



Tecnológicas Avanzadas de Desarrollo  
Grado en Ingeniería Informática en Sistemas de Información - Curso 2017/2018  
**Condiciones proyectos**

## Objetivos

- Poner en práctica los conocimientos adquiridos a lo largo de la asignatura.
- Integrar las distintas tecnologías estudiadas para resolver un único problema.
- Manejo de herramientas colaborativas en el desarrollo software.

## Introducción

Las tecnologías estudiadas en la asignatura se han visto en bloques independientes, sin embargo, durante el desarrollo de aplicaciones reales, lo normal es implementar una única aplicación que haga uso de las distintas tecnologías disponibles según el cometido propio de cada una.

Durante el desarrollo de este proyecto, el alumno tendrá que manejar y aplicar las distintas tecnologías estudiadas a lo largo del curso para poder desarrollar una aplicación similar a la que encontrará en escenarios reales.

## Temática del proyecto

La temática del proyecto será libre y única, de modo que NO podrá haber más de un grupo con la misma temática. Las temáticas de los proyectos, junto con los participantes en el mismo, deberán registrarse a través de WebCT en un foro que se habilitará para ello. En caso de que dos grupos registren la misma temática, ésta será asignada en función del orden de registro.

## Especificaciones

El proyecto consistirá en el desarrollo e implementación de una aplicación sobre la temática elegida. Los proyectos se realizarán en grupos de 3 personas.

El desarrollo del proyecto se realizará utilizando **repositorio GIT**. Para ello cada grupo deberá crear un repositorio GIT, dándose de alta los diferentes integrantes del equipo. Además, deberán dar de alta a otro usuario que hará las veces de observador (profesor). Una vez creado el repositorio y dado de alta a los integrantes, se comunicará la cadena de conexión al repositorio mediante una tarea habilitada en webCT para tal fin.

La aplicación desarrollada deberá cumplir las especificaciones funcionales siguientes:

1. Utilizará Maven como herramienta software para el control de versiones y gestión del ciclo de vida del proyecto.
2. Se implementará mediante el framework Vaadin un front-end para ser utilizado por usuarios de la aplicación y un back-end para la administración de la aplicación. Será necesario dotar a las páginas de una estética adecuada empleando CSS. **duda sobre esto. como se adapta esto a nuestro proyecto? No vale hacer una app para administradores solo y ya esta? Como se adapta vaadin a dos tipos de usuarios distintos?**
3. Cada aplicación deberá constar de, **al menos**, la funcionalidad necesaria para la gestión (inserción, borrado, actualización y búsqueda) de una entidad por participante.
4. El diseño de la interfaz de usuario se realizará empleando Vaadin.
5. Se utilizará el patrón de arquitectura de software MVC.



6. El back-end deberá mostrar **obligatoriamente** algún tipo de gráfico y calendario y utilizar la técnica drag&drop.  
**mysql en azure cuesta dinero**
7. Se almacenarán los datos en una base de datos MongoDB preferentemente, o MySQL. Los script de creación de base de datos deberán contemplar el caso en que previamente exista una base de datos con el mismo nombre, por lo que deberán contener las líneas de código SQL necesarias para su borrado. Así mismo, deberá contemplar la inserción de datos de ejemplo para la aplicación.
8. Podrán usarse librerías, métodos y aspectos referentes a estas tecnologías no tratados en clase. Estas adiciones extraordinarias incidirán en un incremento, por encima de la valoración ordinaria, de la nota.
9. La aplicación podrá ser desplegada en un servidor de aplicaciones perteneciente a un cloud tipo Azure.

La documentación asociada al proyecto deberá recoger los siguientes puntos:

1. Descripción del problema a resolver. Este apartado describirá el tema abordado, sus características generales, descripción del funcionamiento de la organización que se va a plasmar, restricciones generales, aspectos del sistema que se van a cubrir, aspectos que no se van a tratar, etc.
2. Recolección de los requerimientos del problema. Consistirá en expresar, en lenguaje natural y de forma ordenada, la lista de especificaciones que debe cubrir el diseño, así como las posibles restricciones que presente el problema.
3. Elaboración de un esquema funcional. Habrá que diseñar un diagrama funcional de la aplicación que proporcione la interacción entre los distintos componentes (páginas web, base de datos, etc.). También habrá que proporcionar una descripción del funcionamiento detallado de cada uno de los componentes. Concretamente, será necesario un esquema y su correspondiente descripción, donde se especifique la estructura de la web, el propósito de cada uno de sus componentes y la interrelación entre ellos.
4. Diagrama UML del proyecto elaborado. Se describirán las clases implementadas.
5. Se documentará el código tanto internamente (comentarios en el mismo) como externamente (documentación explícita de su funcionamiento e interrelación de sus componentes). Se deberá describir las clases utilizadas, así como sus atributos y métodos mediante el uso de javadoc.
6. Esquema Maven con el diagrama de dependencias. Se realizará una captura del esquema gráfico Maven proporcionado por Netbeans y se describirán las principales dependencias del proyecto.
7. Diseño de la base de datos mediante un diagrama entidad-relación, así como una descripción y justificación de las entidades y relaciones que contiene. En caso de utilizar una base de datos noSQL, se especificará el diseño físico de la base de datos orientado a su implementación en MongoDB. Independientemente del tipo de base de datos elegido, se proporcionará información y una justificación acerca del tipo de atributos, las claves primarias y externas, etc.
8. Manual de uso. Se deberá escribir un manual de usuario que proporcione información necesaria para que un usuario sin conocimientos previos pueda hacer un uso completo de la aplicación.
9. Es conveniente que los elementos extraordinarios añadidos (denominados extras) estén destacados en unos apéndices (independientemente de si aparecen en otras secciones).

## Valoración

---

La aplicación deberá funcionar correctamente abriendo los proyectos incluidos en el material entregado para que pueda ser evaluada. Además deberá ser desplegada en Azure (o cloud análogo) previamente a la sesión de presentación del proyecto.

Parte del trabajo desarrollado se evaluará mediante una exposición oral de una duración de 15 minutos. Dicha exposición deberá abarcar todos los aspectos del trabajo, incidiendo en las ventajas y bondades del sistema desarrollado, y no se centrará en los detalles técnicos salvo que éstos sean de relevancia (Por ejemplo, adiciones extraordinarias sobre las tecnologías vistas en la



asignatura). Será necesario que todos los miembros del equipo participen de una forma activa y equitativa en la presentación.

**Con el fin de evitar posibles problemas técnicos, para la presentación el grupo de alumnos deberá disponer de un equipo informático en el que habrá desplegado el proyecto a presentar, asegurándose de su operatividad previamente a la presentación.**

Los aspectos a evaluar son:

1. Originalidad del trabajo (P. ej. de la temática o la forma de abordar la aplicación).
2. Calidad del informe de la aplicación y del código fuente (P. ej. organización, abundancia de explicaciones, diagramas, calidad del contenido, etc.)
3. Calidad de la presentación (P. ej. calidad de la expresión oral, del contenido de las transparencias, de la demostración de la aplicación, etc.).
4. Uso de herramientas colaborativas a lo largo del cuatrimestre. Este punto será valorado a lo largo del cuatrimestre.
5. *Extras* incluidos: número de entidades incluidas por encima de las obligatorias, uso de material no tratado en clase de las distintas tecnologías abarcadas durante el curso (P. ej. el envío de correos electrónicos, la manipulación de imágenes, la exportación a PDF, etc.). Las valoraciones de este apartado significarán un aumento de la nota por encima de la valoración ordinaria. Especialmente se valorará el despliegue de la aplicación en la nube, de forma que la presentación se realice en base al servicio on-line desplegado.

## Entrega

---

El trabajo realizado deberá entregarse el mismo día de la defensa (se comunicará por webCT el día concreto con suficiente antelación).

La entrega consistirá en un CD que contendrá los siguientes ficheros:

1. Presentación en vídeo con formato mp4 (por si todo falla siempre nos quedará este recurso para la presentación), denominado **proyecto\_TAD.mp4**.
2. Paquete zip conteniendo el proyecto con el código fuente de la aplicación. Denominado **proyecto\_TAD.zip**.
3. Fichero war para realizar el despliegue en un servidor de aplicaciones. Denominado **proyecto\_TAD.war**.
4. Script de creación de la base de datos. Denominado **base\_datos.sql**.
5. Script con los insert necesarios para disponer de suficientes datos de ejemplo en la base de datos. Denominado **datos\_ejemplo.sql**.
6. URL del proyecto desplegado en Azure (o cloud análogo).
7. Documentación asociada al proyecto (**deberá entregarse, además, una copia en papel**).