IMY 220 Project 2025

Playlist Sharing Website

Deliverable 0

Instructions

- This deliverable must be **completed by 07:30 on 8 August** as indicated on ClickUP.
- You must **demo** your deliverable submission in the demo sessions in order to be awarded marks.
- **No late / email submissions will be accepted**.

We recommend you constantly refer to the overall spec as well as your individual deliverable specs to make sure your project meets all of the requirements by the end.

Make sure you also continuously test your project as you add new functionality to make sure everything keeps working as expected.

Part 1: GitHub Repository

You are required to create a **GitHub repository** to store all of your project files and have some form of source control. The repository visibility must be set to **public** so your code can be marked. The **link** for this repository should be submitted on ClickUP along with your wireframes.

1. Create an **account** on GitHub https://github.com/ if you don't already have one.

    You will be using Git and GitHub a lot throughout your academic and non-academic career, so it's advised you create a username that is professional (something like `name-surname`) as it is visible on all of your commits.
2. Create a **new empty repository** and commit your first change (see: https://docs.github.com/en/repositories/creating-and-managing-repositories/quickstart-for-repositories).

    You should set the repository visibility to **public**.
    Add a **README**.
    You can commit your change directly to the **main** branch.
3. Set up **Git** if you haven't already (see: https://docs.github.com/en/get-started/getting-started-with-git/set-up-git)

and **clone** your repository into where you would like your local files to be (see: https://docs.github.com/en/repositories/creating-and-managing-repositories/cloning-a-repository)

4. Once you have your files locally, make one more change to your README in your text editor of choice then **commit and push** this change to your remote repository to make sure everything is set up correctly.

> If you do not have something like GitHub Desktop, you will need to use the **GitHub CLI** (recommended, see: https://docs.github.com/en/repositories/creating-and-managing-repositories/quickstart-for-repositories?tool=cli#commit-your-first-change).

You should now have a new repository with a README that has 3 commits.

Part 2: Wireframing and Visual Design

Before you start coding your project using React, you must have a clear plan of how your website's pages should look. This will help you to easily break up your website's elements and functionality into reusable components. **It is very important to complete this step properly as it will make development down the line a lot smoother for you**.

1. Create wireframes of AT LEAST the following pages (refer to your overall spec for details):
   a. The **Splash** page.
   b. The **Home** page (with the song and playlist feeds - if these feeds will not be visible at the same time on your website, create two different versions of this page which represent each feed when displayed).
   c. The **Profile** page
   d. The **Project** page (that shows an individual project the user will see when a project name is clicked on).
2. You are **strongly encouraged** to create additional wireframes for different parts of your site (e.g., Login, Edit Profile, etc.) to make your project easier to visualise, but these are not required.
3. **Only the Splash page** should be a high fidelity wireframe, i.e., should look as close to the final website as possible (and should include colour, images, fonts, final layout etc.). *Refer to the design guidelines below for guidance*.
4. **The rest of the pages** do **not** need to be high fidelity, only the general layout / hierarchy of the page as well as show where certain content will go (e.g., Headings, text, images (use grey squares or circles for these), buttons, links, etc.). All of this text can be dummy data, BUT it **should be clear what content is needed on what page** (e.g., a project name, project description, project image, project tag(s), etc.).

A wireframe is not a final plan of your website. As you develop, things will likely change as you figure out how to implement certain functionality. These wireframes should be designed to guide you on what data / content you will need at which points in your website.

You may use any tool to complete your wireframes, as long as this tool allows you to export your wireframes as **PDF**. A popular tool to use is **Figma**, for which many tutorials exist on YouTube (like this one: https://www.youtube.com/watch?v=HZuk6Wkx_Eg&ab_channel=FluxAcademy).
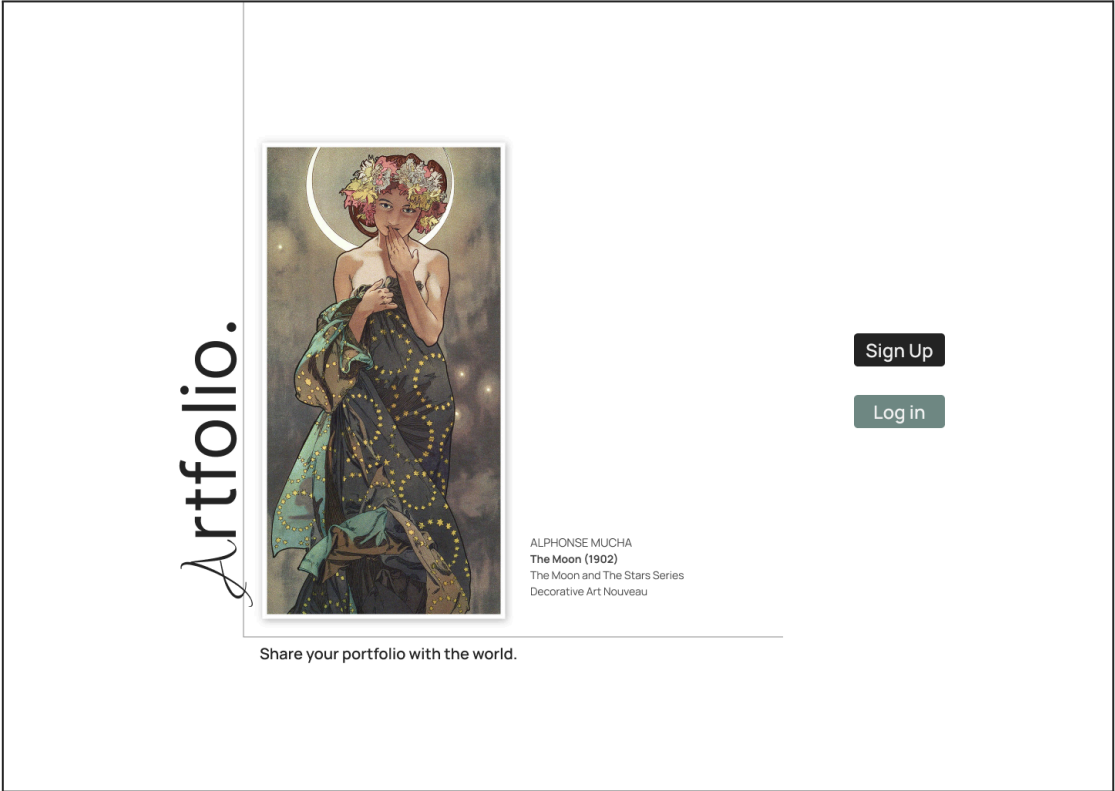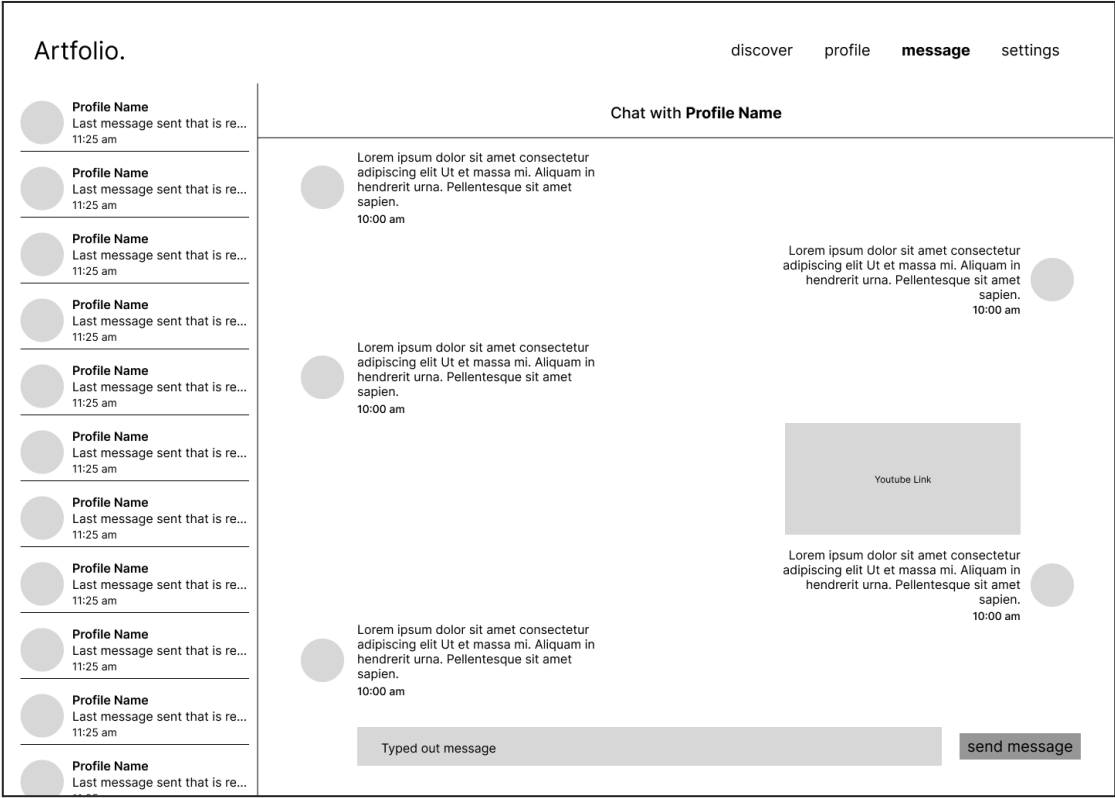

Visual Design Guidelines

- **#1: Be creative!**
- Your website can have a general **theme** (any type of version control website) or can have a specific **theme** (e.g., based on specific types of software). Just keep in mind that you will have to include logical data for your website, so make sure that your theme allows for this.
- Your website needs to have a (preferably distinctive) **name**.
- Design a **logo** for your website. It can be anything. Go look at existing logos for ideas. It can be a text-only logo.
- Choose a **background** for your website. There are some nice patterns on http://subtlepatterns.com/. It can also be a single colour (including plain black/white). HINT: look at existing websites for inspiration. A background that draws away from the contents of the website (i.e., the foreground), is detrimental to the overall website.
- Choose a **colour scheme** that fits with the name, theme, and background of your website. Use colour schemes that match, e.g., use the colour schemes that match your patterns or find colour palettes online. Search for colour scheme creators, e.g., https://www.colourlovers.com/ or https://coolors.co/.
- Choose and embed appropriate **fonts**. You should have at least 2 fonts and not more than 3. There are plenty of websites where you can embed fonts from, e.g., Google Fonts. Do not copy fonts from your friends. Do not use default fonts.
- Your pages (besides the Splash page) must have the same **layout structure throughout** your website (i.e., elements such as the navigation, sidebar (if you have one), etc. should be **consistent**).

Here is an example of a **high-fidelity** wireframe for the splash page of an art portfolio website:

Here is an example of a **low-fidelity** wireframe for the messages page of the same site. *Note how all required data / elements are clearly shown, even when dummy data is used.*

Submission Instructions

Submit the following to ClickUP before the deadline:

Place these in a folder named with your `Position_Surname_D0`
- A text file containing a link to your GitHub repository.
- A PDF containing all of your Wireframes for your different pages.