Test Report (Student #3)



**Group Number:** C1.037  
**Repository:** <https://github.com/DP2-C1-037/Acme-ANS-D04>

**Workgroup Members:**

Student 1: Ignacio Gutiérrez Serrera - [igngutser@alum.us.es](mailto:igngutser@alum.us.es)

Student 2: Adrián Chabrera Rubio - [adrcharub@alum.us.es](mailto:adrcharub@alum.us.es)

Student 3: Miguel Álvarez Raya - [migalvray@alum.us.es](mailto:migalvray@alum.us.es)

Student 4: Salma El Hakimy - [salel@alum.us.es](mailto:salel@alum.us.es)

Student 5: Alejandro González Macías- [alegonmac@alum.us.es](mailto:alegonmac@alum.us.es)

**Date:** 26/05/2025

Table of Contents

[1 Executive Summary 2](#_Toc199183634)

[2 Revision Table 3](#_Toc199183635)

[3 Introduction 4](#_Toc199183636)

[4 Contents 5](#_Toc199183637)

[4.1 Functional Testing 5](#_Toc199183638)

[4.1.1 Flight Assignment 5](#_Toc199183639)

[4.1.2 Activity Log 14](#_Toc199183640)

[4.2 Performance Testing 20](#_Toc199183641)

[4.2.1 Performance charts 20](#_Toc199183642)

[4.2.2 Confidence Intervals 22](#_Toc199183643)

[4.2.3 Hypothesis Testing 22](#_Toc199183644)

[5 Conclusions 24](#_Toc199183645)

[6 Bibliography 25](#_Toc199183646)

# Executive Summary

This testing report presents the results of both functional and performance testing conducted on the system developed in the Acme-ANS-D04 project. It includes a compilation of the test cases implemented, grouped by feature, their effectiveness in detecting bugs, and the performance metrics obtained through testing on two different computers.

The goal of this analysis is to evaluate the system’s stability, reliability, and efficiency under different execution environments.

# Revision Table

|  |  |  |
| --- | --- | --- |
| **Revision Number** | **Date** | **Description** |
| 1.0 | 25/05/2025 | Functional testing |
| 1.1 | 26/05/2025 | Performance testing |

# Introduction

This document outlines the testing process carried out to assess both the functional and non-functional aspects of the system. The primary objective is to verify that the system adheres to the client’s specified requirements and behaves reliably and efficiently across different execution environments.

Functional testing focuses on ensuring that each implemented feature performs as expected and detects any deviation from the intended behaviour.

Performance testing, on the other hand, evaluates the responsiveness and stability of the system under varying workloads and hardware configurations.

The structure of this document is as follows:

* Section 4 presents the functional test cases, organized by feature. Each test case includes a brief description, the observed outcome, and an assessment of how effective it was at uncovering defects.

Also, it analyses the system’s performance by including execution time measurements, relevant statistical charts, 95% confidence intervals for the wall time, and hypothesis testing to compare performance across different machines.

* Section 5 provides a comprehensive summary of the testing results and highlights the most important findings and conclusions drawn from the process.
* Section 6 lists the references and resources used, if applicable.

# Contents

## Functional Testing

Abbreviations used in this section:

* Member: abbreviation of *flight crew member.*
* Assignment: abbreviation of *flight assignment.*

### 4.1.1 Flight Assignment

#### Flight Assignment: ListCompleted

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case ID | Description | Result | Bug Detection Effectivenes |
| TC-01 | Access the list of *completed flight assignments* being logged in as a member | The system shows only completed and published assignments | Low |
| TC-02 | Access the list with a different role | Access is denied | Low |
| TC-03 | Access the list while unauthenticated | Access is denied | Low |

Note: When accessing this list feature, verify that the attributes *Last Update Moment*, *Status*, and *Duty* are displayed.

#### Flight Assignment: ListMyCompleted

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case ID | Description | Result | Bug Detection Effectivenes |
| TC-04 | Access the list of *my* *completed flight assignments* while logged in as a member | The system shows only the completed assignments (published and not published) linked to the current user | Low |
| TC-05 | There are no completed assignments for the current member | The system returns an empty list without error | Low |
| TC-06 | Access the list with a different role | Access is denied | Low |
| TC-07 | Access the list while unauthenticated | Access is denied | Low |

Note: When accessing this list feature, verify that the attributes *Last Update Moment*, *Status*, and *Duty* are displayed.

#### Flight Assignment: ListPlanned

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case ID | Description | Result | Bug Detection Effectivenes |
| TC-08 | Access the list *planned flight assignments* while logged in as a member | The system shows only future and published assignments | Low |
| TC-09 | There are no planned and published assignments | The system returns an empty list without error | Low |
| TC-10 | Access the list with a different role | Access is denied | Low |
| TC-11 | Access the list while unauthenticated | Access is denied | Low |

Note: When accessing this list feature, verify that the attributes *Last Update Moment*, *Status*, and *Duty* are displayed.

#### Flight Assignment: ListMyPlanned

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case ID | Description | Result | Bug Detection Effectivenes |
| TC-12 | Access the list *my* *planned flight assignments* while logged in as a member | The system shows only my future assignments | Low |
| TC-13 | There are no planned assignments for the current member | The system returns an empty list without error | Low |
| TC-14 | Access the list with a different role | Access is denied | Low |
| TC-15 | Access the list while unauthenticated | Access is denied | Low |

Note: When accessing this list feature, verify that the attributes *Last Update Moment*, *Status*, and *Duty* are displayed.

#### Flight Assignment: Show

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case ID | Description | Result | Bug Detection Effectivenes |
| TC-16 | Access a published planned assignment as the owner member | The system shows the assignment correctly | High \*1 |
| TC-17 | Access a published completed assignment as the owner member | The system shows the assignment correctly | Low |
| TC-18 | Access a draft planned assignment as the owner member | The system shows the assignment correctly | Low |
| TC-19 | Access a draft completed assignment as the owner member | The system shows the assignment correctly | Low |
| TC-20 | Access a published assignment as a member who is not the owner | The system shows the assignment correctly | Low |
| TC-21 | Access a draft assignment as a member who is not the owner | Access is denied | Low |
| TC-22 | Access a published assignment with a different role | Access is denied | Low |
| TC-23 | Access a draft assignment with a different role | Access is denied | Low |
| TC-24 | Access a published assignment while unauthenticated | Access is denied | Low |
| TC-25 | Access a draft assignment while unauthenticated | Access is denied | Low |
| TC-26 | Access an assignment using a non-existing ID as a member (e.g., -1) | Access is denied | Low |

Note: When accessing this view, make sure the following action buttons are correctly displayed: *Log, Delete, Publish* and *Update*. When not in *draft mode*, only the button *Log should be visible*.

\*1 This test case helped identify a bug that caused a panic view. The issue arose from loading different legs depending on whether the item was in draft mode. Once this was fixed, a new error occurred due to an incorrect CSV data sample. It has now been resolved.

#### Flight Assignment: Delete

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case ID | Description | Result | Bug Detection Effectivenes |
| TC-27 | Delete a draft assignment as the owner member | The system deletes the assignment and its draft logs correctly \*2 | Low |
| TC-28 | Delete a draft assignment hacking some values | The system deletes the assignment | Medium \*3 |
| TC-29 | Delete a published assignment as the owner member | Access is denied | Low |
| TC-30 | Delete a draft assignment as a member who is not the owner | Access is denied | Low |
| TC-31 | Delete a draft assignment with a different role | Access is denied | Low |
| TC-32 | |  |  |  | | --- | --- | --- | | |  | | --- | | Delete a draft assignment while unauthenticated |  |  | | --- | |  | |  |  | | --- | |  | | Access is denied | Low |
| TC-33 | Delete an assignment using a non-existing ID as a member (e.g., -1) | Access is denied | Low |
| TC-34 | Delete a draft assignment via GET instead of POST as the owner member | Access is denied | Low |

\*2  \*3 The coverage of this functionality is under 100% because there is only a way to make a delete POST withing the app and is via a button, which can only be accessed in an assignment in draft mode and consequently can only be accessed by his owner. However, this could be done externally (e.g. via postman) and that’s why it must be handled in the code and cannot be tested in the application. Moreover, the *unbind* method is not executed as it is empty and there is one line in red that says that.

#### Flight Assignment: Publish

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case ID | Description | Result | Bug Detection Effectivenes |
| TC-35 | Publish a draft assignment as the owner member with all conditions met | The assignment is published successfully | Low |
| TC-36 | Publish a draft assignment when the selected leg has occurred | The system blocks publication and shows an error message | Low |
| TC-37 | Publish a draft assignment that has attributes with no values | The system blocks publication and shows an error message | Low |
| TC-38 | Publish a draft assignment when the member is not available (not in AVAILABLE status) | The system blocks publication and shows an error message | Low |
| TC-39 | Publish a draft assignment when there is an overlapping leg for the same member | The system blocks publication and shows an error message | Low |
| TC-40 | |  |  |  | | --- | --- | --- | | |  | | --- | | Publish a draft assignment when there is already a pilot assigned to the leg and the current duty is also PILOT |  |  | | --- | |  | |  |  | | --- | |  | | The system blocks publication and shows an error message | Low |
| TC-41 | Publish a draft assignment when there is already a copilot assigned to the leg and the current duty is also COPILOT | The system blocks publication and shows an error message | Low |
| TC-42 | Publish an assignment using a non-existing ID as the owner member (e.g., -1) | Access is denied | Low |
| TC-43 | Publish a non-draft assignment as the owner member | Access is denied | Low |
| TC-44 | Publish a draft assignment as a member who is not the owner | Access is denied | Low |
| TC-45 | Publish a draft assignment with a different role | Access is denied | Low |
| TC-46 | |  | | --- | | Publish a draft assignment while unauthenticated |  |  | | --- | |  | | Access is denied | Low |
| TC-47 | Publish a draft assignment when the assigned leg belongs to another airline | Access is denied | Low |

Note: This function should be tested by iterating over each attribute displayed on the form, trying all the possibilities explained in class, while leaving all other attributes empty. Also, keep in mind that read-only attributes must be tested as well.

Note 2: Although the coverage report indicates 100% coverage for this method, some lines are still marked in yellow. This is likely due to internal limitations of the coverage tool when dealing with complex boolean expressions or method calls within conditionals. All relevant branches and paths have been tested and executed during runtime, so this appears to be a false negative from the coverage analysis tool rather than an actual lack of coverage

#### Flight Assignment: Update

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case ID | Description | Result | Bug Detection Effectivenes |
| TC-48 | Update a draft assignment as the owner member with all valid data | The assignment is updated successfully | Low |
| TC-49 | Update a non-draft assignment as the owner member | Access is denied | Low |
| TC-50 | |  | | --- | | Update a draft assignment as a member who is not the owner |  |  | | --- | |  | | Access is denied | Low |
| TC-51 | Update a draft assignment with a different role | Access is denied | Low |
| TC-52 | |  | | --- | | Update a draft assignment while unauthenticated |  |  | | --- | |  | | Access is denied | Low |
| TC-53 | |  |  |  |  | | --- | --- | --- | --- | | |  |  | | --- | --- | | Update a draft assignment with a leg that is in draft mode | | |  | |  |  | | --- | |  | | Access is denied | Low |
| TC-54 | Update a draft assignment with a leg that belongs to a different airline | Access is denied | Low |
| TC-55 | Update a draft assignment with a leg that is already completed | Access is denied | Low |
| TC-56 | Update a draft assignment using a non-existing ID (e.g., -1) as the owner member | Access is denied | Low |

Note: This function should be tested by iterating over each attribute displayed on the form, trying all the possibilities explained in class, while leaving all other attributes empty. Also, keep in mind that read-only attributes must be tested as well.

Note 2**:** Full coverage could not be achieved for the *authorise*() method due to the complexity of certain conditional branches. Some lines involve a combination of conditions that are difficult to simulate simultaneously in a realistic test scenario. Achieving full branch coverage would require specific and potentially inconsistent data states that are not feasible to produce through normal application flows.

#### Flight Assignment: Create

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case ID | Description | Result | Bug Detection Effectivenes |
| TC-57 | Create a new assignment as a member with valid data and valid leg | The assignment is created in draft mode | Low |
| TC-58 | Create a new assignment as a member with a leg in draft mode | Access is denied | Low |
| TC-59 | |  |  |  |  | | --- | --- | --- | --- | | |  |  | | --- | --- | | Create a new assignment as a member with a leg from another airline | | |  | | | Access is denied | Low |
| TC-60 | |  | | --- | | Create a new assignment as a member with a leg already completed |  |  | | --- | |  | | Access is denied | Low |
| TC-61 | Create a new assignment using a non-existing leg ID (e.g., -1) | Access is denied | Low |
| TC-62 | Create a new assignment while logged in with a different role | Access is denied | Low |
| TC-63 | Create a new assignment while unauthenticated | Access is denied | Low |

|  |  |  |
| --- | --- | --- |
|  |  |  |

Note: This function should be tested by iterating over each attribute displayed on the form, trying all the possibilities explained in class, while leaving all other attributes empty. Also, keep in mind that read-only attributes must be tested as well.

### Activity Log

#### Activity Log: List

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case ID | Description | Result | Bug Detection Effectivenes |
| TC-64 | Access the list of logs of my own assignment | The system shows all activity logs (draft and published) | Low |
| TC-65 | |  | | --- | | Access the list of logs of an assignment owned by another member |  |  | | --- | |  | | |  | | --- | | The system shows only published logs |  |  | | --- | |  | | Low |
| TC-66 | |  | | --- | | Access the list of logs with a different role |  |  | | --- | |  | | Access is denied | Low |
| TC-67 | |  | | --- | | Access the list of logs while unauthenticated |  |  | | --- | |  | | Access is denied | Low |
| TC-68 | |  | | --- | | Access the list of logs using a non-existing assignment ID (e.g., -1) |  |  | | --- | |  | | Access is denied | Low |

Note: When accessing this list feature, verify that the attributes *Flight Number*, *Incident type*, and *Severity level* are displayed.

|  |
| --- |
|  |

#### ***Activity*** Log: Show

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case ID | Description | Result | Bug Detection Effectivenes |
| TC-69 | Access a published log as the owner member | |  |  | | --- | --- | | |  | | --- | | The system shows the log correctly | |  |  | | --- | |  | | Low |
| TC-70 | |  | | --- | | Access a published log as a member who is not the owner |  |  | | --- | |  | | |  |  |  | | --- | --- | --- | | |  | | --- | | The system shows the log correctly |  |  | | --- | |  | |  |  | | --- | |  | | Low |
| TC-71 | |  |  |  | | --- | --- | --- | | |  | | --- | | Access a draft log as the owner member |  |  | | --- | |  | |  |  | | --- | |  | | |  | | --- | | The system shows the log correctly |  |  | | --- | |  | | Low |
| TC-72 | |  | | --- | | Access a published log with a different role |  |  | | --- | |  | | |  | | --- | | Access is denied |  |  | | --- | |  | | Low |
| TC-73 | |  |  |  | | --- | --- | --- | | |  | | --- | | Access a draft log as a member who is not the owner |  |  | | --- | |  | |  |  | | --- | |  | | Access is denied | Low |
| TC-74 | |  | | --- | | Access a draft log with a different role |  |  | | --- | |  | | Access is denied | Low |
| TC-75 | |  | | --- | | Access a published log while unauthenticated |  |  | | --- | |  | | Access is denied | Low |
| TC-76 | |  | | --- | | Access a draft log while unauthenticated |  |  | | --- | |  | | Access is denied | Low |
| TC-77 | |  | | --- | | Access a log using a non-existing ID (e.g., -1) |  |  | | --- | |  | | Access is denied | Low |

Note: When accessing this view, make sure the following action buttons are correctly displayed: *Delete, Publish* and *Update*. When not in *draft mode*, no buttons should be displayed

#### Activity Log: Delete

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case ID | Description | Result | Bug Detection Effectivenes |
| TC-78 | Delete a draft log as the owner member | The system deletes the assignment and its draft logs correctly | Low |
| TC-79 | Delete a draft log hacking some values | The system deletes the assignment | Low |
| TC-80 | Delete a published log as the owner member | Access is denied | Low |
| TC-81 | Delete a draft log as a member who is not the owner | Access is denied | Low |
| TC-82 | Delete a draft log with a different role | Access is denied | Low |
| TC-83 | |  |  |  | | --- | --- | --- | | |  | | --- | | Delete a draft log while unauthenticated |  |  | | --- | |  | |  |  | | --- | |  | | Access is denied | Low |
| TC-84 | Delete a log using a non-existing ID as a member (e.g., -1) | Access is denied | Low |
| TC-85 | Delete a draft log via GET instead of POST as the owner member | Access is denied | Low |

Note: As with the previous DELETE service, the only way to perform a POST delete of an unpublished log is externally (e.g., via Postman). For this reason, the coverage is below 100%. Additionally, the *unbind* method is empty, which also contributes to the reduced coverage.

#### Activity Log: Publish

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case ID | Description | Result | Bug Detection Effectivenes |
| TC-86 | Publish a draft log as the owner member when the assignment is published and the leg is completed | The log is published successfully | Low |
| TC-87 | Publish a draft log as the owner member when the assignment is in draft mode | The system blocks the publication and shows an error message | Low |
| TC-88 | Publish a draft log as the owner member when the leg is not completed yet | The system blocks the publication and shows an error message | Low |
| TC-89 | Publish a published log again as the owner member | Access is denied | Low |
| TC-90 | |  | | --- | | Publish a draft log as a member who is not the owner |  |  | | --- | |  | | Access is denied | Low |
| TC-91 | |  |  |  |  |  | | --- | --- | --- | --- | --- | | |  |  |  | | --- | --- | --- | | |  | | --- | | Publish a draft log with a different role |  |  | | --- | |  | |  |  | | --- | |  | |  |  | | --- | |  | | Access is denied | Low |
| TC-92 | |  | | --- | | Publish a draft log while unauthenticated |  |  | | --- | |  | | Access is denied | Low |
| TC-93 | |  | | --- | | Publish a draft log using a non-existing ID (e.g., -1) |  |  | | --- | |  | | Access is denied | Low |

Note: This function should be tested by iterating over each attribute displayed on the form, trying all the possibilities explained in class, while leaving all other attributes empty. Also, keep in mind that read-only attributes must be tested as well.

#### Activity Log: Create

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case ID | Description | Result | Bug Detection Effectivenes |
| TC-94 | Create a new draft log as the owner of the assignment with valid data | The log is created in draft mode | Low |
| TC-95 | Create a log for an assignment that does not exist (e.g., ID -1) | Access is denied | Low |
| TC-96 | Create a log for an assignment owned by another member | The system blocks the publication and shows an error message | Low |
| TC-97 | |  | | --- | | Create a log using a different role | | Access is denied | Low |
| TC-98 | |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | |  |  |  |  |  | | --- | --- | --- | --- | --- | | |  |  |  | | --- | --- | --- | | |  | | --- | | Create a log while unauthenticated |  |  | | --- | |  | |  |  | | --- | |  | |  |  | | --- | |  | |  |  | | --- | |  | | Access is denied | Low |

Note: This function should be tested by iterating over each attribute displayed on the form, trying all the possibilities explained in class, while leaving all other attributes empty. Also, keep in mind that read-only attributes must be tested as well.

#### Activity Log: Update

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case ID | Description | Result | Bug Detection Effectivenes |
| TC-99 | Update a draft log as the owner member with valid data | The log is updated successfully | Low |
| TC-100 | Update a published log as the owner member | Access is denied | Low |
| TC-101 | Update a draft log as a member who is not the owner | The system blocks the publication and shows an error message | Low |
| TC-102 | |  | | --- | | Update a draft log using a different role | | Access is denied | Low |
| TC-103 | |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | |  |  |  |  | | --- | --- | --- | --- | | |  |  | | --- | --- | | |  | | --- | | Update a draft log while unauthenticated | |  |  | | --- | |  | |  |  | | --- | |  | | | Access is denied | Low |
| TC-104 | Update a draft log using a non-existing ID (e.g., -1) | Access is denied | Low |

Note: This function should be tested by iterating over each attribute displayed on the form, trying all the possibilities explained in class, while leaving all other attributes empty. Also, keep in mind that read-only attributes must be tested as well.

|  |  |
| --- | --- |
| |  | | --- | |  | |

## Performance Testing

### Performance charts

Performance testing was conducted to evaluate how the system performs under different hardware configurations. Execution times were collected from two different machines, referred to as PC A and PC B. The performance charts illustrate the distribution of execution times for both environments, revealing that PC B consistently achieves lower execution times compared to PC A.

PC A

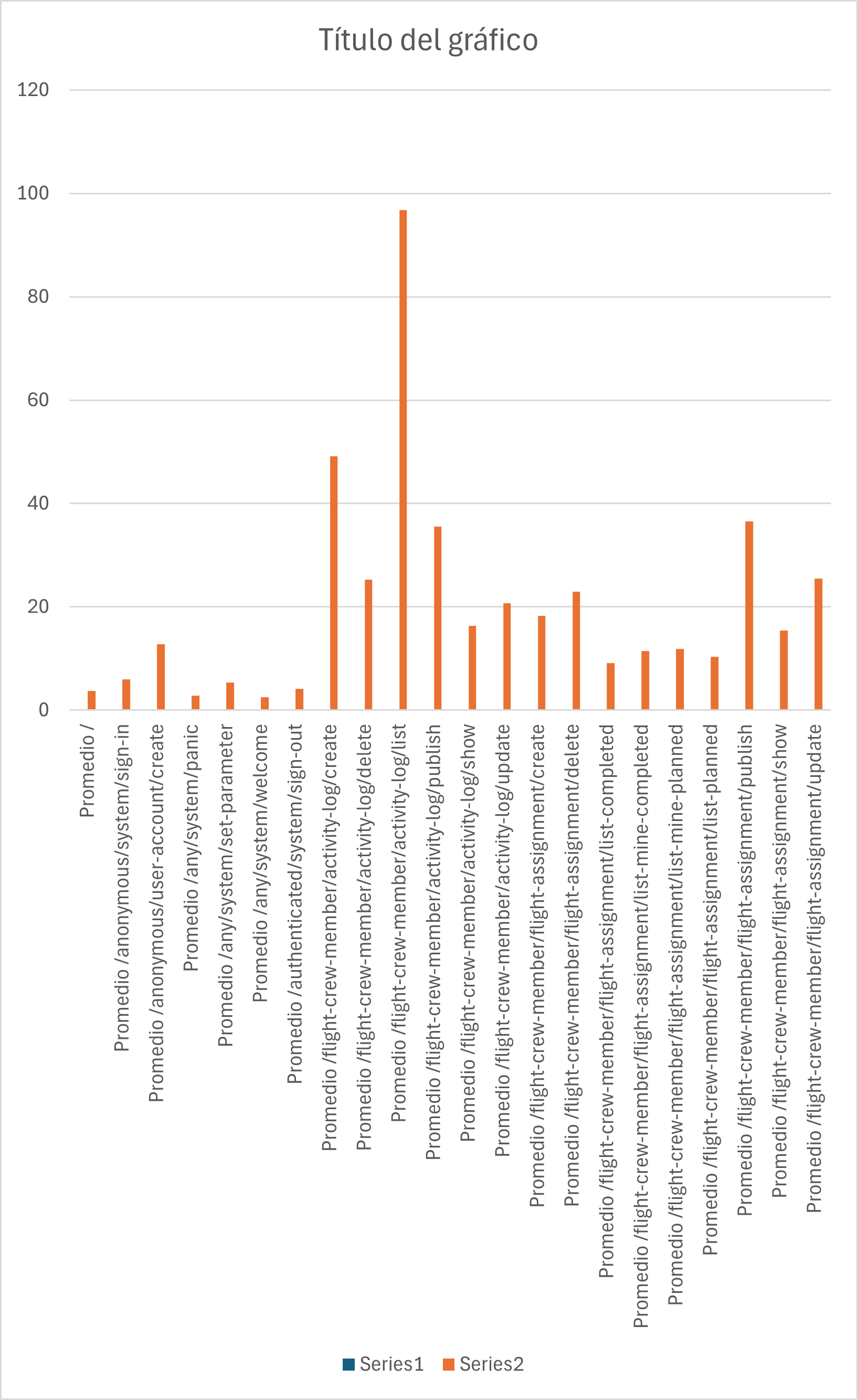
Gráfico

El contenido generado por IA puede ser incorrecto.

Stadistical summary:

|  |  |
| --- | --- |
| *PC A* | |
|  |  |
| Mean | 21.0842244 |
| Standard Error | 0.75860211 |
| Median | 14.8977 |
| Mode | 2.5885 |
| Standard Deviation | 31.743669 |
| Sample Variance | 1007.66052 |
| Kurtosis | 132.214437 |
| Skewness | 9.33466915 |
| Range | 564.7896 |
| Minimum | 1.2752 |
| Maximum | 566.0648 |
| Sum | 36918.4769 |
| Count | 1751 |
| Confidence Level (95.0%) | 1.48786187 |

PC B



Stadistical summary:

|  |  |
| --- | --- |
| *PC B* | |
|  |  |
| Mean | 10.2546509 |
| Standard Error | 0.52506454 |
| Median | 7.12285 |
| Mode | 11.4826 |
| Standard Deviation | 20.6183959 |
| Sample Variance | 425.118249 |
| Kurtosis | 250.356164 |
| Skewness | 14.4218163 |
| Range | 404.1137 |
| Minimum | 0.7663 |
| Maximum | 404.88 |
| Sum | 15812.6717 |
| Count | 1542 |
| Confidence Level (95.0%) | 1.02991651 |

### Confidence Intervals

To assess the statistical reliability of the results, 95% confidence intervals were calculated for the average execution times on each machine:

* PC A had a mean execution time of 21.08 ms, with a 95% confidence interval of ±1.49 ms.
* PC B had a mean execution time of 10.25 ms, with a 95% confidence interval of ±1.03 ms.

These intervals indicate that the observed difference in performance between the two machines is statistically significant.

### Hypothesis Testing

A z-test for two independent means was performed to determine whether the performance difference between PC A and PC B is statistically significant. The null hypothesis stated that both machines have equal mean execution times.

* The test yielded a z-score of 5.33 and a p-value < 0.0000001.
* Since the p-value is far below the standard threshold (0.05), we reject the null hypothesis.

This confirms that PC B is significantly faster than PC A in executing the system operations, demonstrating the influence of hardware configuration on the system's responsiveness.

|  |  |  |
| --- | --- | --- |
| z-Test: Two Sample for Means |  |  |
|  |  |  |
|  | *Before* | *After* |
| Mean | 19.4425085 | 14.5728985 |
| Known Variance | 1007.66052 | 613.906938 |
| Observations | 1942 | 1942 |
| Hypothesized Mean Difference | 0 |  |
| Z | 5.32906828 |  |
| P(Z<=z) one-tail | 4.9359E-08 |  |
| Z Critical one-tail | 1.64485363 |  |
| P(Z<=z) two-tail | 9.8718E-08 |  |
| Z Critical two-tail | 1.95996398 |  |

# Conclusions

This testing report provides a comprehensive evaluation of the system from both a functional and performance standpoint.

On the functional side, the test suite thoroughly covered all core features related to flight assignments and activity logs. All functionalities behaved as expected under normal and boundary conditions, and proper access control was enforced across roles and authentication states. Although most test cases had low bug detection effectiveness—indicating a robust implementation—at least one critical bug was successfully identified and resolved. The coverage was overall high, and even where the test coverage tool reported partial gaps, manual review and runtime analysis confirmed that the relevant paths were executed.

On the performance side, execution time metrics were gathered on two machines (PC A and PC B) to assess how hardware affects system responsiveness. The results showed a clear performance advantage in favor of PC B. Confidence intervals demonstrated that the differences in execution times are statistically reliable, and hypothesis testing confirmed that PC B outperforms PC A with high significance.

In conclusion, the system not only satisfies the functional requirements with strong stability and correctness, but it also performs efficiently in favorable hardware environments. These results validate both the quality of the implementation and the effectiveness of the test strategy.

# Bibliography

Intentionally blank.