

Informe de análisis adrgarcha

Group: C1.04.01

Repository: <https://github.com/C10401-DP2/Acme-L3.git>

Student #1

Name: García Aguilar, Javier
Email: javgaragu1@alum.us.es

Student #2

Name: Cuenca Pérez, Pablo
Email: pabcueper@alum.us.es

Student #3

Name: del Hoyo Carcaboso, David
Email: davdelcar@alum.us.es

Student #4

Name: Zarandieta Ortiz, David
Email: davzarort@alum.us.es

Student #5

Name: García Chavero, Adrián
Email: adrgarcha@alum.us.es

Date: Seville February 16, 2023

Contenido

Resumen ejecutivo	3
REGISTRO DE CAMBIOS	4
Introducción	5
Contenido	6
Conclusión	8
Bibliografía	9

Resumen ejecutivo

En este informe, explicaremos como se han llevado a cabo cada una de las tareas que se me han asignado en el proyecto, incluyendo un requisito grupal y todos los requisitos individuales, junto con las decisiones de análisis que se han tenido que llevar a cabo para que el proyecto funcione correctamente.

REGISTRO DE CAMBIOS

Versión	Causa del Cambio	Responsable del Cambio	Fecha del Cambio
1.0	Versión inicial	Adrián García Chavero	17/03/2023

Introducción

El informe tratará sobre las distintas decisiones de análisis que se han llevado a cabo durante el desarrollo del proyecto para cada uno de los requisitos asignados, así como una explicación de como se ha trabajado en cada una de las tareas.

Para comenzar, se explicarán cada una de las tareas asignadas con una descripción detallada del requisito al que se refiere, y las conclusiones y decisiones de análisis que se han llevado a cabo en la tarea si ha sido necesario.

Para finalizar, en la conclusión se comentará si el trabajo realizado a sido satisfactorio y la dificultad implicada al desarrollar los requisitos.

Contenido

Primeramente, explicaremos la tarea grupal que me asignó el *manager* del equipo, en este caso fue crear la entidad *Peep* y agregarla al modelo de dominio UML:

- **Entidad Peep:** En este caso, no hubo ninguna decisión de análisis relevante, ya que no poseía atributos derivados ni relaciones a tener en cuenta a la hora de implementarlos en el futuro. Por lo tanto, podemos concluir que fue una tarea sencilla, que tampoco supuso ningún problema de dependencias para el resto del proyecto. El requisito en concreto sería el siguiente:

10) **[MANDATORY]** A peep is a message posted by anyone. The system must store the following data about them: an instantiation moment (in the past), a title (not blank, shorter than 76 characters), a nick (not blank, shorter than 76 characters), a message (not blank, shorter than 101 characters), an optional email address, and an optional link.

A continuación, hablaremos sobre cada uno de los requisitos individuales que posee el *Estudiante 5*, los cuales fueron todos desarrollados al completo. Las tareas eran crear un nuevo rol *auditor* (*Role*); crear tres clases llamadas *Audit* (*Entity*), *AuditingRecord* (*Entity*), y *AuditorDashboard* (*Form*); producir datos de ejemplo para cada una de las anteriores; por último, era necesario agregarlos al diagrama UML junto con los demás objetos de mi equipo:

- **Rol Auditor:** En este caso, tampoco hubo ninguna decisión de análisis importante, ya que no tuvimos que crear ninguna propiedad derivada, ni ninguna relación; aunque si esta relacionada con *Audit*, su relación se encuentra en esa entidad. El requisito sería el siguiente:

4) There is a new project-specific role called auditor, which has the following profile data: firm (not blank, shorter than 76 characters), professional ID (not blank, shorter than 26 characters), a list of certifications (not blank, shorter than 101 characters), and an optional link with further information.

- **Entidad Audit:** En esta entidad, si existe una decisión de análisis a tener en cuenta, la cual es que la propiedad *mark*, se trata de un atributo derivado que requiere de el atributo *mark* de la entidad *AuditingRecord*. Por lo tanto, llegamos a la conclusión que lo mejor sería implementarlo en un futuro en un servicio de *Audit*. En cuanto a sus relaciones, tiene una relación *ManyToOne* con *Auditor*, y un atributo *draftMode* ya que se trata de una composición junto con *AuditingRecord*. El requisito es el siguiente:

5) **[MANDATORY]** 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).

- **Entidad AuditingRecord:** El único punto a destacar sería que el periodo en el que la asignatura fue auditada se trata de un atributo derivado de dos nuevos atributos que hemos agregado, los cuales son *initialDate* y *finalDate*, para calcular el periodo correctamente. Así que decidimos implementar este atributo derivado en un servicio de *AuditingRecord* en el futuro. Por otro lado, tiene una relación de composición *ManyToOne* con *Audit*, y una relación *ManyToOne* con la entidad *Course*. El requisito es el siguiente:

- 6) **[MANDATORY]** The system must store the following data about the auditing records of a course: a subject (not blank, shorter than 76 characters), an assessment (not blank, shorter than 101 characters), the period during which the subject was audited (in the past, at least one hour long), a mark ("A+", "A", "B", "C", "F", or "F-"), and an optional link with further information.
- **Form AuditorDashboard:** En este caso, no se realizó ninguna decisión de diseño relevante para el proyecto. El requisito es el siguiente:
- 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.
- **Producir datos de ejemplo:** En concreto, no hubo ninguna decisión de análisis importante. Sin embargo, habría que señalar que surgieron varias dificultades a la hora de popular la base de datos, ya que entraba en conflicto con los demás datos de ejemplo del equipo; aunque afortunadamente fueron solucionados a tiempo. El requisito al que se refiere es el siguiente:
- 8) **[MANDATORY]** Produce assorted sample data to test your application informally. The data must include two auditor accounts with credentials "auditor1/auditor1" and "auditor2/auditor2".

Conclusión

En conclusión, podemos destacar que el trabajo realizado ha sido satisfactorio, ya que se han realizado varias comprobaciones y pruebas al respecto para que todo funcionase correctamente. Y pienso que las decisiones y conclusiones de diseño tomadas han sido las correctas.

Además, podemos concluir que los requisitos han tenido una dificultad adecuada para el nivel en el que nos encontramos. Sin embargo, han surgido conflictos inesperados y, a pesar de ello, he podido seguir a adelante y resolver dichos conflictos para que el proyecto funcione correctamente.

Bibliografía

Intencionalmente en blanco.