

Création d'une Application de Messagerie Instantanée

By : Corto GAYET

Projet FaxProd: conception d'application web de messagerie instantanée.

Contact: faxprod.contact@gmail.com



Présentation de FaxProd



Date de création

17/03/2025



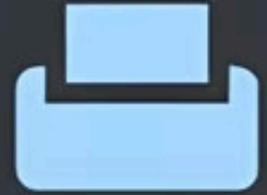
Structure

Entreprise unipersonnelle



Spécialité

Développement web



Fax

Cahier des charges

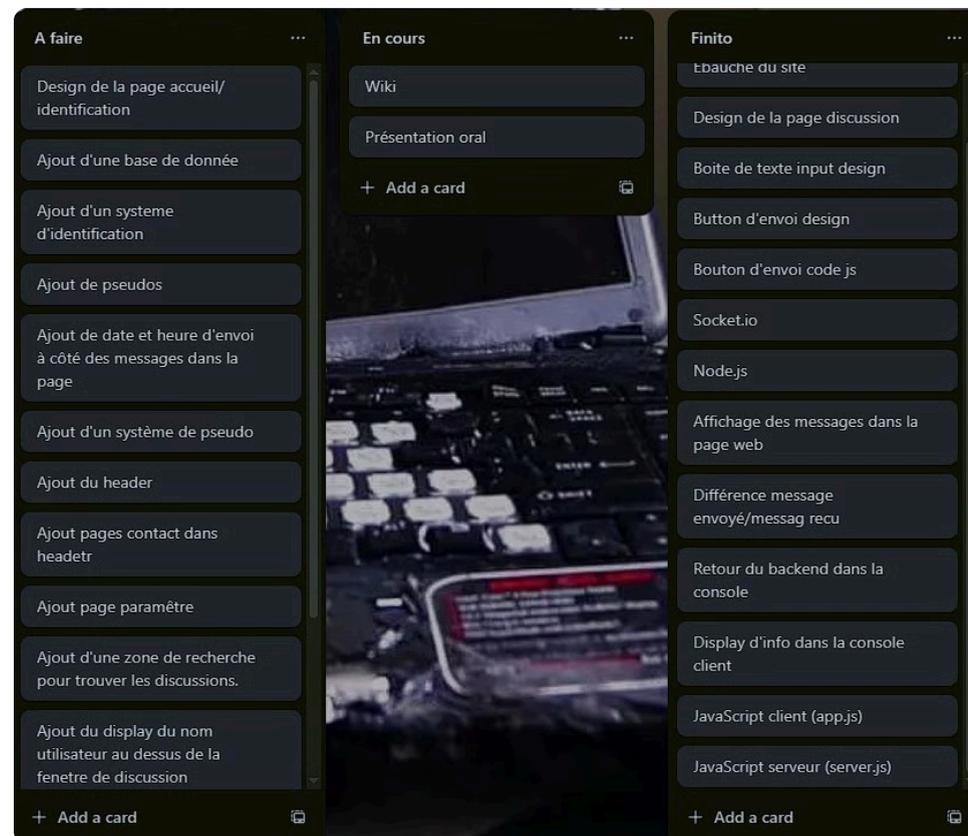
Budget FaX			
	Temps Travail	Coût (Euros)	Extra
Salaire Developpeur	9 Jours	3000	
Logo	4 Jours	1200	
Espace Coworking		50 Mensuel	
Hebergement site		15 Mensuel	
Maintenance		100 Mensuel	
		Total	4365

Budget et Trello

Besoins du projet

Création d'une application de messagerie instantanée

Gestion du backend et de la base de données.



Aperçu des Objectifs

Utiliser de nouvelles technologies

Quelles technologies utiliser pour créer une application de messagerie instantanée ?

Socket.io, Node.js, HTML/CSS

Création et utilisation d'un système de base de données

Découvrir l'utilisation, le fonctionnement et l'optimisation d'une base de données.

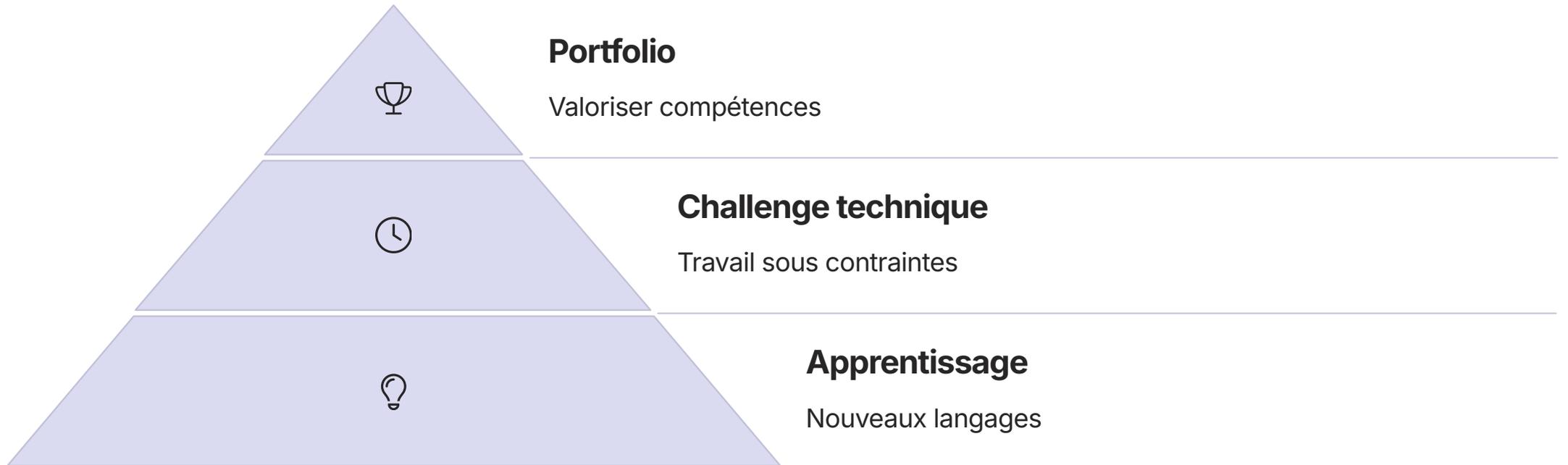
Sql ? ServerSide

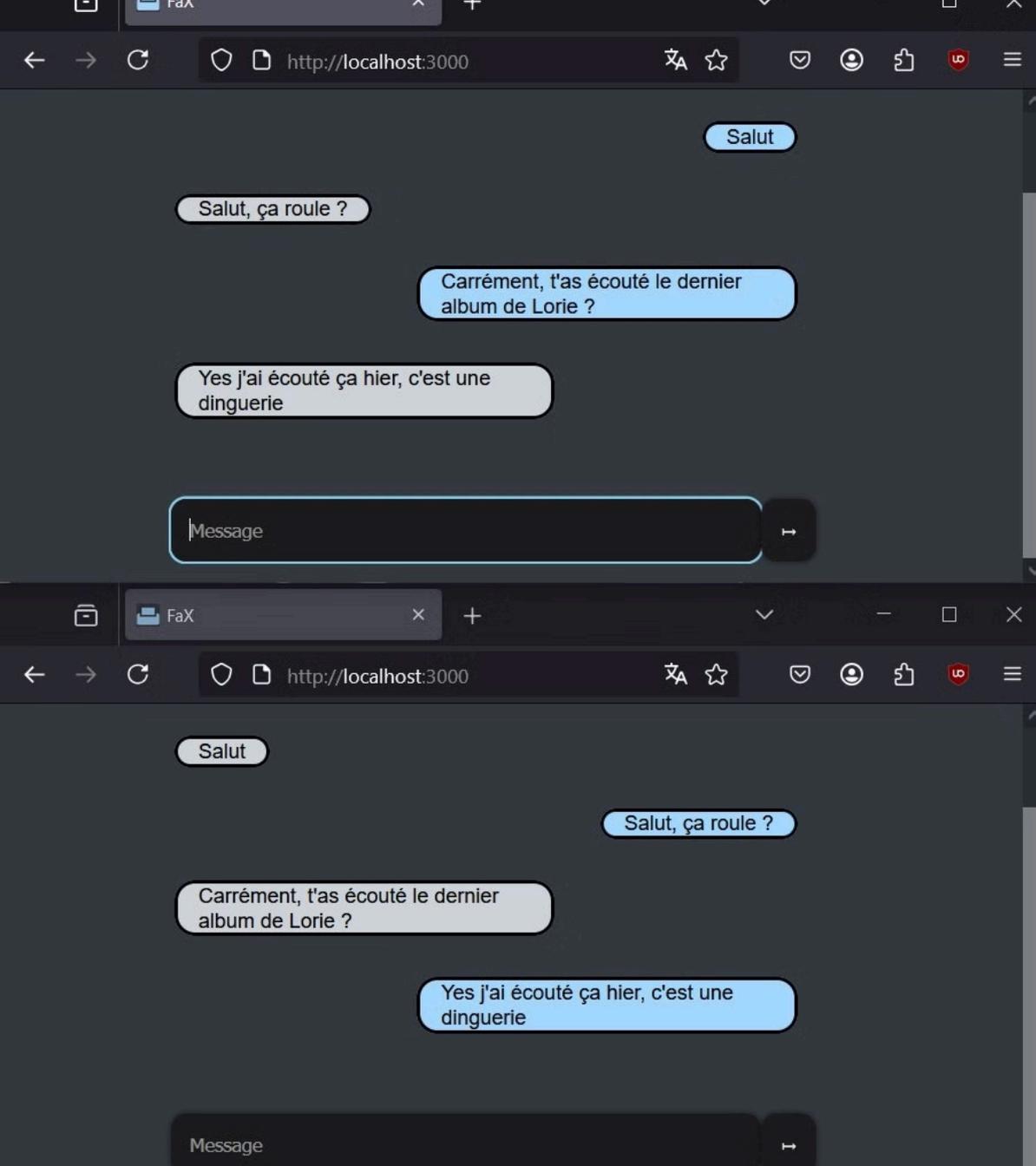
Mener à bien un projet de bout en bout.

Outils organisationnels, planning prévisionnel.

Trello

Besoins du Projet





Objectifs du Projet



Messagerie fonctionnelle

Priorité principale



Projet complet

De la conception à la livraison



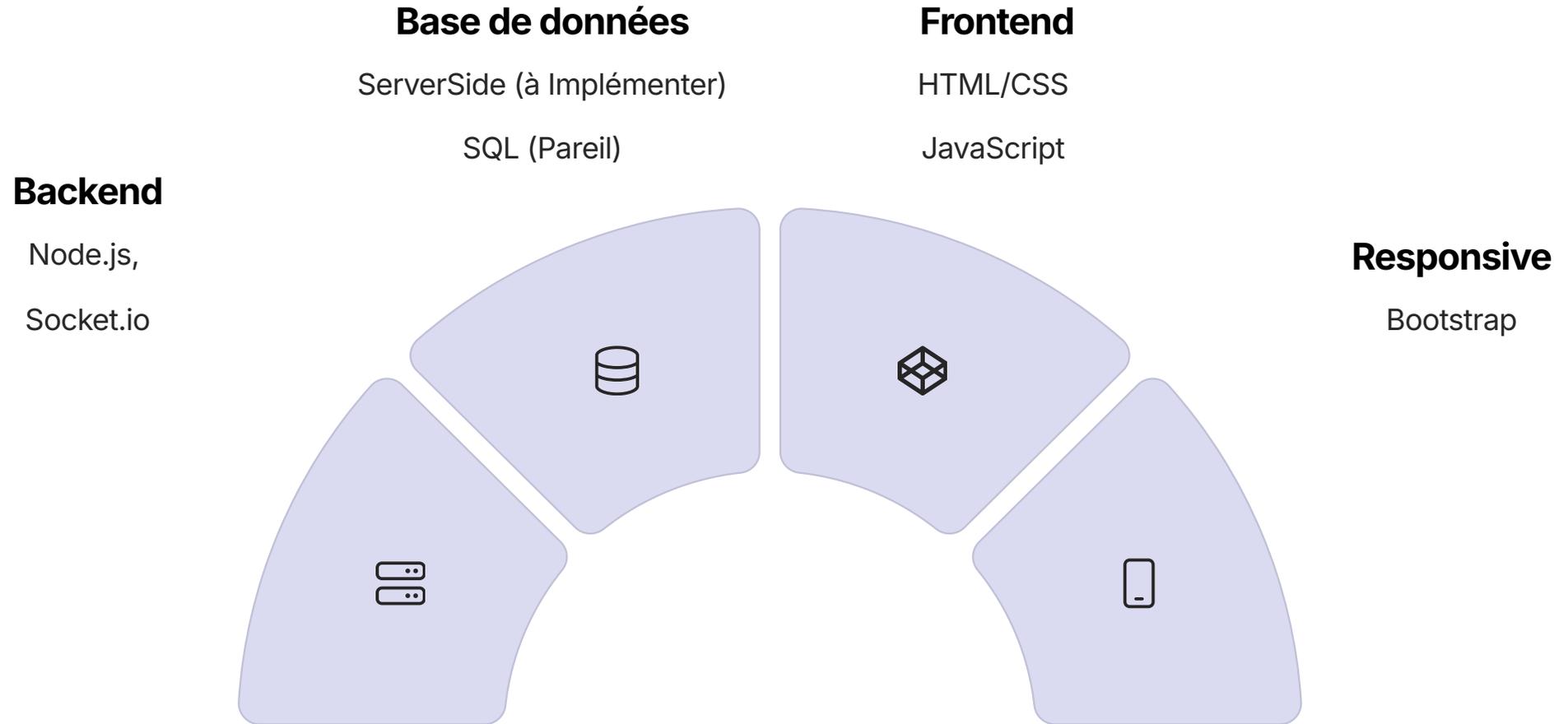
Nouvelles technologies



Base de données ?

Stockage et utilisation des messages.

Technologies Utilisées



Étapes de Réalisation

Frontend

HTML/CSS, (Bootstrap?) responsive design

Apprentissage

Socket.io, Node.js, JavaScript

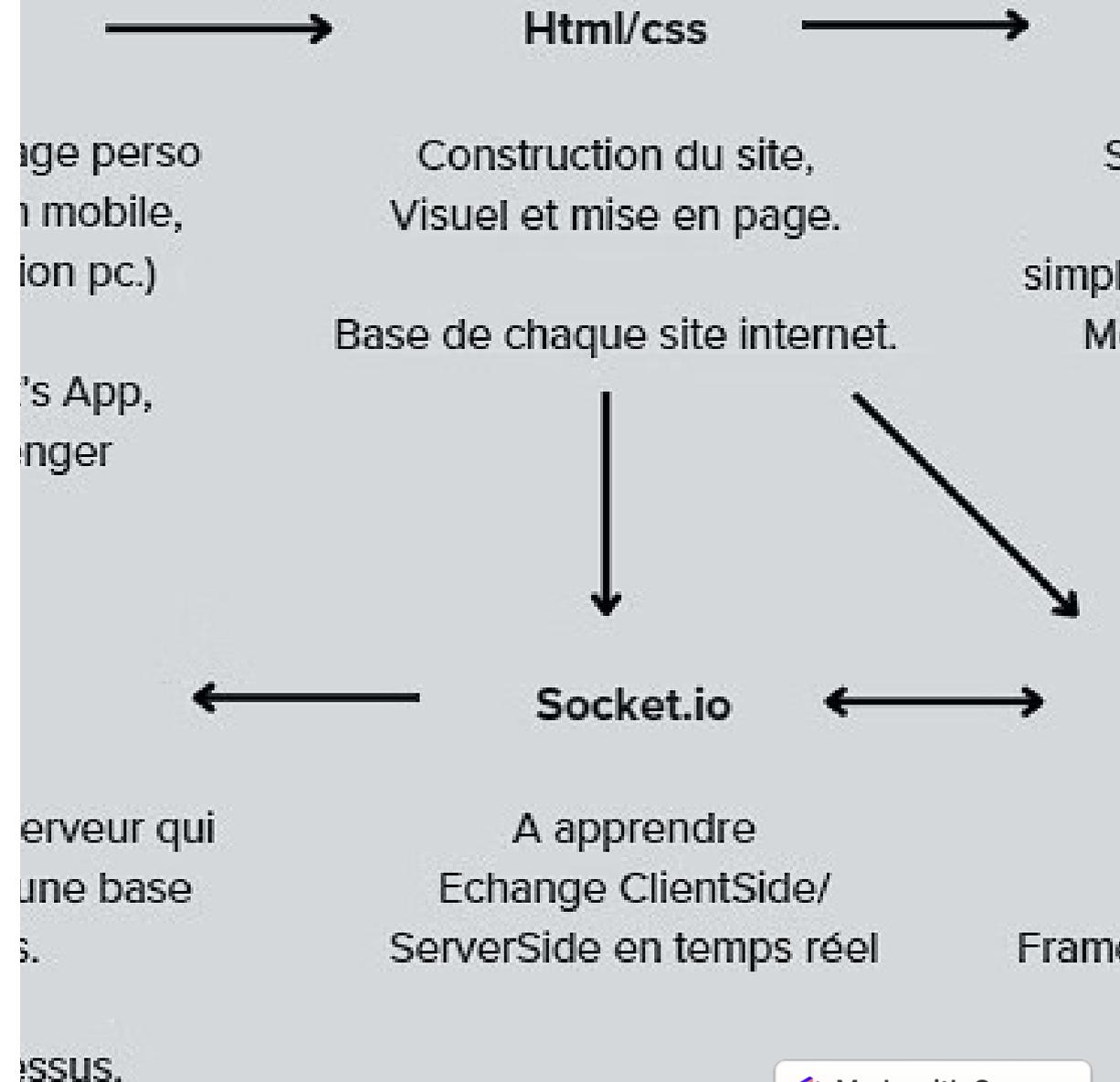
Backend

Node.js et événement socket.io

Serveur

Mise en place, communication BDD

(Base De Données)



Design de l'Application



Charte graphique

Couleurs pastels, douces pour les yeux

Typographie

H1: Calisto MT (impactant)

Paragraphes: Optima (sans empattement)

(web safe)

Inspiration

WhatsApp, Discord, Messenger

Design simple et épuré.

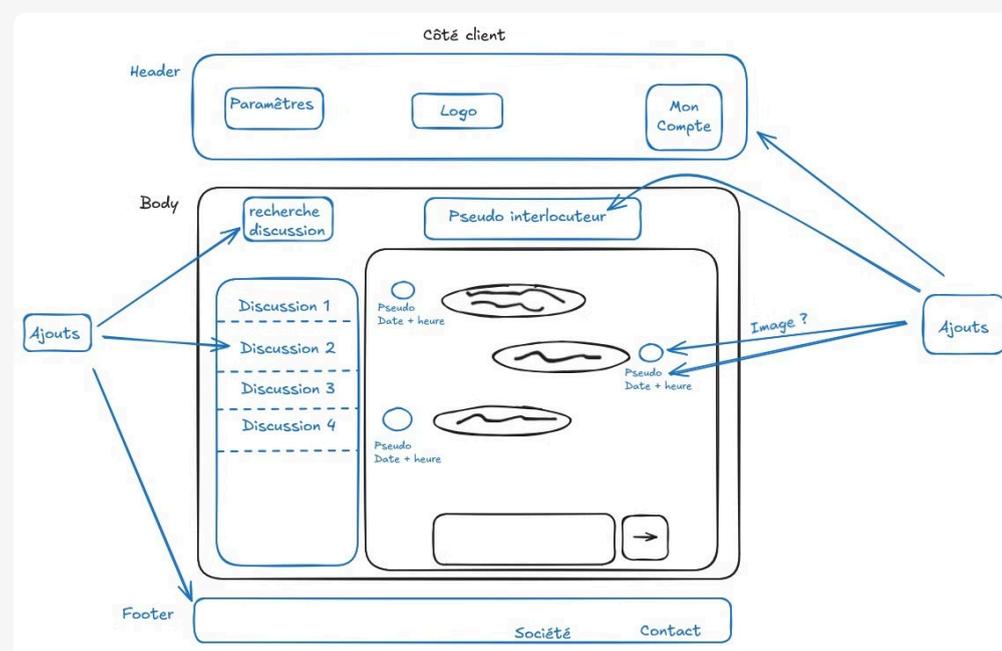
Les problèmes rencontrés

Problèmes	Solutions	Photos
Affichage peu esthétique .	Ajout de plusieurs "divs" ("boîtes") pour construire ma page de discussion : plus lisible.	<pre><!--Grande boîte--> <div class="container"> <!--Boîte de la zone de chat--> <div id="zoneChat"> <!--texte envoyé ici--> </div> </div> <!--Boîte pour entrer son texte--> <div class="boiteTextInpt"> <input type="text" name="text" id="inputTextUser" placeholder="Message" /> <!--Bouton Envoy--> <input type="submit" id="btnClk" value="#8614;" /> </div></pre>
Logique de programmation .	Test avec des fichiers API (Application Programming Interface (exemple de la cuisine)) pour me familiariser avec le format JSON,	<pre>const envoi = { pseudo: pseudo, message: messageInput.value, };</pre>
Événements socket.io	Documentation et test de mon côté	
Lien entre App.js(FrontEnd) et Server.js(BackEnd)	Documentation Socket.io et Node.js	
Implémentation du système de réponse.	Comprendre la logique de programmation et suivre la documentation du JS.	<pre>socket.emit("chat message", envoi); msgDiv.classList.add("envoye");</pre>
Différencier les messages envoyés et les messages reçus.	Système de pseudo (Géré par un nombre aléatoire pour l'instant, je compte ajouter un système d'authentification avec pseudo)	<pre>//si pseudo différent de msg.pseudo if (pseudo !== msg.pseudo) { const boxZoneChat = document.getElementById("zoneChat"); const item = document.createElement("div"); item.textContent = msg.message; //message reçu et non envoyé item.classList.add("recu"); boxZoneChat.appendChild(item); scroll(); }</pre>

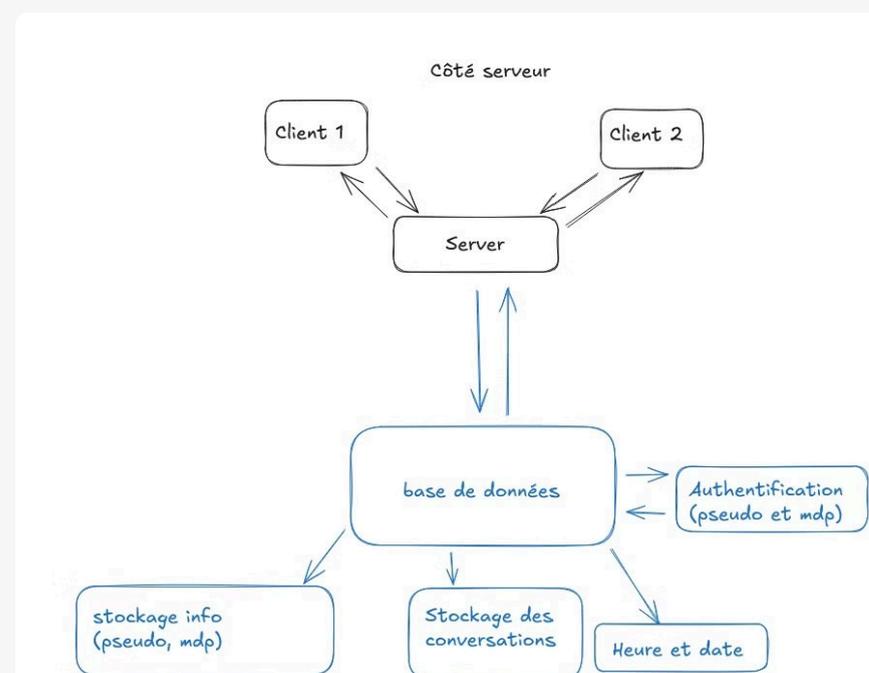
Ajouts de fonctionnalités

-  **Système d'authentification**
Ajout d'un pseudo par utilisateur, plutôt qu'un nombre aléatoire.
-  **Base de données**
Configuration d'une base de données pour conserver les informations envoyées sur le site (discussion, pseudo,...)
-  **Ajout du header et d'une Div pour les discussions**
Ajout esthétique et fonctionnel au site.
-  **Ajout de fonctionnalités**
Continuer le développement de l'application.

FrontEnd



BackEnd



Documentation ?

Wiki débrouillard.

 wiki.lesfabriquesduponant.net 

Messagerie Instantanée Fax — Les Fabriques du Ponant

Corto, stagiaire émérite du #TremplinNumérique6 vous présente la messagerie instantée FaX

Projet [modifier]

Création d'une application de messagerie instantanée
Gestion du backend et de la base de données.

Objectifs [modifier]



Logo
FaX

- # Un challenge intéressant.
- # Nouveaux langages et nouvelles librairies.
- # Travail sous contrainte de temps et rendre un projet fonctionnel.
- # Ajouter un projet concret à mon portfolio et valoriser mes compétences.

Ressources [modifier]

- IDE : [VSCodium](#)
- Outil organisationnel [Trello](#)
- Schémas : [Excalidraw](#)
- Outil de création de présentation et diapos : [Canva](#) & [Gamma](#)
- Design du logo(Fait par IA) [Logopony](#)
- Ouverture et fermeture de connection : [Node.js](#)
- Gestion des sockets de connections : [Socket.io](#)
- Votre présentation finale (celle que vous allez présenter au jury)

Mon Code [Lien Github](#)

Aide technique [modifier]

- Super site de @yann [html/css détaillé](#)
- Introduction au [HTML/CSS](#)
- Css : [Display flex](#)
- Mooc sur le JavaScript [OpenClassRoom Javascript](#)
- Qu'est-ce qu'un fichier [JSON](#) ?
- Qu'est-ce qu'une [API](#)
- Tutoriel construction d'une application simple de messagerie instantanée [Medium.com](#)
- Tutoriel de socket.io (**English**) [Socket.io](#)
- Contact : corto.gayetun@gmail.com (Envoyé un mail si vous avez des questions spécifiques)

Matériel et outils utilisés [modifier]

Un pc et une connection internet, rien de plus.

Merci de votre attention

Des questions ? Suggestions ? N'hésitez pas à me contacter.

Contact: faxprod.contact@gmail.com

