|  |
| --- |
| AzuriaPlanning |

Table des matières

[1 Analyse préliminaire 3](#_Toc131018822)

[1.1 Introduction 3](#_Toc131018823)

[1.2 Objectifs 3](#_Toc131018824)

[1.2.1 Fonctionnalités générales : 3](#_Toc131018825)

[1.2.2 Fonctionnalités détaillées selon le type d’utilisateur : 3](#_Toc131018826)

[1.3 Planification initiale 4](#_Toc131018827)

[2 Analyse / Conception 4](#_Toc131018828)

[2.1 Concept 4](#_Toc131018829)

[2.2 Stratégie de test 4](#_Toc131018830)

[2.3 Risques techniques 5](#_Toc131018831)

[2.4 Planification 5](#_Toc131018832)

[2.5 Dossier de conception 5](#_Toc131018833)

[3 Réalisation 6](#_Toc131018834)

[3.1 Dossier de réalisation 6](#_Toc131018835)

[3.2 Description des tests effectués 6](#_Toc131018836)

[3.3 Erreurs restantes 6](#_Toc131018837)

[3.4 Liste des documents fournis 7](#_Toc131018838)

[4 Conclusions 7](#_Toc131018839)

[4.1 Objectifs atteints 7](#_Toc131018840)

[4.2 Objectifs non-atteints 7](#_Toc131018841)

[4.3 Points positifs 7](#_Toc131018842)

[4.4 Points négatifs 7](#_Toc131018843)

[4.5 Difficultés particulières 7](#_Toc131018844)

[4.6 Suites possibles pour le projet 8](#_Toc131018845)

[5 Annexes 9](#_Toc131018846)

[5.1 Résumé du rapport du TPI / version succincte de la documentation 9](#_Toc131018847)

[5.2 Sources – Bibliographie 9](#_Toc131018848)

[5.3 Journal de travail 9](#_Toc131018849)

[5.4 Manuel d'Installation 9](#_Toc131018850)

[5.4.1 Acquisition des requis 9](#_Toc131018851)

[5.4.2 Configuration de PHP 9](#_Toc131018852)

[5.4.3 Installation et configuration de la base de données 9](#_Toc131018853)

[5.4.4 Installation des dépendances 9](#_Toc131018854)

[5.4.5 Démarrage du build CSS en temps réel 10](#_Toc131018855)

[5.4.6 Démarrage du serveur de développement PHP 10](#_Toc131018856)

# Analyse préliminaire

## Introduction

Dans le cadre de ma 4ème année d’apprentissage au CPNV en tant qu’informaticien, je dois de réaliser un TPI (Travail Pratique Individuel).

En préparation, je dois réaliser un projet Pré-TPI durant le 3ème trimestre (07.02.23 – 31.03.2023) pour un total de 60h de travail.

## Objectifs

### Fonctionnalités générales :

* Consultation du futur voyage, activités et élèves.
* Authentification et redirection vers la page d’accueil en fonction du type d’utilisateur.

### Fonctionnalités détaillées selon le type d’utilisateur :

* En tant que visiteur (personne non authentifiée) :
  + Consultation d’une page d’accueil.
  + Possibilité d’authentification via une page de connexion.
* En tant qu’élève
  + Consultation d’une page d’accueil.
  + Consultation d’une liste d’activités via une page.
  + Consultation d’une liste d’élève via une page.
  + Possibilité de déconnexion via un bouton.
* En tant qu’enseignant en plus des fonctionnalités accessibles aux élèves :
  + Possibilité de créer un nouveau voyage.
  + Possibilité d’importer des élèves via un fichier CSV

## Planification initiale

*Ce chapitre montre la planification du projet. Celui-ci peut être découpé en tâches qui seront planifiées. Il s'agit de la première planification du projet, celle-ci devra être revue après l'analyse. Cette planification sera présentée sous la forme d'un diagramme.*

*Ces éléments peuvent être repris des spécifications de départ.*

# Analyse / Conception

## Concept

*Le concept complet avec toutes ses annexes:*

*Par exemple :*

* *Multimédia: carte de site, maquettes papier, story board préliminaire, …*
* *Bases de données: interfaces graphiques, modèle conceptuel.*
* *Programmation: interfaces graphiques, maquettes, analyse fonctionnelle…*
* *…*

## Stratégie de test

* Tests Système
  + Tests fonctionnels
    - Quoi : Tests des différents scénarios.
    - Qui : Axel Pittet ([Axel.Pittet@cpnv.ch](mailto:Axel.Pittet@cpnv.ch)).
    - Quand : Lors de la fin du Sprint n°2 et lors de la réalisation complète du produit.
    - Pourquoi : Afin de vérifier que le produit fonctionne complètement.
  + Tests de robustesse
    - Quoi : Tests de situations anormales et de sécurité.
    - Qui : Luke Cornaz ([Luke.Cornaz@cpnv.ch](mailto:Luke.Cornaz@cpnv.ch)).
    - Quand : Une fois les tests fonctionnels effectués.
    - Pourquoi : Afin de s’assurer que le produit est équipé contre de possibles erreurs utilisateurs ou attaques pirates basiques.

## Risques techniques

Je ne vois pas vraiment de risques techniques pour ce projet Pré-TPI. Le sujet me semble assez clair et je connais toutes les technologies.

Mise à part cela, j’aimerai m’éloigner un peu plus de ce qu’on a appris au CPNV pour faire quelque chose de plus propre, notamment au niveau des contrôleurs et des modèles en utilisant des classes statiques.

Cela peut accélérer mon travail comme dans la rapidité d’écriture, lecture et utilisation ou peut-être le plomber si difficultés rencontrées.

## Planification

Ma planification se trouve en annexe sur :

<IceScrum> (https://icescrum.cpnv.ch/p/VYGEDU/#/project).

## Dossier de conception

**Matériel** : 1 Ordinateur type CPNV

**Environnement de développement** : Visual Studio Code

**Conception du MCD** : Drawio

**Conception du MLD** : MySQL Workbench

**Hébergement de développement** : Serveur PHP local sur le réseau CPNV

**Hébergement de rendu** : SwissCenter

**Serveur BDD** : MySQL

**Consultation de la BDD** : HeidiSQL

**Framework CSS** : TailwindCSS et DaisyUI

**Langages utilisés** : PHP et JS

**Outils utilisés** : NodeJS

**Architecture** : MVC

# Réalisation

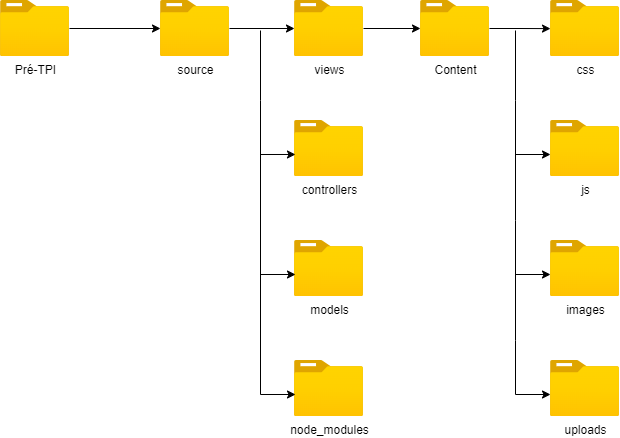
## Dossier de réalisation

**Version du produit** : 1.0.0

**Version OS :** Windows 10

**Version PHP** : 8.2.2

**Version NodeJS** :18.15.0



## Description des tests effectués

Mes tests se trouve en annexe sur :

<IceScrum> (https://icescrum.cpnv.ch/p/VYGEDU/#/project).

## Erreurs restantes

*S'il reste encore des erreurs:*

* *Description détaillée.*
* *Conséquences sur l'utilisation du produit.*
* *Actions envisagées ou possibles.*

## 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

## Objectifs atteints

Globalement toutes les fonctionnalités sont implémentées et fonctionnelles, le produit doit gérer les erreurs et les redirections correctement. L’hébergement est correctement configuré pour fonctionner de manière correcte et avec sécurité des fichiers sensibles.

## Objectifs non-atteints

Rien n’a pas pu être fait durant ce Pré-TPI, toutes les implémentations sont présentes et fonctionne comme expliqué dans le point 4.1.

## Points positifs

J’ai pu développer de mon côté une nouvelle manière de m’organiser et de développer. Grâce à ça, mon efficacité à développer et relire mon code de manière simple et épurée.

## Points négatifs

Je ne suis peut-être pas très satisfait du rendu visuel de mon produit, j’ai essayé de le rendre visuellement simple, malgré cela, le thème de couleurs ne me plaît pas réellement.

## Difficultés particulières

Pour causes de maladie, j’ai perdu énormément de temps de projet, ce qui m’a finalement beaucoup retardé sur mes tâches. Les rattrapés était particulièrement épuisant et stressant.

## Suites possibles pour le projet

Les suites possibles et améliorations seraient :

* Retravailler le code et sa lisibilité.
* Renforcer les sécurités.
* Améliorer le thème graphique.
* Fournir plus d’élément de modification aux enseignants.
* Implémenter des profils d’élèves. (photos, intolérances, maladies, …).
* Implémenter plus d’images pour les voyages et activités.

# Annexes

## Sources – Bibliographie

<https://stackoverflow.com> – Résolutions de problème

<https://www.php.net> – Documentation de PHP

<https://tailwindcss.com> – Installation et documentation de TailwindCSS

<https://daisyui.com> - Installation et documentation de DaisyUI

[https://nodejs.org](https://nodejs.org/en) – Installation de NodeJS

<https://github.com> – Déport de projet et versioning

<https://icescrum.cpnv.ch> – Plateforme de gestion du projet

<https://www.swisscenter.com> – Plateforme d’hébergement du produit

## Journal de travail

## Manuel d'Installation

### Acquisition des requis

Les requis pour la mise en place du projet sont :

* PHP 8.2
* Serveur MySQL
* NodeJS 18.15.0

### Configuration de PHP

La configuration est une configuration de base, mise à part pour l’extension « pdo\_mysql » qui doit être ajoutée dans le fichier « php.ini »

Une image contenant texte

Description générée automatiquement

### Installation et configuration de la base de données

Il faut exécuter le script SQL qui se trouve dans le dossier du projet à la racine. Il créera toutes les tables nécessaires.

Pour le développement, l’utilisateur MySQL est « tpi » avec un champ mot de passe vide n’ayant accès que à la BDD « tpi ».

### Installation des dépendances

Pour l’installation des dépendances, il faudra utiliser la commande :

« npm i » ou « npm install »

Les dépendances seront ensuite téléchargées et implémentés.

### Démarrage du build CSS en temps réel

Tailwind actualise le fichier CSS de développement à chaque sauvegarde d’un fichier contenant du code html (parmi des répertoires qui lui sont fournis.).

Pour exécuter cela, la commande est la suivante :

« npx tailwindcss -i ./views/content/css/tailwind.css -o ./views/content/css/output.css –watch »

### Démarrage du serveur de développement PHP

Un serveur de développement sur le réseau local est utilisé pour le développement, la commande pour le démarrer est la suivante :

php -S [IP]:[Port]

Les parenthèses crochet ne sont ici que pour la démonstration, elles ne doivent pas figurer dans la commande à utiliser.