Software Requirements Specification

for

MovieTracker

Prepared by Dragan Valeriu
Butnaru Dan
Ursu Catalin-Eugen

Faculty of Automatic Control and

Computer Engineering “Gheorghe Asachi”

May 22, 2024

Table of Contents

Revision History

Name	Date	Reason For Changes	Version

1. Introduction

1.1 Purpose

This SRS documents the requirements for a C# Windows Forms application designed for a college project. The application allows users to search for movies and retrieve detailed information about these movies from an external API. Additionally, the application displays trending movies and allows users to rate them, storing these ratings in a local database.

1.2 Document Conventions

This document follows the IEEE standard formatting for software development. The standard defines a regular formatting this document follows including writing to be done in third-person, passive voice as well as readable and grammatically correct text.

1.3 Intended Audience and Reading Suggestions

This document is intended for University teachers and colleagues who want to know more information about this project and it's details.

1.4 Product Scope

The application aims to provide an accessible and user-friendly platform for movie enthusiasts to explore and rate movies. It connects to an external movie database API to fetch and display movie information and trends, enhancing user interaction and engagement through a dynamic and responsive interface.

1.5 References

The API from where we extracted informations about the movies: The Movie Database (TMDB) https://developer.themoviedb.org/docs/

2. Overall Description

2.1 Product Perspective

The Movie Explorer App is a new, self-contained software product designed specifically as a project for a university-level course on software development with C#. It is not part of an existing product family nor does it replace any existing systems. Instead, it is intended to serve as an educational tool that enables students to learn and demonstrate proficiency in several key areas of software engineering, including API integration, user interface design, database management, and application of software development best practices.

Relation to Larger Systems:

While the Movie Explorer App is designed as a standalone product for educational purposes, the skills and technologies it encompasses are applicable to larger, more complex systems in a real-

world setting. Students are expected to draw parallels between this project and commercial software development practices, thus preparing them for future projects that may involve more complex integrations and functionalities.

2.2 Product Functions

- **Search Functionality**: Allows users to search for movies using a search bar.
- **Display Trending Movies**: Shows a selection of trending movies on the main page.
- **Movie Details**: Users can click on any movie image to view detailed information about the movie.
- **Rating System**: Users can rate movies, with ratings being stored in a local database.

2.3 User Classes and Characteristics

- **Casual Users**: Typically movie fans looking for a simple tool to explore movies.
- **Academic Users**: Students and faculty who might use the application to analyze user preferences and data trends in movies for academic purposes.

2.4 Operating Environment

Since this program is developed in .NET Framework 4.8, user's workstations should have at least Microsoft Windows 7 operating system and internet connection.

2.5 Design and Implementation Constraints

- Must be developed using C# Windows Forms.
- Must use the specific API provided for the project.
- Must operate within the security constraints of the university's IT network.

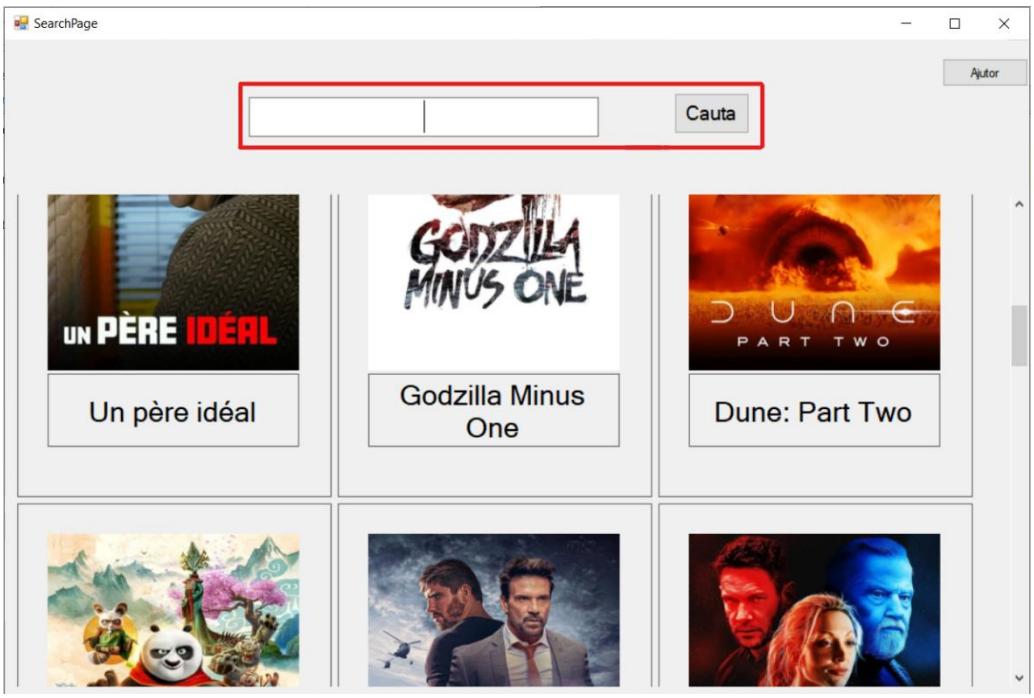
2.6 User Documentation

The user documentation is created using HelpNDoc Help Authoring Tool. The help can be accessed in the application by clicking on "Ajutor" in the top right corner.

2.6.1 Starting page

- This is the page seen when the program is opened.

- This page contains a search bar which can help you search for movies.

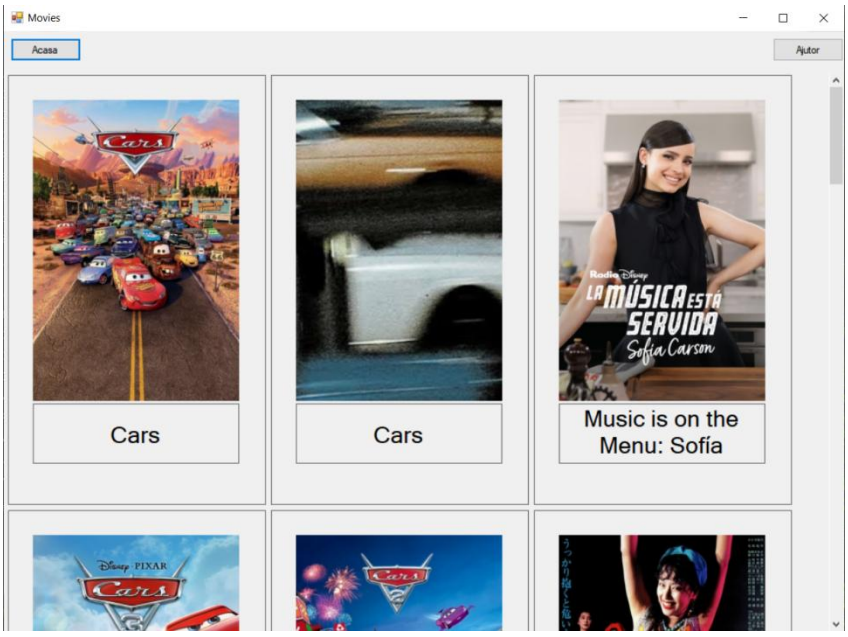


- Under the search bar you can see the movies which are trending at the moment.
- By clicking on any movie image, a new window opens with information about the clicked movie.

2.6.2 Searched movies page

- This page is the one that appears after searching for a movie.

For example this is the search page for "cars":



- In this page you can click on any movie and get more information about it.
- By clicking on "Acasa", you are redirected to the Starting page.

2.6.3 Movie information page

- This page appears after clicking a movie.

- In the right side of the window you can see some information like a rating, release date and a short description.
- Under the photo, you have a rating section where you can rate the movie from 0 to 10.
- By clicking on "Acasa", you are redirected to the Starting page.

2.7 Assumptions and Dependencies

This project strongly depends on The Movie Database (TMDB) API which creates the vast majority of the content used in the application. We assume users have basic operational knowledge of Windows applications.

3. External Interface Requirements

3.1 User Interfaces

- Main window with a search bar at the top.

- Trending movies displayed below the search bar.
- Each movie entry has a clickable interface that shows details and a rating option.

3.2 Hardware Interfaces

Hardware Interfaces will include a mouse, the keyboard, and the display monitor. There is not much heavy hardware needed to run the program other than a simple computer with Windows and a monitor. The mouse left click will allow the user to interact with certain elements. The keyboard is used for typing any movie name for searching details.

3.3 Software Interfaces

API for movie data: RESTful interface, returns JSON.

3.4 Communications Interfaces

Requires HTTP/HTTPS for API communication.

4. System Features

4.1 Feature: Movie Search

- Users can enter search terms to find movies.
- The system queries the API and displays results.

4.2 Feature: Movie Details and Ratings

- Users can click on a movie to view detailed information and submit a rating.
- Ratings are stored in the local database and are used to compute average scores displayed to users.

5. Other Nonfunctional Requirements

5.1 Performance Requirements

The application must be able to run on a minimum of Windows 7. It is recommended to have at least 10 Mb Storage, and 1GB RAM.

5.2 Safety Requirements

This program is safe to use for anyone.

5.3 Security Requirements

This program does not gather any sensitive personal data.

5.4 Software Quality Attributes

- **Usability**: Easy to use interface with clear navigation.
- **Reliability**: Robust handling of API interactions and local database transactions.
- **Maintainability**: Code should be well-documented and modular to allow easy updates and maintenance.

5.5 Business Rules

Users must provide a rating between 1 and 5 stars.

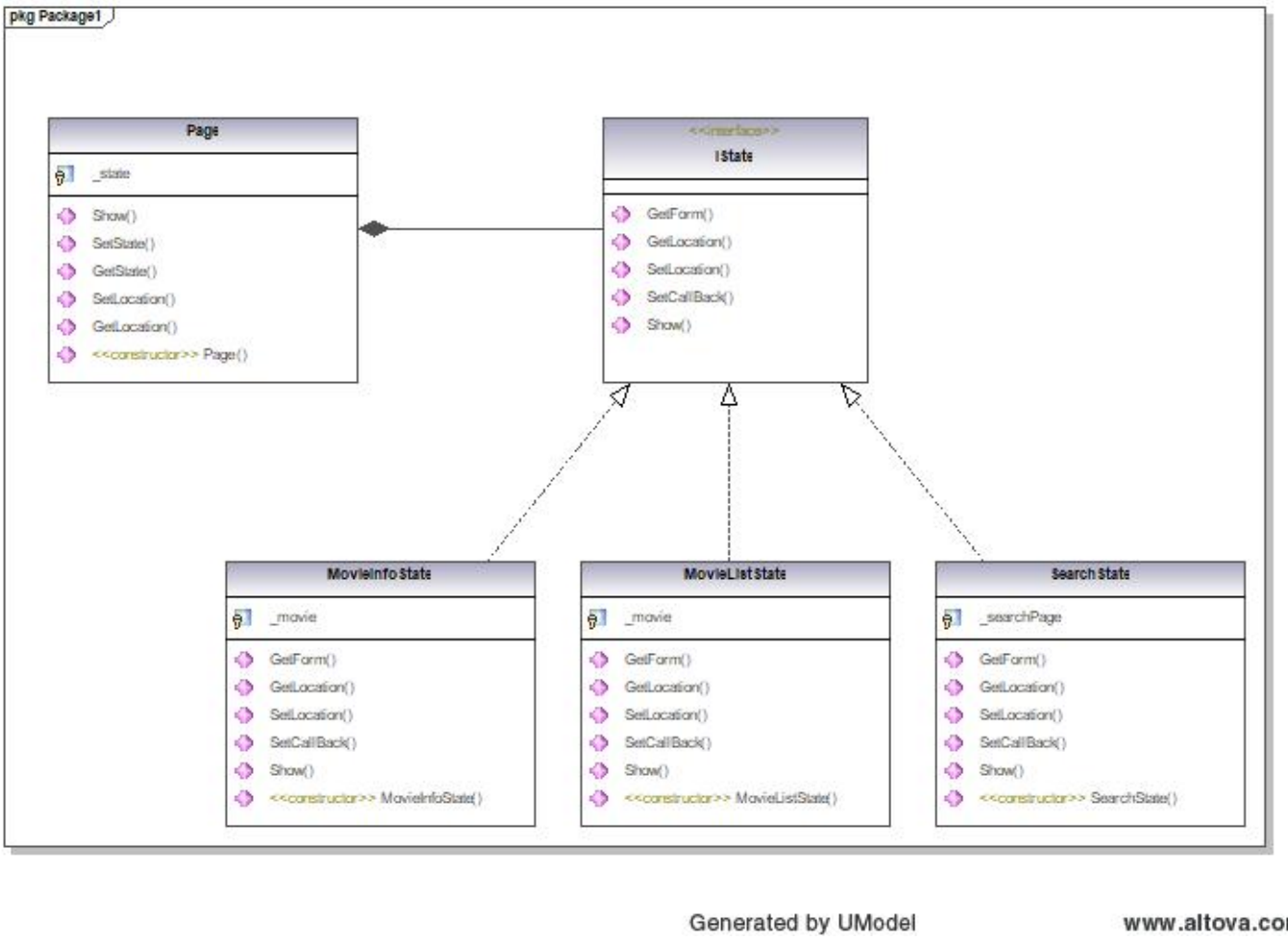
6. Other Requirements

Appendix A: Glossary

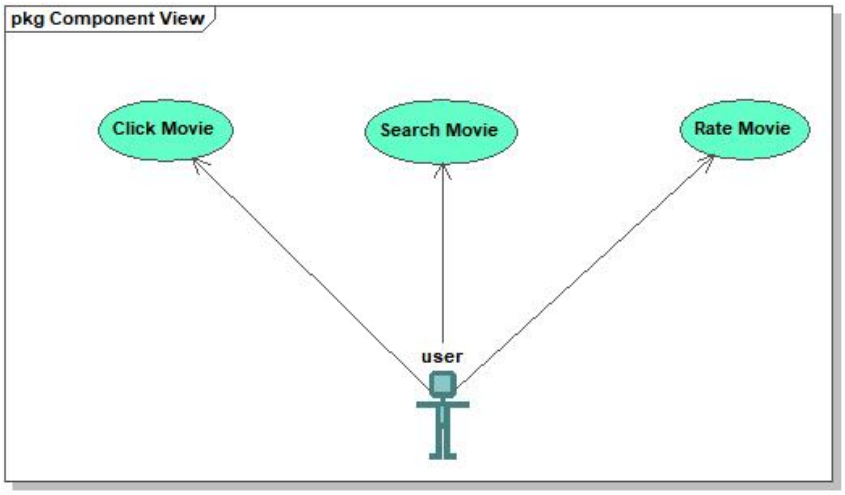
N/A

Appendix B: Analysis Models

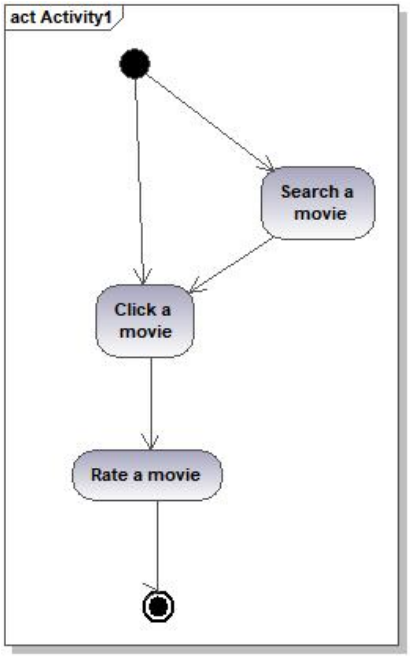
Class Diagram:



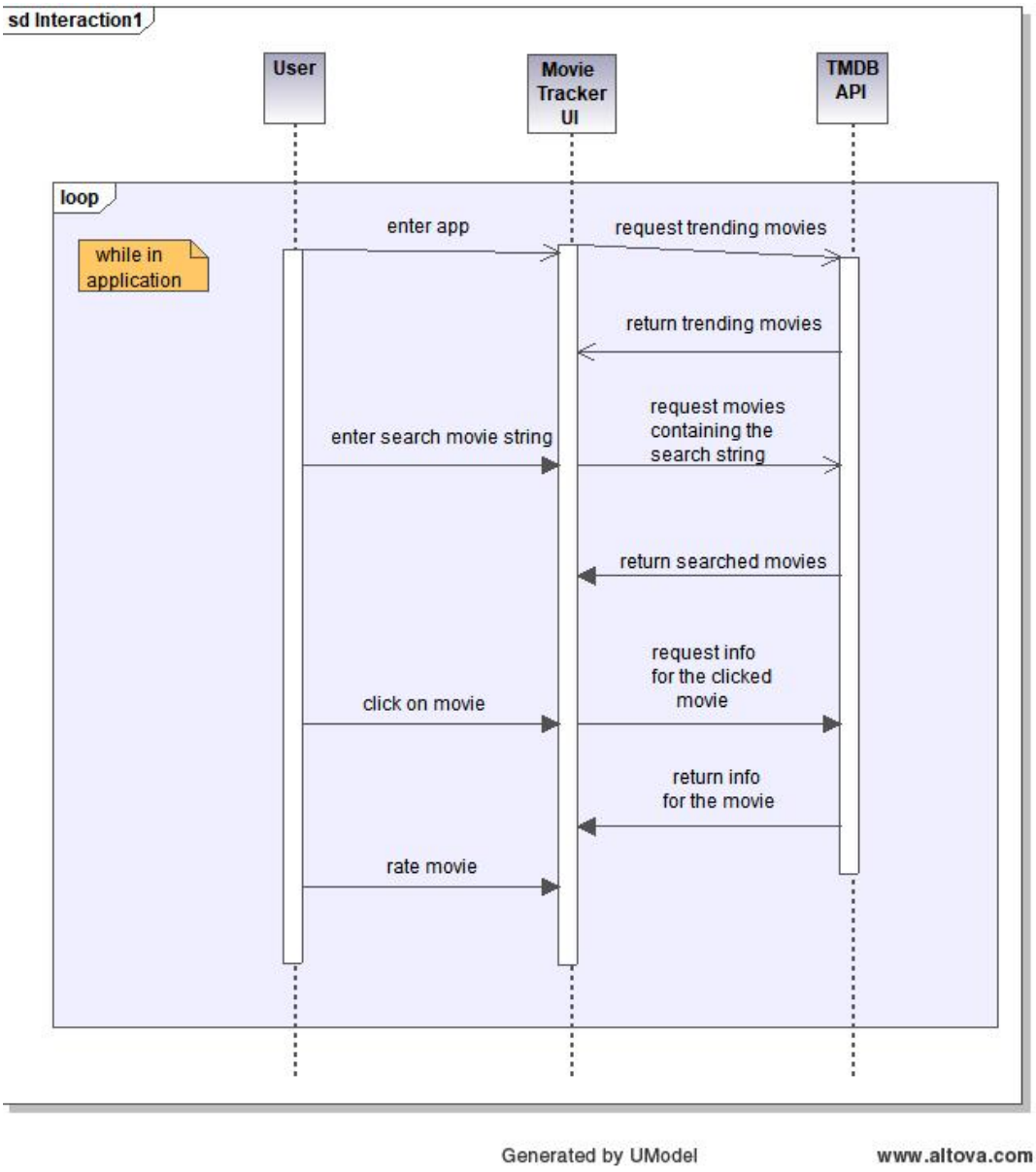
Use Case Diagram:



Activity Diagram:



Sequence Diagram:



Appendix C: Scenarios

N/A