

# Rapport du Deuxième livrable

## Table des matières

Arborescence du deuxième livrable :.....	1
Choix du livrable.....	3
Deuxième livrable.....	4

## Arborescence du deuxième livrable :

### 1. Analyse\_des\_besoins :

#### ○ Cahier\_des\_charges

- Un (deuxième) cahier des charges minimal
- Un cahier des charges cumulé

#### ○ Organisation :

- 1.WBS
- 2.Les matrices d'antériorité
- 3.GANTT

#### ○ Recueil\_des\_besoins

- Recueil\_des\_besoins

### 2. Spécification :

#### ○ Cas d'utilisation

- Cas\_d\_utilisation
- Scénarii
- Etude\_technologie\_web
- **Maquette\_logo**
  - Maquette\_logo (nous avons besoin du dossier pour pouvoir faire les tests sur ce livrable donc nous mettons le dossier dans ce livrable)

### 3. Conception :

- **Diagramme**
  - Diagramme\_composant-connecteur
  - Diagramme\_deploiement
- Implementation\_monolithique\_technologies
- **Conception\_architecturale\_Pagehtml**
  - Conception\_architecturale\_pageHTML (nous avons besoin de la conception pour pouvoir faire les tests sur ce livrable donc nous mettons le dossier dans ce livrable)

### 4. Programmation :

- **Serveur**
  - Rapport\_installation

### 5. Test :

- **Dossier\_de\_test**
  - Document\_test\_validation

## Choix du livrable

### Choix du deuxième livrable

Nous avons choisi dans ce deuxième livrable de faire un processus de développement qui porte sur le serveur à installer. Nous nous concentrons seulement sur ce point donc nous ne retrouvons pas tous les dossiers du premier livrable.

### Choix de l'arborescence du deuxième livrable

Nous avons choisi cette méthode de livrable pour la raison suivante, afin de mener à bien un projet informatique de cette envergure, nous devons rassembler les activités du génie logiciel pour respecter les critères de qualités. Nous avons donc proposé de faire plusieurs livrables avec une caractéristique : les dossiers seront divisés en selon les activités 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 ce premier livrable représente un cycle de vie du produit logiciel. Le produit logiciel est décomposé en plusieurs cycles de vie et sera complété par de nouveaux livrables au fil du déroulement du projet.

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 définiront 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'enchaînement des tâches effectuées par notre équipe. Chaque livrable se basera sur cette méthode. Lorsque ces activités seront 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/problèmes que l'on rencontre. Une fois la programmation terminée nous exécutons les tests de validation qui vont venir valider les besoins fonctionnelles définis dans le cahier des charges puis construire à partir des maquettes un plan répondant aux exigences du client. Les tests d'intégrations vont venir valider la relation des composants (pages html). Nous sommes donc dans un cycle linéaire.

---

## Deuxième livrable

### Analyse des besoins

Le (deuxième) cahier des charges minimal :

Nous proposons un deuxième 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 deuxième 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 définies 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

Études des technologies utilisées :

On définit toutes les technologies utilisées lors de ce projet.

## **Conception**

Conception\_architecturale\_Pagehtml :

La conception architecturale permet de rendre compréhensible le modèle avec les interactions (relations) entre les pages html. Cela permet de faciliter le passage à la programmation par la représentation de ce modèle.

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

Diagramme :

Le diagramme composant-connecteur nous permet d'avoir un aperçu des composants et des liens qu'ils vont avoir à la fin du projet, cela nous guidera pour l'interaction des composants.

Le diagramme de déploiement permet de répartir les composants 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 :

C'est une représentation des interactions entre les différentes technologies que nous devons utiliser.

## **Programmation**

Rapport de l'installation du serveur :

C'est un rapport qui contient du code concernant l'installation du serveur pour pouvoir accueillir la base de données ainsi que les pages de l'application web. Il contient aussi des images de tests sur la connexion sur navigateur et sur la base de données

## **Test**

**Test d'acceptation :**

C'est un type de test qui permet de vérifier si toutes les exigences client, décrites dans le cahier des charges correspondent aux attentes du produit définies dans chaque livrable. Elle est aussi décrite à partir de l'activité de spécification grâce à la réalisation des maquettes.

Nous retrouvons les tests d'acceptation réalisés à partir des attentes des clients et des maquettes, et validés après la programmation si cette dernière correspond bien.

Test d'intégration :

C'est un type de test qui permet de vérifier si les exigences du client sur la navigabilité entre les composants correspondent bien. Les tests sont réalisés à partir du cahier des charges et approfondis dans la conception architecturale dans la partie conception.

Nous retrouvons les tests d'intégration réalisés à partir des attentes des clients et de la conception architecturale. La conception architecturale permet de voir si

une page appelle l'autre. Ces tests sont validés après la programmation de toutes les pages html et si toutes les interactions entre les pages correspondent bien.

Test des technologies :

C'est un type de test qui permet de vérifier le fonctionnement des technologies que nous avons proposés qui interagissent avec le serveur.