Project Technology Overview

Programming Language

- **TypeScript**: TypeScript is a statically typed superset of JavaScript. It provides optional static typing, interfaces, and type inference, enhancing code quality and maintainability.

Frontend

- **ReactJS**: A powerful JavaScript library for building user interfaces, developed by Facebook. React uses a component-based approach to create reusable UI components.
- **Tailwind CSS**: A utility-first CSS framework for quickly building custom, responsive designs. Tailwind provides pre-defined classes to apply directly in your HTML or JSX.
- **Radix-UI**: An accessible UI component library for React that focuses on providing unstyled, high-quality components, letting you customize the styles as needed.
- **Storybook**: A tool for developing UI components in isolation. Storybook allows us to test, view, and document components independently, enhancing UI consistency.

Testing

- **Playwright**: A powerful, cross-browser testing framework for end-to-end tests. Playwright enables automated testing for web applications across multiple browsers and devices.

Backend

- **NestJS**: A progressive Node.js framework that uses TypeScript to build efficient and scalable server-side applications. NestJS offers a modular structure, which makes it suitable for complex projects.
- **PostgreSQL**: A powerful, open-source relational database system. PostgreSQL supports advanced data types and performance optimization, making it a strong choice for managing structured data.

Infrastructure and Deployment

- **Terraform**: An infrastructure-as-code tool that allows defining and managing cloud infrastructure in a declarative language. Terraform supports various cloud providers, including Azure.
- **Docker**: A platform that uses containerization to package and run applications. Docker allows consistent environments across different stages of development, testing, and production.

Authentication

- **Passport**: A Node.js library for user authentication. Passport integrates various authentication mechanisms, including OAuth, JWT, and local authentication, helping secure user access.

Cloud Platform

- **Azure**: Microsoft's cloud platform offering various services for computing, storage, and networking. Azure enables scalable, cloud-based deployments and robust service management.

Design Tools

- **Figma** or **Canva**: Collaborative design tools for creating wireframes, prototypes, and graphic assets for the application's user interface.

Project Management and Collaboration

- **Jira**: A project management tool used for tracking issues, assigning tasks, and managing workflows within the team.
- **Confluence**: A collaboration tool for creating, sharing, and organizing project documentation.
- **Bitbucket**: A Git-based source control tool for managing the project's code, enabling collaboration and version control.

- **Microsoft Teams**: A collaborative communication platform where we can chat, make calls, share files, and hold video meetings. It helps keep the team connected and organized.