Manuel d'installation pour un nouveau collaborateur

# Objectif du manuel

Ce manuel détaillé guidera un nouveau collaborateur à travers le processus d'installation, de configuration et d'utilisation du système de gestion des questionnaires et de leur expédition, en suivant les étapes nécessaires pour configurer son environnement de travail sur un PC Windows.

# 1. Prérequis

Avant de commencer, assurez-vous d'avoir les éléments suivants :  
- Un compte GitHub actif.  
- Un PC Windows avec les droits d'administration.  
- Une connexion Internet stable.

# 2. Installation des outils nécessaires

## Étape 1 : Installer Git et GitHub Desktop

Git est un système de contrôle de version nécessaire pour gérer le code, et GitHub Desktop est une interface graphique pour Git.  
1. Télécharger Git pour Windows :  
- Allez sur https://git-scm.com/downloads.  
- Téléchargez le fichier d'installation pour Windows et lancez-le.  
- Suivez les instructions d'installation, en laissant les options par défaut, sauf si vous avez des préférences particulières.  
2. Télécharger et installer GitHub Desktop :  
- Allez sur https://desktop.github.com.  
- Téléchargez et installez GitHub Desktop.  
- Lors de l'installation, connectez-vous avec votre compte GitHub.

## Étape 2 : Installer Node.js

Node.js est nécessaire pour exécuter l'application et gérer les dépendances.  
1. Télécharger Node.js :  
- Allez sur https://nodejs.org/en/download.  
- Téléchargez la version LTS (Long Term Support).  
- Lancez le fichier d'installation et suivez les instructions pour installer Node.js et npm (le gestionnaire de paquets).

## Étape 3 : Installer un éditeur de texte

Un éditeur de texte est nécessaire pour modifier le code source et gérer le projet.  
1. Télécharger Visual Studio Code :  
- Allez sur https://code.visualstudio.com.  
- Téléchargez et installez Visual Studio Code, un éditeur puissant pour le développement.

## Étape 4 : Vérification de l'installation

Ouvrez une invite de commande et vérifiez que Node.js et Git sont correctement installés :  
git --version  
node --version  
npm --version  
Si les versions s'affichent correctement, vous êtes prêt à continuer.

# 3. Cloner le dépôt GitHub

Vous devrez maintenant cloner le dépôt de l'application pour commencer à travailler.  
1. Cloner le dépôt avec GitHub Desktop :  
- Ouvrez GitHub Desktop.  
- Cliquez sur 'File' > 'Clone Repository'.  
- Entrez l'URL du dépôt GitHub et choisissez un emplacement sur votre disque pour le cloner.  
- Cliquez sur 'Clone'.  
2. Vérification du clonage :  
- Une fois le clonage terminé, ouvrez le dossier dans Visual Studio Code pour vérifier que tous les fichiers sont présents.

# 4. Installer les dépendances

L'application dépend de plusieurs bibliothèques externes. Vous devrez installer ces dépendances à l'aide de npm.  
1. Ouvrir l'invite de commande :  
- Naviguez vers le répertoire du projet cloné.  
- Exécutez la commande suivante pour installer toutes les dépendances :  
npm install

# 5. Lancer le serveur de développement

Pour tester que l'application fonctionne correctement, vous devez démarrer le serveur local.  
1. Lancer l'application :  
- Dans l'invite de commande, exécutez la commande suivante pour démarrer le serveur :  
npm run dev  
- Le serveur démarre et l'application devrait être accessible sur http://localhost:3000.  
2. Accéder à l'application :  
- Ouvrez votre navigateur et allez à http://localhost:3000 pour voir l'application en action.

# 6. Utilisation du cockpit et des questionnaires

Une fois l'application en place, voici les principales fonctionnalités que le collaborateur devra maîtriser :  
1. Gestion des questionnaires :  
- L'utilisateur peut créer, modifier et supprimer des questionnaires à partir de l'onglet dédié.  
- Des options d'édition sont disponibles pour ajouter des questions, modifier des réponses et gérer la logique des questionnaires.  
2. Expédition des questionnaires :  
- Une fois le questionnaire finalisé, il peut être expédié via l'interface en ligne ou être intégré à un projet externe.  
- Le collaborateur devra veiller à ce que toutes les informations soient correctement vérifiées avant l'expédition.  
3. Suivi des résultats :  
- Des statistiques détaillées peuvent être consultées pour analyser les réponses et les retours sur les questionnaires envoyés.

# 7. Gestion des versions et synchronisation avec GitHub

Lorsqu'un collaborateur effectue des modifications sur le code ou les fichiers de l'application, il doit les enregistrer et les pousser sur GitHub.  
1. Faire un commit :  
- Ouvrez GitHub Desktop.  
- Saisissez un message de commit dans la section prévue à cet effet.  
- Cliquez sur 'Commit to main' pour enregistrer vos modifications localement.  
2. Pousser les changements sur GitHub :  
- Cliquez sur 'Push origin' dans GitHub Desktop pour envoyer vos modifications sur le dépôt GitHub.

# 8. Gestion des branches

Si des modifications importantes sont apportées, il peut être nécessaire de travailler sur des branches séparées pour éviter les conflits.  
1. Créer une nouvelle branche :  
- Cliquez sur 'Branch' > 'New branch'.  
- Donnez un nom à la branche et cliquez sur 'Create branch'.  
- Faites vos modifications sur cette branche.  
2. Fusionner une branche :  
- Une fois les modifications terminées, vous pouvez fusionner la branche avec la branche principale ('main').  
- Effectuez un 'Pull request' sur GitHub pour fusionner les modifications.

# 9. Dépannage et questions fréquentes

En cas de problème, voici quelques solutions possibles :  
- Le serveur ne démarre pas : Vérifiez que toutes les dépendances sont installées avec 'npm install'.  
- Problèmes de synchronisation Git : Assurez-vous que vous avez bien configuré votre identifiant GitHub et que votre branche est correctement synchronisée.  
- Problèmes d'affichage de l'application : Essayez de vider le cache de votre navigateur ou de redémarrer le serveur avec 'npm run dev'.

# 10. Conclusion

En suivant ce guide, le collaborateur devrait être en mesure de configurer son environnement, cloner le dépôt, installer les dépendances, et utiliser l'application pour gérer et expédier des questionnaires. Si des problèmes surviennent, ils peuvent être résolus en vérifiant les étapes d'installation ou en consultant la section de dépannage.

**Manuel d'installation pour un nouveau collaborateur**

**Objectif du manuel**

Ce manuel détaillé guidera un nouveau collaborateur à travers le processus d'installation, de configuration et d'utilisation du système de gestion des questionnaires et de leur expédition, en suivant les étapes nécessaires pour configurer son environnement de travail sur un PC Windows.

**1. Prérequis**

Avant de commencer, assurez-vous d'avoir les éléments suivants :

* Un compte GitHub actif.
* Un PC Windows avec les droits d'administration.
* Une connexion Internet stable.

**2. Installation des outils nécessaires**

Voici les étapes pour installer tous les outils dont le collaborateur aura besoin pour travailler efficacement.

**Étape 1 : Installer Git et GitHub Desktop**

Git est un système de contrôle de version nécessaire pour gérer le code, et GitHub Desktop est une interface graphique pour Git.

1. **Télécharger Git pour Windows** :
   * Allez sur <https://git-scm.com/downloads>.
   * Téléchargez le fichier d'installation pour Windows et lancez-le.
   * Suivez les instructions d'installation, en laissant les options par défaut, sauf si vous avez des préférences particulières.
2. **Télécharger et installer GitHub Desktop** :
   * Allez sur <https://desktop.github.com>.
   * Téléchargez et installez GitHub Desktop.
   * Lors de l'installation, connectez-vous avec votre compte GitHub.

**Étape 2 : Installer Node.js**

Node.js est nécessaire pour exécuter l'application et gérer les dépendances.

1. **Télécharger Node.js** :
   * Allez sur https://nodejs.org/en/download.
   * Téléchargez la version LTS (Long Term Support).
   * Lancez le fichier d'installation et suivez les instructions pour installer Node.js et npm (le gestionnaire de paquets).

**Étape 3 : Installer un éditeur de texte**

Un éditeur de texte est nécessaire pour modifier le code source et gérer le projet.

1. **Télécharger Visual Studio Code** :
   * Allez sur <https://code.visualstudio.com>.
   * Téléchargez et installez Visual Studio Code, un éditeur puissant pour le développement.

**Étape 4 : Vérification de l'installation**

Ouvrez une invite de commande et vérifiez que Node.js et Git sont correctement installés : git --version

node --version

npm –version

Si les versions s'affichent correctement, vous êtes prêt à continuer.

### ****3. Cloner le dépôt GitHub****

Vous devrez maintenant cloner le dépôt de l'application pour commencer à travailler.

1. **Cloner le dépôt avec GitHub Desktop** :
   * Ouvrez GitHub Desktop.
   * Cliquez sur "File" > "Clone Repository".
   * Entrez l'URL du dépôt GitHub et choisissez un emplacement sur votre disque pour le cloner.
   * Cliquez sur "Clone".
2. **Vérification du clonage** :
   * Une fois le clonage terminé, ouvrez le dossier dans Visual Studio Code pour vérifier que tous les fichiers sont présents.

### ****4. Installer les dépendances****

L'application dépend de plusieurs bibliothèques externes. Vous devrez installer ces dépendances à l'aide de npm.

1. **Ouvrir l'invite de commande** :
   * Naviguez vers le répertoire du projet cloné.
   * Exécutez la commande suivante pour installer toutes les dépendances :

npm run dev

* + Le serveur démarre et l'application devrait être accessible sur http://localhost:3000.

1. **Accéder à l'application** :
   * Ouvrez votre navigateur et allez à http://localhost:3000 pour voir l'application en action.

**6. Utilisation du cockpit et des questionnaires**

**Onglets principaux du cockpit :**

Une fois l'application en place, voici les principales fonctionnalités que le collaborateur devra maîtriser :

1. **Gestion des questionnaires** :
   * L'utilisateur peut créer, modifier et supprimer des questionnaires à partir de l'onglet dédié.
   * Des options d'édition sont disponibles pour ajouter des questions, modifier des réponses et gérer la logique des questionnaires.
2. **Expédition des questionnaires** :
   * Une fois le questionnaire finalisé, il peut être expédié via l'interface en ligne ou être intégré à un projet externe.
   * Le collaborateur devra veiller à ce que toutes les informations soient correctement vérifiées avant l'expédition.
3. **Suivi des résultats** :
   * Des statistiques détaillées peuvent être consultées pour analyser les réponses et les retours sur les questionnaires envoyés.

**7. Gestion des versions et synchronisation avec GitHub**

**Faire des commits et pousser les modifications :**

Lorsqu'un collaborateur effectue des modifications sur le code ou les fichiers de l'application, il doit les enregistrer et les pousser sur GitHub.

1. **Faire un commit** :
   * Ouvrez GitHub Desktop.
   * Saisissez un message de commit dans la section prévue à cet effet.
   * Cliquez sur "Commit to main" pour enregistrer vos modifications localement.
2. **Pousser les changements sur GitHub** :
   * Cliquez sur "Push origin" dans GitHub Desktop pour envoyer vos modifications sur le dépôt GitHub.

**Gestion des branches :**

Si des modifications importantes sont apportées, il peut être nécessaire de travailler sur des branches séparées pour éviter les conflits :

1. **Créer une nouvelle branche** :
   * Cliquez sur "Branch" > "New branch".
   * Donnez un nom à la branche et cliquez sur "Create branch".
   * Faites vos modifications sur cette branche.
2. **Fusionner une branche** :
   * Une fois les modifications terminées, vous pouvez fusionner la branche avec la branche principale (main).
   * Effectuez un "Pull request" sur GitHub pour fusionner les modifications.

**8. Dépannage et questions fréquentes**

En cas de problème, voici quelques solutions possibles :

* **Le serveur ne démarre pas** : Vérifiez que toutes les dépendances sont installées avec npm install.
* **Problèmes de synchronisation Git** : Assurez-vous que vous avez bien configuré votre identifiant GitHub et que votre branche est correctement synchronisée.
* **Problèmes d'affichage de l'application** : Essayez de vider le cache de votre navigateur ou de redémarrer le serveur avec npm run dev.