

# Cahier des Charges - Projet de Développement

---

## 1. Informations générales

Nom de l'étudiant : Gross Samuel

Titre du projet : Conduit

Date de début : 18 septembre 2025

Date de fin prévue : 16 janvier 2026

## 2. Contexte et objectifs

Décrire le contexte dans lequel s'inscrit le projet et les objectifs principaux à atteindre.

Dans un contexte où la majorité des services de messagerie conservent les données des utilisateurs, malgré des promesses d'anonymat et de chiffrement, ce projet propose une alternative radicale. L'application vise à offrir un service qui **n'appartient à aucune entité centralisée**, qui **ne stocke aucun message échangé** et qui garantit à la fois la **vie privée** et l'**anonymat total** des utilisateurs. L'objectif est de créer un espace de communication réellement sûr, éphémère et libre, où chaque échange disparaît après transmission et ne peut être exploité à des fins commerciales ou de surveillance.

## 3. Description fonctionnelle

Lister les fonctionnalités attendues :

### MVP (version minimale mais fonctionnelle)

- Communication texte entre plusieurs utilisateurs.
- Interface simple mais utilisable.
- Mise en place du chiffrement de bout en bout pour sécuriser les échanges et respecter la confidentialité.
- Ajout du fonctionnement en pair-à-pair et des messages éphémères pour garantir l'anonymat total.

### Améliorations et extensions

- Développement d'interfaces multiplateformes (desktop, Android, iOS).

- Possibilité d'envoyer des photos, vidéos et pièces jointes.
- Intégration des appels vocaux et vidéo.
- Mise en place d'un système de déploiement et de packaging pour faciliter l'utilisation réelle.

## 4. Contraintes techniques

Préciser les langages, frameworks, bases de données, outils ou autres contraintes techniques imposées.

L'application sera développée en **Rust** avec **Cargo** comme gestionnaire de projet et de dépendances. Pour la base de données, l'implémentation initiale utilisera **SQLite**, adaptée pour le prototypage et une première version stable.

La cible principale est une application **desktop** (via Tauri), avec une extension prévue pour **Android**, et éventuellement **iOS** à plus long terme. Ce choix permet de capitaliser sur un socle unique en Rust, tout en gardant une compatibilité multiplateforme.

## 5. Organisation du projet

Découpage en sprints de 2 semaines (28h de travail par sprint).

### Sprint 1 : Communication minimale + sécurité de base

- Messagerie texte entre plusieurs utilisateurs.
- Mise en place d'un minimum de chiffrement dès le départ (même provisoire).
- Attribution d'identifiants anonymes.

### Sprint 2 : Prototype frontend (desktop)

- Première version de l'interface.
- Navigation de base (connexion, chat, paramètres).
- Intégration avec Tauri.

### Sprint 3 : Chiffrement E2EE complet + conformité RGPD

- Intégration du chiffrement de bout en bout.
- Gestion des clés locales.
- Documentation et contraintes RGPD.

#### **Sprint 4 : P2P et messages éphémères**

- Passage à un mode de communication en P2P (WebRTC/DataChannels).
- Mise en place de la suppression automatique des messages.
- Ajout d'un mécanisme de repli via serveur en cas d'échec du P2P.

#### **Sprint 5 : Envoi de médias et pièces jointes**

- Photos, vidéos, documents.
- Compression et chiffrement des fichiers.
- Gestion des limites de taille et formats autorisés.

#### **Sprint 6 : Appels vocaux et vidéo**

- Intégration WebRTC audio/vidéo.
- UI pour passer/recevoir des appels.
- Optimisation pour latence et pertes réseau.

#### **Sprint 7 : Frontend multi-plateformes (desktop + Android/iOS)**

- Portage sur mobile.
- Optimisation UI/UX responsive.
- Tests de performance et consommation de ressources.

#### **Sprint 8 : Déploiement et finalisation**

- CI/CD complet avec builds pour toutes plateformes.
- Documentation finale.
- Packaging et publication.

## **6. Livrables**

Liste des livrables attendus :

- Code source
- Documentation technique
- Tests unitaires
- Présentation finale

## 7. Environnement de développement et production

Décrire l'environnement prévu (serveurs, outils de CI/CD, outils de suivi de projet, etc.)

Github Project pour le suivi de projet, et Github Action avec Cargo pour la CI/CD

## 8. Planning prévisionnel

Présenter le calendrier global (dates de début et de fin de chaque sprint).

18/09 – 19/09 : Debut de projet

25/09 – 03/10 : Sprint 1

09/10 – 17/10 : Sprint 2

23/10 – 31/10 : Sprint 3

06/11 – 14/11 : Sprint 4

20/11 – 28/11 : Sprint 5

04/12 – 12/12 : Sprint 6

18/12 – 26/12 : Sprint 7

08/01 – 16/01 : Sprint 8

## 9. Validation

Signature de l'étudiant : Samuel Gross

Date : 18/09/25

Signature du manager pédagogique : \_\_\_\_\_