WordVerse

Sheetal Lodhi   Sonakshi Raj

August 25, 2024

Introduction

- WordVerse is a web application designed for playing the Wordle game.
- We've got two awesome modes for you: 'Guess My Word' where you try to figure out a mystery word, and 'I'll Guess Yours' where our chatbot tries to read your mind

Why WordVerse?

- WordVerse is an interactive game that requires real-time communication between the client and server. This makes it an ideal project for learning WebSockets, which enable bi-directional communication between the client and server

- It offers us a opportunity to learn golang.

Key Features of WordVerse

- Classic Wordle game.
- NPC Character (bot) that interacts with users and assists in gameplay.
- Responsive design optimized for different devices.
- Real-time communication using WebSocket.
- Backend implemented in Go.
- User-selected secret word functionality.

Technologies and Tools

- **Frontend: React, CSS.**
- **Backend: Go, WebSocket.**
- **Libraries: LaTeX, Gin.**

Challenges we have faced

1. Chatbot response handling.
2. Setting up React was a major pain point.
3. Implementing Gorilla Websocket.
4. Major challenge - Fixing one bug would often trigger another.
5. Balancing college and project life

Our Learnings So Far

1. **Teamwork**
2. **Error solving**
3. **Go**
4. **CLI: Mastered navigating command lines for efficient workflow.**
5. **Gorilla Websocket**
6. **Frontend tech: Learned HTML, CSS, JavaScript for UI building**
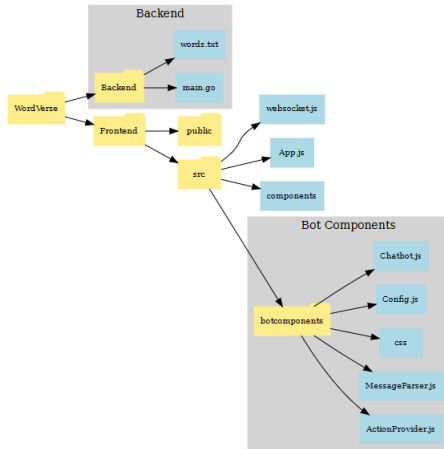7. **React**

Proud Moments

- **Custom-built NPC character (bot) with unique interaction logic.**
- **Efficient and clean file structure that made the project easy to manage.**
- **Successfully implementing real-time communication, enhancing user interaction.**
- **The successful deployment of the project and the positive feedback received.**

Project Timeline

- Week 1-2: Project planning and research
- Week 3-4: Frontend development
- Week 5-6: Backend development
- Week 7-8: Testing and debugging
- Week 9: Deployment and feedback

Understanding the File Structure

References

- **For Go:** `https://go.dev/doc/`
- **For Gin:** `https://gin-gonic.com/docs/`
- **For WebSocket:** `https://websockets.readthedocs.io/en/stable/index.html`
- **For React:** `https://react.dev/`
- **Postman API Documentation:** `https://www.postman.com/api-platform/api-documentation/`

Thank You