

Hito 4 - Manejando perfiles con Django

- Para realizar este desafío debes haber estudiado previamente todo el material disponible en el LMS correspondiente a la unidad.
- Una vez terminado el presente Hito, comprime la carpeta que contiene el desarrollo de los requerimientos solicitados en un archivo .zip y súbelo al sistema LMS.
- Desarrollo desafío: En parejas. Se recomienda utilizar el navegador Google Chrome al acceder a la documentación en inglés, de manera de que el navegador permita traducir automáticamente su contenido.
- Para la realización de este hito necesitarás apoyarte en la documentación de Django y tutoriales complementarios
 - <https://developer.mozilla.org/es/docs/Learn/Server-side/Django/Authentication>
 - <https://docs.djangoproject.com/es/3.2/topics/auth/default/#default-permissions>
 - <https://docs.djangoproject.com/en/3.2/topics/class-based-views/mixins/>
 - <https://docs.djangoproject.com/es/3.2/topics/auth/default/>
 - <https://docs.djangoproject.com/en/3.2/ref/templates/builtins/#if>
 - <https://getbootstrap.com/docs/5.0/examples/>

Habilidades a evaluar

- **Autenticación y autorización de controles de seguridad + Sitio administrativo Django:**
 - Describe las características del modelo de autenticación de Django para el control del acceso al aplicativo.
 - Describe las características del módulo de permisos y autorización disponible en Django para el control del acceso a un recurso.
 - Explica el concepto de Mixins en proyectos Django para control y seguridad de acceso.
 - Implementa el modelo Login/Logout de Django para el control del acceso al aplicativo.
 - Implementa el control de acceso a recursos de vista y de acciones a un aplicativo web utilizando el modelo de autorización y permisos de Django.
 - Reconoce las características básicas del sitio administrativo de Django y su utilidad en el modelo Auth.
 - Implementa el módulo administrativo de Django para la administración de usuarios.
 - Opera el módulo administrativo de Django realizando la administración de usuarios y grupos.

Descripción

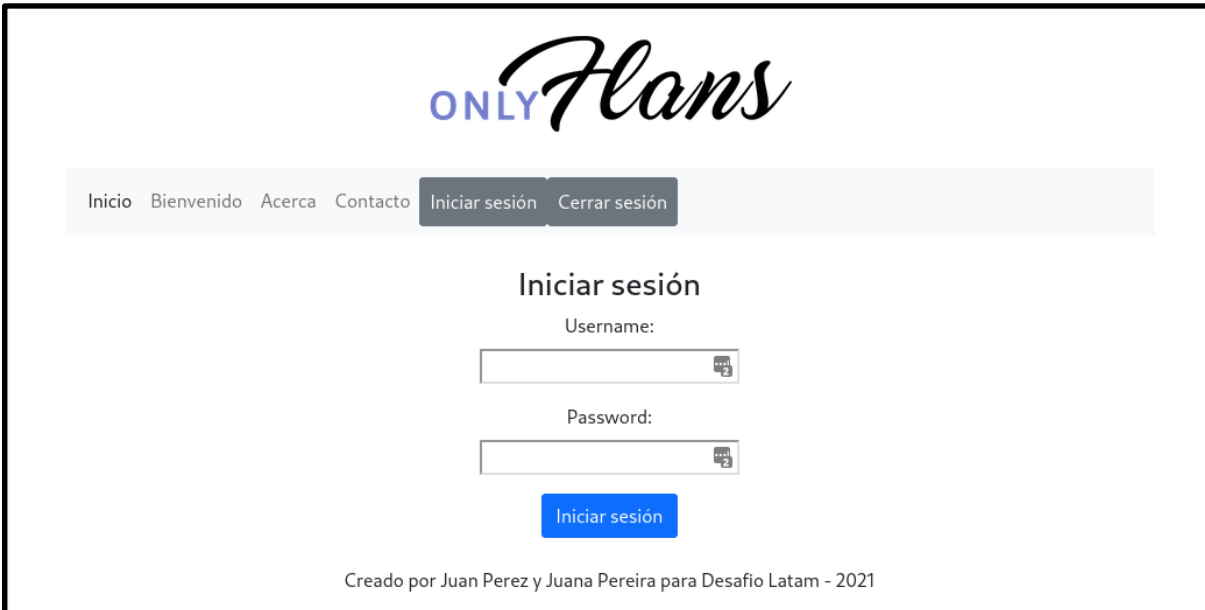
Recapitulemos: en el Hito I pudimos levantar tu proyecto Django desde 0 en un entorno virtual, en el Hito II pudimos agregar contenido personalizado y navegación al sitio web, en el Hito III le añadimos la capacidad de guardar información y permitir su creación tanto a través del panel de administración de Django(en el caso del modelo **Flan**) y a través de formulario personalizado(en el caso del modelo **Contactform**), también aprendimos a filtrar modelos para obtener datos diferenciados en una vista u otra, por ejemplo en la ruta de **Inicio** se muestran todos los **Flan** que tienen configurado **is_private=False** mientras que en la ruta **Bienvenido** se muestran todos los **Flan** que tienen configurado **is_private=True** lo que nos quiere decir que esos productos solo deberían ser mostrados a aquellos usuarios registrados en nuestro sitio web.

El presente hito consiste en añadir la posibilidad de controlar qué usuarios pueden acceder a cierto contenido de la web, de manera de dejar la ruta **Bienvenido** inaccesible para todos aquellos visitantes que no tengan un usuario registrado en nuestro sitio web.

Además de lo anterior, deberemos preparar nuestro sitio web para la presentación que se realizará en el siguiente hito.

Requerimientos

1. Habilita las [url de autenticación de Django](#), agrega las plantillas para [Iniciar sesión](#) y [Cerrar sesión](#) y luego agrega las url de **login** y **logout** como botones en el navbar, realiza las modificaciones necesarias para lograr algo como lo siguiente, puede resultar útil basarse en la plantilla del formulario de contacto creada en el Hito III.



<http://127.0.0.1:8000/accounts/login/>

Una vez realizado lo anterior, ingresa con tus datos de usuario y contraseña creados con anterioridad y presiona el botón **Iniciar sesión**, en el caso de encontrarse con errores, agrega las configuraciones [LOGIN REDIRECT URL](#) y [LOGOUT REDIRECT URL](#) al archivo settings.py de manera de que el usuario sea redirigido a la página de **Bienvenido** cuando inicie sesión y a la página de Inicio cuando cierre sesión.

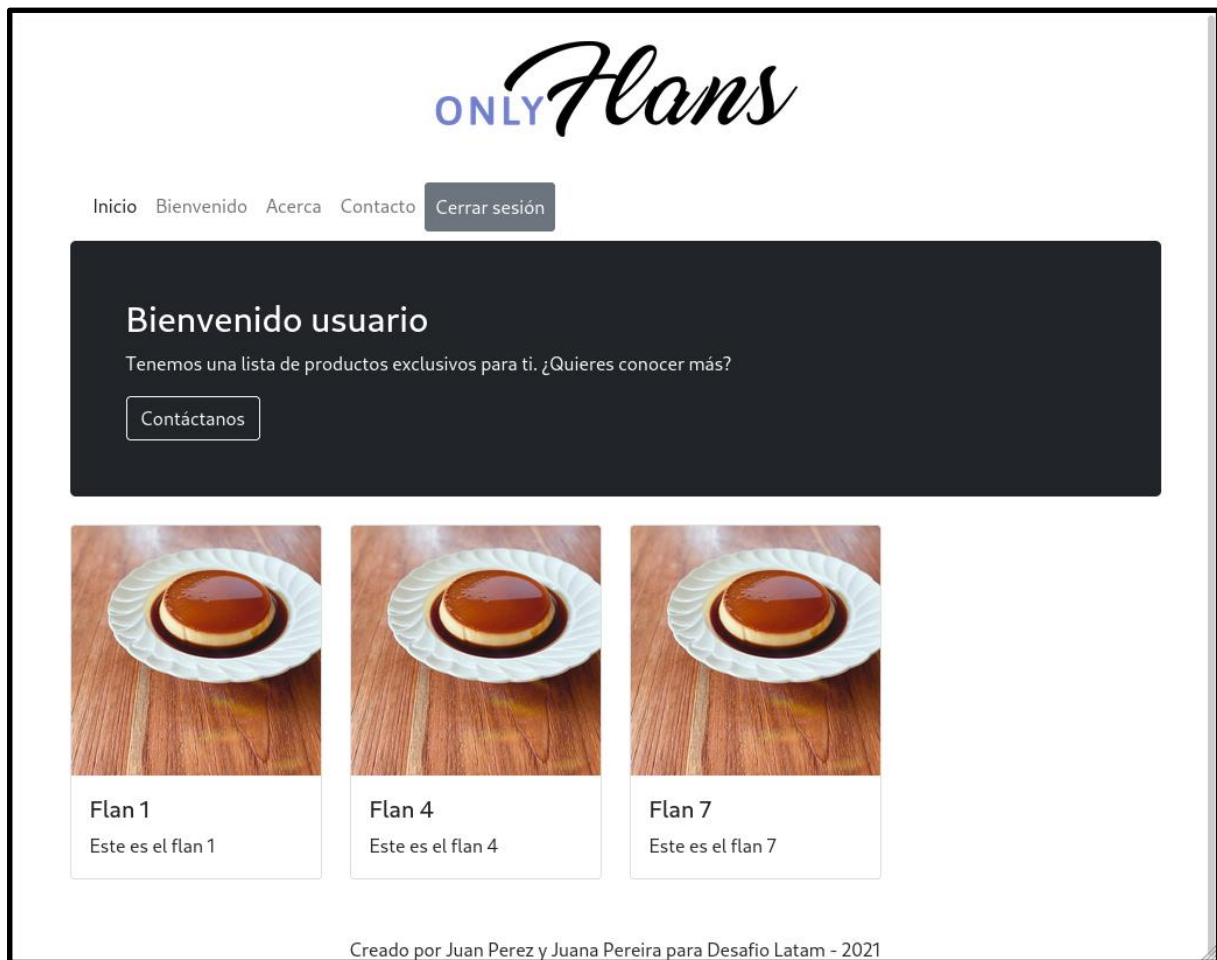
Una vez realizado lo anterior, crea los siguientes “pantallazos” y guárdalos en formato jpg o png, si alguna página se repite, crea ambos pantallazos.

- Página Inicio
- Página Iniciar Sesión
- Página a la que es dirigido el usuario al iniciar
- Página a la que es dirigido el usuario al cerrar sesión

(2 Puntos)

2. Como dijimos en hitos anteriores, la página de **Bienvenida** sólo debe ser accedida por usuarios que se encuentren registrados en el sistema, es por esto que se solicita lo siguiente:
 - Agregar [decorador login_required](#) de manera que no permita a usuarios no registrados acceder a la página de **Bienvenida**
 - Utilizando la [etiqueta {%if%}](#) valida que [un usuario se encuentre autenticado](#) en el sitio web con el fin de conseguir lo siguiente en el navbar:
 - Los enlaces de **Bienvenido** y el de **Cerrar sesión** sólo se muestran cuando un usuario se encuentra autenticado.
 - El enlace de **Iniciar sesión** solo se muestra cuando un usuario no se encuentra autenticado.

Una vez terminado lo anterior, debemos agregar algún elemento de bootstrap adicional que al menos permita diferenciar las vistas de **Inicio** y **Bienvenido**. Es el momento de comenzar a dar más personalidad a tu sitio web, de manera de diferenciarlo previo a su presentación en el Hito V, recuerda que puedes inspirarte por los [ejemplos de bootstrap](#).



Ejemplo de otros elementos en el sitio, <http://127.0.0.1:8000/bienvenido>



Ejemplo de otros elementos en el sitio en formato celular,
<http://127.0.0.1:8000/bienvenido>

Una vez realizado lo anterior, crea los siguientes “pantallazos” y guárdalos en formato jpg o png, si alguna página se repite, crea ambos pantallazos.

- Página Inicio
- Página Bienvenido

(2 Puntos)

3. En el panel de administración de Django, crea 3 usuarios, con username distintos, inicia sesión con ellos en tu web. Adicionalmente, cambia el mensaje de **“Bienvenido usuario”**, por **“Bienvenido <nombreusuario>”** en donde **<nombreusuario>** debe ser el valor correspondiente al nombre de usuario, el cual puede ser obtenido mediante **{{user.get_username}}**.

Luego de realizado lo anterior, crea los siguientes “pantallazos” y guárdalos en formato jpg o png.

- Lista de usuarios creados en el panel de administración de Django
- Página de bienvenida para cada uno de los usuarios

(2 Puntos)

4. Como último punto, debes realizar al menos 2 mejoras que estimes conveniente en el sitio web, con el fin de realizar una exposición en el siguiente hito y final. Estas mejoras pueden ser:
- Agregar un atributo a alguno de los modelos existentes, representando sus valores en el sitio web.
 - Agregar un modelo nuevo y presentar sus datos en alguna vista existente o una nueva vista creada ad-hoc.
 - Agregar una nueva ruta, donde muestres información complementaria pero diferente a la que actualmente tiene el sitio web.

Además de lo anterior, se debe preparar una presentación que permita presentar en un máximo de 15 minutos, al menos los siguientes puntos:

- Descripción general del sitio web
- Características base del sitio web
- Características personalizadas añadidas al sitio web
- Problemas o dificultades con los que se encontraron a la hora de desarrollar el sitio web base
- Problemas o dificultades con los que se encontraron a la hora de añadir características personalizadas.

Dentro de la misma presentación, deberás mostrar el sitio web en funcionamiento, con el fin de demostrar la compleción del proyecto, sus hitos, y facilitar la presentación de los puntos anteriormente descritos.

Luego de realizado lo anterior, crea los pantallazos que creas necesario para demostrar las características y el diseño personalizado, además de incluir un archivo de texto con al menos un resumen de los puntos a presentar.

(4 Puntos)



¡Mucho éxito!