

Dharmsinh Desai University, Nadiad

Faculty of Technology, Department of Computer Engineering

B.Tech. CE Semester – VI

Subject: System Design practice

Project Title:

Build Up System(Construction App)

By:

Kinjal M Shah (Roll no: CE122 Id: 18CEUOD013)

Priyal M Parmar (Roll no: CE83 Id: 18CEUBD010)

Guided By:

Prof. Brijesh S. Bhatt



Dharmsinh Desai University, Nadiad

Faculty of Technology, Department of Computer Engineering

**CERTIFICATE**

This is to certify that System Design Practice project entitled “Construction App” is the bonafied report of work carried out by

1. **Kinjal M Shah (18CEUOD013)**
2. **Priyal M Parmar (18CEUBD010)**

Of Department of Computer Engineering, Semester VI, academic year 2019-20, under our supervision and guidance.

|  |  |
| --- | --- |
| Guide | HOD |
|  |  |
| **Prof. Brijesh S. Bhatt** | **Dr. C. K. Bhensdadia** |
| Assistant Professor of | Head of the Department of |
| Department of Computer | Department of Computer |
| Engineering, Dharmsinh Desai University, Nadiad. | Engineering, Dharmsinh Desai University, Nadiad. |

**CONTENTS**

1. [**Abstract 5**](#_gjdgxs)
2. [**Introduction**](#_2et92p0) **6**
   1. [Brief Introduction](#_1fob9te) 6

2.2 [Technology/Platform/Tools used](#_3znysh7) 6

1. [**Software Requirement Specification**](#_30j0zll) **7**
   1. [Introduction](#_2et92p0) 7

3.1.1 [Purpose](#_tyjcwt) 7

3.1.2 [Document Conventions](#_3dy6vkm) 7

3.1.3 [Intended Audience and Reading Suggestions](#_1t3h5sf) 7

3.1.4 [Product Scope](#_4d34og8) 7

3.2 [Overall Description](#_3dy6vkm) 8

3.2.1 [Product Perspective](#_2s8eyo1) 8

3.2.2 [Product Functions](#_17dp8vu) 8

3.2.3 [User Classes and Characteristics](#_3rdcrjn) 8

3.2.4 [Operating Environment](#_26in1rg) 8

3.2.5 [Design and implementations Constraints](#_lnxbz9) 8

3.3 [External Interface Requirements](#_35nkun2) 9

3.3.1 [User Interfaces](#_1ksv4uv) 9

3.3.2 [Hardware Interfaces](#_44sinio) 9

3.3.3 [Software Interfaces](#_2jxsxqh) 10

3.4 [System Features](#_z337ya) 10

3.5 [Other Nonfunctional Requirements](#_3j2qqm3) 14

3.5.1 [Performance Requirements](#_1y810tw) 14

3.5.2 [Safety Requirements](#_4i7ojhp) 14

3.5.3 [Security Requirements](#_2xcytpi) 14

3.5.4 [Software Quality Attributes](#_1ci93xb) 15

**4. Design 16**

4.1 Class Diagram 16

4.2 [Use-case Diagram](#_qsh70q) 17

4.3 Activity Diagram 20

4.4 sequence Diagram 22

4.5 State Diagram 24

4.6 E-R diagram 27

4.7 Data Dictionary 28

**5. Implementation 30**

**6.** [**Testing**](#_3o7alnk) **31**

**7.** [**Screenshots**](#_ihv636) **32**

**8.** [**Deployment Steps 32**](#_32hioqz)

**9.** [**Conclusion**](#_1hmsyys) **40**

**10.** [**Limitation and Future Extension**](#_23ckvvd) **41**

**11.** [**Bibliography**](#_vx1227) **42**

**1. Abstract**

We create the construction Application (Mobile app) which is very useful and efficient app for

construction of any site .This app we can have the different module such as admin, users and

worker .In users can show the statues of their site and also changes the design of the site if they

required. The construction sites have and different location and it can be managed by the

manager. Manager can also add the worker details and update the status of site. Also manager

can give the request for the row material which is required of the site construction. Admin can

manage the sites or show all the details of worker as well as manager and also modify and

approve for the row material details.so this app is very useful for the construction of the any site

and it is secure application.

# 2. Introduction

### Brief Overview

In this construction application we can reduce the time to prepare history of records instead of paper work. and this app is very useful for construct the sites and show details of the sites .admin can manage the Manager as well as users and approve for row material .manager can have all the details of worker and site construction. and users can show the implementation of their site construction.

### Technology Used

**Front End:**  Android Studio, HTML, CSS

**Back End:** Sqlite3(Database)

**Database:** .Net database

**Language:** JAVA(Mobile App)

**Development Tool:** Android Studio

**Diagram Tool:** Visio 2015

# 3. Software Requirement Specifications(SRS)

## Introduction

### Purpose

The purpose of this document is to present a detailed description of Construction Application. It will explain the purpose and features & the interfaces of the system, what the system will do, the constraints under which it must operate and how the system will react to the different users.

### Document Conventions

We have followed IEEE standard to prepare this document. This document is formatted in natural language sentence.

### Intended Audience and Reading Suggestions

This document can be referred by software developers, documentation writers, project manager and the users.

### Product Scope

The purpose of this system should be managing the construction in efficient way which can be secure and convenient. here the admin, users and manager can handle the system. Admin can manage the data of this construction Application.

## Overall Description

### Product Perspective

It can be used by authorized users.

### Product Functions

* + - * System allows admin to Approve Row material.
      * System allows admin to view all available Details of sites .
      * System allow admin to manage the users and manager.
      * System allow users to check the statues of sites.
      * System allow a manager to manage the worker details.

### User Classes and Characteristics

The Major User Classes in the System would be:

**1.** Admin

* + Admin user can manage all details of the Manager as well as users. Admin can also approve for row material.

1. **Manager**

* Manager can manage the all the details of the worker and also manages the sites.

1. **Users**

* Users can only show the statues of sites.

Operating Environment

As this is an online system so it will work on Android operating systems. Internet connection should be enabled on accessing device. All the devices must have modern phone to support accessing the system.

### Design and Implementation Constraints

* + - * All graphical modules should be supported in browser whereas there are no design constraints if accessing device is mobile.

### Assumptions and Dependencies

* + - * Assuming that Admin knows the data of Users Table.

## External Interface Requirements

### User Interfaces

* + - * The system will be fully compatible with any modern phone that support Android operating system.

### Hardware Interfaces

#### Server Side:

* + - * Operating System: Windows
      * Processor: Intel i3 2.3 GHZ or higher
      * RAM: 4GB or higher
      * Hard-Disk: 1 TB

#### Client Side:

* + - * Operating System: Android

### Software Interfaces

#### Server Side:

* + - * Web Server: ANY
      * Technology: Android, Java
      * Database: SQLite database

#### Client Side:

* + - * The user’s browser should support Android studio.

### Communication Interfaces

* + - * The System will require the HTTP + Java protocol to provide a secure communication over the internet.

## System Features

Users:-

1.Employee(contractor)(End user)

2.Admin

Employee:-

R.1 Login into the system

Description:-Users/admin can login to the system using id and password.

Input:-Role/Credentials.

Output:- successfully

R.1.new user/Admin Registration:-

Description:-Registration to the system using their details.

Input:-enter Name, Email etc,

output:-you can successfully enter the system.

R.2 forgot the password

Description:-user can change the password .

Input:-Emailid, contact number.

output:-data update successfully.

R.2 Enter the details of worker

Description:-user/admin can enter the details of worker.

R.2.1 Add new worker

Description:-user can add new worker details.

Input:- enter details of new worker

output:-data insert successfully.

R.2.2 Update details of worker.

Description:-update the details of worker.

Input:-update the details of workers.

output:-data update successfully.

R.2.3 Delete the worker details

Description:- delete the worker details.

input:-delete the details of worker.

R.3 Request for row material

R.3.1 workers can request for row material

Description: user can request for the row material according to the worker requirement for building the Assets,

Input:- user can request for row material to admin.

Output:-user can request for material.

R.4 show and manage the work status

R.4.1 show the all details of worker

Description: user can show the all work of sites which can be done by day by day status will show and change according the their instruction.

Input:-user can show the details of worker done their work per day.

Output: successful message

R.4.2 manage the work

Description:-user can mange the work according to divide the work into small parts or group and according to their it will done by the worker.

Input:-user can manage the work into the group of worker.

Output: show status of work.

R.5 manage the details via report

Description:-user can mange the details of the worker salary and their work.

Input:-user can manage the details of worker salary,date ,time and the work.

Output: show the data successfully.

R.6 manage status of site

Description:-employee can manage the site status of building assets.

Input:-it can give information about site.

Output:-report generated.

Admin:-

R.7 Login into the system

Description:-Users/admin can login to the system using id and password.

Input:-Role/Credentials.

Output:- successfully

R.7.1.new user/Admin Registration:-

Description:-Registration to the system using their details.

Input:-enter Name, Email etc,

output:-you can successfully enter the system.

R.7.2 forgot the password

Description:-user can change the password .

Input:-Emailid, contact number.

output:-data update successfully.

R.8 Manage row martial

Description:-Admin can mange the row material according to the employee requirement.

Input: Admin can manage the row martial according to request.

Output: show the requirement.

R.9 show the status

R.9.1 show the status of worker

Description: admin can show all the details of worker.

Input: show all the information about the worker.

Output: successful message.

R.9.2 show the status of employee

Description: show all the details of the employee

Input: show the details.

Output: show list of details.

R.10 Task allocation

Description:- Admin can allocate the task for employee as well as worker.

Input:- the details of task allocation handle by employee.

Output:-successfully show the details.

R.11 Approx. Budget information:-

Description:-it can show the and give the approx. budget to the end user

Input: show the details of work and it’s budget.

Output: successful message.

### Other Non-Functional Requirements

### 3.5.1 Performance Requirements

* The average response time delay must be very less. Any page of the application should not take more than 6 seconds to load
* The system may be throttled or slowed down on heavy loads to ensure service for everybody. By throttling is meant that certain functionality may be unavailable during heavy server load.
* The application should be able to support many concurrent users without any performance degradation and in future, installing additional hardware components.

### 3.5.2 Reliability/Availability Requirements

* The system has to be online 24 hours a day, 7 days a week. There is no place for an extended downtime.
* The Mean Time Between Failure (if any) should not be less than 3 months.
* In case of a failure that leads to a system outrage the Mean Time to Repair should not be more than 2 hours.

**3.5.3 Security Requirements**

* There needs to be clearly defined roles of the users. These roles are of 'administrator'. Each person that goes to the system's website will be required to register if they want to do more than just read / browse site content.
* Because of the different roles, passwords and user accounts must be implemented properly preventing from an illegal access.

### 3.5.4 Usability

### The user interface of the system should be very user friendly

### It should not take more than 120 seconds for a new user to register for an account and 90 seconds for a registered user to place an order.

### 3.5.5 Design Constraints

* Java and preferable web technologies should be used for development of the website.

### 3.5.6 Standards Constraint

* All the documents delivered should adhere to the IEEE standards for software engineering.

### 3.5.7 Legal Constraints

* All the images used on the site must be procured through legal channels and there should be no copyright violations.

### 3.5.8 Maintainability

* The system should be developed in such a way that changes can be made easily, whether for bug fixes or to add new functionality.
* The system should be easy enough to maintain that someone else could do it with a manual and a few hours training.

### 3.5.9 Portability

* The system should be portable to various operating environments.
* Should the current hosting become too restricting for the system, the system must be portable enough to be moved over to a new server with minimal downtime.

### 3.5.10 Integrity

* The system should be able to protect and preserve transactions.

### 3.5.11 Manageability

* The system should be developed in such a way that it can be easily reused, deployed and tested.

### 3.5.12 Safety Requirements

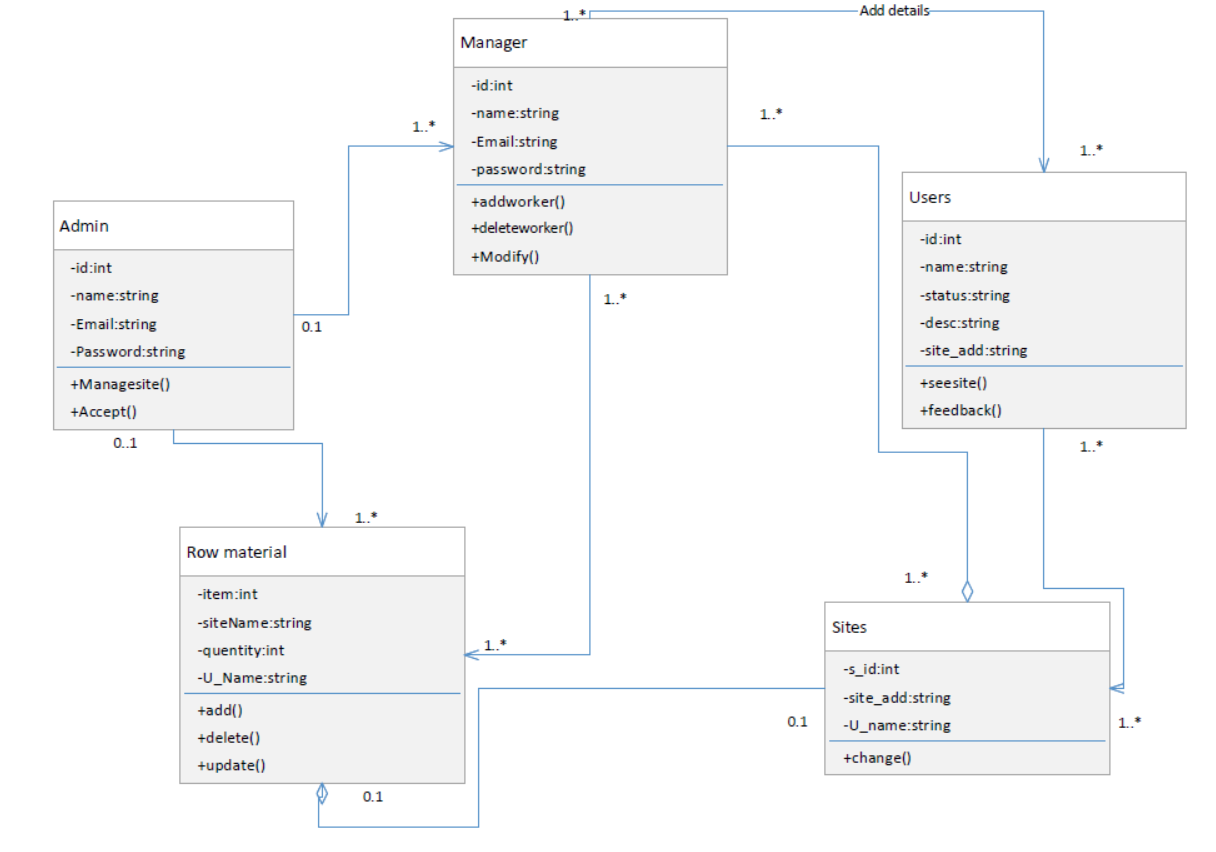
* Test cases for each problem are stored in specific location which normal user cannot access it.

### 3.5.13 Software Quality Attribute

* All the modules are developed in Java Server Pages(JSP) and HTML pages which makes system extensible, robust and reusable.
* System will provide the user very attractive and understandable UI.

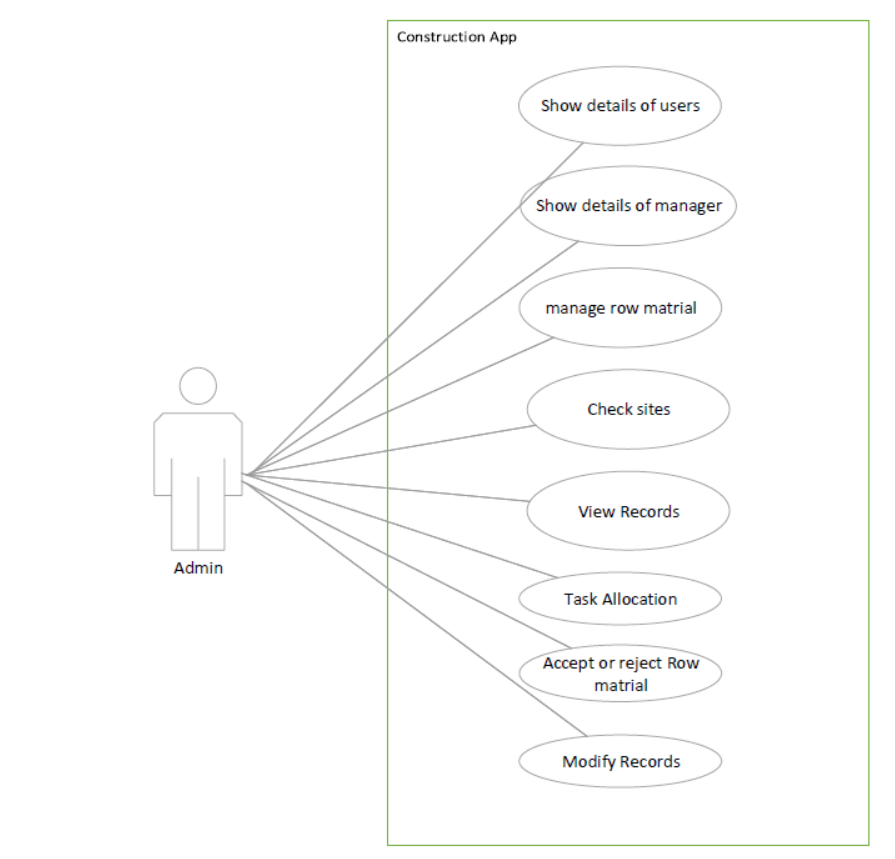
# 4.Design

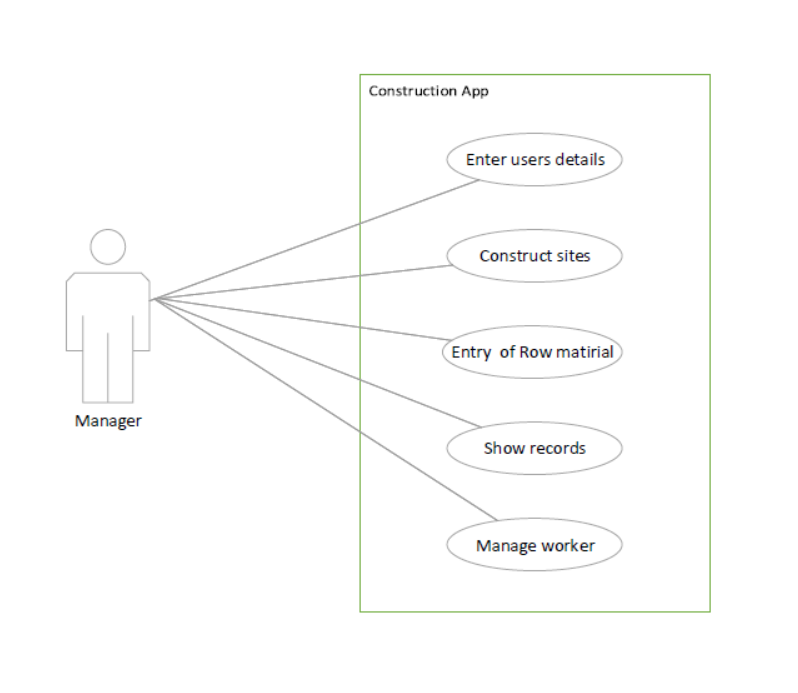
**Class Diagram**

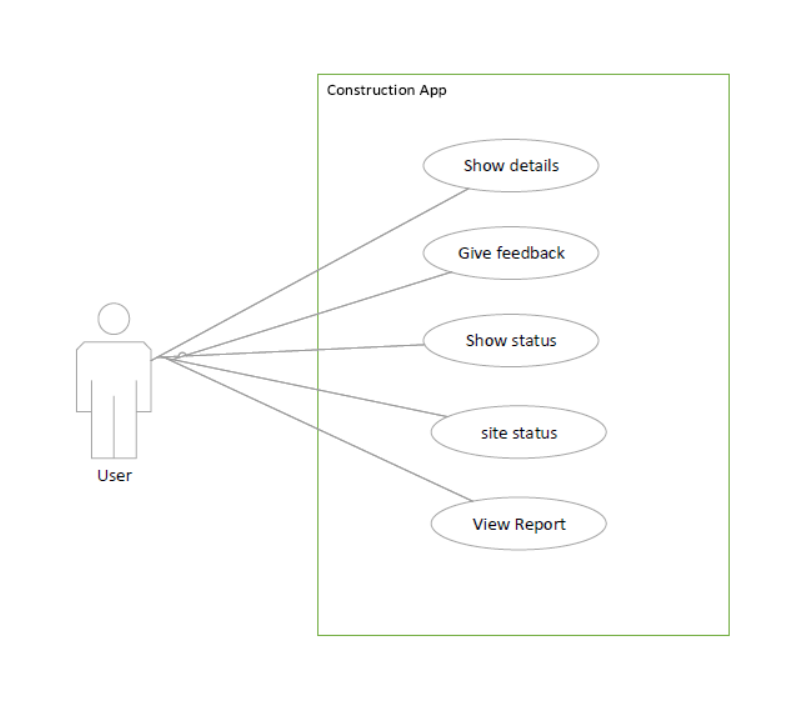


### 

### Use-Case Diagram

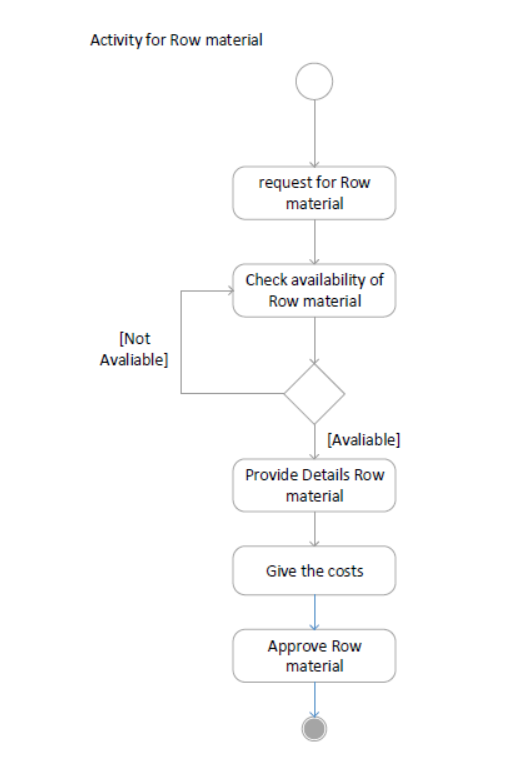


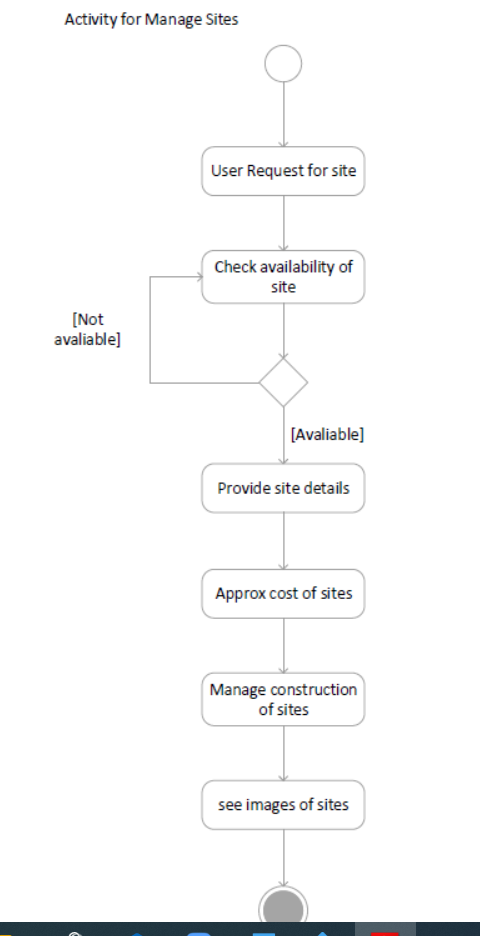




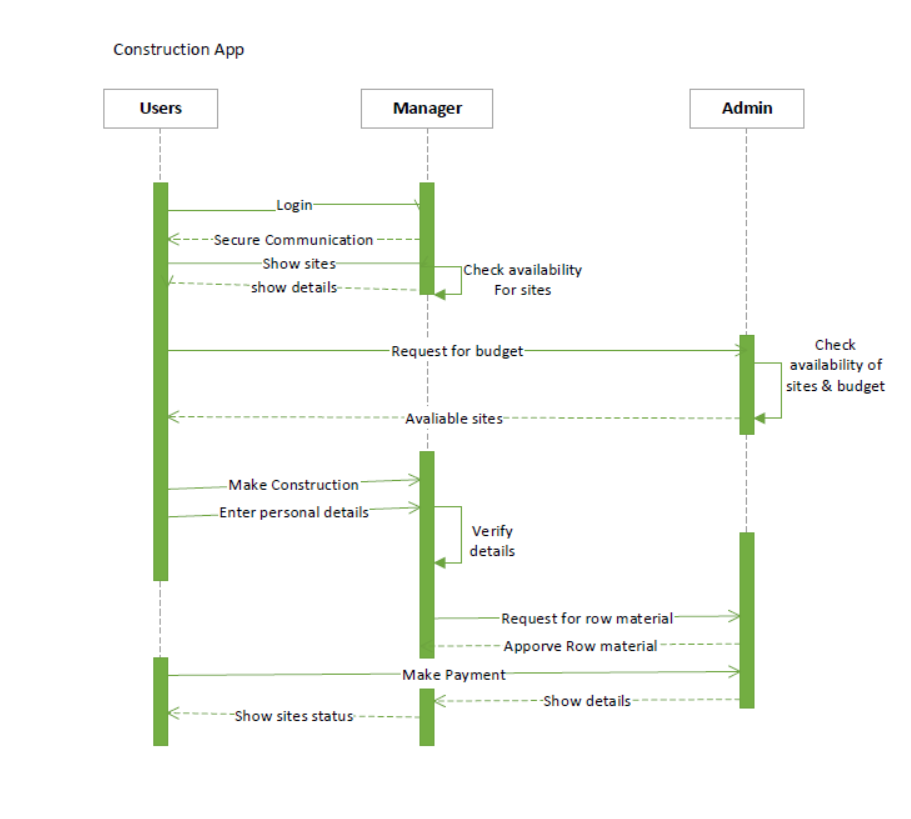
### 

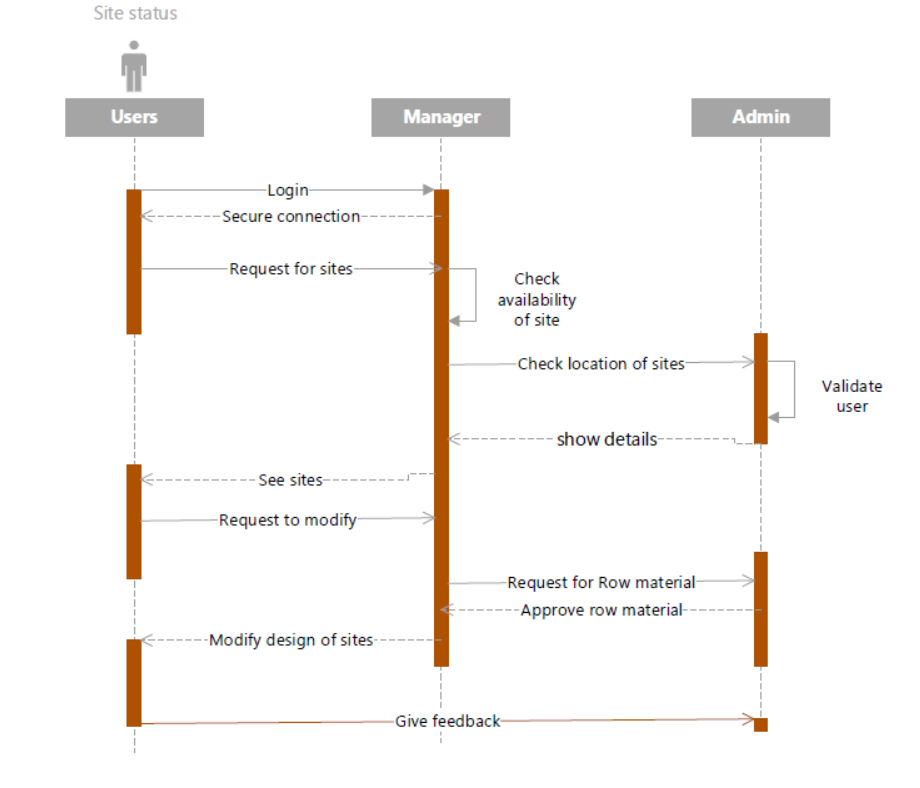
### Activity Diagram



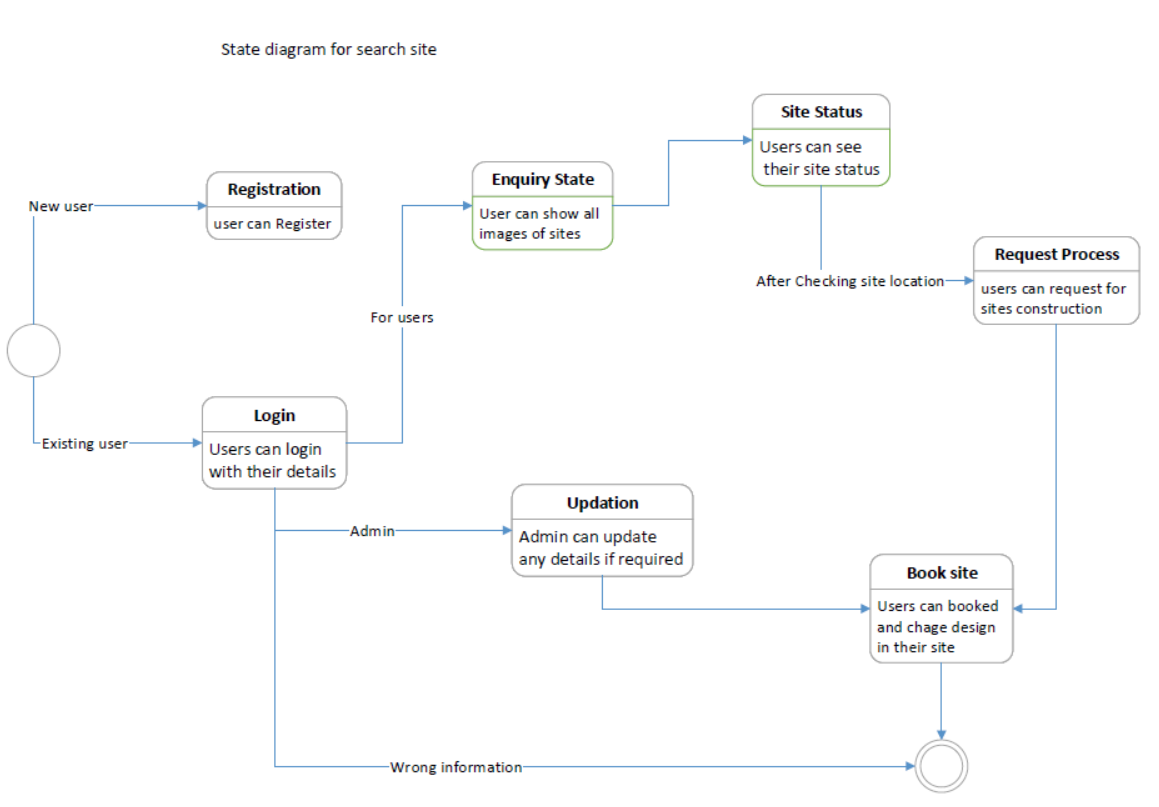


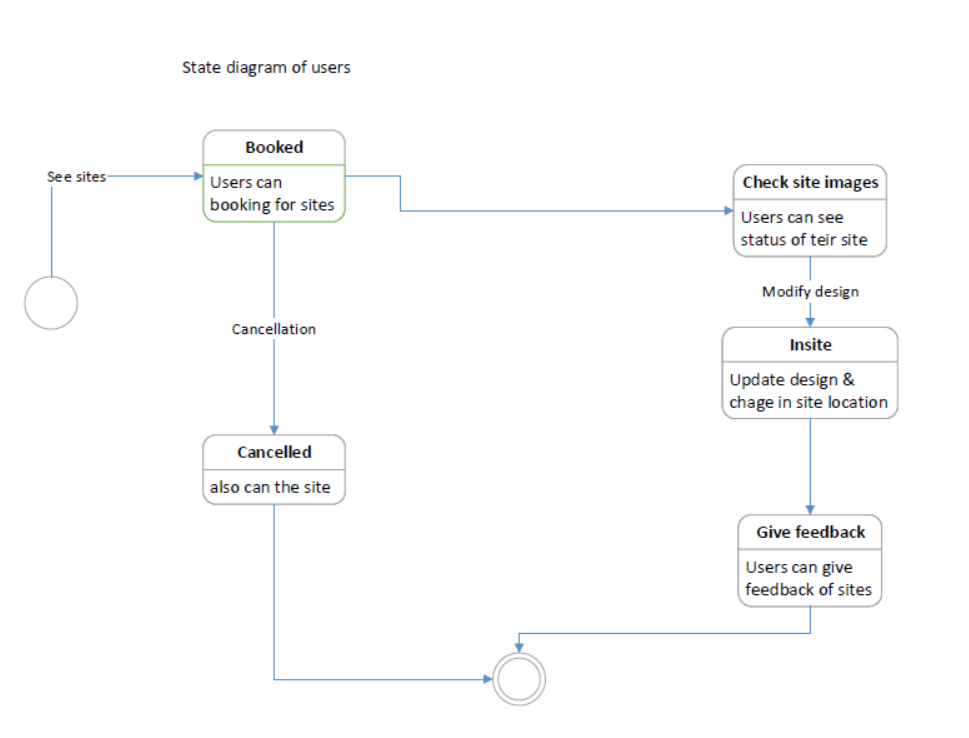
### Sequence Diagram

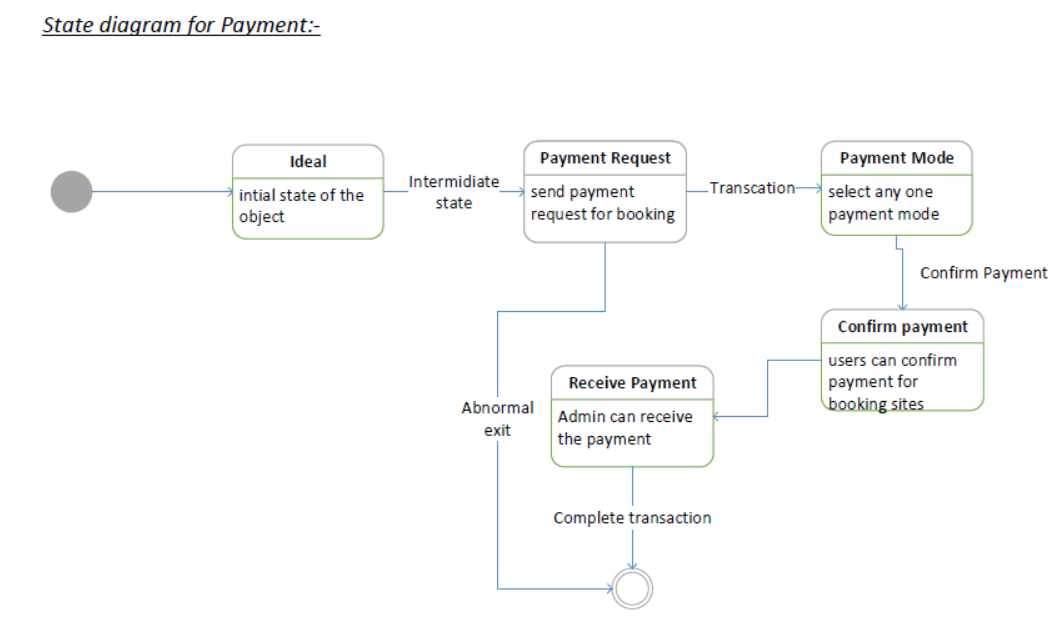




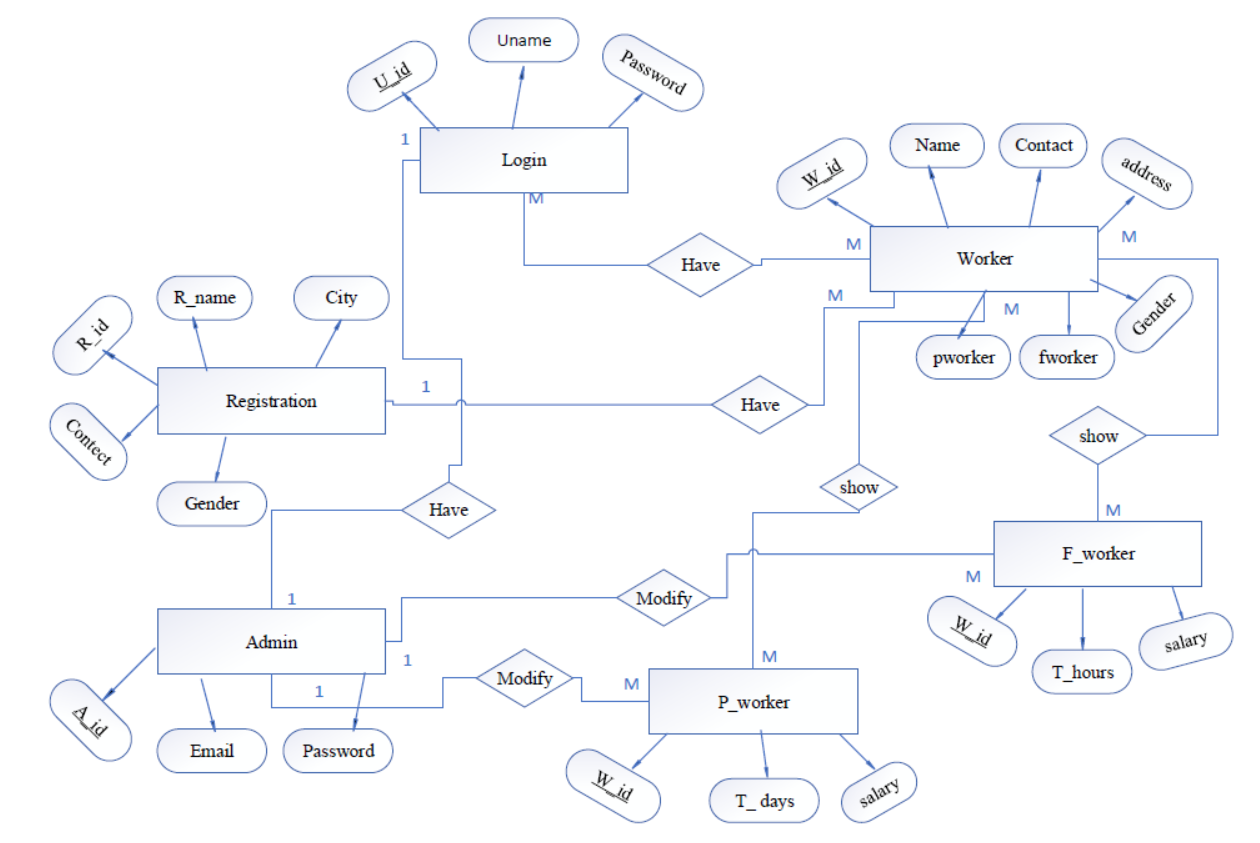
### State Diagram







### E-R Diagram



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 1. Sr\_no | Field\_name | Type | Length | Required | Null | PK/FK |
| 1. | w\_id | Int | - | Yes | No | PK |
| 2. | w\_name | Varchar | 20 | Yes | No | - |
| 3. | Contact no | Number | 10 | Yes | No | - |
| 4. | Address | Varchar | 20 | yes | No | - |
| 5. | Gender | Varchar | 10 | yes | No | - |
| 6. | F\_worker | Varchar | 10 | Yes | No | - |
| 7. | F\_worker | Varchar | 10 | Yes | No | - |

**DATA DICTORY**

1.Worker details

2. full-time workers Details

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Sr\_no | Field\_name | Type | Length | Required | Null | PK/FK |
| 1. | w\_id | Int | - | yes | No | FK |
| 2. | total\_salary | Number | 10 | yes | No | - |
| 3. | total\_days | Int | - | yes | No | - |

3. part-time workers Details

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Sr\_no | Field\_name | Type | Length | Required | Null | PK/FK |
| 1. | W\_id | Int | - | yes | No | FK |
| 2. | Total\_salary | Number | 10 | yes | No | - |
| 3. | Total\_hours | Int | - | yes | No | - |

4.login

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Sr\_no | Field\_name | Type | Length | Required | Null | PK/FK |
| 1. | U\_id | Int | - | yes | No | PK |
| 2. | U\_name | Varchar | 10 | yes | No | - |
| 3. | Pwd | Varchar | 10 | yes | No | - |

5. registration

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Sr\_no | Field\_name | Type | Length | Required | Null | PK/FK |
| 1. | r\_id | Int | - | Yes | No | PK |
| 2. | r\_name | Varchar | 20 | Yes | No | - |
| 3. | City | Varchar | 10 | Yes | No | - |
| 4. | Contact no | Number | 10 | Yes | No | - |
| 5. | gender | Varchar | 10 | Yes | No | - |

6.sites

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Sr\_no | Field\_name | Type | Length | Required | Null | PK/FK |
| 1. | S\_id | int | 10 | yes | no | Pk |
| 2. | Site\_name | varchar | 20 | yes | no | - |
| 3. | Address | varchar | 20 | yes | no | - |
| 4. | M\_id | int | 10 | yes | no | FK |
| 5. | W\_id | int | 10 | yes | no | Fk |

7.Row-Material

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Sr\_no | Field\_name | Type | Length | Required | Null | PK/FK |
| 1. | R\_id | int | 10 | yes | no | PK |
| 2. | item | string | 20 | yes | no | - |
| 3. | quantity | int | 20 | yes | no | - |
| 4. | S\_id | int | 20 | yes | no | FK |
| 5. | M\_id | int | 20 | yes | no | FK |
| 6. | cost | number | 20 | yes | no | - |

### 

# 5. Implementation

## Modules and brief description.

### Add users Module

Users can login in to the system and manage by the admin.

### Add Row material module

Manager can Request for Row martial and construct the sites.

### Approve-Reject Row martial

Admin can approve and reject the row martial Request .

### View Records

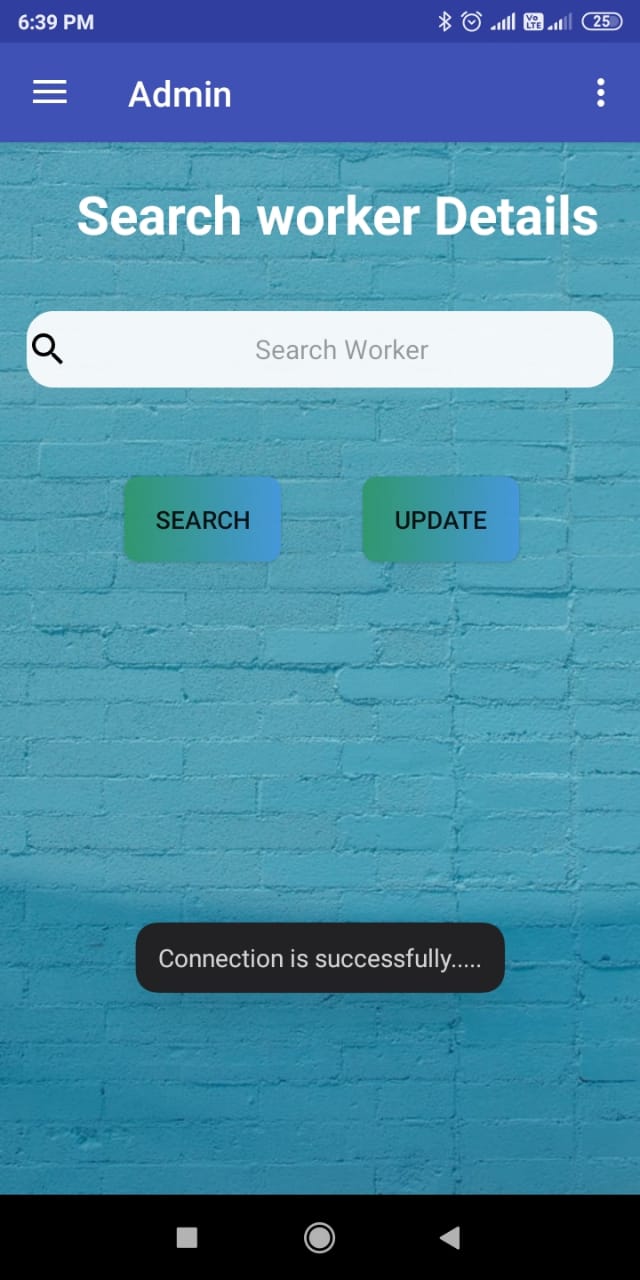
Admin can view all the records.

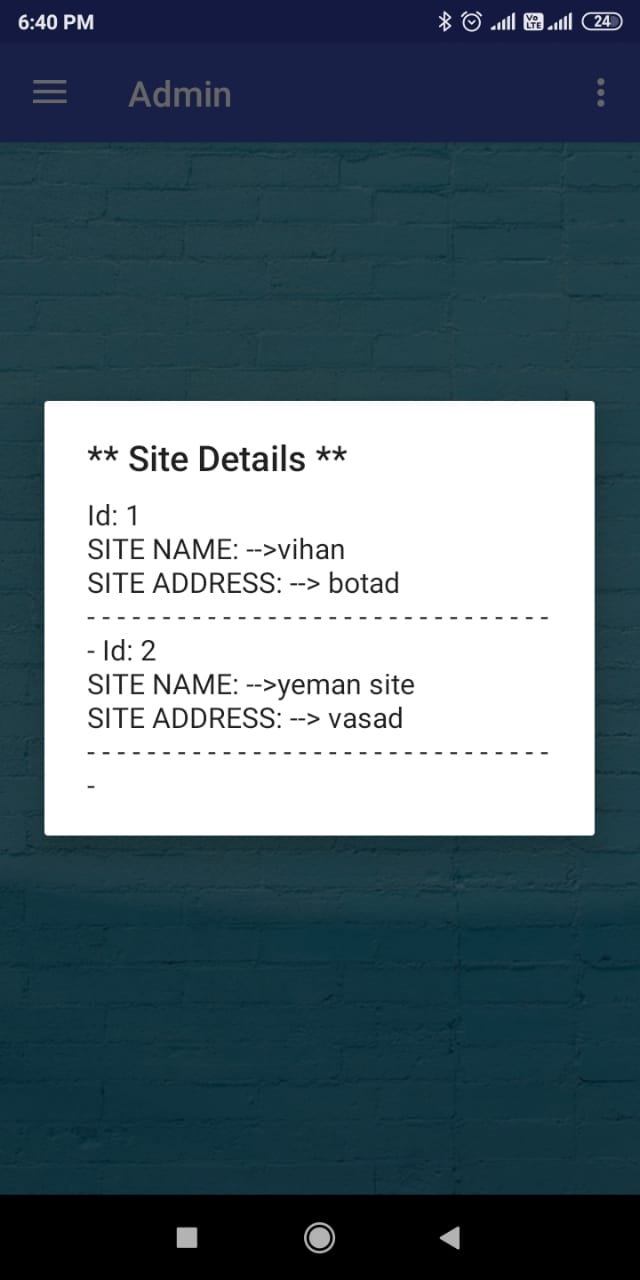
# 

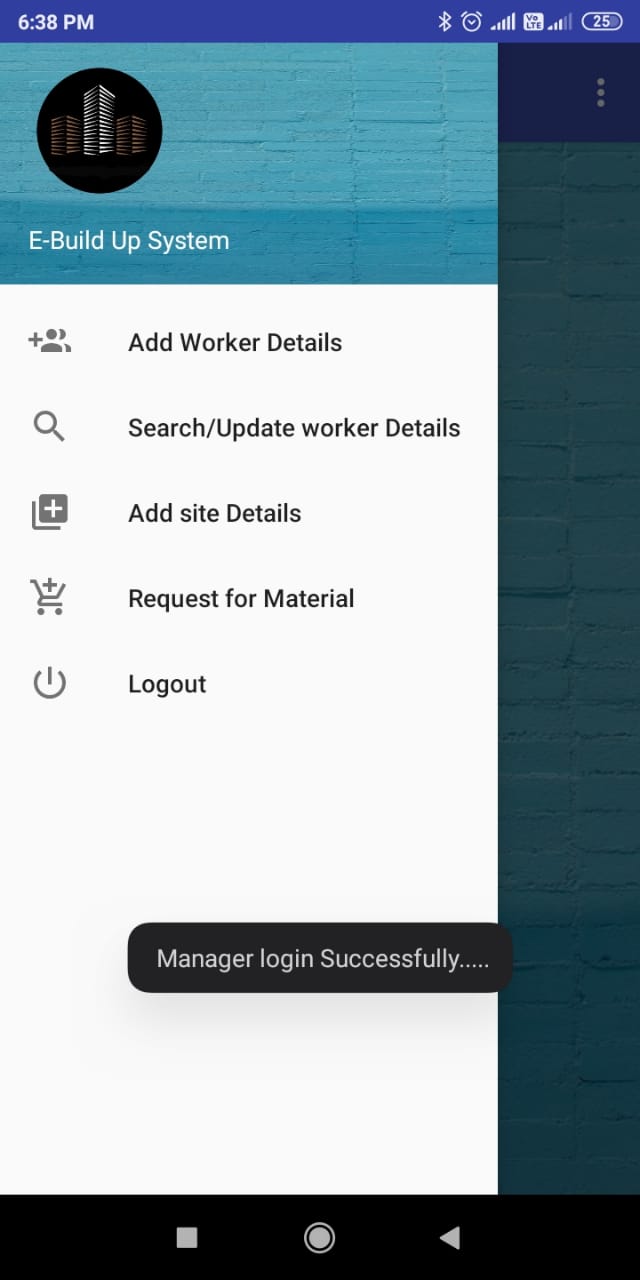
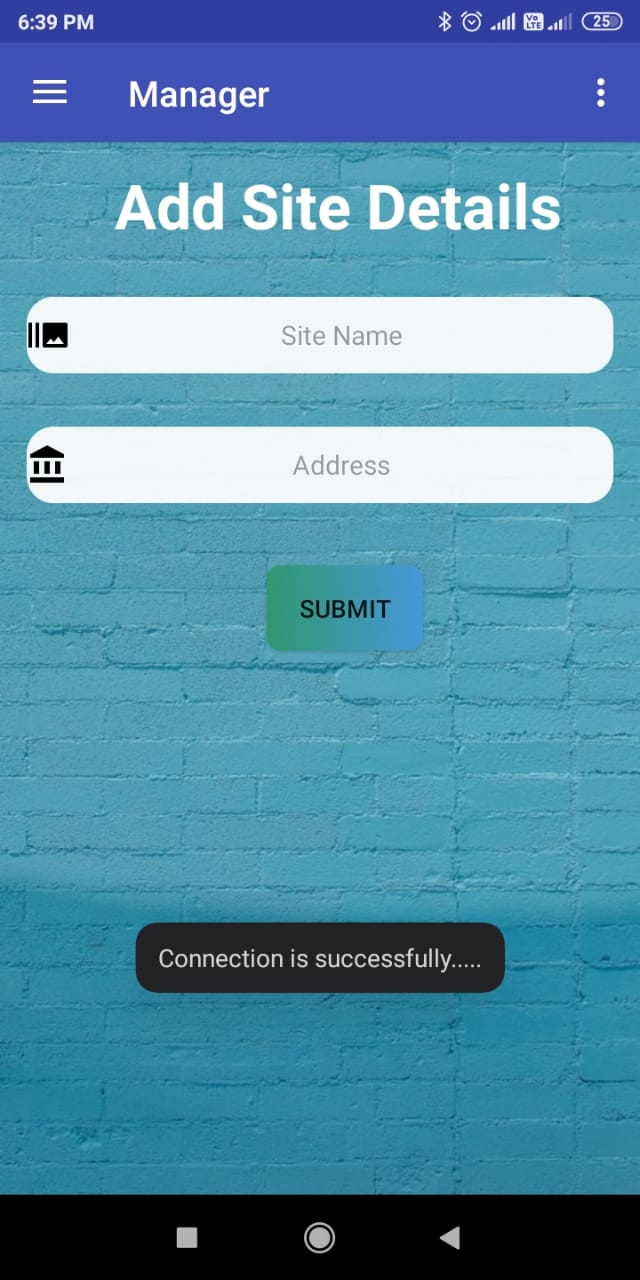
# Testing

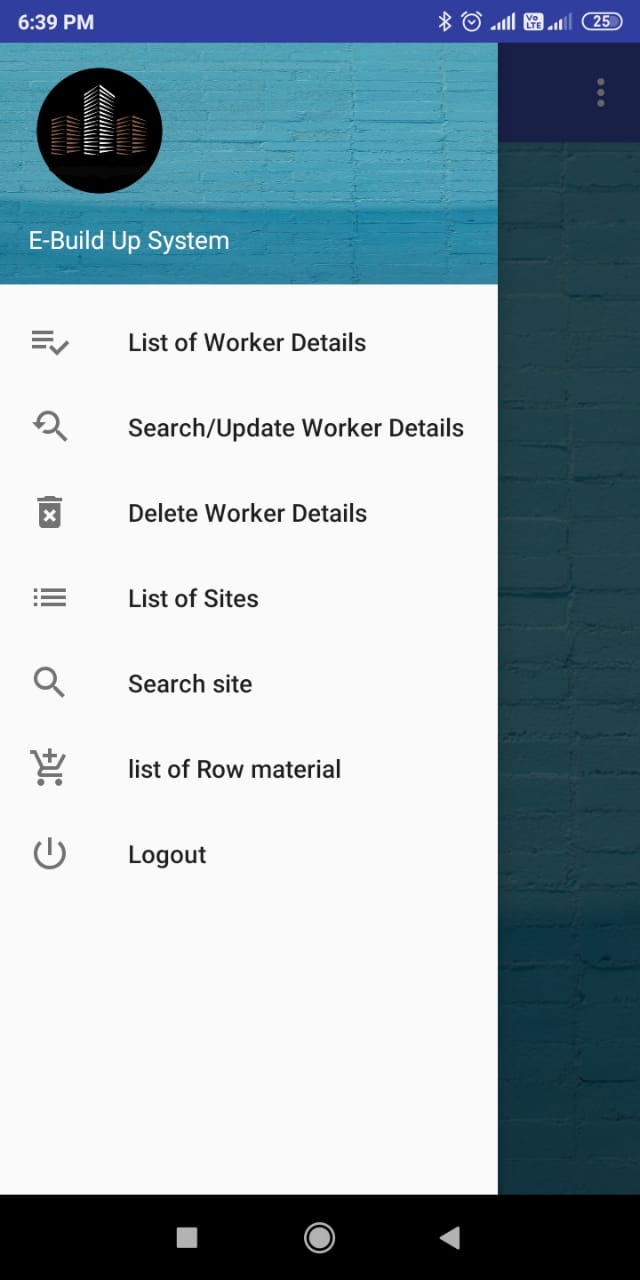
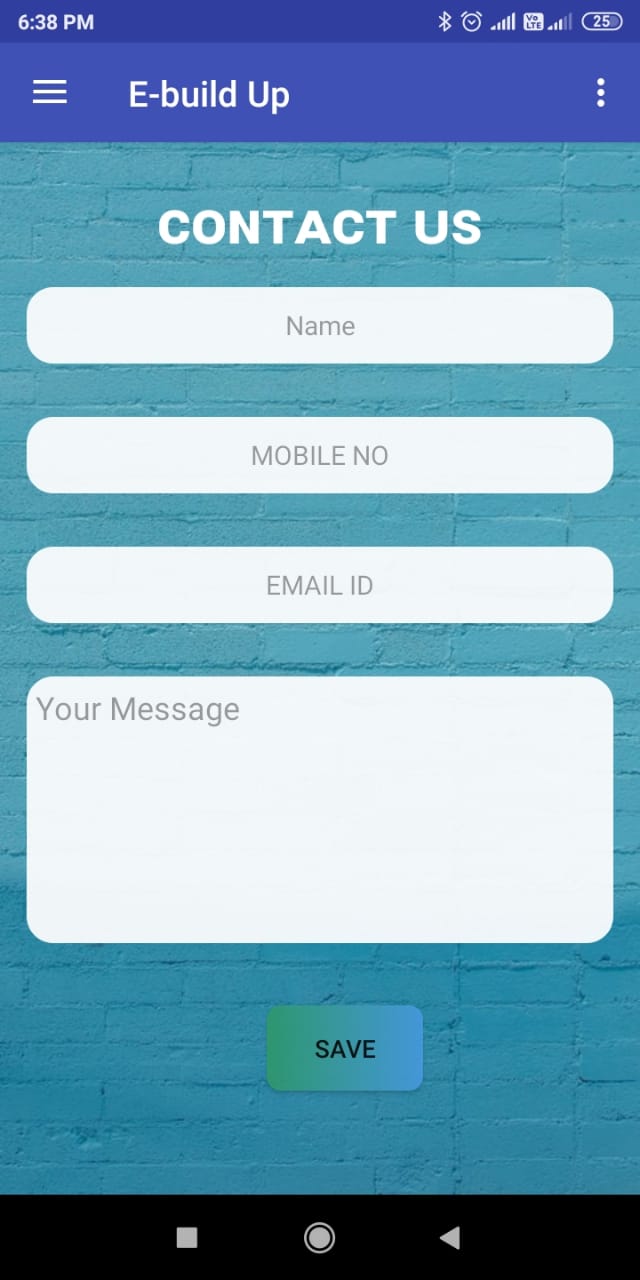
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Entity or Module** | **Test Case**  **Input** | **Expected Output** | **Actual Output** | **Result** |
| Add  User | User  details | “success” | “success” | PASS |
| Add  Worker | Correct credentials | Entry in  Worker table | Entry in worker details table | PASS |
| Add row  martial | Row material Request for site | Approve the data from admin | Request show in manager | PASS |

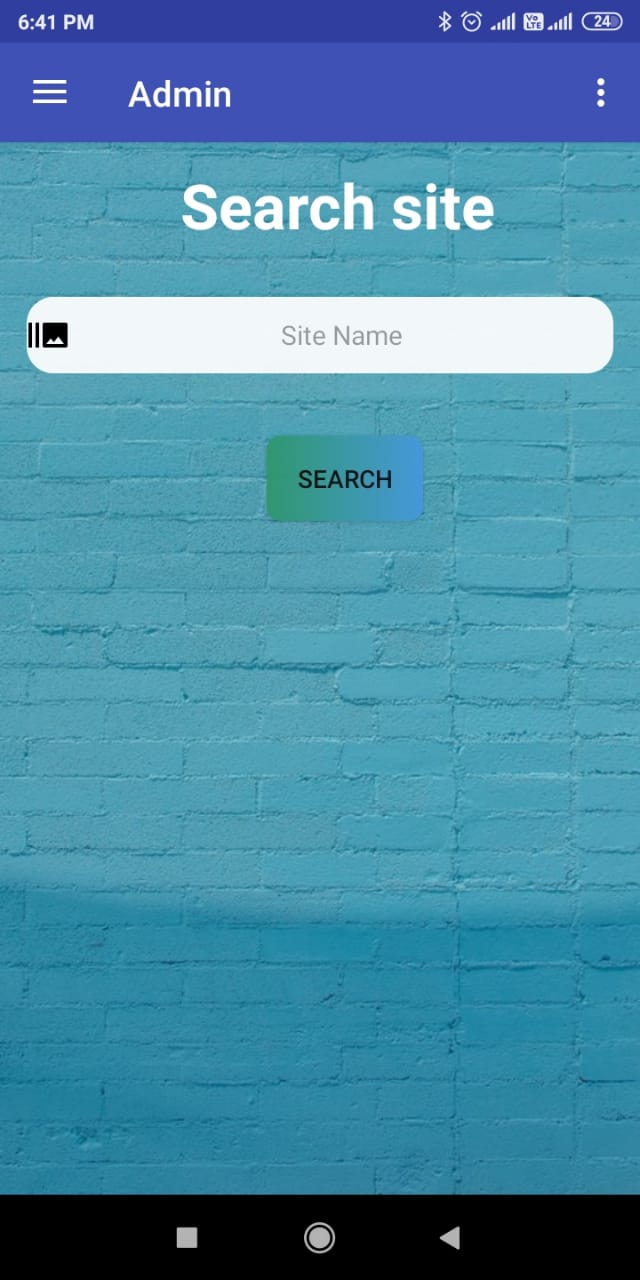
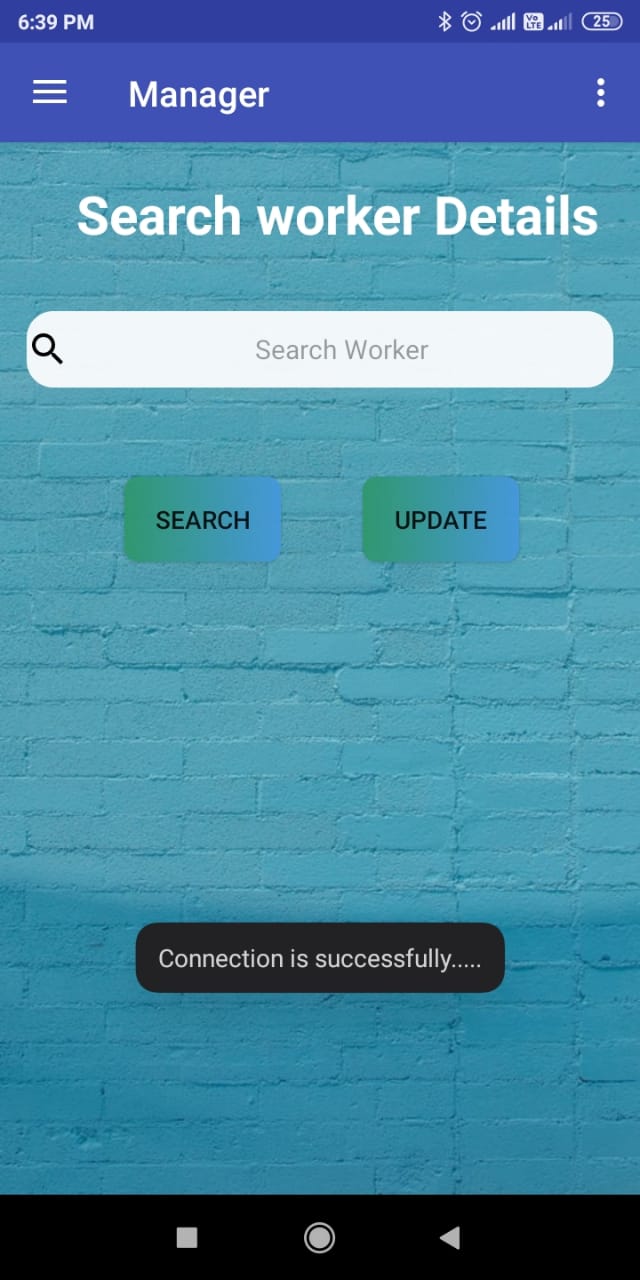
**Screenshots**

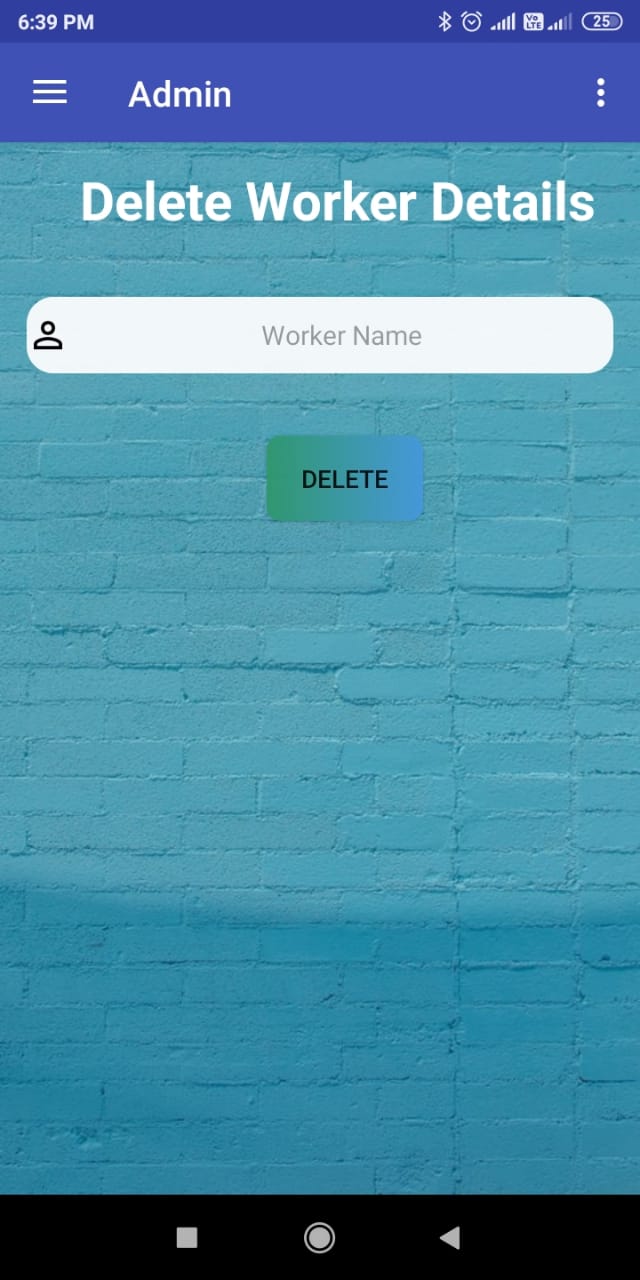
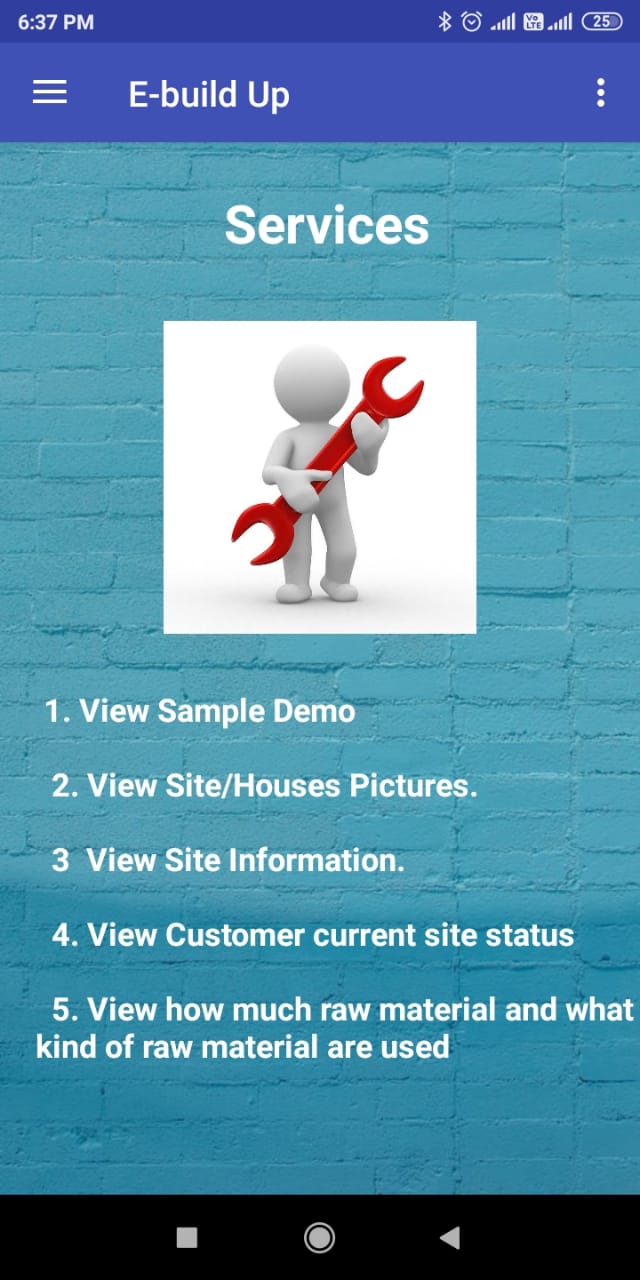


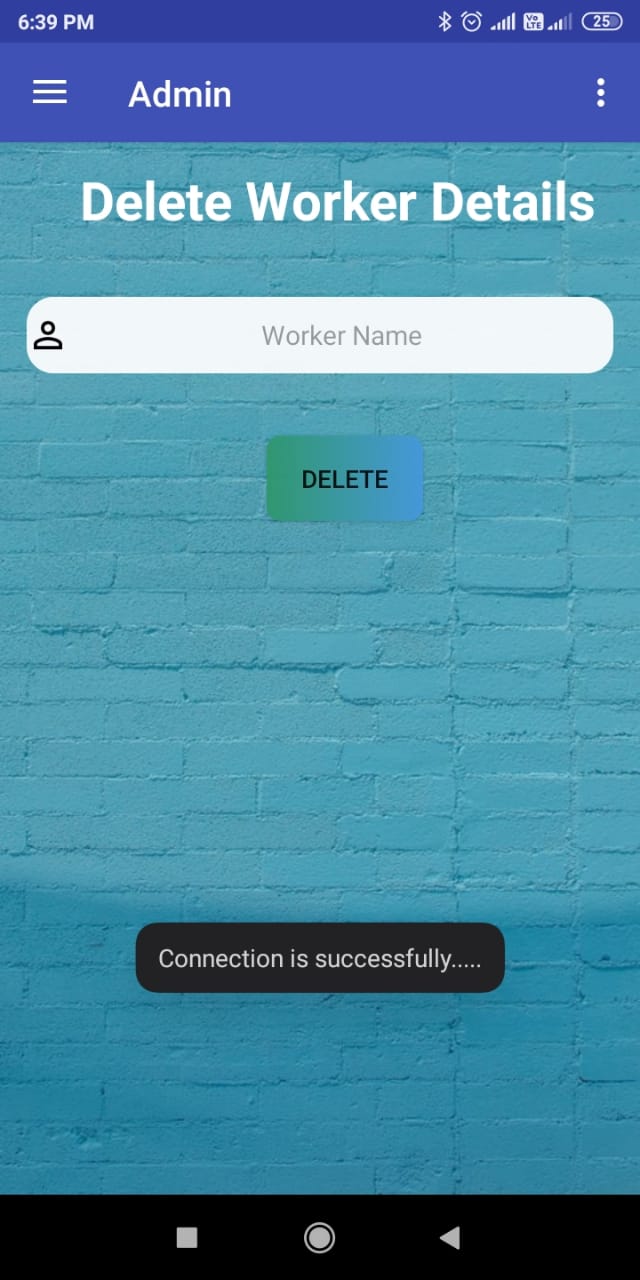


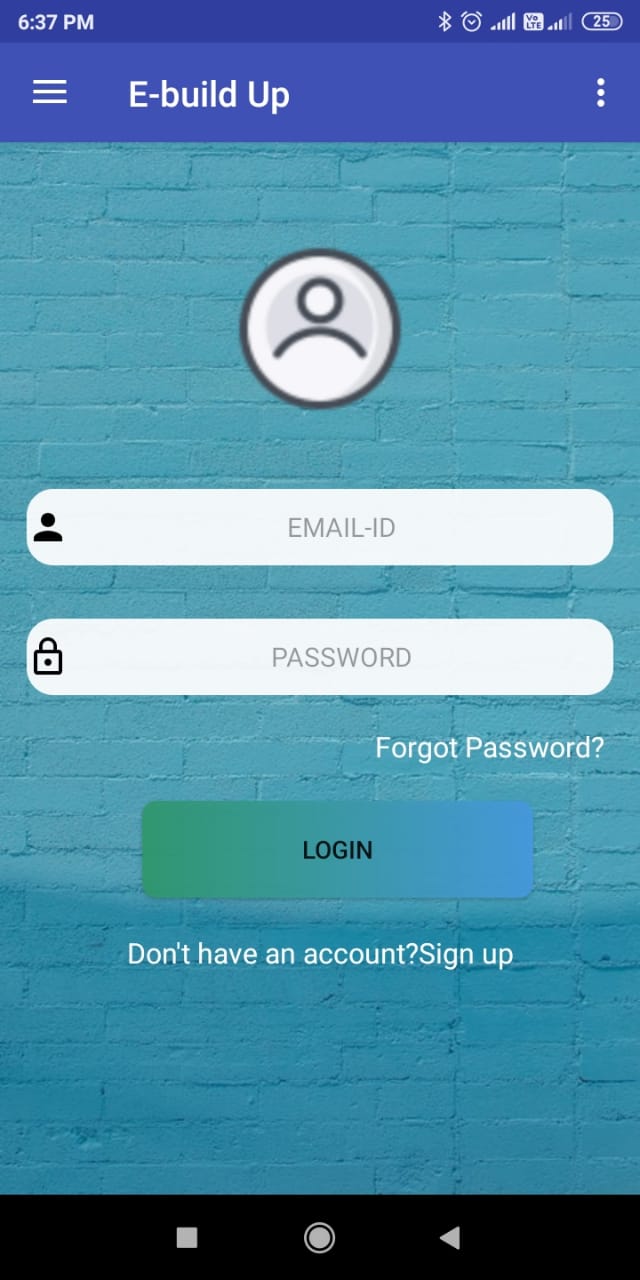
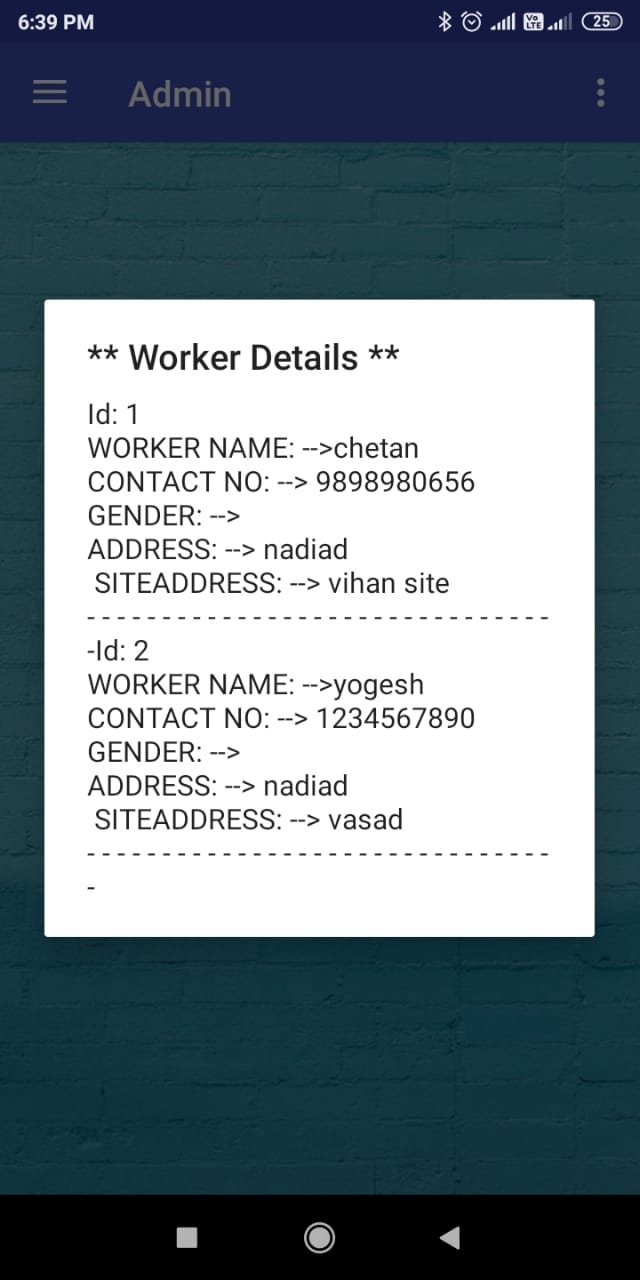
 

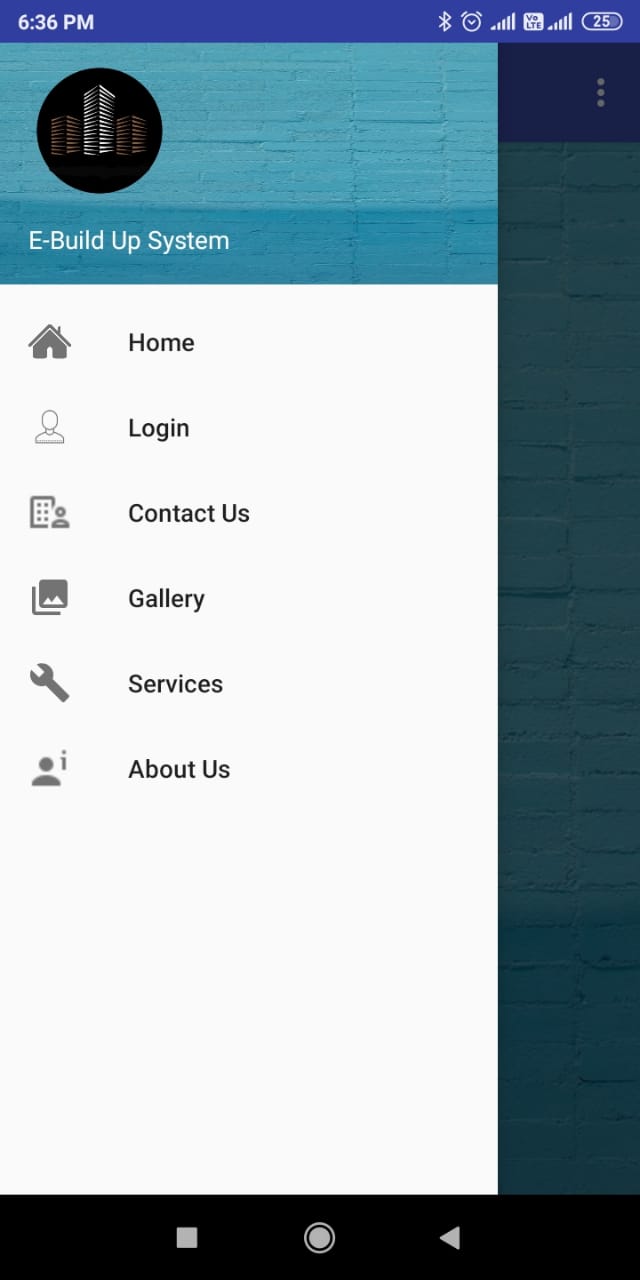
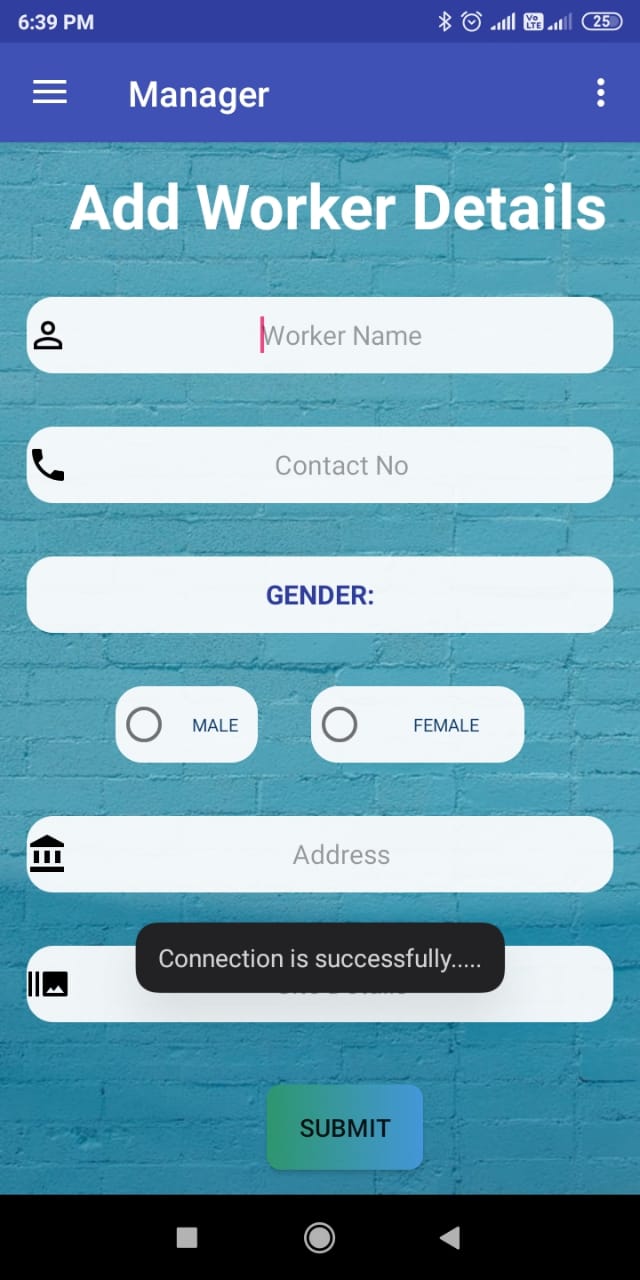
 





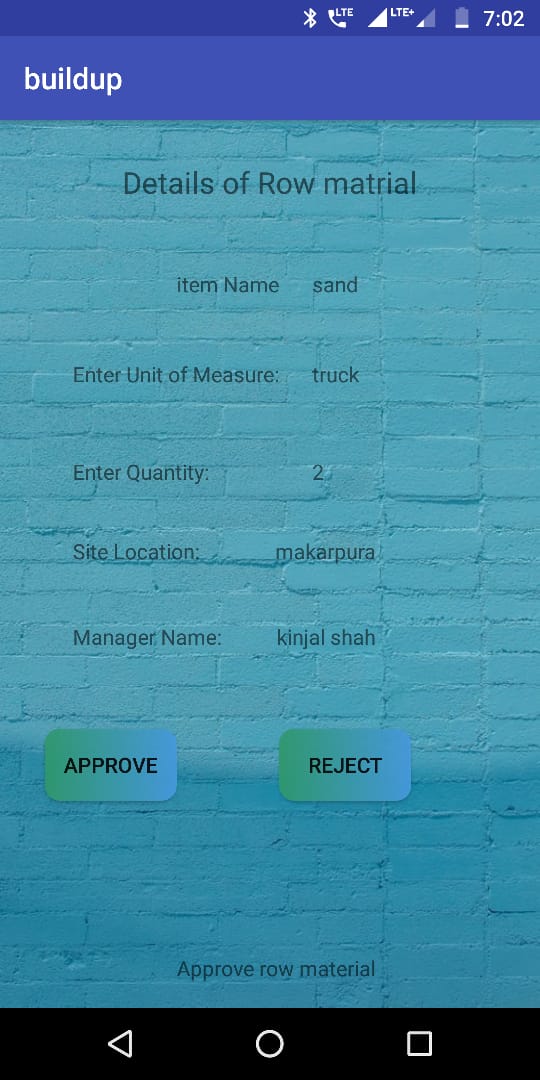
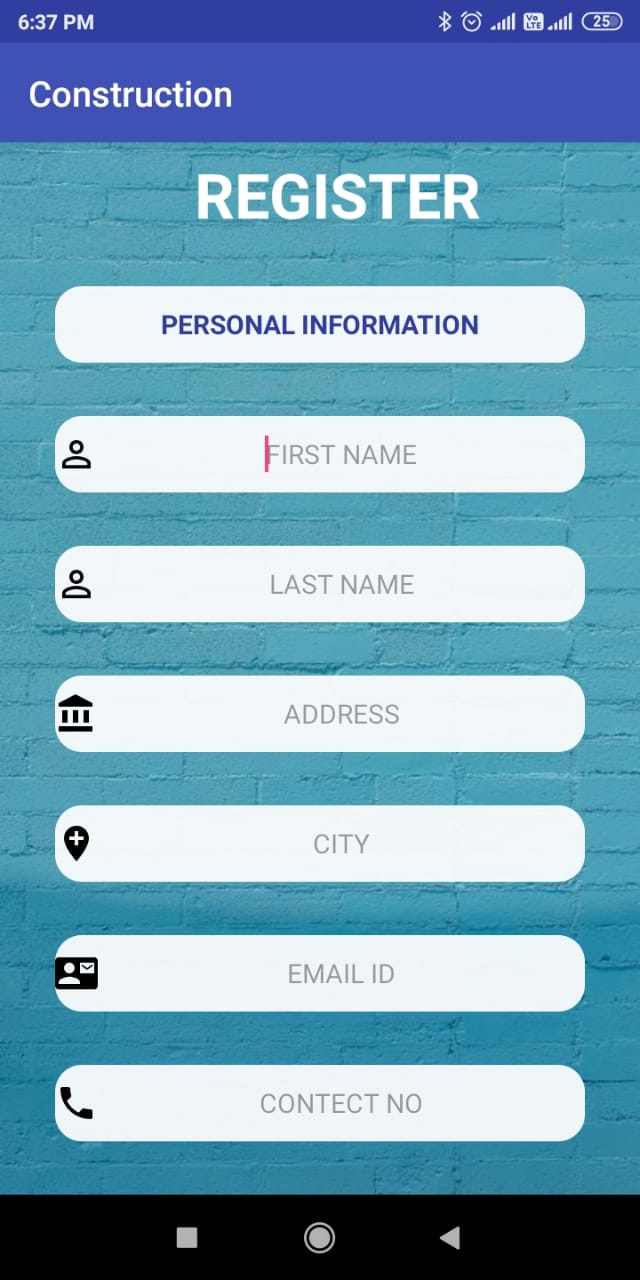










# Conclusion

Development of project was started after gathering all requirements, after gathering requirements overall flow control was defined, all modules were developed separately and then they were integrated. All modules are tested separately as well as after integration.

# 8. Limitations and Future Enhancement

### Limitations

* + - No Interaction of users.
    - This application only support the Android devices.

### Future Enhancement

* Next release may have Payment module.
* We can extend the system using the tracing of the site statues.
* And also may have release the salary of the particular worker.

# 

# 9.Bibliography

### Referred Websites:

* [**www.github.com**](http://www.github.com)
* [**www.stackoverflow.com**](http://www.stackoverflow.com)
* [**www.w3school.com**](http://www.w3school.com)
* <https://github.com/>
* <http://www.Kashipara.com>