

Analysis Report - Student #2

Group: L3-D01

Repository: <https://github.com/adrrf/acme-life-long-learning>

Student #1:

UVUS: manvazmar3

Name: Vázquez Martín, Manuel

Roles: Developer, Tester

Email: manvazmar3@alum.us.es

Student #2:

UVUS: angmunpri

Name: Muñoz Prieto, Ángel

Roles: Developer, Analyst

Email: angmunpri@alum.us.es

Student #3:

UVUS: adrromflo

Name: Romero Flores, Adrián

Roles: Developer, Manager

Email: adrromflo@alum.us.es

Student #4:

UVUS: jorromlim

Name: Romero Limón, Jorge

Roles: Developer, Operator

Email: jorromlim@alum.us.es

Student #5:

UVUS: alematcap

Name: Mateo Capilla, Alejandro

Roles: Developer, Tester

Email: alematcap@alum.us.es

Date: Sevilla a 25 de Mayo de 2023

Introduccion	4
Contenido	5
Entregable 1	5
Entregable 2	5
Entregable 3	7
Entregable 4	10

Introduccion

En este documento se recogerán las distintas decisiones de análisis tomadas por el alumno Ángel Muñoz Prieto, acerca de los requisitos individuales a implementar o diseñar a lo largo de los 4 entregables del proyecto.

Se evaluarán requisitos relacionados con el rol de estudiante y las entidades de 'Enrolment' y 'Activity' en la aplicación. Dichas decisiones se tomarán en base a un problema inicial y el estudio de distintas alternativas posibles para el problema.

Contenido

Entregable 1

En el primer entregable de la asignatura no se necesitó de ninguna tarea de análisis, debido a que por la naturaleza de estas tareas estas no requieren de ello. Por ello se ha tomado la decisión de no elaborar un documento para este entregable.

Entregable 2

En este entregable se realizó para el modelo tecnológico del proyecto la creación de las entidades “Enrolment” y “Activity”, así como la del rol “Student”, acerca de las cuales se han tomado las siguientes decisiones:

- 1) *“There is a new project-specific role called student, which has the following profile data: statement (not blank, shorter than 76 characters), list of strong features (not blank, shorter than 101 characters), list of weak features (not blank, shorter than 101 characters), and an optional link with further information.”*

Tras estudiar este requisito no se ha detectado en él ninguna ambigüedad, incompletitud, error de clasificación o algún requisito implícito, luego no se ha requerido de decisiones de análisis.

- 2) *“An enrolment is a registration of a student in a 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 motivation (not blank, shorter than 76 characters), some goals (not blank, shorter than 101 characters), and a work time (in hours, computed from the corresponding activities).”*

Este requisito presenta una cierta ambigüedad respecto al atributo ‘work time’, ya que pide solo horas, luego no deja explícito si se debe usar un tipo de tiempo con horas, minutos y segundos o un Integer para contar las mismas.

-Alternativa 1: Emplear un tipo Date a la que se derivaría la suma del tiempo del resto de actividades.

-Alternativa 2: Emplear un tipo Integer que resultaría de la suma de las horas de las actividades.

Tras una tarea de análisis se decidió emplear un Integer ya que aportaba la misma información, un nº de horas y presentaba una mayor sencillez respecto al código.

- 3) *“Every enrolment has a workbook that is composed of activities. The system must store the following data about them: a title (not blank, shorter than 76 characters), an abstract (not blank, shorter than 101 characters), an indication on whether it can be considered a theory activity or a hands-on activity, a time period (either in the past or the future), and an optional link with further information.”*

Este requisito solicita que un campo de la actividad se llame ‘abstract’ sin embargo dicho campo está reservado luego tras un estudio sobre cómo proceder y una consulta al profesor acerca de si dicha solución era válida, se decidió usar otro nombre a modo de sinónimo, usando ‘recap’ ya que tanto este como ‘abstract’ consisten en un breve resumen.

Además de lo anterior, se plantea en este requisito la decisión de cómo representar la relación entre “Enrolment” y “Activity”, presentándose las siguientes alternativas:

-Alternativa 1: Crear una entidad “WorkBook”, que actúa de punto intermedio entre las otras dos.

-Alternativa 2: Establecer una composición entre “Enrolment” y “Activity” sin necesidad de una entidad intermedia para esto.

Tras un estudio y una consulta al profesor sobre qué alternativa era más correcta para este requisito, se decidió tomar la segunda y realizar una composición, eliminando así la necesidad de una entidad más.

- 4) *“The system must handle student dashboards with the following data: total number of theory and hands-on activities in his or her workbook; average, deviation, minimum, and maximum period of the activities in his or her workbook; average, deviation, minimum, and maximum learning time of the courses in which he or she has enrolled”*

Para este requisito se plantearon las siguientes alternativas:

-Alternativa 1: Crear una propiedad individual Double por cada elemento del Dashboard.

-Alternativa 2: Crear un mapa como única propiedad de tipo <String, Double> de manera que todas las estadísticas se contengan en un único lugar.

Tras una tarea de análisis bastante sencilla y el visto bueno del profesor se tomó la segunda alternativa, ya que esta no sólo presenta una mayor simplicidad para el programador, sino también para el usuario de la aplicación ya que se manejan mejor estos datos..

- 5) *“Produce assorted sample data to test your application informally. The data must include two student accounts with credentials “student1/student1” and “student2/student2”.”*

Tras estudiar este requisito no se ha detectado en él ninguna ambigüedad, incompletitud, error de clasificación o algún requisito implícito, luego no se ha requerido de decisiones de análisis.

Entregable 3

En este entregable se implementaron en la aplicación de Acme las funcionalidades creadas en el anterior entregable, con funciones básicas cómo listar, crear, o borrar:

1) *“Operations by anonymous principals on user accounts:*

- *Sign up to the system and become a student.”*

Tras estudiar este requisito no se ha detectado en él ninguna ambigüedad, incompletitud, error de clasificación o algún requisito implícito, luego no se ha requerido de decisiones de análisis.

Dicha tarea se ha resuelto añadiendo el botón y formulario pertinente en el apartado de información de usuario.

2) *“Operations by students on user accounts:*

- *Update their profiles”*

Tras estudiar este requisito no se ha detectado en él ninguna ambigüedad, incompletitud, error de clasificación o algún requisito implícito, luego no se ha requerido de decisiones de análisis.

Dicha tarea se ha resuelto añadiendo el botón y formulario pertinente en el apartado de información de usuario.

3) *“Operations by any students on courses:*

- *List the courses in the system that are published.*
- *Show the details of the courses in the system, including their lectures and lecturers.”*

En este requisito pese a que aparentemente no hay problemas respecto a la parte de representar los cursos publicados, se pueden representar de distintas formas sus profesores y asignaturas:

-Alternativa 1: Mostrar información de los profesores y las lectures en la pantalla del curso.

-Alternativa 2: Mostrar la Id del profesor e información propia de la asignatura.

-Alternativa 3: Mostrar la Id del profesor y un botón con un listado de las asignaturas y sus detalles.

Tras una tarea de análisis se ha tomado finalmente la última decisión, ya que el profesor no contiene información distintiva aparte de su Id y las asignaturas pueden sobrecargar de información la página por tanto lo óptimo es crear un listado a parte.

4) *“Operations by students on enrolments:*

- List their own enrolments.

- Show the details of their enrolments.

- Register an enrolment.

- Update or delete an enrolment as long as it is not finalised.

- Finalise an enrolment. This requires providing a valid credit card; the system will only store its holder and its lower nibble”

Tanto el listado como los detalles de cada “Enrolment” no requieren de ninguna decisión de análisis. Estos se han implementado a partir del menú de estudiante de la aplicación.

Para la tarea de creación se planteaban las siguientes 2 alternativas:

-Alternativa 1: Crear un botón en el listado de “Enrolment”

-Alternativa 2: Crear dicho botón en el listado de Cursos.

Tras una tarea de análisis y ya que los “Enrolment” deben estar unidos siempre a un curso se optó por la segunda opción, que elimina el paso posterior de asociar el objeto creado con el curso pertinente.

Luego las tareas de actualización y borrado también carecen de tareas de análisis ya que son soluciones simples como crear el botón correcto en el listado. Es pertinente añadir que debido a la indicación de la tarea, es decir, que solo persista el dueño de la tarjeta y los 4 últimos números, el botón de actualizar no registra los cambios en el número de tarjeta, CVV o fecha de expiración, ya que estos no persisten.

Por último, la tarea de finalizar, al llevar toda la lógica de la tarjeta de crédito necesito de un cierto análisis y se estudiaron las siguientes alternativas:

-Alternativa 1: Una nueva pantalla con un nuevo formulario para estos datos

-Alternativa 2: Crear un segundo formulario que responda a un botón de “finalise” en la misma pantalla.

Finalmente se optó por la segunda opción ya que este presentaba una mayor simplicidad a la hora de crearlo, y el profesor dio su visto bueno en clase.

Tras esto llega una segunda tarea de análisis, el cómo se procesan los datos del pago; apareciendo las siguientes 2 alternativas:

-Alternativa 1: Creación de una nueva entidad para la tarjeta de crédito.

-Alternativa 2: Ampliar la entidad Enrolment

Ya que sólo se deben persistir dos campos nuevos según el requisito, tras una tarea de análisis se escogió el ampliar con estas dos propiedades la entidad "Enrolment", ya que crear una nueva entidad para sólo dos campos implicaba mayor carga a la aplicación para tan sólo esos dos datos.

5) *"Operations by students on workbooks:*

- List the activities in their workbooks.*
- Show the details of their workbooks.*
- Create a new activity in their workbooks, as long as the corresponding enrolment is finalised.*
- Update or delete the activities in their workbooks, as long as the corresponding enrolment is finalised."*

De nuevo al igual que en el anterior requisito no se observan decisiones de análisis en el listado y los detalles de las actividades. Dichas funcionalidades se han implementado mediante un botón "WorkBook" en los detalles del "Enrolment" que las contiene.

En cuanto a la creación, actualizado y borrado de actividades, pese a que el documento indica que es necesario que el "Enrolment" esté finalizado, esto es algo que no tiene sentido ya que para finalizar este hay que realizar el pago del curso al que el estudiante se inscribe, luego sería necesario que dicho estudiante tenga unas ciertas asignaturas añadidas para pagar este curso al que se matricula. Tras consultar esta decisión al profesor en clase finalmente se implementó de manera que se puedan crear, actualizar y borrar actividades mientras que el "Enrolment" no esté finalizado, y una vez este se pague y se cierre, dichas actividades se consideran matriculadas y por tanto, inalterables.

6) *"Operations by students on student dashboards:*

- Show their student dashboards."*

Este requisito no se ha podido implementar por parte del estudiante, por tanto no se ha tomado ninguna decisión de análisis.

Entregable 4

En este último entregable, debido a que no hay que implementar ni diseñar nada, no ha visto el estudiante necesario realizar ninguna tarea de análisis para ello, luego no será necesario un documento de análisis para este entregable..