**Rapport du Troisième livrable**

Table des matières

[**Arborescence du troisième livrable :** 1](#_Toc119272168)

[**Choix du livrable** 2](#_Toc119272169)

[**Troisième livrable** 3](#_Toc119272170)

# **Arborescence du troisième livrable :**

1. Analyse\_des\_besoins :
   * **Cahier\_des\_charges**
     + Un (troisième) cahier des charges minimal
     + Un cahier des charges cumulé
   * **Organisation :**
     + 1.WBS
     + 2.Les matrices d’antériorité
     + 3.GANTT
   * **Receuil\_des\_besoins**
     + Receuil\_des\_besoins
2. Spécification :
   * **Cas d’utilisation**
     + Cas\_d\_utilisation
     + Scénarii
   * Etude\_technologie\_web
3. Conception :
   * **Conception\_architecturale**
     + Conception\_bd\_architecturale
   * **Conception\_détaillée**
     + Conception\_bd\_détaillée
   * **Diagramme**
     + Diagramme\_composant-connecteur
     + Diagramme\_deploiement
   * Implementation\_monolithique\_technologies
4. Programmation :
   * **Bd**
     + Requete\_sql\_pour\_php
     + Requete\_Bd\_creation\_table (SIMFAST)
5. Test :
   * **Dossier\_de\_test**
     + Document\_test\_unitaire

# **Choix du livrable**

**Choix du troisième livrable**

Nous avons choisi dans ce troisième livrable de faire un processus de développement concentrer sur la base de données et sur le serveur à installer. Nous nous concentrons seulement sur ce point, nous retrouvons donc pas les autres dossiers comme Maquette\_web ou page\_html.

**Choix de l’arborescence du deuxième livrable**

Nous avons choisi cette méthode de livrable pour la cause suivante. Afin de mener à bien un projet informatique de cette envergure : la réalisation d’une application web. Nous devons rassembler les activités du génie pour respecter les critères de qualité. Nous avons donc proposé de faire plusieurs livrables (nous allons en dire davantage sur les livrables plus tard) avec une caractéristique : les dossiers on les a divisés en activité du génie logiciel : Analyse des besoins, Spécification, Conception, Programme, Test. Cela permet une meilleure clarté et aussi de suivre un ordre linéaire pour la validation de notre livrable.

**Choix du rendu en plusieurs livrables, Cycle de vie**

Notre équipe suit un processus de développement. Le processus de développement indique la forme dans laquelle les activités sont connectées entre elles. L’ordre dans lequel s’enchaînent les activités s’appelle le cycle de vie du produit logiciel.

Ici dans ce troisième livrable, il représente un cycle de vie du produit logiciel. Le produit logiciel est décomposer en plusieurs cycle de vie et sera compléter par de nouveaux livrables.

Cette partie du rapport est rédigé après avoir fait l’activité du génie logiciel : Analyse des besoins. Avec cette activité nous avons pu déterminer la méthode de décomposition de nos livrables qui nous définirons notre cycle de vie.

Nous avons proposé un cycle de vie en Cascade. (Voir figure) C’est un diagramme d’activité, utilisé pour montrer l’enchainement des tâches effectuées par notre équipe. Chaque livrable se basera sur cette méthode. Lorsque ces activités sont terminées on passera à la suivant dans un nouveau livrable.

Cette méthode de cascade nous permet d’avoir un contrôle sur chaque étape. Elle permet dès que possible de pouvoir créer les tests, ainsi lors du développement à chaque étape on réfléchit aux erreurs que l’on cherche. Une fois la programmation terminée nous exécutons les tests de validation qui vont venir valider les besoins fonctionnelles définit dans le cahier des charges et les demandes qui nous semble utile pour réaliser ce projet. On se trouve donc dans un cycle linéaire.

# **Troisième livrable**

**Analyse des besoins**

Le (troisème) cahier des charges minimal :

Nous proposons un troisième cahier des charges qui n’est pas complet et sera complété au fil du temps. Nous pouvons le retrouver dans le dossier document sur le gitlab du projet

Le cahier des charges cumulé :

Nous proposons un cahier des charges qui cumule les cahiers des charges des anciens livrables.

Organisation :

Dans le dossier organisation disponible sur le troisième livrable de notre dépôt GitLab, nous pouvons retrouver le diagramme WBS des activités à réaliser, les matrices d’antériorité et enfin le diagramme de GANTT pour pouvoir proposer un cycle de vie.

Recueil des besoins :

Nous listons et reformulons les exigences définit par le client de manière claires et concises tout en restant réaliste. Nous pouvons retrouver le recueil des besoins dans le dossier Document du dépôt GitLab.

**Spécification**

Cas d’utilisation :

Nous avons réalisé les cas d’utilisation.

Rapport scénarii :

Nous retrouvons la rédaction des scénarii que nous avons proposé

Etudes des technologies utilisées :

On définit toutes les technologies utilisées lors de ce projet.

**Conception**

Conception\_architecturale :

La conception architecturale permet de rendre compréhensible le modèle avec les interactions (relations) entre les tables de la base de données. Cela permet plus de faciliter au passage à la création de ses tables.

Dans la conception architecturale il y a la représentation des interactions entre toutes les pages que nous devons créer.

Conception\_détaillée :

La conception détaillée est un détaille de chaque table de la base de données. Elle permet de rendre pour chaque table un aperçu des grandes parties de la création de la base de données.

Diagramme :

Diagramme composant-connecteur nous permet d’avoir un aperçu des composant et des liens qui vont avoir à la fin du projet et nous permettra de guider avec l’interaction des composants.

Diagramme de déploiement permet de répartir les composant de la vue où l’utilisateur va observer devant son écran et de séparer au contrôle avec tous les composants et code qui permettra de rendre une page web dynamique.

Implémentation monolithique technologique

Est une représentation des interactions entre les différentes technologies que nous devions utiliser.

**Programmation**

Requête sql pour PHP :

Nous proposons des requêtes sql pour être inséré dans le code en php

SIMFAST :

Nous proposons un code pour la création des tables sur le serveur.

**Test**

Test Unitaire sql pour php:

C’est un type de test qui permet de vérifier si une partie de donnée, indépendante du reste du programme correspondent au résultat attendu.

Ici les tests sont réalisés le code SQL qui va être insérer dans le php donc nous nous occupons juste s’il y a un retour du code que nous proposons pour les pages en php.