

Zalovich Dario

Table with student information: Carrera (WE), Turno (N), Semestre (3), Grupo (A), ID (68168), Día límite (05/07/2024), Hora límite (14:00), Páginas incluidos los anexos (1). Includes a signature and a risk level table (Seguro, Riesgoso).

Advertencia: El obligatorio online no puede contener archivos con fecha y hora de subida posterior a la indicada en el cabezal de este documento.
Lea atentamente las pautas. Deberá inscribirse a la instancia de evaluación de este obligatorio, recomendamos lo realice con, al menos, 24 h de anticipación a la fecha límite.
Modo de entrega: Se habilitará en https://gestion.ort.edu.uy/ un espacio para subir un único archivo (.zip .rar o .pdf) que contenga todos los documentos generados para esta instancia.
Instructivo de modalidad de entrega online disponible en https://aulas.ort.edu.uy/ (FCD Bedelía de Diseño)
Ante cualquier dificultad técnica al momento de la entrega online, comuníquese con: aulas_dis@ort.edu.uy detallando el motivo y sus datos de contacto.

Desarrollo de aplicación web Fullstack

Objetivo
Desarrollar el diseño y prototipo funcional de una aplicación de backoffice en la que un usuario gestione un listado de elementos.
La temática quedará a elección del alumno previa aprobación por parte del docente.
Se podrá modificar la idea principal de la aplicación cubriendo los mismos conceptos y con previa aprobación del docente.

Proceso y funcionamiento
La aplicación debe conectarse mediante Javascript al backend creado.
La temática de la aplicación quedará a criterio del alumno; a modo de ejemplo se podría trabajar con un listado de mascotas de una veterinaria, donde se gestiona el perfil de cada mascota ingresada en el sistema.
Se deberá registrar usuarios y autenticar por medio de un login. Una vez autenticados, se accederá a una pantalla en la que se listarán todos los elementos y se podrá ver la ampliación de cada uno.
Se deberá poder agregar nuevos elementos, editar y eliminar los existentes.
La aplicación deberá estar desarrollada con Javascript utilizando fetch para la comunicación con el servidor.
El backend deberá estar desarrollado con NodeJS, Express, Mongoose, con una base de datos MongoDB alojada en MongoAtlas.

- Se valorará:
- Nivel gráfico de la propuesta.
- El realismo de la aplicación.
- La complejidad que se pueda aportar a la propuesta base.
- El uso creativo de la información obtenida.
- El desarrollo de la programación de acuerdo con los criterios vistos en clase.
- La publicación remota de la aplicación.

- Producto
El producto final deberá ser una
• Una aplicación web (Front end)
• Una API + base de datos (backend)

Se deberá entregar:

- Documentación de la aplicación
 - Idea y descripción de la aplicación
 - Listado de funcionalidades detalladas
 - Prototipo del diseño (diseño de pantallas, pdf o figma)
 - URL de del front y backend publicados
- Video de respaldo de la aplicación funcionando
- Carpeta de la aplicación con estructura
/NOMBREPROYECTO_NOMBREALUMNO
/FRONTEND
/BACKEND

IMPORTANTE: Los ejercicios son individuales, dos alumnos no pueden plantear una solución igual o similar, en el caso de que esto sucediera el valor de la entrega será 0 (cero). El alumno es responsable por la integridad de los datos provistos.



Manejo de Imágenes

Nivel 1: Puedes utilizar imágenes producidas por terceros, pero siempre debes citar la fuente e incluir los créditos.
