|  |  |
| --- | --- |
| ***OFPPT*** |  |
| **Office de la Formation Professionnelle**  **et de la Promotion du Travail** |

**Contrôle Continu 3**

**Module :** Adopter l'approche Agile

|  |  |
| --- | --- |
| **Filière : DD 2ème année Année de formation : 2022/2023** | ***Epreuve : Pratique*** |
| **Niveau : TS** | **Variante : 3** |
| **Durée : 2h00** | **Barème : 20 Pts** |

NB: N'oubliez pas de mentionner votre nom et prénom dans le fichier et de renommer le fichier avec votre **nom\_prénom.docx**

**Nom Prénom :**

**Partie 1 : Gestion d'un projet agile sur JIRA (12 points)**

Remarque importante: Utilisez votre ordinateur et votre compte JIRA pour répondre aux question suivantes, collér après chaque question le(s) capture(s) d'écran(s) de votre travail.

On souhaite gérer un projet informatique pour une solution de gestion de bilbiothèque qui contient un application web et une application mobile en utilisant la méthode SCRUM, et on a opter pour la plateforme JIRA pour gérer notre projet. Le projet Contient les spécifications suivantes :

1. Créer un pojet SCRUM géré par l'entreprise qui porte votre nom. **(1pt)**
2. Le projet est composé de deux Composantes : **(1 pt)**
   1. Application Web
   2. Application mobile
3. Le projet contient deux versions (1.0.0 et 1.1.0 ) **(1pt)**
4. Créez trois Epics **(1pt)**
   1. Gestion des ouvrages
   2. Gestion des abonnés
   3. Gestion des emprunts
5. Créez les User Stories suivantes et liez chacune à son Epic. **(3 pts)**
   1. En tant que bibliothécaire je dois avoir la possibilité de chercher un ouvrage par thème, titre et par prix.
   2. En tant que bibliothécaire je dois avoir la possibilité d'ajouter un nouvel ouvrage par thème.
   3. En tant que bibliothécaire je dois être capable d'imprimer l'inventaire à tout moment.
   4. En tant que bibliothécaire je souhaite avoir la possibilité de gérer les comptes des abonnés.
   5. En tant que bibliothécaire je souhaite visualiser la liste des ouvrages en possession de chaque abonné.
   6. En tant qu'abonné, je dois être capable de chercher les ouvrages par titre, par thème et par auteur.
   7. En tant qu'abonné, je doit être capable de localiser un ouvrage, et entrer en communication avec l'abonné qui l'emprunt actuellement et passer une réservation de l'ouvrage.
   8. En tant qu'abonné, je doit être capable de voir l'historique de mes emprunts.
6. Associez la première User Storie à un score (Story points) de votre choix et liez là à une version et une composante. **(1 pt)**
7. Créez un sprint de deux semaines et démarrez-le avec trois user stories de votre choix. **(1 pt)**
8. Déplacer les user stroies de ce sprint à la case "Done" et finalisez votre sprint. **(1 pt)**
9. Affichez votre Feuille de route. **(1 pt)**
10. Affichez le Burn Down du projet. **(1 pt)**

**Partie 2 : Gestion de version avec GIT (8 points)**

Remarque importante : Utilisez votre ordinateur pour répondre aux questions suivantes, coller après chaque question **toutes les commandes** nécessaires à la manipulation.

1. Créez un dossier sur votre ordinateur qui porte votre nom de famille suivi du mot "\_distant" exemple "benhsain\_distant", entrer dans ce dossier et initialisez le comme **repository GIT Distant**. **(1 pt)**
2. Créez un autre dossier qui porte votre nom suivi du mot "\_local" exemple "benhsain\_local" puis associez le au dossier disant. **(1 pt)**
3. Ouvrez le dossier local en utilisant visual studio code (ou autre editeur de votre choix) créez trois fichiers vides (produit.html, style.css, script.js), Trackez ces trois fichiers et commitez la version git. **(1 pt)**
4. Créez une branche "Gestion produits", basculez sur cette branche, ajoutez un contenu a vos trois fichiers, commitez la version, revenez à la branche master. **(1pt)**
5. Affichez log sous forme d'une seule ligne, affichez les informations du depot origin puis pushez votre travail vers le repository distant. **(1pt)**
6. Le push vous demande de tapez origin master à la fin, faites de tel sorte que vous pouvez pushez directement dans le futur. **(1pt)**
7. Commancez une nouvelle modification sur le fichier produit.html (ajoutez trois images), faites-le passez en mode stash, faites une nouvelle modification (ajoutez trois divs) et faites votre commit+push. Finalisez votre travail qui se trouve en stash et faites votre commit+push. **(1 pts)**
8. En souhaite effacer l'historique des commits effectuer depuis le premier commit sans perdre les modifications dans les fichiers et en gardant les fichiers en mode stage. Proposez une solution **(1pt)**