

Découvrir l'architecture logicielle

L'architecture logicielle décrit **la structure globale d'un logiciel** et la manière dont ses différentes parties s'organisent et coopèrent. Avant de parler de langages, d'algorithmes ou de lignes de code, l'architecture répond à une question simple : **comment le système est construit et comment ses composants interagissent.**

5TTR

6TTR



Découverte

À quoi sert l'architecture logicielle ?

L'architecture sert à **prendre les bonnes décisions avant d'implémenter** :

- découper le système en composants clairs ;
- définir les responsabilités de chacun ;
- organiser les échanges de données ;
- limiter les dépendances inutiles.

Elle permet de rendre un logiciel :

- compréhensible ;
- évolutif ;
- maintenable ;
- testable.

Analogie : construire une maison

Construire un logiciel, c'est comparable à construire une maison.

Avant de poser des briques :

- on dessine un plan ;
- on décide où sont les pièces ;
- on prévoit les portes, les couloirs, les arrivées d'eau et d'électricité.

Ce plan n'explique pas :

- comment on pose chaque brique ;

- quel outil précis est utilisé.

Il explique **comment tout s'assemble**.

L'architecture logicielle joue exactement ce rôle pour un logiciel.

Architecture ≠ code

L'architecture ne décrit pas :

- le détail des fonctions ;
- les algorithmes ;
- la syntaxe d'un langage.

Elle décrit :

- les composants principaux ;
- leurs responsabilités ;
- leurs relations ;
- les règles de communication entre eux.

Deux équipes peuvent écrire un code très différent tout en respectant **la même architecture**.

Pourquoi l'architecture est indispensable

Sans architecture claire :

- le projet devient confus ;
- les dépendances se multiplient ;
- chaque modification devient risquée.

Avec une architecture réfléchie :

- plusieurs personnes peuvent travailler en parallèle ;
 - les erreurs sont plus faciles à isoler ;
 - le système peut évoluer sans être réécrit.
-

Une discipline transversale

L'architecture logicielle concerne :

- les applications web ;
- les API ;
- les programmes embarqués ;
- les systèmes distribués ;
- les applications locales ou connectées.

Quel que soit le contexte, l'objectif reste le même :

ORGANISER UN SYSTÈME COMPLEXE POUR LE RENDRE MAÎTRISABLE.

Ce qu'il faut retenir

- L'architecture logicielle définit **la structure d'un logiciel**.
- Elle se pense **avant** et **pendant** le développement.
- Elle sépare les responsabilités et organise les interactions.
- Elle conditionne la qualité du logiciel sur le long terme.

Cette section du cours abordera les principes essentiels d'architecture logicielle, tels qu'ils sont appliqués dans des projets concrets et professionnels.