

# Projet : FastApiDecoratorBuilder

## Description du projet

Le but du projet *FastApiDecoratorBuilder* est de concevoir un décorateur Python qui transforme une fonction Python en une API FastAPI basée sur la fonction et des configurations définies.

## Objectifs du projet

- **Création de Décorateur** : Construire un décorateur qui transforme une fonction Python en API FastAPI.
- **Gestion de Configurations** : Implémenter un mécanisme de configuration pour l'API.

## Consignes

1. **Développement du Décorateur** :
  - Élaborer un décorateur qui, appliqué à une fonction, génère une API FastAPI correspondante.
2. **Configuration de l'API** :
  - Intégrer une méthode pour configurer les propriétés de l'API générée, telles que les routes et les méthodes HTTP acceptées.
3. **Exemple de test** :
  - Implémenter des cas pratique pour les cas d'usages de votre package.

## Critères d'évaluation

- **Fonctionnalité** : Est-ce que le décorateur produit une API fonctionnelle ?
- **Configurabilité** : La configurabilité de l'API est-elle suffisamment flexible ?
- **Documentation** : Est-ce que votre documentation décrit clairement comment utiliser le décorateur et configurer l'API ?
- **Structure du code** : Organisation et modularité du code, Utilisation appropriée des concepts de programmation orientée objet vus en cours, Clarté et propreté du code (pas de redondance, noms de variables explicites, etc.).

## Ressources supplémentaires

- <https://fastapi.tiangolo.com/>
- <https://realpython.com/primer-on-python-decorators/>

## Note

Assurez-vous d'inclure une documentation détaillée, y compris un fichier `README.md`, qui explique comment utiliser votre décorateur et configurer l'API générée.