

Chapitre 2

Analyse et Spécification des besoins

Dans ce chapitre , nous allons nous focaliser sur les besoins fonctionnels et non fonctionnels qu'offre notre application ,ensuite présenter le diagramme de cas d'utilisation global des deux acteurs : utilisateur final et développeur , les diagrammes de cas d'utilisation détaillés pour finir avec le diagramme de séquence .

2.1 Les acteurs

C'est une entité externe qui interagit avec notre système et apte à agir sur ce dernier . En réponse à une action de la part d'un acteur, le système satisfait ses besoins . Dans le cas de notre application nous citons :

2.1.1 Développeur

C'est la personne expérimentée en développement software qui utilise les modules de la librairie afin de développer une application web qui demande une fonctionnalité de visualisation de données

2.1.2 Utilisateur final

Il est l'utilisateur du produit qui interagit avec les composants graphiques pour visualiser ses données .

2.2 Expression des besoins

Dans cette section , nous présentons les différents besoins fonctionnels et non fonctionnels que notre projet doit assurer.

2.2.1 Expression des besoins fonctionnels

Les besoins fonctionnels représentent les fonctionnalités attendues par chaque acteur qui interagissant avec notre application. Toute solution conceptuelle proposée doit être adéquate à l'ensemble de besoins dans le but d'exprimer clairement les finalités de notre projet PCD et faciliter la phase de développement.

Dans la partie qui suit, nous présentons les fonctions que notre solution doit satisfaire.

À travers cette application l'utilisateur final devra pouvoir :

- Charger des fichiers de données respectant les formats supportés par l'application, stockés sur sa machine cliente.
- Afficher les données dans une grille.
- Afficher les données dans une grille de pivot.
- Visualiser les données en utilisant des graphiques appropriés.

À travers cette application le développeur devra être capable de :

- Intégrer les modules de visualisation développés dans une application Angular.
- Personnaliser les composants de la bibliothèque.
- Manipuler le style des composants de la bibliothèque.

2.2.2 Expression des besoins non fonctionnels

Les besoins non fonctionnels consistent en les contraintes que le système doit satisfaire en garantissant ainsi une bonne utilisation de l'application . Parmi ces besoins nous pouvons énumérer :

- **Compatibilité** : L'application devra supporter les formats de données JSON et CSV.
- **Performance** : L'application devra supporter les données importées et fournir un résultat en un temps raisonnable .
- **Maintenabilité** : L'application devra assurer une bonne qualité de code .
- **Conformité** : L'application devra être conforme à une SPA (Single Page Application) basée sur Angular Framework qui supporte les navigateurs Web majeurs .
- **Extensibilité** : L'application devra assurer une manière systématique d'ajout des fonctionnalités .

2.3 Spécification des besoins

Dans cette section , nous allons nous focaliser sur les diagrammes de cas d'utilisation ainsi que les diagrammes de séquence.

2.3.1 Diagrammes de cas d'utilisation

Dans cette partie , nous allons citer et expliquer les différents cas d'utilisation de l'application . Les cas d'utilisation décrivent les différentes fonctionnalités du développeur et de l'utilisateur final .

2.3.1.1 Diagrammes de cas d'utilisation global

Les fonctionnalités diverses qu'offre notre système sont décrites dans le diagramme de cas d'utilisation global présent ci-dessous .

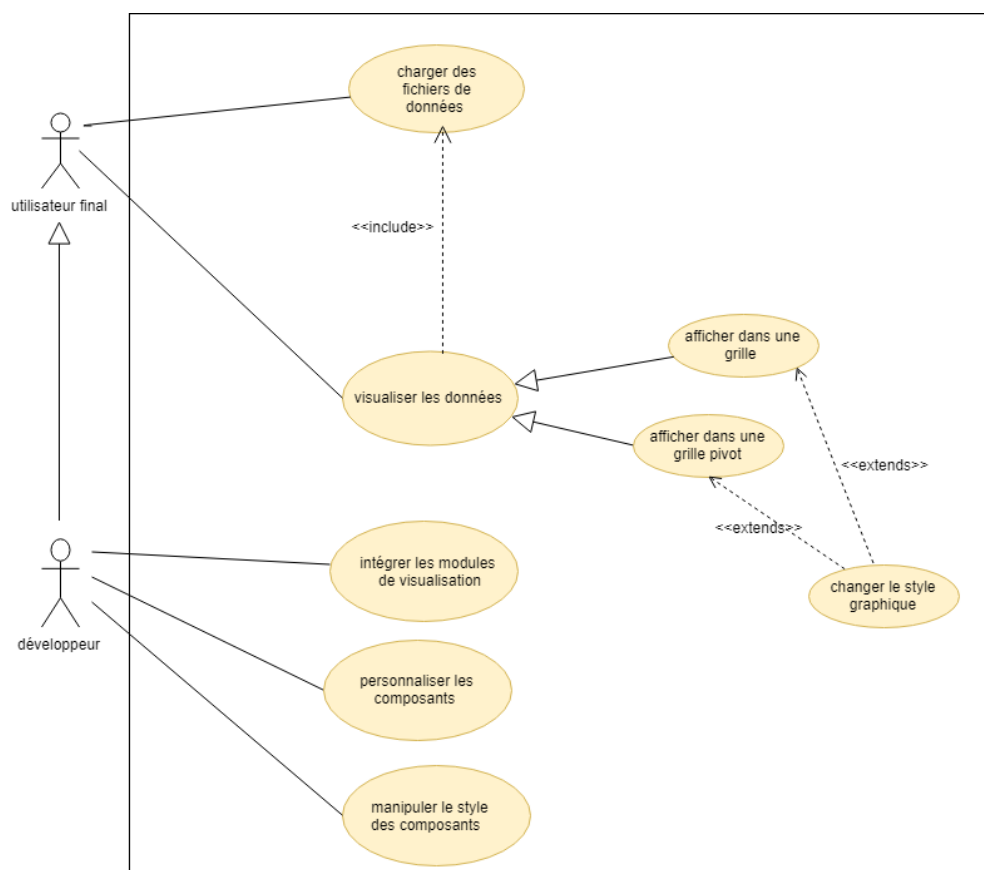


FIGURE 2.1 – diagramme d'utilisation "globale"

2.3.1.2 diagramme de cas d'utilisation "chargement des données"

En effet , l'utilisateur doit choisir le mode d'importation de ses fichiers soit depuis un ordinateur soit à partir d'un url comme l'indique la figure 2.2 .

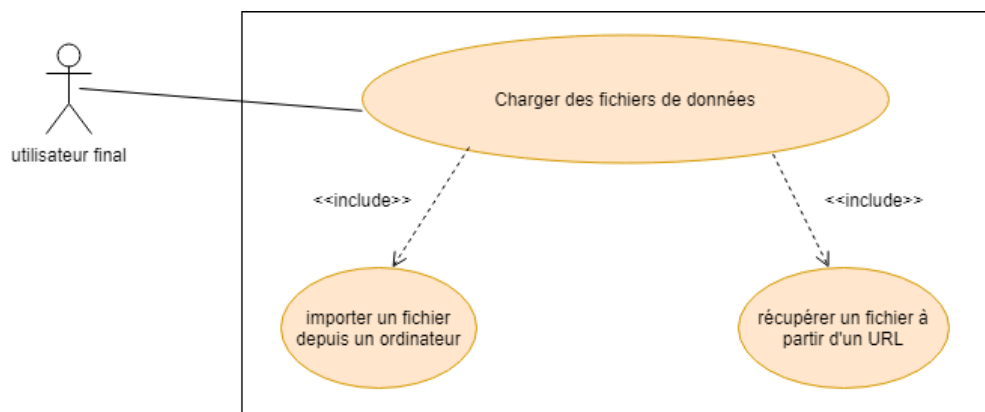


FIGURE 2.2 – diagramme d'utilisation "chargement des données"

2.3.1.3 diagramme de cas d'utilisation "visualiser les données"

En effet , l'utilisateur doit choisir le mode de visualisation souhaité de ses fichiers : histogramme , courbe , camembert . Comme indique la figure 2.3 .

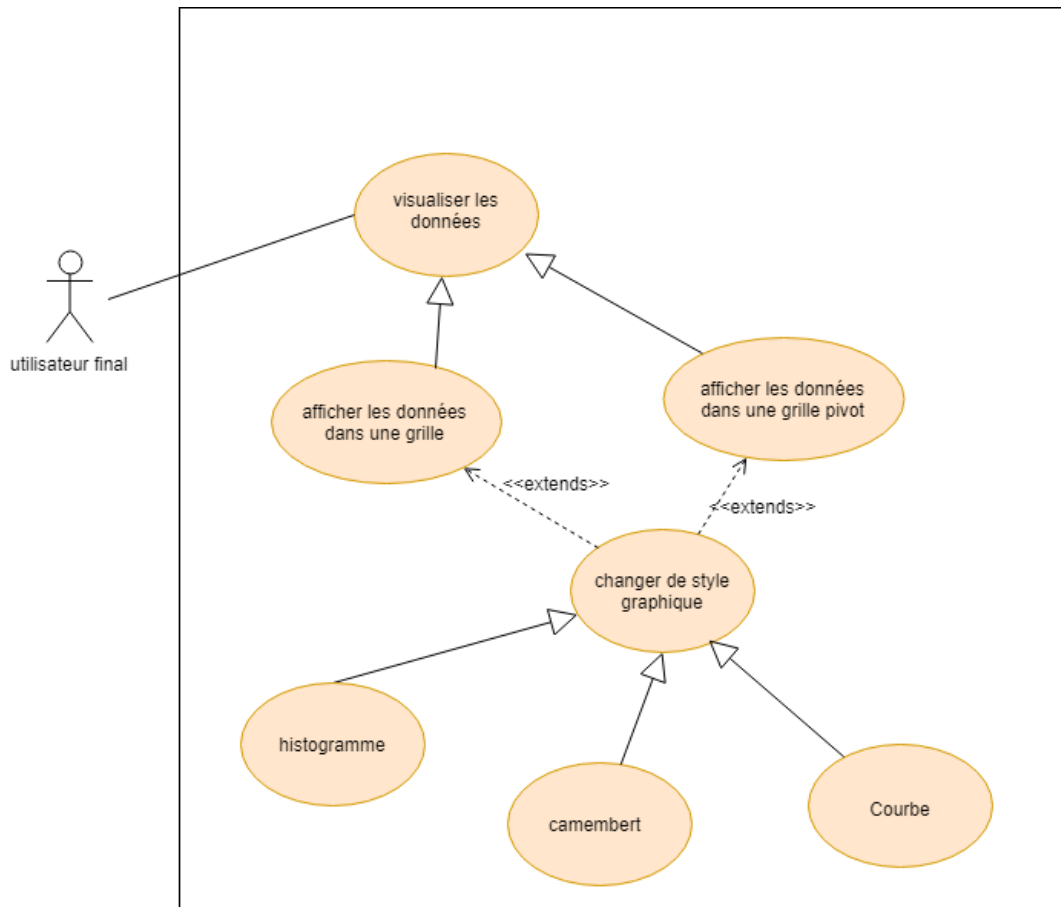


FIGURE 2.3 – diagramme d'utilisation "visualisation des données"

2.3.2 Diagrammes de séquences

Dans cette partie nous allons découvrir les diagrammes de séquences qui offrent une vue d'ensemble et dynamique sur l'interaction des différents acteurs avec le système.

2.3.2.1 Diagramme de séquence "chargement des données depuis un ordinateur"

La figure 2.3 présente le diagramme de séquences système du cas d'utilisation "chargement des données depuis un ordinateur".

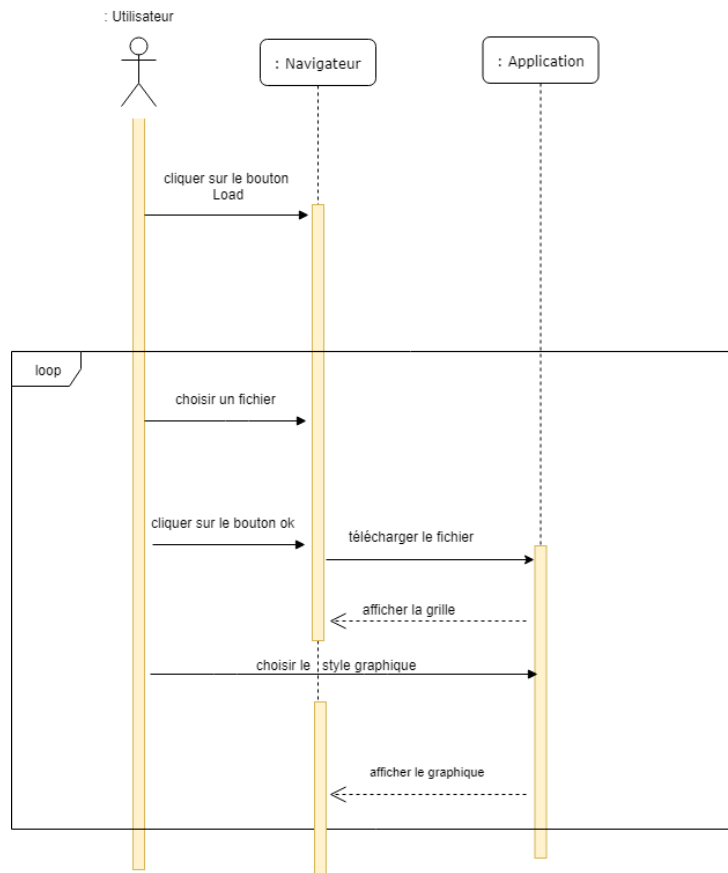


FIGURE 2.4 – diagramme de séquence "chargement des données depuis un ordinateur"

2.3.3 Diagramme système du cas d'utilisation "chargement des données depuis une URL"

La figure 2.4 présente le diagramme de séquences système du cas d'utilisation "chargement des données depuis un ordinateur".

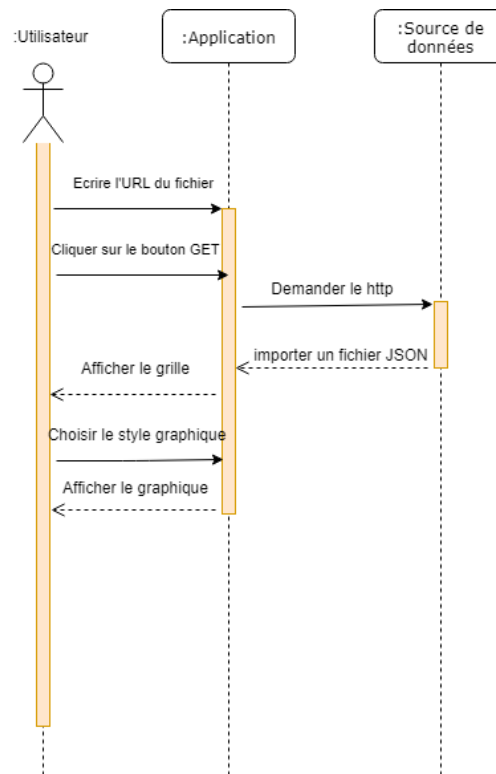


FIGURE 2.5 – diagramme système du cas d'utilisation "chargement des données depuis une URL"

Préconditions : On suppose que :

- L'utilisateur est connecté à internet.
- L'utilisateur a un navigateur Web.
- L'utilisateur a déjà accédé à l'URL de l'application Web.
- L'utilisateur a un fichier JSON qu'il veut visualiser.

Scénario nominal : 1.choisir un fichier à visualisé 1.1 depuis un ordinateur 1.2 depuis un URL 2.afficher la grille 3.choisir le style graphique 4.afficher le graphique

S scénario avec exceptions :

1. choisir un fichier vide 1.choisir un fichier ne possède pas le format approprié

2.4 Conclusion

Nous avons achevé la partie analyse de ce projet dans laquelle nous avons présenté les différents besoins et facettes de l'application d'une façon formelle. Nous avons commencé avec l'introduction des acteurs, suivie d'une énumération des besoins fonctionnels et non fonctionnels.

Aussi , nous avons présenté les diagrammes d'utilisation et les diagrammes de séquence . Maintenant, nous allons passer à l'étape de conception.