|  |
| --- |
| José Coronil Álvarez (joscoralv@alum.us.es)  Miguel García Vizcaíno (miggarviz@alum.us.es)  Javier Ignacio Milá de la Roca Dos Santos (javmildos@alum.us.es)  Emilio Manuel Vázquez Cruz (emivazcru@alum.us.es)  Group C1.028  10/03/2025 |

|  |
| --- |
| Acme AirNav Solutions, Inc. |
| **Analysis Report** |
| https://github.com/Emilio-115/DP2-Acme-ANS |



**Contents**

[Executive Summary 2](#_Toc190960965)

[Revision Table 2](#_Toc190960966)

[Introduction 3](#_Toc190960967)

[Content 3](#_Toc190960968)

[Conclusions 5](#_Toc190960969)

[Bibliography 5](#_Toc190960970)

# Executive Summary

TODO

Tasks requiring documentation must follow a certain structure, so it was decided a template would be created by me with José Coronil providing images for a professional and consistent image.

Tasks requiring planning were managed using a single GitHub Project, using labels, estimates and other features to create views for each student’s and group tasks and to plan the work, conforming to the teacher’s requirements.

# Revision Table

|  |  |  |
| --- | --- | --- |
| Revision number | Date | Description |
| 1 | 10/03/2025 | Initial version |

# Introduction

TODO

The following document is an analysis report for the second deliverable of the Acme-ANS project. It summarizes the analysis performed on the requirements that needed it.

For this delivery, all requirements needing analysis are grouped into two types based on the problems that need to be solved to meet them: requirements involving documents, and requirements involving planning and task management. Analysis for the solutions to those problems are provided in the same order.

# Content

The mandatory individual requirements for the delivery require creating three entities:

* a flight crew member entity
* a flight assignment entity, representing the assignation of a flight crew member to a leg of a flight, and
* an activity log record entity, representing the log of an incident that occurred during a flight by a flight crew member for a leg they were assigned to once the leg is over.

The first two entities are very clear in their relation to other entities, but the activity log record entity could be accomplished in two ways:

1. store the association to a flight crew member and a leg or
2. store the association to a flight assignment, which itself can be used to get the flight crew member and leg.

A pros and cons analysis on these two options was realized:

* Option a has the advantage of having a shorter and cleaner connection, while option b has through go through an additional connection
* Option b is more rigid and strict. The relation itself enforces the idea that a flight crew member can only leave activity log records for a leg they were assigned to, whereas in option a this would need to be validated.
* Option a is more flexible. For example, let’s assume that due to changing requirements in the future a flight assignment needs to be deleted. If option b is taken, either the activity logs would need to be deleted or they would be untraceable to the relevant crew member and leg.

Ultimately, I decided to go for option b for the following reasons:

* The data model better represents the business domain and rules.
* In case of changing requirements, a migration to option a should be easy.
* I think this migration is unlikely to be necessary in the first place.

# Conclusions

TODO

In conclusion, the document delivery and GitHub planning dashboard have been solved giving priority to cohesion and clarity. A standardized template was created for consistency, and a GitHub project was set up to effectively track and manage tasks. These measures align with project guidelines and enhance overall organization.

# Bibliography

Intentionally blank.