# Travler

Rapport TPI et documentation technique

***Mai 2020***

***Travail pratique individuel (TPI)***

***Thomas Fujise***

***Maître d’apprentissage : Dominique Aigroz***

Table des matières

[Travler 1](file:///C:\Users\Thomas\Desktop\PHP\ProjectTPI\docs\technical%20docs\documentation.docx#_Toc40624406)

[Versions 3](#_Toc40624407)

[Introduction 3](#_Toc40624408)

[Rappel de l’énoncé 3](#_Toc40624409)

[Organisation 3](#_Toc40624410)

[Méthodologie 3](#_Toc40624411)

[Product backlog 3](#_Toc40624412)

[Planification 3](#_Toc40624413)

[Maquettage 4](#_Toc40624414)

[Login 4](#_Toc40624415)

[Implémentation 4](#_Toc40624416)

[Base de données 4](#_Toc40624417)

[Structure du projet 4](#_Toc40624418)

[Classes (PHP) 4](#_Toc40624419)

[Librairies et outils externes 4](#_Toc40624420)

[Fonctionnalités principales 5](#_Toc40624421)

[Tests 5](#_Toc40624422)

[Conclusion 5](#_Toc40624423)

## Versions

## Introduction

Ce document présente différents aspects de la réalisation du projet « Travler ». Ce projet a été réalisé dans le cadre de mon Travail Pratique Individuel (TPI) 2020. Il a comme objectif de valider toutes les compétences que j’ai acquises durant ma formation au CFPT en tant qu’Informaticien CFC en développement d’application.

« Travler » est une application web, destinée aux motards, permettant de partager des itinéraires avec les autres utilisateurs de l’application. Les utilisateurs peuvent utiliser plusieurs fonctionnalités implémentées sur le site afin de créer leurs propres itinéraires. Ils ont également la possibilité de commenter et de noter les itinéraires des autres utilisateurs.

## Rappel de l’énoncé

### Matériel et logiciels à disposition

* Un PC standard école avec Windows 10 ,2 écrans
* Serveur Web (Apache avec module PHP et MySQLServer)
* SGBD (MySQLWorkbench)
* IDE (Visual Studio Code avec extensions)
* Logiciel de création de diagramme (MySQLWorkbench, dbdiagram.io)
* Logiciel de création de maquettes d’interfaces (Balsamiq)
* Outil de versionnage de projet (Git avec dépôt distant sur GitHub)
* Outil de bureautique pour les documents (Suite Office)
* API pour gérer les maps ([MapQuest](https://developer.mapquest.com/))
* Navigateur Web (Google Chrome, Microsoft Edge)

### Description de l’application

Travler est une application web, développée à l’aide des langages suivants : PHP, HTML 5, CSS 3, SQL, Javascript et avec l’API « MapQuest » pour gérer les maps. Le but de l’application est de partager des itinéraires réalisables à moto avec les autres utilisateurs. Chaque utilisateur est libre de noter de 0 à 10 et de commenter les itinéraires des autres utilisateurs. Le créateur de l’itinéraire reçoit un mail à chaque fois qu’un autre utilisateur commente son itinéraire. En tant qu’utilisateur non connecté, il ne peut accéder qu’à la page d’accueil où sont affichés tous les itinéraires avec un système de filtrage. Lorsque l’utilisateur clique sur le titre d’un itinéraire, tous les détails de celui-ci sont affichés avec les éventuelles images qui y sont attachées.

### Livrables

|  |  |
| --- | --- |
| Pour le responsable de projet et experts : | Pour le responsable de projet seulement : |
| * Planning | * Accès au dépôt distant sur GitHub |
| * Manuel utilisateur | * Modèles conceptuel et physique |
| * Journal de bord | * Protocole de test |
| * Résumé du projet |  |
| * Rapport du projet |  |

## Organisation

### Méthodologie

### Product backlog

|  |  |
| --- | --- |
| **ID & Nom** | **01 : Mise en place d’un dépôt GIT** |
| **Description (user story)** | En tant que développeur, je souhaite pouvoir accéder à un dépôt distant hébergé sur GitHub. |
| **Critère d’acceptation** | Le dépôt Git est configuré dans le répertoire du projet. Le projet est accessible depuis GitHub à l’adresse indiquée dans la documentation technique |
| **Importance** | Bloquant |

|  |  |
| --- | --- |
| **ID & Nom** | **02 : Création du MCD** |
| **Description (user story)** | En tant que développeur, je souhaite avoir un modèle conceptuel pour pouvoir implémenter la base de données |
| **Critère d’acceptation** | Le modèle conceptuel est créé et est utilisable pour implémenter la base de données |
| **Importance** | Bloquant |

|  |  |
| --- | --- |
| **ID & Nom** | **03 : Implémentation de la base de données** |
| **Description (user story)** | En tant que développeur, je souhaite pouvoir utiliser une base de données MySQL respectant le modèle créé au préalable. Je dois pouvoir utiliser une classe PHP me permettant d’effectuer des opérations sur la base de données. Je souhaite également avoir un dump de ma base de données sur le dépôt distant afin de pouvoir la recréer si nécessaire. |
| **Critère d’acceptation** | La base de données est créée et correspond au modèle. Une classe Database est mise en place et permet d’effectuer les opérations de base sur la base. Le dump est accessible sur le dépôt. |
| **Importance** | Bloquant |

### Planification

## Maquettage

Voici les différentes maquettes d’interfaces utilisateurs que j’ai effectués afin de pouvoir développer l’application web plus facilement

### Visiteur

#### Accueil

Voici l’interface utilisateur pour la page d’accueil de l’application. Elle répertorie les différents itinéraires qui ont été posté par les utilisateurs. Un système de filtrage en disponible en haut de la page. L’utilisateur peut voir les détails d’un itinéraire en cliquant sur le nom de celui-ci.

4

3

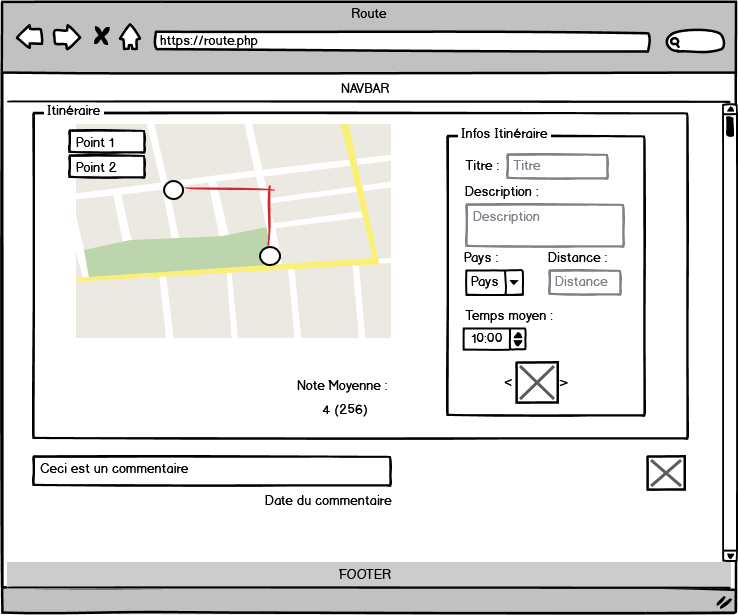
2

1

1. Le pays de départ de l’itinéraire.
2. La note moyenne minimum de l’itinéraire.
3. La distance minimum et maximum désirée pour l’itinéraire.
4. Le temps minimum et maximum désiré pour l’itinéraire.

#### Itinéraire

La page itinéraire qui permet d’afficher les détails d’un itinéraire, elle affiche également tous les commentaires publiés par les utilisateurs et la note moyenne de l’itinéraire.



10

9

8

7

6

5

4

3

2

1

1. Les différents points de l’itinéraire
2. Titre de l’itinéraire
3. Description de l’itinéraire
4. Pays de départ de l’itinéraire
5. Distance de l’itinéraire
6. Temps moyen pour effectuer l’itinéraire
7. Les images publiées par l’utilisateur sur l’itinéraire
8. La note moyenne de l’itinéraire
9. Les commentaires des utilisateurs avec la date, l’heure et le nickname de l’utilisateur
10. L’avatar de l’utilisateur qui a publié le commentaire

### Utilisateur

### Administrateur

## Implémentation

### Base de données

### Structure du projet

### Classes (PHP)

#### Classe tUser

Cette classe contient toutes les informations des différents utilisateurs.

|  |  |  |
| --- | --- | --- |
| Variables | Description | Type |
| nickname | Le pseudo de l’utilisateur | STRING |
| email | L’email de l’utilisateur | STRING |
| country | Le code du pays de résidence de l’utilisateur | STRING |
| birthday | La date d’anniversaire de l’utilisateur | TIMESTAMP |
| role | Le code du rôle de l’utilisateur | INT |
| state | L’état du compte de l’utilisateur | INT |
| activation | Le code qui détermine si le compte est activé | INT |

### Manager (PHP)

#### UserManager

Ce manager contient toutes les fonctions traitant et/ou récupérant des données en relation avec un utilisateur.

##### connectUser

Fonction qui permet de connecter l’utilisateur.

|  |  |  |
| --- | --- | --- |
| Paramètre(s) | Type | Valeur retournée |
| email | STRING |  |
| password | STRING | True si OK, False si un champ est incorrect |

## Librairies et outils externes

## Fonctionnalités principales

## Tests

## Conclusion