

**La Méthode Agile** est une approche itérative et incrémentale de la gestion de projet qui se concentre sur la collaboration et la flexibilité pour livrer des produits de haute qualité dans des délais courts.

*Concepts de gestion de projet*

**Projet** : un projet est une tâche organisée et structurée qui nécessite une planification, une coordination et une gestion appropriées pour réussir

**La Gestion De Projet** est une action temporaire avec un début et une fin, qui mobilise des ressources identifiées (humaines, matérielles, équipements, matières premières, informationnelles et financières) durant sa réalisation

**Ressources** est un élément nécessaire à la réalisation d'une tache ou d'un projet [humaines ,financières ,outil...]

**Livrables** est le résultat Tangibles d’une production Réelle, Appréhendable, Mesurable...

**Charte De Projet** est un document formel, généralement court, qui décrit votre projet dans son intégralité

*Parties prenantes de projet*

Parties prenantes externes		Parties prenantes internes	
Les Clients	Les Fournisseurs	Dirigeants	Managers
Les Utilisateurs	Intermédiaires	Salariés	Actionnaires

*Principaux Rôles dans un projet informatique*

**La maîtrise d'ouvrage (MOA)** : il s'agit du « client » du projet, soit celui qui en Attend des résultats concrets.

**La maitrise d’œuvre (MOE)** : il s'agit du « fournisseur » du projet, soit celui qui réalise l’ouvrage même .

**Le chef de Projet** : est responsable de l’équipe projet en charge de la préparation, de la réalisation et de la finalisation du projet

*Caractéristiques de base d’un projet*

**Objectif /Résultat** : les projets ont des buts et objectifs clairement définis et exposés pour produire de résultat

**Durée/Espace** : les projets se cadencent obligatoirement dans le temps :ils possèdent une date de début et une date de fin ,et se déroulent dans un lieu et contexte spécifiques

**Activités** : chaque projet est inique et nécessite la mise en place d’une organisation et d’un mode de pilotage spécifique

**Ressources** :il est indispensable de bien répartir , les ressources, qu’elles soient humaines, financières ou matérielles, pour qu’un projet puisse aboutir

*Contraintes dans la gestion d’un projet*

Les Contraintes		
Qualité	Couts	Délais

## Méthodes prévisibles (cascades, V, Y)

**M Cascades** : est une approche linéaire et séquentielle des différents phases et activités du projet nécessaires a la livraison du ou des livrables .

EXIGENCES → ANALYSE → CONCEPTION → MISE EN ŒUVRE → VALIDATION → MISE EN SERVICE

**M cycle (Y)** : est une approche de gestion de projet utilisée dans le développement de logiciels et d'autres projets techniques.

**2TUP** : propose un cycle de développement en Y, qui dissocie les aspects techniques de aspects fonctionnels.

<i>Branche fonctionnelle ( \ )</i>	<i>Branche Technique ( / )</i>	<i>Branche Conception Réalisation</i>
<ul style="list-style-type: none"> <li>- Capture des besoins fonctionnels</li> <li>- Analyse</li> </ul>	<ul style="list-style-type: none"> <li>- Capture des besoins techniques</li> <li>- Conception générique</li> </ul>	<ul style="list-style-type: none"> <li>- Conception détaillée</li> <li>- Codage tests Recette</li> </ul>
\$ Conception Préliminaire	\$ Conception Préliminaire	

**M cycle (V)** : est une approche de gestion de projet qui implique la division du processus de développement en deux Phases principales : la phase Spécification et la phase Réalisation.

<i>La phase Spécification ( 1 )</i>	<i>La phase Réalisation ( 3 )</i>
<ul style="list-style-type: none"> <li>\$ Définition des besoins-étude de faisabilité</li> <li>\$Spécification      \$Conception Générale</li> <li>\$Conception détaillée</li> </ul>	<ul style="list-style-type: none"> <li>\$ Recette</li> <li>\$ Validation</li> <li>\$ Tests d'intégration</li> <li>\$ Tests Unitaires</li> </ul>
<i>Réalisation ( 2 )</i>	

**Méthode Kanban** : est une approche de gestion de projet agile qui se concentre sur la visualisation du flux de travail, la limitation du travail en cours et l'amélioration continue.

**Cahier de charges** : est un dossier (ou un simple document), qui définit des éléments ou règles qui doivent être suivi à la lettre dans le cadre de la réalisation d'un projet.

**L'organigramme des taches** : est la décomposition arborescente de l'ensemble des travaux a réaliser dans le cadre du projet

**Matrice des Responsabilités** : est un outil qui vous permet de visualiser les 4 niveaux de responsabilité ( R-A-C-I )

**R**esponsable : Il est le garant de la réalisation • **C**onsulté : Consulté pour la réalisation ou une validation technique

**A**cteur : Il réalise le lot de travaux • **I**nformé : C'est un coéquipier a lui l'on diffuse des documents

**Durée Moyenne de la tache** = ( *Durée Optimiste* + 4\**Durée plus probable* + *Durée Pessimiste* ) / 6

**Diagramme de Gantt** : est un outil de gestion de projet qui permet de visualiser les différentes tâches d'un projet ainsi que leur durée, leur séquence et leur dépendance les unes par rapport aux autres.

**Diagramme de Pert** : est un outil de gestion de projet qui permet de représenter graphiquement les différentes tâches d'un projet

**Scrum** ; est une approche de gestion de projet agile largement utilisée dans le développement de logiciels et de produits complexes.

## Les rôles et Responsabilités ;

**L'équipe de développement** se compose de professionnels qui fournissent un incrément ( FINI ) potentiellement publiable ( Réalisable ) a la fin de chaque Sprint

**Le Scrum Master** ; est charge de promouvoir et supporter Scrum tel que défini dans le guide Scrum

**Le Product Owner** est responsable de maximiser la valeur du produit résultant du travail de l'équipe de développement

## Evénements Scrum ;

**Sprint Planning (Planification des sprints)** ; est la cérémonie que l'équipe scrum réalise en ouverture de sprint pour cadrer le sprint qui démarre

**Sprint** est l'itération ou le cycle de développement du scrum; ce sprint dure en général deux ou trois semaines

**Daily Scrum (mêlée quotidienne)** : est une réunion quotidienne qui rassemble l'ensemble de l'équipe de réalisation qui ne doit pas durer plus de 15 minutes

**Sprint review (revue de Sprint)** a pour objectif de faire un point sur ce qui a été réalisé en cours de sprint

**Sprint rétrospective (Rétrospective Sprint)** a pour objectif de travailler sur deux axes principaux soit l'amélioration continue et la santé de l'équipe

**Artefacts Scrum** sont des éléments permettant de faire fonctionner le cadre de travail Scrum

**Product backlog** : représente le regroupement de l'ensemble des items qui sont pris en charge par l'équipe de développement lors du sprint en cours

**L'incrément** : représente l'ensemble des items qui sont <DONE> dans le sprint en cours ajouté à l'incrément du sprint précédent .

## GIT / GITLAB

**GIT** : est un outil de gestion de version qui permet de stocker un ensemble de fichiers en conservant la chronologie de toutes les modifications qui ont été effectués dessus.

### Glossaire Git ;

**Dépôt (repository)** : espace de stockage public géré par le logiciel de version .

**Gestion de version** :

**Copie locale (clone)** : copie d'une version en local

**Lecture (pull)** : téléchargement d'un fichier situé sur le dépôt

**Écriture (push)** : soumettre les modifications sur le dépôt

**Suivi d'état (status)** : suivre les modifications sur les fichiers

**Créer de version (commit)** : créer des nouvelles versions

**Branche (branch)** : une "copie" de votre projet pour développer et tester de nouvelles fonctionnalités sans impacter le projet de base .

**La zone de travail** : les états de fichiers sont liés à des zones de travail dans GIT . En fonction de son état, un fichier va pouvoir apparaître dans telle ou telle zone de travail .

### Les avantages de SonarQube :

S'intègre facilement dans les pipelines CI / CD

Donne la possibilité de générer plusieurs rapports de la qualité de logiciel

Peut également être intégré au cycle de construction Maven et Gradle

Vérifie presque tout : qualité du code, formatage déclarations de variables, ...

## Configuration des outils ;

- git config—global user.name “Votre nom”
- git config—global user. Email “Votre email”

Créer un nouveau dépôt GIT ; <b>git init</b>
Obtenir un depuis une URL existante ; <b>git clone</b> <a href="https://git@github.com:nom/chemin/vers/dépôt">git@github.com:nom/chemin/vers/dépôt</a>
Ajouter le fichier index.html a l'index : <b>git add index.html</b>
Ajouter tous les fichiers modifiés : <b>git status</b>
Valider les modifications apportées au HEAD : <b>git commit -m "Description du commit"</b>
Créer une branche ; <b>git branch &lt;nom-branche&gt;</b>
Supprimer une branche ; <b>git branch -d &lt;nom-branche&gt;</b>
Passer a une autre branche ; <b>git checkout&lt;nom-branche&gt;</b>
Fusionner une branche dans la branche active ; <b>git merge &lt;nom-branche&gt;</b>

**GITLAB** : est un outil de gestion de code source et de collaboration qui facilite la mise en œuvre des principes agiles.

## Fonctionnalités ;

**Système de suivi des bugs** : est logiciel qui permet d'effectuer un suivi des bugs signalés dans le cadre d'un projet de développement de logiciel .

**L'intégration continue ( CI )** : est un ensemble de pratiques utilisées en génie logiciel consistant a vérifier a chaque modification de code source que le résultat des modifications ne produit pas de régression dans l'application développée .

**La livraison continue ( CD )** : est une approche d'ingénierie logicielle dans laquelle les équipes produisent des logiciels dans des cycles courts , ce qui permet de le mettre a disposition a n'importe quel moment .

**Un wiki** : est une application web qui permet la création , la modification et l'illustration collaboratives de pages à l'intérieur d'un site web .

**Métrique logicielle** ; mesure d'une propriété d'un logiciel , c'est une approche quantité : extraire une mesure de qualité d'un logiciel a partir de l'analyse statistique du code source .

**NDepend** : est un outil de contrôle de qualité logicielle pour les développements .NET

**Testwell CMT++** : est un outil de mesure de complexité pour C et C++

**Testwell CMTjava** : est un outil de mesure de complexité pour java

**SonarQube** ; est un outil de contrôle de qualité logicielle pour les développements dans plus 25 langages dont JAVA, C#, PHP, Python ...

### Les étapes d'installation SonarQube :

- Étapes 1 : installer JDK 11
- Étapes 2 : installer la base de données PostgreSQL 13 et PgAdmin 4
- Étapes 3 : Créer un utilisateur sonar/sonar et une base de données SonarQube
- Étapes 4 : installer serveur SonarQube et SonarScanner
- Étapes 5 : Configurer votre serveur SonarQube
- Étapes 6 : Lancer serveur SonarQube et se connecter via le compte admin/admin