April 26, 2024



Analysis report

**Repository: https://github.com/C1-010/Acme-SF-D03**

Group: C1.010

García Rivero, Andrés - [andgarriv@alum.us.es](mailto:andgarriv@alum.us.es)

López Cubiles, Antonio José - [antlopcub@alum.us.es](mailto:antlopcub@alum.us.es)

Nieto Vicioso, Javier - [javnievic@alum.us.es](mailto:javnievic@alum.us.es)

Díaz Ordoñez, Pablo -  [pabdiaord@alum.us.es](mailto:%20pabdiaord@alum.us.es)

Sevillano Barea, Alejandro - [alesevbar@alum.us.es](mailto:alesevbar@alum.us.es)

Content Table

[1. Executive Summary 2](#_Toc165059697)

[2. Revision Table 2](#_Toc165059698)

[3. Introduction 2](#_Toc165059699)

[4. Contents 3](#_Toc165059700)

[4.1 Information requirements 3](#_Toc165059701)

[5. Conclusions 6](#_Toc165059702)

[6. Bibliography 6](#_Toc165059703)

# 1. Executive Summary

This analysis report aims to identify and address issues of inaccuracy, incompleteness, as well as inconsistencies between requirements and incorrect classifications within the analyzed records. Each analysis record will be examined, detailing a verbatim copy of the requirement it references, detailed conclusions from the analysis, and decisions made to rectify any identified issues. It's important to note that not every requirement will be commented on, only those requiring analysis, as they are few.

# 2. Revision Table

|  |  |  |
| --- | --- | --- |
| Revision number | Date | Description |
| 1 | 26/04/2024 | Creation of the report and finished it |

# 3. Introduction

This analysis report serves as a comprehensive examination of information requirements. Its objective is to identify and rectify any discrepancies, such as inaccuracies, incompleteness, and misclassifications, within the analyzed records.

Each record undergoes careful attention, detailing original requirements, analysis conclusions, and decisions for resolution. While not every requirement is subjected to analysis, this report ensures transparency and accountability by documenting pertinent findings.

It delves into Contract Data, analyzing the validation of budget concerning project cost and advocating for the utilization of cost in money for optimal outcomes. Additionally, it addresses Client Operations on Dashboards, advocating for the consideration of only published entities to ensure clarity in data presentation. Furthermore, it evaluates Client Operations on Progress Logs, concluding that progress logs should only be published when the associated contract is also published.

The structure of this document consists of an Executive Summary providing an overview of the report's objectives, followed by a Revision Table detailing the document's revision history. The main body encompasses the analysis of specific information requirements, discussions on the pros and cons of data types, insights from the “On Your Tutorials” forum and conclusions. Finally, the report closes with a concise conclusion summarizing the findings and a bibliography section.

# 4. Contents

## 4.1 Information requirements

**Requirement 2) - Contract Data**

Verbatim copy of the requirement:

“A **contract** is one or several agreements between the stakeholders involved in the development of a **project**. The system must store the following data about them: a **code** (pattern “[A-Z]{1,3}-[0-9]{3}”, not blank, unique), an **instantiation** **moment** (in the past), a **provider name** (not blank, shorter than 76 characters), a **customer name** (not blank, shorter than 76 characters), some **goals** (not blank, shorter than 101 characters), and a **budget** (less than or equal to the corresponding project cost).”

Analysis:

The requirement specifies the data that must be stored about contracts in the system, including a unique code, instantiation moment, provider name, customer name, goals, and a budget. However, a new consideration has arisen regarding the validation of the budget attribute in relation to the cost of the corresponding project.

Two alternatives have been proposed to address this issue:

**Alternative 1**: Establish a relationship between hours and cost for the project to fulfill the requirement without changing previously clarified aspects.

Pros: Meets the client's needs.

Cons: Appears more complex and seems redundant, as the client originally specified that the cost was in hours and the budget was in money.

**Alternative 2**: Change the data type of the "cost" attribute in projects to Money.

Pros: Simplifies implementation.

Cons: Contradicts the client's original specification and may involve unnecessary changes.

Furthermore, implementing the cost as an Integer introduces uncertainty regarding its implementation, particularly concerning how to convert hours into money.

Conclusion:

Considering the complexity and potential contradictions introduced by the alternatives, it is concluded that using the cost in Money is the most appropriate option due to its simplicity and intuitiveness. However, implementation is pending, awaiting confirmation from "student 1," who is responsible for projects.

**Link**: [Contenido (us.es)](https://ev.us.es/ultra/courses/_85092_1/cl/outline) (Cadena: Incoherencia con atributo cost en Projects.)

Imagen que contiene Interfaz de usuario gráfica

Descripción generada automáticamente

**Requirement 4) – Operations by clients on dashboards**

Verbatim copy of the requirement:

“The system must handle **client** dashboards with the following data: total number of **progress logs** with a **completeness** rate below 25%, between 25% and 50%, between 50% and 75%, and above 75%; average, deviation, minimum, and maximum **budget** of the **contracts**.

Analysis:

The requirement specifies that the system must manage client dashboards, providing various data related to progress logs and contract budgets. However, a question arose regarding whether to include only published contracts and progress logs or all regardless of their draft mode. Two alternatives were considered:

**Alternative 1**: Consider only published entities:

Pros: Makes more sense in a real system, as data for a contract or progress log in draft mode may change.

Cons: Minimal difference in implementation.

**Alternative 2**: Consider all entities, including those in draft mode:

Pros: Simplifies implementation, as it aligns with the current system setup.

Cons: Makes less sense in a real system, as draft entities are subject to change.

After consulting the "On Your Tutorials" forum, where this issue had already been discussed, it was concluded that only published entities should be considered in the statistics displayed on the client dashboards. This decision was confirmed by the client, emphasizing that including draft entities could lead to confusing data interpretation due to their susceptibility to changes.

Therefore, it is determined that the system should only represent statistics based on published contracts and progress logs on the client dashboards to ensure clarity and accuracy in data presentation.

Link: [Contenido (us.es)](https://ev.us.es/ultra/courses/_85092_1/cl/outline) (Cadena: [Analisis] D02-Student#2-008

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

**Requirement 7) – Operations by clients on progress logs**

Verbatim copy of the requirement:

“Operations by **clients** on **progress logs**:

* List the **progress logs** in their **contracts**.
* Show the details of their **progress logs**.
* Create and publish a **progress log**.
* Update or delete a **progress log** as long as it is not published.”

Analysis:

The requirement outlines various operations that clients can perform on progress logs associated with their contracts. However, a question arose regarding the publishing mechanism for progress logs, particularly in relation to the status of the associated contract. Two alternatives were considered:

**Alternative 1**: Automatically publish progress logs when the contract is published:

Pros: Simplifies the publishing process for progress logs, ensuring they are readily available once the associated contract is published.

Cons: Contrary to the client's requirement, as this approach would prevent the publishing of progress logs for contracts that have already been published, which goes against the client's expectations and workflow.

**Alternative 2**: Allow publishing of progress logs only when the contract is published:

Pros: Ensures that progress logs are only published when the associated contract is published, reducing the risk of inconsistencies.

Cons: Needs more effort in the implementation.

After consulting with the client during the Follow Up session on April 24, 2024, it was clarified that progress logs should only be published when the associated contract is published. Therefore, the conclusion reached is that progress logs can be published when the contract is published, but if the contract is not published, the progress logs should not be published. However, clients should still be able to create, update, and delete progress logs, even if there contracts are not published.

Therefore, the system should allow clients to perform operations on progress logs as outlined in the requirement, with the restriction that progress logs can only be published if the associated contract is also published.

# 5. Conclusions

In conclusion, this analysis report has systematically addressed issues of inaccuracy, incompleteness, and inconsistencies within the analyzed records, striving for transparency and accountability in resolving discrepancies. Through meticulous examination, it has provided insights into Contract Data, advocating for the utilization of cost in money for optimal outcomes. Additionally, it has shed light on Client Operations on Dashboards, emphasizing the importance of considering only published entities for clarity in data presentation. Moreover, it has evaluated Client Operations on Progress Logs, concluding that progress logs should only be published when the associated contract is also published. This report serves as a comprehensive guide for ensuring data integrity and effective decision-making within the analyzed system.

# 6. Bibliography

Intentionally blank.