Online Theater Ticketing Software Design Specification

By: Angelina Mom, Brandon Slusser, Vincent Huynh, and Conor Murphy

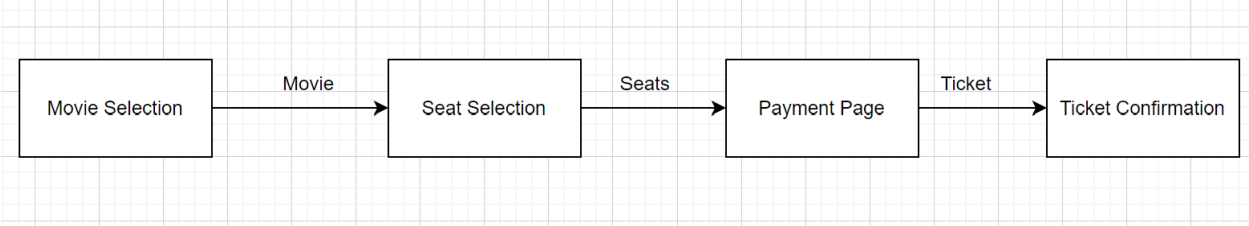
https://github.com/amom1053/Software-Design-Specification

(Please commit any changes onto this repository)

System Description (Brief overview of system)

The purpose of this system will be to choose a movie, then pick out a seat, and to pay for the ticket. The system will be intended for movie theaters to have the customers go through the entire process of getting their tickets online. In this system we will be building classes with attributes along with operations in order for a customer to pick out and pay for a ticket in an efficient manner. The steps for the entire process will follow a straight line with the selection of the movie directly moving to the seat choice which will go to the payment process which will finish with a ticket confirmation.

Architectural diagram



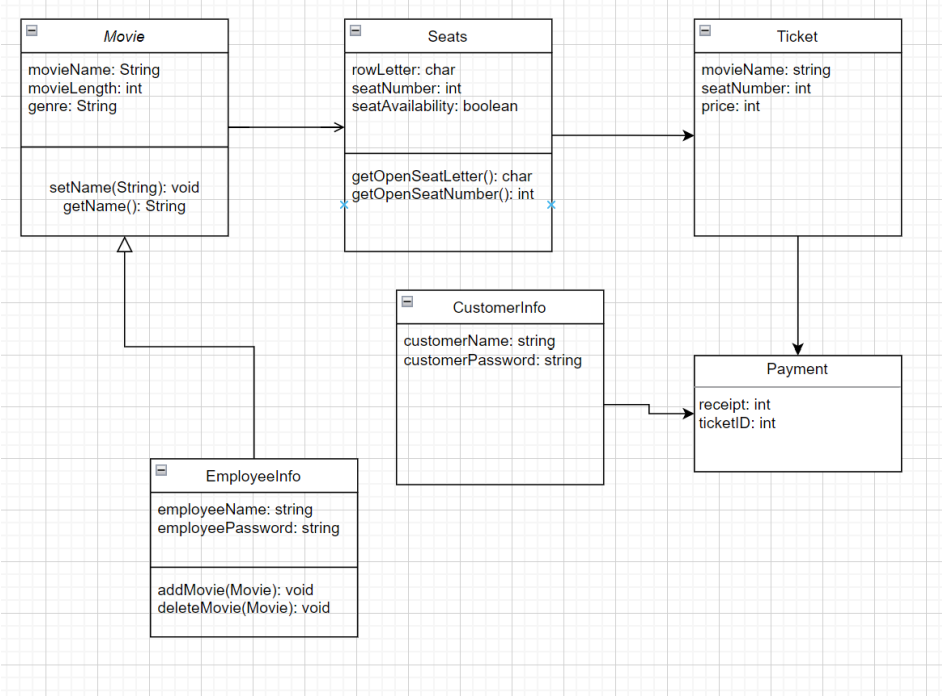
https://drive.google.com/file/d/1GczHMnxtBrQLPHaztREFe_aUGopSOfZG/view?usp=sharing

Architectural Diagram Description

The architectural diagram shows how a normal customer would view the system. All customers would start at the movie selection page. After they choose a movie it would move on to the seat selection page so they can choose their seat. After choosing a seat it would show the payment page. After purchasing the ticket, they will receive a ticket confirmation page. A customer can

not access the seat, payment, and ticket confirmation page without first clicking on the movie selection page. Hence, why the diagram is linear.

UML Class Diagram



https://drive.google.com/file/d/14T24Y2ZmfYA9v__EwmtIC1d0nvIHJule/view?usp=sharing

Description of classes:

Movie Class:

-Parent Class to Employee Info class, derived from Seats class

-Has three variables: movieName(String), movieLength(int), and genre(String)

-Has two functions setName(String) and getName()

Seats Class:

-Parent class to Movie class, derived from Ticket class

-Has three variables: rowLetter(char), seatNumber(int), and seatAvailabiltiy(boolean)

-Has two functions getOpenSeatLetter() and getOpenSeatNumber()

Ticket Class:

-Parent class to Seats class, derived from Payment class

-Has three variable: movieName(String), seatNumber(int), price(int)

Payment Class:

-Parent class to Ticket Class and Customer Info Class

-Has two variables: receipt(int) and ticketID(int)

Customer Info Class

-Derived from Payment class

-Has two variables: customerName(String) and customerPassword(String)

Employee Info Class:

-Derived from Movie Class

-Has two variables: employeeName(String) and employeePassword(String)

-Has two functions: addMovie() and deleteMovie()

Description of attributes:

Movie Class:

- movieName this is what the name of the movie is and it is a String data type.
- movieLength this is how long the movie is and it is represented as an integer.
- genre this is what genre the movie belongs to and it is a String.

Seats Class:

- rowLetter is a character data type that is used to hold the row letter that a given seat is in.
- seatNumber this is the seat number for a given seat and it is represented as an integer.
- seatAvailability this attribute determines whether a seat is available or not and it is a boolean data type.

Ticket Class:

- movieName from the movie class this attribute says what the name of the movie is and is a string data type.
- seatNumber from the seats class this attribute says what the seatNumber is for a given seat and is an integer.
- price this is an integer data type that says what the price is.

Payment Class:

- receipt this attribute is an integer and has the receipt for the payment.
- ticketID is an integer data type that holds the ticket identification.

Customer Info Class

- customerName this attribute is a string that holds the name of the customer.
- customerPasswords holds the customer's password and is represented as a string data type.

Employee Info Class:

- employeeName this attribute holds the name of the employee and is a string data type.
- employeePassword is a string that holds the password for an employee.

Description of operations

Movie Class:

setName(String): void – Sets the variable MovieName based on the user input from getName()

getName(): String –Gets the name of the movie from the user.

Seats Class:

getOpenSeatLetter(): char -- Returns an open seat letter from the user.

getOpenSeatNumber(): int -- Returns an open seat number from the user

Employee Info Class:

addMovie(Movie): void – Takes movie as a parameter, and adds it

deleteMovie(Movie): void – Takes movie as a parameter, and deletes it

* descriptions should be detailed and specify datatypes, function interfaces, parameters, etc..

Development plan and timeline

Angelina Mom: Architectural Diagram, UML Class Diagram, and Architectural Diagram Description

Brandon Slusser: Description of Attributes

Vincent Huynh: System Description

Conor Murphy: Class Description and Attributes of Operations