Cahier des Charges

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 mars 2017

Table des matières

1 Motivations du projet

1.1 But du projet

1.1.1 Contexte du projet

De nos jours, les masses de données collectées sont de plus en plus importantes. L'objectif principal de cette collecte de données est d'en extraire une valeur ajoutée. Or, ces données à l'état brut sont difficilement exploitables dû à leur volume et à leur complexité.
Notre produit correspond au travail indispensable d'analyse de ces données, afin de faciliter leur exploitation.

1.1.2 Objectif du projet

Ce projet a pour but de fournir aux utilisateurs une application qui se chargera d'une part de structurer les données, les analyser et les visualiser, et d'autre part de préparer ces données pour un chargement via des API.

1.2 Parties prenantes

1.2.1 Maître d'ouvrage

Notre interface de contrôle, de chargement, et d'analyse de données est développée pour l'entreprise **DCbrain**.
Le projet a été lancé en collaboration avec l'**UVSQ**.

1.2.2 Client

DCbrain est également l'entreprise qui va bénéficier des paquets finaux après leur développement.

1.2.3 Autre partie prenante

Les industriels clients de DCbrain sont des parties prenantes indirectes. Notre travail doit pouvoir être utilisé par DCbrain pour renforcer leur application.

1.3 Utilisateurs du produit

En premier lieu, l'application va servir aux membres de DCbrain étant donné que leurs clients industriels (Total, ERDF,...) leur fournissent les fichiers CSV, afin qu'ils puissent appliquer des analyses descriptives sur les données dans le but de repérer des anomalies sur les réseaux de ces derniers. Puis en second lieu, il se pourrait que DCbrain veuille déployer l'application pour ses clients, dans ce cas les membres de ces organismes deviendront des utilisateurs.

On peut supposer l'absence de restrictions d'utilisations, étant donné l'absence d'exigence de la part du maître d'œuvre sur ce sujet.

2 Contraintes sur le Projet

2.1 Contraintes imposées

2.1.1 Contraintes sur la conception :

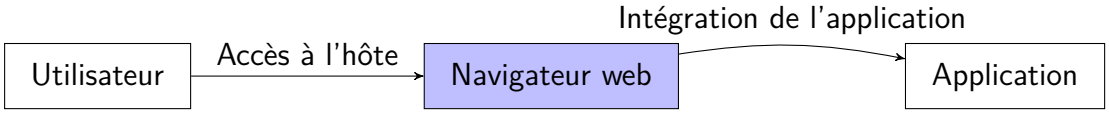
Contrainte	Fiche
1. Le produit doit fournir une application web	[style=unboxed,leftmargin=0.2cm] **Description :** notre produit sera une application fonctionnant sur un navigateur web, appelée ***applet***. **Justification :** assure une très grande portabilité et fournit à l'utilisateur une interface interactive. **Critère de satisfaction :** on peut lancer l'application sur un navigateur web.

2. Le produit doit être développé avec un langage de programmation compatible avec l'analyse de données	[style=unboxed,leftmargin=0.2cm]**Description :** le langage de programmation choisi doit inclure une bibliothèque qui permet d'analyser les données (taches de data mining et de machine Learning). **Justification :** permettre une analyse de données la plus efficace possible. **Critère de satisfaction :** on peut analyser les données de manière efficace.
3. Le produit doit fournir une API d'analyse de données en sortie	[style=unboxed,leftmargin=0.2cm]**Description :** l'application doit intégrer des API d'analyse descriptive de données qui pourront être livrées en sortie au client. **Justification :** permettre une réutilisabilité des fonctionnalités majeures du produit. **Critère de satisfaction :** on peut exporter une API d'analyse de données en sortie.

Choix des langages de programmation : les contraintes 1 et 2 nous permettent de nous fixer sur le choix du langage : ***Java EE***. D'une part, ce langage est orienté pour le développement d'applications web robustes et distribuées, déployées et exécutées sur un serveur d'applications. D'autre part, ce langage qui est basé sur Java est doté de plusieurs modules d'analyse de données, dont l'outil ***Weka*** par exemple.
De plus, l'applet Java doit être intégrée dans une page web pour être exécutée : on va donc avoir recours à des langages de balisage comme ***HTML*** pour la présentation et ***CSS*** pour la mise en forme.

2.1.2 Environnement de fonctionnement :

Le produit va fournir une application web ou ***applet***. L'environnement technologique de ce genre d'application sont les navigateurs web. Ces navigateurs web jouent le rôle d'interface entre l'utilisateur et l'application.



Le produit doit donc être compatible avec tous les navigateurs web bureau (pas de version mobile exigée) prenant en charge les fonctionnalités des dernières versions des langages (***HTML5*** et ***CSS3***), par exemple ***Google Chrome*** et ***Mozilla Firefox***.

2.1.3 Applications partenaires :

Le produit va fournir une API en sortie. Il doit donc prendre en compte de l'environnement d'intégration de cette API, c'est à dire que cette API doit être compatible avec les outils du client, l'entreprise DCbrain.

2.1.4 Temps dont disposent les développeurs du projet :

Le produit doit être rendu avant le 26/05/2017.

2.1.5 Budget du projet :

La réalisation du produit n'exige pas de ressources financières. Aucun budget n'est donc nécessaire.

2.2 Glossaire et conventions de dénomination

[style=unboxed,leftmargin=0.2cm]**API :** *Application Programming Interface*, constituent les paquets utilisables par les développeurs (intégrés), qu'on va livrer au client en plus de l'application elle-même. **ADD :** *Analyse Descriptive de Données*, fonctionnalité d'analyse de description statistique des données.

2.3 Faits et hypothèses déterminants

2.3.1 Facteurs influençant le produit, mais qui ne sont pas des contraintes imposées sur les exigences :

Rien à mettre à priori, mettez des trucs si vous en trouvez

2.3.2 Hypothèses que l'équipe fait sur le projet :

rien pour l'instant

3 Exigences fonctionnelles

3.1 Périmètre de l'ouvrage

3.1.1 Diagramme de contexte : flux élémentaires

Le diagramme ci-dessous définit le contexte du travail : Diag à insérer ici

3.1.2 Contexte du travail

— développeurs : développeur souhaitant apporter de la VA à son produit / rééviter de coder lui-meme blabla ...

— industriels : utilisateurs sans connaissances en informatique ni en ADD présupposées => simple blabla...

3.2 Périmètre de l'œuvre

3.2.1 Diagramme de cas d'utilisation

Le diagramme ci-dessous définit le perimètre d'utilisation de notre travail : Diag à insérer ici

3.2.2 Descritpion sommaire des cas d'utilisation

L'utilisation de notre travail va se limiter à deux cas, un pour chaque type d'utilisateur :

— développeurs : il prend les fct, exporte les vals, intègre notre travil dans l'appli qu'il développe blabla ...

— industriels : lance l'applet cherche un fichier blabla ...

3.3 Exigences fonctionnelles et exigences sur les données

3.3.1 Exigences fonctionnelles

Voir organigramme - on liste tout (numero aux exigences) justification et critere de satisfaction

3.3.2 Exigences sur les données

Début de spécification : csv formaté, format d'exportation, vals tmp stockées pour les calculs + cardinalitées

4 Exigences non fonctionnelles

4.1 Interface utilisateur du produit

4.1.1 Exigences d'apparence

Le produit devra adopter une apparence simple, agréable et devra être facile à comprendre dès sa première prise en main. Le maitre d'œuvre a émis le souhait d'avoir de l'Anglais comme langue utilisée pour l'affichage textuel.

4.1.2 Exigences de style

L'application s'adresse à un public professionnel, alors l'une des exigences concernant le style de l'interface est qu'il devra opter pour un style classique, au goût du jour et professionnel. L'affichage des résultats et autre devra susciter la confiance auprès des utilisateurs.

4.2 Utilisabilité

Le produit devra être simple d'utilisation et facile à comprendre dès sa première prise en main, afin d'éviter une formation concernant la manipulation du produit pour les futurs utilisateurs. Il aidera également l'utilisateur à analyser une quantité importante de données.

4.3 Exigences de performance

Dans le template y a pas cette partie

mais y a "Adéquation du produit avec son environnement" qui découle de la partie "Contraintes/Enviror

Les exigences de performance du futur produit concerne concerne la latence acceptable