E-COMMERCE FOR MOBILE

A PROJECT REPORT

Submitted By

AKASH (21BIT008)

in partial fulfilment

Of

U18ITI5304L

SOFTWARE ENGINEERING LABORATORY

IN

INFORMATION TECHNOLOGY

KUMARAGURU COLLEGE OF TECHNOLOGY

NOVEMBER2023

BONAFIDE CERTIFICATE

Certified that this project report "**E-COMMERCE FOR MOBILE**" is the Bonafide work of **AKASH A (21BIT003)** who carried out the project work under my supervision.

SIGNATURE

Dr. Divya Vadlamudi

Head of Department

Information Technology

Kumaraguru College

Of Technology

SIGNATURE

SUPERVISOR

Mrs. Sathyavathi S

Assistant Professor(SRG)

Information Technology

Kumaraguru College

Of Technology

TABLE OF CONTENTS

1. INTRODUCTION

1.1 DOCUMENT SCOPE AND PURPOSE:

The document outlines the scope and purpose of the mobile e-commerce project, defining its key objectives, functionalities, and target audience. It serves as a comprehensive guide for project stakeholders, detailing the development and implementation of a user-friendly and secure mobile platform to facilitate seamless online shopping experiences.

1.2 TARGET AUDIENCE

The target audience for E-Commerce is the customers and vendors.

1.3 PRODUCT OVERVIEW (INCLUDING CAPABILITIES, SCENARIOS FOR USING THE PRODUCT, ETC.)

The capabilities of this product includes all the vendor activities like login, signup, Adding product, Updating product, and customer activities includes signup, login, searching product and add to Wishlist, add to cart, and buy product activities.

1.4 ACRONYM / ABBREVIATIONS

GUI -Graphical User Interface

SRS -Software Requirement System

SDLC –System Development Lifecycle

POSTGRE SQL- Which is used for adding, removing, updating and modifying in database.

2. PROJECT MANAGEMENT PLAN

2.1 LIFE CYCLE MODEL USED

Agile SDLC model is a **combination of iterative and incremental process models** with focus on process adaptability and customer satisfaction by rapid delivery of working software product. **Agile Methods break the product into small incremental builds.** These builds are provided in iterations. It promotes continuous iteration of development and testing throughout the software development lifecycle of the project. It describes software development approaches that employ continual planning, learning, improvement, team collaboration, evolutionary development, and early delivery. It encourages flexible responses to change.

Scrum is an **agile development methodology** used in the development of Software based on an **iterative** and **incremental processes**. Scrum is adaptable, fast, flexible and effective agile framework that is designed to deliver value to the customer throughout the development of the project. The **primary objective** of Scrum is to **satisfy the customer's need** through an environment of transparency in communication, collective responsibility and continuous progress. The development starts from a general idea of what needs to be built, elaborating a list of characteristics ordered by priority (product backlog) that the owner of the product wants to obtain.

We started working on the E-Commerce project on 19.07.2023, This first meet is called "Sprint Planning", where the whole team, including the Product Owner, meet

on the first day of the Sprint and conduct a Sprint Planning session. This is the very first thing to happen as soon as the Sprint commences. Sprint Planning ought to be prepared for.

The most important preparation is to ensure that the Product Backlog has been refined to an appropriate level of detail, with estimates and acceptance criteria (this is the purpose of Product Backlog Refinement). Also, the team should have an idea of their capacity for this Sprint, which is to say, how much work they believe they can take on.

Following this a Sprint Backlog is planned, is more than just a selection of work with an end goal in mind. It is also a plan for how that goal will be achieved, and how the associated work will be performed.

Once the team have planned their Sprint Backlog they can start work. If they have planned things out as tasks, they will collaborate with each other, as a team, to make sure that those tasks are completed.

Every working day, at the same time, the Development Team will meet and plan what they will do to bring them closer to the Sprint Goal. This meeting is called the Daily Scrum and it should never take more than 15 minutes.

If a team has collaborated efficiently, they'll have worked together to meet the Sprint Goal, managing any risks and adjusting their plans as necessary.

They'll have demonstrated control throughout the Sprint through an even burndown of work remaining, where each member saw it as their personal responsibility to help complete work in progress. They'll have a valuable increment to demonstrate to the Product Owner and any invited stakeholders. A Review is something a team should look forward to.

A Sprint Review is also an inspect-and-adapt opportunity. It's a good time for the Product Owner to explain how well the product is performing, to get feedback first-hand from any invited parties, and to draw any lessons which might be used to improve the Product Backlog further.

If any work has not been completed, for whatever reason, then this will also be reviewed and re-estimated on the Product Backlog for possible planning into future sprints.

We had a series of review meets conducted on 25.07.2023, 01.08.2023, 08.09.2023, 15.10.2023 and 11.11.2023 where we discussed with the whole scrum team and decided on changes, or the enhancements required in the project.

In the first week we developed a prototype to better understand the user requirements and came up with our Project Requirements.

In the second week, we began to build the fronted for the user side using REACT, in the following week we also completed building the fronted for the admin module using the same tools. In both the weeks we had a review meet to confirm the layout and looked out for any changes. Following this, in the fourth and the fifth week we started working on the backend for both the admin and user module simultaneously. In the sixth week we showed our final project, where few changes and features to be integrated were mentioned.

We had our final review meet on 17.10.2023, where we finalized our project and began our testing works and on 18.10.2023, we completed our project.

2.2 RISK ANALYSIS

This system entails identifying, evaluating, prioritizing, and controlling risks in the blood bank system. This system faces several types of risk:

- Operational risk to day-to-day operations
- Project risk, encountered when developing the software
- Security risk, including cyber security.

2.3 HARDWARE AND SOFTWARE RESOURCES REQUIREMENT

HARDWARE REQUIREMENTS:

Processor: intel core

Installed memory: 8 GB or higher

Speed: 1.40 GHz or faster

Operating system: 32/64 Bit operating system, x86/x64-based processor

SOFTWARE REQUIREMENTS:

Operating system: Windows 10

Data base: Postgresql

Web server: Django

Web technologies: HTML,CSS, REACT

IDE and tools: PGADMIN

Editor: VS Code

3. LITERATURE SURVEY

3.1 SURVEY:

Following are the main problem in E-Commerce

- Data is not always reliable and some human errors might have occurred, for example details mismatch.
- Stock updation is not much efficient when done manually.

These problems can be overcome by using this system.

4. REQUIREMENTS SPECIFICATION

4.1 STAKEHOLDERS FOR THE SYSTEM

- Admin of the system
- vendors of the system.
- Customers

4.2 FUNCTIONAL & NON-FUNCTIONAL REQUIREMENTS

Functional requirements for vendor module:

1) Login and Register
2) Display product
3) Add product
4) Update quantity
5) Change password
6) Change status of order

Functional requirements for User module:

1) Login and Register
2) Select categories
3) View product
4) Select product
5) Add to wishlist
6) Add to cart
7) Remove from wishlist
8) Remove from cart
9) Checkout from cart
10) Check status of order

Non-functional requirements:

- The system should be easily accessible, efficient and user friendly.
- Adding updating the items should be kept in the track.

4.3 USER NEEDS

We all know that our expectations and needs are growing steadily. In the same way, the users of this product will be having certain needs and expectations that are to be satisfied.

4.4 SYSTEM FEATURES

This system includes features such as,

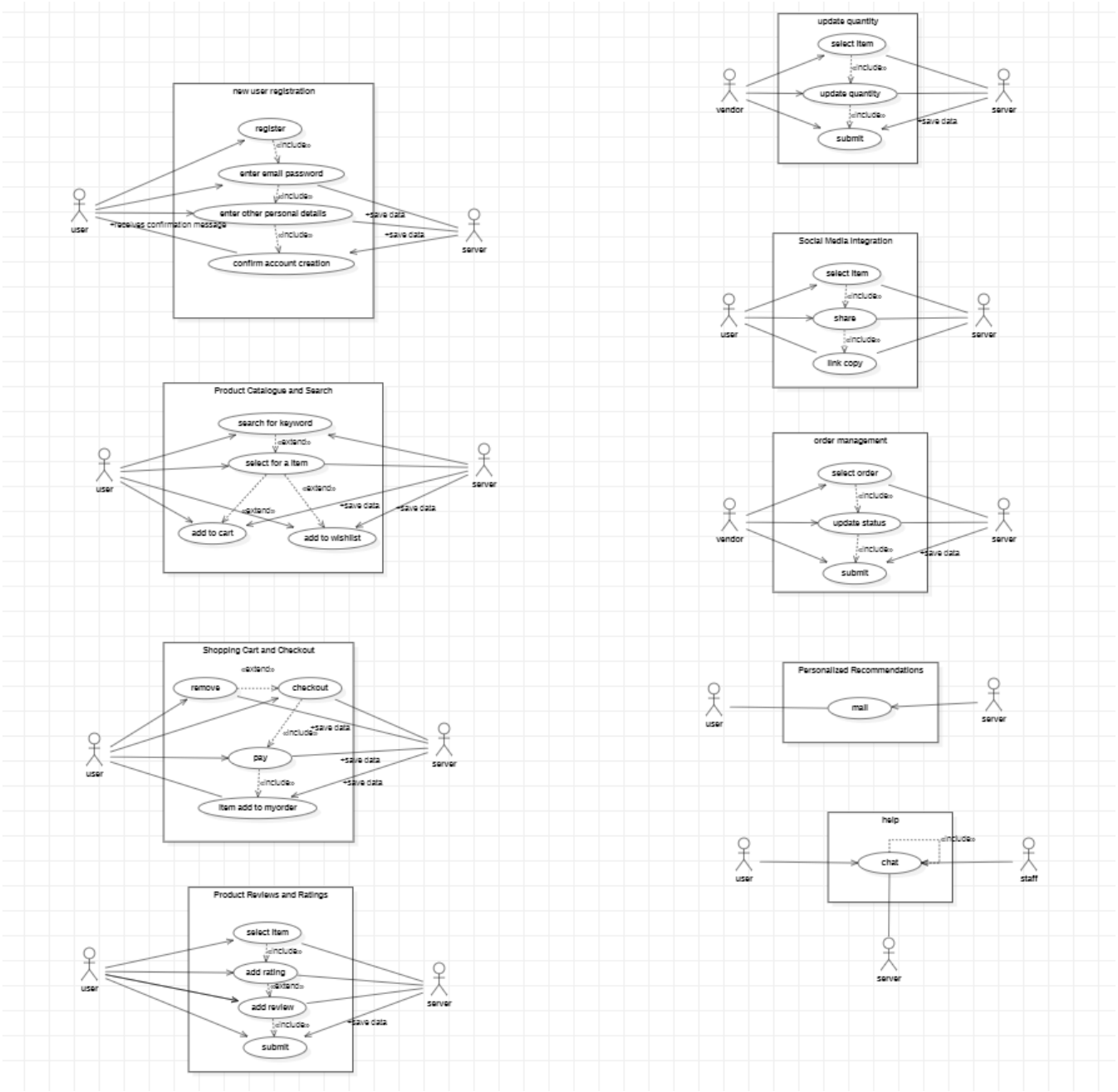
- Efficient and separate interface for both Vendor and the customer
- Registration for new customer
- Separate action access for both vendor and the customer.

4.5 UML DIAGRAMS

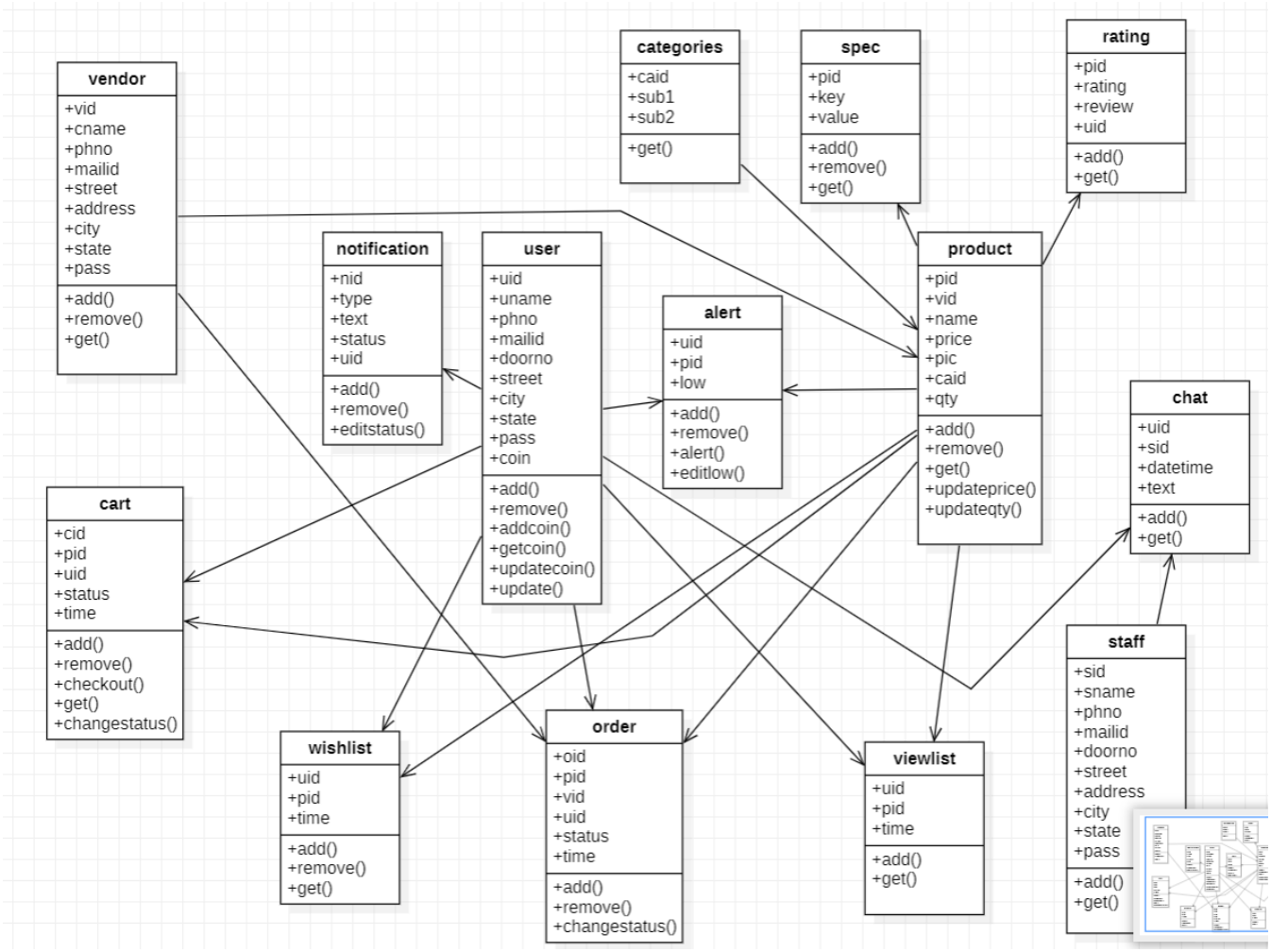
UML is an acronym that stands for unified modelling language. UML is a modern approach to modelling and documenting a software. In fact, it is one of the most popular business process modelling techniques.

It is based on diagramatic representation of software components. By using visual representations, we are able to better understand possible flaws or errors in software or business processes.

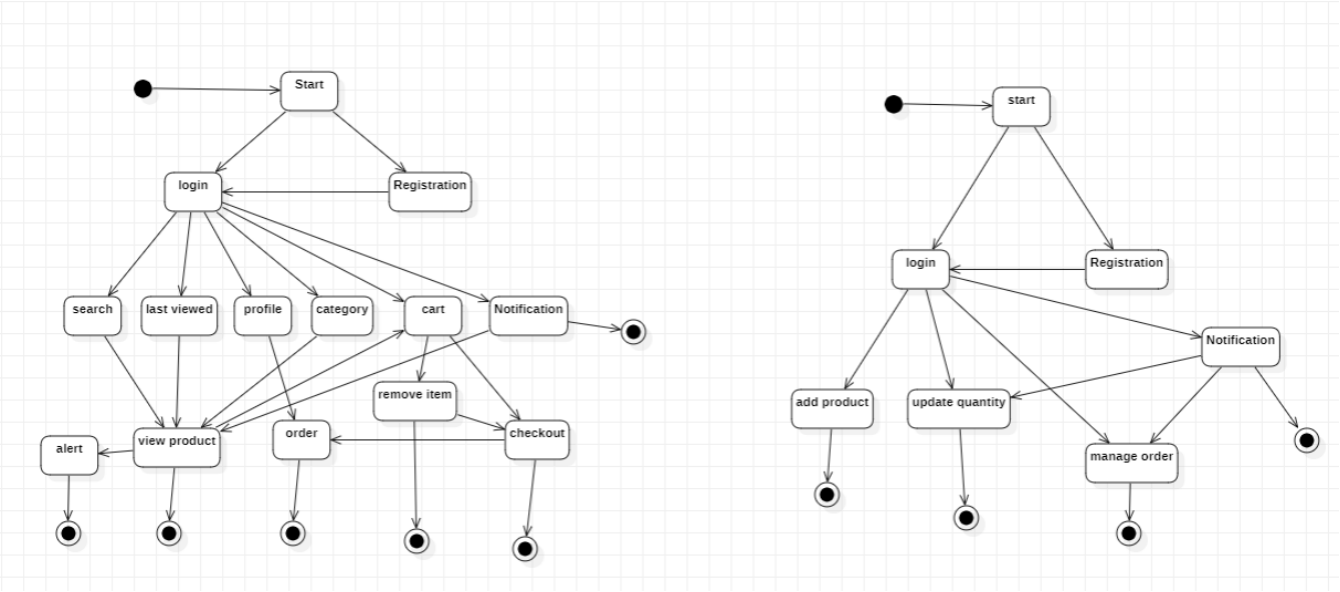
4.5.1 USE CASE DIAGRAM



4.5.2 CLASS DIAGRAM


vendor
+vid
+cname
+phno
+mailid
+street
+address
+city
+state
+pass
+add()
+remove()
+get()
categories
+caid
+sub1
+sub2
+get()
spec
+pid
+key
+value
+add()
+remove()
+get()
rating
+pid
+rating
+review
+uid
+add()
+get()
notification
+nid
+type
+text
+status
+uid
+add()
+remove()
+editstatus()
user
+uid
+uname
+phno
+mailid
+doorno
+street
+city
+state
+pass
+coin
+add()
+remove()
+addcoin()
+getcoin()
+updatecoin()
+update()
alert
+uid
+pid
+low
+add()
+remove()
+alert()
+editlow()
product
+pid
+vid
+name
+price
+pic
+caid
+qty
+add()
+remove()
+get()
+updateprice()
+updateqty()
chat
+uid
+sid
+datetime
+text
+add()
+get()
cart
+cid
+pid
+uid
+status
+time
+add()
+remove()
+checkout()
+get()
+changestatus()
staff
+sid
+sname
+phno
+mailid
+doorno
+street
+address
+city
+state
+pass
+add()
+get()
wishlist
+uid
+pid
+time
+add()
+remove()
+get()
order
+oid
+pid
+vid
+uid
+status
+time
+add()
+remove()
+changestatus()
viewlist
+uid
+pid
+time
+add()
+get()

4.5.3 ACTIVITY DIAGRAM



5 MODULES

5.1MODULE 1

vendor module includes,

1) Login and Register
2) Display product
3) Add product
4) Update quantity
5) Change password
6) Change status of order

- **MODULE 2**

User module includes,

1) Login and Register
2) Select categories
3) View product
4) Select product
5) Add to wishlist
6) Add to cart
7) Remove from wishlist
8) Remove from cart
9) Checkout from cart
10) Check status of order

5.1.2 SYSTEM DESIGN & IMPLEMENTATION

- **DESIGN APPROACH**

At this stage an overall design of the system architecture and physical design which includes User Interface and Database design. It is at this stage that any faults can be identified before moving onto the next stage. The output of this stage is the design specification which is used in the next stage of implementation. The system was designed to allow the administrator that is tenant and admin to view, edit, delete and add data to the database.

- **ARCHITECTURE DESIGN**

Designing the application architecture involves network technologies and making decisions on how the system data, processes and interfaces are to be distributed.To do this,the data and process models that were created during the requirement analysis were analyzed. The E-commerce is based on the two tier architecture which is made up of two logical tiers i.e.,the presentation tier and middle tier

- **The Presentation tier:** When designing the presentation tier of this The E-commerce, REACT together with Django which is a programming language was used
- **Middle tier:** The middle tier was designed to capture information like,the login details ,adding items, updating items, view , search product.

- **UI DESIGN**

For the flexibility of the user, the interface has been developed in graphical user interface mode. The normal interface is applied through browser.

The GUI's at the top level has been categorized as:

1. Vendor user interface
2. Customer user interface

The Vendor user interface concentrates on the consistent information that is practically, pact of the organizational activities and which needs proper authentication for the data collection. The interfaces help to Add, update the product.

The Customer user interface concentrates on the consistent information that is practically, pact of the organizational activities and which needs proper authentication for the data collection. The interfaces help to view the product, search the product.

- **DESIGN PATTERNS**

In software engineering, a design pattern is a general repeatable solution to a commonly occurring problem in software design. A design pattern isn’t a finished design that can be transformed directly into code. It is a description or template for how to solve a problem that can be used in many different situations.

Design patterns represent the best practices used by experienced object-oriented software developers.

TRACEABILITY FROM REQUIREMENTS TO DETAILED DESIGN MODEL

When we choose a backend for an enterprise level application we have so many options like Oracle, Sybase, MySQL, and POSTGRESQL however we choose POSTGRESQL as our database, and it has so many features which is ideal for our application, includes

- Ease of use
- Secure
- Client/Server Architecture
- Scalability
- Flexibility
- Supported as various OS
- Productivity

6.6 CHOICE OF PROGRAMMING LANGUAGE & CODING STANDARDS

At this stage, I will begin coding as per design specifications. The output of this step is one or more product components built according to a predefined coding standard and debugged, tested and integrated to satisfy the system architecture requirement.

Frontend: R E A C T , HTML and CSS were used to develop the UI since they are cost-effective multiplatform development platform, have consistency across multiple browsers and supports offline browsing.

Backend: POSTGRESQL, this is because the records of details such as login details, add items and view items details and other such information can be stored and retrieved efficiently using database.

6.7 WEB SITE DIRECTORIE:

User site directories:

Root Directory-All web pages related to vendor and customer functions.

5.1.3 REFERENCES

- **REFERENCE LINKS**

https://www.w3schools.com/html/

https://www.w3schools.com/css/

https://www.w3schools.com/mysql/

5.1.4 TECHNOLOGY USED

- **FRONT END TECHNOLOGY**

We have used REACT,HTML, CSS.

The **Hypertext Markup Language**, or **HTML** is the standard markup language for documents designed to be displayed in a web browser. It can be assisted by technologies such as Cascading Style Sheets (CSS) and scripting languages such as JavaScript.

Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included clues for the appearance of the document.

- **BACK END TECHNOLOGY**

Software used POSTGRESQL.

PostgreSQL is an advanced, enterprise class open source relational database that supports both SQL (relational) and JSON (non-relational) querying. It is a highly stable database management system, backed by more than 20 years of community development which has contributed to its high levels of resilience, integrity, and correctness. PostgreSQL is used as the primary data store or data warehouse for many web, mobile, geospatial, and analytics applications.

Database Name: ecom

Tables:

(Table names – Attributes)

1)appuser→ uid, uname, phno, mailid, doorno, street, city, state, pas, coin

2) vendor →vid, vname, phno, mailid, street, City, state, pas

3)categories →caid, sub1, sub2

4) product →pid, vid, caid, name, price, qty

5) spec → pid, key, value

6) rating→pid, uid, rating, review

7) notification→nid, uid, vid, typ, text, state

8)cart→cid, pid, uid, state , time

9) wishlist→ wid, pid, uid, time

10)order→ oid, pid, uid, state, time

11)viewlist→vid, pid, uid, time

5.1.5 TESTING

- **TESTING GOALS**

- Phase wise testing of the system was conducted to test the software against any kind of malfunctioning.

- Logical testing was conducted to test every aspect of each form, report and query as soon as it was implemented, using valid, invalid and extreme data. Test data was added to each code module and results compared with the expected results. Each menu item was tested in turn to ensure that no function had been missed out.

- The whole ranges of tests were conducted to ensure that no errors had been introduced in the system. In system testing, performance and acceptance standards were developed and sub-standard performance or service interruptions that result in system failure were checked. System testing was conducted in a phase wise manner; the involved phases were as follows:-

1. Program Testing (To test logical and syntactical errors).

2. System Testing (To test system integrity).

3. String Testing (To test the interactivity of the programs).

- **TESTING TOOLS USED**

Test Complete is a functional testing platform that offers various solutions to automate testing for desktop, web, and mobile applications by Smart Bear Software.
Test Complete offers the following features:

- GUI testing.
- Scripting Language Support – JavaScript, Python.
- Test visualizer.
- Scripted testing.
- Test recording and playback.

- **TESTING METHODOLOGY**

Program level testing, modules level testing integrated and carried out. There are two major type of testing they are

1. White Box Testing.
2. Black Box Testing.

White box testing:

White box sometimes called "Glass box testing" is a testcase design that uses the control structure of the procedural design to drive test case. The following tests were made on the system,

a) All independent paths within a module have been exercised once. In our system, ensuring that case was selected and executed checked all case structures. The bugs that were prevailing in some part of the code where fixed.

b) All logical decisions were checked for the truth and falsity of the values.

Black box testing:

Black box testing focuses on the functional requirements of the software. This is black box testing enables the software engineering to derive a set of input conditions that will fully exercise all functional requirements for a program.

1) Interface errors
2) Performance in data structure
3) Performance errors
4) Initializing and termination errors.

• TEST CASES

- It is the process of finding differences between the expected behaviour specified by system models and the observed behaviour implemented system.

- The goal is to design tests that executes defects in the system and to reveal problems.

- The program to be tested is executed with a set of test cases, and the output of the program for the test cases is evaluated to determine if the program is performing as expected.

- The success of testing in revealing errors in program depends critically on test cases.

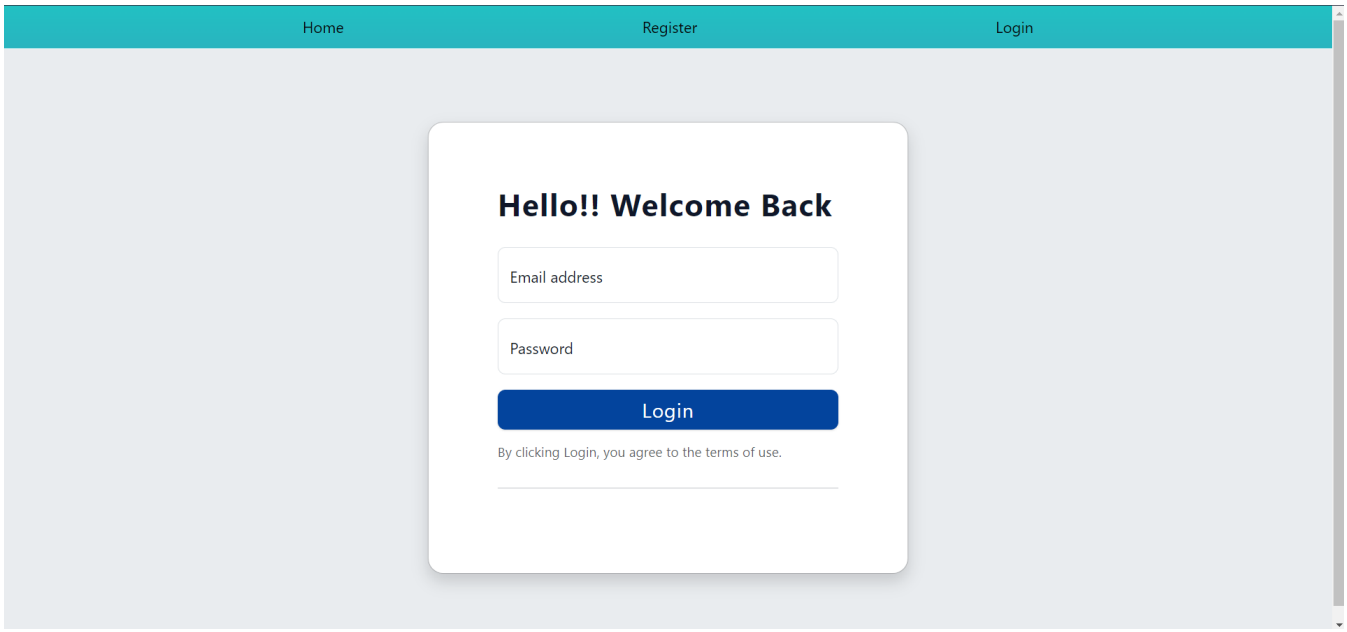
- **BUG LIST**

- It will involve both unit testing of individual code module, system testing of the integrated product and acceptance testing will be conducted. Will ensure bugs are corrected before moving on the next stage of modules.

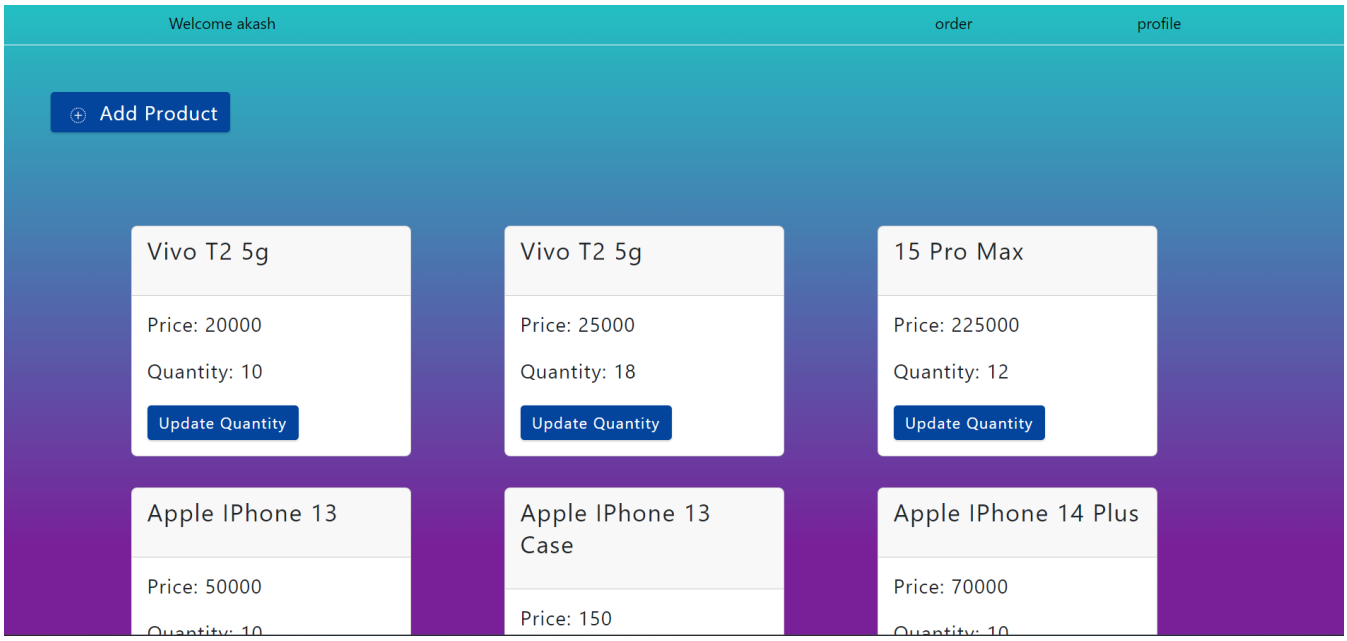
- It involves modifications on the system to improve performance. These modifications are recorded for documentation and system update.

5.1.6 APPENDIX

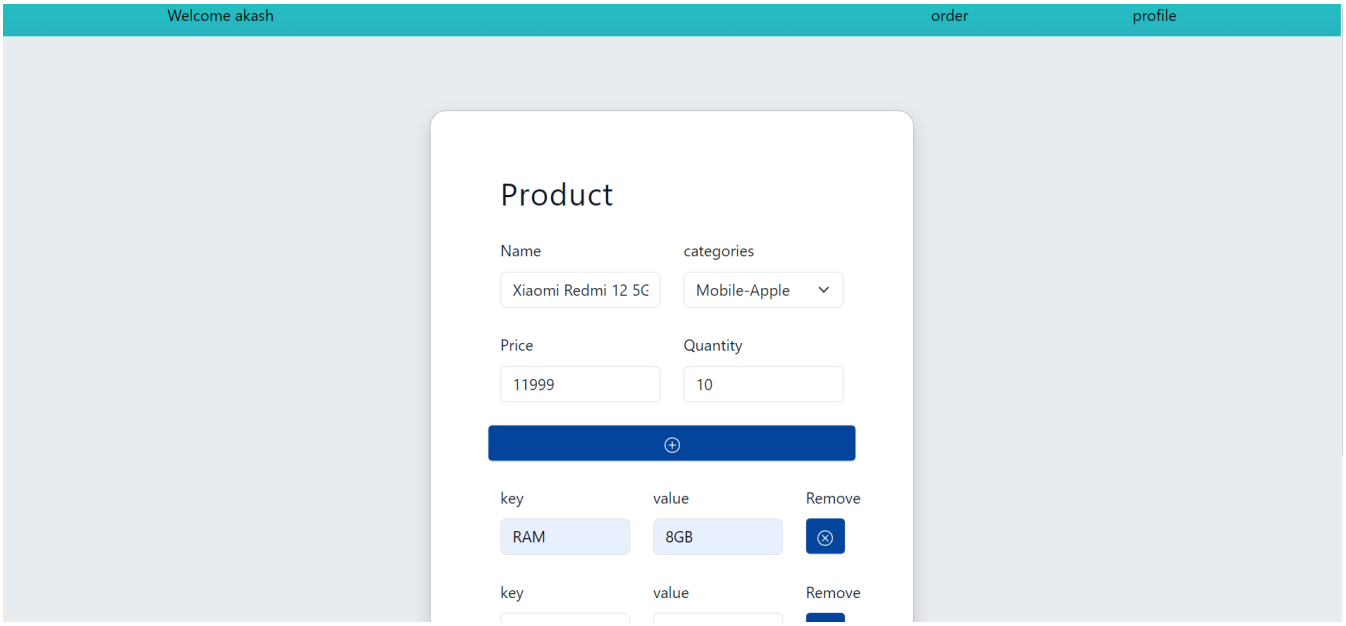
1) Vendor login:-



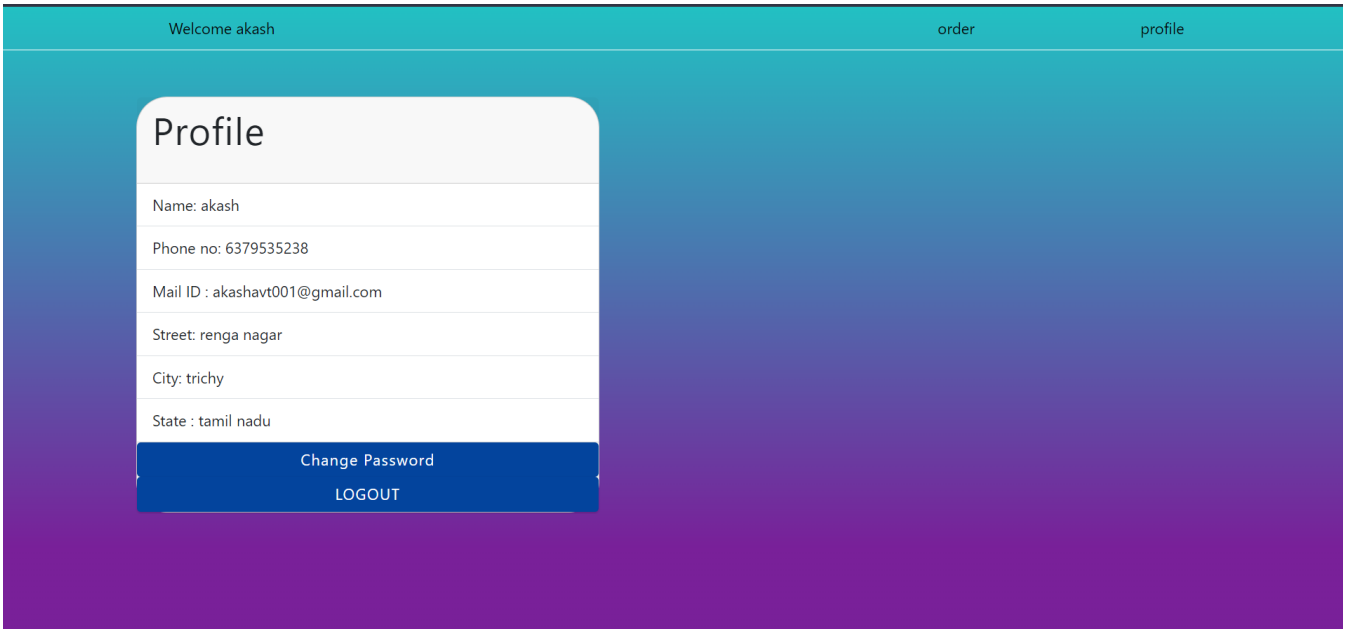
2) Vendor homepage:



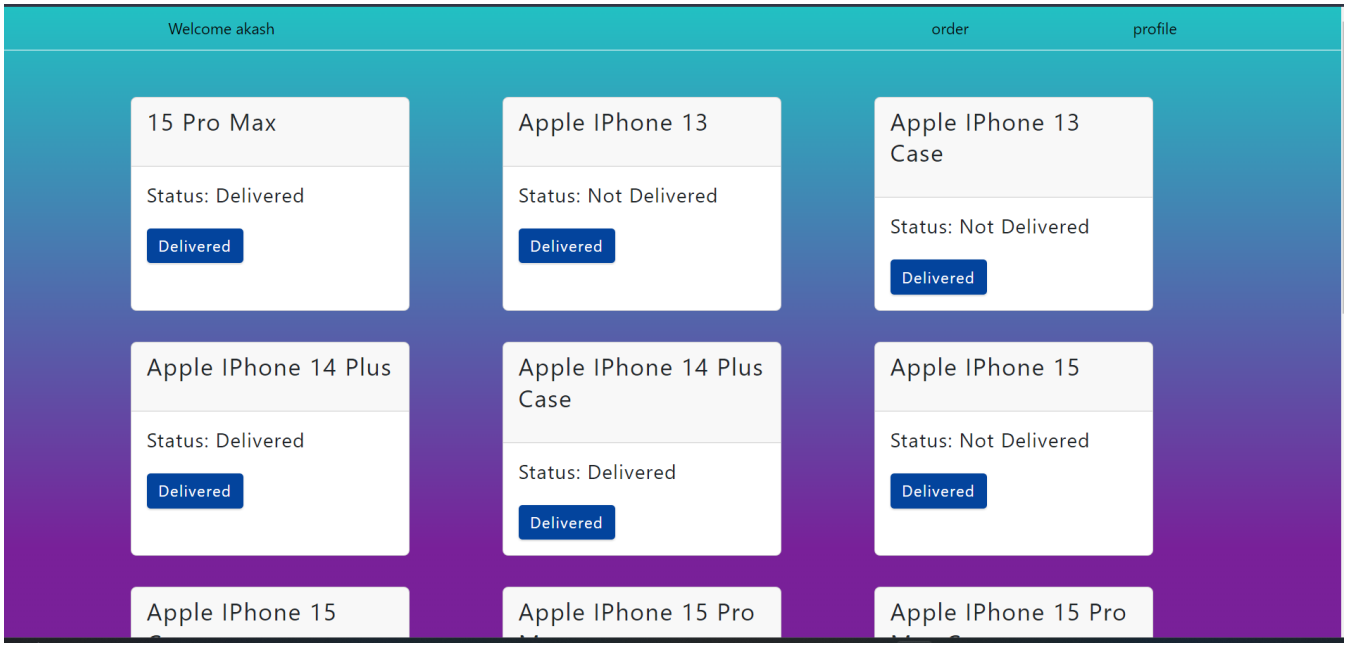
3) Add product:



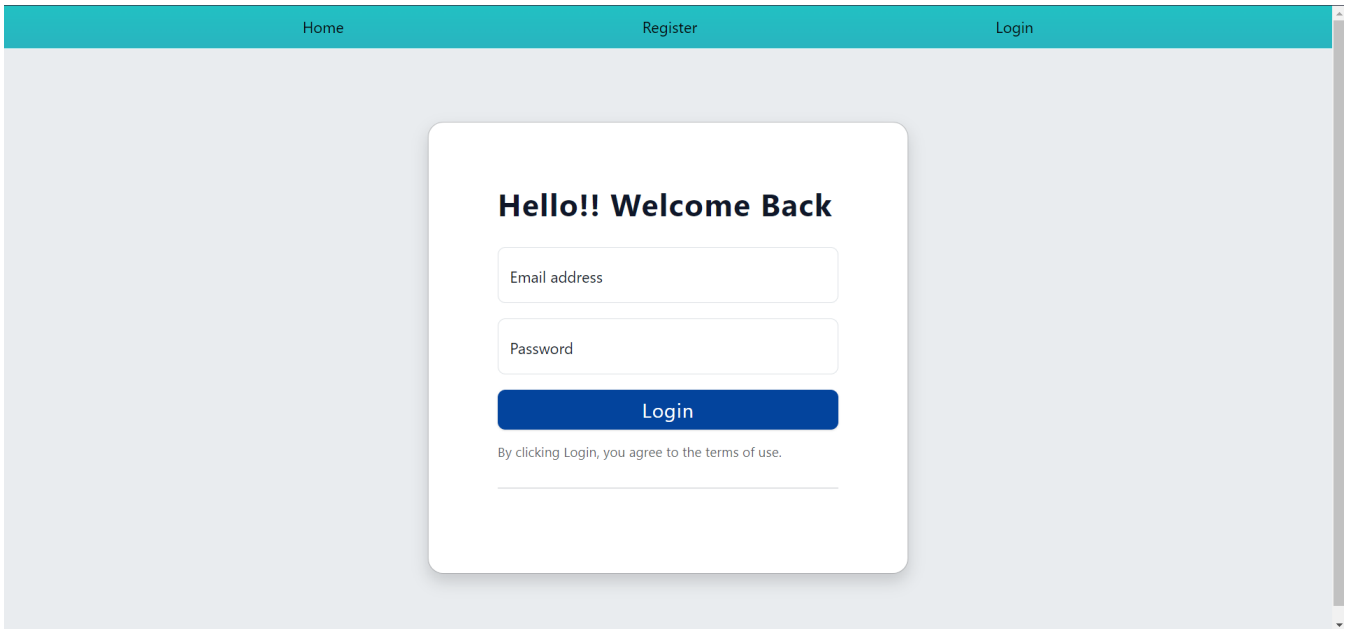
4) Profile:



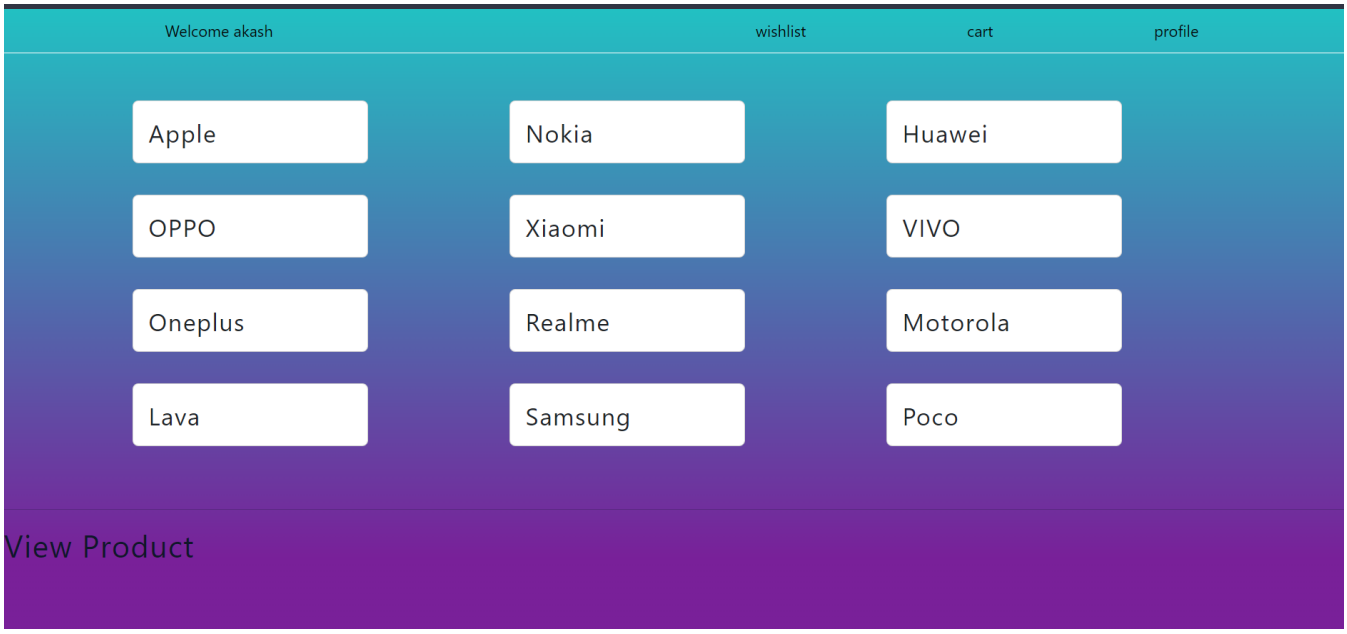
5) Order:



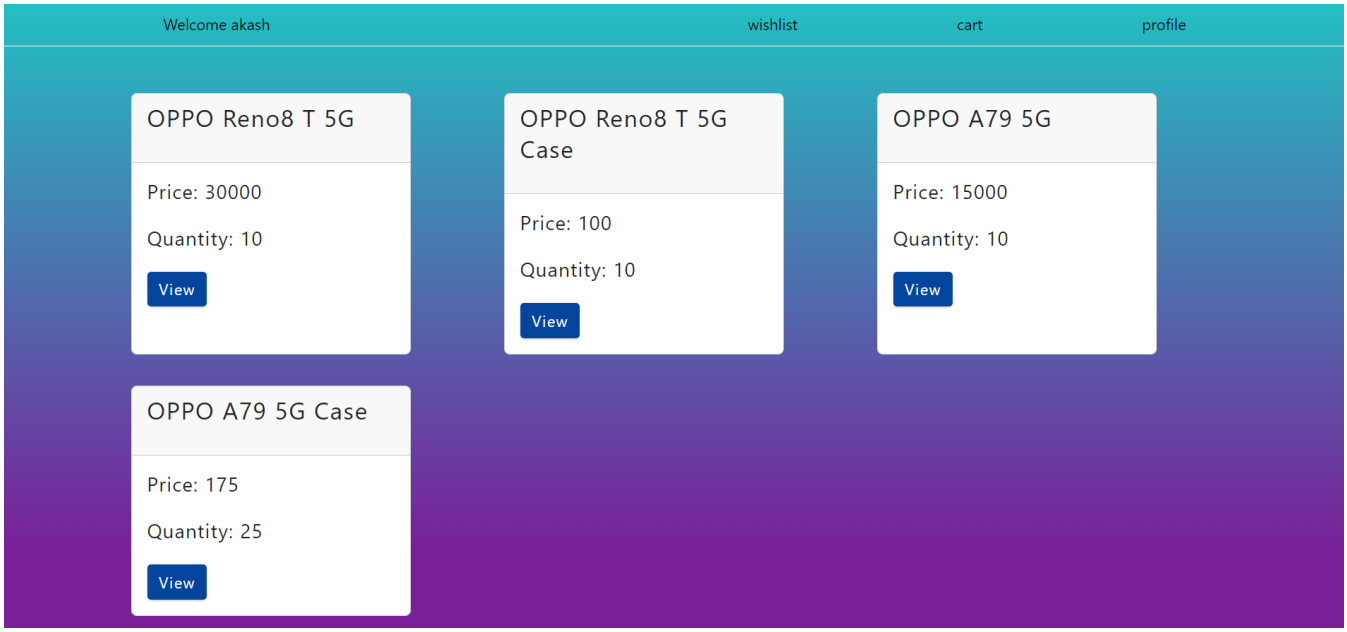
6) User login:



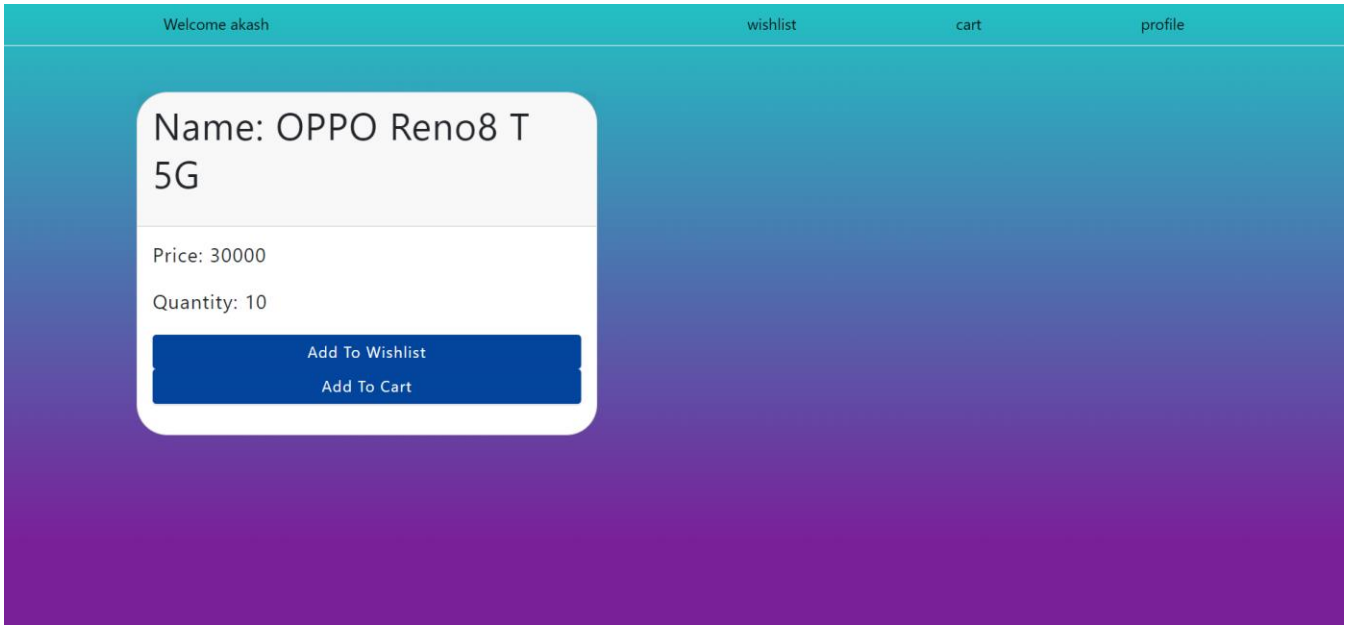
7) User home:



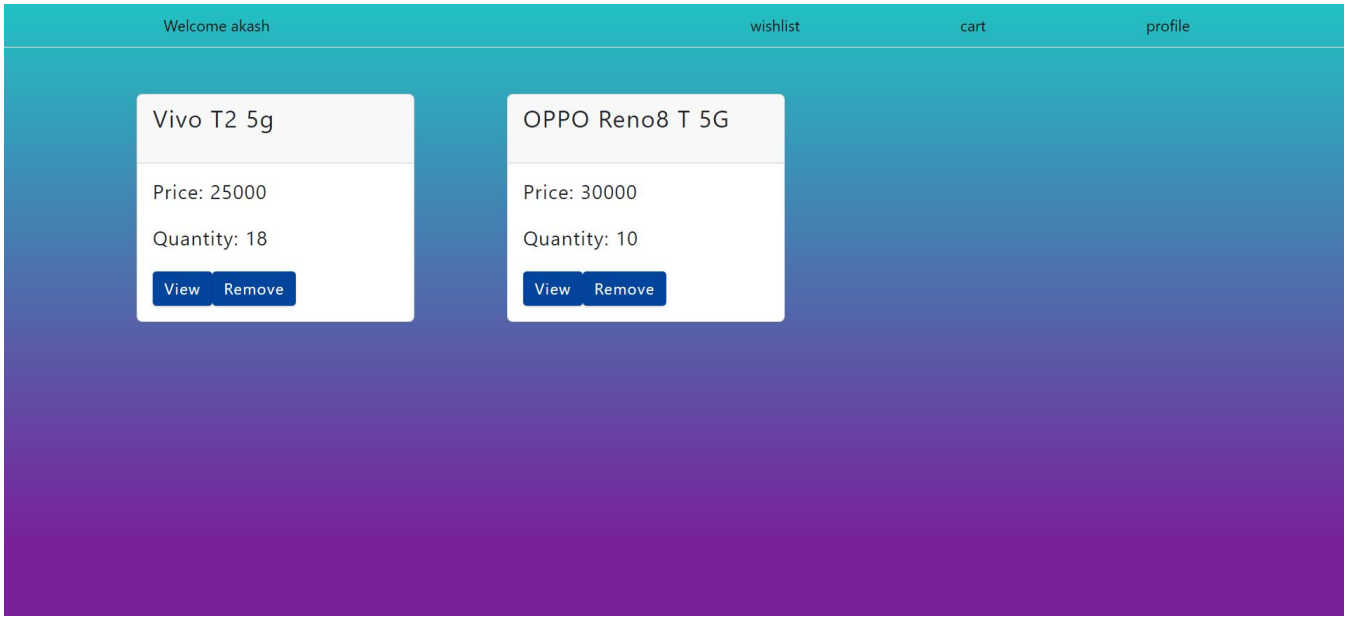
8) List product:



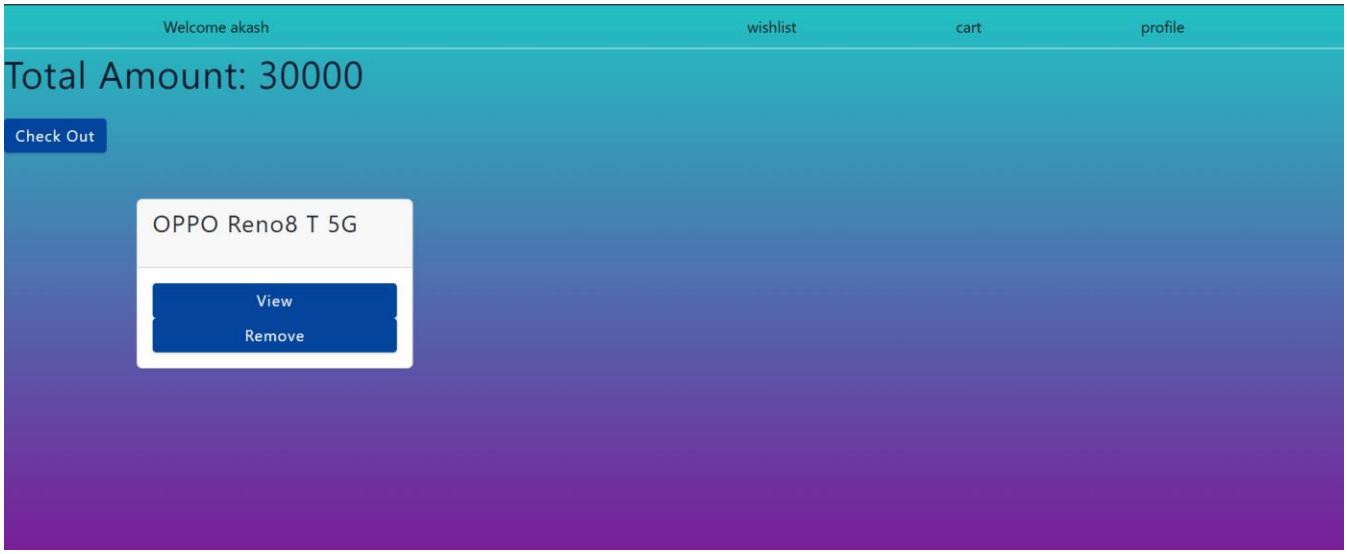
9) View product:



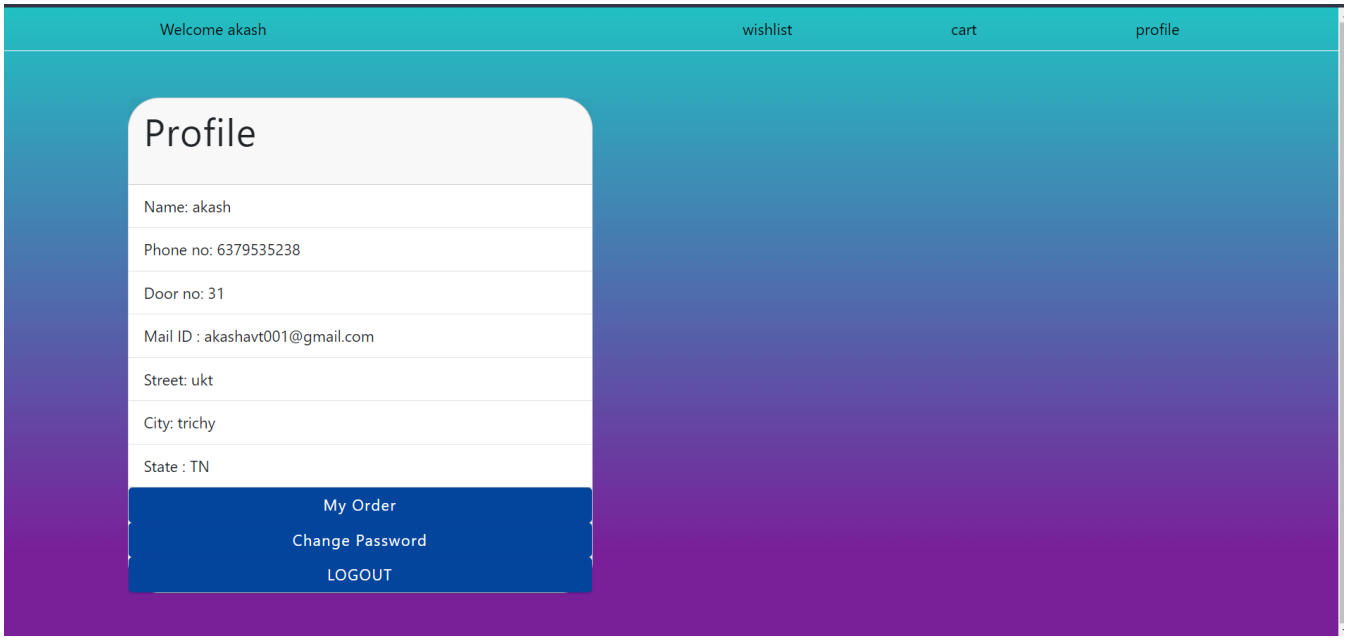
10)Wishlist :



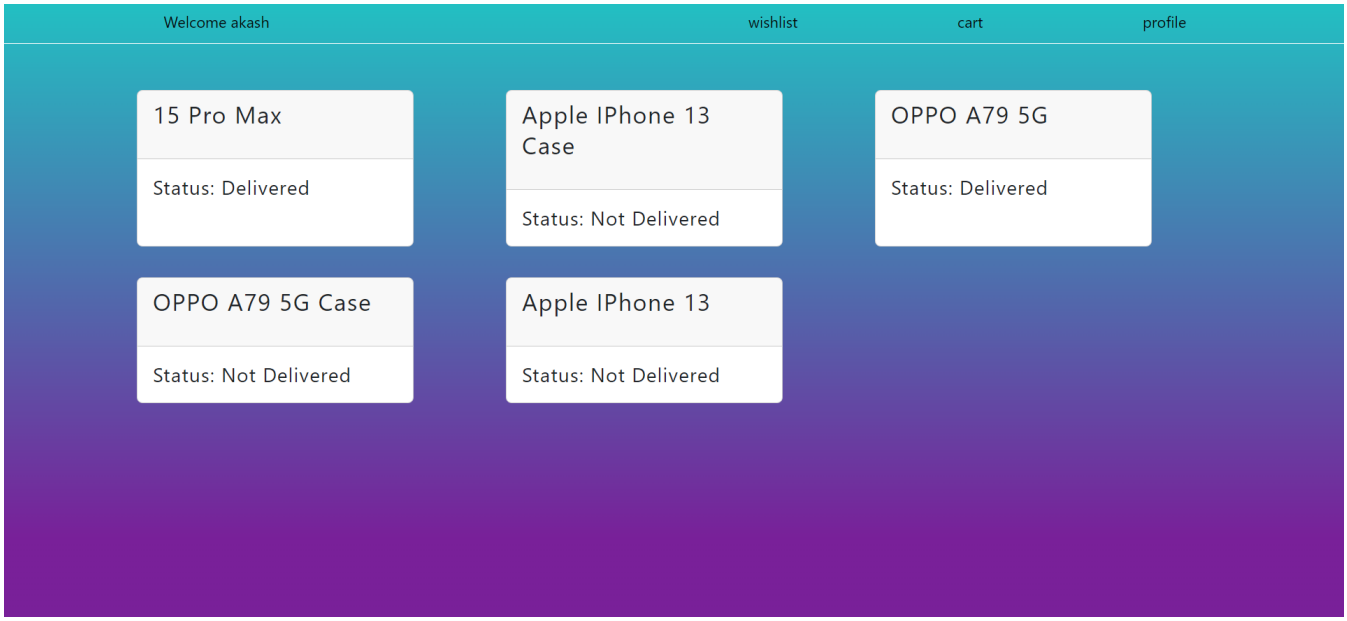
11)cart:



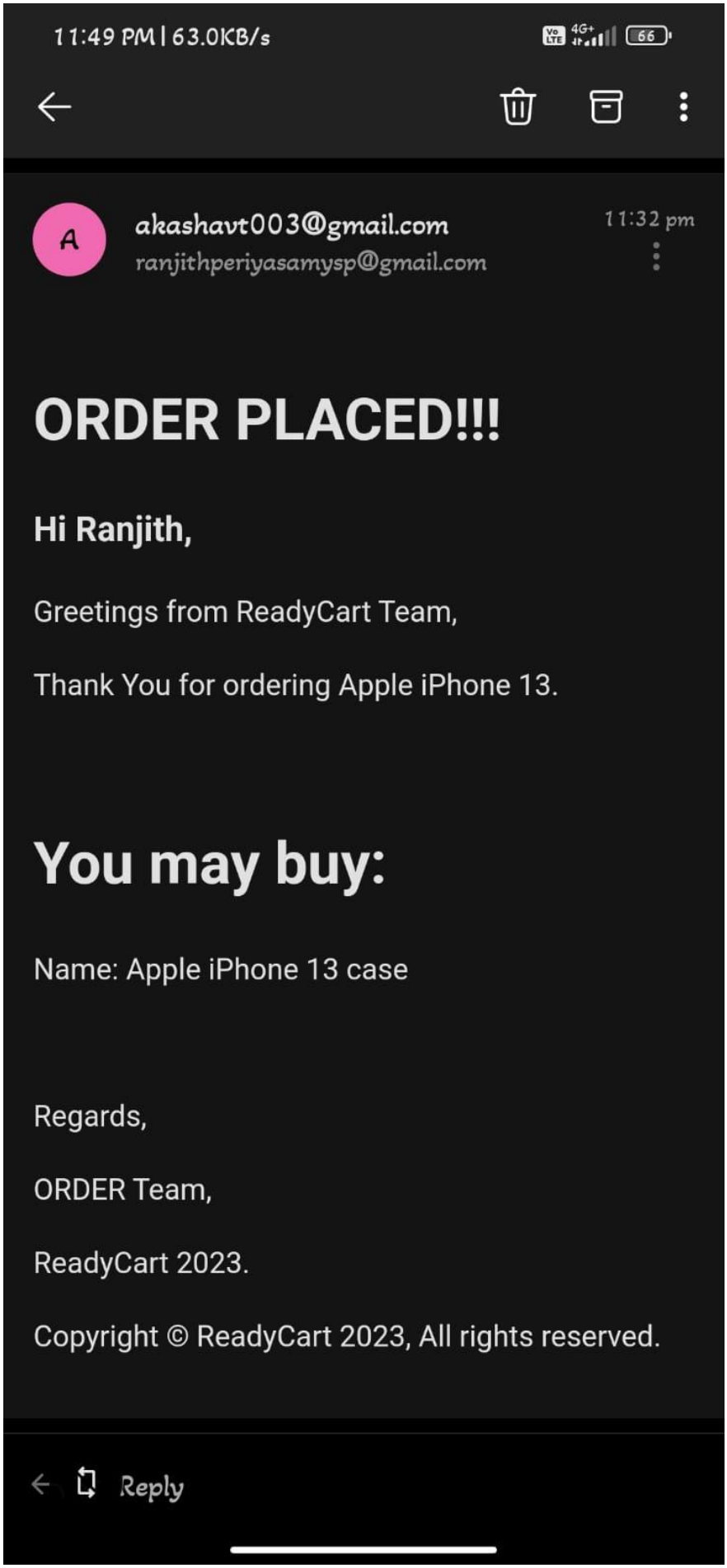
12)Profile:



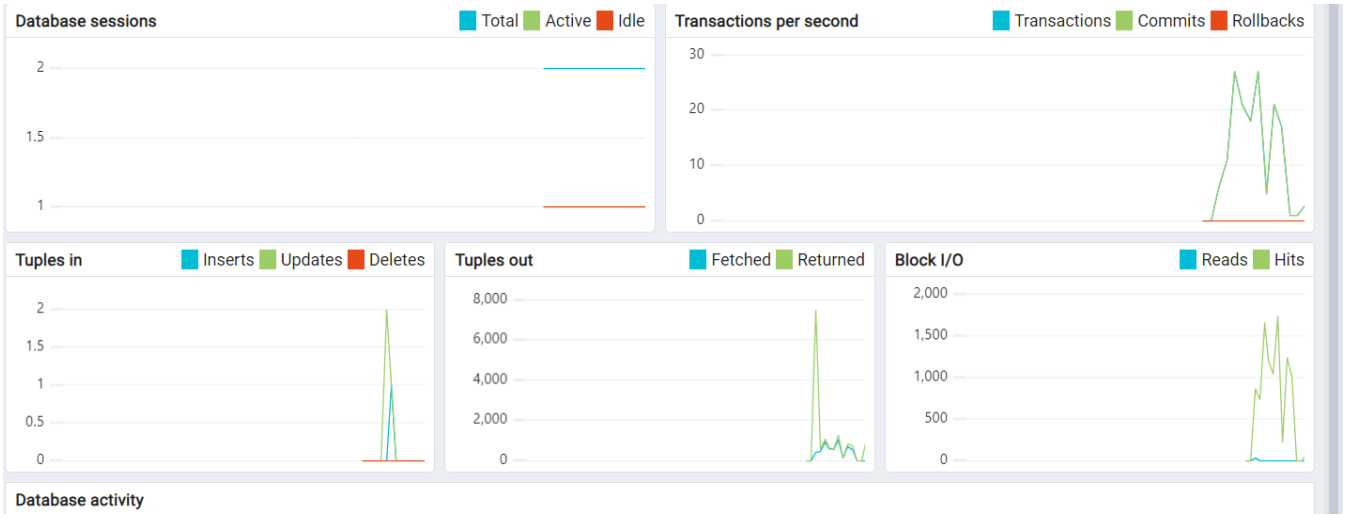
13)Order


Welcome akash
wishlist
cart
profile
15 Pro Max
Status: Delivered
Apple IPhone 13 Case
Status: Not Delivered
OPPO A79 5G
Status: Delivered
OPPO A79 5G Case
Status: Not Delivered
Apple IPhone 13
Status: Not Delivered

14)Mail


11:49 PM | 63.0KB/s
A
akashavt003@gmail.com
ranjithperiyasamysp@gmail.com
11:32 pm
ORDER PLACED!!!
Hi Ranjith,
Greetings from ReadyCart Team,
Thank You for ordering Apple iPhone 13.
You may buy:
Name: Apple iPhone 13 case
Regards,
ORDER Team,
ReadyCart 2023.
Copyright © ReadyCart 2023, All rights reserved.
Reply

14)Database:



LINK FOR SOURCE CODE:

https://github.com/akash1106/ecom.git