

COE 528 PROJECT REPORT

UML Use Case Diagram

The UML diagram in our case describes the functionality and features available to both types of users i.e., Owner and Customer.

Use case name: Login

Participating actors: Owner and Customer

Flow of Events:

1. Customer / Owner Login's Home Page option is required to be chosen by the user.
2. The credentials are authenticated based on Customer or Owner's Login.
3. Owner can Create or Delete the Customer and Add or Delete a new book.
4. Creating the Customer, creates a text file for that customer with unique username, password and points as entered by the Owner. 0 minimum point is required to create a customer.
5. Deleting a Customer, deletes the text file for the customer associated with the unique username.
6. The Owner can now log out from the system.
7. Customers can login to the application using username and password, which is authenticated using the text file that was created by Owner.
8. The Customer can check and redeem their Points and they can buy a new book. The text file for the specific customer is used or updated whenever these features are used.
9. A Customer can logout, after the changes have been made.

Entry Condition: Owner or User Logs into the application, their credentials are verified.

Exit Condition: Owner or Customer Logs out of the application.

Quality Requirements:

1. A customer can be created with a minimum 0 points in account.
2. A customer can buy a new book through 2 options: Either redeem the points or buy it online.

UML CLASS DIAGRAM

At first, the person is asked to choose a login portal i.e., Owner or Customer Login which is handled by the “UserState” class. The “OwnerState” and “customerState” classes are associated with the User, as they are used by the user to login and access the features based on their role. The given classes let the Owner add or remove customers by creating and deleting text files, storing information such as username, password, points, checking the book price and title and logging new books. The “customerState” is a customer’s portal, it lets the customer login and lets them use different features such as redeem and add points, check their book status and its price by using information from the text file.

STATE DESIGN PATTERN

According to our UML diagram, the state design pattern is implemented for the “customerState” class. This class has a composition relationship with the “userState” interface which implements “Gold” and “Silver” classes. The points that a customer has determines the state of the customer. We used a State design pattern as it helps in changing the states without affecting other states. It promotes code reuse by allowing the same states to be used across multiple states. If a customer has points more than 1000 then he/she is considered to be a “Gold member” and if the points are below 1000 then they are in the category of “Silver member”.