Movie Theater Ticketing System

Software Requirements Specification

1.0.3

03/27/24

Group #5
Tim Cho
Jahnavi Panchal
Benjamin Vanhuang

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

Revision History

Date	Description	Author	Comments
02/14/24	1.0.0	Tim, Jahnavi, Ben	Filling out until section 3.6
02/28/24	1.0.1	Tim, Jahnavi, Ben	Filling out until section 4.2
03/13/24	1.0.2	Tim, Jahnavi, Ben	Filling out Design Constraints section
03/27/24	1.0.3	Tim, Jahnavi, Ben	Software Architecture Diagram & Data Management Strategy

Document Approval

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

Signature	Printed Name	Title	Date
	Tim, Jahnavi, Ben	Software Eng.	
	Dr. Gus Hanna	Instructor, CS 250	

Table of Contents

1. Introduction

1.1 Purpose

The main purpose of this document is to effectively communicate specifications of our novel and robust movie theater ticketing system. This document is strictly prohibited to individuals irrelevant to the project. The document is exclusively intended for project personnel such as software developers, quality assurance team, project managers, stakeholders, architects, documentation team, and contractors. The Movie Theater Ticketing System (MTTS) project provides digital kiosks and a website for movie ticketing. The budget for this project is $500,000.

1.2 Scope

Movie Theater Ticketing System (MTTS) project aims to implement the most well-recognized and secure platforms, frameworks, algorithms, and cybersecurity to design and build a robust software system. Amazon Web Services (AWS) will be our primary third-party to utilize a lot. MTTS aims to provide a user-friendly online ticketing system via secure transactions. The software consists of several different agendas such as reservation system, customer service system, membership management, current showing movie system, and cookie systems. MTTS will utilize cookies to gather marketing intelligence and utilize it for the benefit of customers. Unnecessary information will be strictly prohibited to prompt customers. There ought to be solely 4 different types of user roles: MTTS personnel, Administrator/IT, Registered user members, guests. There will be multiple payment methods for the users comfortability and transactions will be communicated directly between the customer and the bank. MTTS will only receive a transaction receipt from the bank to verify legitimacy of payment.

1.3 Definitions, Acronyms, and Abbreviations

MTTS	Movie Theater Ticketing System
AWS	Amazon Web Service
SRS	Software Requirements Specification
GUI	Graphic User Interface
SMS	Short Message Service
API	Application Programming Interface
QR	Quick Response
DB	Database
SIEM	Security Information & Event Management

IT	Information Technology
IDS/IPS	Intrusion Detection System / Intrusion Prevention System
TLS/SSL	Transport Layer Security / Secure Sockets Layer
SOC	Security Operations Center
DMZ	Demilitarized Zone

1.4 References

"How to Write Effective Use Cases." How to Write Effective Use Cases?, www.visual-paradigm.com/tutorials/writingeffectiveusecase.jsp.

Whitney, Ellen. "Introduction to Gathering Requirements and Creating Use Cases." Gathering Requirements and Creating Use Cases, An Introduction, www.codemag.com/Article/0102061/Introduction-to-Gathering-Requirements-and-Creating-Use-Cases.

"Software Requirements Specifications: Best Practices and SRS." AltexSoft, AltexSoft, 15 Sept. 2020,

www.altexsoft.com/blog/software-requirements-specification/

1.5 Overview

The rest of the SRS will be an overview of the specific requirements for the system. It will cover the external interface, functional requirements, non-functional requirements, use cases, and an analysis model.

2. General Description

Factors that may affect the product include: website traffic, physical location traffic, user demand, user interface, system reliability, marketing, website responsiveness, and payment processing.

2.1 Product Perspective

The product should take info from movie databases for use as display, payments on the website should be secure, the system should accurately maintain the same information as the physical location.

2.2 Product Functions

The software should be expected to provide an interface for the user to browse movies. The user will be able to purchase and reserve a seat for selected movies. The software will be up to date and provide an overview of future screenings.

2.3 User Characteristics

The eventual users of the MTTS have a diverse demographic ranging from teenagers to elderly individuals, with different educational backgrounds, experience with technology, and technical expertise. Understanding the characteristics of these users is crucial for designing a user-friendly and accessible system that caters to their needs effectively.

2.4 General Constraints

a) Regulatory policies
b) Hardware limitations (e.g., signal timing requirements)
c) Interfaces to other applications
g) Higher-order language requirements
h) Signal handshake protocols
i) Reliability requirements;
k) Safety and security considerations.

2.5 Assumptions and Dependencies

The following factors would affect the requirements stated in the SRS:

- Industry standards, customer feedback, business needs, market trends, budget constraints, growth of the company, and security threats

3. Specific Requirements

3.1 External Interface Requirements

3.1.1 User Interfaces

Website Interface:

- Visually appealing with an assortment of current movie screenings at the home page.
- Responsive regardless of web browser, along with support for major browsers e.g., Google Chrome, Firefox, Microsoft Edge, Safari, and Internet Explorer.

Kiosk Interface:

- Fully touch screen and custom GUI to suit an in-person experience, selecting movies will be simple and intuitive, along with payment options.
- Users will have an option to change the language of the text on the kiosk with the touch of a button.

3.1.2 Hardware Interfaces

Ticket Printers:

- Connected through USB and ethernet.
- Takes info from the system to print out a physical ticket for the user.

Card Reader:

- Connected through USB and will be a part of the kiosk to allow the user to pay through use of credit card.

QR Code Scanner:

- Connected through USB to allow an employee to scan a user's QR code and serve as proof of payment

3.1.3 Software Interfaces

Database and APIs:

- Uses a database (MySQL, PostgreSQL) for data storage and retrieval of user profiles or movie details.
- Third party APIs such as IMDb will be used to obtain up-to-date information regarding movies.

Payment Gateway:

- Provides support for major credit cards e.g., Visa, MasterCard, Chase, Discover, American Express, etc.
- Payment receipts will be sent to a user's choice of communication interface

3.1.4 Communications Interfaces

Notifications:

- Users will be able to opt in/out for receiving emails or SMS for promotions, special offers, and confirmations.

3.2 Functional Requirements

3.2.1 Movie Selection & Seat Reservation System

3.2.1.1 Introduction

- This feature will allow the user to select from a catalog of movies and reserve a seat for their wanted screening time

3.2.1.2 Inputs

- User's input for either searching up a movie manually or clicking on a featured movie
- User input for selecting a specific screening time to watch a movie
- User input to select a seat during that screening time to reserve

3.2.1.3 Processing

- The system will check for if a user's selected movie will be available
- The system checks for if the different screening times will be available
- The system checks the seating availability for those selected screening times

3.2.1.4 Outputs

- Users will be able to see if certain screening times are available
- Users will see which seats are available for their selected screenings
- Users will be given a confirmation for their booking

3.2.1.5 Error Handling

- The system will not allow a user to select a seat from a screening that is already full
- If a user searches for a movie that is not in the database they will be greeted with a page prompting them to search for a different movie
- If a user's payment for their ticket does not go through, they will be notified and not receive the ticket

3.2.2 Refund System

3.2.2.1 Introduction

- This feature allows the user to get a refund if they meet the right criteria. As scams are present, this refund system enforces secure and validated refund transactions. To detect potential fraud, we will implement AWS Sage Maker to customize and train our model and use its API endpoint to detect potential fraud. There is also a specific time frame that allows refunds to be possible. If passed that time frame, refunds will not be made.

3.2.2.2 Inputs

- User input for proof of purchase such as email confirmation, transaction ID, and physical/digital receipt.
- User input for the date and time of purchase.
- User input for reason of refund; multiple choice options of common refund reasons in addition to "other" options.
- User input for refund method: credit to their original payment method, a store credit, or an exchange for a different product.

3.2.2.3 Processing

- The system checks if the date of purchase is not passed the limited time frame of possible refund transactions.
- The refund request is manually processed by our customer service department and determines if refund is appropriate or not.
- Refund request process will take no longer than 21 days.

3.2.2.4 Outputs

- Users will receive an email from the customer service department on whether their refund request has been approved and processed or denied.
- If a refund request is approved, confirmation email will be sent to the users and refund transaction will be done no later than 21 days after request.
- If a refund request is denied, an inconvenience email will be issued to the users and provide with sufficient information to the reason of denial.

3.2.2.5 Error Handling

- To detect potential fraud/scam, a customized AWS Sage Maker model will be used to flag potential fraud/scam requests. These refund requests will be manually monitored by our IT department for validation of legitimacy.
- To prevent fraud/scam, all refund requests are manually processed by our customer service department. The users' proof of purchase is manually verified via our purchase DB.
- If the date of purchase surpasses the time frame of eligible refund request policy, the system will notify the user immediately upon entering input on the software.
- In case the user still wants a reconsideration of a denied refund request, there will be a "reconsider" button at the end of the refund request denial email to be reviewed by our customer service department. These cases will be elevated to a more important stage where both the customer service department and the IT department will pay extra attention to.

3.2.3 Food Order System

3.2.3.1 Introduction

- This feature will allow the user to select from a menu of movie-theater snacks/drinks and place an order to get delivered to their seat

3.2.3.2 Inputs

- User's input for searching the menu for snacks/drinks.
- Users input for selecting a specific snack/drink that they would like to order
- Users input for making customizations for the snack/drink that they order
- Users input for finalizing their order
- Users input for confirming their seat for their snack to be delivered
- Users input for adding their payment

3.2.3.3 Processing

- The system will check for the user's selected snack/drink
- The system will check for any customizations made for their snack/drink
- The system will confirm the location of the users seat
- The system will confirm the validity of the users payment information
- The system will send the order confirmation and information to the kitchen

3.2.3.4 Outputs

- User will be able to see that their payment is processed
- User will be able to see a confirmation message that their order is being processed
- Users will be able to see a message that their order is being prepared, and will be delivered to their seat shortly.

3.2.3.5 Error Handling

- The system will not be able to process unavailable or sold out snacks
- The system will not be able to process orders for invalid payments

3.3 Use Cases

3.3.1 Use Case #1

Actor: Online Customer

The actor in this use case will be an average adult that wants to watch a movie and wants to reserve a seat before the screening. The actor already has purchased tickets before and knows how to navigate the website.

Pre-Condition:

1. Customer has access to the internet and knows how to get to the movie ticketing website
2. Customer already has registered an account on the website

Customer Login:

1. Customer selects their chose web browser
2. The customer will search on their web browser for our movie ticketing website
3. The customer will then login to the website with their account (assuming they already have an account on the website)

Movie Selection:

1. The customer will be presented with a plethora of relevant movie screenings
2. The customer browses the catalog of movies until they find a suitable movie
3. The customer selects their chosen movie

Screening Time Selection:

1. The customer will then select what day they will watch the movie
2. The customer will then select what time in that day they will watch the movie

Seat Reservation:

1. The system will present the user with a visual representation of the available seating for a user's selected movie and timing
2. The customer will select how many tickets to buy
3. The customer will select what seats they want and confirm with the system

Payment:

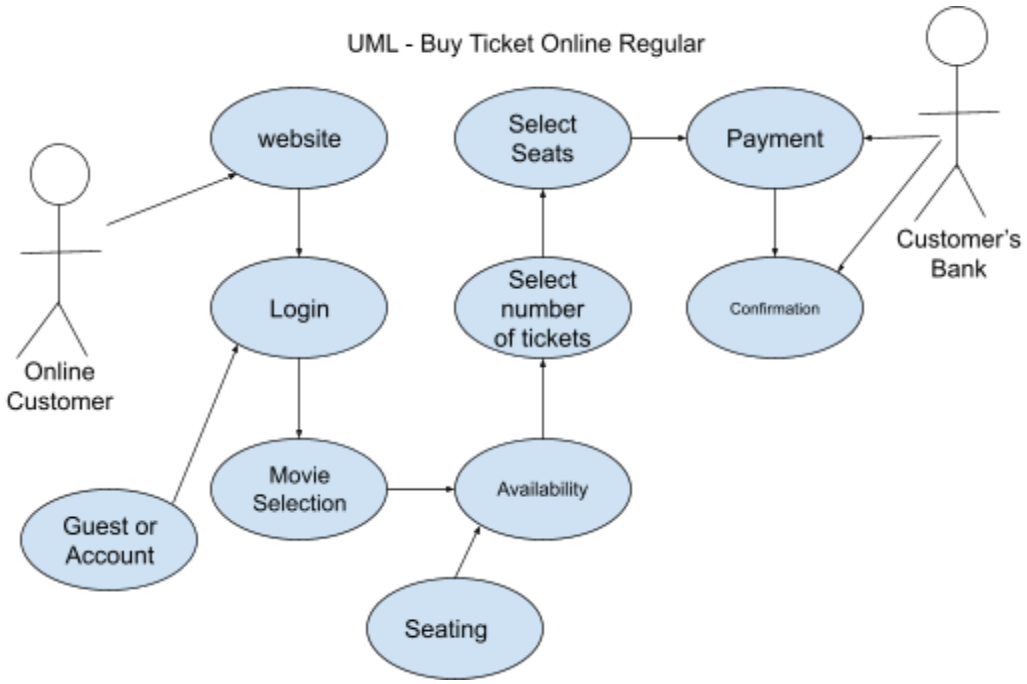
1. After selecting their preferred seat, the customer will be sent to a page to confirm their movie selection, time, and seat reservation.
2. The customer will check a boxes as to where they want to receive their payment confirmation (SMS or email)
3. The customer will then confirm the information, put their payment information, and confirm their payment

Proof of Payment:

1. The customer will be presented with a page that serves as a receipt for their payment, along with a QR code to serve as a ticket for the actual movie theater

Post-Condition:

1. Customer has a receipt for their payment
2. The customer's card is charged with the correct amount
3. The customer has a QR code for use in the movie theater



3.3.2 Use Case #2

Actor: The actor in this case is requesting a refund due to an emergency situation, specifically where the actor's wife is sent to the emergency room with an unknown cause. The request refund is denied and the actor submits a "reconsideration" request to finally get approved and get his/her refund.

Pre-Condition:

1. The customer has the purchase of proof via an email confirmation.
2. The customer knows where the refund link is in the homepage.

Refund Request:

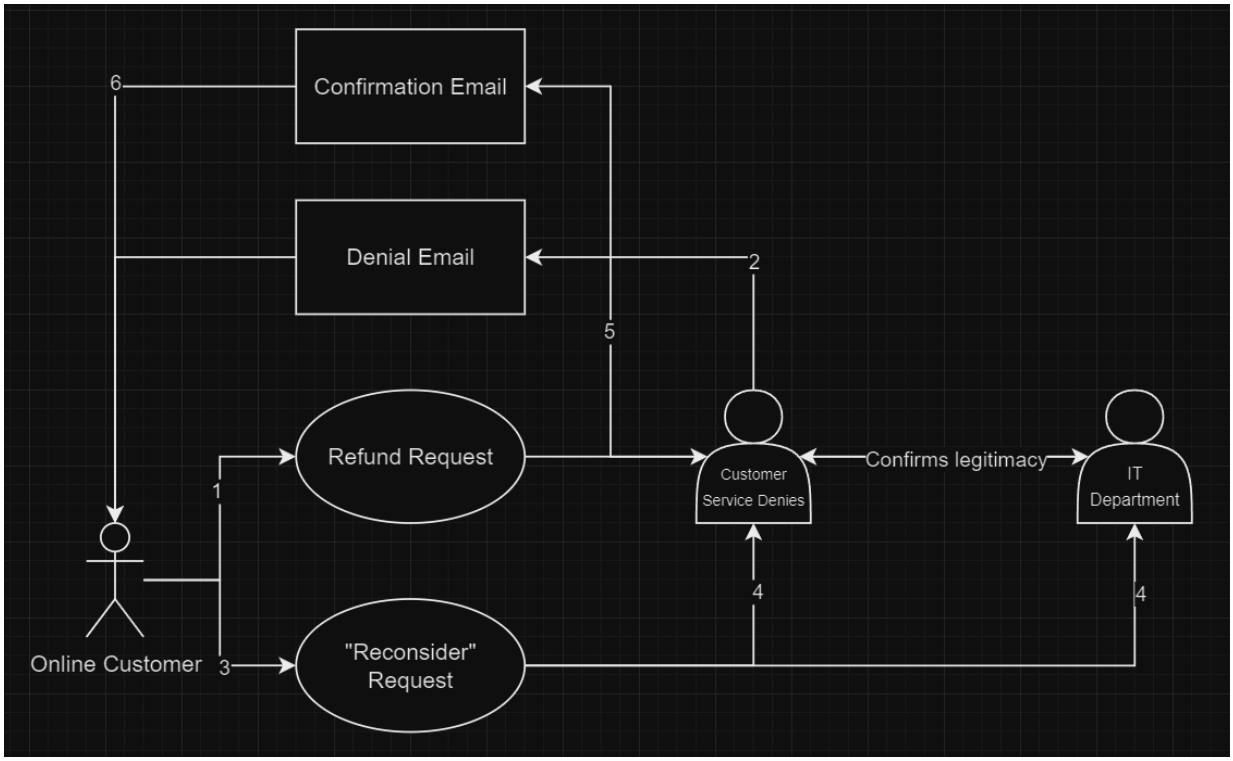
1. The customer inputs all necessary information required for a refund request.
2. It will take the user no later than 21 to get a response back from the customer service.

Refund Request Status Update:

1. The customer will be notified through their email about their refund request status.
2. If a refund request is approved, he/she will receive the money no later than 21 days upon request.
3. If a refund request is denied, he/she will receive an inconvenience email with an option to "reconsider" the request.
4. In this use case, the actor's refund request got denied and he/her resubmitted the request with the "reconsider" option in the email.

Post-Condition:

1. Since the refund request was “reconsidered”, the customer service department and the IT department got to have a closer look at the request and were able to finally approve the refund request.
2. It took the actor no later than 21 days to successfully get his refund approved and transacted.



3.3.3 Use Case #3

Actor: A regular customer who is already in the theater and wants to order snacks or drinks from their seat.

Pre-Conditions:

1. Customer has bought a ticket and is seated
2. Customer has access to the internet and snack/drink menu

Snack/Drink Selection:

1. Customer is prompted with the catalog snacks and drinks
2. Customer selects what drinks or snacks they want

Payment:

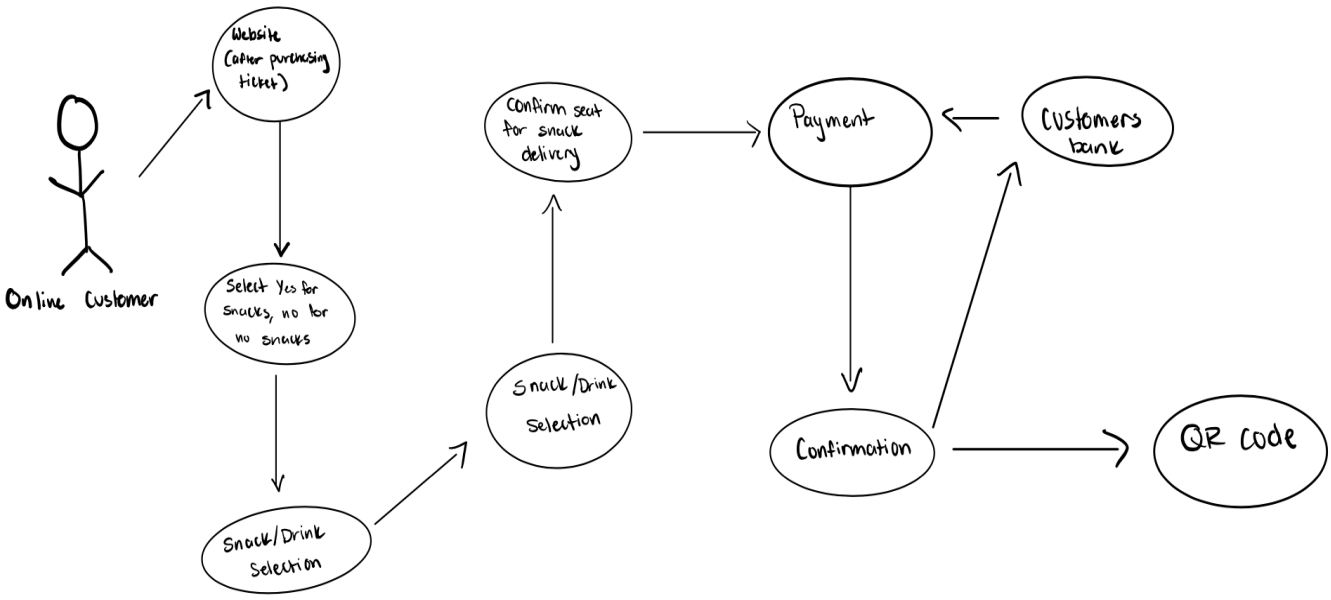
1. Customer is prompted to enter their card information
2. Customer will then put in their card info and confirm their payment

Payment Confirmation:

1. After confirming their payment, the customer will be presented with a payment confirmation page that also has an estimated wait time for their food
2. The payment confirmation page will also serve as a way for staff at the theater to know which person ordered the snacks/drinks

Post-Conditions:

1. Customer has a confirmation for the payment of their snacks/drinks
2. Customer is notified of when their snacks/drinks are ready
3. Customer will have their snacks/drinks



3.4 Classes / Objects

3.4.1 <Class / Object #1>

- Class: User ID/Password for account
- Object: User logging into ticketing system

3.4.1.1 Attributes

- **Attributes** of the class and object include the user logging into the system to proceed with the selection process

3.4.1.2 Functions

- Functions of the class and object include the user logging in to use the functionality of all three use cases

3.4.2 <Class / Object #2>

- Class: Selection process for movie, seat, and snack
- Objects: Specific movie, seat number, specific snack

3.5 Non-Functional Requirements

3.5.1 Performance

- Our system implements a membership account where we provide discounts and offers to specific ranges of age or by luck.
- 5 minutes will be given to complete the purchase process. If not finished by 5 minutes, the customer needs to redo the purchase process.
- After purchase, there will be a quick feedback pop-up for customers to submit.

3.5.2 Reliability

- Our system will be able to provide service to up to 10 million concurrent users.
- Our system supports 3 main languages: English, Spanish, and Swedish
- Our system has a separate DB for all the movie theaters, also partitioned to each individual theater.

3.5.3 Availability

- Movie tickets will be available for only 20 local theaters in San Diego; the standard time zone will be pacific time (PST).
- Our system will be up and running 24/7 hours.
- In case our website/service goes down, we have an error page to properly let the users know.

3.5.4 Security

- We have a separate IT department that monitors all traffic among all the servers required for the MTTS project.
- We will have a centralized server that implements SIEM via the Splunk program to monitor logs and traffic by the IT department.
- We will also implement IDS/IDP such as the Snort program to take necessary actions to detect potential hackers and malicious network activity.
- Apart from setting up host firewalls, our system will have a centralized server that functions as a firewall. Cisco networks will be used to allow and deny specific ports.
- Our website will have a verified TLS/SSL certificate to access.
- DMZ network infrastructure will be implemented to separate the public and private network users.

3.5.5 Maintainability

- Our IT department has a group of SOC analysts, ranging from level 1 to level 3, to monitor network traffic and is in charge of incident response in case things turn to the worst.
- All MTTS personnel will undergo monthly Cyber Awareness training where the IT department will educate about the basic Cybersecurity hygiene.
- For security purposes, our IT department has a group of penetration testing experts where they are granted to hack our system and report it in efforts to find vulnerabilities and fix them.

3.5.6 Portability

- MTTS only supports browser access; no mobile app.
- In case of disaster, backups will be made on different servers.
- Migration of DB and servers will be implemented in case it is necessary.

3.6 Inverse Requirements

- The system will not allow scammers and frauds.
- The system will not satisfy every customer.
- The system will not allow bulk buyers to buy tickets and resell them. (20 max ticket purchase at one time).

3.7 Design Constraints

Budgetary Constraints:

- Budget for MTTS is $500,000.
- The scope of the project will affect the cost.
- The cost of the project needs to be sustainable through sales.
- We need to consider how saturated the market is for movie theaters.
- Cost of hiring people to maintain the system or work in the theater.
- Cost of website domain and servers.
- Cost of unexpected expense

Payment Security:

- We need to consider different payment methods on the website.
- The system needs to not save payment information to make payment info more secure.
- The system needs to send payment information directly to the bank and nowhere else.
- Our AWS Sage Maker will check for potential frauds, however its data will then be deleted from the system and only its status will be recorded in our Centralized Log Server (CLS).
- We need to consider cryptocurrency as another payment method on the website.
- As the quantum computing era is coming very soon, we need to research methods to enforce a secure transaction since quantum computing may make our system vulnerable to brute-force attacks.

System Maintenance:

- People need to be hired to keep system maintenance, the amount of people on this also needs to be adjusted according to how much traffic the website gets.
- System servers need to be maintained and kept up.
- Backup is always required to run a big project like the MTTS.
- Backup servers will be physically located at a different location from the main server room in case of natural calamities.
- Server rooms are only accessible by personnel granted authorization directly from the Director of the IT Department.
- Background check for even the janitors to ensure the server room is exploit-free.
- Firewalls, DMZ, and many more enterprise network security solutions will be implemented to maintain system security and uptime.
- Necessary training will be provided to IT personnel to train for what to do in case of server failure.

Hardware Constraints:

- Depending on the size of our server room, the data storage will be limited.
- In case of data storage reaching full capacity, there will be another server that will receive overflowing data storage.
- In case of overflowing data storage, the IT department will implement more servers to ensure more data is available.
- In case of hardware failure, there will be backups ready to keep the server rolling.

Data Privacy:

- Our system will not read or store any payment data; instead, it will be directly handled by the payer and the designated bank system to ensure data privacy.
- Our system will keep track of the activity of each user for marketing purposes and coupon events.
- Our system ensures to implement every single security solution possible to implement secure communication with our server and the users.
- Cookies will be implemented to gather intel for marketing purposes.
- Our system will not share user data without prior notice and authorization.
- We only keep hashed passwords and not text passwords to enhance security.

3.8 Logical Database Requirements

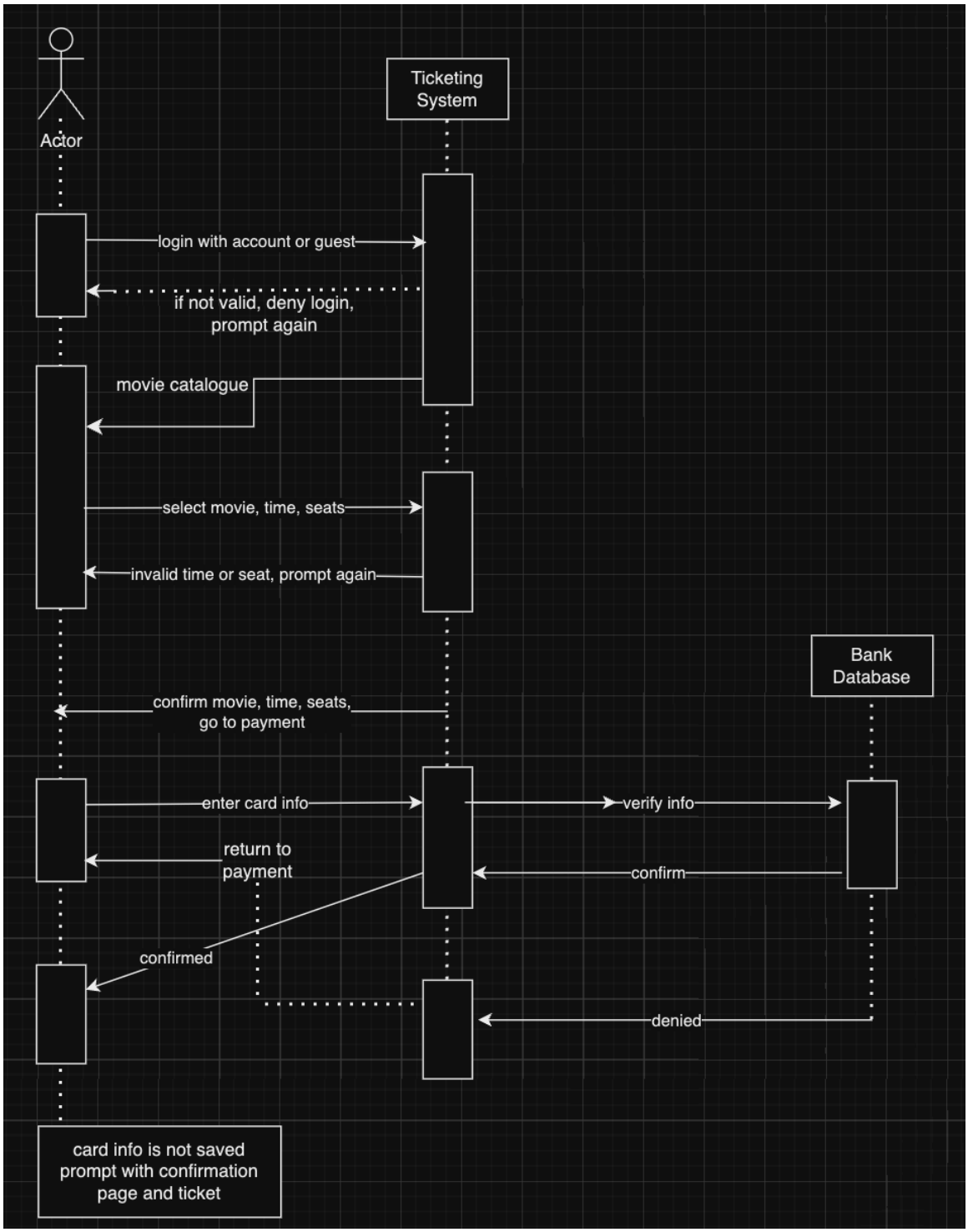
Will a database be used? If so, what logical requirements exist for data formats, storage capabilities, data retention, data integrity, etc.

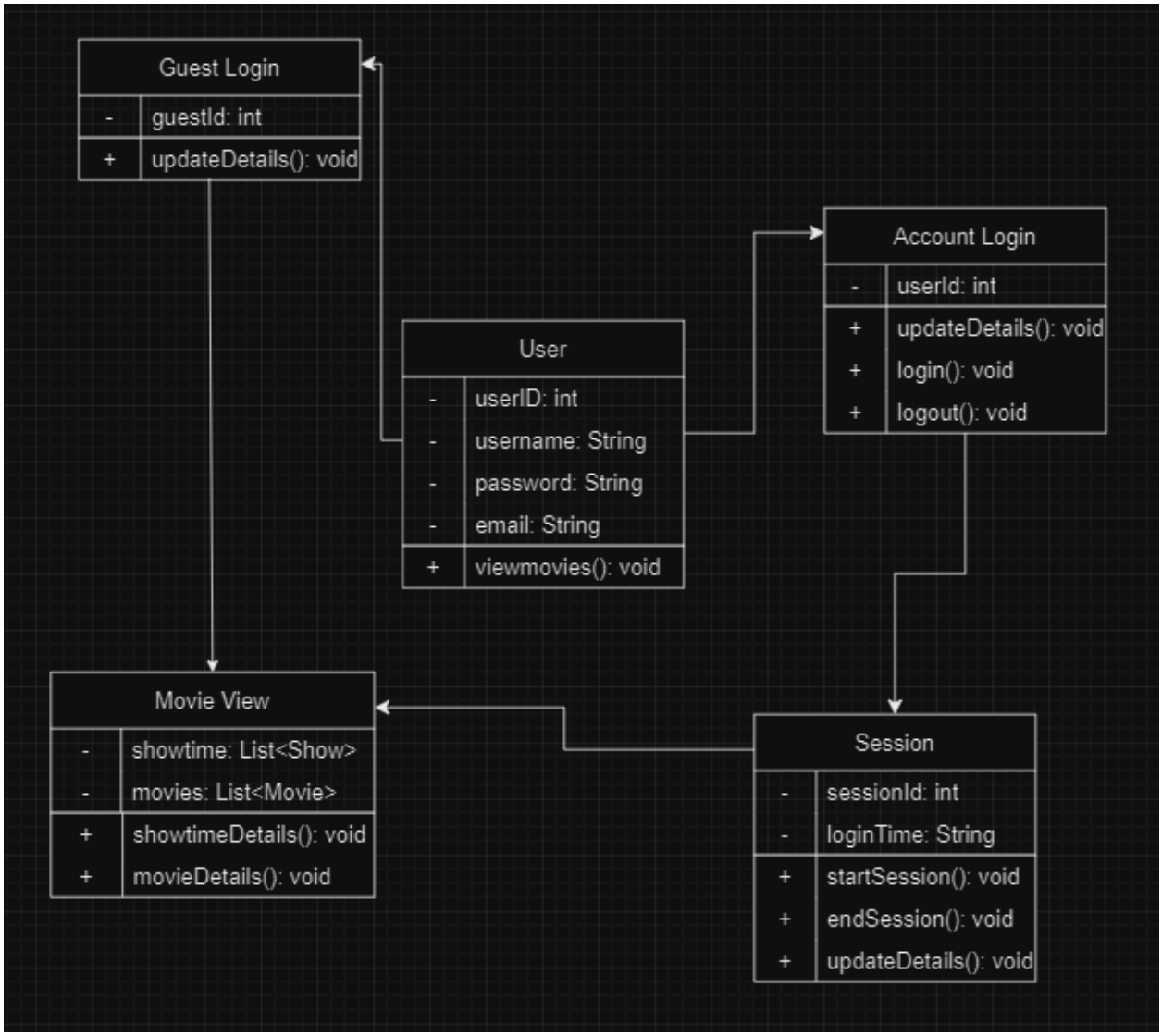
3.9 Other Requirements

Catchall section for any additional requirements.

4. Analysis Models

4.1 Sequence Diagrams

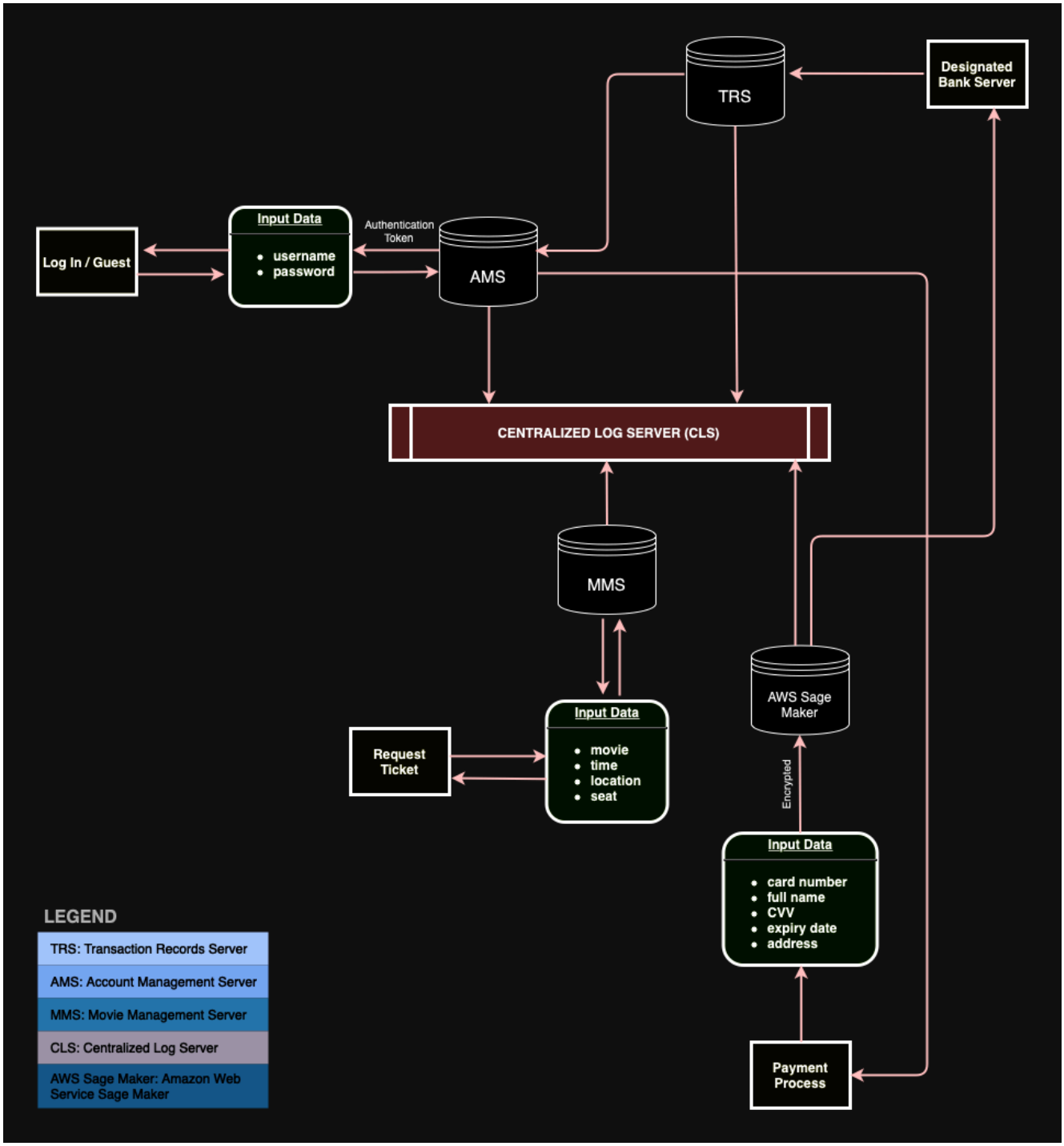




User Class:

1. Stores information for the userID as an integer, stores the username and password as a string and calls viewmovies() method for the movie view class.
2. It branches to the Guest Login class when user info is not provided or goes to account login where it takes the userID and calls updateDetails() method to update info for the userID. Then calls login() or logout() depending on user input, which will log the user in or out.
3. After Account login it goes to session class where it contains the sessionID and loginTime which goes into the updateDetails(). startSession() info is recorded into the updateDetails() and endSession() will call logout().
4. The Movie View class is then called, which contains the showtime and movie lists and provides details for those lists through showtimeDetails() and movieDetails().

4.3 Data Flow Diagrams (DFD)



Log-in / Guest:

1. Users input the following information if they have an account: username and password.
2. Users bypass the login page if they click the “Login as Guest” button.

3. Assuming the user has an account, their account will be authenticated by the Account Management Server (AMS). If provided credentials are verified, an authentication token will be given to the user.
4. All of the traffic logs will be sent and stored in the Centralized Log Server (CLS).

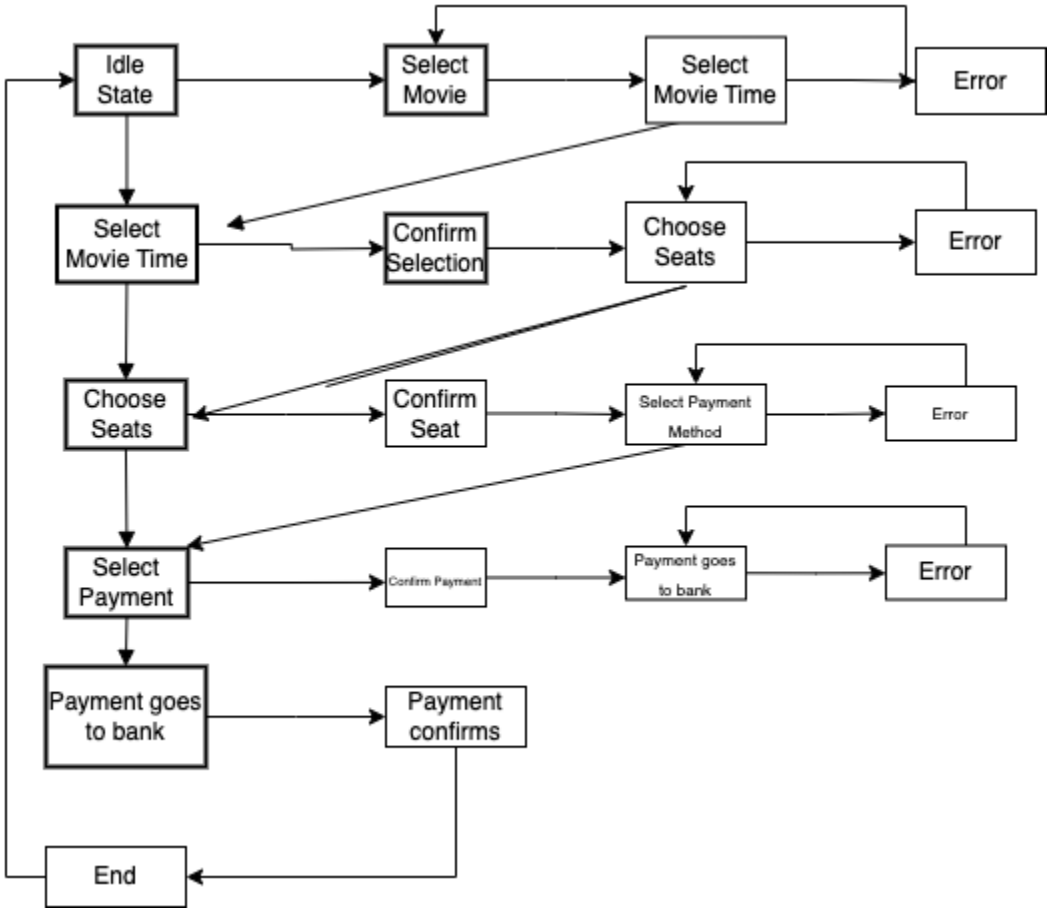
Request Ticket:

1. Users input the following information to buy a ticket/s: movie, time, location, and seat.
2. Those data will be sent to the Movie Management Server (MMS) to verify movie availability for the allotted data.
3. All of the traffic logs will be sent and stored in the Centralized Log Server (CLS).

Payment Process:

1. Users input the following information: card number, full name, CVV, expiry date, and address.
2. These encrypted data will first be sent to the AWS Sage Maker Server to check for potential frauds. If detected, it will automatically deny payment and its log will be sent to the IT department.
3. All of the traffic logs will be sent and stored in the Centralized Log Server (CLS).
4. If it passes the potential fraud detection process, it will then be sent its encrypted data to the Designated Bank Server where the payment process will be held.
5. Once the bank approves the transaction, its receipt will be sent and stored in the Transaction Records Server (TRS) and it will be linked to the AMS of the account who purchased it.
6. All of the traffic logs will be sent and stored in the Centralized Log Server (CLS).

4.2 State-Transition Diagrams (STD)



1. Idle State: Welcome! Please log in to select your movie
    a. Idle state leads to movie selection and movie time selection, if an error occurs it gets returned to the movie selection state.

2. Select Movie Time State:
    a. Select Movie Time State leads to confirming selection and choosing seats, if an error occurs it goes back to choosing seats.

3. Choose seats state:
    a. Choosing a seat state leads to confirming seat selection and selecting payment method, if an error occurs, it goes back to selecting payment method.

4. Select Payment state
    a. Leads to confirmation, and sending payment info to the bank database, if an error occurs it resends to the bank database.

5. Payment going to bank state
    a. Leads to confirmation, which leads to end, which results in going back to idle state.

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.

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