|  |
| --- |
| Tpi  Site de génération de labyrinthe réalisables  Get Me Out |

*Zubieta Pablo*

****

*SI-CA2a*

*04.05.23*

Table des matières

[1 Analyse préliminaire 3](#_Toc131154975)

[1.1 Introduction 3](#_Toc131154976)

[1.2 Organisation 3](#_Toc131154977)

[1.3 Objectifs 3](#_Toc131154978)

[1.4 Planification initiale 3](#_Toc131154979)

[2 Analyse 4](#_Toc131154980)

[2.1 Cahier des charges détaillé 4](#_Toc131154981)

[2.2 Définition de l’audience 7](#_Toc131154982)

[2.3 Analyse concurrentielle 8](#_Toc131154983)

[2.4 Cas d’utilisation 8](#_Toc131154984)

[2.5 Etude de faisabilité 11](#_Toc131154985)

[2.6 MCD 12](#_Toc131154986)

[2.7 Nom du site et du domaine 12](#_Toc131154987)

[2.8 Stratégie de test 12](#_Toc131154988)

[2.9 Budget initial 15](#_Toc131154989)

[2.10 Planification détaillée 15](#_Toc131154990)

[2.11 Historique 16](#_Toc131154991)

[3 Conception 19](#_Toc131154992)

[3.1 Analyse de l’environnement 19](#_Toc131154993)

[3.2 Détermination de l’arborescence du site et des rubriques 19](#_Toc131154994)

[3.3 Définition de la charte graphique 19](#_Toc131154995)

[3.4 Maquette graphique 20](#_Toc131154996)

[3.5 Conception de la Base de données 21](#_Toc131154997)

[3.6 Conception du Code 22](#_Toc131154998)

[3.7 Plugins et librairies 23](#_Toc131154999)

[3.8 Choix de la formule d’hébergement 23](#_Toc131155000)

[4 Réalisation 24](#_Toc131155001)

[4.1 Dossier de réalisation 24](#_Toc131155002)

[4.2 Description des tests effectués 25](#_Toc131155003)

[4.3 Erreurs restantes 26](#_Toc131155004)

[4.4 Dossier d'archivage 26](#_Toc131155005)

[5 Mise en service 27](#_Toc131155006)

[5.1 Rapport de mise en service 27](#_Toc131155007)

[5.2 Liste des documents fournis 30](#_Toc131155008)

[6 Conclusions 31](#_Toc131155009)

[7 Annexes 32](#_Toc131155010)

[7.1 Sources – Bibliographie 32](#_Toc131155011)

[7.2 Manuel d'Installation 32](#_Toc131155012)

[7.3 Manuel d'Utilisation 32](#_Toc131155013)

[7.4 Wireframe 33](#_Toc131155014)

[7.5 Diagramme de flux 38](#_Toc131155015)

[7.6 Archives du projet 42](#_Toc131155016)

# Analyse préliminaire

## Introduction

Le projet Get Me Out est un site web permettant aux enseignants du cpnv de trouver un covoiturage près de chez eux en fonction de leur horaire. Ce site crée avec un Framework PHP : Laravelel.

## Organisation

Elève : Zubieta Pablo [zubieta-rodriguez.pablo-fernando@cpnv.ch](mailto:zubieta-rodriguez.pablo-fernando@cpnv.ch) 0764788611

Responsable de projet : Benzonana Pascal [benzonana.pascal@cpnv.ch](mailto:benzonana.pascal@cpnv.ch)

## Objectifs

* Être capable de mener un projet Web à son terme.
* Comprendre le fonctionnement d’un MVC.
* Faire et savoir gérer une planification.

## Planification initiale

La planification de ce projet sera faite sur Icesrum

Elle se divise en 3 sprints principaux.

Le premier pour la prise en main de Laravel et le login

Le second l’implémentation de l’algorithme de planification des covoiturages

Le dernier pour mise en place du mailing et des API

# Analyse

## Cahier des charges détaillé

### INFORMATIONS GENERALES

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Candidat** | Nom : | **ZUBIETA** | **Prénom :** | Pablo |
|  |  | Pablo-Fernando.ZUBIETA- [RODRIGUEZ@cpnv.ch](mailto:RODRIGUEZ@cpnv.ch) |  | |
| **Lieu de travail :** |  | | | |
| **Orientation :** | * 88601 Développement d’applications   ☒ 88602 Informatique d’entreprise   * 88603 Technique des systèmes | | | |
| **Chef de projet** | **BENZONANA** | | Pascal | |
| : [pascal.benzonan](mailto:pascal.benzonana@cpnv.ch)[a@cpnv.ch](mailto:a@cpnv.ch) | | : 076 230 23 13 | |
| **Expert 1** | Nom: **OBERSON** | | Prénom: Bernard | |
| : [oberson.bernard@gmail.com](mailto:oberson.bernard@gmail.com) | |  | |
| **Expert 2** | Nom: **BERTINO** | | Prénom: Yves | |
| : [yves@bertino.ch](mailto:yves@bertino.ch) | |  | |
| **Période de réalisation :** | Du lundi 2 mai 2023 à 8h00 au mercredi 31 mai 2023 à 8h45 | | | |
| **Horaire de travail :** | Lundi 09h50-12h15 13h30-16h55 *Pentecôte 29 mai*  Mardi 08h00-12h15 13h30-16h55  Mercredi 08h00-12h15 -  Jeudi 08h00-12h15 13h30-16h55 *Examen Ascension 18 mai*  *ECG 1er juin matin*  Vendredi - - *Pont de l'Ascension 19 mai Toutes les demi-journées ont une pause obligatoire de 15 minutes, sauf si elles commencent à 9h50 ou si elles se terminent à 15h05.*  Voir en annexe en fin de document | | | |
| **Nombre d'heures :** | 90 heures | | | |
| **Planning** | Analyse : 20% | | | |
| Implémentation : 45% | | | |
| Tests : 10% | | | |
| Documentations : 25% | | | |
| **Présentation** | Dates retenues : 12 ou 13 juin 2023 | | | |

1. PROCÉDURE
   * Le candidat réalise un travail personnel sur la base d'un cahier des charges reçu le 1er jour.
   * Le cahier des charges est approuvé par la i-CQ VD. Il est en outre présenté, commenté et discuté avec le candidat. Par sa signature, le candidat accepte le travail proposé.
   * Le candidat a connaissance de la feuille d’évaluation avant de débuter le travail.
   * Le candidat est entièrement responsable de la sécurité de ses données.
   * En cas de problèmes graves, le candidat avertit au plus vite les deux experts et son CdP.
   * Le candidat a la possibilité d’obtenir de l’aide, mais doit le mentionner dans son dossier.
   * A la fin du délai imparti pour la réalisation du TPI, le candidat doit transmettre par courrier électronique le dossier de projet aux deux experts et au chef de projet. En parallèle, une copie papier du rapport doit être fournie sans délai en trois exemplaires (L’un des deux experts peut demander à ne recevoir que la version électronique du dossier). Cette dernière doit être en tout point identique à la version électronique.

### TITRE

Site de génération de labyrinthes réalisables

#### **SUJET**

Le projet consiste à créer un site Web permettant de générer des labyrinthes solubles.

Les labyrinthes seront générés à partir d’un point central défini soit par l’utilisateur soit par l’application.

### MATÉRIEL ET LOGICIEL À DISPOSITION

1 ordinateur standard du CPNV avec Windows 10 professionnel et la suite Office et des crédits Azure

Les autres logiciels disponibles sont :

* Windows 10
* Suite Office
* Balsamiq
* MySQL Workbench
* PHPStorm
* Outils de gestion de projet : GitHub Project / Notion

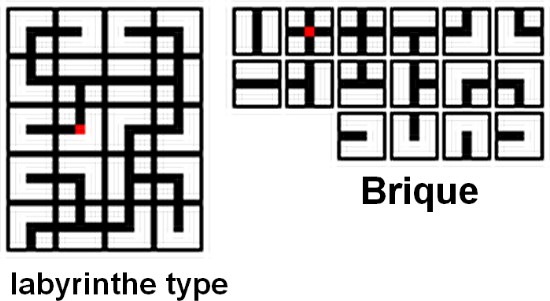
### PRÉREQUIS

Le candidat possède de bonnes connaissances programmation Web et modélisation (I-CT 101, 104, 105, 120, 133, 151) et sait mettre en œuvre les différents concepts nécessaires au développement d’application. La candidate a acquis déjà une expérience en programmation Web et a déjà travaillé avec une base de données MySQL.

### DESCRIPTIF DU PROJET

Le candidat devra identifier les besoins de création de différents modules pour le projet final, il devra pour s’organiser et visualiser la finalité du projet, créer une maquette du site et modéliser sa base de données. Une fois que cela sera fait, il pourra ensuite passer à la programmation et réalisation du projet.

Le site devra générer des labyrinthes solubles à partir d’un point central (point rouge) comme par exemple sur l’image ci-dessous. La sortie se trouve sur la brique qui touche le bord extérieur du carré/rectangle (cercle vert). La représentation des briques utilisables n’est pas exhaustive.



Dans un premier temps, il créera sa base de briques pour pouvoir dans un 2ème temps générer les labyrinthes.

#### **Génération manuelle :**

L’utilisateur après avoir choisi la taille du labyrinthe pourra déposer ses briques en indiquant les points d’entrée et de sortie du labyrinthe. L’application se chargera de vérifier que le labyrinthe proposé soit réalisable. Les tailles de labyrinthe exprimées en nombre de briques sont par exemple :

- 4 x 5

- 6 x 5

- 10x10

#### **Génération automatique :**

Après que l’utilisateur aura choisi sa taille de labyrinthe, l’application se chargera de le créer et de vérifier s’il est soluble. Tant qu’il ne l’est pas, l’application procède à des tentatives

### 7 LIVRABLES

Le candidat est responsable de livrer à son chef de projet et aux deux experts :

* + 02.05.2023 : une planification initiale contenant la liste des sprints ainsi que les dates/heures des sprint reviews (confirmées avec le PO).
  + A chaque fin de sprint :
    - Un commit signifiant clairement le livrable pour le sprint
    - Une documentation mise à jour
    - Un déploiement du site quand la réalisation aura débuté
  + A la fin du TPI, le candidat livrera :
    - Les sources et les données sur le dépôt,
    - La documentation sur le dépôt,
    - Un CD ou clé USB avec les sources, la documentation et le journal à jour
  + Le journal de travail et le rapport devront être transmis par courriel deux fois par semaine.

### 8 POINTS TECHNIQUES ÉVALUÉS SPÉCIFIQUES AU PROJET

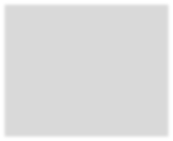
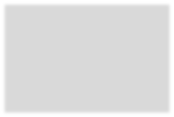
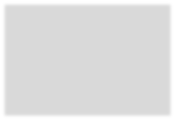
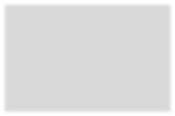
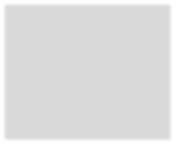
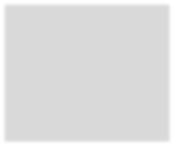
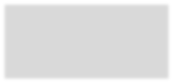
La grille d’évaluation définit les critères généraux selon lesquels le travail du candidat sera évalué (documentation, journal de travail, respect des normes, qualité, …).

En plus de cela, le travail sera évalué sur les sept points spécifiques suivants :

1. *Le site affiche les différentes briques possibles.*
2. *L’utilisateur peut choisir ses tailles de labyrinthes (min 4x4)*
3. *Ergonomie et facilité d'utilisation du produit (Bastien et Scapin)*
4. *L’application teste si le labyrinthe est réalisable.*
5. *L’utilisateur peut retrouver les labyrinthes qu’il a créés et ceux qu’il a résolus*
6. *L’utilisateur peut générer des labyrinthes manuellement*
7. *L’utilisateur peut générer des labyrinthes automatiquement*
8. VALIDATION

|  |  |  |
| --- | --- | --- |
|  | Lu et approuvé le : | Signature : |
| ZUBIETA-RODRIGUEZ  Pablo-Fernando |  |  |
| OBERSON Bernard |  |  |
| BERTINO Yves |  |  |
| Pascal BENZONANA |  |  |

1. HORAIRE DE TRAVAIL



## Définition de l’audience

Toutes les personnes faire ou résoudre des labyrinthes.

## Analyse concurrentielle

## Cas d’utilisation

## Etude de faisabilité

Il existe une infinité de manière de faire une application génératrice de labyrinthe. Utiliser Unity, une diversité de langage allant du JS au Python en passant par le Ruby, Avec des Framework ou en natif, Avec du sql ou du nosql. Ayant beaucoup apprécier le PHP c’est vers ce langage que je me suis orienté.

Il existe j’avais 3 principale piste pour faire le Backend de cette application, qui en est l’aspect principale.

* Le PHP natif : étant la matière enseignée au cpnv c’est de loin l’approche que je maitrise le mieux.
* Laravel : est donc un Framework PHP très commun et populaire pour le bakcend, il est d’ailleurs enseigné au CPNV.
* Symphony : est également un Framework PHP.

Mon choix c’est porté sur Laravel. Pour trois raisons. La première est que Laravel est une technologie employée dans l’industrie. La Deuxième est qu’étant enseigné ici je pense que se sera plus simple de me faire aider en cas de difficulté. Et la dernière que c’est le Framework que j’ai employé pour faire mon Pré-Tpi donc une technologie, je commence à bien comprendre.

## MCD

## Nom du site et du domaine

## Stratégie de test

L’utilisation de la méthode agile rend la stratégie de test très évidente.

Les tests sont créés à l’écriture des stories Et sont effectuer lors de la sprint review.

## Budget initial

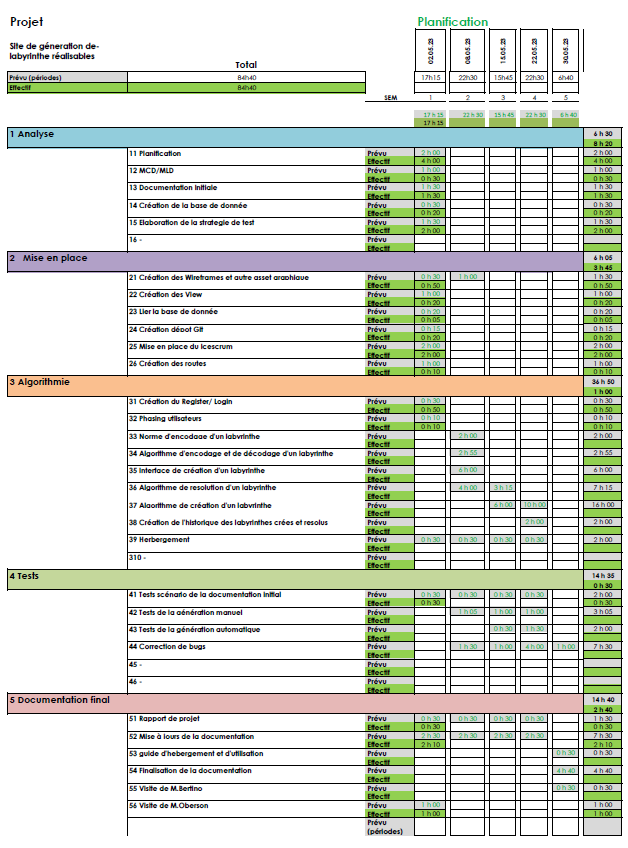
C’est 0.00 CHF

Pour un petit site comme ça l’hébergement est négligeable selon mes calcule il est des 1,35 CHF/par année

Sauf si on considère le prix des licences dans ce cas

* C’est PHPstorm = 247 CHF
* Balsamiq = 89 CHF
* Total = 336 CHF

## Planification détaillée



## Historique

# Conception

## Analyse de l’environnement

Environement de développement

Windows 10 family

Mariabd 10.10

Phpstorm 2022.2.1

Laravel9.0 ->composer 2.5.2

Php8.1

Hébergement

Php8.1

Apache/2.4.37

Mysql 8.0.32-cll-lve

## Détermination de l’arborescence du site et des rubriques

## Définition de la charte graphique

## Maquette graphique

## Conception de la Base de données

## Conception du Code

## Plugins et librairies

Etant donné que ce projet c’est fait sous Laravel il y a une tonne de librairie associer tel que Eloquent (la librairie qui gérer l’accès à la base de données).

On peut citer également la Bootstrap qui mon Framework css.

## Choix de la formule d’hébergement

Donc le CdP a choisi l’hébergement sur swisscenter. Je n’ai pas beaucoup plus de précision là-dessus

# Réalisation

## Dossier de réalisation

### Répertoire

1. Répertoire "physique" du projet :
   * Sur la machine SC-C236-PC04
   * C:\Tpi
2. Repository distant :
   * https://github.com/PabloZubieta/Tpi

### Dossier bases de données

|  |  |
| --- | --- |
| **Nom** | **Description** |
| MCD\_V1 | Fichier contenant le MCD de la DB |
| MLD\_V1 | Fichier contenant le MLD de la DB |

### Dossier documentation

|  |  |
| --- | --- |
| **Nom** | **Description** |
| CDC-PBA-Pablo-Zubieta\_Labyrinthe\_v1\_2 | Cahier des charges |
| Timesheet.html | Journal de travail |
| "documentation.pdf" | La documentation que vous êtes en train de lire |
| "Planification.pdf" | La planification |

### Dossier code

## Description des tests effectués

Listes des tests

## Erreurs restantes

|  |  |  |  |
| --- | --- | --- | --- |
| **Erreur** | **Description** | **Conséquences** | **Action** |

## Dossier d'archivage

# Mise en service

## Rapport de mise en service

### Publication chez l’hébergeur

### Guide d’installation

* Pour l’installation il vous fait évidement php8.1 et mariadb
* Il faudra également composer que vous trouverez à cette adresse : <https://getcomposer.org/>
* Une fois tous ses programmes installation réside dans l’obtention de la librairie Vendor
* Si vous récupérer mon projet dans le git vous aurez tout le projet sauf la librairie Vendor qui n’es pas synchroniser
* Vous devrez donc crée un projet Laravel et copier le dossier vendor que composer va vous générer
* Exécuter les scripts sql contenu dans le dossier database de manière à crée la base de données
* Vous localiser dans le dossier Code et lancer la commande : php artisan serve
* Et votre projet devrait tourner

### Erreurs restantes

|  |  |  |  |
| --- | --- | --- | --- |
| ***Erreur*** | ***Description*** | ***Conséquences*** | ***Action*** |

*.*

## Liste des documents fournis

* Documentation

# Conclusions

# Annexes

## Résumé du rapport

## Sources – Bibliographie

<https://www.youtube.com/watch?v=FC7JQItVqzM>

<https://www.youtube.com/watch?v=MYyJ4PuL4pY>

<https://laravel.com/docs/9.x>

<https://www.w3schools.com>

<https://roadmap.sh/backend>

## Manuel d'Installation

Voir guide d’installation

## Manuel d'Utilisation

Voir publication chez l’hébergeur

## Wireframe

## Diagramme de flux

## Archives du projet

[*https://github.com/PabloZubieta/Tpi*](https://github.com/PabloZubieta/Tpi)