Capstone Project

REELATABLE

Discover Yourself Through Your Movies

May 2024

Team

Ankita Suresh Shanbhag

Saurabh Chachra

Hrishikesh Srinivas Nagaraju

Kinshuk Nigam

Faculty Advisor

Dr. Marti Hearst

Master of Information Management & Systems

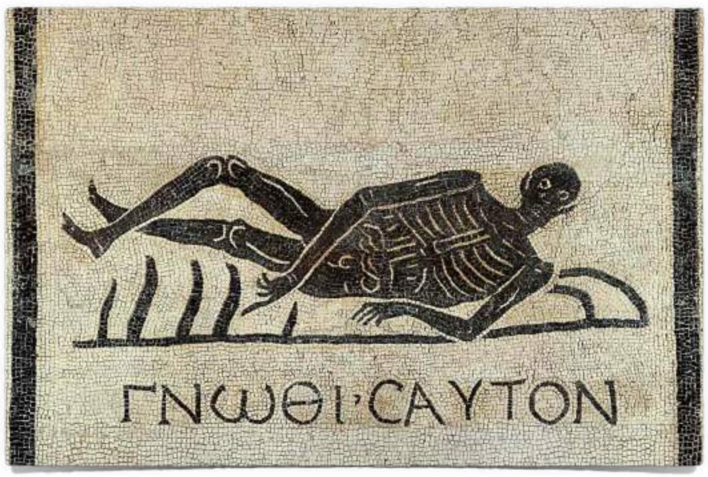
School of Information

University of California, Berkeley

Table of Contents

Purpose

Self-Understanding



The Greek motto gnōthi sauton (know thyself, nosce te ipsum)

Socrates believed that all philosophical commandments could be reduced to one idea: 'Know thyself.'

There is an extensive body of scholarly research underscoring the significance of self-understanding for psychological well-being and healthy functioning of individuals. Self-concept clarity is positively associated with self-esteem (Campbell, 1990). The findings from Lewandowski and Nardone (2012) suggest that higher self-concept clarity individuals may be at an advantage in developing relationships. Self-concept clarity may be beneficial in a variety of relationship situations and contexts (see Gurung et al., 2001). Self-awareness also contributes to better decision making and team performance (Dierdorff & Rubin 2015).

Lack of Systematic Methods to Improve Self-Understanding

However, in the quest for deeper self-understanding, there are significant challenges due to the lack of systematic methods available. Currently, the most accessible option is to engage with online psychological assessments, such as the Big 5 personality traits. These tools are advantageous due to their quickness, ease of access, affordability, and scalability. However, the complexity of human psychology, which encompasses thousands of psychological dimensions, adds to the difficulty, as it's not straightforward to identify which dimensions are most consequential for an individual. Hence, psychological tests fall short in their comprehensiveness. Consequently, while these online tools offer a starting point, they do not provide a thorough pathway to deeper self-understanding.

On the other hand, therapy offers a more comprehensive approach but is hindered by its high costs and lack of accessibility, making it an impractical option for the majority of the world's population. This dichotomy between the accessibility of online tests and the thoroughness of professional therapy presents a significant barrier in the field of psychological self-assessment.

Goal

Design and build a product that will help a large number of users understand themselves better. The product should be designed to be rapidly scalable with variable cost close to zero.

Theoretical Framework

How Self-Understanding Works

Self-understanding involves categorizing one's identity through various descriptors encapsulated by the question, "Who am I?" This process helps individuals develop a greater understanding of self-concept by examining personality traits, social roles, and existential affiliations (Schwartz et al. 2017). Therefore, enhancing self-understanding could significantly benefit from methodologies that support and refine the categorization of self into various descriptors under the "Who am I?" inquiry.

However, the vast variety of possible self-descriptors (e.g., idealist, optimist, compassionate, feminine, inquisitive), introduces two important questions:

1. Is it necessary for individuals to understand the scientific definitions of each descriptor to effectively categorize themselves? This approach seems neither efficient nor desirable.

2. How can we effectively narrow down these descriptors to the psychological dimensions most significant to different individuals? Identifying and focusing on key psychological traits that resonate personally can streamline the self-understanding process, making it more accessible and tailored to individual needs.

Categorization

Prototype Theory of Categorization

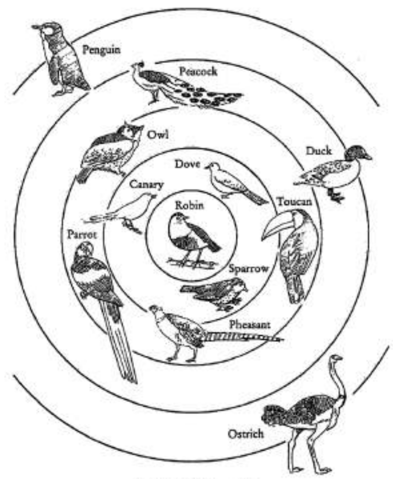


Figure 1 Birdiness rankings

The bird category, from Aitchison (2012: 69)

According to Rosch (1978), people rely less on abstract definitions of categories than on a comparison of the given object or experience with what they deem to be the object or experience best representing a category ("prototype"). Hence, it could be argued that the process of self-categorization can be better supported by providing prototypical examples of a category rather than its definition.

How can we provide prototypical members of the trait category that represent them?

Mar and Oatley (2008) suggested that “The function of fiction is the abstraction and simulation of social experience”. Black and Barns (2015) found that film narratives, as well as written narratives, may facilitate the understanding of others’ minds.Further, even before story writers start writing a story, they etch out the psychological characteristics of a character in detail (McKee, 2005). Therefore, fictional characters might serve as excellent prototypical examples of various psychological characteristics.

Significant Psychological Dimensions

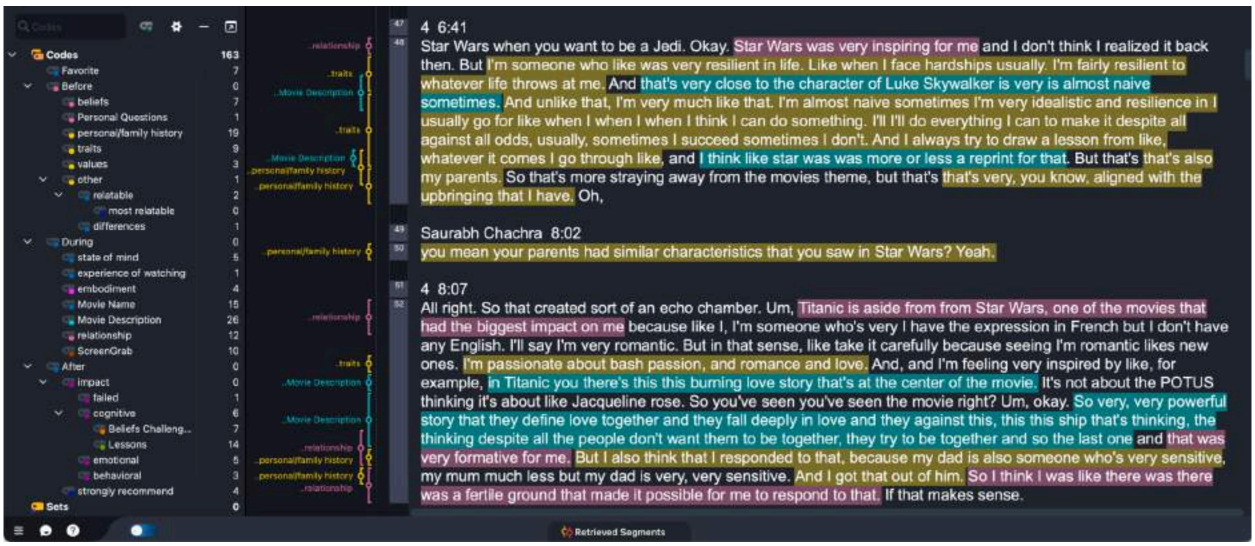
To answer the second question introduced earlier (“How can we effectively narrow down these descriptors to the psychological dimensions most significant to different individuals?”), we conducted three kinds of studies

1. Semi-structured interviews with movie enthusiasts

2. Structured interviews with movie enthusiasts
3. Literature review in storytelling

Relationship between Movies and the Psychology of Movie Enthusiasts

The purpose of this study was to understand the relationship between movies and the psychology of movie enthusiasts. We conducted semi-structured interviews with 8 movie enthusiastsWe asked the participants a variety of questions like, “How have movies helped you get through tough times or made sense of things happening in your life?”, “Can you recall a moment in a movie where you felt a personal connection or that it resonated with your own life experiences?”.



Qualitative Data Coding Using MAXQDA: Coded the transcripts of the interviews to reveal major themes and patterns.

However, one question that elicited the most interesting responses was *“Can you tell me about a movie that made an impact on you? It doesn't have to be a masterpiece, just any film that resonated with you personally.”* In their responses to this question, each of the participants invariably ended up talking extensively about themselves: their childhood experiences, personality dispositions, family histories and how the protagonists of these movies represent something deeply personal about them. We concluded that this question can help us narrow down to the psychological dimensions that are most significant to different individuals.

However, this raised another question: *Would we find the same characteristic in other movies that made an impact on the participants?* In other words, *Would a list of movies that made a deep impact would show a pattern?*

Patterns (Structured Interviews)

The primary research question here was whether a discernible pattern could be identified from a list of movies that have made a deep impact on an individual. To investigate this, we recruited 9 movie enthusiasts. To each participant, we first asked to list such films. Following this, participants were asked to reflect on each movie listed, articulating why they believed these films had left a significant impact on them. We consistently found psychological patterns in participants' movies.

For one participant, our analysis revealed that the theme of 'lost friendship' prominently figured in six out of their top eleven films. Upon further inquiry into why this theme recurrently surfaced in their favorite movies, the participant spontaneously articulated a personal narrative, revealing that the challenge of forming and maintaining friendships has been a significant struggle throughout their life. This method was applied consistently across all nine participants, allowing us to rapidly unearth profound insights.

Literature Review in Storytelling

There can be a large variety of psychological characteristics in a movie character like personality traits, quirks, values and beliefs, Inner conflicts, etc. The purpose of this study was to investigate what categories of psychological characteristics make characters and stories most relatable? Conducted a literature review of Story by Robert McKee, Save The Cat by Blake Snyder, and articles by StudioBinder blog.

The findings highlighted four key categories of characteristics that are determined by writers before writing a story, making their stories more engaging and relatable:

1. beliefs that guide characters' choices throughout the narrative;
2. emotional needs or desires that drive their actions;
3. character flaws or weaknesses that hinder their ability to fulfill their needs/desires; and
4. character strengths that enable them to overcome their flaws and fulfill their desires.

These insights were instrumental in shaping the design of various prompt elements for GPT, enhancing its ability to generate relatable and compelling content.

Cheaper, Efficient, and Scalable Way to Uncover Patterns

To make this process cheaper, efficient, and scalable we decided to work on three areas:

1. Automate the process of soliciting a list of movies. This could be achieved by developing an app.

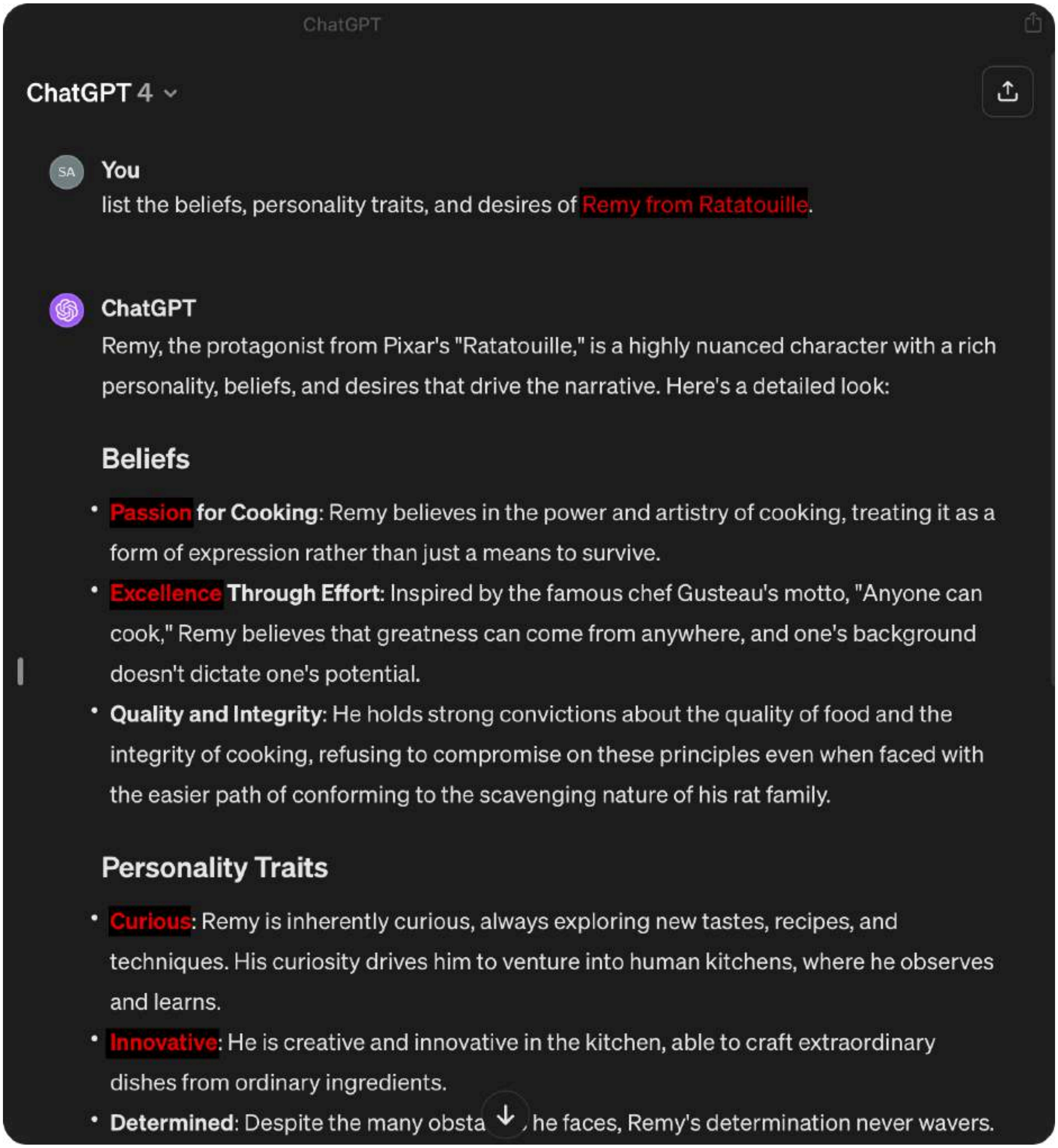
2. Instead of asking users for the reason why the movie impacted, we could build a database of psychological characteristics in the movies. Allowing users to pick characteristics that resonated with them, instead of asking them to reflect, would reduce cognitive load on the user.

3. Finally, automate the process of assessing the patterns

Database of Psychological Characteristics in the Movies

To build a database of psychological characteristics in movies, we considered two main options. The first involved crowdsourcing: an engaging activity for movie enthusiasts in which users share their best guess of character traits of their favorite characters, and then they see what their friends believe and how much users agree with each other, fostering a community-driven data collection. This method, however, presents several challenges such as ensuring data quality and consistency due to varied interpretations among contributors, maintaining participant engagement, scaling the database management as contributions grow, addressing potential biases and representation issues, upholding strict privacy and ethical standards, and guaranteeing data verification to avoid fraud.

The second option pivoted towards leveraging Large Language Models like GPT. We noticed that GPT was excellent at character analysis. We created a database that includes 500 distinct psychological characteristics and validated the database's accuracy with movie enthusiasts, who confirmed the depth and relevance of the analyses produced by GPT. This technological approach streamlined the dataset building process. You may view the faceted navigation interface of this database.



Ratatouille Character Analysis by ChatGPT-4

User Experience Research

We conducted a multi-part study with 5 movie enthusiasts to better understand our target user group and test the efficacy and usability of our system.

Understanding the User

Post-Movie Engagement Behaviors

Since this product involves engaging with content related to a movie after a user has watched the movie, we wanted to understand people's post-movie engagement behaviors. We interviewed five movie enthusiasts and we learned about the diverse range of activities that these users partake in. One participant engages deeply by collecting movie merchandise, seeking related books, and viewing movie edits on YouTube. Another enjoys listening to soundtracks and reading reviews from fellow movie-goers on Letterboxd. A third delves into discussions about movie meanings on forums like Reddit and Quora. These insights demonstrate the diverse and rich ways enthusiasts interact with films beyond just watching them.

Proactive Effort Towards Self-Understanding

We also discussed their proactive efforts toward self-understanding. We were surprised to learn that many of our participants had been putting in significant active effort to understand themselves. One participant had been exploring fundamental aspects of their identity, like gender and ethnicity, and used daily conversations with family and inspirational content on Pinterest for reflection. Another participant, struggling to focus and professional conflicts, was utilizing online tests, Google searches, and professional consultations. A third had been engaging in written and verbal self-reflection, discussing their own behavior with friends to deepen their understanding. Through these interviews we surfaced some of the ways in which people are actively trying to understand themselves..

Usability Testing - Phase 1

Choosing the list of most impactful movies

We wanted to understand how users are choosing their list of movies. One participant chose movies that left them wanting more of similar stories. Another chose films based on how memorable the storylines were and potential for rewatching. Another participant used the heuristic of how much they refer to that movie.

Engagement

Understanding oneself is a complex process that demands continuous motivation. We gauged participant engagement in the content by presenting them with a 14-page document detailing protagonists' characteristics from their favorite movies. Although participants were told they could stop reading at any time, they spent an average of six and a half minutes thoroughly engaged, evidenced by frequent laughter and verbal reactions.

Efficacy of the Methodology

Do these characteristics reflect participants' own characteristics?

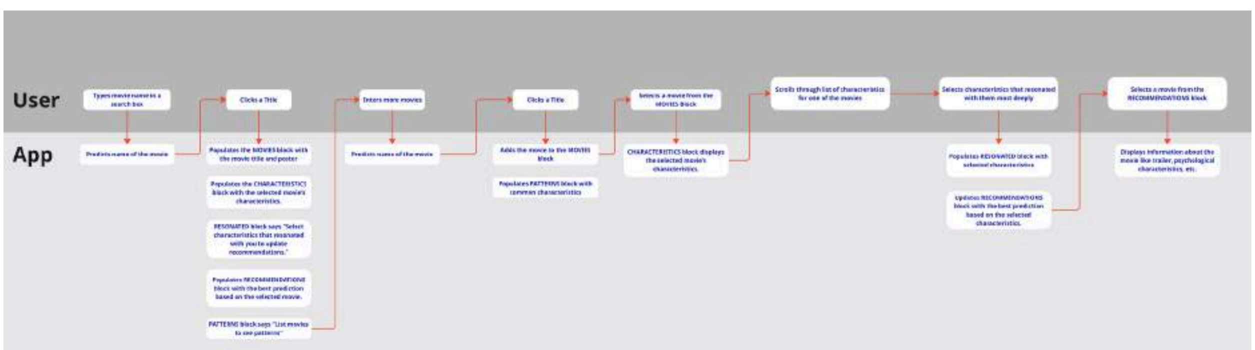
We examined participants' preferences for characters by presenting them with two lists of psychological patterns—one derived from their own favorite films and another from a different participant's favorites. We asked participants to choose between two hypothetical movies, each featuring a protagonist embodying traits from one of the lists. Intriguingly, all participants consistently chose the movie with traits from their own list, indicating a preference for familiar psychological patterns.

We further explored participants' emotional and cognitive responses to these traits. Without informing them of the traits' origins, we asked them to reflect on each characteristic individually and share their thoughts and feelings. The participants in general showed great enthusiasm in claiming most of the patterns described something about themselves. Surprisingly, their enthusiasm was higher in claiming the patterns in flaws than other characteristics that were generally positive. One participant notably was surprised by the protagonist centrality in their movie choices

User Experience Design

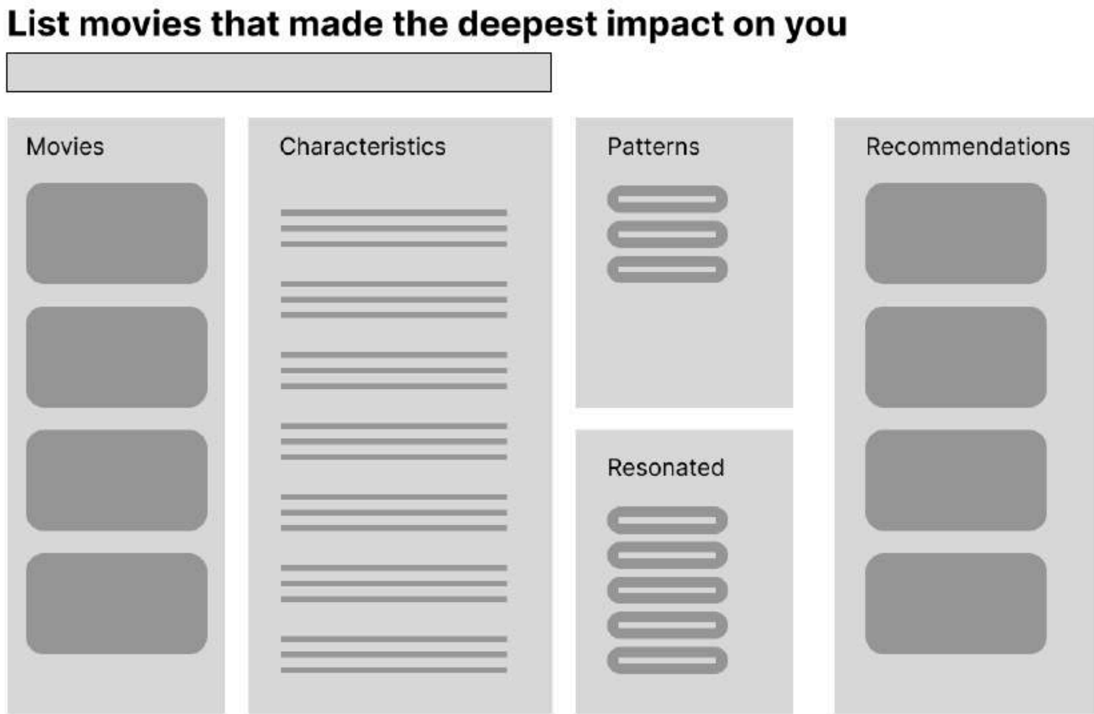
Service Blueprint

Based on the insights that we gained from our research, we designed the following Service Blueprint. This provided the engineer with the expected interactions between the user and the app.



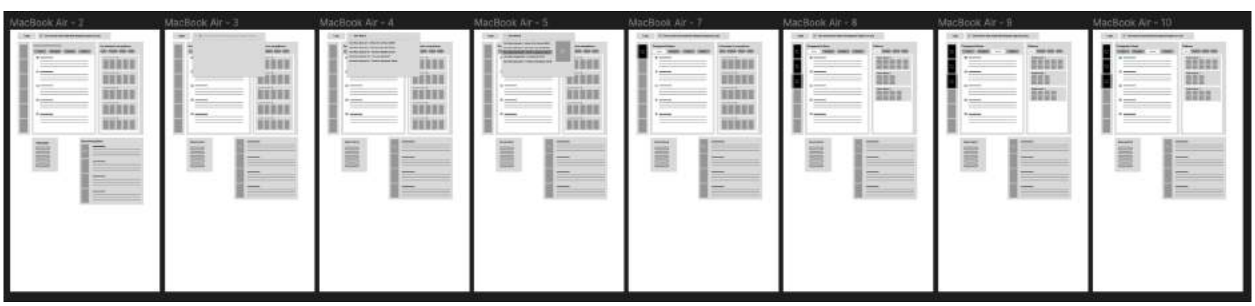
Design Iterations

Since our app involves presenting a variety of information that updates with multiple sequentially inputs by the user, we decided to design our app in the style of a dashboard.



Low-fidelity Design

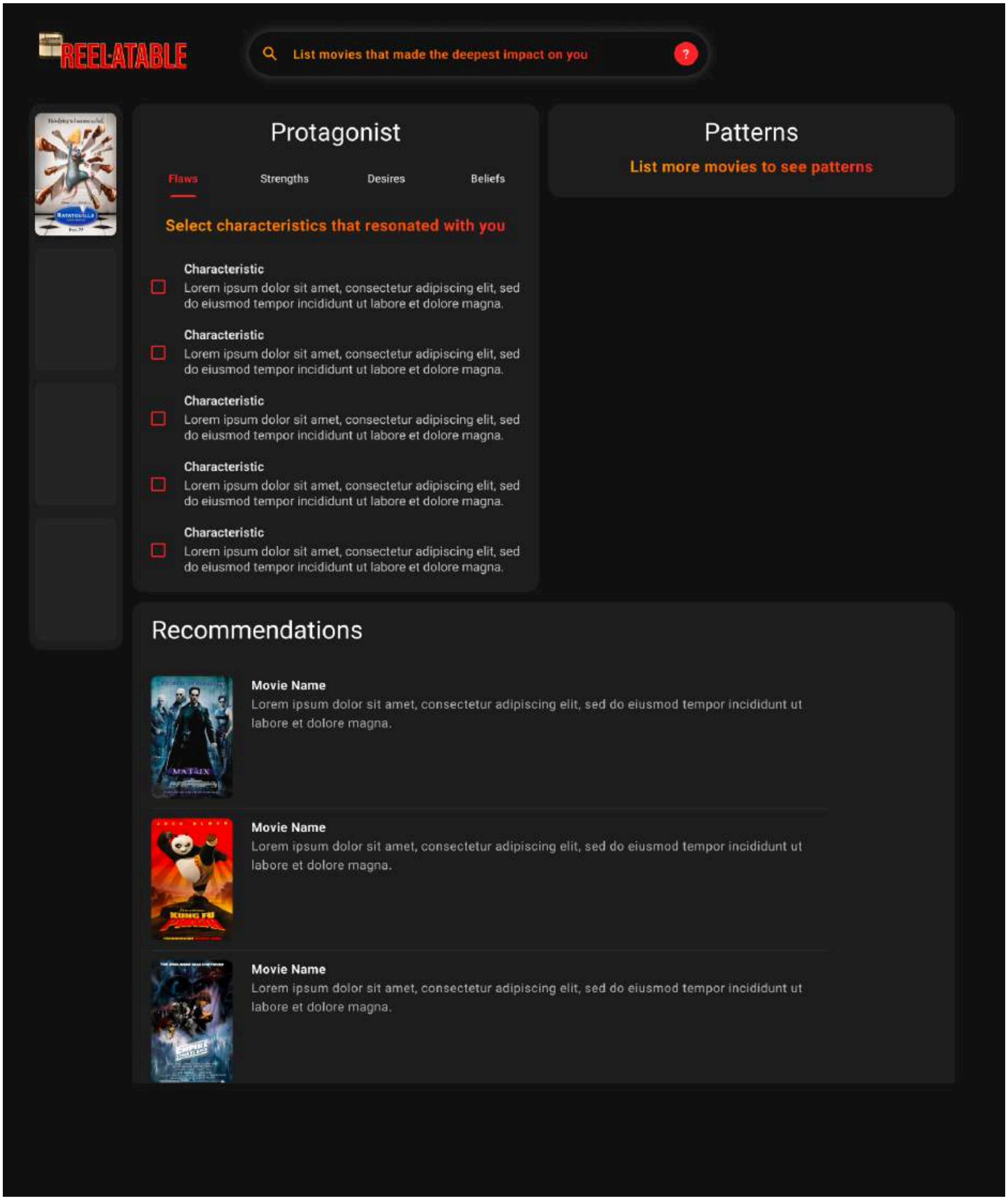
The following prototype complements the Service Blueprint.



Low-fidelity Figma Prototype (Link)

Through the usability study for the low fidelity prototype, we learned two major insights

1. Pattern did not need to have movie posters.
2. “Resonated” Container needed to be next to Characteristics for easy connection (Gestalt).

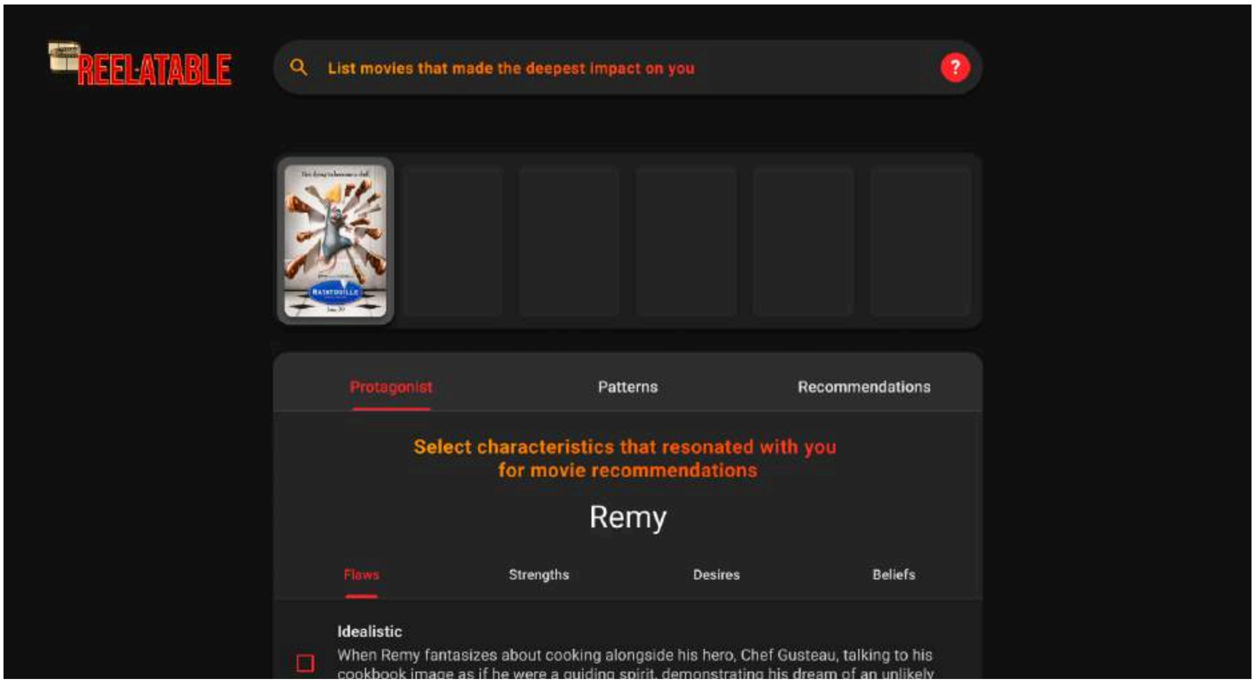


First Iteration of the High-Fidelity design([Link](#))

From the usability of the first iteration of the high fidelity prototype, we gained the following insights

- There was too much information in a single view and the participants experienced an information overload.
- Users did not perceive the movie list as a list.

In the subsequent interactions we changed the layout into a Tabbed Navigation, moved the movie list under the search bar.



Version 4 of the Design Prototype ([Link]())

Market Analysis

Target User Group

Our ideal customer persona is *people who want to understand themselves better.*

We hypothesized, and validated through user research, that movies can be a fun and interesting way to learn more about ourselves. By using movies as a tool, we can make the process of self-understanding more engaging and enjoyable.

So, our ideal users are *those who are interested in introspection and also love watching movies*. They'll be able to explore their thoughts and feelings through the stories and characters they see on screen, making the journey of self-discovery both enlightening and entertaining.

Market

The Total Addressable Market is the entire global market of individuals who are interested in self-understanding and personal growth. This includes anyone who might find value in introspection through various media, particularly those who enjoy movies.

The global self-improvement market is estimated at $39.2 billion. Since the global streaming market has a penetration of 18%, we can assume that our TAM is 18% of $39.2 billion or ~$7 billion.

Note that this is a conservative estimate as it is likely many users in the self-improvement market are also video streaming consumers (as both correlate with wealth and income), and the penetration of video streaming in this market may be higher, resulting in a higher TAM.

Competitive Landscape

While there are various competitors in the mental well-being and self-understanding space, and others that provide an engaging platform to increase user engagement, very few (eg. Headspace and Calm) operate in both spaces.

Self-Understanding Only

Competitor	**Goal**	**Enables Self-understanding**	**User Effort**	**Explainable**	**Engaging**
Talkspace	Online therapy and counseling services	✓	High		
Reflectly	Mood journaling app for self-awareness	✓	High	✓	
Truity	Personality assessment and insights platform	✓	Low		

Entertainment Only

Competitor	**Goal**	**Enables Self-understanding**	**User Effort**	**Explainable**	**Engaging**
Netflix	Streaming platform for movies and TV shows		Low		✓
Letterboxd	Social platform for movie enthusiasts		Low		✓

Both

Competitor	Goal	Enables Self-understanding	User Effort	Explainable	Engaging
Headspace	Meditation and mindfulness app		Low		✓
Calm	Relaxation and meditation app		Low		✓

While tools like Headspace and Calm, as meditation and relaxation apps respectively, offer low user effort and are engaging, they are not directly focused on self-understanding.

Overall, **there's a clear need for a solution that effectively aids in self-understanding, is engaging, requires low user effort, and is explainable**—a balance not fully achieved by any single competitor in the current landscape.

Product Considerations

Forces of Progress

Push Forces (Dissatisfaction with the Current State):

- **Lack of Self-Knowledge:** The user might feel a general sense of not understanding themselves well. They might have questions about their motivations, values, or desires.
- **Difficulty with Introspection:** They might struggle to analyze their own thoughts and feelings on their own.
- **Unsatisfying Self-Discovery Methods:** Traditional methods of self-exploration (e.g., journaling, personality tests) might feel boring or ineffective.

Pull Forces (Desire for Improvement):

- **Increased Self-Awareness:** The user desires a deeper understanding of their inner world.
- **Personal Growth:** They want to learn and grow as a person.
- **Improved Decision-Making:** They hope understanding themselves better will lead to better life choices.
- **Greater Well-Being:** They believe self-knowledge can contribute to a happier and more fulfilling life.
- **Engaging Self-Discovery:** They enjoy learning through stories and visual media, making movies an attractive tool for self-exploration.

Habit:

- **Comfort with the Status Quo:** The user might be comfortable with their current level of self-understanding, even if it's not ideal. They might be hesitant to invest time or effort in a new approach.

Anxiety:

- **Fear of the Unknown:** Delving into self-discovery can be confronting. The user might be anxious about what they might learn about themselves.
- **Analysis Paralysis:** They might worry about "overthinking" things and getting stuck in analysis instead of taking action.

Jobs to be done (JTBD)

Based on the forces of progress we identified for users interested in self-understanding, here are the potential "jobs to be done" they might be trying to accomplish:

JTBD1: Uncover deeper truths about themselves in a fun and engaging way.
JTBD2: Make self-understanding more explainable.
JTBD3: Feel confident and supported throughout their journey of self-understanding.

Core Job:
When I find traditional methods of self-discovery boring, I want to use an engaging way to gain insights into myself so that I can understand myself better.

Product Features

1. **Objective/Goal:**
    1. We are building a web app with the primary goal of helping users improve their self-understanding. The product will engage users in meaningful interactions around their favorite movies while improving self-understanding as a by-product.
2. **Features:**
    1. Users can read psychological analysis of their favorite movie characters which would be novel and interesting information for the user. Often users only capture psychological characteristics subconsciously while watching the movie. This leads to an emotional impact that is strong but not often understood. The system makes it conscious, explaining why the movie might have resonated with them.
    2. Users can find out psychological patterns in their favorite movies. This helps them understand themselves and makes them feel understood.
    3. Users can find movies that are similar to a movie that they really liked. Currently, there is no easy way to find a movie based on the type of characters.
    4. System explains exactly why a movie is being recommended.

Product Roadmap

The product roadmap outlines the development phases for a movie recommendation app designed to help users understand themselves better. The roadmap is divided into four phases: Research, Product Management, Design, and Development. During the Research phase, the team conducted user research to understand user pain points and test different prototypes. Specific tasks include prompt engineering, user group identification, usability testing, and market analysis. The Product Management phase includes market analysis, user segmentation and JTBDs (Jobs To Be Done) analysis. During the Design phase, the team created wireframes, low-fidelity prototypes, and high-fidelity prototypes.

The Development phase was not scheduled until after the Design phase. During this phase, the team developed the backend, frontend, and machine learning model for the app. The process started with building the data infrastructure. Once the database of movies was ready, the backend and frontend development started parallelly. Lastly the entire team tested the entire app together.

Engineering

Reelatable is powered by a Machine Learning service that enables hybrid search using Retrieval-Augmented Generation (RAG). This ML service creates vector embeddings of movies, their plotlines, and certain characteristics traits of the protagonist identified through user research, and upserts them to a vector database.

The backend service created using Flask then runs queries against this vector database and augments them with some local processing and querying to power endpoints that are then consumed by the frontend service.

The frontend service is created using Flutter to enable multi-platform development. While this project has been designed primarily for the web, a key strength of our choice of technology stack is that we have also been able to build an Android app.

Additionally, many technology choices in this project are motivated by the following considerations -

- Keeping it modular, so we can easily swap between tools and technologies to respond to user research in an agile manner
- Keeping it lightweight, from the perspectives of computation, cost and effort, since we are bound by a tight timeline and a stringent budget

System Diagram


ML Service
TMDB API
Top 2500 movies (names + metadata)
MOVIE JSON
Engineered Prompts to extract named entities
GPT-3.5-TURBO (CHAT COMPLETION API)
WIKIPLOTS EXTRACTOR
Wikipedia descriptions
MOVIE PLOT JSON
Named Entity Recognition
NAMED ENTITIES JSON
DENSE EMBEDDINGS
768 dimensions
ALIBABA-NLP/GTE-BASE-EN-V1.5 (TEXT EMBEDDINGS API)
SPARSE EMBEDDINGS
12 dimensions
PCA
Dimensionality selected based on some qualitative experiments - can benefit from further tuning
ALIBABA-NLP/GTE-BASE-EN-V1.5 (TEXT EMBEDDINGS API)
Best ranked model on Hugging Face MTEB benchmark
Constraints:
• <1GB memory
• English-only
• SentenceTransformers-supported
TRAIT FACETS
Validation, Error Checking, and Reprompting
TRAITS LINKER
Batched upserting
Intermediate Artifacts
PINECONE INDEX
PICKLED PCA MODEL
Flask Backend Service
QUERY >2500 UNIQUE VECTORS WITH RANDOM SEED
Workaround for query_all (not supported by Pinecone)
MOVIE TRAITS
ALIBABA-NLP/GTE-BASE-EN-V1.5 (TEXT EMBEDDINGS API)
Replicating hybrid search functionality on the client side to work with selection of subset of traits
PCA
HYBRID RAG SEARCH
QUERY 1 VECTOR WITH RANDOM SEED + TITLE FILTER
Assuming every movie title is unique
EXTRACT TRAITS
CLUSTERING
K-means; no.of clusters computed from silhouette score
MOVIE PLOTLINES
ALIBABA-NLP/GTE-BASE-EN-V1.5 (TEXT EMBEDDINGS API)
FIND REPRESENTATIVE TRAITS
Public API
GET GET_ALL_MOVIES/
With local caching
GET GET_MOVIE_METADATA/
POST GET_MOVIE_RECOMMENDATIONS/
POST SEARCH_BY_TRAITS/
POST GET_MOVIE_PATTERNS/
Flutter Frontend Service
Deployment
DOCKER
KUBERNETES
GODADDY
Domain + add-ons
WEB
NGINX
GOOGLE KUBERNETES ENGINE CLUSTERS
• Standard clusters with highmem CPU (>8GB), not compute-optimized to keep it cost-effective
• Nodes with 3 pods and autoscaling to enable redundancy
DOCKER
KUBERNETES
MOBILE (ANDROID)
Build and install apk

Machine Learning Service

Code: [Colab]

The Machine Learning service is authored in Python, using Google Colab. This service creates vector embeddings to upsert to a Pinecone vector index. The movies (titles and some metadata) are collected using The Movie Database API, and are processed into dense and sparse embeddings for each movie.

Dense Embeddings

- Source: Plotlines for each movie from Wikipedia:Database download, using WikiPlots Extractor library
- Generation:
    - LM Input: Movie plotline
    - LM: Model with text embedding support - GTE-Base-EN v1.5
    - Process: The LM encodes the semantic content of the movie's metadata into vector space
    - Output: Dense embeddings representing the entire plotline of the movie

Sparse Embeddings

- Source: Characteristics of the protagonist as generated by GPT 3.5 Turbo
- Generation:
    - LM Mode: Q&A mode.
    - Process: The LM extracts a set of tags related to psychological traits like flaws, strengths (personality traits), desires and flaws. These attributes are then passed through the GTE-Base-EN v1.5, and reduced in dimensionality using PCA (with a tunable parameter for the number of dimensions set to 12)
    - Output: Sparse embeddings representing the extracted tags in a lower-dimensional space

Storage

- Index: Pinecone vector index using dot product for vector similarity metric (to enable hybrid search)
- Data: Dense and sparse embeddings
- Metadata:
    - Movie name
    - Additional relevant metadata (release year, ratings, image url etc.)

Another artifact that is passed from the ML service to the backend is a pickled PCA model that is fit to the traits of all the movies.

Additional Considerations

Data Selection

Popular Movies from TMDB were selected. We specifically picked popular movies from there, meaning movies that are widely known and watched. Then, we used WikiExtractor to get the plots for these movies, and after sanitizing and deduplicating the inputs, we reduced the list to around 2500 movies.

- **Depth of Information**: TMDB has a variety of metadata, and although we are not using most fields today, this gives us the option to increase functionality and react to user feedback faster.
- **Cost and Toil**: TMDB API is free to use and well-supported, so it was preferred to other alternatives. Even though Wikipedia is not as easy to work with because of the lack of a dedicated API for this, existing community contributions were easy to piggyback on.
- **Data Reliability**: Wikipedia's community contribution model makes it a more trusted source for movie plotlines than most alternatives.

Model for Generating Attributes/Traits

Due to budget constraints, we initially adopted OpenAI's GPT-3.5 for generating a characteristics database for the movies.

- **Model Upgrade Considerations:** Transitioning to GPT-4 was considered for its potential to improve database quality and recommendation accuracy but was not implemented to maintain financial viability.
- **Computation Limitations:** The project was constrained by the computation capacity of the free version of Google Colab. Upgrading to Colab Pro to facilitate model fine-tuning was deemed cost-prohibitive, so we made the decision to avoid it.
- **Potential for Fine-Tuning:** Developing a fine-tuned model could significantly enhance recommendation precision. However, the required effort, resources, and experimentation to achieve optimal results would substantially increase project costs and complexity.

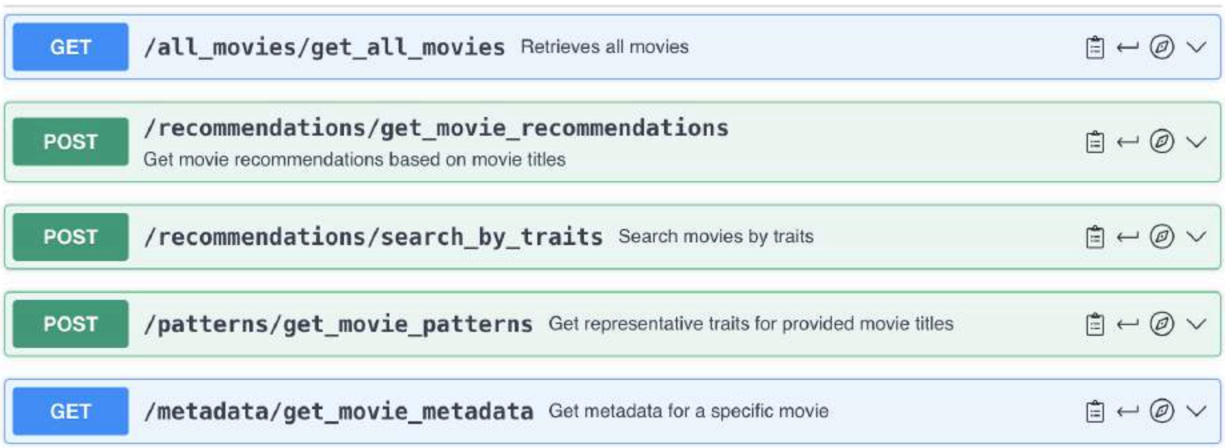
Sparse Embedding Generation Strategy

- **Embedding Strategy:** The decision to bypass a simple one-hot encoding approach, which couldn't capture semantic relationships, led us to adopt a method that maintains the semantic context of traits. We chose to use a language model capable of generating meaningful embeddings.
- **Language Model Selection:** We selected a compact model (under 1GB memory usage) that was fine-tuned for English and compatible with the SentenceTransformers library. Based on the MTEB table from Hugging Face, which is considered a reliable benchmark for language models for text embeddings, the GTE-Base-EN v1.5 by Alibaba NLP was chosen for its high rating.
- **Dimensionality Reduction:** Principal Component Analysis (PCA) was utilized to reduce the dimensionality of the embeddings to 12. This decision was based on preliminary assessments of the clustering quality of sparse embeddings from a sample set of movies. Although more detailed methods like analyzing cumulative explained variance could have been used, they were

deemed beyond the scope of this project. The chosen dimensionality ensured a stable and meaningful clustering without overly complicating the model or the process.

Backend service

Code: [Github]
The backend is created in Python using the Flask framework. The API has five major endpoints -



Link to Swagger

Get_all_movies

Endpoint: /all_movies/get_all_movies
Method: GET

Retrieves a list of all movies in the Pinecone database. This endpoint is computationally intensive, so results are cached after the first call to improve performance.

Get_movie_patterns

Endpoint: /patterns/get_movie_patterns
Method: POST

Identifies clustered patterns and representative traits of the user's selection of traits from the selected movie list.

Get_movie_recommendations

Endpoint: /recommendations/get_movie_recommendations
Method: POST

Provides movie recommendations using a hybrid RAG search to find proximal movies in the vector database, based on a weighted ratio of dense and sparse embeddings (alpha), with a default of 0.5. The weight ratio is dynamically adjusted based on the performance and relevance of the recommendations. This ratio is tweaked based on user research, and for the scope of this project, we are using a ratio of 0.5 to use both dense and sparse embeddings in the search.

Search_by_traits

Endpoint: /recommendations/search_by_traits
Method: POST

Allows users to search for movies based on specific traits.

Get_movie_metadata

Endpoint: /metadata/get_movie_metadata
Method: GET

Retrieves detailed metadata for a specific movie based on its title.

Retrieval-Augmented Generation (RAG) Hybrid Search

- The Pinecone database retrieves the closest movie vectors based on the query vector, which is a combination of sparse and dense embeddings. Sparse embeddings are generated on the backend server using the same model as the one in the ML service, and the pickled pre-fit PCA model is used for dimensionality reduction.

Additional Considerations

Clustering strategy

To retrieve representative traits, we are creating K-means clusters and identifying the trait that is closest to the centroid of the largest cluster.

- **Number of clusters**: We are using silhouette score to identify the optimal number of clusters. To prevent too many clusters from being formed, we are setting a minimum based on the number of traits selected.
- **Representative trait selection**: We are considering the largest cluster as the best indicator of the user's selected traits. If multiple clusters are tied for size, the cluster that is closest to the centroid of the global population is considered. Within this cluster, the trait that is closest to the centroid of the cluster is deemed the most 'representative' trait. These choices need to be further vetted through user research.

Frontend service

Code: Github

The frontend of the project was developed using Flutter due to its user-friendly nature, extensive online resources, and seamless integration with Flask backend, enabling the creation of multi-platform applications. For this project, we built both a web app and an Android app using Flutter.

Functionality Overview

- Home Tab: Upon initialization, the app fetches the latest list of all movies from the backend, populating the search bar dropdown to facilitate easy movie selection. It displays movie characteristics metadata fetched from the Pinecone database as users select movies. Users can select characteristics that resonate with them, which are stored for future reference.
- Patterns Tab: This tab triggers an API call to retrieve clustered patterns and representative traits based on the selected movies. These patterns, representing clusters of characteristics shared among chosen movies, are displayed and updated dynamically as more movies are added.
- Recommendations Tab: Offers personalized movie recommendations based on the selected movies and traits. It displays movie posters, and users can click on a poster to read an overview of the movie, aiding in their decision-making process.

Output to User

- Movie Characteristics: Displays a list of characteristics for movies chosen by the user.
- Patterns: Shows a list of movie characteristics that are similar between movies chosen by the user.
- Recommendations Based on Movies: Provides a list of movies closely matching the user's preferences based on the movies that resonated with them.
- Recommendations Based on Resonated Characteristics: Offers a list of movies closely matching the user's preferences based on the characteristics that resonated with them.

This structure allows for a coherent, user-friendly interface that facilitates easy navigation and interaction across various features and functionalities of the application.

Additional Considerations

Caching

The first get_all_movies call is very time-intensive, so we added a load animation while the user is waiting for the response from the server. We implemented caching to accelerate recurring use.

- **Local storage**: all_movies is stored as a blob locally. While this may end up reducing performance of the frontend in inexplicable ways, this reduces latency of repeated movie name retrieval significantly.

Web-first

While Flutter is inherently responsive across platforms, some platform-specific work is often required for the fit and finish.

- **Optimized for web**: The app is functionally correct and performant on Android, but the layouts are designed for web in the interest of conserving time and effort.

Downstreaming design changes

The Minimum Viable Product (MVP) implementation lagged behind the design phase, as the decision was made not to impede the progress of the design and user research efforts while the MVP was being developed. This, however, reduced the lead time to incorporate design updates into the front-end, and we opted for functionality over UI quality.

Deployment

The deployment strategy for the web application involves a combination of Docker and Google Kubernetes Engine (GKE) to ensure a robust, scalable, and manageable rollout of services. This strategy is designed to optimize deployment processes and ensure high availability and scalability while not exceeding the project budget.

Docker

Objective: Containerize the frontend and backend services to ensure environment consistency and streamline deployment processes.
Action:

- Backend Service: The Flask application along with its machine learning components is packaged into a Docker container. This encapsulation includes all necessary dependencies, ensuring that the service operates uniformly regardless of the deployment environment.
- Frontend Service: The Flutter application is built into static files and served via a lightweight Docker container using Nginx, optimizing delivery and performance.

Google Kubernetes Engine (GKE)

Objective: Leverage managed Kubernetes services for deploying and scaling the application with high availability.
Action:

- Cluster Setup: Deploy the application on GKE using standard clusters configured with high-memory CPUs to handle memory-intensive operations efficiently.

- Service Management: Kubernetes services are defined to manage network traffic to both frontend and backend components, with load balancing to ensure even distribution of client requests.

Android APK build

Objective: Enable one-off Android app building and installation without a dedicated release process
Action:

- Configuration of signing in Gradle: Generate a keystore and add keystore information to build.gradle.
- Installation: Build the release apk and install to an Android Virtual Device booted up from within Android Studio.

Additional Considerations

GKE Cluster Setup

The GKE Clusters are set up using standard deployment, and are configured to use high-memory CPUs.

- **OOM Errors**: Clusters deployed with Autopilot and no customization during setup often encountered OOM (out of memory) errors and were not suitable for on-server model operations.
- **Computation**: The calls to the backend are extremely slow (in some cases, an order of magnitude slower than what's observed with a local server). However, we still opted against increasing computational capacity and/or deploying some of the parallelizable workloads onto GPUs or TPUs, because they can easily exceed our budget if we are not being very observant. The cost we incur is a serious degradation in performance and very high latency, but the flows are still functional. Given more time, we would have liked to evaluate the performance bottlenecks and explore strategies to optimize the API response times. This could involve assessing the feasibility of using a GPU or exploring alternative cloud hosting solutions that offer better performance within the budget constraints.
- **Cost**: Standard deployment might incur high costs if not carefully observed (especially if vertical scaling is enabled). Therefore, we added budget thresholds and monitored costs carefully so we are well within budget. If we were to continue with this project, we would conduct a comprehensive cost analysis to identify areas where costs can be optimized or reduced. This may involve reevaluating the choice of cloud provider, negotiating better pricing plans, or exploring cost-effective alternatives for the employed models and services.
- **Redundancy**: With standard deployment, the number of nodes is configurable. We are using 3 nodes as that is the recommended minimum to minimize downtime while maintaining a cost-efficient setup. This comes at the cost of scalability, however.

For the application flow and links to the code, refer to the Appendix - Final Report.

Limitations & Future Work

We believe Reelatable has a lot of potential for improvement. Some areas of improvement we identified are as follows -

Engineering

- Improving deployment through dedicated cloud resources to increase scalability and stability, and improved CI/CD for faster development iterations
- Increased profiling and performance monitoring to enhance performance, quality, and observability
- Expanding the data sources and improving the cleaning and validation flows
- Enhanced models (eg. GPT-4 instead of GPT-3.5-Turbo, and Mistral or Gecko instead of GTE-Base) for better model performance
- Fine-tuning model used for Named Entity extraction to increase relevance and quality

User Research

- Conducting quantitative analyses to determine how accurately the patterns represent the user's own significant psychological dimensions
- Validating parameter choices (eg. hybrid search parameters) through usage and/or additional user research
- Narrowing down the broader group of movie enthusiasts to a group that would benefit the most from this
- So far usability testing was done on Design Prototypes. Usability testing for the app is yet to be done.

UI Design

- Refined interface design
- Streamlined navigation
- Increased functionality to enhance user engagement and satisfaction

Product Design

- Beyond movies, this methodology might be applicable to other forms of art like novels (which is another narrative artform) and songs which are written around emotions.

Conclusion

Our project began with a comprehensive research on self-understanding and relevant psychological characteristics as it relates to movies. We conducted usability studies and qualitative as well as quantitative research to guide out product design. We continuously iterated our design based on user feedback and embraced challenges as opportunities for growth. We leveraged cutting edge technologies such as Retrieval-Augmented Generation using large language models and vector databases to power the core of our application.

In closing, we extend our gratitude to all stakeholders, research participants, academic advisors and supporters who have contributed to the success of this project. Together, we have embarked on a journey of exploration, discovery, and transformation, and we look forward to the continued evolution and impact of our platform in the years to come.

Contributions

Ankita Shanbhag

- Engineering system design and architecture
- Data selection of movie metadata and database creation of 2500 movies
    - Integration with APIs and external libraries for data collection
    - Data cleaning
    - Named Entity extraction and linking
    - Prompt engineering for metadata
    - Data validation
- Machine learning
    - Pinecone index creation
    - Embedding creation and upsert
    - Pinecone index retrieval
    - RAG + Hybrid search based on patterns
- Back-End Development
    - API creation and documentation
    - On-server embedding generation
    - Clustering, including performance and behavior optimization
    - Some API performance optimization, using pickling etc
- Front-End Development
    - Design of frontend components and flows
    - Error handling
    - Integrating asynchronous processing
    - Backend integration using the public API
    - Basic caching support
- Deployment
    - GCP setting up and scaffolding

    - Containerization using Docker
    - Kubernetes setup and rollout
    - Cluster management and optimization
    - Android app support and Gradle + manifest updates
    - Web domain and DNS configuration

Saurabh Chachra

- Literature review
    - Self-Concept Clarity
    - Categorization Theory
    - Storytelling
- UX Research
    - Understanding the Target User Group
    - Usability Testing
    - Efficacy of Methodology of patterns representing the user
- Product Concept and Design
- UX Design
    - Service Blueprint
    - Design System
    - Prototyping
- Prompt Engineering

Hrishikesh Srinivas Nagaraju

- Researched movie structure and film development to understand key aspects of creating a movie.
- Created Product Roadmap
    - Estimated timeline (First Half)
    - Created Gantt chart
- Conducted initial GPT API testing:
    - Prompt engineering
    - Understanding API documentation
    - Performance testing the API
    - Contextualizing the API
- Frontend development
    - Built flows for movie protagonist characteristics, Patterns and list of resonating characteristics
    - Created API request and response formats to integrate with backend
    - API calls and data parsing
- UX Research:
    - Usability study

    - Asking questions during interviews
    - Taking notes
    - Collect data
    - Preprocess data and prepare for analysis
    - Analyzing feedback
    - Recruiting participants
- Created the Economic & Business analysis section
    - Analyzed Pricing Strategies
    - Value networks
    - Regulations

Kinshuk Nigam

- Market Analysis
    - Secondary Research
    - User Reviews for products
    - Are movies relatable to users
- Choosing the Target User
    - Research on user segments
    - Problem prioritization
    - Target user group selection
- Product Value Proposition
- Competitor analysis with direct and indirect competitors
- Product Management
    - Forces of Progress
    - Jobs to be done
    - Product Features
- Project Management
    - Product Roadmap Planning (Second Half)
    - Trello Project Management
    - Leading Scrums for updates, roadblocks and next steps
- MVP scalability:
    - Coded initial GPT API to work with sample 10 movies
- Initial Backend Development:
    - Coded the /getmovies API (now /Get_all_movies)
    - Coded the /getpatterns API (now /Get_movie_patterns)
    - Integrate with Frontend requests
    - Test run APIs

References

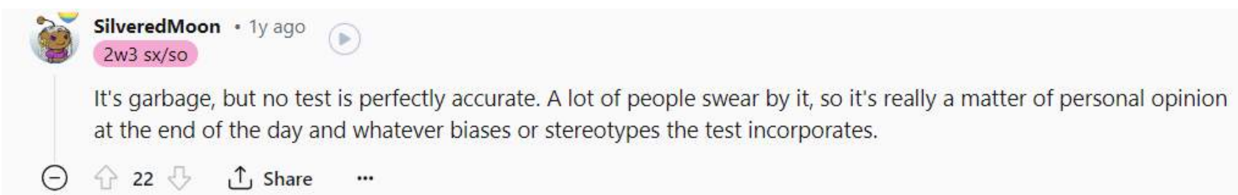
1. Aitchison, J. (2012). Words in the mind: An introduction to the mental lexicon. John Wiley & Sons.
2. Alibaba-NLP/gte-base-en-v1.5 · Hugging Face. (n.d.). Huggingface.co. Retrieved May 3, 2024, from https://huggingface.co/Alibaba-NLP/gte-base-en-v1.5
3. Black, J., & Barnes, J. L. (2015). Fiction and social cognition: The effect of viewing award-winning television dramas on theory of mind. Psychology of Aesthetics, Creativity, and the Arts, 9(4), 423.
4. Briggs, J. (n.d.). Getting Started with Hybrid Search | Pinecone. Www.pinecone.io. https://www.pinecone.io/learn/hybrid-search-intro/
5. Campbell, J. D. (1990). Self-esteem and clarity of the self-concept. Journal of personality and social psychology, 59(3), 538.
6. Dierdorff EC, Rubin RS. (2015). Research: We're not very self-aware, especially at work. Harvard Business Review, March 12. https://hbr.org/2015/03/research-were-not-very-self-aware-especially-at-work
7. Getting Started. (n.d.). The Movie Database (TMDB). https://developer.themoviedb.org/reference/intro/getting-started
8. Gurung, R. A., Sarason, B. R., & Sarason, I. G. (2001). Predicting relationship quality and emotional reactions to stress from significant-other-concept clarity. Personality and Social Psychology Bulletin, 27(10), 1267-1276.
9. Lewandowski Jr, G. W., & Nardone, N. (2012). Self-concept clarity's role in self–other agreement and the accuracy of behavioral prediction. Self and Identity, 11(1), 71-89.
10. markriedl. (2024, January 15). markriedl/WikiPlots. GitHub. https://github.com/markriedl/WikiPlots
11. McKee, R. (2005). Story. Dixit.
12. MTEB Leaderboard - a Hugging Face Space by mteb. (n.d.). Huggingface.co. https://huggingface.co/spaces/mteb/leaderboard
13. Quora. (n.d.). Do people relate films with their real lives? [Question]. Retrieved from https://www.quora.com/Do-people-relate-films-with-their-real-lives
14. Reddit. (2011, December 6). What's the opinion on the Truity Enneagram test? [Online forum post]. Retrieved from https://www.reddit.com/r/Enneagram/comments/14bbcuo/whats_the_opinion_on_the_truity_enneagram_test/
15. Reddit. (2020, December 26). PSA: Stop taking the freaking Truity Enneagram test! [Online forum post]. Retrieved from https://www.reddit.com/r/Enneagram/comments/kka1l8/psa_stop_taking_the_freaking_truity_enneagram/
16. Rosch, E. (1978). Principles of categorization. In Cognition and categorization (pp. 27-48). Routledge.
17. Schneider, M. (2018, February 13). Why films are relatable and create a physical emotional response. 34th Street Magazine. Retrieved from

https://www.34st.com/article/2018/02/why-films-are-relatable-and-create-a-physical-emotional-response
18. Schwartz, S. J., Meca, A., & Petrova, M. (2017). Who am I and why does it matter? Linking personal identity and self-concept clarity. Self-concept clarity: Perspectives on assessment, research, and applications, 145-164.
19. Wikipedia:Database download. (2021, August 25). Wikipedia. https://en.wikipedia.org/wiki/Wikipedia:Database_download

Appendix - Market Analysis

Online personality tests can often prove to be intricate and challenging to decipher, presenting a level of opacity that complicates users' understanding. These assessments typically employ complex algorithms and psychological frameworks to analyze and categorize individuals based on their responses. However, the inner workings of these algorithms are often obscured from users, leading to a lack of transparency in how conclusions about personality traits are reached. Additionally, the nuances of human personality are vast and multifaceted, making it difficult for any test to capture the full complexity of an individual accurately. Furthermore, the language used in these tests may be technical or abstract, further distancing users from a clear comprehension of their results. Consequently, users may find themselves grappling with interpretations that feel detached from their self-perception, highlighting the challenges inherent in navigating the intricacies of online personality assessments.

PSA: Stop taking the freaking Truity enneagram test then using the screenshot to ask "What am I??" Eclectic Energies is your friend. Crunchy, but friendly.



What's the opinion on the Truity enneagram test?

Movies hold a unique place in people's lives as powerful vessels of storytelling that often resonate deeply with personal experiences and emotions. People frequently find themselves relating their own lives to the narratives depicted on screen, drawing parallels between the characters' journeys, conflicts, and triumphs, and their own. Whether it's identifying with a protagonist's struggles, finding solace in shared themes of love or loss, or seeking inspiration from characters who overcome adversity, individuals often use movies as a mirror to reflect upon their own circumstances and feelings. Furthermore, the immersive nature of cinema, with its visual and auditory elements, allows viewers to immerse themselves fully in the narrative, fostering a sense of connection and empathy with the characters and their stories. As a result, the experiences and lessons portrayed in movies can profoundly impact individuals' perceptions of themselves and the world around them, influencing their beliefs, values, and personal growth.

Quora

Do people relate films with their real lives?

Vishal Bajpai

Thinker, Analyzer and Observer · Author has **80** answers and **129.6K** answer views · 6y

Originally Answered: Do you relate your life with movies?

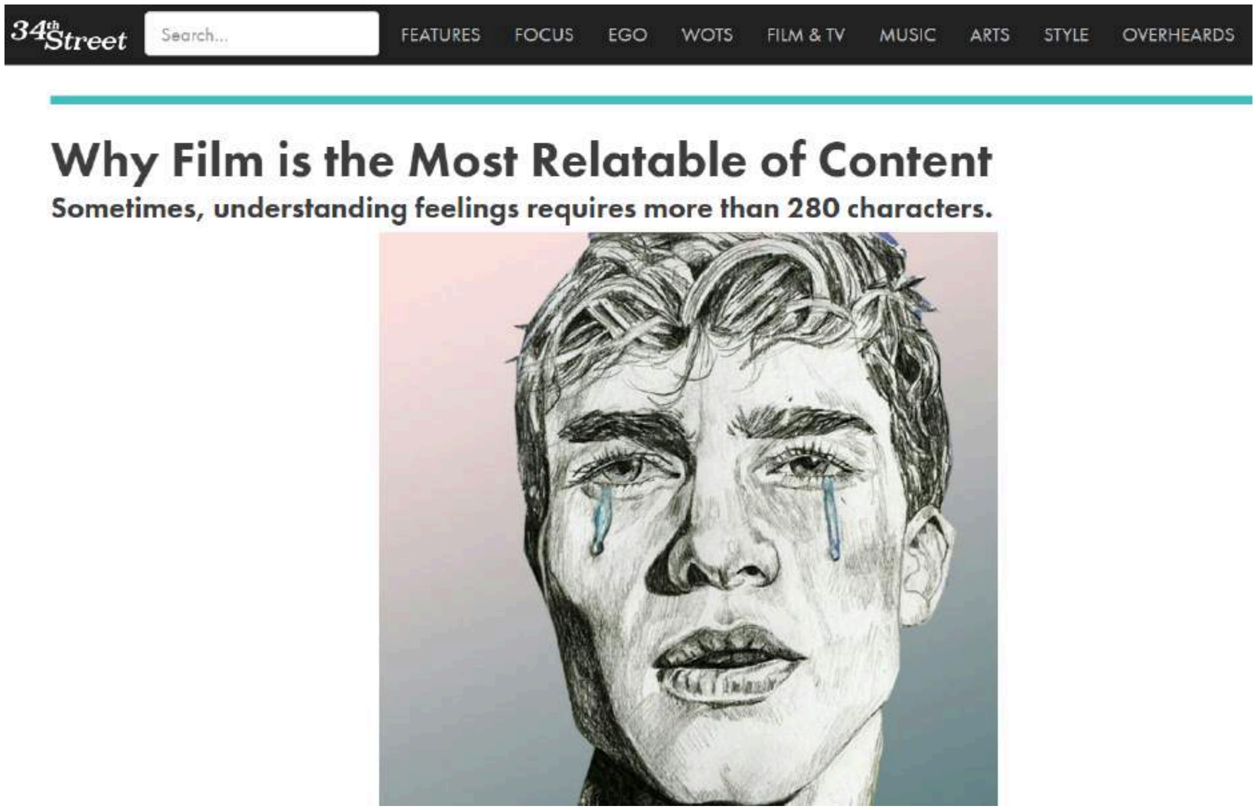
Yes. Everytime I watch a movie I feel like there is something for me to unfold. Sometimes I am not having this intuition, so I don't get urge to spend my priceless time on something which supposedely won't add anything. Although stories vary from one to another movie, characters still resonate with you. As I cleared, this is not true for every single movie I am watching. Maybe I may end up not liking it at all. But most of the times I wouldn't watch it knowing beforehand that it doesn't make sense for me even when it does for others. And it is quite common thing that movies you are getting attracted to are the ones you can relate with. There is strong connection between any form art and humans unless it evokes their emotions.

4.1K views · View upvotes

1 of 5 answers

Upvote · 6

Do people relate films with their real lives? - Quora

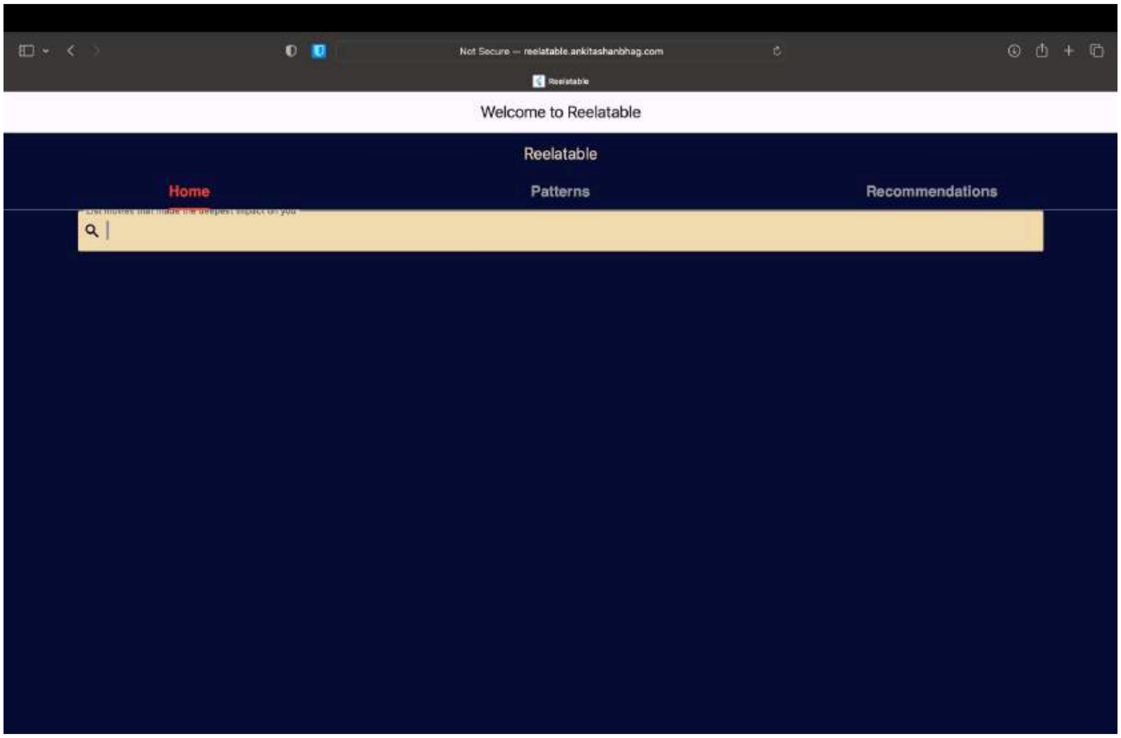


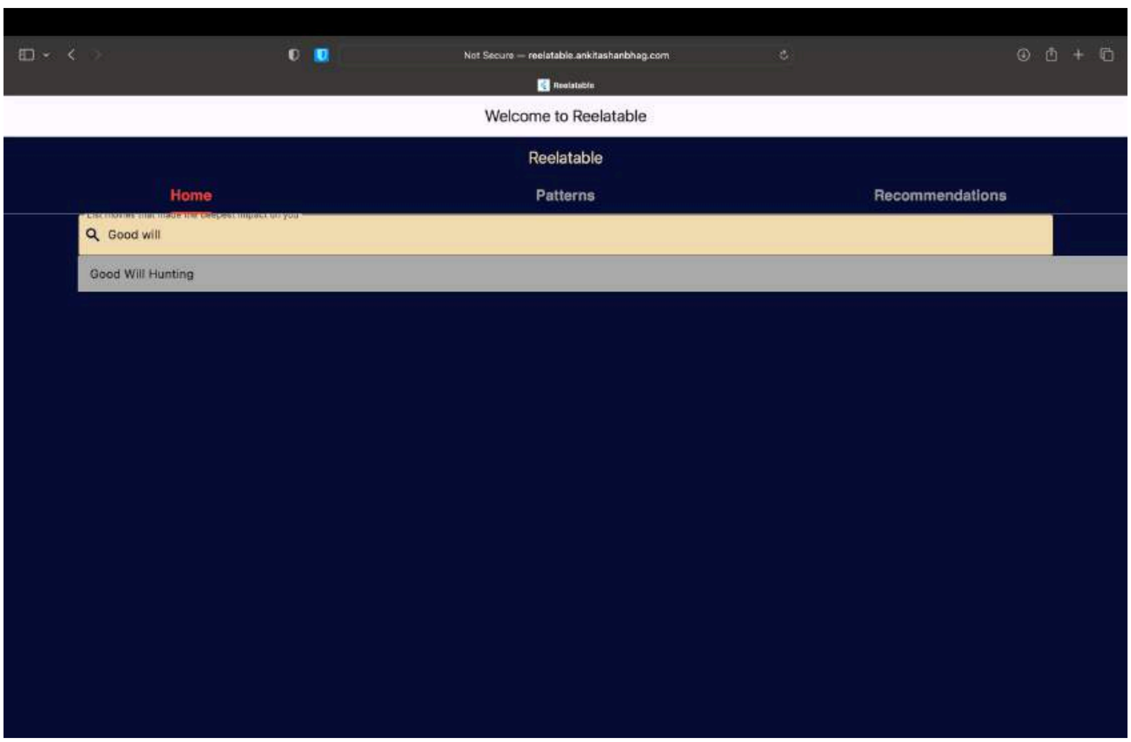
[*Why Film is the Most Relatable of Content | 34th Street Magazine*]()

The above article discusses the impacts of film on human emotions and connections. Films have a unique ability to evoke empathy and stir genuine emotions by presenting relatable stories and characters in a believable manner. Movies offer a deeper exploration of universal experiences and emotions. They serve as a lens through which viewers can reflect on their own lives and vulnerabilities, providing a richer and more meaningful form of entertainment and emotional connection.

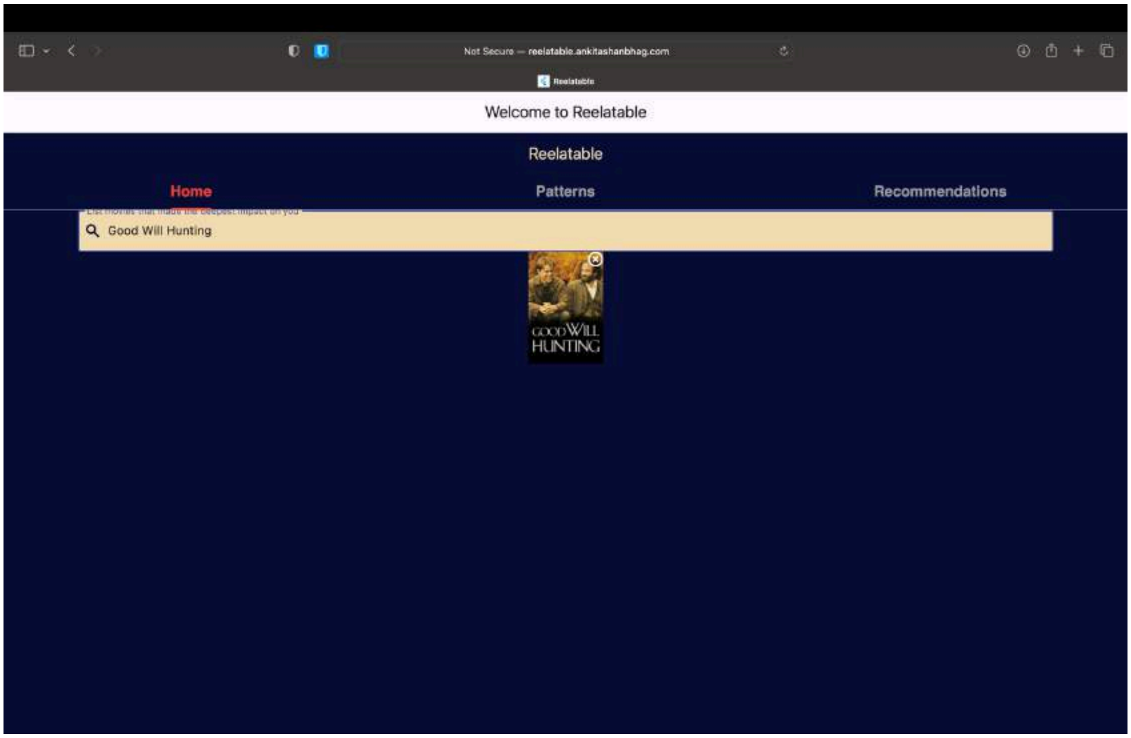
Appendix - Application Flow

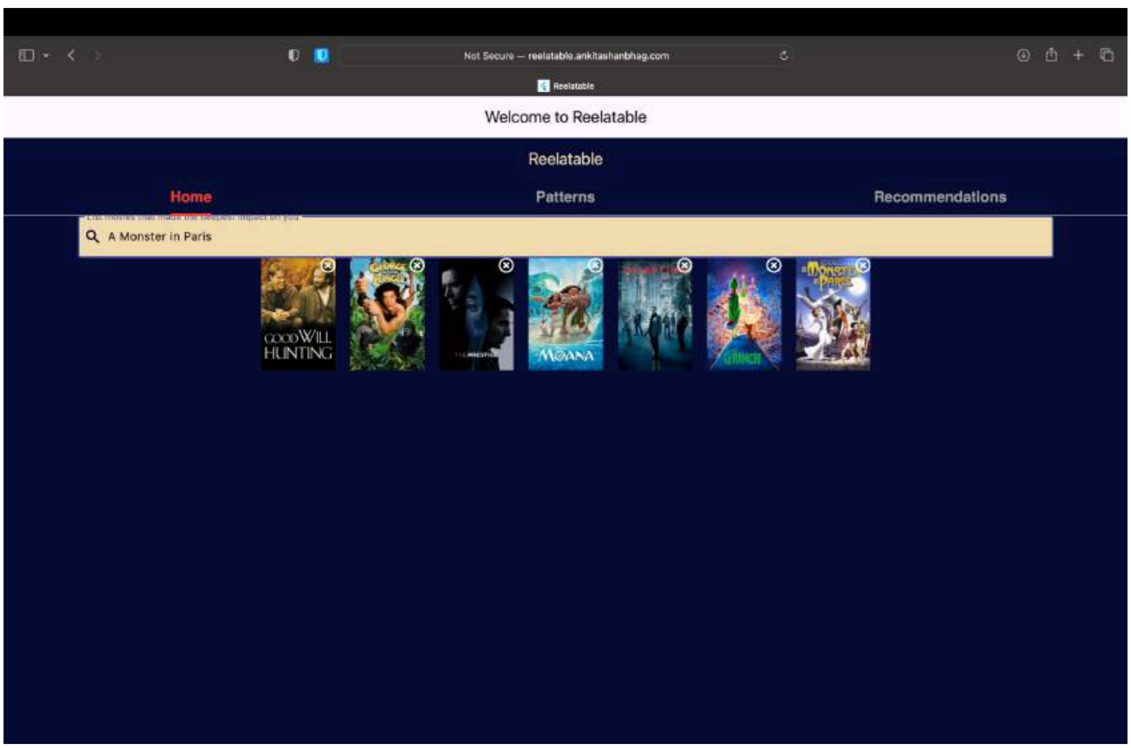
Step 1: The movie dropdown appears based on the user's input, allowing them to search for and select movies.



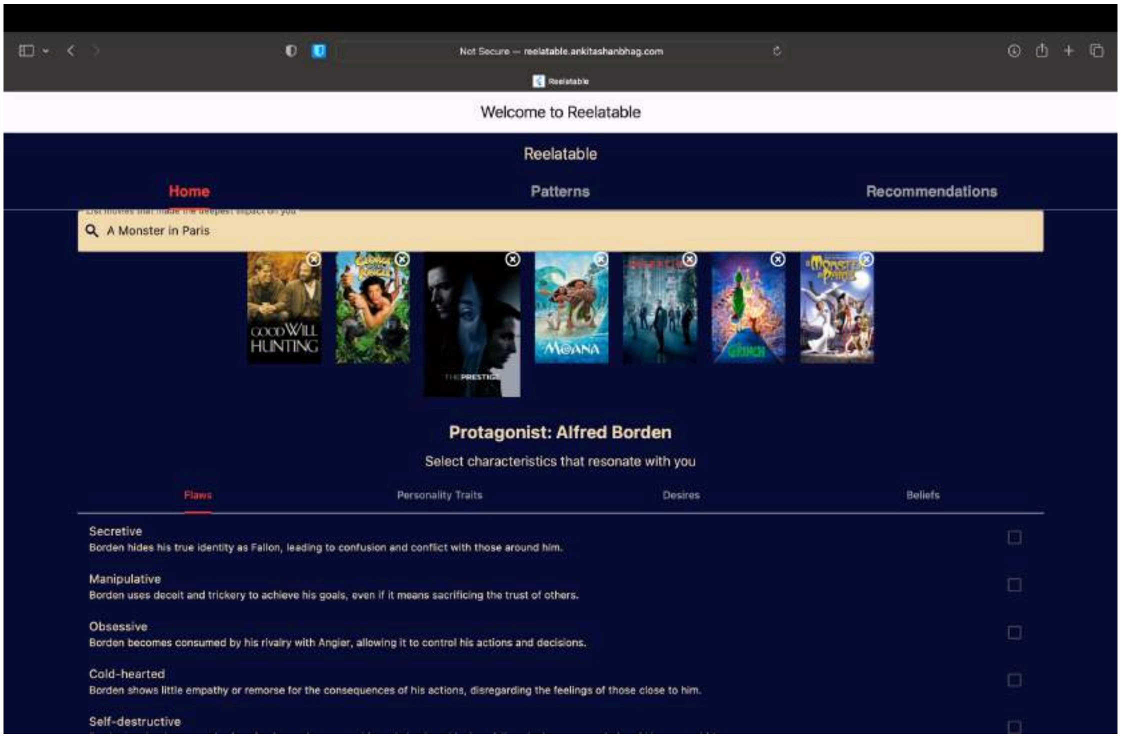


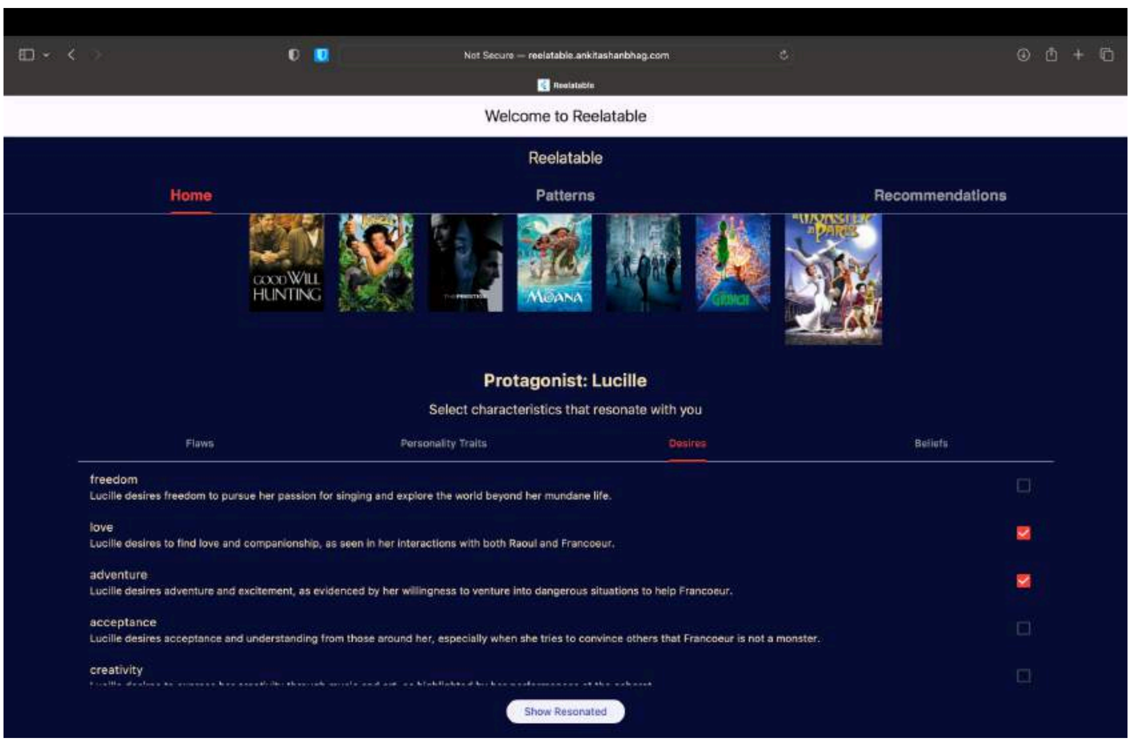
Step 2: Users can add and delete movies that resonated with them, creating a personalized list of favorite films.



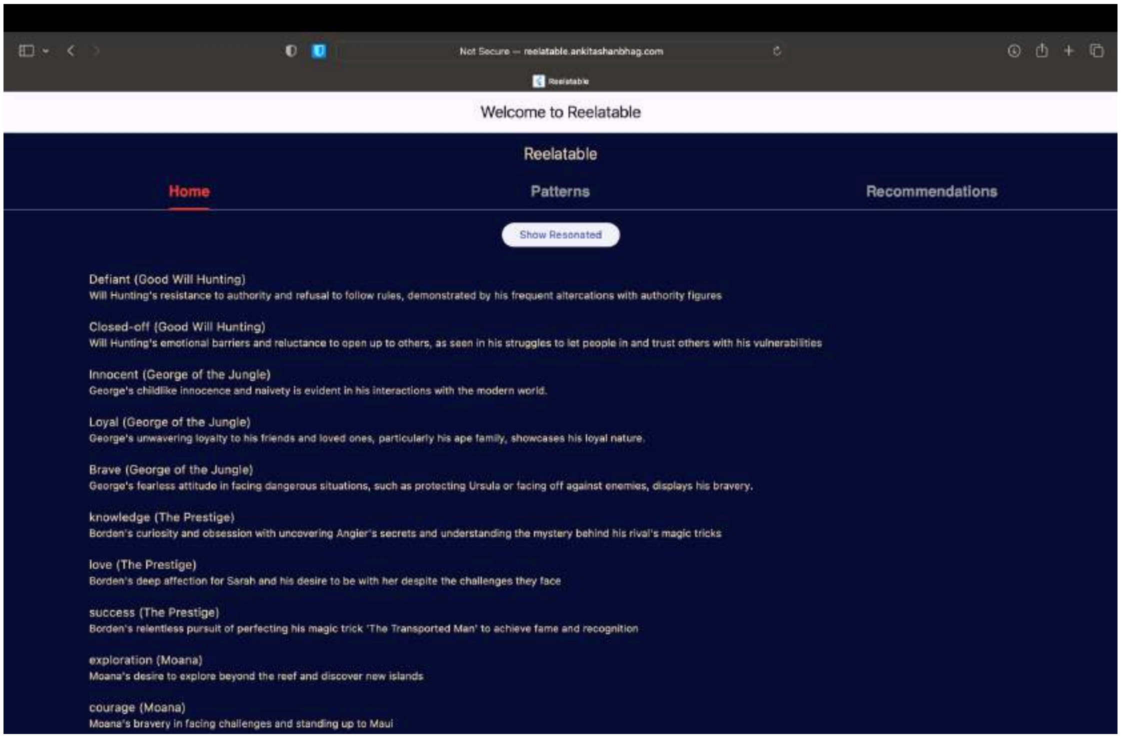


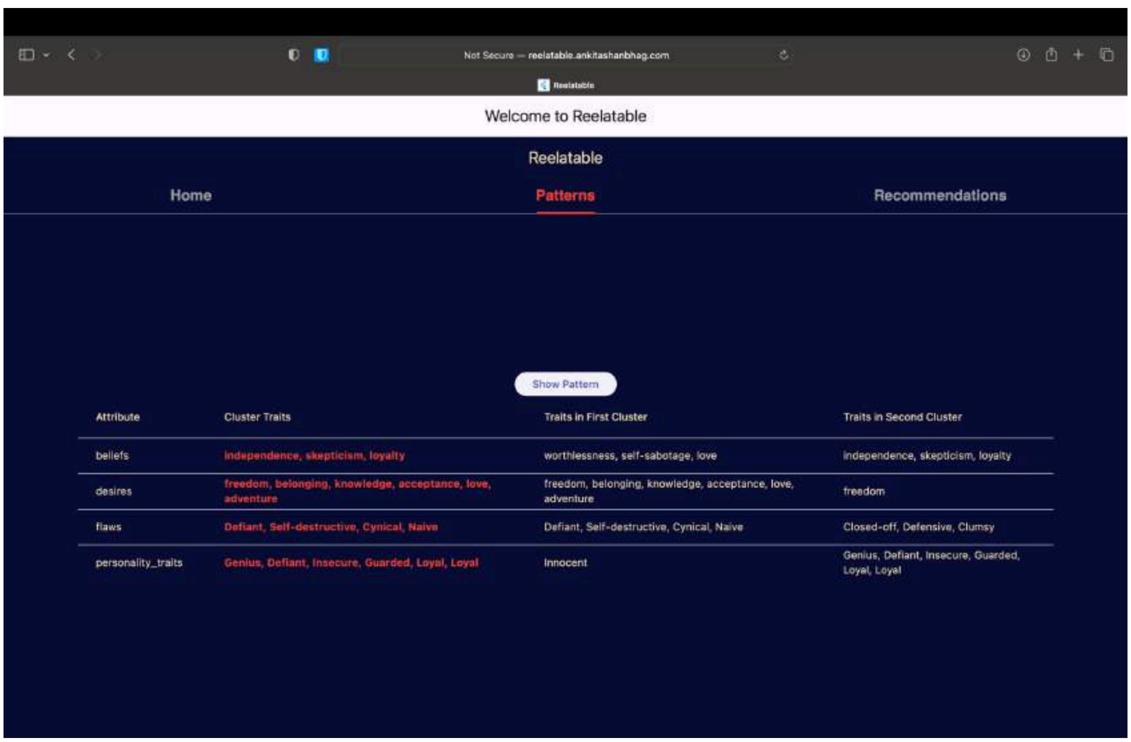
Step 3: Users can read about the characteristics of the movie protagonists and select the ones that resonated deeply with them.



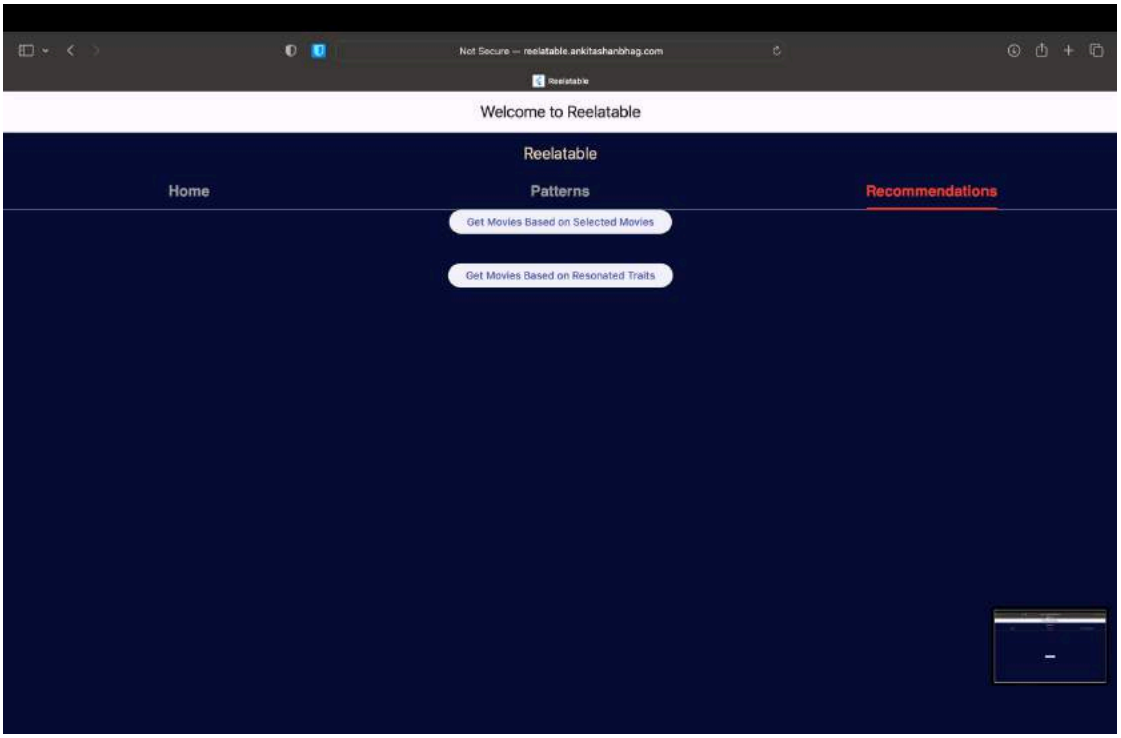


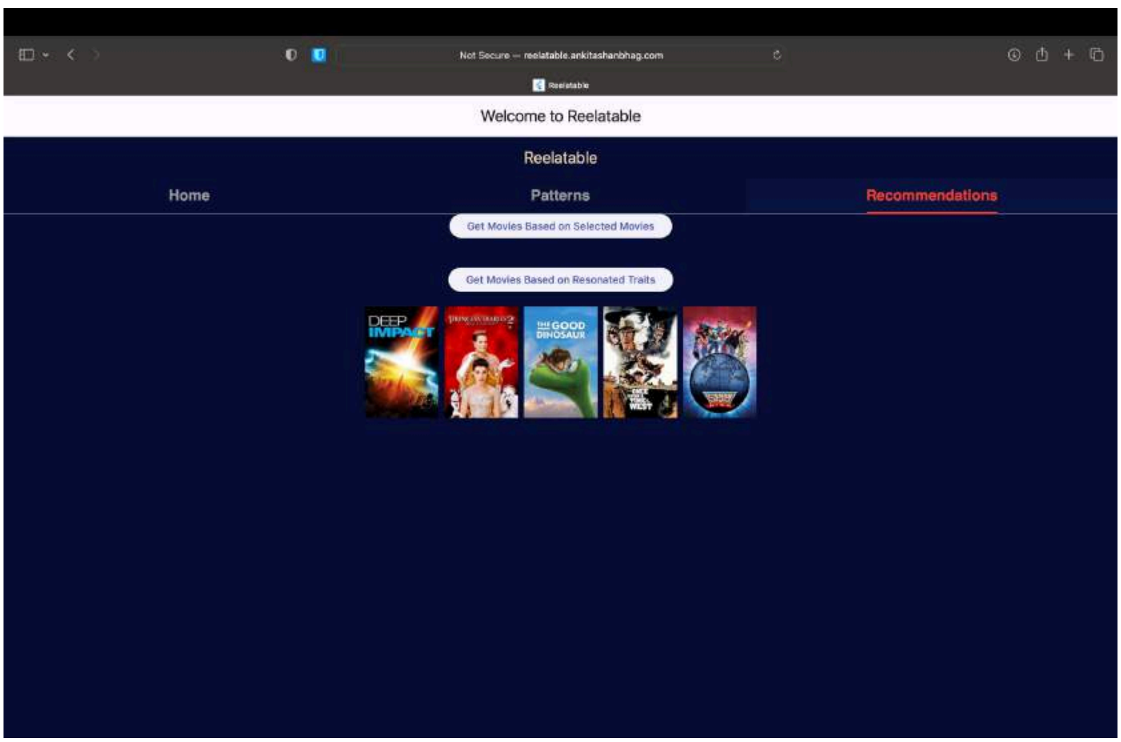
Step 4: Users can view the characteristics they selected as resonating with them, providing a visual representation of their preferences.



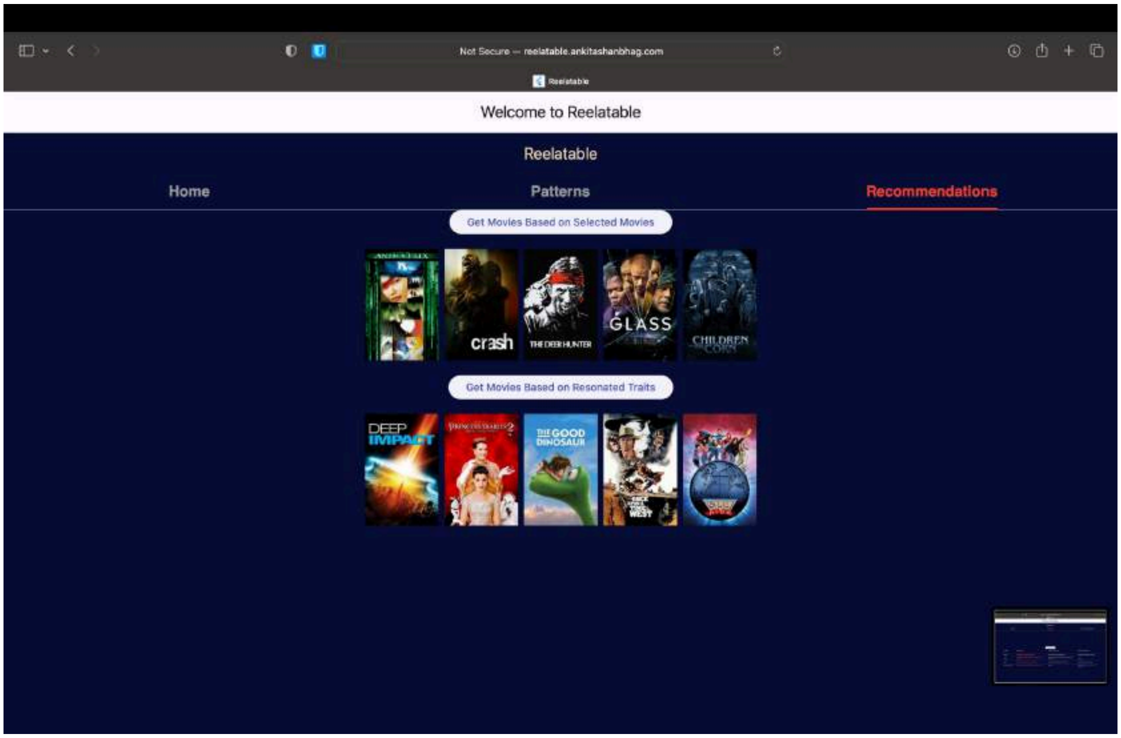


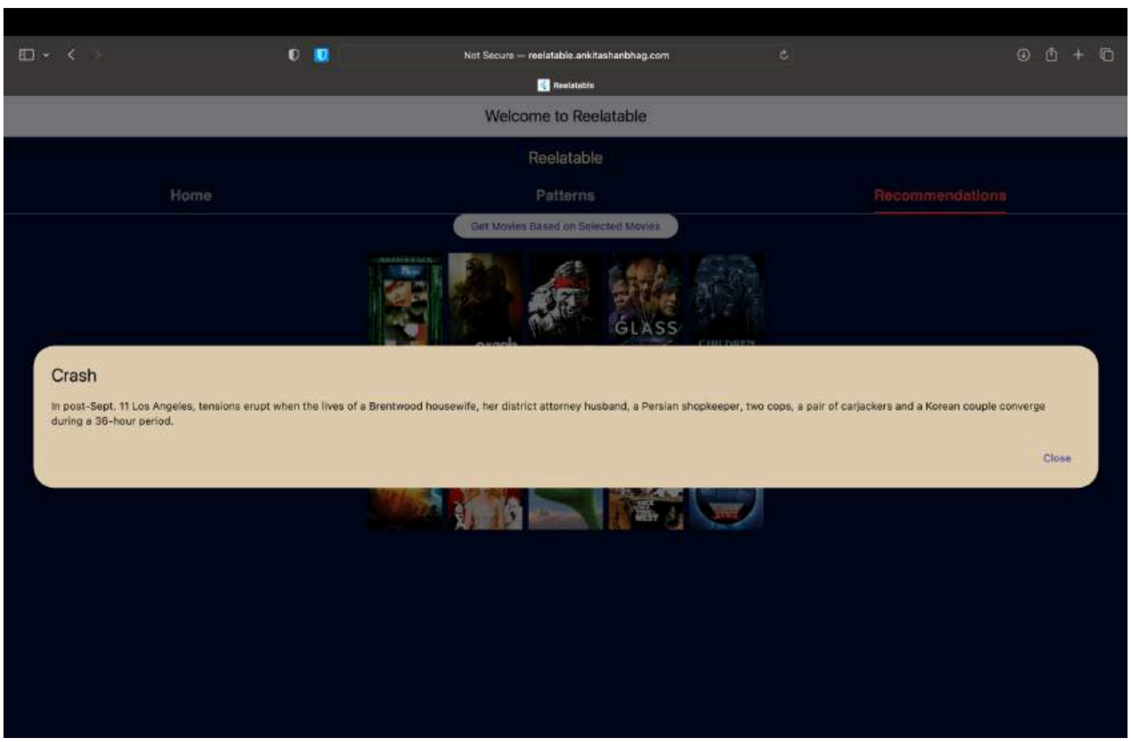
Step 5: Users can get movie recommendations based on the resonated characteristics they chose on the Home page, tailoring the suggestions to their personal inclinations.


Not Secure — reelatable.ankitashanbhag.com
Reelatable
Welcome to Reelatable
Reelatable
Home
Patterns
Recommendations
Get Movies Based on Selected Movies
Get Movies Based on Resonated Traits


Not Secure — reelatable.ankitashanbhag.com
Reelatable
Welcome to Reelatable
Reelatable
Home
Patterns
Recommendations
Get Movies Based on Selected Movies
Get Movies Based on Resonated Traits
DEEP IMPACT
THE GOOD DINOSAUR

Step 6: Users can get movie recommendations based on the movies they added as favorites on the Home page. They can click on the movie image to view the overview.





Appendix - Code

- Reelatable_Movie_Charateristics_Retrieval.ipynb
- reelatable_movie_recommendations
- https://github.com/AnkitaShanbhag30/flutter_application_reelatable
- https://github.com/AnkitaShanbhag30/flask_application_reelatable
- https://app.swaggerhub.com/apis/AnkitaSureshShanbhag/Reelatable/1.0.0

Appendix - Problem Statement, Vision and Value Proposition

Problem Statement

Scores of positive psychology practices like mood journaling, meditation, mindfulness, awe-walks, and reciprocal self-disclosure have been shown to be highly effective at enhancing mental well-being and beneficial for almost anyone. A number of such practices are aimed at improving self-awareness. However, a prevalent issue with these practices in the real world is the lack of adoption and engagement; they do not seamlessly integrate into people's existing lifestyles and interests.

Vision

Our vision is to weave positive psychology practices into daily life, enhancing well-being by reducing user friction. Leveraging the flywheel effect, each positive interaction propels further engagement, creating a sustainable cycle of mental health improvement.

Value Proposition - why users want this now

We offer a personalized movie recommendation engine that recommends movies that 'resonate' with users. This service not only enhances entertainment but also promotes reflection on personal values, seamlessly integrating introspection with enjoyment for deeper self-awareness and engagement.

Appendix - Product Management

Product Roadmap

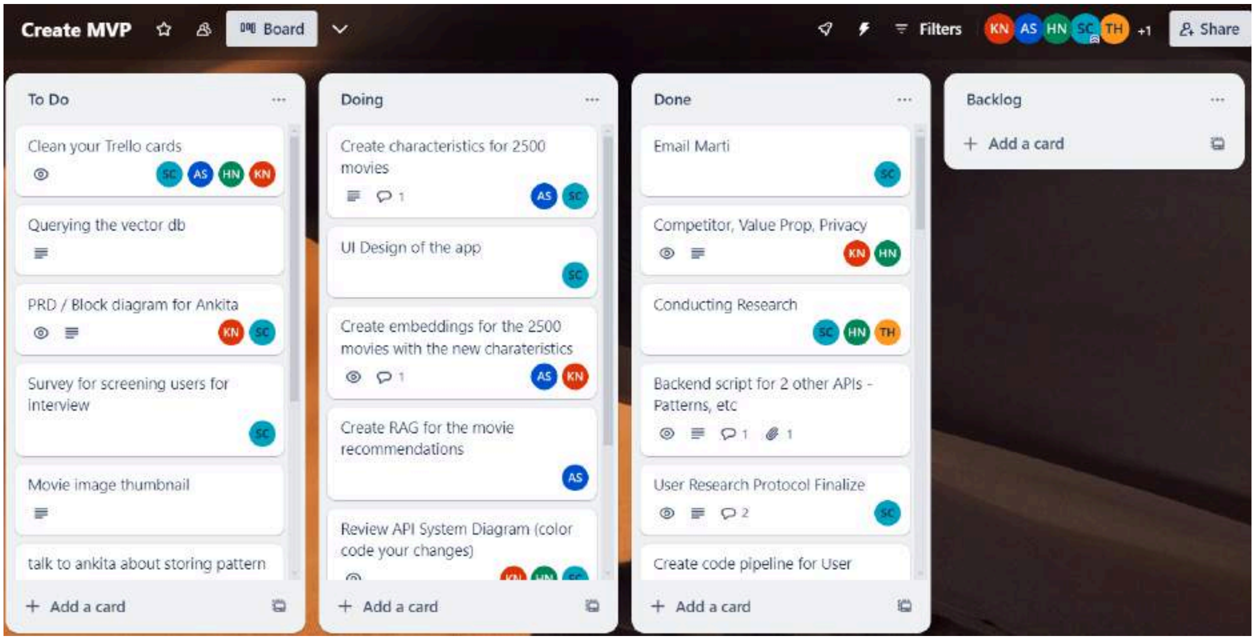
The product roadmap outlines the development phases for a movie recommendation app designed to help users understand themselves better. The roadmap is divided into four phases: Research, Product Management, Design, and Development. During the Research phase, the team conducted user research to understand user pain points and test different prototypes. Specific tasks include prompt engineering, user group identification, usability testing, and market analysis. The Product Management phase includes market analysis, user segmentation and JTBDs (Jobs To Be Done) analysis. During the Design phase, the team created wireframes, low-fidelity prototypes, and high-fidelity prototypes.

The Development phase was not scheduled until after the Design phase. During this phase, the team developed the backend, frontend, and machine learning model for the app. The process started with building the data infrastructure. Once the database of movies was ready, the backend and frontend development started parallelly. Lastly the entire team tested the entire app together.

				Mid Project report on 8th March			Done				
							In Progress				
							To Do				
		February			March			April			
Task		Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7	Week 8	Week 9	Week 10
Research	Prompt Engineering			Saurabh Chachra							
Research	User Group/Participants	Hrishikesh Naga	Hrishikesh Nagaraju								
Research	User Painpoints	Kinshuk Nigam	Kinshuk Nigam								
Research	User tests							Saurabh Chachra & Hrishikesh Nagaraju			
Research	Usability testing										
Product Manage	Market Analysis		Kinshuk Nigam								
Product Manage	User Segmentation and JTBDs										
Product Manage	Project Management				Kinshuk Nigam						
Design	Wireframing			Saurabh Chachra							
Design	Low Fidelity Prototype			Saurabh Chachra							
Design	High Fidelity Prototype						Saurabh Chachra				
Development	Setup Development Environment	Ankita Shanbhag	Ankita Shanbhag	Ankita Shanbhag							
	Prompt Engineering - Automation Script		Hrishikesh Nagaraju								
	Prompt Engineering - 10 movie sample loop script			Kinshuk Nigam							
	Pormpt Engineering script integration with db			Kinshuk Nigam							
Development	Backend Development				Kinshuk Nigam					Ankita Shanbhag	
Development	Frontend Development				Hrishikesh Nagaraju						
Development	ML Model Development			Ankita Shanbhag							
Development	Integration										
Development	Tuning and Testing								Ankita, Kinshuk, Saurabh & Hrishikesh		

Roadmap

We used Trello for project management. We used Sprint Planning meetings to decide on what to focus on. We had Scrums every Monday and Thursday to discuss our progress, clear roadblocks and discuss the next steps. The meetings would entail brainstorming of ideas too.



Appendix - Engineering

MVP

The plan was to test the prompts using code. We developed a script to run the prompts using the GPT 3.5 model and give us the characteristics of the movies. The MVP would loop for 10 movies. This test ensured that the basic functionality of using GPT to build the database works fine.

Frontend

Framework Selection:

During the early stages of development, several frameworks such as react, django and flutter were being considered for implementation for this project. After considering all the options and our goals for this project, we decided to go forward with Flutter for the following reasons:

- Cross-platform Compatibility: Flutter would enable us to create natively compiled applications for multiple platforms from a single codebase. This was a key factor in our selection because it would save us significant time compared to developing separate applications.

- Fast Development Cycle: Flutter's hot reload feature enabled us to iterate on our prototype rapidly, and quickly allowed us to experiment. We felt this would accelerate the development and allow for quicker feedback loops.
- Rich UI Capabilities: We wanted to use Flutter's widget library and customizable design elements to rapidly create a rich UI.

Understanding User Flow and Website Layout:

The initial step in the front-end development process involved understanding the user flow and mapping out the website layout. By analyzing user interactions and navigation paths, we identified key functionalities and content priorities to inform Reelatable's website structure.

Data Element Identification:

Next, we conducted a thorough analysis to identify all the data elements required to recreate the Figma design accurately. This involved defining the necessary data fields, structures, and relationships to ensure seamless integration with the front-end interface.

API Call Strategy:

Following the established user flow, we formulated a strategy to determine the number of API calls required and the optimal timing for their execution. By aligning API calls with user actions and workflow stages, we aimed to minimize latency and enhance the overall responsiveness of the application.

API Data Request and Response Design:

The next phase entailed designing the formats for API data requests and responses. This involved determining the specific data to include in API requests and defining the expected format for receiving data from the API.

Functional Prioritization over Appearance:

Throughout these initial stages of development, our primary focus was on prioritizing functional requirements and user flows before considering visual aesthetics and CSS styling. By emphasizing functionality and usability, we ensured that the core features of the application were not at risk.

API Request and Response Formats

API to get movie characteristics:

API Specs Request

Parameter	Type	Required	Description

movie_name	String	Yes	This is the input that we get from the user. User will provide the correct movie name.

API response format

Parameter	Type	Required	Description
movie_name	String	Yes	The name of the movie as provided in the request.
protagonist_name	String	Yes	The name of the movie's protagonist.
Characteristics	Array	Yes	An array of different characteristics types with their respective details.
├─ flaws	Array	Yes	A list of the protagonist's flaws with details.
└─ name	String	Yes	The name of the flaw.
└─ description	String	Yes	A description of the flaw.
├─ strengths	Array	Yes	A list of the protagonist's strengths with details.
└─ name	String	Yes	The name of the strength.
└─ description	String	Yes	A description of the strength.
├─ desires	Array	Yes	A list of the protagonist's desires with details.
└─ name	String	Yes	The name of the desire.
└─ description	String	Yes	A description of the desire.
├─ beliefs	Array	Yes	A list of the protagonist's beliefs with details.
└─ name	String	Yes	The name of the belief.
└─ description	String	Yes	A description of the belief.

API to get Patterns:

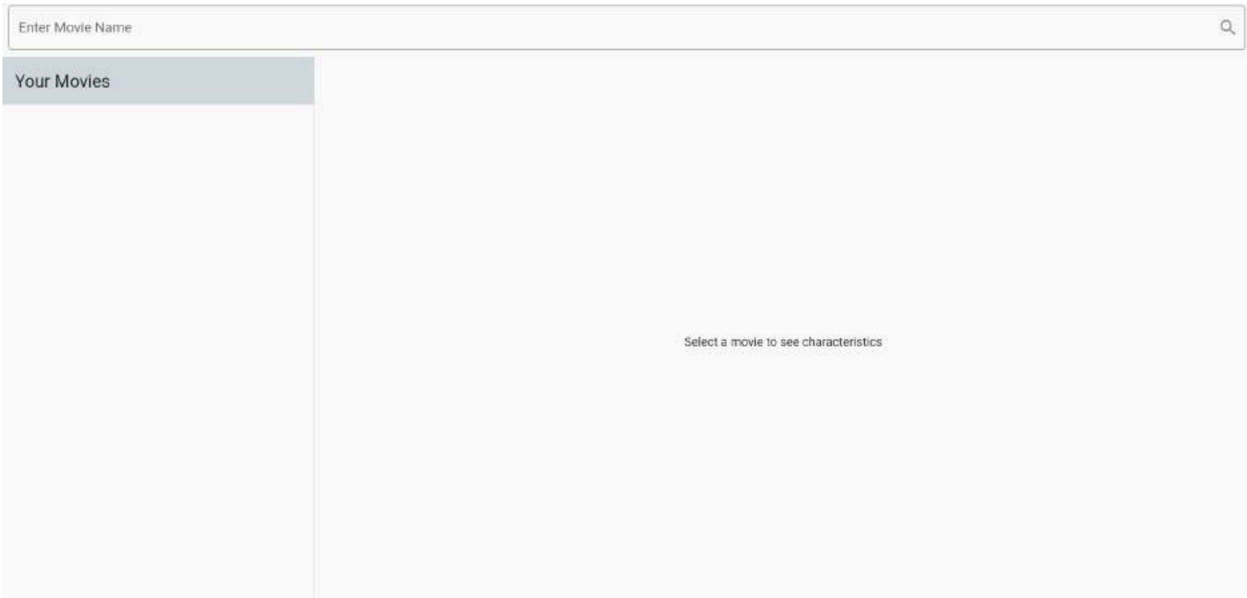
API Specs Request

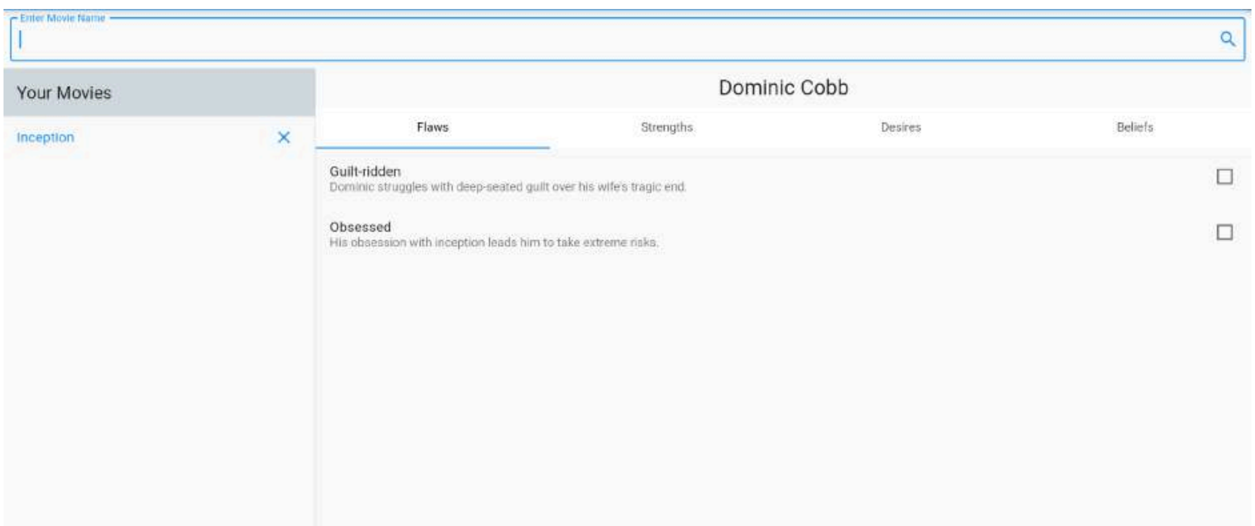
Parameter	Type	Required	Description
movie_name	Array	Yes	An array of movie names for which the user wants to find common patterns. Each element in the array is a string representing a movie name.

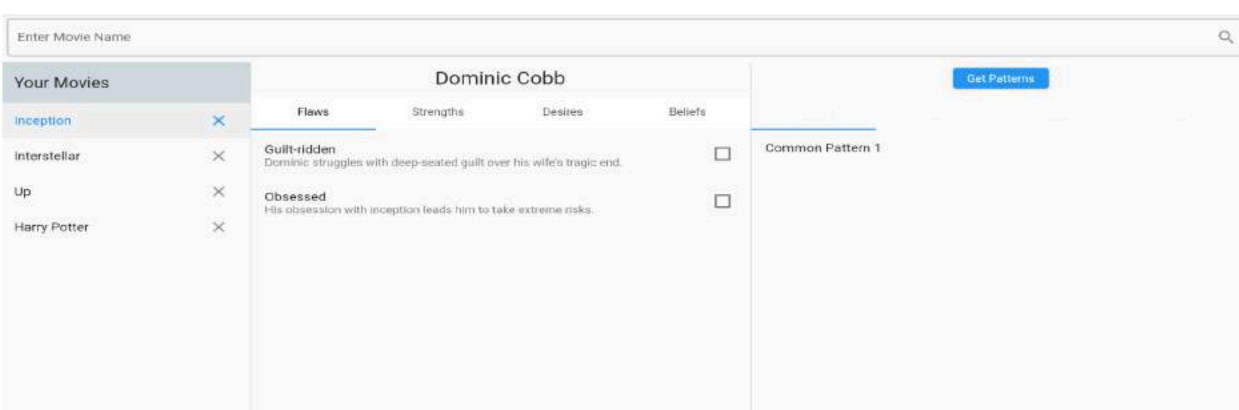
API Response format

Parameter	Type	Required	Description
patterns	Array	Yes	An array of patterns that are common across multiple movies based on characteristics.
├─ name	String	Yes	The name or title of the common pattern.
├─ characteristic	String	Yes	The type of characteristic the pattern relates to (e.g., "Flaw", "Strength").
└─ movies	Array	Yes	An array containing movies that exhibit the pattern.
├─ movie_name	String	Yes	The name of the movie contributing to the pattern.
├─ protagonist_name	String	Yes	The name of the protagonist in the movie.
└─ characteristic_name	String	Yes	The name of the specific characteristic from the movie that contributed to the pattern.

Initial iterations of frontend developments:



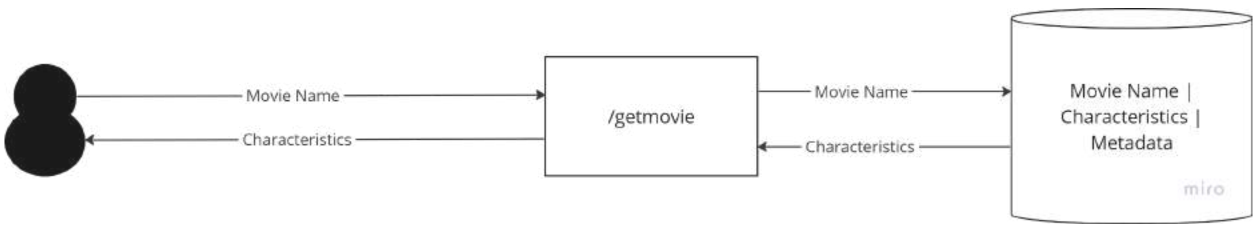




Backend: Intermediate server design to interact with the backend

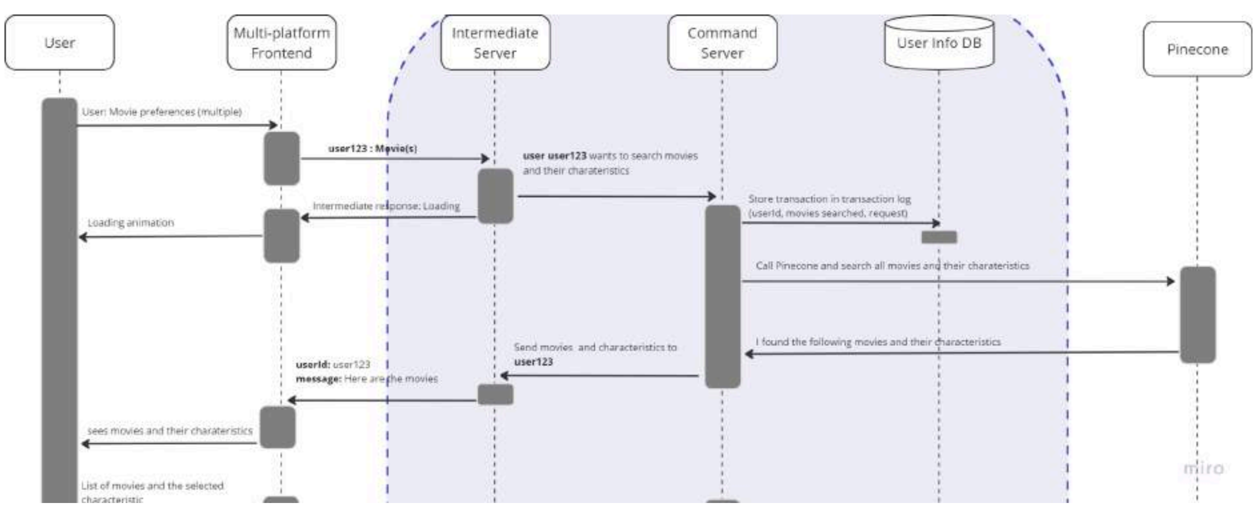
/getmovie: API to get the characteristics of a single movie
Input: Single Movie Name (JSON)
Output: Movie Characteristics (JSON)

While the pinecone database was getting ready with 2500 movie data, the API was built to serve user's requests to get movie characteristics based on movie name. To test the API, the backend used dummy data instead of pinecone data to avoid database dependency.



/getpattern:
Input: List of Movie Names (JSON)
Output: Common Characteristics between the Movie Names (JSON)

The API was used to get the common characteristics from a list of movies. These common characteristics are called patterns. Since the pinecone db wasnt ready, the API initially was made with dummy data.



Appendix - Economic and Business Analysis

Note to Reader:

Before proceeding with the economic and business analysis outlined in this section, I want to use this section to inform the reader of certain assumptions about the project. These assumptions were made because at the time when this analysis started the project was still in the development phase, where numerous aspects of the product were still being defined and explored. These assumptions about features, pricing strategies etc, were made to envision potential scenarios and evaluate their implications. I want to add that this analysis was conducted as a part of another class "Information Technology Economics, Strategy, and Policy" and while this work has helped us understand the product in its market, it did not dictate the actual product design or development decisions.

Furthermore, I want to clarify that certain considerations, such as the potential sale of user data to third parties, were explored solely for analytical purposes and do not reflect any actual business practices or ethical lapses we envision for this product. The inclusion of such considerations was intended to foster a thorough understanding of the economic landscape surrounding the project, without endorsing any unethical behavior.

By acknowledging the speculative nature of the analysis and the ethical boundaries maintained throughout the process, my aim was to provide transparency and context to the findings.

Economies of scale

Fixed Costs:

1. Development and maintenance of the platform/software.
2. Initial setup and ongoing maintenance of the movie database.
3. Salaries of team members (developers, data analysts, psychologists).
4. Marketing expenses.

Variable Costs:

1. Server and hosting expenses, which may scale with user activity.
2. fees for using large language models. which may scale with number of queries/prompts used

The cost curve would have high fixed costs, but have low marginal costs as more users start to use the platform. It will have high fixed cost, but average cost would go down with more Q. Thus, this platform shows strong economies of scale.

Supply side economies

- Economies of scope could arise from leveraging the existing infrastructure, technology, and expertise developed for understanding user preferences through movie data to offer additional related services or products. For instance, including analysis of other forms of media such as books, TV shows, or music and providing recommendations for the same.
- Since this is an online platform, there is no economy of density.

Network effects

Reelatable does not directly show network effects, that is single users learning more about themselves and getting movie recommendations does not benefit from more people using it. However, there are some nuanced ways in which there could be some weak to medium network effects:

1. **User-generated content**: As more users engage with the platform and input their movie preferences, the database of psychological characteristics linked to movies grows richer. This, in turn, enhances the accuracy and relevance of the insights provided to users, making the service more valuable to both existing and new users.
2. **Social sharing and referrals**: Users who have a positive experience with Reelatable are likely to share their insights or recommend the platform to their friends and social networks. This can lead to an increase in the user base, further enhancing the value of the service for everyone involved.

I argue that the value function is proportional to n(logn). As the number of customers (n) increases, the value of the service grows linearly due to the increased diversity of movie preferences and user data available for analysis. However, the relationship is not purely linear because of the diminishing marginal returns associated with each additional user. I argue that at some point, the marginal value of increased accuracy of the algo decreases as more users join the platform. So to adjust for this marginal decrease in utility, logn adjusts n.

Switching costs

I argue that the switching costs for Reelatable are very low. If users use Reelatable to just get movie recommendations, then it can be very easy to just switch to platforms such as Netflix, HBO Max, Amazon prime, where not only users will get recommendations, but also be able to actually stream the content. Additionally, there are no long-term contracts or commitments, since Reelatable just earns revenue through ads or subscriptions.

From the self understanding angle, there are also other competitors such as mood journaling apps or personality tests. However it is hard to gauge the switching costs since Reelatable offers a new way to learn more about yourself through movies.

Overall, since it is a new concept, I argue that switching costs are very low.

Potential Revenue Sources

1. End Users : They could be charged for premium features, personalized insights, or in-depth analysis of their movie preferences linked to psychological traits, personalized movie recommendations based on preferences and traits.
2. Advertisers: Given the potential advertisement model, businesses interested in targeting the platform's user demographic might pay for ad space.
3. Data Buyers: Companies interested in consumer behavior, preferences, and psychological insights might pay for anonymized user data.
4. Content Creators/Producers: Film studios or streaming services might be interested in insights to understand audience preferences better or for targeted marketing.
5. Affiliate Marketing: Earnings from referrals to movie streaming platforms or related merchandise.

Pricing Strategy Evaluation

(a) Uniform Pricing: Charging a flat rate for access to the platform might be simple and straightforward but may not cater to the varied value perceived by different users. I do not think this is a viable option since the value of using this platform is not very clear, at least before using it, and a flat fee would create another barrier to entry.

(b) Market Segmentation: Differentiating users based on their engagement level, preferences, or demographics could allow for tailored pricing strategies. We could target people who are movie enthusiasts, people who go to therapy, people who use mood journaling apps. Advanced analysis of personality traits could be offered to these segments.

(c) Versioning/Volume-based Pricing: Offering different tiers of service (basic, premium) again by providing more analysis of personality traits, more movies, advanced recommendation with reasoning, as to why the recommended movie is best suited for the individual.

(d) Personalized Pricing: Tailoring prices to individual users based on their usage patterns and frequency of use could be an option. But this is unlikely as it could raise perceived fairness issues.

(e) Bundle Pricing: Offering bundles that include related services (e.g., movie streaming) would improve value perception but could complicate the value proposition and has very high fixed costs.

(f) Decoy Pricing: Not applicable. Could it be done if we had multiple versions of the product, but is too far fetched.

(g) Price Skimming/Penetration Pricing: Skimming is not applicable since this business low initial prices to gain market share (penetration) could be effective strategies depending on market readiness and competition but risk alienating early adopters or devaluing the service.

(h) Pricing at Zero (Free): Most feasible. Offering the service for free, supported by ads or data monetization, could maximize user base growth. This might affect user experience due to ads, but seems like the most feasible option, since the barrier to entry is minimal.

(i) Price Conditioning: Not very applicable at this stage, hard to increase willingness to pay when the switching cost is very low.

(j) Dynamic/Algorithmic Pricing: Not very applicable, since offering does not really depend on supply or demand. There is no element of competition or scarcity. High risk of being perceived as unfair, and it is unclear on what basis the algorithm will decide prices.

Switching Costs and Lock-in Strategies

Reelatable would be a first mover into this market. Given the low switching costs, some possible strategies:

1. Community Building: Foster a user community around shared movie experiences and insights, increasing the platform's value with network effects. This would add a social aspect to movies along with sharing our personality traits with friends. Similar to what Letterboxd does.
2. Integration: Integrate with other services such as social media to embed Reelatable deeply into users' digital ecosystems.
3. Data Accumulation: The more a user interacts with the platform, the more personalized and accurate the insights become, creating a natural lock-in. This would also enhance the unique value proposition, where the blend of movie preferences and psychological insights offer a service that's hard to replicate.

Final Pricing Strategy

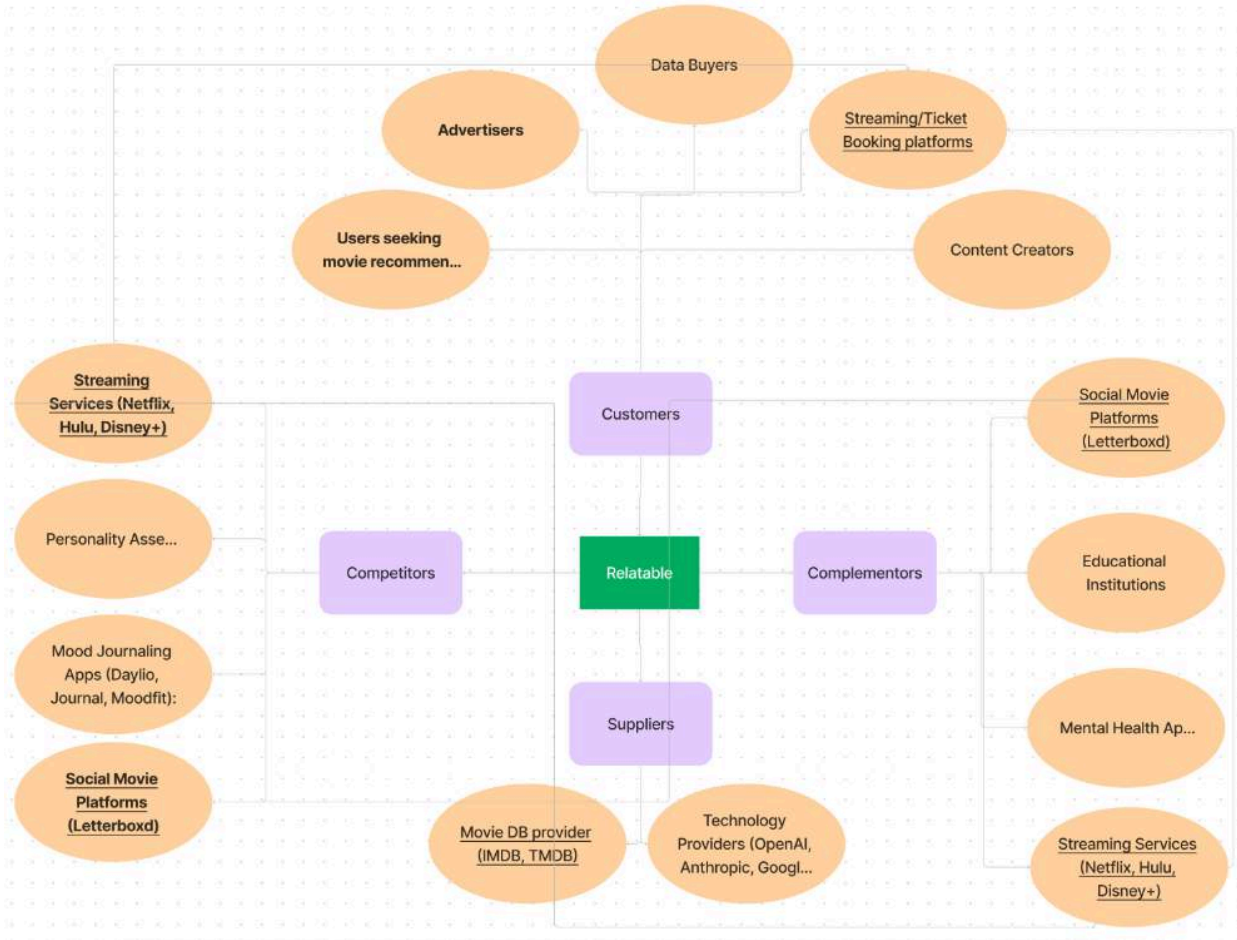
Considering Reelatable's unique position, a suitable pricing strategy would be:

The Freemium Model: Offer basic insights and functionalities for free to attract a broad user base, supported by ads, with premium features available for a subscription fee. This balances the need for a wide user base (for data and network effects) with revenue generation. This involves pricing at Zero and Versioning. The premium tier could offer more in-depth insights, and ad-free experiences to cater to varying user needs and willingness to pay.

This is a new space so there are no direct competitors, however if we consider Letterboxd as the closest competitor, its pricing strategy is similar in that it has a free to use platform which it monetizes with ads, and also a premium plan with additional features.

FYI: Letterboxd is a social networking platform where users can track, rate, review, and discuss films, as well as follow friends and other users to see their film activity.

Value Network



Customers

- Individual Users: Individuals seeking self-awareness through movie preferences.
- Advertisers: Businesses targeting the platform's user demographic.
- Data Buyers: Entities interested in psychological insights and consumer behavior.
- Content Creators (directors, writers, actors, etc.): Film studios/streaming services seeking audience insights. This could be considered in the data buyers category
- Streaming platforms or Ticket Booking sites (Netflix, HBO Max, Fandango etc): Earnings from referrals to movie streaming platforms or related merchandise or ticket booking sites incase it is a new movie

Suppliers

- Movie Database Providers (IMDB, TMDB): Sources of comprehensive movie metadata and content.
- Technology Providers (OpenAI, Anthropic, Google BARD, Pinecone): Companies offering tech infrastructure, LLMs, and other tech solutions.

Competitors

- Streaming Services (Netflix, Hulu, Disney+): Direct competitors in movie recommendations, even though without the psychological analysis aspect.
- Personality Assessment Tools (Truity, HIGH5 Test, DiSC, Big 5): Online platforms offering insights into personal characteristics.
- Social Movie Platforms (Letterboxd): Communities centered around movie watching and reviews.
- Mood Journaling Apps (Daylio, Journal, Moodfit): Apps that provide insights into users' moods and preferences, potentially offering an alternative path to self-awareness.

Complementors

- Social Media/movie Platforms (Facebook, Twitter, Letterboxd): Can enhance user engagement through sharing and discussions.
- Educational Institutions: Could use the platform as a tool for studies in psychology, film studies, and media.
- Mental Health Apps (Calm, Headspace, Moodfit etc.): Could complement the self-awareness aspect by providing deeper psychological insights or therapeutic advice.

Existing Rivals: No existing rival as this is a new space, and a novel approach mixing Movie preferences, recommendations and personality insights.

Potential New Entrants:

- Social Movie Platforms: Platforms like Letterboxd or IMDb where users rate, review, and discuss movies. These platforms could potentially integrate features similar to "Reelatable" to enhance their offerings, making them direct competitors.
- Video Streaming Services (Netflix, Hulu, Disney+, etc.): These platforms already offer sophisticated movie recommendation systems based on user preferences and viewing history. They could potentially integrate more personal insights into their recommendation algorithms, making them substitutes.
- Tech Giants with Recommendation Algorithms: Companies like Google or Amazon could potentially enter this niche by leveraging their vast data and sophisticated

algorithms to offer similar insights based on users' entertainment choices, including movies, books, music, etc. They even have their own LLM’s.

Substitute Products:

- Personal Insight Platforms: These include a range of online services offering personality tests, mood tracking, and psychological assessments, such as Myers-Briggs type indicators or the Big Five personality tests. They provide users insights into their personalities without the unique angle of using movie preferences.
- Mood Journaling Apps (Daylio, Journal, Moodfit): Apps that provide insights into users' moods and preferences, potentially offering an alternative path to self-awareness.
- Content Discovery Platforms (YouTube, TikTok): These platforms could serve as substitutes if they start offering more personalized content recommendations based on deep psychological profiling, moving beyond mere entertainment preferences.

Value Derivation from Complementors

- Social Media/movie Platforms: By enabling users to share their movie-based psychological insights on social media, Reelatable can gain virality, increasing its user base and enhancing its data pool for better analytics.
- Educational Institutions: Collaboration with these institutions can validate the platform's psychological models and increase its credibility, potentially leading to a broader user base and enhanced data quality.
- Mental Health Apps: Integration or partnerships with these apps can offer users a more comprehensive self-awareness toolkit, enhancing user retention and the value proposition of Reelatable.

Impact on the Value Network

Adds Value:

- Users get better insights about themselves and better recommendations.
- Advertisers get ad space to generate more revenue
- Data buyers get access to psychological insights and consumer behavior, for research, ads etc
- Streaming platforms or Ticket Booking sites get value as more users watch recommended content on their sites or buy tickets, creating an additional funnel.
- Social media/movie apps get more user engagement when users share content from Reelatable on their platforms

- Movie database providers as well as tech providers such OpenAI or Pinecone get additional revenue

Subtracts value:

- Streaming platforms: may lose certain engagement as users may rely less on their recommendations. Since Reelatable will recommended the movie and also where to watch it, if this funnel is big enough, streaming services may have to compete to be featured on Reelatable
- Movie Database Providers or review sites (IMDB, TMDB, Rotten Tomatoes): Users could rely less on generic ratings or scores provided by these sites, instead opting for personalized recommendations from Reelatable, lowering engagement on their platforms.
- Personality Assessment Tools: May lose engagement as people are more attracted to Reelatables more engaging way to show personal insights

Regulations

Structural Regulations

- **Current Application:** As a new entrant providing insights into user psychology through movie preferences, Reelatable might not yet be directly subjected to stringent structural regulations. However, its use of consumer data means it must navigate existing digital marketplace frameworks and data protection laws.
- **Potential Future Regulation:** As Reelatable grows, if it gains significant market share or if its business practices are deemed to limit competition, it could face increased scrutiny under antitrust laws. Additionally, expansion into areas like content curation or direct partnerships with streaming platforms might subject it to media and content distribution regulations.

Behavioral Regulations

- **Data Protection and Privacy:** Reelatable is may be subject to data protection regulations like GDPR in the EU or CCPA in California due to its potential processing of personal data. Compliance with these regulations is crucial to ensure user trust and avoid penalties.
- **Content Regulation:** If Reelatable starts curating content or becomes more involved in content recommendations, it might need to comply with regulations governing content rating systems, copyright laws, and potentially platform liability rules.

- **AI and Algorithmic Transparency:** The use of LLMs and AI algorithms places Reelatable within the scope of emerging regulations focused on algorithmic accountability, ethical AI use, and transparency. This includes ensuring that algorithms do not perpetuate biases or infringe on user rights.

Benefits and Harms from Government Regulation

Benefits:

- **Consumer Trust:** Compliance with stringent data privacy regulations reassures users that their personal information and psychological insights derived from movie preferences are handled securely and ethically. For instance, adhering to GDPR principles of data minimization, purpose limitation, and user consent can enhance Reelatable's reputation as a trustworthy platform.
- **Fair Competition:** Antitrust and fair competition regulations can prevent market dominance by larger entities, ensuring a level playing field for Reelatable.

Harms:

- **Financial Burden:** The costs associated with ensuring compliance with a broad spectrum of regulations, from data protection to AI ethics, can be substantial, especially for a startup or a small business. These costs might include legal fees, technology investments to ensure privacy compliance, and ongoing costs related to audits and regulatory reporting.
- **Innovation Constraints:** Overly prescriptive regulations might limit the scope of data Reelatable can analyze or the types of psychological insights it can offer, stifling innovation. For example, stringent consent requirements might reduce the amount of data available for analysis, limiting the depth and accuracy of insights Reelatable can generate.
- **Barrier to New Markets:** Regulatory complexity and variability across jurisdictions can make it challenging and costly for Reelatable to enter new markets. Each new market might require significant adjustments to comply with local regulations, delaying launches and limiting growth opportunities.

Neutrality

In the context of Reelatable, neutrality pertains to the impartiality and unbiased nature of its service offerings. It may be in terms of how it analyzes user data to generate psychological insights or how it recommends content based on these insights. While Reelatable isn't a traditional platform in the sense of a social network or a marketplace, it functions as a platform where users' movie preferences are linked with psychological traits to provide personalized insights and movie recommendations.

- **Content Recommendations:** Ensuring that movie recommendations are solely based on the algorithm's understanding of user preferences and psychological insights, without external biases or influences, is crucial. Avoiding partnerships or agreements that prioritize certain movies, genres, or studios over others ensures that recommendations remain unbiased. Transparency in the recommendation algorithms and criteria used for suggestions can further reinforce neutrality.
- **Psychological Insights:** The analysis and interpretation of users' preferences to infer psychological traits must be done without any bias towards particular outcomes. This ensures that the insights provided are genuinely reflective of the user's preferences and not influenced by external factors. Continuously auditing algorithms for biases can help maintain neutrality in psychological assessments. Engaging independent experts for periodic reviews could also enhance credibility.