

University of Sevilla

Higher Technical School of Computer Engineering

## D04 – Student #3 Testing Report



Degree in Computer Engineering - Software Engineering

Desing and Testing II

Course 2024 – 2025

Date	Version
23/05/2025	v1.0

Practice Group: C3.035	
Members	Email
Isabel Sánchez Castro	isasancas@alum.us.es

**Repository:** <https://github.com/DP2-2025-C1-040/Acme-ANS>

## Table of contents

1. Revision Table .....	3
2. Execution summary.....	4
3. Introduction .....	5
4. Functional Testing.....	6
5. Performance testing .....	10
6. Conclusions .....	13
7. Bibliography .....	14

## 1. Revision Table

Date	Version	Description	Delivery
23/05/2025	v1.0	First report's version	D04
24/05/2025	v2.0	Add comparison between two computers	D04
25/06/2025	v3.0	Corrections First Call	C2
11/10/2025	v4.0	Third Call modifications	C3

## 2. Execution summary

For this testing document, each of the implementations present in the mandatory requirements of the D04 deliverable have been developed and explained.

Both the system performance and the functional performance of each and every one of the requested functions have been evaluated, in this case, those related to 'flight assignment' and 'activity log'. The methodology provided in Lesson 04, 'S01 - Formal testing' and 'S02 - Performance testing' was followed.

The system performs robustly in terms of functionality, although there are areas that could use a little attention.

### 3. Introduction

This document is divided into two distinct sections:

1. Functional testing: a list of implemented test cases, grouped by functionality, will be presented. For each one, a description and an indication of how effective it is in detecting errors will be given. For effectiveness, the code coverage and a screenshot of the code will be used to check that all possible decisions have been tested during the execution of the program to avoid the existence of errors.
2. Performance testing: appropriate graphs and a 95% confidence interval will be provided for the time taken for the requests in the tests on two different computers. In addition, after the tests on the different computers, it will be indicated which of these is the most powerful and offers the best performance.








## 4. Functional Testing

Tests cases related to flight assignment:

- Create flight assignment:
  - Description: the restrictions of all the fields of the flight assignment creation form are tested with values related to positive, negative and hacking cases.
  - Coverage: 99.6%
  - Effectiveness: coverage is almost 100%, so it is almost perfect. We notice some yellow lines in some cases of null objects, for example 'crewMember != null', because in no case this object is going to be null.
- Delete flight assignment:
  - Description: the elimination of a flight assignment is tested, with positive, negative and hacking cases. In addition, its associated activity logs are checked for deletion.
  - Coverage: 56.1%
  - Effectiveness: in this case, the coverage is 56% due to the unbind method and some "object != null", as in no case this method is entered but it is necessary for the other services.
- List landed flight assignments:
  - Description: the list of landed flight assignments associated with the currently logged-in flight crew member is tested. The different hacking cases are also verified.
  - Coverage: 100%
  - Effectiveness: perfect coverage.
- List programmed flight assignments:
  - Description: the list of programmed flight assignments associated with the currently logged-in flight crew member is tested. The different hacking cases are also verified.
  - Coverage: 100%
  - Effectiveness: perfect coverage.
- Publish flight assignment:
  - Description: the publication of a flight assignment is tested. It is checked that if the leg has already occurred, it cannot be published and the different cases of hacking.
  - Coverage: 97.9%

- Effectiveness: almost perfect coverage with some yellow lines in some cases of null objects.
- Show flight assignment:
  - Description: the display of a flight assignment, with its corresponding flight crew member and associated leg, is tested. In addition, the different hacking cases are tested.
  - Coverage: 98.9%
  - Effectiveness: near-perfect coverage. We notice some yellow lines in some cases of null objects, for example 'assignment != null', because in no case this object is going to be null.
- Update flight assignment:
  - Description: the constraints of all fields of a flight assignment update form are tested with values for positive, negative and hacking cases.
  - Coverage: 99.2%
  - Effectiveness: near-perfect coverage. Same case as above with null objects.

In short, the only red thing that can be seen is the unbind method of the delete service, since it is not executed at any time, as it is impossible to enter this condition. On the other hand, the lines in yellow have already explained the reason for their cause. For all these reasons and for having an average coverage of 93.4% in flight assignment, it is considered that the existence of potential failures or bugs is negligible.

 acme.features.crew.assignment	 93,4 %
>  CrewAssignmentDeleteService.java	 56,1 %
>  CrewAssignmentPublishService.java	 97,9 %
>  CrewAssignmentUpdateService.java	 99,2 %
>  CrewAssignmentShowService.java	 98,9 %
>  CrewAssignmentCreateService.java	 99,6 %
>  CrewAssignmentController.java	 100,0 %
>  CrewAssignmentListServiceLanded.java	 100,0 %
>  CrewAssignmentListServiceProgrammed.java	 100,0 %









Test cases relating to activity log records:

- Create activity log:
  - Description: the restrictions of all the fields of the activity log creation form are tested with values related to positive, negative and hacking cases.
  - Coverage: 100%
  - Effectiveness: perfect coverage.
- Delete activity log:
  - Description: the elimination of an activity log is tested, with positive, negative and hacking cases.
  - Coverage: 70.5%
  - Effectiveness: coverage is around 70% due to the unbind method, as in the delete flight assignment. This is not a cause for concern as the unbind method is the same as in other services and is already proven to work correctly in those services.
- List activity logs:
  - Description: the list of activity logs associated with the currently logged-in flight crew member is tested. The different hacking cases are also verified.
  - Coverage: 100%
  - Effectiveness: perfect coverage.
- Publish activity log:
  - Description: the publication of a activity log is tested. It is checked that if the associated flight assignment is not published, the activity log cannot be published and the different cases of hacking.
  - Coverage: 100%
  - Effectiveness: perfect coverage.
- Show activity log:
  - Description: the display of a activity log, with its corresponding flight assignment, is tested. In addition, the different hacking cases are tested.
  - Coverage: 100%
  - Effectiveness: perfect coverage.



- Update activity log:
  - Description: the constraints of all fields of a activity log update form are tested with values for positive, negative and hacking cases.
  - Coverage: 100%
  - Effectiveness: perfect coverage.

In short, the only red thing that can be seen is the unbind method of the delete service, since it is not executed at any time, as it is impossible to enter this condition. On the other hand, the lines in yellow have already explained the reason for their cause. For all these reasons and for having an average coverage of 95.3% in activity log, it is considered that the existence of potential failures or bugs is negligible.

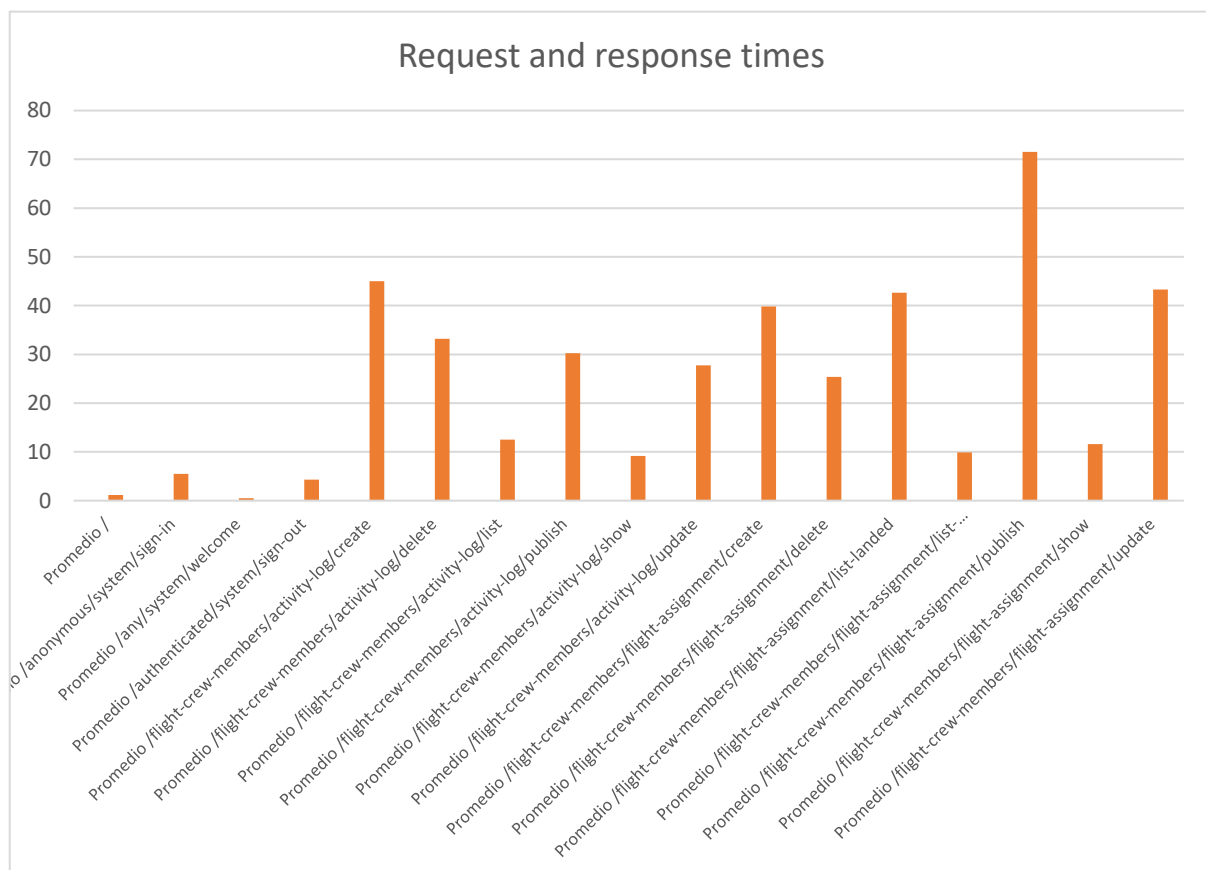
acme.features.crew.activityLog		95,3 %
> CrewActivityLogDeleteService.java		70,5 %
> CrewActivityLogController.java		100,0 %
> CrewActivityLogCreateService.java		100,0 %
> CrewActivityLogListService.java		100,0 %
> CrewActivityLogPublishService.java		100,0 %
> CrewActivityLogShowService.java		100,0 %
> CrewActivityLogUpdateService.java		100,0 %

## 5. Performance testing

After performing the set of tests for the appropriate functionalities, all the steps shown in 'S02 - Performance testing' have been carried out, obtaining the following results:

Promedio /	1,18750092
Promedio /anonymous/system/sign-in	5,4842303
Promedio /any/system/welcome	0,52432617
Promedio /authenticated/system/sign-out	4,33166
Promedio /flight-crew-members/activity-log/create	45,029025
Promedio /flight-crew-members/activity-log/delete	33,1611167
Promedio /flight-crew-members/activity-log/list	12,5271276
Promedio /flight-crew-members/activity-log/publish	30,2173435
Promedio /flight-crew-members/activity-log/show	9,19304348
Promedio /flight-crew-members/activity-log/update	27,762835
Promedio /flight-crew-members/flight-assignment/create	39,7958607
Promedio /flight-crew-members/flight-assignment/delete	25,4078
Promedio /flight-crew-members/flight-assignment/list-landed	42,6656038
Promedio /flight-crew-members/flight-assignment/list-programmed	9,92002162
Promedio /flight-crew-members/flight-assignment/publish	71,5111667
Promedio /flight-crew-members/flight-assignment/show	11,6129977
Promedio /flight-crew-members/flight-assignment/update	43,315005
Promedio general	12,8928727

As can be seen in the images above, the average time taken by the system to perform a request is approximately 12.9 ms, i.e. 0.013 seconds, which is quite fast.



Furthermore, the bar chart clearly shows that the requests that take the longest time are those that handle a larger amount of data and validations, specifically those related to the publish and update of a flight assignment and create an activity log. This is partly due to the fact that several validations of the FlightAssignment class and the associated ActivityLog class (because access to the associated activity logs requires access to flight assignments).

Although the performance tests without indexes met the established requirements (with an average of 13 ms), it was decided to add indexes on the most queried fields. The appropriate comparisons are made of the different values obtained in each test and we obtain:

Before				After		
Media	12,8928727			Media	9,08329574	
Error típico	0,79219113			Error típico	0,53713397	
Mediana	6,29665			Mediana	4,7351	
Moda	0,4877			Moda	0,2585	
Desviación estándar	21,3744853			Desviación estándar	14,4926669	
Varianza de la muestra	456,868623			Varianza de la muestra	210,037394	
Curtosis	9,61980721			Curtosis	7,57141077	
Coeficiente de asimetría	2,87375366			Coeficiente de asimetría	2,645228	
Rango	154,7567			Rango	92,1211	
Mínimo	0,2235			Mínimo	0,1905	
Máximo	154,9802			Máximo	92,3116	
Suma	9386,0113			Suma	6612,6393	
Cuenta	728			Cuenta	728	
Nivel de confianza(95,0%)	1,55525532			Nivel de confianza(95,0%)	1,05451883	
Interval (ms)	11,3376173	14,448128		Interval (ms)	8,02877691	10,1378146
Interval (s)	0,01133762	0,01444813		Interval (s)	0,00802878	0,01013781

The indexes reduced the average to 9.1 ms (has improved by 30%), which improves the stability of the system and prepares it for higher data volume scenarios.

To determine whether the average times before and after the changes can be considered the same or not, a z-test was performed with the following results:

Prueba z para medias de dos muestras		
	before	after
Media	12,8928727	9,08329574
Varianza (conocida)	456,868623	210,037394
Observaciones	728	728
Diferencia hipotética de las m	0	
z	3,98024787	
P(Z<=z) una cola	3,4422E-05	
Valor crítico de z (una cola)	1,64485363	
Valor crítico de z (dos colas)	6,8843E-05	
Valor crítico de z (dos colas)	1,95996398	

Given this two-tailed p-value result ( $\approx 0$ ), we can conclude that, being in the interval between  $[0, 0.05)$ , the changes made have been fruitful and have helped to improve performance.

The performance of the system will now be compared on two different computers. The first computer will be the one on which all the above tests have been performed and the second will be another team member's computer (a bit more powerful than mine). Here are the results:

Computer 1				Computer 2			
Media	9,08329574			Media	6,9727261		
Error típico	0,53713397			Error típico	0,40977822		
Mediana	4,7351			Mediana	3,95595		
Moda	0,2585			Moda	0,3842		
Desviación estándar	14,4926669			Desviación estándar	11,0564208		
Varianza de la muestra	210,037394			Varianza de la muestra	122,244442		
Curtosis	7,57141077			Curtosis	12,2583189		
Coeficiente de asimetría	2,645228			Coeficiente de asimetría	3,00224684		
Rango	92,1211			Rango	99,6946		
Mínimo	0,1905			Mínimo	0,1456		
Máximo	92,3116			Máximo	99,8402		
Suma	6612,6393			Suma	5076,1446		
Cuenta	728			Cuenta	728		
Nivel de confianza(95,0%)	1,05451883			Nivel de confianza(95,0%)	0,80448989		
Interval (ms)	8,02877691	10,1378146		Interval (ms)	6,16823621	7,77721599	
Interval (s)	0,00802878	0,01013781		Interval (s)	0,00616824	0,00777722	

We can see the difference in results.

Prueba z para medias de dos muestras		
	Computer 1	Computer 2
Media	9,08329574	6,9727261
Varianza (conocida)	210,037394	122,244442
Observaciones	728	728
Diferencia hipotética de las medias	0	
z	3,12400756	
P(Z<=z) una cola	0,00089203	
Valor crítico de z (una cola)	1,64485363	
Valor crítico de z (dos colas)	0,00178406	
Valor crítico de z (dos colas)	1,95996398	

As can be seen, the value of the two-tail p-value is 0.00178, value which is in the Interval [0, 0.05).

## 6. Conclusions

After creating this testing report, it has been concluded that this phase of the project lifecycle is vital. Verifying that all developed functions work correctly and are thoroughly checked to minimise errors or failures, as well as ensuring that performance is optimised as much as possible, are fundamental aspects for the client. A thoroughly tested system allows the end-user to use it quickly and intuitively, avoiding problems that can detract from their experience.

In addition, a meticulous testing process contributes to customer satisfaction and product reputation, ensuring that quality and efficiency expectations are consistently met.

## 7. Bibliography

- 06 Annexes - Material provided in the subject Design and Testing II by the University of Seville.
- L04 - S01 - Formal testing - Material provided in the subject Design and Testing II by the University of Seville.
- L04 - S02 - Performance testing - Material provided in the subject Design and Testing II by the University of Seville.