Ticketing System

Swaraj Khatri
Samantha Georges
Ronessa Mose
David Prieto

System Description

The Ticketing System utilizes various classes, methods, and functions to achieve the client's goals for their software. Through the use of diagrams, the Software Design Specification offers an outline and description of the system for software engineers to interpret into code.
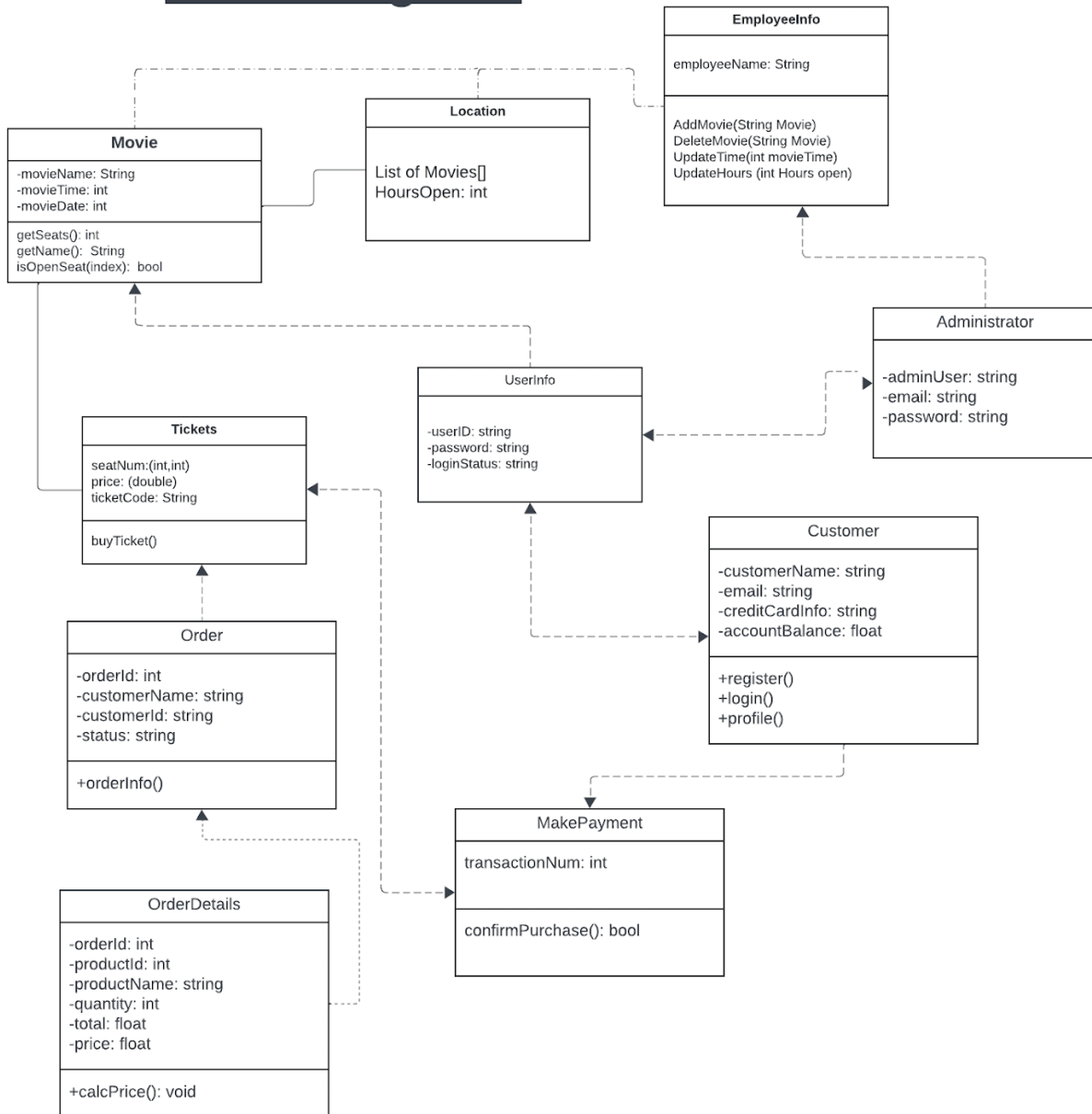
Software Architecture Overview

1. Class: Location – Bundles in movies being shown and operating hours.
   a. List_Of_Movies is an Array that holds the string value of the names of the movies.
   b. HoursOpen stores an int value giving us the operating hours
      i. Employees should be able to insert/delete and change everything in the scope.
2. Class: EmployeeInfo – Bundles in the data that the employer would have access to.
   a. Employees would give their name as a string value if the value is correct then it would grant access to the operation.
      i. The employee would be able to change the functions in the location class. They'd be able to change the time and movies.
3. Class: Movie – Bundles up all the information about the movies and seats.

   a. There would be the movie name, movie time, and movie date. This would all be values.
      i. It will get the name of the movie.
      ii. It will get the index of the seat.
      iii. It will have a function to see if the seat is open or empty.
4. Class: Customer – This would bundle all the information about the customer.
   a. The information that will be taken in are the username, email address, credit card information of the user, and account balance.
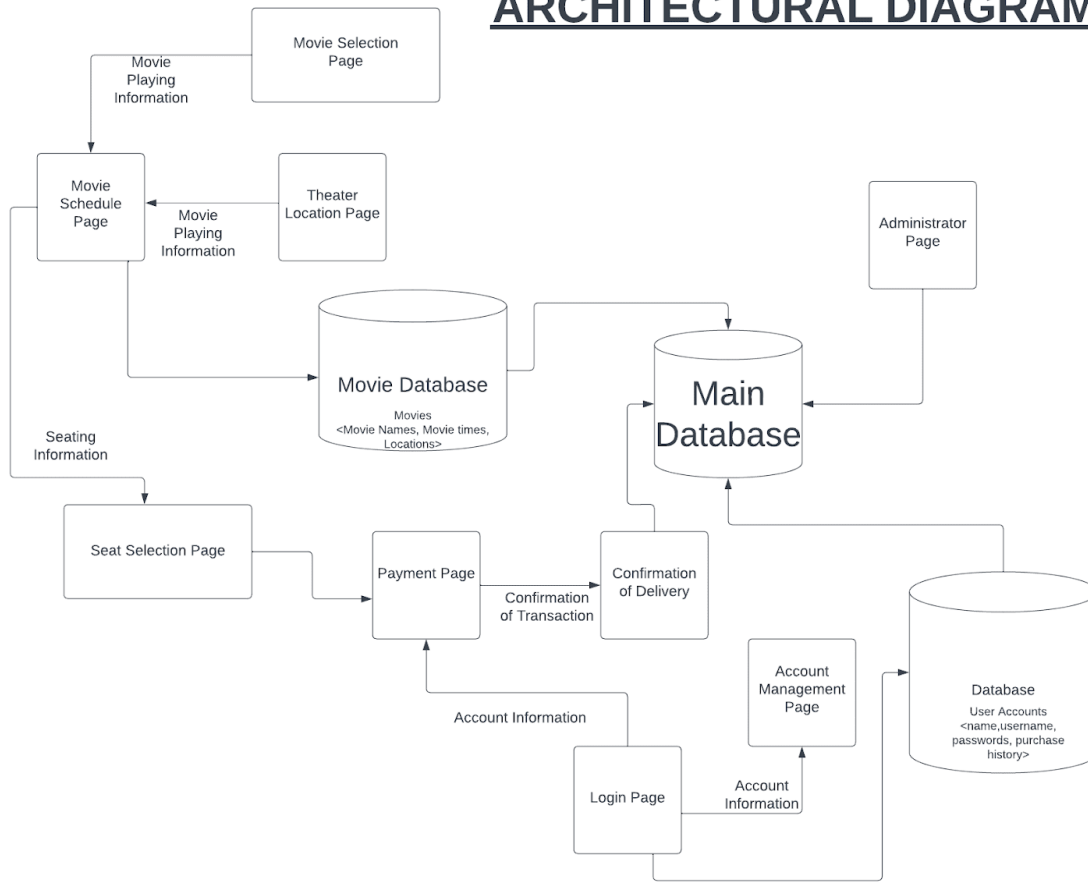   i. The functions will allow the user to register.

  ii. It will also give users the option to login and it will take them to the next page.

5. Class: Administrator – This class grants permissions to employees and customers.
 a. This allows employees access to change the name of the movie, seating, and location.

6. Class: Tickets - The ticket class allows the user to gather all of the information needed: from seating number, ticket price, and movie availability.
 a. The seating number function gives the user the information of where inside the theater.

  i. Let the user know where to sit.

 b. The price function gives the user the total ticket price.

  i. The user will know the cost.

 c. Allows the user to know which movie they purchased to watch.

  i. Will inform the user of the movie.

7. Class: UserInfo – This class provides employees with specific user information including ID, password, and login status.
 a. This class works with the Customer class to implement additional user login and status details.

8. Class: MakePayment – Combines all the information to handle the payment.
 a. It is going to assign a number for the transaction that should work as a receipt.
  i. Confirm purchase will just be a boolean value that passes in and gives customers a receipt.

9. Class: The Order class allows the user to receive unique login information based on their: ID number, name, and status.
    a. This is going to take the customer information, ID, and status.
        i. This will be a function that gives all of the information about what was selected that would be imputed to the user.

10. Class: OrderDetails – This class provides administrators and employees with the order ID, product ID, product name, quantity, price, and total price.
    a. calcPrice(): This method will calculate and return the total price of the product.

UML Diagram

EmployeeInfo

employeeName: String

AddMovie(String Movie)
DeleteMovie(String Movie)
UpdateTime(int movieTime)
UpdateHours (int Hours open)

Location

List of Movies[]
HoursOpen: int

Movie

-movieName: String
-movieTime: int
-movieDate: int

getSeats(): int
getName(): String
isOpenSeat(index): bool

Administrator

-adminUser: string
-email: string
-password: string

UserInfo

-userID: string
-password: string
-loginStatus: string

Tickets

seatNum:(int,int)
price: (double)
ticketCode: String

buyTicket()

Customer

-customerName: string
-email: string
-creditCardInfo: string
-accountBalance: float

+register()
+login()
+profile()

Order

-orderId: int
-customerName: string
-customerId: string
-status: string

+orderInfo()

MakePayment

transactionNum: int

confirmPurchase(): bool

OrderDetails

-orderId: int
-productId: int
-productName: string
-quantity: int
-total: float
-price: float

+calcPrice(): void

ARCHITECTURAL DIAGRAM

Development plan and timeline

Partitioning of tasks

- ○ UML Diagram: Entails the planning out of various classes and functions revolving around user information, employee information, administration, the payment process, and ticketing.
- ○ Architectural Diagram: Map out all of the major components of the system.
- ○ Overview: Describe classes, attributes, and operations involved in the Ticketing System's SDS.

Team member responsibilities:

UML Diagram	Architectural Diagram	Overview & System

		Description
Ronessa Mose Swaraj Khatri David Prieto Samantha Georges	Ronessa Mose Swaraj Khatri David Prieto	Swaraj Khatri Samantha Georges David Prieto Ronessa Mose