|  |
| --- |
| Bjorn’s Brew  Menu Ordering Application |
| **Software System Design** |

| Sean Ward [svw5927@psu.edu](mailto:svw5927@psu.edu) |
| --- |

# Table of Contents

[**Table of Contents**](#_ir9emmwbgzs9) **2**

[**Business Description**](#_9tcmmhhepwi5) **3**

[Locations](#_b6crgqvtfe42) 3

[**Business Requirements**](#_lyhrlbia43l3) **4**

[Problem Statement](#_c4t6lecfzi7w) 4

[Proposed Solution](#_rujcpef3uwgf) 4

[Business Goals](#_gqg6468mdkf3) 4

[Functional Requirements](#_iuiebtgxdwwz) 5

[Non-Functional Requirements](#_9ehkjdojzoay) 6

[Usability](#_qn4k3puwes75) 6

[Performance](#_zb3omolggykf) 6

[Security](#_m76sycujcy6a) 6

[**System Design**](#_mle1vtwl7ug7) **7**

[Use Case Analysis and Diagram](#_pnkjgwtmzc33) 7

[Actors](#_rs6ul9ijp9ee) 7

[Use Cases](#_4fw50igvmxxh) 7

[Domain Model](#_qqj1h3lj8fdd) 9

[Class Diagram](#_dmubh9e60wkc) 10

[Sequence Diagrams](#_808p73ehidd3) 11

[Use Case Scenario 1 - Place Order](#_gqiebft4qctg) 11

[Use Case Scenario 2 - Fulfill Order](#_kdzhdzcauxlt) 12

[Use Case Scenario 3 - Add Menu Item](#_e8eiwd5kdb3z) 13

[Use Case Scenario 4 - Remove/Pause Menu Item](#_sw3zjwfi1lij) 14

[State Diagram](#_ivia923zkwis) 15

[Activity Diagram](#_nqtsthk7epa1) 16

[Component Diagram](#_z3a65y9ezmyr) 17

[Deployment Diagram](#_9g1aksdbxtta) 18

[Skeleton Classes](#_sjywmwocvh9k) 19

[Data Models](#_i81rrihhq7je) 19

[Interfaces](#_3p80br3scntx) 20

# 

# Business Description

Bjorn’s Brew is a small chain of coffee shops operating in the greater Salt Lake City area. “Bjorn’s” as it is affectionately referred to by its customers, aims to provide high quality coffee, tea, pastries and other breakfast items to its customers at reasonable prices. Bjorn’s has made a great name for themselves in Salt Lake City by providing their goods through both drive through services and sit down locations.

### Locations

**Foothill Drive**

Foothill Drive is primarily drive through based. They have a single drive up payment window that is serviced through two drive up menu locations. These menu locations employ speakers and microphones for the customers to communicate their orders to the barista’s. Foothill Drive also uses a walk up counter where customers can place their orders directly to a barista if they are not in their cars.

**Highland Drive**

Highland Drive is primarily drive through based. They have 2 drive up payment windows that are each serviced through their own drive up menu locations. These menu locations employ speakers and microphones for the customers to communicate their orders to the barista’s. Highland Drive also uses a walk up counter where customers can place their orders directly to a barista if they are not in their cars.

# Business Requirements

## Problem Statement

As the popularity of Bjorn’s Brew increases, the amount of patrons that they are servicing each day is greatly increasing. Since their locations rely so heavily on their drive through ordering experience, this can lead to long lines of cars and increased wait times for their patrons. Their ordering process is starting to develop a bottleneck at their drive up menus where customers order their items verbally through the microphone.

## Proposed Solution

Bjorn’s would like to create an online menu ordering application where customers are able to place their orders prior to arriving at any of their locations. These online orders will be placed using a mobile device. This application will allow for faster order fulfillment leading to less wait time for the customers.

## Business Goals

Reduce wait times for patrons

Reduce friction in ordering process

Spread order requests over larger period of time

Streamline inventory management

## Functional Requirements

**User Interface**

The system shall provide an interface for users to order goods.

The system shall allow users to select the location where they will be ordering from.

The system shall display a menu (based on location) of all the goods that Bjorn’s Brew provides.

The system shall allow users to select items from the menu

The system shall allow users to provide customizations on their orders

The system shall allow users to pre order their selected items

The system shall allow users to choose their pick up time.

The system shall allow users to pre pay for their order.

The system shall integrate with current POS System

The system shall allow users to cancel their orders.

**Employee Interface**

The system shall provide an interface for employees to receive orders

The system shall organize orders based on pickup time

The system shall allow orders to be marked as “assembling”

The system shall allow orders to be marked as “assembled”

The system shall allow order to be marked as fulfilled

# 

## Non-Functional Requirements

### Usability

The system must be highly usable. To encourage patrons of Bjorn’s Brew to start placing their orders through the application, all friction in the application ordering process must be minimum.

### Performance

The system must

### Security

The system will be handling sensitive data (customer billing info, credit cards) and must therefore be highly secure. All credit card data must be encrypted to protect the business and the customer.

# System Design

## Use Case Analysis and Diagram

### Actors

| **Type** | **Actor** | **Goal Description** |
| --- | --- | --- |
| Primary | | |
|  | Customer | Order their items with the least amount of friction as possible. Wait in line for the shortest amount of time. Feel comfortable using an application to place their orders. |
|  | Barista | Deliver the order to the customer as efficiently as possible. Receive orders in a well organized manner. |
|  | Manager | Ensure a productive workplace environment for the workers. Provide excellent service to customers. |
| Secondary | | |
|  | Technical Administrator |  |
|  | Owner | Increase bottom line. Streamline process to be able to provide as many orders as possible efficiently. Promote brand trust to patrons. |

### Use Cases

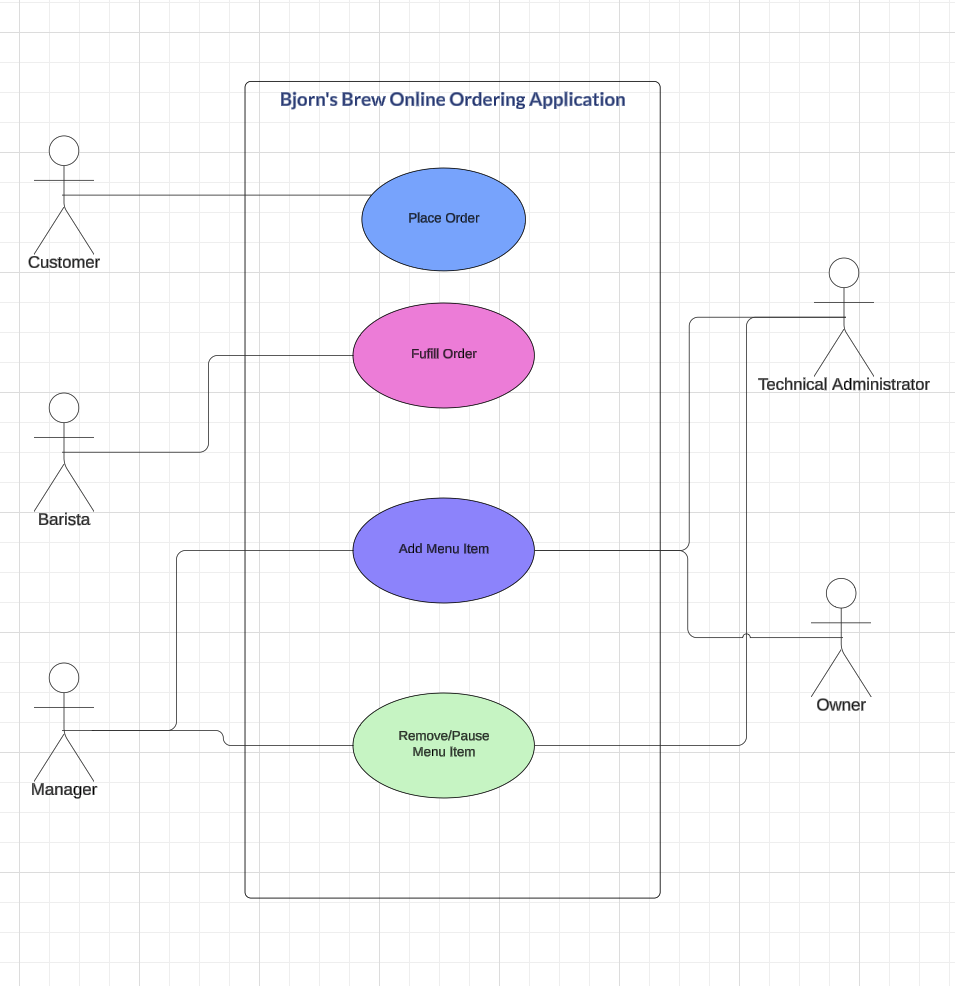
Use Case 1 - Place Order

Use Case 2 - Fulfill Order

Use Case 3 - Add Menu Item

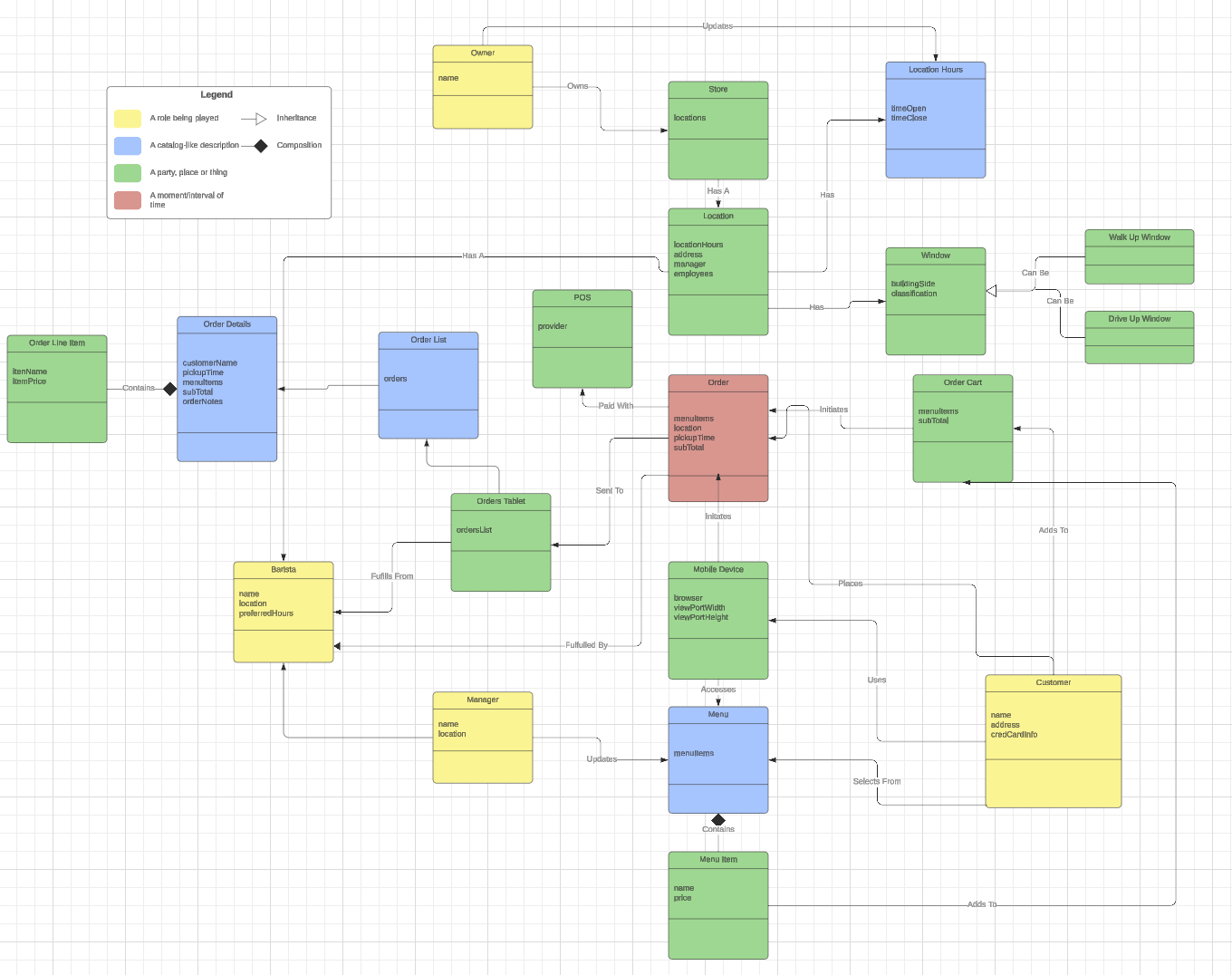
Use Case 4 - Remove/ Pause Menu Item

Figure 1: Use Case Diagram



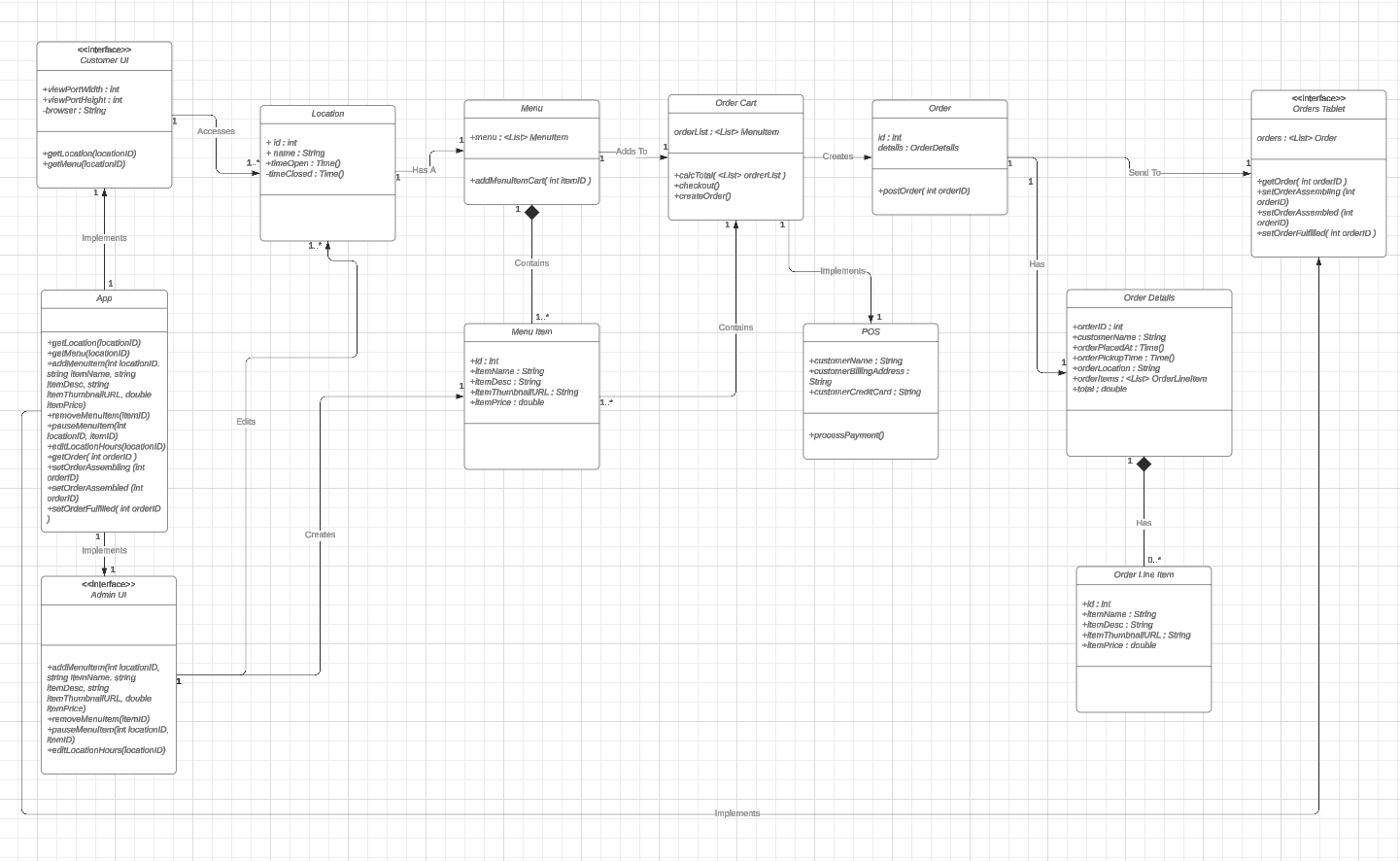
## Domain Model

Figure 2: Domain Model



## Class Diagram

Figure 3: Design Class Diagram



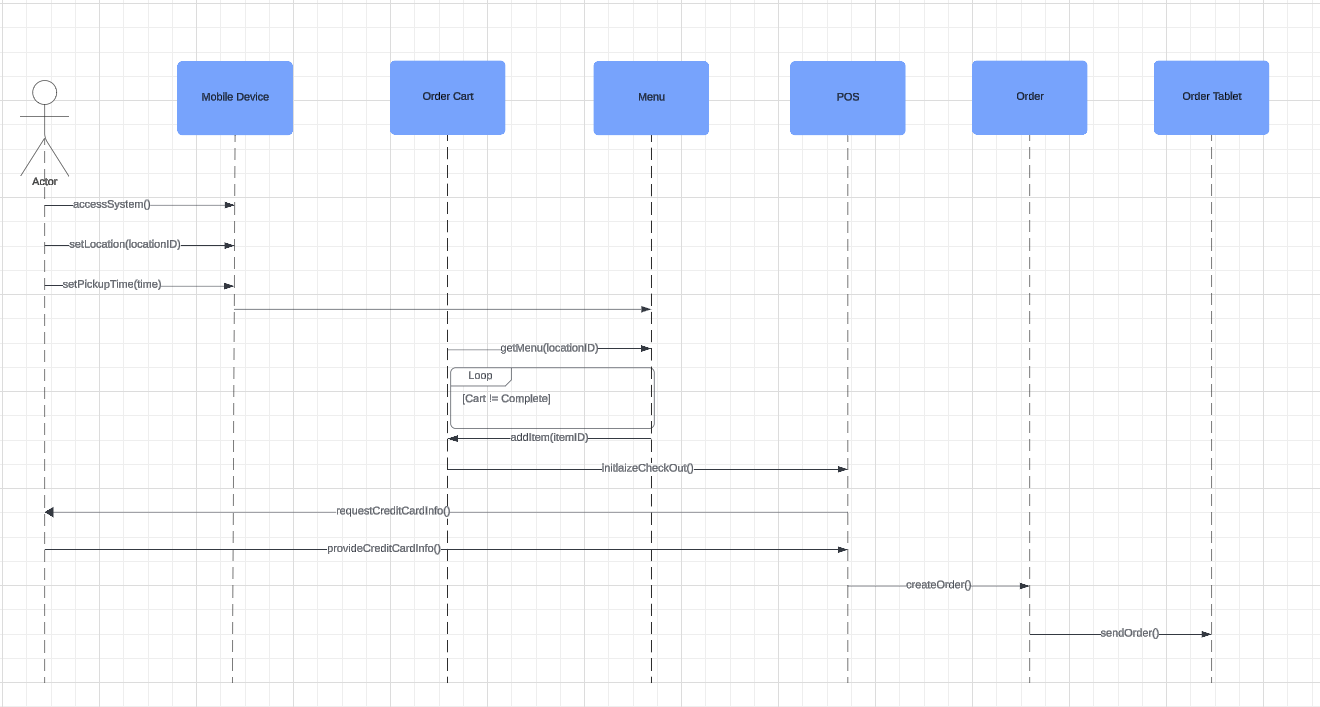
Design Decisions

## Sequence Diagrams

### Use Case Scenario 1 - Place Order

1. Customer accesses system through mobile device
2. Customer selects New Order
3. Customer selects location
4. Customer selects pickup time
5. Customer is directed to location menu
6. Customer selects menu item
7. Customer adds menu item to cart
8. Customer repeats steps 6 and 7 until satisfied with their cart
9. Customer selects checkout
10. Total in calculated and displayed to the customer
11. Customer inputs their credit card information
12. Customer’s card is charged
13. Order details are sent to the location

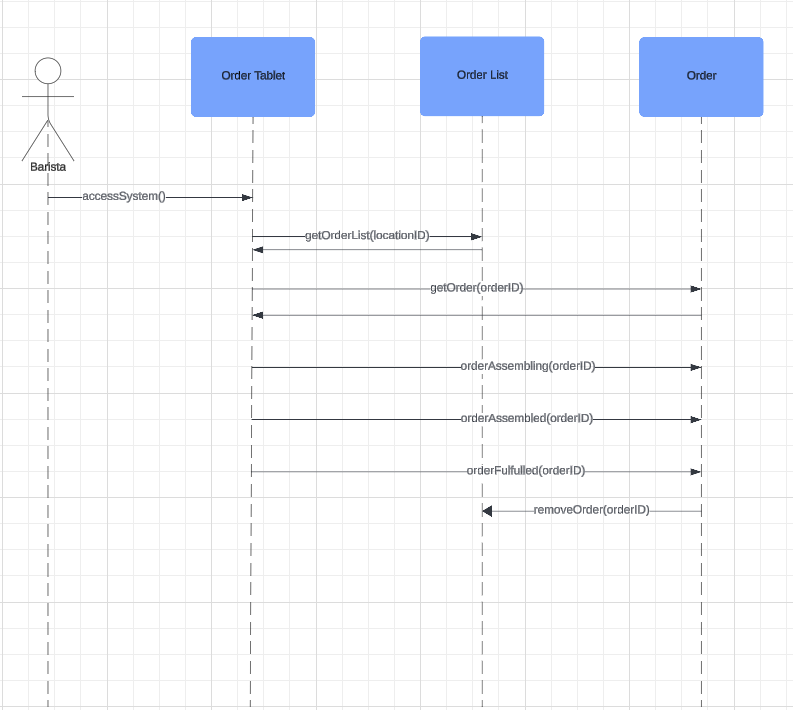
Figure 4: Sequence Diagram - Use Case 1



### Use Case Scenario 2 - Fulfill Order

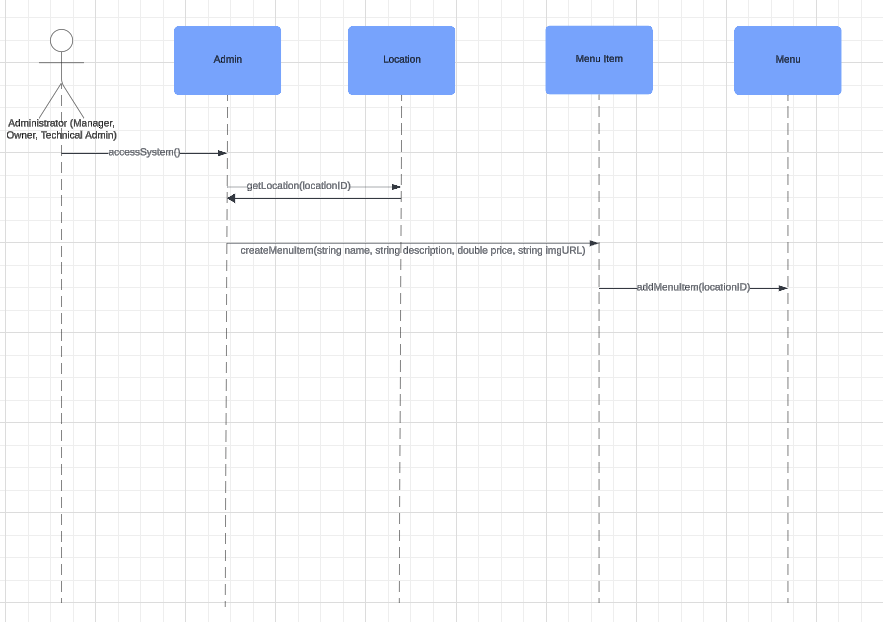
1. Barisa accesses system through orders tablet
2. Barista selects order based on pickup time
3. Order details are displayed to the barista
4. Barista marks the order as “assembling”
5. Barisa assembles the order
6. Barista marks order as “assembled”
7. Customer arrives at fulfillment window
8. Barista provides customer with their assembled order
9. Barista marks the order as “fulfilled”
10. Order is removed from Order List

Figure 5: Sequence Diagram - Fulfill Order



### Use Case Scenario 3 - Add Menu Item

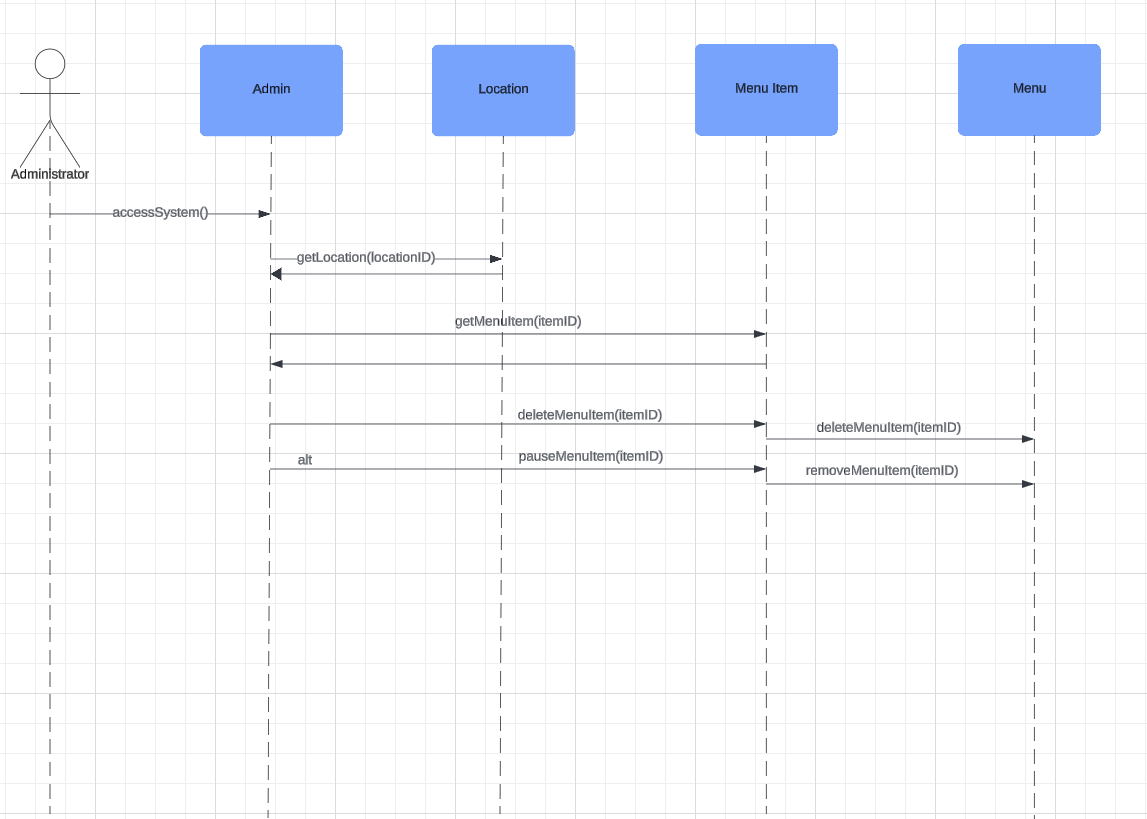
1. Administrator logs in to system backend
2. Administrator selects location
3. Administrator selects add menu item
4. System provides new menu item form
5. Administrator inputs menu item name, menu item price, menu item description, menu item thumbnail
6. Administrator submits new menu item

Figure 6: Sequence Diagram - Add Menu Item

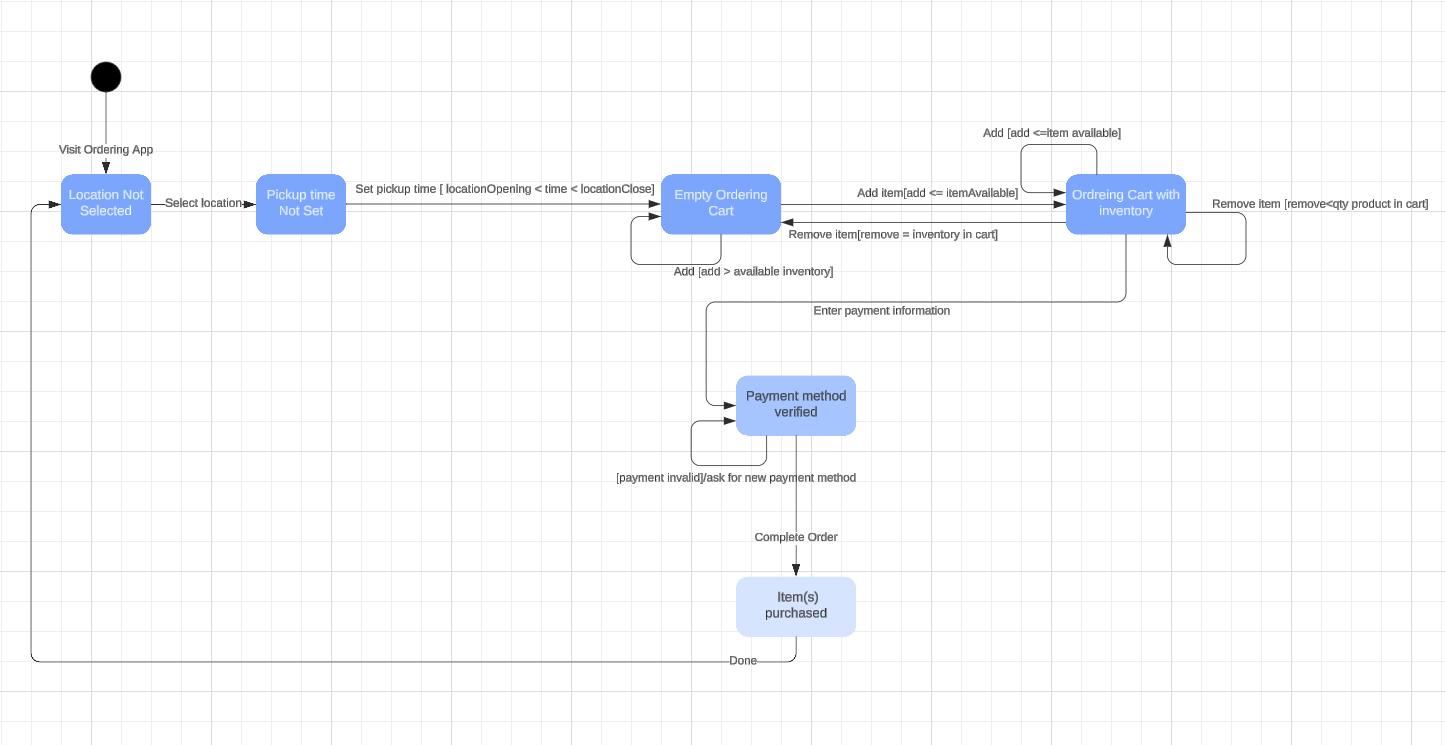
### Use Case Scenario 4 - Remove/Pause Menu Item

1. Administrator logs into system backend
2. Administrator selects location
3. Administrator accesses location menu
4. Administrator selects a menu item
5. Administrator selects remove item
6. System deletes menu item
7. Administrator selects pause item
8. System removes the menu item from menu, does not delete the item in database.

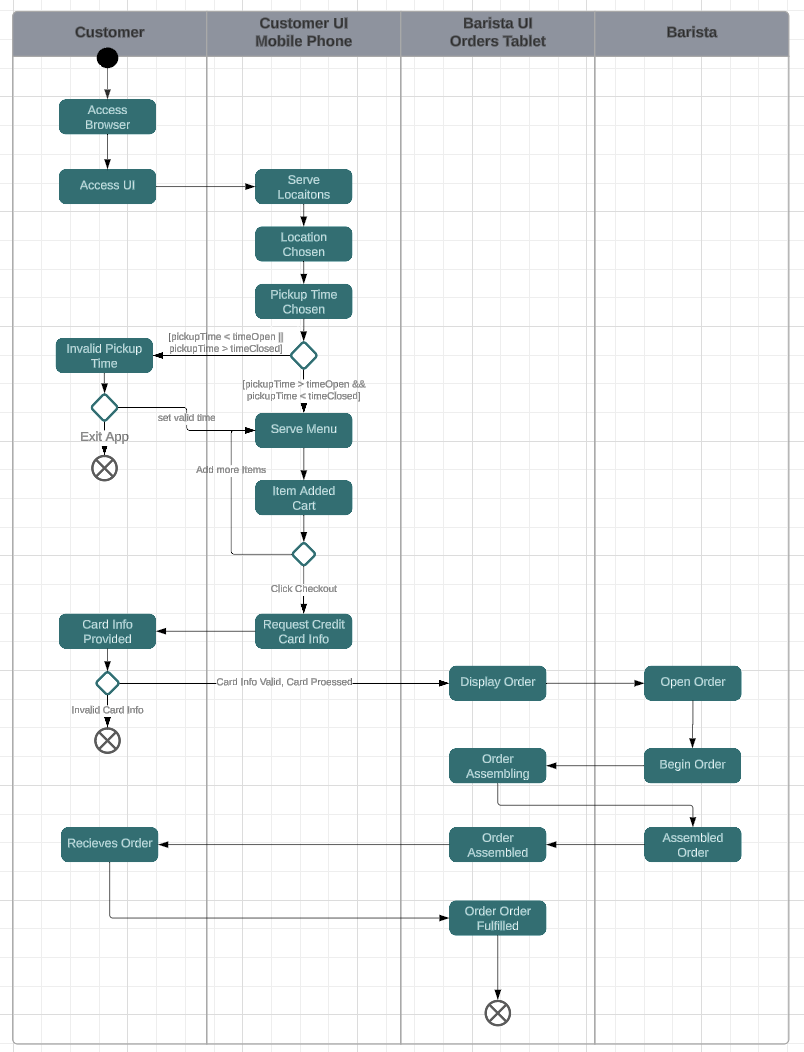
Figure 7: Sequence Diagram - Add Menu Item



## State Diagram

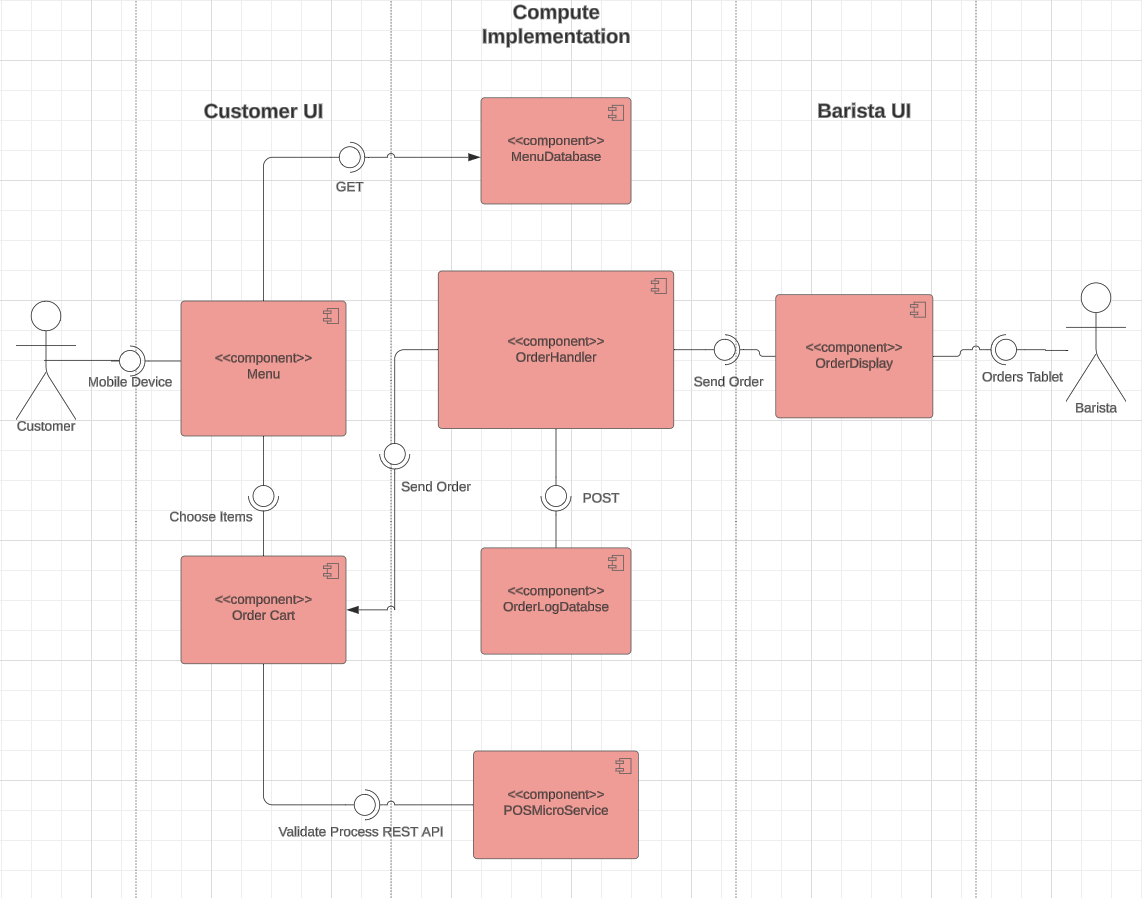
Figure 8: State Diagram - Use Case 1

## Activity Diagram

Figure 9: Activity Diagram - Use Case 1 & Use Case 2

## Component Diagram

Figure 10: Component Diagram



## Deployment Diagram

Figure 11: Deployment Diagram



## Skeleton Classes

### Data Models

public class Location{

public int ID;

public String name;

public String address;

public Time timeOpen;

public Time timeClosed;

public Menu menu;

}

public class Menu{

public <List> MenuItem items;

public void addMenuItemCart(int itemID){}

}

public class MenuItem{

public int itemID;

public String itemName;

public String itemDesc;

public String itemThumbnailUrl;

pubic double itemPrice;

}

public class Order{

public int id;

public OrderDetails details;

public void postOrder(int orderID){}

}

public class OrderDetails{

public int orderID;

public String customerName;

public Time orderPlacedAt;

public Time orderPickupTime;

public String orderLocation;

public <List>OrderLineItem orderItems;

public double total;

}

public class OrderLineItem{

public int itemID;

public String itemName;

public String itemDesc;

public String itemThumbnailUrl;

pubic double itemPrice;

}

### Interfaces

public class OrderDispay{

public <List> Order orders;

public void getOrder( int orderID){}

public void setOrderAssembling(int orderID){}

public void setOrderAssembed(int orderID){}

public void setOrderFulfilled(int orderID){}

}

public class CustomerUI{

public int ViewPortWidth;

public int ViewPortHieght;

public String browser;

public void getLocation(locationID){}

public void getMenu(locationID){}

}

public class CustomerUI{

public void addMenuItem( int locationID, String itemName, String itemDesc, String itemThumbnailURL, double itemPrice)

public void removeMenuItem( int itemID)

public void pauseMenuItem( int itemID)

public void editLocationHours (int locationID)

public void updateLocationHours (Time timeOpen, Time timeClosed)

}