

## Test Plan/Report

arBit | Twelve Articles | 12/1/19

### Manual Testing Approach / Description:

#### ➤ Black-box testing

Black-box testing treats the software as a "black box," examining functionality without any knowledge of internal implementation, without seeing the source code.

- Verified that given a certain input the correct output was produced by checking interaction between Firebase and a user's local phone. For example, if the options were shown in real time to every user when someone in the room inputted or deleted an option.
- Verified that no duplicate usernames were used in the same room.
- Verified that no duplicate room names exist
- Verified that on press of exit button, Firebase user count updated correctly and if last user in the room exited, that room got deleted.

### Testing Scenarios:

- Creating room
  - sanity checks for input field, not empty, or only spaces
  - Check for duplicates in database
- Join room
  - Sanity check for input field, not empty or only spaces
- Adding single or multiple options
  - Testing duplicated options or no options entered in text field to make sure alert pop up message displays correctly
  - Format option to capitalize first letter and rest lowercase to standardize option list
- Deleting options added by the specific user only
- Testing with single and multiple users in the room
- Testing waiting for all users to input their options before anyone can vote
- Tested the vote page to ensure every option is unique voted for (No blanks or duplicates)
- Exiting the room and checking that user and submission count correctly updates
- Testing having last person leave the room and making sure the whole room is deleted from firebase
- Option list layout and vote button check to make sure buttons don't block user from being able to vote for an item (css styling)
- All buttons perform as intended(back goes back, submit performs some action etc.)
- Ensuring correct yelp result displayed
  - Most relevant (top) option is linked to the option
  - Business photo or placeholder added to preview