War Card Game Simulator

Author: Ekram Bhuiyan

Date: May 26, 2020

**Users** -

1. The player who can interact with the system from their desktop device (not supporting mobile front ends at the moment)
2. For this version of the system, we won’t support player authentication and if another player with a link to access the system tries to play the game, they will be blocked until the first player completes their session.
3. The authorization the player owning the session has will be in terms of:

* Ability to interact with a similar adversary
* Ability to type in their name
* Ability to see the arena
* Ability to end the session
* Ability to consent to some tracked meta information being emailed to them through the action of providing their email address
* Ability to draw cards to play against an adversary

There is only one player who can own a session at a time, and they will be the ones interacting with our system.

**UI Flow** -

1. User sees page that allows them to begin a session or not. A simple button will do initially. If the user is not able to begin a session (due to another user being active), the system will prompt them with why it was the case that they can’t begin the game. If there is no other player owning the session, then they may begin. The assumed language will be English and the system will only support English for now.
2. A similar UI will be presented as what is shown here:

<https://cardgames.io/war/>

1. The user will have the ability to type their name (this is required to start playing the game), and they will have the option to end the session at any time. The user will
2. When the user exits the session, they will be prompted with a modal that will ask them if they’d like to have some meta information mailed to them. This will be for now:

* How long they were playing (in seconds)
* The name of the user that they selected

**Implementation details at a glance**

The difficult part will be setting up the front end animations. For simplicity, I will use VueJS and try to rely on third-party plugins. Besides the animations, the task should be simple to build. I won’t be using translations outside of those defined for English.

I’ll use a python flash API to support backend calls. The whole system should be able to run locally.

Information I’ll have to track is related to an active game

* The user name
* The consent to having their time sent to them
* The time is taken
* Who currently owns the session
* When was the last active
* If a user has exited a session

A challenge will be to determine who owns the sessions. In this case, for instance, if a user refreshes the page, they should still own the session but if a user exited the game, or hasn’t been active for some time, then they should be kicked out if another user wants to play.

I can use WebSockets in this case for server-to-server messaging to possibly kick users out, and I’ll create a SQL DBMS to track the state I need.

The application will rely on event-driven interactions with users polling to see if they can have access to the game of not.