|  |
| --- |
| ÉquiPlanner |

Table des matières

[1 Analyse préliminaire 4](#_Toc197950309)

[1.1 Introduction 4](#_Toc197950310)

[1.2 Contexte 4](#_Toc197950311)

[1.3 Objectifs 4](#_Toc197950312)

[1.3.1 Points techniques évalués spécifique au projet 4](#_Toc197950313)

[1.5 Planification initiale 6](#_Toc197950314)

[2 Analyse / Conception 8](#_Toc197950315)

[2.1 Méthodologie de travail 8](#_Toc197950316)

[2.2 Stratégie de sauvegarde 8](#_Toc197950317)

[2.3 Public visé 8](#_Toc197950318)

[2.4 Concept 9](#_Toc197950319)

[2.4.1 Maquette graphique 9](#_Toc197950320)

[2.4.2 Modèle conceptuel de données 17](#_Toc197950321)

[2.4.3 Modèle logique des données 19](#_Toc197950322)

[2.4.4 Modèle physique des données 21](#_Toc197950323)

[2.5 Stratégie de test 22](#_Toc197950324)

[2.6 Risques techniques 23](#_Toc197950325)

[2.7 Planification 23](#_Toc197950326)

[2.7.1 Dates et Horaires de travail 23](#_Toc197950327)

[2.7.2 Déroulement du travail 23](#_Toc197950328)

[2.7.3 Planification détaillée 25](#_Toc197950329)

[2.8 Dossier de conception 29](#_Toc197950330)

[2.8.1 Matériel à disposition 29](#_Toc197950331)

[3 Réalisation 29](#_Toc197950332)

[3.1 Dossier de réalisation 29](#_Toc197950333)

[3.1.1 Gestion de la base de données 29](#_Toc197950334)

[3.2 Description des tests effectués 31](#_Toc197950335)

[3.3 Erreurs restantes 31](#_Toc197950336)

[3.4 Liste des documents fournis 31](#_Toc197950337)

[4 Conclusions 31](#_Toc197950338)

[5 Annexes 33](#_Toc197950339)

[5.1 Table des illustrations 33](#_Toc197950340)

[5.2 Résumé du rapport du TPI / version succincte de la documentation 33](#_Toc197950341)

[5.3 Sources – Bibliographie 33](#_Toc197950342)

[5.4 Journal de travail 34](#_Toc197950343)

[5.5 Manuel d'Installation 35](#_Toc197950344)

[5.6 Manuel d'Utilisation 35](#_Toc197950345)

[5.7 Archives du projet 35](#_Toc197950346)

*NOTE L’INTENTION DES UTILISATEURS DE CE CANEVAS:  
Toutes les parties en italiques sont là pour aider à comprendre ce qu’il faut mettre dans cette partie du document. Elles n’ont donc aucune raison d’être dans le document final.*

*De plus, en fonction du type de projet, il est tout à fait possible que certains chapitres ou paragraphes n’aient aucun sens. Dans ce cas il est recommandé de les retirer du document pour éviter de l’alourdir inutilement.*

# Analyse préliminaire

## Introduction

Le but de ce projet et de réaliser une application web permettant aux membres d’un club de sport de réserver des créneaux pour les terrains ou les équipements sportifs.

## Contexte

Un club sportif souhaite mettre à disposition ses ressources (terrains ou équipement) aux membres sans pour autant disposer de solutions informatiques complexes ou coûteuses. Ce projet vise à développer une développer une application web simple, fonctionnelle et intuitive, permettant aux membres d’un club de réserver une ressource disponible, à une date et un horaire précis.

Les ressources seront prédéfinies dans la base de données. Aucun espace administrateur n’est nécessaire : les membres pourront uniquement consulter et réserver ce qui est déjà disponible.

## Objectifs

Voici la liste des livrables à la fin du TPI :

* Une planification initiale.
* Un rapport de projet contenant en autres un plan de tests manuels.
* Un journal de travail.
* Script(s) SQL (structure + utilisateurs + ressources).
* Code source complet disponible dans un repository GIT.
* README.md Clair et détaillé (explication du contenu de repository et liens vers toutes les ressources qui s’y trouvent.

Voici la listes des fonctionnalités à mettre en place durant ce projet :

* Connexion à l’application via des comptes prédéfinis.
* Consultation des ressources disponibles (terrains équipement).
* Réservation d’un créneau horaire pour une ressource.
* Affichage des réservations passées et à venir par l’utilisateur.
* Annulation d’une réservation encore à venir.

### Points techniques évalués spécifique 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 7 points spécifiques suivants (Point A14 à A20) :

* Mise en place d’une structure MVC : séparation claire entrer les modèles (accès BDD), les vues (affichage HTML) et les contrôleurs(logique métier).
* Connexion sécurisée à la base de données via PDO et utilisateur MySQL dédié (pas d’utilisation du compte *root*)
* Authentification fonctionnelle avec gestion de session et espace personnel accessible après connexion.
* Affichage dynamique des ressources disponibles : seules les ressources non réservées pour la date /heure choisies sont proposées.
* Réservation dynamique d’une ressource : enregistrement conditionnel à la disponibilité réelle lors de la soumission.
* Liste des réservations pour l’utilisateur connecté avec statut(confirmée/annulée)
* Annulation d’une réservation à venir par l’utilisateur connecté : mise à jour du statut de la réservation dans la base de données.

**Remarque :**

Le recours à des outils en ligne d’intelligence artificielle (ex. : Chat GPT) doit être mentionné et ne peut servir que d’inspiration à la réalisation. En cas d’abus, l’évaluation du TPI en tiendra compte.

*Ce chapitre énumère les objectifs du projet. L'atteinte ou non de ceux-ci devra pouvoir être contrôlée à la fin du projet. Les objectifs pourront éventuellement être revus après l'analyse.*

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

## Planification initiale

Voici la planification initiale du projet par semaine :

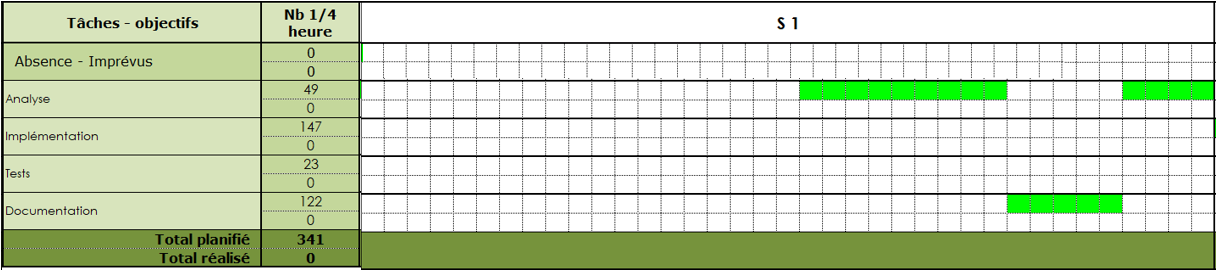


Figure 1 : planification initiale, semaine 1

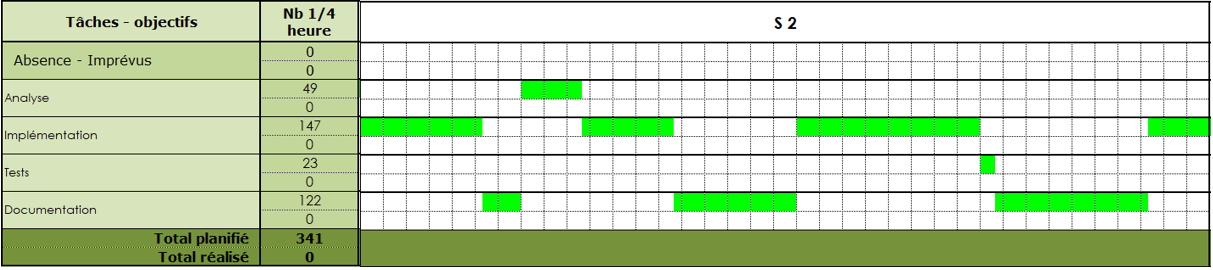


Figure 2 : planification initiale, semaine 2

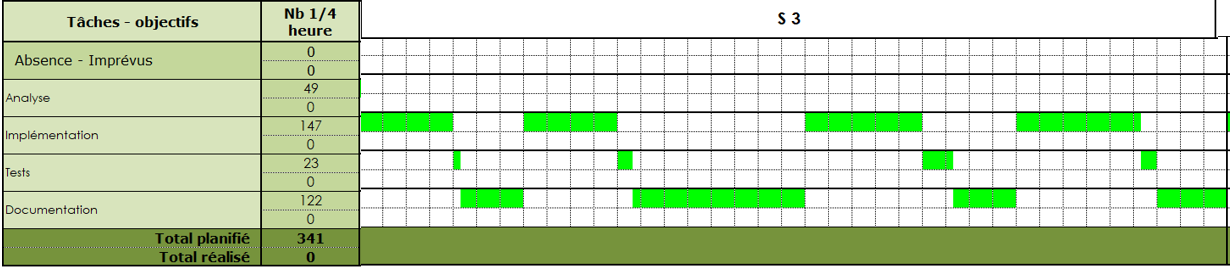


Figure 3 : planification initiale, semaine 3

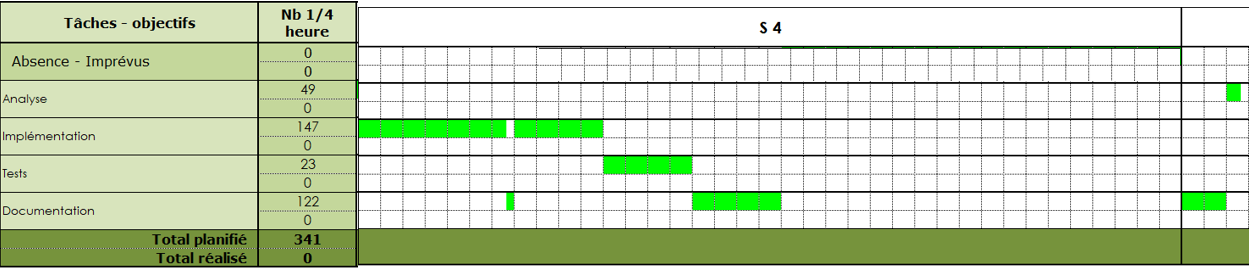


Figure 4 : planification initiale, semaine 4 et 5

# Analyse / Conception

## Méthodologie de travail

Pour ce TPI, je souhaite appliquer une méthodologie de travail en cascade. En effet, cette méthode me paraît être la plus adaptée pour le projet. Elle se décompose en 6 étapes :

1. **La définition des besoins.**

Il s’agit de recueillir les attentes et exigences du commanditaire et définit l’orientation du reste du projet.

1. **L’analyse des besoins**

Les exigences sont analysées pour en faire des objectifs concrets.

1. **La conception**

Permet de découper et de planifier les tâches pour le bon fonctionnement du projet.

1. **La mise en œuvre**

Réalisation des tâches de travail.

1. **La validation**

Le produit terminé est contrôlé, si des modifications sont nécessaires, elles sont effectuées puis le produit est validé.

1. **La mise en service**

Le produit est mis en service.

## Stratégie de sauvegarde

Afin de ne pas subir d’incident de perte de données, j’ai décidé de prévoir un enregistrement de mon travail sur le disque dur du post de travail avant chaque pause et d’en copier l’intégralité sur mon disque dur externe ainsi que de le mettre en ligne sur le Git à la fin de chaque demi-journée afin d’assurer des backups. De cette façon, je suis à l’abri de perdre plus de perdre plus d’une demi-journée de travail.

## Public visé

Le public cible pour ce projet sont les membres du club sportif souhaitant réserver des terrains et de l’équipements.

## Concept

### Maquette graphique

Pour la conception de ma maquette graphique, j’ai décidé d’utiliser le logiciel Pencil Project que j’ai eu l’occasion à maintes reprises d’utiliser durant mes projets ou cours à l’ETML.

La maquette se décompose en 7 pages :

1. Page de connexion
2. Page d’espace personnel
3. Page du choix de ressource
4. Page de réservation
5. Page de validation
6. Page d’historique des réservations
7. Page d’annulation de réservations

Voici la page de connexion :

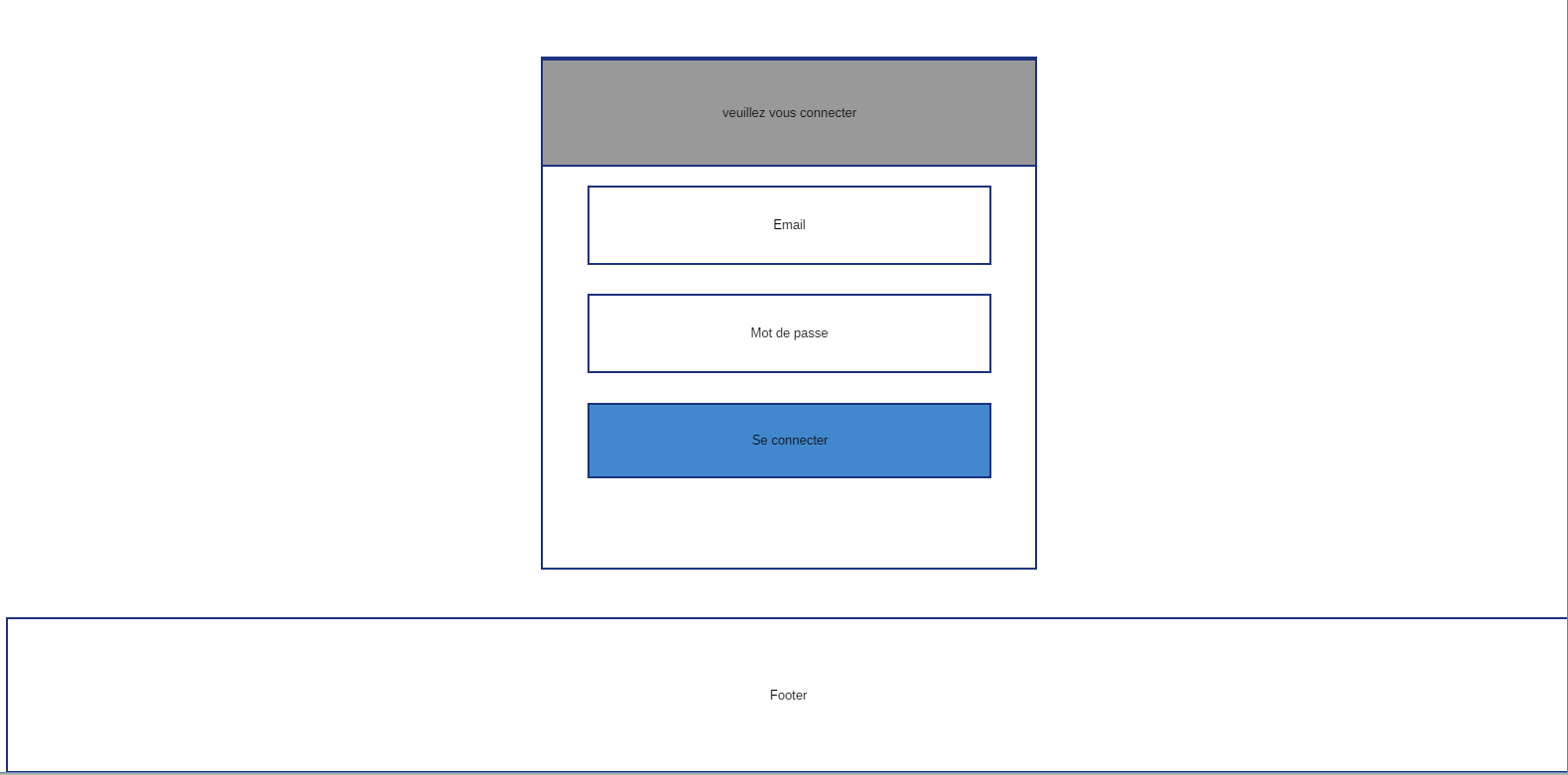


Figure 5 : page de connexion

Voici la page d’espace personnel :

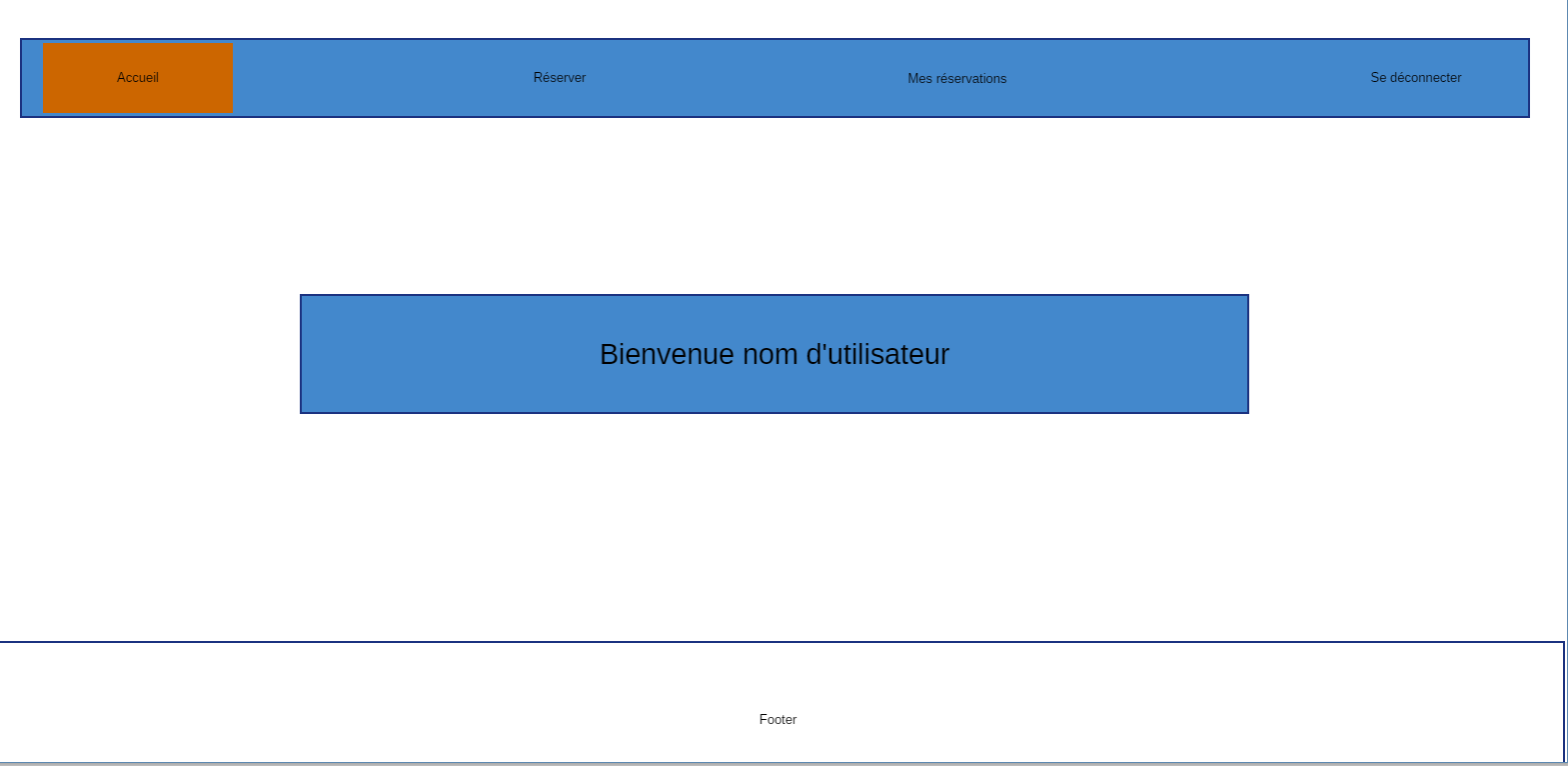


Figure 6 : page d'espace personnel

Page du choix des ressources :

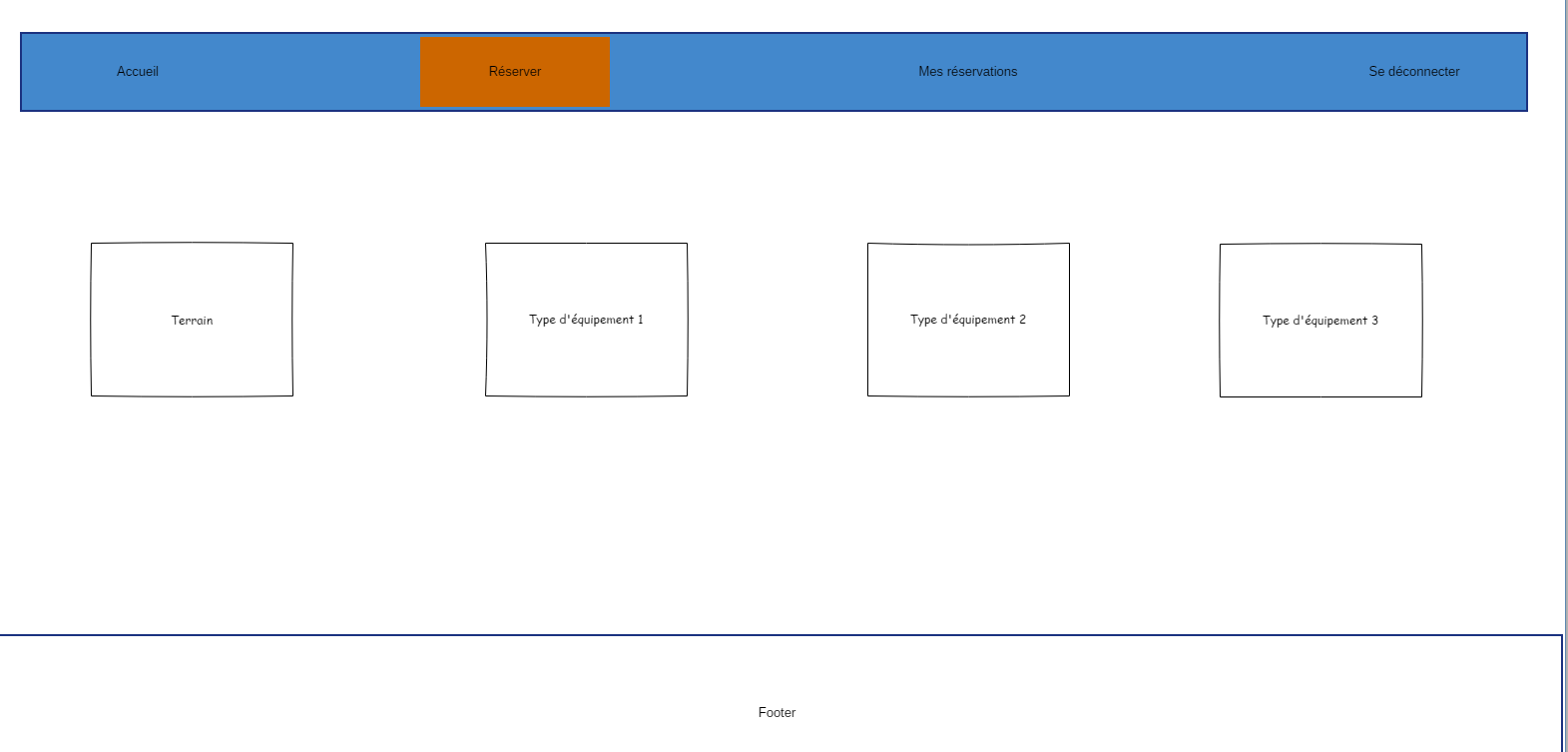


Figure 7 : page du choix des ressources

Page de réservation :

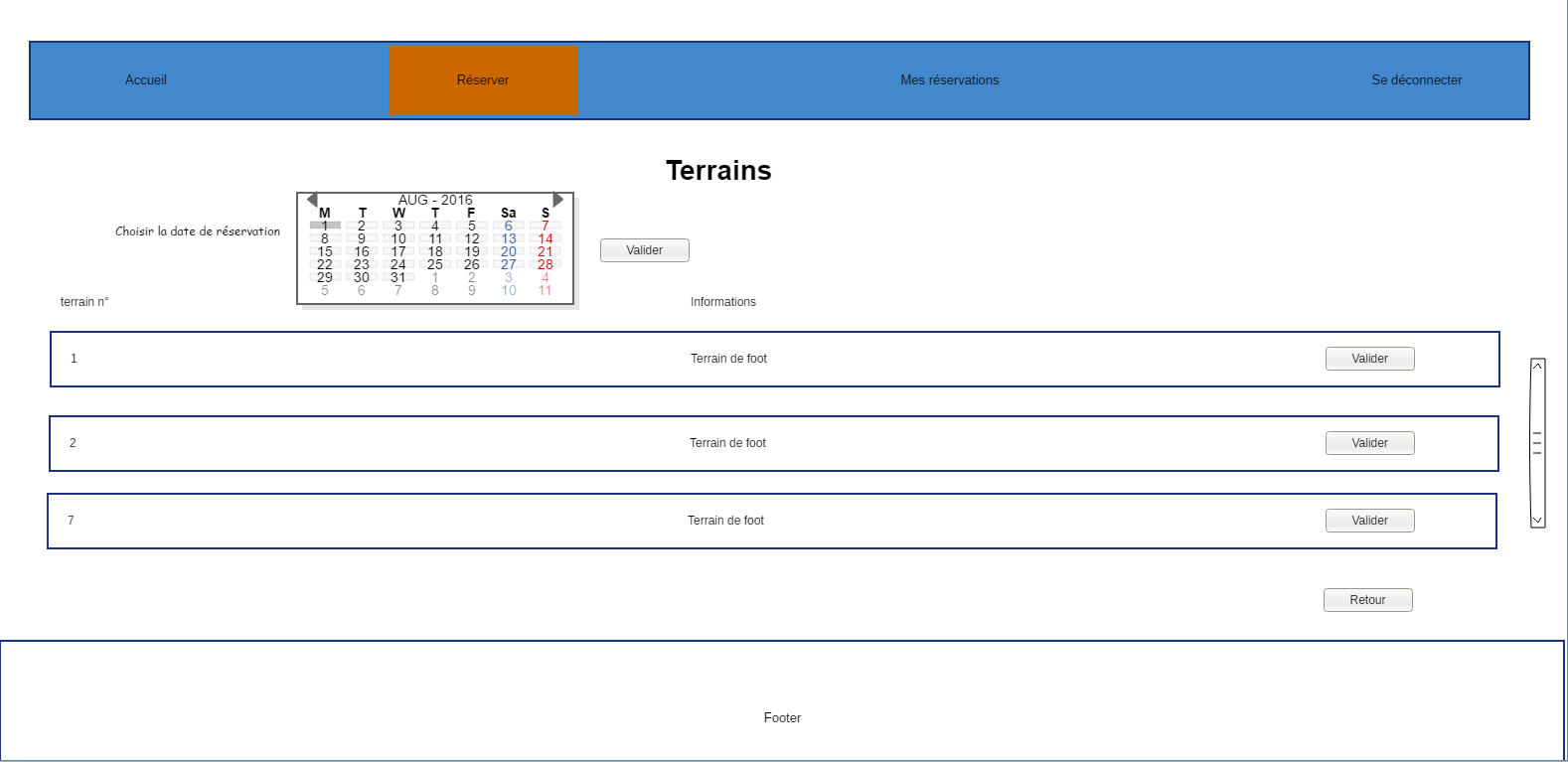


Figure 8 : Page de réservation

Page de validation :

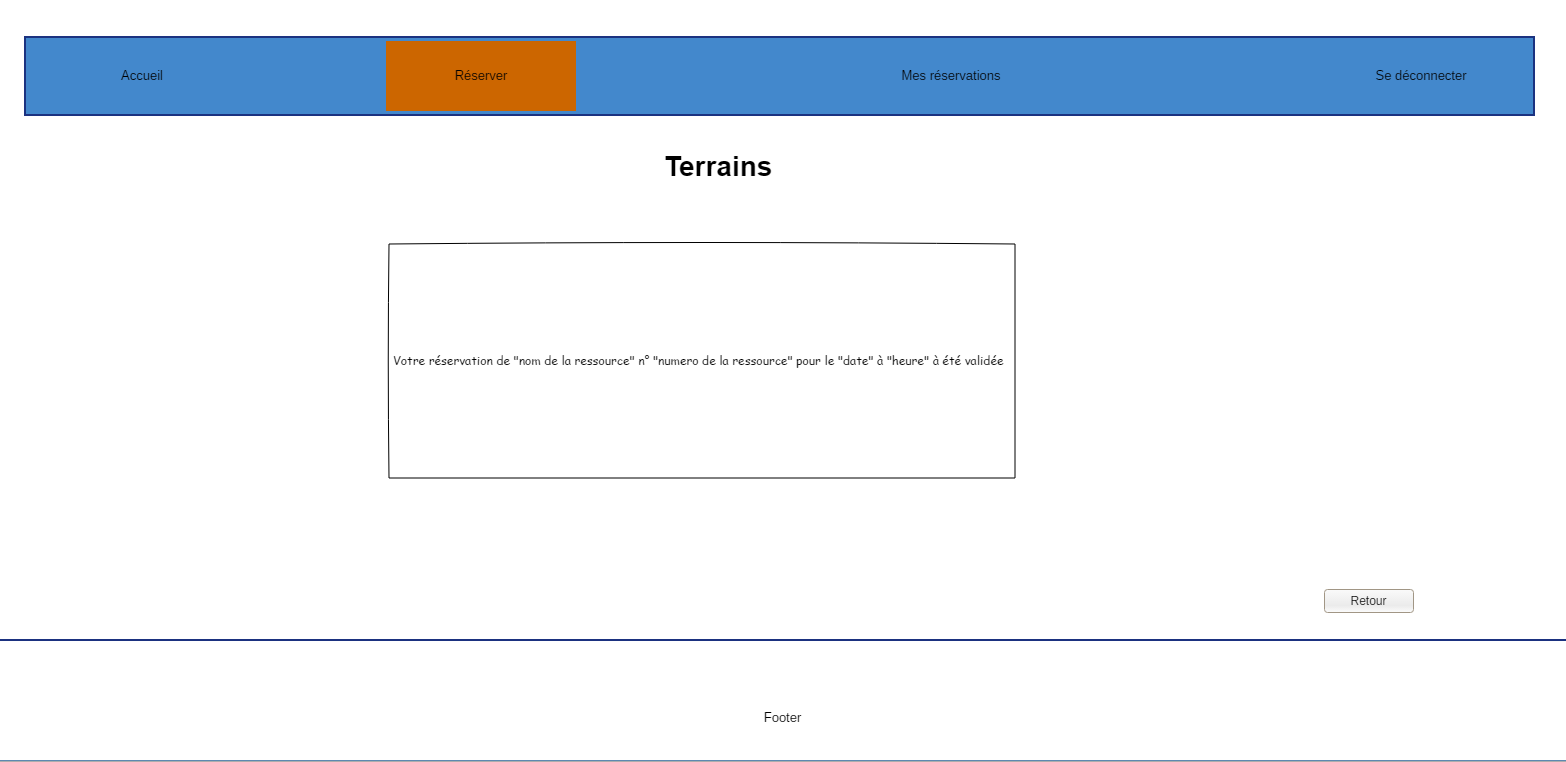


Figure 9 : page de valuidation

Page d’historique des réservations :



Figure 10 : page d'historique des réservations

Page d’annulation des réservations :



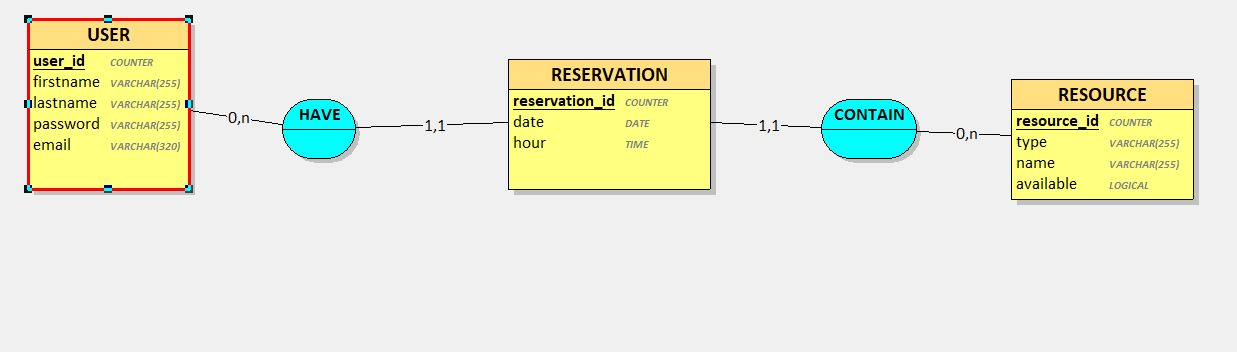
Figure 11 : page d'annulation des réservations

### Modèle conceptuel de données

Pour la création du modèle conceptuel de données, j’ai choisi le logiciel looping avec lequel j’ai déjà eu l’occasion de pratiquer. Looping est un logiciel libre d’utilisation qui permet la conception de MCD, MLD et permet la génération de script SQL é partir de celui-ci.

Voici le MCD réalisé sur Looping :

Figure 12 : MCD



#### **Les tables**

* **La table USER**

Cette table contient les données des membres du club de sport.

**user\_id** est l’identifiant de la table. Sont type est un **Counter.** Il est **NOT NULL** car chaque membre doit posséder un identifiant.

**firstname** est l’attribut qui comporte le prénom du membre. C’est un **VARCHAR** d’une longueur de **255,** car cela est suffisant pour un prénom.Il est **NOT NULL.**

**lastname** est l’attribut qui comporte le nom du membre. C’est un **VARCHAR** d’une longueur de **255,** car cela est suffisant pour un nom.Il est **NOT NULL.**

**password** est l’attribut qui comporte le mot de passe du membre. C’est un **VARCHAR** d’une longueur de **255,** car une fois le mot de passe hashé, il fera toujours la même longueur de 60 caractères. Cependant si dans le futur l’algorithme change il se peut que le nombre de caractères soit plus grand. Il est **NOT NULL.**

**email** est l’attribut qui comporte l’adresse mail du membre. C’est un **VARCHAR** d’une longueur de **320** car la longueur maximale que peut atteindre une adresse mail est de 320 caractères.Il est **NOT NULL.**

* **La table Reservation**

Cette table contient les données des réservations.

**reservation\_id** est l’identifiant de la table. Sont type est un **Counter.** Il est **NOT NULL** car chaque réservation doit posséder un identifiant.

**date** est l’attribut qui comporte la date de la réservation. C’est un **DATE.** Il est **NOT NULL.**

**hour** est l’attribut qui comporte l’heure de la réservation. C’est un **TIME.** Il est **NOT NULL.**

* **La table Resource**

Cette table contient les données des ressources du club de sport.

**type** est l’attribut qui comporte le type de ressource. C’est un **VARCHAR** d’une longueur de **255.** Il est **NOT NULL.**

**name** est l’attribut qui comporte le nom de la ressource. C’est un **VARCHAR** d’une longueur de **255.** Il est **NOT NULL.**

**available** est l’attribut qui définit si la ressource est disponible ou pas. C’est un **booléen.** Il est **NOT NULL.**

#### **Les associations**

L’association **HAVE**:

Un membre peut avoir 0 à N réservation alors qu’une réservation ne peut être associé à un seul membre.

L’association **CONTAIN**:

Une réservation ne peut contenir qu’une seule ressource tandis qu’a l’inverse, une ressource peut être dans 0 à N réservations.

### Modèle logique des données

Le modèle logique des données à été réalisé avec looping. Les clés étrangères ont été renommées pour être conforme aux conventions de codage de l’ETML.

Voici le MLD :

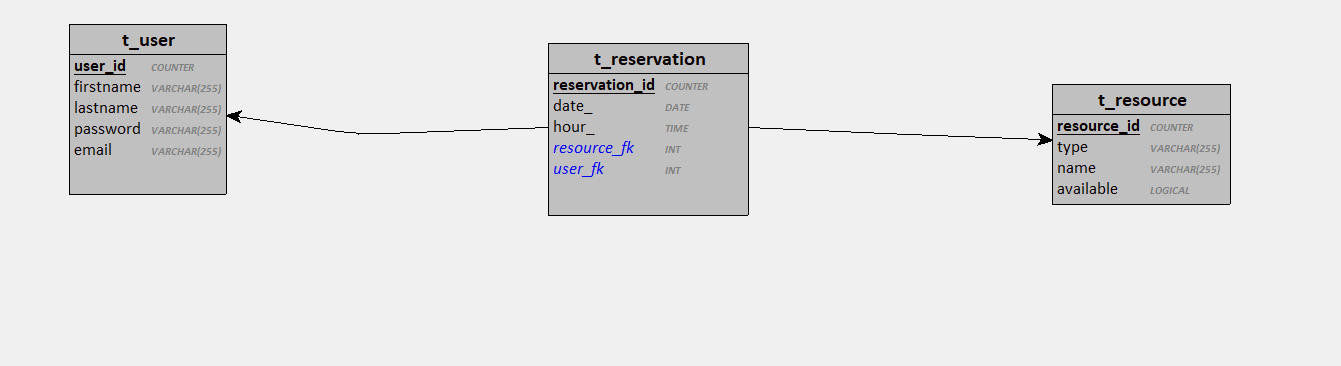


Figure 13 : MLD

### Modèle physique des données

Voici le modèle physique des données (MPD) :

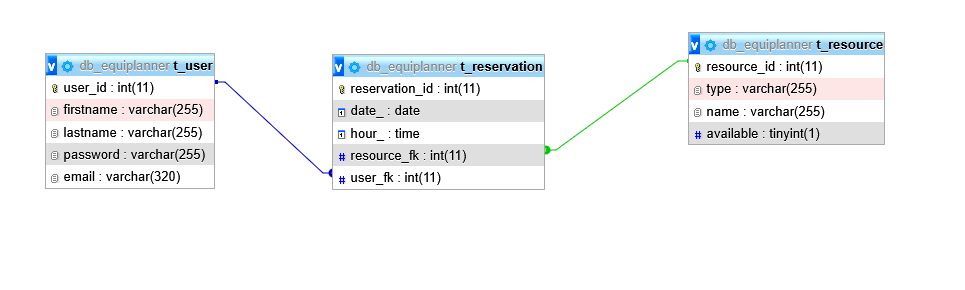


Figure 14 : MPD

*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

Pour tester les fonctionnalités de mon projet, j’ai décidé de créer des cas avec scénario. Chaque étapes des scénarios seront spécifiées et les résultats attendus seront définis à l’aide d’un tableau. Avec cette méthode, je peux clairement et aisément définir la réussite ou l’échec d’un test.

En cas d’échec, les modifications sont apportées et le test est réitéré

Voici un exemple de test :

|  |  |  |
| --- | --- | --- |
| Nom du test : | Exemple |  |
| Identifiant : | Exemple-01 |  |
| Objectif : | Se connecter à son compte |  |
| Prérequis : | - |  |
| Cas de test : | | |
| Etape | Action | Résultat attendu |
| 1 | Cliquer sur le bouton connexion | La page de connexion s’ouvre demandant l’adresse mail et le mot de passe |
| 2 | Entrer ses informations | L’adresse mail est affichée tandis que le mot de passe et caché |
| 3 | Cliquer sur le bouton se connecter | La page home du membre s’ouvre |

*Décrire la stratégie globale de test:*

* *types de des tests et ordre dans lequel ils seront effectués.*
* *les moyens à mettre en œuvre.*
* *couverture des tests (tests exhaustifs ou non, si non, pourquoi ?).*
* *données de test à prévoir (données réelles ?).*
* *les testeurs extérieurs éventuels.*

## Risques techniques

Le risque principal que je perçois serait de négliger la documentation au profit de la réalisation. Je devrais donc me tenir au maximum à ma planification initiale.

* *Risques techniques (complexité, manque de compétences, …).*

*Décrire aussi quelles solutions ont été appliquées pour réduire les risques (priorités, formation, actions, …).*

## Planification

### Dates et Horaires de travail

Le TPI débute le jeudi 8 mai 2025 à 08h00 et se termine le lundi 2 juin 2025 à 10h00.

Les horaires de travail sont :

### Déroulement du travail

Le projet se déroule sur un total de 90 heure réparti en 20% d’analyse, 40% d’implémentation, 10% de tests et 30% de documentation.

Lors du premier jour du TPI, une rencontre avec l’expert n°1 et le chef de projet s’effectue pour la prise de connaissance du cahier des charges et la validation de celui-ci.

Au milieu du projet, l’expert n°2 rend une visite au candidat afin de contrôler le travail effectué jusqu’à présent et de répondre aux éventuelles questions du candidat.

Une fois arrivé au terme du projet, le candidat remet son travail aux experts et au chef de projet.

### Planification détaillée

Pour la réalisation de ma planification détaillée lors du premier jour, j’ai choisi d’utiliser le model Excel de l’ETML. Celui-ci permet de découper les taches en quart d’heure ce qui permet d’avoir une grande maniabilité de planification.

Voici planification détaillée :

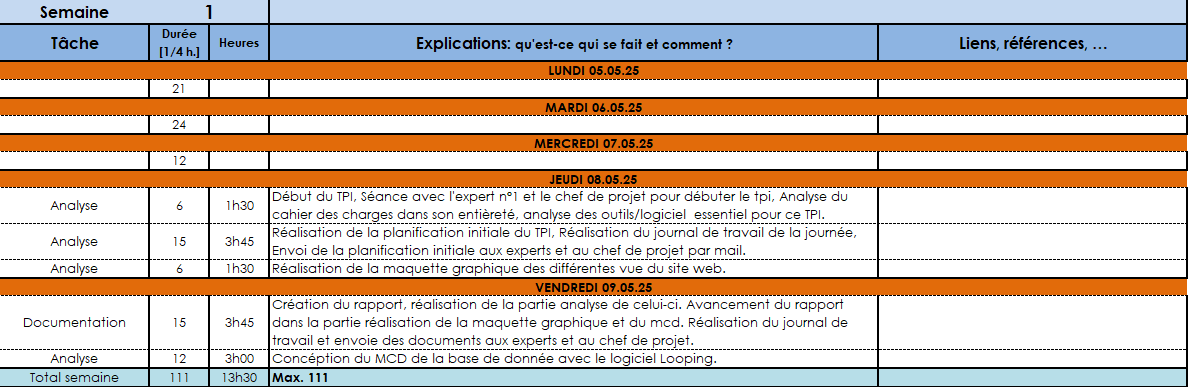


Figure 15 : planification détaillée, semaine 1

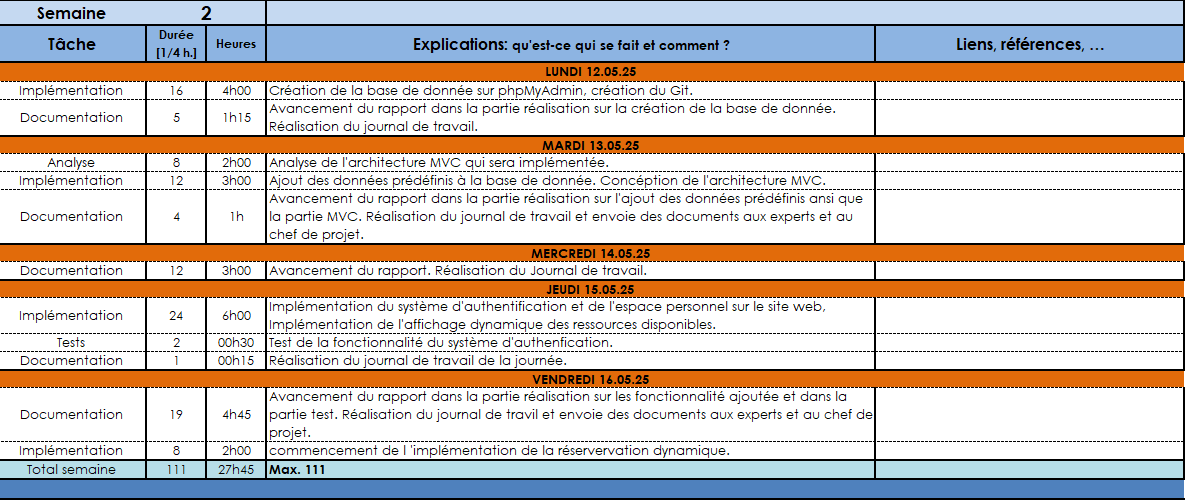


Figure 16 : planification détaillée, semaine 2

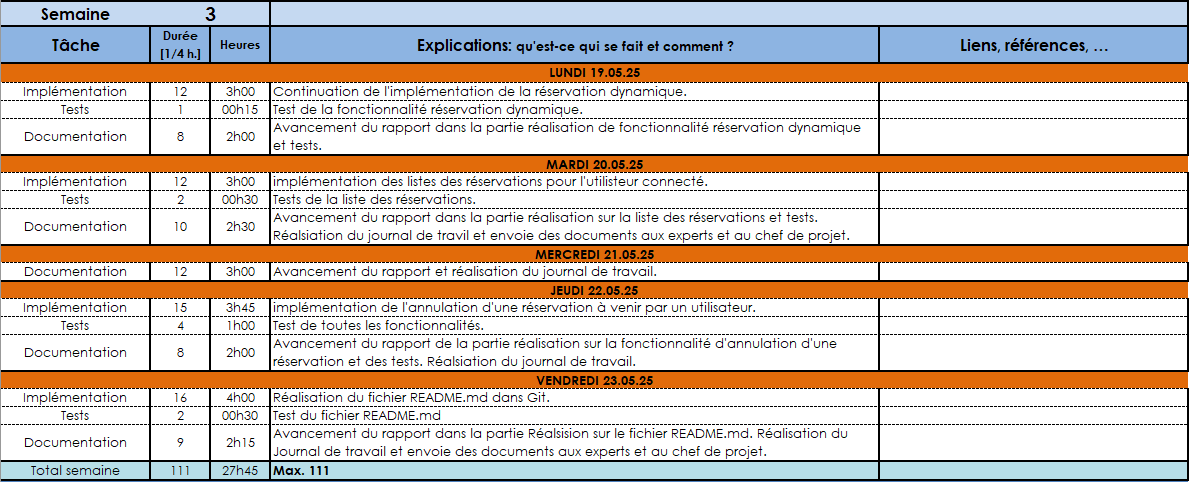


Figure 17 : planification détaillée, semaine 3

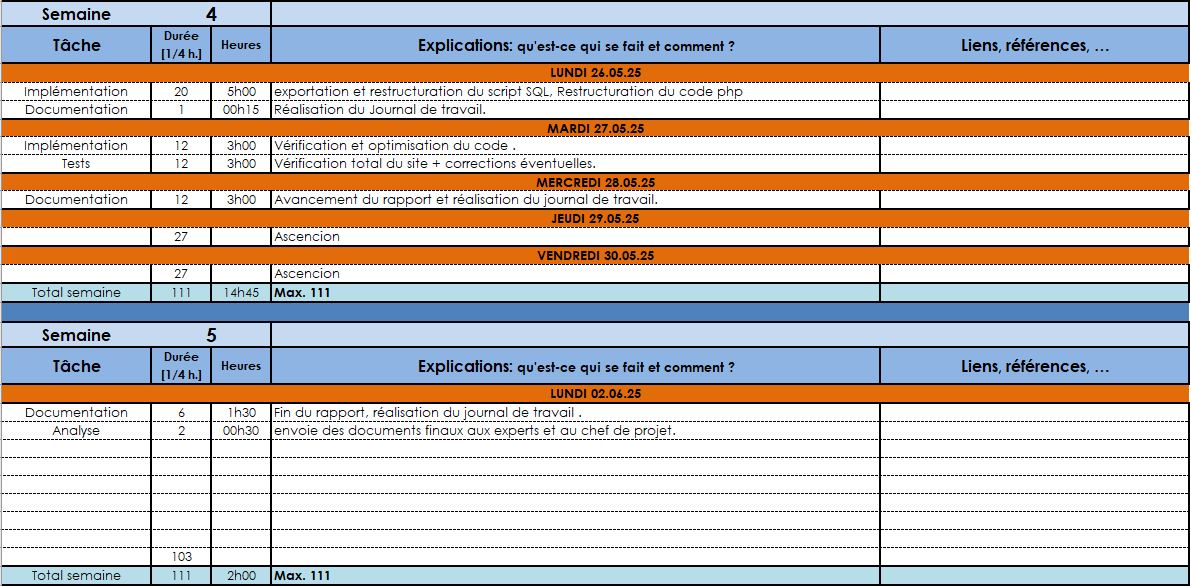


Figure 18 : planification détaillée, semaine 4 et 5

## 

## Dossier de conception

### Matériel à disposition

* 1 poste de travail tournant sur Windows 10 (configuration standard ETML)
* Suite Office 365
* Logiciel Looping
* Logiciel Visual Studio Code
* Logiciel Pencil
* Serveur local uWAMP(apache, PHP,MySQL)
* GIT+platforme GitHub

*Fournir tous les document de conception:*

* *le choix du matériel HW*
* *le choix des systèmes d'exploitation pour la réalisation et l'utilisation*
* *le choix des outils logiciels pour la réalisation et l'utilisation*
* *site web: réaliser les maquettes avec un logiciel, décrire toutes les animations sur papier, définir les mots-clés, choisir une formule d'hébergement, définir la méthode de mise à jour, …*
* *bases de données: décrire le modèle relationnel, le contenu détaillé des tables (caractéristiques de chaque champs) et les requêtes.*
* *programmation et scripts: organigramme, architecture du programme, découpage modulaire, entrées-sorties des modules, pseudo-code / structogramme…*

***Le dossier de conception devrait permettre de sous-traiter la réalisation du projet !***

# Réalisation

## Dossier de réalisation

### Gestion de la base de données

#### UWamp

**UWamp** est la vérsion portable de **Wamp.**

UWamp permet de créer un environnement local qui me permet de développer mon application web dans un environnement controlé UWamp utilise *MYSQL* pour gérer la base de données, le serveur *Apache* pour gérer les requêtes web et *PHP* pour les scripts.

Cela permet de réaliser des sites web dynamiques et de les tester en local en toute sécurité.

Pour la création de la base de données que j’ai nommée *db\_EquiPlanner*, j’ai utilisé le code SQL proposé par looping suite à la création du MCD/MLD. Je l’ai directement inséré dans l’onglet SQL de *phpMyAdmin*. J’ai dû modifier le type de donnée pour les identifiants pour indiquer que les ***COUNTER*** étaient des ***INT AUTO\_INCEMENT*** pour chaque table ainsi que dans la table ***RESOURCE*** l’attribut ***AVAILABLE*** *a été spécifié que c’était un* ***BOOLEAN.*** Une fois ces étapes réalisées, ma base de données était créée avec l’intégralité de ses tables.

#### Créations des membres du club

Une fois la base de données créée, j’ai pu créer les utilisateurs. Pour cette étape, je me suis rendu dans la table *USER* sous l’onglet insérer. Depuis cet onglet j’ai pu rentrer les informations spécifiques aux membres qui sont :

* Le prénom
* Le nom
* L’adresse mail
* Le mot de passe

Inutile de rentrer leur identifiant étant donné que l’attribut **user\_id** est en **AUTO\_INCREMENT**.

Le pattern pour le mot de passe et l’adresse mail des utilisateurs est le suivant :

Pour le mot de passe :

Leurs initiales en majuscule suivi de 12345

Pour l’adresse mail :

Leur prénom commençant avec une majuscule . leur nom commençant par une majuscule @ equi.com.

Exemple pour l’utilisateur John Doe :

Adresse mail = [John.Doe@equi.com](mailto:John.Doe@equi.com)

Mot de passe = JD12345

#### Ajout des ressources

Pour l’ajout des ressources, je me suis rendu dans la table *RESOURCE* depuis laquelle j’ai pu définir le **type** des ressources (ex. terrain, ballon) le **nom** des ressources et leur **disponibilité**(oui / non).

*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, dictionnaire des données, reconstruction du logiciel - cible à partir des sources.*

*NOTE : Evitez 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*

## 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

## Table des illustrations

[Figure 1 : planification initiale, semaine 1 5](#_Toc197939142)

[Figure 2 : planification initiale, semaine 2 5](#_Toc197939143)

[Figure 3 : planification initiale, semaine 3 6](#_Toc197939144)

[Figure 4 : planification initiale, semaine 4 et 5 6](#_Toc197939145)

[Figure 5 : page de connexion 9](file:///D:\TPI\Document\Rapport\mbayogr-rapport.docx#_Toc197939146)

[Figure 6 : page d'espace personnel 10](#_Toc197939147)

[Figure 7 : page du choix des ressources 11](file:///D:\TPI\Document\Rapport\mbayogr-rapport.docx#_Toc197939148)

[Figure 8 : Page de réservation 12](file:///D:\TPI\Document\Rapport\mbayogr-rapport.docx#_Toc197939149)

[Figure 9 : page de valuidation 13](#_Toc197939150)

[Figure 10 : page d'historique des réservations 14](file:///D:\TPI\Document\Rapport\mbayogr-rapport.docx#_Toc197939151)

[Figure 11 : page d'annulation des réservations 15](file:///D:\TPI\Document\Rapport\mbayogr-rapport.docx#_Toc197939152)

[Figure 12 : MCD 16](file:///D:\TPI\Document\Rapport\mbayogr-rapport.docx#_Toc197939153)

[Figure 13 : MLD 19](file:///D:\TPI\Document\Rapport\mbayogr-rapport.docx#_Toc197939154)

[Figure 14 : MPD 20](#_Toc197939155)

[Figure 15 : planification détaillée, semaine 1 24](file:///D:\TPI\Document\Rapport\mbayogr-rapport.docx#_Toc197939156)

[Figure 16 : planification détaillée, semaine 2 25](#_Toc197939157)

[Figure 17 : planification détaillée, semaine 3 26](file:///D:\TPI\Document\Rapport\mbayogr-rapport.docx#_Toc197939158)

[Figure 18 : planification détaillée, semaine 4 et 5 27](#_Toc197939159)

[Figure 19: journal de travail, semaine 1 31](file:///D:\TPI\Document\Rapport\mbayogr-rapport.docx#_Toc197939160)

## Résumé du rapport du TPI / version succincte de la documentation

## Sources – Bibliographie

*Liste des livres utilisés (Titre, auteur, date), des sites Internet (URL) consultés, des articles (Revue, date, titre, auteur)… Et de toutes les aides externes (noms)*

## 

## Journal de travail

Voici le journal de travail de la semaine 1 :

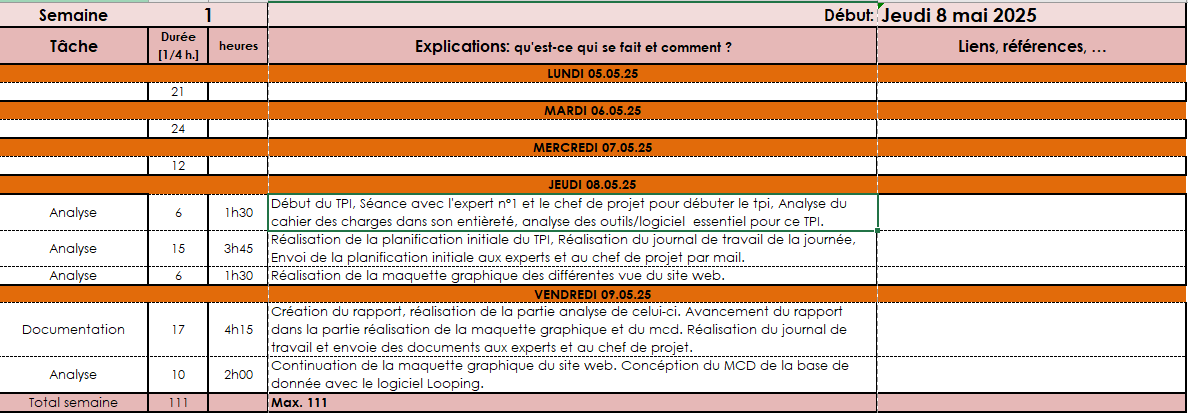


Figure 19: journal de travail, semaine 1

## 

## Manuel d'Installation

## Manuel d'Utilisation

## Archives du projet

*Media, … dans une fourre en plastique*