# COM S/ SE 319: Construction of User Interfaces
# Fall 2019
# Group No. 55: 2nd Release Report

1. **Successful Implemented Story Cards for Demo 2:**
   - Story Card 1:
     - Name of the Story: As a user, I would like to connect to a game so I can play on the server side
     - Assigned Team Member: Emma Paskey
     - Tasks Accomplished For This Story Card:
       - Create button "Create Game" and "Join Game"
       - Setup Socket.io to take in multiple users to a single room
       - Handle success and errors when connecting

   - 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 9:
     - Name of the Story:   As a user, I'd like to pick my screen name to play the game
     - Assigned Team Member: Emma Paskey
     - Tasks Accomplished For This Story Card:
       - Restructure app to take advantage of page router (handled in story 12)
       - On the home.js create an input box and submit button
       - Handle invalid usernames (username < 1 character)
       - Upon submit, redirect to new page "Mode.js" and pass username to that page

   - 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 11:
  - Name of the Story: As a user, I'd like to enter a game code and join that game lobby
  - Assigned Team Member: Emma Paskey
  - Tasks Accomplished For This Story Card:
    - Handle invalid socket.io rooms
    - Add user to room and redirect to lobby
    - Add game code input screen & submit button

- 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. **Design Documentation (UML Diagram for Story Cards):**
   ○ *Figure 1 - Use case diagram for story card 1, 9, 10, and 11*
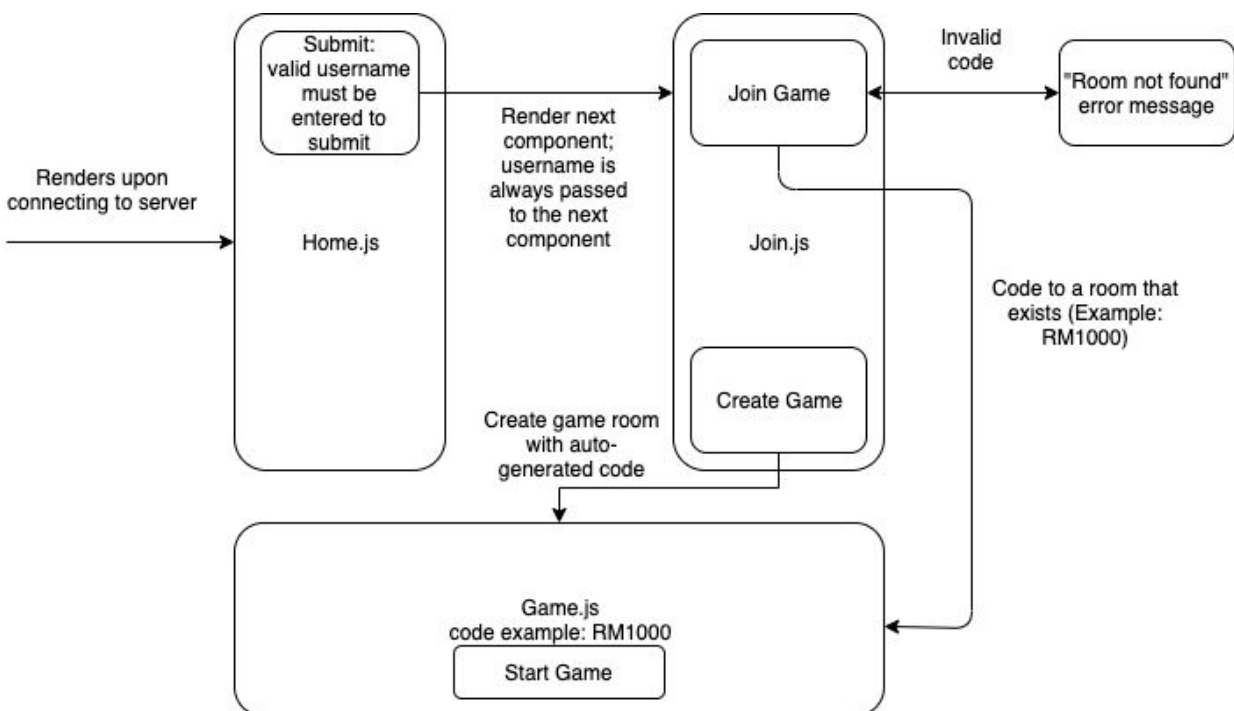


*1 - "As a user, I would like to connect to a game so I can play on the server side."*
*9 - "As a user, I'd like to pick my screen name to play the game."*
*10 - "As a user, I'd like to pick if I'm creating a game or joining a game."*
*11 - "As a user, I'd like to enter a game code and join that game lobby."*

*15 - "As a user, I'd like a full party before being allowed to start a game"*

### 3. UI Description with Screenshots:

**Story Card 9: As a user, I'd like to pick my screen name to play the game**

Home screen for web app. Prompts user to enter username. Submit only enabled if username is longer than 1 character.



*Figure 4: Story Card 9: As a user, I'd like to pick my screen name to play the game*

**Story Card 10: As a user, I'd like to pick if I'm creating a game or joining a game**

Options to pick to either join a game or create a new game are given to the user.

# Create or Join Game?
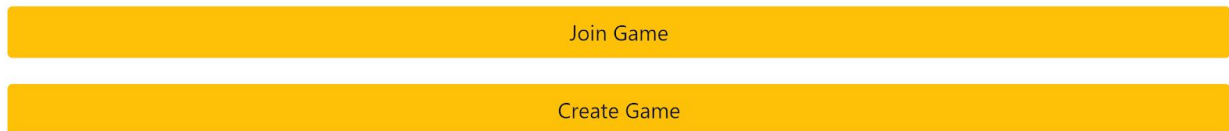
Join Game

Create Game

*Figure 5: Story Card 10: As a user, I'd like to pick if I'm creating a game or joining a game*

**Story Card 15: As a user, I'd like a full party before being allowed to start a game**

Clicking Create Game takes you to this screen and puts you as the first person in the lobby. You aren't able to start the game until a full part of 6 people have joined the lobby.

# Welcome to the Lobby!

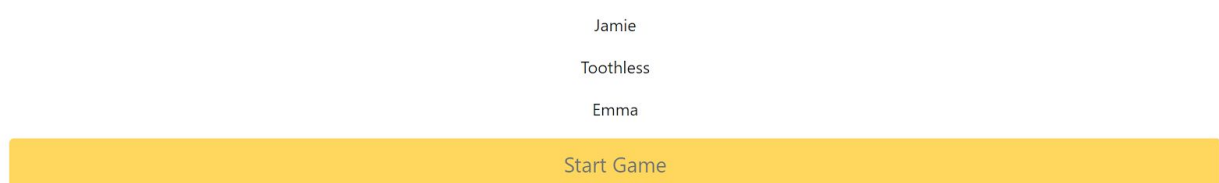# Game Code: RM1000

Jamie

Toothless

Emma

Start Game

*Figure 5: Story Card 15: As a user, I'd like a full party before being allowed to start a game*

**Story Card 11: As a user, I'd like to enter a game code and join that game lobby**

Clicking Join Game takes you to this screen where you can enter a game room code, invalid room will post an error message to the user that the room was not found.

# Enter Game Code

RM1000

Submit

Not Connected

*Figure 5: Story Card 11: As a user, I'd like to enter a game code and join that game lobby*

**Story Card 8: Connect Work done in Story 5 w/ Socket Stuff**

Once a player has submitted their giph, it shows up on all the other user's screens.
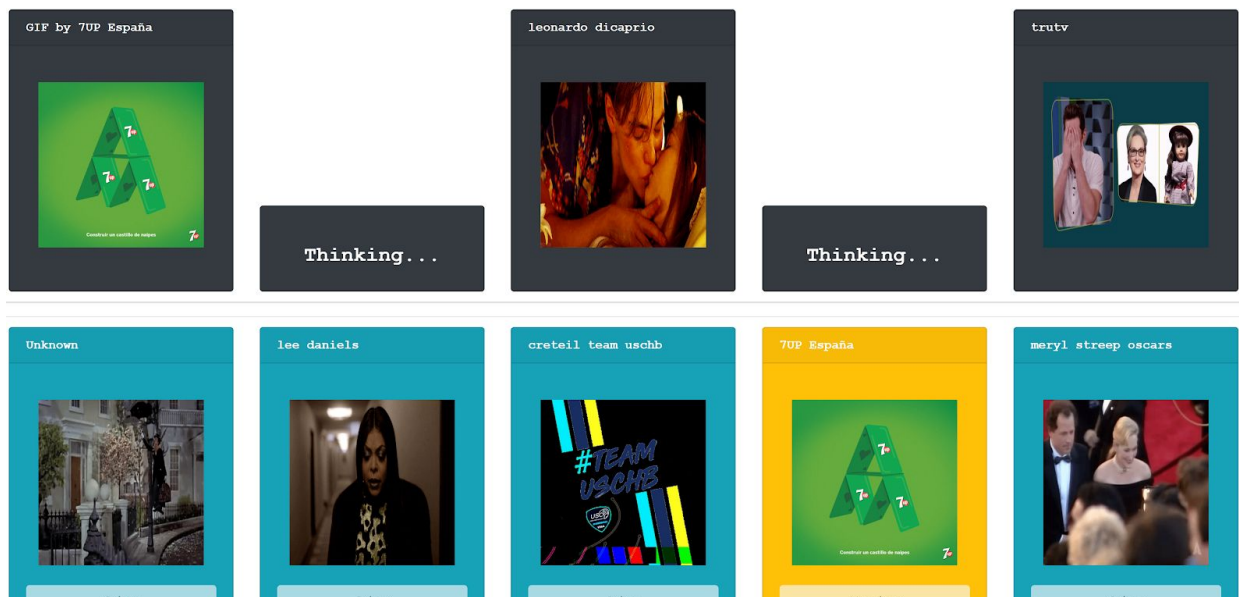


*Figure 5: Story Card 8: Connect Work done in Story 5 w/ Socket Stuff*

## 4. <u>Testing:</u>

For story card 12, if you run `npm test` inside the server folder, then it will run the test that was created for that card. All other testing was done by running `npm start` inside both client and server folders and using the` localhost:3000` endpoint. To test our application, do the following steps.

1. `npm start` inside client (will auto open localhost:3000`)
2. `npm start` inside server
3. Enter a username (leaving empty will throw back an error) and submit
4. Click "Create Game" which will take you to the lobby page for that game room
5. Open up 4 other tabs and click Join Game instead of Create Game

6. Enter in game code (found on the creator's lobby page)
7. Clicking submit (if invalid room will throw back an error) will take you to that lobby
8. Once all 6 people have joined, one the screens click "Start Game" and see all pages change to choose your giph
9. Select and submit a giph and see it appear on everyone's screen
10. Open new tab, and click Create Game and see that it takes you to a different lobby page

5. **Plans for 3rd Release:**
   ○ Story 3: As a user, I would like to have different text cards for each round so that they aren't the same every time
   ○ Story 6: As a user, I'd like to see the scoreboard as to see where I rank with other players
   ○ Story 14: As the judge, I'd like to select my favorite gif and end the round.
   ○ Story 7: As a user, I'd like to be able to receive a new card at the end of a turn so I have new giphs
   ○ Story 16: Setup Firebase Hosting