

Go Fish Game API

This API is to allow the creation of a web/REST based client, so a user can play on the internet. The functions required for this to happen are pretty simple, as go fish is a simple game. Most of the endpoints are for the client to request details, with only a few for user interaction.

User Stories

1. As a player, I want to be able to begin a new game, with custom numbers of bots and names, as I want this game to be personalized
2. As a player, I want to know the scores of all the players at all times, so I know how I'm doing
3. As a player, I want to know what's in my hand, that way I can know what to ask for
4. As a player, I need to be able to find out the number of cards in my opponents hands so I can formulate a strategy

Client Needs

The client needs data on the game in order to show the appropriate information to the user, this includes: - The number of cards in each player's hand - The number of cards in the deck - The bots' actions on their turns - The score of each of the players - The player's hand

All of these things should be able to be retrieved, but not changed. They are controlled by the server. This makes them perfect candidates for GET requests

User Needs

The only inputs from the user into the game are the initial setup (how many bots, the names of the bots, the name of the player), and when the player is asking a bot for a card, the bot and the card. This means that we don't need very many PUT/POST requests.

Technologies

Client

This would probably be implemented in JavaScript and HTML/CSS. The client would load the page, then make calls to the server to retrieve information

Server

This server could be written in any language that can create a REST API and can interact with Python (unless you want to rewrite the game in a different language). Obviously writing the server in Python would be easier, as there would then be no need for an FFI. Some Python frameworks that would make it fairly easy include: - web.py - Flask - Bottle

Changes Required

The server will need some way to store the current games, and use unique keys for each game to access them. This will necessitate the use of a database of some sort to store and retrieve this information.