Software Requirements Specification (SRS)

Project Title: PortaBuild – Portfolio Builder Platform

1. Introduction

1.1Purpose

The purpose of PortaBuild is to empower students and professionals especially those without coding knowledge to create, manage, and customize their own online portfolios effortlessly. This platform enables users to build a professional digital presence that highlights their education, skills, projects, and achievements with just a few clicks.

1.2 Scope

PortaBuild is a full-stack web application designed to generate and manage online portfolios dynamically.
Each user can:

- Register and automatically receive a starter portfolio with editable content.
- Access a personal dashboard to manage all portfolio sections (profile, stats, projects, education, skills, experience, services, and more).
- Select and apply visual themes to personalize their portfolio.
- Control visibility for each section, showing or hiding it as desired.
- Share their portfolio via a unique public URL (e.g., https://portabuild.com/username).
- For non-registered users, the system provides:
- A homepage explaining the platform.
- A public user directory showcasing featured portfolios.
- Contact options for the PortaBuild team.
- The system aims to be scalable, mobile-friendly, and eventually deployable as a SaaS (Software as a Service) product.

1.3 Intended Audience

This document is intended for:

- Developers and designers involved in the PortaBuild project.
- Internee.pk supervisors and evaluators.
- Future collaborators or investors evaluating the system architecture and scalability.

1.4 System Overview

- PortaBuild follows a modular MERN (MongoDB, Express.js, React.js, Node.js) architecture.
- The frontend is powered by React + Vite, styled with Tailwind CSS and shadcn/ui components.
- The backend uses Express.js with JWT-based authentication, and MongoDB Atlas for data storage.
- Docker is used for containerization, ensuring consistency across deployments.

2. Overall Description

2.1 Product Perspective

PortaBuild is a standalone web-based system, acting as both a **portfolio management tool** and a **portfolio hosting service**.
Each user interacts with:

- **Frontend** (Admin Dashboard + Public Portfolio)
- **Backend API** (Handles CRUD, authentication, and data management)
- **Database Layer** (Stores user data, themes, and portfolio content)

2.2 Product Features

- No-code portfolio creation.
- Fully responsive and PWA-ready.
- Role-based content access (User/Admin).
- Live updates changes reflect instantly on the public portfolio.
- Section visibility control.
- Multiple professional themes.
- Mobile-first design for accessibility.

2.3 User Characteristics

Users are generally students, job seekers, or professionals from non-technical backgrounds.
They expect:

- A simple and intuitive interface.
- Mobile access and ease of customization.
- Instant reflection of updates on their live portfolio.

2.4 Operating Environment

- **Frontend:** React + Vite + TailwindCSS

- **Backend:** Node.js + Express.js
- **Database:** MongoDB Atlas
- **Hosting:** Vercel (frontend) and Render (backend via Docker)
- **Containerization:** Docker + GitHub Actions CI/CD pipeline

3. System Features

3.1 Authentication & User Management

- Users can register, log in, and log out securely using JWT cookies.
- On registration, a starter portfolio with default data is generated automatically.
- Passwords are hashed and stored securely using bcrypt.

3.2 Profile Management

- Manage **Hero Section** — profile image, name, tagline, and description.
- Manage **About Section** — personal summary, image, and professional tagline.
- Add, update, or delete buttons (e.g., Hire Me, LinkedIn).
- Each button supports **icon, text, and link**.

3.3 Overview Sections

Each user's portfolio includes several predefined sections that they can **edit or toggle visibility** for:

Section	**Editable Attributes**	**Example**
Stats	Title, Description, Cards (icon, title, description)	"Projects Completed – 20+"
Services	Title, Description, Cards (icon, title, description)	"Web Development – MERN Stack"
Projects	Title, Description, Cards (image, title, description, button)	"Portfolio Builder – Built with React"
Skills	Title, Description, Cards (icon, title)	"React, MongoDB, Node.js"
Education	Title, Description, Cards (icon, title, description)	"BSc Software Engineering – UET Mardan"
Experience	Title, Description, Cards (icon, title, description)	"MERN Developer – ProductBox (2024)"
Certificates	Title, Description, Cards (icon, title, description)	"Cisco Packet Tracer Course"

Each section supports:

- CRUD operations for cards and buttons.

- Visibility toggle (Show / Hide section).
- Live preview on dashboard.

3.4 Theme Customization

- Users can select from multiple predefined themes.
- Each theme includes attributes like colors, fonts, border style, shadows, and layout type.
- Example fields:
  primaryColor, secondaryColor, backgroundColor, fontFamily, buttonStyle, etc.

3.5 Contact Section

- Manage contact information and social media links (GitHub, LinkedIn, Instagram, Twitter, YouTube, Email).
- Users can update or disable any link individually.

3.6 Public User Directory

Displays all published portfolios with pagination and sorting by popularity.

Links directly to user portfolio URLs.

4. Non-Functional Requirements:

Category	Description
Performance	System must handle at least 1000 concurrent requests efficiently.
Scalability	Built to support future SaaS expansion.
Security	JWT authentication, bcrypt password hashing, and CORS policies.
Usability	Fully responsive UI; supports desktop, tablet, and mobile.
Maintainability	Code structured using modular components and service-based APIs.
Reliability	Docker ensures consistent environment across deployments.
Availability	99% uptime guaranteed via Render and Vercel hosting.

5. Technology Stack:

Layer	Technologies
Frontend	React.js, Vite, TailwindCSS, shadcn/ui
Backend	Node.js, Express.js
Database	MongoDB Atlas
Auth	JWT + bcrypt
Deployment	Render (backend), Vercel (frontend)
Containerization	Docker
CI/CD	GitHub Actions (auto image build & deploy)

6. Future Scope

- Launch PortaBuild as a full SaaS platform with subscription tiers.
- Add analytics for portfolio views and engagement.
- Provide domain linking (custom subdomains or custom domains).
- Include AI-based section suggestions and content generation.
- Enable export/import of portfolio data.
- Offer premium themes and custom layouts.

7. Conclusion

PortaBuild simplifies professional portfolio creation by providing a **no-code, fully customizable platform** for students and professionals.
It ensures scalability, real-time customization, and professional aesthetics empowering users to build their online identity with zero technical barriers.