ReeldIn
Making Movies Matter

Project proposal & Statement of Work

Jonathan Houge, Project Manager

Cooper Harris, Elizabeth Hong, Julius Ramirez, Farrah Rodriguez

Greg Chism, Advisor

Date: February 19th, 2024

Revision History Table		Template Date 1/12/24
Version	Summary of Changes	Date
V1	Project Title Page & Executive Summary for Solo Idea / Pitch	1/17/24
V2	Updated Title Page & Executive Summary, plus User/Market Research and Product Features for Group Pitch	1/24/24
V3	Full Rough Draft of Proposal – updated and added to previous sections (Title Page, Executive Summary, User/Market Research, Product Features) as well as filling out new sections (Project Timeline & Gantt Chart, Ethics, Approvals, and Appendix).	2/5/24
V4	Revision in response to Instructor and Advisor feedback as well as individual / group findings – updated executive summary, user/market research (empathy interviews & survey), feature 1, feature 4 title, feature 6, feature 8, and ethics.	2/12/24
V5	Revision in response to instructor, advisor, and peer feedback - updated name ('Meticulous Moving Matching' to 'ReeldIn'), executive summary, user/market research (competition and survey), ethics table, and renewed signatures.	2/19/24

Contents

1. Executive Summary

This section was written by Jonathan Houge.

Movies are a highly valued creative outlet and pastime, but frequently, it's hard to choose one. With millions available and an abundance releasing daily, the hurdle of picking something to watch can be a hassle. *ReeldIn* intends to 'match' a user with films. Our product is intended to be a sleek, quick, and efficient web-based application that matches user inputs with a roster of films, offering a new way of handling indecisiveness and giant catalogs.

Overchoice (choice overload) **is a real problem**. Sheena Iyengar and Mark Lepper's Jam Study (2000) proved that out of two groups, one given twenty-four options and one given six, "**people who saw the large display were one-tenth as likely to buy as people who saw the small display**." [1] There are plenty of movie websites like IMDb and Letterboxd, but their recommendation systems are never ending and other options feel lackluster . While there is market competition, it's clear that users aren't satisfied with what's available today. Our application intends to offer more options and more flexibility through questions that will query our database, giving a more accurate & tailored result to the user in a satisfying way to curb choice overload.

The process of development is laid out below, in Table 1.

Stage Title	Details	End Product
Planning	Structure, Organization, Features, Mockups, Storyboarding	Roadmap
Pre-Development	Start of backend (django), frontend (HTML/CSS/JS), database	Foundation
Initial Development	Flesh out each part, more focus on frontend, dummy data for database pop.	Prototype (Alpha)
Evaluation (1)	Surveying & interviewing with application	Updated Requirements
Main Development	API usage & database population, algorithm creation, etc.	Live Website (Beta)
Evaluation (2)	Further surveying & interviewing with updated application	List of Final Tasks
Final Stretch	Bug Fixes, UI Tune Ups, etc.	Final product

Table 1 Process of Development

[1] More Isn't Always Better by Barry Schwartz, 2006 (https://hbr.org/2006/06/more-isnt-always-better)

2. User/Market research.

Overall Market

This section was written by Farrah Rodriguez.

Watching movies at home transformed when streaming services hit the scene. Today various streaming platforms like Netflix, Hulu, and Disney+ dominate the landscape and allow film buffs to enjoy their favorites in the comfort of their own home. Dedicated movie searching and reviewing platforms such as IMDb and Letterboxd play a crucial role in helping users find movies to watch. Although these sites were growing before, the COVID-19 pandemic brought them a surge of users as movie theaters shut down across the country. Letterboxd saw a significant increase in users going from 1.8 million in March 2020, to 4.1 at the end of 2021 to over 10 million in September 2023. [2] As of December 2023, IMDb reported having almost 8 million user reviews and hosting over 200 million unique monthly visitors. [3] [4] While theaters are open again, users continue to utilize streaming services and these websites.

Competition

IMDB & Letterboxd and pickamovieforme written by Jonathan Houge.
Likewise and Matched written by Elizabeth Hong.
Filmillion written by Julius Ramirez.

IMDB & *Letterboxd* - Both websites encourage user rating and community interactions between fellow movie fans. Yet, they also have recommendations based on related films, users and staff can craft lists with themes, and one can filter the never-ending pile of movies via filters. Letterboxd is the more contemporary of the two, being newer and more geared towards 'movie social media'. It has a more vibrant, intertwined community with the encouragement of logging movies by date watched, reviewing every time a watch occurs, and an interface promoting others' activity on the platform. IMDb has been around since the 90s' and, as a result, is mainly just a place to read about a movie or tv show that one has heard of with an easy-to-navigate, albeit unexciting, user interface. Our product will be different from both services by being streamlined. Our recommendations will be our primary focus, avoiding being bogged down by the operations of other aspects (i.e. the main reason these other sites are used). By only caring about movie recommendations, our application will be a one-stop-shop for getting a movie recommendation and getting out.

'*pickamovieforme.com*' - A website with a concept similar to ours, the 'Movie Recommendation Engine' asks six questions including - "How are you today?", "What comes closest to your occasion?", "Please choose any genre you're interested in", and "How old would you like the movie to be?". These are obvious questions, yet their answers leave something to be desired. The 'age of film' question, for example, only offers films up to the last twenty years. Plus, the site seems to only query from a list of 760 movies. There are also odd sections for "top movies by" for actors and genres and articles like top lists and reviews. There's an official post on Facebook (the place where it seems the most popular) that seems to suggest that the intention of this site is to be a place to stop wasting time and just get to watching. This is a great example of what *ReeldIn* shouldn't fall into - offering

[2] Letterboxd, Online Haven for Film Nerds, Gets a New Owner by Benjamin Mullin, 2023 (https://www.nytimes.com/2023/09/29/business/media/letterboxd-new-owner.html)
[3] Press Room Stats, IMDb, 2023 (https://www.imdb.com/pressroom/stats/)
[4] IMDb LinkedIn Description (https://www.linkedin.com/company/imdb-com/)

features with minimal functionality while other websites do it better and have already carved a significant market share.

Matched - This application is self-described as a streaming guide for couples. The application's interface design is based on the Tinder app. The purpose of this application works as one person would swipe through the movies generated and they would select for the movie to be swiped to the right for a "yes" and swipe left for a "no". Then, the other person would go through the process of selecting movies into yes/no categories until there was a match on the same movie provided by both people. This format generally works as taking the hassle of picking a movie that both people would like to watch. *ReeldIn* is an application that is similar, but is still significantly different.

Likewise - Available in both app and website form, the application's purpose is to combine recommendations for TV shows, movies, books, and podcasts, to give personal recommendations using AI (artificial intelligence). The AI is called Pix and the recommendations are not limited, you can ask Pix "more" and Pix will provide you more shows or movies based on the information that you provide on what you are interested in looking for. On Likewise if the user doesn't know how to start a conversation with Pix, they can choose one of the sample options provided like, "Hey Pix! How about a rom-com on Hulu or Netflix that's not too cheesy?", and the user can see all of what is available including: the summary, ratings, genres, and more. *Likewise* is similar to *ReeldIn*, but the primary difference is that the search and recommendations are AI powered.

Filmillion - *Filmillion* is both an app and website developed in 2012 that asks the user questions in order to either guess the movie a user is thinking about or generate a list of recommendations based on the answers. The app only features the guessing game while the website has both modes. Both the app and site do not appear to be maintained anymore, with the last iOS update being over 2 years ago and the last social media post for *Filmillion* being made in October 2018. The dynamic movie recommendation feature is similar to ours in that it relies on responses from the user, however Filmillion generates a generalized list. As a result of this generalized nature, or perhaps due to ineffective recommendation algorithms, some of the recommended movies go directly against some of the responses provided to filmillion. There is also the option to rate a particular movie however it seems to neither be stored anywhere on the site nor taken into account during the recommendation process. *ReeldIn* offers a more personalized experience that can take into account a user's profile while also tailoring recommendations based on the user's current attitude/desires, making *ReeldIn* the perfect companion for any movie finding situation.

Empathy Interviews
This section features interviews conducted by Cooper Harris, Jonathan Houge, Julius Ramirez, and Elizabeth Hong.

1. Autumn Harris (Interviewed by Cooper Harris - Relationship: Parent)

Key Insights: Autumn's movie choices are heavily influenced by her mood and the company she's with. She values recommendations from family and co-workers. Unexpected violence, excessive language, and a lack of plot in a movie can lead to dissatisfaction.

Pain Points: Difficulty in finding suitable movies, especially when searching for a specific mood or genre. Unpleasant surprises in content can result in a negative viewing experience.

Product Solution: The product can offer personalized movie recommendations based on mood and genre preferences. It can also provide detailed content information, such as violence levels, language, and plot complexity.

2. Caleb Harris (Interviewed by Cooper Harris - Relationship: Sibling)

Key Insights: Caleb relies on mood, recommendations from TikTok, and a prepared list of movie recommendations from friends. He values humor but is open to various genres. High expectations for humor can lead to disappointment if not met.

Pain Points: The challenge of finding movies that match his mood quickly. Disliking a movie when it falls short of humor expectations.

Product Solution: Implement features that offer quick access to mood-based movie suggestions. Additionally, the product can provide information on the humor level in movies, managing expectations and reducing the likelihood of disappointment.

3. Parker Harris (Interviewed by Cooper Harris - Relationship: Sibling)

Key Insights: Parker's movie choices are influenced by mood, and he prefers comedy, action, thriller, and possibly drama. He values good plot, pacing, and acting. Ratings from TikTok and online platforms impact his decision-making.

Pain Points: Dislike for overly lengthy movies, reliance on ratings for decision-making.

Product Solution: The product can include filters for movie duration, allowing users to find movies that match their preferred time commitment. Additionally, it can integrate TikTok and other platform ratings for a comprehensive evaluation of a movie's quality.

4. Rebecca Myrick (Interview by Jonathan Houge - Relationship: Parent)

Key Insights: Rebecca usually picks movies that she seems more predisposed to enjoy, as well as being influenced by genre, cast, and novel adaptations where she enjoyed the book. She'll use social media such as Instagram and Pinterest as well as word of mouth. She's open to all genres and has a watchlist she's usually watching from rather than adding to, spending time watching unseen movies rather than rewatching.

Pain Points: Rebecca doesn't like current blockbuster films, mainly ones overly reliant on CGI. She usually knows she's not enjoying a film if she's thinking about doing other things (i.e. wasting her time).

Product Solution: Questions about book adaptations and possibly, in addition to adding movie watching information from external sites like Letterboxd and IMDb, adding GoodReads information to find book adaptations. A more extensive "have you watched this?" checklist when it comes to watching a recommendation, like scale questions for things like "how captivated did you find the movie?"

5. Russell Griffin (Interview by Julius Ramirez - Relationship: Sibling)

Key Insights: Russell relies almost entirely on the streaming platforms algorithm but occasionally will look for a movie if he has seen an interesting clip of it from TikTok. He does not usually spend much time searching for a movie, only around 5 minutes, however if the movie he picks doesn't hold his interest then he'll do some other task instead of looking for another movie. He said that he often looks at the actors (and whether or not he likes them) first but ultimately chooses the movie based on the poster/preview.

Pain Points: Russell doesn't like when a movie is slow paced or visually unappealing and doesn't particularly like watching movies he's already seen. He says a common frustrating aspect is searching for the platform a movie is on, only to realize that the movie isn't available/on a platform he has.

Product Solution: Our product can filter movies based on platforms the user has access to, along with providing the ability to rotate through different options for a particular movie's poster/preview so that the user can get a better understanding of the film.

6. Darren Yuen Qai Hng (Interview by Elizabeth Hong - Relationship: Classmate)

Key Insights: Darren uses a mix of Netflix, Disney Plus, and Hulu to find movies. He will watch a movie when he has free time and isn't picky about the kind of movie he wants to watch. Since he isn't too picky about the movies he likes to watch, he can find a movie from these streaming sites in 5 minutes, but that is because the streaming sites already have a large selection of movies for him to pick from.

Pain Points: Darren doesn't think very highly about their recommendations being helpful and instead would rather browse specific content from other sources on his own.

Product Solution: Our product can provide the user with personalized recommendations that make the process of finding something to watch is much easier for the user. Reducing the amount of time it takes to locate a movie to watch and help the user find good recommendations without having to search the internet for them instead.

Survey

This section was written by Farrah Rodriguez.

Survey: https://uarizona.co1.qualtrics.com/jfe/form/SV_9AKOKAocmlzmBGS
Through Qualtrics, our team surveyed 23 people about their movie watching habits and their experience with movie recommendations from websites and streaming services. Most of these people were students from the University of Arizona or other Universities around the country, sourced from Discord and other group chats. Some were post-college professionals and two others were the parents of a group member.

This bar chart shows the age range of the participants.

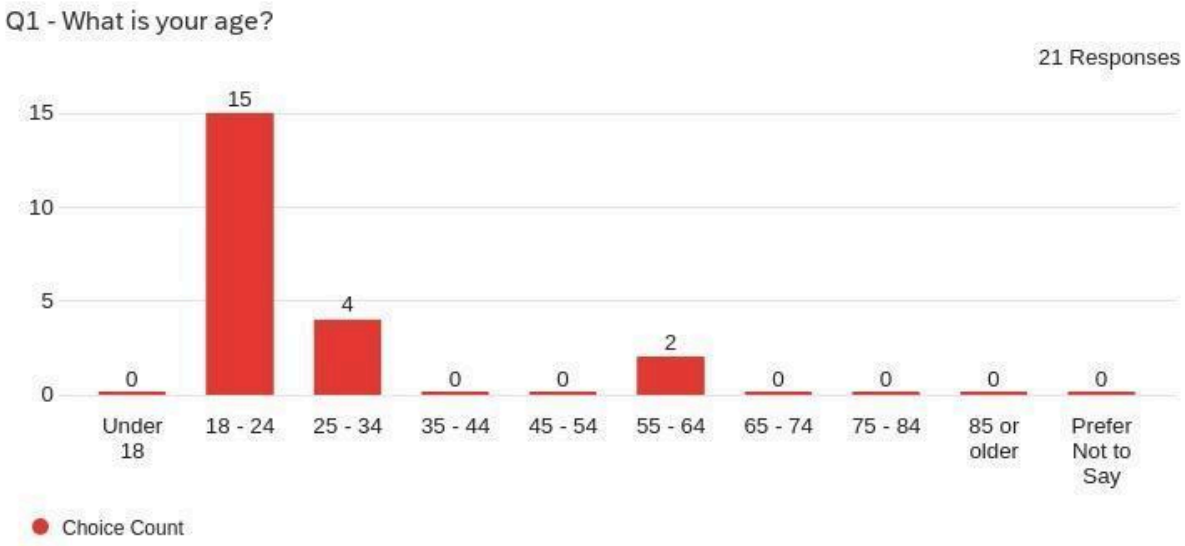


Figure 1: Age of Respondents

This pie chart displays the gender split in participants.

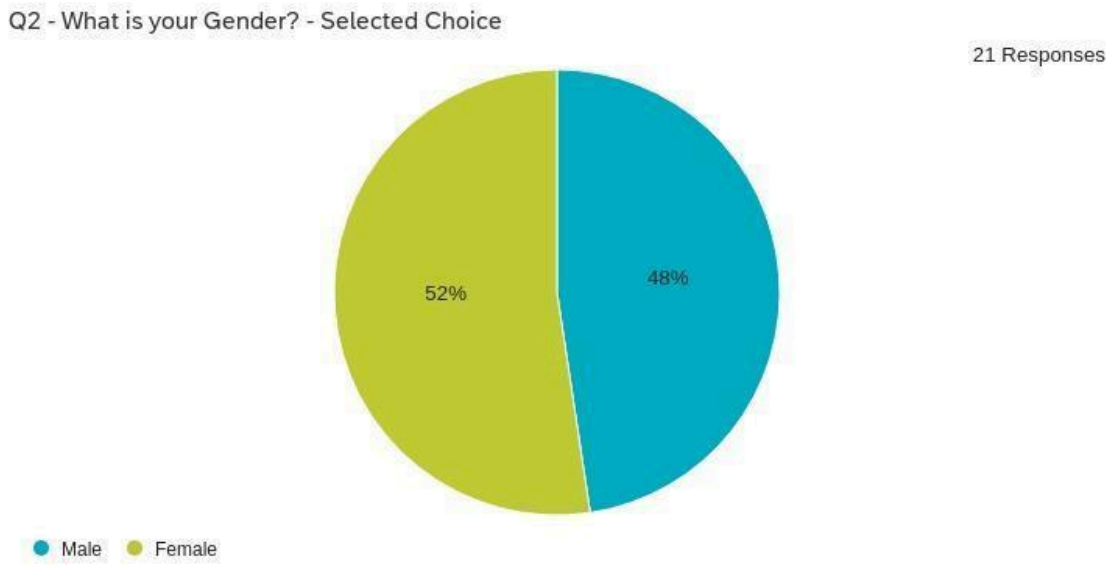


Figure 2: Gender of Respondents

Participants were also asked about existing services for finding movies. This table shows what services respondents use, don't use, and have never heard of.

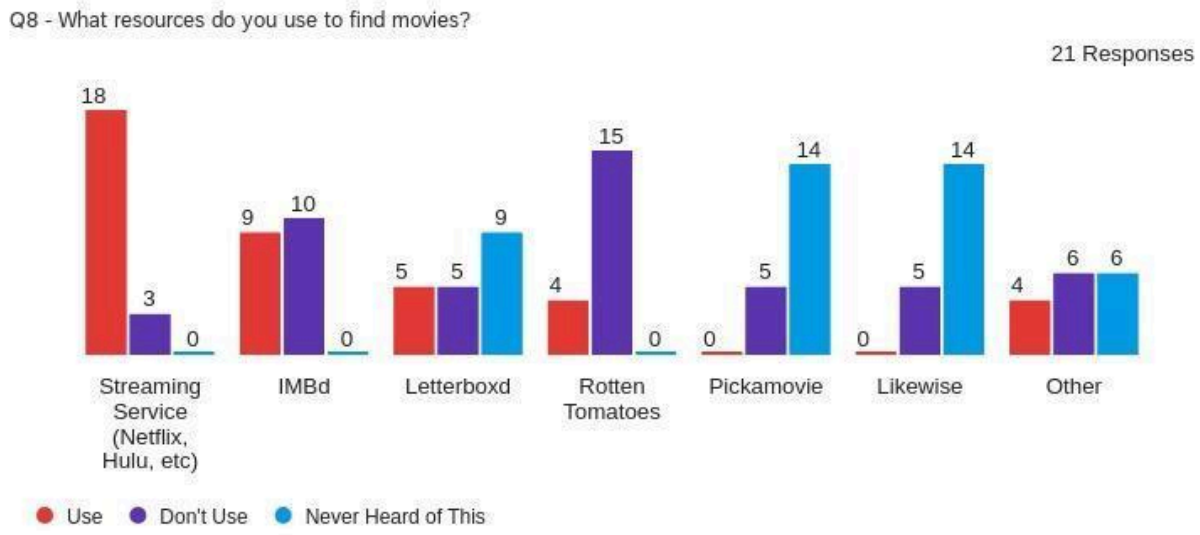


Figure 3: Knowledge of Existing Services for Finding Movies

Some significant results of our survey are shown in this table.

Question	Percentage	Frequency
Respondents who often to very often had trouble finding a movie to watch	55%	12/22
Respondents who rarely to very rarely watched recommended movies	44%	8/18
Respondents who remained neutral on watching recommended movies	28%	5/18
Respondents who rarely to very rarely relied on recommendations from websites or streaming services to pick a movie	73%	16/22
Respondents who rarely to very rarely felt recommendations matched their interests	16%	3/18
Respondents who remained neutral on whether recommendations matched their interests	39%	7/18

Table 2: Summary of Results

This table shows how many participants had trouble or did not not have trouble finding a movie and what services they use. Had trouble = often to very often had trouble. Did not have trouble = rarely to very rarely had trouble. Neutral responses have been excluded.

	Use Streaming	Use IMDb	Use Letterboxd	Use Rotten Tomatoes	Use Other	Use Nothing
Had trouble finding a movie	10	6	2	4	3	1
Did not have trouble finding a movie	6	3	3	0	0	0

Table 3: Cross Tabulation Table – Finding Movies x Service Used

This data reveals several insights that support the need for our website.

1. Difficulty finding movies to watch
    a. 55% of participants often have trouble finding a movie to watch. Even respondents who use services had trouble finding movies with 10 who use streaming services also saying they have trouble finding a movie. This suggests that there may be challenges associated with choosing a movie to watch, potentially due to overchoice or difficulty finding content that matches their interests.
    b. Our site will eliminate overchoice by initially providing a limited amount of recommendations. Only after the user prompts for more will they receive more.
2. Reluctance to watch movies recommended through existing services
    a. 44% of respondents reported that they rarely watched movies recommended to them by services. This shows dissatisfaction in their provided recommendations.
    b. We will increase satisfaction in recommendations by providing multiple avenues to receive recommendations, through a series of questions and answers, imported data from the user, user's likes and dislikes (on our site), and friend's liked movies.
3. Limited reliance on Recommendations from Services
    a. A significant portion of respondents, 73%, said that they rarely relied on recommendations from services. However, 19 out of 23 or 83% of respondents reported using at least one listed service to find movies. This indicates a lack of trust in these services recommendations.
    b. To increase trust in our recommendations we will give users a series of questions which will result in a series of recommendations based on their given answers. This makes them a part of the process and partially places the responsibility of the recommendations on the user.
4. Mismatch in Recommendations
    a. A small portion of respondents, 16%, reported that recommendations from services rarely to very rarely matched their interests while 39% remained neutral. This indicates a potential disconnect between recommended movies and personal preferences.
    b. To ensure we provide movie recommendations that match user interests we will use all methods previously mentioned in insight 2b as well as give them the ability to search through their recommendations using filters. This will allow them to narrow down their recommendations to specific genres, directors, producers, etc.

3. Product Features

Feature 1: Django Backend.

This section was written by Jonathan Houge and Cooper Harris.

Our backend will be made with Django, an open-source option. The backend will be made up of models for User, Movie, and Movie-Thoughts, as well as scripts to manipulate them based on frontend requests and behind-the-scenes workings.

This table shows the parameters of the backend feature which will be built using Django:

Models	Fields	Comment
User Model	Username, Password, Recommendations, Letterboxd Account, IMDb Account, Friends, etc.	User's identifier, password for account, array of past recommendations, accounts for external applications, array of friended users, etc.
User Thoughts Model	Movie, Rating, Similar, etc.	Movie model id, "Disliked, Neutral, Liked", "Do you want something similar", etc.
Movie Model	Title, year, tagline, director, cast, streaming services, etc.	Self explanatory.

Table 4 Feature 1: ReeldIn Backend Models

Security Considerations:

1. Use HTTPS:
We will use HTTPS to encrypt data in transit. This is crucial for securing sensitive information transmitted between the client and the server.
2. Configure Django Settings Securely:
We will ensure that the Django settings are configured securely, including setting `DEBUG` to `False` in production, using a strong `SECRET_KEY`, and restricting the `ALLOWED_HOSTS` to only trusted domains.
3. Database Security:
We will apply the principle of least privilege to database users. Use strong passwords and limit database access to only necessary operations. Considering tools like Django's ORM to prevent SQL injection attacks.
4. Input Validation:
We will validate and sanitize all user inputs to prevent common vulnerabilities like Cross-Site Scripting (XSS) and Cross-Site Request Forgery (CSRF) using Django forms and serializers to handle input validation.
5. Authentication and Authorization:
We will implement strong authentication mechanisms using Django's built-in authentication system. Use secure password storage mechanisms (e.g., bcrypt) and implement multi-factor authentication if possible.
6. Security Middleware:
We will use Django's security middleware to enhance security. This includes protection against clickjacking (`X-Content-Type-Options`), content sniffing (`X-Content-Type-Options`), and cross-site scripting (`X-XSS-Protection`).

Feature 2: Movie Database Creation (using The Movie Database's API).

This section was written by Farrah Rodriguez and Julius Ramirez

The Movie Database, IMDB, and Reelgood provide APIs that allow us to obtain movie, cast, and crew related information. With this information guests can search for movies and recommendations can be given based on the data from the database.

This table shows how we can interact with The Movie Database specifically and what notable data we can get:

Possible Usages	Route	Notable parameters
Obtain movie details	https://api.themoviedb.org/3/movie/{movie_id}	Adult,belongs_to_collection, budget, genres[n].name/id, overview, original_language, popularity, poster_path, release_date, runtime, status, tagline, vote_average, vote_count
Obtain TMDB reviews for a movie	https://api.themoviedb.org/3/movie/{movie_id}/reviews	results[n].author_details.rating
Obtain methods to watch film (buy, rent, stream)	https://api.themoviedb.org/3/movie/{movie_id}/watch/providers	results[country].buy[n].provider_name, results[country].rate[n].provider_name, results[country].flatrate[n].provider_name
Obtain a person (cast/crew) details	https://api.themoviedb.org/3/person/{person_id}	adult, birthday, imdb_id, known_for_department, popularity, profile_path
Obtain a person's movie credits	https://api.themoviedb.org/3/person/{person_id}/movie_credits	cast[n], crew[n]

Table 5: TMDB API Parameters

Feature 3: HTML/CSS/Javascript Frontend.

This section was written by Elizabeth Hong and Farrah Rodriguez.

The Frontend of our site will be created with HTML, CSS, and JavaScript.

This table shows the pages we will have and their function as well as key elements.

Page	Function	Key Elements

Home Page	The Home Page will introduce users to the site and request that they sign up or login if they already have an account.
	
	This page will also show the users what movies are recommended, based on their preference.
		• Header
		• Sign In Button
		  • Popup
		• Title
		  • Paragraph
		  • Footer
		• Movies
		• Filter

User/Profile Page	The User's Page/Profile Page will allow the user to view movies they've rated and change their settings.They will also be able to upload data from other sites like IMDB.	• Upload Data Button
		• Settings
		• See Friends
		• List
		  • Liked
		  • Neutral
		  • Dislike

Table 6: Site Pages

Feature 4: User Experience (UX).

This section was written by Elizabeth Hong.

The features displayed for the user experience and user interface design that will allow the user to easily interact with the applications features. The primary focus of this feature is to provide the user with consistent, desirable, and useful interactions in order to improve the user experience.

Parameters	Min	Max	Comment
UX Consistency	n/a	n/a	The font, shape, and overall structure should be consistent throughout all of the application.
UX Desirable	n/a	n/a	The application's efficiency should allow the user to feel comfortable and connect easily with.
UX Usefulness	n/a	n/a	The application's visualization should help with moving around the application easily, like having symbols.

Table 7: Feature 4 Parameters

Feature 5: Inputs for Random Selection & Movie Selection Based on Them

This section was written by Jonathan Houge

Based on answers to questions, a database will be queried via SQL, grabbing a finite amount of films based on the results. The user answers questions on the frontend and then a get request is sent to the backend, the movies are filtered through, and the recommendations are sent back to the frontend, displayed for the user.

A table that goes more into detail follows.

Parameter	Min	Max	Comments:
Answers	10	30	Array of answers, parsed into singular words for SQL queries
Output Count	3	10	How many films to generate - if the user asks for more, increase to 6, then end with 10 (if you ask for more, message like "just pick one!")
User's Seen Films	0	1000 +	Currently logged in user's seen films, ensures that there are no duplicates

Table 8: Feature 5

Feature 6: Movie Recommendation Based on User's Data

This section was written by Farrah Rodriguez.

User's will be able to provide their own data from other websites such as IMDB for use on our website. This will allow users to gain more personalized recommendations on our site without having to manually input all of their previous information. This will provide greater ease-of-use for the user in their transition to our site. Relevant information includes: movie ratings, watchlist, and other lists created by the user.

User Steps:

1. Users will be given directions on how to access their data on other sites.
2. Once they have their CSV file, they will be instructed to upload it to our site.
3. Their information will be added to their profile and their recommendations will be updated.

Recommendations will be generated using **Content-Based Collaborative Filtering with Cosine Similarity**. Using movies that the user has already rated on the other site, we will be able to generate recommendations.

Developer Steps:

1. Clean and format data from User's CSV file
2. Populate the User's profile with watched movies and represent it as a feature vector
3. Create feature vectors using chosen movie attributes such as: genre, director, producer, plot, keywords, rating, etc.
4. Calculate the cosine similarity between movies and the user's profile
5. Filter out movies that the user has already rated/watched
6. Rank the movies based on their cosine similarity and recommend the top N to the user

Feature 7: Better Searches

This section was written by Cooper Harris.

The search feature is designed to enhance the user experience by providing fast and accurate results. Users will be able to filter by tag, description, and personal recommendations. Shown below are key aspects of the search function.

The following tables lists the parameters.

Parameter	Min	Max	Comments
Filters	0	5+	The number of possible filters may increase
Recommendations	0	n/a	The number of recommended movies when searching will vary by user
Tags	0	5+	The number of ways movies can be tagged may increase
Movie providers	0	n/a	Users may search for movies only offered by a certain service

Table 9: Feature 7

Feature 8: User Accounts

This section was written by Julius Ramirez.

ReeldIn allows users to create accounts that can store details about their movie preferences and watch history, allowing for quicker and more personalized recommendations. A user can have the following info saved:

The following table shows various user specific parameters,their internal representation, and usage comments:

Parameter	Internal representation / Security considerations	Usage & Comments
Username/Password	String	Usernames may consist of letters, numbers, and some symbols. Username length must be between 3-16 characters inclusive. No repeat usernames are allowed. Passwords will be salted and hashed and can contain letters, numbers, and some symbols. Password length must be between 8-32 characters inclusive and must include a mixture of symbols and numbers (minimum 2 of either).
Movies liked, neutral movies, movies disliked	List of integer movie ID's	3 separate lists. A rating is categorized as either: liked, neutral, or disliked. Re-rating a movie will remove it from the previous list (if necessary). Ratings from imported data will ultimately be placed in one of these lists based on various factors (i.e 0.0-4.0 = dislike, 4.1-6.0 = neutral, 6.1-10 = liked)
Genre's typically liked, genre's typically disliked, genres to not recommend	List of integer genre IDs	3 separate lists, used within recommendation algorithms. Asked during account creation but can be modified.
Movies to not recommend, movies to watch later	List of integer movie IDs	2 separate lists, used within recommendation algorithms. Movies to not recommend are asked during account creation but can be modified.
Friends	List of integer User IDs	Friends can be added so that their recently watched/rated movies can be recommended. Blocking is not a feature.
Cast/Crew typically liked, Cast/Crew typically disliked, Cast/crew to not recommend	List of integer actor IDs	3 lists, used within recommendation algorithms. Asked during account creation but can be modified.

Table 10: Feature 8

User accounts also require security measures in order to protect user data. The following table shows some security features and their applications:

Security Feature	Min	Max	Comments
2FA	Knowledge Factor (i.e “favorite movie?”)	Possession Factor (Google authenticator)	This can allow for more secure password resets/logins
Login limiting/lockout	10 minute timeout after 5 failed attempts	10 minute timeout after 5 failed attempts with each consecutive fail adding another hour of wait time capped at 3 hours.	Prevents brute-force attacks. Timer may be reset through login/2FA
Account Privacy	Invisible/visible to searches	Varying degrees of invisibility (i.e invisible, searchable only to friends of friends)	Users must be made aware that their recently watched movie data may be used to recommend movies to their friends
Data storage	Salted/hashed passwords	Encrypted storage(i.e AES-256 via python’s *cryptography* library)	Protects user data in the event of database compromise. Project manager can hold exclusive access to the salt key.

Table 11: Feature 9 Security Considerations

4. Project Timeline & Gantt Chart

Written by Elizabeth Hong and Jonathan Houge.

Milestone	Date
Team Formation	1/29/24
Pre-Signed Planning: Draft 1	1/29 - 2/5/24
Pre-Signed Planning: Draft 2	2/6 - 2/12/24
Pre-Signed Planning: Final Draft	2/13 - 2/19/24
Signed Proposal	2/19/2024
Planning	2/19 - 2/22/24
Foundation Development	2/23 - 3/1/24
SPRING BREAK	3/2 - 3/10/24
Foundation Development: Minor Fixes	3/11 - 3/13/24
Initial Development	3/14 - 3/21/24
Evaluation (1)	3/21 - 3/23/24
Main Development	3/24 - 4/16/24
Evaluation (2)	4/17 - 4/19/24
Final Stretch: final testing, bug fixes, final checks, etc.	4/20 - 4/23/24
Poster / Presentation Prep	4/20 - 4/23/24
Poster Demo	4/23/24
Final Stretch (2)	4/23 - 4/27/24
Project Due Date	4/28/24
Poster / Presentation Refinement	4/28 - 5/1/24
Poster Final / iShowcase	5/1/24

Table 12: Milestone Schedule

Gantt Chart:
https://docs.google.com/spreadsheets/d/12dYsDBrh014mkPpSKW76lXOGdSNOe644o9g5LHgb_G8/edit?usp=sharing

5. Ethics

This section was written by Julius Ramirez and Jonathan Houge.

#	Question	Generally	Data Breach	Mitigation / Comments
1	Could a user sell drugs or other illegal items on your platform?	N	N	
2	Could a user of your platform engage in sex trafficking?	N	N	
3	Could a user sell class notes or cheat on their homework on your platform?	N	N	
4	Could a stalker use your project to find someone?	N	N	
5	Could your app be used to spy on or track individuals?	N	N	
6	Could your app/software access the camera or microphone and record things without users being aware?	N	N	
7	If someone uses your platform, could they be re-traumatized or have their mental health impacted in some way?	M	N	A user could be recommended a troubling film. Genre/content filters selected by the user/system can minimize this from happening.
8	Could your algorithm promote material that would traumatize or upset individuals?	M	N	A user could be recommended a troubling film. Genre/content filters can minimize this from happening.
9	Would your users be upset if the data you collect was given to someone else?	M	M	Users may be upset if their email and friend info are given to others without their consent, however this will not be done, the only data being exchanged is the recently watched movies between friends, however the user will not have direct access to that info. In the event of a data breach, any info being given to others is likely to upset them. Database security measures will be taken into account (see section 3 feature 1 security considerations).
10	Could a data leak potentially lead to identity theft?	N	N	

11	If your site was hacked, would users of that product potentially lose their job, spouse, or family?	N	N	
12	Should there be an age limitation on your product?	M	N	Due to the possibility of exposure to mature content, a minimum age of 13+ might be best. This could be done through a simple 13+ age check during account creation that wouldn't be saved to the database.
13	Could someone use your product to find, contact, and potentially commit elder abuse?	N	N	
14	If the data on your platform was breached, could it be used to blackmail the users?	N	N	
15	Does the existence of your project imply that a particular racial group, gender, religion or other protected category is inherently bad, gross, or unwanted?	N	N	
16	Could your product be used to commit hate crimes against a specific group?	N	N	
17	Does the primary content of your game or algorithm focus on something considered deeply unethical?	N	N	
18	Does your game or software contain race, gender, or other stereotypes?	N	N	
19	Could users of your app scam other individuals?	N	N	
20	Is your particular algorithm biased towards predicting correctly only for one race, gender, or other group?	M	N	Users could feed questions that might favor films created by / geared to certain groups. To combat this, offensive films can be excluded from recommendations and/or the site altogether.
21	Are the users of your project, players of your game, or those being surveyed for your data aware of how their data will be used?	M	N	Survey initially didn't disclose data use but was updated. Everything else will too.
22	What are the possible misinterpretations of your results? For example - would a white supremacist or misogynist be stoked about your results if they misinterpreted it?	M	N	Result misinterpretations could appear in the form of a poor movie recommendation.
23	Does the use or purchase of your data potentially contribute to a dangerous group or regime?	N	N	
24	Could your virtual reality environment cause injury to the user?	N	N	

25	Are your study participants or game players aware that their data will be collected and used?	M	N	The survey we sent out initially didn't, but now does & beta testers will be notified too.
26	Does your game or app contain addictive design elements without benefit to the user?	N	N	
27	Does your survey contain an aspect of compulsion or unusually large incentive, that would command users to take it even if it was to their detriment?	N	N	
28	Could your research outcomes harm an individual or entity?	N	N	

Table 13: An ethics table outlining ethical concerns / solutions

1. Approvals

The signatures of the people below indicate an understanding of the purpose and content of this document by those signing it. By signing this document, you indicate that you approve of the proposed project outlined in this Statement of Work, the division of work, the Ground Rules and that the next steps may be taken to create a Product Specification and proceed with the project.

This document is based upon and supersedes V4. Deviations, (versus clarifications), from the PDR have been clearly noted. For any requirements not listed in this SOW, the PRD requirements shall remain in effect.

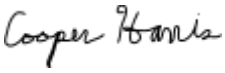
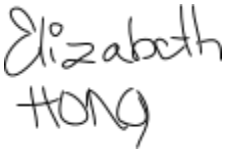
Approver Name	Title	Signature	Date
Jonathan Houge	Team Project Manager		2/18/24
Cooper Harris	Team Member		2-16-24
Elizabeth Hong	Team Member		2-16-24
Julius Ramirez	Team Member		2-16-24
Farrah Rodriguez	Team Member		2-17-24
Greg Chism	Advisor		
Michael McKisson	Instructor		

Table 14: Involved Parties Signatures

Section	Author	Word Count
Executive Summary & Table 1	Jonathan Houge	205 + N/A
User/Market Research: Overall Market	Farrah Rodriguez	147
User/Market Research: Competition	Jonathan Houge Elizabeth Hong Julius Ramirez	358 263 203
User/Market Research: Empathy Interviews	Cooper Harris Jonathan Houge Julius Ramirez Elizabeth Hong	302 169 194 172
User/Market Research: Survey & Figures 1-3 + Tables 2-3	Farrah Rodriguez	521 + N/A (x5)
Product Features: Feature 1 & Table 4	Cooper Harris Jonathan Houge	235 + N/A
Product Features: Feature 2 & Table 5	Julius Ramirez Farrah Rodriguez	58 + N/A
Product Features: Feature 3 & Table 6	Elizabeth Hong Farrah Rodriguez	29 + N/A
Product Features: Feature 4 & Table 7	Elizabeth Hong	47 + N/A
Product Features: Feature 5 & Table 8	Jonathan Houge	65 + N/A
Product Features: Feature 6	Farrah Rodriguez	225
Product Features: Feature 7 & Table 9	Cooper Harris	43 + N/A
Product Features: Feature 8 & Tables 10-11	Julius Ramirez	67 + N/A (x2)
Project Timeline	Elizabeth Hong Jonathan Houge	N/A
Gantt Chart	Farrah Rodriguez Julius Ramirez	N/A
Ethics	Julius Ramirez Jonathan Houge	N/A
Ground Rules Additions (including code template)	Cooper Harris Farrah Rodriguez Jonathan Houge	91 43 + N/A 41

Table 13: Author Table

2. Appendix

A. Advisor Engagement

1) Project Team Responsibilities

- The Project Manager will set up and facilitate a weekly call/meeting with the Faculty Advisor. The Project Team will provide weekly status updates to the Faculty Advisor including upcoming deliverables, critical issues, and any adjustments to the Project Plan.
- Documents will be provided to the Faculty Advisor with adequate time for review and signature. The time necessary for review will be agreed with the Advisor. The minimum review time will be 3 days prior to the document due date.
- Design files will be provided to the Faculty Advisor as requested in a format agreed to with the Advisor.
- Support requirements will be clearly requested from the Faculty Advisor with the dates required and an adequate time for fulfilling the request.
- Modifications requests to the Project Plan by Faculty Advisor will be reviewed and agreed to within 1 week of the request.

2) Faculty Advisor Responsibilities

- The Faculty Advisor will provide knowledge and expertise to help the group stretch their skills.
- The Faculty Advisor will participate in a weekly or bi-weekly call/meeting with the Project Team to review the project status, upcoming deliverables, priorities, issues, and progress to the agreed Project Plan.
- The Faculty Advisor will provide document review, feedback and approval, rejection, approval with contingencies with adequate time for the Project Team to meet the course due dates.
- The Faculty Advisor will provide feedback to requested support requirements from the Project Team. This includes feedback and guidance on design implementations decisions, design files, test plans, test procedures and test results.
- The Faculty Advisor shall provide technical advice and guidance to the Project Team answering inquiries approximately 1 hour per week.
- Modifications to the Project Plan by the Project Team will be resolved and documented within 1 week of the request.
- Grade the finalized project using a skill-based rubric
- Attend iShowcase in May.

B. Ground Rules

As a team and as individual team members, we agree to:

1. **Stay focused on our objectives and goals.**
   Each time the team meets, we will clearly define our objectives and desired outcomes at the beginning of the meeting. We will politely remind team members if we are getting off track.
2. **"Sidebar" any issues that are relevant but not consistent with the immediate objectives.**
   Occasionally, important matters are raised that are not relevant to the immediate goals of the meeting. To keep the group on track, but avoid losing the issue, create a "sidebar" where these topics can be listed and discussed later.
3. **Listen when others are speaking.**
   We will listen and consider others' input before adding our own comments.
4. **All viewpoints will have an opportunity to be heard.**
   We understand that some team members may be quieter than others. We will make an effort to get each team member's viewpoint and that no one dominates the discussion.
5. **Differences of opinion will be discussed respectfully**
   We will identify areas of agreement before assessing areas of disagreement. We will encourage each other to look beyond our own point of view. We will discuss different ideas respectfully. As a team, we will weigh the merits of different opinions and agree on a process for choosing a direction. All team members will respect and follow the decision or direction.
6. **Look for the good points in new ideas.**
   We will endeavor to explore the value in each idea as we assess and select our path forward.
7. **Focus on the future, not the past.**
   We will use our past experience to inform our decisions, but focus the discussion on the future objectives. Blame for past performance is counterproductive, we will focus on finding solutions.
8. **Agree upon specific action items and next steps.**
   At the end of each meeting and discussion, we will summarize and agree on specific next steps, action items and assignments.
9. **Accountability**
   As team members, we will each be responsible for our individual assignments and contribution to achieving the team objectives and goals. We will honor our responsibilities and not let our team members down.
10. **Punctuality**
    We will strive to start and end meetings on time to respect everyone's schedule. Being punctual demonstrates a commitment to the team and ensures that valuable time is used efficiently.
11. **Constructive Feedback**
    When providing feedback, we will focus on constructive criticism that helps improve ideas or processes. Avoiding negative language and personal attacks to create a positive and collaborative team environment.
12. **Flexibility**
    Recognize that plans may need adjustment based on evolving circumstances. Being flexible and adaptable allows the team to respond effectively to unexpected challenges and opportunities.

13. **Code Readability**

    We agree to make our code as clear as we can by including a comment for each function, class, and method as well as inline comments when necessary. Function comments should include a description of functionality as well as parameters and output.

    Template:

    /*

     * Author: Author's Name

     * Purpose: Description of use including any necessary information such as parameters and output.

     */

14. **Proper Git Etiquette**

    We agree to use source control in an effective and cooperative way. Pull requests will be used to merge new code into master, with others offering constructive feedback to ensure code is of the highest possible quality.