

|  |   |  |              |
|--|---|--|--------------|
| <b>CICLO FORMATIVO DE GRADO SUPERIOR</b><br><b>DESARROLLO DE APLICACIONES WEB</b><br>PROYECTO TRANSVERSAL - CASINO |  <b>EFA</b><br>MORATALAZ<br>Profesores: <b>DC, DGC, JSR</b> | Asignaturas: Todas las técnicas<br>Fecha: 2º Trimestre | <b>Nota:</b> |
| Alumno: _____  |   |  |              |

## PROYECTO TRANSVERSAL – CASINO

### 1. INTRODUCCIÓN

“Un proyecto transversal es una iniciativa que se desarrolla en una organización o institución con el propósito de abordar y resolver problemas, objetivos o necesidades que atraviesan diferentes áreas, departamentos o sectores de esta. La característica principal de un proyecto transversal es que busca integrar la colaboración y coordinación entre múltiples partes interesadas dentro de la organización.”

En nuestro caso, el proyecto transversal abarca las 4 asignaturas técnicas (diseño de interfaces web, desarrollo web en entorno cliente, desarrollo web en entorno servidor y despliegue de aplicaciones web). En el mismo orden en el que están escritas las asignaturas, podemos trazar un proyecto software desde su diseño hasta su despliegue, pasando por todo el desarrollo necesario para aportar la funcionalidad a dicho proyecto.

Este año se nos ha encargado diseñar un casino online **por parejas** que incluya un conjunto de juegos de azar para un único usuario que accede a nuestra aplicación web. Para garantizar que cada equipo diseña casinos diferentes, será necesario elegir:

- Nombre del casino
- Logo e iconos corporativos
- Color corporativo diferenciado en componentes web (menús, botones, etc.)
- Imágenes y otros elementos visuales

El acceso a la aplicación web se realizará mediante un *login*, que comprobará si el usuario y la contraseña introducidos son correctos según la información almacenada en la base de datos. Si no está registrado, será necesario registrarse mediante un formulario de *registro* que incluya tantos campos como sean necesarios. Se valorará positivamente incluir aspectos adicionales (captcha, términos y condiciones, restablecer contraseña, etc.).

Una vez el usuario haya sido verificado, se le va a redirigir a la pantalla principal, donde podremos encontrar:

- Información del casino (nombre, logo, imágenes, etc.)
- Juegos disponibles
- Pie de página con información de contacto del casino, redes sociales, formulario de contacto, posibles patrocinadores, etc.

Los juegos se podrán mostrar en la sección de juegos de forma estática (imagen, nombre del juego y botón para jugar) o de forma dinámica y agrupados en diferentes categorías (por ejemplo, mediante un carrusel). Al pulsar sobre uno de los juegos, se va a redirigir al usuario a la página web de ese juego.

## 2. JUEGOS

A lo largo del curso, en las diferentes asignaturas, hemos ido diseñando juegos de casino que nos podrán ser de utilidad en este proyecto transversal, sin embargo, será necesario diseñar nuevos juegos para mejorar la experiencia del usuario.

### 2.1. Máquina tragaperras y Dino

En el caso de los juegos ya creados, simplemente será necesario incluirlos en este proyecto como una parte de un todo. Al realizar el proyecto en parejas, tenéis la posibilidad de elegir el que más os guste e incluirlo en el casino. Ten en cuenta que, aunque estos juegos ya estén creados, será necesario realizar ciertas mejoras, tanto a nivel de diseño como a nivel de funcionalidad.

A nivel diseño se mejorará aquello que los profesores de la parte de *front-end* os sugieran, mientras que a nivel de funcionalidad debéis adaptar el juego a los conceptos teóricos vistos posteriormente al diseño del juego. Por ejemplo, en la asignatura de desarrollo web en entorno cliente hemos utilizado AJAX, jQuery, JSON, web workers... que ahora incluiremos en juegos como la ruleta o la carrera de galgos. En este caso, una posible funcionalidad que rediseñar es la forma en la que se guardan las tiradas y cómo se realizan peticiones GET() y POST() en la base de datos.

Ten en cuenta que es necesario aplicar el patrón MVC para todo el proyecto, tal y como se ha visto en las asignaturas de cliente y servidor.

### 2.2. Nuevos juegos

Para añadir funcionalidad al casino, se podrá crear **1 juego más o realizar una variación de uno existente**. En este caso, se le otorga al equipo la posibilidad de elegir su propio diseño a la hora de crear los juegos. Se puede coger un juego ya existente y adaptarlo a nuestro casino, o también se puede crear un juego de casino totalmente desde 0. Por supuesto, es altamente recomendable consultar a los profesores si el juego que se quiere incluir puede ser válido dentro del proyecto. Los nuevos juegos podrán ser seleccionados desde la pantalla principal y, una vez dentro, el funcionamiento será el mismo que en los juegos ya existentes. Puedes consultar diferentes juegos y slots de forma gratuita en [Mystake4](#).

### 2.3. ¿Cómo jugar?

En algunos casos es algo complicado aprender un nuevo juego y entender la forma en la que se puede ganar o perder el dinero, por lo que en algunos casinos se ofrece al usuario la posibilidad de ver pequeños videos o tutoriales dentro de la web.

Recuerda que una imagen vale más que mil palabras, y un video más que mil imágenes (más o menos), por lo que el mejor manual de usuario puede estar incrustado dentro del propio proyecto.

### 3. USUARIOS Y ROLES

El usuario es el eje principal de este proyecto software, por lo que será necesario cuidar todos los aspectos relacionados con este. En primer lugar, desde el momento del registro y del inicio de sesión, será necesario almacenar estos datos en la base de datos.

Una vez el usuario entre en la pantalla principal, tendrá la posibilidad de consultar su **monedero** y ver de cuánto dinero dispone. Además, se le dará la posibilidad de depositar más saldo para que pueda seguir jugando.

La forma en la que el usuario extrae el saldo de su cuenta es de libre diseño para el equipo, pudiendo ser desde un simple mensaje o alerta, o incluso la posibilidad de mandar ese dinero a una cuenta bancaria (con sus respectivas comprobaciones).

En cada momento de juego, el usuario debe poder consultar su saldo disponible en el juego en el que esté situado y el saldo total del que dispone en su monedero. En cada juego se podrá añadir más saldo al monedero.

Ten en cuenta que el usuario no debe apostar o jugar si no dispone de saldo suficiente.

Para añadir un componente exclusivo a nuestro casino, se utilizarán distintos **roles** para los usuarios registrados. Contaremos con dos tipos de usuarios:

- **Usuarios free:** Pueden registrarse mediante el formulario de registro existente y sólo tendrán acceso a los juegos de las máquinas tragaperras y del Dino. Del mismo modo, pueden añadir y retirar saldo ficticio para jugar a su gusto.
- **Usuarios vip:** Se registrarán mediante un nuevo formulario de registro que incluirá los campos necesarios para incluir una tarjeta de crédito ficticia o una cuenta bancaria. Tener en cuenta que las tarjetas de crédito deben incluir una verificación de longitud de los campos, la fecha de expiración y el CVC. Por otro lado, la cuenta bancaria debe incluir la verificación de longitud de campos y el propietario de la cuenta bancaria. Estos usuarios tendrán acceso a todos los juegos existentes, es decir, la tragaperras, el Dino y, además, el nuevo juego que se haya desarrollado por el grupo.

Como consideraciones a tener en cuenta, deberás crear una página principal en función del usuario que inicie sesión. En cuanto a la base de datos, será necesario distinguir los usuarios en tablas distintas dependiendo de si son usuarios de pago o no. Al hacer un inicio de sesión, se deberá comprobar en ambas tablas si existe el usuario y devolver qué tipo de usuario es para cargar una página principal u otra.

Además, al registrar a un nuevo usuario, se deberá comprobar que se existe en la base de datos, en ambas tablas, ya que no se permitirá registrar a un usuario de pago si ya existe dicho usuario en la versión gratuito. Con esto, podéis sugerir al usuario un cambio de rol y hacer las modificaciones necesarias para ello.

## 4. ASIGNATURAS

Para facilitar el trabajo, cada profesor indica qué conceptos vistos a lo largo del curso deben estar presentes en este proyecto:

### 4.1. DISEÑO DE INTERFACES WEB

Será necesario crear un paquete de diseños y prototipos de la aplicación web al completo, de forma que se obtenga una estructura sobre la que construir la aplicación web.

En cuanto a aspectos técnicos, será necesario que la aplicación sea **responsive** y que se trabajen todos los conceptos tratados a lo largo del curso en cuanto a diseño de las interfaces web.

### 4.2. DESARROLLO WEB EN ENTORNO CLIENTE

Los lenguajes de programación que se deberán utilizar son HTML5, JavaScript y CSS.

Conceptos que se deben incluir:

- Formularios
- Programación en JavaScript
- Programación de eventos
- Modelo-Vista-Controlador
- Web workers
- JSON
- AJAX
- jQuery

### 4.3. DESARROLLO WEB EN ENTORNO SERVIDOR

Los lenguajes de programación que se deberán utilizar en esta parte serán Spring y Thymeleaf. Adicionalmente, se deberá realizar script para la parte de base de datos:

- Un script para la creación de las tablas y otros objetos.
- Un script para la carga de datos
- Un script para la destrucción de las tablas.

Los conceptos que se deben incluir en esta parte son los vistos durante el curso:

- @Controller
- @RequestMapping
- @RequestParam

- @PathVariable
- @Service
- @Autowired
- @Primary (Opcional)
- @Qualifier (Opcional)
- @Repository
- @PersistenceContext
- Patrón DTO
- Patrón DAO 50% / CrudRepository 50%
- Modelo-Vista-Controlador
- JPA
- Thymeleaf (En la parte de la vista)
- Spring API-REST (Opcional)

#### **4.4. DESPLIEGUE DE APLICACIONES WEB**

Será necesario mostrar cómo desplegar la aplicación en un servidor local o en la nube, tal y como se ha visto en clase.

Para ello, será necesario adjuntar a la entrega un pequeño manual indicando qué pasos se han seguido para el despliegue de la aplicación del casino en el servidor y, si fuera necesario, cualquier consideración necesaria para demostrar cómo se ha llegado a una solución.

## 5. ENTREGA Y EVALUACIÓN

La entrega la realizará sólo uno de los miembros del equipo, incluyendo el nombre y apellido de ambos miembros en el comprimido. Por ejemplo: DanielGonzalez-DanielCuenca.zip

Este proyecto se entregará en la tarea habilitada en el Moodle en la asignatura de desarrollo web en entorno cliente. La fecha de entrega aparecerá en la propia tarea (también se dirá en clase) y es inamovible por cuestiones logísticas y de temporalización de los periodos de exámenes y prácticas.

El proyecto transversal incluye:

- Diseños, bocetos, prototipo y maquetas utilizadas
- Comprimido con el código fuente
- Manual de usuario (si no está incrustado en la propia web)
- Diagrama relacional del modelo de datos.
- Script con el modelo de datos en Oracle
- Script con la carga inicial de datos en Oracle
- Script destructor del modelo de datos en Oracle
- Manual explicativo sobre el despliegue de la aplicación en local y en la nube

Todo lo enumerado anteriormente se entregará en la tarea y, adicionalmente, se realizarán presentaciones para mostrar a los compañeros y al tribunal los proyectos realizados.

El tribunal estará compuesto por los 4 profesores involucrados en las asignaturas de este proyecto transversal y se evaluarán dos aspectos:

- 1) **Presentación y demo (30%)**: claridad de la presentación, exposición de los componentes, demostración de la aplicación web en funcionamiento...
- 2) **Proyecto (70%)**: incluirá ítems como funcionalidades, dificultad desarrollada, conceptos teóricos (MVC, AJAX, jQuery, Spring...)

Cada profesor, por su parte, evaluará todo lo relacionado con su asignatura y, además, evaluarán la presentación realizada ante el tribunal.

Las presentaciones se realizarán durante la última semana del curso en las fechas propuestas por los profesores del proyecto transversal y, del mismo modo que la entrega, las fechas son inamovibles. La duración estimada para una presentación debe ser de unos **15 minutos** y el turno de dudas y consultas del tribunal tendrá una duración de unos 10 minutos como máximo.