# Portal Chess

Strike Team Java

# Introduction to XGame

Play Portal Chess online

Challenge other users

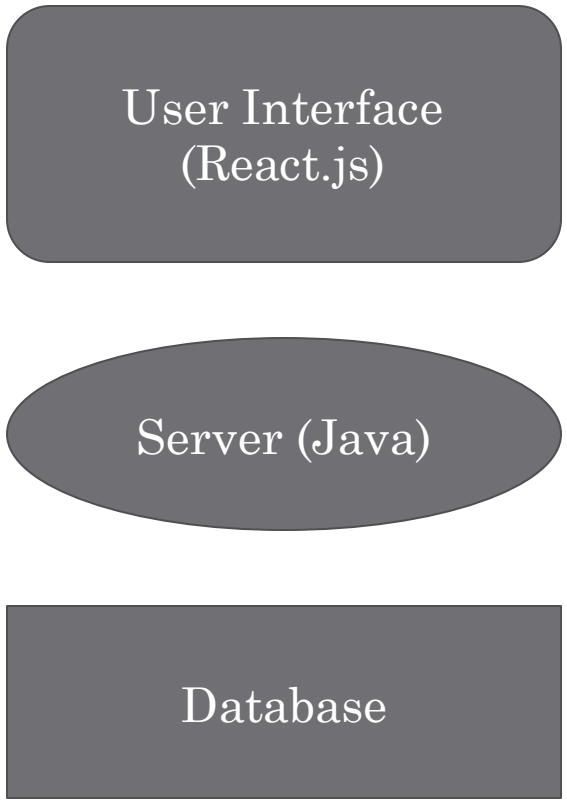
Create an account

View your match history

# What is Portal Chess?

- Has the same base rules as ordinary chess
- Each player has two portal pieces they start with, can place them anywhere on the fourth row at the start of the game
- Player can only move their own colored portal, same moves as a King
- If a piece lands on a portal, then it is teleported to the other portal of the same color unless it is blocked. Piece comes out of portal in same direction it moved into the portal.

# High Level Project Design




UI will be built with React and will send API requests to the JAVA server to obtain information to be displayed to the user

Restful API that will provide all the information about the application to the frontend. All logic for gameplay will be here.

Database will store information about games, invitation, and users

# User Stories

## Must Have

- Create an account
  - Register email
  - Create username
- Log in to the account
- Create match
- Invite opponents to the match
- Start game when the first opponent accepts the invitation
- Play Portal Chess

## Should Have

- Send invites to multiple users
- Accept/ reject invites from another user
- Close and return to matches at will
- Know when a game is over and who has won

# User Stories (cont.)

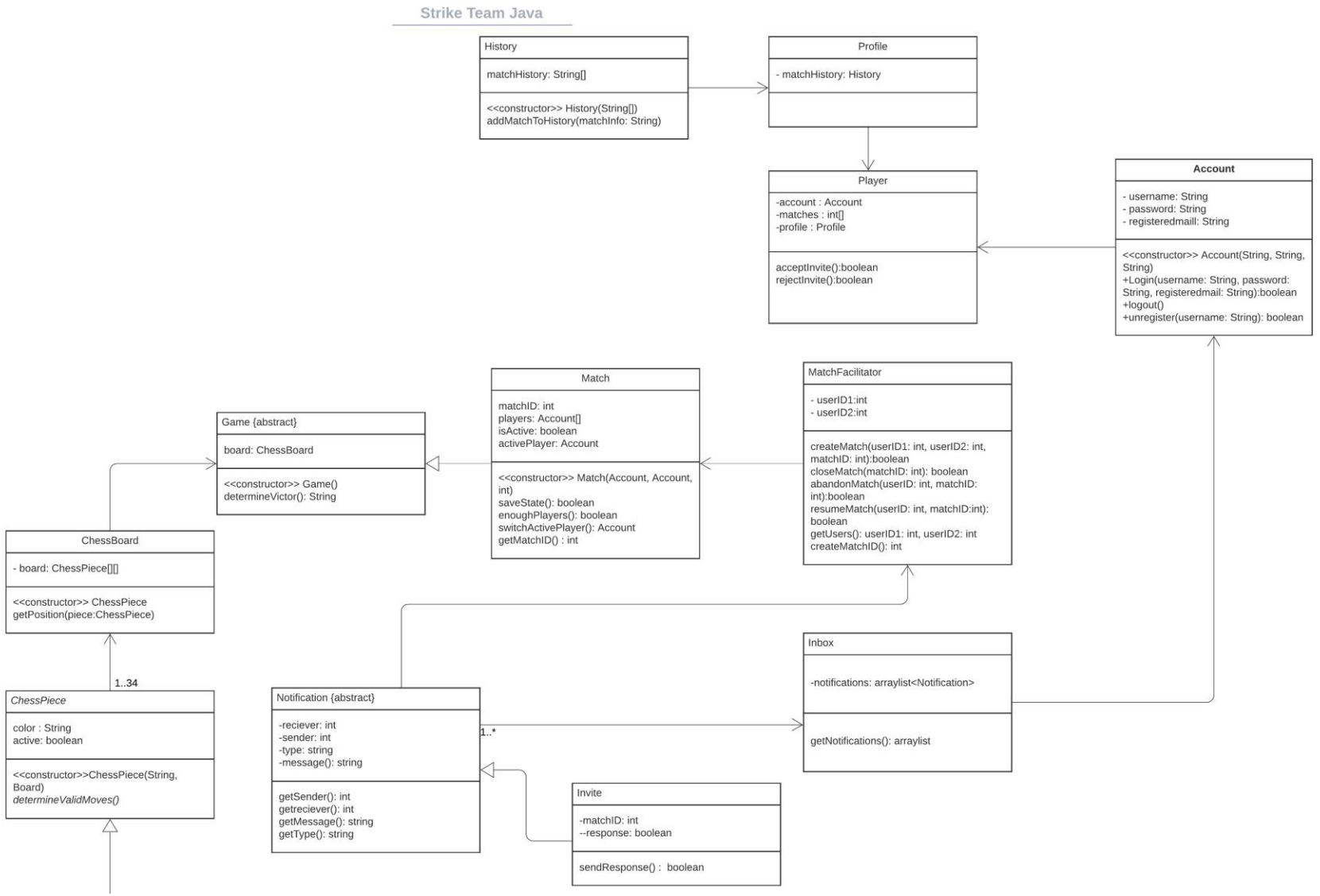
### Could Have

- Be able to log out
- Unregister account
- Switch between multiple games at once
- Abandon a game at any time

### Would Have

- View game history- Players, start and end date/times, winner/loser of the match, whether a game was abandoned
- Be able to tell whose turn it is

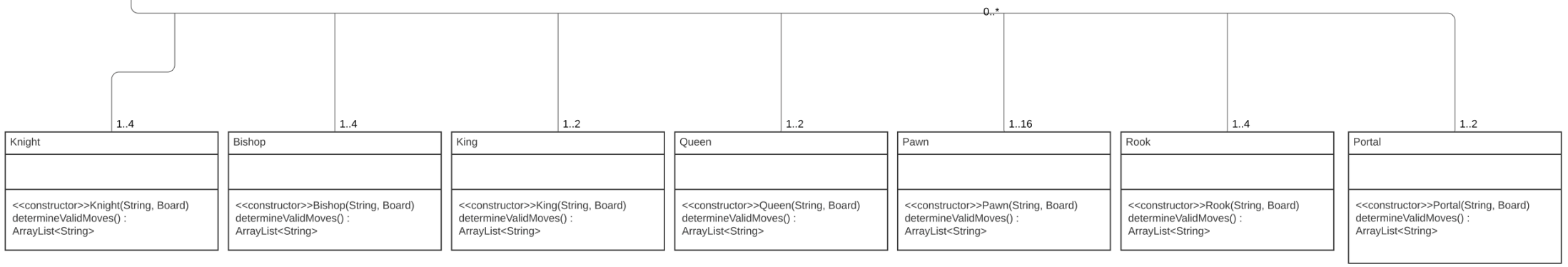
# Class Diagram

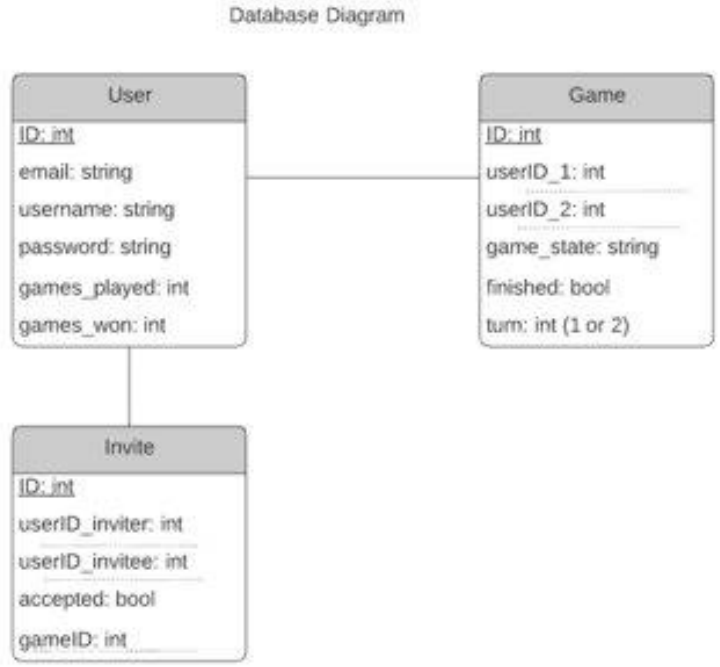



To next slide

# Class Diagram

To previous slide



# CRC Cards

| Account | |
|---|---|
| • Create account<br>• Login to account<br>• Log out of account<br>• Unregister account | • Player |

| Player | |
|---|---|
| • Accept invites<br>• Reject invites<br>• Notify inviter if invite is rejected | • Account<br>• MatchFacilitator |

| Profile | |
|---|---|
| • Contains win/loss history from matches<br>• Contains abandoned match history | |

| MatchFacilitator | |
|---|---|
| • Create match<br>• Close match<br>• Abandon match<br>• Resume saved match<br>• Switch between matches | • Player |

# CRC Cards

| Match | Game |
| --- | --- |
| • Save state of the match in progress<br>• Start when enough players join<br>• Determine if game is over<br>• Reflect victor of game determined by Game (if applicable)<br>• Keep track of active player | • MatchFacilitator |

| History | |
| --- | --- |
| • Stores match history | • Profile |

| Abstract<br>Game | Match |
| --- | --- |
| • Determine victor (if applicable) | • Match |

| Inbox | |
| --- | --- |
| • Contains notifications | |

| Invite | Notification |
| --- | --- |
| • Be sent to player's inbox | • Player |

| Abstract<br>Notification | Invite |
| --- | --- |
| • Be sent to player's inbox | • Player |

# CRC Cards

| ChessBoard | |
|---|---|
| • Show location of each piece | • ChessPiece |

| Abstract<br>ChessPiece | Knight, Pawn, King, Queen, Rook, Bishop, Portal |
|---|---|
| • Store color | • Knight<br>• Pawn<br>• King<br>• Queen<br>• Rook<br>• Bishop<br>• Portal |

| Knight | ChessPiece |
|---|---|
| • Determine valid moves | |

| Pawn | ChessPiece |
|---|---|
| • Determine valid moves | |

| Bishop | ChessPiece |
|---|---|
| • Determine valid moves | |

# CRC Cards

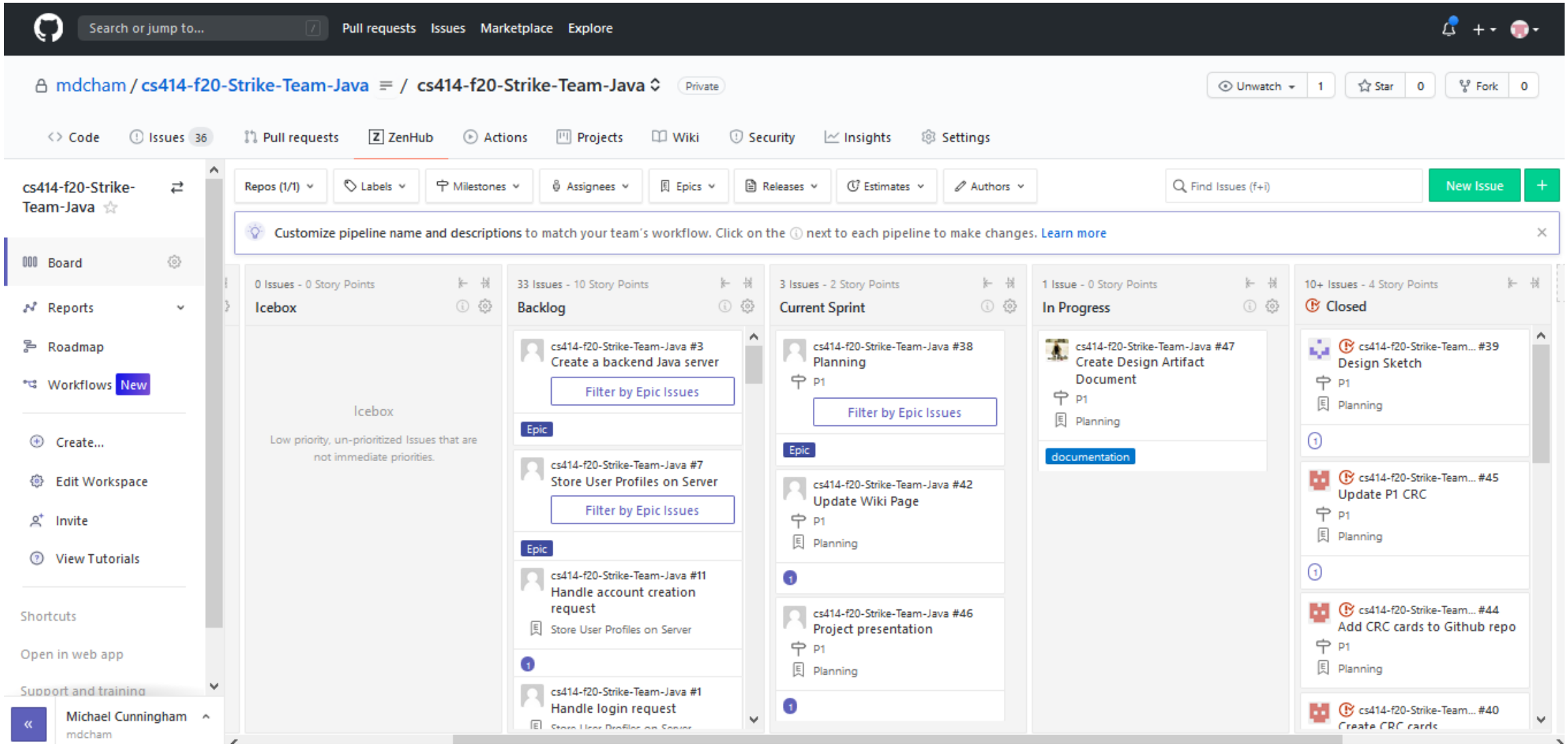
| King | ChessPiece |
| --- | --- |
| • Determine valid moves | |

| Queen | ChessPiece |
| --- | --- |
| • Determine valid moves | |

| Rook | ChessPiece |
| --- | --- |
| • Determine valid moves | |

| Portal | ChessPiece |
| --- | --- |
| • Determine valid moves | |

# Kanban Board



# Scrum results

- **09/10/2020 - Setup Github repo, created epics/tasks in Zenhub**
- **09/14/2020 - Created questions and scheduled a meeting with PO**
- **09/17/2020 - Meeting with PO, reworked user stories gave each of them a priority, got answers to questions**
- **09/22/2020 - Created CRC cards for all the planned classes, closed issue #40**
- **09/22/2020 - Created wiki page on Github, started uploading documents to Github, started on UML diagram**
- **09/23/2020 - Finalized UML diagram, discussed presentation possibilities**
- **09/24/2020 - Created PowerPoint presentation, set video recording time for everyone**

# Going Forward

- Setting up a Maven build for server
- Create a React app for the client
- Following proper Git etiquette
  - Reviewing other users' code
  - Never updating Master directly