

Direction Régionale de Casa Sottat

EVALUATION DE FIN DE MODULE
AU TITRE DE L'ANNEE : 2023/2024

V1

Filière : DIA_DEVOWFS_TS

Niveau : TS

N° du module : M202

Intitulé du module : Approche Agile

Date d'évaluation : 19/01/2024

Année de formation : 2A

Epreuve : TH/PR

Durée : 2H30

Coefficient : 3

Barème/40

Partie 1 : (15pts)

Exercice 1 : QCM (6pts)

1. Définir la charte d'un projet (1pt) :

- a. Un diagramme de Gantt
- b. Un document formel énonçant les objectifs, la portée, les parties prenantes, les ressources et les contraintes d'un projet.
- c. Un outil de gestion de configuration

2. Citer les méthodes prévisibles de gestion de projet (1pt)

- a. Scrum
- b. Modèle en cascade (Waterfall)
- c. Cycle en V

3. Définir le manifeste Agile (1pt)

- a. Un document juridique
- b. Un ensemble de valeurs et de principes pour guider le développement logiciel agile
- c. Un diagramme de flux de travail

4. Définir DevOps ? (1pt)

- a. Une méthodologie de gestion de projet
- b. Une approche visant à améliorer la collaboration entre les équipes de développement (Dev) et d'exploitation (Ops), automatisant les processus de livraison logicielle.
- c. Un langage de programmation

5. Définir le diagramme de PERT ? (1pt)

- a. Un diagramme circulaire représentant les processus d'une entreprise
- b. Un outil de gestion de projet utilisé pour représenter graphiquement les tâches et les dépendances d'un projet, mettant l'accent sur les relations entre les activités.
- c. Un protocole de communication réseau

6. Donner des exemples de métriques de SonarQube ? (1pt)

- a. Lignes de code écrites par jour
- b. Nombre de bugs détectés
- c. Couverture de code

Exercice 2 : Agile Scrum (9pts)

1. En se basant sur les fonctionnalités décrites, extraire 5 user stories (5pts) :
Les stagiaires d'un établissement de formation, ont l'opportunité de développer une application Web baptisée "EventDEV". Cette application permettra aux utilisateurs d'organiser, et de participer à des événements. Chaque événement peut avoir des participants, un lieu, une date et une description.
2. Définir le rôle du Scrum Master (1pt) ?
3. Donner une définition d'un sprint (1pt) ?
4. Définir le Daily Scrum (1pt) ?
5. Quels sont les différents types de tickets utilisés en Jira ? (1pt)



Partie 2 : (25pts)

Exercice 1 : Planification d'un projet (10pts)

Un projet peut être décomposé en 10 tâches, dans le tableau ci-dessous, on indique pour chaque tâche, sa durée et les tâches immédiatement antérieures :

Tache	A	B	C	D	E	F	G	H	I	J
Antériorité	-	-	-	A	B	C	D	A, E	B, F	G, H, I
Durée	10	8	5	5	3	4	2	9	7	5

1. Tracer le graphe PERT, et calculer les dates au plus tôt et au plus tard pour chaque tâche (5pts)
2. Calculer les marges libres et les marges totales des tâches B et I (2pts)
3. Déterminer le chemin critique (1pt)
4. Déduire la durée du projet (1pt)
5. Si la durée de la tâche E devient 6 au lieu de 3, quelle conséquence cela aurait-il sur la durée du projet ? (1pt)

Exercice 2 : Git/Gitlab (10 pts – 1pt pour chaque question)

Nous avons créé un dépôt distant sur GitLab. Ecrire les commandes git pour :

1. Cloner le dépôt distant sur votre ordinateur local.
2. Créer un fichier "test.txt" sur votre ordinateur, et ajoutez-y du contenu.
3. Ajouter le fichier à l'index.
4. Enregistrer les modifications dans l'historique de votre dépôt en créant un commit.
5. Envoyer les commits vers le dépôt distant.
6. Récupérer les dernières modifications du dépôt distant.

7. Créer une nouvelle branche « new-branch » à partir de la branche courante. Basculer vers cette nouvelle branche.
8. Afficher l'historique des commits.
9. Afficher les différences entre les fichiers dans votre dépôt local et le dernier commit.
10. Annuler des commits.

Exercice 3 : DevOps/CI-CD/Gitlab-CI (5pts)

1. Déclarer un pipeline CI/CD Gitlab dans un repository en ajoutant le fichier « .gitlab-ci.yml » selon les contraintes suivantes: (5pts)

Ce fichier de configuration doit définir trois étapes : build, test et deploy. Chacune de ces étapes doit contenir un ou plusieurs jobs, et ces jobs doivent être exécutés en utilisant des commandes spécifiées dans la section "script".

En particulier, dans la phase de build, trois jobs supplémentaires doivent être ajoutés : "build_1", "build_2" et "build_3". Ces jobs doivent être similaires et doivent utiliser la commande "echo" pour afficher des messages dans la console, indiquant le début et la fin de chaque étape de build.

Les autres jobs des étapes test et deploy doivent également utiliser la commande "echo" pour afficher des messages indiquant le début et la fin de chaque étape respective.