**INDIVIDUAL ANALYSIS REPORT**

Group C1.04.05

05/03/2023

Author(s):

Fernando José Mateos Gómez,[*fermatgom@alum.us.es*](mailto:fermatgom@alum.us.es)

**Table of contents**

[**Table of contents**](#_heading=h.30j0zll) **2**

[**1. Executive summary**](#_heading=h.1fob9te) **3**

[**2. Revision table**](#_heading=h.3znysh7) **3**

[**3. Introduction**](#_heading=h.2et92p0) **3**

[**4. Contents**](#_heading=h.tyjcwt) **3**

[4.1.](#_heading=h.3dy6vkm) Requirements 3

[4.2.](#_heading=h.1t3h5sf) Diagrams 4

[**5. Conclusions**](#_heading=h.4d34og8) **4**

[**6. Bibliography**](#_heading=h.2s8eyo1) **4**

1. **Executive summary**

This report summarizes the analysis needed for the first deliverable for Fernando Mateos Gómez’ work.

1. **Revision table**

| **No** | **Date** | **Description** |
| --- | --- | --- |
| 1.0 | 05/03/2023 | Initial draft of the document. |

1. **Introduction**

The analysis of the requirements is fundamental in every project to get the necessities of the customer right and understand the domain with precision. The documents provided to us already give a detailed expression of what the requirements might be and what they need to be completely fulfilled. If any of those requirements implores extra analysis it will be documented on these analysis reports throughout the different deliverables of the project.

1. **Contents**

For this deliverable, an analysis and modeling of the requirements and data to save at the database.

**4.1. Requirements**

In this section I will explain my decisions on the requirements I had to make a decision, it’s pros and cons, only if they proceed.

**Requirement 5:**  “*An audit is a document with auditing records regarding a published course. The system must store the following data about them: a code (pattern “[A-Z]{1,3}[0-9][0-9]{3}”, not blank, unique), a conclusion (not blank, shorter than 101 characters), some strong points (not blank, shorter than 101 characters), some weak points (not blank, shorter than 101 characters), and a mark (computed as the mode of the marks in the corresponding auditing records; ties must be broken arbitrarily if necessary).*”

**Decisions:** For this requirement I had to make 3 different decisions. First of all, about the “code” of each audit, i didn’t find any cons for simplifying the regular expression to /*[A-Z]{1,3}\d{4}*/*,* not only it makes it cleaner omitting the extra number in the middle, but it also does the same.

On the other hand, it also asks about saving “strong” and “weak” points. Among creating two more Entities, and saving this fields as a string with some kind of separator, the best option is to save it as a string( simply because we do not need to save any other kind of information about this field )

For the delimiter of these fields, “\r\n” or “;” were being thought of, and ended up using the semicolon character.

**Pros:** It is a single character instead of 2 and when enumerating points it should not be common

**Cons:** If the user uses it while describing a point, it will break it in two, while using an “end of line” character would avoid it, by simply writing another line with the new point.

*Check Forum “[D&T] Calculus of Marks”.*

Another point to analyze in this requirement is the calculus of the marks for each audit.

There were, initially 2 options ( calculate it using a numerical conversion to our system or using a conversion of 0 to 6 and calculate the mean )

These options require an extra amount of calculus, and were not thought of with queries to the database, they had two cons; slowness and problems to select randomly a mark.

With queries in mind, these options are discarded, and this feature is calculated using a query to the database in the near future.

**Requirement 7:**  “*The system must handle auditor dashboards with the following data: total number of audits that they have written for theory and hand-on courses; average, deviation, minimum, and maximum number of auditing records in their audits; average, deviation, minimum, and maximum time of the period lengths in their auditing records*”.

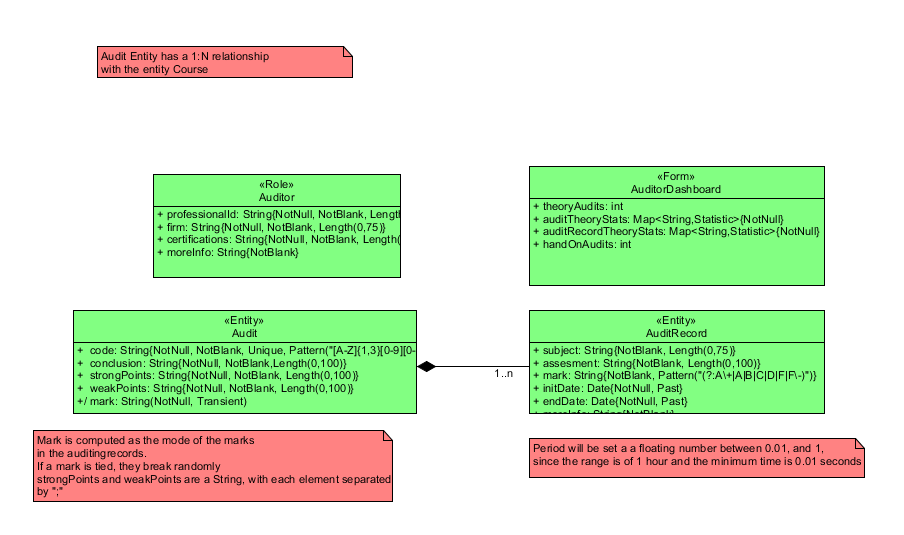
**Decisions**: Since all the requirements require some kind of statistical calculus, a decision to create a module Statistic as a “helper” is made.

**Pros:** We can unify all the behavior and calls to the services in one place, without having this functionality separated across the application.

**Cons:** No cons have been found.

This Module will not be a Static class but an Object that only will be used in the dashboards. *Check Forum “[D&T] Auxiliar Class”.*

**4.2. Diagrams**



Diagrams stored at “*~/Student #5/D2/UML Individual Fernando.uxf*”

1. **Conclusions**

<Intentionally blank>

1. **Bibliography**

<Intentionally blank>