

CinemaPark

Web Application Documentation

By:
Tomadur Alamro

July 2018

Contents

- 1 Introduction 2
- 2 Analysis 2
 - 2.1 User Requirements 2
 - 2.1.1 Functional Requirements 2
 - 2.1.2 Non-Functional Requirements 2
 - 2.2 Use-case Diagram 3
 - 2.3 Scenario Description 3
 - 2.4 User Messages 6
- 3 Testing 6
 - 3.1 Usability Testing 6
 - 3.2 Functionality Testing 8

1 Introduction

CinemaPark is a web application similar to IMDB. It is a web application for online service guide related to the information of movies. You can find all the newly released movies at this web application. You can also get anything that you want to know about any movie on this web application. With CinemaPark, you can:

- See the latest movies.
- Search movies by movie title.
- Add a movie or remove it from your favorites.
- View movie details (Movie Name, Poster, Movie Description, Genres, Release Date, Rating).

CinemaPark web application is developed using *JavaScript, CSS and HTML*. The CinemaPark is compatible with the most common browsers. Also, I used TMDb. TMDb is a great API to use for searching for and displaying information about movies and TV shows. TMDb is available for everyone to use.

2 Analysis

2.1 User Requirements

2.1.1 Functional Requirements

1. The user shall be able to see the latest movies.
2. The user shall be able to search by movie title.
3. The user shall be able to add movie to his/her favorites.
4. The user shall be able to remove movie from his/her favorites.
5. The user shall be able to view movie details.

2.1.2 Non-Functional Requirements

- Usability : The web application should be easy to use.
- Correctness: The web application should do the functions properly.
- Operability : The web application should be operated in a reliably efficient manner.
- Reliability: The web application should works as and when needed.
- Accuracy : All input fields should be validated.

2.2 Use-case Diagram

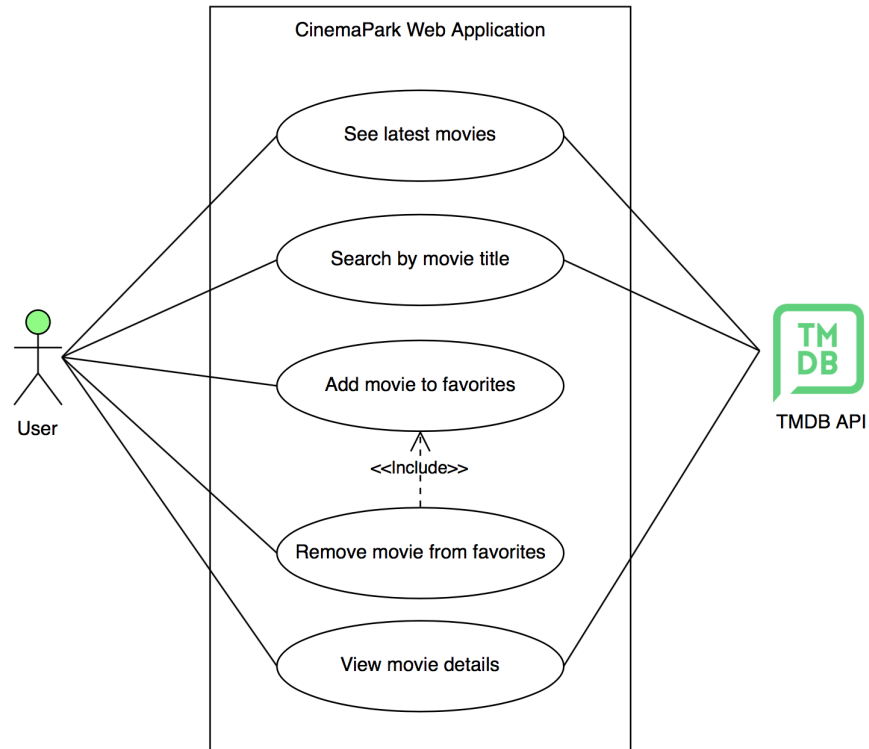


Figure 1: CinemaPark web application use case.

2.3 Scenario Description

Use Case Number	UC01
Use Case Name	See latest movies.
Actors	User, System
Pre-conditions	Internet connection
Basic Flow	<ol style="list-style-type: none">1. The user open CinemaPark home page.2. The system fetches the latest movies from TMDb API.3. The system shows the latest movies on the home page.
Error Flow	None.

Use Case Number	UC02
Use Case Name	Search by movie title.
Actors	User, System
Pre-conditions	Internet connection
Basic Flow	<ol style="list-style-type: none">1. The user open CinemaPark home page.2. The user enters the title of the movie on the search box.3. The user clicks on the search icon.4. The system fetches the movies that match with the entered title.5. The system shows the result on the page.
Error Flow	<p>A.[Error flow A: The user left search box blank After step 3]</p> <ol style="list-style-type: none">1. The system shows MSG01. <p>B.[Error flow B: The user enters an invalid movie title]</p> <ol style="list-style-type: none">1. The system shows MSG02.

Use Case Number	UC03
Use Case Name	Add movie to favorites.
Actors	User, System
Pre-conditions	Internet connection
Basic Flow	<ol style="list-style-type: none">1. The user open CinemaPark home page.2. The user clicks on the movie.3. The user clicks on "Add to my favorites" button.4. The system adds the movie to favorites.5. The system shows MSG003.
Error Flow	<p>A.[Error flow A: The user clicks on "Add to my favorites" button while it is already added to favorites]</p> <ol style="list-style-type: none">1. The system shows MSG04.

Use Case Number	UC04
Use Case Name	Delete movie from favorites.
Actors	User, System
Pre-conditions	<ul style="list-style-type: none">• Internet connection.• My favorites contain at least one movie.
Basic Flow	<ol style="list-style-type: none">1. The user open My favorite page.2. The user clicks on trash icon.3. The system shows MSG05.4. The user clicks on "Yes" button.5. The system deletes the movie.6. The system shows MSG06
Error Flow	None.

Use Case Number	UC05
Use Case Name	View movie details.
Actors	User, System and TMDb API
Pre-conditions	Internet connection
Basic Flow	<ol style="list-style-type: none">1. The user clicks on the movie.2. The system loads details from TMDb API.3. The system shows movie selected details.
Error Flow	None.

2.4 User Messages

Message ID	Type	Message
MSG01	Error Message	Enter a word or phrase to search on.
MSG02	Error Message	Sorry, we couldn't find any movies matching.
MSG03	Success Message	Movie added to favorites.
MSG04	Error Message	Movie already added to favorites.
MSG05	Confirm Message	Are you sure you want to delete this movie from your favorites?
MSG06	Success Message	Movie deleted from favorites.

3 Testing

The testing phase consists of testing the CinemaPark web application. The types of testing conducted for CinemaPark web application are: (1) usability and user experience (UX) testing, and (2) functionality testing. The usability testing achieved by designing a questioner consists of several questions that appropriate for the application type and measure multiple factors of the application usability (e.g. learnability and flexibility). The questions designed to have two types of questions, which are rating questions that have three degree (strongly agree, agree, disagree) and open questions that require more than one-word answers and come in a sentence form. Usually, people do not like to answer the open questions, so I add a few of them and make them not required to be answered.

The functionality testing is done by design all possible functional test cases for every use-case of the application. The test case template used in the functional testing has five columns: (1) use-case ID, (2) test case title, (3) test steps, and it contains all the test steps in details and in the order in which it requires to be executed, (4) expected result that including the expected action such as creating file or showing error message, and (5) actual result, which is the result after the test execution.

3.1 Usability Testing

The result of usability testing for CinemaPark web application obtained by asking different users to use the web application and fill the questioner. The CinemaPark web application is evaluated by 5 users. The questions of CinemaPark questioner along with the factor to be measured and the result are as follows:

- Q1 : The web application is easy to use? [Ease to use]
- Q2 : Can you navigate to the home page from any page? [Ease to navigate]
- Q3 : Is it easy to navigate between the web application pages? [Ease to navigate]
- Q4 : Do the buttons correspond to its functionality? [Predictability]
- Q5 : Do the error messages clear and understood? [Ease to understand]
- Q6 (open question) : What do you like about the web application?
- Q7 (open question) : What do you suggest to improve the web application?

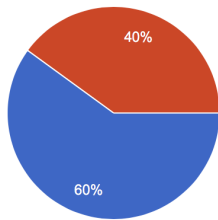


Figure 2: Q.2 Result

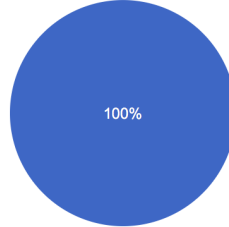


Figure 3: Q.3 Result

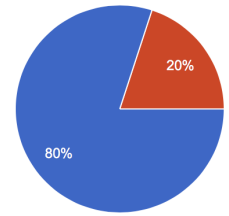


Figure 4: Q.4 Result

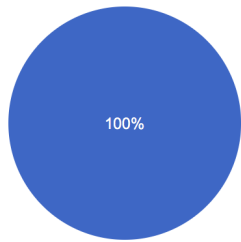


Figure 5: Q.4 Result

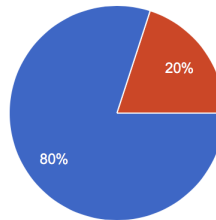
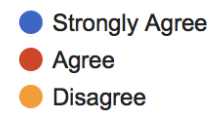


Figure 6: Q.5 Result



What do you like about the web application?

5 responses

beautiful designing
easy to use
I like how it is easy to use and the design
Simplicity
The design is simple and the buttons are self-self-describe

Figure 7: Q.6 Result

What do you suggest to improve the web application?

3 responses

add search to fav list
nothing
Add rating and comments to each movie - share movie with friends through social media

Figure 8: Q.7 Result

3.2 Functionality Testing

The number of test cases for the web application is 8 test cases.

Use Case Number	UC01
Test Case Title	Verify CinemaPark web application shows the latest movies.
Test Steps	<ul style="list-style-type: none">• Click on CinemaPark" home page.
Expected Result	The system shows the latest movies.
Actual Result	As Expected.

Use Case Number	UC02
Test Case Title	Verify Cinema web application shows the movies match with the user entered.
Test Steps	<ul style="list-style-type: none">• Write the movie title on search box• Click on search icon.
Expected Result	The system shows the matched results.
Actual Result	As Expected.

Use Case Number	UC02
Test Case Title	Verify search box does not accept to left it blank.
Test Steps	<ul style="list-style-type: none">• Click on search icon without entering a movie title.
Expected Result	The system the error message: "Enter a word or phrase to search on."
Actual Result	As Expected.

Use Case Number	UC02
Test Case Title	Verify entered title does not exist.
Test Steps	<ul style="list-style-type: none">• Write the movie title on search box• Click on search icon.
Expected Result	The system the error message: "Sorry, we couldn't find any movies matching."
Actual Result	As Expected.

Use Case Number	UC03
Test Case Title	Verify the movie add to favorites.
Test Steps	<ul style="list-style-type: none">• Click on the movie.• Click on "Add to my favorites" button.
Expected Result	The system adds the movie to favorites and shows the success message: "Movie added to favorites."
Actual Result	As Expected.

Use Case Number	UC03
Test Case Title	Verify double-click on "Add to my favorites" does not allow.
Test Steps	<ul style="list-style-type: none">• Click on the movie.• Double-click on "Add to my favorites" button.
Expected Result	The system shows the error message: "Movie already added to favorites."
Actual Result	As Expected.

Use Case Number	UC04
Test Case Title	Verify when the user deletes the movie from favorites the system delete it.
Test Steps	<ul style="list-style-type: none">• Open My favorites page.• Click on trash button.• Confirm.
Expected Result	The system shows the success message: "Movie deleted from favorites."
Actual Result	As Expected.

Use Case Number	UC05
Test Case Title	Verify CinemaPark show movie details
Test Steps	<ul style="list-style-type: none">• Click on the movie.
Expected Result	The system shows selected movie details.
Actual Result	As Expected.