AndChill Application Design

Team AndChill: Rosemond Dorleans, John Parsons, Arman Rahman, Hugo Zul

Motivation

AndChill is a dating application on the web platform that helps users develop meaningful romantic relationships. When a user creates an AndChill account they enter a network of relationship seekers who value partners who have common interests as them.

As part of the AndChill experience, when displayed with potential matches, a user has access to that potential match's name, location, bio and interests. These interests are supplemented by suggested activities which they might enjoy and are further magnified by the absence of their potential match's picture. Upon a mutual liking, users are connected via a chat and are shown each other's pictures.

We believe this advances current online dating applications by our attempt to foster genuine relationships, something often overlooked in the market. Furthermore our app's design is such that it satisfies the following purposes:

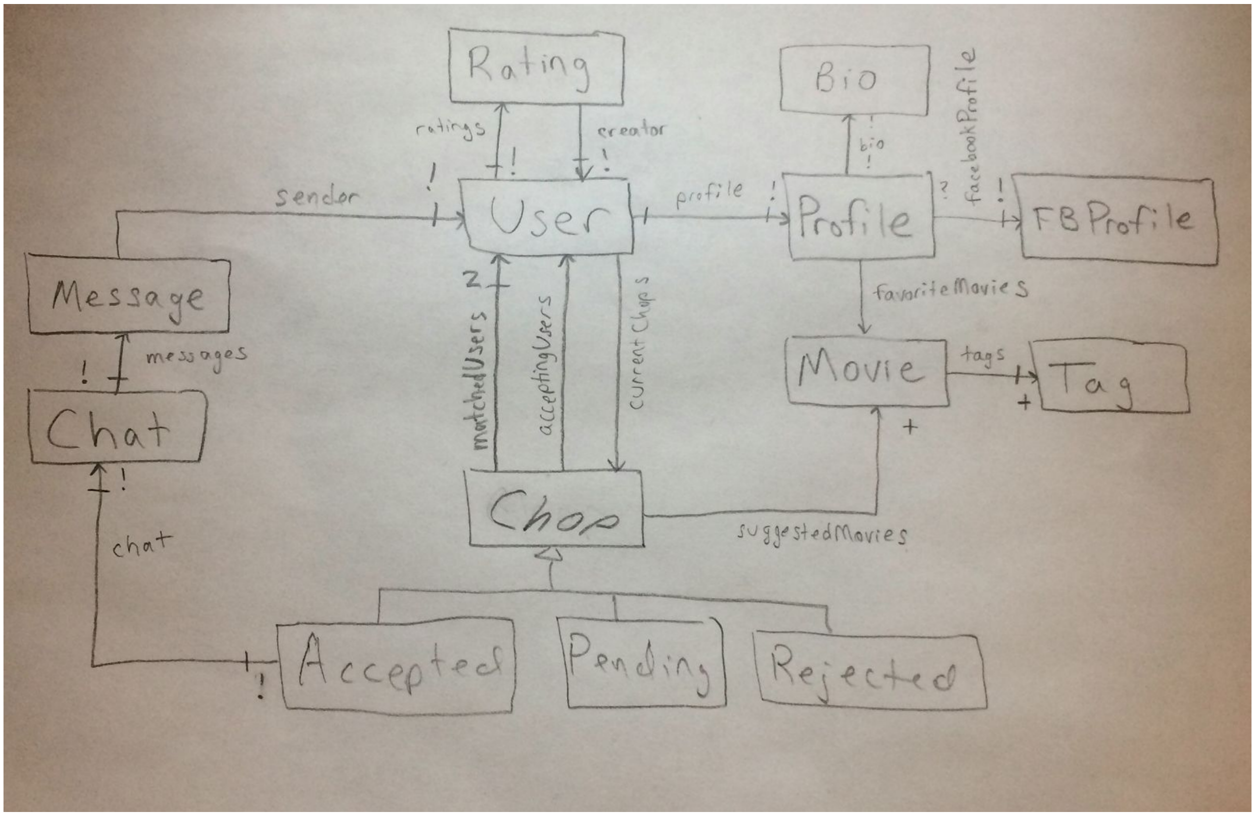
- Allow users to find potential love matches based on mutual interests instead of attraction alone
    - The visual aspect in most other dating apps (e.g. tinder) are overly emphasized which results in an unsatisfying user experience as human relationships are founded on more than physical attraction. We provide our users with the ability to find these common interests before meeting in person.
- Allow users to foster potential relationships by communicating with their compatible matches
    - Before going on a date, relationship seekers naturally want to be able to communicate with their date beforehand to reaffirm their compatibility.
- Make the dating experience as comfortable, safe, and fun as possible
    - Given the anonymity of the internet, users want to be sure they're speaking to real people free of malicious intentions.
- Streamline the process of arranging a first date by suggesting mutually agreeable movies to watch
    - People are naturally indecisive and would benefit from suggestions of what to do with their dates.

Concepts

- **Chop:** A Chop is a "chill opportunity:" an offer to begin communicating with another user with similar interests that contains basic information about the other user, but no picture. Two users must "accept" Chops for each other for the Chop to be completed.

  - **Operational principle**: If user A accepts a Chop for user B and user B accepts a Chop for user A, then users A and B have demonstrated interest in spending time together and should be allowed to communicate.
  - **Purpose**: Allow users to find potential love matches based on mutual interests instead of attraction alone
- **Chat:** A Chat is an instant messaging session between two users who have mutually accepted a Chop.
  - **Operational principle:** Two users A and B use a Chat to determine if they are indeed mutually compatible and to arrange a meet-up if so.
  - **Purpose:** Allow users to foster potential relationships by communicating with their compatible matches
- **Rating**: A Rating is a grade of some user A submitted by another user B who has a mutually accepted Chop with A. These Ratings are averaged and shown to the entire network of users.
  - **Operational principle:** All Ratings a user A receives are averaged and displayed to other users.
  - **Purpose**: Make the dating experience as comfortable, safe, and fun as possible
- **Suggestion**: A Suggestion is an activity or list of activities that two users A and B would both enjoy doing based on the information they supply to the application.
  - **Operational principle:** If users A and B receive a Suggestion S, then it is likely that A and B would enjoy watching pieces of S together.
  - **Purpose**: Streamline the process of arranging a first date by suggesting mutually agreeable movies to watch

Data Model



Definitions

- A Chop c is available to be shown to a User u if (u, c) is in currentChops
- A FBProfile represents a User's profile on Facebook
- Tags are items that describe a Movie – for example (but not limited to), a genre, director, or lead actor
- A User u has accepted a Chop c if (c, u) is in acceptingUsers
- A Chop c has been mutually accepted if for both users u, v such that (c, u) and (c, v) are in matchedUsers, (c, u) and (c, v) are in acceptingUsers

Textual Constraints

- For any User u, there must be ≤ 10 Chops c such that (u, c) is in currentChops
- If a Chop is mutually accepted, then it must be an Accepted Chop
- A Rejected Chop may not be in a User's currentChops or acceptedChops
- If a Chop ever becomes Rejected, it may not become Pending or Accepted
- Once a Chat has been assigned to an Accepted Chop, it may not be reassigned (the Chop may be un-assigned, though)
- No two Chops may have the same two Users in matchedUsers

- For some Chop c and User u, (c, u) may only be in acceptingUsers and (u, c) in currentChops if (c, u) is in matchedUsers (Users must be matched in a Chop to view or accept it)

Insights

- There is no need to explicitly store Rejected Chops; it is possible to filter matchedUsers to find them instead, which reduces the required number of relations
- As the data model is written, if a Chop goes from Accepted to Rejected, then neither User may access the Chat – this is good for security, but the data model also supports storing the Chat on the server with no Chop for the sake of completeness
- The usefulness of Tags (they will help produce suggestedMovies) is not immediately apparent from the data model; these will act behind-the-scenes in the actual application, so it makes some sense that they are not central
- The Movies for a Chop in suggestedMovies should not be static: if there is an update that allows for better suggestions, then those suggestions should be permitted by the data model
- If a user decides to un-match from a Chop, the Chop would become rejected but a Chat log of the conversation would still exist (it just wouldn't be associated with a Chop anymore). This is good for record preservation

Security Concerns

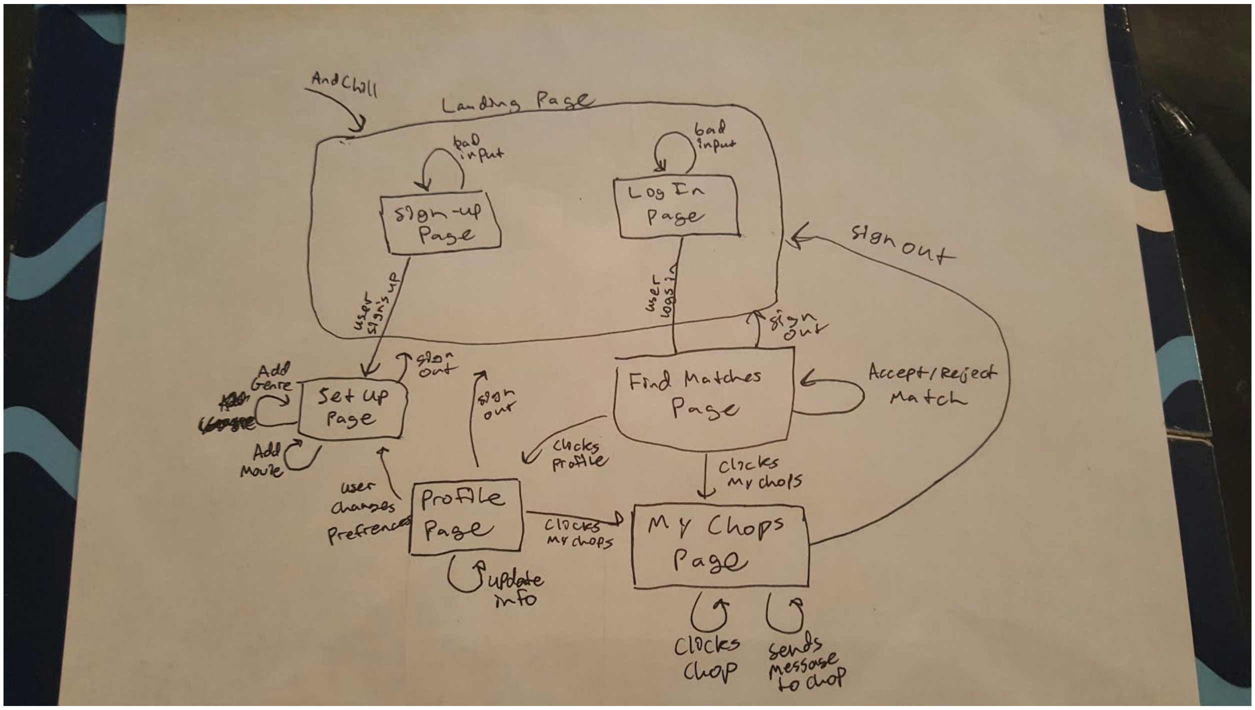
The following threats could affect our web application:

- Network eavesdropping occurs between the browser and Web server to capture client credentials.
- An attacker captures an authentication cookie to spoof identity.
- SQL injection occurs, enabling an attacker to exploit an input validation vulnerability to execute commands in the database and thereby access and/or modify data in such a way to get more matches.

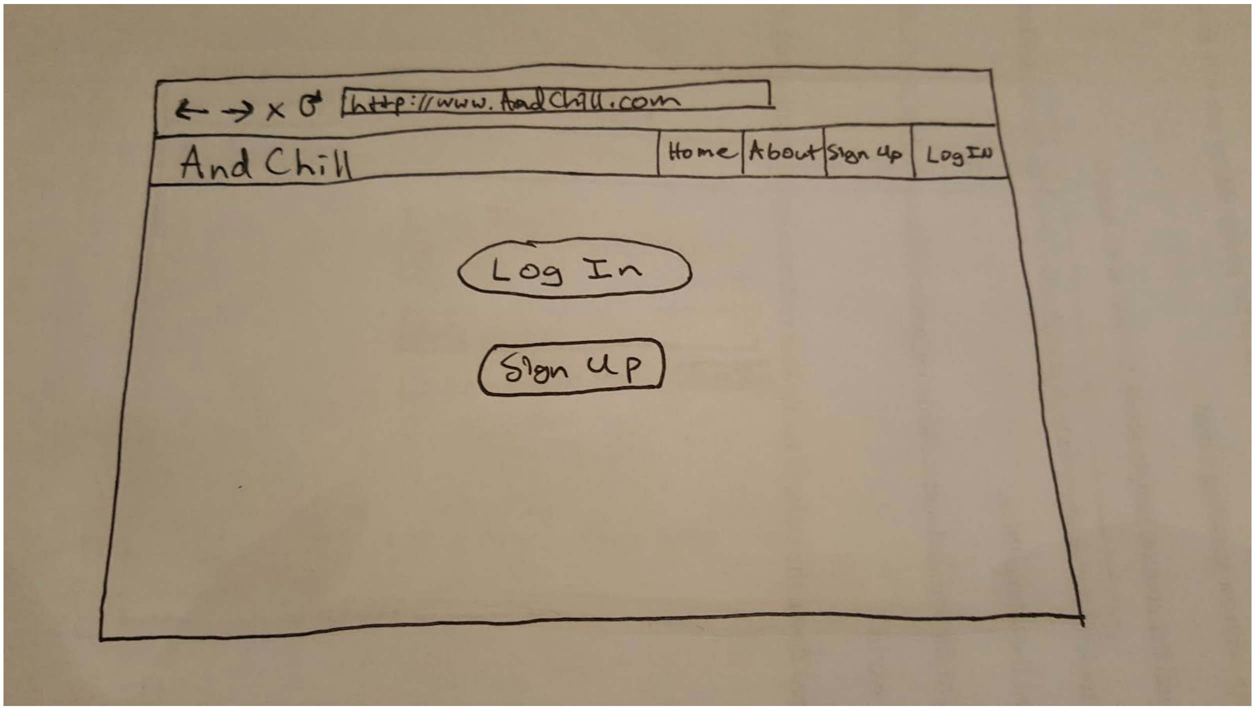
The main security concern lies in protecting the information of our users. Conversations should only be seen by the receiver and the sender. Furthermore, users should only be able to message users they have been matched to. You can only see a user's bio, interests, age, and distance from you if they are a one of your chops. Pictures can only be seen once the users are matched. In addition, users should not be able to modify their own ratings, or that of others. We should ensure that a user is not able to access the account information of another user.

In order to ensure this, we can encrypt users' information and conversations to prevent XSS and CSRF attacks. We plan on doing server side checks to ensure that only certain users can do certain things using libraries like passport.js. For web attacks, we'll be using CSRF tokens in all our forms and we can avoid XSS by sanitizing user inputs in chat boxes.

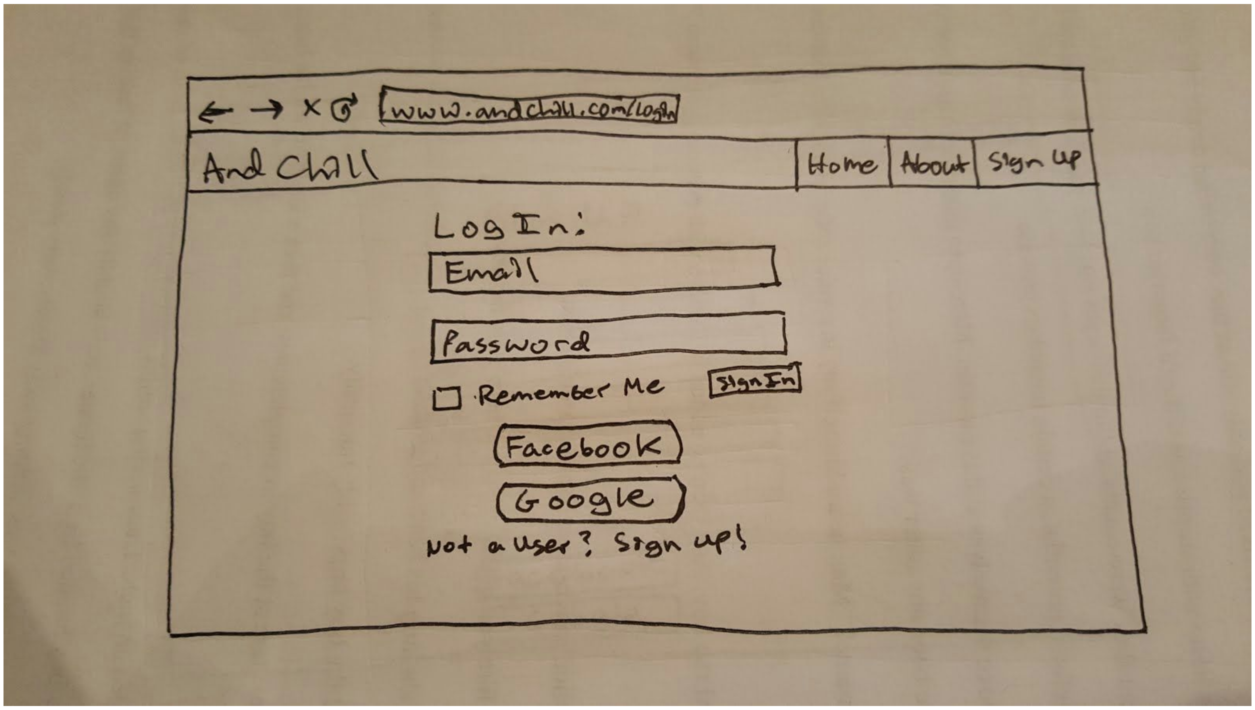
User Interface



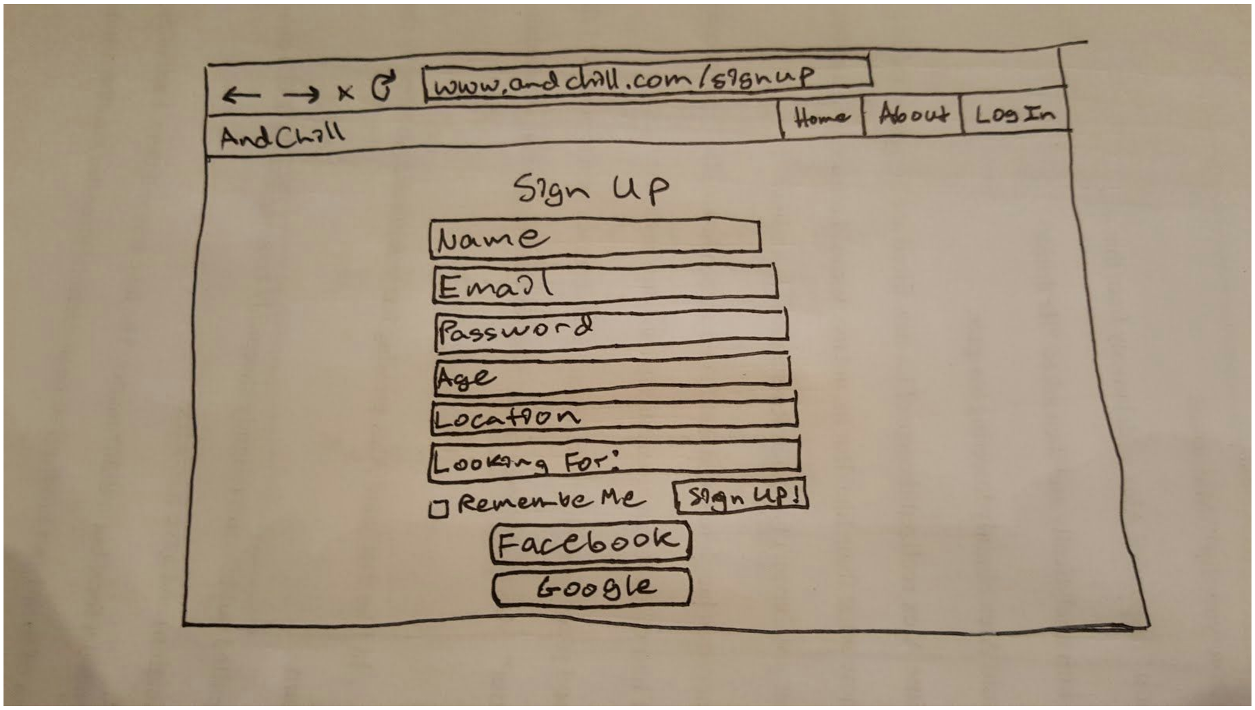
Home Page:



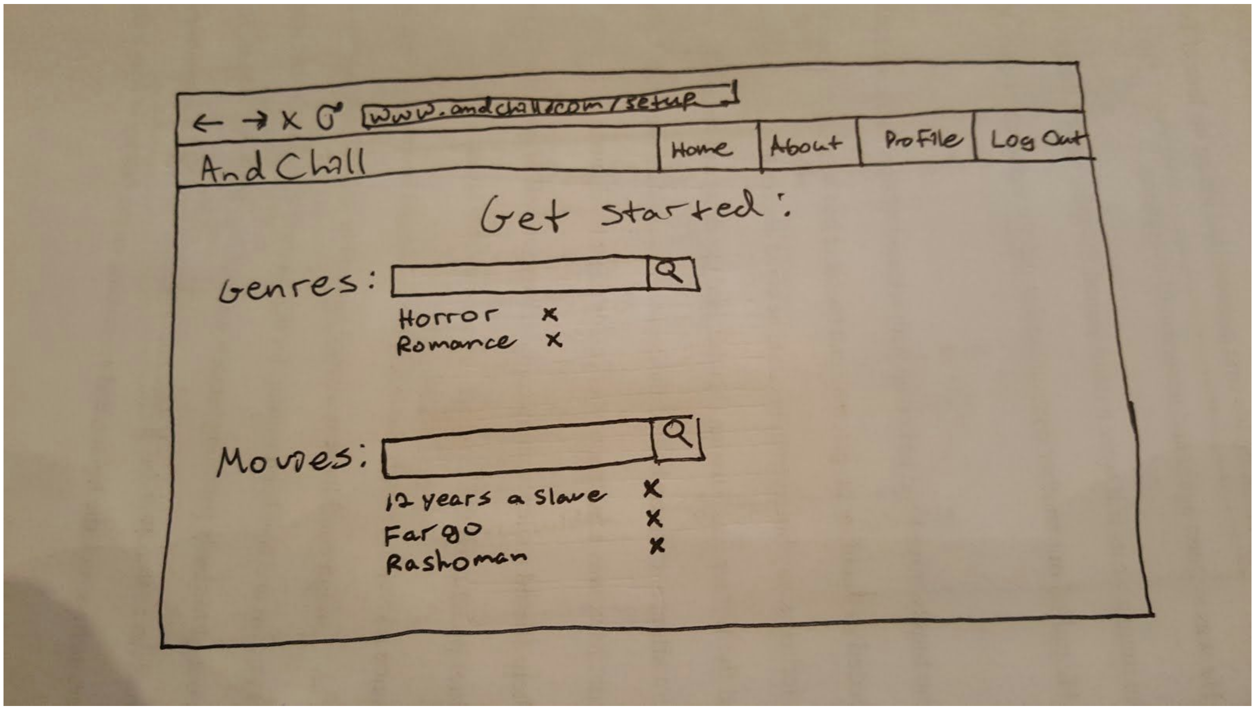
Log In:



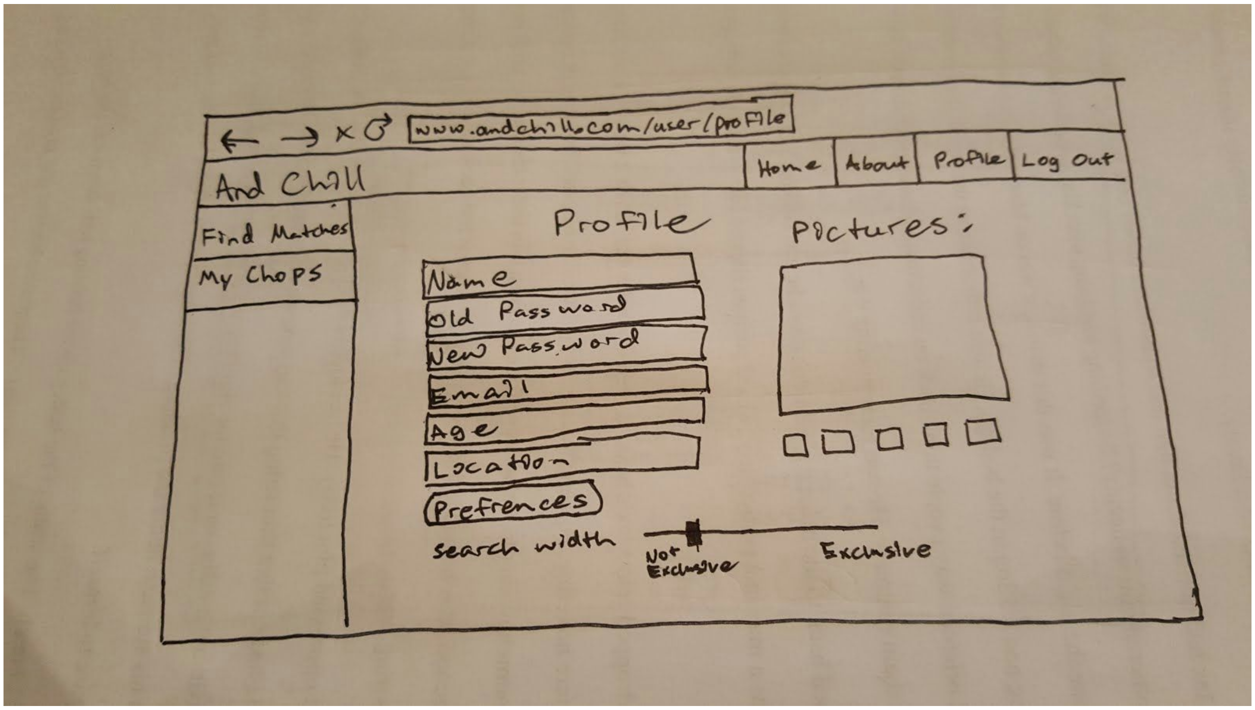
Sign Up:



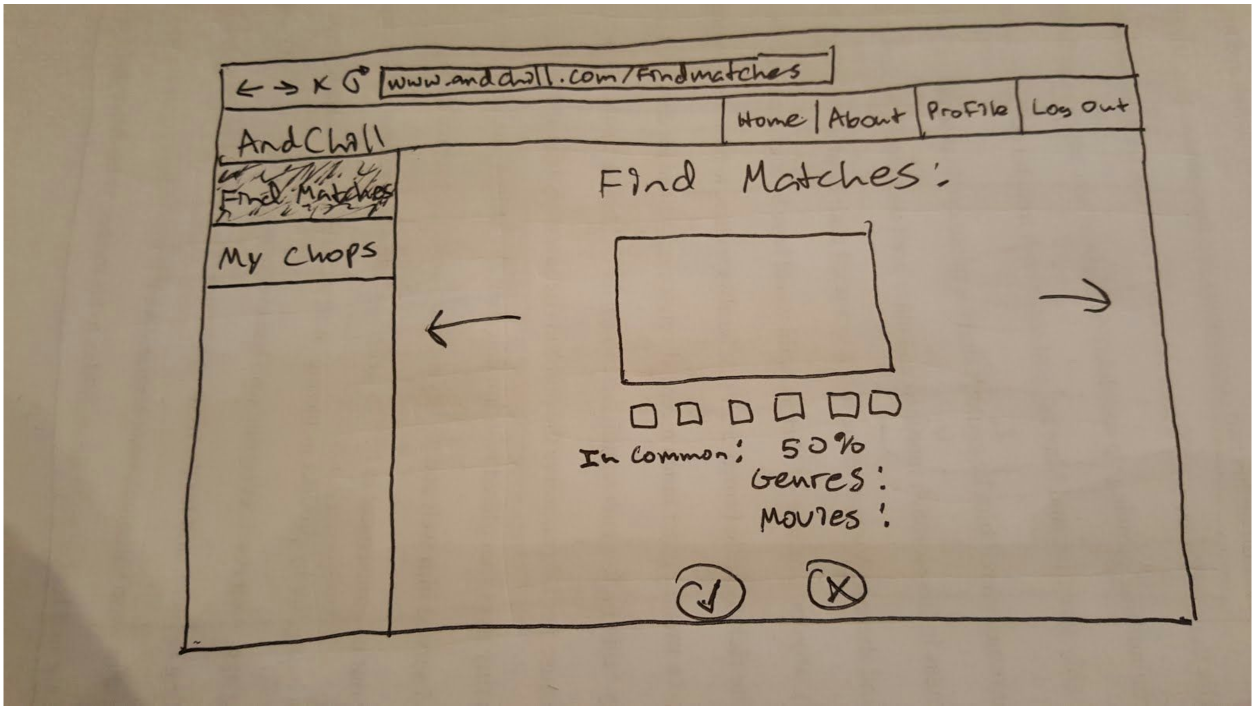
Set Up:


← → x ↻ www.andchill.com/setup
And Chill
Home
About
Profile
Log Out
Get started:
Genres:
Horror x
Romance x
Movies:
12 years a Slave x
Fargo x
Rashoman x

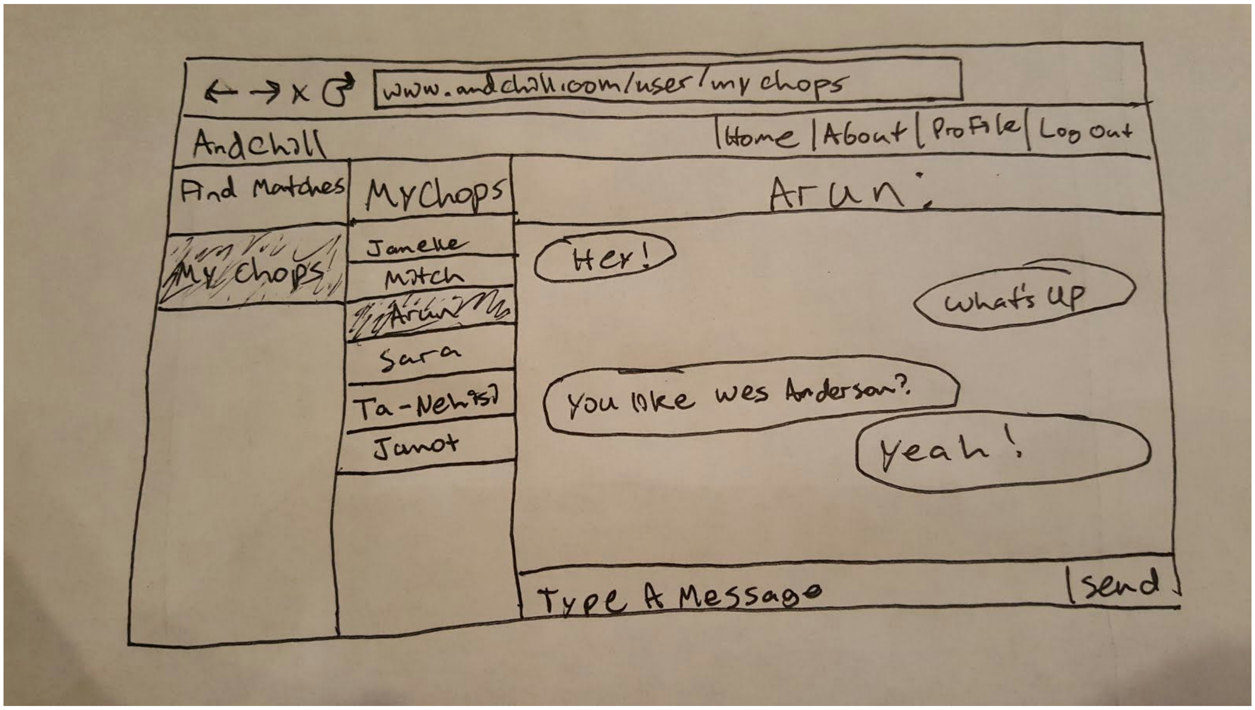
Profile:


← → x ↻ www.andchill.com/user/profile
And Chill
Home
About
Profile
Log Out
Find Matches
My Chops
Profile
Name
Old Password
New Password
Email
Age
Location
Prefrences
search width
Not Exclusive
Exclusive
Pictures:

Find Matches:



My Chops:



Hey!

what's up

you like wes Anderson?

yeah!

Design Challenges

The core of AndChill's functionality is its matching algorithm. The algorithm is tasked with evaluating two users' interests and deciding whether or not they meet the Chop threshold. Furthermore user will be displayed some potential activities to do with their Chop. We believe this problem is split up into three parts:

- **1) Interests:**
    - **Challenge:** We'll have to make a decision about the kinds of interests we support. An example would be cinematic interests like movies, actors, directors or genres.
    - **Options:** Large range of interests or narrow, more focused interests
    - **Chosen Solution:** more focused interests-- probably just cinematic related for the scope of this class
    - **Why:** Narrowing the scope of interests is beneficial since users will be more likely to have things in common, making it easier to find Chops. However with a large range of interests we might run into challenges with not finding any matches for a user. This issue relies heavily on the size of our user base. All things considered, we believe the effective solution is to only allow their cinematic interests. We want to assign the most value to the cinematic interests because we're planning on using IMDB API which will facilitate the process of finding suggested activities and Chops. In addition we believe our users will have a keen interest in finding users who have similar movie interests.
- **2) Matching:**
    - **Challenge:** Our algorithm will iterate over users' interests and decide whether 2 users have enough in common to be each other's Chop. We'll have to decide on how to construct the matching parameters such that users will receive enough satisfying Chops.
    - **Options:** Sort potential matches based on common interests and display in that order; only display potential matches with some threshold of common interests
    - **Chosen Solution:** Sort potential matches based on common interests and display in that order.
    - **Why:** Although we may be sacrificing quality of Chop, we believe this is a better solution since we don't expect a very strong user base (initially at least) and not having any chops at all would be a very debilitating user experience. As user's location will be given the most weight for a potential match.
- **3) Suggested Activity**
    - **Challenge:** Finding appropriate suggested activities based on users' mutual interests
    - **Options:** Choose the closest point of common interest between potential matches(i.e. choose shared favorite movies first, then movies from favorite

        director, then genre) or a suggestion based our user base’s favorite movies,directors and genres.
    - **Chosen Solution:** Choose the closest point of common interest
    - **Why:** The later solution would be computationally intensive since we may potentially have to iterate through our user bases interests and aggregate information. Simply going off of 2 users interests would be easier computationally wise and would be more likely to result with suggestions specifically tailored to the Chop.

In addition to the algorithm, the following capture the expected design challenges of the remaining core features of our app:

- **UI:**
    - **Challenge:** In most other dating apps the picture(s) of a potential match are given the most page real estate and are a critical part of online dating experience. Since AndChill takes a different approach and doesn’t show pictures until a Chop is accepted. The challenge is how to make keep the searching for matches UI fresh and exciting to the user in lieu of pictures.
    - **Options:** Provide a cartoony stock image to placehold the user’s actual image (something like facebook’s stock image when you don’t upload a profile picture) , give extra weight to some visual metric of compatibility, give extra visual weight to specific things both users have in common (e.g. “You both liked movie X” in large font).
    - **Chosen Solution:** Some combination of the first two options
    - **Why:** Having some overall measure of compatibility is an intuitive concept that can reassure users on their interest in a person. Highlight very specific interests they have in common might not be the best thing, especially in the case when users aren’t the most passionate about a certain similar interest however something like “You and Maria are 95% compatible” is a power statement we would like to make. In addition a stock image could be displayed in order to give the page some more visual appeal.

- **Concepts:**
    - **Challenge:** How readily we want to update a user’s Rating after he/she signs up. Given that a user’s overall Rating is an average of their previous ratings, new users are susceptible to being given an unfair low rating by one of their initial matches and then being stigmatized by future matches because of their low Rating.
    - **Options:** Automatically give users some number of full ratings when they first make their profile, effectively inflating the Rating system. Alternatively, new users’ Ratings will be hidden (or display the highest Rating) until some number of real Ratings have been placed.
    - **Chosen Solution:** Hide the true Rating of a user until enough data points are collected.

- **Why:** Artificially inflating the Rating system might be contrary to its purpose since it would take unpleasant users' Rating longer to reflect their bad sportsmanship. Whereas after a couple of bad Ratings with the suggested solution, a user's Rating will be low much faster. We plan on optimizing the minimum number of Ratings to best reflect the goal of protecting users.

- **Data Model:**
    - **Challenge:** The relationship between a Chop and users. A relationship between these two elements should exist because a Chop become an accepted Chop once both users accept it.
    - **Options:** A set relation between a Chop and Users or and ordered relation.
    - **Chosen Solution:** A set relation
    - **Why:** A set relation provides the best solution since a Chop linking Hugo and John could be interpreted to be distinct from one that links John and Hugo. By using a set relation, it would be easier to avoid redundancy when our algorithm is deciding what Chops to create next.