

Correction TP1 : Le modèle en Cascade (Waterfall)

Exercice 1 : Élaboration d'un plan de projet en Cascade

1. phases du modèle en cascade

Phase 1 : Analyse des besoins

- **Activités principales :**
 - Réunions avec le client pour comprendre les besoins
 - Analyse des exigences fonctionnelles (gestion livres, membres, emprunts, rapports)
 - Documentation des besoins détaillés
- **Livrables :**
 - Document de spécification des exigences
 - Critères d'acceptation
- **Ressources :**
 - Chef de projet
 - Analyste fonctionnel
- **Durée estimée :** 2 semaines

Phase 2 : Conception

- **Activités principales :**
 - Design de la base de données (tables livres, membres, emprunts)
 - Conception de l'architecture système
 - Conception des interfaces utilisateur
- **Livrables :**
 - Schéma de la base de données
 - Document de conception détaillée
 - Maquettes des interfaces
- **Ressources :**
 - Architecte logiciel
 - Designer UI/UX
- **Durée estimée :** 3 semaines

Phase 3 : Développement

- **Activités principales :**
 - Développement des fonctionnalités
 - Développement des tests unitaires
 - Documentation technique
- **Livrables :**
 - Code source
 - Tests unitaires
 - Documentation
- **Ressources :**
 - Développeurs (2-3)
- **Durée estimée :** 6 semaines

Phase 4 : Tests

- **Activités principales :**
 - Tests fonctionnels
 - Tests d'intégration
 - Tests de performance
 - Tests d'acceptation avec le client
- **Livrables :**
 - Plan de tests
 - Rapports de tests
 - Liste des bugs corrigés
- **Ressources :**
 - Testeurs (QA)
- **Durée estimée :** 2 semaines

Phase 5 : Déploiement

- **Activités principales :**
 - Installation en production
 - Documentation utilisateur
 - Formation des bibliothécaires
- **Livrables :**
 - Application déployée
 - Manuel utilisateur
 - Support technique initial
- **Ressources :**
 - Équipe de déploiement
 - Formateur
- **Durée estimée :** 1 semaine

Phase 6 : Maintenance

- **Activités principales :**
 - Surveillance en production
 - Correction de bugs
 - Mises à jour
- **Livrables :**
 - Rapports de maintenance
 - Liste des bugs corrigés
- **Ressources :**
 - Support technique
- **Durée estimée :** 1 semaine (maintenance continue)

2. Planning séquentiel

Jalons principaux :

- J1 : Validation des spécifications (fin phase 1)
- J2 : Validation de la conception (fin phase 2)
- J3 : Code freeze (fin phase 3)
- J4 : Validation des tests (fin phase 4)
- J5 : Mise en production (fin phase 5)

Points de validation :

- Revue des spécifications avec le client
- Revue de conception technique
- Démonstration des fonctionnalités développées
- Validation des tests d'acceptation
- Formation des utilisateurs

Dépendances :

- La conception ne peut démarrer qu'après validation des spécifications
- Le développement nécessite la validation de la conception
- Les tests requièrent les fonctionnalités développées
- Le déploiement nécessite la validation des tests

Durée totale estimée : 14 semaines

3. Identifier les risques et défis potentiels de l'approche en Cascade**1. Rigidité du modèle :**

- Difficulté à s'adapter aux changements de besoins en cours de projet
- Phases séquentielles sans possibilité de retour en arrière
- Manque de flexibilité pour intégrer de nouvelles fonctionnalités

2. Détection tardive des problèmes :

- Les bugs ne sont découverts que tard dans le cycle de développement
- Coûts élevés des corrections en fin de projet
- Risque d'accumulation de dette technique

3. Implication limitée des parties prenantes :

- Feedback tardif des utilisateurs finaux
- Participation réduite du client après la phase initiale
- Risque de développer un produit ne répondant pas aux attentes

4. Contraintes de planification :

- Difficulté à estimer précisément les délais au début du projet
- Dépendances strictes entre les phases ralentissant le développement
- Impossibilité de livrer des versions intermédiaires fonctionnelles

5. Documentation excessive :

- Temps important consacré à la documentation formelle
- Risque de documentation obsolète si changements
- Surcharge administrative au détriment du développement

6. Risques liés à l'intégration :

- Intégration tardive des composants
- Problèmes d'incompatibilité découverts tardivement
- Difficultés à corriger les problèmes d'architecture