USC Viterbi

INFORMATION TECHNOLOGY PROGRAM

ITP 303 Final Project Summary

INSTRUCTIONS: Type out your answer directly below each question. If a question does not apply to your project, type in N/A. When finished, save this file as a PDF and upload it to the itpwebdev server via FileZilla. Then, add a link to this PDF in your student_page.html. Label it "Final Project Summary."

BASICS

Your name?

John Meyering

What is the topic of your final project?

I made a knock-off of a game called Quiplash.

Who is the intended audience?

People who like Quiplash. (i.e. people who enjoy party games like Cards Against Humanity)

Provide a brief summary of the functionalities of your project.

-Anybody can "host" a game on their device.

-Anybody who makes an account can join games hosted by them or friends.

-Anybody who makes an account can customize their in-game display name, color, and font. (they can also delete their account if they so please)

-To get the best experience out of the hosting platform, I used an HTML canvas to display the game on the host's screen. Because I wanted to complete this assignment in less than a lifetime, I made the canvas size static. This shouldn't matter much because the original Quiplash does the same thing-> you can't properly host on a phone, you need a screen big enough for everybody to watch.

-Those who login and join a game can play from their phone, it's the expected way for a group of friends to play on the couch. (A smart TV can host the game, and everyone can play from their phones. There is no login or inputting required to host (aside from entering a room key), so a TV is fine for the job)

FRONT-END

What is the total page count?

4

List the names of any external stylesheet used in this project below.

style.css, join.css, login.css, profile.css, host.css

List any CSS libraries/frameworks used in this project (e.g. Bootstrap) below.

USC Viterbi

INFORMATION TECHNOLOGY PROGRAM

Bootstrap

List any JavaScript libraries/frameworks used in this project below.

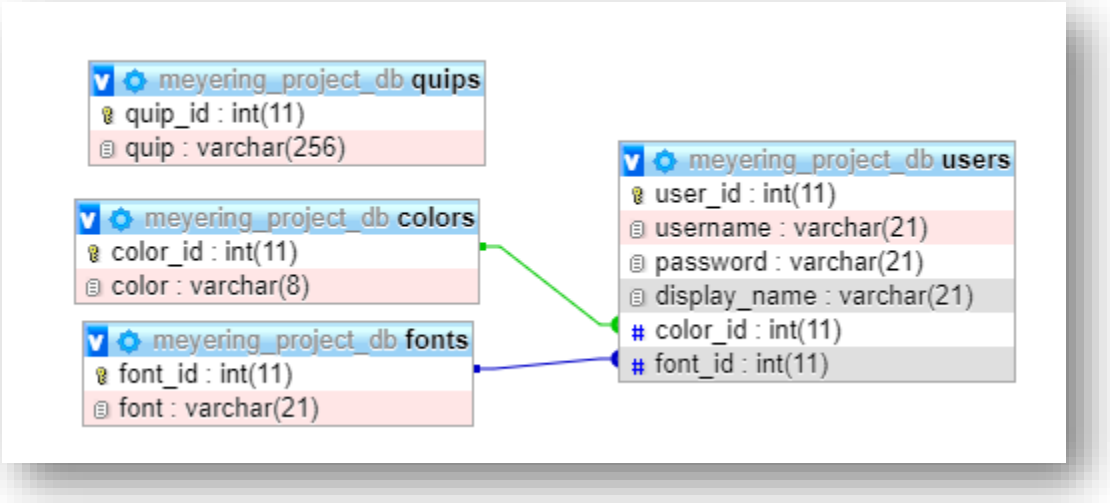
JQuery

How does your project meet the Interactivity requirement?

In join.php there are voting and "quipping" inputs that are generated based on WebSocket info from host.php.

BACK-END

Attach a screenshot of the final database diagram (just the diagram, do not include any records) below.



Where in your project do you insert a record to the database (the 'C' in CRUD)?

When a user is created (via src/register_service.php).

Where in your project do you search and display record(s) from the database (the 'R' in CRUD)?

When a user is trying to register, we check if their desired username is taken (via src/register_service.php).

When a user is trying to login, we check if their username + password combo is legitimate (via src/login_service.php).

When a user goes to profile.php, we download a list of fonts (via src/font_service.php).

When a user goes to host.php and starts a game, we download a list of "Quip Prompts" (via src/quip_service.php).

Where in your project do you update and existing record(s) the database (the 'U' in CRUD)?

When a user updates their display name, color, or font on profile.php (via src/display_name_service.php, src/display_color_service.php, and profile.php).

Where in your project do you delete existing record(s) from the database (the 'D' in CRUD)?

When a user decides to delete their account on profile.php (via src/delete_user_service.php).

USC Viterbi

INFORMATION TECHNOLOGY PROGRAM

MISC

What two "extra" requirements did you implement and where can they be found?

I implemented WebSockets in join.js and host.js.

I did Frontend <-> Backend AJAX for all of my record updating in profile.js.

There is a bunch more, this project was insane. (PHP WebSocket Server + hosting it on raspberry pi and getting a domain name, HTML canvas + JS game stuff)

If you used any APIs or JS plugins/frameworks as one of the extra requirements, list the name and a link to their documentation.

N/A

If your project requires any admin credentials (i.e. only admin users can access a certain page), list the credentials below.

N/A

Provide any other information that you think the grader/instructor should know when grading your final project below.

Sometimes my apartment loses power and I need to reset my Raspberry Pi and run the Websocket Server code manually (I don't have it set to do so on boot, I didn't have time to set it up). I will make sure that the Websocket Server is running when I get up everyday until I get a grade, but some days I don't get up until like 1PM. It should more or less always be on from 1PM until 3AM though.

The game has a strict 4 player requirement. It will wait until 4 players have connected. After that anybody who connects can vote, but can't compete. (Works just like regular Quiplash: just share the room key and a thousand+ people on Twitch.tv can join in on the voting fun)

Don't set your color to white, you won't see it on the canvas. I didn't have time to update the player card background dynamically.

If you want later as a voter, if your display name matches the display name of a player already in the game then you can play as that player. That's clearly a problem, but I didn't have time to implement player_ids in my WebSocket Server stuffs. The takeaway is that you should make sure that every player's display name is distinct and that the voters don't use the player's name either.

The voter's names can all be the same though, as long as they don't match a player's name.