|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Description | User Action | System Response | Current Functionality | Additional Comments |
| A user can create an account | Generate unique information and create an account | Account creation and access to dashboard(s) | Functional | Works as expected and does not crash |
| A user can find a game to play | Access search functionality and have visual cues | Display and return results with keywords and images | Functional | Random game shuffle and |
| A user can link their Steam Information | Insert their respective API and ID in the provided fields | Generate a new profile in the system and give suggestions | Functional | Allows valid steam user to link information |
| A user can find a link to the Steam marketplace to buy a game | No more than a few clicks from search results. A provided link. | Link to the Steam marketplace with the game embedded | In progress | We need to add a more visible link for user to access |
| A user can find friends on the platform | A search input bar that finds other users | Return a list of corresponding users with a keyword or character | Functional | Easily accessible through friends list button |
| For security purposes  Verify there is a limit  On unsuccessful login attempts so that  The user cannot use  Brute force for login | User is allowed to enter credentials maybe only 4-5 times | Blocks user from being able to try logging in for a set period if they exceed this number | In progress | For security purposes. This is an area that needs a bit of work |
| Display message saying incorrect login OR password. This way a hacker will not know which one is incorrect | If the user enters either their username or password incorrectly it does not tell the user which field as it will make it easier for a hacker | Displays username OR password as invalid | Functional | Message is not displayed telling a malicious hacker which piece of information was incorrect |
| Once logged in, verifying going between different pages does not log the user out | After logging in the user should be able to move between different pages on one login | Keeps user logged in | Functional | User remains logged in at all pages |
| Make sure we are using the correct data structures within each function so that we are making sure we are getting the best speed and optimization | A technical requirement | System should run in a quick and optimized manner | In progress | While the app does not seem terribly slow, certain areas such as where there are 2 For loops could be handled with a better data structure |
| Check the functionality of both the resets and login buttons on the homepage | User should be able to click on either of these or they should work as intended | System should respond to these buttons | In progress | These buttons need to be added |
| Make sure all the CSS text boxes, web page cells, etc. are properly aligned | Technical requirement | Should display info in a clear to read manner | Functional/ In progress | Looks good for now, but for demo 4 we will focus on cleaning it up a bit |
| Verify we have no broken links or images | When user clicks on links or phots, they should all lead to a valid source and photos that can be viewed on the site | All links and photos should be valid | Functional | Nothing seems to crash on invalid input |
| Verify all the endpoints in the express API database. Make sure all information can be sent and received properly. | When user is inputting their data, it is essential it is safely and securely being saved | User data should be able to be read and written to the database | Functional | App does not seem to crash here from any post, get, delete, etc. |
| Make sure unauthorized users do not have access to database data and queries | Make sure each user only has access to their data and that the database cannot easily be hacked | System should not let unauthorized users in database | Functional/ In progress | All unit testing for these types of events is passing for now, but we will continue to check this in depth the next week |
| Verify only valid accounts can sign on. Make sure there is no way to bypass this maliciously | Technical requirement | System should have security checks in place ensuring login functionality cannot be bypassed | Functional | There is not easy way to bypass this security that we can find for now |
| Verify that only lower- and upper-case letters as well as list of allowed special characters are allowed for login. | When user logins and function checks this makes sure that only valid characters are allowed | Displays error message | In progress | We need to make a list of only allowed characters |
| Only allow registered and valid steam accounts to be used for search | When user enters steam info, verifies it as a valid account | Displays error message and tells user to enter new account info | Functional | Only registered users are allowed to login and use search |
| Verify user login information and details are sent and received to the database in a safe matter and not easily intercepted | When user enters info, it should be sent securely | Sends and receives data in a secure and encrypted manner | In progress | We need to double check everything from the backend is being sent in a secure fashion |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  | \ |
|  |  |  |  |  |
|  |  |  |  |  |

Testing Document Group 4

These are the current but not complete number of user use cases and application functionality of our application. The application continues to be in development and the “X” portion of it is being worked on. Our progress includes: xxxx and we are targeting a completion date of “X” for this segment. Other areas of interest include X and X. These are critical parts of the application as it allows users to differentiate us from other game launchers and why they should come to our application.

Current Developments:

Horizon goals:

Internal Questions:

Overall Thoughts: