

UNIVERSIDAD NACIONAL DE INGENIERÍA

FACULTAD DE CIENCIAS

Documentación del proyecto



**Título de proyecto:** Tienda virtual de productos farmacéuticos con reconocimiento de voz

**Curso:** Interacción Humano-Computador (CC451A)

**Integrantes:**

Reyes Gutierrez Jose Daniel	20162732A
Vargas Torres Eros Aylthon	20182203D
Oliva Valdivia Abel Alejandro	20102198I
Yacolca Huaman Jesus Miguel	20183004E

**Profesor:**

Ciro Nunez

**Ciclo:** 2021-2

## **Resumen**

FarmUni es una página de ventas de productos tal como los que venderían MiFarma Farmacia Universal o Inkafarma solo que lo novedoso en nuestra aplicación es que implementamos una navegación por voz que va a permitir dirigirse a opciones dadas en la página ya sea perfil, tienda virtual, menú principal solo activando el botón de comenzar reconocimiento de voz y pronunciando el comando respectivo. Las otras funcionalidades que tiene FarmUni son como las de una página de ventas normal ya sean agregar lo que se quiere comprar a un carrito de compras, pagar lo que se agregó al carrito de compras , y por ultimo un apartado de sugerencias donde el cliente pueda decirnos alguna recomendación.

## **Vista Principal**

- Mediante nuestro interface de voz implementada gracias a la librería react speech recognizer, el usuario podrá navegar en la página, interactuando con nuestra aplicación.
- Esta interface de voz podrá ser usada en la tienda virtual como filtrador de productos, e incluso se podrá hacer una compra con una simple frase.

## **Usuarios de la Aplicación:**

- **Usuario cliente:**  
Este tipo de usuario accederá a nuestra aplicación y podrá navegar gracias nuestro reconocimiento de voz e incluso seleccionar productos.  
Este tipo de usuario puede ser: usuario registrado o usuario sin registrar.
- **Usuario Administrador:**  
Este usuario tendrá permisos especiales incluyendo las funciones CRUD para la lista de productos en nuestra tienda Virtual.

## **Algunas de nuestras actividades:**

- Para el usuario sin registrar se crea la opción de registrarse para que puedan ser partícipes de la tienda virtual, en la cual deben colocar sus datos personales.
- El usuario registrado tiene la opción de interactuar con la página web. Esta interacción se basa en poder realizar compras o quitar productos dependiendo de su tipo de registro que tenga. Según este se subdivide en 2 grupos más: usuario cliente y administrador.
- El usuario cliente es capaz de realizar compras en la plataforma mediante un pago en efectivo o su tarjeta de crédito
- También puede contribuir dando sugerencias sobre la experiencia de usuario en la página.
- El usuario administrador tiene la capacidad de agregar o quitar productos de la tienda según la situación lo amerite.

Adicionalmente para una mayor diversidad de usuarios se implementará una interfaz de voz, la cual permitirá una interacción más sencilla y rápida en la página. Un usuario podrá moverse con facilidad a través de la página, además de poder realizar diversas acciones tales como elegir el producto (medicamento) a comprar.

### **Alcance del proyecto**

Llegar a mejorar y optimizar el código ya hecho, hacerlo más accesible y mucho más rápido para pueda ser de ayuda todos nuestros usuarios, en especial a aquellos con alguna discapacidad.

### **Librerías Usadas**

- axios
- bootstrap
- modules
- react
- react-bootstrap
- react-dom
- react-router-dom
- react-scripts
- react-validation
- sweetalert2
- sweetalert2-react-content
- validator
- web-vitals
- react speech recognizer

### **Librerías Adicionales**

- faker
- @fortawesome/fontawesome-svg-core
- @fortawesome/free-solid-svg-icons
- @fortawesome/react-fontawesome

### **Restricciones de nuestro Proyecto:**

- 1) Nos enfocamos más en frontend más que en el backend para poder hacer los requerimientos del reconocimiento de voz.
- 2) Nos centramos en el cliente más que administrador, en otras palabras no se implementaron funciones CRUD para la tienda virtual modo administrador.
- 3) No se implementó una base de datos para almacenar los mensajes de las sugerencias, lo cual se hará más adelante.

- 4) No se llegó a completar un carrito de compras, donde el usuario cliente pueda ver sus productos comprados y el historial de compra.
- 5) En la tienda virtual nos faltó un filtrador de productos y una función agregar que funcionara con el código de voz.

### **Forma de Inicializar el Proyecto Farmuni**

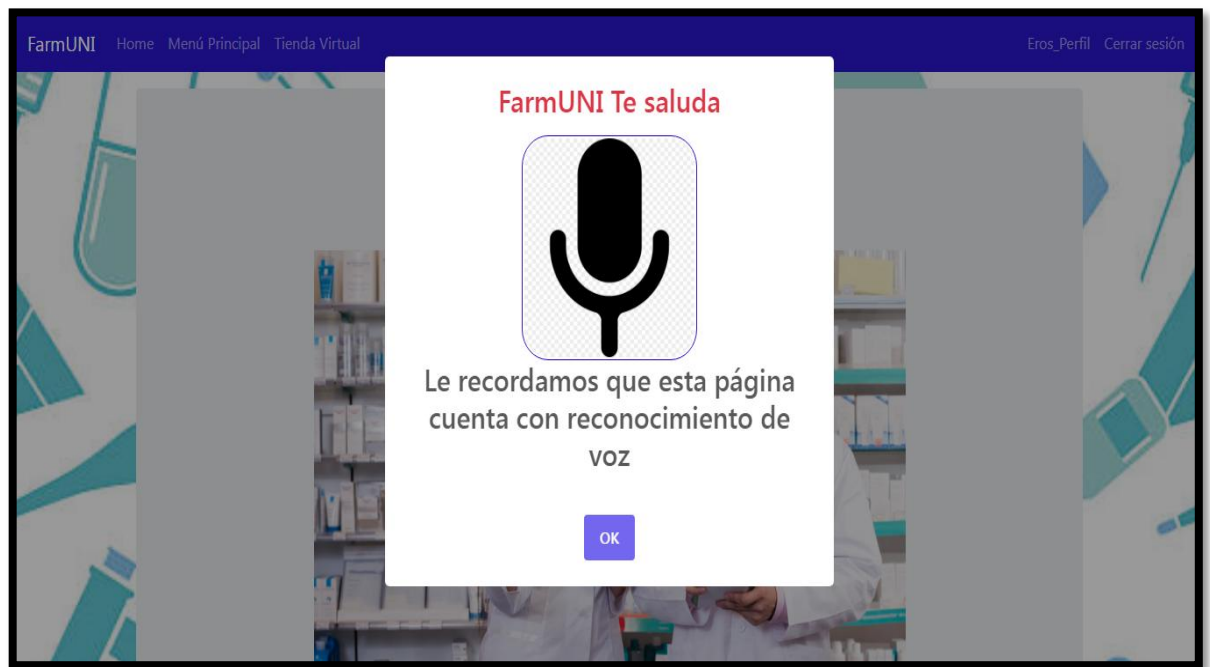
- Acceder a dos consolas diferentes y entra con `cd backend` y `cd frontend` respectivamente
- En la primera consola ejecutar los comando `yarn install`
- Una vez terminada de cargar las dependencias del backend, usar el comando `yarn start` , esto conectara la base de datos a mongo atlas.
- En la segunda consola ejecutar los comando `yarn install`
- Una vez terminada de cargar las dependencias del frontend, usar el comando `yarn start`, esto iniciara la página.

### **Interface e Imágenes**

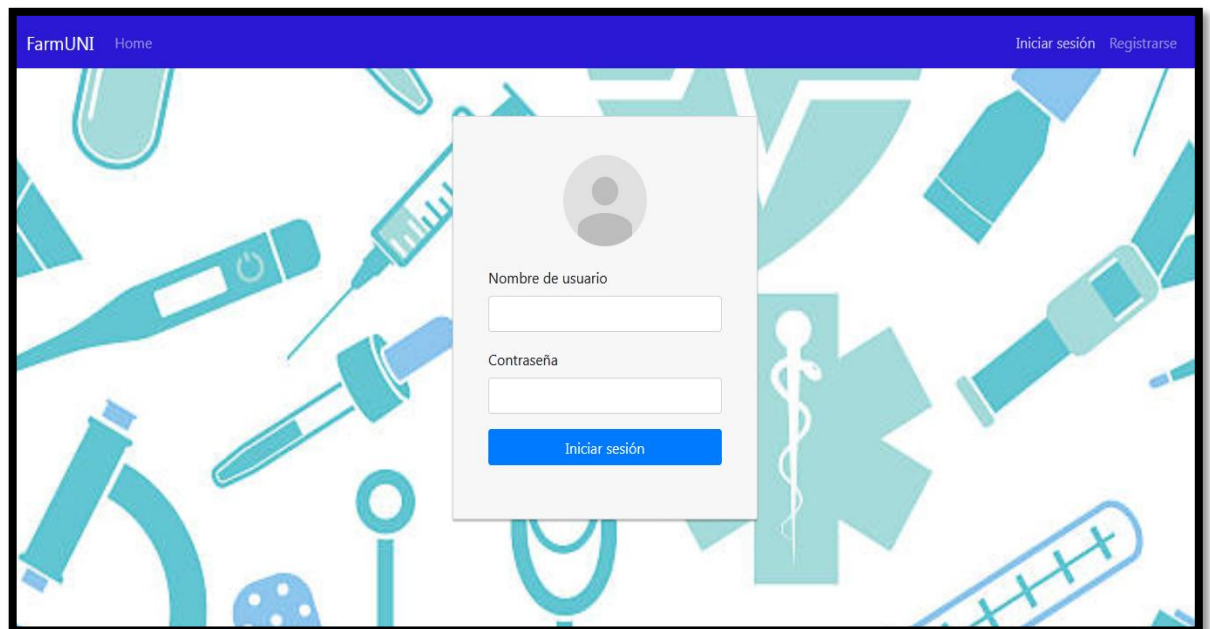
- 1) Inicio de pantalla fase 1



## 2) Inicio de pantalla fase 2



## 3) Página Login



#### 4) Página Registro

FarmUNI Home [Iniciar sesión](#) [Registrarse](#)

Nombre

Apellido

Correo

Telefono

DNI

#### 5) Conexiones a Mongo Atlas:

farmuni-project Atlas Realm Charts

cliente  
tipousuarios  
usuarios

INSERT DOCUMENT

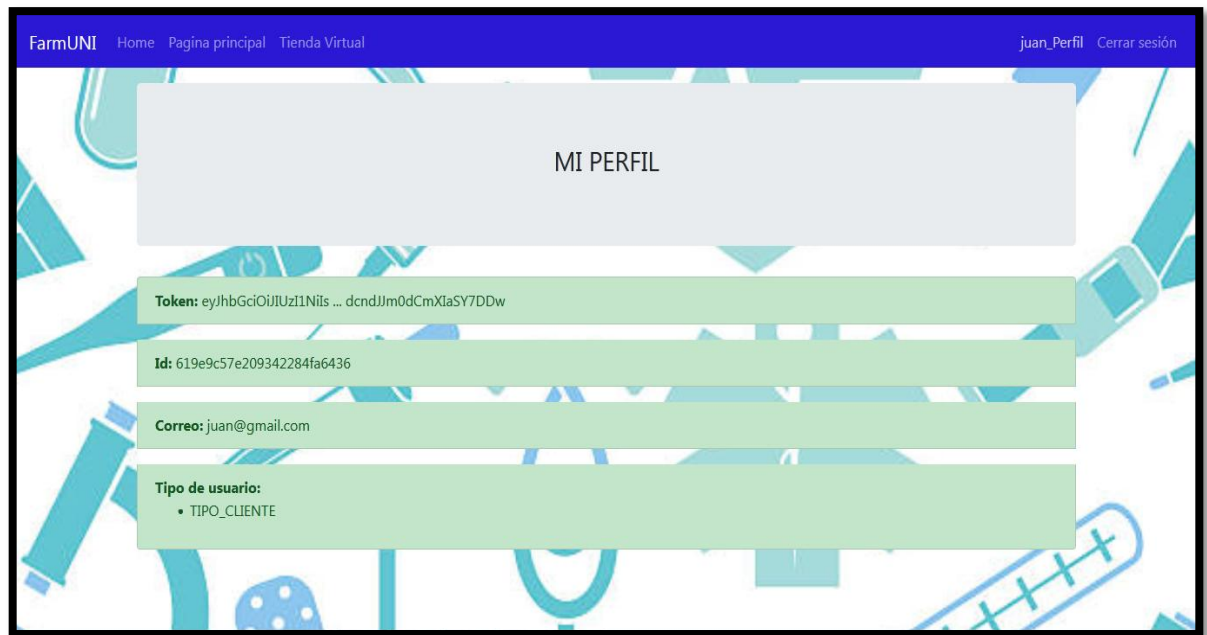
FILTER { field: 'value' } OPTIONS Apply Reset

QUERY RESULTS 1-6 OF 6

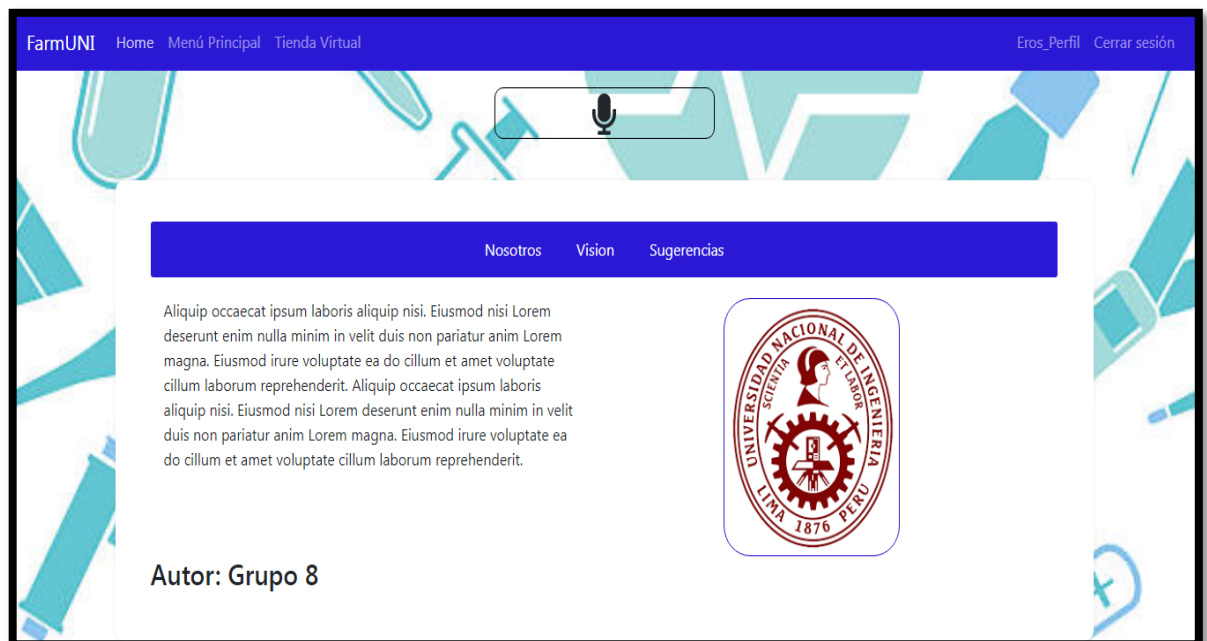
```
{
  "_id": ObjectId("619a7ddcf9e631fb8e80696"),
  "dni": "88774455",
  "nombreCliente": "alex",
  "apellidoCliente": "armiso",
  "direccion": "luz",
  "telefono": "77777777",
  "idUsuario": ObjectId("619a7dbf9e631fb8e80694"),
  "_v": 0
}
```

```
{
  "_id": ObjectId("619a905ffa9e631fb8e806a4"),
  "dni": "44778877",
  "nombreCliente": "karina",
  "apellidoCliente": "torres",
  "direccion": "junio",
  "telefono": "885522963",
  "idUsuario": ObjectId("619a905ffa9e631fb8e806a2"),
  "_v": 0
}
```

## 6) Perfil del Usuario



## 7) Menú Principal



## 8) Página de la tienda Virtual

FarmUNI
Home
Menú Principal
Tienda Virtual
Eros\_Perfil
Cerrar sesión

TIENDA FARMUNI

ALCOHOL GEL
1000 ML

Gel Antibacterial

Vitamina C
20 tabletas

Vitamina C + Zinc.Efervescente

Tabletas

Dolor fuerte
26 SOB X 2 TABS

Panadol Forte Tableta

Reduce la fiebre y alivia

Masoorilla con Valvula Eco Air Mask Talla M Negro x 1 Unidad

\$125.00

Parche Circular Opticulous Nexcare para Adultos x 20 Unidades

\$152.00

Parches Recortables para Pies Flat Works Molexón x 4 Unidades

\$135.00

Contravena 300mg x 60 Capsulas

\$43.00

Albiterm Alcohol Medicinal 70° x 120 ml

\$13.20

Solución para Lentes de Contacto Opti-Free x 120 ml

\$13.50

Dulcinea 5 mg x 20 Tabletas

\$13.00

Cinta Quirúrgica de Papel Micropore Nexcare 2.5 cm x 9.1 m

\$15.00

Parche 28 Casilleros x 1 Unidad

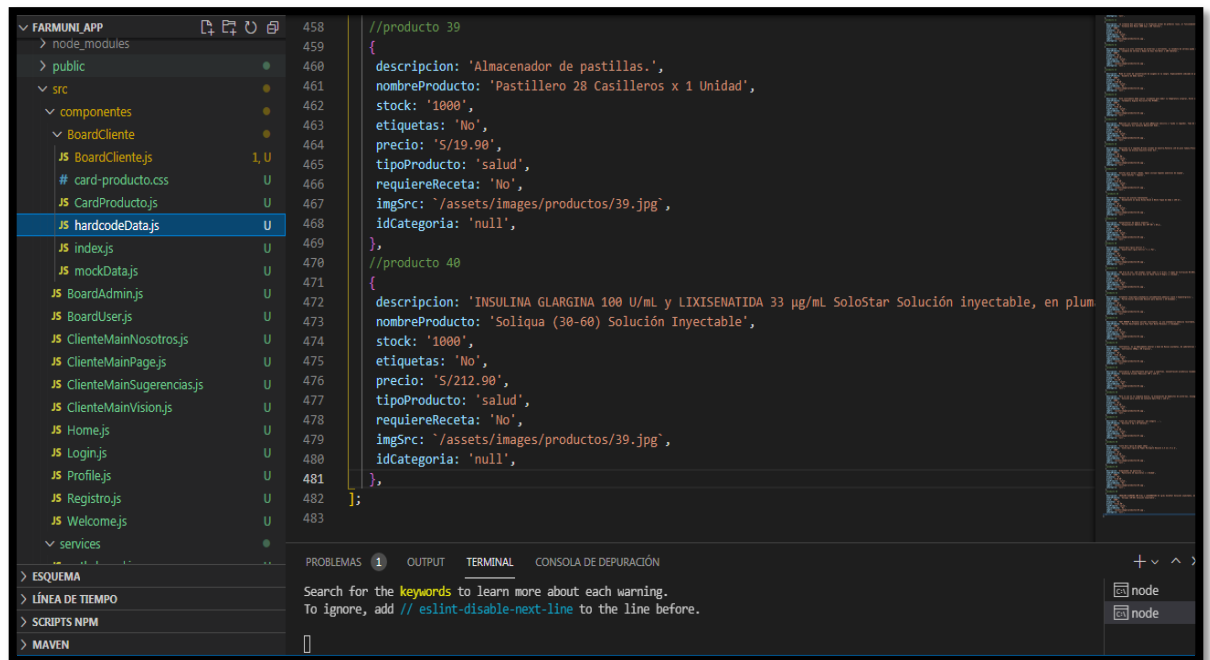
\$15.00

Organizador de Pastillas

\$15.00



9) Código de la Data ingresada manualmente : 40 productos

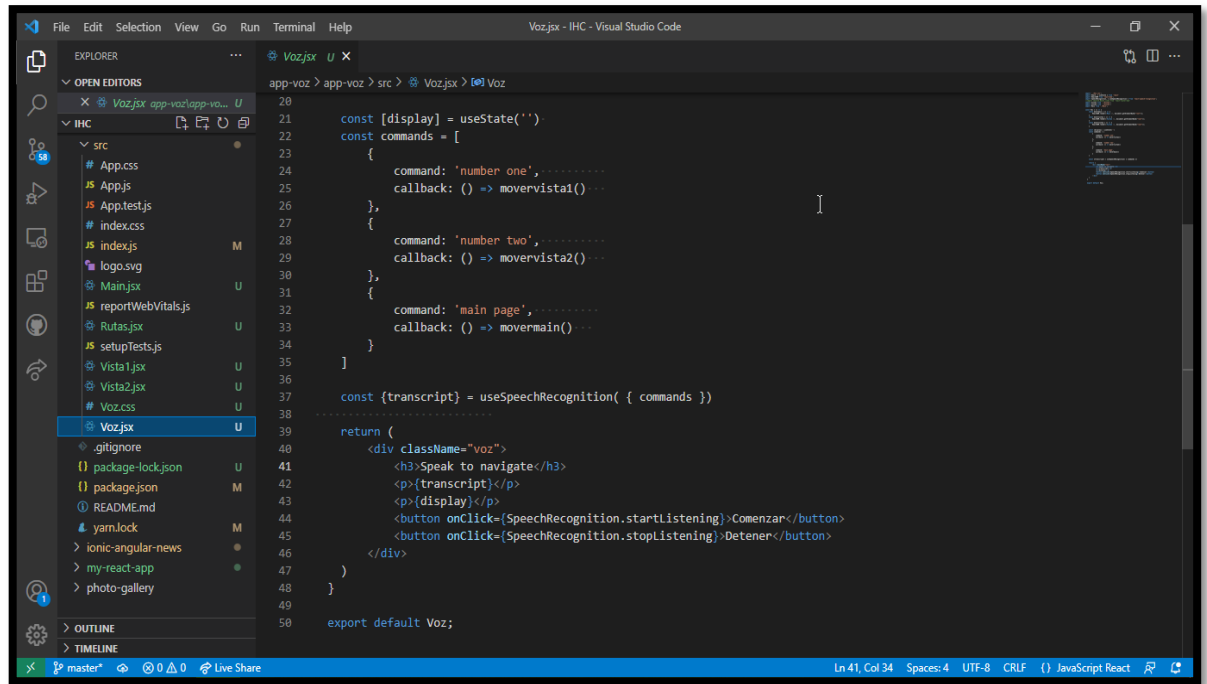


```
458 //producto 39
459 {
460   descripcion: 'Almacenador de pastillas.',
461   nombreProducto: 'Pastillero 28 Casilleros x 1 Unidad',
462   stock: '1000',
463   etiquetas: 'No',
464   precio: '$/19.90',
465   tipoProducto: 'salud',
466   requiereReceta: 'No',
467   imgSrc: '/assets/images/productos/39.jpg',
468   idCategoria: 'null',
469 },
470 //producto 40
471 {
472   descripcion: 'INSULINA GLARGINA 100 U/mL y LIXISENATIDA 33 µg/mL SoloStar Solución inyectable, en plum
473   nombreProducto: 'Soliqua (30-60) Solución Inyectable',
474   stock: '1000',
475   etiquetas: 'No',
476   precio: '$/212.90',
477   tipoProducto: 'salud',
478   requiereReceta: 'No',
479   imgSrc: '/assets/images/productos/39.jpg',
480   idCategoria: 'null',
481 },
482 ];
483
```

10) Data guardada para la creación del carrito de compras (efecto de agregar):



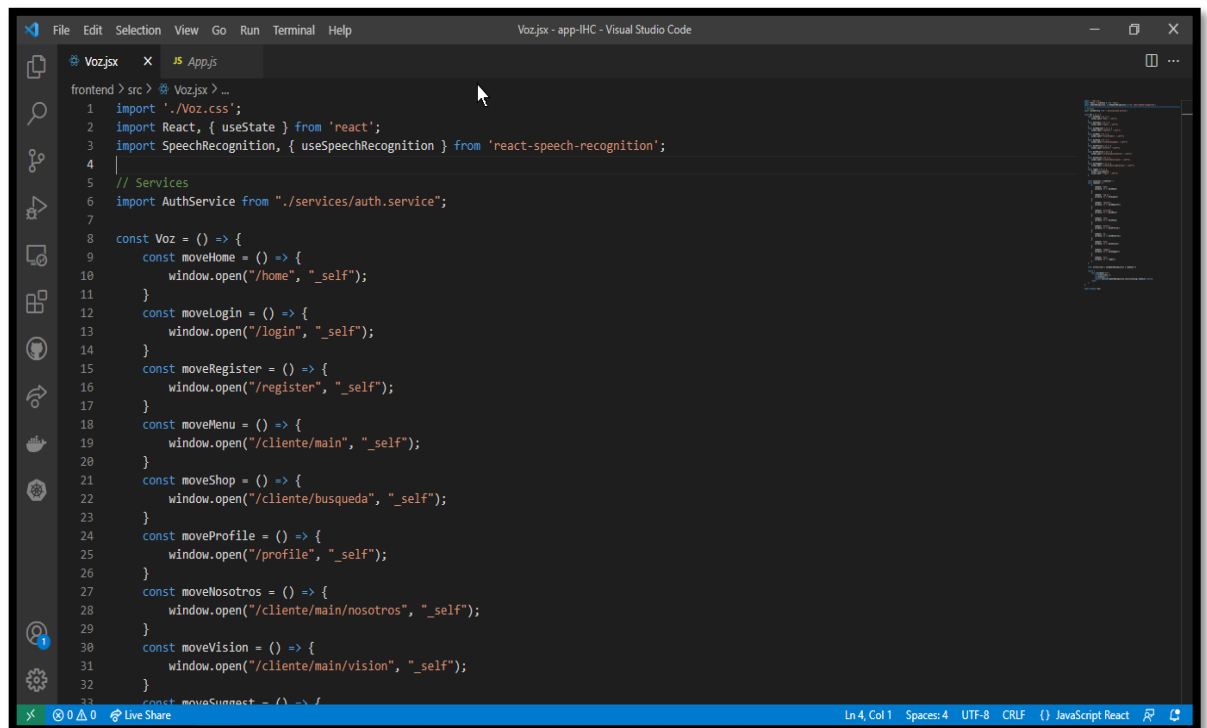
## 11) Código del reconocimiento de voz 1ra fase



The screenshot shows the Visual Studio Code editor with the file `Voz.jsx` open. The code implements the first phase of voice recognition using `useState` and `useSpeechRecognition`. It defines a `commands` array with three entries: 'number one', 'number two', and 'main page'. Each entry has a corresponding `callback` function that calls `movervista1()`, `movervista2()`, and `movermain()` respectively. The `transcript` state is updated by `useSpeechRecognition`. The JSX part of the code renders a `div` with the class `voz`, containing a heading, a paragraph for the transcript, and two buttons: 'Comenzar' (which starts listening) and 'Detener' (which stops listening). The status bar at the bottom indicates the file is at line 41, column 34, using UTF-8 encoding with CRLF line endings.

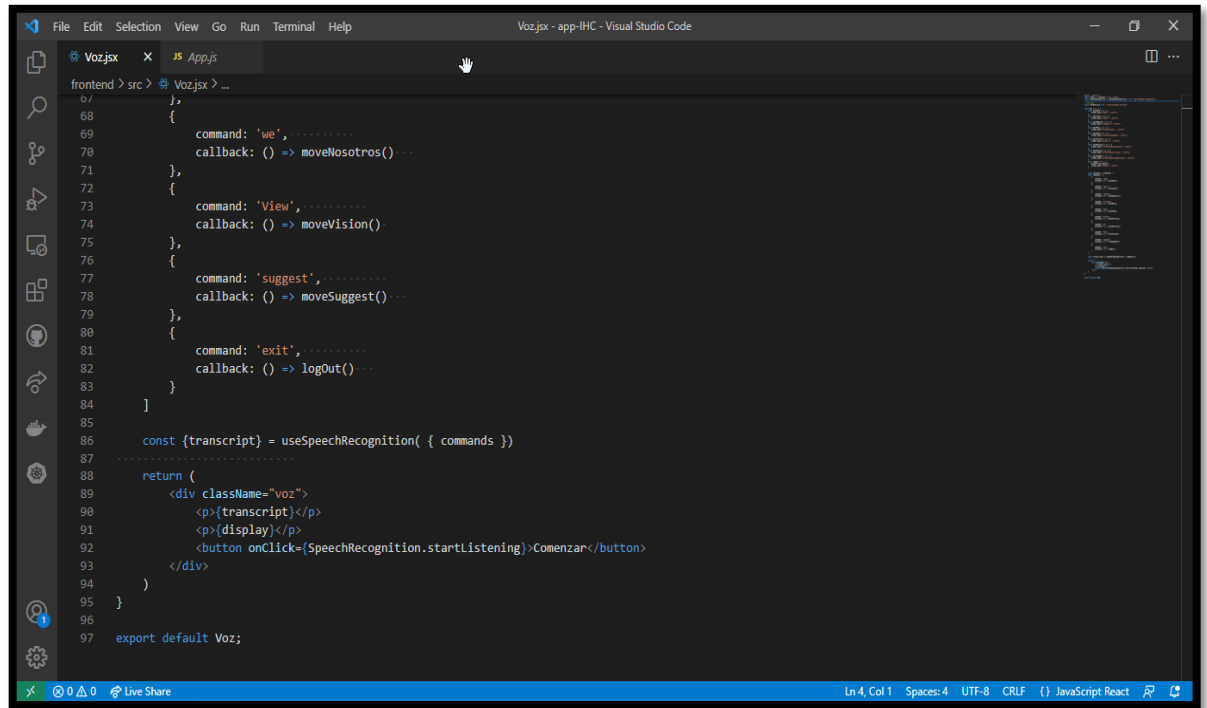
```
20
21 const [display] = useState('')
22 const commands = [
23   {
24     command: 'number one', .....
25     callback: () => movervista1() ...
26   },
27   {
28     command: 'number two', .....
29     callback: () => movervista2() ...
30   },
31   {
32     command: 'main page', .....
33     callback: () => movermain() ...
34   }
35 ]
36
37 const {transcript} = useSpeechRecognition({ commands })
38
39 return (
40   <div className="voz">
41     <h3>Speak to navigate</h3>
42     <p>{transcript}</p>
43     <p>{display}</p>
44     <button onClick={SpeechRecognition.startListening}>Comenzar</button>
45     <button onClick={SpeechRecognition.stopListening}>Detener</button>
46   </div>
47 )
48
49 export default Voz;
```

## 12) Código del reconocimiento de voz 2da fase



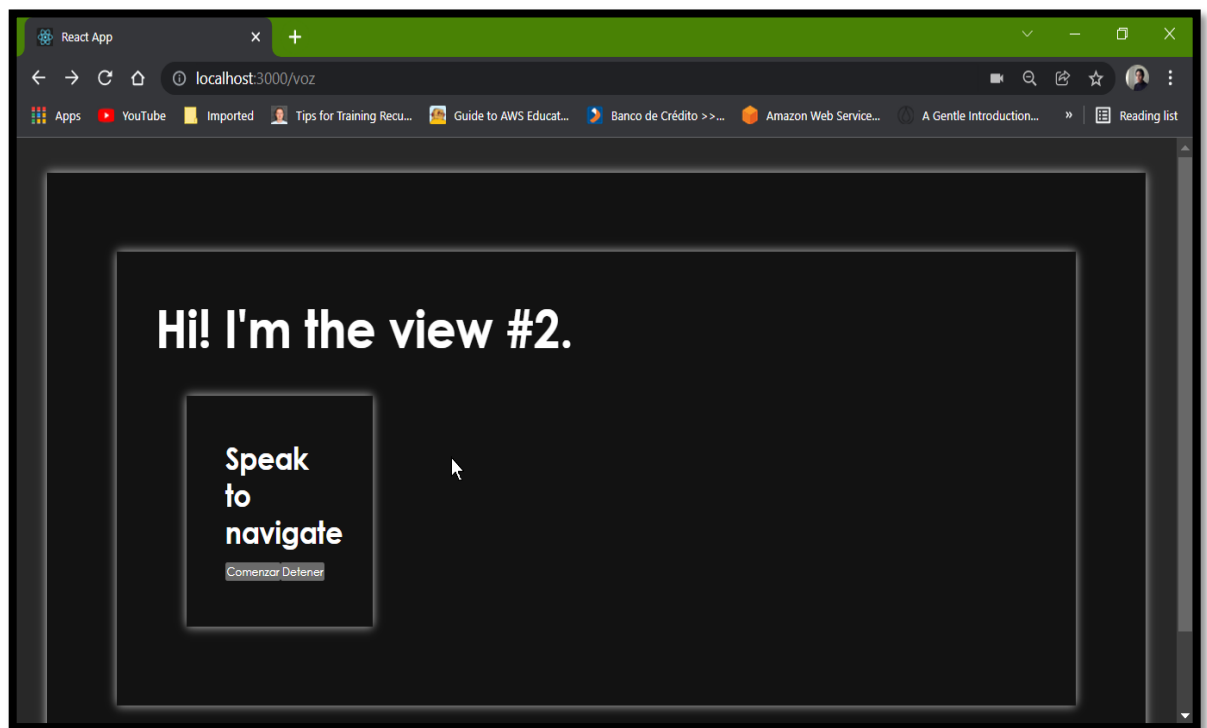
The screenshot shows the Visual Studio Code editor with the file `App.js` open. The code implements the second phase of voice recognition by importing `AuthService` and defining a series of navigation functions. These functions use `window.open` to open different pages based on the voice command. The functions are: `moveHome` (opens '/home'), `moveLogin` (opens '/login'), `moveRegister` (opens '/register'), `moveMenu` (opens '/cliente/main'), `moveShop` (opens '/cliente/busqueda'), `moveProfile` (opens '/profile'), `moveNosotros` (opens '/cliente/main/nosotros'), and `moveVision` (opens '/cliente/main/vision'). The status bar at the bottom indicates the file is at line 4, column 1, using UTF-8 encoding with CRLF line endings.

```
1 import './Voz.css';
2 import React, { useState } from 'react';
3 import SpeechRecognition, { useSpeechRecognition } from 'react-speech-recognition';
4
5 // Services
6 import AuthService from './services/auth.service';
7
8 const Voz = () => {
9   const moveHome = () => {
10     window.open("/home", "_self");
11   }
12   const moveLogin = () => {
13     window.open("/login", "_self");
14   }
15   const moveRegister = () => {
16     window.open("/register", "_self");
17   }
18   const moveMenu = () => {
19     window.open("/cliente/main", "_self");
20   }
21   const moveShop = () => {
22     window.open("/cliente/busqueda", "_self");
23   }
24   const moveProfile = () => {
25     window.open("/profile", "_self");
26   }
27   const moveNosotros = () => {
28     window.open("/cliente/main/nosotros", "_self");
29   }
30   const moveVision = () => {
31     window.open("/cliente/main/vision", "_self");
32   }
33 }
```

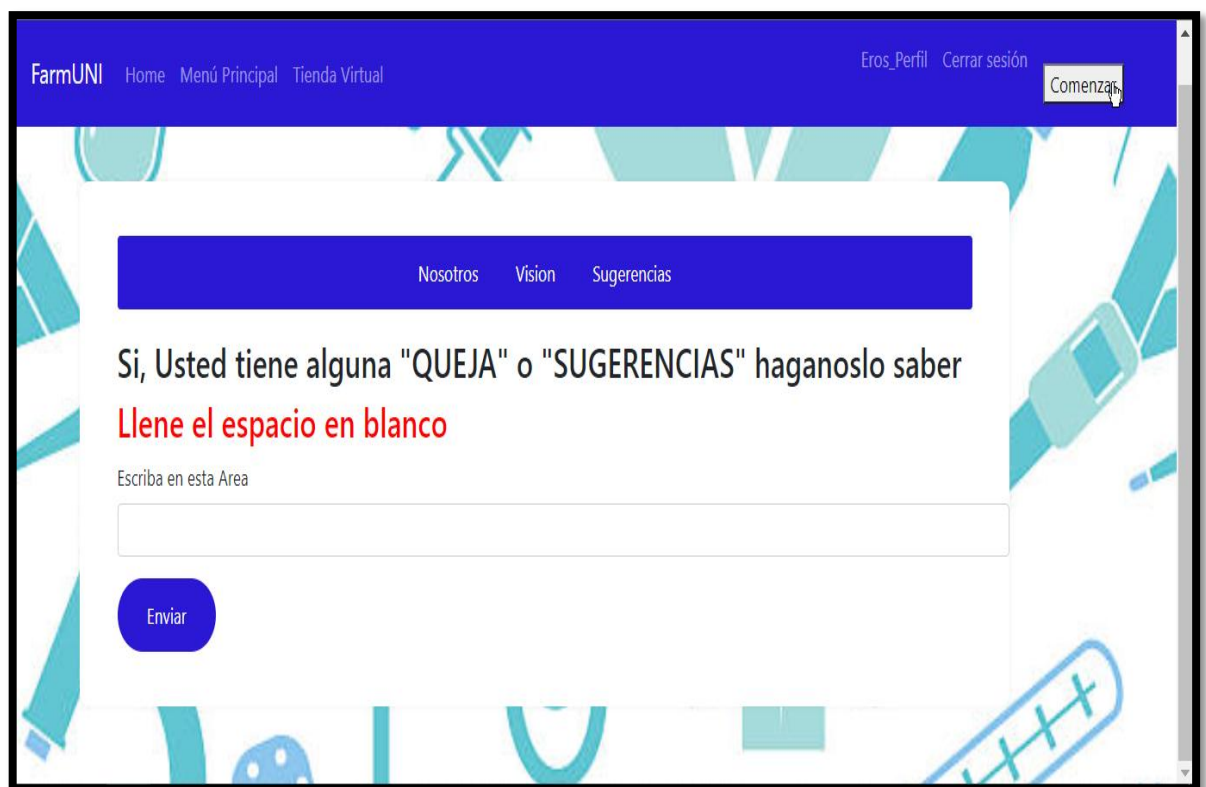


```
0 /
68 {
69   command: 'we', .....
70   callback: () => moveNosotros() ...
71 },
72 {
73   command: 'View', .....
74   callback: () => moveVision() ...
75 },
76 {
77   command: 'suggest', .....
78   callback: () => moveSuggest() ...
79 },
80 {
81   command: 'exit', .....
82   callback: () => logOut() ...
83 }
84 ]
85
86 const {transcript} = useSpeechRecognition( { commands })
87 .....
88 return (
89   <div className="voz">
90     <p>{transcript}</p>
91     <p>{display}</p>
92     <button onClick={SpeechRecognition.startListening}>Comenzar</button>
93   </div>
94 )
95 }
96
97 export default Voz;
```

13) Interface del reconocimiento de voz 1ra fase



14) Interface del reconocimiento de voz 2da fase “Con el botón Comenzar”



1

- TIPO\_CLIENTE