

Introducción al Desarrollo Web/Móvil

Práctica I

Cliente Web y API

Portafolio personal

Descripción:

En la primera práctica de la asignatura, usted creó su portafolio personal utilizando únicamente HTML y CSS. Dado que muchas organizaciones trabajan con frameworks, es necesario que actualice su portafolio para que utilice estas herramientas.

Backend:

La API debe tener un único endpoint con la URL `"/api/profile"`, el cual debe ser un método GET. Este endpoint debe devolver un JSON con los siguientes atributos:

1. Name: Nombre de la persona.
2. Lastname: Apellido de la persona.
3. Email: Correo electrónico de la persona.
4. City: Ciudad de origen de la persona.
5. Country: País de origen de la persona.
6. Summary: Resumen de la persona.
7. Frameworks: Este debe ser un arreglo que contenga objetos con los siguientes atributos:
 - a. Name: Nombre del framework.
 - b. Level: Nivel de conocimiento en el framework.
 - c. Year: Año en que empezó a utilizar el framework.
8. Hobbies: Este debe ser un arreglo que contenga objetos con los siguientes atributos:
 - a. Name: Nombre del hobby.
 - b. Description: Descripción del hobby.

La Figura 1 muestra un ejemplo del JSON que debe devolver el único endpoint.

Frontend:

La aplicación web debe consumir el único endpoint de la API y utilizar esta información para crear el portafolio personal en la URL `"/portfolio"`. La Figura 2 expone un ejemplo de lo anterior.

```
{
  "name": "Ricardo",
  "lastname": "Urvina",
  "email": "rickyurvina@alumnos.ucn.cl",
  "city": "Quito",
  "country": "Ecuador",
  "summary": "Soy Ricardo Urvina, desarrollador...",
  "frameworks": [
    {
      "name": "Laravel",
      "level": "Experto",
      "year": 2016
    },
    {
      "name": "React",
      "level": "Avanzado",
      "year": 2018
    }
  ],
  "hobbies": [
    {
      "name": "Fútbol",
      "description": "Jugar fútbol con amigos"
    },
    {
      "name": "Robótica",
      "description": "Construir robots"
    }
  ]
}
```

Figura 1. Ejemplo de JSON.

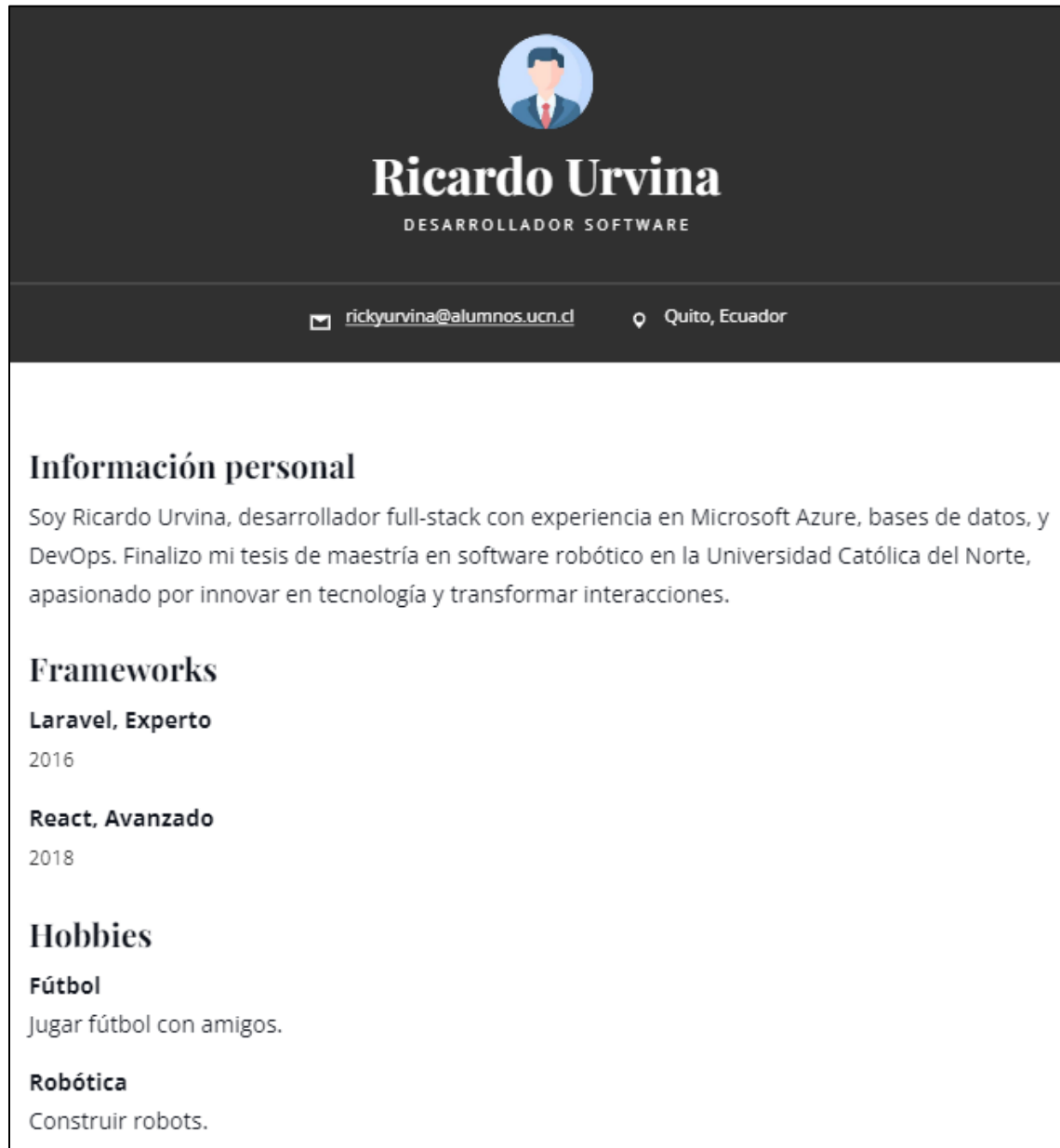


Figura 2. Ejemplo de Portfolio.

Consideraciones:

- Se debe usar GitHub para versionar su trabajo, debe ser solo un repositorio para el cliente web y la API. **No se permiten commits después de la fecha de entrega.**
- Se debe desarrollar la interfaz de usuario utilizando el framework o librería web escogido, además se debe integrar un framework o biblioteca de CSS como Bootstrap o Tailwind CSS para el diseño.
- La aplicación debe tener un diseño agradable para la vista de un usuario, debe tener un diseño elegante y vistoso. **No se permite el uso de templates de terceros.**

Entregables:

La entrega de la práctica debe contener:

1. Ambos componentes, cliente web y API en un mismo repositorio de **GitHub**.
2. Archivo **README.md** en la raíz del proyecto con las instrucciones para levantar los proyectos de web y de la API. Estas instrucciones deben estar dirigidas a un usuario que **no tiene las tecnologías ocupadas instaladas** en su máquina personal.

Rúbrica de evaluación:

- La Tabla I corresponde a la rúbrica del taller, la nota del taller será calculada sobre el puntaje total con un 60% de exigencia.

Tabla I: Rúbrica de evaluación.

Componente	Descripción	Puntaje
Uso de Git	El proyecto se encuentra versionado con Git en GitHub o BitBucket. El proyecto se encuentra en un único repositorio de GitHub que contiene tanto el cliente web como la API.	5
Archivo README.md	Se encuentra el archivo README.md con las instrucciones para levantar el proyecto.	5
endpoint	Se ha implementado un controlador que atiende las solicitudes de la URL <code>"/api/portfolio"</code> con el método GET.	20
Interfaz Web	Se ha implementado una interfaz web elegante y agradable a la vista	10
Retorno de JSON	El endpoint devuelve un JSON válido con la estructura visualizada en la Figura 1.	10
Llamado a la API	La aplicación web consume el único endpoint de la API.	20
Responsividad	La aplicación web tiene un diseño agradable, elegante y vistoso, que se adapta a diferentes dispositivos y tamaños de pantalla (responsividad).	15
Framework de CSS	Se ha integrado un framework o biblioteca de CSS (por ejemplo, Bootstrap o Tailwind CSS) para mejorar el diseño de la aplicación.	15
Total		100

Good Luck! 😊 🍀