**CS673 Software Engineering** 

**Team 1 - Knowitall**

**Tests Report**

| Team Member | Role(s) | Signature | Date |
| --- | --- | --- | --- |
| Daniel Makover | Team Leader | *Daniel Makover* | 11/11/21 |
| Kun Mo | Team Leader | *Kun Mo* | 11/11/21 |
| Gunnar Nichols | QA Leader | *Gunnar Nichols* | 11/11/21 |
| Heli Kolambekar | Requirements Leader | *Heli Kolambekar* | 11/11/21 |
| Haoyi Zhu | Design and Implementation Leader | *Haoyi Zhu* | 11/11/21 |
| Weiye Xu | Configuration Leader | *Weiye Xu* | 11/11/21 |
| Yuwei Wu | Security Leader | *Yuwei Wu* | 11/11/21 |

**Revision history**

| **Version** | **Author** | **Date** | **Change** |
| --- | --- | --- | --- |
| **1.0** | **Gunnar Nichols** | **11/9/21** | **Drafted** |
|  |  |  |  |

[Introduction](#_87t9hln2vjz0)

[Test Summary](#_sm5odwyvuk3j)

[Tests Reports](#_pqso2mbjyzx4)

[Testing Metrics](#_mtfbusfb0eq3)

[References](#_15tmymhipvdv)

[Glossary](#_8n34lvocupub)

# Introduction

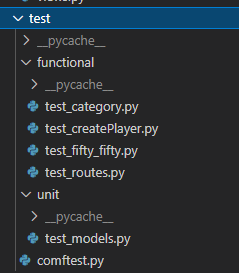
The automated tests for knowitall are based on the pytest-flask plugin, which is an expansion of the capabilities of pytest to be better suited for testing a flask application. The current scope of the testing is preliminary and constitutes making sure each object has the structure it requires, and that each route loads initially.

# Test Summary

Our testing scripts can be found in the tests folder located in the root directory of our application. When the command ‘$python -m pytest’ is run from the root directory, all of the test functions (beginning with test\_\*) in any of the test files (beginning with test\_\*) will be executed and the summary of the results are provided in the terminal.

The tests are grouped by type within the folder describing the testing activities. Currently functional tests can be found in the ‘functional’ folder, while unit tests can be found in the ‘unit’ folder.

Test Directory:



# Tests Reports - Current Tests

The full test case table can be found here: [Test Cases](https://docs.google.com/spreadsheets/d/15znXkVjUvw8YXIBSycquCkoALb_QzPvO2Z4BOEkRmY4/edit#gid=0)

Here are the first few tests/columns

| Test case name | Models (4) | FiftyFifty | Home Route |
| --- | --- | --- | --- |
| New or old: | New | New | New |
| Test items: | Game, Player, LeaderboardScore and Question Models | FiftyFifty Lineline Route | Home Page |
| Test priority: | medium | Low | Medium |
| Dependencies: | N/A | N/A | N/A |
| Preconditions: | N/A | N/A | N/A |
| input data: | Object Model Values | fiftyfifty route | home route |
| Test steps: | Assign Values to Object, make sure they hold | Create App  Navigate to 50/50 url  Make sure response has expected content | Create App  Navigate to home url  Make sure response has expected content |
| Postconditions: | N/A | N/A | N/A |
| Expected output: | All asserts true | All asserts true | All asserts true |
| Actual output: | All asserts true | All asserts true | All asserts true |
| Pass or Fail: | Pass | Pass | Pass |
| Bug id/link: | N/A | N/A | N/A |
| Additional notes: | Each model has a test | Each game function should have a test | Each Route should have a test |

# Testing Metrics

*Metric: What it is. How it is tracked. How it is analyzed and used improve or understand the project*

* + - 1. *Tests Written:* Total number of tests written. This is a measurement of how many tests have been written
      2. *Tests Passed:* Test Passed / Tests Written. This is a measurement of how well the current code passes the written tests.

Tests Written: 12

Tests Passed: 100%

This will be expanded during iteration3 while we are striving to actually follow the TDD process.

# References

Pytest Documentation: https://docs.pytest.org/en/6.2.x/contents.html

Pytest-flask Documentation: <https://pytest-flask.readthedocs.io/en/latest/>

Pytest-flask tutorial: https://testdriven.io/blog/flask-pytest/

# Glossary