Design Document

for

DigiCampus

**Version 2.0**

**Prepared by**

**Group #: 1 Group Name: Achievers**

| **Dishay Mehta** | **200341** | **dishaymehta952@gmail.com** |
| --- | --- | --- |
| **Abhishek Pardhi** | **200026** | **abhipardhi326@gmail.com** |
| **Samarth Arora** | **200849** | **samartharora03@gmail.com** |
| **Ankur Kumar** | **200140** | **ankurrk04@gmail.com** |
| **Shashwat Gupta** | **200923** | **guptashashwatme@gmail.com** |
| **Ananya Agrawal** | **200117** | **ananyaagrawal704@gmail.com** |
| **Girik Maskara** | **200387** | **girikmaskara552002@gmail.com** |
| **Sarthak Kohli** | **200886** | **sarthak811kohli@gmail.com** |
| **Aayush Kumar** | **200008** | **aayushk0072@gmail.com** |
| **Aryan Vora** | **200204** | **aryanvora23@gmail.com** |

| **Course:** | **CS253** |
| --- | --- |
| **Mentor TA:** | ***Aman Aryan*** |
| **Date:** | **27/04/22** |

Content

**Contents ii**

**Revisions ii**

**1** **Context Design 1**

1.1 Context model 1

1.2 human Interface Design 1

**2 Architecture design 2**

**3** **object-oriented design 3**

3.1 Use case diagram 3

3.2 Class diagram 3

3.3 sequence diagram 3

3.4 State diagram 3

**4 Project Plan 4**

**5** **Other Requirements 5**

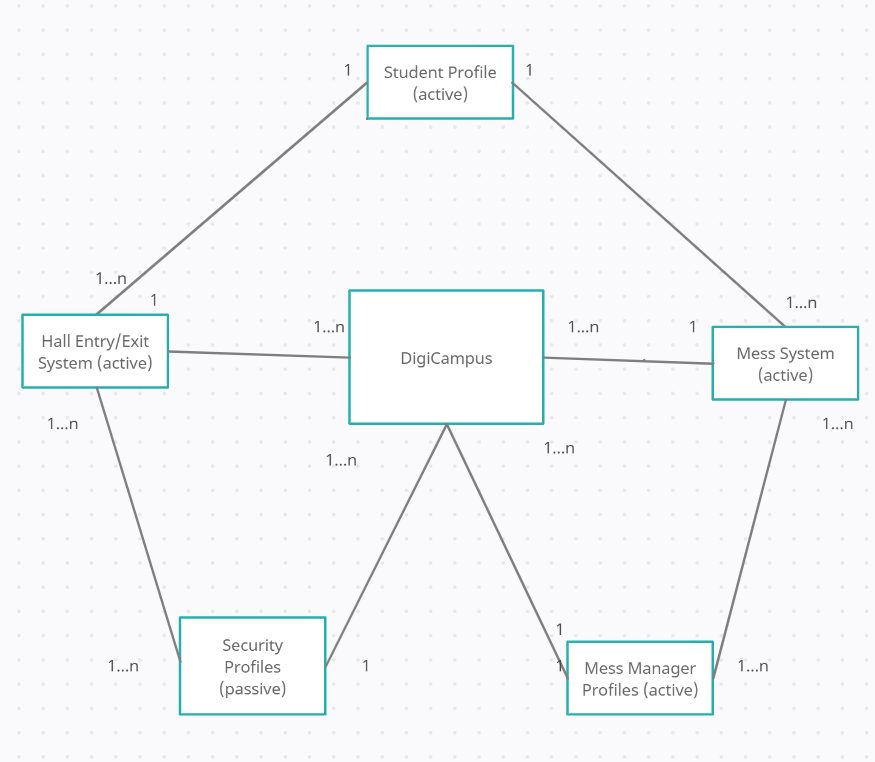
**Appendix A - Group Log 6**

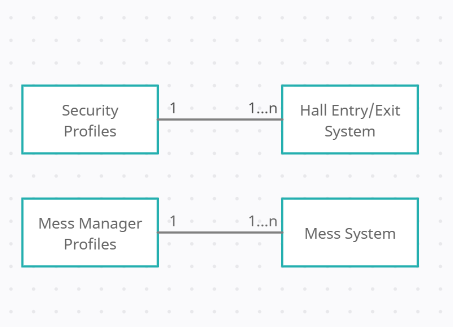
**Revisions**

| **Version** | **Primary Author(s)** | **Description of Version** | **Date Completed** |
| --- | --- | --- | --- |
| 1.0 | Dishay Mehta,Abhishek Pardhi,Samarth Arora,Ankur Kumar,Shashwat Gupta,Ananya Agarwal,Girik Maskara,Sarthak Kohli,Aayush Kumar,Aryan Vora | Design Document created | 15/02/22 |
| 2.0 | Dishay Mehta,Abhishek Pardhi,Samarth Arora,Ankur Kumar,Shashwat Gupta,Ananya Agarwal,Girik Maskara,Sarthak Kohli,Aayush Kumar,Aryan Vora | Version 2.0 updated | 27/04/22 |

# Context Design

## Context Model

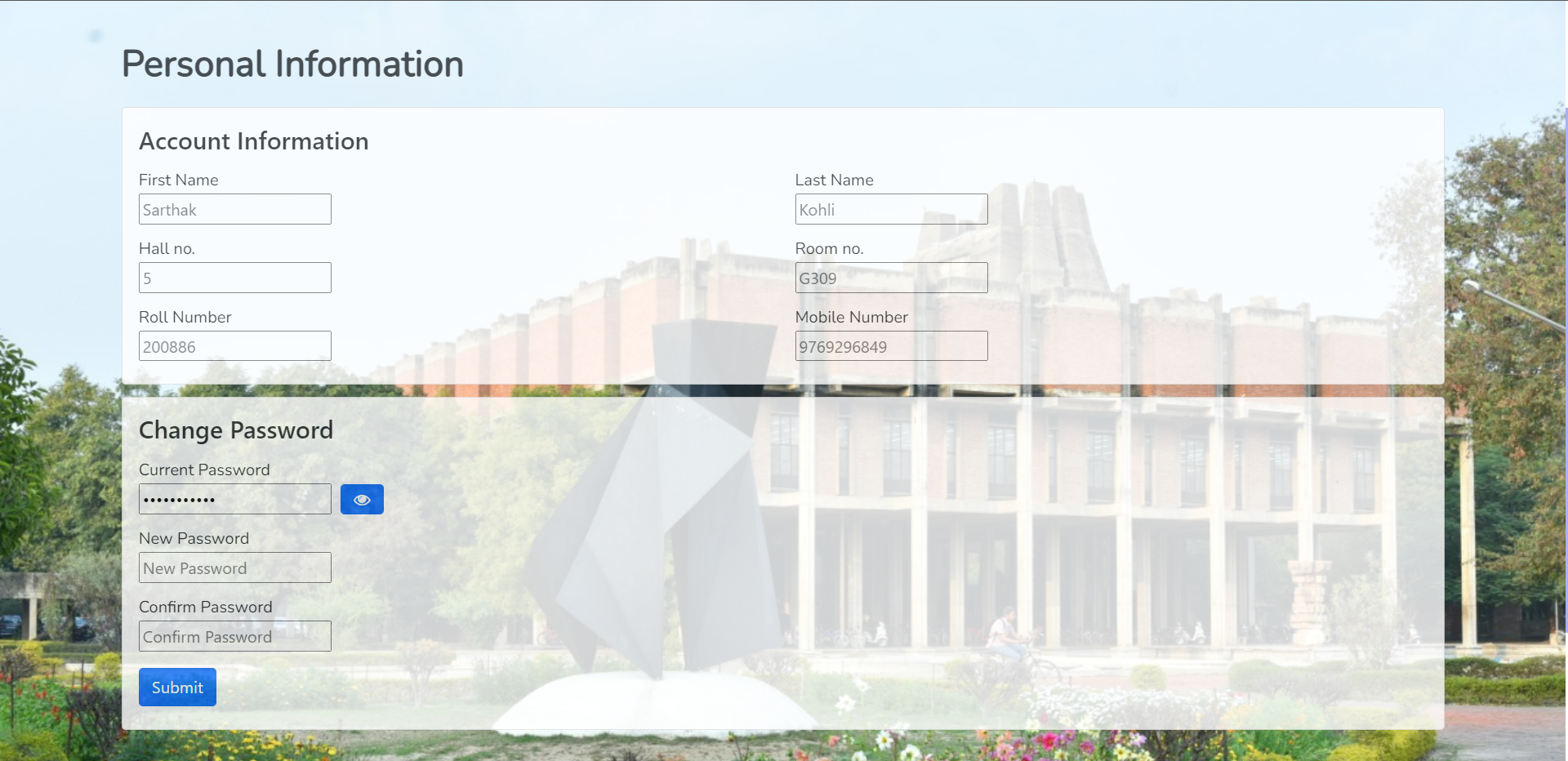




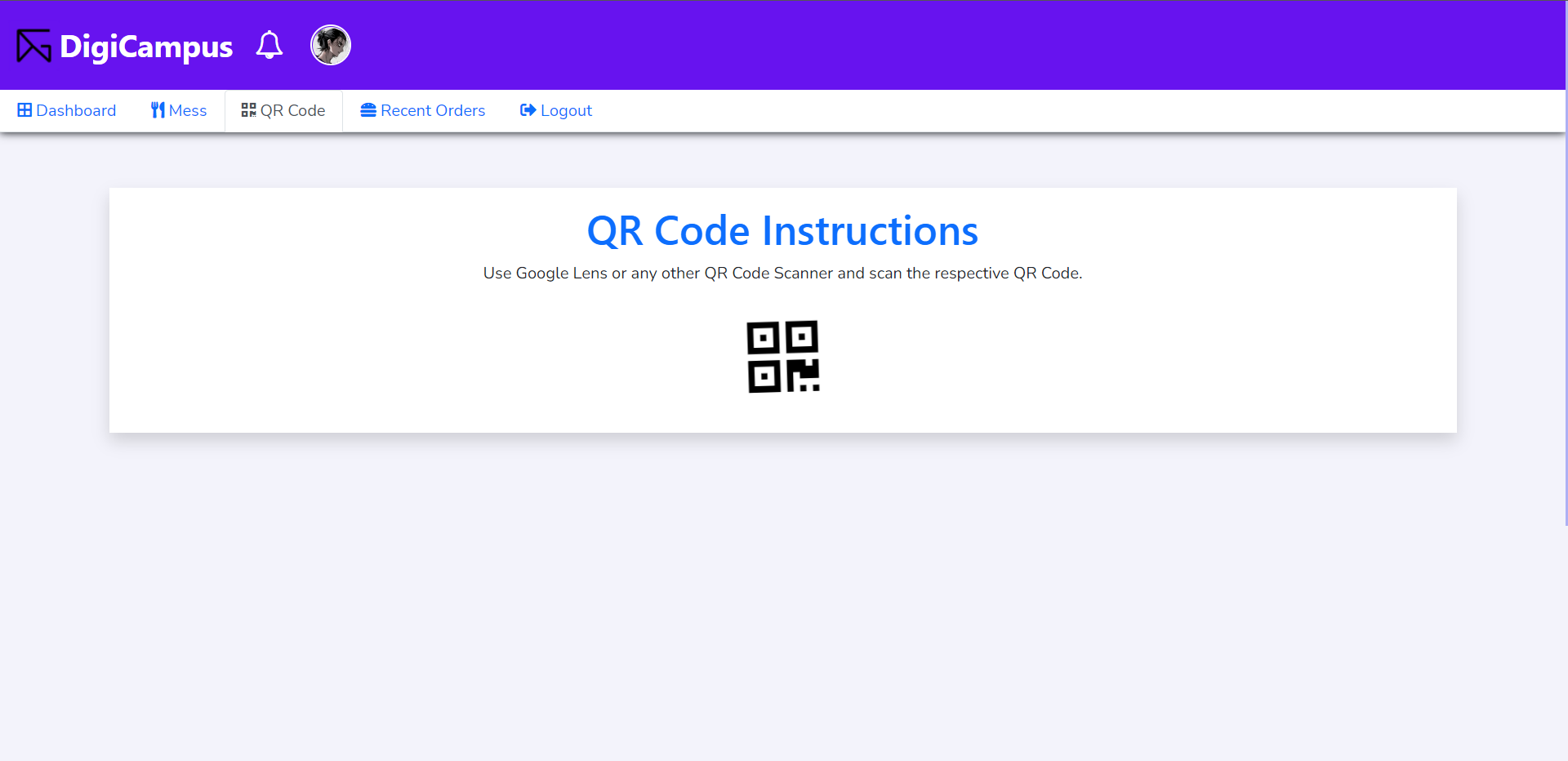
## Human Interface Design

The first version of the human interface looks like this:

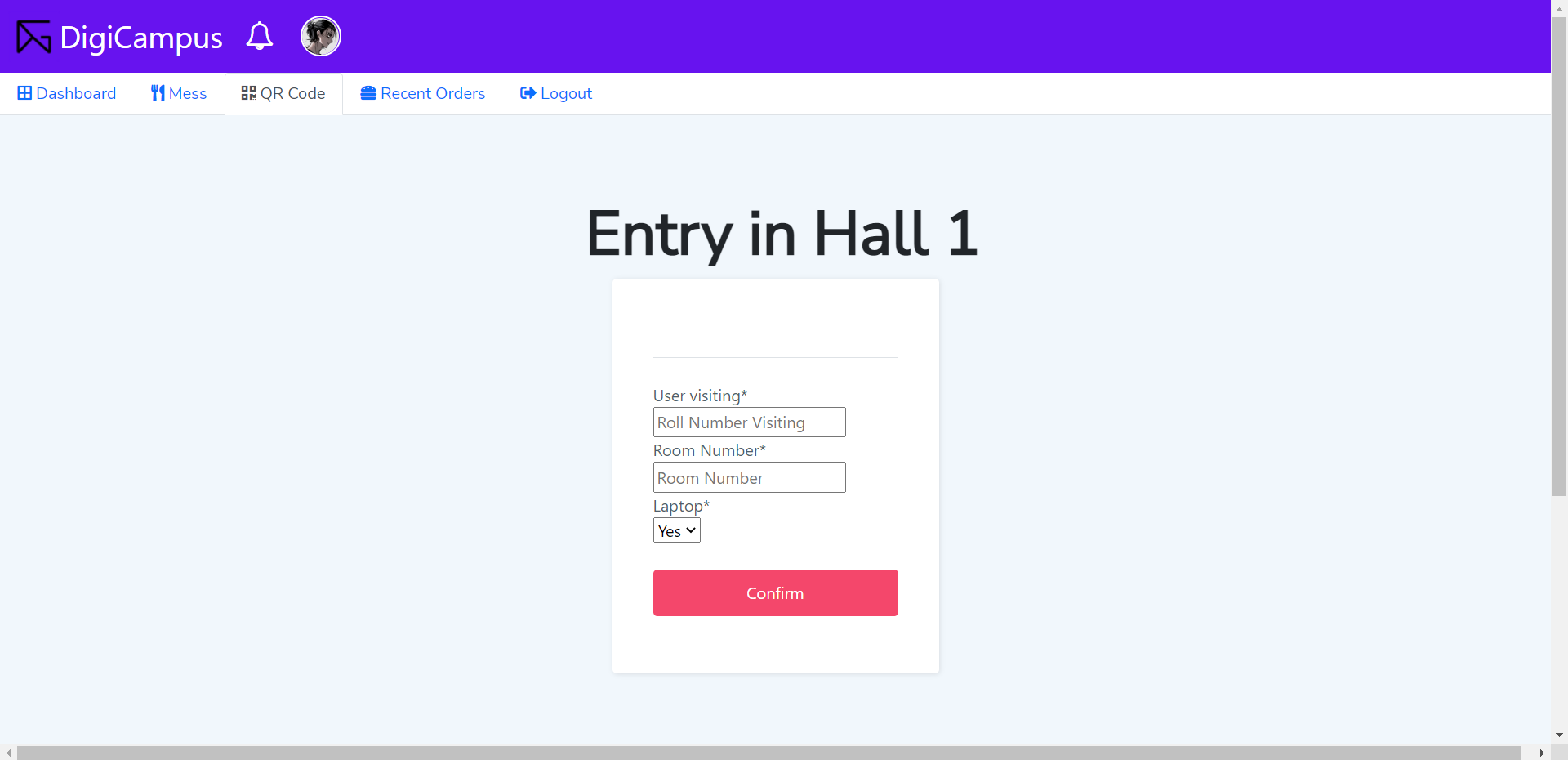
* **Student Profile:**



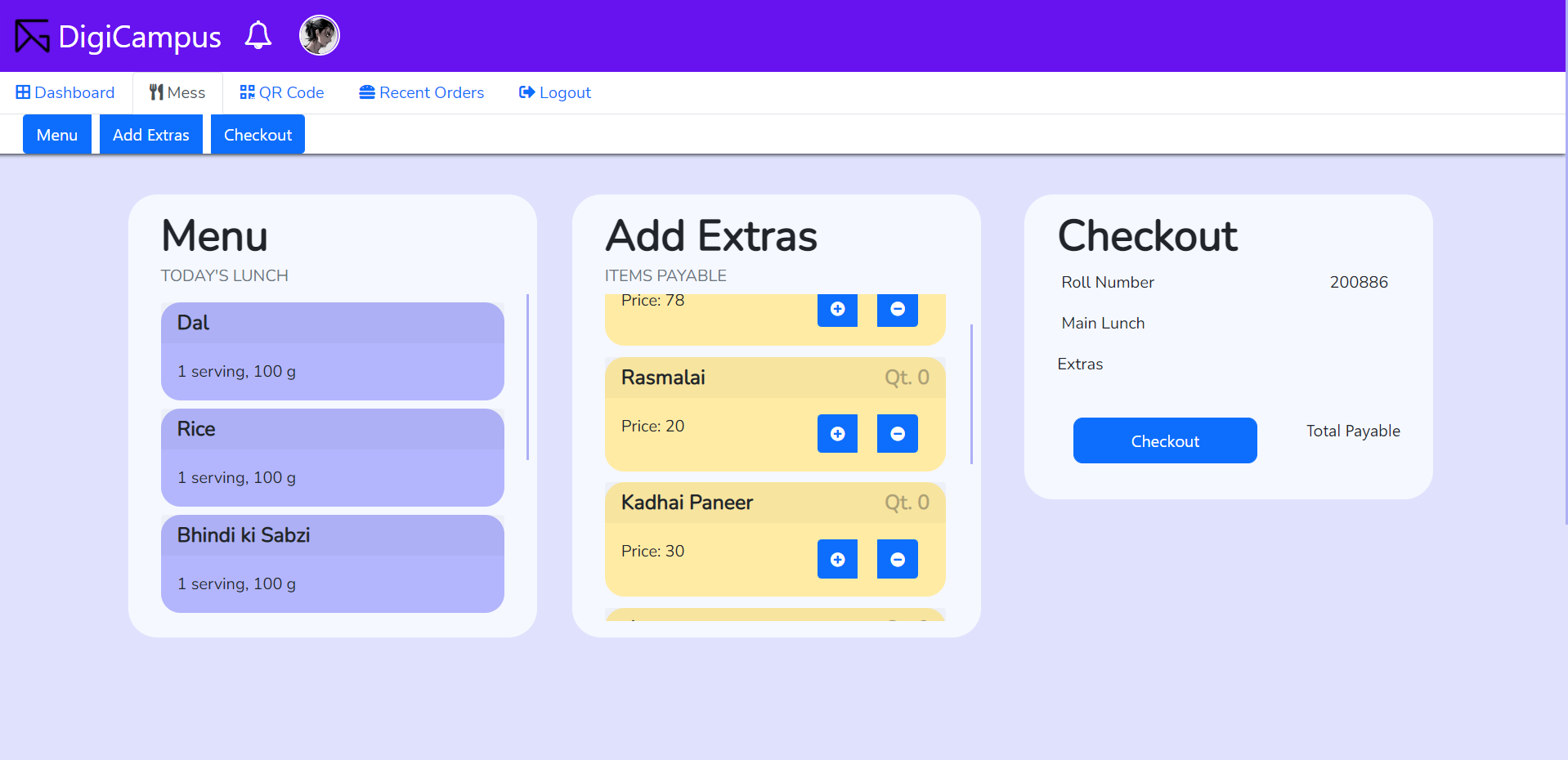
* **Scanning Code At The Gate**



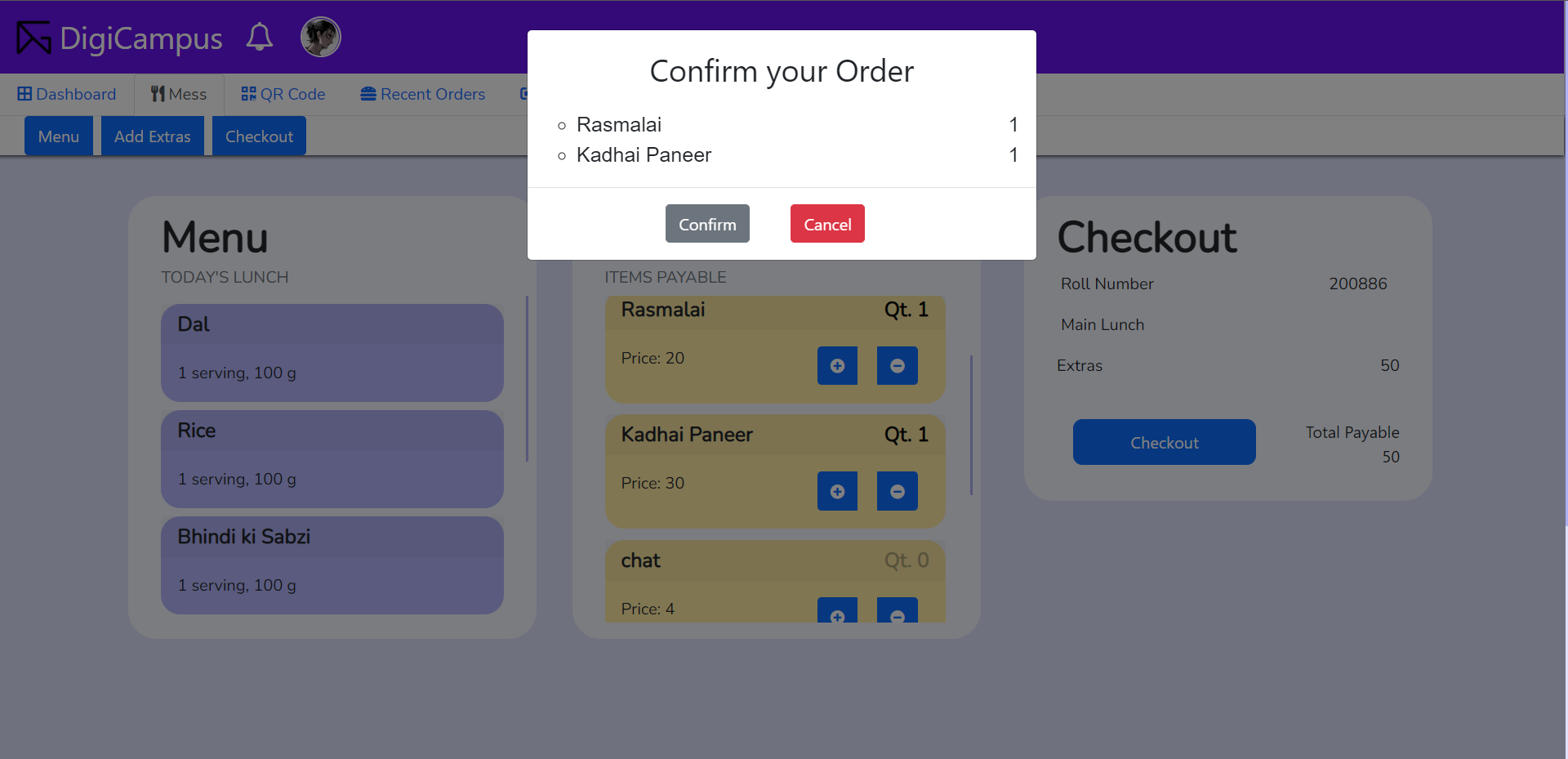
* **Entry/Exit Details**



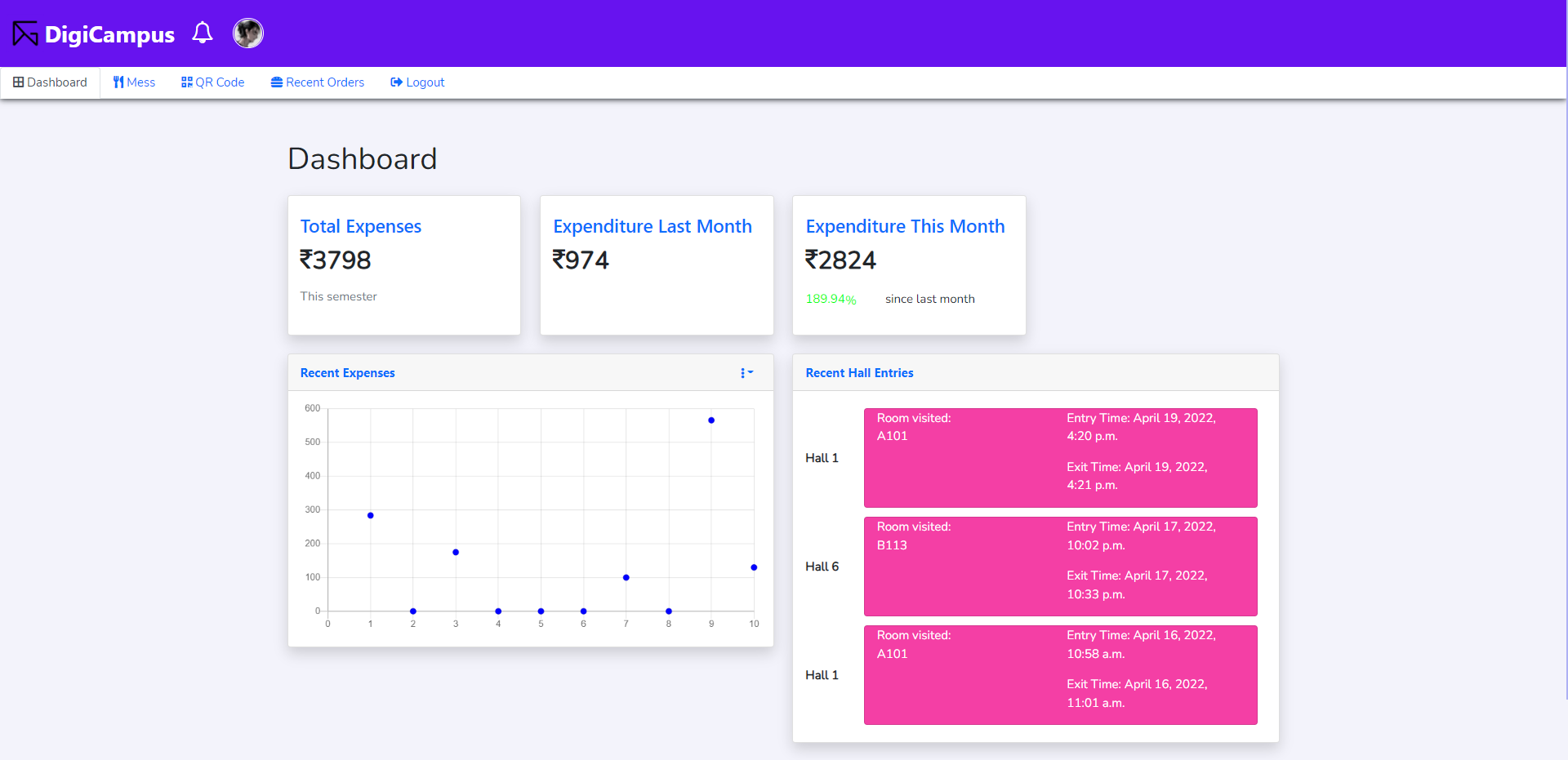
* **Mess Menu and Extras**



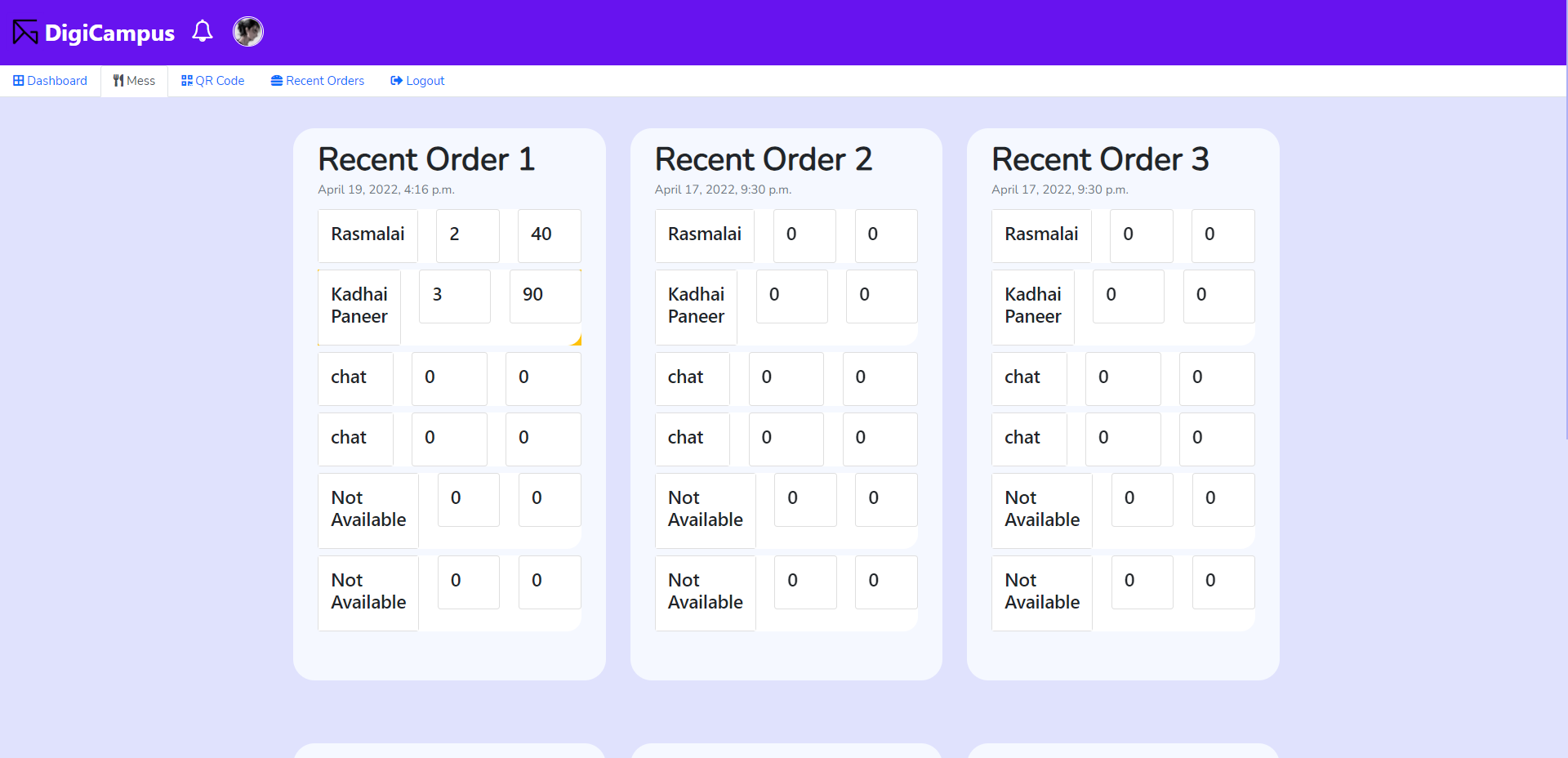
* **Confirmation Page**



* **Student Dashboard**



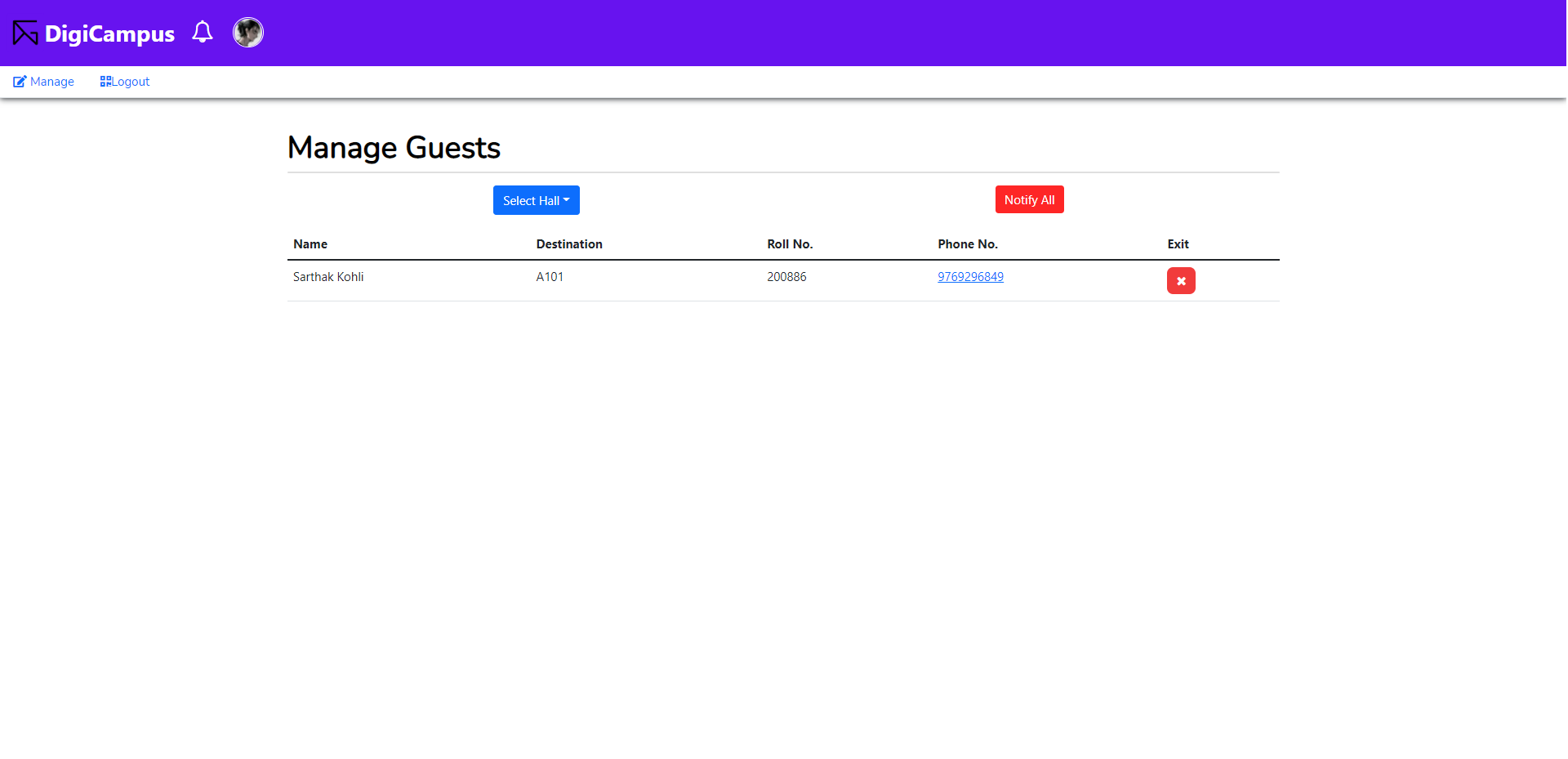
* **Order List**

****

**Mess Updation:**

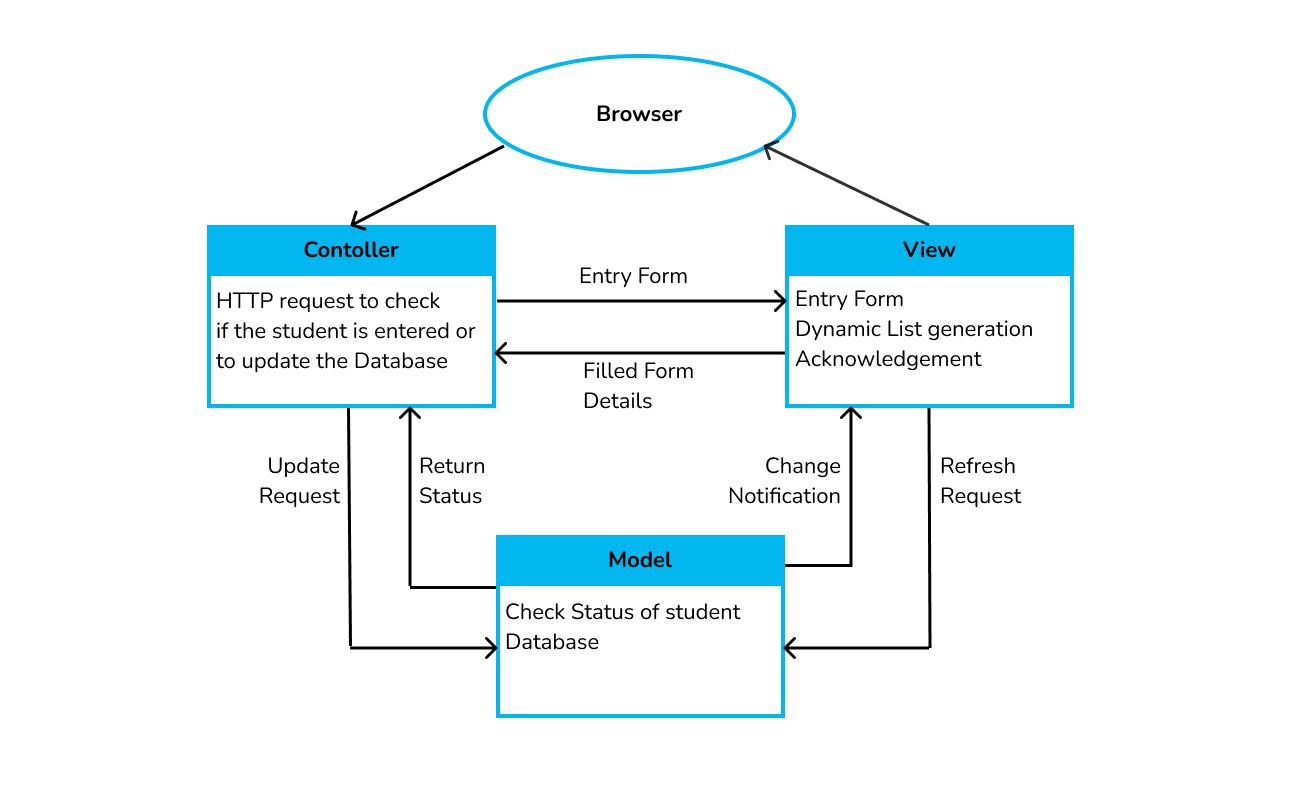


**Security Interface:**



# Architecture Design

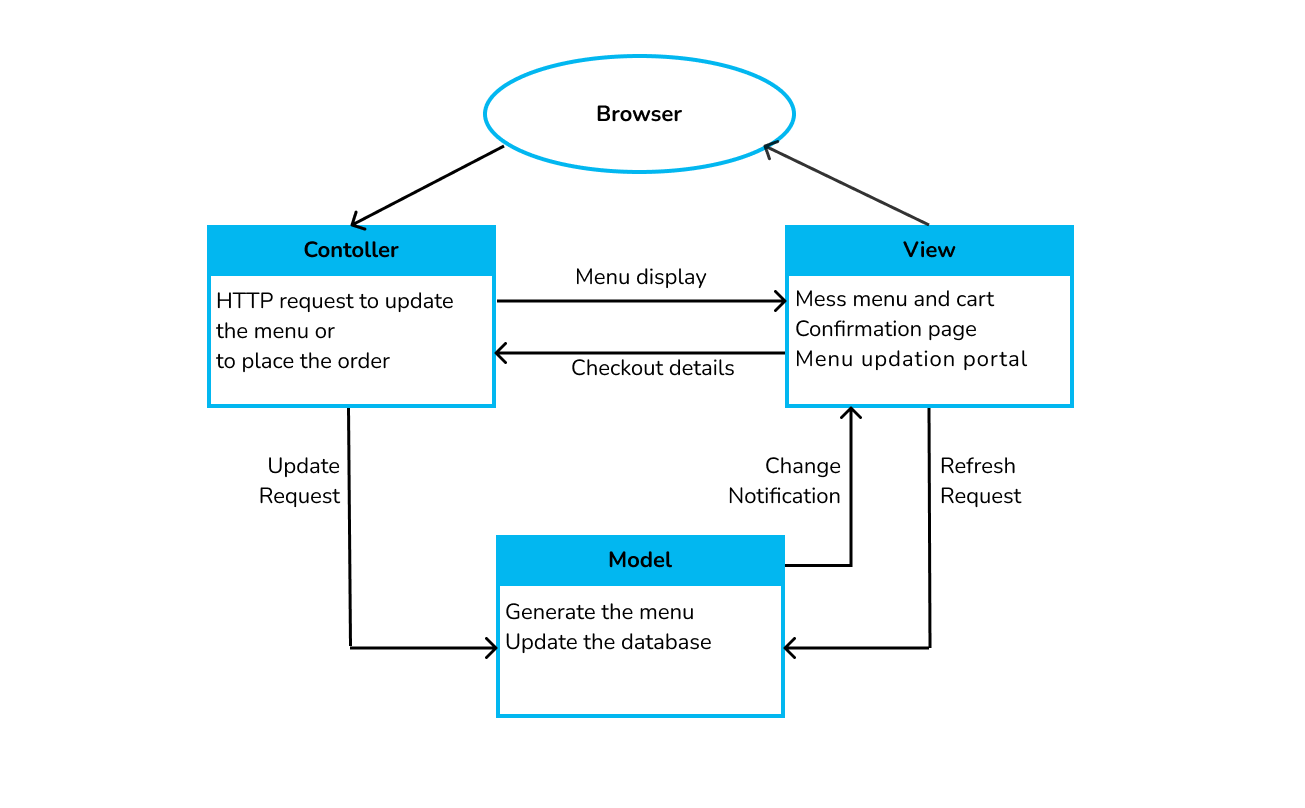
* **For Entry/Exit :**



The student scans the QR code **→** Control goes to the Controller which makes a request to check if the student is inside or outside **→** Request comes to the Model which checks for a student in the Database **→** If present, the student is given the exit acknowledgment and the dynamic database is manually updated **/** Else, Student is given entry form **→** Student fills entry form and details go to Controller which sends it to Model **→** Model updates the Student database marking student present inside.

The Dynamic List Generation makes it simpler for us to refresh the Student Database manually once the Entry/Exit of any student is done.

* **For Mess :**



Here we have two functionalities :

1. For Mess Manager: Mess manager sends an HTTP request to update the mess menu **→** Control goes to Model to Generate the menu **→** New menu is updated **→** Menu updation portal is displayed with an updated menu.
2. For Students: Student sends HTTP request to place an order **→** Control goes to view where mess menu and cart is displayed **→** Students place order in the cart and go to confirmation page **→** Change notification request is sent which updates student database **→** Checkout details is displayed to be shown at the mess.

# Object Oriented Design

## Use Case Diagrams

**Use Case Diagram for the entire System:**

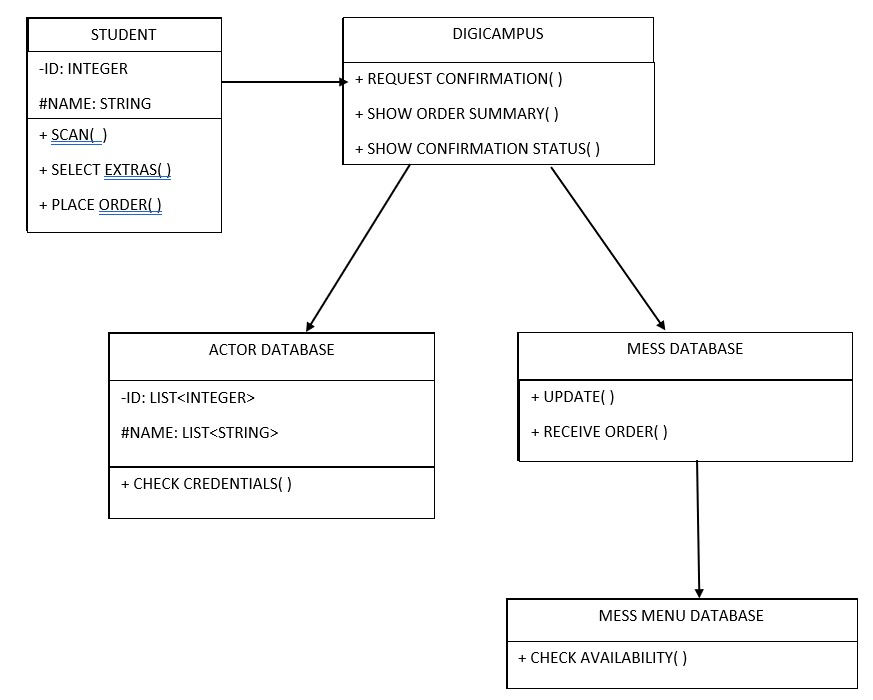
## 

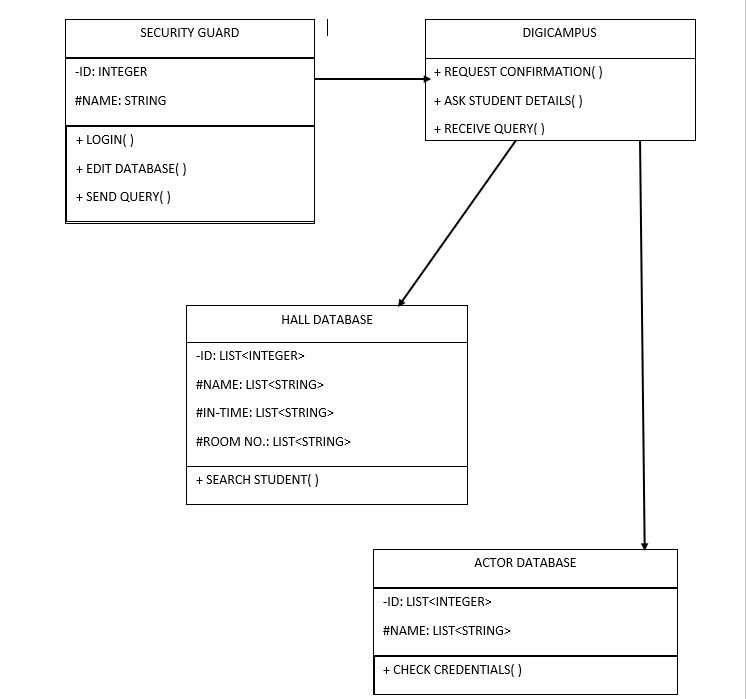
## 

**USE CASE DIAGRAMS FOR VARIOUS FUNCTIONALITIES:**

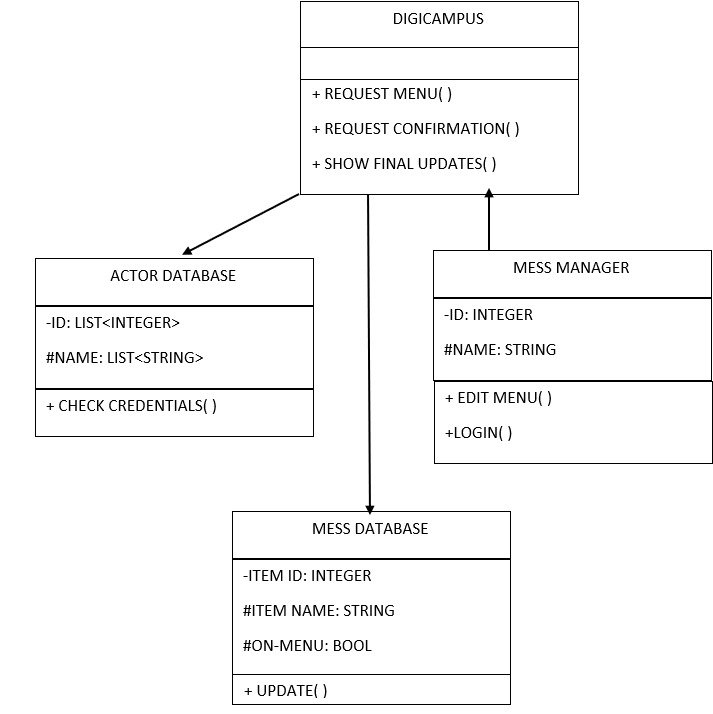
| USER CONFIGURATION |  |
| --- | --- |
| ENTERING A HALL |  |
| EXITING A HALL |  |
| NOTIFICATION |  |
| QUERY OF STUDENTS INSIDE THE HALL |  |
| ORDERING MESS EXTRAS |  |
| ORDER HISTORY |  |
| MENU UPDATION |  |
| ~~EDITING HALL ENTRY/EXIT DETAILS~~ |  |

## 3.2 Class Diagrams

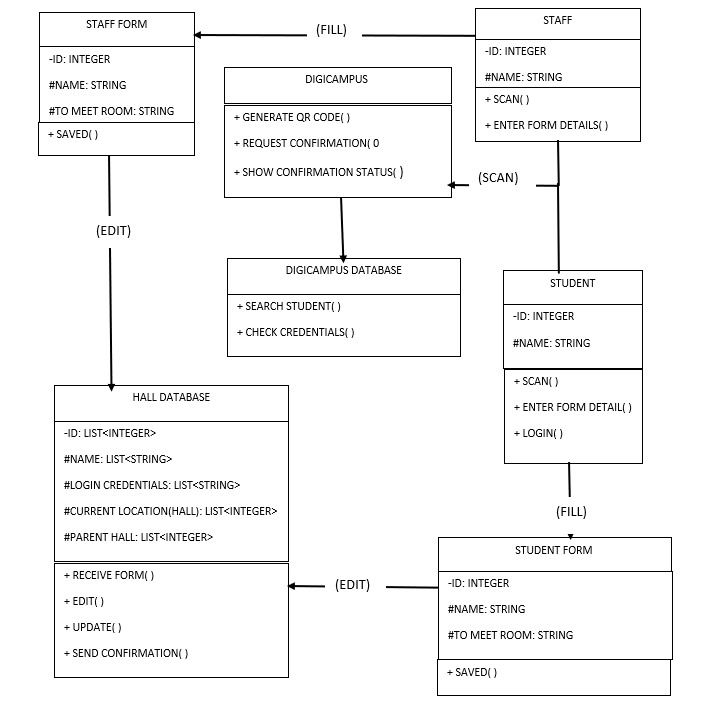
1. **FOR STUDENT:** 
2. **FOR SECURITY GUARDS:**



**(C) MENU EXTRAS SYSTEM:**

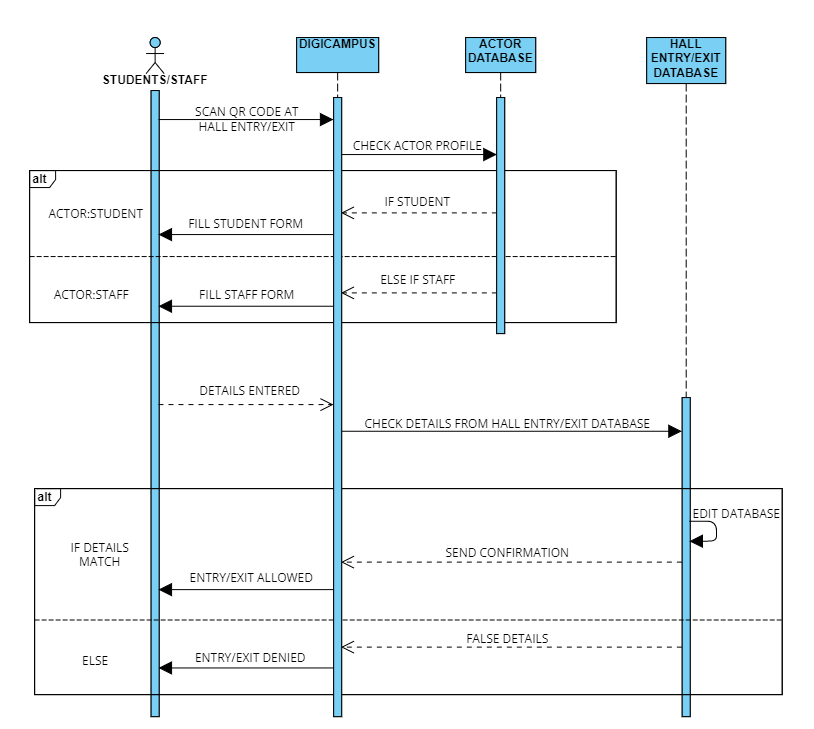


**(D) HALL ENTRY SYSTEM:**

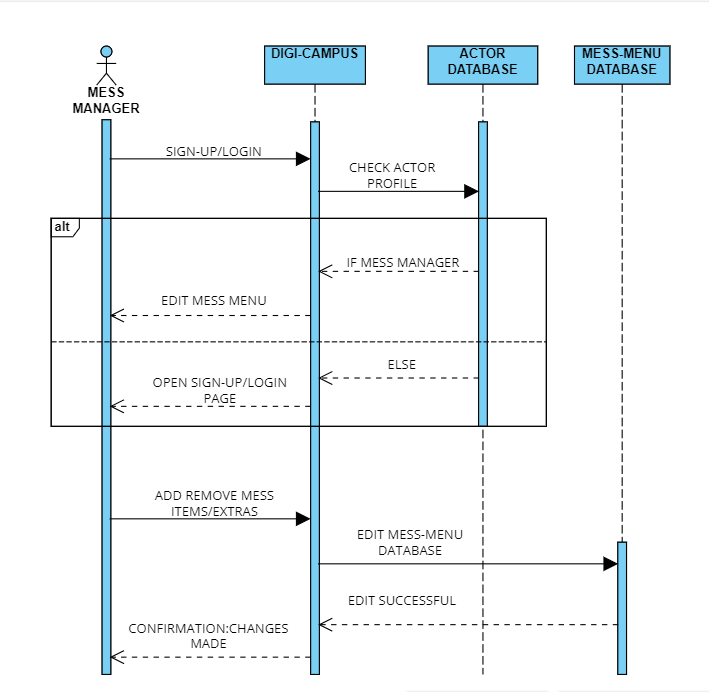


## 3.3 Sequence Diagrams

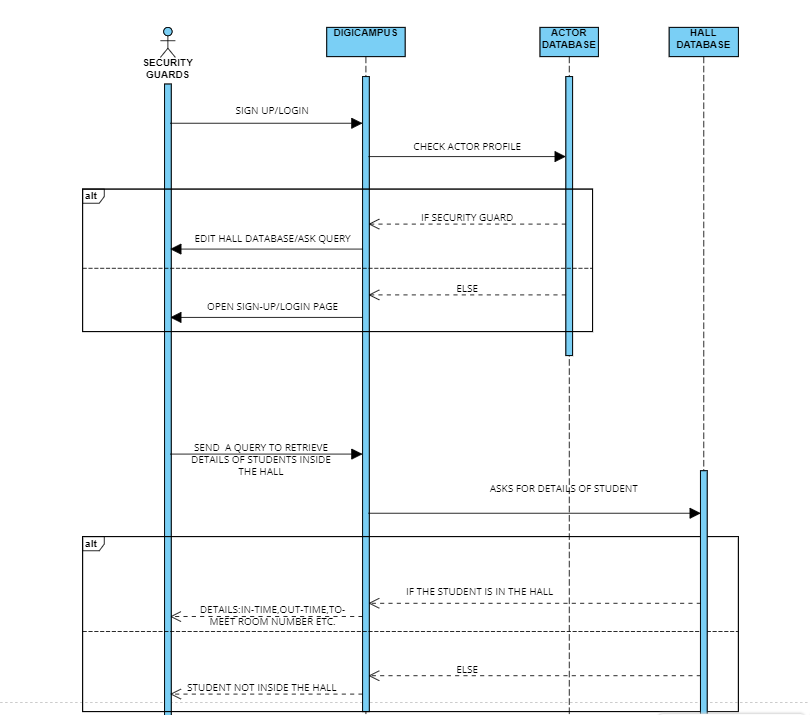
1. **ACTOR: STUDENT/STAFF-HALL/ENTRY-EXIT SYSTEM**



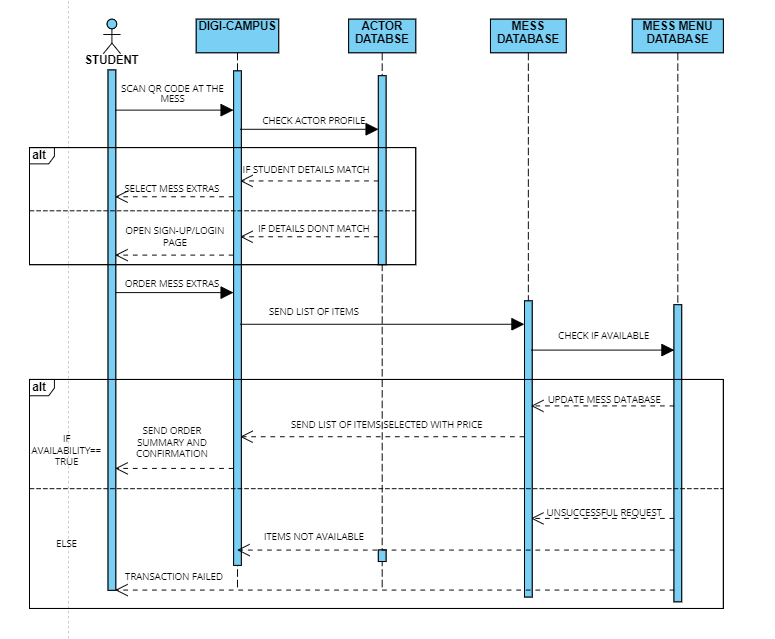
**(B) ACTOR: MANAGER**



**(C)ACTOR: SECURITY GUARDS**

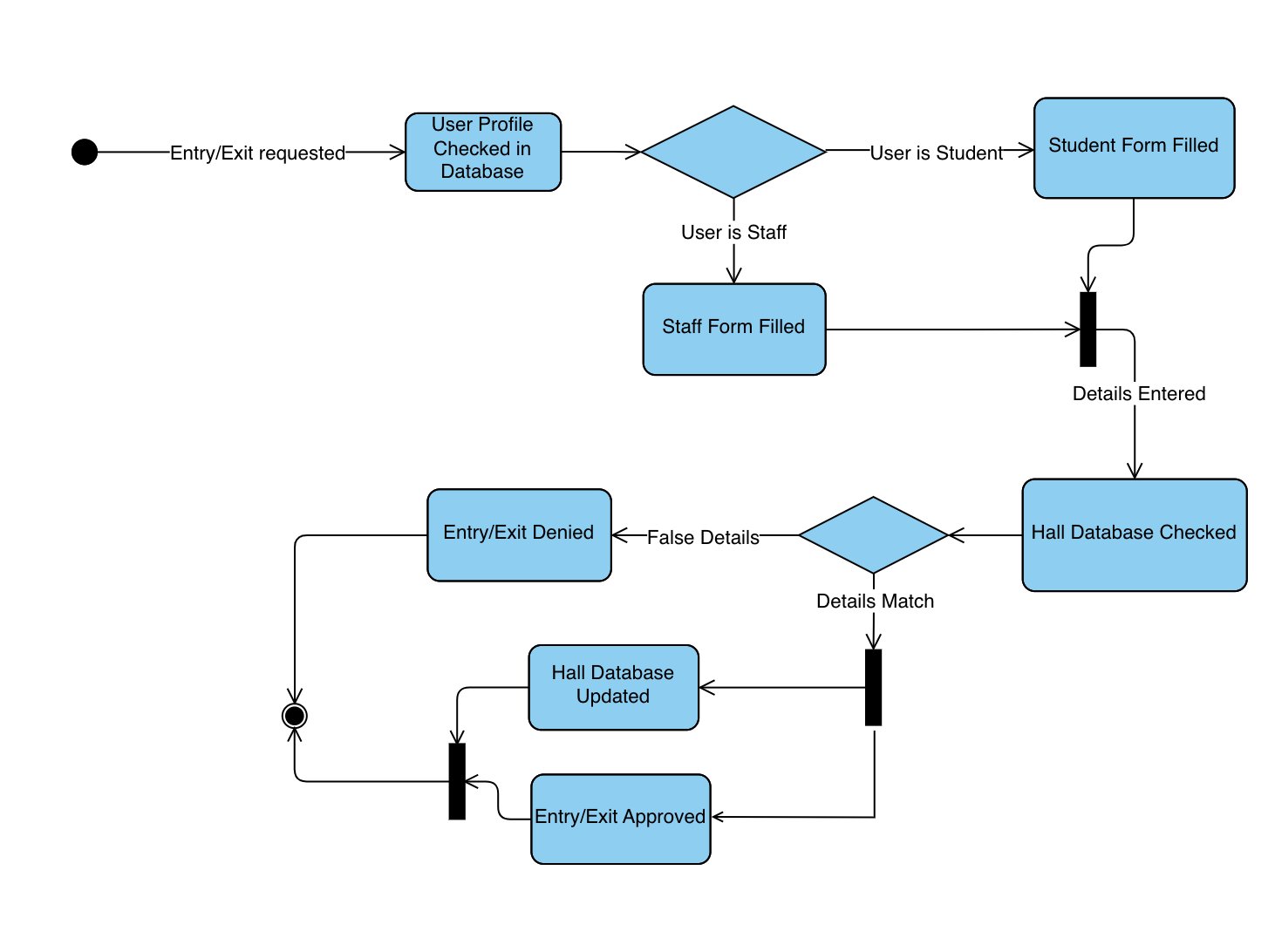


**D)ACTOR: STUDENTS-MESS EXTRAS SYSTEM**

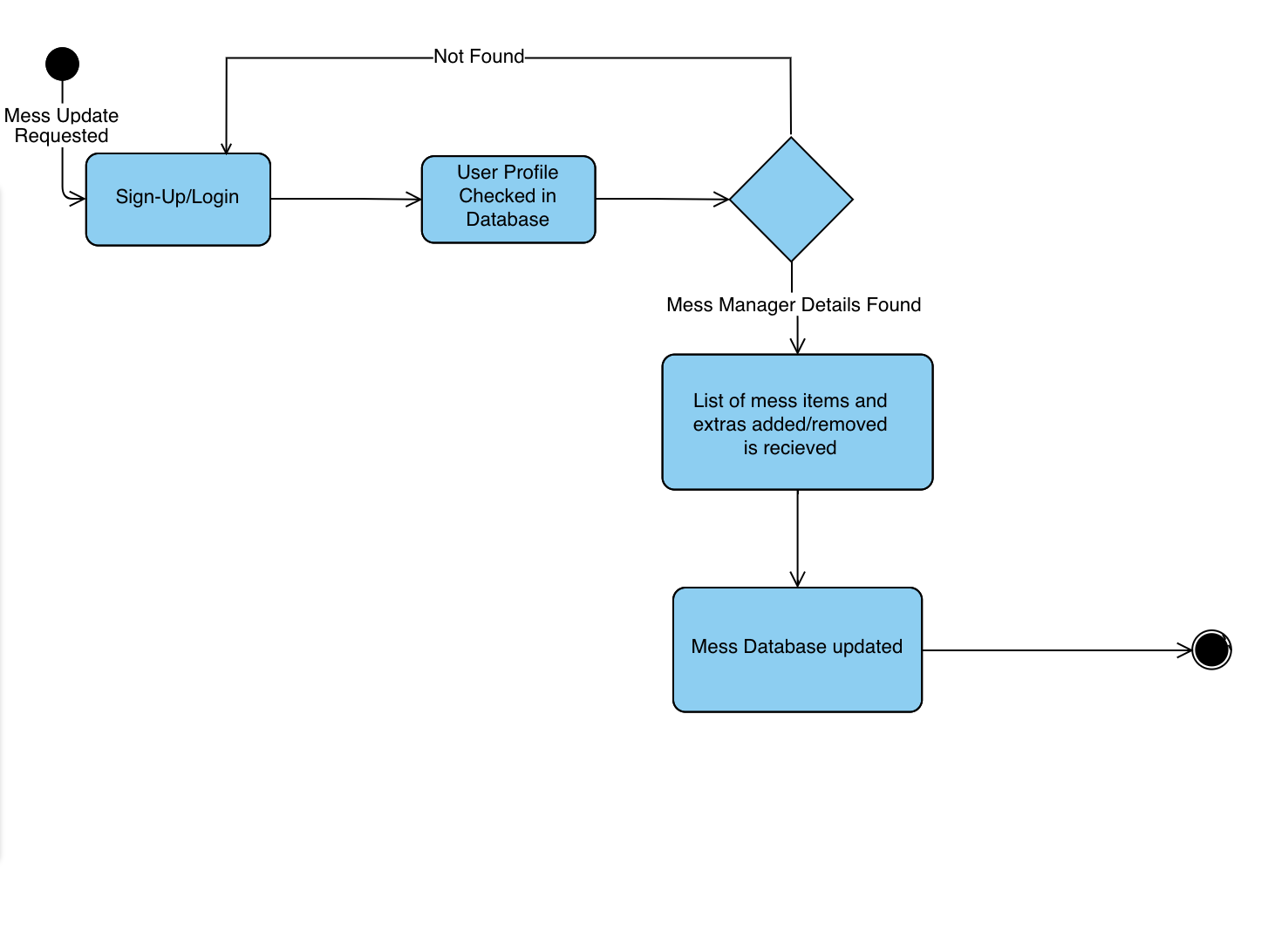
****

## 3.4 State Diagrams

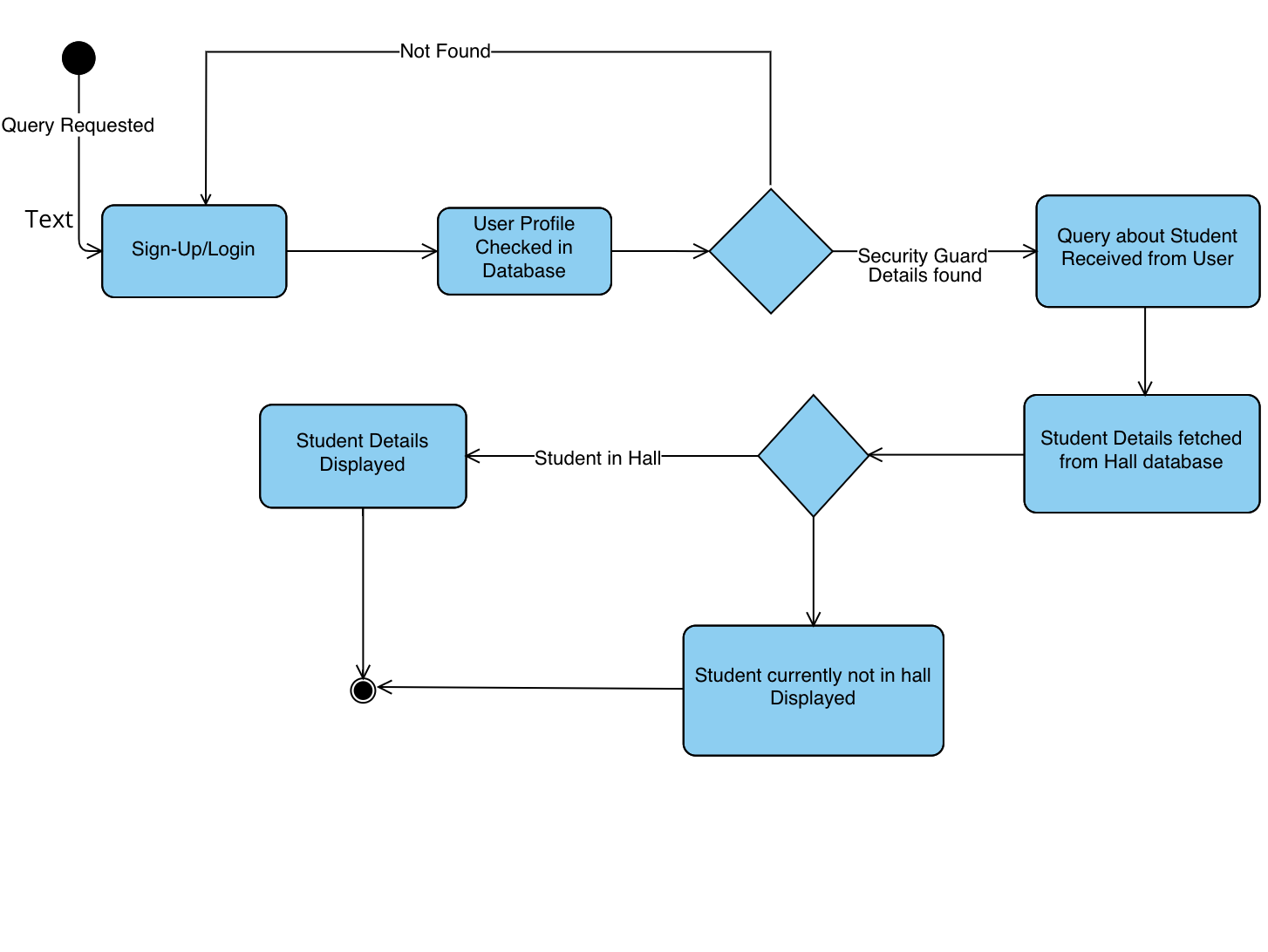
1. **ACTOR: STUDENT/STAFF-HALL/ENTRY-EXIT SYSTEM**



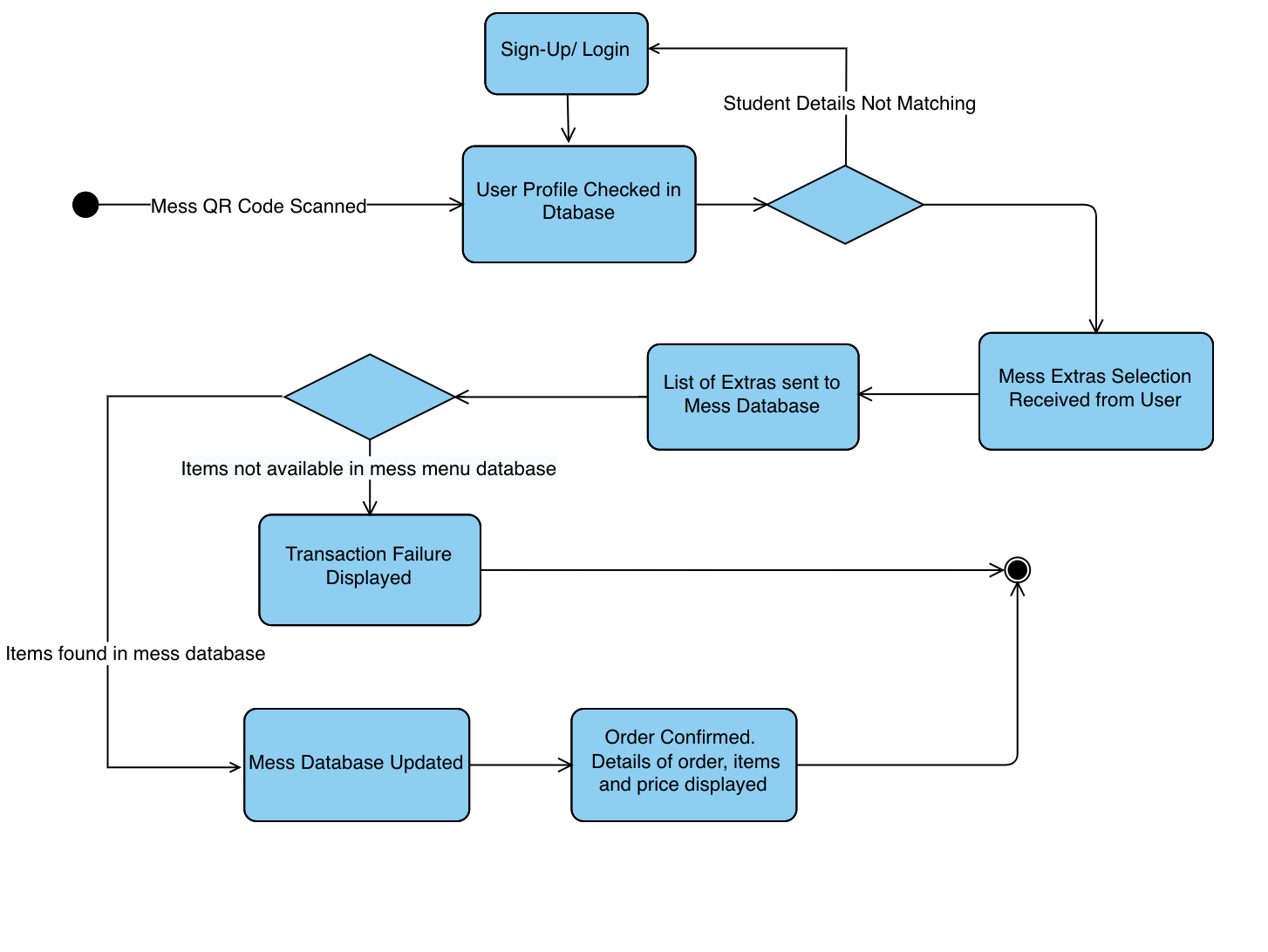
1. **ACTOR: MANAGER**



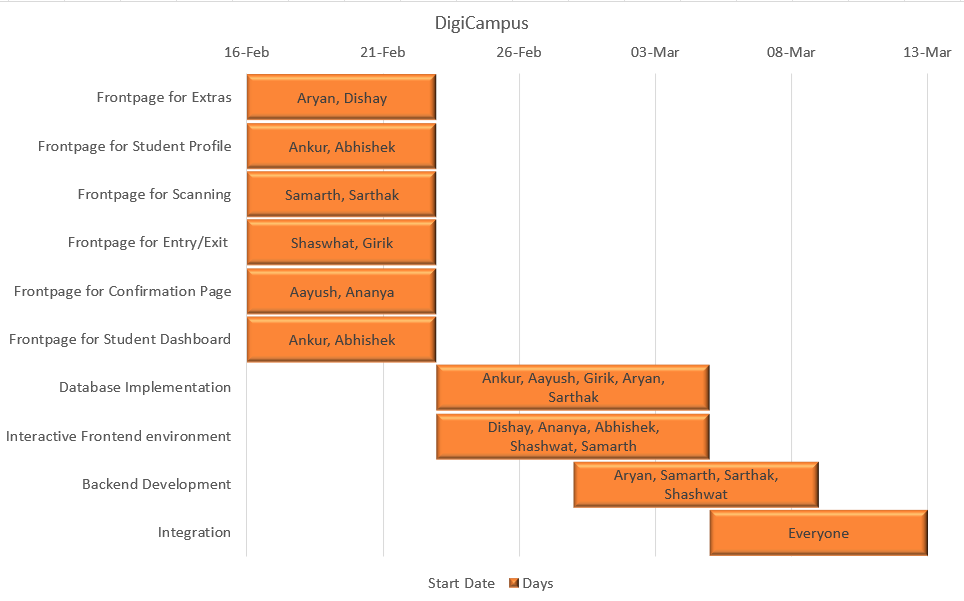
**C) ACTOR: SECURITY GUARDS**



**D)ACTOR: STUDENTS-MESS EXTRAS SYSTEM**



# Project Plan

**

**TEST PLAN FOR OUR SOFTWARE:**

* **ANALYZING THE PRODUCT:**
* Review the product documentation so as to be fully informed about the product details, target audience, and how the product will function on the ground level.
* If any point is unclear, we may interview customers or the target audience and make some changes in the software and documentation if they are needed at all.
* **TEST OBJECTIVES:**
* Finding as many software defects as possible.
* Creating bug-free software before release.
* **TEST STRATEGY:**
* **Defining the Scope of Testing:**

Defining clearly what things will be tested(hardware, software, integration) and metrics of testing.

* **Tentative priority of Testing Type:**

1. UNIT TESTING(will be done during the implementation phase as well)
2. INTEGRATION TESTING
3. SYSTEM TESTING
4. API TESTING
5. CODE IMPROVEMENT

* **Creating Test Logistics:**
* Preliminary tests will be done by our team **(Alpha testing)** in the priority as mentioned earlier.
* Preparation will be done for the beta testing of other groups’ projects.
* **Beta testing** will be done by peer groups.
* The Testing will be done in the tentative time frame **14th March,2022-10th April 2022.**
* **Test Criteria:**
* Tentatively, the team has decided to choose Exit Criteria as Test Criteria. However, this is subject to change as per the team’s requirements.
* **Test Deliverables:**
* The Team will submit a Beta Testing Report(of other group’s project) and address to Beta testing Feedback.
* The team will finally submit Final Project Report.

# Other Requirements

We will be using the IIT Kanpur servers and student databases for implementing the system.

The database can be imported from the Pingala servers.

**Appendix A - Group Log**

6th February: Aryan, Dishay, Sarthak, and Samarth meet to discuss the plan.

6th February: Groups divided to work on each section.

8th February: Aryan and Dishay work on Section 1.

9th February: Abhishek and Ankur completed Architecture Design.

10th February: Abhishek and Ankur completed Human Interface Design.

10th February: Aryan, Dishay, and Shashwat complete Section 1

12th February: Samarth and Sarthak finished Sequence Diagrams.

14th February: Aryan added Gantt Chart

15th February: Sarthak and Samarth made the test plan for the project.

15th February: Sarthak completed Use case diagrams.

15th February: Ananya completed Class diagrams.

15th February: Aayush completed state diagrams.

23rd April: Aryan updated to Version 2.0