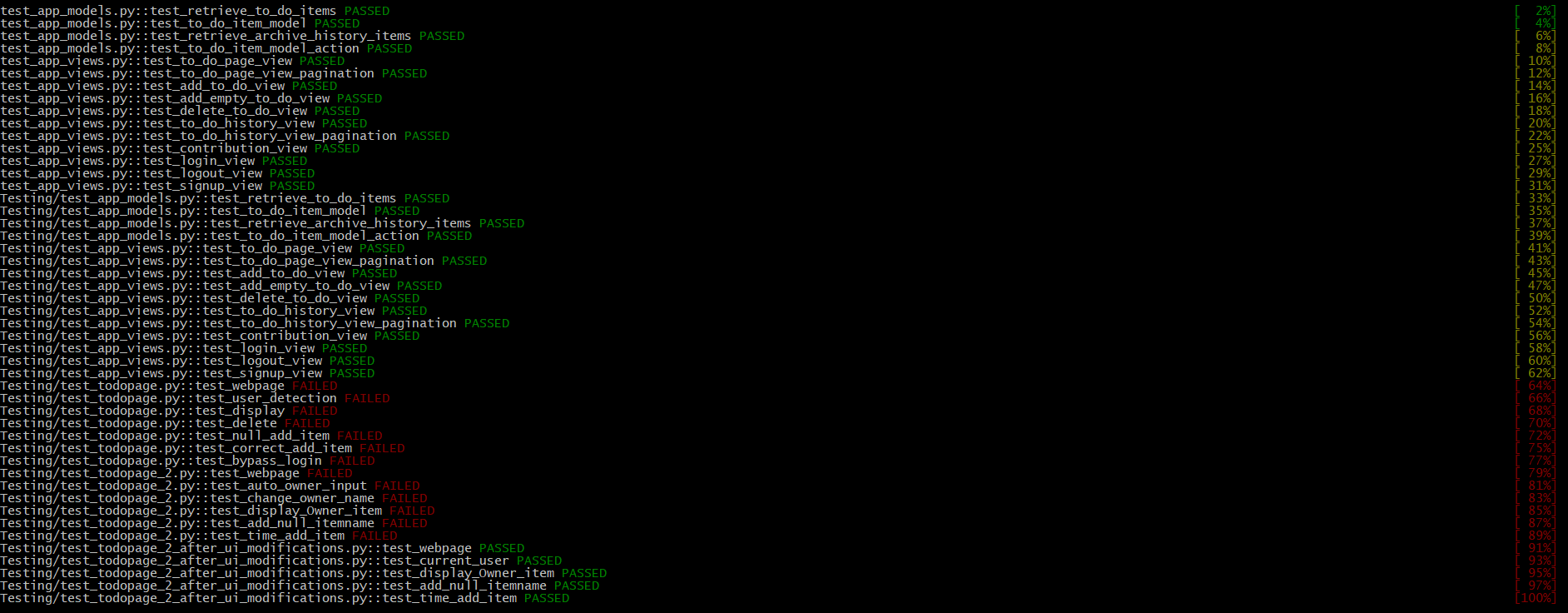
# **Justifications of Results**

## Introduction

This justification applies to users who clone the completed **ETI-To-Do-App** project from the repository and run the pytest -v --cov command on the root folder of the project as some of the tests results seen would be different from the screenshots captured during development and testing of this web application.

## Developer Unit Tests

Under the **To\_do\_page** folder, all developer unit tests inside *test\_todopage.py* and *test\_todopage\_2.py* would fail if a user were to clone this project and run the pytest command inside this folder as seen in **Figure 1**.



**Figure 1**

One of the reasons identified for the failure is due to inclusion of the authentication views of the Django Authentication system for a change of requirements of the User Login Page in Sprint 2 to allow new users to sign up. As a result, the URL path for the login page was changed from <http://127.0.0.1:8000/login/login> to <http://127.0.0.1:8000/accounts/login>.

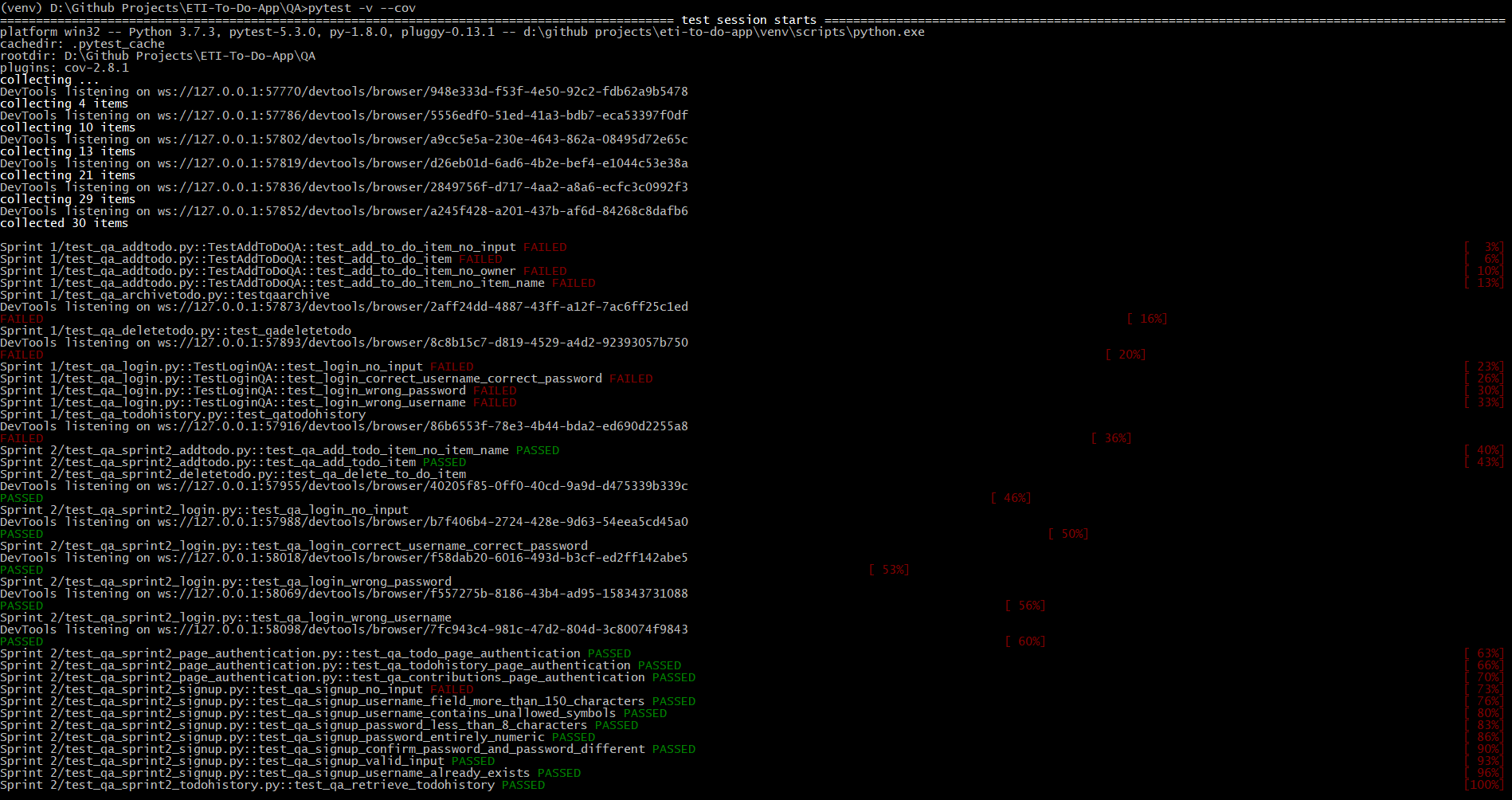
Since the developer unit tests both *test\_todopage.py* and *test\_todopage\_2.py* were completed and ran before the developer assigned to the issue for change of requirements for the User Login Page had finished working on the changes, the User Login Page URL used in these two files were not changed to the new URL.

With the other reason being the redesign of the To-Do Page and the To-Do History Page to improve the User Experience of the pages, the main HTML elements that used to display each To-Do Item and To-Do Item’s History was changed from an unordered list to a table with rows. This has resulted in the selenium web driver in the two developer unit test files not being able to find the elements which displayed data for the To-Do Item and To-Do Item’s History records.

After these turns of events, it was decided that there was a need of a new unit test file *test\_todopage\_2\_after\_ui\_modifications.py* to be created to consolidate all the changes made so far as well as retain the original unit tests done.

## QA Unit Tests

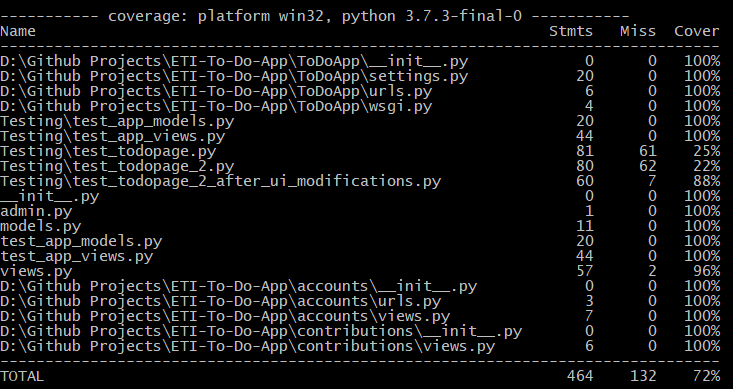
This is the tests results you would see if you clone the completed **ETI-To-Do-App** project from the repository and run the pytest command with the coverage option --cov on the **QA** folder of the project shown in Figure 2 below.



**Figure 2**

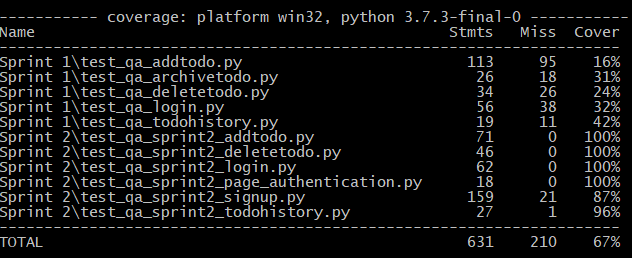
## Coverage

This is the coverage results you would see if you clone the completed **ETI-To-Do-App** project from the repository and run the pytest command with the coverage option --cov on the **To\_do\_page** of the project shown in Figure 3 below.



**Figure 3**

This is the coverage results you would see if you clone the completed **ETI-To-Do-App** project from the repository and run the pytest command with the coverage option --cov on the **QA** folder of the project shown in Figure 4 below.



**Figure 4**