CS 220: COMPUTER NETWORKS

Final Project Report

Instructor: Dr. Shah Khalid

"SOFTWARE HOUSE MANAGEMENT SYSTEM"

by:

- Hussain Mehdi Kazmi
  CMS: 366039
- Izzah Aslam
  CMS: 377547
- Hamza Ali
  CMS: 387120

GitHub link

Software House Management System

Overview

Lorem The system we developed was software house management system, The system we developed was a software house management system or generally it may be termed as a company/employee management system when subjected to trivial changes. It provides a beautiful GUI-based approach to manage your company. All the controls are available on one screen, all working efficiently and very smoothly. Also you have full assigned access to the privileges assigned to you and there is also authorization system to keep unauthorized users away. The database is stored and managed using the most latest mySQL Workbench that is free as well as convenient. Overall, the system provides all the features to the user he requires to manage and monitor his company.

Tools Used

The tools used in development were the most professional up to date and latest, providing important features of app development such as reliability, convenience and accessibility of all the important accessories. These identify as:

VSCode by Microsoft for Python, Frontend(Graphical User Interface), backend(database connectivity) and logic implementation. It also has the ability to run embedded SQL in Python that was used extensively in our project. Use of this tool added to our learning of professional tool usage and development.

MySQL Workbench for database creation, normalization, data insertion, query running, testing and monitoring of the database. It is a famous, open source and powerful relational database management system.

Technologies Used

The technologies used in development of this project were the latest and trending in professional desktop app development, providing reliability as well as efficiency.
These include:
Database: ***MySQL Workbench***
GUI Development: ***Python tkinter***
GUI Styling: ***Python customtkinter***
App. database connectivity: ***Python MySQL.connector***

Goals

1. Secure the company data through a reliable authorization mechanism.
2. Provide the user a suitable approach to the data.
3. Efficient management of the database.
4. Provide an interactive graphic-based visualization.
5. Effective access to the controls for the data deletion, updation, insertion and management.
6. Self-explanatory commands.
7. Beautiful GUI with focus on reliability and efficiency.
8. Providing fast query processing.
9. Availability of django web connectivity so that admin may locally manage the data of the organization.
10. Settings for the user to customize the interface as he wants to. Theme selection, color scheming and UI scaling options are available, many more in progress.
11. Additional features such as real time clock, notifications and sidebars for graphical report of the in-progress projects.

Specifications

This Software House Management System project will help the organization with the management of the information of the employees. The system will centralize the management system and will provide different options, through which access to data will become accessible. The system will be based on the internet so that any user can use it from any place with ease. The Employee Management System software will make a report of each employee of the organization at the end of the month so that the organization will have information about the work of each employee. This Employee Management System will not only reduce the time but as well it will make the system efficient.

Milestones

- ☑ Assembling a project team
- ☑ Project approval
- ☑ Start database development phase
- ☑ Getting database normalized
- ☑ Start GUI development phase
- ☑ Getting our GUI ready
- ☑ GUI connecting to Database
- ☑ Project kick-off meeting
- ☑ Completing critical amendments
- ☑ Producing key project deliverables
- ☑ Reaching project goals and objectives
- ☑ Project completion
- ☑ Project submission

Normalization :

Erd is in **BCNF** . there are no non-prime attributes determining prime attributes or non-prime determining non-prime attributes .

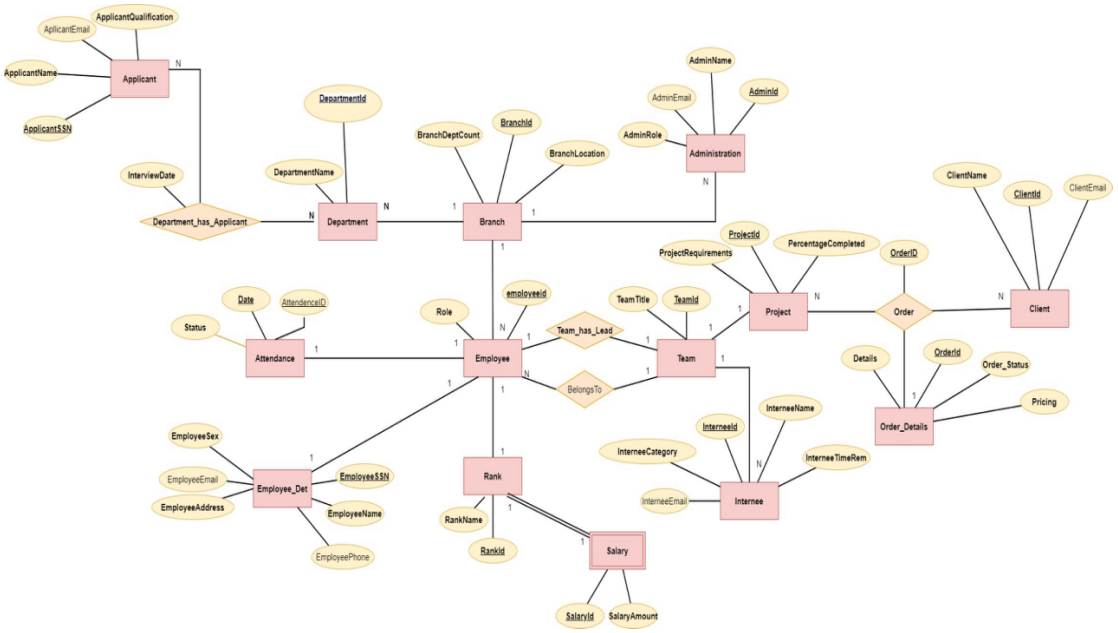
Main tables :

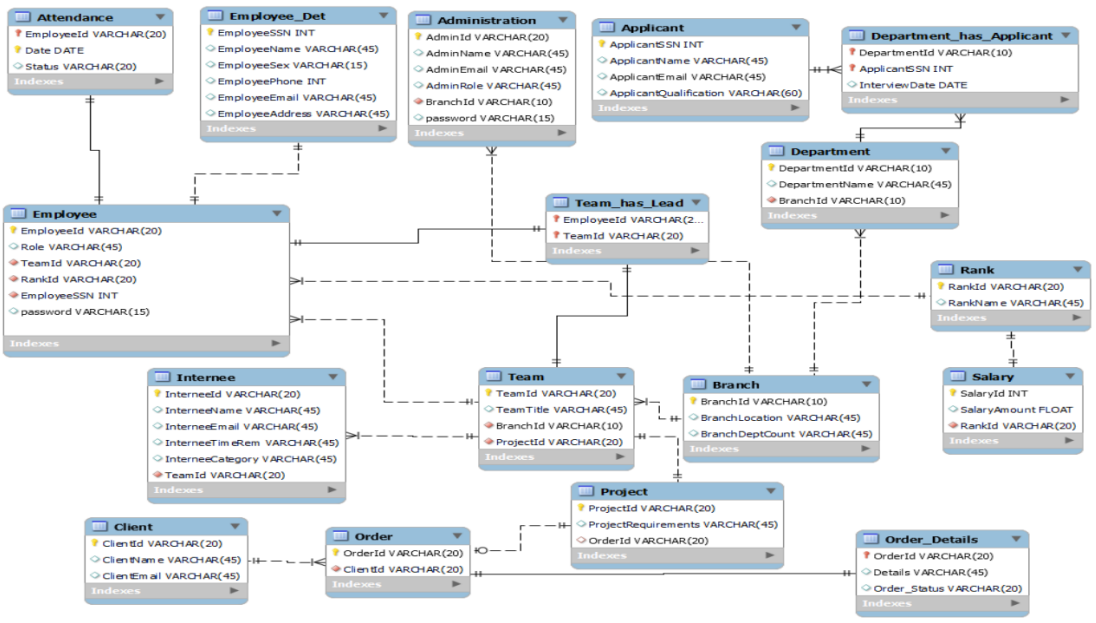
- ❖ Employees
- ❖ Administration

- Projects
- clients
- Branch
- applicants

Which were normalized to tables given in ERD .making separate table for contracts of projects from client, teams of employees ,departments in branches on which applicants table depend ,salary is given according to rank so cannot be placed at employee table where primary key is empl_ID etc

Entity relationship model:





Main table is of administration where all information regarding admins of different branches are recorded.

Only branch table is connected with it as it manages overall branch activities and users.branches have different departments and applicants can apply and given interview date .

clients will give order whose information will be stored in order_details table while different teams of employees for specific projects will be made.

Team and employees have two relations one of employee will be the leader (One-to-One relation) while other employee will belong to team(Many-to-One relation).

Employees get salary according to their rank employee role decide whether he is programmer , review programmer,graphic developer etc while ranks decide is he senior junior or trainee so their is total participation of weak entity Salary as all values of rank their should be a salary value given

Another thing here is for employee_details primary key is SSN As person is uniquely identify by its SSN but employee on work is known by its id number so emp_id will be used as a foreign key on other tables.

❖ Login page:

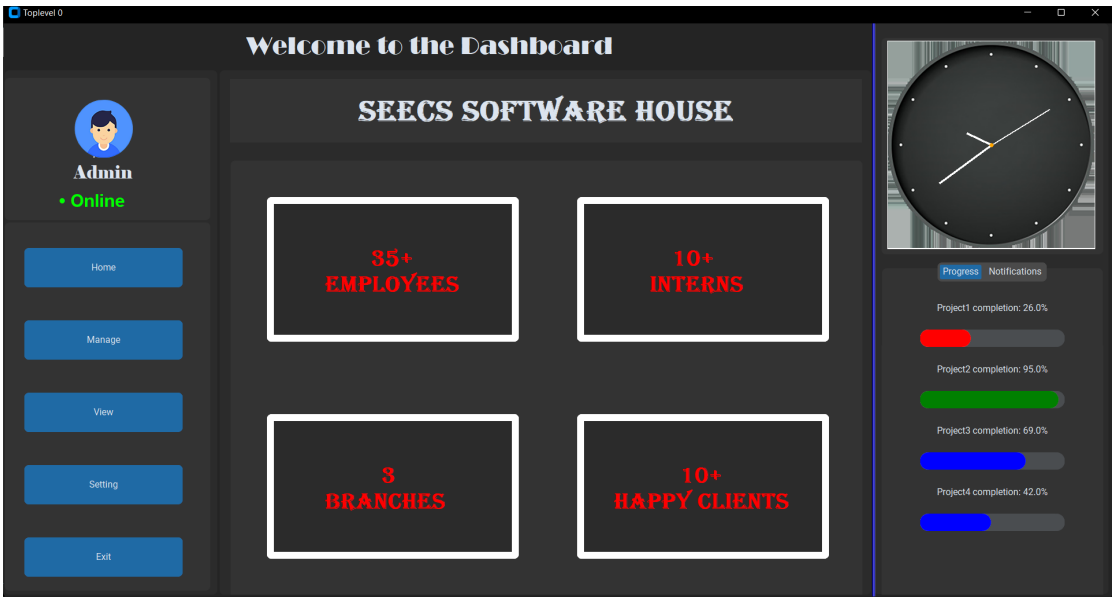
Preview:

Email and password are fetched from the database to check if the email and password entered are correct before signing in the user.

If the email and password given are not correct it will display an error another option provided is forget password if user forgets password he can click it and email will be sent to his email about password.

❖ Home page:

Preview:

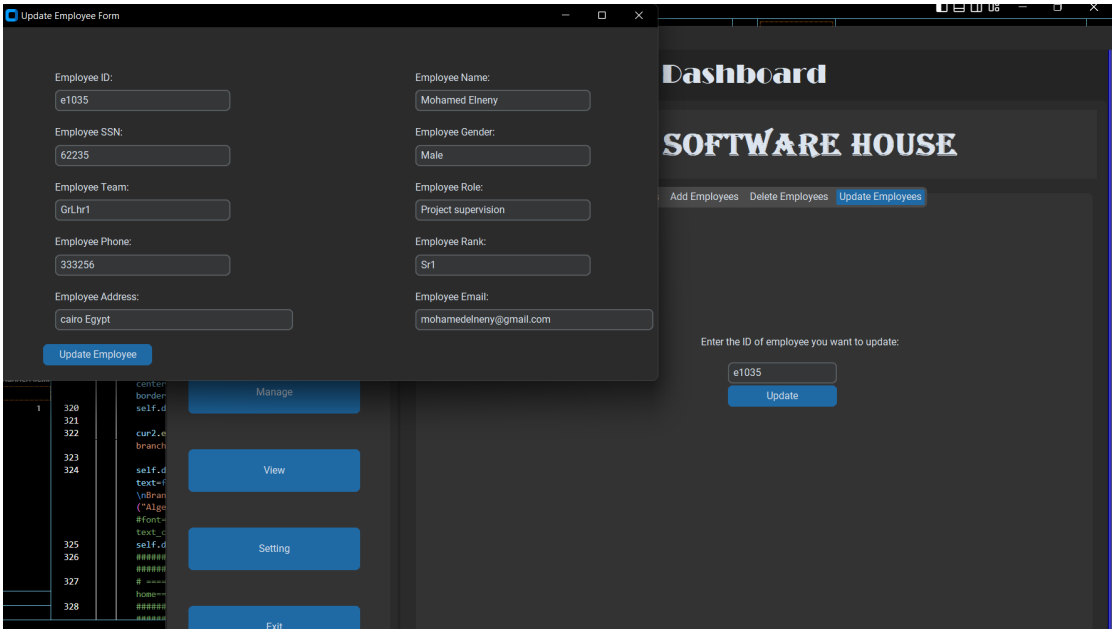


At home page overall count of employees , department , internee etc is given .So the admin can have a view at left bottom corner project completion status is shown that what percentage of projects have completed while percent of project rejected etc. and notification will also be displayed there if the client had ordered something etc .at right top corner online status is shown

❖ Manage page:

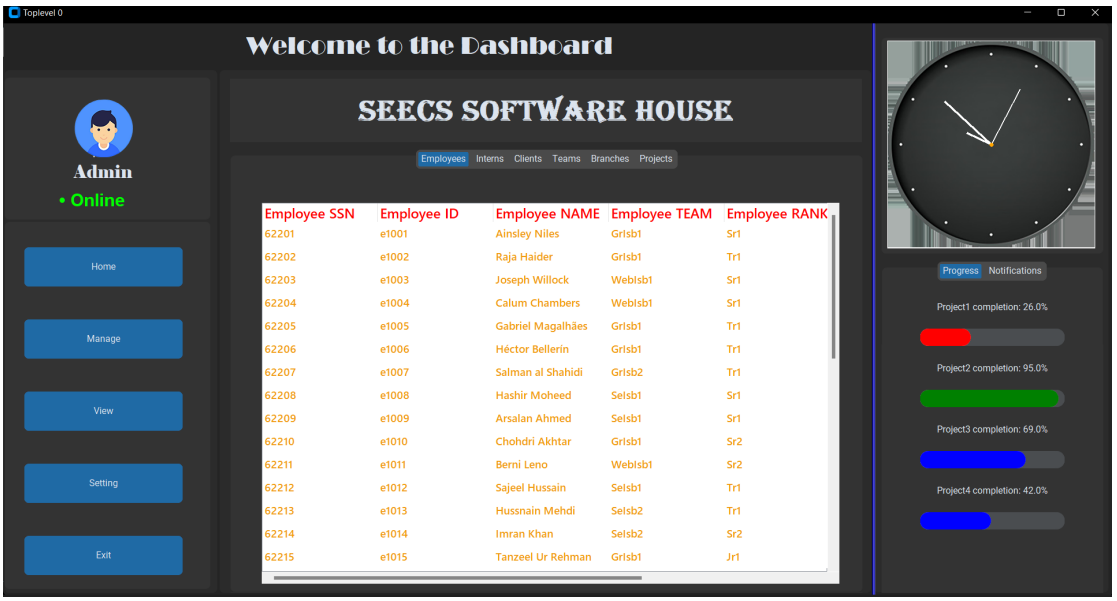
Preview:

Sample Click Preview Employee Update option:



Updating record in database option is also provided whenever employee records have to be provided all information will be recorded while for updating and deletion only Id number is required.

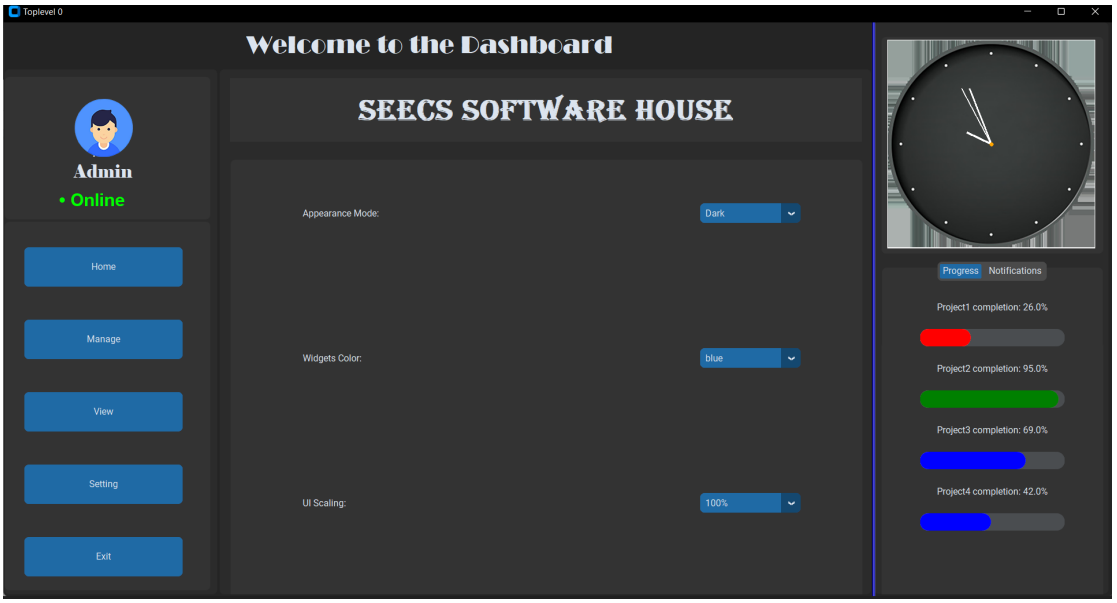
❖ View page:



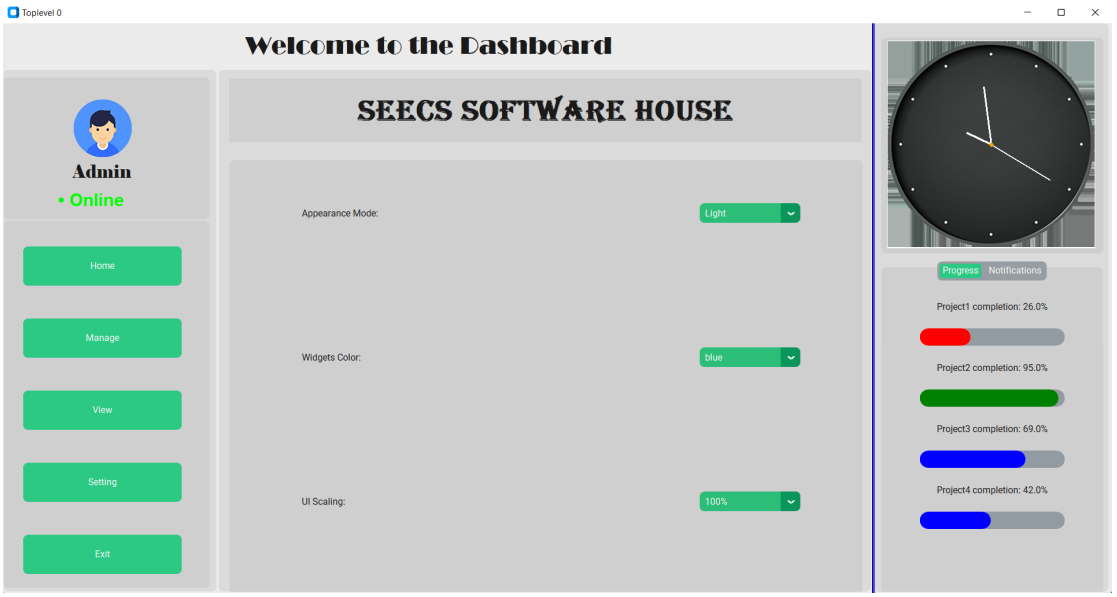
Different views of employees, interns, client, teams, branches and project are shown in this tab.

❖ Setting page:

Preview for light mode, blue theme color and 100% Scaling:



Preview for light mode, green theme color and 100% Scaling:



Options to change the color of text or mode while zooming in and out option is also provided here

❖ Exit Button:

Exit button as shown above logs out the user and terminates the GUI.

Code Reusability:

The employee table is created by this application, and a few rows of data are added to it. If you desire to utilize this code as a starting point for a comparable table in your project, you can modify the table's structure and the data being inserted to fit your needs. For instance, you could want to add or remove columns from the table or change the data types or choices for the current columns. You can change the data being entered by changing the values or adding more rows. Nevertheless, if your project requires a significantly different table structure or set of data, this code might not be readily transferable. In that case, you would need to develop

Implementation and Future trends:

We have been successful in developing the desktop app for the project successfully. If anyone wants to use it practically for his organization, he may use it solely or may connect it to the organization website using any Python backend framework such as Django or so. This will benefit him in the sense that he would be able to manage the orders, projects, progress, employees and other elements of the organization locally. Code has been neatly written, it is error proof and it can be modified to fit one’s exact needs. We hope we have been successful in developing a practical and interactive app that did help us learn many useful concepts, also it may prove to be helpful to many organizations and individuals.