

```
# ITO-DicomTools
```

ITO-DicomTools est une application Python conçue pour faciliter la manipulation et la transmission des fichiers DICOM via TCP/IP. Elle propose une interface conviviale permettant de charger, afficher et envoyer des fichiers DICOM, en faisant un outil précieux pour les professionnels de l'imagerie médicale et les développeurs travaillant avec les normes DICOM.

```
![image] (https://github.com/user-attachments/assets/727774aa-7e9b-4862-a232-e0fbafef133cc)
```

```
---
```

```
## **Table des Matières**
```

1. [Contribution] (#contribution)
2. [Fonctionnalités] (#fonctionnalités)
3. [Installation] (#installation)
4. [Utilisation] (#utilisation)
 - [Ouverture de fichiers DICOM] (#ouverture-de-fichiers-dicom)
 - [Réception de fichiers DICOM] (#réception-de-fichiers-dicom)
 - [Édition des métadonnées] (#édition-des-métadonnées)
 - [Sauvegarde de fichiers DICOM] (#sauvegarde-de-fichiers-dicom)
 - [Transmission de fichiers DICOM] (#transmission-de-fichiers-dicom)
5. [FAQ et dépannage] (#faq-et-dépannage)
6. [Structure du code] (#structure-du-code)
7. [Dépendances] (#dépendances)
8. [Compilation] (#compilation)

```
---
```

```
## Contribution
```

```
- **Auteur Principal** : Cyril Bouvart
```

```
---
```

```
## Fonctionnalités
```

- **Charger des fichiers DICOM** : Sélectionnez et chargez des fichiers DICOM depuis votre système local.
- **Recevoir des fichiers DICOM** : Recevez des fichiers DICOM par TCP/IP.
- **Afficher/modifier les métadonnées DICOM** : Affichez et modifiez les métadonnées détaillées des fichiers DICOM chargés.
- **Sauvegarder des fichiers DICOM** : Sauvegardez les fichiers modifiés sur votre système local.
- **Envoyer des fichiers DICOM** : Transmettez des fichiers DICOM à un serveur spécifié via TCP/IP.

```
---
```

```
## **Installation**
```

Dézippez l'archive DicomTools.zip contenant l'exécutable dicomtools.exe. Cet exécutable "stand-alone" permet de lancer le programme sans nécessité d'installer Python sur le serveur local.

Utilisation

Ouverture de fichiers DICOM

1. Pour ouvrir un fichier DICOM depuis votre poste, cliquez sur **Ouvrir** pour faire apparaître la pop-up de sélection de fichier.
2. Une fois le fichier ouvert, il est affiché dans l'interface utilisateur. A tout moment vous pourrez revenir dessus en cliquant sur **le nom du fichier** en haut de la fenêtre.
3. Pour fermer le fichier ouvert, cliquez sur la **croix** en haut à droite de l'interface utilisateur.

Réception de fichiers DICOM

1. Ouvrez l'onglet de réception.
2. Configurez les champs suivants :
 - **Port**
 - **AET Émetteur**
3. Cliquez sur **Recevoir** pour lancer le serveur DICOM.
4. Une fois le fichier reçu, il est affiché dans l'interface utilisateur.

![image] (<https://github.com/user-attachments/assets/63ced0fc-894c-4e69-a825-e9191825bf08>)

**Édition des métadonnées

1. Après ouverture/réception, utilisez l'interface pour visualiser les tags DICOM.
2. Modifiez les champs nécessaires dans les zones de texte associées.

**Sauvegarde de fichiers DICOM

1. Pour enregistrer sur votre poste le fichier DICOM ouvert, cliquez sur **Sauvegarder** pour faire apparaître la pop-up d'enregistrement.

**Transmission de fichiers DICOM

1. Pour transmettre via TCP/IP le fichier ouvert, configurez les informations suivantes dans l'onglet **Émission** :
 - **Adresse IP**
 - **Port**
 - **AET Récepteur**
2. Cliquez sur **Envoyer** pour transmettre le fichier au destinataire.

![image] (<https://github.com/user-attachments/assets/6043eb78-a170-4269-99a4-f19bee383407>)

**FAQ et dépannage

**Problèmes fréquents

1. **Erreur de transfert syntax UID :**
 - L'application ne prend en charge que les Sop Class XRayRadiationDoseSRStorage et ModalityWorklistInformationFind. Si besoin de rajouter un autre type de fichier, merci de contacter l'auteur.
3. **Échec de l'association DICOM :**
 - Vérifiez l'AET, le port et l'IP du destinataire.

Structure du code

L'application est structurée en plusieurs modules :

- ****main.py**** : Point d'entrée de l'application. Configure l'interface utilisateur et gère les interactions.
- ****dicom_handler.py**** : Contient des fonctions utilitaires pour ouvrir et sauvegarder les fichiers DICOM.
- ****tcp_handler.py**** : Contient des fonctions utilitaires pour recevoir et transmettre les fichiers DICOM par TCP/IP.
- ****view_app.py**** : Contient les fonction permettant l'affichage de l'application, des fichiers DICOM et des fenêtre de chaque onglets.
- ****view_error.py**** : Permet l'affichage des messages à destination de l'utilisateur (pop-ups).
- ****view_filename.py**** : Permet l'affichage du nom du fichier chargé et du bouton de cloture du fichier.
- ****view_file_explorer.py**** : Crée un conteneur défilable pour afficher les données DICOM.
- ****view_menu.py**** : Permet l'affichage du menu de gauche (ouvrir, recevoir, sauvegarder, transmettre).
- ****view_receiving.py**** : Crée un formulaire pour définir le port et l'AET pour la réception TCP/IP.
- ****view_sending.py**** : Crée un formulaire pour définir l'IP, le port, l'AET et l'AEC pour l'envoi TCP/IP.

Dépendances

Un fichier requirements.txt contient toutes les dépendances Python nécessaires à l'exécution de ce programme.

- [flet](https://flet.dev/) : Pour construire l'interface utilisateur.
- [pydicom](https://pydicom.github.io/) : Pour lire et écrire des fichiers DICOM.
- [pynetdicom](https://pydicom.github.io/) : Pour recevoir et transmettre des fichiers DICOM par TCP/IP.

Compilation

Pour recompiler l'exécutable :

```
```bash
flet build windows
```
```