

# COM S/ SE 319: Construction of User Interfaces

Fall 2019

## Individual Report for 2nd Release

Group No. 55

Project Title: Giphs Against Humanity

Name: Jamie Sampson

### **1. Worked on Story Cards No.**

- Story Card 8:
  - Name of the Story: Connect Work done in Story 5 w/ Socket Stuff
  - Assigned Team Member: Jamie Sampson
  - Tasks Accomplished For This Story Card:
    - On submit of giph, send giph info to other clients
    - Setup list of rooms and list of players and various properties stored on server side
- Story Card 10:
  - Name of the Story: As a user, I'd like to pick if I'm creating a game or joining a game
  - Assigned Team Member: Jamie Sampson
  - Tasks Accomplished For This Story Card:
    - Create 'Mode.js' that has two buttons "Join Game" and "Create Game"
    - Clicking "Join Game" renders new component "Join.js"
    - Clicking "Create Game" renders new component "Lobby"
    - On "Create Game" a default room
- Story Card 12:
  - Name of the Story: Make tests for /giphy API
  - Assigned Team Member: Jamie Sampson
  - Tasks Accomplished For This Story Card:
    - Add Jest dependency
    - Create test for /giphy api endpoint
- Story Card 13:
  - Name of the Story: As a user, I'd like the option to join more than one game room so that multiple teams can play at a time.
  - Assigned Team Member: Jamie Sampson
  - Tasks Accomplished For This Story Card:
    - Upon clicking "Create Game" multiple times, should make x many rooms that users can "Join"

- For each room, all prior functionality that worked in one room works for all rooms
- Story Card 15:
  - Name of the Story: As a user, I'd like a full party before being allowed to start a game
  - Assigned Team Member: Jamie Sampson
  - Tasks Accomplished For This Story Card:
    - Submit is disabled until 5 players have joined room
    - Upon submitting, emit start game message to all users to update their views
    - As users join the room, post their usernames to the lobby view
- Story Card 17:
  - Name of the Story: Connect Socket.io from server with frontend application
  - Assigned Team Member: Jamie Sampson
  - Tasks Accomplished For This Story Card:
    - Remove front end code on server side with socket.io
    - Add socket connection to first page and pass socket as props to other pages
    - Add callback handlers for server messages in componentWillMount()
    - Send message from client to server and back to client

## **2. Issues/Learning**

Learning how socket.io works on both the server and client side. This was difficult because of the necessary placement of socket callback handlers on the client side and how socket connections worked from one page to another. Using Jest as a testing framework was also new to me. As I continue to work with React, I'm gaining a better understanding on passing props and the flow of components from one "frame" to another, but as it is, it takes trial and error to get things to be broken up enough on the code side, but flow on the client side.

I continue to learn how express works and was excited to learn I could write JS functions in the same file as the endpoints / connections that we were establishing (is probably better if they weren't in the same file, but that's another wall to overcome).

## **3. Individual GIT statistics/analytics**

sampson1

13 commits

sampson1@iastate.edu

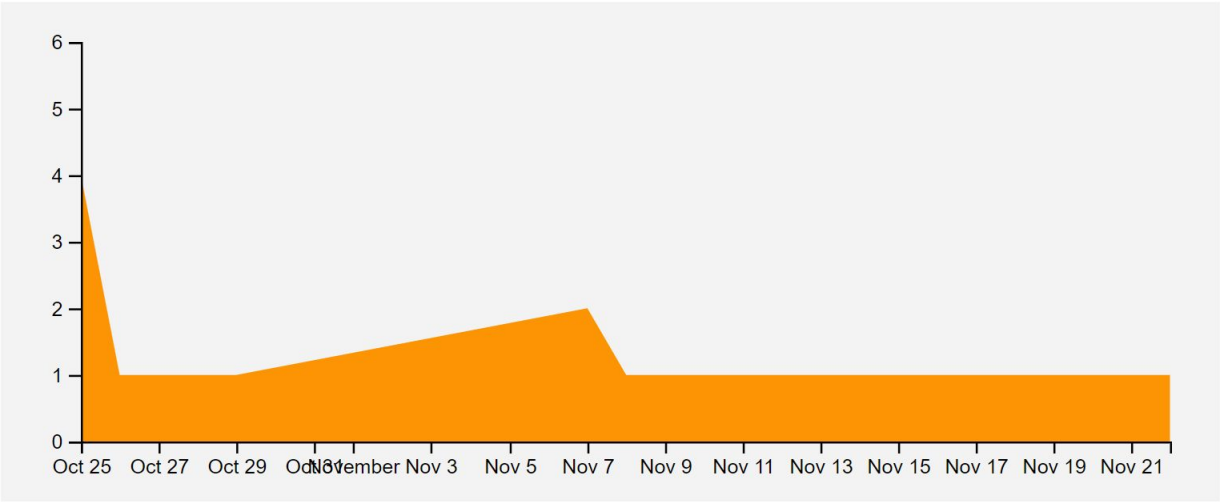
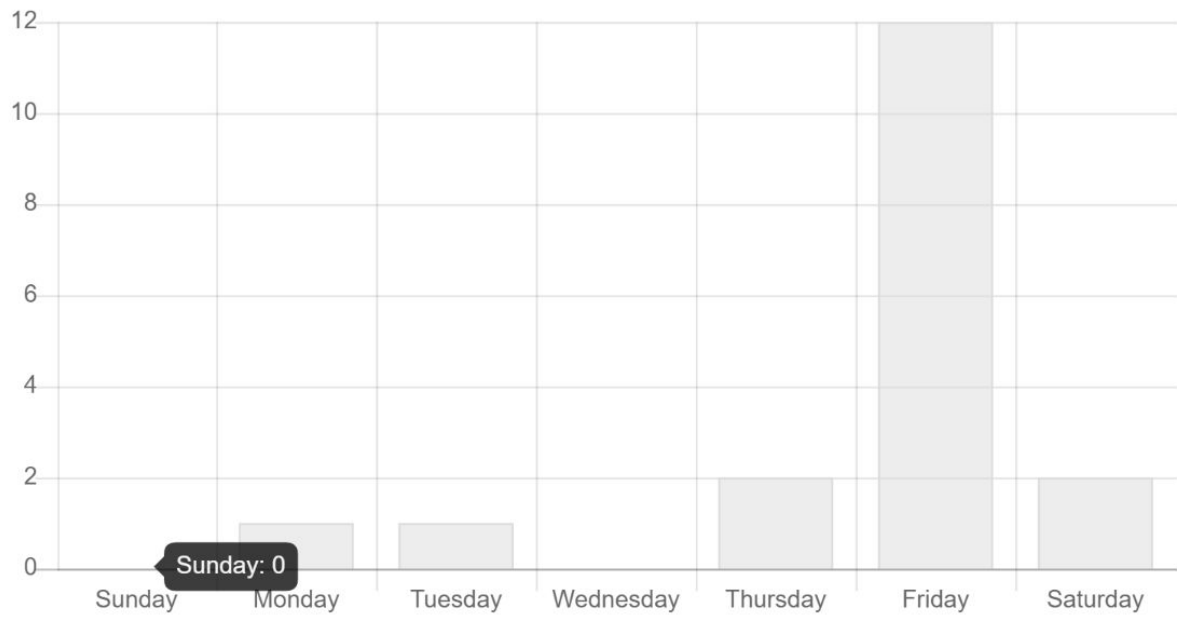
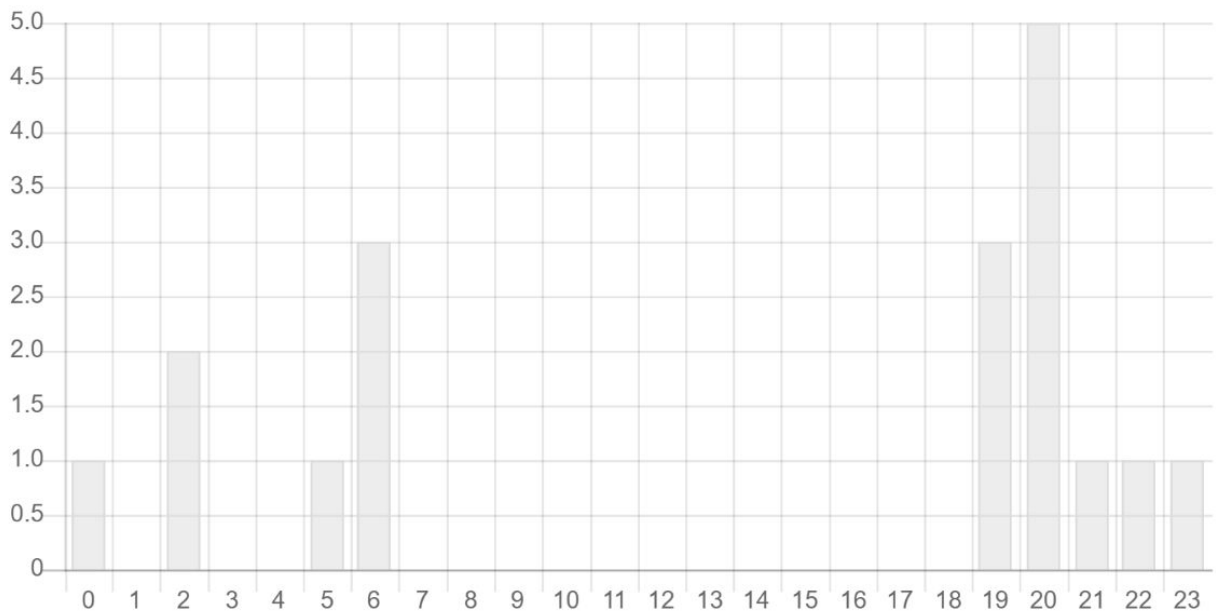


Figure 1: My Contribution

Commits per weekday

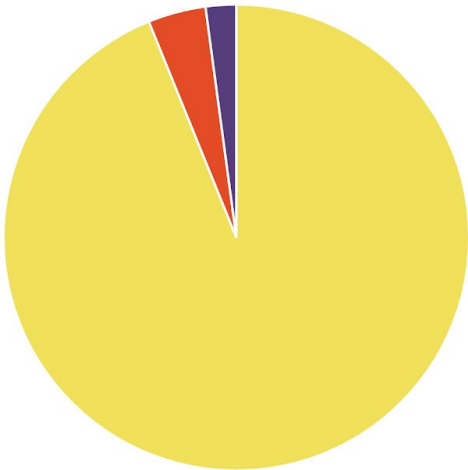


Commits per day hour (UTC)



Programming languages used in this repository

JavaScript	93.87 %
HTML	4.01 %
CSS	2.12 %



Commit statistics for master Oct 25 - Nov 22

- Total: **18 commits**
- Average per day: **0.6 commits**
- Authors: **2**

master

g55

Commits per day of month



Figure 2: Charts