

Présentation Cahier des Charges

Sonny Klotz - Jean-Didier Pailleux - Malek Zemni

UVSQ

28 mars 2017

- 1 Introduction
- 2 Motivations du projet
- 3 Contraintes
- 4 Exigences fonctionnelles
- 5 Exigences non fonctionnelles
- 6 Autres aspects
- 7 Conclusion

Projet de L3 informatique UVSQ, remis par DCbrain.

Le travail est décrit dans ce cahier des charges :

- Motivations
- Contraintes
- Exigences fonctionnelles
- Exigences non fonctionnelles
- Autres aspects du projet

- 1 Introduction
- 2 Motivations du projet
- 3 Contraintes
- 4 Exigences fonctionnelles
- 5 Exigences non fonctionnelles
- 6 Autres aspects
- 7 Conclusion

Problème : masses de données importantes et difficilement exploitables

Objectif : fournir un outil d'analyses préliminaires de ces données

Problème : masses de données importantes et difficilement exploitables

Objectif : fournir un outil d'analyses préliminaires de ces données

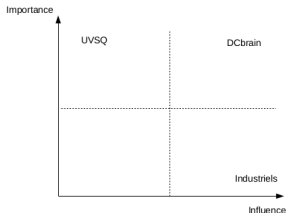
Parties prenantes :

Problème : masses de données importantes et difficilement exploitables

Objectif : fournir un outil d'analyses préliminaires de ces données

Parties prenantes :

- **Maitre d'ouvrage** : DCbrain, module Projet de l'UVSQ
- **Client** : DCbrain
- **Autres** : industriels

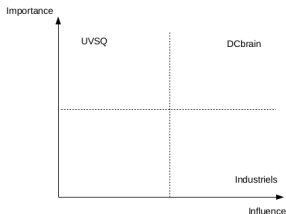


Problème : masses de données importantes et difficilement exploitables

Objectif : fournir un outil d'analyses préliminaires de ces données

Parties prenantes :

- **Maitre d'ouvrage** : DCbrain, module Projet de l'UVSQ
- **Client** : DCbrain
- **Autres** : industriels



Utilisateurs : membres de DCbrain + industriels du domaine

- 1 Introduction
- 2 Motivations du projet
- 3 Contraintes**
- 4 Exigences fonctionnelles
- 5 Exigences non fonctionnelles
- 6 Autres aspects
- 7 Conclusion

3 exigences contraignantes pour le produit :

3 exigences contraignantes pour le produit :

- Fournir une application web ***applet***

3 exigences contraignantes pour le produit :

- Fournir une application web ***applet***
- Développé avec un langage permettant une analyse de données

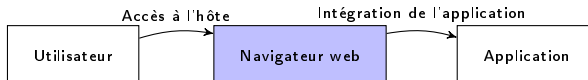
3 exigences contraignantes pour le produit :

- Fournir une application web ***applet***
- Développé avec un langage permettant une analyse de données
- Fournir API d'analyse de données en sortie

3 exigences contraignantes pour le produit :

- Fournir une application web ***applet***
- Développé avec un langage permettant une analyse de données
- Fournir API d'analyse de données en sortie

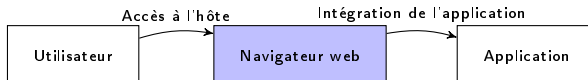
Environnement de fonctionnement : celui d'une ***applet***



3 exigences contraignantes pour le produit :

- Fournir une application web **applet**
- Développé avec un langage permettant une analyse de données
- Fournir API d'analyse de données en sortie

Environnement de fonctionnement : celui d'une **applet**

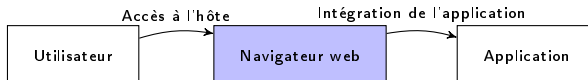


Applications partenaires : outils de DCbrain (intégration API)

3 exigences contraignantes pour le produit :

- Fournir une application web **applet**
- Développé avec un langage permettant une analyse de données
- Fournir API d'analyse de données en sortie

Environnement de fonctionnement : celui d'une **applet**



Applications partenaires : outils de DCbrain (intégration API)

Temps et budget : rendu avant le 26/05/2017, aucun budget.

- 1 Introduction
- 2 Motivations du projet
- 3 Contraintes
- 4 Exigences fonctionnelles**
- 5 Exigences non fonctionnelles
- 6 Autres aspects
- 7 Conclusion

Acteurs

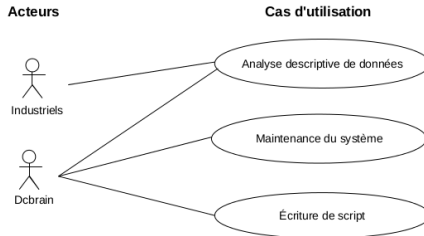


Cas d'utilisation

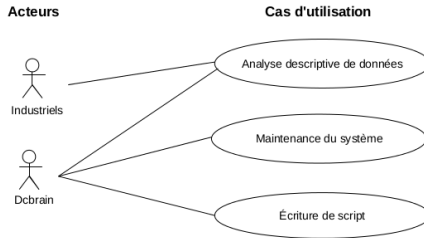
Analyse descriptive de données

Maintenance du système

Écriture de script



- **ADD** : Importation .csv -> Echantillon -> Analyse
- **Maintenance** : Mises à jour d'extension et de correction
- **Script** : Installation des paquets -> Programmation

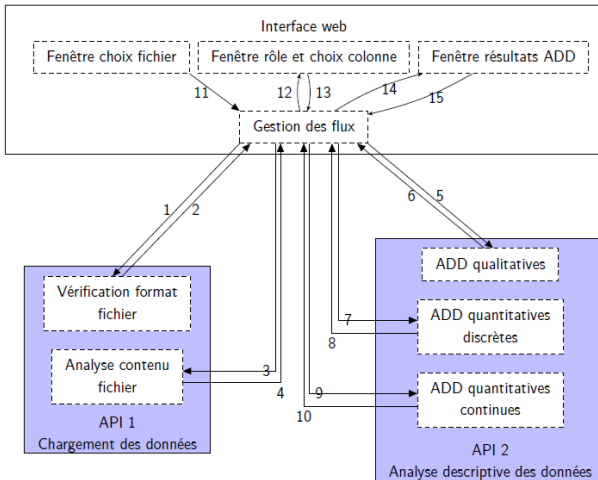


- **ADD** : Importation .csv -> Echantillon -> Analyse
- **Maintenance** : Mises à jour d'extension et de correction
- **Script** : Installation des paquets -> Programmation

Format du fichier .csv

- **Timestamp** : Date et heure des mesures
- **Père et Enfant** : Connexion où les mesures sont faites
- **Mesure (unité)** : Valeur

Exigences fonctionnelles



Légende :

Famille

Module

informations transmises

API 1 : Chargement des données

Vérification format fichier :

- format csv
- ouverture en lecture
- texte brut non formaté

API 1 : Chargement des données

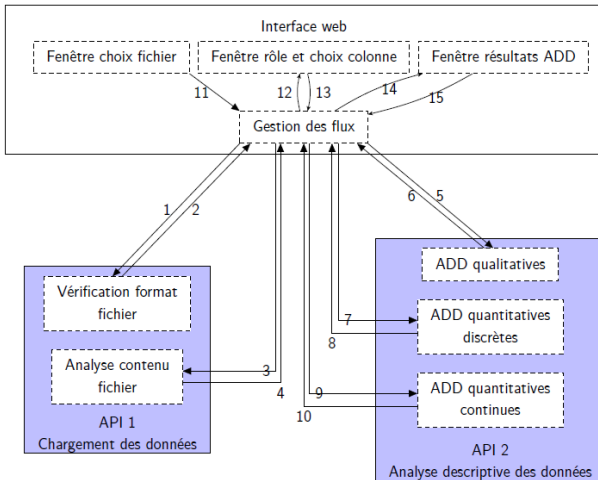
Vérification format fichier :

- format csv
- ouverture en lecture
- texte brut non formaté

Analyse contenu fichier :

- **structure 1** : contenu du fichier et nb lignes et colonnes
- **structure 2** : type, rôle et données erronées des colonnes

Exigences fonctionnelles



Légende :

Famille

Module

informations transmises

API 1 : Chargement des données

Vérification format fichier :

- format csv
- ouverture en lecture
- texte brut non formaté

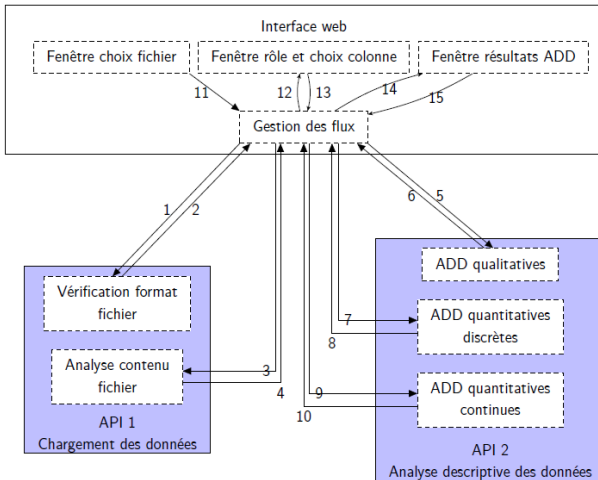
Analyse contenu fichier :

- **structure 1** : contenu du fichier et nb lignes et colonnes
- **structure 2** : type, rôle et données erronées des colonnes

API 2 : Analyse descriptive des données

- **Données** : Données d'une colonne
- **Retour** : Informations statistiques et représentations graphiques
- **ADD quantitatives continues** : Discrétisation des valeurs

Exigences fonctionnelles



Légende :

Famille

Module

informations transmises

Interface web

Gestion des flux :

- Lancement de l'application
- Flux d'exécution : Branchements et arrêts de l'application
- Flux de données : Rôle d'interface pour communiquer les données

Interface web

Gestion des flux :

- Lancement de l'application
- Flux d'exécution : Branchements et arrêts de l'application
- Flux de données : Rôle d'interface pour communiquer les données

Fenêtre choix fichier :

- Choix du fichier : Parcours du système de fichier - Drag Drop
- Validation du choix

Interface web

Gestion des flux :

- Lancement de l'application
- Flux d'exécution : Branchements et arrêts de l'application
- Flux de données : Rôle d'interface pour communiquer les données

Fenêtre choix fichier :

- Choix du fichier : Parcours du système de fichier - Drag Drop
- Validation du choix

Fenêtre rôle et choix colonne :

- Titre - Nb de lignes et colonnes - Echantillon
- Données erronées (numéro ligne, type attendu - valeur reçue)
- Navigation
- Sélection colonne : attribution nom et lancement de l'analyse

Interface web

Gestion des flux :

- Lancement de l'application
- Flux d'exécution : Branchements et arrêts de l'application
- Flux de données : Rôle d'interface pour communiquer les données

Fenêtre choix fichier :

- Choix du fichier : Parcours du système de fichier - Drag Drop
- Validation du choix

Fenêtre rôle et choix colonne :

- Titre - Nb de lignes et colonnes - Echantillon
- Données erronées (numéro ligne, type attendu - valeur reçue)
- Navigation
- Sélection colonne : attribution nom et lancement de l'analyse

Fenêtre résultats ADD :

- **Affichage** : Informations statistiques et représentations graphiques
- **Exportation** : Sauvegarder les résultats en local

- 1 Introduction
- 2 Motivations du projet
- 3 Contraintes
- 4 Exigences fonctionnelles
- 5 Exigences non fonctionnelles**
- 6 Autres aspects
- 7 Conclusion

Interface utilisateur : Interface simple, agréable, intuitive et interactive en anglais.

Interface utilisateur : Interface simple, agréable, intuitive et interactive en anglais.

Utilisabilité : Simple d'utilisation et facile à comprendre.

Interface utilisateur : Interface simple, agréable, intuitive et interactive en anglais.

Utilisabilité : Simple d'utilisation et facile à comprendre.

Performance : Réponse fluide et affichage des résultats de l'ordre de la minute.

Interface utilisateur : Interface simple, agréable, intuitive et interactive en anglais.

Utilisabilité : Simple d'utilisation et facile à comprendre.

Performance : Réponse fluide et affichage des résultats de l'ordre de la minute.

Précision et exactitude : Calculs très précis pour la Moyenne et la Variance. Sinon le plus précis possible.

Interface utilisateur : Interface simple, agréable, intuitive et interactive en anglais.

Utilisabilité : Simple d'utilisation et facile à comprendre.

Performance : Réponse fluide et affichage des résultats de l'ordre de la minute.

Précision et exactitude : Calculs très précis pour la Moyenne et la Variance. Sinon le plus précis possible.

Maintenabilité : Doit pouvoir être maintenu par ses utilisateurs finaux ou par des développeurs. Permettre l'insertion d'éventuels API supplémentaires.

Interface utilisateur : Interface simple, agréable, intuitive et interactive en anglais.

Utilisabilité : Simple d'utilisation et facile à comprendre.

Performance : Réponse fluide et affichage des résultats de l'ordre de la minute.

Précision et exactitude : Calculs très précis pour la Moyenne et la Variance. Sinon le plus précis possible.

Maintenabilité : Doit pouvoir être maintenu par ses utilisateurs finaux ou par des développeurs. Permettre l'insertion d'éventuels API supplémentaires.

Sécurité : Accès à partir d'un navigateur web. Manipulation des données fiables. Pas de manipulation de données à caractère personnel.

Plan

- 1 Introduction
- 2 Motivations du projet
- 3 Contraintes
- 4 Exigences fonctionnelles
- 5 Exigences non fonctionnelles
- 6 Autres aspects**
- 7 Conclusion

- **Question ouverte** sur l'esthétisme et présentation des résultats.

- **Question ouverte** sur l'esthétisme et présentation des résultats.
- **Choix du langage : Python** compatible avec les applications web et pour les calculs scientifique.
HTML et **CSS** : Présentation et mise en formes des pages.

- **Question ouverte** sur l'esthétisme et présentation des résultats.
- **Choix du langage** : **Python** compatible avec les applications web et pour les calculs scientifique.
HTML et **CSS** : Présentation et mise en formes des pages.
- **Tâche à faire** : Spécifications, Développement de l'application et Compte-rendu.

- **Question ouverte** sur l'esthétisme et présentation des résultats.
- **Choix du langage** : **Python** compatible avec les applications web et pour les calculs scientifique.
HTML et **CSS** : Présentation et mise en formes des pages.
- **Tâche à faire** : Spécifications, Développement de l'application et Compte-rendu.
- **Contrôle de la finalisation** : Test Unitaires, de validation et acceptation.

- **Question ouverte** sur l'esthétisme et présentation des résultats.
- **Choix du langage** : **Python** compatible avec les applications web et pour les calculs scientifique.
HTML et **CSS** : Présentation et mise en formes des pages.
- **Tâche à faire** : Spécifications, Développement de l'application et Compte-rendu.
- **Contrôle de la finalisation** : Test Unitaires, de validation et acceptation.
- **Estimation des coûts** de 565 lignes de code.

- **Question ouverte** sur l'esthétisme et présentation des résultats.
- **Choix du langage** : **Python** compatible avec les applications web et pour les calculs scientifique.
HTML et **CSS** : Présentation et mise en formes des pages.
- **Tâche à faire** : Spécifications, Développement de l'application et Compte-rendu.
- **Contrôle de la finalisation** : Test Unitaires, de validation et acceptation.
- **Estimation des coûts** de 565 lignes de code.
- **Répartition** de 3 modules chacun.

- **Question ouverte** sur l'esthétisme et présentation des résultats.
- **Choix du langage** : **Python** compatible avec les applications web et pour les calculs scientifique.
HTML et **CSS** : Présentation et mise en formes des pages.
- **Tâche à faire** : Spécifications, Développement de l'application et Compte-rendu.
- **Contrôle de la finalisation** : Test Unitaires, de validation et acceptance.
- **Estimation des coûts** de 565 lignes de code.
- **Répartition** de 3 modules chacun.
- **Documentation utilisateur** fournie.

Plan

- 1 Introduction
- 2 Motivations du projet
- 3 Contraintes
- 4 Exigences fonctionnelles
- 5 Exigences non fonctionnelles
- 6 Autres aspects
- 7 Conclusion**

Deux langages potentiels Java et Python. Notre choix s'est finalisé sur Python.

Deux langages potentiels Java et Python. Notre choix s'est finalisé sur Python.

- **Python** : Langage préconisé par les startups pour construire des applications.

Deux langages potentiels Java et Python. Notre choix s'est finalisé sur Python.

- **Python** : Langage préconisé par les startups pour construire des applications.
- **Java** : Utilisé par les entreprises ayant les moyens pour le développement et la maintenance du système.

Deux langages potentiels Java et Python. Notre choix s'est finalisé sur Python.

- **Python** : Langage préconisé par les startups pour construire des applications.
- **Java** : Utilisé par les entreprises ayant les moyens pour le développement et la maintenance du système.

Difficultés rencontrées : Apprendre une nouvelle démarche et pratique dans un projet à grande ampleur.