

Desarrollo Web Full Stack con Python y JavaScript

TRABAJO FINAL CURSO 2020

Enunciado del Problema a Resolver

Una clínica de Optometría necesita un sistema web en Django que le permita gestionar el diagnóstico de sus pacientes y la venta de los productos Ópticos para los mismos. Para ello se requiere:

1. Un sistema con un login de usuario con los siguientes roles:
 - a. Secretaría
 - b. Profesional medico
 - c. Ventas
 - d. Taller
 - e. Gerencia
2. El sistema gestionará tres Modelos esenciales:
 - a. Turnos
 - b. Pedidos
 - c. Pacientes
3. El rol de Secretaría permite agregar, modificar o eliminar los turnos de los Pacientes.
4. Cada Paciente tiene su historial médico (solo el Profesional médico puede agregar Observaciones al historial médico).
5. Cada Profesional médico puede ver el listado de Pacientes filtrando por día, mes o año.
6. El Profesional médico solo puede ver los Pacientes asignados a él.
7. El rol de Ventas puede generar un pedido para el Paciente, donde detalla el Producto que quiere adquirir, el precio (pueden ser más de un producto), un subtotal, tipo de pago (tarjeta de crédito, debido, billetera virtual o efectivo).
 - a. El producto tiene nombre, si está clasificado como Lente tendrá la opción de Lejos/Cerca, Izquierda/Derecha, si incluye Armazón o no.
 - b. Una vez que se genera el pedido queda en estado "Pendiente".
 - c. Luego el rol de Ventas puede cambiar el estado a "Pedido" o mandarlo a "Taller".
8. El rol de Taller solo visualiza la lista de pedidos (con todos los detalles del producto sin los precios). El Taller puede confirmar cambiando el estado del pedido a "Finalizado".
9. Gerencia puede visualizar todos los datos y necesita los siguientes reportes:
 - a. Pacientes que asistieron a los turnos en la semana/mes.
 - b. Pacientes que no asistieron a los turnos en la semana/mes.
 - c. Pacientes que hicieron por lo menos un Pedido en la semana/mes.
 - d. Productos más vendidos en el mes.
 - e. Ventas totales por mes clasificadas por Vendedores.

Requisitos Obligatorios

1. Se deben cumplir cada uno de los puntos definidos en el enunciado para la aprobación del trabajo.
2. El sistema debe estar implementado en Django.
3. El sistema debe implementar Bootstrap.
4. El frontend se deja a libre albedrío, pero debe poder visualizarse correctamente en dispositivos móviles (responsivo).
5. Se pueden agregar todos los Modelos adicionales que se necesiten.
6. Se debe implementar sobre base de datos SQLite.
7. Se debe publicar un video de YouTube que demuestre el funcionamiento de todo el sistema.
 - a. El video debe tener en el título el nombre del curso completo.
 - b. En la descripción del video debe estar el nombre del curso, el nombre de la persona que presenta y su número de documento con el que se registró al curso.
 - c. En la descripción del video tiene que estar anotado el tiempo (minuto y segundo) de cada uno de los (9) puntos resueltos en el enunciado, mostrando su funcionalidad y la cumplimentación de los requerido.
8. Se debe publicar un enlace con el código fuente publicado en GitHub con el usuario de la persona registrada en el curso.
9. Se debe adjuntar el código fuente comprimido en .zip

Cualquier falla en cumplimentar uno de los requisitos o puntos del enunciado **derivará en la desaprobarción** del Trabajo Final sin posibilidad a recuperación. Es condición necesaria para la certificación de Desarrollador Full Stack con Python y JavaScript la aprobación del presente trabajo.

Enlace del Formulario de Presentación

Todos los inscriptos, sin distinción, deberán presentar el Trabajo Final 2020 por medio del siguiente formulario <https://forms.gle/yaBkU1RtDTzCpS1u8>