|  |  |
| --- | --- |
| Logotipo  Descripción generada automáticamente | ***C.F.G.S. DESARROLLO DE APLICACIONES MULTIPLATAFORMA*** |
| **CURSO 2024/2025**  ***Módulos integrados en el MiniReto:***   * ***Acceso a Datos (AD)*** * ***Desarrollo de Interfaces (DI)*** | |
| ***Duración: 5 días lectivos***  ***15 horas totales*** | ***Organización: Grupos de 5 o 6 alumnos*** |

|  |
| --- |
| ***MiniReto***  ***ProyecStore - Gestión de repositorio de Proyectos fin de ciclo*** |
| ***Desarrollar una aplicación que permita gestionar y almacenar los documentos y la información de los Proyectos Fin de Ciclo y futuros Proyectos intermodulares de los alumnos, incluyendo documentación, fotos, software asociado, información de los alumnos, calificaciones y composición de los tribunales de evaluación.*** |

|  |
| --- |
| **Objetivos específicos del reto** |

**Objetivos de resultados**

1. Realizar una API Rest Spring con uso de librería JdbcTemplate para acceso a una BD MySQL de Gestión de Proyectos y su consumo por aplicaciones clientes
2. Realizar una aplicación de escritorio en C# (utilizando WinForms) con buena usabilidad que permita la gestión de un repositorio de proyectos y de la información relacionada con la presentación de estos. La aplicación consumirá una API de acceso a una BD de Gestión de Proyectos.
3. Coordinar el trabajo de desarrollo de la API y de la aplicación de escritorio en función de los requisitos actuales y futuros de esa aplicación de escritorio.

**Objetivos del proceso**

1. Fomentar el trabajo en equipo y la colaboración entre los estudiantes.
2. Desarrollar habilidades de comunicación y negociación para lograr acuerdos y tomar decisiones en conjunto.
3. Potenciar la capacidad de resolución de problemas mediante la aplicación de conocimientos y técnicas de diferentes módulos.
4. Desarrollar habilidades de planificación y organización del proyecto, desde la definición de objetivos y fases hasta la asignación de tareas y seguimiento del progreso.
5. Fomentar la creatividad y la innovación en la búsqueda de soluciones originales y eficaces para el problema planteado.
6. Potenciar la capacidad crítica y reflexiva para evaluar el trabajo realizado y aprender de los aciertos y errores.
7. Fomentar el compromiso y la responsabilidad en el cumplimiento de las metas y plazos establecidos en el proyecto.
8. Desarrollar habilidades de presentación y defensa del proyecto ante un público especializado o no especializado.
9. Promover el respeto y la valoración de la diversidad de enfoques y perspectivas que pueden aportar las diferentes disciplinas involucradas en el proyecto.

|  |
| --- |
| **El Reto. Descripción** |

En este reto, debéis desarrollar una aplicación de escritorio que permita gestionar y almacenar los documentos y la información de los **Proyectos Fin de Ciclo** y futuros **Proyectos intermodulares** de los alumnos, incluyendo documentación, fotos, software asociado, información de los alumnos, calificaciones y composición de los tribunales de evaluación. A continuación, se describen los datos que se usan en la aplicación y las funciones de los usuarios de la aplicación.

La aplicación trabaja con una base de datos MySQL cuyo script de importación se entrega a los equipos. La aplicación de escritorio accederá a los datos de la base de datos consumiendo un servicio API Rest que se debe desarrollar.

La aplicación gestionará un repositorio de los proyectos que han entregado alumnos de ciclos formativos en el centro y que han obtenido calificación de aprobado. De todo proyecto que se almacene en el repositorio, quedará registrado:

* Un nombre (no se puede repetir).
* Tipo de proyecto (Fin\_Ciclo, Intermodular y otros posibles)
* Un resumen descriptivo.
* Palabras clave de búsqueda
* Año académico en el que se presenta (por ejemplo, si es 2024-25, se registra 2024).
* Ciclo formativo.
* Fecha de presentación.
* Calificación obtenida.
* Alumnos que lo desarrollaron
* Profesores que lo evaluaron
* Profesor tutor del proyecto (el que realiza el seguimiento)
* Logo del proyecto (url del archivo)
* Documento memoria del proyecto (url del archivo)
* Software y documentación adicional (url del archivo)
* Comentarios

De todos proyectos almacenados se han de registrar los datos de los alumnos que los desarrollaron/presentaron. Dado que puede haber distintos tipos de proyectos, un alumno podría haber desarrollado más de un proyecto. Un proyecto puede ser desarrollado por varios alumnos.

Los datos de los alumnos que se registran son:

* DNI
* Nombre
* Apellidos
* Dni
* Correo Electrónico y contraseña encriptada (creadenciales de acceso)
* Teléfono
* Genero
* Fecha de nacimiento
* Si es o no alumno actualmente
* Ciclo Formativo (se tiene el último que cursó o el actual)

Los alumnos actuales de segundos cursos de CFGS del centro podrán acceder a la aplicación, aunque no tengan proyectos entregados, para conocer información básica de los proyectos del repositorio.

En cada año académico se podrán añadir los alumnos que, no estando registrados aún, tengan que realizar proyectos en el curso. Cuando presenten sus proyectos y los aprueben se cargarán los datos de esos proyectos y se les asignarán a los alumnos correspondientes. Los datos de los alumnos se podrán cargar y modificar de forma manual o a través de un fichero de carga (CSV, por ejemplo).

Todos los profesores que han evaluado proyectos y los actuales que tengan que evaluar nuevos proyectos deberán estar registrados. Todos los profesores actuales registrados podrán consultar toda la información sobre proyectos y básica sobre alumnado y profesorado participante. Habrá uno o varios profesores actuales que actuarán como administradores y podrán realizar cualquier operación sobre los datos de la aplicación.

Los datos que se registran de los profesores son:

* DNI
* Nombre
* Apellidos
* Correo Electrónico
* Contraseña de acceso
* Teléfono
* Genero
* Fecha de nacimiento
* Especialidad
* Activo actualmente
* Si es administrador en la aplicación

En la tabla **realizan** se tiene información sobre los proyectos que realiza cada alumno. Un proyecto podría ser realizado por varios alumnos.

En la tabla **evaluan** se tiene información sobre los profesores que han realizado la evaluación de cada proyecto presentado. Un proyecto es evaluado por varios profesores.

Tanto alumnos como profesores se identifican con una cadena UUID.

Se tendrá registrada información sobre los ciclos formativos para los que se tiene el repositorio de proyectos. De cada ciclo formativo se tiene la siguiente información:

* Código
* Nombre
* Familia profesional
* RD de título
* Orden de currículo

En principio, las operaciones de actualización de información sobre los ciclos formativos no se realizan desde la aplicación, sino directamente en base de datos.

Los administradores podrán:

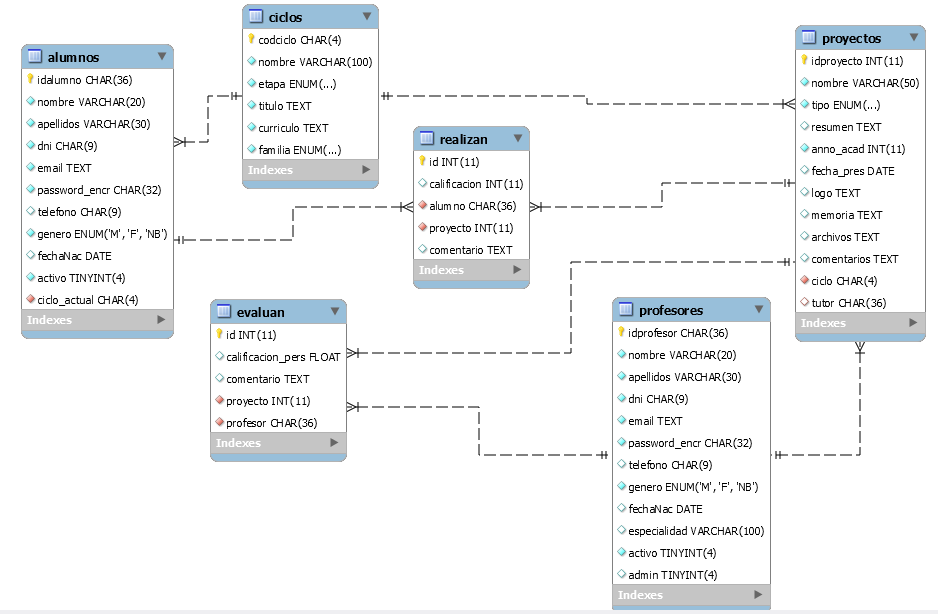
* Realizar operaciones de actualización de datos de alumnado.
* Realizar operaciones de actualización de datos de profesorado.
* Realizar operaciones de actualización de datos de proyectos.
* Asignar proyectos a profesorado para su evaluación.
* Asignar proyectos a alumnado.
* Consultar cualquiera de los datos registrados.

Un profesor activo y no administrador podrá:

* Consultar información básica sobre todos los proyectos almacenados.
* Consultar información completa sobre los proyectos que ha evaluado o que tiene asignados.
* Calificar a los alumnos de un proyecto asignado, siempre que no tengan calificación.
* Subir archivos de proyectos asignados y no evaluados.
* Añadir comentarios a proyectos asignados.

Un alumno actual podrá **(no requerido)**:

* Consultar información básica de todos los proyectos de su ciclo formativo.
* Subir archivos de sus proyectos no evaluados.
* Consultar toda la información de sus proyectos.
* Añadir comentarios a sus proyectos.



|  |
| --- |
| **Objetivos específicos del Minireto / Resultados de Aprendizaje** |

|  |  |  |
| --- | --- | --- |
| Objetivo específico | MODULOS | RESULTADOS DE APRENDIZAJE |
| 1. Realizar una API Rest Spring con uso de librería JdbcTemplate para acceso a una BD MySQL de Gestión de Proyectos y su consumo por aplicaciones clientes  2. Realizar una aplicación de escritorio en C# con buena usabilidad para que administradores hagan gestión de repositorio de proyectos consumiendo una API de acceso a una BD de Gestión de Proyectos  3. Coordinar el trabajo de desarrollo de la API y de la aplicación de escritorio en función de los requisitos actuales y futuros de esa aplicación de escritorio | ACCESO A DATOS | **RA2. Desarrolla aplicaciones que gestionan información almacenada en bases de datos relacionales identificando y utilizando mecanismos de conexión.** |
| 1. Realizar una aplicación de escritorio en C# (utilizando WinForms) que presente una buena usabilidad que permita el acceso, por parte de profesores y alumnos, al repositorio para la consulta de los diferentes tipos de trabajos. | DESARROLLO DE INTERFACES | **RA1. Genera interfaces gráficos de usuario mediante editores visuales utilizando las funcionalidades del editor y adaptando el código generado.** |

|  |
| --- |
| 1. **Equipos de trabajo** |

**Formación y compromiso**

Los equipos conformado con criterios de heterogeneidad. Para que exista un compromiso en el trabajo en equipo de trabajo se redactará y firmará en la primera jornada un **contrato de equipo** con el que se comprometen todos sus miembros.

**Roles en el equipo y coordinación**

Es muy importante la correcta coordinación entre los miembros de un equipo de trabajo para ser eficientes con el resultado a conseguir y disponer de un clima de trabajo apropiado. En los equipos habrá un miembro que tendrá el rol de **coordinador** y que será el encargado de coordinar las tareas que deban realizar los miembros del equipo y dirigir las reuniones del equipo, entre otras funciones.

Las funciones del coordinador deberán ser decididas antes de la primera jornada y establecidas en los contratos de equipo.

**El trabajo diario**

Los alumnos realizarán reuniones periódicas donde se decidirán las tareas a realizar en cada jornada y se analizarán los problemas y necesidades. Se asignan las tareas a los que van a ser responsables de realizarlas.

Se redactará en el cuaderno de bitácora, del que se entrega una plantilla, los detalles de lo realizado en la jornada anterior.

Al final de la jornada deberá producirse una reunión en la que se confirmen las tareas previstas que se han terminado y sobre los problemas de las que no se hayan podido resolver en tiempo. Se podrá cumplimentar en esta reunión el cuaderno de bitácora con lo realizado en la jornada.

En cada equipo se podrán distribuir las tareas de la forma que se acuerde que sea más adecuada y eficiente. Aunque sería conveniente para alcanzar los resultados de aprendizaje que los miembros de los equipos participasen en tareas asociadas a los dos módulos (DI y AD), puede que esto no sea eficiente especialmente en cuanto a tiempos.

**El seguimiento**

Cada profesor asistirá al aula en las horas que tiene asignadas en su módulo.

El profesor velará por:

* En su caso, la comprobación de que los alumnos participan en las reuniones de trabajo,
* Comprobar que el coordinador realice su labor de forma correcta.
* La resolución de las dudas en cuanto a los objetivos del reto.
* La resolución, en la medida de lo posible, de las dudas técnicas sobre el desarrollo del reto.
* El seguimiento del trabajo desarrollado en los equipos.
* El seguimiento del trabajo individual del alumnado
* Facilitar el aprendizaje y el trabajo colaborativo en el aula

Cuando un profesor no pueda resolver un problema u duda de carácter técnico, lo comunicará al al profesor del módulo relacionado con el problema.

|  |
| --- |
| 1. **Herramientas de seguimiento** |

Las herramientas que el profesorado utilizará para el seguimiento serán:

|  |  |
| --- | --- |
| **Seguimiento del equipo** | Github, cuaderno de trabajo y observación. |
| **Seguimiento del alumnado** | Cuaderno de trabajo y observación. |

|  |
| --- |
| 1. **Herramientas de comunicación y colaboración** |

Se especifican las formas en las que se produce la comunicación entre los miembros de los equipos y entre el alumnado y los profesores.

|  |  |
| --- | --- |
| **Dentro del alumnado** | Oral en el aula, Teams, correo electrónico y WhatsApp |
| **Entre alumnado y profesorado** | Teams, correo electrónico y oral en el aula |

|  |
| --- |
| 1. **Tareas a realizar, recursos elaborados y formato** |

En el proceso del reto se deberán generar los siguientes recursos finales:

|  |
| --- |
| **RESULTADOS / RECURSOS GENERADOS** |
| Documentación del reto realizado con Markdown e incluido en repositorio GitHub |
| Código fuente de la API en repositorio GitHub |
| Proyecto en Visual Studio con el código de la aplicación de escritorio e interfaz generado |
| Cuaderno de trabajo de los equipos (DOC) |
| Ejecutables de las aplicaciones y aplicaciones desplegadas |

En la documentación del reto se deberá hacer referencia al menos a:

* El nombre del reto y una descripción corta.
* Los miembros del equipo
* Los resultados obtenidos
* Las tecnologías utilizadas en el desarrollo
* Los resultados esperados para cada módulo
* La valoración de lo realizado
* Las mejoras propuestas
* Bibliografía y webgrafía

En la última jornada del reto, cada equipo hará una **exposición** en la que muestren y expliquen el funcionamiento de la aplicación de escritorio y, si lo consideran procedente, del funcionamiento de alguna parte importante de la API Rest.

|  |
| --- |
| 1. **Especificaciones del reto para cada módulo** |
| **Módulo: Acceso a Datos** |
| En **Archivos de Teams** tenéis un archivo “**BASE DE DATOS REPOSITORIO PROYECTOS.PDF**” que incluye el esquema relacional de la base de datos de repositorio de proyectos y su descripción y un archivo “**script importacion BD proyectos.sql**” que debes usar para importar la base de datos MySQL.    A partir de esa base de datos debes crear una API Rest desarrollada con Spring y librería **JdbcTemplate,** que incluya los métodos necesarios para que las aplicaciones clientes puedan trabajar con la base de datos para la gestión de repositorio de proyectos. Es importante dotar de una funcionalidad básica a la API para que pueda evolucionar el desarrollo de las aplicaciones clientes.    Al trabajar con un repositorio de proyectos, se tratará de almacenar en repositorio la documentación, software y logo de cada proyecto. Se deberá implementar en la API funcionalidad para que al hacer el cliente una petición de almacenamiento, se suba el fichero adjunto, se guarde en una ruta de almacenamiento y se registre en base de datos la correspondiente ruta.  La API debe estar documentada con **swagger**.    Los desarrollos de la API deben estar dentro del repositorio GitHub del equipo.    Dentro de la documentación elaborada con MarkDown, también se indicarán los **endpoints** de la API, las capas en las que se ha estructurado y otras cuestiones consideradas de interés.    Se pueden incluir mejoras como la implementación de una API segura. |

|  |
| --- |
| **Módulo: Desarrollo de Interfaces** |
| Se debe crear una aplicación de escritorio que haga uso de la API para la comunicación con la base de datos.  Se debe diseñar la interfaz de la aplicación de escritorio que incluya los formularios correspondientes teniendo en cuenta el acceso y funciones que puede realizar cada usuario, administrador, profesor y alumno (opcional). Por tanto, será necesaria la utilización de los controles y elementos necesarios para poder acceder a los datos de los repositorios de proyectos, así como para poder realizar tareas como la subida de archivos o la introducción de comentarios.  Se deberá tener en cuenta los criterios de usabilidad y accesibilidad, permitiendo un uso eficaz y de forma autónoma por parte de todos los usuarios.  El desarrollo de la aplicación debe estar dentro del repositorio GitHub del equipo. |

|  |
| --- |
| 1. **Criterios de evaluación** |

En la siguiente tabla se especifican los resultados de aprendizaje que se trabajan en el reto inicial para cada módulo y los criterios de evaluación que se utilizarán.

|  |  |
| --- | --- |
| **MÓDULO PROFESIONAL:** | **ACCESO A DATOS** |
| **RA. Resultados de Aprendizaje.** | **Criterios de evaluación** |
| **RA2. Desarrolla aplicaciones que gestionan información almacenada en bases de datos relacionales identificando y utilizando mecanismos de conexión.** | 2.a) Se han valorado las ventajas e inconvenientes de utilizar conectores.  2.c) Se utilizado el conector idóneo en la aplicación  2.f) Se han desarrollado aplicaciones que modifican el contenido de la base de datos  2.h) Se han desarrollado aplicaciones que efectúan consultas |

|  |  |
| --- | --- |
| **MÓDULO PROFESIONAL:** | **DESARROLLO DE INTERFACES** |
| **RA. Resultados de Aprendizaje.** | **Criterios de evaluación** |
| **RA1. Genera interfaces gráficos de usuario mediante editores visuales utilizando las funcionalidades del editor y adaptando el código generado.** | 1.g) Se ha desarrollado una aplicación que incluye el interfaz gráfico obtenido. |

|  |
| --- |
| 1. **Recursos** |

* Personas de contacto
  + Luis Jesús Herrero de Cos (ACCESO A DATOS)
  + Felipe Moreno Díaz (DESARROLLO DE INTERFACES)