CSE 201 - Advanced Programming
Project 2
Facility Management Services (FMS) System

The purpose of this project is to create a desktop application to manage a system similar to FMS in IIIT Delhi.

Departments in the system:

- Electricity
- Heating, Ventilation, Air-Conditioning(HVAC)
- Audio/Video
- Security
- Housekeeping

Users in the system:

- GM or Admin
- Electricity supervisor and staff
- HVAC supervisor and staff
- Audio/video supervisor and staff
- Security supervisor and security staff
- Housekeeping supervisor and staff

Functional Requirements:

- GM, supervisors and staff can login and logout of the system.
- Every user gets a unique ID and username when added to the system.
- There can be only one admin of the system and one supervisor from each department.
- Registration of the new staffer should be approved by the admin or supervisor.
- **GM or Admin:**
    - Can add/view/delete supervisors and staff.
    - Can assign task to supervisors.
    - Can approve/reject or keep the logistics requests by supervisors on hold.
    - Whatever is visible to supervisors and staff in the system should also be visible to the admin.
- **Department Supervisors:**
    - Can add/view/delete his/her department staff.
    - Supervisors can assign tasks to an individual staff member or to some staff members or to all staff members at once.
    - Department supervisors maintain logistics of their respective departments, approve requirement requests by staff.
    - Supervisors can send logistics approval requests for inventory to GM.
    - Send leave to GM.
    - View task reports.
- **Staff:**
    - Send logistics requirement for task to department supervisor.
    - Send leave to department supervisor.
    - Update status of the task to ONGOING or COMPLETE.

    - Generate task report (Can be saved in pdf or text format).
- When a staffer is being assigned a task, his/her status (Busy (till when), Available, on Leave (till when)) should be visible to department supervisors/admin.

- **Tasks:**
    - Tasks may or may not have a completion deadline.
    - Staffer reports back on his/her assigned task with one of the three statuses:
      1. NOT STARTED (default)
      2. ONGOING
      3. COMPLETE
    - If status is COMPLETE then staffer should have the option to generate a task completion report.
    - Task assignments and reportings must all be accompanied with a system generated timestamp.

Bonus: (Max 20%)

1. **(10%)** Give the users in the system an option to login into the system using Google OAuth.
2. **(1%)** For every design pattern that you use in your implementation. **(max 5%)**
1. **(2%)** If you generate doxygen comments. Refer: (http://www.stack.nl/~dimitri/doxygen/manual/docblocks.html)
2. **(1%)** If all the classes in your system are less than 200 lines.
3. **(1%)** If every function in your system is less than 50 lines.
4. **(1%)** If you use CVS/SVN/GIT with more than 50 commits and not more than 3 commits in any day.

Some things to consider:

- This project will require you to design a lot of forms and a different interface for the users mentioned above. Here, I have just given 2 layouts to give an idea about how you should start with system's interface. Rest is an open design problem and it is upto you how user friendly you want the system to be.

- It is recommended to first design a system with the help of use case diagrams, class diagrams, activity diagrams, etc, discuss them with TAs and then start with the implementation. You will also need to make a report for the project which should include these diagrams.

- As you do not know about databases and database query languages, it is recommended to keep the data in csv files and access them using APIs or by writing your own custom functions.

- Specifically for data, there can be a separate file for the admin, supervisors, staff (can be a separate file for every deparatment staff), inventory (logistics), etc. Remember that you would need to read as well as update these csv files, so an efficient design is the key here.

- There can be more forms needed in the system which might come up when you will start designing the system.

To summarize the tasks for you:

Task 1: Design the system (functionalities) with the use of different diagrams.

Task 2: Design a database (type of content in csv files) based on the designed system in task 1.

Task 3: Implement a GUI application.

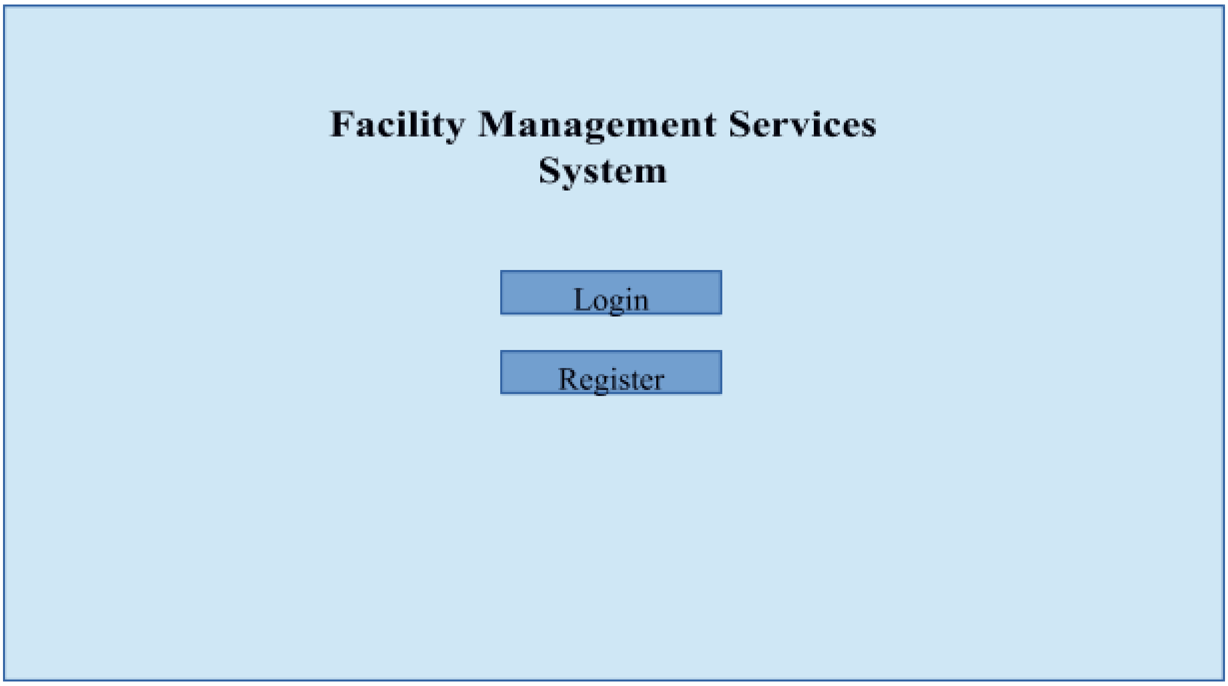
Some Sample Form Content:

- Login – username, password
- Registration – ID (system generated), Type (Staffer/supervisor), Name, Username, password, DOB, Address, department, etc.
- Task – taskID (System Generated), Task Name, Task Description Department, Supervisor, Staff (One or more), Equipment, status, deadline, etc
- Leave – to Whom (Admin, Supervisor), Reason, Date range, etc.
- Logistic approval request – ID, Item(s) with quantities, department,etc.
- Logistic requirement request for task, ID, item(s) with quantities, task Reference ID, etc.
- Task Report – ID, Task ID, Task Name, Task Description, Items used, Time Taken, Comments.

Reference Layouts:

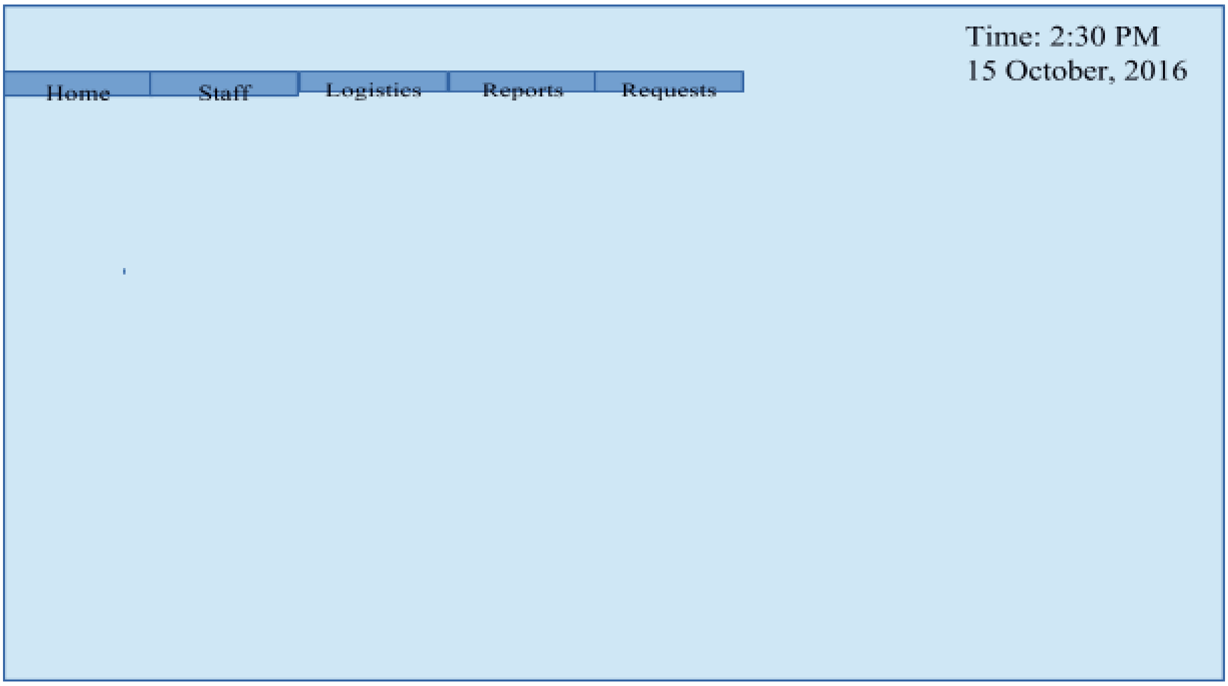
These are very basic layouts and are in no way constraining you in terms of functioning. Your design can be very different (should be) depending upon how much effort you put in thinking about the application and different functions in the system.

1. Welcome Screen



2. Home screen of admin (Will be different for supervisor and staff)

Home can show the user the last time he/she logged in and a summarization of other tabs.


Time: 2:30 PM
15 October, 2016
Home
Staff
Logistics
Reports
Requests