Research Internship Portal

Course Name: Database Management Lab
Course Code: CO303

Submitted By:

Naladala Indukala (Roll No: 15CO230)

R Aparna (Roll No: 15CO236)

Table of Contents

1. Introduction:

Our project aims to simplify the process of research internship management by implementing a portal for indirect communication between professors and students. It encourages students to take up projects according to their interests and tries to match them with professors, ensuring maximum satisfaction for both students and professors through least regret algorithm.

Initially, professors register on the Research Internship Portal and upload their projects. Simultaneously, students register on the portal and are given view access to the entire project database. After all the project details have been uploaded, the students are allowed to choose a maximum of 7 projects and rank the projects according to their interests. After the deadline for project submission has passed, the list of students who have applied to a project is submitted to the concerned professor via the portal. The professor then ranks the students based on their merit and suitability to the project.

Least Regret Algorithm is then applied on professors' ranking of students for projects and students' ranking of projects. Ultimately, best possible effort is made to assign as many students as possible to projects while also keeping in mind the students' interests and professors' preferences.

2. Project Description:

- Registers professors and allows them to upload their project details (Professors are encouraged to submit more than one project proposal to maximise chances of a student being assigned to them.)
- Registers students, allows them to choose projects and upload relative ranking of projects chosen
- Provides professors a list of students who have chosen their projects
- Allows professors to rank the students in order of merit and suitability to project
- Uses least regret algorithm on professors' ranking of students and students' ranking of projects to match students with projects

3. Objectives (specific to NITK):

- Introduces a summer internship program for first/second year B.Tech. students at NITK Surathkal to increase their knowledge in their areas of interest and give them exposure at a very early stage in their careers
- Makes research internship management easy and efficient by making it online
- Encourages interaction between students and professors of respective departments early on
- Introduces freshers and sophomores to research culture at NITK

4. Requirement Analysis:

4.1. Expectations of End Users from the Project

The users of our software are the professors, the students and admin. The requirements of the users from the software are as follows:

- **Professors:** Their personal information such as contact details, publications, research areas, etc should be properly stored in the database. They should be able to upload their projects and get a list of students who are interested in working with them. They should be able to rank the students interested in working with them according to their preference and should be assigned an intern who best suits their project.

- **Students:** Students should be allowed to add and update their details such as academics, training information, project choice, etc in the database. Complete details about projects and professors related to their areas of interest should be available to them. They also need an efficient and impartial method which allocates them projects.

- **Admin**: Admin refers to the governing body of the college. Admin should be given read and write rights to the entire database which comprises of information pertaining to students, professors and projects. He should be able to verify the registered students and professors, i.e., professors and students who have registered on the portal should verify their accounts either through their college email-ids or through the admin.

  Both students and professors need to be given personal IDs and passwords so that access to their respective information is limited only to them. Admin should also be given personal ID and password for maintaining the database.

4.2. User Characteristics:

- **Professor**: Needs to have basic knowledge of operating computers
- **Student**: Needs to have basic idea of how to operate a computer and how to search for projects in database using keywords
- **Admin**: Needs to have basic knowledge about databases and should be good at operating computers

4.3. Functional Requirements (specific to Database Management)

4.3.1. Professor Registration:

- Username (prof_id) and password (pswd) are given to professor, which is stored in database for future logins.
- Professor adds his profile information to the database: Name (p_name), Email ID (email_id), Research Areas (res_areas), Publications (pubs), Contact details (p_phone).

4.3.2. Uploading Project Proposals:

- Professors are asked to upload projects on the database, which are assigned project ids (proj_id) for future reference.
- Professors have to furnish all details pertaining to their projects: Project Description (proj_des), Student Role (std_role), Duration (time_period), Stipend (stipend).

4.3.3. Student Registration:

- Assigns username (std_id) and password (pswd) to student, which is stored in database for future logins.
- Student adds his profile information to the database: Name (std_name), CGPA (cgpa), Academic Details (acad_det), Training Details (trn_det).

4.3.4. Verification of Professor and Student Accounts:

- Heads of Departments are asked to approve/disprove respective professors' and students' accounts by a specific date through admin.
- Apart from through admin, verification of account can also be done via the professors' and students' respective official college Email ID.
- Verified professor and student accounts are retained in database whereas others are discarded.

4.3.5. Uploading Project Choices:

- Students are allowed to view the database of projects and are asked to choose a maximum of 7 projects along with their relative ranking by a certain date.
- List of students who have chosen a particular project is forwarded to respective professors along with their academic details.

4.4. Applications to be built on top of DB (Front End)

- **Addition of Professor and Student records to database:**
  When a professor creates an account through front end, a new tuple is added to Professor Table which contains the professor's profile details. Similarly, creation of new student account adds a new tuple to the Student Table.

- **Profile Updating by Professors and Students:**
  GUI interface is provided for professors and students to indirectly update their details in the database.

- **Uploading of Projects by Professors and Selection of Projects by Students:**
  GUI interface is provided for professors and students to indirectly add details related to project uploading and project selection respectively. These details are entered in the database. The changes made in the database are expressed through ER diagram and relational schema.

- **Matching of Students and Projects:**
  Students' choice of projects and professors' ranking of projects is retrieved from database; least regret algorithm is applied and results of the algorithm are made available to the students and professors. The results are indicated in student's profile as the "internship assigned to them" and in professor's profile as the "student intern assigned to them".

4.5. Most Frequent Operations on Database

- **Prior to release of complete project list**: Updating of project details and profile information of professors and students

- **After release of complete project list**: Updating of student details and students' choice of projects

- **After deadline for submission of project list:** No further changes are made to content of database. Most frequent DB operation now is retrieving students' list of projects and professors' ranking of students.

4.6. Non Functional Requirements

Capacity:
Supports 50 professors, 150 projects and 400 students at a time in the database.

Security:

- Admin password is needed to access central database; so apart from the admin no one can access the database (apart from individual users accessing their own profiles).
- Professors and students need usernames and passwords to access their profiles; this will ensure unauthorized person doesn't manipulate details.

Reliability:
Reliable through information backup; backup is maintained which will ensure records of database are not lost.

Maintainability:
Error log maintained: The number of times wrong password for a particular user is entered is maintained and if it exceeds a certain value, higher authorities are informed.

4.7. External Interface:

- **Software Interface:**
  Front End: Django Framework, HTML and CSS (for webpages)
  Back End: SQLite Database
  Languages Used: C++, Python

- **Hardware Interface:**
  OS: Ubuntu
  Processor: Intel i5 or higher
  Hard disk: 40 GB
  RAM: 256 MB

- **Communication Interface:**
  GUI based Django application must be connected to central database which contains professor, student and project details.

5. ER Diagram:

We have five entities in our database model: Project, Professor, Student, and Admin. The attributes of the entities Project, Professor and Student are explained in Section 4.3.

- The entity Admin has two attributes: Username (admin_id) and Password (pswd).

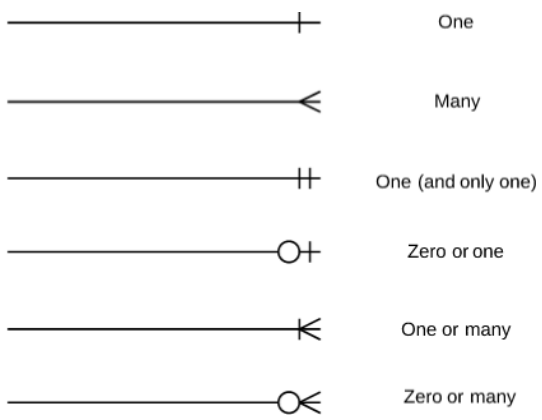
We have depicted the relationship between the entities involved in the database through an ER diagram.

Tool Used: ERDPlus tool has been used to construct ER diagram. ERDPlus has graphical interface, and the ease of constructing ER diagram makes this tool user-friendly.

Legend for ERDPlus:

The following aspects of ERDPlus are different from what has been taught in class:

- Attribute type **Optional** is also included, indicated by (O). An optional attribute may not have a value in it and can be left blank.
- **Cardinality**: Refers to the maximum number of times an instance in one entity can be associated with instances in the related entity.
- **Ordinality**: Refers to the minimum number of times an instance in one entity can be associated with an instance in the related entity.
- **Cardinality** and **Ordinality** are represented by the styling of a line and its endpoint, as denoted by the chosen notation style.



Cardinality Representation in ERDPlus

- **Ordinality:**
    - **Mandatory Participation**: Indicates that a certain role (with its associated entity set) is required for the relationship to exist between the entities concerned. It is denoted as shown below:

    - **Optional Participation:** Indicates that it is not necessary for all the instances of the entity set to participate in the relationship. It is denoted with a line with ‘O’ at the end.

Screenshot of ER Diagram has been placed in the zip folder.

6. Relational Schema

Screenshot of Relational Schema has been placed in the zip folder. Relational Schema was obtained through conversion from ER diagram. This conversion was done directly via ERDPlus.

7. Normalization

- Normalization is the process of analysing the given relational schema based on its functional dependencies and primary keys to achieve desirable properties of minimizing redundancy and minimizing insertion, deletion and updation anomalies.

- First Normal Form (1NF)'s definition states that if a relational schema is in 1NF, then the domain of any attribute in the schema must include only single (atomic) values. To ensure that the relational schema is in 1NF, a new relation "Professor_res_areas" has been made with two attributes res_areas and prof_id.

- Second Normal Form (2NF)'s definition states that a relation schema R is in 2NF if every non-prime attribute A in R is fully functionally dependent on the primary key of R. The relational schema of Research Internship Portal is in 2NF.

- Third Normal Form (3NF)'s definition states that a relation schema R is in 3NF is it satisfies 2NF and no nonprime attribute of R is transitively dependent on the primary key. The relational schema of Research Internship Portal is in 3NF.

- A relation schema R is in Boyce – Codd Normal Form (BCNF) if whenever a nontrivial functional dependency X->A holds in R, then X is a superkey of R. The relational schema of Research Internship Portal is in BCNF.

- We haven't checked whether the relation schema is in Fourth Normal Form (4NF) and Fifth Normal Form (5NF).

- Since the relation schema is in BCNF, it is safe to infer that the relation schema of Research Internship Portal is "good".

8. Least Regret Algorithm

- In Research Internship Portal, we have a set of *p* professors and *s* students who are ranking students and projects respectively. Each of them independently produces an ordered list of the objects he or she chooses to rank. The goal is to produce one list that is the collation of the various ordered lists.

- We can assume that each student's point of view is equally weighted and so is each professor's point of view. Additionally, we assume that point of view of professor has three times the weightage of the point of view of a student. (This assumption has been made by considering 1:30 as the ratio of number of teachers to number of students)

- Thus, we collate the ordered lists while considering the above mentioned assumptions. The result of the collation is a list of three-tuples whose elements are project, professor and student. The highest ranked tuple in the list provides the maximum satisfaction – to both the professor and the student assigned.

- In the next step, we begin assigning projects as per the list and delete redundant entries. For example, if the highest ranked three-tuple involves professor A, student B, project C; then all three-tuples in the list which contain either A, B or C as elements of tuple need to be removed. After this is done, we repeat this procedure on the newly obtained list.

- In this way, we have ensured that the least regret algorithm doesn't make errors like assigning one professor to two students or one student to two professors. Using the concept of "best effort", as many students as possible are assigned projects while keeping in mind the preferences of the students and the professors.

9. Features of Research Internship Portal:

9.1 Login Page:

- Separate login pages are present for Students, Professors and Admin.

- **Student Login Page: (Snapshot 1.1 and Snapshot 1.2)**
  - Unregistered student can create his account by clicking on "New Student Registration" present in Student Login page.
  - Registered students have to enter their respective usernames and passwords to access their profiles. After login, they can perform functions like applying for projects, editing profile, ranking projects, etc.

- **Professor Login Page: (Snapshot 1.3 and Snapshot 1.4)**
  - Unregistered professor can create his account by clicking on “New Professor Registration” present in Professor Login page.
  - Registered professors have to enter their respective usernames and passwords to access their profiles. After login, they can perform functions like editing profile, uploading projects, viewing list of students who applied for project, etc.

- **Admin Login Page: (Snapshot 1.5)**
  - Admin accounts are created through Django administration’s “create superuser” functionality.
  - Admin has to enter username and password for complete access to the database containing information regarding students, professors and projects.

    Snapshots 1.1 – 1.6 have been placed in the folder “Login”.
    ***Snapshot 1.6** shows “Invalid Login” which occurs when the username-password pair doesn’t exist in the database.*

9.2 Account Registration:

- **Student Account Registration: (Snapshot 2.1)**
  - Student has to go to “New Student Registration” in the Student Login page. He enters profile details, i.e., username, student name, password, date of birth, email ID, contact number. Password has to be confirmed by entering it twice.
  - Registration takes place only when contact number and email ID are valid. When student tries to pick a username which has already been picked by some other user, then suitable message is shown (**Snapshot 2.2**).
  - After account has been created, it needs to be verified through student’s college mail id (**Snapshot 2.3**) or through admin verification (**Snapshot 2.4**). After verification, the student is able to access his profile and use the portal.

- **Professor Account Registration: (Snapshot 2.5)**
  - Professor has to go to “New Professor Registration” in the Professor Login page. He enters profile details, i.e., username, professor name, password, contact number, email id and research interests (details).
  - Registration takes place only when contact number and email ID are valid. When professor tries to pick a username which has already been picked by some other user, then suitable message is shown.
  - After account has been created, it needs to be verified through professor’s college mail id or through admin verification. After verification, the professor is able to access his profile and use the portal.

    Snapshots 2.1 – 2.5 have been placed in the folder “Registration”.

9.3 Student Account Functionalities:

- **Edit Profile: (Snapshot 3.1)**
  - Student can change his profile details like email id and contact details. He can upload his resume as well, which will indicate his academic details and previous research work.

- **Change Password:**
  - Student can change his password by entering the new password and the old password.

- **Intern Profiles:**
  - Student can view the projects posted by the professors who are registered on the portal.
    (**Snapshot 3.2** and **Snapshot 3.3**)
  - He can apply to a maximum of 7 projects. He has to assign ranking to the projects selected by him. (**Snapshot 3.4**)
  - The student selects projects based on the project description, stipend offered, duration and his research interests.

- **Notifications for New Research Internships (Snapshot 3.5)**
  - A list of newly added internships is posted here. These are the internships which have been newly posted by registered professors after the student's last logout.

- **Internship Status (Snapshot 3.6)**
  - It indicates whether the student has been assigned a project or not (after application of least regret algorithm). If the student has been assigned a project, the intern position and assigned professor's username are mentioned.
  - This functionality works only after the application of least regret algorithm.

    Snapshots 3.1 – 3.6 have been placed in the folder "Student".

9.4 Professor Account Functionalities:

Snapshot 4.1 shows the profile of the professor's account.

- **Edit Professor Info: (Snapshot 4.2)**
  - Professor can change his profile details like research interests, email id. He can add a list of his publications, his previous research papers, etc.

- **Change Password: (Snapshot 4.3)**
  - Professor can change his password by entering the new password and the old password.

- **Add Intern Profile (Snapshot 4.4):**
  - Professor uploads project along with details like project description, stipend, duration of project, intern position (unique attribute), intern requirements (like branch and degree of student intern).

- **Posted Internships (Snapshot 4.5)**
  - Professor can view the projects posted by him. He can also view the list of students who have applied for his projects.
  - This list contains link to the profiles of the mentioned students. Professor evaluates their profiles and ranks students according to merit and suitability to project. (**Snapshot 4.6**)

- **Assigned Intern (Snapshot 4.7)**
  - It indicates whether the professor has been assigned an intern or not (after application of least regret algorithm). If the professor has been assigned a student intern for his proposed project then the assigned student's username and intern position are mentioned.
  - This functionality works only after the application of least regret algorithm.

  Snapshots 4.1 – 4.7 have been placed in the folder "Professor".

9.5 Admin Account Functionalities:

First admin account is created using Django administration.

- **Database Control**
  - Admin can view the entire database and change the database whenever he wants. (**Snapshot 5.1 and Snapshot 5.2**)
  - He has the right to delete student accounts, professor accounts and uploaded projects (**Snapshot 5.3**)
  - Deletion of students' accounts from database by admin has been enabled to ensure that students don't indulge in malpractices.
  - Deletion of uploaded projects from database by admin has been enabled in case there is any issue with the proposed projects.

- **Freezing of Research Internship Portal**
  - Admin ensures that before the entire project list is made available, no student can use the portal to apply for any project. He can only view the details of the uploaded projects.
  - Also, admin freezes the portal after the deadline for submission of ranking of projects by students has passed. No further changes can be made by students.
  - Similarly, a deadline is set for the professors to submit ranking of students for their projects. After this deadline has passed, professors can't make any changes in their accounts.

- All activities which can be carried out by students and professors in Research Internship Portal can be carried out by the admin as well.

- **Notifications** functionality has been enabled in Admin account which can be used by professors to send notifications to students who have applied for their projects. Professors can send notifications to these students for purposes like scheduling an interview, getting extra details like students' publications. (**Snapshot 5.4**)

- **Change Password** functionality is also available for Admin account.

Snapshots 5.1 – 5.4 have been placed in the folder "Admin".

10. Project Code

Project Code is available at the following GitHub repository:

https://github.com/Aparna16/research_internship_portal.git

The details of packages required for running the code have been mentioned in README file in the repository.
The code has to be executed in the same way as a regular Django application.
The screenshots of terminal taken while running the code have been placed in the folder "*Runcode*".

11. Key Learnings

- Gained the ability to develop database solution for real world problem
- Gained proficiency in SQL
- Understood and executed the steps in designing, developing and querying databases
- Learnt new frameworks like Django
- Gained new skills like CSS and HTML

12. Future Scope of the Project

- Research Internship Portal has been tested on localhost while the real world scenario will have more challenges to deal with, prominently with the handling of large queries. So an efficient algorithm and indexing may be required.

- As of now, the students are not able to efficiently use **keywords** to search for projects in the database. Since the number of projects is less as of now, it is not an issue. But when we scale this portal for big institutions, it becomes important to have an efficient search algorithm for finding projects according to keywords.

- Notifications sent by professors to students for interviews, getting training details, etc. are currently enabled only through admin. This feature needs to be extended such that direct communication between professors and students is possible.

- After the first year of deploying this software, additional features like previous years' projects and internship experiences can be documented to give the current student users an idea about research internships and the portal.

13. References

- www.djangoproject.com
- www.w3schools.com
- www.tutorialspoint.com
- www.dev.mysql.com