

Mini Projet II: Gestion de projet de développement

Leçon 1: Introduction à la gestion de projet en développement logiciel

Théorie (1h)

- Vue d'ensemble de la gestion de projet et des cycles de vie logiciel (SDLC)
 - Définition et objectifs du SDLC
 - Importance d'une approche structurée
 - Les différentes phases du cycle de vie
- Différences entre les méthodologies classiques (prédictives) et agiles (adaptatives)
- Critères de choix d'une méthodologie selon:
 - La taille et complexité du projet
 - La stabilité des exigences
 - L'expertise de l'équipe
 - Le niveau d'implication du client
 - Les contraintes de temps et budget
 - La gestion des risques
 - Les attentes des parties prenantes

Pratique (1h)

- Étude de cas : Analyser différents scénarios de projets réels et identifier la méthodologie la plus appropriée
- Discussion en groupe : Partage d'expériences sur les défis rencontrés dans différents types de projets

Leçon 2: Méthodologies Classiques - Le modèle en Cascade (Waterfall)

Théorie (1h)

- Historique et principes fondamentaux du modèle en Cascade
- Les phases séquentielles:
 - Étude de faisabilité
 - Analyse et spécification des besoins
 - Conception (architecture et détaillée)
 - Développement et tests unitaires
 - Intégration et tests système
 - Tests alpha
 - Tests bêta
 - Tests d'acceptation
 - Déploiement
 - Maintenance
 - Maintenance corrective
 - Maintenance adaptative

- Maintenance perfective
- Documentation exhaustive et livrables par phase
- Avantages:
 - Structure claire et simple à comprendre
 - Jalons bien définis
 - Documentation complète
 - Idéal pour les projets à exigences stables
 - Renforce les bonnes pratiques
- Inconvénients:
 - Peu flexible aux changements
 - Détection tardive des problèmes
 - Retour client tardif
 - Cycle de développement long
 - Tests tardifs

Pratique (1h)

- Exercice : Élaborer un plan de projet en Cascade pour une application simple
- Atelier : Identifier les risques et défis potentiels de l'approche en Cascade

Leçon 3: Méthodologies Classiques - Le Modèle en V

Théorie (1h)

- Évolution du modèle en Cascade
- Structure du Modèle en V:
 - Branche descendante (développement):
 - Analyse des besoins
 - Spécifications fonctionnelles
 - Architecture technique
 - Conception détaillée
 - Branche ascendante (validation):
 - Tests unitaires
 - Tests d'intégration
 - Tests système
 - Tests d'acceptation
- Importance de la traçabilité entre les phases
- Focus sur la qualité et la validation
- Avantages et inconvénients par rapport au modèle en Cascade

Pratique (1h)

- Exercice : Concevoir un plan de tests aligné avec les phases de développement
- Cas pratique : Application du Modèle en V sur un projet simple

Leçon 4: Méthodologies Agiles - Introduction et Principes

Théorie (1h)

- Vue d'ensemble des méthodologies agiles
- Le Manifeste Agile et ses valeurs fondamentales:
 - Les individus et leurs interactions plus que les processus et les outils

- Un logiciel fonctionnel plus qu'une documentation exhaustive
- La collaboration avec les clients plus que la négociation contractuelle
- L'adaptation au changement plus que le suivi d'un plan
- Les 12 principes agiles
- Comparaison avec les approches traditionnelles

Pratique (1h)

- Discussion de groupe : Analyse des valeurs agiles vs traditionnelles
- Exercice : Identifier les situations où l'agilité apporte le plus de valeur
- Étude de cas : Transformation d'un projet en cascade vers l'agile

Leçon 5: Kanban - Vue d'ensemble et Concepts

Théorie (1h)

- Origines et principes fondamentaux de Kanban
- Concepts clés:
 - Visualisation du flux de travail
 - Limitation du travail en cours (WIP)
 - Gestion du flux
 - Politiques explicites
 - Boucles de feedback
 - Amélioration collaborative et évolution
- Métriques Kanban essentielles

Pratique (1h)

- Création et configuration d'un tableau Kanban
- Exercice : Définition des limites WIP appropriées
- Simulation : Gestion du flux et identification des goulots d'étranglement

Leçon 6: Scrum - Fondamentaux et Framework

Théorie (1h)

- Introduction au framework Scrum
- Les trois piliers empiriques:
 - Transparence
 - Inspection
 - Adaptation
- Valeurs Scrum:
 - Engagement
 - Courage
 - Focus
 - Ouverture
 - Respect
- Les rôles Scrum:
 - Product Owner
 - Scrum Master
 - Developers

Pratique (1h)

- Exercice : Attribution et simulation des rôles Scrum
- Atelier : Création d'un Product Backlog
- Discussion : Responsabilités de chaque rôle

Leçon 7: Scrum - Événements et Artéfacts

Théorie (1h)

- Les événements Scrum:
 - Sprint Planning
 - Daily Scrum
 - Sprint Review
 - Sprint Retrospective
- Les artéfacts Scrum:
 - Product Backlog
 - Sprint Backlog
 - Increment
- Engagements pour chaque artéfact:
 - Product Goal
 - Sprint Goal
 - Definition of Done

Pratique (1h)

- Simulation d'un Sprint complet
- Exercice : Rédaction d'user stories efficaces
- Atelier : Définition d'une Definition of Done

Leçon 8: Scrum en Action

Théorie (1h)

- Patterns d'implémentation courants
- Adaptation de Scrum selon le contexte
- Métriques et KPIs dans Scrum
- Scaled Scrum pour les grandes organisations

Pratique (1h)

- Étude de cas : Scrum dans des entreprises réelles
- Exercice : Résolution de problèmes Scrum courants
- Simulation : Sprint Review et Retrospective

Leçon 9: Comparaison et Intégration des Approches

Théorie (1h)

- Comparaison détaillée : Scrum vs Kanban
- Hybridation des approches (Scrumban)
- Choix de la méthodologie selon le contexte
- Facteurs de succès et pièges à éviter

Pratique (1h)

- Analyse de scénarios : Choix de la meilleure approche
- Discussion : Retours d'expérience et leçons apprises
- Exercice : Création d'un framework hybride

Leçon 10: Projet Final et Évaluation**Théorie (30 min)**

- Synthèse des méthodologies étudiées
- Critères de sélection d'une méthodologie
- Tendances futures dans les méthodes agiles

Pratique (1h30)

- Projet final en équipe
- Présentation et défense des choix méthodologiques
- Évaluation par les pairs et retours d'expérience