**Batoor Delivery Services**

**Problem Statement:**

**Title:**

Automating the manual ordering system of Batoor Delivery Services, Swabi and making it easy-to-use for everyone through internet.

**Description:**

Currently, Batoor Delivery Services have a manual system of taking orders from their customers. Customers call them on their mobile number and place their order. They note all their details on paper and make a record of it. The customers don’t have any visual of the restaurants and food they offer.

If the system gets digitalized and have a website. The user will have a visual of restaurants listing and the meals they offer. They will be able to select anything of their choice. They will also be able to make on-demand orders online without the process of making a phone call. The paper-based records will also be replaced with the data of each order stored in the database.

**Software Requirements Specification**

**Batoor Delivery Services**

**[Version 1.0]**

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| <28/06/20> | <1.0> | SRS 1.0 | Group-1 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Table of Contents**

1. Introduction
   1. Purpose
   2. Scope
   3. Intended Audience
   4. Team Members
   5. References
2. The Overall Description
   1. Product Perspective
   2. User Characteristics
   3. Constraints
   4. Assumptions and Dependencies
3. External interface Requirements
   1. User Interfaces
   2. Hardware Interfaces
   3. Software Interfaces
   4. Communications Interfaces

1. System Features

4.1. System Features

4.2. Functional Requirements

1. Other Non-Functional Requirements

5.1 Performance Requirements

5.2 Software System Attributes

5.3 Security Requirements

5.4 Software Quality Attributes

1. **Introduction**

The web application is to be made for Batoor Delivery Services. This will be used by their customers to place on-deman d orders and do online shopping from the registered restaurants. The admin panel will get a display of the orders placed and they can add/edit items & restaurants.

**1.1 Purpose**

This software requirements specification defines external interface, performance, and software system attributes requirements of **Batoor Delivery Services**. This document is intended for the following group of people:

* Developers for the purpose of maintenance and new releases of the software.
* Management of the company
* Documentation writers
* Testers
  1. **Scope**

This document applies to **Batoor Delivery Services** software. This software facilitates the customers of the company to place online on-demand orders and do online shopping from their registered restaurants. The admin will have a display of two portions: orders to be approved and orders approved. They can add/edit items and restaurants as well.

The software has two panels of administration and users. They can access it by providing login credentials. The output then comprises of an interactive display that let the users select the desirable function that he wants to perform.

The software is expected to be completed in 3 months approximately.

* 1. **Intended Audience**

The intended audience consists of customers of Batoor Delivery Services, Swabi.

* 1. **SRS Team Members**

The SRS team members consists of:

* Saifullah Amin ([saifullahaminkhan@gmail.com](mailto:saifullahaminkhan@gmail.com))
* Shahid Ali ([alishahidcr7@gmail.com](mailto:alishahidcr7@gmail.com))
* Muhammad Junaid ([Junaid\_355@hotmail.co.uk](mailto:Junaid_355@hotmail.co.uk))
* Maazullah Kakar ([maazkakar77@gmail.com](mailto:maazkakar77@gmail.com))
  1. **References**
* www.google.com
* [www.wikipedia.com](http://www.wikipedia.com)
* IEEE Software Requirements Specification Std. 830-1993
* Batoor Delivery Services, Swabi
* www.w3schools.com

1. **Overall Description**

**2.1 Product Perspective**

* Batoor is a delivery service provider based in Swabi.
* This software allows customer of the company to place orders online
* The customer can also do online shopping from the registered restaurants.
* The customer will have option to login / create account.
* The admin of the web app will have a display of to be approved and approved orders.
* The admin can add/edit items and restaurants.
* When the order is approved by the admin the user will receive a message of confirmation on their numbers.
* This software is a web application consisting of two panels.
* Data is stored in a database.
* It must be used online through internet.
  1. **User Class and Characteristics**

There are different users that will interact with the system. They are listed below:

* **Admin:** The admin of the web app will have a display of order to approve and approved orders. He can approve / not approve the orders. The approved orders will be stored, and the admin can access it any time. The admin will have an option to add/edit items and restaurants. These functionalities will be displayed categorically with a good interface.
* **User / Customer:** The user / customer will have the option to make on-demand order online or do online shopping from selected restaurants. The user will have the option to login / create account. These functionalities will be displayed categorically with a good interface.
  1. **Constraints**

The major constraints that the project has are as follows:

* The user can access only those functionalities that he is allowed to.
* There is no facility of online payment for now, which can be added in the future.
* The user can only do online shopping from the registered restaurants.
* The application can be accessed only through the internet.

**3. External Interface Requirements**

**3.1 User Interfaces**

The user interface for the software shall be compatible to any browser such as Internet Explorer, Mozilla or Netscape Navigator by which user can access to the system. The interface provided to the user should be a very user-friendly one and it should provide an optional interactive help for each of the service listed. The interface provided is a menu driven one and the following screens will be provided:

1. A simple landing page is displayed at the start, which will have some details about the company. It will have options to login as admin / customer.
2. By selecting the login button, login screen is provided for entering the required username/password for the specific panel.
3. An unsuccessful login leads to a reattempt screen for again entering the same information. The successful login leads to a screen displaying a list of available functionalities.
4. In case of admin, the admin is provided with to be approved and approved orders through an interface. An option of add/edit items and restaurants will also be provided.
5. In case of user / customer, the user is provided with options of on-demand delivery and online shopping from restaurants.

**3.2 Hardware Interfaces**

* Since the application must run over the internet, all the hardware shall require connecting internet will be hardware interface for the system. As for e.g. Modem, WAN – LAN, Ethernet Cross-Cable.

**3.3 Software Interface Requirements**

* To save different records and data this software will have to use MongoDB.

**3.4 Communication Interface Requirements**

* The communication protocol used shall be TCP/IP.
* Protocol used for data transfer shall be File Transfer Protocol (FTP).

**4. System Features / Functional Requirements**

* **Orders Display**
  + The software shall display to-be-approved and approved to the admin
  + The software shall allow the admin panel to send confirmation messages to the user for their orders with delivery charges.
  + The software shall allow the admin to see the records of all approved orders.
  + The agent should have an option of submitting delivery charges prior to approving online shopping order.
  + The agent should have option of submitting price & delivery charges for on-demand delivery after approval
* **Add/Edit Items and Restaurants**
  + The software shall allow the admin to add/edit items for online order.
  + The software shall allow the admin to add/edit restaurants.
* **Login / Create Account** 
  + The software shall allow the admin to login with the credentials.
  + The software shall allow the customer to login to the website.
  + The software shall allow the customer to create account for the website.
* **On-Demand Delivery**
  + The software shall allow the customer to place on-demand delivery orders.
  + The order will require name, contact, and address of pickup and drop-off locations.
  + The software shall allow the customer to state the product the customer wants to order.
* **Online Shopping (Restaurants)**
  + The software shall allow the customer to do online shopping from the selected restaurants.
  + The software will have a display of all the restaurants and the food offered by them. The user will select the desired meal and add it to the cart. The user at last, will checkout from the cart and make the final order.
* **Search Box**
  + The search box will allow the user to search any of their desired item.
* **Shopping Cart**
  + Shopping Cart will help user to buy multiple items in a single go. User just have to navigate to different products and press Add to Card button on each Product Page.
* **Feedback**
  + This will allow the user to give his reviews or suggestions about the product and services provide by the System.
* **Technical Issues**
  + This system will work on client-server architecture. It will require an internet server, and which will be able to run application. The system should support some commonly used browser such as IE or Google Chrome etc.
* **Validity Checks** 
  + In order to gain access to the system, the user is required to enter his/her correct username and password. Also, if the user is an administrator, he is required to enter his login id in order to access and change the facilities provided by the system.

**5. Nonfunctional Requirements**

**5.1 Performance Requirements**

The following list provides a brief summary of the performance requirements for this software

* + - * **Capacity:** The software should be available to the audiencefor 24 hours service. The software should be able to operate on all major web-browsers with all of its fundamental functions. It should not slow-down the system. Even at peak hours without affecting the quality of service of the system
      * **Dynamic requirements:**
        + The website opening load should not exceed from 2 to 3 sec in normal and under the heavy load not more than 5 to 6 sec.
        + The login credential verification time must not exceed 1 sec. under normal server workload and 3 sec. under peak server workload.
        + Creating user account should not exceed 1 sec in normal workload and 2 sec in heavy workload.
        + Sending approval SMS from admin to user must not exceed 1 sec.
* **Quality:** The goal is to produce quality software. As the quality of a software is difficult to measure quantitatively, the following ways or guidelines will be used when judging the quality of the software:

**1.** **Consistency** – All code will be consistent with respect to the style (this is implied when adhering to the standard).

**2. Test Cases** – All functionality will be tested thoroughly.

**5.2 Software System Attributes**

* **Availability:** The software will be available 24 hours a day and 7 day per week for its users having internet connection. In case of any problem accessing the application, the support team will solve the problem as soon as possible.
* **Maintainability**: The system should be maintainable, any changes required should be applicable on the system at any time without any increase in errors.

**5.3 Security Requirements:**

The system should provide a secure login to the users by using advanced secure login algorithms and provide access only to the authorized users as security is the key requirement of this system. The password shall be 6-14 characters long. Passwords can contain digit, hyphen and underscore.

The system’s back-end servers shall never display a customer’s password. The customer’s password may be reset but never shown. The system’s back-end servers shall only be accessible to authenticated administrators. The system’s back-end databases shall be encrypted.

**5.4 Software Quality Attributes:**

System must be:

* Consistent in performance
* Safe and Secure
* Robust
* Scalable
* Flexible
* User friendly
* Efficient
* Inter-operable
* Upgradable

**Domain Model**

**A close up of text on a black background

Description automatically generated**

**Class Model**

**A close up of text on a white background

Description automatically generated**

**Entity Relationship Diagram (ERD)**

**A close up of text on a white background

Description automatically generated**

**Use Case Model**

**A picture containing map, drawing

Description automatically generated**

**Use Cases**

**UC-1**

**Use Case :** Log In

**Actors :** Customer, Admin

**Purpose :** Actors login to their account.

**Overview :** Actors loads the website, click on their login button and enter their login credentials. Upon authentication actor is taken to their desired window.

**Type :** Primary and essential

**Typical Course Of Events**

**Section:** Main

**Actor Action System Response**

1. This use case begins when the customer loads the website for login.
2. The actor clicks on their login button.

1. When the actor is:
   1. Admin: see section ‘Admin Login’
   2. Customer: see section ‘Customer Login’
2. The system loads the login page for the actor

**Section:** Admin Login

1. The admin provides his login credentials2. Upon authentication, the admin is loaded the admin window.

**Section:** Customer Login

1. The customer provides his login credentials2. Upon authentication, the customer is loaded the customer window.

**Alternative Courses**

Line 2 (Admin Login, Customer Login) : If account does not exist it will throw an error message.

Line 2 (Admin Login, Customer Login) : If actor enter wrong username or password it will also give an error message.

**UC-2**

**Use Case :** Register

**Actors :** Customer

**Purpose :** Customer register their account.

**Overview :** The customer clicks on the register button. The user is asked to input some information about himself and password. Upon checking all the data, a user account is created.

**Type :** Primary and essential

**Typical Course Of Events**

**Actor Action System Response**

1. This use case begins when the customer clicks on create account button.
2. The customer provides information about himself and sets password
3. The system loads a window where user is asked some information.
4. The system checks the data, if valid a user account is created.

**Alternative Courses**

Line 4: The customer input data is not in the required format, it will show a message

Line 4: The customer password isn’t according the set guidelines, it show a message

**UC-3**

**Use Case :** Order Approval

**Actors :** Admin, Customer

**Purpose :** Customer gives an order for online shopping / on-demand delivery for the admin to approve it.

**Overview :** The customer gives an order for online shopping / on-demand delivery. The admin receives it approve / not approve it.

**Type :** Primary and essential

**Typical Course Of Events**

**Actor Action System Response**

1. This use case begins when the customer checks out from the cart or gives an order for on-demand delivery.
2. The admin receives the order and approve / disapprove it.
3. The system sends the order or approval to admin.
4. The system sends a message about approval / disapproval to the customer.

**Alternative Courses**

Line 4: Either the delivery riders are not available, or the item isn’t available with the shop / business the order is disapproved

**UC-4**

**Use Case :** Shopping delivery charges

**Actors :** Admin, Customer

**Purpose :** Admin provides the delivery charges for online shopping order.

**Overview :** The customer submit order for online shopping. Upon approval, the admin provides delivery charges for the order. Approval details with the delivery charges are send to the customer in the form of text.

**Type :** Primary and essential

**Typical Course Of Events**

**Actor Action System Response**

1. This use case begins when the customer checks out from the cart and the order detail is sent to the admin for approval.
2. The admin following the terms and conditions input delivery charges for the specific order.
3. The system sends the delivery charges with approval information to the customer in the form of text.

**UC-5**

**Use Case :** On-demand order price & delivery charges.

**Actors :** Admin, Customer

**Purpose :** Admin provides the price & delivery charges for on-demand order.

**Overview :** The customer submit order for on-demand delivery. Upon approval, the admin the order is picked form the shop/business. After knowing about the price the admin input the price and delivery charges (according to company guidelines) into the website. Approval details with the delivery charges are send to the customer in the form of text.

**Type :** Primary and essential

**Typical Course Of Events**

**Actor Action System Response**

1. This use case begins when the customer submit order for on-demand delivery and the order detail is sent to the admin for approval.
2. The admin after getting information about price from the shop/business, inputs price and delivery charges (according to company guidelines) into the website,
3. The system sends the price and delivery charges with approval information to the customer in the form of text.

**UC-6: Browse**

**Use Case:** Browse

**Actors:** Customer

**Purpose:** To see and select different categories of items

**Overview:** When the customer is logged in the customer will be able to see items on the list and will add its desired items into cart for later process.

**Type:** Primary, Essential

**Typical Course of Events:**

**Actor Actions: System Response:**

1. The use case begins when the customer 4. The system will show a logged on to the websites to view a list of items &

items and the list of restaurants. restaurants.

2. The customer can select and see items

3. The customer will be able to see

the list of restaurants and can order from it.

**UC-7:**

**Use Case:** Add to cart

**Actors:** Customer

**Purpose:** Select items and add it to cart for delivery

**Overview:** The customer will choose an item and add it to the cart for delivery.

**Type:** Secondary

**Typical Course of Events:**

**Actor Actions: System Response**

1. This use case begins when the customer 4. The system will add the

selects the items and what to receive it. selected item to the cart.

1. The customer will be given the

choice to add items to the cart

3. When the customer choses or selects an

item they can then add it to the cart for delivery.

**UC-8:**

**Use Case:** Checkout

**Actors:** Customer

**Purpose:** Will exit from cart and confirm the order

**Overview:** After the choosing and adding the item to the cart the customer will exit the cart and wait for its order to complete

**Type:** Primary, Essential

**Typical Course of Events:**

**Actor Actions: System Response:**

1. The use case begins

when the customer exits the cart 3. The system will send message

to the admin.

1. After ordering the item the

customer will checkout

from the system.

**UC-9:**

**Use Case:** On-demand delivery

**Actors:** Customer

**Purpose:** To place order for an on-demand delivery

**Overview:** The customer will select an item from a list and place an on-demand delivery then the system will issue a delivery request for the admin and the admin will approve or disapprove the request.

**Type:** Primary, Essential

**Typical course of events:**

**Actor Actions:**

1. The use case begins when the customer selects an item and want a delivery
2. When the customer selects an item and add to the cart after the checkout the price will be generated
3. After the check out the customer will request for on demand delivery
4. During the request, the place and the time will be set for the delivery and will be approved by admin

**System Response**

1. The system will add the selected item to the cart and generate price
2. The system will save the given info and send it to the admin

**UC-10**

**Use Case :** Add Item

**Actors :** Admin

**Purpose :** Admin adds an item to the software for online shopping.

**Overview :** The admin after getting access through login credentials to his panel will select the option for add item. Some information would be provided by him. After that if the item doesn’t already exists in the database, an item would be added.

**Type :** Primary and essential

**Typical Course Of Events**

**Actor Action System Response**

1. This use case begins when the admin logs in to his portal and selects add item option.
2. The admin gives the relevant information submits it.
3. The system asks for relevant information.
4. The system after authentication adds it to the database

**Alternative Courses (UC-10):**

**Line 3:** If the item with the same info is already there in the database associated with the same database it will throw an error.

**UC-11**

**Use Case :** Edit Item

**Actors :** Admin

**Purpose :** Admin edits an item.

**Overview :** The admin after getting access through login credentials to his panel will select the option for edit item. A window will be provided of the things that can be changed and the admin will edit it.

**Type :** Primary and essential

**Typical Course Of Events**

**Actor Action System Response**

1. This use case begins when the admin logs in to his portal and selects edit item option.
2. The admin edit the relevant information submits it.
3. The system shows all the information that can be edited.
4. The system after authentication edits it in the database

**UC-12**

**Use Case :** Add Restaurant

**Actors :** Admin

**Purpose :** Admin adds a restaurant to the software for online shopping.

**Overview :** The admin after getting access through login credentials to his panel will select the option for add restaurant. Some information would be provided by him. After that if the restaurant doesn’t already exists in the database, it would be added.

**Type :** Primary and essential

**Typical Course Of Events**

**Actor Action System Response**

1. This use case begins when the admin logs in to his portal and selects add restaurant option.
2. The admin gives the relevant information submits it.
3. The system asks for relevant information.
4. The system after authentication adds it to the database

**Alternative Courses:**

**Line 3:** If a restaurant with the same info is already there in the database it will throw an error.

**UC-13**

**Use Case :** Edit restaurant

**Actors :** Admin

**Purpose :** Admin edits a restaurant’s details.

**Overview :** The admin after getting access through login credentials to his panel will select the option for edit restaurant. Editable information would be showed to him for editing.

**Type :** Primary and essential

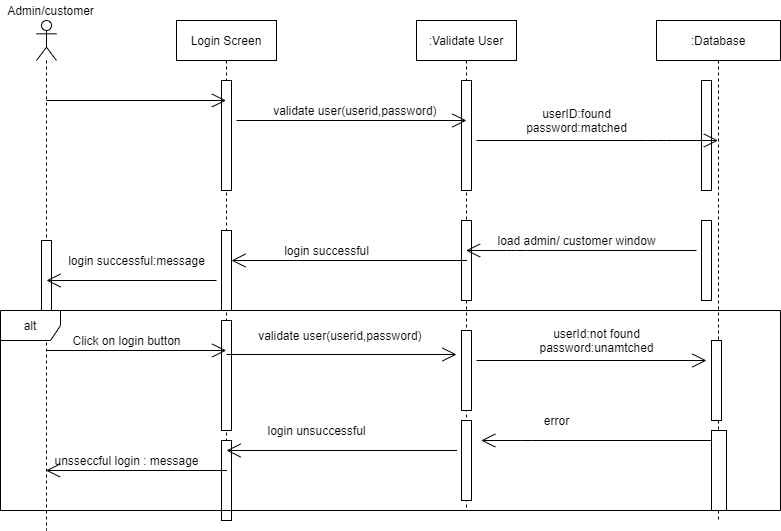
**Typical Course Of Events**

**Actor Action System Response**

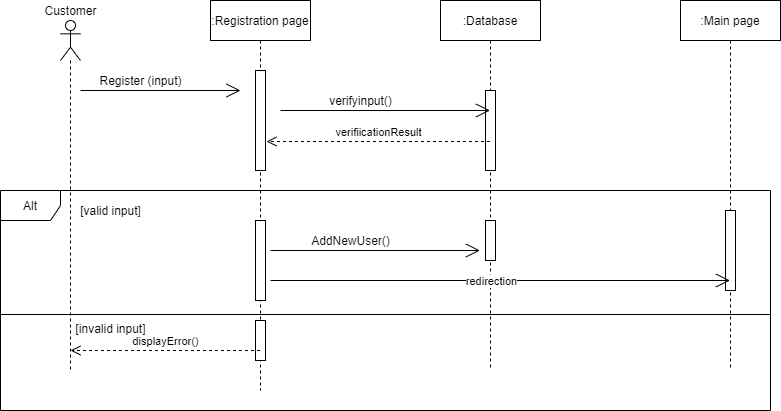
1. This use case begins when the admin logs in to his portal and selects edit restaurant option.
2. The admin edits the information and submits it.
3. The system shows editable information.
4. The system after authentication adds it to the database

**Sequence Diagrams for Batoor Delivery Services**

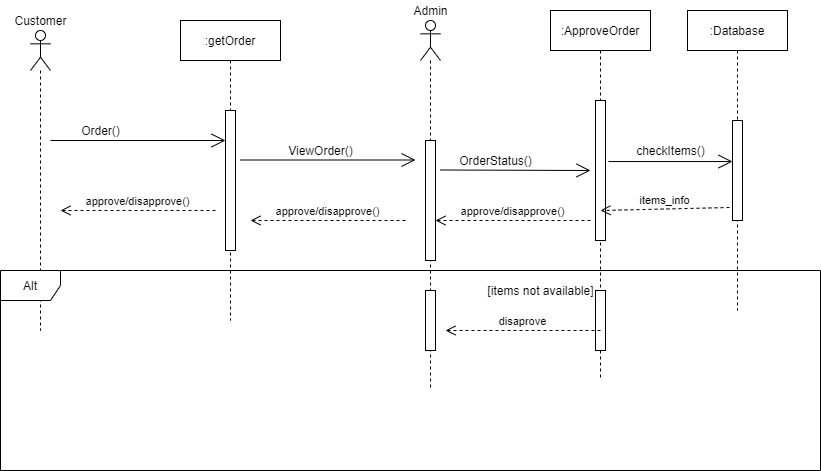
**SD1: Login**

****

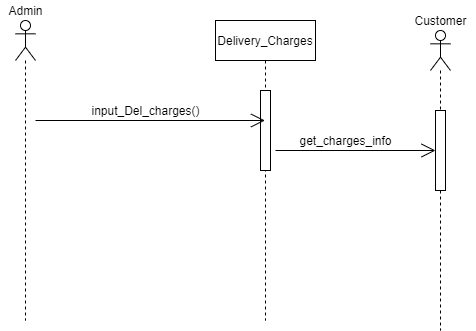
**SD 2: Registrar**

****

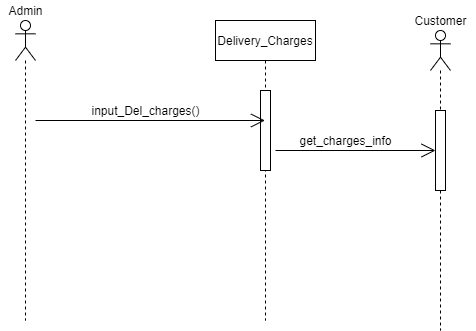
**SD 3: Order Approval**

****

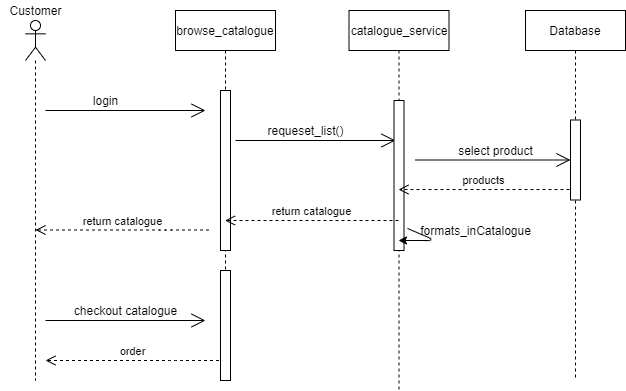
**SD 4: Delivery Charges on Online Shopping**

****

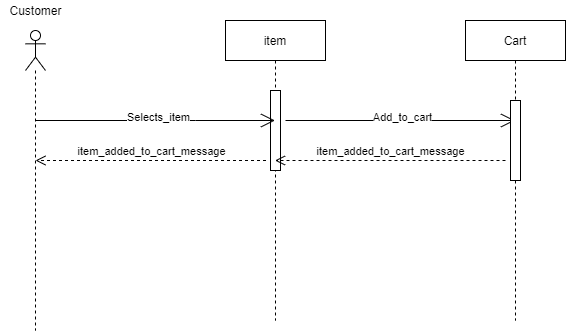
**SD 5: Delivery Charges on On-demand Delivery**

****

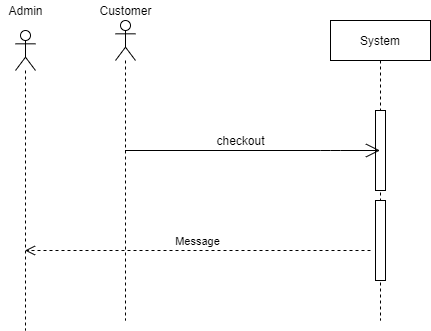
**SD 6: Browse**

****

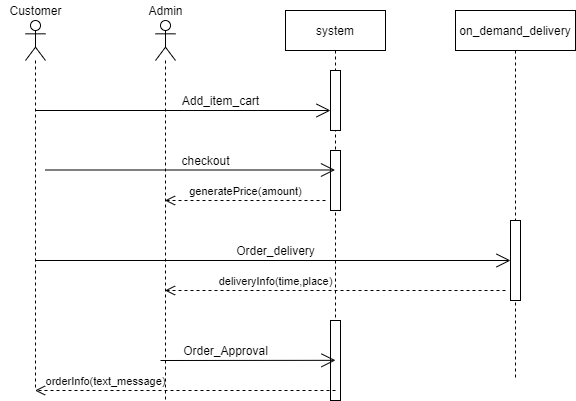
**SD 7: Add to Cart**

****

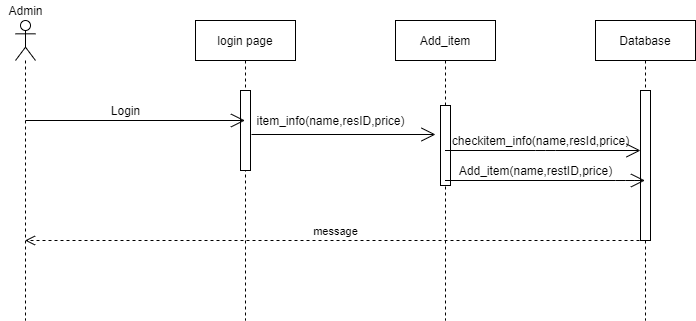
**SD 8: Checkout**

****

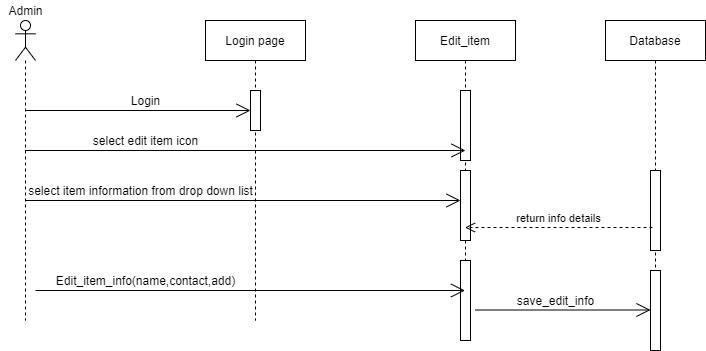
**SD 9: On-demand Delivery**

****

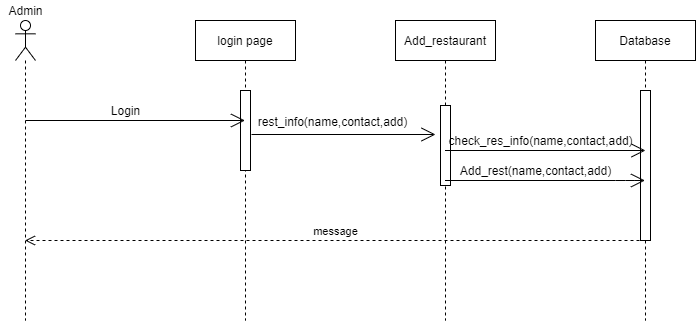
**SD 10: Add Item**

****

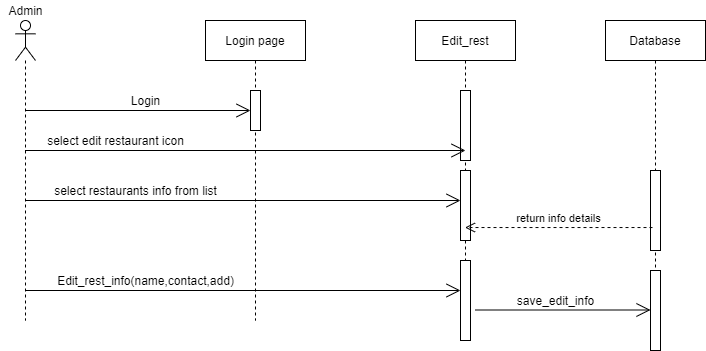
**SD 11: Edit Item**

****

**SD 12: Add Restaurant**

****

**SD 13: Edit Restaurant**

****

**System Sequence Diagrams for Batoor Delivery Services**

**SSD 1: Login**

A screenshot of a cell phone

Description automatically generated

**SSD 2: Register**

A screenshot of a social media post

Description automatically generated

**SSD 3: Order Approval**

A screenshot of a cell phone

Description automatically generated

**SSD 4: Delivery Charges on Online Shopping**

A screenshot of a cell phone

Description automatically generated

**SSD 5: Delivery Charges on On-demand Delivery**

A screenshot of a cell phone

Description automatically generated

**SSD 6: Browse**

A screenshot of a cell phone

Description automatically generated

**SSD 7: Add to Cart**

A screenshot of a social media post

Description automatically generated

**SSD 8: Checkout**

A screenshot of a social media post

Description automatically generated

**SSD 9: On-demand Delivery**

A screenshot of a cell phone

Description automatically generated

**SSD 10: Add Item**

A screenshot of a social media post

Description automatically generated

**SSD 11: Edit Item**

A screenshot of a cell phone

Description automatically generated

**SSD 12: Add Restaurant**

A screenshot of a cell phone

Description automatically generated

**SSD 13: Edit Restaurant**

A screenshot of a social media post

Description automatically generated