



# Gobang

High-Level Design Document

DAVID 1155216841

Chak Ming, 1155157388

Zizhe CHEN, 1155173938

Lana McKay, 11552169070

Serena Poonawalla, 1155216924

Department of Computer Science and Engineering

The Chinese University of Hong Kong

February 3<sup>rd</sup>, 2024

## Table of Contents

1 Introduction.....	3
1.1 Project Overview .....	3
1.2 Features .....	3
2 System Architecture .....	4
2.1 Technologies .....	4
2.2 Architecture Diagram .....	5
2.3 System Components.....	6

# 1 Introduction

## 1.1 Project Overview

Gobang is a Chinese strategy board game played on a 19 x 19 Go board. To set up the game, one player is relegated to white stones (traditionally Go pieces) and the other to black stones. The players take turns placing a stone on the board at empty intersections in the grid. The first player to connect 5 stones in a row, column or along a diagonal wins the game. Gobang is widely popular in Asia, and so for this project we are looking to recreate the household board game as an easily accessible and user-friendly web application. The web application will incorporate all the essential aspects of the gameplay, with additional features being built on top of it. These features include but are not limited to user accounts, online multiplayer, and readily available in-game statistics to improve player performance. We hope that our players will enjoy our modern twist on a timeless classic!

## 1.2 Features

The proposed Gobang web application is designed to deliver an intuitive and engaging user experience, achieved through the implementation of the following core and additional features:

### Primary Features (to be implemented in early release):

- **User Authentication:** This includes user registration, login functionality, and maintaining security and user data integrity.
- **Home Page:** This central hub acts as the initial touchpoint, guiding users to a variety of functionalities. It enables users to kick-start their gameplay, customize their settings, revisit past games through the replay feature, and compare scores on the leaderboard, among other options.
- **Game Interface:** The main interface for gameplay, consisting of a grid-based board allowing players to place their stones in alternating turns. The board's state updates with each move.
- **User Settings:** A settings page to customize the application experience, including adjustments for sound effects, volume, font size, and UI.
- **Game Mechanics:** Integral game functions such as player moves, game termination conditions, and move retraction capabilities.
- **Information Display:** Real-time updates and display of game-related information, such as start time, elapsed time, player names, current player and their stone type, and the current Gobang configuration.
- **Replay Functionality:** A feature enabling players to record and revisit past games for performance reviews.

- **Chatroom:** A real-time chat feature to facilitate textual communication between players during a game.

**Additional Features (subject to modification or exclusion based on development progress):**

- **Friend List and Direct Challenges:** A feature allowing players to add friends within the game and challenge them to matches directly could enhance the social aspect of the game.
- **Leaderboard:** In addition to the scoreboard, a leaderboard could rank players based on their scores, fostering a competitive environment and encouraging continuous gameplay.
- **Achievements and Rewards:** Implementing a system where players earn achievements or rewards based on their gameplay can increase engagement and provide a sense of progression.
- **Visual Assistant:** An alert system to notify a player of potential imminent loss.
- **Game AI:** A single-player mode with varying difficulty levels, guided by an AI opponent.

## 2 System Architecture

### 2.1 Technologies

#### Front-end:

This application uses HTML, CSS and JavaScript to build a basic UI and render simple game screens. Meanwhile, React and Bootstrap are applied to polish the UI and enable more natural interaction. Moreover, in the additional feature of this application, Three.js that is based on WebGL will be implemented for a more gorgeous and immersive game screen.

#### Back-end:

Node and Express are the frameworks that will be employed for the back end of this application so that a major part of the application will share the same programming language, which is easier for developers to work with. What's more, Python will be embedded into the back end to enable more advanced and cleverer AI players.

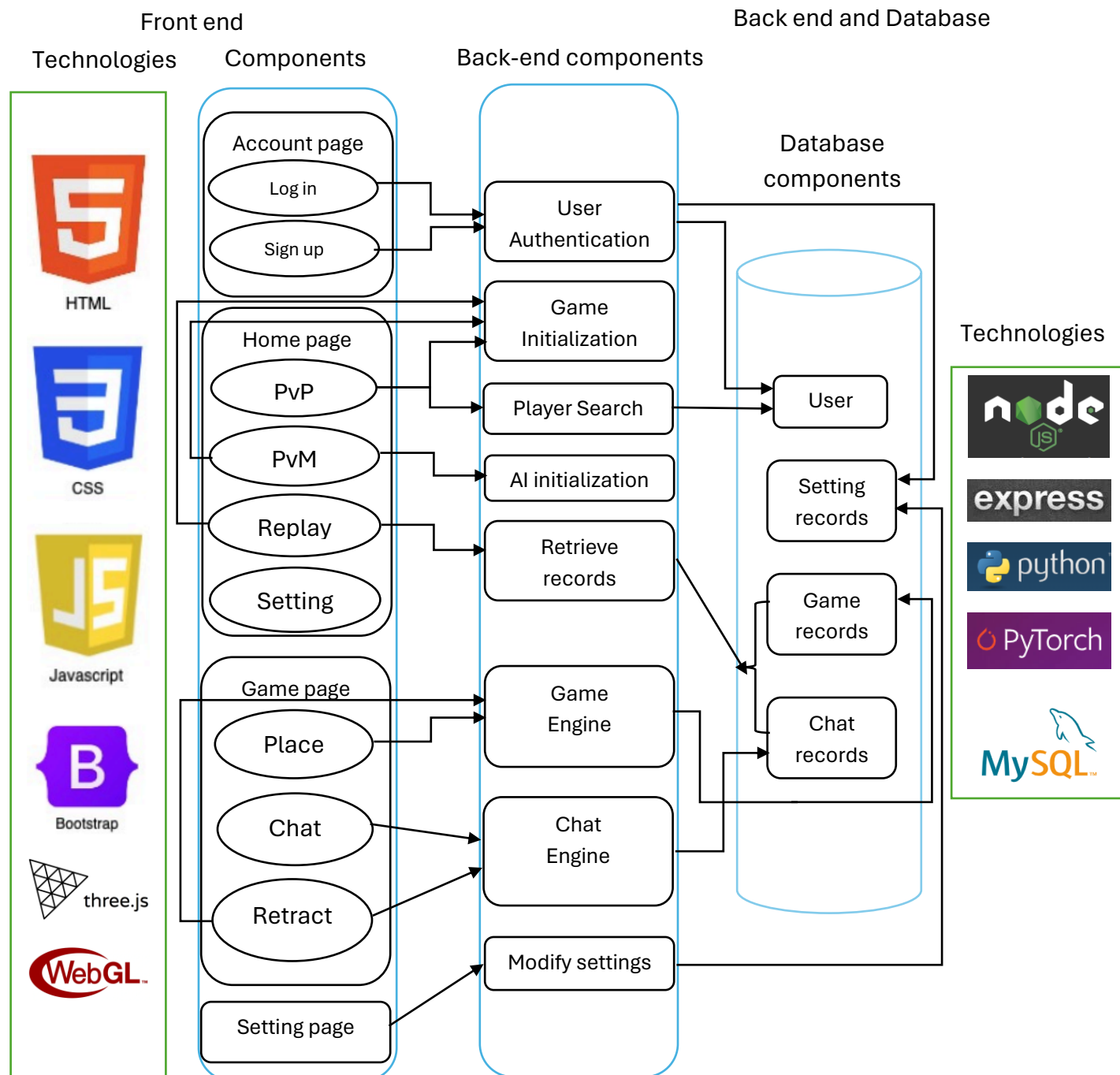
#### Database:

Because the number of entities in this application is not large and their relationship is relatively fixed, the database schema will be consistent during the development.

Therefore, this application adopts a relational database manager system, MySQL, to make the data clear and neat.

Last but not least, Git is used to manage the development of this application.

## 2.2 Architecture Diagram



## 2.3 System Components

Here is a more detailed page by page description of different components show in the Architecture diagram.

### Account Page:

There are two components, Login and Sign up, in the account page. They both interact with the User Authentication component in the back end. When the User Authentication component receive HTTP request with username and hashed password, it will retrieve User entity from database and compare the hash code. If they are same, then user can login, and the User Authentication component will retrieve the previous setting of this user to render subsequent pages with this setting. When the User Authentication component receive Sign up request with username and hashed password, it will store them in database and create a default setting for this user.

### Home Page:

The first component in Home Page is “PvP”, which is playing Gobang with human player. It will call the Game Initialization and Player Search component in the back end. Game Initialize component renders and sends the Game Page before another human player is found so that players can spend less time waiting for Game Page loading. Player search components changes the status of player in database and allows them to be paired with other players that are also searching for game. The second component is “PvM”, which is playing Gobang with machine agent. Besides Game Initialization component, it will call AI initialization component. AI initialization component decides parameters and models of AI based on difficulty setting of this player. The Third component is “Replay” the game record. It will also call Game Initialization component to prepare the game page for showing the game record and chat record that retrieved by calling Retrieve Record component. The fourth component is Setting, which will redirect user to Setting Page.

### Game Page:

The first component in Game Page is “Place” the stone. It interacts with the Game Engine, which has functions like verifying this placement, buffering the placement data, and updating the database. After Game Engine verifying that this placement is legal, the data will be buffered, and the game will continue. The second component is “Chat” with another player. It interacts with the Chat Engine, which will detect whether there are aggressive words, store the chat record, and display the chat message. The third component is “retract” the previous placement. Besides interacting with Game Engine to retrieve the data, it also interacts with Chat Engine to ask for the agreement of another

player. If another player agrees with retraction, then game will be reset to target status and the retraction will be recorded.