

# Desarrollo de Software 2024 (1er. Cuat)

[Página Principal](#) / [Mis cursos](#) / [DDS 2024 Q1](#) / [Recuperatorio](#) / [Recuperatorio 3K1](#)

## Recuperatorio 3K1

✓ **Hecho:** Ver

**Por hacer:** Hacer un envío

**Apertura:** jueves, 4 de julio de 2024, 10:40

**Cierre:** jueves, 4 de julio de 2024, 15:00



**DESARROLLO DE SOFTWARE**  
**Evaluación Sumativa**

### Instrucciones:

1. Descargar el archivo RecParcial\_DDS\_3k1.zip (adjunto más abajo)
2. Descomprimir el archivo RecParcial\_DDS\_3k1.zip en la carpeta seleccionada para realizar el parcial
3. Programar la funcionalidad descrita más adelante con el título **Consigna**.  
**Importante:** debe instalar las librerías dependientes del back y del front:  
Ejecutar el comando `npm install` en ambos proyectos (misma ruta donde está el archivo package.json).
4. Al finalizar el desarrollo, generar un unico archivo zip que contenga las carpetas backend y el fronted, excluyendo las carpetas node\_modules de ambos proyectos.  
El archivo zip tiene que nombrarse con el siguiente patrón: `<apellido>_<nombre>_<legajo>.zip``  
Por ejemplo, si te llamas `Juan Garcia Saravia` y tu legajo es `45044`, el archivo que debés subir será: `garcia_saravia_juan_45044.zip`  
Este zip generado debe se entregado como respuesta de esta tarea.

### Consigna:

Desarrollar el **backend** en el port 4000 y el **frontend** en el port 3000.

**Desarrollo del backend:** Se deberá agregar una funcionalidad de consulta para un recurso nuevo "Operadores":

- Desarrollar el código para conectar con la BD, crear la tabla si no existe e insertar los valores asignados. En el archivo `script_operadores.sql` (que está en la raíz del proyecto) están los script sql necesarios.
- Crear el modelo de sequelize que mapee la tabla Operadores, acorde a la estructura descrita en el punto anterior.
- Programar una ruta webapi con las siguientes características:
  - Un router para el endpoint "operadores" que mediante el verbo/método GET reciba un parámetro: "NombreComercial" y utilizando el ORM del punto anterior, recupere los registros filtrando por el parámetro recibido. Deberá devolver todos los registros en cuyo campo "NombreComercial" contenga el valor del parámetro recibido. Devolver todos los campos de la tabla.
  - La aplicación deberá registrar como middleware el router con la ruta `/api/operadores`.
- Asegure la funcionalidad probando las siguientes url desde el browser:
  - `http://localhost:4000/api/operadores`  
Debe devolver todos los registros.
  - `http://localhost:4000/api/operadores?NombreComercial=COM`  
Debe devolver los registros que contengan "COM" en el NombreComercial

**Desarrollo del frontend:** Se deberá agregar una interface de consulta para el recurso “operadores” desarrollado en el backend

- Desarrollar los componentes con el siguiente esquema:



- Agregar el código necesario para poder realizar la consulta de operadores. Se deberá realizar los componentes de react y consumir el backend desarrollado en el punto anterior.
- En el componente Operadores, incluir la interface de búsqueda con el campo a filtrar (“NombreComercial”) y el botón buscar.

Deberá cumplir los siguientes requisitos:

- - Consumir vía axios el endpoint del back.
  - Utilizar para el formulario de búsqueda la librería react-hook-form
- En el componente OperadoresListado incluir una tabla de html para mostrar los resultados de la búsqueda, incluya en la misma todos los campos. (no se pide paginación). Los resultados deben estar ordenados por el contenido del campo “NombreComercial”
- Agregar al componente menú de la aplicación, el acceso al nuevo componente.

*Evaluacion:*

Consigna	Puntaje
<b>BackEnd</b>	
Conectar Db, crear tabla ...	10
Modelo Sequelize	10
EndPoint GET (*)	20
Filtrar con parámetro	10
<b>FrontEnd</b>	
Componente Operadores	10
Componente OperadoresListado (*)	20
Filtrar con parámetro	10
Componente Menu	10

Total

(\*) Estos puntos son requisitos mínimos para aprobar.

Escala de notas

NOTA	PORCENTAJE	CALIFICACIÓN
1		No Aprobado
2		No Aprobado
3		No Aprobado
4	55% a 57%	Aprobado
5	58% a 59%	Aprobado
6	60% a 68%	Aprobado
7	69% a 77%	Aprobado
8	78% a 86%	Aprobado
9	87% a 95%	Aprobado
10	96% a 100%	Aprobado

Tiempo de resolución: 1h 30 minutos.

 [RecParcial\\_DDS\\_3k1.zip](#) +

4 de julio de 2024, 09:04

Agregar entrega

Estado de la entrega

Estado de la entrega	Todavía no se han realizado envíos
Estado de la calificación	No calificada
Tiempo restante	27 minutos 25 segundos restante

Criterios de calificación

Ejecución	No despliega nada o no ejecuta o se bloquea <b>0 puntos</b>	Despliega pero no tiene en cuenta los tests unitario <b>1 puntos</b>	Despliega y tiene en cuenta los tests unitarios pero presenta errores <b>2 puntos</b>	Despliega, tiene en cuenta los tests y funciona correctamente <b>3 puntos</b>
Acceso a Datos con Sequelize	No presenta código de acceso a datos <b>0 puntos</b>	Presenta código de acceso a datos pero sin filtros o el código presentado tiene errores <b>1 puntos</b>	Presenta código de acceso a datos e implementa los filtros pero no funciona <b>2 puntos</b>	Presenta código de acceso a datos e implementa los filtros correctamente <b>3 puntos</b>

Endpoints	No presenta código de implementación de endpoints <b>0 puntos</b>	Presenta código de implementación de endpoints pero no para los filtros o está mal implementado <b>1 puntos</b>	Presenta código de implementación de endpoints incluidos los filtros pero no funciona <b>2 puntos</b>	Presenta código de implementación de endpoints incluidos los filtros funcionando correctamente <b>3 puntos</b>
Tests Unitarios	No tiene en cuenta los tests unitarios <b>0 puntos</b>	Tiene en cuenta los tests unitarios acordados pero no todos o tienen errores graves <b>1 puntos</b>	Tiene en cuenta todos tests unitarios acordados pero no pasan o no son correctos <b>2 puntos</b>	Tiene en cuenta todos los tests unitarios acordados, son correctos y pasan todos <b>3 puntos</b>
Implementación de componentes Formulario y Listado	No implementa componentes de React <b>0 puntos</b>	Implementa alguna idea de componente pero incompleta o sin lineamientos de React <b>1 puntos</b>	Implementan sólo un componente o ambos pero con errores <b>2 puntos</b>	Implementa ambos componentes correctamente <b>3 puntos</b>
Acceso al Backend con Axios	No implementa el acceso al backend <b>0 puntos</b>	Implementa el acceso al backend pero no usa Axios <b>1 puntos</b>	Implementa acceso al backend con Axios pero está incompleto o tiene errores <b>2 puntos</b>	Implementa correctamente el acceso al backend con Axios <b>3 puntos</b>
Renderización de resultados	No renderiza los resultados <b>0 puntos</b>	Renderiza sólo resultados iniciales <b>1 puntos</b>	Renderiza resultados correctamente y maneja el evento al filtrar <b>2 puntos</b>	
Apreciación general del parcial	Entrega muy pobre o incompleta <b>0 puntos</b>	Entrega medianamente completa y correcta pero con errores <b>2 puntos</b>	Entrega completa y correcta <b>4 puntos</b>	

[◀ Trabajo Práctico Integrador - Entrega 2](#)

Ir a...

 [Contactar con el soporte del sitio](#)

Usted se ha identificado como Abigail Sara Canaan (Cerrar sesión)  
DDS 2024 Q1

- Autogestión
- UTN Facultad Córdoba
- WebMail Alumnos
- Busqueda Biblioteca Central
- Español - Internacional (es)
- Deutsch (de)
- English (en)
- Español - Internacional (es)

[Resumen de retención de datos](#)

[Descargar la app para dispositivos móviles](#)