

Tarea 3

Asignatura: **Desarrollo web en entorno servidor**

Ciclo Superior Desarrollo de Aplicaciones Web.



Condiciones de entrega.

La forma de entregar los ejercicios será en un fichero ZIP con el formato **ApellidosNombreTarea3.zip**. Cada uno de los ficheros deberá funcionar al ejecutarse.

Puntuación.

El boletín está puntuado sobre 10 puntos. En caso de no entregarlo en fecha, hay una prórroga de 3 días en las que se puede entregar con penalización de 1 punto por día. A partir del tercer día la tarea será puntuada con un cero.

El ejercicio debe funcionar cuando se ejecute.

Ejercicios.

WEB MONTECASTELO

OJO: La web debe funcionar. En caso de que no se muestre lo pedido o dé un error, no se puntúa.

Para esta tarea debes importar el script **BDTarea3.sql** adjunto.

Continuaremos trabajando sobre la web que creamos en la Tarea 2.

GENERAL (1 punto)

- Crea los Context necesarios para importar todas las entidades de la base de datos en la web. (0,25 puntos)
- Configura la cadena de conexión `ConexionTarea3` en `appsettings.json` para establecer la conexión a la base de datos (0,25 puntos)
- Se debe poder cambiar de sección tanto con enlaces del menú como botones en la página. (0,5 puntos)

SECCIÓN ASIGNATURAS (2 puntos)


1. Crear el modelo Asignatura, basándote en la tabla de la base de datos (0,5 puntos)
2. Crear el controlador AsignaturaController, con las acciones necesarias Index y Detalles (0,5 puntos)
3. La URL para acceder a todas las asignaturas debe ser **Asignatura/Lista** para mostrar todas las asignaturas y **Asignatura/Detalles/Id_Asignatura** para mostrar una en específica (0,5 puntos)
4. Crear las vistas necesarias para las acciones del controlador AsignaturaController (0,5 puntos)
 - a. Mostrar todas las asignaturas

Asignaturas		
ID	Nombre	Acciones
1	Programación en C#	Detalles
2	Desarrollo de aplicaciones web	Detalles
3	Bases de datos	Detalles
4	Diseño de interfaces	Detalles
5	Programación orientada a objetos	Detalles
6	Redes de computadoras	Detalles
7	Sistemas operativos	Detalles
8	Ingeniería de software	Detalles
9	Inteligencia artificial	Detalles
10	Criptografía	Detalles

- b. Ver una asignatura en detalle

Detalles de la asignatura

Programación en C#



La asignatura de Programación en C# es una materia fundamental en la formación de cualquier estudiante de ingeniería en sistemas o ciencias de la computación, ya que proporciona las bases para el desarrollo de aplicaciones y sistemas de software modernos y robustos. C# es un lenguaje de programación moderno, orientado a objetos y diseñado por Microsoft, que se utiliza ampliamente en el desarrollo de aplicaciones empresariales y de escritorio, así como en el desarrollo de juegos y aplicaciones móviles. La sintaxis de C# es fácil de aprender y su diseño permite una programación más segura y estructurada, lo que lo convierte en una excelente opción para cualquier proyecto de software a gran escala. En la asignatura de Programación en C#, los estudiantes aprenderán los conceptos fundamentales de programación, como la sintaxis básica del lenguaje, el control de flujo, la programación orientada a objetos, la manipulación de datos y la gestión de excepciones. También se enseñarán los conceptos avanzados de programación, como el desarrollo de aplicaciones con interfaces gráficas de usuario (GUI), la programación de eventos, el acceso a bases de datos y la programación en red. Los estudiantes también aprenderán a utilizar las herramientas y tecnologías más importantes para el desarrollo de aplicaciones en C#, como el Entorno de Desarrollo Integrado (IDE) de Visual Studio, el Framework .NET de Microsoft y la plataforma de desarrollo ASP.NET para aplicaciones web. La asignatura de Programación en C# es esencial para cualquier estudiante que desee seguir una carrera en la industria del software, ya que proporciona las habilidades y conocimientos necesarios para desarrollar aplicaciones modernas y eficientes. Además, el lenguaje de programación C# es una de las habilidades más solicitadas por las empresas en la industria de la tecnología, por lo que los estudiantes que dominen esta materia tendrán una gran ventaja en el mercado laboral.

¡Aprende todo sobre esta asignatura!

[Volver a la lista](#)


SECCIÓN ESTUDIANTES (2 puntos)

1. Crear el modelo Estudiante, basándote en la tabla de la base de datos (0,5 puntos)
2. Crear el controlador EstudianteController, con las acciones necesarias Index y Detalles (0,5 puntos)
3. La URL para acceder a los estudiantes debe ser **Estudiante/Lista** para mostrar todos los estudiantes y **Estudiante/Detalles/Id_Estudiante** para mostrar uno en específico (0,5 puntos)
4. Crear las vistas necesarias para las acciones del controlador EstudianteController (0,5 puntos)

- a. Mostrar todos los estudiantes

Estudiantes		
ID	Nombre	Acciones
1	Juan Perez	Detalles
2	Maria Rodriguez	Detalles
3	Carlos Fernandez	Detalles

- b. Ver una estudiante en detalle

Detalles del estudiante		
Maria Rodriguez		
	EDAD DEL ALUMNO: 16	CURSO AL QUE PERTENECE: 2º ESO
	¡Los mejores estudiantes!	
Volver a la lista		

SECCIÓN PROFESORES (2 puntos)

1. Crear el modelo Profesor, basándote en la tabla de la base de datos (0,5 puntos)
2. Crear el controlador ProfesorController, con las acciones necesarias Index y Detalles (0,5 puntos)
3. La URL para acceder a todos los profesores debe ser **Profesor/Lista** para mostrar todos los profesores y **Profesor/Detalles/Id_Profesor** para mostrar uno en específico (0,5 puntos)
4. Crear las vistas necesarias para las acciones del controlador ProfesorController (0,5 puntos)
 - a. Mostrar todos los profesores

Profesores			
Id	Nombre	Materia	
1	Ana Lopez	Matematicas	Detalles
2	Pedro Martinez	Lengua	Detalles
3	María García	Ciencias	Detalles

- b. Ver un profesor en detalle

Detalles de la asignatura

Ana Lopez



EDAD DEL PROFESOR: 32

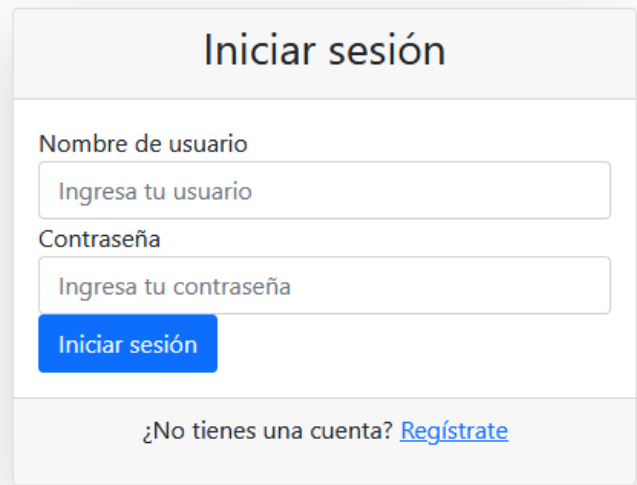
MATERIA QUE IMPARTE EL PROFESOR: Matematicas

¡Increíbles profesores!

[Volver a la lista](#)

SECCIÓN LOGIN (2 puntos)

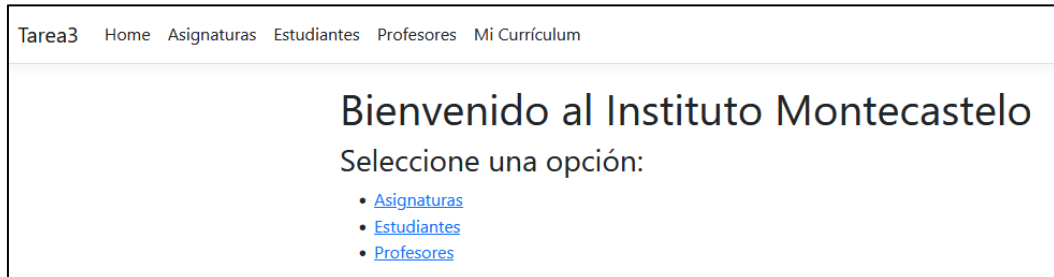
1. Crear el modelo Login, basándote en la tabla de la base de datos (0,5 puntos)
2. Crear el controlador LoginController, con las acciones necesarias Index y Detalles (0,5 puntos)
3. La URL para acceder a la página de Login debe ser **Login/Logueate** (0,5 puntos)
4. El Login debe ser la página por defecto de tu aplicación, y solamente se puede entrar en la aplicación si te logueas. En cuanto te logueas, te debe llevar a la página Home. P.D. No vale escribir la ruta a mano (0,5 puntos)



The image shows a mockup of a login form. At the top, there is a header with the text "Iniciar sesión". Below this, there are two input fields: the first is labeled "Nombre de usuario" and contains the placeholder text "Ingresa tu usuario"; the second is labeled "Contraseña" and contains the placeholder text "Ingresa tu contraseña". Below the password field is a blue button with the text "Iniciar sesión". At the bottom of the form, there is a link that says "¿No tienes una cuenta? [Regístrate](#)".

SECCIÓN HOME (1 punto)

1. A la sección HOME llegaremos después de haber hecho el Login, y aquí podremos seleccionar cualquiera de las otras secciones. (0,5 puntos)



2. La URL para acceder a tu currículum debe ser **Login/MiCurrículum** (0,5 puntos)
3. Debes utilizar la misma vista que usaste en la Tarea 2

