

Compte-rendu 8 : réunion du 10/12/2024

SIForms : Application de saisie de formulaires

Auteurs :
Adèle DESMAZIERES
Léa EL ABOUD
Kevin XU

Destinataires :
Paul GUYOT
Julien KERLIDOU
Cédric BESSE
Lamia LARAQUI

Responsable : **Adèle DESMAZIERES**

Type : **Compte-rendu 8 : réunion du
10/12/2024**

Date : **10/12/2024**

Nombre de pages : **2**

Statut : **Final**

Adresse : typst.app/PISTL

Version	Date	Modification	Rédacteur
v.0.1	10/12/2024	Rédaction du CR	Adèle DESMAZIERES

Ordre du jour

1. Revue de la STR	1
1.1. Objets métiers	1
1.2. Services	1
1.3. Versions des technologies	1
1.4. Diagramme de séquences	1
2. Revue du code du backend	2
3. Maquette Figma	2
4. Premier accès à l'application	2
5. Plannification	2

1. Revue de la STR

1.1. Objets métiers

- **Form et Entry** : regrouper les objets Entry et Form ensemble, car ils ont une relation 1-1 pour le moment
- **Pré-remplissage** : réflexion à faire sur la différence entre une entrée pré-remplie et une entrée validée. Par exemple, ajouter un attribut qui indique si l'entrée est pré-remplie ou finale.
- **Model** : indiquer que le couple (name, version) est unique
- **Session** : reste active jusqu'à 30 jours après la dernière activité, puis déconnecte l'utilisateur. Peut être géré avec le JWT (JSON Web Token)
- **Document** : L'id du document est bien présent dans le JSON de réponse. Integer de l'app pas forcément identique à celui du serveur. Id de document et id de Form. Côté serveur, la clef conserve l'id de document de l'application mobile. Les id des entrées et documents sont diff entre app et serv. Quelle est la clef du Document côté serveur ?
- **Document.path** : Uniquement dans l'application pour localiser le chemin du document. Dans le serv on aura un BLOB (Binary Large Object)

1.2. Services

- **Suppression de compte** : pas utile pour les professionnels.
- **Suppression de réponse** : possible que dans le frontend, pour permettre à un utilisateur de libérer la mémoire de son appareil mobile
- **Use case de consultation** : que pour les formulaires envoyés, pas les brouillons. Ceux-ci ne peuvent qu'être modifiés.
- **Protocole d'échange** : celui décrit correspond au protocole TCP avec une fenêtre. Connexion en HTTPS. Question : est-ce séquentiel (serveur répond à la fin de la requête), ou parallèle ? Pas besoin de répondre avec ACK car HTTPS renvoie déjà un ACK. Juste envoyer une photo en entier avec POST, attendre le résultat. Conclusion : protocole pas en Go-Back-N ARQ mais seulement Stop-And-Wait ARQ.

1.3. Versions des technologies

Recommandations des professionnels :

- Django LTS 4.2
- PostgreSQL 16 ou 17

1.4. Diagramme de séquences

Gérer d'autres erreurs du serveur, pas seulement le Timeout.

- Timeout du serveur : l'application renvoie les données.
- Autres erreurs du serveur (ex. : erreur réseau) : l'application renvoie les données.

2. Revue du code du backend

Les classes `Model.model` du serveur Django doivent correspondre à notre STR, et correspondront aux données de la base PostgreSQL. Il faut refaire correspondre `Form` à `Model` et `Response` à `Entry`, avec les mêmes attributs que dans notre diagramme de classes.

Données du serveur :

- **données des modèles** : stockées directement dans Postgre en JSON, et pourront être modifiées grâce au système Django admin
- **données des documents photo** : stockées soit directement dans base de données, soit en fichiers à côté de la base, peu importe.

3. Maquette Figma

D'habitude les dev utilisent un système qui communique avec les UX designers pour simplifier leur développement, mais pour nous pas besoin de repasser sur la maquette car nous n'exigeons pas que le UI/UX soit pixel perfect. Notre maquette sert surtout à voir les workflows, etc, inutile de la reprendre maintenant. Donc notre maquette a servi seulement pour la confection, ce n'est pas un livrable, même si dans d'autres projets, c'est une interface de communication entre designer et développeur.

Par contre, utile d'avoir un design system (voir lib équivalent de `material UI` en `ReactNative`). Une librairie d'UI a des design system (cases à cocher, ...) mais aussi du fonctionnel (comme des champs intelligents).

4. Premier accès à l'application

Le premier accès à l'application par les techniciens se fait grâce à un lien envoyé à l'équipe projet. La personne chargée de l'intervention technique télécharge l'app, crée un compte, puis accès à un formulaire pré-rempli.

5. Plannification

Prochaine réunion en visio le vendredi 20/12/2025 à 15h. D'ici là, corriger STR, corriger le backend et démarrer le frontend.