

Provisional Design Plan Brief

Tech Stack Overview:

Our project leverages a robust and scalable tech stack designed to meet the demands of modern web applications. We utilize PostgreSQL for persistence, hosted on Amazon RDS for reliability and scalability. The frontend is powered by React, facilitated by Vite for an optimized development experience and styled with Tailwind CSS for flexible UI design. The backend employs Python with Django, allowing for a flexible and widely tested web services implementation.

CI/CD & QA:

We embrace GitHub Actions for CI/CD, automating our integration and deployment pipelines to ensure code quality and streamline delivery. Quality assurance is bolstered by SonarCloud, integrated with GitHub Actions for continuous code quality checks and automated testing, ensuring our codebase remains clean and maintainable according to the defined Quality Gates.

Project Structure & Documentation: (not yet inserted into our workflow at the time of this demo)

Our monorepo approach facilitates streamlined codebase management, with JIRA for project tracking and GitHub Wiki serving as our knowledge center. Documentation is a key focus, utilizing tools like Sphinx, Docusaurus, Swagger, and SchemaSpy to provide comprehensive, up-to-date documentation across all project components. This meticulous documentation strategy ensures that our architecture, API endpoints, and database schema are well-documented and accessible.

Security & Performance: (not yet inserted into our workflow at the time of this demo)

Security is paramount, with strategies for authentication (OAuth), data encryption (TLS, AES), and dedicated quality gates scans through our OWASP oriented quality gates with SonarCloud for a continuous check of bad practices through the code development.

Monitoring, Logging, & Compliance:

Due to our inexperience with these tools, we as a team don't feel comfortable implementing the tools listed on this subsection. Nonetheless, we find value in acknowledging their usefulness and, after a careful investigation, we present them:

We propose APM tools (New Relic, Datadog) for real-time performance monitoring, centralized logging (ELK Stack, Loki) for efficient log management, and alerting mechanisms for immediate issue notification. Compliance with data privacy laws (GDPR, CCPA) and license compliance are rigorously maintained.

Accessibility:

We believe that our work must be enjoyable for everyone, as such we expect to be able to incorporate [WCAG guidelines](#) for accessibility and inclusion. An accessible internet is possible if we think about those commonly left behind while we are designing and coding.