Test Results Document

for

Pandemic Tracker

**Version 1.1 approved**

**Prepared by Máté Hekfusz**

**New York University Abu Dhabi**

**13 May 2020**

1. **Purpose**

The Test Results Document (TRD) summarises the major bugs and defects discovered during the testing process of *PandemicTracker*. It will not cover bugs found and fixed during construction. For the included bugs, the document will state the module that had the defect, the test case that discovered them (for test cases refer to the Test Plan or the individual test case documents), a description for the bug, their root cause, and their resolution.

Overall, the team encountered five major bugs in all three (Webscraping, Database, and Display) components, all thanks to the extensive testing process. All of them were fixed.

1. **Test Results**

|  |  |
| --- | --- |
| Bug 1 | |
| Module | Retrieve data, Scrape data, Request HTTP |
| Test Case ID | TEST-WS-001-2 |
| Test Case Name | General Webscraping Validation Test – Alt Flow |
| Description | The scrapers crashed when the internet connection was cut midway through the process. |
| Root Cause | The bug was caused by the Requests module which establishes connections to the scraped resources. Naturally, it requires connectivity. |
| Resolution | An exception catcher was added to the page requests which gracefully terminates the particular scraper without internet. An additional check was added to ensure the pages were loaded appropriately. |

|  |  |
| --- | --- |
| Bug 2 | |
| Module | Retrieve data, Scrape data, Request HTTP |
| Test Case ID | TEST-WS-001-2 |
| Test Case Name | General Webscraping Validation Test – Alt Flow |
| Description | The scrapers crashed when they tried to process a site that they were not configured for. Though this should never happen in production (as the resources are coded into the scrapers), this was nonetheless a failed test case. |
| Root Cause | Each scraper had to be manually crafted for the page it scraped, and thus anything different caused crashes, generally because of indexing. |
| Resolution | Exception catchers and sanity checks were added to each scraper that detect when the page is not the correct one through heuristics and terminate gracefully if it is not. |

|  |  |
| --- | --- |
| Bug 3 | |
| Module | Retrieve data, Scrape data, Request HTTP |
| Test Case ID | TEST-WS-100-2 |
| Test Case Name | StatsScraper Validation Test – Alt Flow |
| Description | StatsScraper crashed when it tried to scrape before Johns Hopkins uploaded the case data for the day before: e.g. if it was trying to scrape for 13 May when the most recent data was for 12 May. |
| Root Cause | StatsScraper detects the time of scraping and dynamically seeks the most recent data to scrape. This could result in an index error if the case data was not updated yet for the day. |
| Resolution | A two-step sanity check was implemented to resolve this issue: if there is no recent data to grab, StatsScraper will try to scrape the data from the day before that. If that fails also, it gracefully terminates. |

|  |  |
| --- | --- |
| Bug 4 | |
| Module | Update database |
| Test Case ID | TEST-DB-001-1 |
| Test Case Name | Query Correctness |
| Description | Upon trying to enter scraped data into the database with a query, an error message saying that ‘mySQL cannot accept arguments %d’ showed and the query failed. |
| Root Cause | MySQL, at least through PyMySQL needs string (%s) arguments even if the database takes integers or different datatypes. |
| Resolution | The query’s arguments were changed to %s. This resulted in no change in data correctness as the database was set up with the correct datatypes beforehand. |

|  |  |
| --- | --- |
| Bug 5 | |
| Module | Display comparison for countries |
| Test Case ID | TEST-CC-001-1 |
| Test Case Name | Compare Countries Template Test |
| Description | When another country was chosen in the list, the graphs did not change to reflect the country change. Other things changed on the page, but the graph did not. |
| Root Cause | The JavaScript code was not effectively connecting the graph element with the HTML select element, the change in graphs was not triggered by changing the select element. |
| Resolution | This issue was solved by writing a script that would parse the countries’ Jinja data into a JSON file that could be searched and match the graph name to the select element name. Then, the graph would be “re-made” and rendered every time the select element changes. |