

# Présentation Génie Logiciel

équipe DevSquad



## Notre équipe:



Adam Messaoudi



Ines Bessaad



Haythem Loueti



Ranim Rabah



# Sommaire

1. Le principe de fonctionnement de chaque cycle de vie
2. Les avantages et inconvénients de chaque modèle
3. Comparaison entre les deux cycles de vie
4. Types d'applications qui utilisent ces cycles de vie



# **1. Le principe de fonctionnement de chaque cycle de vie**



# Cycle de vie en cascade

Le principe de fonctionnement du modèle de cycle de vie en cascade est basé sur une approche séquentielle linéaire pour le développement de logiciels. Le modèle est appelé "en cascade" car chaque phase est terminée avant de passer à la suivante. Les phases du modèle en cascade sont les suivantes :

- ❖ Analyse des besoins
- ❖ Conception
- ❖ Implémentation
- ❖ Tests
- ❖ Maintenance



# Cycle de vie en spirale

Le modèle en spirale permet de gérer les risques et les incertitudes de manière efficace tout au long du projet. Il s'agit d'un processus itératif qui permet de réduire les risques à chaque itération en évaluant les résultats et en apportant des améliorations.

Le modèle est appelé "en spirale" car il suit une boucle continue de phases de développement itératives, qui se répètent jusqu'à ce que le logiciel soit terminé.

Les quatre phases principales du modèle en spirale sont :

- ❖ Planification
- ❖ Analyse des risques
- ❖ Développement
- ❖ Evaluation



## 2. Les avantages et inconvénients de chaque modèle

# Les avantages et les inconvénients du cycle de vie cascade :



Avantages	Inconvénients
<b>Structure claire :</b> Le modèle de cycle de vie en cascade suit un processus séquentiel structuré et bien défini, ce qui le rend facile à comprendre et à suivre.	<b>Rigidité :</b> Le modèle de cycle de vie en cascade est très rigide et il est difficile de modifier les étapes du processus de développement une fois qu'elles ont été achevées. Les changements nécessaires peuvent donc entraîner des retards et des coûts supplémentaires.
<b>Documentation :</b> Chaque phase du modèle de cycle de vie en cascade est accompagnée de documentation détaillée, ce qui facilite la gestion du projet, la communication avec les parties prenantes et la maintenance future du logiciel.	<b>Difficulté de tester :</b> Les tests sont effectués à la fin de chaque phase du processus, ce qui peut rendre difficile la détection de certains problèmes, en particulier ceux qui ne se manifestent qu'à des étapes ultérieures.
<b>Meilleure planification :</b> Le modèle de cycle de vie en cascade permet une planification plus précise et une estimation des coûts, car chaque phase est clairement définie et les résultats sont facilement prévisibles.	<b>Manque de flexibilité :</b> Le modèle de cycle de vie en cascade ne convient pas aux projets complexes qui ont besoin de beaucoup de flexibilité et d'itération. Il peut être difficile de s'adapter aux changements de conditions ou de besoins des clients.
<b>Contrôle de qualité :</b> Le modèle de cycle de vie en cascade permet un contrôle de qualité rigoureux à chaque étape du processus de développement, ce qui peut aider à identifier les erreurs et les problèmes dès le début.	<b>Manque d'implication du client :</b> Le modèle de cycle de vie en cascade implique généralement une faible interaction avec les clients et les utilisateurs finaux, ce qui peut entraîner des lacunes dans les exigences et les fonctionnalités attendues.



# Les avantages et les inconvénients du cycle de vie en spirales :



Avantages	Inconvénients
<b>Flexibilité</b> : Le modèle de cycle de vie en spirale permet une plus grande flexibilité que le modèle en cascade. Il permet aux développeurs de revenir en arrière et de corriger les erreurs à tout moment, tout en conservant une structure claire et définie	<b>Complexité</b> : Le modèle de cycle de vie en spirale est plus complexe que le modèle en cascade, ce qui peut rendre le processus de développement plus difficile à gérer et à suivre.
<b>Réduction des risques</b> : Le modèle de cycle de vie en spirale est conçu pour identifier et atténuer les risques à chaque étape du processus de développement, ce qui permet de réduire les risques liés au développement de logiciels.	<b>Coût et temps</b> : Le modèle de cycle de vie en spirale peut être plus coûteux et prendre plus de temps que le modèle en cascade en raison de la nécessité de répéter plusieurs étapes du processus de développement.
<b>Interaction avec les clients</b> : Le modèle de cycle de vie en spirale implique une interaction plus étroite avec les clients et les utilisateurs finaux, ce qui permet de mieux comprendre leurs besoins et de concevoir des logiciels qui répondent à ces besoins.	<b>Nécessite une expertise technique</b> : Le modèle de cycle de vie en spirale nécessite une expertise technique et une connaissance approfondie du processus de développement de logiciel pour être appliqué de manière efficace.
<b>Livraison itérative</b> : Le modèle de cycle de vie en spirale permet de livrer des versions de travail du logiciel à des intervalles réguliers, ce qui permet de recevoir des commentaires réguliers de la part des clients et de s'assurer que le développement est en phase avec les besoins des clients.	<b>Risque de sur-ingénierie</b> : Le modèle de cycle de vie en spirale peut entraîner un risque de sur-ingénierie, c'est-à-dire qu'il peut conduire à la création de fonctionnalités et de processus qui ne sont pas nécessaires pour répondre aux besoins des clients.



### 3. Comparaison entre les deux cycles de vie



# Comparaison entre les cycles de vie

On va procéder à la comparaison sur plusieurs niveau :

## Approche :

Le modèle en cascade est linéaire, ce qui signifie que chaque étape doit être terminée avant de passer à la suivante.

Le modèle en spirale est itératif, ce qui signifie que les étapes se répètent à plusieurs reprises, en boucle.

## Risques :

Le modèle en cascade est plus risqué, car le projet une fois qu'une étape du projet est terminée on ne peut pas la modifier.

Le modèle en spirale permet plus de flexibilité et de possibilités de modifier le projet au fur et à mesure qu'il avance.

## Coût :

Le modèle en cascade est moins coûteux, car chaque étape est terminée avant de passer à la suivante, ce qui permet de mieux planifier les coûts.

Le modèle en spirale est plus coûteux, car il est plus flexible et nécessite une planification plus détaillée à chaque étape.



# Comparaison entre les cycles de vie

## Complexité :

Le modèle en cascade est plus simple et plus facile à comprendre, car chaque étape est clairement définie.

Le modèle en spirale est plus complexe, car il nécessite une planification détaillée et une évaluation continue à chaque étape.

## Adaptabilité :

Le modèle en cascade est moins adaptable, car une fois qu'une étape est terminée, il est difficile de revenir en arrière.

Le modèle en spirale est plus adaptable, car il permet des ajustements en cours de route.



## 4. Types d'applications qui utilisent ces cycles de vie

# Types d'applications

Type d'application	Cycle de vie en cascade	Cycle de vie en spirale
Applications de gestion de bases de données	✓	✓
Applications Financiers	✓	✓
Applications de production de masse	✓	✗
Applications de contrôle de qualité	✓	✗
Applications de développement de jeux	✗	✓
Applications de développement de logiciels de sécurité	✗	✓
Applications de développement de logiciels embarqués	✗	✓
Applications de développement de logiciels à haut risque	✗	✓

# Merci de votre attention

Equipe DevSquad

