



SunVolt

Autores:

Javier Pardo González
Carlos Sánchez Díaz
Pablo Díaz Calderón

March 11, 2024

Índice

1	Contrato Mercantil	2
2	Arquitectura Global	3
3	Tecnologías a usar en el Proyecto	4
3.1	Backend	4
3.2	Frontend	4
4	Interfaces y estructuras de datos	5
4.1	Interfaces de usuario	5
4.2	Estructuras de datos	5
5	Mockup de la Interfaz Gráfica de Usuario	6
6	Manual de Uso	9

1 Contrato Mercantil

1. *Proveedor de Servicios:* SunVolt
2. *Cliente:* [Nombre del cliente]

Objeto del Contrato: El presente contrato tiene por objeto la prestación por parte de SunVolt de los servicios de desarrollo, implementación y mantenimiento de una solución para la gestión de energía solar.

Descripción del Proyecto: SunVolt desarrollará una solución que permitirá al Cliente aprovechar eficientemente la energía del sol para generar electricidad. La solución incluirá funcionalidades para el monitoreo en tiempo real, análisis de rendimiento, alertas de mantenimiento, y más.

Obligaciones de SunVolt:

1. Diseño y desarrollo de la solución.
2. Pruebas y corrección de errores de la solución.
3. Mantenimiento y actualizaciones periódicas de la solución.

Obligaciones del Cliente:

1. Proporcionar toda la información necesaria para el desarrollo de la solución.
2. Realizar los pagos acordados en las fechas establecidas.

Precio y Forma de Pago: El precio del servicio será de [precio] a pagar en [número de plazos] plazos de [cantidad por plazo].

Vigencia: Este contrato tendrá una vigencia de [número de meses/años], contados a partir de la fecha de firma del presente contrato.

2 Arquitectura Global

En esta sección, vamos a presentar la arquitectura global de nuestra aplicación, SunVolt. Para facilitar la comprensión del funcionamiento de la aplicación, vamos a utilizar dos herramientas visuales: un diagrama de flujo y un diagrama de secuencia.

El **diagrama de flujo** proporcionará una visión general de cómo interactúa un usuario con la aplicación, desde el registro y el inicio de sesión hasta la interacción con diferentes características de la aplicación.

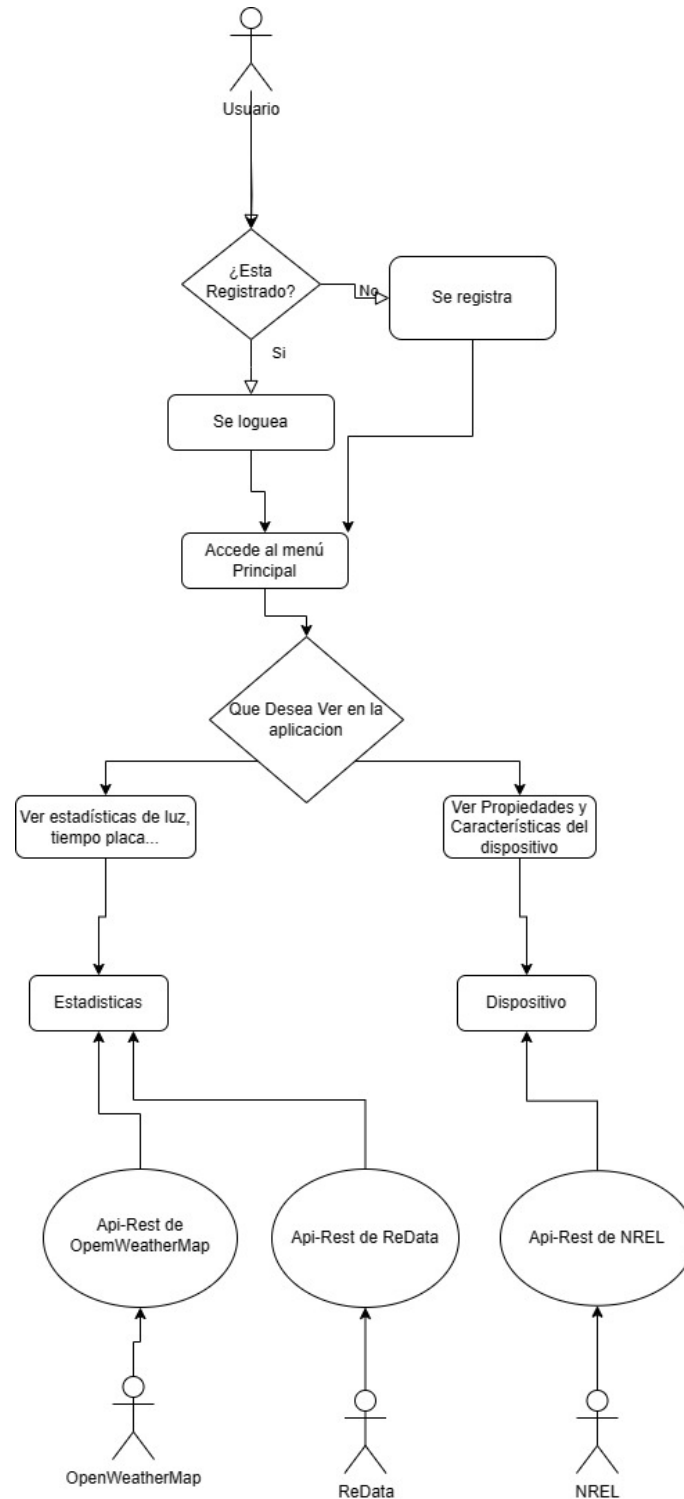


Figure 1: Diagrama de flujo de la interacción del usuario

El **diagrama de secuencia** se centrará en la secuencia de interacciones entre los diferentes componentes de la aplicación durante un escenario de uso específico.

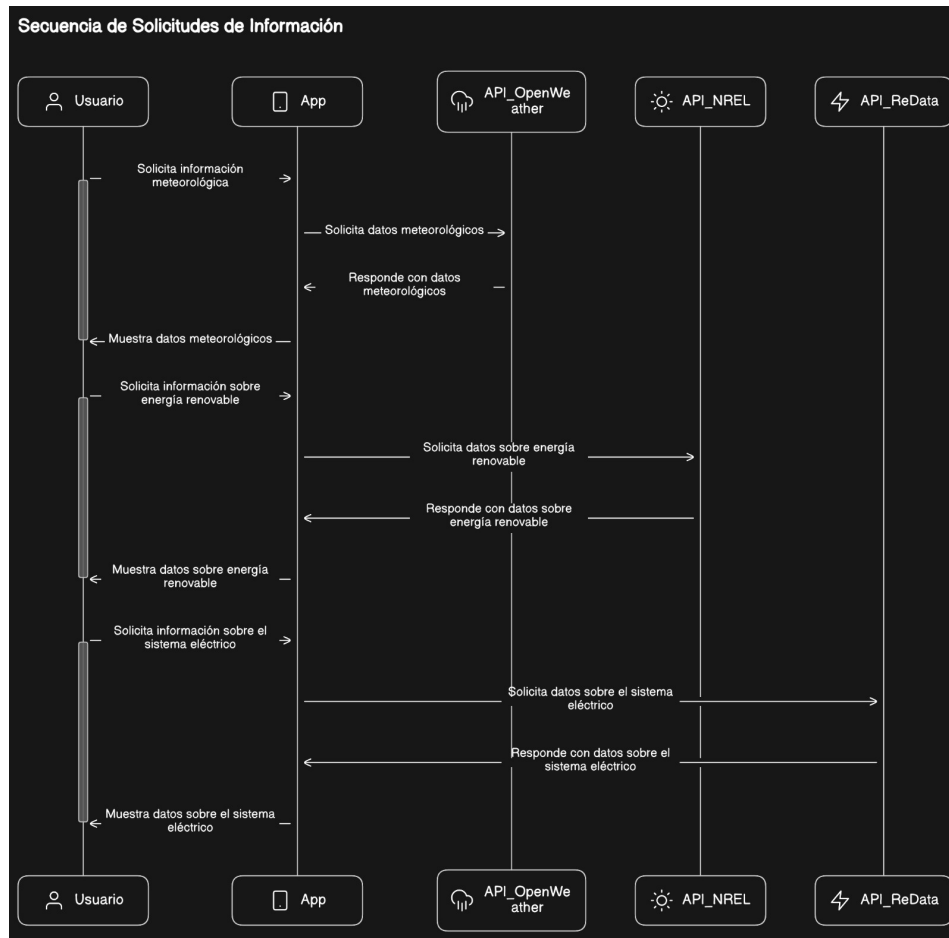


Figure 2: Diagrama de secuencia de la interacción del usuario

Estos diagramas nos ayudarán a entender cómo se estructura la aplicación y cómo fluyen los datos a través de ella. A continuación, procederemos a detallar estos diagramas.

3 Tecnologías a usar en el Proyecto

3.1 Backend

Para el backend de nuestra aplicación, utilizaremos *Python*. Python es un lenguaje de programación versátil y fácil de usar que nos permitirá conectarnos a las APIs necesarias para obtener los datos de producción de energía solar, los precios de la electricidad y las condiciones meteorológicas. Utilizaremos bibliotecas específicas de Python para realizar las solicitudes HTTP y procesar los datos recibidos de las APIs.

3.2 Frontend

Para el frontend de nuestra aplicación, estamos considerando usar *HTML*, *CSS* y *JavaScript*. Estas tecnologías nos permitirán crear una interfaz de usuario interactiva y fácil de usar, donde los usuarios podrán visualizar la información sobre la producción de energía solar, los precios de la electricidad y las condiciones meteorológicas de manera clara y efectiva. Utilizaremos CSS para el diseño y la presentación, y JavaScript para agregar funcionalidades interactivas a la interfaz de usuario.

4 Interfaces y estructuras de datos

En esta sección, describiremos las interfaces de usuario y las estructuras de datos fundamentales que utilizaremos en nuestra aplicación para la gestión y optimización de energía solar. Nuestra aplicación se centrará en proporcionar a los usuarios herramientas intuitivas y eficientes para aprovechar al máximo su uso de la energía solar, integrando datos de diversas fuentes a través de APIs especializadas. Para garantizar una experiencia fluida y una gestión efectiva de la información, diseñaremos interfaces de usuario claras y funcionales, mientras implementamos estructuras de datos robustas para organizar y gestionar los datos de manera eficiente. A continuación, detallaremos las características específicas de nuestras interfaces de usuario y las estructuras de datos clave que respaldarán la funcionalidad de nuestra aplicación.

4.1 Interfaces de usuario

Nuestra aplicación ofrecerá una interfaz de usuario completa que facilitará a los usuarios la gestión y optimización de su uso de energía solar. Algunas características específicas de la interfaz incluirán:

- **Panel de control principal:** Los usuarios accederán a un panel de control principal donde encontrarán información clave sobre la producción de energía solar, los precios de la electricidad y las condiciones meteorológicas actuales.
- **Pronóstico de producción solar:** Implementaremos una sección dedicada a mostrar el pronóstico de producción de energía solar utilizando datos obtenidos de la API de NREL. Los usuarios podrán ver estimaciones precisas de cuánta energía solar se espera generar en los próximos días.
- **Información sobre precios de la electricidad:** Integraremos datos en tiempo real sobre los precios de la electricidad utilizando la API de Open Energy. Esto permitirá a los usuarios tomar decisiones informadas sobre cuándo utilizar la energía solar generada y cuándo utilizar la red eléctrica.
- **Previsión meteorológica:** Incluiremos una sección que mostrará la previsión meteorológica para la ubicación del usuario utilizando la API de OpenWeather. Esto ayudará a los usuarios a planificar su consumo de energía en función de las condiciones climáticas previstas.
- **Herramientas de análisis y recomendaciones:** Desarrollaremos herramientas interactivas de análisis de datos que proporcionarán a los usuarios recomendaciones personalizadas sobre cómo optimizar su uso de energía solar en función de los datos recopilados de las APIs integradas.

4.2 Estructuras de datos

Para respaldar la funcionalidad de nuestra aplicación, utilizaremos diversas estructuras de datos para organizar y gestionar la información de manera eficiente:

- **Almacenamiento de datos de producción solar:** Mantendremos una base de datos actualizada con los datos históricos y pronosticados de producción de energía solar proporcionados por la API de NREL.
- **Registro de precios de la electricidad:** Utilizaremos estructuras de datos para almacenar y actualizar los precios de la electricidad en tiempo real obtenidos de la API de Open Energy.
- **Datos meteorológicos:** Implementaremos una estructura de datos para almacenar y procesar los datos meteorológicos actuales y previstos proporcionados por la API de OpenWeather.
- **Sincronización y gestión de datos:** Desarrollaremos un sistema de sincronización eficiente que integre y actualice automáticamente los datos de las tres APIs para garantizar la precisión y la actualización constante de la información presentada a los usuarios.

Con estas interfaces y estructuras de datos, nuestra aplicación estará equipada para proporcionar a los usuarios una herramienta completa y efectiva para gestionar y optimizar su uso de energía solar de manera inteligente y sostenible.

5 Mockup de la Interfaz Gráfica de Usuario

En esta sección, vamos a presentar varios diseños gráficos que representan cómo sería la interfaz de usuario de nuestra aplicación, SunVolt. Estos diseños, también conocidos como mockups, son representaciones visuales de la interfaz de usuario que se utilizan para planificar y comunicar cómo se verá y funcionará la aplicación.

Los mockups que vamos a mostrar cubren varias partes clave de la aplicación:

1. **Inicio de sesión:** Muestra cómo los usuarios pueden registrarse e iniciar sesión en la aplicación.



Figure 3: Mockup del inicio de sesión

2. **Página principal:** Proporciona información sobre la previsión del tiempo del día en el que nos encontramos, además del consumo actual del hogar del usuario, el consumo actual del usuario por parte de la planta solar y el consumo actual del usuario de la red eléctrica. Aparte, también se pueden visualizar tanto la capacidad de la batería como el precio actual de la luz. Por último, se puede acceder a la pestaña de recomendaciones pulsando en el botón inferior de la página.

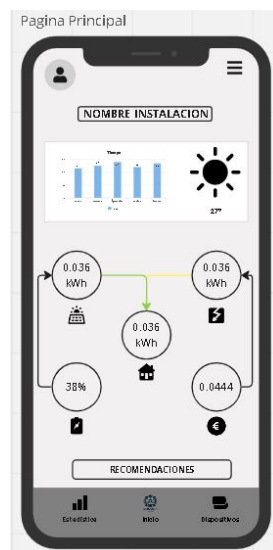


Figure 4: Mockup de la página principal

3. **Recomendaciones:** Presenta cómo la aplicación proporciona recomendaciones personalizadas a los usuarios en función de su consumo y la previsión del tiempo.



Figure 5: Mockup de las recomendaciones

4. **Estadísticas:** Muestra cómo los usuarios pueden ver y analizar diversas estadísticas relacionadas con su consumo tanto de la red eléctrica como por parte de la planta solar.



Figure 6: Mockup de las estadísticas

5. **Dispositivos:** Detalla cómo los usuarios pueden interactuar con los dispositivos conectados.

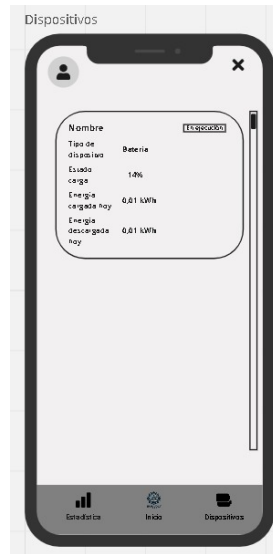


Figure 7: Mockup de los dispositivos

6. **Perfil:** Permite a los usuarios ver y modificar su perfil y configuración de la aplicación.



Figure 8: Mockup del perfil

6 Manual de Uso

En esta sección, vamos a proporcionar un manual de uso para los tres archivos: **NREL.py**, **REData.py** y **OpenWeatherMap.py**. Este manual te guiará sobre cómo ejecutar estos archivos y cómo configurar cualquier token necesario.

Antes de ejecutar estos archivos, es importante instalar todas las bibliotecas o paquetes necesarios. Para hacer esto, deberás ejecutar el archivo **requirements.txt**. Aquí tienes un ejemplo de cómo podrías hacerlo:

```
pip install -r requirements.txt
```

Una vez que hayas instalado todas las dependencias necesarias, puedes proceder a ejecutar los archivos:

1. **NREL.py y REData.py**: Estos dos archivos no requieren ningún token para su ejecución. Simplemente puedes ejecutarlos en tu entorno Python. Asegúrate de tener todas las dependencias necesarias instaladas. Aquí tienes un ejemplo de cómo podrías ejecutar estos archivos:

```
python NREL.py
python REData.py
```

2. **OpenWeatherMap.py**: Para este archivo, necesitarás registrarte y obtener un token. Sigue estos pasos:

- (a) Ve a la página web de OpenWeatherMap y regístrate para obtener una cuenta.
- (b) Una vez que hayas creado tu cuenta, podrás generar un token. Este token es necesario para hacer solicitudes a la API de OpenWeatherMap.

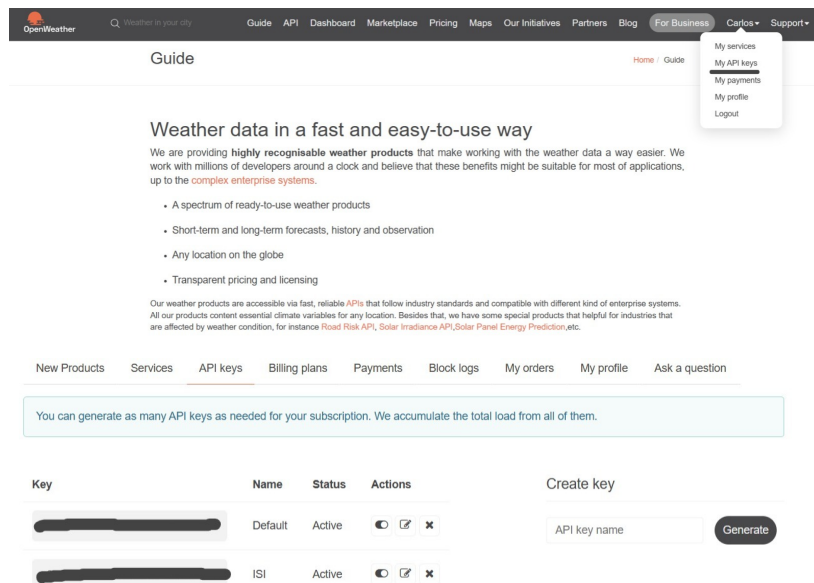


Figure 9: Generación del token en OpenWeatherMap

- (c) Una vez que hayas obtenido tu token, debes sustituirlo en el archivo **OpenWeatherMap.py** donde pone **TOKEN**. Aquí tienes un ejemplo de cómo podría verse:

```
API_KEY = "tu_api_key"
```

- (d) Después de reemplazar el token, puedes ejecutar el archivo **OpenWeatherMap.py** de la misma manera que los otros archivos:

```
python OpenWeatherMap.py
```