**Training and Placement Cell**

A project submitted to

**UKA TARSADIA UNIVERSITY**

in partial fulfillment of the requirements for the degree of

**Bachelor of Science**

in

**Information Technology**

for

**5 Years Integrated M.Sc. (IT)**

By

**Shah Manav** (201706100110032),

**Sharma Bhavani** (201606100110047),

**Shah Vikrant** (201606100110062)

Guided by

**Ms. Bhumika Patel**

**Professor**



**Babu Madhav Institute of Information Technology**

### Uka Tarsadia University

### Bardoli – 394350

### May 2020

**BABU MADHAV INSTITUTE OF INFORMATION TECHNOLOGY, UTU**

**CERTIFICATE**

This is to certify that **Shah Manav (201606100110032), Sharma Bhavani (201606100110047), Shah Vikrant (201606100110062),** has submitted project entitled **“Training and Placement Cell”** as the partial fulfillment for the award of the degree of Bachelor of Science in Information Technology for 5 Years Integrated M.Sc. (IT) in 2019 – 2020.

**Date:**

**Place:**

**Ms. Bhumika Patel Dr. Jitendra Nasriwala**

Guide Program Coordinator,

**External Examiner**



**Babu Madhav Institute of Information Technology,**

**Uka Tarsadia University,**

**Bardoli - 394350**

Acknowledgement

The success and final outcome of this project required a lot of guidance and assistance from many people and we extremely privilege to have got this all along completion of our project.

We respect and thank **Dr. Jitendra Nasriwala**, for providing us this opportunity to do the project work and giving us support and guidance which made me complete the project duly.

We heartily thank our internal project guide **Ms. Bhumika Patel**, Professor, BMIIT for her guidance and suggestions during this project work.

We are thankful to get constant encouragement, support and guidance from all teaching staffs of BMIIT which helped us in successfully completing of our project work.

Shah Manav [201606100110032]

Sharma Bhavani [201606100110047]

Shah Vikrant [201606100110062]

**Table of content**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Chapters** | **Particulars** | | | **Page no.** |
| **1** | **Introduction** | | |  |
| 1.1 | Project Definition | |  |
| 1.2 | Purpose | |  |
|  | 1.3 | Scope | |  |
|  | | | | |
| **2** | **Software Development Platform** | | |  |
| 2.1 | Technology Used | |  |
|  | 2.2 | Technology Description (Tools Description) | |  |
|  | | | | |
| **3** | **System Analysis** | | |  |
| 3.1 | Identification of need | |  |
| 3.2 | Functional Requirement | |  |
| 3.3 | Modules | |  |
| 3.4 | Non-functional Requirement | |  |
| 3.5 | User Characteristics | |  |
|  | | | | |
| **4** | **UML Diagrams** | | |  |
| 4.1 | Use case Diagram | |  |
| 4.2 | Activity Diagram | |  |
|  | | | | |
| **5** | **System Design** | | |  |
| 5.1 | Database Design | |  |
| 5.1.1 | Data Dictionary |  |
|  | | | | |
| **6** | **System Implementation** | | |  |
| 6.1 | User Interface Design | |  |
|  | | | | |
| **7** | **Testing** | | |  |
| 7.1 | Test Cases | |  |
|  | 7.2 | Validation Screenshots | |  |
|  | | | | |
| **8** | **Future Enhancement** | | |  |
|  |  | | |  |
| **9** | **References** | | |  |

**Chapter 1: Introduction**

**1.1 Project Definition:**

This project is aimed at developing an online application for the Training and Placement Cell of the University. The 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 T&P Cell of the college to manage the student information with regards to placement. Students logging in should be able to upload their personal and academic details. Company representatives and T&P Committee Member logging in may access/search any information put up by Students. T&P Committee Member should be able to interact with Students for Pre/In-Placement activities and should be able to interact with Company Representative for the same. Also, the T&P Committee Member should be able to manage the interaction between Company representatives and Students. Company Representative logging in should be able to view Student profiles and request candidates as per there requirements.

**1.2 Purpose**

For training and placement of the Student in university, T&P Committee members have to collect the information and CVs of Students and recruiting companies and manage them manually and arranges them according to various streams. If any modification is required that is to be also done manually. So, to reduce the manual procedures required to manage CVs and the information of various recruiters, a new system is proposed which will be helpful to perform these tasks digitally.

**1.3 Scope**

Student and Company Information shall be stored. Marks and attendance of Students regarding various activities shall be stored. Resume and various informatics and statistical reports such related to Companies, Students and T&P cell shall be generated. Companies and Student profiles shall be categorized and departmentalized as per various streams and requirements. Event notifications for various activities shall be sent to Students and Companies and T&P Committee Members.

**Chapter 2: Software Development Platform**

**2.1 Technology Used**

|  |  |
| --- | --- |
| Type | Web Based Application |
| Front-End | HTML5, CSS3, BOOTSTRAP 4, JavaScript, jQuery |
| Back-End | Core PHP [phpMyAdmin], MySQL, PDO |
| Operating System | Windows 10 |
| Tools | Umlet (for diagrams), MS PowerPoint, MS Word, SubmileText3, Amp |

**2.1 Technology Description**

* **Core Php**
* **PHP** is a general-purpose scripting language that is especially suited to server-side web development, in which case **PHP** generally runs on a web server.
* **Html**
  + - **HTML** is used to create electronic documents (called pages) that are displayed on the World Wide Web.
* **jQuery**
* **jQuery** is a concise and fast JavaScript library that can be used to simplify event handling, HTML document traversing, Ajax interactions and animation for speedy website development.
* **Bootstrap**
* **Bootstrap** is a free and open source front end development framework for the creation of websites and web apps. The **Bootstrap** framework is built on HTML, CSS, and JavaScript (JS) to facilitate the development of responsive, mobile-first sites and apps.
* **JavaScript**
* **JavaScript** is a dynamic computer programming language. It is lightweight and most commonly used as a part of web pages, whose implementations allow client-side script to interact with the user and make dynamic pages. It is an interpreted programming language with object-oriented capabilities.

**Chapter 3: System Analysis**

**3.1 Identification of need:**

This system needs to be developed due to below stated reasons

* The T&P Cell Members were overloaded by the manual work for the activities of Pre/In Placement phases.
* To managing Recruiting Companies and the applying Students for the same.
* To automatically generate various reports and maintaining records.
* To reduce manpower required for managing activities and events for the same.
* To reduce manual complexity of the procedures and protocols being followed.

**3.2 Functional Requirement:**

* **Login**

-Registered user will be able to Login to the system.

**Assumptions:**

-User is already registered to the system.

-User is known to their credentials required for login.

**Details Accepted:**

-User’s email address or Password as username.

-User’s confidential password.

**Process:**

-User enters their all valid & required credentials.

-Credential data check with the system database.

-If given credentials are incorrect then message of “Invalid Credentials” is to be displayed.

-If given credentials are correct then get user’s unique id, user role and check user validation status.

-If user status is ‘Pending’ or ‘Rejected’ redirect to service page with a respective message.

-If user status is ‘Validated’ redirect to user’s respective dashboard based on user role.

* **Logout**

-User logged in to the system will be able to logout at will.

**Assumptions:**

-User is already logged into the system

**Process:**

-Destroy the current session and redirect to the index page of the system.

* **Forgot Password**

-User will be able to reset password without logging in to the system via OTP received on email.

**Assumptions:**

-User is already register to the system and has lost/forget their password.

-User knows their username i.e. either email address or phone number.

**Details Accepted:**

-One Time Password or OTP.

-New Password to be set as the Password.

-Confirm New Password to verify with the New Password.

**Process:**

-User enters their username i.e. either email address or phone number.

-A One Time Password or OTP is sent to the user’s email address and redirect to OTP-Verification page.

-User enter the OTP on the OTP-Verification page.

-The entered OTP is matched with the sent OTP (in mail) stored in session.

-If the entered OTP does not match with the sent OTP then show message of “Invalid OTP”.

- If the entered OTP does match with the sent OTP then redirect to New Password Page.

-On New Password page user enters new password and confirms the new password by re-typing it.

-If the New Password matches the Confirm password then update user’s password in the system database.

-If the New Password does not match the Confirm password then show message of “Passwords Do Not Match”.

* **Change Password**

-User will be able to update/change their password.

**Assumptions:**

-User knows their current confidential Password.

**Details Accepted:**

-User’s current password.

-New Password to be set as new password.

-Confirm New Password to verify with the New Password.

**Process:**

-User enters all valid & required details.

-Current password is checked with the system database.

-If Current Password does not match with the database then show message “Invalid Current Password”.

-If Current Password does match with the database then check New Password with Confirm Password.

-If New Password does not match with Confirm Password then show message “Passwords do not match”.

-If New Password does match with Confirm Password then update New Password in the system database and redirect to dashboard.

* **Register**

-A person will be able to register to the system by providing valid details.

**Assumptions:**

-Person registering to the system is a Student or Faculty of Uka Tarsadia University or a person representing a Company registering as a company.

**Details Accepted:**

-Details accepted for registration will be different for Student, Faculty and Company accepted through a set-up wizard before which a One Time Password is accepted.

-For Student:

First Name, Last Name, Enrollment Number, Email Address, Contact Number, Date of Birth, Gender, Parent’s Name, Parent’s Email Address, Parents Contact Number, Profile Pic, About, Degree and Year of Passing the degree.

-For Faculty:

First Name, Last Name, Email Address, Contact Number, Gender, Profile Pic, About.

-For Company:

Name, Registered Year, Email Address, Contact Number, Alternate Contact Number (HR), Head Office Address, Website, Logo, Technologies, No of Current Employees, Maximum Package, Minimum Package, HR’s Full Name, HR’s Email Address.

**Process:**

-A person enter initial registration details i.e. Email address, Password and Password.

-If Password and confirm password does not match then show message of “Passwords do not match”.

- If Password and confirm password not match then send One Time Password or OTP their email for email verification and redirect to their OTP-Verification page.

-User enters the OTP on the OTP-Verification Page.

-The entered OTP is matched with the sent OTP (in mail) stored in session.

-If the entered OTP does not match with the sent OTP then show message of “Invalid OTP”.

- If the entered OTP does match with the sent OTP then redirect to their respective setup wizard based on their user type.

-User enters all valid and required details in the setup wizard.

-A 16-character salt is generated and encrypted password is generated using SHA1 hash function and combination of password and the generated salt.

-User enter data into the setup wizard, encrypted password and salt value are stored into the system database.

-If user type is Student the redirect to student dashboard.

-If user type is Faculty or Company then redirect to service page with message “You are not verified please contact the Committee Head for more details” and set notification to Committee Member to validate newly registered user.

* **Validate User**

-Committee Head will be able to validate a Faculty or Company to use all functionalities.

**Assumptions:**

-Committee Head has externally (Outside the System) verified the user as faculty of Uka Tarsadia University or Company willing to provide training or placement based on their respective criteria.

**Process:**

-Committee Head will receive notification in their dashboard after a user is registered as Company or Faculty.

-After external (Outside the System) verification Committee Head will choose to validate user to use full functionality of the system or deny/reject the user.

-If user is validated then update validation status of the user in the system database as ‘Validated’ (default ‘Pending’).

-If user is rejected then update validation status of the user in the system database as ‘Rejected’.

* **View User Profile**

-A user will be able to view other users’ profile.

* **Update Profile**

-A user will be able to update their profile.

**Details Accepted:**

-Details accepted for registration will be different for Student, Faculty and Company accepted through a set-up wizard before which a One Time Password is accepted.

-For Student:

First Name, Last Name, Enrollment Number, Email Address, Contact Number, Date of Birth, Gender, Parent’s First Name, Parent’s Last Name, Parent’s Email Address, Parents Contact Number, Profile Pic, About, Degree and Year of Passing the degree.

-For Faculty:

First Name, Last Name, Email Address, Contact Number, Gender, Profile Pic, About.

-For Company:

Name, Registered Year, Email Address, Contact Number, Alternate Contact Number (HR), Head Office Address, Website, Logo, Technologies, No of Current Employees, Maximum Package, Minimum Package, HR’s Full Name, HR’s Email Address.

**Process:**

-User enters all valid & required details.

-Entered data is updated into the system database and user is redirected to user’s respected dashboard.

* **View Events**

-A Committee member will be able to view upcoming, past and canceled events.

* **Create Events**

-Committee Member will be able to generate events.

**Details Accepted:**

-Event Title

-Date of the event

-Time of the event

-Venue of the event

-Department

-Batch

-Type of the event

- Nature of event

- Category of event

-Description

**Process:**

-Committee Member enter all valid and required details.

-Enter details are stored into the system database.

-Set notification for the event generated to all the targeted users.

* **Cancel Events**

-A Committee member will be able to cancel an event.

**Assumptions:**

-There is an upcoming event already generated to be canceled.

* **Restore Events**

-A Committee member will be able to restore an event.

**Assumptions:**

-There is an event already canceled to be restored.

- The date of event to be restored is in the future.

* **Apply for Events**

-A user will be able to apply/deny to participate in an upcoming event.

**Assumptions:**

-There is an upcoming event already generated to be applied for or to be rejected.

**Process:**

-If user applies for event then insert application record for the event in the system database.

- If user applies for event then remove notification for the event.

* **Event Notification**

-System will generate notification for all users about upcoming events and the events they have applied for.

**Assumption:**

-An event is generated or an event is coming within 48 hours for which the user has applied for.

* **Attendance**

-Committee Member will be able to fill attendance of students for each event.

**Assumption:**

-There is an event is generated and has been already happened.

**Details:**

-Student’s Id

- Student’s Batch

- Event Id

-Student’s Attendance(‘Present’/’Absent’);

**Process:**

-From the list of participants (Student) can select absent participants

-The attendance is stored into the system database.

* **Create Test**

-Committee Member will be able to generate test.

**Assumption:**

-There is an event is generated with type ‘Test’.

**Details Accepted:**

-Id of the event

-Test’s title

-Batch

-Total marks

-Passing marks

**Process:**

-Committee Member enters all valid & required details.

-Entered details are stored into the system database.

* **View Test**

-Committee Member will be able to view upcoming/past test.

**Assumption:**

-There is a test already generated.

* **Fill Marks**

-Committee Member will be able to fill marks for a test taken.

**Assumption:**

-There is an event’s test is generated and has already been taken.

**Details Accepted:**

-Student’s Id

-Student’s Marks obtained

- Batch

**Process:**

-Committee Member selects a test for which marks are to be filled.

-Committee Member enters all valid & required details,

-Entered details are stored into the system database.

* **Update Resume**

-Student will be able to update their resume.

**Assumption:**

-Student’s personal details are up-to date.

**Details:**

-Technical skills

- Professional skills

- Languages known

- Projects

- Achievements

- Carrier objectives

Process:

-Student enters all valid & required details.

-Entered data are stored in the system database.

* **Generate Resume**

-System will be able to generate student resume with the details provide by student.

**Assumption:**

-Resume details are already updated.

* **View Resume**

-Committee Member and Company will be able to view/download student resume through user profile.

* **Upload Resume**

-Student will be able to upload Resume documents.

**Assumption:**

-Student has a PDF document of their resume or has generated and downloaded their resume using the system.

* **Upload Materials**

-Committee Member and Company will be able to upload materials for In/Pre Placement activities.

**Assumption:**

-Materials to be uploaded are related to preparation and practice for In/Pre placement activities and are PDF documents.

**Details:**

- Batch

- Material type

* **View Materials**

-Students will be able to view/download materials for In/Pre Placement activities.

* **View Students**

- Committee Member will be able to view Students.

* **View Deactivated Students**

- Committee Head will be able to view Students.

* **Deactivate Student**

-Committee Head will be able to deactivate a student.

**Assumption:**

-Given student is already registered to the system.

* **Restore Student**

-Committee Head will be able to restore a deactivated student.

**Assumption:**

**-**Given student is already deactivated.

* **View Faculties/Committee Member**

- Committee Head will be able to view Faculties.

* **View Deactivated Faculties/Committee Member**

- Committee Head will be able to view deactivated Faculties.

* **Deactivate Faculties/Committee Member**

-Committee Head will be able to deactivate a faculty.

**Assumption:**

-Given faculty is already registered to the system.

* **Restore Faculties/Committee Member**

-Committee Head will be able to restore a deactivated faculty.

**Assumption:**

**-**Given faculty is already deactivated.

* **Create batch**

-Committee Head will be able to create a new batch.

* **View Batch**

-Committee Member will be able to view a batch information.

* **Search Student**

-Committee Member will be able to search Students.

* **Shortlist Students**

-Company will be able to create and maintain shortlists of students as they want.

**Assumption:**

-Some placement activity(s) have already been conducted where company and students have interacted.

**Process:**

**-** Company will select an event generated for them.

-Company selects students they want to shortlist.

-A list of selected students is stored into the system database.

* **Recommend Students**

-Committee Head will be able to recommend a Student to a Company.

**Assumption:**

-Student being suggested has applied for placement events for the company being suggested to.

**Details Accepted:**

-Student’s Id

-Company’s Id

-Description/Reason for recommendation.

**Process**

-Committee Head enters all valid and required details.

-Entered data is stored into the system database.

-Set notification of recommendation for the respective company.

* **View Company**

-Committee Member will be able to view a companies .

* **View Deactivated Companies**

- Committee Head will be able to view deactivated companies.

* **Deactivate Company**

-Committee Head will be able to deactivate a company.

**Assumption:**

-Given company is already registered to the system.

* **Restore Company**

-Committee Head will be able to restore a deactivated company.

**Assumption:**

**-**Given company is already deactivated.

* **Create Broadcast list**

-Committee head will be able to create and add/remove companies.

* **Send Broadcast**

-Committee head will be able to send a broadcast to companies in a broadcast list.

**Assumptions:**

-A broadcast list is created and atleast one company is added to it.

**Details:**

**-**Broadcast list

**-**Subject

**-**Message

* **View Placement Report**

-Any user will be able to view reports subjected to placement.

* **Generate Reports**

-Committee Head will be able to generate reports of placement, training, events, attendance, and marks.

**Details Accepted:**

-Company/Department/Course/Batch/Test.

-Report Type

**Process:**

-Committee Head enters all valid and required details

-Report is generated and displayed on the dashboard.

-Report PDF is generated and downloaded to the user’s system.

* **Invite Company**

-Committee Head will be able to invite registered companies for up-coming events.

**Assumptions:**

-An event is already generated for which the invitation if to be sent.

**Details Accepted:**

-Companies

-Event Name

-Description of event (Mail Body)

**Process:**

-Committee Head enters all valid and required details.

-Set a notification for all the companies selected for the invitation.

* **Send Invitation Mail**

-Committee Member will be able to send invitation mails to registered as well as non-registered companies for up-coming events.

**Assumptions:**

-An event is already generated for which the invitation if to be sent.

**Details Accepted:**

-Companies

-Email address of companies

-Event Name

-Description of event (Mail Body)

**Process:**

-Committee Head enters all valid and required details.

-Send an email to all the companies selected for invitation.

* **Generate Placement Schedule**

-System will be able to generate placement schedule based.

**Assumptions:**

-There companies registered to the system willing to participate in the placement process.

**Details Accepted:**

-Companies

-Start date of the schedule.

-No of days to be given for each company.

**Process:**

-Committee Head enters all valid & required details.

-Generate the placement schedule based on average package offered (High to Low) and store schedule into the system database.

-Display schedule on the user dashboard.

* **Edit Placement Schedule**

-Committee Member will be able to edit placement schedule or reschedule a company.

**Assumptions:**

-There is a placement schedule already generated and stored into the system database.

**Details Accepted:**

-Companies

-Rescheduled date of the Company.

**Process:**

-Committee Head enters all valid & required details.

-Generate the placement schedule and store schedule into the system database.

-Display schedule on the user dashboard.

* **Download Placement Schedule**

-Committee Member will be able to download placement schedule.

* **Send Training Offer**

-Company will be able to send offer for training to a student.

**Assumptions:**

-The student receiving the offer has applied and attended placement events for the Company offering.

**Details Accepted:**

-Training Title

-Stipend offered

-Description

-Technology

-Offer letter (PDF)

**Process:**

-User enters all valid & required details.

-Offer data is stored into the system database.

-Set notification of the offer to the receiving Student.

* **Validate Training Offer**

-Committee Head will be able to validate training offers before publishing them to students.

**Assumptions:**

-Committee Head has received training offers for students from a company.

* **Accept Training Offer**

-Student will be able to accept/deny a training offer from a company.

**Assumptions:**

-The student has received a training offer from a company.

**Process:**

-If Student accepts the offer then store training data into the system database.

-If Student denies the offer then drop the offer notification.

* **Send Placement Offer**

- Company will be able to send offer for placement to a student.

**Assumptions:**

-The student receiving the offer is already under training at the company.

**Details Accepted:**

-Job Title

-Package offered

-Description

-Technology

-Offer letter (PDF)

**Process:**

-User enters all valid & required details.

-Offer data is stored into the system database.

-Set notification of the offer to the receiving Student.

* **Validate Placement Offer**

-Committee Head will be able to validate Placement offers before publishing them to students.

**Assumptions:**

-Committee Head has received placement offers for students from a company.

* **Accept Placement Offer**

-Student will be able to accept/deny a placement offer from a company.

**Assumptions:**

-The student has received a Job offer from a company.

**Process:**

-If Student accepts the offer then store Placement data into the system database.

-If Student denies the offer then drop the offer notification.

**3.3 Modules:**

* **Module 1: Login**

A registered user can login to the system by providing their login credentials that are their email address or phone number and password. A user can reset or update password through forget password. After user is satisfied using the system, they can logout from the system.

* **Module 2: Registration**

Any student, faculty of Computer Science department of Uka Tarsadia University and Company Representative (as a Company) can register to the system by providing details required in their respective wizard after providing active email address and their desired password. Any user entity (other than student) registered to the system is to be validated by the user outside the system once the user is validated the user will be accepted by the Committee Head to use full functionality of the System.

* **Module 3: Manage Profile**

Any user entity will be able to update their profile as in personal details and professional details and will also be able to upload documents if required. Company will be able to view student and committee member profile; Student will be able view company profile and Committee member will be able to view student profile and company profile.

* **Module 4: Events**

Any user entity will be able to view events generated by a committee member and will also be able to apply or deny for the event. An event can be a seminar, meeting, test, etc. A test event will have a test generated and for any given test marks can be updated by a committee member. All the users will get a notification when an event is generated, test is generated and marks are updated. Committee Member will also be able to fill attendance for any event

* **Module 5: Resume**

Student will be able to update professional and academic details and generate a resume with same details and Committee Members and Company will be able to users will be able to view Students resume through their profile.

* **Module 6: Materials**

Committee Member and company will be able to upload and remove materials for pre/in placement activities and Students will be able to view download these materials.

* **Module 7: Shortlist**

Company will be able search students and maintain a shortlist of students they want for training/placement. Committee head will be able to recommend a student to a company.

* **Module 8: Reports**

Any user entity will be able to view graphical and statistical reports subjected to placement. Committee member will be able to generate and download reports subjected to placement, events, tests, attendance and marks.

* **Module 9: Invitation**

Committee Head will be able to invite a Company whether registered to the system. A registered company will receive invitation as notification in the system as well as via email.

* **Module 10: Broadcast**

Committee Head will be able to create broadcast lists and add or remove companies to broadcast lists. A broadcasted message will be sent to the companies’ respective email address

* **Module 10: Placement Schedule**

Committee Head will be able to select which companies to include in the placement schedule and generate the same and will also be able to reschedule any company.

* **Module 11: Training and Placement Acceptance**

Company will be able to publish offers for Training and Placement provided by the companies after reviewing them. Once Published Student will be able to accept-deny the offer as per their desire.

* **Module 12: Manage Users**

Committee Head will be able to upgrade user role of Faculty to Committee Member and degrade Committee Member to Faculty. Committee Head will also be able to give up role of ‘Committee Head’ to another Faculty or Committee Member. Also, Committee Head will be able to activate and deactivate all the users individually.

**3.4 Non-Functional Requirement:**

* **Security**
* The Passwords in the system will be protected by the SHA1 hash function applied on to the password concatenated with a salt value unique for each individual user.
* The system will be protected by a Secure Socket Layer or SSL to run over https and provide security to sensitive information.
* **Responsive**
* The system will be designed to work on any modern era digital device with browser compatibility irrespective of its display size, resolution, orientation and aspect ratio.
* **Accessibility**
* The system will be accessible around the globe on modern era digital device with an active internet connection.
* **Performance**
* The system will be designed with performance in mind where database procedure will be used reduce data access time and processing load on the web server.
* **Portability**
* The system will be portable on the bases of database by using PDO for database connectivity.

**3.5 User Characteristics**

**Users:**

* Committee Head

- Login

- Forgot Password

- Change Password

- Manage Profile

- Create Events

- View Resume

- View Placement Reports

- Generate Reports

- Send Invitation

- Edit Placement Schedule

- Generate Placement Schedule

- Manage Faculty

- Manage Test

- Fill Marks

- Fill Attendance

- Recommend Student

* Committee Member

- Login

- Forgot Password

- Change Password

- Manage Profile

- Create Events

- View Resume

- View Placement Reports

- Manage Materials

- Generate Placement Reports

- Manage Test

- Fill Marks

- Fill Attendance

* Faculty

- Login

- Forgot Password

- Change Password

- Manage Profile

- View Placement Reports

- View Resume

* Company

- Login

- Forgot Password

- Change Password

- Manage Profile

- View Placement Reports

- View Resume

- Manage Shortlist

- Manage Materials

- Send Training/Placement Offer

* Student

- Login

- Forgot Password

- Change Password

- Manage Profile

- Apply/Deny Events

- Update Resume

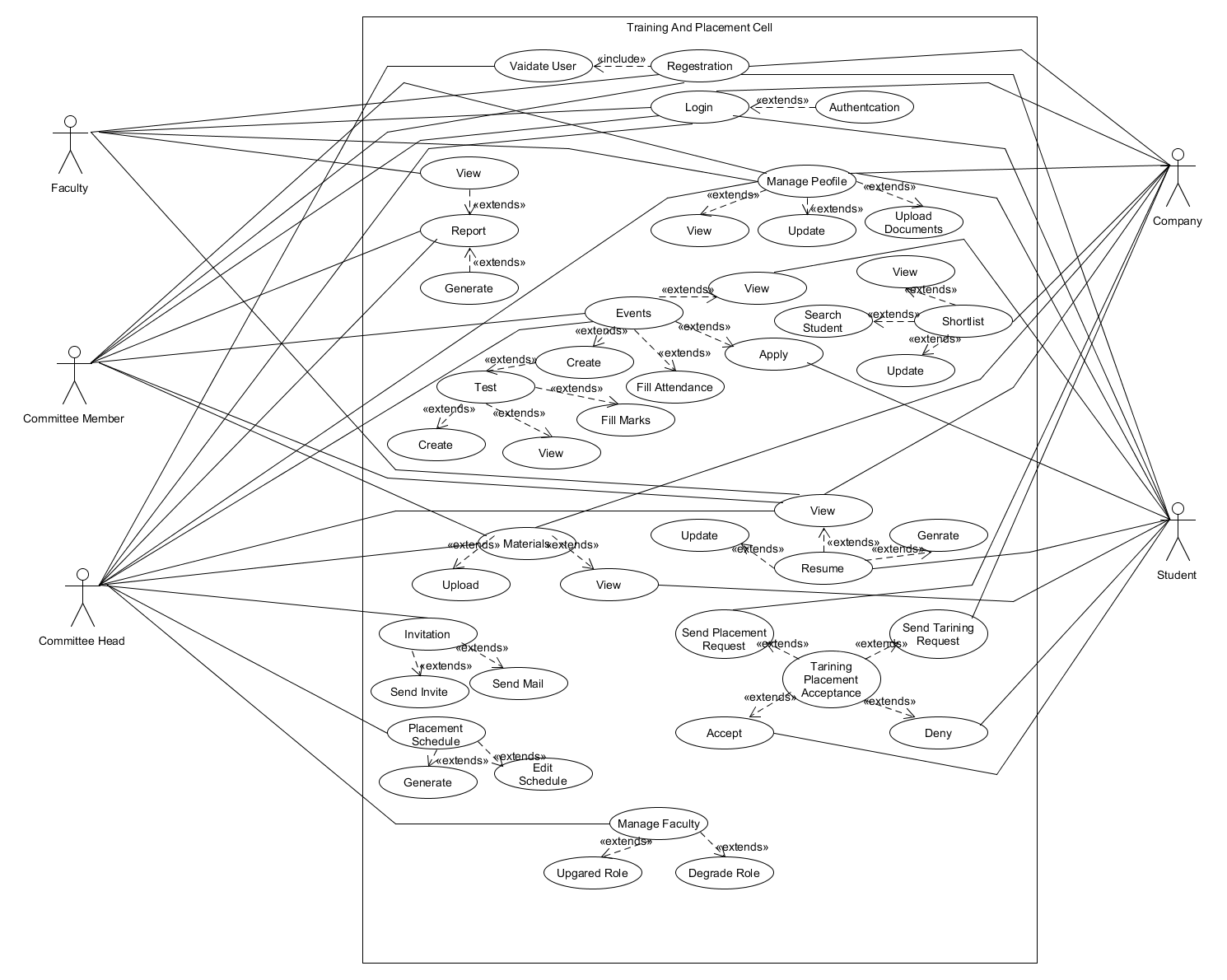
- Generate Resume

- View/Download Materials

- Accept/Deny Training/Placement Offer

**Chapter 4: UML Diagrams**

**4.1 Use Case Diagram**

****

**4.2 Activity Diagrams**

* **Registration**
* **Login**
* **Manage Events**
* **Manage Test**
* **Materials**
* **Manage Profile**
* **Resume**
* **Invitation**
* **Placement Schedule**
* **Shortlist**
* **Training and Placement Acceptance**
* **Reports**

**Chapter 5: System Design**

**5.1 Data Dictionary**

1**.TABLE NAME: tbl\_academic\_10TH\_details**

|  |  |  |  |
| --- | --- | --- | --- |
| FIELD | DATATYPE | CONSTRAINTS | DESCRIPTION |
| ACADEMIC\_10TH\_ID | int (11) | Primary Key | Table's Unique Id |
| ACADEMIC\_10TH\_STUDENT\_ID | int (11) | Foreign Key | Reference to the primary key of the student |
| ACADEMIC\_10TH\_BOARD | varchar (255) | Not Null | Student’s 10th Board |
| ACADEMIC\_10TH\_SCHOOL | varchar (255) | Not Null | Student’s 10th School |
| ACADEMIC\_10TH\_PERCENTAGE | float | Not Null | Student’s 10th percentage |

**2.TABLE NAME: tbl\_academic\_12TH\_details**

|  |  |  |  |
| --- | --- | --- | --- |
| FIELD | DATATYPE | CONSTRAINTS | DESCRIPTION |
| ACADEMIC\_12TH\_ID | int (11) | Primary Key | Table's Unique Id |
| ACADEMIC\_12TH\_STUDENT\_ID | int (11) | Foreign Key | Reference to the primary key of the student |
| ACADEMIC\_12TH\_BOARD | varchar (255) |  | Student’s 10th Board |
| ACADEMIC\_12TH\_SCHOOL | varchar (255) |  | Student’s 10th School |
| ACADEMIC\_12TH\_STREAM | varchar (20) |  |  |
| ACADEMIC\_12TH\_PERCENTAGE | float |  | Student’s 10th percentage |

|  |  |  |  |
| --- | --- | --- | --- |
| FIELD | DATATYPE | CONSTRAINTS | DESCRIPTION |
| ACADEMIC\_BACHELOR\_ID | int (11) | Primary Key | Table's Unique Id |
| ACADEMIC\_ BACHELOR \_STUDENT\_ID | int (11) | Foreign Key | Reference to the primary key of the student |
| ACADEMIC\_ BACHELOR \_UNIVERSITY | varchar (255) | Not Null |  |
| ACADEMIC\_ BACHELOR \_INSTITUTE | varchar (255) | Not Null |  |
| ACADEMIC\_ BACHELOR \_SPECIALIZATION | varchar (255) | Not Null |  |
| ACADEMIC\_ BACHELOR \_DEGREE | varchar (255) | Not Null |  |
| ACADEMIC\_ BACHELOR \_SEM\_1 | float | Not Null |  |
| ACADEMIC\_ BACHELOR \_SEM\_2 | float | Not Null |  |
| ACADEMIC\_ BACHELOR \_SEM\_3 | float | Not Null |  |
| ACADEMIC\_ BACHELOR \_SEM\_4 | float | Not Null |  |
| ACADEMIC\_ BACHELOR \_SEM\_5 | float | Not Null |  |
| ACADEMIC\_ BACHELOR \_SEM\_6 | float | Not Null |  |
| ACADEMIC\_ BACHELOR \_SEM\_7 | float | Not Null |  |
| ACADEMIC\_ BACHELOR \_SEM\_8 | float | Not Null |  |

**3.TABLE NAME: tbl\_academic\_bachelor\_details**

**4.TABLE NAME: tbl\_academic\_details**

|  |  |  |  |
| --- | --- | --- | --- |
| FIELD | DATATYPE | CONSTRAINTS | DESCRIPTION |
| ACADEMIC\_DETAILS\_ID | int (11) | Primary Key | Table's Unique Id |
| ACADEMIC\_ DETAILS \_STUDENT\_ID | int (11) | Foreign Key | Reference to the primary key of the student |
| ACADEMIC\_ DETAILS \_10TH\_ID | varchar (255) | Foreign Key | Reference to the primary key of the 10th details |
| ACADEMIC\_ DETAILS \_12TH\_ID | varchar (255) | Foreign Key | Reference to the primary key of the 12th details |
| ACADEMIC\_ DETAILS \_DIPLOMA\_ID |  | Foreign Key | Reference to the primary key of the diploma details |
| ACADEMIC\_ DETAILS \_BACHELOR\_ID | varchar (20) | Foreign Key | Reference to the primary key of the bachelor details |
| ACADEMIC\_ DETAILS \_MASTER\_ID | float | Foreign Key | Reference to the primary key of the master details |

**5.TABLE NAME: tbl\_academic\_diploma\_details**

|  |  |  |  |
| --- | --- | --- | --- |
| FIELD | DATATYPE | CONSTRAINTS | DESCRIPTION |
| ACADEMIC\_DIPLOMA\_ID | int (11) | Primary Key | Table's Unique Id |
| ACADEMIC\_ DIPLOMA \_STUDENT\_ID | int (11) | Foreign Key | Reference to the primary key of the student |
| ACADEMIC\_ DIPLOMA \_UNIVERSITY | varchar (255) | Not Null |  |
| ACADEMIC\_ DIPLOMA \_INSTITUTE | varchar (255) | Not Null |  |
| ACADEMIC\_ DIPLOMA \_SPECIALIZATION | varchar (255) | Not Null |  |
| ACADEMIC\_ BACHELOR \_SEM\_1 | float | Not Null |  |
| ACADEMIC\_ BACHELOR \_SEM\_2 | float | Not Null |  |
| ACADEMIC\_ BACHELOR \_SEM\_3 | float | Not Null |  |
| ACADEMIC\_ BACHELOR \_SEM\_4 | float | Not Null |  |
| ACADEMIC\_ BACHELOR \_SEM\_5 | float | Not Null |  |
| ACADEMIC\_ BACHELOR \_SEM\_6 | float | Not Null |  |

**6.TABLE NAME: tbl\_academic\_master\_details**

|  |  |  |  |
| --- | --- | --- | --- |
| FIELD | DATATYPE | CONSTRAINTS | DESCRIPTION |
| ACADEMIC\_MASTER\_ID | int (11) | Primary Key | Table's Unique Id |
| ACADEMIC\_ MASTER \_STUDENT\_ID | int (11) | Foreign Key | Reference to the primary key of the student |
| ACADEMIC\_ MASTER \_UNIVERSITY | varchar (255) | Not Null |  |
| ACADEMIC\_ MASTER \_INSTITUTE | varchar (255) | Not Null |  |
| ACADEMIC\_ MASTER \_SPECIALIZATION | varchar (255) | Not Null |  |
| ACADEMIC\_ MASTER \_SEM\_1 | float | Not Null |  |
| ACADEMIC\_ MASTER \_SEM\_2 | float | Not Null |  |
| ACADEMIC\_ MASTER \_SEM\_3 | float | Not Null |  |
| ACADEMIC\_ MASTER \_SEM\_4 | float | Not Null |  |

**7.TABLE NAME: tbl\_attendance**

|  |  |  |  |
| --- | --- | --- | --- |
| FIELD | DATATYPE | CONSTRAINTS | DESCRIPTION |
| ATTENDANCE\_ID | int (11) | Primary Key | Table's Unique Id |
| ATTENDANCE\_EVENT\_ID | int (11) | Foreign Key | References to Events primary key |
| ATTENDANCE\_STUDENT\_ID | int (11) | Foreign Key | References to Students primary key |
| ATTENDANCE\_FACULTY\_ID |  | Foreign Key | References to Faculties primary key who is Filling  the attendance |
| ATTENDANCE | tinyint (1) | Not Null | Attendance as 1(Present) and 0(Absent) |

**8. Table Name: tbl\_batch**

|  |  |  |  |
| --- | --- | --- | --- |
| FIELD | DATATYPE | CONSTRAINTS | DESCRIPTION |
| BATCH\_ID | int (11) | Primary Key | Table's Unique Id |
| BATCH\_DEPARTMENT | varchar (20) | Not Null | Name of Department |
| BATCH\_DEGREE | varchar (10) | Not Null | Title of Course/Degree |
| BATCH\_PASSING\_YEAR | char (4) | Not Null | Year Of Admission |
| BATCH\_SEMESTER | int (11) | Not Null |  |
| BATCH\_D2D |  | Not Null |  |
| BATCH\_TYPE |  | Not Null |  |

**9. Table Name: tbl\_broadcast\_list**

|  |  |  |  |
| --- | --- | --- | --- |
| FIELD | DATATYPE | CONSTRAINTS | DESCRIPTION |
| BATCH\_PASSING\_YEAR | int (11) | Not Null |  |
| BROADCAST\_LIST\_NAME | varchar (255) | Not Null |  |
| BROADCAST\_LIST\_CREATED\_BY | int (11) | Not Null |  |
| BROADCAST\_LIST\_DATE | date | Not Null |  |
| BROADCAST\_LIST\_STATUS | tinyint (11) | Not Null |  |

**10. Table Name: tbl\_broadcast\_list\_members**

|  |  |  |  |
| --- | --- | --- | --- |
| FIELD | DATATYPE | CONSTRAINTS | DESCRIPTION |
| BROADCAST\_LIST\_MEMBER\_ID | int (11) | Primary Key | Table's Unique Id |
| BROADCAST\_LIST\_MEMBER\_BROADCAST\_ID | int (11) | Foreign Key | Reference to the primary key of the broadcast |
| BROADCAST\_LIST\_MEMBER\_COMPANY\_ID | int (11) | Foreign Key | Reference to the primary key of the company |
| BROADCAST\_LIST\_MEMBER\_STATUS | tinyint (11) | Not Null |  |

**11. Table Name: tbl\_company**

|  |  |  |  |
| --- | --- | --- | --- |
| FIELD | DATATYPE | CONSTRAINTS | DESCRIPTION |
| COMPANY\_ID | int (11) | Primary Key | Table's Unique Id |
| COMPANY\_NAME | varchar (30) | Not Null | Company's Name |
| COMPANY\_REGISTERED\_YEAR | char (4) | Not Null | Company's Registered Year |
| COMPANY\_ADDRESS | varchar (255) | Not Null | Company's Address |
| COMPANY\_EMAIL | varchar (255) | Unique | Company's Email |
| COMPANY\_PHONE\_NUMBER\_1 | varchar (10) | Unique | Company's Phone Number 1 |
| COMPANY\_PHONE\_NUMBER\_2 | varchar (10) | Unique | Company's HR's Phone Number |
| COMPANY\_WEBSITE | varchar (100) | Unique | Company's Website |
| COMPANY\_ACCEPTED | char (1) | Not Null | Company's Status about system access  as P(Pending), V(Validate) and R(Rejected) |
| COMPANY\_PASSWORD | varchar (40) | Not Null | SHA1 Encrypted User Password |
| COMPANY\_LOGO | varchar (50) | Not Null | Company's Logo |
| COMPANY\_ABOUT | varchar (255) | Not Null | Company's About |
| COMPANY\_TECHNOLOGIES | varchar (255) | Not Null | Company's Technologies |
| COMPANY\_NO\_OF\_EMPLOYEES | int (11) | Not Null | Company's No Of Employees |
| COMPANY\_MAXIMUM\_PACKAGE | bigint (20) | Not Null | Company's Maximum Package |
| COMPANY\_MINIMUM\_PACKAGE | bigint (20) | Not Null | Company's Minimum Package |
| COMPANY\_HR\_NAME | varchar (50) | Not Null | Company's Hr Name |
| COMPANY\_HR\_EMAIL | varchar (255) | Unique | Company's Hr Email |

|  |  |  |  |
| --- | --- | --- | --- |
| FIELD | DATATYPE | CONSTRAINTS | DESCRIPTION |
| COMPANY\_DOCUMENT\_ID | int (11) | Primary Key | Table's Unique Id |
| COMPANY\_DOCUMENT\_STUDENT\_ID | int (11) | Foreign Key | Reference to the primary key of the student |
| COMPANY\_DOCUMENT\_COMPANY\_ID | int (11) | Foreign Key | Reference to the primary key of the company |
| COMPANY\_DOCUMENT\_TYPE | varchar (10) | Not Null | Document's Type |
| COMPANY\_DOCUMENT\_NAME | varchar (50) | Not Null | Name of the document file stored as |

**12. Table Name: tbl\_company\_document**

|  |  |  |  |
| --- | --- | --- | --- |
| FIELD | DATATYPE | CONSTRAINTS | DESCRIPTION |
| DOCUMENT\_ID | int (11) | Primary Key | Table's Unique Id |
| DOCUMENT \_UPLOADER\_ID | int (11) | Foreign Key | Reference to the primary key of the student |
| DOCUMENT \_UPLOADER\_TYPE | char (2) | Not Null | Reference to the primary key of the company |
| DOCUMENT\_TYPE | char (2) | Not Null | Document's Type |
| DOCUMENT\_DEPARTMENT | varchar (10) | Not Null | Name of the document file stored as |
| DOCUMENT\_DEGREE | varchar (20) | Not Null |  |
| DOCUMENT\_TITLE | varchar (255) | Not Null |  |
| DOCUMENT\_NAME | varchar (255) | Not Null |  |

**13. Table Name: tbl\_documents**

|  |  |  |  |
| --- | --- | --- | --- |
| FIELD | DATATYPE | CONSTRAINTS | DESCRIPTION |
| EVENT\_ID | int (11) | Primary Key | Table's Unique Id |
| EVENT\_COMPANY\_ID | int (11) | Foreign Key | Reference to the primary key of the company |
| EVENT\_GENRATED\_BY | int (11) | Not Null | Reference to the Committee Member/Head's primary  key who generated the event |
| EVENT\_NAME | varchar (20) | Not Null | Event's Title |
| EVENT\_DESCRIPTION | varchar (255) | Not Null | Event's Description |
| EVENT\_VENUE | varchar (255) | Not Null | Event's Venue |
| EVENT\_DATE | date | Not Null | Event's Date |
| EVENT\_BATCH\_ID | char (4) | Foreign Key | Event Batch’s Id |
| EVENT\_TIME | varchar (5) | Not Null | Event's Time |
| EVENT\_TYPE | varchar (255) | Not Null | Event's Type |
| EVENT\_CATEGORY | tinyint (1) | Not Null |  |
| EVENT\_CANCELED | tinyint (1) | Not Null | Event's status whether the event is cancelled as 1(Cancelled) or  0(Not Cancelled) |

**14. Table Name: tbl\_event**

**15. Table Name: tbl\_event\_application**

|  |  |  |  |
| --- | --- | --- | --- |
| FIELD | DATATYPE | CONSTRAINTS | DESCRIPTION |
| EVENT\_APPLICATION\_ID | int (11) | Primary Key | Table's Unique Id |
| EVENT\_APPLICATION\_STUDENT\_ID | int (11) | Foreign Key | Reference to the Student's primary  key who applied for the event |
| EVENT\_APPLICATION\_EVENT\_ID | int (11) | Foreign Key | Reference to he Event's primary  key for which the student has applied for |

**16. Table Name: tbl\_faculty**

|  |  |  |  |
| --- | --- | --- | --- |
| FIELD | DATATYPE | CONSTRAINTS | DESCRIPTION |
| FACULTY\_ID | int (11) | Primary Key | Table's Unique Id |
| FACULTY\_FIRST\_NAME | varchar (20) | Not Null | Faculty's First Name |
| FACULTY\_LAST\_NAME | varchar (20) | Not Null | Faculty's Last Name |
| FACULTY\_GENDER | char (1) | Not Null | Faculty's Gender |
| FACULTY\_EMAIL | varchar (255) | Unique | Faculty's Email |
| FACULTY\_PHONE\_NUMBER | varchar (10) | Unique | Faculty's Phone Number |
| FACULTY\_ABOUT | varchar (255) | Not Null | Faculty's About |
| FACULTY\_PROFILE\_PIC | varchar (255) | Not Null | Faculty's Profile Pic |
| FACULTY\_ACCEPTED | char (1) | Not Null | Faculty's Status about system access as P(Pending), V(Validate) and R(Rejected) |
| FACULTY\_PASSWORD | varchar (40) | Not Null | SHA1 Encrypted User Password |
| FACULTY\_ROLE | char (2) | Not Null | Faculty's Role |

|  |  |  |  |
| --- | --- | --- | --- |
| FIELD | DATATYPE | CONSTRAINTS | DESCRIPTION |
| LOGIN\_ID | int (11) | Primary Key | Table's Unique Id |
| LOGIN\_REFERENCE\_ID | int (11) | Unique | Reference to the User's Primary key |
| LOGIN\_USER\_TYPE | char (2) | Not Null | Type of User as ST(Student),CP(Company),FC(Faculty),  CM(Committee Member) and  CH(Committee Head) |
| LOGIN\_USER\_EMAIL | varchar (255) | Unique | Email Address of the User |
| LOGIN\_USER\_PHONE\_NUMBER | varchar (255) | Unique | Phone Number Address of the User |
| LOGIN\_USER\_PASSWORD | varchar (40) | Not Null | SHA1 Encrypted User Password |
| LOGIN\_USER\_SALT | varchar (16) | Not Null | User's Salt Value |

**17.Table Name: tbl\_login**

**18.Table Name: tbl\_marks**

|  |  |  |  |
| --- | --- | --- | --- |
| FIELD | DATATYPE | CONSTRAINTS | DESCRIPTION |
| MARKS\_ID | int (11) | Primary Key | Table's Unique Id |
| MARKS\_TEST\_ID | int (11) | Foreign Key | Reference to the primary key of the Test marks filled for |
| MARKS\_STUDENT\_ID | int (11) | Foreign Key | Reference to the primary key of the Student whose  marks are filled |
| MARKS\_OBTAINED | int (11) | Not Null | Marks Obtained by the student |

|  |  |  |  |
| --- | --- | --- | --- |
| FIELD | DATATYPE | CONSTRAINTS | DESCRIPTION |
| NOTIFICATION\_ID | int (11) | Primary Key | Table's Unique Id |
| NOTIFICATION\_SENDER\_ID | int (11) | Not Null | Reference to the primary key of the sender of the  notification |
| NOTIFICATION\_RECEIVER\_ID | int (11) | Not Null | Reference to the primary key of the receiver of the  notification |
| NOTIFICATION\_SENDER\_TYPE | char (2) | Not Null | Sender's user type |
| NOTIFICATION\_RECIVER\_TYPE | char (2) | Not Null | Receiver’s user type |
| NOTIFICATION\_TIME\_STAMP | timestamp | Not Null | Timestamp of the time of the notification generated |
| NOTIFICATION\_DESCRPTION | varchar (255) | Not Null | Notification's description |
| NOTIFICATION\_REMOVE\_STATUS | tinyint (1) | Not Null | Notification's Status whether the notification is  removed by the receiver as 1(Removed) or 0(Not Removed |

**19.Table Name: tbl\_notification**

**20.Table Name: tbl\_placement**

|  |  |  |  |
| --- | --- | --- | --- |
| FIELD | DATATYPE | CONSTRAINTS | DESCRIPTION |
| PLACEMENT\_ID | int (11) | Primary Key | Table's Unique Id |
| PLACEMENT\_TRAINING\_ID | int (11) | Foreign Key | Reference to the primary key of the Training |
| PLACEMENT\_OFFERED\_PACKAGE | int (11) | Not Null | Offered Annual Package by the company |
| PLACEMENT\_BOND | int (11) | Not Null | Reference to the primary key of the company  document with type 'BOND' |

**21. Table Name: tbl\_placement\_schedule**

|  |  |  |  |
| --- | --- | --- | --- |
| FIELD | DATATYPE | CONSTRAINTS | DESCRIPTION |
| PLACEMENT\_SCHEDULE\_ID | int (11) | Primary Key | Table's Unique Id |
| PLACEMENT\_SCHEDULE\_COMPANY\_ID | int (11) | Foreign Key | Reference to the primary key of the company |
| PLACEMENT\_SCHEDULE\_START\_DATE | Date | Not Null | Date allocated to the company |
| PLACEMENT\_SCHEDULE\_END\_DATE | Date | Not Null | Date allocated to the company |
| PLACEMENT\_SCHEDULE\_STATUS | tinyint (1) | Not Null | Date allocated to the company |

**22.Table Name: tbl\_recommendation**

|  |  |  |  |
| --- | --- | --- | --- |
| FIELD | DATATYPE | CONSTRAINTS | DESCRIPTION |
| RECOMMENDATION\_ID | int (11) | Primary Key | Table's Unique Id |
| RECOMMENDATION\_STUDENT\_ID | int (11) | Foreign Key | Reference to the Student's primary key being  recommended |
| RECOMMENDATION\_COMPANY\_ID | int (11) | Foreign Key | Reference to the Company's primary key to which  Student is being recommended to |
| RECOMMENDATION\_DESCRIPTION | varchar (255) | Not Null | Description or reason for recommending the student |

**23.Table Name: tbl\_resume**

|  |  |  |  |
| --- | --- | --- | --- |
| FIELD | DATATYPE | CONSTRAINTS | DESCRIPTION |
| RESUME\_ID | int (11) | Primary Key | Table's Unique Id |
| RESUME\_STUDENT\_ID | int (11) | Foreign Key | Reference to the Student's primary key |
| RESUME\_TECHNICAL\_SKILLS | varchar (255) | Not Null | List of Technical Skills separated by '#' |
| RESUME\_PERSONAL\_SKILLS | varchar (255) | Not Null | List of Personal Skills separated by '#' |
| RESUME\_LANGUAGES | varchar (255) | Not Null | List of languages known separated by '#' |
| RSEUME\_PROJECTS | varchar (255) | Not Null | List of Projects separated by '#' |
| RESUME\_ACHIVEMENTS | varchar (255) | Not Null | List of Achievements separated by '#' |
| RESUME\_CARRIER\_OBJECTIVE | varchar (255) | Not Null | Description of Carrier objective |

**24.Table Name: tbl\_resume\_document**

|  |  |  |  |
| --- | --- | --- | --- |
| FIELD | DATATYPE | CONSTRAINTS | DESCRIPTION |
| RESUME \_DOCUMENT\_ID | int (11) | Primary Key | Table's Unique Id |
| RESUME \_DOCUMENT\_STUDENT\_ID | Foreign Key | Not Null | Reference to the primary key of the student |
| RESUME \_DOCUMENT\_NAME | varchar (255) | Not Null | Name of the document file stored as |

|  |  |  |  |
| --- | --- | --- | --- |
| FIELD | DATATYPE | CONSTRAINTS | DESCRIPTION |
| SELECTION\_LIST \_ID | int (11) | Primary Key | Table's Unique Id |
| SELECTION\_LIST \_COMPANY\_ID | int (11) | Foreign Key | Reference to the primary key of the company maintain the list |
| SELECTION\_LIST\_YEAR | int (11) | Unique | Reference to the primary key of the student being shortlisted |
| SELECTION\_LIST\_NAME | int (11) | Not Null | Number of round when the student is shortlisted |
| SELECTION\_LIST\_TYPE | char (2) |  |  |

**25.Table Name: tbl\_selection\_list**

**26.Table Name: tbl\_shortlist**

|  |  |  |  |
| --- | --- | --- | --- |
| FIELD | DATATYPE | CONSTRAINTS | DESCRIPTION |
| SHORTLIST\_ID | int (11) | Primary Key | Table's Unique Id |
| SHORTLIST\_RECOMMENDATION\_ID |  | Foreign Key | Reference to the primary key of the recommendation |
| SHORTLIST\_COMPANY\_ID | int (11) | Foreign Key | Reference to the primary key of the company maintain the list |
| SHORTLIST\_STUDENT\_ID | int (11) | Foreign Key | Reference to the primary key of the student being shortlisted |
| SHORTLIST\_TYPE | int (11) | Not Null | Number of round when the student is shortlisted |

**27.Table Name: tbl\_student**

|  |  |  |  |
| --- | --- | --- | --- |
| FIELD | DATATYPE | CONSTRAINTS | DESCRIPTION |
| STUDENT\_ID | int (11) | Primary Key | Table's Unique Id |
| STUDENT\_FIRST\_NAME | varchar (20) | Not Null | Student's Frist Name |
| STUDENT\_LAST\_NAME | varchar (20) | Not Null | Student's Last Name |
| STUDENT\_ENROLLMENT\_NUMBER | varchar (15) | Unique | Student's Enrollment Number |
| STUDENT\_EMAIL | varchar (255) | Unique | Student's Email |
| STUDENT\_PHONE\_NUMBER | varchar (10) | Unique | Student's Phone Number |
| STUDENT\_DATE\_OF\_BIRTH | Date | Not Null | Student's Date Of Birth |
| STUDENT\_GENDER | char (1) | Not Null | Student's Gender |
| STUDENT\_PASSWORD | varchar (40) | Not Null | SHA1 Encrypted User Password |
| STUDENT\_PARENT\_NAME | varchar (20) | Not Null | Student's Parent Name |
| STUDENT\_PARENT\_PHONE\_NUMBER | varchar (10) | Not Null | Student's Parent Phone Number |
| STUDENT\_PARENT\_EMAIL | varchar (255) | Not Null | Student's Parent Email |
| STUDENT\_PROFILE\_PIC | varchar (255) | Not Null | Student's Profile Pic |
| STUDENT\_ABOUT | varchar (255) | Not Null | Student's About |
| STUDENT\_ADDRESS | varchar (255) | Not Null | Student's Address |
| STUDENT\_BATCH\_ID | int (11) | Not Null | Reference to the Batch's primary key |
| STUDENT\_ADMISSION\_YEAR | char (4) | Not Null | Student's Passing Year |
| STUDENT\_STATUS | tinyint (1) |  |  |

**28.Table Name: tbl\_test**

|  |  |  |  |
| --- | --- | --- | --- |
| FIELD | DATATYPE | CONSTRAINTS | DESCRIPTION |
| TEST\_ID | int (11) | Primary Key | Table's Unique Id |
| TEST\_EVENT\_ID | int (11) | Foreign Key | Reference to the primary key of the event for which test is created |
| TEST\_NAME | varchar (20) | Not Null | Test's Title |
| TEST\_DESCRIPTION | varchar (255) | Not Null | Test's Description |
| TEST\_OBJECTIVE | tinyint (1) | Not Null | Test's Objective as 1(In placement) or 0(Preplacement) |
| TEST\_TOTAL\_MARKS | int (11) | Not Null | Test's Totals Marks |
| TEST\_PASSING\_MARKS | int (11) | Not Null | Test's Passing Marks |

**29.Table Name: tbl\_training**

|  |  |  |  |
| --- | --- | --- | --- |
| FIELD | DATATYPE | CONSTRAINTS | DESCRIPTION |
| TRAINING\_ID | int (11) | Primary Key | Table's Unique Id |
| TRAINING\_STUDENT\_ID | int (11) | Foreign Key | Reference to the primary key of the student |
| TRAINING\_COMPANY\_ID | int (11) | Foreign Key | Reference to the primary key of the company |
| TRAINING\_OFFERED\_STIPEND | int (11) | Not Null | Monthly stipend offered |
| TRAINING\_TIME\_STAMP | timestamp | Not Null |  |
| TRAINING\_ACCEPTED | char (1) | Not Null |  |

**Chapter 6:** **User Interface Design**

**Chapter 7: Testing**

**7.1: Test Cases**

**7.2: Validations**

**Chapter 8:** **Future Enhancement**

**Chapter 9: References/Bibliography**