**Projet personnel : Simulateur de production d’énergie solaire en fonction de la météo**

**Objectif :**

Développer un outil logiciel en Python permettant de simuler la production d’énergie électrique d’un panneau solaire photovoltaïque installé en un lieu donné, en tenant compte des conditions météorologiques locales. Ce projet vise à illustrer l’impact direct de l’irradiance solaire et des facteurs environnementaux sur la performance énergétique des installations solaires, thème central dans la transition énergétique actuelle.

**Description du projet :**

Le simulateur récupère des données météorologiques, telles que l’irradiance solaire et la température, à partir de sources publiques ou d’API météo. Ces données sont utilisées pour calculer la puissance instantanée produite par un panneau solaire, en appliquant un modèle physique simple intégrant la surface du panneau, son rendement nominal et un facteur de performance pour les pertes réelles (ombrage, température, câblage, etc.).

La production énergétique est ensuite calculée sur des périodes données (heure, jour), et visualisée graphiquement pour permettre une analyse claire et intuitive des variations liées à la météo. Le projet inclut également une interface utilisateur basique développée avec Streamlit, facilitant l’entrée des paramètres (localisation, surface du panneau) et l’affichage des résultats.

**Étapes clés du projet :**

1. **Collecte des données météo** : Intégration d’une API météo pour récupérer en temps réel ou en données historiques les paramètres essentiels (irradiance, température).
2. **Modélisation physique** : Implémentation de la formule de calcul de la puissance photovoltaïque en fonction de l’irradiance et correction selon la température.
3. **Calcul de l’énergie produite** : Agrégation des puissances instantanées sur des intervalles de temps pour estimer la production énergétique totale.
4. **Visualisation des résultats** : Création de graphiques interactifs pour représenter la production solaire et ses variations sur différentes périodes.
5. **Développement d’une interface utilisateur** : Utilisation de Streamlit pour simplifier l’usage du simulateur par des utilisateurs non techniques.
6. **Documentation et mise en ligne** : Rédaction d’une documentation claire du code, mise à disposition sur un dépôt GitHub public.

**Compétences mobilisées :**

* Programmation Python, utilisation d’API externes
* Modélisation physique appliquée aux systèmes énergétiques
* Analyse et traitement de données temporelles
* Visualisation de données avec matplotlib/plotly
* Développement d’interfaces simples avec Streamlit

Ce projet permet de renforcer mes compétences à l’interface entre informatique et énergie, tout en démontrant ma capacité à concevoir des outils concrets pour analyser les performances des énergies renouvelables.

**Production (kWh) = Puissance nominale (kWc) x Ensoleillement (h) x PR**

Une image contenant texte, capture d’écran, Police

Le contenu généré par l’IA peut être incorrect.

**Donc Production (kWh) = Surface x R x Ensoleillement (h) x PR**

**Production annuelle (en kWh) = Puissance (en kWc) × Ensoleillement moyen (en kWh/m²/an) × Rendement global de l’installation**

[How to Calculate Solar Panel Output? | Sunbase](https://www.sunbasedata.com/blog/how-to-calculate-solar-panel-output?utm_source=chatgpt.com)

Je fais un projet python sur vscode. Le but est de récupérer les données nécessaires comme l'irradiation solaire sur un fichier, pour l'envoyer à un autre fichier. Cette autre fichier fera des calculs physiques à partir des données récupérées, pour simuler la production d'énergie d'un panneau solaire. En t'aidant de la partie usage de cette page, code moi le fichier API\_météo qui récupèrera les données heure par heure pour une durée donnée

python -m streamlit run app.py

python -m streamlit run test.py

SOURCES:

[Colors and borders - Streamlit Docs](https://docs.streamlit.io/develop/concepts/configuration/theming-customize-colors-and-borders)

https://www.flaticon.com/free-icon