|  |
| --- |
| Pre-Tpi  Ecolopnv |

*Zubieta Pablo*

****

*SI-CA2a*

*06.03.23*

Table des matières

[1 Analyse préliminaire 3](#_Toc129966169)

[1.1 Introduction 3](#_Toc129966170)

[1.2 Organisation 3](#_Toc129966171)

[1.3 Objectifs 3](#_Toc129966172)

[1.4 Planification initiale 3](#_Toc129966173)

[2 Analyse 4](#_Toc129966174)

[2.1 Cahier des charges détaillé 4](#_Toc129966175)

[2.2 Définition de l’audience 4](#_Toc129966176)

[2.3 Analyse concurrentielle 4](#_Toc129966177)

[2.4 Cas d’utilisation 5](#_Toc129966178)

[2.5 Etude de faisabilité 8](#_Toc129966179)

[2.6 MCD 9](#_Toc129966180)

[2.7 Nom du site et du domaine 9](#_Toc129966181)

[2.8 Stratégie de test 9](#_Toc129966182)

[2.9 Budget initial 9](#_Toc129966183)

[2.10 Planification détaillée 9](#_Toc129966184)

[2.11 Historique 10](#_Toc129966185)

[3 Conception 11](#_Toc129966186)

[3.1 Analyse de l’environnement 11](#_Toc129966187)

[3.2 Détermination de l’arborescence du site et des rubriques 11](#_Toc129966188)

[3.3 Définition de la charte graphique 11](#_Toc129966189)

[3.4 Maquette graphique 11](#_Toc129966190)

[3.5 Conception de la Base de données 11](#_Toc129966191)

[3.6 Conception du Code 11](#_Toc129966192)

[3.7 Plugins et librairies 12](#_Toc129966193)

[3.8 Choix de la formule d’hébergement 12](#_Toc129966194)

[4 Réalisation 13](#_Toc129966195)

[4.1 Dossier de réalisation 13](#_Toc129966196)

[4.2 Description des tests effectués 13](#_Toc129966197)

[4.3 Erreurs restantes 13](#_Toc129966198)

[4.4 Dossier d'archivage 13](#_Toc129966199)

[5 Mise en service 14](#_Toc129966200)

[5.1 Rapport de mise en service 14](#_Toc129966201)

[5.2 Liste des documents fournis 14](#_Toc129966202)

[6 Conclusions 14](#_Toc129966203)

[7 Annexes 15](#_Toc129966204)

[7.1 Sources – Bibliographie 15](#_Toc129966205)

[7.2 Manuel d'Installation 15](#_Toc129966206)

[7.3 Manuel d'Utilisation 15](#_Toc129966207)

[7.4 Archives du projet 15](#_Toc129966208)

# Analyse préliminaire

## Introduction

Le projet ecolopnv 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.

*Plus de detail*

## 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

Voir en annexe

## Définition de l’audience

*Ce projet est conçu pour les enseignant du cpnv disposant d’une voiture*

## Analyse concurrentielle

Il y a, à ma connaissance, aucune application de covoiturage spécifiquement pour les enseignant du cpnv.

Pour ce qui est de blablacar ou E-voiturage.ch, elles ne disposent pas des horaires des enseignants et ne peuvent donc pas auto suggérer des trajets.

Donc ce site n’a pas vraiment de concurrence.

## Cas d’utilisation

Use case : l’administrateur ajoute L’edt

Scénario

|  |  |  |
| --- | --- | --- |
| Action | Conditions particulières | Réaction |
| L’administrateur  Ajoute le nouvel edt dans la base de données |  | Active un trigger qui lance la drop la table des horaires et lance une procédure stockée |
| La procédure stockée traite d’edt |  | Elle remplit la table horaire avec nouveau les horaire |
| Elle sélectionne tous les utilisateurs uniques de la table horaire |  | Et les insère dans la table utilisateur |

Use case : l’utilisateur s’enregistre sur le site pour participer au covoiturage

Scénario

|  |  |  |
| --- | --- | --- |
| Action | Conditions particulières | Réaction |
| L’utilisateur clique sur le sign in de la navbar |  | L’utilisateur est redirigé sur la page sign in |
| L’utilisateur remplit le champ acronyme | L’utilisateur possédant cet acronyme doit exister dans la base de données |  |
| L’utilisateur remplit le champ email | L’entrée doit être un email |  |
| L’utilisateur sélectionne une des entrées dans le champ lieux d'habitation |  |  |
| L’utilisateur remplit le champ place de voiture | La valeur doit être nombre entier naturel |  |
| L’utilisateur remplit le champ mot de passe et confirmation | Les deux champs doivent être identique |  |
| L’utilisateur clique sur le bouton valider | Tous les champs doivent être remplit | L’utilisateur en question est activé, et il est authentifié et redirigé sur la page d’accueil |

Extension : Les champ sont mal remplis

|  |  |  |
| --- | --- | --- |
| Action | Conditions particulières | Réaction |
| L’utilisateur clique sur le bouton valider |  | Une erreur décrivant le problème apparait sur le champ en question |

Use case : l’utilisateur consulte son horaire

Scénario

|  |  |  |
| --- | --- | --- |
| Action | Conditions particulières | Réaction |
| L’utilisateur clique sur le login de la navbar |  | L’utilisateur est redirigé sur la page login |
| L’utilisateur remplit le champ acronyme du login | L’utilisateur possédant cet acronyme doit être activé |  |
| L’utilisateur remplit le champ mot de passe | Le mot de passe doit être correct |  |
| L’utilisateur clique sur le bouton valider |  | L’utilisateur est authentifié et redirigé sur la page d’accueil |
| L’utilisateur clique sur my schedule |  | L’utilisateur est redirigé sur la page schedule ou son horaire est affiché |

Extension : Le champ sont mal remplis

|  |  |  |
| --- | --- | --- |
| Action | Conditions particulières | Réaction |
| L’utilisateur clique sur le bouton valider |  | Une erreur décrivant le problème apparait sur le champ en question |

Extension : l’utilisateur accède sans se connecter

|  |  |  |
| --- | --- | --- |
| Action | Conditions particulières | Réaction |
| L’utilisateur rentre l’url de la page sans être connecter |  | Il est redirigé sur la page login |

Use case : l’utilisateur modifie ses informations

Scénario

|  |  |  |
| --- | --- | --- |
| Action | Conditions particulières | Réaction |
| L’utilisateur clique sur le login de la navbar |  | L’utilisateur est redirigé sur la page login |
| L’utilisateur remplit le champ acronyme du login | L’utilisateur possédant cet acronyme doit être activé |  |
| L’utilisateur remplit le champ mot de passe | Le mot de passe doit être correct |  |
| L’utilisateur clique sur le bouton valider |  | L’utilisateur est authentifié et redirigé sur la page d’accueil |
| L’utilisateur clique sur my profile |  | L’utilisateur est redirigé sur la page profile ou son profile est affiché |
| L’utilisateur clique sur l’un des boutons de modification de |  | Les champs de valeur sont remplacés par des entrée de formulaire et un bouton valider apparait |
| L’utilisateur clique sur le bouton valider |  | La page est rechargée avec les informations mise à jour |

Extension : Le champ sont mal remplis

|  |  |  |
| --- | --- | --- |
| Action | Conditions particulières | Réaction |
| L’utilisateur clique sur le bouton valider |  | Une erreur décrivant le problème apparait sur le champ en question |

Extension : l’utilisateur accède sans se connecter

|  |  |  |
| --- | --- | --- |
| Action | Conditions particulières | Réaction |
| L’utilisateur rentre l’url de la page sans être connecter |  | Il est redirigé sur la page login |

Use case : l’utilisateur veut valider ou rejeter son covoiturage

Scénario

|  |  |  |
| --- | --- | --- |
| Action | Conditions particulières | Réaction |
| L’utilisateur clique sur le login de la navbar |  | L’utilisateur est redirigé sur la page login |
| L’utilisateur remplit le champ acronyme du login | L’utilisateur possédant cet acronyme doit être activé |  |
| L’utilisateur remplit le champ mot de passe | L’e mot de passe doit être correct |  |
| L’utilisateur clique sur le bouton valider |  | L’utilisateur est authentifié et redirigé sur la page d’accueil |
| L’utilisateur clique sur my carpooling |  | L’utilisateur est redirigé sur la page carpooling ou ses covoiturages sont affichés |
| L’utilisateur clique sur l’un des boutons de modification de |  | Les champs de valeur sont remplacés par des entrée de formulaire et un bouton valider apparait |
| L’utilisateur clique sur le bouton valider |  | La page est rechargée avec les informations mise à jour |

Extension : Le champ sont mal remplis

|  |  |  |
| --- | --- | --- |
| Action | Conditions particulières | Réaction |
| L’utilisateur clique sur le bouton valider |  | Une erreur décrivant le problème apparait sur le champ en question |

Extension : l’utilisateur accède sans se connecter

|  |  |  |
| --- | --- | --- |
| Action | Conditions particulières | Réaction |
| L’utilisateur rentre l’url de la page sans être connecter |  | Il est redirigé sur la page login |

Extension : l’utilisateur refuse le covoiturage

|  |  |  |
| --- | --- | --- |
| Action | Conditions particulières | Réaction |
| L’utilisateur clique sur le bouton reject |  | La page est rechargée avec les informations mise à jour |

## Etude de faisabilité

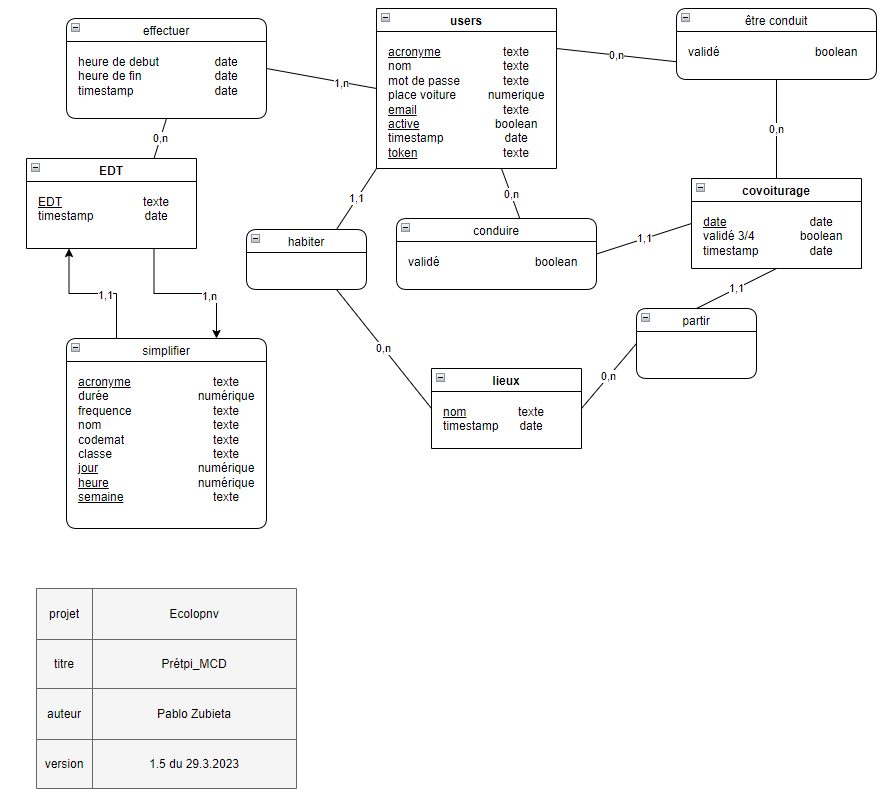
Il existe une infinité de manière de faire un site web. Utiliser un CMS, une diversité de langage allant du JS au Python en passant par le Ruby, Avec des Framework ou en natif, Avec du sql ou du nosql. Ayant beaucoup apprécier le PHP c’est vers ce langage que je me suis orienté.

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 est l’envie de défi c’est important de sortir de sa zone de confiance quand on est dans un processus d’apprentissage.

## MCD



## Nom du site et du domaine

[http://covoiturage.mycpnv.ch](http://covoiturage.mycpnv.ch/)   domaine louer par le CdP, meme si j’aimais bien Ecolopnv.

## 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.

Listes des tests

|  |  |
| --- | --- |
| Nom | Description |
| user login | Given :a user in the database When :enter his username and his right password Then :he’s log in the Website |
| wrong password | Given :a user in the database When :enter his Email and his wrong password Then :he’s not log in the website and informe that he got his password wrong |
| Blank log | Given :login left blank When :try to confirm Then :the user can’t enter |
| user entry | Given :given any user When :the user enter his email address his town, the number of seat in his car and his password twice Then :he is inserted in the databases |
| sign in log | Given :user conforme information When :he sign in Then :he get log |
| different passwords | Given :two different input on the password field When :user validated Then :the not the same password error is display to the user |
| entry existe | Given :the user has enter a username already in the database When :he validated Then :account already exist error is display to the user |
| user phasing | Given :the user is loged in When :he is on the home menu Then :his navbar is : my carpooling my profile my schedule logout |
| unconnected phasing | Given :the user is not log in When :he is on the home menu Then :his navbar is: log in sign in |
| user name | Given :the user is loged in When :he is navigating Then :his name appered on the navbar |
| homepage content | there is the homepage content |
| navbar | there is a nav bar |
| gabarit | there is a gabarit |
| user edt | Given : a edt data When : the user is on his schedule page Then : he can read his schedule |
| create carpooling | Given : a place and a time When : the function is call Then : it create a carpooling entry with a driver and a users\_has carpooling entry with the passenger |
| carpooling general | Given : a time and the entry in the database When : the function is used Then : a carpooling is asigned to all the users in the database |
| one person carpooling | Given : a place or time where only one person need a carpooling When : the function is used Then : no entry will be created in the database (cause there’s ne carpooling himself) |
| schedule in db | Given : a user in the database When : he activate his account Then : a entry is created in the user has edt table with his schedule |
| Place creation | Given : a place not contain in the databases When : a user write it in his form Then : it’s inserted in the place table and the id link as a forein key to the user |
| Place created | Given :a place contain in the databases When : a user write it in his form Then : the id link as a forein key to the user |
| acces | i can acces the web via http://covoiturage.mycpnv.ch |
| no regression | there no regression between the local version and the online one |
| diferent driver | Given : a carpooling record  When :a new carpooling is sugested with the same users  Then :the driver can't be same as the carpooling record |
| new carpooling | Given : a list carpooling  When : some of them fail  Then : a new carpooling instance is generate |
| mail to driver | Given : the driver of a carpooling  When : generate the carpooling entry  Then : send a mail to the driver |
| mail to user | Given : a user of a carpooling  When : generate the carpooling entry  Then : send a mail to the user |
| mail token link | Given : a user  When : the carpooling entry is created and the mail send  Then : the mail contains a link with this user token |
| validate | Given : a user in a carpooling  When : he use the links he recived by mail  Then : the user carpooling entry is validate |
| validation view | Given : a user  When : use his mail link  Then : he his redirected to a validate webpage |

## 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

Etant donné que le projet se fait Agile la planification est sous forme de Sprint

Il n’y a pas vraiment de planification détaillé appart les dates des Sprint.

Reception du CDC 01.02

Sprint 1 du 09.02 au 24.02

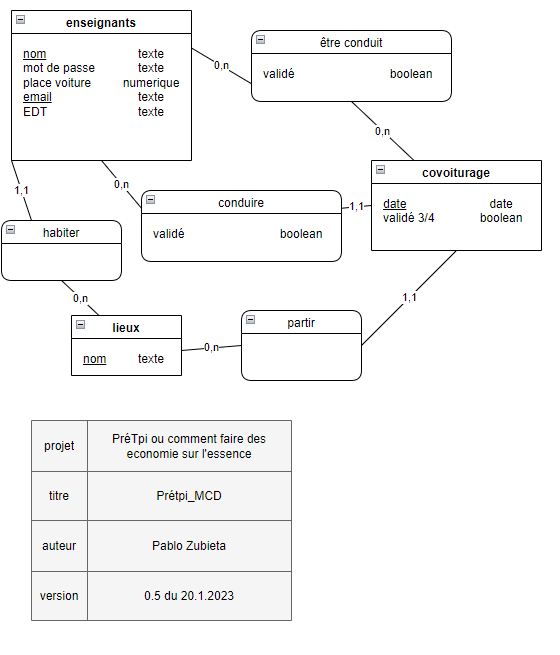
Sprint 2 du 28.03 au 10.03

Sprint 3 du 14.03 au 24.03

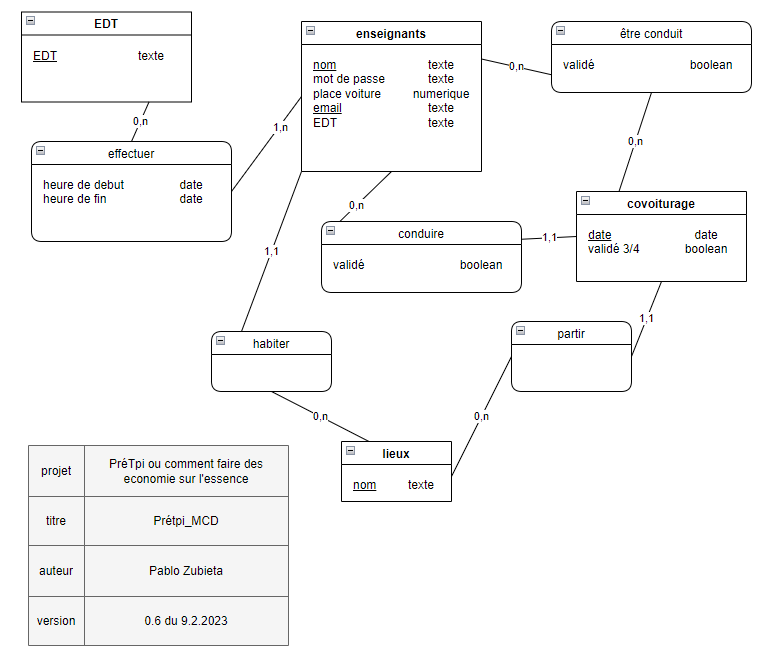
Et release du projet le 29.03

## Historique

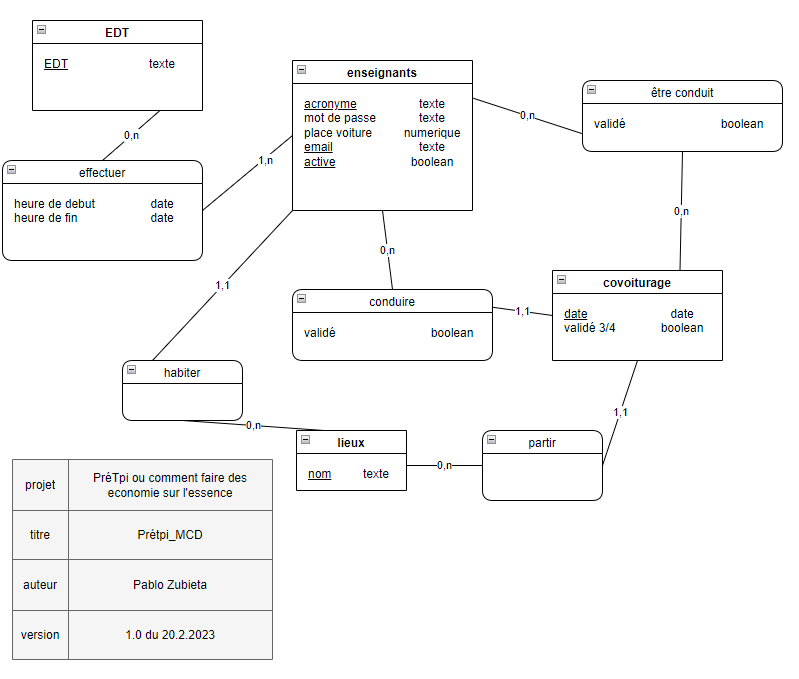
MCD première version du 20.01 obsolète depuis 09.02



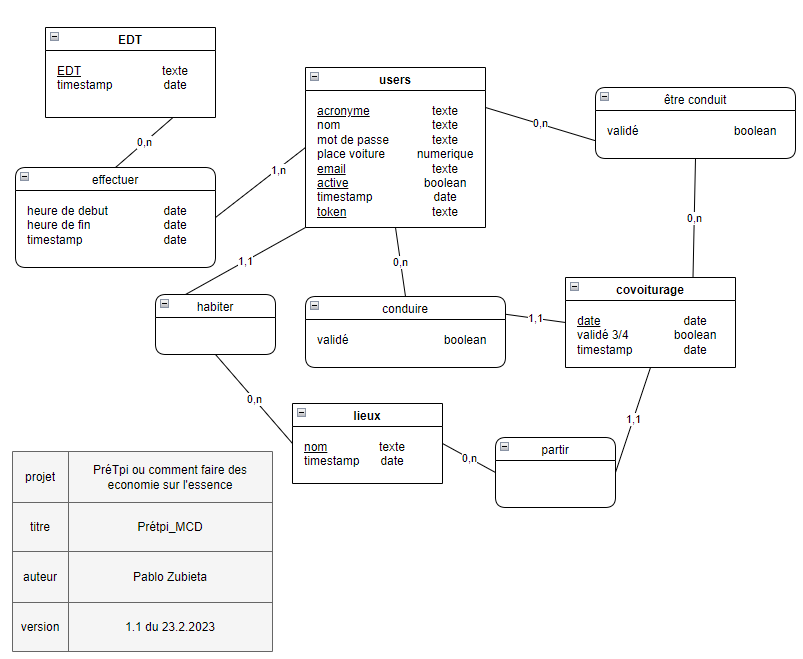
MCD deuxième version du 09.02 obsolète depuis 09.02



MCD troisième version du 20.02 obsolète depuis 23.02



MCD quatrième version du 23.02 obsolète depuis 29.03



MLD première version du 23.02 obsolète depuis 28.02

Une image contenant diagramme

Description générée automatiquement

# 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

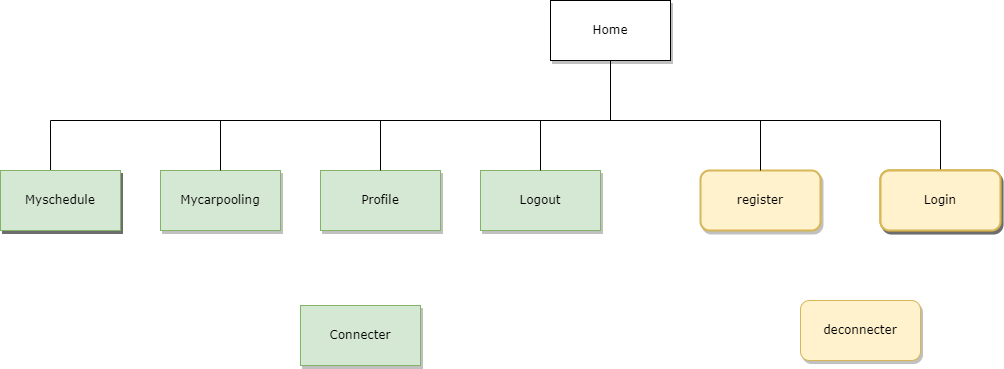
Hébergement

Php8.1

Apache/2.4.37

8.0.32-cll-lve

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



## Définition de la charte graphique

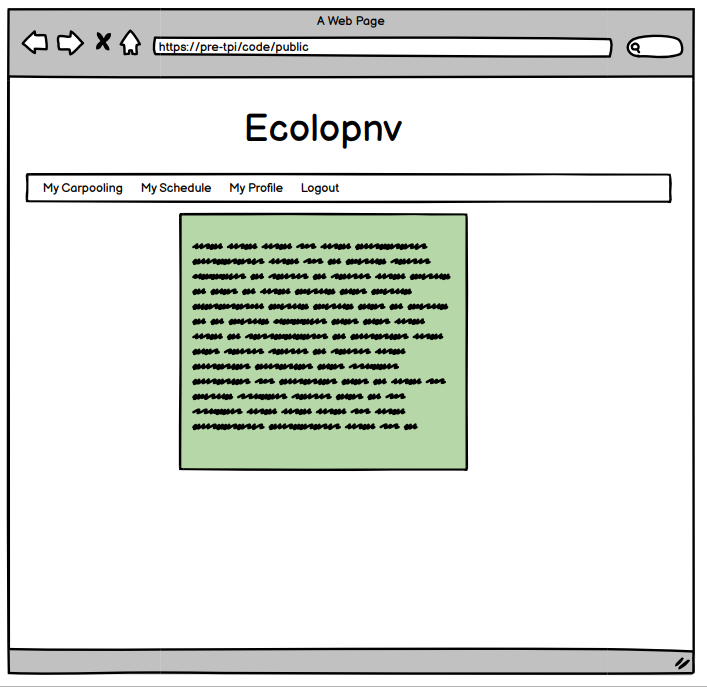
Pour ce qui est de la charte graphique J’ai pris arrangement de couleur autour du Vert, parce que vous savez l’écologie tout ça.



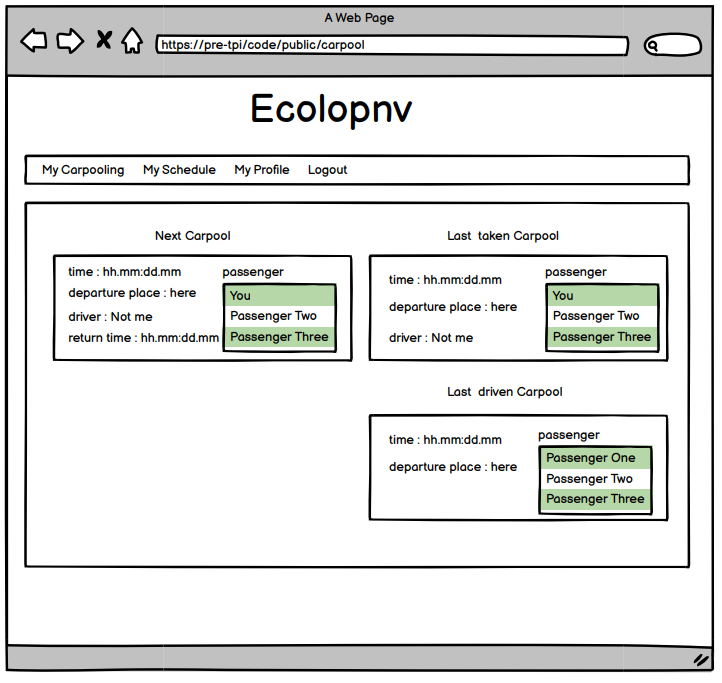
## Maquette graphique

Voici deux des wireframes de projet les autres seront en annexe.

La homepage :

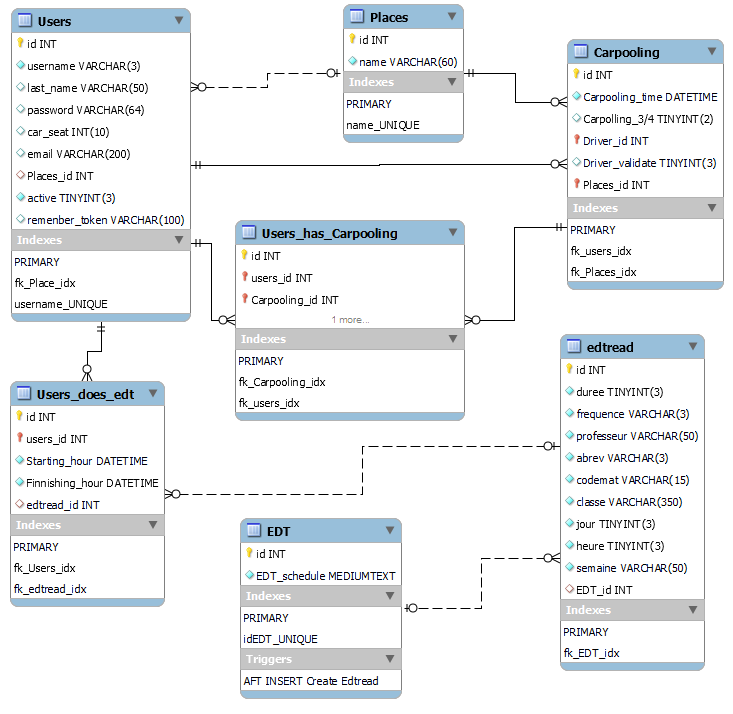


La Page Carpooling :



## Conception de la Base de données

Voici le MLD Actuel :



L’implémentation dans la base de données n’est pas exactement celle-ci, car

## Conception du Code

Voici un des diagrammes de flux intéressant les autres seront en annexe.

## Plugins et librairies

Etant donnée 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:\xampp\htdocs\Pretpi
2. Repository distant :
   * https://github.com/PabloZubieta/Pre\_Tpi

### Dossier bases de données

|  |  |
| --- | --- |
| **Nom** | **Description** |
| CreateDB.sql | Script de création de la base de données (DB) |
| MCD\_V4 | Fichier contenant le MCD de la DB |
| MLD\_V2 | Fichier contenant le MLD de la DB |
| Edt\_insertion.sql | Script d’insertion de l’EDT dans la base de données |
| EPX\_CRS\_SEMAINE | Ficher texte de l’edt |

### Dossier documentation

|  |  |
| --- | --- |
| **Nom** | **Description** |
| Pré-TPI-22-Zubieta\_Plateforme de partage de co-voiturage | Cahier des charges |
| Timesheet.html | Journal de travail |
| "documentation.pdf" | La documentation que vous êtes entrain de lire |
| "Journal de Bord.pdf" | Le journal de bord du projet |

### Dossier code

|  |  |
| --- | --- |
| **Nom** | **Description** |
| App/http/controller : | Tous les fichier de Classe Controllers de ce site |
| CarpolingController.php |
| UserController.php |
| PlaceController.php |
| ScheduleController.php |
| Users\_has\_carpooling  Controller.php |
| App/Model : | Tous les fichier de Classe Model de ce site |
| Carpoling.php |
| User.php |
| Place.php |
| Schedule.php |
| Users\_has\_carpooling.php |
| Route/web | Toutes les Routes de ce site |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

### Autres fichiers

|  |  |
| --- | --- |
| **Nom** | **Description** |
| dailytask.php | Script php d’execution journalière |

## Description des tests effectués

Listes des tests

|  |  |
| --- | --- |
| Nom | Description |
| user login | Success |
| wrong password | Success |
| Blank log | Success |
| user entry | Success |
| sign in log | Success |
| different passwords | Success |
| entry existe | Success |
| user phasing | Success |
| unconnected phasing | Success |
| user name | Success |
| homepage content | Success |
| navbar | Success |
| gabarit | Success |
| user edt | Success on dev Failed in prod |
| create carpooling | Success on dev Failed in prod |
| carpooling general | Success |
| one person carpooling | Success |
| schedule in db | Success |
| Place creation | Success |
| Place created | Success |
| acces | Success |
| no regression | Success |
| diferent driver | Failed |
| new carpooling | Failed |
| mail to driver | Failed |
| mail to user | Failed |
| mail token link | Failed |
| validate | Failed |
| validation view | Failed |

## Erreurs restantes

|  |  |  |  |
| --- | --- | --- | --- |
| *erreur* | *Description* | *Conséquences* | *Action* |
| *Middleware* | *Depuis le navigateur en insérant l’url l’utilisateur peut accéder à des routes qui ne doivent pas se produire.* | *Exemple un utilisateur non inscrit peut inscrire /logout ce qui va lui produire une erreur.* | *Faire des recherche sur le fonctionnement du middleware de laravel, corriger les routes en fonction* |
| *Nb place négative* | *Un utilisateur peut rentrer un nombre de place négative* | *Il est possible que l’algorithme de création d’un covoiturage puisse être altérer* | *Mettre une contrainte dans le formulaire de création et de modification sur* |

## Dossier d'archivage

*Décrire de manière détaillée les archives du projet.*

*Sur le rendu*

***Attention: les documents de réalisation doivent permettre à une autre personne de maintenir et modifier votre projet sans votre aide !***

# Mise en service

## Rapport de mise en service

*Fournir une description:*

*Viste du projet*

*+guide insta local*

* *de la publication chez l’hébergeur*
* *de l'installation du projet chez le client*
* *des tests officiels effectués chez le client et/ou par le client.*
* *des erreurs répertoriées   
  - description détaillée   
  - conséquences pour le client  
  - actions envisagées.*

## Liste des documents fournis

*Lister les documents fournis au client avec votre produit, en indiquant les numéros de versions*

* *le rapport de projet*
* *le manuel d'Installation (en annexe)*
* *le manuel d'Utilisation avec des exemples graphiques (en annexe)*
* *autres…*

# Conclusions

*Développez en tous cas les points suivants:*

* *Objectifs atteints / non-atteints*
* *Points positifs / négatifs*
* *Difficultés particulières*
* *Suites possibles pour le projet (évolutions & améliorations)*

# Annexes

## 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>

<https://www.educba.com/data-science/data-science-tutorials/sql-tutorial/>

<https://mariadb.org/wp-content/uploads/2023/02/MariaDBServerKnowledgeBase.pdf>

<https://www.youtube.com/watch?v=NrBJmtD0kEw>

## Manuel d'Installation

## Manuel d'Utilisation

## Wireframe

## Diagramme de flux

## Cahier des charges

## Archives du projet

<https://cloud.icescrum.com/p/PRETPIVOIT/#/taskBoard/211854/task/1079566>

[*https://github.com/PabloZubieta/Pre\_Tpi*](https://github.com/PabloZubieta/Pre_Tpi)