

cahier de besoin pdp

2023

# 1 Généralité sur les besoins

## 1.1 Besoins fonctionnels:

1. Le formulaire doit permettre de créer des questions de différents types : choix multiple, choix unique, case à cocher, zone de texte, etc.
2. Gestion des questions: le système doit permettre la création, la modification et la suppression de questions dans le formulaire.
3. Gestion des réponses: le système doit enregistrer les réponses des utilisateurs de manière efficace et sécurisée.
4. Présentation des questions: le système doit permettre la personnalisation de l'apparence du formulaire, incluant la mise en forme des questions et des choix de réponse.
5. Ordre de préférence: le système doit permettre aux utilisateurs de définir un ordre de préférence entre différentes options.
6. Envoi automatique d'email: le système doit envoyer un email personnalisé à chaque personne qui remplit le formulaire.
7. Import/Export des données: le système doit permettre l'import et l'export des données de formulaire dans un format texte ou JSON.

## 1.2 Besoins non fonctionnels:

1. Le formulaire doit avoir une interface utilisateur conviviale et facile à utiliser pour les répondants.
2. Performance: le système doit être rapide et fonctionner de manière fluide, même en cas de grand nombre de réponses au formulaire.
3. Sécurité: le système doit garantir la confidentialité des données des utilisateurs et n'être accessible qu'à l'administrateur du formulaire.
4. Fiabilité: Le formulaire doit être rapide et efficace pour améliorer l'expérience utilisateur et éviter les temps d'attente inutiles.
5. Accessibilité: le système doit être facile à utiliser et accessible à un large public. Le formulaire doit être conforme aux normes d'accessibilité pour permettre aux personnes atteintes de handicaps par exemple de remplir le formulaire.
6. Conformité aux normes: le système doit respecter les normes en vigueur en matière de sécurité et de protection des données.

## **2 Besoins fonctionnelle et exigence technique Backend**

### **2.1 Exigences fonctionnelles pour le backend :**

1. Créer un modèle de données pour stocker les informations de formulaire et les réponses de formulaire.
2. Définir les API REST pour les opérations CRUD sur les formulaires et les réponses.
3. Implémenter un mécanisme d'authentification pour protéger les données de formulaire.
4. Mettre en place un mécanisme d'envoi de courrier électronique personnalisé pour les réponses de formulaire.
5. Créer un outil d'exportation pour permettre l'exportation des formulaires et des réponses au format texte ou JSON.

### **2.2 Exigences techniques pour le backend :**

1. Utiliser le framework .... pour le développement du backend.
2. Utiliser une base de données relationnelle comme MySQL ou PostgreSQL pour stocker les données du formulaire.
3. Implémenter des tests unitaires pour chaque API REST.
4. Mettre en place des journaux pour faciliter le débogage.
5. Utiliser un système de gestion de version comme Git pour le contrôle de version.

### **3 Besoins fonctionnelle et exigence technique frontend**

#### **3.1 Exigences fonctionnelles pour le frontend :**

1. Créer une interface utilisateur pour la création, la visualisation et la modification de formulaires.
2. Permettre la personnalisation de l'apparence du formulaire par des développeurs front-end.
3. Afficher les réponses de formulaire sous forme de graphiques ou de tableaux.
4. Permettre aux utilisateurs de classer les choix par ordre de préférence.

#### **3.2 Exigences techniques pour le frontend :**

1. Utiliser le framework Quasar pour le développement du frontend.
2. Utiliser HTML, CSS et JavaScript pour la création de l'interface utilisateur.
3. Utiliser des bibliothèques graphiques comme Chart.js pour l'affichage des données de formulaire sous forme de graphiques.
4. Utiliser un système de gestion de version comme Git pour le contrôle de version.