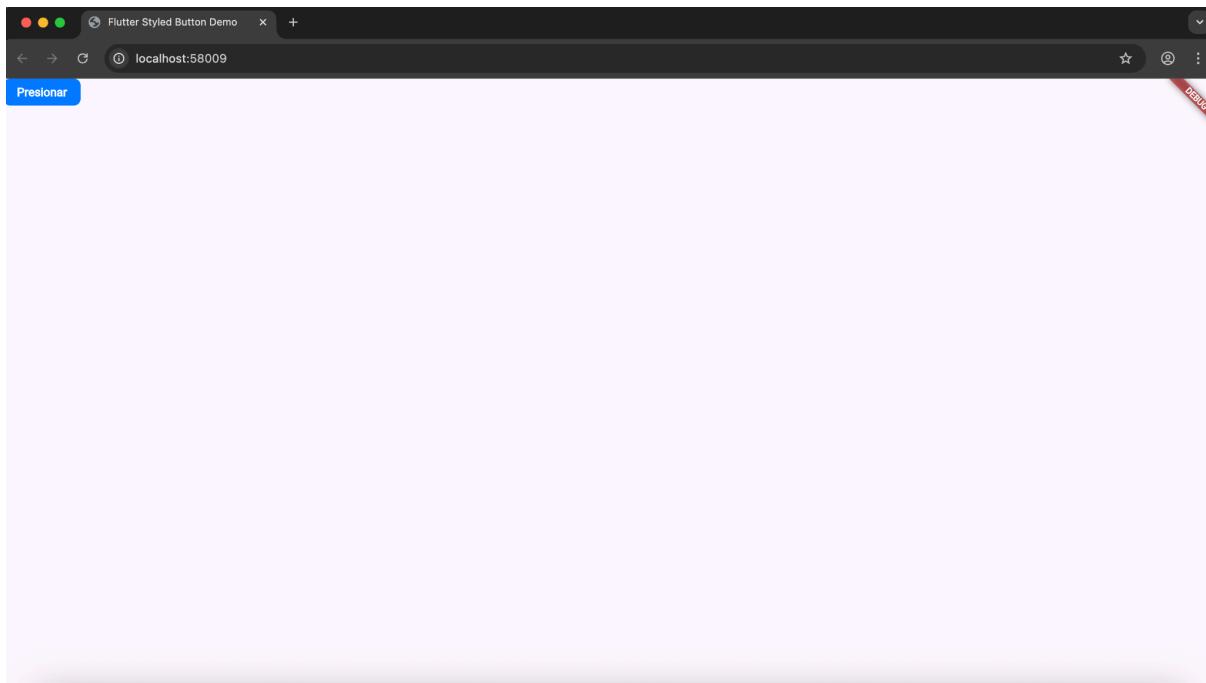


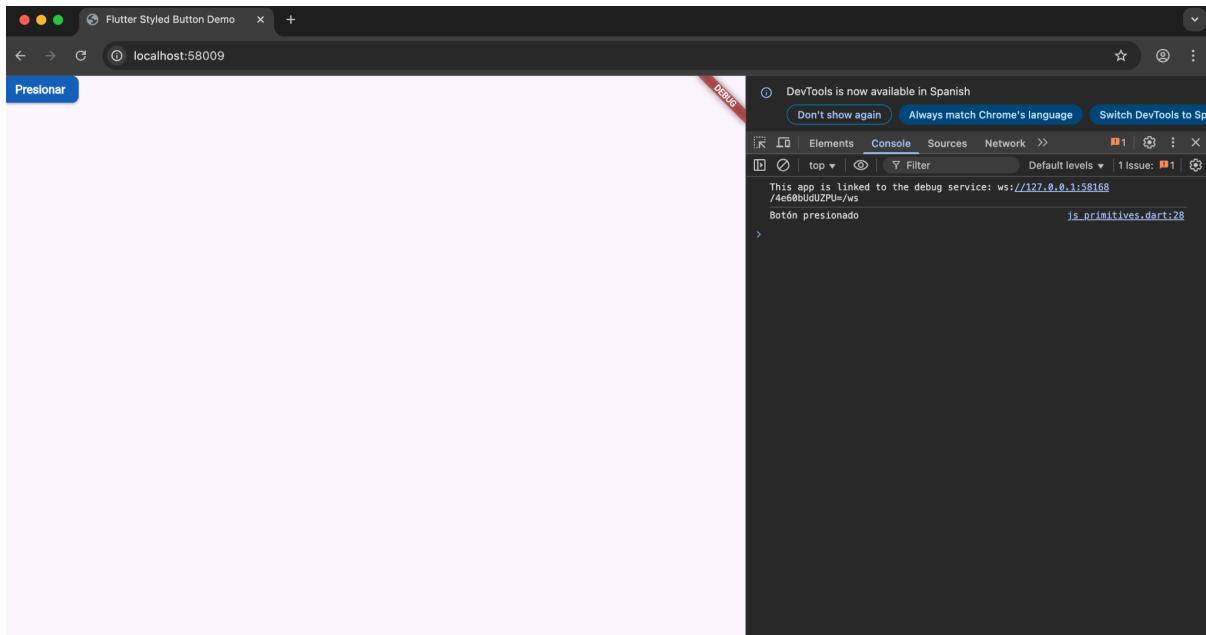
CASO 1

caso 1 base print

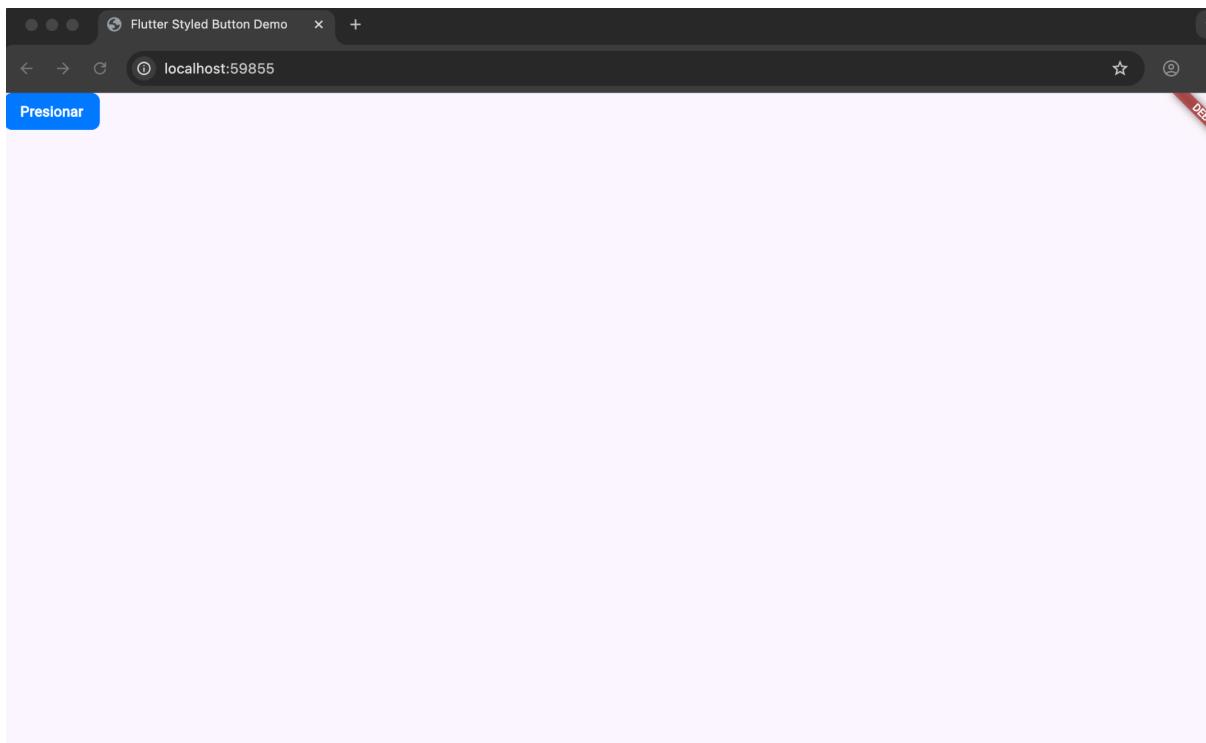
resultados:

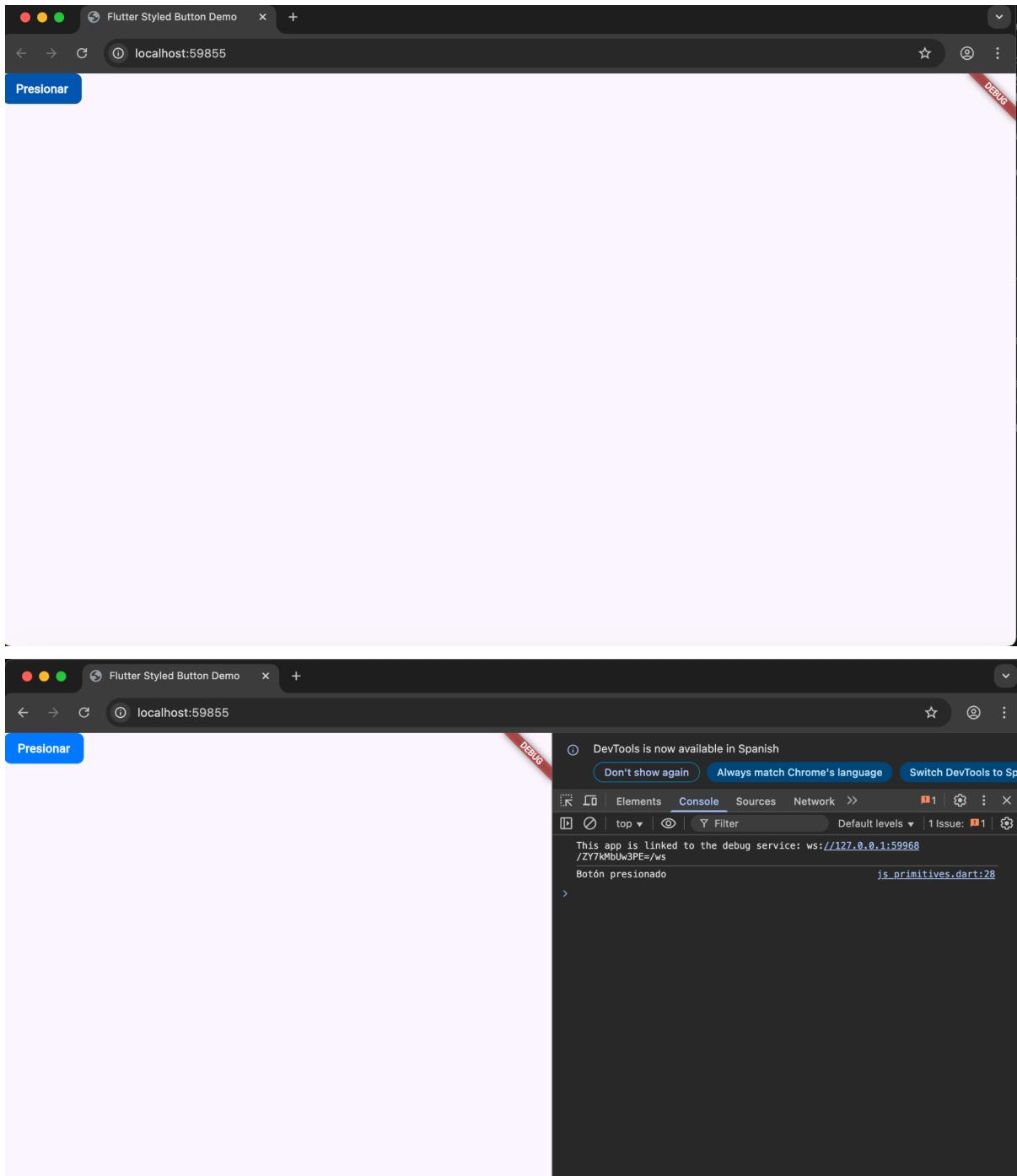
claude:



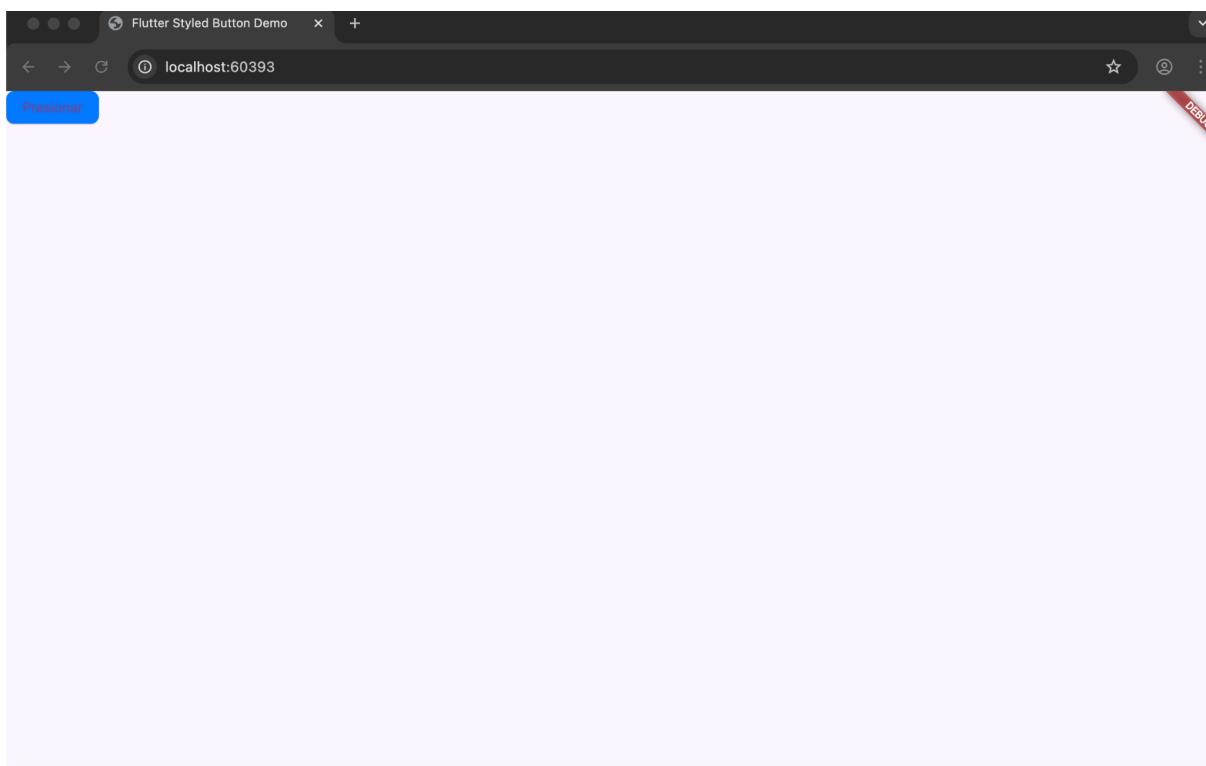
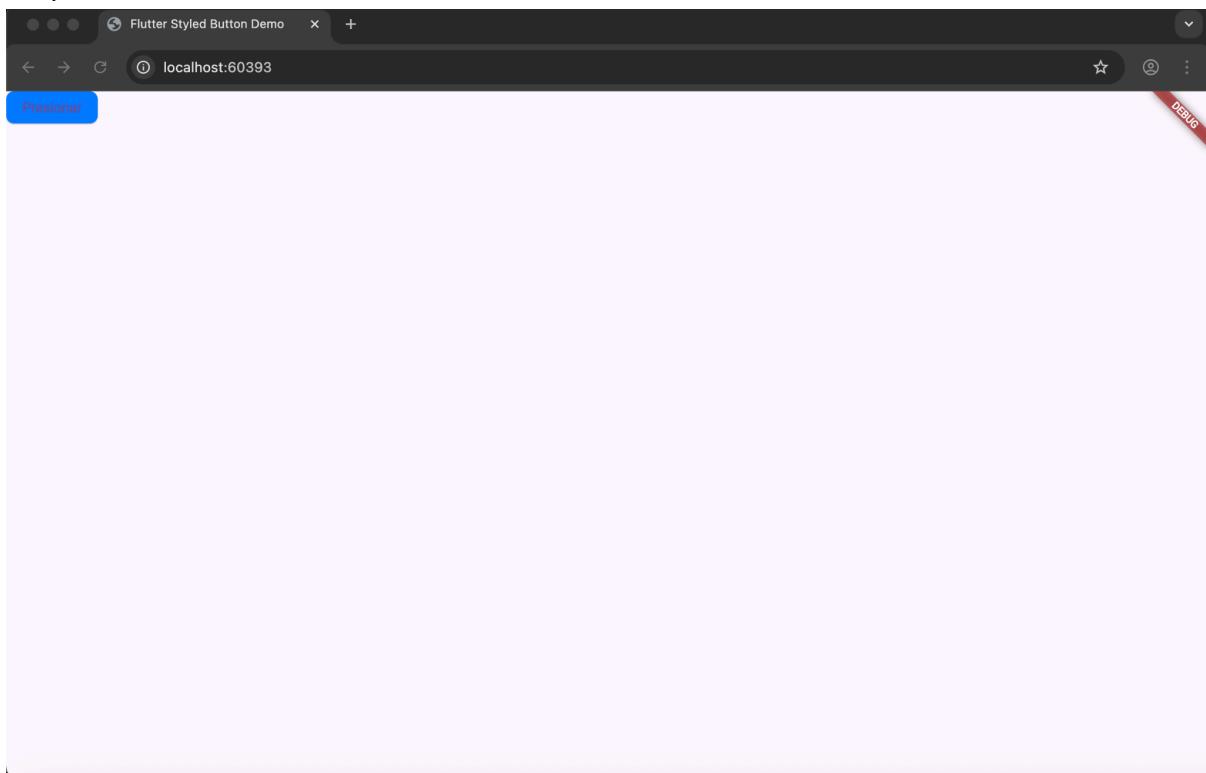


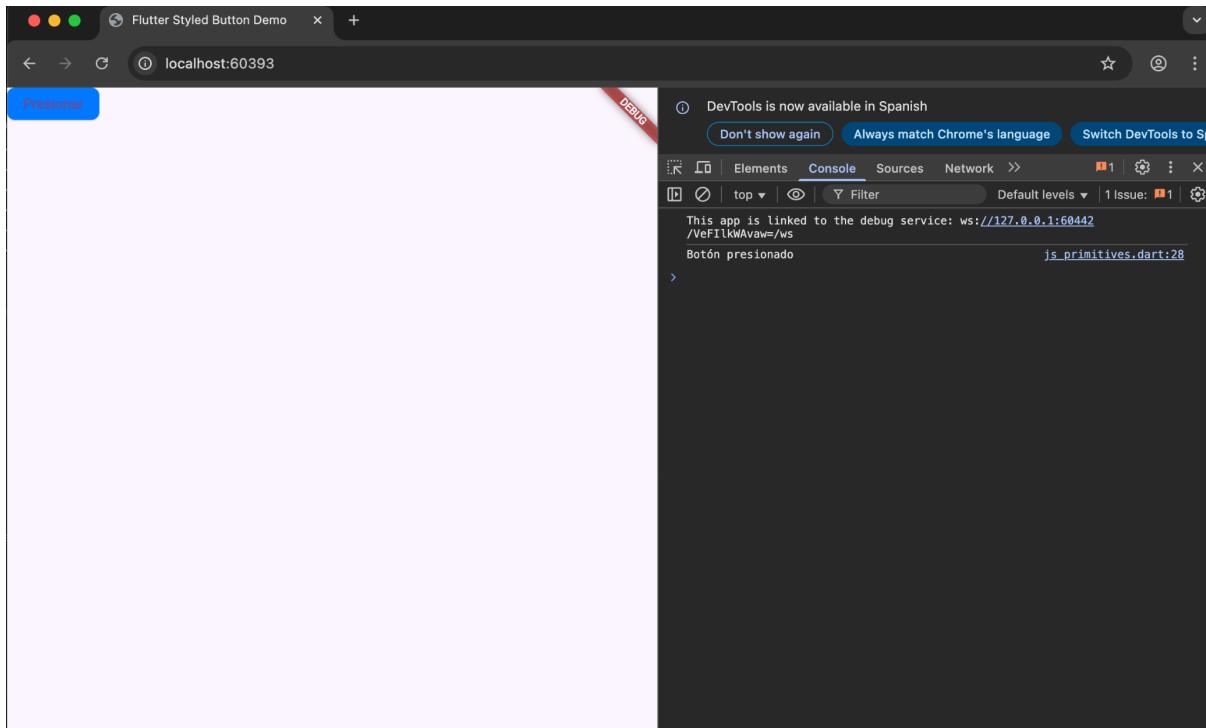
chatgpt:



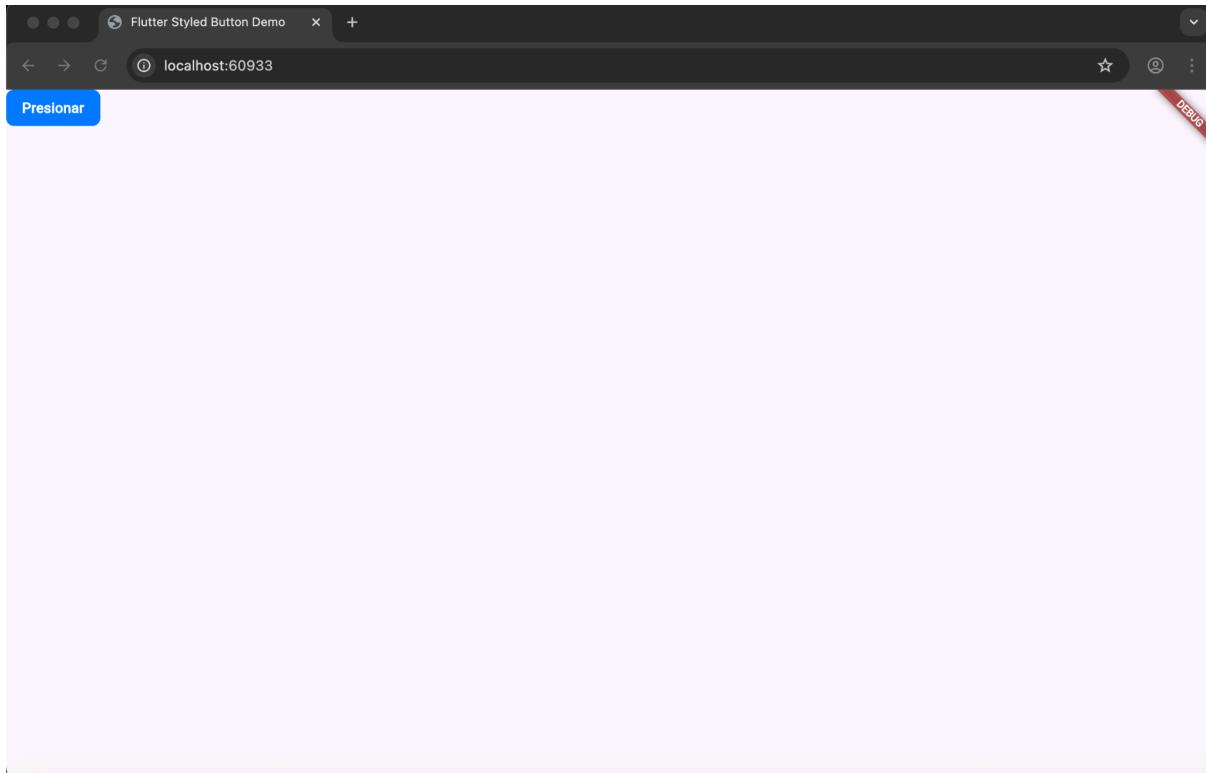


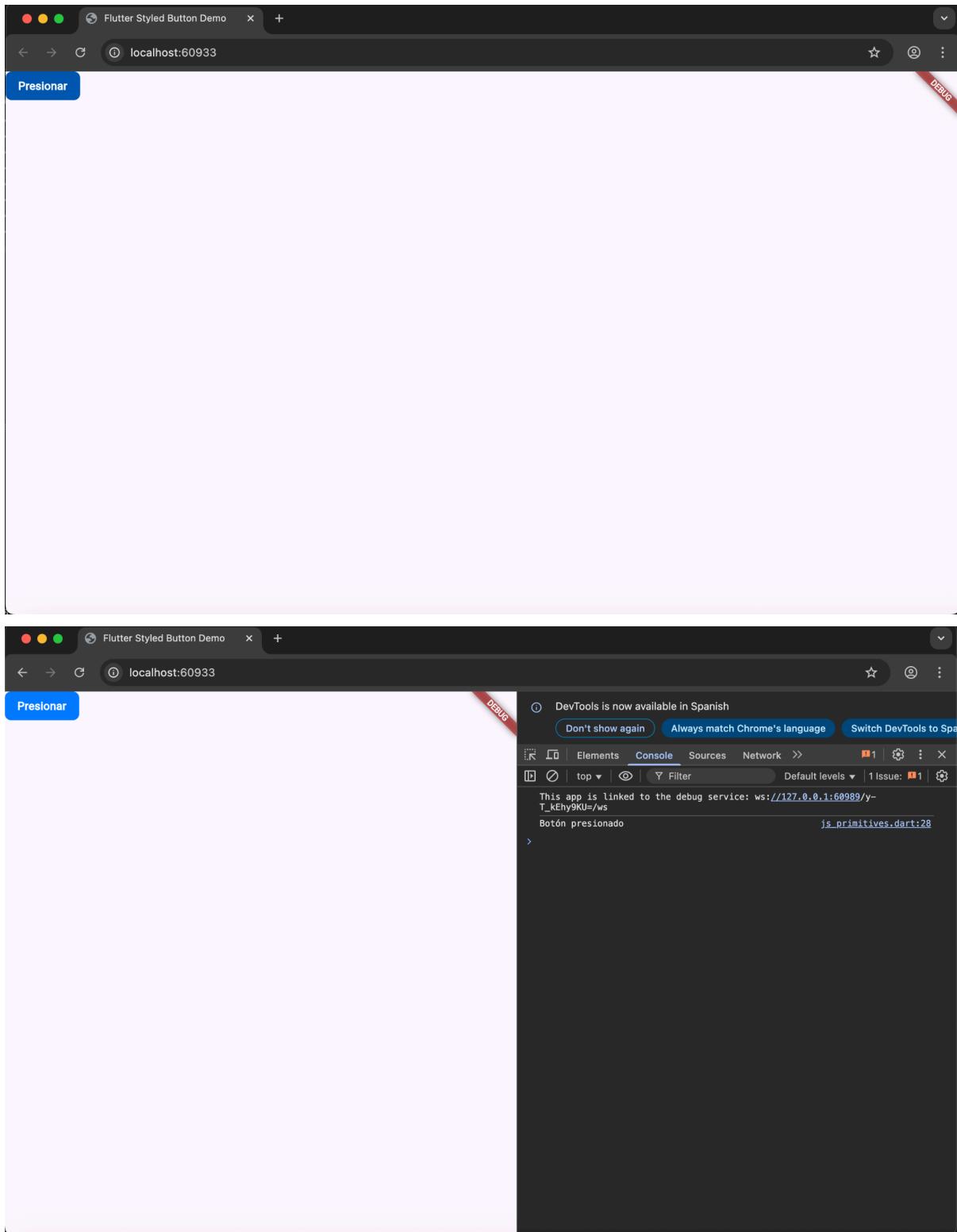
deepseek:





codex:





CASO 2

PROMPT CASO 2

Necesito que traduzcas este componente de React (JavaScript) a Flutter (Dart).

El componente representa una barra lateral de navegación con las siguientes características:

- Debe ocupar toda la altura de la pantalla y estar alineada a la izquierda.
- Color de fondo gris claro (#f8f9fa).
- Contiene 3 ítems de navegación: Inicio, Perfil, Configuración.
- El ítem seleccionado debe resaltarse con fondo azul (#007BFF) y texto blanco.
- Al hacer clic en cada ítem, debe cambiar el contenido mostrado en el panel principal.
- El código debe estar bien estructurado, idealmente con el Sidebar como un widget separado.

A continuación te paso los archivos originales en React:

****App.jsx****

```
```jsx
import React, { useState } from 'react';
import Sidebar from './components/Sidebar';

function App() {
 const [selected, setSelected] = useState('Inicio');

 return (
 <div style={{ display: 'flex' }}>
 <Sidebar onSelect={setSelected} />
 <div style={{ marginLeft: '220px', padding: '20px' }}>
 <h1>{selected}</h1>
 <p>Estás en la sección: {selected}</p>
 </div>
 </div>
);
}

export default App;
```

```

 </div>
 </div>
);
}

export default App;
```
---



**Sidebar.jsx**



```jsx
import React, { useState } from 'react';
import './Sidebar.css';

const Sidebar = ({ onSelect }) => {
 const [activeItem, setActiveItem] = useState('Inicio');

 const items = ['Inicio', 'Perfil', 'Configuración'];

 const handleClick = (item) => {
 setActiveItem(item);
 onSelect(item);
 };

 return (
 <div className="sidebar">
 {items.map((item) => (
 <div
 key={item}
 className={`sidebar-item ${activeItem === item ? 'active' : ''}`}
 onClick={() => handleClick(item)}
 >
 {item}
 </div>
))}
 </div>
);
}

export default Sidebar;
```

```

Sidebar.css

```
```css
.sidebar {
 width: 200px;
 height: 100vh;
 background-color: #f8f9fa;
 padding-top: 20px;
 position: fixed;
 left: 0;
 top: 0;
}

.sidebar-item {
 padding: 12px 16px;
 cursor: pointer;
 color: #000;
}

.sidebar-item.active {
 background-color: #007BFF;
 color: #fff;
}
````
```

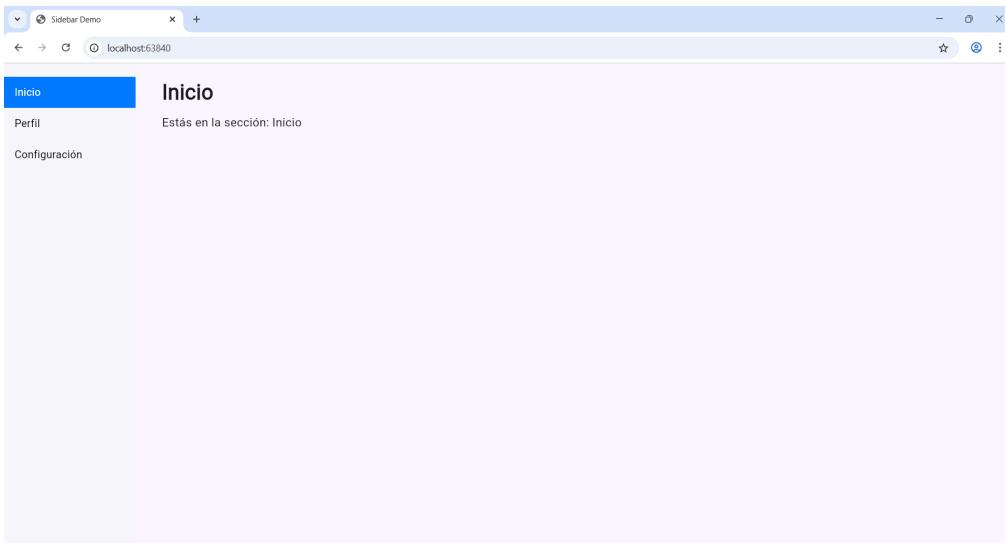


Por favor, devolveme el equivalente en Flutter Web, con:

- Un archivo `main.dart` principal.
- Un widget separado para el Sidebar (por ejemplo `sidebar.dart`).

Gracias

CHATGPT :



CASO DEEPSEEK:

The image displays three separate screenshots of a web browser window titled "Flutter Sidebar Demo" running at "localhost:51638". Each screenshot shows a sidebar navigation menu with three items: "Inicio", "Perfil", and "Configuración".

- Inicio Page:** The "Inicio" item is highlighted in blue. The main content area displays the message "Estás en la sección: Inicio".
- Perfil Page:** The "Perfil" item is highlighted in blue. The main content area displays the message "Estás en la sección: Perfil".
- Configuración Page:** The "Configuración" item is highlighted in blue. The main content area displays the message "Estás en la sección: Configuración".

Each screenshot includes a small red "DESEJO" watermark in the bottom right corner.

CODEX:

The image displays three separate browser windows, each showing a different page of a web application. All three windows have the same header and sidebar navigation.

Header: localhost:52244

Navigation Bar: Includes back, forward, search, and refresh buttons, along with a star icon and a 'DESEJO' button.

Left Sidebar: Contains three items: Inicio (highlighted in blue), Perfil, and Configuración.

Page Content: Shows the current section with a breadcrumb message: "Estás en la sección: [Section Name]".

Screenshot 1: Inicio

Section Name: Inicio

Breadcrumb: Estás en la sección: Inicio

Screenshot 2: Perfil

Section Name: Perfil

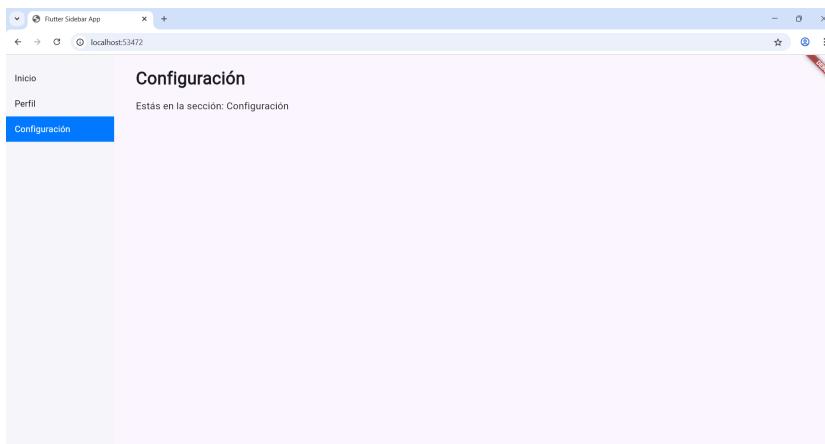
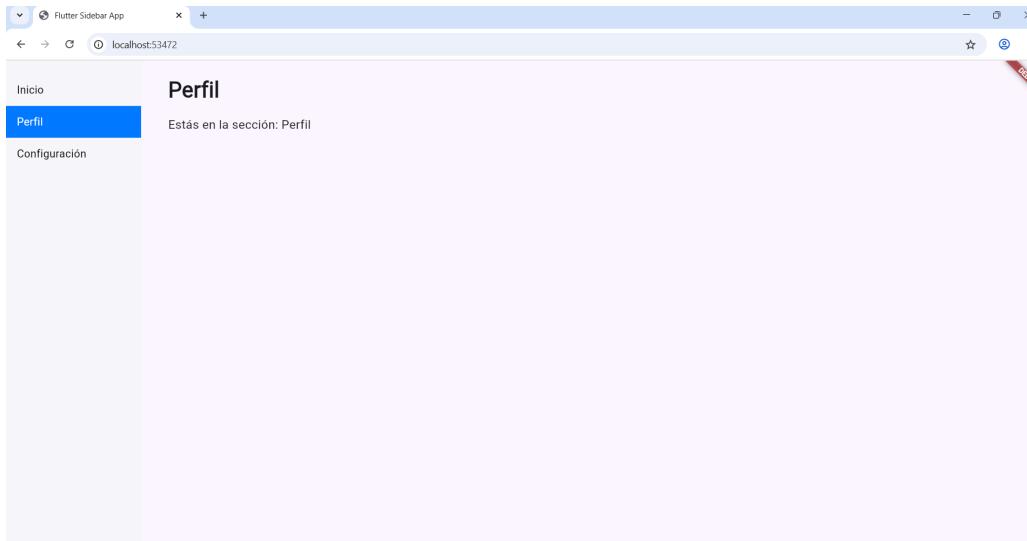
Breadcrumb: Estás en la sección: Perfil

Screenshot 3: Configuración

Section Name: Configuración

Breadcrumb: Estás en la sección: Configuración

Claude:



Comparación Modelos Caso 2

| | Modelo | Sidebar separado | Altura pantalla | Color de fondo | Ítems requeridos | Resaltado ítem activo | Cambio de vista | Estructura general | Cumplimiento genera |
|---|----------|------------------|-----------------|----------------|------------------|-----------------------|-----------------|--------------------|---------------------|
| 1 | Claude | ✓ | ✓ | ✓ #f8f9fa | ✓ 3 | ✓ #007BFF + blanco | ✓ | ✓ buena | Muy alto |
| 2 | ChatGPT | ✓ | ✓ | ✓ #f8f9fa | ✓ 3 | ✓ #007BFF + blanco | ✓ | ✓ buena | Muy alto |
| 3 | DeepSeek | ✓ | ✓ | ✓ #f8f9fa | ✓ 3 | ✗ | ✓ | ✓ simple | Medio |
| 4 | Codex | ✓ | ✓ | ✓ #f8f9fa | ✓ 3 | ✓ #007BFF + blanco | ✓ | ✓ buena | Muy alto |

CASO 3

Prompt

Necesito traducir esta pequeña aplicación SPA de React (JavaScript) a Flutter (Dart para Web). La aplicación tiene las siguientes características:

- Debe incluir tres vistas: Inicio, Perfil y Configuración.
- Utiliza React Router para navegación sin recargar la página.
- Tiene un menú de navegación persistente (lateral o superior) que permite cambiar entre vistas.
- El estado de usuario (nombre, email, etc.) se comparte entre las vistas.
- Se deben pasar datos del Perfil a la vista de Configuración.
- Cada vista debe estar en un componente o widget separado.
- El diseño debe ser simple, con fondo claro y resaltado visual en la vista activa.

Te paso el código en React a continuación:

App.css:

```
.app-container {
  min-height: 100vh;
  background: #f7f8fa;
  display: flex;
  flex-direction: column;
}

.main-content {
  flex: 1;
  padding: 2rem;
}
```

App.jsx:

```
import React from 'react';
import { BrowserRouter as Router, Routes, Route } from "react-router-dom";
import { UserProvider } from "./UserContext";
import NavBar from "./NavBar";
import Home from "./Home";
import Profile from "./Profile";
import Settings from "./Settings";
import "./App.css";

function App() {
  return (
    <UserProvider>
      <Router>
        <div className="app-container">
          <NavBar />
          <main className="main-content">
            <Routes>
              <Route path="/" element={<Home />} />
              <Route path="/profile" element={<Profile />} />
              <Route path="/settings" element={<Settings />} />
            </Routes>
          </main>
        </div>
      </Router>
    </UserProvider>
  );
}

export default App;
```

CustomButton.css:

```
.custom-btn {
  background: #007BFF;
  color: #fff;
  font-weight: bold;
  text-align: center;
  padding: 8px 16px;
```

```
border: none;
border-radius: 8px;
cursor: pointer;
transition: background 0.2s;
display: inline-block;
}

.custom-btn:hover {
  background: #0056b3;
}
```

CustomButton.jsx:

```
import React from "react";
import "./CustomButton.css";

const CustomButton = ({ children, onClick, type = "button" }) => (
  <button className="custom-btn" type={type} onClick={onClick}>
    {children}
  </button>
);

export default CustomButton;
```

Home.jsx:

```
import React from "react";

const Home = () => (
  <div>
    <h2>Home</h2>
    <p>Welcome to the Home page!</p>
  </div>
);

export default Home;
```

main.jsx:

```
import { StrictMode } from 'react'
import { createRoot } from 'react-dom/client'
import App from './App.jsx'
```

```
createRoot(document.getElementById('root')).render(  
  <StrictMode>  
    <App />  
  </StrictMode>,  
)
```

NavBar.css:

```
.navbar {  
  display: flex;  
  gap: 2rem;  
  background: #fff;  
  border-bottom: 1px solid #e0e0e0;  
  padding: 1rem 2rem;  
  font-size: 1.1rem;  
}  
  
.navbar a {  
  color: #333;  
  text-decoration: none;  
  padding: 0.5rem 1rem;  
  border-radius: 6px;  
  transition: background 0.2s, color 0.2s;  
}  
  
.navbar a.active {  
  background: #e3eafe;  
  color: #1976d2;  
  font-weight: bold;  
}  
  
.navbar a:hover {  
  background: #f0f4ff;  
}
```

NavBar.jsx:

```
import React from "react";  
import { NavLink } from "react-router-dom";  
import "./NavBar.css";
```

```

const NavBar = () => {
  return (
    <nav className="navbar">
      <NavLink to="/" end className={({ isActive }) => isActive ? "active" : undefined}>
        Home
      </NavLink>
      <NavLink to="/profile" className={({ isActive }) => isActive ? "active" : undefined}>
        Profile
      </NavLink>
      <NavLink to="/settings" className={({ isActive }) => isActive ? "active" : undefined}>
        Settings
      </NavLink>
    </nav>
  );
};

export default NavBar;

```

Profile.jsx:

```

import React from "react";
import { useUser } from "./UserContext";
import { useNavigate } from "react-router-dom";
import CustomButton from "./CustomButton";

const Profile = () => {
  const { user } = useUser();
  const navigate = useNavigate();

  const handleGoToSettings = () => {
    navigate("/settings", { state: { user } });
  };

  return (
    <div>
      <h2>Profile</h2>
      <p><strong>Name:</strong> {user.name}</p>
      <p><strong>Email:</strong> {user.email}</p>
    </div>
  );
};

```

```

<CustomButton onClick={() => { console.log("Botón presionado");
handleGoToSettings(); }}>
    Edit in Settings
</CustomButton>
</div>
);
};

export default Profile;

Settings.jsx:
import React from "react";
import { useLocation } from "react-router-dom";
import { useUser } from "./UserContext";
import CustomButton from "./CustomButton";

const Settings = () => {
    const { user, setUser } = useUser();
    const location = useLocation();
    const profileData = location.state?.user;

    const [form, setForm] = React.useState(profileData || user);

    const handleChange = (e) => {
        setForm({ ...form, [e.target.name]: e.target.value });
    };

    return (
        <div>
            <h2>Settings</h2>
            <form
                onSubmit={(e) => {
                    e.preventDefault();
                    setUser(form);
                    console.log("Botón presionado");
                }}
            >
                <div>
                    <label>Name: </label>
                    <input name="name" value={form.name} onChange={handleChange} />

```

```
</div>
<div>
  <label>Email: </label>
  <input name="email" value={form.email} onChange={handleChange} />
</div>
<CustomButton type="submit">Save</CustomButton>
</form>
</div>
);
};

export default Settings;
```

UserContex.jsx:

```
import React, { createContext, useContext, useState } from "react";

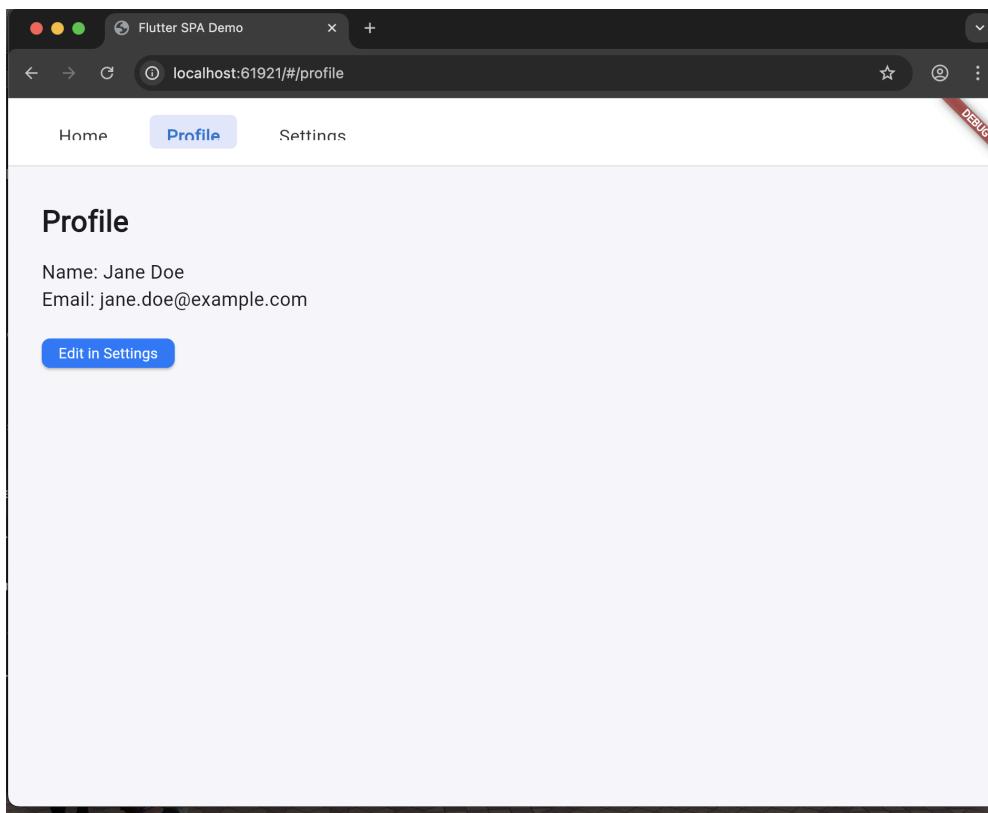
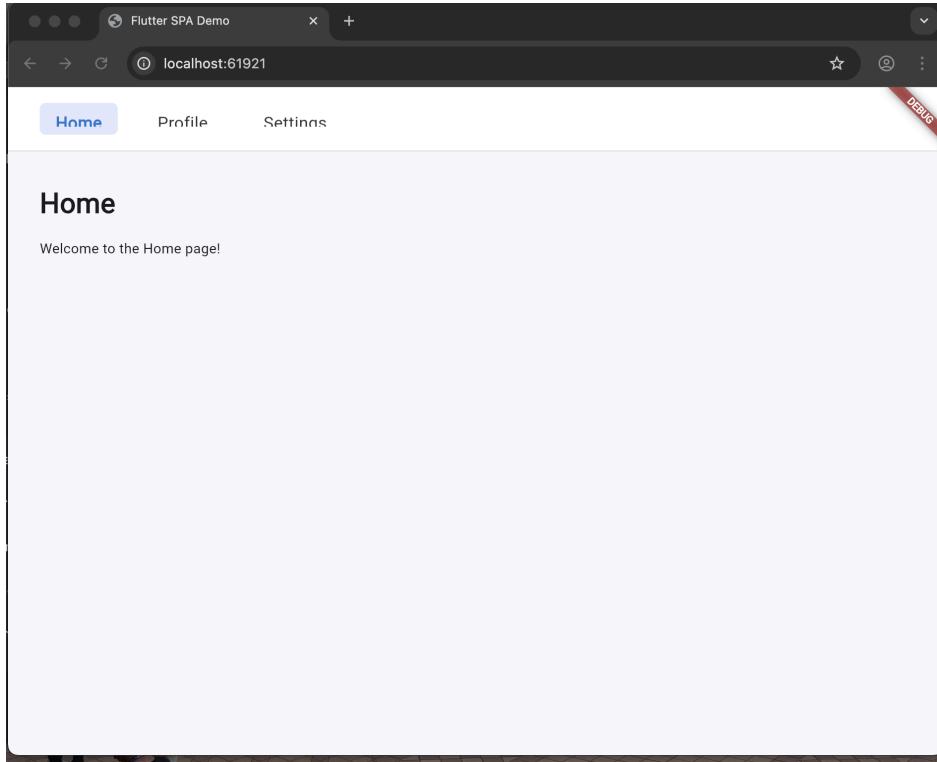
const UserContext = createContext();

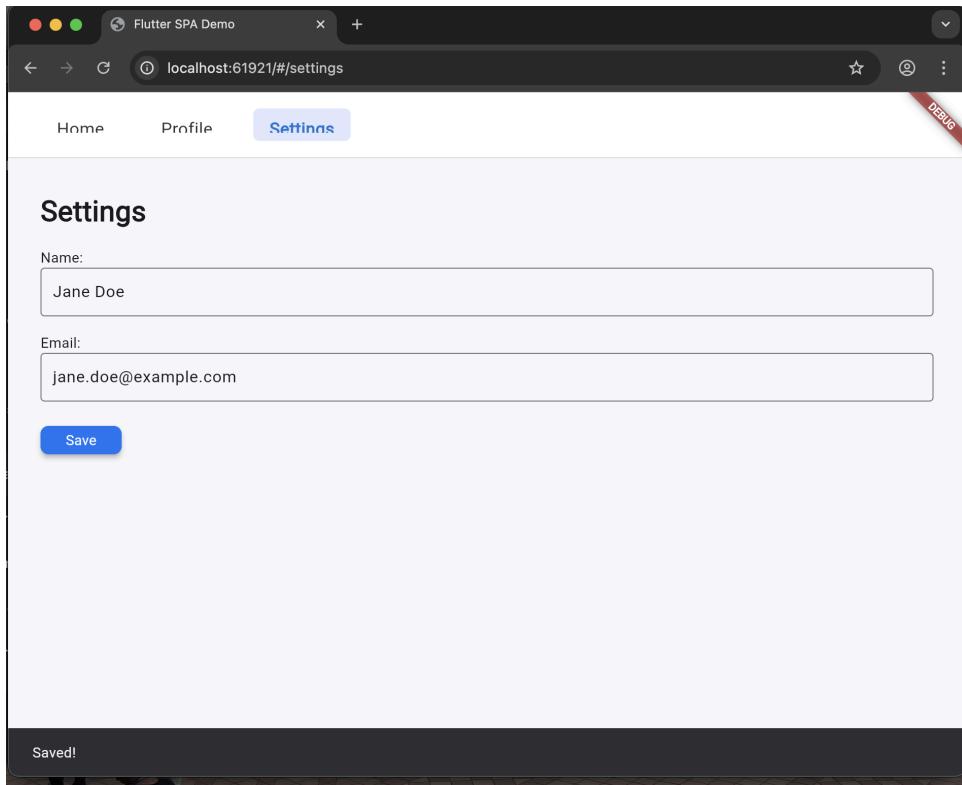
export const useUser = () => useContext(UserContext);

export const UserProvider = ({ children }) => {
  const [user, setUser] = useState({
    name: "Jane Doe",
    email: "jane.doe@example.com",
  });

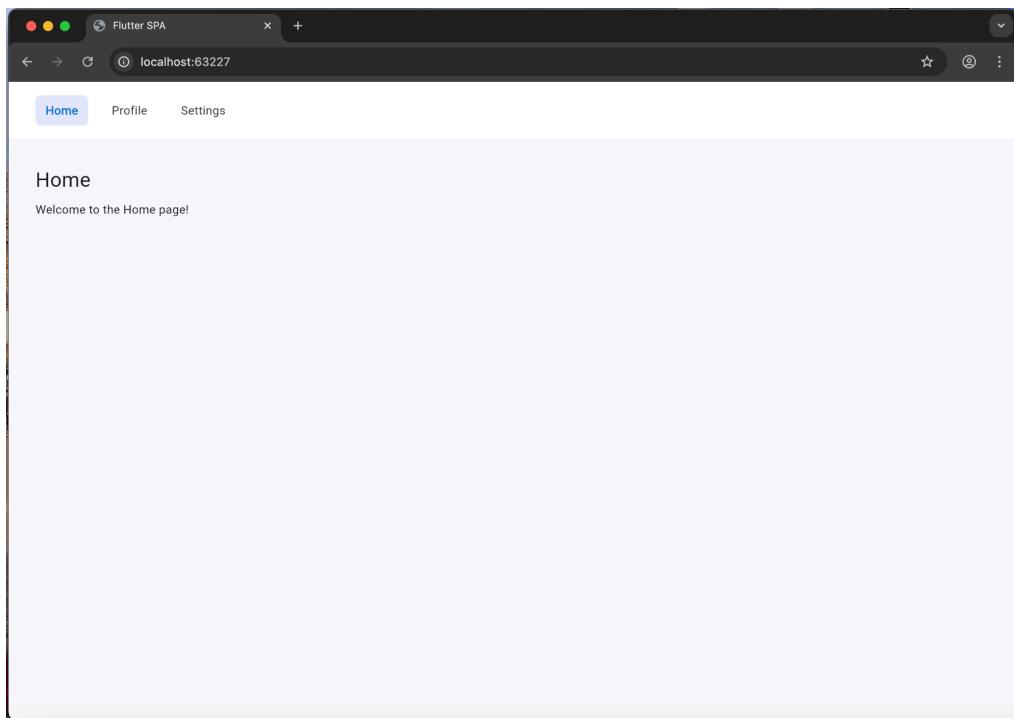
  return (
    <UserContext.Provider value={{ user, setUser }}>
      {children}
    </UserContext.Provider>
  );
};
```

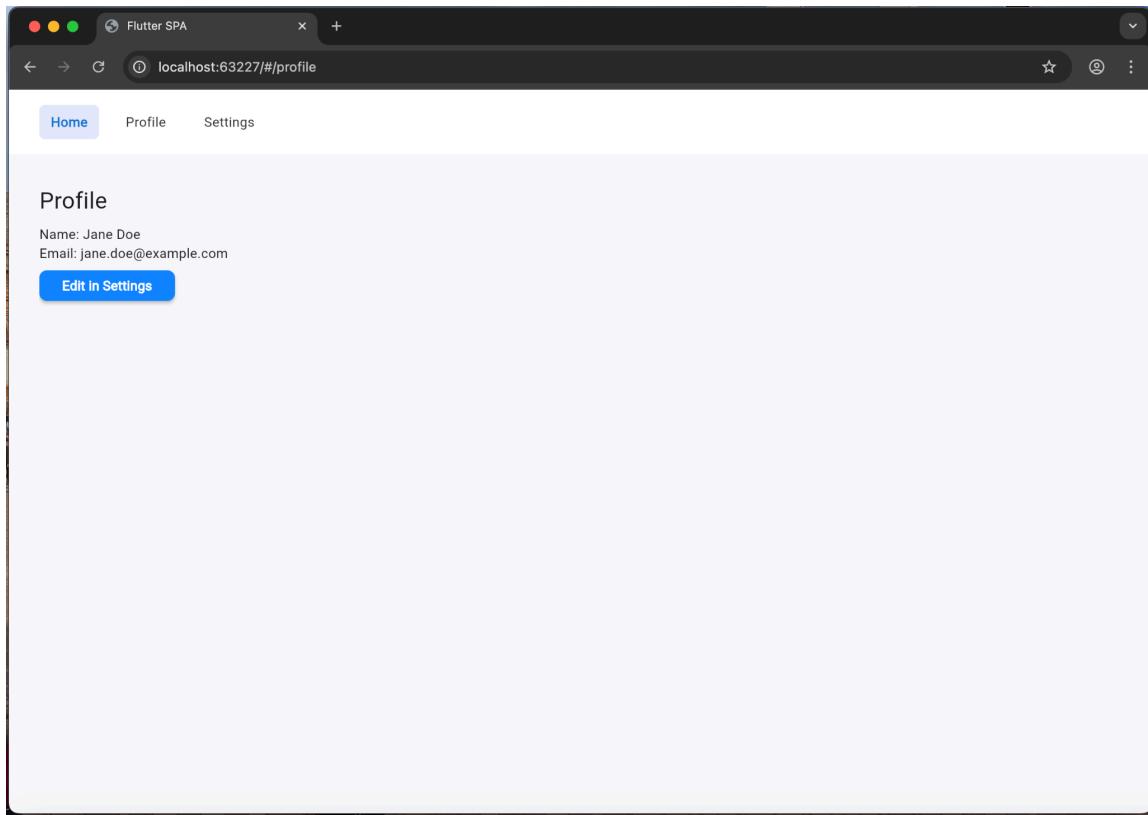
ChatGPT

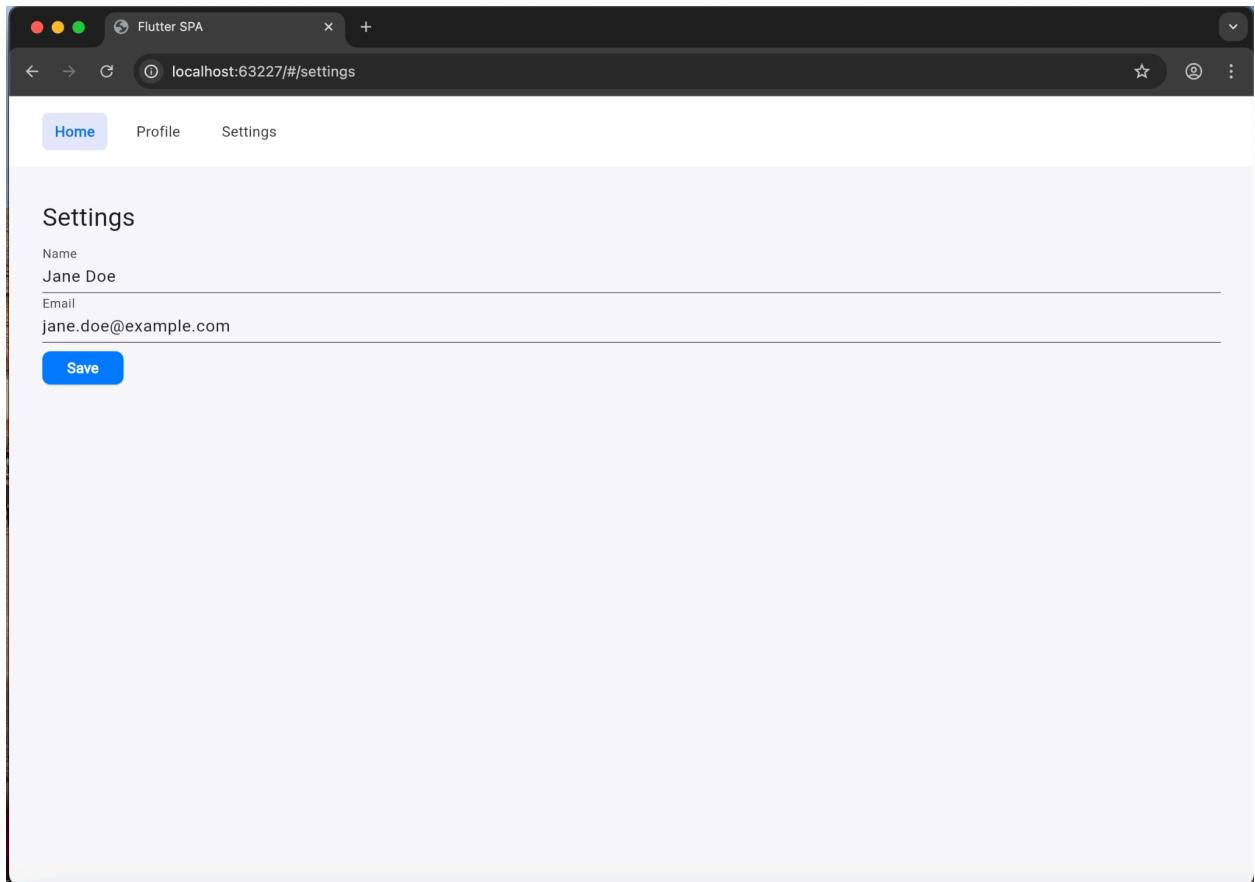




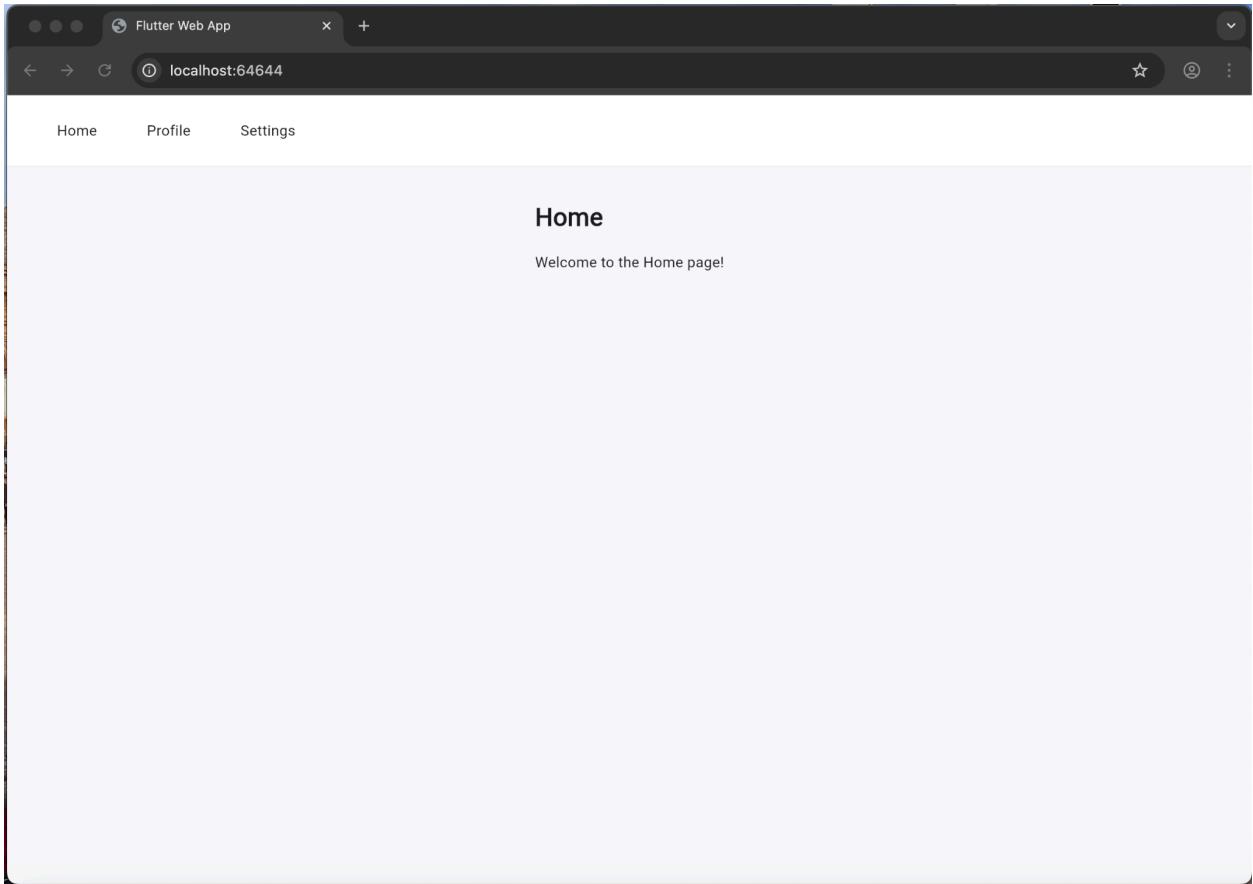
CODEX

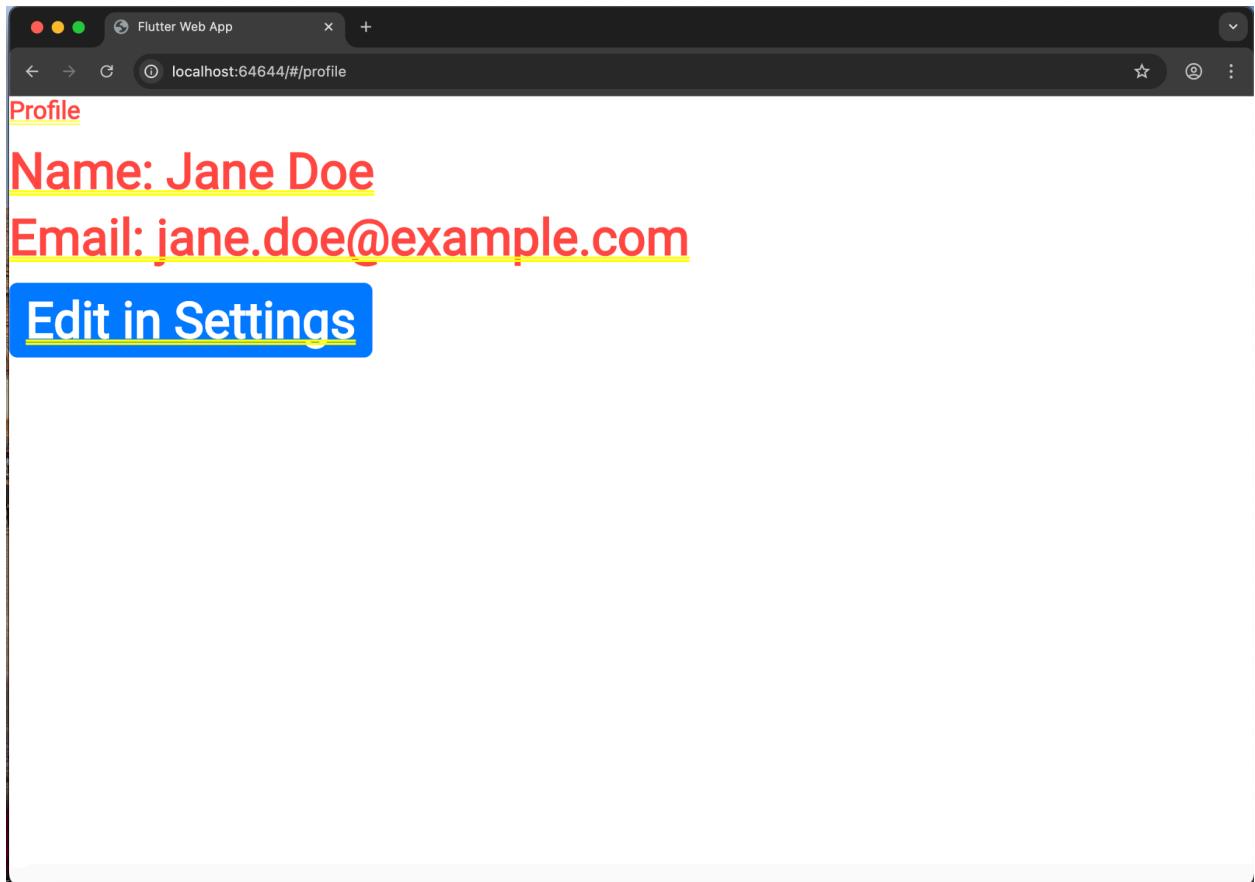


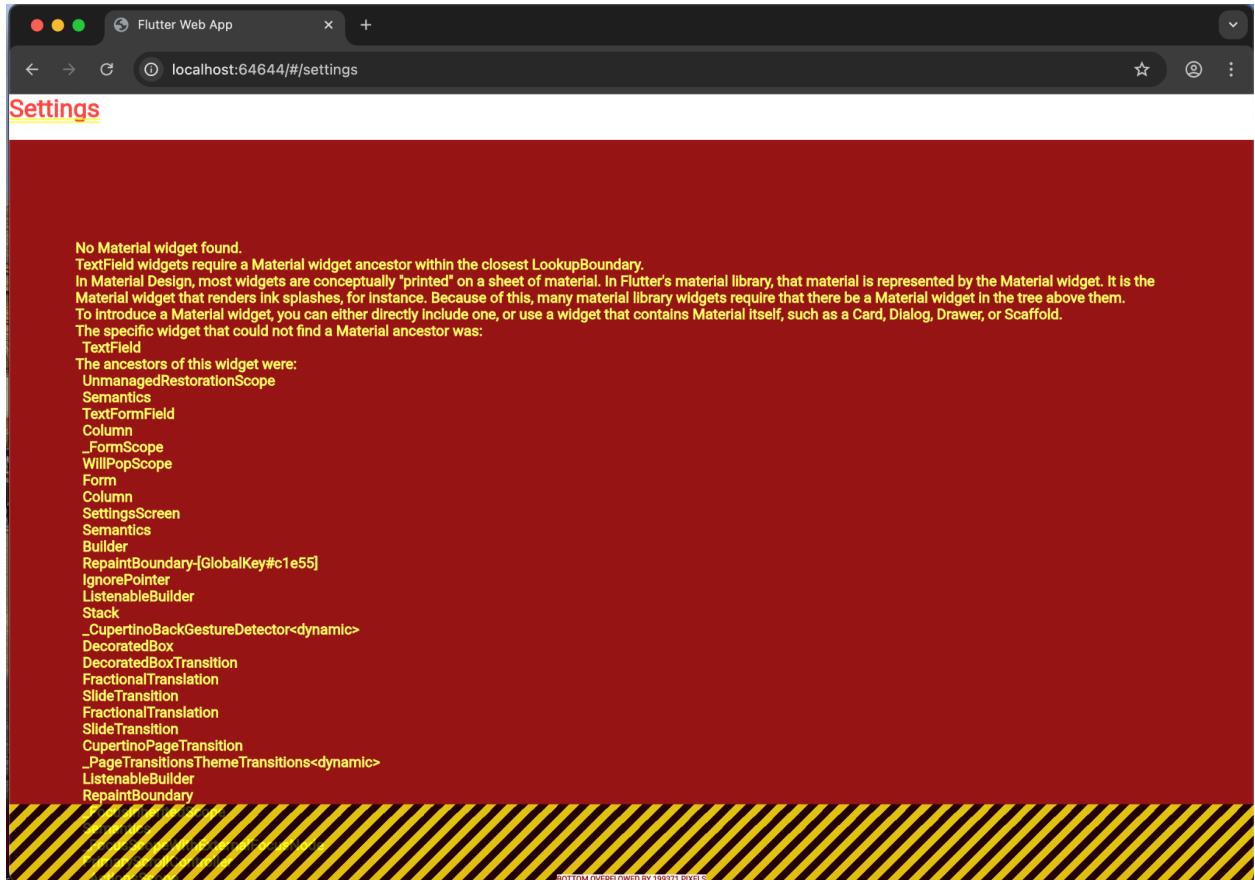




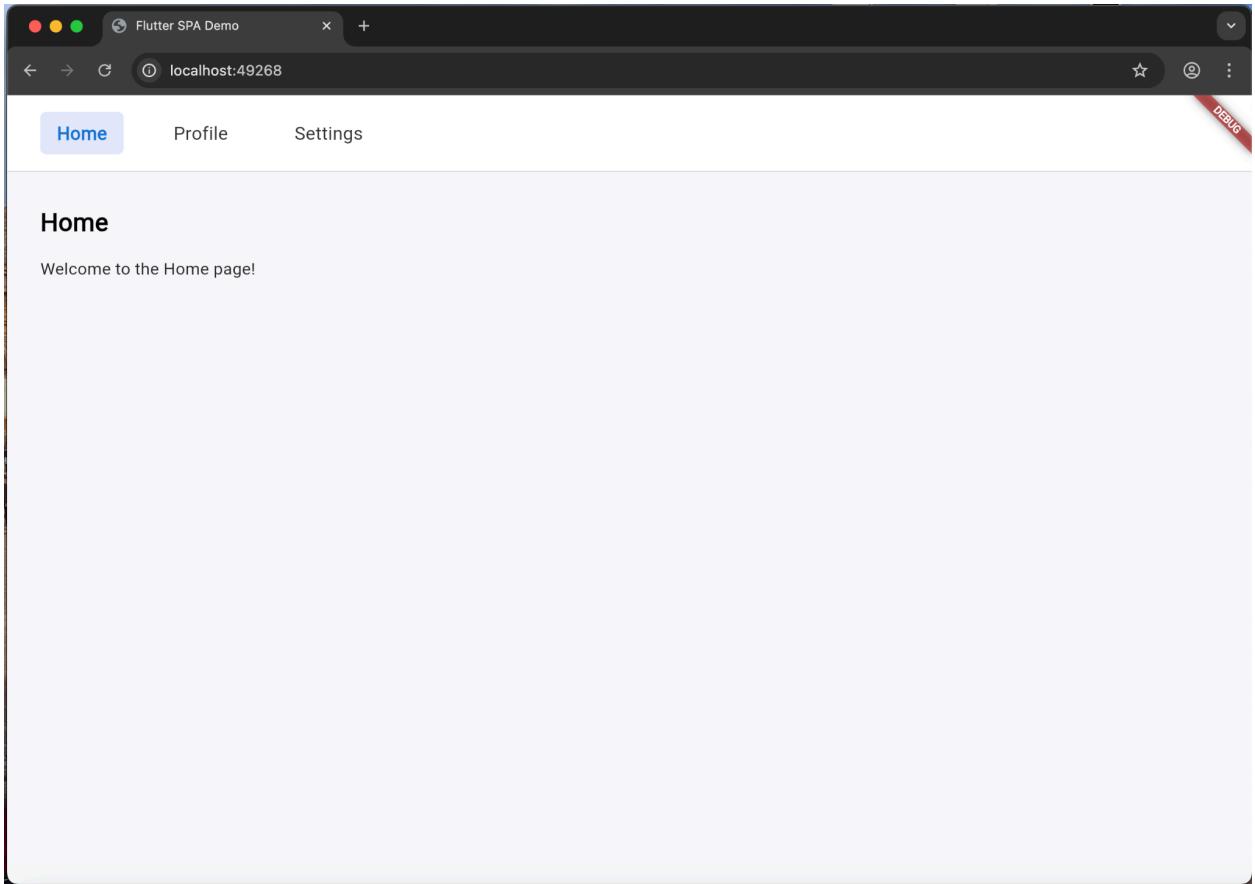
DeepSeek

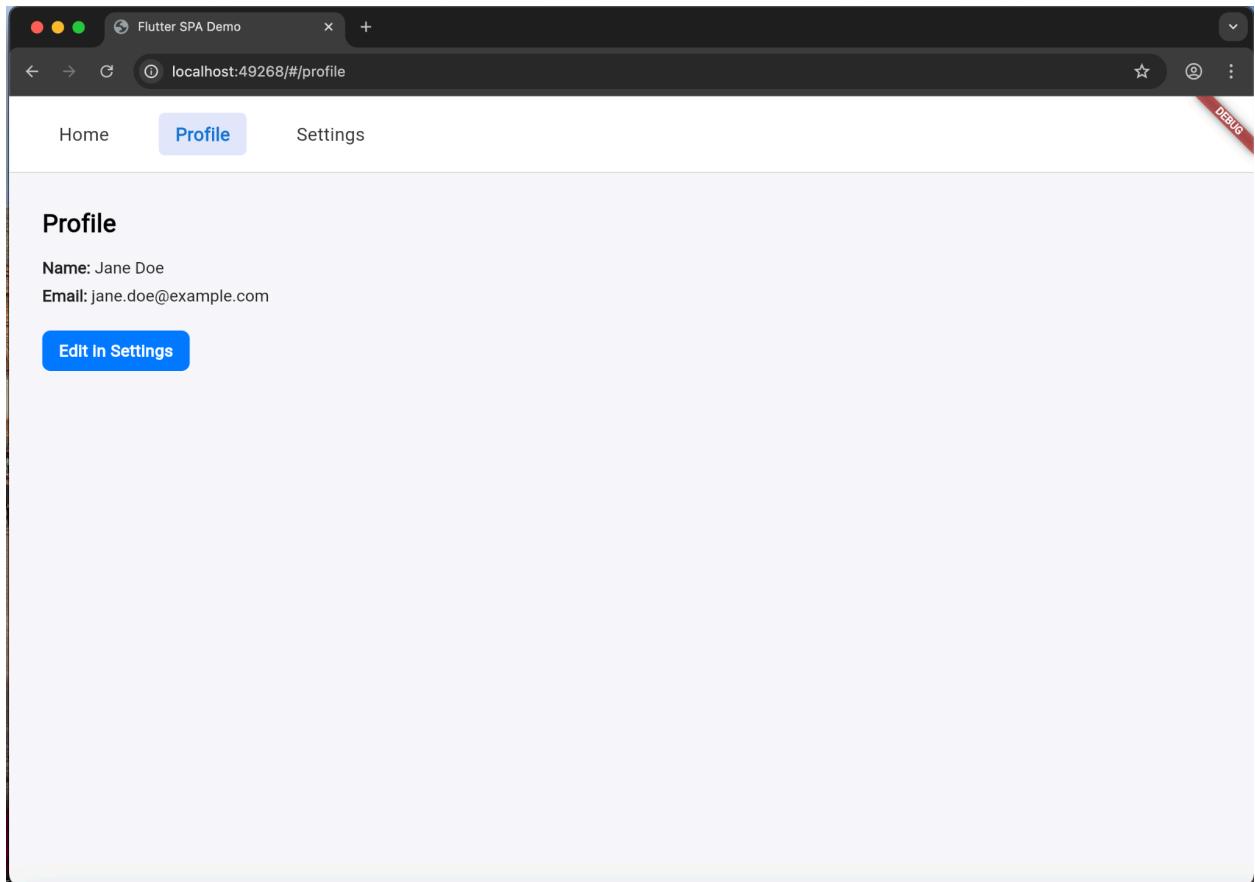


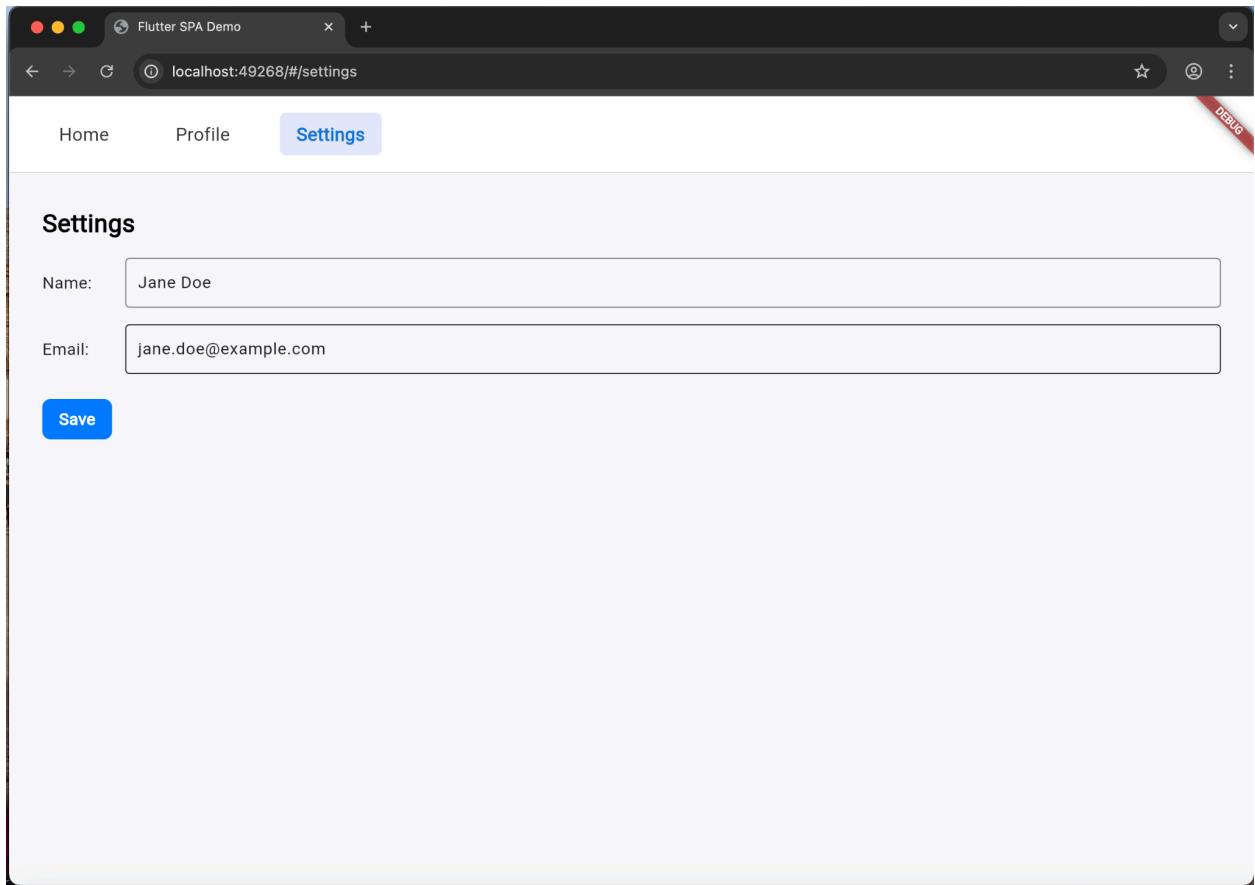




Claude







Caso 4

Prompt

Quiero que traduzcas este pequeño programa de Java (consola) a Python. El

programa permite convertir temperaturas entre Celsius y Fahrenheit. Debe funcionar de la siguiente forma:

- Solicita al usuario que ingrese un valor numérico (temperatura).
- Luego solicita que indique el tipo de conversión: de Celsius a Fahrenheit o de Fahrenheit a Celsius.
- Aplica la fórmula correspondiente:
 - $^{\circ}\text{F} = (^{\circ}\text{C} \times 9/5) + 32$
 - $^{\circ}\text{C} = (^{\circ}\text{F} - 32) \times 5/9$
- Muestra el resultado con dos decimales en la consola.
- Debe manejar entradas inválidas (por ejemplo, letras en lugar de números o selecciones incorrectas).

Te paso el código Java a continuación:

```
import java.util.Scanner;
import java.util.InputMismatchException;
/**
 * Temperature Converter Console Program
 * Converts between Celsius and Fahrenheit with error handling
 */
public class TemperatureConverter {
    private static final Scanner scanner = new Scanner(System.in);

    public static void main(String[] args) {
        System.out.println("== Temperature Converter ==");
        System.out.println("Welcome to the Temperature Converter!");
        System.out.println();

        boolean continueProgram = true;

        while (continueProgram) {
            try {
                // Get temperature input
                double temperature = getTemperatureInput();

                // Get conversion type
                int conversionType = getConversionType();

                // Perform conversion and display result
                performConversion(temperature, conversionType);

                // Ask if user wants to continue
                continueProgram = askToContinue();
            } catch (Exception e) {
                System.err.println("An unexpected error occurred: " + e.getMessage());
                System.out.println("Please try again.");
            }
        }
    }

    private static double getTemperatureInput() {
        System.out.print("Enter temperature: ");
        String input = scanner.nextLine();
        double temperature;
        try {
            temperature = Double.parseDouble(input);
        } catch (NumberFormatException e) {
            System.out.println("Invalid input. Please enter a valid number.");
            return getTemperatureInput();
        }
        return temperature;
    }

    private static int getConversionType() {
        System.out.print("Select conversion type (1 for Celsius to Fahrenheit, 2 for Fahrenheit to Celsius): ");
        String input = scanner.nextLine();
        int conversionType;
        try {
            conversionType = Integer.parseInt(input);
            if (conversionType != 1 & conversionType != 2) {
                System.out.println("Invalid selection. Please enter 1 or 2.");
                return getConversionType();
            }
        } catch (NumberFormatException e) {
            System.out.println("Invalid input. Please enter a valid number.");
            return getConversionType();
        }
        return conversionType;
    }

    private static void performConversion(double temperature, int conversionType) {
        double result;
        if (conversionType == 1) {
            result = (temperature * 9/5) + 32;
        } else {
            result = (temperature - 32) * 5/9;
        }
        System.out.printf("Result: %.2f\n", result);
    }

    private static boolean askToContinue() {
        System.out.print("Do you want to continue? (y/n): ");
        String input = scanner.nextLine();
        return input.equalsIgnoreCase("y");
    }
}
```

```

        System.out.println(); // Add spacing between iterations
    }

    System.out.println("Thank you for using the Temperature Converter!");
    scanner.close();
}

/**
 * Gets temperature input from user with validation
 * @return valid temperature value
 */
private static double getTemperatureInput() {
    while (true) {
        try {
            System.out.print("Enter the temperature value: ");
            double temperature = scanner.nextDouble();
            scanner.nextLine(); // Consume newline
            return temperature;
        } catch (InputMismatchException e) {
            System.err.println("Error: Please enter a valid numeric value.");
            scanner.nextLine(); // Clear invalid input
        }
    }
}

/**
 * Gets conversion type from user with validation
 * @return 1 for Celsius to Fahrenheit, 2 for Fahrenheit to Celsius
 */
private static int getConversionType() {
    while (true) {
        try {
            System.out.println("\nSelect conversion type:");
            System.out.println("1. Celsius to Fahrenheit");
            System.out.println("2. Fahrenheit to Celsius");
            System.out.print("Enter your choice (1 or 2): ");

            int choice = scanner.nextInt();
            scanner.nextLine(); // Consume newline

            if (choice == 1 || choice == 2) {
                return choice;
            } else {
                System.err.println("Error: Please enter 1 or 2 only.");
            }
        }
    }
}

```

```

        } catch (InputMismatchException e) {
            System.err.println("Error: Please enter a valid number (1 or 2).");
            scanner.nextLine(); // Clear invalid input
        }
    }
}

/**
 * Performs the temperature conversion and displays the result
 * @param temperature the input temperature
 * @param conversionType 1 for C to F, 2 for F to C
 */
private static void performConversion(double temperature, int conversionType) {
    double result;
    String fromUnit, toUnit;

    if (conversionType == 1) {
        // Celsius to Fahrenheit: °F = (°C × 9/5) + 32
        result = (temperature * 9.0 / 5.0) + 32;
        fromUnit = "°C";
        toUnit = "°F";
    } else {
        // Fahrenheit to Celsius: °C = (°F – 32) × 5/9
        result = (temperature - 32) * 5.0 / 9.0;
        fromUnit = "°F";
        toUnit = "°C";
    }

    System.out.println("\n--- Conversion Result ---");
    System.out.printf("%.2f%s = %.2f%s\n", temperature, fromUnit, result, toUnit);

    // Additional information for extreme temperatures
    if (conversionType == 1 && temperature <= -273.15) {
        System.out.println("Note: This temperature is at or below absolute zero!");
    } else if (conversionType == 2 && temperature <= -459.67) {
        System.out.println("Note: This temperature is at or below absolute zero!");
    }
}

/**
 * Asks user if they want to perform another conversion
 * @return true to continue, false to exit
 */
private static boolean askToContinue() {
    while (true) {
        System.out.print("\nWould you like to perform another conversion? (y/n): ");

```

```
String response = scanner.nextLine().trim().toLowerCase();

if (response.equals("y") || response.equals("yes")) {
    return true;
} else if (response.equals("n") || response.equals("no")) {
    return false;
} else {
    System.err.println("Error: Please enter 'y' for yes or 'n' for no.");
}

}
```

ChatGPT

```
# Temperature Converter Console Program in Python

def get_temperature_input():
    while True:
        try:
            value = float(input("Enter the temperature value: "))
            return value
        except ValueError:
            print("Error: Please enter a valid numeric value.")

def get_conversion_type():
    while True:
        try:
            print("\nSelect conversion type:")
            print("1. Celsius to Fahrenheit")
            print("2. Fahrenheit to Celsius")
            choice = int(input("Enter your choice (1 or 2): "))
            if choice in [1, 2]:
                return choice
            else:
                print("Error: Please enter 1 or 2 only.")
        except ValueError:
            print("Error: Please enter a valid number (1 or 2).")

def perform_conversion(temperature, conversion_type):
    if conversion_type == 1:
        result = (temperature * 9 / 5) + 32
    else:
        result = (temperature - 32) * 5 / 9
    return result
```

```

        from_unit, to_unit = "°C", "°F"
    else:
        result = (temperature - 32) * 5 / 9
        from_unit, to_unit = "°F", "°C"

    print("\n--- Conversion Result ---")
    print(f"{temperature:.2f}{from_unit} = {result:.2f}{to_unit}")

    if conversion_type == 1 and temperature <= -273.15:
        print("Note: This temperature is at or below absolute zero!")
    elif conversion_type == 2 and temperature <= -459.67:
        print("Note: This temperature is at or below absolute zero!")

def ask_to_continue():
    while True:
        response = input("\nWould you like to perform another conversion? (y/n)"):
    ".strip().lower()
        if response in ["y", "yes"]:
            return True
        elif response in ["n", "no"]:
            return False
        else:
            print("Error: Please enter 'y' for yes or 'n' for no.")

def main():
    print("== Temperature Converter ==")
    print("Welcome to the Temperature Converter!\n")

    while True:
        try:
            temp = get_temperature_input()
            conversion = get_conversion_type()
            perform_conversion(temp, conversion)
            if not ask_to_continue():
                break
        except Exception as e:
            print(f"An unexpected error occurred: {e}")
            print("Please try again.")
        print()

    print("Thank you for using the Temperature Converter!")

```

```
if __name__ == "__main__":
    main()
```

```
● (.venv) juanifraga@192 obligatorio_mdi_backend % cd "/Users/juanifraga/Desktop/Metodoliga de la investigació
n/obligatorio_mdi_backend/python-temperature-converter" && "/Users/juanifraga/Desktop/Metodoliga de la inves
tigación/obligatorio_mdi_backend/.venv/bin/python" converter.py
==> Temperature Converter ==
Welcome to the Temperature Converter!

Enter the temperature value: 5

Select conversion type:
1. Celsius to Fahrenheit
2. Fahrenheit to Celsius
Enter your choice (1 or 2): 1

--- Conversion Result ---
5.00°C = 41.00°F

Would you like to perform another conversion? (y/n): n
Thank you for using the Temperature Converter!
○ (.venv) juanifraga@192 python-temperature-converter % []
```

Codex

```
"""Temperature Converter Console Program

Converts between Celsius and Fahrenheit with error handling.

"""

def get_temperature_input():
    while True:
        try:
            value = float(input("Enter the temperature value: "))
            return value
        except ValueError:
            print("Error: Please enter a valid numeric value.")

def get_conversion_type():
    while True:
        print("\nSelect conversion type:")
        print("1. Celsius to Fahrenheit")
        print("2. Fahrenheit to Celsius")
        choice = input("Enter your choice (1 or 2): ")
        if choice in ("1", "2"):
            return int(choice)
        else:
            print("Error: Please enter 1 or 2 only.")
```

```

def perform_conversion(temperature, conversion_type):
    if conversion_type == 1:
        result = (temperature * 9.0 / 5.0) + 32
        from_unit = "°C"
        to_unit = "°F"
    else:
        result = (temperature - 32) * 5.0 / 9.0
        from_unit = "°F"
        to_unit = "°C"

    print("\n--- Conversion Result ---")
    print(f"{temperature:.2f}{from_unit} = {result:.2f}{to_unit}")

    if conversion_type == 1 and temperature <= -273.15:
        print("Note: This temperature is at or below absolute zero!")
    elif conversion_type == 2 and temperature <= -459.67:
        print("Note: This temperature is at or below absolute zero!")

def ask_to_continue():
    while True:
        response = input("\nWould you like to perform another conversion? (y/n) :").strip().lower()
        if response in ("y", "yes"):
            return True
        elif response in ("n", "no"):
            return False
        else:
            print("Error: Please enter 'y' for yes or 'n' for no.")

def main():
    print("==== Temperature Converter ====")
    print("Welcome to the Temperature Converter!\n")

    continue_program = True
    while continue_program:
        try:
            temperature = get_temperature_input()
            conversion_type = get_conversion_type()
            perform_conversion(temperature, conversion_type)
        except ValueError:
            print("Error: Invalid input. Please enter a valid temperature value.")



```

```

        continue_program = ask_to_continue()

    except Exception as e:
        print(f"An unexpected error occurred: {e}")
        print("Please try again.")

    print()

print("Thank you for using the Temperature Converter!")

if __name__ == "__main__":
    main()

```

```

● (.venv) juanifraga@192 obligatorio_mdi_backend % cd "/Users/juanifraga/Desktop/Metodoliga de la investigación/obligatorio_mdi_backend/python-temperature-converter" && "/Users/juanifraga/Desktop/Metodoliga de la investigación/obligatorio_mdi_backend/.venv/bin/python" temperature_converter.py
== Temperature Converter ==
Welcome to the Temperature Converter!

Enter the temperature value: 77

Select conversion type:
1. Celsius to Fahrenheit
2. Fahrenheit to Celsius
Enter your choice (1 or 2): 2

--- Conversion Result ---
77.00°F = 25.00°C

Would you like to perform another conversion? (y/n): n

Thank you for using the Temperature Converter!
○ (.venv) juanifraga@192 python-temperature-converter %

```

Claude

```

"""
Temperature Converter Console Program
Converts between Celsius and Fahrenheit with error handling
"""

def get_temperature_input():
    """
    Gets temperature input from user with validation
    Returns: valid temperature value
    """

    while True:

```

```

try:
    temperature = float(input("Enter the temperature value: "))
    return temperature
except ValueError:
    print("Error: Please enter a valid numeric value.")

def get_conversion_type():
    """
    Gets conversion type from user with validation
    Returns: 1 for Celsius to Fahrenheit, 2 for Fahrenheit to Celsius
    """
    while True:
        try:
            print("\nSelect conversion type:")
            print("1. Celsius to Fahrenheit")
            print("2. Fahrenheit to Celsius")

            choice = int(input("Enter your choice (1 or 2): "))

            if choice == 1 or choice == 2:
                return choice
            else:
                print("Error: Please enter 1 or 2 only.")
        except ValueError:
            print("Error: Please enter a valid number (1 or 2).")

def perform_conversion(temperature, conversion_type):
    """
    Performs the temperature conversion and displays the result
    Args:
        temperature: the input temperature
        conversion_type: 1 for C to F, 2 for F to C
    """
    if conversion_type == 1:
        # Celsius to Fahrenheit: °F = (°C × 9/5) + 32
        result = (temperature * 9.0 / 5.0) + 32
        from_unit = "°C"
        to_unit = "°F"
    else:
        # Fahrenheit to Celsius: °C = (°F - 32) × 5/9
        result = (temperature - 32) * 5.0 / 9.0
        from_unit = "°F"

```

```

        to_unit = "°C"

    print("\n--- Conversion Result ---")
    print(f"{temperature:.2f}{from_unit} = {result:.2f}{to_unit}")

    # Additional information for extreme temperatures
    if conversion_type == 1 and temperature <= -273.15:
        print("Note: This temperature is at or below absolute zero!")
    elif conversion_type == 2 and temperature <= -459.67:
        print("Note: This temperature is at or below absolute zero!")

def ask_to_continue():
    """
    Asks user if they want to perform another conversion
    Returns: True to continue, False to exit
    """
    while True:
        response = input("\nWould you like to perform another conversion? (y/n):").strip().lower()

        if response in ["y", "yes"]:
            return True
        elif response in ["n", "no"]:
            return False
        else:
            print("Error: Please enter 'y' for yes or 'n' for no.")

def main():
    """Main program function"""
    print("==== Temperature Converter ====")
    print("Welcome to the Temperature Converter!")
    print()

    continue_program = True

    while continue_program:
        try:
            # Get temperature input
            temperature = get_temperature_input()

            # Get conversion type
            conversion_type = get_conversion_type()

```

```

# Perform conversion and display result
perform_conversion(temperature, conversion_type)

# Ask if user wants to continue
continue_program = ask_to_continue()

except Exception as e:
    print(f"An unexpected error occurred: {e}")
    print("Please try again.")

print() # Add spacing between iterations

print("Thank you for using the Temperature Converter!")

if __name__ == "__main__":
    main()

```

```

○ (.venv) juanifraga@192 python-temperature-converter % cd "/Users/juanifraga/Desktop/Metodología de la investigación/obligatorio_mdi_backend/python-temperature-converter" && "/Users/juanifraga/Desktop/Metodología de la investigación/obligatorio_mdi_backend/.venv/bin/python" temperature_converter.py
==> Temperature Converter ==
Welcome to the Temperature Converter!

Enter the temperature value: 0

Select conversion type:
1. Celsius to Fahrenheit
2. Fahrenheit to Celsius
Enter your choice (1 or 2): 1

--- Conversion Result ---
0.00°C = 32.00°F

Would you like to perform another conversion? (y/n): 

```

DeepSeek

```

def main():

    print("==> Temperature Converter ==")
    print("Welcome to the Temperature Converter!")
    print()

    continue_program = True

    while continue_program:
        try:

```

```

# Get temperature input
temperature = get_temperature_input()

# Get conversion type
conversion_type = get_conversion_type()

# Perform conversion and display result
perform_conversion(temperature, conversion_type)

# Ask if user wants to continue
continue_program = ask_to_continue()

except Exception as e:
    print(f"An unexpected error occurred: {str(e)}")
    print("Please try again.")

print() # Add spacing between iterations

print("Thank you for using the Temperature Converter!")

def get_temperature_input():
    while True:
        try:
            temperature = float(input("Enter the temperature value: "))
            return temperature
        except ValueError:
            print("Error: Please enter a valid numeric value.")

def get_conversion_type():
    while True:
        try:
            print("\nSelect conversion type:")
            print("1. Celsius to Fahrenheit")
            print("2. Fahrenheit to Celsius")
            choice = int(input("Enter your choice (1 or 2): "))

            if choice == 1 or choice == 2:
                return choice
            else:
                print("Error: Please enter 1 or 2 only.")
        except ValueError:
            print("Error: Please enter a valid number (1 or 2).")

```

```

def perform_conversion(temperature, conversion_type):
    if conversion_type == 1:
        # Celsius to Fahrenheit: °F = (°C × 9/5) + 32
        result = (temperature * 9.0 / 5.0) + 32
        from_unit = "°C"
        to_unit = "°F"
    else:
        # Fahrenheit to Celsius: °C = (°F - 32) × 5/9
        result = (temperature - 32) * 5.0 / 9.0
        from_unit = "°F"
        to_unit = "°C"

    print("\n--- Conversion Result ---")
    print(f"{temperature:.2f}{from_unit} = {result:.2f}{to_unit}")

    # Additional information for extreme temperatures
    if conversion_type == 1 and temperature <= -273.15:
        print("Note: This temperature is at or below absolute zero!")
    elif conversion_type == 2 and temperature <= -459.67:
        print("Note: This temperature is at or below absolute zero!")

def ask_to_continue():
    while True:
        response = input("\nWould you like to perform another conversion? (y/n) :").strip().lower()

        if response in ('y', 'yes'):
            return True
        elif response in ('n', 'no'):
            return False
        else:
            print("Error: Please enter 'y' for yes or 'n' for no.")

if __name__ == "__main__":
    main()

```

```
● (.venv) juanifraga@192 python-temperature-converter % cd "/Users/juanifraga/Desktop/Metodoliga de la investigación/obligatorio_mdi_backend/python-temperature-converter" && "/Users/juanifraga/Desktop/Metodoliga de la investigación/obligatorio_mdi_backend/.venv/bin/python" temperature_converter.py
*** Temperature Converter ***
Welcome to the Temperature Converter!

Enter the temperature value: 100

Select conversion type:
1. Celsius to Fahrenheit
2. Fahrenheit to Celsius
Enter your choice (1 or 2): 1

--- Conversion Result ---
100.00°C = 212.00°F

Would you like to perform another conversion? (y/n): n

Thank you for using the Temperature Converter!
○ (.venv) juanifraga@192 python-temperature-converter %
```

CASO 5

Prompt

Necesito traducir la siguiente clase Python a Java.

La clase representa a una persona con los siguientes atributos y funcionalidades:

- Atributos:
 - nombre (String)
 - edad (int)
 - email (String)
- Validaciones en el constructor:
 - El nombre no debe estar vacío.
 - La edad debe ser un número entero mayor o igual a 0
 - . - El email debe contener el símbolo '@'.
- Métodos:
 - es_mayor_de_edad: devuelve true si la edad es mayor o igual a 18
 - . - mostrar_info: imprime los datos de la persona de forma legible.

Código original en Python:

```
class Persona: def __init__(self, nombre, edad, email):  
    if edad <= 0:  
        raise ValueError("La edad debe ser positiva.")  
    if "@" not in email:  
        raise ValueError("El email debe contener '@'.")  
    self.nombre = nombre  
    self.edad = edad
```

```

self.email = email

def es_mayor_de_edad(self):
    return self.edad >= 18

def mostrar_info(self):
    print(f"Nombre: {self.nombre}")
    print(f"Edad: {self.edad}")
    print(f"Email: {self.email}")

if __name__ == "__main__":
    persona = Persona("Ilán", 21, "ilan@example.com")
    persona.mostrar_info()
    print("¿Mayor de edad?", persona.es_mayor_de_edad())

```

Requerimiento adicional: La clase Java debe compilar correctamente junto a la siguiente clase Main.java:

```

public class Main {
    public static void main(String[] args) {
        Persona p = new Persona("Ilán", 21, "ilan@example.com");
        p.mostrarInfo();
        System.out.println("¿Mayor de edad? " + p.esMayorDeEdad());
    }
}

```

ChatGPT

```
backend-java > J persona.java
 1  public class Persona {
 2      private String nombre;
 3      private int edad;
 4      private String email;
 5
 6      public Persona(String nombre, int edad, String email) {
 7          if (nombre == null || nombre.trim().isEmpty()) {
 8              throw new IllegalArgumentException("El nombre no debe estar vacío.");
 9          }
10          if (edad < 0) {
11              throw new IllegalArgumentException("La edad debe ser mayor o igual a 0.");
12          }
13          if (email == null || !email.contains("@")) {
14              throw new IllegalArgumentException("El email debe contener '@'.");
15          }
16
17          this.nombre = nombre;
18          this.edad = edad;
19          this.email = email;
20      }
21
22      public boolean esMayorDeEdad() {
23          return edad >= 18;
24      }
25
26      public void mostrarInfo() {
27          System.out.println("Nombre: " + nombre);
28          System.out.println("Edad: " + edad);
29          System.out.println("Email: " + email);
30      }
31  }
32 }
```

```
❖PS C:\Users\ilanp\OneDrive\Documents\Obligatorio_MDI_Backend\obligatorio_mdi_backend\backend-java> java Main
>>
Nombre: Ilan
Edad: 21
Email: ilan@example.com
?¿Mayor de edad? true
```

DeepSeek

```
PS C:\Users\ilanp\OneDrive\Documents\Obligatorio_MDI_Backend\obligatorio_mdi_backend\backend-java> java Main
>>
Nombre: Ilan
Edad: 21
Email: ilan@example.com
?¿Mayor de edad? true
```

```
backend-java > J personajava
 1  public class Persona {
 2      private String nombre;
 3      private int edad;
 4      private String email;
 5
 6      public Persona(String nombre, int edad, String email) {
 7          if (nombre == null || nombre.trim().isEmpty()) {
 8              throw new IllegalArgumentException("El nombre no debe estar vacío.");
 9          }
10          if (edad < 0) {
11              throw new IllegalArgumentException("La edad debe ser positiva.");
12          }
13          if (email == null || !email.contains("@")) {
14              throw new IllegalArgumentException("El email debe contener '@' .");
15          }
16
17          this.nombre = nombre;
18          this.edad = edad;
19          this.email = email;
20      }
21
22      public boolean esMayorDeEdad() {
23          return this.edad >= 18;
24      }
25
26      public void mostrarInfo() {
27          System.out.println("Nombre: " + this.nombre);
28          System.out.println("Edad: " + this.edad);
29          System.out.println("Email: " + this.email);
30      }
31
32      // Getters (opcionales, según necesidades)
33      public String getNombre() {
34          return nombre;
35      }
36
37      public int getEdad() {
```

```
37          public int getEdad() {
38              return edad;
39          }
40
41          public String getEmail() {
42              return email;
43          }
44      }
```

CLAUDE

```
❖ PS C:\Users\ilanp\OneDrive\Documents\Obligatorio_MDI_Backend\obligatorio_mdi_backend\backend-java> java Main
>>
Nombre: Ilan
Edad: 21
Email: ilan@example.com
?¿Mayor de edad? true
```

```
backend-java > J Persona.java
 1  public class Persona {
 2      private String nombre;
 3      private int edad;
 4      private String email;
 5
 6      public Persona(String nombre, int edad, String email) {
 7          if (nombre == null || nombre.isEmpty()) {
 8              throw new IllegalArgumentException("El nombre no debe estar vacío.");
 9          }
10          if (edad < 0) {
11              throw new IllegalArgumentException("La edad debe ser mayor o igual a 0.");
12          }
13          if (email == null || !email.contains("@")) {
14              throw new IllegalArgumentException("El email debe contener '@' .");
15          }
16          this.nombre = nombre;
17          this.edad = edad;
18          this.email = email;
19      }
20
21      public boolean esMayorDeEdad() {
22          return this.edad >= 18;
23      }
24
25      public void mostrarInfo() {
26          System.out.println("Nombre: " + this.nombre);
27          System.out.println("Edad: " + this.edad);
28          System.out.println("Email: " + this.email);
29      }
30  } |
```

CODEX

```
PS C:\Users\ilanp\OneDrive\Documents\obligatorio_MDI_Backend\obligatorio_mdi_backend\backend-java> java Main
>>
Nombre: Ilan
Edad: 21
Email: ilan@example.com
?¿Mayor de edad? true
```

```
backend-java > J personajava
 1  public class Persona {
 2      private String nombre;
 3      private int edad;
 4      private String email;
 5
 6      public Persona(String nombre, int edad, String email) {
 7          if (nombre == null || nombre.isEmpty()) {
 8              throw new IllegalArgumentException("El nombre no debe estar vac\u00edo.");
 9          }
10          if (edad < 0) {
11              throw new IllegalArgumentException("La edad debe ser un n\u00famero entero mayor o igual a 0.");
12          }
13          if (email == null || !email.contains("@")) {
14              throw new IllegalArgumentException("El email debe contener '@'.");
15          }
16          this.nombre = nombre;
17          this.edad = edad;
18          this.email = email;
19      }
20
21      public boolean esMayorDeEdad() {
22          return this.edad >= 18;
23      }
24
25      public void mostrarInfo() {
26          System.out.println("Nombre: " + this.nombre);
27          System.out.println("Edad: " + this.edad);
28          System.out.println("Email: " + this.email);
29      }
30  }
```

| Modelo | Validación nombre vacío | Validación edad negativa | Validación email | Getters incluidos | Mensajes de error claros | Cumple ejecución Main | Claridad del código |
|----------|-----------------------------------------|---------------------------------------|--------------------------------------|-------------------------------------|----------------------------------------------|-----------------------|------------------------------------------------|
| ChatGPT | <code>✓ null + .trim().isEmpty()</code> | <code>⚠ Solo < 0, permite 0</code> | <code>✓ null + !contains("@")</code> | <code>✗ No</code> | <code>✓ Descriptivos, bien redactados</code> | <code>✓ Sí</code> | <code>✓ Ordenado y preciso</code> |
| DeepSeek | <code>✓ null + .trim().isEmpty()</code> | <code>⚠ Solo < 0, permite 0</code> | <code>✓ null + !contains("@")</code> | <code>✓ Sí (getNombre, etc.)</code> | <code>✓ Claros, pero menos detallados</code> | <code>✓ Sí</code> | <code>✓ Limpio y extensible</code> |
| Claude | <code>✓ null + .isEmpty()</code> | <code>⚠ Solo < 0, permite 0</code> | <code>✓ null + !contains("@")</code> | <code>✗ No</code> | <code>✓ Claros, sin redundancias</code> | <code>✓ Sí</code> | <code>✓ Legible y directo</code> |
| Codex | <code>✓ null + .isEmpty()</code> | <code>⚠ Solo < 0, permite 0</code> | <code>✓ null + !contains("@")</code> | <code>✗ No</code> | <code>⚠ Codificación UTF mal mostrada</code> | <code>✓ Sí</code> | <code>⚠ Código correcto pero descuidado</code> |

Todos los modelos (ChatGPT, DeepSeek, Codex y Claude) lograron generar correctamente una clase Persona funcional en Java, incluyendo atributos, constructor con validaciones, y los métodos esMayorDeEdad y mostrarInfo. A nivel de salida por consola, los cuatro modelos imprimen correctamente los datos esperados y la verificación de mayoría de edad. Sin embargo, hay diferencias relevantes en cuanto a estilo, calidad y agregados extra.

ChatGPT:

- Fue el más preciso y directo.
- Implementó correctamente todas las validaciones y métodos solicitados.
- No agregó nada innecesario ni excedente.
- Su código es limpio, entendible y compiló sin errores.

DeepSeek:

- Generó el mismo comportamiento correcto que ChatGPT, pero además añadió métodos getter (getEdad() y getEmail()).
- Estos métodos no eran obligatorios pero pueden ser útiles si se requiere encapsulamiento.
- El resto del código mantiene una estructura clara, sin errores y con excelente validación.

Claude:

- También resolvió correctamente el problema con código limpio y bien indentado.
- Su sintaxis y orden son correctos, aunque no añadió getters ni comentarios, siendo más minimalista.
- La salida fue idéntica a la esperada y no mostró errores.

Codex:

- Logró una solución funcional con el mismo comportamiento que los demás modelos.
- Sin embargo, se observan caracteres extraños en los mensajes de error (probablemente por codificación Unicode mal interpretada), lo cual afecta la claridad del mensaje al usuario final.
- Aparte de eso, cumplió correctamente con la lógica requerida

Para finalizar todos los modelos resolvieron el ejercicio correctamente desde el punto de vista funcional, generando una clase Persona válida en Java. Sin embargo:

ChatGPT y Claude ofrecieron versiones limpias y fieles al prompt.

DeepSeek fue el más completo al incluir métodos adicionales útiles.

Codex funcionó bien, pero su salida quedó afectada por problemas de codificación en los mensajes de error.

En resumen, para un desarrollo profesional o académico donde la limpieza del código y la compatibilidad sean prioritarias, ChatGPT y Claude ofrecen excelentes resultados. Si se busca mayor robustez con encapsulamiento, DeepSeek suma puntos. Por otro lado, Codex cumple, pero con detalles que deben corregirse manualmente.

| Modelo | Modularidad | Legibilidad | Uso del framework | Total |
|---------------|--------------------|--------------------|--------------------------|--------------|
| Claude | 5 | 5 | 5 | 15 |
| ChatGPT | 5 | 5 | 4 | 14 |
| DeepSeek | 5 | 4 | 5 | 14 |
| Codex | 4 | 3 | 4 | 11 |

Caso 1:

| Modelo | Modularidad | Legibilidad | Uso del framework | Total
(15) |
|---------------|--------------------|--------------------|--------------------------|-----------------------|
| Claude | 5 | 5 | 4 | 14 |

| | | | | |
|-----------------|---|---|---|-----------|
| ChatGPT | 4 | 5 | 5 | 14 |
| DeepSeek | 4 | 4 | 4 | 12 |
| Codex | 4 | 4 | 4 | 12 |

Observaciones principales

Uso de StatelessWidget en Claude, excelente organización. Hover con WidgetStateProperty moderno pero menos común.

Uso explícito de MouseRegion y GestureDetector en chat gpt. Hover implementado de forma precisa, aunque más manual.

Correcto uso de ElevatedButton en deepseek, aunque uso mixto de WidgetStateProperty y estilos algo redundantes.

Implementación funcional, hover correcto en codex, pero estructura algo menos limpia que ChatGPT.

Caso 6:

Evaluación estructural (Rúbrica):

| Modelo | Modularidad | Legibilidad | Uso del framework | Total (15) |
|---------------|--------------------|--------------------|--------------------------|-------------------|
| Claude | 5 | 5 | 5 | 15 |
| ChatGPT | 4 | 4 | 4 | 12 |
| DeepSeek | 5 | 4 | 4 | 13 |
| Codex | 4 | 3 | 4 | 11 |

Observaciones

Separación clara de capas. Uso correcto de anotaciones y validaciones.

Validaciones inconsistentes. Organización buena.

Estructura modular clara. Algunas clases acopladas.

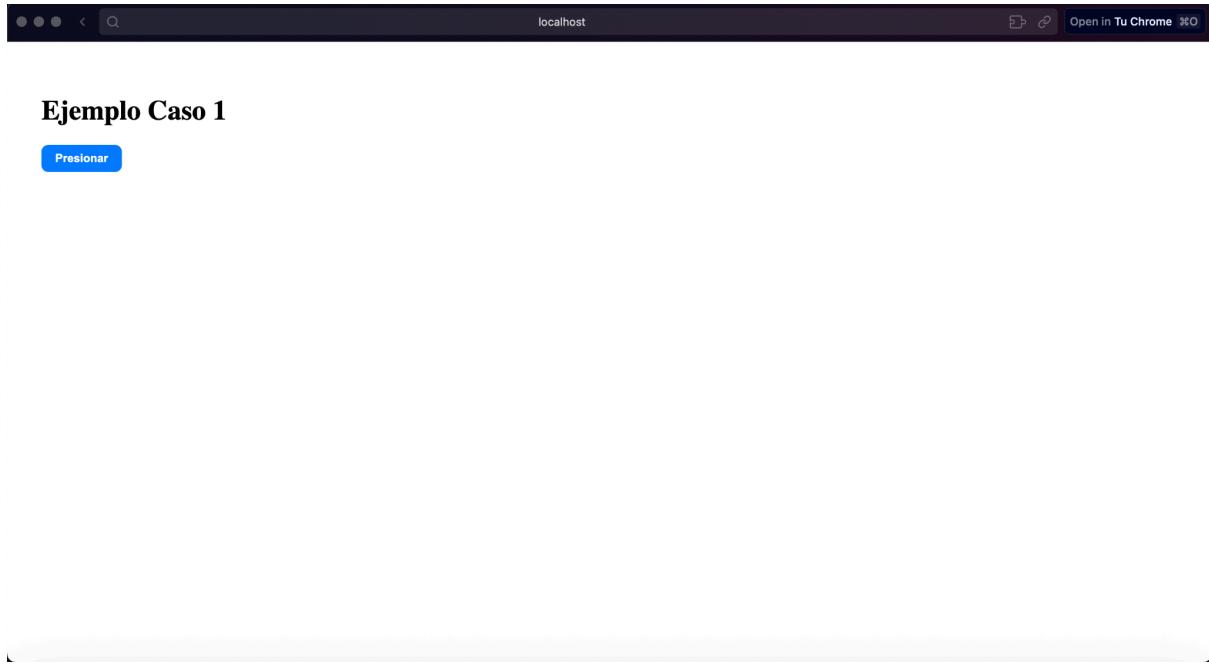
Código funcional pero débil en validaciones y claridad.

Errores frecuentes:

- ChatGPT: no valida correctamente `stock >= 0`.
- Codex: permite precios negativos y omite validaciones.
- Algunos modelos no gestionan errores 404 adecuadamente.

caso 1

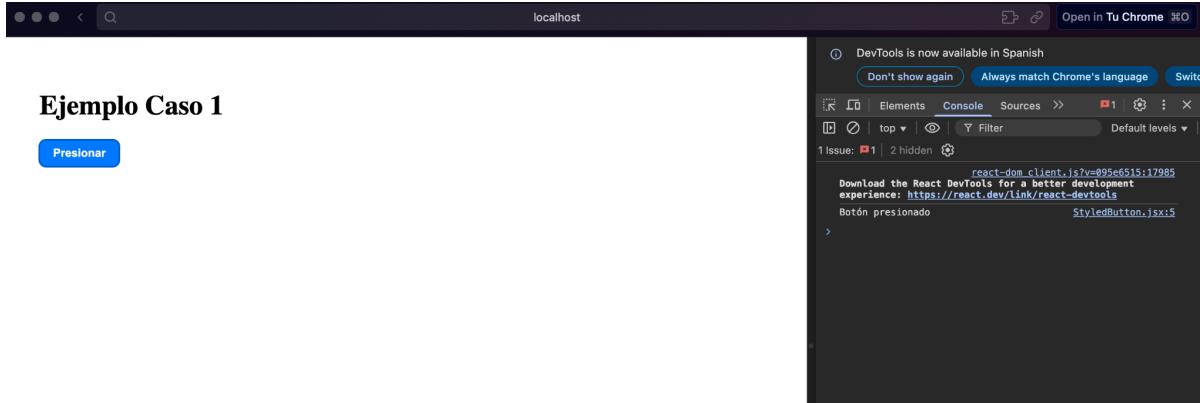
caso 1 base



Ejemplo Caso 1

[Presionar](#)

caso 1 base hover



caso 6:

caso 6 base

```
marceloisrael@MacBook-Pro-de-Marcelo obligatorio_mdi_backend % uvicorn main:app --reload
INFO:     Will watch for changes in these directories: ['/Users/marceloisrael/Documents/obligatorio_mdi_backend']
INFO:     Uvicorn running on http://127.0.0.1:8000 (Press CTRL+C to quit)
INFO:     Started reloader process [42910] using StatReload
INFO:     Started server process [42943]
INFO:     Waiting for application startup.
INFO:     Application startup complete.
INFO: 127.0.0.1:57727 - "POST /productos HTTP/1.1" 200 OK
INFO: 127.0.0.1:57731 - "GET /productos HTTP/1.1" 200 OK
INFO: 127.0.0.1:57733 - "GET /productos/1 HTTP/1.1" 200 OK
INFO: 127.0.0.1:57737 - "PUT /productos/1 HTTP/1.1" 200 OK
INFO: 127.0.0.1:57739 - "DELETE /productos/1 HTTP/1.1" 200 OK
```

```
marceloisrael@MacBook-Pro-de-Marcelo obligatorio_mdi_backend % curl -X POST "http://localhost:8000/productos" \
-H "Content-Type: application/json" \
-d '{"nombre": "Laptop", "precio": 999.99, "stock": 10}'
{"nombre":"Laptop","precio":999.99,"stock":10,"id":1}
marceloisrael@MacBook-Pro-de-Marcelo obligatorio_mdi_backend % curl -X GET "http://localhost:8000/productos"
[{"nombre": "Laptop", "precio": 999.99, "stock": 10, "id": 1}]
marceloisrael@MacBook-Pro-de-Marcelo obligatorio_mdi_backend % curl -X GET "http://localhost:8000/productos/1"
{"nombre": "Laptop", "precio": 999.99, "stock": 10, "id": 1}
marceloisrael@MacBook-Pro-de-Marcelo obligatorio_mdi_backend % curl -X PUT "http://localhost:8000/productos/1" \
-H "Content-Type: application/json" \
-d '{"nombre": "Laptop Gaming", "precio": 1299.99, "stock": 8}'
{"nombre":"Laptop Gaming","precio":1299.99,"stock":8,"id":1}
marceloisrael@MacBook-Pro-de-Marcelo obligatorio_mdi_backend % curl -X DELETE "http://localhost:8000/productos/1"
{"ok":true}
marceloisrael@MacBook-Pro-de-Marcelo obligatorio_mdi_backend %
```

¿Querés que integre esta evaluación estructural directamente en el documento del anteproyecto o la dejamos como anexo? También puedo generar la hoja de cálculo base para registrar estos resultados.

caso 6 chatgpt

```
Problems 23 Output Debug Console Terminal Ports Spell Checker 23 Git GitLens Comments +  
marceloisrael@MacBook-Pro-de-Marcelo obligatorio_mdi_backend % curl -X POST http://localhost:8080/productos  
\  
-H "Content-Type: application/json" \  
-d '{"nombre": "Laptop Dell", "precio": 1200.50, "stock": 10}'  
{"id":1,"nombre":"Laptop Dell","precio":1200.5,"stock":10}  
marceloisrael@MacBook-Pro-de-Marcelo obligatorio_mdi_backend % curl -X GET http://localhost:8080/productos  
[{"id":1,"nombre": "Laptop Dell", "precio": 1200.5, "stock": 10}]  
marceloisrael@MacBook-Pro-de-Marcelo obligatorio_mdi_backend % curl -X GET http://localhost:8080/productos/1  
  
{"id":1,"nombre": "Laptop Dell", "precio": 1200.5, "stock": 10}  
marceloisrael@MacBook-Pro-de-Marcelo obligatorio_mdi_backend % curl -X PUT http://localhost:8080/productos/1  
\  
-H "Content-Type: application/json" \  
-d '{"nombre": "Laptop Dell XPS", "precio": 1350.00, "stock": 8}'  
{"id":1,"nombre":"Laptop Dell XPS","precio":1350.0,"stock":8}  
marceloisrael@MacBook-Pro-de-Marcelo obligatorio_mdi_backend % curl -X DELETE http://localhost:8080/productos/1  
{"ok": true}  
marceloisrael@MacBook-Pro-de-Marcelo obligatorio_mdi_backend %
```

caso 6 deepseek

```
marceloisrael@MacBook-Pro-de-Marcelo obligatorio_mdi_backend % curl -X POST http://localhost:8080/productos  
\  
-H "Content-Type: application/json" \  
-d '{"nombre": "Laptop Dell", "precio": 1200.50, "stock": 10}'  
{"id":1,"nombre":"Laptop Dell","precio":1200.5,"stock":10}  
marceloisrael@MacBook-Pro-de-Marcelo obligatorio_mdi_backend % curl -X GET http://localhost:8080/productos  
[{"id":1,"nombre": "Laptop Dell", "precio": 1200.5, "stock": 10}]  
marceloisrael@MacBook-Pro-de-Marcelo obligatorio_mdi_backend % curl -X GET http://localhost:8080/productos/1  
  
{"id":1,"nombre": "Laptop Dell", "precio": 1200.5, "stock": 10}  
marceloisrael@MacBook-Pro-de-Marcelo obligatorio_mdi_backend % curl -X PUT http://localhost:8080/productos/1  
\  
-H "Content-Type: application/json" \  
-d '{"nombre": "Laptop Dell XPS", "precio": 1350.00, "stock": 8}'  
{"id":1,"nombre":"Laptop Dell XPS","precio":1350.0,"stock":8}  
marceloisrael@MacBook-Pro-de-Marcelo obligatorio_mdi_backend % curl -X DELETE http://localhost:8080/productos/1  
{"ok": true}  
marceloisrael@MacBook-Pro-de-Marcelo obligatorio_mdi_backend % curl -X POST http://localhost:8080/productos  
\  
-H "Content-Type: application/json" \  
-d '{"nombre": "Laptop Dell", "precio": 1200.50, "stock": 10}'  
{"id":1,"nombre":"Laptop Dell","precio":1200.5,"stock":10}  
marceloisrael@MacBook-Pro-de-Marcelo obligatorio_mdi_backend % curl -X GET http://localhost:8080/productos  
[{"id":1,"nombre": "Laptop Dell", "precio": 1200.5, "stock": 10}]  
marceloisrael@MacBook-Pro-de-Marcelo obligatorio_mdi_backend % curl -X GET http://localhost:8080/productos/1  
  
{"id":1,"nombre": "Laptop Dell", "precio": 1200.5, "stock": 10}  
marceloisrael@MacBook-Pro-de-Marcelo obligatorio_mdi_backend % # Actualizar el producto con ID 1  
curl -X PUT http://localhost:8080/productos/1 \  
-H "Content-Type: application/json" \  
-d '{"nombre": "Laptop Dell XPS", "precio": 1350.00, "stock": 8}'  
zsh: command not found: #  
{"id":1,"nombre":"Laptop Dell XPS","precio":1350.0,"stock":8}  
marceloisrael@MacBook-Pro-de-Marcelo obligatorio_mdi_backend % # Eliminar producto con ID 2  
curl -X DELETE http://localhost:8080/productos/1  
zsh: command not found: #  
marceloisrael@MacBook-Pro-de-Marcelo obligatorio_mdi_backend %
```

caso 6 codex

```
marceloisrael@MacBook-Pro-de-Marcelo obligatorio_mdi_backend % curl -X POST http://localhost:8080/productos
\ 
-H "Content-Type: application/json" \
-d '{"nombre": "Laptop Dell", "precio": 1200.50, "stock": 10}'
{"id":1,"nombre":"Laptop Dell","precio":1200.5,"stock":10}%
marceloisrael@MacBook-Pro-de-Marcelo obligatorio_mdi_backend % curl -X GET http://localhost:8080/productos
[{"id":1,"nombre":"Laptop Dell","precio":1200.5,"stock":10}]%
marceloisrael@MacBook-Pro-de-Marcelo obligatorio_mdi_backend % curl -X GET http://localhost:8080/productos/1

{"id":1,"nombre":"Laptop Dell","precio":1200.5,"stock":10}%
marceloisrael@MacBook-Pro-de-Marcelo obligatorio_mdi_backend % curl -X PUT http://localhost:8080/productos/1
\ 
-H "Content-Type: application/json" \
-d '{"nombre": "Laptop Dell XPS", "precio": 1350.00, "stock": 8}'
{"id":1,"nombre":"Laptop Dell XPS","precio":1350.0,"stock":8}%
marceloisrael@MacBook-Pro-de-Marcelo obligatorio_mdi_backend % curl -X GET http://localhost:8080/productos/1

>{"id":1,"nombre":"Laptop Dell XPS","precio":1350.0,"stock":8}%
marceloisrael@MacBook-Pro-de-Marcelo obligatorio_mdi_backend % curl -X DELETE http://localhost:8080/productos/1
marceloisrael@MacBook-Pro-de-Marcelo obligatorio_mdi_backend % curl -X GET http://localhost:8080/productos
[]%
```

caso 6 claudie

The screenshot shows a terminal window with the following content:

```
Problems 488 Output Debug Console Terminal 15 Ports Spell Checker Git GitLens Comments +  
marceloisrael@MacBook-Pro-de-Marcelo obligatorio_mdi_backend % ./test-api.sh  
Probando la API de Productos Spring Boot...  
  
1. Creando un producto...  
Respuesta: {"id":1,"nombre":"Laptop HP","precio":999.99,"stock":10}  
  
2. Creando otro producto...  
Respuesta: {"id":2,"nombre":"Mouse Inalámbrico","precio":25.5,"stock":50}  
  
3. Listando todos los productos...  
Respuesta: [{"id":1,"nombre":"Laptop HP","precio":999.99,"stock":10}, {"id":2,"nombre":"Mouse Inalámbrico","precio":25.5,"stock":50}]  
  
4. Obteniendo producto con ID 1...  
Respuesta: {"id":1,"nombre":"Laptop HP","precio":999.99,"stock":10}  
  
5. Actualizando producto con ID 1...  
Respuesta: {"id":1,"nombre":"Laptop HP Actualizada","precio":1099.99,"stock":5}  
  
6. Probando validación (precio negativo)...  
Respuesta: {"detail":"Error de validación","errors":{"precio":"El precio debe ser positivo","stock":"El stock debe ser mayor o igual a 0","nombre":"El nombre es obligatorio y no puede estar vacío"} }  
  
7. Eliminando producto con ID 2...  
Respuesta: {"ok":true}  
  
8. Listando productos finales...  
Respuesta: [{"id":1,"nombre":"Laptop HP Actualizada","precio":1099.99,"stock":5}]  
  
✓ Pruebas completadas!  
🔗 Ver documentación Swagger en: http://localhost:8080/docs  
marceloisrael@MacBook-Pro-de-Marcelo obligatorio_mdi_backend %
```

Caso 6: Mini API modular (Backend - Nivel Alto)

Prompt estandarizado para los modelos:

Traducí esta mini API construida en Python usando FastAPI a un proyecto en Java con Spring Boot. La API gestiona la entidad Producto y tiene las siguientes características:

- Endpoints CRUD: crear producto, obtener por ID, listar todos, actualizar, eliminar.
- Validaciones:
 - nombre obligatorio (no vacío).
 - precio debe ser positivo.
 - stock debe ser entero ≥ 0 .
- Separación en capas:
 - Modelos o esquemas.
 - Servicios o lógica de negocio.
 - Controladores o rutas.
- Estructura modular: mínimo tres archivos separados.
- Debe incluir documentación automática con Swagger o OpenAPI.

Te paso el código original en FastAPI a continuación:

(se incluye el código base en Python)

Evaluación del Caso 6

Evaluación funcional:

| Modelo | CRUD completo | Validación de campos | Documentación Swagger | Resultado funcional |
|----------|---------------|----------------------|-----------------------|---------------------|
| Claude | ✓ | ✓ | ✓ | Completo |
| ChatGPT | ✓ | ⚠️ Parcial | ✓ | Parcial |
| DeepSeek | ✓ | ✓ | ✓ | Completo |
| Codex | ✓ | ⚠️ Parcial | ✓ | Parcial |

Evaluación estructural (Rúbrica):

| Modelo | Modularidad | Legibilidad | Uso del framework | Total (15) | Observaciones |
|---------------|--------------------|--------------------|--------------------------|-------------------|---------------------------------------------------------------------------|
| Claude | 5 | 5 | 5 | 15 | Separación clara de capas.
Uso correcto de anotaciones y validaciones. |
| ChatGPT | 4 | 4 | 4 | 12 | Validaciones inconsistentes.
Organización buena. |
| DeepSeek | 5 | 4 | 4 | 13 | Estructura modular clara.
Algunas clases acopladas. |
| Codex | 4 | 3 | 4 | 11 | Código funcional pero débil en validaciones y claridad. |

Errores frecuentes:

- ChatGPT: no valida correctamente `stock >= 0`.
- Codex: permite precios negativos y omite validaciones.
- Algunos modelos no gestionan errores 404 adecuadamente.

Documentación de resultados:

- Capturas de Swagger y consola se almacenaron en el PDF adjunto.
- Las implementaciones están organizadas por carpeta en el repositorio.
- Se recomienda completar una hoja de cálculo con:
 - Evaluación funcional (Sí/No)
 - Puntaje estructural (1-5 por criterio)
 - Observaciones detalladas

¿Querés que te prepare ahora una hoja de cálculo con esta estructura para registrar los resultados de todos los casos?

