|  |
| --- |
| Gestion des membres d’un club d’échecs |

Table des matières

[1 Analyse préliminaire 3](#_Toc126765392)

[1.1 Introduction 3](#_Toc126765393)

[1.2 Objectifs 5](#_Toc126765394)

[1.3 Planification initiale 5](#_Toc126765395)

[2 Analyse / Conception 6](#_Toc126765396)

[2.1 Concept 6](#_Toc126765397)

[2.2 Stratégie de test 7](#_Toc126765398)

[2.2.1 Test sur la base de données 7](#_Toc126765399)

[2.2.2 Test sur le site web 8](#_Toc126765400)

[2.3 Risques techniques 9](#_Toc126765401)

[2.4 Planification 9](#_Toc126765402)

[2.5 Dossier de conception 10](#_Toc126765403)

[3 Réalisation 12](#_Toc126765404)

[3.1 Dossier de réalisation 12](#_Toc126765405)

[3.2 Description des tests effectués 12](#_Toc126765406)

[3.3 Erreurs restantes 12](#_Toc126765407)

[3.4 Liste des documents fournis 13](#_Toc126765408)

[4 Conclusions 13](#_Toc126765409)

[5 Annexes 14](#_Toc126765410)

[5.1 Résumé du rapport du TPI / version succincte de la documentation 14](#_Toc126765411)

[5.2 Sources – Bibliographie 14](#_Toc126765412)

[5.3 Journal de travail 14](#_Toc126765413)

[5.4 Planification détaillée 15](#_Toc126765414)

[5.5 MCD et MLD 18](#_Toc126765415)

[5.6 Maquette des pages du site web 19](#_Toc126765416)

[5.7 Manuel d'Installation 21](#_Toc126765417)

[5.8 Archives du projet 21](#_Toc126765418)

# Analyse préliminaire

## Introduction

Le but de ce projet est de réaliser une application web permettant la gestion des membres d’un club d’échecs. Cette application permettra à plusieurs personnes du club de gérer l’ensemble des membres. L’application doit être réalisée en PHP et liée à une base de données MYSQL. Le projet se déroule dans un cadre de TPI de l’ETML dans l’environnement matériel et logiciel correspond à un ordinateur standard de l’ETML, à Visual Studio Code avec un environnement PHP installé, à un serveur web local, une suite Microsoft Office pour la documentation et un dépôt Git.

Les raisons de ce choix de projet sont que nous étions fortement intéressés par le résumé d’un projet en coalition avec d’autres métiers qui se réunisse sur un thème d’écologie. Mais malheureusement, le projet susdit n’est pas assez avancé pour que les enseignants responsables introduisent les apprentis dans le projet. C’est pourquoi le projet sur le thème de l’écologie sera fait en parallèle de manière aléatoire durant ce projet. Le projet principal, sur la gestion des membres d’un club d’échecs, va être réalisé à l’aide de logiciel DB\_Main et Uwamp pour la modélisation de la base de données et l’hébergement de serveur local.

Ce projet, nous ferons utiliser les compétences acquises dans les modules 104 implémentations d’un modèle de base de données,105 bases de données en SQL, 120 programmations d’interface graphique, 133 développements d’application web, 151 bases de données liées à un site web,226 programmations orientées objets, 302 bureautiques avancées, 306 réalisations de petit projet, 403 programmations structurées et 404 programmations Bassée objets.

Les points évalués dans ce projet sont l’authentification avec un nom d’utilisateur et un mot de passe, les opérations CRUD (ajout, modification, suppression et afficher les détails d’un membre) sur un membre, une recherche par multicritère doit être implémenté, une implémentation de filtre permettant d’ordonner les membres selon un critère précis, la modélisation de la base de données respectant la nomenclature Merise et le MCD/MLD / MPD sont présents et corrects, les maquettes des différents écrans doivent être réalisées dans le respect des critères UX et une exportation en PDF de la liste des membres du club doit être possible.

Le projet sera évalué sur ces livrables qui sont une planification initiale, un rapport de projet contenant au minimum l’analyse du projet, l’analyse devra contenir des informations sur les objectifs, le contenu et le public cible avec les conséquences, la réalisation répondant au cahier des charges ainsi qu’une charte graphique, une maquette du design, les schémas de base de données, les tests planifiés et effectués avec les résultats et analyses de ceux-ci, un résumé du projet, une conclusion contenant, en outre, des considérations de réalisation du projet, un journal de travail, le code source de l’application, les scripts de base de données et un guide d’installation de l’application.

## Objectifs

Les objectifs du projet sont aux nombres de sept. Ce sont sept points techniques qui seront évalué chacun indépendamment des autres. Les objectifs sont les suivants :

1. Authentification avec un nom d’utilisateur et mot de passe
2. Les opérations CRUD d’un membre (Ajout, modification, suppression et afficher les détails d’un membre).
3. Une recherche par multicritères doit être implémentée (Un système de recherche « avancée »)
4. Implémentation des filtres permettant d’ordonner les membres selon un critère (Le nom, le prénom, etc.)
5. La modélisation de la base de données respecte la nomenclature Merise et le MCD / MLD / MPD sont présents et corrects
6. Des maquettes des différents écrans doivent être réalisées dans le respect des critères UX (simplicité, cohérence, interaction, crédibilité, etc)
7. Export PDF de la liste des membres

## Une image contenant texte, mots croisés Description générée automatiquementPlanification initiale

Figure 1:Capture d'écran Plannification Initial

L’image ci-dessus montre comment nous avons planifié initialement le projet. En effet nous avons prévu de tester et de rédiger le rapport chaque semaine. Et, nous avons prévu la première semaine des tests pour savoir si l’utilisation de Laravel est possible. Et, nous avons prévu la dernière semaine d’évaluer et réaliser la présentation afin de présenter le projet à la classe les semaines suivantes la fin du projet.

# Analyse / Conception

## Concept

Mise en page / UX / design site

Le concept du site web de gestion des membres d’un club d’échecs est assez simple. La mise en page du site se réalise de la manière suivante. La maquette du site est disponible en annexes. Une page de login avec un formulaire qui regarde dans la base de données si l’utilisateur entrée existe ou non. Si l’utilisateur entrée n’existe pas la page affiche un message d’erreur. Quand on rentre un utilisateur existant avec le bon mot de passe le formulaire de login nous redirige sur la page de liste des membres. La page affiche par défaut tous les membres du club dans un format de tableau, les colonnes sont les champs de la base de données et les lignes sont les informations d’un membre du club. Cependant la dernière colonne du tableau est identique pour tous les membres, cela est pour la fonction CRUD.

Base de données

La base de données pour ce projet est assez petits vu qu’elle fait six tables relationnelles. La conception de la base de données se base sur une table centrale qui est la table t\_member. Cette dernière se constitue d’un identifiant de champ pour le nom, le prénom, la date d’anniversaire, le numéro de téléphone, la licence et pour le rang ainsi que d’une clef étrangère pour le titre obtenue et d’une autre clef étrangère pour la catégorie du membre. Liée à cette table, nous avons la table t\_title et la table t\_category. Ces dernières sont composées d’un identifiant et un champ dédié aux titres et catégories du club d’échecs. Ensuite, nous avons une table d’équipe qui est constituée uniquement d’un identifiant se nommant t\_team. Lié à la table t\_member et t\_team, nous avons la table t\_play. Celle-ci sert à savoir quel membre joue dans quelle équipe. t\_play fait la liaison entre les deux tables et précise si le membre est capitaine de l’équipe ou pas. Le MCD et MLD sont disponibles en annexe. Nous avons aussi implémenté une table t\_user pour la vérification de la page d’authentification. Elle se compose d’un identifiant, d’un nom d’utilisateur et d’un mot de passe haché.

Programmation

La structure de fichier est assez simple. Le code est mis dans un UwAmp pour héberger localement le site. À la racine, dans le dossier « www » du UwAmp, nous trouvons les dossiers « resources », « src », « userContent » et le fichier « index.html » qui redirige sur la page de login du site web. Dans le dossier « resources », il y a un dossier « css » qui contient tous les fichiers .css du site, « images » qui contient toutes les images utiliser dans le site et un dossier « lib » qui contient toutes les librairies utiliser dans le site mais vu que nous n’utilisons pas de librairie dans ce site les scripts de base donnée sont là. Le dossier « userContent » contient tout le contenue propre à l’utilisateur. Maintenant, le dossier « src » qui contient trois dossiers. Le dossier « html » pour les pages web statices, « js » pour les scripts en javascript et « php » pour les pages web dynamiques. Toutes les pages .php communiquent avec la base de données grâce à la page « dbManage.php ».

*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

### Test sur la base de données

1. Vérification que le script SQL pour la création de la base de données ne crée pas d’erreur.
2. Vérification que le script SQL pour l’insertion de donnée test dans la base de données ne crée pas d’erreur
3. Vérification que le fichier dbManage.php se connecte sans erreur à la base de données.
4. Vérification que les requêtes faites dans le fichier dbManage.php ne créent pas d’erreur sans l’intervention d’un utilisateur.

### Test sur le site web

1. Vérification que les requêtes faites dans le fichier dbManage.php ne créent pas d’erreur avec l’intervention d’un utilisateur.
2. Vérification que les requêtes faites dans le fichier dbManage.php ne créent pas d’erreur avec l’intervention d’un utilisateur malveillant.
3. Vérification que les redirections de fichier ne provoquent pas d’erreur.
4. Ajout d’un membre avec le formulaire adéquat.
5. Modification d’un membre avec le formulaire adéquat.
6. Voir les détails d’un membre.
7. Exporter en PDF la liste des membres avec leur informations.
8. Se connecter avec un nom d’utilisateur et un mot de passe.
9. Vérification des filtres permettant d’ordonner les membres selon un critère.
10. Vérification que la recherche multicritère fonctionne sans erreur.

*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

Les risques techniques durant ce projet sont surtout liés à nos compétences. Cela est dû aux faites que nous n’avons plus pratiquer de programmation web depuis deux ans. Les compétences ont été acquises mais plus utilisés depuis longtemps donc il faudra du temps pour retrouver les bonnes habitudes et manières de faire.

La solution est simple. Il faut se remettre dans le bain, pratiquer et surtout se renseigner pour ne pas faire des erreurs. Une relecture des modules correspondant et avec l’aide de sites spécialisés devrait nous aider à accomplir le projet.

## Planification

Nous allons utiliser une méthode des six pas pour ce projet. Ce projet se déroule sur huit semaines. Une semaine type se déroule avec le lundi matin quatre périodes et l’après-midi aussi, le mercredi matin cinq périodes et l’après-midi quatre périodes, le jeudi après-midi uniquement avec quatre périodes et nous finissons la semaine avec vendredi cinq périodes le matin et quatre l’après-midi.

Nous pouvons voir aussi sur la planification détaillée disponible juste après que soit nous surestimons nos capacités où nous sous-estimons la complexité du projet. De ce faite les dernières semaines sont libres mais le projet va durer plus longtemps vu que nous allons participer à un autre projet en même temps. Le second projet va interférer avec le projet principal de manière aléatoire. Le second projet va repousser les échéances du projet principal.

Six pas expliquer

La planification détaillée en annexes.

## Dossier de conception

Pour réaliser ce projet, nous avons utilisé un ordinateur standard de l’ETML sous Windows 10. Nous avons choisi d’utilisé le logiciel DB\_Main pour la modélisation de base de données et le logiciel UwAmp pour l’hébergement de serveur local. Nous avons choisi ces logiciels pour une raison assez simple. C’est que nous avons dans le passé déjà créé des projets de site web dynamique lié avec une base de données dans UwAmp. Et, nous avons déjà modélisé des bases de données dans DB\_Main. Ce sont des logiciels que nous avons eu utilisé dans les modules correspondants.

Nous avons réalisé les maquettes des pages du site web sur Figma. Figma est un outil en ligne qui permet plein de chose dont réaliser des maquettes de site web ou d’application. Nous avons utilisé cet outil dû à la recommandation de notre chef de projet M. Charmier. C’était la première fois que nous utilisions ce logiciel. Les maquettes sont les suivantes, la page de login, login avec erreur de connexion, page d’affichage de la liste des membres – une avec une recherche multicritère et une autre avec une simple recherche – et une page d’affichage, de modification et d’ajout de membre.

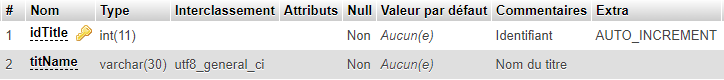
Les maquettes sont disponibles en annexes.

La base de données se découpe en six tables différentes qui sont les suivantes :

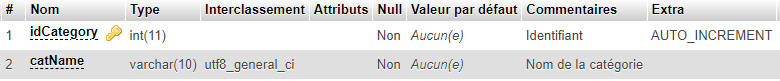
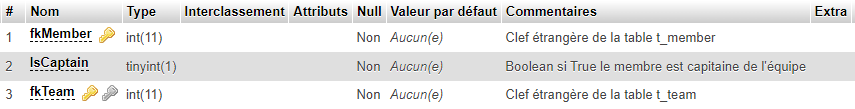
Une image contenant texte

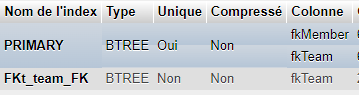
Description générée automatiquement

La table « t\_member » contient les données des membres du club d’échecs. Pour les différencier entre eux, nous avons mis un identifiant unique qui s’incrémente automatiquement à chaque nouveau membre ajouté dans la base de données (idMember). « idMember » est un int de grandeur 11 qui est mis par défaut. Nous avons décidé que pour qu’un nouveau membre existe, il doit avoir obligatoirement entrée son nom (memLastName), prénom (memFirstName), sa date de naissance (memDateBirth), son numéro de téléphone (memPhoneNumber) et à quelle catégorie il appartient (fkCategory). « memLastName » est un varchar d’une longueur de 50 pour pallier les changements de nom et/ou pour en ajouter un en cas de mariage. « memFirstName » est un varchar d’une longueur de 50 pour pallier les changements de prénom en vue de la société actuel. « memPhoneNumber » est un varchar d’une longueur de 20 pour pallier les différentes façons d’entrer un numéro en Suisse. « fkCategory » est un int de grandeur 11 qui est mis par défaut. S’il est licencié (memLicencing), ou qu’il a un classement (memRanking), ou qu’il a un titre (fkTitle), ces informations peuvent être entrée à la création du membre ou plus tard grâce au formulaire de modification de membre. « memLicencing » est un char d’une longueur de 6 parce qu’une licence d’échecs se compose d’une lettre est 5 chiffres. « memRanking » est un int de grandeur de 11 qui est mis par défaut. « fkTitle » est un int de grandeur de 11 qui est mis par défaut.



La table « t\_title » sont les données des titres possibles d’un joueur d’échecs. Pour les différencier entre eux, nous avons mis un identifiant unique qui s’incrémente automatiquement à chaque nouveau ajouté dans la base de données (idTitle). Nous avons décidé que pour qu’un nouveau titre existe, il doit obligatoirement avoir un nom (titName).





*Une image contenant texte

Description générée automatiquement*

*Fournir tous les documents 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

*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

Le rapport : R-dorcapelli-Projet-Approfondissement.docx 32 du 10.02.2023 13:23

*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

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

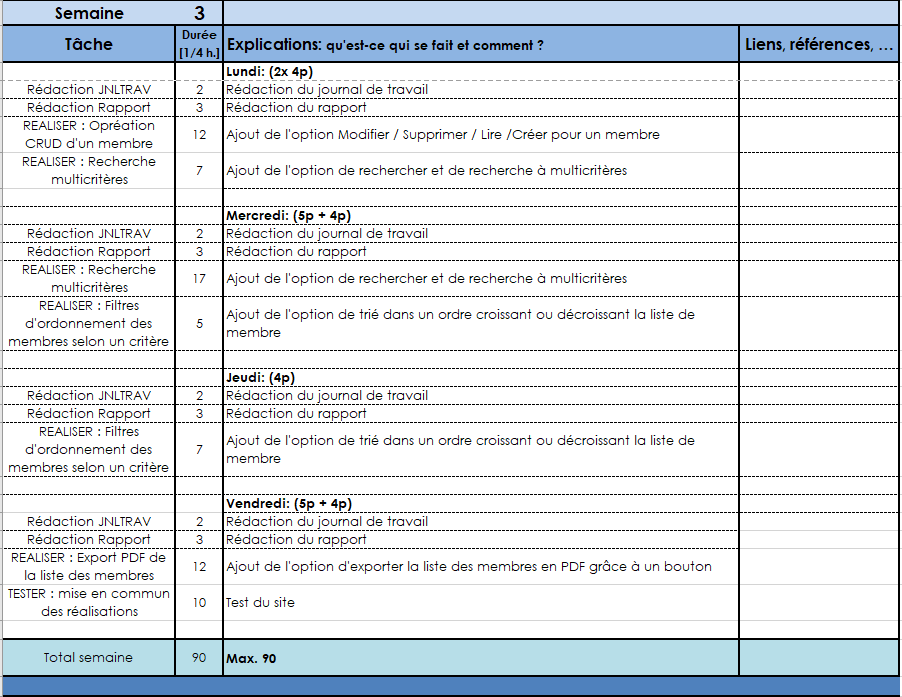
## Sources – Bibliographie

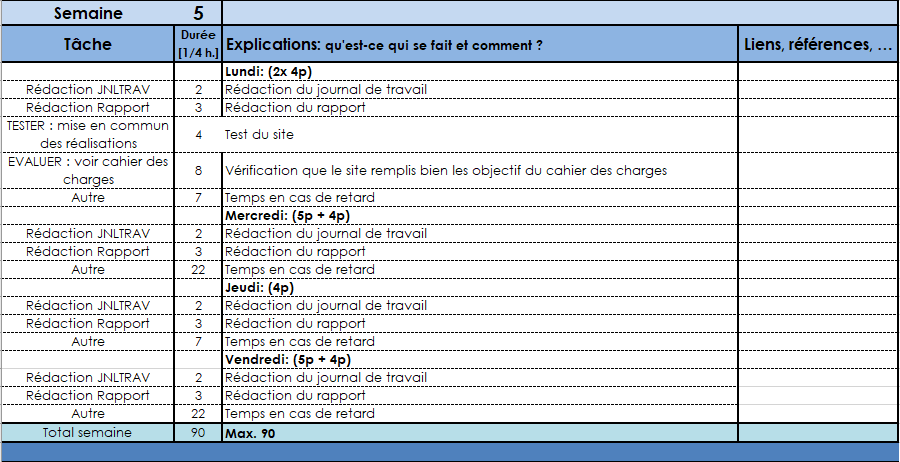
*Liste des livres utilisés (Titre, auteur, date), des sites Internet (URL) consultés, des articles (Revue, date, titre, auteur)… Et de toutes les aides externes (noms)*

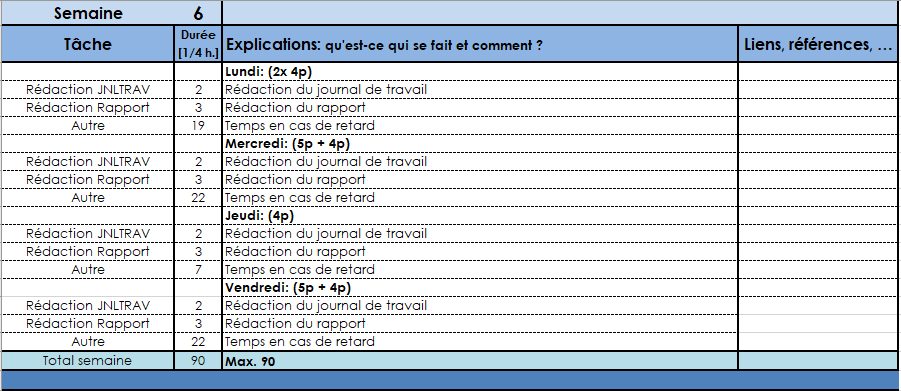
## Journal de travail

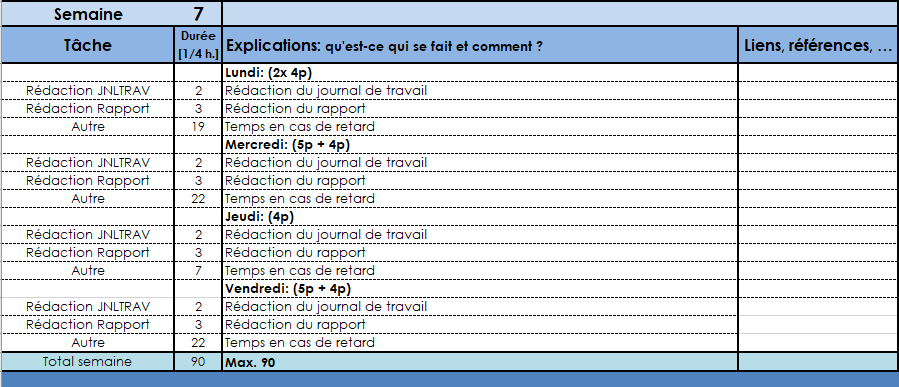
|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Durée** | **Activité** | **Remarques** |
|  |  |  |  |
|  |  |  |  |

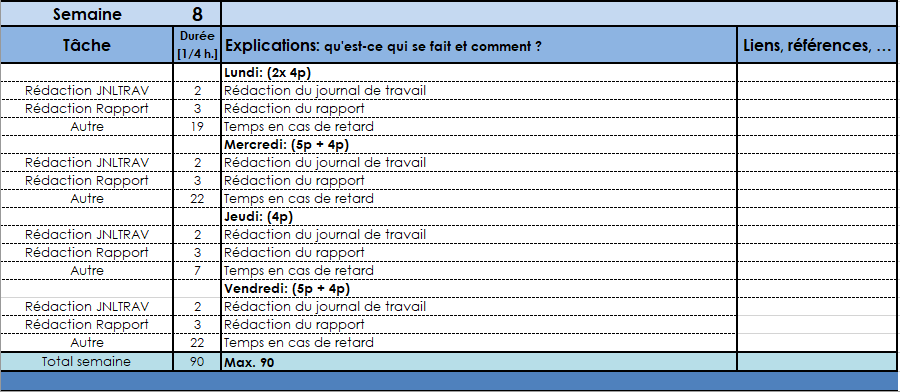
## Planification détaillée

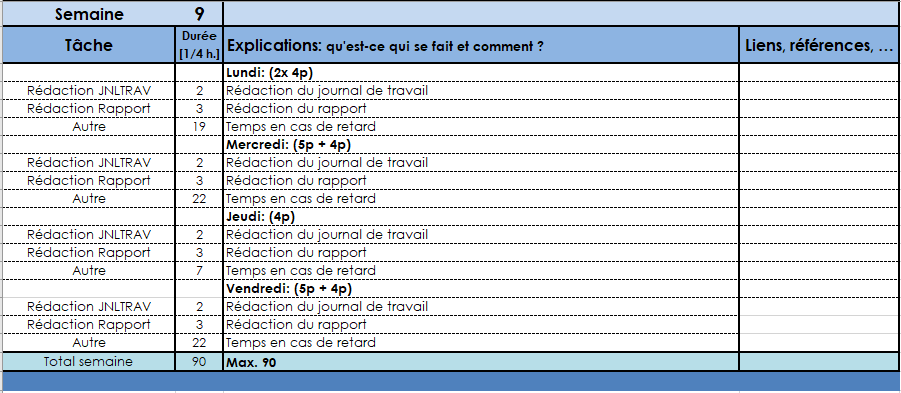




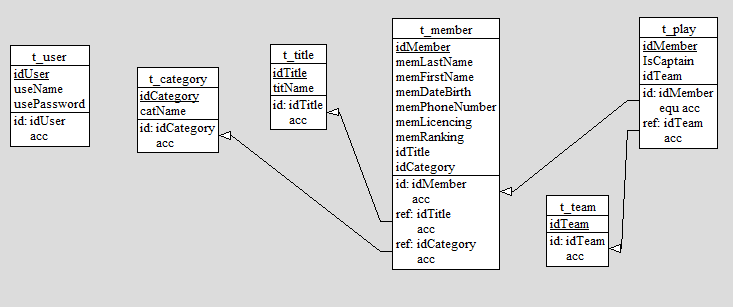








## MCD et MLD

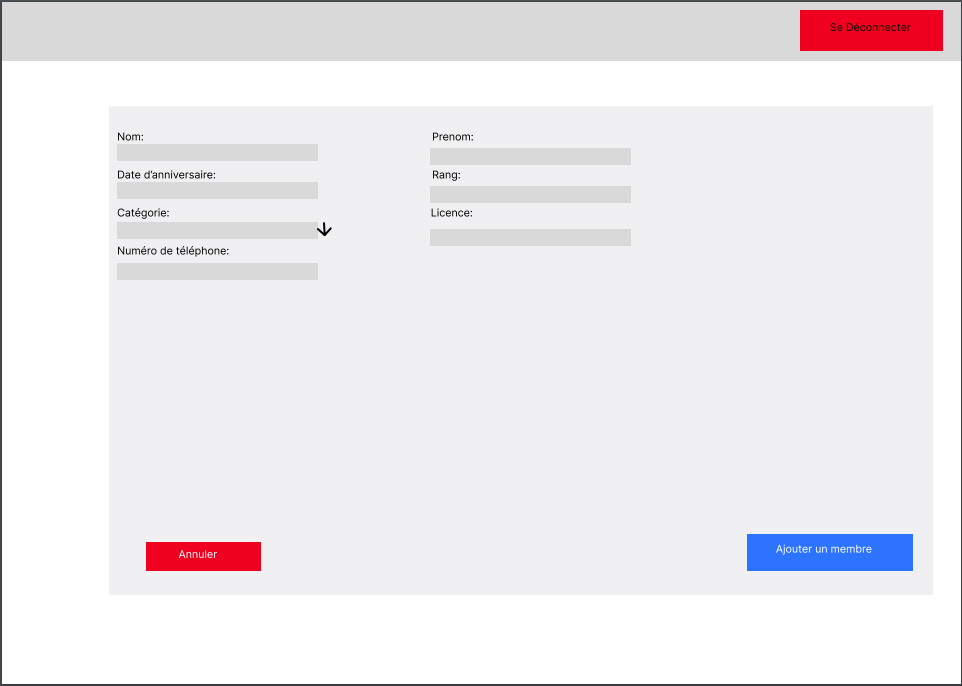


## Maquette des pages du site web

Une image contenant texte

Description générée automatiquementUne image contenant texte

Description générée automatiquement



## Manuel d'Installation

## Archives du projet

*Media, … dans une fourre en plastique*