

Plan Formativo Full Stack Python	
Módulo	Desarrollo de Aplicaciones Web con Django
Tema	Sitio open source de apoyo para seguimiento médico
Nivel de Dificultad	Alta
Duración	3.5 horas
Ejecución	Grupal (3 personas)
Intención del aprendizaje o aprendizaje esperado:	
<ul style="list-style-type: none"> • Generación de aplicación de backend para proyecto front-end existente. • Implementación de interfaces de ingreso de datos y visualización de resumen. • Almacenamiento de datos administrados desde formularios y templates Django en archivos JSON. • Implementación de páginas dinámicas con datos provenientes de archivos de datos almacenados. • Implementación de prototipo de Login y separación de contextos de datos dependiendo del usuario. 	
Planteamiento del Problema:	
<p>En esta evaluación usted tendrá la responsabilidad de habilitar en el framerowk Django el sistema para apoyo de seguimiento médico de pacientes, que fué objeto de estudio en la evaluación del módulo 1 de este curso.</p> <ol style="list-style-type: none"> 1. Efectúe una reunión para recapitular los componentes y funcionalidades del sitio diseñado como evaluación del módulo 1. 2. Confeccione un diagrama que permita observar la arquitectura de su aplicación, considerando el desarrollo de front-end ya realizado y cómo se complementará con funcionalidades de lado servidor. 3. Indique cuál sería la estructura de su proyecto Django, en términos de distribución de archivos, de URLs y de aplicaciones. 	

4. Cree el proyecto Django con su correspondiente entorno virtual y configúrelo.
5. Habilite al menos lo siguiente:
 - Cree un template base que contenga los componentes principales que estarán presentes en todas las páginas del sitio, incluyendo barra de navegación, encabezado, pie de página, etc.
 - Página que permita el enrolamiento de nuevos pacientes, cuyos datos sean ingresados a través de un formulario que utilice la clase Form de Django. El listado de pacientes inscritos hasta el momento deberá quedar almacenado en un archivo con formato JSON.
 - No utilice base de datos para este ejercicio.
 - Página que muestre una tabla con el listado actual de pacientes registrados en la plataforma, con sus datos principales.
 - Interfaz para ingreso y visualización de tablas de históricos de exámenes de distinto tipo (Hemograma, perfil lipídico, colesterol, triglicéridos, orina, perfil bioquímico, coagulación y otros). Tabla de datos para cada examen, al menos los aquí mencionados. Usted deberá decidir si es más conveniente ingresar estos datos como parte del archivo JSON del listado de pacientes, o si debe crear otro independiente. Sin embargo, el sistema deberá ser capaz de reconocer la correspondencia entre datos de exámenes y a qué paciente corresponden.
 - Los 4 gráficos chart.js para visualización de datos relevantes del ejercicio original deberán ser alimentados dinámicamente con datos de cada paciente pasados desde python al template correspondiente, utilizando context.
 - Página para ingreso de usuarios relacionados a cada paciente (Familiares o personas a cargo) que podrán visualizar o administrar la información de éste.
 - Las páginas de resumen de pacientes y usuarios deberán permitir

borrar registros.

- Implemente una interfaz de login con formularios Django que solicite nombre de usuario/paciente y contraseña para ingresar al sistema y visualizar solamente los datos correspondientes a su perfil. No utilice aplicaciones de Login de Django para este ejercicio.
- Implemente la página principal privada con los datos actuales, al momento de la carga. Los datos deben ser pasados como context desde los datos almacenados en archivos.

6. Utilice el mismo repositorio anterior para la evolución de su sitio a Django.

7. ¿Qué servidor HTTP cree apropiado para desplegar su aplicación en producción? ¿Cómo se debe realizar el despliegue del proyecto Django?

8. Revise su desarrollo y asegure que consiste en un producto consistente que permita uso real por usuarios del tipo “Friendly Users” (De prueba), con funcionalidades limitadas pero que presten una utilidad real.

9. El repositorio GitHub debe permitir ver claramente la contribución equitativa de todos los miembros del grupo de trabajo.

Recursos Bibliográficos :

[1] Módulo 5 – Contenido 3: **“Implementar formularios en un aplicativo web utilizando Django para el despliegue y procesamiento de la información para dar solución a un requerimiento”**