

**CEN 4010 Principles of Software Engineering**

Summer 2020

**Milestone 1 Project Proposal and High-level description**

**Group 2 “The Fam” Evaluation:**

-Oscar Aquino (25%)

-Ryan Bates (25%)

-Elizabeth Garcia (25%)

-Jesse Kelly (25%)



A website dedicated to the establishment of quality choices in what to watch established by the consumers, and how to find them.

Revision 1.7 6/16/2020

**TERM PROJECT PROPOSAL – “What 2 Watch”**

Oscar Aquino - oaquino2017@fau.edu

Elizabeth Garcia - elizabethgar2017@fau.edu

Ryan Bates – batesr2013@fau.edu

Jesse Kelly – jkelly2019@fau.edu

**EXECUTIVE SUMMARY**

Do you often find yourself wanting to watch a movie, and then scrolling through the feed only to be discouraged after failing to identify with any of the titles? “What 2 Watch” will be launched with an effort to organize your options into an environment that provides you with a list of options that will resonate with you. One of the key advantages this platform will provide is the ability to have genre related discussions which ultimately should produce a large quantity of suggestions within the comment section alone. This will then create the opportunity for our users to copy and paste whatever might sound interesting to them in the search bar.

The user will have the option of saving any movies that spark their interest into a list of favorites. Being able to create this list will eliminate the annoying task of trying to store a friend’s suggestion in your head for some odd portion of the day. One of our main goals is to provide a comfortable platform where the user can keep their portfolio of favorites all in one place, but we also want to create an environment that ensures a quick solution to the common problem of taking too long when deciding what to watch. Our user interface will include an easy to use navigation system that will allow for quick browsing and simple solutions.

We want to encourage our users to enjoy the process of deciding what to watch, so in an effort to include the folks that like to take their time browsing, we will offer a section that allows the user to flip through a list of suggested titles two at a time. This flip section can be filtered as generally as sorting through all available titles, or as specifically as flipping through a genre that aligns with the user’s preference. Underneath each title we will present the user with the movie cover as well as a short description of the film.

Overall, what we value the most is how our users will enjoy their experience on our platform. This inspires our team to create something that everyone can relate with. Ultimately, it seems not enough thought is put into a plan of attack when it comes to deciding on what to watch, so we’ve decided to do the thinking for you.

**COMPETITIVE ANALYSIS**

|  |  |  |  |
| --- | --- | --- | --- |
| Features | What 2 Watch | Competitor 1 | Competitor 2 |
| User-friendly |  |  |  |
| Effective navigation |  |  |  |
| Web compatibility |  |  |  |
| Social networking |  |  |  |
| Rating system |  |  |  |

What 2 Watch offers a user-friendly platform designed for people to organize their watch list. Our product makes it simpler to search and find names of movies of your liking. Unlike our competitor, we will provide you with the ability to join to discussions related to the genre of movies you are interested in and give you the opportunity to share different titles with the community on the discussion. The rating system will rank the movies with a rating in numbers and not only “good or bad” as our competitors, making our product unique.

**DATA DEFINITION**

User data:

* Profile name
* User chat history
* User credentials
* User reviews
* User favorites list

Movie search:

* Movie tittle

Json or RSSFeed Entity:

* Movie tittle
* Genre
* Year
* Synopsis
* Store link (this is where they are re-directed to a purchase point)
* API KEY
* Final API URL

Entity Class – Movie:

* Movie tittle
* Genre
* Year
* Synopsis
* Reviews
* Rating

**OVERVIEW, SCENARIOS, USE CASES**

The user through a welcome page, will be greeted to log in or register, then they will type in their user credentials from which the database of our choice will authenticate and register/login the user. It is important to note that we may add social media login integration (Facebook or google) /email verification as an extra means of protection and accessibility.

The user will land at the discussion and search home page, if the user wishes to edit their profile, they can do so by going to the profile portal. The user will have a profile portal which stores their profile name, chat history, ability to change their credentials, and their most recent reviews. It is important to note that the chat history is different than reviews, chat focuses in the discussion of a genre while review is their one and only chance to leave a meaningful review for a specific movie. They will also be able to see their movie list, there they can see movies they favorited. The movies will be sorted by tittle but choices to sort through genre will also be presented by a filter tool icon. They may also delete specific movie entities from their list.

After logging in/registering the user will be redirected to the movie portal. There they can join in a discussion that is organized by genre or they can search for a movie of their choice using the search function.

User discussion will be divided by genre and users can join and have a general chat about their movie interest and share interests. Each post will have its specific thread (post page) that will allow for comments.

While inside the movie portal the user can search for any movie they wish, they will be redirected to a specific page where they will have a chance to see the JSON/RSSfeed entities associated with that specific movie. EG: Tittle, synopsis, where to buy, reviews, and rating. An additional button will be provided to add or delete the movie tittle from their list. The user will also be able to interact the movie entity and alter their ratings and they may also leave a review along side their rating submission.

**INITIAL LIST OF HIGH-LEVEL FUNCTIONAL REQUIREMENTS**

1. Search - A user shall be able to search the list of movies by title or genre.
2. Register - Each registered user shall be uniquely identified by his/her username and password
3. List - Each registered user shall have the ability to create and save their list of favorite titles.
4. Comment - The system shall allow the user to post comment sections under the movie genre.
5. Rate - The website shall be able to collect the users rating of each movie title.

**LIST OF NON-FUNCTIONAL REQUIREMENTS**

1. Performance – Response Time, Throughput, Utilization, Static Volumetric
2. Capacity
3. Reliability
4. Maintainability
5. Serviceability
6. Security
7. Usability

**HIGH-LEVEL SYSTEM ARCHITECTURE**

**PLATFORMS**  
-Google Chrome  
-Internet Explorer

Development will focus on implementation to the above listed platforms primarily because one comes standard with one of the most widely used operating systems on the market, and the other is the most widely used browser on the market.  
  
**DEVELOPMENT LANGUAGES**  
-html <https://bootstrap4.com/hotflix-online-movies-tv-shows-cinema-html-template/>  
-CSS <https://bootstrap4.com/hotflix-online-movies-tv-shows-cinema-html-template/>  
-bootstrap <https://bootstrap4.com/hotflix-online-movies-tv-shows-cinema-html-template/>

Primary web development languages we will be using while designing the website. Html, CSS, and bootstrap will be designed and implemented in a means to give the site an appealing look that makes users want to use the website.  
  
**INTERFACING/DATABASE LANGUAGES**  
-php <https://www.php.net/license/index.php>  
-JavaScript <https://bootstrap4.com/hotflix-online-movies-tv-shows-cinema-html-template/>  
-firebase <https://firebase.google.com/terms>

Database languages will be implemented to give a secure means of connectivity. Internal Chat will be implemented allowing users to communicate their ideas regarding movies and shows. Firebase is the primary means by which we wish to use to accomplish the setup. If the interface proves to not allow full capability the aged, but well established php is a backup plan.



**TEAM**

Front End Team Leader/Scrum Master: Ryan Bates  
Back End Team Leader: Oscar Aquino  
Github/Trello Master: Elizabeth Garcia  
Product Owner: All Software Developed and owned by Group 2  
Software Developers: Oscar Aquino, Ryan Bates, Elizabeth Garcia, Jesse Kelly

**CHECKLIST**

TEAM COMMUNICATION: DONE  
TEAM MEETING: DONE  
FRONT END LEAD: DONE  
BACK END LEAD: DONE  
GITHUD MASTER: DONE  
FRAMEWORK READY: DONE  
TEAM SKILLS DEFINED: DONE  
TEAM LEAD CONFIRMED READY M1 SUBMISSION:DONE