



Documento de análisis: Mantenedor de tutoriales

CONFIDENCIAL

Información de uso confidencial
y exclusivo para sus
destinatarios. Si usted ha
recibido esta comunicación
por error, por favor
notifíquelo.

Contenido

Documento de análisis:	0
Mantenedor de tutoriales.....	0
Evolución del documento	2
Introducción	3
Problemática	3
Análisis.....	3
Solución.....	4
Home	4
Agregar Tutorial.	6
Ver detalle de Tutorial.	8
Modificar Tutorial.....	9

Evolución del documento

Versión	NºRev.	Preparado	Revisado	Fecha	Observaciones
1.0	1	Juan Moran		10/09/2020	<i>Creación</i>

Introducción

A continuación, se describen las funcionalidades para crear el mantenedor de tutoriales.

Problemática

El área de servicio al cliente, requiere disponibilizar los tutoriales, los cuales varían de acuerdo al tipo de cliente

Actualmente los clientes ingresan al portal, y en su perfil visualizan los tutoriales disponibles, si el encargado de cliente necesita cambiar esta información, tiene que levantar un requerimiento para que un técnico modifique los tutoriales directamente en base de datos.

Actualmente el 38% de los requerimientos son solicitudes de parametrización de datos por parte del área de servicio al cliente, por lo cual la gerencia solicita la ayuda del área de tecnología, para mejorar este proceso.

Análisis

Se creará un mantenedor de tutoriales, que permita a los encargados de cliente configurar los tutoriales disponibles.

La entidad Tutorial se compone de los siguientes campos:

- Descripción, input de texto. Obligatorio.
- Profesor, input de texto. Obligatorio.
- Materia, input de texto. Obligatorio.
- Fecha, datetime, Obligatorio.

El back-end está resuelto ya que en un requerimiento anterior se desarrollaron los endpoint que afectan a la tabla maestra de tutoriales. Los endpoint a utilizar se encuentran en el archivo adjunto **rayentutorialtestapi.postman_collection**

Por lo tanto, solo se debe desarrollar el Front-end y reutilizar los endpoint antes mencionados.

El desarrollador debe ocupar la tecnología de Fron-end que más domine.

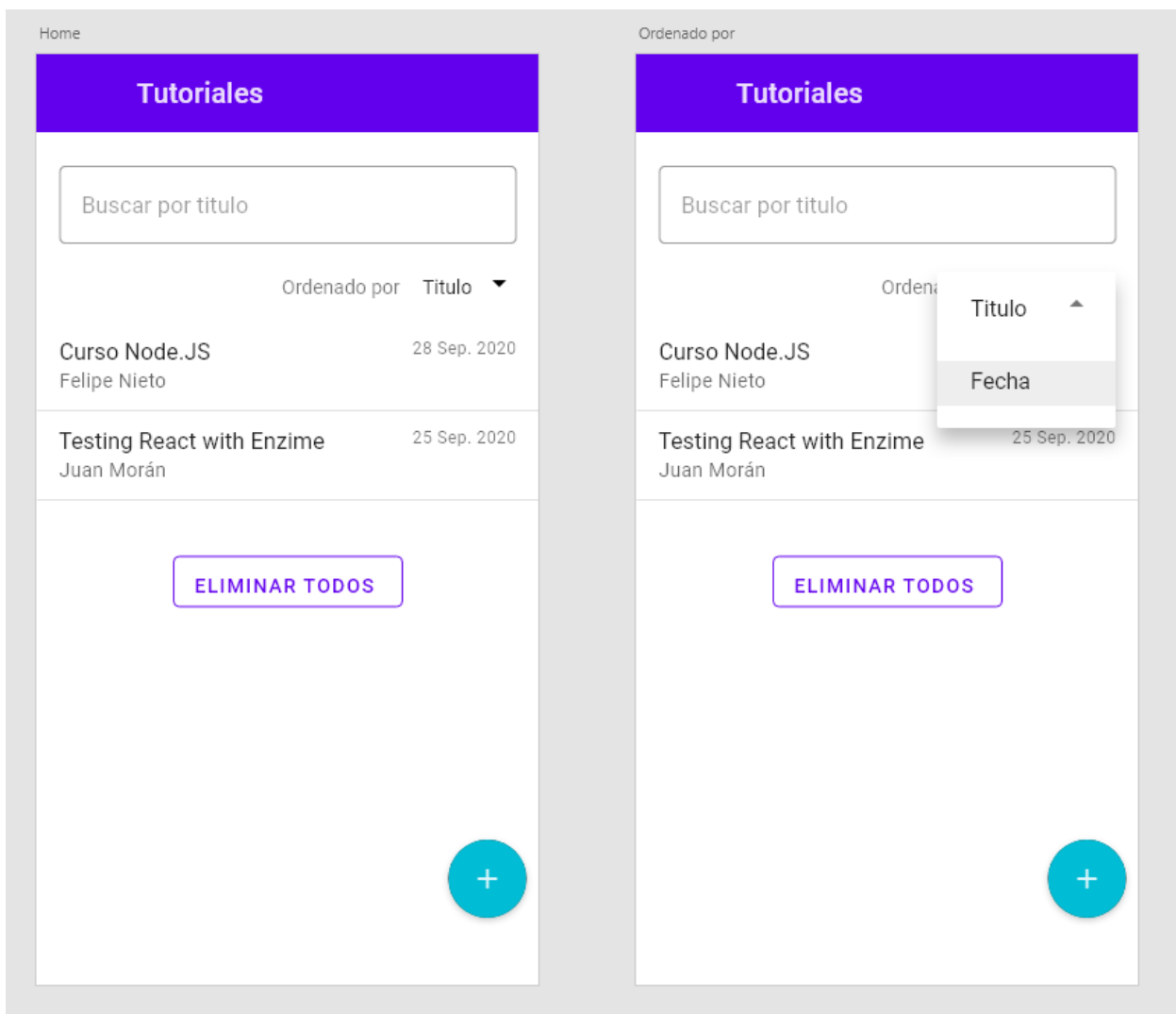
Se pueden basar en el siguiente prototipo:

<https://xd.adobe.com/view/6e84aabe-2dee-47bb-b931-99d8ed66ab67-3bd7/>

Este requerimiento se encuentra estimado en 4 horas contar de la indicación del encargado. Al terminar el plazo de debe enviar el proyecto con la solución al correo cmmercado@rayensalud.com

Solución

Wireframe



Home

Al iniciar la aplicación se debe consultar por los tutoriales ingresados, para esto se debe consultar el endpoint que obtiene todos los tutoriales al llamarlo sin parámetros.

GET <https://rayentutorialtestapp.azurewebsites.net/tutorials>

El cual retorna un arreglo de tutoriales

```
[
  {
    "id": 4,
    "nombre": "Testing React with Enzyme",
    "profesor": "Felipe Nieto",
    "materia": "Desarrollo",
    "fecha": "2020-09-18T00:00:00.000Z",
    "materia_id": null
  },
  {
    "id": 5,
    "nombre": "Curso Node.JS",
    "profesor": "Juan Morán y Felipe Nieto",
    "materia": "Programación",
    "fecha": "2020-09-14T00:00:00.000Z",
    "materia_id": null
  }
]
```

Se debe ignorar el campo *materia_id* para este desarrollo.

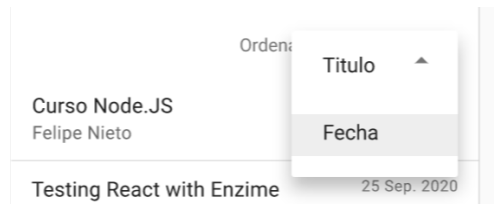
En esta pantalla se debe mostrar una app bar con el título de la aplicación 'Tutoriales',

Además, se debe agregar un campo de búsqueda, que permita acotar la lista de tutoriales. Al ingresar datos en este campo y presionar el botón buscar, se debe consultar el endpoint que obtiene los tutoriales que contienen parte de la descripción recibida. La búsqueda es case-sensitive, por lo que las mayúsculas/minúsculas si son tomadas en cuenta por el endpoint.

GET https://rayentutorialtestapp.azurewebsites.net/tutorial?description=React

```
[
  {
    "id": 4,
    "nombre": "Testing React with Enzyme",
    "profesor": "Felipe Nieto",
    "materia": "Desarrollo",
    "fecha": "2020-09-18T00:00:00.000Z",
    "materia_id": null
  }
]
```

La lista por defecto debe estar ordenada por orden alfabético, pero éste se puede cambiar para que se ordene por fecha. El ordenamiento se hace en el front-end, así que el backend entregará la lista en cualquier orden.



A continuación, se debe mostrar el listado de tutoriales, mostrando como texto primario el título, como secundario el nombre del profesor y la fecha cuando se dictará en el extremo superior derecho.

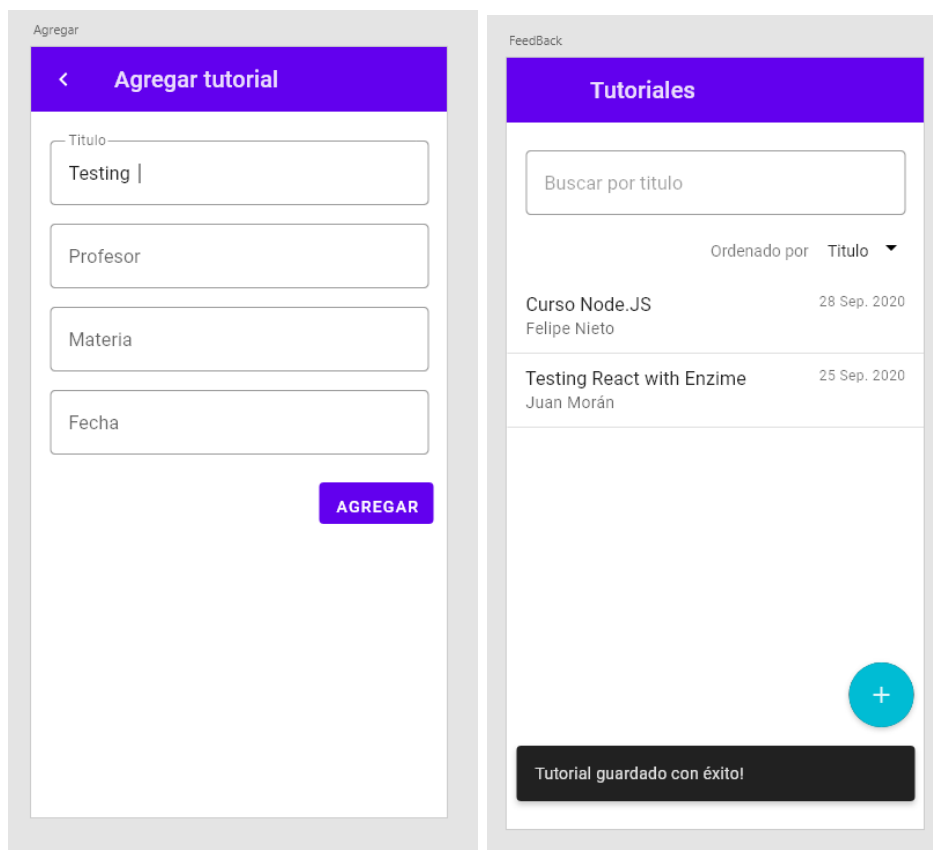
A continuación de la lista se debe agregar el botón 'Eliminar todo' el cual debe quitar los elementos de la lista para esto se debe ocupar el endpoint `deletetutorials`.

DELETE <https://rayentutorialtestapp.azurewebsites.net/deletetutorials>

En el extremo inferior derecho se debe mostrar el botón de acción principal con el icono que represente la acción de agregar.

Agregar Tutorial.

Wireframe



Al hacer clic sobre el botón de acción principal se debe mostrar la pantalla para agregar los tutoriales.

El texto de la app bar debe ser 'Agregar Tutorial' además este debe tener la opción de volver al home de la aplicación.

En el cuerpo se debe mostrar los siguientes inputs:

- Descripción, input de texto. Obligatorio.
- Profesor, input de texto. Obligatorio.
- Materia, input de texto. Obligatorio.
- Fecha, datetime, Obligatorio.

A continuación, se debe adicionar el botón Agregar. Al hacer click se debe enviar al backend la información del tutorial, para esto se debe ocupar el endpoint que crea un tutorial.

POST <https://rayentutorialtestapp.azurewebsites.net/createtutorial>

Al cual se le entrega el nuevo tutorial

```
{
  "nombre": "Advanced React with Hooks",
  "profesor": "Felipe Nieto",
  "materia": "Desarrollo",
  "fecha": "2020-09-21 09:59:00"
}
```

Y devuelve el tutorial ingresado con su id asignado

```
{
  "id": 6,
  "nombre": "Advanced React with Hooks",
  "profesor": "Felipe Nieto",
  "materia": "Desarrollo",
  "fecha": "2020-09-21T00:00:00.000Z",
  "materia_id": null
}
```

Además, si la transacción es exitosa se debe retornar al home de la aplicación, de no tener éxito se debe quedar en la pantalla actual, en ambos casos se debe notificar al usuario del resultado.

Ver detalle de Tutorial.

Wireframe

Ver

< Detalle Tutorial ✎

Titulo
Curso Node.JS

Profesor
Felipe nieto

Materia
Desarrollo

Fecha
28 - 09 - 2020

En el home de la aplicación al hacer clic sobre un elemento de la lista de tutoriales, se debe mostrar toda la información ingresada del tutorial.


El texto de la app bar debe ser 'Detalle Tutorial' además este debe tener la opción de volver al home de la aplicación y editar el tutorial.

En el contenido se debe mostrar los valores precargados para los campos y estos deben estar deshabilitados.

De presionar la opción para modificar (botón del lápiz) debe mostrar en la pantalla Modificación de tutorial.

Modificar Tutorial.

. Wireframe



Editar

< **Modificar Tutorial**

Titulo
Curso Node.JS

Profesor
Felipe nieto

Materia
Desarrollo

Fecha
28 - 09 - 2020

ELIMINAR MODIFICAR

El texto de la app bar debe ser 'Modificar Tutorial' además este debe tener la opción de volver a la pantalla de detalle.

En el contenido se debe mostrar los valores precargados para los campos y estos deben estar habilitados.

A continuación de los campos se debe colocar el botón Guardar, el cual deberá consumir el endpoint correspondiente para actualizar la información del tutorial.

PUT <https://rayentutorialtestapp.azurewebsites.net/updatetutorial/6>

```
{  
  "nombre": "Advanced React with Hooks",  
  "profesor": "Juan Morán",  
  "materia": "Desarrollo",  
  "fecha": "2020-09-21 09:59:00"  
}
```

De tener éxito la transacción se debe volver a la pantalla de detalle, si no se debe permanecer en la misma pantalla y notificar el error.

Alineado con el botón Guardar, pero en el extremo izquierdo, se debe agregar el botón Eliminar, el cual eliminara el tutorial seleccionado, al realizar esta acción se debe volver al home de la aplicación, para realizar esta acción se debe utilizar el endpoint deletetutorial.

DELETE <https://rayentutorialtestapp.azurewebsites.net/deletetutorial/6>