**Shri G.S. Institute of Technology and Science (SGSITS), Indore**

****

**DEPARTMENT OF INFORMATION TECHNOLOGY**

**DATABASE MANAGEMENT SYSTEM**

**IT38513**

**SESSSION 2024 - 25**

**DATABASE ANALYSIS REPORT**

**Submitted To-**

**Asst. Prof. Mukesh Sakle**

**Asst. Prof. Shaivi Barve**

**Submitted By-**

**Name: Adarsh Agrawal**

**Enroll: 0801IT221150**

# INDEX

|  |  |
| --- | --- |
| S.NO. | CONTENTS |
| 1 | INTRODUCTION |
| 2 | ER DIAGRAM |
| 3 | RDBMS |
| 4 | NORMALIZATION |
| 5 | RELATIONAL  ALGEBRA OPERATIONS |
| 6 | CONCLUSION |

# INTRODUCTION

OBJECTIVE:

The primary objective of **YourPlacementBuddy** is to serve as a **one-stop resource hub** for **first-year college students** embarking on their **placement preparation journey**. It provides a **comprehensive collection of guides and resources**, uniquely tailored to their needs. To enhance this experience, we have integrated an **AI-powered roadmap generator** that curates personalized career paths, along with relevant resources sourced from across the web. There is also a community page which allows students to seek guidance, share resources and stay updated on placement trends, coding challenges and interview experience. Additionally, our **database-driven storage** ensures seamless access to saved content, while the **interactive visual roadmap maker** allows users to **customize and create structured mind maps**, helping them navigate their career journey with clarity and confidence.

Key Features:

1. **Roadmap-Based Learning**
   * **Editable Visual Roadmaps**: Graphical roadmaps to track progress.
   * **Branching Options**: Users can customize their roadmap based on their interests
   * **AI-Generated Roadmaps**: Users get personalized roadmaps using **Google Gemini API**
2. **Guidance & Resources**
   * **Links to Hackathons**: Information on ongoing and upcoming competitions
   * **Recommended Courses**: List of useful courses.
   * **Career Advice**: Guides on interview preparation, company selection, and job roles
3. **User Interaction & Customization**
   * **Roadmap Editor**: Allows users to modify roadmaps based on their preferences
   * **Login & Authentication**: Users can log in to save progress
4. **Backend Functionalities (Django + MySQL)**
   * **User Authentication & Progress Storage**
   * **Archive Section for Saved Roadmaps**
   * **Admin Panel for Managing Roadmaps**

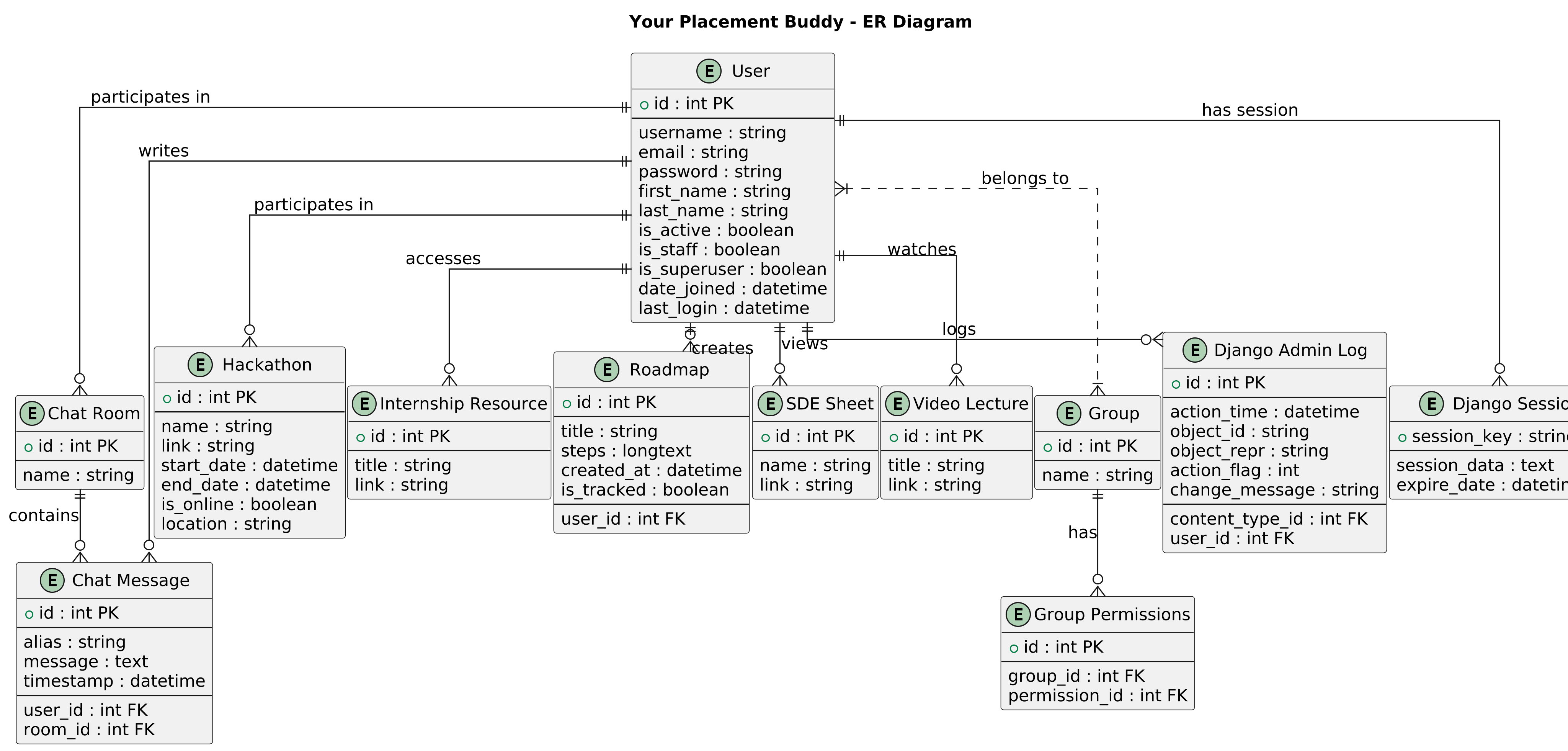
Purpose of The Project:

The YourPlacementBuddy is designed to help students navigate their college journey towards successful placements. With an overwhelming amount of online content, students often struggle to find a clear and structured path for their career preparation. This project provides:  
- AI-generated roadmaps  
- Curated resources for learning and placement preparation  
- Hackathon listings and competitive programming events  
- Roadmap Editor to create a personalized roadmap tailored to your schedule.  
- A user-friendly and ad-free environment for career guidance

Scope of The Project:

1.Roadmap Generation & Editing: Users can create, edit, and track progress on their roadmaps.  
2. Resource Aggregation: The platform provides recommended courses, guides, and learning materials.  
3. Hackathon & Competitive Programming Events: Aggregating links to ongoing challenges.  
4. Backend Functionalities: Data storage, user management, and admin control via Django and MySQL.

# ER DIAGRAM



## (Note: PK and FK refer to Primary Key and Foreign Key respectively)

## User (auth\_user)

* - PK: id
* - username (Unique)
* - email
* - password
* - first\_name
* - last\_name
* - is\_active
* - is\_staff
* - is\_superuser
* - date\_joined
* - last\_login

## Chat Room (accounts\_chatroom)

* - PK: id
* - name (Unique)

## Chat Message (accounts\_chatmessage)

* - PK: id
* - alias
* - message
* - timestamp
* - FK: user\_id → User
* - FK: room\_id → Chat Room

## Hackathon (accounts\_hackathon)

* - PK: id
* - name
* - link
* - start\_date
* - end\_date
* - is\_online
* - location (Nullable)

## Internship Resource (accounts\_internshipresource)

* - PK: id
* - title
* - link

## Roadmap (accounts\_roadmap)

* - PK: id
* - title
* - steps (Longtext)
* - created\_at
* - is\_tracked (Boolean)
* - FK: user\_id → User

## SDE Sheet (accounts\_sdesheet)

* - PK: id
* - name
* - link

## Video Lecture (accounts\_videolecture)

* - PK: id
* - title
* - link

## Group (auth\_group)

* - PK: id
* - name (Unique)

## Group Permissions (auth\_group\_permissions)

* - PK: id
* - FK: group\_id → Group
* - FK: permission\_id

## Django Admin Log (django\_admin\_log)

* - PK: id
* - action\_time
* - object\_id (Nullable)
* - object\_repr
* - action\_flag
* - change\_message
* - FK: content\_type\_id
* - FK: user\_id → User

## Django Session (django\_session)

* - PK: session\_key
* - session\_data
* - expire\_date

## Relationships

* - User (auth\_user) has a One-to-Many relationship with Chat Message (accounts\_chatmessage)
* - User (auth\_user) has a One-to-Many relationship with Roadmap (accounts\_roadmap)
* - Chat Room (accounts\_chatroom) has a One-to-Many relationship with Chat Message (accounts\_chatmessage)
* - Group (auth\_group) has a Many-to-Many relationship with User (auth\_user)
* - Group (auth\_group) has a One-to-Many relationship with Group Permissions (auth\_group\_permissions)

# RDBMS

# schema for chatmessage

# schema for chatroom

# schema for hackathons

# schema for internship resource

# schema for roadmap

# schema for sdesheet

# schema for videolecture

# 

# schema for auth\_group and auth\_permissions

# schema for auth\_user

# NORMALIZATION

To ensure **data integrity** and **minimize redundancy**, normalization was applied to the YourPlacementBuddy database through the following stages:

#### ****First Normal Form (1NF):****

* All tables were structured so that **each column contains atomic values**, and each record is **unique**.
* **Multi-valued attributes** were eliminated.
* Each table has a **primary key**, ensuring that every row is uniquely identifiable.

#### ****Second Normal Form (2NF):****

* **Partial dependencies** were removed by ensuring that **all non-key attributes** are fully functionally dependent on the **primary key**.
* Tables with **composite keys** were further divided to ensure that **non-prime attributes** (attributes not part of the primary key) do not depend on **only part of a composite key**.
* **Example:** The Roadmap table references the User table via user\_id as a **foreign key**, ensuring that each roadmap belongs to a specific user.

#### ****Third Normal Form (3NF):****

* **Transitive dependencies** were removed, ensuring that **non-key attributes** are not dependent on other non-key attributes.
* This reduced the likelihood of **data anomalies** during **insertions, updates, and deletions**.
* **Example:** In the Hackathon table, attributes such as location were separated into distinct tables where necessary, reducing redundancy.

#### ****Boyce-Codd Normal Form (BCNF):****

* **BCNF was applied** to address situations where a table was not fully in 3NF, ensuring that **every determinant is a candidate key**.
* Any **remaining anomalies** due to functional dependencies were resolved by restructuring tables where necessary.
* **Example:** If GroupPermissions had an issue where permission\_id depended on group\_id but was not a candidate key, the table was reorganized to eliminate such dependencies.

By applying **normalization** to the **YourPlacementBuddy database**, we ensured that data is **efficiently structured, redundancy is minimized, and inconsistencies are prevented**. This improves **query performance, data integrity, and maintainability** of the system.

# RELATIONAL ALGEBRA OPERATIONS

# Selection (σ) – Filtering Data

To find all **active users** in the User table:

σis\_active=TRUE​(User)

sql query:

SELECT \* FROM User WHERE is\_active = TRUE;

# Projection (π) – Selecting Specific Columns

**Relational Algebra:**  
To retrieve only **usernames and emails** from the User table:

πusername,email​(User)

SQL Query:

SELECT username, email FROM User;

# Cartesian Product (×) – Combining Two Tables Without a Condition

**Relational Algebra:**  
To combine User and Roadmap tables without specifying any condition:

User×Roadmap

SQL Query:

SELECT \* FROM User CROSS JOIN Roadmap;

# Join (⨝) – Combining Tables with Conditions

### ****Inner Join (θ-Join)****

**Relational Algebra:**  
To find **roadmaps created by each user**:

User⋈User.id=Roadmap.user\_id​Roadmap

**SQL Query:**

SELECT User.username, Roadmap.title

FROM User

JOIN Roadmap ON User.id = Roadmap.user\_id;

### ****Equi-Join (Natural Join)****

**Relational Algebra:**  
To retrieve **chat messages along with user details**:

User⋈User.id=ChatMessage.user\_id​ChatMessage

SQL Query:

SELECT User.username, ChatMessage.message, ChatMessage.timestamp

FROM User

NATURAL JOIN ChatMessage;

# Set Operations – Union, Intersection, Difference

### ****Union (∪) – Combining Two Queries****

**Relational Algebra:**  
To retrieve all **internship and SDE sheet links**:

πlink​(InternshipResource)∪πlink​(SDESheet)

**SQL Query:**

SELECT link FROM InternshipResource

UNION

SELECT link FROM SDESheet;

### ****Intersection (∩) – Common Data Between Two Queries****

**Relational Algebra:**  
To find **users who participated in both chat rooms and hackathons**:

πuser\_id​(ChatRoom)∩πuser\_id​(Hackathon)

**SQL Query:**

SELECT user\_id FROM ChatRoom

INTERSECT

SELECT user\_id FROM Hackathon;

### ****Difference (-) – Subtracting One Query from Another****

**Relational Algebra:**  
To find **users who created roadmaps but never participated in a hackathon**:

πuser\_id​(Roadmap)−πuser\_id​(Hackathon)

SQL Query:

SELECT user\_id FROM Roadmap

EXCEPT

SELECT user\_id FROM Hackathon;

# Aggregation (SUM, COUNT, AVG, MAX, MIN)

* **Relational Algebra:**  
  To count the total **number of users**:

COUNT(User)

SQL Query:

SELECT COUNT(\*) FROM User;

To get the **average number of chat messages per user**:

AVG(COUNT(ChatMessage.id)) grouped by user\_id

SQL Query:

SELECT user\_id, COUNT(\*) / (SELECT COUNT(\*) FROM User) AS avg\_messages

FROM ChatMessage

GROUP BY user\_id;

# CONCLUSION

The **YourPlacementBuddy** database is structured to be **scalable, secure, and efficient**, aligning with the project's objective of guiding students in placement preparation. By applying **relational algebra principles**, **normalization techniques**, and **SQL best practices**, the system ensures **optimal performance** while maintaining a **user-friendly experience**.

# Future Enhancements:

1. Implement AI-based recommendations for roadmaps.

2. Add real-time chat features for community discussions.

3. Improve roadmap visualization using graph-based UI.

4. Integrate with more APIs for fetching external learning resources.