

Group: C1.010

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

February 21th, 2024

Analysis Report

**Repository: https://github.com/C1-010/Acme-SF-D02**

Contenido

[1. Executive Summary 2](#_Toc158936944)

[2. Revision Table 2](#_Toc158936945)

[3. Introduction 2](#_Toc158936946)

[4. Contents 2](#_Toc158936949)

4.1. Information requirements……………………………………………………………………………………..2

4.2. Functional requirements……………………………………………………………………………………….2

4.3. Non-functional requirements………………………………………………………………………………..2

4.4. Testing requirements……………………………………………………………………………………..……..2

4.5. Managerial requirements……………………………………………………………………………….……. 3

[5. Conclusions 3](#_Toc158936950)

[6. Bibliography 3](#_Toc158936951)

# 1. Executive Summary

In this document I will analyze the requirements from this deliverable that require further explanation. I will showcase the decisions I made and any inconvenience.

# 2. Revision Table

|  |  |  |
| --- | --- | --- |
| Revision number | Date | Description |
| 1 | 21/02/2024 | Initial Version |
| 2 | 04/03/2024 | New doubts |
| 3 | 08/03/2024 | Final Revision |

# 3. Introduction

The objective of this document is to analyze the tasks in this deliverable and their requirements. When I find multiple ways to develop a task, I will showcase then and choose one, explaining why or how I made my choice.

# 4. Contents

**4.1. Information requirements**

**Task-002**

**Description:** A training module consists of one or several short-term training activities aimed at extending or updating knowledge and skills related to the topic 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), a creation moment (in the past), some details describing the training module (not blank, shorter than 101 characters), a difficulty level (“Basic”, “Intermediate”, or “Advanced”), an optional update moment (in the past, after the creation moment), an optional link with further information, and an estimated total time.

**Analysis:** This task didn’t look too complicated. Using the theory lessons and the Acme-Jobs starter as a guide, I could create the entity Training Module. The only doubt I had was how to make a constraint that forced the update moment to be after the creation moment. I asked the professor in class, and learned that this type of constraint is made in the services.

**Task-003**

**Description:** Each training module is made up of training sessions. The system must store the following data about them: a code (pattern “TS-[A-Z]{1,3}-[0-9]{3}”, not blank, unique), a time period (at least one week ahead the training module creation moment, at least one week long), a location (not blank, shorter than 76 characters), an instructor (not blank, shorter than 76 characters), a mandatory contact email, and an optional link with further information.

**Analysis:** Just like before, the implementation of the class and its relations wasn’t too hard. However, the implementation of the period attribute got me stuck, and I couldn’t find answers in the forum. I asked the professor to answer my doubts, and in conclusion, I must use an attribute of start of the period and another for the end.

**Task-004**

**Description:** The system must handle developer dashboards with the following data: total number of training modules with an update moment; total number of training sessions with a link; average, deviation, minimum, and maximum time of the training modules.

**Analysis:** This requirement was pretty straightforward and doesn’t need and extensive analysis, as now we now it is implemented in the services.

**Task-013**

**Description:** There is a new project-specific role called developer, which has the following profile data: degree (not blank, shorter than 76 characters), a specialisation (not blank, shorter than 101 characters), list of skills (not blank, shorter than 101 characters), an email, and an optional link with further information.

**Analysis:** This requirement was pretty straightforward and doesn’t need and extensive analysis.

**4.2.** **Functional requirements**

Intentionally blank.

**4.3. Non-functional requirements**

Intentionally blank.

**4.4. Testing requirements**

**Task-004**

**Description:** Produce assorted sample data to test your application informally. The data must include two developer accounts with credentials “developer1/developer1” and “developer2/developer2”

**Analysis:** This requirement is tricky, because it seems that I only need to create the data for those two developers, but I must create data for Training Sessions and Training Modules as well.

**4.5. Managerial requirements**

**Task-014**

**Description:** Produce a UML domain model

**Analysis:** This task was very straightforward, no doubts here.

For the document elaboration I just stick to the instructions from the document “08 Annexes.docx” provided by the lecturers, so there’s no need of further analysis.

# 5. Conclusions

The analysis made for this delivery has been more complex than the previous one, as we begin to dive into more complex programming. However, I feel like I did a good job and confident that the next deliverable will be as smooth as this one, with no panic attacks.

# 6. Bibliography

"08 Annexes.docx" - Material provided in the DP2 subject at the University of Seville.Principio del formulario