**ANDROID BASED ELECTRONIC PRODUCT SERVICING SYSTEM**

**by**

**MD TASLUF MORSHED**

**ID: 191-15-12089**

**AND**

**MD ASSADUJJAMAN TILOK**

**ID: 191-15-12594**

This Report Presented in Partial Fulfillment of the Requirements for the Degree of Bachelor of Science in Computer Science and Engineering

Supervised By

**Dr. MD. FOKHRAY HOSSAIN**

Dean, FSIT &

Director, International Affairs

Daffodil International University

Co-Supervised By

**Mr. MD. AZIZUL HAKIM**

Sr. Lecturer

Department of CSE

Daffodil International University

****

**DAFFODIL INTERNATIONAL UNIVERSITY**

**Dhaka, Bangladesh**

**December 2022**

**APPROVAL**

This Project titled “**Android Based Electronic Product Servicing System**”, submitted by Md Tasluf Morshed and Md Assadujjaman Tilok to the Department of Computer Science and Engineering, Daffodil International University, has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on \*date\*.

**Board of Examiners**

**(Name) Chairman**

**Designation**

Department of CSE [**Font-12**]

Faculty of Science & Information Technology

Daffodil International University

**(Name) Internal Examiner**

**Designation**

Department of CSE

Faculty of Science & Information Technology

Daffodil International University

**(Name) External Examiner**

**Designation**

Department of -------

Jahangirnagar University

**Declaration**

We hereby declare that, this project has been done by us under the supervision of **Dr. MD. FOKHRAY HOSSAIN,** Dean, FSIT & Director, International Affairs, Department of CSEDaffodil International University. We also declare that neither this project nor any part of this project has been submitted elsewhere for award of any degree or diploma.

**Supervised by:**

**Dr. MD. FOKHRAY HOSSAIN**

Dean, FSIT &

Director, International Affairs

Daffodil International University

**Co-Supervised by:**

**Mr. MD. AZIZUL HAKIM**

Sr. Lecturer

Department of CSE

Daffodil International University

**Submitted by:**

**Md Tasluf Morshed**

ID: -191-15-12089

Department of CSE

Daffodil International University

**Md Assadujjaman Tilok**

ID: -191-15-12594

Department of CSE

Daffodil International University

**ACKNOWLEDGEMENT**

First we express our heartiest thanks and gratefulness to almighty God for His divine blessing makes us possible to complete the final year project/internship successfully.

We really grateful and wish our profound our indebtedness to **Dr. MD. FOKHRAY HOSSAIN,** Dean, FSIT & Director, International Affairs, Daffodil International University, Dhaka. Deep Knowledge & keen interest of our supervisor in the field of “*Mobile Application & Management System*” to carry out this project. His endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior draft and correcting them at all stage have made it possible to complete this project.

We would like to express our heartiest gratitude to **Dr. Touhid Bhuiyan,** Professor andHead**,** Department of CSE for his kind help to finish our project and also to other faculty member and the staff of CSE department of Daffodil International University.

We would like to thank our entire course mate in Daffodil International University, who took part in this discuss while completing the course work.

Finally, we must acknowledge with due respect the constant support and patients of our parents.

**Abstract**

In The 21th century while the world is vastly depending on electronic goods and technology in that very time people from Bangladesh are facing difficulties to find a optimal servicing solution for their household necessaries. Sadly a number of factors can be accountable for the issue. Undoubtedly service policy is the root of it. However, this project intendant to build a service system, considered as “**Android Based Electronic Product Servicing System**” which could be the ultimate solution for the purpose. This project expect to reduce a lot of work load that people don't need to go out to find a servicing solution. User can simply register their problem to the application and then we will send a technician to solve the problem in a suitable time.

**CHAPTER 1**

**Introduction**

**1.0 Introduction**

The Project illustrate that “**Android Based Electronic Product Servicing System**” is a virtual store on internet confide on aggregation model, where user can hire technician based on their product of interest. User can simply register themselves by using a valid email or phone number to the system in order to take the services. The system is a package, used by service provider to improve the efficiency to their B2C business.

The most widely used operating system in the smartphone is Android and ios. Therefore, as a developer of the project, we are working on an android app and web application for this service. To make an android app they want to use React native. It's a JavaScript framework that helps us to build an android and ios app. It's built on top of the React framework. For the web application, this will use React framework. For the backend, it will use Nodejs and for the database, developer want to use MongoDB.

**1.1 Project Overview**

“**Android Based Electronic Product Servicing System**” is an optimal servicing solution for their electronic goods. From the site customer can hire technician based on their product requirement by following some easy steps.

**1.2 The Purpose of the Project**

Along with the rapid development of technology, the servicing system is not improving very speedily. It is considered to be a massive problem not only in Bangladesh but also South Asia. Which is generating a lot of controversy while many people are speaking strongly against this issue. Even today, whenever a household product collapse, user have to take this to a service point and wait for a long time to fix it.

Now it is necessary to structure the service system which is based on time efficiency and skilled technicians. This advanced system may upgrade the UN ethical servicing trends by developing a user-friendly application for stakeholders. Therefore, the purpose of this project is to develop a “**Android-based Electronic Product Servicing System**” in order to reinforce the user’s satisfaction.

**1.3 Stakeholders**

* **Visitor:** Visitor can view the available services on the site.
* **Customer:** Customer can choose any services and make payment from the site.
* **Admin:** An Admin have some additional privilege and access including all the privilege that visitor and customer had.

**1.4 Project Timeline**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Task Name | Start | End | Status |  |  |  |  |  |  |  |
|  |  |  |  | Sep- Oct | Nov- Dec | Jan- Feb | Mar- Apr | May- Jun | July- Aug | Sep- Oct |
| Project Proposal | 17-09-21 | 29-09-21 | Complete |  |  |  |  |  |  |  |
| Requirement collect & Analysis | 02-10-21 | 17-12-21 | Complete |  |  |  |  |  |  |  |
| System Design | 24-12-21 | 08-02-21 | Complete |  |  |  |  |  |  |  |
| Coding | 09-01-21 | 24-05-22 | Complete |  |  |  |  |  |  |  |
| Testing | 12-06-22 | 23-08-22 | Complete |  |  |  |  |  |  |  |
| Documentation & Report | 09-8-22 | 05-09-22 | Complete |  |  |  |  |  |  |  |

**CHAPTER 2**

**REQUIREMENTS SPECIFICATION**

**2.1 Functional Requirements**

**2.1.1 Admin requirements**

* Admin login

This system to able to give permission to access the website to the admin and he can access the whole system.

* Database management

Admi controls the database and keep track of all records of product and client details.

* Manage service

Admin can add, Update and delete any service category. He can also view, add, update and delete any service brand name and model number.

* View customer details: Admin views the personal details of the customer.
* Manage Agent

Admin can view all the agents those who have contract with the system. He can add, and update any agent's information.

* Manage Technician

Admin can view all the technicians those who work for the system. He can add, and update any technician’s information.

* Manage Order

Admin can view and update any order. When a repair request is submitted into the system by a customer then admin can view that order. He can accept that order, assign technician for that order and finally confirm the order by updating it.

* Logout

Admin can logout form the system.

**2.1.2 User requirements**

* User Login:

Description of the feature

Using this feature user can login the system. A user should login with his/her user name and password to the system after registration. Invalid user name or password is not allowed to enter the system. A user can also login into the system by a google account.

Functional requirement

* + Username and password will be provided after user registration is confirmed.
  + Password must be hidden from others while typing it in the field.
  + A google login button should be implement to login by Google.
* New user registration:

Description of the feature

A new user will have to register in the system by providing essential details in order to purchase products in the system.

Functional Requirement

* + System should able to verify and validate information.
  + Password should be encrypted to provide security.
* View and update own details:

Description of the feature

Customer can view/update his personal information. Customer can also set default address.

* Choose a service:

Description of the feature

Customer can view all the service that are provided by the system at that time. Customer can also see the which brands are available and their model's number for a product to repair.

* Repair request:

Description of the feature

Customer can select any services, then fill up a form including with the details of the product and select the cross-ponding brand and model number for the product to send a repair request. He can also remove any service request from the cart by clicking remove. He can track his request.

* History:

Description of the feature

Customer can see all his previous service request and its details.

* Notification:

Description of the feature

Customer will get notification on every step of the order.

* Logout:

Description of the feature

Customer can logout from the system.

**2.2 Non-Functional Requirements**

**2.2.1 Performance requirements**

The system must accommodate high number of items without any fault and view information could not take longer than 3 seconds to appear on the screen.

**2.2.2 Usability requirements**

The android app is designed for user friendly environment and easy to use.

**2.2.3 Security**

* Functions of the app must be access in the way they were intended to be accessed.
* Included files shall not be accessed outside of their parent file.
* Administrator can only perform administrative task on pages they are privileged to access. Customer will not be allowed to access the administrator pages.
* API should be access only by authorized users.

**2.2.4 Error-handling**

App must handle its internal error and it should not terminate for an error. It should show the causes of the error to the users.

**2.2.5 Efficiency and Maintainability**

* Page loads should be returned and formatted in a timely fashion depending on the request being made.
* Administrators will have the ability to edit the aspect of the order forms, service descriptions, price and website directly.
* Customer should send a repair request, track his order and get the notification of the order in an efficient manner.

**2.2.6 Reliability Requirement**

System should provide a reliable environment to customers and owner. All orders must be reached at admin without any error.

**CHAPTER 3**

**SYSTEM ANALYSIS & DESIGN**

**3.1 Use Case Diagram**

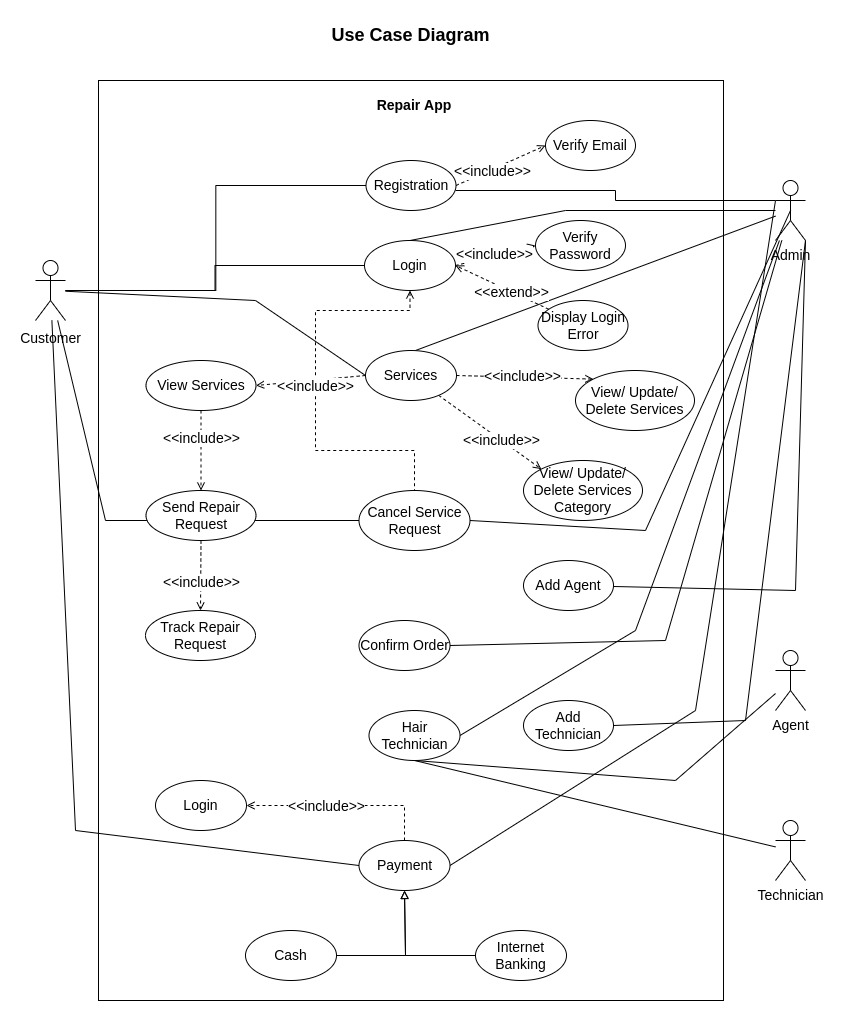


Figure 01: Use Case Diagram

**3.2 Use Case Diagram**

|  |  |
| --- | --- |
| **Use Case ID** | **UC1** |
| Name | Registration. |
| Description | This use case allows users to register into the system to access the relevant functions according to the user's role. The various user roles are customer and admin. To register to the system, all users have to enter their name, email and password. Then a verification email will be sent to the user email to verify their email address. They have to click the link provided in the email. After successfully verify their registration users can now login into the system. |
| Actor | * Admin * Customer |
| Pre-Condition | All the information Field must be filled up. |
| Post-Condition | Users can login into the system. |
| Flow | 1. Fill up the registration form with necessary information. 2. Press Sign Up button. 3. System will verify the given information 4. System will send a verification email to the user's email address 5. User will click the link that is provided in the email 6. System will verify the user account 7. User can now login into the system |
| Include | Verified email address. |

Table 02: Use Case Diagram (Registration)

|  |  |
| --- | --- |
| **Use Case ID** | **UC2** |
| Name | Login. |
| Description | This use case allows users to login into the system to access the relevant functions according to the user's role. The various user roles are customer and admin. Users can login into the system into two ways. By Email and password and by Google login. To login by email and password user have to verify their email address first. By providing the correct email and password or by google login function user can successfully login into the system. They will receive a JWT token for the authorization and redirect to the home page. |
| Actor | * Admin * Customer |
| Pre-Condition | All the information Field must be filled up. |
| Post Condition | Get access to the system. |
| Flow | 1. Go to Login page    1. Enter email address and password.    2. Google login 2. Press Login button. 3. System will verify the account. 4. System will send JWT token for authorization and redirect to the home page 5. Otherwise display Login error. |
| Include | Verify password. |

Table 03: Use Case Diagram (Login)

|  |  |
| --- | --- |
| **Use Case ID** | **UC3** |
| Name | Service. |
| Description | In this use case admin can add, delete and update any service. They can also add, delete and update any product brand and product model. Customer will browse all the available services. They can send select any services and send a repair request to the server. To send a repair request they have to fill up a form then select the product brand and a product model. After submitting the repair request admin will take farther action. |
| Actor | * Admin * Customer |
| Post Condition | Hire technician based on their product requirement. |
| Flow | 1. Go to service section. 2. Select product brand and product model 3. Fill up the request form 4. Send service request 5. Trank the repair request 6. Admin will receive this repair request 7. Admin can add, delete or update any service 8. Admin can add, delete or update product brand and product model |
| Include | * View Service * View, Update or Delete Services * View, Update or Delete Services brand and service model |

Table 04: Use Case Diagram (Service)

|  |  |
| --- | --- |
| **Use Case ID** | **UC4** |
| Name | Cancel Request. |
| Description | Customer is not interested anymore for the service or required service isn’t available on the system. |
| Actor | * Admin * Customer |
| Post Condition | Cancel Service Request in order to reinforce the user’s requirement. |
| Flow | * View order list. * Select the specific booking. * Give a reason and Press Cancel Request button. |
| Include | Login. |

Table 05: Use Case Diagram (Cancel Request)

|  |  |
| --- | --- |
| **Use Case ID** | **UC5** |
| Name | Confirm Order. |
| Description | Admin check all the necessaries and confirm the Order. |
| Actor | * Admin |
| Post Condition | Administrator agent allow Customer for the asking service. |
| Flow | * View request details from system database. * Verify request. * Confirm request. |
| Include | None |

Table 06: Use Case Diagram (Confirm Order)

|  |  |
| --- | --- |
| **Use Case ID** | **UC6** |
| Name | Hair Technician. |
| Description | Admin will select an agent that is close to the order request address. Then admin will assign a technician for the job. System will notify this state by a notification to the user app. |
| Actor | * Admin * Agent * Technician |
| Post Condition | Technician will arrive for the service. |
| Flow | * Check confirmation. * Select an agent * Select a technician close to the location * Notify the current stage by a notification to user |
| Include | None |

Table 07: Use Case Diagram (Hair Technician)

|  |  |
| --- | --- |
| **Use Case ID** | **UC7** |
| Name | Payment. |
| Description | After successfully repair the product, customer will select the payment option. They can make payment either cash on delivery or by internet banking. After a successful payment this process will be end. |
| Actor | * Admin * Customer |
| Pre-Condition | Customer Get the expected servicing solution. |
| Flow | * View service tracker to get an amount total. * Choose payment method. * For instant payment select cash. * For digital payment select internet banking. * Add this service to the History section. |
| Include | Login |

Table 08: Use Case Diagram (Payment)

|  |  |
| --- | --- |
| **Use Case ID** | **UC8** |
| Name | Add Agent. |
| Description | This use case allows admin to add, view and update an agent. To add an agent, they have to fill up a form with necessary information and then click the add button to save the agent. |
| Actor | * Admin |
| Pre-Condition | Fill up the form with necessary information |
| Flow | * Click the add button * Fill up the form * Submit the information. |
| Include | None |

Table 09: Use Case Diagram (Add Agent)

|  |  |
| --- | --- |
| **Use Case ID** | **UC9** |
| Name | Add Technician. |
| Description | This use case allows admin to add, view and update a technician. To add a technician, they have to fill up a form with necessary information and then click the add button to save the technician. |
| Actor | * Admin |
| Pre-Condition | Fill up the form with necessary information |
| Flow | * Click the add button * Fill up the form * Submit the information. |
| Include | None |

Table 10: Use Case Diagram (Add Technician)

**3.3 Activity Diagram**

An activity diagram is used to understand the flow of work that an object or component performs. It can also be used to visualize the interaction between different use cases.

**3.3.1 System Admin Activity Diagram**

Admin manage system content by creating, updating or deleting content from system database as well manage customers, orders, bookings and payments in the system.

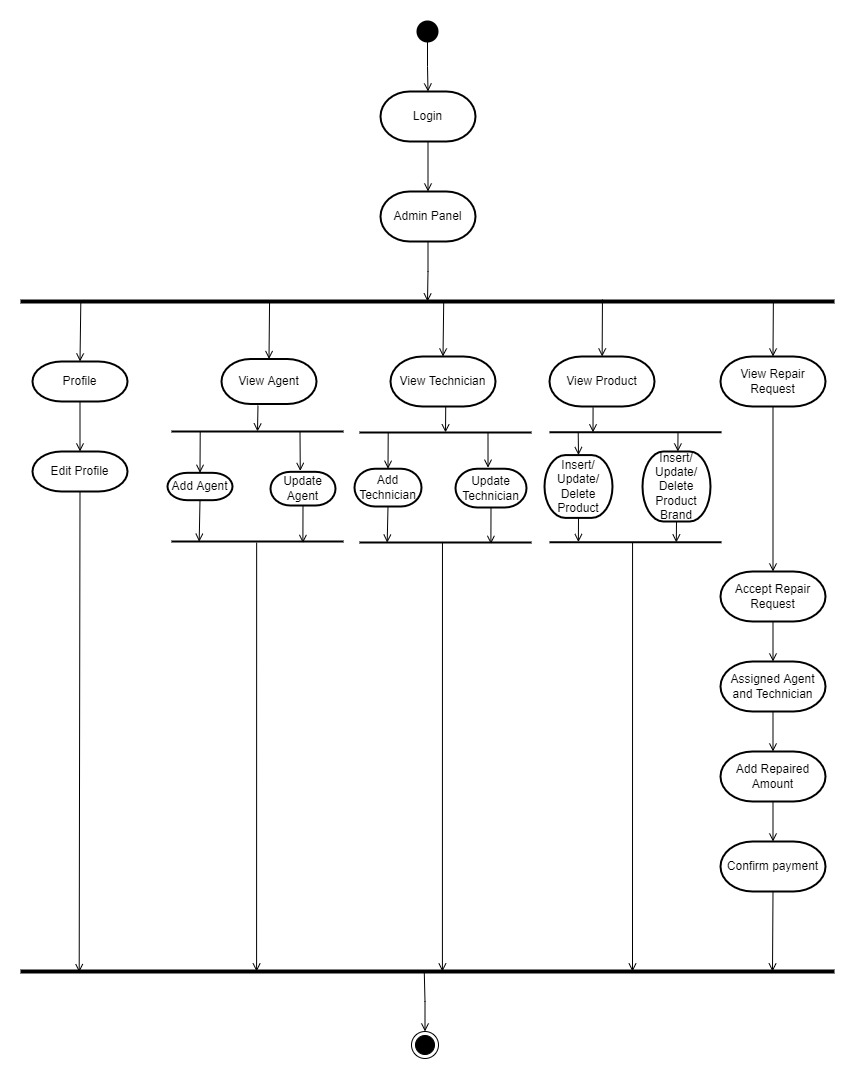


Figure 02: Admin Activity Diagram

**3.3.1 Customer Activity Diagram**

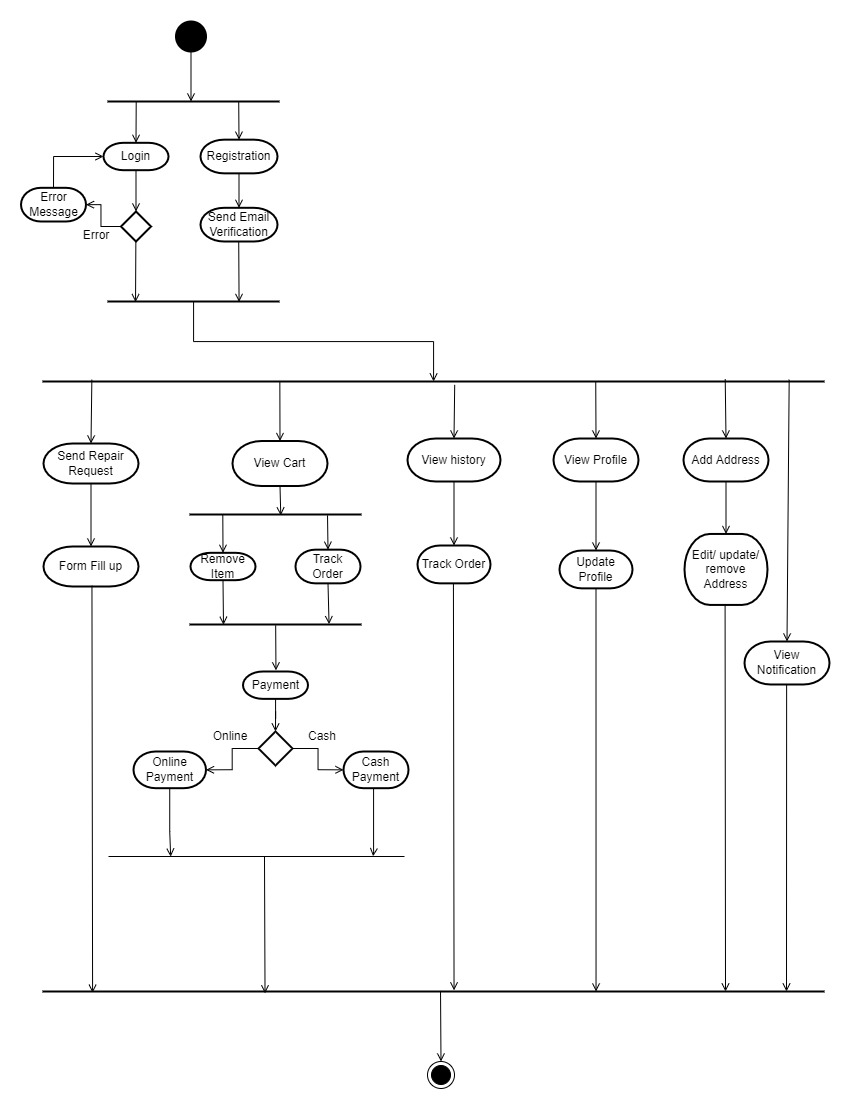


Figure 03: Customer Activity Diagram

**3.4 Sequence Diagram**

A Sequence diagram shows the sequence of messages exchanged by the set of objects performing a certain task.

**3.4.1 Admin Registration**

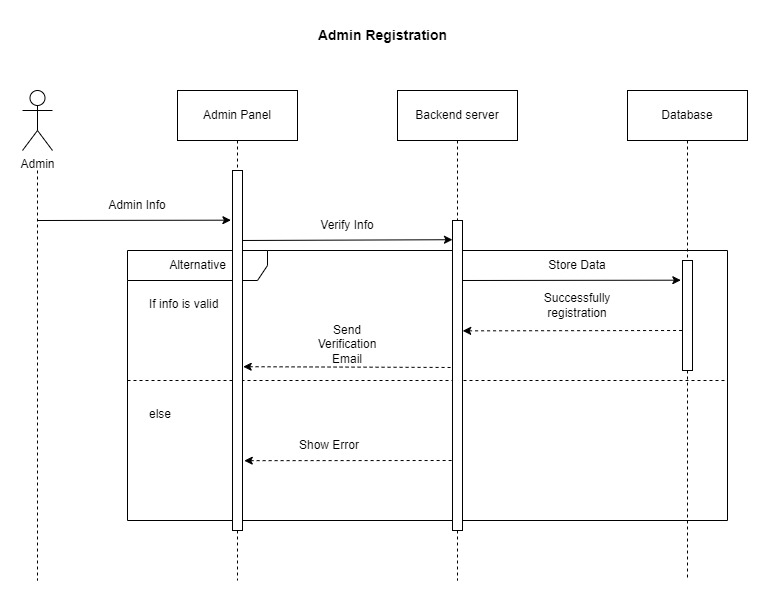


Figure 04: Admin Registration Sequence Diagram

**3.4.2 Admin Login**

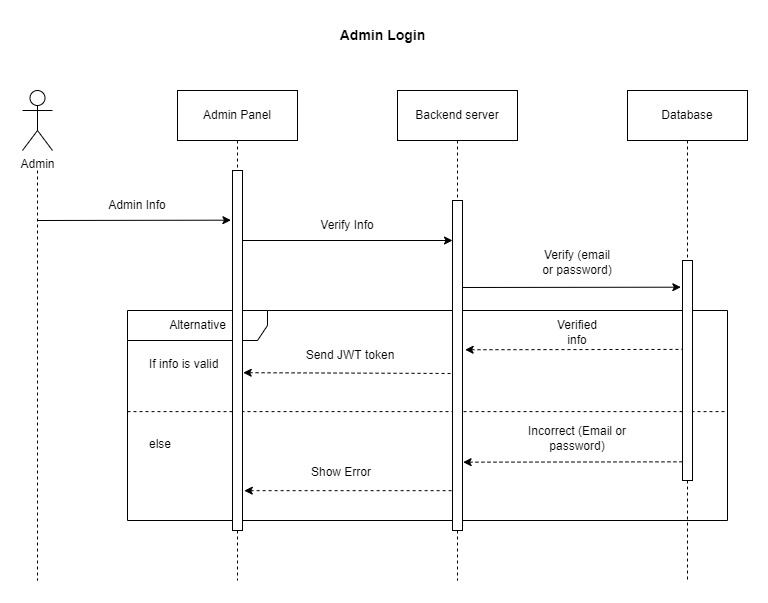


Figure 05: Admin Login Sequence Diagram

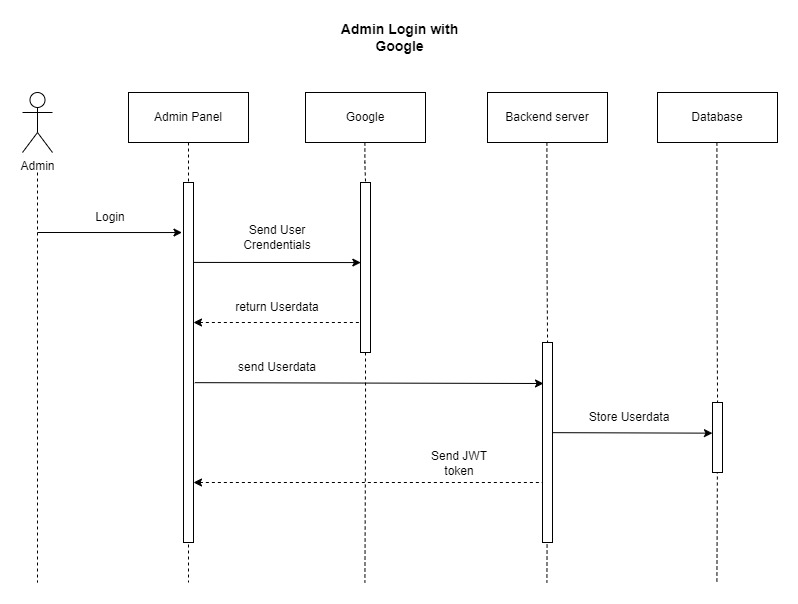


Figure 06: Admin Login with Google Sequence Diagram

**3.4.3 Admin Profile**

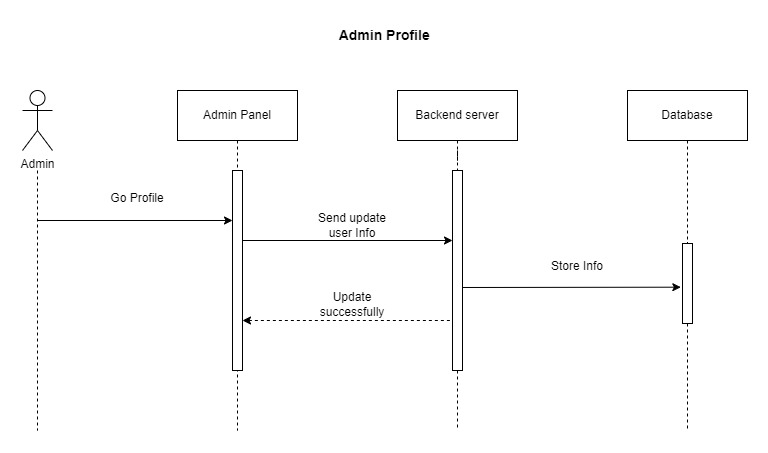


Figure 07: Admin Profile Sequence Diagram

**3.4.4 Admin Forget Password**

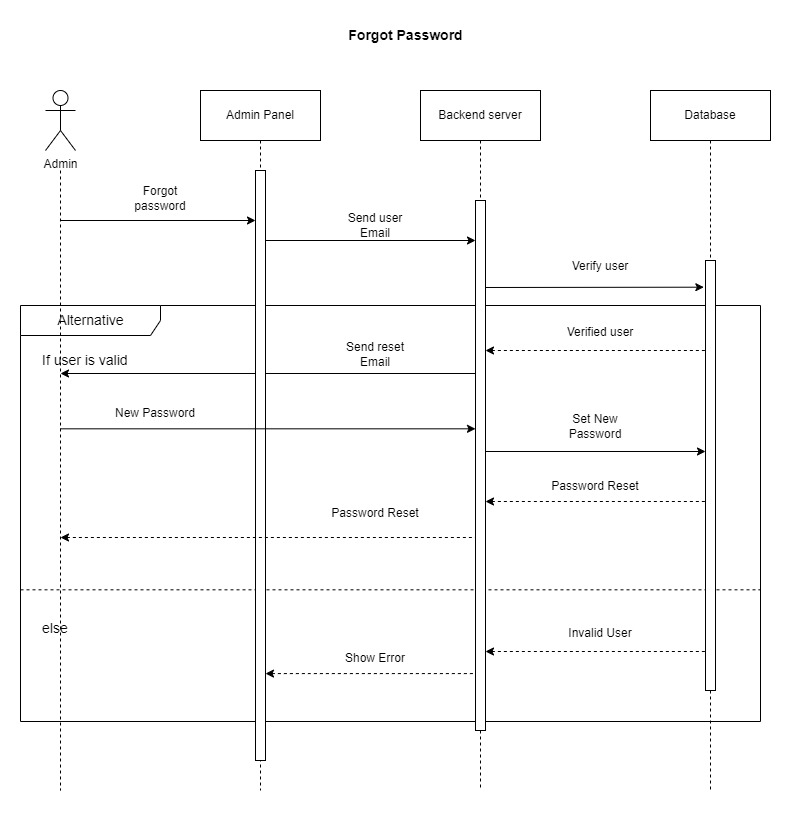
****

Figure 08: Admin Forget Password Sequence Diagram

**3.4.5 Admin View Order**

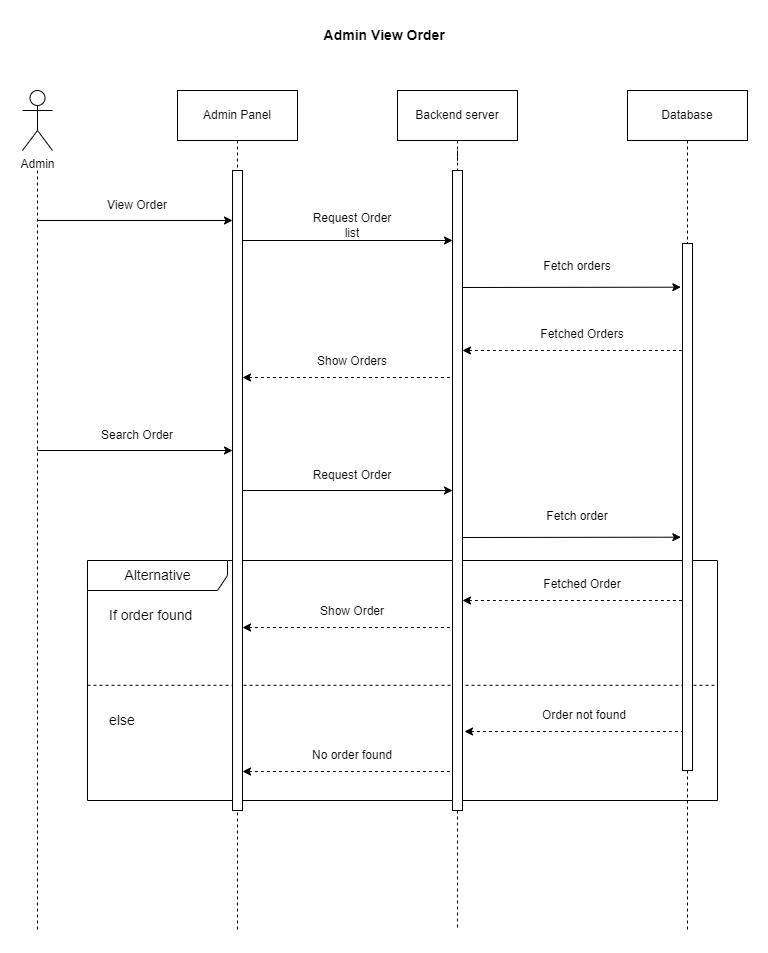
****

Figure 09: Admin View Order Sequence Diagram

**3.4.6 Admin Add Agent**

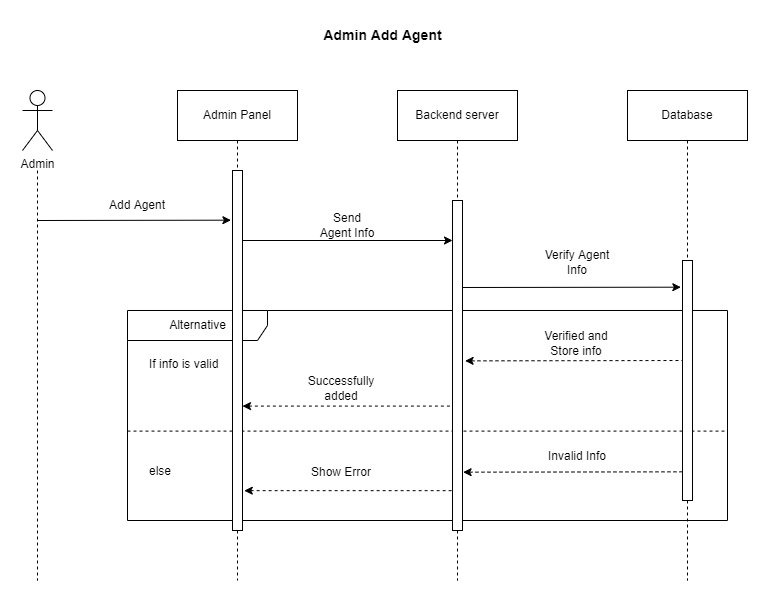


Figure 10: Admin Add Agent Sequence Diagram

**3.4.7 Admin Add Technician**

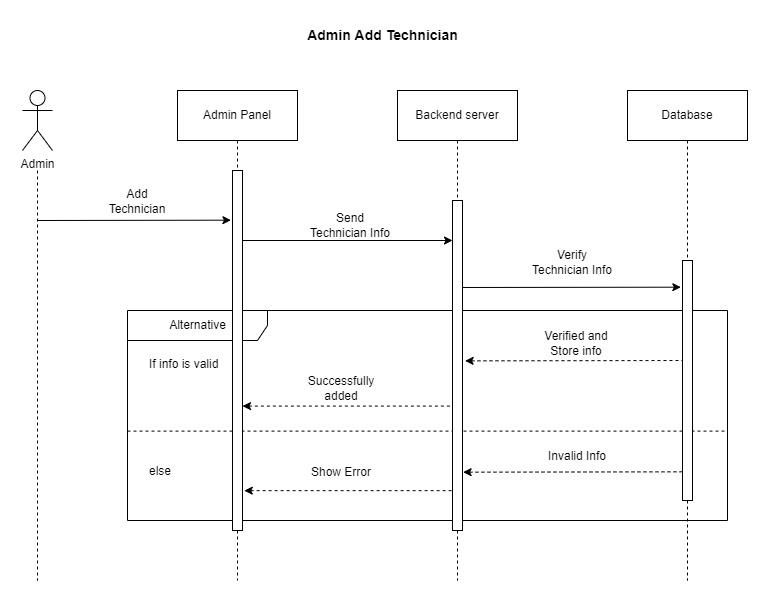
****

Figure 11: Admin Add Technician Sequence Diagram

**3.4.8 Admin Update Order**

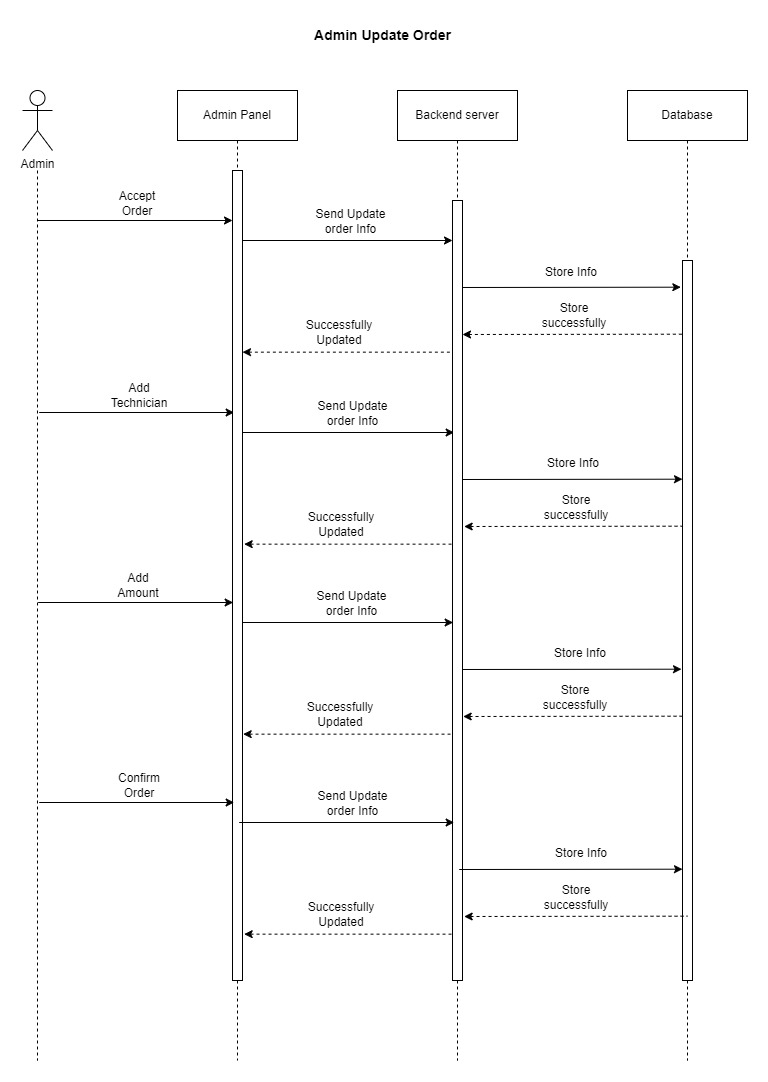


Figure 12: Admin Update Order Sequence Diagram

**3.4.9 Admin Update Agent**

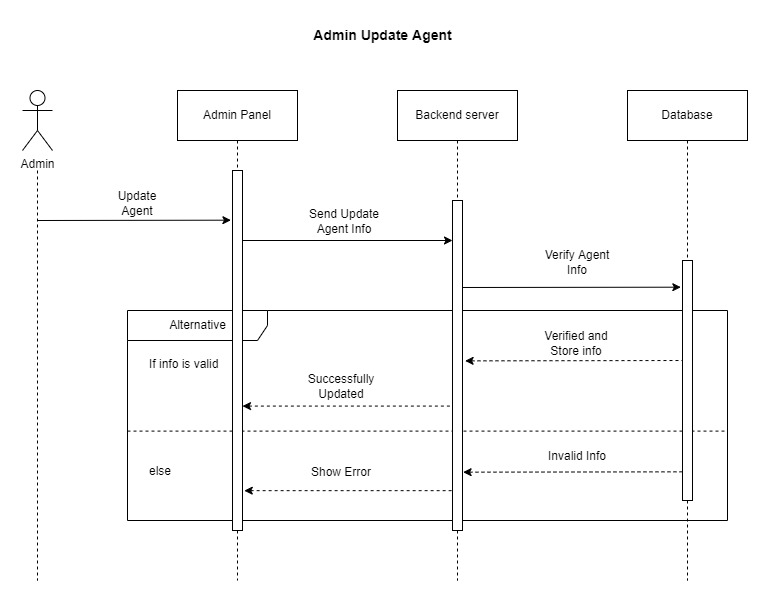


Figure 13: Admin Update Agent Sequence Diagram

**3.4.10 Admin Update Technician**

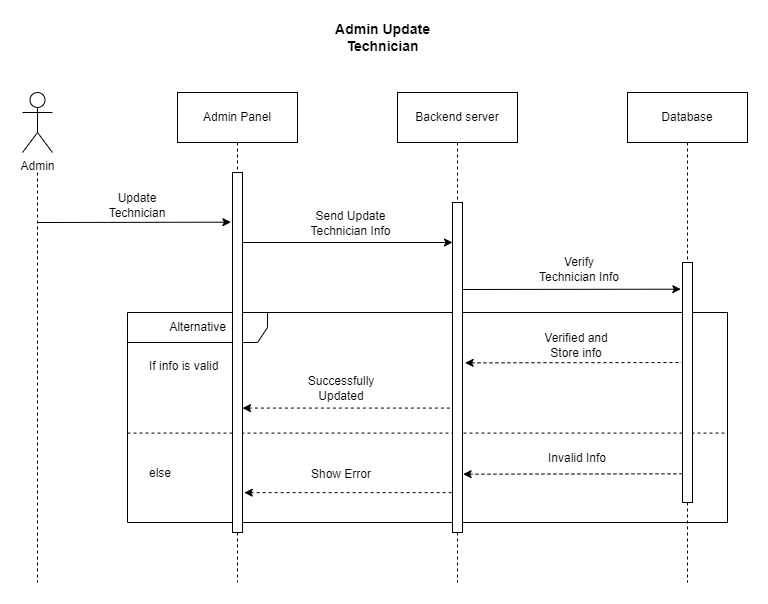
****

Figure 14: Admin Update Technician Sequence Diagram

**3.4.11 Customer Registration**

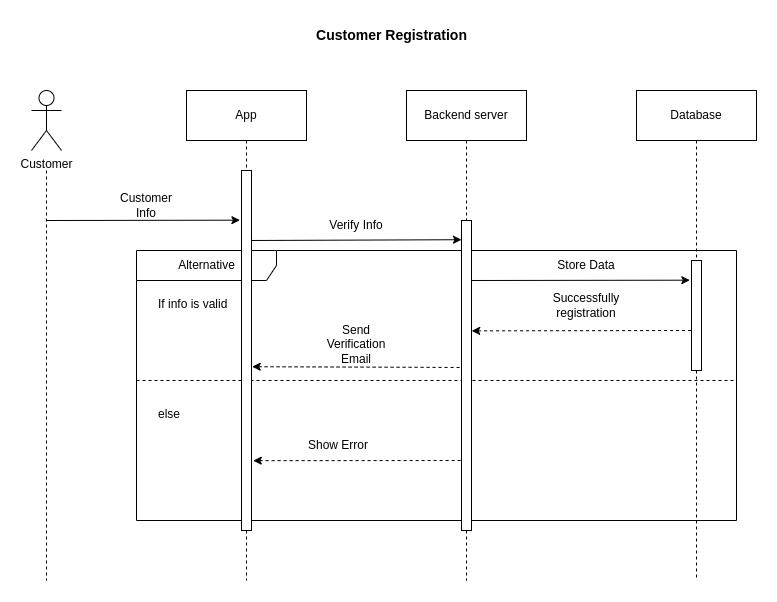


Figure 15: Customer Registration Sequence Diagram

**3.4.12 Customer Login**

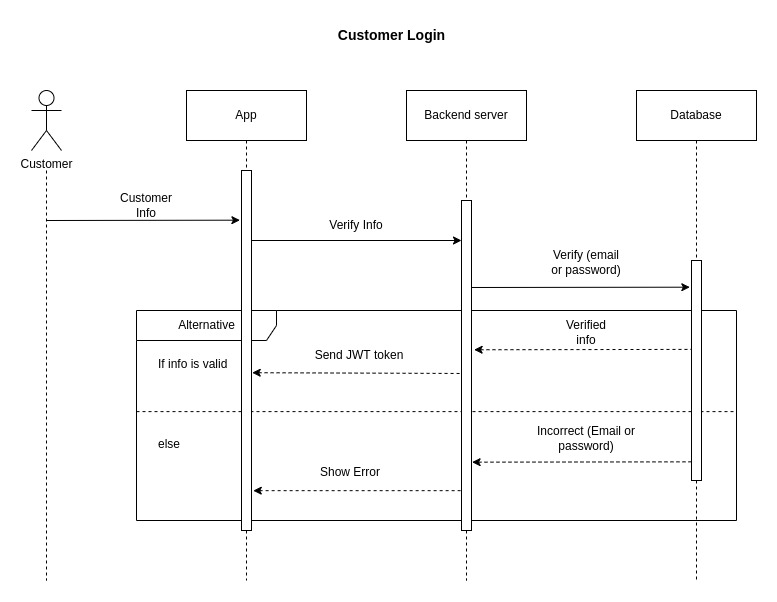
****

Figure 16: Customer Login Sequence Diagram

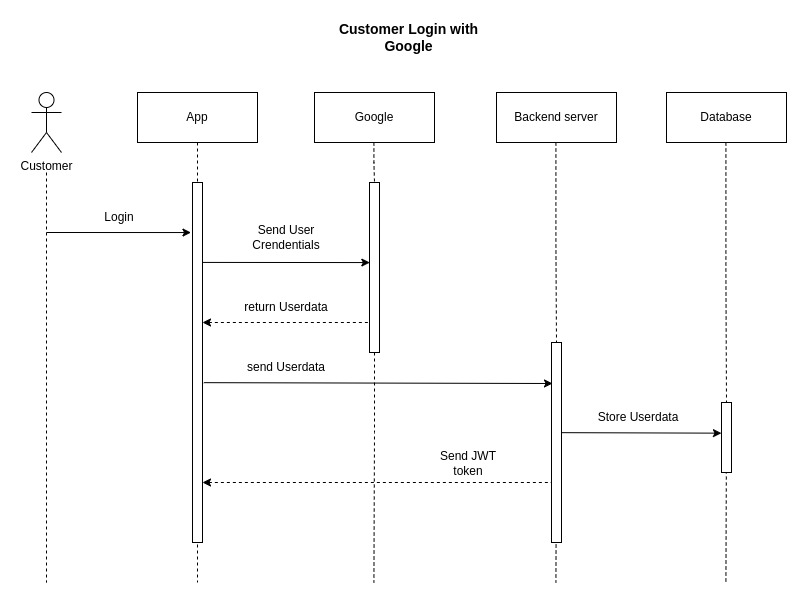
****

Figure 17: Customer Login with Google Sequence Diagram

**3.4.13 Customer Profile**

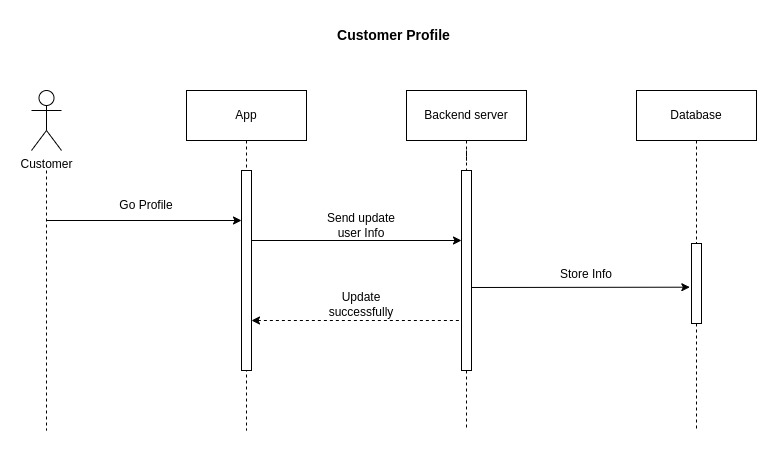
****

Figure 18: Customer Profile Sequence Diagram

**3.4.14 Forget Password**

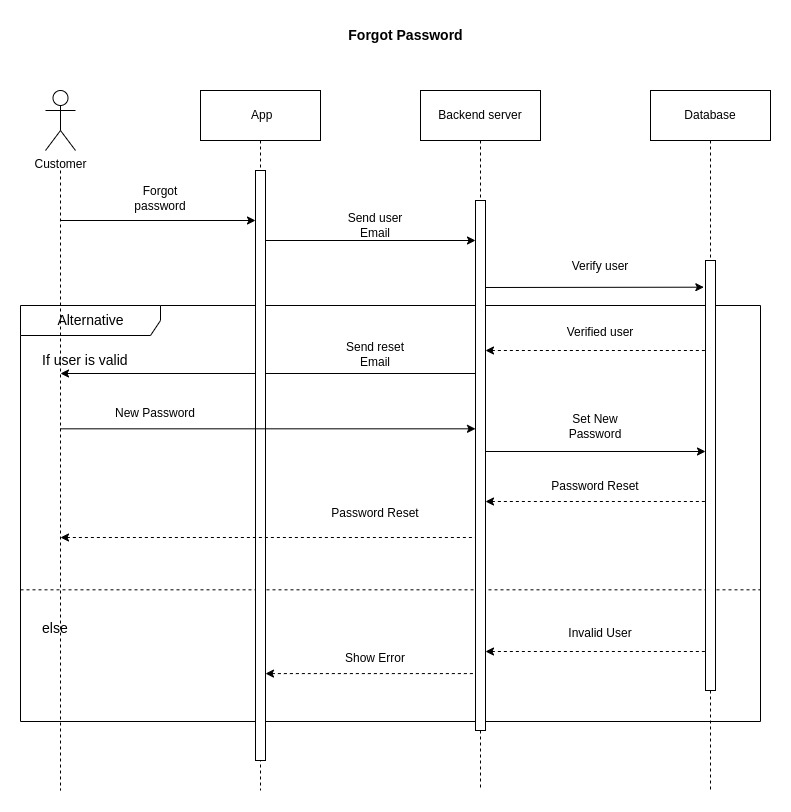
****

Figure 19: Customer Forget Password Sequence Diagram

**3.4.15 Customer Add Order**

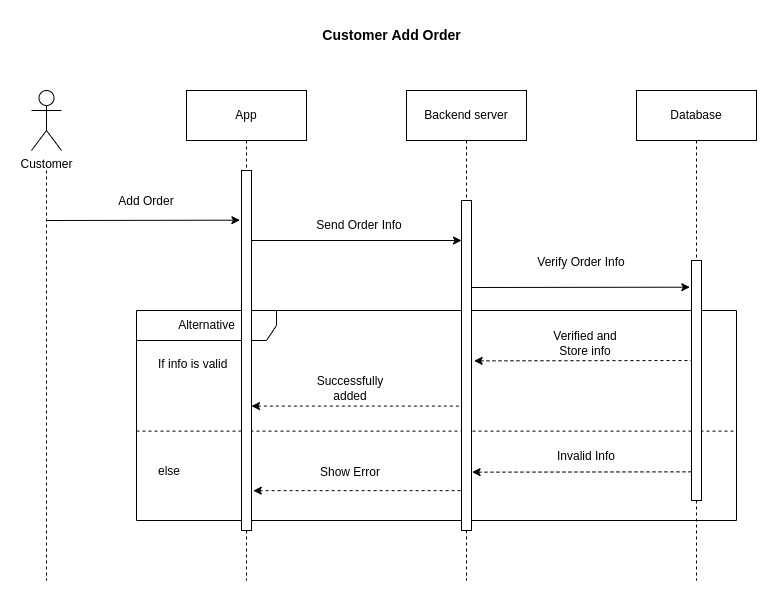
****

Figure 20: Customer Add Order Sequence Diagram

**3.4.16 Customer Add Address**

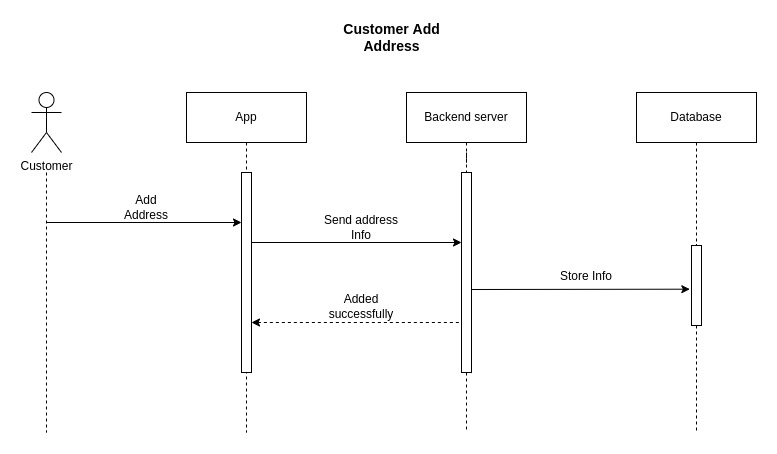
****

Figure 21: Customer Add Order Sequence Diagram

**3.4.17 Customer Update Address**

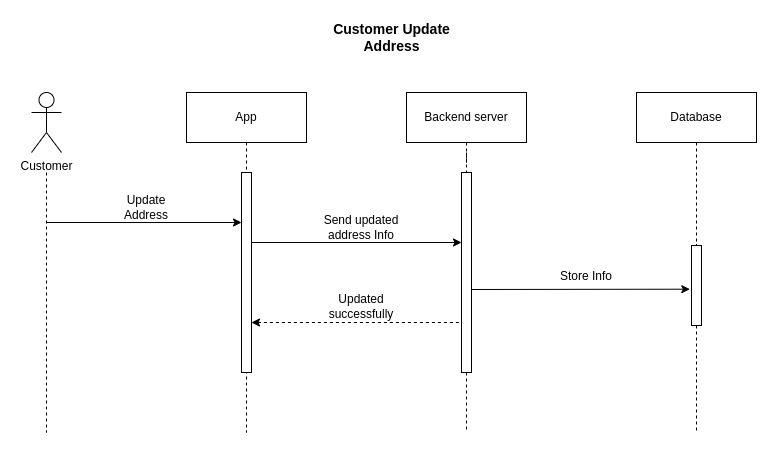
****

Figure 22: Customer Update Address Sequence Diagram

**3.4.18 Cart List**

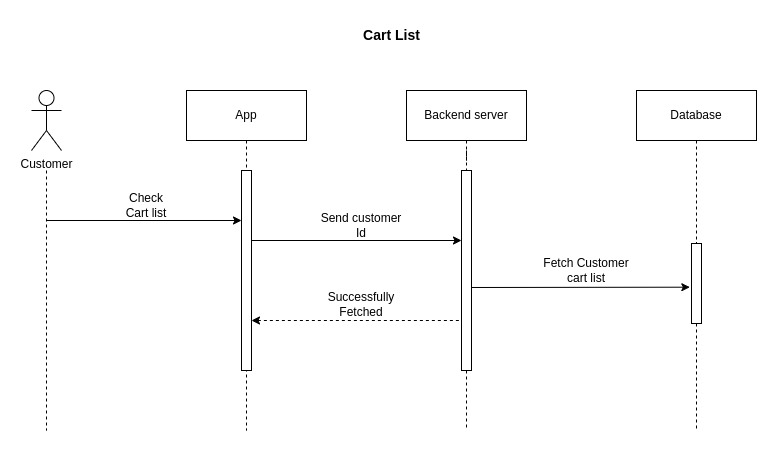
****

Figure 23: Cart List Sequence Diagram

**3.4.19 Track Order**

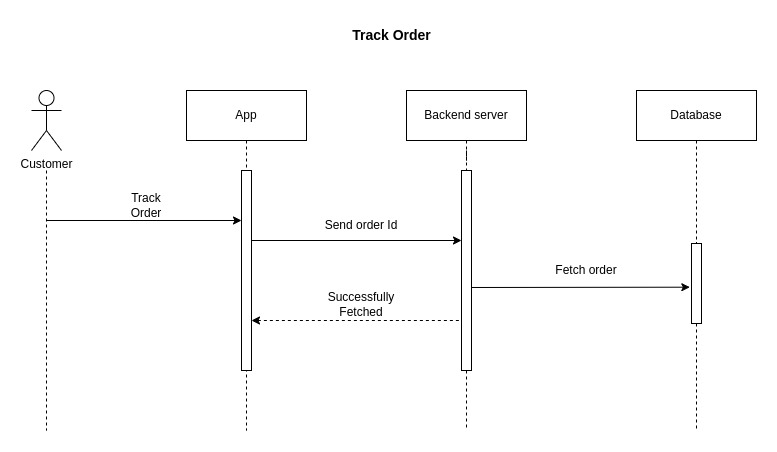
****

Figure 24: Track Order Sequence Diagram

**3.4.20 Order History**

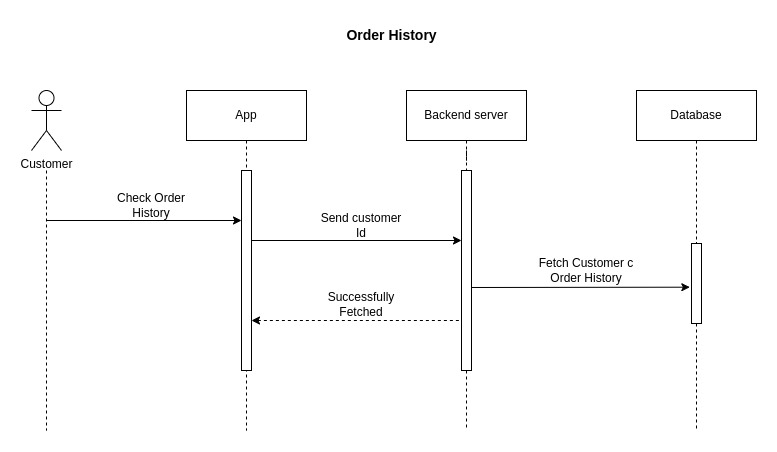
****

Figure 25: Order History Sequence Diagram

**3.4.21 Payment**

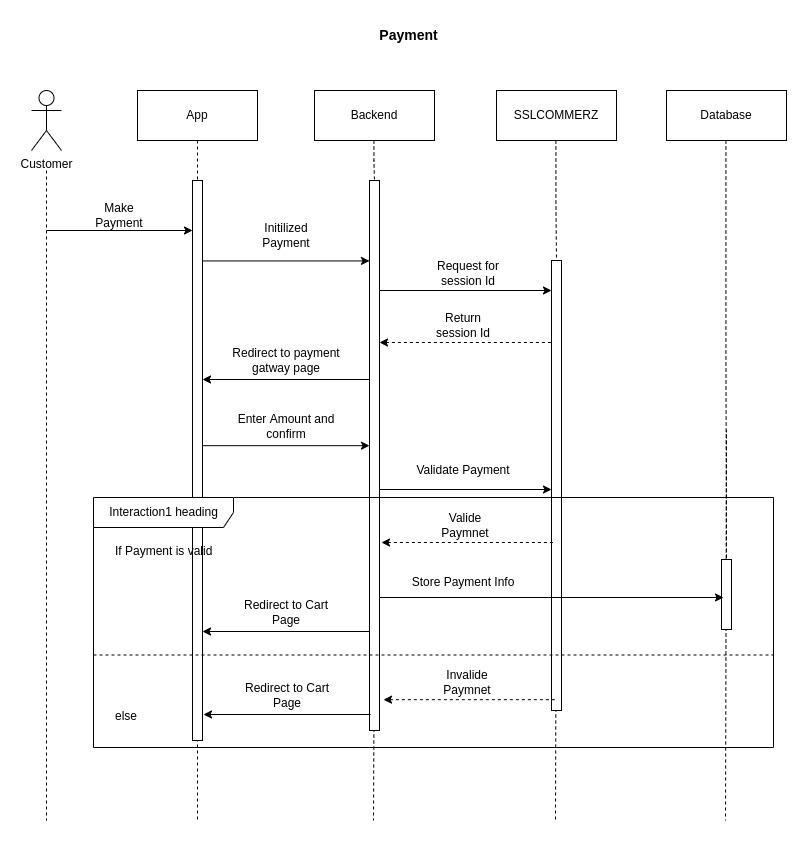
****

Figure 26: Payment Sequence Diagram

**Reference**: Font-10

All references to books, papers, and other publications must be fully and correctly quoted. There are several methods of quoting references. One is to state the name of the author and a serial number in the main text with the full details of the reference in the Reference section of the report, for example:

In the text:

*....The analysis of the algorithms has been extensively reviewed by Yorozu et al. [1]*

*and will ....*

In the References section:

[1] Y. Yorozu, M. Hirano, K. Oka, and Y. Tagawa, “Electron spectroscopy studies on magneto-optical media and plastic substrate interface,” IEEE Transl. J. Magn. Japan, vol. 2, pp. 740–741, August 1987.

***Conference/Journal Papers:***

[1] Author1, Author2, and Author3, “Paper Title”, Conference/Journal, Volume, page number, Month and year.

**Example:**

[1] Y. Yorozu, M. Hirano, K. Oka, and Y. Tagawa, “Electron spectroscopy studies on magneto-optical media and plastic substrate interface,” IEEE Transl. J. Magn. Japan, vol. 2, pp. 740–741, August 1987.

***Books:***

[2] Author, Book Title, Edition/Volume, Publisher, Year, Page number

**Example:**

[2] T. H. Cormen, C. E. Leiserson, R. L. Rivest, C. Stein, Introduction to Algorithms, 3rd Edition, The MIT Press, 2009, pp. 120-122.

***Websites:***

[3] Name/Title of the Website, available at << https://URL>>, last accessed on Date at Time.

**Example:**

[3] Learn about Wikipedia, available at << http://www.wikipedia.org/>>, last accessed on 06-06-2019 at 12:00 PM.

Thank You!

Best of Luck