Movie Theater Ticketing Service

Software Requirements Specification

1.3

14 September 2025

Group 7

Matthew Gallenburger, Nikola Milosavljevic, Ryan Schindler

Revision History

Date	Description	Author	Comments
9/13	1.0 Use Cases & FRs	NM	3.2-3.4, Section 3 MT and RTC
9/14	1.1 NFRs	RS	2.4, 2.5, 3.5 completed
9/18	1.2 User & Context	MG	1.1-2.3
9/19	1.3 Proofreading	RS	3.1, Proofreading

Document Approval

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

Signature	Printed Name	Title	Date
Ryan Schindler	Ryan Schindler	Software Eng.	9/19/2025
	Dr. Gus Hanna	Instructor, CS 250	

Table of Contents

1. Introduction

1.1 Purpose

The purpose of this Software Requirements Specification (SRS) is to plan the requirements for an Online Movie Ticketing and Screening System. This document is intended for use by system developers, project stakeholders, theater managers and instructors reviewing the project. It will articulate and provide enough detail for the software engineers to design, implement and test the system.

1.2 Scope

The Online Movie Ticketing and Screening System will allow potential customers to browse movies that the theater is showing, reserve select seats, and purchase tickets securely through an external payment provider. Customers that purchase a ticket through the system will be sent an e-ticket containing an individual barcode, which will be validated upon entry. Customers will also have the ability to cancel or reschedule their reservations that fall in line with the theater's refund policy: full refund for cancellations that are made at least 24 hours before the time of the showing, with no refunds after that time has expired.

The system will provide managers and authorized employees of the theater with tools to schedule screenings and manage theater availability. While at checkout, customers seats will be held for 8 minutes in order to prevent conflicts with other customers selecting those seats. This product will improve customer convenience, streamline ticket processes, and enhance efficiency for theater staff.

1.3 Definitions, Acronyms, and Abbreviations

1.3.1   E-Ticket: A digital ticket issued by the system that contains an individual barcode for entry.

1.3.2   Seat Hold: Temporary reservation of a customer selected seat for 8 minutes while the payment information is being processed.

1.3.3   Refund Policy: A rule the specifies refund eligibility (full refund if cancelled less than 24 hours before the showing of the movie, otherwise, no refund will be issued).

1.3.4   UC: Use Case

1.3.5   External Payment Provider: A third-party financial institution that processes transactions.

1.3.6   FR: Functional Requirement

1.4 References

1.4.1   CS250-SRS Template (provided by instructor)

1.4.2   Lecture notes on use cases (lecture-use-cases.pdf, provided by instructor)

1.4.3   Group assumptions defined during in-class client interviews

1.5 Overview

1.5.1   *Section 2: Provides a general description of the system which includes product description, product functions, user characteristics, general constraints, assumption, and dependencies.*

1.5.2   *Section 3: Provides specific requirements for the system such as external interface requirements, which also includes user interfaces, hardware interfaces, software interfaces, and communication interfaces. This section will also give a description of both functional and non-functional requirements and outline use cases.*

1.5.3   *Section 4: Provides analysis models such as sequence diagrams.*

1.5.4   *Section 5: Describes the change management process.*

2. General Description

2.1 Product Perspective

The Online Movie Ticketing and Screening System is a standalone web-based application integrated with a single external payment provider. It will replace the traditional process of an in-person and phone-based system required for purchasing movie tickets. The online system will streamline ticket sales, making the process of facilitating sales more accessible for both customers and employees. The system will integrate with the theater's infrastructure that will allow the ability to scan and verify customer's e-tickets.

2.2 Product Functions

The Online Movie Ticketing and Screening System will provide the following functions:

2.2.1   *Purchase Ticket: Customers can select movies, showtimes, and seats, then pay through a third-party payment provider.*

2.2.2   *Seat Hold: While the customer is checking out, selected seats will be held for 8 minutes while they enter in payment information.*

2.2.3   *Cancel/Reschedule Ticket: Customers will have the ability to cancel or reschedule a reservation which is subject to the theater's refund policy.*

2.2.4   *E-Ticket Generation: The system will produce a unique and scannable e-ticket.*

2.2.5   *Payment Processing: Securely handle transactions via a third-party payment provider.*

2.3 User Characteristics

2.3.1   *Customers: Moviegoers of any age, using a device that could include a smartphone, tablet or traditional desktop computer. The system must be user friendly and scale to a mobile device properly.*

2.3.2   *Theater Managers: Manager and authorized employees that are responsible for scheduling screenings. They require administrative access with simple but flexible management tools.*

2.3.3   *Theater Staff (Scanning Tickets): Employees validating tickets at the entrance. These people will need reliable scanning functionality.*

2.4 General Constraints

2.4.1   Seats holds while a customer is checking out will expire after 8 minutes if the purchase was not processed.
2.4.2   Refunds are only issued if the request happens no later than 24 hours before the screening takes place.
2.4.3   All payments must be processed through the designated external payment provider.
2.4.4   The system must generate a unique e-ticket issued to the customer to prevent duplication or any other form of fraud.

2.5 Assumptions and Dependencies

2.5.1   Customers have internet-enabled devices that are able to access the system.
2.5.2   The theater has the necessary hardware for scanning and processing e-tickets.
2.5.3   The external payment provider remains functional and available.
2.5.4   Theater staff and managers will be trained and able to use the system effectively.

Actor List:

Customer (Primary Actor):
- *Purchases, cancels, and reschedules tickets.*
- *Receives e-ticket for entry to the theater.*

Theater Manager (Primary Actor):
- *Schedules and manages movie screenings.*
- *Monitors ticket sales and availability.*

Theater Staff (Supporting Actor):
- *Scans customers' e-tickets prior to entry.*

External Payment Provider (Supporting Actor):
- *Facilitates secure processing for customer payments.*

3. Specific Requirements

For the movie ticketing website, we select three critical use cases (UCs) that inform the specific requirements that the service needs: purchasing a ticket, adjusting a sold ticket, and scheduling the movie screenings. From these three UCs, we then identify 15 functional requirements (FRs) that are necessary for these UCs to function. In turn, a recommended list of class names and properties are provided for design. A minimum traceability list and a requirement traceability matrix are listed below for quick reference, and a UC diagram is included as figure 1.

Limitations of both the project and the users must be kept in mind to ensure a quick, hassle-free experience for both the customers and the employees. The external interface requirements capture the requirements for the website design and layout, whereas the non-functional requirements give strict, testable parameters for user experience.

Minimal Traceability (per UC)
Quick list of related FRs per UC.
- **UC-1 Purchase Ticket** → FR-001, FR-002, FR-003, FR-004, FR-005, FR-006, FR-007, FR-008, FR-011, FR-012, FR-014

- **UC-2 Cancel/Reschedule** → FR-008, FR-009, FR-004, FR-005, FR-006, FR-013, FR-014
- **UC-3 Schedule Screening** → FR-010, FR-015

Requirement Traceability Matrix (RTM)

Use Case	Related FRs	Priority
UC-1 Purchase Ticket	FR-001, FR-002, FR-003, FR-004, FR-005, FR-006, FR-007, FR-008, FR-011, FR-012, FR-014	HIGH
UC-2 Cancel/Reschedule	FR-008, FR-009, FR-004, FR-005, FR-006, FR-013, FR-014	HIGH
UC-3 Schedule Screening	FR-010, FR-015	MEDIUM
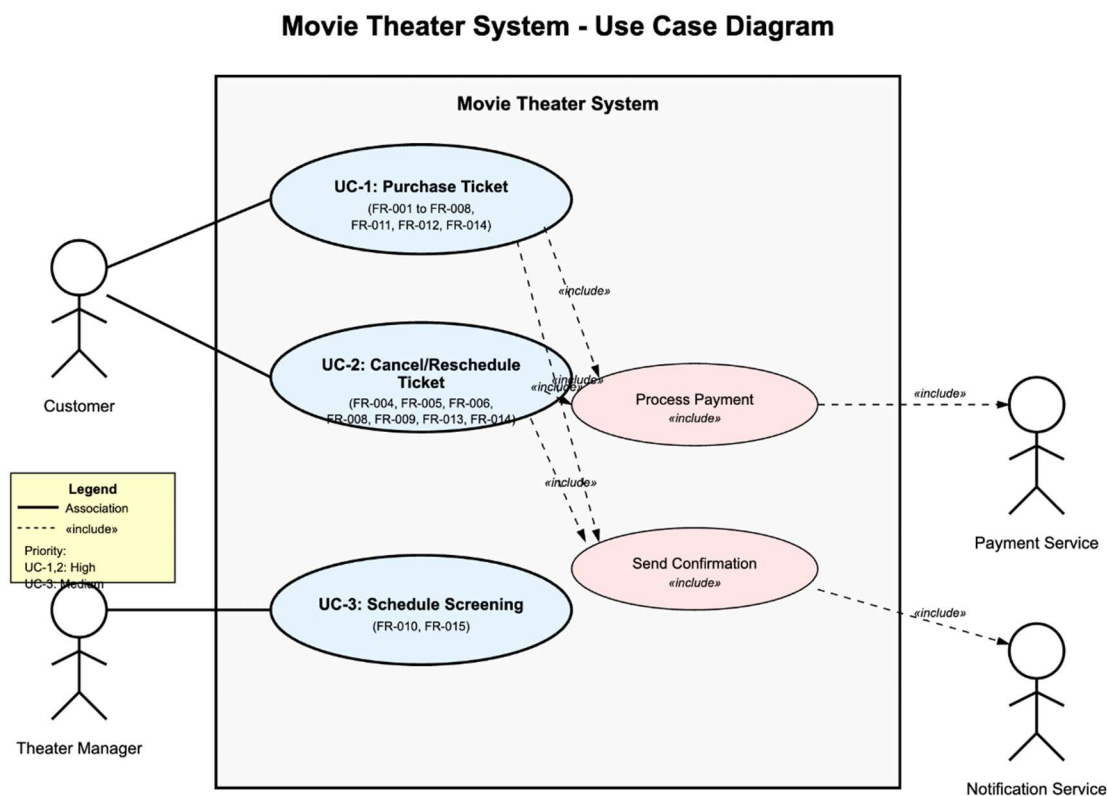
Figure 1. Movie Theater Ticketing System – Use Case Diagram

3.1 External Interface Requirements

3.1.1 User Interfaces

There will be one joint user interface for both the customer and employee for the website. The website will be designed in an easy-to-use and understand website format that minimizes the number of menus needed to navigate to two at most. The employee version will be secure; the employee will have to enter their credentials into the system to log in, and they will be given screening editing privileges depending on whether they are a manager or not.

3.1.2 Hardware Interfaces

The website will primarily be used on computer and laptop hardware interfaces; however, the website can be accessed on mobile devices as well. The software will utilize a responsive web browser, freeing the requirement to be designed for one type of screen. However, the interface will be tailored to a 1920 x 1080 pixel resolution, which is commonly used in most computer screens.

3.1.3 Software Interfaces

The software interface will be designed primarily for Chrome browsers and the three most popular browsers (3.5.6). All code must be designed and optimized for these browsers.

The system will also interface with the SQL databases housing information on customer accounts and movie screenings (FR 3.2.1-4,7-9). These will be rented through the cloud provider Azure. There must also be a logging interface to an audit log so every transaction and adjustment is recorded on the system (FR 3.2.14).

3.1.4 Communications Interfaces

The communication interface will send emails to inform and remind the customers when tickets are created, rescheduled, or cancelled (FR 3.2.7,8). The software will work together with a built-in API for automated emailing.

3.2 Functional Requirements

Catalog of 15 testable FRs. Each ties back to at least one UC.

3.2.1 Screening Listings

The system shall list available screenings by date, movie, auditorium, and format for customer browsing.

3.2.2 Seat Map

The system shall display a **real-time seat map** for a selected screening, indicating Available/Held/Sold states.

3.2.3 Selection

The system shall allow customers to select seat(s) and ticket type(s) for a screening and shall enforce seating rules (e.g., accessibility blocks, avoid orphan single seats).

3.2.4 Seat Holding

The system shall **hold** selected seats for **≥ 8 minutes** once checkout begins and release those seats on payment failure, cancellation, or hold expiry.

3.2.5 Summary

The system shall calculate and present an order summary including ticket prices, fees, and taxes before payment.

3.2.6 Payment Processing

The system shall process payments via the external Payment Service and update order status according to the authorization result.

3.2.7 Ticket Creation

The system shall issue an **electronic ticket** with a unique order ID and scannable code upon successful purchase and shall send confirmations for purchase, cancellation, and reschedule.

3.2.8 Order Retrieval

The system shall allow customers to **retrieve orders** using email and order ID.

3.2.9 Cancellation and Reschedule

The system shall allow customers to **cancel or reschedule** within policy constraints and shall apply the correct **refund/price-difference** outcome.

3.2.10 Manager Privileges

The system shall allow authorized managers to **create/edit/publish/unpublish** screenings and shall detect **scheduling conflicts** for the same auditorium.

3.2.11 Ticket Validation and Reuse

The system shall **validate tickets at entry** and prevent **re-use** of the same ticket code.

3.2.12 Sales Moderation

The system shall prevent **overselling** by applying atomic updates to seat and order states.

3.2.13 Cancellation and Reschedule Confirmation

The system shall display **cancellation/reschedule eligibility** and the calculated refund/credit before the user confirms the action.

3.2.14 Audit Log

The system shall record an **audit log** for order state changes and ticket validations, including timestamp and actor.

3.2.15 Policy Configuration

The system shall allow authorized managers to **configure policy values** used by the system (e.g., seat-hold duration, refund window).

3.3 Use Cases

3.3.1 Purchase Ticket

3.3.1.1 Primary Actor: Customer
3.3.1.2 Supporting Actors: Payment Service (authorization), Notification Service (confirmation delivery)
3.3.1.3 Scope: Movie Theater Ticketing System
3.3.1.4 Level: User goal

3.3.1.5 Preconditions
1. At least one screening is published and has capacity.
2. System clock and payment integration are operational.

3.3.1.6 Trigger
Customer decides to buy a ticket for a specific screening.

3.3.1.7 Main Success Scenario (Basic Flow)
1. Customer selects a date/movie to view available screenings.
2. System displays showtimes and high-level info (time, auditorium, format, price from).
3. Customer selects a screening; system shows the real-time seat map.
4. Customer selects seat(s) and ticket type(s).
5. System places selected seats on **hold** and displays order summary with prices/fees/taxes.
6. Customer proceeds to payment; system submits the charge to Payment Service.
7. Payment Service authorizes payment; system marks seats **sold**, generates **e-ticket(s)**, and records the order.
8. System sends confirmation with ticket(s) to the customer.
9. Customer can retrieve the order later using email + order ID.

3.3.1.8 Alternate / Exception Flows
A1. **Seat taken mid-process:** If a chosen seat becomes unavailable before hold is applied, system informs the customer and prompts reselection; no seats are held.
A2. **Hold expiry before payment:** If the 8-minute window elapses before payment, system releases held seats and returns to seat selection.
A3. **Payment failure/decline:** On failure, system releases holds and shows an error with option to retry.
A4. **Payment timeout/unknown result:** System shows pending status; if not confirmed within the configured timeout, treat as failure and release seats.
A5. **Policy/seat-rule violation:** If a selection violates auditorium rules (e.g., leaves orphan single seats, accessibility blocks), system rejects the selection with guidance.

3.3.1.9 Postconditions
- **Success:** Order status = **CONFIRMED**; seats marked **SOLD**; e-ticket issued; confirmation sent; audit log written.
- **Failure:** No charge captured; any holds released; inventory unchanged.

3.3.2 Cancel or Reschedule Ticket

3.3.2.1 Primary Actor: Customer
3.3.2.2 Supporting Actors: Payment Service (refund/charge diff), Notification Service (confirmation)
3.3.2.3 Scope: Movie Theater Ticketing System
3.3.2.4 Level: User goal

3.3.2.5 Preconditions
1. A confirmed order exists.
2. The request is within policy windows defined by the theater.

3.3.2.6 Trigger
Customer chooses to cancel or reschedule an existing order.

3.3.2.7 Main Success Scenario (Cancel)
1. Customer retrieves the order by email + order ID.
2. System shows cancellation eligibility and refund amount per policy.
3. Customer confirms cancellation.
4. System cancels the order, releases seats, and initiates refund via Payment Service.
5. System sends cancellation & refund confirmation; audit log recorded.

3.3.2.8 Alternate Success Scenario (Reschedule)
RS1. Customer opts to **reschedule** instead of cancel.
RS2. System lists eligible alternative screenings (same movie or theater policy).
RS3. Customer selects new screening and seats; system re-applies hold and recalculates totals.
RS4. If the new price is higher, system collects the difference; if lower, system issues a credit/refund per policy.
RS5. System finalizes the new order, releases prior seats, and sends updated confirmation; audit log recorded.

3.3.2.9 Exception Flows
E1. **Outside policy:** System declines the cancellation/reschedule and explains the policy window.
E2. **No seats available for reschedule:** System allows the customer to keep the current booking or cancel (if eligible).
E3. **Payment failure on price increase:** System keeps the original booking intact and provides retry.

3.3.2.10 Postconditions
- **Success (Cancel):** Order status = **CANCELLED**; seats returned to inventory; refund issued per policy; audit log written.
- **Success (Reschedule):** Original seats released; new order **CONFIRMED**; refund/charge handled; audit log written.
- **Failure:** Original order remains unchanged.

3.3.3 Schedule Screening (Manager Function)
3.3.3.1 Primary Actor: Theater Manager
3.3.3.2 Supporting Actors: —
3.3.3.3 Scope: Movie Theater Ticketing System
3.3.3.4 Level: User goal

3.3.3.5 Preconditions
1. Movie and auditorium configurations exist, including seat map and capacity.
2. Manager is authenticated and authorized.

3.3.3.6 Trigger
Manager decides to create or edit a screening.

3.3.3.7 Main Success Scenario
1. Manager creates a screening with: movie, auditorium, date/time, capacity/seat map, base price,

attributes (e.g., 3D/IMAX).
2. System validates the screening (time window, capacity) and checks for **auditorium conflicts**.
3. Manager sets the screening status to **Published** to make it visible for purchase.

3.3.3.8 Alternate / Exception Flows
A1. **Conflict detected:** System prevents saving and explains the conflict (overlap in the same auditorium).
A2. **Invalid capacity/seat map:** System rejects the change until fixed.
A3. **Unpublish:** Manager may unpublish a screening; it no longer appears in customer listings but remains editable.

3.3.3.9 Postconditions
- **Success:** Screening exists with validated schedule; visible to customers if **Published**.
- **Failure:** No changes saved.

3.4 Classes / Objects

3.4.1 Screening

3.4.1.1 Attributes:
- Publication State: Draft → Published → Unpublished/Cancelled

3.4.1.2 Functions:
- Update Publication

<UC 3.3.1 and FR 3.2.1,2,10>

3.4.2 Auditorium

3.4.2.1 Attributes:
- Seat Map
- Capacity
- Accessibility Constraints

3.4.2.2 Functions:
- Create/Edit Auditorium
- Update Seat

<UC 3.3.1 and FR 3.2.1,2>

3.4.3 Seat

3.4.3.1 Attributes:
- Availability: Available → Held → Sold

3.4.3.2 Functions:
- Change Availability
- Check for Orphan Seat

<UC 3.3.1 and FR 3.2.2,3,4>

3.4.4 Order

3.4.4.1 Attributes:
- Status: Pending → Confirmed → Cancelled/Rescheduled

- Order Number

3.4.4.2 Functions:
- Set Order State
- Link Order

<UC 3.3.1,2 and FR 3.2.8>

3.4.5 Ticket

3.4.5.1 Attributes:
- In-use: Issued → Checked-in

3.4.5.2 Functions:
- Issue Ticket
- Check-in Ticket

<UC 3.3.1 and FR 3.2.7,11>

3.4.6 Payment

3.4.6.1 Attributes:
- Status: Authorized → Captured/Refunded/Failed

3.4.6.2 Functions:
- Check Payment Information
- Validate with Payment Service

<UC 3.3.1,2 and FR 3.2.6,9>

3.4.7 User Roles

3.4.7.1 Attributes:
- Role: Customer, Manager, Payment/Notification Service

3.4.7.2 Functions:
- Set Role Privileges
- Change Role

<UC 3.3.3 and FR 3.2.10>

3.4.8 Policy Values

3.4.8.1 Attributes:
- Seat-Hold Duration
- Refund Window
- Orphan Seat Requirements

3.4.8.2 Functions:
- Configure Policy Values

<UC 3.3.3 and FR 3.2.15>

3.5 Non-Functional Requirements

The software must meet these 6 non-functional requirements: Performance, Reliability, Availability, Security, Maintainability, and Portability. This will ensure quick, hassle-free, and helpful experiences for both the customer and the management.

3.5.1   Performance

The website must load quickly and reliably so that users are not frustrated accessing the site.
3.5.1.1 Quick to load: 95% of loads complete within 5 seconds.
*3.5.1.2 Quick to transac*t: 95% of transactions complete within 8 seconds.
3.5.1.3 Quick to update: 99% of user interactions with the webpage should register within 3
    seconds.
3.5.1.4 Bandwidth: server can handle 1000 simultaneous requests at once.

3.5.2 Reliability

The system must handle any errors that will cripple the service.
3.5.2.1 Error Handling: The system will be able to identify, and in certain cases correct, errors.

3.5.3 Availability

The service will available as often as possible, allowing for occasional maintenance periods.
3.5.3.1 Consistent Up-Time: Server and website is available for public use 99.9% of uptime.

3.5.4 Security

The service will protect customers' data and prevent unauthorized users from accessing and
modifying sensitive data.
3.5.4.1 Authorization: only managers or higher-level clearances shall be able to access secure
    information from databases.
3.5.4.2 Encryption: customer data should be encrypted with current RSA encryption.

3.5.5 Maintainability

The software will be designed to last and work without frequent and drastic revision upon
release. There will be small patches to cover unforeseen problems with the system on a continual
basis.
3.5.5.1 Lifespan: The software should be able to function over the next 3 years with minor
    revisions.
3.5.5.2 Frequency of Revisions: Minor updates will occur every 3 months, while major changes
    will occur every 3 years.

3.5.6   Portability

The website should run without error or problem in the most frequently used browser engines.
3.5.6.1 *Chrome*: Designed for the Chrome browser
3.5.6.2 *Common Compatibility*: Runs without errors in the 3 most common browsers: Edge,
    Safari, and Firefox

3.6 Inverse Requirements

State any *useful* inverse requirements.

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 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