Language: Java

Sign-up Server:
- Purpose: allowing users to sign up their AI players to tournament.
- Representation:
  o Using TCP server for users to transmit their AI players.
  o The AI players should be subclasses under one standard interface in Java.
  o Storing the AI players in Game Server.
- Role: an easy way for users to sign up and transmit their AI players to the tournament, which are ready for game play.

Game GUI:
- Purpose: displaying the current game state of the tournament.
- Representation:
  o Using Java GUI libraries to draw the current game state.
- Role: a graphical interface for users to view the tournament of the Fish game.

Game Logic:
- Purpose: letting signed-up AI players to make movements in order.
- Representation:
  o Perform start-up settings.
  o Retrieve data from Game Service.
  o Execute the movement from next AI player.
  o Updating Game GUI according to the game state from the Game Service.
  o Update the data in the Game Service according to the latest game state.
- Role: middle layer that manipulates both Game GUI and Game Service.

Game Service:
- Purpose: maintaining data of signed-up AI players and the current game state.
- Representation:
  o Storing data in a JSON file.
  o Data can also be created, read, updated, deleted.
- Role: a place to store data since we will not be using database, according to Piazza.

Description:
The Sign-up server is used during the preparation state of the tournament, which is used for collecting data of AI players for the game play, some of the collected data will be stored in the Game Service. Game GUI, Logic and Service are the standard "three-layer" archtecture where the Game GUI is the presentation layer, Game Logic is the application layer, and the Game Service is the data layer. Game GUI will display the current game state fo the tournament graphically. Game logic wil perform the logic of the game, retreive & update the data in the Game Service, and update the Game GUI according to the Game Service. Data in the Game Service will be stored in a JSON file, and it served as a "database" where data can be created, read, updated and delete.