**DELHI TECHNOLOGICAL UNIVERSITY**



**SE-405 SOFTWARE PROJECT MANAGEMENT**

**PROJECT REPORT**

**HOSTEL MANAGEMENT SYSTEM**

**Submitted To: Submitted By:**

**Ms. Kishwar Khan Deepak Kumar**

**2K18/SE/051**

**MOTIVATION**

Managing a hostel is a tedious task that needs a lot of supervision and can be time consuming. The hostel management system is a dynamic as well as practical approach as it makes managing the boarding and other related facilities a lot easier. **One of the key benefits** of a hostel management system is that it reduces the burden on the administration staff and simplifies roles and responsibilities as most of the manual tasks and mundane paperwork can be done through the online system.

A hostel management system ensures that record management and flawless execution of administrative tasks such as managing hostel applications, registration forms, allotment of rooms, fees management, mess payment management, complain issues and easy communication with students.  The hostel administration team can access all the hostel data and manage the workflow from their device at all times – ensuring that updates are made in real time.

It also helps securely store and manage the data of the students, allowing access to a few authorized staff members only.

**Another important benefit** of implementing such a system is that it brings in the transparency in the entire management which helps to develop a trust between the students and the management.

Additionally a hostel that runs smoothly without any issues or errors helps the educational institute in building a stronger brand for itself. It is important for institutes for understand that is not managed well can actually deter students from taking admissions in that institute.

**OBJECTIVE**

Hostel management system is an android application which aims to computerize and simplify the hostel management system. Currently the process involves student filling up the hostel forms for allocation and hand-written applications for specific purposes and submitting them in respective hostel offices, thus making the process very inefficient.

The main benefits of the hostel management system would be:-

1. It would be less effort and time consuming.
2. It automates mundane task.
3. It secures the data of the students.
4. It makes the process of updating data very fast.
5. It reduces the manual work of management.
6. It enhances the reputation of the educational institute.

The app would mainly constitute two types of Users

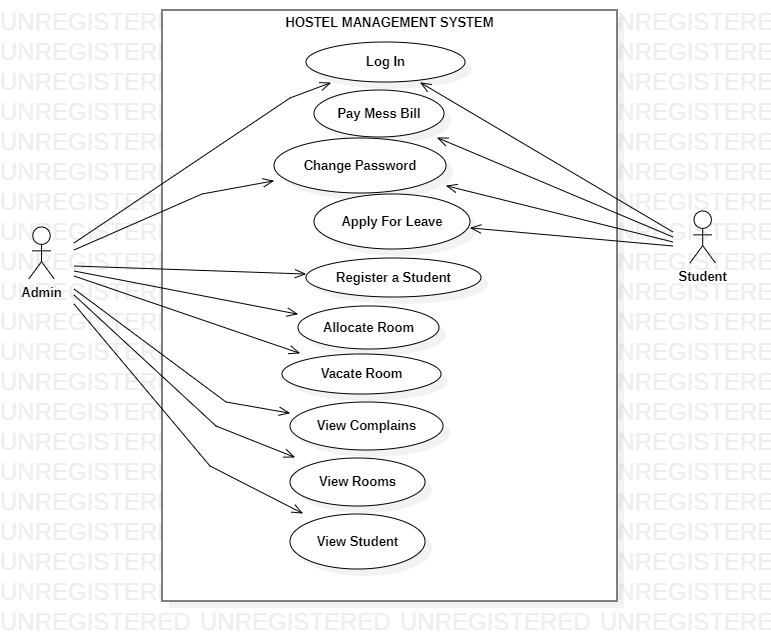
1. Admin or the Hostel office staff
2. Student

Features that are present in the app are as follows:

1. Student
2. Log In
3. Pay Mess Fee
4. Apply for leave
5. Complain about or ask for any specific requirement
6. Change Password
7. Admin
8. Register a Student
9. Allocate Room
10. Vacate Room
11. View Student
12. View Status of Rooms
13. View Complains

**METHODOLGY**

**THE USE CASE DIAGRAM OF THE PROJECT IS AS FOLLOWS:**



**SOFTWARE DEVELOPMENT LIFE CYCLE MODEL**

Software Development Life Cycle Model (SDLC) is a process followed for a software project, within a software organization. It consists of a detailed plan describing how to develop, maintain, replace and alter or enhance specific software. The life cycle defines a methodology for improving the quality of software and the overall development process.

The software development life cycle consists 6 phases which are:-

1. Planning
2. Defining
3. Designing
4. Building
5. Testing
6. Deployment

The SDLC that has been used in the model is Iterative Model.

In the iterative model, iterative process starts with a simple implementation of a small set of software requirements and iteratively enhances the evolving versions until the complete system is implemented and ready to be deployed.

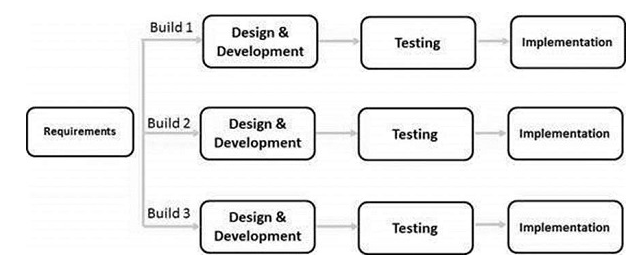
An iterative life cycle model does not attempt to start with a full specification of requirements. Instead development begins only by specifying and implementing just a part of software, which is then reviewed to identify further requirements.

This process is then repeated producing a new version of the software at the end of each iteration of the model.

**Design of Model**

Iterative process starts with a simple implementation of a subset of the software requirements and iteratively enhances the evolving versions until the full system is implemented. At each iteration design modifications are made and new functional capabilities are added. The basic idea behind this method is to develop a system through repeated cycles (iterative) and in smaller portions at a time (incremental).

Illustration of the model:



Iterative and Incremental development is a combination of both iterative design or iterative method and incremental build model for development. "During software development, more than one iteration of the software development cycle may be in progress at the same time." This process may be described as an "evolutionary acquisition" or "incremental build" approach."

In this incremental model, the whole requirement is divided into various builds. During each iteration, the development module goes through the requirements, design, implementation and testing phases. Each subsequent release of the module adds function to the previous release. The process continues till the complete system is ready as per the requirement.

The key to a successful use of an iterative software development lifecycle is rigorous validation of requirements, and verification & testing of each version of the software against those requirements within each cycle of the model. As the software evolves through successive cycles, tests must be repeated and extended to verify each version of the software.

**Iterative Model Pros and Cons**

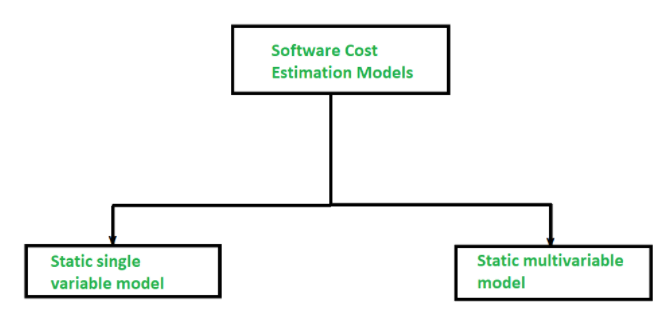
The advantage of this model is that there is a working model of the system at a very early stage of development, which makes it easier to find functional or design flaws. Finding issues at an early stage of development enables to take corrective measures in a limited budget.

The disadvantage with this SDLC model is that it is applicable only to large and bulky software development projects. This is because it is hard to break a small software system into further small serviceable increments/modules.

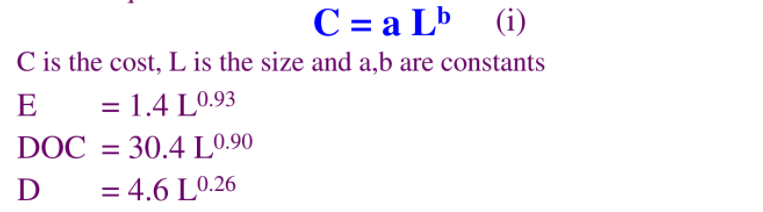
**COST ESTIMATION**

It is important to know how much it will cost in developing and how much development time it will take to complete. These evaluations are needed before development is started and conveyed to the team.

The two major models used in estimating the cost of a project are as ::



In a static model, a single variable is grabbed as a key element for calculating the cost and effort whereas, In a dynamic model, all variables are connected with each other, and there is no primary variable.



The calculation for the project are done as follows :

* **KLOC =** 7.53 KLOC
* **Effort =** 1.4 \*( 7.53)^0.93

           =  9.152 Person/Months

* **DOC(Documentation) =** 30.4\*(7.53)^0.90

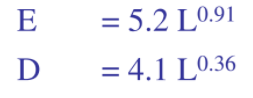
          = 198.743 Number of Pages

* **D(Duration)   =** 4.6\*(7.53)^0.26

       = 7.77 Months

These models are also known as multivariable models. This model is often based on the first equation and actually depends on several variables representing different aspects of the software development environment.

Equations are:



The calculation for the project are done as follows :

* Effort = 5.2\* (7.53)^0.91

           = 32.65

* D = 4.1\*(7.53)^0.36

            = 8.48

**THE MAIN TECHNOLOGIES THAT HAVE BEEN USED ARE AS FOLLOWS:**

1. **Frontend:**
2. **Java**
3. Java is a high-level, class-based, object-oriented programming language that is designed to have as few implementation dependencies as possible. It is a general-purpose programming language intended to let programmers *write once, run anywhere* (WORA), meaning that compiled Java code can run on all platforms that support Java without the need for recompilation.
4. The Java language is a key pillar in Android, an open source mobile operating system. Although Android, built on the Linux kernel, is written largely in C, the Android SDK uses the Java language as the basis for Android applications but does not use any of its standard GUI, SE, ME or other established Java standards.[78]
5. **Android Studio:**
6. **Android Studio** is the officialintegrated development environment (IDE) for Google's Android operating system, built on Jet Brains' IntelliJ IDEA software and designed specifically for Android development. It is available for download on Windows, macOS and Linux based operating systems or as a subscription-based service in 2020. It is a replacement for the Eclipse Android Development Tools (E-ADT) as the primary IDE for native Android application development.
7. **Backend**
8. **Firebase**
9. **Firebase** is a Backend-as-a-Service (Baas). It provides developers with a variety of tools and services to help them develop quality apps, grow their user base, and earn profit. It is built on Google’s infrastructure.
10. Firebase is categorized as a NoSQL database program, which stores data in JSON-like documents.
11. **Key Features of Firebase**
12. Authentication: It supports authentication using passwords, phone numbers, Google, Facebook, Twitter, and more. The Firebase Authentication (SDK) can be used to manually integrate one or more sign-in methods into an app.
13. Real-time database: Data is synced across all clients in real time and remains available even when an app goes offline.
14. Hosting: Firebase Hosting provides fast hosting for a web app; content is cached into content delivery networks worldwide.
15. Test lab: The application is tested on virtual and physical devices located in Google’s data centers.
16. Notifications: Notifications can be sent with firebase with no additional coding.

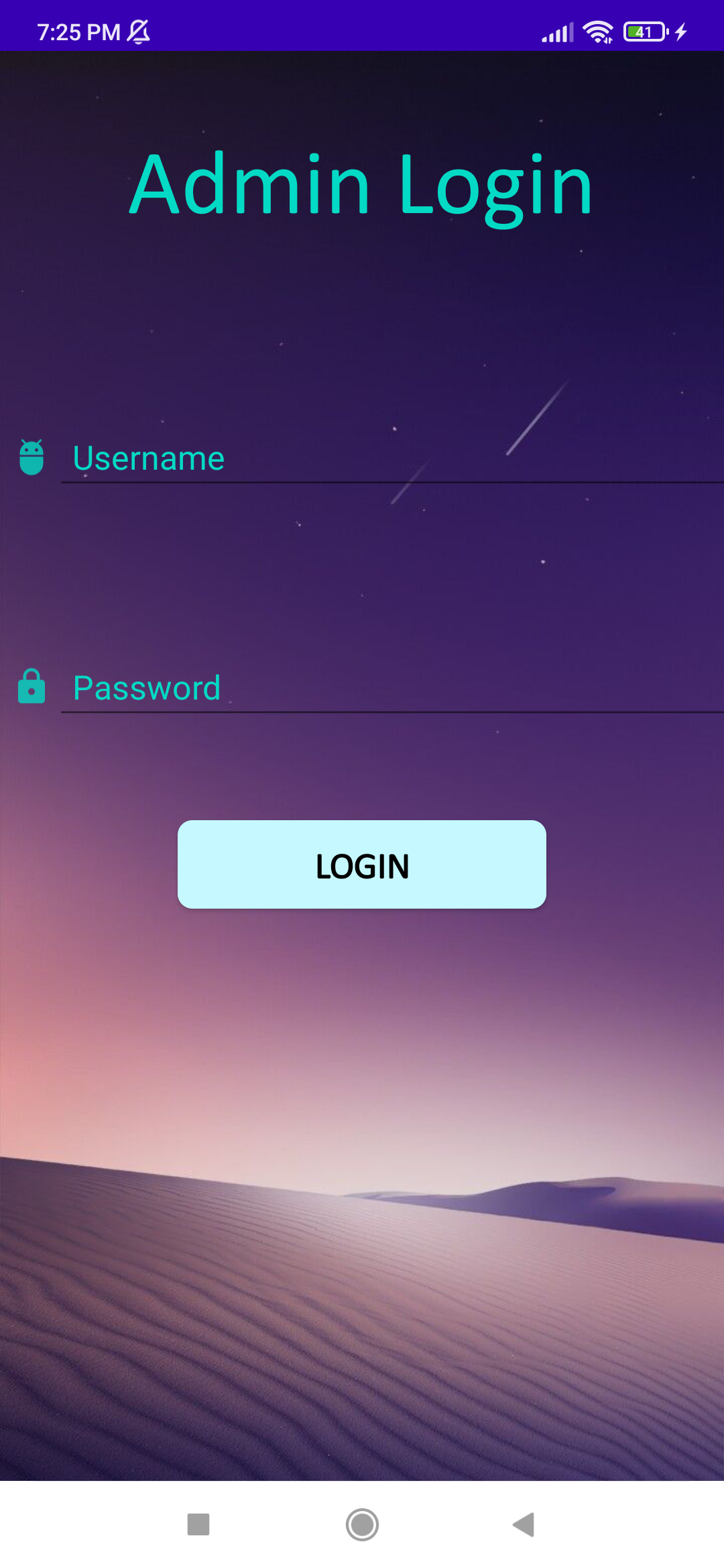
**FUNCTIONALITIES OF THE APP**

1. **STARTING HOME PAGE**

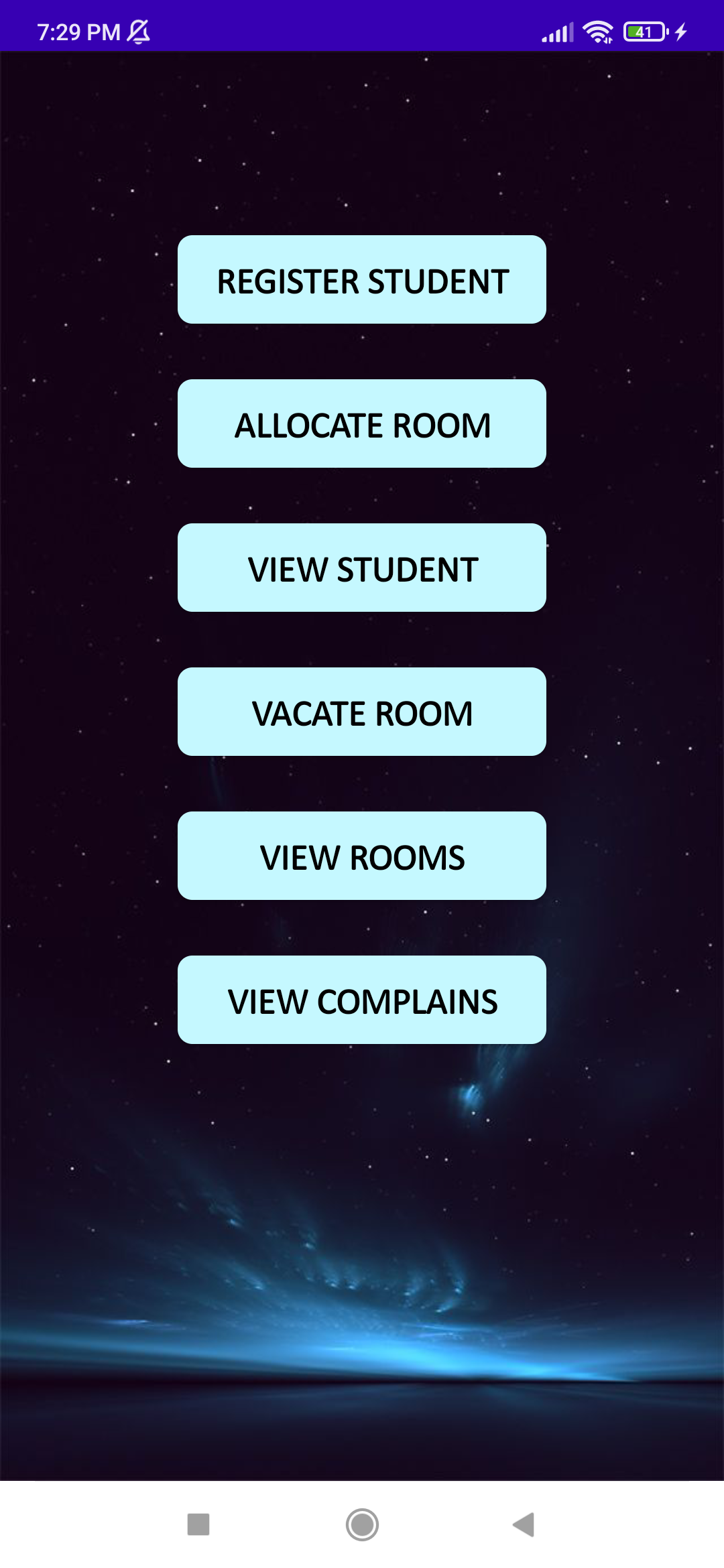


* 1. This is the starting home page, here there are two ways to login either student login or admin login.

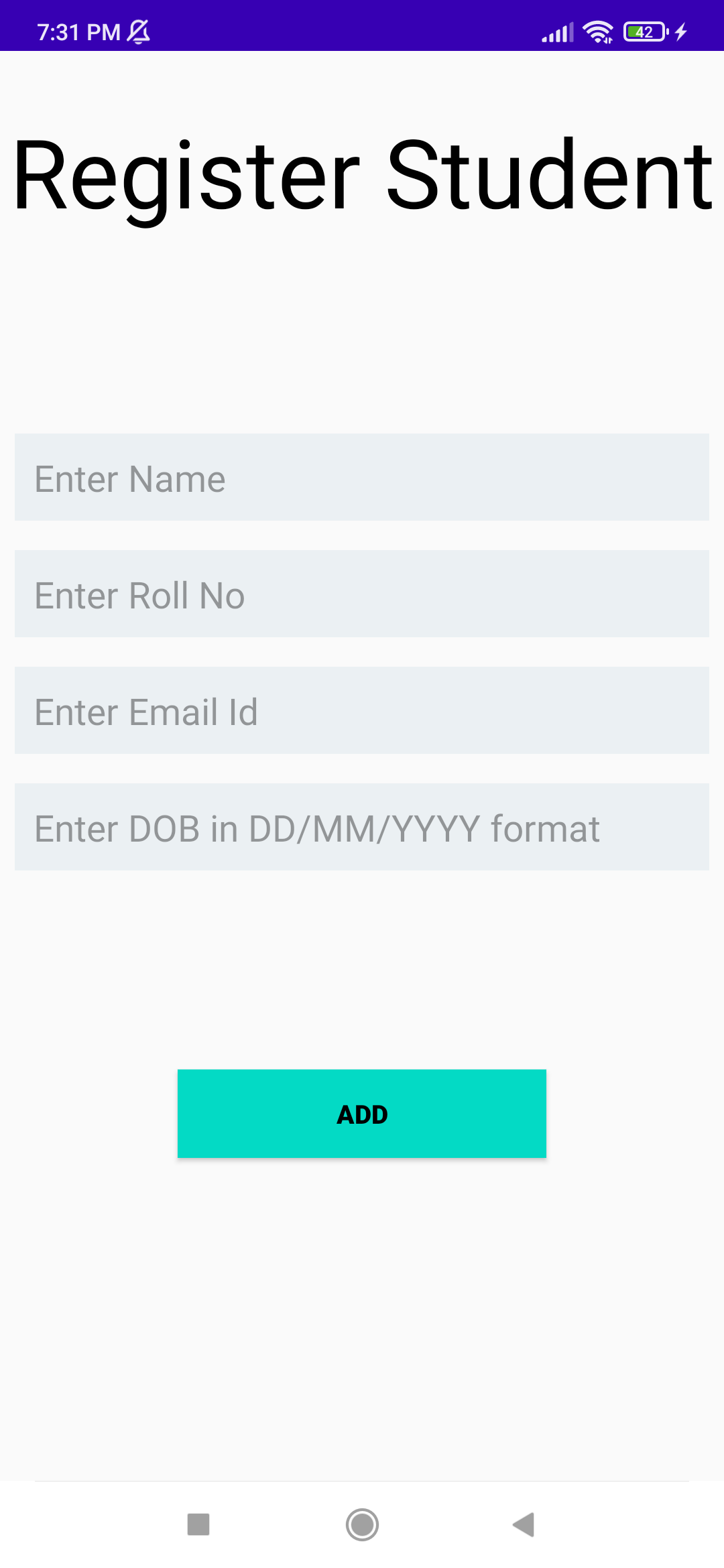
1. **ADMIN LOGIN**



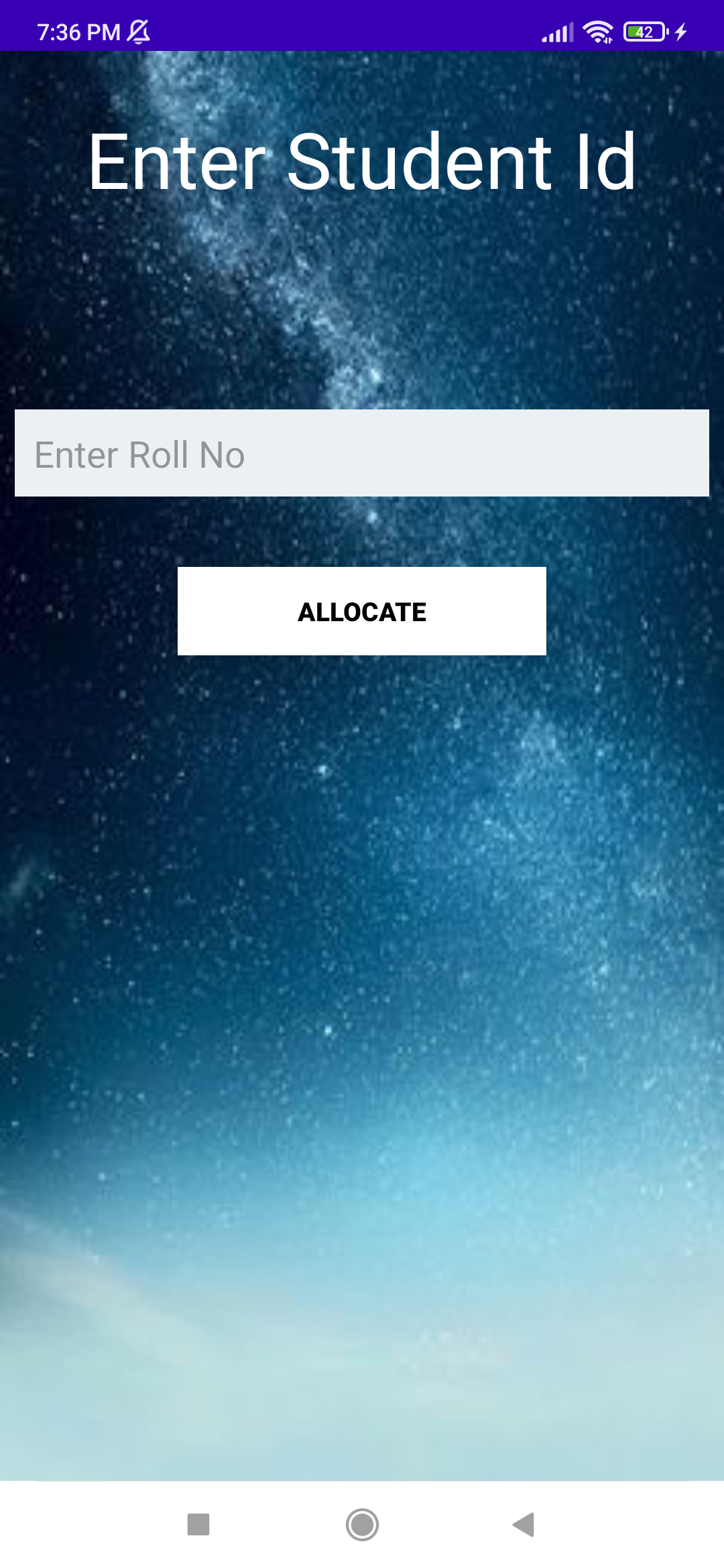
1. The username and password required for admin to login have to be manually added in the database.
2. If the username and password is correct, “Login Successful” is displayed on the screen else “Invalid Login Id or Password” in other case.
3. **ADMIN HOME**



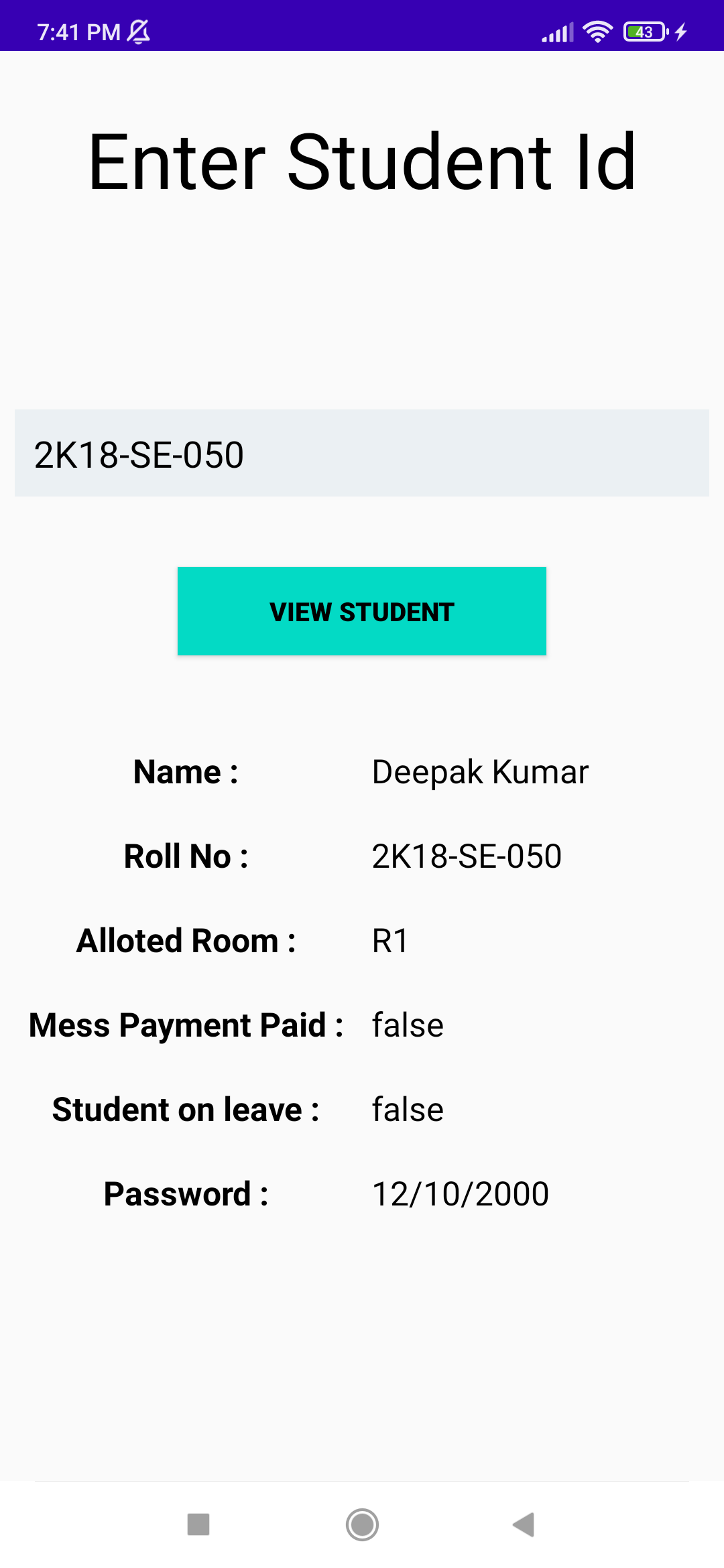
1. After a successful login, the admin home has six different functionalities which are, Register Student, Allocate Room, View Student, Vacate Room, View Rooms and View Complains.
2. **REGISTER STUDENT**



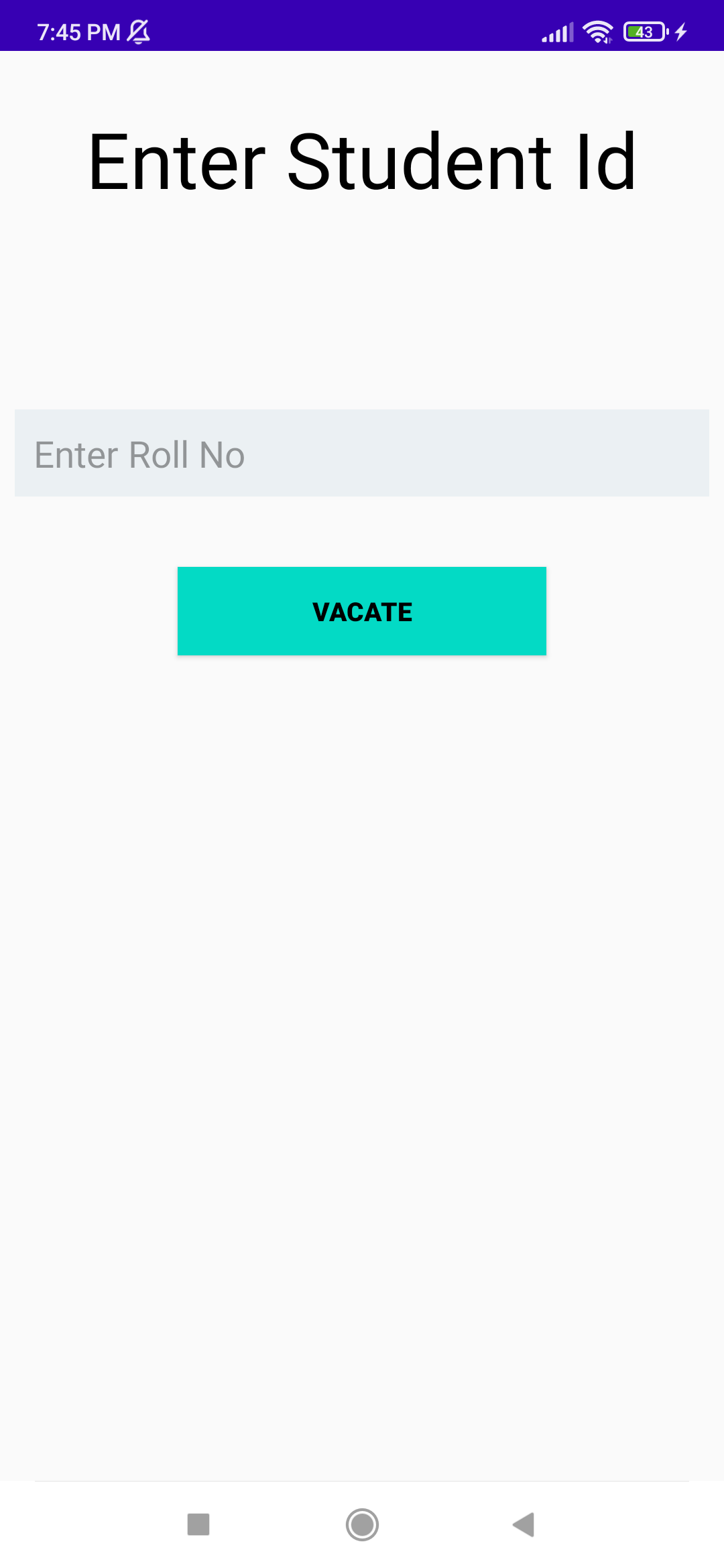
1. For Registering a Student the admin has to enter 4 different details of the student which are the Name, Roll No or the Student Id, the email Id ad the DOB.
2. The name cannot contain any digits.
3. The email should contain ‘@’ and ‘.com’
4. The DOB entered should be in the prescribed format
5. When the student is initially registered, the password is set as the students DOB, room Allotted as -1 which is no room allotted, on Leave as false and Paid Mess Bill as false.
6. **ALLOCATE ROOM**



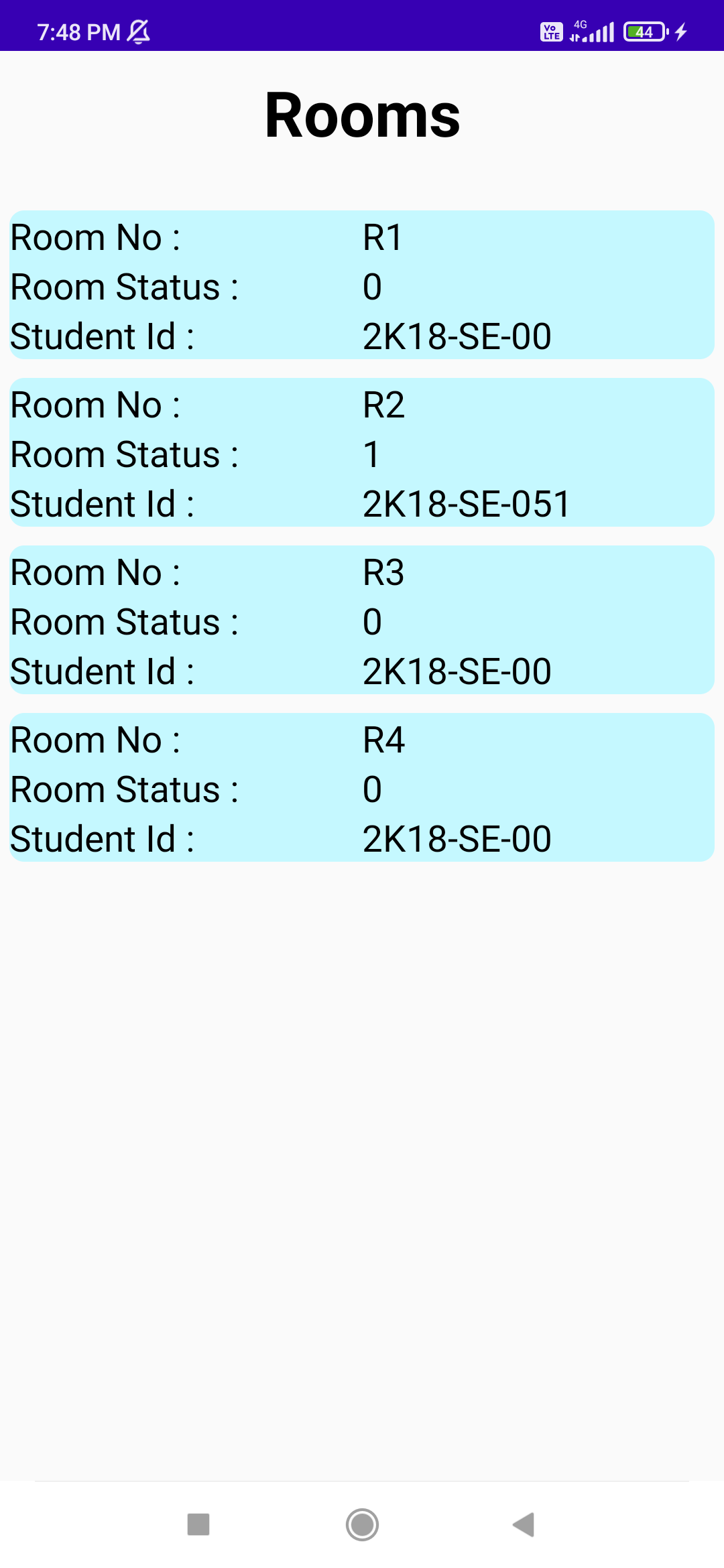
1. Enter the Student Id to allocate the room.
2. If the student is not registered the student will show “Student not registered”.
3. The room is already allotted the app will show “Room already allotted” else the first empty room which is present will be allotted to the student.
4. **VIEW STUDENT**



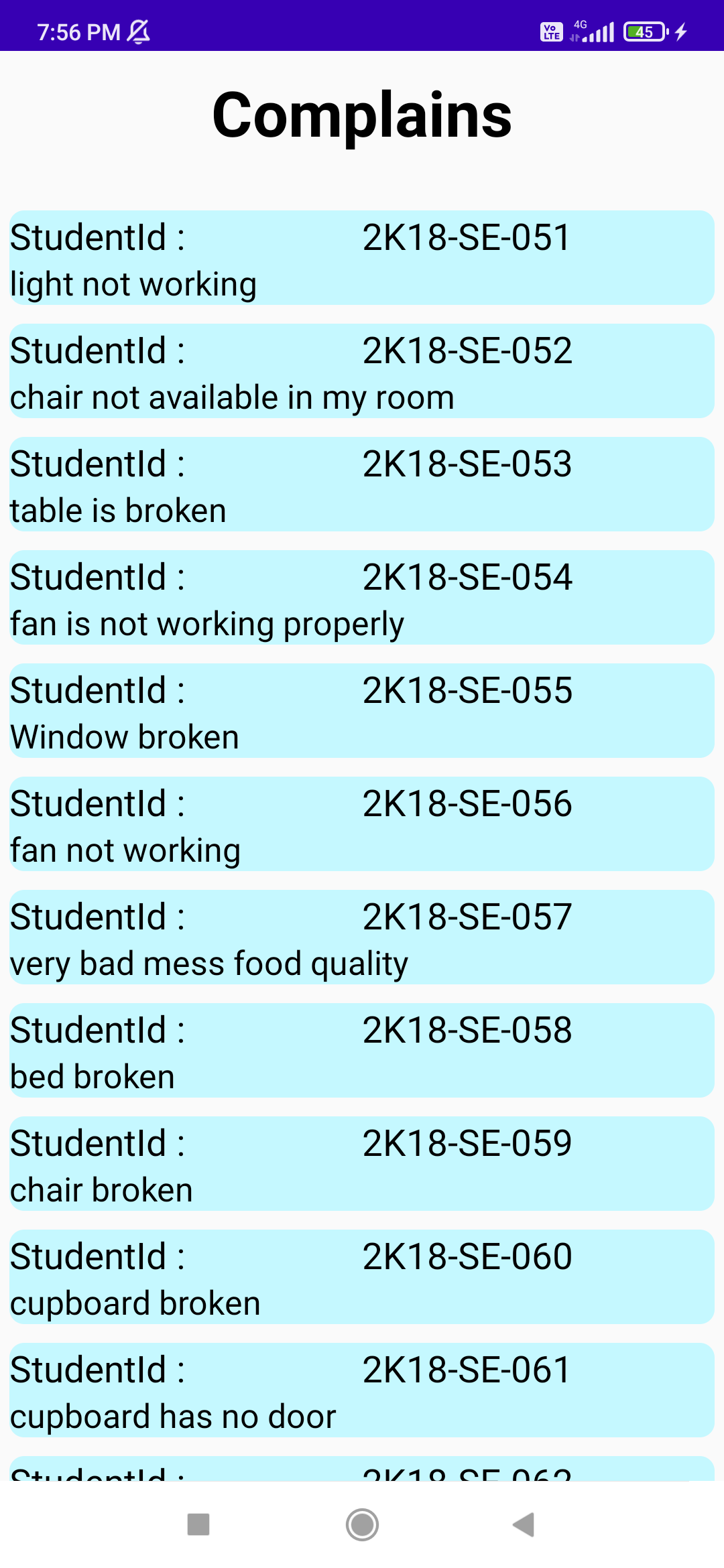
1. If the Student is not registered the app will display “Student Not Registered”
2. If the student is registered, all the details of the student will be displayed in the fields below.
3. **VACATE ROOM**



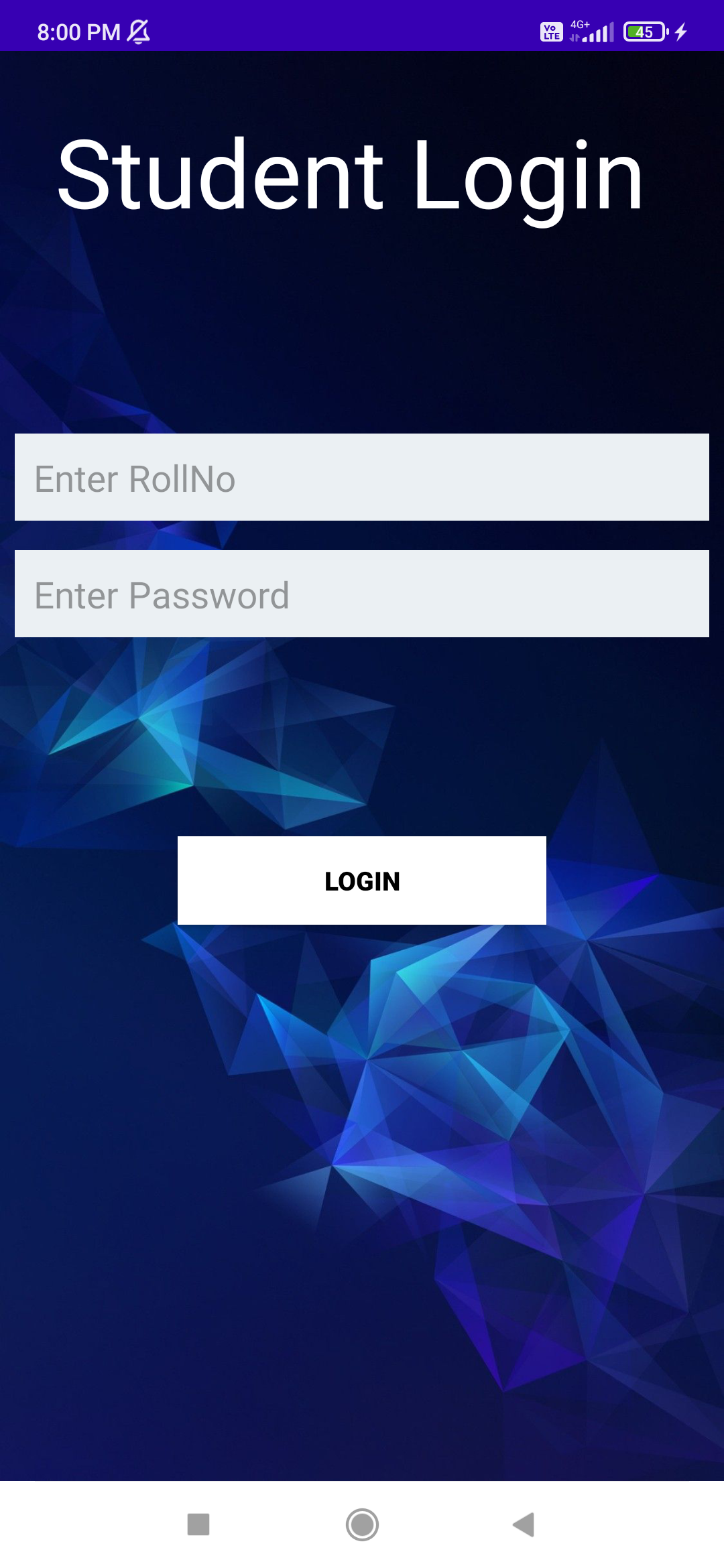
1. If the student is not registered the app will show “Student Not Registered”.
2. If the student is registered and a room as not been allotted to the student the app will show “no room allotted”.
3. If the student is registered and a room has been allotted the, student would be vacated from the room.
4. **VIEW ROOMS**



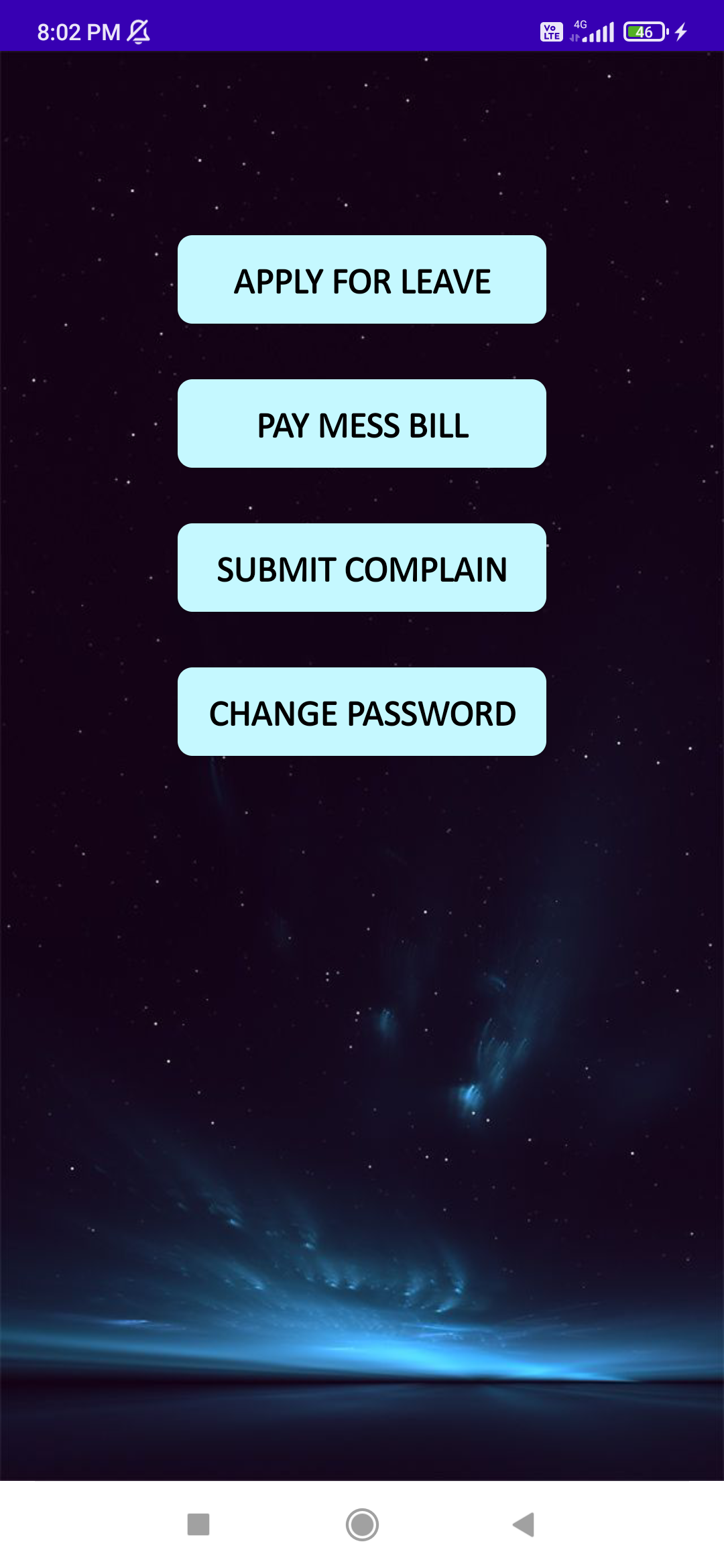
1. All the information about the rooms would be displayed in an orderly manner.
2. Room Status – ‘0’ indicates the room is empty and Room Status – ‘1’ indicates the room is occupied.
3. The default Student Id for an empty room is ‘2K18-SE-00’.
4. **VIEW COMPLAINS**



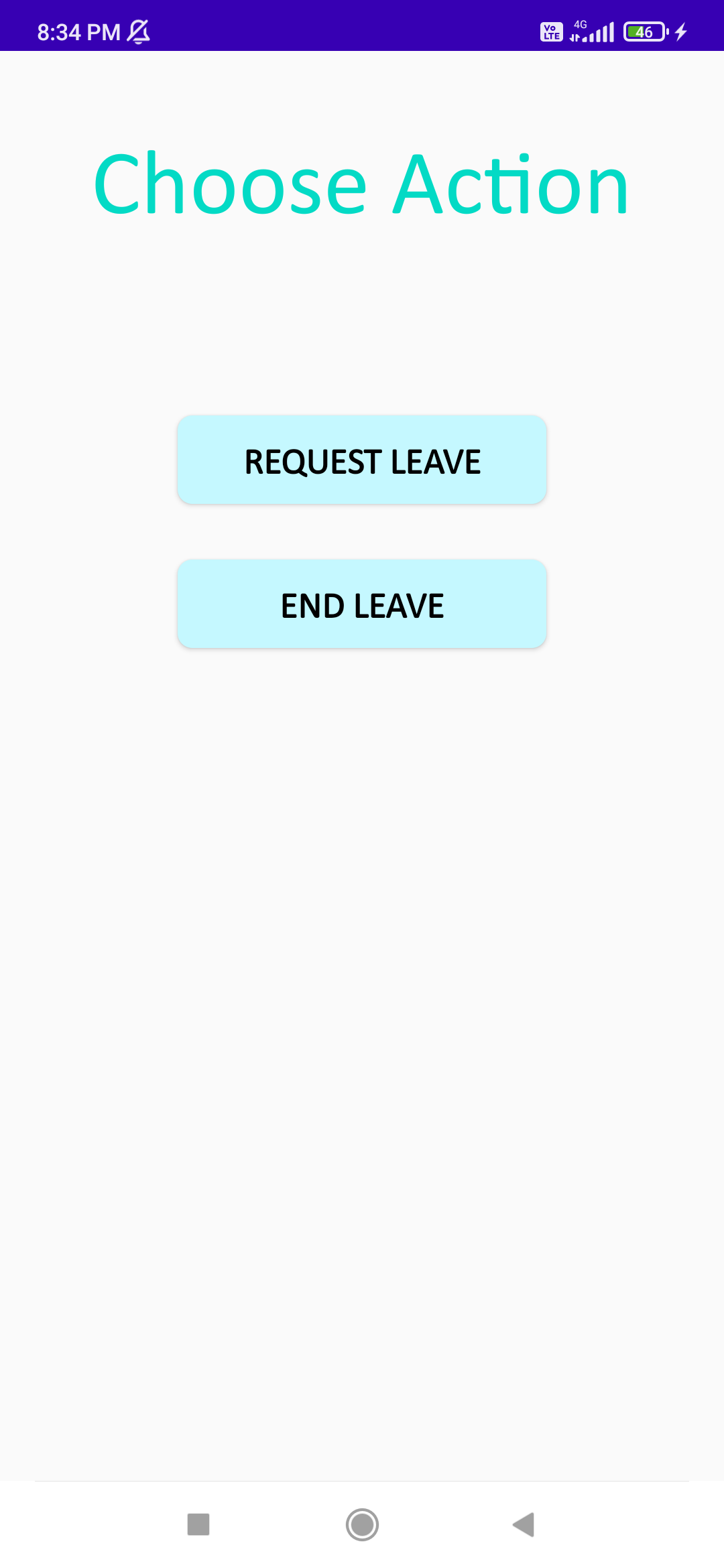
1. All the students complain are listed here
2. Every complain has two parts one is Student Id and the other the grievance of the student.
3. **STUDENT LOGIN**



1. Student has to enter his/her roll no and his/her DOB as password for initial login.
2. Student can change password after logging in.
3. If the Student is not registered in database the app will show student not registered.
4. **STUDENT HOME**



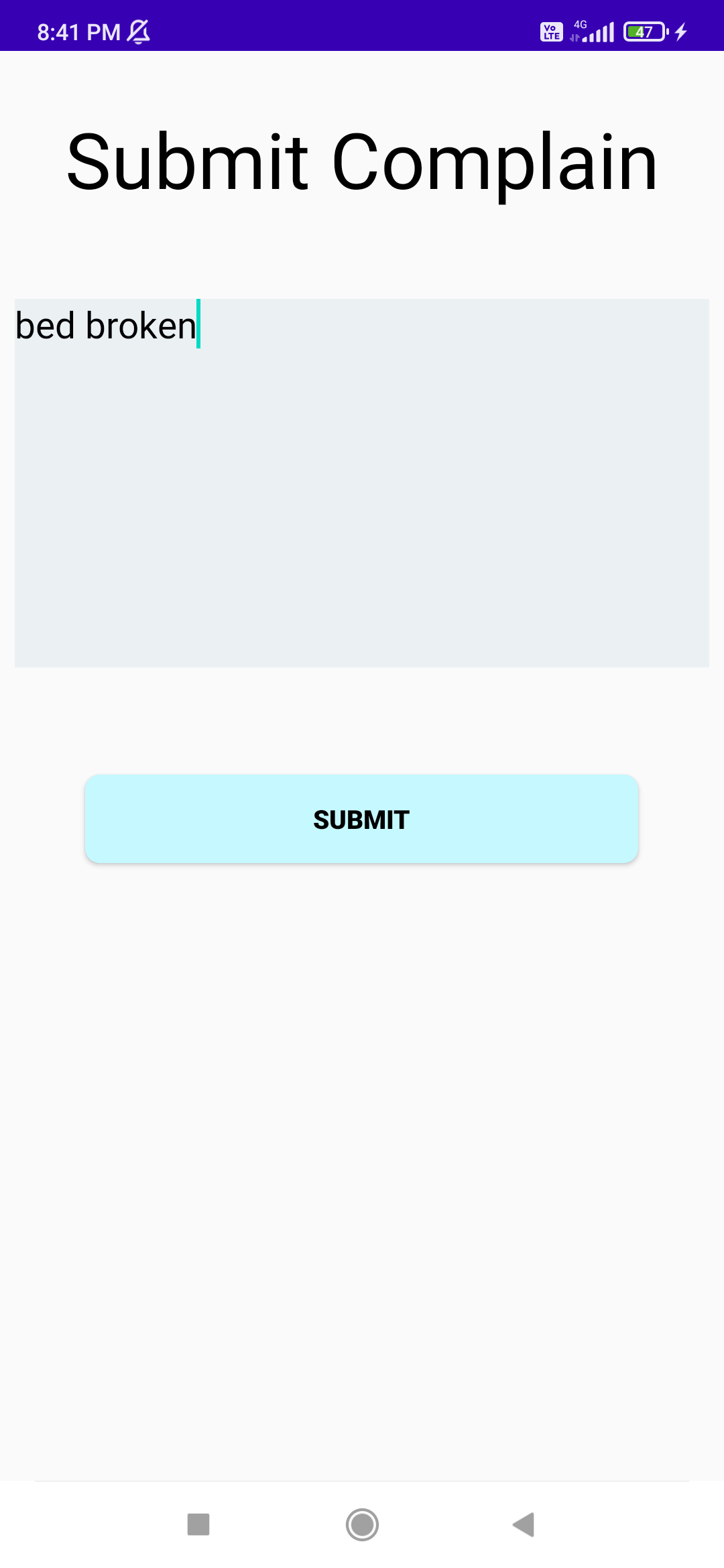
1. The Student Home as four functionalities further which are, Apply for Leave, Pay Mess Bill, Submit Complain and Change Password.
2. **APPLY FOR LEAVE**



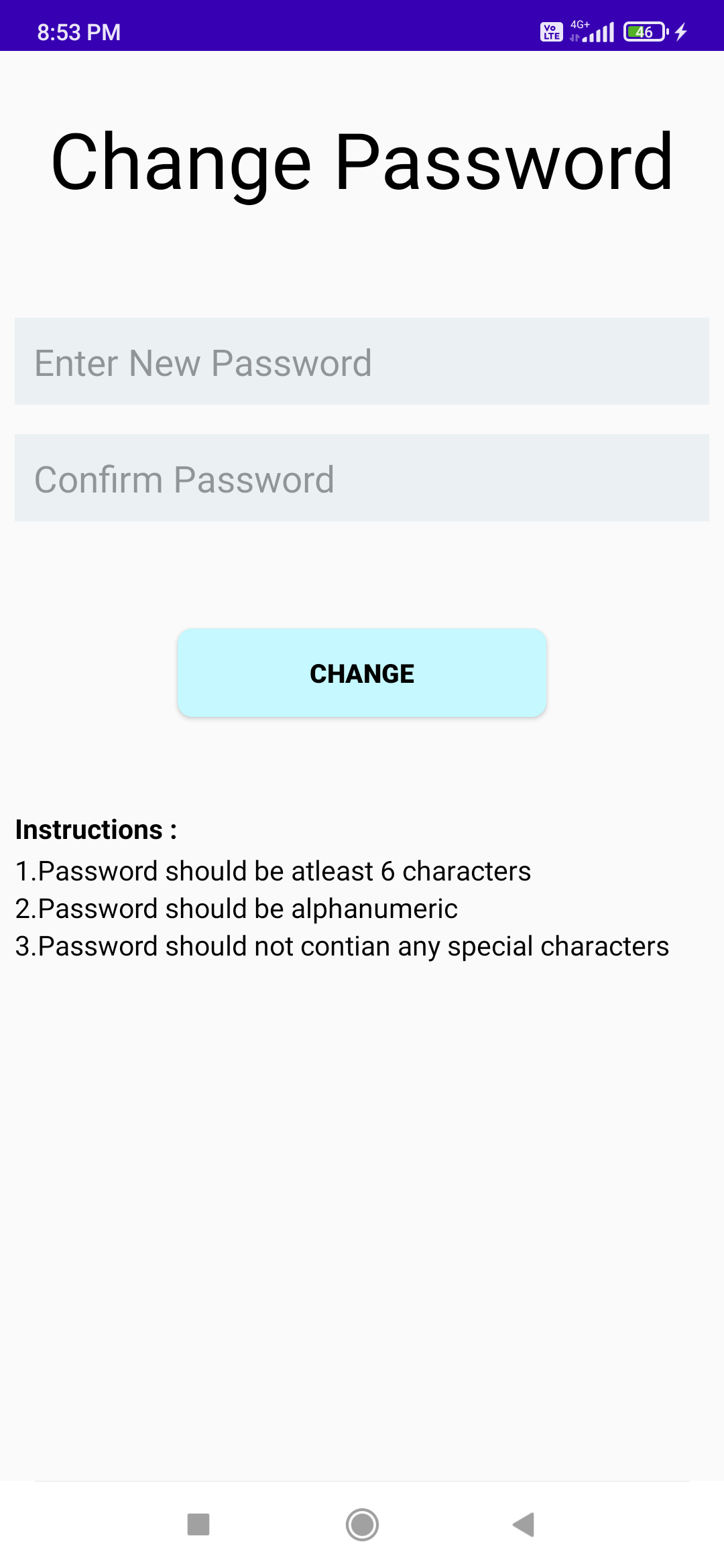
1. The student can request for leave if he is already not on leave else the app would display “already on leave”
2. The student can end leave if already on leave else the app would display “You are on leave currently”.
3. **PAY MESS BILL**



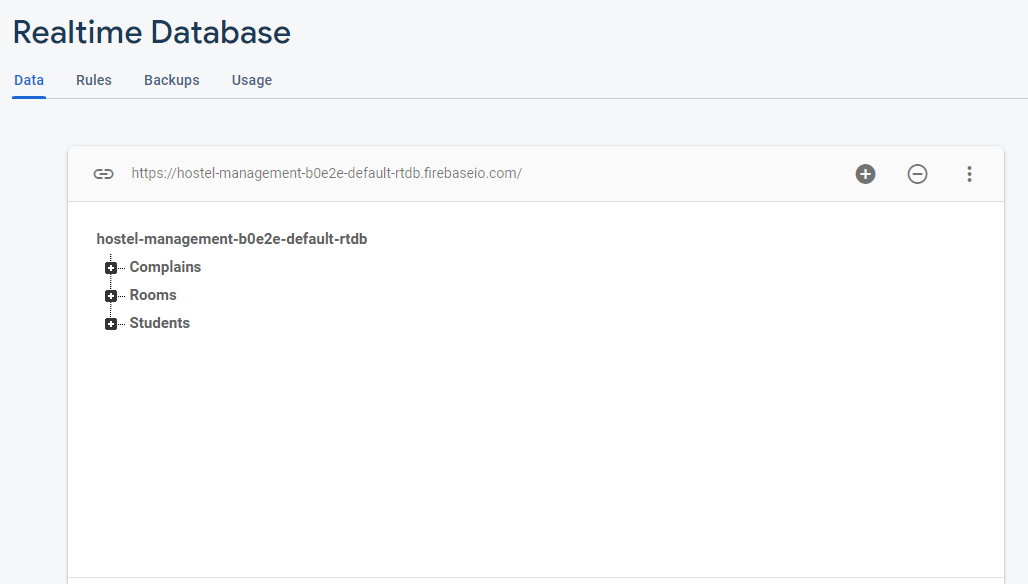
1. If the student has already paid the mess bill this screen would not open.
2. The student has option of paying the mess bill using one of the three options.
3. **SUBMIT COMPLAIN**



1. Student can enter the complaint he/she is facing in his/her room and the same would get added to complains section on the database with his/her Roll no.
2. **CHANGE PASSWORD**



1. Student can change his/her password by following the given rules:
2. Password should be at least 6 characters
3. Password should be alphanumeric
4. Password should not contain any special characteristics
5. Both the fields should have the same text
6. If all the conditions are met password would be updated in the database.
7. **SCREENSHOT OF THE FIREBASE DATABASE**

****

****

**CONCLUSION**

The Hostel Management System is primarily designed to decrease the workload of the hostel management authorities. The usage of Hostel management software should be encouraged in more hostels day by day as it is beneficial for both the students as well as the hostel staff.

The following would be achieved by the usage of hostel management software.

1. Systematic Control over Hostel Activities
2. Accurate Student Data Management
3. Complete Student Security

**FUTURE WORK**

1. Hostel Attendance could be tracked using the app.
2. Important notification could be spread through the app.
3. The project could be scaled up for the number of room it handles.
4. Hostel fees can be paid through the app
5. The complains can be removed from the app once resolved from the app only.
6. Students can have the option of calculating the mess dues by themselves.
7. Lost and found section can be embedded in the app.