Game Connect
Owen Fulton
Valliappan Chidambaram
Riley Kirkham
Mattias Leino
Rachel Rockenfeller

User Acceptance Testing
1. Account Creation:
    a. Select a username from the user database, and enter it into the username field.
    b. Select an email not in the user database, and enter it into the email field.
    c. Type any password in the password field, making sure that an error message comes up while the password is too short, and disappears afterwards.
    d. Type the password again, in the confirm password field, making sure that until you finish entering it an error message appears.
    e. Hit submit, and confirm that you get an error message about the checkbox.
    f. Check the box and submit again, confirming you get an error message about the user already being in the database.
    g. Repeat steps c-f with a new username and existing email (making sure it complains about the email being in the database), and a new username and email (making sure it works).
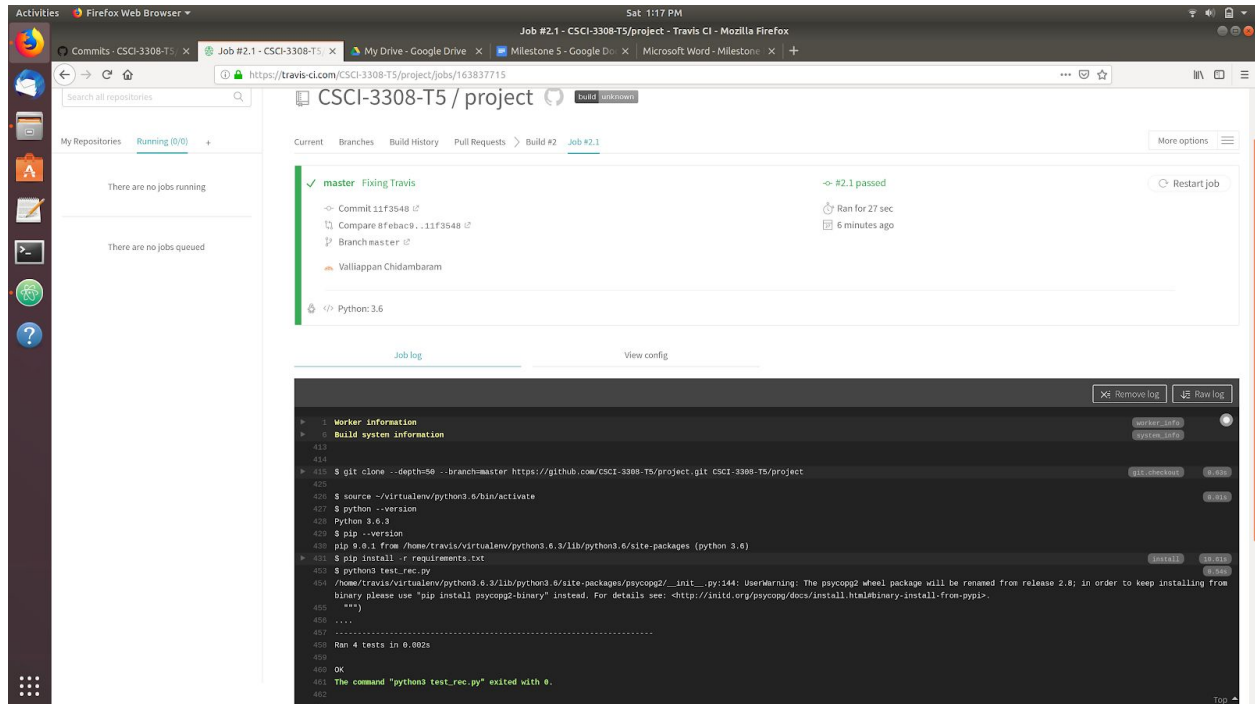2. Login:
    a. Either create an account or use an account which you know the credentials to.
    b. Enter a character in the username field and delete it, making sure an error message pops up.
    c. Enter a character in the password field and delete it, making sure an error message pops up.
    d. Use a username that isn't in the database and any password, making sure it gives you an error when submitted.
    e. Use the username from part a, and an incorrect password, making sure it gives you an error when submitted.
    f. Use the correct username and password from part a, making sure it lets you login, and that it takes you to your dashboard (which should contain all ratings the account had previously made).

g. Logout and log back in with the same credentials, but check the remember me box.

h. Close your browser, and open the site 2 hours later, making sure you get taken straight to your dashboard.

3. Game Rating:

a. Go to the dashboard of your account.

b. Enter the name of a game that isn't in the database, and click one of the stars. No stars should light up and an error message should appear. Verify that the link takes you to the page where you can add a game.

c. Return to the dashboard and enter the name of a game from the database you haven't rated. Click one of the stars. No stars next to the input field should be lit up, and a new row with the appropriate number of stars lit up should appear at the top.

d. Rate more games until you have at least 5 rated.

e. Refresh the page and make sure that games are sorted alphabetically and have retained the correct values.

f. Change the ratings for each a few times (do NOT click on the star corresponding to the current rating). Refresh the page and ensure that the ratings correspond to the final ratings from before.

g. Click on the last start that is lit up for a few of the ratings. None of the stars for those games should stay yellow. Refresh the page and ensure that those games have been removed.

h. Visit the page for one of the games, and ensure that the stars behave as they do on the dashboard (steps e, f, and g).

Automated Tests:

We used TravisCI to automate our testing. We wrote tests for our python scripts using unittest, and we used TravisCI to make sure that they passed whenever we made a commit. This is the build status on the TravisCI website. The test cases can also be run manually with the command "python3 test_rec.py", provided the correct dependencies are installed.