**Testing report**

**Acme Ans D04**

**Dibujo animado de un personaje animado

El contenido generado por IA puede ser incorrecto.**

**Juan Moreno Ríos, juamorrio@alum.us.es**

**C1.023**

**https://github.com/ManunGar/Acme-ANS-D04  
Seville may 24, 2025**

Content

[1. Executive Summary 3](#_Toc199145005)

[2. Revision table 4](#_Toc199145006)

[3. Introduction 5](#_Toc199145007)

[4. Functional Testing 6](#_Toc199145008)

[4.1 Test cases for list features 6](#_Toc199145009)

[4.2 Test cases for show features 6](#_Toc199145010)

[4.2 Test cases for create features 6](#_Toc199145011)

[4.4 Test cases for publish features 11](#_Toc199145012)

[5. Performance Testing 14](#_Toc199145013)

[5.1 First device performance tests 14](#_Toc199145014)

[5.2 Second device performance tests 14](#_Toc199145014)

[5.3 Comparison between the two devices 17](#_Toc199145088)

[6. Conclusion 19](#_Toc199145089)

[7. Bibliography 20](#_Toc199145090)

# 1. Executive Summary

This document presents the results of the tests carried out on the functionalities developed by student 2 for the role of customer in the Acme ANS system. The functional tests were designed to verify the correct behavior of the system in different situations: valid entries, invalid and unauthorized access, verifying that the established validations are met without detecting serious errors.

In addition, performance tests were carried out on two different devices. The response times recorded were statistically analyzed, including the calculation of 95% confidence intervals and the comparison of means using a Z-test, in order to evaluate the efficiency of the system in different environments.

The results obtained allow us to conclude that the tested functionalities are robust, stable and adequately meet the established requirements, both from the functional and performance point of view.

# 2. Revision table

| **Version** | **Date** | **Description** |
| --- | --- | --- |
| 0.0 | 16/05/2025 | Initial version |
| 1.0 | 20/05/2025 | Implementing Functional Testing |
| 2.0 | 23/05/2025 | Implementing Performance Testing |

# 3. Introduction

The objective of this report is to document and analyze the functional and performance tests applied on the system developed from the Acme ANS project, specifically for the functionalities associated with the role of customer. These tests were implemented by Student 2 and cover key operations related to the Booking, Passenger and BookingRecord entities.

The functional tests have been structured following systematic validation criteria to ensure the correct operation of the system under different conditions of use. At the same time, performance tests have been carried out on different devices, statistically analysing the data collected to obtain relevant conclusions about the efficiency of the application in different technical contexts.

This report provides a comprehensive view of the quality validation process, offering key information for its evaluation and continuous improvement.

# 4. Functional Testing

In this section, the different test cases carried out by Student 2, corresponding to requirements 8 and 9, will be presented. The description of the test case, the result of the test case, and the effectiveness of finding bugs will be displayed.

## 4.1 Test cases for list features

|  |  |  |
| --- | --- | --- |
| Description | Result | Bugs detected |
| List Reservations | All bookings belonging to a customer were shown | None |
| List reservations that don't belong to the authenticated customer | An error appeared access not authorise | None |
| List bookings with another user who is not a customer | An error appeared access not authorise | None |
| List passengers | All passengers belonging to a customer were shown | None |
| List passengers belonging to a reservation | All passengers belonging to a booking were shown | None |
| List passengers that do not belong to the authenticated buyer | An error appeared access not authorise | None |
| List bookings with another user who is not a customer | An error appeared access not authorise | None |

## 4.2 Test cases for show features

|  |  |  |
| --- | --- | --- |
| Description | Result | Bugs detected |
| Show a booking | The data related to that booking was shown | None |
| Show a reservation that doesn't belong to the authenticated customer | An error appeared access not authorise | None |
| Show a booking with another user who is not a customer | An error appeared access not authorise | None |
| Show a passenger | The data related to that passenger was shown | None |
| Show a passenger that does not belong to the authenticated customer | An error appeared access not authorise | None |
| Show a passenger with another user who is not a customer | An error appeared access not authorise | None |

## 4.2 Test cases for create features

|  |  |  |
| --- | --- | --- |
| Description | Result | Bugs detected |
| Create a booking with all fields empty | The system prevented the creation of the booking with non-blocking errors | None |
| Create a booking with a flight that shouldn't be accessible to the customer | An error appeared access not authorise | None |
| Create a booking with an invalid location code | The system prevented the creation of the booking with a non-blocking error in the code format | None |
| Create a booking with a repeat location code | The system prevented the creation of the booking with a non-blocking error indicating that a booking with that code already existed | None |
| Create a booking with a valid location code | The system allowed the creation of the booking | None |
| Create a booking by modifying the purchase date (read-only value) | The system allowed the creation of the booking by restoring the default value of the purchase date | None |
| Create a booking without changing the purchase date | The system allowed the creation of the booking | None |
| Create a booking with a changed travel class value | An error appeared access not authorise | None |
| Create a booking with a valid travel class value (value within the values listed) | The system allowed the creation of the booking | None |
| Create a reservation without a value of the last nibble of the credit card | The system allowed the creation of the booking since it is an optional value | None |
| Create a reservation with an invalid value of the last nibble of the credit card | The system prevented the creation of the booking with non-blocking errors | None |
| Create a booking with a valid last nibble value | The system allowed the creation of the booking | None |
| Create a booking by modifying the price (read-only value) | The system allowed the creation of the booking by restoring the default value of the price | None |
| Create a booking without changing the price | The system allowed the creation of the booking | None |
| Create a booking without being a user with the customer role | An error appeared access not authorise | None |
| Create a booking with a hacked flight value with a string value "Hacked" | An error appeared access not authorise | It was detected that this hacking casuistry was not correctly controlled in the authorise method |
| Create a relationship between a booking and a passenger | The system allowed the creation of the relationship | None |
| Create a relationship between a booking and a passenger with empty values | The system prevented the creation of the non-blocking error relationship | None |
| Create a relationship between a booking and an existing passenger | The system prevented the creation of the relationship with a non-blocking error indicating that the relationship already existed | None |
| Create a relationship between a booking and a passenger with hacked values from one of the fields | An error appeared access not authorise | None |
| Create a passenger with all empty values | The system prevented the creation of the passenger with a non-blocking error | None |
| Create a passenger with the full name that is too long | The system prevented the creation of the passenger with a non-blocking error | None |
| Create a passenger with the full name with non-Latin values | The system allowed the creation of the passenger | None |
| Create a passenger with the full name with a valid Latin value | The system allowed the creation of the passenger | None |
| Create a passenger with an invalid email address | The system prevented the creation of the passenger with a non-blocking error | None |
| Create a passenger with a valid email address | The system allowed the creation of the passenger | None |
| Create a passenger with an invalid passport | The system prevented the creation of the passenger with a non-blocking error indicating that they had to follow the format | None |
| Create a passenger with a valid passport | The system allowed the creation of the passenger | None |
| Create a passenger with an invalid birthday | The system prevented the creation of the passenger with a non-blocking error indicating that they had to follow the format | None |
| Create a passenger with a future birthday | The system prevented the creation of the passenger with a non-blocking error indicating that it must have been in the past | None |
| Create a passenger with a past birthday | The system allowed the creation of the passenger | None |
| Create a passenger with no specified special needs | The system allowed the creation of the passenger since it is an optional value | None |
| Create a passenger with specified special needs by exceeding the 50-character limit | The system prevented the creation of the passenger with a non-blocking error | None |
| Create a passenger with specified special needs within the allowed range of characters | The system allowed the creation of the passenger | None |
| Create a passenger without being a system-authorized customer | An error appeared access not authorise | None |

4.3 Test cases for update features

|  |  |  |
| --- | --- | --- |
| Description | Result | Bugs detected |
| Update a booking with all fields empty | The system prevented the creation of the booking with non-blocking errors | None |
| Updating a booking with a flight that shouldn't be accessible to the customer | An error appeared access not authorise | None |
| Update a booking with an invalid location code | The system prevented the booking update with a non-blocking error in the code format | None |
| Update a booking with a repeat location code | The system prevented the booking from being updated with a non-blocking error indicating that a booking with that code already existed | None |
| Update a booking with a valid location code | The system allowed the updating of the booking | None |
| Update a booking by changing the purchase date (read-only value) | The system allowed the update of the booking by restoring the default value of the purchase date | None |
| Update a booking without changing the purchase date | The system allowed the updating of the booking | None |
| Update a booking with a changed travel class value | An error appeared access not authorise | None |
| Update a booking with a valid travel class value (value within the values listed) | The system allowed the updating of the booking | None |
| Updating a booking without a last nibble value | The system allowed the update of the booking since it is an optional value | None |
| Update a booking with an invalid value of the last nibble of the credit card | The system prevented booking update with non-blocking errors | None |
| Update a booking with a valid value of the last nibble of the credit card | The system allowed the updating of the booking | None |
| Update a booking by changing the price (read-only value) | The system allowed the booking to be updated by restoring the default value of the price | None |
| Update a booking without changing the price | The system allowed the updating of the booking | None |
| Update a booking without being a user with the customer role | An error appeared access not authorise | None |
| Update a booking that doesn't belong to the authenticated customer | An error appeared access not authorise | None |
| Update a booking already published by the customer | An error appeared access not authorise | None |
| Update a booking with a hacked flight value with a string value "Hacked" | An error appeared access not authorise | It was detected that this hacking casuistry was not correctly controlled in the authorise method |
| Update a passenger with all empty values | The system prevented the passenger from updating with a non-blocking error | None |
| Updating a passenger with a full name that is too long | The system prevented the passenger from updating with a non-blocking error | None |
| Update a passenger with the full name with non-Latin values | The system allowed the passenger to be updated | None |
| Update a passenger with the full name with a valid Latin value | The system allowed the passenger to be updated | None |
| Update a passenger with an invalid email | The system prevented the passenger from updating with a non-blocking error | None |
| Update a passenger with a valid email address | The system allowed the passenger to be updated | None |
| Updating a passenger with an invalid passport | The system prevented the passenger from updating with a non-blocking error indicating that they should follow the format | None |
| Updating a passenger with a valid passport | The system allowed the passenger to be updated | None |
| Update a passenger with an invalid birthday | The system prevented the passenger from updating with a non-blocking error indicating that they should follow the format | None |
| Update a passenger with a future birthday date | The system prevented the passenger from updating with a non-blocking error indicating that it must have been in the past | None |
| Updating a passenger with a past birthday | The system allowed the passenger to be updated | None |
| Upgrade a passenger without specified special needs | The system allowed the passenger to be updated as it is an optional value | None |
| Upgrade a passenger with specified special needs by exceeding the 50-character limit | The system prevented the passenger from updating with a non-blocking error | None |
| Upgrade a passenger with specified special needs within the allowed range of characters | The system allowed the passenger to be updated | None |
| Upgrade a passenger without being a system-authorized purchaser | An error appeared access not authorise | None |
| Update a passenger already published by the system | An error appeared access not authorise | None |
| Update a passenger that doesn't belong to the authenticated customer | An error appeared access not authorise | None |

## 4.4 Test cases for publish features

|  |  |  |
| --- | --- | --- |
| Description | Result | Bugs detected |
| Publish a booking with all fields empty | The system prevented the publication of the booking with non-blocking errors | None |
| Publish a booking with a flight that shouldn't be accessible to the customer | An error appeared access not authorise | None |
| Publish a booking with an invalid location code | The system prevented the publication of the booking with a non-blocking error in the code format | None |
| Publish a booking with a repeat location code | The system prevented the publication of the booking with a non-blocking error indicating that a reservation with that code already existed | None |
| Publish a booking with a valid location code | The system allowed the publication of the booking | None |
| Publish a booking by modifying the purchase date (read-only value) | The system allowed the publication of the booking by restoring the default value of the purchase date | None |
| Publish a booking without changing the purchase date | The system allowed the publication of the booking | None |
| Publish a booking with a changed travel class value | An error appeared access not authorise | None |
| Publish a booking with a valid travel class value (value within the values listed) | The system allowed the publication of the booking | None |
| Publish a booking without a last nibble value | The system prevented the publication of the booking with non-blocking errors since the existence of the value in the system is mandatory | None |
| Publish a booking with an invalid value of the last digits of the credit card | The system prevented the publication of the booking with non-blocking errors | None |
| Publish a booking with a valid value of the last nibble of the credit card | The system allowed the publication of the booking | None |
| Publish a booking by modifying the price (read-only value) | The system allowed the publication of the booking by restoring the default value of the price | None |
| Publish a booking without changing the price | The system allowed the publication of the booking | None |
| Publish a booking without being a user with the customer role | An error appeared access not authorise | None |
| Publish a booking that does not belong to the authenticated customer | An error appeared access not authorise | None |
| Publish a booking already published by the customer | An error appeared access not authorise | None |
| Publish a booking with a hacked flight value with a string value "Hacked" | An error appeared access not authorise | It was detected that this hacking casuistry was not correctly controlled in the authorise method |
| Publish a passenger with all empty values | The system prevented the passenger with a non-blocking error from posting | None |
| Publish a passenger with a full name that is too long | The system prevented the passenger with a non-blocking error from posting | None |
| Publish a passenger with the full name with non-Latino values | The system allowed the passenger to publish | None |
| Publish a passenger with the full name with a valid Latin value | The system allowed the passenger to publish | None |
| Publish a passenger with an invalid email | The system prevented the passenger with a non-blocking error from posting | None |
| Publish a passenger with a valid email address | The system allowed the passenger to publish | None |
| Publish a passenger with an invalid passport | The system prevented the passenger from posting with a non-blocking error indicating that they should follow the format | None |
| Publish a passenger with a valid passport | The system allowed the passenger to publish | None |
| Publish a passenger with an invalid birthday | The system prevented the passenger from posting with a non-blocking error indicating that they should follow the format | None |
| Publish a passenger with a future birthday | The system prevented the passenger from posting a non-blocking error indicating that it must have been in the past | None |
| Publish a passenger with a past birthday | The system allowed the passenger to publish | None |
| Publish a passenger with no specified special needs | The system allowed the passenger to be published as it is an optional value | None |
| Publish a passenger with specified special needs over the 50-character limit | The system prevented the passenger with a non-blocking error from posting | None |
| publish a passenger with specified special needs within the allowed range of characters | The system allowed the passenger to publish | None |
| publish a passenger without being a system-authorized customer | An error appeared access not authorise | None |
| publish a passenger that doesn't belong to the authenticated customer | An error appeared access not authorise | None |

# 

# 5. Performance Testing

Performance tests have been conducted on two different devices. The results on both devices will then be displayed, as well as the final comparison between the two.

## 5.1 First device performance tests

Regarding the first device, the following results were obtained:

|  |  |  |
| --- | --- | --- |
| **Average /** |  | 1,628 |
| **Average /anonymous/system/sign-in** | | 2,76817581 |
| **Average /any/system/welcome** | | 0,94417984 |
| **Average /authenticated/system/sign-out** | | 1,70798889 |
| **Average /customer/booking/create** | | 25,2645421 |
| **Average /customer/booking/delete** | | 14,66826 |
| **Average /customer/booking/list** | | 5,34145429 |
| **Average /customer/booking/publish** | | 13,3401214 |
| **Average /customer/booking/show** | | 5,14438947 |
| **Average /customer/booking/update** | | 18,5348857 |
| **Average /customer/booking-record/create** | | 21,6789909 |
| **Average /customer/passenger/create** | | 7,85007959 |
| **Average /customer/passenger/delete** | | 11,1066 |
| **Average /customer/passenger/list** | | 3,406 |
| **Average /customer/passenger/list-menu** | | 5,20178182 |
| **Average /customer/passenger/publish** | | 8,83639024 |
| **Average /customer/passenger/show** | | 5,1278 |
| **Average /customer/passenger/update** | | 7,31071667 |
| **General average** | | 5,28100046 |

Where its descriptive statistics are as follows:

|  |  |
| --- | --- |
| *First device* | |
|  |  |
| Mean | 5,16499009 |
| Typical error | 0,39662411 |
| Median | 4,1467 |
| Mode | #N/D |
| Standard deviation | 6,08013246 |
| Sample variance | 36,9680108 |
| Curtosis | 49,2134884 |
| Asymmetry coefficient | 5,97949125 |
| Range | 62,6584 |
| Min | 0,7306 |
| Max | 63,389 |
| Sum | 1213,77267 |
| Count | 235 |
| Confidence level (95,0%) | 0,78141043 |

And therefore, its confidence interval is defined as:

|  |  |  |
| --- | --- | --- |
| Interval(ms) | 4,38357966 | 5,94640052 |
| Interval(s) | 0,00438358 | 0,0059464 |

## 5.2 Second device performance testing

For the second device, the following results were obtained:

|  |  |  |
| --- | --- | --- |
| **Average /** |  | 1,84503986 |
| **Average /anonymous/system/sign-in** | | 3,11999594 |
| **Average /any/system/welcome** | | 1,10291853 |
| **Average /authenticated/system/sign-out** | | 1,95001392 |
| **Average /customer/booking/create** | | 28,7942834 |
| **Average /customer/booking/delete** | | 20,5868202 |
| **Average /customer/booking/list** | | 7,72434152 |
| **Average /customer/booking/publish** | | 16,9787771 |
| **Average /customer/booking/show** | | 6,18608247 |
| **Average /customer/booking/update** | | 26,9499356 |
| **Average /customer/booking-record/create** | | 23,4408545 |
| **Average /customer/passenger/create** | | 7,75793076 |
| **Average /customer/passenger/delete** | | 14,0406796 |
| **Average /customer/passenger/list** | | 3,4431 |
| **Average /customer/passenger/list-menu** | | 5,51371818 |
| **Average /customer/passenger/publish** | | 9,35979037 |
| **Average /customer/passenger/show** | | 5,39151717 |
| **Average /customer/passenger/update** | | 13,0999187 |
| **General average** | | 6,44032988 |

Where its descriptive statistic is as follows:

|  |  |
| --- | --- |
| *Second device* | |
|  |  |
| Mean | 5,15131353 |
| Typical error | 0,41833753 |
| Median | 4,0602 |
| Mode | #N/D |
| Standard deviation | 6,41299283 |
| Sample variance | 41,1264771 |
| Curtosis | 45,9330003 |
| Asymmetry coefficient | 5,88197808 |
| Range | 62,4837 |
| Min | 0,7488 |
| Max | 63,2325 |
| Sum | 1210,55868 |
| Count | 235 |
| Confidence level (95,0%) | 0,8241892 |

And therefore, its confidence interval is defined as:

|  |  |  |
| --- | --- | --- |
| Interval(ms) | 4,32712433 | 5,97550273 |
| Interval(s) | 0,00432712 | 0,0059755 |

## 5.3 Comparison between the two devices

Using the z-test to compare the results obtained on both devices, the following is obtained:

|  |  |  |
| --- | --- | --- |
| z-test for two-sample means | | |
|  |  |  |
|  | *Second* | *First* |
| Mean | 6,56378934 | 5,32456234 |
| Variance (known) | 90,0417729 | 61,2804414 |
| Remarks | 659 | 659 |
| Hypothetical difference of the means | 0 |  |
| z | 2,58608164 |  |
| P(Z<=z) one tail | 0,0048537 |  |
| Critical value of z (one tail) | 1,64485363 |  |
| Critical z value (of colas) | 0,00970739 |  |
| Critical z value (of colas) | 1,95996398 |  |

As can be seen, a value of 0.00970739 has been obtained. Being less than 0.05, you can compare the values of both with each other because the difference in performance between the two is significant. Therefore, the first device, having a lower average than the second, is more powerful according to the data collected during the tests.

# 6. Conclusion

Throughout this report, the functionalities assigned to the buyer role in the system have been exhaustively evaluated, through functional and performance tests. Functional tests have made it possible to verify that the system responds correctly to valid, erroneous and unauthorized scenarios, without detecting critical errors. All the planned validations have worked as expected, thus ensuring the integrity of the system's behavior.

Performance tests have shown significant differences between the devices used. Through statistical analysis and Z-testing, it has been confirmed that the first device has a better average performance than the second, which is relevant for future compatibility and optimization analyses.

Overall, the results obtained demonstrate a high degree of reliability and stability in the functionalities tested. This validation process makes a decisive contribution to ensuring the quality of the software before its final deployment.

# 7. Bibliography

Intentionally blank.