

## Assignment: Portfolio Landing Page Replication

### Objective:

Replicate the web page exactly as specified in the provided Figma design using Next.js, ensuring that all components are well-structured and placed in appropriate directories. Finally, host the website on Vercel or a similar platform.

### Task Details:

1. **Replicate the Web Page:**
  - Access the Figma design: [Figma Design](#)
  - Your task is to replicate the web page **exactly** as shown in the Figma design. Pay close attention to every detail, including layout, typography, colors, images, and responsiveness.
2. **Component Design and Directory Structure:**
  - Break down the web page into reusable components (e.g., header, footer, sections, buttons, etc.).
  - Place each component in a well-organized directory structure within the Next.js project. For example:
    - `components/`: For reusable components
    - `pages/`: For page-level components
    - `styles/`: For CSS or styled-components
    - `public/`: For static assets like images and fonts
3. **Use Next.js Framework:**
  - Implement the design using the Next.js framework. Ensure the application is optimized for performance and SEO, leveraging the features of Next.js such as Server-Side Rendering (SSR) or Static Site Generation (SSG) where appropriate.
4. **Hosting:**
  - Host the final website on Vercel or any other hosting platform that supports Next.js. Ensure the website is live and accessible via a public URL.

### Submission Requirements:

1. **GitHub Repository:**
  - Create a GitHub repository for the project.
  - The repository should include all source code, properly commented and organized.
  - Include a `README.md` file that explains how to set up and run the project locally.
2. **Vercel Hosting:**
  - Provide a live link to the hosted website on Vercel (or the chosen platform).
  - The hosted website should match the Figma design exactly.

### Evaluation Criteria:

1. **Design Accuracy:**
  - The web page must replicate the Figma design perfectly in terms of layout, responsiveness, and visual elements.
2. **Component Design and Directory Structure:**
  - Components should be well-designed, reusable, and placed in the appropriate directories.
  - The directory structure should follow best practices, making the project easy to navigate and maintain.
3. **Code Quality:**
  - The code should be clean, well-documented, and follow best practices for React and Next.js development.
4. **Hosting:**
  - The website should be successfully hosted and accessible via the provided URL.
  - The hosting should reflect the quality of the design and code implementation.

## Submission

1. **Link to vercel hosted website**
2. **Link to github repo for the above code**