Analyse Détaillée du Sous-module : Communication

# 1. Contexte

Le sous-module Communication vise à centraliser la gestion des échanges entre utilisateurs appartenant à des entreprises partenaires, via un système de messagerie instantanée robuste et sécurisé. Il s’intègre dans le cadre global de la plateforme collaborative.

# 2. Objectifs Fonctionnels

- Mise en place d’un système de chat temps réel (type WebSocket).

- Envoi et réception de fichiers (PDF, Word, Excel).

- Support de la messagerie vocale.

- Historique des échanges (journalisation).

- Notifications pour nouveaux messages.

# 3. Acteurs Concernés

- WORKSPACE OWNER

- WORKSPACE OWNER DELEGATE

- ADMINISTRATOR

- USER

# 4. Dépendances

Sous-module : Relations inter-entreprises

- BT-47.2 : Besoin de relation entre entreprises pour autoriser la communication.

- Classe : ChatUser utilisée pour identifier les utilisateurs actifs.

# 5. Données Gérées

- ChatUser : Identité, entreprise, rôle.

- Message : ID, contenu, type (texte, fichier, audio), date.

- Conversation : Participants, historique.

- PièceJointe : Type, taille, chemin, format.

- NotificationMessage

# 6. Étapes du Développement et Temps Estimé

|  |  |  |  |
| --- | --- | --- | --- |
| Étape | Description | Responsable | Durée estimée |
| 1 | Analyse des besoins fonctionnels et techniques | Saidane Nour, Ghouma Dhia | 2 jour |
| 2 | Conception de la base de données (Modèle Message, Conversation) | Ghouma Dhia | 2 jour |
| 3 | Mise en place de l’infrastructure WebSocket | Saidane Nour | 4 jours |
| 4 | Développement du backend (envoi/réception, upload fichiers/audio) | Ghouma Dhia | 3 jours |
| 5 | Intégration frontend (UI chat, composants audio/fichier) | Saidane Nour | 4 jours |
| 6 | Système de notifications (real-time + base de données) | Ghouma Dhia | 10 jours |
| 7 | Tests unitaires + intégration | Les deux | 2 jours |
| 8 | Documentation + démonstration | Les deux | 1 jour |

# 7. Contraintes et Risques

- Sécurité des fichiers échangés (antivirus, scan de contenu).

- Temps réel → nécessite une gestion efficace des WebSockets et scalabilité.

- Stockage → volume des pièces jointes/audio dans la base ou sur un serveur distant.

- Support multiformat : messages texte, fichiers divers, audio.

- Interopérabilité avec les autres modules via les classes communes (ChatUser).

# 8. Livrables Attendus

- Modèle de données

- WebSocket Gateway

- API REST pour fichiers/audio

- Interface utilisateur (React, Vue ou autre)

- Manuel d’utilisation

- Tests et démonstration