

# Ejercicio obligatorio final

Durante el modulo hemos trabajado sobre las herramientas y metodologías que se producen a lo largo del ciclo de vida de un software. En esta práctica vamos a realizar algunas de las tareas que engloba este ciclo desde el punto de vista del equipo de desarrollo.

El alumno deberá ir creando y completando los diferentes apartados que se soliciten, como si de una memoria de proyecto se tratase.

## Fase de definición

Vamos a trabajar sobre una aplicación en la cual podrás incluir tus propias aportaciones en el diseño. Se trata de una aplicación para la **gestión de eventos de un centro de formación**. Los requerimientos de esta aplicación son:

- Se precisa **gestionar y almacenar** el alta de las **personas** que se quieran inscribir a los **eventos**. Se guardarán el DNI, nombre, teléfono, mail y lugar de residencia.
- El programa debe poder **gestionar** las **ubicaciones** donde se producirán los **eventos**, almacenando su dirección, aforo y nombre.
- Los eventos tendrán una fecha de celebración, título del evento, el precio y una lista de los ponentes que además de los datos que almacenamos en las personas, añadiremos el título académico o cargo por el cual se presenta al evento.
- Los asistentes podrán inscribirse vía web al evento que deseen después de darse de alta. Aunque el administrador podrá confirmar la asistencia de los participantes y generar los eventos.
- La aplicación puede estar realizada en cualquier **lenguaje de programación**, pero debe estar **orientado a objetos**.
- **Tarea:** Añade algún nuevo requisito a la aplicación según tu criterio (Mínimo tres).

## Fase de diseño

Ahora realizaremos los principales diagramas UML para el diseño de una aplicación. Elabora tanto el diagrama de clases como el de casos de uso. No te olvides de incluir los requerimientos que has incluido en el punto anterior.

- **Tarea:** Diagrama de casos de uso.
- **Tarea:** Diagrama de Clases.

## Interfaces

Para realizar cualquier programa debemos disponer de un medio de comunicación con los usuarios. Utilizando algún programa para realizar mockups crea un boceto de las principales pantallas que tendrá la aplicación. Formulario de alta de asistentes, creación de eventos, login... <https://www.initcoms.com/blog/5-herramientas-gratis-para-crear-mockups/>

- **Tarea:** Bocetos.

## Tecnologías:

Otro punto importante para cubrir cuando estamos en la fase de análisis y diseño de una aplicación son los recursos y tecnologías que usaremos

- **Tarea:** Lista de lenguajes que utilizaremos para la creación de la aplicación.
- **Tarea:** Lista de herramientas de software que utilizaremos para la creación de la aplicación.
- **Tarea:** Lista de elementos de hardware que utilizaremos para la creación de la aplicación.

## Fase de implementación:

Aunque no tenemos como objetivo realizar una aplicación funcional, si se nos solicita que codifiquemos alguna de las clases del modelo generado en el diagrama de modelado.

Crea la codificación de las clases participantes y eventos.

La codificación debe tener todos los atributos expuestos en los requerimientos del enunciado, más los que hayas podido aportar.

También se solicita la construcción del o los métodos constructores, métodos “**getters & setters**” y algún método propio que consideres.

Para el cálculo del precio del evento, debemos tener un método que nos devuelva el importe del evento más el IVA de 21%.

También se nos solicita que realicemos una correcta documentación del código, por lo que tendremos que incluir los comentarios en el formato correcto. (Ejemplo: JavaDoc, PhpDoc...).

- **Tarea:** Código de las clases personas y eventos.
- **Tarea:** Documentación (Javadoc)

## Fase de pruebas

Toda pieza de software antes de ser implantada necesita pasar por una fase de pruebas para comprobar su correcto funcionamiento. Una de las pruebas más comunes en el desarrollo son las unitarias.

Realizar una serie de pruebas a algunos de los métodos que has implantado en el código anterior, por ejemplo, el cálculo de IVA. Para ello puedes usar un framework como Junit.

- **Tarea:** Código de la función Test de las pruebas unitarias.

DIAGRAMA CASOS DE USO: <https://www.youtube.com/watch?v=orvAkFFWo5o>