**Semester Project Documentation** (CS – 101 – Computing & AI)

*(You are required to follow the mentioned font size and style for the documentation. In this document heading are of 14 font size while regular text is of 12 font size, while the font style is “Times New Roman”)*

**Semester Project Title**: Hostel Management System

**Student Details** (Fill according to number of team/group members)

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Student Name** | **Student Reg #** | **Student Degree** |
| **Student-1** | Muhammad Talal Iqbal | 202476 | BSAI |
| **Student-2** | Atta Ur Rahman Sheikh | 202122 | BSAI |
| **Student-3** | Mazhar Guftar | 202288 | BSAI |

1. **Main Features**

*(You will mention features of your application as shown for the example of uber application each by numbering as 1, 2, 3 … etc., you are required to use the font size 12 and times new roman font)*

1. Registration and Login mechanism for Students and Admin.
2. Hostel Allocation based on student gender.
3. Management of Student Profiles and Hostel Details.
4. Student Room Transfer Requests.
5. Notifications and Parcel Alerts for Students.
6. Admin Complaint Management and Announcements.
7. Report Generation for Hostel Statistics.
8. **Types of Users & Requirements**

*(Just Mention the types of users for your project as shown in following uber example, and their user specific requirements)*

1. Student**:**
   1. Log in to their account using registration number and password.
   2. View and edit their profiles.
   3. Submit room transfer requests.
   4. Access parcel alerts and notifications.
   5. View announcements from the admin.
2. Admin**:**
   1. Log in using a master key.
   2. Manage student accounts (add, search, and update student data).
   3. Handle room swap and complaint requests.
   4. Make announcements and send notifications.
   5. Generate reports on hostel statistics.

**3. Requirements Breakdown**

*(Write requirements of each feature of your project by numbering as show in example for uber application. For example: Requirements of feature # 1. It will be mentioned as 1.1, 1.2. Similarly for feature # 2 it will be 2.1, 2.2, 2.3 … etc.)*

1. Registration and LoginMechanism**:**
   1. Display a menu for users to select login type (Student/Admin).

1.2 Save student details in a structured file during registration.

1.3 Verify user credentials during login by comparing with stored data.

2. Hostel Allocation:

2.1 Read student data from a file and calculate male/female counts.

2.2 Assign hostels based on gender and room availability.

2.3 Save updated allocations in a new file.

1. Student Profile Management:

3.1 Display detailed student profiles, including name, contact, and hostel details.

3.2 Allow password updates with proper verification.

1. Room Transfer Requests:

4.1 Accept and log room transfer requests.

1. Notifications and Parcel Alerts:

5.1 Display notifications specific to students from a file.

5.2 Provide a placeholder for parcel alerts.

1. Complaint and Announcement Management:

6.1 Track student complaints from a file.

6.2 Allow admins to resolve and log complaints.

6.3 Post announcements to a file accessible by students.

1. Report Generation:

7.1 Calculate hostel statistics such as total students, male/female counts, and hostel occupancy.

7.2 Display statistics in the console.

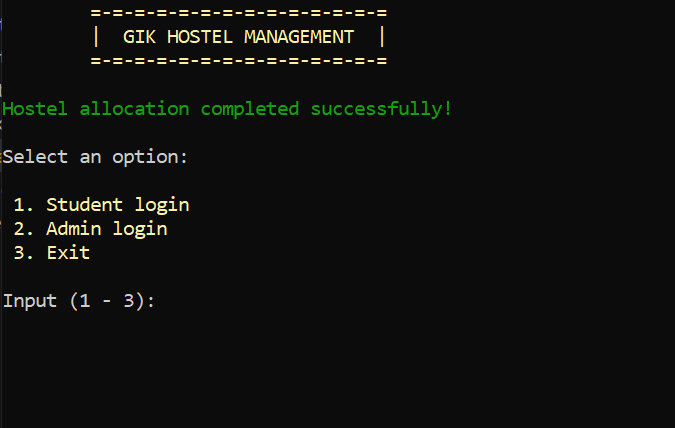
1. **Features to Codding Matrix**

*(In the following table you will mention the following items for each feature, mention the items in each column for each feature of your application)*

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Sr**  **#** | **Feature Name** | **Programming Concept Used** | **No. of Files Used by File Handling** | **No. of Arrays Created** | **No. Pointers / References Used or Created** | **No. of Variables & Objects Created** | **Functions Created** | **Line of Code Written** |
| 1 | (For Example)  Login & Register | Arrays, functions, references and file | 1 (read/write) | 2 | 3 | 6 | 3 | 70 |
| 2 | Hostel Allocation | Arrays, loops, conditional logic | 1 (read/write) | 2 | 2 | 6 | 1 | 60 |
| 3 | Profile Management | Arrays, functions, user input | 0 | 1 | 3 | 5 | 2 | 80 |
| 4 | Room Transfer Requests | Functions | 0 | 0 | 1 | 2 | 1 | 20 |
| 5 | Notifications & Parcel Alerts | File handling, functions | 1 (read) | 1 | 1 | 2 | 2 | 50 |
| 6 | Complaint Management | File handling, functions | 1 (read/write) | 0 | 2 | 3 | 3 | 70 |
| 7 | Report Generation | Loops, functions | 0 | 2 | 1 | 2 | 1 | 130 | 0 |  |
|  |  |  |  |  |  |  |  |  |

1. **Project Screenshots**

*(Take feature wise screenshots of main running screens of your project and paste here)*



A screenshot of a computer

Description automatically generated

A computer screen with text and numbers

Description automatically generated

A screen shot of a black screen

Description automatically generated

A screenshot of a computer

Description automatically generated

A screen shot of a computer

Description automatically generated

A screenshot of a computer error

Description automatically generated

A screenshot of a computer program

Description automatically generated