Design Document

for

Restaurant Billing Management System

Version<1.0>

Prepared by Bhukya Vasanth Kumar:

(Based on SRS Version <1.0> prepared by Bhukya Vasanth Kumar)

Project Owner: **Mr. Konu Pruthviraj**

Course: **CS4097D Object Oriented Systems Laboratory**

Date: **October 18, 2021**

Glossary

Acronyms and Abbreviations

UML	Unified Modelling Language
RBMS	Restaurant Billing Management System
DD	Design Document

Table of contents

1. Detailed Design through UML diagrams

1.1 System model using Class Diagram

Class Diagram in the Unified Modelling Language is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods) and the relationships among classes.

1.1.1 Class Diagram

In software engineering, a class diagram in the **Unified Modelling Language (UML)** is **a type of static structure diagram** that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among objects.

Purpose of Class Diagrams

1. Shows static structure of classifiers in a system
2. Diagram provides a basic notation for other structure diagrams prescribed by UML
3. Helpful for developers and other team members too
4. Business Analysts can use class diagrams to model systems from a business perspective

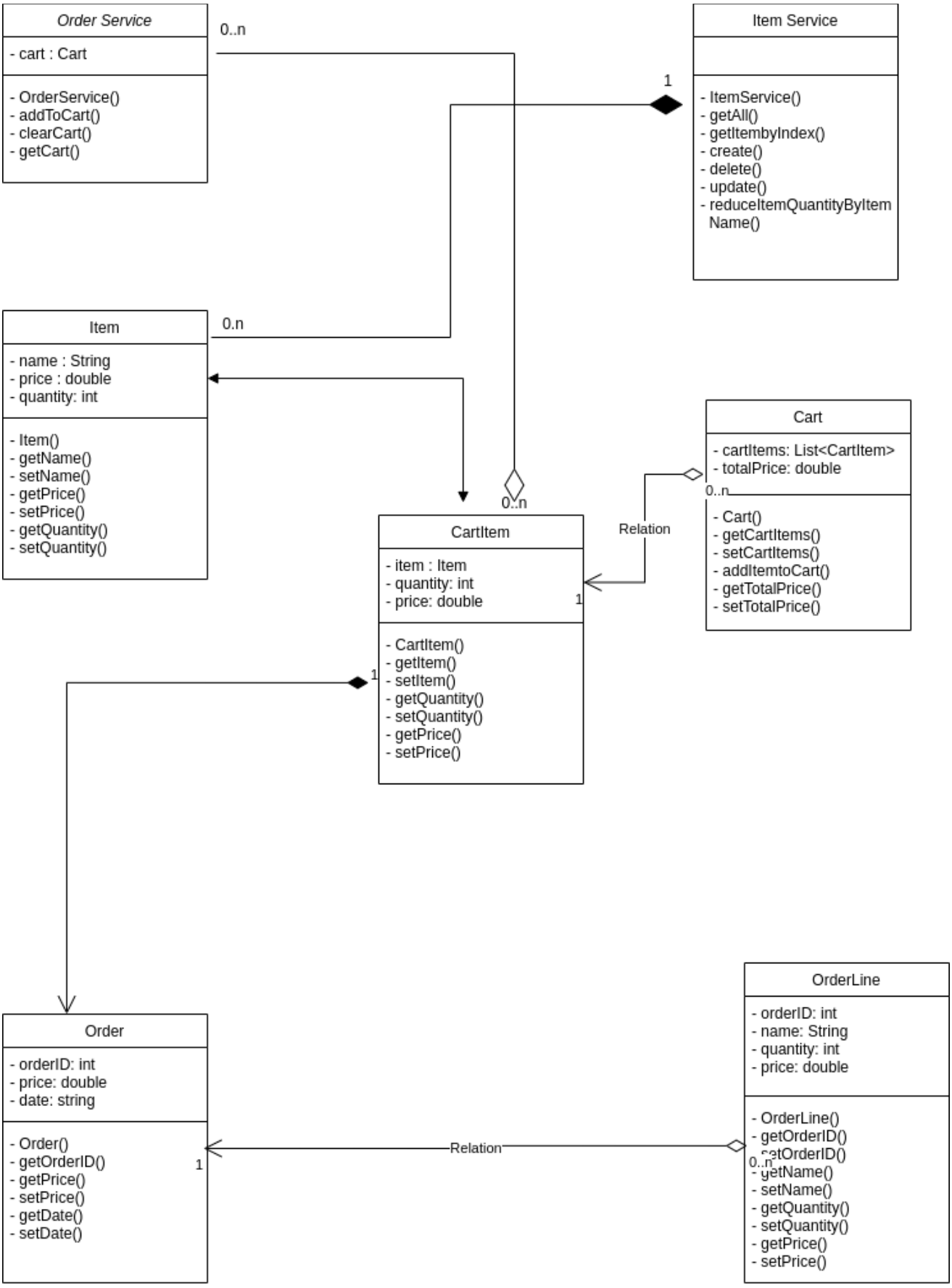
A UML class diagram is made up of:

- A set of classes and
- A set of relationships between classes

What is a Class

A description of a group of objects all with similar roles in the system, which consists of:

- **Structural features** (attributes) define what objects of the class "know"
    - Represent the state of an object of the class
    - Are descriptions of the structural or static features of a class
- **Behavioural features** (operations) define what objects of the class "can do"
    - Define the way in which objects may interact
    - Operations are descriptions of behavioural or dynamic features of a class

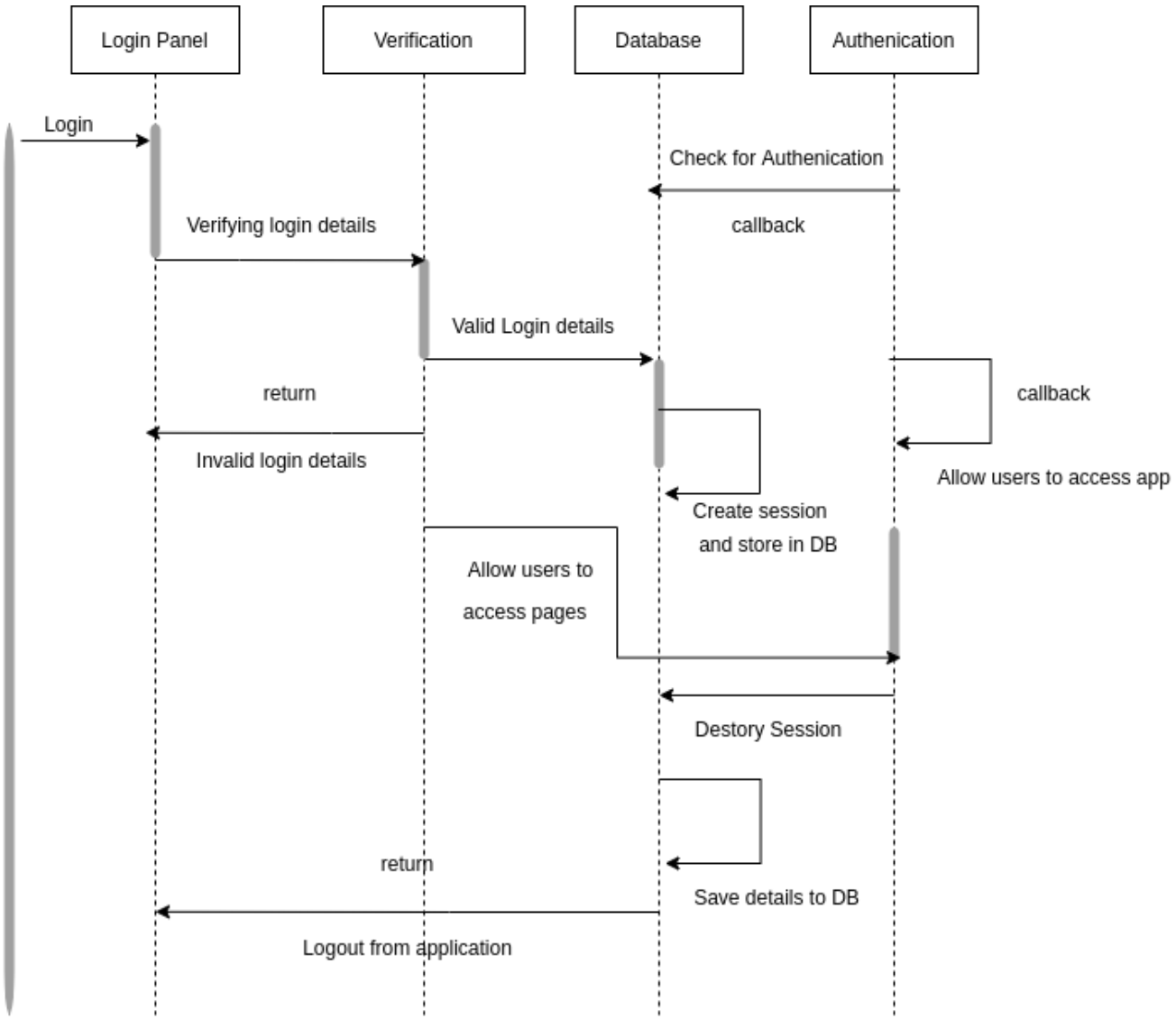

Order Service
- cart : Cart
- OrderService()
- addToCart()
- clearCart()
- getCart()
0..n
1
Item Service
- ItemService()
- getAll()
- getItembyIndex()
- create()
- delete()
- update()
- reduceItemQuantityByItem Name()
Item
- name : String
- price : double
- quantity: int
- Item()
- getName()
- setName()
- getPrice()
- setPrice()
- getQuantity()
- setQuantity()
0.n
0..n
Cart
- cartItems: List<CartItem>
- totalPrice: double
- Cart()
- getCartItems()
- setCartItems()
- addItemtoCart()
- getTotalPrice()
- setTotalPrice()
0..n
Relation
CartItem
- item : Item
- quantity: int
- price: double
- CartItem()
- getItem()
- setItem()
- getQuantity()
- setQuantity()
- getPrice()
- setPrice()
1
1
Order
- orderID: int
- price: double
- date: string
- Order()
- getOrderID()
- getPrice()
- setPrice()
- getDate()
- setDate()
1
Relation
OrderLine
- orderID: int
- name: String
- quantity: int
- price: double
- OrderLine()
- getOrderID()
- setOrderID()
- getName()
- setName()
- getQuantity()
- setQuantity()
- getPrice()
- setPrice()
0..n

1.2 System Interactions through Sequence Diagrams

Sequence diagrams are interaction diagrams that show the sequence of messages exchanged by the set of objects performing a certain task. A sequence diagram shows, as parallel vertical lines (lifeline), different processes or objects that live simultaneously, and as horizontal arrows, the messages exchanged between them, in the order in which they occur.

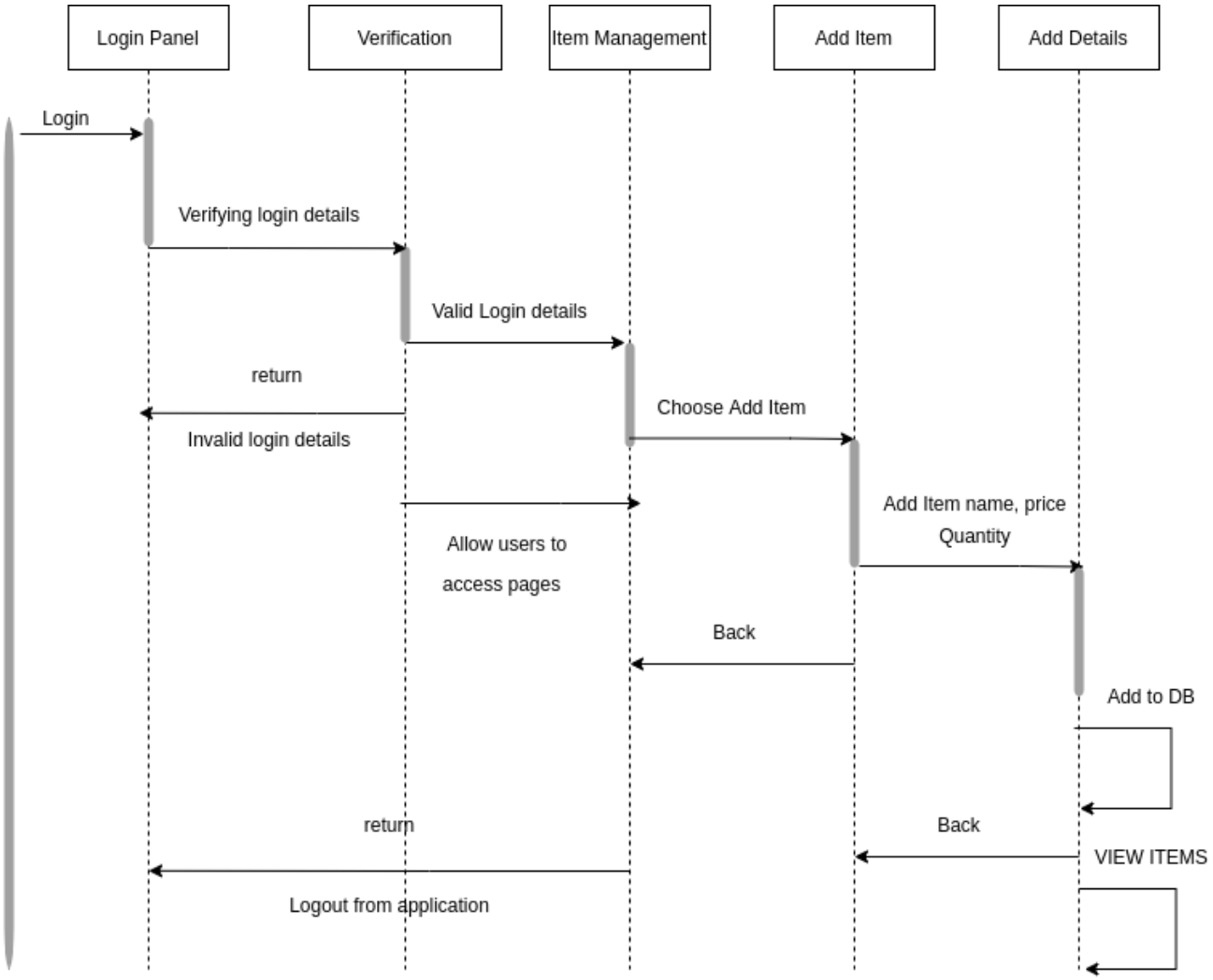
1.2.1 <Login Sequence Diagram>

This is the sequence diagram of login for all actors we have in this project. They can login using their userID and password. If they are valid, they can get access to the Database and further authentication will happen.



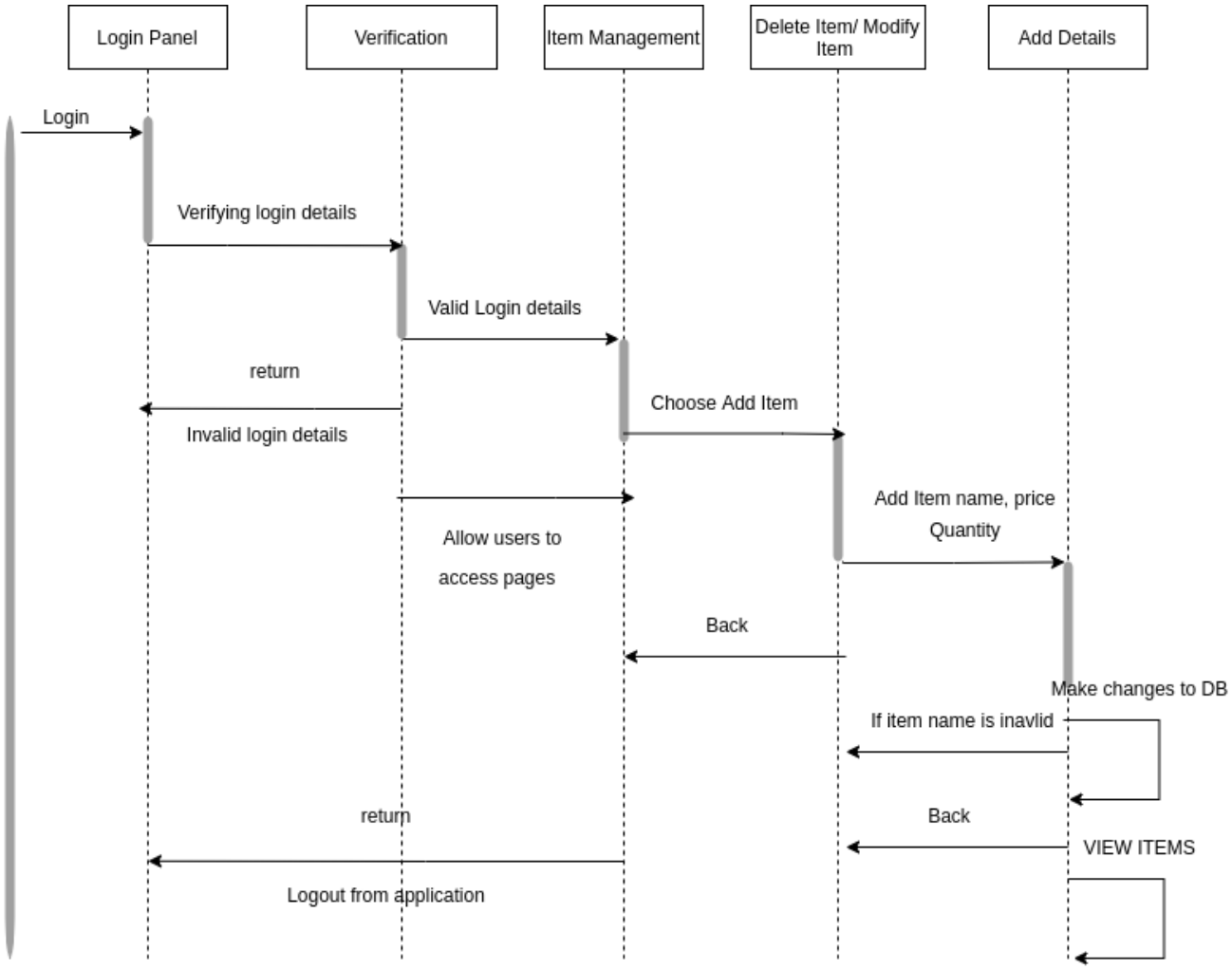
1.2.2 <Add items into the Inventory>

This sequence diagram will allow users to add items in the inventory after logging in. Users can go to Item Management and the can select ADD ITEM option and have to enter the Item name, price and the quantity of item. They can later select VIEW ITEMS to see all the items in the inventory.



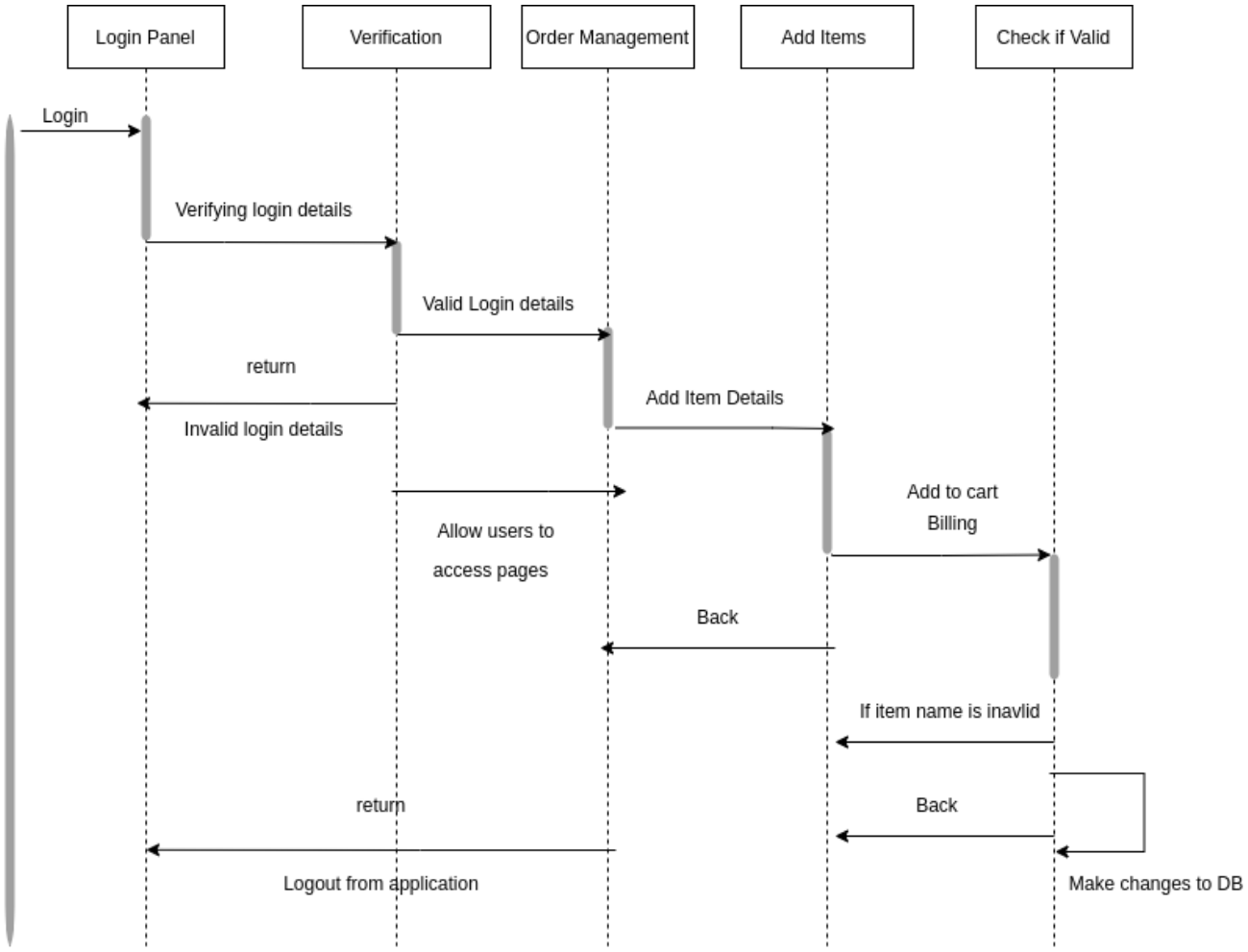
1.2.3 <Delete or Modify items in the Inventory>

This sequence diagram will allow users to delete or modify items in the inventory after logging in. Users can go to Item Management and the can select DELETE ITEM / MODIFY ITEM option and have to enter the Item name. Based on the item name, you can delete the item or modify the details of that item. They can later select VIEW ITEMS to see all the items in the inventory.



1.2.4 <Order Items>

This sequence diagram will allow users to order items from the inventory after logging in. Users can go to Order Management and the can enter Item name and quantity they want to order. If the item and the respective quantity is available, order can be placed in the cart. Later, Bill is displayed.



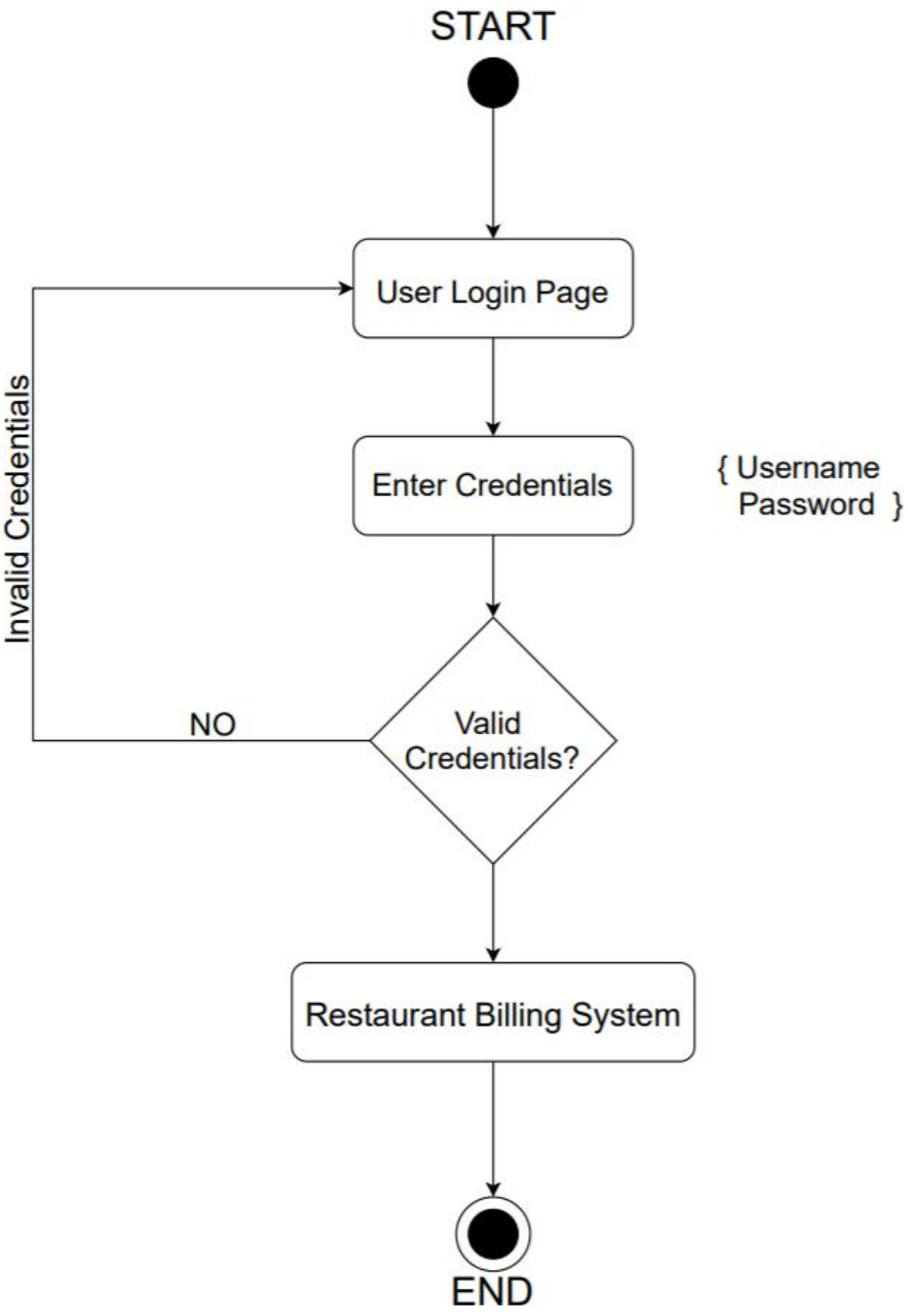
1.3 Control and Data Flows through Activity Diagrams

Activity diagrams show the interaction between the various IT systems that are involved in the message exchange. These diagrams, also known as control flow and object flow diagrams, are a type of behavioural diagram in the UML (unified modelling language). They describe the sequential, conditional, and parallel composition of lower-level behaviours using a graphical notation.

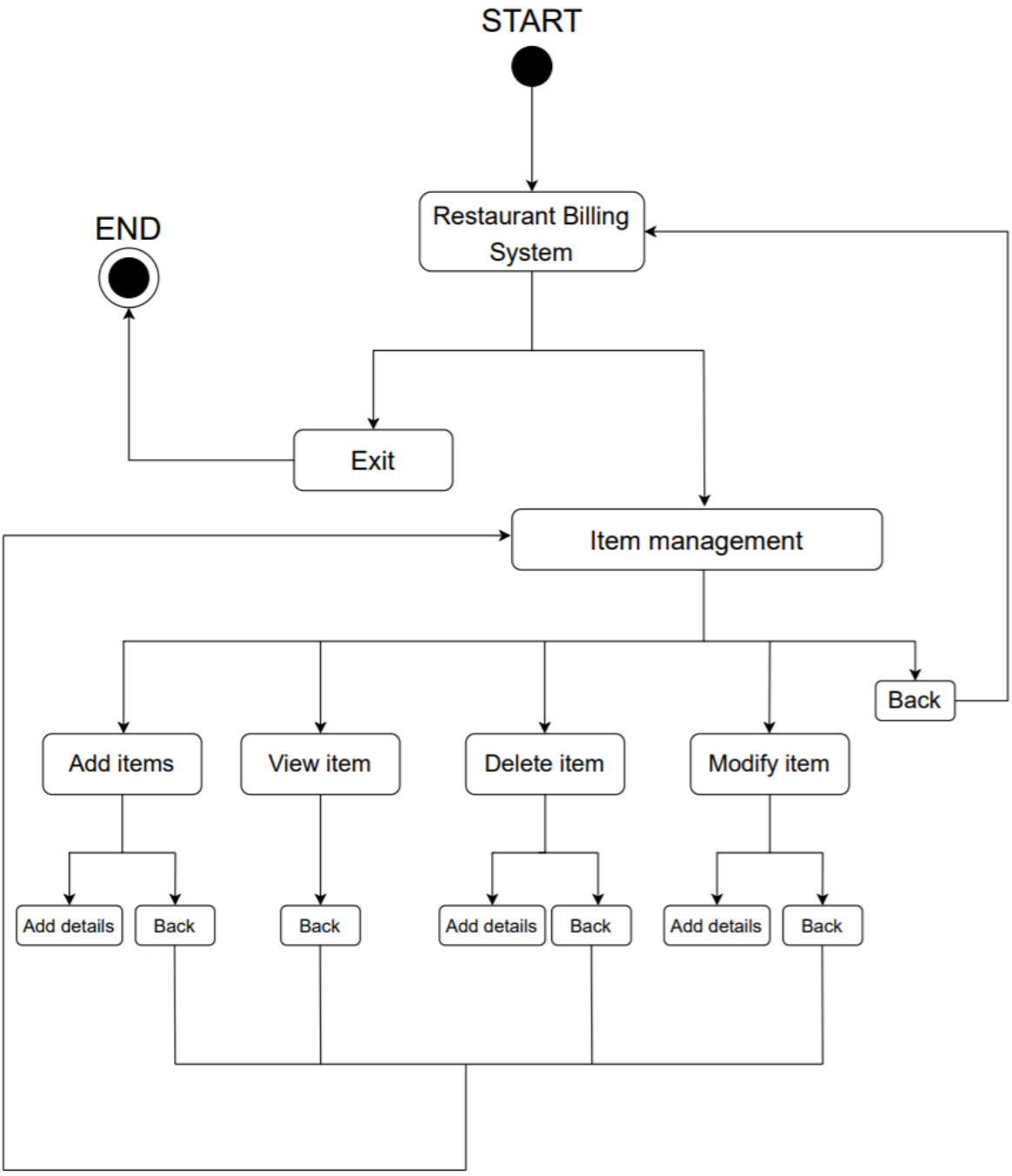
Edges, which are depicted as arrows, connect the individual components of the activity diagram and represent the control flow and object flow (edge) of the activity. The control flow determines the flow within an activity. The incoming arrow starts an individual step of an activity. After this step is completed the flow continues along the outgoing arrow. The object flow describes the flow of objects and data within activities, Edges can be labelled with a name (close to the arrow).

The object flow in an activity diagram shows the path of one or more business objects between the various activities.

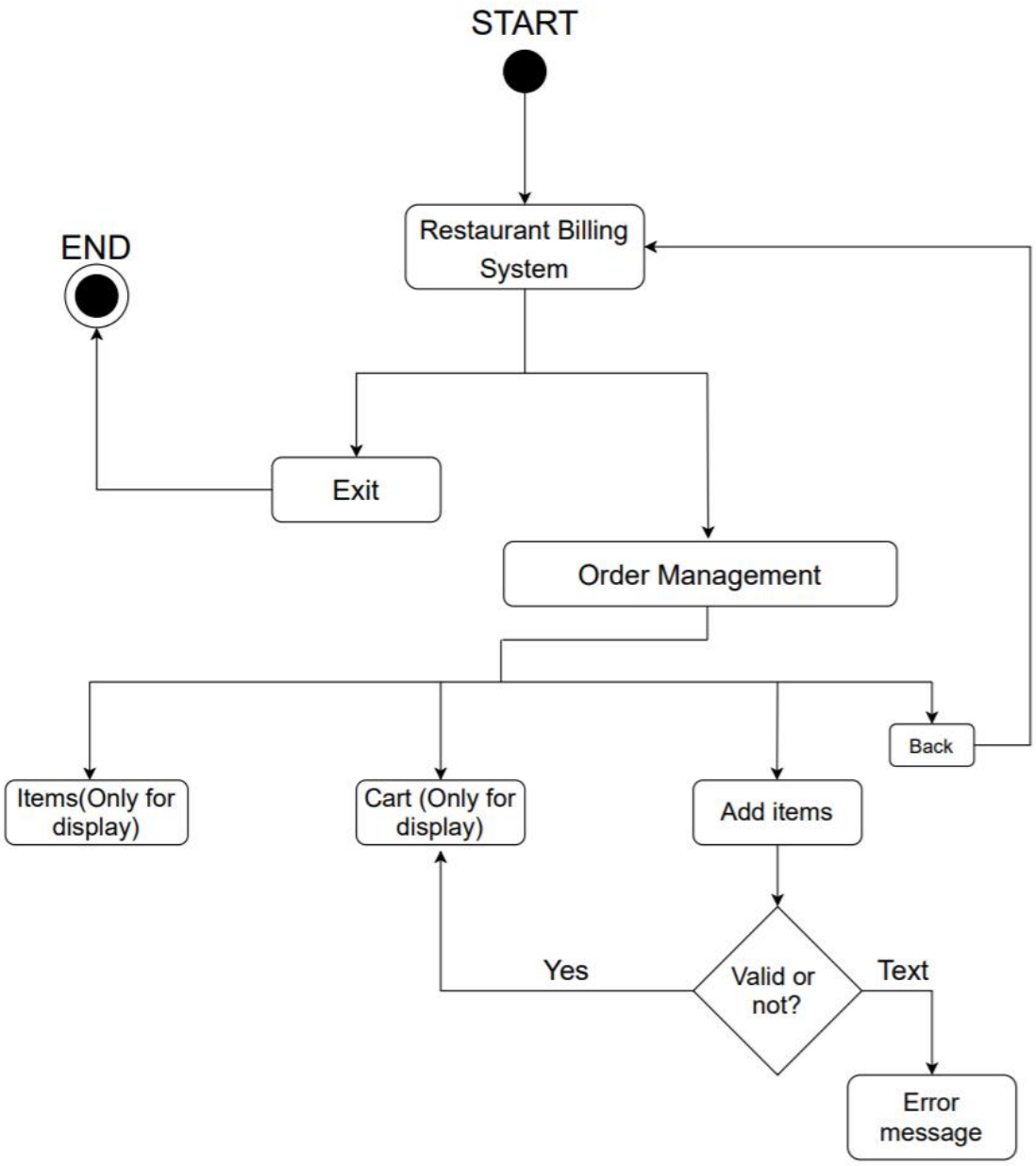
1.3.1 <Login Activity>



1.3.2 <Items Activity>



1.3.3 <Order Activity>



References

https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-class-diagram/

https://www.tutorialspoint.com/uml/uml_class_diagram.htm

https://en.wikipedia.org/wiki/Sequence_diagram

https://www.tutorialspoint.com/uml/uml_activity_diagram.htm