The Boys Movie Theater

Software Requirements Specification

Version 4.0

10/24/24

Group 9
Michael Dinh
Nathan Fermo
Toby Zaremba
Kyle Potente

Prepared for
CS 250- Introduction to Software Systems
Instructor: Gus Hanna, Ph.D.
Fall 2023

Revision History

Date	Description	Author	Comments
<09/26/24>	<Version 1>	<Group 9>	First revision regarding information on the introduction, general description, and specific requirements of our movie ticketing system.
<10/10/24>	<Version 2>	<Group 9>	The second revision includes SWA AND UML diagrams in 4.1, descriptions of the diagrams, and the development and timeline section.
<10/24/24>	<Version 3>	<Group 9>	The third revision has updated descriptions for the SWA and UML diagram (as indicated in the TA notes from the last assignment). Test Plans are also written at the bottom of the document relating to the test plans on the Excel sheet.
<11/7/24	<Version 4>	>Group 9>	The fourth revision includes the data management strategy and an updated SWA diagram to match.

Document Approval

The following Software Requirements Specification has been accepted and approved by the following:

Signature	Printed Name	Title	Date
	Michael, Nathan, Toby, Kyle	Software Eng.	
	Dr. Gus Hanna	Instructor, CS 250	

Table of Contents

Revision History ii

Document Approval ii

1. Introduction 1

1.1 Purpose 1
1.2 Scope 1
1.3 Definitions, Acronyms, and Abbreviations 1
1.4 References 1
1.5 Overview 1

2. General Description 2

2.1 Product Perspective 2
2.2 Product Functions 2
2.3 User Characteristics 2
2.4 General Constraints 2
2.5 Assumptions and Dependencies 2

3. Specific Requirements 2

3.1 External Interface Requirements 3
3.1.1 User Interfaces 3
3.1.2 Hardware Interfaces 3
3.1.3 Software Interfaces 3
3.1.4 Communications Interfaces 3
3.2 Functional Requirements 3
3.2.1 <Functional Requirement or Feature #1> 3
3.2.2 <Functional Requirement or Feature #2> 3
3.3 Use Cases 3
3.3.1 Use Case #1 3
3.3.2 Use Case #2 3
3.3.3 Use Case #3 3
3.3.4 Use Case #4 3
3.4 Classes / Objects 3
3.4.1 <Class / Object #1> 3
3.4.2 <Class / Object #2> 3
3.5 Non-Functional Requirements 4
3.5.1 Performance 4
3.5.2 Reliability 4
3.5.3 Availability 4
3.5.4 Security 4
3.5.5 Maintainability 4
3.5.6 Portability 4
3.6 Inverse Requirements 4
3.7 Design Constraints 4
3.8 Logical Database Requirements 4
3.9 Other Requirements 4

4. Analysis Models 4

4.1 Sequence Diagrams 5
4.3 Data Flow Diagrams (DFD) 5
4.2 State-Transition Diagrams (STD) 5

5. Change Management Process 5

A. Appendices 5

A.1 Appendix 1 5
A.2 Appendix 2 5

1. Introduction

The introduction to the Software Requirement Specification (SRS) document should provide an overview of the complete SRS document. While writing this document please remember that this document should contain all of the information needed by a software engineer to adequately design and implement the software product described by the requirements listed in this document. (Note: the following subsection annotates are largely taken from the IEEE Guide to SRS).

1.1 Purpose

The purpose of this document is to develop and present a detailed description of The Boys Movie Theater, an online movie theater ticketing system. The document will explain the system's specific features and functions. Additionally, the document will describe how the system responds to external interactions. The document is intended to provide both the development team and clients with a clear understanding of the system's functionality and overall design. This document is not intended for anyone unaffected by our product.

1.2 Scope

The movie theater system is an online movie ticketing platform designed to sell movies digitally or in-person via a movie ticket at The Boys Movie theaters in San Diego. If the in-person option is selected, the system will allow users to view available movie showtimes, select seats, purchase tickets, and receive digital confirmation through email or SMS. Alternatively, users also have the option to make a digital purchase of a movie that they can stream through the website on a device. Purchases at in-person locations are also permitted through kiosks. Purchases can be made through secure and safe transactions through our provided payment options, which will be directly communicated to the banks they are associated with. Our software system will hold a strong foundation in terms of securities, frameworks, and algorithms to satisfy customers. We hold a strong connection with all our third parties to further ensure the safety of users when accessing our website and committing transactions. The user roles are divided as follows with the given titles: Registered users with accounts, guests, administrators, and employees. The software will cover customer service, account reward distribution for authenticated users, a database consisting of current movie air and showtimes, and cookies to encrypt user information.

1.3 Definitions, Acronyms, and Abbreviations

This subsection should provide the definitions of all terms, acronyms, and abbreviations required to properly interpret the SRS. This information may be provided by reference to one or more appendixes in the SRS or by reference to other documents.

SRS	Software Requirements Specification
FAQ	Frequently Asked Questions
SMS	Short Message Service

NFC	Near-field communication technology related to contactless payments through the communication of electronic devices.
MTA	Mail transfer agent email server software to receive and relay emails from different devices.
OAuth	Open authorization protocol with emails.
IT	Information Technology
MTTR	Mean time to repair
AES	Advanced Encryption Standard
PCI DSS	Payment Card Industry Data Security Standard
GDPR	General Data Protection Regulation
API	Application Programming Interface
SIEM	Security Information and Event Management
QR	Quick Response Code
DB	Database
MPAA	Motion Picture Association of America to rate intended audience of movies
TRS	Transactions Records Server

1.4 References

The Institute of Electrical and Electronics Engineers. IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications. IEEE Computer Society, 1998.

1.5 Overview

The upcoming sections of this document will provide a detailed description of the product and outline all necessary requirements. This includes: Section Two: General Overview of The Boys Movie Theater Ticketing System, Section Three: Functional and Non-Functional Requirements, Section Four: Analysis Models, and Section Five: Change Management Process. Section 2 will offer an overview of the project's core characteristics, while Section 3 will delve into both functional and non-functional requirements, including areas such as security, maintainability, and portability. Additionally, Section 4 will present a visual representation of the system. Lastly, Section 5 will define the steps for updating the Software Requirements Specification (SRS) in the event of changes to the project's scope or requirements.

2. General Description

This section of the SRS should describe the general factors that affect 'the product and its requirements. It should be made clear that this section does not state specific requirements; it only makes those requirements easier to understand.

2.1 Product Perspective

The product will be designed for ease of access for both the users and administrators, improving the customer experience while also streamlining management efficiency. It should display information taken from movie databases, provide a visual graphic for each movie listed on the site, and also scrape online review sites to display reviews and critic quotes for each movie. The product will also handle payments securely, and constantly update the online website to maintain the same information as the in-person site.

Other functions that the product will include are:

- Provide an interactive map for each movie listing to allow users to select open seats of their choice.
- Has support for three different languages: English, Spanish, and Swedish.
- Allow users to register for loyalty accounts that can store personal information, payment details, purchase history, and loyalty points.
- Can block bots to prevent ticket scalpers from mass-buying tickets to high demand movies.
- Utilize a queueing system for managing large volumes of requests in a short period of time.
- Will convert ticket prices to the user's local currency.
- Has a customer support portal to answer user feedback and solve issues efficiently.
- Will limit the number of purchasable tickets to 20 in a single transaction.
- Will allow tickets to be purchased within 2 weeks before showtime as well as 10 minutes after showtime.

2.2 Product Functions

This subsection of the SRS should provide a summary of the functions that the software will perform.

The software will allow the user to browse through the website's catalog of movies. Then the software will be able to gather input from the user and allow them to pick a movie from our website and have the option to either get a barcode in their email to view the movie at a theater or alternatively get a link in order to view the video on their own. Then the user will be asked to sign in. If the customer already has an account the software will detect that and use pre-loaded data in order to continue. If the user does not have an account the data from the user will then be

collected, from email to SMS in order to deliver the selected method of viewing. With this data in mind the website will securely hold the email and SMS in order to maintain security of the customer.

2.3 User Characteristics

This subsection of the SRS should describe those general characteristics of the eventual users of the product that will affect the specific requirements. (See the IEEE Guide to SRS for more details).

The general user of this software will be very curious as they want to view many movies and eventually experience these movies through our website. In addition to that the eventual users will love movies as they have the ability to browse as many as they like. If the users love to explore they will view everything on the website. When the customer is doing these processes on our site we must understand how the user functions and we have to make sure our security and privacy is good as we keep important information.

2.4 General Constraints

This subsection of the SRS should provide a general description of any other items that will limit the developer's options for designing the system. (See the IEEE Guide to SRS for a partial list of possible general constraints).

Hardware Constraints - No support for Windows operating systems before Windows 10
User Constraint - Could be indecisive on movies or not like movies presented
Website Constraints - Doesn't show all movies only the one's available also can crash if there are too many users

2.5 Assumptions and Dependencies

Here is a list of some of the assumptions and dependencies that may affect the system:

- The assumption that a constant stream of Internet connectivity will be present in order to access the system's platform for customers. This includes The machines that would be present at the movie theater itself
- Payment methods that aren't already listed in the system(Such as foreign banking, or a digital currency that the system doesn't recognize) will not be supported
- Mobile App integration: we aren't planning on dedicating an app for easier accessibility for Mobile users. If mobile users still insist they can still access the website through the safari app present on most mobile devices with limited functionality
- Legal and Regulatory Compliance: The software is expected to comply with relevant legal and regulatory frameworks, including data protection laws (e.g., GDPR) and financial transaction regulations within the countries where the system operates.
- API Stability: The system depends on the availability and stability of external APIs, such as movie databases or geolocation services, to fetch real-time data and provide an optimal user experience.

- There will include User Authentication Services: such as email verification and password resets

3. Specific Requirements

This will be the largest and most important section of the SRS. The customer requirements will be embodied within Section 2, but this section will give the D-requirements that are used to guide the project's software design, implementation, and testing.

Each requirement in this section should be:

- *Correct*
- *Traceable (both forward and backward to prior/future artifacts)*
- *Unambiguous*
- *Verifiable (i.e., testable)*
- *Prioritized (with respect to importance and/or stability)*
- *Complete*
- *Consistent*
- *Uniquely identifiable (usually via numbering like 3.4.5.6)*

Attention should be paid to the carefully organize the requirements presented in this section so that they may easily accessed and understood. Furthermore, this SRS is not the software design document, therefore one should avoid the tendency to over-constrain (and therefore design) the software project within this SRS.

3.1 External Interface Requirements

3.1.1 User Interfaces

3.1.1.1 The system shall provide a responsive and interactive web-based interface that is accessible through desktops and laptops that support both Windows 10-11 and macOS Sierra.
3.1.1.2 The system shall provide a web-based interface optimized for mobile devices that is accessible to both iOS and Android devices.
3.1.1.3 The system shall provide an extensive menu with a simple graphical user interface, making it simple for users to navigate and search for movies based on title, genre, date, and location.
3.1.1.4 The system shall provide a real-time visual representation of seating arrangements, allowing users to view availability and select their desired seats.
3.1.1.5 The system shall present a confirmation screen that will provide a summary of the user's purchase, showing their seat details, movie details, date, and time before they finalize their payment.
3.1.1.6 The system shall notify users for a successful payment and ticket booking.
3.1.1.7 The system shall have an interface to register for a loyalty account, allowing them to store their personal information, their payment details, as well as track their transaction history and their loyalty points.
3.1.1.8 The system shall ensure that only one device can concurrently login to a user account in order to strengthen account security.
3.1.1.9 The system shall limit the amount of purchasable tickets in a single transaction to 20.

3.1.1.10 The system shall allow users to choose between English, Spanish, and Swedish when navigating through the online site.
3.1.1.11 The system shall provide a ticket page after confirming their purchase, providing the user with a unique barcode that the workers can scan at the theater.

3.1.2 Hardware Interfaces

3.1.2.1 The system shall allow users to receive a digital barcode via email after purchasing an in-person ticket, which can then be scanned at the theater for ticket validation.
3.1.2.2 The system shall install touchscreen self-service kiosks at the entrance of theaters to allow for ticket purchasing, seat selection, and printing of tickets.
3.1.2.3 The system shall support contactless payments by integrating NFC-enabled payment terminals at the theaters, allowing mobile-based payment such as Apple Pay or Google Wallet that users may choose to use.

3.1.3 Software Interfaces

3.1.3.1 The system shall interface with third-party payment processing services to securely handle user transactions and ensure data protection during payment.
3.1.3.2 The system shall integrate several user authentication services such as Google or Facebook that will allow for a simple and secure account login and management.
3.1.3.3 The system shall integrate a queueing system in order to manage high volumes of requests on the interface during peak times to allow for easier scalability and a smoother user experience.

3.1.4 Communications Interfaces

3.1.4.1 The system shall notify users through email for ticket purchase confirmations, exchanges, and refunds.
3.1.4.2 The system shall allow users to subscribe to email newsletters and SMS services, allowing them to receive updates on new movie releases, promotions, and special events.
3.1.4.3 The system shall utilize blockchain technology to generate unique and non-replicable barcodes in order to ensure the security of each ticket.
3.1.4.4 The system shall provide a customer support portal to answer user inquiries and efficiently solve issues.

3.2 Functional Requirements

This section describes specific features of the software project. If desired, some requirements may be specified in the use-case format and listed in the Use Cases Section.

3.2.1 <Purchasing Tickets and Seat Selection>

3.2.1.1 Introduction
The system shall allow users to browse available movies in the catalog, add tickets for their selected movies into the purchasing cart, select available seats for their movies on an interactive seating diagram for each screening room, and complete their purchases at the checkout page of the interface. Depending on the selected movie theater location, different movies will be made available for purchase at different screening times for each screening room. After the user completes their selection of a movie theater location, selected movies, its showtimes, and available seats for that screening room, they can finalize their purchases through secure online

third-party payments via credit/debit, or NFC-enabled devices for in-person transactions at any of our movie theater locations. The system shall also update movie showtimes and open seat availabilities in real time. Upon finalizing their payments, users will receive a digital ticket barcode sent directly to their email and an SMS message notifying them that their purchase has been completed with a link to the barcode. They will also receive points rewards for each movie ticket purchased.

3.2.1.2 Inputs

- The user shall utilize the system's interface to browse and select movies included in the catalog of our database, which takes into account the most popular and newest movies. Only a select few of the latest and most popular movies will be available for selection at in-person theater locations, but other movies can be selected from our database for portable viewership anywhere on compatible devices, such as any desktop computer or laptop with Windows 10 and above, or mobile devices that support iOS 16 and above, and Android 12 and above.
- The user can click on a movie's internal link, they will be taken to another page on the same tab that will have information regarding that movie. There will also be a playable trailer of the movie that the users can click and view.
- Users shall be able to cast their rating out of 5 stars through the internal links for each movie that will be updated in hour intervals in the system.
- Users shall be able to select their seats from a real-time seating chart which will give an overview of the available seats.
- The user shall be able to add up to a max of 4 tickets into their shopping cart.
- The user shall be able to finalize their purchases for items in the shopping cart, where they will be able to select and input their payment details such as a credit/debit card from selected providers, including Visa, Chase Bank, Bank of America, and Wells Fargo, as well as the option to pay with PayPal.

3.2.1.3 Processing

- The system shall organize a list of available movies determined by the user's selected movie theater location. Upon verifying the available movies for the selected movie theater, the system will cross-check the information with the theater's database to check and see if the movie is available for booking for their desired time selection.
- If a user decides to select a movie for in-home/portable streaming, the system shall check and verify the user's device compatibility requirements before allowing the user to add the movie to their cart. The system shall check the user's device ID and IP address to determine whether or not the device is associated to the user's account.
- The system shall verify the number of movie tickets in a user's cart before they can make a purchase. If the total ticket number exceeds 4, the system shall deny the user from finalizing any payments and request they remove a certain number of tickets till the total is 4 or less.
- When the user selects the desired seats that are available in the seat maps interface for a selected movie screening room, the system shall reserve their selected seats for a maximum amount of 10 minutes. Once this allotted time has passed, those seats will no longer be locked and can be selected by other users.

- The system shall process the user's payment through a third-party gateway with Visa, Chase, Bank of America, and Wells Fargo. They will verify if the payment has been successful or not.
- If the user is paying in person, their devices must be NFC-enabled, where they can then make a secure tap-to-pay payment.
- Upon successful payment, the system shall generate a unique digital code that includes a barcode. If the purchase was for online viewership of the movie, the person will have access to the movie on a separate page of the website that links their purchased data of movies to their account.
- The system shall store their purchase history in the database, as well as the receipts, and distribute the 100 reward points per movie ticket purchase if applicable.

3.2.1.4 Outputs

- Upon successful payment, the system shall notify the user of the digital ticket with the barcode through email and an SMS message (if they have opted to include their phone number) that can be used to scan into one of our in-person movie theaters, as well as information regarding their seating selection.
- Upon successful payment for the online streaming option of a movie, the system shall notify the user on how to access the movie on the website through specific instructions sent to their email address that will lead them to another tab of the page.
- The system shall grant users 100 reward points per movie ticket purchase (only if they have an account with the website associated with our movie theater), which can be used to redeem a free movie ticket once 1000 points have been accumulated. The system will update the user's account with said points per ticket purchased, which can be viewed in the user's profile section of the website.
- Upon accumulating 1000 reward points, the system will notify the user through email and an SMS message about their ability to redeem the points.

3.2.1.5 Error Handling

- If the user's entered payment information is invalid or declined by the third-party gates, they will notify the system, which will in turn notify the user to re-enter or select another payment option.
- If the user fails to finalize their payment for any selected movie tickets for one of our movie theater locations with the designated amount of time for seat reservation, the system shall notify the user stating that the session has expired, and release any reserved seats back into the seat maps interface. The user will also be prompted to reselect their seats.
- The system shall prevent users from clicking on seats in the seating map that another customer has already reserved and documented into the selected movie theater's database.
- If the user attempts to check out their shopping cart with more than 4 movie tickets for any movie screening, the system shall notify the user and prompt them to remove a selected number of tickets till the requirement of 4 or fewer is met.
- If the user attempts to enter an invalid or expired promo code that the system periodically offers to users through notifications to their email and SMS messages, the system shall notify the user of the code's ineligibility and prompt them to use another code, or proceed with the transaction without one.

- If a user's device is not compatible with our system's minimum requirements for operating systems, the system shall notify the user about the incompatibility before they proceed with their purchase.

3.2.2 <Creating an Account>

3.2.2.1 Introduction

The system shall allow users to create an account associated with the website, which will provide users with an enhanced experience, including rewards points with every movie ticket purchase which can be redeemed later on when enough points are accumulated for a free movie ticket, and email/SMS messages regarding promotional offers. The system shall also allow tracking of their purchase history to provide personal recommendations on the home page.

3.2.2.2 Inputs

- The user shall input their full name, email address, a unique username not already present in the database that holds all user login information, and a password that requires a minimum length of 8 characters. The password must also contain at least 1 capital letter, number, and special character to meet the system's security requirements.
- The user can opt to sign in with a Google account (Gmail) if they possess one.
- The user can opt to input their phone number to receive SMS messages from the system or Customer Service department entailing details such as promotional deals on movie tickets and updates.

3.2.2.3 Processing

- The system shall validate the user's email and check to see if the email has already been registered in the database.
- The system shall verify if the user's entered password meets the system's security requirements of containing a minimum of 1 capital letter, 1 number, 1 special character, and a minimum length of 8 characters.
- Upon successfully validating the user's information and inputs, the system shall encrypt the entered password, store the user's information in the database, and send a verification email to the provided email address with a link to confirm their account creation.

3.2.2.4 Outputs

- The system shall send a verification email to the user's email address with a link to confirm and activate the account creation.
- Upon successful verification of the user's email within 24 hours, the system shall grant the user a 50% discount on their first 2 ticket purchases, as well as a free drink that can be redeemed at any of our theater locations.
- Upon confirmation of the user's account, they will have access to rewards points and promotional deals as additional features on the website.

3.2.2.5 Error Handling

- If the user's email address is already in use, the system shall prompt the user to input another email address or log in if they already have an account.
- If the user does not confirm their email address, they will be unable to receive rewards points upon making a purchase or redeeming promotional deals and sales, but can still access their account and utilize the website for movie ticket purchases.

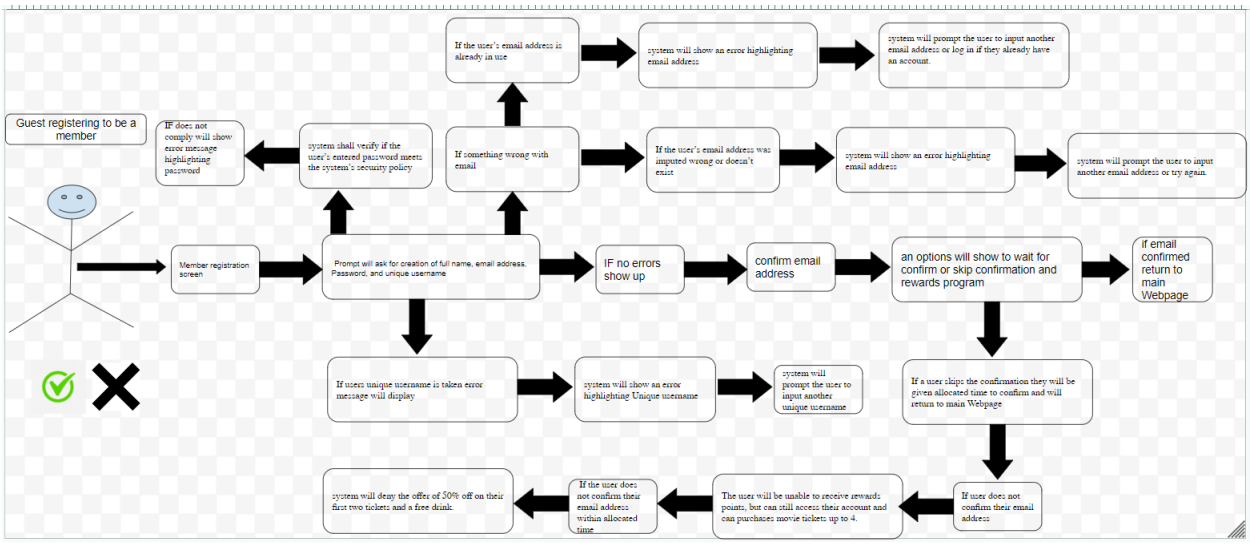
- If the user does not confirm their email address within the allotted amount of time, the system shall deny the offer of 50% off on their first two tickets and a free drink.
- If the user's username is already in use, the system shall prompt the user to input another username, while also providing suggestions based on the current entered username they are attempting to use.
- If the user's password does not meet the system's safety criteria, the system shall prompt the user to enter a new password.

3.3 Use Cases

3.3.1 Use Case #1

3.3.1.1: Model Representation of the registration process of a member

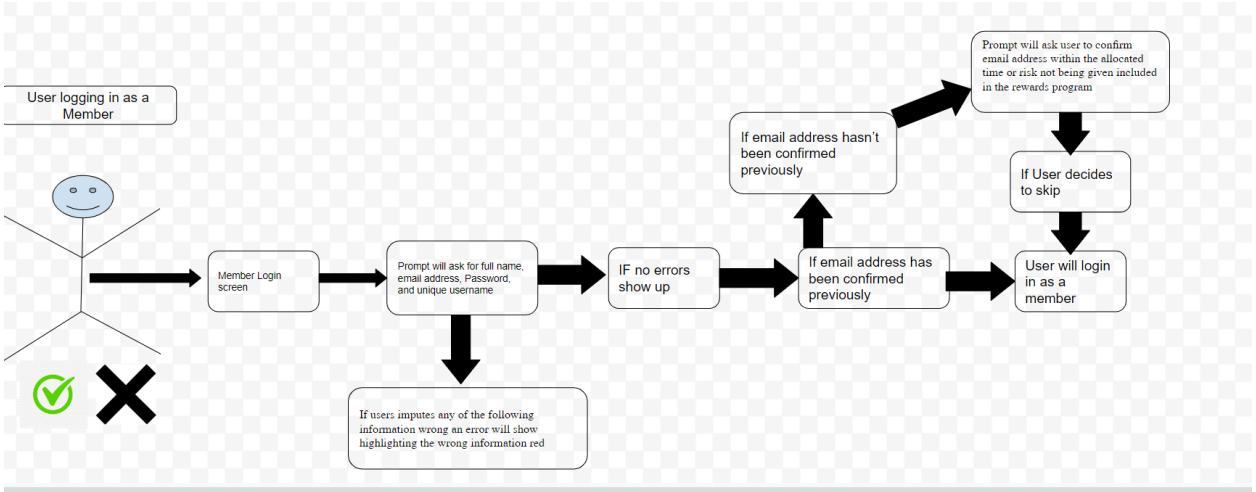
- Guests who try to become a member will go through this system:
- Member registration screen -> Prompt will ask for creation of full name, email address, and unique username -> IF no errors show up -> user will be prompted to confirm email address->an options will show to skip confirmation and rewards program -> if they confirmed email they will return the main Webpage
-
- system shall verify if the user's entered password meets the system's security policy-> IF does not comply will show error message highlighting password
-
- If the user's email address is already in use -> system will prompt the user to input another email address or log in if they already have an account.
-
- If the user's email address was imputed wrong or doesn't exist -> system will show an error highlighting email address -> system will prompt the user to input another email address or try again.
-
- If a user skips the confirmation they will be given allocated time to confirm ->If the user does not confirm their email address -> they will be unable to receive rewards points upon making a purchase or redeeming promotional deals and sales, but can still access their account and utilize the website for movie ticket purchases. -> If the user does not confirm their email address within allocated time -> system will deny the offer of 50% off on their first two tickets and a free drink.
-
- If users unique username is taken error message will display -> system will prompt the user to input another unique username



3.3.2 Use Case #2

Model Representation of the login process of a member

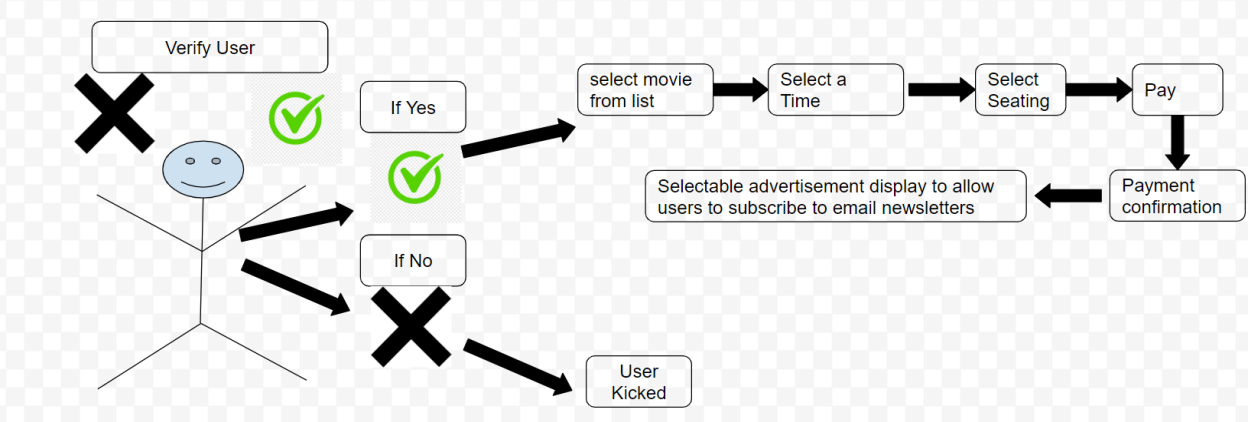
- User logging in as a Member -> Member login screen -> Prompt will ask for full name, email address, Password, and unique username -> If no errors show up -> If email address has been confirmed previously -> User will login in as a member
-
- If users imputes any of the following information wrong an error will show highlighting the wrong information red
-
- If email address hasn't been confirmed previously ->Prompt will ask user to confirm email address within the allocated time or risk not being given included in the rewards program -> If User decides to skip



3.3.3 Use Case #3

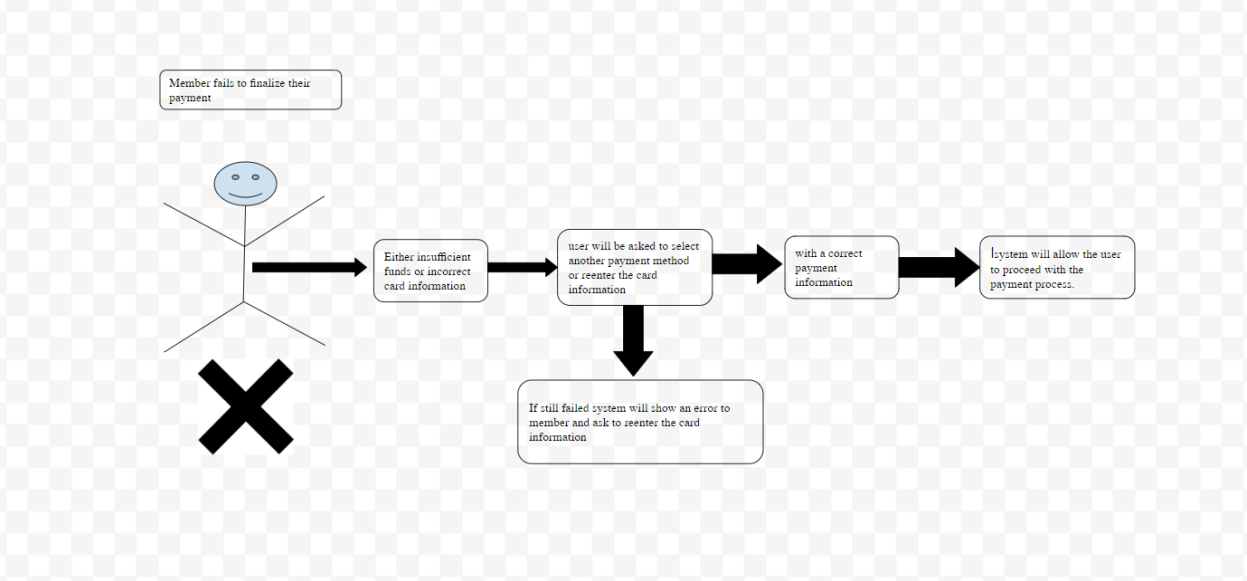
Model Representation of the buying process of the user(any person who logins to the system That isn't guest)

- verify user -> select movie from list -> select a time -> select seating -> pay -> selectable advertisement display to allow users to subscribe to email newsletters



3.3.4 Use Case #4

Model Representation of Error management of payment details



- If a Member fails to finalize their payment -> Either insufficient funds or incorrect card information -> user will be asked to select another payment method or reenter the card information -> with a correct payment information _> system will allow the user to proceed with the payment process.

3.4 Classes / Objects -(more niche description, refer to submission TA comments)

3.4.1 <Class / Object #1>

Account Input/ Object #1
3.4.1.1 Attributes

-See if Account is Already Present or if Password Is Forgotten
-If Password is forgotten ask for username
-Send email for recovery
-Create Account
-Take Email
-Take New Password
-Username
-Password
-Log into account
-Allow them to select movies

3.4.1.2 Functions
<Reference to functional requirements and/or use cases>

3.4.2 <Class / Object #2>

Purchase Function/ Object #2

- Show recommended movies that are trending to drive up sales
- Then sort movies by genres
- After movie is selected show details of movie
- Show button to purchase
- Show options of either getting a ticket for a movie theater or download of movie
- Add to cart
- Get Card information for payment
- When payment is accepted send either ticket or video file
- Send confirmation email with ticket/video
- Show thank you and redirect them to site
- Show other movies similar to movie purchased

…

3.5 Non-Functional Requirements

Non-functional requirements may exist for the following attributes. Often these requirements must be achieved at a system-wide level rather than at a unit level. State the requirements in the following sections in measurable terms (e.g., 95% of transaction shall be processed in less than a second, system downtime may not exceed 1 minute per day, > 30 day MTBF value, etc).

3.5.1 Performance

- The system shall process 98% of all transactions within a second or less.
- The system shall update the seating map interface in real time within 2 seconds or less.
- The website shall load all pages and interactions such as the catalog, transaction page, transaction history, movie links, trailers, etc, within 3 seconds or less on average.

3.5.2 Reliability

- The system's uptime will be held 99.9% of the time, resulting in no more than 8.77 hours of downtime.
- The system shall be able to handle up to 10 thousand concurrent users.
- The system shall support English, Spanish, and Swedish as the language options.

3.5.3 Availability

- The system shall be accessible to users 24/7, except during system downtime or maintenance. When the system is down, users shall be notified when they attempt to access the website.
- Movie ticket purchases and availability can only be made for the 15 movie theaters located in California.

3.5.4 Security

- The system shall use AES-256 encryption to protect every user's sensitive data such as payment information and user credentials from unauthorized users. AES-256 encryption will also be applied when users opt to have their sessions remembered without the need to log in with their credentials through encrypted and secure cookies.
- The IT department shall utilize Splunk's SIEM program to monitor and detect security threats, user authentication login attempts, financial transactions, and employees/departments who handle customer data (ie. Customer Service, IT, Finance and Accounting, Management).
- The user's sensitive data shall comply with PCI DSS and GDPR rules to maintain trust with users and ensure that they feel safe and secure with our system.
- The system shall automatically log users when they exit the website unless they have opted to save their sessions through the "Remember Me" option. Users will have the ability to save their sessions indefinitely or for 30 days on their current device.

3.5.5 Maintainability

- The MTTR shall be maintained within 3 hours or less for system failures by the IT department.
- A division within the IT department will run a monthly penetration test on our system to spot any potential vulnerabilities and holes in the security and functionality.

3.5.6 Portability

- The system shall be compatible with the following: Windows 10 and above, macOS Sierra and above, iOS 16 and above, and Android 12 and above. Users can access the website on desktop computers, laptops, mobile devices, and tablets.

- Users will be able to access all the website's functionalities across the above-mentioned electronic platforms. For mobile devices, the jQuery Mobile library shall be used to ensure that users can interact with all buttons correctly.
- The system shall be able to operate smoothly on the following websites: Google Chrome, Microsoft Edge, and Safari.

3.6 Inverse Requirements

3.6.1 Invalid Email/Password for Account Creation
If the user inputs an invalid email address during account information, or in the case that the OAuth is unable to detect the email used in their system, then our system shall prompt the user to enter another correct email address. If the user inputs a password that does not meet the system's security requirements, the system shall prompt the user to enter a new password that meets the requirements. Only when these two requirements have been met can the user proceed with the creation of their account.
3.6.2 Incorrect Username/Password
If the user inputs an incorrect username/password associated with an account, the system shall warn the user that the username/password is incorrect and state that they should re-enter their login information. The option "Forgot Username" and "Forgot Password" will also be available for users to click on the login interface which will redirect them to another page. On this page, users will have the ability to enter some account information such as their email address or phone number (depending on which method they prefer to be contacted with) and answer a few security questions regarding their account to see if it aligns with the information in the database. The user would also have the alternative to contact customer service to access their account information in the system's database if they are having more nuanced trouble with logging in through an MTA (mail transfer agent). If the user fails to type in the correct password associated with the correct username 3 times, the system shall lock the user's account until further action is taken with the customer service department for security purposes of the user's account.
3.6.3 Exceeding Ticket Purchases
If the user attempts to check out their shopping cart with more than 20 movie tickets, the system shall warn the user to remove x number of tickets until the total amount is equal to or less than 20.
3.6.4 Failed Payment Transaction
If the user fails to finalize their payment transaction through means such as insufficient funds in their selected payment option (notified by the bank associated with the third-party gate system) or incorrect card information, then the system shall notify the user to either select another payment method or to re-enter their card information. Once the user has provided the correct payment information, then will the system proceed with the payment process.
3.6.5 Invalid Digital Ticket Barcode
If the user attempts to scan an expired ticket barcode of their device for a movie at any of our movie theater locations, the system's ticket validation process will notice that the ticket's information (ie. ticket ID, movie theater location, showtime, and date of ticket usability) does not align with the database and notify the staff at the movie theater location.
3.6.6 Unavailable Seating in Seating Map

If the user attempts to select a seat on the seating map that is already taken (denoted by an X marker), the system shall inform the user that the seat they are currently attempting to reserve has already been taken. The system will then prompt the user to select another available seat.
3.6.7 Invalid Promo Code
If the user inputs an invalid promo code during checkout, the system shall warn the user that the code is expired/invalid, and to proceed with or without another code.
3.6.8 If the user attempts to redeem their rewards points when they have not accumulated enough points (1000 per free movie ticket), the system shall warn the user that they cannot proceed with their purchase.
3.6.9 Incompatible Device for Portable Streaming
If the user's device does not meet the system's requirements to stream the movie online after a successful purchase, the system shall notify the user that their device does not meet the streaming requirements and to use another device that does qualify.

3.7 Design Constraints

Specify design constrains imposed by other standards, company policies, hardware limitation, etc. that will impact this software project.

3.8 Logical Database Requirements

Will a database be used? If so, what logical requirements exist for data formats, storage capabilities, data retention, data integrity, etc.

3.9 Other Requirements

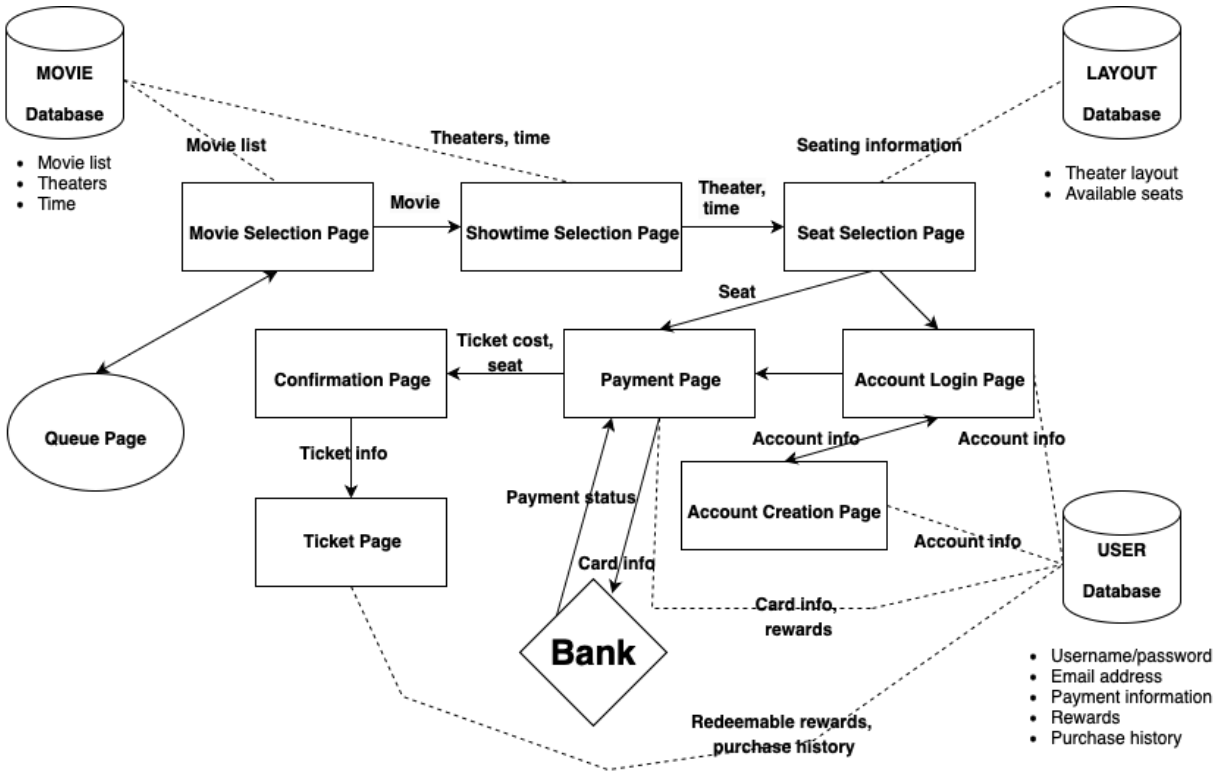
Catchall section for any additional requirements.

4. Analysis Models

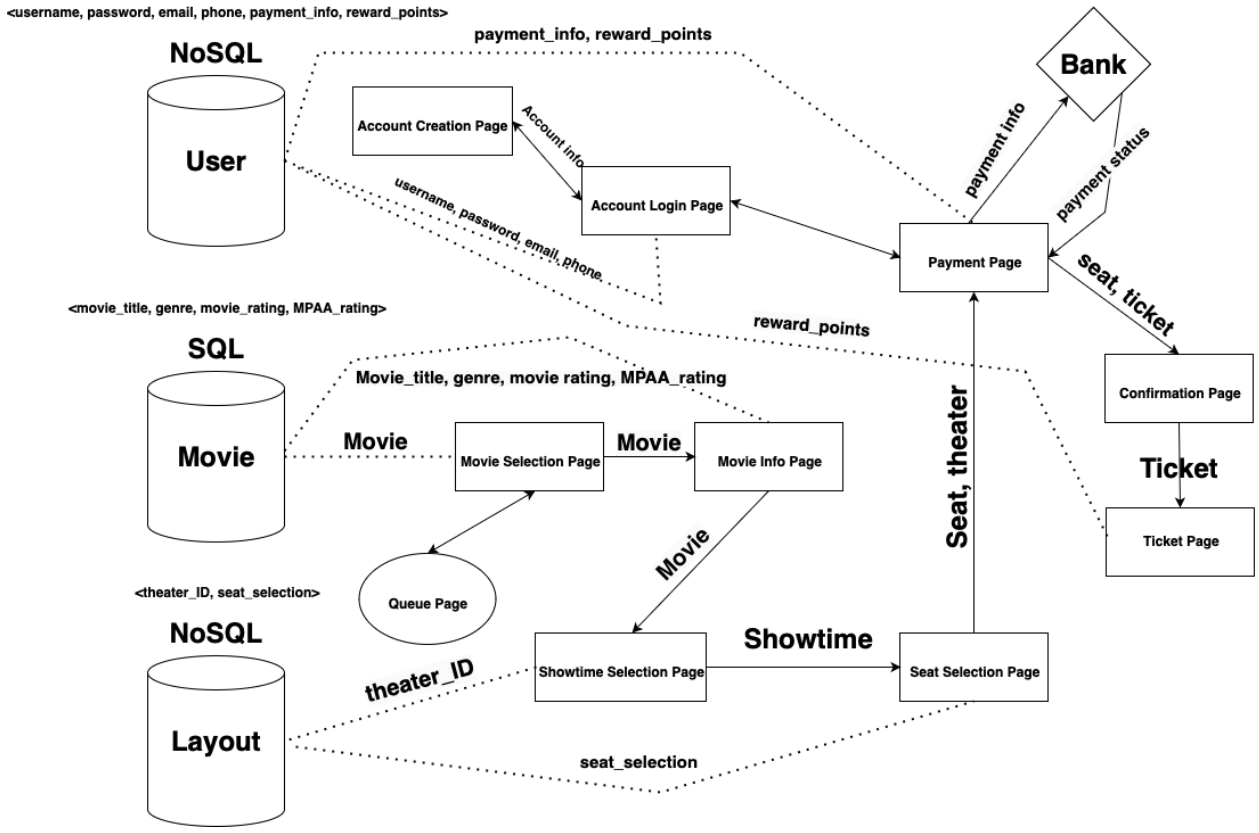
List all analysis models used in developing specific requirements previously given in this SRS. Each model should include an introduction and a narrative description. Furthermore, each model should be traceable to the SRS's requirements.

4.1 Sequence Diagrams

4.1.1 Software Architecture Diagram (SWA)



Revised SWA Diagram



4.1.1.1 SWA Diagram

The software architecture diagram portrays the basic layout of The Boys Movie Theater ticketing system. This visual representation of the diagram will demonstrate how each component of the software system and website are connected and their affiliation to the databases.

- The middle left database stores information on the movie list, theaters, and show times. The movie selection page of the website receives the movie list information from the database to accurately update available movies. The movie selection page communicates with the queue page to maintain server traffic, where if too many users attempt to access the website, some of them will be sent to the queue. This transitions over to the showtime selection page where it accurately shows data on the theaters and showtimes for said available movie chosen from the database, before displaying it for the user.
- The bottom-left database keeps track of the seating. It receives data on the movie theater room's layouts and available seats for each room. When a user selects their desired theater location and showtime for a movie, it transitions over to the seat selection page that uses the data in this database to accurately determine available seats and eliminate any possibilities of users taking a seat that is already occupied by another user in the system.
- The top left database stores sensitive information including usernames/passwords, payment information, rewards, and purchase history.
    - Account Login Page: Upon finalizing seat selection, the system will check whether or not the user is accessing an account stored in the database. In the case that the user is already logged in, the system will send their account information to the payment page and move them there. If the user is not, the system will prompt the user to either login, create an account, or proceed to the payment page as a guest. If the user opts to log in with an existing account, the database will make sure that the username and password match the user's corresponding information. Otherwise, they will be sent to the account creation page.
    - Account Creation Page: Users will have the ability to create their own account with a unique username and password, as well as an association to an email. This information will be stored in the database where it would be accessible in other pages that require the user's information (ie. payments).
    - Payment Page: The seat selection page and account login page/creation page sends information to the payment page. The page then retrieves the user's payment information stored in the database if they are logged in and have opted to have their payment information saved. Otherwise, they will have to manually input their payment information where it will be stored in the database unless opted not to. This page also interacts with the bank systems that have made an agreement with our company to handle validations of the payment, as well as payment information like the card used and payment status.
    - Confirmation Page: After finalizing payment information and confirming the purchase from the previous page, it leads to the confirmation page. From here, the system will display the purchase summary for the ticket purchase including the movie, theater location, showtime, and seating.
    - Ticket Page: The confirmation page then leads to the ticket page where the system presents a digital ticket for the user through a QR code. This information will also

be sent to the user's contact information such as their email and/or phone number, depending on whether or not that information is stored within the database. In the case that it is not, for example, a guest, they will be required to input their email address to have another accessible method to the ticket. The database will distribute rewards into the database and log them into the user's account if applicable. The system will not access reward points data for guest accounts as they are ineligible for rewards.

Updated SWA Diagram Points:

- The movie database stores information data on the current movie lists for each theater location (gets updated in real time) and the available showtimes. This interacts with the movie selection page and showtime selection page, where the movie database feeds data to these parts to have them accurately updated for users accessing these pages.
- The layout database stores data on the theater layouts and available seats for each movie theater and their theater rooms for movies. This interacts with the seat selection page providing and updating the seating arrangement and availability in real time.
- The user database stores data on user's account information such as login credentials, email addresses, payment information, purchase history, and reward points associated to each individual account. The account login, creation, and payment page pull data from this database to verify the user's information accurately and securely.

4.1.2 Data Management Strategy

When determining the number of databases required, we considered factors and relations in data in comparison to other data that is not so related, though ultimately connected to ensure the whole system functions properly. In this case, a total of three databases will be implemented to maintain our system's core functionalities and data security, while also maintaining efficient retrieval and storage. In terms of database types, we ultimately decided on using two NoSQL databases and one SQL database. Although SQL databases were considered for the usage of structured and relational data, the NoSQL database would benefit more from cases involving nonstructured data with more dynamic schemas, as well as scalability.

4.1.2.1 Movie Database

- The movie DB stores any information related to movies. This includes movie titles, available theater locations for said movies, and their showtimes.
- This is an SQL database that contains the movie's title, genre, total movie rating, theater_ID, and Motion Picture Association of America (MPAA) rating.
- The SQL database decision is based on its ability to support filtering and sorting by genre, movie rating, theater_ID (location), and MPAA rating.
- The movie DB will be connected to the Layout DB to dynamically check seat availability based of the user's selected movie and showtime retrieved from the data in the movie DB.

	movie DB				
WHO?	TITLE	GENRE	TOTAL MOVIE RATING	THEATER ID	MPAA RATING
PERSON 1	Transformers ONE	Action/Sci-fi	4.9/5	Theater A134	PG
PERSON 2	Smile 2	Horror/Mystery	3.8/5	Theater A134	R
PERSON 3	Venom: The Last Dance	Action/Sci-fi	4.0/5	Theater A134	PG-13

4.1.2.2 Layout Database

- The layout DB stores and manages all information related to the seating configuration and availability at each theater location's rooms.
- This is a NoSQL database that contains the theater_ID and seat_selection (seats ranging from A1-J20 which will remain consistent for all theater rooms at every location).
- The layout DB will work alongside the movie DB after users have selected their desired movies.
- The NoSQL decision is based on how each seat chart contains a unique code (A1-J20), meaning that the data is more nonstructured depending on which seats are available or not in the layout DB. Given the layout formatting of the data, a NoSQL DB would allow for

more dynamic adjustments when seating availability changes.

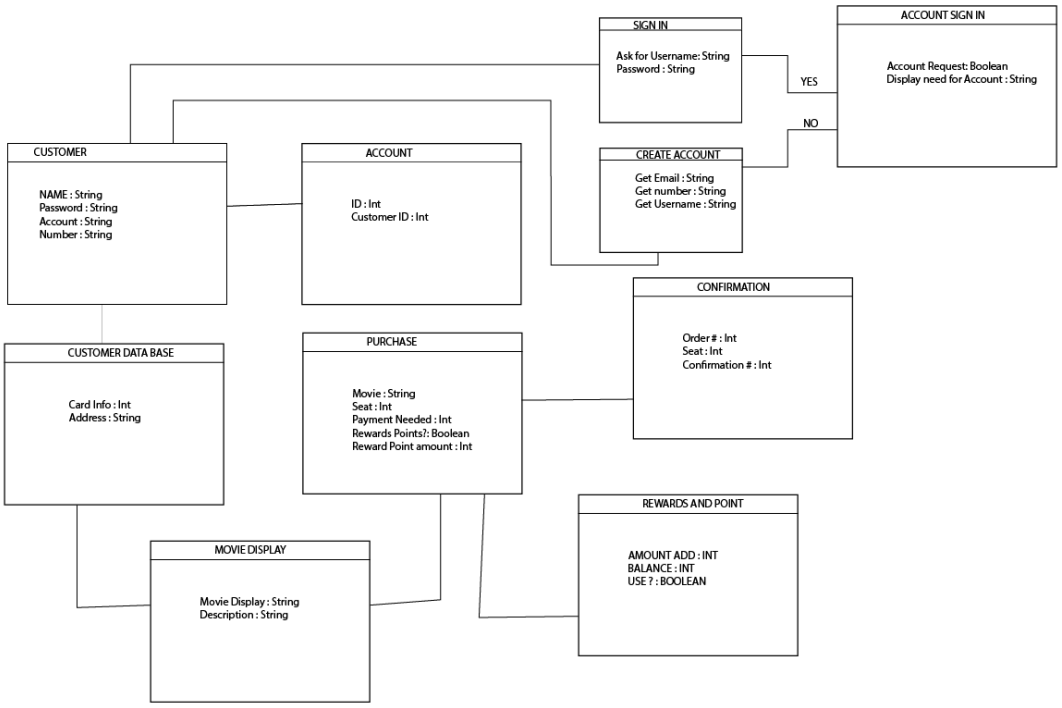
	layout DB		
WHO?	theater_ID	Seat Available?	seat_selection
PERSON 1	THEATER A134	Yes	B18,B19
PERSON 2	THEATER A134	Yes	C1,C2,C3,C4
PERSON 3	THEATER A134	Yes	I10,I11,I12

4.1.2.3 User Database

- The user DB stores all data regarding the user's personal/important information such as login information and payment information.
- This is a NoSQL database that contains the user's username, password, email, phone number (varies from user to user depending on if they have opted for it), payment_information, and reward_points.
- The NoSQL decision allows for more flexibility when working with user data. Such instances include the ability to efficiently manage optional fields ie. phone numbers and handle encrypted payment information in a secure fashion.
- The user DB will work closely with how the system handles transactions and receipts ie. the Transactions Records Server (TRS), as well as supporting user authentication on login.

	user DB					
WHO?	PHONE NUMBER	EMAIL	USER	PASSWORD	payment_information	rewards_points
PERSON 1	(681)544-9360	madmax03@fakemail.com	MaxM0103	max114213!	Credit: Discover *****123	500
PERSON 2	(143)887-0986	jdoe@fakemail.com	JohnDoe43	Doeboy378$$	Debit: Chase Bank *****123	900
PERSON 3	(143)887-0986	aligatoral@fakemail.com	Al123	alman&&!23	Credit: American EX *****123	200

4.1.3 UML Class Diagram



4.1.3.1
Description of the UML diagram above on the basic functionality of the website and its purpose. This basic functionality includes the sign in process, the display of customer data once signed in, the selection/display of movies for customers, The purchases process of said selected movies and the rewards/points program implemented for deals.

- Sign in Process

Description of the Sign in Process Block: This process outlines the steps for user authentication in the application, determining whether the user needs to sign in or create a new account. It includes input prompts for both existing users and new users, ensuring a seamless transition between account access and account creation.

- **Account Request** (Boolean): Whether the user has an account or not.
- **Display Need for Account** (String): If the user needs an account to proceed, the system asks them to sign in or create a new account.
- If the user has successfully signed in:
- **Ask for Username** (String): Prompt the user to input their username.

    - **Password** (String): Prompt the user to input their password.
    - If the user couldn't successfully signed in create an account:
    - **Get Email** (String): Collect the user's email.
    - **Get Number** (String): Collect the user's phone number.
    - **Get Username** (String): Collect the desired username.
- Display of Customer Data

Description of the Display of Customer Data Block: The display of customer data includes essential fields for user identification and security. Each data point is designed to enhance user experience while ensuring privacy and security through encryption and adherence to best practices. The structure of the data collection emphasizes both user convenience and stringent data protection protocols.

    - **Name** (String): Capture the user's full name.
    - **Password** (String): Store the user's password.
    - **Account** (String): Store the username or account identifier.
    - **Number** (String): Store the user's phone number.
    - more private Customer Database:
    - **Card Info** (Int): Store the user's card information (likely for payment).
    - **Address** (String): Store the user's address.
    - About the Account Information
    - **ID**(Int): A Unique Identifier will be made for the account in the system to make it easier for the system itself to define this number will be encrypted through the process of the AES-256 encryption to protect every user
    - **Customer ID**(Int): THe Unique identifier for the customer that is typically the user/members username that will be used to identify their account
- Selection/Display of Movies

Description of the Selection/Display of Movies Block: This section outlines the process of selecting and displaying movies within the application. It focuses on presenting users with a comprehensive view of available movies, including essential details that aid in their decision-making.

    - **Movie Display** (String): Display available movies for the user to choose from.
    - **Description** (String): Display details or descriptions of the movies.
- Purchases Process

Description of the Purchases Process Block: This section describes the steps involved in the purchasing process for movie tickets within the application. It outlines the essential components necessary for users to select a movie, choose their seats, and complete their payment.

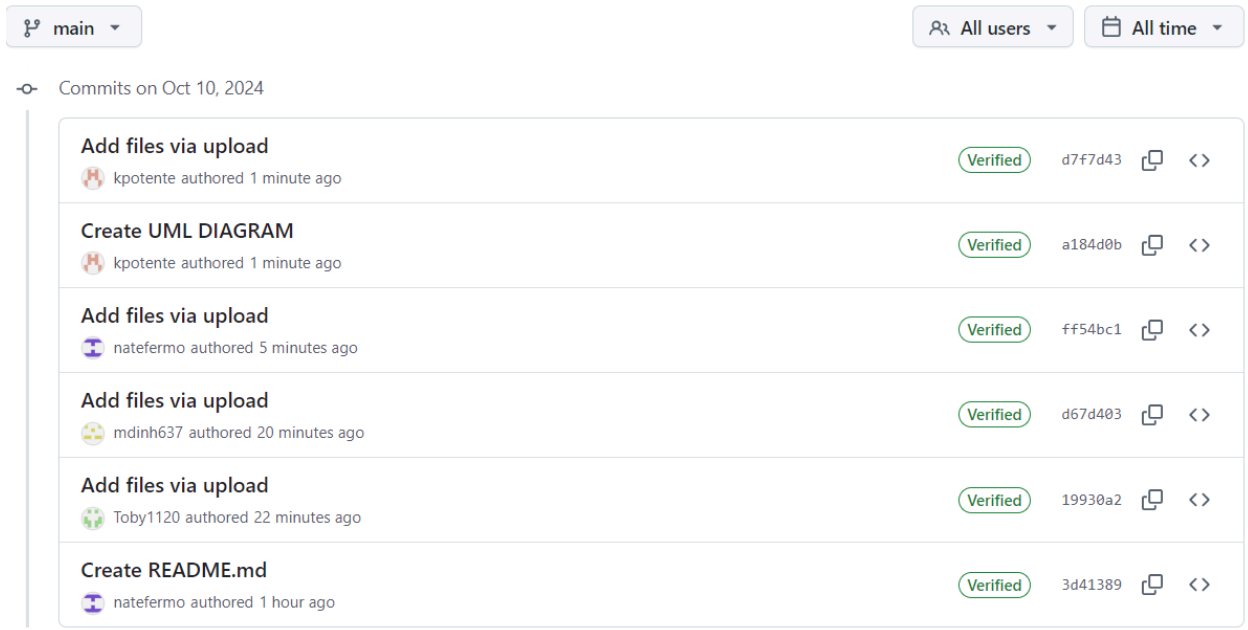
    - **Movie** (String): User selects a movie.
    - **Seat** (Int): User selects a seat.
    - **Payment Needed** (Int): Display the amount of money needed for the purchase.
- Rewards/Points Program

Description of the Rewards/Points Program Block: The Rewards/Points Program is designed to incentivize user engagement within the application. It allows users to earn rewards points through various activities, which can later be redeemed for discounts and special offers.

    - **Rewards Points** (Boolean): Whether the user has earned rewards points.

- **Rewards Point Amount** (Int): Amount of points the user has accumulated.

(Proof of commit from group members)
https://github.com/natefermo/CS-250-Group-9-Movie-Ticketing-System-/commits/main/



4.1.4 Development Plan and Timeline

4.3 Data Flow Diagrams (DFD)

4.2 State-Transition Diagrams (STD)

5. Change Management Process

Identify and describe the process that will be used to update the SRS, as needed, when project scope or requirements change. Who can submit changes and by what means, and how will these changes be approved.

5.1 Unit Tests

5.1.1: Verify new accounts get a 50% discount off their first two tickets after a successful email verification

Target/Scope: This test case is designed to verify whether the system recognizes new accounts. It will also ensure that the system will apply the correct discount at the payment page and that the payment history will display the expected total.

Test input:

- New account is created with a unique email address and phone number.
- Select a movie, theater, and showtime
- Choose a single seat, proceed to payment page
- Check whether correct total is shown at the payment history
- Repeat previous steps with two seats selected this time around

Pass/Fail Criteria:

Pass: The payment page should show that the first ticket's total is halved. When testing the second time by selecting two seats, the payment page should only apply the discount to the first ticket. The payment history should show both of these results.

Fail: The system does not properly apply the discount to one or both of the tickets. The test case will also fail if the discount is applied to the third ticket as well.

5.1.2: Verify that the user is only able to reserve at most 20 tickets at a time.

Target/Scope: This is designed to verify that users are restricted to reserving a maximum of 20 tickets at a time. The system shall prevent the user from proceeding to the payment page if they exceed the ticket limit and should display the appropriate error message.

Test input:

- Select a movie, theater, and a showtime
- Attempt to select 21 or more seats on the seat selection page during the booking process
- Try to proceed to the payment page

Pass/Fail Criteria:

Pass: The system shows an error message when the user selects their 21st seat, displaying that the ticket limit has been exceeded. It also will prevent the user from proceeding to the payment page until the user only has a maximum of 20 seats selected on the theater map.

Fail: The system does not display an error message when the user selects 21 or more seats. The test case may also fail if the system allows the user to proceed to the payment page even if it does display the error message to the user.

5.1.3: Verify whether system correctly responds to users attempting to book tickets to a sold out showtime

Target/Scope: This test case will aim to establish how the system responds to attempts by users trying to book tickets for a sold out showtime. It aims to validate that the system will correctly handle these attempts by notifying the user and preventing them from proceeding to the payment page.

Test input:

- Select a movie and a theater.
- Choose a showtime that is fully booked
- Try to select seats
- Proceed to payment page and receive an error

Pass/Fail Criteria:

Pass: When a user attempts to select their seats, the system should display an error message that indicates there are no seats available for that showtime. It shall also prevent them from proceeding to the payment page.

Fail: The system allows the user to select seats from the theater map from a sold out showtime. The test case may also fail if the error message is not displayed to the user, or if the user is able to proceed to the payment page despite the sold out showtime.

5.2 System Tests

5.2.1: Verify that when users create an account, valid inputs are used for the required information for the success of a valid account.

Target/Scope: This test case will verify whether or not new users looking to create an account for our website use valid and unique information. Such information includes a unique username and password that complies with the system's requirements, as well as a valid email address (phone number is optional).

Test input:

- User inputs their first and last name (does not need to be verified but is required to be filled out).
- User inputs a unique username in the registration page of the website.
- User inputs a password that complies with the system's requirements (ie. includes at least 1 unique character, an uppercase letter, and a number).
- User inputs a valid and correctly formatted email address.
- User can opt to input a valid phone number.
- Upon submission of information for account creation, the system will check to see the validity of all imputed information.
- Successfully create the account and store information in the database.
- Receive email/SMS notification notifying the creation of the account.
- Repeat the steps but with invalid inputs (ie. username already in use, password that does not meet the system's criteria, incorrect email address, and/or incorrect phone number if opted).

Pass/Fail Criteria:

Pass: The system will notify the user through an email/SMS notification inquiring about the confirmation of the account's creation. Upon accessing this link/message and confirming, the user's account will be successfully made and saved into our database. In the cases of incorrectly imputed information, the system will request the user to use another username, input a valid password, request the use of another email address, and/or request the use of another phone number.
Fail: The system either generates the account with an improper username/password into the database or does not save a validly created account into the database (email and phone number input should not be affected as they are handled through third-party services ie. Gmail or SMS gateways).

5.2.2: Verify that when users access the website without signing into an existing account, the system will automatically associate them with guests without the added benefits of having an account.
Target/Scope: This test will verify whether a user is accessing the website through an existing account recorded in the database or not. Once the system identifies that the user is not logged into an existing account, they will be treated as a guest without the added benefits provided to those who access the website with an existing account (ie. reward points for movie ticket purchases, email/SMS notifications for limited time deals/sales, etc).
Test input:

- User accesses the website without the system recognizing them as being logged in to an existing account.
- The user can access and browse the website's features.
- The user will succeed in selecting and making movie purchases, but will not receive reward points per movie ticket bought.
- Log into an existing account and attempt to make purchases on movies again.
- Receive reward points per movie ticket purchased.

Pass/Fail Criteria:
Pass: The system treats the user as a guest while still having access to the functionalities of the website. Users can browse movies, select movies and seatings, and finalize purchases without the benefit of reward points/deals.
Fail: The system fails to recognize the user as a guest and offers exclusive deals tied to an account as pop-ups/codes on the main page. Users will be able to use these discounted deals without being logged into an account. The system provides reward points to the user despite having no account to associate and register those reward points (ie. saved through IP).

5.2.3: Verify when the user selects a theater the movie is actually showing
Target/Scope: This test case will verify whether or not there is a movie showing at the location desired. When a movie is tested each movie will have data of which movies are displayed. If the movie will be displayed at the location, then there will be an option to purchase the ticket.
Test input:

- User searches for the desired movie from the list of movies.
- User selects a movie and showtime.

- User selects a movie theater location that currently does not have the movie at said showtime or at all.
- User redirected to theater selection and asked to select another location that has the movie available.

Pass/Fail Criteria:
Pass: System prompts user to reselect another theater location for desired movie.
Fail: System proceeds with the theater selection despite there being no movie in that theater's database.

5.3 Integration Tests

5.3.1: Verify that when users finalize and submit their payment information on the transaction page, that information will be transmitted securely through the bank/payment gateways.

Target/Scope: This test will verify whether the user's payment information is correct through the bank/payment gateways in a secure form of transmission. In the case that it is, then the bank's system will proceed with the verification and notify the system to finalize the payment. In the case that users do not opt to have their payment information saved, then none of that data will be recorded in the log server aside from the receipt of the transaction.
Test input:

- User finalizes their desired movie selection, showtime, seating, and theater location.
- User inputs any desired and available deals/offers (if applicable and logged into an existing account).
- User confirms and goes to the payment page.
- User selects their desired payment method; "credit/debit card" for the following banks: Chase Bank, Bank of America, and Wells Fargo. Can also select the payment options PayPal and Visa.
- User enters their payment information.
- User clicks "Confirm Payment".
- User receives email/SMS notification confirming payment.
- Repeat the steps while selecting another payment option.

Pass/Fail Criteria:
Pass: The bank's system successfully and securely receives the user's payment information before processing its validity. The bank will notify our system about the approvement before finalizing the payment successfully. Users with an account will receive reward points upon a successful transaction.
Fail: Users who are logged in do not receive reward points despite a successful transaction. The system may also incorrectly validate the user's payment process and proceed with it regardless.

5.3.2: Verify that users can only finalize purchases after selecting a valid seat that is still available on the seating map as a form of integration.

Target/Scope: This test case will verify whether or not the seat selected is available for purchase. If the seat is not available it will not allow the user to purchase and will prompt the user to select another seat that is available (ie. indicated through highlighted seats that are not shaded out).
Test input:

- User clicks on desired seat (highlighted indicating it is available) on the seating map.
- System recognizes the that the seat is available in the database that stores the seating arrangement and availability for each movie theater location.
- User will be able to proceed to finalize purchase with selected seats.
- Another test case scenario, select seat that is already occupied on the seating map by another user (indicated by a faded out seat icon).
- System will notice its occupation and deny the user to proceed to purchase.

Pass/Fail Criteria:
Pass: The system allows the user to proceed to the transaction module after selecting an available seat to finalize purchases.
Fail: The system does not recognize the selected seat as an unavailable one (selected by another user) and allows the user to proceed with the purchase.

5.3.3: API Integration Module

Target/Scope: A verification process that the system can fetch movie data from external APIs such as where we buy the movie IP from without failures. Some Pre-Conditions would include: The application is installed and configured correctly, and API endpoints are available and functional. This Test should ensure the error message aligns with user-friendly practices and provides guidance for resolution. This test should be executed in various environments to confirm consistency in error handling.

Test input:

1. Launch the application.
2. Navigate to the "Movies" section.
3. Trigger the fetch operation for movie data from the external API.
4. Simulate a failure in the external API (e.g., by disconnecting the internet or using a mock service that returns an error).
5. Attempt to load movie data again.

Pass/Fail Criteria:
Pass: An appropriate error message is displayed, indicating the API is unavailable.
Fail: No error message is displayed, or the message is incorrect.

5.3.4: Failure Payment Process Module

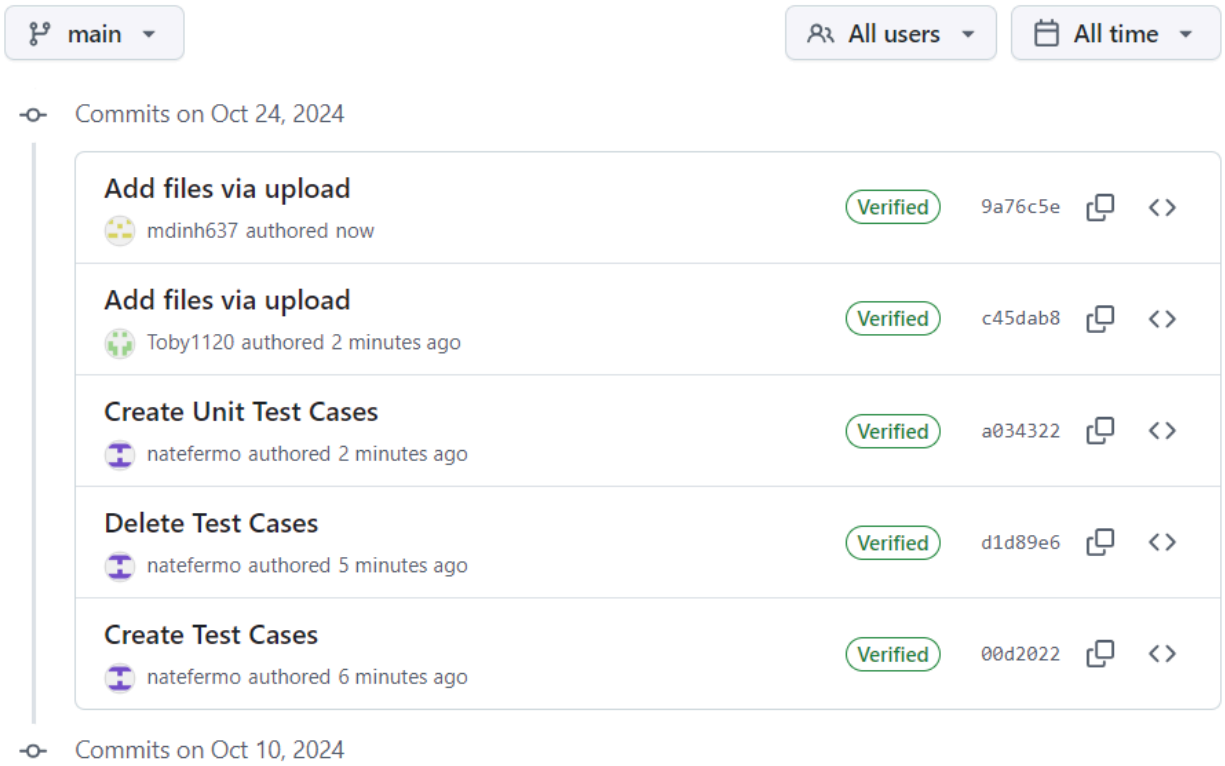
Target/Scope: A verification process that analyzes when the payment system fails due to unrecognized payment information, the user will then be prompted to re-enter payment details. Some pre-conditions include: The user has selected tickets and is on the payment page. Payment processing services are operational. This test hopes to verify that the prompt is clear and user-friendly, providing options for resolution. It aims to ensure that the application handles different types of unrecognized payment information consistently.

Test input:

1. Launch the application.
2. Select a movie and showtime.
3. Choose the number of tickets.
4. Enter valid payment information that is not recognized by the bank (e.g., a blocked card or an unregistered account).
5. Attempt to finalize the payment.

Pass/Fail Criteria:
Pass: User is prompted to either select another payment method from supported cards (e.g., Visa, Chase Bank, Bank of America, Wells Fargo) or re-enter card information.
Fail: User is not prompted correctly, or the prompt does not provide options for reentering payment details.

Test Cases Sheet
https://docs.google.com/spreadsheets/d/1WKea9bmt-N2eKijaIMLWgEEOI8K9tTDM/edit?gid=967985635#gid=967985635
Proof of github:
https://github.com/natefermo/CS-250-Group-9-Movie-Ticketing-System-/commits/main/



A. Appendices

Appendices may be used to provide additional (and hopefully helpful) information. If present, the SRS should explicitly state whether the information contained within an appendix is to be considered as a part of the SRS's overall set of requirements.

Example Appendices could include (initial) conceptual documents for the software project, marketing materials, minutes of meetings with the customer(s), etc.

A.1 Appendix 1

A.2 Appendix 2