Talkies At Ease

Software Requirements Specification

Version 3

June 10, 2024

Group 9
Aiko Lauryn San Nicolas
Krish Kiranbhai Monpara
Michael Morgan

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

Revision History

Date	Description	Author	Comments
<5/25/2024>	<Version 1>	<Michael, Aiko, Krish Monpara>	<First Revision>

Document Approval

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

Signature	Printed Name	Title	Date
	<Your Name>	Software Eng.	
	Dr. Gus Hanna	Instructor, CS 250	

Table of Contents

1. Introduction

This document is intended to organize and analyze the entirety of our Movie Ticketing software system, through detailing the entire needs of what our system must contain. It also will entail the strengths of our system and define all the functionalities our software will be capable of. The system's requirements will also be provided within this document.

1.1 Purpose

The overall purpose of this document is to collect, maintain, and organize all of the concepts and objectives that will be used to define the system, with the intended consumers in mind. Also, we will plan and organize a course of action that we believe the product will be used for as a way to understand the project better and predict possible changes in the future for this software, although these changes may later be changed.

To reiterate, this SRS is intended to help sort our ideas and better plan out our understanding of what this software system will be. We will describe the software's user interface, hardware/software requirements, and more. This document was designed to be able to act as an aide in the software delivery lifecycle (SDLC) process.

1.2 Scope

The movie ticketing webpage intends to make buying tickets for movies more accessible and easier, by allowing customers to order these tickets right from their phones. The system will be based on a relational database that can store related movies, possibly through genres, to allow ease of searching for users. The database for the webpage will be able to support users purchasing tickets.

1.3 Definitions, Acronyms, and Abbreviations

DB	Database
UX	User Experience
GUI	Graphical User Interface
API	Application Programming Interface
SSL	Secure Sockets Layer

1.4 References

- https://squareup.com/us/en/invoices

1.5 Overview

The remaining sections of this document provide a general description, including consumer characteristics, the hardware system, and the functional and data requirements of the product. We will break down the general description of the project in section 2 of this document. The section following the general description gives the data, non-function, and function requirements. It also incorporates the external interface requirements and gives a detailed description of all of our requirements.

2. General Description

This document speaks on our mission of creating a movie ticketing system that allows easier access for users to encourage a simple route to order movie tickets. This document also will explain the major features that will be implemented in our designed system. This section does not comprise our precise requirements for the software yet it allows you to understand those requirements simply. The following SRS will explain the designed functions with consumer constraints in mind.

2.1 Product Perspective

The product functions as the core idea of processing user inputs that includes a summary of the movies, showtime, number of tickets/seats, and theatre location. It will have user accounts that will record movie purchases, recommendations, or similar upcoming movies. At the closing request of purchase, it will demand payment gateways that support multiple ways of payment and support APIs for third-party integrations

2.2 Product Functions

- **Frontend** Allow user flexibility to book, schedule, cancel, select, or purchase the available service of tickets, showtimes, and movies in the software. A function of the user having the freedom of creating an account, accessing its history record, modify their recommendations, able to sign up for the rewards system and smooth UX.
- **Backend** Report general bugs or crashes to the management team for fixations. Report daily performance rate, updates and required improvisations.

2.3 User Characteristics

Users of the system will be able to find information on any movies and their pertaining showtimes at the given theater, and related setting (date/time) and purchase available tickets to the movie for the showtimes. The customer will ultimately be able to do the following functions:

- Purchase a movie ticket
    - Will be able to search for all available showtimes for the movie given date/time
    - Can be purchased within two weeks in advance, or up to 10 minutes past showtime
    - A person can only purchase up to 20 tickets at one time.
    - Discounts will be shown upon the selection screen, i.e. student, military or senior ticket prices, which will be verified with ID.me
    - Users will be able to see assigned seating charts for the deluxe theaters, but not for standard theaters as they are first-come first-serve for seats.

- Refund a movie ticket
    - Refund must be within 1 hour prior to showtime, and in person

- Save movie tickets onto the phone for offline use.

- Users will be able to see how many tickets have already been sold for the movie, and the cost per ticket before purchase.
    - Users will be able to see the number of tickets available after selecting a movie and showtime
    - Users will be able to see the price of the tickets upon selecting seats, e.g. standard pricing for standard seats, or deluxe seating pricing, if available

Employees or other management will be able to perform these following functions:

- Upload new movies and their schedules for the next week
- Keep track of possible number of tickets that could be sold given one theater

2.4 General Constraints

The System should hold use of software up to 1000 users at a time without lowering the performance of UX. In some cases of instant use of software during new releases the load might shoot up to 150%, such instances must be handled without crashing.

2.5 Assumptions and Dependencies

The operating system is assumed to have the user consent for using third-party for collections and management of user activity. While the user privacy is invaded the system is assumed to also protect them from illegal party frameworks who might destroy any users data or exploit it. Thus depending on signed Terms and Conditions (T&C) the software must take such instances.

3. Specific Requirements

3.1 External Interface Requirements

3.1.1 User Interfaces

- The user interface for this shall be compatible entirely through browser on mobile phones, iOS or Android.

- The user interface shall be implemented through software packages like Squarespace, or use of HTML language

- The user interface will be able to display all of the available movies that can be configured onto the page.

- The user interface will allow users to directly select which movie they'd like to buy tickets for.

- The user interface will allow consumers to see all available information, in regards to the movie such as showtimes, type of theater and available tickets.
    - The interface will show available seats for the respective theater. (e.g. total seats for standard theater is 150, whereas deluxe theaters will contain 75 seats.)

- The user interface will allow users to receive updates on any conflicting information when trying to purchase movie tickets. (e.g. showtime no longer available, tickets no longer available)
    - The user interface will be updated within real time to accurately show which movie tickets are still available for consumer purchase

- The interface will display a timer, in which it will count down from 5 minutes to show how long users will be allotted after beginning the purchasing process before automatically vacating the tickets for others to purchase.

- The interface will ask for user feedback after each purchase, feedback being in range of a selection of smiley faces for people to choose from.
-

3.1.2 Hardware Interfaces

This system will require an internet connection to be used, and as such, all required hardware must be connected to the internet for usage. Examples included would be Modem, WAN-LAN connection, etc.

3.1.3 Software Interfaces

- This webpage will employ the use of its configurator to identify all of the components that the user will need.
- The site will make use of the Content Manager to get the movie specifications, information and promotions.
- The software system will communicate with other systems like FreshBooks or SquareInvoices to identify available payment methods , validate the payments and process payment.
- The system will employ a credit management system for handling all available financial exchanges.
- The software system will work with a CRM (Customer Relationship Management) System to provide any consumer with available support.
- The software system will communicate with a Sales system, such as Zoho, for ticket order management.

3.1.4 Communications Interfaces

The movie ticketing system shall use a HTTP protocol in order to access communication over the internet, and any intranet communications will be through the IP protocol.

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 <Functional Requirement or Feature #1> - Graphical User Interface

3.2.1.1 Introduction

The system shall provide a uniform look. The system shall provide a preview, digital image for each movie showing. The system shall provide use of icons and toolbars.

3.2.1.2 Inputs

The user will scroll through listed movie options and input their preferred movie.

3.2.1.3 Processing

The system will process the user's preferred movie, showtime, and theater.

3.2.1.4 Outputs
The system will provide the user with a digital movie ticket.

3.2.1.5 Error Handling
The system will be updated with new movies and digital images, and older movies and digital images will need to be removed.

3.2.2 <Functional Requirement or Feature #2> - Personal User Account

3.2.2.1 Introduction
The user will be able to make an account with their personal information on the system.

3.2.2.2 Inputs
The user will input their name and contact information.

3.2.2.3 Processing
The system will maintain the user's personal information.

3.2.2.4 Outputs
The system will create a unique database pertaining to the specific user.

3.2.2.5 Error Handling
The system will have to establish personal accounts for multiple users.

3.3 Use Cases

3.3.1 Use Case #1 - Buy movie ticket
The user will be able to purchase a movie ticket at a given theater at their preferred time and date.
Actors: User
Basic Flow:

1. User decides on movie and specific theater
2. User will input their preferred movie and theater into the system
3. User chooses a preferred seating, time and date to watch said movie
4. User will input their preferred seating, time and date into the system
5. User will then be prompted for their payment information for the transaction
6. The system will accept payment
7. User will receive movie ticket(s) accessible through a mobile device

Alternate Flows:

1. User decides to cancel their transaction before payment
2. User decides to see movie at a different location, time, or date before payment

3.3.2 Use Case #2 - Refund movie ticket

The user can choose to get a refund for their purchased movie ticket up to 1 hour before showtime.
Actors: User
Basic Flow:

1. User decides to not attend the showtime of the movie
2. User will select the available option to get a refund on a movie ticket
3. User inputs information pertaining to their transaction
4. User will receive a payment equal to their initial purchase price

Alternate Flows:

1. User decides to not receive a refund

3.3.3 Use Case #3 - Review a movie

The user will be able to leave a review for any movie that they have purchased a ticket for, after the given showtime is over.
Actors: User
Basic Flow:

1. User creates a personal account on the system
2. User watches preferred movie(s)
3. User goes to the section of the system dedicated to movie reviews
4. User selects their purchased movie and showtime to leave a review on
5. User inputs their review as a public message displayed on the system

Alternate Flows:

1. Amount of words on the review section exceeds the limit indicating an error message to shorten it, regulating within words
2. The user tries to review the same movie more than once, leading to multiple reviews; must restrict the user from performing such.
3. User invalid to review due to history of baseless and inappropriate comments carried.

3.4 Classes / Objects

3.4.1 <Class / Object #1> - User class

3.4.1.1 Attributes

The user has access to use the system on a mobile device. The user is able to complete their transaction mobily, without having to physically wait in line. The user is given options for different showtimes, allowing the user to plan their schedule to their discretion.

3.4.1.2 Functions

Given the user has a computer or mobile device connected to the internet, the user will be able to purchase a movie ticket.

3.4.2 <Class / Object #2> - Ticket Class

3.4.2.1 Attributes

The system will provide ticket options for different movies and showtimes. The system will recognize different pricing options depending on each ticket owner's age (child's ticket, adult ticket, or senior ticket). The system will provide listed age-ranges recognizing someone as either child, adult, or senior.

3.4.2.2 Functions

The system will prompt the user to choose a specific theater, movie, and time. The system will prompt the user to input the amount of child, adult, and/or senior tickets correlating to each ticket owner's age.

…

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 product shall be based online through a webpage and has to be run from a dedicated web server. The product shall take initial load time depending on internet connection strength which also depends on the media from which the product is run. Software performance will ultimately depend on consumer hardware components of the device that is running the webpage.

3.5.2 Reliability

The system shall provide a recognized agreement with an internet service provider who will supply internet availability.

3.5.3 Availability

Users will be able to access the system through any given web browser, given they have internet connection. Users will be able to choose their preferred theater, movie, time, and date given listed options.

3.5.4 Security

The software system will make use of Secure Sockets Layer (SSL) throughout all transactions, in order to protect any and all customer information.

3.5.5 Maintainability

The system shall maintain information regarding purchased movie tickets and their subsequent owner(s).

3.5.6 Portability

Users are able to access the system through a computer, laptop, or mobile device. Users will receive a QR code of their movie ticket which will be accessible from a computer, laptop, or mobile device.

3.6 Inverse Requirements

- The software designed should not perform errands outside its requirements mentioned by this document.
- Personal account details when entered by a user should not be auto-saved without the consent of the user. Like credit card details and forbidden data
- Overbooking the theater must be restricted. The last user that booked the ticket must be notified about the excessive booking than the actual availability.
- Must not delete a user's past activity if has an account and logged out of that session.
- Users can not transfer ticket rewards from one account to another. In case of account transfer, direct the user to customer support.

3.7 Design Constraints

- Authorization to an account's personnel data (Name, Contact No., Payment Information, Location, Email Add.) must be attainable by a user who owns that account, under role-based access control (RBAC).
- The Software when accessed by theaters must be compatible, for instances of on-site payment by user or ticket printing at the venue. Hence the system should update with the existing hardware API's of the theater system as well.
- Customers may be on a network connection that is slow and expensive, with bandwidth limitations and data caps. In such times the availability of payment online must be refrained, void of multiple-bookings.

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.

- Data Types in New Roman Times that include Movie name, description, location, and time(PST). For everything including personnel details, different tabs, T&C forms or notifications must be typed in MySQL (Usually with large storage value). Monetary value in US$ with the function of decimal displays "0.99".
- Storage capabilities include using tabs to form sections; home, movies, events, and profile. For Mobile application vertical view. Web application horizontal and vertical view accessible, hence storage of data to be spread according in vertical mode.
- Data retention of user's activity and personnel records must be secured and backed up weekly to restrict data loss. Inactivity of user while in a session for a significant duration must be ignored and secure the data activity of user.

- Data Integrity of users to be sustained through validation of email address; using such to store customer details and thus avoiding data leakage.

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 the SRS's requirements.

4.1 Sequence Diagrams

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.

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