**LORANS-CANO Gurvan**

**LE BERRE Samuel**

**BOUDET Alexandre**

**MARTEL Benoît**

**Département INFORMATIQUE 1ère année – Groupe A**

**Année 2016-2017**

**M2107 - PROJET DE PROGRAMMATION**

**Cahier des charges prévisionnels**

Création d’un éditeur de base de données



**IUT de Vannes**

M. LEFEVRE, professeur référent

Table des matières

[I- Contexte du projet 2](#_Toc479279327)

[II – Fonctionnalités du logiciel 3](#_Toc479279328)

[A) De base 3](#_Toc479279329)

[B) Optionnelles 3](#_Toc479279330)

[III – Diagramme de GANTT 1](#_Toc479279331)

[A) Version de base 1](#_Toc479279332)

[B) Version optionnelle 4](#_Toc479279333)

[IV – MPM 6](#_Toc479279334)

[A) Version de base 6](#_Toc479279335)

[B) Version optionnelle 6](#_Toc479279336)

# **Contexte du projet**

Le but de ce projet est de réaliser un éditeur de base de données accessible via Java (JDBC). Ce dernier doit comporter une interface texte ainsi qu’une interface graphique avec des fonctionnalités de base que l’on peut retrouver dans de nombreux logiciel de traitement de texte ou autres (Enregistrer, Enregistrer sous, Ouvrir, Exporter sous un autre format). Par ailleurs il faudra implémenter les fonctionnalités de base d’une base de données (BD, métadonnées, données : tables, champs, enregistrement) et d’éventuelles statistiques. Le logiciel doit assurer le fait de pouvoir éditer, insérer, supprimer des tables ou des enregistrements. L’utilisateur pourra rechercher une donnée pour pouvoir la remplacer. Dans la mesure où l’utilisateur manipulera des données nous devrons établir une sécurité en ajoutant un login et un mot de passe. Et enfin des fonctionnalités optionnelles pourront être ajoutés tels que : un jeu pour apprendre le SQL, mini-quiz, etc.)

**Bêtes à cornes du projet :**

UTILISATEUR DE BASES DE DONNEES

FICHIERS SQL

EDITEUR DE BASES DE DONNEES

MANIPULER UNE BASE DE DONNEES

A qui rend-il service ?

Sur quoi agit-il ?

Dans quel but ?

# **II – Fonctionnalités du logiciel**

## De base

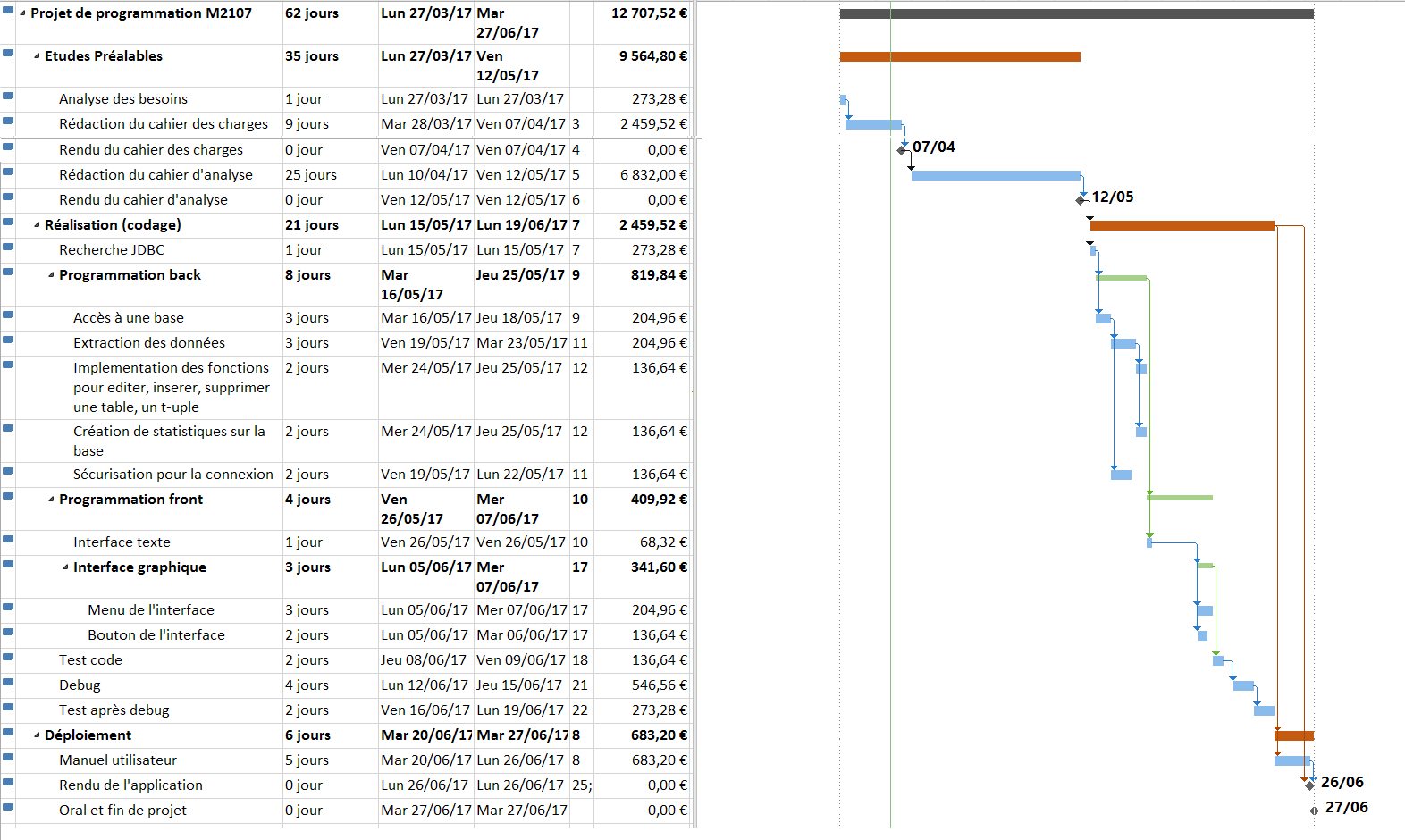
* Accéder à une base de données depuis Java
* Posséder une interface texte (console où sera affiché le résultat)
* Posséder une interface graphique (menu, boutons …)
* Ouvrir, enregistrer, enregistrer sous, nouveau fichier (stocker dans des fichiers où sont enregistré les requêtes et permettre leur lecture etc.)
* Visualiser des informations (BD, métadonnées, données, table, champs, fichier d’enregistrement) et statistiques (pourcentage pour des valeurs selon les requêtes)
* Édition, insertion, suppression d’une table, d’un t-uple et enregistrement automatique
* Sécurité pour la connexion à la base de données (attribuer droits aux utilisateurs)
* Posséder un champ de saisie de texte (pour l’écriture des requêtes SQL)

## Optionnelles

* Mini quiz pour apprendre le SQL
* Pouvoir créer une table juste avec un enchaînement de clic sur des boutons et de saisie de nom d’attribut
* Onglet aide (nouvelle fenêtre qui s’ouvre avec des commandes de bases, des conseils et autres)
* Représentation graphique (schéma relationnel, diagramme de classes)
* Changer le fond/les couleurs (esthétisme)
* Exporter sous un autre format
* Imprimer le fichier SQL qui a été chargé

# **III – Diagramme de GANTT**

## Version de base



Le projet de programmation durera 62 jours ouvrés à partir du lundi 27 Mars qui marque le début du projet.

La première étape de ce projet est l’étude préalable qui nous est demandée de réaliser afin d’évaluer les besoins et les fonctionnalités à apporter à notre application. Cette étape nécessitera 2 rendus, le cahier des charges et le cahier d’analyse. Le cahier des charges définit ce qui sera produit dans la globalité tandis que le cahier d’analyse est beaucoup plus précis dans sa description du programme qui sera rendu. Cette période dure 35 jours et se finira le vendredi 12 Mai.

Suit alors la réalisation du logiciel qui commencera le lundi 15 Mai et finira le lundi 19 Juin pour une durée totale de 21 jours.

Cette étape commencera par des recherches sur le JDBC (Java DataBase Connectivity), qui est un langage de java spécifique aux bases de données, que nous ne connaissons pas mais qui nous est vital pour la poursuite du travail.

Après cette phase commencera la programmation du back c’est-à-dire tous le programme qui s’exécutera derrière l’interface graphique.

* On commencera par créer un accès à la base de données, nous devrons donc créer une base de données à laquelle nous fournirons un accès à l’aide d’un profil.
* Puis nous programmerons l’extraction des données, on va faire en sorte de pouvoir récupérer toutes les données stockées dans la table afin de les utilisés dans le reste de notre logiciel.
* Ensuite on implémentera les fonctions qui permettent de créer, d’insérer et de supprimer une table ou un t-uple : peut-être à l’aide d’une fonction de JDBC.
* On utilise maintenant l’extraction des données afin de calculer des statistiques suivant les colonnes d’une table ou autres : il s’agira d’une requête exercée sur la base.
* On finira la programmation du back par la sécurisation de la connexion : il faut donner des droits aux comptes existant depuis la création de l’accès à une base afin qu’ils puissent modifier ou non une table, la supprimer ou même extraire ses données.

La programmation du font se déroulera juste après, le font représente ce que verra l’utilisateur du logiciel. Cette période de programmation commencera par :

* La réalisation d’une interface texte qui a pour but de nous renvoyer le résultat des requêtes dans un espace dédié à cet effet.
* La création d’une interface graphique :
  + L’élaboration d’un menu regroupant les interactions les plus basiques de chaque logiciel comme nouveau, ouvrir, enregistrer ou encore enregistrer sous. Ces fonctions sont essentielles au bon fonctionnement du programme car si on ne peut sauvegarder nos requêtes alors la table elle-même devient inutile.
  + Placement de boutons visant à améliorer la rapidité et la simplicité d’utilisation du logiciel.
  + Prévoir le redimensionnement de la fenêtre que l’on peut modifier à l’aide d’un clic de souris afin qu’il n’y ait pas de problème en cas de changement de proportions.

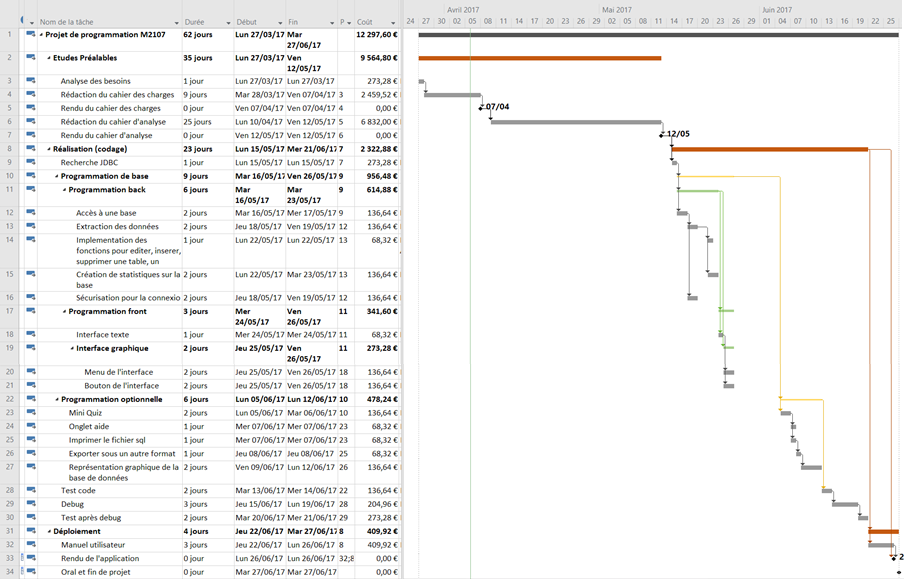
Maintenant que le logiciel est produit il faut :

* Faire des tests afin de trouver les fonctionnalités qui ne marche pas ou les erreurs qui se produisent lors de l’utilisation du logiciel.
* S’en suit une phase de débogage où l’on cherche à résoudre les bugs précédemment trouvés afin qu’il ne soit plus présent dans la version finale du logiciel.
* On procède à une seconde phase de tests qui n’est que la répétition de la première phase de tests après la correction des erreurs.

Le logiciel est maintenant prêt à être donné au client.

Vint l’étape du déploiement du logiciel, qui commence le 20 Juin. Cette étape consiste à la rédaction du manuel d’utilisation qui facilitera la prise en main et la compréhension de notre logiciel par l’utilisateur. Une fois le manuel rédigé, il faut rendre le logiciel et passer l’oral de fin de projet qui est une synthèse courte de 3 min sur notre expérience durant la conception d’un logiciel. Cette étape s’étend sur une durée de 6 jours.

## Version optionnelle



Le diagramme de version optionnel reprend toutes les actions du diagramme de base mais diffère par leur temps d’exécution ainsi la programmation du back et du font dure 9 jours (12 jours de base) et le déploiement ne dure plus que 4 jours (6 jours de base). La phase de débogage (correction des problèmes) sera de 4 jours soit un jour de plus. Ce gain de temps sera utilisé pour ajouter une phase à l’étape de réalisation, il s’agit de la phase de programmation optionnelle qui se déroulera après la programmation du font. L’étape durera 6 jours et durant ce temps on créera :

* Un mini quiz SQL qui vise à apprendre les bases du SQL à l’utilisateur tels que l’insertion, la création de table.
* L’ajout d’un bouton d’aide sur l’interface qui sera une sorte de tutoriel du logiciel avec des exemples, etc.
* La possibilité d’imprimer le fichier SQL qui est ouvert dans le logiciel.
* L’export sous un autre format : peut-être un fichier Excel ou autres formats inhabituel pour un fichier SQL.
* La représentation graphique de la base, il existe différentes façons de représenter une table (UML, etc.) il serait donc intéressant de les représenter pour l’utilisateur.

# **IV – MPM**

## Version de base

Fusion PDF

## Version optionnelle

Fusion PDF