CSE 3105/CSE 3137

OBJECT ORIENTED ANALYSIS AND DESIGN

FALL 2020

COURSE PROJECT: *Media Browser Application*

Requirements Analysis Document

Group 1

Ecenur Atıgan– 180315063

Çisem Akman– 170315020

Barış Alp Aslan – 170315009

Ömer Şirin – 150315013

Gülben Emiroğlu – 160315041

Erdem Emiroğlu – 170315026

20 November 2020

Table of Contents

1 Introduction

Our program opens and displays movies and behind-the-scenes footage, movie credits and sound files. The aim of the program is to combine movie and music application in one application. We developed it with reference to common apps such as Netflix, Youtube and Spotify.

2 Current System

Our program is similar to Netflix, Fizy, Spotify, ITunes, PuhuTV, BluTV, Amazon Prime, Youtube Music, SoundCloud etc.

- Fizy, Spotify, Youtube Music and SoundCloud are music applications. As a difference, we have added movies, trailers and behind the scenes(images/videos) on these applications.
- Netflix, PuhuTV, BluTV and Amazon Prime are movie/video applications. As a difference, we have added musics and audio files on these applications.
- ITunes is similar to our program, it includes both songs and movies but only people who have IOS can use ITunes. In our program also android users can download.

3 Proposed System

3.1 Overview

This application comes with a number of songs, movies and images- from the oldest to the newly released ones that you can download to your computer. You can give a like and leave comments under them- even create a list with your favorite ones.

It also includes trailers, behind the scenes that you can watch in your own language.

3.2 Functional Requirements

User must be able to;

- Zoom in/out images
- Choose video/movie quality
- Change playback speed
- Show subtitles in any language
- Play/pause videos and musics
- Take forward/ Rewind videos and musics
- Create playlists
- Show lyrics
- Change application language
- Give likes and leave comments
- Download contents

Resource Manager must be able to;

- Add/ Delete/ Modify music, video, images
- Start voting
- Manage voting time

3.3 Nonfunctional Requirements

- ✓ The free time for the movie selected by voting every week is one week. (Usability)
- ✓ User can't comment or like without premium membership. (Usability)
- ✓ User can't download content without premium membership. (Usability)
- ✓ User can login or continue as guest. A guest user can become a premium member and set a password. Premium member can edit their profile. (Reliability)
- ✓ Membership trial period can't be more than 30 days. (Supportability)

3.4 System Models

3.4.1 Scenarios

Identifying Scenario

Scenario name: watchingMovie

Participating actors: alice:User

Alice was a transfer student who came to Turkey from Canada. She saw the movie that her friend watch from computer "Lord of the Rings: The Fellowship of the Ring" and decided to download the application. To watch the rest of the series, she logged in as a guest user. But since she couldn't watch the movie without subscription, she became a Premium member. Then she downloaded the movie to watch it without internet connection. While she was watching the movie, she found the musics that she wondered from Movie description. She created a playlist then added her favourite movie musics. After she finished the movie, in movie description she found backstage videos and photographs. She downloaded a photo that she liked and set wallpaper.

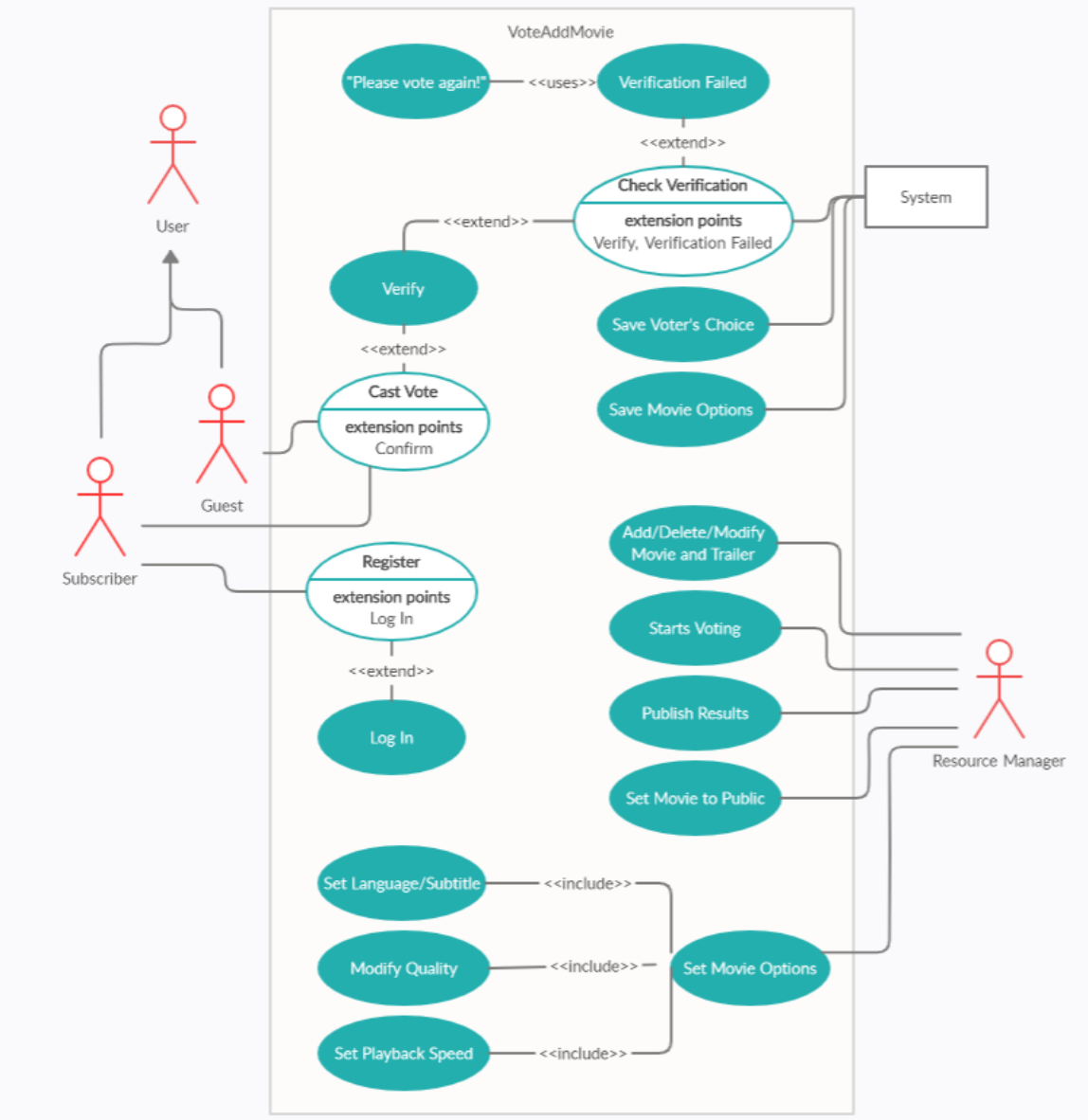
Other Scenarios

- FavouritePhotograph: Admin Jack uploads some photographs and also photographs of the movie Braveheart to the system. Helin the user, because of like Braveheart movie, enters the Watchy application. She likes one of the photographs because of zoom photo and she notice a little detail in the photo. After that, she downloads the photo to her computer.

- PartyMovie: Rory was searching the internet for the movie 'Annabelle' to watch with her friends at her birthday party next week. Watchy application appeared before her and she logged in as a guest user to check the application. She realized that the movies were paid and she had to be a Premium member to watch. However, when she entered the movie tab, she saw that every week a movie was made public for free by voting. Since the movie she was looking for was in the vote, she immediately voted.

- ListenMusic: Jack opened the Watchy app to listen to music. He logged in as a guest user and searched the song "Jolene" by using search bar. He started listening to the top recommended song. He signed up to play the next song, then created a profile. He added a comment to the song then created a playlist and added the song. He backward the song and read the lyrics while listening music.

3.4.2 Use Case Model



Name: VoteAddMovie

Participating Actors: User (Subscriber – Guest), Resource Manager

Flow of Events:

1_ ResourceManager loads the movie into the system by enabling the AddMovie function and organizes the movie options (language, quality, playback speed).

2_ System saves the movie options.

3_ ResourceManager starts voting. If the user is a Premium member, he / she logs in and participates in the vote, otherwise he / she can vote directly as a guest user. Then User verifies his / her vote.

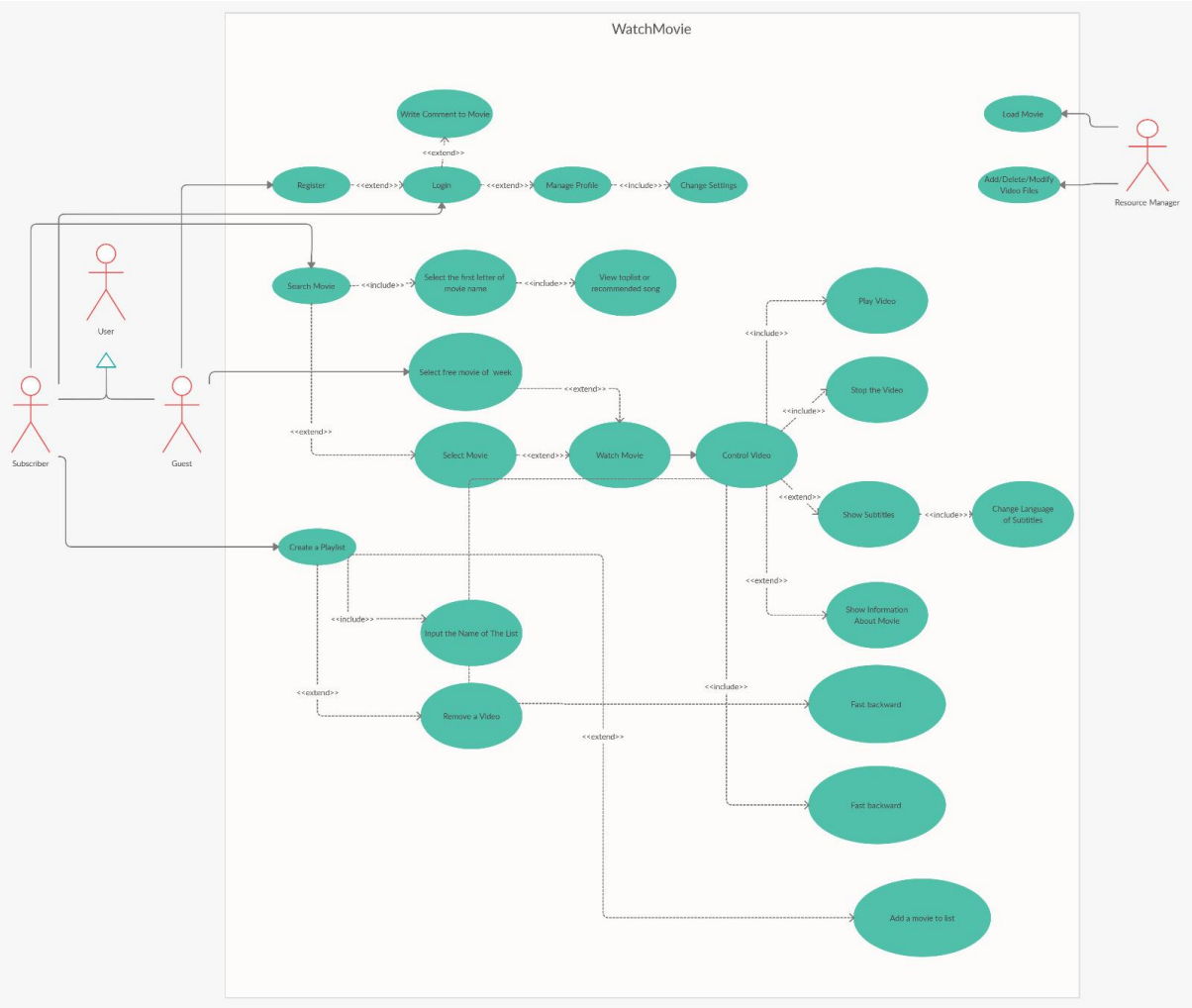
4_ System checks the verification of vote. If not verified, it reflects the message 'Please vote again.' if the vote is verified, it saves User's choice.

5_ ResourceManager evaluates the saved selections and publishes the result. Then ResourceManager sets movie to public for free for one week.

Entry condition= ResourceManager logged in to upload the movie and submit it for voting.

Exit Condition= ResourceManager released the highly rated movie for free.

Qualification Requirements= The free period of the movie is 1 week.



Name: WatchMovie

Participating Actors: User (Subscriber – Guest), Resource Manager

Flow of Events:

1_ ResourceManager loads the movie into the system by enabling the AddMovie function and organizes the movie options (language, quality, playback speed).

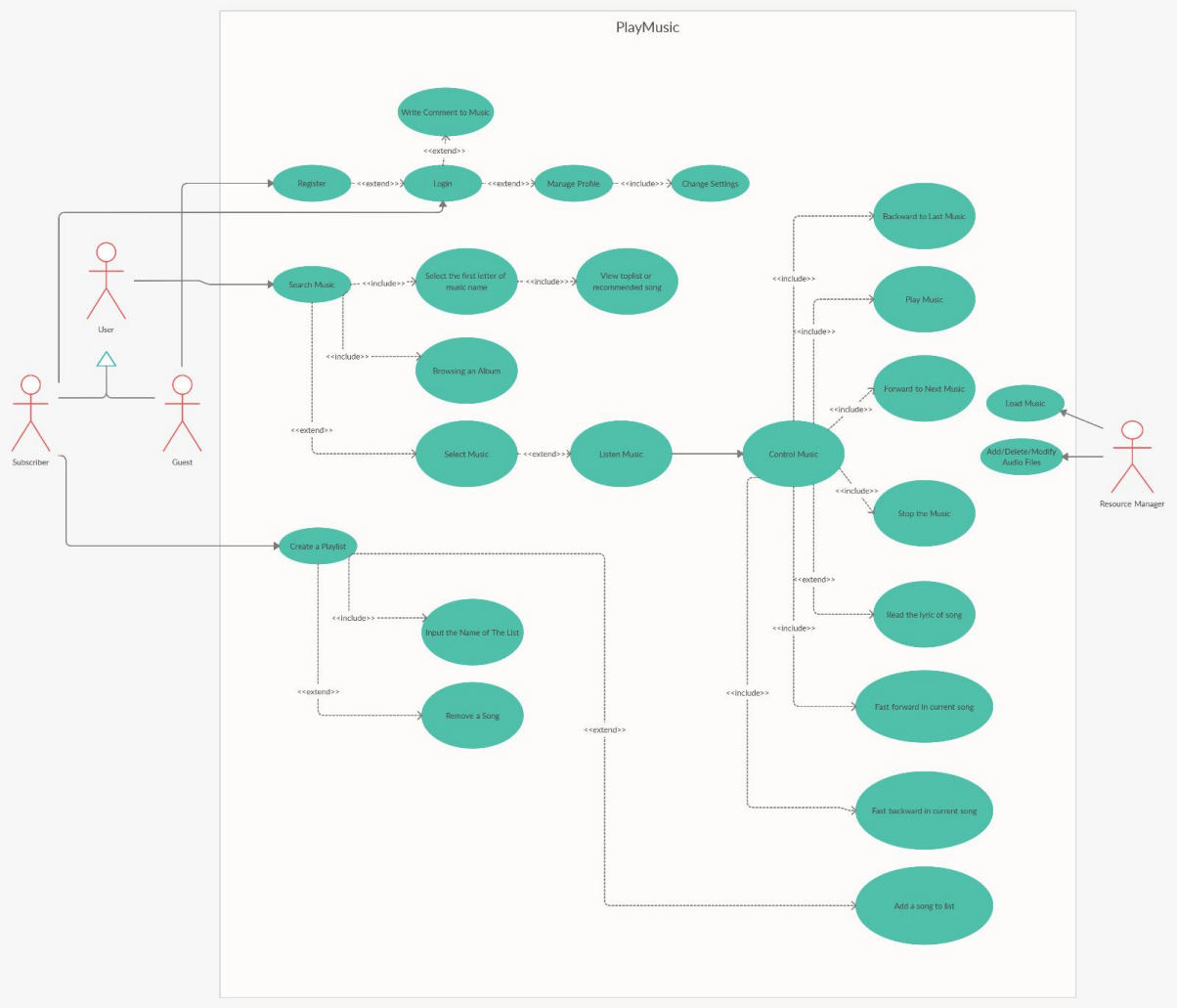
2_ System saves the movie options.

3_ User enters the application, and if is a Premium member can search any movie he/she wants and watch, create a playlist and download it. If user is a Guest, only can watch the free movie of the week.

Entry condition= ResourceManager logged in to upload the movie and video.

Exit Condition= User watched the movie.

Qualification Requirements= Guest can only watch voted free movie of the week.



Name: PlayMusic

Participating Actors: User (Subscriber – Guest), Resource Manager

Flow of Events:

1_ ResourceManager loads the audio file into the system by enabling the PlayMusic function. Then added lyrics.

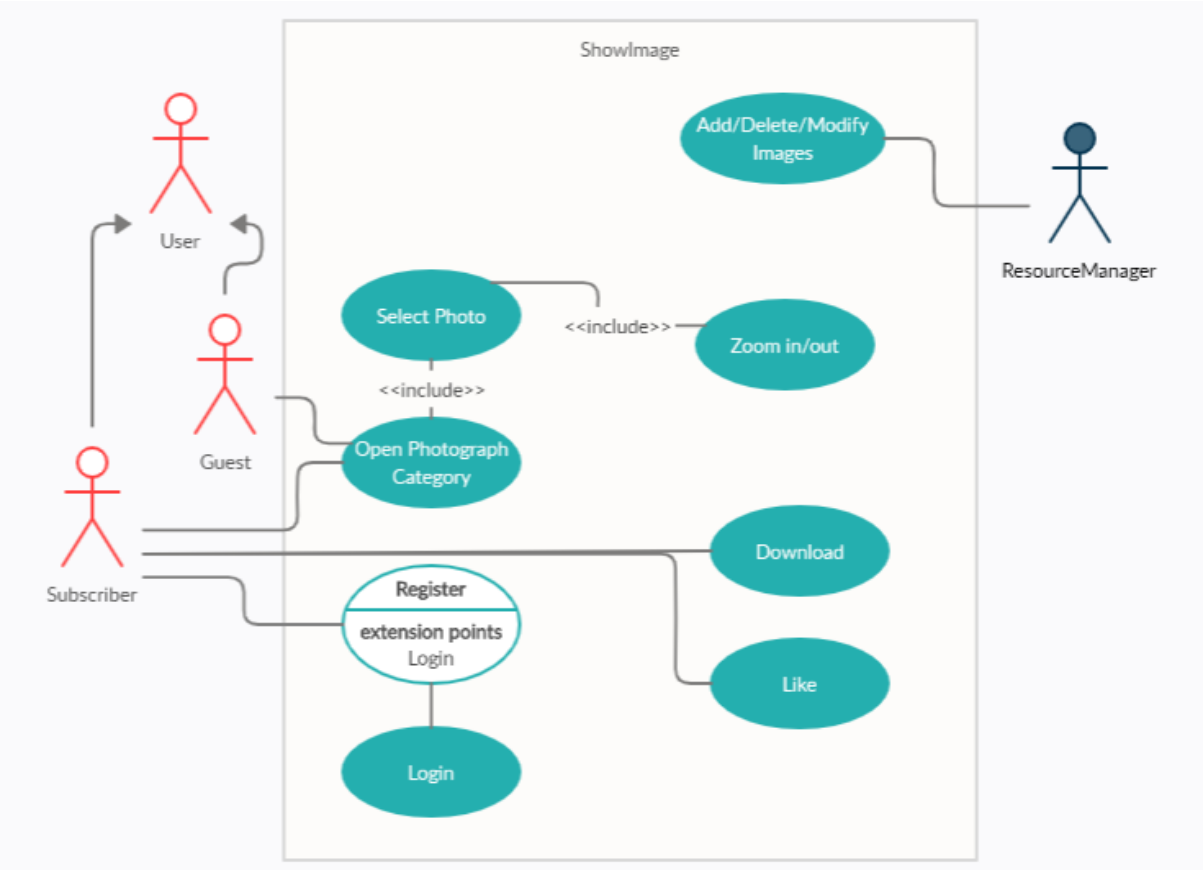
2_ System saves the lyrics options.

3_ User enters the application, and if is a Premium member can search any music he/she wants and listen, create a playlist and download it. If user is a Guest, must buy Premium membership.

Entry condition= ResourceManager logged in to upload the audio file.

Exit Condition= User listened the music.

Qualification Requirements= Guest must be a Premium member to benefit from the music section.



Name: ShowImage

Participating Actors: User (Subscriber – Guest), Resource Manager

Flow of Events:

1_ ResourceManager loads the image file into the system by enabling the ShowImage function.

2_ System saves the images.

3_ User enters the application, and if is a Premium member can download and like any music he/she wants. If user is a Guest, only can view image.

Entry condition= ResourceManager logged in to upload the image file.

Exit Condition= User viewed the image.

Qualification Requirements= Guest must be a Premium member to download and like the image.

3.4.3 Object Model

<Object model section documents in detail all the objects we identified, their attributes, and, operations. As each object is described with textual definitions, relationships among objects are illustrated with class diagrams.>

Step 5 activity

3.4.4 Dynamic Models

<Dynamic models section documents the behavior of the object model in terms of state machine diagrams and sequence diagrams. Although this information is redundant with the use case model, dynamic models enable us to represent more precisely complex behaviors, including use cases involving many actors.>

Step 5 activity

3.4.5 User Interface Mock-ups

<Mock-ups illustrating the user interface of the system and navigational paths representing the sequence of screens.>

Step 4 activity

4 Glossary

- ✓ **Movie:** The video file uploaded by the resource manager and the user can watch.
- ✓ **Addmovie:** The function that Source Manager uploads the video file to the system.
- ✓ **Language:** Language of the program.
- ✓ **Subtitle:** Text file that displayed at the bottom of movies that translate or transcribe the dialogue or narrative.
- ✓ **Quality:** Resolution quality of the movie.
- ✓ **Playbackspeed**: Making movie and video faster or slower.
- ✓ **Vote:** Choosing the most wanted movie.
- ✓ **Premium member:** Paying for the application
- ✓ **Guest user:** Entering without paying, limited user.
- ✓ **System:** The place where the Resource Manager process.
- ✓ **ResourceManager:** Admin of the application.
- ✓ **Result:** The most voted movie.
- ✓ **Public**: Content that can be reachable from everyone.
- ✓ **Free:** Membership system where the guest user continues without signing up.
- ✓ **Playlist:** Creable list for movies or music.
- ✓ **Audio file:** File format containing musics and movie soundtracks
- ✓ **PlayMusic:** The function by which the user can access and control the audio file.
- ✓ **Lyrics:** The words of a song that act simultaneously with the song.
- ✓ **Imagefile:** Viewable file format.
- ✓ **Download:** The user copying (data) from the application system to his/her own system.
- ✓ **VoteAddMovie:** The function for voting movies for free.
- ✓ **WatchMovie:** The function where the user views the video format file.
- ✓ **ShowImage:** The function where the user views the file in the image format.