

Five In A Row

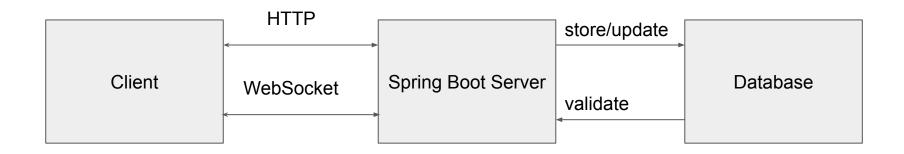
Team Goofy Object

Gaochao Wang, Yuxiang Chen, Tianyu Cao, Xiaoting Jin, Chikei Loi

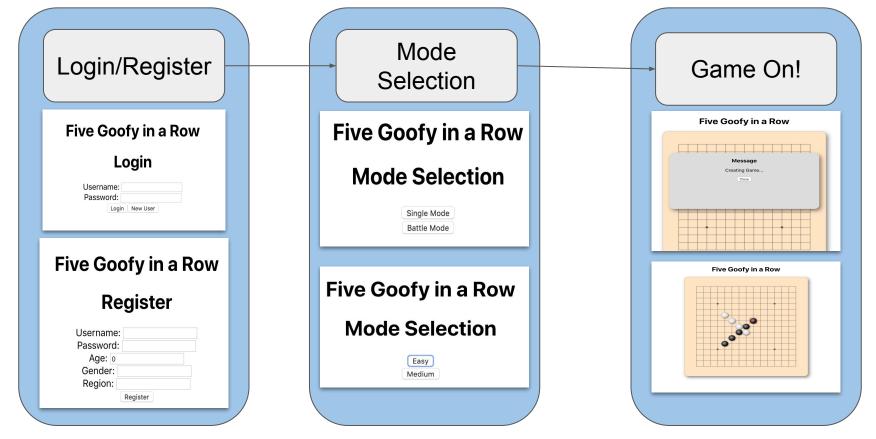
System Architecture

An interactive web game: multiplayer platform, Al player

React.Js, Spring Boot Framework, WebSocket, MySQL, Heroku Cloud



Component - Front End



Component - BackEnd Service

SpringBoot : RESTful HTTP service

"/register": doPost, userInfo as request body.

userInfo: username, password, age, gender, region.

applied Builder Pattern to build userInfo entity and store into MySQL

"/login": doGet, userInfo as request body

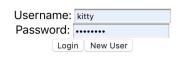
check input username and password matches with the one in database after login, user is able to choose single mode or battle mode to play

Five Goofy in a Row Register



Five Goofy in a Row

Login



Component - Database

JDBCTemplate

data schema

Register/Login

Win Rate - In Progress

Win rate: 99%

Ranking - In Progress

Rank: 3

User Table	
username VARCHAR(255)	
password VARCHAR(255)	
age	INT
gender	VARCHAR(255)
region	VARCHAR(255)
wins	INT
loses	INT

UserInfo username : String - password : String age:int - gender : String - region : String - wins : int loses int + UserInfo(builder : Builder) + getRegion(): String + setRegion(region : String) : void + getUsername(): String + setUsername(username : String) : void + getPassword(): String + setPassword(password : String) : void + getAge(): int + setAge(age : int) : void + getGender(): String + setGender(gender : String) : void + getWins(): int + setWins(wins : int) : void + getLoses(): int + setLoses(loses : int) : void

Builder - username : String password : String - age : int - gender : String - region : String - wins : int - loses : int + newlnstance(): Builder - Builder() + setUsername(username : String) : Builder + setPassword(password : String) : Builder + setAge(age : int) : Builder + setGender(gender : String) : Builder + setRegion(region : String) : Builder + setWins(wins : int) : Builder + setLoses(loses : int) : Builder + build(): UserInfo + buildFromJSONObject(userInfoObject: JSONObject): UserInfo

Component - Game Service

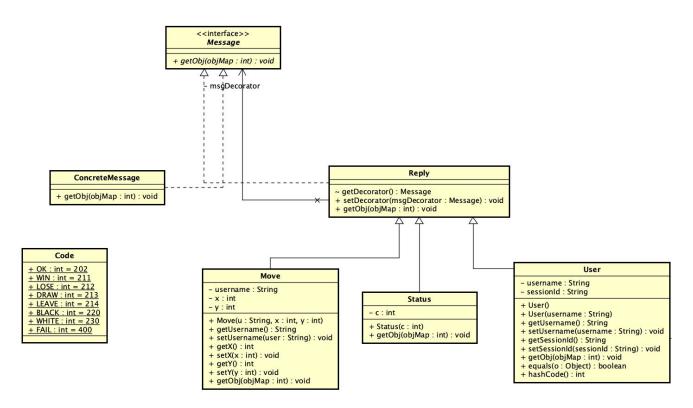
Game Controller:

- void createAiGame(User user)
- Void addToQueue(User user)
- void putPiece(Move move)

Web Socket:

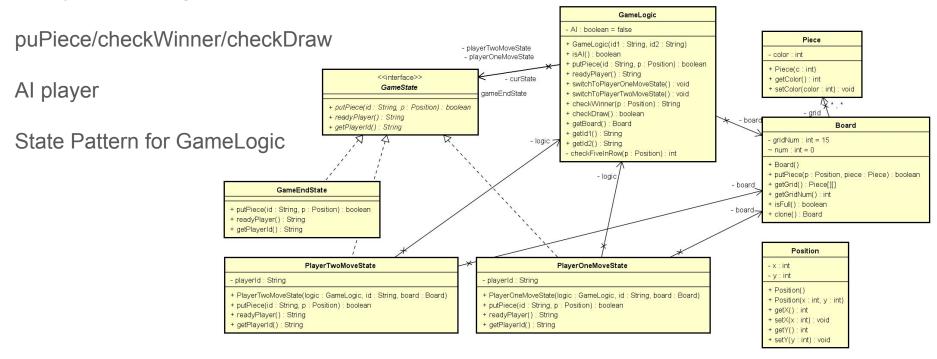
- Join notify players if game created
- Update player's movement (WIN,LOSE,DRAW,MOVE)

Design Pattern: Decorator



Component - GameLogic

Entity: GameLogic, Piece, Board, Position



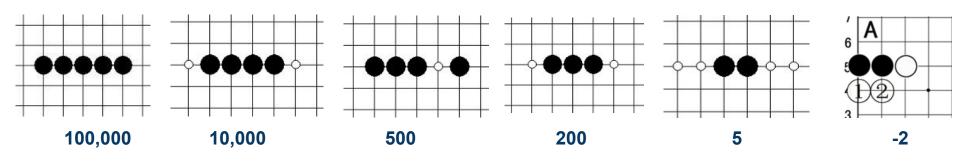
Component - GameLogic

Factory Pattern for Black/White Piece and Board AbstractPieceFactory Abstract Factory Pattern + getBlackPieceFactory(): PieceFactory + getWhitePieceFactory(): PieceFactory BoardFactory - BoardFactory() <<interface>> + createBoard(): Board PieceFactory + createPiece(): Piece WhitePieceFactory BlackPieceFactory Board + createPiece(): Piece + createPiece(): Piece - gridNum : int = 15 ~ num : int = 0 + Board() Piece - grid + putPiece(p : Position, piece : Piece) : boolean - color : int + getGrid(): Piece[][] + getGridNum(): int + Piece(c:int) + isFull(): boolean + getColor(): int + clone(): Board + setColor(color : int) : void

Component - Al

AlCommonMethod:

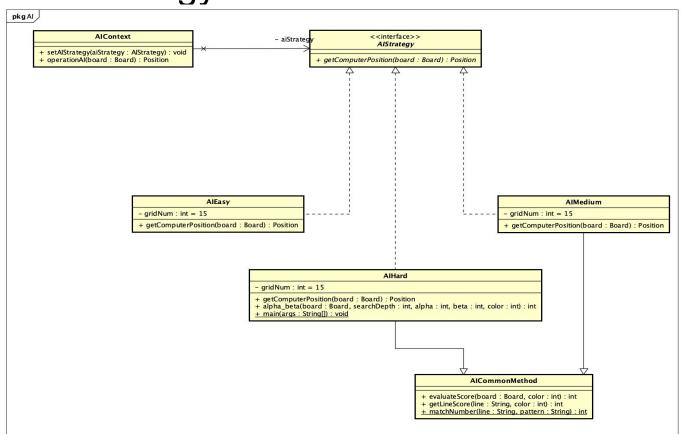
- evaluateScore(Board board, int color)
 - Evaluate current chess on the board and calculate scores for both black and white chess
- getLineScore(String line, int color)
 - Calculate total score for one direction (horizontal, vertical, diagonal)
- matchNumer(String line, String pattern)
 - Judge the type of chess shape (three in a line, four in a line, five in a line)



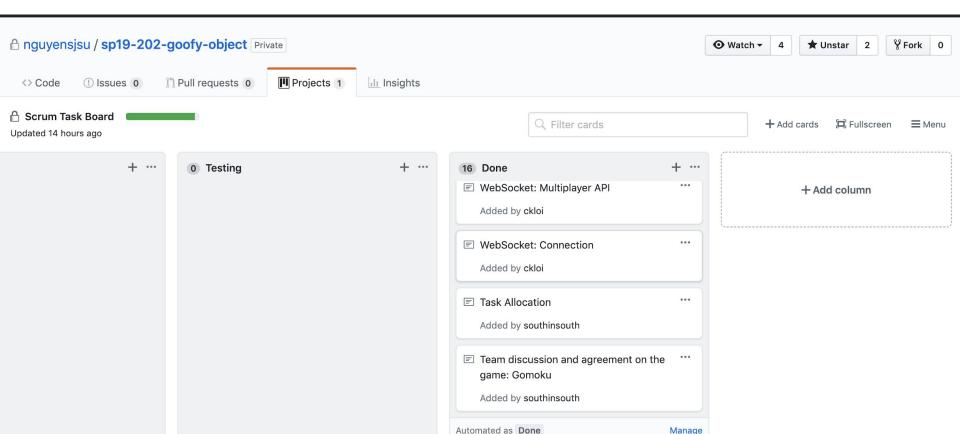
Design Pattern - Strategy

Strategy Pattern

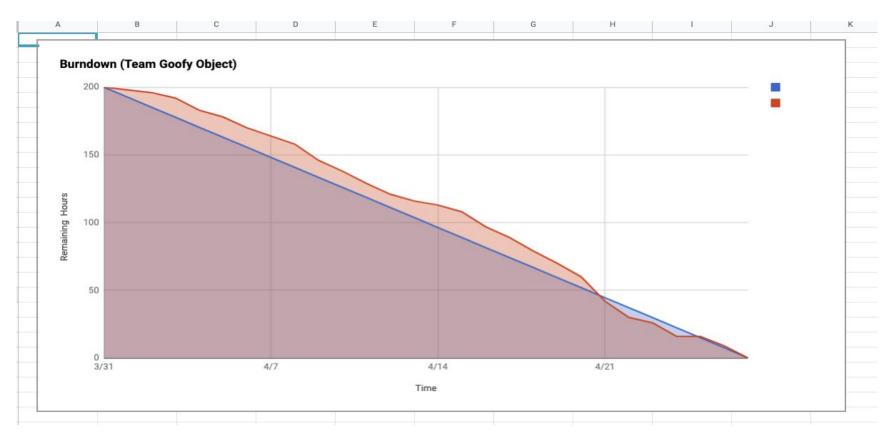
for Al



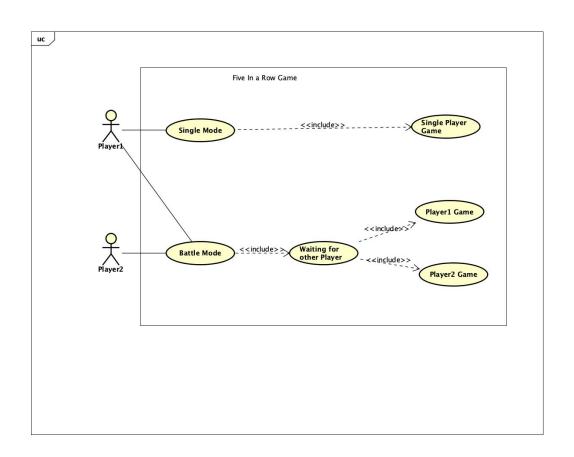
Scrum Board



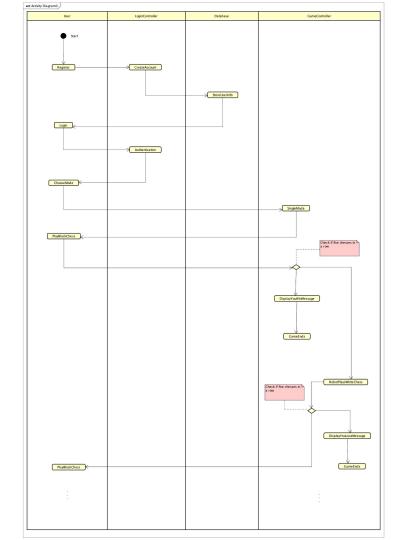
Burndown Chart



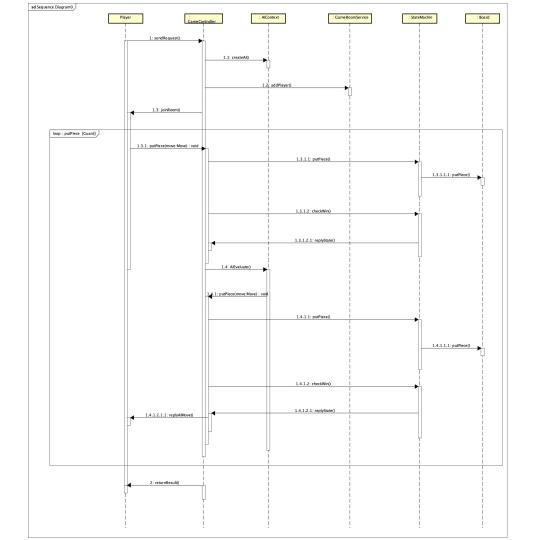
Use Case Diagram



Activity Diagram



Sequence Diagram



Demo

Thank you!

Try It Out!

https://five-in-row-202.herokuapp.com