Team 5 Project Charter
BoilerDate

Team Members: Youngjun Yoo, Chaewon Lee, Jeongbin Lee, Hosung Ryu

Problem Statement

Currently, there are many dating apps available, such as Tinder and Bumble. However, Purdue STEM students might find it challenging to find a suitable match on these platforms. BoilerDate is an alternative dating web app that specifically targets Purdue students, with a focus on those in STEM majors. This is because students in STEM are known for their academic strength, but are also often introverted and may have weak social skills. BoilerDate will be catered towards students who want to find their significant other, not only based on common dating app features but also on common academic goals and interests.

Background Information

Audience

Purdue students, especially those in STEM fields, may struggle to find friends or a significant other, whether that be because of their heavy workload or weak social skills. Our web application will help these students converse and meet up with other students by allowing them to filter their potential matches by common interests or desired qualities.

Similar Platforms

There are similar applications such as Tinder, Bumble, and Hinge that anyone can use to find a date or significant other. These applications allow users to create a profile and connect with other users who have the qualities they are looking for.

Limitations

While many of these existing applications are already widely in use, there are a few limitations that are common to these applications. For example, there are no restrictions on who can use these apps, meaning that you cannot know for certain that your potential match is who they say they are (e.g. not within 100 miles, 50 years old, etc.). Our goal is to make a platform that will be solely catered to Purdue students to ensure a sense of security when meeting up with a potential match. Moreover, we aim for students to find a match that's supported by common academic goals, which other platforms do not consider. Finally, these applications do not allow users to filter their potential matches based on common interests, while ours can.

Requirements (Backlog)

Functional Requirements

1. As a user, I would like to create a BoilerDate account using my Purdue email, name, gender, and age.
2. As a user, I would like to log in and manage my BoilerDate account.
3. As a user, I would like to reset my password if I forget it.
4. As a user, I would like to upload and show my photos for my profile.
5. As a user, I would like to select and display my physical traits such as height.
6. As a user, I would like to display a text box about me that shows up at the top of my profile (bio).
7. As a user, I would like to choose and display my interests and hobbies for my profile.
8. As a user, I would like to choose and display my skills for my profile.
9. As a user, I would like to choose and display my relationship goals for my profile.
10. As a user, I would like to choose and display my personality type for my profile.
11. As a user, I would like to choose my academic and educational information such as GPA, major, etc.
12. As a user, I would like to select my lifestyle and show it on my profile.
13. As a user, I would like to choose my career information such as career goals, etc.
14. As a user, I would like to edit my profile whenever I want.
15. As a user, I would like to select an option for all of the items listed under number (16) when building my profile.
16. As a user, I would like to filter out those who do not match my preference.
    a. As a user, I would like to filter potential matches by GPA.
    b. As a user, I would like to filter potential matches by major.
    c. As a user, I would like to filter potential matches by gender.
    d. As a user, I would like to filter potential matches by age range.
    e. As a user, I would like to filter potential matches by interests and hobbies.
    f. As a user, I would like to filter potential matches by career goals.
    g. As a user, I would like to filter potential matches by relationship goals (friendship, significant other, etc.).
    h. As a user, I would like to filter potential matches by personality type.
    i. As a user, I would like to filter potential matches by skills (coding, bilingual, etc.).
    j. As a user, I would like to filter potential matches by companies they interned/worked at (potential referral giver).
    k. As a user, I would like to filter potential matches by degree (BA, MA, PhD).
    l. As a user, I would like to filter potential matches by US citizenship / green card.
    m. As a user, I would like to filter potential matches by lifestyle.
17. As a user, I would like to set up my privacy settings.

18. As a user, I would like to see the profile of the potential match.
19. As a user, I would like to like another user.
20. As a user, I would like to dislike another user.
21. As a user, I would like to match with another user if I like another user and the user likes me.
22. As a user, I would like to see the list of users and their profiles that liked me.
23. As a user, I would like to see the list of users and their profiles that I liked.
24. As a user, I would like to see the list of users and their profiles that I matched with.
25. As a user, I would like to send messages to my match.
26. As a user, I would like to receive messages from my match.
27. As a user, I would like to see the next user and their profile after liking or disliking the current user.
28. As a user, I would like to receive a notification (email) when matched with another user.
29. As a user, I would like to receive a notification (email) when another user likes me.
30. As a user, I would like to receive a notification (email) when a matched user sends me a message.
31. As a user, I would like to display my LinkedIn profile.
32. As a user, I would like to display my GitHub username.
33. As a user, I want to present my most significant feature in a dedicated section of my profile.
34. As a user, I would like to be able to log out of my account.
35. As a user, I would like to be able to block a matched user.
36. As a user, I would like to be able to block an unmatched user.
37. As a user, I would like to be able to report a matched user.
38. As a user, I would like to be able to unmatch a matched user.
39. As a user, I would like to be able to delete my account and remove my private information.
40. As a user, I would like to receive a text message when there is a notification (if time allows).
41. As a user, I would like to set up plans with my match (if time allows).
42. As a user, I would like to know the geographical location and distance of my match (if time allows).
43. As a user, I would like to be able to use the most recent version of the web application without any errors (if time allows).
44. As a premium user, I would like to decorate my profile with extra features (if time allows).
45. As a premium user, I would like to boost my appearance in other users' potential match lists (if time allows).

Non-Functional Requirements

Architecture & Performance

We plan to divide the work between the frontend and the backend for agile development and for better collaboration that matches each other's skills. The backend will be developed using a JavaScript backend - Express.js framework for building RESTful APIs with Node.js. Express.js is a microframework that allows greater flexibility and customization for building web applications. It's also well integrated with other Node.js technologies and will allow us to be scalable by handling routing, 5000 HTTP requests per second, and error handling on its own. This will ensure that new profiles and edits will be viewable within 2 minutes. Moreover, users will be able to see the next profile after liking/disliking the current profile within 3 seconds.

This will allow the development to be accelerated by eliminating potential compatibility issues with the frontend, which will be developed using JavaScript - React.js and Tailwind CSS/BootStrap. The frontend will be pulling data from the MongoDB backend using API requests.

Security

The ability to safely store user's data is one of the most paramount components of BoilerDate since we collect numerous personal data from each Purdue student such as their Purdue email, age, and name. To prevent any form of stealing and exploitation of sensitive data, we will implement a hash table and employ appropriate hash algorithms to safely store the data and allow easy retrieval when they are used in the application. With this data structure, we will ensure that each transaction of data from the database to our user interface is properly processed. We will also force users to use strong passwords and determine their safety level by reasonable criteria (e.g. include at least 8 characters, one special symbol, capital letter, etc.).

Usability

The user interface should be aesthetic and clean so that any user will be able to navigate through the application without difficulty and confusion. Since the primary purpose of this app is to find a match, it's important for users to feel comfortable while using the application to increase their screen time, which would further increase their probability of finding a potential match. Similar platforms mentioned above are only available on mobile apps, while our application is primarily on the web and supports both PC and mobile environments. This will give users quick access to our application across any device without the need for any extra downloads. Finally, we will add alternative text to images when possible to aid visually impaired users.

Hosting/Deployment

We are separating our development works into frontend and backend, and we will be deploying each frontend and backend separately. We plan to deploy the backend using free hosting cloud services like Vercel/Render so that our server runs 24 hours per day. We will also be using MongoDB Atlas, a fully-managed cloud database, that handles all the complexity of deploying, running, and scaling MongoDB in the cloud.