A PROJECT REPORT ON

PLACEMENT MANAGEMENT PORTAL USING DJANGO WEB FRAMEWORK

Major project submitted in partial fulfillment of the requirements for the award of the degree of

BACHELOR OF TECHNOLOGY
IN
INFORMATION TECHNOLOGY
(2018 - 2022)

BY

M. Surya Prakash	**18241A1238**
K. Manush	**18241A1226**
D. Rishikesh	**18241A1252**
S. Sridhar	**18241A1257**

Under the Esteemed guidance of

Ms. L. Sukanya

Assistant Prof, Dept of IT

DEPARTMENT OF INFORMATION TECHNOLOGY
GOKARAJU RANGARAJU INSTITUTE OF ENGINEERING AND TECHNOLOGY
(AUTONOMOUS)
HYDERABAD

CERTIFICATE

This is to certify that it is a bonafide record of Major Project work entitled **"PREDICTING AIR QUALITY INDEX USING MACHINE LEARNING"** done by **M.Surya (18241A1238), K. Manush (18241A1226), D.Rishikesh (18241A1252), S. Sridhar (18241A1257),** students of **B. Tech (IT)** in the Department of Information Technology, Gokaraju Rangaraju Institute of Engineering and Technology during the period 2018-2022 in the partial fulfillment of the requirements for the award of degree of **BACHELOR OF TECHNOLOGY IN INFORMATION TECHNOLOGY** from GRIET, Hyderabad.

Ms. L. Sukanya ,
Assistant prof
(Internal Project Guide)

Dr.N.Ganapathi Raju ,
(Head of the Department)

(Project External)

ACKNOWLEDGEMENT

We take the immense pleasure in expressing gratitude to our Internal guide **Ms.L. Sukanya , Assistant Prof,** Information Technology, GRIET. We express our sincere thanks for her encouragement, suggestions and support, which provided the impetus and paved the way for the successful completion of the project work.

We wish to express our gratitude to **Dr.N.Ganapathi Raju,** our Project Co-coordinator **Mr.G.Vijendar Reddy and K.Archana**, for their constant support during the project.

We express our sincere thanks to **Dr. Jandhyala N Murthy,** Director, GRIET**,** and **Dr. J. Praveen,** Principal, GRIET**,** for providing us the conductive environment for carrying through our academic schedules and project with ease.

We also take this opportunity to convey our sincere thanks to the teaching and non-teaching staff of GRIET College, Hyderabad.

Email: suryaprakashmone98@gmail.com
Contact No: 9666619257
Address: SouthTeachersColony, Kavali.

Email: manushkumar007@gmail.com
Contact No: 7659092420
Address: Nandipet, Nizamabad.

Email: deshettyrishironaldo@gmail.com
Contact No: 9100510535
Address: KPHB Colony, Hyderabad.

Email: sripathisridhar04@gmail.com
Contact No: 9505457914
Address: Vidyanagar, Karimnagar

DECLARATION

This is to certify that the project entitled "**Placement Management Portal Using DJANGO framework"** is a bonafide work done by us in partial fulfillment of the requirements for the award of the degree **BACHELOR OF TECHNOLOGY IN INFORMATION TECHNOLOGY** from Gokaraju Rangaraju Institute of Engineering and Technology, Hyderabad.

We also declare that this project is a result of our own effort and has not been copied or imitated from any source. Citations from any websites, books and paper publications are mentioned in the Bibliography.

This work was not submitted earlier at any other University or Institute for the award of any degree.

M. Surya Prakash	**18241A1238**
K. Manush	**18241A1226**
D. Rishikesh	**18241A1252**
S. Sridhar	**18241A1257**

TABLE OF CONTENTS

11. LIST OF FIGURES

ABSTRACT

The training and placement officer has to inform around thousands of students about every single training and placement related activity. So to make the placement process easy and effective for the training and placement department as well as the students, a web application can be developed. This application can help the placement officers to provide the details of upcoming companies by logging in to their respective profiles. The students can also view the companies in their profiles and also post their status if applied or not for the jobs by clicking on the link provided.

This application will also enable the department incharge to monitor if a student has applied for a specific job profile or not.This process reduces the time for an industry to pick the candidates according to their need

1. INTRODUCTION

1.1 Introduction to Project

A college Campus Recruitment System that consists of a student login, company login and an admin login. The project is beneficial for college students, various companies visiting the campus for recruitment and even the college placement officer. The software system allows the students to create their profiles and upload all their details including their marks onto the system. The admin can check each student details and can remove faulty accounts. The system also consists of a company login where various companies visiting the college can view a list of students in that college and also their respective resumes. The software system allows students to view a list of companies who have posted for vacancy. The admin has overall rights over the system and can moderate and delete any details not pertaining to college placement rules. The system handles student as well as company data and efficiently displays all this data to respective sides.

1.2 OBJECTIVES

- The major objective of campus placement is to identify the talented and qualified professionals before they complete their education.
- It provides employment opportunities to students who are pursuing or in the final stage of completing the course.
- It is a cumbersome activity and hence majority of the companies

find it difficult to trace the right talent. Many students do not understand the importance of placement training that is being imparted, whether it is an aptitude training or soft skills. They show the least interest in this due to various factors viz., projects, assignments or more of activities loaded by the colleges as part of their curriculum thinking that it is not useful. It is the responsibility of the companies training on placement to make the students equipped on all aspects of career development along with creating a very good impact in them which makes them feel every minute they spend in the placement training session is worth being there and will help them in getting placed in their dream companies.

1.3 Existing System:

Existing system does all process manually. Placement officers register the information of students, they post information about companies to each department incharge. If any modifications or updates are required in the profile of any student, it has to be done manually. Also each department incharge has to recheck with students if they applied or not.

DRAWBACKS IN EXISTING SYSTEM :

1. All the records are maintained manually.
2. The department or management carry out this job manually making

it complicated and tedious most of the time. This process is so difficult when number of user's increases.

1.4 Proposed System

The proposed system can overcome all the limitation of the existing system, such as student's information is maintained in the database, it gives more security to data, ensures data accuracy, reduces paper work and save time, the placement officer has to just post link and information about the company and the student can easily login and apply by clicking on the link. Also, through this system the department incharge will be able to easily monitor, if a student applied or not for a company. This makes information flow efficient and paves way for easy report generation and reduces the space. Proposed system is cost effective.

Advantages of Proposed System:

1. High-quality placements bring good benefits and positive impacts on students as well as for the colleges.

1.5 SCOPE

- This app can be used in all educational institutions.
- Can be used anywhere any time as it is a web-based application

(user location doesn't matter).

- No restriction that you need to be logged into the web page always.
- No manual work of managing placement related details.
- Less time consumption as many functions take place in just one click.
- For TPO and department incharge , it saves time in accessing the application for a job detail of candidates.

2. REQUIREMENT

The hardware and software component of a computer system that are required to install and use software efficiently. The software manufacturer will list the system requirements on the software package. If your computer system does not meet the system requirements then the software may not work correctly after installation System requirements for operating systems will be hardware components, while other application software will list both hardware and operating system requirements. System requirements are most commonly seen listed as minimum and recommended requirements. The minimum system requirements need to be met for the software to run at all on your system, and the recommended system requirements, if met, will offer better software usability.

2.1 Software Requirements

OS : Windows 7,8,10 Ultimate,Linux,Mac

Front-End

:HTML,CSS,Bootstrap,JavaScript,Jquery

Backend : Python.

Framework : Django.

Database : MySql

2.2 Hardware Requirements

System	:	Intel I-3, 5, 7 Processor.
Hard Disk	:	500 GB.
Floppy Drive	:	1.44 Mb.
Monitor	:	14’ Colour Monitor.
Mouse	:	Optical Mouse.
Ram	:	2Gb(Minimum).

FUNCTIONAL REQUIREMENTS

Outputs from computer systems are required primarily to communicate the results of processing to users. They are also used to provide a permanent copy of the results for later consultation.

The various types of outputs in general are:

External Outputs, whose destination is outside the organization.

- Internal Outputs whose destination is within organization and they are the user's main interface with the computer.
- Operational outputs whose use is purely within the computer department.
- Interface outputs, which involve the user in communicating directly.
- Understanding user's preferences, expertise level and his business requirements through a friendly questionnaire.
- Input data can be in four different forms - Relational DB, text files, .xls and xml files. For testing and demo you can choose data from any domain. User-B can provide business data as input.

NON-FUNCTIONAL REQUIREMENTS

The major non-functional Requirements of the system are as follows

1. **Usability-** This application is easy to learn and understand. It is user-friendly and visually appealing.
2. **Performance-** This application supports multiple users at a

time. This will also allow any new updates, as the application is dynamic.

3. **Security Requirements-**The application assigns specific functions for different users. It also provides privacy for the users

3. LITERATURE SURVEY

Each and every process in the existing system is carried out manually. The college training and placement officer had to refer all the records of previous years for even minor details. This used to be tedious and more time consuming than it sounds. It becomes more difficult when the number of students increase each passing year. There are other limitations of existing system. In manual placement management system all the task is done by human interventions. Therefore there is maximum chance of errors. The files are not stored in hierarchical form. Thus searching for a particular becomes complex task. Updating certain information is difficult and ambiguous which may lead to data redundancy due to the chances of duplication of information. Not every students are aware of the placement updates by training and placement officer of the college, therefore they may lose an opportunity to grab a seat for job interview.

OBJECTIVE

1. To provide recruitment to students:

Student recruitment should be recognized as a key component in the sustainability and success of an institution and must be a prime responsibility of an institution's strategic plan. Equipping the Training and placement office to recruit the qualified students and the right number of students is critical to the long-term sustainability of an institution, and IT plays an important role.

2. Planning and organizing various Placement drives in campus:

Students should use the fair in order to grab the opportunity understand what companies are looking for in candidates and what skills are vital for their future roles. Career/Job Fairs are a valuable opportunity to hone

networking skills and gain practice in articulating their value to a prospective employer.

This project consists of '3' important modules for which all the actions are listed below

TPO:

1. Can view the profiles.
2. Can create and delete the profile.
3. Can change password.
4. Can add company details.

Department incharge:

1. Can view the profiles.
2. Can create and delete the profile.
3. Can change password.
4. They get the details of the candidates registered and unregistered for a specific company.
5. Can view the companies provided by the TPO.

Candidate:

1. Can view the profiles.
2. Can create and delete the profiles.
3. Can change password.
4. Can view the company details provided by the TPO.
5. Can register for vacancy in a company through the link provided by the

TPO.

4. TECHNOLOGY

FRONT END: 4.1 HTML

Hypertext Markup Language (HTML) is the standard markup language for documents designed to be displayed in a web browser. It can be assisted by technologies such as Cascading Style Sheets (CSS) and scripting languages such as JavaScript. Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.

HTML elements are the building blocks of HTML pages. With HTML constructs, images and other objects such as interactive forms may be embedded into the rendered page. HTML provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. HTML elements are delineated by tags, written using angle brackets. Tags such as <img /> and <input /> directly introduce content into the page. Other tags such as <p> surround and provide information about document text and may include other tags as sub-elements. Browsers do not display the HTML tags, but use them to interpret the content of the page.

HTML can embed programs written in a scripting language such as JavaScript, which affects the behaviour and content of web pages. Inclusion of CSS defines the look and layout of content. The World Wide Web Consortium (W3C), former maintainer of the HTML and current maintainer of the CSS standards, has encouraged the use of CSS over explicit presentational HTML since 1997.

4.2 JAVASCRIPT

JavaScript s a high-level, interpreted scripting language that conforms to the ECMAScript specification. JavaScript has curly-bracket syntax, dynamic typing, prototype-based object- orientation, and first-class functions. Alongside HTML and CSS, JavaScript is one of the core technologies of the World Wide Web. JavaScript enables interactive web pages and is an essential part of web applications. The vast majority of websites use it and major web browsers have a dedicated JavaScript engine to execute it. As a multi-paradigm language, JavaScript supports event-driven, functional, and imperative (including object-oriented and prototype-based) programming styles. It has APIs for working with text, arrays, dates, regular expressions, and the DOM, but the language itself does not include any I/O, such as networking, storage, or graphics facilities. It relies upon the host environment in which it is embedded to provide these features.

Initially only implemented client-side in web browsers, JavaScript engines are now embedded in many other types of host software, including server-side in web servers and databases, and in non-web programs such as word processors and PDF software, and in runtime environments that make JavaScript available for writing mobile and desktop applications, including desktop widgets.

The terms Vanilla JavaScript and Vanilla JS refer to JavaScript not extended by any frameworks or additional libraries. Scripts written in Vanilla JS are plain JavaScript code. Google's Chrome extensions, Opera's extensions, Apple's Safari 5 extensions, Apple's Dashboard Widgets, Microsoft's Gadgets, Yahoo! Widgets, Google Desktop Gadgets, and Serence Klipfolio are implemented using JavaScript.

4.3 JQUERY:

jQuery is a JavaScript library designed to simplify HTML DOM tree traversal and manipulation, as well as event handling, CSS animation, and Ajax. It is free, open-source software using the permissive MIT License. As of May 2019, jQuery is used by 73% of the 10 million most popular websites. Web analysis indicates that it is the most widely deployed JavaScript library by a large margin, having 3 to 4 times more usage than any other JavaScript library. jQuery's syntax is designed to make it easier to navigate a document, select DOM elements, create animations, handle events, and develop Ajax applications. jQuery also provides capabilities for developers to create plug-ins on top of the JavaScript library. This enables developers to create abstractions for low-level interaction and animation, advanced effects and high-level, themeable widgets. The modular approach to the jQuery library allows the creation of powerful dynamic web pages and Web applications.

The set of jQuery core features—DOM element selections, traversal and manipulation—enabled by its selector engine (named "Sizzle" from v1.3), created a new "programming style", fusing algorithms and DOM data structures. This style influenced the architecture of other JavaScript frameworks like YUI v3 and Dojo, later stimulating the creation of the standard Selectors API. Microsoft and Nokia bundle jQuery on their platforms.

4.4 CSS:

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language like HTML.CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript.CSS is designed to enable the separation of presentation and content, including layout, colors, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple web pages to share formatting by specifying the relevant CSS in a separate .css file, and reduce complexity and repetition in the structural content.

CSS information can be provided from various sources. These sources can be the web browser, the user and the author. The information from the author can be further classified into inline, media type, importance, selector specificity, rule order, inheritance and property definition. CSS style information can be in a separate document or it can be embedded into an HTML document. Different styles can be applied depending on the output device being used; for example, the screen version can be quite different from the printed version, so that authors can tailor the presentation appropriately for each medium. The style sheet with the highest priority controls the content display. Declarations not set in the highest priority source are passed on to a source of lower priority, such as the user agent style. The process is called cascading. One of the goals of CSS is to allow users greater control over presentation. Depending on the browser and the web site, , a user may choose from various style sheets provided by the designers, or may remove all added styles and view the site using the browser's default styling, or may override just the red italic heading style without altering other attributes.

4.5 BOOTSTRAP:

Bootstrap is a free and open-source CSS framework directed at responsive, mobile-first front-end web development. It contains CSS- and (optionally) JavaScript-based design templates for typography, forms, buttons, navigation and other interface components. Bootstrap is the third- most-starred project on GitHub, with more than 135,000 stars, behind only freeCodeCamp (almost 305,000 stars) and marginally behind Vue.js framework. According to Alexa Rank, Bootstrap getbootstrap.com is in the top-2000 in US while vuejs.org is in top-7000 in US. Bootstrap is a web framework that focuses on simplifying the development of informative web pages (as opposed to web apps). The primary purpose of adding it to a web project is to apply Bootstrap's choices of color, size, font and layout to that project. As such, the primary factor is whether the developers in charge find those choices to their liking. Once added to a project, Bootstrap provides basic style definitions for all HTML elements. The result is a uniform appearance for prose, tables and form elements across web browsers. In addition, developers can take advantage of CSS classes defined in Bootstrap to further customize the appearance of their contents. For example, Bootstrap has provisioned for light- and dark-colored tables, page headings, more prominent pull quotes, and text with a highlight.

Bootstrap is a web framework that focuses on simplifying the development of informative web pages (as opposed to web apps). The primary purpose of adding it to a web project is to apply Bootstrap's choices of color, size, font and layout to that project. As such, the primary factor is whether the developers in charge find those choices to their liking. Once added to a project, Bootstrap provides basic style definitions for all HTML elements. The result is a uniform appearance for prose, tables and form elements across web browsers. In addition, developers can take advantage of CSS classes defined in Bootstrap to further customize the appearance of their contents. For example, Bootstrap has provisioned for light- and dark-colored tables, page headings, more prominent pull quotes, and text with a highlight.

BACK END: 4.6 PYTHON :

Python is an interpreted, high-level, general-purpose programming language. Created by Guido van Rossum and first released in 1991, Python's design philosophy emphasizes code readability with its notable use of significant whitespace. Its language constructs and object-oriented approach aim to help programmers write clear, logical code for small and large-scale projects. Python is dynamically typed and garbage-collected. It supports multiple programming paradigms, including procedural, object-oriented, and functional programming. Python is often described as a "batteries included" language due to its comprehensive standard library.

Python was conceived in the late 1980s as a successor to the ABC language. Python 2.0, released 2000, introduced features like list comprehensions and a garbage collection system capable of collecting reference cycles. Python 3.0, released 2008, was a major revision of the language that is not completely backward-compatible, and much Python 2 code does not run unmodified on Python 3. Due to concern about the amount of code written for Python 2, support for Python 2.7 (the last release in the 2.x series) was extended to 2020. Language developer Guido van Rossum shouldered sole responsibility for the project until July 2018 but now shares his leadership as a member of a five-person steering council. Python interpreters are available for many operating systems. A global community of programmers develops and maintains CPython, an open source [32] reference implementation. A non-profit organization, the Python Software Foundation, manages and directs resources for Python and CPython development.

4.7 MYSQL :

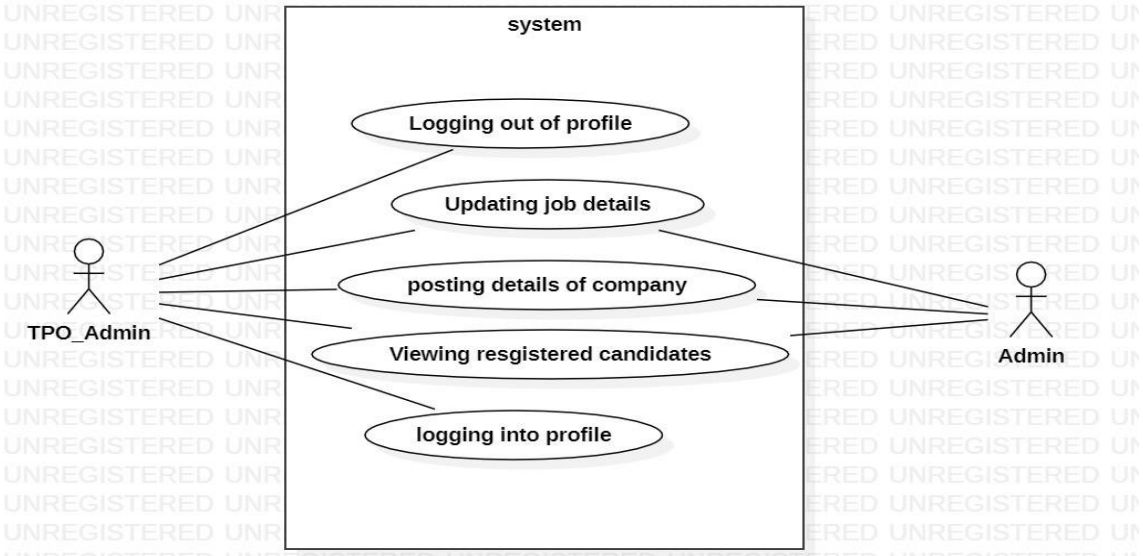
MySQL is an open-source relational database management system (RDBMS) based on Structured Query Language (SQL). Its name is a combination of "My", the name of co-founder Michael Widenius's daughter, and "SQL", the abbreviation for Structured Query Language. A relational database organizes data into one or more data tables in which data types may be related to each other; these relations help structure the data. SQL is a language programmers use to create, modify and extract data from the relational database, as well as control user access to the database. In addition to relational databases and SQL, an RDBMS like MySQL works with an operating system to implement a relational database in a computer's storage system, manages users, allows for network access and facilitates testing database integrity and creation of backups. MySQL is pretty easy to master in comparison with other database software like Oracle Database, or Microsoft SQL Server. MySQL can run on various platforms UNIX, Linux, Windows, etc. You can install it on a server or even in a desktop. Besides, MySQL is reliable, scalable, and fast.

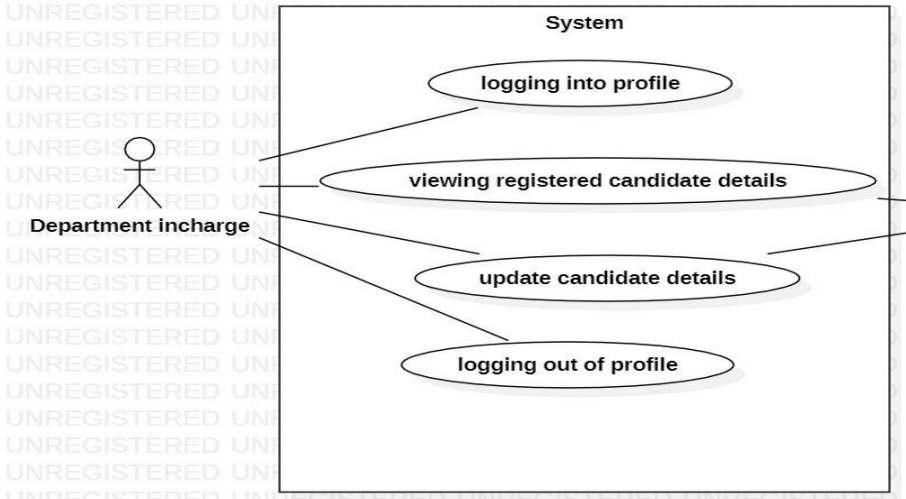
MySQL is a component of the LAMP web application software stack (and others), which is an acronym for Linux, Apache, MySQL, Perl/PHP/Python. MySQL is used by many database- driven web applications, including Drupal, Joomla, phpBB, and WordPress. MySQL is also used by many popular websites, including Facebook, Youtube, Twitter and so on.

5. DESIGN REQUIREMENT ENGINEERING

5.1 Use case Diagram

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use case), and any dependencies between those cases. The main purpose of a use case diagram is to show system functions are performed for which actor. Roles of the actors in the system can be depicted.





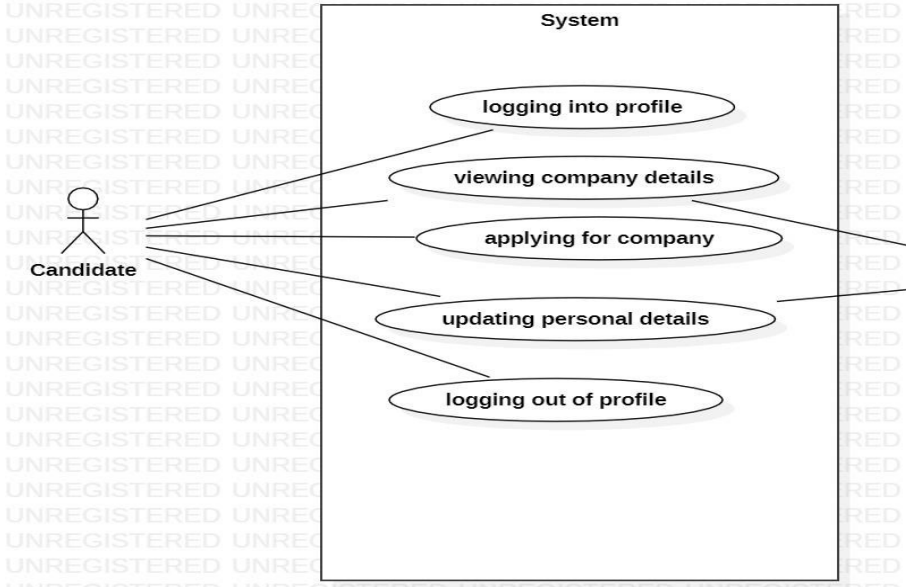


Fig 1. Use Case diagram for TPO, Department incharge Candidate

5.2 Sequence Diagram

A sequence diagram in UML is a kind of interaction diagram which shows how each process of the system operates with one another and in what order. It is a constructed as a message sequence chart. Sequence diagrams are sometimes called event diagrams or timing diagrams.

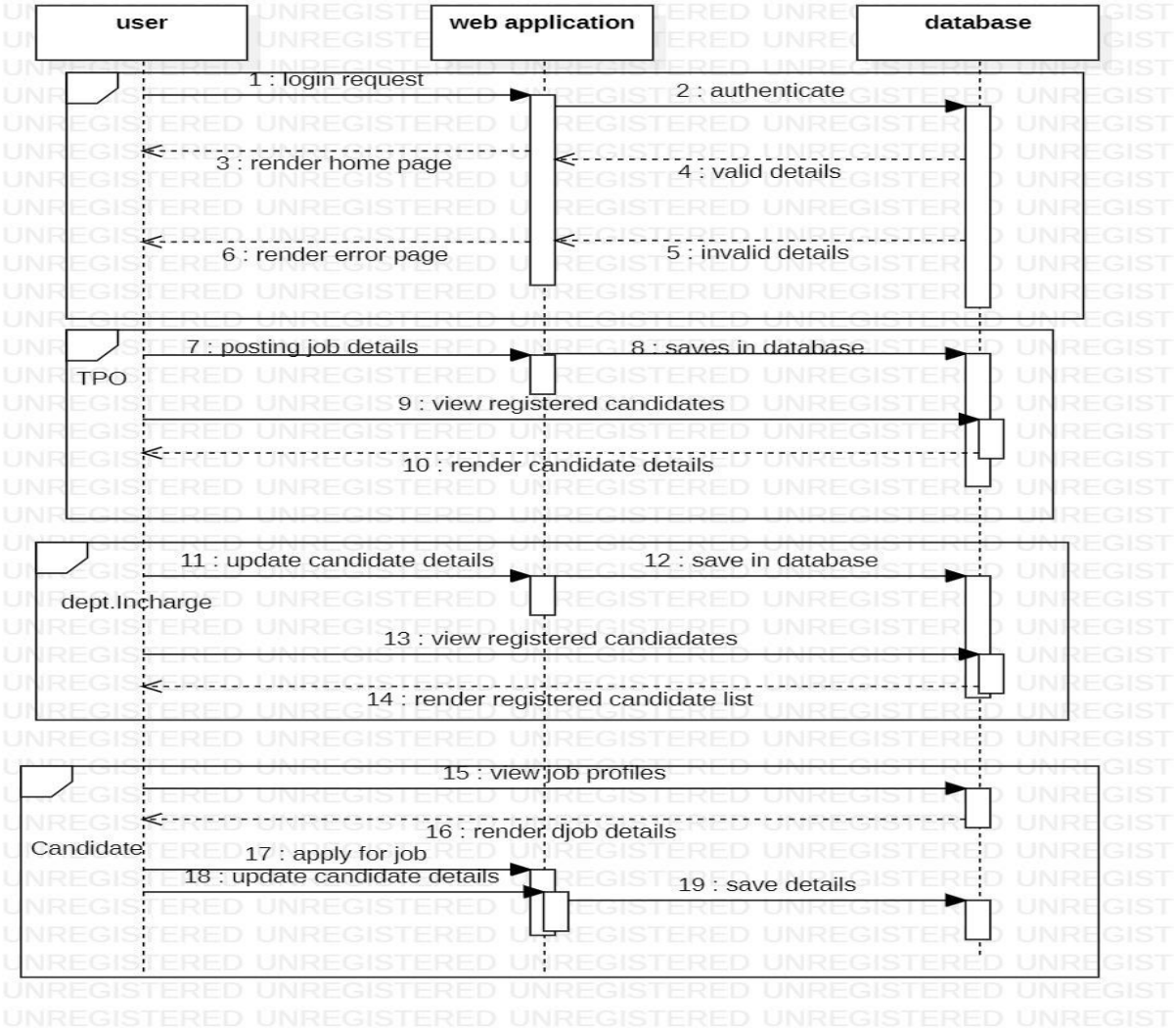


Fig 2. Sequence Diagram

5.3 Activity Diagram

Activity diagram is another important behavioral diagram in UML diagram to describe dynamic aspects of the system. Activity diagram is essentially an advanced version of flow chart that modeling the flow from one activity to another activity.

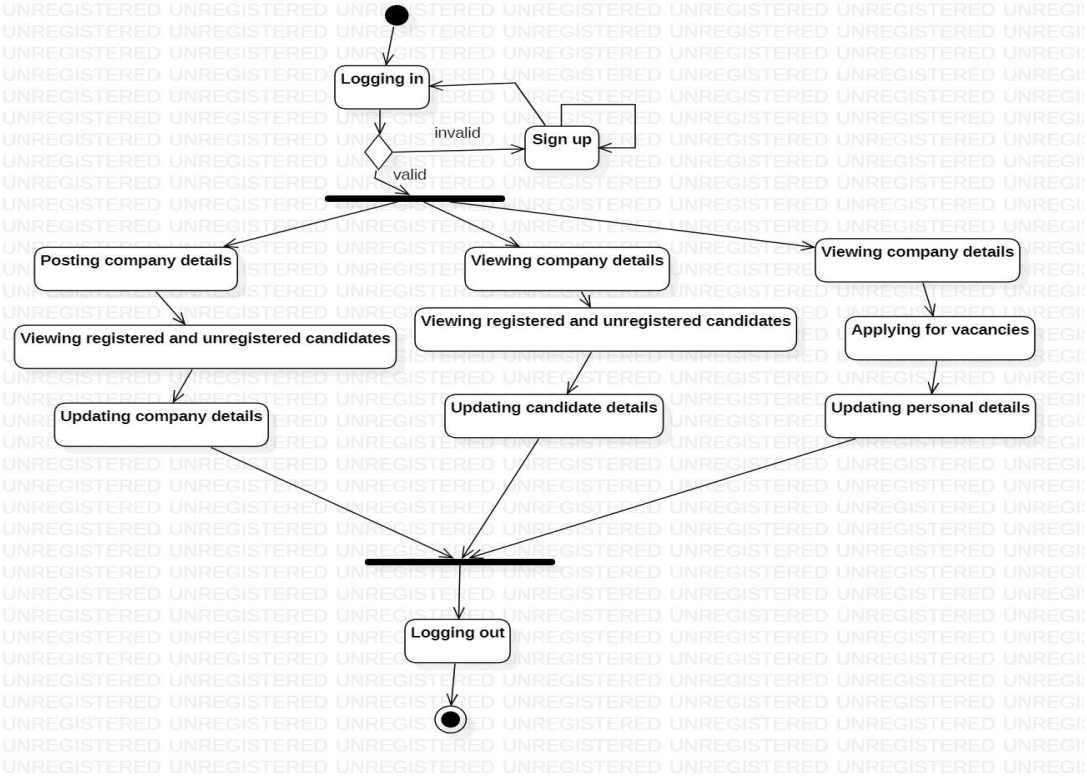


Fig 3. Activity Diagram

---The Activity Diagram shows the activities performed by the TPO, department incharge and candidate.

5.4 Class Diagram

Class diagrams are widely used to describe the types of all objects in a system and their relationships. Class diagrams model class structure and contents using design elements such as classes, packages and objects.

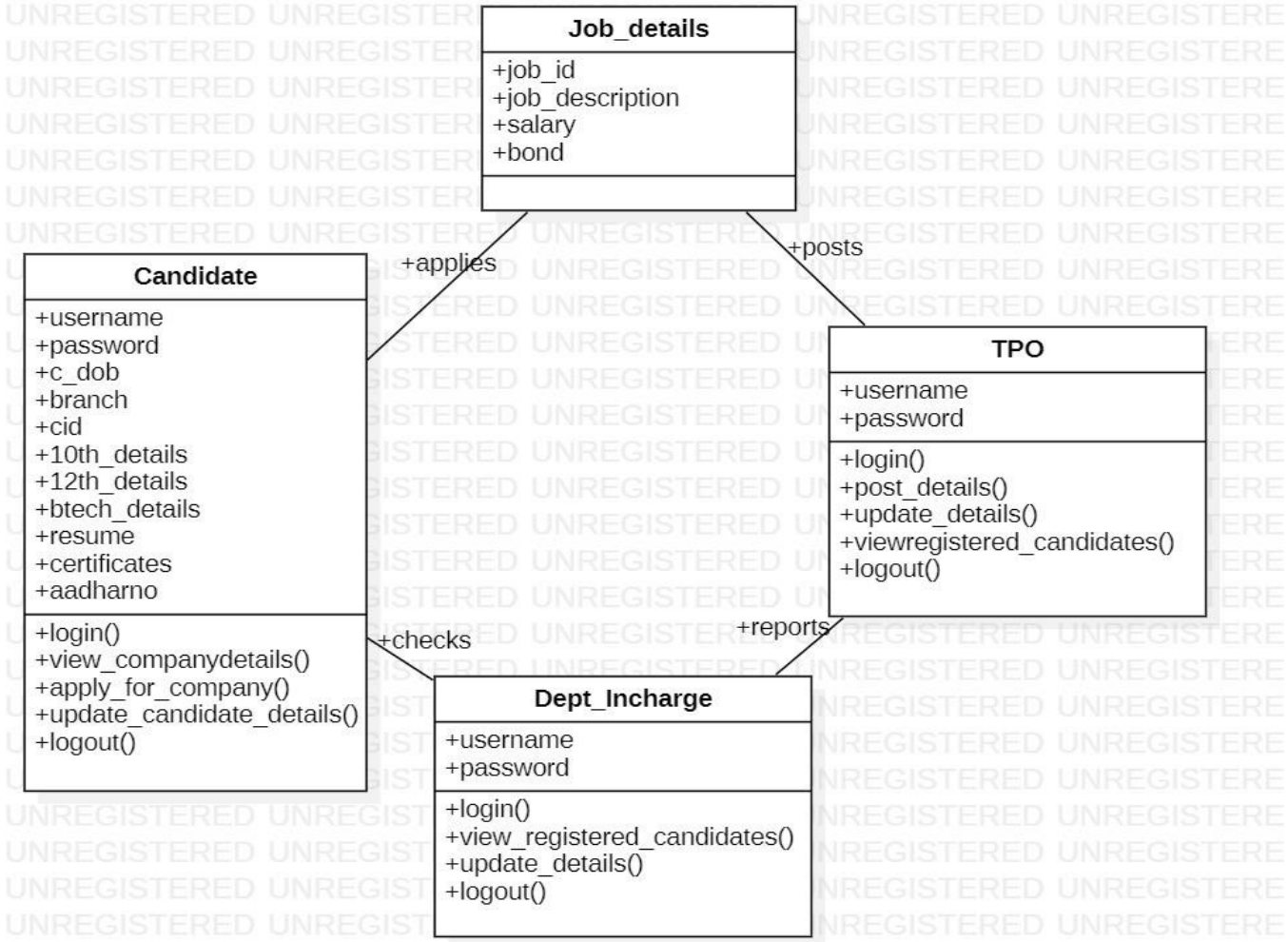


Fig 4. Class Diagram

5.5 Component Diagram

Component diagram is a special kind of diagram in UML. The purpose is also different from all other diagrams discussed so far. It does not describe the functionality of the system but it describes the components used to make those functionalities. Components in the component diagram include documents, database table, files, and executables.

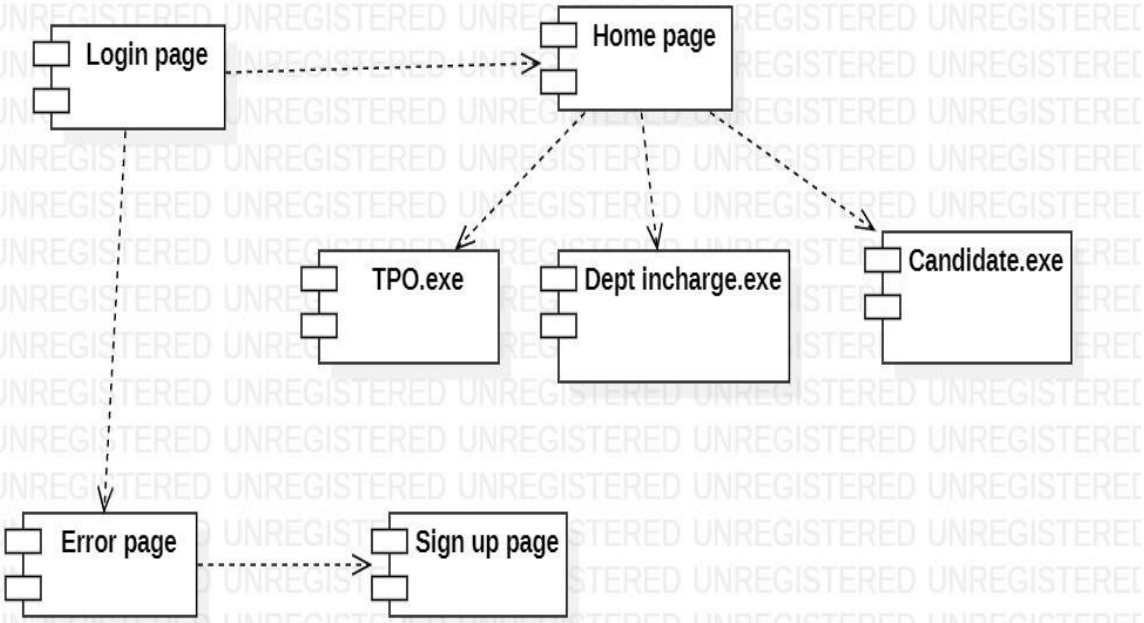


Fig 5. Component Diagram

---The Component Diagram shows all the components of the application. The various components are login page , home page, error page, TPO.exe, Dept incharge.exe, Candidate.exe.

5.6 Collaboration Diagram

A collaboration diagram, also called a communication diagram or interaction diagram, is an illustration of the relationships and interactions among software objects in the Unified Modeling Language (UML).

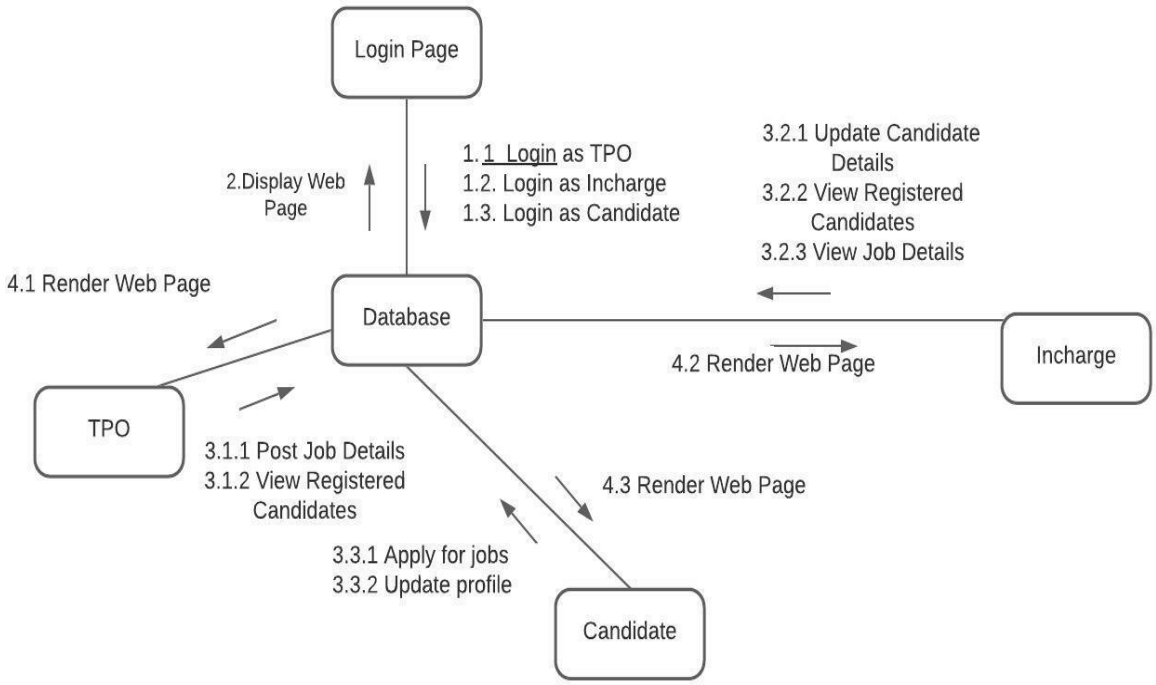


Fig 6. Collaboration Diagram

---The above figure shows the roles, functionality and behaviour of the individual objects of the application.

5.7 Deployment Diagram

A deployment diagram is a UML diagram type that shows the execution architecture of a system, including nodes such as hardware or software execution environments, and the middleware connecting them. Deployment diagrams are typically used to visualize the physical hardware and software ofa system.

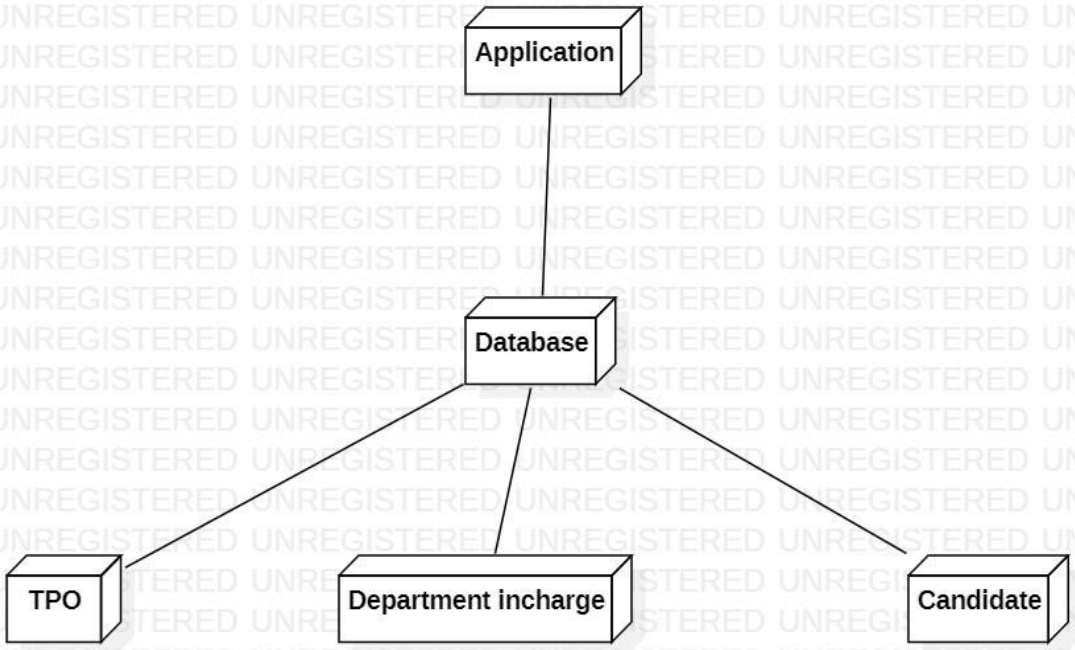


Fig 7. Deployment Diagram

---This diagram illustrates the physical hardware and software of the system.

5.8 Architecture

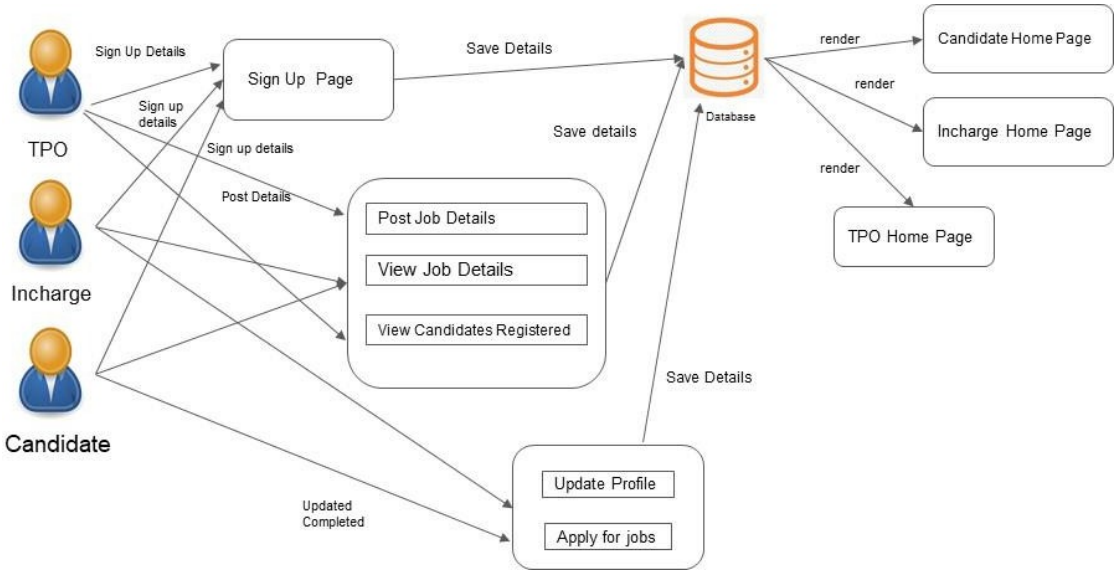


Fig 8. Architecture Design

This figure shows the working of the web application. It shows how the TPO, department incharge and the candidate login to their respective profiles and perform their various functions.

5.9 E-R DIAGRAM

An entity relationship diagram (ERD) shows the relationships of entity sets stored in a database. An entity in this context is a component of data. In other words, E-R diagrams illustrate the logical structure of databases.

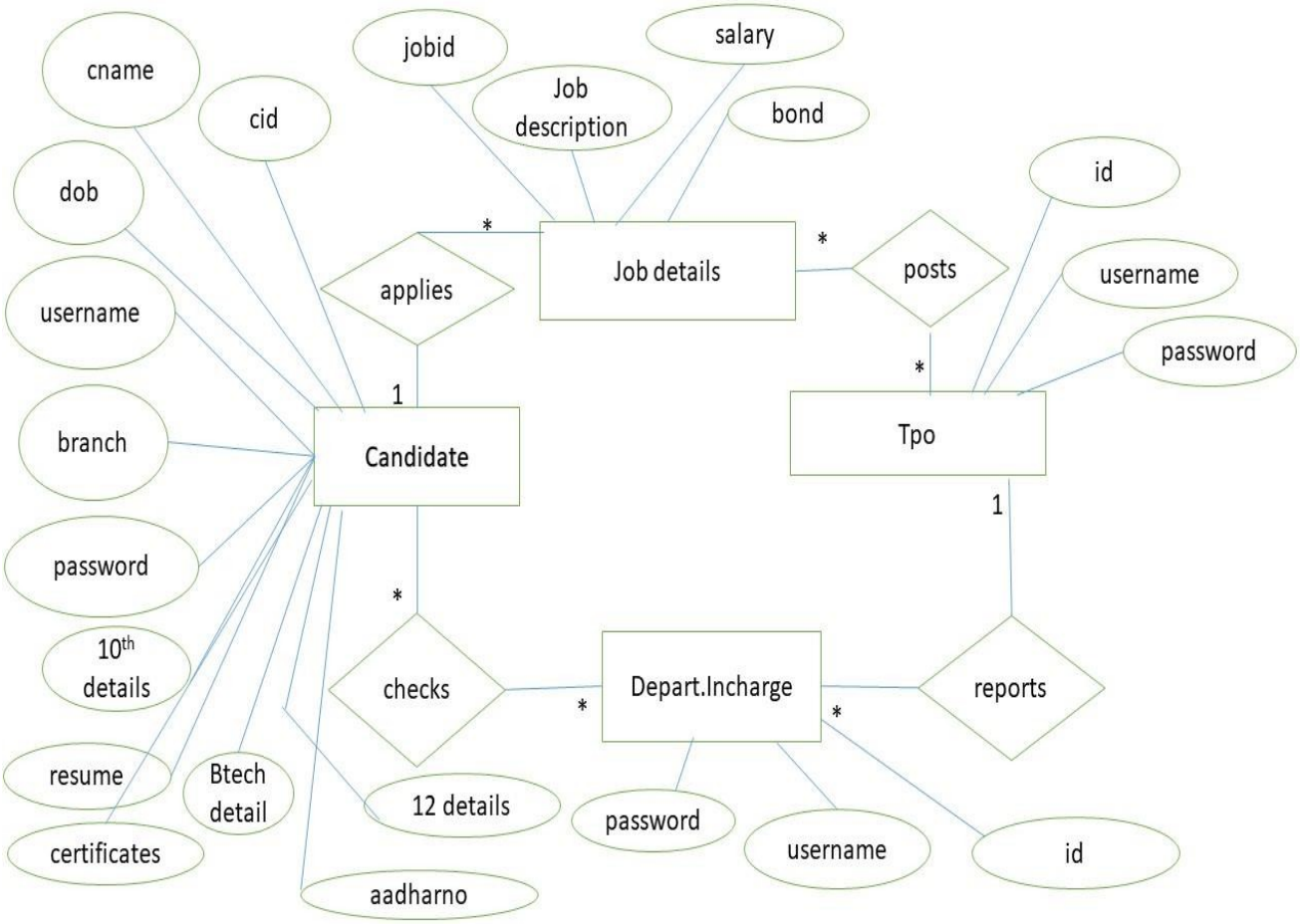


Fig 9. E-R Diagram

6. IMPLEMENTATION

MODULE DESCRIPTION:

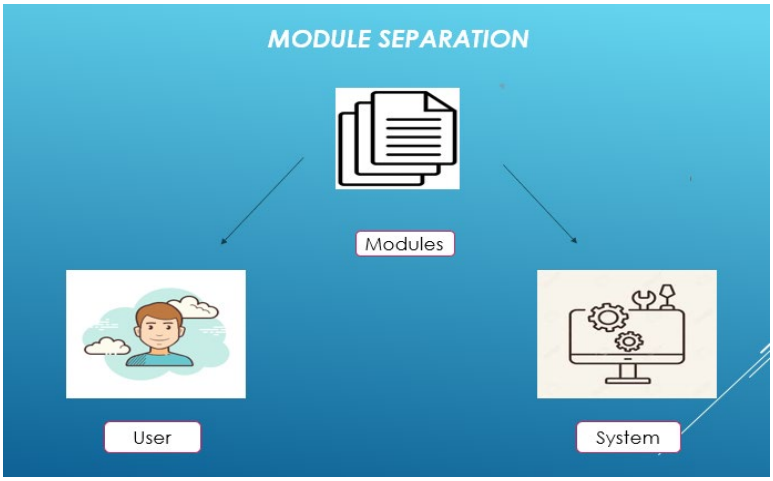


Fig 10. Module Division

6.1 MODULE –1 (USER)

- ❑ It is all about the role of user in which TPO, Department Incharge and student gives the input parameter for their login into their respective accounts after their registration. This module mainly focuses on these following things:

1. A HTML page is created for this approach and this is the front – end of the project.
2. During initial state of creation of this application TPO register at Django admin so that everything is under control of him/his.
3. This webpage consists of several components like registration page containing login and password and other page is login page for TPO, Department Incharge and student.

- ❖ SUBMIT : There will be a submit button which is present on next to input for every registration and login. Whenever we click on this button it will redirect to our necessary requirements like update profile, delete student, modify details, etc.

6.2 MODULE -2 (system)

- ❑ It defines all about the internal process as a back-end.
- ❑ Our main work is focus on fast and efficient that should be simple and easy to use for that we have followed the following:

=>Since we have used Django when we want to make any CRUD operations or any other then we may not need to change entire HTML, CSS, JS. We can change make some modification for all the operations using Django itself at a single place.

=>Database we use here is inbuilt of Django web framework that is SQLite which contains nodes, shards, replicas and other things which make our data process fast in searching, analyzing and many things.

- ⇒ For any kind of queries as per our input users for getting requirement data we are using database as our interface for data we store along with other data containing in it.
- ⇒ The above searching and analyzing of data will be done automatically by these web frameworks since it can handle large amount of data.
- ⇒ This kind of work done easily with the help of sharding which makes our work fast and efficient that is what we want and it will do the work.

6.3 Code Snippets :

Department Incharge:

1.Department Incharge Login Page:

```
{%extends  "campus/base.html"%}

{%block  body_block%}
    <div  class="container">
      <div  class="row">
        <div  class="col-md-6  mx-auto  text-center  mb-5  section-heading">

          <br>
          <h4  class='text-success'><strong>DEPARTMENT
INCHARGELOGIN</strong></h4>
        </div>
      </div>
    </div>
  <div    class="container">

<div  class="jumbotron  ">
        <table  class="hi">
            <tr>

        <form  method="post"  action='{%  url  "dept_login"  %}'>
        {%  csrf_token  %}
        {{form.as_table}}


            <tr>  <td><p><button  type="submit"  class="btn  btn-sm  btn-primary
">Login</button></p></td></tr>
        </form>
                </tr>
                <tr><td><center>
        <a  href="{%  url  'dept_register'  %}"  class="  d-
```

```
inline-flex"    style="color:deepskyblue"><b>New  user?
Register  here</b></a>
                </center></td>
        </tr>
        </table>
    </div>
  </div>

{%endblock%}
```

2.Department Incharge Register Page:

```
<!DOCTYPE  html>
{%extends  "campus/base.html"%}
{%block  body_block%}
{%  load  static  %}
<!--  <script  type  =  "text/javascript"
src="https://ajax.googleapis.com/ajax/libs/jquery/2.1.3/jquery.min.js">
</script>
-->
<script  type='text/javascript'  src="{%  static
'campus/js/jquery-3.3.1.min.js'  %}"></script>

    <div  class="container-fluid">

          <div  class="row">
              <div  class="col-md-6  mx-auto  text-center  mb-5  section-heading">
              <br>
                <h2  class="text-success">Department  Incharge  Register</h2>
              </div>
            </div>
          </div>
          <div  class="container">
          <div  class="jumbotron"  style="background-color:  gainsboro  !important;">
              <div  class="login">
                 {%  if  messages  %}
                     <ul>
                         {%  for  message  in  messages  %}
                         <li>{{  message  }}</li>
                         {%  endfor  %}
                     </ul>
```

```
                {%  endif  %}
                {%  if  form.errors  %}
            {%  for  field  in  form  %}
                {%  for  error  in  field.errors  %}
                    <div  class="alert  alert-danger"  style="color:  #bd2130;">
                        <strong>{{  error|escape  }}</strong>
                    </div>
                {%  endfor  %}
            {%  endfor  %}
            {%  for  error  in  form.non_field_errors  %}
                <div  class="alert  alert-danger">
                    <strong  style="color:  #bd2130">  {{  error|escape  }}</strong>
                </div>
            {%  endfor  %}
        {%  endif  %}

                 <form  method="post"  action='{%  url  "dept_register"  %}'>
                    {%  csrf_token  %}
                    <table>
                       {{  form.as_table  }}
                        <tr>
                           <td></td>
                            <td><button  type="submit"  class="btn  btn-sm
 btn-primary"  name="submit"  value="Register";>REGISTER</button></td>
                        </tr>
                    </table>
                </form>
            </div>
        </div>
    </div>
    <script>
        $(document).ready(function(){
          $('[for="id_hr_name"]').html('Department')
          $('.helptext').addClass('d-none')
      })
      </script>
{%endblock%}
```

3. Department Incharge Home Page:

```
<!DOCTYPE  html>
{%extends  "campus/base.html"%}

{%block  body_block%}
```

```
    <div  class="container">
      <div  class="row">
        <div  class="col-md-6  mx-auto  text-center  mb-5  section-heading">

          <br>
          <h2  class='text-success'>DASHBOARD</h2>
        </div>
      </div>
    </div>
    <div  class="container">
      <div  class="container">
        <div  class="jumbotron"  style="background-color:  gainsboro  !important;">
            <table  class="pos2">
                <tr>
                    <td><a  href="
 http://127.0.0.1:8000/company/company_login/jp/    "  class="btn  btn-lg  btn-
 primarymx-5  d-inline-flex  "><b  style="font-size:  2rem">Post
 vacancy</b></a></td></tr><tr>
                <td><a  href="http://127.0.0.1:8000/company/company_login/jd/
 "class="btn  btn-lg  btn-primary  mx-5  d-inline-flex"><b  style="font-size:
 2rem">Update  vacancy</b></a></td></tr><tr>

                </tr>
                <tr>  <td  ><a  href="
 http://127.0.0.1:8000/company/company_login/deletevacan/  "  class="btn  btn-lg
 btn-
primary  mx-5  d-inline-flex"><b  style="font-size:
2rem">Deletevacancy</b></a></td>

                </tr>
                <tr>  <td  ><a  href="
 http://127.0.0.1:8000/company/company_login/viewpos/  "  class="btn  btn-lg
 btn-primary  mx-5  d-inline-flex"><b  style="font-size:  2rem">Posted
 vacancies</b></a></td>

                </tr>
                <tr>
                  <td>
                  <a  href="{%  url  'applied_studs_comp'  %}"  class="btn  btn-
 lg  btn-primary  mx-5  d-inline-flex"><b  style="font-size:  2rem">Applied
 students</b></a>

                </td>
```

```
                </tr>
            </table>
        </div>

      </div>
    </div>
{%endblock%}
```

4. Department Incharge Total Vacancies Page:

```
<!DOCTYPE html>
{%extends "campus/base.html"%}

{%block body_block%}
    <div class="container-fluid">
           <div class="row">
              <div class="col-md-6 mx-auto text-center mb-5 section-heading">
                <br>
                <h2 class='text-success'>Update Vacancies</h2>
              </div>
           </div>
         </div>
         <div class="container">
         <div class="jumbotron" style="background-color: gainsboro !important;">
             <form method="post">
{% csrf_token %}
                 <table border="1px" style="text-align: center;border-
collapse:collapse;background-color:rgb(176, 240, 195)" >
                    <tr >
                        <th style="font-size:1.5rem;color: #16156d">job id</th>
                        <th style="font-size:1.5rem;color: #16156d">Company</th>
                        <th style="font-
size:1.5rem;color:#16156d">Designation</th>
                        <th style="font-size:1.5rem;color: #16156d">Salary</th>
                        <th style="font-size:1.5rem;color: #16156d">Bond</th>
                        <th style="font-size:1.5rem;color: #16156d">IT</th>
                        <th style="font-size:1.5rem;color: #16156d">Mech</th>
                        <th style="font-size:1.5rem;color: #16156d">Civil</th>
                        <th style="font-size:1.5rem;color: #16156d">Ece</th>
                        <th style="font-size:1.5rem;color: #16156d">EEE</th>
```

```
                        <th  style="font-size:1.5rem;color:  #16156d">CSE</th>
                        <th  style="font-size:1.5rem;color:
                        #16156d">Chemical</th>
 <th  style="font-size:1.5rem;color:  #16156d">Posted  Date</th>
 <th  style="font-size:1.5rem;color:  #16156d">Till  Date</th>
  <th  style="font-size:1.5rem;color:  #16156d">Edit</th>
                    </tr>
                    {%    for  i  in  y  %}
                    <tr>
                    <td>{{  i.job_id  }}</td>
                    <td>{{  i.company_name  }}</td>
                    <td>{{  i.designation  }}</td>
                    <td>{{  i.salary  }}</td>
                    <td>{{  i.bond_years  }}</td>
                    <td>{{  i.information_technology  }}</td>
                    <td>{{  i.mech  }}</td>
                    <td>{{  i.civil  }}</td>
                    <td>{{  i.ece  }}</td>
                    <td>{{  i.eee  }}</td>
                    <td>{{  i.cse  }}</td>
                    <td>{{  i.chemical  }}</td>
                    <td>{{  i.posted_date  }}</td>
                    <td>{{  i.till_date  }}</td>
                    <td  ><a  href='{%  url  "jobdesc_edit"  i.job_id
                    %}'>edit</a></td>
                    </tr>
                    {%  endfor  %}
                </table>
            </form></div></div></div>
{%  endblock  %}
```

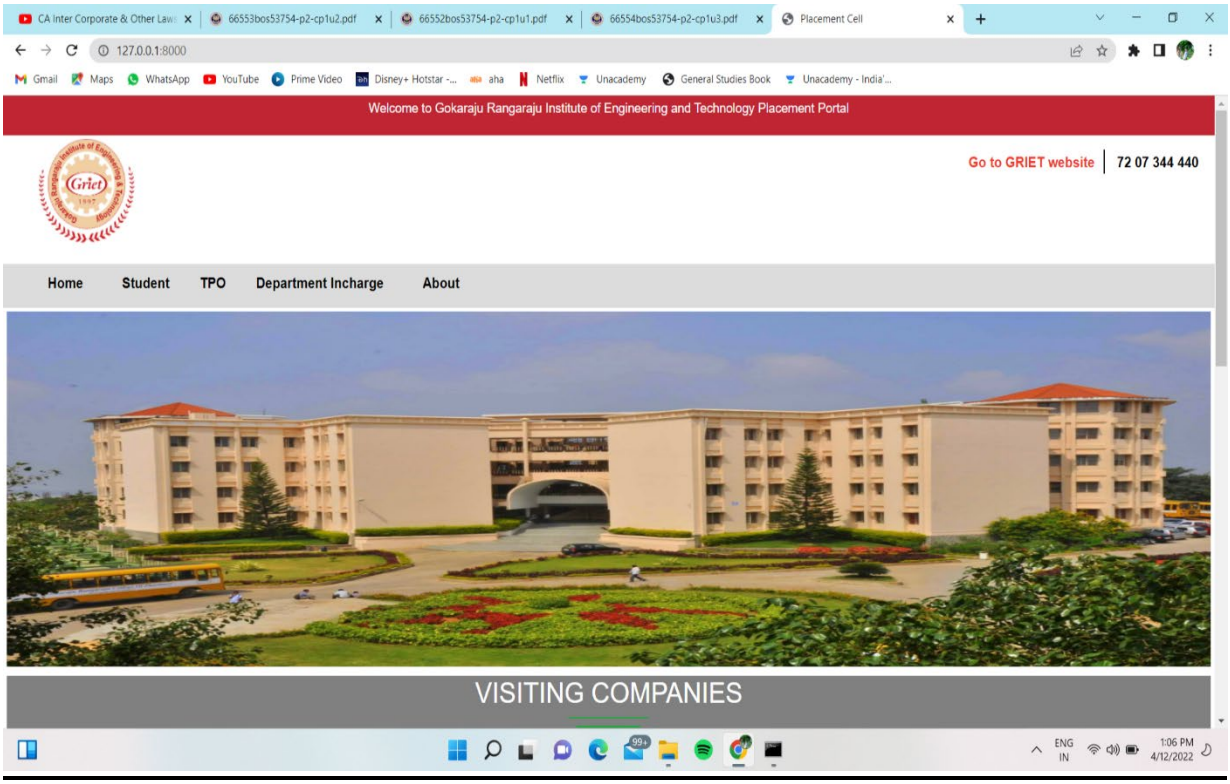
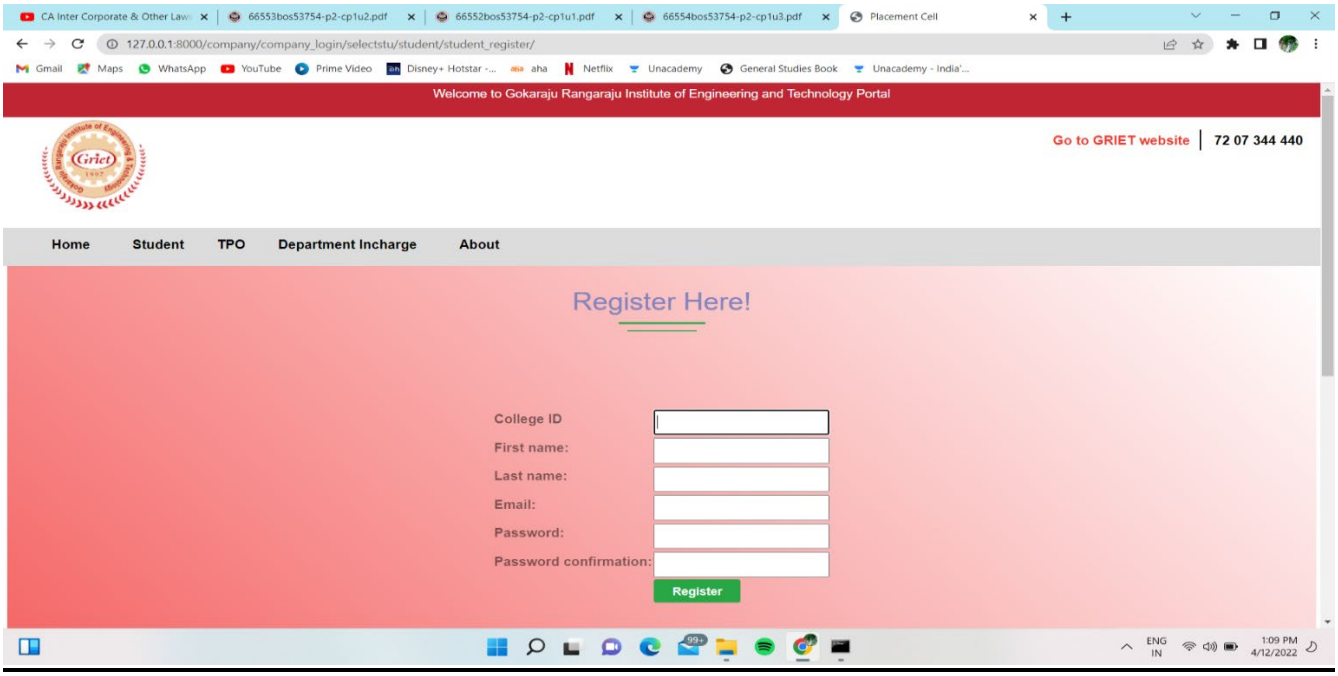


Fig 11. Home Page Interface



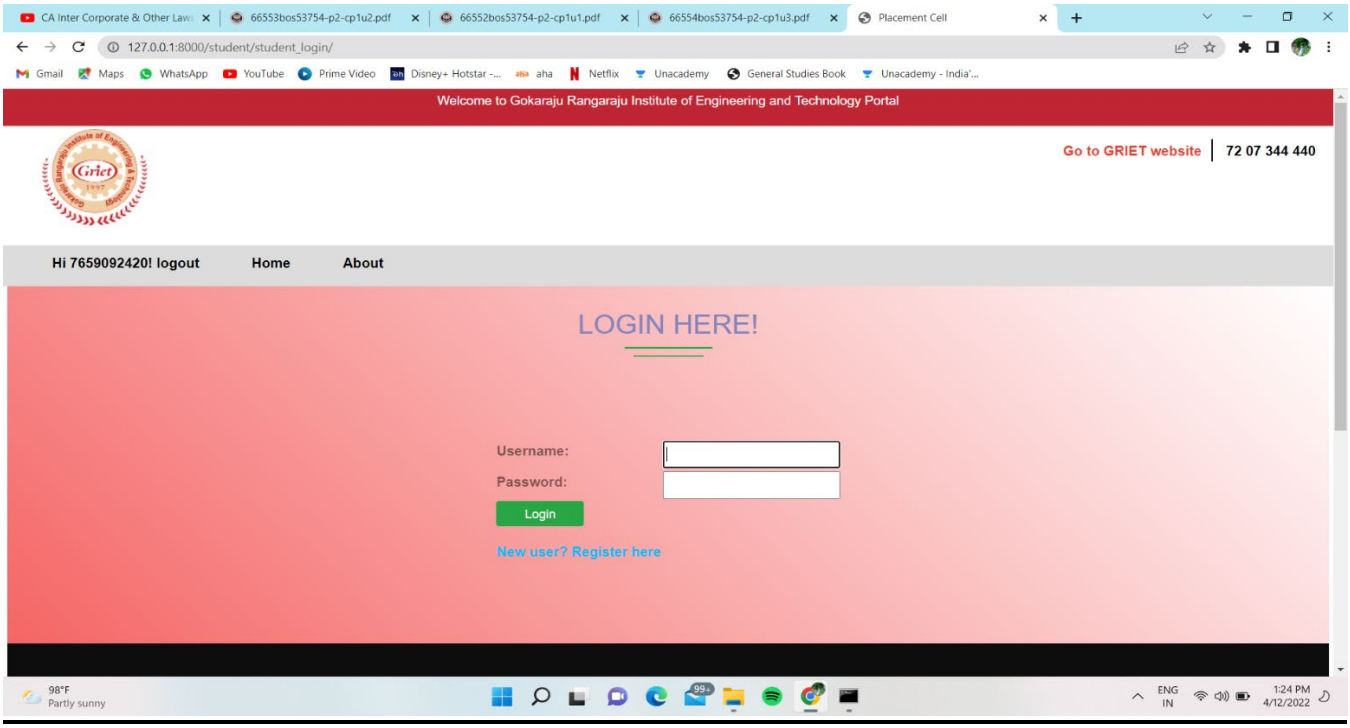
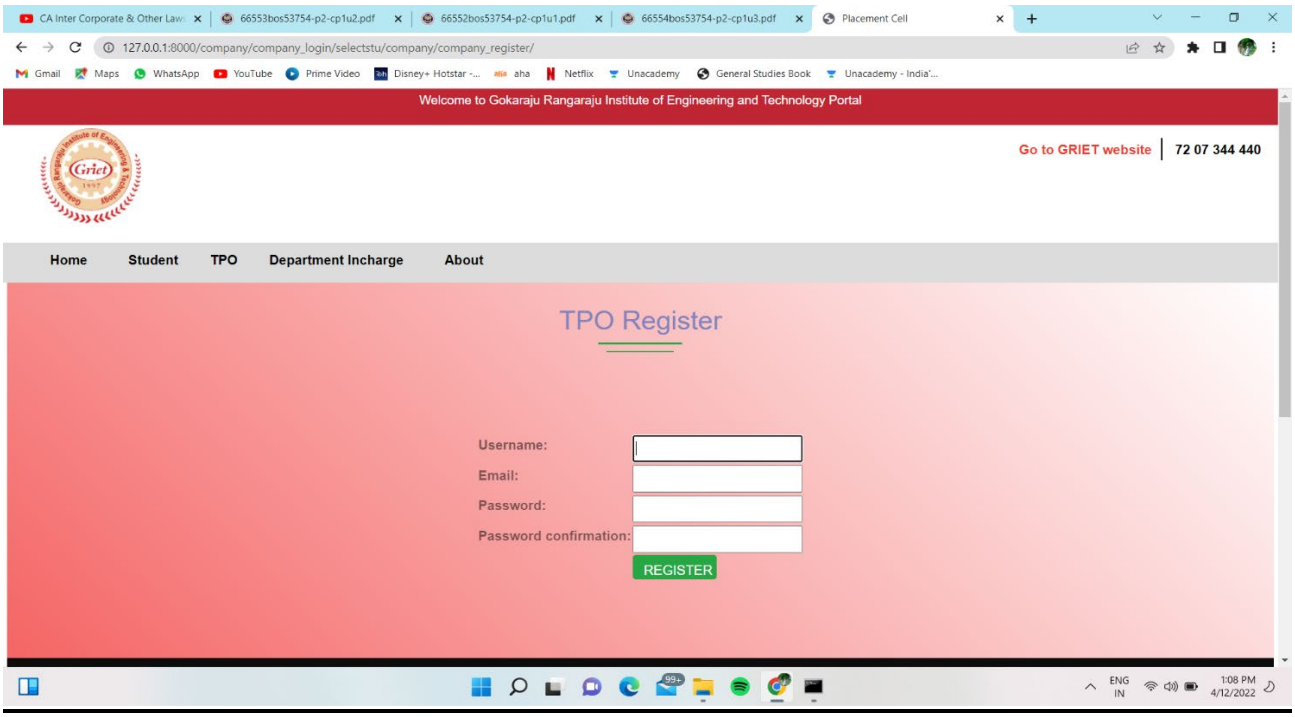


Fig 12. Student Register &Login Page



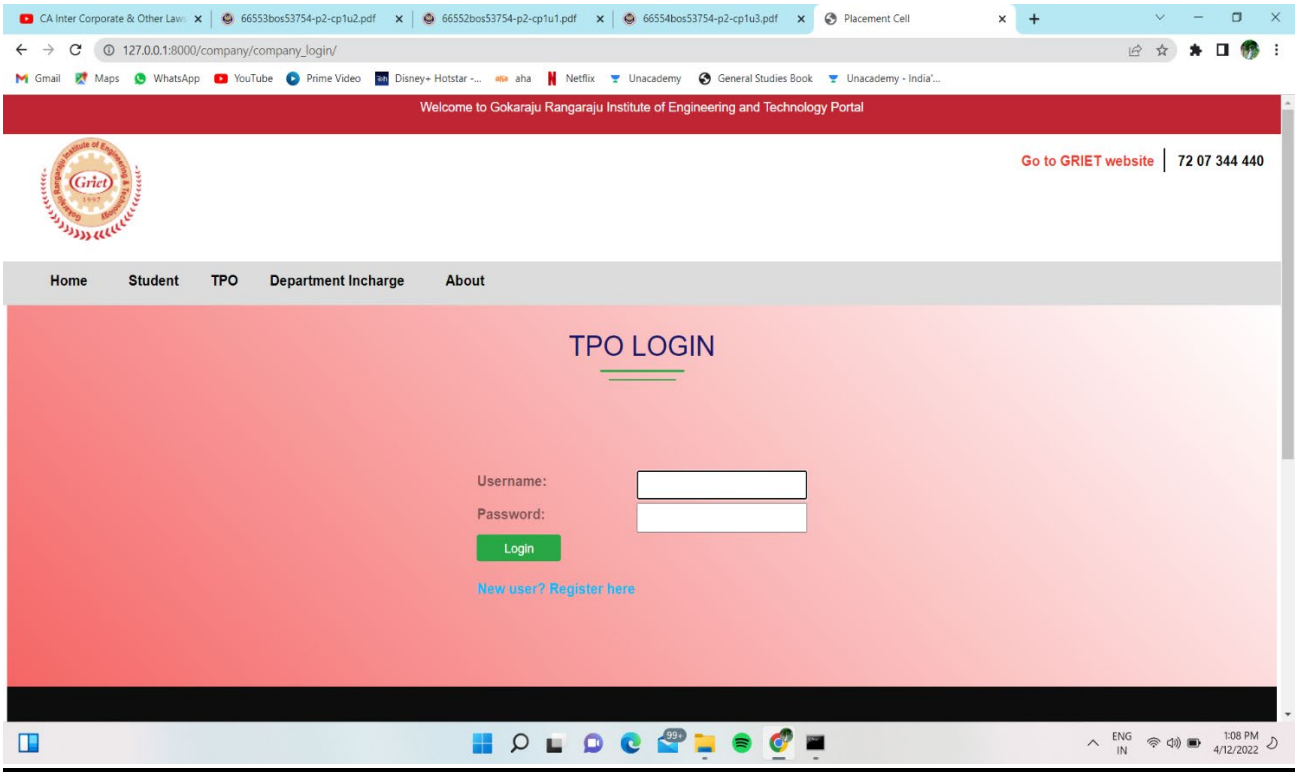
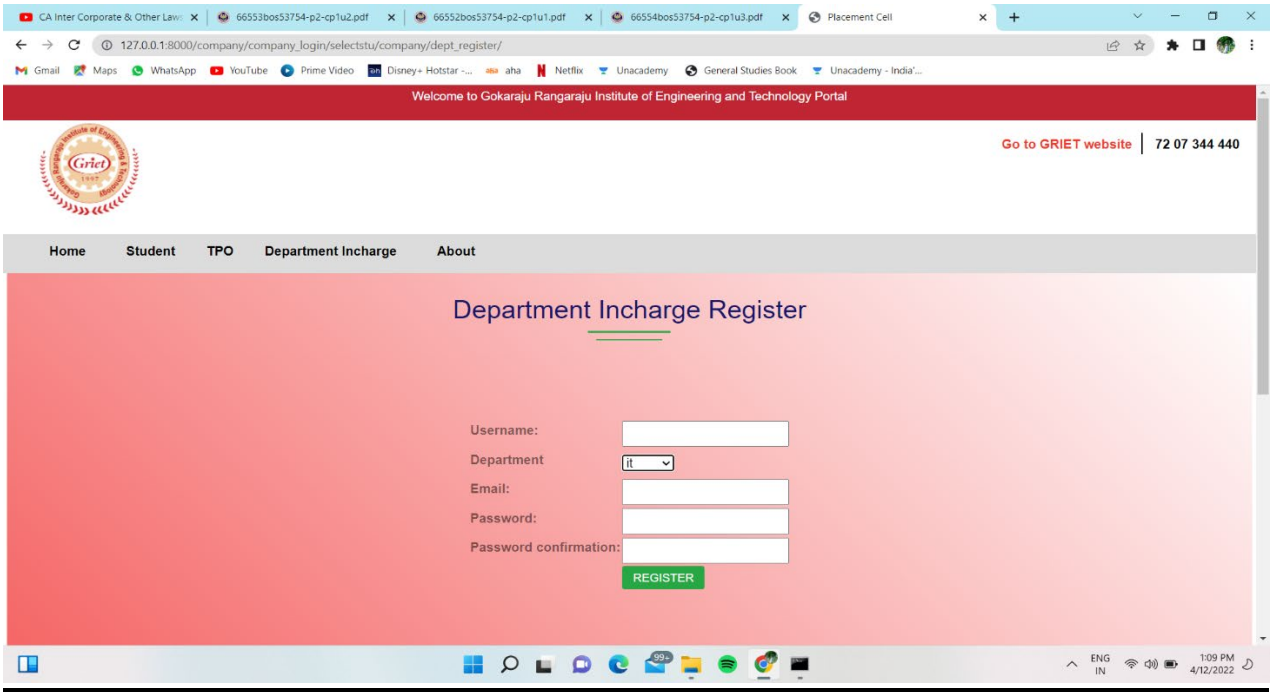


Fig 13. TPO Register & Login Page



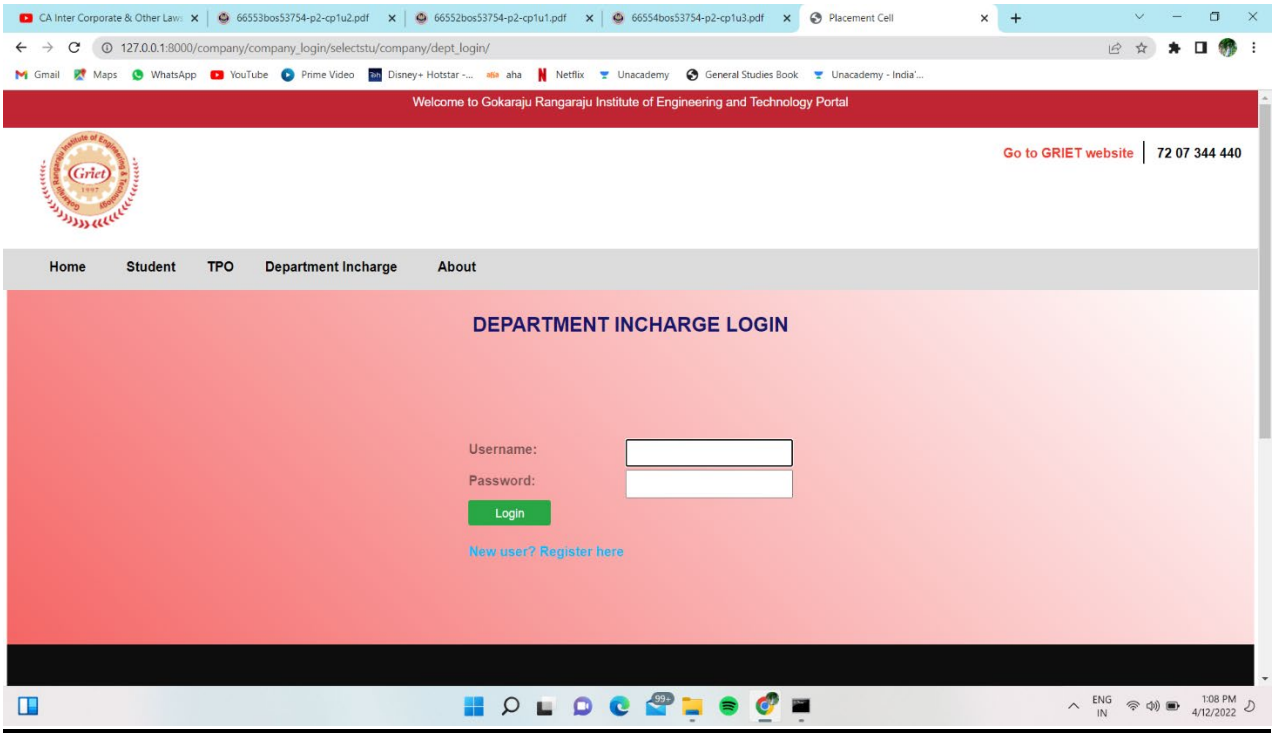


Fig 14. Dept Incharge Register & Login Page

7. SOFTWARE TESTING

7.1 Test Plan:

In software project management, software testing, and software engineering, verification and validation (V&V) is the process of checking that a software system meets specifications and that it fulfills its intended purpose. It may also be referred to as software quality control. It is normally the responsibility of software testers as part of the software development lifecycle. Validation checks that the product design satisfies or fits the intended use (high-level checking), i.e., the software meets the user requirements. This is done through dynamic testing and other forms of review. Verification and validation are not the same thing, although they are often confused. Boehm succinctly expressed the difference between

Validation : Are we building the right product?

Verification : Are we building the product right?

According to the Capability Maturity Model (CMMI-SW v1.1)
Software Verification: The process of evaluating software to determine whether the products of a given development phase satisfy the conditions imposed at the start of that phase.

Software Validation: The process of evaluating software during or at the end of the development process to determine whether it satisfies specified requirements.

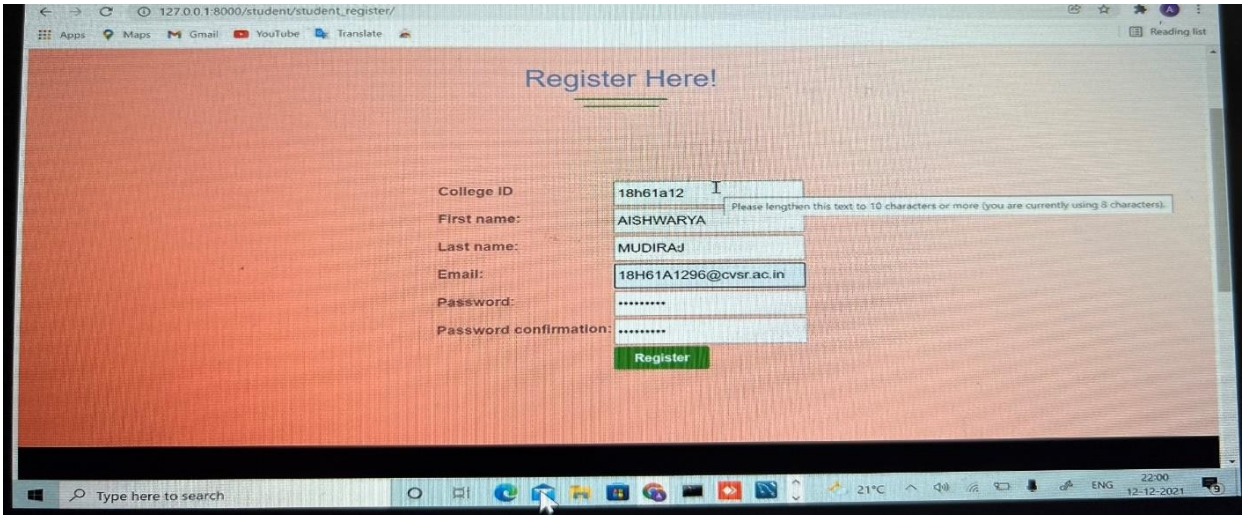
7.2 Testing on our System:

TEST CASES:

Test cases are built around specifications and requirements, i.e., what the application is supposed to do. Test cases are generally derived from external descriptions of the software, including specifications, requirements and design parameters. Although the tests used are primarily functional in nature, non-functional tests may also be used. The test designer selects both valid and invalid inputs and determines the correct output, often with the help of an oracle or a

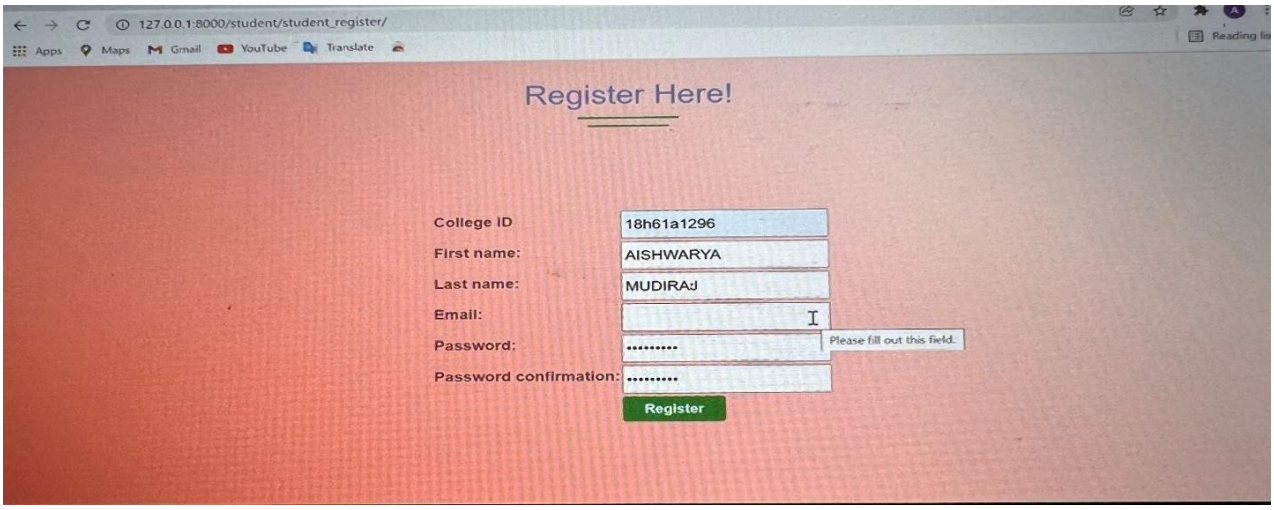
previous result that is known to be good, without any knowledge of the test object's internal structure.

TEST CASE 1:



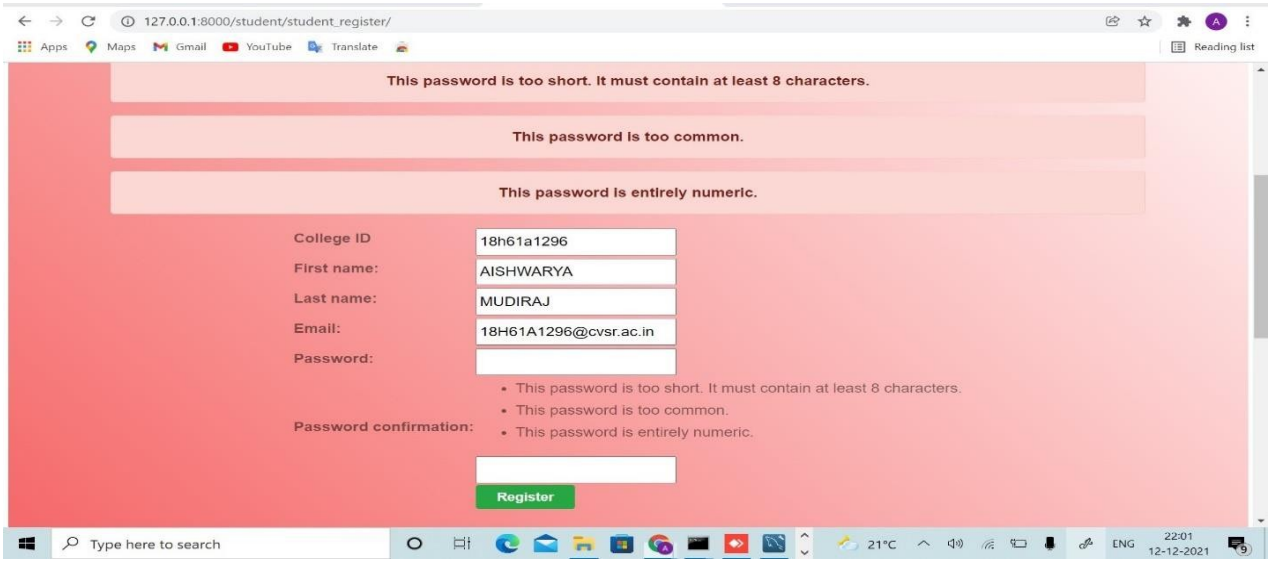
--This case checks whether the College ID given is minimum of 10 characters or not. College ID oflength less than 10 characters is not acceptable

TEST CASE 2:



--If any field is left blank a pop-up to please fill out this field occurs.

TEST CASE 3:



--The password given at the time of register must satisfy the following constraints: must be at least 8 characters, should not be too common or entirely numeric.

8.RESULTS

DEPT VIEW:

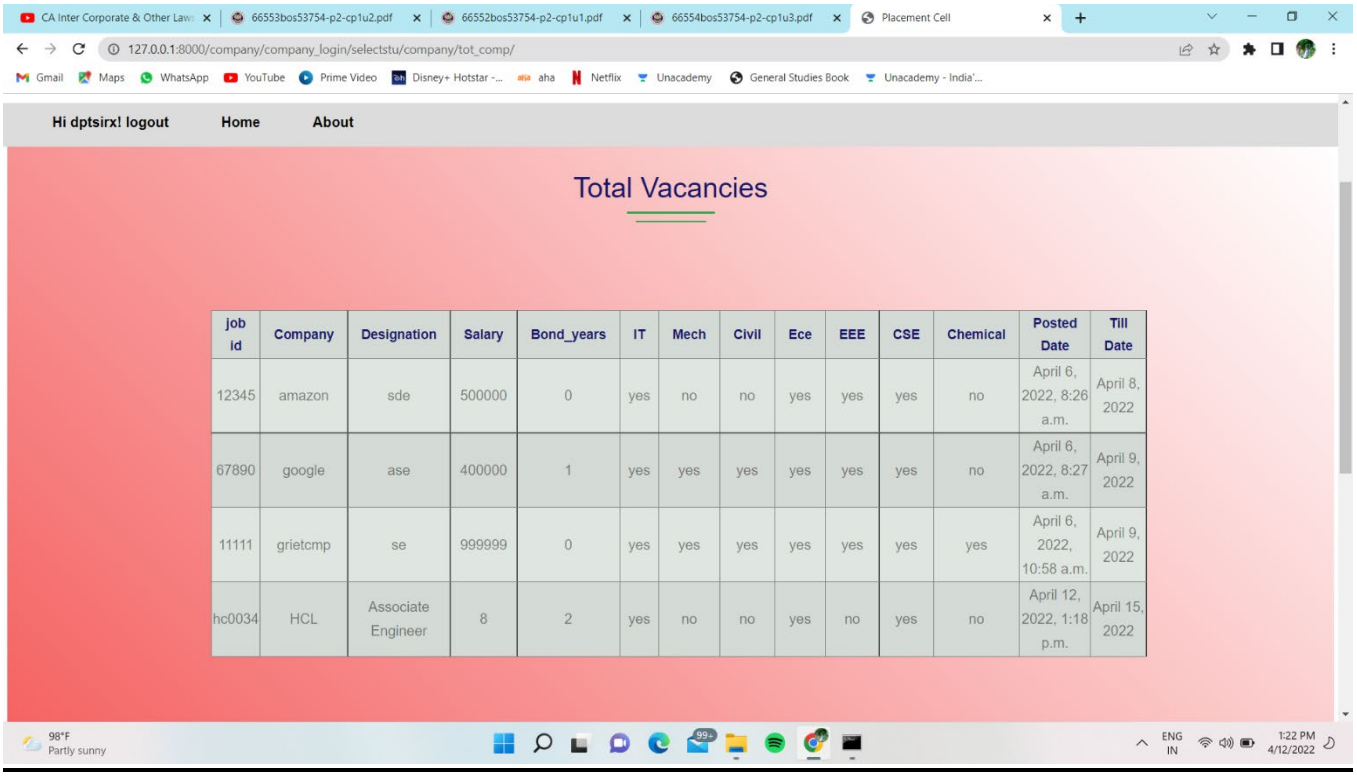


Fig 15. Dept Incharge view page

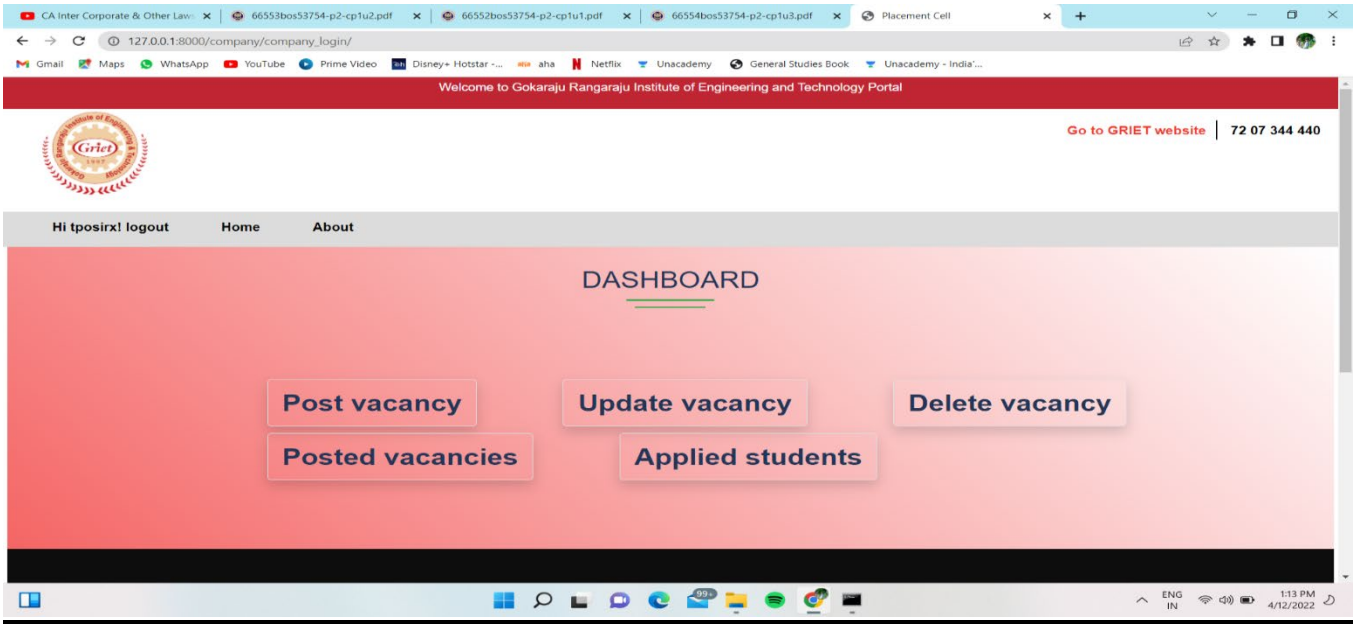


Fig 16. TPO Dashboard

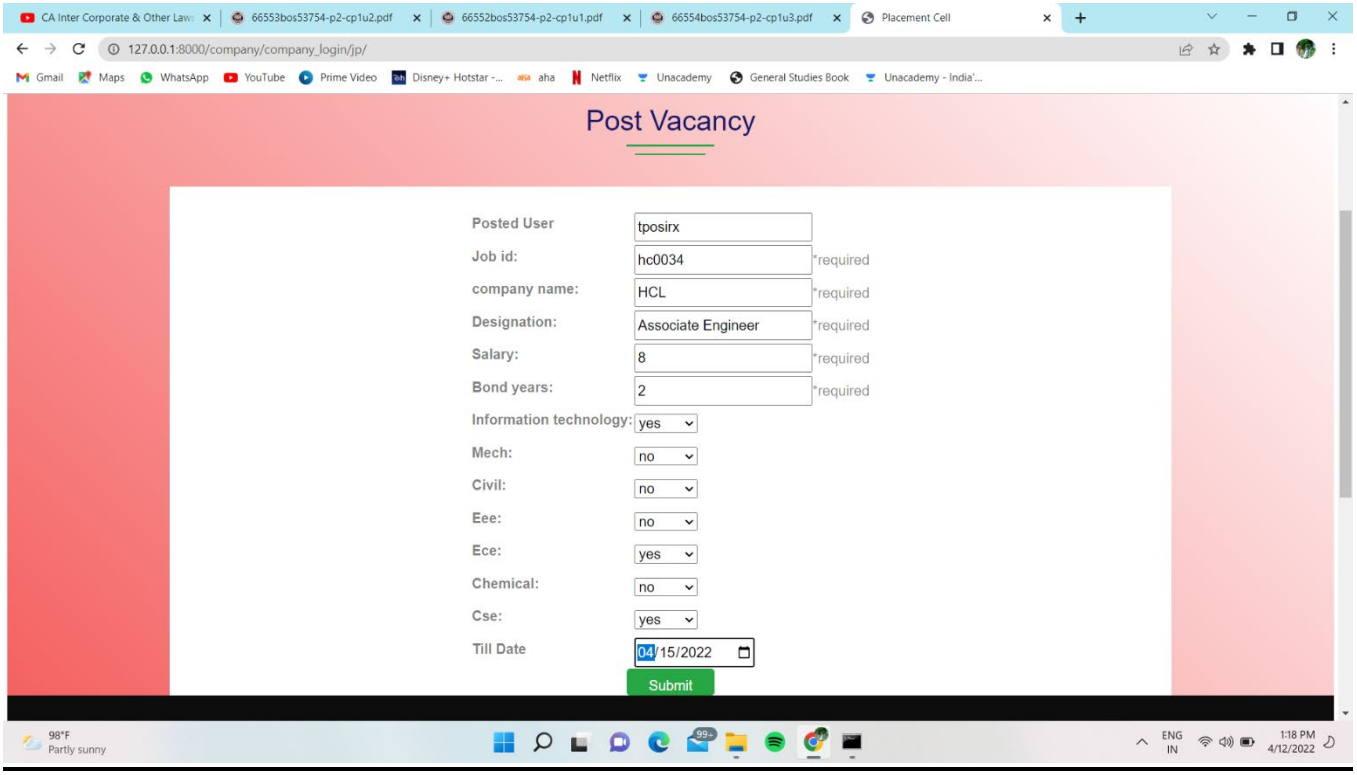


Fig 17. TPO Post Vacancy

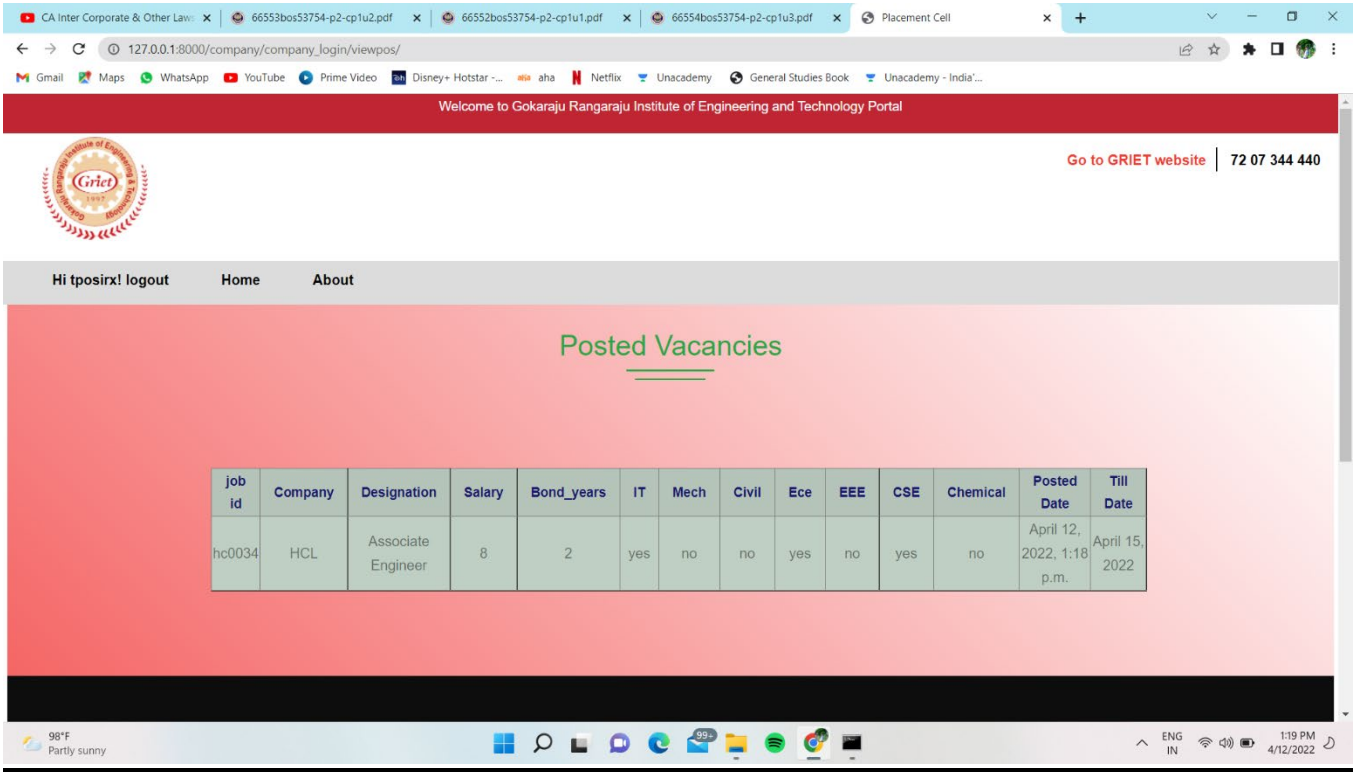


Fig 18. TPO Posted Vacancies

9. CONCLUSION AND FUTURE ENHANCEMENTS

Conclusion:

Increasing need of comfort and inculcating all the data at one place has always been a challenging process for everybody. With the introduction of this web based training and placement portal we promise to make the lives of students and administration a little easier by proposing an alternative for the current system being used. Easy accessibility and functioning of this portal will allow easy management of the allocation process during placement period. With the increasing demand of digitalization in every aspect of day to day activities we can anticipate the great demand for such portals in the near future and the comfort it will bring with it to the lives of all. Also the rapidly increasing concerns of global warming due to increase deforestation for large amount of paper that it requires we here have a minor role to save Mother Nature. So we hope all of you can sit back and relax and enjoy the luxury of Digitalization. More so in this busy and exhausting life we are saving one of the most crucial factor that keeps us running that is human energy.

Future Enhancements:

The future enhancement of this project could be the change of the design of front end which could be in form of disclaimers or pop-ups and also viewing the profile picture of each student in their respective profiles. Another enhancement to this project will be the report generation of various reports like student list, company list etc. which could be printed on paper for future reference. Also updates to student profile by their respective department incharge can be made to keep the veracity of the student profile intact. The generation of notifications to respective emails can also be made to make the application more effective.

10. BIBLIOGRAPHY

Journals:

[1]. Santhosh Kumar H," Online Training and Placement Management system" ,International Journal of Engineering Research Technology (IJERT),ICACT - 2016Conference Proceedings.

[2]. Mythili M, Aishwarya R, Shenbagam P, Sandhiya C," E Placement Management",International Jour- nal of Pure and Applied Mathematics(IJPAM), Volume 119 No. 102018,1823-1834.

[3]. Anjali v, Jeyalakshmi PR, Anbubala R,Sri Mathura devi, Ranjini.V, "Web BasedPlacement Management System", International Journal of Computer Science and Information Technologies(IJCSIT), Vol. 7 (2) , 2016, 760-763.

[4]. Rajnesh Tripathi, Raghvendra Singh, Jaweria Usmani, "Campus Recruitment and Placement Sys- tem", International Conference on Recent Innovations in Science and Engineering(Icrise-18), April, 2018. [5]. Laracasts https://laracasts.com/

Websites:

- https://getbootstrap.com/
- https://docs.djangoproject.com/en/4.0/
- https://www.uml-diagrams.org/
- https://www.tutorialspoint.com/software_testing_dictionary/manual_testing.html
- https://www.w3schools.com/python/
- https://www.tutorialspoint.com/python/index.htm

TEXTBOOKS :

- Programming Python , Mark Lutz Head First Python, Paul Barry .
- Core Python Programming, R. Nageswara Rao Learning with Python,

Allen B.Downey