CONSEJERÍA DE EDUCACIÓN Curso 23/24

### Recuperación 1ºTrimestre - DWEC

#### Nombre:

#### **EJERCICIO:**

1. Crear un proyecto nuevo con *vite*, elimina y limpia los archivos que creas necesario. Usa los archivos *index.html* y *style.css* facilitados. Si todo ha ido bien, al ejecutar la app con vite deberás ver la siguiente interfaz.

# **Using ReqRes API**

- 2. Crear una clase llamada *user.js* donde:
  - Todos los atributos de clase son privados y el constructor solo tiene como parámetro el *id*.
  - o Atributos de clase: id, email, first\_name, last\_name y avatar
  - Se deben crear *setter y getter* computados para todos los atributos.
  - Se debe crear un array estático con los valores de la cabecera: #, First
     Name, Last Name, Email
- 3. Obtener los datos
  - Al iniciar la aplicación se deben traer los datos de manera asíncrona desde el api: <a href="https://regres.in/api/users?page=1">https://regres.in/api/users?page=1</a>
  - Una vez recibido los datos se deben parsear mediante una *custom promise* a la clase que se ha creado en el punto anterior.
- 4. Con los datos ya pareados, mediante JavaScript crear los elementos html necesarios para el rellenado de la tabla:
  - La tabla está basada en Bootstrap, puedes ver la documentación en el siguiente enlace:
  - o <a href="https://getbootstrap.com/docs/5.3/content/tables/">https://getbootstrap.com/docs/5.3/content/tables/</a>,



TEL: 951297929



WEB: http:iesportada.org

I.E.S PORTADA ALTA

### CONSEJERÍA DE EDUCACIÓN Curso 23/24

```
<thead>
#
 First
 Last
 Handle
</thead>
1
 Mark
 Otto
 @mdo
2
 Jacob
 Thornton
 @fat
3
 Larry the Bird
 @twitter
```

- Ten en cuenta que la cabecera son datos estáticos y se debe usar el array estáticos que has creado en el ejercicio 2.
- Para el rellenado del *cuerpo* de la tabla debes usar los datos que has parseado en el ejercicio 3.
- Si todo va bien el resultado debe ser el siguiente

#	First Name	Last Name	Email
1	George	Bluth	george.bluth@reqres.in
2	Janet	Weaver	janet.weaver@reqres.in
3	Emma	Wong	emma.wong@reqres.in
4	Eve	Holt	eve.holt@reqres.in
5	Charles	Morris	charles.morris@reqres.in
6	Tracey	Ramos	tracey.ramos@reqres.in



TEL: 951297929



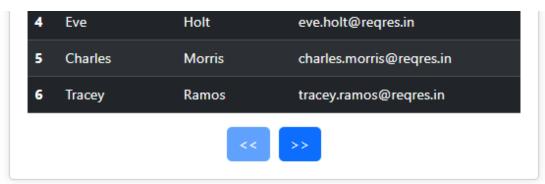
# I.E.S PORTADA ALTA

Calle Cómpeta 3 – 29007, Málaga

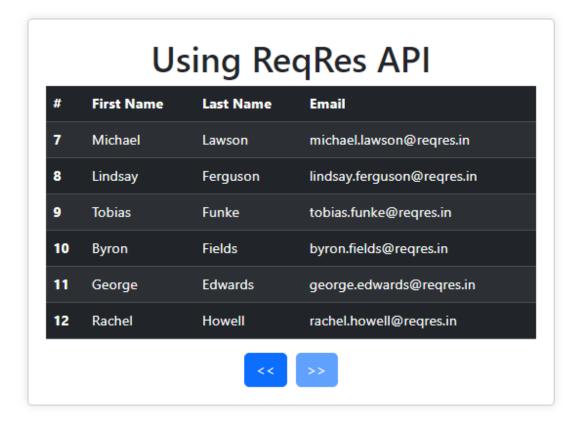
WEB: http://esportada.org

CONSEJERÍA DE EDUCACIÓN Curso 23/24

- 5. Añadir paginación
  - Añadir dos botones tras la tabla, con las clases de Bootstrap "btn btnprimary"
  - $\circ$  Debes usar los caracteres << y>> como el texto de los botones. Se debe ver como la siguiente imagen



Ten en cuenta que la primera vez el botón de la izquierda está deshabilitado, si pulsamos el botón de la derecha se vuelve a llamar al endpoint de reqres dado en el ejercicio 3 pero con el parámetro *page* igual a 2. El resultado debe ser el siguiente:





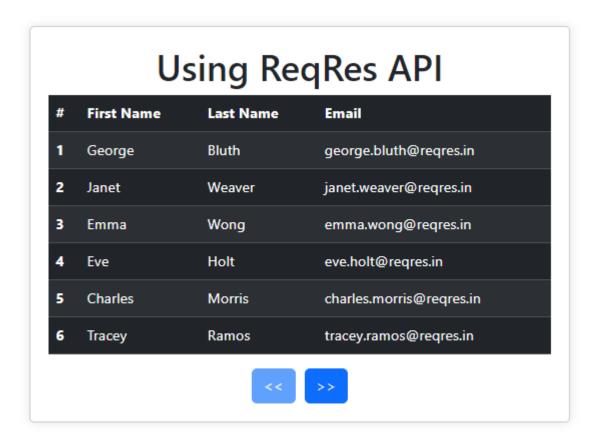
TEL: 951297929



Calle Cómpeta 3 – 29007, Málaga

#### CONSEJERÍA DE EDUCACIÓN Curso 23/24

 Como puedes ver ahora se habilita el botón izquierdo y se deshabilita el derecho. Si pulsamos sobre el botón izquierdo se vuelve hacer una llamada al api con el valor de *page* igual a 1. El resultado debe ser como el de la siguiente imagen



#### Normas de examen y entrega:

- Se deben añadir todos los archivos necesarios en un zip, excluir la carpeta dist y node\_module.
- Cuando yo lo descargue y lo descomprima ejecutaré npm install y después npm run dev, si el proyecto no despliega el examen no se evaluará.
- o Queda totalmente prohibido el uso de herramientas tipo ChatGPT
- Solo se permite el uso de apuntes y tareas de la asignatura.
- En caso de necesidad solo se podrá usar la documentación facilitada en <u>https://developer.mozilla.org/es/</u> cualquier otra web no está permitida



TEL: 951297929



I.E.S PORTADA ALTA
Calle Cómpeta 3 – 29007, Málaga

WEB: http:iesportada.org

### CONSEJERÍA DE EDUCACIÓN Curso 23/24

#### Rúbrica de evaluación

Tarea	Puntuación
El proyecto se ha creado con vite y se ha limpiado/eliminado todos los archivos innecesarios. Al ejecutar el proyecto con npm todo funciona bien.  Se carga de forma correcta el estado inicial de la app, haciendo uso del index.html y del style.css facilitados, como se ve en la primera imagen	0.25
Se ha implementado la clase <i>user</i> de manera correcta	1
Se obtienen los datos desde la API de forma correcta	0.5
Se parsea la información obtenida de la api a la clase creada haciendo uso de promesas	1.5
Se crea la tabla y se rellena mediante JavaScript haciendo uso de los datos parseados	4
Se implementa la paginación	
Se tendrá en cuenta el control de errores mediante try/catch	0.25

Nota: El uso de buenas prácticas y la buena implementación se tendrán en cuenta





I.E.S PORTADA ALTA

Calle Cómpeta 3 – 29007, Málaga

TEL: 951297929 WEB: http://esportada.org