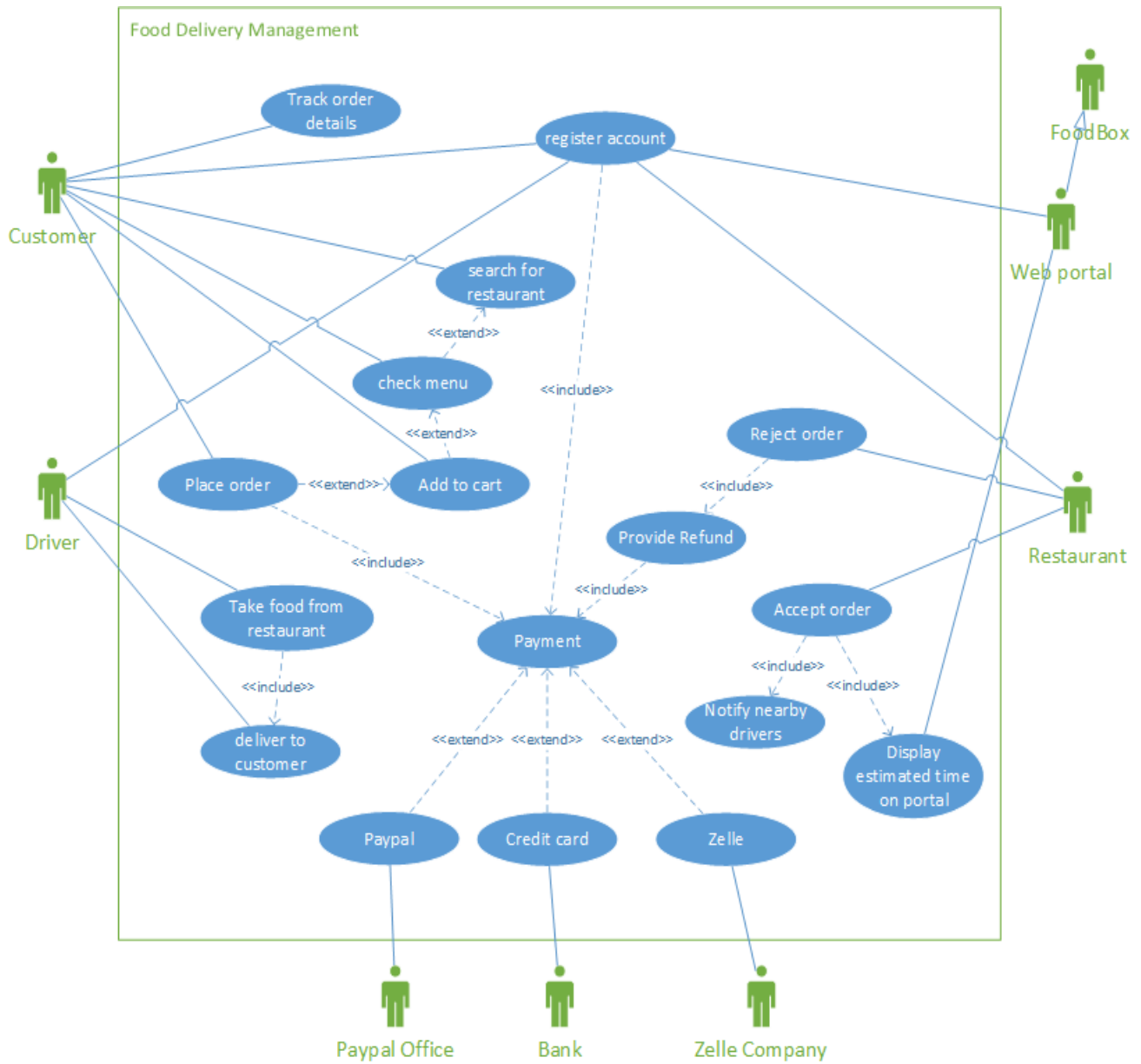


## Food delivery Management System

---

FoodBox is an online business that serves many customers by allowing them to order food for delivery from local restaurants. The company has a web portal that has a listing of local restaurants for a zip code. The restaurants will be able to register with the portal and list their menus for an annual subscription fee of \$190 or a monthly subscription fee of \$25. Drivers, who are willing to pick up food from restaurants and deliver to customers will be able to register with the portal for an annual subscription fee of \$95 or a monthly subscription fee of \$15. Customers will be able to search for restaurants based on the zip code and view a list of all restaurants registered with FoodBox around that area. They will be able to view the menu for a particular restaurant and place an order. The payment is complete when the order is placed. Once an order is placed, the restaurant can either accept or reject the order. If an order is rejected, the customer must be refunded for the payment made. If an order is accepted by the restaurant, the restaurant provides an estimated time in the portal when the order will be ready for pickup. The application searches for drivers who are near the restaurant and available to deliver the food and notifies the selected driver to drive to the restaurant at the time when the order is expected to be ready for pickup. The driver is also provided with the address of the customer to whom the delivery needs to be made. The portal also provides a way for users to register for an account with the system, track their orders, view their order history and process a payment using credit card, paypal or zelle.

## 1. Use Case Diagram:



## Classes, Attributes, and Operations:

### User

+User\_id: int  
+User\_name: String  
+User\_Password:String  
+User\_address: string  
+User\_phone:int

+editUser()  
+searchUser()  
+deleteUser()  
+addUser()

### Login

+Login\_username: String  
+Login\_password: String  
+Login\_ID:string  
+Login\_date:date  
+Login\_history:string

+enterUsername ()  
+enterPassword()  
+forgotPassword()  
+forgotUsername()

### Search engine

+search\_id: string  
+search\_history:string  
+search\_page:string  
+search\_results:string

+addSearch()  
+editSearch()  
+newsearch()  
+deleteSearchHistory()

### Food Menu

+food\_ID: string  
+food\_details:string  
+food\_timing: date  
+food\_Rating:int

+searchFoodItems()  
+requestFoodInquiry()  
+hideFoodItems()  
+editFoodIngredients()

### Restaurant

+restaurant\_ID: string  
+restaurant\_name: string  
+restaurant\_address: string  
+restaurant\_phone: string  
+restaurant\_ratings: int  
+restaurant\_type: string

+selectRestaurant()  
+editrestaurantDetails()  
+checkRestaurantstars()  
+selectRestaurantType()

### shopping Cart

+cart\_ID: string  
+cart\_itemcount: int  
+cart\_limit:int  
+cart\_price:int

+addToCart()  
+editCart()  
+checkCartPrice()  
+checkCartLimit()

### Bill

+Billing\_ID:string  
+billing\_details:string  
+billing\_address:string  
+billing\_date: date

+EditBill()  
+confirmBill()  
+AcceptBill()  
+RejectBill()

### order details

+Order\_ID: String  
+Order\_Location:String  
+Order\_date:date  
+Order\_detail:string

+TrackOrder()  
+EditOrderDetails()  
+CancelOrder()  
+shareOrderDetails()

### Driver

+driver\_ID: String  
+driver\_name: String  
+driver\_address: String  
+driver\_phone: String

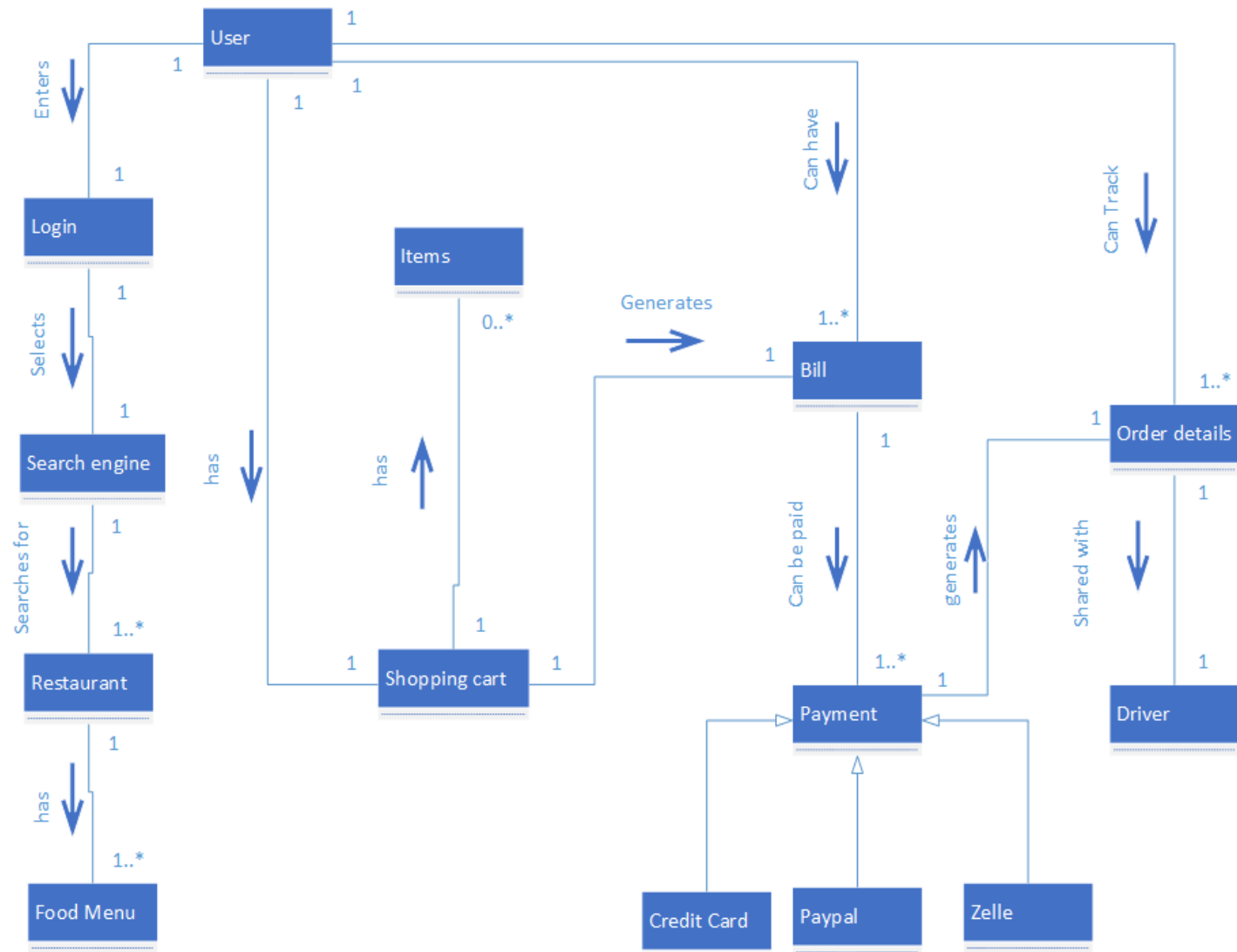
+getDriverDetails()  
+callDriver()  
+messageDriver()  
+rateDriver()

### Payment

+payment\_ID: String  
+payment\_date: date  
+payment\_address: String  
+payment\_details: String

+useCreditCard()  
+usePaypal()  
+UseZelle()  
+CancelPayment()

## CLASS DIAGRAM



**Sequence Diagram for the following Use Case Description:**

<b>Use Case Name:</b>	Create Order
<b>Primary Actor(s):</b>	Customer
<b>Brief Description:</b>	This use case describes how an order will be created using the system.
<b>Trigger:</b>	Customer accesses the portal and creates an order.
<b>Precondition:</b>	None
<b>Normal flow of events:</b>	<ol style="list-style-type: none"><li>1. Customer searches for a restaurant by name or zip code.</li><li>2. Customer accesses the restaurant's menu.</li><li>3. Customer selects the food they want to order and adds the items to cart.</li><li>4. Customer fills in billing and delivery information.</li><li>5. Customer makes the payment by zelle, credit card or paypal.</li><li>6. The system creates the order and notifies the restaurant.</li></ol>
<b>Alternate/Exception flow:</b>	<ol style="list-style-type: none"><li>1. <i>The customer's credit card is rejected and the payment was not complete. Customer is asked to provide alternate card or use alternate payment methods.</i></li><li>2. <i>The billing information provided by the customer is invalid. Customer must enter valid billing information</i></li></ol>

