System Component Description

Andrew Leung and Fiona Gridley

In order to create a tournament system for the Fish game, there are the following software components: There is a *tournament manager* which:

- Keeps track of participating players and their mappings to game pieces
- Gives out *play API* keys and assigned penguin-colors to participating players
- Houses and makes calls for running a game to the *referee* which is an interface for:
    - Creating a board
    - Creating and maintaining mappings from players to their pieces
    - Running placement rounds and handling player turns in interaction with the *game service*
    - Keeping score and identifying winners and losers
- In interaction with a *prize service*, pays the winner

A *game service* which handles the application of actions to a given game board:

- Placing and moving penguins
- Enforcing valid moves and actions

In terms of exposed APIs, there should be a *tournament API* for allowing user interaction with the tournament via the *tournament manager*:

- Signing up for a tournament and charging entry via a *payment service*, giving back access info
- Forwarding new player information to the *tournament manager*

There should also be a *play API* which exposes endpoints for users to play the game:

- Get the board state → respond with the current board and scores, whether it's your turn
- Place a penguin → respond with an updated board or a rejection
- Move a penguin → respond with an updated board or a rejection

The view layer will consist of two components to handle the interactions between the user and game

- View-only visualization of game play to allow human users to watch their AI players' performance
- Command-line interface for direct interactions between human and software which including:
    - Signing up for tournaments, making payments, plugging in AI players, and receiving prize money

The model layer will consist of data definitions and current game state management, because there is no need for a database as none of the game or player information needs to persist between games.

- Game components
    - Board: Tiles, Fish (Location), Penguins (Color, Location)
    - Referee: Scores, Player turn, Current board
    - Player: Score, Penguins
    - User: Players
- Game Rules
    - Valid moves
    - Scoring and ranking (prize assignment)
    - Penguin number and color assignment
    - Player turn order and current player turn assignment
    - Game end state