

Alejandro Leal Castaño
Mario Alonso Núñez
Jose Eduardo Raimundo Fernando
05/05/2023

La Cesta Verde

Índice de contenidos

1. La idea
2. Objetivos y requisitos
3. Vista general del Frontend
 1. Sección de Acceso
 2. Sección de Catalogo
 3. Sección de Producto
 4. Sección de Perfil
 5. Sección de Negocio
 6. Sección de Cesta
4. Aplicaciones Multimedia
 1. Obtención automática de valores nutricionales
 2. Implementación de mapas
 3. Generación de códigos QR
5. Conclusiones

Objetivos y requisitos

- Objetivos:
 - Gestión de usuarios y perfiles
 - Gestión de productos
 - Gestión de compraventa
 - Gestión del catálogo
- Requisitos no funcionales:
 - Escalabilidad
 - Información gráfica y visual
 - Seguridad


Vista general del Frontend

Sección de acceso

Ionice App

localhost:8100/access/login

Invitado



Cesta Verde

test1@gmail.com

.....

ACCEDER

¿No tienes cuenta? [Crear una aquí](#)

Cesta Verde

test1@gmail.com

.....

ACCEDER

¿No tienes cuenta? [Crear una aquí](#)

Sección de acceso

localhost:8100/access/createAccount

CestaVerde

Nuevo usuario

test1

test1@gmail.com

.....

☐ Acepto los Términos y condiciones

REGISTRAR

¿Ya tienes una cuenta? [Ingresa aquí](#)

Nuevo usuario

test1

test1@gmail.com

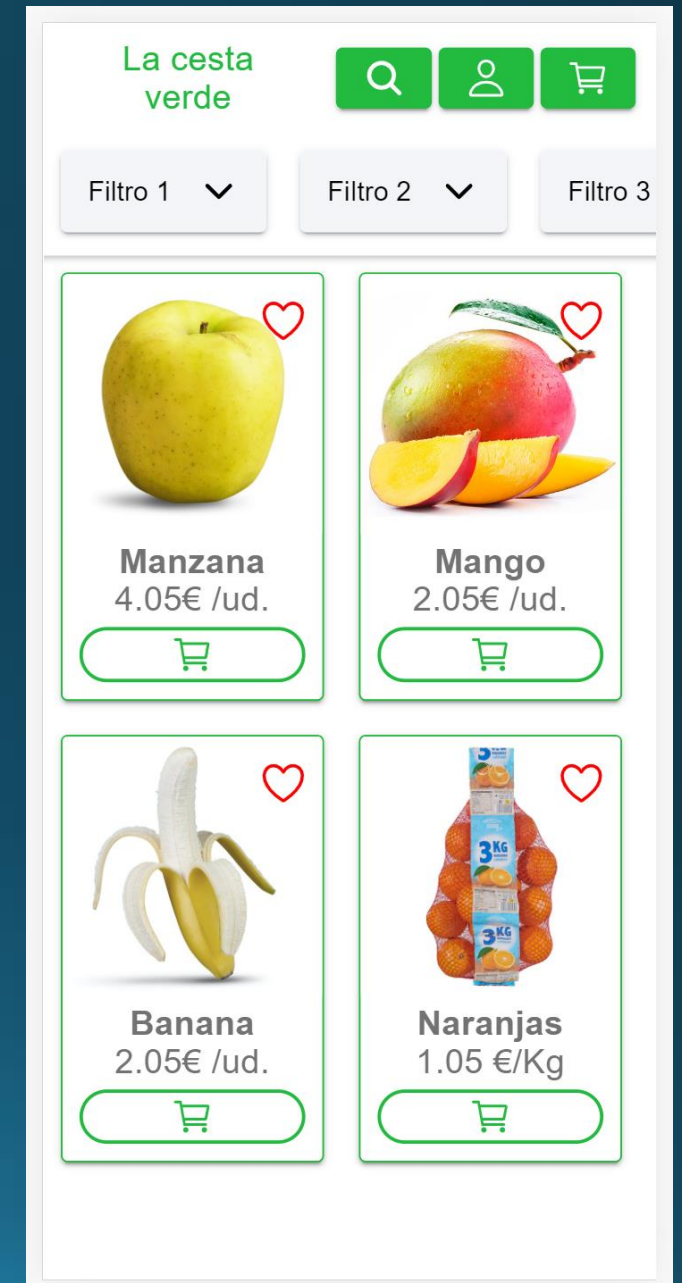
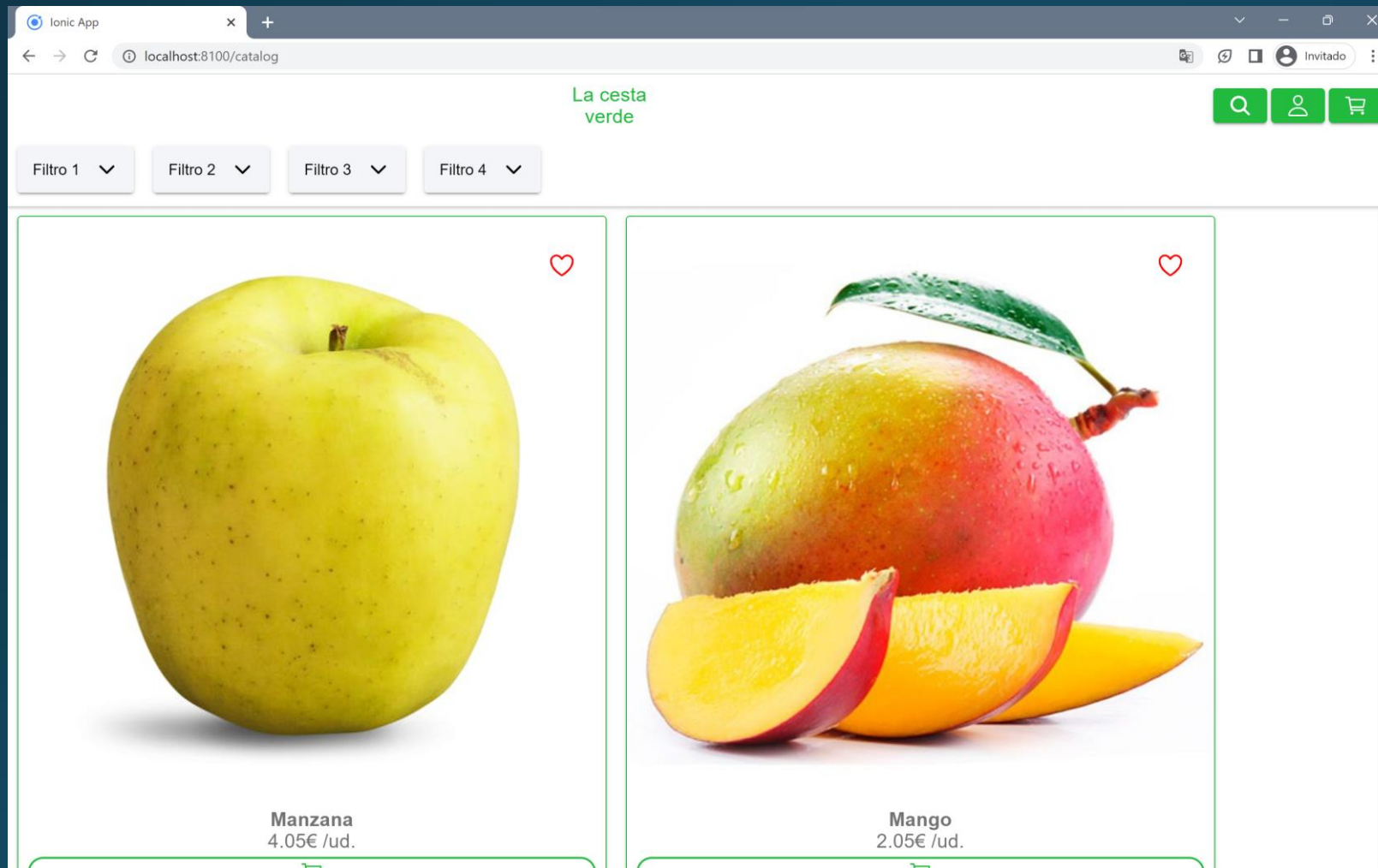
.....

☐ Acepto los Términos y condiciones

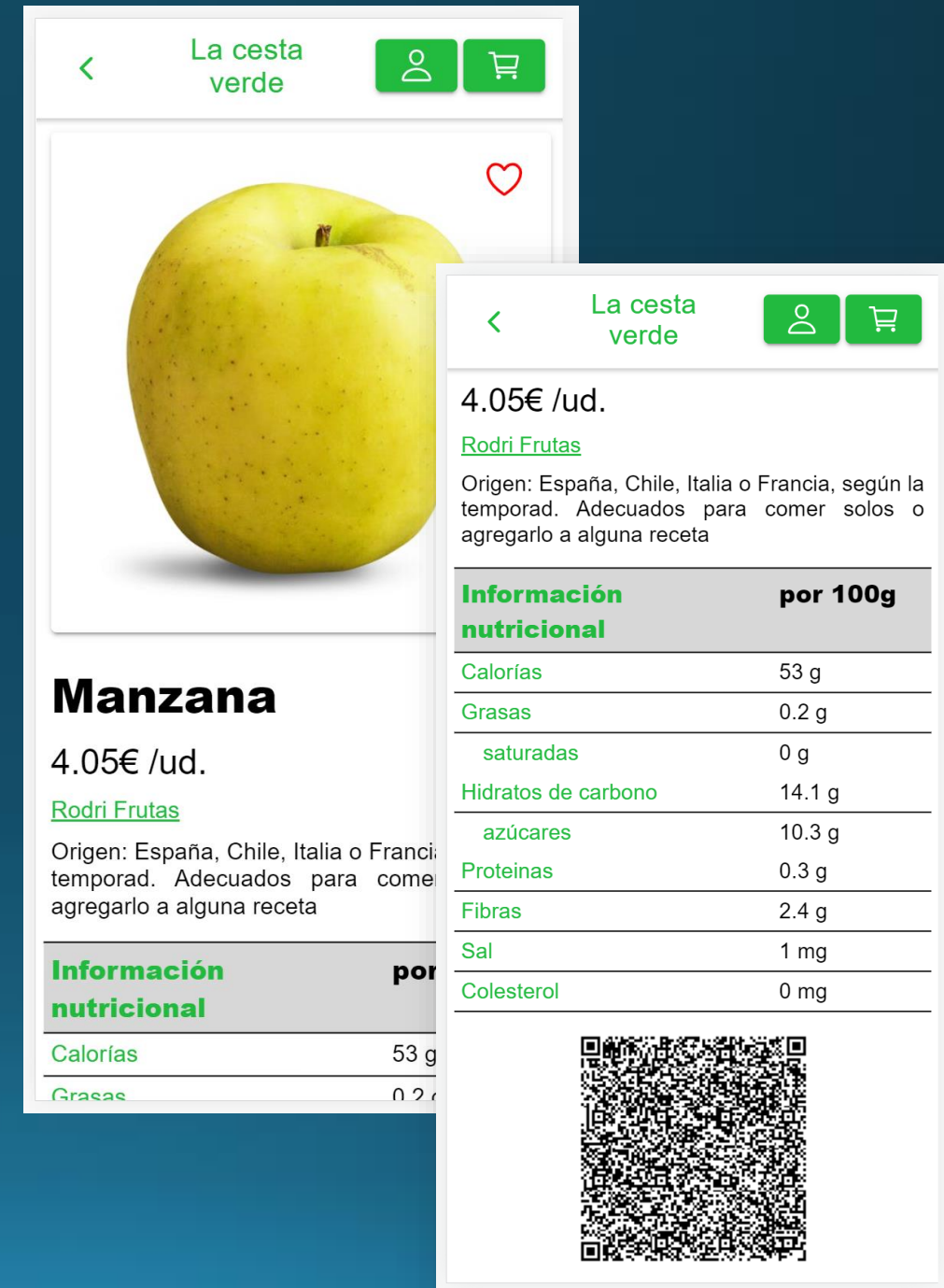
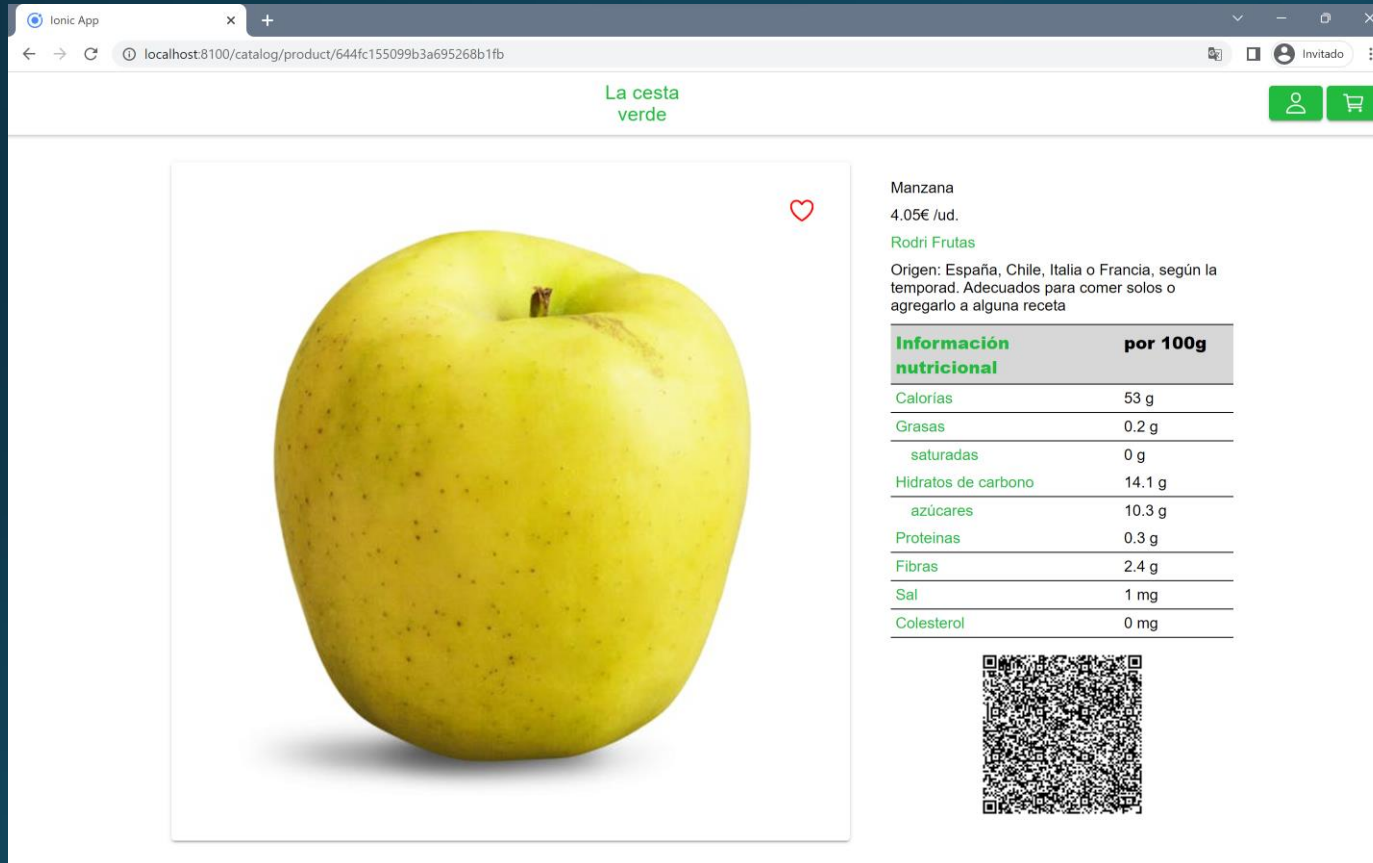
REGISTRAR

¿Ya tienes una cuenta? [Ingresa aquí](#)

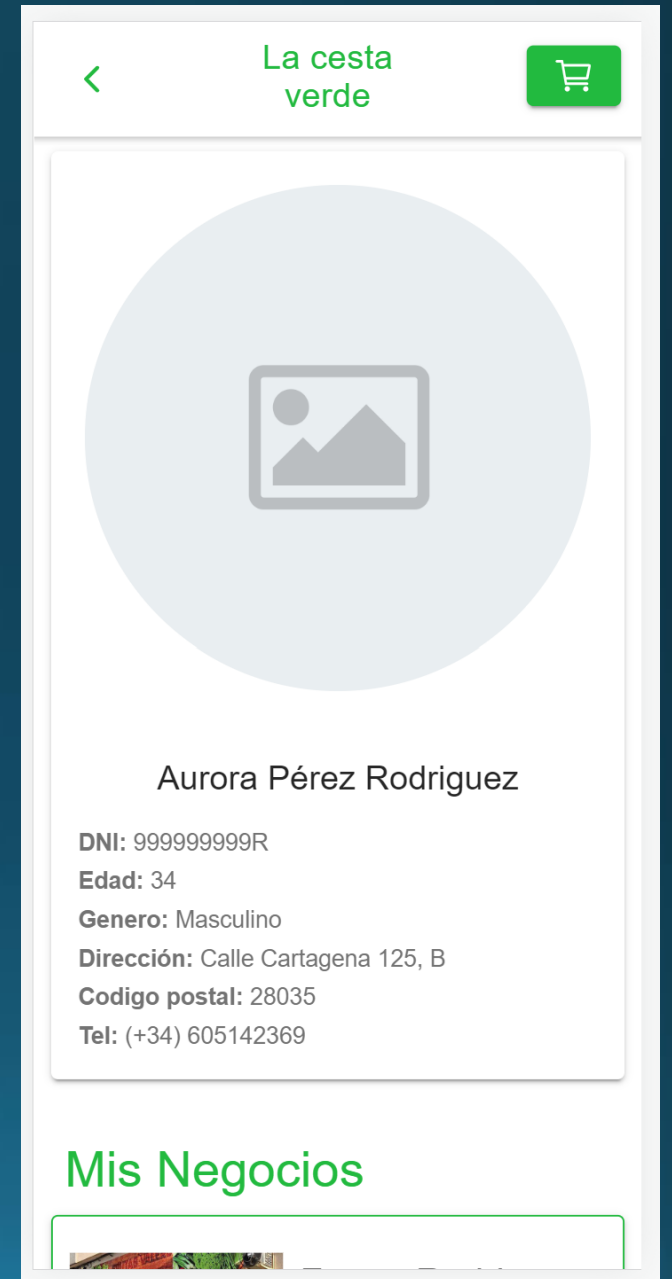
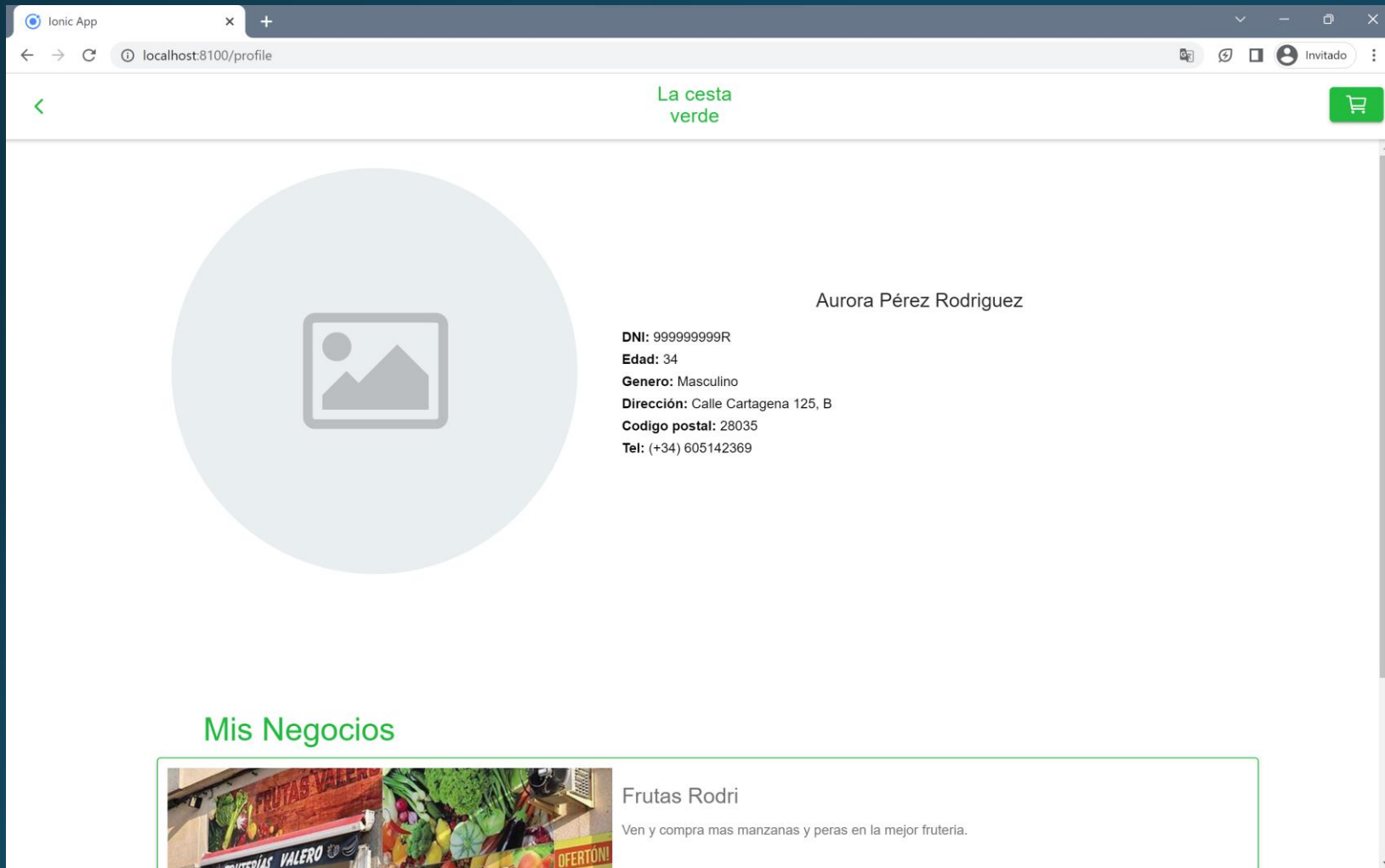
Sección de catálogo



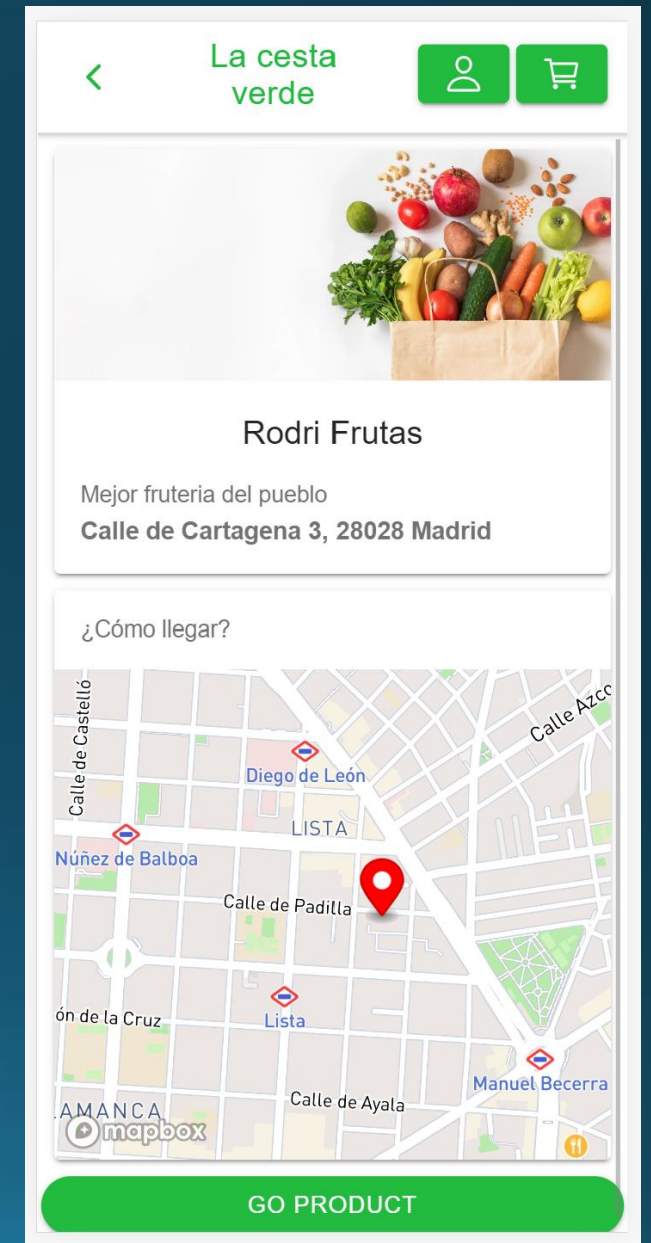
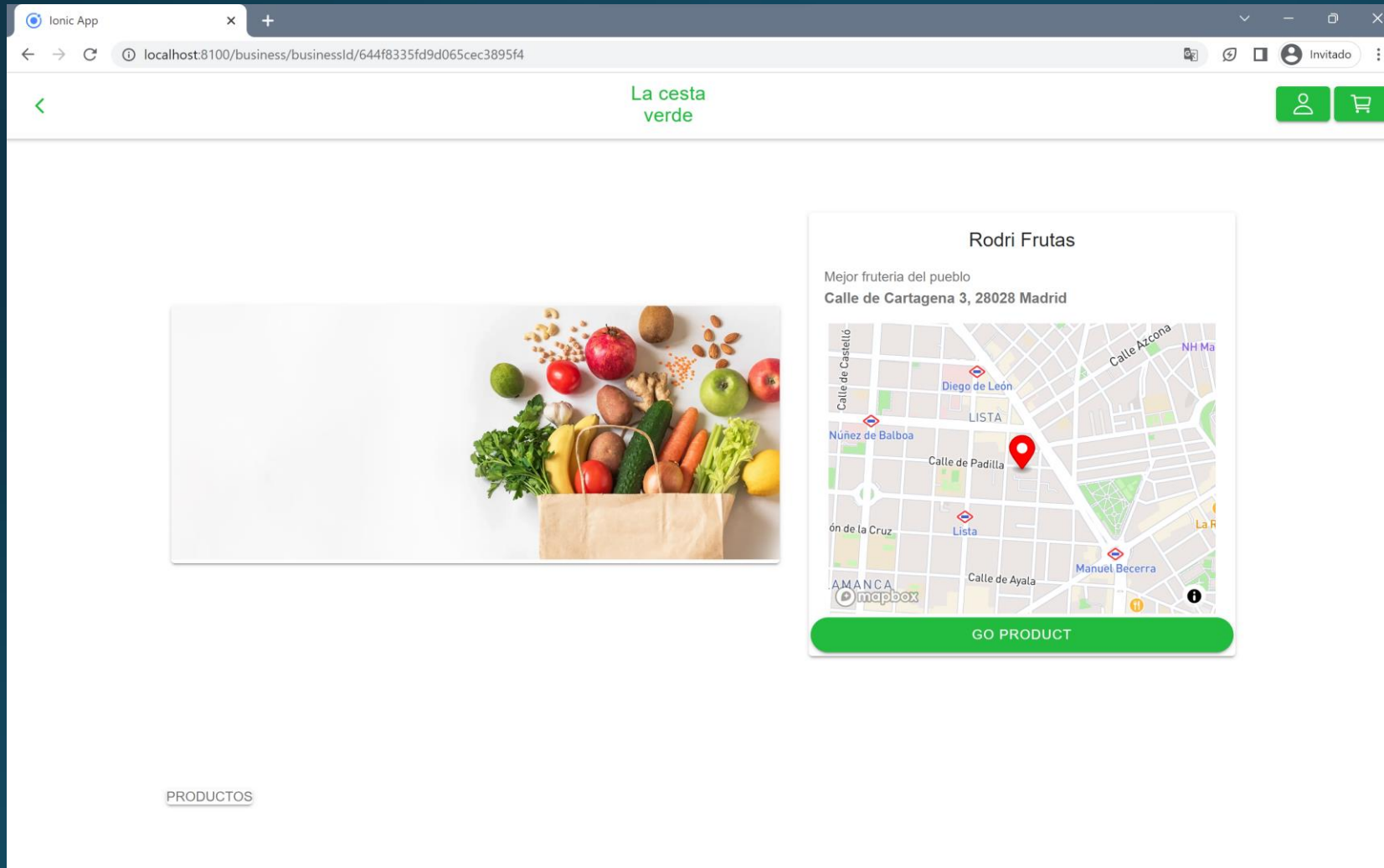
Sección de producto



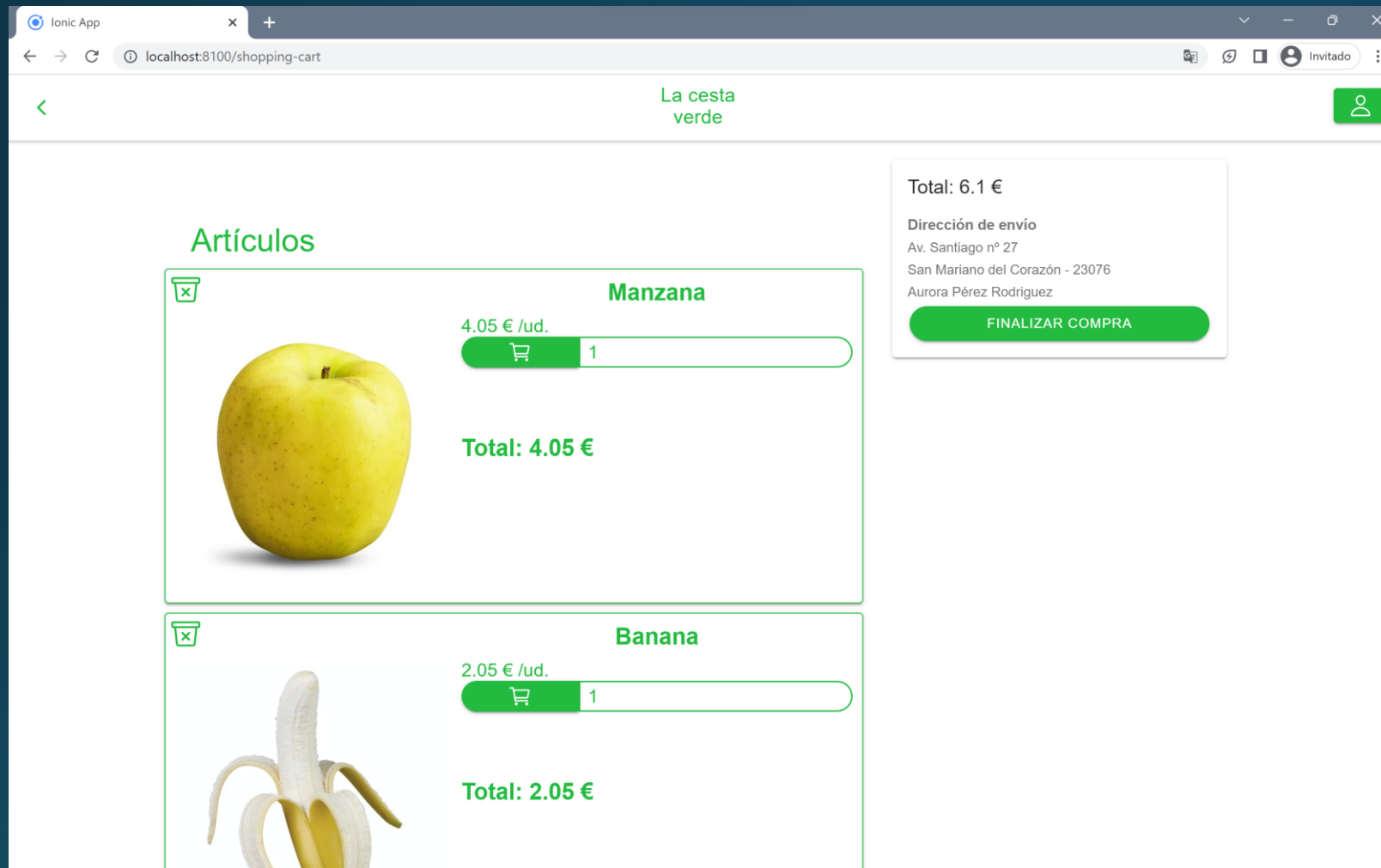
Sección de perfil



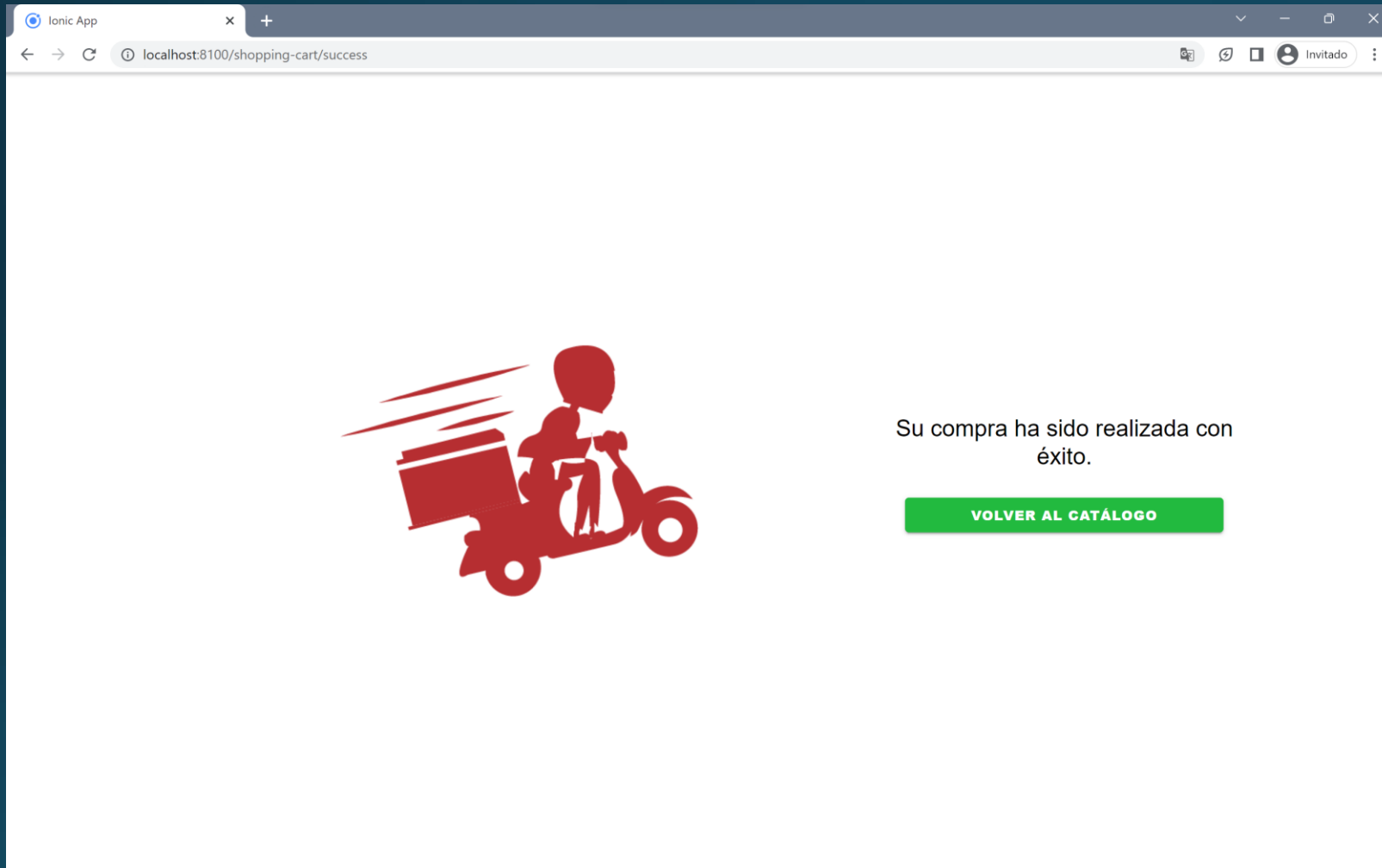
Sección de negocio



Sección de cesta



Sección de cesta



Su compra ha sido realizada con éxito.



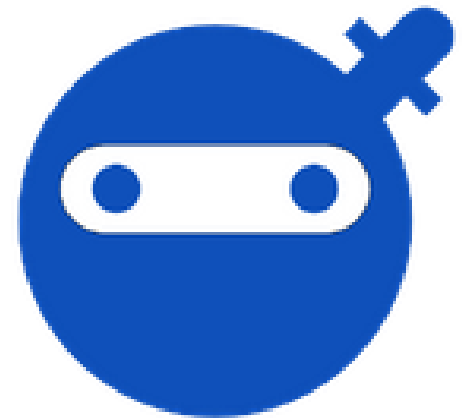
VOLVER AL CATÁLOGO

Aplicaciones Multimedia

Obtención automática de valores nutricionales



fatsecret



Nutrition

Obtención automática de valores nutricionales

The screenshot displays a REST client interface with a POST request to `http://localhost:3000/api/v1/product/nutritionalInfo`. The request body is a JSON object with the property `"productName": "zanahoria"`. The response is shown in the 'Body' tab, displaying a detailed JSON object with various nutritional values for 'zanahoria'.

Request:

```
POST http://localhost:3000/api/v1/product/nutritionalInfo
```

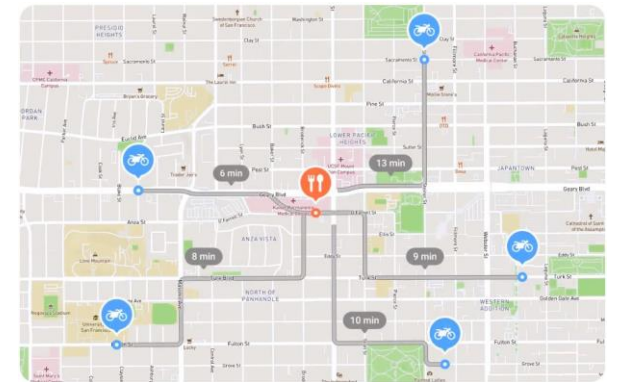
Body (JSON):

```
{  "productName": "zanahoria"}
```

Response (JSON):

```
{  "cantidad": "100g",  "calorias": "34 g",  "grasa_total": "0.2 g",  "grasa_saturada": "0 g",  "proteina": "0.8 g",  "sodio": "57 mg",  "potasio": "30 mg",  "colesterol": "0 mg",  "carbohidratos_total": "8.3 g",  "fibras": "3 g",  "azucar": "3.4 g"}
```

Implementación de mapas



Generación del Código Qr.

angularx-qrcode



Conclusiones

- Correcto uso de aplicaciones híbridas
- Diseño de interacción en base a diferentes dispositivos.
- Compenetración Frontend – Backend – Base de datos
- Uso de APIs y contenido multimedia:
 - Obtención de información nutricional automatizada.
 - Generación de códigos QR.
 - Generación de mapas.
- Definición de estilos y logotipo