

# Cours Génie Logiciel (partie 1)

EILCO-ING2

ABAKARIM Fadwa

# Définitions

**Un logiciel** est un ensemble de programmes, d'instructions et de données qui permettent à un ordinateur d'exécuter des tâches spécifiques.

## Exemples :



Logiciel de Bureautique



Logiciel de Navigation Web



Logiciel de Messagerie



Logiciel de Communication

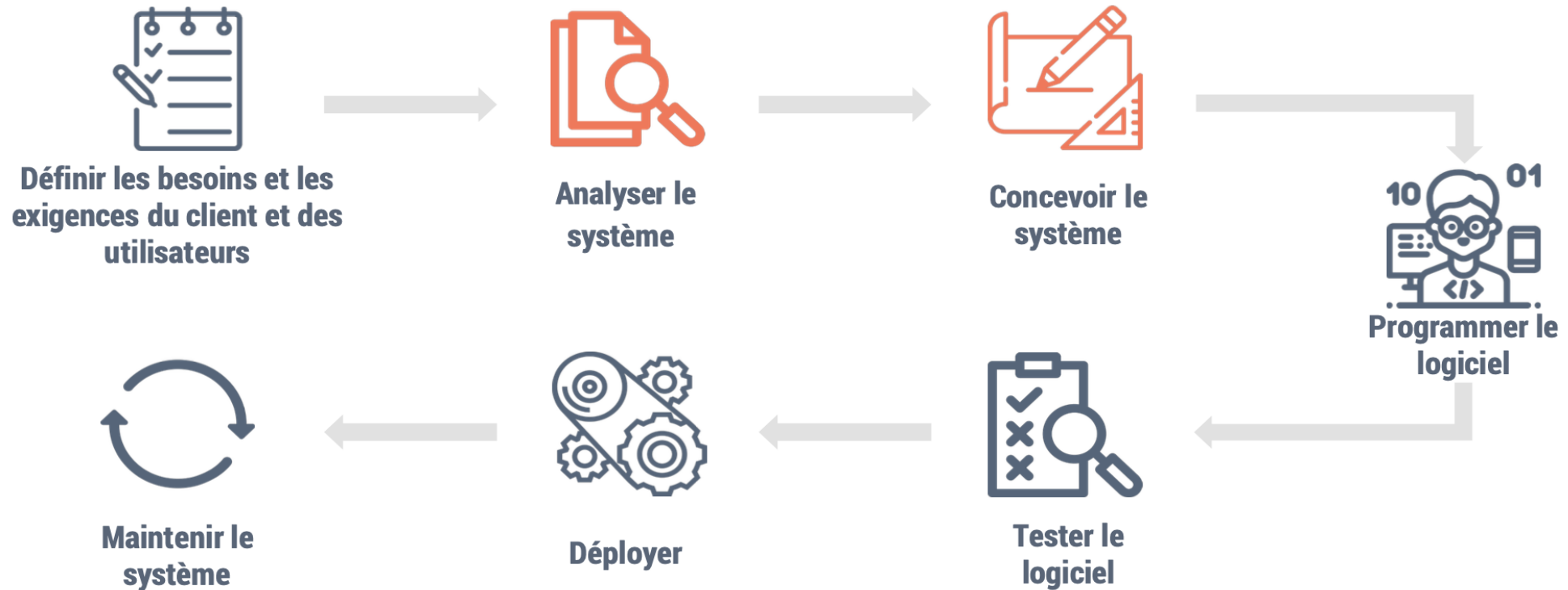
# Définitions

**Le génie logiciel** est une discipline de l'informatique qui consiste à concevoir, développer, tester et maintenir des logiciels de manière structurée et méthodique.

Son objectif est de créer des logiciels de haute qualité, fiables, et faciles à maintenir.

# Les étapes de développement logiciel

Le processus de développement logiciel contient un certain nombre d'étapes :



# Les étapes de développement logiciel

Etape 1 : Définition des besoins et des exigences.

- Objectif : Comprendre précisément ce dont le client et les utilisateurs ont besoin.
- Comment : Nous discutons avec eux pour identifier :
  - Qui va utiliser le logiciel ?
  - Quelles actions ils doivent pouvoir effectuer ?



# Les étapes de développement logiciel

## Etape 2 : Analyser le système.

- Objectif : Affiner et approfondir les besoins définis à l'étape précédente.
- Détails : À ce stade, on spécifie plus précisément le fonctionnement interne du futur logiciel en se concentrant sur le COMMENT cela doit fonctionner.



# Les étapes de développement logiciel

## Etape 3 : Conception du système.

- Objectif : La conception du système consiste à définir les choix techniques nécessaires pour la réalisation du logiciel.
- Comment : On utilise souvent des diagrammes ou des schémas pour montrer clairement comment le logiciel va fonctionner, ce qui aide les développeurs à le créer.



# Les étapes de développement logiciel

Etape 4 : Programmer le logiciel.

La programmation est l'étape où les développeurs écrivent le code du logiciel à l'aide de langages de programmation (comme Java, Python, C++), de systèmes de gestion de bases de données, et d'autres outils techniques.





# Les étapes de développement logiciel

Etape 5 : Tester le logiciel.

Objectif : Durant les tests, les développeurs vérifient que le logiciel fonctionne correctement et qu'il répond bien aux besoins définis au début du projet.

Détails : Cette phase peut inclure des tests avec le client et/ou les utilisateurs pour valider que le logiciel correspond à leurs attentes.



# Les étapes de développement logiciel

## Etape 6 : Déployer.

L'objectif du déploiement est d'installer le logiciel dans l'environnement réel des utilisateurs (serveurs, ordinateurs, etc.). Cela inclut l'installation, la configuration et la mise en service du logiciel.



# Les étapes de développement logiciel

Etape 7 : Maintenir le système.

La maintenance consiste à corriger les erreurs, résoudre les problèmes qui apparaissent, et effectuer des mises à jour pour améliorer le logiciel ou l'adapter aux nouveaux besoins des utilisateurs.



# Cycle de vie d'un logiciel

Les étapes de développement d'un logiciel ne suivent pas toujours un ordre linéaire strict.

On parle alors de **cycles de vie**, qui sont des méthodes permettant d'organiser ces étapes de différentes manières en fonction des critères spécifiques au projet (comme sa taille, sa complexité, ou les attentes du client).

# Cycle de vie d'un logiciel

Il existe plusieurs modèles de cycle de vie du logiciel, chacun ayant ses propres caractéristiques. Parmi les plus courants, on trouve :

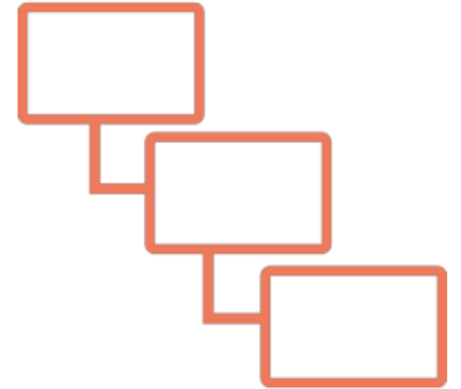
- Le modèle en cascade.
- Le modèle en V.
- Le modèle en spirale.
- Le modèle évolutif.

# Le modèle en cascade

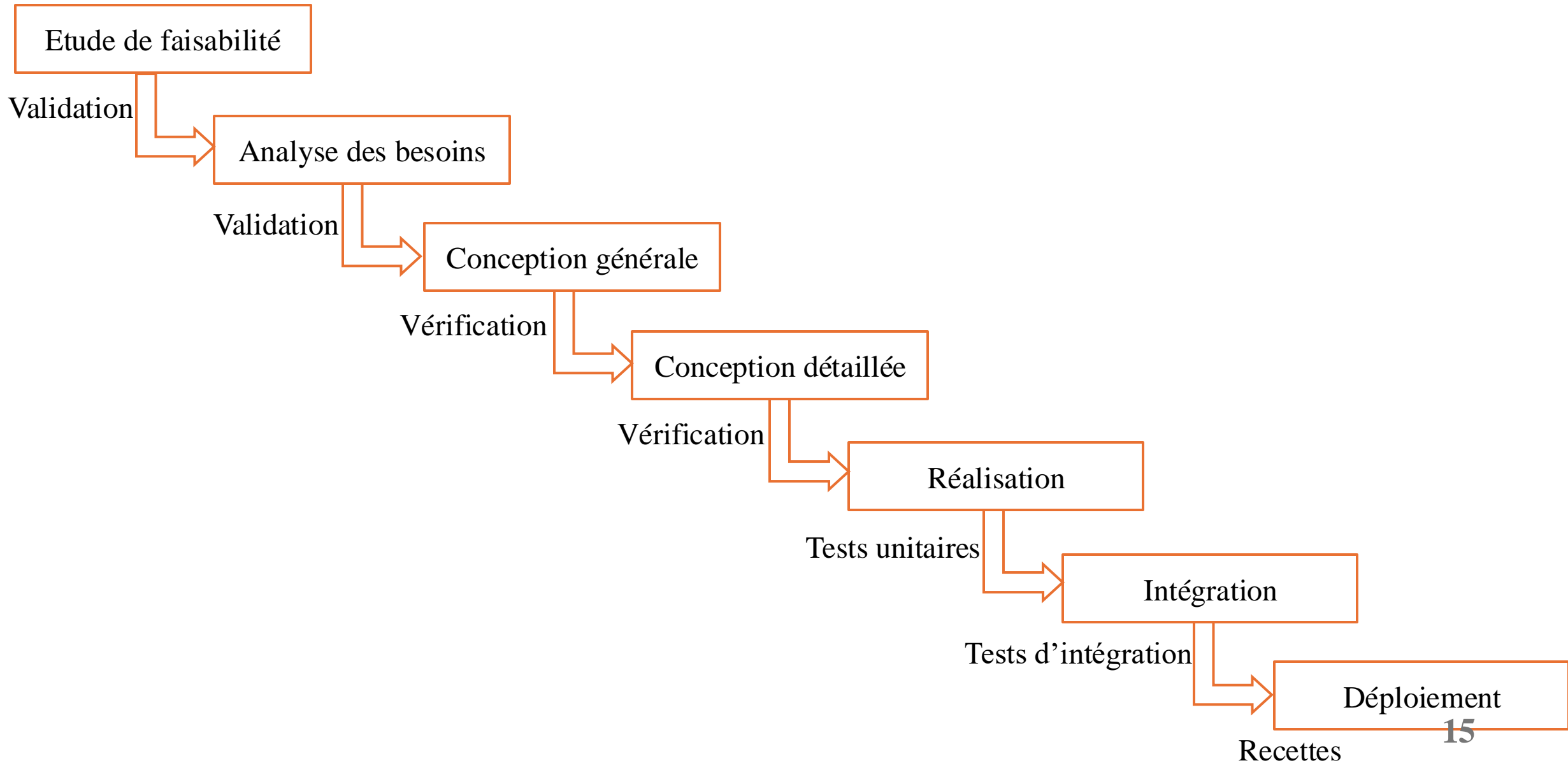
Le modèle en cascade (Waterfall model) est un des modèles de cycle de vie du développement logiciel les plus traditionnels et linéaires.

Dans ce modèle, les étapes de développement sont effectuées de manière séquentielle, une après l'autre, sans retour en arrière possible.

Chaque étape doit être entièrement terminée avant de passer à la suivante, comme une cascade qui descend de niveau en niveau.



# Le modèle en cascade



# Le modèle en cascade

1. Étude de faisabilité : On vérifie si le projet est possible et utile à réaliser.
2. Analyse des besoins : On définit ce que le logiciel doit faire en discutant avec le client et les futurs utilisateurs.
3. Conception générale : On planifie le fonctionnement global du logiciel.
4. Conception détaillée : On spécifie les détails techniques pour réaliser le logiciel.



# Le modèle en cascade

5. Réalisation : Les développeurs écrivent le code pour créer le logiciel.
6. Intégration : On assemble les différentes parties du logiciel et fait des vérifications pour s'assurer qu'il fonctionne bien.
7. Déploiement : Le logiciel est installé pour que les utilisateurs puissent l'utiliser.

# Le modèle en cascade

## Avantages :

- **Facile à suivre** : Les étapes sont bien organisées et se déroulent dans un ordre clair, ce qui le rend simple à comprendre.
- **Étapes bien définies** : Chaque étape est distincte, ce qui permet de suivre facilement les progrès.
- **Bonne documentation** : À chaque étape, on prend des notes précises, ce qui aide à mieux comprendre tout le projet.
- **Bonne gestion** : Ce modèle convient aux projets où on sait exactement ce qu'on veut dès le début.

# Le modèle en cascade

## Inconvénients

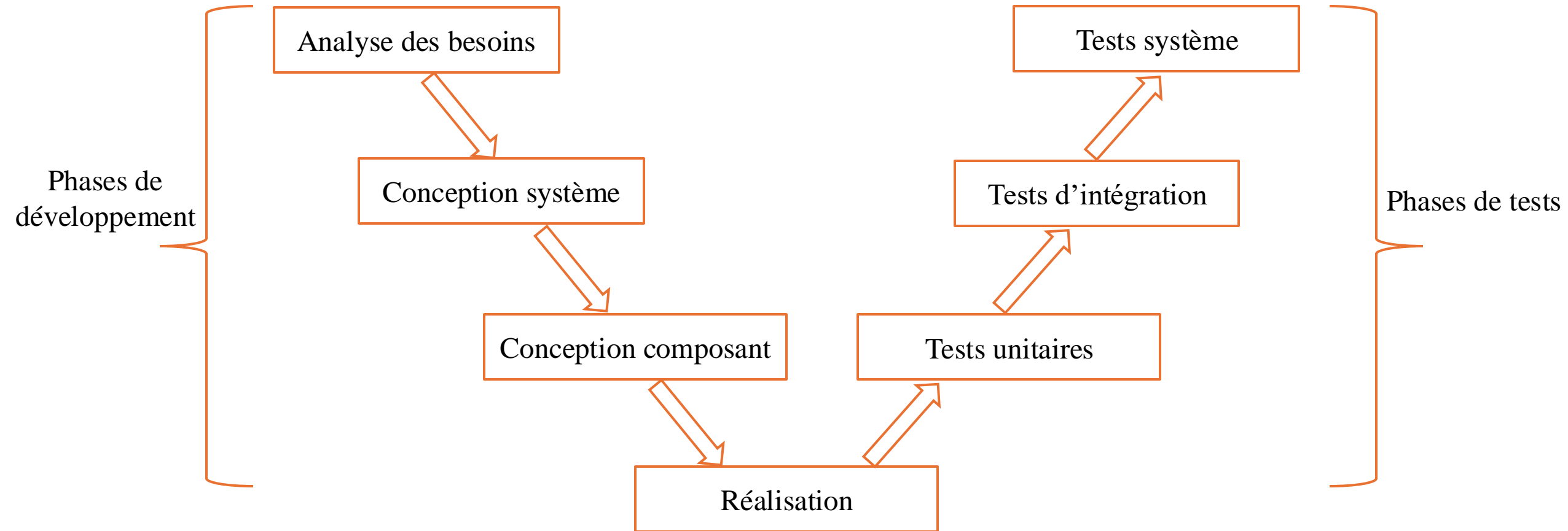
- **Peu flexible** : Difficile de revenir en arrière après chaque étape.
- **Risque élevé d'erreurs** : Les erreurs dues à des exigences mal comprises ou changeantes ne sont découvertes qu'à la fin.
- **Temps d'attente pour les clients** : Les clients découvrent le produit final uniquement à la fin, ce qui peut générer des insatisfactions si le produit ne répond pas aux attentes.
- **Adaptabilité limitée aux changements** : Moins adapté aux projets où les exigences peuvent évoluer en cours de développement.

# Le modèle en V

Le modèle en V est une méthode de développement logiciel qui, comme le modèle en cascade, suit un processus séquentiel. Cependant, il met l'accent sur la vérification et la validation à chaque étape de développement.

Dans ce modèle, chaque étape de développement a une phase de test associée, créant ainsi une structure en forme de "V".

# Le modèle en V



# Le modèle en V

1. Analyse des besoins : Cette étape est équivalente à la définition des besoins et des exigences. Elle consiste à identifier et à documenter les attentes du client et des utilisateurs.
2. Conception système : On crée une vue d'ensemble du logiciel, en identifiant ses principales fonctionnalités et sa structure générale. C'est une première idée de ce que le système fera.

# Le modèle en V

3. Conception des composants : On divise le système en composants spécifiques et on procède à l'analyse et à la conception détaillée de chaque composant. Cela inclut le fonctionnement interne de chaque module pour préparer la phase de réalisation.
4. Réalisation (programmation des composants) : À cette étape, on écrit le code pour chaque composant, en suivant les plans de conception. Cela correspond à la phase de programmation.

# Le modèle en V

5. Tests unitaires : Chaque composant est testé de façon indépendante pour s'assurer qu'il fonctionne correctement et répond aux spécifications établies dans les étapes précédentes.
6. Tests d'intégration : Une fois les composants créés, on les assemble et on vérifie qu'ils fonctionnent bien ensemble.
7. Tests système : On teste le système dans son ensemble pour s'assurer que toutes les parties travaillent ensemble de manière fluide et cohérente.



# Le modèle en V

## Avantages :

- **Clarté et organisation** : Les étapes sont bien définies et suivent un ordre logique, ce qui facilite la gestion du projet.
- **Tests réguliers** : Des tests sont effectués à chaque étape pour repérer rapidement les erreurs et éviter les problèmes plus tard.
- **Qualité assurée** : Les tests à chaque phase garantissent que le logiciel fonctionne correctement et est de bonne qualité.

# Le modèle en V

## Inconvénients

- **Rigidité** : Le modèle en V suit une séquence stricte, rendant difficile l'ajustement des éléments majeurs en cours de projet si des changements importants sont nécessaires.
- **Coût élevé des tests** : L'accent mis sur les tests à chaque phase peut augmenter les coûts et le temps de développement, notamment si de nombreux tests sont nécessaires.
- **Adaptabilité limitée** : Si les exigences ou les besoins du client changent après la phase de conception, il peut être compliqué de réajuster le système, ce qui peut entraîner des retards.

# Exercice 1

Veillez remplir ce tableau pour déterminer la différence entre le modèle en cascade et le modèle en V.

Caractéristique	Modèle en Cascade	Modèle en V
Approche		
Tests		
Coût et temps		
Risques		

# Exercice 2

Une entreprise adopte un processus de développement logiciel en suivant une série d'étapes bien définies : collecte des exigences, conception, développement, intégration, tests et déploiement.

Dans ce modèle, l'équipe de développement ne prévoit pas de revenir aux étapes précédentes, sauf si des problèmes majeurs sont découverts lors des tests. Ce processus est surtout efficace pour des projets avec des spécifications claires et des objectifs bien définis dès le début.

- Identifiez le modèle de cycle de vie du développement logiciel utilisé par cette entreprise.

# Exercice 3

Une entreprise suit un processus de développement logiciel structuré où chaque phase est accompagnée d'une phase de test correspondante.

Après avoir analysé les besoins et conçu le système, l'équipe de développement crée des modules individuels, chacun étant testé séparément avant d'être intégré au système complet. Les tests sont effectués de manière systématique à chaque étape du développement pour garantir la conformité aux exigences.

- Identifiez le modèle de cycle de vie du développement logiciel utilisé par cette entreprise.

Suite ...