



SHENKAR UNIVERSITY COLLEGE

DEPARTMENT OF COMPUTER



Shenkar

SCIENCE

BEBS-BOOKSHOP SYSTEM

(SYSTEM TO BUY/FIND BOOK WITH THE CART)

DEVELOPE BY:

IDAN KARIO 300853751

idankario@gmail.com

SHANI LEVI 302853619

shanilevi01011@gmail.com

Content

1. Introduction 2

- a. System Overview 2
- b. Purpose 2
- c. Scope 2
- d. Constraints 3

2. System Architecture 4

- a. Architectural Description and Design: Roles, Activities and Data 4
- b. The Life Cycle of the System 5

3. Design 7

- a. Data Design - Database Description 7
- b. Structural Design - Class Diagram 8
- c. Interactions Design 9
 - i. Use Cases 9
 - ii. Sequence Diagram 10
 - iii. Activity Diagram / State / Processes 11
- d. Description of Algorithmic Components 13
- e. Software Architecture Pattern 13
 - i. N-tier: Data, Logic, Service, Presentation tiers etc. 13
 - ii. Optional: MVC- Model, View, Controller structure (or else) **Error! Bookmark not defined.**

4. Verification 13

- a. Validation and Evaluation Plan 13
- b. Testing Platform 13

5. Project Management 14

- a. Schedule / Gantt (possible print screen or sharable link) 14
- b. Team Roles - final 15

1. Introduction

a. System Overview

- ❖ Book Easy Buy System or BEBS is a software designed on a monitor in the store's shopping cart to "provide automation support" for the process of buying, searching, receiving book information and facilitating the actual sale.
- ❖ BEBS will provide the total price that the customer has to pay and will allow him to pay through it. Its will transfer the information of purchases on the database.
- ❖ BEBS will update the inventory online. Its will provide notifications of deficiencies and transfer the information on trade orders to a database. And give to manager a notifications for shortages or low-stock of a book in the store.

b. Purpose

- ❖ Improving service for customers
- ❖ Promotion
- ❖ Increasing profits
- ❖ Streamlining sales / buying work processes
- ❖ Iteratively manages inventory

c. Scope

- ❖ The BEBS System is on the monitor of cart in our book shop it will allow any user to create an account. The customer, through the process of account creation, will have the option to become a member of the shop. The system will allow to search, add, and pay books through it.
- ❖ The BEBS also allows a manager to manage the inventory with full create, retrieve, update and delete (CRUD) functionality with regards to books in the system.
- ❖ It will also allow, customers and managers to interact with a promotion system that handles percentage-off promotions that can be applied to member's orders. This interaction includes the creation (by managers) and send promotions to members of the shop.

d. Constraints

The problem BEBS try to solve:

- In the bookstore chain, it is not possible to keep track of the existing inventory in each store.
- Problem of write read to the data base with few update each time.
- Customers do not know what benefits exist in the store that are right for them.
- Customers are unable to keep track of the total price in their shopping cart
- Difficulty in continuously monitoring book sales made by customers
- Difficulty in continuously monitoring the store's profits by month / week

The solution:

- Customers receive a real-time price and catalog of the products in the basket, Customer receives real-time special offers.
- The system will allow the store manager to enter specials in real time and the system will display them to registered customers
- The system will help the store track book sales and in-store book shortages
- The system will alert store managers of the amount of books missing for order delivery from suppliers
- The system in real time will provide the user an efficient and fast search for books
- A manager can provide the customer with promotions
- The system will help keep track of store profits from book sales

2. System Architecture

a. Architectural Description and Design:

Figure 1 depicts the high-level system architecture. The system will be constructed from multiple distinct components:

Roles

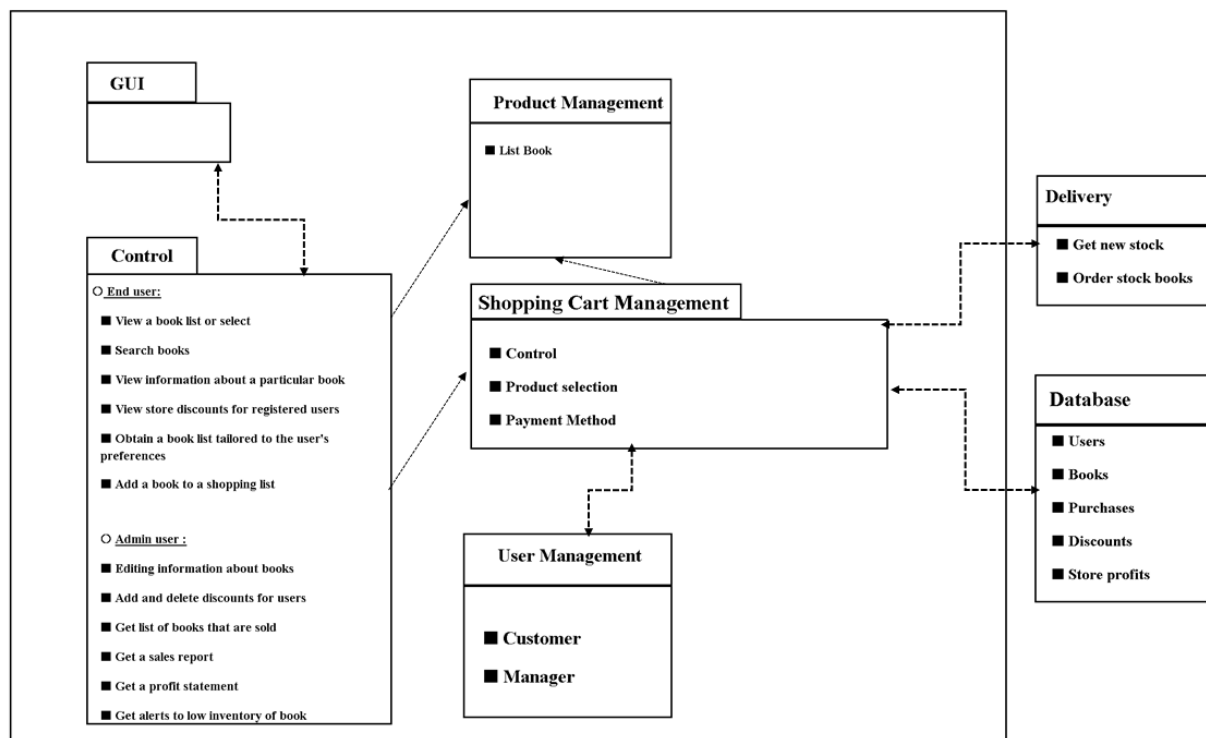
- Customers - are people who need services in the store. Registered customers have a user Account and profile, while anonymous customers do not have an account and can use System without connecting.
- Manager- needs to order new inventory for the store and provide discounts for the store and registering customers.
- Administrators are the "super users" who manage the BEBS software.

Activities

- Customer / Manager- Interface
Window interface for building, order or editing orders.
Searching book or checking available inventory.
Viewing and editing profiles or register.
- Interface inventory- for executing such as view, search ect.
- Wii Library- for maintaining Bluetooth connections to Wii input devices and generating input events.

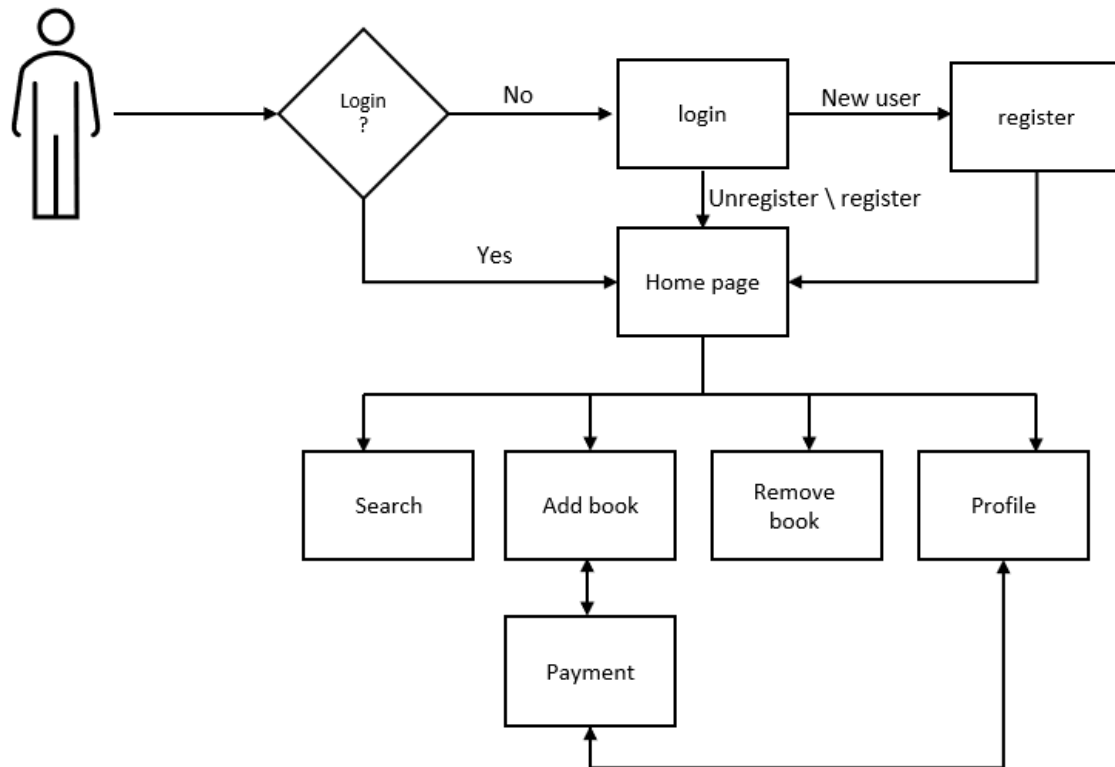
Data

- Data Model- The classification needed to organize orders/Purchases, Tasks, Profiles, profit, etc.
- Data Storage- The interface for storing, importing the data model and raw collected data.

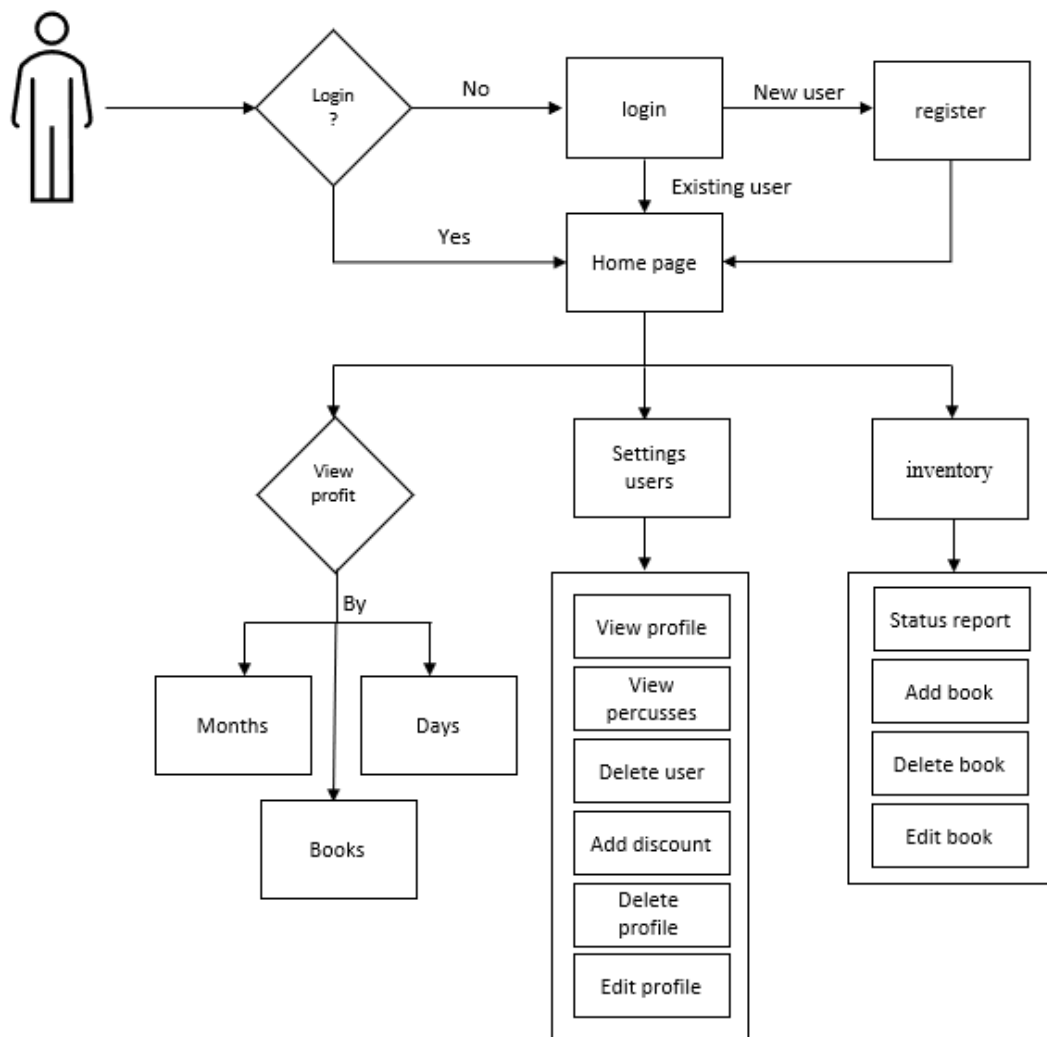


b. The Life Cycle of the System

- Customers

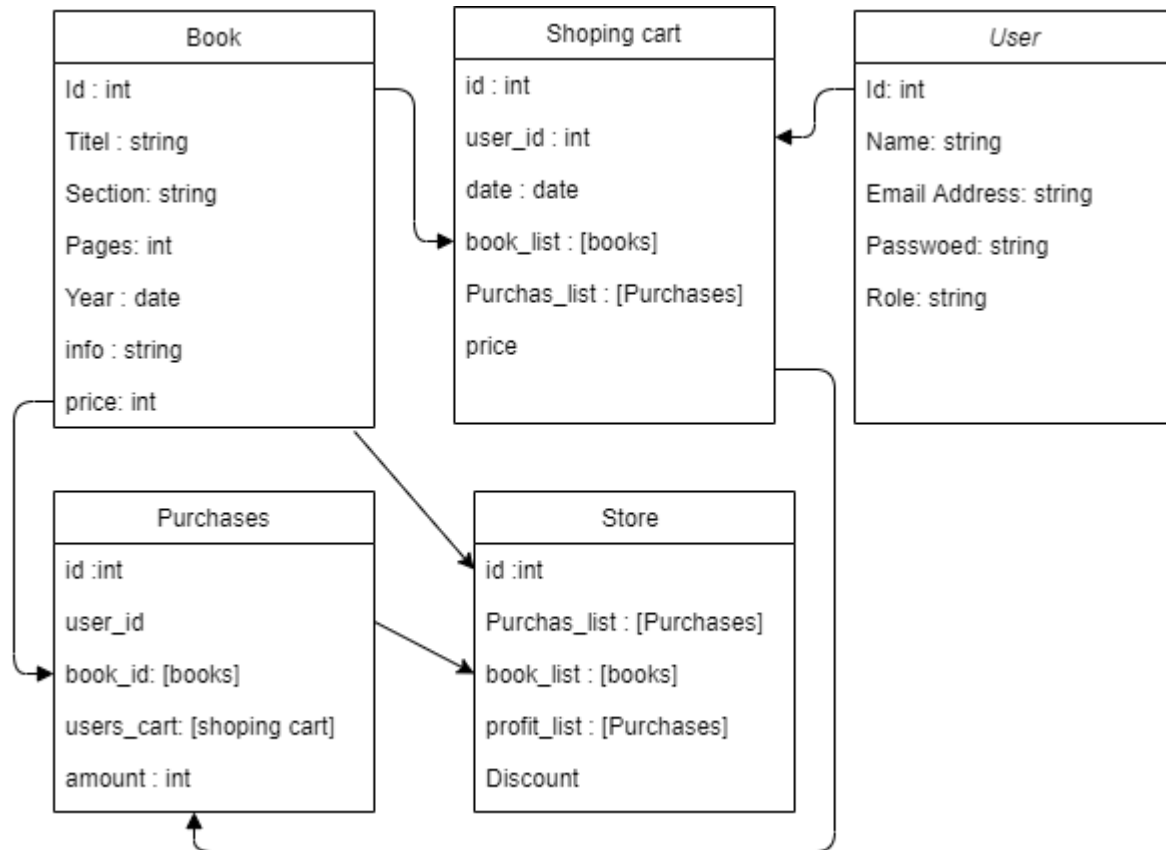


- System manager:

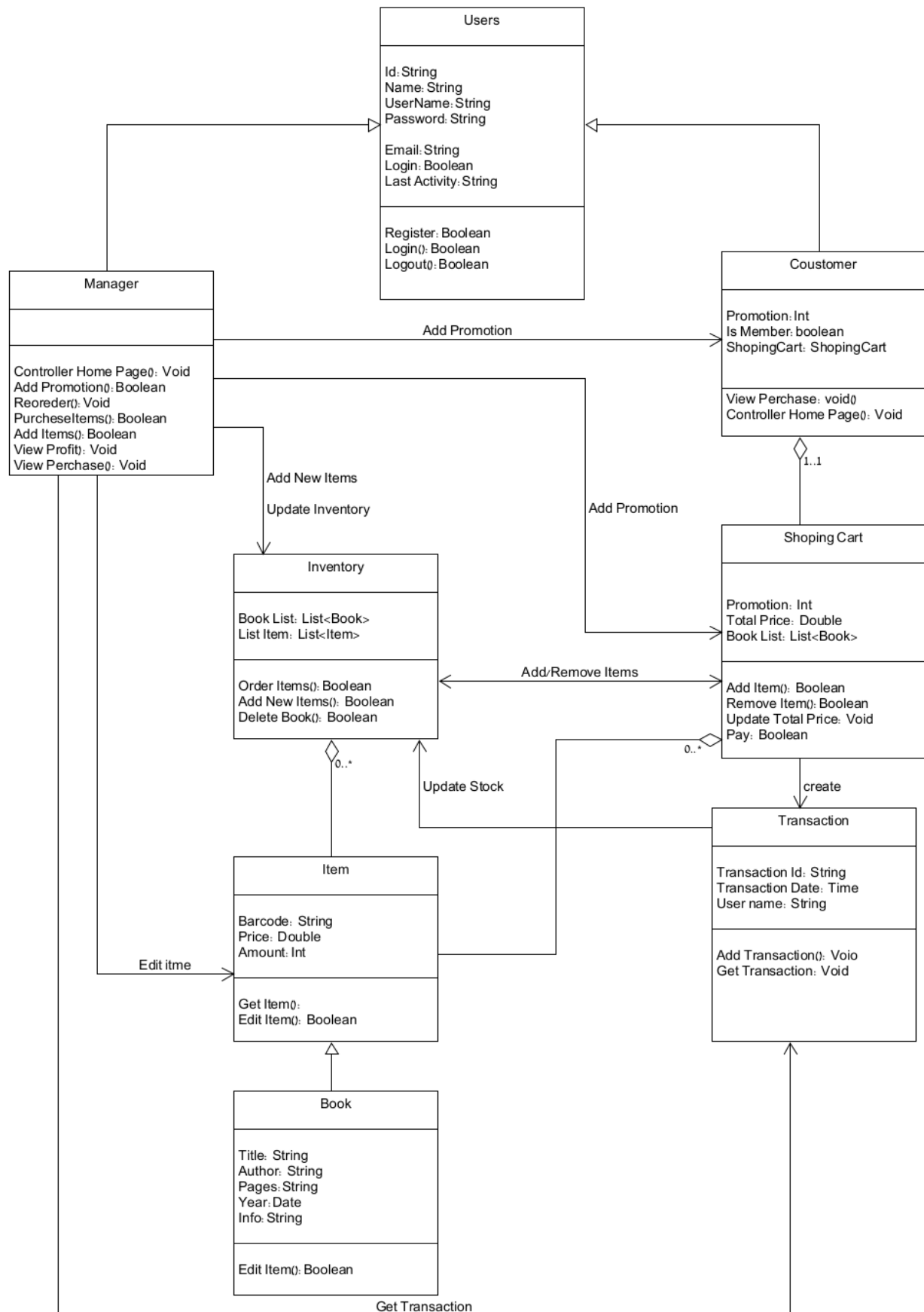


3. Design

a. Data Design - Database Description

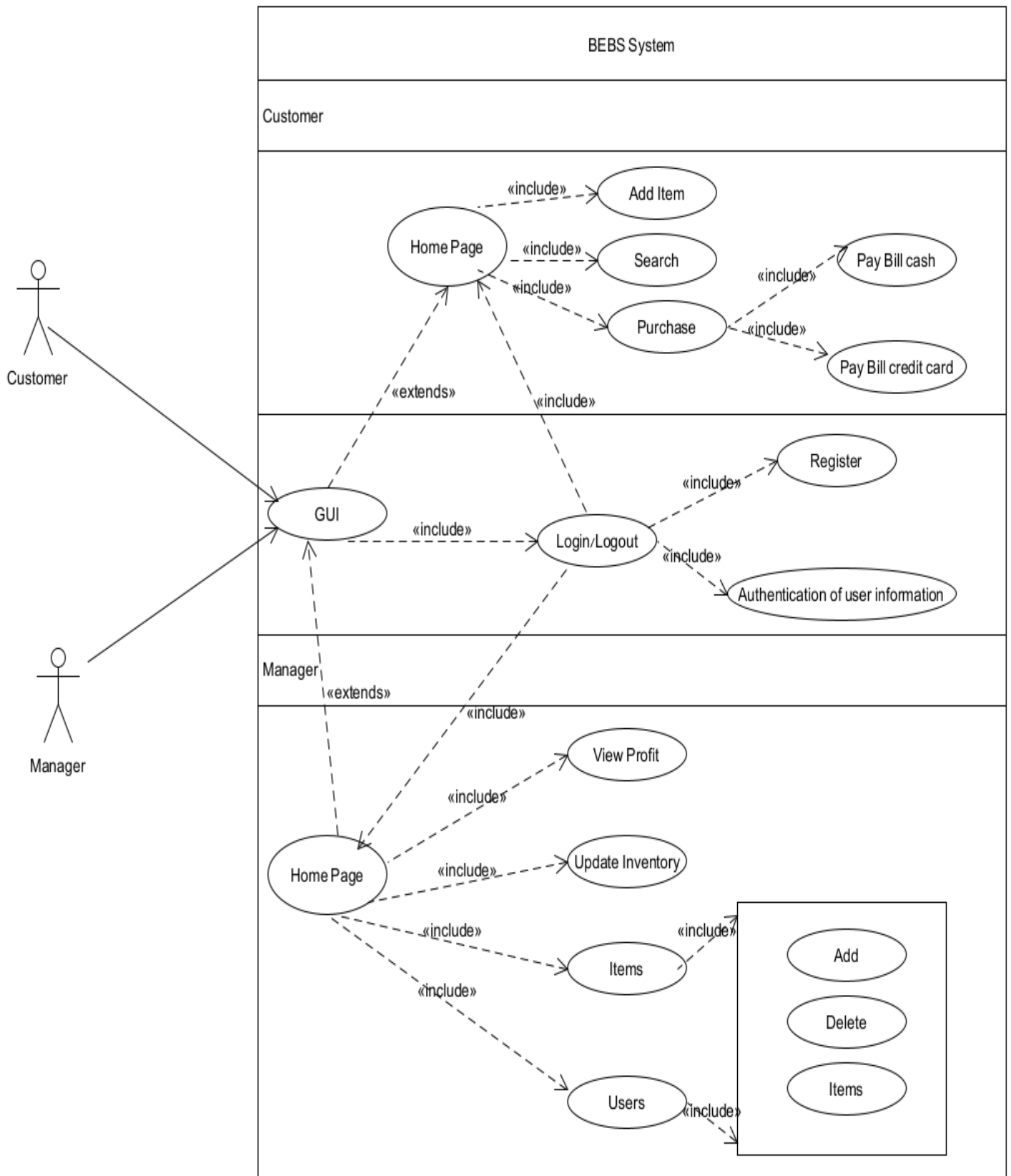


b. Structural Design - Class Diagram

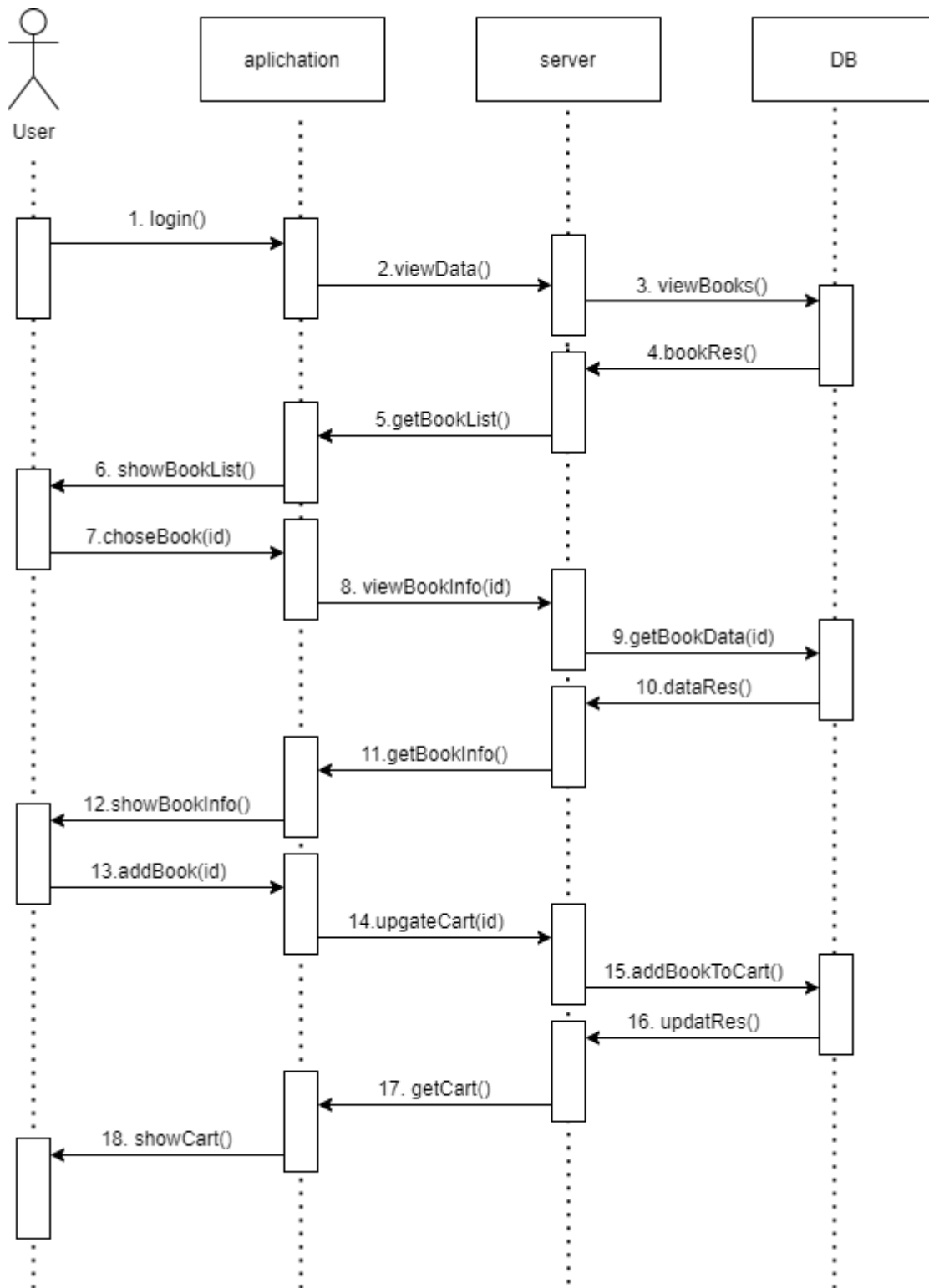


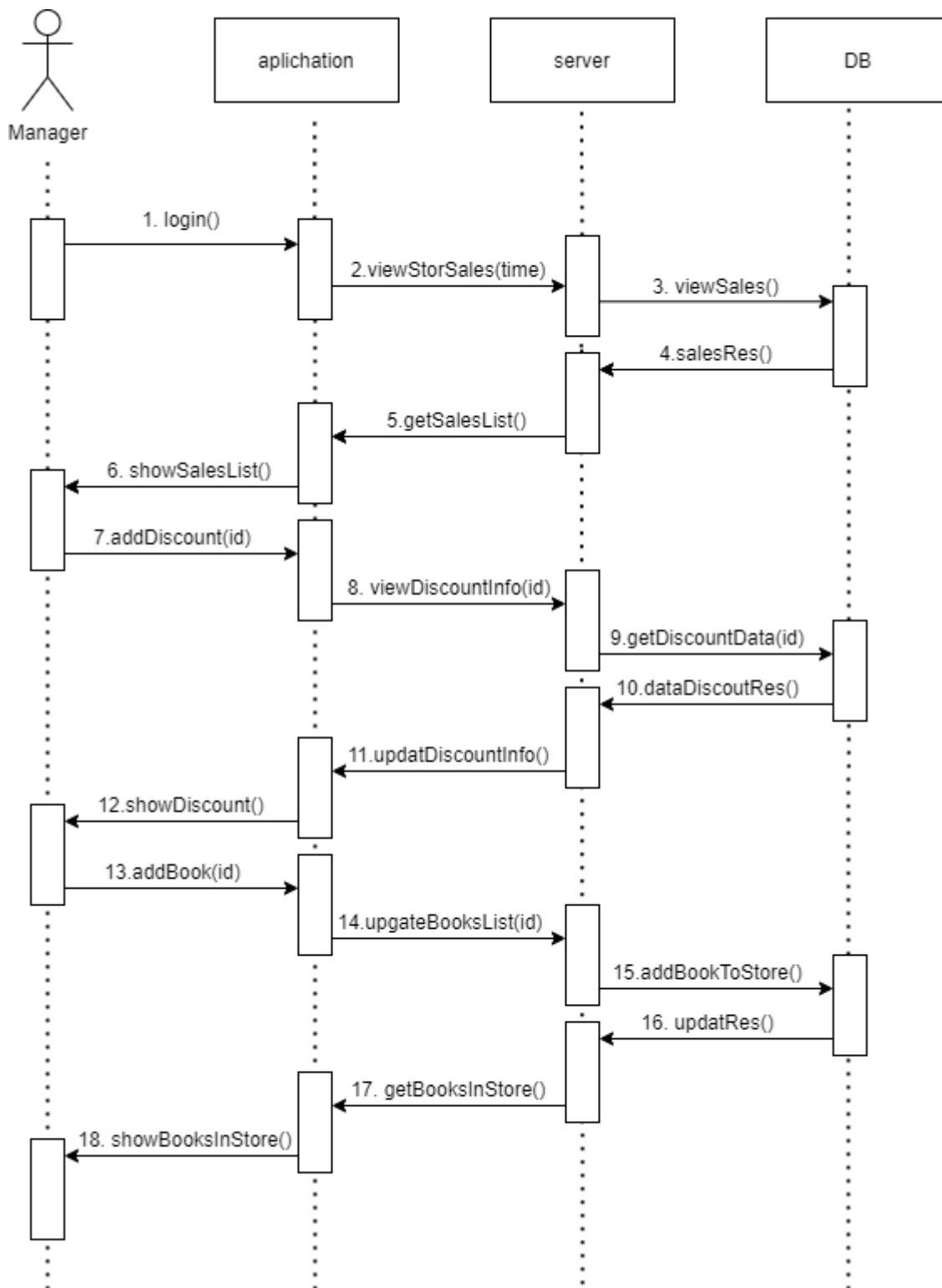
c. Interactions Design

- Use Cases

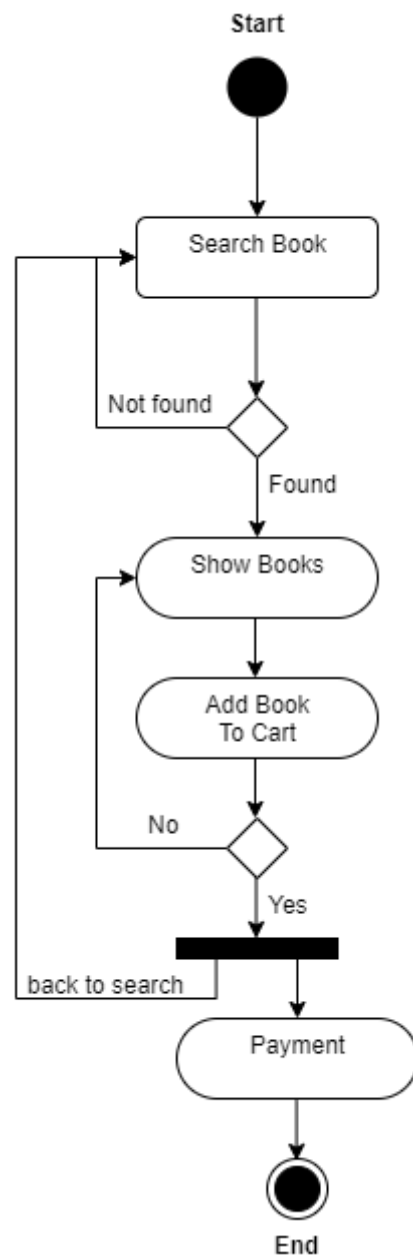


- Sequence Diagram





- Activity Diagram / State / Processes



d. Description of Algorithmic Components

- Search algorithm- Will search for books and users according to the customer's requirements and will search in the fastest time
- Bakery algorithm is a computer algorithm devised by computer scientist Leslie Lamport, as part of his long study of the formal correctness of concurrent systems, which is intended to improve the safety in the usage of shared resources among multiple threads by means of mutual exclusion.

e. Software Architecture Pattern

- N-tier: Data, Logic, Service, Presentation tiers etc.
The MVC pattern provides clean object separation to:
Data retention models,
Views to display all or part of the data, and
Controllers for handling events that affect the model or display.

4. Verification

a. Validation and Evaluation Plan

- feedback's delay to the user We'll need to make sure the user gets the correct updates for store discounts
- according to system administrator changes

b. Testing Platform

- **Debugging environment**
For management and tracking of system bugs we will use GitHub.
- **Debugging tool:**
We will use the Visual Studio system that supports the passage of the code in the debugging
- **User Testing:**
We will perform a QA test based on user experiences using the system