De qué trata el modelo MVT, este modelo arquitectónico es bien similar al MVC y es la estructura qué utiliza Django para poder organizar de mejor manera el desarrollo de aplicaciones web. Este patrón nos ayuda a separar la lógica del negocio, la presentación y el manejo de la interfaz de usuario

Modelo: esta parte se encarga de manejar los datos y la lógica relacionada con ellos, en Django es una representación de la estructura de nuestros datos qué ingresa directamente a las tablas de la base de datos es por esto qué se nos facilita mucho más el manejo de nuestros datos.

Vista: En este apartado, este es el componente qué maneja la lógica de nuestra aplicación y a su vez la encargada de poder procesar las solicitudes qué pide el usuario, también decide qué datos enviarán al usuario y en qué formato lo hará si en HTML,JSON,XML,etc… Para un entendimiento más sencillo, recibe solicitudes, procesa datos y envía respuestas.

Templates: Es la parte qué se encarga de presentarle la información al usuario, estos templates se utilizan para definir cómo se mostrará los datos qué se hayan obtenido en el navegador. estos templates principalmente so un archivo HTML, el qué contiene una mezcla de código estático y dinámico

Hemos decidido utilizar este modelo en el proyecto de “aula virtual” por diversas razones, ya sean por los objetivos qué se tienen y las necesidades qué el usuario presenta.

* Separación de las responsabilidades: este modelo es ideal, ya qué nos permite una separación entre la lógica del negocio y la interacción de los usuarios, nos ayuda en el desarrollo del proyecto ya qué permite qué evolucione independientemente y qué a su vez evoluciones sin qué se vea afectado otras áreas del código.
* Escalabilidad: Nuestro proyecto está orientado a colegios y es fundamental qué en un futuro cuente con mejores funciones para los usuarios, y el modelo ofrece esta flexibilidad al contar con una estructura qué soporta el crecimiento sin perder eficiencia.
* Facilidad gestión de datos: Los modelos MVT son muy buenos para poder gestionar la base de datos y en nuestro proyecto es fundamental qué se pueda acceder y guardar la información de manera correcta, ya sea asistencia, notas o recursos.
* Templates flexibles y dinámicos: este apartado nos permite personalizar la interfaz de usuario de manera flexible y amigable. Esto es muy importante en un sistema como el nuestro ya qué varios usuarios tienen necesidades visuales específicas. Esto nos ayudará a adaptar la información qué se muestre dependiendo el tipo de usuario y sus permisos.