# ULTRA VISION

# MUSIC & VIDEO RENTAL MANAGEMENT SYSTEM

# **Problem Definition**

Ultra-Vision is a music and video rental shop that rents Music, Music Videos, Movies and TV Boxets.

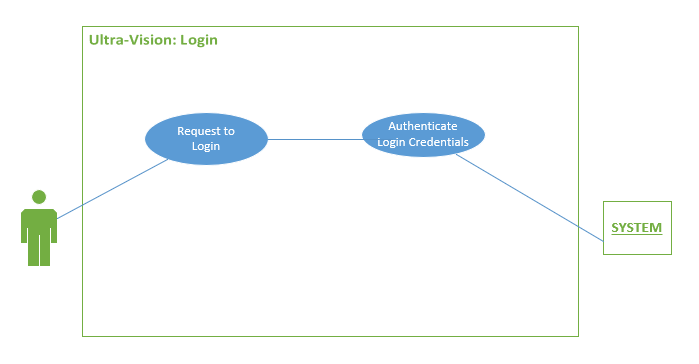
The shop is planning on opening more outlets located throughout Ireland and so, is in need of a computerized inventory system to track all the titles present in the store in order of categories, all the customers of the store along with their subscription plans and all the rental transaction logs. All of which was previously being done through the conventional paper based inventory tracking system

# **List of Requirements that the system fulfill**

The system designed fulfills all the requirements mentioned above, it provides the following functionalities:

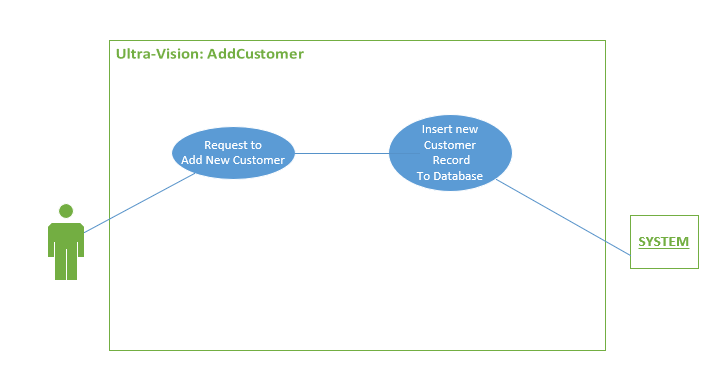
* Add new Customer
* Edit existing Customer
* Search for existing Customers
* Add new Title
* Search for existing Titles
* Create a rental transaction log
* Display rental history
* Update rental log status

# **Usecase Description**

1. **UC\_Login** 

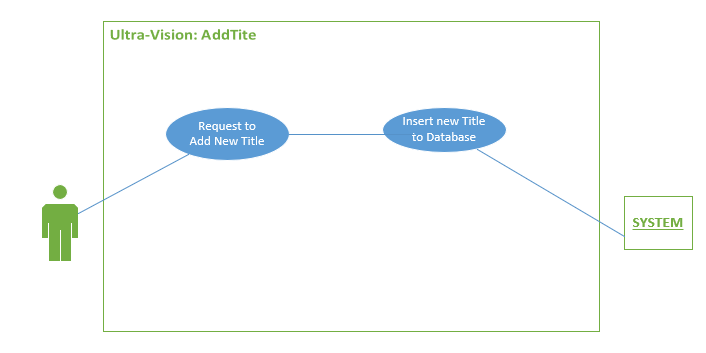
|  |  |
| --- | --- |
| **Brief Description** | This use case is started by the System. It provides verification of the details that the user enters at time of login to enhance security. |
| **Pre Conditions** | 1. Suitable connection to the Internet |
| **Basic Flow** | 1. User enter his/her credentials 2. User Logs into the system   2.1. Verification of username and password done by the System |
| **Alternative Flow** | 1. **2AF1**: User had entered wrong credentials. 2. **2AF2**: Go to Step 1 |
| **Post Conditions** | User is redirected to homepage |

1. **UC\_AddCustomer**



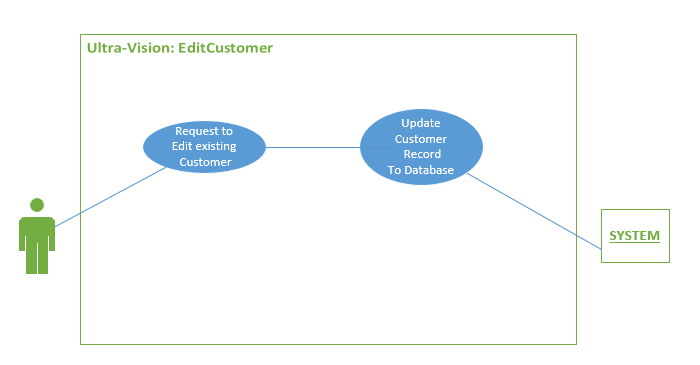
|  |  |
| --- | --- |
| **Brief Description** | This use case is started by the User. It requests the system to add a new customer to the database with the entered data as its attributes. |
| **Pre Conditions** | 1. User must be logged in. |
| **Basic Flow** | 1. User Selects the Add customer option from the homepage menu 2. Enters the required details needed for a customer to be registered 3. User is shown a message that the customer is added successfully to the database. |
| **Alternative Flow** | **3AF1**: Connection to the database is lost and the customer is not registered.  **3AF2**: Go to Step 1. |
| **Post Conditions** | User is redirected to homepage |

1. **UC\_AddTitle**



|  |  |
| --- | --- |
| **Brief Description** | This use case is started by the User. It requests the system to add a new title to the database with the entered data as its attributes. |
| **Pre Conditions** | 1. User must be logged in. |
| **Basic Flow** | 1. User Selects the Add Title option from the homepage menu 2. Enters the required details needed for a title to be registered 3. User is shown a message that the title is added successfully added to the database. |
| **Alternative Flow** | **3AF1**: Connection to the database is lost and the title is not registered.  **3AF2**: Go to Step 1. |
| **Post Conditions** | User is redirected to homepage |

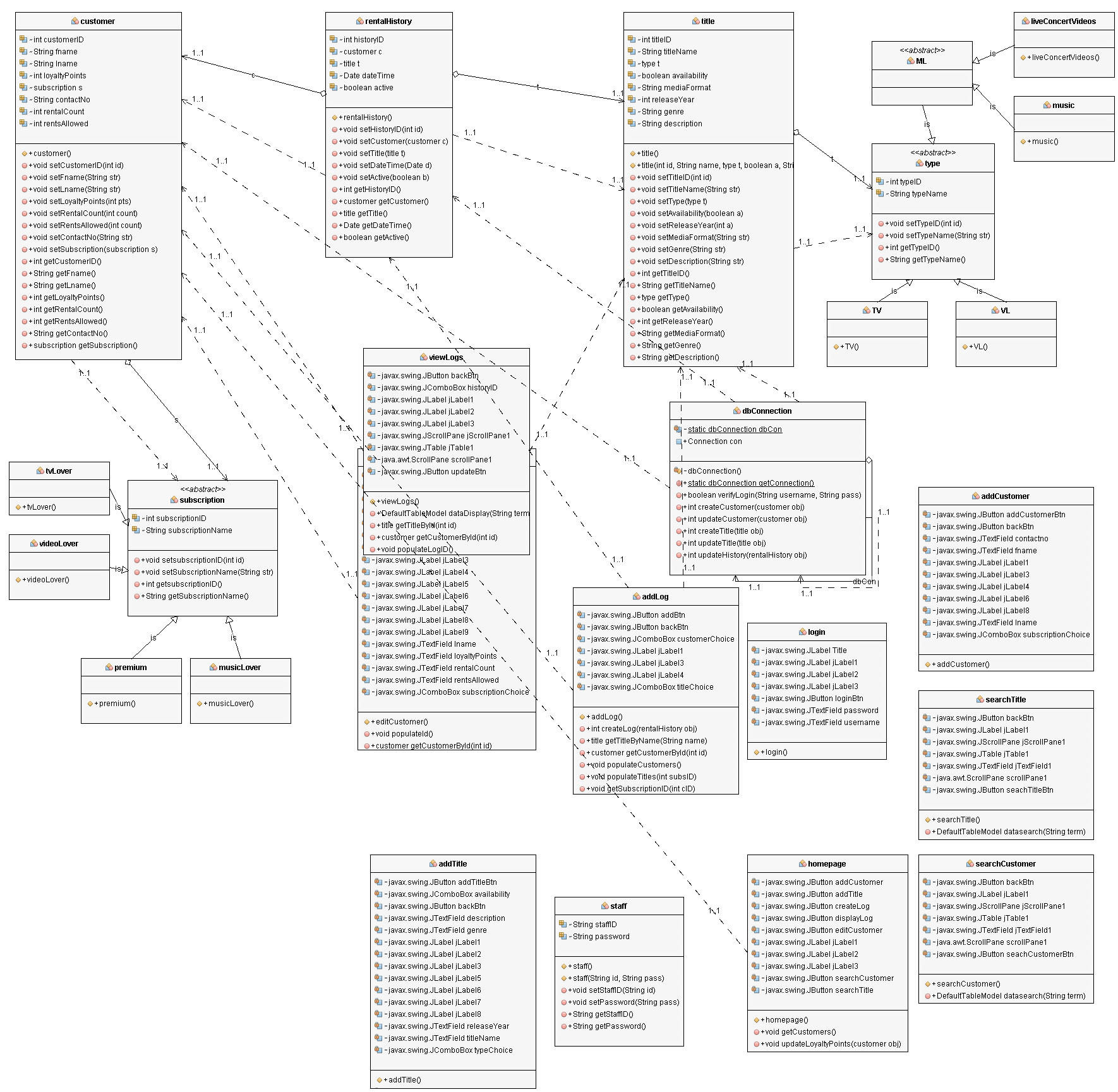
1. **UC\_EditCustomer**

****

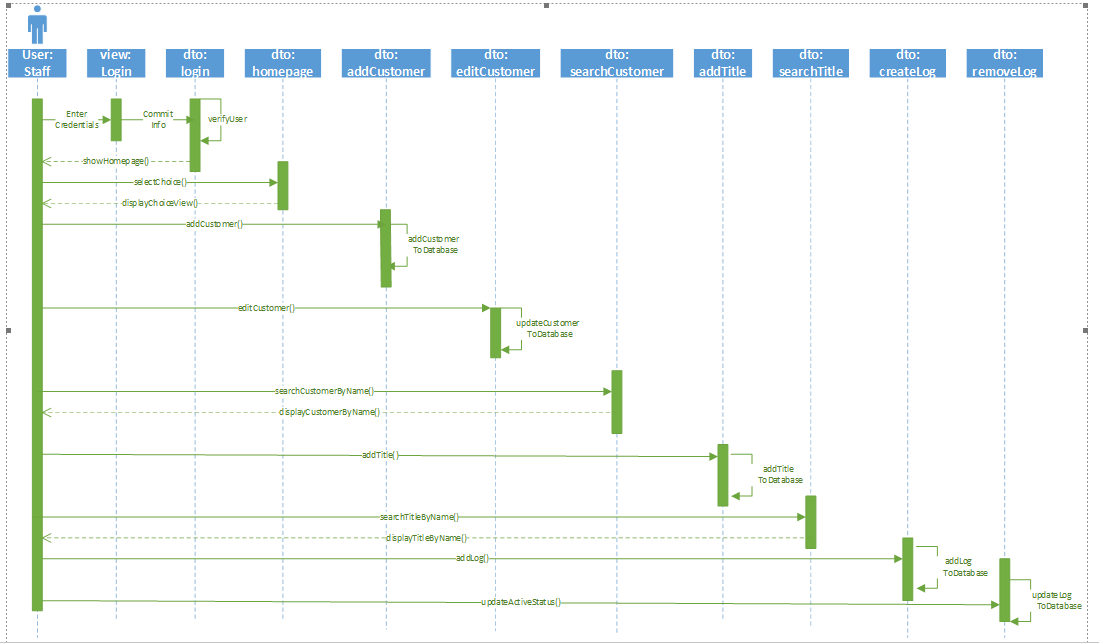
|  |  |
| --- | --- |
| **Brief Description** | This use case is started by the User. It requests the system to edit/update one or more attributes of an existing customer to the database. |
| **Pre Conditions** | 1. User must be logged in.  2. Customer must be already registered in the system. |
| **Basic Flow** | 1. User Selects the Edit customer option from the homepage menu 2. Updates at least one of the customer attributes. 3. User is show a message that the customer is updated successfully. |
| **Alternative Flow** | **3AF1**: Connection to the database is lost and the customer is not registered.  **3AF2**: Go to Step 1. |
| **Post Conditions** | User is redirected to homepage |

# **UML: CLASS DIAGRAM**

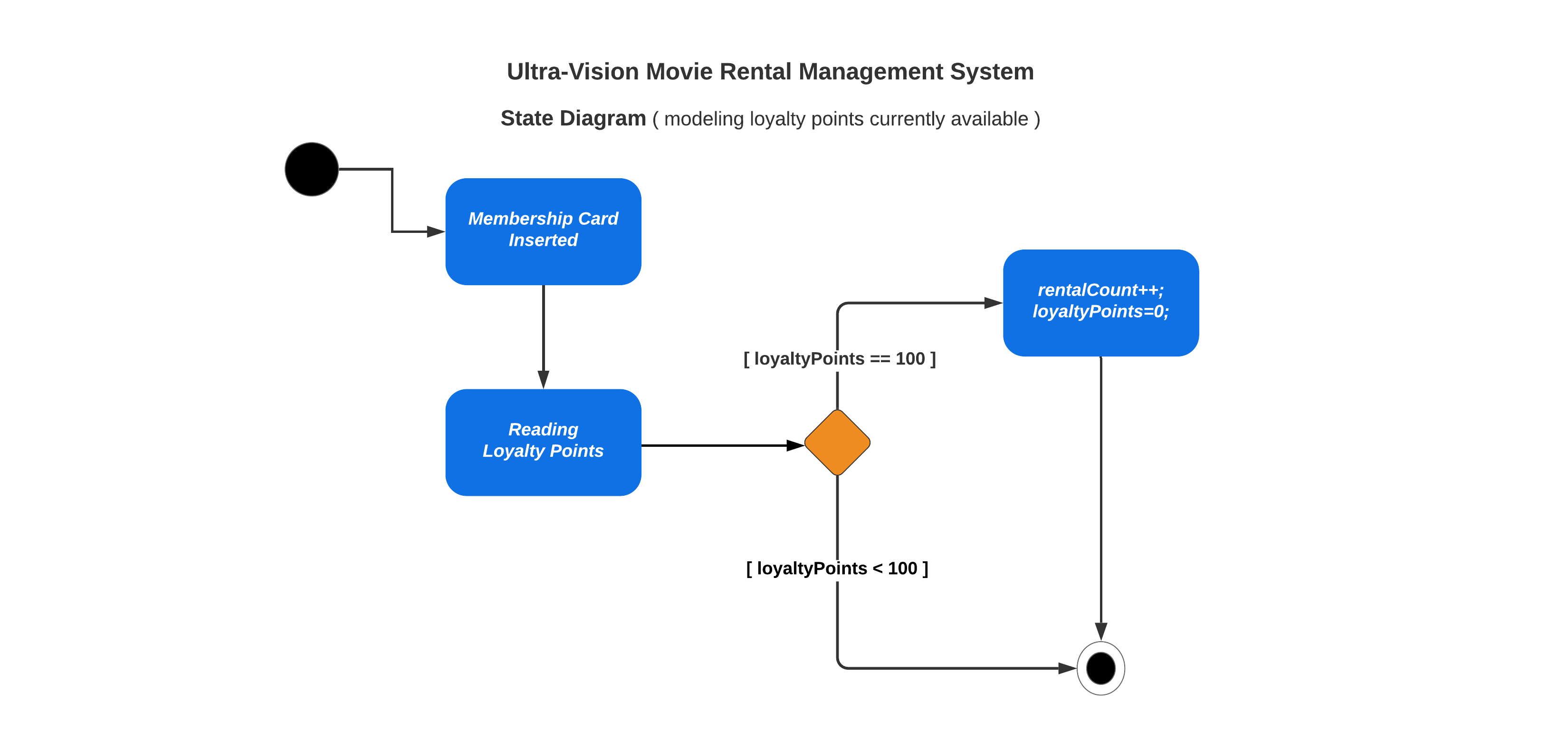
NOTE: Kindly view this in zoomed in form

****

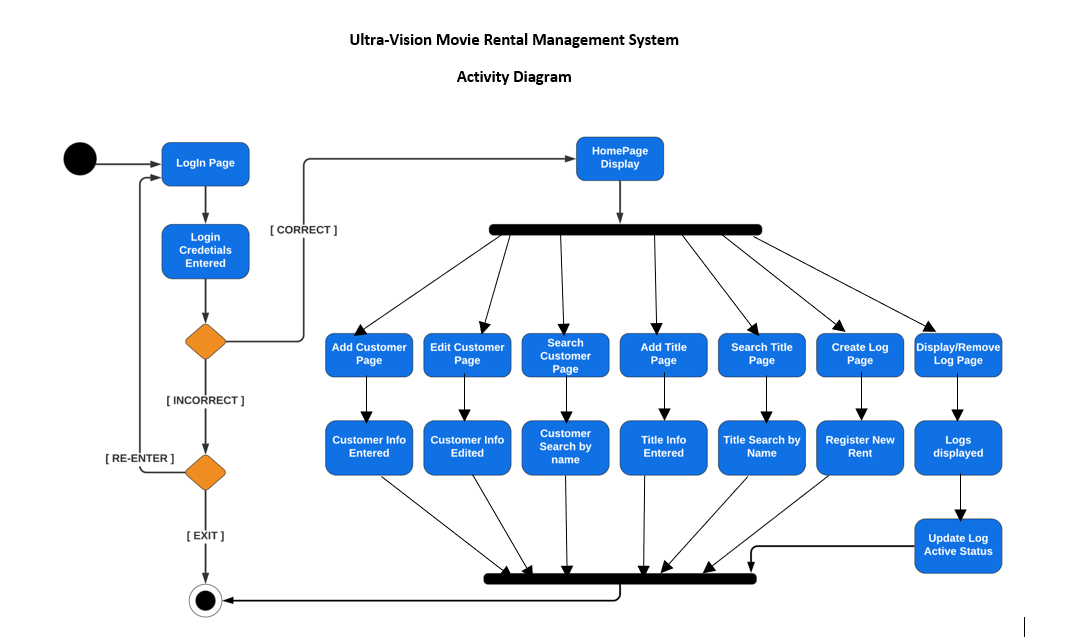
# **UML: Sequence Diagram**



# **UML: STATE DIAGRAM**



# **UML: ACTIVITY DIAGRAM**



# **Task division among the two team mates**

**TEAMMATE #1:**

|  |  |
| --- | --- |
| **CODE** | All functionalities related to customer and login were done by teammate#1.  Java files namely:   * ADD CUSTOMER * EDIT CUSTOMER * SEARCH CUSTOMER * LOGIN |
| **UI** | All the JFrame UI related to above controller were designed by teammate#1 |
| **SYSTEM ANALYSIS & DESIGN** | Class, State and 2 Use Case diagrams were made by teammate#1 |
| **DATABASE** | DataBase creation on phpMyAdmin were done by teammate#1 |
| **DOCUMENTATION** | Documentation was a collective task |

**TEAMMATE #2:**

|  |  |
| --- | --- |
| **CODE** | All functionalities related to title and rental logs were done by teammate#2.  Java files namely:   * ADD TITLE * SEARCH CUSTOMER * CREATE LOG * VIEW/UPDATE LOGS |
| **UI** | All the JFrame UI related to above controller were designed by teammate#2 |
| **SYSTEM ANALYSIS & DESIGN** | Activity, Sequence and 2 Use Case diagrams were made by teammate#2 |
| **DATABASE** | DataBase connection establishment through the code were done by teammate#2 |
| **DOCUMENTATION** | Documentation was a collective task |