# Concevoir une application informatique

# Projet Raleyd

DIF2 Bloc 2 IOT Groupe 3

|  |  |  |  |
| --- | --- | --- | --- |
| **VERSIONS** | | | |
| **Nom – Fonction** | **Intitulé** | **Date** | **version** |
| Jean Rafael Mendes – Etudiant | Sommaire + Contexte analyse du besoin et preparation | **09/11/2022** | **1.0** |
| Loyd Simon -- Etudiant | Base de données + serveur | 15/11/2022 | 1.1 |
|  |  |  |  |

Sommaire

[Concevoir une application informatique 1](#_Toc119331843)

[Projet Raleyd 1](#_Toc119331844)

[I. INTRODUCTION, ANALYSE DU BESOIN & PREPARATION 4](#_Toc119331845)

[a. Introduction 4](#_Toc119331846)

[b. Analyse fonctionnelle 4](#_Toc119331847)

[c. Schéma du besoin ou « Bête à corne » 5](#_Toc119331848)

[d. Diagramme de séquence 6](#_Toc119331849)

[e. Reformulation du besoin 9](#_Toc119331850)

[f. Git 9](#_Toc119331851)

[g. PyCharm 9](#_Toc119331852)

[h. HTML/CSS 10](#_Toc119331853)

[i. SQL 10](#_Toc119331854)

[j. Figma 10](#_Toc119331855)

[k. Jmerise 11](#_Toc119331856)

[l. Jira 12](#_Toc119331857)

[II. BASE DE DONNEES 12](#_Toc119331858)

[a. Schéma MCD 12](#_Toc119331859)

[b. Schéma MLD 12](#_Toc119331860)

[c. Transformation MLD en MPD 12](#_Toc119331861)

[III. SERVEUR WEB 12](#_Toc119331862)

[a. Script python 12](#_Toc119331863)

[b. Script base de données 12](#_Toc119331864)

[IV. PARTIE BACK\_END 12](#_Toc119331865)

[a. Création requêtes SQL 12](#_Toc119331866)

[b. Récupération, conversion et sauvegarde des données dans la base de données 12](#_Toc119331867)

[V. INTERFACE WEB 12](#_Toc119331868)

[a. Wireframe 12](#_Toc119331869)

[b. Maquette 12](#_Toc119331870)

[c. Création de l’interface responsive 12](#_Toc119331871)

[VI. PREPARATION DE LA SOUTENANCE 12](#_Toc119331872)

[a. Rédaction PowerPoint 12](#_Toc119331873)

[b. Rédaction de ce rapport 12](#_Toc119331874)

[VII. CONCLUSION 12](#_Toc119331875)

## INTRODUCTION, ANALYSE DU BESOIN & PREPARATION

### Introduction

« La société "Lepetit" créé en 1992, qui a fait 1,275 milliard d'€ de CA en 2017 ayant racheté la startup "Atmos" qui commercialise des objets connectés pour la maison. "Lepetit" envisage la modernisation d'un produit "phare" de la startup "Atmos" qui depuis le rachat ne fait que perdre en popularité. »

Atmos vend des stations météorologiques fonctionnant sur un modèle Cloud (données centralisées sur un serveur).

Notre groupe vient d'être recruté dans cette importante compagnie et nous sommes dédiée à la création d'un prototype applicatif et électronique. Le groupe est composé de trois étudiants :

* Loyd Simon, le chef de projet
* Jean Rafael Mendes
* Leopold Ardouin

Notre direction composé d’un directeur technique et d’un directeur administratif et financier nous accordera un créneau d’une vingtaine de minutes, afin de présenter notre prototype fonctionnel (application et hardware), nous veillerons à faire une présentation brève mais “efficace et commerciale” de notre solution en mettant en avant les points forts et faibles de ce projet, ainsi que les points de vigilance ou améliorations que nous proposerons avant de passer à une phase d’industrialisation.

### Analyse fonctionnelle

La première chose à faire dans notre rôle de créateur/concepteur de produits, est de faire l’analyse fonctionnelle pour optimiser la reconception du produit en nous appuyant sur les fonctions que doit réaliser le produit.

L’analyse fonctionnelle que nous avons faite, a eu pour objectif de :

* Reformuler le besoin
* Prendre en compte les besoins utilisateur
* L’utilisation d’outils approprié pour formaliser le besoin
* Prioriser les tâches à effectuer

Elle nous a permis d’éviter les pièges classiques de la conception de produit tel que :

* Éviter le hors-sujet, le manque d’objectivité
* Mauvaise gestion des priorités

A savoir que dans le milieu professionnel, les premières étapes de l’analyse fonctionnelle concernent tous les acteurs d’un même projet et sont faites en relations avec tous les intervenants. C’est seulement dans un deuxième temps que l’analyse fonctionnelle devient technique, et oriente les concepteurs (notre groupe) vers des solutions techniques.

### Schéma du besoin ou « Bête à corne »

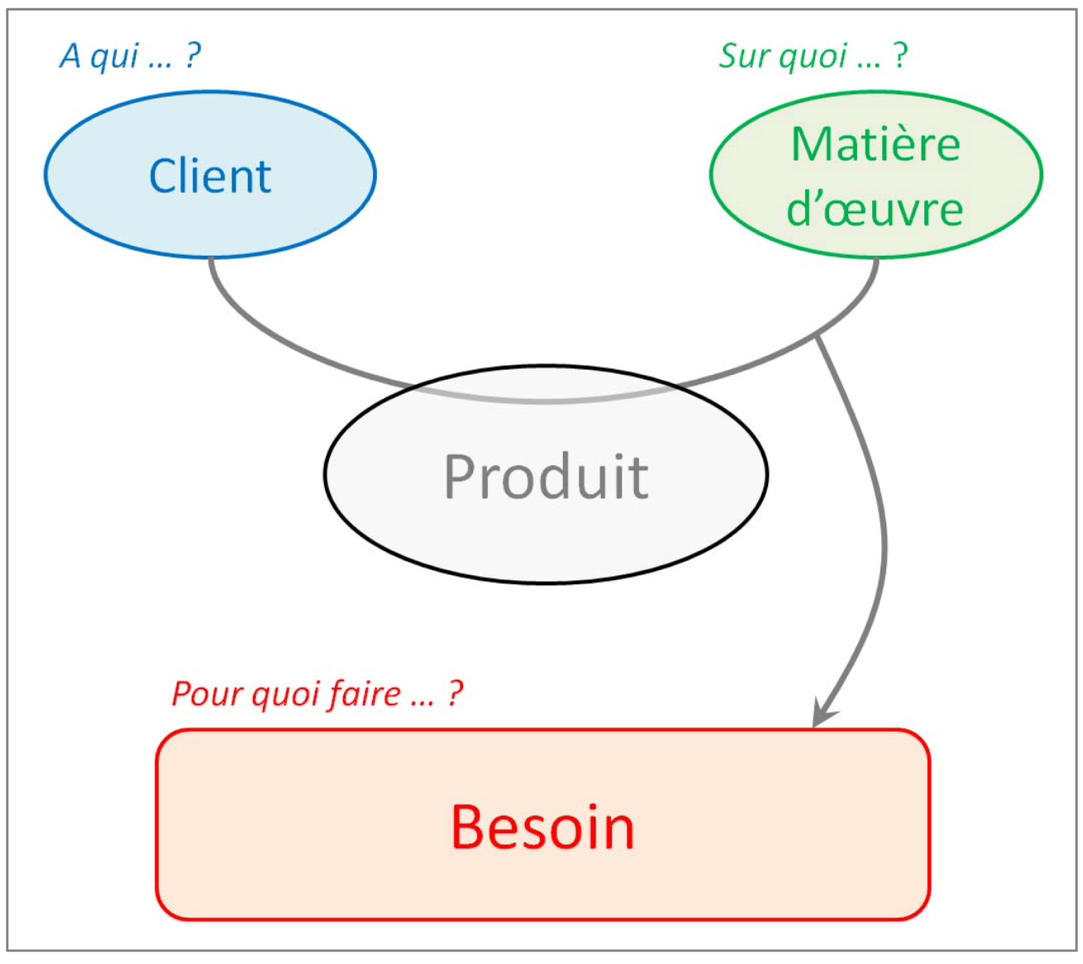


Figure 1 Bête à corne du cours de Mr Eric Artigala

Pour répondre aux trois questions présente sur le schéma « A qui … ? », « Sur quoi … ? » et « Pour quoi faire … ? » nous avons utilisé l’énoncé du besoin suivant :

« Le produit rend service au client en agissant sur la matière d’œuvre pour satisfaire le besoin »

Dans notre cas nous avons obtenu le résultat suivant pour le schéma du besoin :

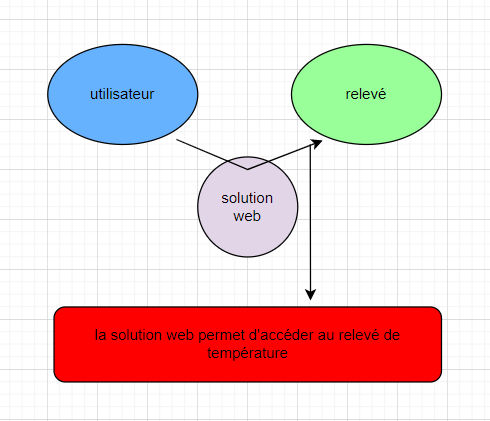


Figure 2 Bête à corne du projet Raleyd

« Le solution web rend service à l’utilisateur en agissant sur le relevé en satisfaisant sa demande d’accès »

### Diagramme de séquence

Un diagramme de séquence UML (unified modeling language) montre comment un ensemble d’objets interagissent dans un processus et dans le temps. Cela affiche les messages qui passent entre des participants et des objets dans le système, ainsi que l’ordre dans lequel ils se produisent. Le diagramme est conçu comme une méthode normalisée de visualisation dans les domaines du développement logiciel et en conception orientée objet.

Dans le schéma que nous avons créé (page suivante) nous avons inclus les participants et objets suivant :

* L’utilisateur ou User
* Le site web ou interface web
* La partie serveur web, la partie backend
* La base de données
* L’API
* Internet
* La box, la passerelle et la sonde

Une image contenant texte, intérieur, carrelé

Description générée automatiquement

Figure 3 Diagramme de séquence du projet Raleyd

### Reformulation du besoin

Après avoir fait l’analyse fonctionnelle complète nous avons retenu que le besoin est la conception d’une maquette fonctionnelle de l'application web de visualisation des données, ainsi qu'un prototype fonctionnel d'une sonde et du serveur de collecte des données météorologiques.

### Git

Par soucis d’efficience et de gain de temps, nous avons choisi de créer un repository (dépôt) sur GitHub accessible pour tout le monde (open-source) et modifiable uniquement par le groupe. Cela permettra à la direction de consulter le code source ainsi que nos documents la veille de la présentation de notre travail.

Le lien : https://github.com/Rafaelllllllll/ProjetMeteo

Les avantages de l’utilisation de Git dans le développement d’un projet sont multiples. Cela permet de créer un dépôt distant du code source et des autres éléments émanent de ce même projet. Contrairement au simple dépôt local présent dans le dossier d’un collaborateur, celui-ci peut-être consulter et mis à jour en même temps par toute l’équipe. Quand, dans un schéma « classique », les collaborateurs doivent s’échanger les fichiers via clés USB ou disques dur.

Tableau 1 Fonctionnement projet sans et avec Git

|  |  |
| --- | --- |
| Fonctionnement sans Git | Fonctionnement avec Git |
| Chaque utilisateur travaille sur ses fichiers ou en lecture seule sur un serveur | Utilisation de branche pour travailler en même temps |
| Faire des copier/coller et correction manuellement | Fusion des fichiers |
| Fichier inaccessible | Open source, consultable par tout le monde, peut servir de « portfolio » |

### PyCharm

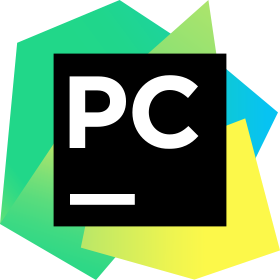


Figure 4 Logo PyCharm

Pour mener à bien notre projet, nous avons choisi comme langage de développement backend, python. En suivant cette logique, l’idéal pour nous a été d’adopter l’IDE PyCharm. Pycharm est un environnement de développement utilisé principalement pour la programmation en python. Il contient un débogueur graphique et il permet aussi l’analyse de code. L’intégration de logiciel de gestion de versions est présente, dans notre cas Git est compatible. Il permet également le développement web.

### HTML/CSS

Une image contenant texte, trousse de secours

Description générée automatiquement

Figure 5 Logo CSS/HTML

Le HTML « HyperText Markup Language » ou « langage de balisage hypertexte » en français est un langage de balisage, c’est-à-dire un langage qui va nous permettre de définir les différents contenus d’une page. Le rôle du HTML est crucial puisqu’il va être le langage pour indiquer aux navigateurs de quoi est constituée chaque page et ce qu’ils doivent afficher. Grâce au HTML, on va par exemple pourvoir indiquer que tel contenu est un texte qui n’est qu’un paragraphe ( <p> ), que tel autre contenu est un texte qui est un titre de niveau 1 dans notre page ( <h1> ), que tel autre contenu est une liste ( <ul>, <li> ), etc.

Le CSS « Cascading StyleSheets » ou « feuilles de styles en cascade » en français a été créé en 1996, soit 5 ans après le HTML. Il va servir à mettre en forme les différents contenus définis par le HTML en leur appliquant des styles. On va ainsi par exemple pouvoir définir la taille, la couleur ou l’alignement de certains contenus HTML.

### SQL

Le SQL (Structured Query Language) est un langage permettant de communiquer avec une base de données. Ce langage informatique est notamment très utilisé par les développeurs web pour communiquer avec les données d’un site web. Nous avons utilisé ce langage pour faire le lien entre notre interface web et notre base de données phpMyAdmin mais aussi pour créer la base de données.

### Figma

Notre équipe n’ayant pas accès à une licence adobe pour le maquettage de notre interface web, nous avons fait le choix de prendre l’outil Figma. Figma est un éditeur graphique et un outil de prototypage gratuit. Il est aussi utilisé comme support de présentation et de collaboration en temps réel.

Une image contenant texte

Description générée automatiquement

Figure 6 Screenshot de la maquette du projet Raleyd

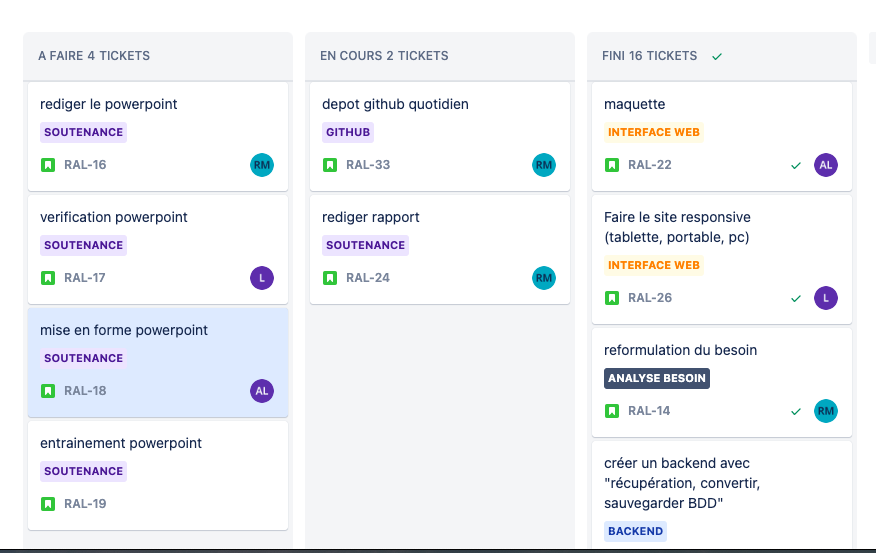
### Jmerise

Merise est une méthode d'analyse, de conception et de gestion de projet informatique.

Jmerise est un logiciel dédié à la modélisation des modèles conceptuels de donnée pour Merise. Ils permet les relations réflexives, la généralisation et la spécialisation des entités. Il peut générer le MLD et le script MySQL de création d’une base de données.

### Jira

Pour l’organisation et la répartition de nos tâches, nous avons utilisé l’outil Jira. Il est un puissant outil de gestion du travail de groupe. Il permet de distribuer et répartir les tâches pour chaque collaborateur via des tickets et des statuts de ticket : à faire, en cours ou terminé(e). Les délais peuvent être indiquer, et des alertes personnalisable sont possible pour les fins de délais.

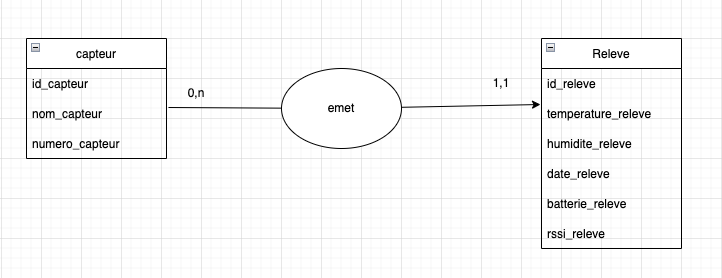


## BASE DE DONNEES

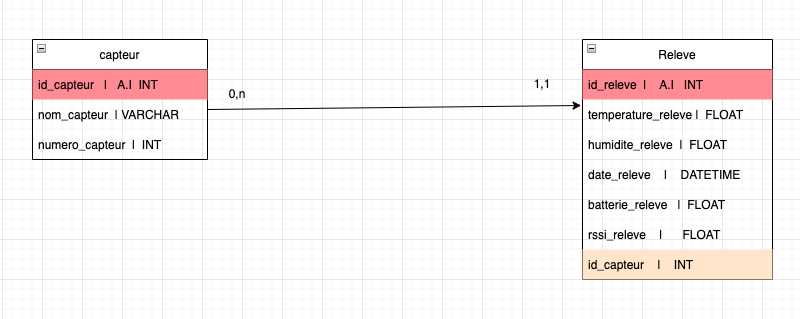
### Schéma MCD

Le modèle conceptuel de données sert à conceptualiser l'application. Il met en évidence deux éléments : les entités et les associations. Par ailleurs, Il nous a fallu reprendre le besoin de notre client et réfléchir à toutes les données nécessaires au bon fonctionnement de l’application.

Le MCD ou Modèle conceptuel des données est un modèle et une étape importante permettant de concevoir par la sémantique les entités, les relations et les règles de gestion nécessaires pour ensuite passer au MLD.



### Schéma MLD



Tenant compte du cas de cardinalité (0,n) - (1,1) nous avons ajouté une clé secondaire dans le tableau relevé pour faire référence au capteur.

### Transformation MLD en MPD

Après avoir réalisé la représentation sémantique de notre modèle, nous avons réalisé la modèle physique ou MPD qui sera la représentation logique de chaque tableau à concevoir dans la base de données.

Un MPD permet de créer les scripts dont vous avez besoin pour développer votre base de données.

## SERVEUR WEB

### Serveur

Le système sera déployé sur un serveur, ce qui permettra de déployer un site web, une base de données et les scripts nécessaires au système.

Via un script, le serveur va faire une requête à l’API pour récupérer les données des capteurs sous format JSON et convertira ces données dans un format lisible.

( Images en annexes )

## PARTIE BACK\_END

### Création requêtes SQL

### Récupération, conversion et sauvegarde des données dans la base de données

## INTERFACE WEB

### Wireframe

### Maquette

### Création de l’interface responsive

## PREPARATION DE LA SOUTENANCE

### Rédaction PowerPoint

### Rédaction de ce rapport

## CONCLUSION

[Figure 1 Bête à corne du cours de Mr Eric Artigala 5](#_Toc119329252)

[Figure 2 Bête à corne du projet Raleyd 6](#_Toc119329253)

[Figure 3 Diagramme de séquence du projet Raleyd 9](#_Toc119329254)

[Figure 4 Logo PyCharm 10](file:////Users/rafaelmendes/Desktop/CESI/Concevoir%20une%20application%20informatique/DIF2%20Bloc%202%20IOT%20Groupe%203.PDF.docx#_Toc119329255)

[Figure 5 Logo CSS/HTML 10](file:////Users/rafaelmendes/Desktop/CESI/Concevoir%20une%20application%20informatique/DIF2%20Bloc%202%20IOT%20Groupe%203.PDF.docx#_Toc119329256)

[Figure 6 Screenshot de la maquette du projet Raleyd 11](#_Toc119329257)

[Tableau 1 Fonctionnement projet sans et avec Git 9](#_Toc119329265)