
Software Requirements Specification

for

Los Portales Theater

Version 1.0 approved

Prepared by Justyn Rippie, Al Yazzie, Matt Mitchell, Nick Thompson

ENMU CS472: Software Engineering

February 26, 2022

Table of Contents

1.	<i>Introduction</i>	1
1.1	Overview	1
2.	<i>Overall Description</i>	1
2.1	User Classes and Characteristics	1
2.2	Operating Environment	1
2.3	User Documentation	1
2.4	Assumptions and Dependencies	2
3.	<i>External Interface Requirements</i>	2
3.1	User Interfaces	2
3.2	Software & Communication Interfaces	2
4.	<i>System Features</i>	2
5.	<i>Other Nonfunctional Requirements</i>	9
5.1	Performance Requirements	9
5.2	Security Requirements	9
5.3	Software Quality Attributes	10
5.4	Business Rules	10
6.	<i>Supporting Documents</i>	10

Revision History

Name	Date	Reason For Changes	Version
Original	2/26/2022		1.0

1. Introduction

1.1 Overview

The purpose of this document is to outline the software developed to allow the administrator of the Los Portales theater to effectively allow customers to purchase tickets from a web application and track those ticket sales. This document provides an overview of that application including both the frontend and backend processes. This document is intended for both software developers and stakeholders.

The following are specific requirements that must be provided to the client:

- *A graphical system to select seats. Seats sold must be marked in red and seats available must be marked in green.*
- *The system must display all plays and their show times and dates within a year*
- *A customer must be able to select multiple seats from one or many shows*
- *A customer must be able to add a seat(s) to the shopping cart.*
- *The system needs to have a customer registered in order to buy seats*
- *The shopping cart should be editable for the customer.*
- *The system must inform the user of a successful transaction showing the seats specific to a play*
- *The system must always be online*
- *A theater admin can create, update and delete seat prices for a specific play*
- *A theater admin must be able to generate reports of sales for a specific play*

2. Overall Description

2.1 User Classes and Characteristics

Please refer to the following URL to and see the file titled: Theater UML class.

<https://github.com/rip97/CS472-Project/tree/main/Stage-1>

2.2 Operating Environment

The ticketing and reporting web application will be hosted by a cloud hosting provider whom will assume responsibilities for the following services:

- *Microsoft Sever SQL Database*
- *HTTPS Certificate*
- *Capable of hosting Microsoft .NET-Core Web Application*
- *Domain Name Service*

2.3 User Documentation

A User Manual will be provided upon software delivery. The following topics will be included in such document:

- Creating a show
- Updating a show
- Canceling a show
- Generating a sales report
- Logging into the system
- Creating a new Admin account

For Customer inquiries, an FAQ will be available on the website.

2.4 Assumptions and Dependencies

Assumptions will help the vendor define timelines, the scope of the project, and effort needed to complete the project.

Technology Assumptions

- The client has no current Information Technology Infrastructure
- The client has one desktop computer that is connected to the internet
- The client will need a cloud hosting platform to host the system and will include:
 - Microsoft Sever SQL Database
 - HTTPS Certificate
 - Capable of hosting Microsoft .NET-Core Web Application
 - Domain Name Service

3. External Interface Requirements

3.1 User Interfaces

Please refer to the following URL to see documents tied to this software
<https://github.com/rip97/CS472-Project/tree/main/Stage-1>

3.2 Software & Communication Interfaces

The client will need a cloud hosting platform to host the system and will include:

- Microsoft Sever SQL Database
- HTTPS Certificate
- Capable of hosting Microsoft .NET-Core Web Application
- Domain Name Service

The cloud hosting provider requires the following communication standards:

- HTTPS (Port 80, 443)
- SMTP Server (Port 25)
- DNS (Port 53)
- NTP (Port 123)

4. System Features

Use Case ID:	001		
Use-Case Name:	Customer Registers		
Created By:	Justyn Rippie	Last Updated By:	
Date Created:	2/23/2022	Date Last Updated:	
Actors:	Customers		
Description:	The customer will need to register with the booking system by providing their: first name, last name, email address, phone number, date of birth. By registering with the booking system, this allows them to complete a purchase and allow an administration to run reports on ticket sales.		
Preconditions:	None		
Postconditions:	User will be able to select tickets and purchase tickets.		
Priority:	High		
Frequency of use:	Low		
Normal Course of Events:	A new customer will visit the booking systems and will require them to create an account in order purchase tickets.		
Alternative Courses:	If a customer tries to create a preexisting account, the system will notify the user of their account and ask them to log in to compete the purchase.		
Exceptions:	None		
Includes:	None		
Special Requirement	None		
Assumptions:			
Notes and Issues:	N/A		

Use Case ID:	002		
Use-Case Name:	Customer adds ticket to shopping cart		
Created By:	Justyn Rippie	Last Updated By:	
Date Created:	2/23/2022	Date Last Updated:	
Actors:	Customers		
Description:	A customer will be presented with a list of plays with and the dates they are showing. They will then click on the name of the play to select tickets and be presented with the theater seats to choose a seat(s). When they click on a seat(s) they will be asked if they want to add this to their shopping cart. A customer will be able to do this for different plays.		
Preconditions:	A customer must have a register account with the booking system.		
Postconditions:	A customer can proceed to checkout		
Priority:	High		
Frequency of use:	High		
Normal Course of Events:	A customer will be able to add several seats associated with any show to their shopping cart.		
Alternative Courses:	If the customer does not have an account, the system will redirect them to create an account to purchase tickets and explain why this is necessary.		
Exceptions:	None		
Includes:	None		
Special Requirement	A customer needs an account with the system.		
Assumptions:	A user will have already a registered account with the system.		
Notes and Issues:	N/A		

Use Case ID:	003		
Use-Case Name:	Customer Completes Purchase		
Created By:	Justyn Rippie	Last Updated By:	
Date Created:	2/23/2022	Date Last Updated:	
Actors:	Customers		
Description:	Once a customer has selected the seats for the plays they wish to attend, they complete their purchase by clicking check out in their shopping cart. The user will then provide a payment method (credit card) to complete their purchase.		
Preconditions:	A user must have a preexisting account.		
Postconditions:	User's cart will be emptied, and seats will be sold.		
Priority:	High		
Frequency of use:	High		
Normal Course of Events:	A normal check out will result in a successful transaction of the customers seats for the play(s). The system will notify the user of the successful transaction and email them a receipt of their purchase and information the seats and play. The users shopping cart will then be cleared.		
Alternative Courses:	If the payment method is denied, the system should alert the user if the transaction fails and not complete the purchase and allow them to try to enter their payment method again. All items in the cart should remain.		
Exceptions:	None		
Includes:	None		
Special Requirement	None		
Assumptions:	The user will already have an existing account		
Notes and Issues:	N/A		

Use Case ID:	004		
Use-Case Name:	Theater Admin can create play		
Created By:	Justyn Rippie	Last Updated By:	
Date Created:	2/26/2022	Date Last Updated:	
Actors:	Theater Admin		
Description:	A theater admin will log into the administrator side of the system. They will then be able to select an option to create a new play and edit the following: show times/dates, seat prices, title, description.		
Preconditions:	Must be a Theater Admin User		
Postconditions:	A new play is added to the		
Priority:	High		
Frequency of use:	High		
Normal Course of Events:	A new play will be added to the system and stored into the database. The customer will and theater admin will be able to view the newly created show.		
Alternative Courses:	If the show ID in the database already exists, the system should alert the user that the show exists and present the name and show ID.		
Exceptions:	None		
Includes:	None		
Special Requirement	None		
Assumptions:	The theater admin will be trained on how to perform this action		
Notes and Issues:	N/A		

Use Case ID:	005		
Use-Case Name:	Theater Admin Updates Show		
Created By:	Justyn Rippie	Last Updated By:	
Date Created:	2/26/2022	Date Last Updated:	
Actors:	Theater Admin		
Description:	A theater admin will log in into the Administrator side of the system and have an option to update any information for any show. This will include updating: show times/dates, seat prices, title, description		
Preconditions:	Must be a Theater Admin User		
Postconditions:	A show will update the selected information the theater admin updates		
Priority:	High		
Frequency of use:	Medium		
Normal Course of Events:	The selected show fields will be updated from what the theater admin changes. The system should ask the user if they are sure they want to update before database is written with new information.		
Alternative Courses:	If any action fails, the system should not update the database and keep the old information.		
Exceptions:	None		
Includes:	None		
Special Requirement	None		
Assumptions:	A theater admin will have training on this selected action		
Notes and Issues:	N/A		

Use Case ID:	006		
Use-Case Name:	Theater Admin Deletes Show		
Created By:	Justyn Rippie	Last Updated By:	
Date Created:	2/26/2022	Date Last Updated:	
Actors:	Theater Admin		
Description:	A theater admin will have the option to delete a show. The delete option will remove the show from the system and the database.		
Preconditions:	Must be a Theater Admin User		
Postconditions:	A show will update the selected information the theater admin updates		
Priority:	High		
Frequency of use:	Medium		
Normal Course of Events:	The system will locate the show name and remove it from the system. The theater admin will revive a warning they're about to delete and ask if they want to proceed. If yes, the system will remove the show; the theater admin will receive a notification the show was removed. If the theater admin selects no, the system will not remove the show.		
Alternative Courses:	If the system cannot locate or delete the show, the system should do nothing and inform the theater admin that deletion cannot be performed, or the show name cannot be found.		
Exceptions:	None		
Includes:	None		
Special Requirement	None		
Assumptions:	A theater admin will have training on this selected action		
Notes and Issues:	N/A		

Use Case ID:	007		
Use-Case Name:	Theater Admin Generates Sales Script		
Created By:	Justyn Rippie	Last Updated By:	
Date Created:	2/26/2022	Date Last Updated:	
Actors:	Theater Admin		
Description:	A theater admin has an option to generate a report on ticket sales. The report will include the show name. The report will tell the theater admin the number of seats sold and the revenue generated. They should also see how many tickets are left.		
Preconditions:	Must be a Theater Admin User		
Postconditions:	The report will be generated and presented to the theater admin.		
Priority:	High		
Frequency of use:	High		
Normal Course of Events:	The report will list the show name, number seats sold, the total amount revenue generated for the show, and the number of seats left. The report should include the seat number sold and seat number left.		
Alternative Courses:	If the report cannot be generated, the system should alert the user with an error to report back to vendor so they may contact support.		
Exceptions:	None		
Includes:	None		
Special Requirement	None		
Assumptions:	A theater admin will have training on this selected action		
Notes and Issues:	N/A		

5. Other Nonfunctional Requirements

5.1 Performance Requirements

Interaction with the system from either the Administrator or Customer will not take significant amounts of time. This will include when the customer is purchasing tickets or modifying their account information. It will also allow the system administrator to effectively update play information seamlessly.

5.2 Security Requirements

The system will protect customer account information:

- Credit/Debit Card Information
- Addresses
- Email

- Full Name
- Date of Birth
- Username and Password

The software should adhere to C#/.NET Key Security Concepts provided by Microsoft at:
<https://docs.microsoft.com/enus/dotnet/standard/security/key-security-concepts>

5.3 Software Quality Attributes

Availability: The web application will be running 24/7-365

Reliability: Dependent on cloud hosting provider (should have no less than 5% downtime)

Portability: With this being a web application, it does not require on-prem software installation (i.e. no executable files).

Usability: Will provide a simplistic interface for both Admins and Customers

5.4 Business Rules

Customer:

- Logging into the system
- Creating a new account
- Purchasing tickets
- Browse available plays

System Admin:

- Creating a show
- Updating a show
- Canceling a show
- Generating a sales report
- Logging into the system
- Creating a new Admin account

6. Supporting Documents

Please refer to the following documents listed in GitHub repository for further information:
<https://github.com/rip97/CS472-Project/tree/main/Stage-1>

Appendix A: Analysis Models

Please refer to the following documents listed in GitHub repository:

- 1) Theater UML Class.pdf
- 2) Cost Estimate – Los Portales.pdf
- 3) Sequence Diagrams – Los Portales.pdf
- 4) Stage 1_Critical Path Method.pdf
- 5) State 1_PERT Diagram.pdf
- 6) Uses Cases – Los Portales.pdf