Wolverine Work: Final Presentation

By: Taylor Williams, Mnahil Syed,

Emily Buzle & Kylie Callison

Introduction

Team Members

Taylor Williams

- Computer Science Major, 4th year
- Front-End, Backend + Database

Mnahil Syed

- Cybersecurity major, 4th year
- Front-End, Backend

Emily Buzle

- Computer Science Major, 4th year
- Front-End, Backend

Kylie Callison

- Cybersecurity major, 4th year
- Front-End, Backend + Database

Wolverine Work's Goal



Our Objective

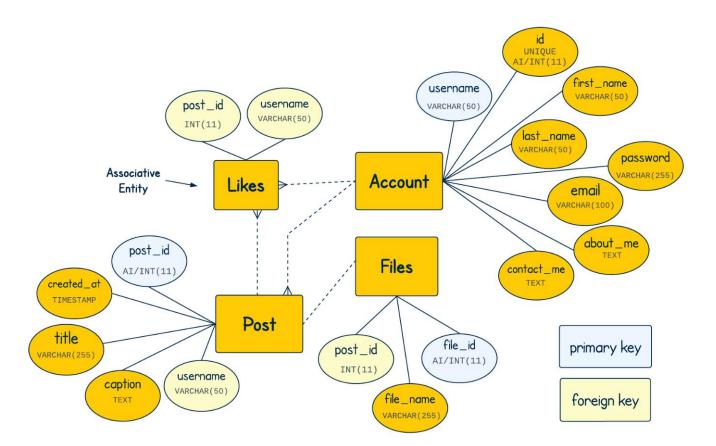
Wolverine Work is a website where the University of Michigan students can showcase their student work for anyone to see, whether its other students, faculty, potential employers or just casual viewers.

Importance

Wolverine Work creates a central spot that allows for students to display their talents and gain recognition for their hard working skills. This also creates a space for the University to showcase the skills their students have to offer.

Database Schema Design

WOLVERINE WORK DATABASE



Interfaces & Implementation

Design Tools

Color Theme: University of Michigan

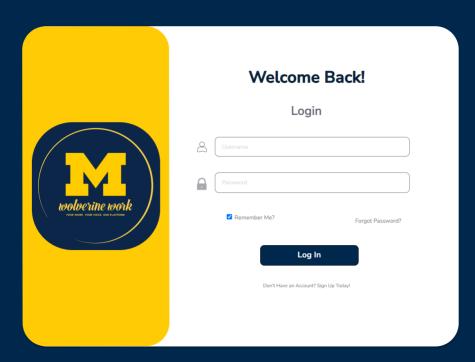


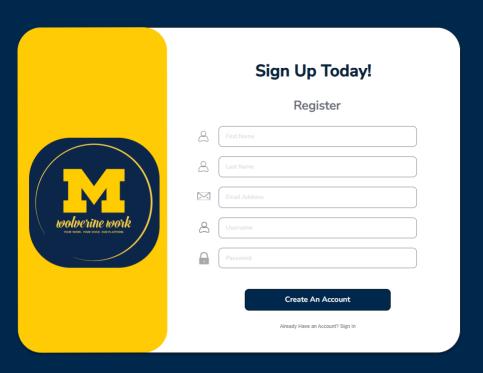
Login and Register Pages

- University students will be able to log into the website, by entering their username and password.
- If not already registered, they can sign up by inputting a first name, last name, valid umich email,
- username, and password. This account information will be stored on the website's database.

- Users have the ability to easily navigate between the login page and register page as needed.
- If a user forgot their password, they can navigate to the forgot password page for assistance.

- Passwords are hashed and emails are validated via PHP, and stored in the database with SQL insertion.
- SQL statements are prepared to prevent SQL injections.



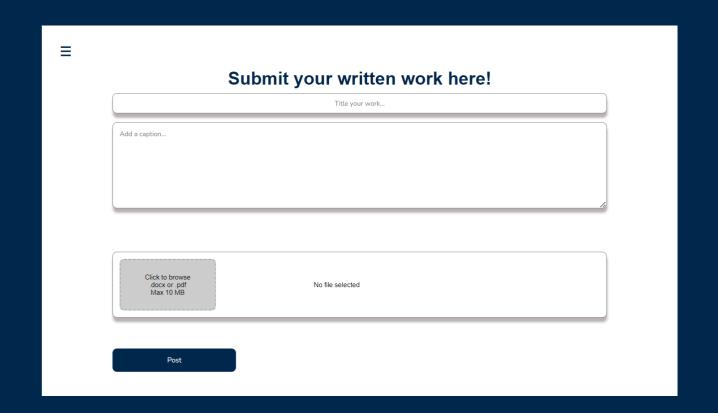


Login Page Register Page

Create A Post Page

- Users that are logged in can submit posts they want to share to the website in either DOC or PDF format.
- Users will also need to input a Title and Caption as part of their submission.
- Redirects to homepage after submission
- Side navigation bar and footer can also redirect user.

- Submissions/Posts are processed with PHP and stored into the database with SQL insertion.
- Individual files are also stored in a separate table within the database and within a "uploads" folder.
- SQL statements are prepared.

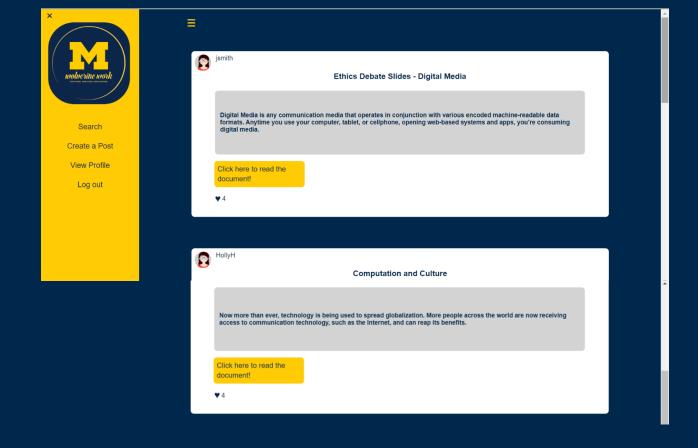


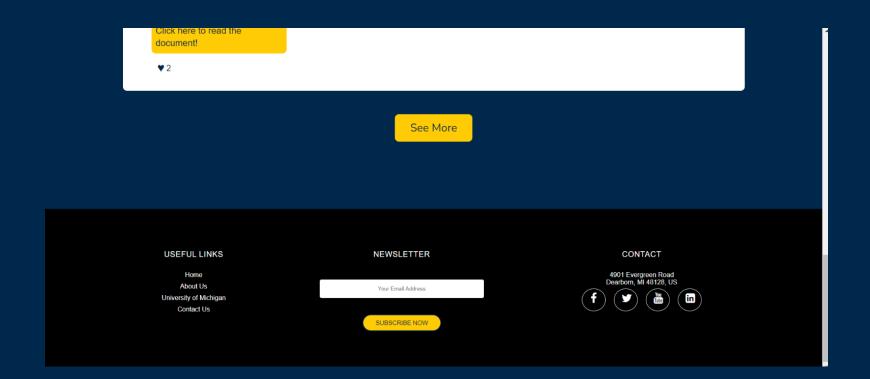
Home Page

- Home page displays all posts for all users to see, whether a user has an account or not.
- Users are given previews of student work. If they want to read more they can navigate to "click to see read the document" button which will redirect them to the full DOC or PDF file of the post.

- Users can see more posts by pushing the "See More" button.
- Footer is included at the bottom of page (Also on Create a Post and Profile pages).

Users can also like posts, the like count will be displayed/updated with PHP and JavaScript. "Like" information will be stored in the database. If like count does not go up when a user likes a post, that means they already have liked the post, users can only like a post once.

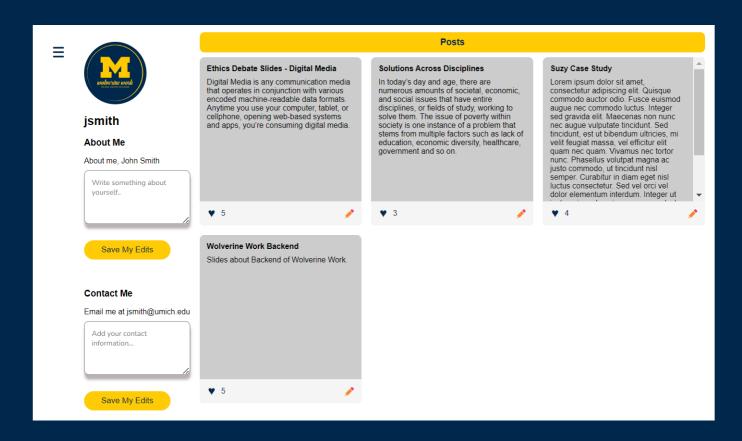




Profile Page

- Allows users to manage profile information, including a profile picture, about me section, and contact details.
- Displays a collection of posts related to the specific user who is logged in, including like and edit buttons/functionalities.

- The application connects to a database for storing user profiles, post data, and other relevant information.
- Like counter algorithm prevents users from liking a post more than once.
- Profile editing logic allows users to edit their "about me" and "contact me" sections with dynamic UI updates.
- Clicking post title allows users to download the post documents,

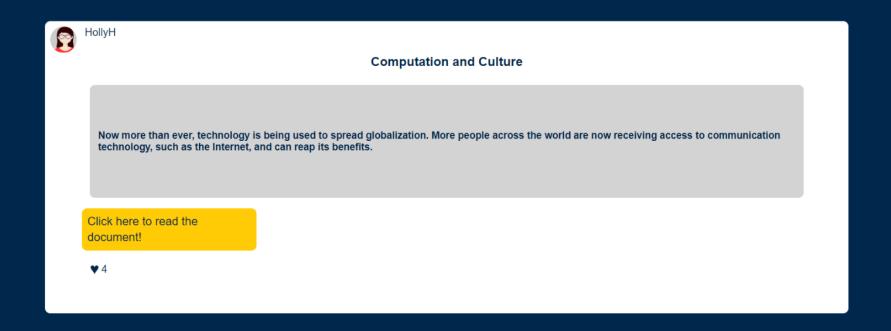


Search

- User can search any term relating to the username, title or caption.
- Posts will appear like the homepage for simplicity and readability.
- The user is able to do all the different functionalities that the homepage provides as well like liking, viewing the whole essay, etc.

• The application connects to a database to retrieve the essays that relate to the username, title or caption typed into the search engine.





Thank you! Any Questions?