



Pune Institute of Computer Technology, Pune

DBMS Mini Project (AY 2022-23)

Placement-Tracker

Class- TE1

Batch- L1

Sem- V

Group members (Roll no and Name):

31123 –Sushilkumar Dhamane

31132 – Shreyash Halge

31144 – Sayeed Khan

Lab Teacher Name: Prof. Vijayendra Bagade

Title: **Placement Tracker**

Problem Definition:

Web application for Placement Tracker. Users can fill post their Experience. Users can log in after registration. Users can see all post, and search the post according to their name, company id, and package. Users can add the post to the Dashboard and can do all CRUD operation with their own profile, only read operation is being performed to other user post. The user can see only his own post of other uses. The admin can see all the post and can do all available operation. Admin can create and update the post and confirms the post. Admin can see all the users and can modify their post.

Learning Objective:

- To deal with Placement Tracker in an easy and efficient manner.
- Create a strong database using MySQL that prevents any outside or inside attacks.
- To learn the authentication-based secure login system.

Learning Outcome:

After successfully implementing the mini-project students will:

- Learn web technologies like HTML, CSS, Django, MySQL, bootstrap.
- Implement various MySQL queries in a real-time application.

Abstract:

Placement Tracker is a web application that aims at computerization for posting all placement/Internship Questions at one place. The purpose of this application is to make student aware of internship and placement recruitment process. Student can see placement and internship experiences and can also filter the post based on company, student name, domain. An admin can see all the Post, edit the update, and confirms the Post. The admin can create the post and update the post accordingly. The application provides a user-friendly user interface to interact with the system. The admin authentication provides different services for admin which include the above- mentioned services. The services provided by the backend are handled by the MySQL queries controlled by the MySQL Workbench 8.0 CE. HTML and CSS would be at the frontend and provide a graphical user interface that relates to the user, while the MySQL database will be at the back end to handle the data storage process.

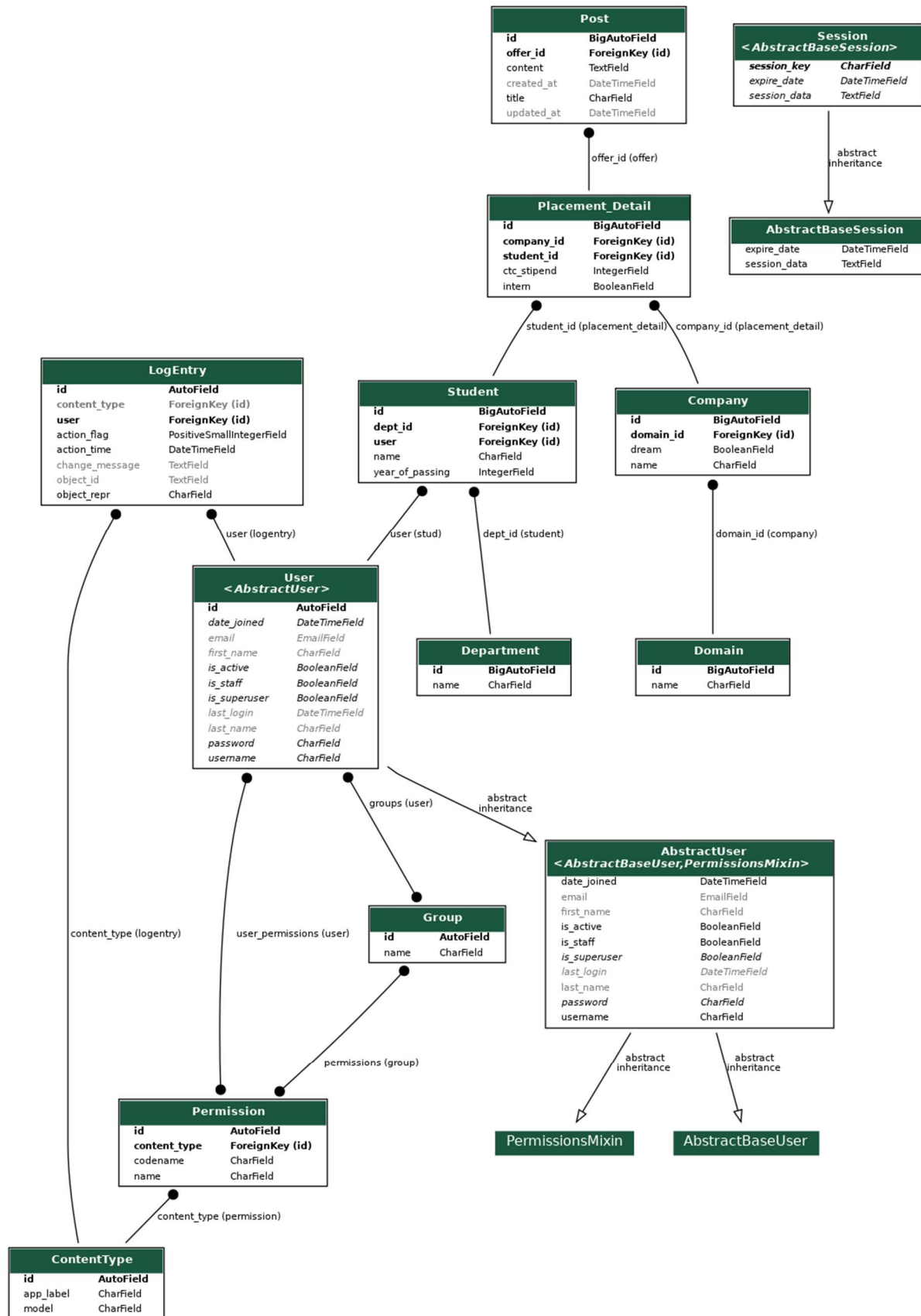
S/W&H/W requirements:**S/W -**

- OS: Windows
- IDE: Visual Studio Code
- Database - MySQL

H/W -

- Processor – Intel Core i5
- RAM – 8GB
- Hard Disk – 1TB
- Keyboard
- Monitor

ER DIAGRAM:



User roles & their description:

1. User

- User can view all the Post details without login or registration.
- User can search for the post and filter it by name, company_id and stipend.
- User can place post only if he is logged in and verified by the authentication - system
- User can view the post placed and see the post details.

2. Admin

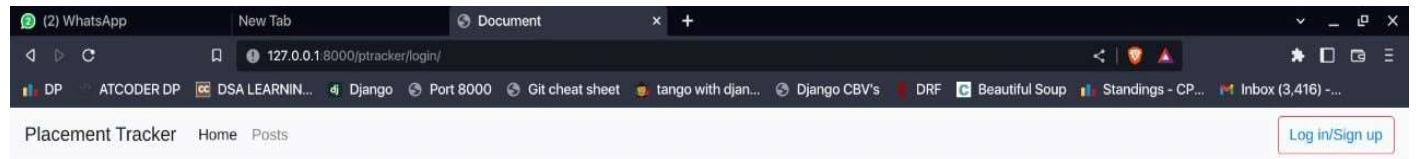
- Manages the database.
- Maintains the user roles.
- Create, view, and update the post.
- Direct Access to the Database.
- Manages all post and confirms the post or deletes the post.
- Possible Error Checks.
If the post is not available admin can post the of topic related post in database.

Functional Description:

- . There are two types of user profiles: User and Admin.
- The User can perform the following functionalities:
 - 1) Register on the application: Requires Username, password, confirm password, passing year and Department of the Student for registration.
 - 2) Log in to view all Post: Require Username and password for login.
 - 3) View All Post and search for the product. - Searching can be done by inputting any word present in the Search bar with different Parameter.
 - 4) Filter the Post according to company, student name and stipend.
 - 5) Adding the Post to the database.
 - 6) View the Post: user needs to be authenticated to view his own Post.
- Our basic target is to keep a database of users, products, and post in concern for the admin.
- The admin can perform the following functionalities:
 - 1) See the Post
 - 2) Update the Post: - Confirm the Post according to the condition. Change the Post Parameters after post
 - 3) Create Post form: Enter the name, Title, Your Post, Company, intern (Bool), and CTC.
 - 4) Managing User roles: Update the role of the User to the admin.

Screenshots:

1) User Registration:



2) SignUp:

The screenshot shows a web browser window with the address bar displaying `127.0.0.1:8000/jptracker/register/`. The browser's tab bar includes 'WhatsApp', 'New Tab', and 'Document'. The page title is 'Placement Tracker' with navigation links for 'Home' and 'Posts'. A 'Log in/Sign up' button is located in the top right corner.

Register

Username

Password

Confirm Password

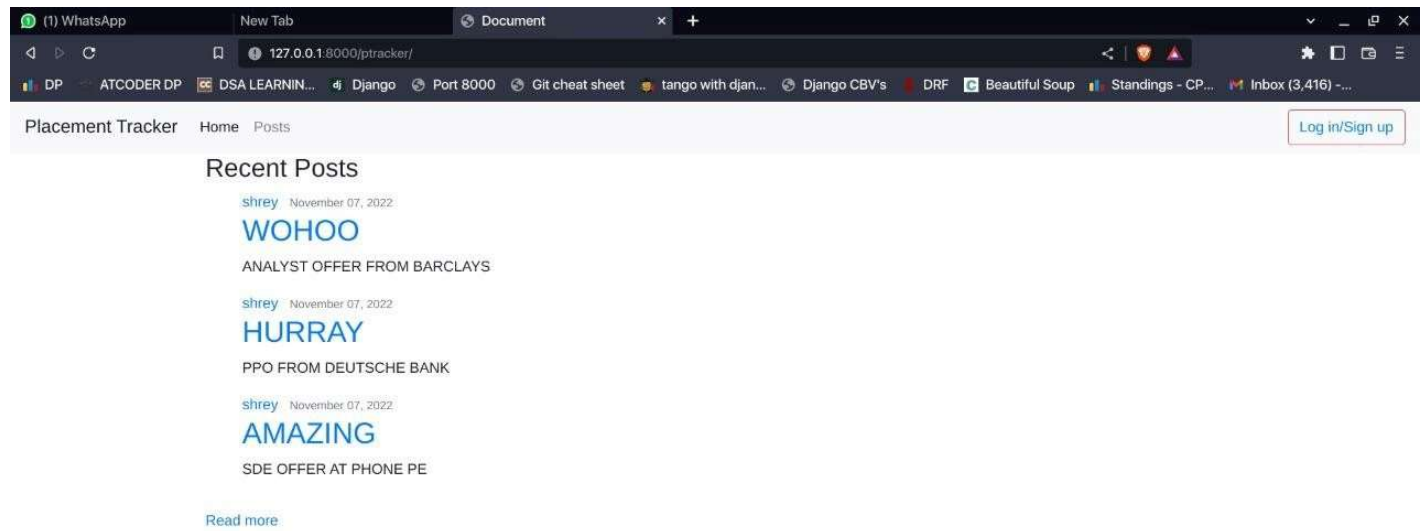
Year of Passing

Department

[Sign up](#)

Already a member? [Login](#)

3)Dashboard:



4) My Post:

Placement Tracker Home Posts shrey Logout

Title

Your post

Company

Barclays

Intern ☐

CTC

Add Post

Conclusion:

By successfully implementing the Placement Tracker we learned web technologies such as HTML, CSS, Django, Bootstrap and implemented MySQL queries in real-time time applications.

Role-based authentication is implemented successfully. The two different user roles perform certain activities according to their roles.

Hence, the application 'Placement Tracker was successfully developed and implemented.