|  |
| --- |
| 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  13/03/2025 |

|  |
| --- |
| 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 4](#_Toc190960969)

[Bibliography 4](#_Toc190960970)

# Executive Summary

An activity log entity, representing a reported incident, must be created. It must be reported by a flight crew member on a leg they were assigned to. This could be accomplished by relating the activity log entity to a flight crew member and a leg directly, or by relating it to a flight assignment which itself has a member and leg.

The second approach was chosen due to enforcing that a flight crew member can only create an activity log for a leg they’ve been assigned to, and was later validated.

# Revision Table

|  |  |  |
| --- | --- | --- |
| Revision number | Date | Description |
| 1 | 10/03/2025 | Content and analysis |
| 2 | 13/03/2025 | Executive summary, introduction, conclusion, and adding the validation of the approach |

# Introduction

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, the only requirement that required analysis was IRQ5, creating the activity log entity. This document focuses on the choice of whether to have a more direct and flexible relation or a more rigid and correct one.

# 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 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 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 more direct relation, while option b has through go through an additional entity to get to the leg.
* Option b is more rigid and strict. The relation itself enforces the idea that a flight crew member can only leave activity logs 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.

This approach was validated on the follow-up session of 11/03/2025.

# Conclusions

In conclusion, the activity log entity will be linked to the flight assignment entity instead of directly to a flight crew member and leg to enforce the requirement that a flight crew member can only record activity logs for a leg they were assigned to.

# Bibliography

Intentionally blank.