CalendarEnMieux

Documentation TPI

Keanu Trosset

2021

Table des matières

CPNV | SI-C4b

[1 Analyse préliminaire 3](#_Toc476150016)

[1.1 Introduction 3](#_Toc476150017)

[1.2 Organisation 3](#_Toc476150018)

[1.3 Objectifs 3](#_Toc476150019)

[1.4 Planification initiale 3](#_Toc476150020)

[2 Analyse 3](#_Toc476150021)

[2.1 Cahier des charges détaillé 3](#_Toc476150022)

[2.2 Définition de l’audience 4](#_Toc476150023)

[2.3 Analyse concurrentielle 4](#_Toc476150024)

[2.4 Cas d’utilisation 4](#_Toc476150025)

[2.5 Etude de faisabilité 5](#_Toc476150026)

[2.6 MCD 5](#_Toc476150027)

[2.7 Nom du site et du domaine 7](#_Toc476150028)

[2.8 Stratégie de test 7](#_Toc476150029)

[2.9 Budget initial 7](#_Toc476150030)

[2.10 Planification détaillée 7](#_Toc476150031)

[2.11 Historique 8](#_Toc476150032)

[3 Conception 9](#_Toc476150033)

[3.1 Analyse de l’environnement 9](#_Toc476150034)

[3.2 Détermination de l’arborescence du site et des rubriques 9](#_Toc476150035)

[3.3 Définition de la charte graphique 9](#_Toc476150036)

[3.4 Maquette graphique 9](#_Toc476150037)

[3.5 Conception de la Base de données 9](#_Toc476150038)

[3.6 Conception du Code 9](#_Toc476150039)

[3.7 Plugins et librairies 9](#_Toc476150040)

[3.8 Choix de la formule d’hébergement 9](#_Toc476150041)

[4 Réalisation 11](#_Toc476150042)

[4.1 Dossier de réalisation 11](#_Toc476150043)

[4.2 Description des tests effectués 11](#_Toc476150044)

[4.3 Erreurs restantes 11](#_Toc476150045)

[4.4 Dossier d'archivage 11](#_Toc476150046)

[5 Mise en service 13](#_Toc476150047)

[5.1 Rapport de mise en service 13](#_Toc476150048)

[5.2 Liste des documents fournis 13](#_Toc476150049)

[6 Conclusions 13](#_Toc476150050)

[7 Annexes 14](#_Toc476150051)

[7.1 Sources – Bibliographie 14](#_Toc476150052)

[7.2 Journal de bord de chaque participant 14](#_Toc476150053)

[7.3 Manuel d'Installation 14](#_Toc476150054)

[7.4 Manuel d'Utilisation 14](#_Toc476150055)

[7.5 Archives du projet 14](#_Toc476150056)

# Analyse préliminaire

## *Introduction*

Ce projet a été fait dans le cadre de mon Travail Personnel Individuel, plus connu sous le nom de TPI. J’ai choisi ce sujet de projet parce que je trouve qu’il a l’air plus simple qu’il en a l’air. La gestion de l’interface et la facilité d’utilisation est un point clé et c’est ça qui m’a plus.

## *Organisation*

Candidat : Trosset Keanu [keanu.trosset@cpnv.ch](mailto:keanu.trosset@cpnv.ch)

Chef de projet : BENZONANA Pascal [Pascal.BENZONANA@cpnv.ch](mailto:Pascal.BENZONANA@cpnv.ch)

Expert 1 : MONTEMAYOR Ernesto [ernesto@bati-technologie.ch](mailto:ernesto@bati-technologie.ch)

Expert 2 : OBERSON Bernard [oberson.bernard@gmail.com](mailto:oberson.bernard@gmail.com)

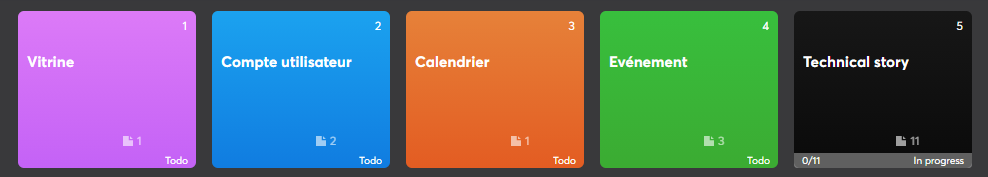
## *Objectifs*

1. Affichage du calendrier de façon journalière, hebdomadaire et mensuelle.
2. Création d’événement avec affichage avec gestion d’erreur.
3. Ergonomie et facilité d’utilisation du produit.
4. Travail déposé chaque vendredi sur un repository.
5. Modification d’un événement avec gestion d’erreur.
6. Suppression d’un événement.
7. Possibilité d’intégrer une récurrence à un événement sur une durée définie.

## Planification initiale

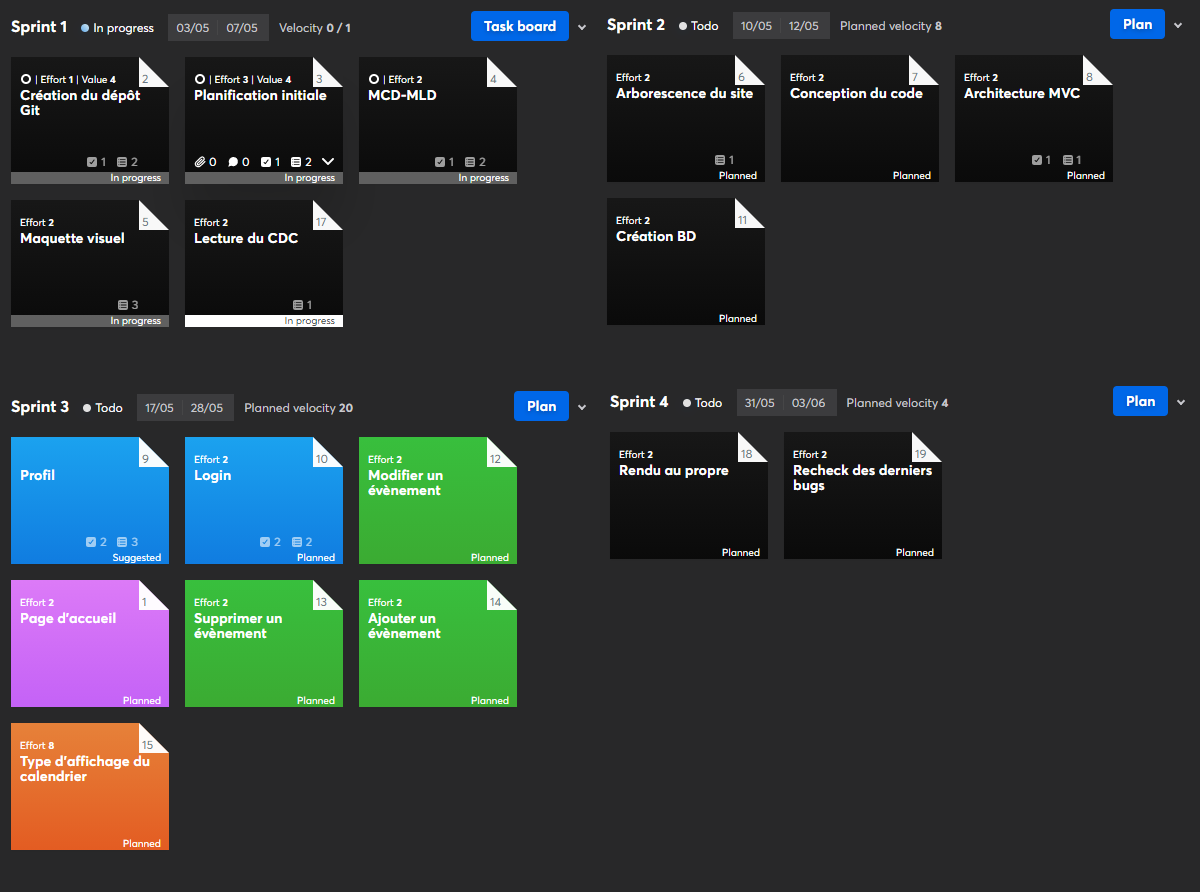
Voici ma planification initiale. Ce n’est pas un Gant comme d’habitude puisque je trouvais cela peu pertinent de faire une planification initiale en Waterfall sachant que je travaille en agile. Après quelque recherche sur le net sur le sujet j’ai pris la décision de prendre un screenshot de mon IceScrum.

Si cela avait été demander par les experts ou le chef de projet, cela aurait été facile de le transposé en gantt.

Les couleurs correspondent à ces sections :

En premier se trouve un tableau avec un descriptif plus parlant des sprints et ensuite les sprints détaillés avec les stories :

|  |  |  |  |
| --- | --- | --- | --- |
|  | Sujet | Date de début | Date de fin |
| Sprint 1 | Analyse | 03.05.2021 | 07.05.2021 |
| Sprint 2 | Conception | 10.05.2021 | 12.05.2021 |
| Sprint 3 | Réalisation | 17.05.2021 | 28.05.2021 |
| Sprint 4 | Rendu final | 03.05.2021 | 03.0.2021 |

**

# Analyse

## *Cahier des charges détaillé*

L’application Gestion de calendrier sera développée en PHP avec une base de données MySQL.

Le candidat fera une planification initiale où il définira les différents sprints avec les stories associées au fur et à mesure de l’avancement du TPI.

**Partie publique :**

L’utilisateur pourra voir un bref descriptif du site sur la page d’accueil et aura accès à une page de connexion (et d’inscription. Les éléments nécessaires pour l’inscription au site sont :

- Email

- Pseudo

- Mot de passe avec confirmation

Pour la connexion, seuls le mail et le mot de passe seront demandés.

L’utilisateur pourra s’inscrire s’il n’a pas de compte ou entrer directement ses données de connexion s’il en a un

**Partie personnelle :**

L’utilisateur après s’être connecté verra une page affichant la semaine en cours avec les événements qui y sont associés. Une popup avec une alarme s’affichera si l’heure et le jour auxquels s’est connecté l’utilisateur correspond à un événement proche (moins de 15 minutes) ou est en cours.

L’utilisateur pourra choisir le type d’affichage (jour, semaine, mois, année)

Il pourra également définir des types d’événements avec une couleur associée

***Gestion de son calendrier***

L’utilisateur pourra gérer son calendrier personnel avec les fonctionnalités suivantes :

- Ajouter un événement avec :

o le sujet ou nom de l’événement

o le lieu

o la date et l’heure de début et de fin avec une gestion d’erreur

o le type (professionnel, privé, etc.) avec la couleur associée

o Choisir la récurrence s’il y en a une et définir quelle type (jour, semaine, mois, année) avec une durée définie

- Éditer un événement existant. Il pourra modifier tous les paramètres cités ci-dessus.

- Supprimer un événement existant. Lors de la suppression d’un événement récurrent, l’utilisateur aura le choix de :

o supprimer seulement l’événement sélectionné

o supprimer l’occurrence sélectionnée et les suivantes

o supprimer toutes les occurrences liées à l’événement choisi

Le site devra être responsive et l’utilisation d’un template existant est autorisée

## *Définition de l’audience*

Cette application pourrait être un besoin inter-entreprises pour ne pas avoir à utiliser le calendrier Outlook par exemple.

## Analyse concurrentielle

La fonction calendrier de Outlook est un excellent exemple. Il est complet et marche super bien.

La personnalisation des évènements, des couleurs et de l’ensemble est très complète et j’aimerai bien refléter ça dans mon TPI.

Le problème que je ne voudrai pas reproduire c’est que si l’on cherche une fonctionnalité dans l’ajout d’un évènement, nous pouvons rapidement nous perdre dans toutes les fonctionnalités. J’aimerai que mon programme reste simple et rapide d’utilisation.

## Cas d’utilisation

|  |  |
| --- | --- |
| Identifiant | User |
| En tant que | User |
| Je veux | Naviguer sur le site |
| Pour | Regarder mon calendrier |
| Priorité | Normal |

| Action | Condition particulière | Réaction |
| --- | --- | --- |
| Cliquer sur le logo du site. |  | Affiche la page d’accueil. |
| Cliquer sur « Mon calendrier ». |  | Affiche la page du calendrier personnel s’ouvre si l’utilisateur est connecté. |
| Cliquer sur « Login » | Si la personne n’est pas logué. | Affiche la page pour se loguer et change le bouton en profil si la personne se logue. |
| Cliquer sur « Profil ». | Si la personne est logué. | Affiche la page de profil et change le bouton en login si la personne se déconnecté. |
| Cliquer sur « Contact ». |  | Affiche la page de contact. |

|  |  |
| --- | --- |
| Identifiant | Visiteur |
| En tant que | Visiteur |
| Je veux | Naviguer sur le site sans être connecter |
| Pour | Pouvoir accéder a des moyens de login sans le bouton « login » |
| Priorité | Normal |

| Action | Condition particulière | Réaction |
| --- | --- | --- |
| Cliquer sur le bouton Mon Calendrier |  | Affiche la page de login |

|  |  |
| --- | --- |
| Identifiant | User |
| En tant que | User |
| Je veux | Crée un évènement |
| Pour | Pouvoir voir mon anniversaire |
| Priorité | Normal |

| Action | Condition particulière | Réaction |
| --- | --- | --- |
| Je clique « Mon Calendrier ». | Je me suis déjà connecté au préalable. | La page « Mon Calendrier » s’affiche. |
| Je clique sur le bouton « ajouter un évènement ». |  | La page « Ajouter un évènement » s’affiche. |
| Je rentre les informations pour crée mon évènement. | Toutes les informations sont correctes. | L’évènement est créé et afficher sur le calendrier. |

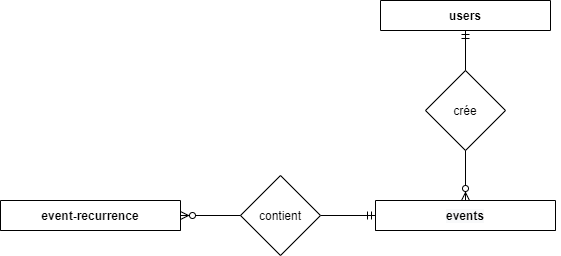
## Etude de faisabilité

Pour le template j’ai utilisé l’application NicePage qui permet de crée des pages HTML juste en déplaçant et en ajoutant ce qu’il appelle des « blocs ». Il y a ensuite la possibilité d’exporter le tout en HTML-CSS.

Le plus gros problème va être de rendre le site dynamique avec le JavaScript.

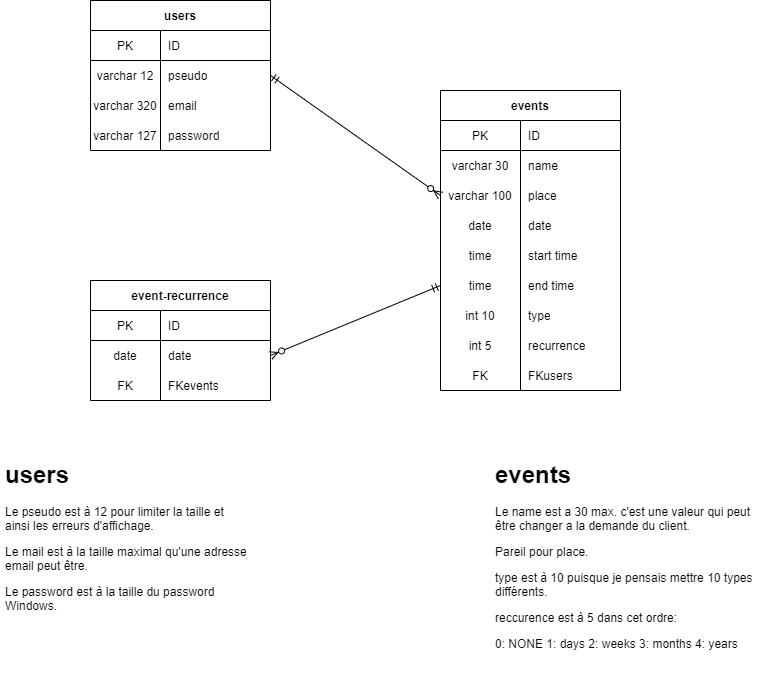
## MCD / MLD

Voici le MCD de mon projet. 3 tables y sont représentées.



|  |  |
| --- | --- |
| users | Contient les utilisateurs inscrits sur le site. Un utilisateur peut créer zéro ou plusieurs évènement. |
| events | Contient les évènements. Un évènement est créé par un utilisateur. Un évènement est détenu par un seul utilisateur. Un évènement contient zéro ou plusieurs récurrence. |
| event-recurrence | Contient les dates de récurrences de chaque évènement si il en a. une récurrence est détenu par un seul évènement. |

Voici le MLD de mon projet. Les commentaires en bas de l’image sont ici pour donner plus de précisions par rapport au champs des tables.



|  |  |  |
| --- | --- | --- |
| users | pseudo | Contient le nom d’utilisateur de la personne. |
| users | email | Contient l’email de la personne qui sera utilisé au moment de la connexion. |
| users | password | Contient le mot de passe. |
| events | name | Contient le nom de l’évènement. |
| events | place | Contient la place de l’évènement. |
| events | date | Contient la date de création de l’évènement. Format date standard |
| events | start time | Contient l’heure de début de l’évènement. Format time qui est en heure. |
| events | end time | Contient l’heure de fin de l’évènement. Format time qui est en heure. |
| events | type | Contient le type (Professionnel, privé, etc..) |
| events | recurrence | Contient la récurrence (tous les jours, semaines, mois, années) de l’évènement. |
| events | FKusers | Contient le nom de l’utilisateur qui a créé l’évènement. |
| event-recurrence | date | Contient les dates des récurrences. |
| event-recurrence | FKevents | Contient l’ID de l’évènement auxquelles il est rattacher. |

## Nom du site et du domaine

Je vais héberger mon site sur swisscenter. Je propose donc comme nom de domaine : CalenMieu.cpnv.ch

## *Stratégie de test*

|  |  |
| --- | --- |
| Identifiant | User |
| En tant que | Visiteur |
| Je veux | Crée un évènement |
| Pour | Afficher mon anniversaire sur le calendrier |
| Priorité | Normal |

| Action | Condition particulière | Réaction |
| --- | --- | --- |
| Je clique « Mon Calendrier ». | Je ne me suis déjà connecté au préalable. | La page « Login » s’affiche. |
| Je me logue donc sur la page de Login. |  | La page « Mon Calendrier » s’affiche. |
| Je clique sur le bouton « ajouter un évènement ». |  | La page « Ajouter un évènement » s’affiche. |
| Je rentre les informations pour crée mon évènement. | Pas toute mes informations sont correcte. | La page affiche que une information est incorrecte. |
| Je rentre les informations pour crée mon évènement. | Il manque une informations. | La page affiche que une information est vide. |

## *Budget initial*

L’accès à swisscenter est payant et payer par le CPNV.

L’accès à la suite office était agréable donc pris en compte.

J’utilise un éditeur de texte gratuit nommé Atom pour le code.

## *Planification détaillée*

*Révision de la planification initiale (Gantt) du projet :*

* *Planning indiquant les dates de début et de fin du projet ainsi que le découpage connu des diverses phases (jalons).*
* *Listes des tâches, durées et interdépendances.*

*Il s’agit ici de détailler la planification initiale du projet.   
Elle sera ensuite affinée régulièrement (ajout/suppressions de tâches, durées…). Si les délais doivent être ensuite modifiés, le responsable de projet doit être avisé, et les raisons doivent être expliqués dans l’historique.*

## *Historique*

Aucun changement du cahier des charges n’a été demandé pour le moment.

# Conception

## Analyse de l’environnement

Pour le template, j’ai utilisé une application nommé NicePage qui permet de crée son template et ensuite l’exporter en HTML.

Comme navigateur web pour le développement, j’ai utilisé Google Chrome. Pour certain test j’ai tout de même utilisé Edge et FireFox vers la fin.

Pour le code j’utilise un système d’exploitation Windows 10 avec l’éditeur de texte Atom. Il s’y trouve des librairies pour le bon fonctionnement :

* Php-server
* Php-debug
* Ide-php

Pour la gestion du code j’utilise le site web GitHub ainsi que l’application Github Desktop.

Pour la gestion du projet en méthode Agile, j’utilise le site web IceScrum.

Pour les maquettes visuelles j’ai utilisé l’application balsamiq.

Pour le MCD-MLD ainsi que l’architecture du site j’ai utilisé le site web Draw.io.

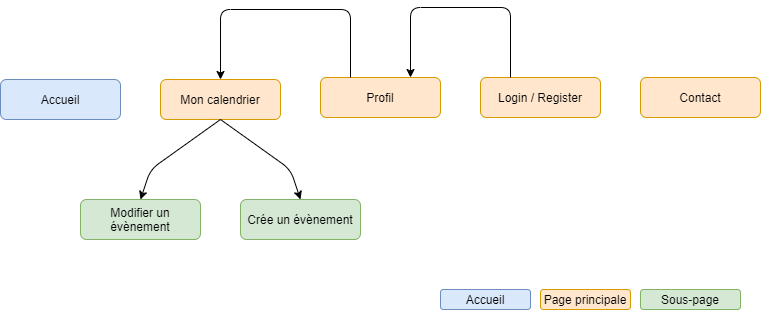
Pour ces documents ainsi que le journal de travail, j’ai utilisé la suite office et plus spécialement Word et Excel.

Pour la lecture et l’envoie de PDF, j’ai utilisé l’application Acrobat Reader.

Pour éditer la base de donnée, j’ai utilisé l’application HeidiSQL qui est un outil d’administration de base de données.

Pour la gestion de fichier sur swisscenter, j’ai utilisé l’application Filezilla

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



## Définition de la charte graphique

C’est ici que je vais parler du choix de mon template. J’ai choisi cet agencement de page parce que je voulais une page d’accueil qui montre que l’outil est pro. Sachant en plus que pour voir le calendrier il faut être loger.

Pour l’affichage du calendrier j’ai choisi un design assez sobre.

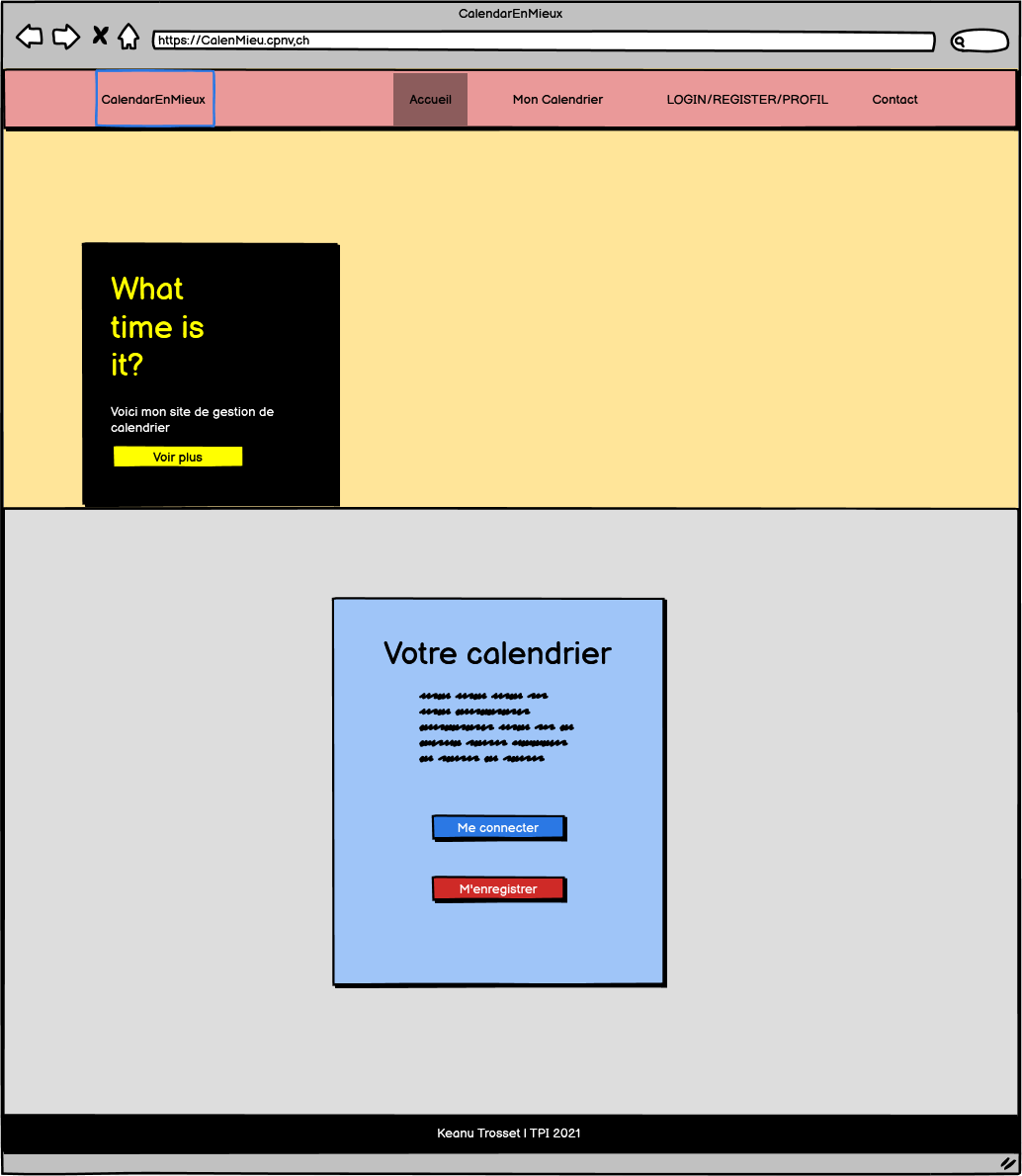
La couleur bleu et rouge/jaune est assez présente dans ce projet.

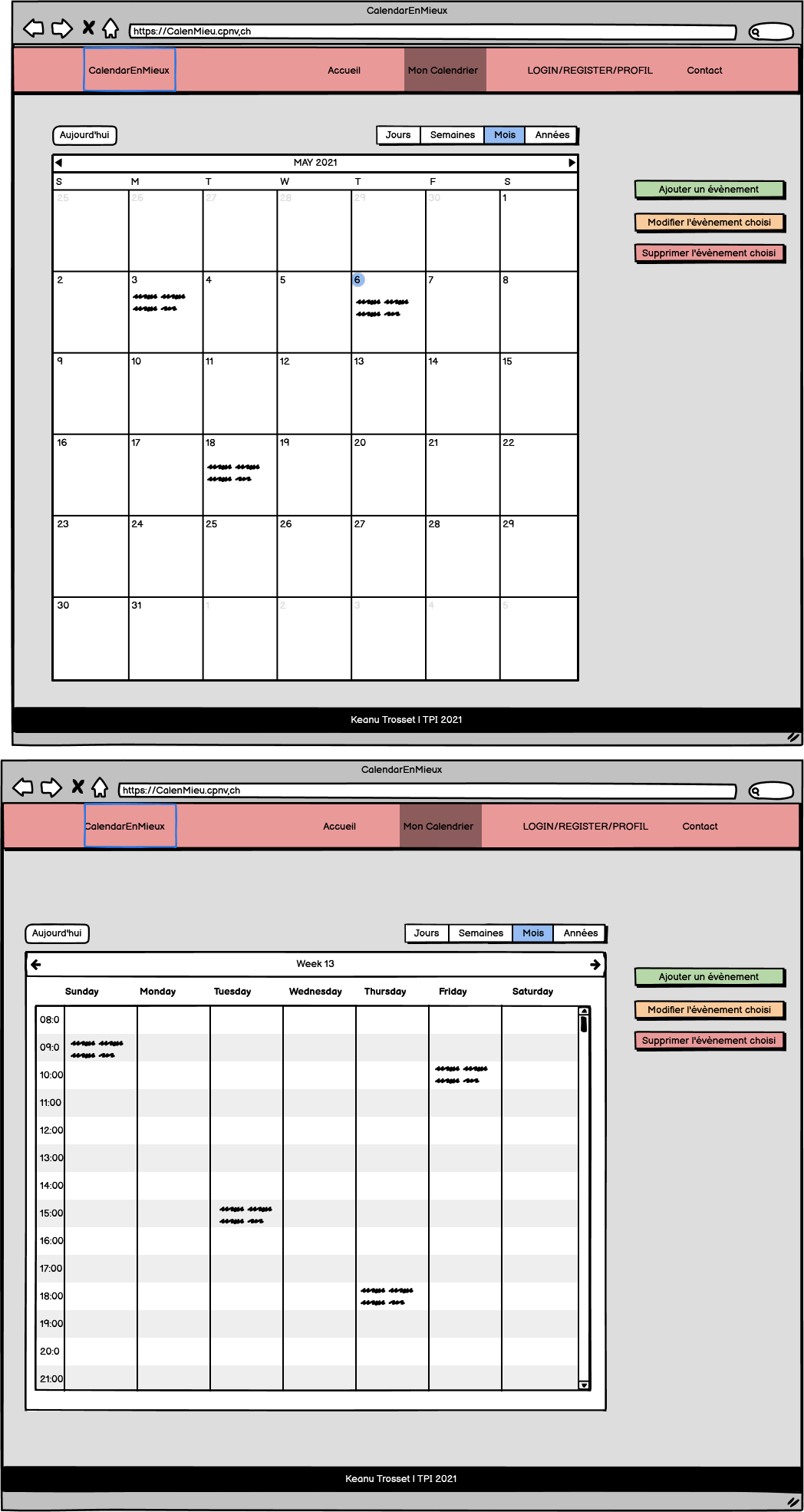
Le bleu est, je trouve, assez passe partout. Il permet de souligner un point important ou juste attirer l’œil.

Le rouge est un choix personnel avec le jaune qui s’en rapproche pour ne pas utiliser trop de couleur différente.

Il y a eu parfois un peu de vert. Il est là par exemple pour le bouton « ajouter un évènement ». J’ai trouvé cela pertinent d’en mettre à ce moment-là.

## Maquette graphique

**Voici la page accueil. Quand nous arrivons sur le site nous avons un petit texte expliquant ce qu’est le site avec un bouton voir plus qui nous fait juste aller plus bas sur la page vers la prochaine section. Celle-ci invite l’utilisateur à se connecter où se crée un compte pour commencer à utiliser le calendrier.

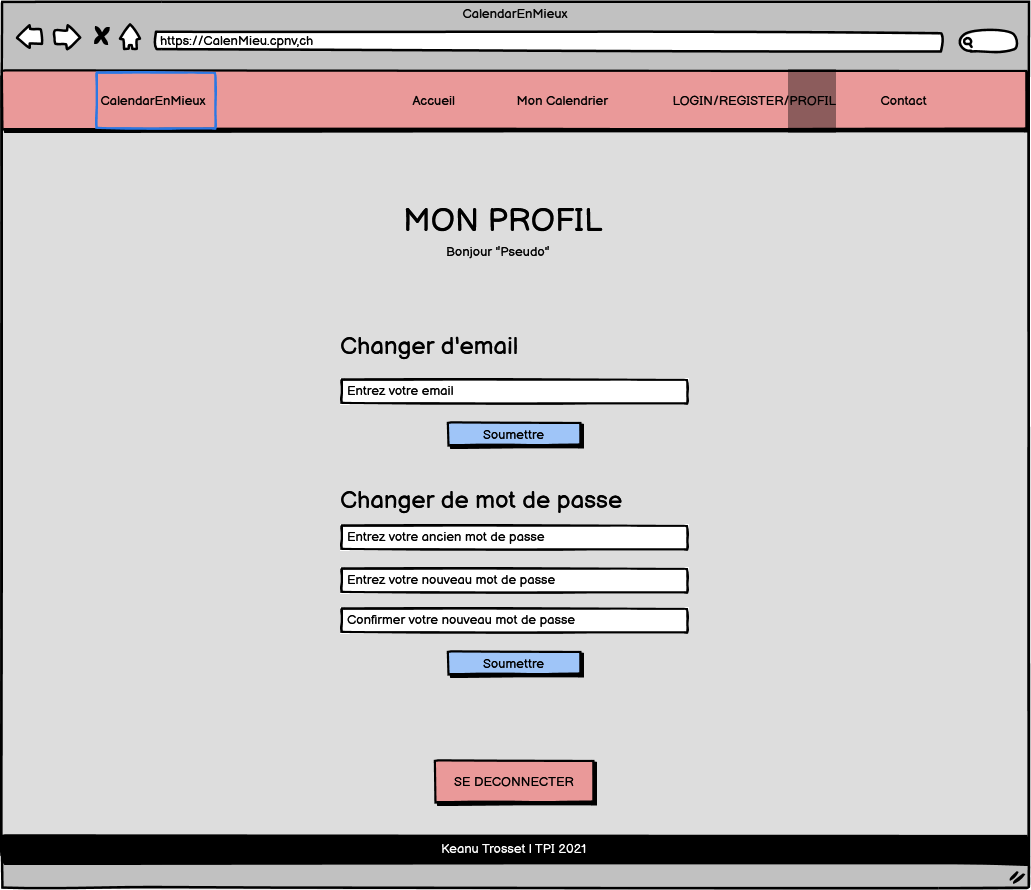
**Ici nous retrouvons la fonctionnalité principale de mon site web. Il y a des boutons pour ajouter, modifier ou supprimer des évènements.

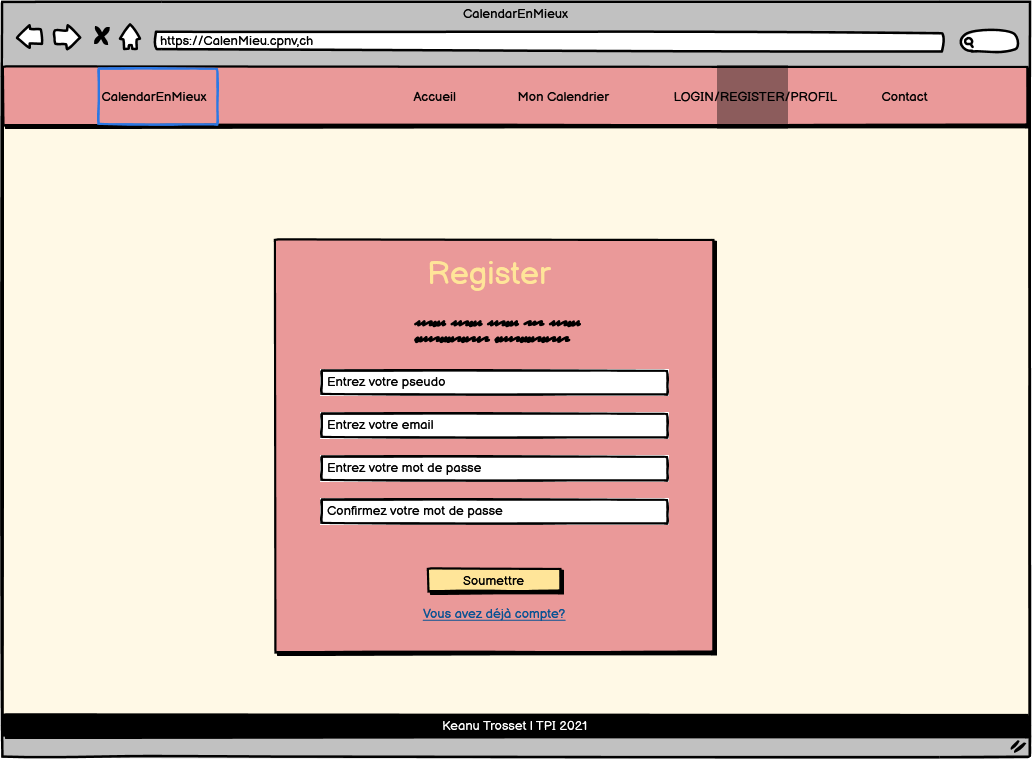
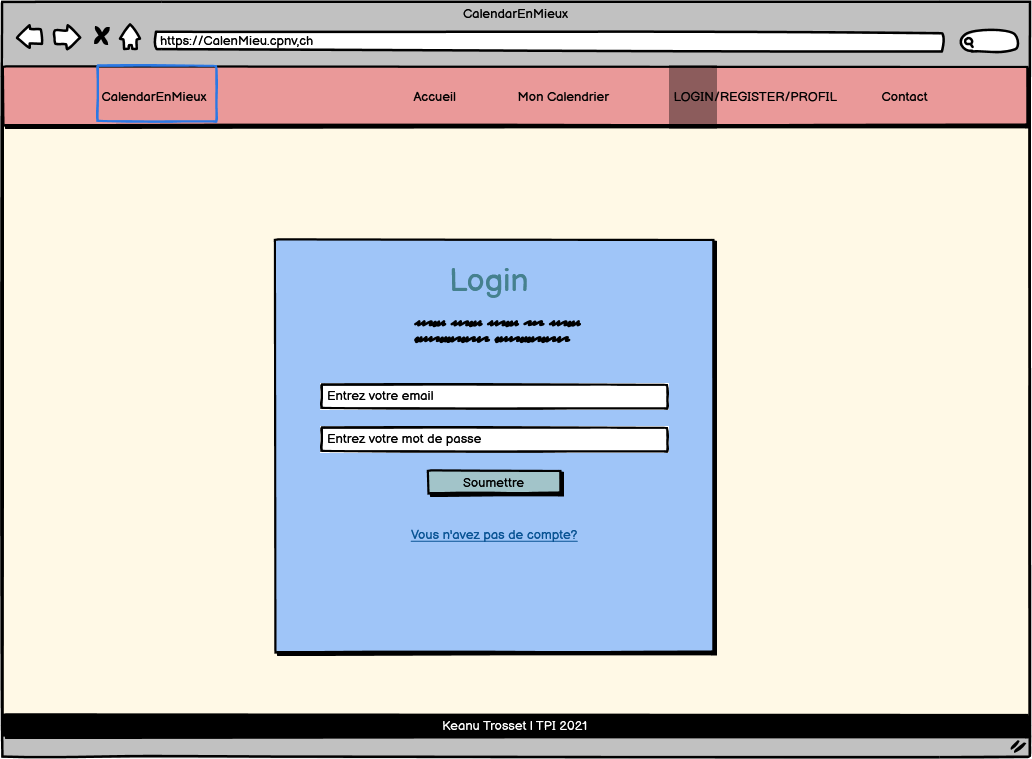
Le bouton Aujourd’hui est là pour revenir à la page qui montre le jour d’aujourd’hui.

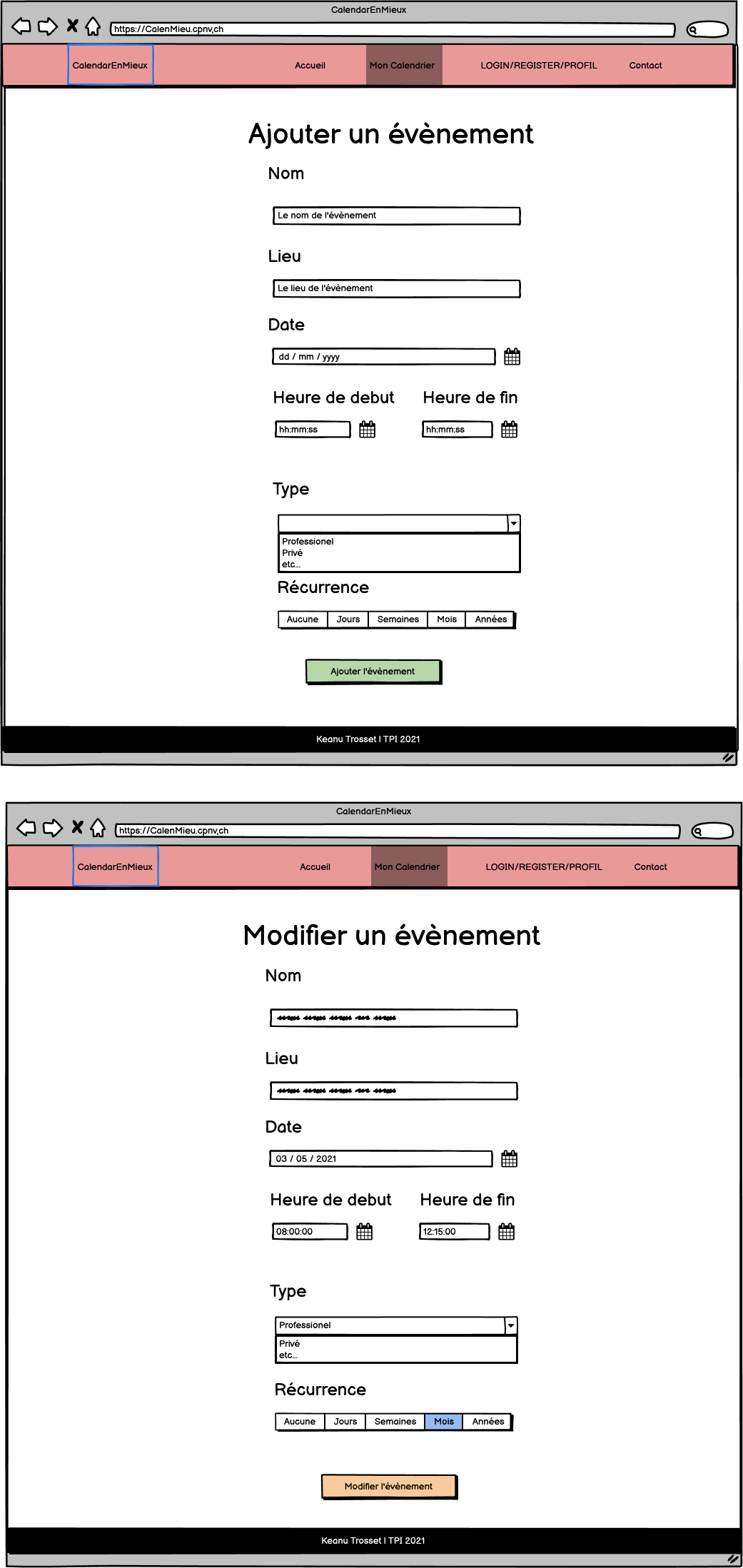
En premier, nous voyons le calendrier avec vue en mois.

Tandis qu’en second, nous voyons le calendrier avec la vue semaines.

Voici la page profil. Elle est ici un peu optionnellement puisque ce n’est pas demander dans le cahier des charges. La plus grande utilité est surtout au niveau du bouton « Se deconnecter ». Le bouton du profil est vu à chaque maquette mais sera bien évidemment visible qu’un fois connecté. Il remplacera d’ailleurs le « Login / Register ».

**

**Voici les pages Register et Login. Rien de très différents à un autre système d’enregistrement ou de connexion.

**Ici se trouve les pages pour ajouter ou modifier un évènement.

Il faut ici inscrire les informations utiles à la création d’un évènement.

## Conception de la Base de données

*Modèle MLD (Modèle relationnel) de la base de données.*

*Description précise des données (type, dimensions, contraintes..).*

## Conception du Code

*Diagrammes UML d’activité, diagramme d’état, diagramme de flux, ou pseudocode pour chaque cas d’utilisation.*

## Plugins et librairies

*Liste des composants additionnels utilisés (par exemple forum, classes, images, modules, etc).*

## Choix de la formule d’hébergement

Comme déjà dis plus haut, j’ai choisi d’héberger sur swisscenter.

Étant donné que contrairement à certain camarade je n’ai pas de nom de domaine, c’était la solution gratuite la plus simple. L’application Filezilla une fois configuré est très simple a utilisé et ne fais pas perdre beaucoup de temps pour un simple envoie de fichier sur le serveur.

De plus, la gestion de la base de donnée avec PHPMyAdmin est très agréable.

# Réalisation

## *Dossier de réalisation*

*Décrire la réalisation "physique" de votre projet*

* *les répertoires où le logiciel est installé*
* *la liste de tous les fichiers et une rapide description de leur contenu (des noms qui parlent !)*
* *les versions des systèmes d'exploitation et des outils logiciels*
* *la description exacte du matériel*
* *le numéro de version de votre produit.*
* *programmation et scripts: librairies externes, reconstruction du logiciel cible à partir des sources.*

*NOTE : Inutile d’inclure les listings des sources à moins que vous ne désiriez en expliquer une partie vous paraissant importante. Dans ce cas n’incluez que cette partie…*

## *Description des tests effectués*

*Pour chaque partie testée de votre projet, il faut décrire:*

* *les conditions exactes de chaque test*
* *les preuves de test (papier ou fichier)*
* *tests sans preuve: fournir au moins une description*

## *Erreurs restantes*

*S'il reste encore des erreurs:*

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

## *Dossier d'archivage*

*Décrire de manière détaillée les archives du projet.*

***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:*

* *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.php.net/>

<https://www.w3schools.com/>

<https://stackoverflow.com/>

Jérôme Jaquemet, camarade de classe.

## *Journal de bord de chaque participant*

|  |  |  |  |
| --- | --- | --- | --- |
| Date | **Durée** | **Activité** | **Remarques** |
|  |  |  |  |

## *Manuel d'Installation*

## *Manuel d'Utilisation*

## *Archives du projet*

*Une archive format .zip contenant toute la documentation / maquettes / code*

*Support (CD / Dropbox) à définir avec le responsable de projet et les experts.*