**Contents**

1. Background information
2. Test Results for each feature
   1. Average Time to fix or close Issues
   2. Developers who resolved issues
   3. Number of issues raised per file
   4. Number of comments per issue
   5. Average lines of code per commit
   6. Barchart - Number of lines contributed by each user
   7. Pie chart - Number of commits made by each user
   8. Timely Manner - Average time between commits
   9. Timeline - Time of commits in a repository
   10. Filelist - List of files showing the team member that is responsible
   11. Most frequently changed file
   12. Most frequently changed file by each developer

**Background Information**

|  |  |
| --- | --- |
| **Application Name** | GitHub Inspector |
| **Date of Creation** | 18/09/2019 |
| **Created by Name** | Ayesha Ali - 29587778  Ryley Angus - 22647597  Shreya Kathuriya - 29506751  Akshith Patil - 29717973 |
| **Environment** | Windows 10, Android Phone |

**Average Time to fix or close Issues**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test ID** | **Test Scenario** | **Test cases** | **Test data** | **Pre-conditions** | **Post-conditions** | **Expected Results** | **Actual Results** | **Status (Pass/ Fail)** |
| TC\_1 | Testing the output | Enter a valid URL of a public GitHub repository | Enter the following URL in the Timeline page  [https://github.com/flyrightsister/udemy-react-testing-projects](https://github.com/flyrightsister/udemy-react-testing-projects.git) | URL of the Github repository should be of a public repo | Average issue resolution time should be displayed | Average time is 19-20 hours with a single issue | *As expected* | Pass |
| TC\_2 | Testing the output | Enter a valid URL of a public GitHub repository | Enter the following URL in the Timeline page  <https://github.com/Pragmatists/testing-examples> | URL of the Github repository should be of a public repo | Average issue resolution time should be displayed | “No issues exist for the provided repo” should be displayed to the user | *As expected* | Pass |
| TC\_3 | Testing the output | Enter a valid URL of a public GitHub repository | Enter the following URL in the Timeline page  [https://github.com/googlesamples/assistant-sdk-python](https://github.com/googlesamples/assistant-sdk-python.git) | URL of the Github repository should be of a public repo | Average issue resolution time should be displayed | Average time should be roughly 15 weeks and 15 hours | *As expected* | Pass |

**Developers who resolved issues**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test ID** | **Test Scenario** | **Test cases** | **Test data** | **Pre-conditions** | **Post-conditions** | **Expected Results** | **Actual Results** | **Status (Pass/ Fail)** |
| TC\_1 | Testing the output | Enter a valid URL of a public GitHub repository | Enter the following URL in the Timeline page  [https://github.com/flyrightsister/udemy-react-testing-projects](https://github.com/flyrightsister/udemy-react-testing-projects.git) | URL of the Github repository should be of a public repo | Developer names and the number of issues they resolved should be displayed | Exactly one issue should be shown as resolved, by a single developer | *As expected* | Pass |
| TC\_2 | Testing the output | Enter a valid URL of a public GitHub repository | Enter the following URL in the Timeline page  <https://github.com/Pragmatists/testing-examples> | URL of the Github repository should be of a public repo | Developer names and the number of issues they resolved should be displayed | “No available issue data” should be displayed to user | *As expected* | Pass |
| TC\_3 | Testing the output | Enter a valid URL of a public GitHub repository | Enter the following URL in the Timeline page  [https://github.com/googlesamples/assistant-sdk-python](https://github.com/googlesamples/assistant-sdk-python.git) | URL of the Github repository should be of a public repo | Developer names and the number of issues they resolved should be displayed | 15-25 columns should be visible, with the user ‘Fleker’ registering the most resolved issues | *As expected* | Pass |

**Number of comments per issue**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test ID** | **Test Scenario** | **Test cases** | **Test data** | **Pre-conditions** | **Post-conditions** | **Expected Results** | **Actual Results** | **Status (Pass/ Fail)** |
| TC\_1 | Testing the output | Enter a valid URL of a public GitHub repository | Enter the following URL in the Timeline page  [https://github.com/flyrightsister/udemy-react-testing-projects](https://github.com/flyrightsister/udemy-react-testing-projects.git) | URL of the Github repository should be of a public repo | A chart showing showing the number of comments per issue should be displayed | Issues 6 and 7 should each have one comment. No other issue should have a comment. | *As expected* | Pass |
| TC\_2 | Testing the output | Enter a valid URL of a public GitHub repository | Enter the following URL in the Timeline page  <https://github.com/Pragmatists/testing-examples> | URL of the Github repository should be of a public repo | A chart showing showing the number of comments per issue should be displayed | A column of zero height should be shown for a single issue/pull request | *As expected* | Pass |
| TC\_3 | Testing the output | Enter a valid URL of a public GitHub repository | Enter the following URL in the Timeline page  [https://github.com/googlesamples/assistant-sdk-python](https://github.com/googlesamples/assistant-sdk-python.git) | URL of the Github repository should be of a public repo | A chart showing showing the number of comments per issue should be displayed | Several hundred issues should be shown, with a peak of 57 comments for issue #236. | *As expected* | Pass |

**Average lines of code per commit**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test ID** | **Test Scenario** | **Test cases** | **Test data** | **Pre-conditions** | **Post-conditions** | **Expected Results** | **Actual Results** | **Status (Pass/ Fail)** |
| TC\_1 | Testing the output | Enter a valid URL of a public GitHub repository | Enter the following URL in the Timeline page  [https://github.com/flyrightsister/udemy-react-testing-projects](https://github.com/flyrightsister/udemy-react-testing-projects.git) | URL of the Github repository should be of a public repo | A chart showing showing the average amount of additions and deletions per developer, per commit should be displayed | Three columns, where one is much larger than the others | *As expected* | Pass |
| TC\_2 | Testing the output | Enter a valid URL of a public GitHub repository | Enter the following URL in the Timeline page  <https://github.com/Pragmatists/testing-examples> | URL of the Github repository should be of a public repo | A chart showing showing the average amount of additions and deletions per developer, per commit should be displayed | Two differently sized columns, where one is more negative and the other more positive | *As expected* | Pass |
| TC\_3 | Testing the output | Enter a valid URL of a public GitHub repository | Enter the following URL in the Timeline page  [https://github.com/googlesamples/assistant-sdk-python](https://github.com/googlesamples/assistant-sdk-python.git) | URL of the Github repository should be of a public repo | A chart showing showing the average amount of additions and deletions per developer, per commit should be displayed | 14 columns for 14 different developers. Two should be substantially larger than the others. | *As expected* | Pass |

**Pie chart - Number of lines contributed by each user**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test ID** | **Test Scenario** | **Test cases** | **Test data** | **Pre-conditions** | **Post-conditions** | **Expected Results** | **Actual Results** | **Status (Pass/ Fail)** |
| TC\_1 | Testing the output | Enter a valid URL of a public GitHub repository | Enter the following URL in the Timeline page  [https://github.com/flyrightsister/udemy-react-testing-projects](https://github.com/flyrightsister/udemy-react-testing-projects.git) | URL of the Github repository should be of a public repo | Pie chart showing the number of lines coded per person should be shown | Pie chart showing each author, and their respective number of lines of code is generated | *Pie chart shows equal distribution of code instead of showing actual distribution* | Fail |
| TC\_2 | Testing the output | Enter a valid URL of a public GitHub repository | Enter the following URL in the Timeline page  <https://github.com/Pragmatists/testing-examples> | URL of the Github repository should be of a public repo | Pie chart showing the number of lines coded per person should be shown | Pie chart showing each author, and their respective number of lines of code is generated | *Pie chart shows equal distribution of code instead of showing actual distribution* | Fail |
| TC\_3 | Testing the output | Enter a valid URL of a public GitHub repository | Enter the following URL in the Timeline page  [https://github.com/googlesamples/assistant-sdk-python](https://github.com/googlesamples/assistant-sdk-python.git) | URL of the Github repository should be of a public repo | Pie chart showing the number of lines coded per person should be shown | Pie chart showing each author, and their respective number of lines of code is generated | *Pie chart shows equal distribution of code instead of showing actual distribution* | Fail |
| TC\_4 | Testing the output | Enter a valid URL of a public GitHub repository | Enter the following URL in the Timeline page  <https://github.com/Pragmatists/testing-examples> | URL of the Github repository should be of a public repo | Pie chart showing the number of lines coded per person should be shown | Pie chart showing each author, and their respective number of lines of code is generated | *Pie chart shows equal distribution of code instead of showing actual distribution* | Fail |

**Pie chart - Number of commits made by each user**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test ID** | **Test Scenario** | **Test cases** | **Test data** | **Pre-conditions** | **Post-conditions** | **Expected Results** | **Actual Results** | **Status (Pass/ Fail)** |
| TC\_1 | Testing the output | Enter a valid URL of a public GitHub repository | Enter the following URL in the Timeline page  [https://github.com/flyrightsister/udemy-react-testing-projects](https://github.com/flyrightsister/udemy-react-testing-projects.git) | URL of the Github repository should be of a public repo | Bar chart showing the number of commits per person should be shown | Bar chart showing each author, and their respective number of commits is generated | Respective bar chart is generated | Pass |
| TC\_2 | Testing the output | Enter a valid URL of a public GitHub repository | Enter the following URL in the Timeline page  <https://github.com/Pragmatists/testing-examples> | URL of the Github repository should be of a public repo | Bar chart showing the number of commits per person should be shown | Bar chart showing each author, and their respective number of commits is generated | Respective bar chart is generated | Pass |
| TC\_3 | Testing the output | Enter a valid URL of a public GitHub repository | Enter the following URL in the Timeline page  [https://github.com/googlesamples/assistant-sdk-python](https://github.com/googlesamples/assistant-sdk-python.git) | URL of the Github repository should be of a public repo | Bar chart showing the number of commits per person should be shown | Bar chart showing each author, and their respective number of commits is generated | Respective bar chart is generated | Pass |
| TC\_4 | Testing the output | Enter a valid URL of a public GitHub repository | Enter the following URL in the Timeline page  <https://github.com/Pragmatists/testing-examples> | URL of the Github repository should be of a public repo | Bar chart showing the number of commits per person should be shown | Bar chart showing each author, and their respective number of commits is generated | Respective bar chart is generated | Pass |

**Timeline Test Results**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test ID** | **Test Scenario** | **Test cases** | **Test data** | **Pre-conditions** | **Post-conditions** | **Expected Results** | **Actual Results** | **Status (Pass/ Fail)** |
| TC\_1 | Testing the output (Timeline) | Enter a valid URL of a public GitHub repository | Enter the following URL in the Timeline page  [https://github.com/flyrightsister/udemy-react-testing-projects](https://github.com/flyrightsister/udemy-react-testing-projects.git) | URL of the Github repository should be of a public repo | Timeline should be displayed | Horizontal timeline is presented with 54 commits that shows date, time and name of the committer. | Timeline with 54 commits is displayed | Pass |
| TC\_2 | Testing the output (Timeline) | Enter a valid URL of a public GitHub repository | Enter the following URL in the Timeline page  <https://github.com/Pragmatists/testing-examples> | URL of the Github repository should be of a public repo | Timeline should be displayed | Horizontal timeline is presented with 61 commits that shows date, time and name of the committer. | Timeline with 61 commits is displayed | Pass |
| TC\_3 | Testing the output (Timeline) | Enter a valid URL of a public GitHub repository | Enter the following URL in the Timeline page  [https://github.com/googlesamples/assistant-sdk-python](https://github.com/googlesamples/assistant-sdk-python.git) | URL of the Github repository should be of a public repo | Timeline should be displayed | Horizontal timeline is presented with 331 commits that shows date, time and name of the committer. | Timeline with 331 commits is displayed | Pass |
| TC\_4 | Testing the output (Timeline) | Enter a valid URL of a public GitHub repository | Enter the following URL in the Timeline page  <https://github.com/Pragmatists/testing-examples> | URL of the Github repository should be of a public repo | Timeline should be displayed | Horizontal timeline is presented with 512 commits that shows date, time and name of the committer. | Timeline with 512 commits is displayed | Pass |

**Timely manner- Average time between commits**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test ID** | **Test Scenario** | **Test cases** | **Test data** | **Pre-conditions** | **Post-conditions** | **Expected Results** | **Actual Results** | **Status (Pass/ Fail)** |
| TC\_1 | Testing the output (List of average time between commits for each user) | Enter a valid URL of a public GitHub repository | Enter the following URL in the Timeline page  [https://github.com/flyrightsister/udemy-react-testing-projects](https://github.com/flyrightsister/udemy-react-testing-projects.git) | URL of the Github repository should be of a public repo | Average time between commits for should be displayed | List of average time between commits for 3 contributors in project should be presented | Average time between commits for 3 contributors is illustrated | Pass |
| TC\_2 | Testing the output (List of average time between commits for each user) | Enter a valid URL of a public GitHub repository | Enter the following URL in the Timeline page  <https://github.com/Pragmatists/testing-examples> | URL of the Github repository should be of a public repo | Average time between commits for should be displayed | List of average time between commits for 2 contributors in project should be presented | Average time between commits for 2 contributors is illustrated | Pass |
| TC\_3 | Testing the output (List of average time between commits for each user) | Enter a valid URL of a public GitHub repository | Enter the following URL in the Timeline page  [https://github.com/googlesamples/assistant-sdk-python](https://github.com/googlesamples/assistant-sdk-python.git) | URL of the Github repository should be of a public repo | Average time between commits for should be displayed | List of average time between commits for 15 contributors in project should be presented | Average time between commits for 15 contributors is illustrated | Pass |
| TC\_4 | Testing the output (List of average time between commits for each user) | Enter a valid URL of a public GitHub repository | Enter the following URL in the Timeline page  <https://github.com/Parchive/par2cmdline> | URL of the Github repository should be of a public repo | Average time between commits for should be displayed | List of average time between commits for 21 contributors in project should be presented | Average time between commits for 21 contributors is illustrated | Pass |

**Filelist**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test ID** | **Test Scenario** | **Test cases** | **Test data** | **Pre-conditions** | **Post-conditions** | **Expected Results** | **Actual Results** | **Status (Pass/ Fail)** |
| TC\_1 | Testing the output (displaying the file list for each author) | Enter a valid URL of a public GitHub repository | Enter the following URL in the File page  [https://github.com/flyrightsister/udemy-react-testing-projects](https://github.com/flyrightsister/udemy-react-testing-projects.git) | URL of the Github repository should be of a public repo | File list for each author should be shown | After each author, the files they are responsible for should be shown | The output enters an infinite loop. | FAIL |
| TC\_2 | Testing the output (displaying the file list for each author) | Enter a valid URL of a public GitHub repository | Enter the following URL in the File page  <https://github.com/Pragmatists/testing-examples> | URL of the Github repository should be of a public repo | File list for each author should be shown | After each author, the files they are responsible for should be shown | After each author, the files they are responsible for is shown | PASS |
| TC\_3 | Testing the output (displaying the file list for each author) | Enter a valid URL of a public GitHub repository | Enter the following URL in the File page  [https://github.com/googlesamples/assistant-sdk-python](https://github.com/googlesamples/assistant-sdk-python.git) | URL of the Github repository should be of a public repo | File list for each author should be shown | After each author, the files they are responsible for should be shown | After each author, the files they are responsible for is shown | PASS |
| TC\_4 | Testing the output (displaying the file list for each author) | Enter a valid URL of a public GitHub repository | Enter the following URL in the File page  <https://github.com/Parchive/par2cmdline> | URL of the Github repository should be of a public repo | File list for each author should be shown | After each author, the files they are responsible for should be shown | After each author, the files they are responsible for is shown | PASS |

**Most frequently changed file**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test ID** | **Test Scenario** | **Test cases** | **Test data** | **Pre-conditions** | **Post-conditions** | **Expected Results** | **Actual Results** | **Status (Pass/ Fail)** |
| TC\_1 | Testing the output (displaying the most accessed file) | Enter a valid URL of a public GitHub repository | Enter the following URL in the Maxfile page  [https://github.com/flyrightsister/udemy-react-testing-projects](https://github.com/flyrightsister/udemy-react-testing-projects.git) | URL of the Github repository should be of a public repo | File most accessed in the repository should be shown | File most accessed should be shown under the heading “Most accessed file:” | File most accessed is not shown under the heading “Most accessed file:” | FAIL |
| TC\_2 | Testing the output (displaying the most accessed file) | Enter a valid URL of a public GitHub repository | Enter the following URL in the Maxfile page  <https://github.com/Pragmatists/testing-examples> | URL of the Github repository should be of a public repo | File most accessed in the repository should be shown | File most accessed should be shown under the heading “Most accessed file:” | File most accessed is shown under the heading “Most accessed file:” | PASS |
| TC\_3 | Testing the output (displaying the most accessed file) | Enter a valid URL of a public GitHub repository | Enter the following URL in the Maxfile page  [https://github.com/googlesamples/assistant-sdk-python](https://github.com/googlesamples/assistant-sdk-python.git) | URL of the Github repository should be of a public repo | File most accessed in the repository should be shown | File most accessed should be shown under the heading “Most accessed file:” | File most accessed is shown under the heading “Most accessed file:” | PASS |
| TC\_4 | Testing the output (displaying the most accessed file) | Enter a valid URL of a public GitHub repository | Enter the following URL in the Maxfile page  <https://github.com/Parchive/par2cmdline> | URL of the Github repository should be of a public repo | File most accessed in the repository should be shown | File most accessed should be shown under the heading “Most accessed file:” | File most accessed is shown under the heading “Most accessed file:” | PASS |

**Most frequently changed file by each developer**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test ID** | **Test Scenario** | **Test cases** | **Test data** | **Pre-conditions** | **Post-conditions** | **Expected Results** | **Actual Results** | **Status (Pass/ Fail)** |
| TC\_1 | Testing the output (displaying the most accessed file per developer) | Enter a valid URL of a public GitHub repository | Enter the following URL in the Maxfile page  [https://github.com/flyrightsister/udemy-react-testing-projects](https://github.com/flyrightsister/udemy-react-testing-projects.git) | URL of the Github repository should be of a public repo | File most accessed for each developer should be shown. | After each author,the most accessed file should be shown. | Each author and the file accessed by them is shown multiple times | FAIL |
| TC\_2 | Testing the output (displaying the most accessed file per developer) | Enter a valid URL of a public GitHub repository | Enter the following URL in the Maxfile page  <https://github.com/Pragmatists/testing-examples> | URL of the Github repository should be of a public repo | File most accessed for each developer should be shown. | After each author,the most accessed file should be shown. | After each author,the most accessed file is shown. | PASS |
| TC\_3 | Testing the output (displaying the most accessed file per developer) | Enter a valid URL of a public GitHub repository | Enter the following URL in the Maxfile page  [https://github.com/googlesamples/assistant-sdk-python](https://github.com/googlesamples/assistant-sdk-python.git) | URL of the Github repository should be of a public repo | File most accessed for each developer should be shown. | After each author,the most accessed file should be shown. | After each author,the most accessed file is shown. | PASS |
| TC\_4 | Testing the output (displaying the most accessed file per developer) | Enter a valid URL of a public GitHub repository | Enter the following URL in the Maxfile page  <https://github.com/Parchive/par2cmdline> | URL of the Github repository should be of a public repo | File most accessed for each developer should be shown. | After each author,the most accessed file should be shown. | After each author,the most accessed file is shown. | PASS |