**Cuestionario:**

1. ¿Qué es Django y por qué lo usaríamos?

Django es un software que podemos utilizar para desarrollar aplicaciones web de forma rápida y eficiente.Con Django, puedes llevar tus aplicaciones web desde una idea hasta su lanzamiento en cuestión de horas. Django se encarga de gran parte de las complicaciones del desarrollo web, para poder concentrarte en escribir tu aplicación sin tener que invertir demasiado tiempo. Es gratuito y de código abierto.

Django incluye muchos extras que podes usar para gestionar tareas básicas de desarrollo web.Además, se encarga de la autenticación de usuarios, la administración de contenido, los mapas del sitio, las fuentes RSS.

Django prioriza seguridad y ayuda a los desarrolladores a evitar errores comunes, como la inyección SQL, los scripts entre sitios, la falsificación de solicitudes entre sitios y el clickjacking. Su sistema de autenticación de usuarios proporciona una forma segura de gestionar las cuentas de usuario y las contraseñas.

1. ¿Qué es el patrón MTV (*Model-Template-View*) en django? (simplificado de MVC). Compará MTV con MVC.

La arquitectura Modelo-Vista-Plantilla, también conocida como MVT, es un patrón de diseño de software dentro de Django que separa las capas de una aplicación web en distintos componentes.

El primero es el modelo,que maneja todo lo relacionado con la información, esto incluye cómo acceder a esta, la validación, relación entre los datos y su comportamiento.

Posteriormente, la vista,es un enlace entre el modelo y el template. Decide qué información será mostrada y por cual template.

Por último, el template decide cómo será mostrada la información.

El MTV es una versión específica de MVC. Si bien tienen muchas similitudes,

MVC divide la responsabilidad entre el Modelo, la Vista y el Controlador, mientras que MVT asigna roles diferentes al Modelo, la Vista y la Plantilla. La diferencia más notable es que en MVT, la Vista procesa las solicitudes y se coordina con el Modelo, mientras que en MVC, el Controlador desempeña esa función.

Si vamos a trabajar con Django, el MTV es más recomendable ya que ayuda a un desarrollo más rápido.

1. ¿Qué entendemos por *app* en django?

Una "app" en Django es un módulo o componente autónomo de una aplicación web que encapsula una funcionalidad específica o un conjunto de características relacionadas. Estas están diseñadas para ser altamente modulares,autónomas y reutilizables, además de ser un componente conectable y adaptable a diferentes necesidades.

1. ¿Qué es el flujo *request-response* en django?

El ciclo de solicitud-respuesta de Django es un concepto fundamental que describe cómo el framework web Django procesa las solicitudes HTTP entrantes y genera las respuestas HTTP correspondientes. Este ciclo es el mecanismo principal mediante el cual Django gestiona las interacciones de los clientes y sirve aplicaciones web. A continuación, se explica el ciclo de solicitud-respuesta de Django:

https://clouddevs.com/django/request-response-cycle/#:~:text=Django's%20request%2Dresponse%20cycle%20is,and%20generates%20corresponding%20HTTP%20responses.

1. ¿Qué es el concepto de ORM (*Object-Relational Mapping*)?
2. ¿Qué son los *templates* en django?
3. Leé con detenimiento este *django at a glance*: <https://docs.djangoproject.com/en/5.2/intro/overview/>
4. ¿Cómo se lo instala? <https://docs.djangoproject.com/en/5.2/intro/install/>

Ahora que saben un poco más sobre django, los invito a hacer el siguiente tutorial, hasta la parte 8 inclusive:

* <https://docs.djangoproject.com/en/5.2/intro/tutorial01/>

**Glosario:**

**Fuentes RSS:** formato web que permite a los usuarios recibir actualizaciones de sus sitios web favoritos sin necesidad de visitar cada uno de ellos

**Clickjacking:** Ataque cibernético a un usuario donde se lo engaña para que haga clic en un elemento de una página web que parece inofensiva, pero en realidad está superpuesta a un elemento oculto que realiza una acción maliciosa.