BansilalRamnathAgarwal Charitable Trust’s

Vishwakarma Institute of Information Technology

# Pune – 411048.



Project Report

On

**Online ‘C’ Exam System**

##### Submitted By

##### Student Name Roll Number

Pratik Shah 3448

Department of Computer Engineering

Bachelor of Engineering (Computer Engineering)

University of Pune.

Year 2012 – 13

# CERTIFICATE

This is to confirm that the following students of T.E. Computer, Vishwakarma Institute of Information Technology, Pune

**“Rohit Jainani” “Prashant Sethi”**

**“Pratik Shah”**

have successfully completed the mini-project on

##### *“ONLINE ‘C’ Exam system”*

in the partial fulfillment of the requirements for the completion of T.E. in Computer Engineering in 2012-13 as prescribed by the University of Pune.

#### Guide Head of Department

**Prof. S. B. Tatale Prof. S.R.Sakhare**

###### **Principal**

**Mrs. B.S.Karkare**

# **ACKNOWLEDGMENT**

I feel great pleasure in submitting this seminar report on “ONLINE ‘C’ Exam System”**.** I wish to express true sense of gratitude towards my seminar guide, **Prof. S. B. Tatale** who at very discrete step in study of this seminar contributed his valuable guidance and help to solve every problem that arose.

I thank our H.O.D., **Prof. Mr. .S. R. Sakhare** for opening the doors of the department towards the realization of this SDTL Project.

Most likely I would like to express my sincere gratitude towards my family for always being there when I needed them the most. With all respect and gratitude, I would like to thank all the people, who have helped me directly or indirectly. I owe my all success to them.

Pratik Shah

**ABSTRACT**

This is a project implemented in J2EE for online exam system. This project consist of a system and a mechanism to take examinations online.The project consists of web application and a GUI associated with it. This project also consists of a database that holds the questions and the answers associated with it.The project will have all MCQs questions based on various fields and in a stipulated time the students have to answer the question.The project will be implemented by using servlets and concepts of JSP. Even in some cases OOPs concepts will be used namely those of inheritance and polymorphism. This project is created in Netbeans ver. 6.9.1 using JDK 1.5 and Apache Tomcat Server. This project has a future application helping colleges across India to take exams without even their being any paper based question papers and providing students with immediate results thus reducing the Universities time to check papers and reducing human effort.This System is currently being implemented in Pune University for FE exams.

**TABLE OF CONTENTS**

**1. INTRODUCTION**

1.1 Purpose……………………………………………………….

1.2 Project Scope………………………………………………….

1.3 Product Perspective…………………………………………..

1.4 Product Features……………………………………………..

1.5 Assumptions and Dependencies……………………………..

1.6 External Interface Requirements…………………………….

1.6.1 User Interfaces…………………………………..

1.6.2 Software Interfaces………………………………

1.7 Other Nonfunctional Requirements…………………………..

1.7.1 Performance Requirements………………………

1.7.2 Security Requirements …………………………..

1.7.3 Software Quality Attributes……………………...

# **2**.**Software Engineering Methodology adopted**

2.1 User Classes and Characteristics…………………………………

2.2 Operating Environment………………………………………….

2.3 Design Constraints………………………………………………..

2.4 Models………………………………………………………….

2.5 Use Case Diagram……………………………………………..

2.6 Data Flow Diagram…………………………………………….

**3**. **Technology Details**

3.1 Technology Details……………………………………………..

3.2 Architecture…………………………………………………….

* 1. Implementation Constraints……………………………………..

**4. Test Cases**

4.1 Technology Constraints…………………………………………….

4.2 Implementation………………………………………………………

4.3 Servlets………………………………………………………………

4.4 JSP……………………………………………………………………

**5. Simulation Results and Analysis**

**LIST OF DIAGRAMS**

Figure 1.1: Use case diagram…………………………………………………………….

Figure 2.1: Data flow diagram (level 0)…………………………………………………..

Figure 3.1: Data flow diagram (level 1)…………………………………………………..

Figure 3.1: Data flow diagram (level 2)……………………………………………………

Figure 5.1 Screenshot of Project……………………………………………………………

**CHAPTER 1**

**INTRODUCTION**

# **1.1 PURPOSE**

The main aim of the project is to provide students for an easy way to emboss themselves and familiarize themselves with C/C++ programming, from any region with connection to the Internet. It opens up a new perspective in examination taking.

**1.2 PROJECT SCOPE**

The scope of this project is worldwide. Anyone with access to the Internet can access this examination website. This gives a whole new meaning to global knowledge. However the main target for this examination is for Engineering students, or those employees in a company who feel themselves getting rusty in the basics. Our scope is further dividing the examination based on the knowledge level of the examinee. If you are new to C language then the ‘Recruit Level’ helps gaining a basic knowledge of the language. However if you are prepared to challenge the language to its utmost element then the ‘Veteran Level’ is the one for you. This scoping gives a huge range to the number of people this website will be of use to.

**1.3 PROJECT PERSPECTIVE**

This project’s major perspective is to impart the knowledge of the major programming language C. C being the first major language, and having a variety of uses in the real world is a necessary evil to be known in the computing world. And our software aims to provide those currently in, or about to enter the business world with a means to improve or learn this language so as they are not left behind in this fast and all the time expanding computing world.

**1.4 PRODUCT FEATURES**

* Allows Users to take the exam from anywhere in the world.
* Immediate computation of the answers.
* Sends user a Global Certificate which holds much value.
* Completely free to take.
* Allows fast and direct connection to administrators in case of a emergency during the exam.

**1.5 ASSUMPTIONS AND DEPENDENCIE**

Our assumption for this project is that examinee has atleast a basic knowledge of the use of the internet and a basic clue as to what the C Language is. When examinee logs in, an authentication is sent to the followings e-mail for verification of the details and for generating their username and password.

**1.6 EXTERNAL INTERFACE REQUIREMENTS**

**1.6.1 User Interfaces**

When the customer first goes to our website he is welcomed at “Home Page” where some information is provided about our company. User is also provided with options in different tabs like Login, Rules, Centers, About us and Contact us.

To get a view of the cars that are available for the customer, customer first needs to log in if he is already registered customer or create a new account if he is a new customer. Once customer has logged in to the portal he can view the rules and regulations. At the end a conformation notification will be generated that the customer has chosen to take the exam.

**1.6.2 Software Interfaces**

We have connected our portal sysytem to MS Access database using JDBC-ODBC connection driver.

1.7Other Nonfunctional Requirements

1.7.1 Performance Requirements

Examinee’s need to have an internet enabled device with high speed internet access to it on customer side.

**1.7.2 Security Requirements**

State of the art security which is constantly monitored by L-SAT Cambridge, Tata-Docomo who are our sponsors. This helps prevent any cheating or hacking software or hacker to enter and steal the answers beforehand. Also if one such person is caught in the act, his IP address is notified to the OPB and is further banned from appearing in any form of online examinations hence forth.

**CHAPTER 2**

**SOFTWARE ENGINEERING METHODOLGY**

**2.1 User classes and characteristics**

**Use Case Templates**

**1. Use Case for Online Examination Project**

* 1. Use Case ID=RPP-1463
  2. Use Case Name = Online Examination
  3. Actors = Customer
  4. Prerequisites = Site to be opened on the browser.
  5. Steps:-
     1. Specify whether you have an account on the website or not.
     2. If yes login into your account.
     3. If a new user then create an account.
     4. Select the center available to you
     5. Select mode of the examination
     6. Read all rules and regulations
     7. Give the examination within the time limit
     8. Submit the final answers
     9. Receive examination score.

**User Characteristics**

This system will be used by the examinee interested in giving the examinataion. The examinee can perform the following functions:-

1. Browse through the website and review the rules

2. Choose the exam to give.

3. Give the exam.

**3.2 Operating Environment**

|  |  |
| --- | --- |
| Software used | Description |
| Operating  system | We have chosen virtual Windows XP operating system for its best support. |
| JavaServerPages | To implement the project we have chosen JavaServerPagesfor itsability to combine HTML and JAVA code together for dynamic web application. |

**3.3 Design Constraints**

The online examination project must be designed to allow web usability. That is, the product must be designed in such a way that will be easy to use and visible on most of the browsers.

**3.4 Models**

Our Project follows the Waterfall Model.

This model goes through the process of analysis, design, coding, testing and maintenance. The following are the main processes taking place in this model :-

* + 1. Communication: - Communication between examinee and administrator.
    2. Planning:- Examination Questions , timing and processing.
    3. Modeling: - Detail answer correction and data entropy.
    4. Construction :- (i) Coding

(ii) Testing

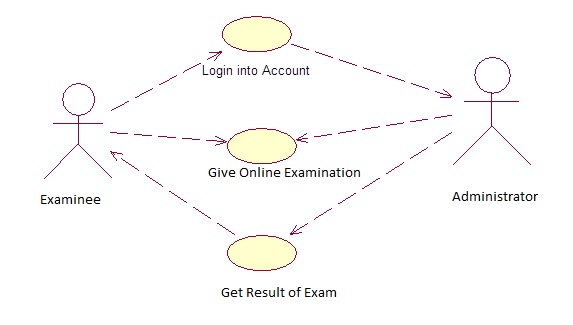
* + 1. Deployment: - Software delivery, support and feedback from sponsors and .testers.

There have been some variations from the typical waterfall model for this project lifecycle.They are:

1. Maintenance has been omitted from the current project.

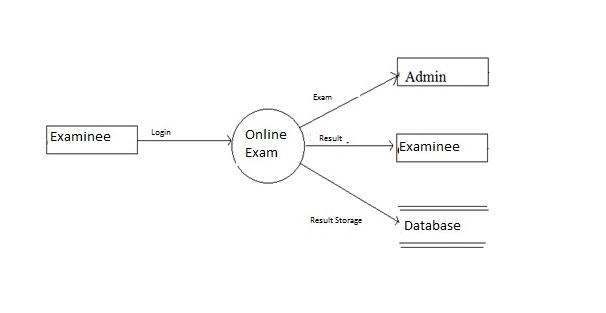
2. Not all testing methods which are present in theoretical model areimplemented.

**3.5 USE CASE DIAGRAM**

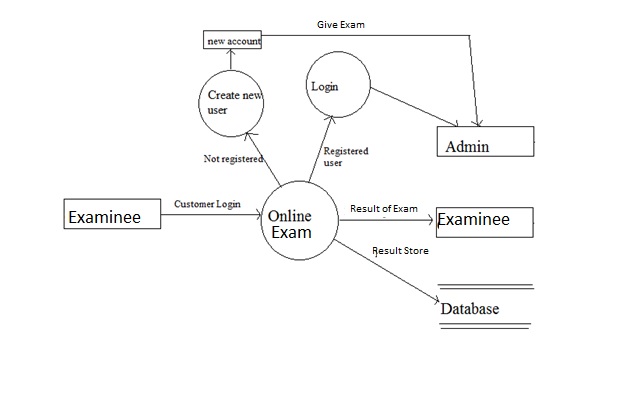
****

**Use case diagram 1**

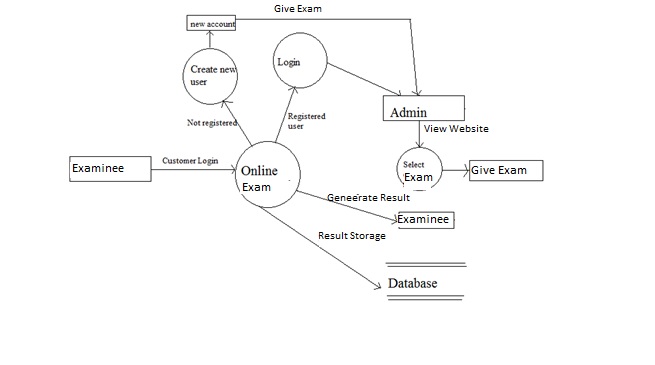
**DATA FLOW DIAGRAM**

**LEVEL 0 DFD :**

**LEVEL 1 DFD:**

****

**LEVEL 2 DFD:**

****

**CHAPTER 4**

**TECHNOLOGY DETAILS AND CODING**

**4.1 Technology Details**

This project is a simple car rental website that is developed in JSP using Eclipse.

1. Front end (HTML)

2. Coding (JAVA)

3.Back-end (Access)Database

**4.2 Implementation Constraints**

User interface is only in English i.e. no other language option isavailable.

Anyone can be the customer and browse through the website.

**4.3 SERVLETS**

A Servlet is a [Java class](http://en.wikipedia.org/wiki/Java_class) in [Java EE](http://en.wikipedia.org/wiki/Java_EE) that conforms to the Java Servlet API, a protocol by which a Java class may respond to [HTTP](http://en.wikipedia.org/wiki/HTTP) requests. They are not tied to a specific client-server protocol, but are most often used with this protocol. The word "Servlet" is often used in the meaning of "HTTP Servlet". Thus, a [software developer](http://en.wikipedia.org/wiki/Software_developer) may use a servlet to add [dynamic content](http://en.wikipedia.org/wiki/Dynamic_web_page) to a [Web server](http://en.wikipedia.org/wiki/Web_server) using the [Java platform](http://en.wikipedia.org/wiki/Java_platform). The generated content is commonly [HTML](http://en.wikipedia.org/wiki/HTML), but may be other data such as [XML](http://en.wikipedia.org/wiki/XML). Servlets are the [Java](http://en.wikipedia.org/wiki/Java_%28software_platform%29) counterpart to non-Java dynamic Web content technologies such as [CGI](http://en.wikipedia.org/wiki/Common_Gateway_Interface) and [ASP.NET](http://en.wikipedia.org/wiki/Active_Server_Pages). Servlets can maintain [state](http://en.wikipedia.org/wiki/State_%28computer_science%29) in [session](http://en.wikipedia.org/wiki/Session_%28computer_science%29) variables across many server transactions by using [HTTP cookies](http://en.wikipedia.org/wiki/HTTP_cookie), or [URL rewriting](http://en.wikipedia.org/wiki/URL_rewriting).

A Web container is essentially the component of a Web server that interacts with the servlets. The Web container is responsible for managing the lifecycle of servlets, mapping a URL to a particular servlet and ensuring that the URL requester has the correct access rights. Java servlets are a key component of server-side Java development. A servlet is a small, pluggable extension to a server that enhances the server's functionality. Servlets allow developers to extend and customize any Java-enabled server--a web server, a mail server, an application server, or any custom server--with a hitherto unknown degree of portability, flexibility, and ease.

A servlet is a Java programming language class used to extend the capabilities of servers that host applications accessed via a request-response programming model. Although servlets can respond to any type of request, they are commonly used to extend the applications hosted by Web servers. For such applications, Java Servlet technology defines HTTP-specific servlet classes. The [javax.servlet](http://java.sun.com/j2ee/tutorial/api/javax/servlet/package-summary.html) and [javax.servlet.http](http://java.sun.com/j2ee/tutorial/api/javax/servlet/http/package-summary.html) packages provide interfaces and classes for writing servlets.The package [javax.servlet.http](http://java.sun.com/javaee/6/docs/api/javax/servlet/http/package-summary.html) defines [HTTP](http://en.wikipedia.org/wiki/HTTP)-specific subclasses of the generic servlet elements, including session management objects that track multiple requests and responses between the Web server and a client. Servlets may be packaged in a [WAR file](http://en.wikipedia.org/wiki/WAR_%28Sun_file_format%29) as a [Web application](http://en.wikipedia.org/wiki/Web_application).

Servlets can be generated automatically from [JavaServer Pages](http://en.wikipedia.org/wiki/JavaServer_Pages) (JSP) by the [JavaServer Pages compiler](http://en.wikipedia.org/wiki/JavaServer_Pages_compiler). The difference between Servlets and JSP is that Servlets typically embed HTML inside Java code, while JSPs embed Java code in HTML.

#### 4.3.1 SERVLET LIFECYCLE

The lifecycle of a servlet begins when it is loaded into Application Server memory and ends when the servlet is terminated or reloaded.

The servlet engine (the WebSphere Application Server - Express function that processes servlets and JSP files) creates an instance of the servlet. The servlet engine creates the servlet configuration object and uses it to pass the servlet initialization parameters to the init() method. The initialization parameters persist until the servlet is destroyed and are applied to all invocations of that servlet.

If the initialization is successful, the servlet is available for service. If the initialization fails, the servlet engine unloads the servlet. The WebSphere Application Server - Express administrator can set a Web application and its servlets to be unavailable for service. In such cases, the Web application and servlet remain unavailable until the administrator changes them to be available.

**4.3.2 Servicing requests**

WebSphere Application Server - Express receives a client request. The servlet engine creates a request object and a response object. The servlet engine invokes the servlet service() method, passing the request and response objects.

The service() method gets information about the request from the request object, processes the request, and uses methods of the response object to create the client response. The service method can invoke other methods to process the request, such as doGet(), doPost(), or methods you write.

The servlet engine stops a servlet by invoking the servlet's destroy() method. Typically, a servlet's destroy() method is invoked when the servlet engine is stopping a Web application which contains the servlet. The destroy() method runs only one time during the lifetime of the servlet and signals the end of the servlet.

After a servlet's destroy() method is invoked, the servlet engine unloads the servlet, and the Java virtual machine eventually performs garbage collection on the memory resources associated with the servlet.

1. Servlets are normal Java classes which are created when needed and destroyed when not needed. Since Servlets run within a Servlet Container, creation and destruction of Servlets is the duty of Servlet Container and not yours.
2. Implementing the init() and destory() methods of Servlet interface allows you to be told by the Servlet Container that when it has created an instance of your Servlet and when it has destroyed that instance.
3. An important point to remember is that your Servlet is not created and destroyed for every request it receives, rather it is created and kept in memory where requests are forwarded to it and your Servlet then generates response.

Advantages of Java Servlets

1. Portability
2. Powerful
3. Efficiency
4. Safety
5. Integration
6. Extensibilty
7. Inexpensive

**4.3.4 Need of Servlets**

Servlets by definition are discrete, reusable applications that run on a server. A servlet does not necessarily have to be tied to one individual application. You could create a library of servlets to be used across multiple applications.

**4.5 JSP**

Java Server Pages is a J2EE technology for building web applications. Java Server Page (JSP) is a technology for controlling the content or appearance of Web pages through the use of Servlet, small programs that are specified in the Web page and run on the Web server to modify the Web page before it is sent to the user who requested it. A JSP page is a textual document that describes how to create a dynamic response to a request .Whereas a Java Server Page calls a Java program that is executed by the Web server, an Active Server Page contains a [script](http://searchenterpriselinux.techtarget.com/sDefinition/0,,sid39_gci212948,00.html) that is [interpreted](http://whatis.techtarget.com/definition/0,,sid9_gci212373,00.html) by a script interpreter (such as [VBScript](http://searchwinit.techtarget.com/sDefinition/0,,sid1_gci213279,00.html) or [JScript](http://searchsoa.techtarget.com/sDefinition/0,,sid26_gci213578,00.html)) before the page is sent to the user.

JSP technology builds on:

1) Template, or static, content

2) Dynamic data

3) Encapsulation of functionality through JavaBeans and tag libraries

## 4.3.5 JSP Page Translation:

A java Servlet file is generated from the JSP source file. This is the first step in its tedious multiple phase life cycles. In the translation phase, the container validates the syntactic correctness of the JSP pages and tag files. The container interprets the standard directives and actions, and the custom actions referencing tag libraries used in the page.

The generated java Servlet file is compiled into a java Servlet class. The translation of a JSP source page into its implementation class can happen at any time between initial deployment of the JSP page into the JSP container and the receipt and processing of a client request for the target JSP page.

## 4.3.6 Class Loading:

The java Servlet class that was compiled from the JSP source is loaded into the container.

In the execution phase the container manages one or more instances of this class in response to requests and other events.  
The interface JspPage contains jspInit() and jspDestroy(). The JSP specification has provided a special interface HttpJspPage for JSP pages serving HTTP requests and this interface contains \_jspService(). jspInit() method is called immediately after the instance was created. It is called only once during JSP life cycle. This method is called for every request of this JSP during its life cycle. This is where it serves the purpose of creation. Oops! it has to pass through all the above steps to reach this phase. It passes the request and the response objects. \_jspService() cannot be overridden.This method is called when this JSP is destroyed. With this call the servlet serves its purpose and submits itself to heaven (garbage collection). This is the end of jsp life cycle.

**4.3.7 What are advantages of using JSP?**

There are plenty advantages of using JSP. In general, JSP allows developers to easily distribute application functionality to a wide range of page authors. These authors do not have to know the Java programming language or know anything about writing servlet code, so they can concentrate on writing their HTML code while you concentrate on creating your objects and application logic.

JSP pages easily combine static templates, including HTML or XML fragments, with code that generates dynamic content.

JSP pages are compiled dynamically into servlets when requested, so page authors can easily make updates to presentation code. JSP pages can also be precompiled if desired.

JSP tags for invoking JavaBeans components manage these components completely, shielding the page author from the complexity of application logic.

Developers can offer customized JSP tag libraries that page author’s access using an XML-like syntax.

Web authors can change and edit the fixed template portions of pages without affecting the application logic. Similarly, developers can make logic changes at the component level without editing the individual pages that use the logic.

**4.3.8 Goals of JSP**

a) Web designers can design and update pages without learning Java programming language

b) Programmers for Java platform can write codes without dealing with web page design

c) JSP allows web designer to write standard HTML pages containing tagsthat run powerful programs based on Java technology.

d) While keeping the benefits of Java Servlet, JSP supports separation of presentation and business logic.

**Benefits of JSP**

1) Platform independent

2) Roles separation

3) Reuse of components and tag libraries

4) Separation of dynamic and static content

5) Encapsulation of functionality

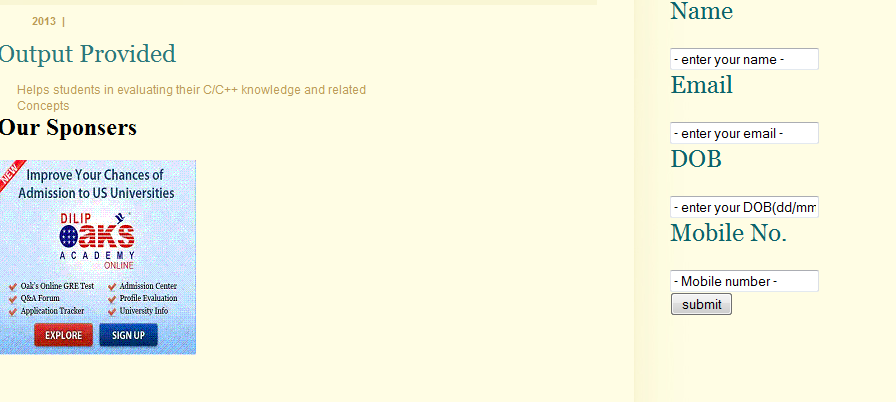
6) Integral parts of J2EE

**CHAPTER 5**

**TESTING**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test case Id** | **Test case name** | **Test data** | **Steps to be executed** | **Tests performed** | **Expected Result** | **Pass/Fail** |
| **1** | Username field (existing user) | Accepts the username of the manger who access database | **-** | **The field should not be blank and user should be registered.** | **After entering the correct username the system will verify it .** | Pass |
| **2** | Password field (  For managers and not students) | Accepts the password | **Valid and registered password should be entered.** | **The field should not be blank and password should be registered.** | **The password is checked and if it is valid then the user is logged onto the system.** | Pass |
| **3** | First name  (new user) | Accepts the first name of user who wants to create an account. | **New user required to enter his/her first name.** | **First name should only contain characters and should not blank.** | **The first name is registered.** | Pass |
| **4** | D.O.B | Accept the date of birth of the student who wants to give the exam. | **New user required to enter his/her date of birth in DD/MM/YYY**  **format** | **This field should only contain D.O.B. and must not be blank** | **The DOB is registered.** | Pass |
| **5** | Email id | Accepts the email id of user who wants to create an account. | **New user required to enter his/her email id.** | **The user must enter a valid email address and not leave the field blank** | **The email id is registered.** | Pass |
| **6** | Mobile number | Accepts the contact number of user who wants to create an account. | **New user required to enter his/her contact number.** | **Contact number field should not be empty. Only numbers required and the length must be ten.** | **The contact number is registered.** | Pass |

**Screenshot of project**

****

**CHAPTER 6**

**SIMULATION RESULTS AND ANALYSIS**

**Results**

This online ‘C’ Exam System has been successfully implemented. The main motive of is to provide students with an examination so that they can test themselves on’C’ knowledge and come to known about the importance of the language

**Analysis:**

This simulation results thus prove that this examintion could be an perfect solution for a online exam system and would help students progress with their knowledge of subject

**CHAPTER 7**

**CONCLUSION AND FUTURE SCOPE**

**Conclusion:**

By developing a website for Online exam sytemservices using JSP, the website can be opened anywhere and on any browser. The development of the website requires programming knowledge in the use of Java/HTML, but equally important is the creativity in designing the GUI. If the developed online portalis deployed in the market, it would be good business in the field of examination thus providing students with much faster and accurate results.

**Future Scope**

This project has huge prospects of further development into a good nationwide website providing exam services throughout India.

**REFERENCES**

* Introduction to java by Khalid mughal
* [www.W3Schools.com](http://www.W3Schools.com)
* Book to ‘C’ by yanshwant kanekar