

INPUT IT

Fecha

29/09/2023

Desarrollador

Sebastian Menacho

1. Introducción:

"Input it" es una página de formulario que permite a los usuarios enviar una respuesta, editarla y explorar las respuestas de otros usuarios.

2. Arquitectura:

La aplicación está escrita en TypeScript y utiliza algunas librerías como "React" para el frontend y "Node.js" para el backend.

2.1. Frontend: React, React router, Redux toolkit, Axios, Vite, Tailwind, React phone number input (selector de código de número por país).

2.1. Backend: Node, Express, Cors, Sequelize, Pg (PostgreSQL), Node fetch (la opción más ligera para peticiones HTTPS en el servidor), Dotenv, UUID (generador del id de la respuesta enviada por el usuario).

3. Base de datos:

La aplicación tiene solo un modelo definido ("Answer") para almacenar las respuestas de los usuarios. Aparte de: **answer_id**, **full_name**, **phone_number**, **start_date**, **preferred_language**, **how_found**, **newsletter_subscription**. El modelo cuenta con la columna "**session_id**" que almacena el UUID que se genera luego de que el usuario haya enviado una respuesta.

4. Consideraciones:

La aplicación trabaja con una base de datos online. El archivo .env ubicado en la carpeta "SERVER" contiene la URL de esta base de datos. Además, el token que se usa para conseguir el código de país viene de "ipinfo.io"; este token también está ubicado en el archivo .env.

El dato "**session_id**" es guardado únicamente en el localStorage de manera que si el usuario cierra el navegador perderá la facultad de editar su respuesta. El usuario nunca se va a ver obligado a crear una cuenta para enviar una respuesta porque no hay autenticación de ningún tipo.

5. Métodos:

Cuatro de las 5 rutas disponibles están relacionadas con el modelo "Answer" de la base de datos. La quinta ruta **`/countryCode`** recupera el código país del usuario.

`getAllAnswers()`

- Retorna todas las respuestas almacenadas en la base de datos. Es usado para mostrar las respuestas en **`/results`** (en el cliente), después de que el usuario haya enviado su respuesta.

`getAnswerById()`

- Retorna la respuesta que coincide el ID pasado por parámetro. Es usada para mostrar la respuesta del usuario en **`/results`** (en el cliente), después de que el usuario haya enviado su respuesta. Y, también es usado para cargar la respuesta en **`/form?edit=true`**, después de que el usuario haga click en "Editar".

`postAnswer()`

- Crea una nueva instancia en el modelo "Answer" con el objeto pasado por el cuerpo de la petición. Además, crea un UUID para luego poder recuperar la respuesta. En el cliente, el endpoint asociado es usado en la ruta **`/form`** y la ruta **`/form?edit=true`**.

`putAnswerById()`

- Edita la respuesta que coincida con el **`"session_id"`** pasada por parámetros. Usa el objeto pasado por el cuerpo de la petición para editar la respuesta.

`getCountryCode()`

- Retorna el código de país del usuario. Hace una petición a ["https://ipinfo.io/179.6.6.159?token=\\${IP_TOKEN}"](https://ipinfo.io/179.6.6.159?token=${IP_TOKEN}) con el **`IP_TOKEN`** que viene del archivo `.env`. Este código es usado en el elemento **`"PhoneInput"`** de la librería `"react-phone-number-input"` (el elemento **`"PhoneInput"`** es renderizado en la ruta **`/form`** y **`/form?edit=true`**).

6. Enlaces:

La base de datos está desplegada en f10.com, el servidor en render.com y el cliente en vercel.com. La página que recopila la dirección IP del usuario para mostrar el código país es ipinfo.io.

App URL

<https://inputit.vercel.app>