

Spécifications

Sonny Klotz - Jean-Didier Pailleux - Malek Zemni

*Interface de chargement, de contrôle
et d'analyse statistique des données
pour la constitution d'un graphe de flux*

13/04/2017



Module *Projet*

Table des matières

1	Package Chargement des données	1
1.1	Module Vérification format fichier	1
1.2	Module Analyse contenu fichier	2
2	Package Analyse descriptive des données	2
3	Package Interface web	2

Introduction

Ce document va décrire l'ensemble des exigences fonctionnelles que doit satisfaire notre produit final, c'est-à-dire les différentes fonctionnalités que notre application va fournir. Cette description va prendre en compte les caractéristiques des outils de développements choisis.

Notre outil, Python, étant un langage de programmation hybride, on utilisera d'une part la programmation objet pour le développement des interfaces graphiques, et d'autre part la programmation fonctionnelle pour les calculs. Dans Python, le type de données n'est connu qu'à l'exécution (typage dynamique), par conséquent, ces types ne seront pas indiqués dans les signatures des fonctions et les structures des classes. Ils seront précisés dans un paragraphe explicatif.

Pour les parties qui s'appuient sur une interaction avec l'utilisateur, notre démarche d'écriture des spécifications va essentiellement prendre en compte l'*expérience utilisateur*¹. La description va donc qualifier le résultat et le ressenti de l'utilisateur lors de la manipulation de l'interface fournie.

Les fonctionnalités de notre application seront présentées selon les modules de l'organigramme établi dans le cahier des charges. Ces modules eux-mêmes seront regroupés en packages. Ce document va donc décrire, pour chaque package de l'organigramme, les fonctionnalités de ses modules : d'abord ceux du package de chargement des données, ensuite ceux du package d'analyse descriptive des données et enfin ceux du package de l'interface web.

1 Package Chargement des données

Ce package va être livré au client pour une intégration externe. On pourra donc parler de API.

1.1 Module Vérification format fichier

Ce module va vérifier le format du fichier fourni en entrée en 3 points. Il aura donc 3 fonctionnalités :

1. Fonctionnalité de vérification de l'ouverture du fichier :

```
1 def verifOuverture(fichierCSV)
```

Paramètres :

-
1. <http://uxdesign.com/ux-defined>

`fichierCSV : TextIoWrapper` (type représentant les entrées/sorties en général, dont les fichiers)

Type retour : booléen

Description : cette fonction prend en entrée le fichier CSV ouvert. Elle vérifie que le paramètre `fichierCSV` contient des informations cohérentes d'un fichier, et renvoie un booléen vrai si c'est le cas, faux sinon.

2. Fonctionnalité de vérification de l'extension du fichier ouvert :

```
1 verifExtenstion(fichierCSV)
```

Paramètres :

`fichierCSV : TextIoWrapper` (type représentant les entrées/sorties en général, dont les fichiers)

Type retour : booléen

Description : cette fonction prend en entrée le fichier CSV ouvert. Elle vérifie que le paramètre `fichierCSV` est...

1.2 Module Analyse contenu fichier

2 Package Analyse descriptive des données

Ce package va être livré au client pour une intégration externe. On pourra donc parler de API.

3 Package Interface web

Conclusion