0";

        dT = sT.parse(dateAfter);

        sqlStatementType = 2;
    }
    if(!((String)pForm.get("checkBefore")).equals("")) {
        dayBefore = ((String)pForm.get("beforeDay"));
        if (dayBefore.length() < 2)
            dayBefore = "0" + dayBefore;
        monBefore = ((String)pForm.get("beforeMon"));
        yearBefore = ((String)pForm.get("beforeYear"));

        dateBefore = yearBefore + "-" + monBefore + "-" +
dayBefore + " 00:00:00:0";

        dTB = sT.parse(dateBefore);

        sqlStatementType = 3;
        //errors.add("subject",new
        ActionMessage("error.subject.required"));
    }

    if(!((String)pForm.get("checkBefore")).equals("")) &&
(!((String)pForm.get("checkAfter")).equals("")){

        sqlStatementType = 4;
        //errors.add("subject",new
        ActionMessage("error.subject.required"));
    }
    if(!((String)pForm.get("checkSubOnly")).equals("")) {
        sqlStatementType = 5;
        if(!((String)pForm.get("checkBefore")).equals("")){
            sqlStatementType = 6;

        }
        else
        if(!((String)pForm.get("checkAfter")).equals("")) {

            sqlStatementType = 7;

        }
        if(!((String)pForm.get("checkBefore")).equals(""))
&& (!((String)pForm.get("checkAfter")).equals("")){

            sqlStatementType = 8;
            //errors.add("subject",new
            ActionMessage("error.subject.required"));
        }

        //errors.add("subject",new
        ActionMessage("error.subject.required"));
    }
}

```

```

        if(!((String)pForm.get("checkMsgOnly")).equals("")) {
            sqlStatementType = 9;
            if(!((String)pForm.get("checkBefore")).equals("")){
                sqlStatementType = 10;
            }
        }
        else
        if(!((String)pForm.get("checkAfter")).equals("")) {

            sqlStatementType = 11;

        }
        if(!((String)pForm.get("checkBefore")).equals(""))
        && (!((String)pForm.get("checkAfter")).equals("")){
            // if (((String)pForm.get("checkBefore")) >
            ((String)pForm.get("checkAfter"))){
                // sqlStatementType = 7;
                // }

                sqlStatementType = 12;
                //errors.add("subject",new
                ActionMessage("error.subject.required"));
            }

            //errors.add("subject",new
            ActionMessage("error.subject.required"));
        }
        if(!((String)pForm.get("checkUserOnly")).equals("")) {
            sqlStatementType = 13;
            if(!((String)pForm.get("checkBefore")).equals("")){
                sqlStatementType = 14;
            }
        }
        else
        if(!((String)pForm.get("checkAfter")).equals("")) {

            sqlStatementType = 15;

        }
        if(!((String)pForm.get("checkBefore")).equals(""))
        && (!((String)pForm.get("checkAfter")).equals("")){

            sqlStatementType = 16;

        }
    }

    List mList = new ArrayList();

    System.out.println("    Case =    " + sqlStatementType);

    switch (sqlStatementType){
    case 1:

```

```

        if(con!=null)
        {
            if (multiKeyword == 0)
            {
                System.out.println("select * from
messages where f_id = " + forumID + " AND (subject LIKE '%" +
((String)pForm.get("keyword")) + "%' AND message LIKE '%" +
((String)pForm.get("keyword")) + "%' AND user_name LIKE '%" +
((String)pForm.get("keyword")) + "%')");
                rs = dba.executeQuery(con,
                "select * from messages where f_id = " + forumID + " AND (subject LIKE
                '%" + ((String)pForm.get("keyword")) + "%' AND message LIKE '%" +
                ((String)pForm.get("keyword")) + "%' AND user_name LIKE '%" +
                ((String)pForm.get("keyword")) + "%')");
                setList(rs);

                rs = dba.executeQuery(con,
                "select * from messages where f_id = " + forumID + " AND (subject LIKE
                '%" + ((String)pForm.get("keyword")) + "%' AND message NOT LIKE '%" +
                ((String)pForm.get("keyword")) + "%' AND user_name NOT LIKE '%" +
                ((String)pForm.get("keyword")) + "%')");
                setList(rs);

                rs = dba.executeQuery(con,
                "select * from messages where f_id = " + forumID + " AND (subject NOT
                LIKE '%" + ((String)pForm.get("keyword")) + "%' AND message LIKE '%" +
                ((String)pForm.get("keyword")) + "%' AND user_name NOT LIKE '%" +
                ((String)pForm.get("keyword")) + "%')");
                setList(rs);

                rs = dba.executeQuery(con,
                "select * from messages where f_id = " + forumID + " AND (subject NOT
                LIKE '%" + ((String)pForm.get("keyword")) + "%' AND message NOT LIKE
                '%" + ((String)pForm.get("keyword")) + "%' AND user_name LIKE '%" +
                ((String)pForm.get("keyword")) + "%')");
                mList = setList(rs);
            }
            else if (multiKeyword == 1)
            {
                rs = dba.executeQuery(con,
                "select * from messages where f_id = " + forumID + " AND (" +
                orSubKeyword + " OR " + orMsgKeyword + " OR " + orUserKeyword + ")");
                mList = setList(rs);
            }
            fSB.setForumMsg(mList);
        }
        else{
            con.close();
            return mapping.findForward("failure");
        }
        break;
    case 2:

        SimpleDateFormat sTB = new
SimpleDateFormat("yyyy-MM-dd hh:mm:ss:S");

```

```

        if(con!=null)
        {
            if (multiKeyword == 0){
                System.out.println("select * from
messages where f_id = " + forumID + " AND (subject LIKE '%" +
((String)pForm.get("keyword")) + "%' AND message LIKE '%" +
((String)pForm.get("keyword")) + "%' AND date_time >= '" +
sTB.format(dT) + "')");

                rs = dba.executeQuery(con, "select * from
messages where f_id = " + forumID + " AND (subject LIKE '%" +
((String)pForm.get("keyword")) + "%' AND message LIKE '%" +
((String)pForm.get("keyword")) + "%' AND user_name LIKE '%" +
((String)pForm.get("keyword")) + "%' AND date_time >= '" +
sTB.format(dT) + "')");

                setList(rs);

                rs = dba.executeQuery(con, "select * from
messages where f_id = " + forumID + " AND (subject LIKE '%" +
((String)pForm.get("keyword")) + "%' AND message NOT LIKE '%" +
((String)pForm.get("keyword")) + "%' AND user_name NOT LIKE '%" +
((String)pForm.get("keyword")) + "%' AND date_time >= '" +
sTB.format(dT) + "')");

                setList(rs);

                rs = dba.executeQuery(con, "select * from
messages where f_id = " + forumID + " AND (subject NOT LIKE '%" +
((String)pForm.get("keyword")) + "%' AND message LIKE '%" +
((String)pForm.get("keyword")) + "%' AND user_name NOT LIKE '%" +
((String)pForm.get("keyword")) + "%' AND date_time >= '" +
sTB.format(dT) + "')");

                setList(rs);

                rs = dba.executeQuery(con, "select * from
messages where f_id = " + forumID + " AND (subject NOT LIKE '%" +
((String)pForm.get("keyword")) + "%' AND message NOT LIKE '%" +
((String)pForm.get("keyword")) + "%' AND user_name LIKE '%" +
((String)pForm.get("keyword")) + "%' AND date_time >= '" +
sTB.format(dT) + "')");

                mList = setList(rs);
            }
            else if (multiKeyword == 1){
                rs = dba.executeQuery(con, "select * from
messages where f_id = " + forumID + " AND (" + orSubKeyword + " OR " +
orMsgKeyword + " OR " + orUserKeyword + ") AND date_time >= '" +
sTB.format(dT) + "')");

                mList = setList(rs);
            }

            fSB.setForumMsg(mList);
        }
        else{
            con.close();
            return mapping.findForward("failure");
        }
    }

```

```

    }
    break;
case 3:

    SimpleDateFormat sT2= new SimpleDateFormat("YYYY-
MM-dd hh:mm:ss:S");

    if(con!=null)
    {
        if (multiKeyword == 0){
            rs = dba.executeQuery(con,
            "select * from messages where f_id = " + forumID + " AND (subject LIKE
            '%" + ((String)pForm.get("keyword")) + "%' AND message LIKE '%" +
            ((String)pForm.get("keyword")) + "%' AND user_name LIKE '%" +
            ((String)pForm.get("keyword")) + "%' AND date_time <= '" +
            sT2.format(dTB) + "')");

            setList(rs);

            rs = dba.executeQuery(con,
            "select * from messages where f_id = " + forumID + " AND (subject LIKE
            '%" + ((String)pForm.get("keyword")) + "%' AND message NOT LIKE '%" +
            ((String)pForm.get("keyword")) + "%' AND user_name NOT LIKE '%" +
            ((String)pForm.get("keyword")) + "%' AND date_time <= '" +
            sT2.format(dTB) + "')");

            setList(rs);

            rs = dba.executeQuery(con,
            "select * from messages where f_id = " + forumID + " AND (subject NOT
            LIKE '%" + ((String)pForm.get("keyword")) + "%' AND message LIKE '%" +
            ((String)pForm.get("keyword")) + "%' AND user_name NOT LIKE '%" +
            ((String)pForm.get("keyword")) + "%' AND date_time <= '" +
            sT2.format(dTB) + "')");

            setList(rs);

            rs = dba.executeQuery(con,
            "select * from messages where f_id = " + forumID + " AND (subject NOT
            LIKE '%" + ((String)pForm.get("keyword")) + "%' AND message NOT LIKE
            '%" + ((String)pForm.get("keyword")) + "%' AND user_name LIKE '%" +
            ((String)pForm.get("keyword")) + "%' AND date_time <= '" +
            sT2.format(dTB) + "')");

            mList = setList(rs);

        }
        else if (multiKeyword == 1){
            rs = dba.executeQuery(con,
            "select * from messages where f_id = " + forumID + " AND (" +
            orSubKeyword + " OR " + orMsgKeyword + " OR " + orUserKeyword + ") AND
            date_time <= '" + sT2.format(dTB) + "')");

            mList = setList(rs);

        }

        fSB.setForumMsg(mList);
    }
    else{
        con.close();
        return mapping.findForward("failure");
    }
}

```



```

        break;
    case 4:
        SimpleDateFormat sTBT= new
SimpleDateFormat("yyyy-MM-dd hh:mm:ss:S");

        if(con!=null)
        {
            if (multiKeyword == 0){
                rs = dba.executeQuery(con,
                "select * from messages where f_id = " + forumID + " AND (subject LIKE
                '%" + ((String)pForm.get("keyword")) + "%' AND message LIKE '%" +
                ((String)pForm.get("keyword")) + "%' AND user_name LIKE '%" +
                ((String)pForm.get("keyword")) + "%' AND date_time >= '" +
                sTBT.format(dT) + "' AND date_time <= '" + sTBT.format(dTB) + "')");
                mList = setList(rs);

                rs = dba.executeQuery(con,
                "select * from messages where f_id = " + forumID + " AND (subject LIKE
                '%" + ((String)pForm.get("keyword")) + "%' AND message NOT LIKE '%" +
                ((String)pForm.get("keyword")) + "%' AND user_name NOT LIKE '%" +
                ((String)pForm.get("keyword")) + "%' AND date_time >= '" +
                sTBT.format(dT) + "' AND date_time <= '" + sTBT.format(dTB) + "')");
                setList(rs);

                rs = dba.executeQuery(con,
                "select * from messages where f_id = " + forumID + " AND (subject NOT
                LIKE '%" + ((String)pForm.get("keyword")) + "%' AND message LIKE '%" +
                ((String)pForm.get("keyword")) + "%' AND user_name NOT LIKE '%" +
                ((String)pForm.get("keyword")) + "%' AND date_time >= '" +
                sTBT.format(dT) + "' AND date_time <= '" + sTBT.format(dTB) + "')");
                setList(rs);

                rs = dba.executeQuery(con,
                "select * from messages where f_id = " + forumID + " AND (subject NOT
                LIKE '%" + ((String)pForm.get("keyword")) + "%' AND message NOT LIKE
                '%" + ((String)pForm.get("keyword")) + "%' AND user_name LIKE '%" +
                ((String)pForm.get("keyword")) + "%' AND date_time >= '" +
                sTBT.format(dT) + "' AND date_time <= '" + sTBT.format(dTB) + "')");
                mList = setList(rs);
            }
            else if (multiKeyword == 1){
                rs = dba.executeQuery(con,
                "select * from messages where f_id = " + forumID + " AND (" +
                orSubKeyword + " OR " + orMsgKeyword + " OR " + orUserKeyword + ") AND
                date_time >= '" + sTBT.format(dT) + "' AND date_time <= '" +
                sTBT.format(dTB) + "'");
                mList = setList(rs);
            }

            fSB.setForumMsg(mList);
        }
        else{
            con.close();
            return mapping.findForward("failure");
        }
        break;

```

```

        case 5:
            if(con!=null)
            {
                if(multiKeyword == 0)
                {
                    rs = dba.executeQuery(con,
"select * from messages where f_id = " + forumID + " AND subject LIKE
'" + ((String)pForm.get("keyword")) + "%'");
                    mList = setList(rs);
                }
                else if (multiKeyword == 1)
                {
                    rs = dba.executeQuery(con,
"select * from messages where f_id = " + forumID + " AND " +
orSubKeyword);
                    mList = setList(rs);
                }
                fSB.setForumMsg(mList);
            }
            else{
                con.close();
                return mapping.findForward("failure");
            }
            break;
        case 6:
            if(con!=null)
            {
                SimpleDateFormat sT6= new
SimpleDateFormat("yyyy-MM-dd hh:mm:ss:S");
                if (multiKeyword == 0)
                {
                    rs = dba.executeQuery(con, "select * from
messages where f_id = " + forumID + " AND (subject LIKE '" +
((String)pForm.get("keyword")) + "%' AND date_time <= '" +
sT6.format(dTB) + "')");
                    mList = setList(rs);
                }
                else if (multiKeyword == 1)
                {
                    rs = dba.executeQuery(con,
"select * from messages where f_id = " + forumID + " AND (" +
orSubKeyword + " AND date_time <= '" + sT6.format(dTB) + "')");
                    mList = setList(rs);
                }
                fSB.setForumMsg(mList);
            }
            else{
                con.close();
                return mapping.findForward("failure");
            }
            break;
        case 7:
            if(con!=null)
            {

```

```

SimpleDateFormat sT7= new
SimpleDateFormat("yyyy-MM-dd hh:mm:ss:S");
    if (multiKeyword == 0)
    {
        rs = dba.executeQuery(con,
"select * from messages where f_id = " + forumID + " AND (subject LIKE
'%" + ((String)pForm.get("keyword")) + "%' AND date_time >= '" +
sT7.format(dT) + "')");
        mList = setList(rs);
    }
    else if (multiKeyword == 1)
    {
        System.out.println("
***** in <= ");
        rs = dba.executeQuery(con,
"select * from messages where f_id = " + forumID + " AND (" +
orSubKeyword + " AND date_time >= '" + sT7.format(dT) + "')");
        mList = setList(rs);
    }

    fSB.setForumMsg(mList);
}
else{
con.close();
return mapping.findForward("failure");
}

break;
case 8:
    if(con!=null)
    {
        SimpleDateFormat sT8= new
SimpleDateFormat("yyyy-MM-dd hh:mm:ss:S");
        if (multiKeyword == 0)
        {
            rs = dba.executeQuery(con,
"select * from messages where f_id = " + forumID + " AND (subject LIKE
'%" + ((String)pForm.get("keyword")) + "%' AND date_time >= '" +
sT8.format(dT) + "' AND date_time <= '" + sT8.format(dTB) + "')");
            mList = setList(rs);
        }
        else if (multiKeyword == 1)
        {
            rs = dba.executeQuery(con,
"select * from messages where f_id = " + forumID + " AND (" +
orSubKeyword + " AND date_time >= '" + sT8.format(dT) + "' AND
date_time <= '" + sT8.format(dTB) + "')");
            mList = setList(rs);
        }
        fSB.setForumMsg(mList);
    }
    else{
con.close();
return mapping.findForward("failure");
}
}

```

```

        break;
    case 9:
        if(con!=null)
        {
            if (multiKeyword == 0)
            {
                rs = dba.executeQuery(con,
                "select * from messages where f_id = " + forumID + " AND message LIKE
                '%" + ((String)pForm.get("keyword")) + "%'");
                mList = setList(rs);
            }
            else if (multiKeyword == 1)
            {
                rs = dba.executeQuery(con,
                "select * from messages where f_id = " + forumID + " AND " +
                orMsgKeyword );
                mList = setList(rs);
            }
            fSB.setForumMsg(mList);
        }
        else{
            con.close();
            return mapping.findForward("failure");
        }
        break;
    case 10:
        if(con!=null)
        {
            SimpleDateFormat sT6= new
            SimpleDateFormat("yyyy-MM-dd hh:mm:ss:S");
            if (multiKeyword == 0)
            {
                rs = dba.executeQuery(con,
                "select * from messages where f_id = " + forumID + " AND (message LIKE
                '%" + ((String)pForm.get("keyword")) + "%' AND date_time <= '" +
                sT6.format(dTB) + "')");
                mList = setList(rs);
            }
            else if (multiKeyword == 1)
            {
                rs = dba.executeQuery(con,
                "select * from messages where f_id = " + forumID + " AND (" +
                orMsgKeyword + " AND date_time <= '" + sT6.format(dTB) + "')");
                mList = setList(rs);
            }
            fSB.setForumMsg(mList);
        }
        else{
            con.close();
            return mapping.findForward("failure");
        }
        break;
    case 11:
        if(con!=null)
        {

```

```

SimpleDateFormat sT7= new
SimpleDateFormat("yyyy-MM-dd hh:mm:ss:S");
    if (multiKeyword == 0)
    {
        rs = dba.executeQuery(con,
"select * from messages where f_id = " + forumID + " AND (message LIKE
'%" + ((String)pForm.get("keyword")) + "%' AND date_time >= '" +
sT7.format(dT) + "')");

        mList = setList(rs);
    }
    else if (multiKeyword == 1)
    {
        rs = dba.executeQuery(con,
"select * from messages where f_id = " + forumID + " AND (" +
orMsgKeyword + " AND date_time >= '" + sT7.format(dT) + "')");
        mList = setList(rs);
    }
    fSB.setForumMsg(mList);
}
else{
con.close();
return mapping.findForward("failure");
}

break;
case 12:
    if(con!=null)
    {
        SimpleDateFormat sT8= new
SimpleDateFormat("yyyy-MM-dd hh:mm:ss:S");
        if (multiKeyword == 0)
        {
            rs = dba.executeQuery(con,
"select * from messages where f_id = " + forumID + " AND (message LIKE
'%" + ((String)pForm.get("keyword")) + "%' AND date_time >= '" +
sT8.format(dT) + "' AND date_time <= '" + sT8.format(dTB) + "')");
            mList = setList(rs);
        }
        else if (multiKeyword == 1)
        {
            rs = dba.executeQuery(con,
"select * from messages where f_id = " + forumID + " AND (" +
orMsgKeyword + " AND date_time >= '" + sT8.format(dT) + "' AND
date_time <= '" + sT8.format(dTB) + "')");
            mList = setList(rs);
        }
        fSB.setForumMsg(mList);
    }
    else{
con.close();
return mapping.findForward("failure");
}

break;
case 13:
    if(con!=null)
    {

```

```

        if (multiKeyword == 0)
        {
            rs = dba.executeQuery(con,
"select * from messages where f_id = " + forumID + " AND user_name LIKE
'" + ((String)pForm.get("keyword")) + "%'");
            mList = setList(rs);
        }
        else if (multiKeyword == 1)
        {
            rs = dba.executeQuery(con,
"select * from messages where f_id = " + forumID + " AND "+
orUserKeyword );
            mList = setList(rs);
        }
        fSB.setForumMsg(mList);
    }
    else{
        con.close();
        return mapping.findForward("failure");
    }

    break;
case 14:
    if(con!=null)
    {
        SimpleDateFormat sT6= new
SimpleDateFormat("yyyy-MM-dd hh:mm:ss:S");
        if (multiKeyword == 0)
        {
            rs = dba.executeQuery(con,
"select * from messages where f_id = " + forumID + " AND (user_name
LIKE '" + ((String)pForm.get("keyword")) + "%' AND date_time <= '" +
sT6.format(dTB) + "')");
            mList = setList(rs);
        }
        else if (multiKeyword == 1)
        {
            rs = dba.executeQuery(con,
"select * from messages where f_id = " + forumID + " AND (" +
orUserKeyword + " AND date_time <= '" + sT6.format(dTB) + "')");
            mList = setList(rs);
        }
        fSB.setForumMsg(mList);
    }
    else{
        con.close();
        return mapping.findForward("failure");
    }

    break;
case 15:
    if(con!=null)
    {
        SimpleDateFormat sT7= new
SimpleDateFormat("yyyy-MM-dd hh:mm:ss:S");
        if (multiKeyword == 0)
        {

```

```

        rs = dba.executeQuery(con,
"select * from messages where f_id = " + forumID + " AND (user_name
LIKE '%" + ((String)pForm.get("keyword")) + "%' AND date_time >= '" +
sT7.format(dT) + "')");

        mList = setList(rs);
    }
    else if (multiKeyword == 1)
    {
        rs = dba.executeQuery(con,
"select * from messages where f_id = " + forumID + " AND (" +
orUserKeyword + " AND date_time >= '" + sT7.format(dT) + "')");
        mList = setList(rs);
    }
    fSB.setForumMsg(mList);
}
else{
    con.close();
    return mapping.findForward("failure");
}

break;
case 16:
    if(con!=null)
    {
        SimpleDateFormat sT8= new
SimpleDateFormat("yyyy-MM-dd hh:mm:ss:S");
        if (multiKeyword == 0)
        {
            rs = dba.executeQuery(con,
"select * from messages where f_id = " + forumID + " AND (user_name
LIKE '%" + ((String)pForm.get("keyword")) + "%' AND date_time >= '" +
sT8.format(dT) + "' AND date_time <= '" + sT8.format(dTB) + "')");
            mList = setList(rs);
        }
        else if (multiKeyword == 1)
        {
            rs = dba.executeQuery(con,
"select * from messages where f_id = " + forumID + " AND (" +
orUserKeyword + " AND date_time >= '" + sT8.format(dT) + "' AND
date_time <= '" + sT8.format(dTB) + "')");
            mList = setList(rs);
        }
        fSB.setForumMsg(mList);
    }
    else{
        con.close();
        return mapping.findForward("failure");
    }

    break;
default:
    break;
}

con.close();

request.setAttribute("ForumSearchBean", fSB);

```

```

return mapping.findForward("success");
}
else{
    con.close();
    return mapping.findForward("invalid");
}
}

private int getForumID(Connection con, String fName){

    int fID = 0;
    ResultSet rs;
    try{
        rs = dba.executeQuery(con, "select f_id from
forum where forum_name='" + fName + "'");
        while(rs.next()){
            fID = rs.getInt("f_id");
        }
        return fID;

    }catch (SQLException se){
        se.printStackTrace();
    }
    return 0;
}

private List setList(ResultSet rs) throws SQLException{
    String msgText = "";
    while(rs.next()){
        ForumSearch fS = new ForumSearch();
        fS.setMsg_id(rs.getInt("msg_id"));
        fS.setF_id(rs.getInt("f_id"));
        fS.setParent_msg_id(rs.getInt("parent_msg_id"));
        fS.setUser_name(rs.getString("user_name"));
        fS.setSubject(rs.getString("subject"));
        fS.setMessage(rs.getString("message"));
        fS.setDate_time(rs.getString("date_time"));
        msgText = getKeyword(rs.getString("message"));
        fS.setMsgText(msgText);
        System.out.println("Search **** " +
rs.getString("subject"));
        msgList.add(fS);
    }
    return msgList;
}

private String getKeyword(String searchText){
    String searchMsg = "";
    int kPos = 0;
    String temp = "";
    int oPos = 0;
    if ((searchText.length() > 200) && (searchText !=
null)){

        kPos = searchText.indexOf(searchKeyword);
        if (kPos == 0)

```



```

        {
            searchMsg = searchText.substring(0, 200);
        }else
        {
            oPos = kPos - 100;
            System.out.println("oPos is      "
+ oPos);

            if (oPos < 0){
                oPos = 0;

                System.out.println("Setting opos is      " + oPos);
            }
            System.out.println("oPos is now
" + oPos);

            System.out.println("kPos is now
" + kPos);

            searchText.substring(oPos, kPos );
            searchMsg =
temp = searchText.substring(kPos
, kPos + 100);

            searchMsg = searchMsg + temp;
        }
    }else
    {
        searchMsg = searchText;
    }
    System.out.println("searchMsg      " + searchMsg);

    return searchMsg;
}

}

```

GenAction.java

```

/*
 * Created on 24-Sep-2006
 *
 * TODO To change the template for this generated file go to
 * Window - Preferences - Java - Code Style - Code Templates
 */
package ie.edu.galway.abaltaabaschool.actions;

/**
 * @author Cathy
 *
 * TODO To change the template for this generated type comment go to
 * Window - Preferences - Java - Code Style - Code Templates
 */
import javax.servlet.http.HttpServletRequest;

```

```

import javax.servlet.http.HttpServletResponse;

import org.apache.struts.action.Action;
import org.apache.struts.action.ActionForm;
import org.apache.struts.action.ActionForward;
import org.apache.struts.action.ActionMapping;

import java.sql.*;

import ie.edu.galway.abaltaabaschool.database.DBApp;
import ie.edu.galway.abaltaabaschool.beans.HomeBean;

public class GenAction extends Action
{
    public ActionForward execute(ActionMapping mapping, ActionForm
form, HttpServletRequest request, HttpServletResponse response) throws
Exception
    {

        HomeBean hB = new HomeBean();
        DBApp dba = new DBApp();

        String page_name = "";

        if(request.getParameter("page_name")!=null)
            page_name = request.getParameter("page_name");
        else
            return mapping.findForward("failure");

        if(page_name.indexOf("'") > 0 )
            return mapping.findForward("failure");

        Connection con = dba.getConnection();
        ResultSet rs = null;

        if(con!=null)
        {
            rs = dba.executeQuery(con, "select * from contents
where page_name='"+page_name+"' order by location");
        }
        else
            return mapping.findForward("failure");

        String tempData = "";
        //get the title
        if(rs.next() && rs.getString("location").equals("a"))
        {
            hB.setTitle(rs.getString("data"));
        }
        else
            return mapping.findForward("failure");

        //get the data
        while (rs.next())
        {
            tempData += rs.getString("data");
        }
    }
}

```

```

    }
    hB.setData(tempData);

    con.close();
    con = null;

    request.setAttribute("HomeBean", hB);
    return mapping.findForward("success");
}
}

```

HomeAction.java

```

/*
 * Created on 24-Sep-2006
 *
 * TODO To change the template for this generated file go to
 * Window - Preferences - Java - Code Style - Code Templates
 */
package ie.edu.galway.abaltaabaschool.actions;

/**
 * @author Cathy
 *
 * TODO To change the template for this generated type comment go to
 * Window - Preferences - Java - Code Style - Code Templates
 */

import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

import org.apache.struts.action.Action;
import org.apache.struts.action.ActionForm;
import org.apache.struts.action.ActionForward;
import org.apache.struts.action.ActionMapping;

import java.sql.*;

import ie.edu.galway.abaltaabaschool.database.DBApp;
import ie.edu.galway.abaltaabaschool.beans.HomeBean;

public class HomeAction extends Action
{
    public ActionForward execute(ActionMapping mapping, ActionForm form,
    HttpServletRequest request, HttpServletResponse response) throws
    Exception
    {

        HomeBean hB = new HomeBean();
    }
}

```

```

DBApp dba = new DBApp();

Connection con = dba.getConnection();
ResultSet rs = null;

if(con!=null)
{
    rs = dba.executeQuery(con, "select * from contents where
page_name='index.jsp' order by location");
}
else
    return mapping.findForward("failure");

String tempData = "";
//get the title
if(rs.next() && rs.getString("location").equals("a"))
{
    hB.setTitle(rs.getString("data"));
}
//get the data
while (rs.next())
{
    tempData += rs.getString("data");
}
hB.setData(tempData);

con.close();
con = null;

request.setAttribute("HomeBean", hB);
return mapping.findForward("success");

}
}

```

Logout.java

```

package ie.edu.galway.abaltaabaschool.actions;

import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;

import org.apache.struts.action.Action;
import org.apache.struts.action.ActionForm;
import org.apache.struts.action.ActionForward;
import org.apache.struts.action.ActionMapping;

public class LogoutAction extends Action {

```

```

        public ActionForward execute(ActionMapping mapping, ActionForm
form, HttpServletRequest request, HttpServletResponse response) throws
Exception
    {
        HttpSession session = request.getSession();
        session.invalidate();

        return mapping.findForward("success");
    }
}

```

ReleaxResponseAction.java

```

package ie.edu.galway.abaltaabaschool.actions;

import java.sql.Connection;
import java.sql.SQLException;

import ie.edu.galway.abaltaabaschool.util.Realax;

import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

import org.apache.struts.action.Action;
import org.apache.struts.action.ActionForm;
import org.apache.struts.action.ActionForward;
import org.apache.struts.action.ActionMapping;
import org.apache.struts.action.DynaActionForm;
import ie.edu.galway.abaltaabaschool.beans.RealexResponseBean;
import ie.edu.galway.abaltaabaschool.database.DBApp;
import ie.edu.galway.abaltaabaschool.util.MailUtil;
import java.sql.ResultSet;

public class RealexResponseAction extends Action{
    public ActionForward execute(ActionMapping mapping, ActionForm
form, HttpServletRequest request, HttpServletResponse response) throws
Exception
    {
        Realax realexUtil = new Realax();
        RealexResponseBean rsb = new RealexResponseBean();
        DynaActionForm postForm = (DynaActionForm)form;

        String timestamp = postForm.get("TIMESTAMP").toString();
        String merchantid =
postForm.get("MERCHANT_ID").toString();
        String result = postForm.get("RESULT").toString();
        String order_id = postForm.get("ORDER_ID").toString();
        String message = postForm.get("MESSAGE").toString();
        String authcode = postForm.get("AUTHCODE").toString();
        String pasref = postForm.get("PASREF").toString();
        String shalhash = postForm.get("SHALHASH").toString();
    }
}

```

```

        String testString = timestamp + "." + merchantid + "." +
order_id + "." +
        result + "." + message + "." + pasref + "." +
authcode;
        String testStringHashed =
realexUtil.calcSHA1(testString);
        testString = testStringHashed + "." + Realax.SECRET;
        testStringHashed = realexUtil.calcSHA1(testString);

        rsb.setOrder_id(order_id);
//      Check to see if hashes match

        if (shalhash.compareTo(testStringHashed) != 0) {
            rsb.setHashesMatch(false);
        }
        else {
//      the hashes were ok - the response came from Realax
Payments

            rsb.setHashesMatch(true);

            if (result.compareTo("00") == 0) {

                rsb.setPaymentOK(true);
                sendResponseMail(order_id);
            }
            else{
                rsb.setPaymentOK(false);
            }
        }
        Connection con = null;
        try{
            DBApp dba = new DBApp();
            con = dba.getConnection();
            dba.updateDonation(con, shalhash, result,
order_id);

            con.close();
        }
        catch(Exception e){
            e.printStackTrace();
            rsb.setPaymentOK(false);
        }
        finally{
            if(con!=null){
                con.close();
            }
        }

//TODO: remove this line
        //sendResponseMail(order_id);
        request.setAttribute("realexResponseBean", rsb);

        return mapping.findForward("success");
    }

    public void sendResponseMail(String orderId) throws
SQLException{
        Connection con = null;

```

```

        try{
            DBApp dba = new DBApp();
            con = dba.getConnection();

            ResultSet rs = dba.executeQuery(con, "select *
from donations where order_id = '"+orderId+"'");

            if(rs.next()){
                double amt = (rs.getInt("amount") /
100);

                String content = "Dear " +
rs.getString("name") + "\n\nThank you very much for supporting Abalta.
Your donation of EUR " + amt + " " + getEmail();

                MailUtil mU = new
MailUtil(rs.getString("email"), "Abalta Donation - Thank you.", content,
"support@abaltaabaschool.galway.edu.ie");

                mU.smtpSend(MailUtil.MAILHOST);
            }
            con.close();

        }
        catch(Exception e){
            e.printStackTrace();
        }
        finally{
            if(con!=null){
                con.close();
            }
        }
    }

    public String getEmail() throws SQLException{
        DBApp dba = new DBApp();
        Connection c = null;
        c = dba.getConnection();

        ResultSet rs = dba.executeQuery(c, "select email from email
where id = '1'");
        while (rs.next()){
            String e = rs.getString("email");

            c.close();
            c = null;
            return e;
        }
        c.close();
        c = null;

        return " ";
    }
}

```

```
}
```

RegisterFormAction.java

```
/*
 * Created on 07-Oct-2006
 *
 * TODO To change the template for this generated file go to
 * Window - Preferences - Java - Code Style - Code Templates
 */
package ie.edu.galway.abaltaabaschool.actions;
/**
 * @author Cathy
 *
 * TODO To change the template for this generated type comment go to
 * Window - Preferences - Java - Code Style - Code Templates
 */

import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

import org.apache.struts.action.Action;
import org.apache.struts.action.ActionForm;
import org.apache.struts.action.ActionForward;
import org.apache.struts.action.ActionMapping;
import org.apache.struts.action.DynaActionForm;
import org.apache.struts.action.ActionMessages;
import org.apache.struts.action.ActionMessage;

import ie.edu.galway.abaltaabaschool.database.DBApp;

import java.sql.Connection;
import java.sql.SQLException;

public class RegisterFormAction extends Action
{
    public ActionForward execute(ActionMapping mapping, ActionForm form,
    HttpServletRequest request, HttpServletResponse response) throws
    Exception
    {
        if (isCancelled(request)){
            return mapping.findForward("cancel");
        }

        DynaActionForm addressForm = (DynaActionForm)form;
        DBApp dba = new DBApp();

        //Create object of ActionMesssages
        ActionMessages errors = new ActionMessages();
        //Check and collect errors
    }
}
```



```

        if(((String)addressForm.get("name")).equals("")) {
            errors.add("name",new ActionMessage("error.name.required"));
        }

        if(((String)addressForm.get("address")).equals("")) {
            errors.add("address",new
ActionMessage("error.address.required"));
        }

        if(((String)addressForm.get("phone")).equals("")) {
            errors.add("phone",new
ActionMessage("error.phone.required"));
        }
        if(((String)addressForm.get("email")).equals("")) {
            errors.add("email",new
ActionMessage("error.emailaddress.required"));
        }
        if(((String)addressForm.get("username")).equals("")) {
            errors.add("username",new
ActionMessage("error.username.required"));
        }
        if(((String)addressForm.get("password")).equals("")) {
            errors.add("password",new
ActionMessage("error.password.required"));
        }
        if(((String)addressForm.get("cpassword")).equals("")) {
            errors.add("cpassword",new
ActionMessage("error.cpassword.required"));
        }

        if(!(((String)addressForm.get("cpassword")).equals((String)addressForm.g
et("password")))) {
            errors.add("cpassword",new
ActionMessage("error.confirm.password.required"));
        }
        if (((String)addressForm.get("username")).indexOf("'') > 0) {
            errors.add("username",new
ActionMessage("error.invalidusername"));
        }
        else
        {
            try{

                if(dba.isDuplicateUsername(((String)addressForm.get("username")))){
                    errors.add("username",new
ActionMessage("error.userduplicate"));
                }
            }catch (SQLException se)
            {
                errors.add("username",new
ActionMessage("error.invalidusername"));
            }
        }
        //Saves the error
        saveErrors(request,errors);

```

```

//Forward the page
if(errors.isEmpty()){

    Connection con = dba.getConnection();
    if(con!=null)
    {
        //System.out.println("**** User***** : " +
        (String)addressForm.get("name") + " " +
        (String)addressForm.get("address") + " " +
        (String)addressForm.get("phone") + " " +
        (String)addressForm.get("email") + " " +
        (String)addressForm.get("username") + " " +
        (String)addressForm.get("password"));
        dba.addUser(con, (String)addressForm.get("name"),
        (String)addressForm.get("address"), (String)addressForm.get("phone"),
        (String)addressForm.get("email"), (String)addressForm.get("username"),
        (String)addressForm.get("password"), "Forum");
    }
    else
        return mapping.findForward("failure");

    con.close();
    con = null;

    return mapping.findForward("success");
}
else{
    return mapping.findForward("invalid");
}
}
}

```

Beans: ie\edu\galway\abaltaabaschool\beans

Donations.java

```

package ie.edu.galway.abaltaabaschool.beans;

public class Donations {
    private boolean redirect;
    private String merchant_id;
    private String order_id;
    private String currency = "EUR";
    private int amount;
    private String timeStamp;
    private String sha1Hash;

```

```

/**
 * @return the redirect
 */
public boolean isRedirect() {
    return redirect;
}

/**
 * @param redirect the redirect to set
 */
public void setRedirect(boolean redirect) {
    this.redirect = redirect;
}

/**
 * @return the amount
 */
public int getAmount(){
    return amount;
}

/**
 * @param amount the amount to set
 */
public void setAmount(int amount) {
    this.amount = amount;
}

/**
 * @return the currency
 */
public String getCurrency() {
    return currency;
}

/**
 * @param currency the currency to set
 */
public void setCurrency(String currency) {
    this.currency = currency;
}

/**
 * @return the merchant_id
 */
public String getMerchant_id() {
    return merchant_id;
}

/**
 * @param merchant_id the merchant_id to set
 */
public void setMerchant_id(String merchant_id) {
    this.merchant_id = merchant_id;
}

```

```

    /**
     * @return the order_id
     */
    public String getOrder_id() {
        return order_id;
    }

    /**
     * @param order_id the order_id to set
     */
    public void setOrder_id(String order_id) {
        this.order_id = order_id;
    }

    /**
     * @return the sha1Hash
     */
    public String getSha1Hash() {
        return sha1Hash;
    }

    /**
     * @param sha1Hash the sha1Hash to set
     */
    public void setSha1Hash(String sha1Hash) {
        this.sha1Hash = sha1Hash;
    }

    /**
     * @return the timeStamp
     */
    public String getTimeStamp() {
        return timeStamp;
    }

    /**
     * @param timeStamp the timeStamp to set
     */
    public void setTimeStamp(String timeStamp) {
        this.timeStamp = timeStamp;
    }
}

```

FAQBean.java

```

package ie.edu.galway.abaltaabaschool.beans;

import java.util.List;

/**

```

```

* @author Cathy
*
* TODO To change the template for this generated type comment go to
* Window - Preferences - Java - Code Style - Code Templates
*/
public class FaqBean {

    private List faq;

    public List getFaq() {
        return faq;
    }
    public void setFaq(List faq) {
        this.faq = faq;
    }
}

```

ForumBean.java

```

/*
* Created on 13-Jan-2007
* @author Cathy
*/

package ie.edu.galway.abaltaabaschool.beans;

import java.util.List;

public class ForumBean {
    private List forum;

    public List getForum() {
        return forum;
    }
    public void setForum(List forum) {
        this.forum = forum;
    }
}

```

ForumMessageBean.java

```

/*
* Created on 13-Jan-2007
* @author Cathy
*/

```

```

package ie.edu.galway.abaltaabaschool.beans;

import java.util.List;

public class ForumMessageBean {

    private List forumMsg;

    /**
     * @return the forumMsg
     */
    public List getForumMsg() {
        return forumMsg;
    }

    /**
     * @param forumMsg the forumMsg to set
     */
    public void setForumMsg(List forumMsg) {
        this.forumMsg = forumMsg;
    }

}

```

ForumNewPostBean.java

```

package ie.edu.galway.abaltaabaschool.beans;

public class ForumNewPostBean {
    private String sUsername;
    private String sSubject;
    private String sMsg;
    /**
     * @return the sMsg
     */
    public String getSMsg() {
        return sMsg;
    }
    /**
     * @param msg the sMsg to set
     */
    public void setSMsg(String msg) {
        sMsg = msg;
    }
    /**
     * @return the sSubject
     */
    public String getSSubject() {

```

```

        return sSubject;
    }
    /**
     * @param subject the sSubject to set
     */
    public void setSSubject(String subject) {
        sSubject = subject;
    }
    /**
     * @return the sUsername
     */
    public String getSUsername() {
        return sUsername;
    }
    /**
     * @param username the sUsername to set
     */
    public void setSUsername(String username) {
        sUsername = username;
    }
}

```

ForumSearchBean.java

```

package ie.edu.galway.abaltaabaschool.beans;

import java.util.List;

public class ForumSearchBean {
    private List forumMsg;

    /**
     * @return the forumMsg
     */
    public List getForumMsg() {
        return forumMsg;
    }

    /**
     * @param forumMsg the forumMsg to set
     */
    public void setForumMsg(List forumMsg) {
        this.forumMsg = forumMsg;
    }
}

```

HomeBean.java

```
/*
 * Created on 24-Sep-2006
 *
 * TODO To change the template for this generated file go to
 * Window - Preferences - Java - Code Style - Code Templates
 */
package ie.edu.galway.abaltaabaschool.beans;

/**
 * @author Cathy
 *
 * TODO To change the template for this generated type comment go to
 * Window - Preferences - Java - Code Style - Code Templates
 */
public class HomeBean {
    /**
     * @return Returns the data.
     */
    public String getData() {
        return data;
    }
    /**
     * @param data The data to set.
     */
    public void setData(String data) {
        this.data = data;
    }
    /**
     * @return Returns the title.
     */
    public String getTitle() {
        return title;
    }
    /**
     * @param title The title to set.
     */
    public void setTitle(String title) {
        this.title = title;
    }
    private String title;
    private String data;
}
```

ReleaxResponseBean.java

```

package ie.edu.galway.abaltaabaschool.beans;

public class RealexResponseBean {
    private boolean hashesMatch;
    private boolean paymentOK;
    private String order_id;

    /**
     * @return the hashesMatch
     */
    public boolean isHashesMatch() {
        return hashesMatch;
    }

    /**
     * @param hashesMatch the hashesMatch to set
     */
    public void setHashesMatch(boolean hashesMatch) {
        this.hashesMatch = hashesMatch;
    }

    /**
     * @return the paymentOK
     */
    public boolean isPaymentOK() {
        return paymentOK;
    }

    /**
     * @param paymentOK the paymentOK to set
     */
    public void setPaymentOK(boolean paymentOK) {
        this.paymentOK = paymentOK;
    }

    /**
     * @return the order_id
     */
    public String getOrder_id() {
        return order_id;
    }

    /**
     * @param order_id the order_id to set
     */
    public void setOrder_id(String order_id) {
        this.order_id = order_id;
    }
}

```

BusinessObjects: ie\edu\galway\abaltaabaschool\businessobjects

FAQ.java

```
package ie.edu.galway.abaltaabaschool.businessobjects;

public class FAQ {

    private String question;
    private String answer;
    private int id;

    /**
     * @return the id
     */
    public int getId() {
        return id;
    }

    /**
     * @param id the id to set
     */
    public void setId(int id) {
        this.id = id;
    }

    /**
     * @return the question
     */
    public String getQuestion() {
        return question;
    }

    /**
     * @param question the question to set
     */
    public void setQuestion(String question) {
        this.question = question;
    }

    /**
     * @return the answer
     */
    public String getAnswer() {
        return answer;
    }

    /**
     * @param answer the answer to set
     */
    public void setAnswer(String answer) {
        this.answer = answer;
    }
}
```

```
}
```

Forum.java

```
package ie.edu.galway.abaltaabaschool.businessobjects;
```

```
public class Forum {
```

```
    private String fName;  
    private int id;
```

```
    /**  
     * @return the id  
     */
```

```
    public int getId() {  
        return id;  
    }
```

```
    /**  
     * @param id the id to set  
     */
```

```
    public void setId(int id) {  
        this.id = id;  
    }
```

```
    /**  
     * @return Returns the data.  
     */
```

```
    public String getForumName() {  
        return fName;  
    }
```

```
    /**  
     * @param data The data to set.  
     */
```

```
    public void setForumName(String fName) {  
        this.fName = fName;  
    }
```

```
}
```

ForumMessages.java

```
package ie.edu.galway.abaltaabaschool.businessobjects;
```

```
public class ForumMessages {
```

```
    private int msg_id;  
    private int f_id;  
    private int parent_msg_id;  
    private String user_name;
```

```

private String subject;
private String message;
private String date_time;
private String indent;
private String strUser;
private String strForum;

/**
 * @return the strForum
 */
public String getStrForum() {
    return strForum;
}

/**
 * @param strForum the strForum to set
 */
public void setStrForum(String strForum) {
    this.strForum = strForum;
}

/**
 * @return the strUser
 */
public String getStrUser() {
    return strUser;
}

/**
 * @param strUser the strUser to set
 */
public void setStrUser(String strUser) {
    this.strUser = strUser;
}

/**
 * @return the date_time
 */
public String getDate_time() {
    return date_time;
}

/**
 * @param date_time the date_time to set
 */
public void setDate_time(String date_time) {
    this.date_time = date_time;
}

/**
 * @return the f_id
 */
public int getF_id() {
    return f_id;
}

/**

```

```

    * @param f_id the f_id to set
    */
    public void setF_id(int f_id) {
        this.f_id = f_id;
    }

    /**
     * @return the message
     */
    public String getMessage() {
        return message;
    }

    /**
     * @param message the message to set
     */
    public void setMessage(String message) {
        this.message = message;
    }

    /**
     * @return the parent_msg_id
     */
    public int getParent_msg_id() {
        return parent_msg_id;
    }

    /**
     * @param parent_msg_id the parent_msg_id to set
     */
    public void setParent_msg_id(int parent_msg_id) {
        this.parent_msg_id = parent_msg_id;
    }

    /**
     * @return the subject
     */
    public String getSubject() {
        return subject;
    }

    /**
     * @param subject the subject to set
     */
    public void setSubject(String subject) {
        this.subject = subject;
    }

    /**
     * @return the msg_id
     */
    public int getMsg_id() {
        return msg_id;
    }

    /**
     * @param msg_id the msg_id to set

```

```

    */
    public void setMsg_id(int msg_id) {
        this.msg_id = msg_id;
    }

    /**
     * @return the indent
     */
    public String getIndent() {
        return indent;
    }

    /**
     * @param indent the indent to set
     */
    public void setIndent(String indent) {
        this.indent = indent;
    }

    /**
     * @return the user_name
     */
    public String getUser_name() {
        return user_name;
    }

    /**
     * @param user_name the user_name to set
     */
    public void setUser_name(String user_name) {
        this.user_name = user_name;
    }
}

```

ForumSearch.java

```

package ie.edu.galway.abaltaabaschool.businessobjects;

public class ForumSearch {
    private int msg_id;
    private int f_id;
    private int parent_msg_id;
    private String user_name;
    private String subject;
    private String message;
    private String date_time;
    private String msgText;
    /**
     * @return the msgText
     */
    public String getMsgText() {
        return msgText;
    }
}

```

```

    }
    /**
     * @param msgText the msgText to set
     */
    public void setMsgText(String msgText) {
        this.msgText = msgText;
    }
    /**
     * @return the date_time
     */
    public String getDate_time() {
        return date_time;
    }
    /**
     * @param date_time the date_time to set
     */
    public void setDate_time(String date_time) {
        this.date_time = date_time;
    }
    /**
     * @return the f_id
     */
    public int getF_id() {
        return f_id;
    }
    /**
     * @param f_id the f_id to set
     */
    public void setF_id(int f_id) {
        this.f_id = f_id;
    }
    /**
     * @return the message
     */
    public String getMessage() {
        return message;
    }
    /**
     * @param message the message to set
     */
    public void setMessage(String message) {
        this.message = message;
    }
    /**
     * @return the msg_id
     */
    public int getMsg_id() {
        return msg_id;
    }
    /**
     * @param msg_id the msg_id to set
     */
    public void setMsg_id(int msg_id) {
        this.msg_id = msg_id;
    }
    /**
     * @return the parent_msg_id

```

```

    */
    public int getParent_msg_id() {
        return parent_msg_id;
    }
    /**
     * @param parent_msg_id the parent_msg_id to set
     */
    public void setParent_msg_id(int parent_msg_id) {
        this.parent_msg_id = parent_msg_id;
    }
    /**
     * @return the subject
     */
    public String getSubject() {
        return subject;
    }
    /**
     * @param subject the subject to set
     */
    public void setSubject(String subject) {
        this.subject = subject;
    }
    /**
     * @return the user_name
     */
    public String getUser_name() {
        return user_name;
    }
    /**
     * @param user_name the user_name to set
     */
    public void setUser_name(String user_name) {
        this.user_name = user_name;
    }
}

```

Database: ie\edu\galway\abaltaabaschool\database

DBApp.java

```

/*
 * Created on 12-Aug-2006
 *
 * TODO To change the template for this generated file go to
 * Window - Preferences - Java - Code Style - Code Templates

```



```

    */
package ie.edu.galway.abaltaabaschool.database;

import java.sql.Connection;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import java.sql.PreparedStatement;
import java.util.Date;
import javax.naming.InitialContext;
import javax.naming.NamingException;
import javax.sql.DataSource;
import java.text.DateFormat;
import java.text.*;
import java.util.*;

/**
 * @author Cathy
 *
 * TODO To change the template for this generated type comment go to
 * Window - Preferences - Java - Code Style - Code Templates
 */
public class DBApp {

    public Connection getConnection()
    {
        Connection con = null;

        try
        {
            DataSource ds = getDataSource();
            con = ds.getConnection();
        }
        catch(SQLException se)
        {
            se.printStackTrace();
        }
        return con;
    }

    public DataSource getDataSource()
    {
        DataSource ds = null;
        try
        {
            InitialContext ctx = new InitialContext();
            ds = (DataSource)ctx.lookup("java:comp/env/jdbc/MySqlDS");
        }
        catch(NamingException ne)
        {
            ne.printStackTrace();
        }
        return ds;
    }
}

```

```

        public ResultSet executeQuery(Connection connection,String sql)
throws SQLException
        {
            Statement stmt = connection.createStatement();
            return stmt.executeQuery(sql);
        }

        public void executeUpdate(Connection connection,String sql)
throws SQLException
        {
            Statement stmt = connection.createStatement();
            stmt.executeUpdate(sql);
        }

        public boolean isDuplicateUsername(String userName) throws
SQLException
        {
            Connection connection = getConnection();
            String sql = "Select Uname from users where LCASE(Uname)
='" + userName.toLowerCase() + "'";
            ResultSet rs = executeQuery(connection, sql);
            if(rs != null && rs.next()){
                connection.close();
                return true;
            }
            connection.close();
            return false;
        }
    }
    //add new user to system
    public void addUser(Connection connection, String name, String
address, String phone, String email, String uName, String password,
String uType) throws SQLException
    {
        String sql = "insert into users values(null,
?,?,?,?,?,?)";
        PreparedStatement ps = connection.prepareStatement(sql);
        ps.setString(1, name);
        ps.setString(2, address);
        ps.setString(3, phone);
        ps.setString(4, email);
        ps.setString(5, password);
        ps.setString(6, uName);
        ps.executeUpdate();
        ps.close();
        addRole(connection, uType, uName);
    }

    public void addRole(Connection conn, String uType, String uName)
throws SQLException
    {
        String sql = "insert into roles values(null, ?,?)";
        PreparedStatement ps = conn.prepareStatement(sql);
        ps.setString(1, uType);
        ps.setString(2, uName);
        ps.executeUpdate();
        ps.close();
    }
}

```

```

//      add new forum message to system
public void addForumMessage(Connection connection, int fId, int
fParentId, String fUName, String subject, String message) throws
SQLException
{
    String sql = "insert into messages values(null,
?, ?, ?, ?, ?, now())";
    PreparedStatement ps = connection.prepareStatement(sql);
    ps.setInt(1, fId);
    ps.setInt(2, fParentId);
    ps.setString(3, fUName);
    ps.setString(4, subject);
    ps.setString(5, message);
    ps.executeUpdate();
    ps.close();
}

// add donation
public void submitDonation(Connection connection, String name,
String address, String phone, String email, int donation) throws
SQLException
{
    String sql = "insert into donations values(null,
?, ?, ?, ?, ?, null, null)";
    PreparedStatement ps = connection.prepareStatement(sql);
    ps.setString(1, name);
    ps.setString(2, address);
    ps.setString(3, phone);
    ps.setString(4, email);
    ps.setInt(5, donation);
    ps.executeUpdate();
    ps.close();
}

public void updateDonation(Connection connection, String hash,
String result, String orderId) throws SQLException,
NumberFormatException{
    String sql = "update donations set hash_match = ?,
result = ? where order_id = ?";
    PreparedStatement ps = connection.prepareStatement(sql);
    ps.setString(1, hash);
    ps.setString(2, result);
    ps.setInt(3, Integer.parseInt(orderId));
    ps.executeUpdate();
    ps.close();
}

public double getDonationTotal() throws SQLException{
    String sql="select sum(amount)/100 as amt from donations
where result = '00'";
    Connection connection = getConnection();
    ResultSet rs = executeQuery(connection, sql);
    double amt = 0;
    if(rs != null && rs.next()){

```

```

        amt = rs.getDouble("amt");
    }
    connection.close();
    return amt;
}

public int getOrderID(Connection connection) throws SQLException
{
    String sql = "select LAST_INSERT_ID()";
    ResultSet rs = executeQuery(connection, sql);
    if(rs!=null && rs.next()){
        return rs.getInt(1);
    }
    return -1;
}
}

```

Stats: ie\edu\galway\abaltaabaschool\stats

Donations.java

```

package ie.edu.galway.abaltaabaschool.stats;

import java.awt.*;
import org.jfree.chart.JFreeChart;
import org.jfree.chart.plot.ThermometerPlot;
import org.jfree.data.general.ValueDataset;
import org.jfree.ui.RectangleInsets;
import org.jfree.chart.servlet.ServletUtilities;
import ie.edu.galway.abaltaabaschool.database.DBApp;

import java.sql.*;

import java.awt.Color;

import org.jfree.chart.*;
import org.jfree.chart.axis.*;
import org.jfree.chart.entity.*;

```

```

import java.io.IOException;
import javax.servlet.http.HttpSession;

import org.jfree.data.general.DefaultValueDataset;

public class Donations {

    public String createChart(ValueDataset valuedataset, HttpSession
session)
    {
        String fileName = "";

        try{
            ThermometerPlot thermometerplot = new
ThermometerPlot(valuedataset);
            JFreeChart jfreechart = new JFreeChart("Donations",
JFreeChart.DEFAULT_TITLE_FONT, thermometerplot, false);

            jfreechart.setBorderPaint(Color.white);
            jfreechart.setBorderVisible(false);
            jfreechart.setBackgroundPaint(Color.white);

            thermometerplot.setInsets(new RectangleInsets(5D, 5D,
5D, 5D));
            thermometerplot.setPadding(new RectangleInsets(10D, 10D,
10D, 10D));
            thermometerplot.setThermometerStroke(new
BasicStroke(2.0F));
            thermometerplot.setThermometerPaint(Color.lightGray);
            thermometerplot.setUnits(1);

            double high = 1000;
            high = 1000 + (valuedataset.getValue().intValue()/1000 *
1000);

            thermometerplot.setRange(0, high);
            thermometerplot.setRangeAxis(new NumberAxis("EUR"));
            thermometerplot.setAxisLocation(2);
            thermometerplot.setUnits(ThermometerPlot.UNITS_NONE);
            thermometerplot.setOutlinePaint(Color.white);
            thermometerplot.setUseSubrangePaint(false);
            thermometerplot.setShowValueLines(false);
            thermometerplot.setMercuryPaint(new Color(102,102,153));

            thermometerplot.setFollowDataInSubranges(false);

            thermometerplot.setSubrangePaint(ThermometerPlot.CRITICAL,
Color.WHITE);
            thermometerplot.setSubrangePaint(ThermometerPlot.NORMAL,
Color.WHITE);
            thermometerplot.setSubrangePaint(ThermometerPlot.WARNING, Color.WHITE);

```

```

        ChartRenderingInfo info = new ChartRenderingInfo(new
StandardEntityCollection());

        ServletUtilities.setTempFilePrefix("public");
        fileName =
ServletUtilities.saveChartAsJPEG(jfreechart, 105, 300, info, session);

        ChartUtilities.writeImageMap(new
java.io.PrintWriter(System.out), fileName, info, false);

        System.out.flush();

    }
    catch(IOException ioe){
        ioe.printStackTrace();
    }

    return fileName;
}

public DefaultValueDataset getDonationDataset(){

    try{
        DBApp dba = new DBApp();
        DefaultValueDataset dataset = new
DefaultValueDataset(dba.getDonationTotal());
        return dataset;
    }
    catch(SQLException se){
        se.printStackTrace();
    }

    return new DefaultValueDataset(0);
}
}

```

Util: ie\edu\galway\abaltaabaschool\util

MailUtil.java

```

package ie.edu.galway.abaltaabaschool.util;

import java.io.IOException;
import java.util.Properties;

import javax.mail.*;
import javax.mail.internet.*;
import java.util.Date;
import java.util.*;
import java.io.*;

public class MailUtil
{
    public final static String MAILHOST =
"abaltaabaschool.galway.edu.ie";

    public MailUtil(String to,String subject,String message)
    {
        this.to = to;
        this.subject = subject;
        this.message = message;
    }

    public MailUtil(String to,String subject,String message,String from)
    {
        this.to = to;
        this.subject = subject;
        this.message = message;
        this.from = from;
    }

    private String to;
    private String from;
    private String subject;
    private String attachment;
    private String cc;
    private String bcc;
    private String message;

    public void setTo(String to)
    {
        this.to = to;
    }
    public void setFrom(String from)
    {
        this.from = from;
    }
    public void setSubject(String subject)
    {
        this.subject = subject;
    }
    public void setAttachment(String attachment)
    {
        this.attachment = attachment;
    }
}

```

```

public void setCc(String cc)
{
    this.cc = cc;
}
public void setBcc(String bcc)
{
    this.bcc = bcc;
}
public void setMessage(String message)
{
    this.message = message;
}

public String getTo()
{
    return to;
}
public String getFrom()
{
    return from;
}
public String getSubject()
{
    return subject;
}
public String getAttachment()
{
    return attachment;
}
public String getCc()
{
    return cc;
}
public String getBcc()
{
    return bcc;
}
public String getMessage()
{
    return message;
}

public void smtpSend(String mailhost) throws MessagingException
{
    Properties props = System.getProperties();
    props.put("mail.smtp.host", mailhost);
    Session session = Session.getInstance(props, null);
    Message msg = new MimeMessage(session);
    if (from != null)
        msg.setFrom(new InternetAddress(from));
    else
        msg.setFrom();

    msg.setRecipients(Message.RecipientType.TO,

```



```

InternetAddress.parse(to,
false));
    if (cc != null)
        msg.setRecipients(Message.RecipientType.CC,
InternetAddress.parse(cc,
false));
    if (bcc != null)
        msg.setRecipients(Message.RecipientType.BCC,
InternetAddress.parse(bcc,
false));

    msg.setSubject(subject);

    msg.setText(message);

    msg.setHeader("X-Mailer", "Abalta");
    msg.setSentDate(new Date());

    // send the thing off
    Transport.send(msg);
    System.out.println("\nMail was sent successfully.");

}

public Folder openFolderIMAP(String host,String user,String
password,String mbox,int freq) throws
NoSuchProviderException,MessagingException,InterruptedException
{
    Properties props = System.getProperties();

    // Get a Session object
    Session session = Session.getInstance(props, null);
    // session.setDebug(true);

    // Get a Store object

    Store store = session.getStore("imap");

    // Connect
    store.connect(host, user, password);
    Folder folder = store.getFolder(mbox);

    folder.open(Folder.READ_WRITE);

    return folder;
    // Add messageCountListener to listen for new messages
}

public String[] parseMessage(Message m,String[] fields)
{
    String[] csr_details = new String[fields.length+1];
    try
    {

```

```

int pos = 0, pos2 = 0;
Part p = m;
StringBuffer buffy = new StringBuffer(m.getContent().toString());
for(int i=0;i<fields.length;i++)
{
    pos = buffy.indexOf(fields[i]);
    if(pos!=-1)
    {
        pos = buffy.indexOf(":",pos);
        pos2 = buffy.indexOf("\r",pos);

        if(pos!=-1&&pos2!=-1)
        {
            if((pos + 2) > pos2)
                pos++;
            else
                pos = pos + 2;
            csr_details[i] = buffy.substring(pos,pos2);
        }
        else
        {
            csr_details[fields.length] = "Error";//Send this mail
to the Error folder
            return csr_details;
        }
    }
    else
    {
        csr_details[fields.length] = "Invalid";
        return csr_details;
    }
}

catch(MessagingException me)
{
    me.printStackTrace();
}
catch(IOException ioe)
{
    ioe.printStackTrace();
}
csr_details[fields.length] = "OK";
return csr_details;
}
}

```

Releax.java

```
package ie.edu.galway.abaltaabaschool.util;
```

```

import java.security.MessageDigest;
import java.util.Calendar;

public class Relax {
    public final static String SECRET = "zEHickEtho";
    public final static String MERCHANT_ID = "abalta";
    public final static String CURRENCY = "EUR";

    /*
    This function creates the timestamp in the format required by
    Realex Payments .
    */
    public String getTimestamp() {
        String timestamp;
        Calendar now = Calendar.getInstance();

        timestamp = "" + now.get(Calendar.YEAR);

        if ((now.get(Calendar.MONTH) + 1) < 10) {
            timestamp += "0" + (now.get(Calendar.MONTH) + 1);
        } else {
            timestamp += "" + (now.get(Calendar.MONTH) + 1);
        }

        if (now.get(Calendar.DAY_OF_MONTH) < 10) {
            timestamp += "0" +
now.get(Calendar.DAY_OF_MONTH);
        } else {
            timestamp += "" + now.get(Calendar.DAY_OF_MONTH);
        }

        if (now.get(Calendar.HOUR_OF_DAY) < 10) {
            timestamp += "0" + now.get(Calendar.HOUR_OF_DAY);
        } else {
            timestamp += "" + now.get(Calendar.HOUR_OF_DAY);
        }

        if (now.get(Calendar.MINUTE) < 10) {
            timestamp += "0" + now.get(Calendar.MINUTE);
        } else {
            timestamp += "" + now.get(Calendar.MINUTE);
        }

        if (now.get(Calendar.SECOND) < 10) {
            timestamp += "0" + now.get(Calendar.SECOND);
        } else {
            timestamp += "" + now.get(Calendar.SECOND);
        }

        return timestamp;
    }
    /*

```

SHA1

Below is the code for creating the digital signature using the algorithm.

```
*/

public String calcSHA1(String toBeHashed) {
    byte[] digestValue = new byte[0];
    StringBuffer sb = new StringBuffer();
    MessageDigest md;

    try {
        md = MessageDigest.getInstance("SHA");
        md.update(toBeHashed.getBytes());
        digestValue = md.digest();

        for (int i = 0; i < digestValue.length; i++) {
            String c = Integer.toHexString(digestValue[i]);
            if (digestValue[i] < 0) c = c.substring(6);
            if (c.length() < 2) c = "0" + c;
            sb.append(c);
        }

    } catch (Exception e) {
        e.printStackTrace();
    }

    return sb.toString();
}
```

Appendix D – The Build Code

build.xml

```
<project name="Abalta" basedir=".." default="copyToWebapp">

  <!-- Local system paths -->
  <property name="servlet.jar" value="/jvasoft/lib/servlet.jar"/>
  <property name="jdbc20ext.jar" value="/jvasoft/lib/jdbc2_0-stdext.jar"/>
  <!-- NOTE: If "dist" target is used, a local
    "projects/lib" directory will be utilized or created -->
  <property name="distpath.project" value="/projects/lib"/>
  <property name="resinWebapp.project" value="D:/Program Files/resin-3.0.21/resin-
3.0.21/webapps"/>

  <!-- Project settings -->
  <property name="project.title" value="Cathy's Abalta Project "/>
  <property name="project.distname" value="ROOT"/>
  <property name="project.version" value="1.1"/>

  <!-- Path settings -->
  <property name="doc.path" value="/doc/api"/>
  <property name="doc.src" value="/src/java"/>

  <!-- classpath for Struts 1.1 -->
  <path id="compile.classpath">
    <pathelement path="lib/activation.jar"/>
    <pathelement path="lib/dsn.jar"/>
    <pathelement path="lib/pop3.jar"/>
    <pathelement path="lib/imap.jar"/>
    <pathelement path="lib/mailapi.jar"/>
    <pathelement path="lib/smtp.jar"/>
    <pathelement path="lib/commons-beanutils.jar"/>
    <pathelement path="lib/commons-digester.jar"/>
    <pathelement path="lib/struts.jar"/>
    <pathelement path="lib/jfreechart-1.0.5.jar"/>
  </path>
</project>
```

```

        <pathelement path="lib/jcommon-1.0.9.jar"/>
    <pathelement path="classes"/>
    <pathelement path="D:/Program Files/resin-3.0.21/resin-3.0.21/lib/jsdk-24.jar"/>
    <pathelement path="\${classpath}"/>
</path>

```

```

<!-- Check timestamp on files -->
<target name="prepare">
    <tstamp/>
</target>

```

```

<!-- Copy any resource or configuration files -->
<target name="resources">
    <copy todir="classes" includeEmptyDirs="no">
        <fileset dir="src/java">
            <patternset>
                <include name="**/*.conf"/>
                <include name="**/*.properties"/>
                <include name="**/*.xml"/>
            </patternset>
        </fileset>
    </copy>
</target>

```

```

<!-- Normal build of application -->
<target name="compile" depends="prepare,resources">
    <javac srcdir="src" destdir="classes">
        <classpath refid="compile.classpath"/>
    </javac>
</target>

```

```

<!-- Remove classes directory for clean build -->
<target name="clean"
    description="Prepare for clean build">
    <delete dir="classes"/>
    <mkdir dir="classes"/>
</target>

```

```

<!-- Build Javadoc documentation -->
<target name="javadoc"
    description="Generate JavaDoc API docs">

```

```

<delete dir="./doc/api"/>
<mkdir dir="./doc/api"/>
<javadoc sourcepath="./src/java"
  destdir="./doc/api"
  classpath="${servlet.jar}:${jdbc20ext.jar}"
  packagenames="*"
  author="true"
  private="true"
  version="true"
  windowtitle="${project.title} API Documentation"
  doctitle="&lt;h1&gt;${project.title} Documentation (Version
${project.version})&lt;/h1&gt;"
  bottom="Copyright &#169; 2002">
  <classpath refid="compile.classpath"/>
</javadoc>
</target>

<!-- Build entire project -->
<!-- <target name="project" depends="clean,prepare,compile,javadoc"/> -->
<target name="project" depends="clean,prepare,compile"/>

<!-- Create binary distribution -->
<target name="dist"
  description="Create binary distribution">

  <mkdir dir="${distpath.project}"/>

  <jar
    jarfile="${distpath.project}/${project.distname}.jar"
    basedir="./classes"/>
  <copy
    file="${distpath.project}/${project.distname}.jar"
    todir="${distpath.project}"/>

  <war
    basedir=".."
    warfile="${distpath.project}/${project.distname}.war"
    webxml="web.xml">
    <exclude name="${distpath.project}/${project.distname}.war"/>
  </war>

</target>

```

```

<!-- Build project and create distribution-->
<target name="all" depends="project,dist"/>

<target name="copyToWebapp" depends="all">
  <!--<copy
    file="${distpath.project}/${project.distname}.war"
    todir="${resinWebapp.project}"/> -->
    <unzip src="${distpath.project}/${project.distname}.war"
dest="${resinWebapp.project}/ROOT"/>

</target>

</project>

```


Appendix E – The Database Code

Contents Table

```
CREATE TABLE `contents` (  
  `id` int(11) NOT NULL auto_increment,  
  `page_name` varchar(50) default '',  
  `location` varchar(50) default '',  
  `data` longtext NOT NULL,  
  `active` varchar(50) NOT NULL default '',  
  PRIMARY KEY (`id`)  
) ENGINE=MyISAM AUTO_INCREMENT=64 DEFAULT CHARSET=latin1;  
  
INSERT INTO `contents` (`id`,`page_name`,`location`,`data`,`active`)  
VALUES  
  (56,'contact.jsp','b','<p>&Aacute;balta School<br />No 6, Cuan Glas<br />Bishop O&#8217Donnell Road<br />Galway.<br />Phone: 091-589646<br />Email: <a href=\"mailto:info@abaltaabaschool.galway.edu.ie\">info@abaltaabaschool.galway.edu.ie</a></p>','yes'),  
  (55,'contact.jsp','a','Contact Information','yes'),  
  (54,'header.jsp','b','It is &Aacute;balta&#8217s mission to provide the highest quality of education to its students with autism.','yes'),  
  (53,'header.jsp','a','Mission','yes'),  
  (52,'mission.jsp','a','Mission Statements','yes'),  
  (51,'testimonials.jsp','a','Parent Testimonials','yes'),  
  (50,'index.jsp','a','&Aacute;balta School','yes'),  
  (47,'literature.jsp','a','Suggested Reading','yes'),  
  (46,'aba.jsp','a','Applied Behaviour Analysis','yes'),  
  (45,'literature.jsp','o','<p><i>When Everybody Cares: Case Studies of ABA with People with Autism,</i> by Bobby Newman, Ph.D., CBA</p>','yes'),  
  (44,'literature.jsp','m','<p><i>Unraveling the Mystery of Autism and Pervasive Developmental Disorder: A Mother&#8217s Story of Research and Recovery,</i> by Karyn Seroussi and Bernard Rimland</p>','yes'),  
  (43,'literature.jsp','n','<p><i>Teaching Language to Children with Autism or Other Developmental Disabilities,</i> by Mark L. Sundberg, Ph.D., and James W. Partington, Ph.D.</p>','yes');  
INSERT INTO `contents` (`id`,`page_name`,`location`,`data`,`active`)  
VALUES  
  (42,'literature.jsp','m','<p><i>Special Diets for Special Kids,</i> by Lisa Lewis. General information and recipes for a gluten- and casein-free diet.</p>','yes'),  
  (41,'literature.jsp','l','<p><i>Sound of a Miracle,</i> by Annabel Stehli. A mother&#8217s story of her child&#8217s recovery from hypersensitive hearing-induced autism.</p>','yes'),  
  (40,'literature.jsp','k','<p><i>Somebody, Somewhere,</i> by Donna Williams. A woman&#8217s continuing experience living with autism and sensory dysfunction.</p>','yes'),  
  (39,'literature.jsp','j','<p><i>Right From the Start: Behavioral Intervention for Young Children with Autism. A Guide for Parents and
```

Professionals,</i> by Sandra L. Harris, Ph.D., and Mary Jane Weiss, Ph.D.</p>','yes'),

(38,'literature.jsp','i','<p><i>The Out-of-Sync Child, Recognizing and Coping with sensory Integration Dysfunction</i> by Carol Stock Kranowitz.</p>','yes'),

(37,'literature.jsp','h','<p><i>Nobody, Nowhere,</i> by Donna Williams. A woman's experience living with autism and severe sensory dysfunction.</p>','yes'),

(36,'literature.jsp','g','<p><i>Let Me Hear Your Voice, a Family's Triumph Over Autism,</i> Catherine Maurice. A mother's story of finding and using ABA to help her children recover from autism.</p>','yes');

INSERT INTO `contents` (`id`,`page_name`,`location`,`data`,`active`)

VALUES

(35,'literature.jsp','f','<p><i>Diet Intervention and Autism: Implementing a Gluten Free and Casein Free Diet for Autistic Children and Adults: a Guide for Parents,</i> by Marilyn Le Breton and Rosemary Kessick.</p>','yes'),

(34,'literature.jsp','e','<p><i>The Child with Special Needs,</i> by Stanley Greenspan and Serena Weider. Describes Floortime intervention, which can help your child become more emotionally connected with you.</p>','yes'),

(33,'literature.jsp','d','<p><i>Behavioral Intervention for Young Children with Autism,</i> edited by Catherine Maurice, Gina Greene, and Stephen C. Luce. Includes a basic curriculum for Discrete Trial Therapy programs.</p>','yes'),

(32,'literature.jsp','c','<p><i>A Work in Progress: Behavior Management Strategies and a Curriculum for Intensive Behavioral Treatment of Autism,</i> by Ron Leaf and John McEachin, editors.</p>','yes'),

(31,'literature.jsp','b','<p>The following books can provide you with more information about ABA, Discrete Trial Therapy, and autism interventions. </p>','yes'),

(30,'testimonials.jsp','i','<p>We would encourage any parent to learn and use the skills of applied behavioural analysis. Certainly, since implementation of this approach, we have seen great progress in our little girl.</i>" -- Veronica and Walter McInerney</p>','yes');

INSERT INTO `contents` (`id`,`page_name`,`location`,`data`,`active`)

VALUES

(29,'testimonials.jsp','h','Task Analysis: the breakdown of tasks into small manageable steps.Rules of reinforcement: immediate positive reward of good behaviour.Behaviour analysis: looking at the environment to determine the possible cause of the behaviour.','yes'),

(28,'testimonials.jsp','g','<p>"<i>As parents, we have found that our daughter has made great improvement in all aspects of her academic learning and understanding since she started learning through the ABA method. The ABA principles, mentioned below, have enabled us to solve many behavioural problems we encountered along the way and have also provided us with the skills to teach our daughter effectively. For example,</i></p>','yes'),

(27,'testimonials.jsp','f','<p>"<i>The teaching method of Applied Behaviour Analysis, together with the great staff, consultants/therapists at Ábalta, and Shauna's home tutor, have all given us the indescribable pleasure of seeing our daughter's transformation over the past two years. Shauna has developed into a much happier child who is now comfortable in any

environment. She has gone from a child with very poor communication skills to a child who can now communicate her needs with little frustration, and from a child who could not sit and apply herself to one task to a child who now enjoys and embraces learning. Shauna is starting mainstream school in September on a part-time basis, a stage we never thought possible. Our sincere thanks to you all!

-- Audrey Foy and Vincent Marchant

(26,'testimonials.jsp','e','<p>"<i>We are happy that Seamus is attending the Ábalta School and is now getting an appropriate education. The ABA approach, together with Speech and Language Therapy and Occupational Therapy, are having a very positive effect on his general development. The presence of behavioural plans designed specifically for Seamus and continuous assessment have ensured that Seamus is reaching his potential in all areas.</i>"

-- Joe and Mary Neylon

INSERT INTO `contents` (`id`,`page_name`,`location`,`data`,`active`)
VALUES

(25,'testimonials.jsp','d','<p>"<i>After Liam's diagnosis, we were greatly concerned by the lack of services and autism-specific placements available to him and feel that valuable time was wasted before we were fortunate enough to become involved in Ábalta. It is entirely due to the hard work and dedication of the parents and staff involved in Ábalta that Liam has made such wonderful progress. To our delight, Liam will be starting school, part-time, with his typically developing peers this September.</i>"

-- Claire Haynes

(24,'testimonials.jsp','c','<p>"<i>Having tried other education options, we feel that the combination of the one-to-one ABA programme, along with the input from his Speech Therapist and Occupational Therapist, has greatly improved Conor's quality of life since he started attending Ábalta. Because he has learned many new skills, he is a much happier child overall.</i>"

-- Catherine and John Moloney

(23,'testimonials.jsp','b','<p><i>"Setting up Ábalta was a huge leap of faith, but one that, thank God, has really paid off. Colum loves going into school, and with one-to-one teaching, I know all the time he spends at school is constructive. The staff are all young and energetic and really open to trying all approaches to help move Colum on. This is backed up by weekly hands-on work with our professional team of behavioural, speech, and occupational therapists. Having looked at schools around Ireland, I believe Ábalta is one of the best centres for intervention for autistic children in the country.</i>"

-- Nicola and James McNabb

INSERT INTO `contents` (`id`,`page_name`,`location`,`data`,`active`)
VALUES

(22,'mission.jsp','p','<p>Upon assessment, the occupational therapist creates an individualised occupational therapy program and provides in situ training to teaching staff, parents, and consulting therapists to ensure appropriate implementation. When possible, these programmes are embedded in the ABA curriculum in consultation with the Educational Director to ensure smooth and consistent application. Programmes are monitored and updated to ensure they are continually meeting the needs of the child. Ongoing training of teaching staff and parents on relevant topics takes place as needed, both within and outside the students' school day.</p>

(21,'mission.jsp','o','<p>Occupational Therapy: Ábalta's occupational therapy service aims to assess each child within his or

her school, home, and/or social environments to devise an intervention programme specific to each child. Each child is assessed with regard to their skills and deficits in the following areas: fine motor, gross motor, planning and sequencing, proprioceptive, vestibular, tactile, visual, and auditory.</p>','yes'),

(20,'mission.jsp','n','<p>Provision of training to staff and parents is recommended and carried out in consultation with the Educational Director.</p>','yes');

INSERT INTO `contents` (`id`,`page_name`,`location`,`data`,`active`) VALUES

(19,'mission.jsp','m','<p>Intervention aims to enable those involved with the child to provide the optimum communication environment. Both direct and indirect therapy is offered, and recommendations, aims, and objectives are integrated within the ABA curriculum.</p>','yes'),

(18,'mission.jsp','l','<p>Speech and Language Therapy: Ábalta’s speech and language therapy service aims to establish the nature of each child’s communication skills, strengths, and needs, in terms of development and his/her speech, language, and interaction. This is achieved by working within a multi-disciplinary framework with parents, teachers, tutors, and consulting therapists.</p>','yes'),

(17,'mission.jsp','k','We respect the right to privacy for all children and families at Ábalta.</p>','yes'),

(16,'mission.jsp','j','We strive to create a positive educational environment for our students.','yes'),

(15,'mission.jsp','i','We believe each parent should take an active role in their child's educational future; thus, they are invited to participate in the planning of their child’s educational program.','yes');

INSERT INTO `contents` (`id`,`page_name`,`location`,`data`,`active`) VALUES

(14,'mission.jsp','h','We are committed to recognising each child as an individual and appreciating their unique personalities.','yes'),

(13,'mission.jsp','g','We are committed to ensuring the safety and well-being of our students at all times.','yes'),

(12,'mission.jsp','f','Every child with autism has the right to an education that is grounded in scientific evidence of efficacy.','yes'),

(7,'aba.jsp','e','<p>Parental input is essential in a quality DTT/ABA programme. Parents provide information not otherwise attainable, including whether skills are generalising from the school setting to home, what self-help skills and social skills need to be addressed, etc. Thus, parents must be involved in their childrens programmes from the start, by attending IEP meetings, attending programme review meetings, and actively generalising skills learned in school to the child’s other environments.</p>','yes'),

(6,'aba.jsp','d','<p>Each child has a curricular program designed specifically for them, according to their needs. Programmes are constantly updated to reflect children’s progress and changing needs. All areas of development are considered when programs and IEPs are developed, including communication skills, social skills, self-help skills, and academic skills.</p>','yes');

INSERT INTO `contents` (`id`,`page_name`,`location`,`data`,`active`) VALUES

(5,'aba.jsp','c','<p>DTT/ABA breaks down difficult concepts into small, manageable parts that can then be taught successfully. As

children's skills grow, the concepts are re-combined into gradually more difficult concepts.

(4,'aba.jsp','b','<p>Applied Behaviour Analysis, or ABA, refers to a category of interventions that uses positive reinforcement to increase and decrease behaviours so that a person's quality of life improves. There are several teaching modalities that employ ABA; at Ábalta School, we use a form of ABA called Discrete Trial Therapy. This methodology was studied extensively, and research found that DTT/ABA provided intensively resulted in 47% of the children studied reaching a level of skills so high as to make them indistinguishable from their typically developing peers.</p>','yes'),

(11,'mission.jsp','e','Every child with autism has the right to an education.','yes'),

(10,'mission.jsp','d','<p>Ábalta is guided by the following tenets:','yes');
INSERT INTO `contents` (`id`,`page_name`,`location`,`data`,`active`) VALUES

(9,'mission.jsp','c','<p>It is Ábalta's mission to provide ongoing training and support to the families of our students.</p>','yes'),

(8,'mission.jsp','b','<p>It is Ábalta's mission to provide the highest quality of education to its students with autism. This is achieved by only using a scientifically proven educational methodology, specifically Applied Behavioural Analysis. By doing so, Ábalta is able to provide each child with an education that is tailored to his or her individual needs. </p>','yes'),

(3,'index.jsp','d','<p>Ábalta's logo consists of a number of building blocks which represent the ethos behind the education methodology used in the school, known as Applied Behavioural Analysis (ABA). This methodology breaks down educational tasks into tiny building blocks which a child with autism can be taught. The blocks are put together to teach a child a skill.</p>','yes'),

(2,'index.jsp','c','<p>Ábalta School expanded in 2003 to cater for 12 children at a site in Cuan Glas in Galway. In 2005 Ábalta School opened an additional facility in Newcastle in Galway to cater for an additional six children. Ábalta currently caters for 18 children and has had four of its students transfer into mainstream services.</p>','yes');
INSERT INTO `contents` (`id`,`page_name`,`location`,`data`,`active`) VALUES

(1,'index.jsp','b','<p>Ábalta School was borne out of the identified need for children with autism to get a specific focused education. In July, 2001, four sets of parents came together to establish the school and had it up and running by September, 2001. It took eight months of hard work, political negotiating, and lobbying, culminating with a high court case to attain funding from the state. </p><p>Ábalta opened in September, 2001, with six students. Four of these children had, prior to attending Ábalta, been educated in home-based ABA programmes. The other two children had been receiving other types of autism services.</p>','yes'),

(60,'register.jsp','a','Registration','yes'),

(61,'register.jsp','b','<tr><td align="center" colspan="2">Please enter your details</tr>','yes');

Donations Table

```
CREATE TABLE `donations` (  
  `order_id` int(10) unsigned NOT NULL auto_increment,  
  `name` text NOT NULL,  
  `address` text NOT NULL,  
  `phone` varchar(45) NOT NULL default '',  
  `email` text NOT NULL,  
  `amount` int(10) unsigned NOT NULL default '0',  
  `hash_match` varchar(100) default NULL,  
  `result` text,  
  PRIMARY KEY (`order_id`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

Email Table

```
CREATE TABLE `email` (  
  `id` int(11) NOT NULL default '0',  
  `email` text,  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

FAQ Table

```
CREATE TABLE `faq` (  
  `id` int(10) unsigned NOT NULL auto_increment,  
  `question` varchar(200) NOT NULL default '',  
  `answer` text NOT NULL,  
  `owner` varchar(20) NOT NULL default '',  
  `archive` varchar(15) NOT NULL default '',  
  `active` varchar(15) NOT NULL default '',  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
INSERT INTO `faq` (`id`,`question`,`answer`,`owner`,`archive`,`active`)  
VALUES
```

```
(1,'<i><b>What is Autism</i></b>','</br>Autism is a disability that  
affects the normal development of the brain in areas of social  
interaction and communication. The first signs of autism usually appear  
as developmental delays before the age of 3. Autism is described as a  
&quot;spectrum&quot; disorder. This means that the symptoms and  
characteristics of autism can present themselves in a wide variety of  
combinations and can range from mild to severe. Two children with the
```

same diagnosis can act very differently from one another and have varying skills. Autism is a serious developmental neurological disorder, marked by severe difficulties in communication and forming relationships with people, in developing language and in using abstract concepts. Characteristics include repetitive and limited patterns of behaviour and obsessive resistance to tiny changes in familiar surroundings or routines. ', 'admin', 'yes', 'yes'),

(2, '*How can I have my child educationally assessed and have an individual education plan (IEP) drawn up?</i>', '

If your child is attending school, the school should be asked about its arrangements for psycho-educational assessment. Psycho-educational assessment, in this context, is concerned with the identification of the child's educational needs arising from ASD and from any intellectual disability. It is not intended to diagnose or confirm the existence of an ASD.<p>In schools served by the National Educational Psychological Service (NEPS), the principal can request the involvement of the area educational psychologist. The school should be able to draw up an IEP for the child on the basis of the educational psychologists report and any other existing professional reports. The child's parents should also be consulted about the IEP, which should lay out specific objectives to be achieved by the child within a particular time-frame and should indicate arrangements for review.</p><p>If the school is not yet covered by NEPS, the school may commission a psychological assessment under the Scheme for Commissioning Psychological Assessments (SCPA). This scheme was introduced by the DES as a temporary measure to provide access to psychological assessments for schools not yet covered by NEPS. Details concerning the SCPA are available on the DES website at www.irlgov.ie/educ</p><p>If your school-age child is not currently attending any school, the DES will arrange for a psycho-educational assessment.</p>', 'admin', 'yes', 'yes');*

Forum Table

```
CREATE TABLE `forum` (
  `f_id` int(10) unsigned NOT NULL auto_increment,
  `forum_name` varchar(70) NOT NULL default '',
  `active` varchar(45) NOT NULL default '',
  PRIMARY KEY (`f_id`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

Messages Table

```
CREATE TABLE `messages` (
  `msg_id` int(10) unsigned NOT NULL auto_increment,
  `f_id` int(10) unsigned NOT NULL default '0',
  `parent_msg_id` int(10) unsigned NOT NULL default '0',
```

```

    `user_name` varchar(45) NOT NULL default '',
    `subject` text NOT NULL,
    `message` text NOT NULL,
    `date_time` datetime NOT NULL default '0000-00-00 00:00:00',
    PRIMARY KEY (`msg_id`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

```

Roles Table

```

CREATE TABLE `roles` (
  `id` int(10) unsigned NOT NULL auto_increment,
  `role_type` varchar(45) NOT NULL default '',
  `uname` varchar(45) NOT NULL default '',
  PRIMARY KEY (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

```

Users Table

```

CREATE TABLE `users` (
  `UID` int(10) unsigned NOT NULL auto_increment,
  `Name` varchar(45) NOT NULL default '',
  `Address` varchar(45) NOT NULL default '',
  `Phone` varchar(45) NOT NULL default '',
  `Email` varchar(45) NOT NULL default '',
  `Password` varchar(45) NOT NULL default '',
  `Uname` varchar(45) NOT NULL default '',
  PRIMARY KEY (`UID`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

```


References:

[1] Steve Masover (June 2004) Model-View-Controller: A Design Pattern for Software.

Retrieved July 26th, 2007 from Berkeley IST-ASAG: Talks and Publications. Website:

<http://ist.berkeley.edu/as/ag/pub/pdf/mvc-seminar.pdf>

[2] eNode Inc (2002) Model-View-Controller Pattern. Retrieved July 26th, 2007 from

ENode Inc. Website: <http://www.enode.com/x/markup/tutorial/mvc.html>

[3] Brian Kotek (Oct 2002) MVC design pattern brings about better organization and code reuse. Retrieved July 26th, 2007 from TechRepublic. Website:

<http://articles.techrepublic.com.com/5100-22-1049862.html>

[4] Apache Foundation, Struts Introduction. Retrieved July 27th, 2007 from the Apache Software Foundation. Website:

<http://struts.apache.org/1.x/userGuide/introduction.html#controllerConcepts>

[5] Struts and model-view-controller design pattern (2004). Retrieved August 1st, 2007 from IBM Software Information Center. Website:

<http://publib.boulder.ibm.com/infocenter/wsadhelp/v5r1m2/index.jsp?topic=/com.ibm.etools.struts.doc/topics/cstrdoc001.html>

[6] Roger Sessions (March 2001). Java 2 Enterprise Edition (J2EE) versus The .NET Platform Two Visions for eBusinesses. Retrieved July 26th, 2007 from ObjectWatch Inc.

Website: <http://www.objectwatch.com/FinalJ2EEandDotNet.doc>

[7] Scott McPherson (2000) JavaServer Pages: A Developer's Perspective. Retrieved July 26th, 2007 from Sun Developer Network (SDN).

<http://java.sun.com/developer/technicalArticles/Programming/jsp/>

[8] JavaBeans FAQ: General Questions. Retrieved July 27th, 2007 from Sun Developer Network (SDN). Website:

<http://java.sun.com/products/javabeans/faq/faq.general.html#Q1>

[9] What is JDBC? Retrieved July 26th, 2007 from Rose India. Website:

<http://www.roseindia.net/jdbc/what-is-jdbc.shtml>

[10] David Flanagan (2002) Java in a Nutshell. O'Reilly (4th Edition), California.

[11] Java Authentication and Authorization Service (JAAS). Retrieved July 27th, 2007 from Sun Developer Network (SDN). Website: <http://java.sun.com/products/jaas/>

[12] Scott Oaks, (2001) Java Security. O'Reilly (Second Edition). California

- [13] An Overview of Resin (2003). Retrieved July 27th, 2007 from Media Wiley.
Website: http://media.wiley.com/product_data/excerpt/36/04714310/0471431036.pdf
- [14] What is MySQL? (n.d.). Retrieved July 27th, 2007 from MySQL. Website:
<http://dev.mysql.com/doc/refman/4.1/en/what-is-mysql.html>
- [15] Realex Payments (n.d.). Retrieved July 27th, 2007 from Realex Payments. Website:
<http://www.realexpayments.com/en/homepage.aspx>
- [16] Roger S. Pressman (2005). Software Engineering: A Practitioner's Approach (6th ed.) New York: McGraw-Hill
- [17] Software Engineering Methodology (2003). Retrieved August 1st, 2007 from VILabs.
Website: http://www.vi-labs.com/outsourcing/software_engineering_methodology.html
- [18] JSP Architecture Overview. Retrieved August 1st, 2007 from Mathematics, Statistics and Computer Science, Marquette University. Website:
<http://www.mscs.mu.edu/~schwerm/TechInfo/jsp/jspArch.html>
- [19] Model-View-Controller (MVC) with JavaScript (2006). Retrieved August 1st, 2007 from Alex@Net. Website: <http://www.alexatnet.com/node/8>

[20] Struts and model-view-controller design pattern (2004). Retrieved August 1st, 2007 from IBM Software Information Center. Website:

<http://publib.boulder.ibm.com/infocenter/wsadhelp/v5r1m2/index.jsp?topic=/com.ibm.etools.struts.doc/topics/cstrdoc001.html>

[21] Don Wells (2006). The Rules and Practices of Extreme Programming. Retrieved August 1st, 2007 from Extreme Programming, A Gentle Introduction. Website:

<http://www.extremeprogramming.org/index.html>

[22] Struts Introduction (2007). Retrieved August 2nd, 2007 from The Apache Software Foundation. Website: <http://struts.apache.org/1.x/userGuide/introduction.html>

[23] Java Sun. JDBC Introduction (1995-2007). Retrieved August 3rd, 2007 from The Java Tutorials. Website:

<http://java.sun.com/docs/books/tutorial/jdbc/overview/index.html>

[24] Java Sun. JAAS Overview (n.d.). Retrieved August 3rd, 2007 from The Sun Developer Network, Reference Documentation. Website:

<http://java.sun.com/products/jaas/reference/docs/index.html>

[25] A Little Ant Can Lighten the Load (2007). Retrieved August 3rd, 2007 from Intranet Journal. Website: <http://www.intranetjournal.com/java-ant/>

- [26] Models and Software Frameworks (2007). Retrieved August 3rd, 2007 from CISL Research. Website: <http://www.cisl.ucar.edu/research/models.jsp>
- [27] Robert Englander (1997). Developing JavaBeans. Retrieved August 3rd, 2007 from O'Reilly. Website: <http://www.oreilly.com/catalog/javabeans/chapter/ch01.html>
- [28] Realauth Developer's Guide, Version 3.0. (2007). Realex Payments.
- [29] INTERNET USAGE STATISTICS - The Big Picture (2007). Retrieved August 9th, 2007 from Internet World Stats, Usage and Population. Website: <http://www.internetworldstats.com/stats.htm>
- [30] What is Autism (2006) Retrieved August 5th, 2007 from Irish Autism Action. Website: http://autismireland.ie/1_what_is/index.html
- [31] Java 2 Platform, Enterprise Edition (J2EE) FAQ, 1994-2007. Retrieved August 6th, 2007 from Sun Developer Network. Website: <http://java.sun.com/javaee/overview/faq/j2ee.jsp#j2ee>
- [32] Roger S. Pressman (2005). Software Engineering, A Practitioner's Approach (6th ed.) NY :McGraw-Hill.

- [33] JFreeChart (2005-2007). Retrieved August 6th, 2007 from <http://www.jfree.org/jfreechart/>
- [34] Class ThermometerPlot (n.d.). Org.jfree.chart.plot. Retrieved August 6th, 2007 from <http://www.jfree.org/jfreechart/api/javadoc/org/jfree/chart/plot/ThermometerPlot.html>
- [35] Security Provided Concepts (2007). Retrieved August 6th, 2007 from Developing Security Providers. Website: <http://edocs.bea.com/wles/docs42/dvspisec/concepts.html>
- [36] MyCharity (n.d.) Retrieved August 6th, 2007 from MyCharity. Website: <http://www.mycharity.ie/>
- [37] TinyMCE: Installation (August 2007). Moxicode Systems. Retrieved August 6th, 2007 from Moxicode Documentation Wiki. Website: <http://wiki.moxiecode.com/index.php/TinyMCE:Installation>
- [38] TinyMCE, WYSIWYG Editor (n.d.). Moxicode Systems. Retrieved August 6th, 2007 from Moxicode. Website: <http://tinymce.moxiecode.com/>