



PROYECTO EN GRUPO

Aplicación web

Objetivos

- Diseñar una aplicación web corporativa en **grupos**.
- Programar una aplicación web en el entorno .NET usando Visual Studio. Esto implica seguir el diseño realizado previamente, hacer uso del lenguaje de programación C# y desarrollar la aplicación con el interfaz de usuario Web Forms de ASP.NET.
- Aprender las habilidades necesarias para el trabajo en grupo. Supone la gestión y el seguimiento del proyecto, reparto y planificación de las tareas, realización de autoevaluaciones y establecimiento de reglas de funcionamiento interno del grupo.
- Documentar el proyecto a lo largo de todas sus fases, indicando cuáles han sido los requisitos abordados, el diseño realizado y su implementación. También deberéis indicar cuáles han sido los problemas encontrados en el grupo y las soluciones aportadas. Por último, deberéis llevar a cabo el seguimiento del proyecto, horas dedicadas por cada miembro y trabajo realizado.
- Usar git en un proyecto en grupo y utilizar el repositorio remoto GitHub (<https://github.com>).

Descripción

Se ha de desarrollar una aplicación web de forma colaborativa. La aplicación a desarrollar será elegida por el grupo de alumnos en cuestión (mínimo 2, máximo 3 alumnos) y estará orientada a ofrecer determinados servicios en Internet (hay varios ejemplos en el anexo de la práctica), como p. ej. tienda virtual de informática, farmacia online, etc. Se ha de describir la propuesta y presentarla al profesor para que la evalúe según los requisitos especificados en este enunciado.

Los alumnos deberán utilizar el sistema de control de versiones git para gestionar el código y realizar las distintas entregas. Para ello se hará uso del repositorio GitHub (<https://github.com>).

Habrà un **coordinador** del grupo encargado de gestionar el proyecto en github. El coordinador debe (ver transparencias detalladas): Crear el repositorio, añadir miembros, crear ramas por cada miembro, coordinar el seguimiento del proyecto, gestionar tareas, fusionar ramas. Por su tarea como coordinador, de estar bien realizada podrá tener hasta 0,5 pto adicional.

Requisitos mínimos

- Cada una de las páginas debe validar que se han introducido los campos obligatorios, el tipo de los campos, su tamaño y que su valor sea correcto (p.e. si es una fecha debe comprobar que sea correcta, si es un NIF debe comprobar que la letra sea correcta, etc.). Para las validaciones de los datos utilizaremos controles de validación, expresiones regulares y excepciones.
- Habrá, al menos, dos EN por alumno, y por cada entidad podremos realizar todas las operaciones CRUD (create, read, update, delete) que sean posibles (en los CAD asociados). Se implementará la búsqueda por palabra clave sobre, al menos, una EN en la aplicación Web. Por ejemplo:
 - EN producto: podremos crear producto, mostrar producto, editar producto y borrar producto, además de buscar productos por palabra clave (ej, tipo, precio, código...).

Importante: de las dos EN requeridas por alumno, una podrá ser sustituida por funcionalidades adicionales (ver apartado posibles mejoras) a implementar sobre otra EN.

- Para las entidades que necesitemos dar de alta elementos podremos hacerlo directamente en la BB.DD. para que ésta no esté vacía (datos realistas).
- La BB.DD. será creada en el entorno SQLServer y habrá que entregar el esquema de la BB.DD.



La aplicación tendrá dos perfiles de usuario. La página principal de la aplicación cliente tendrá un menú (**control menu ASPNET**) siempre visible situado en una **página maestra**. Algunas opciones de este menú cambiarán según estemos en la parte pública o privada de la web.

Parte pública:

- En esta parte el usuario podrá acceder a la aplicación web sin loguearse y consultar información en la misma. Por ejemplo en una tienda virtual una acción permitida para este usuario sería consultar los productos y categorías, pero no podría comprarlos.
- Desde la parte pública podremos ver la información básica de la empresa gestionada y de la aplicación y darnos de alta como clientes (registro) para acceder a la parte privada de la web.
- Se deberá poder navegar por la lista de “productos” ofrecidos, los cuales tendrán una **imagen asociada**, y se deberá poder consultar su información detallada.
- Se deberá poder realizar búsquedas con filtro por uno o más campos sobre una EN del sistema.

Parte privada:

- Una vez registrado, el usuario puede loguearse y acceder a todas las secciones de la aplicación, haciendo uso de los servicios ofertados
- Además este usuario podrá *editar los datos de su perfil* al registrarse (modificar nuestros datos como cliente web, darnos de baja, o consultar nuestros datos.).
- En la parte privada se deberá almacenar, en la **sesión web**, la información sobre el cliente que sea necesaria para mostrar información personalizada o llevar a cabo otras funciones.
- Se deberán ofrecer tantos **servicios** como alumnos hayan en el grupo sobre las EN del sistema (p. ej. compra, reservas, etc).

Posibles mejoras

Se valorarán las mejoras que proporcionen una contribución relevante para la propuesta realizada de aplicación web. Estas mejoras **deberán ser realizadas en .NET**. Algunas ideas:

- Internacionalización
- Personalización
- Uso de servicios web externos (p.ej. Google Maps)

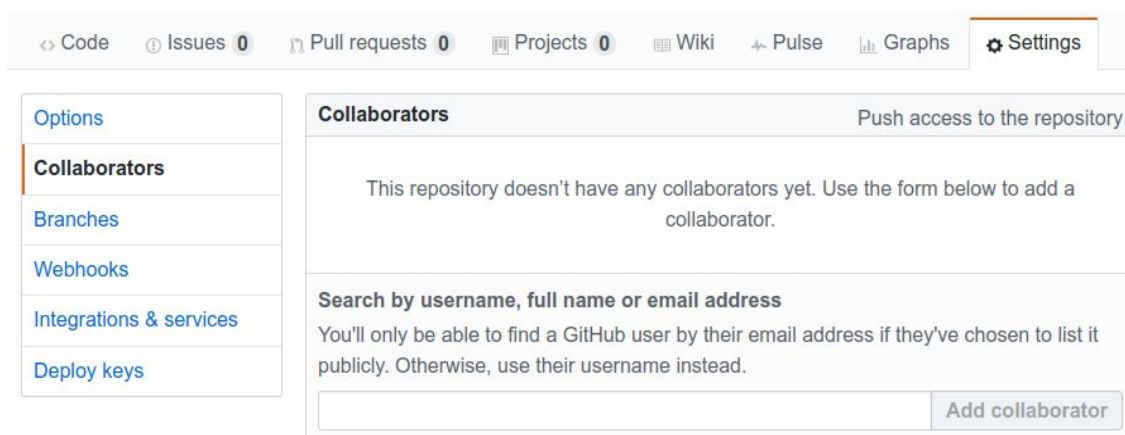
- Enlace con redes sociales (p. ej. Twitter, Facebook)
- **Funcionalidad adicional de relevancia que se ajuste al proyecto.**
- ...

Control de versiones: git + github

Para que sea más sencillo el desarrollo en grupo de la práctica vamos a emplear **git** junto con **github**. Esto también hará posible que vuestro profesor de prácticas siga la evolución del proyecto y el trabajo individual de cada estudiante en cada momento.

Para ello el **coordinador del grupo** creará un repositorio **PRIVADO** en github utilizando su cuenta creada con el email institucional de la UA (@alu.ua.es), de este modo todo vuestro código sólo será visible por vosotros y vuestro profesor o profesora de prácticas.

Una vez creado el repositorio para vuestro proyecto en github deberéis añadir como colaboradores a vuestro profesor de prácticas y a vuestros compañeros de grupo, para ello id a la pestaña '**Settings**' del mismo y en el menú de la izquierda seleccionad '**Collaborators**':



Importante: Cada colaborador del proyecto debe tener una cuenta creada en github.

Todo proyecto en github puede incluir un archivo llamado 'README.md' en el cual se describe el proyecto, evolución o lo que el desarrollador quiera. Nosotros lo empleamos para describir nuestro proyecto, su nombre, lo que hace, los desarrolladores que componen el grupo de trabajo, etc.

La extensión '**.md**' es una abreviatura de [Markdown](https://guides.github.com/features/mastering-markdown/) (<https://guides.github.com/features/mastering-markdown/>), un lenguaje de marcado muy ligero que permite escribir documentos con negritas, cursivas,

títulos, imágenes, enlaces, etc...

Uso de ramas (ver presentación github para detalles)

Para evitar al máximo conflictos durante el desarrollo de la aplicación web:

- Cada componente del grupo deberá trabajar sobre una **rama** en el repositorio **llamada como su DNI/NIE**. De este modo, tanto el coordinador del grupo, como el profesor de prácticas podrán comprobar el trabajo individual de cada persona.
- Habrá una rama **develop** sobre la que se irán integrando los cambios realizados para verificar que funciona correctamente (antes de integrarlo en master). **Cada alumno debe hacer merge entre su rama y develop frecuentemente**, es decir, cada vez que se acabe una funcionalidad deberá resolver los conflictos que se generen en su rama con los últimos cambios que contiene develop. Cuando no hayan conflictos en la rama del alumno, se podrán integrar sus cambios en esta rama develop.
- Deberéis nombrar un **coordinador del grupo**, éste será la persona encargada de integrar los commits correspondientes de cada integrante del proyecto en la rama master y develop (además de las tareas que se detallan en la presentación de github).

Entregas

1. Entrega de la propuesta

El responsable entregará un documento con el nombre de todos los componentes del grupo, en el cual se especificará el grupo de prácticas. La descripción de la propuesta contendrá los siguientes apartados:

- **Descripción:** donde se explicará en general el tipo de aplicación web que se desea implementar.
- **Parte pública:** se deberá especificar la funcionalidad de la parte pública de la web (sin estar registrado). Es conveniente usar una lista en forma de viñetas (como esta), donde explicaremos un poco cada opción.
- **Listado EN Pública:** Además se entregará un listado con las EN que hemos identificado en esta parte.
- **Parte privada:** se deberá especificar la funcionalidad de la parte privada de la web (una vez registrado). Es conveniente usar una lista en forma de viñetas, donde explicaremos un poco cada opción.
- **Listado EN Privada:** Además se entregará un listado con las EN que hemos identificado en esta parte.
- **Posibles mejoras:** En los apartados anteriores habremos especificado la funcionalidad teniendo en cuenta los requisitos mínimos especificados más

arriba, en este apartado describiremos las posibles mejoras que pensemos añadir. No se trata de una lista cerrada, pero es conveniente describir las ideas que se os ocurran para luego implementar las más convenientes.

La entrega se hará en GitHub, en el fichero **README.md** de la rama *master*, con la etiqueta (comando git tag) *propuesta*, asociado a un Milestone llamado *deadline1*.

Fecha límite: 05/04/2020

2. Entrega de EN/CAD y esquema de la BB.DD.

- Se creará un proyecto de tipo biblioteca de clases en C#, en Visual Studio, en la cual añadiremos las clases necesarias para representar cada una de las EN de nuestra aplicación. Cada clase tendrá los atributos necesarios y las propiedades para hacerlos públicos, así como las cabeceras de los métodos que necesitaremos con las llamadas a los CAD.
- Por cada EN se deberá entregar la clase CAD correspondiente con las cabeceras de los métodos.
- Además habrá que hacer un esquema de la BB.DD. Este esquema podrá hacerse usando diagrama de clases o EERR.

La entrega se hará en la rama *master* con la etiqueta (comando git tag) *ENCAD*, asociado a un Milestone llamado *deadline2*. Esto lo hará el coordinador del grupo. En el fichero **README.md** se indicará claramente dónde está el esquema de la BB.DD. (nombre del fichero y ubicación).

Fecha límite: 12/04/2020

3. Entrega de la interfaz

- Se entregará el proyecto web de ASP.NET, con las páginas web principales: la página principal con el menú ASP.NET (página maestra) y cada opción del menú. Los manejadores deben existir pero no es necesario que compile, en esta entrega se valorará la interfaz básica.

La entrega se hará en la rama *master*, con la etiqueta (comando git tag) *interfaz*, asociado a un Milestone llamado *deadline3*. Esto lo hará el coordinador del grupo.

Fecha límite: 03/05/2020

4. Entrega final

Una vez finalizado el proyecto, esta entrega contará con el siguiente contenido:

- Propuesta del proyecto entregada y documento explicando si ha habido cambios en la misma, dificultades encontradas, problemas con personas en el grupo, problemas de planificación, etc. Toda esta información estará en el fichero **README.md** de esta rama.
- **Solución conteniendo el Proyecto de librería y el proyecto web**, incluyendo BB.DD., imágenes, y todo lo necesario para su funcionamiento.
- Instrucciones de instalación, así como cualquier requisito para su correcto

funcionamiento.

- Usuario y contraseña necesarios en su caso.
- Tareas hechas por cada miembro del grupo. Ejemplo:

Nombre	Alta usuario.aspx, cad usuario, en usuario...
--------	---

- Presentación (si continúa la suspensión de clases presenciales se entregará un google slides con audio).

La entrega se hará en la rama *master con la etiqueta final* (comando git tag), *con el Milestone deadline4*. Esto lo hará el coordinador del grupo. En el fichero **README.md** además de la documentación explicada arriba junto con la tabla de tareas, se indicará claramente dónde está la presentación (nombre del fichero y ubicación/enlace).

Fecha límite: 24/05/2020

Guía de evaluación y requisitos técnicos

A continuación se describen los aspectos a valorar de la aplicación web desarrollada:

- **Facilidad de uso de la interfaz:** La interfaz debe ser fácil de usar y su manejo debe ser intuitivo. Los mensajes de ayuda al usuario, así como los mensajes de error, deben ser adecuados y facilitar al usuario la interacción con la aplicación.
- **Diseño de la interfaz:** La interfaz debe tener un aspecto homogéneo en todas las páginas y hacer uso de controles ASP.NET y hojas de estilo CSS. Adicionalmente se pueden usar para la interfaz AJAX, Silverlight, etc.
- **Capa EN y validación:** Los campos de las EN deben ser privados y se deben usar propiedades para acceder a ellos. La validación de datos se debe hacer mediante el uso de controles de validación web.
- **Capa CAD y gestión de errores:** Se deben implementar todos los métodos CRUD (create, read, update, delete) y hacer uso de excepciones para gestionar los posibles errores. Adicionalmente se debe hacer uso de los métodos conectado y desconectado para el acceso a la BB.DD..
- **Documentación del código:** El código de la aplicación debe estar suficientemente documentado mediante la inclusión de comentarios en el código fuente.
- **Uso de GIT:** El uso de GIT durante el desarrollo de la aplicación es **obligatorio**. Asimismo es **obligatorio** que **cada miembro** del grupo trabaje en **una rama** cuyo nombre sea su DNI/NIE.



Además, para establecer la nota final de esta parte de asignatura se valorarán las **mejoras** introducidas con respecto a la propuesta original, los **documentos entregados** (propuesta, EN/CAD y esquema de la BB.DD., etc.) y la **presentación** del proyecto (o bien la utilizada en la clase presencial o bien el google slides con audio, si continúa la suspensión de clases presenciales).

Importante:

- **El proyecto en Github deberá ser privado, si se detecta copia**, los alumnos implicados **suspenderán esta práctica** automáticamente. Por ello os recomendamos que borreís vuestros archivos una vez finalicéis la sesión de prácticas y dejéis cerrado el repositorio git.
- **Cada entrega parcial no realizada o insuficiente restará 1 punto de la nota final.**
- **Si el programa no compila**, la nota de la práctica será **cero**.
- **Si no se ha usado git**, la calificación de la práctica será **cero**.
- Aquellos alumnos que **no hayan trabajado en su rama** suspenderán la práctica con la calificación de **cero**.
- El último commit de cada una de las entregas deberá tener una fecha anterior o igual a la fecha de entrega publicada. **Cualquier commit con fecha posterior a la de la entrega no será tenido en cuenta.**
- **Los requisitos mínimos especificados en la práctica deben cumplirse o la nota final será cero.**

Anexo: Proyectos de ejemplo

A continuación se describen 3 posibles alternativas funcionales para realizar el proyecto:

Comercio electrónico

Una aplicación que pretende la venta de artículos por Internet. La vista de cliente realizada en Web Forms está encargada de los siguientes aspectos:

- El cliente accede a una página principal donde se muestran los productos en oferta, desde los cuales se puede acceder directamente al producto. Además se muestran las categorías de productos donde puede consultar la lista de productos.
- Una vez accedido al producto se mostrará una imagen grande, una descripción del producto y además el número de artículos que tenemos disponibles. El cliente puede introducir el número de artículos que considere necesario en su carrito.

- Creamos un carrito de la compra donde el cliente almacenará los artículos que desee comprar.
- En la parte PRIVADA, el cliente podrá realizar un pedido a partir del carrito de la compra, para realizar dicho pedido se necesita que el cliente esté registrado, solicitamos la clave y el password. Si no está registrado se mostrará un formulario de registro con los datos personales, la clave y el password. Para el pedido el cliente debe indicar la dirección, el tipo de pago, si es con tarjeta o al contado, si es con tarjeta solicitar el número. Una vez realizado el pedido nos iremos a la página principal.

Agencia de viajes

Una aplicación que pretende la reserva de viajes a través de internet. La vista de cliente PÚBLICA realizada en Web Forms está encargada de los siguientes aspectos:

- El cliente accede a una página principal donde se muestran los viajes en oferta, desde los cuales se puede acceder a los detalles de los mismos. Además se muestran los diferentes tipos de viaje donde puede consultar la lista de viajes ofertados.
- También habrá un buscador de viajes (por fechas, precio, ciudad destino etc).
- Una vez accedido al detalle del viaje se mostrará más información sobre el mismo: qué incluye, número de días, imágenes, etc.
- El cliente tendrá la opción de registrarse para acceder a la parte privada de la Web. Si es un nuevo usuario se mostrará un formulario de registro con los datos personales, la clave y el password.

Registrándose (vista PRIVADA), el usuario también podrá realizar las siguientes acciones:

- Una vez accedido al detalle del viaje el cliente podrá desde aquí, reservar el viaje consultado.
- Se podrán reservar viajes cerrados (es decir paquetes con vuelo+estancia etc) o se podrán reservar vuelos y/o estancias en hoteles o albergues. Existirá por tanto, una opción para reservar vuelos (pudiendo elegir fechas y horarios) y otra para la reserva de hoteles (pudiendo elegir entre varios ofertados y la fecha de la reserva).

- El cliente podrá gestionar las reservas realizadas desde una página Web.
- El cliente también podrá confirmar una reserva. Para la compra el cliente debe indicar la dirección, el email, el número de la tarjeta de crédito a utilizar. Una vez realizado la compra se enviará un email de confirmación al usuario.

Sistema de distribución de vuelos

Desarrollamos una aplicación para la gestión y control de los vuelos y billetes en diferentes aeropuertos del mundo. La vista de cliente PÚBLICA realizada en Web Forms está encargada de los siguientes aspectos:

- El cliente accede a una página principal donde se muestran los vuelos en oferta, desde los cuales se puede acceder a los detalles de los mismos.
- También habrá un buscador de viajes (por fechas, tarifas, ciudad destino, pasajeros, compañía aérea, etc.). Entendemos por viaje un compendio de vuelos que nos llevan desde un aeropuerto origen a otro destino. Debe mostrarse los resultados ordenados por diferentes criterios, precio, menor tiempo, menor número de vuelos, misma compañía aérea, etc.
- Una vez accedido al detalle del viaje se mostrará más información sobre cada vuelo: aeropuerto origen y destino, compañía aérea, información sobre el avión, las comidas, las plazas disponibles, las tarifas, etc.
- El cliente tendrá la opción de registrarse para acceder a la parte privada de la Web. Si es un nuevo cliente se mostrará un formulario de registro con los datos personales, incluyendo su pasaporte, una clave y el password.

Registrándose (vista PRIVADA), el cliente también podrá realizar las siguientes acciones:

- Una vez accedido al detalle del viaje el cliente podrá desde aquí, reservar o comprar los billetes de los diferentes vuelos, cada uno con una determinada tarifa. Para la compra el cliente debe indicar la dirección, el email, el número de la tarjeta de crédito a utilizar. Una vez realizado la compra se enviará un email de confirmación al cliente.
- El cliente podrá consultar las compras realizadas que le mostrarán los billetes comprados, y la información sobre todos los detalles de los vuelos.
- El cliente podrá consultar las reservas indicándoles el estado de la reserva.