Department of Computer Engineering
Faculty of Technology
Dharmsinh Desai University, Nadiad

PROJECT REPORT ON

Online Rent Services

By

Name	Roll No	ID
Harshit Tarsariya	CE136	18CEUBG080

B.Tech CE Semester-VI

Subject: Object Oriented Software Engineering

Guided by:

Prof. Mrudang T. Mehta
Associate Professor Dept. of
Computer Engineering. (DDU)

Dharmsinh Desai University, Nadiad
Faculty of Technology
Department of Computer Engineering

CERTIFICATE

This is to certify that the practical / term work carried out in the subject of **Object Oriented Software Engineering** and recorded in this journal is the bonafied work of

HARSHIT TARSARIYA (CE136) (18CEUBG080)

of B.Tech semester **VI** in the branch of **Computer Engineering** during the academic year **2020-2021**.

Prof. Mrudang T. Mehta
Assistant Professor of Department of Computer Engineering, Dharmsinh Desai University, Nadiad

Dr. C. K. Bhensdadia
Head of Department of Computer Engineering, Dharmsinh Desai University, Nadiad

Table of Contents

Abstract

Online Rent Services aims for providing the platform to individuals to rent an item or give the item on rent and earn some amount from rent. Buyer has to pay the rent as well as the deposits which will be refunded later on.

Software Requirements Specification

For

“Deal On Rent”

Online Rent Services

Version 1.0 approved

Prepared by Harshit Tarsariya

12th December, 2020

Table of Contents

Revision History

Name	Date	Reason For Changes	Version

1. Introduction

1.1 Purpose

The main purpose of this document is to give detailed understanding of the product, Online Rent Services. Online Rent Service is a platform which fills gap between party who requires their product to be given on rent and the party who want to buy product on rent, without any brokerage.

1.2 Document Conventions

The Document uses following conventions:

1) Convention for Main headings

Font name: Arial
Font type: Bold
Font color: Black
Font size: 18

2) Convention for details

Font name: Arial
Font type: simple
Font color: Black
Font size: 14

3) Comments is in italics and written between /* and */

1.3 Intended Audience and Reading Suggestions

The intended audience for the document is developers, managers and the client who are going to build the product. Read the Scope and then read requirements thoroughly.

1.4 Product Scope

The purpose of the Online Rent Services is to fill the gap between the one party who needs to rent products and the other party who needs to buy products on rent. The product is available to any person who needs renting support.

The Scope of application allows users to:

- Put Product on rent
- Buy Product on rent

- Track Product
- Cancel Product
- Pay Deposits
- Rating to any renter

2. Overall Description

2.1 Product Perspective

Online Rent Services is extended version of online selling where products are just sold, here the products can be given on rent. People can also track the orders they have made.

2.2 Product Functions

Online Rent Services provide functions like uploading product on rent, buy product on rent, providing ratings to product as well as renter, Cancel product, Track order and Pay deposits.

2.3 User Classes and Characteristics

There are three user classes:

- End user: This user is able to perform all task which are not privileged
- Courier Service user: This user is able to accept and reject the task of shipping
- Admin user: This user is able to perform privileged tasks

2.4 Operating Environment

Operating Environment for Online Rent Services are:

- Operating System: Windows, Linux, MACOS
- Browser: Latest Chrome, Mozilla Firefox or any other browser

2.5 Assumptions and Dependencies

The transactions performed in application is assumed to be atomic and are performed successfully. For location related services Google API is used.

3. External Interface Requirements

3.1 User Interfaces

- Front End: Angular, HTML, CSS, JS

- Back End: NodeJs

3.2 Hardware Interfaces

- Windows, Linux, MACOS
- A Browser which supports HTML, JS

3.3 Software Interfaces

Software has Front-end in Angular is interfaced with NodeJs which is interfaced with MongoDB for fetching and storing the data.

3.4 Communications Interfaces

Email Services(SMTP) are used to mail user during registration process. All input data from user is taken in Electronic forms. HTTP protocol is used in communicating with NodeJs through API call.

4. System Features

4.1 User Authentication

4.1.1 Description and Priority

Feature is used during authentication of any type of users. It is High priority as it relates with the security issues.

4.1.2 Stimulus/Response Sequences

User has to provide all input values in the form and has to submit form and success or failure response is provided by the System.

4.1.3 Functional Requirements

R1.1 Register

Input: First name, Last name, username, password, email, mobile number

Output: Successful message and mail is sent for confirmation for registered mail

Description: When user provides all information data is stored in database and user is successfully registered

R1.2 Login

Input: Unique Username/Email, password

Output: Message for logged in

Description: After providing details , user is validated against credentials and on successful validation redirected to particular home page

R1.3 Forget Password

Input: Email
Output: OTP is sent on the Email if it is registered
Description: After providing username , otp is sent on email and user is able to change password

4.2 User Profile

4.2.1 Description and Priority

Feature is used in User Registered Profile where user can edit and view profile. It is low priority as it seems to be not main functionality.

4.2.2 Stimulus/Response Sequences

User has to provide request for the profile and profile is shown in response.

4.2.3 Functional Requirements

R2.1 View Profile
Input : User selection
Output : Users profile with data is shown
Description : When user is selecting profile to see then user is shown the data of particular user.

R2.2 Edit Profile

R1.2.1 Edit password
Input: Password, Confirm Password
Output: Changes on Page
Description: After editing password database is reflected.

R1.2.2 Edit other information
Input: Edit Selection
Output: Edited Profile after edit
Description: Logged In User changes the profile details such as first name, last name, email, mobile number

4.3 Rent Item

4.3.1 Description and Priority

Feature includes the items to be rented and the item to be rent. It has high priority as it is the main functionality.

4.3.2 Stimulus/Response Sequences

User has to provide all input and particular action is performed.

4.3.3 Functional Requirements

R3.1 Upload Rent Item

Input: Name, renting price, deposits, availability, minimum days to be rented, Image, tag to which item is related.

Output: Item posted on platform

Description: When all goes successful then database is updated and item posted on platform

R3.2 Buy Rented Item

Input: Select buy input, timeinterval for which item is to be taken, and pay deposits

Output: Item bought successfully and on the way

R3.3 Delete Rent Item
Input: Delete input
Output: Item is deleted from the platform
Description: User who has uploaded item to be given on rent selects delete button and if there is no person having that item on rent then item can be removed.

R3.4 Edit Rent Item
Input: Edit Input
Output: Item is updated on platform
Description: User who has uploaded item can edit its details

4.4 Courier Item

4.4.1 Description and Priority

Feature is used by courier service to ship item. It has High priority as it related with the delivering of item.

4.4.2 Functional Requirements

R4.1 Accept item to courier

Input: select accept input

Output: Successful message

Description: When user buys item, courier service accepts the order to be delivered to the buyers address.

R4.2 Delivered item

Input: Select item accepted

Output: Delivered Successfully message

Description: When courier service delivers the item to buyers they can be item as delivered

R4.3 Track item
Input: Select item ordered
Output: Status of shipment
Description: User can track the status of the item ordered

4.5 Statistics

4.5.1 Description and Priority

Feature is used by user to look certain stats. It is of medium level priority

4.5.2 Functional Requirements

R5.1 Past orders on item

Input: Item

Output: Stats of past orders placed by people on that item

R5.2 Past orders by particular user
Input: Past order input
Output: Stats of past orders
Description: Shows the past orders placed by users

R5.3 Ratings
Input: Item
Output: Rating provided by the person who buys the product
Description: User after taking item on rent can provide ratings to products which is publicly visible

5. Other Nonfunctional Requirements

5.1 Performance Requirements

The System must be interactive and delays involved should be less. So in system fetching data and storing, there should not much delay. Fetching all data from database at a time is not reasonable approach, so data should be retrieved as and when required by the user.

5.2 Safety Requirements

User details should be securely stored in database. The main security concern is user login, so proper login mechanism should be created to prevent security flaws.

5.3 Security Requirements

Validation of user during registration is must. Consider the certain attack like CSRF and provide solution to them accordingly. The password of the users stored in database should be encrypted.

5.4 Software Quality Attributes

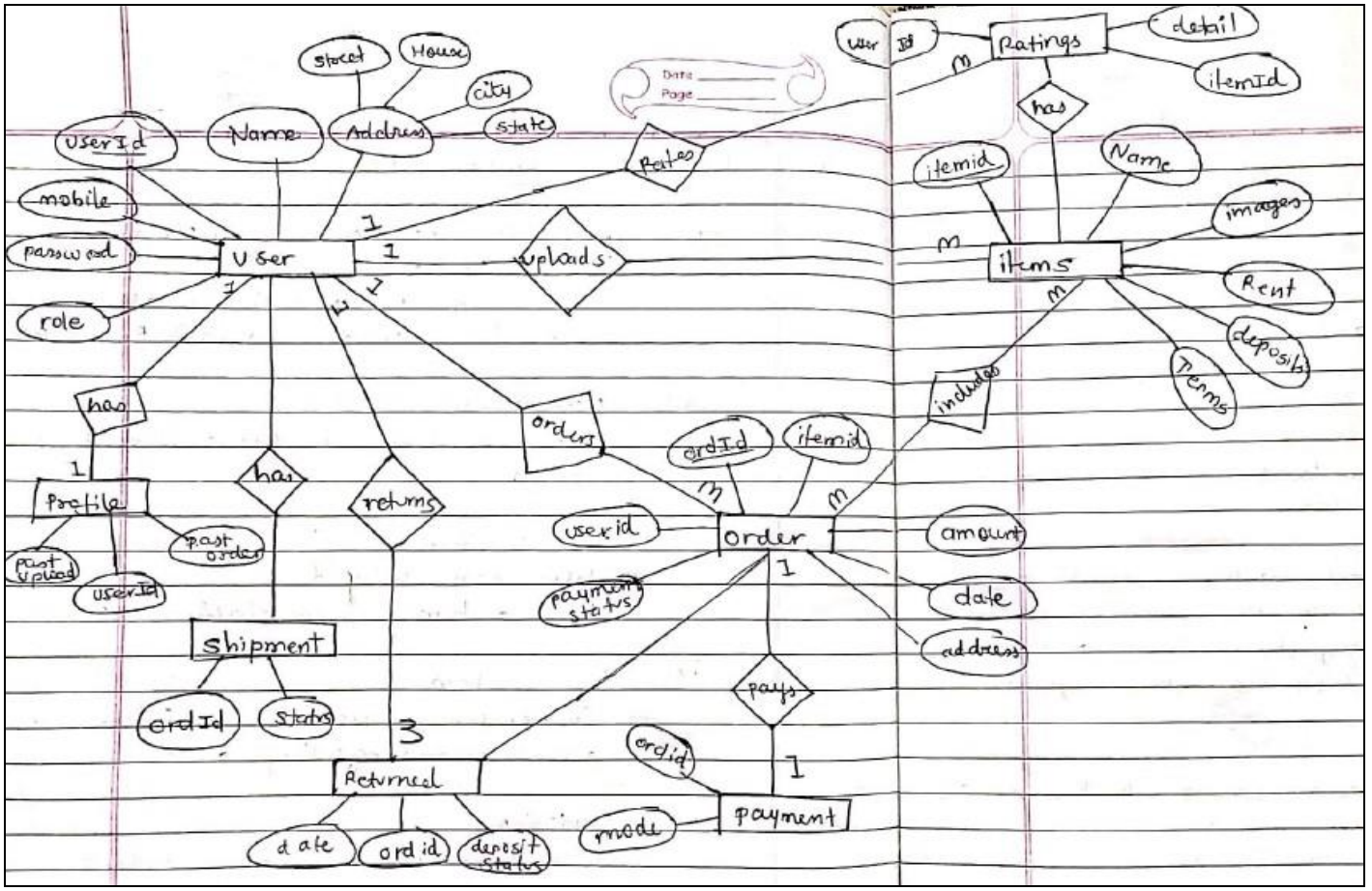
- Availability of item: If item is not hold by any party then that item should be available to any user
- Correctness: The product should be delivered to the party who had ordered
- Maintainability: The product should maintained by the admin so that fake user can be recognized
- Usability: The Courier service should accept orders such that maximum number of customers need is satisfied

5.5 Business Rules

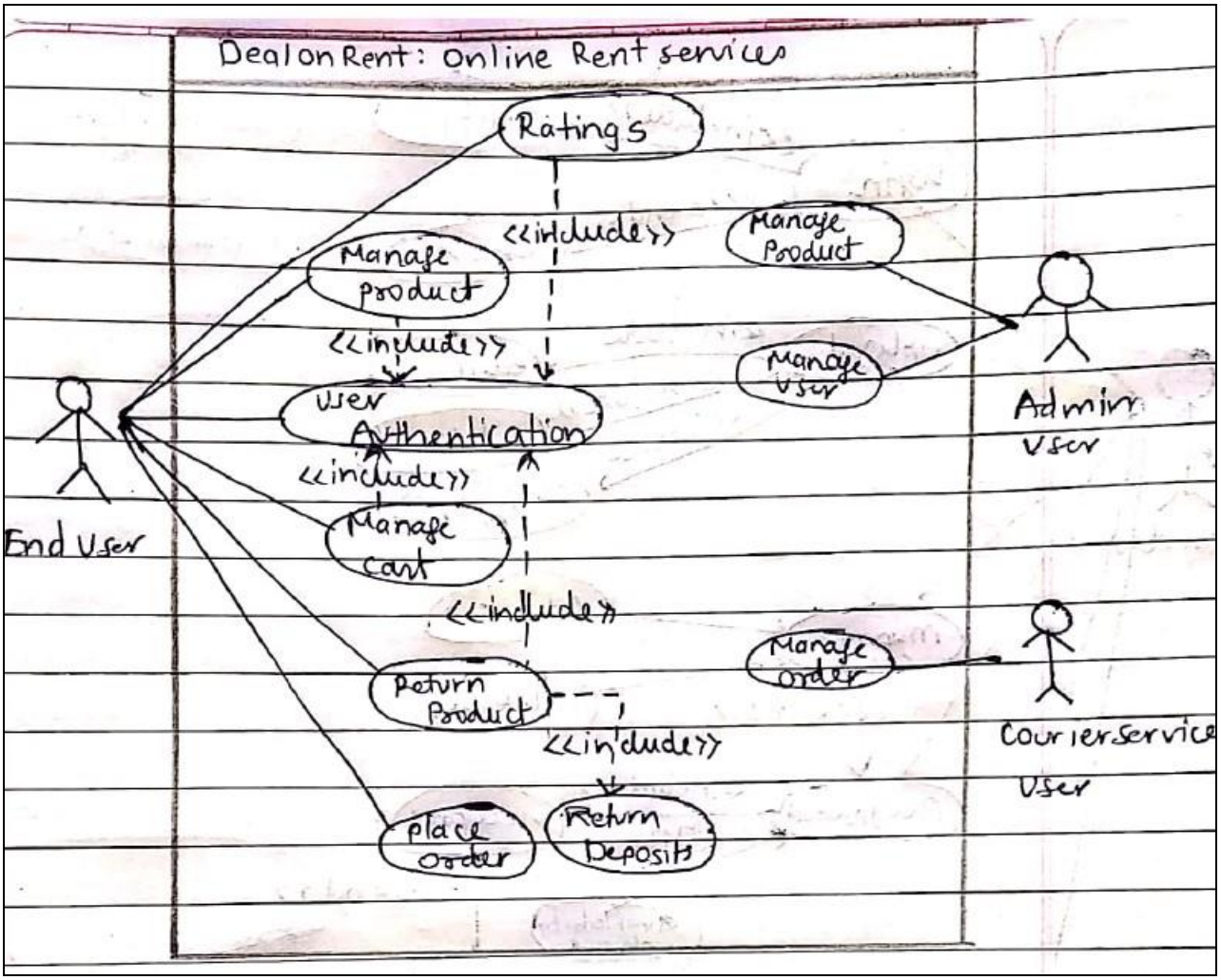
The admin is only user who can perform privilege task such as removing item from platform, warn user for breaking certain policies and guidelines , and the user’s history of orders.

Analysis and Design

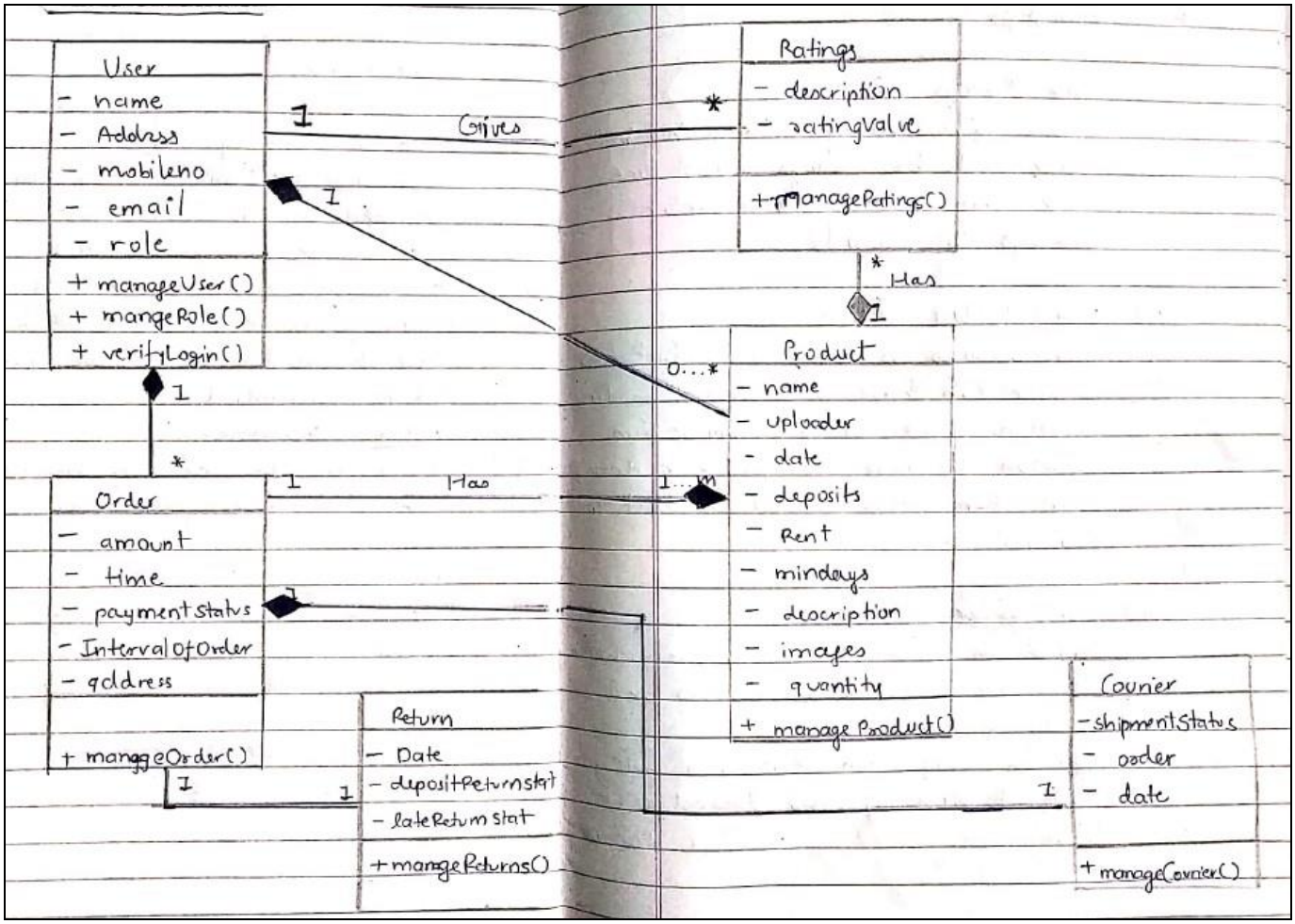
1.E-R Diagram:



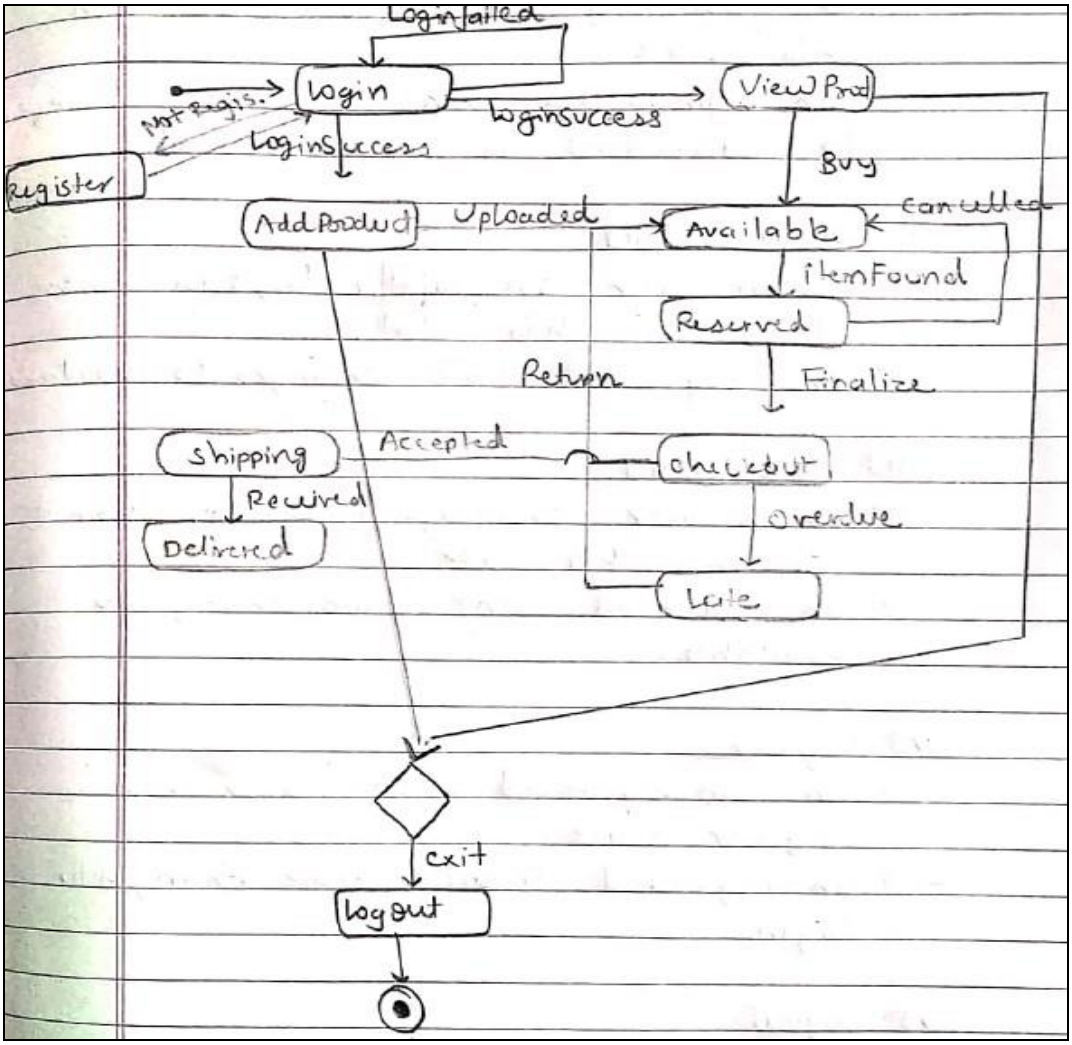
2. USE CASE DIAGRAM:



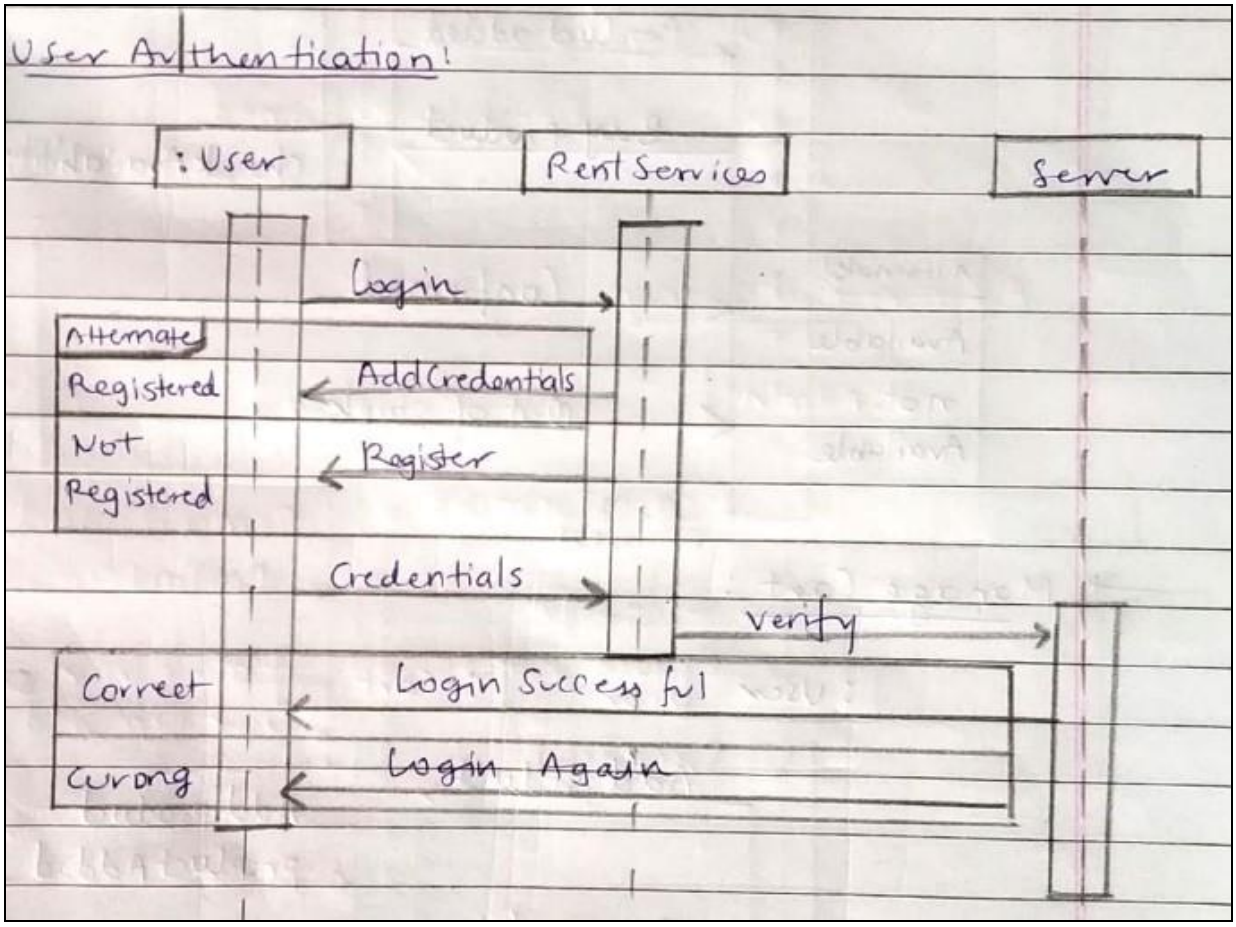
3. CLASS DIAGRAM:



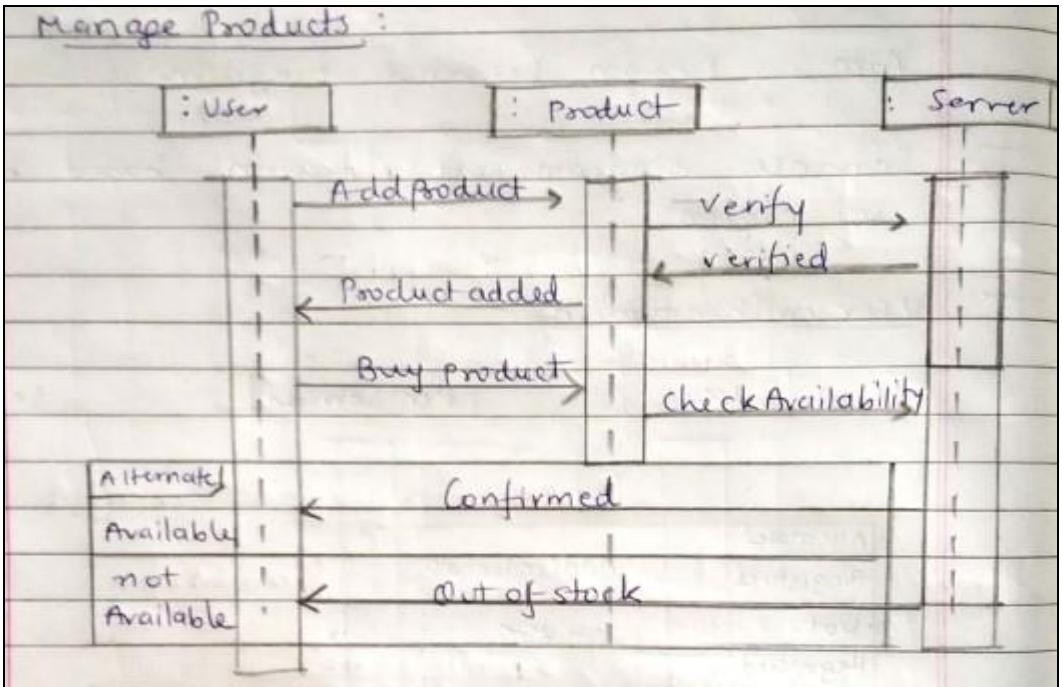
4. STATE DIAGRAM :



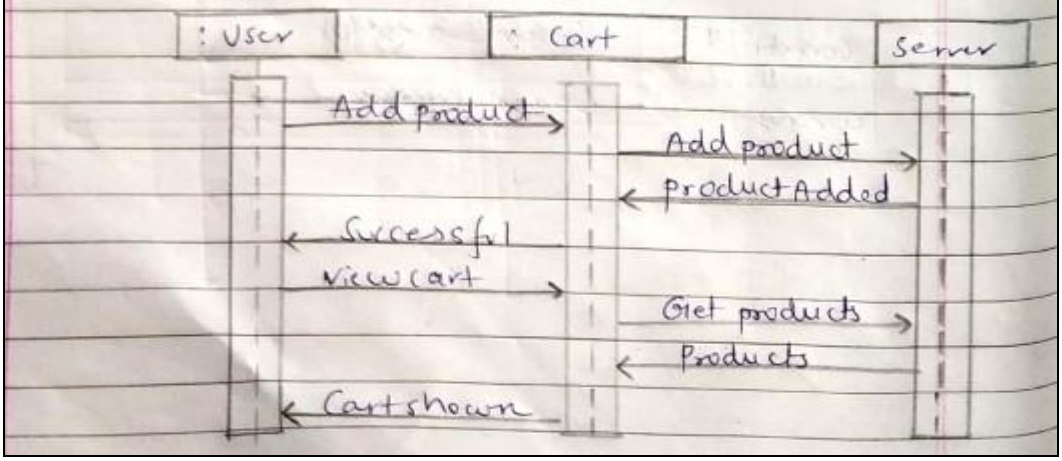
5. SEQUENCE DIAGRAM:



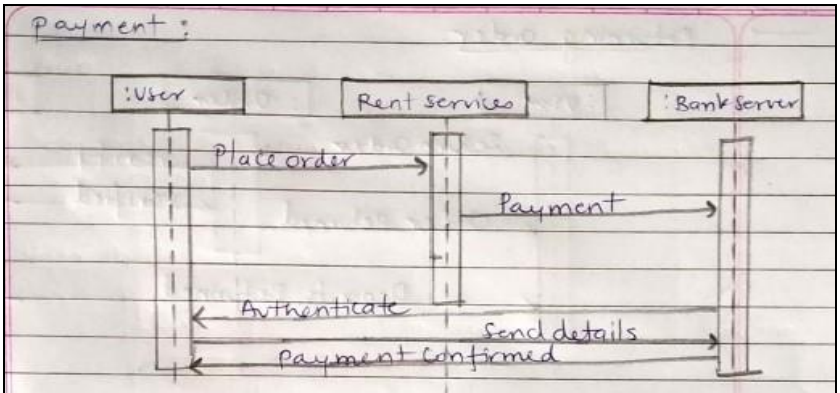
Manage Products :



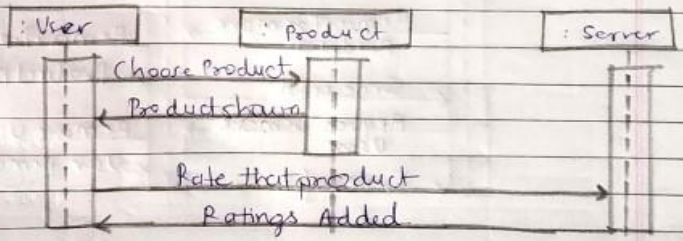
Manage Cart :



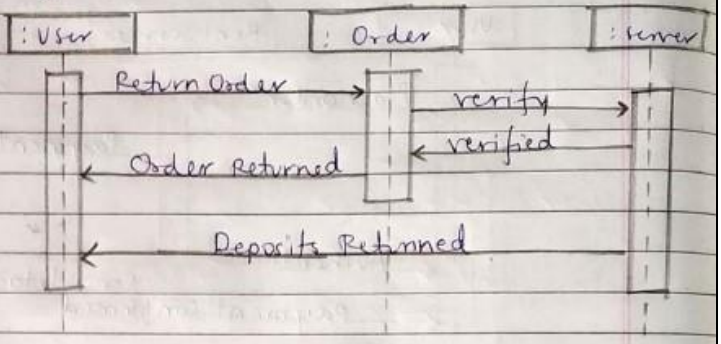
Payment :



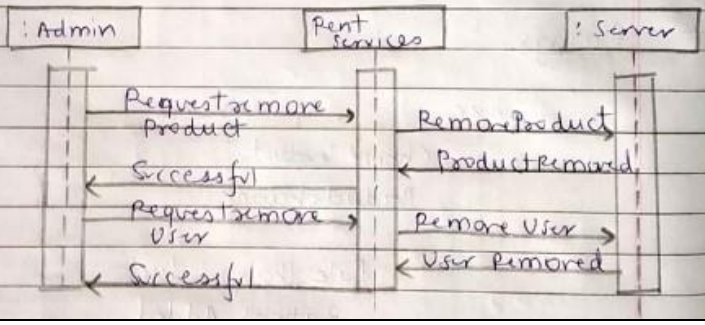
Ratings :



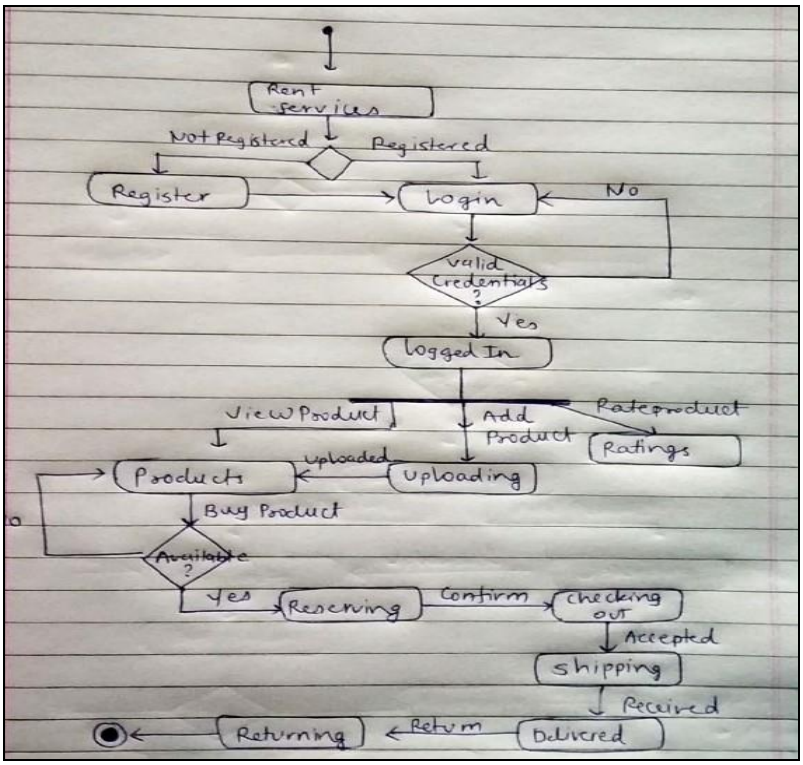
Returning order :



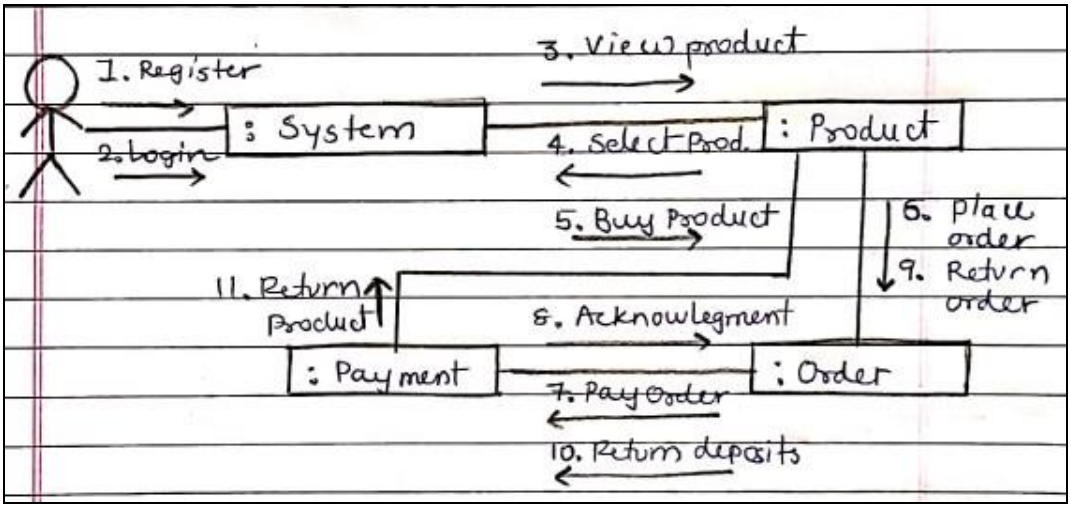
Admin Managing :



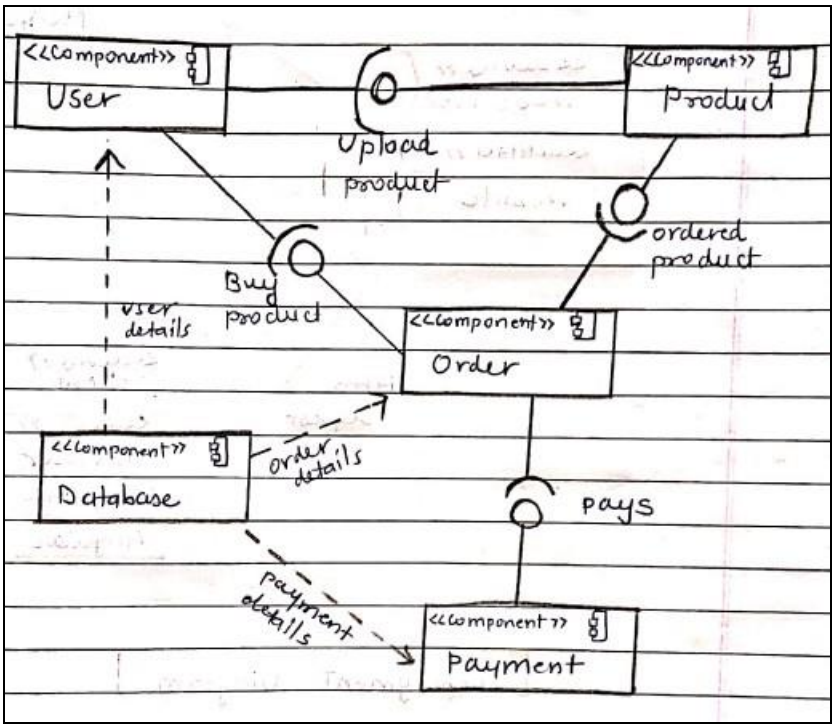
6. ACTIVITY DIAGRAM:



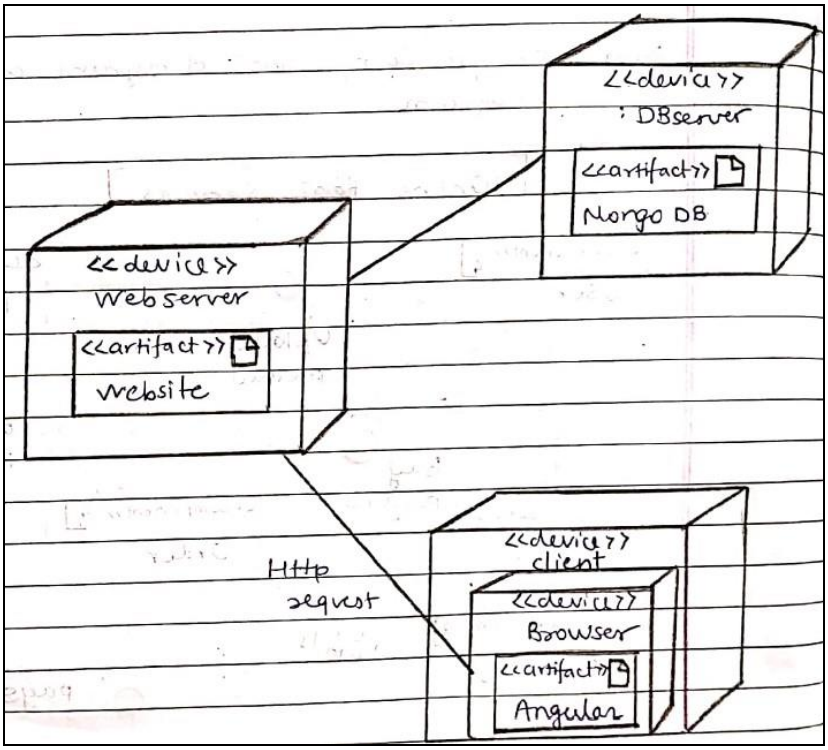
7.COLLABRATION DIAGRAM:



8.COMPONENT DIAGRAM:



9.DEPLOYMENT DIAGRAM:



10.PACKAGE DIAGRAM:

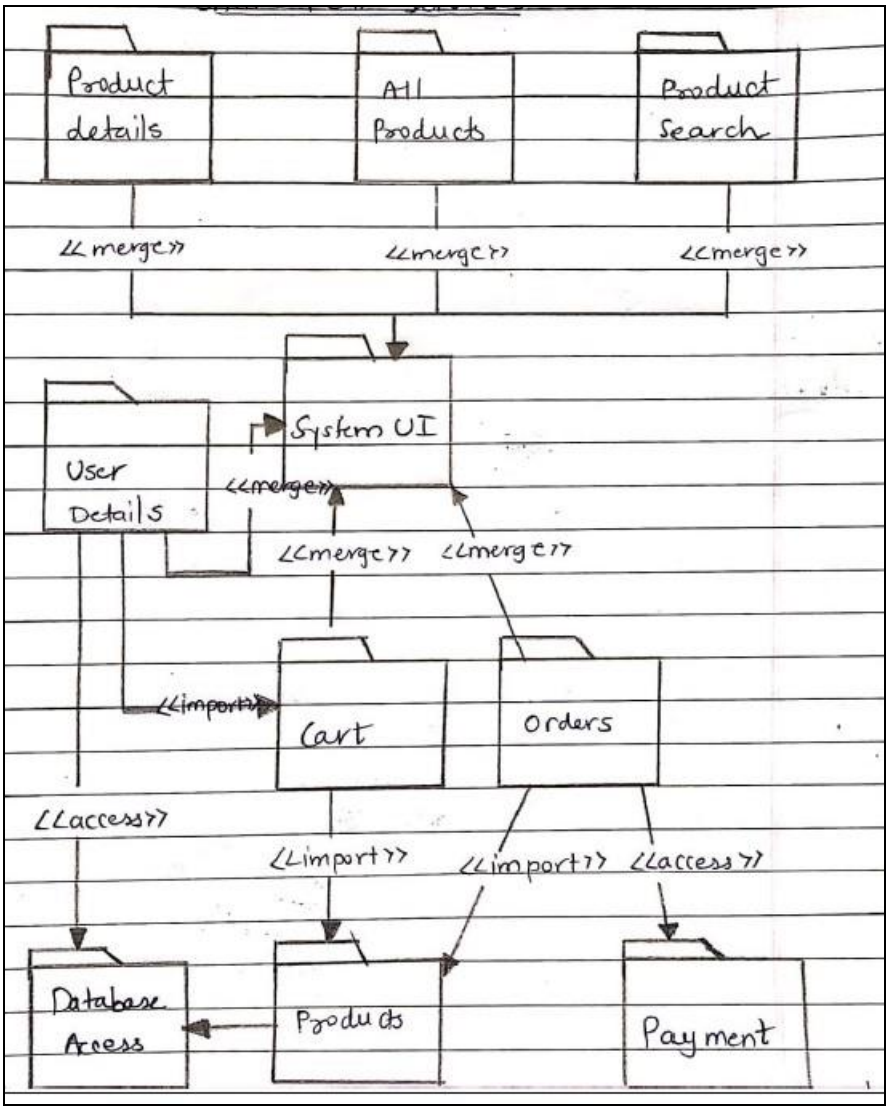


TABLE DESIGN

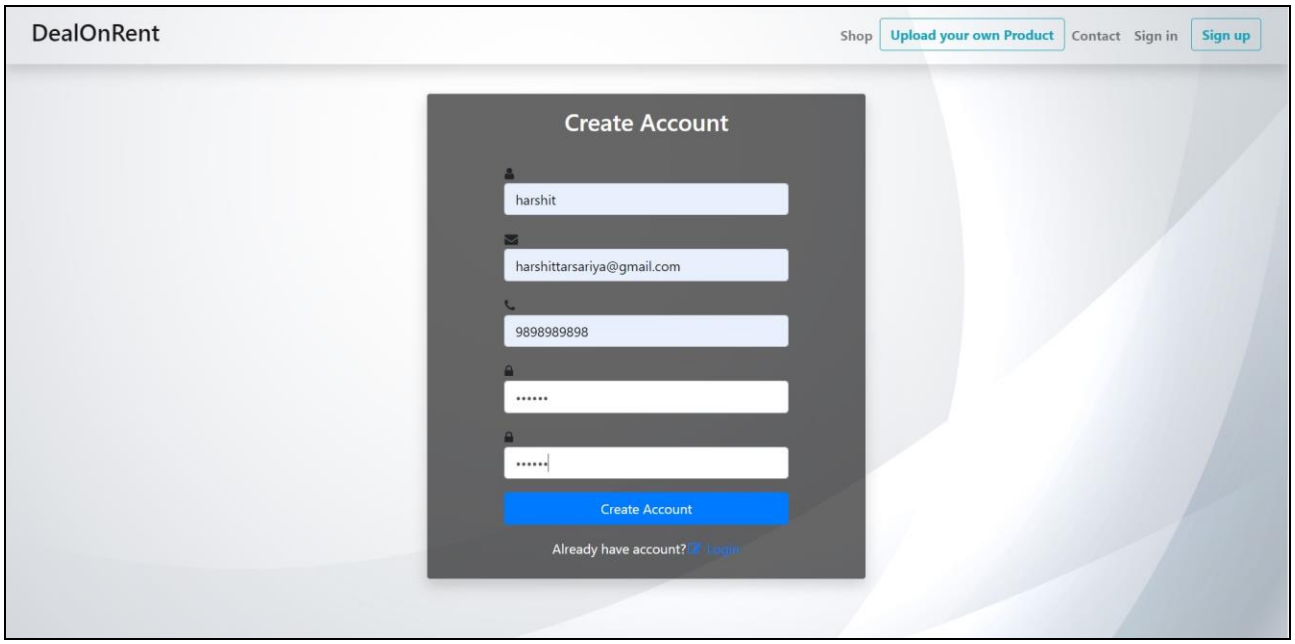
(MONGODB)

User’s Table:

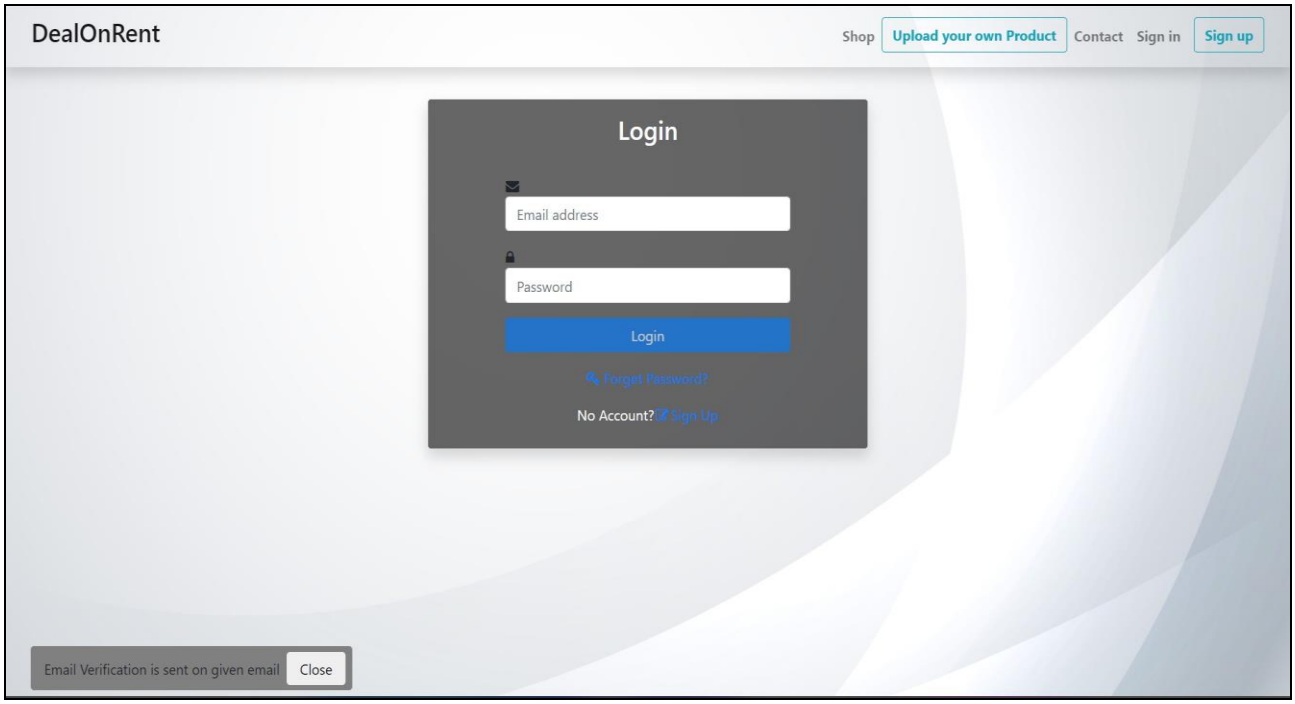
Product’s Table:

SCREENSHOTS OF PROJECT:

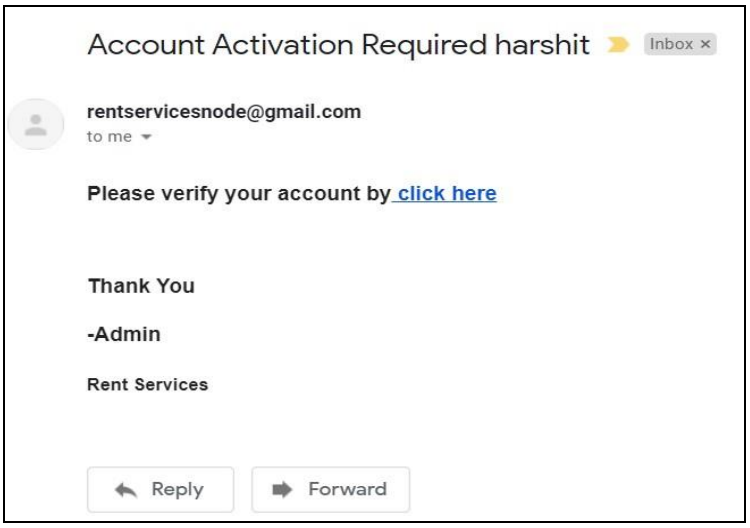
Registration



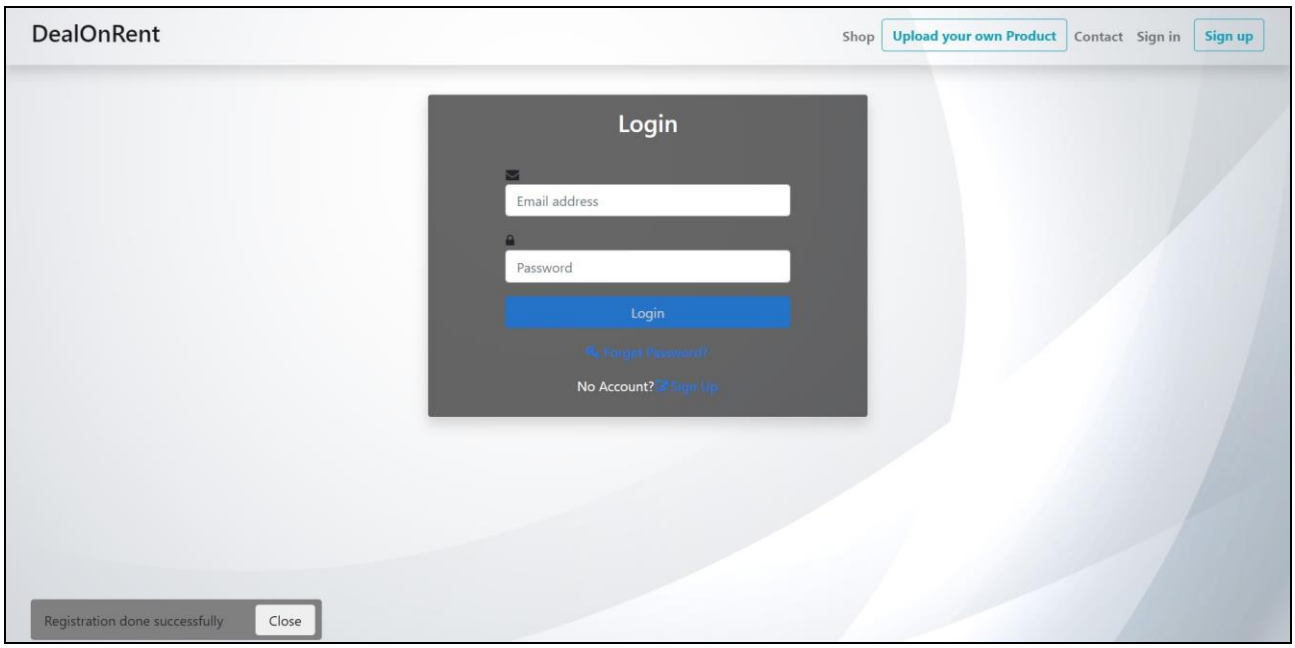
Registration Link sent on registered mail



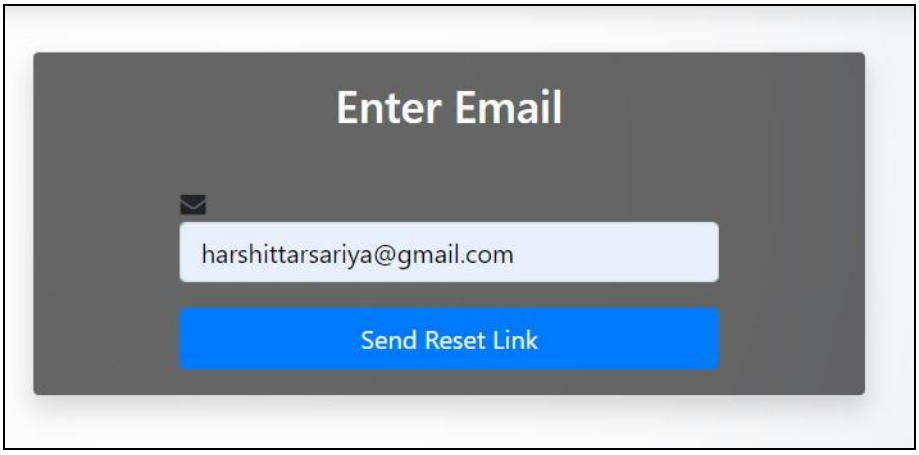
Verifying mail for account



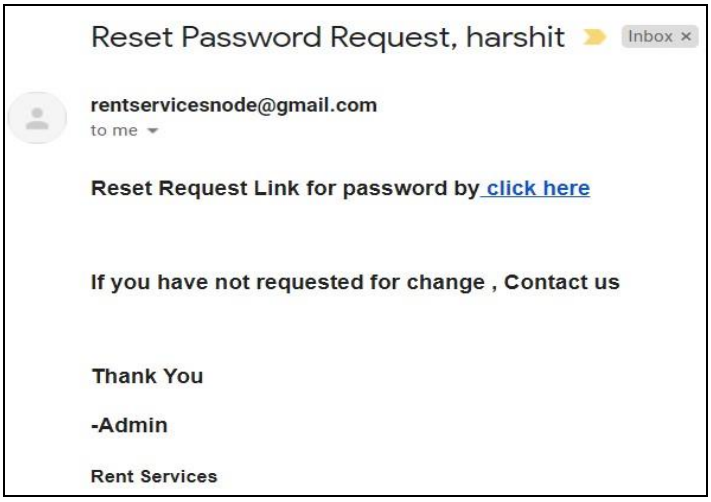
Registered Successfully



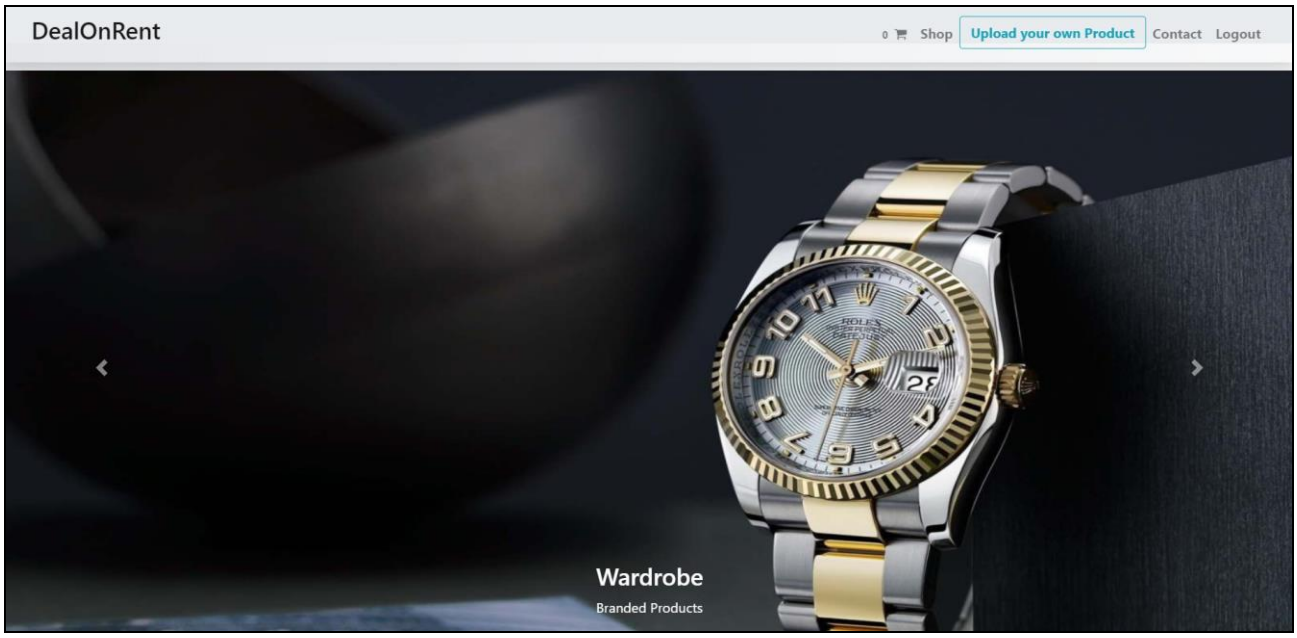
Forget Password: Written Registered Email



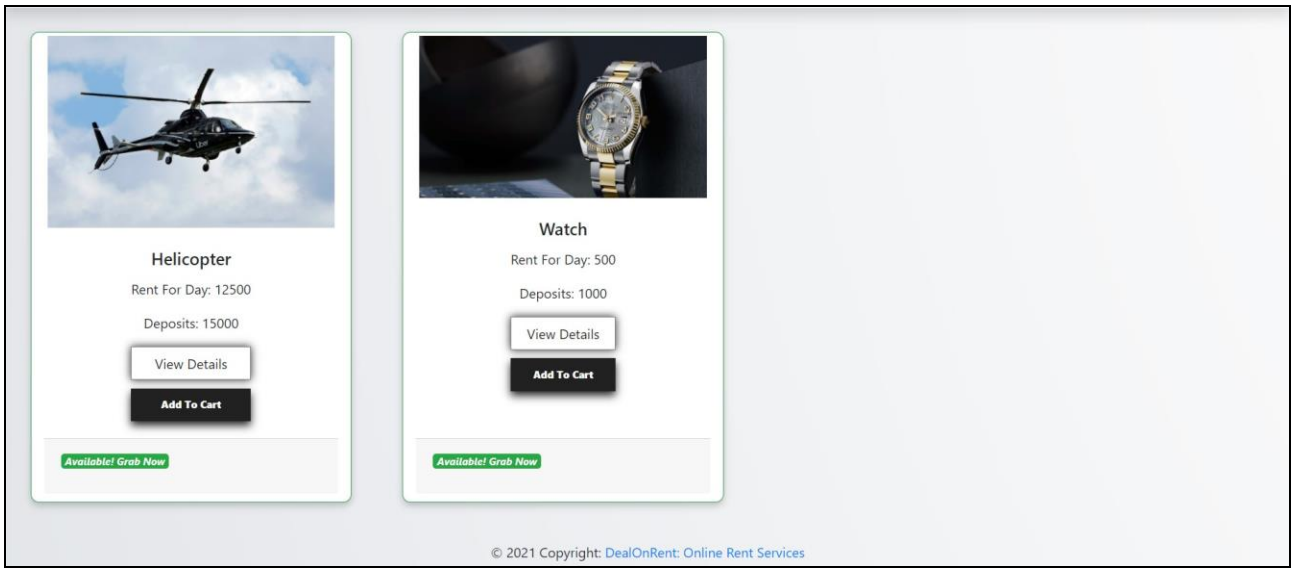
Reset Password link, which will be valid only for some time



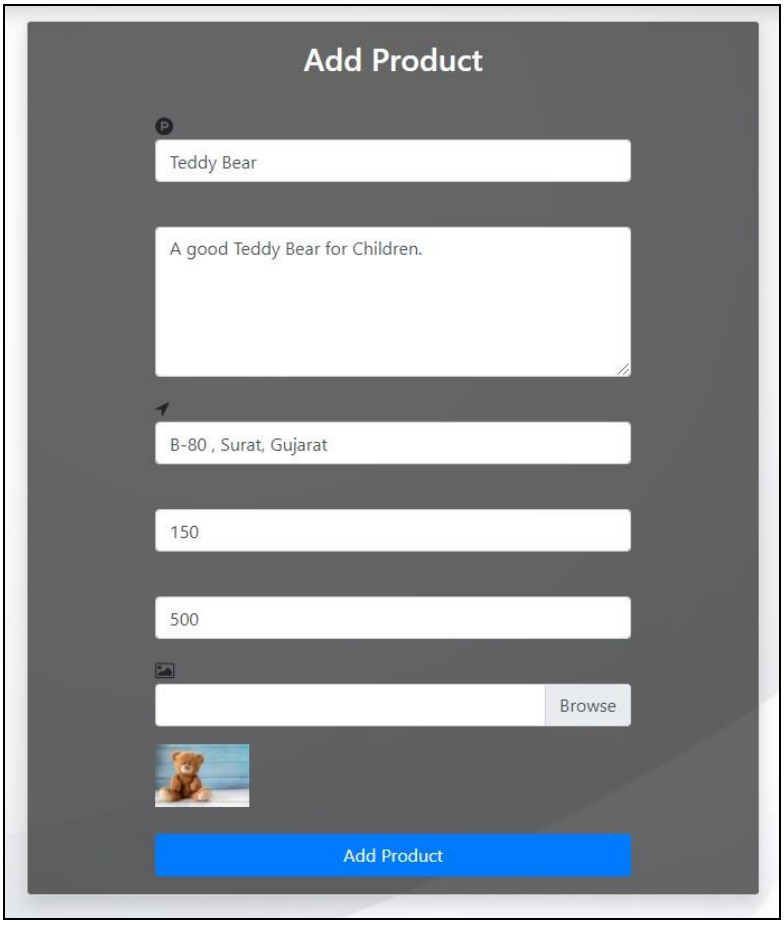
Home Page



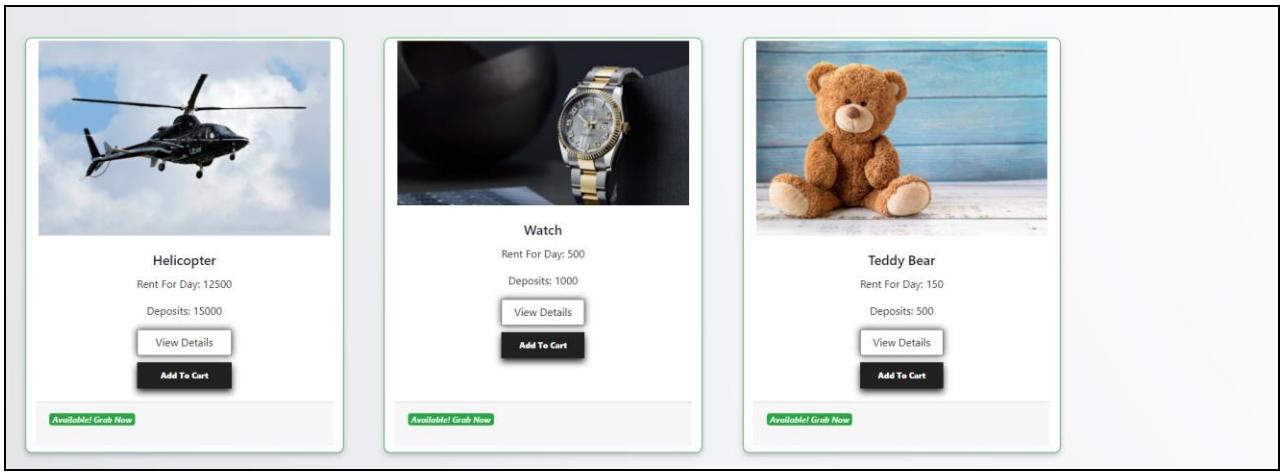
Products



Add your own Products



Product added Successfully



IMPLEMENTATION DETAILS

- Project is developed using Angular as Front end and NodeJs as Backend.
- Server side state management is done using JSON Web tokens.
- Verification and Reset Links are also the JSON Web tokens with limited lifetime.
- State management at front end is done using Redux.
- When any request is to be made http client is used which calls backend server at specific endpoint and request is fulfilled.
- Various services are created to handle request from common ground.
- Dependency Injection feature of Angular is used for injecting services.

TESTING

Functionalities	Field Input	Expected output	Actual output
Login	Email and Password	Redirect to Home page	Redirected to Home page
Registration	Details in Form	Send Mail for verification	Mail sent for verification
Add to cart	Product	Product added to cart	Product added to cart
Add Product	Product details with Image	Product Added to Wardrobe	Product added to Wardrobe
Logout	No input needed	Redirect to login page	Redirect to login page
Forget Password	Enter Registered Email	Reset Password Link sent on mail	Reset Password Link sent on mail

CONCLUSION AND FUTURE EXTENTION

Conclusion:

The project was completed with following functionalities:

- Login
- Registration
- Forget Password
- Home Page Products shown

- Add Product
- Add to cart
- Verifying by Mail

Of-course there is always a scope of extention. Online Rent Services can be extended by adding the remaining functionalities of buying. Google authentication can also be included. Coupons for benefits of customers can be made. Brand advertisement on Home Page. Product Recommendation system can be added with help of Machine Learning.

REFERENCES

- https://stackoverflow.com/
- https://nodejs.org/en/
- https://redux.js.org/
- https://www.lucidchart.com