Eden Presentation

By Ella Larson, Addison Getz, Gavin Walker, Leonard Huang, Alexander Carmichael

Topics

01. Description

02. Tools

03. Methodologies

04. Architecture Diagram

05. Challenges

06. Future Scope

Description

- Community for Plant Enthusiasts: Eden connects plant lovers who want to share their passion and experiences.
- Post & Share: Users can post plant photos, share care tips, and discuss challenges with others.
- Search & Tags: Easily find specific plant advice through tagged posts and a robust search feature.
- Customizable Profiles: Users can personalize profiles with bios, profile pictures, and featured posts to showcase their style.
- Discover & Learn: The discover page inspires users with plant posts, fostering a space for advice, stories, and gardening insights.

Tools

Github Repository ★★★★★

Purpose: Version Control System (VCS) for code collaboration and centralized storage.

GitHub Projects ★★★☆☆

Purpose: Project tracking and task management with boards and progress tracking.

Render ★★★☆☆

Purpose: Cloud platform for deploying and hosting web applications.

Handlebars.js ★★★★★

Purpose: Templating engine for rendering HTML with dynamic data.

PostgreSQL ★★★★☆

Purpose: Relational database management system for data storage and querying.

Node.js ★★★★☆

Purpose: JavaScript runtime for building server-side applications.

Mocha ★★☆☆☆

Purpose: a testing framework that provides the structure to run tests

Chai ★★★★★

Purpose: an assertion library that simplifies writing readable and expressive tests

HTML ★★★★★

Purpose: Standard markup language for creating web pages.

CSS ★★★★★

Purpose: JavaScript runtime for building server-side applications.

Express.js ★★★★★

Purpose: Web application framework for Node.js to simplify server-side routing and middleware

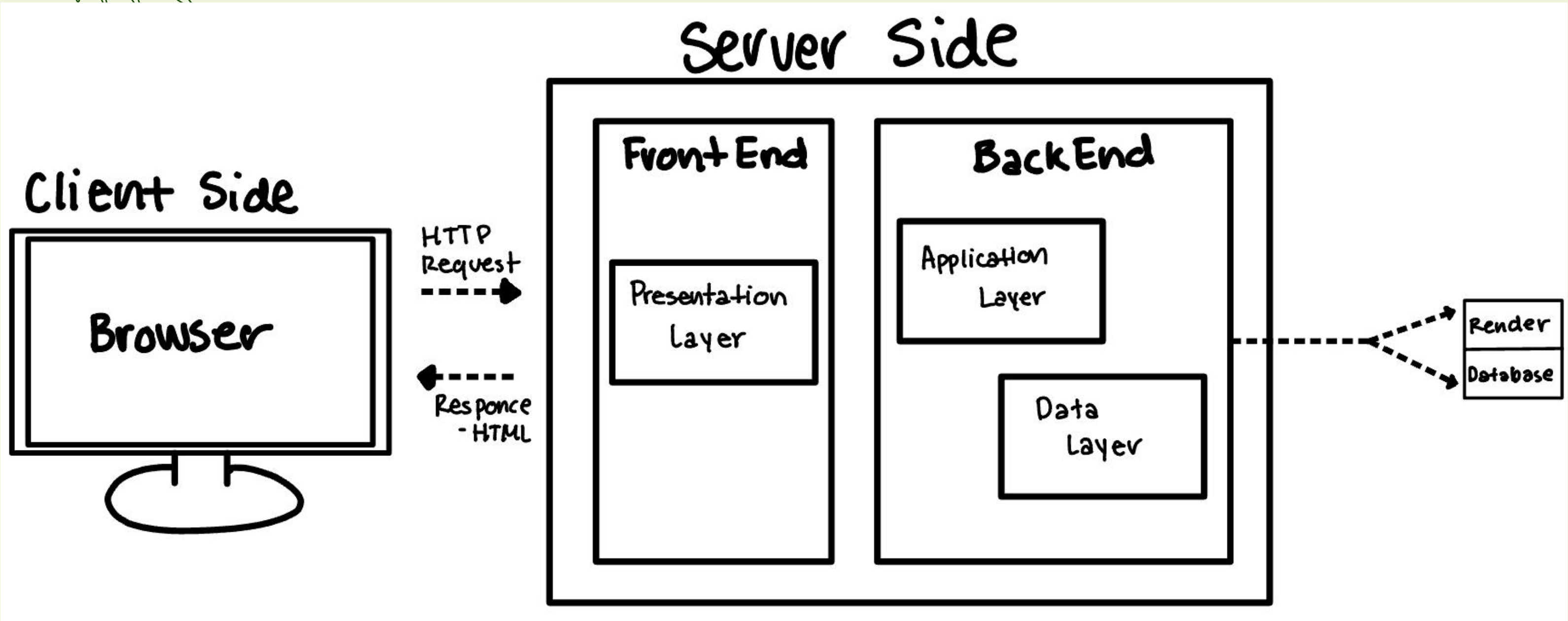
Teams ★★★★★

Purpose: to help teams communicate and collaborate online through chat, video calls, and file sharing.

Methodologies

- **Iterative Development** – ★★★☆☆ : Allowed continuous improvement based on feedback, but required consistent communication.
- **Pair Programming** – ★★★★★ : Effective for troubleshooting but time-intensive.
- **Peer Code Reviews** during Team Meetings – ★★★★☆ : Ensured code quality and reduced bugs.

Architecture Diagram



Challenges

1. **Communication :** Team members often worked independently without seeking clarification or updating others on their progress. This led to a lot of work having to be redone, wasting a lot of our time.
   - This issue was resolved by having a team discussion emphasizing the importance of clear communication.

2. **Time Management:** We struggled with team members working last minute, as well as not committing their work regularly. This caused confusion in what was being done and how far along we were in the project.
   - We addressed this by discussing committing work more often and agreeing to update the team on our progress more often.

Future Scope

Features we would like to add in the future...

Videos

Allow users to upload videos in their posts

Connections

Allow users to add friends so they can easily stay connected

General Care Page

Add a page with general plant care information for new plant lovers

Demo Time!