

TABLE OF CONTENTS

S.No.	Assignment
1	Project Overview
2.	Analysis Phase
2.1	Use Cases
2.1.1	Use-Case diagram
2.1.2	Use Case Templates
2.2	Swimlane diagrams
2.3	Data Flow Diagrams (DFDs)
2.3.1	DFD Level 0
2.3.2	DFD Level 1
2.4	Software Requirement Specification in IEEE Format
2.5	User Story Cards
3.	Design Phase
3.1	Class Diagram and Object Diagram
3.2	Sequence Diagram
3.3	Collaboration Diagram
3.4	Database Design : ER Diagram
3.5	State Chart Diagrams
4.	Implementation
4.1	Component Diagrams
4.2	Deployment Diagrams
4.3	Screenshots of Working Project
5.	Testing
5.1	Cyclomatic Complexity (All modules)
5.2	Test Cases
5.3	Test Reports

1 Project Overview

All In One Management system

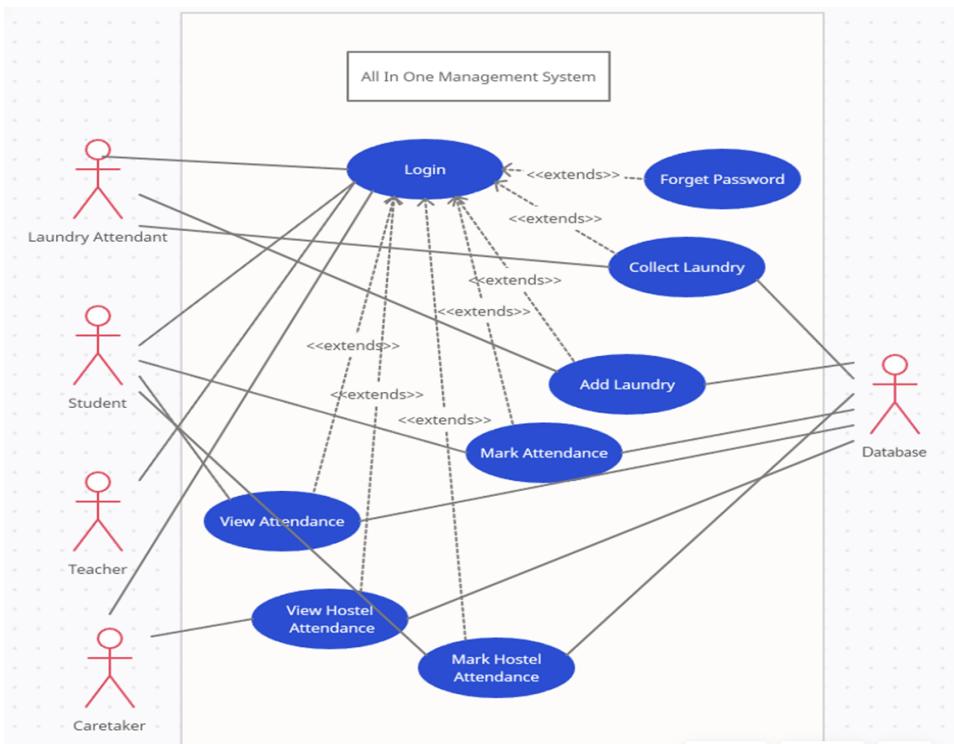
The goal of this project is to design a simple yet useful mobile application which would help automate various functions which include automated laundry system, hostel and class attendance system and timetable update feature. In this application , a user would have to Sign in , following which he would be given access to the specific features, e.g a caretaker would be able to access the hostel attendance system only. Here are the features that can be provided by our mobile app :

- 1)**Class Attendance:** Attendance would be taken by scanning the barcode of the student's ID card , which would indeed prevent proxies and also help save class time
- 2) **Hostel Attendance :** Attendance would be taken by scanning the barcode of the student's ID card , and since this utilises only the smartphone's camera so this would work even on a low end smartphone.
- 3)**Laundry Activity tracker:** This would utilise the barcode of the Student's ID card to add laundry items and since the checklist is common for all the hostels it could be used for all the hostels. Also using this we can retrieve the status of previous laundry also to prevent misplacement of items.

2. Analysis Phase

2.1 Use-Case

2.1.1. Use case diagram



2.1.2 Use case Template

1. Use Case Title	Login
2. Abbreviated Title	Login
3. Use Case ID	1.0
4. Actors	Admin , End user , Database

5. Description	Through this process different kinds of users (such as laundry attendant, student, teacher, caretaker) will be able to access our application contents.
5.1 Pre Condition	User must have an Account
5.2 Task Sequence	<ol style="list-style-type: none"> 1. Reach to the login page 2. Enter Credentials (rollno/email & password) 3. Click Login to Authenticate
5.3 Post Condition	User will be logged In successfully if the user has entered valid credentials otherwise the user will be notified that they have entered invalid credentials.
6. Modification History	11th September 2022
7. Author	Harishankar Kumar,Gurpreet Singh,Abhilash Jena,Abhinandan Singla

1. Use Case Title	Forgot password
2. Abbreviated Title	Forgot password
3. Use Case ID	2.0
4. Actors	Admin , End user , Database
5. Description	Through this process, any legitimate user can forget password and recover their account.
5.1 Pre Condition	User must have an Account, and the forget password request be legitimate (mail confirmation/etc success)

5.2 Task Sequence	<ol style="list-style-type: none"> 1. Reach login page 2. Click on forget password 3. Verify your identity
5.3 Post Condition	User password will be reset/sent back in case of valid identity validation, else an error be shown.
6. Modification History	11th September 2022
7. Author	Harishankar Kumar,Gurpreet Singh,Abhilash Jena,Abhinandan Singla

1. Use Case Title	Collect laundry
2. Abbreviated Title	Collect laundry
3. Use Case ID	3.0
4. Actors	Laundry attendant, Database
5. Description	Through this process laundry attendant will collect laundry on behalf of student and give it to them.
5.1 Pre Condition	Laundry attendant must be logged in.
5.2 Task Sequence	<ol style="list-style-type: none"> 1. Get logged in. 2. Click on Laundry fragment. 3. Click on laundry number/roll no to collect.

5.3 Post Condition	Laundary will be registered collected, and the clothes will be manually given to user.
6. Modification History	11th September 2022
7. Author	Harishankar Kumar,Gurpreet Singh,Abhilash Jena,Abhinandan Singla

1. Use Case Title	Add laundry
2. Abbreviated Title	Add laundry
3. Use Case ID	4.0
4. Actors	Laundary attendant, Database
5. Description	Through this process laundry attendant will add clothes on behalf of student.
5.1 Pre Condition	Laundary attendant must be logged in.
5.2 Task Sequence	<ol style="list-style-type: none"> 1. Get logged in 2. Add laundry with details regarding different types of clothes and their numbers. 3. Add student roll number and submit.
5.3 Post Condition	If successfully added laundry and it doesn't exceed the limit of clothes (10 currently) then print a receipt with laundry number and list of cloths else show error.

6. Modification History	11th September 2022
7. Author	Harishankar Kumar,Gurpreet Singh,Abhilash Jena,Abhinandan Singla

1. Use Case Title	Mark attendance
2. Abbreviated Title	Mark attendance
3. Use Case ID	5.0
4. Actors	Student, Database
5. Description	Through this process any student can mark their attendance on any given day.
5.1 Pre Condition	Student must be logged in with their student account.
5.2 Task Sequence	<ol style="list-style-type: none"> 1. Get logged in 2. Go to attendance fragment 3. Click on mark attendance button.
5.3 Post Condition	Mark attendance if the request is valid, else give error.
6. Modification History	11th September 2022
7. Author	Harishankar Kumar,Gurpreet Singh,Abhilash Jena,Abhinandan Singla

1. Use Case Title	View attendance
2. Abbreviated Title	View attendance
3. Use Case ID	6.0
4. Actors	Student , Database
5. Description	Through this process any student can see their attendance statistics, the amount of classes they missed etc.
5.1 Pre Condition	Student must be logged in with their student account.
5.2 Task Sequence	<ol style="list-style-type: none"> 1. Get logged in 2. Go to attendance fragment 3. See their attendance history
5.3 Post Condition	A list with attendance, date/time and class information will be shown.
6. Modification History	11th September 2022
7. Author	Harishankar Kumar,Gurpreet Singh,Abhilash Jena,Abhinandan Singla

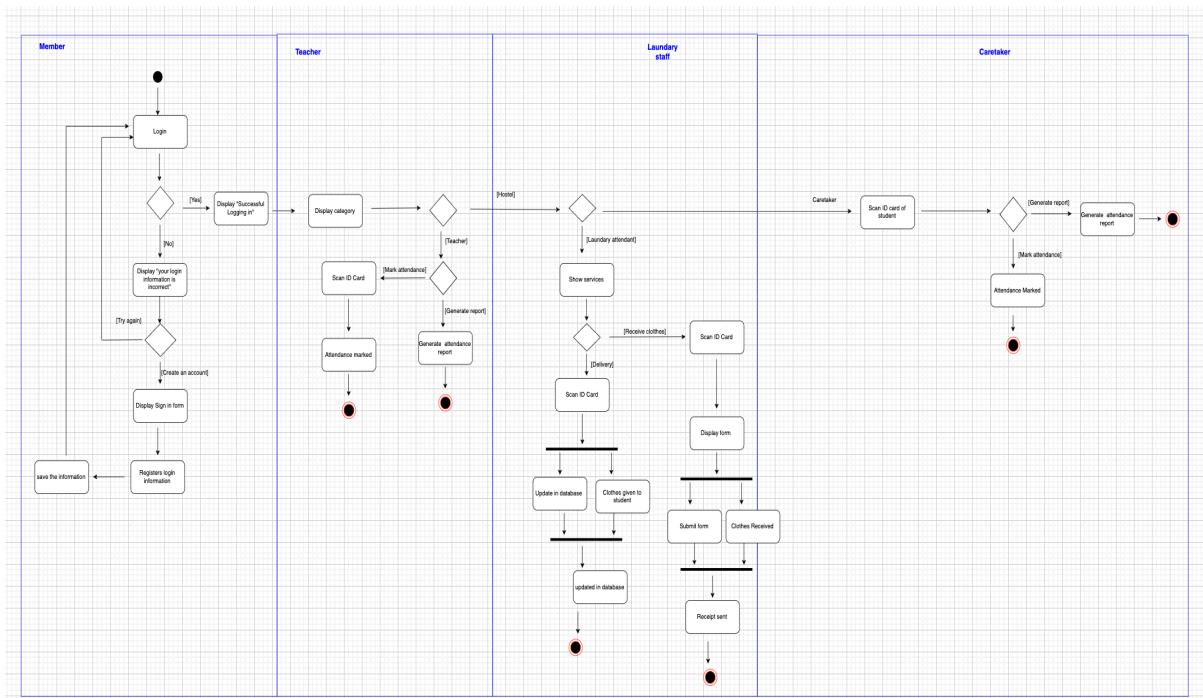
1. Use Case Title	Mark hostel attendance
2. Abbreviated Title	Mark hostel attendance
3. Use Case ID	7.0

4. Actors	Student, Database
5. Description	Through this process the residents of hostel can mark their respective attendance on a daily basis.
5.1 Pre Condition	Student must be logged in with their accounts.
5.2 Task Sequence	<ol style="list-style-type: none"> 1. Get logged in 2. Go to hostel attendance page 3. Click on mark attendance button
5.3 Post Condition	Hostel attendance will be marked successfully, if any error then it will be shown.
6. Modification History	11th September 2022
7. Author	Harishankar Kumar,Gurpreet Singh,Abhilash Jena,Abhinandan Singla

1. Use Case Title	View hostel attendance
2. Abbreviated Title	View hostel attendance
3. Use Case ID	8.0
4. Actors	Caretaker , Database
5. Description	Through this process, caretaker can see attendance! caretaker can see all the attendance of student and the number of their absents etc.

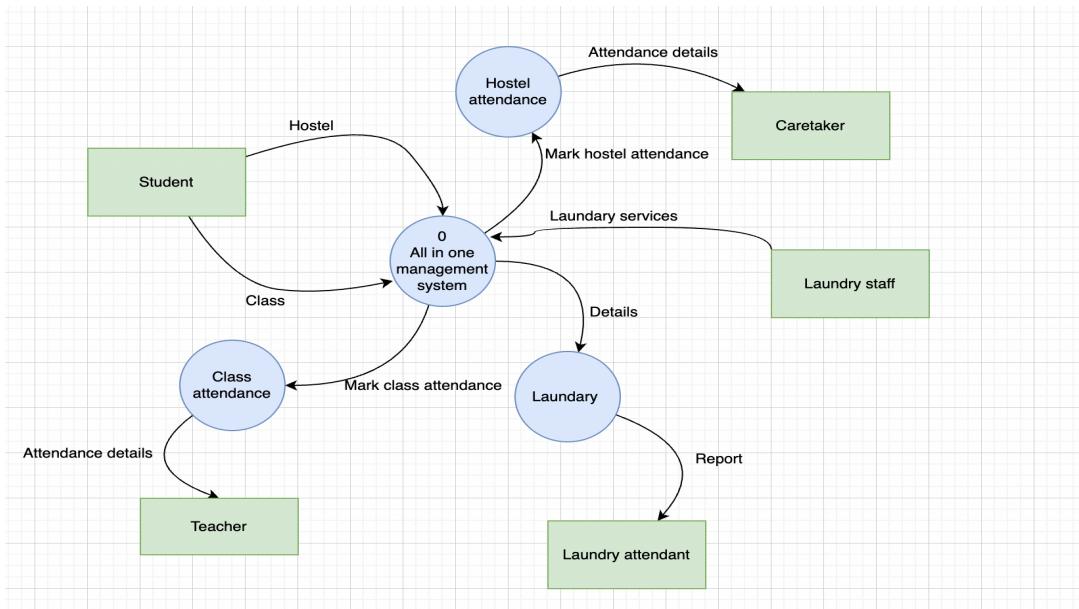
5.1 Pre Condition	Caretaker must be logged in with their accounts.
5.2 Task Sequence	<ol style="list-style-type: none"> 1. Get logged in 2. Go to attendance page 3. Click on view attendance and then enter roll no of student.
5.3 Post Condition	All the previous attendance will be shown.
6. Modification History	11th September 2022
7. Author	Harishankar Kumar,Gurpreet Singh,Abhilash Jena,Abhinandan Singla

2.2 Swimlane Diagram

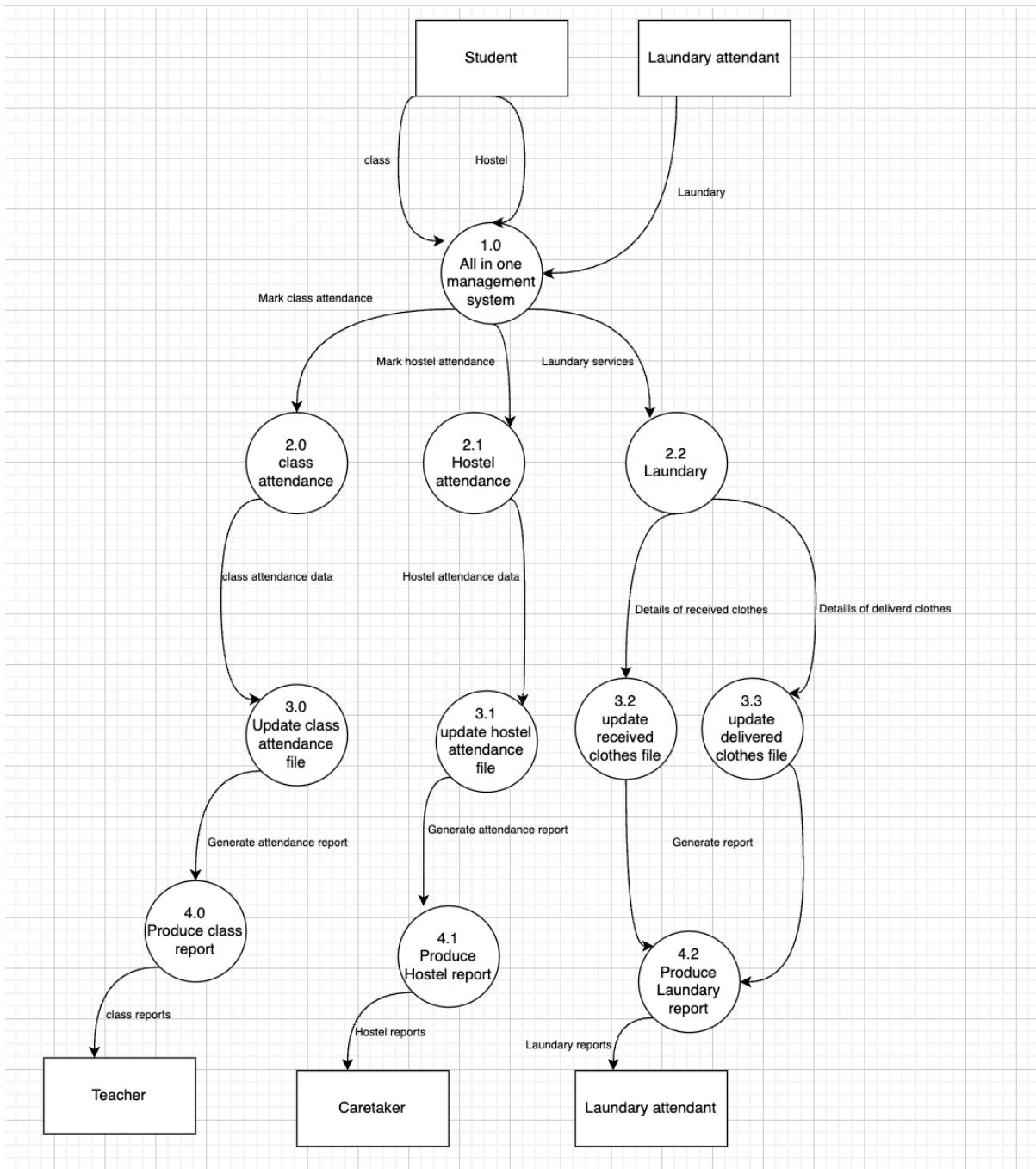


2.3 Data flow diagrams(DFD'S)

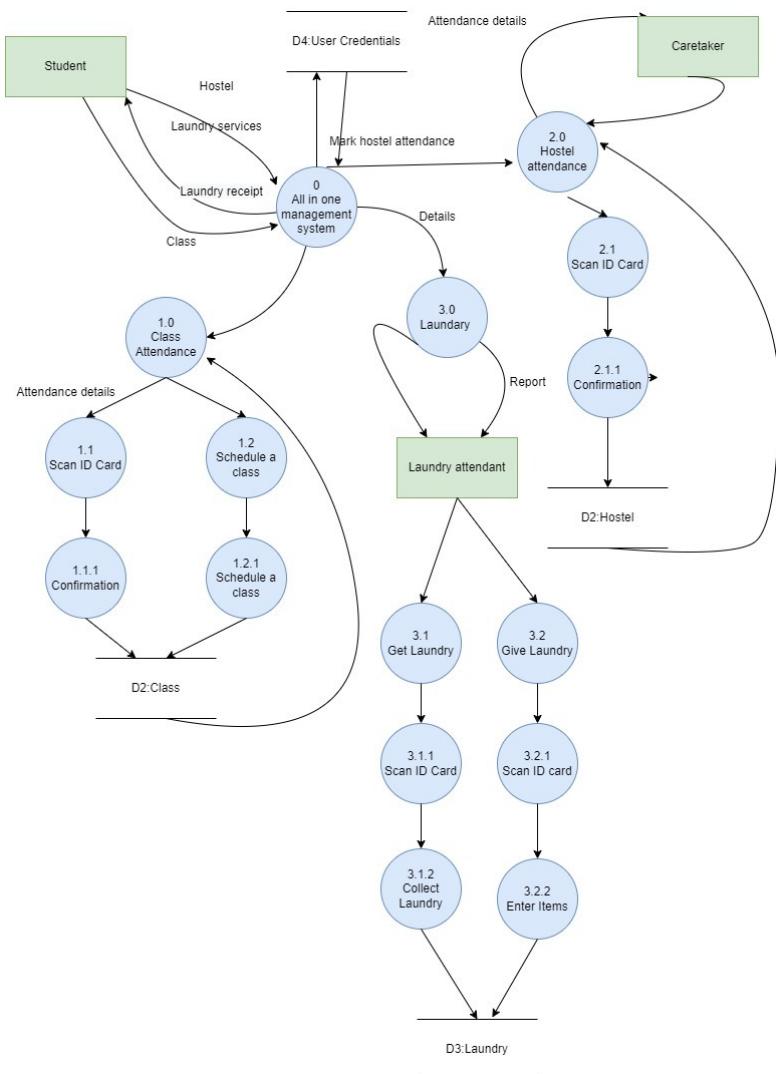
2.3.1 DFD Level 0



2.3.2 DFD Level 1



2.3.2 DFD Level 2



2.4 Software Requirement Specifications in IEEE Format

Functional Requirements:

Class Attendance

Input	Teacher would have to login first, after which he/she would be able to select the subgroup and then use his/her phone to scan the bar code of the id card
Processing	we can update the attendance directly to our database
Outputs	Would show if attendance was marked successfully

Hostel Attendance

Input	Caretaker would have to login first, then scan the ID card and only then he could mark the student's attendance
Processing	Attendance will be updated into the database
Outputs	Would show if attendance was marked successfully

Laundry Activity tracker

Input	Concerned authorities would have to scan the barcode of student's ID card, and then he/she can update the list of items given by the student
Processing	We can add this data to our database
Outputs	Would show the list of items that have been given for laundry

Non-Functional Requirements:

- 1)Cost:** Since it's a small scale project and all the software used in the project are completely free so there is no cost associated with the project.
- 2)Security:** Since only an authorized user can login into the app , it is completely secure.
- 3)Compatibility:** This mobile app is compatible with all android devices.
- 4)Portability:** Since our mobile app doesn't require any external hardware, so it is completely portable.

2.2 Feasibility Report

Schedule Feasibility:

Currently our team members have worked with backend technologies like Nodejs, Express and MongoDB. We will begin with wireframing and designing of the UI which would take 2 weeks, and backend APIs will consume another 2 weeks , after which we will start with learning flutter which will take 2 months approximately. In all it would take around 3 months time to completely deliver the project and hence we would be able to finish the project within the given deadlines.

Technical Feasibility:

Our mobile app is developed using flutter and the backend APIs would be developed using Nodejs, Express and we will be using MongoDB as our database.

Economical Feasibility:

All the software used to develop our mobile application are absolutely free.

Operational Feasibility:

Once the authorized user has the app installed he/she can simply login into the app and use it directly without any further setup.

Legal Feasibility:

Our mobile app complies to all the rules it would absolutely face no issues from a legal perspective.

Cultural/Behavioral Feasibility:

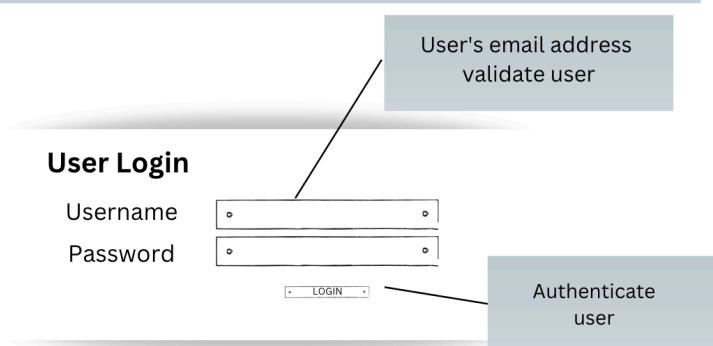
Our mobile app doesn't offend any culture ,also as the UI is quite simple and doesn't require any external hardware or special training, so it would be easily accepted by the authorities who wish to use it.

2.5 User Story Cards

1.

#0001 USER LOGIN

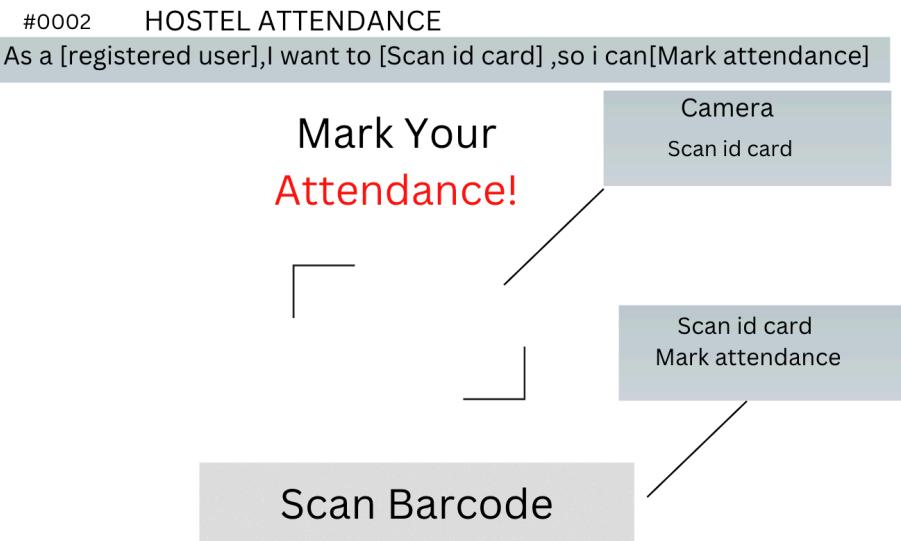
As a [registered user], I want to [login], so I can [access subscriber content]



Confirmation

1. Success-valid user logged in and referred to home page
 - a. 'Remember me' ticked-store-cookie / automatic login next time.
 - b. 'Remember me' not ticked - force login next time.
2. Failure -display message:
 - a. "Email address in wrong format"
 - b. "Unrecognised user name,please try again"
 - c. "Incorrect password, please try again"
 - d. "Service unavailable,please try again"
 - e. Account has expired - refer to account renewal sales page.

2.



Confirmation

1. Success-id card scanned
display message: "Attendance marked"

2. Failure -Continue Scanning

3.

#0003 COLLECT LAUNDRY

As a [registered user], I want to [Scan id card] , so i can[Collect laundry]

Scan Your
Id Card!

Camera
Scan id card

Scan id card
Clothes collected

Scan Barcode

Confirmation

1. Success-id card scanned
Redirect to items quantity page

2. Failure -Continue Scanning

4.

#0004

Add Session

As a [registered user], I want to [Add new session], so I can [mark attendance]

Subject Name

Room No

Start Time

End Time

Lecture details

Add to sessions

Submit

Create new session

Confirmation

1. Success-

Redirect to Sessions page

2. Failure -

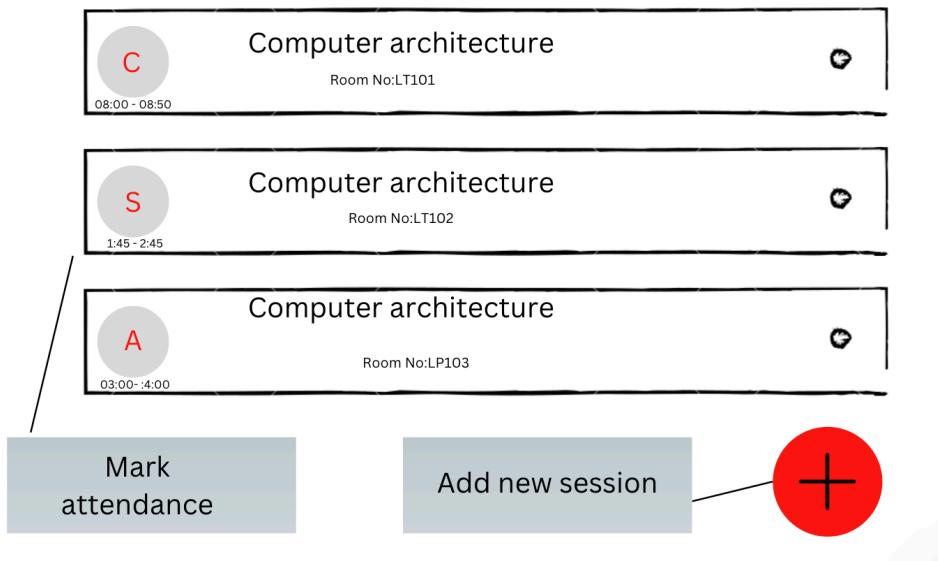
display message: Some field is empty

5.

#0005

Sessions

As a [registered user], I want to [Select session], so I can [Mark attendance]



Confirmation

1. Mark attendance

[Redirect to Sessions page](#)

2. Add new session

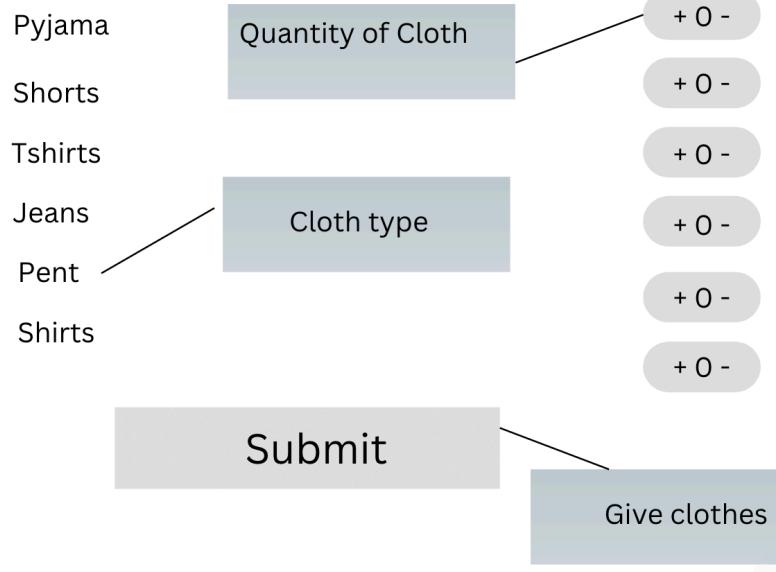
[Redirect to add session page](#)

6.

#0006

Laundry

As a [registered user], I want to [Add clothes quantity], so I can [Collect laundry]



Confirmation

1. Success

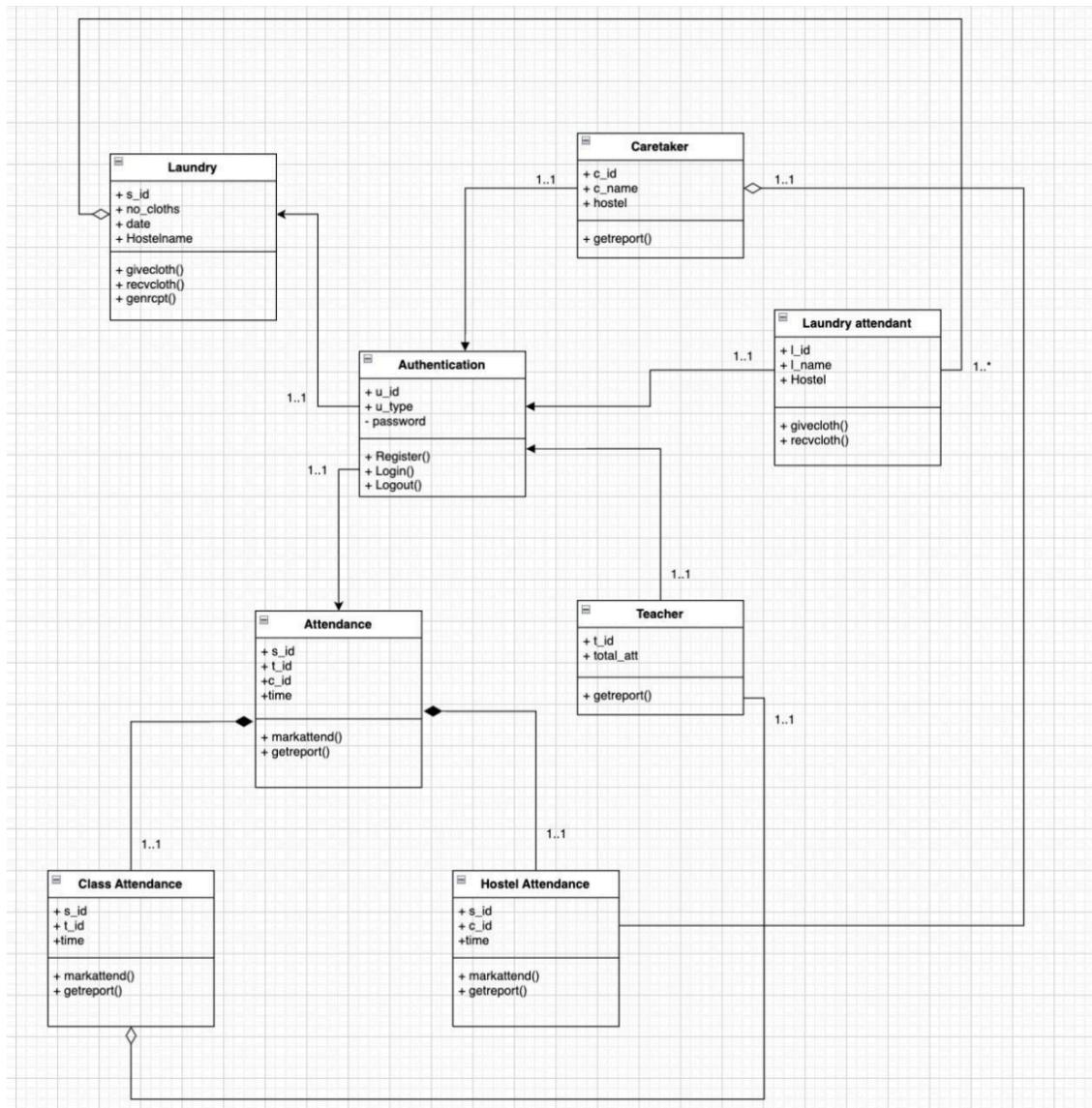
- a. Display message: Laundry taken

2. Failure

- a. Poor internet connectivity

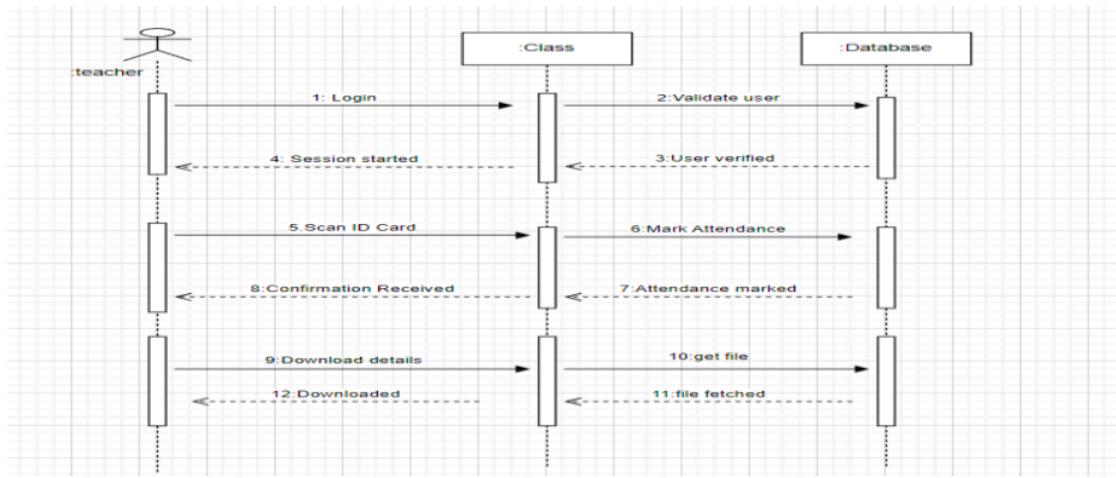
3. Design Phase

3.1 Class Diagram and Object Diagram

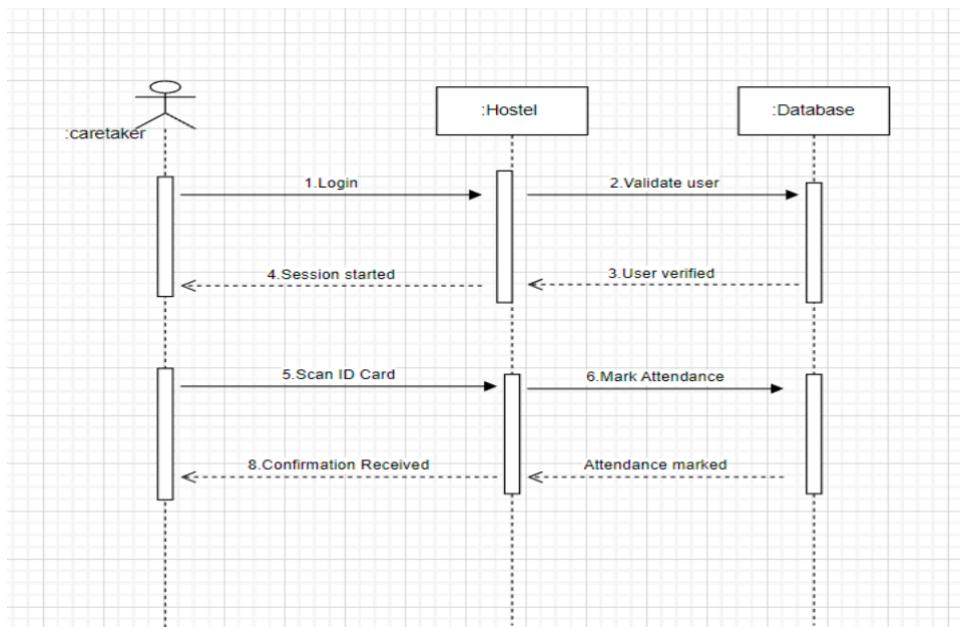


3.2 Sequence Diagram

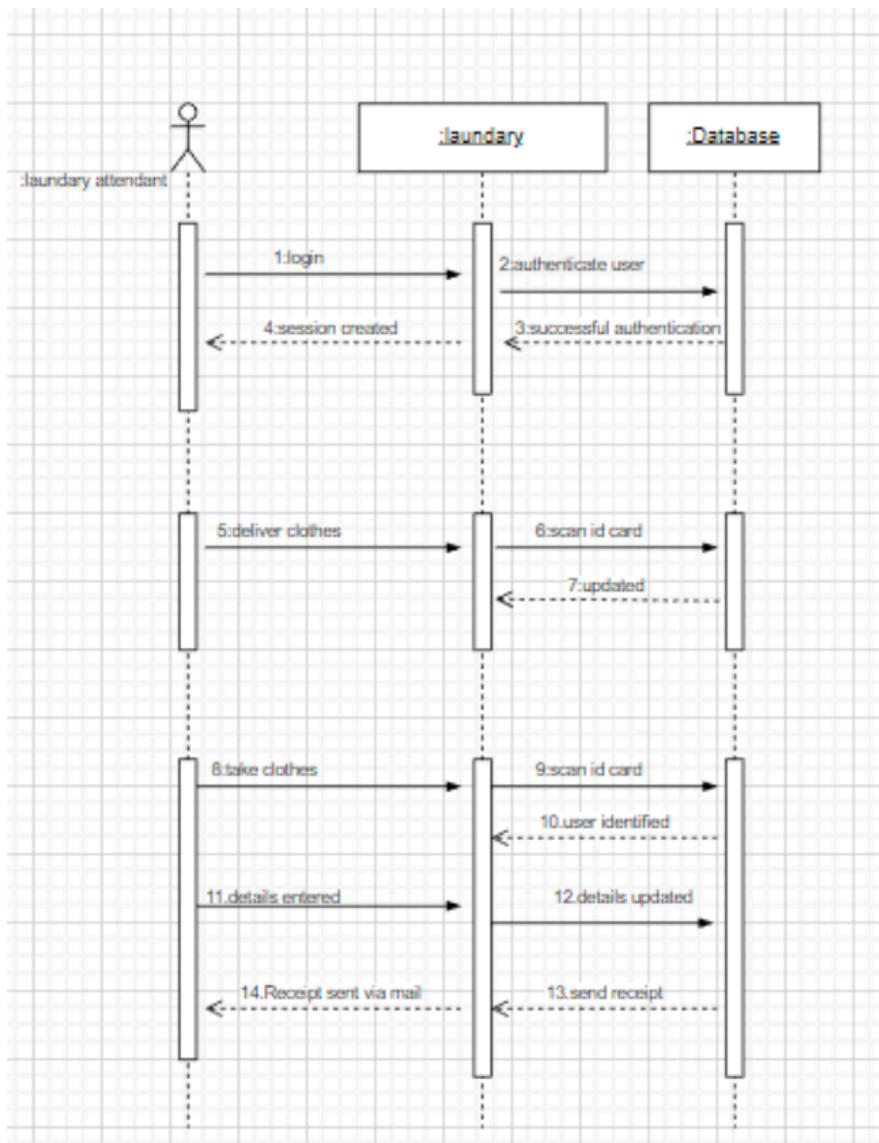
1. Class attendance



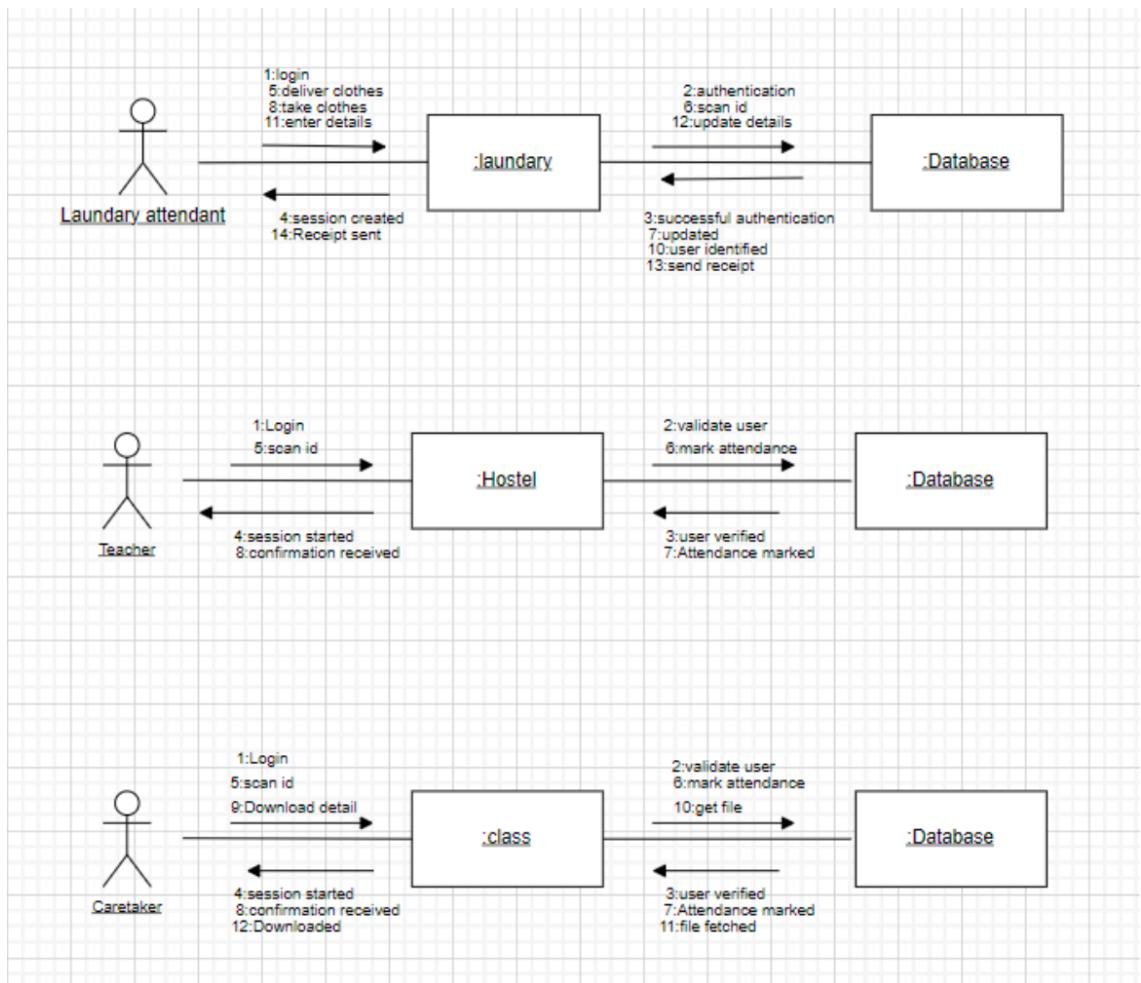
2. Hostel attendance system



3. Laundry management system

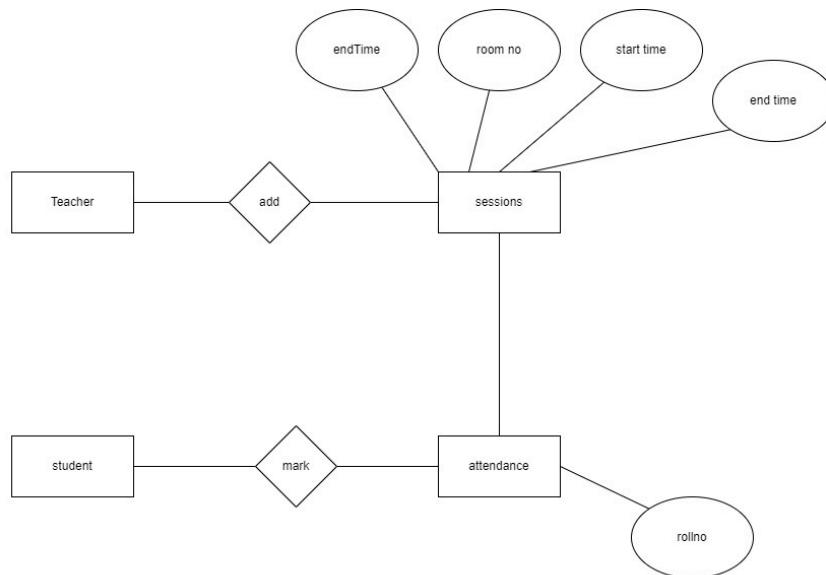


3.3 Collaboration Diagram

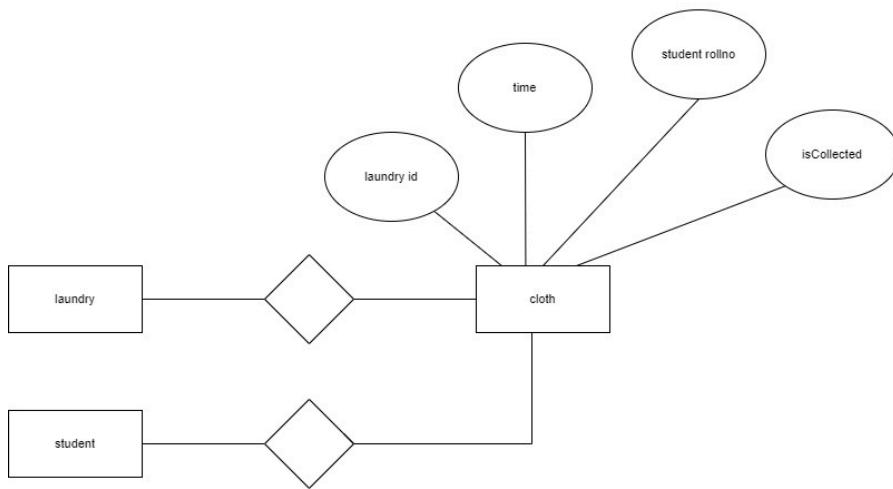


3.4 Database Design : ER Diagram

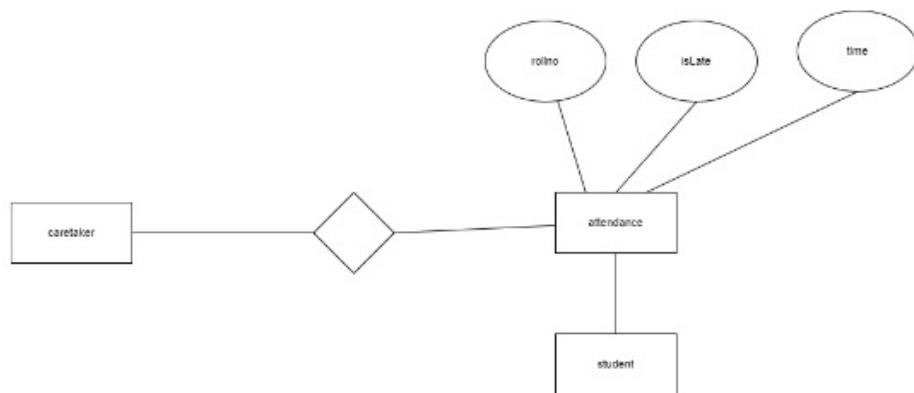
For Teacher



For Laundry

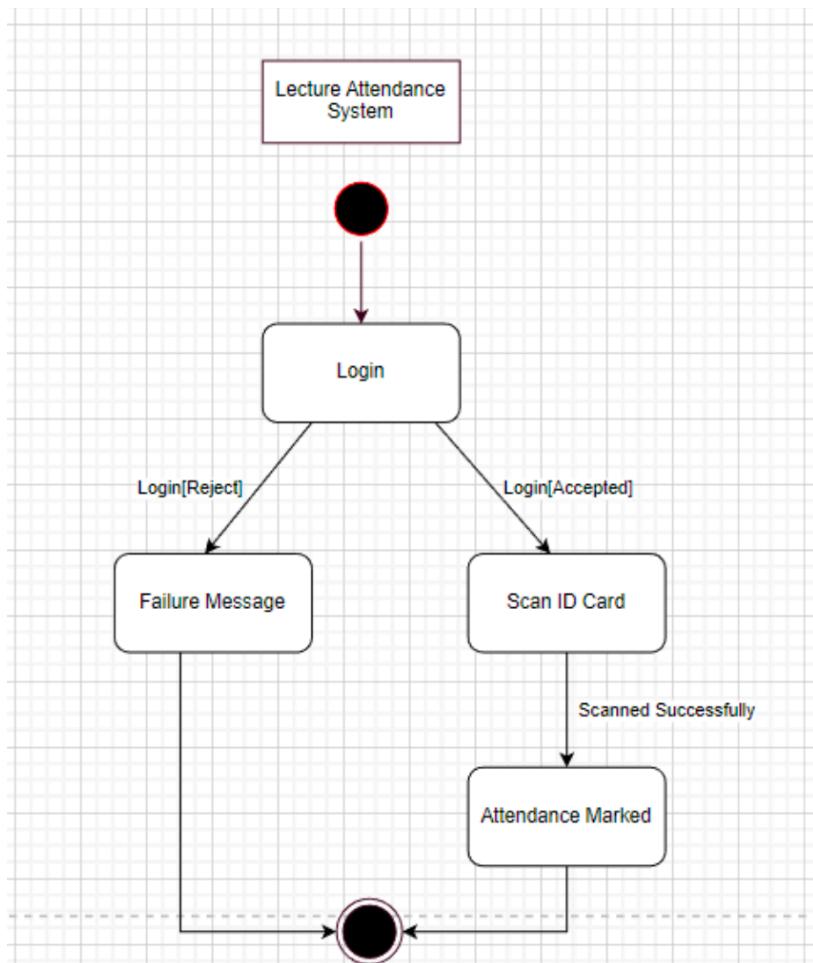


For caretaker

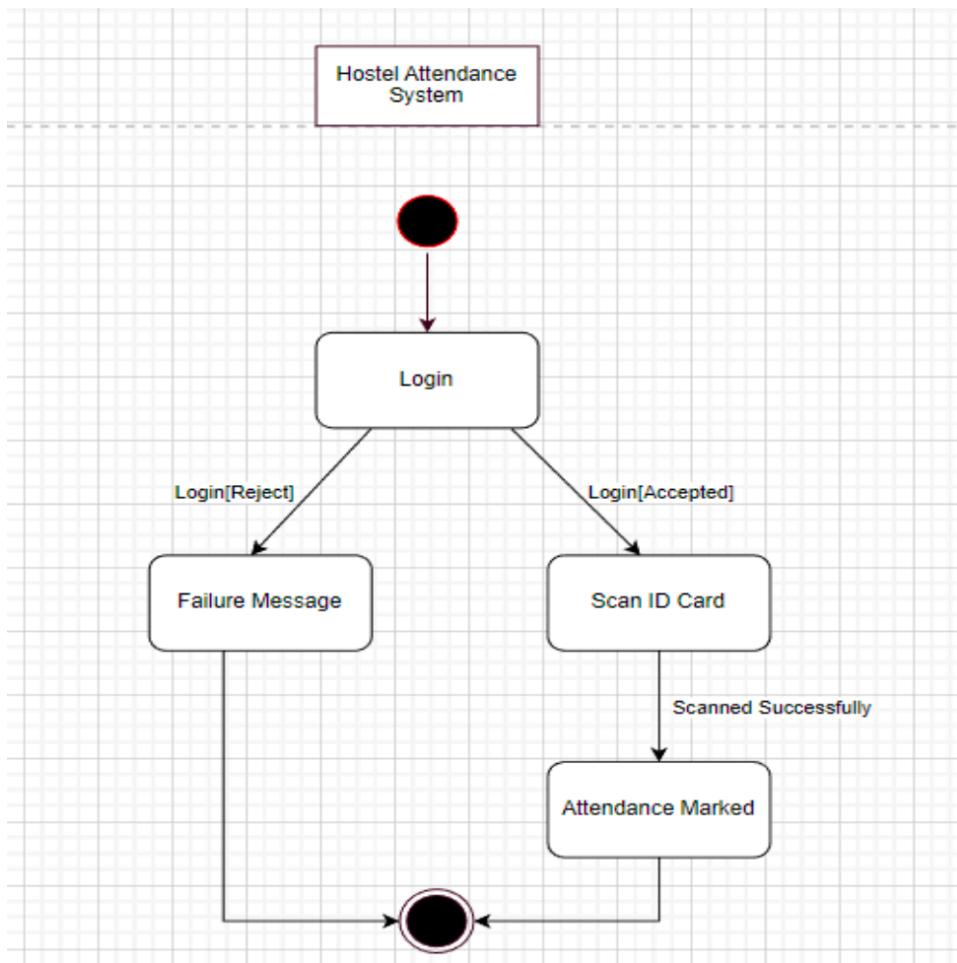


3.5 State Diagram

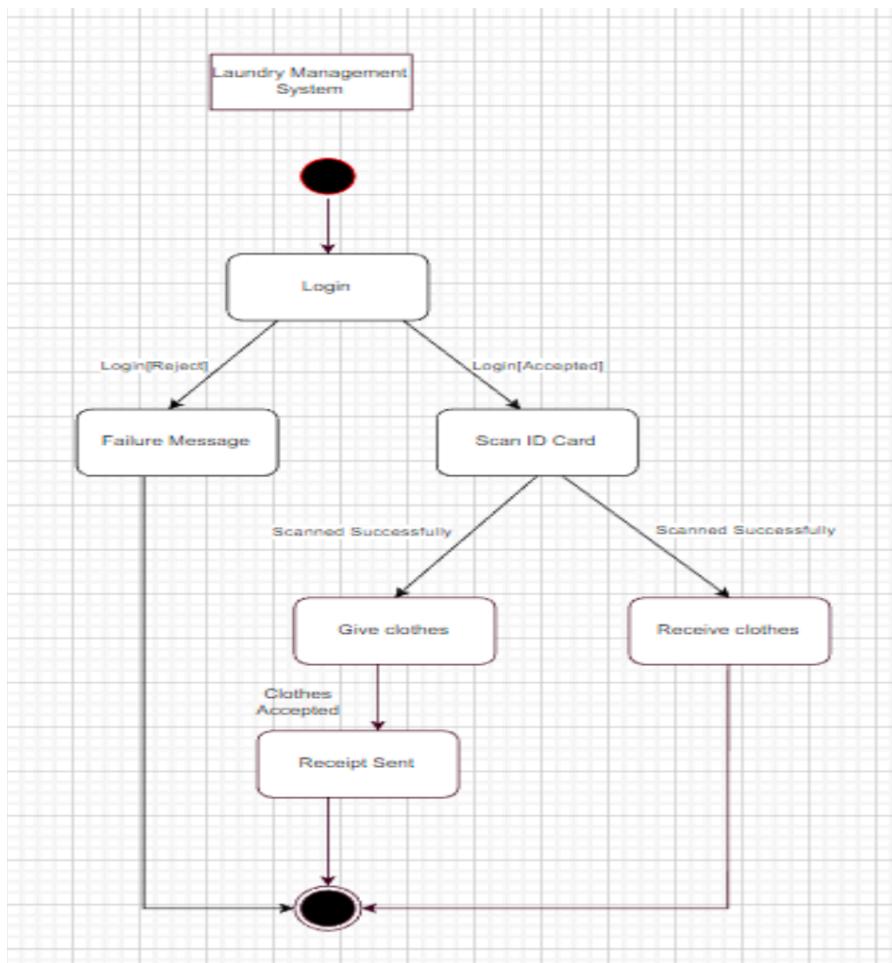
1. Class attendance



2. Hostel attendance system

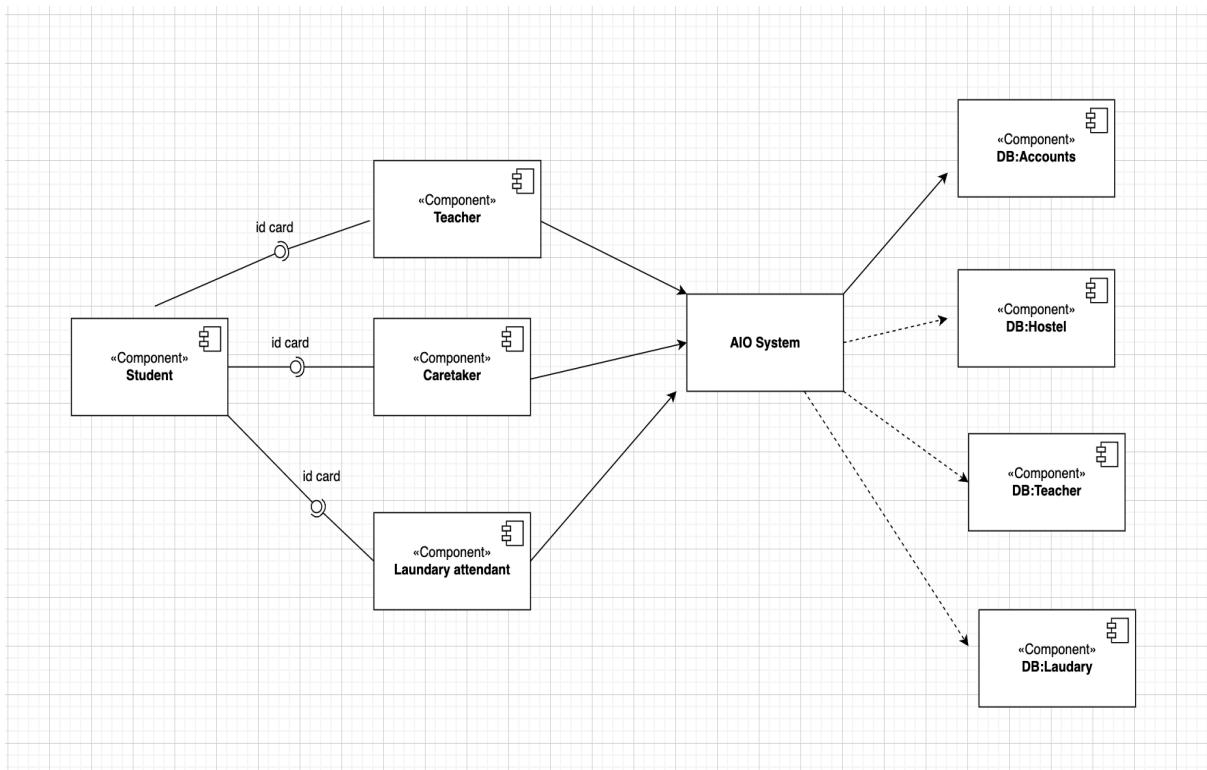


3. Laundry management system

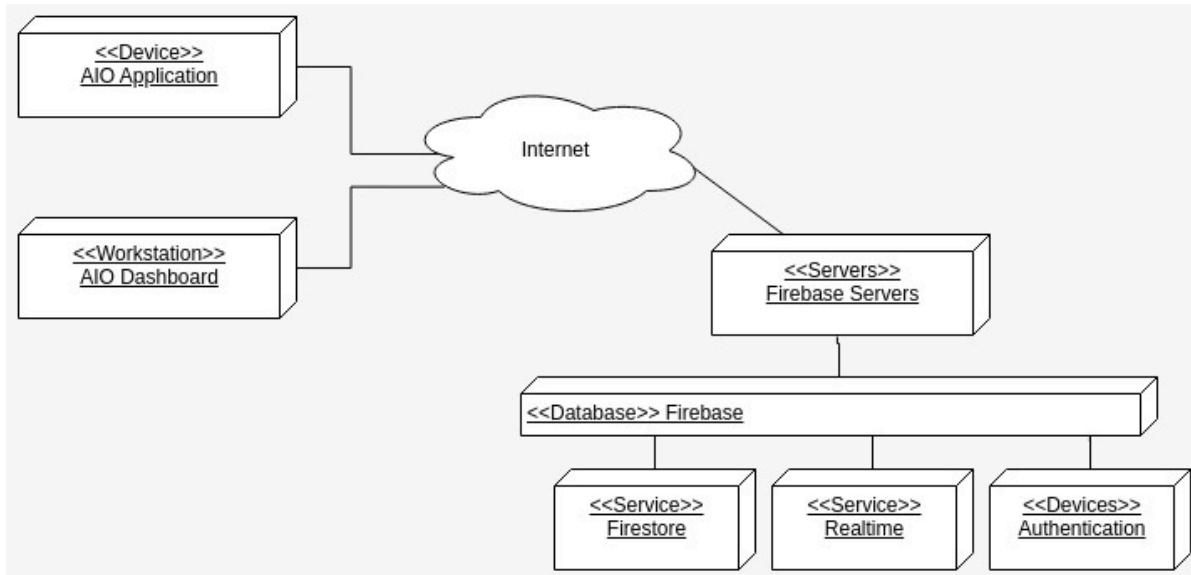


4.Implementation

4.1 Component Diagram



4.2 Deployment Diagram



4.3 Screenshots of working project

Sessions

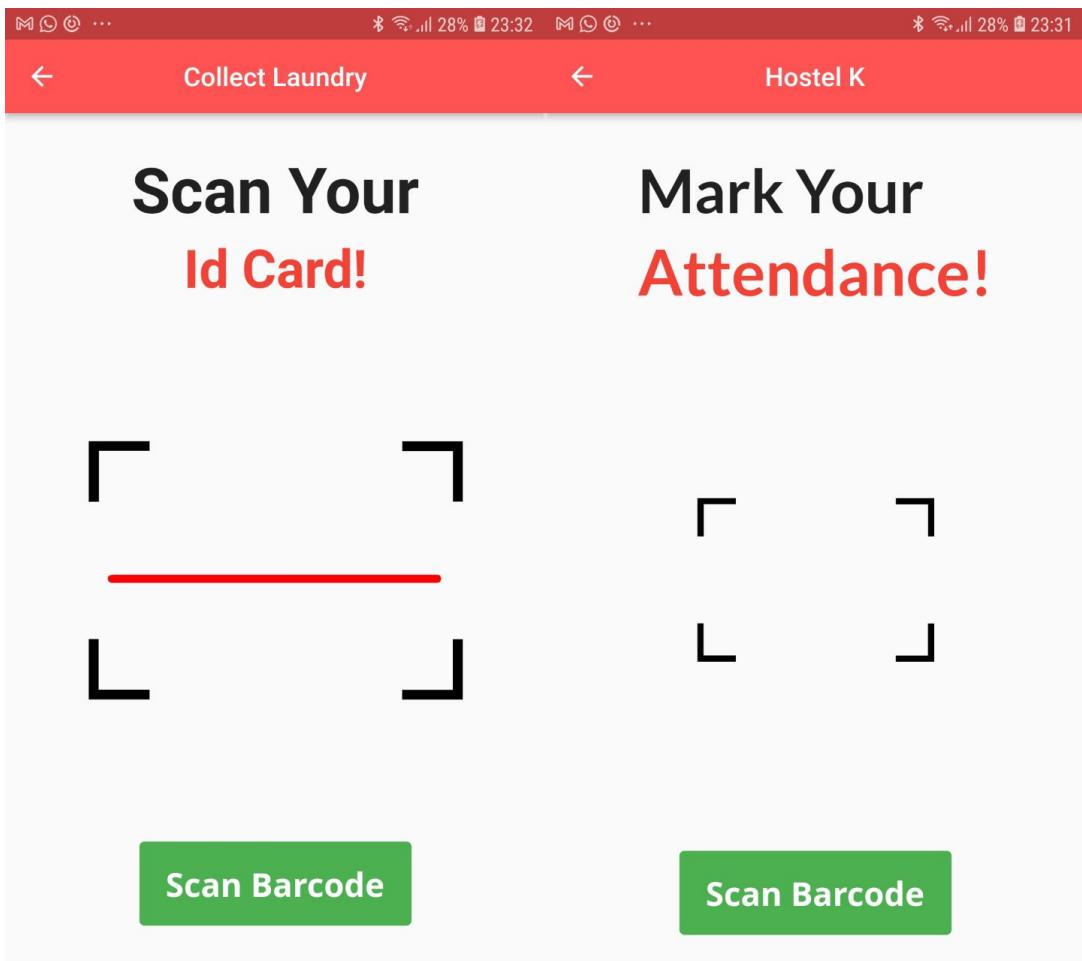
computer architecture
Room No - It 101
06:20 - 08:20

Software Engineering
Room No - SE lab102
21:44 - 19:45

Artificial intelligence
Room No - Lp-102
22:09 PM - 14:09 PM

Machine Learning
Room No - Lt-402
17:59:00 - 15:59:00

Network programming
Room No - Lt-123
15:06 PM - 20:06 PM

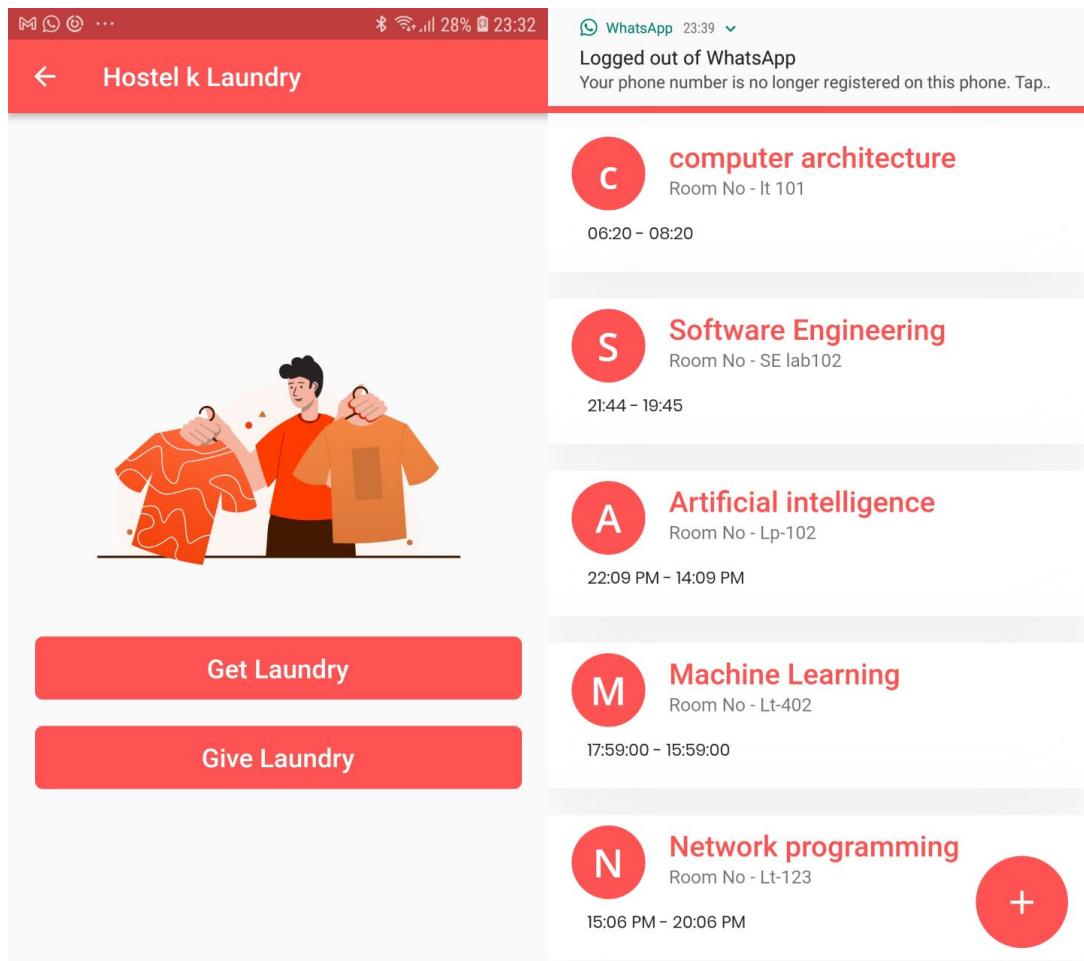


Laundry

	Jeans	+ 0 -
	Pant	+ 0 -
	Pyjama	+ 0 -
	Shorts	+ 0 -
	Shirts	+ 0 -
	T-Shirts	+ 0 -
	Kurta/Salwar	+ 0 -

Add Session

	Subject Name
	Room No
	Start Time
	End Time
Submit	



5. Testing

5.1 Cyclomatic Complexity

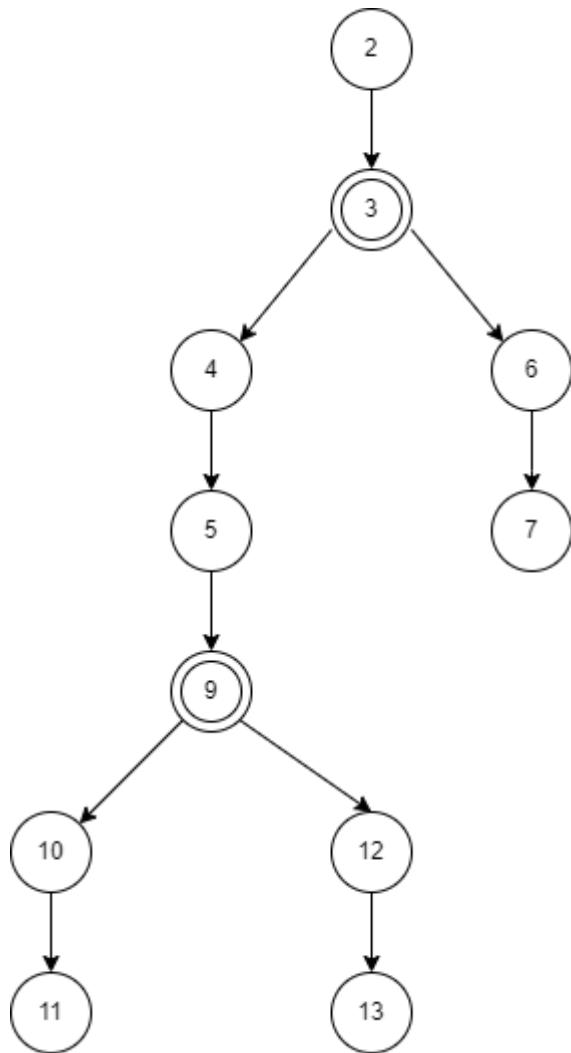
```

1.     Future scanBarcodeNormal() async {
2.         String barcodeScanRes;
3.         try {
4.             barcodeScanRes = await FlutterBarcodeScanner.scanBarcode(
5.                 '#ff6666', 'Cancel', true, ScanMode.BARCODE);
6.         } on PlatformException {
7.             barcodeScanRes = 'Failed to get platform version.';
8.         }
9.         if (barcodeScanRes == '-1') {

```

```

10.     return '-1';
11.     print('cancel');
12. } else {
13.     return barcodeScanRes;
14. }
15. }
```



Cyclometric complexity=no of predicate nodes+1 => 3

5.2 Test Cases

Test Case #:1.1

System: All in one Management System

Test Case Name: Test Login

Subsystem: LOGIN

Designed by: Abhilash Jena

Design Date: 15/11/2022

Executed by: Abhilash Jena

Execution Date: 15/11/2022

Short Description: Testing Login Page

Pre-conditions

Phone is Connected to Internet

User has already Signed up

Step	Action	Expected System Response	Pass/ Fail	Comment
1	Click on the app	System displays a Login Page	PASS	
2	Enter your credentials	System displays the Login Page	PASS	
3	Click the Sign In button	System checks if the user's data is present	PASS	
	Check post-condition 1		PASS	

Post-conditions

1. User's data was present in the database

2. User's data was not present in the database, error was returned

Case #:1.2

Test Case Name: Hostel

Attendance

System: All in one Management System

Subsystem: Attendance

Designed by: Abhilash Jena

Design Date: 15/11/2022

Executed by: Abhilash Jena

Execution Date: 15/11/2022

Short Description: Marking hostel attendance

Pre-conditions

The User has a valid Thapar ID card

Phone is connected to the internet

Step	Action	Expected System Response	Pass/ Fail	Comment
1	Click on mark hostel attendance	System displays an option with mark hostel option	PASS	
2	Click on scan ID card option	Camera opens up and allows user to Scan ID card	PASS	
3	Scan ID card	The system displays a message of successful acknowledgement as “Attendance marked”	PASS	
4	Check post-condition 1		PASS	

Post-conditions

1. Users record was added into the database

Case #:1.3

Test Case Name: Laundry

Collect Check

System: All in one Management System

Subsystem: Laundry System

Designed by: Abhilash Jena

Design Date: 15/11/2022

Executed by: Abhilash Jena

Execution Date: 15/11/2022

Short Description: Collecting Laundry

Pre-conditions

Phone is connected to the internet

User has a valid Thapar ID card

Step	Action	Expected System Response	Pass/Fail	Comment
1	Click on give laundry		PASS	
2	Click on scan ID card option	Camera opens up to scan the ID card	PASS	
3	Scan ID card	Barcode is captured	PASS	
4	Click on collect button	The system returns a positive acknowledgement	PASS	
5	Check Post-condition 1		PASS	
6	Repeat steps 1,2,3,4		PASS	
7	Enter wrong information	Card won't be scanned and user won't be redirected to the acknowledgement page	PASS	

Post-conditions

1.The IsCollected variable is changed to true

Case #:1.4

Check

Management System

Test Case Name: Laundry giving

System: All in one

Subsystem: Laundry System

Designed by: Abhilash Jena

Design Date: 15/11/2022

Executed by: Abhilash Jena

Execution Date: 15/11/2022

Short Description: Giving Laundry

Pre-conditions

Phone is connected to the internet

User has a valid Thapar ID card

Step	Action	Expected System Response	Pass/ Fail	Comment
1	Click on give laundry			
2	Click on scan ID card option	Camera opens up to scan the ID card	PA SS	
3	Scan ID card	Barcode is captured	PA SS	
4	Select the items	System displays the item name and quantity	PA SS	
5	Check Post-condition 1		PA SS	
6				

Post-conditions

1. Corresponding

Case #:1.5

Test Case Name: Lecture Attend.

System: All in one Management System

Subsystem: Lecture Attendance

Designed by: Abhilash Jena

Design Date: 15/11/2022

Executed by: Abhilash Jena

Execution Date: 15/11/2022

Short Description: Marking class attendance

Pre-conditions

The User has a valid Thapar ID card

Phone is connected to the internet

Step	Action	Expected System Response	Pass/ Fail	Comment
1	Click on the course name	System displays the courses of the teacher	PASS	
2	Click on scan ID card option	Camera opens up and allows user to Scan ID card	PASS	
3	Scan ID card	The system displays a message of successful acknowledgement as “Attendance marked”	PASS	
4	Check post-condition 1		PASS	

Post-conditions

1. Users record was added into the database

Test Case #:1.6**System:** All in one Management System**Designed by:** Abhilash Jena**Executed by:** Abhilash Jena**Short Description:** New class Schedule testing**Test Case Name:** Test**Subsystem:** Class Scheduling**Design Date:** 15/11/2022**Execution Date:** 15/11/2022**Pre-conditions**

Phone is Connected to Internet

User has already Signed up

Step	Action	Expected System Response	Pass/ Fail	Comment
1	Click on the plus(+) button	System displays a plus(+) button	PASS	
2	Enter the details of the class	System displays a form with start time,end time,name of the	PASS	
3	Click the submit	Class was added onto the application	PASS	
	Check post-condition 1		PASS	

Post-conditions

- 1.User's data was present in the database
- 2.User's data was not present in the database, error was returned

5.3 Test Report

Functions	Description	%TC Executed	%TC Passed	TCs pending	Priority
Give Laundry	New laundry record is initialised	100%	100%	0	High
Get Laundry	isCollected variable was updated	100%	100%	0	High
Mark Hostel Attendance	Attendance saved in database	100%	100%	0	High
Schedule a new class	Class was added into the database	100%	100%	0	High
Mark Class Attendance	Attendance saved in database	100%	100%	0	High
User Login	User logged in successfully when data was present in the database	100%	100%	0	