Ameya Barve, Quinn Colognato, Mary Cotrupi (Team 7)

Professor Plotka

Intro to ITWS Section 1: Term Project Write-Up

22 Apr. 24

Project Write-Up: MentorMatch

**Project Description**

Our application, *MentorMatch*, aims to match students with tutors who fit their needs. Students with previous course experience can sign up to tutor students who need assistance in certain classes. Other students who may need a bit of guidance in classes can swipe either “left” on tutors who may not be a right fit for them, or “right” on tutors that they would want help from. When a student looks for a tutor, they can see the tutor’s subject focus, experience, and preferences. After looking at the tutor's information, students can decide whether they want this tutor's help. Furthermore, after a student swipes right on a tutor, they will see their “matches” on the match page. Although other tutoring sites may exist, *MentorMatch* strives to revolutionize the tutor-finding process by making it effortless and enjoyable for our users.

**Proposal Summary**

This project is intended to serve RPI students who need help with their coursework. Currently, we plan on having tutors adopt a focus on subjects like Computer Science, Math, Science, or English. We plan on benefitting the students of RPI as much as possible, so these core courses are the start of accomplishing this goal.

*MentorMatch* is different from other applications, as its purpose is to offer a fun twist to finding a mentor for you. While other tutoring applications may require you to fill out multiple forms to request help, our application simplifies this process. This is done by ensuring users spend as little time filling out tedious forms as possible. When using *MentorMatch*, only an account is needed to find a tutor. Additionally, *MentorMatch* allows users to choose their tutor, whereas other tutoring sites might auto-assign users a mentor. *MentorMatch* also will enable students to be mentored one-on-one at any time that is good for the tutor and student. This allows for much more flexibility than when compared to the current-set office hours for classes. After all, no student wants to spend hours waiting in long office hours queues, especially when trying to solve a small problem. *MentorMatch* solves all these issues and ensures the user a good experience.

As our project progressed, we each took up different roles and became skilled in a specific area. Ameya took on the database integration with PHP and MySQL. Quinn handled most of the JavaScript involving the swipe page. Finally, Mary styled all the pages with skillful CSS, while all assisted in the site's overall HTML structure.

**Project Plan**

The first part of our plan involved choosing a topic and focus areas. Inspired by Tinder, we decided to create a tutoring application, and have our focus areas be HTML/CSS and database usage. Then, we drew out a basic site structure (see our IA diagram below), made a Kanban board, and delegated tasks. However, our plan did not initially involve using databases, or our later addition of a preferences page. As time went on, we steadily worked on the application and made changes to our design as we discovered which features were and were not feasible. We ultimately accomplished functional swipe, login, and signup pages, and implemented a database to store user information and matches/preferences from these pages. By the time of project presentation, we had added to our plan goals of adding a premium service, messaging functionality between mentors and mentees, and more (see "Project Summary").

**Information Architecture (Updated)**



**Decisions Made**

At the start of our project, we discussed having only a home page, login page, sign-up page, and swipe page; we planned on only utilizing HTML, CSS, and JavaScript/jQuery. However, we made some changes in our final product. Firstly, we changed our Information Architecture diagram to include a user landing page after logging in (preferences page), as initially, users were sent straight to our swiping/matching page. Now, our landing page includes three sections: the first section displays their preferences towards their tutors (subjects, time of day, etc.), the second displays their matches, and the third section links to the swipe page. Additionally, all of these pages now utilize PHP, as they need to access the active user’s profile via the database. Another addition to the IA, were the folders for the databases and testimonials in our “Resources Folder”. We also planned on creating a premium feature, which might include faster response time from mentors and additional user preferences, among other features. However, Match+ has not yet been implemented. The actual ability to filter which tutors were recommended to a user was also not implemented due to time.

Furthermore, we decided to change our focus tracks to area one (HTML, CSS, and graphics for page layout and design), and as our main focus, area four (pulling real data from a database). Initially, we chose areas one and two (two being JavaScript interactivity) but realized our project focused more on styling and database usage.

Ultimately we still stuck to our basic plan: we kept a database (JSON file) to store our mentor’s profiles and maintained the initial design of the swipe, home, and signup/login pages.

**Challenges & Solutions**

Swipe Page:

* Issue: We had trouble implementing JavaScript to build the swipe mechanism for the match cards.
* Resolution: We separated our concerns. First, we created a Card class that had necessary functions such as appending and removing a card. Then, we had a separate swipe JavaScript file that ran each of these functions.

Database Integration

* Issue: It was difficult to combine Ajax and PHP to associate mentors with users in the SQL database.
* Resolution: Initially, when an active user swipes right on a mentor, that mentor's ID would be added to the active user's profile in the database. We also ran into issues when displaying the user's mentors. To access and then display each of the matches, we had to make an Ajax call inside of an Ajax call. This complicated the displaying process, and also didn’t properly display the mentors most of the time. To fix this, instead of the mentor's ID, we inputted the image URL of each mentor into the user’s profile; this made printing easier, as we didn’t need to make an Ajax call inside of an Ajax call.

**Project Summary**

We ended up implementing most of what we had initially planned and more. We were able to create functional home, login, signup, and swipe pages as planned. Additionally, we were able to create a user landing page (preferences), and handle database manipulation. Through this, we learned how to utilize PHP and MySQL. It also made us more comfortable working with Ajax calls, JavaScript functionality, CSS, and HTML.

In the future, we plan to implement features that we weren’t previously able to due to time constraints. This includes the premium service Match+, a messaging interface between mentors/mentees, and preference sorting and added functionality. Other long-term plans include creating a login and landing page for the tutors and establishing better site security, such as ensuring that two users can't have the same username. Ultimately, we were able to accomplish most of our initial goals. The aspects of our website that we were unable to complete required the use of PHP, which we had not decided to use until later in the project.