

PdfCommander

Tout ce qu'on attend d'un éditeur PDF professionnel, sur votre machine, sans abonnement ni envoi de document.

PdfCommander est un éditeur PDF multiplateforme écrit en .NET et Avalonia. Lecture, annotation, signature électronique, formulaires, conversion, reconnaissance de texte : tout s'exécute localement. Son autre moitié est une ligne de commande, pour traiter des milliers de documents sans ouvrir la fenêtre.

État : En cours de finition, au sein de la Commander Suite. Le code source est sous licence MIT ; l'édition commerciale et son parcours d'achat ne sont pas encore ouverts. Le comparatif interne du 5 septembre 2026 dit chantier par chantier ce qui est prêt.

À qui ça sert

Les métiers qui manipulent des documents sensibles

Dossiers sociaux, actes, contrats : le document ne quitte pas le poste, ce qui simplifie beaucoup la question de la conformité.

Ceux qui traitent par lots

Tampons, filigranes, fusions, conversions, signatures en série : la ligne de commande s'automatise dans un script ou une tâche planifiée.

Les postes qui paient une licence à l'année

Remplacer un éditeur PDF par abonnement par un logiciel installé localement, acheté une seule fois.

Ce que ça change concrètement

1

Un éditeur complet, pas une visionneuse

Texte, images, pages, en-têtes et pieds de page, filigranes, arrière-plans : le document se modifie vraiment.

2

Plus de vingt types d'annotation

Surlignage, formes, tampons, dessin à main levée, mesures, fils de commentaires : la relecture à plusieurs est couverte.

3

Signature électronique et vérification

Signature au format PKCS#7 et CMS, validation eIDAS, signature par lots, chiffrement par certificat.

4

Treize formats d'export

Word, Excel, PowerPoint, HTML, EPUB, RTF, SVG, Markdown, CSV et images : le PDF redevient un document modifiable.

5 Une ligne de commande qui se décrit elle-même

Plus de cent commandes d'opération, et une commande qui énumère les capacités disponibles : l'automatisation se construit sans deviner.

Comment ça marche

Le produit sépare nettement son cœur métier de ses interfaces, ce qui permet d'avoir la même logique dans la fenêtre et dans les scripts.

1

Cœur

Modèles de documents et services métier, sans aucune dépendance à l'interface graphique.

2

Fenêtre

Application de bureau Avalonia : lecture, annotation, formulaires, signature, aperçu.

3

Ligne de commande

La commande pdfcmd expose plus de cent opérations pour le traitement par lots et les tâches planifiées.

4

Extensions

Un kit de développement permet d'ajouter des greffons chargés et déchargés à la demande.

Windows, macOS et Linux partagent le même cœur. L'acquisition depuis un scanner est disponible sous Windows et Linux ; sous macOS, la découverte du matériel existe mais l'acquisition n'est pas encore branchée.

Chiffres vérifiables

Chaque chiffre ci-dessous est daté et provient du dépôt ou d'une mesure publiée. Rien n'est extrapolé.

Plus de 100 commandes

opérations disponibles en ligne de commande, plus les commandes de licence et de service

Source : README du dépôt. Deux comptages coexistent dans la documentation (108 et 110) : nous retenons la borne basse commune.

13 formats

formats d'export du PDF vers un document modifiable

Source : README du dépôt, section des points forts.

Plus de 20 types

types d'annotation pour la relecture et le commentaire

Source : README du dépôt, section des points forts.

Pour démarrer

Compilation depuis les sources, avec le SDK .NET. Le dépôt voisin NexusFile doit être cloné à côté : les deux produits partagent des briques de la Commander Suite.

1. Compiler

```
dotnet build src/PdfCommander.Cli/PdfCommander.Cli.csproj -c Debug
dotnet build src/PdfCommander.UI/PdfCommander.UI.csproj -c Debug
```

2. Lancer la fenêtre

```
dotnet run --project src/PdfCommander.UI/PdfCommander.UI.csproj
```

3. Vérifier

Le dépôt ne fige aucun compteur de tests dans sa documentation : la commande donne le résultat du jour.

```
dotnet test src/PdfCommander.Tests/PdfCommander.Tests.csproj
```

Écrivez-nous si vous voulez essayer une version compilée : nous fournissons un binaire et recueillons vos retours avant l'ouverture commerciale.

Ce qui n'est pas encore prêt

Nous préférons l'écrire noir sur blanc plutôt que de le faire découvrir après la signature.

- Le remplacement de texte directement dans la page, au fil du document, est le chantier en cours : la fonction refuse encore de s'exécuter en attendant le moteur de composition typographique.
- La signature horodatée de niveau le plus élevé est refusée tant que les structures de validation à long terme ne sont pas complètes.
- Les formulaires et la conformité PDF/A sont partiels : la validation existe, la conversion n'est pas prouvée.
- Le parcours d'achat n'est pas branché : le produit n'est pas encore vendu.

Licence, dépôt et contact

Licence

Code source sous licence MIT. L'édition distribuée comportera une activation commerciale distincte, dont le parcours n'est pas encore ouvert.

Où le trouver

- Fiche produit sur ce site
- Nous écrire pour essayer

Éditeur

Contact

AGILE UP — Structure d'ingénierie logicielle
indépendante, Vincennes.

contact@agile-up.com
agile-up.com