Online Theater Ticketing Software Design Specification

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

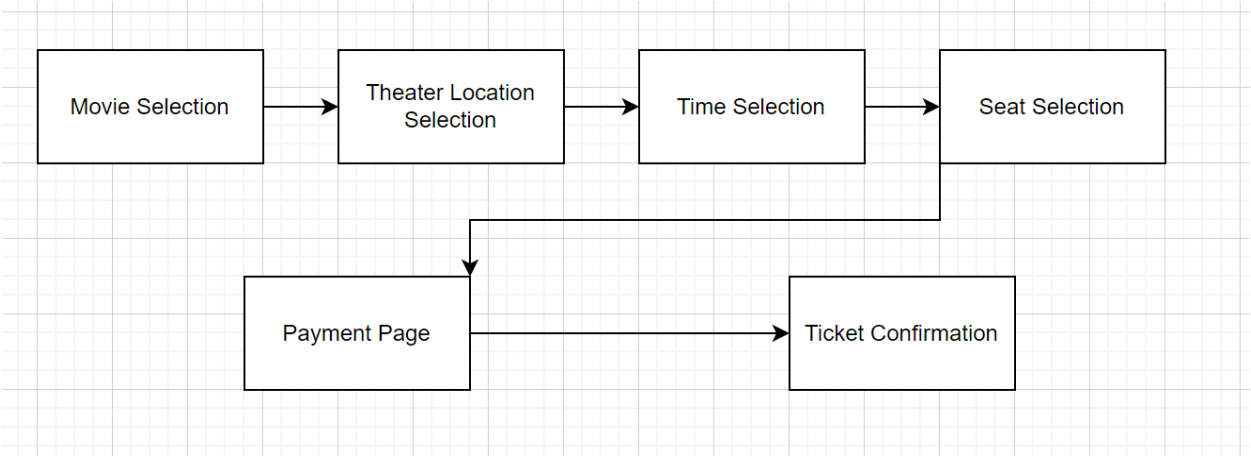
https://github.com/amom1053/Software-Design-Specification

(Please commit any changes onto this repository)

System Description (Brief overview of system)

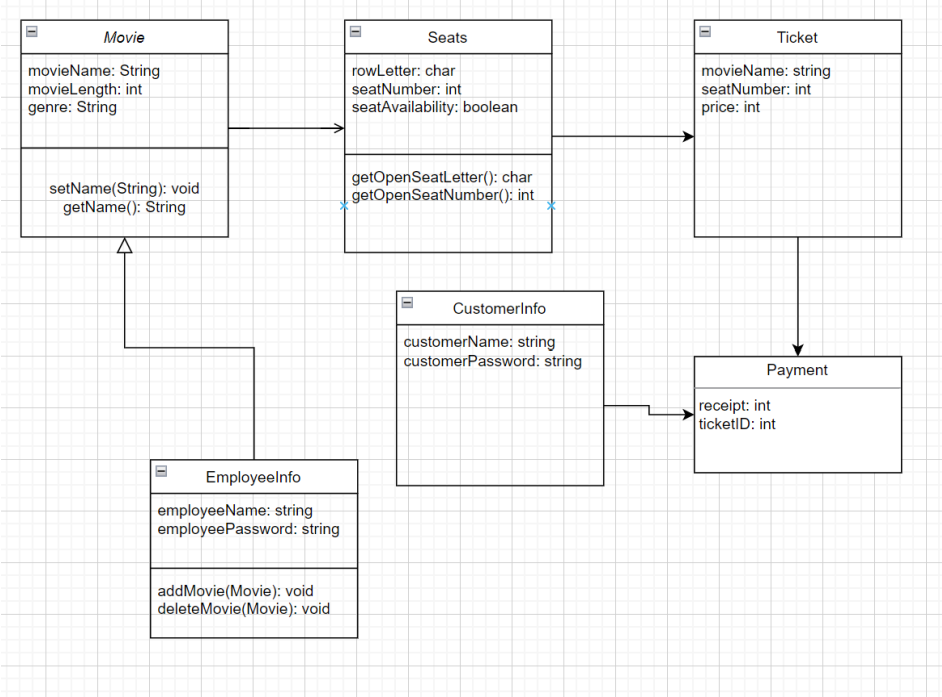
Software Architecture Overview

Architectural diagram



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

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:

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

- Partitioning of tasks
- Team member responsibilities

Angelina Mom: Architectural Diagram & UML Class Diagram

Brandon Slusser:

Vincent Huynh:

Conor Murphy: