


Sesión # 9 Componente Práctico

Consumiendo un servicio Web - Detección del estado de red

1. Para el siguiente componente práctico usted como instructor debe guiar al tripulante en el desarrollo y escritura del código.
2. Lea el enunciado del tripulante
3. Revise el repositorio https://github.com/misionticmovil/red_egresados . Recuerde que este repositorio ya cuenta con todo desarrollado, se encuentra dividido por ramas (una por tema, donde se indica también la sesión) y es la base para que usted como instructor le indique a los tripulantes qué realizar. **ESTE ES EL REPOSITORIO DE INSTRUCTORES (GUIA SOLUCIÓN)**
4. La solución de esta sesión se encuentra en la rama s9-network-state del repositorio punto anterior.
5. Para realizar el componente práctico es necesario que usted haga uso del [template](#) (el mismo usado por los estudiantes), y durante el avance del componente práctico proceda a resolver dudas y explicar el funcionamiento del código. **ESTE ES EL REPOSITORIO PARA QUE UTILICEN COMO TEMPLATE INSTRUCTORES Y TRIPULANTES - UTILICE LA RAMA S9-NETWORK-STATE**
6. Explique lo que se planea hacer en el enunciado.
 - a. Regístrese en la API: <https://misiontic-2022-uninorte.herokuapp.com/login>
Nota: Utilice su correo uninorte



- b. Confirme su correo electrónico (luego del registro, llegará un mensaje a su cuenta uninorte)
- c. Acceda en el menú izquierdo a la opción 'API KEY' y haga clic en el botón Generar



Endpoints

API Key

Cerrar Sesión



Ofertas de Trabajo - Work Pool

GET /jobs?cat={CATEGORY}&limit={LIMIT}

API KEY: Tu ApiKey debe enviarse como parte de Headers

CATEGORY: Categoría de las ofertas de trabajo:
android | **aws** | **blockchain** | **database** | **devops** | **frontend**. Default: Todas.

LIMIT: Cantidad máxima de ofertas de trabajo a obtener. Min: 2, Max: 10, Default: 5.

Response:

```
{
  "jobs": [
    {
      "title": "Job 1",
      "description": "Lorem ipsum dolor sit amet, consectetur adipiscing.",
      "payment": 6500000,
      "category": "devops",
      "experience": "Senior"
    },
    {
      "title": "Job n",
      "description": "Lorem ipsum dolor sit amet, consectetur adipiscing.",
      "payment": 2500000,
      "category": "android",
      "experience": "Junior"
    }
  ]
}
```



Endpoints

API Key

Clave API

Es necesario que cuentas con una clave API para hacer consumo de los endpoints del servicio web.



GENERAR

- d. Copie la clave generada y guárdela
- e. Implemente el método factory en **public_job.dart**. Conversión JSON a Objeto

```
factory PublicJob.fromJson(Map<String, dynamic> map) {
  return PublicJob(
    title: map['title'],
    description: map['description'],
    payment: map['payment'],
    category: map['category'],
    experience: map['experience']);
}
```

- f. En el archivo **work_pool.dart** debe añadir el siguiente código y pedir a los estudiantes replicarlo:

- **Variables de conexión al web service**

```
final String baseUrl =
'misioctic-2022-uninorte.herokuapp.com'; //URL DEL SERVICIO
final String apiKey = SU KEY COPIADA
```

- **Uri para hacer las peticiones**

```
var queryParameters = {'limit': limit.toString()};
```

```
var uri = Uri.https(baseUrl, '/jobs', queryParameters);
```

- **Solicitud**

```
final response = await http.get(  
  uri,  
  headers: <String, String>{  
    'Content-Type': 'application/json; charset=UTF-8',  
    // We add our service ApiKey to the request headers  
    'key': apiKey  
  },  
);
```

- g. En el archivo **connectivity.dart** debe añadir el siguiente código y pedir a los estudiantes replicarlo:

Actualización de estado de conectividad

```
set connectivity(ConnectivityResult connectivity) {  
  _connected.value = connectivity != ConnectivityResult.none;  
}
```

- h. En el archivo **app.dart** debe añadir el siguiente código y pedir a los estudiantes replicarlo:

La escucha para el string de conectividad

```
Connectivity().onConnectivityChanged.listen((connectivityStatus) {  
  log("connection changed");  
  
  connectivityController.connectivity = connectivityStatus;  
});
```

- i. En el archivo **login_screen.dart** debe añadir el siguiente código y pedir a los estudiantes replicarlo:

Condiciones al validar el estado de red:

```
if (connectivityController.connected) {  
  var result = await AuthManagement.signIn(  
    email: emailController.text,  
    password: passwordController.text);  
  controller.authenticated = result;
```

```
} else {  
    Get.showSnackBar(  
        GetBar(  
            message: "No estas conectado a la red.",  
            duration: const Duration(seconds: 2),  
        ),  
    );  
}
```