**Project Name: Interactive Portfolio**

**Objective:**

The goal is to create an interactive portfolio website that showcases various applications, allowing viewers to interact with the apps directly. The portfolio will be built as a single-page web application (SPA) using React and will be hosted on GitHub Pages.

**Functional Requirements:**

1. **App Features:**
   * **RESTful API Web App**:
     + The app will handle simple CRUD (Create, Read, Update, Delete) operations.
     + It should support file upload and download functionality.
     + Interaction with the app will be facilitated via API calls (e.g., through a form or buttons).
   * **Real-Time Metaverse Chat Game**:
     + A metaverse-inspired chat game will use WebRTC or WebSocket to facilitate real-time communication.
     + Viewers can interact with each other within a metaverse-style environment.
   * **Todo App with Web Speech API + OpenAI**:
     + The app will allow users to add, update, and delete tasks using voice commands through the Web Speech API.
     + Integration with OpenAI should provide smart suggestions and task management functionalities.
   * **Falling Block Puzzle Game**:
     + A classic Falling Block Puzzle Game will be embedded within the portfolio, offering interactive gameplay.
   * **Simple Mobile OS Design**:
     + A minimalistic design of a mobile operating system interface will be displayed on the web for demonstration purposes.
     + The design should be intuitive and resemble a basic operating system UI.
2. **Design and Layout:**
   * **Skill Stack Page**:
     + A dedicated page should be created showcasing your skills, with associated logos for each technology used in the apps.
   * **Title Page**:
     + Your personal picture should be prominently displayed on the title page.
     + The title page should reflect the professional nature of the portfolio and include your name, title, and a brief introduction.
   * **Responsive Design**:
     + The portfolio should be responsive, working across both standard PC and mobile screen sizes.
     + Design should adjust seamlessly based on screen size, providing an optimal experience on both devices.
3. **Technical Requirements:**
   * **Single-Page Application (SPA)**:
     + The portfolio should be built as a single-page application using React to ensure smooth navigation and a dynamic user experience.
   * **Hosting**:
     + The app should be deployed on GitHub Pages for easy access and sharing.
4. **Non-Functional Requirements:**
   * **Performance**:
     + The website should load quickly and be optimized for performance.
     + Considerations should be made for optimizing image sizes and code splitting for faster page load times.
   * **Accessibility**:
     + The portfolio should adhere to basic accessibility guidelines (e.g., proper use of ARIA attributes).
     + All interactive elements should be usable with a keyboard and screen reader.
   * **Security**:
     + Sensitive data should be appropriately handled, especially for the RESTful API app that supports file uploads/downloads.