

# Organigramme

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*

14/03/2017

# 1 Présentation de l'organigramme

Le projet peut être décomposé en 3 grands paquets :

- Une **interface graphique web** : module d'interface utilisateur qui va faire transiter les informations entre les différentes APIs
- Une **API 1 de chargement des données** : composée de 3 modules
- Une **API 2 d'analyse descriptive des données** : composée de 3 modules

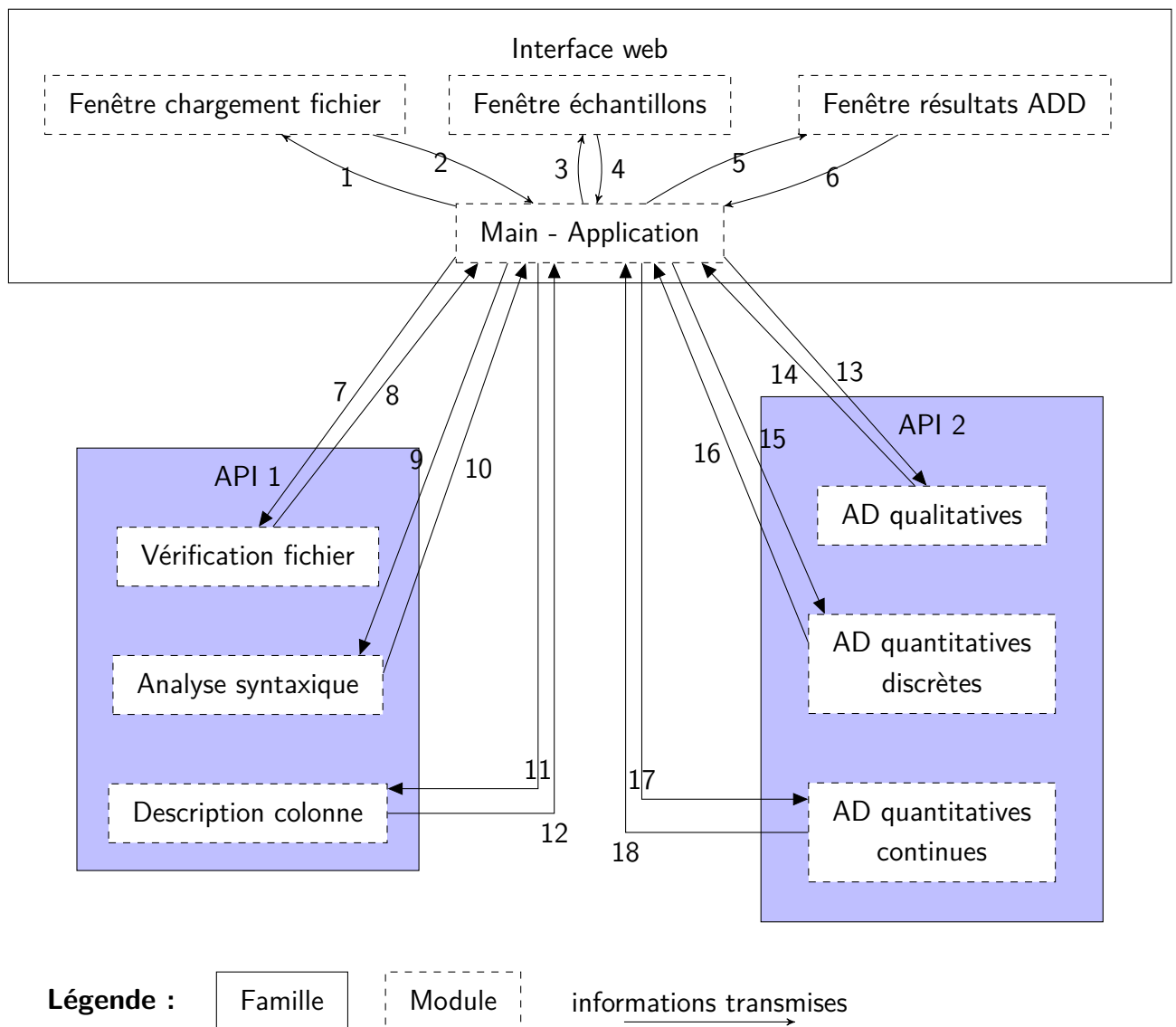


FIGURE 1 – Organigramme des différents modules du logiciel

**Notes :**

- (1)
- (2)

- (3)
- (4)
- (5)
- (6)
- (7) Fichier `.csv` : lancement de la vérification de son format
- (8) Fichier OK ou ERREUR : ouverture ou format du fichier
- (9) Fichier `.csv` : lancement de l'analyse syntaxique de son contenu
- (10) **Structure 1** contenant les données du fichier, le nombre de lignes et le nombre de colonnes (taille de la structure)
- (11) **Structure 1** : lancement de la description des colonnes de la **Structure 1**
- (12) **Structure 2** contenant 3 champs d'informations sur chaque colonne : le type, le rôle et les données erronées
- (13)
- (14)
- (15)
- (16)
- (17)
- (18)

## 2 Fonctionnalités des modules

### 2.1 API 1 : Chargement des données

L'API de chargement des données va être divisée en 3 modules :

#### 2.1.1 Vérification format fichier :

Ce module va vérifier le fichier fourni en entrée en plusieurs points :

- l'ouverture du fichier a réussi
- le fichier est un `.csv` contenant du texte brut non formaté (pas de mise en forme avec des balises ou autres)
- le fichier est accessible en lecture

#### 2.1.2 Analyse syntaxique contenu :

Ce module va exécuter une analyse syntaxique sur le texte du fichier `.csv` afin de détecter les différentes données et les stocker dans une structure.

```

1 //On initialise une structure dynamique pouvant stocker les
  ↳ données du fichier (tableau 2D dynamique ou liste de
  ↳ listes)
2 Pour toute ligne du fichier
3   Lire les caractères de la ligne jusqu'à sa fin
4   Si on lit un caractère de séparation (une virgule, un
  ↳ point-virgule ou une tabulation) ou une fin de ligne, on
  ↳ stocke les caractères lus dans la case correspondante de
  ↳ la structure

```

Ce module fournit donc une structure contenant les données du fichier, le nombre de lignes et le nombre de colonnes (taille de la structure).

### 2.1.3 Description contenu colonnes :

Ce module va donner des informations préliminaires sur les données de chaque colonne du fichier :

- le type de chaque colonne
- le rôle de chaque colonne
- les données manquants ou erronées de chaque colonne

```

1 //On initialise une structure dont la taille est le nombre de
  ↳ colonnes qui contient 3 champs
2 Pour toute colonne du fichier
3   Lire chaque donnée et la comparer au type attendu
4   Si la donnée ne correspond pas au type, signaler dans la
  ↳ structure une donnée erronée
5   A chaque fin de lecture de colonne, attribuer un rôle à
  ↳ celle-ci

```

Ce module fournit donc une structure contenant 3 informations sur chaque colonne : le type, le rôle et les données erronées.

## 2.2 API 2 : Analyse descriptive des données

### 2.2.1 Module 1 :

### 2.2.2 Module 2 :

### 2.2.3 Module 3 :

Ce module est utilisé pour fournir des informations de statistiques descriptives sur les colonnes de données qui lui seront fournies. Ses fonctionnalités se décomposent en

trois pour les différents types de données à traiter :

1. qualitatif : calcul des effectifs et fréquences d'apparition
2. quantitatif discret :
  - effectifs, effectifs cumulés, fréquence et fréquence cumulée
  - moyenne
  - médiane et autres quantiles
  - variance et écart-type
  - anomalies : boîte à moutaches de Tukey
  - symétrie : coeff de Pearson ou coeff de Yule
  - aplatissement : coeff de Fisher
3. quantitatif continu : on regroupe les valeurs en classe d'intervalles, on peut ensuite appliquer les mêmes techniques que pour les variables discrètes

## 2.3 Interface graphique web

Le module de l'interface graphique s'occupera de la manière dont l'application sera représentée à l'écran pour l'utilisateur, ce qui correspond au positionnement des éléments textuels, boutons et des fonctionnalités disponible dans une fenêtre. Voici une liste des principales fenêtres composant ce module avec leurs fonctionnalités :

1. Fenêtre de chargement pour récupérer un fichier `.csv` avec validation du choix pour passer à la prochaine fenêtre (En renseignant son chemin dans le système de fichier, ou de la manière d'un Drag & Drop).
2. Fenêtre de contrôle préliminaire avec visualisation d'un échantillon du `.csv`.
  - Affichera le nombre de lignes/colonnes contenu dans le `.csv`.
  - Affichera le titre du fichier.
  - Affichera un échantillon du contenu du `.csv` (environ les 1000 premières lignes) avec un système de scroll.
  - Affichage des lignes erronées (numéro de la ligne + contenu + type d'erreur).
  - Mise en place d'un système de navigation sous forme d'onglet (Onglet erreurs, onglet échantillon,...). Cela permettra d'éviter que la fenêtre contienne trop d'informations.
  - Affichage d'un bouton pour passer à l'étape de l'analyse de données.
3. Fenêtre correspondant à l'étape de l'analyse de données.

- L'Utilisateur devra sélectionner la colonne avec un clic, puis pourra lancer l'analyse sur celle-ci.
- La fenêtre affichera les résultats de l'étude qualitative (Médiane, Quantile et anomalie) d'une part et de l'étude quantitative (Histogramme et Diagramme de secteur) d'autre part.
- Un bouton pour lancer l'exportation des résultats sera disponible (Écriture dans un nouveau fichiers).

### 3 Coûts

Module	Nombre de lignes	Coût
Salut	113	nada

## 4 Sources

- <http://www.math.univ-toulouse.fr/~baccini/zpedago/asde.pdf>
- <http://www.math.univ-toulouse.fr/~besse/Wikistat/pdf/st-l-des-uni.pdf>
- <http://iml.univ-mrs.fr/~reboul/cours2.pdf>
- <http://commons.apache.org/proper/commons-math/userguide/stat.html>
- <https://hal.archives-ouvertes.fr/halshs-00287751/document>
- Wikipédia : Données aberrantes.