
Compte rendu de Réunion

Compte-Rendu
N°002
05/09/2025

Première réunion avec le client : BAJON Consulting

Appel à l'ordre

Date de la réunion : 05/09/2025

Heure de début : 9h30

Lieu : IUT Blagnac B103

Réunion organisée par : Bilon, Client
Type de Réunion : Contact avec client
Animateur : Bilon
Rédacteur : Simon
Maître du temps : Pierre

Participants	Absents
<ul style="list-style-type: none">- Simon (Scrum Master)- Bilon- Titouan- Robin- Pierre	//

Précédente réunion

Date et heure : 03/09/2025 à 10h

Synthèse : Parler de la mise en place de l'environnement de travail, du contact du client ainsi que le commencement des livrables attendus (recueil de besoins).

Objectifs de cette réunion

Faire une réunion en visio avec les membres du projets et le client pour pouvoir lui poser des questions sur ce qu'il attend, ce que nous avons pas compris ainsi que pour choisir les technologies de développement que l'on utilisera ensuite.

Ordre du jour

Sujet à aborder	Description	Présentateur	Temps imparti
- L'attendu	Voir avec le client de ce à quoi devrait ressembler l'application finale.	Groupe et client	30 min
- Technologies de développement	Choisir les meilleures technologies pour le projet.	Groupe et client	30 min
- Synthèse de la réunion avec le client et organisation	Faire un point de la réunion avec l'équipe et commencer à répartir les tâches.	Groupe	15 min

Attendus

Ordre du jour n°2 : Organisation du travail			
Éléments d'actions	Responsable	Date d'échéance	Résultat attendu
Sujet 1 : L'attendu	Bilon	05/09/2025	Comprendre à quoi devrait ressembler le site final.
Sujet 2 : Technologies de développement	Robin	05/09/2025	Savoir quelles technologies de développement l'on utilisera pour ce projet.
Sujet 3 : Synthèse de la réunion avec le client et organisation	Groupe	05/09/2025	Faire un point avec le groupe sur le travail attendu puis répartir les tâches et commencer le projet.
Remarque et commentaire	//	//	//

Résultats

Ordre du jour n°2 : Organisation du travail

Sujets	Résultats
Sujet 1 : L'attendu	<p>1. Objectifs du projet :</p> <ul style="list-style-type: none"> - Proposer plusieurs offres a commencé par l'offre 'essentiel'. - Créer une application indépendante du site existant pour les audits. - Mettre en place un outil de formulaire pour automatiser les processus d'audit. - Offrir une solution hautement paramétrable et évolutive. <p>2. Fonctionnement de l'audit :</p> <ul style="list-style-type: none"> - L'audit se déroule sur 2 jours : <ul style="list-style-type: none"> - Jour 1 : Recueil d'informations via interviews et questionnaires. - Jour 2 : Analyse, restitution, et recommandations concrètes. - L'audit inclut : <ul style="list-style-type: none"> - Analyse des données - Restitution des résultats - Conseils personnalisés <p>3. Structure du formulaire</p> <ul style="list-style-type: none"> - Chapitres thématiques avec plusieurs questions. - Notation variable selon les types de questions. - Possibilité d'ajouter : <ul style="list-style-type: none"> - Texte en gras/italique/couleurs - Liens - Photos ? - divers élément tel que radio button - Saisie des informations personnelles en début de formulaire : nom, entreprise, poste, fonction, email. - Navigation flexible : possibilité de revenir sur les questions. - Les clients reçoivent uniquement le rapport final, pas le questionnaire. <p>4. Génération de livrables</p> <ul style="list-style-type: none"> - À la fin de l'audit : <ul style="list-style-type: none"> - Consultation et édition du rapport - Réouverture possible pour modification - Génération de documents Word ou PowerPoint - Diagramme de compétences - Radar chart - Heat map (optionnelle) - Documents structurés en 4 à 5 pages : <ul style="list-style-type: none"> - Page de garde - Informations auditeur/entreprise - Sommaire - Résultats - Conseils - Visualisations graphiques <p>5. Sécurité et authentification</p> <ul style="list-style-type: none"> - Authentification obligatoire pour accéder à l'outil. - Preuve de sécurité exigée pour garantir la confidentialité des données. - Stockage des audits dans une base de données sécurisée.

<p>Sujet 2 : Technologies de développement</p>	<p>L'objectif est de concevoir une application responsive et multiplateforme, capable de s'adapter à différents supports tout en garantissant une bonne évolutivité. Le projet étant destiné à vivre sur le long terme, les choix technologiques doivent être robustes et pérennes.</p> <p>1. Technologies envisagées</p> <p>Pour le front-end, deux options principales ont été discutées :</p> <ul style="list-style-type: none"> - Angular : Découvert par l'équipe l'année dernière, ce framework offre une architecture solide et une bonne évolutivité. Il repose sur TypeScript, ce qui facilite le débogage et bénéficie d'une large communauté. Toutefois, il peut montrer des limites en performance si l'application devient trop lourde, et sa courbe d'apprentissage est relativement élevée. - React.js : Bien que sa syntaxe mêlant HTML et JavaScript soit plus complexe que celle de Vue.js, React est plus puissant pour gérer des logiques complexes. Il est également très utilisé dans le monde professionnel. En version PWA, il est particulièrement adapté aux usages mobiles. <p>Pour le back-end, deux approches ont été évoquées :</p> <ul style="list-style-type: none"> - Spring Boot : Framework Java bien connu de l'équipe, il propose une configuration simplifiée, une grande modularité et une bonne intégration avec les bases de données et les API REST. Cependant, il reste assez lourd et nécessite une bonne maîtrise de l'écosystème Spring. - Nest.js : Utilise TypeScript et propose une architecture structurée. Bien qu'il soit plus complexe à prendre en main que Node.js + Express, il est plus adapté à une utilisation sur le long terme. Node.js + Express reste une bonne solution pour prototyper rapidement, mais moins évolutive. <p>Côté base de données, le choix s'est porté sur Supabase, une solution open source basée sur PostgreSQL. Elle intègre un système de CRUD natif et une authentification, ce qui permet de se passer d'un back-end si la logique métier est simple. PostgreSQL reste une référence fiable et performante.</p> <p>2. Points de vigilance</p> <p>L'utilisation d'un CMS a été évoquée, mais jugée trop limitée pour les besoins futurs du projet. Il ne permet pas de tout faire et pourrait freiner l'évolution de l'application.</p> <p>Enfin, le choix des technologies devra tenir compte de la durée de vie du projet, de sa scalabilité, et de sa capacité à répondre aux besoins mobiles, notamment via des solutions comme les PWA.</p>
<p>Sujet 3 : Synthèse de la réunion avec le client et organisation</p>	<p>1. Proposition de développement par version :</p> <ul style="list-style-type: none"> - Version 1 (V1) : Mise en place des fonctionnalités de base <ul style="list-style-type: none"> - Connexion utilisateur - Création de formulaire avec uniquement des boutons radio - Version 2 (V2) : Ajout de la génération de rapport <ul style="list-style-type: none"> - Transformation des réponses en livrable structuré - Version 3 (V3) : Intégration des composants complémentaires - Enrichissement du formulaire - Ajout de fonctionnalités avancées

2. Répartition des rôles

- Recueil des besoins : Simon
- Conception des maquettes : Pierre, Bilon
- Base de données (Supabase) : Titouan, Simon, Robin

3. Architecture du projet

Le projet repose sur une structure en trois fichiers distincts :

- Front-end : Interface utilisateur
- Back-end : Traitement des données et logique métier
- Base de données : Stockage des informations via Supabase