

# **CODERS' HUB**

**By**

**Shah Yash Paragkumar (110260107013)**

**Guided By**

**Prof. Sandip Chauhan**  
[M.Tech. (CSE) NIT TRICHY]

**Head of Department of Computer Engineering**

**Kalol Institute of Technology & Research Centre  
Gujarat Technological University**

A Project Report Submitted to Gujarat Technological University

In Partial Fulfillment of the Requirements of the Degree of

**Bachelor of Engineering**

**In**

**Computer Engineering**

**2014-15**



**Kalol Institute of Technology & Research Centre**

# CERTIFICATE

*This is to certify that project work embodied in this report entitled “**Coders’ Hub**” was carried out by **Mr. Yash Paragkumar Shah** at **Kalol Institute of Technology & Research Centre** for partial fulfillment of B.E. degree to be awarded by Gujarat Technological University. This project work has been carried out under my supervision and is to the satisfaction of department. The students work has been published/accepted for publication.*

Date:

Place:

Signature and Name of Guide

Signature and Name of Head of Dept.

Signature and Name of  
Principal

Seal of  
Institute



KALOL INSTITUTE OF TECHNOLOGY  
& RESEARCH CENTRE

---

Date: 21/5/2015

## **CERTIFICATE**

*This is to certify that the Project work entitled “**Coders’ Hub**” is a bonafide work carried out by **Shah Yash Paragkumar** (**110260107013**) under my guidance in partial fulfillment for the course of **PROJECT-II** of Computer Engineering (8<sup>th</sup> Semester) in Kalol Institute of Technology & Research Centre of Gujarat Technological University, Ahmadabad during the Academic Year 2014–15.*

**Internal Project Guide**

**Prof. Sandip Chauhan**

**Date of Submission**

**Prof. Sandip Chauhan**

**Head of Department**  
**Computer Engineering**



# GUJARAT TECHNOLOGICAL UNIVERSITY

CERTIFICATE FOR COMPLETION OF ALL ACTIVITIES AT ONLINE PROJECT PORTAL

B.E. SEMESTER VIII, ACADEMIC YEAR 2014-2015

Date of certificate generation : 20 May 2015 (19:56)

This is to certify that, ***Yash Paragkumar Shah*** (*Enrolment Number-110260107013*) working on project entitled with ***Coders' Hub*** from ***Computer Engineering*** department of ***Kalol Institute Of Technology & Research Centre, Kalol*** had submitted following details at online project portal.

Submitted <b>Four</b> Periodic Progress Reports (PPR)	<b>Completed</b>
Business Model Canvas (Image)	<b>Uploaded</b>
Business Model Canvas (Report)	<b>Uploaded</b>
Patent Drafting Exercise (PDE)	<b>Completed</b>
Final Project Report	<b>Uploaded</b>
Plagiarism Search Report	<b>Uploaded</b>

Name of Student : Yash Paragkumar Shah

Name of Guide : Mr. Sandip Kumar  
Sondabhai Chauhan

Signature of Student : \_\_\_\_\_

\*Signature of Guide : \_\_\_\_\_

#### Disclaimer :

This is a computer generated copy and does not indicate that your data has been evaluated. This is the receipt that GTU has received a copy of the data that you have uploaded and submitted as your project work.

\*Guide has to sign the certificate, Only if all above activities has been Completed / Uploaded.

Date: 20.5.2015

Words: 5037

Plagiarised sources: 35

Plagiarised: 1%

<http://www.slideshare.net/jitharadhamesh/free-software-upload-download>

plagiarised from source: >1%

1. No amount of words written here will suffice

2. our sense of gratitude towards all of

[http://www.academia.edu/7838248/Consumer\\_buyingBehaviour\\_towards\\_fertilizers.\\_-By\\_Ram\\_Ahir](http://www.academia.edu/7838248/Consumer_buyingBehaviour_towards_fertilizers._-By_Ram_Ahir)

plagiarised from source: >1%

1. No amount of words written here will suffice

2. our sense of gratitude towards all of

[http://www.gtu.ac.in/ExamCent/Winterexam2013/cent\\_me\\_w2013.pdf](http://www.gtu.ac.in/ExamCent/Winterexam2013/cent_me_w2013.pdf)

plagiarised from source: >1%

1. Kalol Institute Of Technology & Research

2. Kalol Institute Of Technology & Research Centre

<https://www.linkedin.com/groups/Kalol-Institute-Technology-Research-Centre-3681962>

plagiarised from source: >1%

1. Kalol Institute Of Technology & Research

2. Kalol Institute Of Technology & Research Centre

<http://www.minglebox.com/college/kalol-institute-of-technology-research-centre-kalol>

plagiarised from source: >1%

1. Kalol Institute Of Technology & Research

2. Kalol Institute Of Technology & Research Centre

<http://www.kirc-kalol.org/>

plagiarised from source: >1%

1. Kalol Institute Of Technology & Research

2. Kalol Institute Of Technology & Research Centre

<https://www.facebook.com/kitrckalol>

plagiarised from source: >1%

1. Kalol Institute Of Technology & Research

2. Kalol Institute Of Technology & Research Centre

<http://kirc.ac.in/>

plagiarised from source: >1%

1. Kalol Institute Of Technology & Research

2. Kalol Institute Of Technology & Research Centre

<https://www.youtube.com/user/JavaeLearn>

plagiarised from source: >1%

1. easily create Web content that has both static and dynamic

<http://www.youtube.com/watch?v=lRkJEPMp-KU>

plagiarised from source: >1%

1. easily create Web content that has both static and dynamic

<https://glassfish.java.net/javaee5/webtier/webtechnologies.html>

plagiarised from source: >1%

1. easily create Web content that has both static and dynamic

<http://zetcode.com/java/jetty/servlets/>

plagiarised from source: >1%

1. easily create Web content that has both static and dynamic

<https://docs.oracle.com/cd/E19879-01/819-3669/6n5sg7b0v/index.html>

plagiarised from source: >1%

1. easily create Web content that has both static and dynamic

<http://docs.oracle.com/javaee/5/tutorial/doc/bnagx.html>

plagiarised from source: >1%

1. easily create Web content that has both static and dynamic

<http://www.slideshare.net/vinodthebest/vinod-sir-struts-2-part-1>

plagiarised from source: >1%

1. JSP technology projects all the dynamic capabilities of Java Servlet

<http://www.roseindia.net/jsp/jsp-tutorials.shtml>

plagiarised from source: >1%

1. JSP technology projects all the dynamic capabilities of Java Servlet

<http://www.logicweb.com/kb/install-lamp-server-apache-mysql-php-rhel-centos-scientific-linux/>

plagiarised from source: >1%

1. widely used LAMP open source web application software

<http://www.gperi.ac.in/computer-eng.html>

plagiarised from source: >1%

1. Kalol Institute Of Technology & Research Centre

<http://www.gnucoop.com/tech/>

plagiarised from source: >1%

1. widely used LAMP open source web application software

<https://help.asmallorange.com/index.php?/Knowledgebase/Article/View/169>

plagiarised from source: >1%

1. widely used LAMP open source web application software

<http://www.rocketlab.ws/capabilities/web-development/>

plagiarised from source: >1%

1. widely used LAMP open source web application software

<http://www.rocketlab.ws/capabilities/web-applications/>

plagiarised from source: >1%

1. widely used LAMP open source web application software

<https://hub.scaleway.com/mysql.html>

plagiarised from source: >1%

1. widely used LAMP open source web application software

<http://en.wikipedia.org/wiki/MySQL>

plagiarised from source: >1%

1. widely used LAMP open source web application software

<http://www.musicalandroid.com/blog/category/ufxloops>

plagiarised from source: >1%

1. User can download the whole project

<http://about.me/ersagar>

plagiarised from source: >1%

1. Kalol Institute Of Technology & Research

<http://www.slideshare.net/luckpiya/deploying-java-beans-in-jsp>

plagiarised from source: >1%

1. Java Server Pages ... technology allows you

<http://www.oracle.com/technetwork/middleware/reports/reports-technicalwp-128359.pdf>

plagiarised from source: >1%

1. Java Server Pages ... technology allows you

[http://www.roguewave.com/getattachment/f0f4a945-2465-4270-965a-2f3217512090/JWAVE-Architectures?site\\_name=RogueWave](http://www.roguewave.com/getattachment/f0f4a945-2465-4270-965a-2f3217512090/JWAVE-Architectures?site_name=RogueWave)

plagiarised from source: >1%

1. Java Server Pages ... technology allows you

<http://www.mamce.org/ctl/downloads/cse/gnanajeevan.pdf>

plagiarised from source: >1%

1. Java Server Pages ... technology allows you

<http://www.slideshare.net/vishnukuntal/java-project-report-online-banking-system>

plagiarised from source: >1%

1. easily create Web content that has both static and dynamic

<http://kirc.ac.in/kip/>

plagiarised from source: >1%

1. Kalol Institute Of Technology & Research

<http://www.slideshare.net/ssakpi/java-platform-and-ide-netbeans-67-for-developing-enterprise-and-web-applications>

plagiarised from source: >1%

1. easily create Web content that has both static and dynamic

[http://www.gtu.ac.in/ExamCent/Summer\\_Exam\\_2015/cent\\_me\\_s2015.pdf](http://www.gtu.ac.in/ExamCent/Summer_Exam_2015/cent_me_s2015.pdf)

plagiarised from source: >1%

1. Kalol Institute Of Technology & Research Centre

<https://github.com/odoo/docker-docs/tree/master/mysql>

plagiarised from source: >1%

1. widely used LAMP open source web application software

# DECLARATION OF ORIGINALITY

I hereby certify that I am the sole author of this report and that neither any part of this work nor the whole of the work has been submitted for a degree to any other University or Institution.

I certify that, to the best of my knowledge, my work does not infringe upon anyone's copyright nor violate any proprietary rights and that any ideas, techniques, quotations, or any other material from the work of other people included in my report, published or otherwise, are fully acknowledged in accordance with the standard referencing practices. Furthermore, to the extent that I have included copyrighted material that surpasses the bounds of fair dealing within the meaning of the Indian Copyright Act, I certify that I have obtained a written permission from the copyright owner(s) to include such material(s) in my work and have included copies of such copyright clearances to my appendix.

I declare that this is a true copy of my report, including any final revisions, as approved by my supervisor.

Date:

Place:

Shah Yash Paragkumar  
(110260107013)

## ACKNOWLEDGEMENT

Every work that completes successfully stands on the constant encouragement, goodwill and support of the people. Through this, I would like to express my gratitude to all those who have contributed to accomplish my project successfully.

I am highly indebted to **Ms. Bhavika Thakkar** for her guidance and constant supervision as well as for providing necessary information regarding project. And also for his support in completing the project report.

I would like to take this opportunity to thank **Prof. Sandip Chauhan** of Computer Engineering Department, KITRC to provide me such an opportunity to carry out such a great project and for being a constant source of inspiration right from seeking project to the successful completion of the project and also guiding me throughout my project.

My obligation remains to all those people and friends who have directly or indirectly helped me in successful completion of my project. No amount of words written here will suffice for my sense of gratitude towards all of them.

**With Sincere Regards,**  
Yash P. Shah

# TABLE OF CONTENTS

	Page No.
<b>Title Page</b>	
<b>Certificate Page</b>	
<b>College Certificate Page</b>	
<b>PMMS Certificate Page</b>	
<b>Plagiarism Report</b>	
<b>Declaration of Originality Page</b>	
<b>Acknowledgements</b>	
<b>Table of Content .....</b>	<b>i</b>
<b>List of Figures .....</b>	<b>iii</b>
<b>List of Tables .....</b>	<b>vi</b>
<b>List of Abbreviations .....</b>	<b>vii</b>
<b>List of Symbols .....</b>	<b>viii</b>
<b>Abstract .....</b>	<b>xi</b>
<b>CHAPTER-1 INTRODUCTION</b>	<b>1</b>
1.1 Project Summary .....	1
1.2 Goals and Objectives .....	1
1.3 Scope .....	2
1.4 Technologies and Literature Review of Past work/system	4
<b>CHAPTER-2 PROJECT MANAGEMENT</b>	<b>17</b>
2.1 Project Planning and Scheduling.....	17
2.1.1 Project Development Approach and Justification...	17
2.1.2 Project Plan .....	19
2.1.2.1 Milestones.....	19
2.1.2.2 Deliverables.....	20
2.1.2.3 Roles .....	21
2.1.2.4 Responsibilities .....	21
2.1.2.5 Resources.....	21
2.2 Risk Management .....	22
2.2.1 Risk Identification.....	23
2.2.2 Risk Analysis.....	23
2.2.3 Risk Planning .....	24
<b>CHAPTER-3 SYSTEM REQUIREMENT STUDY</b>	<b>26</b>
3.1 User Characteristic .....	26
3.2 Hardware and Software Requirements .....	26
3.3 Constraints .....	27

	<b>Page No.</b>
<b>CHAPTER-4 SYSTEM ANALYSIS</b>	29
4.1 Study of Current System.....	29
4.2 Problems and Weakness of Current System.....	29
4.3 Requirements of New System.....	29
4.3.1 Functional Requirements.....	29
4.3.2 Nonfunctional Requirements.....	30
4.4 Feasibility Study .....	30
4.4.1 Technical Feasibility .....	30
4.4.2 Economical Feasibility.....	31
4.4.3 Operational Feasibility.....	31
4.6 Functions of System .....	32
4.6.1 Use Cases .....	32
4.6.2 Data Flow Diagram .....	33
4.6.2.1 Context Level .....	33
4.6.2.2 First Level: User .....	34
4.6.2.3 First Level: Admin .....	35
4.6.3 Class Diagram .....	36
4.6.4 Activity Diagram.....	37
4.7 Data Model .....	38
4.7.1 ER Diagram .....	38
4.8 Database Design .....	39
4.9 Main Modules of New System .....	42
<b>CHAPTER-5 IMPLEMENTATION DETAIL</b>	43
5.1 Implementation Environment .....	43
5.2 Security Feature .....	43
5.3 Coding Standard .....	43
<b>CHAPTER-6 INPUT AND OUTPUT DESIGN</b>	45
6.1 Admin Layout .....	45
6.2 User Layout .....	51
<b>CHAPTER-7 TESTING</b>	66
7.1 Testing Objectives .....	66
7.2 Testing Model (Testing Strategy) .....	66
7.3 Test cases & Results .....	69
<b>CHAPTER-8 CONCLUSION AND FUTURE WORK</b>	72
8.1 Conclusion .....	72
8.2 Future Work .....	72
<b>REFERENCES &amp; BIBLIOGRAPHY.....</b>	<b>xiii</b>
<b>APPENDIX</b>	
• PPRs	
• PDE Reports	

# LIST OF FIGURES

<b>Figure No.</b>	<b>Description of Figure</b>	<b>Page No.</b>
Figure 1.1	JSP Model 2 Architecture	5
Figure 1.2	Servlet Architecture	6
Figure 1.3	MVC architecture	7
Figure 1.4	Hibernate Architecture	8
Figure 2.1	Agile Development Model	18
Figure 2.2	Time line	22
Figure 4.1	Use case diagram	32
Figure 4.2	DFD-Context Level	33
Figure 4.3	DFD-First Level : User	34
Figure 4.4	DFD-First Level : Admin	35
Figure 4.5	Class Diagram	36
Figure 4.6	Activity Diagram	37
Figure 4.7	ER Diagram	38
Figure 6.1	Login Page	45
Figure 6.2	Admin Dashboard	46
Figure 6.3	Manage User	46
Figure 6.4	Create Package	47
Figure 6.5	Manage Complain	47
Figure 6.6	Manage Feedback	48
Figure 6.7	Create FAQ	48
Figure 6.8	Payment Tracking	49
Figure 6.9	Manage Contact	49
Figure 6.10	Manage Project	50
Figure 6.11	User Login	51
Figure 6.12	Registration	51
Figure 6.13	Forgot Password	52
Figure 6.14	User Dashboard	52
Figure 6.15	Create Project	53

Figure 6.16	Create Project FileIO (1)	53
Figure 6.17	Create Project FileIO (2)	54
Figure 6.18	Create Project FileIO (3)	54
Figure 6.19	Create Project FileIO (4)	55
Figure 6.20	Form Create (1)	55
Figure 6.21	Form Create (2)	56
Figure 6.22	Form Create (3)	56
Figure 6.23	Form Display (1)	57
Figure 6.24	Form Display (2)	57
Figure 6.25	Create Form IO (1)	58
Figure 6.26	Create Form IO (2)	58
Figure 6.27	Create Form IO (3)	59
Figure 6.28	Create Form IO (4)	59
Figure 6.29	Create Form IO (5)	60
Figure 6.30	Generated Project	60
Figure 6.31	Generated JSPs	61
Figure 6.32	Generated Controllers, DAOs, VOs	61
Figure 6.33	Generated Controller (1)	62
Figure 6.34	Generated Controller (2)	62
Figure 6.35	Generated Controller (3)	63
Figure 6.36	Generated DAO (1)	63
Figure 6.37	Generated DAO (2)	64
Figure 6.38	Generated DAO (3)	64
Figure 6.39	Generated VO	65
Figure 6.40	Generated vo.hbm	65
Figure 6.41	Generated hibernate.cfg.xml	66
Figure 6.42	Generated Required_JSP (1)	66
Figure 6.43	Generated Required_JSP (2)	67
Figure 6.44	Generated Display_JSP (1)	67
Figure 6.45	Generated Display_JSP (2)	68
Figure 6.46	Generated Edit_JSP (1)	68
Figure 6.47	Generated Edit_JSP (2)	69
Figure 6.48	Execution of Generated Project (1)	69

Figure 6.49	Execution of Generated Project (2)	70
Figure 6.50	Execution of Generated Project (3)	70
Figure 6.51	Frequently Asked Questions	71
Figure 6.52	Pricing Table	71
Figure 6.53	PayPal Payment	72
Figure 6.54	Contact Us	72
Figure 6.55	Find Us	73
Figure 6.56	Feedback	73
Figure 6.57	Complaint	74
Figure 6.58	User Profile	75
Figure 6.59	Edit User Profile	75
Figure 6.60	404 Error Page	76
Figure 6.61	500 Error Page	76
Figure 7.1	Accepting Testing	81

# LIST OF TABLES

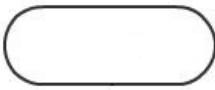
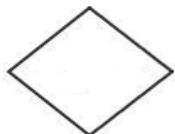
<b>Table No.</b>	<b>Description of Table</b>	<b>Page No.</b>
Table 1.1	HTML	10
Table 1.2	CSS	11
Table 1.3	Java Script	12
Table 1.4	XML	15
Table 2.1	Project Plan	19
Table 2.2	Milestone	20
Table 2.3	Risk Type – Possible Risk	24
Table 2.4	Risk Planning	25
Table 3.1	Hardware Requirements	26
Table 3.2	Software Requirements	27
Table 4.1	Login	39
Table 4.2	User registration	39
Table 4.3	Component List	39
Table 4.4	Project List	39
Table 4.5	Form List	40
Table 4.6	Payment	40
Table 4.7	Image Path	40
Table 4.8	Complain	40
Table 4.9	Contact	41
Table 4.10	Package Information	41
Table 4.11	Frequently Ask Question (FAQ)	41
Table 4.12	Feedback	41
Table 7.1	Test Cases	71

## ABBREVIATIONS

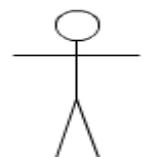
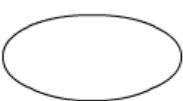
<b>Abbreviated Forms</b>	<b>Expanded Forms</b>
HTML	Hyper Text Markup Language
CSS	Cascading Style Sheets
HTTP	Hypertext Transfer Protocol
JSP	Java Server Pages
JDBC	Java Database Connectivity
API	Application Programming Interface
J2EE	Java Platform, Enterprise Edition
J2SE	Java Platform, Standard Edition
JVM	Java Virtual Machine
JRE	Java Runtime Environment
JDK	Java Development Kit
RMI	Remote Method Invocation
RDBMS	Relational Database Management System
JAR	Java Archive
GUI	Graphical User Interface
PERT	Program Evaluation and Review Technique
IT	Information Technology
SPMP	Software Project Management Plan
IDP	Industry Defined Problem/Project
PPT	PowerPoint Presentation
ISO	International Organization for Standardization

# LIST OF SYMBOLS

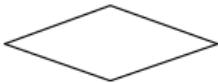
## ❖ Flow Diagram

SYMBOL	DEFINATION
	Input & Output
	Condition
	Module

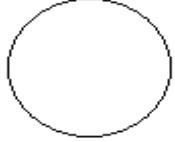
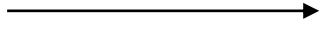
## ❖ Use Case Diagram

SYMBOL	DEFINATION
	Actor
	Use Case

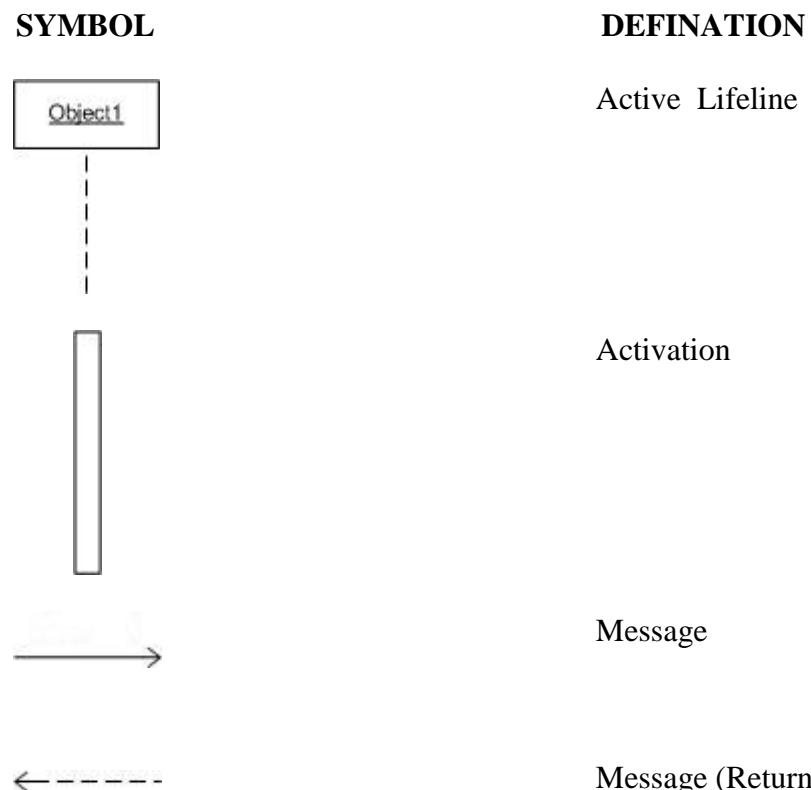
## ❖ ER Diagram

SYMBOL	DEFINITION
	Entity
	Relationship
	Attribute
	Derived Attribute

## ❖ Data Flow Diagram

SYMBOL	DEFINITION
	Input & Output
	Process
	Flow of Data
	Data Storage

## ❖ Sequence Diagram



# ABSTRACT

**Submitted By**

**Shah Yash Paragkumar (110260107013)**

**Supervised By**

**Prof. Sandip Chauhan**  
[M.Tech. (CSE) NIT TRICHY]

Definition of my project is Coders' Hub. 'HUB' means the "center-point", thus, the "CODERS' HUB" is similar to a center-point for every single coder, if they want their project done very efficiently and quickly by just filling up the required FORM. It is aimed towards the developers who have time consuming Projects as well as for those students who wants to make their common Project functionalities done by just filling a form, and invest their saved-time to learn new Technologies, to think of something out of the box, to implement their idea more effectively, etc. Among many advantages that the Internet gives us, one of the crucial one is online coding. Now-a-days web applications are the necessity for all Business area. Some really enjoys it, and others try to keep it as simple and short as possible. Well, online coding is the best solution for both those who consider it a time consuming as well as for those who do it on a daily basis.

The main advantage of online developing is that it allows the Developer to develop the code using online platform without leaving his house. Furthermore, the Internet is available 24\*7, so the Developer doesn't have to hurry or worry about finding a parking spot. Administrators can add new features and functionalities of this web application. It is a Website which provides the CLIENT, which in my case is The Developer, the Website having all the basic Functionalities required by him in-Built.

It consumes LESS TIME because here the developers don't have to write the whole code for the Project and at the end it provides the facility to Directly Download the project, too. After filling up the Required FORM, the developer can download the whole project fulfilling his requirements, to his system & for further use, he can easily edit the project by using JAVA-integrated development environment. i.e., Eclipse. The Main OBJECTIVE of "Coders' Hub" is to minimize the work efforts of the Developer by Generating code which fulfills the BASIC Requirements of the Developer.

## **CHAPTER 1: INTRODUCTION**

### **1.1 Project Summary**

The Online Project Creation web application is intended to provide complete solutions for customers through online platform to develop web application. It will enable developers to develop online web application, purchase them online without having to visit developing company physically. The administration module will enable a system administrator to approve and reject requests for new developer.

### **1.2 Goals and Objectives**

The objective of the project is to achieve the goals listed below which will help user to interact with the system and make maximum benefit out of it.

- Automatic Project creation
- Client can download Project(s)
- Easy Management of client profile.
- Easy to develop project as per the user's requirement
- Easy Up gradation of project
- Easy and accurate tracking of project.
- Reduce Complexity
- Better management of Project
- Better management of Modules of project
- Better management of Components
- Easily understandable software for end user and also for developer
- Customer Satisfaction
- Integration of PayPal
- Easy Payment Tracking
- Obtains high level of security
- Authentic Mailing functionality

### **1.3 Scope:**

- This project provides online development of the web application. The main benefit of this application is used at corporate level or it can be used simply to develop web application.
- It helps the developer who need to develop the web application.
- This web application can be used for the private use as well as for the public use, too.
- In this project, each and every user have the authority to develop the web application and they can also purchase and download the created Project.

**Administrator:**

- Can Login as the Administrator.
- Can Delete/ Block Users.
- Can Manage Projects.
- Can Create Price-Plan Package.
- Can Manage Complain and Feedback.
- Can Create FAQs.
- Can Track the Payment.
- Can Manage Contacts.

**Customer/Developer:**

- Can create Project.
- Can refer to the FAQs, and Help me page.
- Can select appropriate Price Plan.
- Can find the location of the Administrator.
- Can contact the Administrator.
- Can post comments on Feedback, and Complaint page.
- Can Manage his/her Profile.
- Can download the created Web application's Zip file.

**Advantages:**

Online Project Creation's advantages are that it saves the most important thing of any Developer, i.e. Time. The main reason why the Developer should use the Coders'

Hub is that by using this web-portal they can invest their crucial time in some Creative things rather than creating some common code repeatedly.

It has not been an easy road for creating the whole project based on the Developer's requirements. The main advantage of Online Project Creation is that it saves time of the developer and useless efforts of them in making simple code.

### **Easy Project Management**

Firstly, Developers has to submit their requirements by filling up several forms, and based on those requirements, the dynamic project creation takes place. Afterwards the developer can create more components in his/her project as well. Even the Developer can create several new forms inside the project, too.

### **Easy Form Management**

After creating the Project, Several Form can be created as well by the Developer. They can modify the Forms created in a certain project as well. After Updating the Forms, The Project gets modified, too. And at the End the Developer has to download the New Updated Project.

### **Can Easily Download Project**

After the Form Creation, Developer has to submit the Project and Click on the Make Project. By clicking there, The Zip file gets generated. Afterwards the Zip file can be downloaded by the Developer.

### **Easy Payment Tracking**

As, PayPal Integration is used in my Project, the Tracking becomes easier. The Developer has to select the appropriate Price Plan. According the Price Plan, the Developer creates the Projects. So, to Admin the Payment Tracking becomes Easier.

## **1.4 Technologies and Literature Review of Past Work/System**

The technologies used to develop “**Coders’ Hub**” are as follow:

1. JSP 2.0
2. Servlet
3. MVC
4. Hibernate 3.0
5. MySQL 5.0
6. HTML
7. CSS
8. JavaScript
9. Ajax
10. XML
11. Visio
12. Apache Tomcat
13. Eclipse J2EE

### **1) JSP 2.0**

Java Server Pages (JSP) technology allows you to easily create Web content that has both static and dynamic components. JSP technology projects all the dynamic capabilities of Java Servlet technology but provides a more natural approach to creating static content. The main features of JSP technology are

- A language for developing JSP pages, which are text-based documents that describe how to process a request and construct a response
- Constructs for accessing server-side objects
- Mechanisms for defining extensions to the JSP language

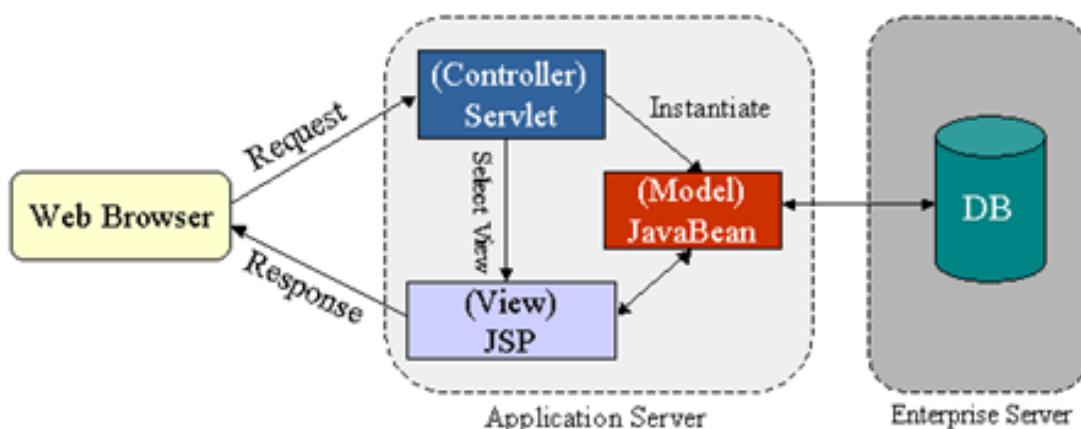
Once the page has been translated and compiled, the JSP page's servlet for the most part follows the Servlet Life Cycle:

1. If an instance of the JSP page's servlet does not exist, the container:
  - a. Loads the JSP page's servlet class
  - b. Instantiates an instance of the servlet class
  - c. Initializes the servlet instance by calling the JSP init method

2. Invokes the JSP service method, passing a request and response object.

### JSP Architecture:

The Model 2 architecture is a hybrid approach for serving dynamic content, since it combines the use of both servlets and JSP. It takes advantage of the predominant strengths of both technologies, using JSP to generate the presentation layer and servlets to perform process-intensive tasks. Here, the servlet acts as the controller and is in charge of the request processing and the creation of any beans or objects used by the JSP, as well as deciding, depending on the user's actions, which JSP page to forward the request to. This approach typically results in the cleanest separation of presentation from content, leading to clear delineation of the roles and responsibilities.



**Figure 1.1: JSP Model 2 Architecture**

### 2) Servlets

Java Servlets are programs that run on a Web or Application server and act as a middle layer between requests coming from a Web browser or other HTTP client and databases or applications on the HTTP server.

Using Servlets, you can collect input from users through web page forms, present records from a database or another source, and create web pages dynamically.

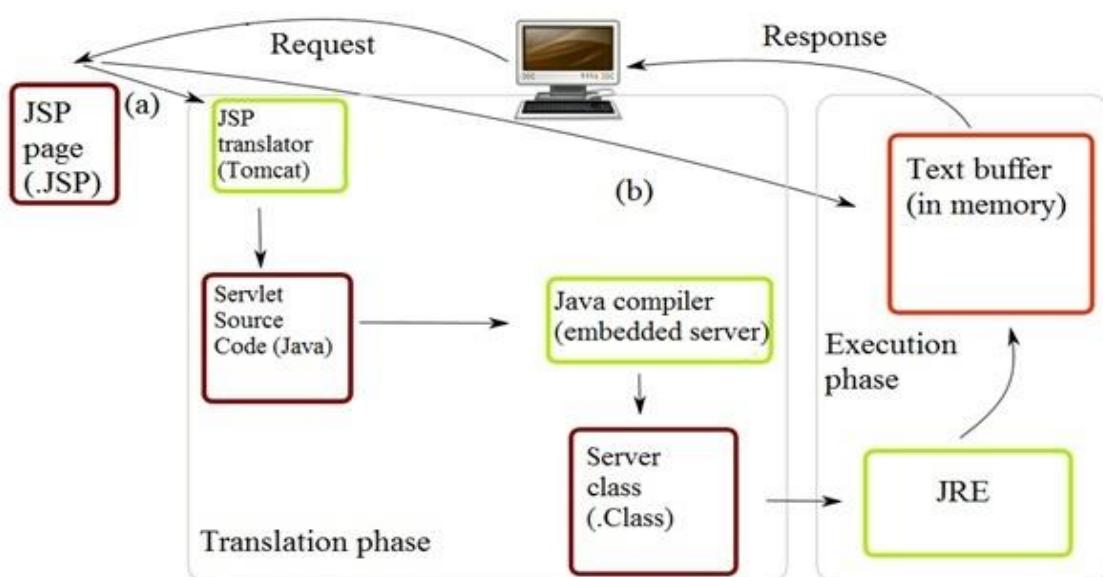
Java Servlets often serve the same purpose as programs implemented using the Common Gateway Interface (CGI). But Servlets offer several advantages in comparison with the CGI.

- Performance is significantly better.

- Servlets execute within the address space of a Web server. It is not necessary to create a separate process to handle each client request.
- Servlets are platform-independent because they are written in Java.
- Java security manager on the server enforces a set of restrictions to protect the resources on a server machine. So servlets are trusted.
- The full functionality of the Java class libraries is available to a servlet. It can communicate with applets, databases, or other software via the sockets and RMI mechanisms that you have seen already.

### Servlet Architecture:

Following diagram shows the position of Servlets in a Web Application.



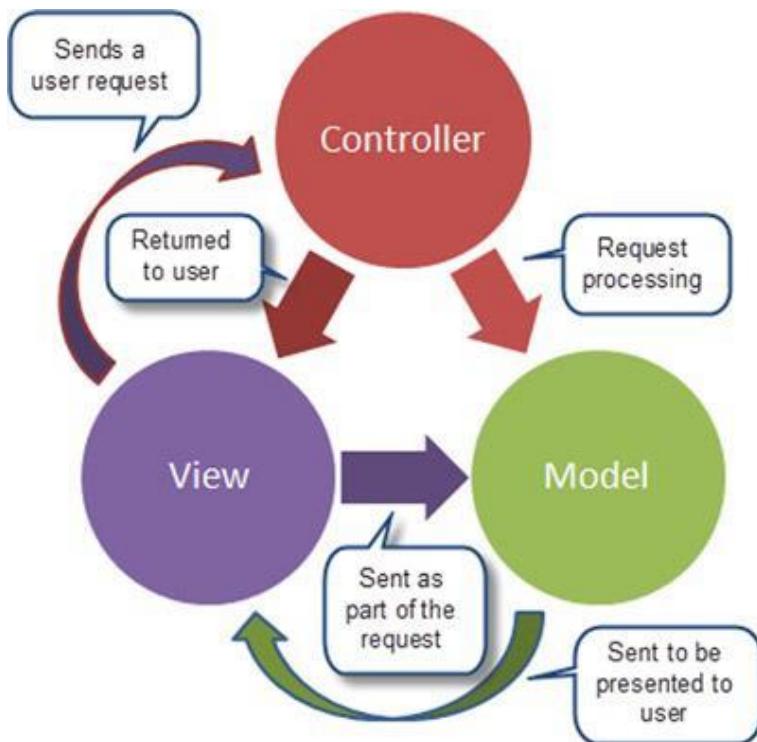
**Figure 1.2: Servlet architecture**

### 3) MVC

Model–view–controller (MVC) is a software architectural pattern for implementing user interfaces. It divides a given software application into three interconnected parts, so as to separate internal representations of information from the ways that information is presented to or accepted from the user.

The central component of MVC, the model, captures the behavior of the application in terms of its problem domain, independent of the user interface. The model directly manages the data, logic and rules of the application. A view can be any output representation of information, such as a chart or a diagram; multiple views of the

same information are possible, such as a bar chart for management and a tabular view for accountants. The third part, the controller, accepts input and converts it to commands for the model or view.



The model–view–controller design defines the interactions between them.

**Figure 1.3: MVC architecture**

A controller can send commands to the model to update the model's state (e.g., editing a document). It can also send commands to its associated view to change the view's presentation of the model (e.g., by scrolling through a document).

A model notifies its associated views and controllers when there has been a change in its state. This notification allows the views to produce updated output, and the controllers to change the available set of commands. In some cases an MVC implementation may instead be 'passive' and other components must poll the model for updates rather than being notified.

A view requests information from the model that it uses to generate an output representation to the user.

#### 4) Hibernate 3.0

Hibernate is an Object-Relational Mapping(ORM) solution for JAVA and it raised as an open source persistent framework created by Gavin King in 2001. It is a powerful, high performance Object-Relational Persistence and Query service for any Java Application.

Hibernate maps Java classes to database tables and from Java data types to SQL data types and relieve the developer from 95% of common data persistence related programming tasks.

Hibernate sits between traditional Java objects and database server to handle all the work in persisting those objects based on the appropriate O/R mechanisms and patterns.



**Figure 1.4: Hibernate architecture**

- **What is Hibernate?**
  - Hibernate is an Object Relational Mapping (ORM) Solution for JAVA.
  - It maps the Java classes to the Database Tables.
  - It is a powerful, high performance object/relational persistence and Query service.
  - Hibernate Provides 3 full featured query facilities :
    - Hibernate Query Language
    - Hibernate Criteria Query
    - Support for Query in native SQL dialect
  - Take less development time
  - Support Automatic key Generation
  - XML Binding

- Has Eclipse support
- **Advantages Over JDBC:**
  - Relational Persistence for JAVA
  - Transparent Persistence
  - Database independent code
  - Support for Query Language
  - Optimize performance with caching
  - Easily scalable

## **5) MySQL 5.0**

MySQL, the most popular Open Source SQL database management system, is developed, distributed, and supported by MySQL.

### **MySQL Features:**

- MySQL is a database management system.
- MySQL is a relational database management system.
- MySQL software is Open Source.
- The MySQL Database Server is very fast, reliable, and easy to use.
- MySQL Server works in client/server or embedded systems.
- A large amount of contributed MySQL software is available.
- It is very fast thread based memory allocation system.
- MySQL written in C and C++ language.
- MySQL code is tested with different compilers.
- Programming libraries for Java, PHP etc. are available to connect MySQL Database.

## **6) HTML**

Hypertext Markup Language (HTML) is the main markup language for displaying web pages and other information that can be displayed in a web browser.

HTML is written in the form of HTML elements consisting of tags enclosed in angle brackets (like <html>), within the web page content. HTML tags most commonly

come in pairs like <h1> and </h1>, although some tags, known as empty elements, are unpaired, for example <img>. The first tag in a pair is the start tag, the second tag is the end tag (they are also called opening tags and closing tags). In between these tags web designers can add text, tags, comments and other types of text-based content.

**Table 1.1 HTML**

<b>Filename extension</b>	.html, .htm
<b>Internet media type</b>	text/html
<b>Type code</b>	TEXT
<b>Developed by</b>	World Wide Web Consortium & WHATWG
<b>Type of format</b>	Markup language
<b>Extended to</b>	XHTML

The purpose of a web browser is to read HTML documents and compose them into visible or audible web pages. The browser does not display the HTML tags, but uses the tags to interpret the content of the page.

HTML elements form the building blocks of all websites. HTML allows images and objects to be embedded and can be used to create interactive forms. It provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. It can embed scripts in languages such as JavaScript which affect the behavior of HTML webpages.

The World Wide Web is composed primarily of HTML documents transmitted from web servers to web browsers using the Hypertext Transfer Protocol (HTTP). However, HTTP is used to serve images, sound, and other content, in addition to HTML. To allow the Web browser to know how to handle each document it receives, other information is transmitted along with the document.

## **7) CSS**

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation semantics (the look and formatting) of a document written in a markup language. Its most common application is to style web pages written

in HTML and XHTML, but the language can also be applied to any kind of XML document, including plain XML, SVG and XUL.

**Table 1.2 CSS**

<b>Filename extension</b>	.css
<b>Internet media type</b>	text/css
<b>Developed by</b>	World Wide Web Consortium
<b>Type of format</b>	Style sheet language

CSS is designed primarily to enable the separation of document content (written in HTML or a similar markup language) from document presentation, including elements such as the layout, colors, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple pages to share formatting, and reduce complexity and repetition in the structural content (such as by allowing for table less web design). CSS can also allow the same markup page to be presented in different styles for different rendering methods, such as on-screen, in print, by voice (when read out by a speech-based browser or screen reader) and on Braille-based, tactile devices. It can also be used to allow the web page to display differently depending on the screen size or device on which it is being viewed. While the author of a document typically links that document to a CSS style sheet, readers can use a different style sheet, perhaps one on their own computer, to override the one the author has specified.

CSS specifies a priority scheme to determine which style rules apply if more than one rule matches against a particular element. In this so-called cascade, priorities or weights are calculated and assigned to rules, so that the results are predictable.

Prior to CSS, nearly all of the presentational attributes of HTML documents were contained within the HTML markup; all font colors, background styles, element alignments, borders and sizes had to be explicitly described, often repeatedly, within the HTML. CSS allows authors to move much of that information to another file, the style sheet, resulting in considerably simpler HTML.

Headings (`h1` elements), sub-headings (`h2`), sub-sub-headings (`h3`), etc., are defined structurally using HTML. In print and on the screen, choice of font, size, color and emphasis for these elements is presentational.

## **8) Java Script**

Java Script is a prototype-based scripting language that is dynamic, weakly typed and has first-class functions. It is a multi-paradigm language, supporting object-oriented, imperative, and functional, programming styles.

**Table 1.3 Java Script**

<b>Filename extension</b>	.js
<b>Developed by</b>	Netscape Communications Corporation, Mozilla Foundation
<b>Type of format</b>	Scripting language

JavaScript was formalized in the ECMA Script language standard and is primarily used in the form of client-side JavaScript, implemented as part of a Web browser in order to create enhanced user interfaces and dynamic websites. This enables programmatic access to computational objects within a host environment.

The most common use of JavaScript is to write functions that are embedded in or included from HTML pages and that interact with the Document Object Model (DOM) of the page. Some simple examples of this usage are:

- Loading new page content or submitting data to the server via AJAX without reloading the page.
- Animation of page elements, fading them in and out, resizing them, moving them, etc.
- Interactive content, for example games, and playing audio and video.
- Validating input values of a web form to make sure that they are acceptable before being submitted to the server.

- Transmitting information about the user's reading habits and browsing activities to various websites. Web pages frequently do this for web analytics, ad tracking, personalization or other purposes.
- Unlike the relationship between VBScript and VB, Java script is not a, subset of Sun's Java language. The two languages share some common syntax, but Netscape developed use Jscript, not Sun. Jscript is a powerful scripting language. Developers commonly use Jscript to write client-server script because it's the common standard for browser scripting, and not all browsers can run VBScript.
- You don't have limit yourself to Jscript on the client; you can use it on the server as well. To use Jscript on the server, you can change the default ASP.NET language to Jscript by making is run at server.
- JavaScript is a compact, object-based scripting language for developing client and server Internet applications. Netscape Navigator 2.0 interprets JavaScript statements embedded directly in an HTML page, and Live wire enables you to create server-based applications.
- In a client application for Navigator, JavaScript statements embedded in an HTML page can recognize and respond to user events such as mouse clicks, form input, and page navigation. For example, you can write a JavaScript function to verify that users enter valid information into a form requesting a telephone number or zip code or Date or Email ID. Without any network transmission, an HTML page with embedded.
- JavaScript can interpret the entered text and alert the user with a message dialog if the input is invalid. Or you can use JavaScript to perform an action (such as play an audio file, execute an applet, or communicate with a plug-in) in response to the user opening or exiting a page.

## **9) AJAX**

Ajax is a group of interrelated web development techniques used on the client-side to create asynchronous web applications. With Ajax, web applications can send data to, and retrieve data from, a server asynchronously (in the background) without interfering with the display and behavior of the existing page. Data can be retrieved using the XMLHttpRequest object. Despite the name, the use of XML is not required (JSON is often used instead), and the requests do not need to be asynchronous.

Ajax is not a single technology, but a group of technologies. HTML and CSS can be used in combination to mark up and style information. The DOM is accessed with JavaScript to dynamically display, and to allow the user to interact with the information presented. JavaScript and the XMLHttpRequest object provide a method for exchanging data asynchronously between browser and server to avoid full page reloads.

## **10) XML**

Extensible Markup Language (XML) is a markup language that defines a set of rules for encoding documents in a format that is both human-readable and machine-readable. It is defined in the XML 1.0 Specification produced by the W3C, and several other related specifications, all gratis open standards.

The design goals of XML emphasize simplicity, generality, and usability over the Internet. It is a textual data format with strong support via Unicode for the languages of the world. Although the design of XML focuses on documents, it is widely used for the representation of arbitrary data structures, for example in web services.

**Table 1.4 XML**

<b>Filename extension</b>	.xml
<b>Internet media type</b>	application/xml, text/xml
<b>Developed by</b>	World Wide Web Consortium
<b>Type of format</b>	Markup language

Many application programming interfaces (APIs) have been developed for software developers to use to process XML data, and several schema systems exist to aid in the definition of XML-based languages.

## **11) VISIO**

Microsoft Visio is a diagramming tool that can be used to visually communicate technical as well as non-technical representation of ideas, processes, concepts, structures, layouts, software models, blue prints etc... Microsoft Visio eliminates the laborious process of creating diagrams by providing the tools to create complex diagrams in a user friendly manner. Whether you want to create an office layout, a basic flowchart or an organizational structure, you can do it with minimal effort using Microsoft Visio. The latest version of Visio 2010 which simplifies complexity with dynamic, data-driven visuals and new ways to share on the web.

## **12) Apache Tomcat**

Apache Tomcat (or Jakarta Tomcat or simply Tomcat) is an open source servlet container developed by the Apache Software Foundation (ASF). Tomcat implements the Java Servlet and the Java Server Pages (JSP) specifications from Sun Microsystems, and provides a "pure Java" HTTP web server environment for Java code to run.

Tomcat should not be confused with the Apache web server, which is a C implementation of an HTTP web server; these two web servers are not bundled together. Apache Tomcat includes tools for configuration and management, but can also be configured by editing XML configuration files.

## **13) ECLIPSE J2EE:**

Eclipse is a toolkit which is designed for the creation of complex projects, providing fully dynamic web application utilizing EJB's. This consists of EJB tools, CMP, data mapping tools & a universal test client that is designed to aid testing of EJB's.

## **CHAPTER 2: PROJECT MANAGEMENT**

### **2.1 Project planning and scheduling**

Effective management of a software project depends on thoroughly planning the process of project. A well planned strategy leads to the best and optimal use of the resources available and ensures completion of project on time.

Project plan sets out the resources available to the project, the work breakdown and a schedule for carrying out work. The project needs a lot of research and thus scheduling was a difficult task as there was a need for carrying out a lot of study about various techniques and testing them at various stages, thus maintaining the schedule was also difficult.

#### **2.1.1 Project Development Approach (Process Paradigm) and Justification**

We have chosen the Agile Development Model for this system development. In any other development model initially developers have to identify requirements fully. If requirement changes then they have to change their methods of development, or if some requirements have to be removed then, after each iteration they can do modification in requirements.

While in Agile model customer can change their requirements any time. While coding is going on, parallel testing is simultaneously going on. Henceforth, the modification in code is easy and the team can work faster.

In software application development, agile software development (ASD) is a methodology for the creative process that anticipates the need for flexibility and applies a level of pragmatism into the delivery of the finished product. Agile software development focuses on keeping code simple, testing often, and delivering functional bits of the application as soon as they're ready.

The goal of ASD is to build upon small client-approved parts as the project progresses, as opposed to delivering one large application at the end of the project.

## Spiral Model

The **spiral model** is a risk-driven process model generator for software projects. Based on the unique risk patterns of a given project, the spiral model guides a team to adopt elements of one or more process models, such as incremental, waterfall, or evolutionary prototyping.

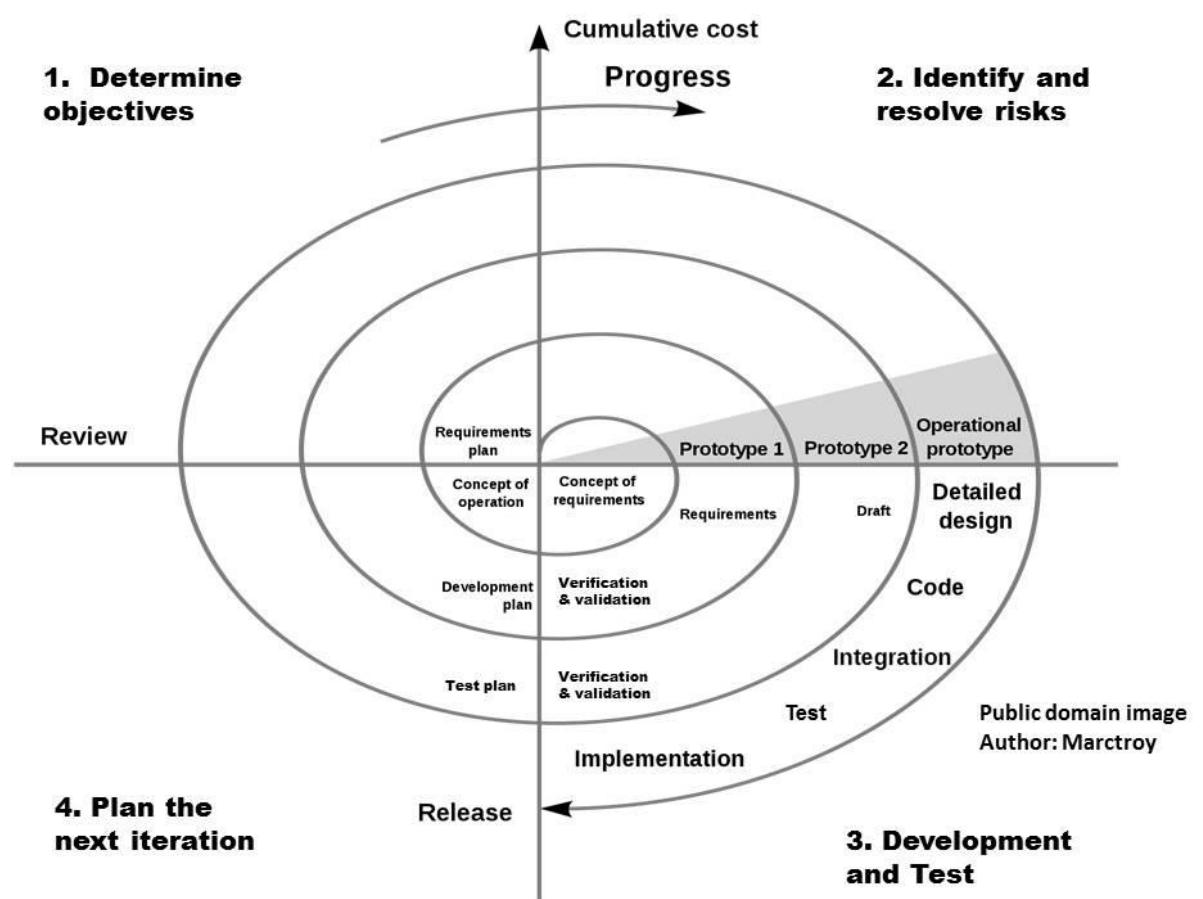


Figure 2.1: Agile Development Model

### 2.1.2 Project Plan

**Table 2.1 Project Plan**

Task Description	Time Taken	Year
Domain Understanding	July	2014
Requirement gathering and analysis	August	2014
Define Objectives	September	2014
System Design	October	2014
Partial documentation	November, December	2014
Implementation	January, February, March	2015
Testing	April	2015
Final Documentation	May	2015

#### 2.1.2.1 Milestones

Management needs information. As project is tangible, this information can only be provided as documents that describe the state of the project being developed without this information it is impossible to judge progress at different phases and therefore schedules cannot be determined or updated. Milestone is an end point of the project process activity.

**Table 2.2 Milestone**

Project process activity	Milestone
Project Plan	Project schedule
Requirement collection	User requirements, System requirements
Design  Data Design  1) User Interface Design	System Design Documents
Implementation  Code for giving security  1) Code for reports	Access rights  Report Generation
Testing	Setting validation and error messages

### 2.1.2.2 Deliverables

At each milestone there should be formal output such as report that can be represented to the management.

- Milestones are the completion of the outputs for each activity.
- Deliverables are the requirements definition and the requirements specification.
- Milestone represents the end of the distinct, logical stage in the project.
- Milestone may be internal project results that are used by the project manager to check progress.
- Deliverables are usually Milestones but reverse need not be true.
- We have divided the software process into activities for the following milestone that should be achieved.

### **2.1.2.3 Roles**

This phase defines the role and responsibilities of each and every member involved in developing the system. To develop this system there was only one member working on the whole application. The democratic arrangement at the senior engineer's level is used to decompose the problem into small parts.

### **2.1.2.4 Responsibilities**

Each member was responsible for each and every part of developing the system. Each of the group members has sufficient knowledge in several programming languages. Each member has to work to the whole with complete unity to end up each activity in time. The task is divided so each member should complete it so that the next member so carry with her tasks smoothly.

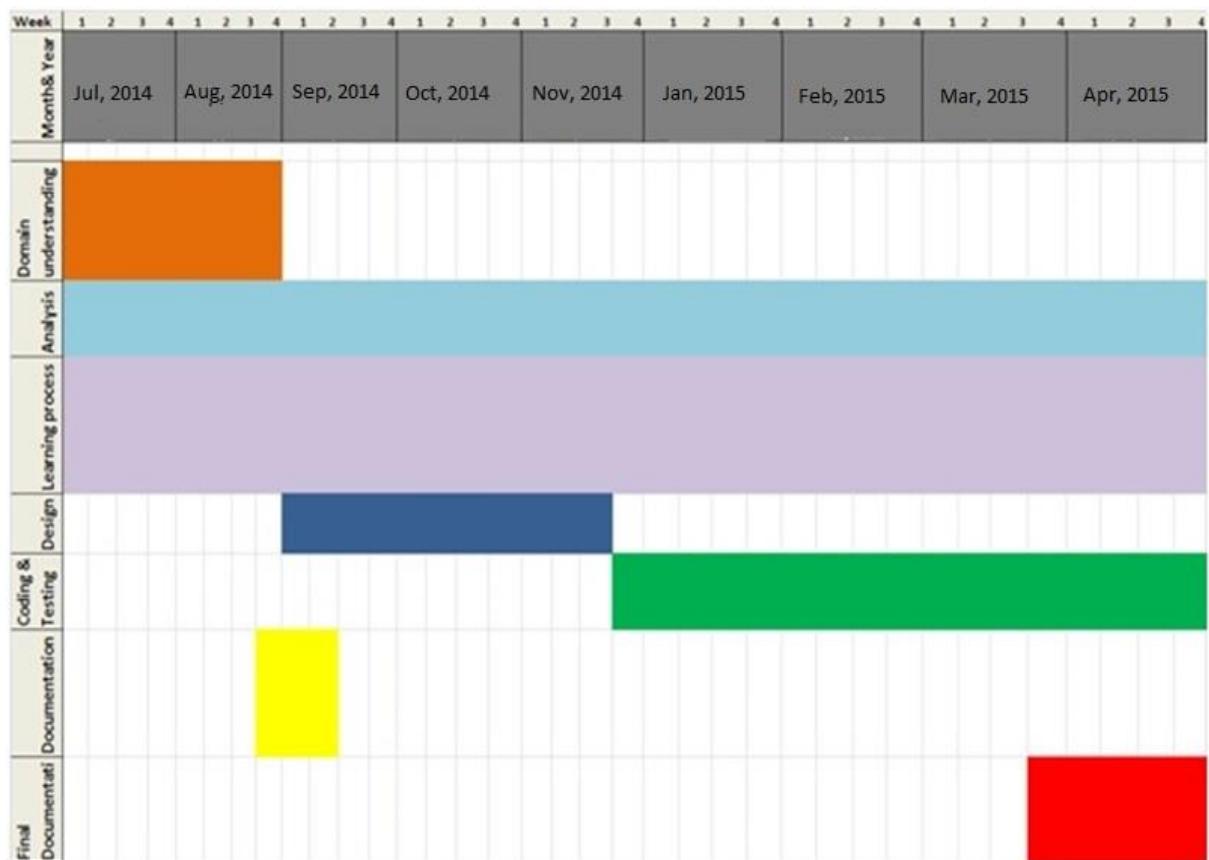
### **2.1.2.5 Resources**

Eclipse Luna is integrated development environment it contains base work space and extensible plugin system for the customer in the environment written in JAVA.

MySQL 5.0 is popular choice of database used in web application and it is central component of the widely used LAMP open source web application software stack.

JDBC, java based database access technology from the oracle corporation. this technology is an API for JAVA programing language that define how client can access the database.

JDK 1.6 java development toolkit is generic and used to develop java based application.it allows language construct like classes and methods.



**Figure 2.2: Time line**

## 2.2 Risk Management

Risk management is an extremely important task in all projects and so it is in this project as well. Software is a difficult undertaking. So lots of things can go wrong. It is for this reason that being prepared understanding the risks and taking proactive measure to avoid or manage them is a key element of good Software Project Management.

Recognizing what can go wrong is the first step called Risk identification. Next each risk is analyzed to determine that it will occur and the damage that it will do if it does occur. Once this information is established, risks are ranked, by probability and impact. Finally a plan is developed to manage those risks with high probability.

### 2.2.1 Risk Identification

Risk is an inevitable concept of software projects and it mainly concerns future happenings. I want to produce a well working system therefore I have to consider all the possible defects and unexpected conditions. First of all I have to define possible risks and develop a management style for each of them. After a deep investigation I identified the following possible risks of our project.

- The tasks that are planned in the schedule may overflow
- Some team members may not focus on the project deeply.
- A team member may leave the group.
- The team members may not be suitable for group work.
- The motivation of team members may not be sufficient.
- The number of people in the project may not be enough for the project.

### 2.2.2 Risk Analysis

For the risk management I select the most catastrophic risks of this list and I developed special several strategies in order to handle or block these unexpected situations. According to our risk table the most effective risks are as show below:

**Table 2.3 Risk Type – Possible Risk**

Risk	Probabilities	Effects
Website components which should be reused contain defects limit their functionality	Moderate	Serious
Change of requirements which require proposal Of major design rework.	High Moderate	Serious Serious
Scheduling slippage:  The time required for the development of website is underestimated, so schedule slippage will occur	High	Serious
The size of website is underestimated	High	Tolerable

Power Failure	High	Tolerable
Inexperienced team member	Medium	Tolerable

### **2.2.3 Risk Planning**

- A risk plan is a list of all risks that threaten the project, along with a plan to mitigate some or all of those risks. Some people say that uncertainty is the enemy of planning. If there were no uncertainty, then every project plan would be accurate and every project would go off without a hitch. Unfortunately, real life intervenes, usually at the most inconvenient times. The risk plan is an insurance policy against uncertainty.
- Into higher the mitigation response is the action plan to eliminate, reduce, or minimize the probability of a risk event occurring and or the impact of the project risk event should it occur. The output from the activity is a Risk Mitigation Plan that contains a set of actions directed at minimizing the potential occurrence or impacts of risks on a project and a Risk Contingency Plan to be activated. For low impact, low probability risks, a mitigation plan may not be prepared, rather these risk items will be monitored to ensure that they do not transpire or evolve risks.

**Table 2.4 Risk Planning**

Risk	Plan
Requirement Changes	Derive traceability information to access requirements, Change impact and maximize information hiding.
Schedule risk	To reduce this risk, we are going to complete our project according to our schedule.
2) performance	Investigate Database which can effectively process.

## **CHAPTER 3: SYSTEM REQUIREMENT**

System requirements study involves a clear and thorough understanding of the product to be developed with the view of removing all ambiguities from customer perception.

### **3.1 User Characteristics**

There are two types of users in this application client and admin. Admin can manage application and database and also account of all the processes done in the system. Client submit their requirement for project such as modules, forms etc. and can get their project and can download that project in framework like hibernate, MVC etc.

### **3.2 Hardware and Software Requirements**

**Table 3.1 Hardware requirements**

<b>Requirements</b>	<b>Description</b>
Processors	Pentium 4 or higher
RAM	256 MB or higher
Monitor	800×600 or larger
Hard disk	40 GB or Above

**Table 3.2 Software requirements:**

Requirements	Description
Platform	Windows XP/ or Higher with MS-office
Database	MySQL 5.0
Tool	Eclipse Luna
Server	Apache tomcat 5.0 or higher

### 3.3 Constraint

- **Hardware constraints:**

I have used HIBERNATE framework to build the application and for that Minimum 512 MB RAM is required with 800MB of Hard Disk Space with minimum 1 GHz processor. This is one time hardware requirement. Because after hosting the site, client does not need extra hardware specification.

- **Design constraints:**

To avoid duplication in the database I have used primary key as well as some times also used unique key. To establish relationship between columns of multiple tables, I have used foreign key constraints.

- **Reliability**

This application must be reliable means it should not be crashed during the execution. I have cared properly to prevent software from getting crashed. I have used unlimited bandwidth to give high performance in the heavy traffic on the application site.

- **Availability**

I have considered all the basic requirements of the users before developing the site. So I have provided most of the features to be useful in the application, like submit requirements, updation in requirements, analysis of requirements, project management.

- **Security**

Security is also a prime requirement of any application. For the security purpose I have used Login authentication before using any feature of the application. I have also encrypted the password of the user in the database to make more secure database.

- **Maintainability**

The application is online when I want to maintain anything I have to put site offline. The maintainability of site will be little bit hard but I have designed the application in a way that it will be easy to maintain the application.

- **Portability**

I have used JAVA, HIBERNATE as a back end language and SERVLET and JSP as front end. For business login, JAVA compiler compiles any of the syntaxes written in JAVA language.

## **CHAPTER 4: SYSTEM ANALYSIS**

### **4.1 Study of Current Systems**

To update any content or any part of site it requires using FTP and changing the content manually.

### **4.2 Problems and Weakness of Current System**

Right now in the current scenario of developing web application, we can see that the basic functionalities are so poor and time consuming. There are many developers in market and too much cheating with price and quality of product is taking place. Sometimes, the user does not get satisfaction with the developer. So, this project is made with the solution to provide online functionality and make the work faster and easier.

### **4.3 Requirements of new System**

Online developing is that it allows people to develop using online platform without leaving their house, to compare the prices of as many customer as they want, and also to develop as many things as they can afford without having to worry about how they will build to them, because the online code generator websites also deliver the product to the buyer's PC. Furthermore, the Internet is open 24 hours a day, 365 days a year, so you don't have to hurry or worry about finding a parking spot. Administrators can add new features and functionalities of this web application.

#### **4.3.1 Functional Requirement**

- User can create account.
- After that user can login and submit their project details like modules, form details.
- Admin can approve request and track payment.
- User can download the whole project
- Easy payment using PayPal integration

- User can also make changes in requirement and can request for updation.
- User will able to give feedback.

#### **4.3.2 Non-Functional Requirements**

- Should work efficiently even on slow internet connections.
- Dynamic contents like success story are expected.
- Secure Database.
- Secure Payments.
- Communication between admin and client.
- Allow requirements changes and can request for updation
- Provide detailed sitemap.
- Provides customer support and feedback facility.
- Customer satisfaction

#### **4.4 Feasibility Study**

A feasibility study is a preliminary study undertaken to determine and document a project's viability. The results of this study are used to make a decision whether to proceed with the project, or table it. If it indeed leads to a project being approved, it will - before the real work of the proposed Project starts - be used to ascertain the likelihood of the project's success. It is an analysis of possible alternative solutions to a problem and a recommendation on the best alternative.

Four types of project feasibility have been considered:

- Technical Feasibility
- Economic Feasibility
- Operational Feasibility

##### **4.4.1. Technical Feasibility:**

The following factors suffice for considering the given project as Technically Feasible.

- The system developed in Java technology which is well known and today we can easily get the technical help of Java technology from the internet.
- The system development in Java technology is specified by client.
- I have used this technology and similar types of tools that can be useful to develop this system.
- The project is required to be implemented using Hibernate 3.0 and Microsoft SQL Server 2008 which are readily available for the development environment.

#### **4.4.2. Economic Feasibility:**

Economic feasibility is very important in development of the software for any company. Because it gives an idea, whether the project going to be developed can be completed at a cost affordable both by the client and developer. The availability of the required hardware and software used to develop our project makes it economically very feasible. Also the company is having all the other required resources needed for the project hence the project is feasible with respect to economy.

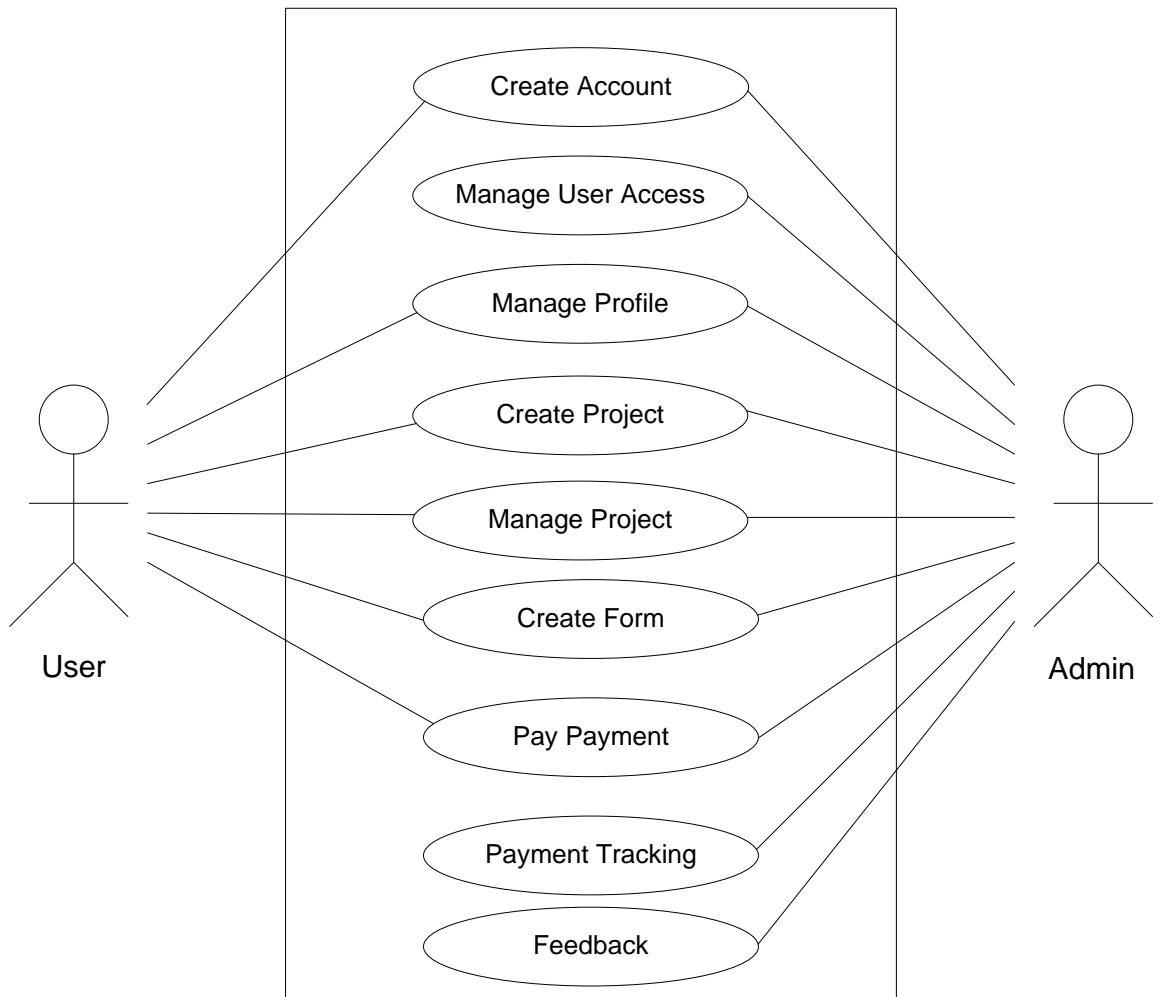
#### **4.4.3 Operational Feasibility:**

Proposed System is beneficial only if they are turned into Information Systems that will meet the organization's operating requirements. This test of feasibility asks if the system will work when it is developed and deployed. Are there any major barriers to implementation? The following factors suffice for considering the given project as operational Feasible.

As the System is going to be developed at the place where it is going to be implemented, the track of the operations related to the software is constantly monitored by them and sufficient support is available.

## 4.6 Function of System

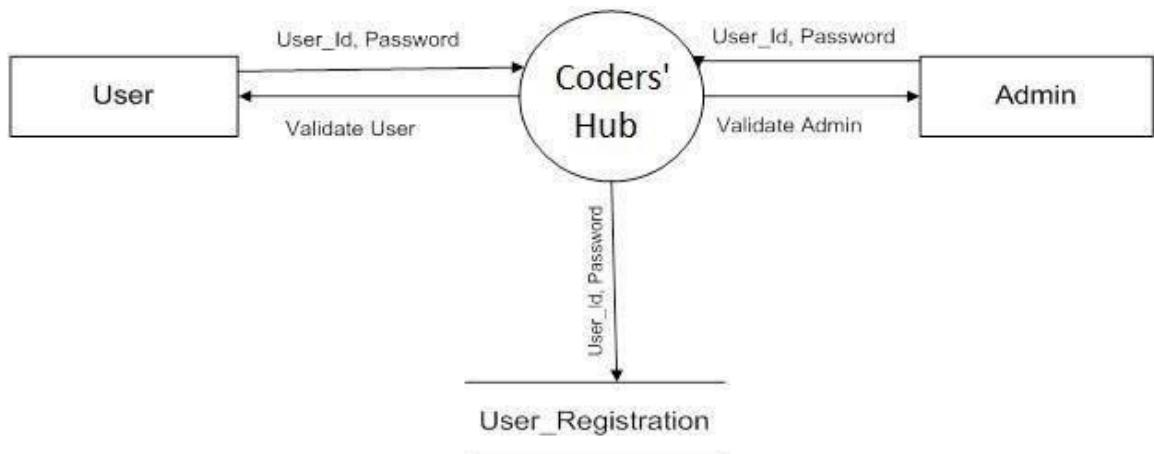
### 4.6.1 Use cases



**Figure 4.1: Use case diagram**

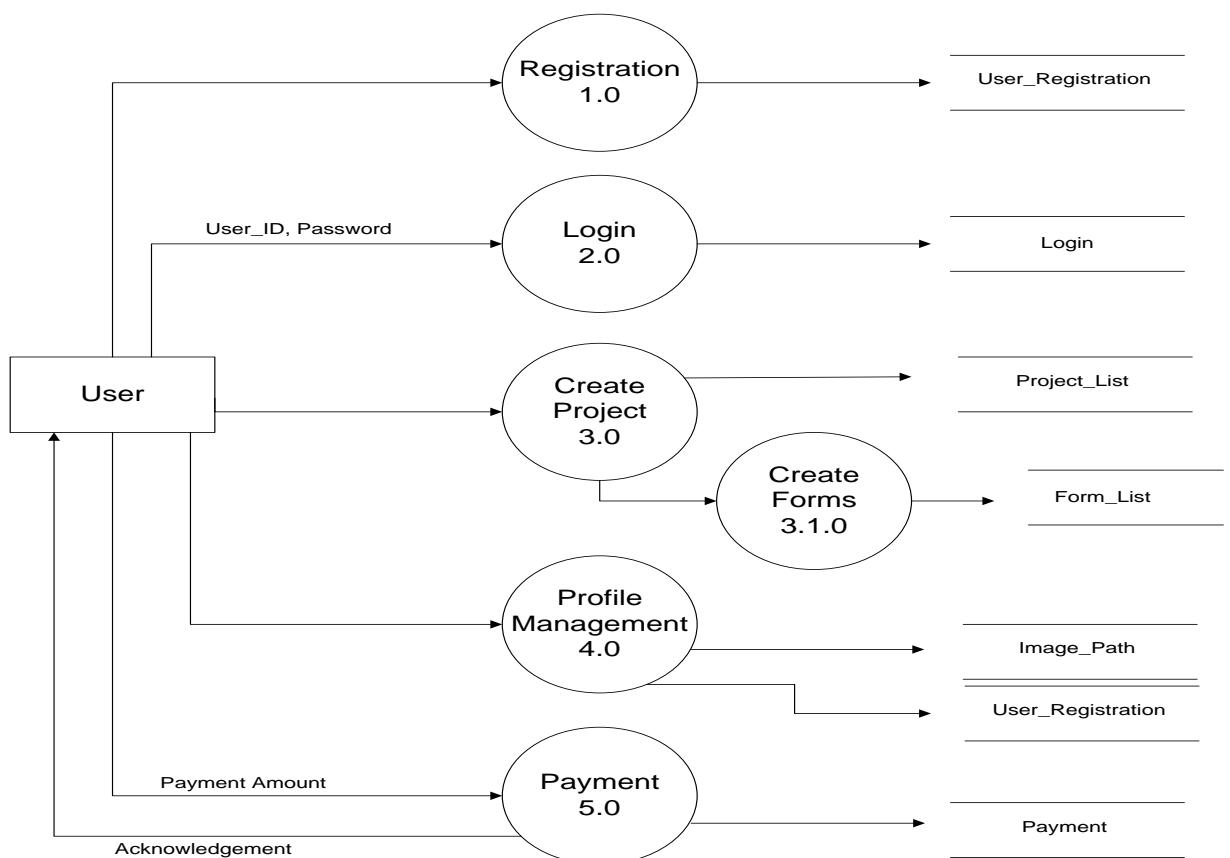
## 4.6.2 Data Flow Diagrams

### 4.6.2.1 Context Level



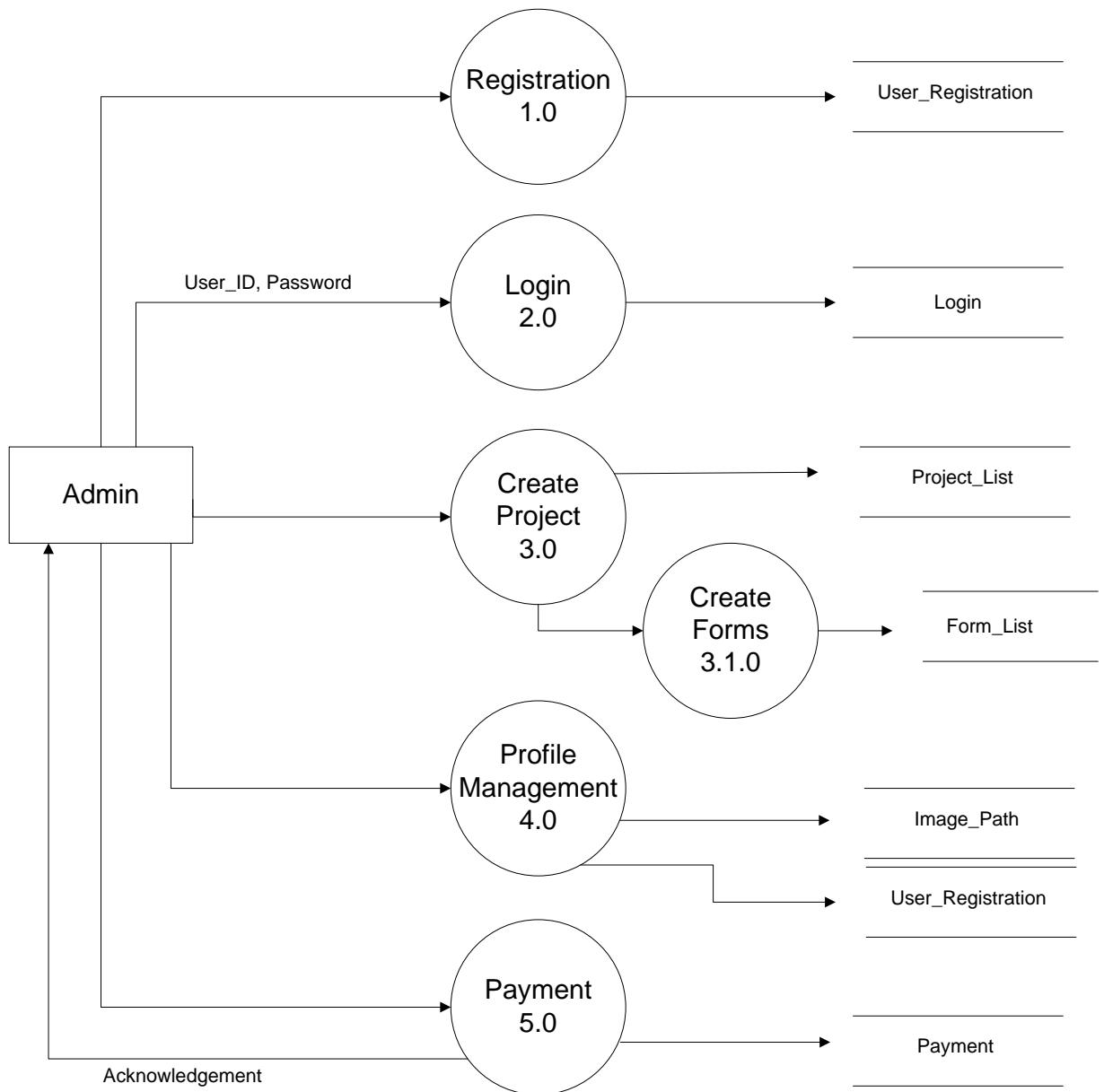
**Figure 4.2: DFD-Context Level**

### 4.6.2.2 First Level: User



**Figure 4.3: DFD-First Level: User**

**4.6.2.3 First Level: Admin**



**Fig. 4.4 DFD: First Level: Admin**

#### 4.6.3 Class Diagram:

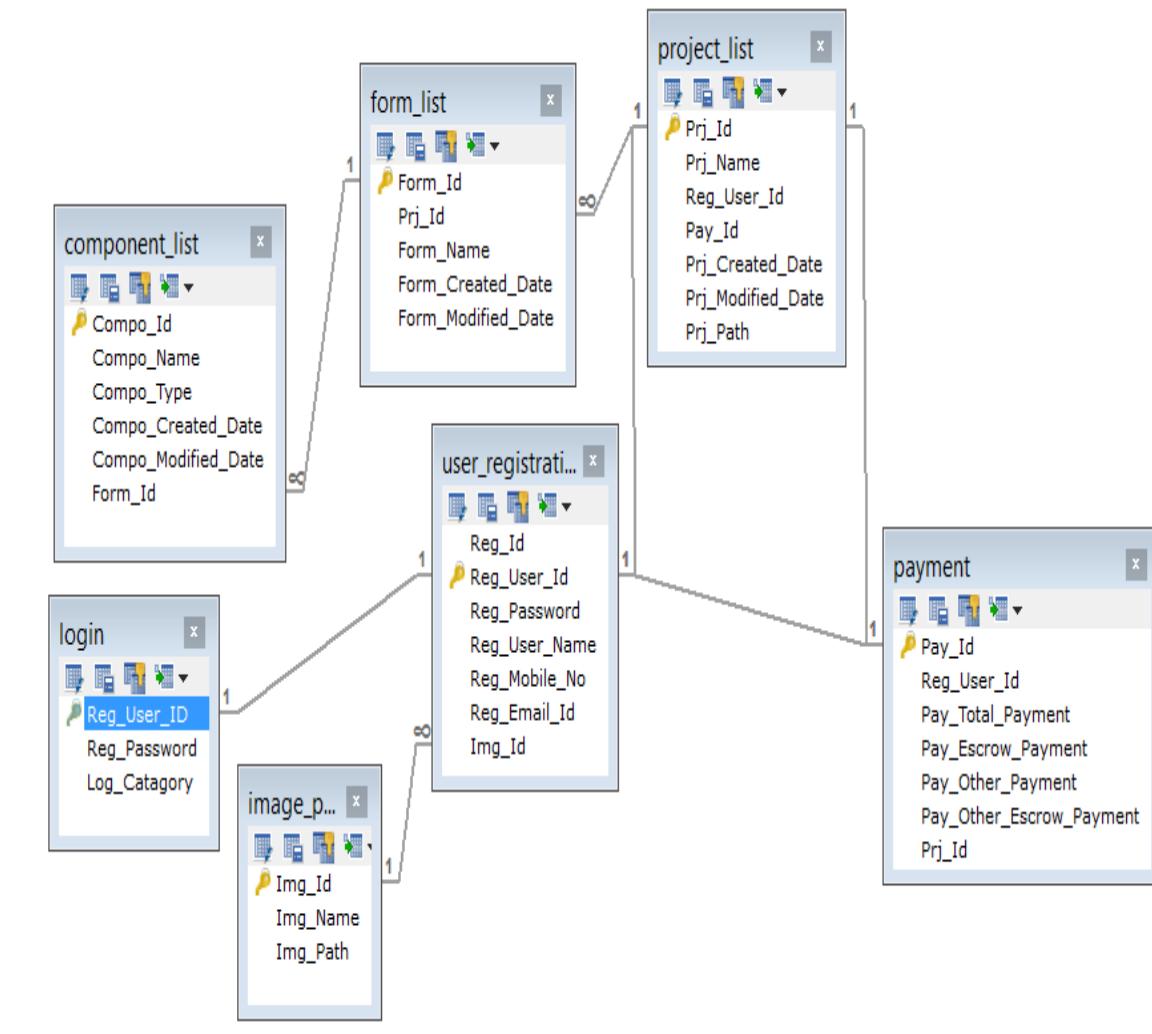


Figure 4.5: Class Diagram

#### 4.6.4 Activity Diagram:

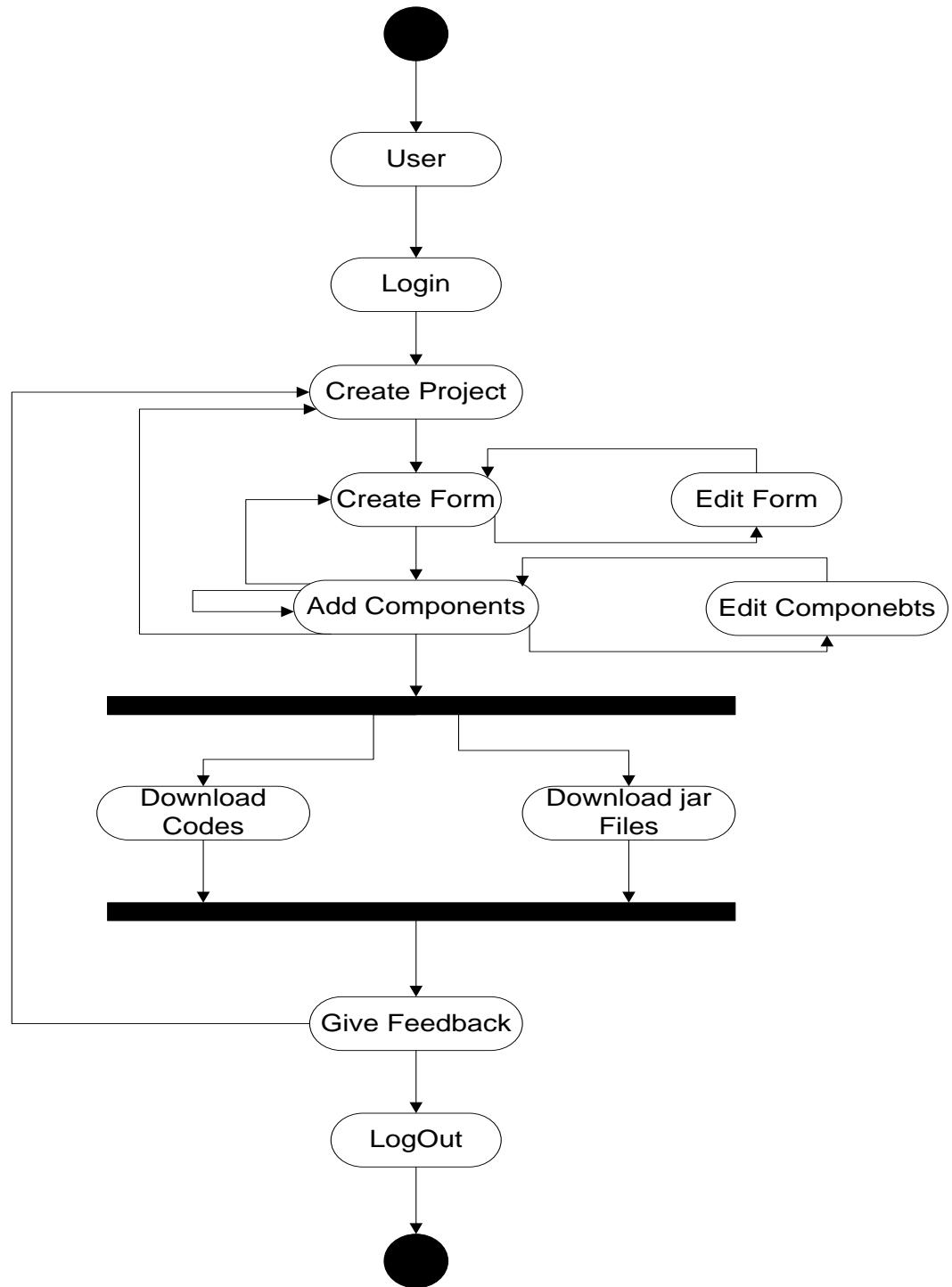


Figure 4.6: Activity Diagram

## 4.7 Data Modeling

### 4.7.1 ER Diagram

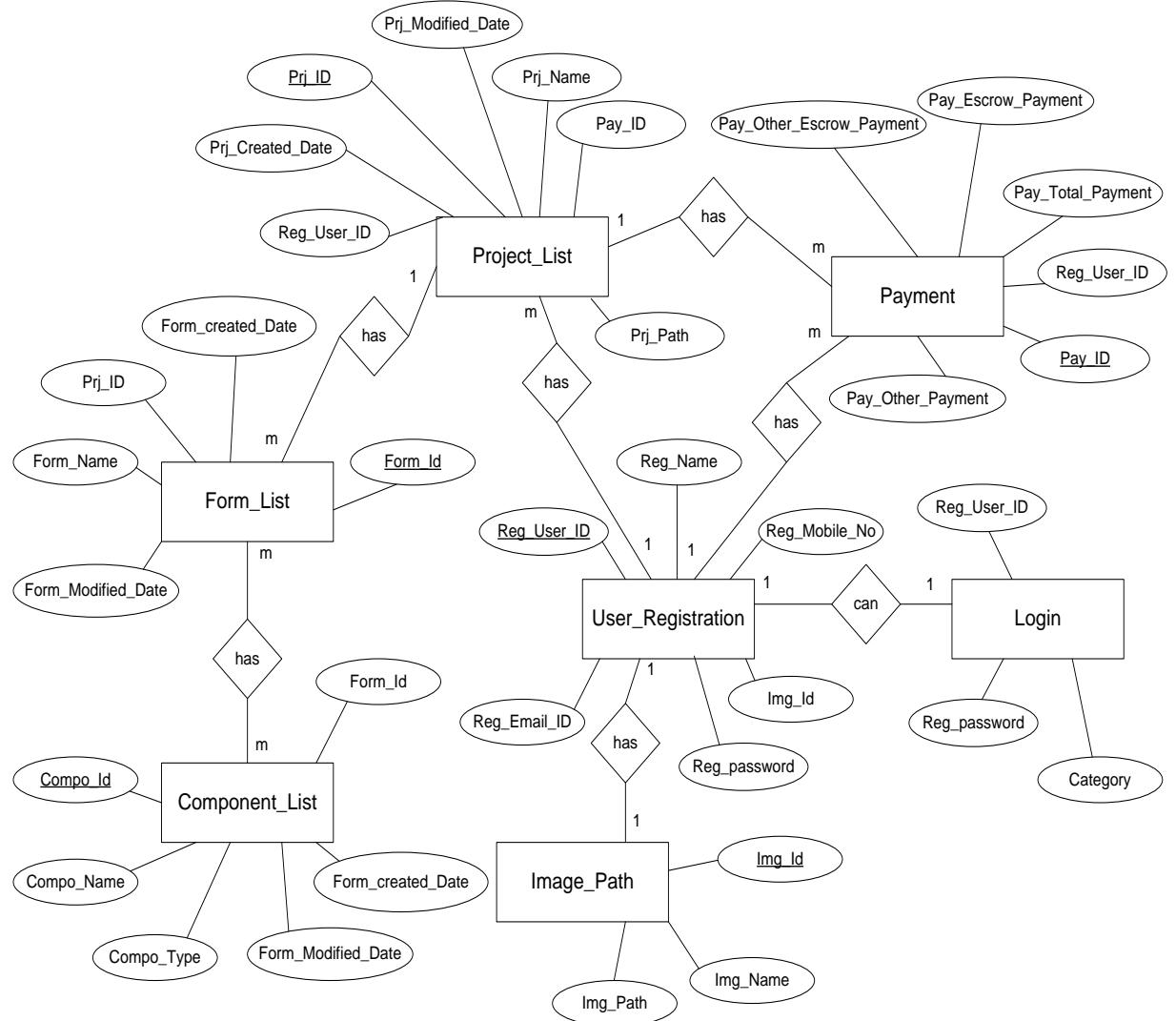


Figure 4.7: ER Diagram

## 4.8 Database Design

### Login

**Table 4.1 login**

Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?	Zerofill?	Comment
Login_ID	int	11		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
UserName	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Password	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
User_type	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Action	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

### User\_Registration

**Table 4.2 User registration**

Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?	Zerofill?	Comment
Reg_ID	int	11		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
FirstName	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
LastName	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Address	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Email	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
City	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Gender	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
login_id	int	11		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

### Component\_List

**Table 4.3 Component List**

Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?	Zerofill?	Comment
Comp_ID	int	11		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
CompLable	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
CompName	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
CompType	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
CompDataType	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
CompCreatedDate	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
CompModifiedDate	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Form_id	int	11		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

### Project\_List

**Table 4.4 Project List**

Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?	Zerofill?	Comment
Project_ID	int	11		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
ProjectName	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
login_id	int	11		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

## Form\_List

**Table 4.5 Form List**

Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?	Zerofill?	Comment
Form_ID	int	11		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
FormName	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Project_id	int	11		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

## Payment

**Table 4.6 Payment**

Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?	Zerofill?	Comment
Pay_Id	int	15		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Reg_User_Id	int	15		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Pay_Total_Payment	int	15		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Pay_Escrow_Payment	int	15		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Pay_other_Payment	int	15		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Pay_other_Escrow_Pay	int	15		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

## Image\_Path

**Table 4.7 Image Path**

Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?	Zerofill?	Comment
id	int	11		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
url	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
filename	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
login_id	int	11		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

## Complain

**Table 4.8 Complain**

Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?	Zerofill?	Comment
Complain_ID	int	11		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Description	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
login_id	int	11		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

## Contact

**Table 4.9 Contact**

Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?	Zerofill?	Comment
contact_id	int	11		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
email	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
name	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
message	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

## Package\_Information

**Table 4.10 Package Information**

Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?	Zerofill?	Comment
Package_ID	int	11		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Package_Name	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Package_Description	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
No_Of_Packages	int	11		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Package_Amount	double			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Package_Type	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

## FAQ

**Table 4.11 Frequently Ask Question (FAQ)**

Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?	Zerofill?	Comment
FAQ_ID	int	11		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
que	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
ans	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

## Feedback

**Table 4.12 Feedback**

Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?	Zerofill?	Comment
Feedback_ID	int	11		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Description	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
login_id	int	11		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

## 4.9 Main Modules of the System

- **Home**

Display the Site map of my application.

- **About us**

Gives the details of the website.

- **Create Project**

Developer can create his/her own project.

- **Create Forms**

Developer can create the forms per project.

- **Payment**

There are two types of Payment activities in our system.

1. Payment Tracking

- The whole tracking of Developer's payment is done by admin.

2. PayPal Integration.

- Developer does the payment for his/her project.

- **Account**

Details of all Developers. Admin can add or remove Developers.

- **Feedback**

Reviews of the Site provided by the Developer.

- **Management**

Admin can manage the users as well as the created projects.

## **CHAPTER 5: IMPLEMENTATION DETAIL**

### **5.1 Implementation Environment**

#### **Single Vs. Multi –User**

The application is multi-user as it is launched on the internet; all users can access it simultaneously, as the service is going to be provided on to the cloud.

#### **GUI Vs. Non-GUI**

As far as the application is concerned, GUI will be provided by the cloud provider.

### **5.2 Security Feature**

Security means protecting the data from accidental or intercepting events. The data and programs must be protected from the threats, theft, fire, disk corruption and any other physical destruction. In this project data is secured by password & many constraints have to be satisfied before the user can access the data.

### **5.3 Coding Standard**

Coding standards are rules that programmers follow to help in ensuring that their Source Code is easy to read & maintain. Habit of using coding standards is a sign of Good Programmer.

#### **Naming Convention**

Use of full English descriptors that accurately describe the variable, functions, subroutines and control names. Use of mixed case to make names readable.

#### **Function Naming**

Function/Subroutine names have a strong active verb as the first word. Function/Subroutine names must begin with a lowercase letter with each subsequent new word in uppercase and subsequent letters in each word in lowercase i.e. Hungarian notation. For example classic Root.

### **Variable Naming**

Have followed Hungarian notation for naming variables. Variable's names starts with lowercase letter then follow Hungarian notation with first three indicating the data type of the variable.

### **Indentation**

If else structure, for Next, While and, Do while constructs have been properly intended.

### **General Code Cleanup:**

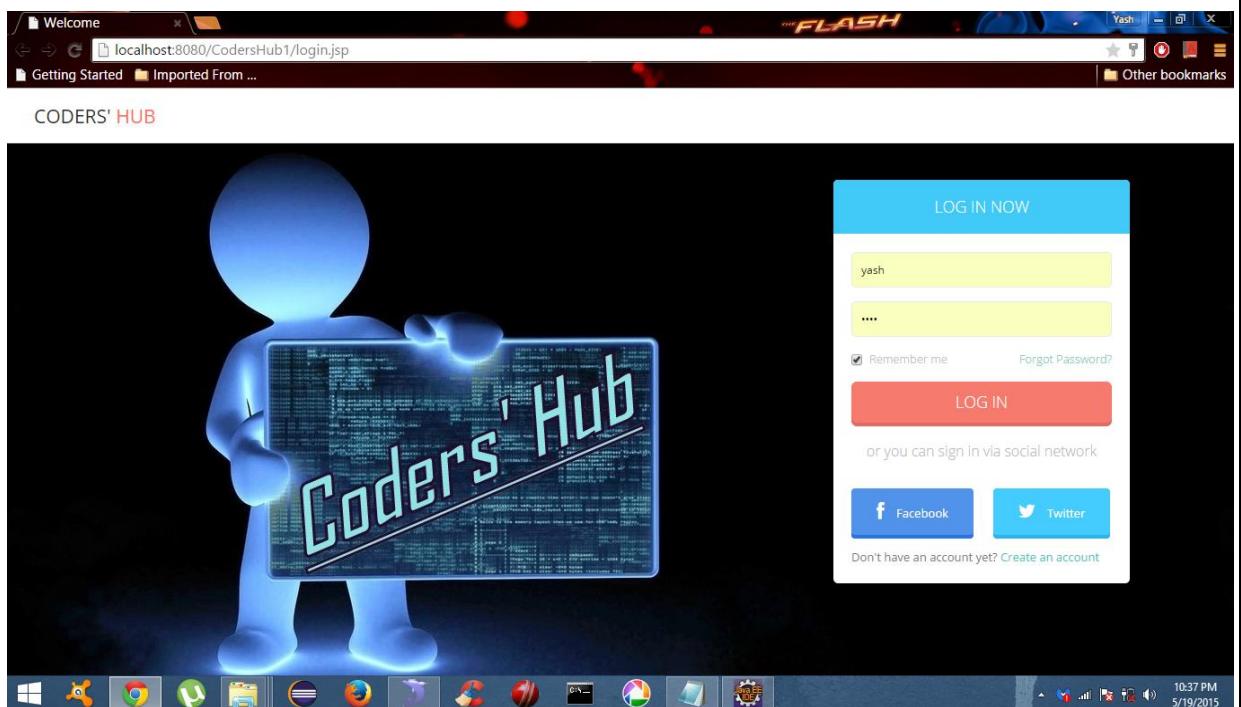
1. Removed unused & disposable code.
2. Removed unused variables.

### **Structuring of code:**

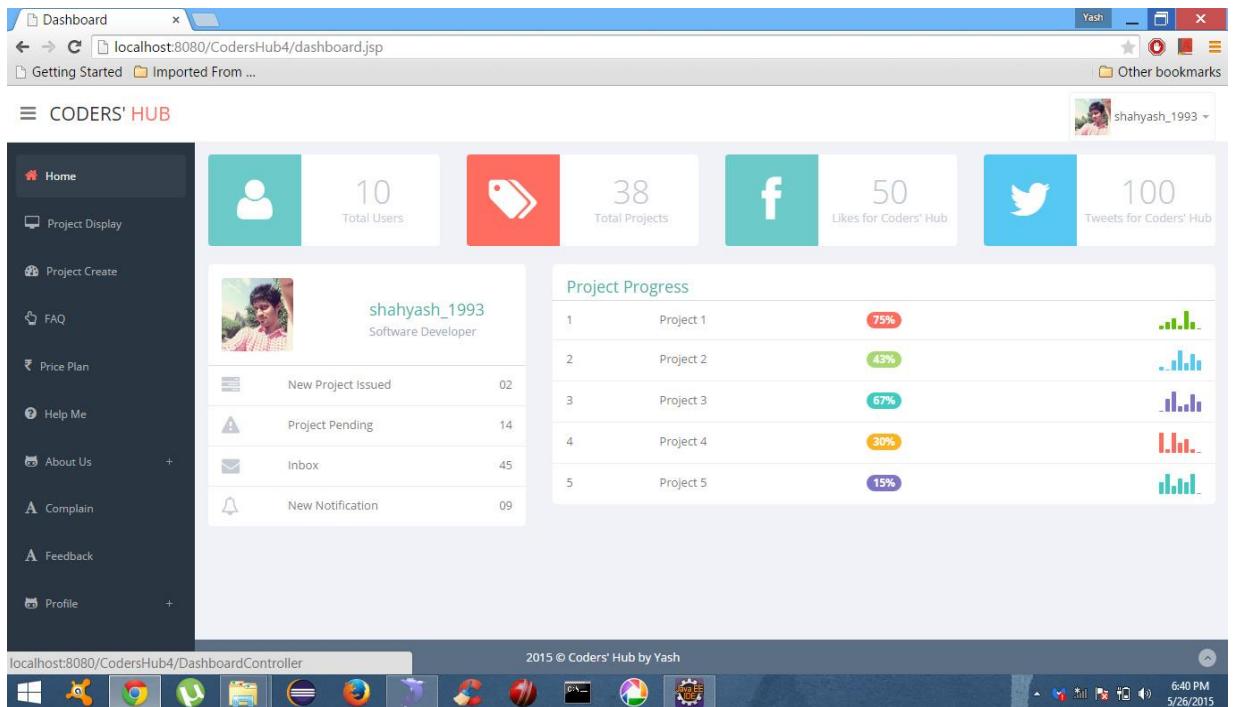
Divided the code into logical entities and have put each entity into a subroutine/function. The Subroutine name reflects the functionality it achieves.

## CHAPTER 6: INPUT AND OUTPUT DESIGN

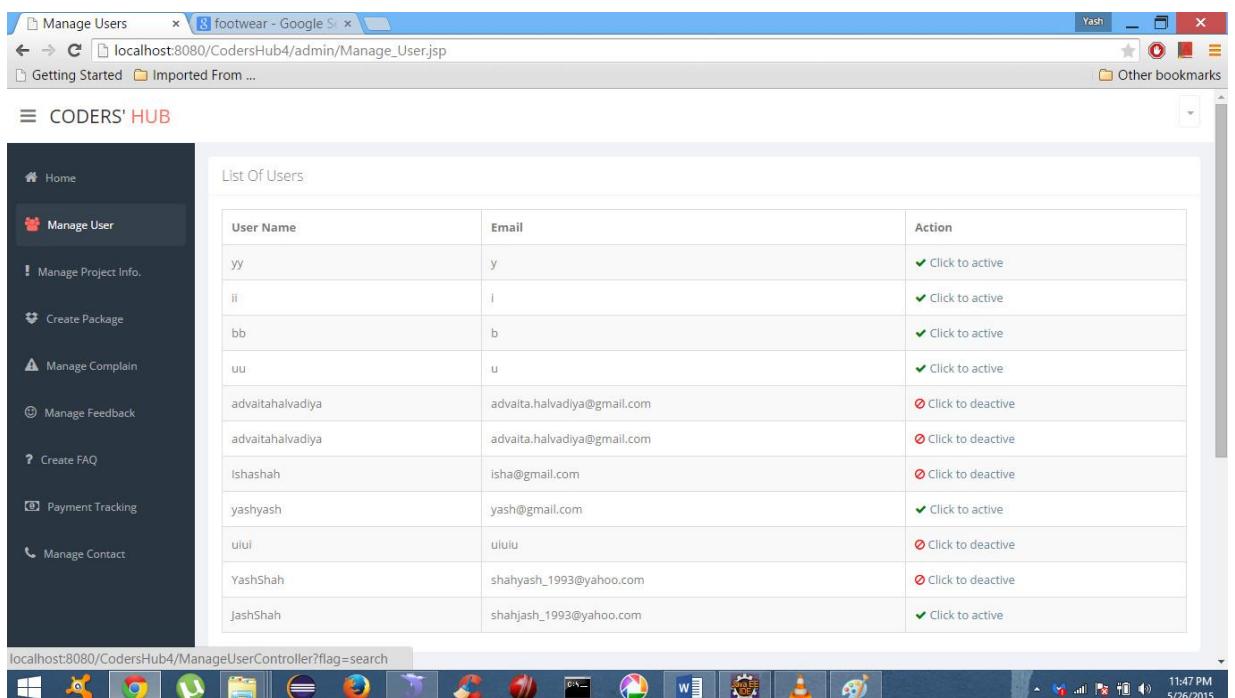
### 6.1 Admin Layout



**Figure 6.1: Login Page**



**Figure 6.2: Admin Dashboard**



**Figure 6.3: Manage User**

The screenshot shows a web application interface titled 'Create Package'. On the left, there is a vertical sidebar with a dark background containing several menu items: Home, Manage User, Manage Project Info., Create Package (which is highlighted in red), Manage Complain, Manage Feedback, Create FAQ, Payment Tracking, and Manage Contact. The main content area has a light gray background and contains the following fields:

- Package Name :
- Package Description :
- Package Amount :
- Number Of Project :
- Package Type:
  - Diamond
  - Platinum
  - Gold
  - Silver

A blue 'Submit' button is located at the bottom right of the form. The browser's address bar shows the URL `localhost:8080/CodersHub4/admin/Create_Package.jsp`. The taskbar at the bottom of the screen displays various icons for other applications like File Explorer, Task Manager, and Media Player.

**Figure 6.4: Create Package**

The screenshot shows a web application interface titled 'Users Complain'. The left sidebar is identical to Figure 6.4, featuring the 'Manage Complain' option as the active item. The main content area is titled 'Complain Of Users' and displays a table of complaints. The table has columns for User\_Id, User\_Name, Description, and Delete. There are five entries listed:

User_Id	User_Name	Description	Delete
14	jash_shah123	desc4	
13	shahyash_1993	desc3	
12	aakash99	desc5	
11	yash_11	desc2	
10	isha	desc1	

Below the table, it says 'Showing 1 to 5 of 5 entries' and includes navigation buttons for 'Previous', 'Next', and page number '1'. The browser's address bar shows the URL `localhost:8080/CodersHub4/admin/Manage_Complain.jsp`. The taskbar at the bottom of the screen displays various icons for other applications.

**Figure 6.5: Manage Complain**

The screenshot shows a web application interface titled "CODERS' HUB". On the left, a sidebar menu lists various administrative functions: Home, Manage User, Manage Project Info., Create Package, Manage Complain, Manage Feedback (which is currently selected), Create FAQ, Payment Tracking, and Manage Contact. The main content area is titled "FeedBack Of Users" and displays a table of feedback entries. The table has columns for "User\_Id", "Description", and "Delete". There are three entries: User\_Id 4 with description "desc3", User\_Id 3 with description "desc2", and User\_Id 1 with description "desc1". Each entry has a "Delete" link next to it. Below the table, a message says "Showing 1 to 3 of 3 entries". At the bottom right of the content area, there are "Previous" and "Next" navigation buttons. The status bar at the bottom of the browser window shows the time as 11:49 PM and the date as 5/26/2015.

**Figure 6.6: Manage Feedback**

The screenshot shows a web application interface titled "CODERS' HUB". On the left, a sidebar menu lists various administrative functions: Home, Manage User, Manage Project Info., Create Package, Manage Complain, Manage Feedback, Create FAQ (which is currently selected), Payment Tracking, and Manage Contact. The main content area is titled "Create FAQ" and contains two input fields: "Question :" and "Answer :". Both fields have placeholder text: "Enter The Question" and "Enter The Answer" respectively. A "Submit" button is located at the bottom right of the form. The status bar at the bottom of the browser window shows the time as 11:51 PM and the date as 5/26/2015.

**Figure 6.7: Create FAQ**

The screenshot shows a web browser window titled "Payment Tracking" with the URL "localhost:8080/CodersHub4/admin/Payment\_Tracking.jsp". The page is titled "CODERS' HUB" and displays a "Payment Record of User Name" table. The table has columns: User ID, User Name, Project ID, Project Name, Payment Done, and Payment Remaining. Two entries are shown:

User ID	User Name	Project ID	Project Name	Payment Done	Payment Remaining
104	Laurel	01	event management	\$70	\$930
101	Tommy	02	cloud management	\$970	\$30

Below the table, it says "Showing 1 to 2 of 2 entries" and there are "Previous" and "Next" buttons. The left sidebar contains links for Home, Manage User, Manage Project Info., Create Package, Manage Complain, Manage Feedback, Create FAQ, Payment Tracking (which is selected), and Manage Contact.

**Figure 6.8: Payment Tracking**

The screenshot shows a web browser window titled "Contact List" with the URL "localhost:8080/CodersHub4/admin/Manage\_Contact.jsp". The page is titled "CODERS' HUB" and displays a "Contact List" table. The table has columns: Contact ID, Name, Email-ID, Message, and Delete. Three entries are shown:

Contact ID	Name	Email-ID	Message	Delete
1	name1	mail1	msg1	✖
2	name2	mail2	msg2	✖
3	name3	mail3	msg3	✖

Below the table, it says "Showing 1 to 3 of 3 entries" and there are "Previous" and "Next" buttons. The left sidebar contains links for Home, Manage User, Manage Project Info., Create Package, Manage Complain, Manage Feedback, Create FAQ, Payment Tracking, and Manage Contact (which is selected).

**Figure 6.9: Manage Contact**

Project Name	user name	View forms	Delete
proj1	y	View Forms	
iNoProject	i	View Forms	
e-commerce	u	View Forms	
dekhBhai	u	View Forms	
uareawesome	u	View Forms	
login	u	View Forms	
lloogginn	u	View Forms	
lloogginn	u	View Forms	
logoon	u	View Forms	
polic	u	View Forms	
oilmaa	u	View Forms	

**Figure 6.10: Manage Project**

## 6.2 User Layout

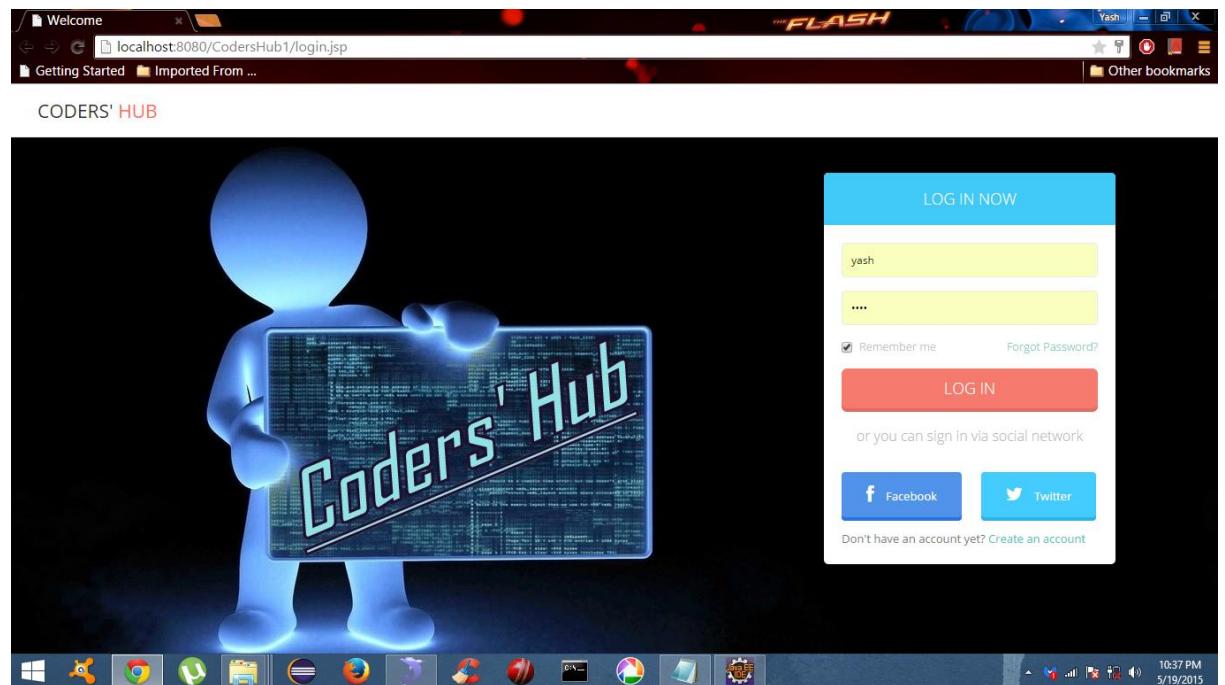


Figure 6.11: User Login

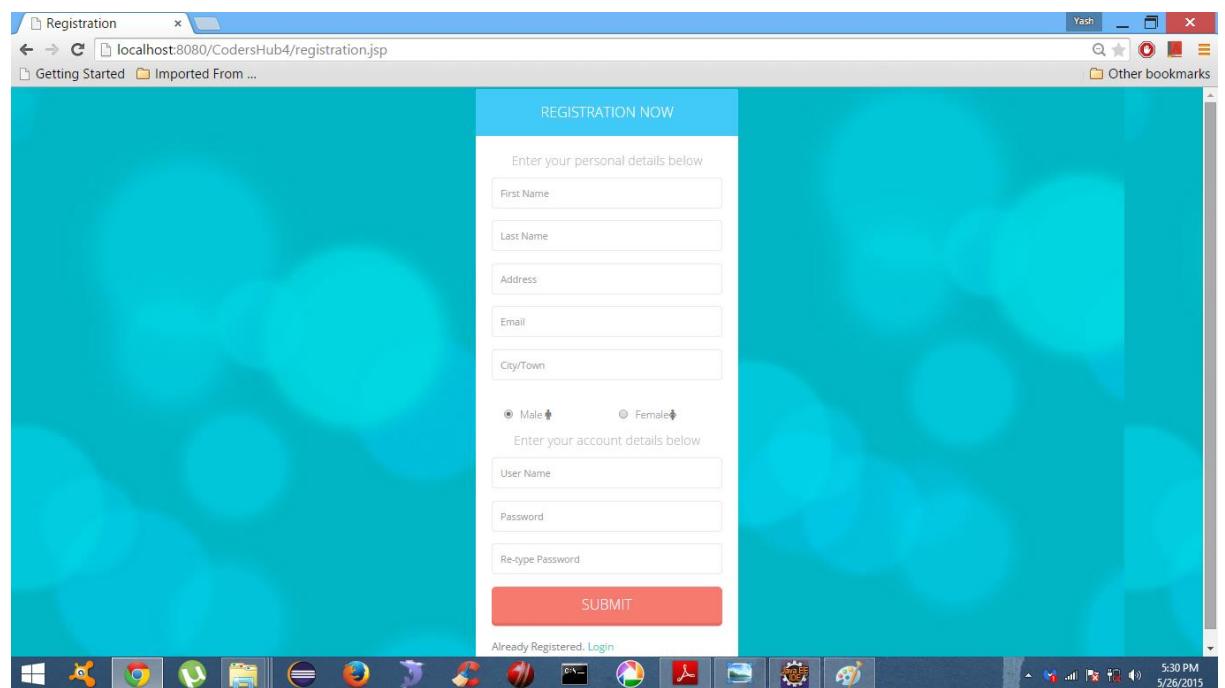
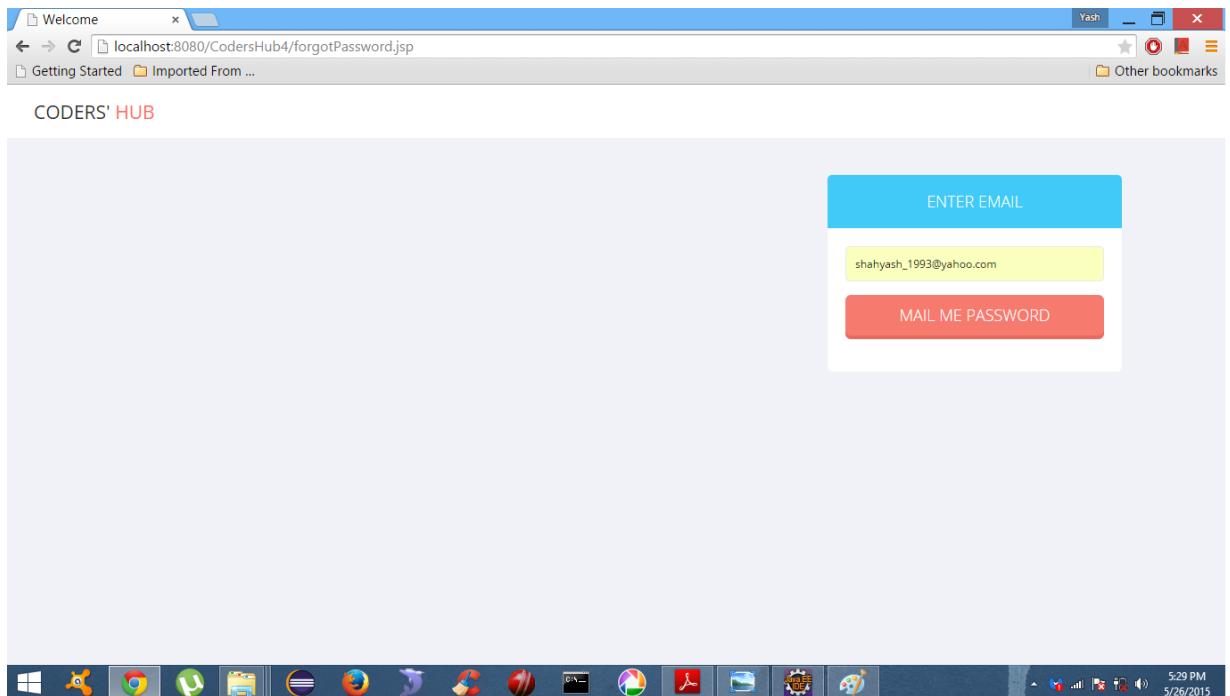
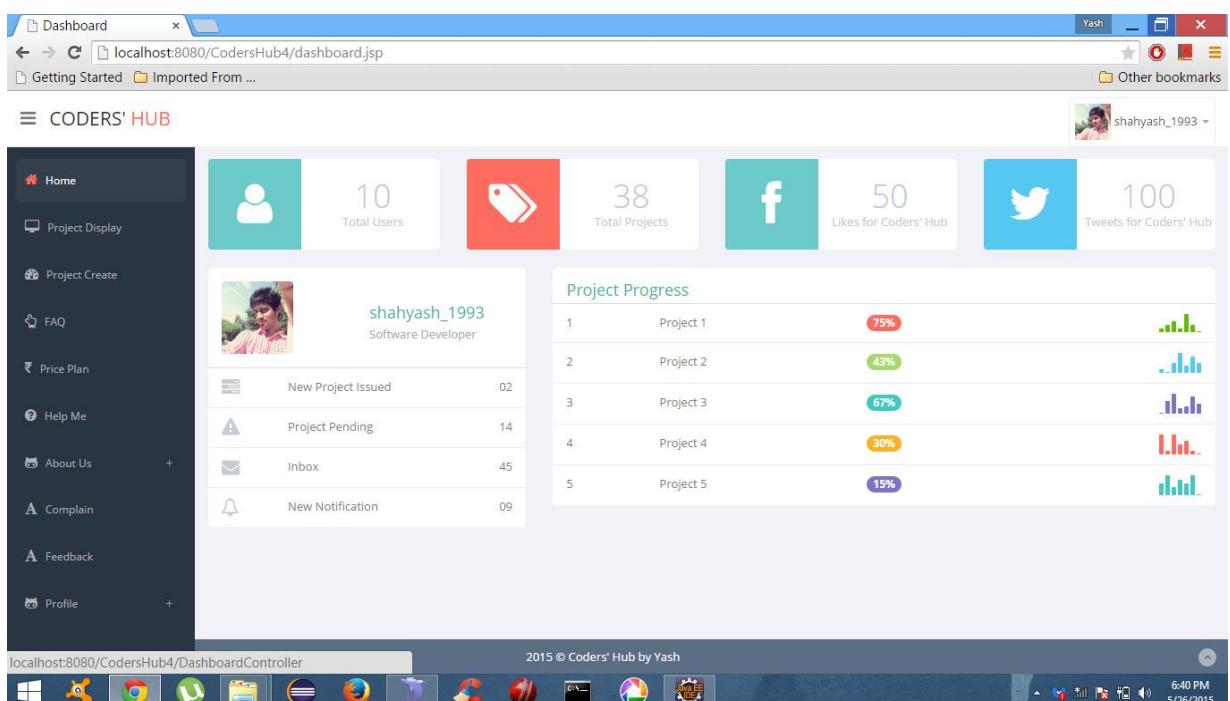


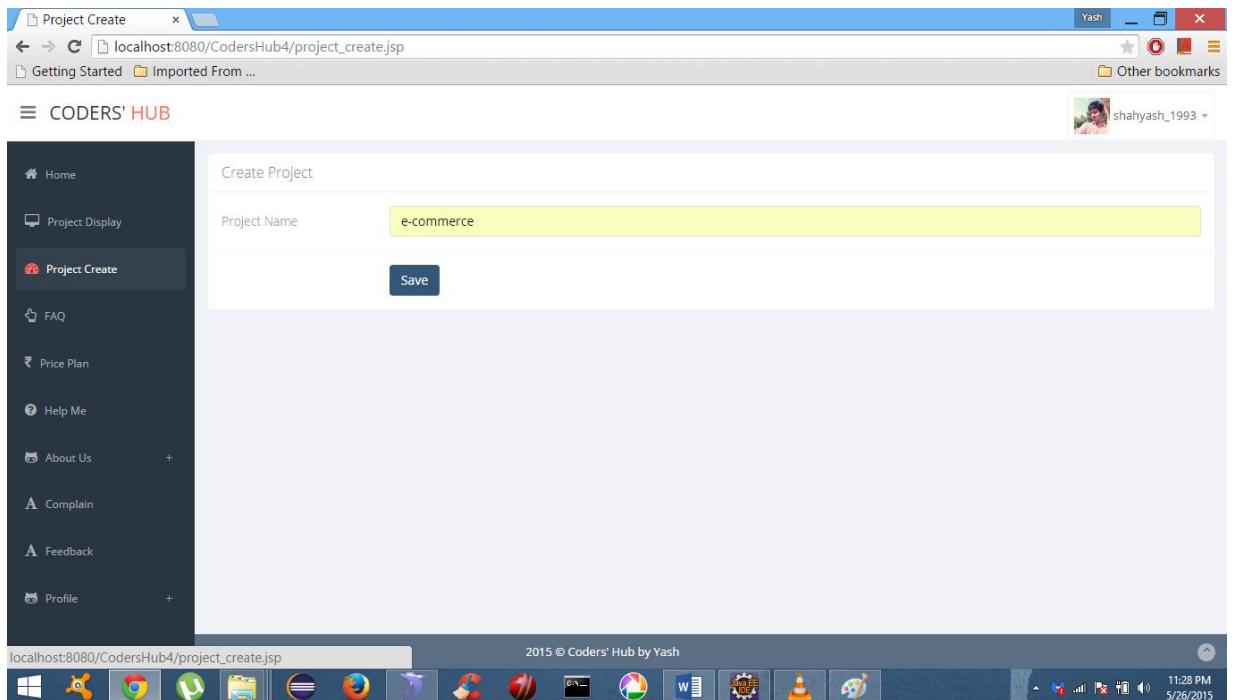
Figure 6.12: Registration



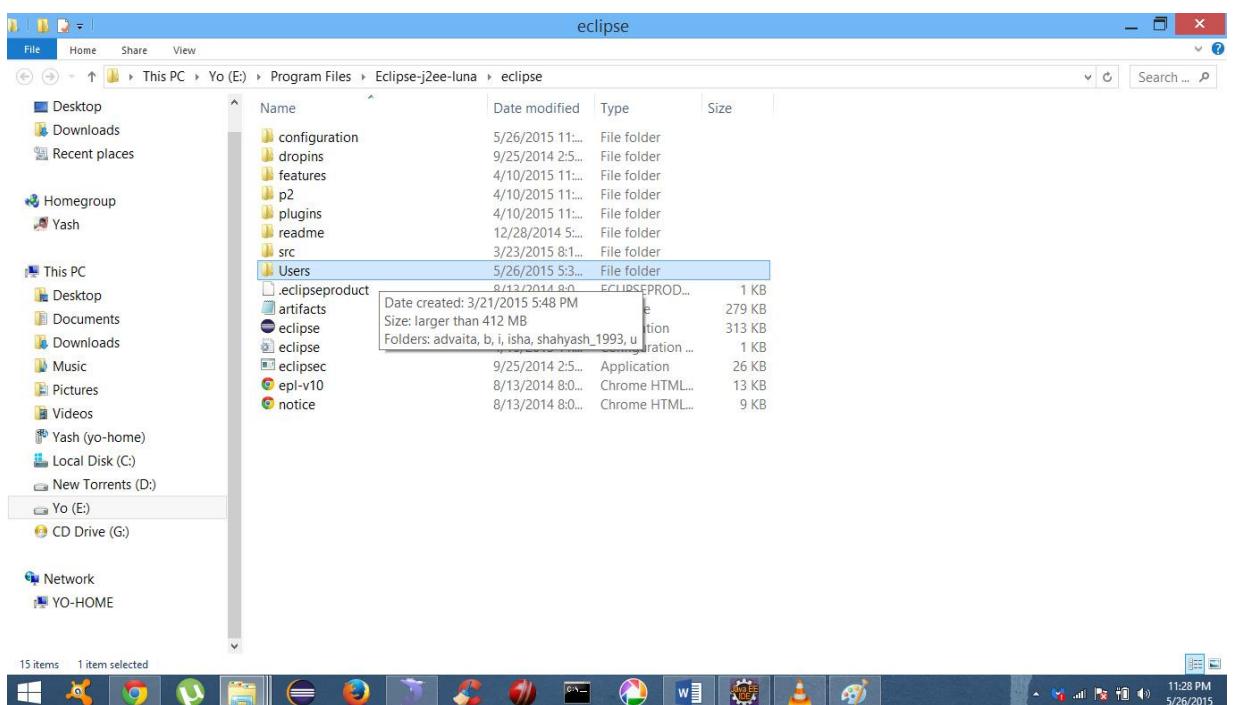
**Figure 6.13: Forgot Password**



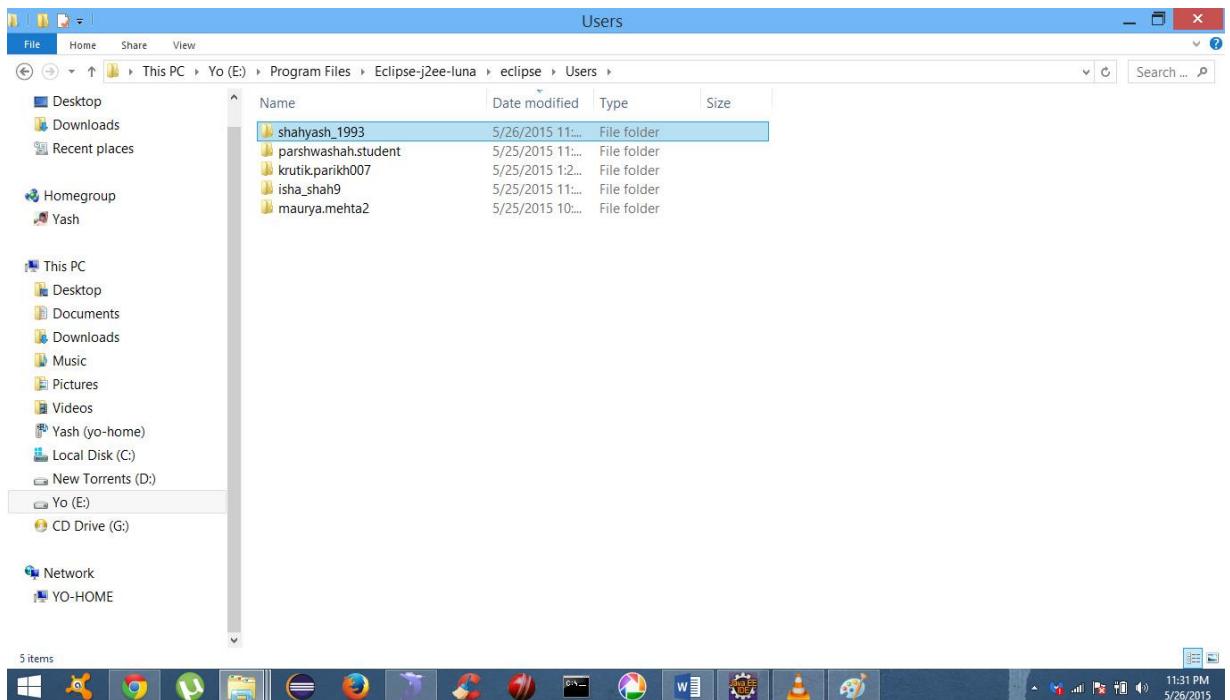
**Figure 6.14: User Dashboard**



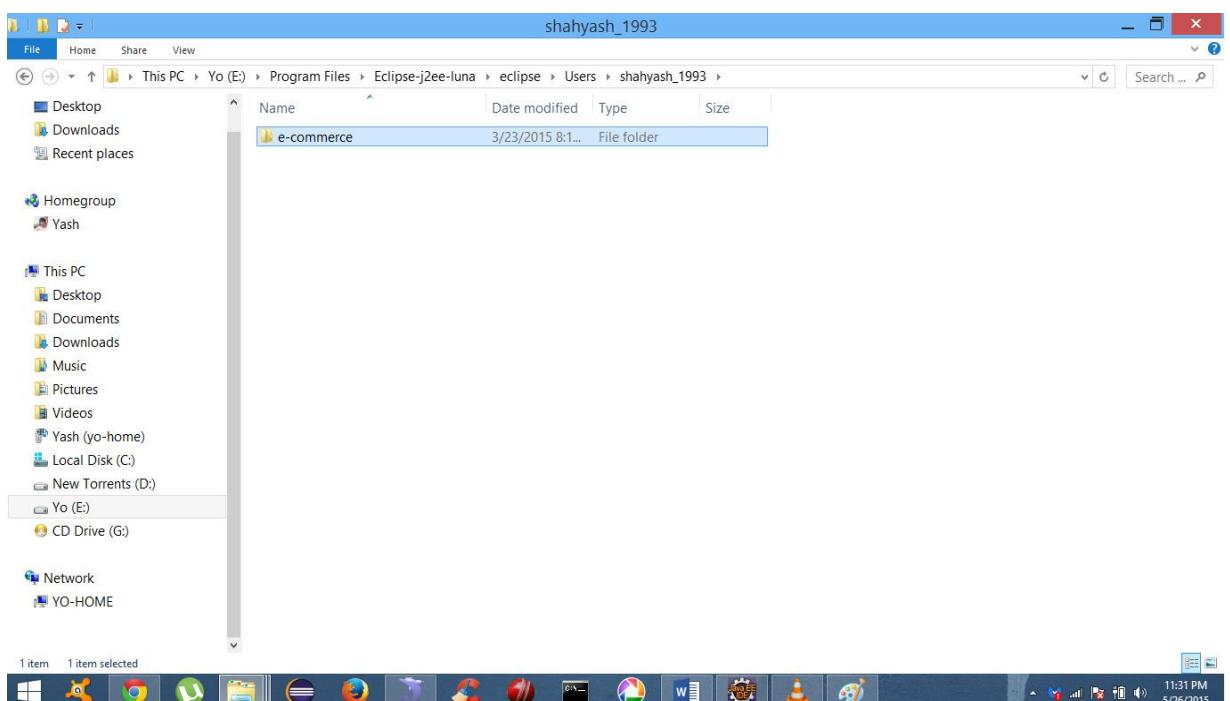
**Figure 6.15: Create Project**



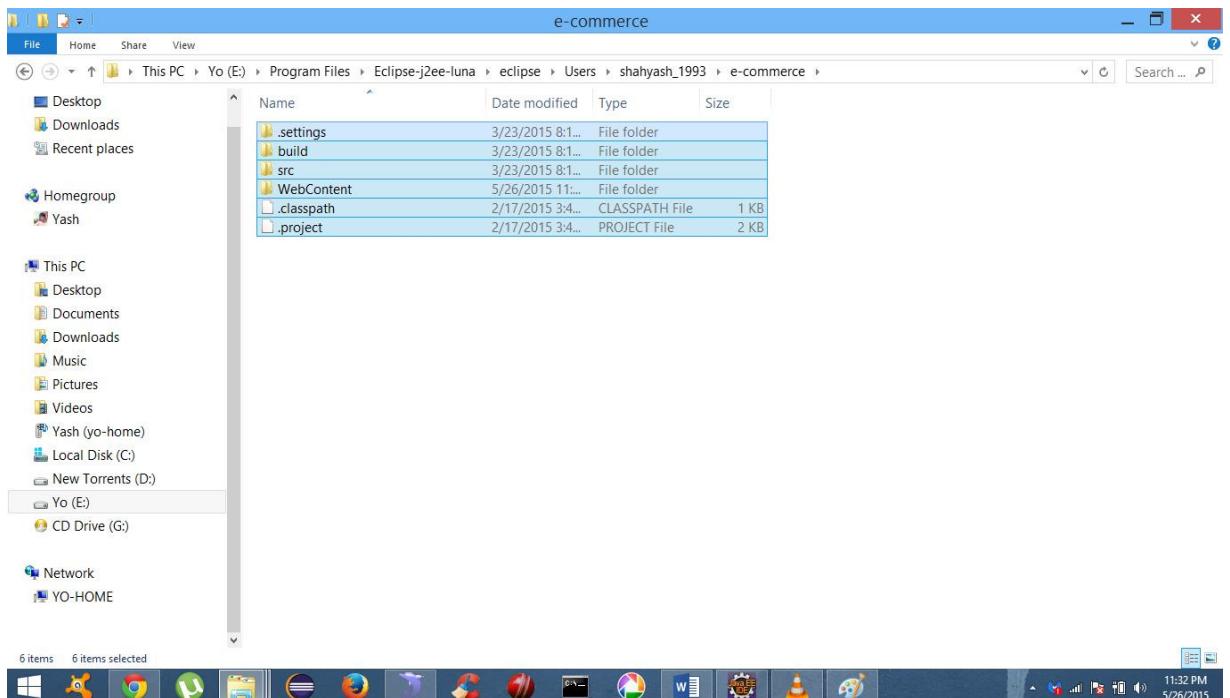
**Figure 6.16: Create Project FileIO (1)**



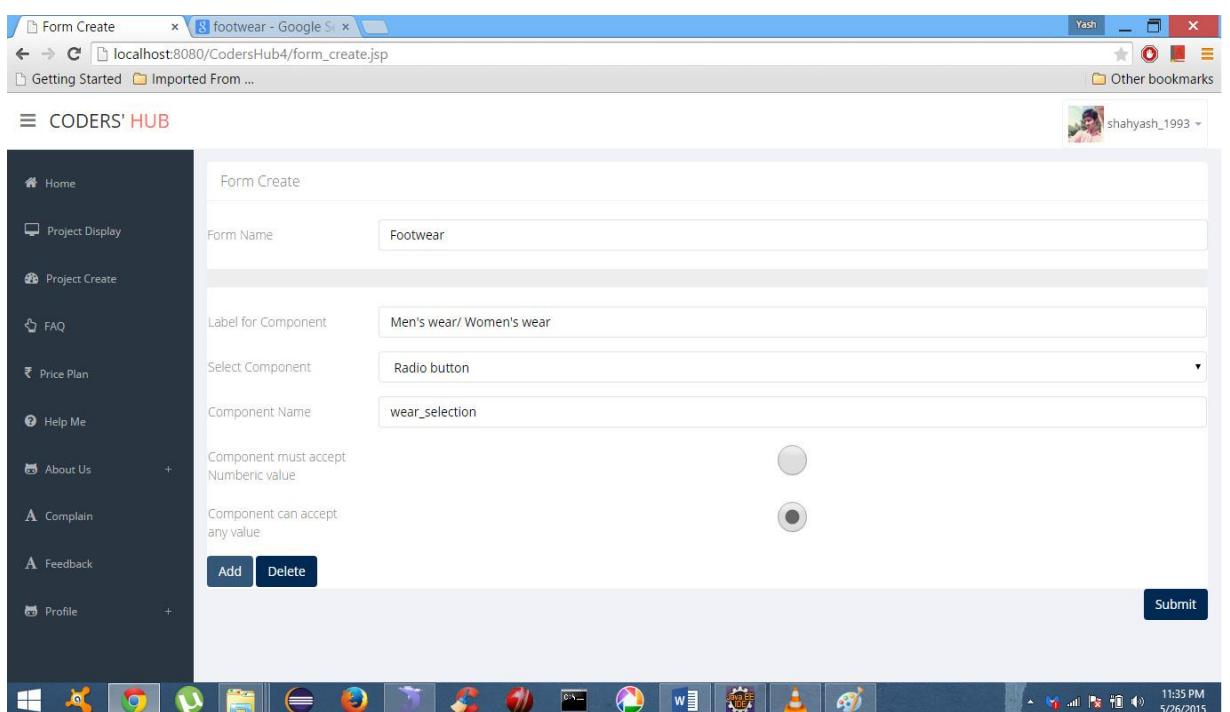
**Figure 6.17: Create Project FileIO (2)**



**Figure 6.18: Create Project FileIO (3)**



**Figure 6.19: Create Project FileIO (4)**



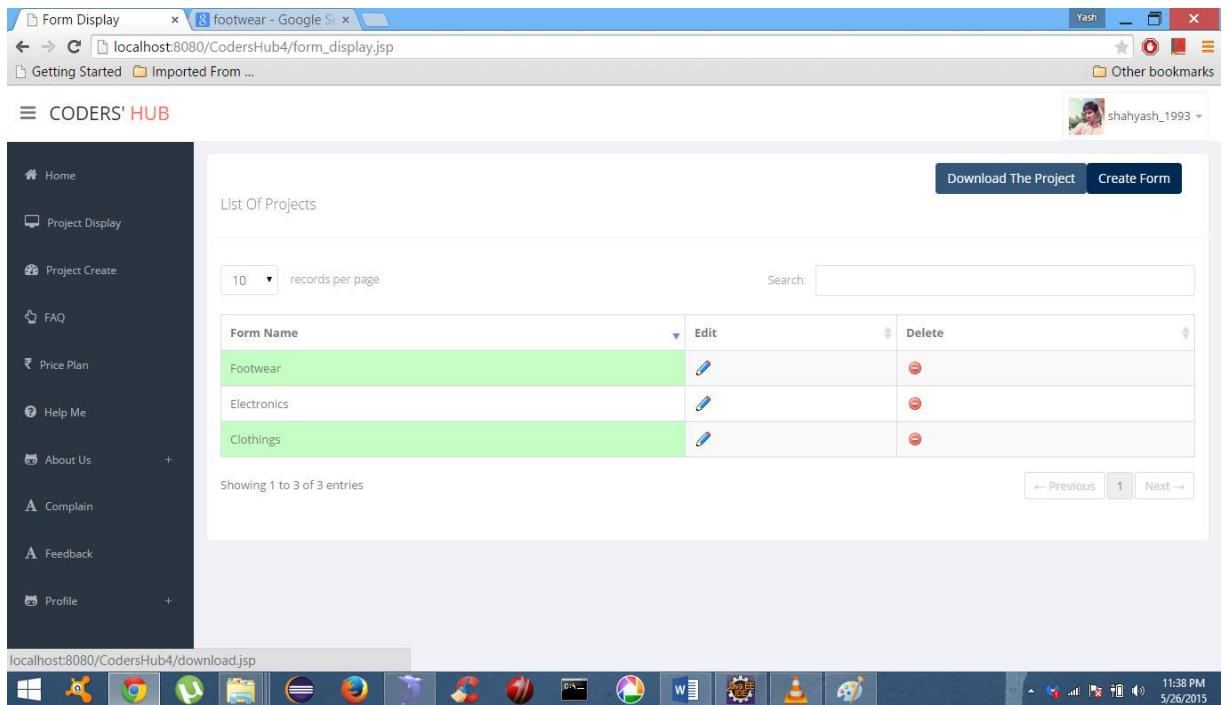
**Figure 6.20: Form Create (1)**

The screenshot shows a web application titled 'Form Create' on a browser window. The URL is 'localhost:8080/CodersHub4/form\_create.jsp'. The page has a sidebar on the left with various links: Home, Project Display, Project Create, FAQ, Price Plan, Help Me, About Us, Complain, Feedback, and Profile. The main content area displays a component configuration form. A note says 'Component can accept any value'. Below it, there are two radio buttons: one selected (black dot) and one unselected (grey dot). A 'Text Area' input field is labeled 'Item Id'. Another note says 'Select Component' and 'Component Name' is set to 'item\_id'. A second set of radio buttons is shown, both unselected (grey dots). At the bottom are 'Add' and 'Delete' buttons, and a large 'Submit' button.

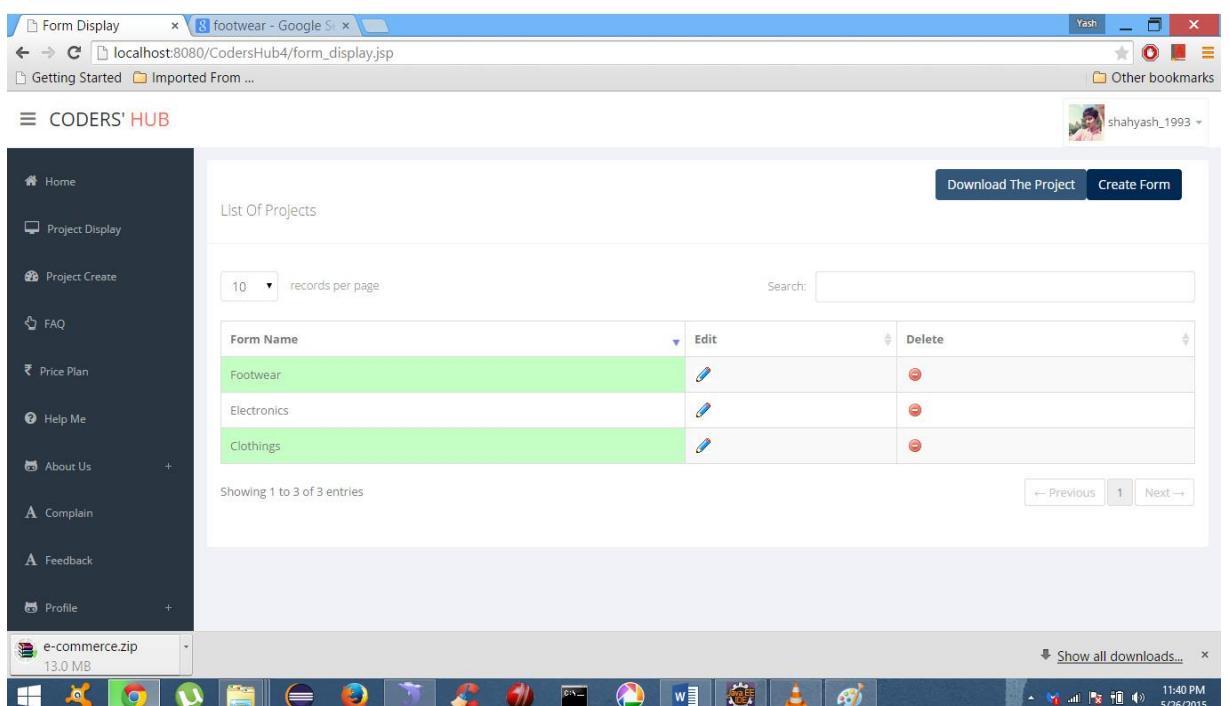
**Figure 6.21: Form Create (2)**

This screenshot shows the same 'Form Create' application as Figure 6.21, but with different component configurations. The first section has a note 'Component value of anything' and two unselected radio buttons. The 'Text Area' input field is labeled 'Item Name'. The second section has a note 'Component value must Number' and two unselected radio buttons. The 'Text Box' input field is labeled 'Select Component' and 'Component Name' is set to 'item\_name'. At the bottom are 'Add' and 'Delete' buttons, and a large 'Submit' button.

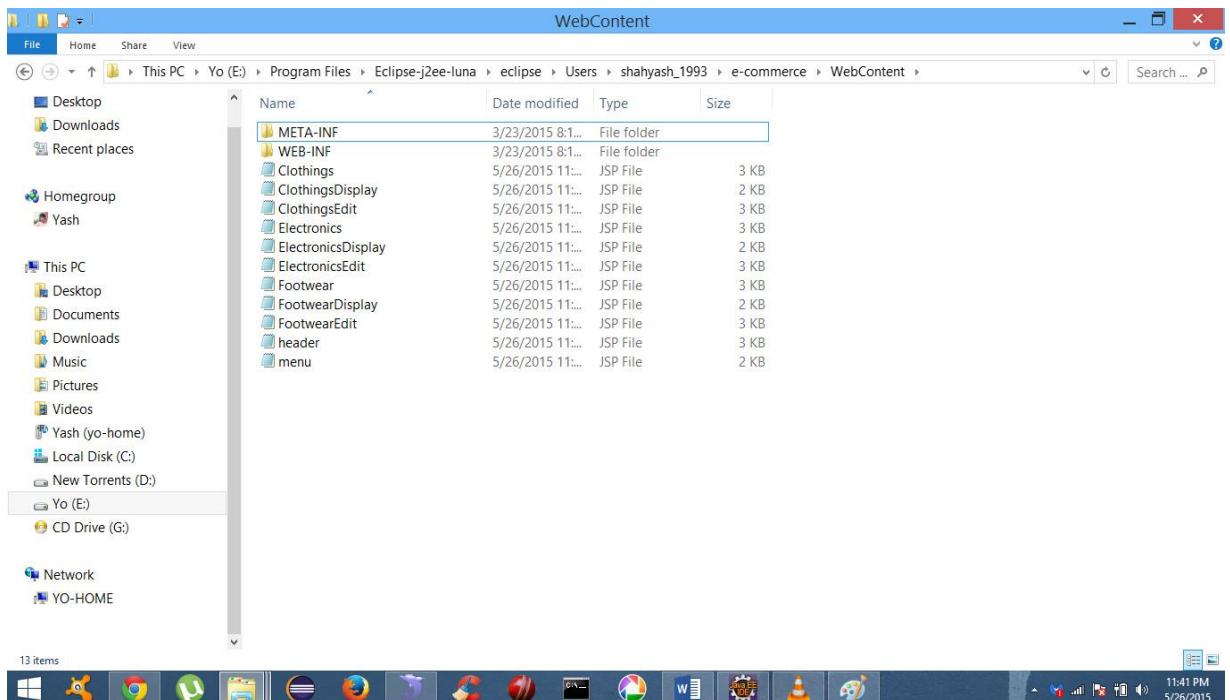
**Figure 6.22: Form Create (3)**



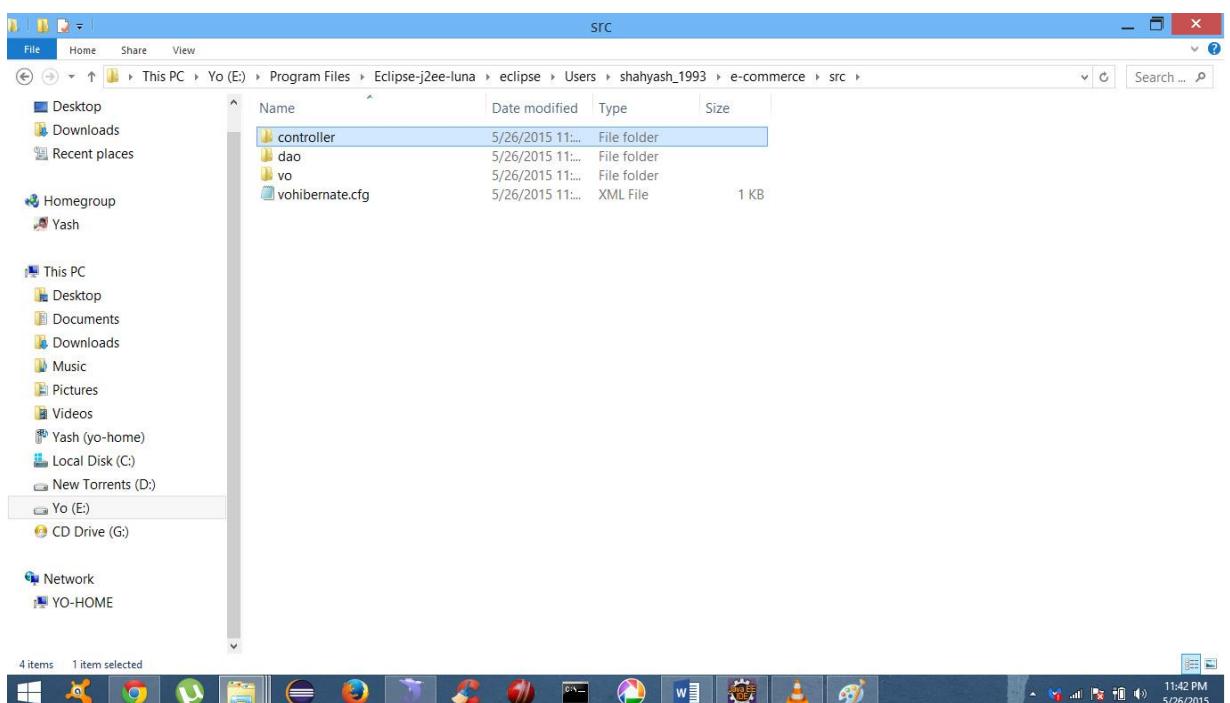
**Figure 6.23: Form Display (1)**



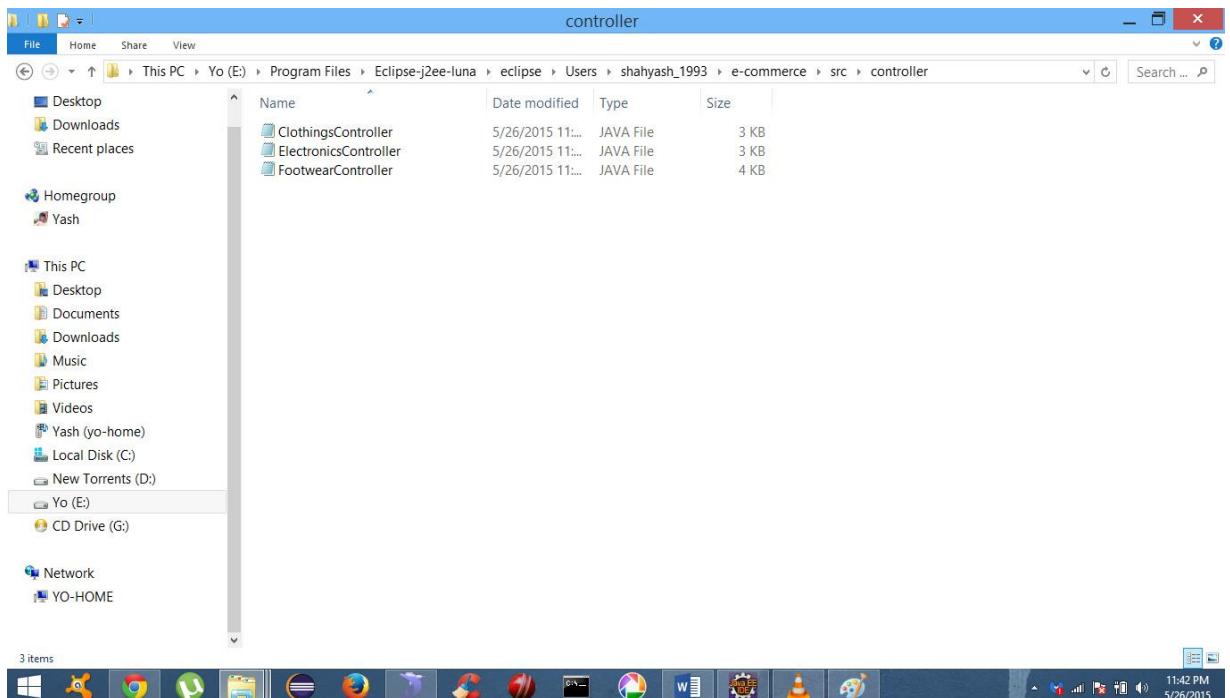
**Figure 6.24: Form Display (2)**



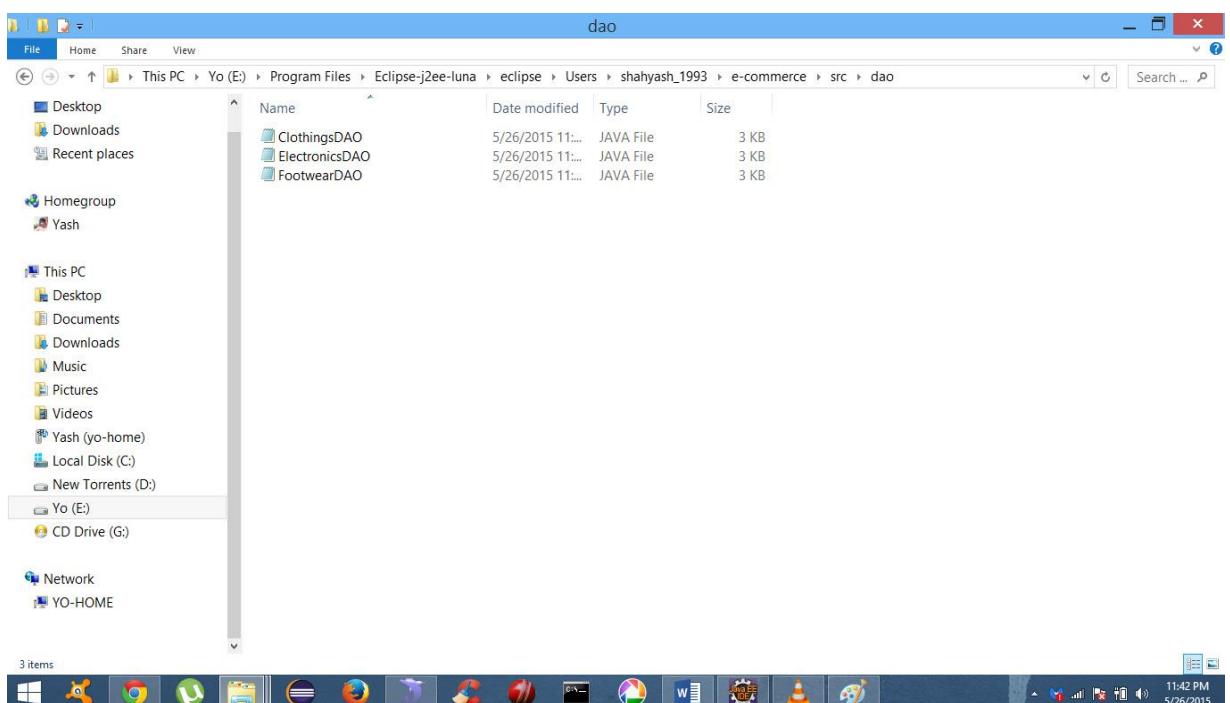
**Figure 6.25: Create Form IO (1)**



**Figure 6.26: Create Form IO (2)**



**Figure 6.27: Create Form IO (3)**



**Figure 6.28: Create Form IO (4)**

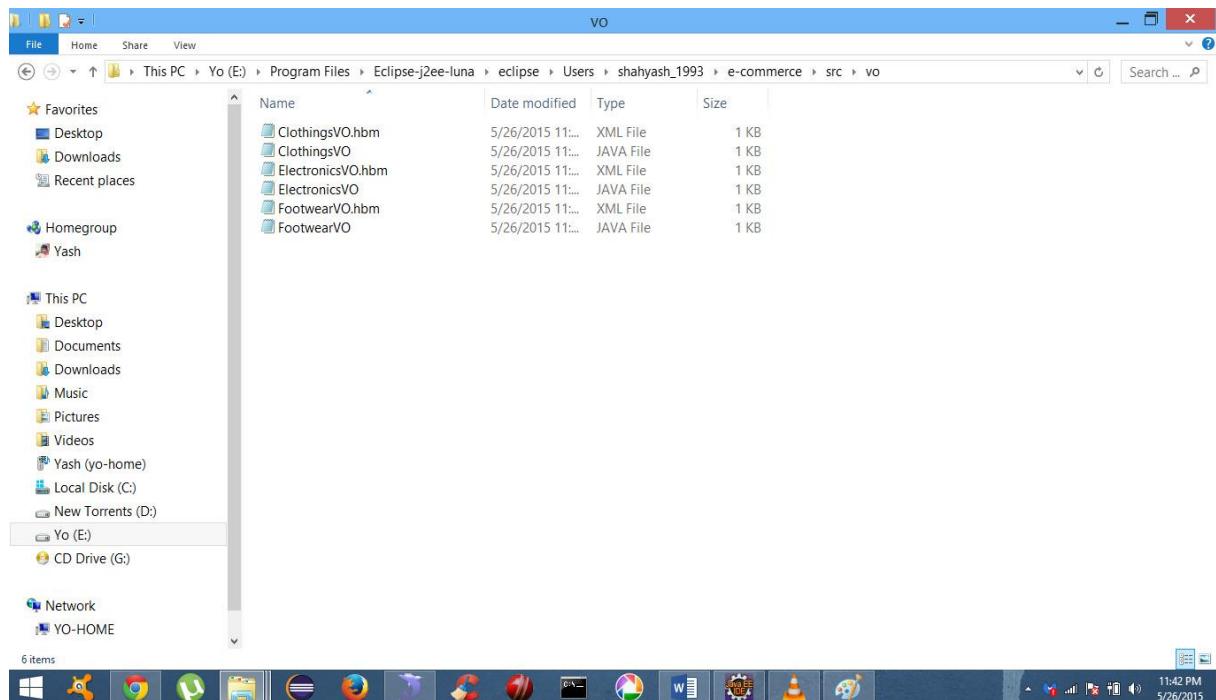


Figure 6.29: Create Form IO (5)

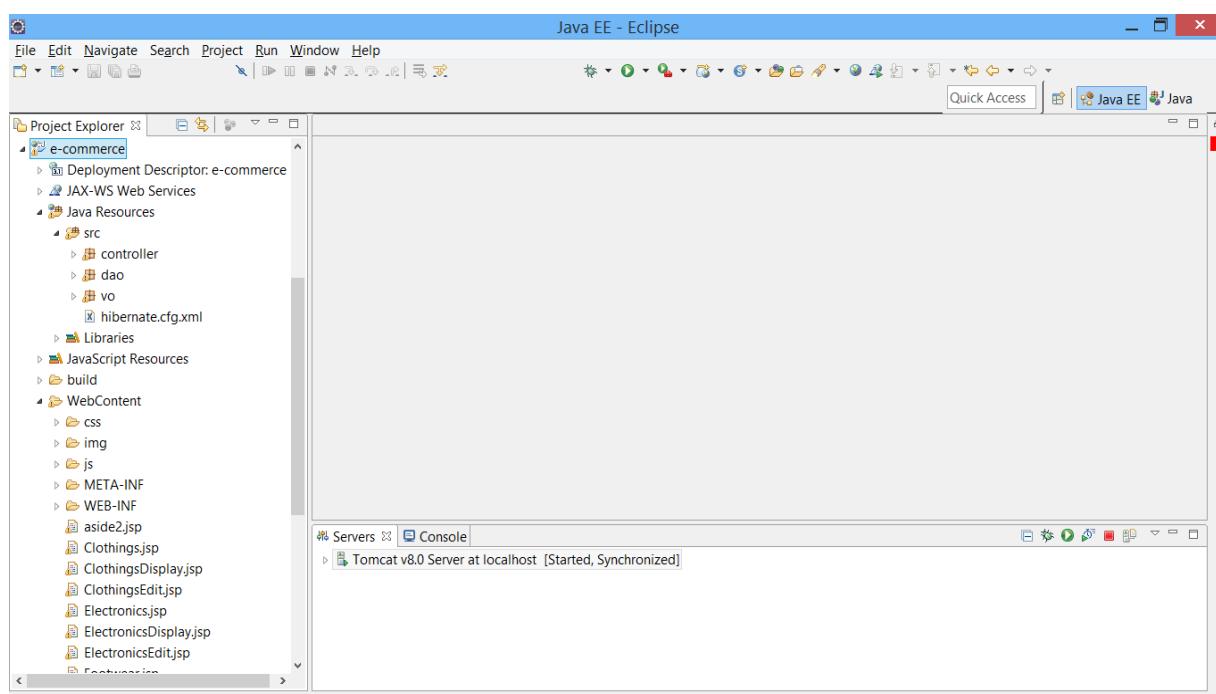
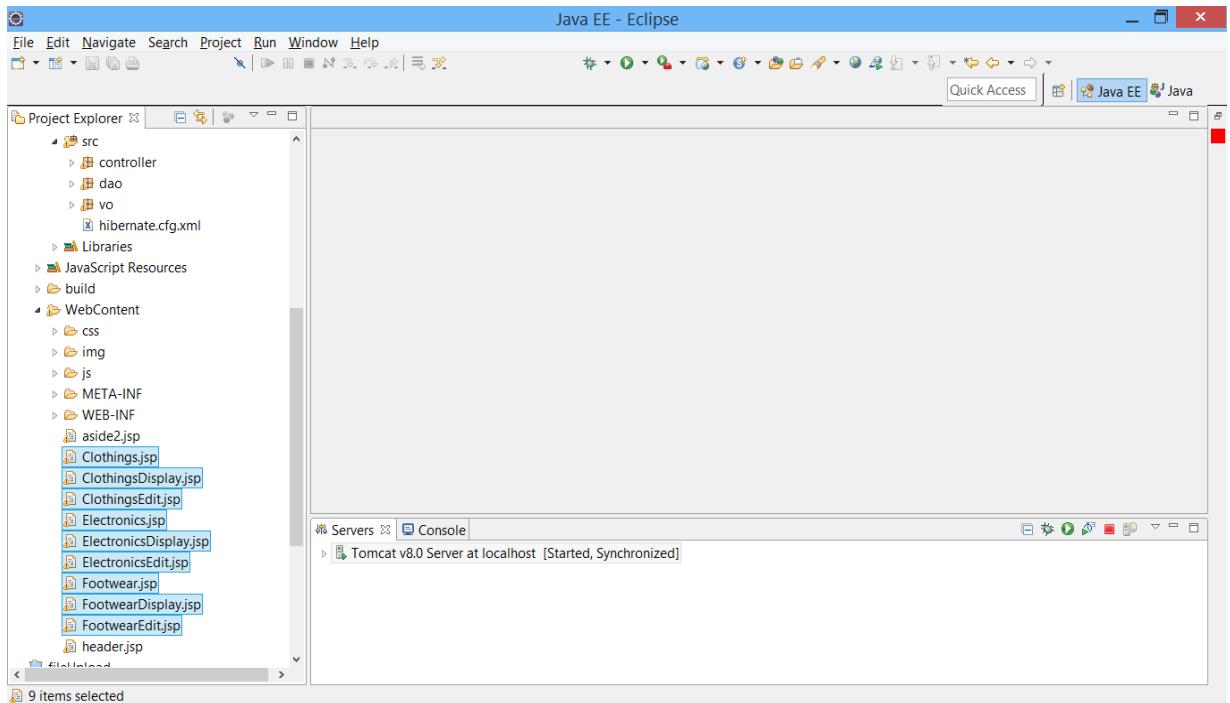
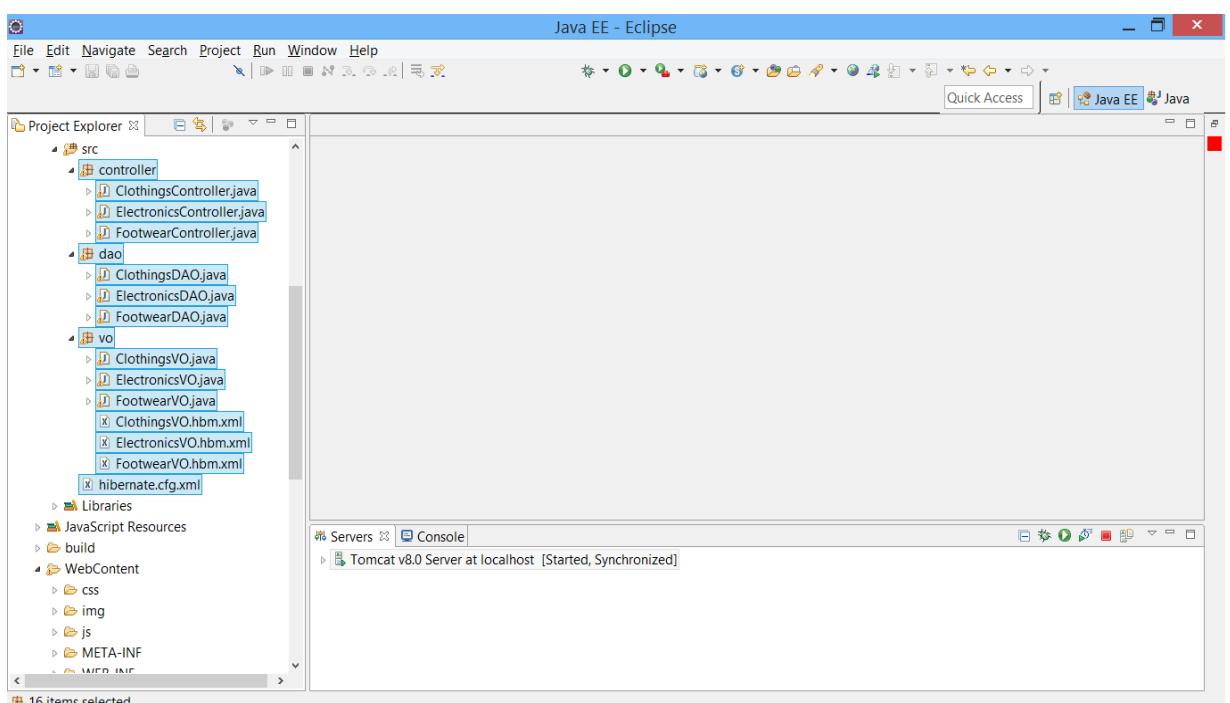


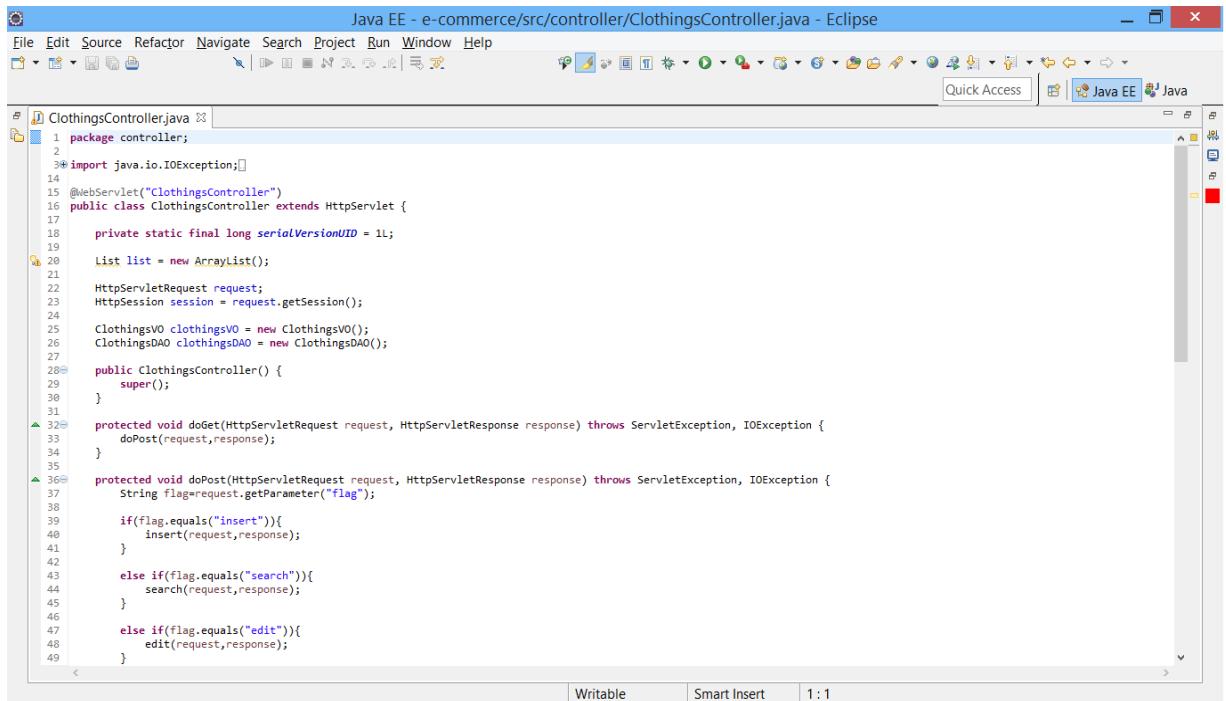
Figure 6.30: Generated Project



**Figure 6.31: Generated JSPs**



**Figure 6.32: Generated Controllers, DAOs, VOs**



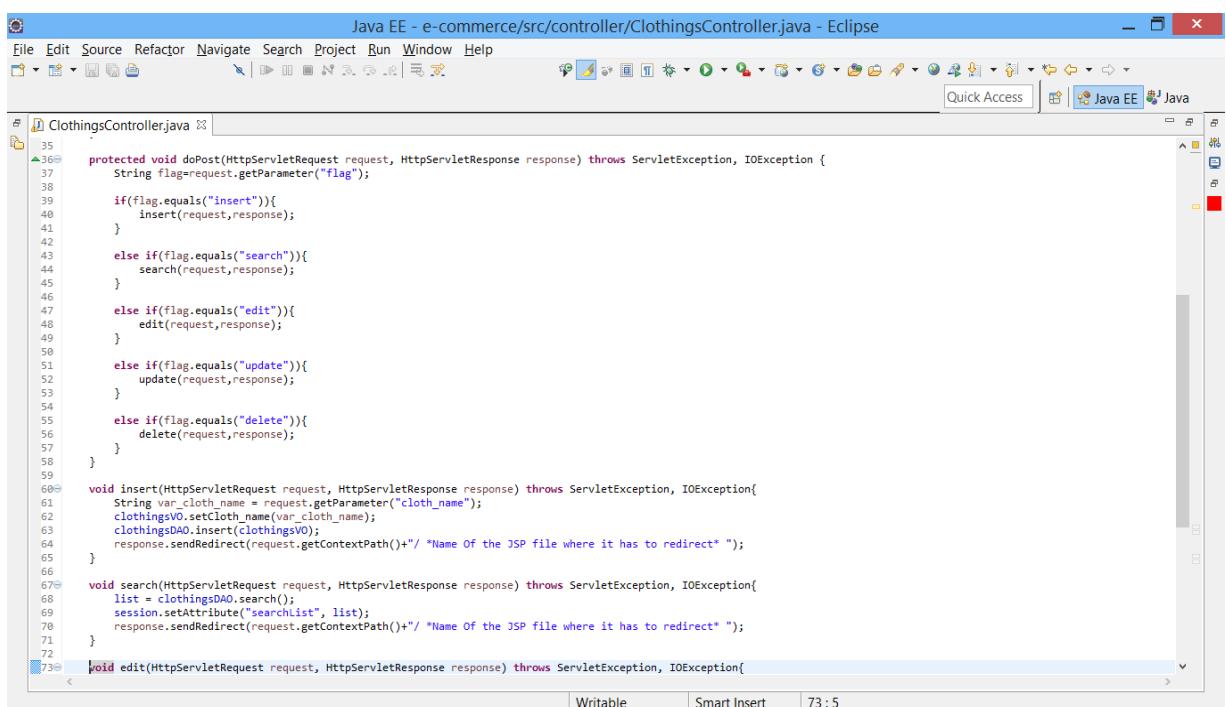
The screenshot shows the Eclipse IDE interface with the title bar "Java EE - e-commerce/src/controller/ClothingsController.java - Eclipse". The code editor displays the following Java code:

```

1 package controller;
2
3 import java.io.IOException;
4
5 @WebServlet("ClothingsController")
6 public class ClothingsController extends HttpServlet {
7
8     private static final long serialVersionUID = 1L;
9
10    List list = new ArrayList();
11
12    HttpServletRequest request;
13    HttpSession session = request.getSession();
14
15    ClothingsVO clothingsVO = new ClothingsVO();
16    ClothingsDAO clothingsDAO = new ClothingsDAO();
17
18    public ClothingsController() {
19        super();
20    }
21
22    protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
23        doPost(request, response);
24    }
25
26    protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
27        String flag= request.getParameter("flag");
28
29        if(flag.equals("insert")){
30            insert(request, response);
31        }
32
33        else if(flag.equals("search")){
34            search(request, response);
35        }
36
37        else if(flag.equals("edit")){
38            edit(request, response);
39        }
40
41    }
42
43    void insert(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException{
44        String var_cloth_name = request.getParameter("cloth_name");
45        clothingsVO.setCloth_name(var_cloth_name);
46        clothingsDAO.insert(clothingsVO);
47        response.sendRedirect(request.getContextPath() + " / Name Of the JSP file where it has to redirect" );
48    }
49
50    void search(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException{
51        list = clothingsDAO.search();
52        session.setAttribute("searchlist", list);
53        response.sendRedirect(request.getContextPath() + " / Name Of the JSP file where it has to redirect" );
54    }
55
56    void edit(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException{
57    }
58
59}

```

**Figure 6.33: Generated Controller (1)**



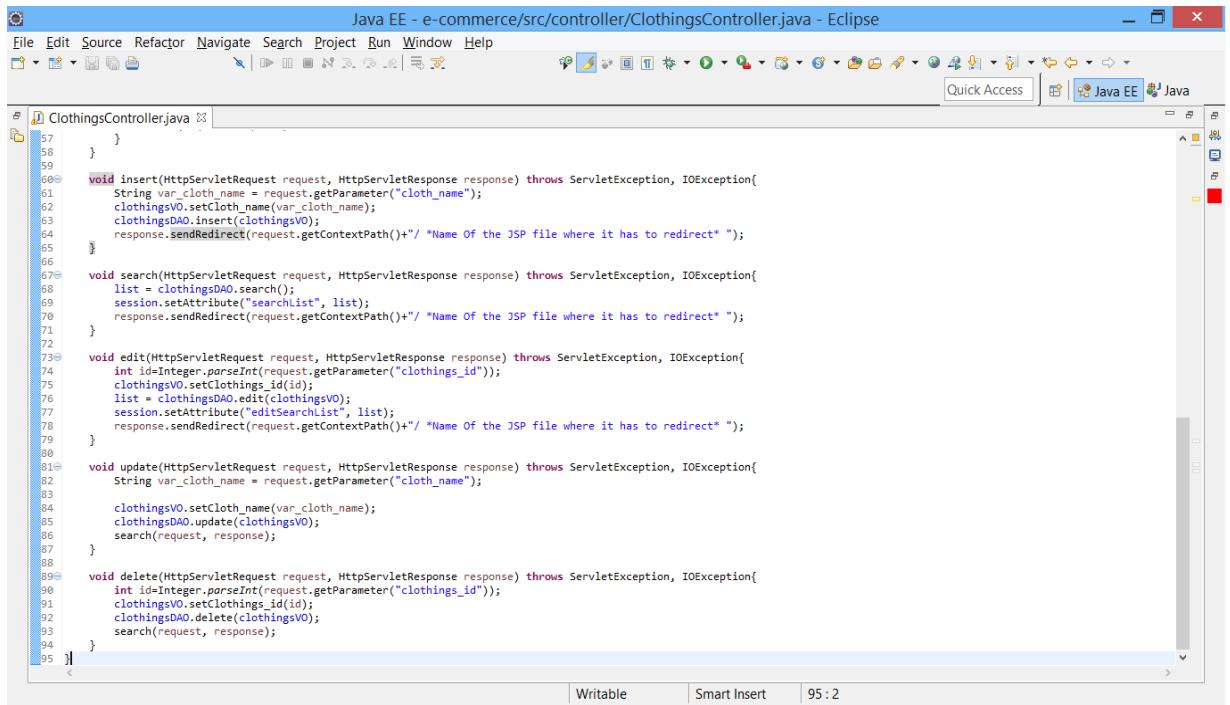
The screenshot shows the Eclipse IDE interface with the title bar "Java EE - e-commerce/src/controller/ClothingsController.java - Eclipse". The code editor displays the following Java code, which includes additional methods:

```

1 package controller;
2
3 import java.io.IOException;
4
5 @WebServlet("ClothingsController")
6 public class ClothingsController extends HttpServlet {
7
8     private static final long serialVersionUID = 1L;
9
10    List list = new ArrayList();
11
12    HttpServletRequest request;
13    HttpSession session = request.getSession();
14
15    ClothingsVO clothingsVO = new ClothingsVO();
16    ClothingsDAO clothingsDAO = new ClothingsDAO();
17
18    public ClothingsController() {
19        super();
20    }
21
22    protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
23        doPost(request, response);
24    }
25
26    protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
27        String flag= request.getParameter("flag");
28
29        if(flag.equals("insert")){
30            insert(request, response);
31        }
32
33        else if(flag.equals("search")){
34            search(request, response);
35        }
36
37        else if(flag.equals("edit")){
38            edit(request, response);
39        }
40
41        else if(flag.equals("update")){
42            update(request, response);
43        }
44
45        else if(flag.equals("delete")){
46            delete(request, response);
47        }
48    }
49
50    void insert(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException{
51        String var_cloth_name = request.getParameter("cloth_name");
52        clothingsVO.setCloth_name(var_cloth_name);
53        clothingsDAO.insert(clothingsVO);
54        response.sendRedirect(request.getContextPath() + " / Name Of the JSP file where it has to redirect" );
55    }
56
57    void search(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException{
58        list = clothingsDAO.search();
59        session.setAttribute("searchlist", list);
60        response.sendRedirect(request.getContextPath() + " / Name Of the JSP file where it has to redirect" );
61    }
62
63    void update(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException{
64    }
65
66    void delete(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException{
67    }
68
69    void edit(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException{
70    }
71
72}

```

**Figure 6.34: Generated Controller (2)**



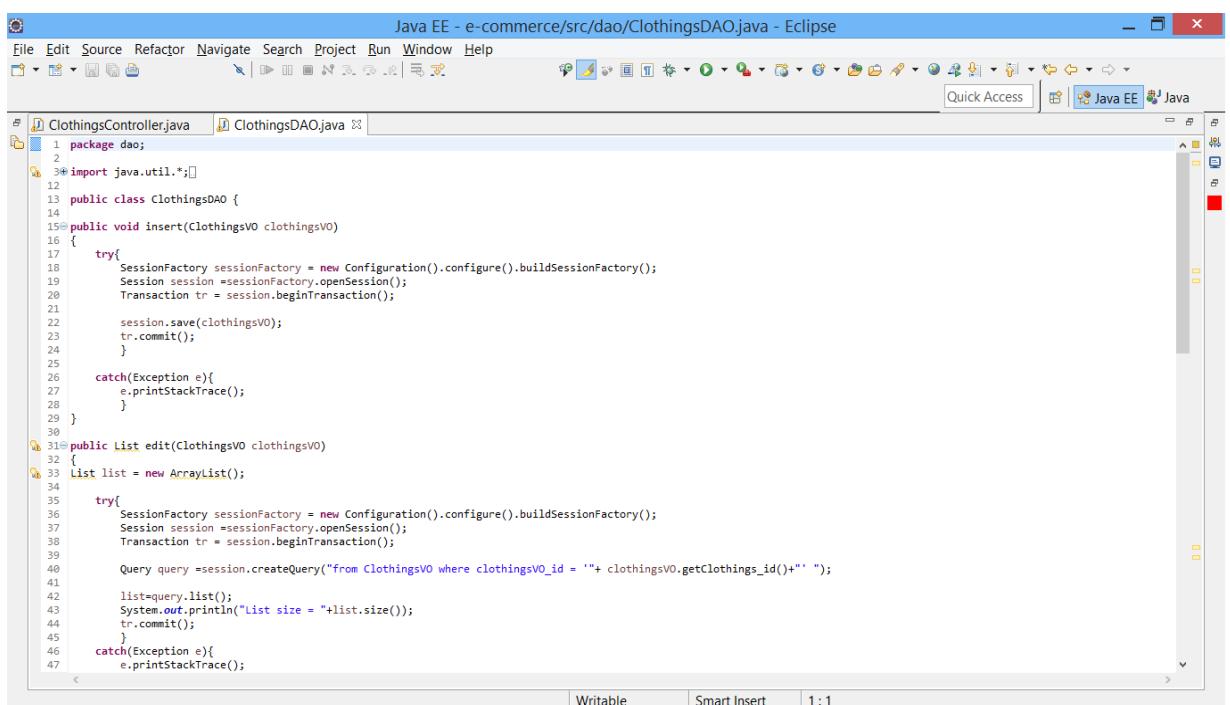
The screenshot shows the Eclipse IDE interface with the title bar "Java EE - e-commerce/src/controller/ClothingsController.java - Eclipse". The code editor displays the generated Java code for the ClothingsController class. The code includes methods for insert, search, edit, update, and delete operations, all interacting with a ClothingsDAO object.

```

1 package controller;
2
3 import java.util.*;
4
5 public class ClothingsController {
6
7     }
8
9
10    void insert(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException{
11        String var_cloth_name = request.getParameter("cloth_name");
12        clothingsVO.setCloth_name(var_cloth_name);
13        clothingsDAO.insert(clothingsVO);
14        response.sendRedirect(request.getContextPath() + "/ Name Of the JSP file where it has to redirect");
15    }
16
17    void search(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException{
18        list = clothingsDAO.search();
19        session.setAttribute("searchList", list);
20        response.sendRedirect(request.getContextPath() + "/ Name Of the JSP file where it has to redirect");
21    }
22
23    void edit(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException{
24        int idInteger.parseInt(request.getParameter("clothings_id"));
25        clothingsVO.setClothings_id(id);
26        list = clothingsDAO.edit(clothingsVO);
27        session.setAttribute("editSearchList", list);
28        response.sendRedirect(request.getContextPath() + "/ Name Of the JSP file where it has to redirect");
29    }
30
31    void update(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException{
32        String var_cloth_name = request.getParameter("cloth_name");
33
34        clothingsVO.setCloth_name(var_cloth_name);
35        clothingsDAO.update(clothingsVO);
36        search(request, response);
37    }
38
39    void delete(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException{
40        int idInteger.parseInt(request.getParameter("clothings_id"));
41        clothingsVO.setClothings_id(id);
42        clothingsDAO.delete(clothingsVO);
43        search(request, response);
44    }
45}

```

**Figure 6.35: Generated Controller (3)**



The screenshot shows the Eclipse IDE interface with the title bar "Java EE - e-commerce/src/dao/ClothingsDAO.java - Eclipse". The code editor displays the generated Java code for the ClothingsDAO class. It includes methods for insert, edit, and a query-based method that uses a SessionFactory and Transaction to interact with the database.

```

1 package dao;
2
3 import java.util.*;
4
5 public class ClothingsDAO {
6
7
8    public void insert(ClothingsVO clothingsVO) {
9        try{
10            SessionFactory sessionFactory = new Configuration().configure().buildSessionFactory();
11            Session session =sessionFactory.openSession();
12            Transaction tr = session.beginTransaction();
13            session.save(clothingsVO);
14            tr.commit();
15        }
16
17        catch(Exception e){
18            e.printStackTrace();
19        }
20    }
21
22    public List edit(ClothingsVO clothingsVO) {
23
24        List list = new ArrayList();
25
26        try{
27            SessionFactory sessionFactory = new Configuration().configure().buildSessionFactory();
28            Session session =sessionFactory.openSession();
29            Transaction tr = session.beginTransaction();
30
31            Query query =session.createQuery("from ClothingsVO where clothingsVO_id = '"+clothingsVO.getClothings_id()+"'");
32
33            list=query.list();
34
35            System.out.println("List size = "+list.size());
36            tr.commit();
37        }
38
39        catch(Exception e){
40            e.printStackTrace();
41        }
42    }
43
44}

```

**Figure 6.36: Generated DAO (1)**

The screenshot shows the Eclipse IDE interface with the title bar "Java EE - e-commerce/src/dao/ClothingsDAO.java - Eclipse". The code editor displays Java code for a DAO class:

```
45     }
46     catch(Exception e){
47         e.printStackTrace();
48     }
49
50     return list;
51 }
52
53 public void update(ClothingsVO clothingsVO)
54 {
55     try{
56         SessionFactory sessionFactory = new Configuration().configure().buildSessionFactory();
57         Session session =sessionFactory.openSession();
58         Transaction tr = session.beginTransaction();
59
60         session.saveOrUpdate(clothingsVO);
61         tr.commit();
62     }
63
64     catch(Exception e){
65         e.printStackTrace();
66     }
67 }
68
69 public void delete(ClothingsVO clothingsVO)
70 {
71     try{
72         SessionFactory sessionFactory = new Configuration().configure().buildSessionFactory();
73         Session session =sessionFactory.openSession();
74         Transaction tr = session.beginTransaction();
75
76         session.delete(clothingsVO);
77         tr.commit();
78     }
79
80     catch(Exception e){
81         e.printStackTrace();
82     }
83 }
```

The code implements methods for updating and deleting `ClothingsVO` objects using a SessionFactory and Transaction.

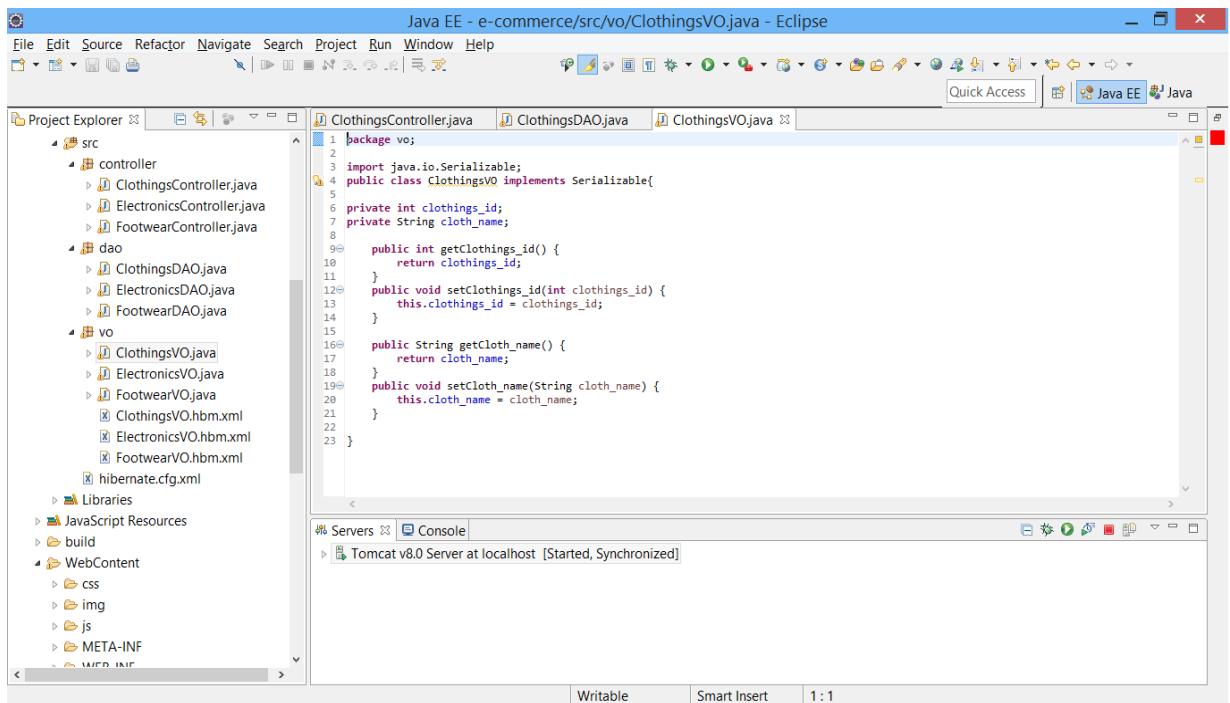
**Figure 6.37: Generated DAO (2)**

The screenshot shows the Eclipse IDE interface with the title bar "Java EE - e-commerce/src/dao/ClothingsDAO.java - Eclipse". The code editor displays Java code for a DAO class:

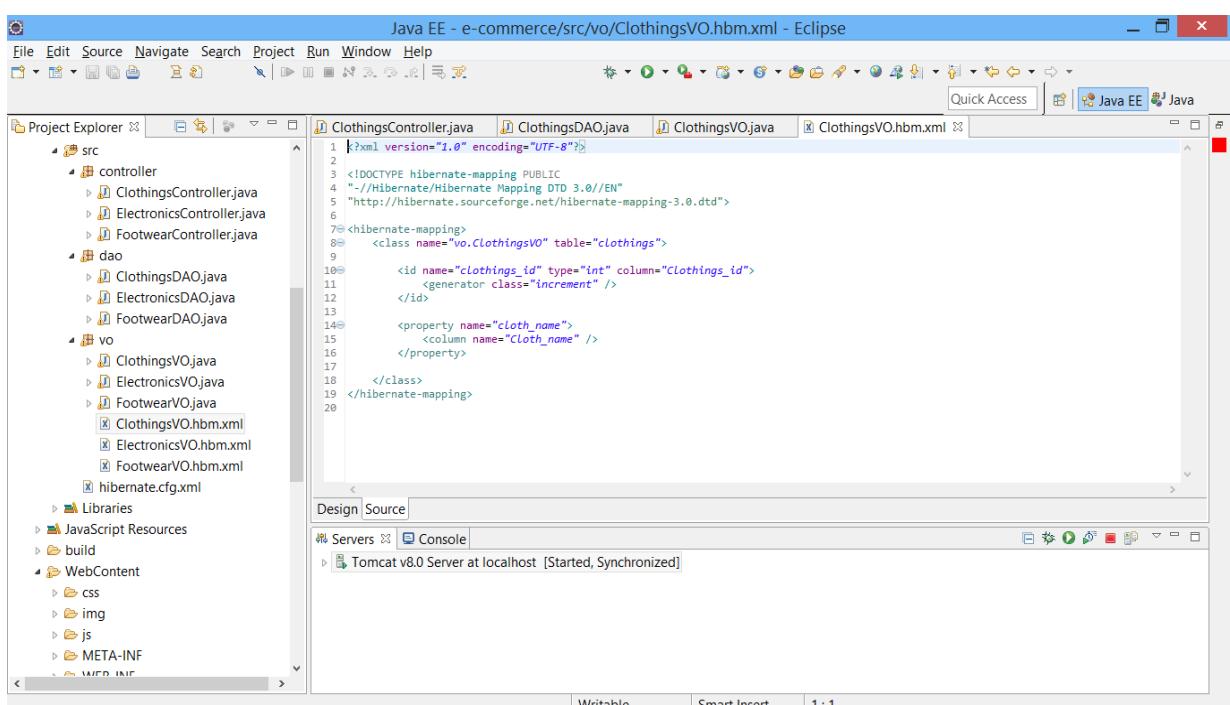
```
68
69 public void delete(ClothingsVO clothingsVO)
70 {
71     try{
72         SessionFactory sessionFactory = new Configuration().configure().buildSessionFactory();
73         Session session =sessionFactory.openSession();
74         Transaction tr = session.beginTransaction();
75
76         session.delete(clothingsVO);
77         tr.commit();
78     }
79
80     catch(Exception e){
81         e.printStackTrace();
82     }
83 }
84
85 public List search()
86 {
87     List list = new ArrayList();
88
89     try{
90         SessionFactory sessionFactory = new Configuration().configure().buildSessionFactory();
91         Session session =sessionFactory.openSession();
92         Transaction tr = session.beginTransaction();
93
94         Query query =session.createQuery("from ClothingsVO");
95
96         list=query.list();
97         System.out.println("List size = "+list.size());
98         tr.commit();
99     }
100    catch(Exception e){
101        e.printStackTrace();
102    }
103
104    return list;
105 }
106 }
```

The code implements a `search` method that queries the database for `ClothingsVO` objects using a named query and prints the list size.

**Figure 6.38: Generated DAO (3)**



**Figure 6.39: Generated VO**



**Figure 6.40: Generated vo.hbm**

The screenshot shows the Eclipse IDE interface with the title "Java EE - e-commerce/src/hibernate.cfg.xml - Eclipse". The Project Explorer view on the left lists several Java files under "src" (controller, dao, vo) and XML files under "vo" (ClothingsVO.hbm.xml, ElectronicsVO.hbm.xml, FootwearVO.hbm.xml). The main editor window displays the content of the hibernate.cfg.xml file:

```

1 <?xml version='1.0' encoding='UTF-8'?>
2 <!DOCTYPE hibernate-configuration PUBLIC
3   "-//Hibernate/Hibernate Configuration DTD//EN"
4   "http://hibernate.sourceforge.net/hibernate-configuration-3.0.dtd">
5<hibernate-configuration>
6<session-factory>
7   <property name="hibernate.connection.driver_class">com.mysql.jdbc.Driver</property>
8   <property name="hibernate.connection.url">jdbc:mysql://localhost/e-commerce3</property>
9   <property name="hibernate.connection.username">root</property>
10  <property name="hibernate.connection.password">root</property>
11  <property name="show_sql">true</property>
12  <property name="dialect">org.hibernate.dialect.MySQLDialect</property>
13  <property name="hibernate.hbm2ddl.auto">update</property>
14  <!-- Mapping files -->
15  <mapping resource="vo/FootwearVO.hbm.xml" />
16  <mapping resource="vo/ElectronicsVO.hbm.xml" />
17  <mapping resource="vo/ClothingsVO.hbm.xml" />
18  </session-factory>
19 </hibernate-configuration>

```

The Servers view shows "Tomcat v8.0 Server at localhost [Started, Synchronized]".

**Figure 6.41: Generated hibernate.cfg.xml**

The screenshot shows the Eclipse IDE interface with the title "Java EE - e-commerce/WebContent/Clothings.jsp - Eclipse". The Project Explorer view on the left lists several Java files and a JSP file. The main editor window displays the content of the Clothings.jsp JSP page:

```

1 <%@ page language="java" contentType="text/html; charset=ISO-8859-1"
2   pageEncoding="ISO-8859-1"%>
3 <!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
4<html>
5<head>
6 <meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
7 <title>Clothings</title>
8</head>
9<body>
10 <%@include file="header.jsp"%>
11 <%@include file="aside2.jsp"%>
12 <!-- main content start-->
13<section id="main-content"> <section
14   class="wrapper site-min-height" ><!-- page start-->
15<div class="row">
16<div class="col-lg-12">
17<section class="panel" > <header class="panel-heading">Clothings</header>
18<div class="panel-body">
19<form class="form-horizontal" id="default"
20   action=<%request.getContextPath()%>/ClothingsController">
21<fieldset title="Step1" class="step" id="default-step-0">
22
23
24<div class="form-group">
25   <label class="col-lg-2 control-label">Cloth Name</label>
26<div class="col-lg-10">
27   <input type="text" class="form-control" placeholder="Cloth Name"
28   name="cloth_name" id="cloth_name">
29
30</div>
31</div>
32<br />
33
34</fieldset>
35<input type="submit" class="finish btn btn-danger" value="Submit" />
36</form>
37</div>
38</section>
39</div>
<
```

**Figure 6.42: Generated Required\_JSP (1)**

```

Java EE - e-commerce/WebContent/Clothings.jsp - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
ClothingsController.java ClothingsDAO.java ClothingsVO.java ClothingsVO.hbm.xml hibernate.cfg.xml Clothings.jsp
33     </fieldset>
34     <input type="submit" class="finish btn btn-danger" value="Submit" />
35   </form>
36 </div>
37 </section>
38 </div>
39 </div>
40 </div>
41 <!-- page end--> </section> </section>
42@ <!-- main content end-->
43 <!-- footer start-->
44@ <footer class="site-footer">
45@   <div class="text-center">
46     Clothings<a href="#" class="go-top"> <i class="fa fa-angle-up"></i></a>
47   </div>
48 </footer>
49@ <!-- footer end-->
50 </section>
51 <!-- js placed at the end of the document so the pages load faster -->
52 <!--<script src="js/jquery.js"></script>-->
53@ <script type="text/javascript" language="javascript"
54   src="assets/advanced-dataTable/media/js/jquery.js"></script>
55 <script src="js/bootstrap.min.js"></script>
56@ <script class="include" type="text/javascript"
57   src="js/jquery.dcjqaccordion.2.7.js"></script>
58 <script src="js/jquery.scrollTo.min.js"></script>
59 <script src="js/jquery.nicescroll.js" type="text/javascript"></script>
60@ <script type="text/javascript" language="javascript"
61   src="assets/advanced-dataTable/media/js/dataTables.js"></script>
62 <script type="text/javascript" src="assets/data-tables/DT_bootstrap.js"></script>
63 <script src="js/respond.min.js"></script>
64 <!-- common script for all pages-->
65 <script src="js/respond.min.js"></script>
66 <!-- script for this page only-->
67@ <script type="text/javascript" charset="utf-8">
68 $(document).ready(function() { $('#example').dataTable( { "aaSorting": [[ 4, "desc" ]] } ); });
69 </script>
70 </body>
71 </html>

```

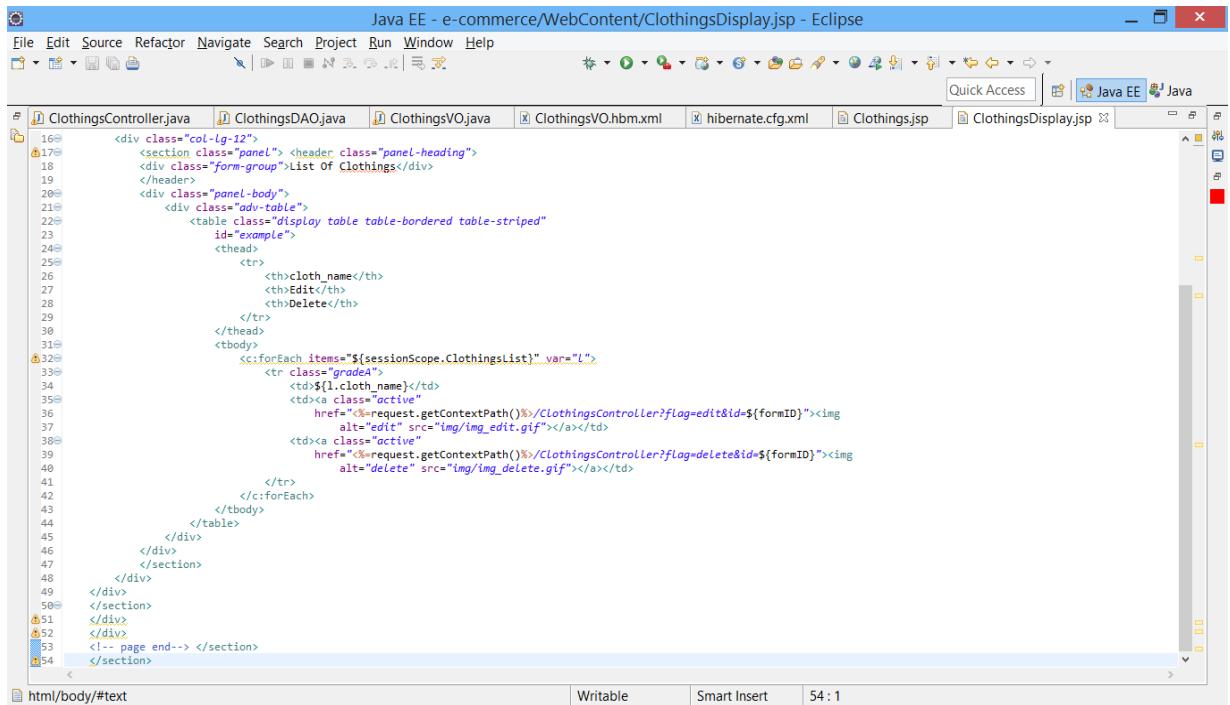
**Figure 6.43: Generated Required\_JSP (2)**

```

Java EE - e-commerce/WebContent/ClothingsDisplay.jsp - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
ClothingsController.java ClothingsDAO.java ClothingsVO.java ClothingsVO.hbm.xml hibernate.cfg.xml Clothings.jsp ClothingsDisplay.jsp
1 <%@ page language="java" contentType="text/html; charset=ISO-8859-1" %>
2 <!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
3 <html>
4   <head>
5     <meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
6     <title>Clothings Display</title>
7   </head>
8   <body>
9     <div class="col-lg-12">
10       <%@include file="header.jsp"%>
11       <%@include file="aside2.jsp"%>
12       <!-- main content start-->
13       <section id="main-content" ><section
14         class="wrapper site-min-height"> <!-- page start-->
15       <div class="row">
16         <div class="col-lg-12">
17           <section class="panel"> <header class="panel-heading">
18             <div class="form-group">List Of Clothings</div>
19           </header>
20           <div class="panel-body">
21             <div class="adv-table">
22               <table class="display table table-bordered table-striped"
23                 id="example">
24                 <thead>
25                   <tr>
26                     <th>cloth_name</th>
27                     <th>Edit</th>
28                     <th>Delete</th>
29                   </tr>
30                 </thead>
31                 <tbody>
32                   <:forEach items="${sessionScope.ClothingsList}" var="l">
33                     <tr class="gradeA">
34                       <td>${l.cloth_name}</td>
35                       <td><a class="active"
36                         href="<% request.getContextPath()%>/ClothingsController?flag=edit&id=${formID}"></a></td>
38                       <td><a class="active"
39                         href="<% request.getContextPath()%>/ClothingsController?flag=delete&id=${formID}"><img

```

**Figure 6.44: Generated Display\_JSP (1)**



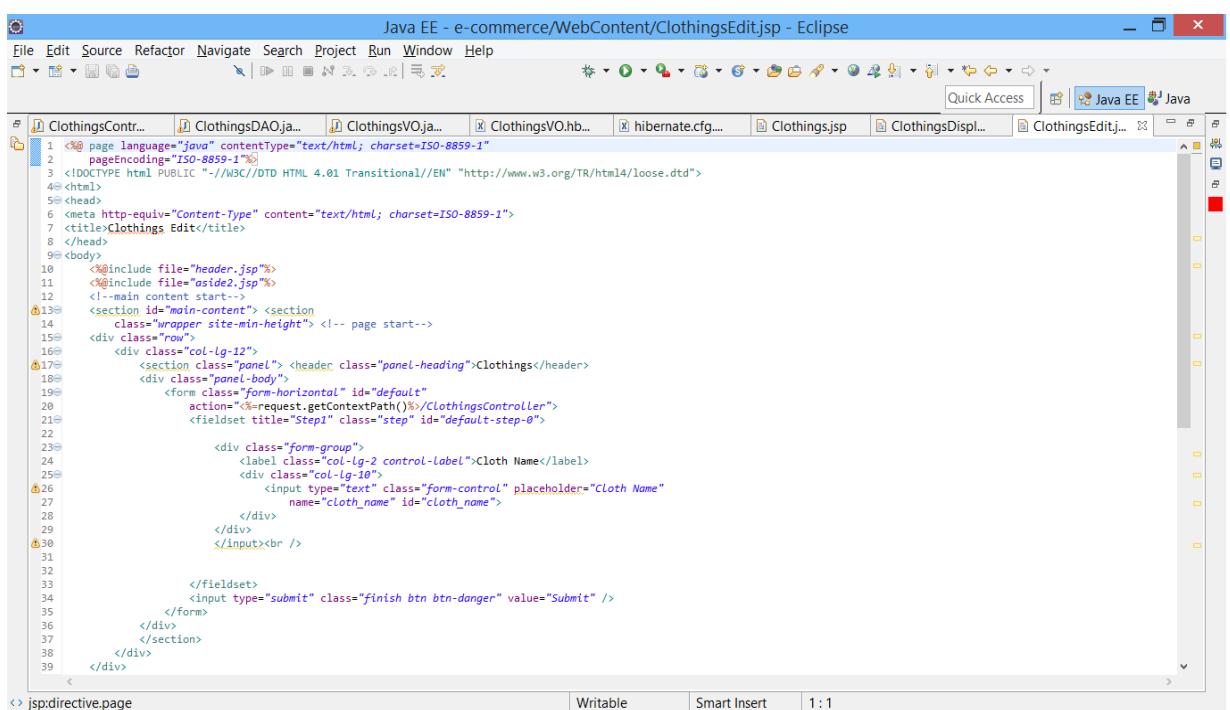
The screenshot shows the Eclipse IDE interface with the title "Java EE - e-commerce/WebContent/ClothingsDisplay.jsp - Eclipse". The code editor displays JSP code for displaying a list of cloths. The code includes imports for various Java classes and beans. It features a header section with a panel heading, a table for listing cloths with columns for name, edit, and delete, and a footer section with a copyright notice.

```

16<div class="col-lg-12">
17    <section class="panel"> header class="panel-heading">
18        <div class="form-group">list OF Clothings</div>
19    </header>
20    <div class="panel-body">
21        <div class="adv-table">
22            <table class="display table-bordered table-striped"
23                id="example">
24                <thead>
25                    <tr>
26                        <th>cloth_name</th>
27                        <th>Edit</th>
28                        <th>Delete</th>
29                    </tr>
30                </thead>
31                <tbody>
32                    <#foreach items="${sessionScope.ClothingsList}" var="l">
33                        <tr class="gradeA">
34                            <td>${l.cloth_name}</td>
35                            <td><a href="<@request.getContextPath()%/ClothingsController?flag=edit&id=${formID}>"></a></td>
36                            <td><a href="<@request.getContextPath()%/ClothingsController?flag=delete&id=${formID}>"></a></td>
37                        </tr>
38                    </#foreach>
39                </tbody>
40            </table>
41        </div>
42    </div>
43    </section>
44</div>
45</div>
46</div>
47<!-- page end --> </section>
48</section>
49</div>
50</div>
51</div>
52</div>
53<!-- page end --> </section>
54</div>

```

**Figure 6.45: Generated Display\_JSP (2)**



The screenshot shows the Eclipse IDE interface with the title "Java EE - e-commerce/WebContent/ClothingsEdit.jsp - Eclipse". The code editor displays JSP code for editing a cloth. The code includes imports for various Java classes and beans. It features a header section with a panel heading, a form for entering cloth details, and a footer section with a copyright notice.

```

1<%@ page language="java" contentType="text/html; charset=ISO-8859-1"%>
2<%@ pageEncoding="ISO-8859-1"%>
3<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
4<html>
5<head>
6<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
7<title>Clothings Edit</title>
8</head>
9<body>
10    <%@include file="header.jsp"%>
11    <%@include file="aside2.jsp"%>
12    <!-- main content start-->
13    <section id="main-content"> <section
14        class="wropper site-min-height"> <!-- page start-->
15    <div class="row">
16        <div class="col-lg-12">
17            <section class="panel"> <header class="panel-heading">Clothings</header>
18                <div class="panel-body">
19                    <form class="form-horizontal" id="default"
20                        action="<@request.getContextPath()%/ClothingsController"
21                        <fieldset title="Step1" class="step" id="default-step-0">
22
23                        <div class="form-group">
24                            <label class="col-lg-2 control-label">Cloth Name</label>
25                            <div class="col-lg-10">
26                                <input type="text" class="form-control" placeholder="Cloth Name"
27                                    name="cloth_name" id="cloth_name">
28                            </div>
29                        </div>
30                        </input><br />
31
32                        </fieldset>
33                        <input type="submit" class="finish btn btn-danger" value="Submit" />
34                    </form>
35                </div>
36            </section>
37        </div>
38    </div>
39</div>

```

**Figure 6.46: Generated Edit\_JSP(1)**

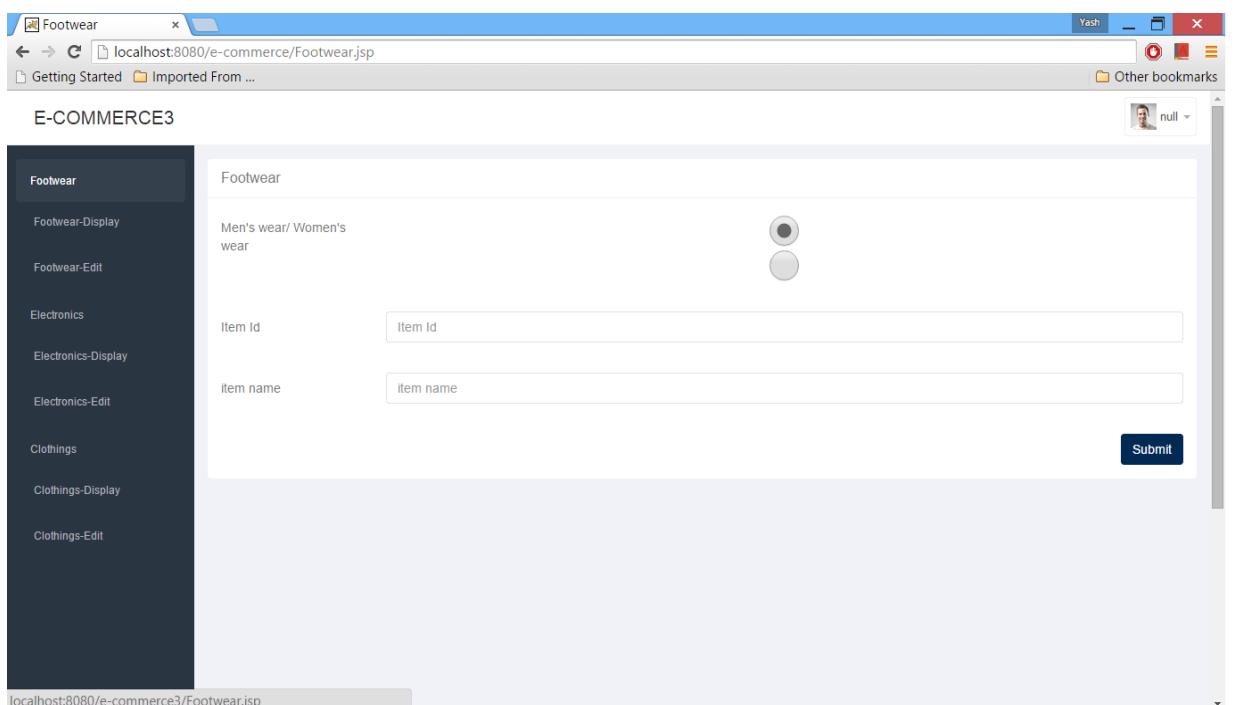
```

Java EE - e-commerce/WebContent/ClothingsEdit.jsp - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
ClothingsContr... ClothingsDAO.ja... ClothingsVO.ja... ClothingsVO.hb... hibernate.cfg... Clothings.jsp ClothingsDisp... ClothingsEdit.j...
34         <input type="submit" class="finish btn btn-danger" value="Submit" />
35     </form>
36   </section>
37   </div>
38 </div>
39 </div>
40 <!-- page end--> </section> </section>
41<!-- main content end-->
42<!-- footer start-->
43<!-- footer end-->
44<div class="text-center">
45   Clothings<a href="#" class="go-top" > <i class="fa fa-angle-up" >/<i></a>
46   <title>Clothings</title>
47 </div>
48 </footer>
49<!-- footer end-->
50</section>
51 <!-- js placed at the end of the document so the pages load faster -->
52 <!--<script src="js/jquery.js"></script>-->
53<script type="text/javascript" language="javascript"
54   src="assets/advanced-dataTable/media/js/jquery.js"></script>
55 <script src="js/bootstrap.min.js"></script>
56<script class="include" type="text/javascript"
57   src="js/jquery.djqaccordion.2.7.js"></script>
58 <script src="js/jquery.scrollTo.min.js"></script>
59 <script src="js/jquery.nicescroll.js" type="text/javascript"></script>
60<script type="text/javascript" language="javascript"
61   src="assets/advanced-dataTable/media/js/jquery.dataTables.js"></script>
62 <script type="text/javascript" src="assets/data-tables/DT_bootstrap.js"></script>
63 <script src="js/respond.min.js"></script>
64 <!-- common script for all pages-->
65 <script src="js/respond.min.js"></script>
66 <!-- script for this page only-->
67<script type="text/javascript" charset="utf-8">
68 $(document).ready(function() { $('#example').dataTable( { "aaSorting": [[ 4, "desc" ]] } ); });
69 </script>
70 </body>
71 </html>
72 |

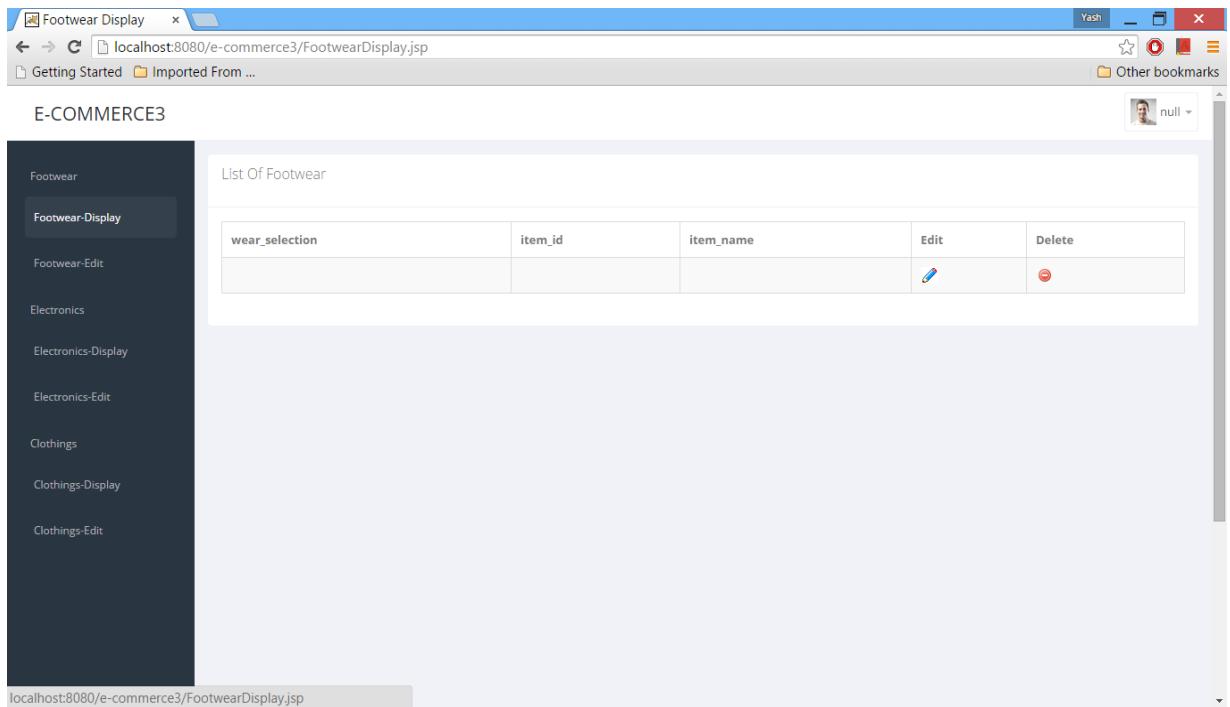
```

Writable Smart Insert 72:1

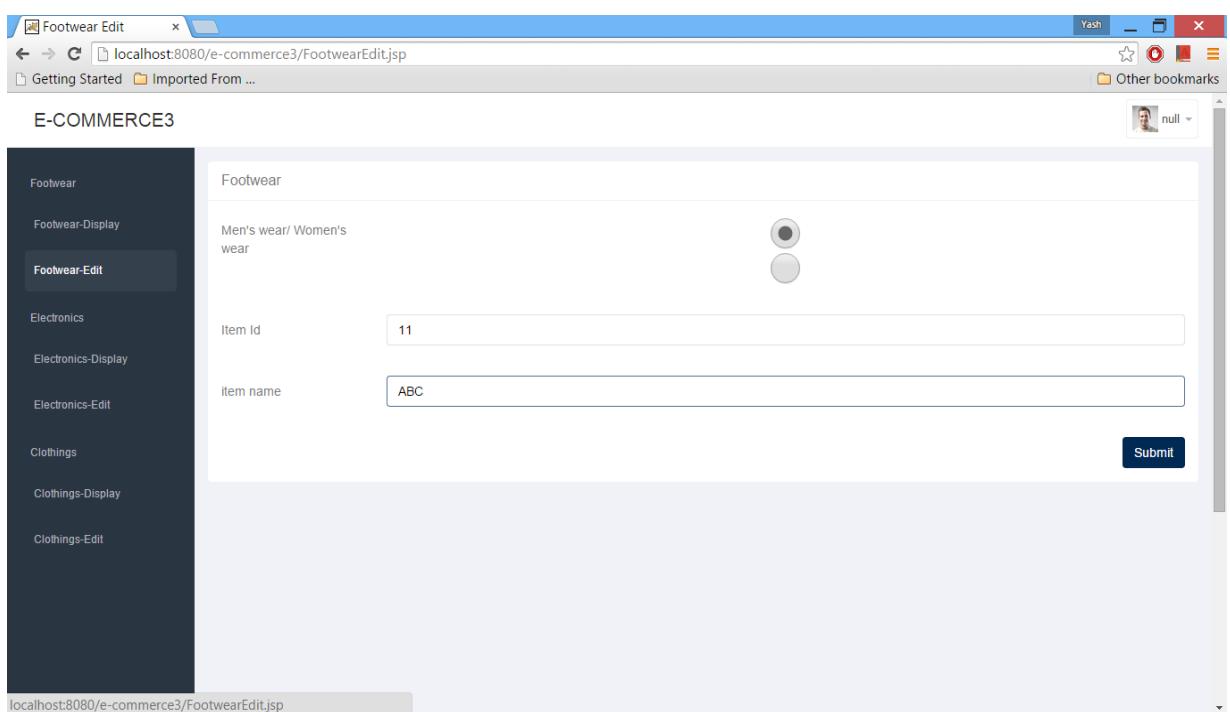
**Figure 6.47: Generated Edit\_JSP (2)**



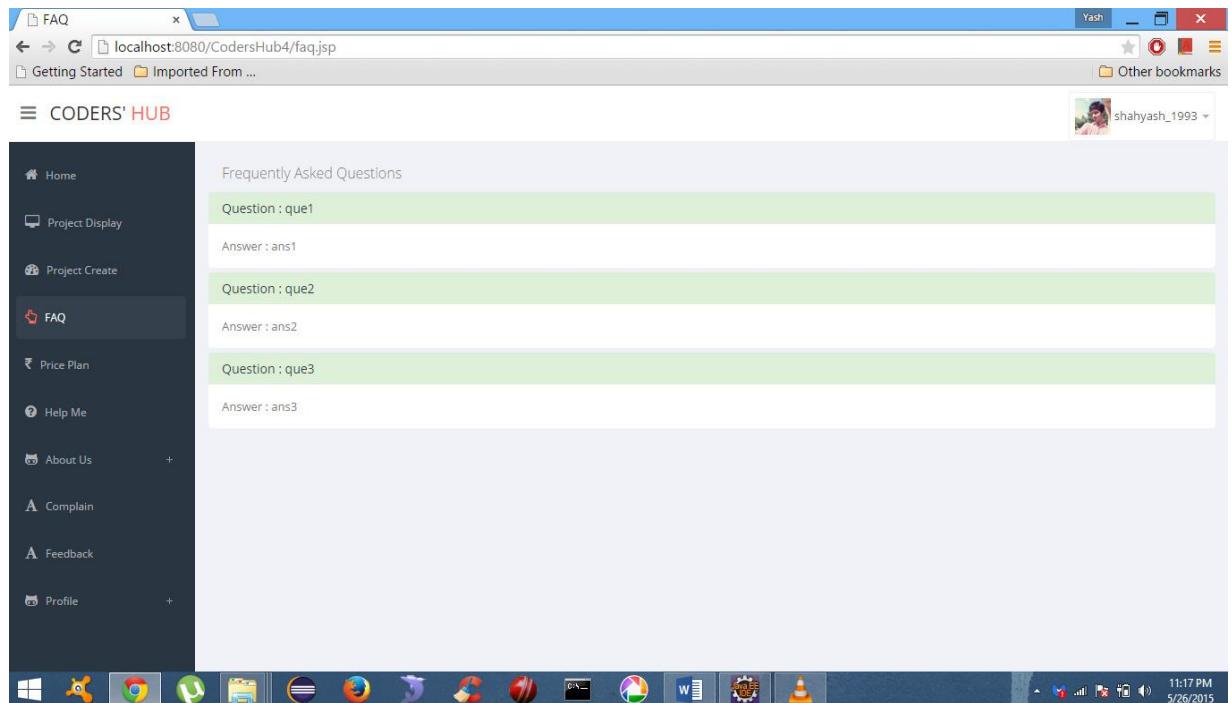
**Figure 6.48: Execution of Generated Project (1)**



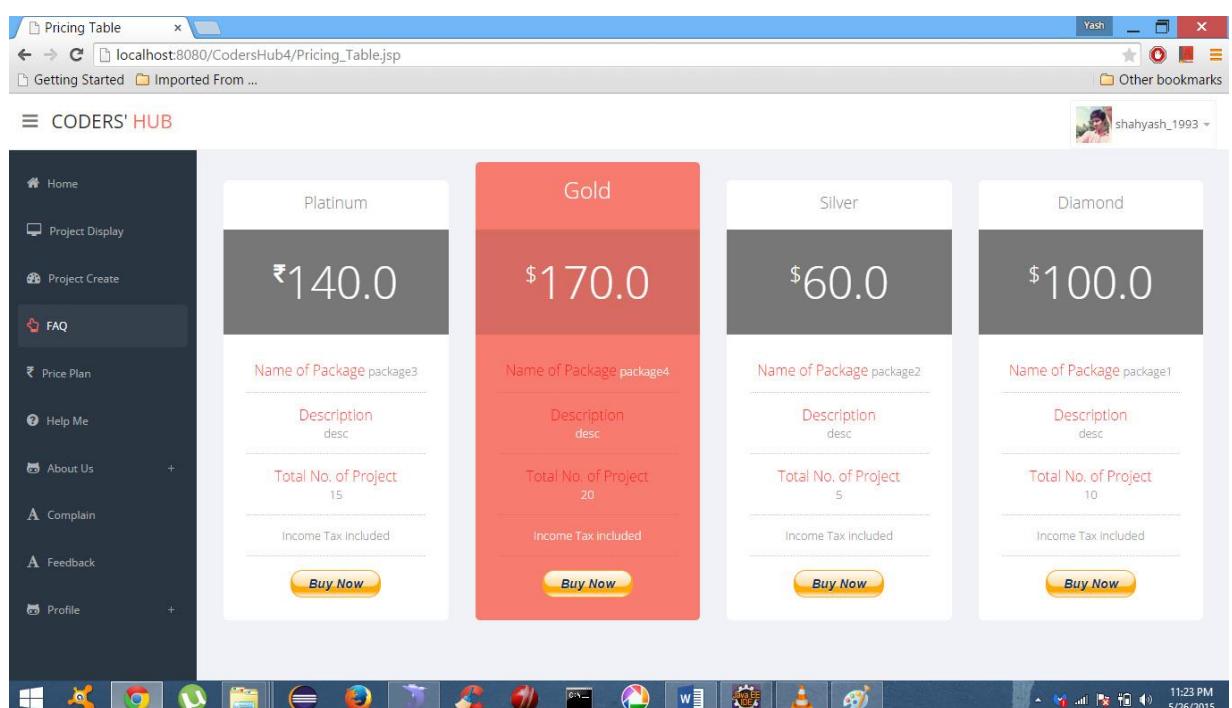
**Figure 6.49: Execution of Generated Project (2)**



**Figure 6.50: Execution of Generated Project (3)**



**Figure 6.51: Frequently Asked Questions**



**Figure 6.52: Pricing Table**

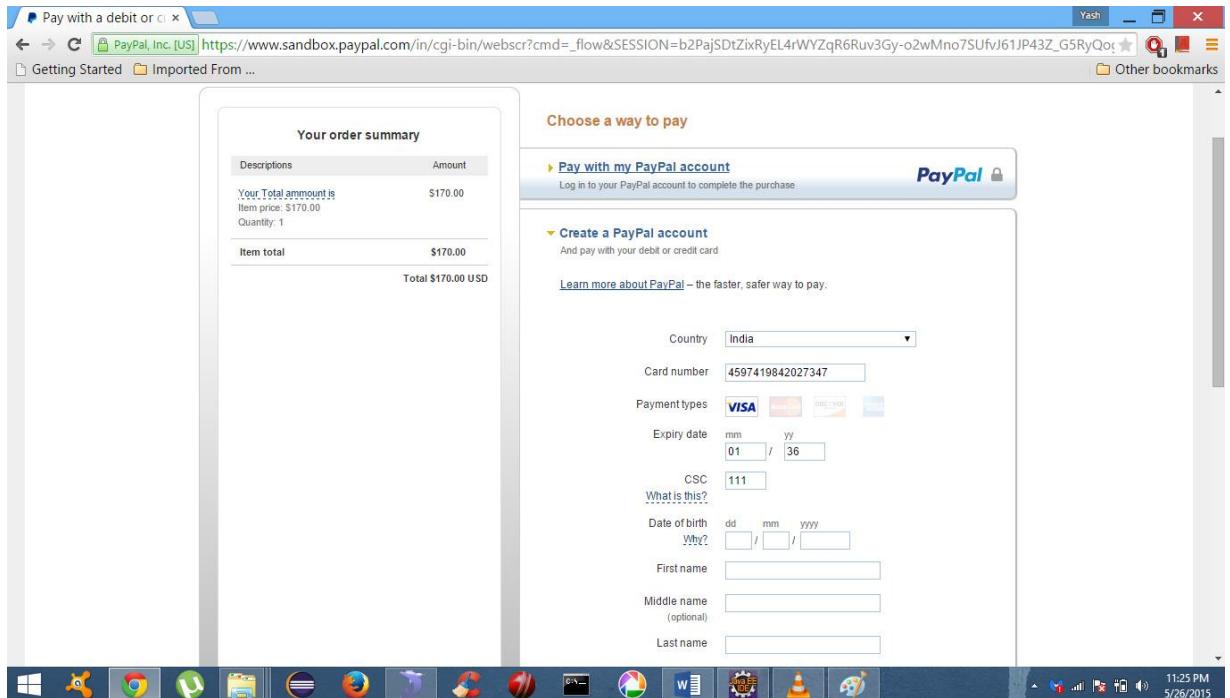


Figure 6.53: PayPal Payment

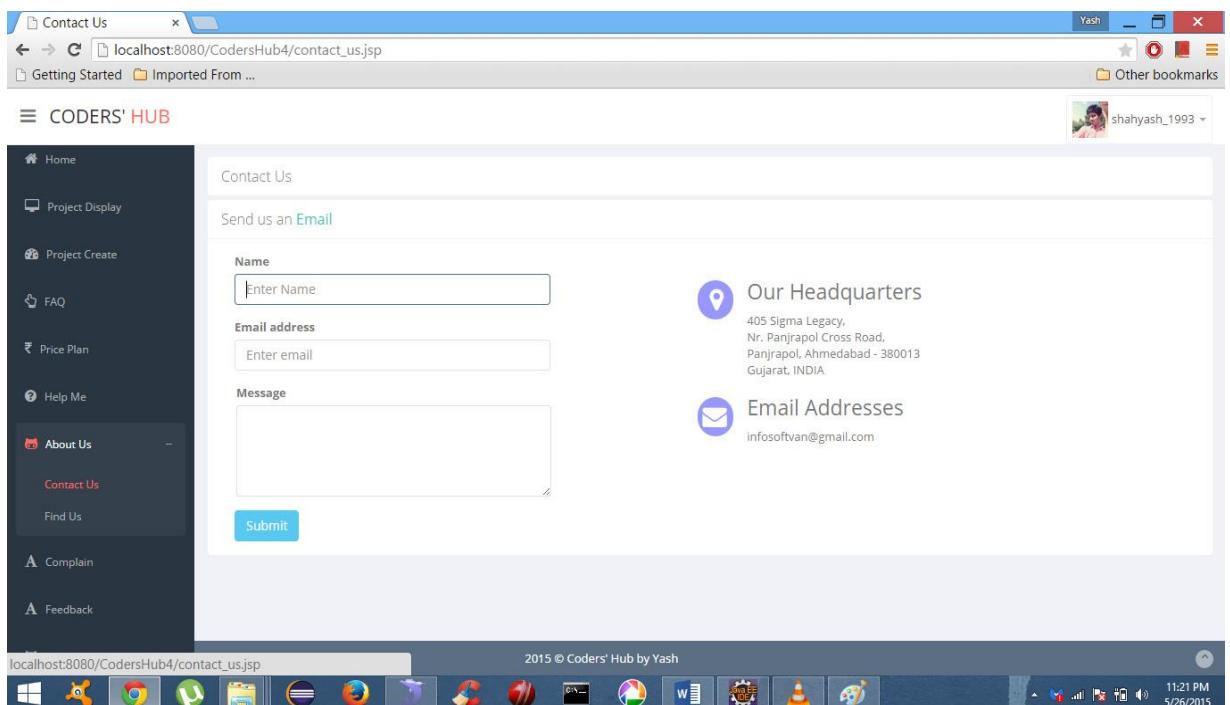


Figure 6.54: Contact Us

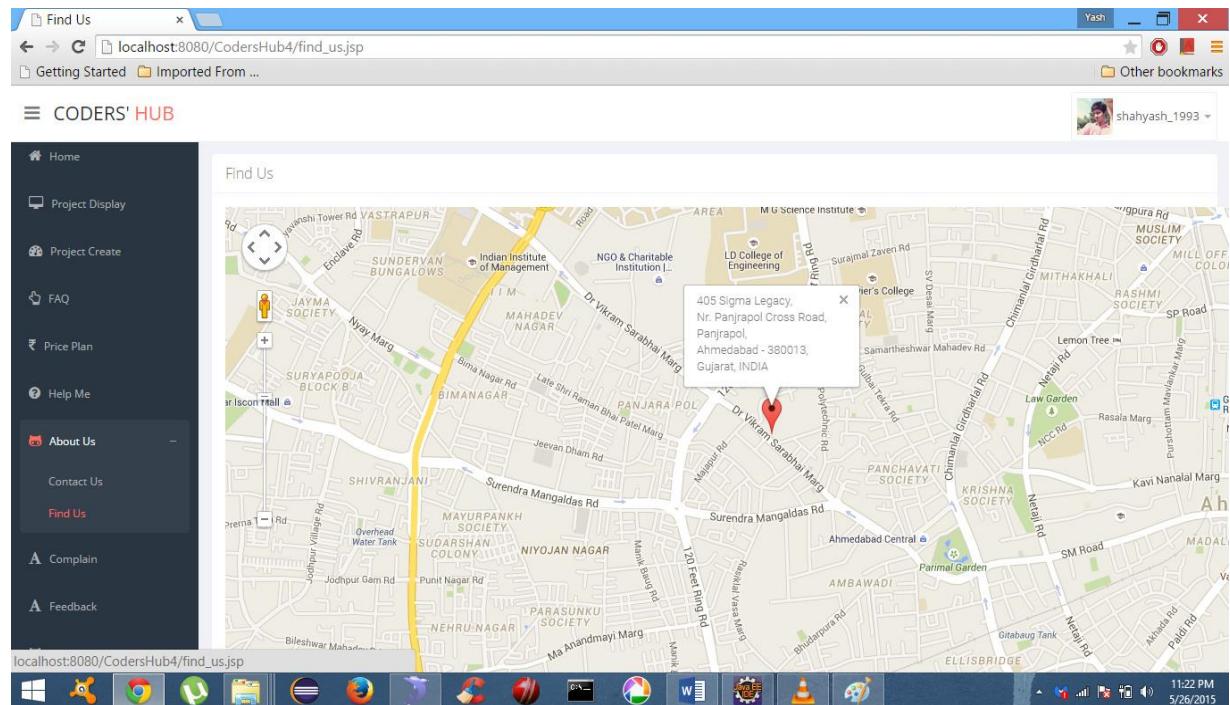


Figure 6.55: Find Us

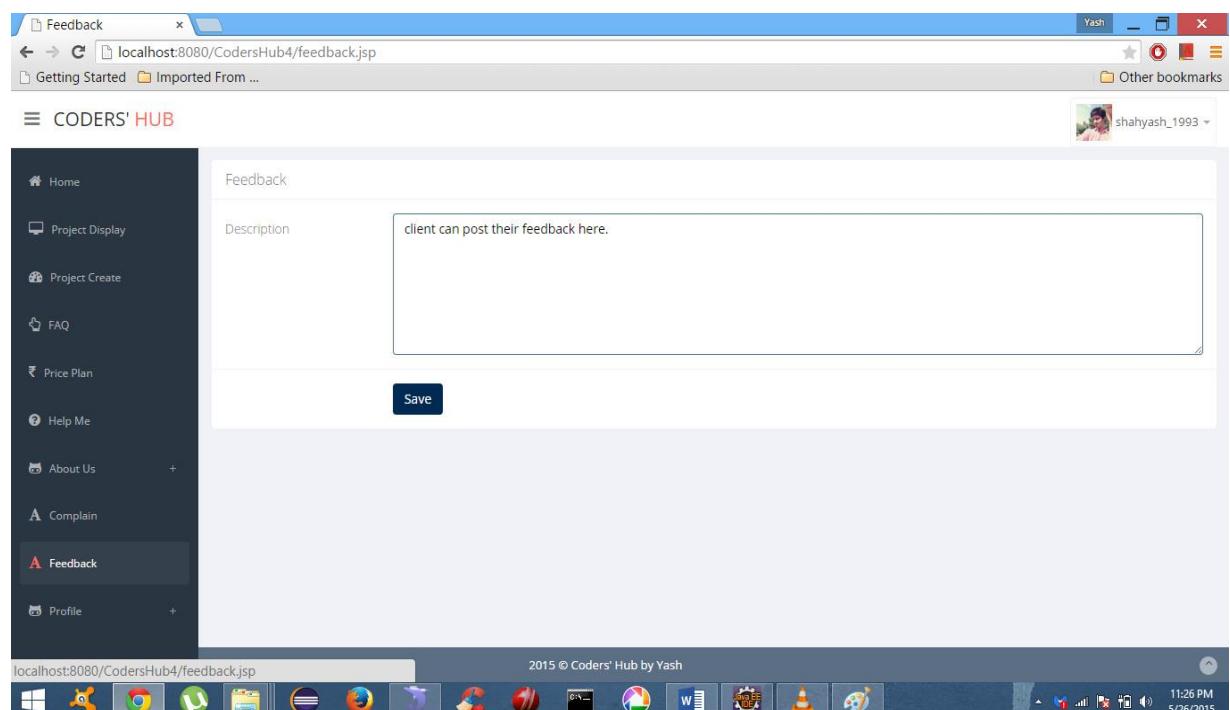
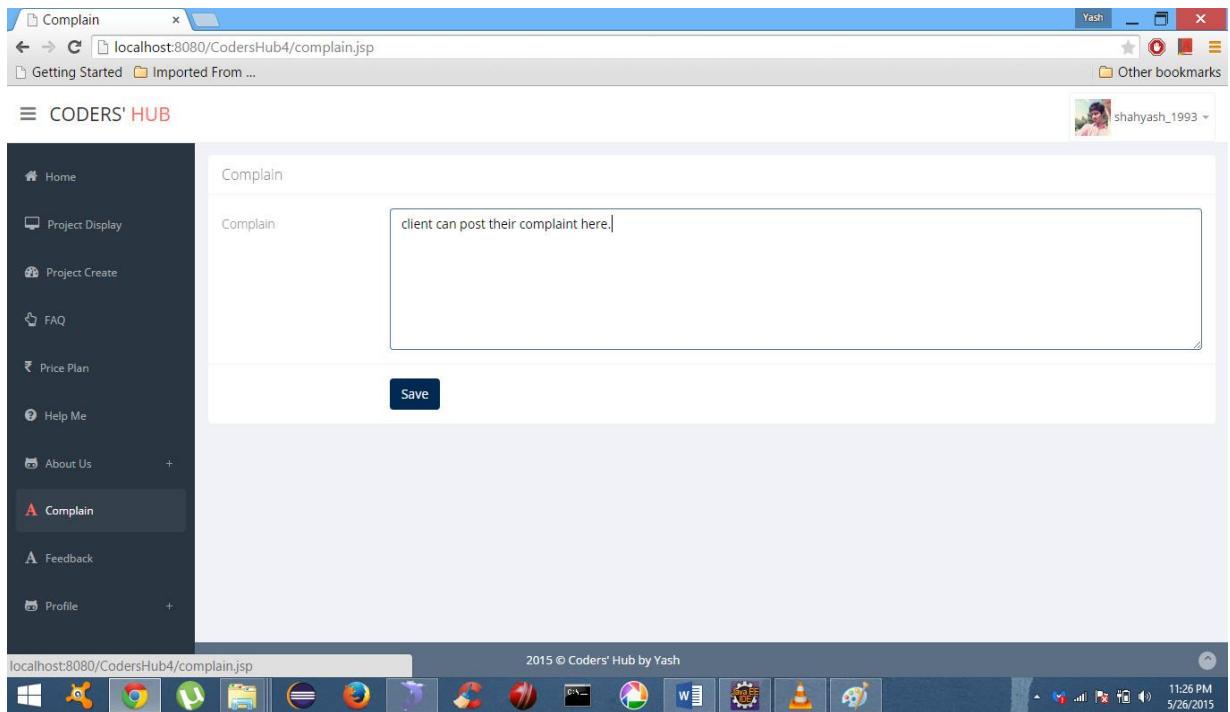
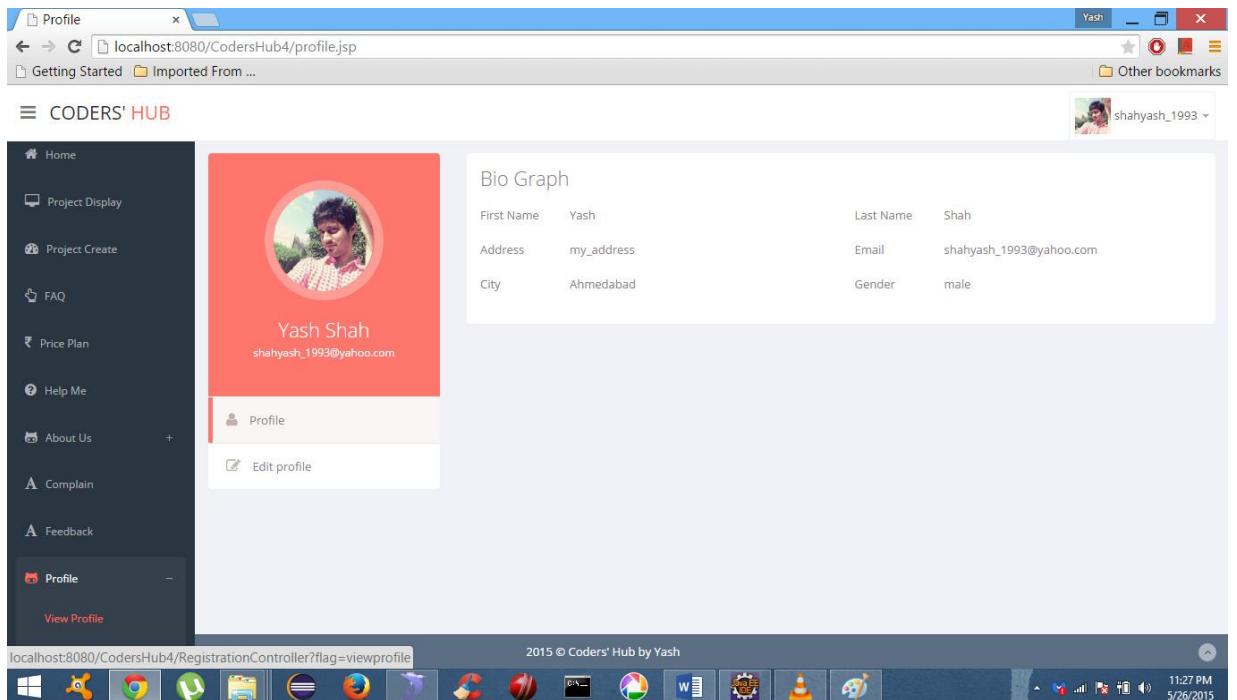


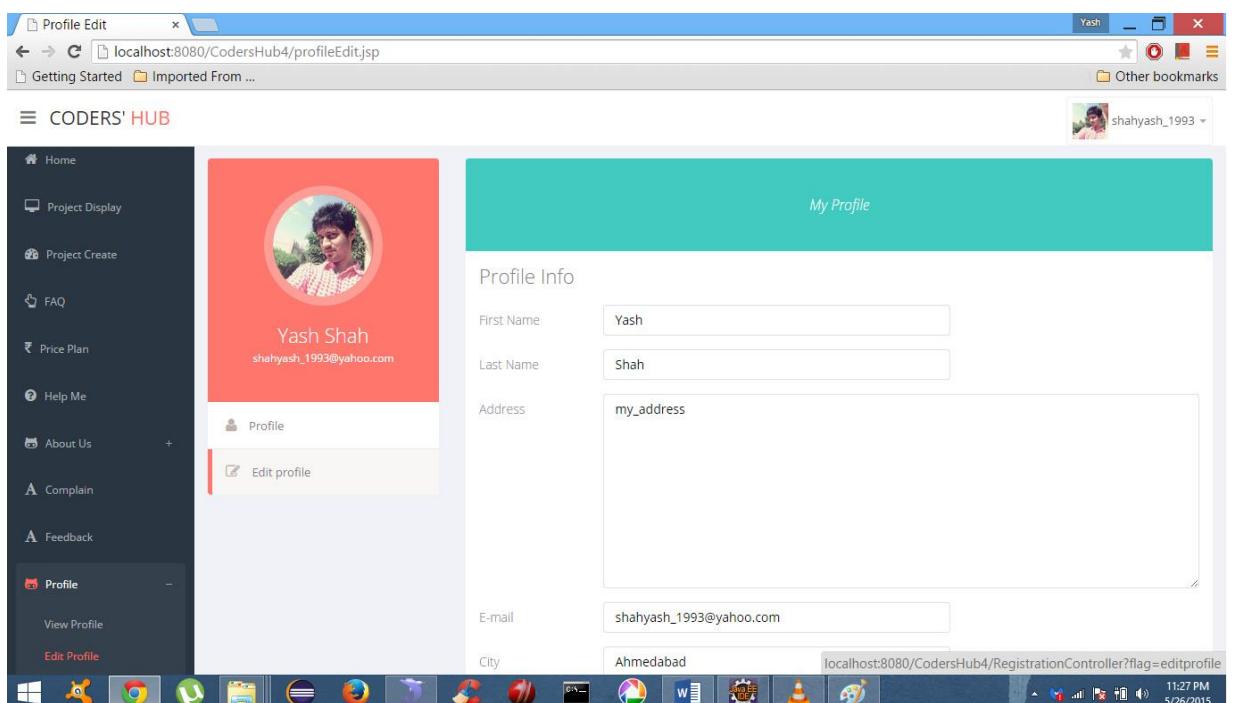
Figure 6.56: Feedback



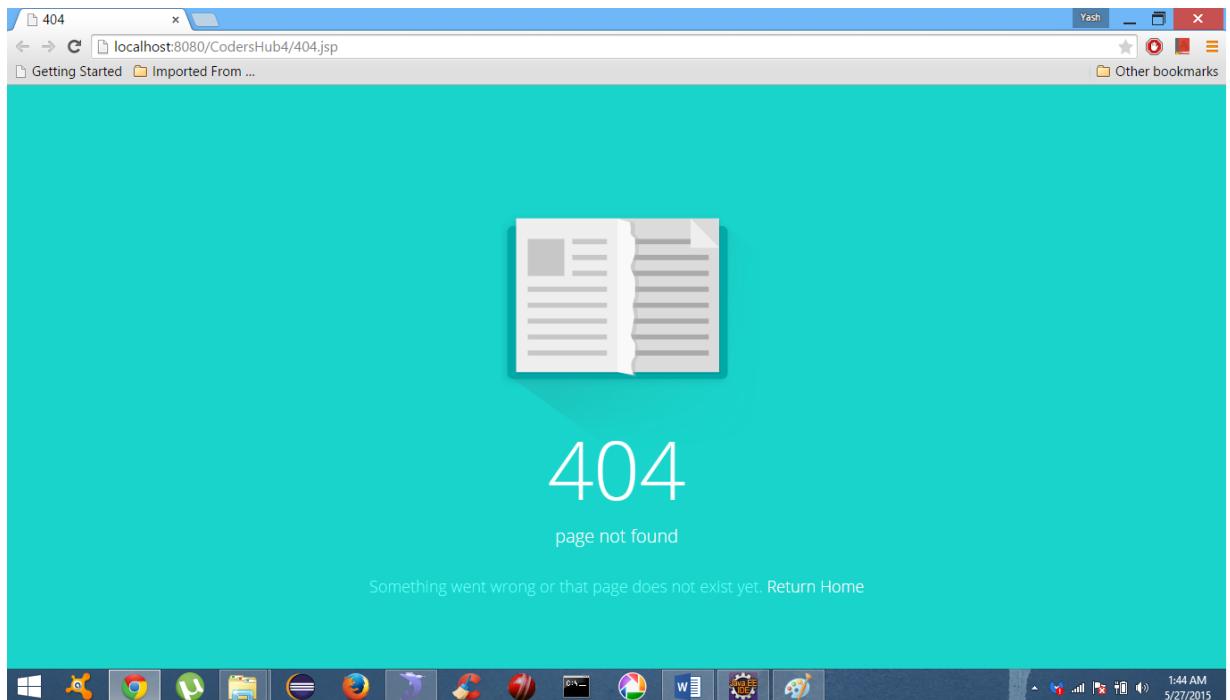
**Figure 6.57: Complaint**



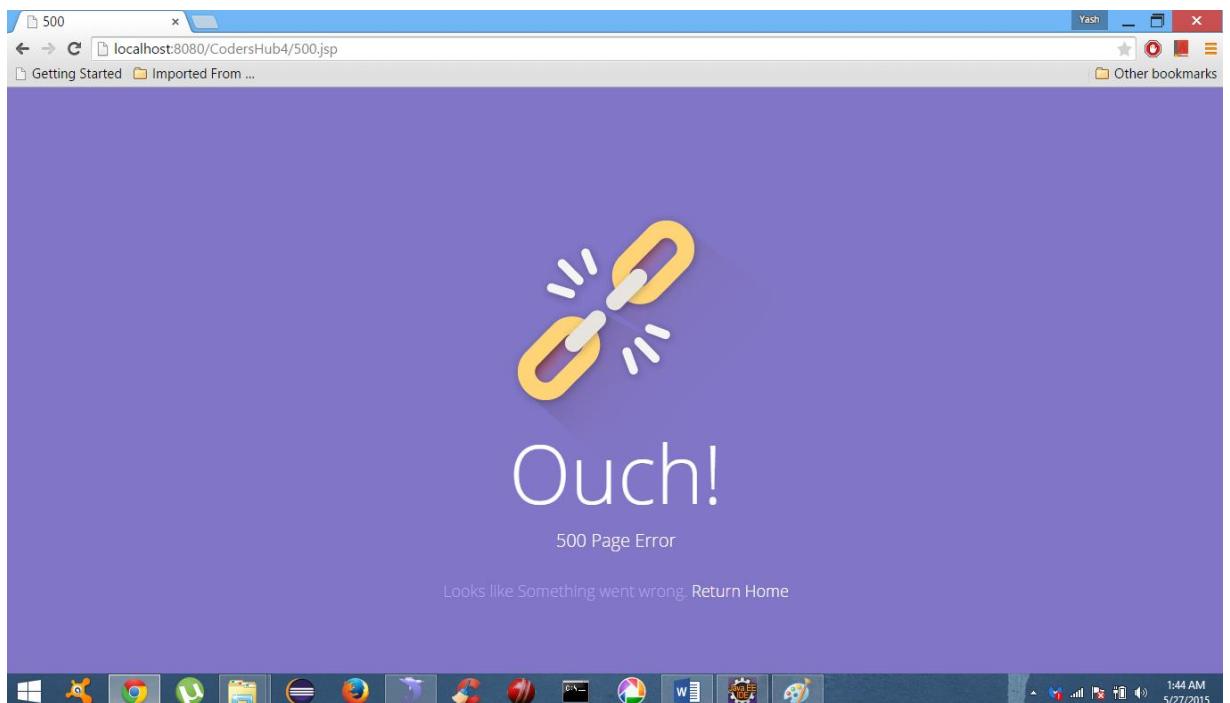
**Figure 6.58: User Profile**



**Figure 6.59: Edit User Profile**



**Figure 6.60: 404 Error Page**



**Figure 6.61: 500 Error Page**

## CHAPTER 7: TESTING

### **7.1 Testing Objectives**

Testing is a multi-step process, which involves exercising the program using data like the real data processed by the program. Examining the outputs of the program & looking for anomalies infer the existence of program defects or inadequacies.

The test activities are normally divided into Verification & Validation. Verification & Validation is a whole life-cycle process. It starts with requirements reviews & continues through design reviews & code inspections to product testing. There should be Verification & Validation activities at each stage of the software process during which defects in the program are normally discovered & the program must then be modified to correct these defects.

The difference between Verification & Validation can be expressed as:-

- Verification: Are we building the system correctly?
- Validation: Are we building the correct system?
- Verification involves checking that the software conforms to its specification i.e. the system meets it's specified functional and non-functional requirements.
- Validation checks whether the software meets the expectations of the customer i.e. whether the result really is, what the customer actually wanted. This focuses on customer satisfaction & is concerned with getting the specification & the result sight.

### **7.2 Testing Model (Testing Strategy)**

Testing strategies is a general approach to the testing process rather than a method of devising particular system or components tests. Different testing strategies may be adopted depending on the type of system to be tested and the development process used.

Basically there are two types of testing strategy:

### **Black box testing**

The testing technique that is going to be used in the project is black box testing. In black box testing the expected inputs to the system are applied and only the outputs are checked.

The working or the other parameters of the functionality are not reviewed or tested on the black box testing technique. There is a specific set of inputs for each and every module which is applied and for each set of inputs the result or the output is verified and if found as per the system working this testing is termed or result is declared as pass.

If the set of inputs that are provided to each module are not giving the outputs as per the expected results from the module then the result of that testing is to be declare failed.

Moreover the bottom up integration of the modules is applied herein so that each module can be verified at the initial stage and if it is found that the independent module is perfectly alright, only then it is going to be integrated with other related modules otherwise the module is checked for flaws and then if it satisfies all the specific requirements of the module, is integrated to other related modules to form and incorporate a system.

Black box testing attempts to find errors in the following categories:

- Incorrect or missing function.
- Interface errors.
- Errors in fetching external data.
- Behavior or performance errors.
- Initialization and termination errors.

## **White box testing**

The purpose of any security testing method is to ensure the robustness of a system in the face of malicious attacks or regular software failures. White box testing is performed based on the knowledge of how the system is implemented. White box testing includes analyzing data flow, control flow, information flow, coding practices, and exception and error handling within the system, to test the intended and unintended software behavior. White box testing can be performed to validate whether code implementation follows intended design, to validate implemented security functionality, and to uncover exploitable vulnerabilities.

White box testing requires access to the source code. Though white box testing can be performed any time in the life cycle after the code is developed, it is a good practice to perform white box testing during the unit testing phase.

White box testing requires knowing what makes software secure or insecure, how to think like an attacker, and how to use different testing tools and techniques. The first step in white box testing is to comprehend and analyze available design documentation, source code, and other relevant development artifacts, so knowing what makes software secure is a fundamental requirement. Second, to create tests that exploit software, a tester must think like an attacker. Third, to perform testing effectively, testers need to know the different tools and techniques available for white box testing. The three requirements do not work in isolation, but together.

As per the project white Box Testing is used. Model because independent paths of modules and loops are the cornerstones of the vast majority of all algorithms implemented in the software. There are five different classes of the loops, which can be defined as follows.

- Simple Loops.
- Nested Loops.
- Concatenated Loops.
- Unstructured Loops.
- Continuous Loops.

### 7.3 Test cases & Results

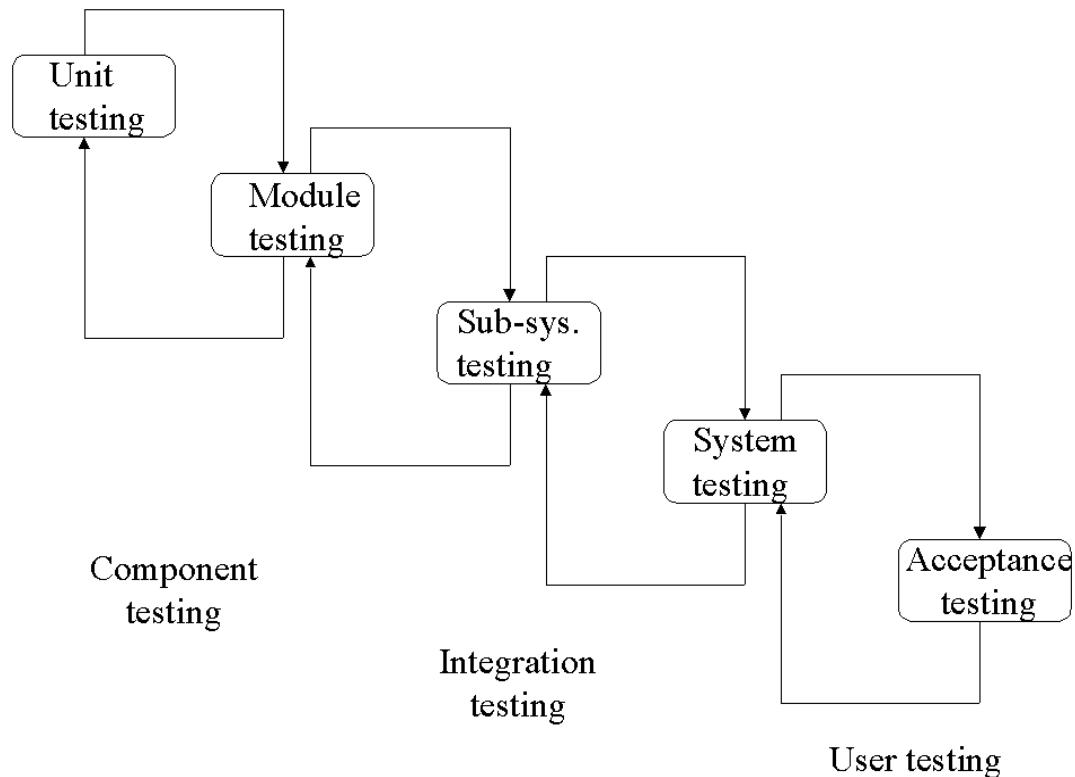
The stages in the testing process that we followed are:

- **Unit Testing:** This is the lowest level of testing that is conducted to remove syntax & logic errors from a single unit. Individual components are tested to ensure that they operate correctly. Each component is tested independently, without other system components.
- **Module testing:** A module is a collection of dependent components such as an object class, an abstract data type or some looser collection of procedures & functions. A module encapsulates related components, so can be tested without other system modules.
- **Sub-System testing:** This phase involves testing collections of modules, which have been integrated into sub-systems. These tests for problems that arise from component interactions. This testing should begin as soon as usable versions of some of the system components are available.
- **System Testing:** The sub-systems are integrated to make up the system. The system as a complete entity is tested over here. This process is concerned with finding errors that result from unanticipated interactions between sub-systems. It is also concerned with validating that the system meets its functional & non-functional requirements & testing the emergent system properties.
- **Acceptance testing:** This is the final stage in the testing process before the system is accepted for operational use. The system is tested with data supplied by the system customer rather than simulated test data. Acceptance testing may reveal errors & omissions in the system requirements definition because the real data exercise the system in different ways from the test data. It may also reveal requirements problems where the system's facilities do not really meet the user's needs or the system performance is unacceptable.

Component  
testing

Integration  
testing

User testing



**Figure 7.1: Accepting Testing**

**Table 7.1 Test cases**

TEST CASE ID	TEST CASE	INPUT	EXPECTED RESULT	ACTUAL RESULT
Login	Login to the System	Valid username & Password is entered.	The homepage is loaded depending On the user.	As per expected
Incorrect login-Detail	Incorrect username or password is Entered	Incorrect username or Password	Sorry, username or password is Incorrect.	As per expected
registration	Registration with s name, id, email id and password etc.	Registering with the name, id, email id and password etc.	It is not allowed. Prompts to use an another email id, name, id and Password.	As per expected
Project Details & Form Details	Check validations by attempting invalid inputs	Empty form name, Empty control name, and control type	Form name has to Be mentioned. Control type has to Be selected.	As per expected
Update Project	Check project Validation.	Empty form name, Empty control name, and control type	Form name has to Be mentioned. Control type has to Be selected. Form name has to	As per expected
Payment Module	Check developer working on single Project.	Account number must Not be empty.	Developer you are Pay for the project.	As per expected

## **CHAPTER 8: CONCLUSION AND FUTURE WORK**

### **8.1 Conclusion**

After envisaging the future aspects I can conclude that this project is going to give efficient & more effective results in comparison to the current applications, because it allows the Developer to choose his/her requirements and give the output instantly if he/she needs it immediately. Other than that you can choose services multiple times & give the Developer possible best database management which is appropriate according to the Developer.

### **8.2 Future Work**

- Drag & Drop functionality will be provided for component selection.
- Development in several other framework will be provided. i.e., Spring.
- Better Performance.
- Several new Payment Modules.
- More Plug-ins.
- Improvement in the Dashboard.

## **REFERENCES & BIBLIOGRAPHY**

### **Books:**

- Head First JSP and Servlets By Kathy sierra
- The Complete Reference JAVA2 By Herbert Scheldt
- Software Engineering-A practitioner's approach By Roger S. Pressman

### **Websites:**

- [www.roseindia.net](http://www.roseindia.net)
- [www.java2s.com](http://www.java2s.com)
- <http://www.w3schools.com/js/default.asp>
- <http://www.w3schools.com/sql>
- <http://www.javatpoint.com/example-of-sending-email-using-java-mail-api>
- <http://www.mkyong.com/tutorials/java-io-tutorials/>
- <http://www.tutorialspoint.com/hibernate/>
- [http://www.tutorialspoint.com/javamail\\_api/](http://www.tutorialspoint.com/javamail_api/)
- <http://www.avajava.com/tutorials/lessons/how-do-i-zip-a-directory-and-all-its-contents.html>

**Periodic Progress Report : First PPR****Project** Coders' Hub

:

**Status** : Reviewed (Freeze)**What Progress you have made in the Project ?**

I have completed several Admin-Side functionalities. I have used Simple CRUD operations of hibernate framework of j2ee to perform certain functionalities of Admin-side like Project Creation, Project Display, Project Updation, Project Deletion.

**What challenge you have faced ?**

Setting data into sessions and retrieving them from session by using JSTL was a bit difficult or kind of tricky for me at the beginning.

**What support you need ?**

basics of JSTL and Hibernate Framework.

**Which literature you have referred ?**

I have referred "www.javapoint.com" & "www.tutorialspoint.com" to understand some of the basics of Hibernate Framework and JSTL.

**Comment by Internal Guide :**

I have Gone through it and student has completed it

**Periodic Progress Report : Second PPR****Project** Coders' Hub

:

**Status** : Reviewed (Freeze)**What Progress you have made in the Project ?**

The Admin-side of my project is now completed and have started working on User-side. I have implemented Several APIs in my project. Mailing Feature is also completed.

**What challenge you have faced ?**

Using the APIs was the toughest task for me.

**What support you need ?**

basics of How to implement APIs in the Project.

**Which literature you have referred ?**

I have referred several articles on "developers.google.com".

**Comment by Internal Guide :**

students have now grip over using APIs

**Periodic Progress Report : Third PPR****Project** Coders' Hub

:

**Status** : Reviewed (Freeze)**What Progress you have made in the Project ?**

I have completed several User-Side Functionalities as such, Creation of 'Dao files', Creation of 'Vo files', Creation of 'vo.hbm files' and creation of hibernate-configuration file for user based on their requirement.

**What challenge you have faced ?**

"Create Form" was the toughest task, because the user may want n numbers of components in his/her Form, so adding a new button every time he/she clicks on the button of my "Create Form" page and saving every single data to the database was really hard to implement.

**What support you need ?**

basics of Java I/O.

**Which literature you have referred ?**

I have referred "<http://www.tutorialspoint.com>" to have the better understanding of the Java I/O.

**Comment by Internal Guide :**

student has done work satisfactorily.

**Periodic Progress Report : Forth PPR****Project** Coders' Hub

:

**Status** : Reviewed (Freeze)**What Progress you have made in the Project ?**

I have completed almost all the User-Side functionalities such as, Creation of 'Controller files', Creation of 'JSP files', Making the 'Zip file' of the whole project.

**What challenge you have faced ?**

It was very hard to create the 'Controller file' & 'JSP file' with the help of the data given by the user.

**What support you need ?**

basics of Java I/O.

**Which literature you have referred ?**

I have referred "<http://www.tutorialspoint.com>" to have the better understanding of the Java I/O.

**Comment by Internal Guide :**

Finally student was able to finish 'Controller file' & 'JSP file' with the help of the data given by the user.

# GTU Innovation Council

## Patent Drafting Exercise (PDE)

**FORM 1**  
**THE PATENTS ACT 1970**  
**(39 OF 1970)**  
**&**  
**THE PATENTS RULES, 2003**  
**APPLICATION FOR GRANT OF PATENT**

**(FOR OFFICE USE ONLY)**

**Application No:**  
**Filing Date:**  
**Amount of Fee paid:**  
**CBR No:** \_\_\_\_\_

### 1. Applicant(s) :

ID	Name	Nationality	Address	Mobile No.	Email
1	Yash Paragkumar Shah	Indian	Computer Engineering , Kalol Institute Of Technology & Research Centre, Kalol , Gujarat Technological University.	9409253400	shahyash_1993 @yahoo.com

### 2. Inventor(s):

ID	Name	Nationality	Address	Mobile No.	Email
1	Yash Paragkumar Shah	Indian	Computer Engineering , Kalol Institute Of Technology & Research Centre, Kalol , Gujarat Technological University.	9409253400	shahyash_1993 @yahoo.com

### 3. Title of Invention/Project:

Coders' Hub

### 4. Address for correspondence of applicant/authorized patent agent in india

**Name:** Yash Paragkumar Shah

**Address:** Computer Engineering , Kalol Institute Of Technology & Research Centre, Kalol , Gujarat Technological University.

**Mobile:** 9409253400

**Email ID:** shahyash\_1993@yahoo.com

**5. Priority particulars of the application(S) field in convention country**

Country	Application No.	Filing Date	Name of the Applicant	Title of the Invention
N/A	N/A	N/A	N/A	N/A

**6. Particulars for filing patent co-operation treaty (pct) national phase Application**

International application number	International filing date as allotted by the receiving office
N/A	N/A

**7. Particulars for filing divisional application**

Original(First) Application Number	Date of filing of Original (first) application
N/A	N/A

**8. Particulars for filing patent of addition**

Original(First) Application Number	Date of filing of Original (first) application
N/A	N/A

**9. DECLARATIONS:****(i) Declaration by the inventor(s)**

I/We, the above named inventor(s) is/are true & first inventor(s) for this invention and declare that the applicant(s).

herein is/are my/our assignee or legal representative.

**Date :** 19 - May - 2015

Name

Signature &amp; Date

1 Yash Paragkumar  
Shah

---

**(ii) Declaration by the applicant(s) in the convention country**

I/We, the applicant (s) in the convention country declare that the applicant(s) herein is/are my/our assignee or legal representative.applicant(s)

**(iii) Declaration by the applicant(s)**

I/We, the applicant(s) hereby declare(s) that:-

- I am/We in possession of the above mentioned invention.
- The provisional/complete specification relating to the invention is filed with this application.
- The invention as disclosed in the specification uses the biological material from India and the necessary permission from the competent authority shall be submitted by me/us before the grant of patent to me/us.
- There is no lawful ground of objection to the grant of the patent to me/us.
- I am/we are the assignee or the legal representative of true & first inventors.
- The application or each of the application, particulars of each are given in the para 5 was the first application in the convention country/countries in respect of my/our invention.
- The application or each of the application, particulars of each are given in the para 5 was the first application in the convention country/countries in respect of my/our invention.
- I/we claim the priority from the above mentioned applications(s) filed in the convention country/countries & state that no application for protection in respect of invention had been made in a convention country before that date by me/us or by any person
- My/Our application in India is based on international application under Patent Cooperation Treaty (PCT) as mentioned in para 6
- The application is divided out of my/our application(s) particulars of which are given in para 7 and pray that this application may be treated as deemed to have been filed on \_\_\_\_\_ under section 16 of the Act.
- The said invention is an improvement in or modification of the invention particulars of which are given in para 8.

#### 10. Following are the attachments with the application:

- (a) Provisional specification/Complete specification
- (b) Complete specification (In confirmation with the international application) / as amended before International Preliminary Examination Authority (IPEA), as applicable (2 copies), No. of pages.. claims.....
- (c) Drawings (In confirmation with the international application)/as amended before the International Preliminary Examination Authority (IPEA), as applicable (2 copies), No. of sheets....
- (d) Priority documents
- (e) Translations of priority documents/specification/international search reports
- (f) Statement and undertaking on Form 3
- (g) Power of Authority
- (h) Declaration of inventorship on Form 5
- (i) Sequence listing in electronic Form
- (j) ..... Fees Rs.XXX in Cash /Cheque/Bank Draft bearing No.XXX Date: XXX on Bank.

I/We hereby declare that to the best of my /our knowledge, information and belief the facts and matters stated herein are correct and I/We request that a patent may be granted to me/us for the said invention.

Dated this 19 day of May , 2015

Name

Signature & Date

1 Yash Paragkumar  
Shah

---



**FORM 2**  
**THE PATENTS ACT, 1970**  
**(39 OF 1970)**  
**&**  
**THE PATENTS RULES, 2003**  
**PROVISIONAL SPECIFICATION**

**1. Title of the project/invention :**

Coders' Hub



**2. Applicant(s) :**

**Yash Paragkumar Shah , ( Indian )**

**Address :Computer Engineering , Kalol Institute Of Technology & Research Centre, Kalol , Gujarat Technology University.**

**3. Preamble to the description :**

The following specification describes the invention.

**4. Description :****a. Field of Application / Project / Invention :**

This project automatically creates the basic project based on the client's requirements code in java.

**b. Prior Art / Background of the Invention / References :**

The model used in the project is Spiral model.

Spiral model deals with the first round of completion of all the Software Development Life Cycles' steps and then it performs again the risk analysis.

All these steps are repeated.

**c. Summary of the Invention/Project :**

Main objective of this project is to generate code online based on the requirement provided by user by filling the form.

The Title of the project is CODERS' Hub.

It's specially developed for the developers who have to spend a lot of time in creating simple codes in Java.

This project envisages bridging the gap between developer and the customer.

Advantage of online-code-development is that it allows people to develop their basic project code online platform at their workplace only.

**d. Objects of the Invention/Project :**

Automatic Basic Project Creation

Easy Management of Client's profile

Clients can download Projects

Develop project as per user's requirement

Easy Upgradation of project

Easy and accurate tracking of project.

Reduce Complexity

Better management of project

Customer Satisfaction

Integration of Paypal.

**e. Drawing(s) :****f. Description of the Invention**

"Coders Hub" is a WEB-portal which provides the CLIENT, which in my case is The Developer, the WEB-SITE having all the basic Functionalities required by the developer in-Built. 'HUB' means the "center-point", so the "CODERS' HUB" is one kind of a center-point for every single coder, if they want their project to be done very efficiently and quickly by just filling up the required FORM in addition to that, It consumes LESS TIME because here developers don't have to write the whole code for the Project and at the end it provides the facility to DIRECTLY DOWNLOAD the project, too. After filling up the Required FORM, the developer can download the whole project fulfilling his requirements, to his/her system & for further use, he can easily edit the project by using JAVA-integrated development environment. i.e., Eclipse. The Main Objective of "Coders' Hub" is to minimize the work efforts of the Developer by Generating code which fulfills the BASIC Requirements of the Developer.

### **g. Examples**

### **h. Unique Features of the Project**

Project is created automatically  
Database files are created automatically  
zip file of project can be downloaded  
email functionality  
online payment can be done using Paypal.

### **5. Date & Signature :**

Date : 18 - May - 2015

\_\_\_\_\_  
Sign and Date  
Yash Paragkumar  
Shah

### **6. Abstract of the project / invention :**

online-code-development allows people to develop code online at their workplace only, whether it is their home or office.  
It Develops such a code that they don't have to worry about how it is built.  
The User can download the whole project at the end to his PC/laptop only.  
The Internet is open 24x7, so Client doesn't have to hurry or worry about clock ticking.  
Administrators can add extra features and functionalities to this web-portal.

**Drawing Attachments :**



**Note :** This is just a mock Patent Drafting Exercise (PDE) for semester 8, BE students of GTU.  
These documents are not to be submitted with any patent office.

**FORM 3**  
**THE PATENTS ACT, 1970**  
**(39 OF 1970)**  
**&**  
**THE PATENTS RULES, 2003**  
**STATEMENT AND UNDERTAKING UNDER SECTION 8**

**1. Declaration :**

I/We, Yash Paragkumar Shah ,

**2. Name, Address and Nationality of the joint Applicant :**

Yash Paragkumar Shah ( Indian )  
 Address : Computer Engineering , Kalol Institute Of Technology & Centre, Kalol , Gujarat Technologycal University.

Here by declare:

- (i) that I/We have not made any application for the same/substantially the same invention outside India.
- (ii) that the right in the application(s) has/have been assigned to,

Name of the Country	Date of Application	Application Number	Status of the Application	Date of Publication	Date of Grant
N/A	N/A	N/A	N/A	N/A	N/A

- (iii) that I/We undertake that up to the date of grant of patent by the Controller , I/We would keep him inform in writing the details regarding corresponding application(s) for patents filed outside India within 3 months from the date of filing of such application.

Dated this 18 day of May , 2015.

**3. Signature of Applicants :**

\_\_\_\_\_  
 Sign and Date  
 Yash Paragkumar Shah

To  
 The Controller of Patent  
 The Patent Office, at Mumbai.