Projet d’Approfondissement



Figure : Logo de Laravel[[1]](#footnote-1)

Dorian Capelli – CIN4B

Lausanne

8 semaines

Grégory Charmier

Table des matières

[1 Spécifications 3](#_Toc125640824)

[1.1 Titre 3](#_Toc125640825)

[1.2 Description 3](#_Toc125640826)

[1.3 Matériel et logiciels à disposition 3](#_Toc125640827)

[1.4 Prérequis 3](#_Toc125640828)

[1.5 Cahier des charges 3](#_Toc125640829)

[1.5.1 Objectifs et portée du projet (objectifs SMART) 3](#_Toc125640830)

[1.5.2 Caractéristiques des utilisateurs et impacts 3](#_Toc125640831)

[1.5.3 Fonctionnalités requises (du point de vue de l’utilisateur) 3](#_Toc125640832)

[1.5.4 Contraintes 3](#_Toc125640833)

[1.5.5 Travail à réaliser par l'apprenti 4](#_Toc125640834)

[1.5.6 Si le temps le permet … 4](#_Toc125640835)

[1.5.7 Méthodes de validation des solutions 4](#_Toc125640836)

[1.6 Les points suivants seront évalués 4](#_Toc125640837)

[1.7 Validation et conditions de réussite 4](#_Toc125640838)

[2 Planification Initiale 4](#_Toc125640839)

[3 Analyse 5](#_Toc125640840)

[3.1 Opportunités 5](#_Toc125640841)

[3.2 Document d’analyse et conception 5](#_Toc125640842)

[3.3 Conception des tests 5](#_Toc125640843)

[3.4 Planification détaillée 5](#_Toc125640844)

[4 Réalisation 6](#_Toc125640845)

[4.1 Dossier de Réalisation 6](#_Toc125640846)

[4.2 Modifications 6](#_Toc125640847)

[5 Tests 6](#_Toc125640848)

[5.1 Dossier des tests 6](#_Toc125640849)

[6 Conclusion 6](#_Toc125640850)

[6.1 Bilan des fonctionnalités demandées 6](#_Toc125640851)

[6.2 Bilan de la planification 6](#_Toc125640852)

[6.3 Bilan personnel 6](#_Toc125640853)

[7 Divers 7](#_Toc125640854)

[7.1 Journal de travail 7](#_Toc125640855)

[7.2 Webographie 7](#_Toc125640856)

[7.2.1 Source : 7](#_Toc125640857)

[8 Annexes 8](#_Toc125640858)

# Spécifications

## Titre

Projet D’Approfondissement

## Description

Projet durant 8 semaines encadrer comme le TPI afin de nous habituer à la préparation spécifique que demande le TPI.

## Matériel et logiciels à disposition

* Un Ordinateur standard de l’ETML
* DB main
* Laravel
* La suite office 365 (Word, Excel, PowerPoint, Teams)
* Sublime Texte

## Prérequis

Module104 BD Implémenter Un modelé

Module 105 BD SQL

Module 302 Bureautique Avancée

Module 306 Projet Réaliser Petit Projet

Module 403 Programmation Structurée

Module 404 Programmation Bassée Objets

Module 426 Développer AGILE

## Cahier des charges

### Objectifs et portée du projet (objectifs SMART)

A compléter. Il s’agit d’ébaucher des réponses aux questions de l’acronyme CQQCOQP (Combien, Quoi, Qui, Comment, Où, Quand, Pourquoi)

### Caractéristiques des utilisateurs et impacts

A compléter… Il s’agit d’identifier le(s) profil(s) de(s) utilisateur-trice(s) type, et les conséquences que cela va avoir sur la conception (couleurs, ergonomie, utilisation, etc.)

### Fonctionnalités requises (du point de vue de l’utilisateur)

A compléter par une espèce de mode d’emploi du produit. S’il s’agissait d’une montre, décrire qu’à part l’heure, il y aura la possibilité d’utiliser un chronomètre, un réveil, …

### Contraintes

Sécurité, backups, disponibilité, système utilisé, interfaces avec autres logiciels, etc.

### Travail à réaliser par l'apprenti

Décrire à quoi doit ressembler le travail produit, ce qu’il faudra rendre …

### Si le temps le permet …

Objectifs complémentairesau cas où le projet n’est pas assez ambitieux dans le temps imparti…

### Méthodes de validation des solutions

Comment les tests vont être entrepris, quels tests doivent être entrepris, etc.…

## Les points suivants seront évalués

* Le rapport
* Les planifications (initiale et détaillée)
* Le journal de travail
* Le code et les commentaires
* Les documentations de mise en œuvre et d’utilisation

## Validation et conditions de réussite

* Compréhension du travail
* Possibilité de transmettre le travail à une personne extérieure pour le terminer, le corriger ou le compléter
* Etat de fonctionnement du produit livré

# Planification Initiale

Figure :Capture d'écran Plannification Initial

# Analyse

## Opportunités

Ce projet me permet de m’améliorer mes connaissances et mes compétences dans la création de site web en particulier découvrir et utilisé Laravel et mettre en place une structure MVC. En ce qui concerne Laravel, je dois entièrement le découvrir et apprendre à l’utilise complétement et en MVC, je dois revoir comment mettre en place la structure, correctement, n’ai pratiquement jamais réaliser de structure MVC.

## Document d’analyse et conception

Ce paragraphe décrit le fonctionnement de manière détaillée.

Autant que possible de manière graphique, imagée, tableaux, etc.

Tous les cas particuliers devraient y être spécifiés…

Il s’agit d’y présenter les fonctionnalités à développer :

Découpage en étapes, en modules, en fonctionnalités, etc.

Formulaires, interfaces graphiques, pages web, etc.

Schémas de navigation, schémas événementiels, structogramme, pseudocode, etc.

Si le projet inclut une base de données :

Dictionnaire des données

Modèle conceptuel des données, modèles logique des données.

## Conception des tests

Ce paragraphe permet de spécifier la stratégie de test qui sera menée au point 5.1

Qui, quand, avec quelles données, dans quel ordre, etc.

## Planification détaillée

Je vais utiliser une méthode AGILE vu les particularités de mon projet. La plus grande particularité qui me fait choisir une méthodologie de travail AGILE est que je vais travailler sur plusieurs projets simultanément. Nous aurons un projet principal qui sera en coalition avec d’autres enseignants et apprentis d’autre établissement dans toute la Suisse romande qui pratique différent métier qu’informaticien.

Notre planification détaillée consiste en sept tâches principales dont deux sortes du lot. Nos tâches sont Informer, Planifier, Décider, Réaliser, Test, Évaluer et Rédaction du journal de travail, Rapport et Présentation. Informer consiste à analyser le projet afin de prévoir sur quel point nous aurons de la difficulté. Planifier consiste à découper le projet en tâches et concevoir la planification initiale et détaillée. Décider consiste à choisir de quelle manière de quel langage de programmation, quelle donnée, quel matériel, quelle interface utilisée. Réaliser consiste à modéliser, programmer, en résumer construire le projet. Tester consiste à effectuer des tests de fonctionnement afin de vérifier si les objectifs sont remplis. Évaluer consiste comme confirmation que tout correspond au cahier des charges. Rédaction du journal de travail, Rapport et Présentation consiste à remplir le journal de travail, construire le rapport et créer la présentation entière qui va être devant la classe. Les deux tâches qui sortent du lot sont Réaliser et Rédaction du journal de travail, Rapport et Présentation parce que la première sera la plus longue et la deuxième sera la plus récurrente.

Le document Excel montrant la planification détaillée est disponible en [Annexes](#_Annexes).

# Réalisation

## Dossier de Réalisation

Cette partie permet de reproduire ou reprendre le projet par un tiers.

Pour chaque étape, il faut décrire sa mise en œuvre. Typiquement :

Versions des outils logiciels utilisés (OS, applications, pilotes, librairies, etc.)

Configurations spéciales des outils (Equipements, PC, machines, outillage, etc.)

Code source commenté des éléments logiciels développés.

Modèle physique d’une base de données.

Arborescences des documents produits.

Il faut décrire le parcours de réalisation et justifier les choix.

## Modifications

Historique des modifications demandées (ou nécessaires) aux spécifications détaillées.

Date, raison, description, etc.

# Tests

## Dossier des tests

On dresse le bilan des tests effectués (qui, quand, avec quelles données…) sous forme de procédure. Lorsque cela est possible, fournir un tableau des tests effectués avec les résultats obtenus et les actions à entreprendre en conséquence (et une estimation de leur durée).

Si des tests prévus dans la stratégie n'ont pas pu être effectués :

raison, décisions, etc.

Liste des bugs répertoriés avec la date de découverte et leur état:

Corrigé, date de correction, corrigé par, etc.

# Conclusion

## Bilan des fonctionnalités demandées

Il s’agit de reprendre point par point les fonctionnalités décrites dans les spécifications de départ et de définir si elles sont atteintes ou pas, et pourquoi.

Si ce n’est pas le cas, estimer en « % » ou en « temps supplémentaire » le travail qu’il reste à accomplir pour terminer le tout.

## Bilan de la planification

Distinguer et expliquer les tâches qui ont généré des retards ou de l'avance dans la gestion du projet. Indiquer les différence entre les planifications initiales et détaillées avec le journal de travail.

## Bilan personnel

Si c’était à refaire:

Qu’est-ce qu’il faudrait garder ? Les plus et les moins ?

Qu’est-ce qu’il faudrait gérer, réaliser ou traiter différemment ?

Qu’est que ce projet m’a appris ?

Suite à donner, améliorations souhaitables, …

Remerciements, signature, etc.

# Divers

## Journal de travail

## Webographie

### Source :

1. Image page de Titre : nom du site : Wikipédia, titre de la page : Laravel, Modifié le 6 mars 2022 à 15 :28, <https://fr.wikipedia.org/wiki/Laravel>
2. Guide Vidéo sur Laravel : nom du site : Youtube, titre de la page : LaraGuide ¦ Découverte, Modifié le 29 juillet 2020, <https://www.youtube.com/playlist?list=PLMWEEzYqZ0em1vnBx8F5GZd94Ejph4TjO>
3. Documentation officiel laravel : nom du site : Laravel, titre de la page : Getting Started, Modifié le 2011-2023, <https://laravel.com/docs/9.x>

Références des sites Internet consultés durant le projet.

# Annexes

Listing du code source (partiel ou, plus rarement complet)

Guide(s) d’utilisation et/ou guide de l’administrateur

Etat ou « dump » de la configuration des équipements (routeur, switch, robot, etc.).

Extraits de catalogue, documentation de fabricant, etc.

1. Référence sous-chapitre Source élément 1 [↑](#footnote-ref-1)