A Project Report on

Laundry Service

By

Adity Salunkhe

MCA – II, SEM – III, DIV-C

2021-2022

To

Savitribai Phule Pune University

Pune- 411041

In Partial Fulfillment of the Degree of

Master in Computer Application (M. C. A.)

Under The Guidance Of

Prof. Balchandra Doddi

Sinhgad Technical Education Society's

SINHGAD INSTITUTE OF MANAGEMENT,

Vadgaon Bk, Pune

(Affiliated to Savitribai Phule Pune University, Approved by AICTE

& Accredited by National Board of Accreditation, New Delhi)

Date:-

CERTIFICATE

This is to certify that Mr. / Ms. _____, has successfully / partially completed his/her project work entitled **"_____"** in partial fulfillment of MCA-II SEM-III Mini Project for the year 2021-2022. He / She have worked under our guidance and direction.

Prof. Balchandra Doddi	**Dr. Chandrani Singh**
Project Guide	**Director, SIOM-MCA**

Examiner 1	**Examiner 2**

Date:

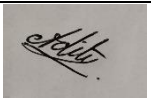
Place:

DECLARATION

We certify that the work contained in this report is original and has been done by us under the guidance of my supervisor(s).

- The work has not been submitted to any other Institute for any degree or diploma.
- We have followed the guidelines provided by the Institute in preparing the report.
- We have conformed to the norms and guidelines given in the Ethical Code of Conduct of the Institute.
- Whenever we have used materials (data, theoretical analysis, figures, and text) from other sources, we have given due credit to them by citing them in the text of the report and giving their details in the references.

Name and Signature of Project Team Members:

Sr. No.	Seat No.	Name of students	Signature of students
01	348	Adity Salunkhe	

ACKNOWLEDGEMENT

We have immense pleasure in expressing our sincerest and deepest sense of gratitude towards our guide Ms. _____ for the assistance, valuable guidance and co- operation in carrying out this Project successfully. We have developed this project with the help of Faculty members of our institute and we are extremely grateful to all of them. We also take this opportunity to thank Head of the Department Dr. Chandrani Singh, for providing the required facilities in completing this project. We are greatly thankful to our parents, friends and faculty members for their motivation, guidance and help whenever needed.

Thank You,

Student Name

SINHGAD INSTITUTE OF MANAGEMENT

PUNE-411 041

Index

Contents	Page number
CHAPTER 1 : INTRODUCTION	
1.1 Existing System	
1.2 Need for System	
1.3 Operating Environment Hardware and Software	
CHAPTER 2 : PROPOSED SYSTEM	
2.1 Proposed System (Introduction of system)	
2.2 Module specifications (Scope)	
2.3 Objectives of System	
CHAPTER 3 : ANALYSIS & DESIGN	
3.1 Use Case Diagrams	
3.2 Activity Diagram	
3.3 Class Diagram	
3.3 Module Hierarchy Diagram	
3.4 Table specifications (Database design)	
3.5 Data dictionary	
CHAPTER 4 : USER MANUAL	
4.1 User Interface Screens (Input)	
4.2 Output Screens with data	
4.3 Data Reports	
4.4 Sample program code	
4.5 Limitations and Bibliography	

INTRODUCTION

Abstract based laundry app brings Laundry services easy to you. It offers convenient and hassle free way of getting your wash, laundry and dry cleaning done.

Sit back and relax now because now you can schedule clothes washing , ironing and dry cleaning on demand though your mobile app at the most affordable rates. In the age of time scarcity and instant gratification, what we want , is one touch access to all. And here , the laundry app comes to the rescue! We are here to relieve you and your chores.

Using this app You will get your wash & Dry Clean ready and delivered very conveniently.

User need to create a account in app first. They get their credential details. After login , User can view the laundry services and pick a service for themselves.

EXISTING SYSTEM AND NEED FOR SYSTEM

Existing System

All the existing system are work in manually. Customers place the order directly to the laundry station and ask for their services which are available with charges details.
Existing are not efficient to take care of laundry service, so we are implementing a user friendly app for this purpose.

Need For System

Laundry Services app will list all the local laundry service providers. When a customer places the order, you will obtain their complete information about user whom the order will be delivered.

Scope

Laundry apps are the latest addition to the on-demand app services. These apps have made an influence on our lives and continue to expand over other services. People do not prefer to spare their time washing clothes instead of doing their favourite activities. Experts predict that the usage of on-demand laundry apps will increase to a great extent in the near future. So if you are a budding entrepreneur laundry apps can be a great idea to start your venture. Get in touch with a laundry app development company to learn more about the development process and cost of developing the application.

PROPOSED SYSTEM

Proposed System composed of a app with admin monitors the work using the admin panel in application. Users and admin interacts with the system through application.
User can view all the services and pick the one he needs.
User can do complaints if require and contact through app.
Admin can view the User details.
Handle order details

I. Washing orders list
II. Dry Cleaning orders list
III. Ironing orders list

Modules:
User Registration
Admin
Ordering for washing clothes
Ordering for dry cleaning clothes
Ordering for ironing clothes
Order Summary

MODULE SPECIFICATIONS

Modules

- Category
  - Inserting Orders category
- Washing of clothes
  - Displaying Categories for washing.
  - Book no. of orders for washing types of clothes
  - Checkout and place a orders.
- Dry Cleaning of clothes
  - Displaying types of clothes for Dry Cleaning.
  - Book no. of orders for dry cleaning for types of clothes
  - Checkout and place a orders.
- Ironing clothes
  - Displaying Categories for ironing clothes.
  - Book no. of orders for ironing types of clothes
  - Checkout and place a orders.

- Orders List
- Registered Users list

OBJECTIVE OF SYSTEM

The main motive of building Laundry Application is **to provide employment**. Those washer-men who are unemployed or which have less work. This Application is also helpful for working people, student or those people who are staying outside their houses and they have no time to wash their clothes.

OPERATING ENVIRONMENT

HARDWARE

Processor : Pentium-II or higher

Processor Speed :533 MHZ

Hard Disk Space :20 GB (min.)

Ram Memory :32 MB (64 MB recommended)

Android device

SOFTWARE

Operating System : Windows Operating System

Programming Language : Java

Front End Software : Xml

Database Server : SQLite

IDE : Android Studio 4.2

Android : Pixel 4 API 30

Client Side

HARDWARE

- Minimum of 8gb to 16gb storage in mobile.
- 4G mobile phones or android.

SOFTWARE

Operating System : PlayStore, Android 8.0 Oreo

SYSTEM ANALYSIS

FEASIBILITY STUDY

By using '**Online Laundry service '** users can login or register.

There are three aspects in the feasibility study:

- Technical Feasibility
- Economical Feasibility
- Operational Feasibility

Technical feasibility:

Our database's purpose is to create, establish and maintain a workflow among various entities in order to facilitate all concerned users in their various capabilities or roles. Permission to the users would be granted based on the roles specified. Therefore, it provides the technical guarantee of accuracy, reliability and security. The work for the project is done with the current equipment and existing software technology. Necessary bandwidth exists for providing a fast feedback to the users irrespective of the number of users using the system.

Economical feasibility:

As no paper work is required, no new costly software is required so the software cost is less and there are more gains by using the computer system, because of the other extra resources. There are sufficient benefits in creating the system to make the cost acceptable. It would be beneficial because only one-time development efforts required. Today all people are aware of technologies so no special skills required to run the system.

Operational feasibility:

Operational feasibility means considering whether it is feasible to operate the system in our organization environment. In operational feasibility, the interaction with software, very user friendly and accessing speed is faster.

The *Laundry Service Application* is user friendly anyone can easily understand and use this product.

Need of feasibility study

- It determines the potential of the existing system.
- It finds or determines all the problem of existing system.
- To determine all the goals of the system.
- It finds all possible solution of the problem of existing system (that becomes proposed system).
- It finds the technology required to solve these problems
- It determines really which solution is easy for operation from the point of view of customer or employees such that it requires very less time with 100% accuracy.

USER REQUIREMENTS

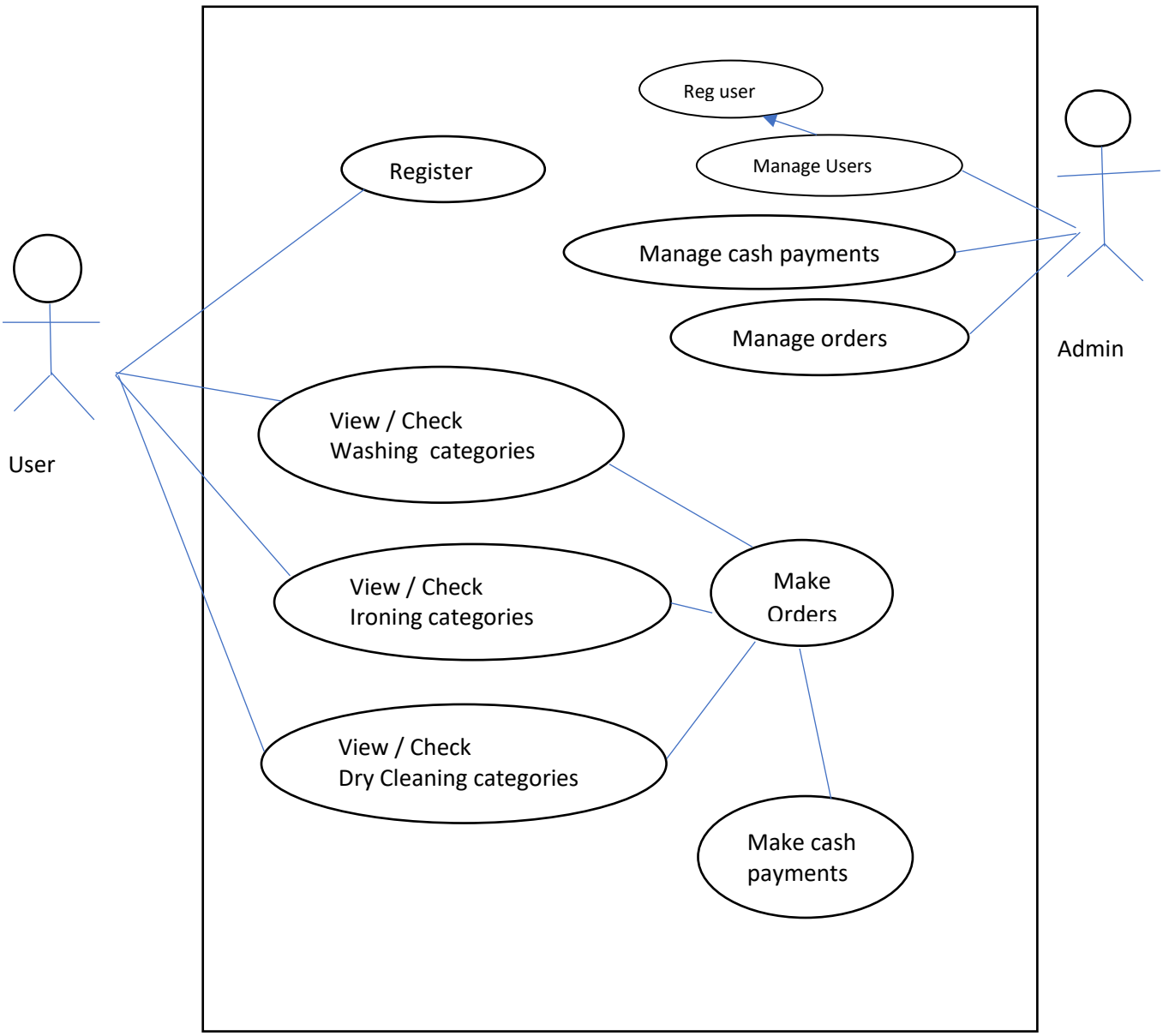
A good set of user requirements are needed for any project, especially computer system projects, to be successful. This is where many projects fail, in that they do not specify correctly what the system should do. In fact many systems have just been given a deadline for delivery, a budget to spend, and a vague notion of what it should do.

The root of this problem is:

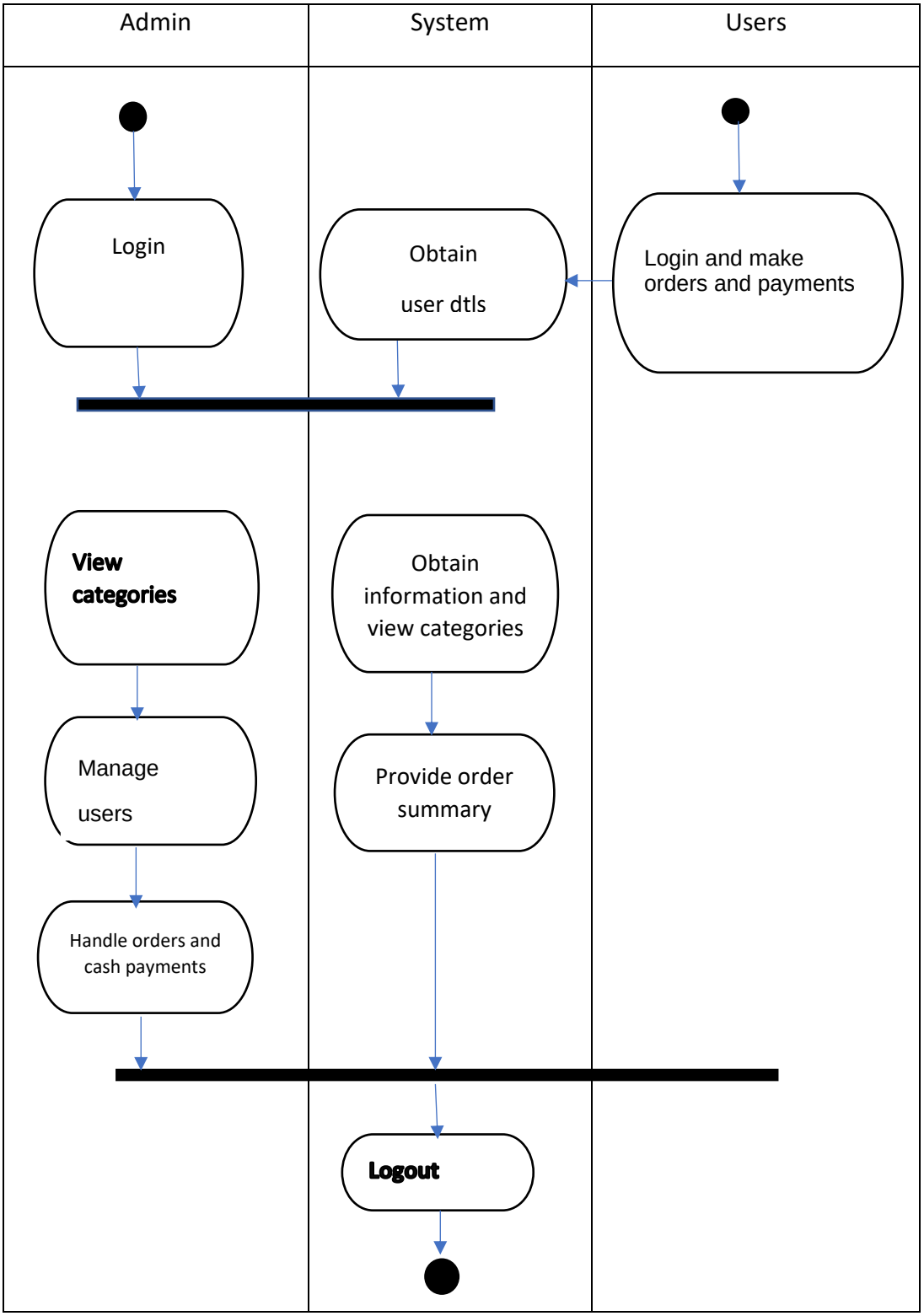
- Computer systems developers rarely have as good an idea of how a business runs and should run, compared with a business user,
- Business users have little idea of what a computer system could achieve for them.

As a result paralysis sets in and business management time is concentrated on meeting timescales and budgets, rather than what is going to be delivered.

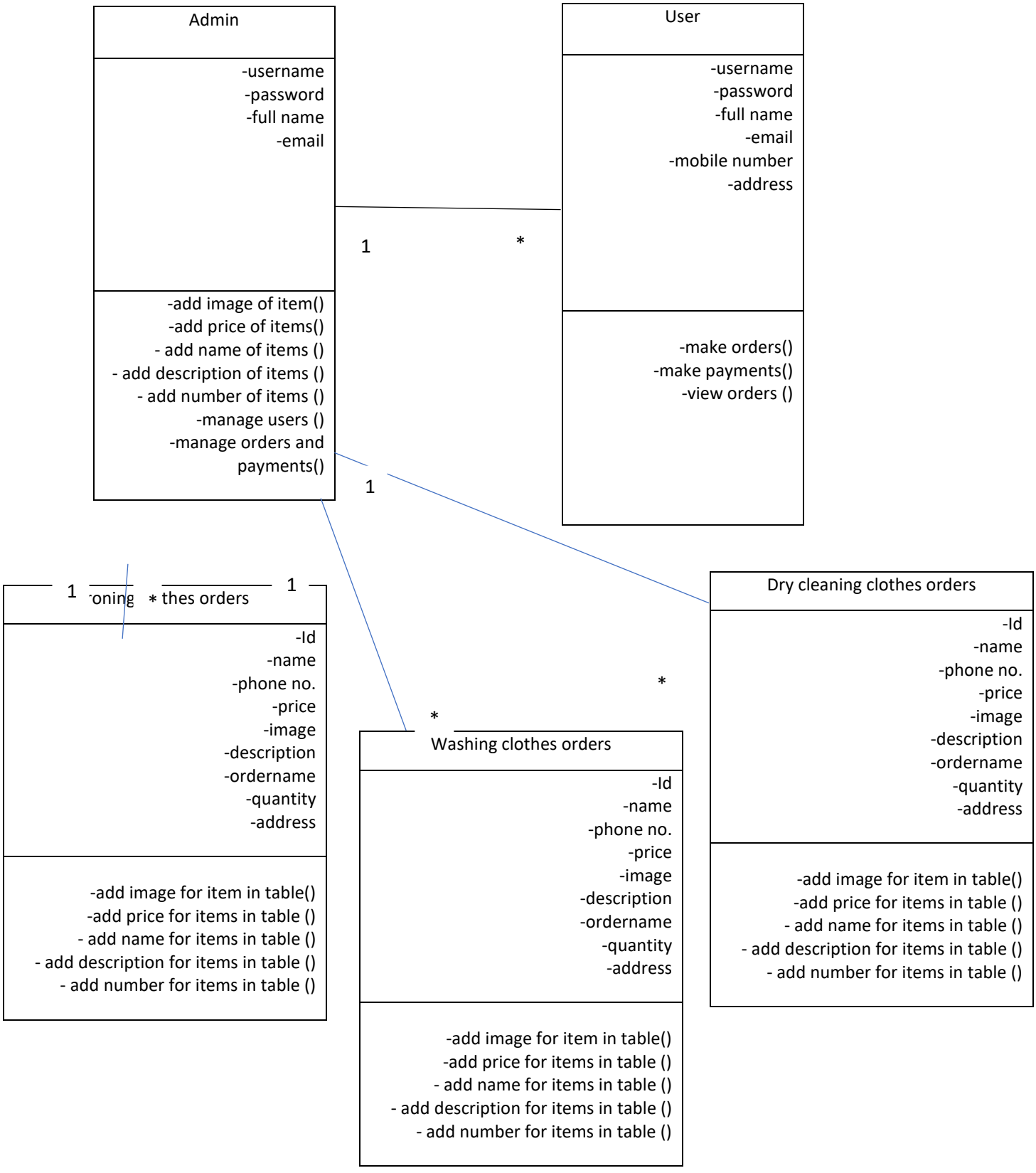
USE CASE DIAGRAM



ACTIVITY DIAGRAM



Class diagram



MODULE HIERARCHY DIAGRAM

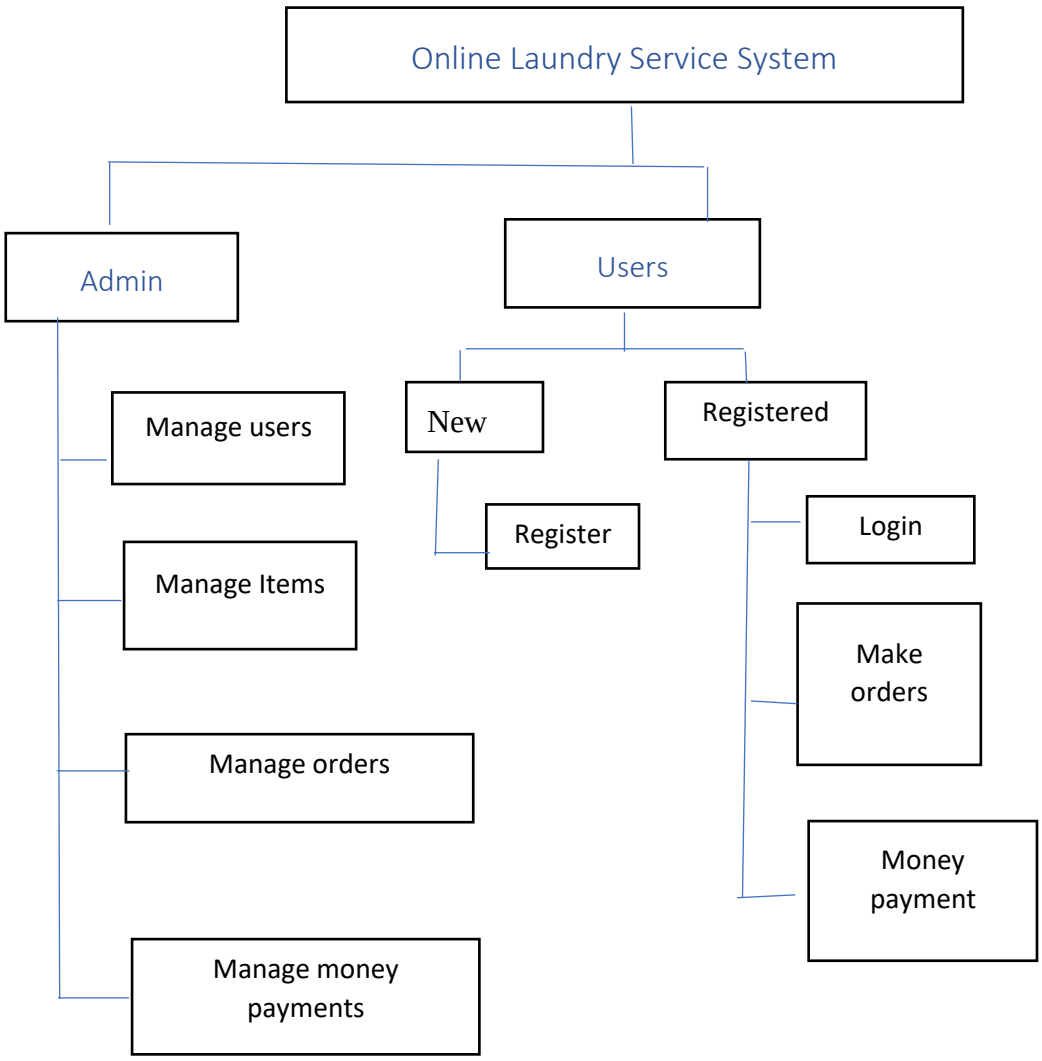


TABLE SPECIFICATION

Table name: User

Field Name	Field Type	Description	Constraints
id	Integer	Users id	Primary key
username	Varchar	Users username	Unique key
email	Varchar	Users mail	Unique key
password	Varchar	Users password	

Table name: items

Field Name	Field Type	Description	Constraints
id	Integer	id	Primary key
name	Varchar	name	
price	Integer	Price	
description	text	Description of item	
image	string	Image of item	

Table name: Washing orders

Field Name	Field Type	Description	Constraints
id	Integer	id	Primary key
name	Varchar	name	
price	Integer	Price	
description	text	Description of item	
image	string	Image of item	
Phone number	Integer	Mobile number field for user	
address	Varchar	Address field for user	
quantity	Integer	Quantity of item	
Order number	Integer	Order number	

Table name: Dry cleaning orders

Field Name	Field Type	Description	Constraints
id	Integer	id	Primary key
name	Varchar	name	
price	Integer	Price	
description	text	Description of item	
image	string	Image of item	
Phone number	Integer	Mobile number field for user	
address	Varchar	Address field for user	
quantity	Integer	Quantity of item	
Order number	Integer	Order number	

Table name: Ironing clothes orders

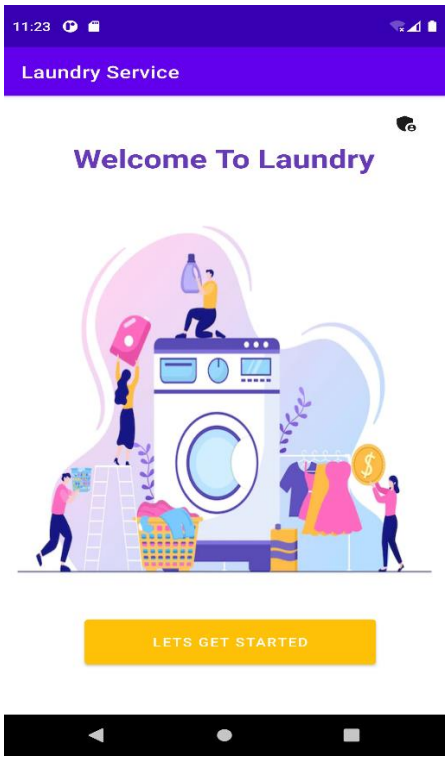
Field Name	Field Type	Description	Constraints
id	Integer	id	Primary key
name	Varchar	name	
price	Integer	Price	
description	text	Description of item	
image	string	Image of item	
Phone number	Integer	Mobile number field for user	
address	Varchar	Address field for user	
quantity	Integer	Quantity of item	
Order number	Integer	Order number	

DATA DICTIONARY

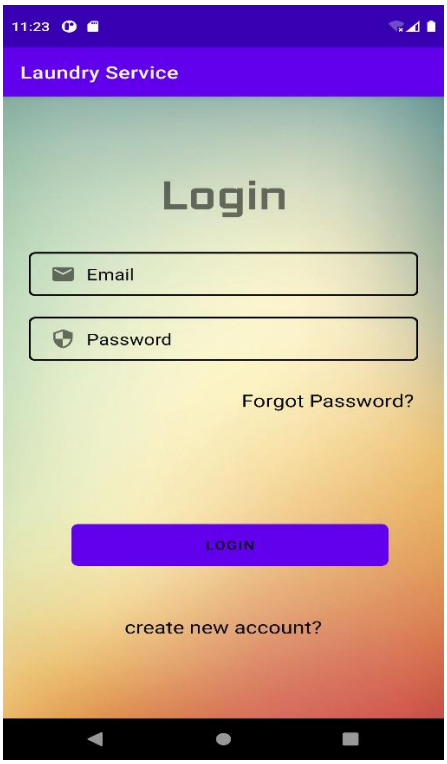
Field Name	Field Type	Description	Constraints
id	Integer	id	Primary key
name	Varchar	name	
price	Integer	Price	
description	text	Description of item	
image	string	Image of item	
Phone number	Integer	Mobile number field for user	
address	Varchar	Address field for user	
quantity	Integer	Quantity of item	
Order number	Integer	Order number	
username	Varchar	Users username	Unique key
email	Varchar	Users mail	Unique key
password	Varchar	Users password	

INPUT SCREENS

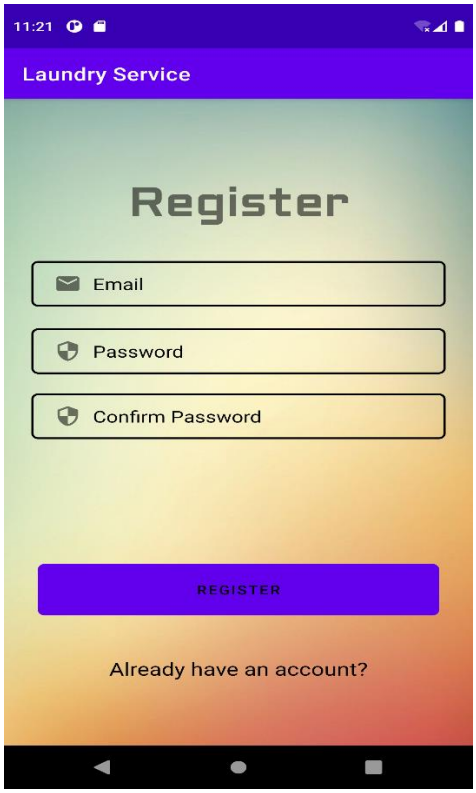
Start Page



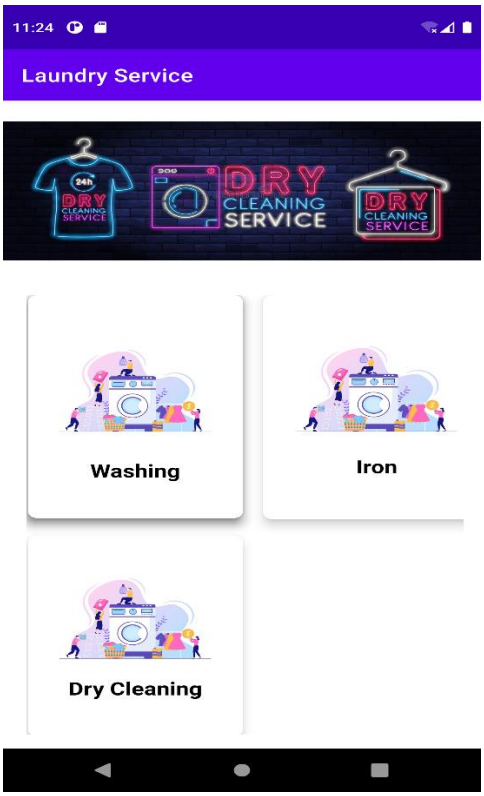
Users Login page



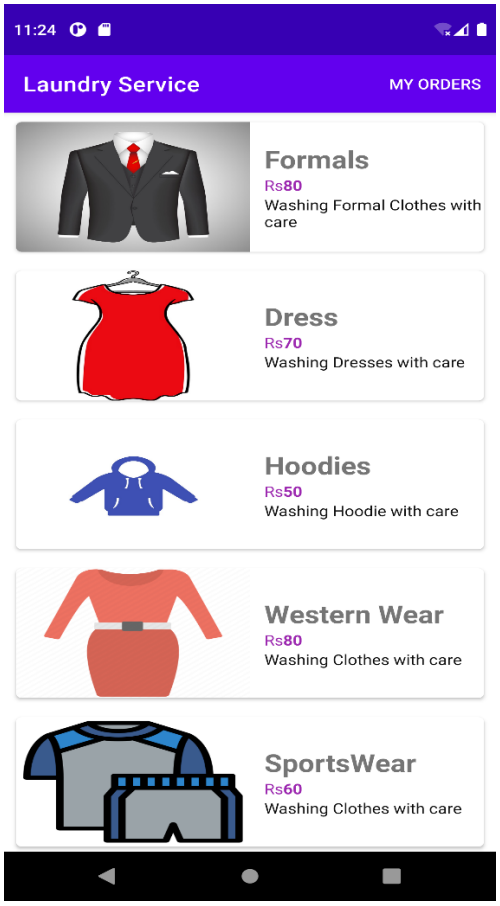
User Registration



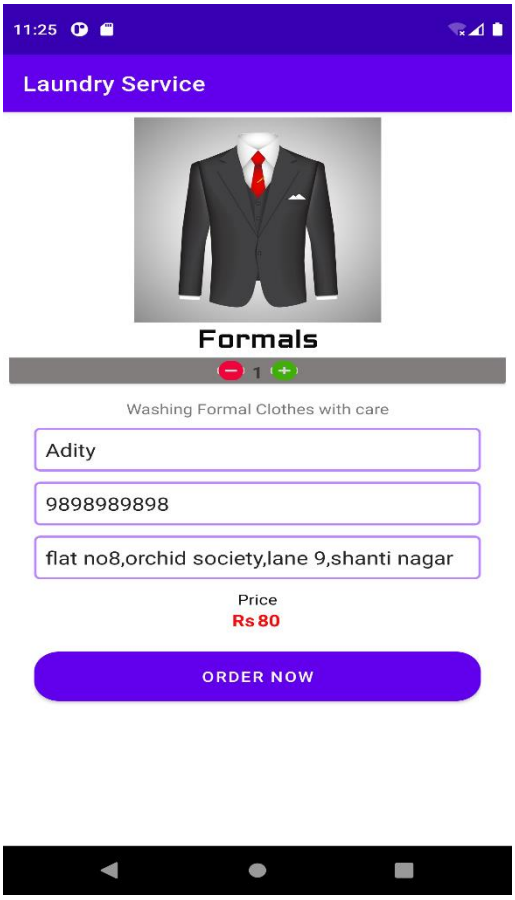
Homepage



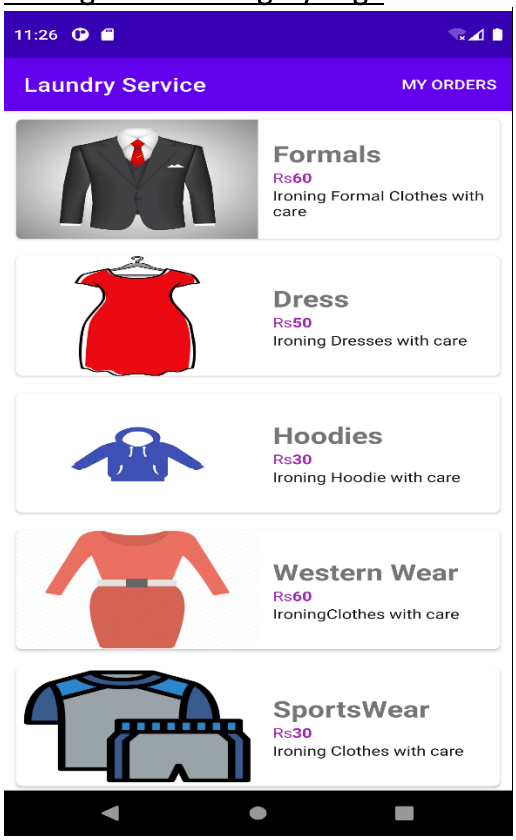
Washing clothes category Page



Details Activity

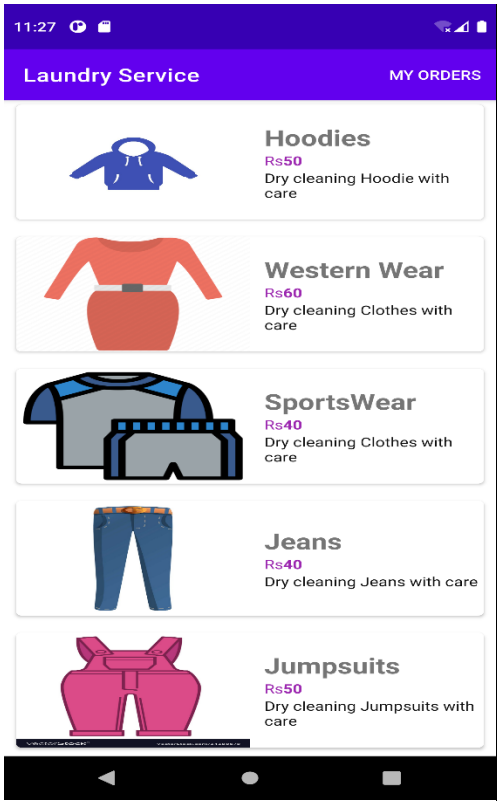


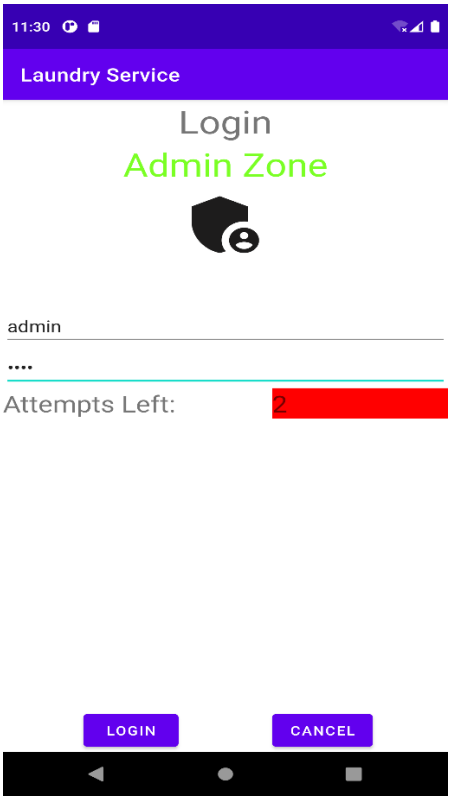
Ironing clothes category Page

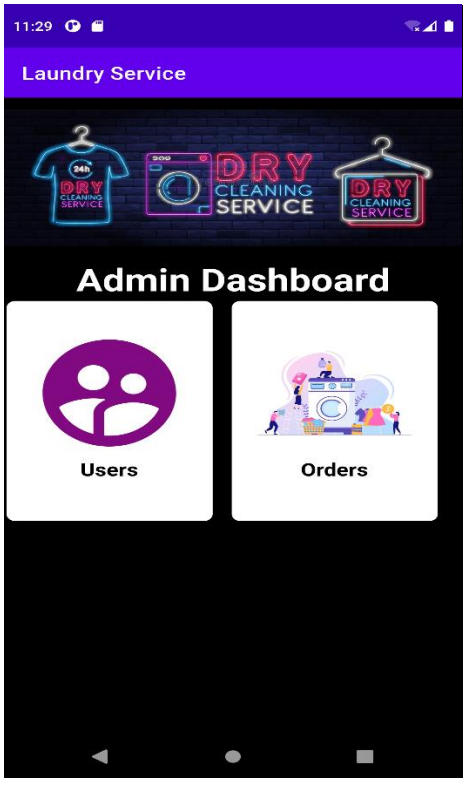


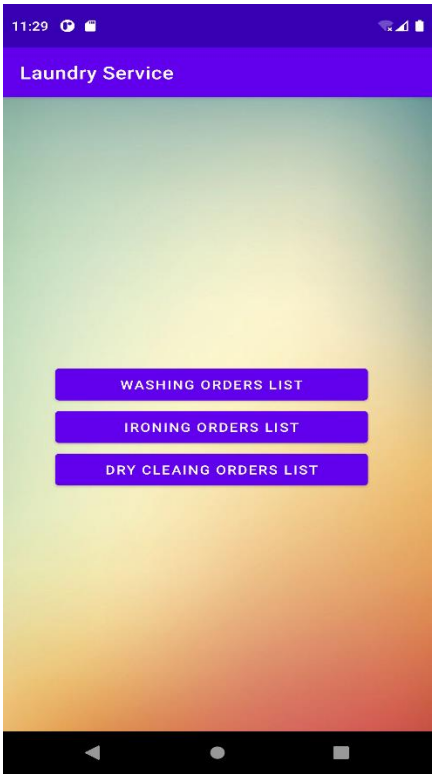
Details For Ironing Activity

DryCleaning clothes category Page



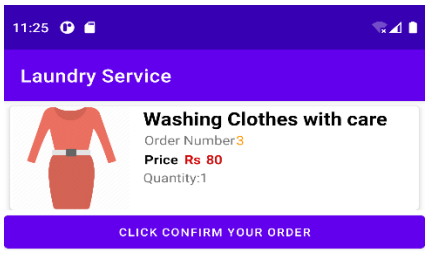




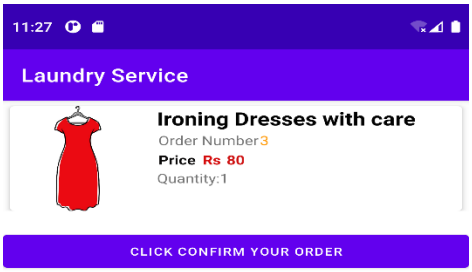


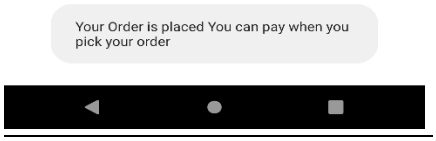
OUTPUT SCREENS

Order Placed for washing category



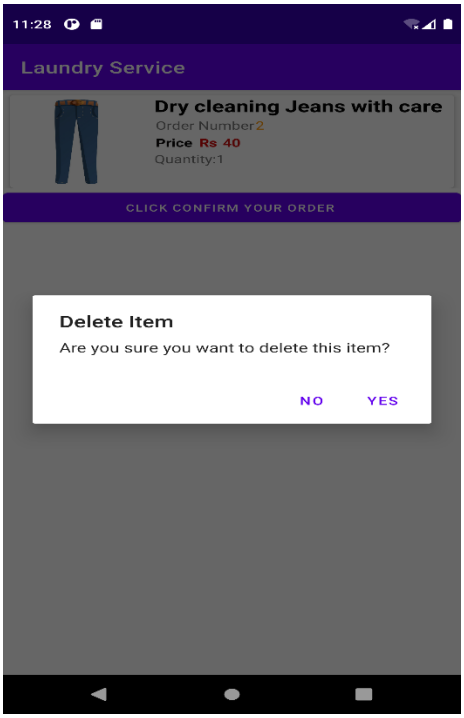
Order Placed for ironing category



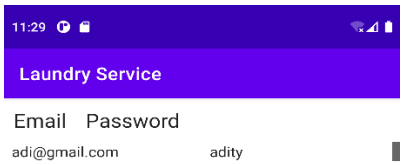


Order Placed for dry cleaning category

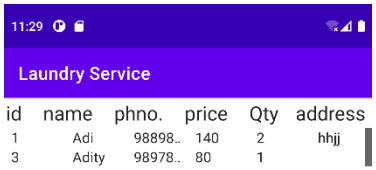
Order Placed for drycleaning category



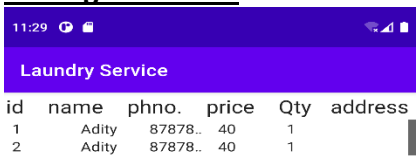
Users list



washing orders list



Ironing orders list



dry cleaning orders list

DATA REPORTS

Registered users

id	Name	Phone No.	Price	Quantity	address	Order No.	Password	Email	ordername	Description
1	Adity	9898989898	60	2	dighi	2	adity	adi@gmail.com	Dress	washing
2	Mayuri	8787876787	120	2	dighi	3	mayuri	123@gmail.com	Formals	ironing
	Aish	9898989898	60	2	dighi	1	adity	aish@gmail.com	Dress	washing
	S Jha	8787876787	120	2	dighi	4	mayuri	jhai@gmail.com	Formals	ironing
	Onkar	9898989898	60	2	dighi	5	adity	onkar@gmail.com	Dress	washing

SAMPLE CODE

Login Page

```
package com.example.application1;


import androidx.appcompat.app.AppCompatActivity;


import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.view.WindowManager;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.EditText;

import android.widget.TextView;
import android.widget.Toast;


public class Login extends AppCompatActivity {
TextView createNewAccount;
EditText _txtUser,_txtPass;
Button _btnLogin;
DBHelper DB;


    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_login);
        _txtUser=(EditText)findViewById(R.id.inputEmail) ;
```

```
_txtPass=(EditText)findViewById(R.id.inputPassword);
_btnLogin=(Button)findViewById(R.id.btn_login);

createNewAccount=findViewById(R.id.create_a_new_account);
DB=new DBHelper(this);




_btnLogin.setOnClickListener(new View.OnClickListener() {
   @Override
   public void onClick(View view) {
      String user=_txtUser.getText().toString();
      String pass=_txtPass.getText().toString();

      if(user.equals("")||pass.equals(""))
         Toast.makeText(Login.this,"Please enter all the
fields",Toast.LENGTH_SHORT).show();
      else{
         Boolean checkuserpass=DB.checkusernamepassword(user,pass);
         if(checkuserpass){
            Toast.makeText(Login.this,"Sign in
Successfully!!",Toast.LENGTH_SHORT).show();
            Intent intent=new Intent(getApplicationContext(),HomeActivity.class);
           // Intent intent=new Intent(getApplicationContext(),Iron.class);
            //Intent intent=new Intent(getApplicationContext(),Dry_cleaning.class);
            //Intent intent=new Intent(getApplicationContext(),Washing_clothes.class);
            startActivity(intent);
         }else{
```

```
                Toast.makeText(Login.this,"Invalid
Credentials!!",Toast.LENGTH_SHORT).show();
            }


        }
//          if(user.equals("admin")||pass.equals("admin"))
//              Toast.makeText(Login.this,"Sign in
Successfully!!",Toast.LENGTH_SHORT).show();
//              Intent intent=new Intent(getApplicationContext(),AdminPanelActivity.class);
//              startActivity(intent);




      }




    });



    createNewAccount.setOnClickListener(new View.OnClickListener() {
      @Override
      public void onClick(View v) {
        startActivity(new Intent(Login.this,Register.class));
      }
    });
  }
}
```

REGISTER

```
package com.example.application1;


import androidx.appcompat.app.AppCompatActivity;


import android.content.Intent;
import android.os.Bundle;


import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;


public class Register extends AppCompatActivity {
TextView alreadyhaveanaccount;
EditText username, password, confirmpassword;
Button signup;
DBHelper DB;
   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_register);
      username=findViewById(R.id.inputEmail1);
      password=findViewById((R.id.inputPassword1));
      confirmpassword=findViewById(R.id.Confirm_password_);
      signup=findViewById(R.id.btn_signup) ;
```

```
        alreadyhaveanaccount=findViewById(R.id.already_have_account);
        DB=new DBHelper(this);

        signup.setOnClickListener(view -> {
            String user=username.getText().toString();
            String pass=password.getText().toString();
            String repass=confirmpassword.getText().toString();

            if(user.equals("")||pass.equals("")||repass.equals(""))

                Toast.makeText(Register.this,"Please enter all the
fields",Toast.LENGTH_SHORT).show();
            else {
                if(pass.equals(repass)){
                    Boolean checkuser=DB.checkusername(user);
                    if(!checkuser){
                        Boolean insert=DB.insertData(user,pass);
                        if(insert){
                            Toast.makeText(Register.this,"Register
Successfully!!",Toast.LENGTH_SHORT).show();
                            Intent intent=new Intent(getApplicationContext(),Login.class);
                            startActivity(intent);
                        }
                        else {
                            Toast.makeText(Register.this,"Register
Failed!!",Toast.LENGTH_SHORT).show();
                        }
```

```
                }
                else{
                    Toast.makeText(Register.this,"User Already Exist. Please Sign
in",Toast.LENGTH_SHORT).show();
                }
            }
            else {
                Toast.makeText(Register.this,"Passwords Not
Matching",Toast.LENGTH_SHORT).show();
            }
        }
    });

    alreadyhaveanaccount.setOnClickListener(v -> startActivity(new
Intent(Register.this,Login.class)));


  }
}
```

Admin Login

```
package com.example.application1;

import androidx.appcompat.app.AppCompatActivity;
import android.view.View;
import android.graphics.Color;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;
import android.content.Intent;

import android.os.Bundle;



public class AdminPanelActivity extends AppCompatActivity {
   Button b1,b2;
   EditText ed1,ed2;

   TextView tx1;
   int counter = 3;
   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_admin_panel);
```

```
        b1 = (Button)findViewById(R.id.button);
        ed1 = (EditText)findViewById(R.id.editText);
        ed2 = (EditText)findViewById(R.id.editText2);

        b2 = (Button)findViewById(R.id.button2);
        tx1 = (TextView)findViewById(R.id.textView3);
        tx1.setVisibility(View.GONE);

        b1.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                if(ed1.getText().toString().equals("admin") &&
                        ed2.getText().toString().equals("admin")) {
                    Intent intent=new
Intent(AdminPanelActivity.this,AdminDashboard.class);
                    startActivity(intent);
                    Toast.makeText(getApplicationContext(),
                            "Redirecting...",Toast.LENGTH_SHORT).show();
                }else{
                    Toast.makeText(getApplicationContext(), "Wrong
Credentials",Toast.LENGTH_SHORT).show();

                        tx1.setVisibility(View.VISIBLE);
                    tx1.setBackgroundColor(Color.RED);
                    counter--;
                    tx1.setText(Integer.toString(counter));
```

```
                if (counter == 0) {
                    b1.setEnabled(false);
                }
            }
        }
    });

    b2.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            finish();
        }
    });
  }
}
```

LIMITATIONS AND BIBLIOGRAPHY

The system at present does not take care of the money payment methods, as the consolidated constructs need SSL standards and are critically to be initiated in the first face, this application right now has cash payments the application of the credit card transactions and UPI payment is applied as a developmental phase in the coming days. The system needs more elaborative technicality for its inception and evolution

References Used:

- Youtube videos
- www.w3schools.com and www.googleapis.com
- stackoverflow.com