https://easymatch-hw5.firebaseapp.com/

Since my HW4 submission was pretty much a reflection of a completed project, some write-up material expected here will likely be found within the HW4 write-up.

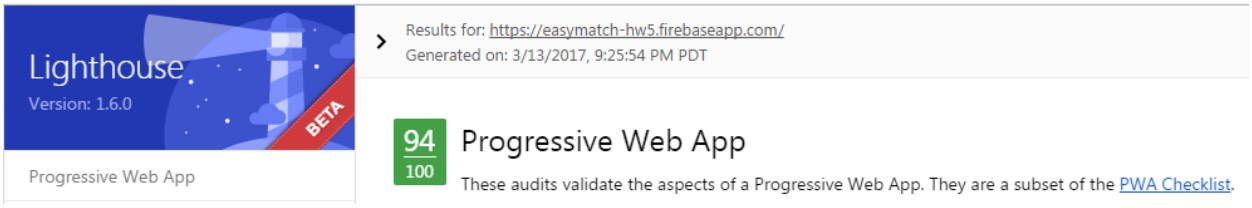
The website uses a SPA pattern, utilizing Bootstrap for the frontend design, jQuery for DOM manipulation, and Firebase for backend implementation. Since the site is a single-page app, all activity only takes place on a single HTML page that is updated in response to user actions. This page is designed to be as small as possible (**8KB)** so that it loads quickly on all types of connections. Linked resources such as modules and plugins are loaded asynchronously. The site also makes use of minified CSS and JavaScript files, but you can view the unminified version of these files within the Google Drive folder.

Minifying the JavaScript file (logic.js) decreased its size from **40KB** to **25KB**, and the CSS file (index.css) decreased from (**10KB**) to (**7KB**).

According to FireBug (a Firefox webdev debugging extension), the site will download **215KB** of data, almost all of which will be cached on an initial visit. After caching and after a page refresh, assuming no new data is retrieved, the communication will require **5KB** of traffic, mostly in the form of HTTP headers. The unminified JavaScript code (logic.js) is organized and structured, also it has commentary throughout. The CSS file also has an unminified version (index.css) within the Google Drive.

Images that users upload are optimized for quick retrieval. Depending on the complexity of the image, each can range in size between **5KB** and **15KB**. Details concerning image handling are mentioned in HW4 write-up.

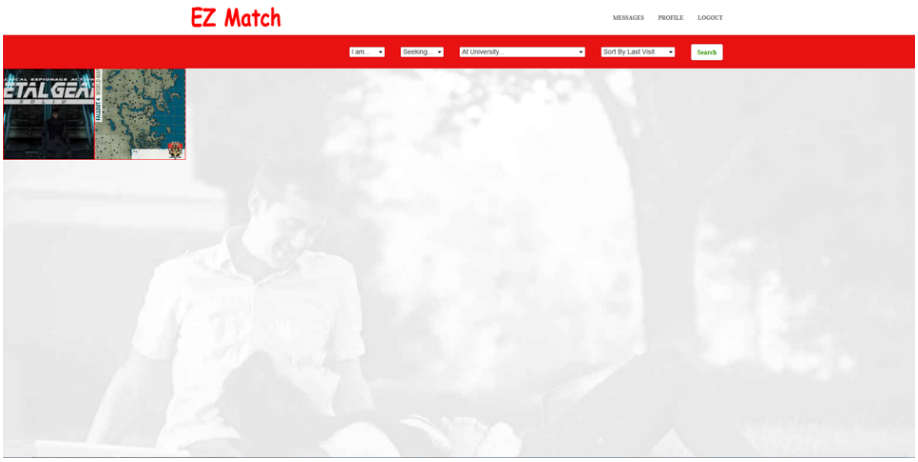
PWA features have been implemented and the Lighthouse report is below.



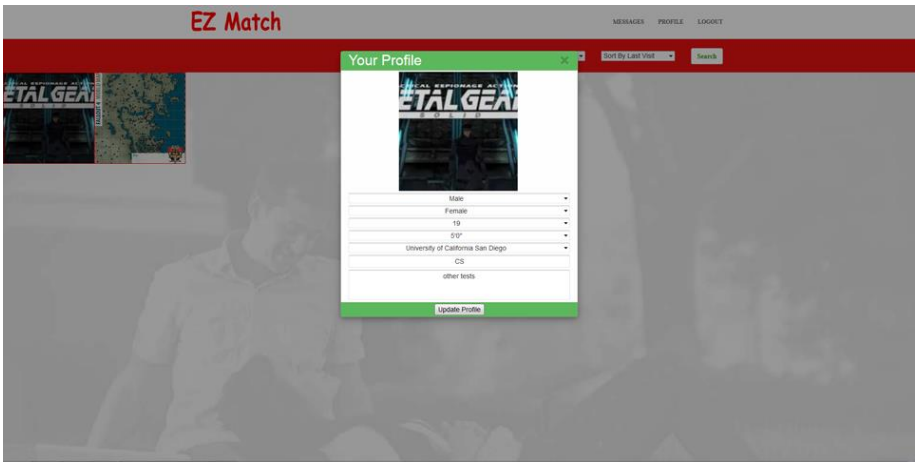
Demo Instructions

The site is a dating platform for college students. I wanted to give it a minimalistic design that is packed with visual content (images). I opted for a minimalistic design because most college students have a smart phone; they usually surf the web on these devices because they're too preoccupied with their daily routines to haul around a laptop. The design of the site is intended to cater to these mobile users such that they can immediately use the platform without having to navigate excessively. Desktop users aren't left out; the website will scale depending on the user's screen size and display additional content. Regardless of the device the site is viewed on, its SPA pattern provides a seamless user experience that responds to the user's input.

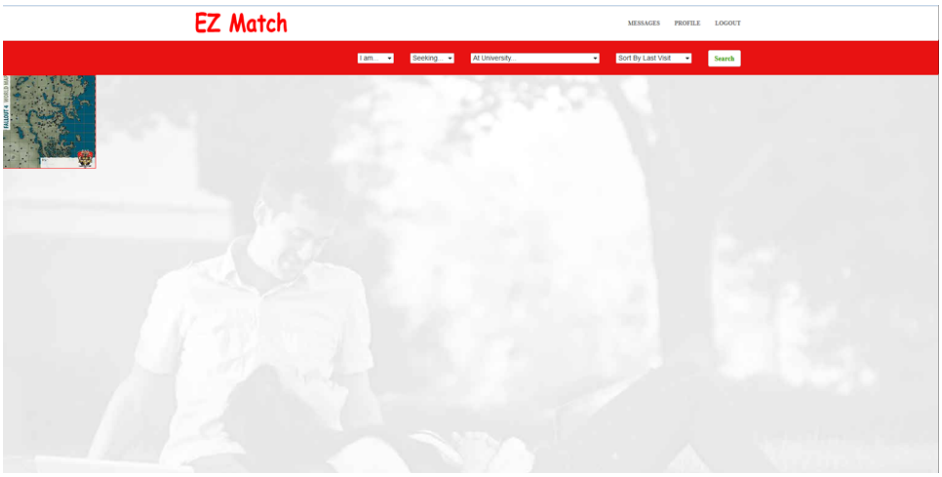
Upon visiting the site, you will immediately be shown the main interface with content loaded in the image gallery. The header of the site contains navigation elements for both logged in and logged out states. The user can log in with either email/password credentials, or with their Google account. Upon logging in, the user can update their profile (*Dex), run searches, and message other users. The search bar only contains a minimal amount of search parameters for the sake of efficiency; height and age are shown within the search results. Before the user can message anyone, they must configure their profile, which consists of a profile image and information. To message someone, just click on their profile and enter the message you want to send within the message input area and send it. This initial message will create a conversation within the sending and receiving user's "messages" area. The messages area contains all of a user's ongoing conversations.



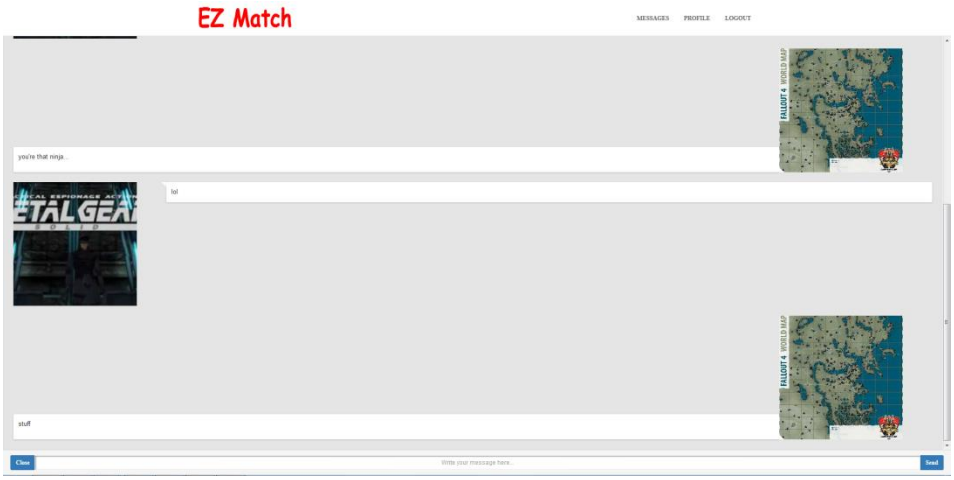
Visiting the site will show the 50 most recent visitors. There are just two in the database.



The user's profile, showing their profile image and information.



Messages area. This user has one conversation.



Viewing that conversation. Chat is updated in real time.

Summary

By designing the platform as a responsive webapp, it can be readily accessed from any device while only requiring a web browser. It was designed for mobile users in mind with minimal actions required to access its functionality. The contents of the database can be browsed without an account, so users can immediately interact with the service as a visitor, but contacting other users requires registration. By making the platform open and accessible, a user-base can be more easily accrued. This can allow for more sophisticated options to be made available at a later date that require a higher degree of commitment (downloading, installing, etc…), such as a mobile app.