

Task-by-Task Guide

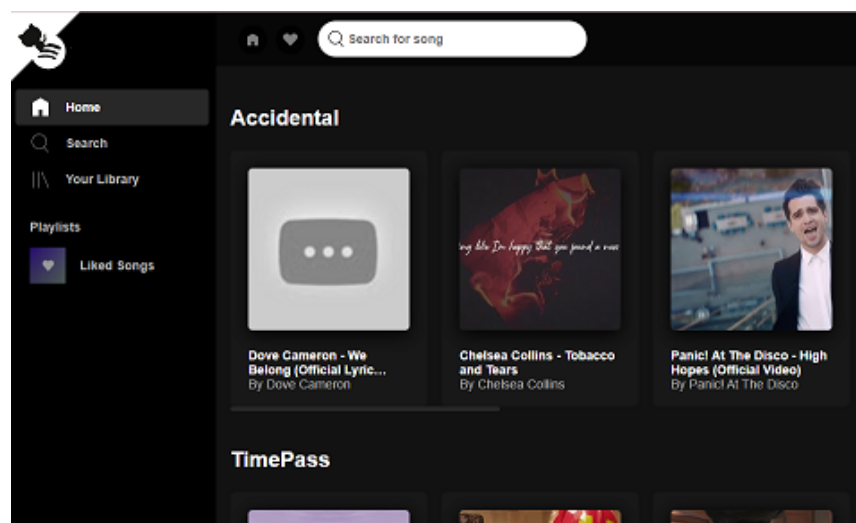
This document contains high-level guidance on how to complete your project. Although this guidance does not offer step-by-step explanation or a tutorial on how to complete your task, it will help you understand the project's requirements and the milestones needed to craft an attractive product in a professional way.

Step 1: Getting Started

In this step, you will create the GitHub repository that you will be working on for this project. This part is important because you will be showcasing your work to future employers using this GitHub repository. To create your repository, you will fork the existing project code from [Lushlyrics GitHub repository](#)

After you create your repository, you will proceed with cloning it to your local development environment. Then you will create a local branch to continue the rest of the work in this step and the following steps.

After running the existing project locally on your machine, you can observe that the current project setup allows anonymous users to access it. This means that any user on the internet is able to view the website, browse music albums, and play songs. The below figure illustrates the current website setup.



Your task in this step is to secure access to the website by blocking unauthenticated users from accessing the website. In other words, users will not be able to use LushLyrics's website or access any of its content. The figure below illustrates one potential outcome of applying this restriction. Your output does not need to be as fancy as this illustration since you will be implementing a login screen in the following tasks.



After you complete this step, make sure to commit your changes to the local branch and push them to the remote branch to make sure your changes are not lost. You will also need to commit your code frequently while working on this project and push it to the remote repository to ensure code availability.

Implementing User Authentication and Authorization

In this step, you will create user-friendly and attractive registration and login forms. You will also implement user registration and login workflows at the backend part of the website, test the registration and login process, and commit your changes to the local and remote branch.

Let's begin with the login and registration forms. These forms need to be user-friendly and attractive for users to encourage them to use the website. You can implement these two forms in various ways (Here's where your creativity and passion become handy). We've included a registration form below for you to get a sense of what is required in this step and to ignite your creativity.

The image shows a registration form titled "Registration" set against a purple gradient background. The form is a white rectangle containing six input fields arranged in two columns. The left column has fields for "Full Name", "Email", and "Password". The right column has fields for "Username", "Phone Number", and "Confirm Password". Each field has a placeholder text: "Enter your name", "Enter your email", "Enter your password", "Enter your username", "Enter your number", and "Confirm your password". Below the input fields is a wide, horizontal button with a blue-to-purple gradient, labeled "Register".

After designing your login and registration forms, it's time to do some work at the backend part of the website where you create methods to process user registration by creating user accounts and storing them in the database. Similarly, you will implement the required methods to login and logout users in and out of the application.

Once you complete the login and registration processes, you can move on to implementing a password recovery feature to help users recover their accounts in case they forgot their credentials. You can do this password recovery feature in various ways such as using emails or phone users phone numbers as a password recovery mechanism. Do not forget to design the necessary forms for this workflow.

By the end of this step, users should be able to create accounts and login to the website by providing their credentials. They should also be able to recover their accounts if they lose their credentials.

By the end of this step, users should be able to create accounts and login to the website by providing their credentials.

Creating a Pull Request

In this step, you will commit any changes you have made to the local branch, then push those changes to your remote branch on GitHub. After uploading your changes, you will merge the branch you have been working on with the main branch on your GitHub repository.