Replit Project Brief: React Portfolio with Backend & GitHub Pages Hosting

Project Goal:

Create a multilingual academic-style portfolio website using React on Replit.

The site must visually resemble the Hugo Academic Theme, but be built entirely with a separate frontend (React) and backend (API or CMS-ready).

1. Architecture

The project must be split into two parts:

- Frontend: React (JavaScript/TypeScript), responsive, hosted on GitHub Pages
- Backend: Node.js or lightweight API server (hosted on Replit, Render, or another backend platform)

The frontend should fetch portfolio data (projects, experience, etc.) from the backend via HTTP requests (e.g., REST API).

2. Design & Layout

Recreate the look and layout of https://academic-demo.netlify.app/:

- Navbar: About, Projects, CV, Blog, Talks
- Hero section with image, name, intro, download CV buttons
- Featured projects, timeline layout for experience
- Multilingual switch (EN/DE), dark/light toggle

3. Content Source

All content will be stored in backend (JSON or database) to enable future updates without modifying frontend code.

Examples:

- CVs (English and German)
- GitHub project links

- Profile info and texts

- Contact/social links

4. Core Features

- Separate backend API (to update portfolio info later)

- React SPA with routing (React Router)

- Responsive design

- Download CV buttons (2 versions)

- Language switch (EN/DE)

- Dark/light mode

- Content fetched via API

5. Update Mechanism

Include an admin-only function or config file in backend where user can update:

- Project list

- CV links

- Text content

Frontend will reflect updated data from API.

6. Hosting Plan

- Frontend: GitHub Pages (built using Vite or Create React App and deployed via GitHub Actions or manual script)

- Backend: Hosted on Replit, Render, or Railway (supports Node.js/Express API)

7. Assets Required

- Profile photo

- English & German CVs

- GitHub repo links

- Project descriptions

- Contact details

- Domain (optional)

Note: This modular portfolio setup helps Saad Hasan easily maintain and scale the site by updating backend content without needing to redeploy frontend. The multilingual and GitHub Pages-ready setup ensures professional visibility.