









CORTEXIA - Assistant Intelligent de Réunions

Assistant de transcription et de compte-rendu de réunions en temps réel, développé avec Electron et React.



Fonctionnalités MVP

-  Transcription temps réel (FR/EN)
-  Résumé automatique des points clés et décisions
-  Extraction d'actions avec responsables et échéances
-  Génération de compte-rendu et email de suivi en français
-  Stockage local sécurisé
-  Export Markdown, PDF et texte



Installation

Prérequis

1. **Node.js** (version 18 ou supérieure)
 - Télécharger depuis <https://nodejs.org/>
 - Vérifier l'installation : `node --version`

Lancer le projet

```
# Installer les dépendances
npm install

# Lancer en mode développement
npm start
```

L'application s'ouvrira automatiquement dans une fenêtre Electron.



Développement

```
# Lancer uniquement Vite (interface web)
npm run dev

# Lancer uniquement Electron
npm run electron

# Build de production
npm run build
```

Architecture

```
cortexia/
├── electron/          # Code Electron (main process)
│   └── main.js        # Point d'entrée Electron
├── src/
│   ├── components/   # Composants React
│   │   ├── NewSession.jsx
│   │   ├── ActiveSession.jsx
│   │   └── SessionReport.jsx
│   ├── styles/       # Styles CSS
│   │   └── app.css
│   ├── App.jsx        # Composant principal
│   ├── main.jsx       # Point d'entrée React
│   └── index.html     # HTML principal
├── package.json       # Dépendances npm
└── vite.config.js     # Configuration Vite
```

Technologies

- **Frontend:** React 18 + Vite
- **Desktop:** Electron 28
- **Transcription:** Web Speech API (MVP) → à remplacer par Whisper/Deepgram
- **LLM:** À intégrer (OpenAI GPT-4, Claude)
- **Stockage:** SQLite + JSON (local)

Utilisation

1. Créer une nouvelle session

- Donnez un titre à votre réunion
- Choisissez la source audio (microphone ou système)
- Sélectionnez la langue (FR/EN)

- Acceptez les conditions de consentement
- Cliquez sur "Démarrer la session"

2. Pendant la session

- Parlez naturellement, la transcription apparaît en temps réel
- Utilisez "Marquer ce moment" pour noter les points importants
- Mettez en pause si nécessaire
- Cliquez sur "Terminer" quand vous avez fini

3. Après la session

- Consultez le résumé automatique
- Visualisez les actions extraites
- Lisez les décisions prises
- Copiez l'email de suivi ou exportez en Markdown



Roadmap

Sprint 1 (Semaine 1) - MVP Fonctionnel

- ☒ Création session + capture micro
- ☒ Transcription temps réel
- ☒ Interface utilisateur complète
- ☒ Résumé et génération de rapport

Sprint 2 (Semaine 2) - Production Ready

- ☐ Intégration API de transcription professionnelle (Whisper/Deepgram)
- ☐ Intégration LLM pour résumés intelligents (GPT-4/Claude)
- ☐ Extraction réelle d'actions et décisions via IA
- ☐ Export PDF avec mise en forme
- ☐ Stockage SQLite des sessions
- ☐ Paramètres avancés et personnalisation

Sprint 3 - Améliorations

- ☐ Support multi-locuteurs avec identification
- ☐ Templates personnalisables
- ☐ Synchronisation cloud (optionnel)
- ☐ Plugins d'intégration (Slack, Teams, Notion)

- ☐ Mode offline complet

Conformité & Légal

- ⚠ Affichage bandeau "Transcription en cours" pendant l'enregistrement
- ✅ Case consentement obligatoire avant démarrage
- 🔒 Données stockées localement par défaut
- 📅 Journal d'audit basique intégré

Important : Vous êtes responsable de la conformité RGPD lors de l'utilisation de cette application. Assurez-vous d'avoir le consentement de tous les participants avant d'enregistrer une réunion.

Dépannage

L'application ne démarre pas

```
# Vérifier Node.js
node --version

# Réinstaller les dépendances
rm -rf node_modules package-lock.json
npm install
```

La transcription ne fonctionne pas

- Vérifiez les permissions microphone dans votre navigateur/OS
- Web Speech API nécessite Chrome/Edge (pas Firefox)
- Testez votre microphone dans les paramètres système

Erreur lors de l'installation

```
# Sur Windows, installer les build tools si nécessaire
npm install --global windows-build-tools

# Puis réinstaller
npm install
```

Contribution

Les contributions sont les bienvenues !

1. Fork le projet
2. Créez votre branche (`git checkout -b feature/AmazingFeature`)
3. Committez vos changements (`git commit -m 'Add AmazingFeature'`)
4. Push vers la branche (`git push origin feature/AmazingFeature`)
5. Ouvrez une Pull Request

Licence

MIT

Auteur

Développé avec ❤️ pour améliorer la productivité des réunions

Note: Ce projet est un MVP en développement actif. L'API Web Speech utilisée est temporaire et doit être remplacée par une solution professionnelle (Whisper, Deepgram, AssemblyAI) pour la production.

Technologies à intégrer pour la production :

- Whisper API (OpenAI) ou Deepgram pour transcription professionnelle
- GPT-4 ou Claude pour résumés et extraction intelligente
- SQLite avec chiffrement pour stockage sécurisé
- jsPDF ou Puppeteer pour export PDF de qualité