



MIS RECETAS

Paula Blazquez Fors | ETP Xavier | 2023/202

App: <https://paulabf2.pythonanywhere.com/>

Github: <https://github.com/PaulaBlazquezF/MisRecetas>

MOTIVACIÓN

En un mundo cada vez más ocupado y demandante, encontrar tiempo para planificar comidas nutritivas y deliciosas puede resultar todo un desafío. A menudo nos encontramos atrapados en la monotonía de las mismas recetas una y otra vez, o peor aún, recurriendo a opciones poco saludables por falta de inspiración o tiempo.

Es en este contexto surge mi aplicación de recetas, diseñada con el propósito de facilitar la vida de quienes buscan una alimentación variada, equilibrada y sabrosa. Mi objetivo es brindar una solución integral a las necesidades tanto personales como sociales que surgen en torno a la preparación de alimentos.

A nivel personal, me gusta mucho darle importancia a una alimentación adecuada para el bienestar físico y emocional. No soy yo personalmente la que me ocupo de las dietas en mi casa, pero si vivo día a día el cansancio de mi madre al tener que pensar los menús de la semana, así pues, la aplicación viene a resolver este problema ofreciendo una amplia variedad de recetas adaptadas a diferentes gustos, restricciones dietéticas y niveles de habilidad culinaria. Además, la posibilidad de tener una base de datos personalizada permite que pueda incluir recetas de platos que sí nos gustan, de las que ya sabemos para guardarlas y usarlas cuando las necesitemos.

A nivel social, sabemos de la importancia de compartir experiencias gastronómicas y crear vínculos significativos a través de la comida. La integración con una API de recetas ofrece una fuente inagotable de inspiración, promoviendo la exploración de nuevas culturas culinarias y técnicas de cocina.

En resumen, la motivación de la aplicación de recetas ha sido la de hacer menos pesado a la hora de pensar en que cocinar para ti, para tu familia o para la gente que quieras. Es una herramienta que busca empoderar a las personas para que tomen el control de su alimentación de manera saludable, creativa y socialmente enriquecedora. Por último, estoy comprometida en brindar una experiencia que no solo satisfaga las necesidades nutricionales, sino que también inspire y fortalezca la motivación a cocinar, la misma que tengo yo.

ARQUITECTURA

A continuación, vamos a exponer como está estructurada nuestra aplicación web, que herramientas hemos usado para el correcto funcionamiento de esta.

Para empezar, codificamos nuestra aplicación web con **Python**, que es un lenguaje de programación de alto nivel que se destaca por su sintaxis clara y legible. Es ampliamente utilizado en una variedad de campos, desde desarrollo web hasta inteligencia artificial y análisis de datos. Su simplicidad y versatilidad lo convierten en una opción popular entre desarrolladores de todo el mundo.

A parte también se ha usado el microframework de Python **Flask**. A diferencia de otros frameworks más completos como Django, Flask es minimalista y flexible, lo que lo hace ideal para crear aplicaciones web simples y rápidas. Flask sigue el principio de "batteries not included", lo que significa que proporciona solo las herramientas básicas necesarias para crear una aplicación web, dejando al desarrollador la libertad de elegir y agregar las extensiones que necesite. Flask se estructura en torno al concepto de "rutas" y "vistas", donde cada ruta define una URL y una función asociada que se ejecuta cuando se accede a esa URL, así pues en nuestro proyecto podemos ver como encontramos nuestra carpeta *templates* que tiene su css asociado correspondiente y que se vincula con nuestro archivo principal del proyecto (app.py) a través de rutas (@app.route()).

Además, para generar contenido dinámico se ha usado el motor de plantillas de **Jinja**. Este permite la creación de plantillas HTML con marcadores de posición para datos que serán llenados dinámicamente por la aplicación. Jinja utiliza una sintaxis similar a la de Python, lo que facilita su comprensión para los desarrolladores que ya están familiarizados con el lenguaje

En conclusión, Python ha sido el lenguaje de programación principal utilizado en el proyecto, Flask se utiliza como framework para el desarrollo web, proporcionando una estructura ligera para construir aplicaciones web, y Jinja se emplea para la generación de contenido dinámico en las páginas HTML. Juntos, estos elementos han formado una base sólida para la realización y ejecución de mi página web.

REFERENCIAS DE TERCEROS

Para el correcto funcionamiento y la usabilidad de la aplicación, hemos usado además del material e herramientas que hemos descrito en el apartado anterior, ciertos elementos provenientes de fuentes externas, específicamente, código proporcionado por otros individuos para uso personal. A continuación, detallamos las partes de este proyecto que no son de mi autoría.

1. **Index.html**

El archivo *index.html* de mi proyecto ha sido elaborado a partir de un *template* de un individuo que ponía a disposición el código en la aplicación CodePen, en esta se encontraba tanto el código html, como el código css del archivo (*index.css*).

Este código fuente que usamos, no hemos usado todo el código exactamente como se nos presentaba, sino que hemos hecho unos cuantos cambios para que se adecuara a las necesidades, estos han sido:

- Hemos cambiado todo lo que hace referencia a texto, no hay ningún texto que sea el mismo que el código fuente.
- Se ha añadido las rutas pertinentes para que nuestra aplicación tenga interactividad con otras paginas de esta.
- Se ha cambiado una de las dos imágenes que se presentan en la página.

2. **Login**

La funcionalidad de registro y login de la página, hoy por hoy no tiene funcionalidad, se ha propuesto como sugerencia de mejora para un futuro, es decir, los datos que se introducen en estos formularios no funcionan. Así pues, el archivo *login.html* se ha extraído de un código externo de la pagina CodePen, igual que su respectivo css (*login.css*). Finalmente, este archivo requería de funciones JavaScript para su interactividad que se encuentran en el archivo *login.js* también proporcionado de una fuente externa.

Por último, hacemos referencia a que hay partes del código de estilos que hemos usado para el resto de nuestra aplicación que se ha realizado a partir del código externo nombrado anteriormente, como por ejemplo, los colores de la página web, para que toda esta sea uniforme.

3. **Api**

Se ha usado una api externa, para poder realizar la funcionalidad de la página *ideas.html* que teníamos pensado. Para ello se ha usado la pagina ApiRapid, que nos ha proporcionado una apiKey personal que nos permite hacer las llamadas a la api que nosotros hemos escogido.

PROPUESTAS DE MEJORA

- Implementar la función de autenticación de usuarios, para que cada usuario cuente con su sesión y la interfaz sea personalizada según el usuario.
- Poder Marcar como favorita una receta que encuentres en el apartado ideas, y añadir a la base de datos de recetas del usuario.
- Con datos de productos de supermercado, me hubiera gustado implementar la función de poder hacer la lista de la compra, según tus necesidades.
- Hacer útil para aplicación de móvil.