Team 314 Project specification

Proposal and Function List

● A copy of the project proposal you are implementing.

We plan to build an online-game distribution system called WeDraw, which offers interactive multiplayer gaming and social networking service for its users. We will use Django application framework and use the websocket Django Channels to handle real-time web networking feature. Our web application enables users to create game rooms, join an existing room, quick start, invite friends to a room, and play the game we will mainly focus on, “Draw & Guess”.

The game rule for “Draw & Guess” is pretty simple, each round one player draws a picture based on a certain topic within a period of time while others trying to guess what that topic is, points will be added once the player makes a correct guess. We plan to support the drawing feature using rich HTML5 Canvas Library such as Fabric.js. Besides, the game room may contain 2-5 people per game, once a player gets into the game room, he must get ready within a certain time and the host has the power of ejecting players. Players are also allowed to chat with each other during the game, APIs such as Tornado and gevent might be helpful to support our chat room.

Other than the main features discussed above, to make the game a little easier, the program will show more hints after a short period of time and the drawer can pass the current topic he/she finds it hard to draw. One way to improve the interaction between users is that we allow users to throw emojis to the drawer, such as “flowers” for “Good drawing”, “eggs” for “What is that?”. Besides, we could also let users share their drawings to their social network such as facebook if they really enjoy their drawings by using The Facebook Crawler. We will also introduce background music to our application by using APIs such as SoundCloud and Spotify.

● A complete list of all functionality (i.e., the actions) of your project, and an English description of each action. This is your product backlog. We strongly suggest that you organize these features into groups/modules based on related functionality. For each action (or group/module), clearly specify which team member(s) are primarily responsible for the success of that action (or group/module).

1.User account related actions (Yiran Zhou, Can Liu, Yi Meng):
SignIn: Registered users may log in using their username and password. Redirect users to the home page.
SignUp: New user may register for the site. New users must provide a username, email address and password. Registering for the site requires the user to provide a valid email address and will send a confirmation link to their email.
ConfirmEmail: This action will active the user and leaves the user logged in as the newly-registered user.
RestRequest: This action allows the user to reset their password by providing their username and email address. It checks whether the username and the email address exists in the database, if both of them exist, send a reset password link to the email provided by the user.
ConfirmReset: When user clicks on the link, redirect him to the reset password form.

RestPassword: This action creates a reset password form, which contains new password field and confirm password field. If the form is validate, change the user's password to the newly reset password and log the user in.

2.Game room related actions (responsibility assignment may be revised later):
CreateRoom(Yiran Zhou): This action allows the user to create a game room with 3 players (for now, may expand the number of players). The user who creates the room is the host of the game, and is the only one who is able to start the game. When room is created, the host stay at the room page to wait other users to join.
QuickStart (Yi Meng): This action allows the logged in user to quickly and randomly join a game room that is not full. If no room is available at the time, this action alerts the user that no room is found.
JoinGame (Yi Meng): Join allows the user to join a specific game room, (ideally the user can see whether the room is full or not) if the room is full, the user cannot get in. When the user joins a room, his status is set to "not ready".
LeaveGame(Can Liu): Leave redirects the user back to the homepage, a user cannot leave the game room once his status becomes "ready". The host on the other hand, can leave the game room any time he wants.
Ready/Unready(Can Liu): Ready sets the user's status from not ready to ready, the game can only starts with all players being ready. Unready just sets the user's status back to unready (kind of like follow and unfollow in grumblr).
AutoEject（Can Liu）: This action is for ejecting users who are not ready in regular time. The user who is ejected will be redirected to home page. The room model will delete this user.
ChangeHost(Yiran Zhou): If host of room leaves room, our game will change host of room automatically based on player's enter-room time.
DeleteRoom(Yiran Zhou): If user leave a room and room becomes empty, this room will be deleted.

3.Game workflow related actions (responsibility assignment may be revised later):
StartGame(Yiran Zhou): This action is for the game host only. This action cannot be evoked unless all players' status are "ready". This action redirects all players to the "playing page", creates a new form ready for the first player to draw.
SelectWord(Can Liu): At the beginning of each player's round, randomly select a word from the database and display it only to the player who is currently drawing. Also store this word into the current answer model in order to judge who makes a correct guess.
Guess(Can Liu): This action creates a form for each guessing player, it takes in the user's input (his guess), validates it and compares it to the answer. If the guess is correct, sends a congratulation message, else sends a wrong answer message.
JumpToNextDrawer(Yiran Zhou): after a user finish painting, our game will move to next drawer. If there is no next drawer and current round of game is not last round, our game will set it turns to first drawer and begin next round. Otherwise, game is end and redirects to end game function.
CountScores (Yi Meng): If the player makes a correct guess, adds point to his account, if the guess was wrong, nothing happens. For more complexity, we could add the points based on time, the player who makes the correct guess first gets 10 points, the second 5 points, etc.
EndGame/DisplayResult (Yi Meng): When the last player's round is finished, end the game and display a score result for all players. At this time, all players are able to be ready again or leave the game room, the host is able to start the game again.

4.JS in Canvas manipulate and count down. These are actions for painting itself (Can Liu, Yiran Zhou, Yi Meng):
Draw: This action allows the player to draw anything he wants, and display his drawing to all the players including himself in real-time using WebSocket.
Erase: Allow the user to erase his drawing at any time.
ChooseTools: This action allows users to choose drawing tools, if the user choose pencil, calls draw function, if the user choose eraser, calls erase function.
AutoEnd: This action is for stopping drawing if time is up. When the time is up, jump to the next player or new a round.

These are helper actions (Can Liu, Yiran Zhou, Yi Meng):
TimeCount: This action displays a time count down, we need it both in the game room and on the play page.
ConnectBack: This action is used to logged a user back when he got disconnected and tried to connect back to the game. Redirect the user back to the game, show him the current state.