Movie Theater Ticketing System

Software Requirements Specification
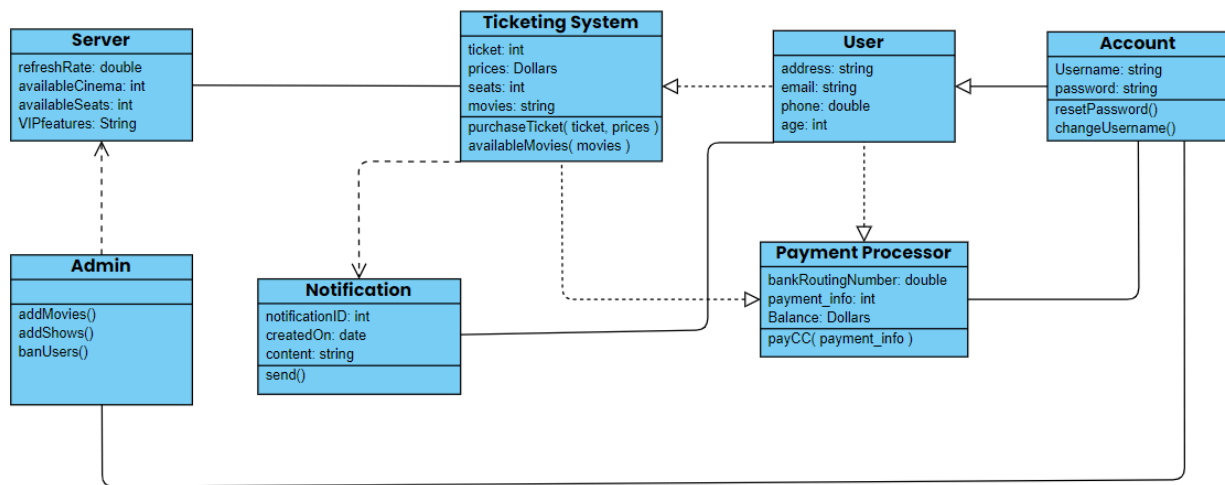
Version 1.2

02/14/2024

Group 22
Fernando Quintana

System Description:

An online movie ticketing system is a software hosted within an online server that allows users to purchase a ticket in order to see a film inside a designated theater location. It is used to give real time feedback to show what films are currently available, the time slots that those films can be seen and the different locations that have scheduled for those movies. This is done with a designated app and/or website that sends data packages to the user to see the provided information within an appropriate User Interface.

Software Architecture Overview

UML Class Diagram

Class: Server
Description of Class: The main system that gives back information provided on films and other data.

Attributes:

- refreshRate: double - how fast data can be provided about specific topics.
- availableCinema: int- how many theaters will carry specific films.
- password: string - stores the password of the user's account.
- VIPfeatures: string - info on what premium status gives to the users.

Operations: None as most of the Attributes are called from here.

Class: Ticketing System
Description of Class: The system that manages the prices and location for the seats.

Attributes:

- ticket: int - How much each seat costs within a movie theater.
- Prices: Dollars - The cost of what is being purchased from seats to food.
  Seats: int - How many seats are available.
- Movies: string - The movies that are available within the movie theater.

Operations:

- purchaseTickets(tickets, prices) - obtains the prices of the chosen movie and how many.
- availableMovies (movies) - Show the user what movies are available to be shown and any additional info within the movie

Class: User
Description of Class: The person who is using the interface of the ticketing system
Attributes:

- address: int - Known location to provide info on local movie theaters.
- email: string - email address to give notifications on deals & for signing in to the website/app.
- phone: double - To give notifications & function for double verification when signing in.
- Age: int - To verify age limit & give deals on birthday

Operations: They do not give any additional operators besides keep information on themself

Class: Account
Description of Class: The main system of setting to review threw website settings and ID verify.
Attributes:

- Username: string - Secondary login username besides email & to customize the user
- Password: string - The main system to lock other users in order for security & to keep the creator of the account to access the system.

Operations:

- resetPassword() - Allows the user to change the password to a new one after completing a 2-Step verification.
- changeUsername() - Allows the user to change the username to a new one after an email verification.

Class: Admin
Description of Class:
Attributes: No operators available as Admin is for management in order to adjust movie information as needed
Operations:

- addMovies() - add films as they release.
- addShows() - add additional films as they release.
- banUsers() - operator to ban any problematic user that causes issues within a movie theater.

Class: Notification
Description of Class: System to notify the user of verified purchase, the time of the movie and advertisements
Attributes:

- notificationID: int - Formatting on notification if it on android, iphone, email or phone number message.
- createdOn: date - A system set to remind the user of when the movie they bought is on in the day.
- Content: string - The information of the notification

Operations:

- send() - push a notification to the user on important info through email and/or phone

Class: Payment Processor
Description of Class: called upon from the ticketing system to make the transaction function once a ticket is purchased.
Attributes:

- bankRoutingNumber: double - The location of the bank account from the user to withdraw the money to complete the transaction
- payment_info: int - what the payment is for in order to give to the bank
- Balance: Dollars - The total amount being withdrawn from the account

Operations:

- payCC(payment_info) - The operation done to transfer money once purchased.

SWA Diagram

Description of classes
Description of attributes
Description of operations
***descriptions should be detailed and specify datatypes, function interfaces, parameters, etc..**

Development plan and timeline

Partitioning of tasks
Week 1- 2 : Preparations & diagrams
Week 3- 4 : Coding development & resource management
Week 5- 6 : Server development
Week 7- 8 : Finalization on website/app software
Week 9- 10 : Bug Fixing
Week 11- 12 : User Interface development
Week 13- 14 : Testing
Week 15- 16 : Bug Fixing
Week 17- 18 : Adding additional or missing features
Week 19- 20 : Testing
Week 21- 22 : Bug Fixing
Week 23- 24 : Finalizations & submission
Team member responsibilities
Fernando Quintana - The sole member responsible within the team in order to complete the assignment

Verification Test Plan: