

*IT PROJECT ON*

***Placement Management System***

*SUBMITTED TO:*

**Department of Computer Science & Engineering**

*by*

Vishal Vimal Kumar Nishant Aishwarya

Registration No:160905071 Registration No:160905262

Roll Number:21 Roll Number:33

Semester and Section:7C Semester and Section:7C

Name & Signature of Evaluator 1 Name & Signature of Evaluator 2 (Aug 2019)

## ****Acknowledgement****

The success and final outcome of this project required a lot of guidance and assistance from many people and I am extremely privileged to have got this all along the completion of my project. All that I have done is only due to such supervision and assistance and I would not forget to thank them.

I respect and thank Mr. Amit Kumar, for providing me an opportunity to do the project work and giving us all support and guidance, which made me complete the project duly. I am extremely thankful for providing such a nice support and guidance, although he had a busy schedule.

I would not forget to remember Mrs. Anita Kini, for her encouragement and more over for her timely support and guidance till the completion of our project work.

I am thankful to and fortunate enough to get constant encouragement, support and guidance from all Teaching staffs of Computer Science which helped us in successfully completing our project work. Also, I would like to extend our sincere esteems to all staff in laboratory for their timely support.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Table of Contents** | | | | |
|  | | | | Page No |
| **Chapter 1** | | | **INTRODUCTION** |  |
|  | **1.1** | Placement management system | |  |
|  | **1.2** | Applications of Placement Portal | |  |
|  | | | | |
| **Chapter 2** | | | BACKGROUND THEORY and/or LITERATURE REVIEW |  |
|  | **2.1** | Introduction to Programming Environment | |  |
|  | **2.2** | Requirement Analysis | |  |
|  | **2.3** | Functional Requirement | |  |
|  | | | | |
| **Chapter 3** | | | **METHODOLOGY** |  |
|  | **3.1** | Objective | |  |
|  | **3.2** | Approach | |  |
|  | | | | |
| **Chapter 4** | | | **RESULTS AND DISCUSSION** |  |
|  | **4.1** | Code | |  |
|  | **4.2** | Excel Sheet | |  |
|  | **4.3** | Screen Shots | |  |
|  | | | | |
| **Chapter 5** | | | **CONCLUSIONS AND FUTURE ENHANCEMENTS** |  |
|  | | | | |
| **REFERENCES** | | | |  |
| **PLAGIARISM REPORT** | | | |  |

**INTRODUCTION**

**1.1** Placement Management System

This project is aimed at developing an web application for the Training and Placement Department of the College. The system is a web application that can be accessed throughout the organization with proper login provided. This system can be used as a web application for the Training and Placement Officers (TPO) of the college to manage the student information with regard to placement. Students logging should be able to upload their information in the form of a CV. The key feature of this project is that it is a onetime registration. Our project provides the facility of maintaining the details of the students. It also provides a requested list of candidates to recruit the students based on given query. Administrator logging in may also search any information put up by the students. This project will aid colleges to practice full IT deployment. This will also help in fast access procedures in placement related activities

**1.2** Applications of placement portal

The earlier system was not computerized, transactions in the previous system were done manually by maintaining records. It takes much time for a placement officer to collect and approve the details of students. There is poor communication between students and placement officer. Students may not obtain the desired information. It is difficult to coordinate students, companies, and interviews. The proposed system is a web-based application. The system allows students to access details of recruitments. The system allows students to access any material posted by various departments. Recruiters can access the student details. It is easy for one to access desired information through the well-defined interfaces. For the purpose of training and placement of the Students. This application will perform various tasks for student administrative authority. This project helps a better student support and time-saving factor. The various types of reports will be available quickly without any effort. Training and placement provide infrastructural facility to conduct the group discussion.

This project is aimed at developing a web portal for training and placement. Campus placements are conducted in all colleges for all the education fields. Various software and other sector companies are conducting campus recruitment process for selecting candidates. When campus selections are conducted in colleges the students have to provide their curriculum vitae to the concern TPO officer for attending the campus interviews. This routine process is maintained manually. Many colleges have automated system but have some limitations. Students uploaded their CVs early in the year, leaving them as it is in time. Lists were produced for each company based on the present information. Students have to check their mail regularly. Searching is done based on the company criteria and requirement. The student will get notify rough mailing only. The students were not being made aware of the T&P activity other than placements. The information like company technical question paper, Resume Format, Job details are not available to the student. There was no record kept of the past students and their working status. There was less communication between the student with the Training and Placement department. So that this can be improved by designing Advanced software.

**BACKGROUND THEORY**

2.1 Introduction To Programming Environment

[C#](https://www.geeksforgeeks.org/csharp-programming-language/)is a general-purpose, modern and object-oriented programming language pronounced as **“C sharp”**. It was developed by Microsoft led by Anders Hejlsberg and his team within the .Net initiative and was approved by European Computer Manufacturers Association (ECMA) and International Standards Organization (ISO). C# is among the languages for [Common Language Infrastructure](https://en.wikipedia.org/wiki/Common_Language_Infrastructure) and the current version of C# is version 7.2. C# is a lot similar to Java syntactically and is easy for the users who have knowledge of C, C++ or Java.

A bit about .Net Framework:  
.Net applications are multi-platform applications and framework can be used from languages like C++, C#, Visual Basic, COBOL etc. It is designed in a manner so that other languages can use it.  
know more about [.Net Framework](https://en.wikipedia.org/wiki/.NET_Framework)

MySQL is the world's most popular open source database software, with over 100 million copies of its software downloaded or distributed throughout its history. With its superior speed, reliability, and ease of use, MySQL has become the preferred choice for Web, Web 2.0, SaaS, ISV, Telecom companies and forward-thinking corporate IT Managers because it eliminates the major problems associated with downtime, maintenance and administration for modern, online applications.

Many of the world's largest and fastest-growing organizations use MySQL to save time and money powering their high-volume Web sites, critical business systems, and packaged software — including industry leaders such as Yahoo!, Alcatel-Lucent, Google, Nokia, YouTube, Wikipedia, and Booking.com. The flagship MySQL offering is MySQL Enterprise, a comprehensive set of production-tested software, proactive monitoring tools, and premium support services available in an affordable annual subscription.

MySQL is a key part of LAMP (Linux, Apache, MySQL, PHP / Perl / Python), the fast-growing open source enterprise software stack. More and more companies are using LAMP as an alternative to expensive proprietary software stacks because of its lower cost and freedom from platform lock-in. MySQL was originally founded and developed in Sweden by two Swedes and a Finn: David Axmark, Allan Larsson and Michael "Monty" Widenius, who had worked together since the 1980's. More historical information on MySQL is available on Wikipedia.

## 2.2 Requirement Analysis

We are overcoming the difficulty of student details which were manual in the current system and here we generate detailed information about the students which will save our time to inform each and every batch and section and student profile is maintained.

2.3 Functional Requirement

This section describes the functional requirements of the system for those requirements which are expressed in the natural language style. A faculty member should be able to login to the system through the first page of the application, and mention his required roll number and he should get the details of the student with that roll number. An administrator can login into his account and he will update the student information.

**METHODOLOGY**

3.1 Objective

3.1.1 About the Present System

In Various colleges, training and placement officers have to manage the student’s profile and documents of students for their training and placement manually. Placement Officer have to collect the information of various companies who want to recruit students and notify students time to time about them. Placement Officer have to arrange profiles of students according to various streams and notify them according to company requirements. If any modifications or updates are required in the profile of any student, it has to searched and to be done it manually.

3.1.2 About the Project

This project is to facilitate students in college, company to register and communicate with Placement Office. The users can access easily to this and the data can be retrieved easily in no time. In the main page there are options for a new register, a registered student to directly login using username and password, submit resume. In the student registration form, we can give personal details, educational qualifications, and professional skills and upload resume. The job details of the placed students will be provided by the administrator. The administrator plays an important role in our project. They provide approval of student registration and updating. In this project we create a search engine for administrator, who can search everything about the student and company.

3.2 Approach

Student Profile: Create profiles on the system and update automatically

Students can create profiles on the system directly. Create, edit and maintain student profiles including demographic information, assignments, and placement schedules attended which can be viewed by recruiters. create multiple job profiles, update their knowledge and skills and chart their own career path. Enable students to sign up and create an account in the with options to register via Google, Facebook and Linkedin login details. Create student groups with options to claim to take over admin rights, add members to the group using advanced search function and create events using online calendar and invite individual users and companies for sponsorship.

• Create and manage student accounts

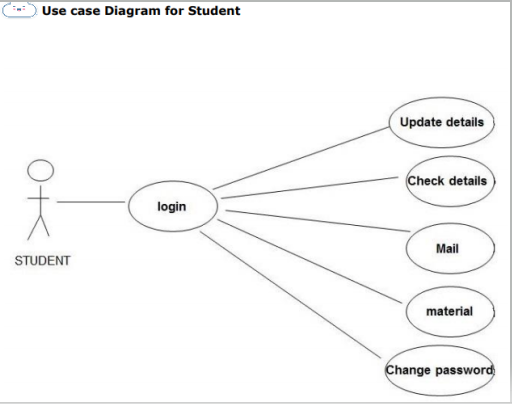
• Job profile creation

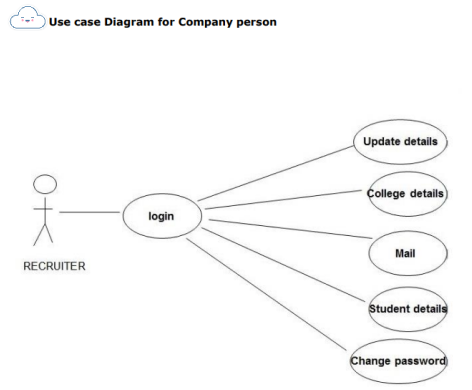
• Events and calendar sync

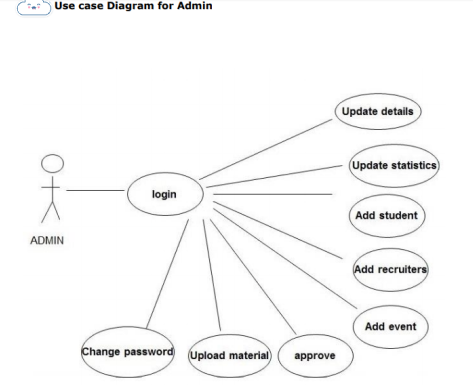
• Create multiple profiles

As a Student: Enable students to create job profiles, upload resumes, search and apply for jobs and internships with real-time status tracking from anywhere and anytime.

As a Administrator: Capture data and reports related to students and employers which can be accessed by administrators with a single click in a safe and secure environment.







**RESULTS AND DISCUSSION**

4.1 Code Execution

**Student Profile**

student\_login.aspx.cs

using System;

using System.Collections.Generic;

using System.Data.SqlClient;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

namespace placement\_management\_system

{

public partial class student\_login : System.Web.UI.Page

{

protected void Page\_Load(object sender, EventArgs e)

{

}

protected void login\_btn\_click(object sender, EventArgs e)

{

SqlConnection con = new SqlConnection();

con.ConnectionString = @"Data Source=(LocalDB)\v11.0;AttachDbFilename='C:\Users\OSLAB\Source\Repos\placement-management system\placement\_management\_system\placement\_management\_system\db\pmsdb.mdf';Integrated Security=True;Connect Timeout=30";

con.Open();

SqlCommand cmd = new SqlCommand();

cmd.Connection = con;

cmd.CommandText = "SELECT \* from student\_table where collage\_id = '" + clg\_id.Text.ToString() + "'";

SqlDataReader reader = cmd.ExecuteReader();

if(reader.Read()){

if (((String)reader["password"].ToString()).Equals((String)pass.Text.ToString()))

{

Session["student\_id"] = reader["student\_id"];

Response.Redirect("dashboard.aspx");

}

else

{

alert.InnerHtml = "<script>alert('Invalid Password');</script>";

}

}

else

{

alert.InnerHtml = "<script>alert('Invalid Username');</script>";

}

con.Close();

}

}

}

select\_company.aspx.cs

using System;

using System.Collections.Generic;

using System.Data.SqlClient;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

namespace placement\_management\_system.student

{

public partial class select\_company : System.Web.UI.Page

{

protected void Page\_PreRender(object sender, EventArgs e)

{

if (Session["student\_id"] == null)

{

Response.Redirect("student\_login.aspx", true);

}

SqlConnection con = new SqlConnection();

con.ConnectionString = @"Data Source=(LocalDB)\v11.0;AttachDbFilename='C:\Users\OSLAB\Source\Repos\placement-management-system\placement\_management\_system\placement\_management\_system\db\pmsdb.mdf';Integrated Security=True;MultipleActiveResultSets=True;Connect Timeout=30";

con.Open();

SqlCommand c = new SqlCommand();

c.Connection = con;

c.CommandText = "SELECT \* from student\_table where student\_id = '" + Session["student\_id"].ToString() + "'";

SqlDataReader r = c.ExecuteReader();

r.Read();

name.Text = r["full\_name"].ToString();

String selected = null;

String company\_name = null;

String selary = null;

String post = null;

String location = null;

String id = null;

String sel = null;

String text = null;

SqlCommand cmd = new SqlCommand();

cmd.Connection = con;

cmd.CommandText = "SELECT \* from company\_table";

SqlDataReader reader = cmd.ExecuteReader();

LiteralControl imageGallery = new LiteralControl();

while (reader.Read())

{

if (float.Parse(r["cpi"].ToString()) >= float.Parse(reader["required\_min\_cpi"].ToString()) && reader["required\_branches"].ToString().Contains(r["branch"].ToString()))

{

SqlCommand ccmd = new SqlCommand();

ccmd.Connection = con;

ccmd.CommandText = "SELECT \* from company\_choice where company\_id='" + reader["company\_id"].ToString() + "' and student\_id='" + Session["student\_id"].ToString() + "'";

SqlDataReader rdr = ccmd.ExecuteReader();

sel = "false";

selected = "";

text = "select";

if (rdr.Read())

{

text = "remove";

selected = "selected";

sel = "true";

}

company\_name = reader["company\_name"].ToString();

selary = reader["tentative\_salary"].ToString();

post = reader["job\_post"].ToString();

location = reader["job\_location"].ToString();

id = reader["company\_id"].ToString();

String str = String.Format(@"<li id='companies' runat='server'>

<div class='collapsible-header'><span class='new badge' data-badge-caption='{0}'></span><b>{1}</b></div><div class='collapsible-body'><table class='striped'><tr><td style='padding-left:6%;'>Expected Job Post</td><td>{2}</td></tr><tr><td style='padding-left:6%;'>Expected Annual Job Selary</td><td>{3} &#8377;</td></tr><tr><td style='padding-left:6%;'>Expected Job Location</td><td>{4}</td></tr><tr><td style='padding-left:6%;'><a class='waves-effect waves-light btn' href='view\_company\_details.aspx?id={5}&src=select'>view more details</a></td><td><a class='waves-effect waves-light btn' href='\_select\_company.aspx?id={6}&select={7}'>{8}<a/></td></tr></table></div></li>", selected, company\_name, post, selary, location, id, id, sel, text);

imageGallery.Text += str;

}

this.companies.Controls.Add(imageGallery);

}

con.Close();

}

}

}

**Admin Profile**

view\_company.aspx.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

namespace placement\_management\_system.admin

{

public partial class view\_company : System.Web.UI.Page

{

protected void Page\_Load(object sender, EventArgs e)

{

if (Session["admin\_id"] == null)

{

Response.Redirect("admin\_login.aspx", true);

}

}

}

}

view\_student.aspx.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

namespace placement\_management\_system.admin

{

public partial class view\_student : System.Web.UI.Page

{

protected void Page\_Load(object sender, EventArgs e)

{

if (Session["admin\_id"] == null)

{

Response.Redirect("admin\_login.aspx", true);

}

}

}

}

**Company Profile**

view\_students.aspx.cs

using System;

using System.Collections.Generic;

using System.Data.SqlClient;

using System.IO;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

namespace placement\_management\_system.company

{

public partial class view\_students : System.Web.UI.Page

{

protected void Page\_Load(object sender, EventArgs e)

{

if (Session["company\_id"] == null)

{

Response.Redirect("company\_login.aspx", true);

}

SqlConnection con = new SqlConnection();

con.ConnectionString = @"Data Source=(LocalDB)\v11.0;AttachDbFilename='C:\Users\OSLAB\Source\Repos\placement-management-system\placement\_management\_system\placement\_management\_system\db\pmsdb.mdf';Integrated Security=True;Connect Timeout=30";

con.Open();

SqlCommand cmd = new SqlCommand();

cmd.Connection = con;

cmd.CommandText = "SELECT \* from company\_table where company\_id = '" + Session["company\_id"].ToString() + "'";

SqlDataReader reader = cmd.ExecuteReader();

reader.Read();

name.Text = reader["company\_name"].ToString();

con.Close();

}

protected void DownloadFile(object sender, EventArgs e)

{

try

{

SqlConnection con = new SqlConnection();

con.ConnectionString = @"Data Source=(LocalDB)\v11.0;AttachDbFilename='C:\Users\OSLAB\Source\Repos\placement-management-system\placement\_management\_system\placement\_management\_system\db\pmsdb.mdf';Integrated Security=True;Connect Timeout=30";

con.Open();

SqlCommand cmd = new SqlCommand();

cmd.Connection = con;

cmd.CommandText = "select refrance\_documents from student\_table where student\_id='" + (sender as LinkButton).CommandArgument + "'";

SqlDataReader obj = cmd.ExecuteReader();

byte[] bytes = (byte[])obj.GetValue(0);

con.Close();

Response.Clear();

Response.ContentType = "application/force-download";

Response.AddHeader("content-disposition", "attachment; filename=ref\_doc.zip");

Response.BinaryWrite(bytes);

Response.End();

}

catch (Exception eeee)

{

MemoryStream ms = new MemoryStream();

TextWriter tw = new StreamWriter(ms);

tw.WriteLine("Student has not uploaded refrance documents");

tw.Flush();

byte[] bytes = ms.ToArray();

ms.Close();

Response.Clear();

Response.ContentType = "application/force-download";

Response.AddHeader("content-disposition", "attachment; filename=file.txt");

Response.BinaryWrite(bytes);

Response.End();

}

}

}

}

set\_schedule.aspx.cs

using System;

using System.Collections.Generic;

using System.Data.SqlClient;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

namespace placement\_management\_system.company

{

public partial class set\_schedule : System.Web.UI.Page

{

protected void Page\_Load(object sender, EventArgs e)

{

if (Session["company\_id"] == null)

{

Response.Redirect("company\_login.aspx", true);

}

SqlConnection con = new SqlConnection();

con.ConnectionString = @"Data Source=(LocalDB)\v11.0;AttachDbFilename='C:\Users\OSLAB\Source\Repos\placement-management-system\placement\_management\_system\placement\_management\_system\db\pmsdb.mdf';Integrated Security=True;Connect Timeout=30";

con.Open();

SqlCommand cmd = new SqlCommand();

cmd.Connection = con;

cmd.CommandText = "SELECT \* from company\_table where company\_id = '" + Session["company\_id"].ToString() + "'";

SqlDataReader reader = cmd.ExecuteReader();

reader.Read();

name.Text = reader["company\_name"].ToString();

con.Close();

}

protected void submit\_Click(object sender, EventArgs e)

{

SqlConnection con = new SqlConnection();

con.ConnectionString = @"Data Source=(LocalDB)\v11.0;AttachDbFilename='C:\Users\OSLAB\Source\Repos\placement-management-system\placement\_management\_system\placement\_management\_system\db\pmsdb.mdf';Integrated Security=True;MultipleActiveResultSets=True;Connect Timeout=30";

con.Open();

SqlCommand ccmd = new SqlCommand();

ccmd.Connection = con;

ccmd.CommandText = "update company\_table set tentative\_interview\_date = '" + interview\_date.Text + "', tentative\_written\_test\_date = ''" + test\_date .Text+ "' where company\_id='" + Session["company\_id"].ToString() + "'";

ccmd.ExecuteNonQuery();

con.Close();

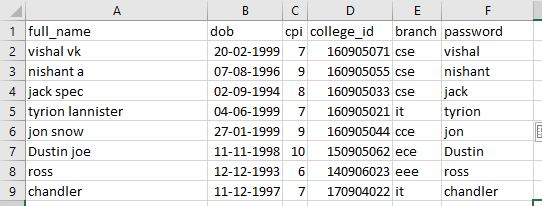
}

}

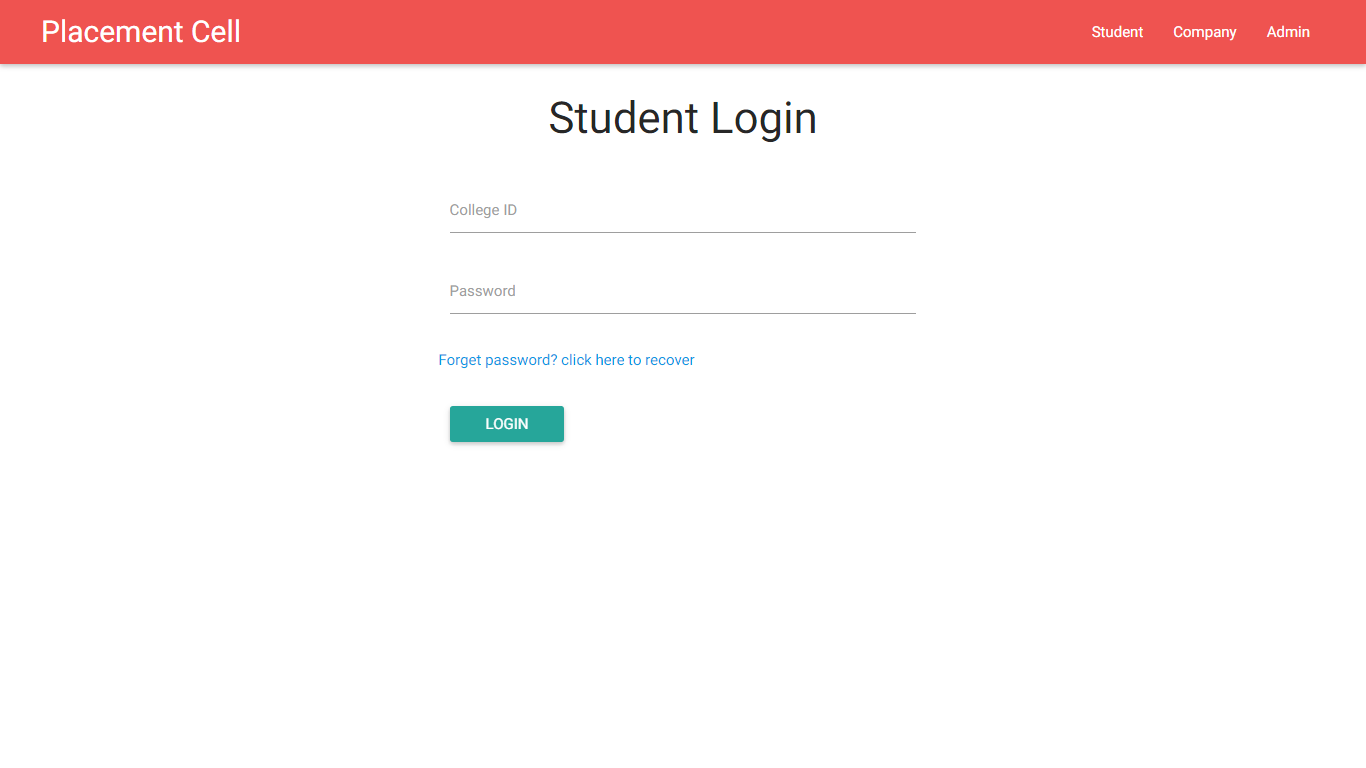
}

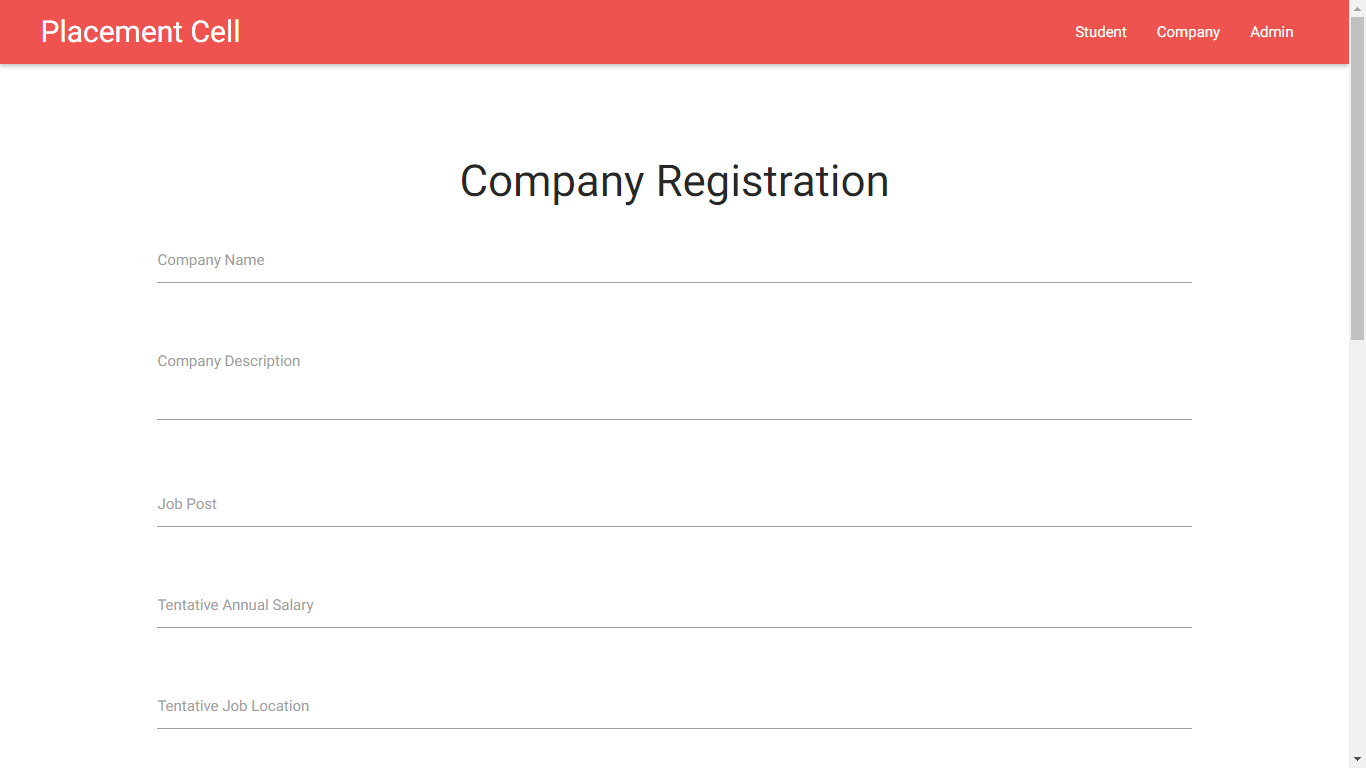
4.2 Excel Sheet

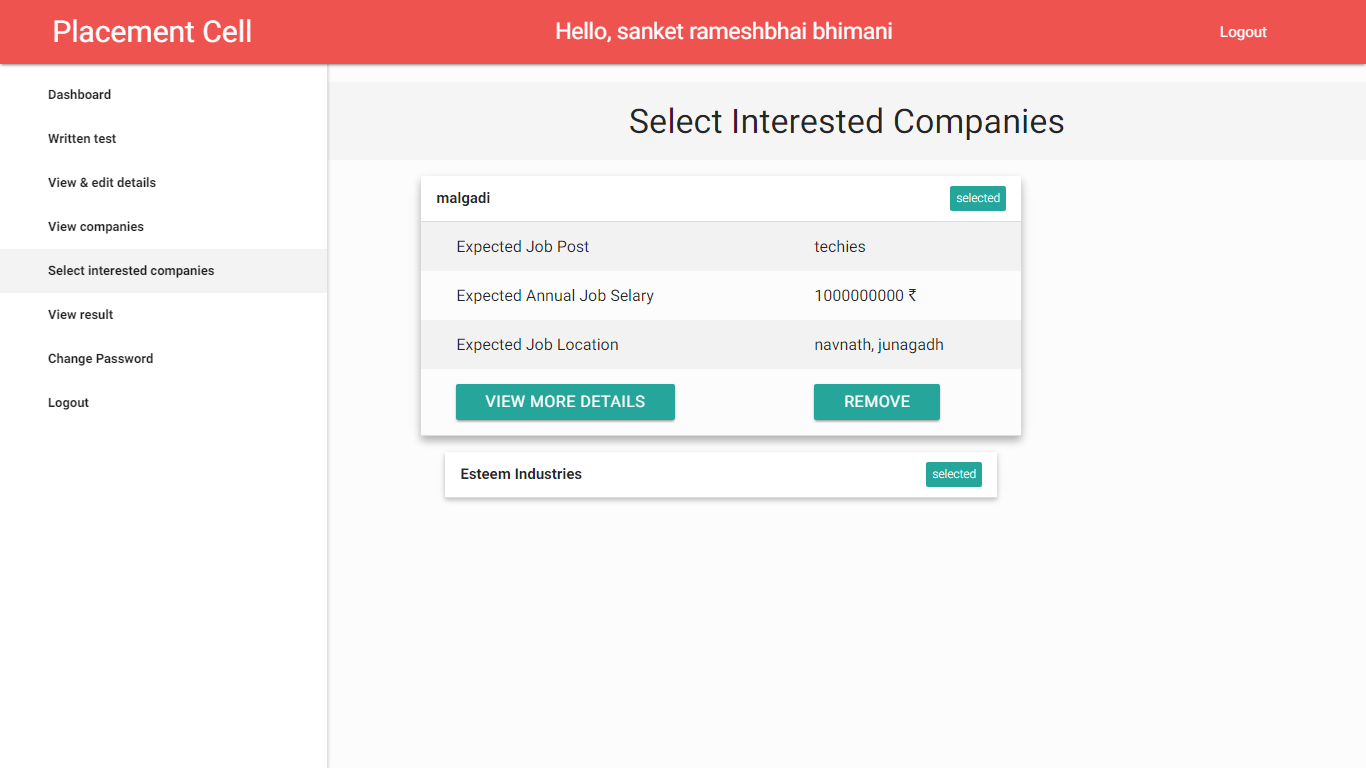
student\_data.xls



4.3 Screenshot’s of output







**CONCLUSIONS AND FUTURE ENHANCEMENTS**

From a proper analysis of positive points and constraints on the component, it can be safely concluded that the product is a highly efficient GUI based component. This component can be easily plugged in many other systems. Also, the component is user friendly. Generally, the TPO ‘s of the Colleges has to face a lot of problems in management of the Students information. This all information has to be managed manually. So, there is a need to develop a system that can solve the mentioned problem. This software comes with just that solution.

Future Enhancement

Future scope of the project: Though our project is itself matured enough but still betterment is always an open door. In this case also we can add some features to this software to make this software more reliable. These are as follows: -

-Firstly, during the development of the project my prime object was to keep the hardware & software requirement as minimum as possible so that it supports maximum user base.

- Secondly, the searching procedure should be very strong like placement officer can search student as fast as possible.

- Thirdly, modify the project with better approach with more graphics.

- Fourthly, the back-up procedure can be incorporated to make sure of the database integrity.

- Fifthly, recruiter can visit any time through this application and communicate with Placement officer.

- Sixthly, Placement officer can contact with both student and company through message. Student and company can also send message to Placement officer.

**REFERENCES**

1. <https://www.slideshare.net/mehul53/placement-management-system>
2. <http://ijcsit.com/docs/Volume%207/vol7issue2/ijcsit2016070268.pdf>
3. Web Based Placement Management System Anjali.V #1 , Jeyalakshmi.PR\*2, Anbubala.R#3,Sri Mathura devi.G#4,Ranjini.V#5 # Computer Science Department, Saranathan College of Engineering, Venkateswara Nagar, Panjappur, Trichy-12.
4. <https://www.iitms.co.in/higher-education-erp/training-and-placement/>
5. A Web Portal for Training and Placement Portal using Wireless Networks Sheetal Kanthale JSPM'S Rajarshi Shahu College of engg.
6. <http://academicscience.co.in/admin/resources/project/paper/f201804161523881757.pdf>
7. <https://www.c-sharpcorner.com/uploadfile/mahesh/dataadapter-properties-and-methods-in-ado-net/>
8. <https://docs.microsoft.com/en-us/dotnet/api/system.data.sqlclient.sqldataadapter?view=netframework-4.8>
9. <https://docs.microsoft.com/en-us/dotnet/api/system.web.ui.webcontrols.gridview?view=netframework-4.8>

10)<https://www.tutorialspoint.com/android/android_grid_view.htm>