31/05/2022

TPI KeepItFit

**L’application fit faite pour vous**

****

Table des matières

[1 Analyse préliminaire 4](#_Toc104883671)

[1.1 Introduction 4](#_Toc104883672)

[1.2 Objectifs 4](#_Toc104883673)

[1.3 Planification initiale 6](#_Toc104883674)

[2 Analyse / Conception 6](#_Toc104883675)

[2.1 Concept 6](#_Toc104883676)

[2.1.1 MCD 7](#_Toc104883677)

[2.1.2 Maquettes 8](#_Toc104883678)

[2.2 Use cases 12](#_Toc104883679)

[2.3 Stratégie de test 13](#_Toc104883680)

[2.4 Risques techniques 13](#_Toc104883681)

[2.5 Planification 13](#_Toc104883682)

[2.6 Dossier de conception 14](#_Toc104883683)

[2.6.1 Choix du matériel 14](#_Toc104883684)

[2.6.2 Schéma de navigation 15](#_Toc104883685)

[2.6.3 MLD 16](#_Toc104883686)

[3 Réalisation 16](#_Toc104883687)

[3.1 Dossier de réalisation 16](#_Toc104883688)

[3.1.1 Architecture du code 17](#_Toc104883689)

[3.1.2 Script / librairies externe 18](#_Toc104883690)

[3.1.3 Points techniques 18](#_Toc104883691)

[3.2 Description des tests effectués 18](#_Toc104883692)

[3.3 Erreurs restantes 19](#_Toc104883693)

[3.4 Liste des documents fournis 19](#_Toc104883694)

[4 Conclusions 19](#_Toc104883695)

[4.1 Objectif atteints / non atteints 19](#_Toc104883696)

[4.2 Points positifs / négatif 20](#_Toc104883697)

[4.3 Difficultés particulières 20](#_Toc104883698)

[4.4 Suite possible pour le projet 20](#_Toc104883699)

[5 Annexes 22](#_Toc104883700)

[5.1 Résumé du rapport du TPI / version succincte de la documentation 22](#_Toc104883701)

[5.2 Sources – Bibliographie 22](#_Toc104883702)

[5.3 Manuel d'Installation 22](#_Toc104883703)

[5.4 Manuel d'Utilisation 22](#_Toc104883704)

[5.5 Archives du projet 22](#_Toc104883705)

# Analyse préliminaire

## Introduction

Dans le cadre de mon projet TPI je dois développer un site web axé sur le fitness.

Le but est qu’un utilisateur puisse créer un programme en fonction de ses envies et de son état physique (âge, poids, taille).

Cet utilisateur pourra donc avoir son programme personnaliser et exporter son programme en PDF pour pouvoir ensuite l’imprimer.

Un utilisateur admin sera charger de créer les differents exercices ainsi que les lieux matériels, zones ciblées et

## Objectifs

Le site web proposé permettra de proposer des programmes d’entraînements de sport en fonction d’objectifs définis par l’utilisateur.

Le site doit couvrir les cas suivants :

* Partie « public »
  + « Accueil » : une page d’accueil présentera le site.
* Un utilisateur pourra s’enregistrer sur le site. Un email lui sera envoyé pour confirmer sa demande de connexion.
* Partie « utilisateur connecté »
* Un utilisateur connecté pourra choisir son programme de sport (par exemple se muscler, perdre du poids…) puis donner des informations sur lui-même (son âge, son poids, sa taille…) puis le lieu où il s’entraînera (maison, salle de sport, extérieur…).
* Un programme adapté à sa demande lui sera proposé avec une description, les zones ciblées du corps, le matériel nécessaire, la difficulté, une durée, un nombre d’exercices et un planning avec les différents exercices. Le candidat sera libre d’ajouter des informations supplémentaires.
* En cliquant sur chacun des exercices du planning, un descriptif apparaîtra.
* Un PDF pourra être généré présentant le programme et le planning proposés. Chaque activité du planning proposée aura une description succincte visible sur le PDF.
* Partie « administrateur »
* Il pourra rentrer les programmes avec les informations nécessaires à l’utilisateur qui souhaite en trouver un adapté à son objectif.
* Il pourra aussi entrer les exercices proposés dans les programmes.

Toutes les informations persistantes seront stockées dans une base de données. Celle-ci sera laissée au choix du candidat.

Le site sera responsive, il devra permettre une lecture et une navigation conviviales et optimales, qu’il soit consulté depuis un téléphone mobile, une tablette ou un ordinateur.

Le candidat sera libre de choisir sa méthode de gestion de projet. Il devra justifier son choix.

## Planification initiale

Pour la planification j’ai utilisé la méthode Agile.

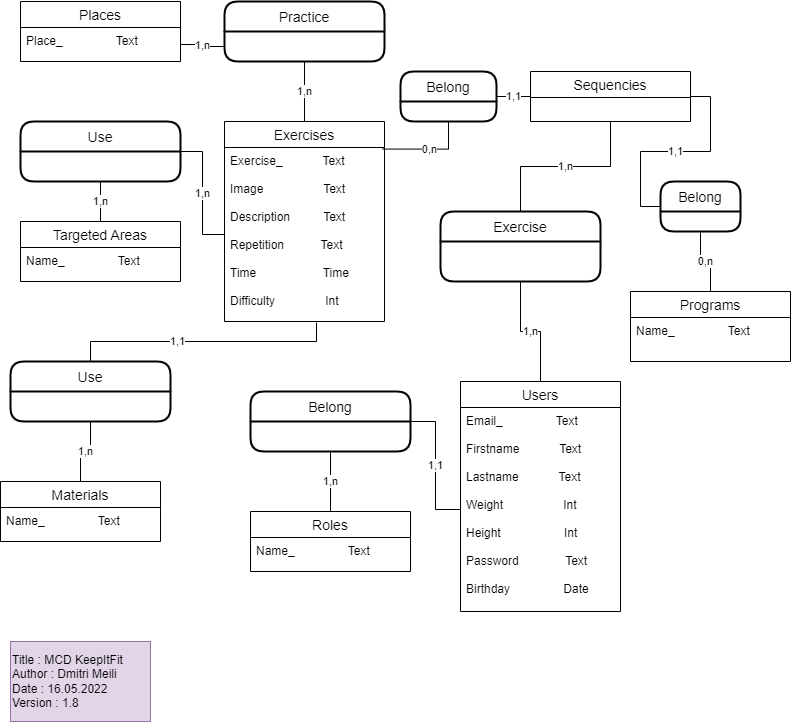
|  |  |  |
| --- | --- | --- |
| Sprints | Date | Objectif |
| Sprint 1 | 02.05  06.05 | * Définition du projet * Conception maquettes * Conception MCD * Conception des User Stories * Documentation Analyse |
| Sprint 2 | 08.05  13.05 | * Conception MLD * Réalisation de la partie « Accueil » du site & Test unitaire & acceptation * Élaboration de la BDD * Réalisation de la partie « Login » du site & Test unitaire & acceptation * Réalisation de la partie « Création de programme » du site & Test unitaire & acceptation * Documentation Réalisation |
| Sprint 3 | 16.05  20.05 | * Réalisation de la partie « Admin » du site & Test unitaire & acceptation * Design de l’interface du site * Documentation Réalisation |
| Sprint 4 | 23.05  31.05 | * Correction des bugs * Déployer le site sur Swisscenter & Test de performance * Finalisation de la Documentation & manuels d’utilisation/d’installation * Livraison |
| Fin de projet | | |

# Analyse / Conception

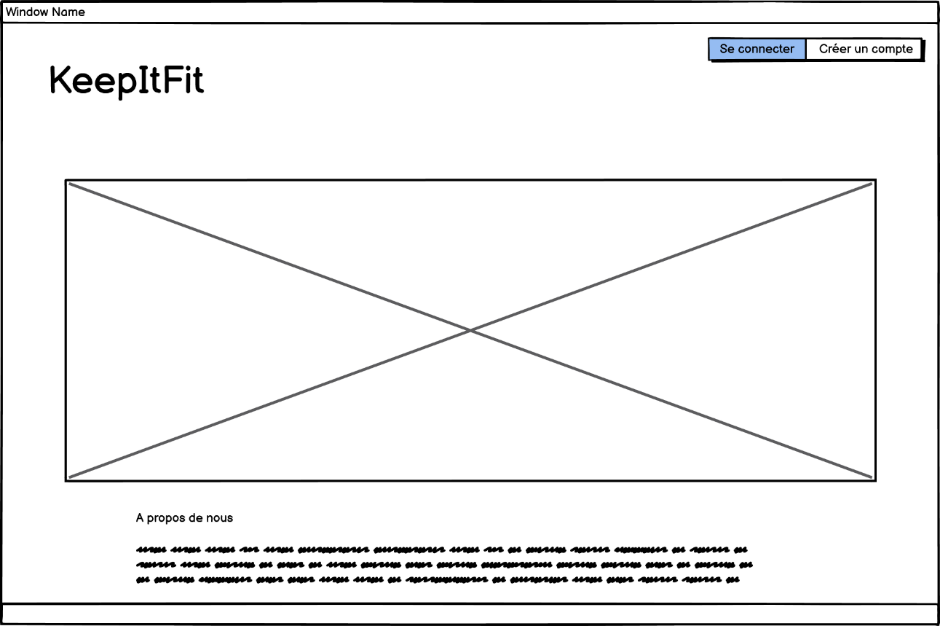
## Concept

### MCD

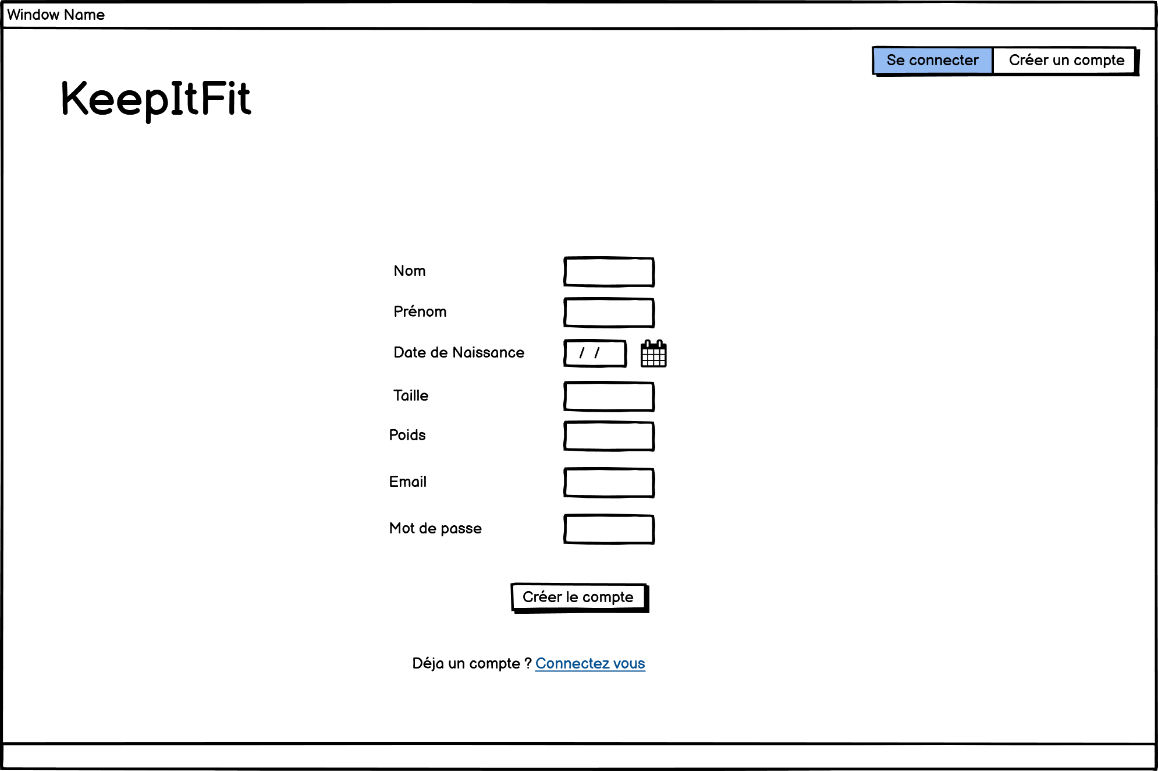
J’ai convenu avec l’expert Gilbert Gruaz que le MCD serais fait en anglais



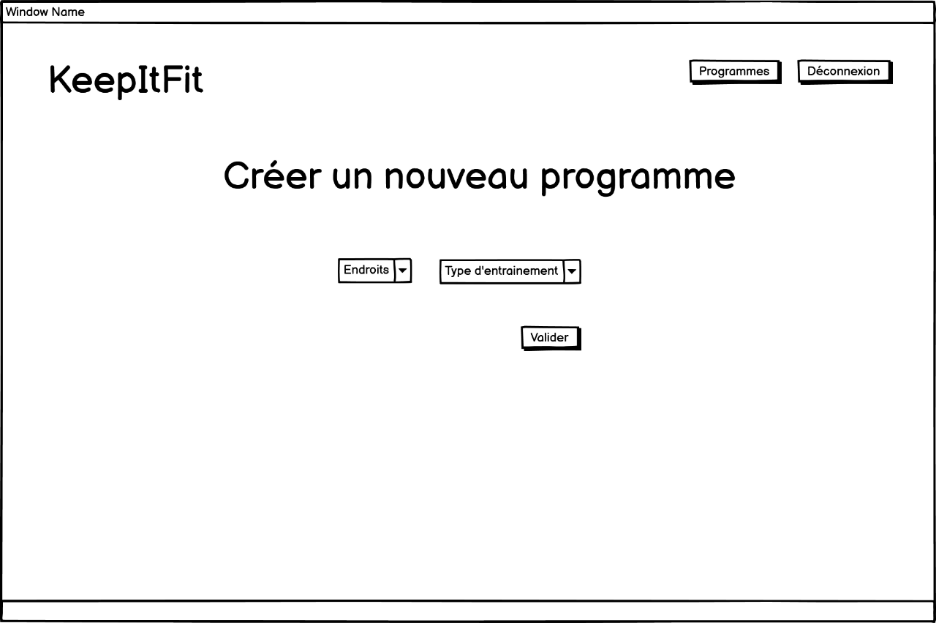
### Maquettes



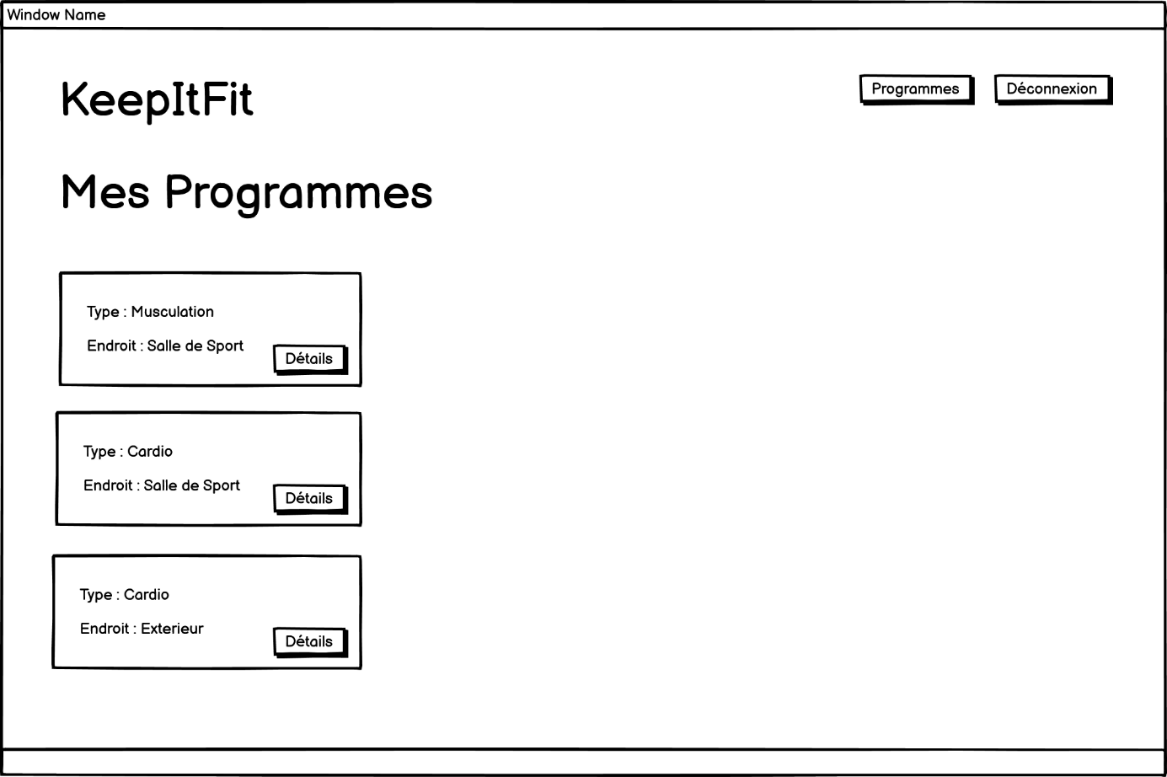
**Page d’accueil**

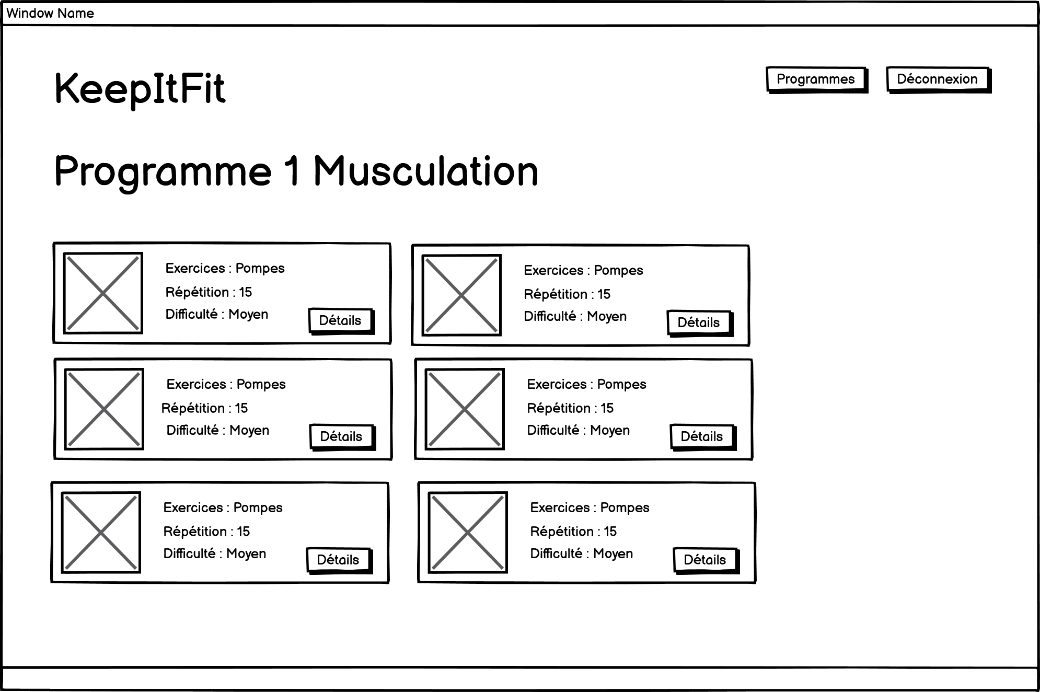
****Page d’inscription**

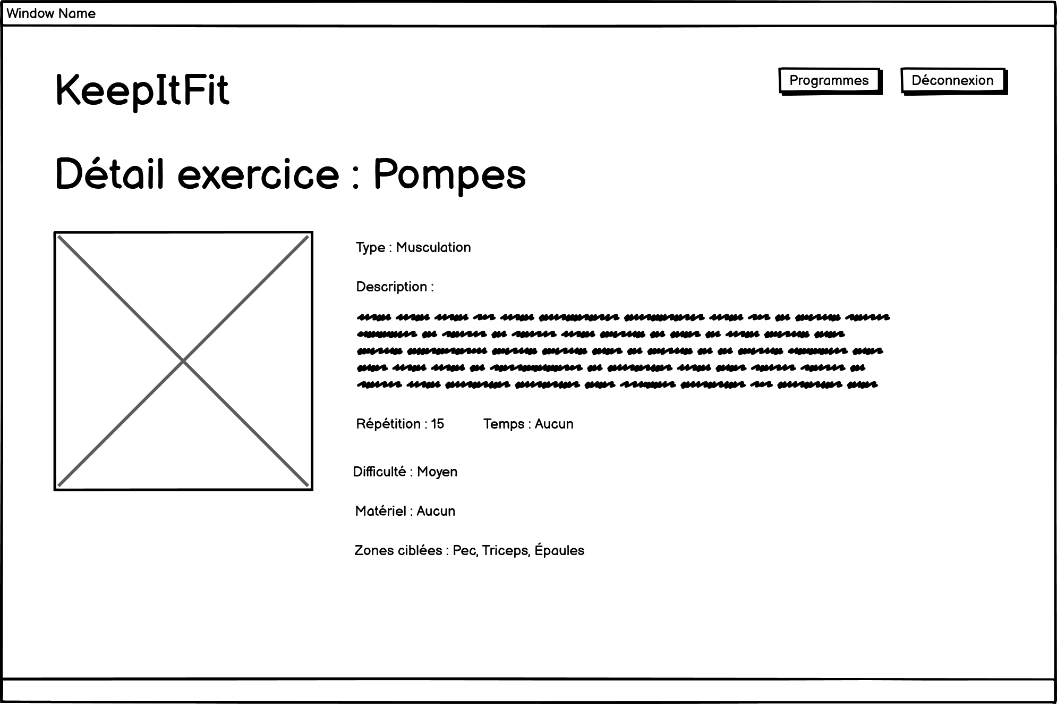
**Page de Création de programme**

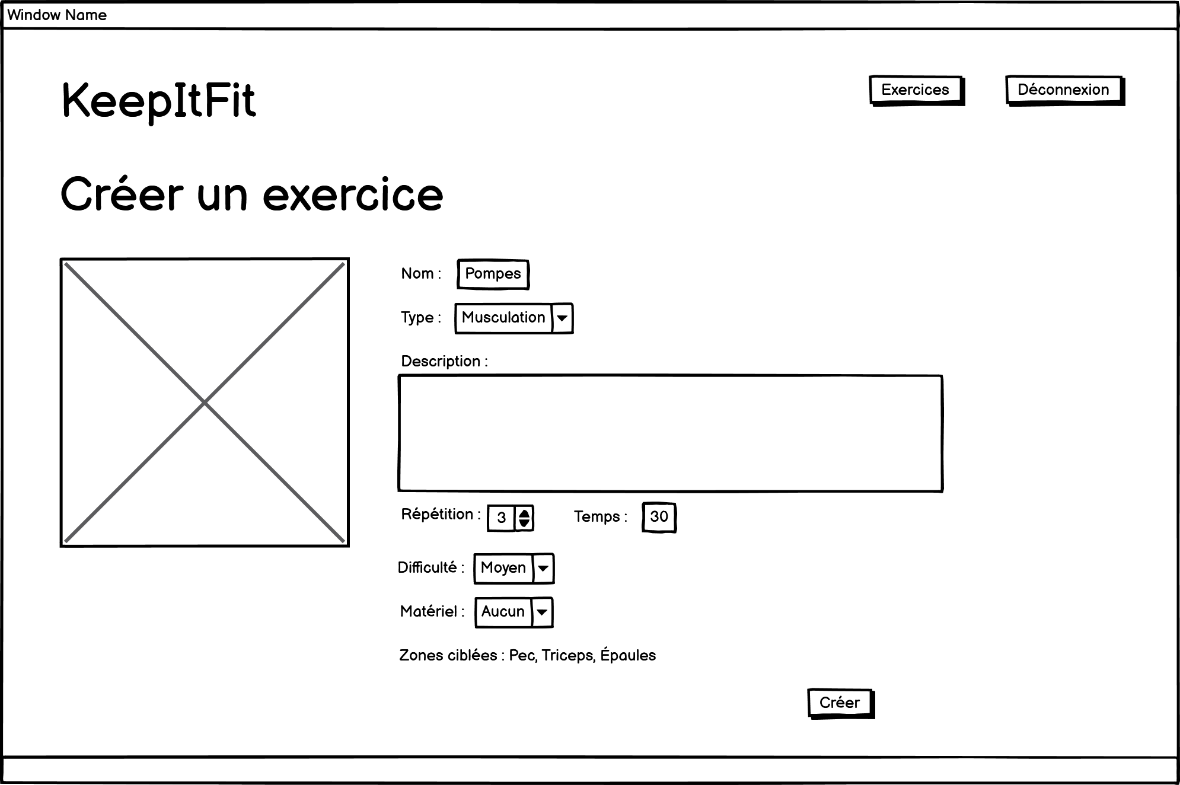
**

**Page de Programmes Personnel**

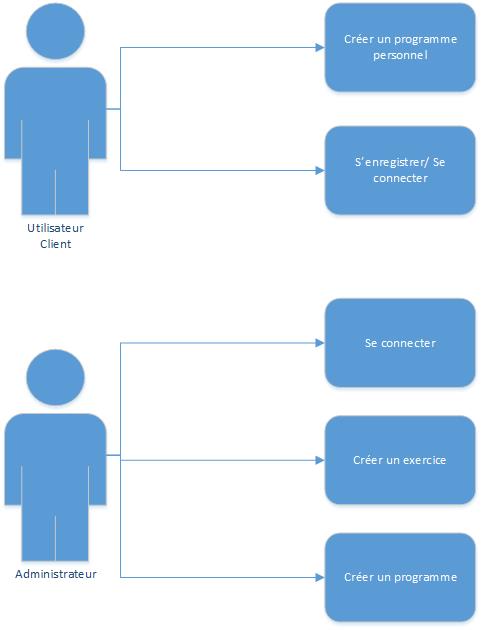
**

****Page d’un Programme Personnel**

****Page de Détail d’exercice**

****Page Création d’exercice**

## Use cases



## Stratégie de test

Pour le développement du site web, j’ai opté d’utilisation de PHP Storm (version 8.1.2), installé sur l’ordinateur du CPNV et mon ordinateur personnel avec un système d’exploitation Microsoft Windows 10.

Pour la stratégie de test je compte principalement faire des test unitaire tout du long de la réalisation du site. J’aurais aussi des test d’acceptation dans Ice Scrum qui servirons à valider les différentes User Stories.

Je déploierai mon site sur l’hébergeur « Swisscenter » pour faire les tests fonctionnels finaux.

## Risques techniques

Je pense que développer la fonction qui servira a créer un programme personnaliser pour un utilisateur sera assez complexe à faire.

De plus j’aurais un léger manque de compétence au niveau de la création d’un PDF et l’upload d’une image car je ne l’ai jamais fait.

## Planification

J’ai décidé d’utiliser la méthode agile pour ce projet.

Le projet sera découpé en 4 sprint distinct d’une semaine.

Vous pouvez retrouver la planification du projet sur [IceScrum](https://icescrum.cpnv.ch/p/SPORTTRAIN/#/project)

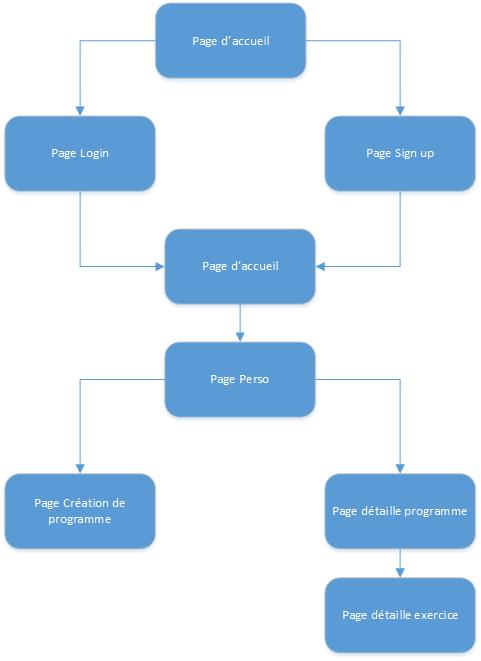
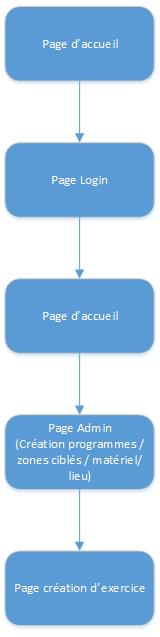
## Dossier de conception

### Choix du matériel

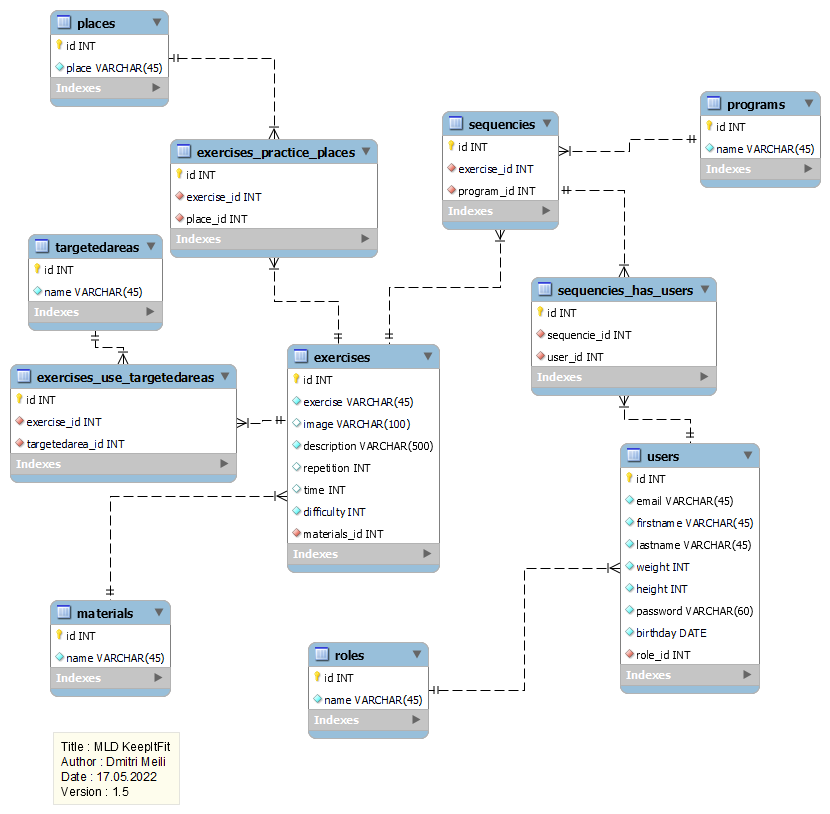
* PC du CPNV
* PC personnel
* Système d’exploitation Windows 10
* PhpStorm 2021.3.2
  + PHP 8.1.2
  + Xdebug v3.1.4
* MySQL Workbench 8.0 CE
  + Server details : 8.0.28
* HeidiSQL
* Maquettes
  + Balsamiq

### Schéma de navigation

**Schéma Admin Schéma Utilisateur**

****

### MLD

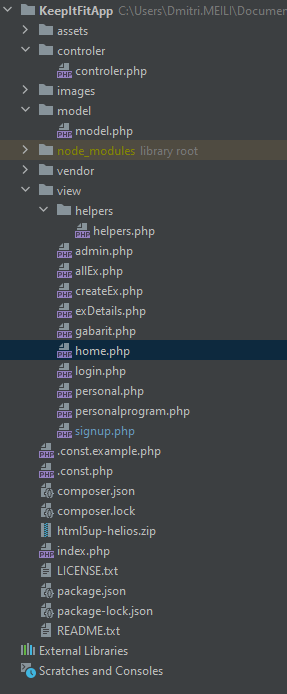


# Réalisation

## Dossier de réalisation

### Architecture du code

J’ai utilisé le model MVC



* Le dossiers **« assets »** contient les données pour le bon fonctionnement du Template
* Le dossier **« controler »** contient le fichier « controler » qui contient toutes les fonctions logiques concernant les actions effectuées par l'utilisateur.
* Le dossier **« model »** contient toutes les fonctions qui servent à faires des requêtes SQL
* Le dossier **« vendor »** est le dossier créer automatiquement lors de l’installation de Mpdf pour générer un PDF
* Le dossier **« images »** contient toutes les images utilisé dans le site.
* Le dossier **« node\_modules »** contient les données pour utiliser bootstrap.
* Le dossier **« view »** contient les interfaces graphiques que l’utilisateurs verra
  + Le dossier **« helpers »** contient les fonctions qui modifie légèrement la vue sans le besoin du controler.
* Le fichier **«.const.php »** » contient l’identifiant pour se connecter à la base de donnée
* Le fichier **«. const.example.php »** contient la structure utilisé pour se connecter à la base de donnée.
* Le fichier **« index.php »** fais le lien entre la vue et les fonctions du controler.

### Script / librairies externe

#### Scripts

* Le script qui contient seulement la structure de la BDD se trouve ici :
  + KeepItFit\Doc\BDD\scripts\ scriptStructure.txt
* Le script SQL contenant la structure de la BDD ainsi que des données se trouve ici :
  + KeepItFit\Doc\BDD\scripts\setupscript.sql

#### Libraries externes

* J’utilise la libraires Mpdf version : 8.1.1 pour générer un PDF d’un programme choisi
* J’utilise la librairie Node Modules pour l’utilisation de bootstrap

Chose que j’ai faites (interessante, technique)

### Points techniques

#### Création d’un programme personnaliser

*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

**Manque de données :**

Pour la création d’un programme il faut minimum 6 exercices qui possède le même lieu et le même type de programme. Je n’ai pas ajouté des exercices pour chaque possibilité donc pour tester si la création de programmes fonction il faut utiliser **(Lieu : Salle de fit / Type : Musculation)**

L’utilisateur peut donc pour l’instant que se créer un programme (Salle de fit / Musculation).

En tant qu’admin il suffit de créer de nouveaux exercice pour chaque possibilité mais je ne l’ai pas fait par manque de temps.

**Création d’exercice**

L’admin peut créer un exercice mais quand il créer un exercice il choisit le matériel utilisé,le type de programmes auquel il est assigné, une zones ciblée et un lieux dans lequel on peut faire cet exercice.

L’admin peut créer des zones ciblées programmes etc… ainsi que les supprimer.

Sauf que le problème est que si l’admin supprime une zone ciblée tous les exercices sont qui utilise cette zone ciblée sont affecter et il risque d’y avoir des problèmes de fonctionnement

Pour régler ça il faudrait que je fasse une fonctionnalité pour update un exercice déjà créer et aussi de faire en sorte que si un exercice est touché par cette suppression, que la table intermédiaire affiche une valeur Null

# Conclusions

## Objectif atteints / non atteints

|  |  |  |
| --- | --- | --- |
| **Objectif** | **Atteint / non atteint** | **Commentaire** |
| Partie « public » |  |  |
| Un utilisateur peut s’enregistre | O |  |
| Un mail de confirmation lui est envoyer | O/N | Quand l’utilisateur s’enregistre un mail lui est envoyer mais il n’y pas de lien pour qu’il confirme son inscription |
| Un utilisateur peut se connecter | O |  |
| Partie « utilisateur connecté » |  |  |
| L’utilisateur peut créer un programme personnaliser | O | L’utilisateur créer son programme par rapport au lieu choisi (salle de fit etc..) et du style de programmes qu’il veut (musculation etc..) |
| En cliquant sur le programme les exercices s’affichent | O |  |
| En cliquant sur un exercice les détails s’affichent | O |  |
| Un PDF du programme peut être générer | O |  |
| Partie « administrateur » |  |  |
| L’admin peut créer des exercices pour que l’utilisateurs puissent les utiliser dans ses programmes | O |  |
| L’admin peut ajouter du matériel (Haltères etc..), programmes (Musculation etc..), lieu (Salle de fit etc..) et Zones ciblée (Biceps etc..). ces info seront utiliser pour créer un exercice | O |  |
| Toutes les informations persistantes seront stockées dans une base de données | O | Les données son stocker sur une BDD MySQL |
| Le site sera responsive | O/N | Il y’a des problèmes avec la vue mobile |

## Points positifs / négatif

## Bilan personnel

Le projet c’est plutôt bien passé, j’ai progresser dans mes compétences de développement et j’ai acquérir de nouvelles connaissances en PHP.

J’ai terminé le projet assez satisfait de ce que j’ai produit. Le site en générale fonction même s’il y a des fonctionnalités qui ne sont pas complète. J’ai eu un peu de mal à gérer mon temps surtout que je suis tombé malade durant mon projet

Pendant mon projet j’ai appris comment générer un PDF en utilisant la librairie Mpdf et j’ai pu y ajouter les informations du programmes

Durant le projet j’ai rencontré quelque difficulté par exemple le bruit constant dans la classe ainsi que la période de chaleur qu’il y a eu de plus j’étais entre des serveurs ce qui n’aidait pas. Tout ceci à pas mal impacter ma concentration et donc ma qualité de travail

Mis appart ça je suis content de ce que j’ai accompli. En 1 mois j’ai réussi a créer un site fonctionnel de A-Z et j’en suis fière

## Difficultés particulières

Le bruit ambiant presque constant dans la classe sur toute la période du TPI ainsi que le la chaleur pesante pendant les deux semaines ou il faisait très chaud (météo + serveurs des collègues) ont clairement impacter mes capacités de concentration et donc ma qualité de travail.

## Suite possible pour le projet

Après le TPI je compte garder ce projet et l’améliorer dans un premier temps en perfectionnant l’algorithme pour créer un programme et en rendant le tout plus responsive.

Une fois que mon site sera « complet » je compte essayer de le refaire en application mobile.

Je pense que j’utiliserais le langage Kotlin ou flutter pour développer l’application.

# Annexes

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

**Résumé du TPI**

**Situation de départ**

Pour mon projet TPI j’ai eu pour but de créer un site web utilisant du HTML/PHP avec une base de donnée MySQL. Le sujet de mon TPI était de faire un site de sport dans lequel un utilisateur peut se connecter ou s’enregistrer. Cet utilisateur une fois connecter peut créer son programme personnaliser en fonction de deux paramètre : le lieu où il veut faire son entrainement et le type d’entrainement qu’il veut faire par exemple (Salle de fitness / Musculation).

**Mis en œuvres**

J’ai dû passer quelques jours pour adapter le Template à mon site. Le développement de ma fonctionnalité création de programmes personnaliser était ma la fonctionnalité principale de mon site mais aussi la fonctionnalité la plus compliquer à développer. J’ai dû réfléchir longuement a la logique que j’utiliserais pour développer l’algorithme

**Résultats**

Le site a été déployer sur l’hébergeur Swisscenter, on y a donc accès en tapant <http://keepitfit.mycpnv.ch/> sur internet. Un utilisateur peut se connecter et se créer un programme perso et le consulter, voir les détails des différents exercices ainsi que de générer un PDF du programme pour voir tous les exercices.

L’admin peut créer de nouveaux exercices et ajouter de nouveaux matériels, lieux, zones ciblées et type de programmes.

## Sources – Bibliographie

J’ai utilisé pour tous mes problèmes en PHP/HTML/CSS

Php.net : <https://www.php.net/>

Stack Overflow : <https://stackoverflow.com/>

W3school : <https://www.w3schools.com/>

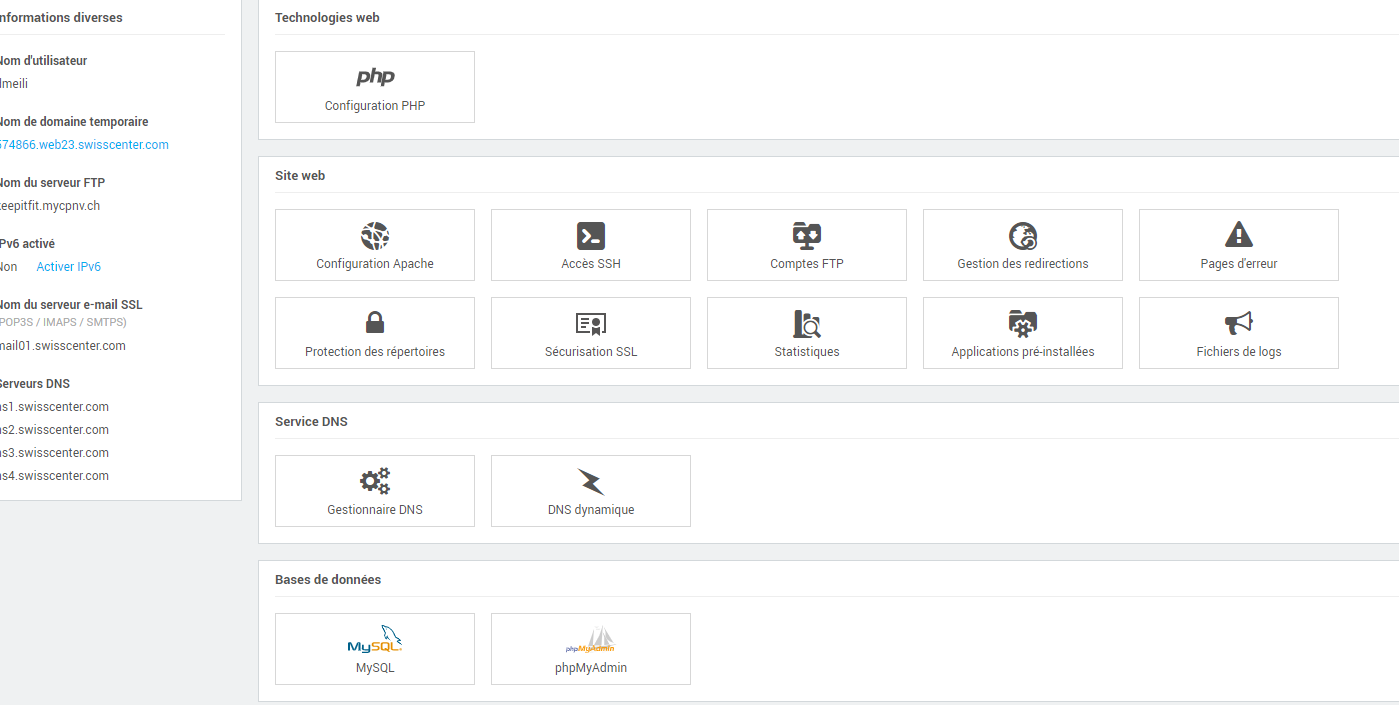
Le liens du tuto pour Mpdf

<https://www.youtube.com/watch?v=MnIps8Yc8CY>

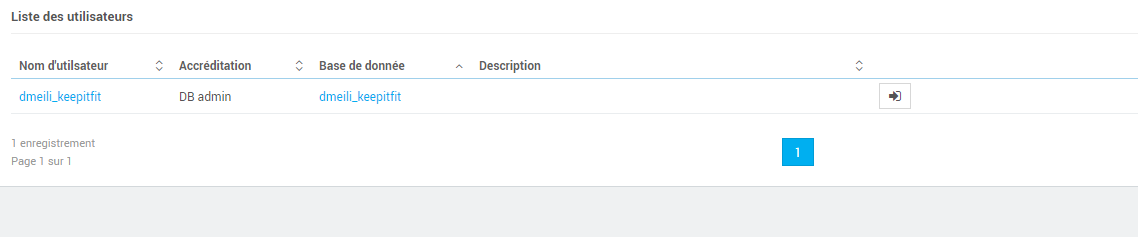
## 

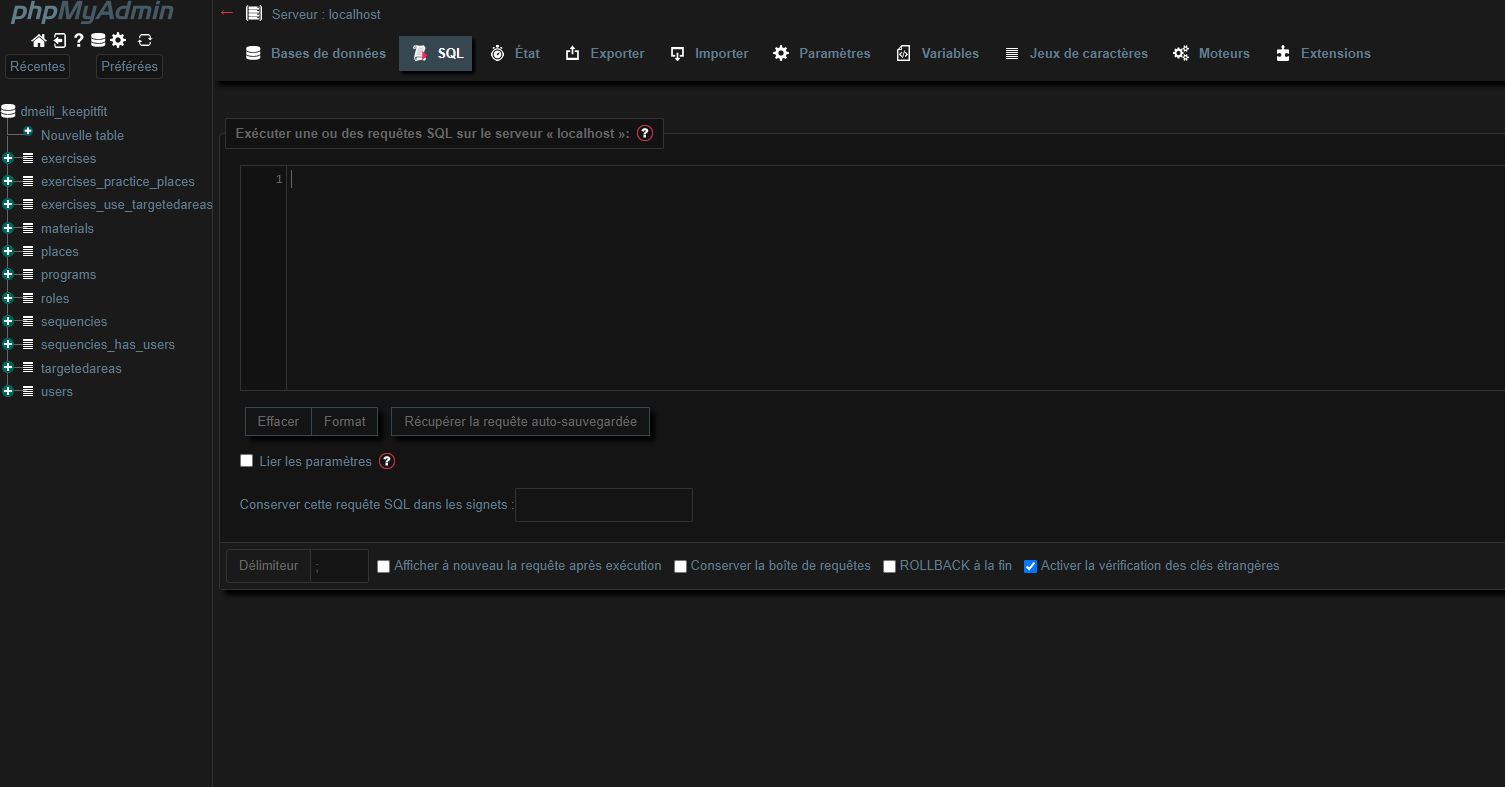
## Manuel d'Installation

**Update le site sur Swisscenter avec SSH**

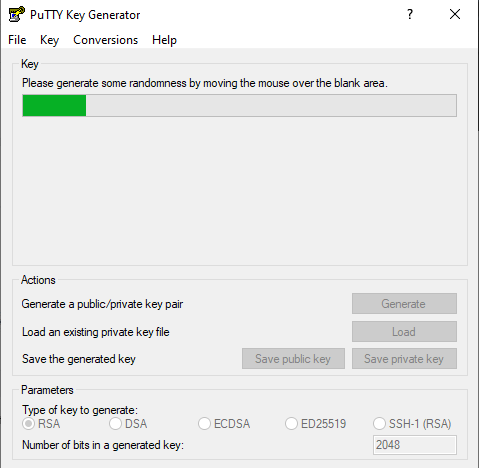
**Se connecter à Swisscenter puis aller dans « phpMyAdmin »**

**Choisir la base de donnée keepitfit**



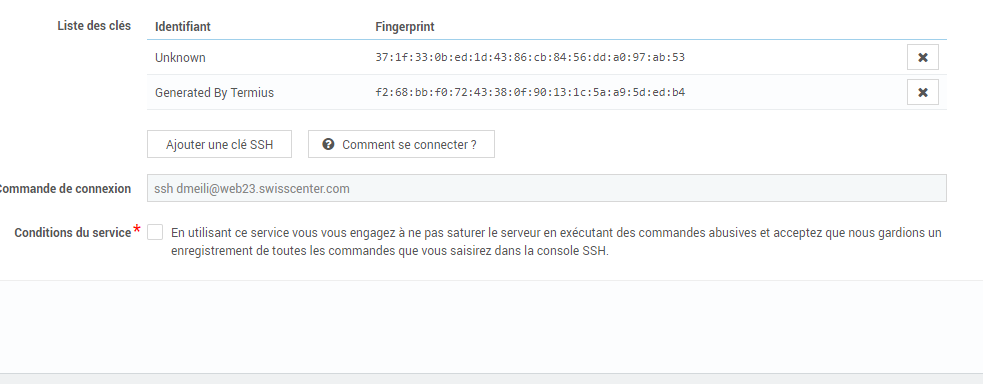
**Copier votre script SQL en s’assurant que le nom de la base de donnée est celui de Swisscenter « dmeili\_keepitfit »**

**Généré des clés SSH avec putty gen**

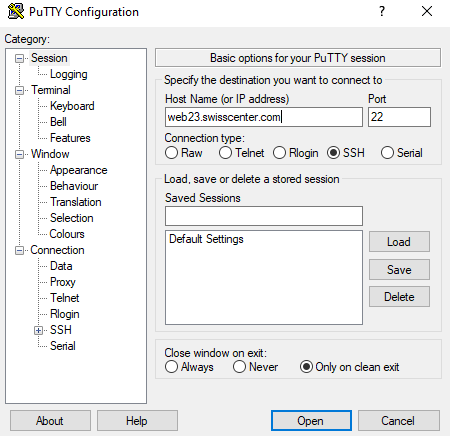


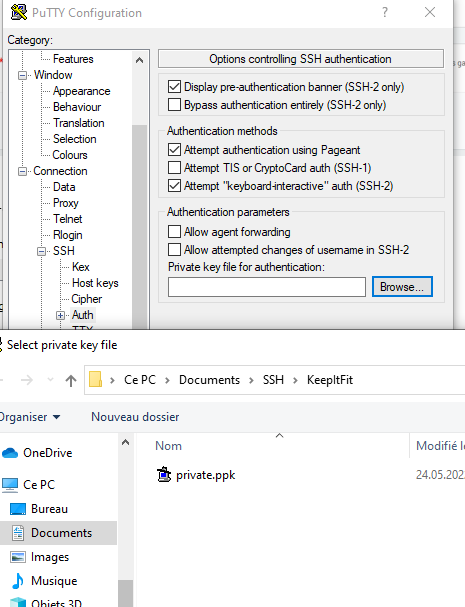
**Sauvegarder les clés dans un endroit sur**

**Ajouter la clé publique dans Swisscenter**



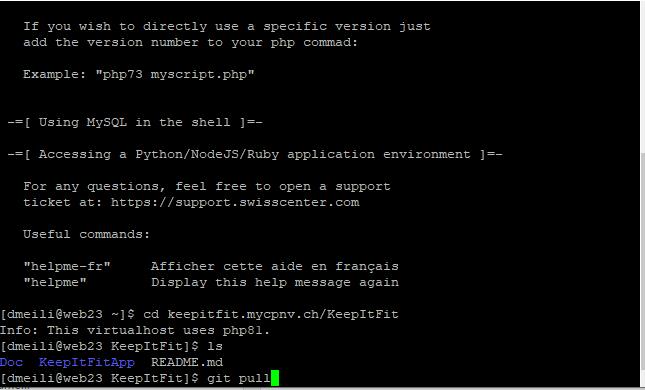
**Se connecter en SSH sur Swisscenter en utilisant Putty**





**Se connecter avec l’utilisateur « dmeili »**

**Se mettre à la racine du git et pull**



Teste en lançant le site <http://keepitfit.mycpnv.ch/>

## Glossaire

**MCD :** Modèle conceptuel de données

**MLD :** Modèle logique de données

**SSH :** (ou Secure SHell) **est** un **protocole** qui facilite les connexions sécurisées entre deux systèmes à l'aide d'une architecture client/serveur et permet aux utilisateurs de se connecter à distance à des systèmes hôte de serveurs.