Universidad de Sevilla  
Escuela Técnica Superior de Ingeniería Informática

ANALYSIS REPORT INDIVIDUAL - D03



**C1.04.02**

**Repositorio**: [https://github.com/josrojrom1/C1.04.02](https://github.com/josrojrom1/C1.04.02%C3%A7)

**Autores:**

Parra Méndez, Pablo, pabparmen@alum.us.es

**Date:** 25/05/2023

**Tabla de contenidos**

[**Resumen**](#_heading=h.gjdgxs) **2**

[**Tabla de revisión**](#_heading=h.30j0zll) **2**

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

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

[**Bibliografía**](#_heading=h.1t3h5sf) **5**

# Resumen

Este documento se trata de un listado de registros de análisis, cada uno de los cuales deberá incluir los siguientes datos: copia textual del requerimiento a que se refiere el registro; conclusiones detalladas del análisis y decisiones tomadas para enmendar el requerimiento.

# Tabla de revisión

|  |  |  |
| --- | --- | --- |
| **Versión** | **Fecha** | **Descripción** |
| V1 | 21/04/2023 | Creación del documento |
| V2 | 25/05/2023 | Refinar documento |

# 

# Introducción

En este documento se va a llevar a cabo el análisis de los requisitos implementados individualmente por el miembro Pablo Parra Méndez.

# Contenidos

* **Operations by anonymous principals on user accounts**:

-Sign up to the system and become an student.

**-Conclusiones de la tarea**: Con esta tarea me di cuenta que al crear los usuarios en el user-config.csv los puse en false, por tanto no iniciaba la sesión. Fue una tarea sencilla mirando el proyecto y gracias a mis compañeros

**-Decisiones durante la tarea**: Consideré que no era necesario crear un showService, que se entraría directamente para actualizar.

* **Operations by assistants on user accounts**:

-Update their profiles.

**-Conclusiones de la tarea**: Continuación de la primera tarea.

**-Decisiones durante la tarea**: Nada importante. Similar a la anterior

* **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.

**-Conclusiones de la tarea**: En esta tarea aprendí a listar y mostrar los cursos, mostrando los datos de las lecturas y lectures.

**-Decisiones durante la tarea**: Según he entendido la tarea y siguiendo los requisitos, al no especificar exactamente una restricción de los cursos que muestra, he decidido que enseñe al estudiante una lista de todos los cursos publicados. Ya que luego él posteriormente al crear una matrícula, puede elegir entre todos los cursos publicados, por lo que entendí que sería de su interés ver una lista de los cursos disponibles logueado desde su rol. Por otro lado, la vista de un curso muestra el nombre de usuario del lecturer y las lecturas.

* **[MANDATORY] 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.

**-Conclusiones de la tarea**: En esta tarea he tenido muchas confusiones en cuanto a finalizar un enrolment, y luego mostrar el workTime de los enrolments. Gracias a esta tarea he aprendido a como pasar un enrolment a un estado finalizado mediante una validación de tarjeta de crédito. Por otro lado, aprendí el uso de un atributo derivado como es workTime

**-Decisiones durante la tarea**:Finalmente decidí introducir en la entidad el atributo boolean IsFinalised que me indicaba que si estaba finalizado lo pasaba a true, junto a los atributos de creditCardHolder y lowerNibble que son los requeridos en la base de datos. El resto de los atributos que exige una validación de una tarjeta de crédito los pide en un form(definidos en el FinaliseService) que aparece cuando queremos finalizar el enrolment, y si al rellenarlo son correctos permite finalizarlo, pero sin guardados en base de datos.

Un problema que ocurre con estos atributos que no se guardan en base de datos, es que, al no ser definidos en la entidad, las restricciones básicas del Framework como dejar un String en un Integer, en este caso da error. Por lo que tuve que convertir todos estos inputs a textbox, y los atributos definidos en el service a String. Luego cree las funciones boolean en el service canConvertToInt y canConvertToDate, que comprobaban si la cadena String introducida sigue el formato tanto de Integer como Date, si no se cumplen, salta una restricción en cada caso. Especialmente en el caso de la función canConvertToDate, tiene integrada la restricción de solo admitir fechas posteriores al momento base, para no introducir tarjetas caducadas.

Por otro lado, el atributo workTime es un atributo derivado que va en función de las actividades que existen en una matrícula y que se va actualizando constantemente según se vayan introduciendo, por lo que decidí que almacenarlo en la base de datos podría suponer un problema. Este atributo es de readOnly y no puede ser modificado en ningún momento.

* **[MANDATORY] Operations by students on workbook:**

-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.

**-Conclusiones de la tarea**: En esta tarea comprendía realizar operaciones en la clase activity, teniendo en cuenta las restricciones heredadas de enrolment.

**-Decisiones durante la tarea**: La principal decisión tomada es la posibilidad de crear una actividad desde el listado de las mismas, y que tuvieras la posibilidad con un select de elegir el enrolment que quieras para esa actividad, siempre y cuando el enrolment esté finalizado. En el update también puedes cambiar el enrolment, pero al igual que cuando creas una actividad en el select solo aparecerán los enrolment que han sido finalizados.

* **Operations by students on student dashboards:**

**-**Show their student dashboards.

**-Conclusiones de la tarea**: En esta tarea comprendí la realización de un dashboard que mostrará todas las estadísticas relacionadas a los enrolments y actividades, como medias,desviaciones,tiempos máximos y mínimos.

**-Decisiones durante la tarea**: La única decisión relevante tomada ya que los atributos a mostrar en el dashboard estaban previamente definidos en el entregable D02, es simplemente cómo calcular el tiempo total y sus medias entre el startTimePeriod y endTimePeriod de las actividades. Finalmente creé una query usando el TIME\_TO\_SEC que calculaba el tiempo establecido entre los dos puntos.

# Bibliografía

-Documento 08 Annexes, de los contenidos de la plataforma virtual de esta asignatura.

-Todos los documentos de la lesson L03 de la plataforma virtual de esta asignatura.