Team Wellington

JunkHub Web App

Team Project

Report on the testing performed on the JunkHub developed site for the final project of IT6036 Security and Server-Side Development.

Testing report

IT6036 – Final Project

# Feature testing

**Introduction:**

This document contains the selection and result of testing our Django website for the purpose of completing IT6036 project assessment.

**Choose test item:**

The test item is the Django web app final project uploaded to the GitHub repository here:

<https://github.com/IT6036-App-Security/secure-web-app-project-2022-q2-team-wellington>

**Features to be tested:**

The sign up function works as intended. Passwords are only accepted if they meet the requirements in the Django Security settings (8 characters long, no common phrases etc)

The sign in function works as intended. Incorrect sign in information should not be accepted. The prerequisite is an existing account.

The profile edit page works as intended. Users can view and edit their account information.

The home page displays correctly. The home page should be populated with a list of items that are for sale, with the links to the relevant item details.

Item listing creation works as intended. A signed in user can create a listing that describes an item they wish to sell and a price. They can post that to the website.

Items can be edited and deleted – A signed in user can delete or edit their own listing *but not anyone else’s.*

Users can leave comments on items (to communicate with the seller) User must log in to do this.

**Features NOT tested:**

We are not testing performance or responsiveness of the website.

We are not testing for quality of content, e.g., appropriate resolution of pictures.

We are not testing any visual or UI design components.

This test is focused on the functionality of being able to log in to the website, view the listings and comment on them, as per the user stories of the first three iterations of development.

**Testing Approach:**

Testing is approached as developers – we implement coded features and then run the server on a local machine to test for acceptable results. Incorrect implementation of the features results in revision of the code or approach to the problem.

Additionally, we will create user tests on the models to ensure the database is acting as required and run these tests with the [python manage.py test] command.

**Pass/Fail Criteria:**

The user profile features will pass if they allow the user to sign up, sign in and log out without causing any errors, and while only accepting correct user data.

The item listing feature will work if it displays a list of all created listings in the local database.

The comment feature will work if it only allows comments from logged in users and displays those comments along with the commenter’s username in the items detail view.

Inbuilt user tests will pass if they return from the test command without any failures or errors.

# Testing

**Test Deliverables**

The tests will be formatted in a table, as below. Each row of the table describes the test, expected action, pass or fail results

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Objective** | **Precondition** | **Input** | **Result** | **Expected** | **Status** |
| 1 | Test signing up user with good password | N/A | Username, acceptable password | Successful sign up | Successful sign up | PASS |
|  | Test signing up user with common password “password” | N/A | Username, unacceptable password | Sign up fails, password too common | Sign up failed. | PASS |
|  | Test logging in with valid user | User has already signed up | **Correct** username, password | Successful log in | Successful log in | PASS |
|  | Test logging in with invalid user | User has already signed up | **Wrong** email, password | Failed log in | Failure to log in | PASS |
|  | Test logging in with invalid password | Website access and existing account. | Email, **wrong** password | Failed log in | Failure to log in | PASS |
|  | Test home page displays correctly | URL is on home page | N/A | All items listed | All items listed | PASS |
|  | Test item view displays correctly | Some items have been listed already | Selecting the link of any of the listed items | The correct item is displayed | Correct item details display | Pass |
|  | Test user can create a listing | User is signed in | Details for the listing | Creates a new listing, adds to home page | New listings successfully created | PASS |
|  | Test user can comment on listing | User is signed in | Comment body | Comment is added to the listing item view | Comment is added to the listing item view | PASS |
|  | Test user cannot comment without signing in | User is in a items details view | N/A | No form, message telling user to log in | No form, message telling user to log in | PASS |
|  | User can sign out | User is signed in | N/A | Log out nav button changes, redirects to home page | Log out nav button changes, redirects to home page | PASS |
| 10 | User can only delete own listing | User is signed in | N/A | No button to delete others listings | User can only delete own listing | PASS |

**Unit Testing Examples:**

Text

Description automatically generatedText

Description automatically generatedText

Description automatically generated