

Report One: Fort Hays State Movie Theaters

CSCI 441

Professor: Hieu Vu

**Members: Jakob Schaefer, Derek Litke,
Andrew Carter, Hanyong Yoon, David
Sowles, Ryan Smith**

Member Contribution

Jakob Schaefer – Formatting, Fully Dressed Use case descriptions, goals, requirements, and analysis, table of contents

Hanyong Yoon – Customer Problem Statement Narratives, Glossary of terms, Use Cases, Stakeholders, Actor and Goals, and Traceability Matrix.

Derek Litke – System Architecture, Plans of Work

Andrew Carter – Use Case Diagrams, System Sequence Diagrams, Actor Diagrams

David Sowles – Requirements, Use Cases, Package Diagram

Ryan Smith - Project Size

Responsibility Matrix

Sub-Project / Task	Jakob	Hanyong	Derek	Andrew	David	Ryan
Cover Page & Contributions	20%	20%	20%	20%	20%	0%
Work Assignment	16%	16%	16%	16%	16%	16%
Customer Problem Statement	15%	40%	15%	10%	10%	10%
Decomposition into sub-problems	15%	40%	15%	10%	10%	10%
Glossary of Terms	15%	40%	15%	10%	10%	10%
Goals, Requirements, and Analysis	40%	15%	15%	15%	10%	5%
Business Goals Hierarchy + Diagram	15%	15%	15%	40%	10%	5%
Functional Requirements	40%	15%	15%	10%	10%	10%
Non-functional Requirements	40%	15%	15%	10%	10%	10%
User Interface Requirements	40%	15%	15%	10%	10%	10%
Use Cases & Actors	15%	40%	15%	10%	10%	10%
Casual Descriptions	15%	40%	15%	10%	10%	10%
Use Case Diagram	15%	15%	15%	40%	10%	5%
Traceability Matrix	40%	15%	15%	10%	10%	10%
Fully Dressed Use Cases	40%	15%	15%	10%	10%	10%
System Sequence Diagrams	15%	15%	15%	40%	10%	5%
User Interface Specification	40%	15%	15%	10%	10%	10%

System Architecture	10%	15%	40%	15%	10%	10%
Subsystems	10%	15%	40%	15%	10%	10%
Architecture Style and Control Flow	10%	15%	40%	15%	10%	10%
Hardware & Resource Requirements	10%	15%	40%	15%	10%	10%
Project Management (Gantt, Roadmap)	10%	15%	40%	15%	10%	10%
References	16%	16%	16%	16%	16%	16%
Final Editing & Integration	16%	16%	16%	16%	16%	16%

Table of Contents

1. Customer Problem Statement
 - a. Customer View
 - b. Staff/Admin View
 - c. Theater Manager
2. Glossary of terms
 - a. Technical Terms
 - b. Non-technical Terms
3. Goals, Requirements, and Analysis
4. Functional Requirements
5. Nonfunctional Requirements
6. UI Images
 - a. Customer Home
 - b. Seat Map Selection
 - c. Admin Dashboard
 - d. Movie Details
 - e. Checkout Page
7. Stakeholders
 - a. Customers
 - b. Theater Employees
 - c. Theater Manager
 - d. Theater Owner
8. Actors and Goals
 - a. Initiating Actors
 - b. Participating Actors
9. Use cases overview

- a. Customer Use cases
- b. Theater Employee Use cases
- c. Use case diagrams

10. Traceability Matrix

11. Fully Dressed Use case descriptions

12. System Sequence Diagrams

13. User Interface Specifications and Effort Estimations

14. System Architecture

- a. Identifying Subsystems
- b. Architecture styles
- c. Mapping subsystems to Hardware
- d. Connectors and network protocols
- e. Global control flow
- f. Hardware Requirements

15. Project Size Estimation

16. Plan of Work

17. References

Customer Problem Statement

Customer (Actor)

As a person that goes to the movies often, buying a ticket has always been a problem for me. Most of the time when the movie is a popular release, I often find myself waiting in a long line to purchase a ticket. Sometimes, I would go through the trouble of waiting just for the ticket to be sold out. There are many problems besides the long line and the unavailable ticket. For example, even when the ticket is available, I don't know what seat is available until it's my turn to purchase the ticket. This is problematic, especially more so if I am with my family or friends, that's because we want to sit together and the only seats remaining are scattered or not enough seats are available for my group.

Another issue I face with the process of buying a movie ticket is not receiving an efficient confirmation, when I purchase a movie ticket it's usually just a small stub that can be easily lost. While some theaters send an email confirmation in case you lose the ticket, old theaters don't have a system like this. If I were to lose my ticket, there is no way to prove my purchase history, and I will have to rebuy if I still want to watch the movie, all while dealing with this frustrating situation. Beyond this issue, I like to keep track of what I've watched in the past, part of the theater experience is remembering the movie. It's almost impossible to remember every movie I've watched in the past, and without a proper system that shows the history of movies I've watched, I would have to rely on going through the collections of stubs that I've collected over the years. Even then, over time, stubs get worn out and it's hard to read the prints or sometimes impossible to read them. Sometimes stubs can get lost, and I'm faced with the problem of trying to remember the movie that I've watched.

In a busy theater, I cannot stress the importance of convenience. Many businesses are modernizing their services, whether it's for ordering food, buying tickets, or shopping. With services evolving with time, it's important for movie theaters to adapt to this evolution. Instead of making customers feel like they're going through an unnecessary obstacle, the

process of buying a ticket should be simple. Sometimes, I feel like I spend more time dealing with the tedious process of buying tickets than I do enjoying the movie.

I think about accessibility too. While I am blessed enough to be in a healthy body, that is not the case for everyone. Everyone can't stand in a long line for a long time. Whether it's the elderly, parents with young children, or people with disabilities, the process of purchasing a ticket can be overwhelming. While kiosks might be a good option to solve this problem, they're often outdated and confusing. A more user-friendly system must be employed for everyone to enjoy the "movie theater" experience.

With all these issues combined, it makes going to the movies less appealing. Why shouldn't I just stay home and watch something at the comfort of my home using various streaming platforms. Instead, every time I go to the movies, I feel like I am fighting through tedious procedures of outdated system that should be simple and convenient. What I want is a simple and reliable solution. I want a system where I can log in, browse what movies are available, and see the available seating charts. I want to pick exactly where I want to sit, purchase tickets without having to wait in line, and get an email confirmation of my purchase right away. I want access to the history of my purchases to view what I've seen in the past to remember my experiences. Generally, what I want is for the ticket-buying process to be stress-free. I want to enjoy going to the movies and not face the frustration of purchasing a ticket.

Staff/Admin (Theater Employee) (Actor)

Working at a movie theater can be rewarding, that's if everything is going according to plan. One problem I face is keeping the theater information accurate; Some examples are having the correct movies posted, accurate movie times, proper ticket prices, list of concession items. I must update all the information related to the movie theater manually. This takes up a lot of time and can sometimes be the reason for mistakes. If I forget to post the correct information, customers will get confused and possibly angry.

Another issue I face is with real-time updates. There are times when movie tickets sell out fast and the showing for it needs to be updated or I must cancel a showing because technical issues like projector not working properly. Making these changes isn't simple, there is no central system I can use to update everything. I would have to adjust everything manually and often leave notes for coworkers, which can lead to miscommunication. Without real-time updates it's difficult to answer the customer's questions like "How many tickets are left?" or "What's the most popular movie currently?" Finding the correct answers for these questions means going through multiple spreadsheets that are manually updated, which can be time consuming and inaccurate.

What I need is a reliable system where I can update everything in one place. I need a system where I can log in with my employee account, update movies and showtimes and concession items, change ticket pricing without worrying about if the information is correct or not. I need the system to have real-time update so I can answer customer questions without hesitation. I need a system that will improve the quality of life when it comes to my job. The system will help me make less mistakes and with it I can provide better service for the customer.

Theater manager (Actor)

As a manager my job is to make sure the entire theater operation is running smoothly. The most difficult part of my job is keeping track of employees. I need to keep track of who's working, what time they've arrived, and how many hours they worked to process their payroll. Currently, this is a manual and time-consuming process. Using time sheets, paper schedules, and outdated clock-in system is not ideal in this day and age. Keeping accurate reports is important for me as a manager. I must keep track of sales numbers, ticket performance, and customer activity. However, without a centralized system keeping track of these reports is a tedious task that can sometimes take hours.

To make my jobs easier, I need a modern centralized system where employees can clock in and out, track their working hours, and calculate their payroll. It would save me time and

headache if I could rely on the system to handle the accuracy of these records automatically. This system would help me tremendously in a way where I wouldn't have to spend hours checking employee timesheets or fixing mistakes that are caused by manual entry. Overall, I need to be able to save time, reduce errors, and make the theater a place where both the customers and staff can have positive experience.

Glossary of Terms

Technical Terms

Digital ticketing – Electronic process of purchasing a ticket through the app. Tickets are delivered digitally

Showtime - A scheduled movie screening with a fixed start time, shows specific auditorium it's linked too.

Purchase History - Record showing all previously purchased items from the movie theater.

Admin panel - Digital interface where employees and managers can update movie schedules, ticket pricing, seat availability, and concession.

User Account - User profile containing login credentials.

Customer Dashboard – Personalized area in the app that shows past purchases, payment preferences.

Non-Technical Terms

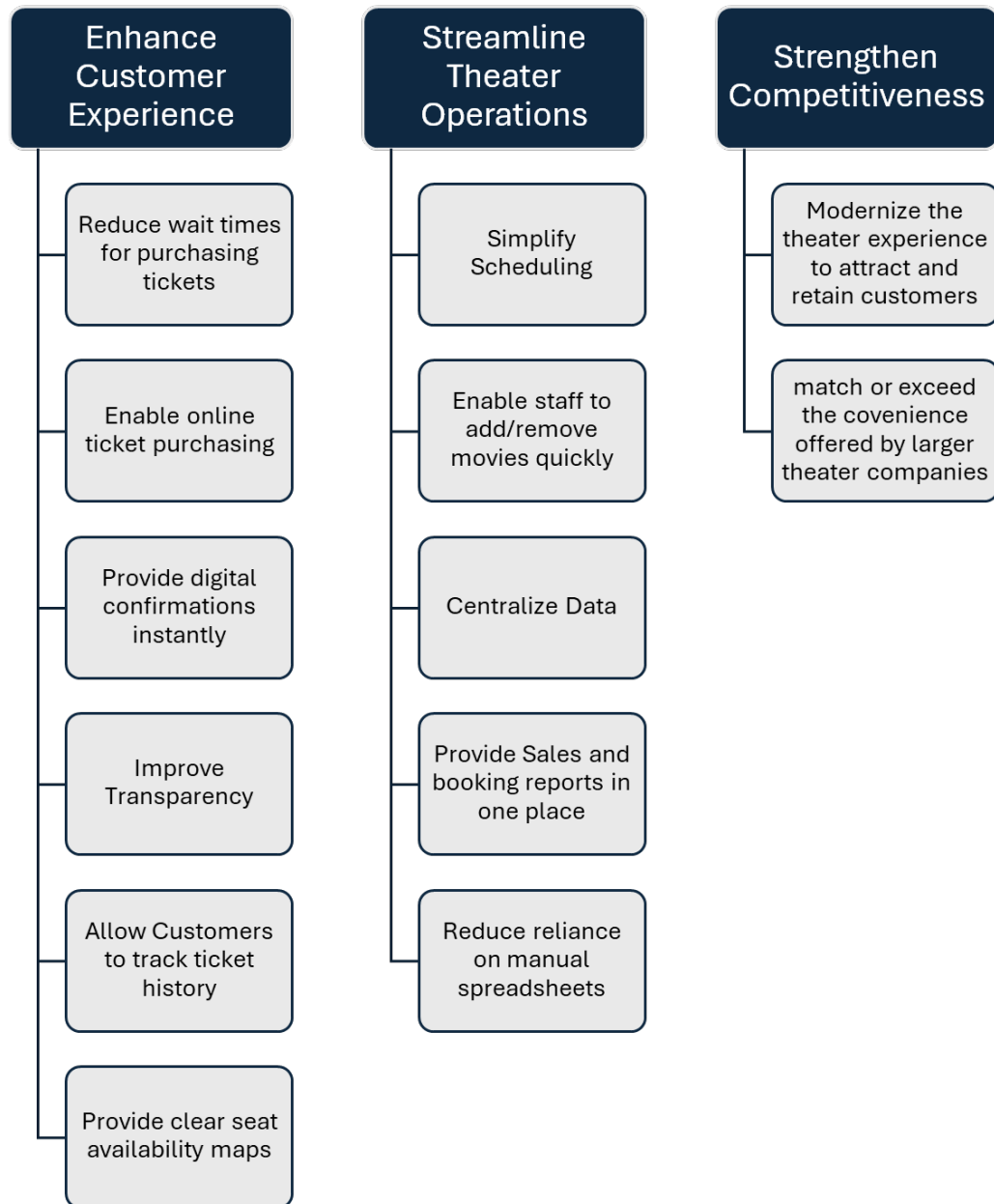
Auditorium - The room where the movie is played with projector, screen, and seats.

Seating Charts – A map of available and sold seats for specific showtime. Customers use this to pick out their seats.

Concessions – Snacks and drinks sold at the theater can be purchased using the app.

Accessibility Features – Design consideration that makes the app and theater experience usable for everyone.

Goals Requirements and Analysis



Functional Requirements

Req ID	Priority	Requirement
REQ-1	High	The system shall allow users to create an account with their email
REQ-2	High	The system shall display movies and showtimes
REQ-3	High	The system shall allow customers to search for movies
REQ-4	High	The system shall provide seat maps for each show time
REQ-5	High	The system shall allow customers to purchase tickets and select seats
REQ-6	Medium	The system shall allow customers to cancel purchased tickets
REQ-7	High	The system shall send ticket confirmations via email
REQ-8	Medium	The system shall allow customers to view past ticket history
REQ-9	High	The system shall allow admins to add/remove movies
REQ-10	High	The system shall allow admins to update showtimes
REQ-11	High	The system shall allow admins to adjust ticket pricing
REQ-12	Medium	The system shall provide admins with sales and booking reports

Nonfunctional Requirements (FURPS+)

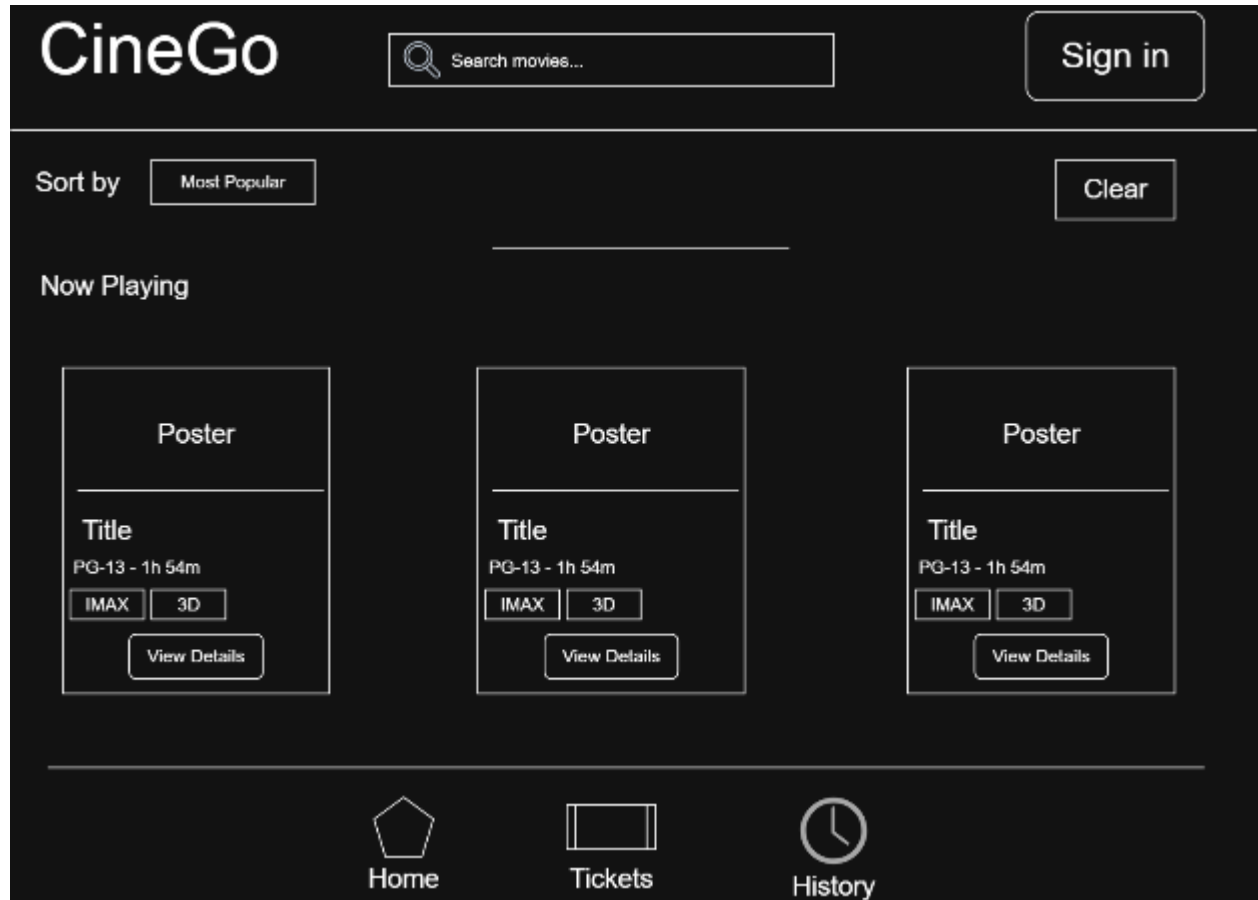
Req ID	Priority	Requirement
REQ-13	High	The system shall load the homepage in under 2 seconds
REQ-14	High	The system shall support mobile and desktop devices
REQ-15	High	The system shall remain available at least 99% of the time
REQ-16	Medium	The system shall support accessibility features
REQ-17	High	The system shall encrypt all sensitive data, including payment information
REQ-18	Medium	The system shall scale to support an increasing number of users and theaters

User Interface Requirements

Req ID	Priority	Requirement
REQ-19	High	The system shall present customers with a clear movie listings page, showing title, showtime, rating, and availability.
REQ-20	High	The system shall provide an interactive seating chart for customers to select seats
REQ-21	Medium	The system shall provide admins with a simple dashboard to manage movies and showtimes
REQ-22	Medium	The system shall provide managers with a dashboard summarizing sales, staffing, and payroll

UI Images

Customer Home



Seat Map Selection

CineGo

Menu

Select Your Seat

Sign in

Seats

Available

Reserved

Selected

Wheelchair Accessible

Companion

A

B

C

K

Movie Title / Date /
Time / Auditorium

Seats

Ticket Type

Price

Taxes:

Fees:

Subtotal:

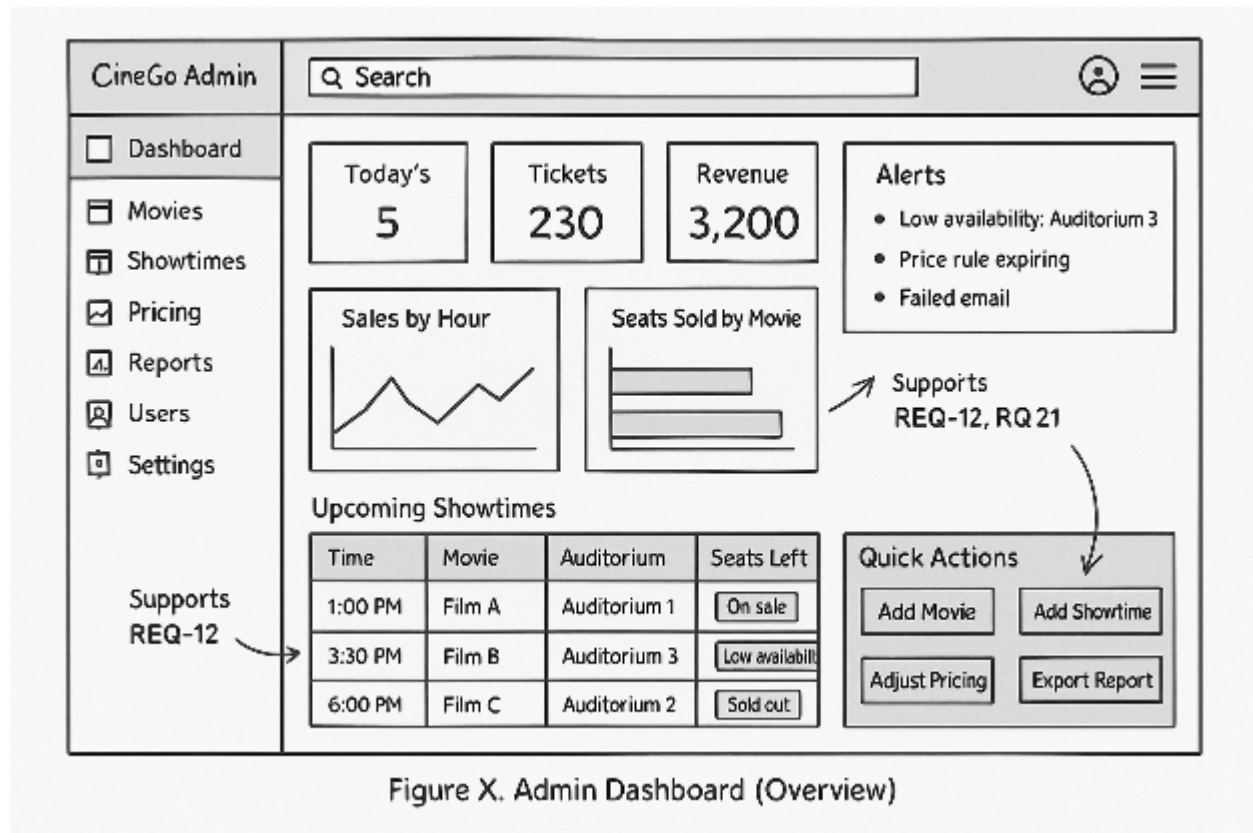
Total:

Continue

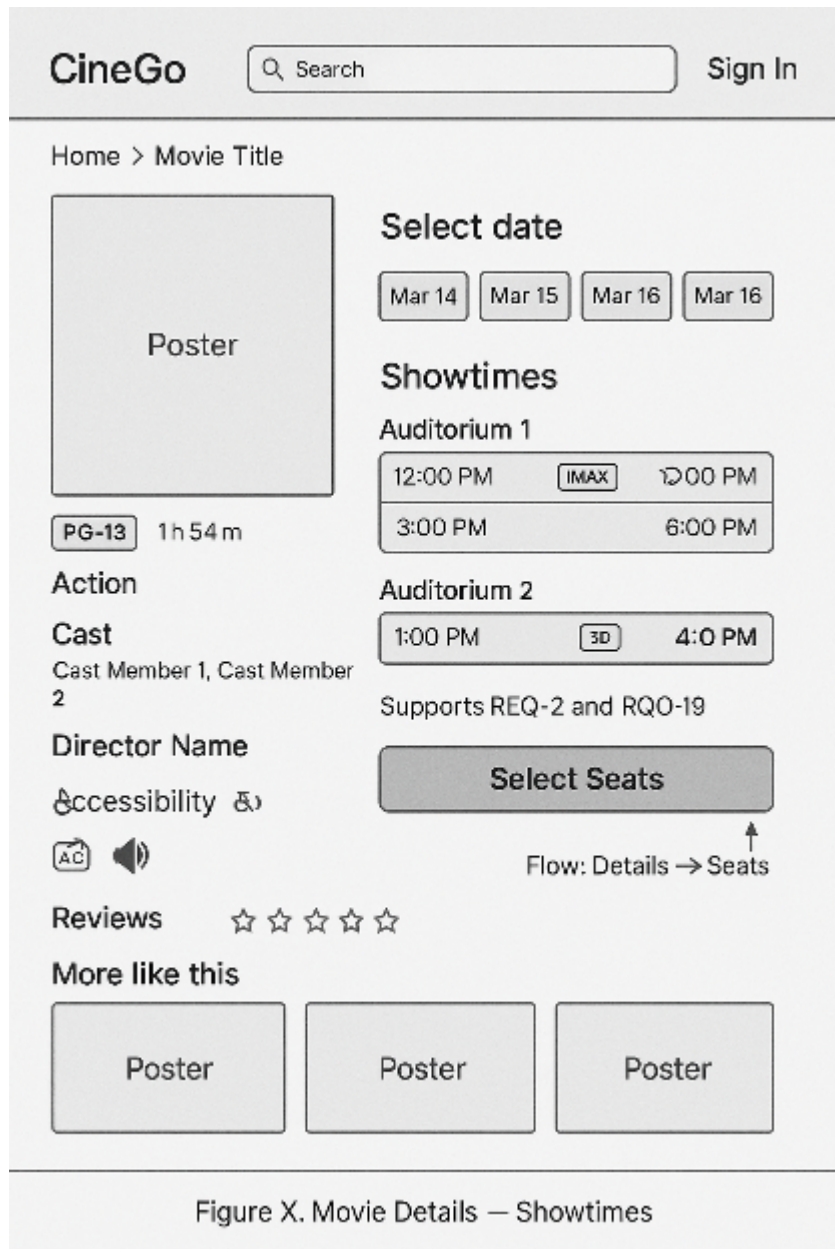
Best Available

Accessibility filters

Admin Dashboard



Movie Details



Checkout Page

Cart > Seats > Checkout > Confirmation

Payment Method

Card number ▾

Card number is required

Expiration date

CVV

CV

Saved Card

-4567

Contact Information

Email

Promo co de

Apply

☐ I agree to the terms and conditions

Pay \$29.75

Secured by SSL

Order Summary

Movie Title

April 8, 2024 5:00 PM

D3, D4

Subtotal

\$25.00

Fees

\$2.50

Tax

\$2.25

Total

\$29.75

Figure 5. Checkout page - Supports REQ-5, REQ-7, REQ-7, REQ-8

Stakeholders

The following are the stakeholders that will benefit the most from our movie ticketing system.

- Customers – Moviegoers, purchases tickets, choose seats, manage purchase history, and receive confirmations.
- Theater Employees – Responsible for updating movies, showtimes, prices, and aids with customer's needs.
- Theater Manager – Overlooks the entire theater operations, sales, staffing, and payrolls.
- Theater Owner – Focuses on the whole theater's profitability, compliances, and customer satisfaction.

Actors and Goals

Initiating Actors

Customers – Find showtimes, see available seats, choose seats, buy ticket, get ticket confirmation email, view/cancel purchases.

Theater Employees – Add and remove listings, showtimes, and pricing in real-time. Help with customers' needs.

Theater Manager - Access manager dashboard and audit reports.

Participating Actors

Payment Gateway - Authorize payments securely.

Email Service - Send ticket purchase confirmation and cancellation email.

Use Cases

Customer

UC-C1: Create Account

Customers sign up using email and password.

REQ-1

UC-C2: Browse and Search Movies

Customers can view movie listings and search for titles.

REQ-2, REQ-3, REQ-19

UC-C3: View Showtimes and Seat Map

Customers can view shows for movies and view seats using seating charts that have availability indicators and accessibility seating.

REQ-4, REQ-20

UC-C4: Purchase Tickets

Customer selects seats, enters payment, confirms order. System processes payment and issues order confirmation.

REQ-5, REQ-7

UC-C5: Cancel Ticket

Customer cancels an order. Purchase gets refunded and the seat used becomes available again.

REQ-6

UC-C6: View Purchase History and Customer Dashboard

Customers can view previous purchases made from the theater.

REQ-8

Theater Employees

UC-E1: Manage Movies

Employees can add new movies and remove outdated ones. Change the needed metadata and assets.

REQ -9

UC-E2: Manage Showtimes

Employees create and edit showtimes and can update changes in real-time.

REQ-10

UC-E3: Adjust Ticket Pricing

Employees can update prices for the tickets.

REQ-11

UC-E4: View Sales and Booking reports

Employees can view reports for sales, booking by its movies, date range, and auditorium.

REQ-12

UC-E5: Use Admin Dashboard

Employees can access a simplified dashboard to manage movies and showtimes from one place.

REQ-21

Theater Manager

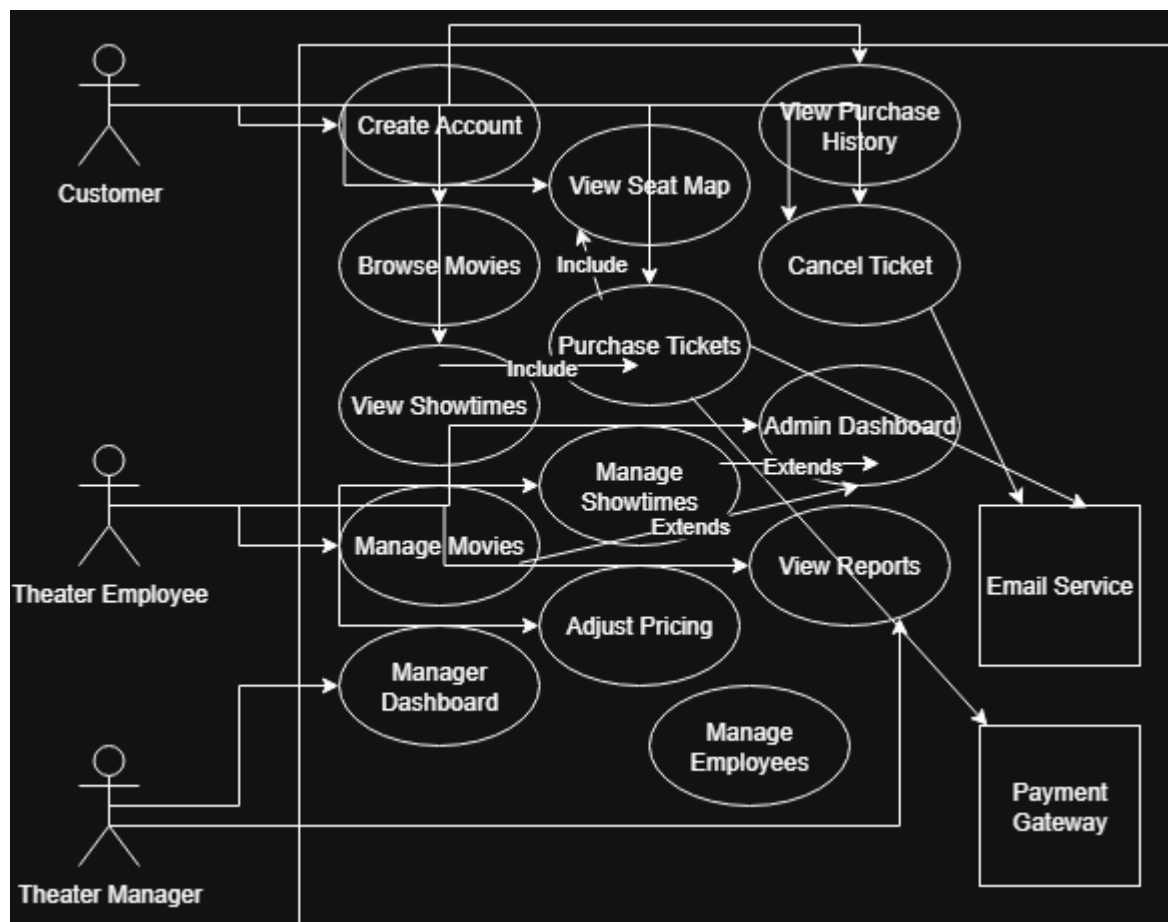
UC-M1: View Manager Dashboard

Managers have access to the Manager dashboard that shows summaries for sales, staffing/payroll information.

REQ-22, REQ-12

Requirements responding to all use cases: REQ-13, REQ-14, REQ-15, REQ-16, REQ-17, REQ-18.

Case Diagram



Traceability Matrix

	UC-C1	UC-C2	UC-C3	UC-C4	UC-C5	UC-C6	UC-E1	UC-E2	UC-E3	UC-E4	UC-E5	UC-M1
REQ1	X											
REQ2		X										
REQ3		X										
REQ4			X									
REQ5				X								
REQ6					X							
REQ7				X								
REQ8						X						
REQ9							X					
REQ10								X				
REQ11									X			
REQ12										X		X
REQ13	X	X	X	X	X	X	X	X	X	X	X	X
REQ14	X	X	X	X	X	X	X	X	X	X	X	X
REQ15	X	X	X	X	X	X	X	X	X	X	X	X
REQ16	X	X	X	X	X	X	X	X	X	X	X	X
REQ17	X	X	X	X	X	X	X	X	X	X	X	X
REQ18	X	X	X	X	X	X	X	X	X	X	X	X
REQ19		X										
REQ20			X									
REQ21											X	
REQ22												X

Fully Dressed Descriptions

UC-C4 Purchase Tickets

Primary Actor – Customer

Participating Actors – Payment Gateway, Email Service

Precondition – Customer is logged in, selects a movie, and a showtime

Postcondition – Ticket purchased, confirmation email sent, seat marked as unavailable

Flow of Events

1. Customer selects movie and showtime
2. System displays seating chart with availability
3. Customer chooses seat(s)
4. Customer enters payment details
5. System sends payment to payment gateway
6. Payment gateway returns approval
7. System issues ticket, reserves seat, and records purchase
8. System triggers email service to send confirmation email

Alternate Flows

1. If payment fails, customer is shown error and asked to retry
2. If seats become unavailable mid-process, system prompts customer to reselect

UC-C5 - Cancel Ticket

Primary Actor – Customer

Participating Actors – Payment Gateway, email service

Precondition – Customer has an active ticket

Postcondition – Ticket Cancelled, seat available, refund processed, cancellation email sent

Flow of events

1. Customer navigates to Purchase History
2. Customer selects a ticket to cancel
3. System verifies eligibility (before showtime, refundable policy)
4. System contacts payment gateway to process refund
5. System updates ticket status and makes seat available
6. System triggers email service to send cancellation confirmation

Alternate flows

1. If non-refundable, system displays an error
2. If refund fails, system alerts customer to contact support

UC-E1 – Manage Movies

Primary Actor – Theater Employee

Precondition – Employee logged into Admin Dashboard

Postcondition – Movie listing created, updated, or deleted

Flow of Events

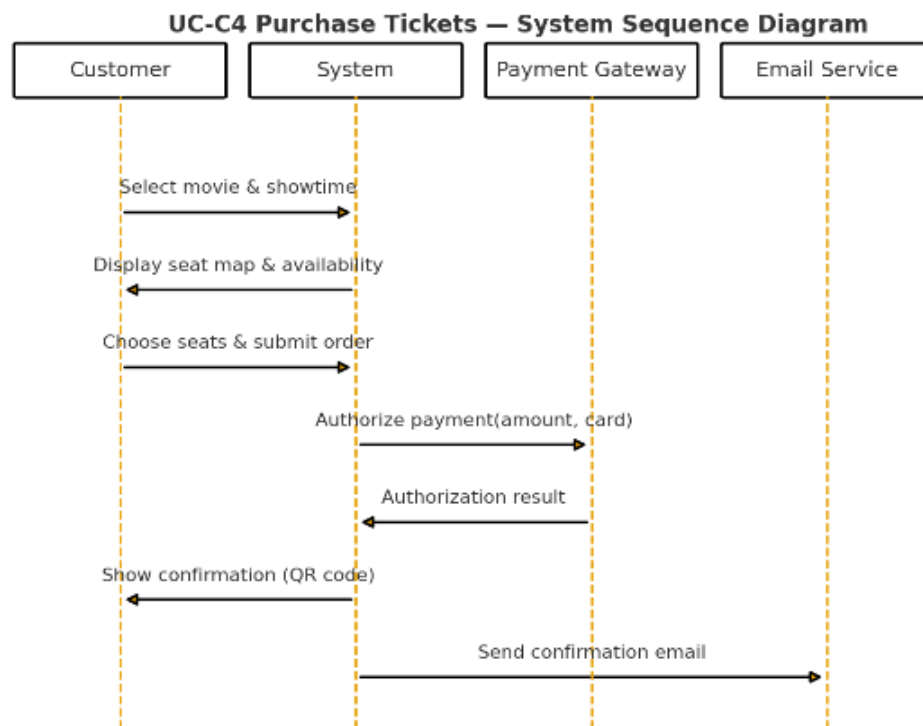
1. Employee logs into Admin Dashboard
2. Employee selects manages movies
3. Employee adds new movie (title, description, poster, runtime, rating)
4. System validates data and saves movie to database
5. Updated listings are visible to customers in real time

Alternate Flows

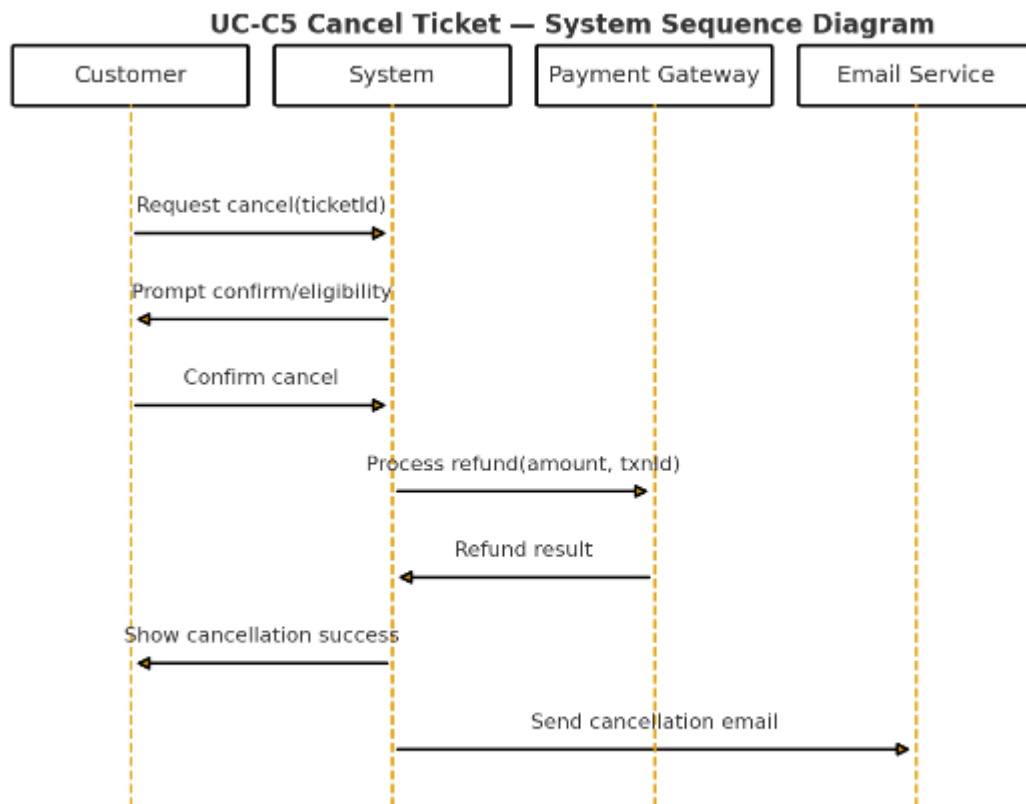
1. If data is missing, system prompts employee to complete form
2. If duplicate movie exists, system alerts employee

System Sequence Diagrams

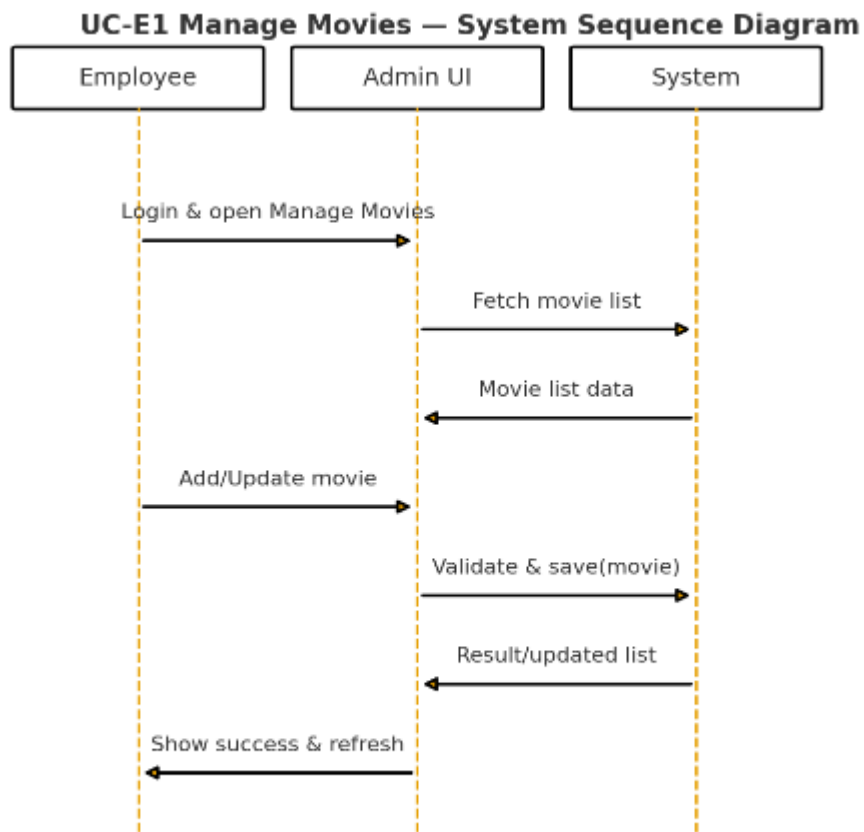
Ticket Purchase



Ticket Cancellation



Manage Movies



User Interface Specification

Ticket Purchase (UC-C4)

Mockup Description

1. Movie Listings page > Buy Ticket Button
2. Showtimes and seat map screen > click seats (green = available, red = taken, blue = selected)
3. Payment page > enter credit card into > confirm purchase
4. Confirmation page > displays ticket > email sent

Effort Estimation

1. Browse to movie – 2 clicks
2. Select showtime – 1 click
3. Select seat – 2 clicks
4. Enter payment – around 20 keystrokes, 1 click
5. Confirm purchase – 1 click
6. Total 25-27 actions

Ticket Cancellation (UC-C5)

Mockup Description

1. Dashboard > Purchase history
2. Select Ticket > Cancel
3. Confirm cancel > refund and email sent

Effort Estimation

1. Navigate – 2 clicks
2. Select ticket – 1 click
3. Confirmation cancel – 1 click
4. Total 4 clicks

Manage Movies (UC-E1)

Mockup Description

1. Dashboard > Sidebar menu > manage movies
2. Mange movies page > Table of existing movies (title, runtime, poster thumbnail, edit/delete buttons)
3. Add movie form > Text fields (title, runtime, description), dropdown (rating), file upload (poster)
4. Edit movie > opens same form with fields pre-filled
5. Delete movie > confirmation dialog with yes/cancel buttons
6. System Response > Movie successfully added/updated/deleted – Listings refresh in real-time

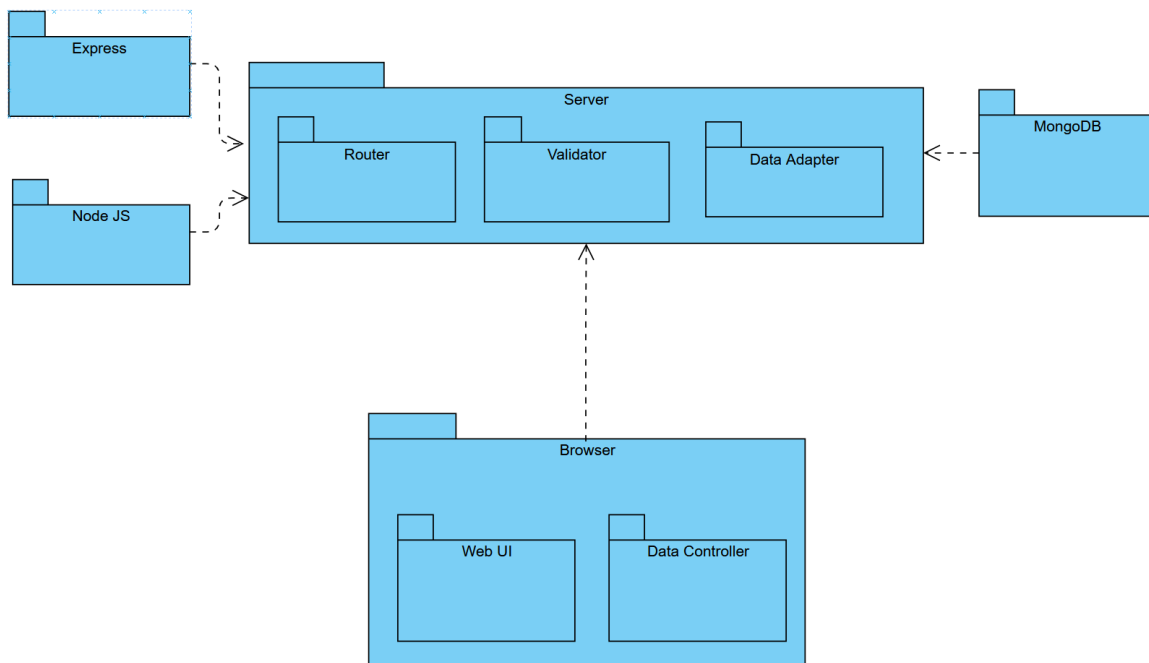
Effort Estimation

1. Navigate to admin dashboard – 2 clicks
2. Select manage movies – 1 click
3. Click add move – 1 click
4. Fill fields
 - a. Title – 10 keystrokes
 - b. Description – 30 keystrokes
 - c. Run time – 3 keystrokes
 - d. Rating 1 click
 - e. Poster 1 click
5. Save movie – 1 click
6. Confirmation
7. Total 50 actions

System Architecture

Identifying Subsystems

1. Customer Interface
 - a. Provides the web and mobile frontends for customers to browse movies, select seats, and purchase tickets.
2. Admin Interface
 - a. Provides the web dashboard for theater staff to manage movies, showtimes, and ticket pricing.
3. Backend Services
 - a. Implements business logic for authentication, user management, ticketing, and payment processing.
4. Database Subsystem
 - a. Stores user accounts, movies, showtimes, tickets, and sales history.



Architecture Styles

The system follows a client server architecture, where the client communicates with a central backend server that handles logic and database operations. The system can be described as a layered architecture:

1. Presentation Layer (Customer/Admin Interfaces)
2. Business Logic Layer (Backend Services)
3. Data Layer (Database Subsystem)

Mapping Subsystems to Hardware

1. Client
 - a. Runs in a standard web browser or mobile browser
2. Server
 - a. Runs backend services on a web server
3. Database
 - a. Runs on a database server or cloud-hosted database system

Connectors and Network Protocols

1. HTTP/HTTPS
 - a. Used for communication between clients and backend server
2. JDBC/SQL
 - a. Used for communication between backend services and the database
3. No custom socket programming or RMI is required

Global Control Flow

1. The system is event driven. Users can perform actions in any order (log in, browse movies, purchase tickets, cancel orders).
2. The backend listens for requests and responds accordingly
3. The system is not real time, it is asynchronous and response based
4. No timers or periodic processes are required for core functionality.

Hardware Requirements

1. Client Side
 - a. Any modern web or mobile browser with internet access, screen display, and basic input devices.
2. Server Side
 - a. A machine capable of running a web server and backend framework (Express.js or Java/Spring) with sufficient CPU/RAM to handle concurrent users
3. Database

- a. Persistent storage system for user, movie, ticket data
4. No specialized hardware is required.

Project Size

Unadjusted Actor Weight (UAW)		
Actor Type	Description	Weight
Simple	The actor is another system which interacts with our system through a defined application programming interface (API).	1
Average	The actor is a person interacting through a text-based user interface, or another system interacting through a protocol, such as a network communication protocol.	2
Complex	The actor is a person interacting via a graphical user interface.	3

Unadjusted Use Case Weight (UUCW)		
Use Case Category	Description	Weight
Simple	Simple user interface. Up to one participating actor (plus initiating actor). Number of steps for the success scenario: ≤ 3 . If presently available, its domain model includes ≥ 3 concepts.	5
Average	Moderate interface design. Two or more participating actors. Number of steps for the success scenario: 4 to 7. If presently available, its domain model includes between 5 and 10 concepts.	10
Complex	Complex user interface or processing. Three or more participating actors. Number of steps for the success scenario: ≥ 7 . If available, its domain model includes ≥ 10 concepts.	15

Use Case	UAW	UUCW	Total
UC-C1	5	10	15
UC-C2	4	5	9
UC-C3	4	10	14
UC-C4	6	10	16
UC-C5	6	10	16
UC-C6	4	5	9
UC-E1	4	5	9
UC-E2	4	5	9

UC-E3	4	5	9
UC-E4	4	10	14
UC-E5	4	5	9
UC-M1	4	10	14
UCP Total			143

Technical Factor	Description	Weight	Perceived Complexity	Calculated Factor (Weight x P.C.)
T1	Distributed system (running on multiple machines)	2	0	0
T2	Performance objectives (are response time and throughput performance critical?)	1	1	1
T3	End-user efficiency	1	2	2
T4	Complex internal processing	1	0	0
T5	Reusable design or code	1	1	1
T6	Easy to install (are automated conversion and installation included in the system?)	0.5	4	2
T7	Easy to use (including operations such as backup, startup, and recovery)	0.5	5	2.5
T8	Portable	2	1	2
T9	Easy to change (to add new features or modify existing ones)	1	2	2
T10	Concurrent use (by multiple users)	1	5	5
T11	Special security features	1	1	1
T12	Provides direct access for third parties (the system will be used from multiple sites in different organizations)	1	0	0
T13	Special user training facilities are required	1	0	0
Total				16.5

Environmental Factor	Description	Weight	Perceived Impact	Calculated Factor
E1	Familiar with the development process (e.g., UML-based)	1.5	2	3
E2	Application problem experience	0.5	2	1
E3	Paradigm experience (e.g., object-oriented approach)	1	3	3
E4	Lead analyst capability	0.5	3	1.5
E5	Motivation	1	4	4
E6	Stable requirements	2	2	4
E7	Part-time staff	-1	5	-5
E8	Difficult programming language	-1	0	0
Total				11.5

UUCP	TCF	ECF
143	0.7	1.1

UCP
110.11

Plans of Work

Phase 1

1. Setup & Design
 - a. Configure GitHub repo
 - b. Design database schema and draft UI mockups
 - c. Define major use cases

Phase 2

1. Core Functionality
 - a. Implement customer ticket submission system
 - b. Build admin login and dashboard
 - c. Connect frontend with backend/database

Phase 3

1. Extended Features
 - a. Add ticket status tracking (open, in progress, closed)
 - b. Implement admin assignment of tickets
 - c. Enhance UI for mobile responsiveness

Phase 4

1. Testing and Integration
 - a. Conduct unit and integration testing
 - b. Refactor code based on feedback
 - c. Document system architecture and usage instructions

Phase 5

1. Finalization
 - a. Prepare final demo
 - b. Complete documentation and user guide
 - c. Deploy project to hosting environment

References

https://eceweb1.rutgers.edu/~marsic/books/SE/book-SE_marsic.pdf - textbook