Veille Dev Web

00000

Frontend vs Backend vs Fullstack

Frontend: Partie visible par l'utilisateur dans un navigateur web, impliquant HTML, CSS, JavaScript.

Backend: Partie serveur, gestion de la base de données, logique métier. Utilise des langages comme Java, Python, Ruby…

Fullstack : Combinaison de frontend et backend, capable de gérer les deux côtés.

Architectures Web

MVC (Model-View-Controller):

Sépare l'application en trois composants principaux : modèle (données), vue (interface utilisateur), contrôleur (logique métier).

Exemple: Frameworks web comme Ruby on Rails, Django (Python), Laravel (PHP), Spring MVC (Java).

Architecture REST (Representational State Transfer):

Basée sur les principes architecturaux du web. Les ressources (data ou services) sont identifiées par des URL et manipulées à l'aide des méthodes HTTP.

Exemple : Utilisé dans les API web, comme les API RESTful.

Architectures Web (suite)

Architecture Microservices:

Divise l'application en petits services indépendants qui peuvent être développés, déployés et mis à l'échelle de manière indépendante.

Exemple : Netflix, Uber.

Architecture PWA (Progressive Web App):

Applications web qui utilisent des technologies modernes pour fournir une expérience utilisateur similaire à celle des applications natives.

Exemple : Twitter, Pinterest.

Frameworks web en python

Django:

Un framework web complet qui suit le modèle MVC. Il offre une ORM intégrée, une gestion automatique de l'administration, et des outils puissants pour le développement rapide d'applications.

Flask:

Micro-framework, facile à utiliser et étendre avec des plugins.

Bottle:

Un micro-framework simple et léger.

Pyramid:

Un framework web modulaire qui permet aux développeurs de choisir les composants dont ils ont besoin.

CherryPy:

Un framework léger qui adopte une approche orientée objet pour le développement web.







Pyramid





Schéma

