Software Requirements Specification Template

Software Engineering

The following annotated template shall be used to complete the Software Requirements Specification (SRS) assignment.

Template Usage:
Text contained within angle brackets (‘<’, ‘>’) shall be replaced by your project-specific information and/or details. For example, <Project Name> will be replaced with either ‘Smart Home’ or ‘Sensor Network’.

Italicized text is included to briefly annotate the purpose of each section within this template. This text should not appear in the final version of your submitted SRS.

This cover page is not a part of the final template and should be removed before your SRS is submitted.

Theater Ticket System

Software Requirements Specification

Version One

1/31/24

Group 12
Vincent Nguyen, Joseph Miranda, William Kate

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

Revision History

Date	Description	Author	Comments
1/31/24	Version 1	Vincent Nguyen Joseph Miranda William Kate	Creation of document with initial draft of edits
1/7/24	Version 2	Vincent Nguyen Joseph Miranda William Kate	Brainstormed ideas for use cases as well as organize our thoughts into document
2/14/24	Version 3	Vincent Nguyen Joseph Miranda William Kate	Finalizing our ideas and turning our ideas into thought out words
2/28/24	Version 4	Vincent Nguyen Joseph Miranda William Kate	Created UML diagrams and team partitioning of tasks
3/13/24	Version 5	Vincent Nguyen Joseph Miranda William Kate	Created the test plans section of the SRS
3/27/24	Version 6	Vincent Nguyen Joseph Miranda William Kate	Created Architecture Design and Data Management section of the SRS

Document Approval

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

Signature	Printed Name	Title	Date
Vincent Nguyen	Vincent Nguyen	Software Eng.	3/27/24
Joseph Miranda	Joseph Miranda	Software Eng.	3/27/24
William Kate	William Kate	Software Eng.	3/27/24
	Dr. Gus Hanna	Instructor, CS 250	

Table of Contents

1. Introduction

This SRS provides an overview of the TTS that is being produced. This includes the project's purpose, scope, functions, characteristics, constraints, requirements, and references to other documents in this SRS. Further details about the aspects of the TTS are found below in their respective document referenced locations.

1.1 Purpose

The purpose of this ticketing software system is to allow users to purchase tickets for a movie theater. The intended audience for this system are the general public who may like to browse or purchase movie tickets. Furthermore, this SRS document provides insight into the functionality and necessary concepts for developer insight.

1.2 Scope

The scope of this project is to create a website that offers movie seating reservations, customer support information, account management, card encryption and storage to an external server, and administration controls. This software will introduce the user to an account login interface with a captcha or creation interface to store user card information and preferences, have a catalog interface of movies available to be purchased with specified genres, ratings from account users, age ratings, theater locations, and showtimes, provide a digital, printable encrypted QR code as a movie ticket, and contain a secure method of storage and encryption/decryption of their card information for future purchases and a purchasing process of a maximum of 5 seconds. This software will not store card information from guest users.

1.3 Definitions, Acronyms, and Abbreviations

SRS - Software Requirement Specification
TTS - Theater Ticketing System
IT - Information Technologist

1.4 References

SRS Doc Example
UseCase_notes
Theater Ticketing Requirements
Theater Ticketing System Qs
Theater-ticketing-reqs
Functional vs Non-Functional Requirements: Ultimate Guide (jelvix.com)
TTS Test Plans

1.5 Overview

The rest of the SRS contains a more detailed description of the general expectations and functions of the TTS with specific requirements, scenarios, and constraints. The rest of the SRS is organized by the following: the intentions of the TTS, the functions of the TTS, the typical user characteristics, design constraints, setting assumptions, external requirements, functional requirements, use cases, software classes, non-functional requirements, other requirements, analytic models, management processes, and appendixes.

2. General Description

The following SRS document holds a detailed description of user functions pertaining to the TTS. Included is the description, characteristics for user efficiency and security are highlighted with an added emphasis on product functionality. This section of the SRS does not list specifics and only provides descriptions that are easy to understand.

2.1 Product Perspective

In comparison to other similar products of movie ticket reservation websites, the TTS will be user-friendly in its user interface with coherent buttons and legible text, provide online payment options for seat reservations, contain the interface and information for account creation and login, and provide theater employees special permissions to modify their locations showtimes and seat reservations.

Unlike other products, the TTS will have the option to email the customer when their movie starts in one hour and email the customer a request for them to rate the movie once the move is done which is displayed on the website.

2.2 Product Functions

The TTS will provide secure card storage with encryption for account users, give users the ability to reserve specific seats if available, contain a catalog interface of available movies with search features, give customers the ability to purchase membership subscriptions, provide movie recommendations, incorporate different pricing depending on age, displays movie show type, contain a regional search for a theater, have the ability to apply discounts at checkout, send a digital and printable qr movie ticket, email customers purchase confirmation with their ticket, solve refund and card decline disputes, display multiple theater locations, have multiple showtime options, allow users to rate a movie after viewing, warn customers when viewing mature content, provide theater employees the option to change movie setting and showtimes, and give the customer an optional 1 hour warning notification before movie starts.

2.3 User Characteristics

The typical user of this product is a customer who wants to view a movie and is purchasing a ticket online to reserve seating with their credit/debit card.
If the user is a theater employee, they can access administrative controls to change movie availability and showtimes or set seat reservations to reserve seats manually once if a customer pays in-person.

2.4 General Constraints

Some general constraints of the TTS are the hardware limitations such as server space/size and the limitations of the server's internet connection.

2.5 Assumptions and Dependencies

Some assumptions made in this SRS are that the operating system will not change that the SRS is on, the hardware can be programmed in Python, the TTS is supposed to hold 10000 users at a time and that theater staff are responsible for verifying age in-person for certain movie and ticket types.

3. Specific Requirements

This specific requirements section of this SRS will describe the general systems of the TTS with the general functions and aspects detailed of the TTS.

3.1 External Interface Requirements

3.1.1 User Interfaces

The website portion of the TTS should have an intuitive graphical user interface for buttons and options that 98 out 100 users are about to understand. The contact customer support page of the website should be updated within a maximum of 10 seconds if a theater chooses to change their support number or email address.

3.1.2 Hardware Interfaces

The TTS will run on a server connected to a wifi router with a maximum of 400 ms of latency to the user. The server will also store up to 500GB of user, movie, and encryption data.

3.1.3 Software Interfaces

The TTS will be coded using Python, stores encrypted data on a server or server service of the stakeholders' choice and offers up to 10 different online payment methods.

3.1.4 Communications Interfaces

The TTS requires an IP address, website domain, and email address.

3.2 Functional Requirements

3.2.1 General Website Functions

3.2.1.1 Introduction
This section describes all of the features to be used in the website application such as clicking on buttons and typing on a keyboard.
3.2.1.2 Inputs
The inputs for this function are the mouse and keyboard inputs.
3.2.1.3 Processing
Depending on which page and what button the user clicks and inputs on, the system will send the respective data needed to the user's computer.
3.2.1.4 Outputs
The page/interface that the user has clicked on will appear on their screen.
3.2.1.5 Error Handling
If a page is not available or outdated, an error message will be displayed.
If the server is unable to send the correct data to the user's computer, an error message will be displayed telling the user to refresh.

3.2.2 Payment Interface and Seat Reservations Function

3.2.2.1 Introduction
This section describes the payment interface and seat reserving functions and the possible scenarios that include this function.
3.2.2.2 Inputs
The inputs for this function are card information, the available seats that the user reserves, discounts from coupon codes that are applied, and the seat request of a theater employee administrator account.
3.2.2.3 Processing
After inputting the seats the user wants to reserve and their payment information, the TTS will process the payment at the price calculated and reserve the seats accordingly so that other users will not be able to reserve the same seats and so that administrators will be able to check which users have reserved which seats. If the user has a customer account, the card payment information that they have chosen to be saved will auto-fill on the payment parameters.
3.2.2.4 Outputs
The user will be emailed their purchase confirmation and tickets and be shown their tickets and seats reserved. An administrator account will be shown the emails and seat positions of all the occupied seats. An user with a customer account will be prompted to have an option to save their card information on their account for their future purchases. If the user has the options enabled, the user will be sent an email reminding them when their movie will start an hour before the movie starts and/or a rating request to rate a movie after watching.
3.2.2.5 Error Handling
If a user reserves a seat that has been reserved by another user during the processing of their payment, an error message will be displayed and the user will be prompted to be able to change their seat reservations or request for an automatic refund.
If a user refunds their reservation, the TTS will clear the seats that have been reserved and send a refund confirmation to the user.

If the card information the user provides gets declined or is incorrect the seat reservations will not be reserved and a respective error message will be displayed.

3.2.3 Administrator Account Functions

3.2.3.1 Introduction
This section describes special controls an administrator account has on the website.
3.2.3.2 Inputs
The inputs for this function are the showing that the administrator account wants to view, the seats that the administrator account wants to reserve or clear, the movie information the administrator account wants to change/create, and the theater information the administrator account was to change.
3.2.3.3 Processing
The server of the TTS will verify if the account has the authorization to make changes to the theater's section on the website then changes in the server if authorized.
3.2.3.4 Outputs
The TTS will output a success message to the administrator account if it was successful in changing the respective information and show the website page with the updated information.
3.2.3.5 Error Handling
If an error occurs, the administrator and the user of the error. The administrator will be able to see the error page and be able to troubleshoot.

3.2.4 Customer Account Functions

3.2.4.1 Introduction
This section describes the functions and features a user with a customer account can do.
3.2.4.2 Inputs
The inputs for this function are the user's login, membership purchase, and setting changes.
3.2.4.3 Processing
The server of the TTS will verify the login information of the user if the user tries to login into the customer login page or decides to create an account. The exclusive features that the stakeholder specifies that is included in the membership purchase will be applied if the input is a membership purchase. The settings of a user will change if the input requests for the settings of a customer account to change.
3.2.4.4 Outputs
The TTS will output a success message to the user such as the confirmation of the creation of the account, the acceptance of the payment information, etc.
3.2.4.5 Error Handling
The user will be given ways to recover from the error in order for the workflow to continue. An example of this would be if a card declined where the user will have the ability to try again and fix or change the card information.

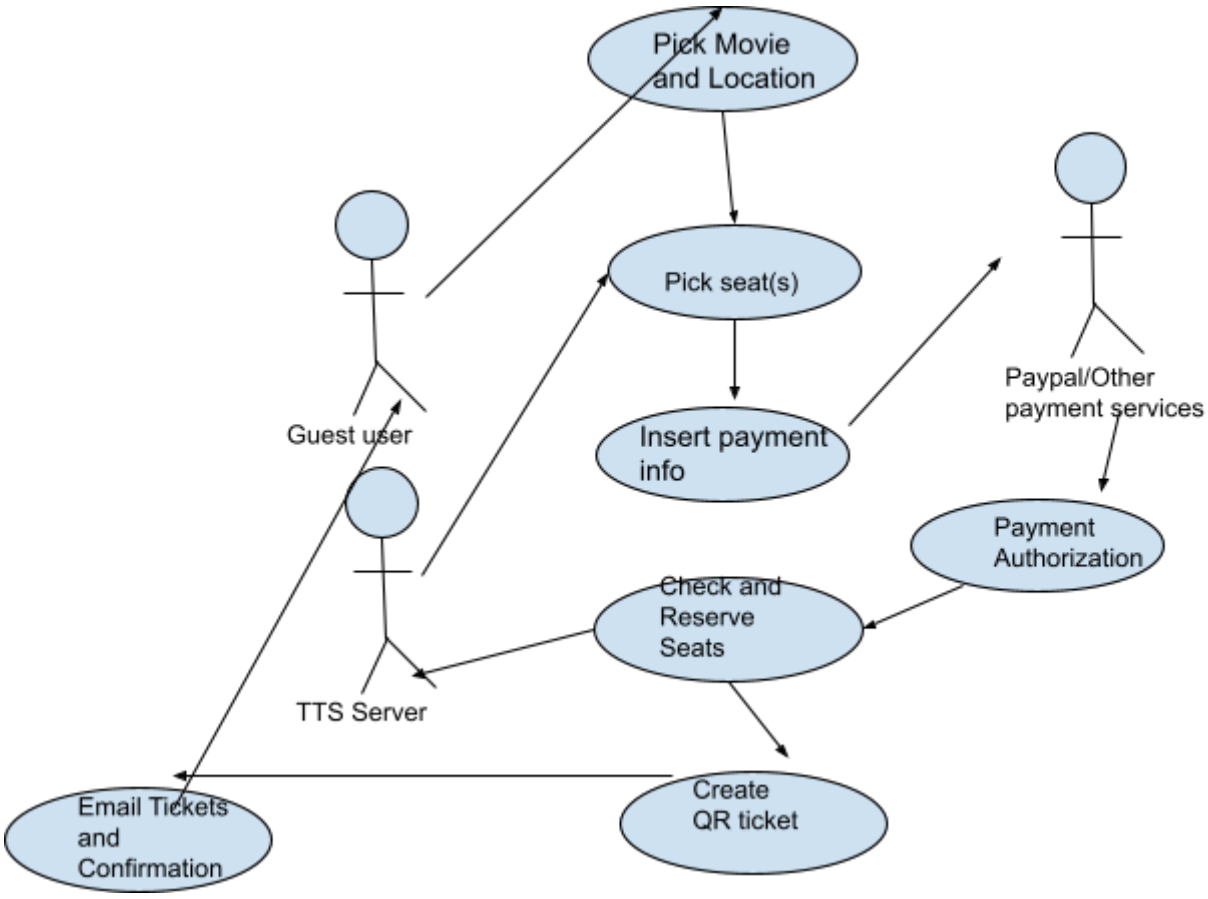
3.3 Use Cases

3.3.1 Use Case #1

A guest user wants to buy a ticket for a certain movie.

Use Case #1	Purchase Ticket(s)
Goal in context	This test case allows the user, who does not own a membership, to choose and purchase a movie and reserve the seating
Precondition	The movie is available and there is seating available
Success end condition	The user receives email confirmation with encrypted QR code of ticket
Failed end condition	Card was declined, movie seating was filled, connection cut out, server timed out.
Primary actor	Guest user
Secondary actor	• PayPal/other • TTS Server
Trigger	The user enters the website
Description	• Step 1 - User picks movie and location • Step 2 - User picks seat(s) • Step 3 - User inserts payment info • Step 4 - Payment gets authorized • Step 5 - System check and reserve seats • Step 6 - System creates encrypted QR code for ticket(s). • Step 7 - QR code gets emailed along with confirmation of purchase
Extensions or Variations	• If Card is declined, a prompt to re enter a different card information is shown • If the movie is filled, a prompt saying the movie is filled is gown and told to choose a different movie or time • If the connection cut out, the TTS server will save the last inputted information before the connection cut out • If server timed out, the user will given the prompt of server being timed out

	and returned to beginning of the site



3.3.2 Use Case #2

A user wants to cancel/refund a reservation for a movie that they have purchased tickets to.

Use Case #2	Cancel/Request Refund for Ticket(s)
Goal in context	This specific test case provides the user a way to cancel their previously purchased ticket reservations, while also prompting them for a refund request.
Precondition	The user would most likely want a refund for canceling their reservation.
Success end condition	The users would be connected with either customer service or an auto generated software protocol to assist and make the

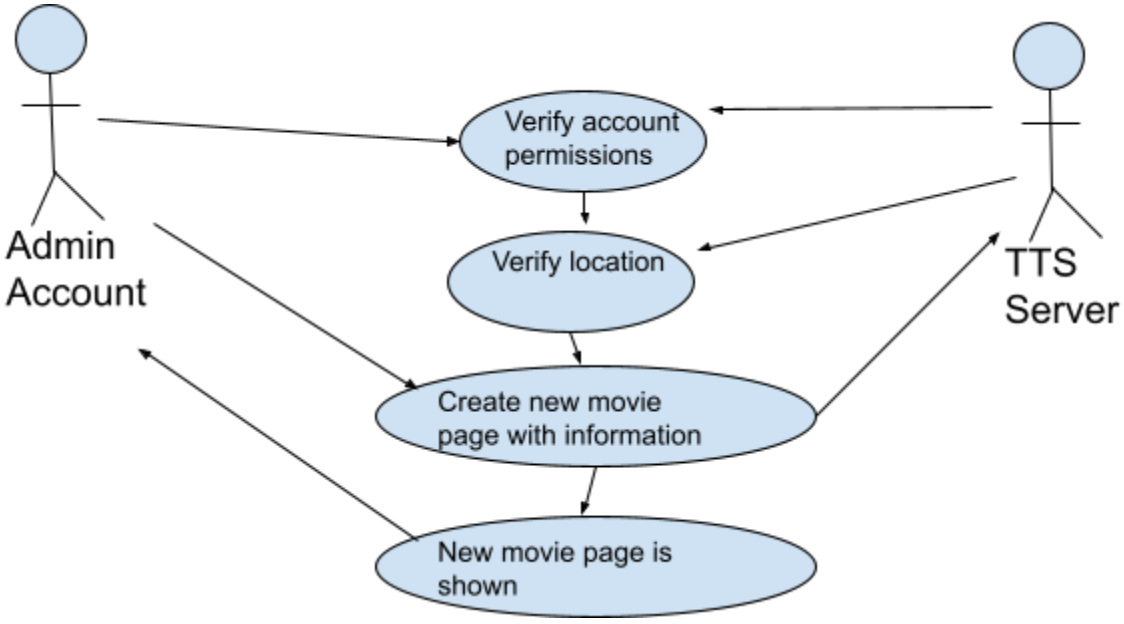
	refund process smooth and efficient.
Failed end condition	The user can not access help services through the TTS software.
Primary actor	Ticket Holder/Customer
Secondary actor	TTS Server/Administrative Team
Trigger	The user selects the cancel/refund function on the website.
Description	• Step 1 - User would like to cancel reservation and clicks “Cancel Reservation” • Step 2 - Asks for secondary confirmation on ticket cancellation • Step 3 - Prompts user to screen for assistance in a refund if said user would like one
Extensions or Variations	• TTS extends to in-person headquarters where the user can be greeted by a TTS educated member. • If connection gets lost, cancellation will halt and user will have to initiate cancellation request again

3.3.3 Use Case #3

A movie theater employee with an administrator account wants to create a new movie option for their theater location.

Use Case #3	Create new movie option
Goal in context	This specific test case provides an administrator account the option to create new movies that are not in the TTS database.
Precondition	The user of the administrator account has all the information they want to put on the web page which includes: genre, showtimes, age rating, and seating capacity.
Success end condition	The TTS server is updated with new movie

	page information and any user will be able to view the new movie page on the catalog.
Failed end condition	The admin client fails to connect to the TTS server, the admin client does not put enough movie information to create a page, or the admin client tries to modify movies for a different theater location that they are not authorized to do.
Primary actor	Administrator user
Secondary actor	TTS server
Trigger	The administrator account clicks on a button to create a new movie option.
Description	• Step 1 - The admin account gets their permissions and location verified by the server. • Step 2 - The admin account inputs movie details and specifics. • Step 3 - The server stores and saves the new movie page information to display to future users. • Step 4 - The admin account gets a success message and is shown the new movie page created.
Extensions or Variations	• The admin account does not fill enough information to create a movie. The TTS notifies the admin to fill in the required fields. The movie page creation goes on as usual. • The admin account tries to create a movie page for a theater location that is not available for their account. The TTS notifies the admin that their location access to that particular theater is denied. The admin account fixes their location field and movie page creation goes on as usual.



3.4 Classes

3.4.1 Seat Reservation

3.4.1.1 Attributes
Stores seat reservation information
3.4.1.2 Functions

- 3.4.1.2.1 General Website Functions
- 3.4.1.2.2 Payment Interface and Seat Reservations Function
- 3.4.1.2.3 Administrator Account Functions
- 3.4.1.2.4 Customer Account Functions

3.4.2 Card Information

3.4.2.1 Attributes
Stores card information that has been encrypted
3.4.2.2 Functions

- 3.4.2.2.1 Payment Interface and Seat Reservations Function
- 3.4.2.2.2 Customer Account Functions

3.4.3 Movie Information

3.4.3.1 Attributes
Stores a catalog of movies to be searched with rating, trailer, showtimes, age options and movie types.
3.4.3.2 Functions

- 3.4.3.2.1 General Website Functions

3.5 Non-Functional Requirements

3.5.1 Performance

The TTS will process requests and inputs with no more than 400 ms of latency and handle up to 10000 concurrent users.

3.5.2 Reliability

The TTS's server will be running at all times unless closed for maintenance or other reasons.

3.5.3 Availability

The website portion of the TTS will be available on all web browsers.

3.5.4 Security

The website will utilize a captcha for account login and creation to prevent bots, send users to a secure payment site to input payment information, and encrypt data before sending data to an external server service with 60 layers of encryption for storage.

3.5.5 Maintainability

The TTS will be documented precisely with 15 minutes of reading time so that ITs will be able to quickly perform maintenance when required.

3.5.6 Portability

The TTS's website application will be accessible on any device with a web browser and internet access so portability is limited by the device size.

3.6 Inverse Requirements

3.7 Design Constraints

Age verification will need to be done in-person due to the limits of this software not being able to properly verify age. The amount of server space for movies, user reservations, and account information are constraints if movies and theater location are continually added. Based on the company's pricing policies and discount rates, the software system will have to adjust to adhere to user requirements.

3.8 Logical Database Requirements

A main database with all the global servers will be located at headquarters for use around the world. This will give each individual user, regardless of the user being a customer or administrator, the ability to use the software system based.

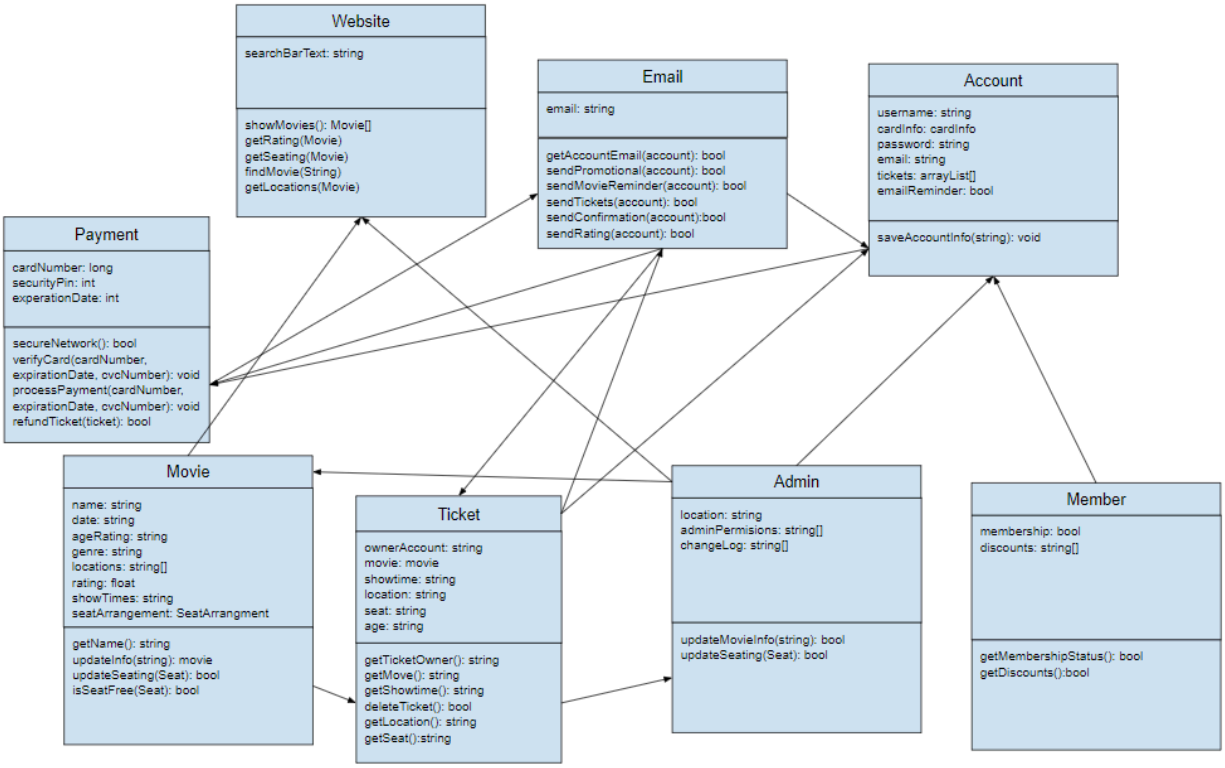
3.9 Other Requirements

In order for software systems to work seamlessly, the user must have a strong, secure network.

4. Analysis Models

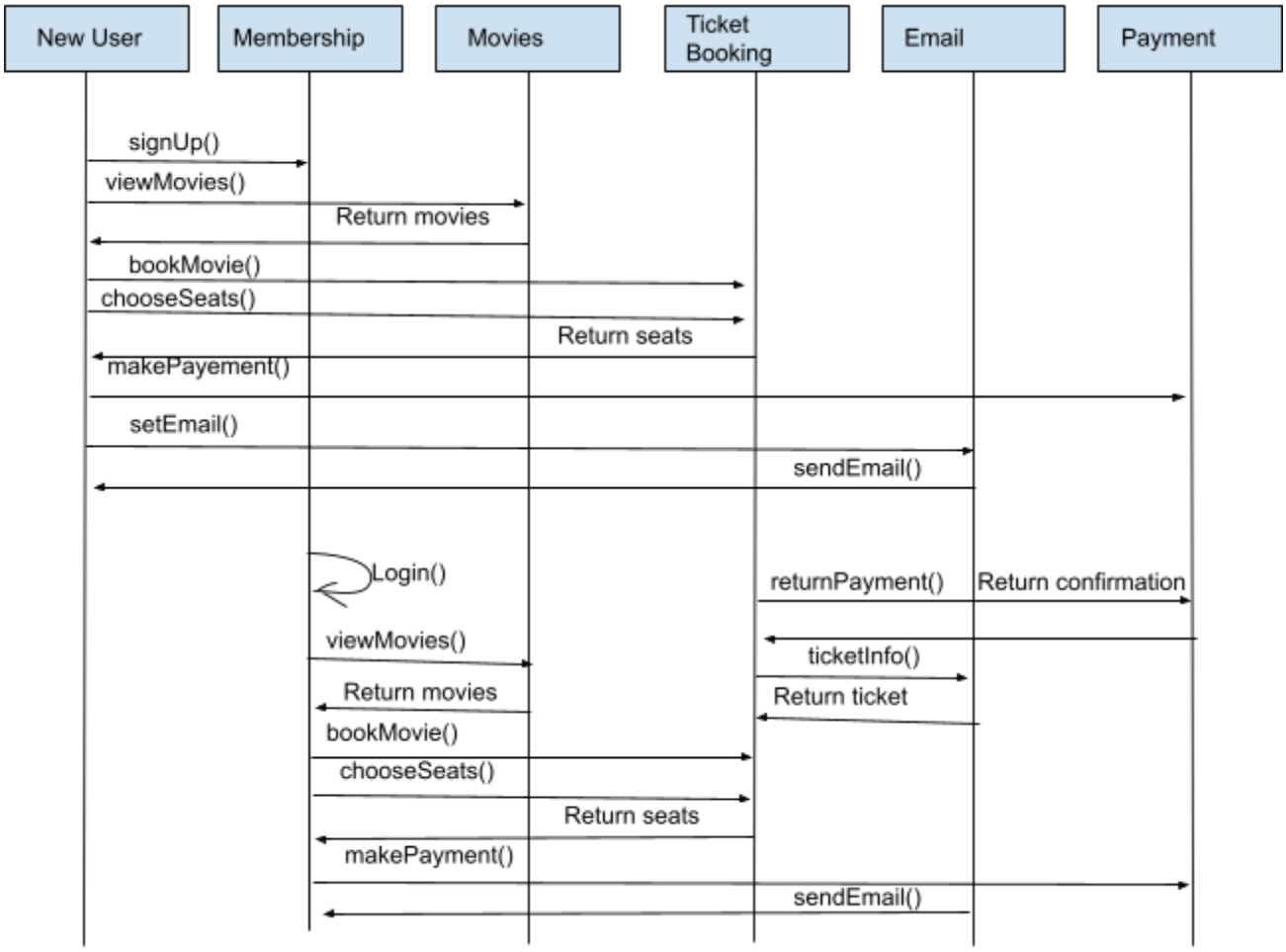
This section of the SRS lists all the analysis models that are used to design the TTS.

4.1 UML Diagram



The UML above shows the overall flow of the Theater Ticketing system with specific references to necessary classes, functions, and variables. The diagram is subject to change based on new necessities that arise, but one should clearly be able to follow along. A website module was added to the UML diagram above.

4.2 SWA Diagram



This SWA Diagram shows the typical user or member of the system booking a ticket to a certain movie. There are different types of actors which are membership users and new users. Both new users will have similar methods such as viewMovies(), makePayment(), and etc. The differences between the two groups will be the login() method and the setEmail() method.

4.3 Development Timeline

In order to make our design a properly functioning code, we must implement each class specified in the UML diagram and create all necessary variables and class functions. For our action plan during Week 1, William is going to work on creating all necessary variables and necessary data holders for user input while Joseph and Vincent are going to focus more on the algorithmic aspect. Once we have all done our respective parts, during Week 2 we are all going to come together to first address the issues that we encountered in our individual developments, then work on creating a solid foundation for the code. Moving forward to Week 3, our goal is to work out all kinks and format the code so that the algorithm works smoothly with all classes, variables,

and functions. From here based on our group requirements to ensure our software system is satisfactory, we will continue the review and edit process.

5. Test Plan

This section of the SRS will discuss more in-depthly some of the test cases and scenarios that may occur while using the TTS.

5.1 Verification Test Plans

5.1.1 Test Plan 1

The first of the test plans tests the movie, payment, seating reservation, and email features of the TTS. The goal for the first test plan is to test the basic functionality, timing, and requirements specified in this SRS. The first five test cases of the *TTS Test Plans* are some of the test cases that will be tested during this test plan. We will be testing the payment loading times, seat reservation positions, email verification, and connection error catching.

5.1.2 Test Plan 2

The second round of test plans focuses more on the actual person dealing with the system rather than the processes in which the user would execute when using the system. The main goal of the second half of the *TTS Test Plans* is to make sure that the user is valid and that all things they do on the website gets linked to their account (if they are not a guest user). The five test cases that we have for this section ensure that both the user and the admin can navigate the site based on their respective roles. In terms of the user, there are test cases that ensure a full gathering of information for their account creation alongside proper verification for tickets to be sent to a valid email address. Additionally, the admin has override features that allow them to edit movies and viewing times on the overall system.

6. Architecture Design and Data Management.

6.1 Data Management Strategy

Card Database

ID	Card Number	Expiration Date	Security Code	Zip Code
0001	****1234	01/28	123	12345
0002	****2345	02/29	234	23456
0003	****3456	03/30	345	34567

0004	****4567	04/31	456	45678

This is considered a SQL diagram. This SQL holds important information pertaining to card info such as card number, security code, etc. The information is then given a unique ID.

Member Database

ID	Name	Member? Yes/No	Phone Number	Email
0001	Mark Walsh	Yes	1 (847)-483-1543	Mwalsh@gmail.com
0002	Susan DuBois	No	1 (365)-736-2788	SusanD008@yahoo.com
0003	Alex Sanger	No	1 (951)-444-6511	ASanger@hotmail.com
0004	Mary Allen	Yes	1 (408)-386-9456	MaryAllen@aol.com

This specific SQL diagram keeps track of all member information, including the name, if they are a member, phone number, and email address

Purchasing Database

ID	Card ID	Member ID
0001	0001	0001
0002	0002	0002
0003	0003	0003

The SQL diagram shown here uses the ID numbers from the previous databases in order for purchases of tickets. It also gives it a unique ID.

6.2 Trade Off discussion

By splitting the databases into three, we avoid having complications of having the whole system failing in case of a crash. However in doing this, expenses for servers will be more expensive but because the TTS stores sensitive card information, it is necessary to have more than one database. The TTS uses a SQL database for account and ticket information, and card information because the information stored is important and confidential. This will be more costly but this measure is necessary. The TTS uses a noSQL database to store movie information because of the ease of scalability and the quicker access to information considering how movie information is constantly sent to the user when a user loads a movie page. The disadvantages of this is that the database is less secure.

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