**EVALUACIÓN 2**

**ASIGNATURA: DWY4101 – DESARROLLO APLICACIONES WEB Y MOBILE**

**PROFESOR:**

**SECCIÓN:**

|  |  |  |
| --- | --- | --- |
| **PUNTAJE MÁXIMO** | **39 puntos** | **NOTA** |
| **PUNTAJE OBTENIDO** |  |
| **FECHA** |  |
| **DURACIÓN** |  |

**INDICACIONES GENERALES**

La nota 4,0 se obtiene logrando un 60% del puntaje total.

Es una Entrega de Encargo con presentación.

El equipo de trabajo será de a dos o tres (el mismo grupo de la entrega 1).

El equipo de trabajo debe entregar el encargo en la semana 12.

El encargo debe ser presentado por el grupo en la misma semana 12.

El proyecto deberá ser desarrollado en las herramientas Django y Python y entregado en un archivo comprimido que contenga una carpeta con toda la estructura del programa.

**APRENDIZAJES ESPERADOS**

* Crear aplicaciones por el lado del servidor para cumplir con las necesidades del negocio.
* Usar herramientas de desarrollo colaborativo para el desarrollo de aplicaciones en equipos de trabajo.
* Usar herramientas de bases de datos para almacenar y recuperar información de su aplicación web móvil.
* Generar mecanismos de seguridad para proteger los datos privados que los clientes provean a la aplicación.
* Unir el front-end con el back-end

**INSTRUCCIONES**

Los estudiantes deberán utilizar el sitio web construido en la unidad 1 e incluirán la programación por el lado del servidor. Además, deberán agregar control de versiones, pruebas unitarias, gestión de sesiones y el control de acceso de usuarios a la aplicación.

**Enunciado**

En etapas anteriores del Proyecto, desarrollaste junto a tus compañeros un formulario para registrar la información necesaria para el sitio.

En esta evaluación, deberás enlazar el formulario con una base de datos con el fin de lograr almacenar en ella la información utilizando el Framework Django.

Además, deberán implementar otro mantenedor, que permita construir la lógica y relación con el mantenedor del formulario. Se necesita: listar, agregar, modificar y eliminar la información. Adicionalmente, se debe preparar una vista que liste la información utilizando por lo menos dos filtros.

Las galerías deben mostrar las imágenes según lo almacenado en la Base de Datos.

El cliente del sitio solicitado, requiere que la cobertura de las pruebas llegue al 100% por lo que, el equipo, deberá construir 2 casos de prueba con al menos 2 casos de borde para cada uno que involucre los métodos realizados en las funcionalidades que hasta ahora tiene el sitio: registro del formulario y mantenedor del sistema.

Para completar el proceso se utilizarán los mecanismos de sesiones, autenticación y autorización entregados por Django Framework.

El cliente ha indicado que el proceso debe considerar que los usuarios registrados en el sistema deban iniciar sesión antes de realizar una solicitud, de forma de generar un registro por cada usuario del sitio.

Existen dos tipos de usuario: cliente y administrador. Se debe considerar que sólo un usuario de tipo administrador puede manipular los registros del sistema (agregar, modificar y eliminar) y el usuario cliente verá la lista de aquellos registros que se encuentran en estado disponible.

Además, se debe implementar el sistema de recuperación de contraseña para todos los usuarios del sitio (cliente y administrador).

# Observaciones a Considerar:

* Utiliza validaciones HTML5 para el control de ingreso de datos en el formulario
* Utiliza Django para la construcción de las nuevas funcionalidades
* Se debe adjuntar evidencia del respaldo del trabajo en el repositorio remoto
* Utiliza Django unnittest para la construcción y ejecución de los casos de prueba
* Elaboren un informe HTML del resultado de las pruebas que pueda ser revisado fácilmente en el repositorio (Puede ser un archivo README) o en documento word

Rúbrica

|  |  |
| --- | --- |
| Sigla | Leyenda |
| TL | TOTALMENTE LOGRADO |
| ML | MEDIANAMENTE LOGRADO |
| PL | PARCIALMENTE LOGRADO |
| NL | NO LOGRADO |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Aspectos a Evaluar** | **TL**  **3** | **ML**  **2** | **PL**  **1** | **NL**  **0** | **Ptos** |
| **Programa** | | | | | |
| 1. Generar un repositorio GIT para el trabajo en colaboración con su equipo en un mismo proyecto | Genera un repositorio GIT para el trabajo en colaboración con su equipo en un mismo proyecto con su master y 2 branch | Genera un repositorio GIT para el trabajo en colaboración con su equipo en un mismo proyecto con su master y 1 branch | Genera un repositorio GIT para el trabajo en colaboración con su equipo en un mismo proyecto sólo en el master | No genera repositorio |  |
| 1. Realizar Commits y Merge durante el desarrollo de su aplicación | Realiza Commits y Merge durante el desarrollo de su aplicación y deja constancia de un trabajo en el tiempo con más de 3 actualizaciones | Realiza Commits y Merge durante el desarrollo de su aplicación y existen 3 actualizaciones | Realiza Commits y Merge durante el desarrollo de su aplicación y existen 2 actualizaciones | No realiza Commits y Merge durante el desarrollo de su aplicación |  |
| 1. Generar aplicación por el lado del servidor capaz de cumplir con el almacenamiento del formulario | Genera aplicación por el lado del servidor capaz de cumplir con el almacenamiento del formulario | Genera aplicación por el lado del servidor capaz de cumplir con el almacenamiento del formulario pero falta un campo | Genera aplicación por el lado del servidor capaz de cumplir con el almacenamiento del formulario pero faltan dos o tres campos | No logra almacenar el formulario |  |
| 1. Generar aplicación por el lado del servidor capaz de tener un nuevo mantenedor: listar, agregar, modificar su información y eliminar el registro | Genera aplicación por el lado del servidor capaz de tener un nuevo mantenedor: listar, agregar, modificar su información y eliminar el registro | Genera aplicación por el lado del servidor capaz de tener 3 operaciones del mantenedor | Genera aplicación por el lado del servidor capaz de tener 1 o 2 operaciones del mantenedor | No genera operaciones en el mantenedor |  |
| 1. Conectar la base de datos con un servidor web para entregar información a la aplicación que liste la información con sus filtros | Conecta la base de datos con un servidor web para entregar información a la aplicación que liste la información con sus filtros | Conecta la base de datos con un servidor web para entregar información a la aplicación que liste la información pero un filtro no funciona | Conecta la base de datos con un servidor web para entregar información a la aplicación que liste la información pero dos filtros no funcionan | Lista la información pero no logra aplicar filtros |  |
| 1. Las galerías de imágenes muestran los registros según lo almacenado | Las galerías de imágenes muestran los registros según lo almacenado |  |  | No logra mostrar las imágenes almacenadas |  |
| 1. Crear casos de prueba unitarias para validar funcionalidades solicitadas  por el cliente | Crear 2 casos de prueba unitarias con 2 casos de borde para validar funcionalidades solicitadas  por el cliente | Crear 2 casos de prueba unitarias con 1 caso de borde para validar funcionalidades solicitadas  por el cliente | Crea un caso de prueba | No crea casos de prueba |  |
| 1. El sistema permite iniciar sesión para los dos tipos de usuario | El sistema permite iniciar sesión para los dos tipos de usuario desplegando sus opciones según su usuario | El sistema permite iniciar sesión para un tipo de usuario desplegando sus opciones según su usuario | El sistema permite iniciar sesión para los dos tipos de usuario pero no despliega sus opciones según el usuario | No inicia sesión |  |
| 1. Implementar el sistema de recuperación de contraseña para todos los usuarios del sitio | Implementa el sistema de recuperación de contraseña para todos los usuarios del sitio |  |  | No implementa sistema de recuperación de contraseña |  |
| 1. Lograr que el sitio se visualice como se define en el requerimiento y permita navegar accediendo a todas las opciones | Logra que el sitio se visualice como se define en el requerimiento y permite navegar accediendo a todas las opciones | Logra que el sitio se visualice como se define en el requerimiento y con un error en su visualización o navegación | Logra que el sitio se visualice como se define en el requerimiento con dos errores en su visualización o navegación | No logra que le sitio se visualice como se define |  |
| 1. Utilizar buenas prácticas de programación | Utiliza buenas prácticas de programación | Utiliza buenas prácticas de programación con uno o dos errores | Utiliza buenas prácticas de programación con tres errores | No utiliza buenas prácticas (más de tres errores) |  |
| **Disertación** | | | | | |
| 1. Exponer demostrando dominio del tema | Demuestra dominio del tema contestando las preguntas de sus compañeros y el profesor | Demuestra conocimiento del tema pero no contesta las preguntas con precisión | No demuestra conocer bien el tema y contesta pocas preguntas con precisión | No demuestra conocimiento en el tema y no contesta las preguntas con precisión |  |
| 1. Comunicación oral | Su volumen de voz es adecuado, su modulación es pertinente, realiza pausas y utiliza un lenguaje formal | Cumple sólo con 3 aspectos de comunicación oral | Cumple sólo con 2 aspectos de comunicación oral | Su volumen de voz no es adecuado, no modula, no realiza pausas y su lenguaje no es formal |  |

|  |  |
| --- | --- |
| **Puntajes** | **Nota** |
| Puntaje Total: 39 | 7.0 |
| Puntaje: 24 | 4.0 |

Escala de Notas

La siguiente escala permite asociar el puntaje obtenido con una nota. En caso de puntajes con decimales, se debe aproximar al número entero de la siguiente manera:

\* Decimal igual o mayor a 0.5 se aproxima al entero superior.

\* Decimal menor a 0.5 se aproxima al entero inferior.

