

# PROYECTO FINAL

## DESARROLLO DE SOFTWARE V

RecipeApi  
"Generador de  
recetas"

PRESENTANDO POR: LUIS RAMÍREZ,  
DAVID ORTEGA Y CARLOS JÁEN



# RecipeApi

## Página Web

RecipeApi es una página web diseñada en JavaScript que ofrece una experiencia interactiva y moderna para los amantes de la cocina. Su objetivo principal es proporcionar un acceso rápido y eficiente a una amplia variedad de recetas de cocina organizadas por categorías, tipos de comida y dificultad. RecipeApi utiliza un enfoque dinámico y atractivo para mejorar la navegación y la experiencia del usuario.







# RecipeApi

## Página Web

### **Características principales:**

#### **Búsqueda Avanzada:**

Los usuarios pueden buscar recetas por nombre, ingredientes.

#### **Recetas Detalladas:**

Cada receta incluye una lista clara de ingredientes, pasos detallados para la preparación y un tiempo estimado para cocinar.

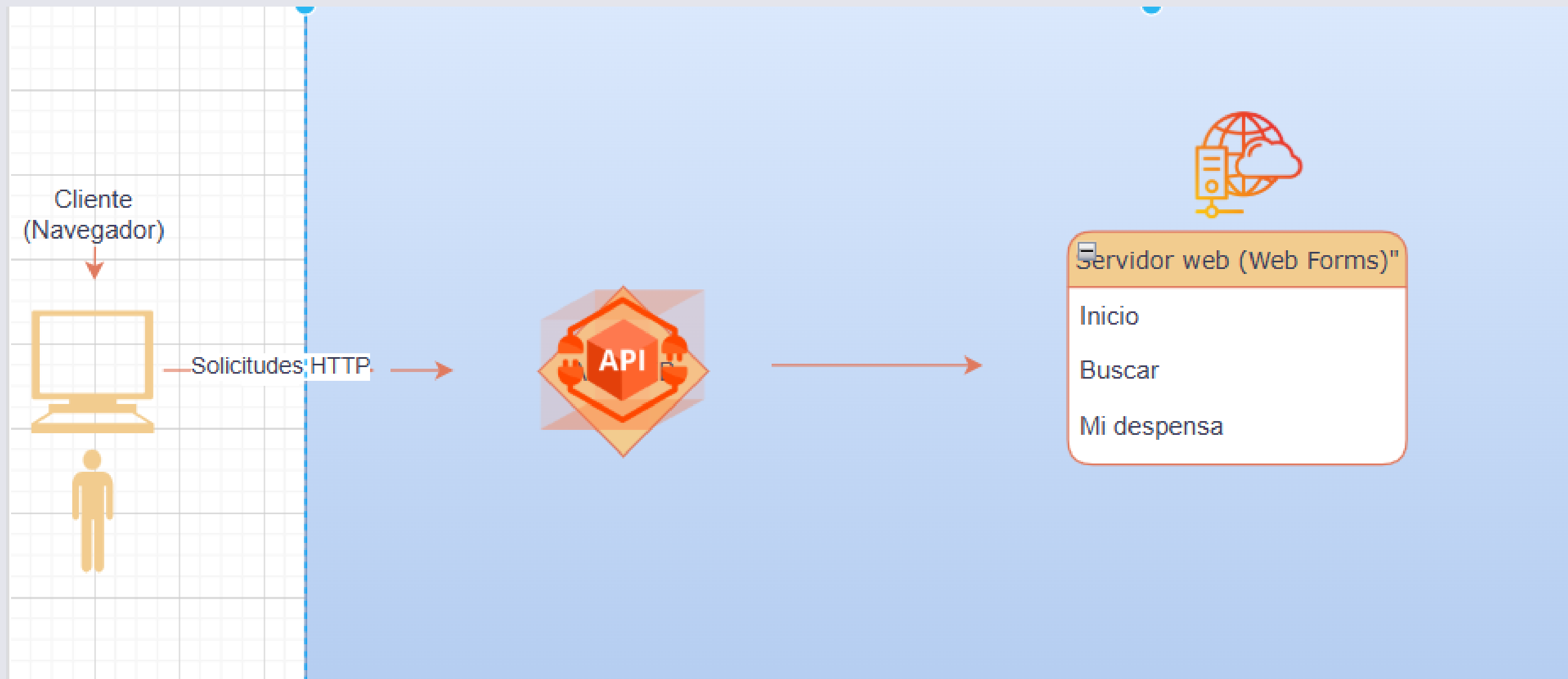
Las recetas cuentan con imágenes de alta calidad.

#### **Integración API:**

RecipeApi aprovecha el poder de APIs públicas o personalizadas para acceder a una extensa base de datos de recetas, proporcionando contenido actualizado y diverso.



# DIAGRAMA DE ALTO NIVEL



# TICKETS GITHUB

The screenshot displays a GitHub Projects board for the project 'ramirez-luis-semestral'. The board is organized into three columns: 'Todo', 'In Progress', and 'Done'. The 'Todo' column is currently empty. The 'In Progress' column is also empty. The 'Done' column contains four completed tasks, each with a title, a priority level (P1 or P2), an estimate, and a status (M or S). The tasks are: 'Implementar la página con la funcionalidad "Mi despensa"' (P1, 10, M), 'Crear página que muestre la información nutricional de cada receta' (P1, 8, M), 'Diseñar la página de búsqueda dinámica' (P1, 10, M), and 'Revisar el diseño final' (P2, 6, S). The board also includes a search bar, a filter dropdown, and a 'New view' button.

2024-SEMII-DESV-1GS222 / Projects / ramirez-luis-semestral

ramirez-luis-semestral

Backlog Team capacity Current iteration Roadmap My items New view

Filter by keyword or by field

**Todo** 0 / 5 Estimate: 0  
This item hasn't been started

**In Progress** 0 / 5 Estimate: 0  
This is actively being worked on

**Done** 11 Estimate: 92  
This has been completed

- ✓ ramirez-luis #9  
Implementar la página con la funcionalidad "Mi despensa"  
P1 10 M
- ✓ ramirez-luis #6  
Crear página que muestre la información nutricional de cada receta  
P1 8 M
- ✓ ramirez-luis #4  
Diseñar la página de búsqueda dinámica  
P1 10 M
- ✓ ramirez-luis #11  
Revisar el diseño final  
P2 6 S

+ Add item



# APLICACIÓN

# RECIPEAPI

"Sabor Inteligente: Tu API de Recetas y Nutrición"

Inicio

Buscar

Detalles

Nutrición

Consejos

Mi Despensa

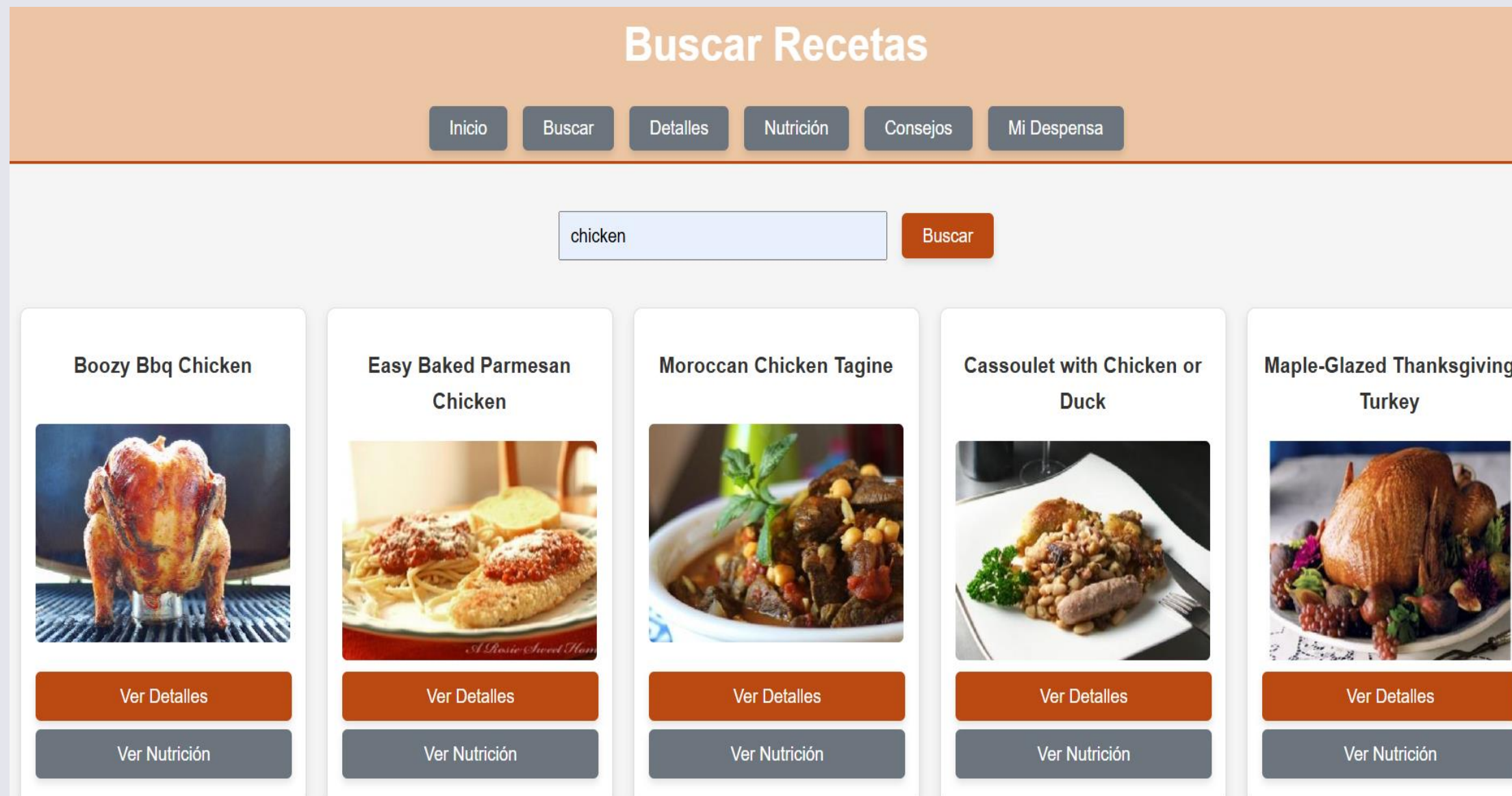


© 2024 Recipe API



# APLICACIÓN

## RECIPEAPI



Comienza a  
buscar  
recetas

Consulta las recetas que  
deseas.

# CÓDIGO FUENTE

```
// Buscar recetas al hacer clic en "Buscar Recetas"
searchRecipesBtn.addEventListener('click', async () => {
  if (ingredients.length === 0) {
    alert('Por favor, agrega al menos un ingrediente.');
```

```
    return;
  }
  const ingredientQuery = ingredients.join(',');
  try {
    const response = await fetch(`https://api.spoonacular.com/recipes/findByIngredients?ingredients=${ingredientQuery}&apiKey=48cd2!`);
    const recipes = await response.json();
    recipesContainer.innerHTML = '';

    if (recipes.length === 0) {
      recipesContainer.innerHTML = '<p>No se encontraron recetas con los ingredientes seleccionados.</p>';
      return;
    }

    recipes.forEach(recipe => {
      const recipeDiv = document.createElement('div');
      recipeDiv.classList.add('recipe');
      recipeDiv.innerHTML = `
        <h3>${recipe.title}</h3>
        
        <a href="recipe-details.html?id=${recipe.id}" class="btn">Ver Detalles</a>
        <a href="nutrition.html?id=${recipe.id}" class="btn">Ver Nutrición</a>
      `;
      recipesContainer.appendChild(recipeDiv);
    });
  } catch (error) {
    console.error('Error al buscar recetas:', error);
  }
});
```



# CÓDIGO FUENTE

```
<!DOCTYPE html>
<html lang="es">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Información Nutricional</title>
  <link rel="stylesheet" href="css/nutrition.css">

  <script type="module">
    import AppFactory from './js/app.js';
    import NavigationFactory from './js/navigation.js';

    document.addEventListener('DOMContentLoaded', () => {
      AppFactory.init();
      NavigationFactory.init();

      // Obtener el ID de la receta desde la URL
      const recipeId = new URLSearchParams(window.location.search).get("id");

      if (recipeId) {
        AppFactory.handlers.loadNutritionInfo(recipeId); // Cargar la información nutricional
      } else {
        document.getElementById('nutrition-info').innerHTML = '<p>No se especificó una receta válida.</p>';
      }
    });
  </script>
</head>
<body>
  <header>
    <h1>Información Nutricional</h1>
    <nav>
      <ul class="nav-links">
        <li><a href="index.html" class="btn" data-page="index.html">Inicio</a></li>
        <li><a href="search.html" class="btn" data-page="search.html">Buscar</a></li>
        <li><a href="recipe-details.html" class="btn" data-page="recipe-details.html">Detalles</a></li>
        <li><a href="nutrition.html" class="btn" data-page="nutrition.html">Nutrición</a></li>
      </ul>
    </nav>
  </header>
</body>
</html>
```

# CÓDIGO FUENTE

```
const AppFactory = (function () {
  const apiKey = "48cd25145c0547acbf9d7af82f1f3314";

  // **htmlElements**: Referencias a los elementos del DOM
  const htmlElements = {
    recipesContainer: document.getElementById("recetas-aleatorias"),
    errorMessage: document.getElementById("error-message"),
    nutritionContainer: document.getElementById("nutrition-info"),
    recipeImage: document.getElementById("recipe-image"),
    btnBack: document.getElementById("btn-back"),
    cookingTipsContainer: document.getElementById("cooking-tips"), // Añadido para consejos
    searchInput: document.getElementById("search-input"), // Campo de búsqueda
    searchBtn: document.getElementById("search-btn"), // Botón de búsqueda
    recipeDetailsContainer: document.getElementById("recipe-details"), // Contenedor para detalles de la receta
  };

  // Función privada para realizar peticiones a la API
  async function fetchFromApi(endpoint) {
    try {
      const response = await fetch(endpoint);
      if (!response.ok) throw new Error("Error al obtener datos de la API");
      return await response.json();
    } catch (error) {
      console.error("Error en fetchFromApi:", error);
      throw error;
    }
  }

  // **handlers**: Lógica principal para manejar eventos y datos
  const handlers = {
    async searchCookingTips(query) {
      if (!query.trim()) return [];

      const endpoint = `https://api.spoonacular.com/recipes/complexSearch?query=${query}&apiKey=${apiKey}`;
      try {
        const data = await fetchFromApi(endpoint);
        const tips = data.results; // Suponiendo que la respuesta tiene un campo "results" con los consejos
      } catch (error) {
        console.error("Error en searchCookingTips:", error);
      }
    }
  };
})();
```

# CÓDIGO FUENTE

```
/* Botones en el header */
.nav-links a {
  text-decoration: none; /* Elimina el subrayado */
  /* padding: 10px 15px; */
  font-size: 1rem;
  border-radius: 5px; /* Bordes redondeados */
  color: white; /* Color inicial del texto */
  background-color: #6c757d; /* Color inicial del botón */
  transition: background-color 0.3s ease, color 0.3s ease; /* Suaviza la transición de color */
}

/* Cambia a naranja al pasar el mouse */
.nav-links a:hover {
  background-color: #ba4a11; /* Color de fondo naranja */
  color: #ffffff; /* Asegura que el texto siga siendo legible */
}

/* Botones */
.btn {
  text-decoration: none;
  padding: 6px 20px;
  font-size: 1rem;
  border-radius: 5px;
  transition: all 0.3s ease, box-shadow 0.3s ease; /* Animación en todos los cambios */
  color: white;
  background-color: #6c757d;
  display: inline-block;
  text-align: center;
  cursor: pointer;
  box-shadow: 0 4px 6px rgba(0, 0, 0, 0.1);
  position: relative;
  overflow: hidden; /* Para animación de fondo */
}
```

```
/* Configuración global */
body {
  font-family: 'Arial', sans-serif;
  margin: 0;
  padding: 0;
  background-color: #f5f5f5;
  color: #333;
  line-height: 1.6;
}

/* Encabezado */
header {
  background-color: #ebc5a4;
  padding: 2px 10px;
  text-align: center;
  color: white;
  border-bottom: 3px solid #ba4a11;
}

header h1 {
  font-size: 2.5rem;
  margin: 0;
}

/* Navegación */
.nav-links {
  list-style: none;
  display: flex;
  justify-content: center;
  gap: 15px;
  padding: 0;
  margin: 20px 0 0;
}

.nav-links li {
  display: inline;
}
```



**GRACIAS!**

