

Movie Rental Store Design

Relationship between components

- Comedy Drama Classical is **movie**
- Movie Rental Store HAS A (Composition) **movies & customers**
- Customer HAS A (Composition) **transactionHistory**

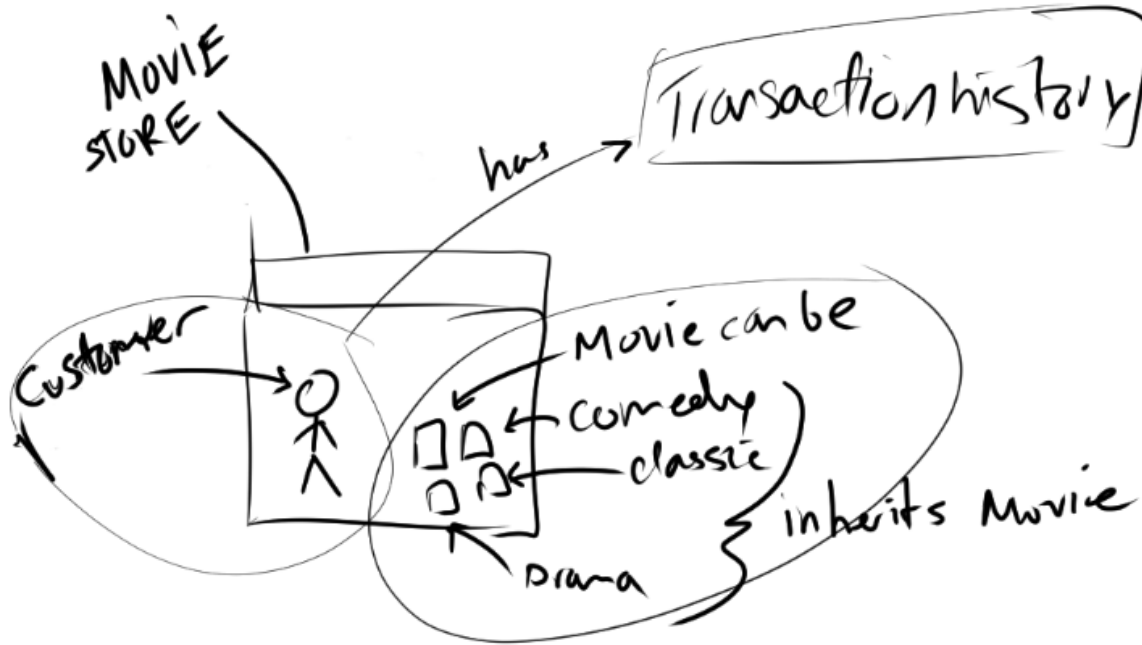


Figure 1. Brief drawing of underlying relationship

UML Diagram

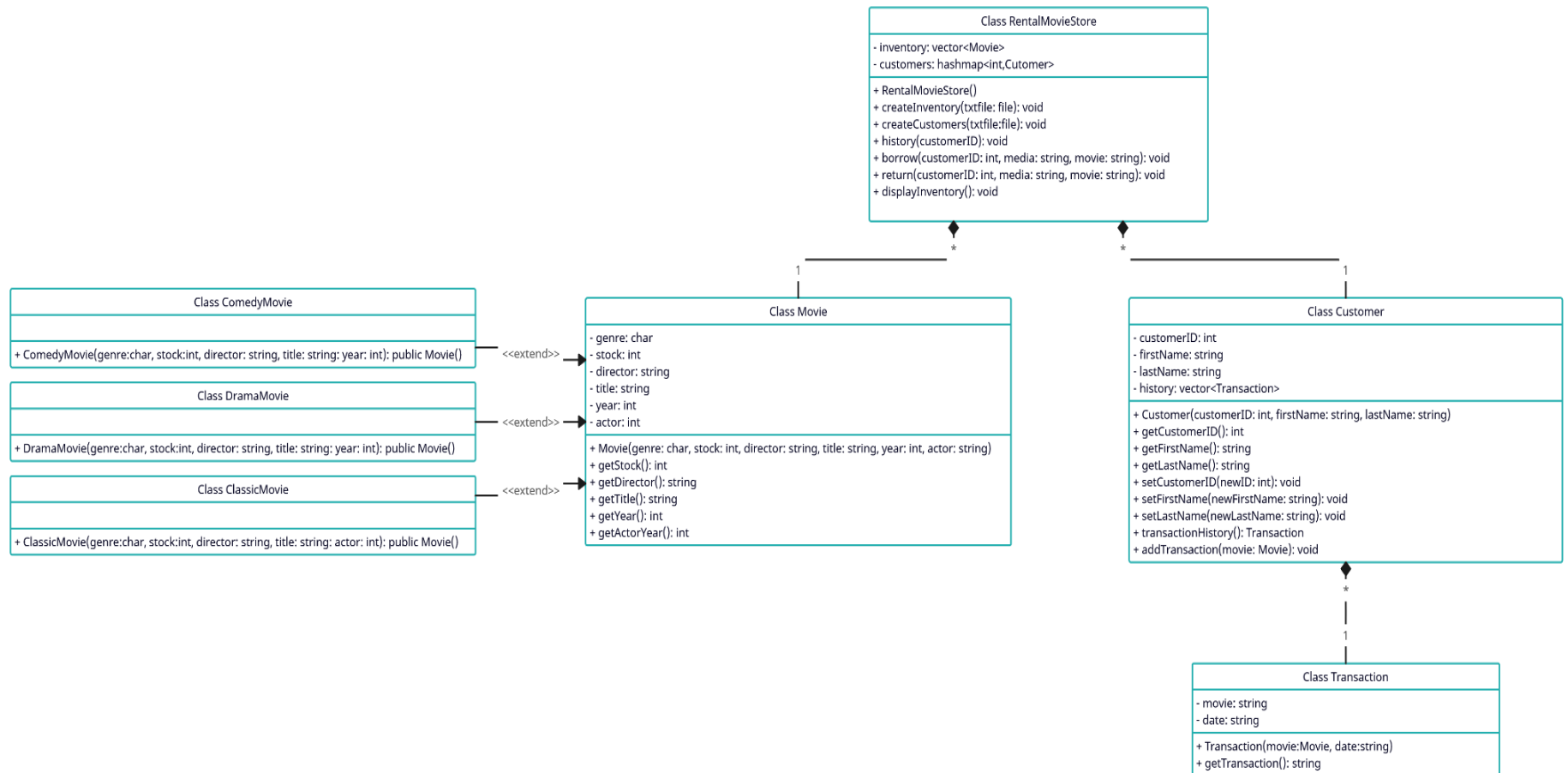


Figure 2. UML Movie Rental Store

Implementation

- **Movie Rental Store**

- Private Members**

- std::vector of Movie // vector of the Movie object
 - A hash map that maps customers ID to customer Object // hashmap stores the customersID to the customer object(customerID, customer Object) -> (key, values)

- Methods**

- void createInventory(txt file) // takes in file and creates inventor story out of it
 - void createCustomers(txt file) // takes in file and makes customers
 - void history(customerID) // history method
 - void borrow(customerID, typeOfMedia, movie type, movie data) // borrow method
 - void return(customerID, typeOfMedia, movie type, movie data) // return method
 - void displayInventory() // displays the total inventory

- **class Movie**

- Private Members**

- char genre // Holds 'F', 'C', 'D'
 - int stock (How many there are left of this movie) // the integer var for stock
 - std::string director // the string variable of director
 - std:: string title // the title for the string
 - int year it released (if it is not a classical movie) // integer that stores the year
 - int major actor release date (if it is a classical movie) // the major actor int
 - virtual Movie(genre, stock, director, title, year it released, major actor) // movie object

- Getter**

- Int getStock() // getter for the stock
 - std::string getDirector() // getter for the Director string
 - std::string getTitle() // getter for the Title
 - int getYear() // the integer for the year
 - Int getActorYear() // getter for the actor's year

- **class ComedyMovie : public Movie** //inherits from Movie class

- Movie(genre, stock, director, title, year it released)

- **class DramaMovie : public Movie** //inherits from Movie class

- Movie(genre, stock, director, title, year it released)

- **class ClassicMovie : public Movie** //inherits from Movie class

- Movie(genre, stock, director, title, year it released, major actor)

- **Customers**

- Private Members**

- int CustomerID // the integer for customer ID
 - std::string firstName // the std for string firstName
 - std::string lastName // the std string for the lastName
 - **vector<Transaction>** History // vector that stores the history
 - Construtor(customerID, firstName, lastName) // Constructor for customer class

- Getters for Customers**

- int getCustomerID() // getter for the customer
 - std::string getfirstName() // getter for the firstName
 - std::string getlastName() // getter for the lastName

- Setters**

- void setCustomerID(int customerID) // setter for the customerID
 - void setfirstName(string firstName) // setter for the firstName
 - void setLastName(string lastName) // setter for the lastName

- Methods**

- void transactionHistory(this->history) // the transaction history constructor
 - friend void Transaction::addtransaction() // adds the transaction to the Transaction class

- **class Transaction**

- Private members**

- std::string movie // movie string variable
 - std::string date // string variable for date

- Methods**

- void addTransaction(movie, date) // method that adds transaction bodies of movie and date