PROJECT ON

PASSPORT SIMULATION SYSTEM

Submitted for partial fulfillment of the award of B.Tech degree in

COMPUTER AIDED SOFTWARE ENGINEERING (CASE) TOOLS LAB

in III/IV B.Tech IT – Sem 2

By

G.Siva Rama Krishna	**316126511021**
P.Sai SriValli	**316126511035**
D.Ranjith	**316126511014**
B.Ajaey	**316126511005**

ANIL NEERUKONDA INSTITUTE OF TECHNOLOGY AND SCIENCES

(Sangivalasa-531162, Bheemunipatnam Mandal, Visakhapatnam Dist.)

(2018-2019)

2018-2019

Department of Information Technology

Anil Neerukonda Institute of Technology & Sciences

Sangivalasa-531162

CERTIFICATE

This is to certify that the mini project work entitled **"PASSPORT SIMULATION SYSTEM"** is a bonafide work carried out by G.SivaRamaKrishna, P.Sai SriValli, D.Ranjith, B.Ajaey bearing University register numbers **316126511021, 316126511035, 316126511014, 316126511005,** respectively in COMPUTER AIDED SOFTWARE ENGINEERING(CASE)TOOLS LAB in III/IV B.Tech, 2st semester in **Information Technology** during the year **2018-2019**. It is certified that all corrections/suggestions indicated for internal assessment have been incorporated in this report. The mini project has been approved as it satisfies the academic requirements with respect to CASE TOOLS LAB.

Lab Internal

Mrs.B.Meena
Assistant Professor, Department of IT

Head of the Department

Prof. Suresh Chittineni
Department of Information Technology

TABLE OF CONTENTS:

LIST OF FIGURES

DECLARATION

We G. Siva Rama Krishna, P.Sai SriValli, D.Ranjith, B.Ajaey bearing the register numbers 316126511021,316126511035,316126511014 and 316126511005 **r**espectively do hereby declare that this mini-project work entitled "**PASSPORT SIMULATION SYSTEM**" was carried out by us under the able guidance of **B.Meena** ,**Assistant Professor** Department of Information Technology, Anil Neerukonda Institute of Technology and Sciences, Sangivalasa. This project work is submitted as a part of **Computer Aided Software Engineering Lab** (CASE TOOLS lab) to the department of **Information Technology** during the academic year 2018-2019.

G.SivaRama Krishna	316126511021
P.Sai SriValli	316126511035
D.Ranjith	316126511014
B.Ajaey	316126511005

ABSTRACT

Passport Simulation System is used in the effective dispatch of passport to all of the applicants. This system adopts a comprehensive approach to minimize the manual work and schedule resources, time in a cogent manner. The core of the system is to get the online registration form (with details such as name, address etc.,) filled by the applicant whose testament is verified for its genuineness by the Passport Simulation System with respect to the already existing information in the database. This forms the first and foremost step in the processing of passport application. After the first round of verification done by the system, the information is in turn forwarded to the regional administrator's office. The application is then processed manually based on the report given by the system, and any forfeiting identified can make the applicant liable to penalty as per the law. The system also provides the applicant the list of available dates for appointment to 'document verification' in the administrator's office, from which they can select one. The system forwards the necessary details to the police for its separate verification whose report is then presented to the administrator. The administrator will be provided with an option to display the current status of application to the applicant, which they can view in their online interface. After all the necessary criteria have been met, the original information is added to the database and the passport is sent to the applicant.

ACKNOWLEDGEMENT

I wish to express my deep sense of gratitude to my lab internal **, Mrs. B.Meena, Assistant Professor, IT Department** for her able guidance and useful suggestions, which helped me in completing this mini project work, in time.

I take immense pleasure in thanking **Prof. Ch.Suresh**, HOD, who had been a source of inspiration and for his timely guidance in the conduct of our mini project work and having permitted me to carry out this mini project work.

I would also like to thank our Faculty members of IT Deparment for all their valuable assistance and encouragement in this project work.

Finally, yet importantly, I would like to express my heartfelt thanks to my beloved parents for their blessings, my friends/classmates for their help and wishes for the successful completion of this project.

G.SivaRama Krishna	316126511021
P.Sai SriValli	316126511035
D.Ranjith	316126511014
B.Ajaey	316126511005

1.INTRODUCTION

Passport Simulation System is an interface between the Applicant and the Authority responsible for the Issue of Passport. It aims at improving the efficiency in the Issue of Passport and reduces the complexities involved in it to the maximum possible extent.

1.1 OBJECTIVE

To simplify the process of applying passport, software has been created by designing through rational rose tool, using HTML and css as a front end and php as a back end. Initially the applicant login the passport automation system and submits his details. These details are stored in the database and verification process done by the passport administrator, regional administrator and police and then passport is issued to the applicant.

1.2 EXISTING SYSTEM

- In existing system the processing of passport is done manually .
- The applicant has to fill in a printed application form and is expected to submit it.
- The application submitted usually takes a long time to reach administrators desk due to the long list of applications already pending.
- The verification documents are to be submitted manually by the applicant waiting in the queue for a long time.
- The administrator have to verify the documents and have to notify the police for personal verification which again is a time consuming process.
- The number of applications that are verified and dispatched in a single day is very less in the existing system.
- It becomes very difficult to tackle the ever increasing applications for passport and as a result of which the applicants face trouble.

1.3 PROPOSED SYSTEM

Passport Automation System, the processing of applications is done with the help of computer.

- The applicant need not go all the way to passport office to submit his printed form; instead he can fill the online application form from the comfort of his home.
- The Administrator is notified instantly about the submitted application.
- The verification documents can be scanned along with electronic signatures whose validity is thoroughly checked and taken into consideration. The applicant need not stand in the queue for a long time.
- The Administrator after verifying documents can easily proceed with other formalities to dispatch the passport to the applicant.
- The number of applications processed and dispatched in a single day is very high when compared to the existing system.
- It becomes very easy to manage the increase in applications.

2.SOFTWARE REQUIREMENTS SPECIFICATIONS

2.1PURPOSE

If the entire process of 'Issue of Passport' is done in a manual manner then it would takes several months for the passport to reach the applicant. Considering the fact that the number of applicants for passport is increasing every year, an Automated System becomes essential to meet the demand. So this system uses several programming and database techniques to elucidate the work involved in this process.this system is simulated from already existing system

2.2 SCOPE

- The System provides an online interface to the user where they can fill in their personal details and submit the necessary details.
- The authority concerned with the issue of passport can use this system to reduce his workload and process the application in a speedy manner.
- Provide a communication platform between the applicant and the administrator.
- Transfer of data between the Passport Issuing Authority and the Local Police for verification of applicant's information.
- Users/Applicants will come to know their status of application and the date in which they must subject themselves for manual document verification

2.3 GENERAL DESCRIPTION

2.3.1 PRODUCT PERSPECTIVE

The modules in the project are:

- Applicant.
- Administrator Officer.
- Regional Officer.
- Police Officer.

2.3.1 PRODUCT FUNCTION

- Insertion
- Retrieval
- Update

2.4 SPECIFIC REQUIREMENTS

2.4.1 User Interface

For the applicant to interact with the system easily, we need to provide interfaces in such a way that a applicant who has no computer knowledge can also easily access.

2.4.2 FUNCTIONAL REQUIREMENTS

- Secure registration of information by the applicants.
- Verification of application by passport administrator, regional officer, police officer.
- Adding eligible application details to the database.
- The applicants under criminal act are not allowed to issue passport.
- Booking appointment for further verification.

2.4.3 DESIGN REQUIREMENTS

Hardware Requirements:

- Processor: Pentium IV or higher
- RAM : 512 MB or higher
- Disk Space :40 GB

Software Requirements

- The front end is Php, HTML and java script
- The back end is Php Sql Manager
- Easy Php
- Web browser

We have taken these specifications because they are widely used. The software used is very common and easy to understand. The hardware requirements are very economical and they are also widely used which makes the project to work effectively.

2.5 FEASIBILITY STUDY

It is necessary and discreet to evaluate the feasibility of a project at the earliest possible time. There may be different ways of checking whether a system is feasible or not. The following feasibility studies were performed to gauge the feasibility of the system. Feasibility study can be divided into three basic forms as follows.

1. Operational feasibility

It refers to the feasibility of the product to be operational. Some products may work very well at design and implementation but may fail in the real time environment. It includes the study of additional human resource required and their technical expertise.

The management issues and user requirements have been taken into consideration. So there is no question of resistance from the users that can undetermined the possible application benefits. The well planned design would ensure the optimal utilization of the computer resources and would help in the improvement of performance status.

2.Technical Feasibility

This test includes a study of function, performance and constraints that may affect the ability to achieve an acceptable system. The technical feasibility of the project is explained as follows:

- The project has been done with a very easy software technology.
- It is user friendly and easy to understand and so it is technically feasible.

3. Economical Feasibility

It refers to the benefits or outcomes we are deriving from the product as compared to the total cost we are spending for developing the product. If the benefits are more or less the same as the older system, then it is not feasible to develop the product. The project is economically feasible as with minimum requirements and with minimum cost we are able to develop.

3.SYSTEM DESIGN

3.1 Design Activities

System design is the transformation of an analysis model into a system design model. During system design, developers define the design goals of the project and decompose the system into smaller sub-systems that can be realized by individual teams. Developers also select strategies for building the system, such as hardware/software strategy, the persistent data management strategy, the global control flow, the access control policy, and the handling of boundary conditions.

System design results in the following products:

Design goals , describing the qualities of the system that developers should optimize.

Software architecture, during the subsystem decomposition in terms of subsystem responsibilities, dependencies among subsystems.

Identifying design goals of project:

The design goals are derived from Non-functional requirements. It identifies the qualities that the system should focus on. We can select design goals based on performance, dependability, cost, maintenance, and end user criteria.

3.2 UML

The Unified Modeling Language (UML) is a visual modeling language used to specify, visualize, construct and document a software intensive system. The embedded real-time software systems encountered in applications such as telecommunications, school systems, aerospace, and defense typically tends to be large and extremely complex. It is crucial in such systems that the software is designed with a sound architecture. A good architecture not only simplifies construction of the initial system, but also, readily accommodates changes forced by a steady stream of new requirements.

The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems. The UML is a very important part of developing objects oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects. Using the UML helps project teams communicate, explore potential designs, and validate the architectural design of the software.

The primary goals in the design of the UML are: Provide users with a ready-to-use,expressive visual modeling language so they can develop and exchange meaningful models.

Provide extensibility and specialization mechanisms to extend the core concepts. Be independent of particular programming languages and development processes.

Provide a formal basis for understanding the modeling language.

4. ANALYSIS

4.1Object Oriented Analysis

Object-oriented analysis (OOA) looks at the problem domain, with the aim of producing a conceptual model of the information that exists in the area being analysed. Analysis models do not consider any implementation constraints that might exist, such as concurrency, distribution, persistence, or how the system is to be built

The sources for the analysis can be a written requirements statement, a formal vision document, and interviews with stakeholders or other interested parties. A system may be divided into multiple domains, representing different business, technological, or other areas of interest, each of which are analysed separately.

4.1.1 Use Case Diagram

Use case diagrams represent the functionality of the system from a user's point of view. Use cases are used during requirement elicitation and analysis to represent the functionality of the system. Use cases focus on the behavior of the system from an external point of view.

An actor in the Unified Modelling Language (UML) specifies a role played by a user or any other system that interacts with the subject. An Actor models a type of role played by an entity that interacts with the subject, but which is external to the subject. Actors may represent roles played by human users, external hardware, or other subjects. Actors do not necessarily represent specific physical entities but merely particular facets (i.e., roles) of some entities that are relevant to the specification of its associated use cases. A single physical instance may play the role of several different actors and a given actor may be played by multiple different instances.

Use case diagrams are usually referred to as behavior diagrams used to describe a set of actions (use cases) that some system or systems (subject) should or can perform in collaboration with one or more external users of the system (actors). Each use case should provide some observable and valuable result to the actors or other stakeholders of the system.

Use case diagrams are in fact twofold - they are both behavior diagrams, because they describe behavior of the system, and they are also structure diagrams - as a special case of class diagrams

where classifiers are restricted to be either actors or use cases related to each other with associations.

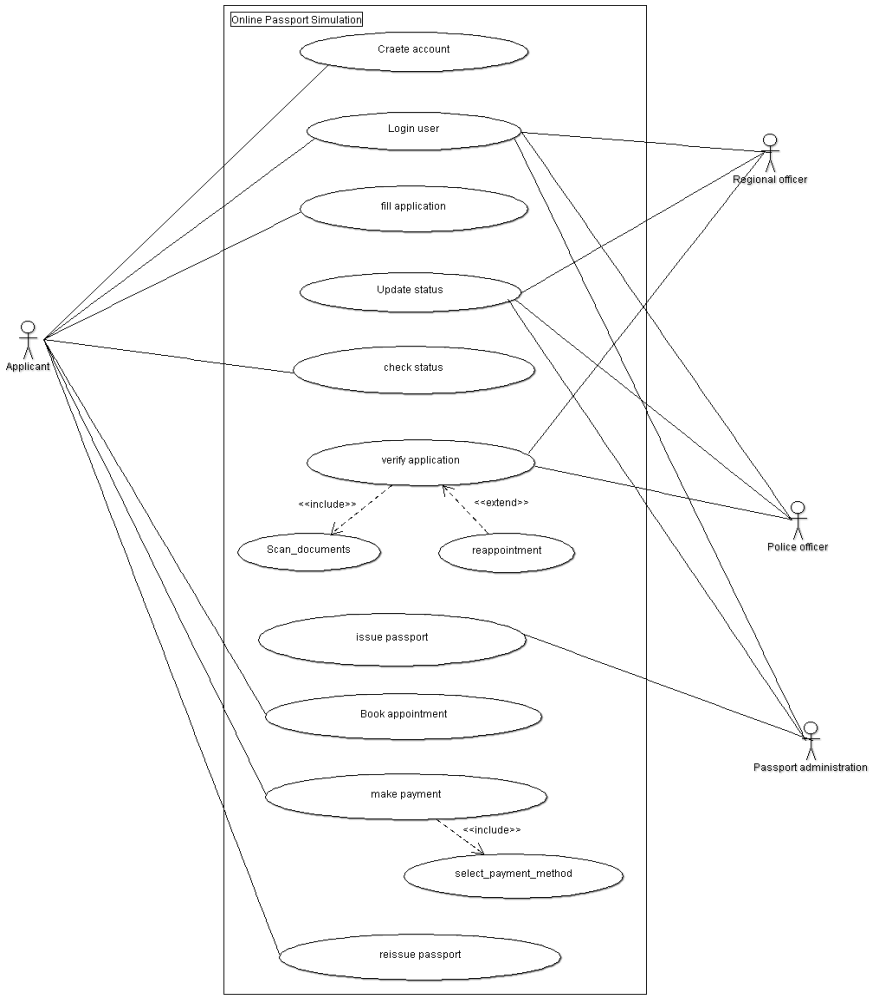


Figure 1:Use Case Diagram

4.1.2 Activity Diagram

An activity diagram describes the behavior of the system in terms of activities. The completion of these operations triggers a transition to other.

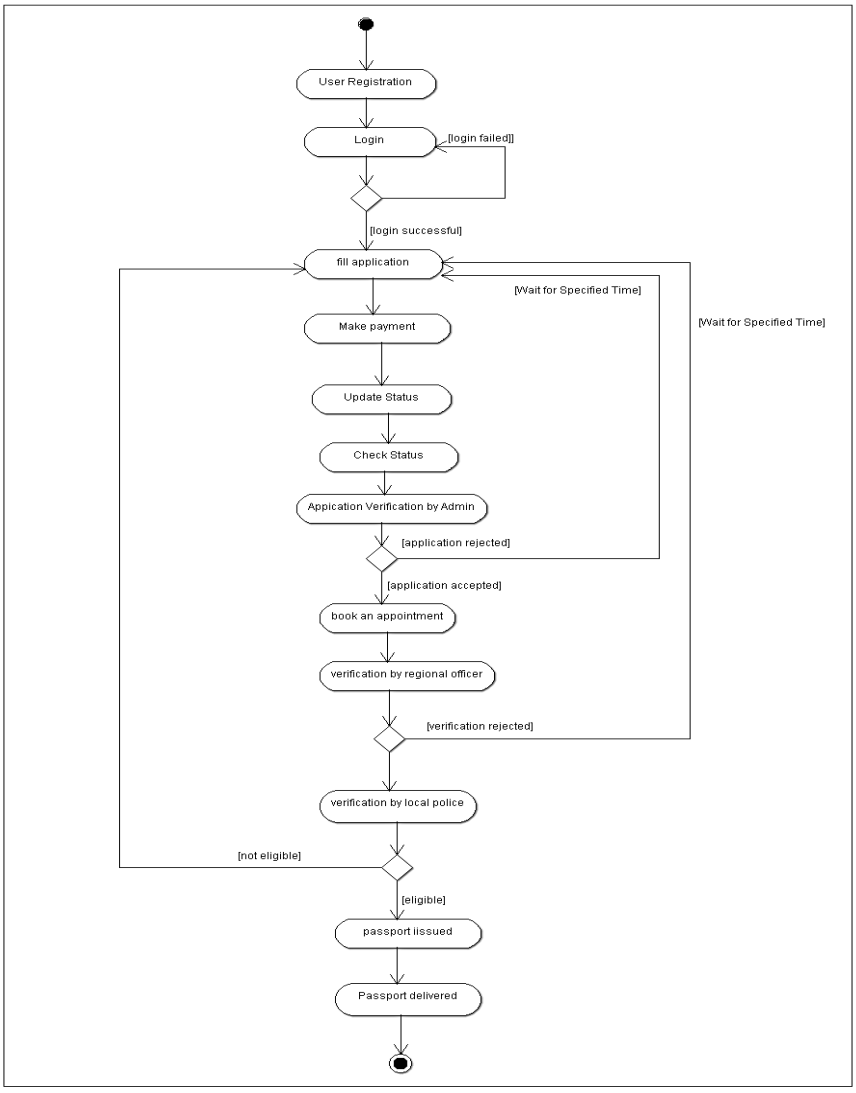


Figure 2.Activity Diagram

4.1.3 Sequence Diagram

Sequence diagrams which are a type of interaction diagrams are used to formalize the dynamic behavior of the system and to visualize the communication among objects. They are useful for identifying additional objects that participate in the use case. These are called Participate Objects.

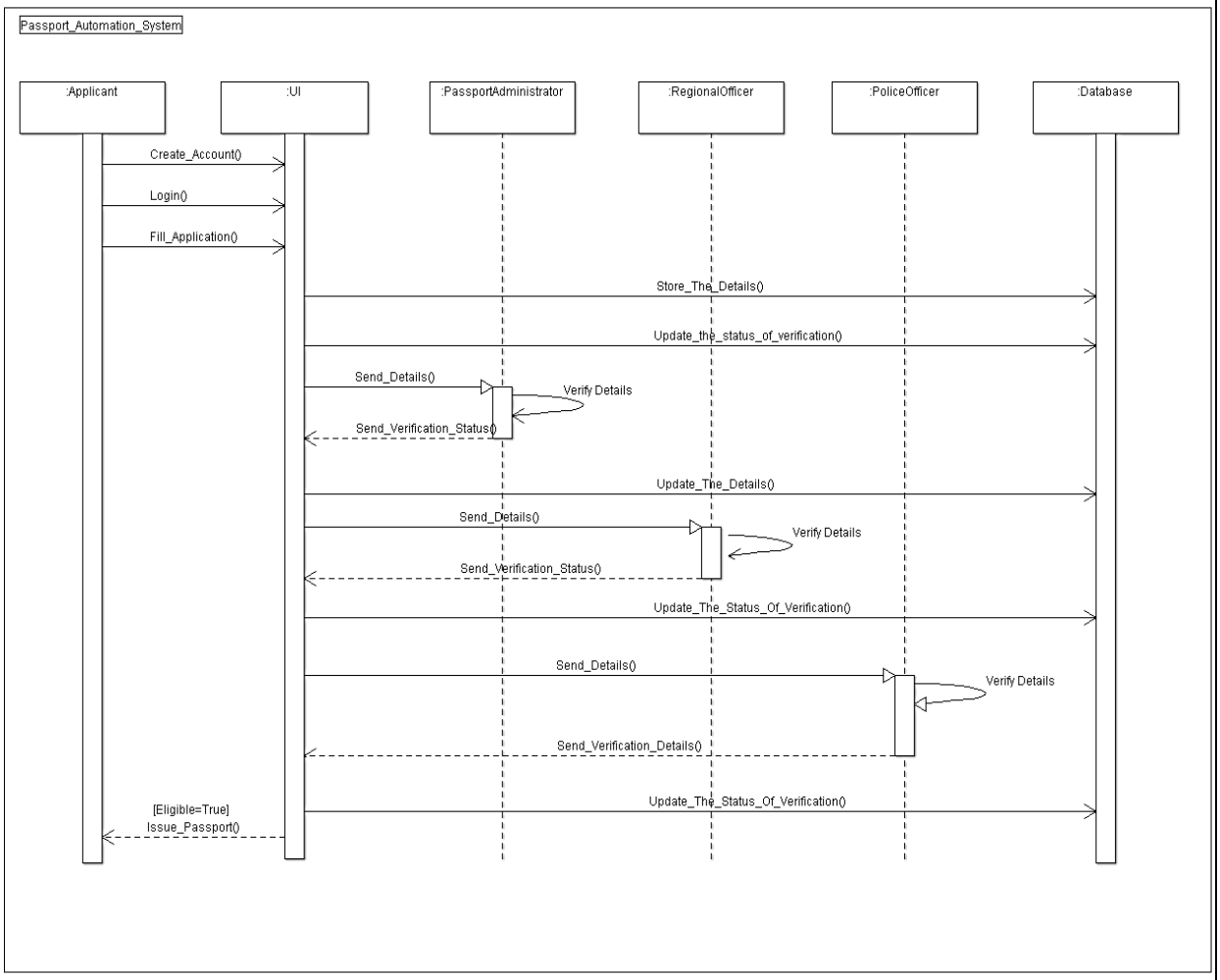


Figure 3.Sequence Diagram

4.1.4 Collaboration Diagram

Collaboration diagrams which are a type of interaction diagrams are used to formalize the dynamic behavior of the system and to visualize the communication among objects by numbering the interactions between them. Figure 4 indicates the collaboration diagram.

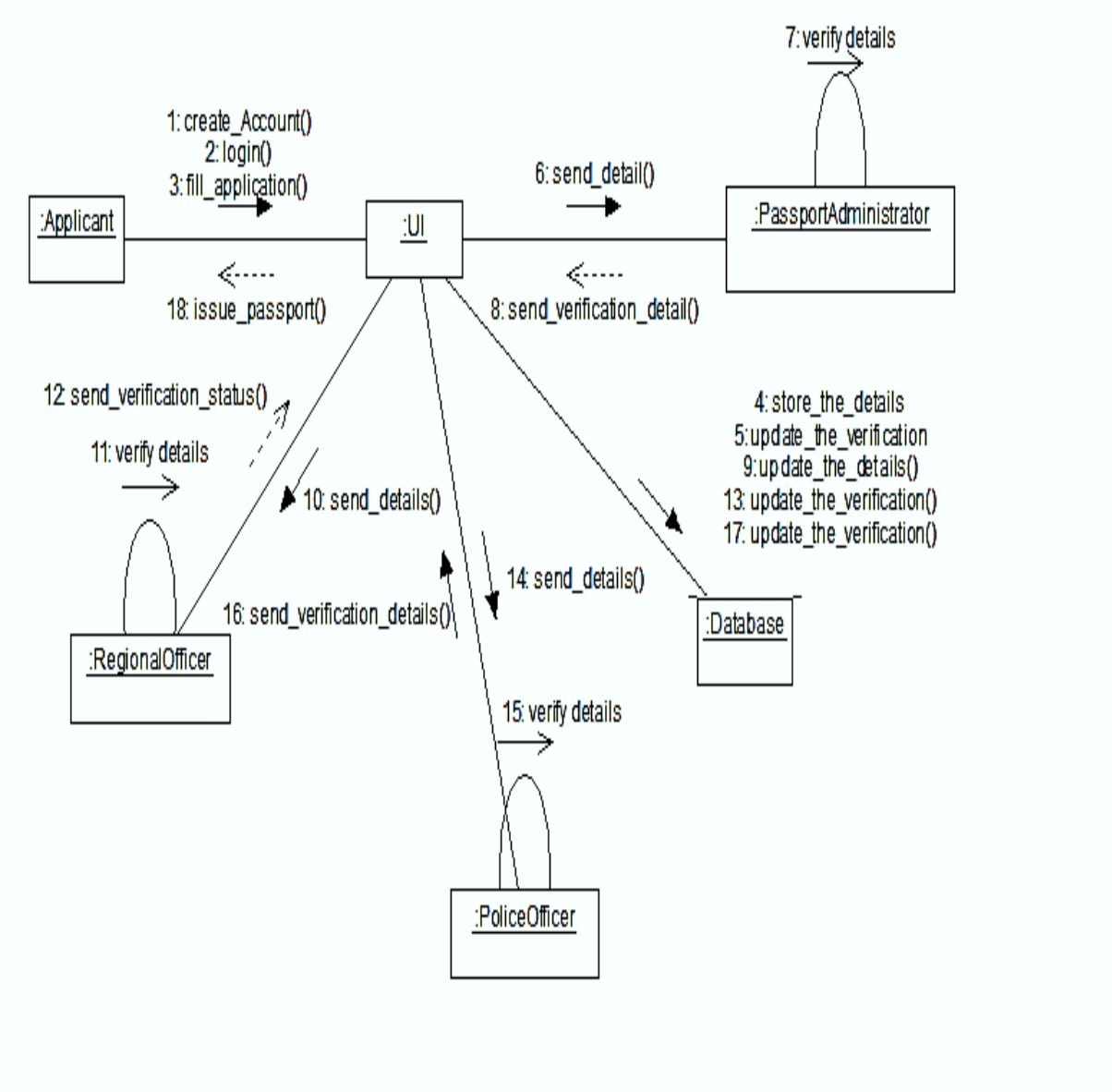


Figure 4.Collaboration Diagram

5. DESIGN

5.1 Object oriented design

Object-oriented design (OOD) transforms the conceptual model produced in object-oriented analysis to take account of the constraints imposed by the chosen architecture and any non-functional – technological or environmental – constraints, such as transaction throughput, response time, run-time platform, development environment, or programming language.

The concepts in the analysis model are mapped onto implementation classes and interfaces. The result is a model of the solution domain, a detailed description of how the system is to be built.

5.1.1 Class Diagram

Class Diagrams are used to describe the structure of the system. Classes are abstractions that specify the common structure and behavior of a set of objects. Objects are instances of classes that are created, modified, and destroyed during the execution of the system. An Object has state that includes the values of its attributes and its links with other objects.

The class diagram is the main building block of object-oriented modeling. It is used for general conceptual modelling of the structure of the application, and for detailed modelling translating the models into programming code. Class diagrams can also be used for data modelling. The classes in a class diagram represent both the main elements, interactions in the application, and the classes to be programmed.

In the diagram, classes are represented with boxes that contain three compartments:

- The top compartment contains the name of the class. It is printed in bold and centered, and the first letter is capitalized.
- The middle compartment contains the attributes of the class. They are left-aligned and the first letter is lowercase.
- The bottom compartment contains the operations the class can execute. They are also left-aligned and the first letter is lowercase.

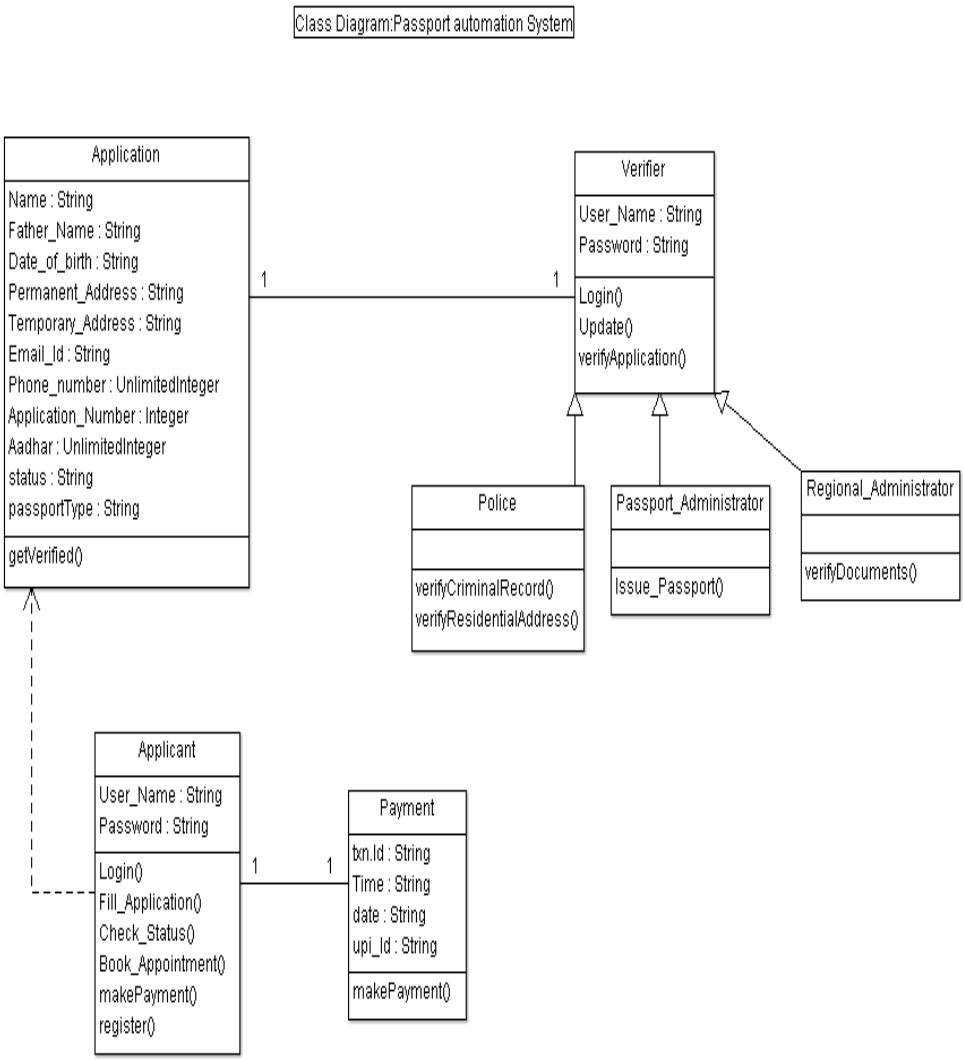


Figure 5.Class Diagram

5.1.2 State Diagram

A UML state chart is a notation for describing the sequence of states an object goes through in response to external events. A state is a condition satisfied by the attributes of an object. A transition represents a change or state triggered by events, condition or time. Actions are small atomic behaviours that are executed at specific points in the state machine. Actions can occur in three places:

- When a transition is taken.
- When a state is entered.
- When a state is exited

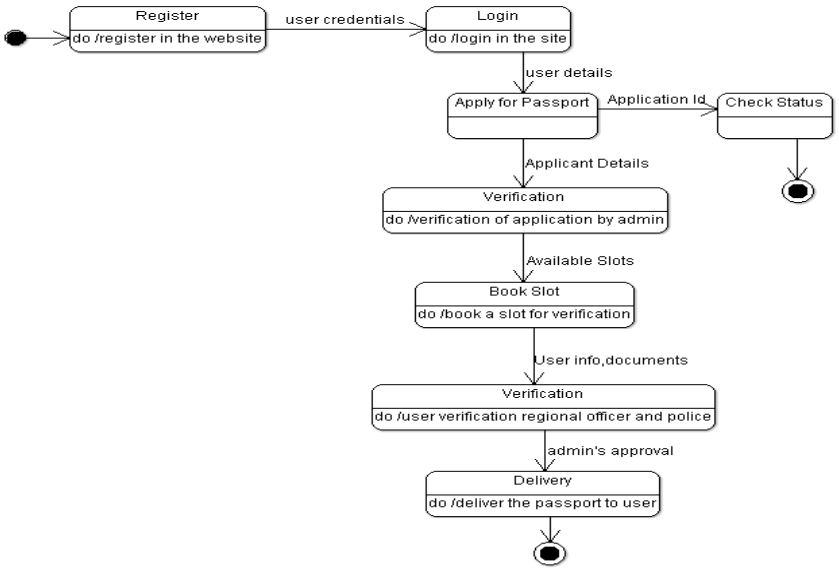


Figure 6. State Diagram

5.1.3 Component Diagram:

A component diagram provides a physical view of the system. Its purpose is to show the dependencies that the software has on the other software components (e.g., software libraries) in the system. The diagram can be shown at a very high level, with just the large-grain components, or it can be shown at the component package level.

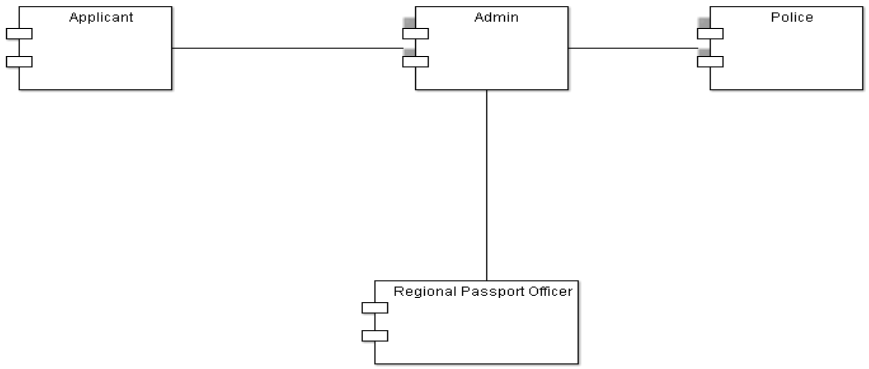


Figure 7.Component Diagram

5.1.4 Deployment Diagram

The deployment diagram shows how a system will be physically deployed in the hardware environment. Its purpose is to show where the different components of the system will physically run and how they will communicate with each other. Since the diagram models the physical runtime, a system's production staff will make considerable use of this diagram.

The notation in a deployment diagram includes the notation elements used in a component diagram, with a couple of additions, including the concept of a node. A node represents either a physical machine or a virtual machine node (e.g., a mainframe node). To model a node, simply draw a three-dimensional cube with the name of the node at the top of the cube. Use the naming convention used in sequence diagrams.

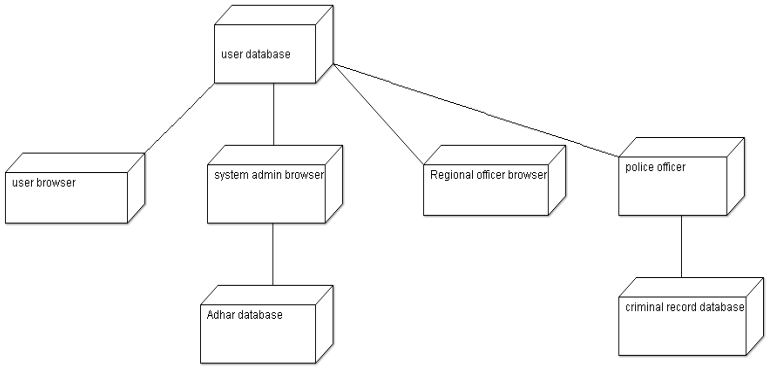


Figure 8.Deployment Diagram

5.2 Database Design

REGISTRATION TABLE:

S.NO	FIELD NAME	DATA TYPE
1	FIRSTNAME	VARCHAR(20)
2	LASTNAME	VARCHAR(20)
3	GENDER	VARCHAR(10)
4	AADHARNO	VARCHAR(20)
5	BIRTHPLACE	VARCHAR(20)
6	STATE	VARCHAR(20)
7	COUNTRY	VARCHAR(20)
8	EDUCATION	VARCHAR(20)
9	EMPLOYMENT	VARCHAR(20)
10	EMAIL	VARCHAR(25)
11	PHONENO	VARCHAR(15)
12	REGISTERID	VARCHAR(20)

SLOT BOOKING:

S.NO	FIELD NAME	DATA TYPE
1	REGISTERID	VARCHAR(20)
2	NAME	VARCHAR(20)
3	SELDATE	VARCHAR(10)
4	SELSLOT	VARCHAR(20)
5	STATUS	INT(3)

6.Coding

Home Page:

```
<!DOCTYPE html>
<html>
<head>
<title>Bootstrap Example</title>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css">
<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.0/css/bootstrap.min.css">
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.3.1/jquery.min.js"></script>
  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.0/js/bootstrap.min.js"></script>
</head>
<body>
<div class="navbar">
  <a href="file:///home/anits/pro1.html">Home</a>
  <a href="#news">login</a>
<a href="srkaboutuspage.html">about</a>
<a href="srkcontactuspage.html">contactus</a>
</div>
<tr>
<table >
    <td width="8"><span  style="width:8px; float:left; display:inline; height:23px; "></span></td>
         <td class="Light_Blue_BG">
          <div class="Content_Bar" style="width:950px;">

<marquee behavior="scroll" direction="left" id="marquee"
```

```
onkeydown="this.setAttribute('scrollamount', 0, 0);"
onmouseover="this.setAttribute('scrollamount', 0, 0);"
onmouseout="this.setAttribute('scrollamount', 6, 0);"><strong>WELCOME TO PASSPORT
SIMULATION WEBSITE</strong>

          </marquee>

          </div>

        </td>

         <td width="22" align="center" valign="middle" class="Light_Blue_BG">

          <input type="button" value="Pause" id="marqueeControlButton"
onClick="controlMarquee();" style="display:none">

          <img id ="image1" alt="stop" src="stop.jpg" width="19" height="19"
onClick="controlMarquee();"/>

          <img id ="image2" alt="start" width="19" height="19" src="start.png"
onClick="controlMarquee();" style="display:none;"/>

         </td>

        </tr>

       </table>

</div>

<div class="row">

  <div class="left" style="background-color:navy;">

    <h2>Menu</h2>

    <input type="text" id="mySearch" onkeyup="myFunction()" placeholder="Search.."
title="Type in a category">

    <ul id="myMenu">

      <li><a href="#">passport admin</a></li>

      <li><a href="#">Regional admin</a></li>

       <li><a href="http://localhost/finalminiprojectsrk/policelogin.php">Police</a></li>

      <li><a href="#">Apply for Passport</a></li>

      <li><a href="#">Appointment Status</a></li>

      <li><a href="#">Passport Act 1967</a></li>

      <li><a href="#">Passport Rules 1980</a></li>

    </ul>

  </div>
```

```
<div class="right">
<br>
<br>
<div class="container" align="left">
  <div id="myCarousel" class="carousel slide" data-ride="carousel">
    <!-- Indicators -->
    <ol class="carousel-indicators">
      <li data-target="#myCarousel" data-slide-to="0" class="active"></li>
      <li data-target="#myCarousel" data-slide-to="1"></li>
      <li data-target="#myCarousel" data-slide-to="2"></li>
    </ol>
    <!-- Wrapper for slides -->
    <div class="carousel-inner">
 <div class="item active" align="left">
        <img src="passportsrk.png" alt="Los Angeles" style="width:100% ;
">           &nbs
      </div>
      <div class="item" align="left">
        <img src="applysrk.png" alt="Chicago" style="width:100%
;">            
      </div>
      <div class="item" align="left">
        <img src="visionsrk.png" alt="New york"
style="width:100%;"><br>         
   
      </div>
    </div>
    <br>
    <!-- Left and right controls -->
    <a class="left carousel-control" href="#myCarousel" data-slide="prev">
      <span class="glyphicon glyphicon-chevron-left"></span>
      <span class="sr-only">Previous</span>
```

```
  </a>
   <a class="right carousel-control" href="#myCarousel" data-slide="next">
     <span class="glyphicon glyphicon-chevron-right"></span>
     <span class="sr-only">Next</span>
     <br><br><br>
   </a>
 </div>
</div>
<center>
   <br><br><br><a href="SlotBooking.html" style="color:#ffffff">
<img onmouseover="bigImg(this)" onmouseout="normalImg(this)" border="0"
src="appoitmentavailibility.png" alt="Smiley" width="160" height="80"
>    
<a href="http://localhost/finalminiprojectsrk/pass2.php" style="color:#ffffff">
<img onmouseover="bigImg(this)" onmouseout="normalImg(this)" border="0"
src="existinguser.png" alt="Smiley" width="160" height="80" ><br><br>
</a>
<a href="http://localhost/finalminiprojectsrk/statusapp.php" style="color:#ffffff">
<img onmouseover="bigImg(this)" onmouseout="normalImg(this)" border="0"
src="appstatus.png" alt="Smiley" width="160" height="80"
>    </a>
<a href="http://localhost/finalminiprojectsrk/srkpassportform.php" style="color:#ffffff">
<img onmouseover="bigImg(this)" onmouseout="normalImg(this)" border="0"
src="registernow.png" alt="Smiley" width="160" height="80" ><br><br></a></center>
<script>
function bigImg(x) {
  x.style.height = "90px";
  x.style.width = "180px";
}
function normalImg(x) {
  x.style.height = "80px";
  x.style.width = "160px";
```

```
}
</script>
</center>
</div>
<div class="right">
</div>
<script>
function myFunction() {
  var input, filter, ul, li, a, i;
  input = document.getElementById("mySearch");
  filter = input.value.toUpperCase();
  ul = document.getElementById("myMenu");
  li = ul.getElementsByTagName("li");
  for (i = 0; i < li.length; i++) {
    a = li[i].getElementsByTagName("a")[0];
    if (a.innerHTML.toUpperCase().indexOf(filter) > -1) {
      li[i].style.display = "";
    } else {
      li[i].style.display = "none";
    }
  }
}
function controlMarquee() {
 var marquee=document.getElementById('marquee');
 if(document.getElementById('marqueeControlButton').value=="Pause"){
 document.getElementById('marqueeControlButton').value="Start";
 document.getElementById("image1").style.display = "none";
 document.getElementById("image2").style.display = "";

 marquee.stop();
```

```
 }
 else if(document.getElementById('marqueeControlButton').value=="Start"){
 document.getElementById('marqueeControlButton').value="Pause";
 document.getElementById("image2").style.display = "none";
 document.getElementById("image1").style.display = "";
 marquee.start();
 }
 }
</script>
</body>
</html>
```

Passport Status Checking:

```
<?php
 session_start();
 if(isset($_SESSION['stat']))
 {
  echo "<center>";
  echo "<h1>APPLICATION STATUS</h1>";
  $con=new mysqli("localhost","root","","sivaram");
  $stat1=$_SESSION['stat'];
  $rows=$con->query("select * from appstatus where registerid='$stat1'");
  while(list($registerid,$regionalofficer,$police,$admin,$status,$regoinalstatus,$policestatus,$
adminstatus)=$rows->fetch_row()){
    echo "<tr><td>registerid:$registerid</td></tr>";
  if($status==1)
   echo "<tr><td>your application is accepted...<br><h3>you will recieve your passport in a
while</h3></td></tr><br>";
  else if($regionalofficer==1&&$$regoinalstatus=0){
   if($admin==1)
    echo "<tr><td>your application is rejected</td></tr><br>";
```

```
  else
   echo "<tr><td>your application isunder review by regional officer</td></tr><br>";
  }
 else if($police==1&&$$policestatus=0){
  if($admin==1)
   echo "<tr><td>your application is rejected</td></tr><br>";
  else
   echo "<tr><td>your application isunder review by regional police officer</td></tr><br>";
  }
 else if($admin==1&&$adminstatus==0)
  {
   echo "<tr><td>your application is rejected</td></tr><br>";
  }
 }
}
?>
```

Appointment Status Checking:

```
<?php
 session_start();
 if(isset($_SESSION['stat']))
 {
  echo "<center>";
  echo "<h1>APPOINTMENT STATUS</h1>";
  $con=new mysqli("localhost","root","","sivaram");
  $stat1=$_SESSION['stat'];
  $rows=$con->query("select * from ppslotbooking where registerid='$stat1'");
  while(list($appointmentstatus,$applid)=$rows->fetch_row()){
   echo "<tr><td>Application Id:$applid</td></tr>";
```

```
  if($appointmentstatus==1)
   echo "<tr><td>Your appointment is confirmed...<br><h3>Kindly bring all your documents for Verification</h3></td></tr><br>";
  else if($appointmentstatus==-1)
   echo "<tr><td>Your appointment is rejected...<br><h3>Please wait for some mote time,</tr><br>";
 }
}
?>
```

Slot Booking:

```
<?php
if(isset($_POST['submit'])){
//Create connection
$conn1 =mysqli_connect("localhost","root","","sivaram");
$sql1 = "SELECT registerid FROM `passportform`";
if($result = mysqli_query($conn1, $sql1))
$row = mysqli_fetch_array($result);
if($row['registerid']===$_POST['aid'])
{
$conn =mysqli_connect("localhost","root","","sivaram");
$sql="INSERT INTO ppslotbooking(registerid,Name,seldate,selslot,status) VALUES('$_POST[aid]','$_POST[name]','$_POST[slotbook]','$_POST[slot]',0)";
mysqli_query($conn,$sql);
}
else
{
$message = "This Application Id Is Not Valid.Please Check";
                                                    echo "<script type='text/javascript'>alert('$message');</script>";
}
}
```

```
?>
```

Login Page:

```
<?php
 if(isset($_POST['Submit'])){
  $email=$_POST['email_id'];
  $password=$_POST['passwordat'];
  $con=new mysqli("localhost","root","","sivaram");
  $sql="SELECT * from registrationform where email='$email' and password='$password'";
  $result=mysqli_query($con,$sql);
  if($result->num_rows>0)
  {
   header('Location:pass2.html');
   exit();
  }
else
  {
   $message = "enter correct mail and password";
   echo "<script type='text/javascript'>alert('$message');</script>";
   return false;
  }
  $con->close();
 }
?>
```

Application Form

```
<?php
 if(isset($_POST['Submit'])){
  $firstname= $_POST['fname'];
```

```
$lastname= $_POST['lname'];
$gender=$_POST['gender'];
$aadhar=$_POST['aadhar'];
$birthvillage=$_POST['birthvillage'];
foreach($_POST['country'] as $select);
$state=$_POST['state'];
$district=$_POST['district'];
$education=$_POST['education'];
$employment=$_POST['employment'];
$email=$_POST['email'];
$phone=$_POST['mobileno'];
if (strpos($email, '.com') == false) {
 $message = "enter correct mail";
 echo "<script type='text/javascript'>alert('$message');</script>";
 return false;
}
else if(strlen($phone)!=10)
{
 $message = "wrong number";
 echo "<script type='text/javascript'>alert('$message');</script>";
 return false;
}
$con=new mysqli("localhost","root","","sivaram");
$sql="SELECT * from passportform where email='$email'";
$result=mysqli_query($con,$sql);
if($result->num_rows>0)
{
 $message = "you already have account";
 echo "<script type='text/javascript'>alert('$message');</script>";
 return false;
```

```
  }
  else
  {
   $appid=$_POST['slotbook'].$phone;
   $sql1="insert into passportform
values('$firstname','$lastname','$gender','$aadhar','$birthvillage','$select','$state','$district','$ed
ucation','$employment','$email','$phone','$appid')";
  if ($con->query($sql1) === TRUE) {
    $sql2="insert into appstatus values('$appid',0,0,0,0)";
    if ($con->query($sql2) === TRUE)
    {
     header('Location:login.php');
    }
   }
  }
  $con->close();
 }
?>


<div class="right">
<h2>Application Form</h2>
<form name="myform" method="post";>
<center>
<label for="fname">First Name: </label>
<input type="text" id="fname" name="fname" required/><br>
<label for="surname">     Surname: </label>
<input type="text" id="lname" name="lname" required/>
<br>
<label for="gender">Gender: </label>
<input type="radio" name="gender" value="Male" checked>Male  
```

<input type="radio" name="gender"
value="Female">Female   &
nbsp;

country: <select name="country[]" id="mySelect">

<option value="China">China</option>

<option value="Denmark">Denmark</option>

<option value="Finland">Finland</option>

<option value="India">India</option>

<option value="france">France</option>

</select>

<label for="aadhar">Aadhar no: </label>

<input type="text" id="aadhar" name="aadhar" required/>

<label for="aadhar">BirthVillage:</label>

<input type="text" id="birthvillage" name="birthvillage" required/>

<label
for="State"> State:
     </label>

<input type="text" id="state" name="state" required/>

<label
for="District">
District: </label>

<input type="text" id="district" name="district" required/>

<label for="Education"> Education: </label>

<input type="text" id="education" name="education" required/>

<label for="Employment">   Employment:</label>

<input type="text" id="employment" name="employment" required/>

<label for="email">Email-ID: </label>

<input type="text" id="email" name="email" required/>

<label for="Mobile no">   Mobile No:</label>

<input type="text" id="mobileno" name="mobileno" required/>

Select Date: <input type="date" name="slotbook" data-date-inline-picker="true">

```
<br><br>
<div class="container">
<input type="submit" name="Submit" class="btn" value="submit"></p></div><br><br>
</center>
</form>
</body>
</div>
```

7. TESTING & TEST CASES

Testing is the process of analyzing a system or a system component to detect the differences between specified and observed behavior of the system. In the passport simulation project to check whether the project works as per the requirements are not testing is required.

Testing Activities

Unit Testing:

Unit testing focuses on the building blocks of the software system, that is, objects and subsystems. There are three motivations behind focusing on components.

In this testing ,we have fifteen layouts and we check whether every layout works as per the requirement

Integration Testing:

Integration testing detects faults that have not been detected during unit testing by focusing on small groups of components. Two or more components are integrated and tested, and when no faults are revealed additional components are added to the group.

In this testing ,we combine all the four layouts as a whole component and performs the testing on the entire websitewhether it was working properly and producing the expected results or not.This can be done by linking all the pages and switvhing from one page to another.

System Testing:

Unit and integration testing focus on finding faults in individual components and the interface between components. Once the components have been integrated, system testing ensures that the complete system complies with the functional and nonfunctional requirements.

In this we will apply for passport and every officer logins and accepts the request.And we check whether the process is going on as per the requirements .

Acceptance testing:

To determine whether a system satisfies its acceptance criteria and business requirements or not. Similar to System testing in that the whole system is checked, but the important difference is the change in focus. It is done by real business users. It enables the customer to determine whether to accept the system or not. It is also called as Beta Testing, application Testing or End User Testing.

7.1Test Cases

To determine whether a system satisfies its acceptance criteria and bussiness requirements or not .similar to System testing ,in that the whole system is checked,but the important difference is the change in focus.In this it is done by bussiness users.

In this the website is handedover to some user and they will check the website and decide whether it is acceptable or not.

Module Name	Applicant Name.
Test Case	To test the validity of the applicant.
Input	An invalid applicant name.
Output	An alert saying that applicant name is not valid.

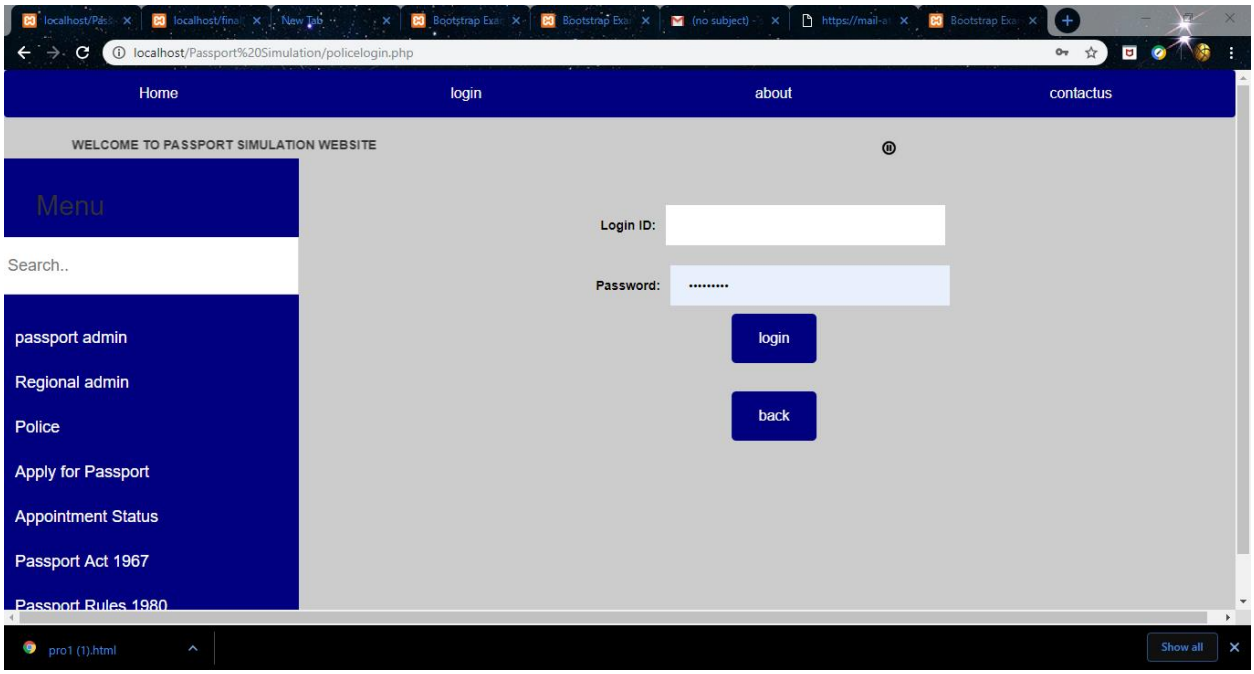
Module Name	New applicants
Test Case	To check if there are any new applicants.
Input	Check all the details of the applicants.
Output	No new applicants.

Module name	Applicant details.
Test case	To test whether the admin can fill the details of the applicant.
Input	Submit without input.
Output	Alerts the admin to enter data which is left empty.

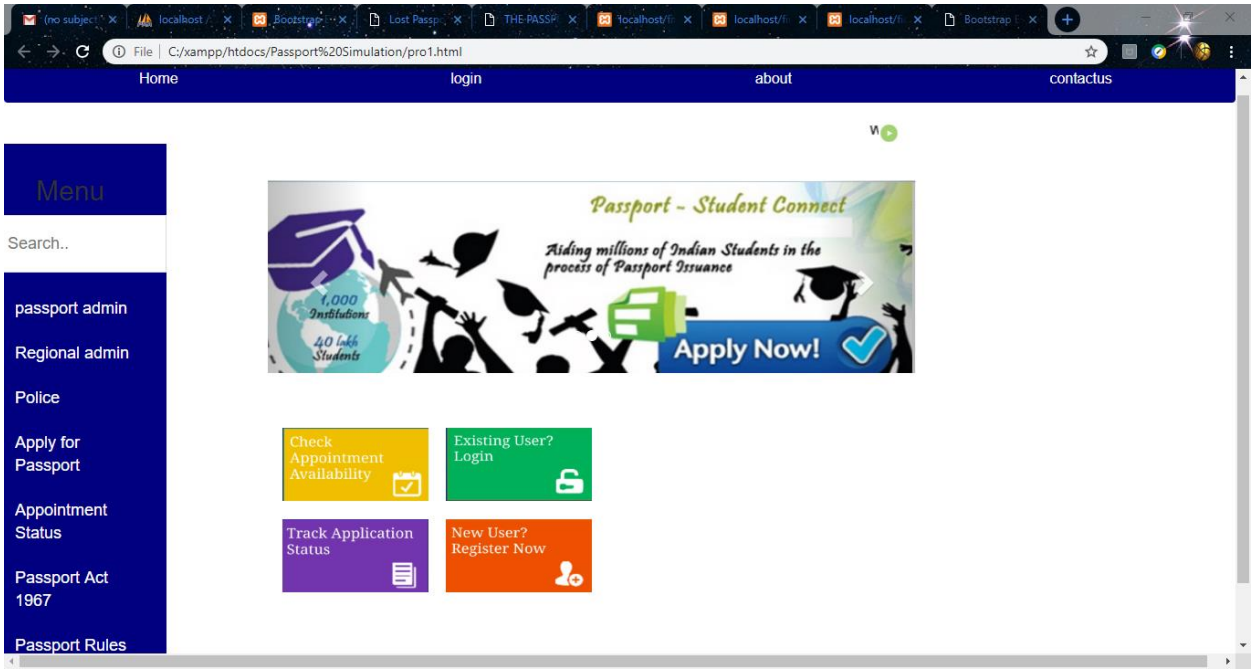
Module Name	Login
Test Case	To check whether the applicant enters his own home page.
Input	Enter the applicant name and password.
Output	Applicant logs into his own home page.

8.RESULTS

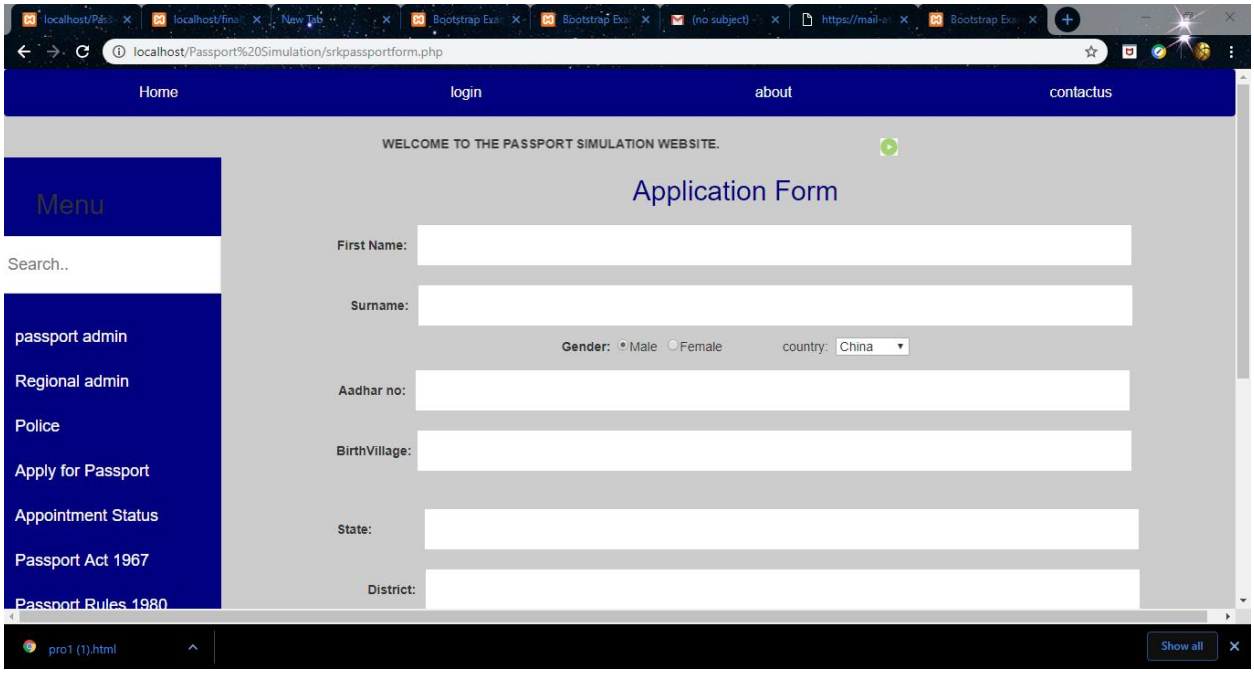
8.1 Login Page:



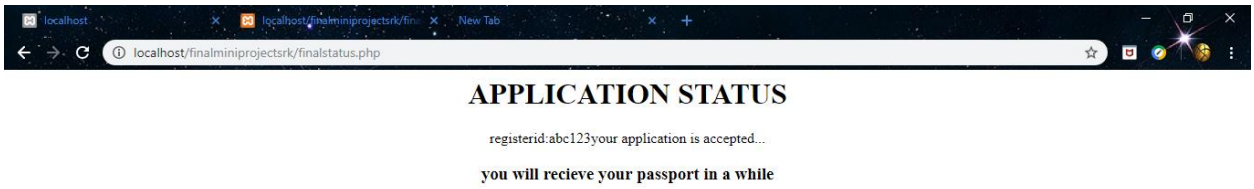
8.2 Home Page:



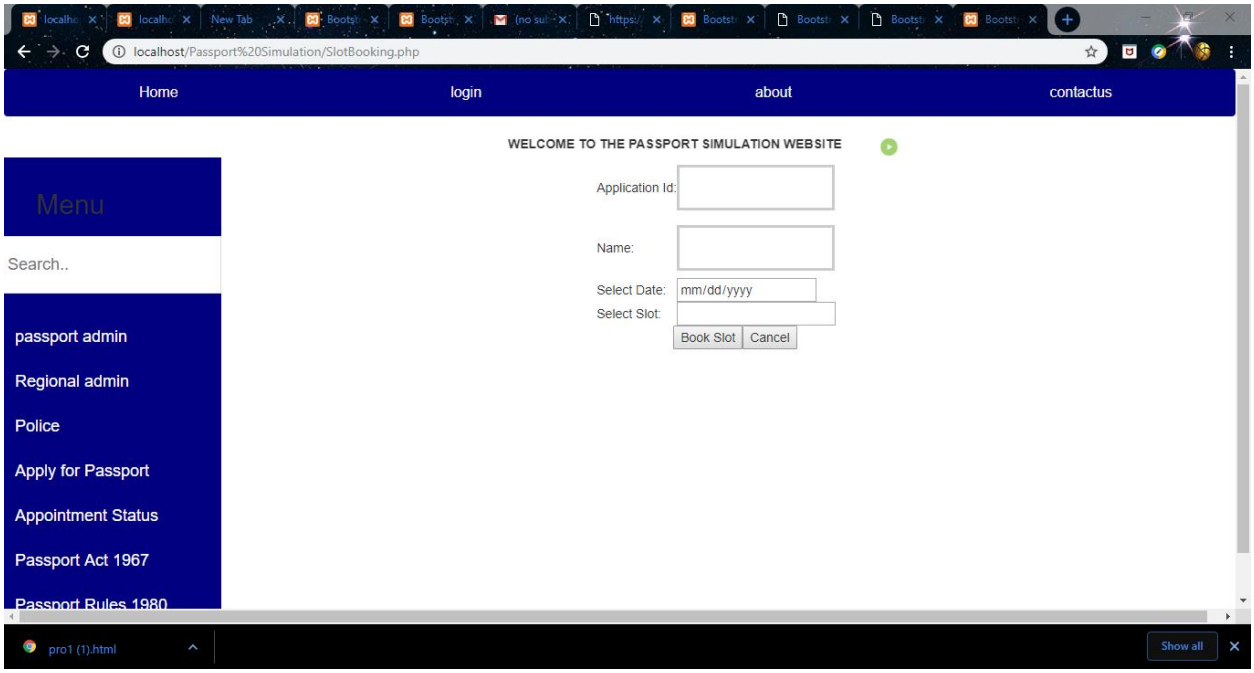
8.3Application form page:



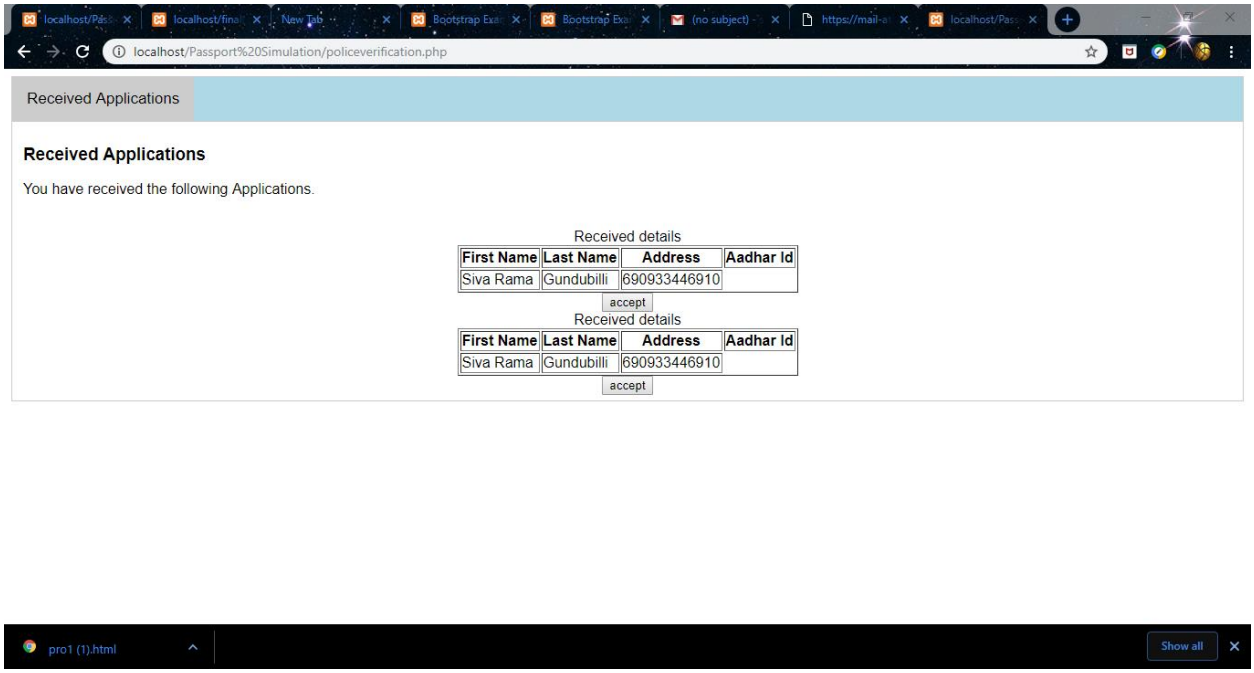
8.4Application Status:



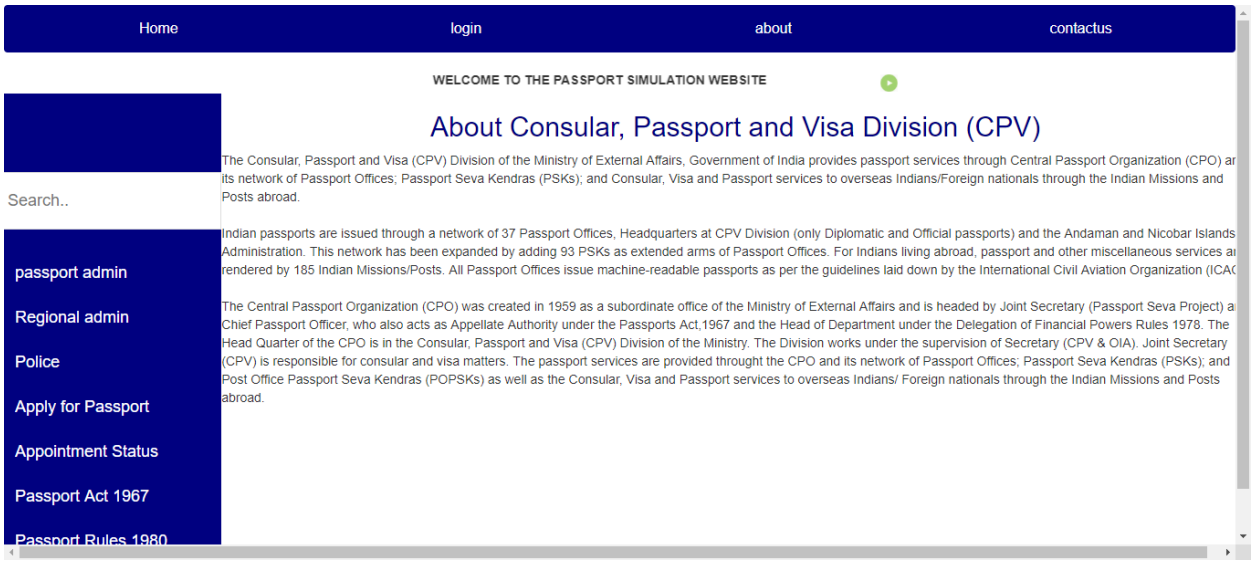
8.5 Slot Booking:



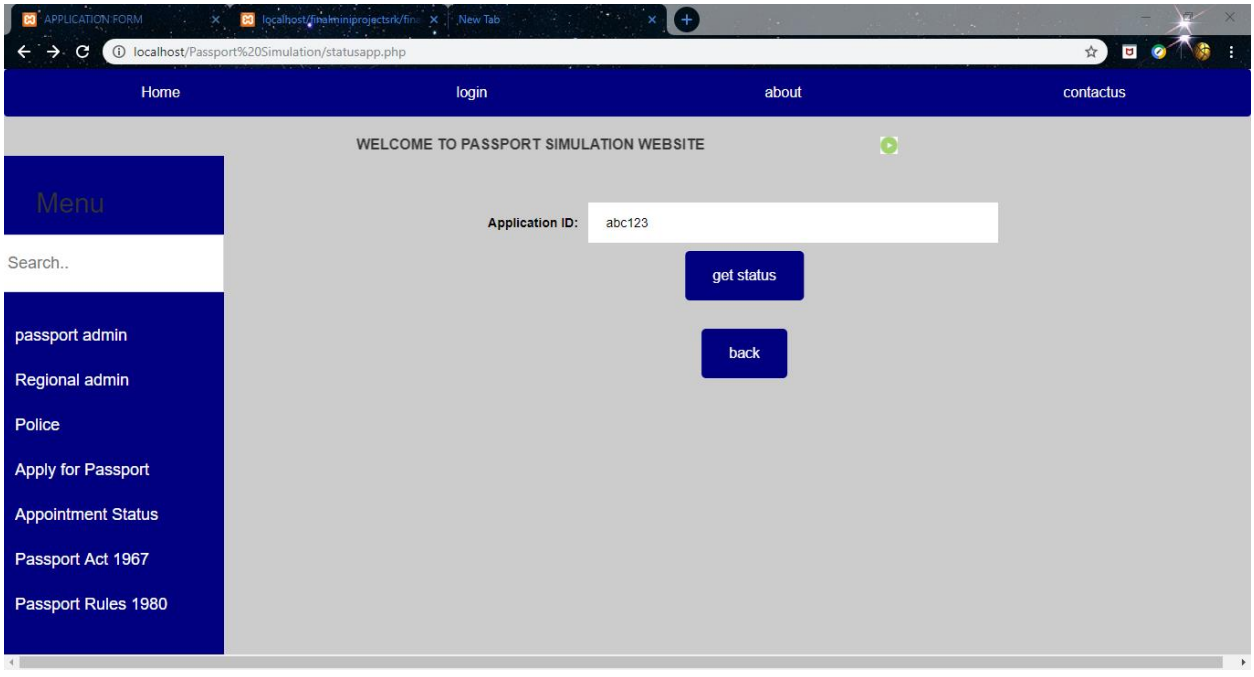
8.6 Dashboard:



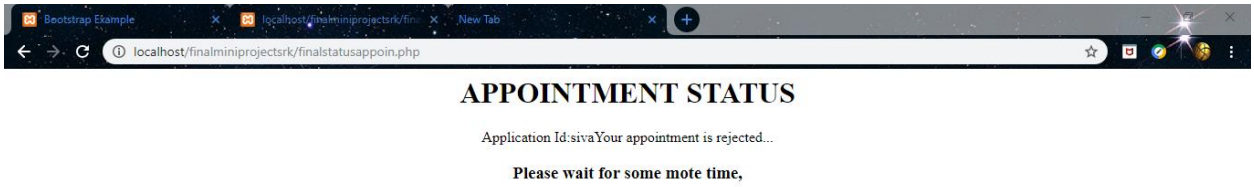
8.7 Contact Page:



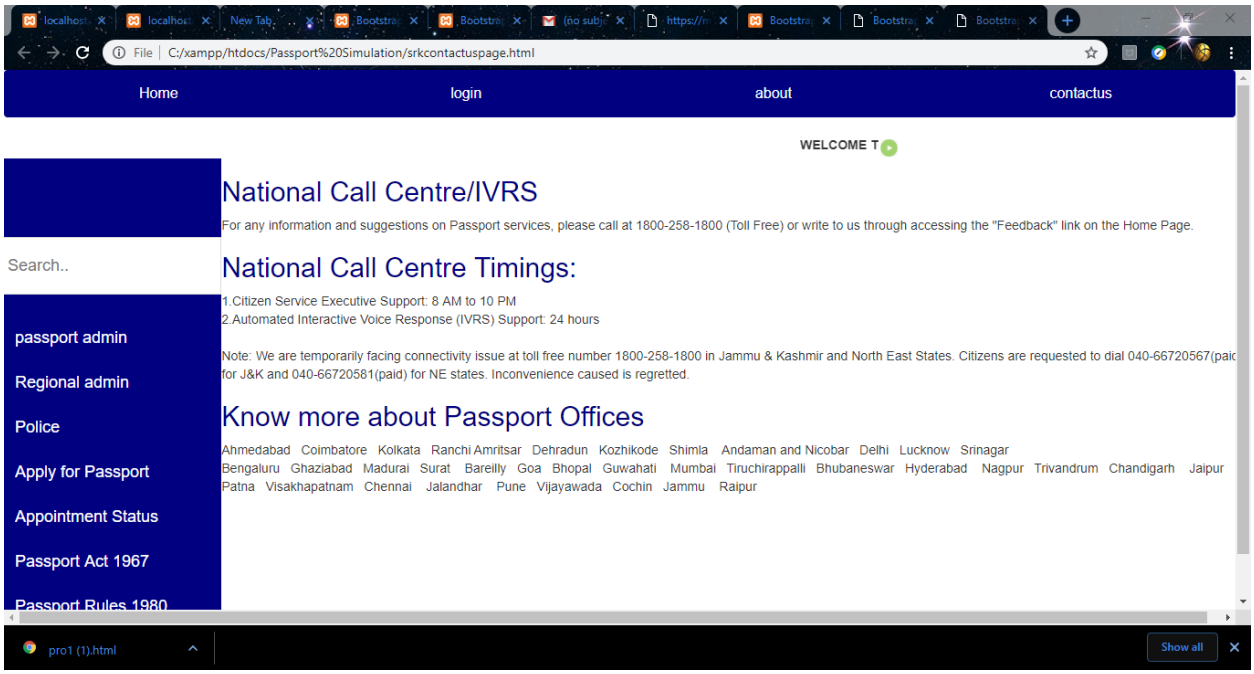
8.8 Status Checking Page:



8.9 Showing Appointment:



8.10 Contact Application Page:



9. CONCLUSION

The purpose of our project is to provide an elegant system satisfying the requisites and attaining goals. We have designed the project so that it can be easily understood by any person irrespective of his knowledge in computer science. The passport Simulation System is expected to function as per the requirements and we expect that it will satisfy the needs of the user.

This system is designed to provide the user with many options and to be near perfect. This project always has scope for further development, to provide the users with more sophisticated functions.

The project provides a system which provides interface between the user and the administrator.

Working on this project left us with an experience of team work, where each of us has contributed immensely.

10. FUTURE IMPROVEMENT

This project can further be extended to:

- Issuing passport to the applicant.
- Options for speedy delivery of passport.
- Different payment systems like credit card, debit card, amazon pay, Phonepe, Paytm.
- Visa services can also be added

11. BIBLIOGRAPHY

1. The Unified Modelling Language

Authors: Grady Booch

Publications: Pearson education

2. Java Server Pages

Author: Hans Bergsten

Publications: Shroff Publishers

3. Object Oriented Software Engineering

Author: Allen H.Dutoit

Publications: Pearson Education