PLACEMENT SOFTWARE

A MINI PROJECT REPORT

Submitted by

MUTHAIAH YADAV.A (812018104303)
THARANINATHAN.M (812018104305)
KIRUBAKARAN.T (812018104701)

In partial fulfillment for the award of the degree of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING

M.A.M COLLEGE OF ENGINEERING AND TECHNOLOGY,

ANNA UNIVERSITY:: CHENNAI 600 025

JANUARY 2022

ANNA UNIVERSITY::CHENNAI 600 025

BONAFIDE CERTIFICATE

Certified that this Project report, **"PLACEMENT SOFTWARE**" is the bonafide work of **MUTHAIAHYADAV.A (812018104303), THARANINATHAN.M (812018104305)** and **KIRUBAKARAN.T (812018104701)** who carried out the mini project work under my supervision.

SIGNATURE

Dr. K. Geetha M.E., Ph.D
Head of the Department,
Department of CSE,
M.A.M College of Engineering
and Technology,
Tiruchirappalli – 621 105.

SIGNATURE

Mr.V. Pugazhenthi, M.E.,
Supervisor,
Department of CSE,
M.A.M College of Engineering
and Technology,
Tiruchirappalli – 621 105.

Submitted for B.E degree viva-voce Examination held at **M.A.M College of Engineering and Technology** on ……

INTERNAL EXAMINER **EXTERNAL EXAMINER**

ACKNOWLEDGEMENT

With warm hearts and immense pleasure, I thank the almighty for his grace and blessing bestowed on me, which drove me to the successful completion of this project work. I take this opportunity to express my sincere gratitude to the respected Director **Dr. M. A. Maluk Mohammed, M.E., Ph.D.,** and the Secretary & Correspondent **Mrs.Fathima Bathool Maluk** who is guiding force for all the activities held in our college.

I express my sincere and humble tone of thanks to our Principal **Dr. X. Susan Christina, M.E., Ph.d.,** for providing me with all facilities needed for the successful completion of my work.

I would like to thank our Head of the Department **Dr.K.Geetha,M.E.,Ph.D.,** for her cooperation, advice and suggestions at every stage of my mini project work.

I would also like to express my sincere thanks to my guide **Mr.V.Pugazhenthi,M.E., Assistant Professor**, Department of CSE for motivating me throughout the mini project work.

I am very proud to external my sincere thanks and gratitude to our project coordinator **Mrs.S.Jagadeeswari, M.E.,** Associate Professor, Department of Computer Science and Engineering, M.A.M College of Engineering and Technology, for her excellent guidance, advice and encouragement which boosted up our energy throughout the project development.

I also thank all the teaching faculty and non-teaching faculty of the Department of Computer Science and Engineering, my parents, and all my friends for their help and support to complete this project successfully.

ABSTRACT

Placement Software is a web based application developed for automating the activities of the Training and placement cell of the Institute. The main objective of this project work is providing the classified ads by compiling and listing available opportunities at various level. This also provides the user-friendly platform to search for and apply to various opportunities and facilitate a student to find a job that matches their qualification and interest.The student academic performance like marks,CGPA,etc personal information are kept maintained in the database resided at the backend server.It also provide the history of the placed student in the previous year.The system this web application can be accessed from any where with proper access previleges.This also facilitate the industry personnel to track the candidate with necessary skill set suitable for this job position.

TABLE OF CONTENTS

LIST OF FIGURE

LIST OF ABBREVIATION

HTML	Hyper Text Markup Language
PHP	Hypertext Preprocessor
GNU	General Public License
TPO	Thyroid Peroxidase
GPC	General Purpose Computer
XSS	Cross Site Scripting
PHPID	PHP-Intrusion Detection System
DOS	Denial of Service
XML	Extensible Markup Language
RDBMS	Relational Database Management System
ODPC	Office of Disability Prevention for Children
DTP	Desktop Publishing
SSL	Secure Sockets Lay

CHAPTER 1

INTRODUCTION

1.1 OVER VIEW

Students can search for the material required for the selection process such as aptitude, reasoning etc and various websites for placement papers. Events happening in the college and the achievements of the student's i.e. selected students' details can be viewed by all the users. So, our project provides a facility of maintaining the details of the students, and gets the requested list of candidates for the companies who would like to recruit the people based on a given query. Placement Cell is a total management and informative system, which provides the up-to date information of all the students in a particular college. The colleges to overcome the difficulty in keeping records of hundreds and thousands of students and searching for a student eligible for recruitment criteria from the whole thing. It helps in effective and timely utilization of the hardware and the software resources. The home page contain various links such as links to login, various services like Events happened, achievements and recruiter details etc. The administrator will create the users and the users will use the accounts created by administrator. When the user enters into his respective page he can update his details, and the details are to be approved by the administrator. All the users have some common services like changing password, updating details, searching for details, checking the details, mailing to administrator, and reading the material uploaded by admin if the user is a student. Administrator has the services to add events and achievements and he can reply to the mails sent by users. He can upload materials, search for student details, and he has the right to approve the students.

1.2 OBJECTIVE

Placement Software acts as a bridge between student and software companies. In existing system there is the manual process to searching the job information. In this project the user can searching the job through this web site. The web pages about a company details are created dynamically based on the user id and password and links are provided to web pages containing information like company general details. Training and Placement also has the facility of viewing a detailed report regarding the student and recruitment of company informatio

CHAPTER 2

SYSTEM ANALYSIS

2.1 EXISTING SYSTEM

The earlier system is not computerized. All transactions in the system are done manually maintaining records. To make this laborious job simple the clients have to computerize the system. The management and all the departments that have been carrying out this job using manually makes the job more complicated and tedious most of the times. So, the best way is computerize computerization of the current environment. For example, in the earlier system placement officer has to collect student details for placements. Approving those student details takes lot of time. Placement officer and students have to consult each other directly if any information is needed. If any new company come for placements, placement officer and his staff has to search the student details and they have to find the eligible candidates for that particular company placement. Here searching for eligible candidates takes lots of time. And sometimes some candidates' details may be missed.

2.1.1 DISADVANTAGES

- It takes so much time for a placement officer to collect students' details and approving the details provided by them.
- Poor communication between students and placement officer, so here intimating about new placements is a hard task.
- Students may not know about company details. Here also poor communication provides a problem.

2.2 PROPOSED SYSTEM

The proposed system is fully computerized, which removes all the drawbacks of existing system. Proposed system is an online application that can be accessed throughout the organization and outside as well with proper login provided. This system can be used as an application for the TPO of the college to manage the student information with regards to placement. Students logging should be able to upload their information in the form of a CV. Company representatives logging in may also access/search any information put up by Students.

2.2.1 ADVANTAGES

- Admin officer can easily collect student’ details and Company details.
- Student can easily know about the company details.
- User friendly website.
- Reduction of paper work.
- Automation of existing manual information systems.
- Reduction of manual processing.
- Keep track of daily information exchange at the server by the administrator.
- Increase in processing and transfer speeds of information over the network.

CHAPTER 3

SYSTEM REQUIREMENTS

3.1 HARDWARE REQUIREMENTS

- CPU type : Intel Pentium 4
- Clock speed : 3.0 GHz
- Ram size : 512 MB
- Hard disk capacity : 40 GB
- Monitor type : 15 Inch color monitor
- Keyboard type : internet keyboard

3.2 SOFTWARE REQUIREMENTS

- Operating system : Windows 10
- Front End : PHP
- Back end : MYSQL server
- Tool : Macromedia Dreamweaver 8

3.3 SOFTWARE DESCRIPTION

PHP: Hypertext Preprocessor (the name is a recursive acronym) is a widely used, general-purpose scripting language that was originally designed for web development to produce dynamic web pages. For this purpose, PHP code is embedded into the HTML source document and interpreted by a web server with a PHP processor module, which generates the web page document. As a general-purpose programming language, PHP code is processed by an interpreter application in command-line mode performing desired operating system operations and producing program output on its standard output channel. It may also function as a graphical application. PHP is available as a processor for most modern web servers and as standalone interpreter on most operating systems and computing platforms. PHP was originally created by Rasmus Lerdorf in 1995 and has been in continuous development ever since. The main implementation of PHP is now produced by The PHP Group and serves as the de facto standard for PHP as there is no formal specification. PHP is free software

released under the PHP License, which is incompatible with the GNU General Public License (GPL) because restrictions exist regarding the use of the term PHP.

Hypertext refers to files linked together using hyperlinks, such as HTML (HyperText Markup Language) files. Preprocessing is executing instructions that modify the output. Below is a demonstration of the difference between HTML and PHP files.

Accessing an HTML Page

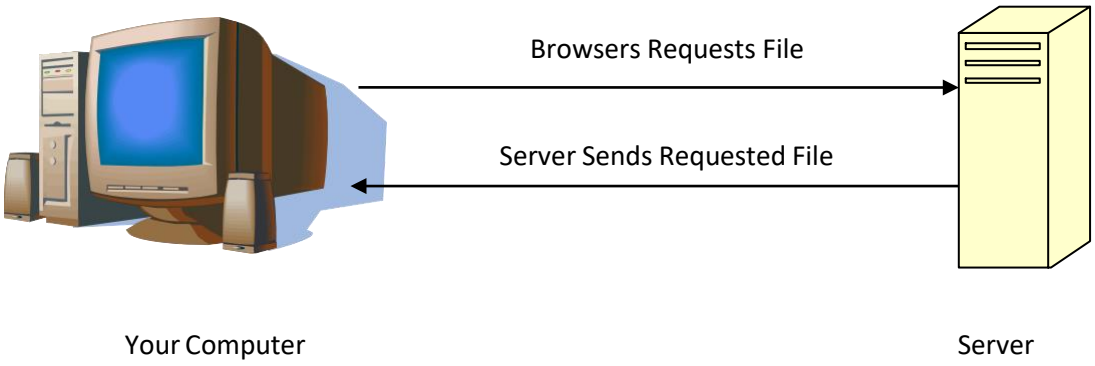


Fig 3 3.1 Accessing an HTML page

1. Your browser sends a request to that web page's server (computer) for the file (HTML or image) you wish to view.
2. The web server (computer) sends the file requested back to your computer.
3. Your browser displays the file appropriately.
4. If you request a PHP file (ends with ".php"), the server handles it differently.

Accessing a PHP Page

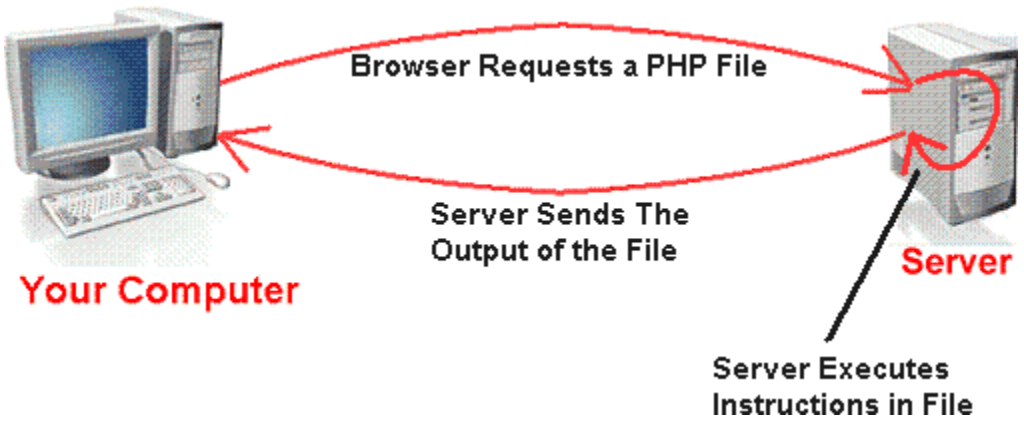


Fig:3.3.2 Accessing a PHP Page

Accessing a PHP Page

1. Your browser sends a request to that web page's server for the PHP file you wish to view.
2. The web server calls PHP to interpret and perform the operations called for in the PHP script.
3. The web server sends the output of the PHP program back to your computer.
4. Your browser displays the output appropriately.

Benefit of PHP

Because the server does processing, the output of PHP files changes when its input changes. For example, most of the pages on the Horticulture site have only two (2) PHP commands:

1. Include the header file that defines the links on the left, the banner, and the quick links at the top.
2. Include the footer file that displays the mission statement and Horticulture contact information.

Because including the files is performed every time the PHP file is accessed, when the header/footer files change, the new content will be immediately updated. In other words, if you add a new link, every page that includes the header will immediately display the new link.

Security

About 30% of all vulnerabilities listed on the National Vulnerability Database are linked to PHP. These vulnerabilities are caused mostly by not following best practice programming rules: technical security flaws of the language itself or of its core libraries are not frequent (23 in 2008, about 1% of the total). Recognizing that programmers make mistakes, some languages include taint checking to detect automatically the lack of input validation which induces many issues. Such a feature is being developed for PHP, but its inclusion in a release has been rejected several times in the past. There are advanced protection patches such as Suhosin and Hardening-Patch, especially designed for Web hosting environments.

PHPIDS adds security to any PHP application to defend against intrusions. PHPIDS detects attacks based on cross-site scripting (XSS), SQL injection, header injection, directory traversal, remote file execution, remote file inclusion, and denial-of-service (DoS)

Syntax

The PHP interpreter only executes PHP code within its delimiters. Anything outside its delimiters is not processed by PHP (although non-PHP text is still subject to control structures described in PHP code). The most common delimiters are <?php to open and ?> to close PHP sections. <script language="php"> and </script> delimiters are also available, as are the shortened forms <?or<?= (which is used to echo back a string or variable) and ?> as well as ASP-style short forms <% or <%= and %>. While short delimiters are used, they make script files less portable as support for them can be disabled in the PHP configuration, and so they are discouraged. The purpose of all these delimiters is to separate PHP code from non-PHP code, including HTML.

The first form of delimiters, <?php and ?>, in XHTML and other XML documents, creates correctly formed XML 'processing instructions'. This means that the resulting mixture of PHP code and other markup in the server-side file is itself well-formed XML.

Variables are prefixed with a dollar symbol, and a type does not need to be specified in advance. Unlike function and class names, variable names are case sensitive. Both double-quoted ("") and here-doc strings provide the ability to interpolate a variable's value into the string. PHP treats newlines as whitespace in the manner of a free-form language (except when inside string quotes), and statements are terminated by a semicolon. PHP has three types of comment syntax: /* */ marks block and inline comments; // as well as # are used for one-line comments. The echo statement is one of several facilities PHP provides to output text, e.g., to a Web browser.

In terms of keywords and language syntax, PHP is similar to most high level languages that follow the C style syntax. if conditions, for and while loops, and function returns are similar in syntax to languages such as C, C++, Java and Perl.

Data types

PHP stores whole numbers in a platform-dependent range, either a 64-bit or 32-bit signedinteger equivalent to the C-language long type. Unsigned integers are converted to signed values in certain situations; this behavior is different from other programming languages. Integer variables can be assigned using decimal (positive and negative), octal, and hexadecimal notations. Floating point numbers are also stored in a platform-specific range. They can be specified using floating point notation, or two forms of scientific notation. PHP has a native Boolean type that is similar to the native Boolean types in Java and C++. Using the Boolean type conversion rules, non-zero values are interpreted as true and zero as false, as in Perl and C++. The null data type represents a variable that has no value. The only value in the null data type is NULL. Variables of the "resource" type represent references to resources from external sources. These are typically created by functions from a particular extension, and can only be processed by functions from the same extension; examples include file, image, and database resources. Arrays can contain elements of any type that PHP can handle, including resources, objects, and even other arrays. Order is preserved in lists of values and in hashes with both keys and values, and the two can be intermingled. PHP also supports strings, which can be used with single quotes, double quotes, nowdoc or heredoc syntax.

Functions

PHP has hundreds of base functions and thousands more via extensions. These functions are well documented on the PHP site; however, the built-in library has a wide variety of naming conventions and inconsistencies. PHP currently has no functions for thread programming, although it does support multi-process programming on POSIX systems.

MY SQL

MySQL is the world's most used open source relational database management system (RDBMS) as of 2008 that run as a server providing multi-user access to a number of databases. The MySQL development project has made its source code available under the terms of the GNU General Public License, as well as under a variety of proprietary agreements. MySQL was owned and sponsored by a single for-profit firm, the Swedish company MySQL AB, now owned by Oracle Corporation.

MySQL is a popular choice of database for use in web applications, and is a central component of the widely used LAMP open source web application software stack—LAMP is an acronym for "Linux, Apache, MySQL, Perl/PHP/Python." Free-software-open source projects that require a full-featured database management system often use MySQL.

For commercial use, several paid editions are available, and offer additional functionality. Applications which use MySQL databases include: TYPO3, Joomla, Word Press, phpBB, MyBB, Drupal and other software built on the LAMP software stack. MySQL is also used in many high-profile, large-scale World Wide Web products, including Wikipedia, Google(though not for searches), ImagebookTwitter, Flickr, Nokia.com, and YouTube.

Interimages

MySQL is primarily an RDBMS and ships with no GUI tools to administer MySQL databases or manage data contained within the databases. Users may use the included command line tools, or use MySQL "front-ends", desktop software and web applications that create and manage MySQL databases, build database structures, back up data, inspect status, and work with data records. The official set of MySQL front-end tools, MySQL Workbench is actively developed by Oracle, and is freely available for use.

Graphical

The official MySQL Workbench is a free integrated environment developed by MySQL AB, that enables users to graphically administer MySQL databases and visually design database structures. MySQL Workbench replaces the previous package of software, MySQL GUI Tools. Similar to other third-party packages, but still considered the authoritative MySQL frontend, MySQL Workbench lets users manage database design & modeling, SQL development (replacing MySQL Query Browser) and Database administration (replacing MySQL Administrator).

MySQL Workbench is available in two editions, the regular free and open source Community Edition which may be downloaded from the MySQL website, and the proprietary Standard Edition which extends and improves the feature set of the Community Edition.

Commandline

MySQL ships with some command line tools. Third-parties have also developed tools to manage a MySQL server, some listed below.

- Maatkit - a cross-platform toolkit for MySQL, PostgreSQL and Memcached, developed in Perl Maatkit can be used to prove replication is working correctly, fix corrupted data, automate repetitive tasks, and speed up servers. Maatkit is included with several GNU/Linux distributions such as CentOS and Debian and packages are available for Programming

MySQL works on many different system platforms, including AIX, BSDi, FreeBSD, HP-UX, eComStation, i5/OS, IRIX, Linux, Mac OS X, Microsoft Windows, NetBSD, Novell NetWare, OpenBSD, OpenSolaris, OS/2 Warp, QNX, Solaris, Symbian, SunOS, SCO OpenServer, SCO UnixWare, Sanos and Tru64. A port of MySQL to OpenVMS also exists.[32]

MySQL is written in C and C++. Its SQL parser is written in yacc, and a home-brewed lexical analyzer. Many programming languages with language-specific APIs include libraries for accessing MySQL databases. These include MySQL Connector/Net for integration with Microsoft's Visual Studio (languages such as C# and VB are most commonly used) and the JDBC driver for Java. In addition, an ODBCinterimage called MyODBC allows additional programming languages that support the ODBC interimage to communicate with a MySQL database, such as ASP or ColdFusion. The HTSQL - URL-based query method also ships with a MySQL adapter, allowing direct interaction between a MySQL database and any web client via structured URLs.

Features

As of April 2009, MySQL offered MySQL 5.1 in two different variants: the open source MySQL Community Server and the commercial Enterprise Server. MySQL 5.5 is offered under the same licences. They have a common code base and include the following features:

- A broad subset of ANSI SQL 99, as well as extensions
- Cross-platform support
- Stored procedures
- Triggers

- Cursors
- Updatable Views
- Information schema
- Strict mode (ensures MySQL does not truncate or otherwise modify data to conform to an underlying data type, when an incompatible value is inserted into that type)
- X/Open XAdistributed transaction processing (DTP) support; two phase commit as part of this, using Oracle's InnoDB engine
- Independent storage engines (MyISAM for read speed, InnoDB for transactions and referential integrity, MySQL Archive for storing historical data in little space)
- Transactions with the InnoDB, and Cluster storage engines; savepoints with InnoDB
- SSL support
- Query caching
- Sub-SELECTs (i.e. nested SELECTs)
- Replication support (i.e. Master-Master Replication & Master-Slave Replication) with one master per slave, many slaves per master, no automatic support for multiple masters per slave.
- Full-text indexing and searching using MyISAM engine
- Embedded database library
- Unicode support (however prior to 5.5.3 UTF-8 and UCS-2 encoded strings are limited to the BMP, in 5.5.3 and later use utf8mb4 for full unicode support)
- ACID compliance when using transaction capable storage engines (InnoDB and Cluster)
- Partititoned tables with pruning of partitions in optimiser
- Shared-nothing clustering through MySQL Cluster
- Hot backup (via mysqlhotcopy) under certain conditions
- Multiple storage engines, allowing one to choose the one that is most effective for each table in the application (in MySQL 5.0, storage engines must be compiled in; in MySQL 5.1, storage engines can be dynamically loaded at run time): Native storage engines (MyISAM, Falcon, Merge, Memory (heap), Federated, Archive, CSV, Blackhole, Cluster, EXAMPLE, Maria, and InnoDB, which was made the default as of 5.5). Partner-developed storage engines (solidDB, NitroEDB, ScaleDB, TokuDB, Infobright (formerly Brighthouse), Kickfire, XtraDB, IBM DB2). InnoDB used to be a partner-developed storage engine, but with recent acquisitions, Oracle now owns both MySQL core and InnoDB.

CHAPTER 4

SYSTEM DESIGN

4.1 SYSTEM ARCHITECTURE

An allocated arrangement of physical elements which provides the design solution for a consumer A system architecture or systems architecture is the conceptual model that defines the structure, behavior, and more views of a system. An architecture description is a formal description and representation of a system, organized in a way that supports reasoning about the structures and behaviors of the system. System architecture can comprise system components, the externally visible properties of those components, the relationships (e.g. the behavior) between them. It can provide a plan from which products can be procured, and systems developed, that will work together to implement the overall system. There have been efforts to formalize languages to describe system architecture; collectively these are called architecture description languages (ADLs).

Various organizations define systems architecture in different ways, including:

- Product or life-cycle process intended to satisfy the requirements of the functional architecture and the requirements baseline.
- Architecture comprises the most important, pervasive, top-level, strategic inventions, decisions, and their associated rationales about the overall structure (i.e., essential elements and their relationships) and associated characteristics and behavior.
- If documented, it may include information such as a detailed inventory of current hardware, software and networking capabilities; a description of long-range plans and priorities for future purchases, and a plan for upgrading and/or replacing dated equipment and software

  The composite of the design architectures for products and their life-cycle processes.

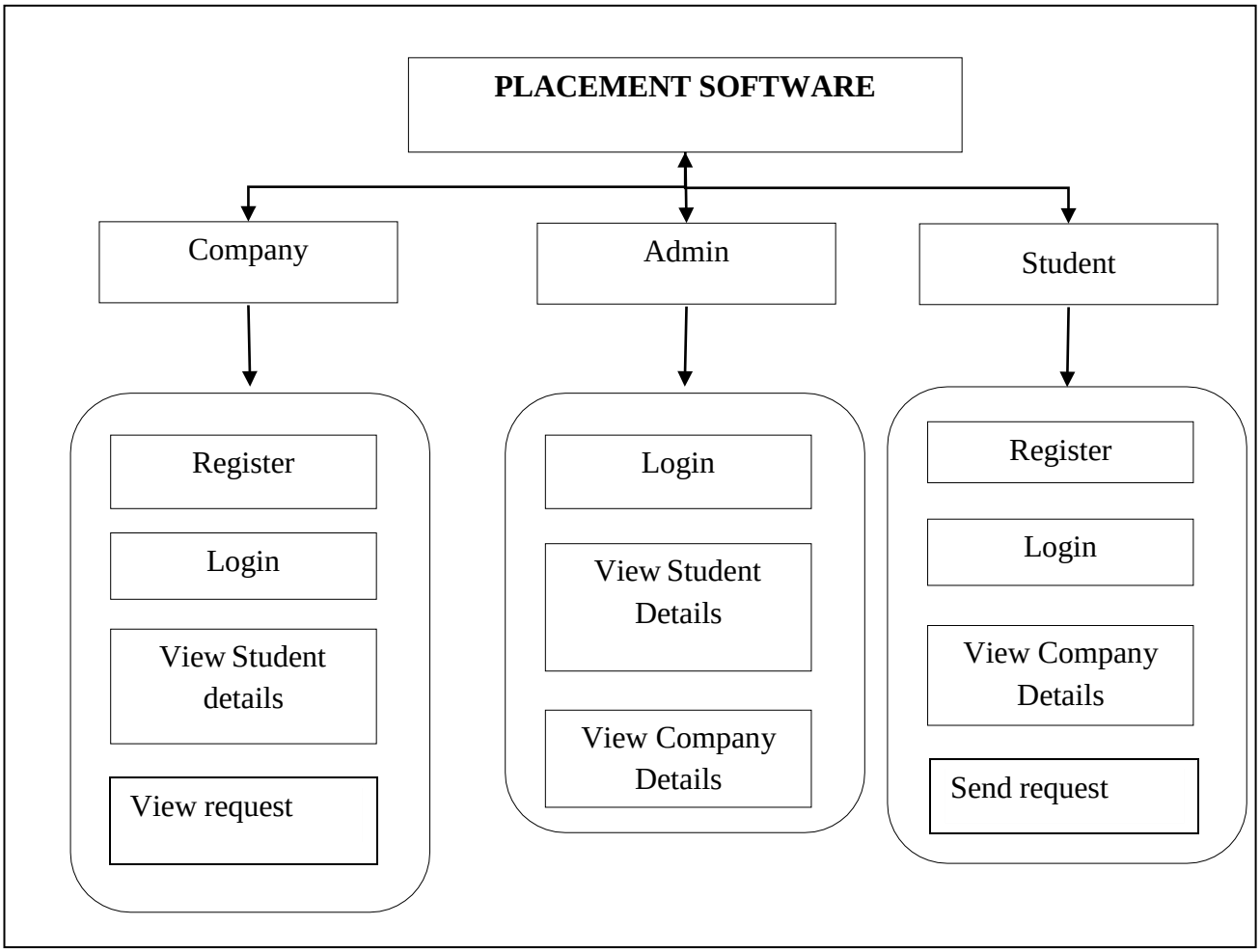


Fig: 4.1.1 System architecture

CHAPTER 5

MODULES DESCRIPTION

5.1 MODULES

Admin

- Login
- View Company Details
- View Student Details

Company

- Register
- Login
- View Student Details
- View Request

Student

- Register
- Login
- View Company Details
- Send Request

5.2 MODULE DESCRIPTION

Admin

- Login

  In this module, the admin can login in the system using his/her username and password.
- View Company Details

  In this module helps admin to view company details. Company Details such as company name, address, contact number etc.
- View Student Details

  In this module helps admin to view the student details. Student Details such as student name, department etc.

Company

- Register

  There is registration form available where new company can create their account by providing required information to the system. The registration form details are like company name, email, mobile number, address, and etc. These details are stored in the database. And then can getting to the username and password in the system.

- Login

  In this module, the company can login the system using his/her username and password.

- View Student Details

  In this module helps company to view the student details. Student Details such as student name, department etc.

- View Request

  In this module the company can view the particular student request. The company admin can accept or reject the student request. In this process the alert notification is automatically sent to the student.

Student

- Register

  There is registration form available where new student can create their account by providing required information to the system. The registration form details are like student name, email, gender, mobile number, address, and etc. These details are stored in the database. And then can getting to the username and password in the system.

- Login

  In this module, the student can login the system using his/her username and password.

- View Company Details

  In this module helps student to view company details. Company Details such as company name, address, contact number etc.

- Send Request

  In this module the student can sent the request to particular company. The student request is viewed by company admin and he/she can accept or reject the request, whatever it is the notification is received by student.

CHAPTER 6

CONCLUSION AND FUTURE ENHANCEMENT

6.1 CONCLUSION

The introduction, problem definition of the project has been completed successfully to college Web based Placement Software by maintaining the student details related to placement in an efficient manner. When campus selections are conducted the students should provide their curriculum vitae to the concern officer for attending the campus interviews. This routine process is maintained manually, like maintenance of their resumes in papers. This can be automated by designing software. So, our project provides a facility of maintaining the details of the students, and gets the requested list of candidates for the companies who would like to recruit the people based on a given query.

6.2 FUTURE ENHANCEMENT

In future we can develop this project in android application with extra features like student query process with company admin and company location tracking system.

CHAPTER 7

APPENDICES

7.1 SAMPLE CODING

Dbconnect.php

```
<?php
$connect=mysql_connect("localhost","root","");
mysql_select_db("train_place",$connect);
?> Index.php<!DOCTYPE HTML>
<html>
<head>
  <title>Trani&place </title>
  <meta name="description" content="website description" />
  <style type="text/css">
<!--
.style1 {color: #FF0000}
-->
  </style>
</head>
<body>
  <table width="100%" border="0">
    <tr>
      <th height="73" bgcolor="#ccffff" scope="col"><h1>Training & Placement</h1>
      </th>
```

```
  </tr>
 </table>
 <table width="100%" border="0">
  <tr>
   <th scope="col"><a href="index.php">Home</a></th>
   <th scope="col"><a href="admin.php">Admin Login</a></th>
   <th scope="col"><a href="company.php">Company Login</a></th>
   <th scope="col"><a href="student.php">Student Login</a></th>
   <th scope="col"><a href="#">About Us </a></th>
  </tr>
 </table>
  <p> </p>
 <div id="content">
  <img src="images\1.png" width="1420" height="400">
 <marquee scrollamount="10"><h1><font color="red">Training & placement</h1></marquee>
<table width="100%" border="0">
  <tr>
   <th height="73" bgcolor="#ccffff" scope="col"><p>copyrights@2019 placement division</p>
   </th>
  </tr>
</table>
```

```
</body>
</html>Admin.php
<?php
	include("dbconnect.php");
	extract($_POST);
	session_start();
if(isset($_POST['btn']))
{
$qry=mysql_query("select * from admin where name='$uname' && psw='$password'");
$num=mysql_num_rows($qry);
if($num==1)
{
echo "<script>alert('welcome to admin login')</script>";
header("location:adminhome.php");
}
else
{
echo "<script>alert('User Name Password Wrong ....')</script>";
}
}
?>
<html>
<head>
```

```
  <title>Trani&place </title>
  <meta name="description" content="website description" />
  <style type="text/css">
<!--
.style1 {color: #FF0000}
-->
  </style>
</head>
<body>
  <table width="100%" border="0">
    <tr>
      <th height="73" bgcolor="#ccffff" scope="col"><h1>Training & Placement</h1>
      </th>
    </tr>
  </table>
  <table width="100%" border="0">
    <tr>
      <th scope="col"><a href="index.php">Home</a></th>
      <th scope="col"><a href="admin.php">Admin Login</a></th>
      <th scope="col"><a href="company.php">Company Login</a></th>
      <th scope="col"><a href="student.php">Student Login</a></th>
      <th scope="col"><a href="#">About Us </a></th>
    </tr>
```

```
</table>
<p> </p>
<div id="content">
<img src="images\1.png" width="1420" height="400">
<form id="form1" name="form1" method="post" action="">
<table width="48%" border="0" align="center">
<tr>
<td colspan="2" rowspan="1"><div align="center" class="style1"><strong><font size="+1">Admin Login</font> </div></td>
</tr>
<tr>
<td width="48%"> </td>
<td width="52%"> </td>
</tr>
</tr>
<tr>
<td height="31"align="center"><span class="style2"><strong>User Name </strong></span></td>
<td><label>
<input name="uname" type="text" id="uname" />
</label></td>
</tr>
<tr>
```

```
        <td height="44" align="center"><span class="style2"><strong>Password</strong></span></td>
        <td><label>
          <input name="password" type="password" id="password" />
        </label></td>
      </tr>
      <tr>
        <td> </td>
        <td rowspan="2"><label>
          <input name="btn" type="submit" id="btn" value="Login" />
          <input type="reset" name="Submit2" value="Cancel" />
        </label></td>
      </tr>
  </table>
</form>
<div>  </div>
 <marquee scrollamount="10"><h1><font color="red">Training & placement</h1></marquee>
<table width="100%" border="0">
    <tr>
     <th height="73" bgcolor="#ccffff" scope="col"><p>copyrights@2019 placement division</p>
     </th>
    </tr>
```

```
</table></body>
</html>Register.php<?php
        include("dbconnect.php");
        extract($_POST);
        session_start();
                if(isset($_POST['btn']))
        {
        $imgpath=$_FILES['file']['name'];
          $errors= array();
     $fname = $_FILES['file']['name'];
    $file_tmp =$_FILES['file']['tmp_name'];
     move_uploaded_file($_FILES['file']['tmp_name'],"resume/".$fname);
   $max_qry = mysql_query("select max(id) from register");
        $max_row = mysql_fetch_array($max_qry);
        $id=$max_row['max(id)']+1;
 $qry=mysql_query("insert into register values('$id','$name','$gender','$age','$email','$phone','$quali','$fname','$address','$dept','$reg')");
if($qry)
{
header("location:userhome.php");
}
else
```

```
{
echo "<script>alert('Data Not Save');</script>";
}
}
?>
<html>
<head>
  <title>Trani&place </title>
  <meta name="description" content="website description" />
  <style type="text/css">
<!--
.style1 {color: #FF0000}
-->
  </style>
</head>
<body>
  <table width="100%" border="0">
    <tr>
      <th height="73" bgcolor="#ccffff" scope="col"><h1>Training & Placement</h1>
      </th>
    </tr>
  </table>
  <table width="100%" border="0">
```

```
  <tr>
    <th scope="col"><a href="index.php">Home</a></th>
    <th scope="col"><a href="admin.php">Admin Login</a></th>
    <th scope="col"><a href="company.php">Company Login</a></th>
    <th scope="col"><a href="student.php">Student Login</a></th>
    <th scope="col"><a href="#">About Us </a></th>
  </tr>
 </table>
 <p> </p>
 <div id="content">
  <img src="images\1.png" width="1420" height="400">
<form id="f1" name="f1" method="post" action="#" enctype="multipart/form-data" >
 <table width="38%" border="0" align="center">
                              <tr>
    <td width="38%" height="35"> </td>
    <td width="62%"> </td>
  </tr>
  <tr>
    <td colspan="2" align="center" ><div class="style5"><h3>New User Registation</h></div></td>
        </tr>
          <tr>
    <td> </td>
```

```
  <td> </td>
</tr>
<tr>
  <td>Name</td>
  <td>
    <input name="name" type="text" id="name" />
  </td>
      </tr>
        <tr>
  <td> </td>
  <td> </td>
</tr>
<tr>
  <td>Gender</td>
  <td><input name="gender" type="radio" value="male" />
    Male
      <input name="gender" type="radio" value="female" />
      Female</td>
</tr>
        <tr>
  <td> </td>
  <td> </td>
</tr>
```

```
<tr>
  <td>Age</td>
  <td>
    <input name="age" type="text" id="age"  />
 </td>
</tr>
        <tr>
   <td> </td>
  <td> </td>
</tr>
<tr>
  <td>Email Id</td>
  <td><input name="email" type="text" id="email" /></td>
</tr>
        <tr>
 <td> </td>
  <td> </td>
</tr>
          <tr>
  <td>Phone Number</td>
  <td><input name="phone" type="text" id="phone"  /></td>
 </tr>
        <tr>
```

```
  <td> </td>
  <td> </td>
</tr>
        <tr>
   <td>Qualifaication</td>
  <td><input type="text" name="quali" id="quali"></td>
</tr>
        <tr>
  <td> </td>
  <td> </td>
</tr>
        <tr>
    <td>Upload Resume</td>
  <td><input type="file" name="file" id="file"></td>
</tr>
        <tr>
  <td> </td>
  <td> </td>
</tr>
          <tr>
  <td>Address</td>
  <td><textarea name="address" id="address"></textarea></td>
   </tr>
```

```
<tr>
    <td> </td>
    <td> </td>
  </tr>
        <tr>
    <td>Department</td>
    <td><input name="dept" type="text" id="dept" /></td>
  </tr>
      <tr>
     <td> </td>
    <td> </td>
  </tr>
  <tr>
  <td>Register Number</td>
    <td><input name="reg" type="password" id="reg" /></td>
     </tr>
       <tr>
    <td> </td>
    <td> </td>
  </tr>
              <tr>
    <td> </td>
    <td><input name="btn" type="submit" id="btn" value="Submit" />
```

```
    <input type="reset" name="Submit2" value="Reset" /></td>
  </tr>
 </table>
</form>
<table width="100%" border="0">
  <tr>
    <th height="73" bgcolor="#ccffff" scope="col"><p>copyrights@2019 placement division</p>
    </th>
  </tr>
</table></body>
</html>Student.php
<?php
        include("dbconnect.php");
        extract($_POST);
        session_start();
if(isset($_POST['btn']))
{
$qry=mysql_query("select * from register where name='$uname' && reg='$password'");
$num=mysql_num_rows($qry);
if($num==1)
{
```

```
$qry=mysql_query("select * from register where name='$uname' && reg='$password'");
$row=mysql_fetch_assoc($qry);
$_SESSION['uid']=$row['id'];
echo "<script>alert('Welcome to User Page')</script>";
header("location:shome.php");
}
ele
{
echo "<script>alert('User Name Password Wrong ....')</script>";
}
}
?>
<html>
<head>
  <title>Trani&place </title>
  <meta name="description" content="website description" />
  <style type="text/css">
<!--
.style1 {color: #FF0000}
-->
  </style>
</head>
<body>
```

```
<table width="100%" border="0">
  <tr>
    <th height="73" bgcolor="#ccffff" scope="col"><h1>Training & Placement</h1>
    </th>
  </tr>
</table>
<table width="100%" border="0">
  <tr>
    <th scope="col"><a href="index.php">Home</a></th>
    <th scope="col"><a href="admin.php">Admin Login</a></th>
    <th scope="col"><a href="company.php">Company Login</a></th>
    <th scope="col"><a href="student.php">Student Login</a></th>
    <th scope="col"><a href="#">About Us </a></th>
  </tr>
</table>
<p> </p>
<div id="content">
 <img src="images\1.png" width="1420" height="400">
<form id="form1" name="form1" method="post" action="">
            <table width="46%" border="0" align="center">
        <tr>
          <td colspan="2" rowspan="1"><div align="center" class="style1"><strong><font
size="+1">Student Login</font> </div></td>
```

```
                </tr>
                        <tr>
                <td width="48%"> </td>
                    <td width="52%"> </td>
                        </tr>
        </tr>
        <tr>
          <td height="31"align="center"><span class="style2"><strong>Student Name
</strong></span></td>
          <td><label>
            <input name="uname" type="text" id="uname" />
          </label></td>
        </tr>
        <tr>
          <td height="44" align="center"><span class="style2"><strong>Register
Number</strong></span></td>
          <td><label>
            <input name="password" type="password" id="password" />
          </label></td>
        </tr>
        <tr>
          <td> </td>
          <td rowspan="2"><label>
```

```
          <input name="btn" type="submit" id="btn" value="Login" />
          <input type="reset" name="Submit2" value="Cancel" />
        </label></td>
      </tr>
                <tr>
        <td> </td>
        <td><label>
         <a href="register.php">New Student Register</a>
        </label></td>
      </tr>
  </table>
</form><table width="100%" border="0">
   <tr>
    <th height="73" bgcolor="#ccffff" scope="col"><p>copyrights@2019 placement division</p>
    </th>
   </tr>
</table></body>
</html>
```

Company.php

```
<?php
        include("dbconnect.php");
        extract($_POST);
```

```
        session_start();
if(isset($_POST['btn']))
{
$qry=mysql_query("select * from cregister where uname='$uname' && psw='$password'");
$num=mysql_num_rows($qry);
if($num==1)
{
$qry=mysql_query("select * from cregister where uname='$uname' && psw='$password'");
$row=mysql_fetch_assoc($qry);
$_SESSION['cname']=$row['cname'];
echo "<script>alert('Welcome To Company Home')</script>";
header("location:companyhome.php");
}
else
{
echo "<script>alert('User Name Password Wrong ....')</script>";
}
}
?>
<html>
<head>
  <title>Trani&place </title>
  <meta name="description" content="website description" />
```

```
 <style type="text/css">
<!--
.style1 {color: #FF0000}
-->
 </style>
</head>
<body>
 <table width="100%" border="0">
  <tr>
   <th height="73" bgcolor="#ccffff" scope="col"><h1>Training & Placement</h1>
   </th>
  </tr>
 </table>
 <table width="100%" border="0">
  <tr>
   <th scope="col"><a href="index.php">Home</a></th>
   <th scope="col"><a href="admin.php">Admin Login</a></th>
   <th scope="col"><a href="company.php">Company Login</a></th>
   <th scope="col"><a href="student.php">Student Login</a></th>
   <th scope="col"><a href="#">About Us </a></th>
  </tr>
 </table>
```

```
 <p> </p>
 <div id="content">
  <img src="images\1.png" width="1420" height="400">
<br />
<br />
<br />
<br /><form id="form1" name="form1" method="post" action="">
            <table width="46%" border="0" align="center">
       <tr>
         <td colspan="2" rowspan="1"><div align="center" class="style1"><strong><font size="+1">Company Login</font> </div></td>
                  </tr>
                        <tr>
                  <td width="48%"> </td>
                     <td width="52%"> </td>
                        </tr>
       </tr>
       <tr>
         <td height="31"align="center"><span class="style2"><strong>User Name </strong></span></td>
         <td><label>
           <input name="uname" type="text" id="uname" />
         </label></td>
```

```
        </tr>
        <tr>
          <td height="44" align="center"><span class="style2"><strong>Password</strong></span></td>
          <td><label>
            <input name="password" type="password" id="password" />
          </label></td>
        </tr>
        <tr>
          <td> </td>
          <td rowspan="2"><label>
            <input name="btn" type="submit" id="btn" value="Login" />
            <input type="reset" name="Submit2" value="Cancel" />
          </label></td>
        </tr>
 <tr>
          <td> </td>
          <td><label>
           <a href="cregister.php">Company Register</a>
          </label></td>
        </tr>
  </table>
</form><br />
```

```
<br />

<br />

<br />

<table width="100%" border="0">

  <tr>

    <th height="73" bgcolor="#ccffff" scope="col"><p>copyrights@2019 placement division</p>

    </th>

  </tr>

</table></body>

</html>
```

7.2 SCREENSHOTS

Fig:B.1 Home page

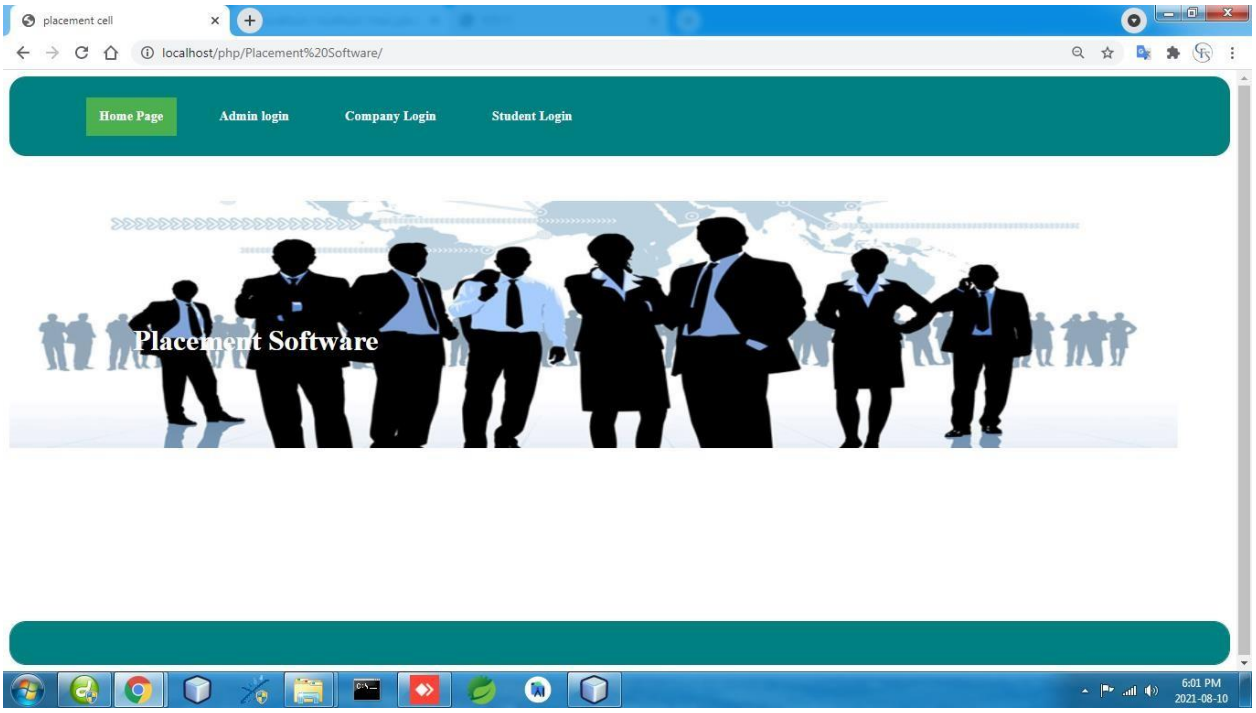


Fig:B.2 Admin Login page

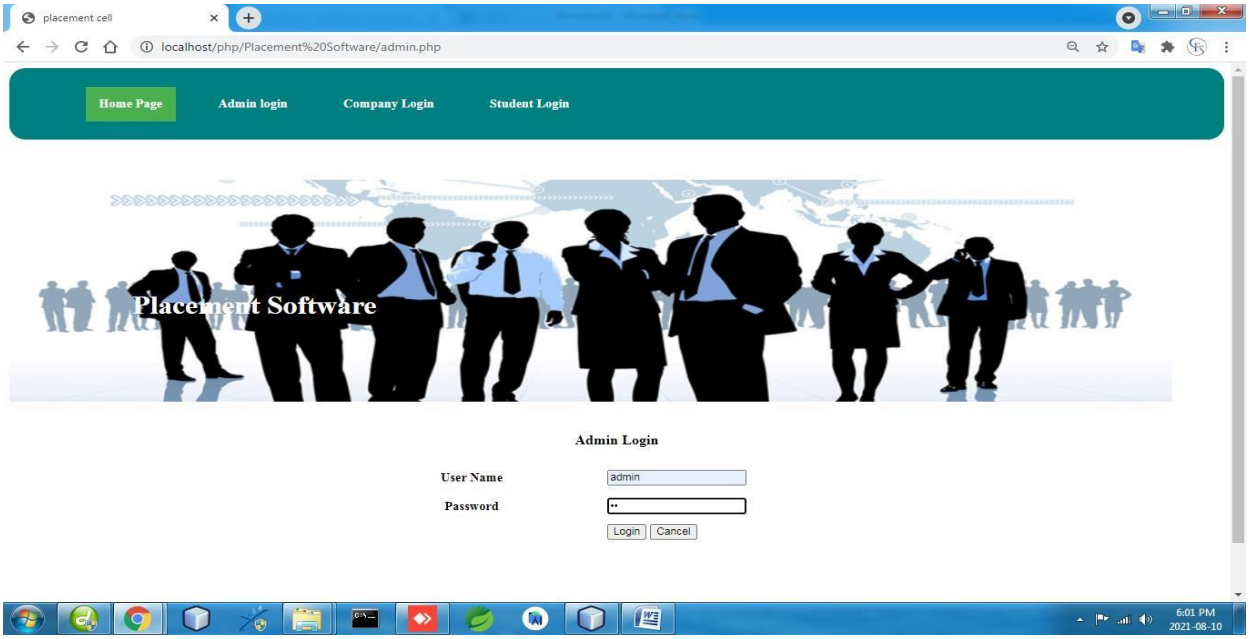


Fig:B.3 Selected students

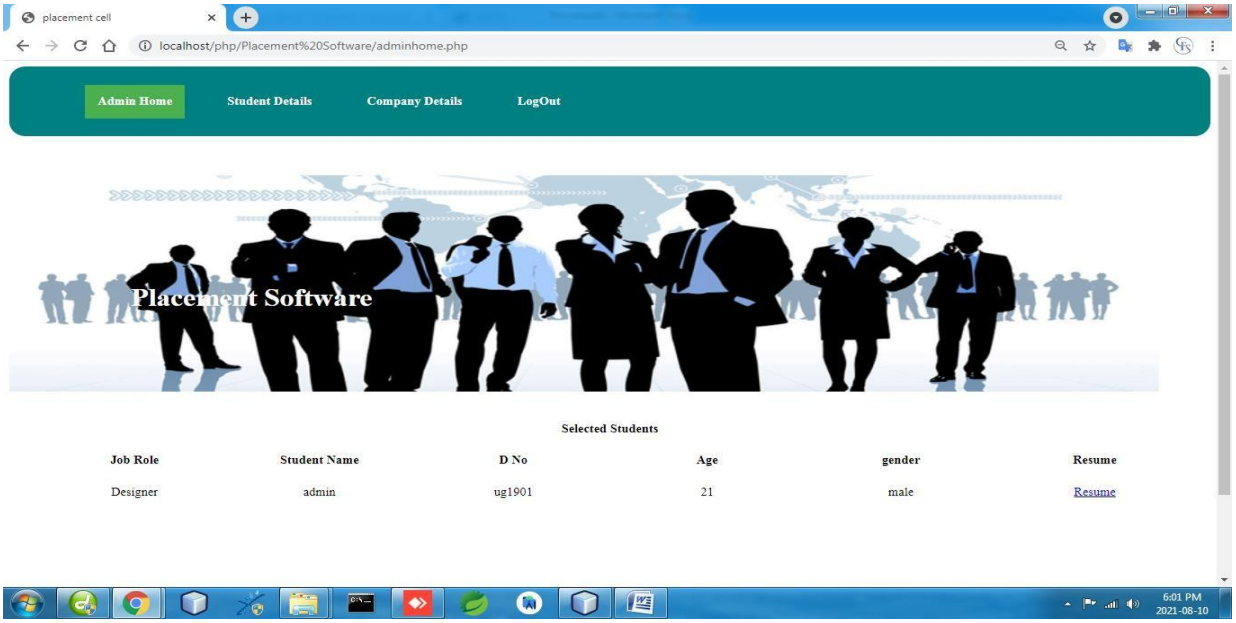


Fig:B.4 Student Details

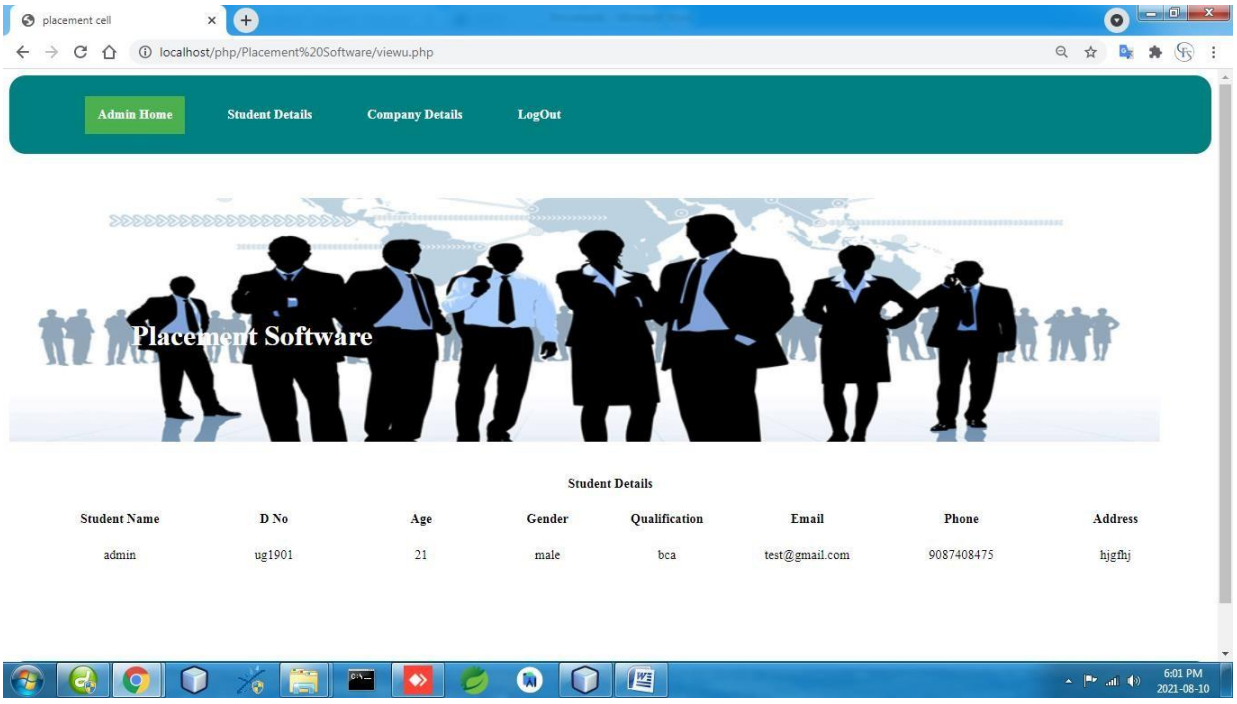


Fig: B.5 Company Details

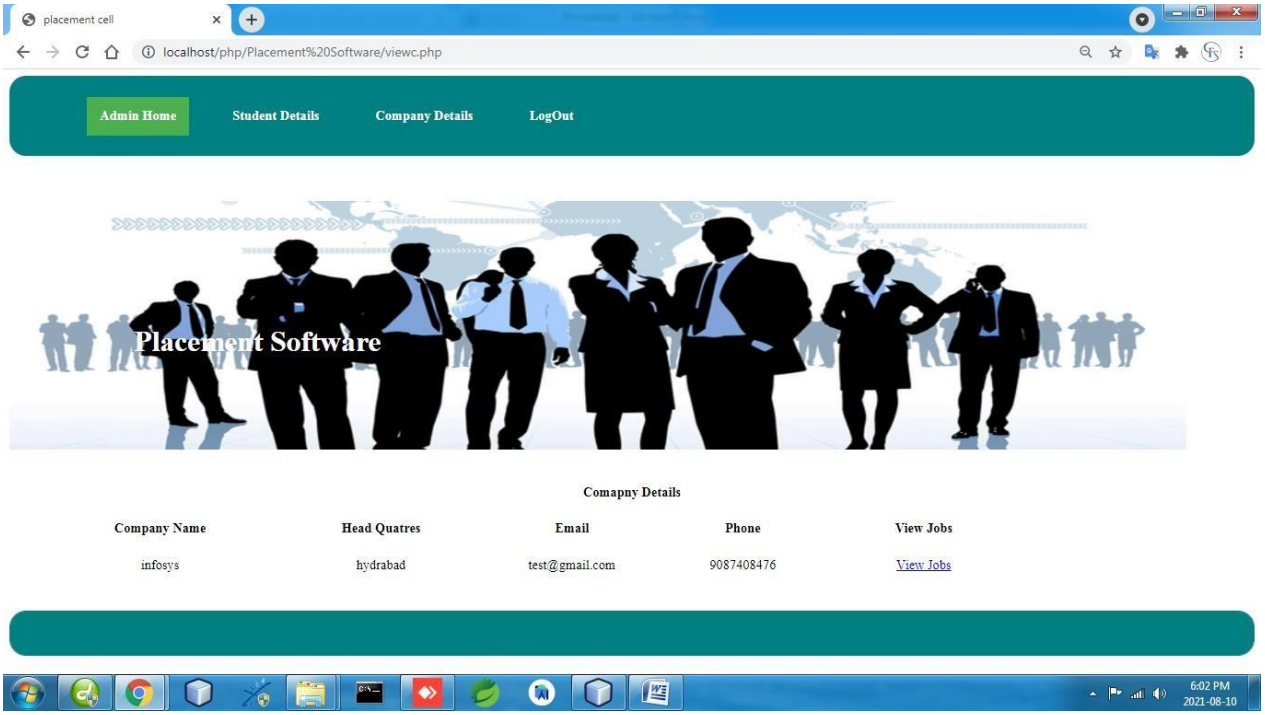


Fig:B.6 Company Jobs Details

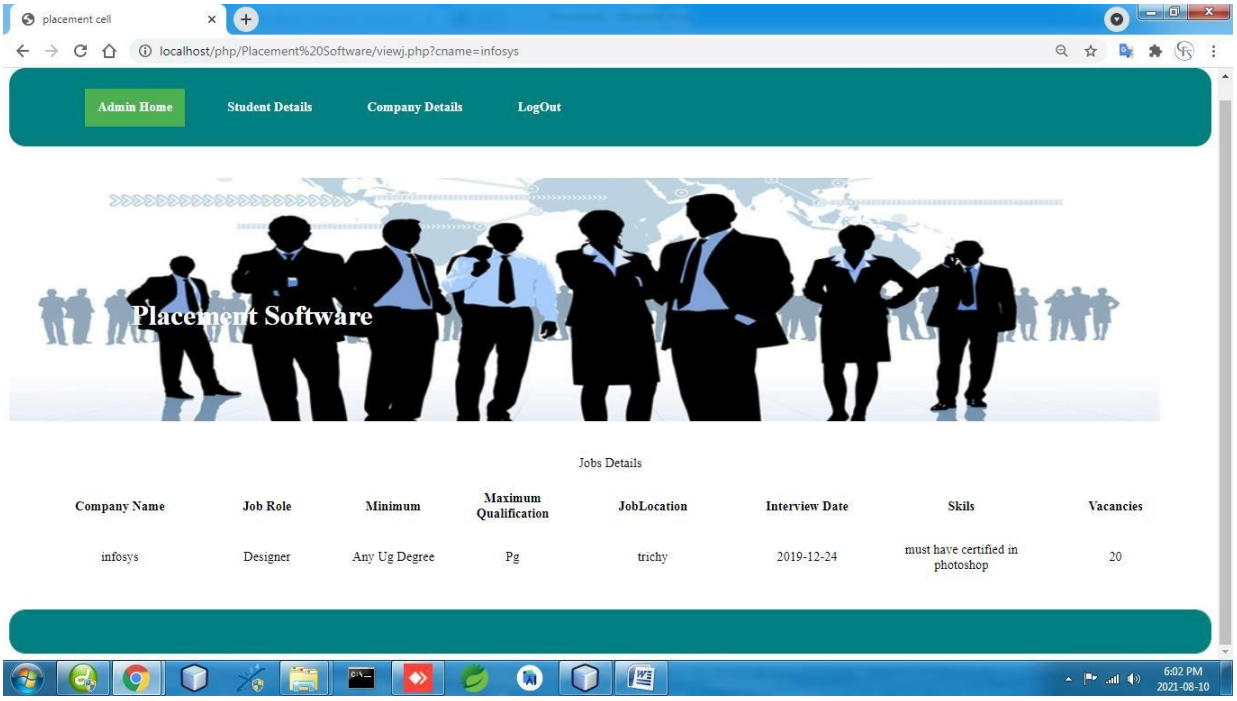


Fig: B.7 Company Registration

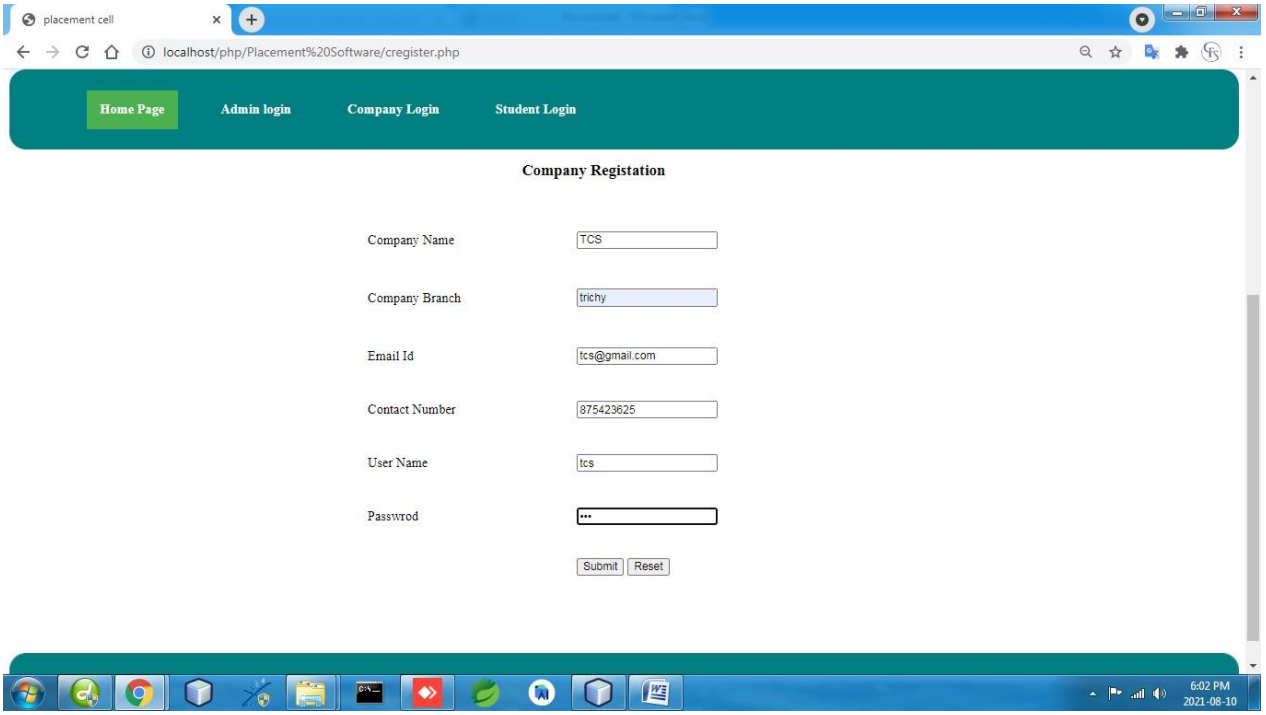


Fig:B.8 Insert Company Details

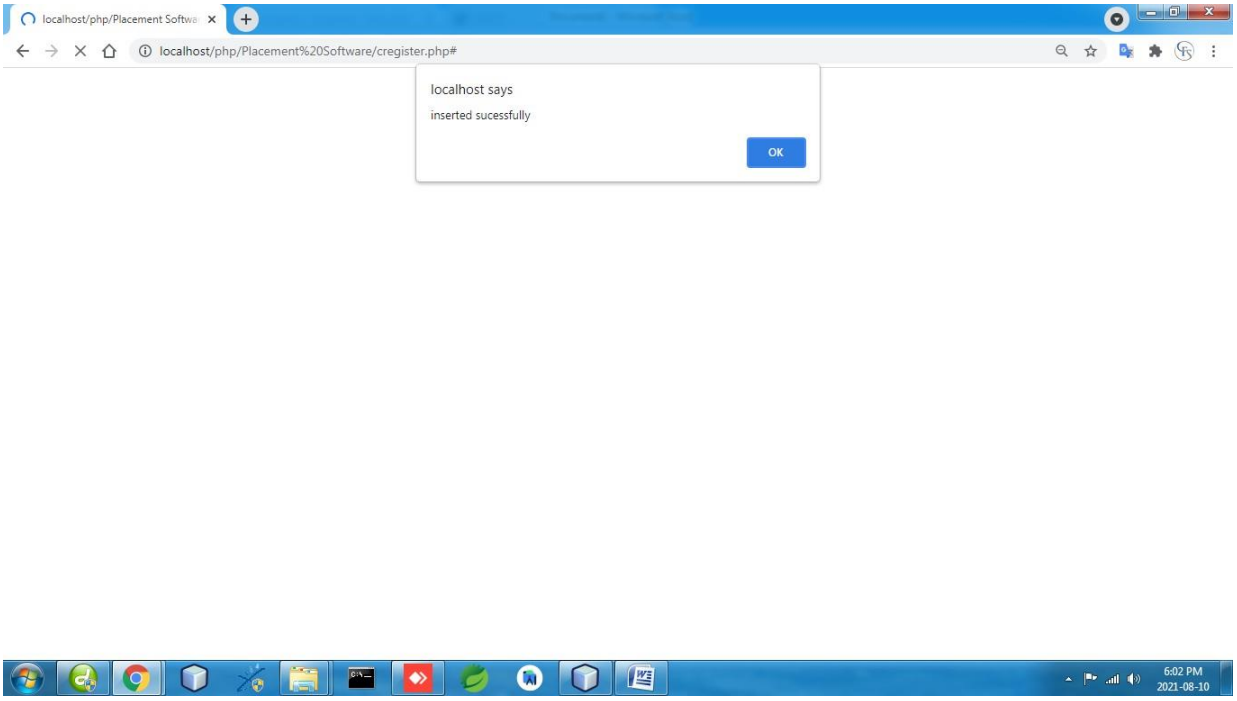
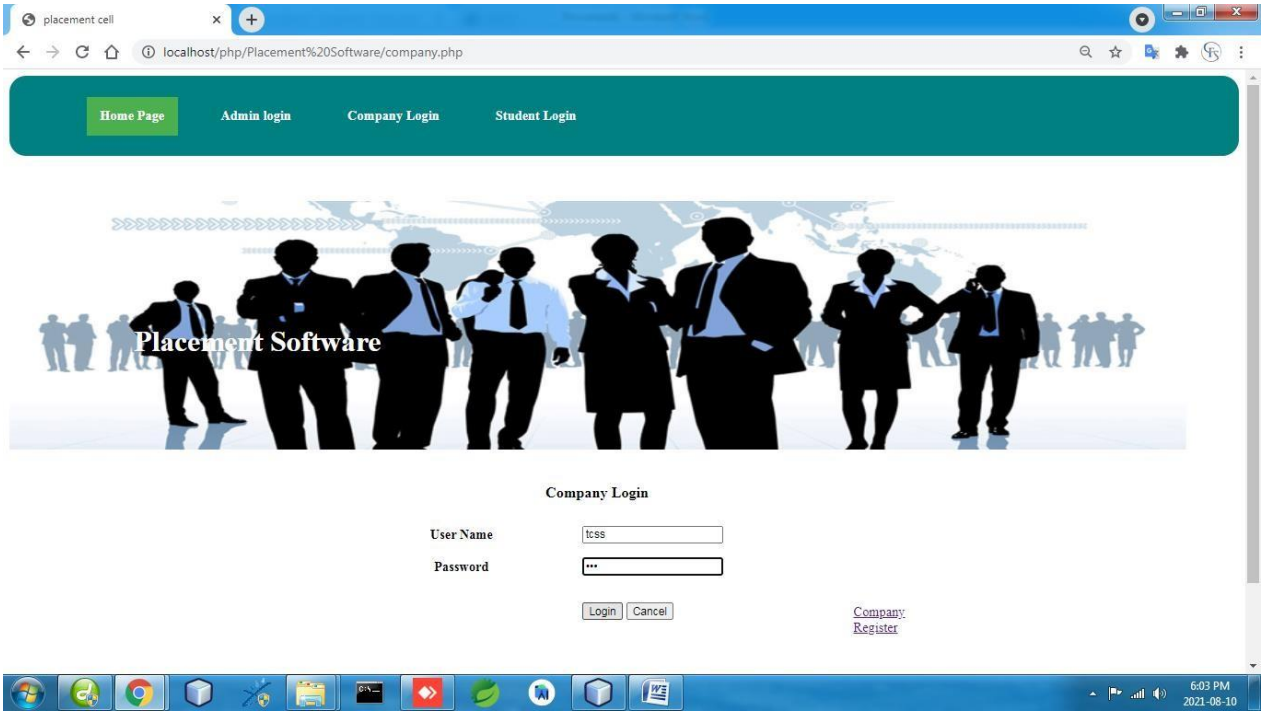


Fig:B.9 Company Login



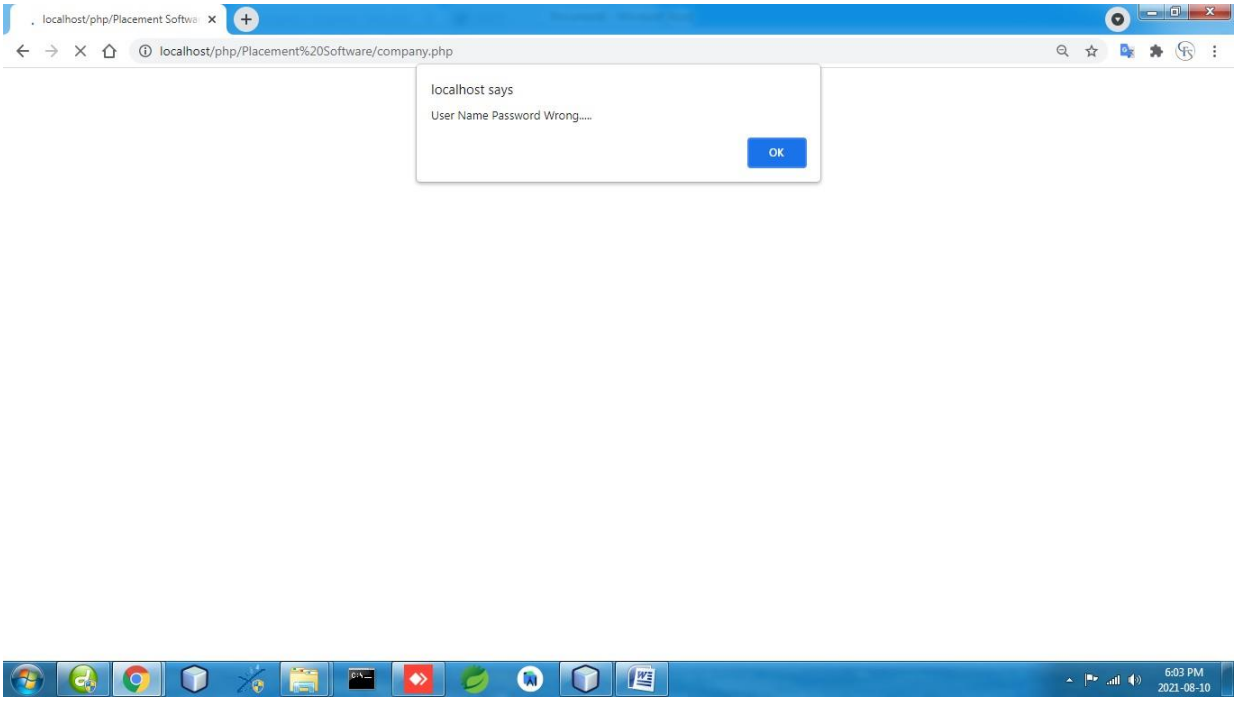


Fig:B.10 Company Job Details Registration

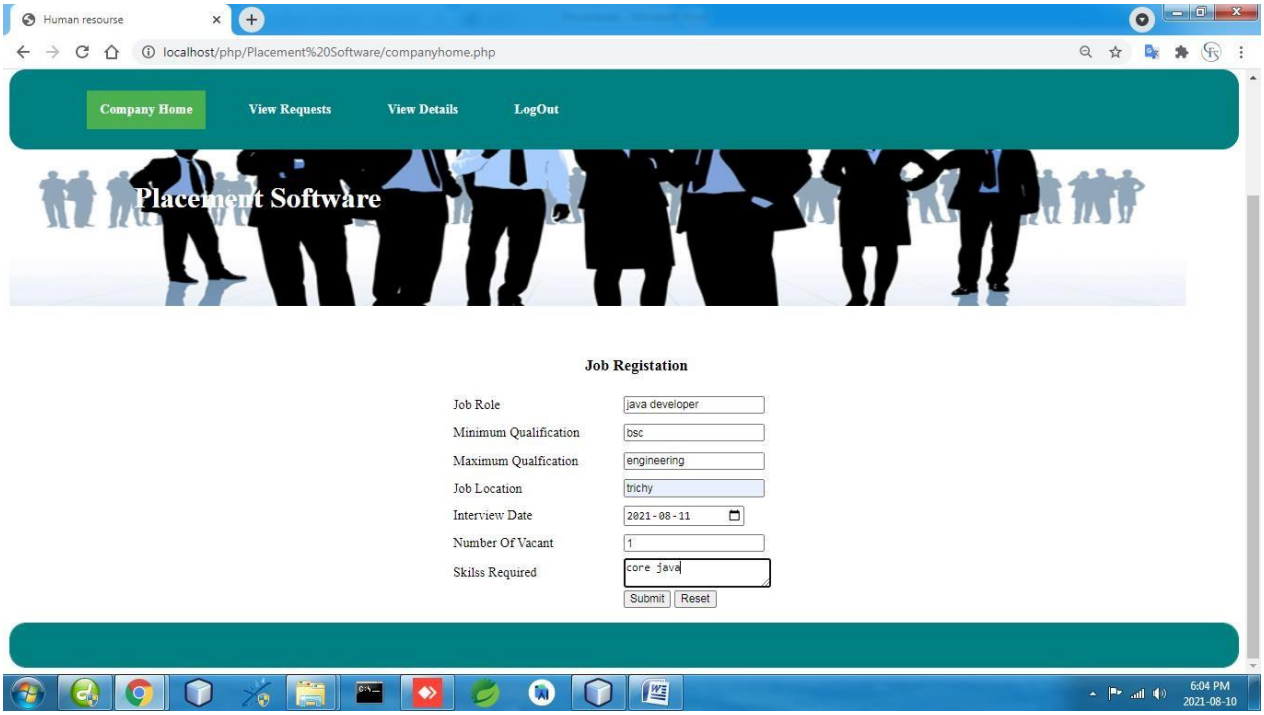


Fig:B.11 Student Job details

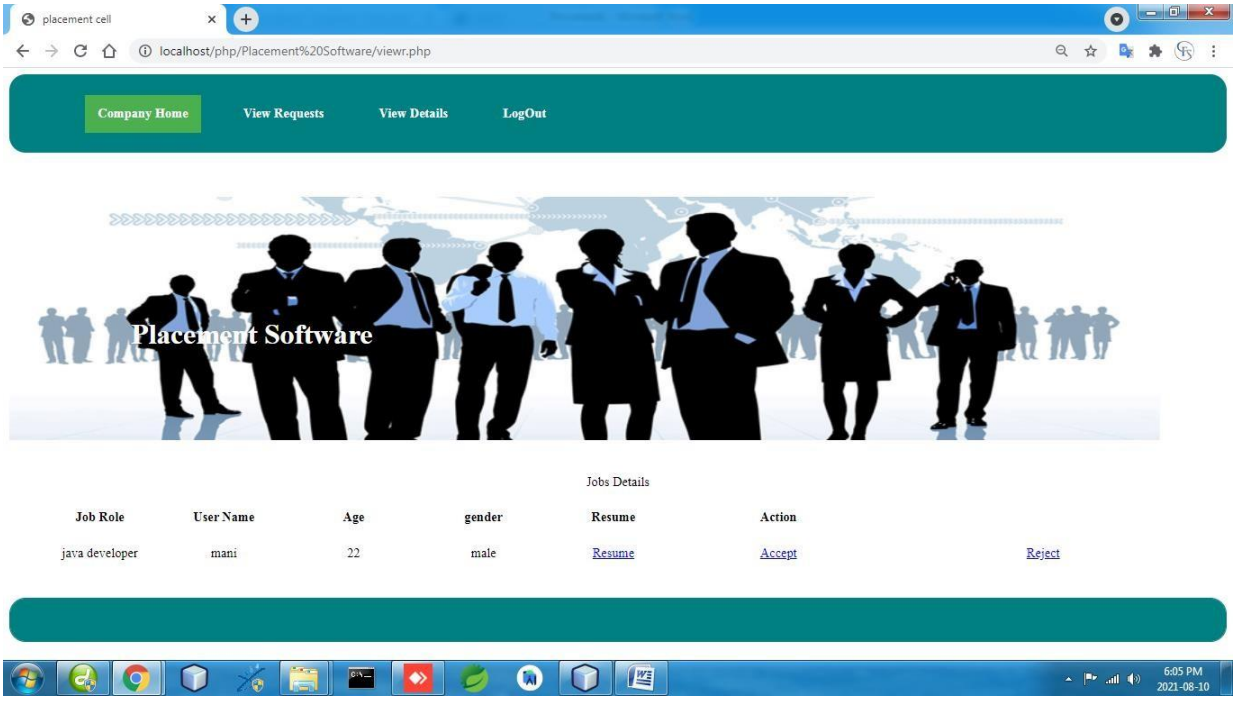


Fig:B.12 Admin job details

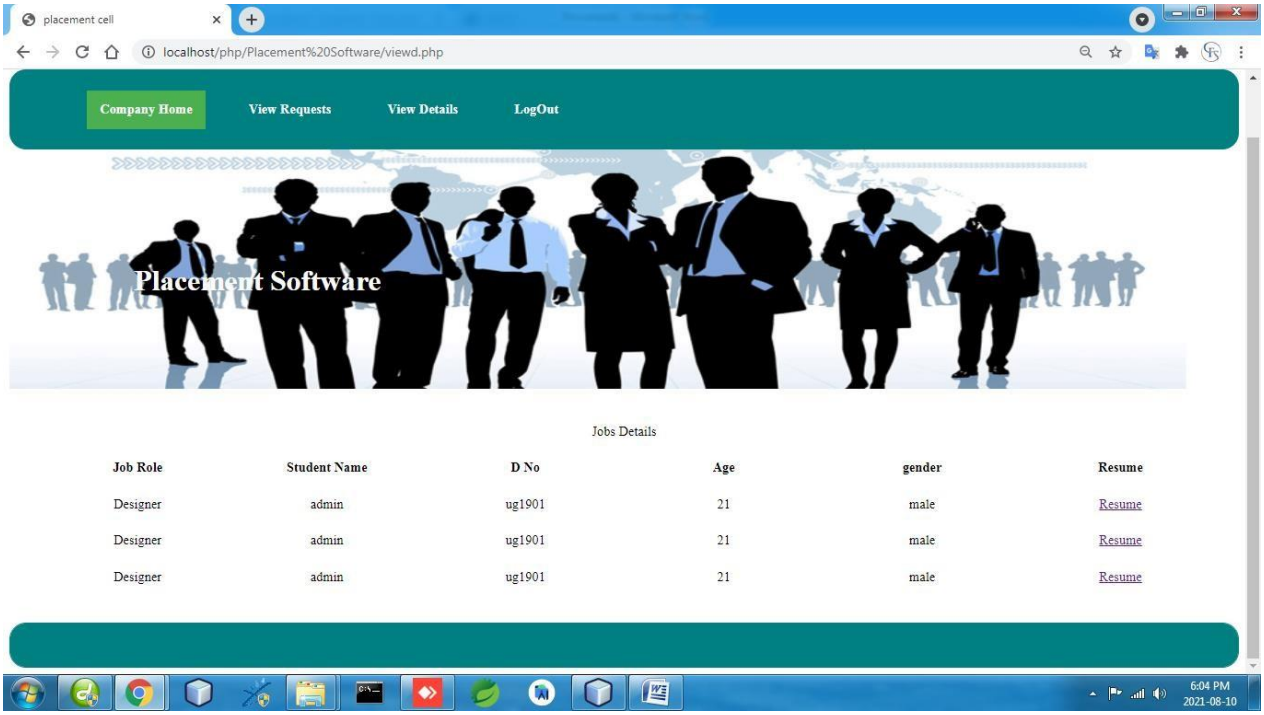


Fig: B.13 New Student Registration

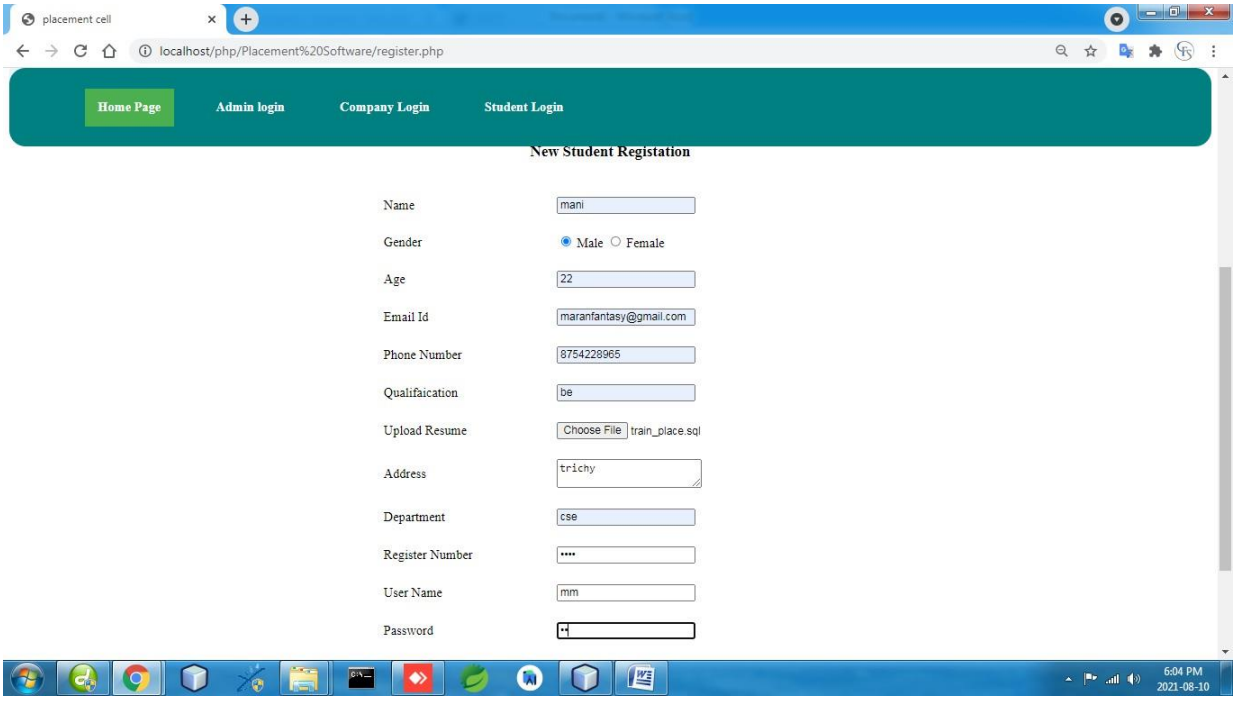


Fig: B.14 Student L ogin Page

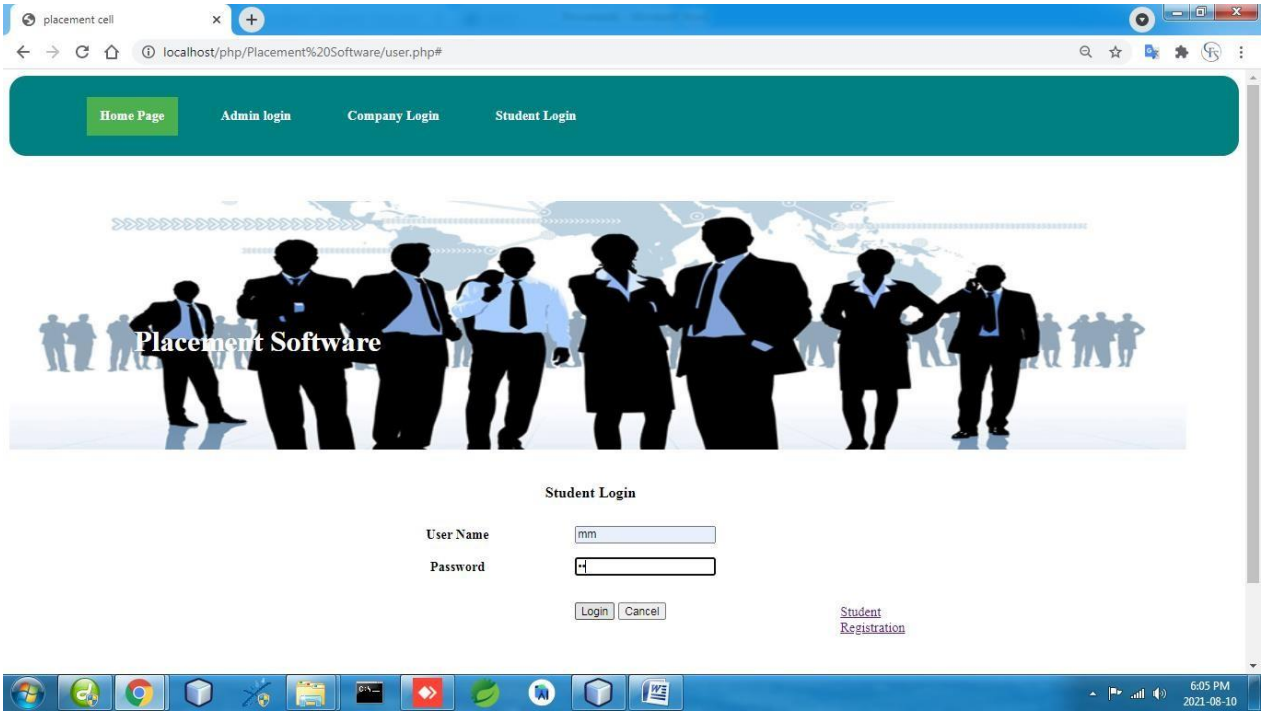


Fig: B.15 Various company Job Details

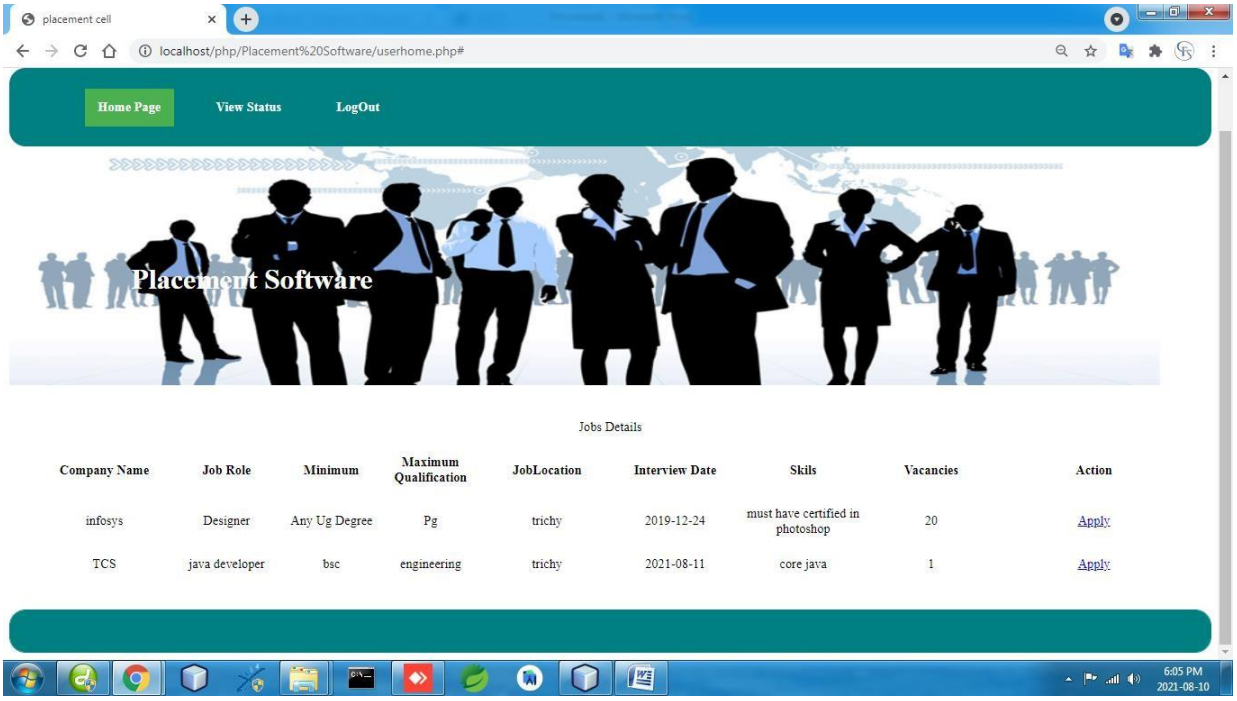
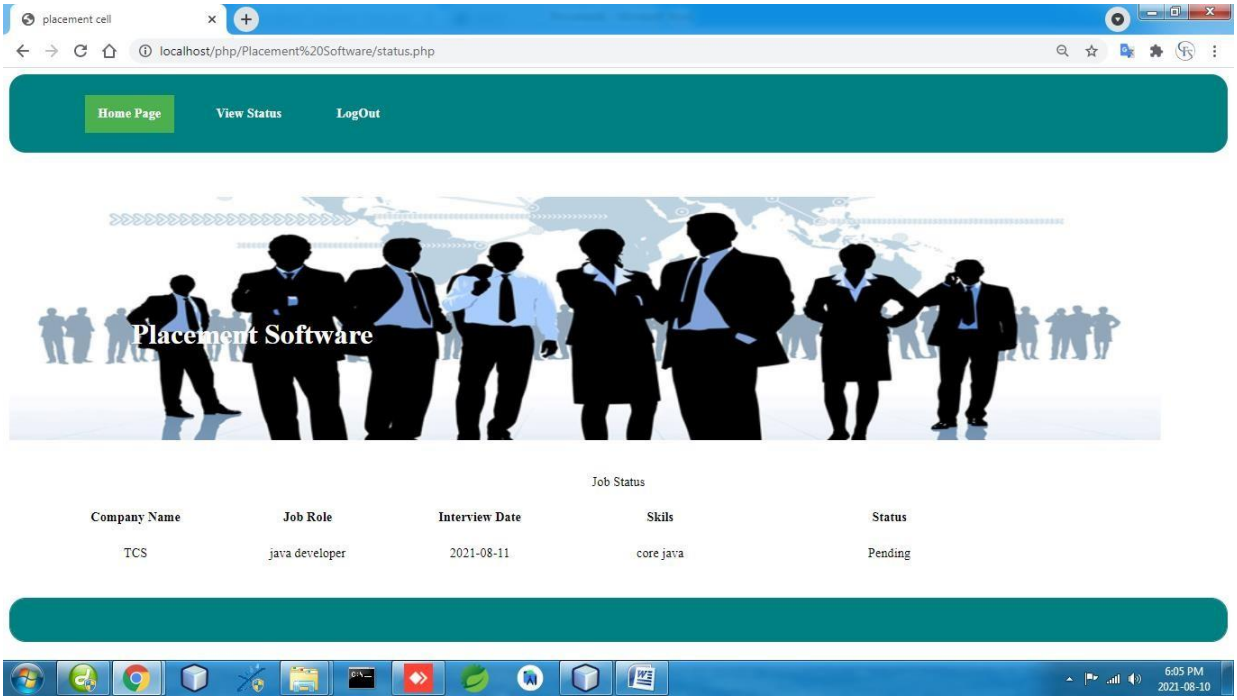


Fig :B.16 Job Status



REFERENCES

BOOK REFERENCE

- Gerken, Till, and Tobias Ratschiller. Web Application Development with PHP. New Riders Publishing, 2000.
- Greenspan, Jay, and Brad Bulger. MySQL/PHP database applications. John Wiley & Sons, Inc., 2001.
- Gutmans, Andi, StigBakken, and DerickRethans. PHP 5 Power Programming (Bruce Perens' Open Source Series). Prentice Hall PTR, 2004.
- Krasnick, Cheryl L. [Book Review] Dark Paradise: Opiate Addiction in America Before 1940. Canadian Society for the History of Medicine, 1984.
- Scollo, Chris, and SaschaShumann. Professional PHP programming. Wrox Press Ltd., 1999.
- Ware, Brent. Open source development with LAMP: using Linux, Apache, MySQL and PHP. Addison-Wesley Longman Publishing Co., Inc., 2002.
- Yank, Kevin. Build your own database driven website using PHP & MySQL. SitePoint Pty Ltd, 2004.

WEBSITE REFERENCE

1. www.freetechbooks.com

2. www.slideshare.com

3. www.w3schools.com

4. www.programmersheaven.com

5. www.phpreferencebook.com