

Tutor Board

By Nathan Roberts, Norhan Abbas, and YuenChen Tsou

Group 19

A portfolio of our web
development project.

Web Development

Montana State University

November 28, 2019

Project Link

`http://nathanroberts.me/demo/`

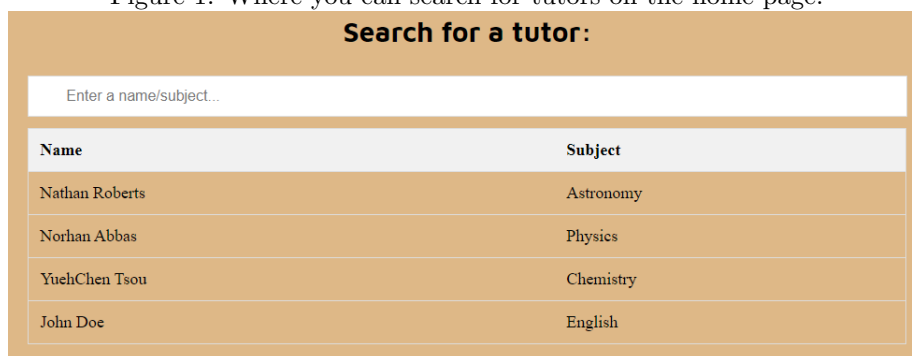
Objective

There are a lot of people out there who are masters at a variety of topics and are willing to share that knowledge. Those people, however, do not know how to find the people who want to learn! The goal of this website is to allow tutors to use an online platform to their advantage, not only to easily attract local students, but to give them the tools they need to organize their own materials and collaborate with their students. This website will specialize in serving volunteer tutors to help struggling or curious students achieve their goals.

Summary

This website is an extension of the DemoLAMP project given to us in assignment 7. But, it went through a complete overhaul. The UI of the site was completely redone, including custom headers, footers, logos, and CSS animations. The home page even includes a small javascript application, allowing anyone to see and filter through tutors currently on the website by their name or the subject they teach (Figure 1).

Figure 1: Where you can search for tutors on the home page.



Search for a tutor:

Enter a name/subject...

Name	Subject
Nathan Roberts	Astronomy
Norhan Abbas	Physics
YuehChen Tsou	Chemistry
John Doe	English

The sign-up and login flow was changed as well, allowing for the user to input their zip code, email address, phone number, and if they plan to be a student or a tutor (Figure 2)! This information is saved to the existing SQL members table by expanding the amount of columns present. But, most importantly, the tutor board itself was added to the website.

Whether you are signed in or not, you can access the tutor board to see what boards tutors have made. When you access the boards tab, you can first add a board and a description which everyone can see, or you can delete a

Figure 2: Sign up page includes more information.

Create an account

Username

☒ Student ☐ Tutor

E-mail address

Phone number

Zip code

Major

Password

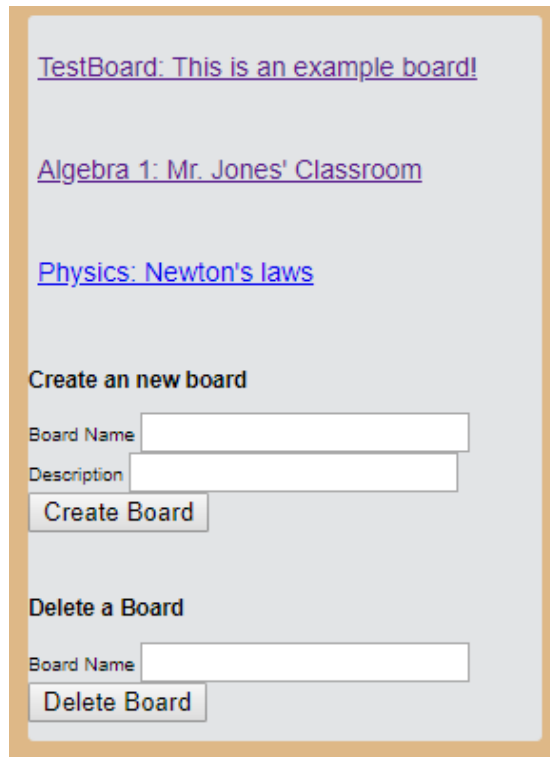
board given its name. Inside of a board, you can add information to one of six sections: goals, mentors, learners, concepts, resources, and meetings. And, all of this information is also saved to a SQL database, using two new tables called boards and boardcontent.

When you create a board, it contains a primary key as one of the columns in the boards table, meaning each row will have a unique identifier that we can always reference without a risk of collision. Then, when you load an existing board, it loads “boards.php?id=(ID)”, with ID being the unique identifier of the board selected. Boardcontent, the SQL table that contains the information

for every board we have, contains a boardID column which matches this unique board ID, so we know what information to load for what board.

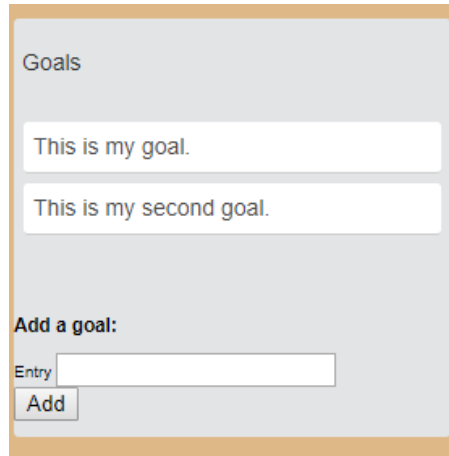
This solution allows for tutors to create a very large amount of boards without any issues regarding performance. And, since these boards are dynamically created based on their input, it requires almost no developer attention after the website is running. This approach was an alternative to creating static boards, where a developer would have to make a unique board manually for every subject (see Figure 3 and Figure 4 regarding the board).

Figure 3: Create and delete boards. Each board has a description.



The screenshot shows a web interface with a light gray background and an orange border. At the top, there are three blue hyperlinks: [TestBoard: This is an example board!](#), [Algebra 1: Mr. Jones' Classroom](#), and [Physics: Newton's laws](#). Below these links, there are two sections. The first section is titled "Create an new board" and contains two input fields: "Board Name" and "Description", each followed by a "Create Board" button. The second section is titled "Delete a Board" and contains one input field: "Board Name", followed by a "Delete Board" button.

Figure 4: "Goals" is one section of each board. You can add information for each section.



Goals

This is my goal.

This is my second goal.

Add a goal:

Entry

Add

Lastly, to summarize, we used basic but historically reliable tools to build this website. We used the LAMP stack, including Apache, MySQL, PHP, HTML, CSS, and even some Javascript.

How To: Boards

1. At the top right, go to **Menu**, and select **Boards**.
2. You can click on one of the existing links to enter a board. You can also create boards (with a name and description) in the bottom text fields. Deleting boards (given the name) is also located here.
3. After entering a board, you will see six sections. Enter information into any of these text fields, and click add. This will persist the information you want in your specified section.

Member Pages: Nathan

His responsibilities included the home page, database work, and the overall theme of the website. Nathan has created a JS app for the home page, where users have the choice to go to the login/sign-up page or go to the menu, where users can join different boards or create new ones. Also, Nathan was able to add a search engine to the home page, where users can search with the mentor's name or the target category, like Physics, Maths, Chemistry, etc. Additionally, he has already done the database work of the website such that users can create or delete boards and can add more stuff to each card as well.

Member Pages: Norhan

Her responsibilities included the main page of the different categories/subjects and the board of each category. Users can navigate from the home page to the boards page, then according to the category they pick, the related cards would show up. Each board has six cards: Goals, Mentors, Learners, Concepts to be covered in the next meeting, Useful websites recommended by mentors, and Meeting times. The main goal of these cards is to have a smoothly organized way for students and mentors to communicate with each other.

Member Pages: YuehChen

Her responsibilities included the login and sign-up page. Since the Demo-LAMP project has been used, the same PHP file was utilized. Yet, YuehChen has completely changed the user interface of the sign-in/sign-up page. As she changed the login flow as well, more information is now needed from the user including the zip code, the user's major, and whether the user is a student or tutor. Besides, YuehChen was able to store the information that is taken from the user such that the website would be able to distinguish between students and mentors.

Potential Risks

1. One of the main risks that our website could face is finding tutors who are willing to help for free. Since there are actually some websites out there that could pay online tutors, people could be leaning towards spending time earning some money instead of helping others for free.

2. Also, with the vastly usage of mobile phones, users could try to access our website via phones instead of laptops. Accordingly, this shapes one of the main potential risks; the website interface accessed from phones might discourage users from using the service provided.

3. Security risks in the case that the interaction between PHP and MySQL was incorrectly done. There is nothing wrong with either of these technologies, but bad programming in the case of not checking user input can cause large problems. SQL injection is a very common form of exploitation even today, so it is very crucial to pay attention to when you are writing any code which interacts with the database.

Conclusion

In conclusion, we learned that coming up with the ideas of the site itself and figuring out how every service or feature should work and look like is as hard as the execution. We jumped into development right away without going over the fine details of it all, and it led to lots of confusion around how certain things should look or act. For example, the database was created on the go without thinking of how the boards actually behaved. This led to the boards and boardcontent tables being created in a way that didn't work well, so it was a struggle to implement the desired functionality using this database scheme. Next time, it would be very valuable to go through every page of the website, listing every detail possible about each aspect of the UI and features present. You should know what to create, how to create it, and a good idea of the architecture required to make it all work together before you start. This way would cause a lot less confusion and allow the site to be made more quickly.