EF\_001 : Le programme pourra calculer une transformée de fourrier d’une série de nombres réels en virgule flottante depuis un document au format CSV.

* Valider par des tests unitaires : recherche d’un fichier csv contenant des nombres réels puis FFT de ces valeurs. Utilisation de fonction type (sinus, Dirac, fonction constante pour valider les résultats à 10-5)

EF\_002 : Les données seront écrites au format x, y, …. Avec x et y des réels

* Test unitaire pour valider le format des CSV.

EF\_003 : La taille de la série sera une puissance de 2.

* Test unitaire

EF\_004 : Les résultats pourront être écrit dans un nouveau fichier au format CSV.

* Test visuelle, et test unitaire.

EF\_005 : Les données seront écrites de la forme : « Re(x) + i Im(x),Re(y) +i Im(y), etc… », avec x et y les coefficients de fourrier.

* Test Unitaire

EF\_006 : Re(x) et Im(x) auront une précision de 3 chiffres après la virgule maximum.

* Test visuelle : génération de donnée et vérification de la précision des données enregistre

EF\_007 : Le programme pourra effectuer la transforme inverse d’une série de fourrier passé en paramètre grâce à un fichier CSV respectant les normes ci-dessus.

* Test unitaire. On vérifie que la transformée de Fourier inverse d’une transformée est égale aux données d’origine

EF\_008 : La taille de la série sera une puissance de 2.

* Test unitaire, génération d’exception en cas de non-respect de la norme

EF\_009 : L’utilisateur pourra choisir le fichier source grâce à un bouton via l’interface graphique.

* Validation visuelle

EF\_010 : L’utilisateur pourra cocher le type de traitement souhaitée sur l’interface graphique

* Validation visuelle.

EF\_011 : L’utilisateur pourra choisir le nom du fichier de sortie.

* Validation visuelle

EF\_012 : L’utilisateur pourra effectuer la transformée de fourrier d’une série de nombre complexe passée en paramètre dans un fichier au format CSV.

* Validation par Test unitaire : recherche d’un fichier csv contenant les données issues d’une exponentielle complexe, puis FFT des ces valeurs et vérification de ces valeurs

EF\_013 : Les données seront au format « Re(x) + i Im(x), Re(y) + i Im(y), … » avec x et y des nombre complexes.

* Test unitaire

EF\_014 : En cas de non-respect des règles citée ci-dessus la donnée ne seras pas traitée.

* Validation visuelle

EF\_015 : La séparation avec les décimales seras faites avec des ‘.’ Exemple : douze virgule 6 s’écriras « 12.6 ».

* Test unitaire

EF\_016 : Un fenêtre POP-UP devra se lancer lors d’une erreur.

* Validation visuelle (Test avec un fichier au mauvais format, sans envoyer de données)

EF\_017 : La fenêtre POP-UP devra contenir le message d’erreur.

* Validation Visuelle

EF\_018 : La fonction des transformer pourra être visible dans un repère x ;y.

* Validation visuelle (présence d’un graphique dans l’interface)

EF\_019 : L’utilisateur pourra enregistrer les résultats dans un fichier csv grâce à un bouton via l’interface graphique.

* Validation visuelle