

**SOFE 4790U: Distributed Systems (Fall 2023)**

**Instructor: Dr. Ahmed Badr**

**Assignment #x**

**Honour code**: By submitting this assignment, I (name and banner id# below) affirm this is my own work, and I have not asked any of my fellow students or others for their source code or solutions to complete this assignment, and I have not offered my source code or solutions for this assignment to any of my fellow students.

LINK to VIDEO: <https://drive.google.com/file/d/1w4DgLcwGQWD3yskGwFNFpx03uSfUCyal/view?usp=sharing>

**Name**: Daniel Kamel

**Banner ID#**: 100754671

1. **Application idea**

The core idea of the application is a recreation of the R/Place social experiment hosted by reddit. The idea of the experiment is that every user of the site has the ability to place one pixel at a time on the shared canvas. Users would collaborate to place pixels together to form a larger image or they could choose to be destructive and overwrite the placements of other people. Instead of implanting through a GUI I created API endpoints that allow for more scripted placements.

1. **Describe the core functionalities (5 unique functions)**

|  |  |  |
| --- | --- | --- |
| **Endpoint** | **Request Type** | **Description** |
| /image | GET | Returns a jpg that represents the current state of place |
| /PlacePixel | PUT | Allows the caller to place a pixel in at the location of their choice with the colour of their choice |
| /ImagePlace | POST | Allows the caller to place an image they provide in base64 encoding at the location of their choice |
| /DeletePixel | DELETE | Removes the Pixel or image placed by another user revealing the original background underneath. |
| /LineDraw | PUT | Allows the user to place multiple pixels at the same time in the shape of a line defined by the constants the user provides in the endpoint. The lines will be of the colour the user provides in the endpoint similar to the /PlacePixel endpoint. |

1. **Challenges and solutions**

**Providing the users a view of the state of the board**

As the application is an API there is now traditional way for users to see what they are drawing. Users can see the board and their art by going to the /image end point. While the Host can see a live updating board displayed with pygame.

**Transferring and drawing the image**

A http request doesn’t typically carry multiple sets of data, such as files and JSON. Both were needed to correctly place the image. The solution was to include the image in the JSON. The image is expected to be encoded into a base64 Format and .

1. **Testing**

Describe each test and include screenshots.

For each of the end points a script was created to test.

The image endpoint can be test via a browser.

A screenshot of a computer

Description automatically generated

The Place pixel endpoint had two scripts to show placing pixels randomly and in a pattern

A screen shot of a screen

Description automatically generatedA screenshot of a computer screen

Description automatically generated

The PlaceImage endpoint had a script that encoded the test image and placed it on the board.

A cartoon of a person and person

Description automatically generated

A screenshot of a game

Description automatically generated

The DrawLine endpoint had a script that sends the request to draw a line across the board

A screenshot of a game

Description automatically generated

The Delete endpoint has a script that sends the request to delete the pixel for every pixel on the in the shape of a line from the initial line that the place cross pixel script used.

A screenshot of a computer game

Description automatically generated