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

Table des matières

[1 Analyse préliminaire 3](#_Toc128121739)

[1.1 Introduction ✓ 3](#_Toc128121740)

[1.2 Objectifs ✓ 4](#_Toc128121741)

[1.3 Planification initiale ✓ 4](#_Toc128121742)

[2 Analyse / Conception 5](#_Toc128121743)

[2.1 Concept ✓ 5](#_Toc128121744)

[2.2 Stratégie de test ✓ 8](#_Toc128121745)

[2.2.1 Test sur la base de données 8](#_Toc128121746)

[2.2.2 Test sur le site web 8](#_Toc128121747)

[2.3 Risques techniques ✓ 9](#_Toc128121748)

[2.4 Planification ✓ 9](#_Toc128121749)

[2.5 Dossier de conception 10](#_Toc128121750)

[3 Réalisation 17](#_Toc128121751)

[3.1 Dossier de réalisation 17](#_Toc128121752)

[3.2 Description des tests effectués ✓ 17](#_Toc128121753)

[3.3 Liste des documents fournis 20](#_Toc128121754)

[4 Conclusions ✓ 21](#_Toc128121755)

[5 Annexes 23](#_Toc128121756)

[5.1 Résumé du rapport du Projet d’Approfondissement 23](#_Toc128121757)

[5.2 Sources – Bibliographie 23](#_Toc128121758)

[5.3 Journal de travail 23](#_Toc128121759)

[5.4 Planification détaillée 24](#_Toc128121760)

[5.5 MCD et MLD 27](#_Toc128121761)

[5.6 Maquette des pages du site web 28](#_Toc128121762)

[5.7 Manuel d'Installation 30](#_Toc128121763)

[5.8 Archives du projet 30](#_Toc128121764)

# 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 identique à celui du 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.

Durant ce projet, nous utiliserons des compétences acquises pendant divers modules. (104,105, 120, 133, 151,226, 302, 306, 403 et 404.).

Pour ce projet, nous pensons consacrer ces 96 heures de telle manière que 37.5% du temps sert à la réalisation du projet en pratique, 37.5% du temps sert à la documentation du projet et 25% restant sert pour le projet d’écologie et autre.

## 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 :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 interroge 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 les opérations CRUD (Create Read Update Delete).

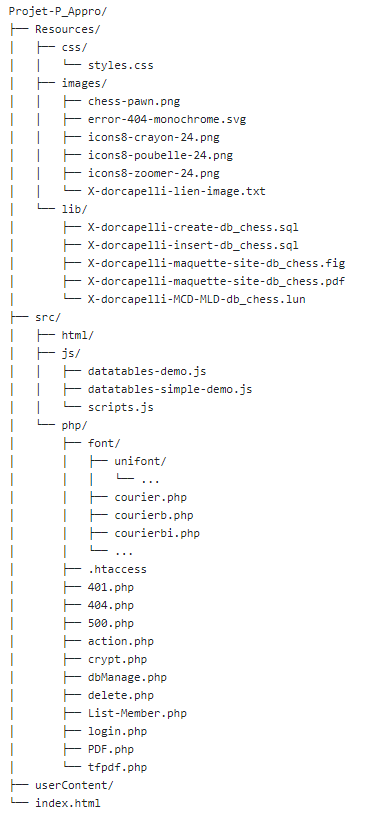
Base de données

La base de données pour ce projet est assez petite vu qu’elle ne comporte que 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 statiques, « 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 ».

Arborescence illustrée ci-dessus :



## Stratégie de test ✓

### Test sur la base de données

|  |  |
| --- | --- |
| Test | Impact en cas d’échec … |
| Vérification que le script SQL pour la création de la base de données aboutisse avec succès | …, le site web n’a pas de base de données intègre à disposition |
| Vérification que le script SQL pour l’insertion de donnée test dans la base de données aboutisse avec succès | …, le site web n’a pas de données à exploité donc aucune vérification de fonctionnalité liée aux données peut être réaliser |
| Vérification que le fichier dbManage.php réussi à se connecte à la base de données. | …, le site ne peut pas accéder aux données |
| Vérification que les requêtes faites dans le fichier dbManage.php aboutisse avec succès sans l’intervention d’un utilisateur. | …, c’est que toute ou que certaines fonctionnalités sont inutilisables |

### Test sur le site web

|  |  |
| --- | --- |
| Test | Impact en cas d’échec |
| Vérification que les requêtes faites dans le fichier dbManage.php aboutisse avec succès avec l’intervention d’un utilisateur | …, c’est que toute ou que certaines fonctionnalités ne sont pas ergonomiques |
| Vérification que les requêtes faites dans le fichier dbManage.php aboutisse avec succès avec l’intervention d’un utilisateur malveillant | …, c’est que toute ou que certaines fonctionnalités ne sont pas sécurisées |
| Vérification que les redirections de fichier sont correctes | …, c’est que certaines ou toutes les pages et/ou fonctionnalités du site ne sont pas accessible ou utilisable |
| Ajout d’un membre avec le formulaire adéquat | …, c’est qu’une fonctionnalité demandée n’est pas utilisable |
| Modification d’un membre avec le formulaire adéquat | …, c’est qu’une fonctionnalité demandée n’est pas utilisable |
| Voir les détails d’un membre | …, c’est qu’une fonctionnalité demandée n’est pas utilisable |
| Exporter en PDF la liste des membres avec leur informations | …, c’est qu’une fonctionnalité demandée n’est pas utilisable |
| Se connecter avec un nom d’utilisateur et un mot de passe | …, c’est qu’une fonctionnalité demandée n’est pas utilisable |
| Vérification des filtres permettant d’ordonner les membres selon un critère | …, c’est qu’une fonctionnalité demandée n’est pas utilisable |
| Vérification que la recherche multicritère fonctionne sans erreur | …, c’est qu’une fonctionnalité demandée n’est pas utilisable |

## 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.

La méthode des six pas est une méthodologie de planification de projet. Elle porte ce nom parce qu’elle consiste à effectuer six étapes. L’ordre des étapes est le suivant. En premier, l’étape nommée « Informer » qui consiste à récolter toutes les informations nécessaires pour le projet. En deuxième, l’étape « Planifier » qui consiste à organiser toutes les autres étapes du projet. En troisième, l’étape « Décider » qui consiste à choisir les outils et moyens utilisé pour le projet. En quatrième, « Réaliser » qui consiste comme son nom l’indique à réaliser le projet. En cinquième, « Contrôler » ou « Tester » qui consiste à vérifier que les fonctionnalités du projet sont fonctionnelles. Et, en dernier, « Evaluer » qui consiste à vérifier que toutes les attentes du projet sont remplies correctement. La méthode des six pas peut être utilisée dans des projets professionnels comme personnels vu de sa facilité d’application.

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.

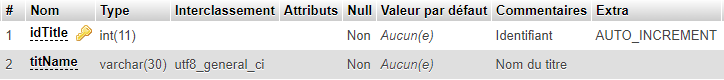
Les maquettes sont disponibles en annexes.

Une image contenant texte

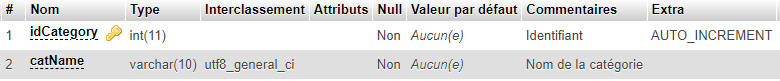
Description générée automatiquementLa base de données, qui est encoder en UTF-8, se découpe en six tables différentes qui sont les suivantes :



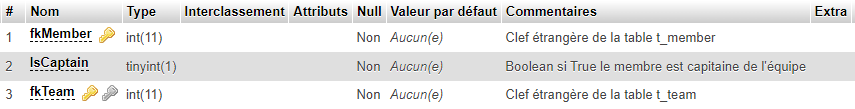
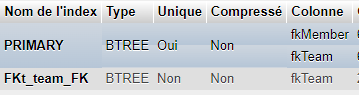
La table « t\_member » contient les données des membres du club d’échecs. Pour les différencier entre eux, nous avons mis un identifiant unique qui s’incrémente automatiquement à chaque nouveau membre ajouté dans la base de données (idMember). « idMember » est un int de grandeur 11 qui est mis par défaut. 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 aux changements de nom et/ou pour en ajouter un en cas de mariage. « memFirstName » est un varchar d’une longueur de 50 pour pallier aux 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 et 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 » contient 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 titre ajouté dans la base de données (idTitle). « idTitle » est un int de grandeur de 11 qui est mis par défaut. Nous avons décidé que pour qu’un nouveau titre existe, il doit obligatoirement avoir un nom (titName). « titName » est un varchar d’une longueur de 30 pour pallier aux changements d’intitulé de titre obtenable.



La table « t\_category » contient les données des catégories avec les quelles un joueur d’échec peut être affiliée. Pour les différencier entre eux, nous avons mis un identifiant unique qui s’incrémente automatiquement à chaque nouvelle catégorie ajoutée dans la base de données (idCategory). « idCategory » est un int de grandeur de 11 qui est mis par défaut. Nous avons décidé que pour qu’une nouvelle catégorie existe, elle doit obligatoirement avoir un nom (catName). « catName » est un varchar d’une longueur de 10 pour pallier aux changements d’intitulé des catégorie.

La table « t\_play » est une table de liaison. Elle est constituée de deux clefs étrangères et d’un boolean. Nous prenons en clef étrangère l’identifiant d’un membre et l’identifiant d’une équipe. Chaque membre peut appartenir qu’à une seule équipe donc la clef étrangère du membre peut apparaitre qu’une seul fois dans la table « t\_play ». La clef étrangère du membre (fkMember) est un int de grandeur de 11 qui est mis par défaut. La clef étrangère du membre (fkTeam) est un int de grandeur de 11 qui est mis par défaut. Chaque équipe doit avoir un capitaine (IsCaptain). « IsCaptain » est un boolean mais MySQL les présente en tant que tinyint de grandeur de 1. Les deux clefs étrangères réunies font la clef primaire unique de la table.

*Une image contenant texte

Description générée automatiquement*

La table « t\_user » est contient les données des utilisateurs qui pourront se connecter sur le site web. Pour les différencier entre eux, nous avons mis un identifiant unique qui s’incrémente automatiquement à chaque nouvelle catégorie ajoutée dans la base de données (idUser). « idUser » est un int de grandeur de 11 qui est mis par défaut. Nous avons décidé que pour qu’un nouvel utilisateur existe, il doit avoir obligatoirement entrée son nom (useName) et son mot de passe (usePassword). « useName » est un char d’une longueur de 10. Nous pensons qu’un utilisateur ne changera pas de nom. « usePassword » est un varchar d’une longueur de 255 pour entrée les mots de passe encrypter dans la base de donnée.



La table « t\_team » est constituée uniquement d’un identifiant unique qui s’incrémente automatiquement à chaque nouvelle équipe ajouté dans la base de données (idTeam). « idTeam » est un int de grandeur de 11 qui est mis par défaut. « idTeam » sert aussi de nom d’équipe.



C’est deux requêtes SQL sert à supprimer un membre de la base de données. La première supprime dans la table « t\_play » les clefs étrangères de l’identifiant du membre qu’on vas supprimer. La seconde supprime toutes les informations du membre voulu.



C’est trois requêtes SQL sert à récupérer toutes les informations des tables concernées. La première récupère toutes les données de la table « t\_category ». Elle est utilisée pour créer des listes déroulantes dans les formulaires. La seconde récupère toutes les données de la table « t\_title ». Elle est utilisée pour créer des listes déroulantes dans les formulaires. La dernière récupère toutes les informations de la table « t\_member » avec une liaison standard avec la table « t\_category ». La deuxième liaison est particulière. " LEFT JOIN " a une fonctionnalité particulière. " Cela permet de lister tous les résultats de la table de gauche même s’il n’y a pas de correspondance dans la deuxième table. "[[1]](#footnote-1) Elle utilisé pour l’affichage de la liste de tous les membres du club.

Une image contenant texte

Description générée automatiquement

Cette requête SQL sert à récupérer les informations d’un membre précis du club. Elle récupère toutes les informations de la table « t\_member » avec une liaison standard avec la table « t\_category ». La deuxième liaison est particulière. " LEFT JOIN " a une fonctionnalité particulière. " Cela permet de lister tous les résultats de la table de gauche même s’il n’y a pas de correspondance dans la deuxième table. "[[2]](#footnote-2) Elle est utilisé pour afficher les informations d’un membre précis.

Une image contenant texte

Description générée automatiquement

Cette requête SQL sert à récupérer toutes les informations des membres correspondant à des critères précis. Elle s’utilise pendant la recherche multicritère. Elle récupère toutes les informations de la table « t\_member » avec une liaison standard avec la table « t\_category ». La deuxième liaison est particulière. " LEFT JOIN " a une fonctionnalité particulière. " Cela permet de lister tous les résultats de la table de gauche même s’il n’y a pas de correspondance dans la deuxième table. "[[3]](#footnote-3) En plus, elle recherche que les membres correspondants aux données entrée dans la recherche multicritère.



Cette requête SQL sert à ajouter un membre au club. Elle insert les données transmises dans la table « t\_member ».

Une image contenant texte

Description générée automatiquement

Cette requête SQL sert à mettre à jour les informations d’un membre. Elle modifie le membre concerné en redéfinissant ces attributs.

Le MCD et MLD sont disponibles en annexe.

*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…*

# 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 ✓

| Test | Impact en cas d’échec … | Résultat attendu | Résultat obtenu | Conclusion | Temps Requis pour corriger et/ou finaliser |
| --- | --- | --- | --- | --- | --- |
| Vérification que le script SQL pour la création de la base de données aboutisse avec succès | …, le site web n’a pas de base de données intègre à disposition | Tous les messages de phpMyAdmin sont en vert et toutes les tables sont créées correctement | OK 100% | Rien d’autre à faire |  |
| Vérification que le script SQL pour l’insertion de donnée test dans la base de données aboutisse avec succès | …, le site web n’a pas de données à exploité donc aucune vérification de fonctionnalité liée aux données peut être réaliser | L’insertion de données factice dans la base de données | OK 100% | Rien d’autre à faire |  |
| Vérification que le fichier dbManage.php réussi à se connecte à la base de données. | …, le site ne peut pas accéder aux données | Aucune erreur php concernant la connexion à la base de données s’affiche | OK 100% | Rien d’autre à faire |  |
| Vérification que les requêtes faites dans le fichier dbManage.php aboutisse avec succès sans l’intervention d’un utilisateur. | …, c’est que toute ou que certaines fonctionnalités sont inutilisables | Aucune erreur php et SQL s’affiche quand les requête SQL sont exécuté | OK 100% | Rien d’autre à faire |  |
| Vérification que les requêtes faites dans le fichier dbManage.php aboutisse avec succès avec l’intervention d’un utilisateur | …, c’est que toute ou que certaines fonctionnalités ne sont pas ergonomiques | Aucune erreur php et SQL s’affiche quand les requête SQL sont exécuté | OK 100% | Rien d’autre à faire |  |
| Vérification que les requêtes faites dans le fichier dbManage.php aboutisse avec succès avec l’intervention d’un utilisateur malveillant | …, c’est que toute ou que certaines fonctionnalités ne sont pas sécurisées | Aucune erreur php et SQL s’affiche quand les requête SQL sont exécuté et aucune injection SQL peut être réaliser | OK 100% | Rien d’autre à faire |  |
| Vérification que les redirections de fichier sont correctes | …, c’est que certaines ou toutes les pages et/ou fonctionnalités du site ne sont pas accessible ou utilisable | Nous pouvons aller et utiliser toutes les pages du site web | OK 100% | Rien d’autre à faire |  |
| Ajout d’un membre avec le formulaire adéquat | …, c’est qu’une fonctionnalité demandée n’est pas utilisable | Le formulaire ne crée pas d’erreur et l’utilisateur ne peut pas ajouter un membre avec des données pas accepter | OK 100% | Rien d’autre à faire |  |
| Modification d’un membre avec le formulaire adéquat | …, c’est qu’une fonctionnalité demandée n’est pas utilisable | Le formulaire ne crée pas d’erreur et l’utilisateur ne peut pas modifier un membre avec des données pas accepter | OK 100% | Rien d’autre à faire |  |
| Voir les détails d’un membre | …, c’est qu’une fonctionnalité demandée n’est pas utilisable | Aucune erreur php s’affiche concernant la récupération et l’affichage des informations | OK 100% | Rien d’autre à faire |  |
| Exporter en PDF la liste des membres avec leur informations | …, c’est qu’une fonctionnalité demandée n’est pas utilisable | Un fichier PDF est ouvert dans une nouvelle page web et contient les informations des membres demandé et peut être imprimer | OK 100% | Rien d’autre à faire |  |
| Se connecter avec un nom d’utilisateur et un mot de passe | …, c’est qu’une fonctionnalité demandée n’est pas utilisable | Aucune erreur php s’affiche concernant la récupération de données ou de vérification de données | OK 100% | Rien d’autre à faire |  |
| Vérification des filtres permettant d’ordonner les membres selon un critère | …, c’est qu’une fonctionnalité demandée n’est pas utilisable | Aucune erreur php ou javascript s’affiche concernant trie des données | OK 100% | Rien d’autre à faire |  |
| Vérification que la recherche multicritère fonctionne sans erreur | …, c’est qu’une fonctionnalité demandée n’est pas utilisable | Aucune erreur php s’affiche concernant la récupération de données et l’affichage | OK 100% | Rien d’autre à faire |  |

## Liste des documents fournis

Le rapport : R-dorcapelli-Projet-Approfondissement.docx

Un fichier Excel contenant la planification initiale, la planification détaillée, le journal de travail et le digramme de comparaison : M-dorcapelli-planification-jnltrav.xlsm

Le manuel d’Installation :

Le code source du site web : Projet-P\_Appro.zip

Les scripts de base de données sont dans le zip du code source dans le chemin : resources\lib\ X-dorcapelli-create-db\_chess.sql et resources\lib\ X-dorcapelli-insert-db\_chess.sql

Rouge = à faire / Orange = à finir / Vert = fini

# Conclusions ✓

Pour la conclusion, nous allons procéder par étape. Les étapes sont les objectifs, les points positifs et négatifs, difficultés particulières, suite possible pour le projet et bilan personnel.

Les sept objectifs, qui étaient demandé, ont été atteints la planification se trompe sur quelle fonctionnalité nous as pris le plus de temps. Cependant, toutes les fonctionnalités sont fonctionnelles et aboutisse avec succès sans erreur.

Les points positifs sont que le projet était transparent. Il nous semblait court et assez aisé et cela était le cas. Un autre point positif inattendu est de refaire de la création de site web dynamique qui nous avez un peu manqué. Les points négatifs sont assez standard comme la documentation toujours aussi envahissante dans le projet. Comme point négatif, nous trouvons aussi la faite que le chef de projet soit présent qu’une demi-journée par semaine pour la surveillance des projets. Cela complique dans certains cas l’avancée du projet. M. Charmier était un chef de projet très compétent et très disponible même en dehors des séances mis à disposition.

Les difficultés particulières, il y en a surtout un vu que les difficultés que nous avons pensé avoir au début du projet ne sont pas arrivé. Par exemple, sur la fonctionnalité d’exportation en PDF, nous semblait très complexe à mettre en place, mais grâce à M. Philipona qui avait déjà mis en place cette fonctionnalité dans d’autres projets, nous avons réussis sans trop de problèmes. Un autre exemple est que pour la fonctionnalité d’implémentation de filtres permettant d’ordonner les membres, nous ne fessions du souci vu la complexité, mais grâce à M. Tauxe nous avons pu utiliser un Bootstrap qui réalise automatiquement la fonctionnalité. La difficulté qui était particulières dans notre cas était, le gros point négatif mentionner plus haut, d’avoir le chef de projet qu’une demi-journée présent pour de l’aide ou des questions mais cette difficulté était adoucie grâce à M. Charmier qui se montrait disponible en tout temps pendant les horaires du projet malgré son absence en classe.

Il y a plusieurs suites possibles au projet. Le code peut être améliorer facilement en prenant du temps. Le rajout de fonctionnalité peut être fait. Le design peut être améliorer pour faciliter l’utilisation et/ou esthétiquement. Un site web n’est jamais réellement fini vu que le client voudra toujours à un moment donné une modification, une amélioration et/ou une mise à jour.

Pour notre bilan personnel, nous avons apprécie le projet en lui-même. Il était rapide mais intense. Et, il nous a fait reprendre goût à la création de site web. Il semble que ce projet nous a un peu réconcilié avec l’informatique. Ce projet, nous a fait prendre un second souffle sur l’informatique en général et grâce à ça nous n’apprécions plus qu’avant le métier d’informaticien. L’ambiance de la classe a joué un rôle dans le projet de ne pas être seul à beaucoup aider moralement et techniquement. Nous avons apprécié réaliser ce projet en général.

# Annexes

## Résumé du rapport du Projet d’Approfondissement

## Sources

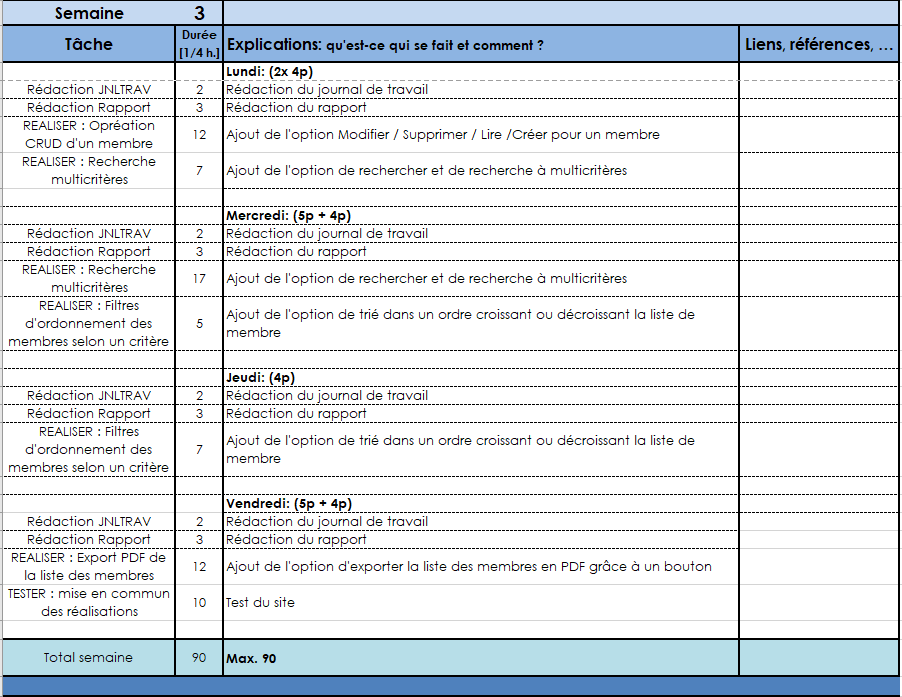
1. SQL LEFT JOIN, explication de la commande "LEFT JOIN" en SQL, <https://sql.sh/cours/jointures/left-join>

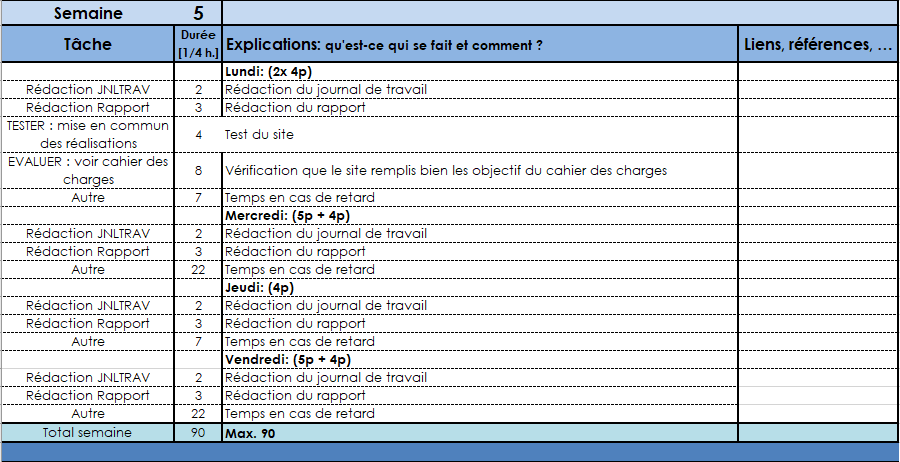
*Liste des livres utilisés (Titre, auteur, date), des sites Internet (URL) consultés, des articles (Revue, date, titre, auteur)… Et de toutes les aides externes (noms)*

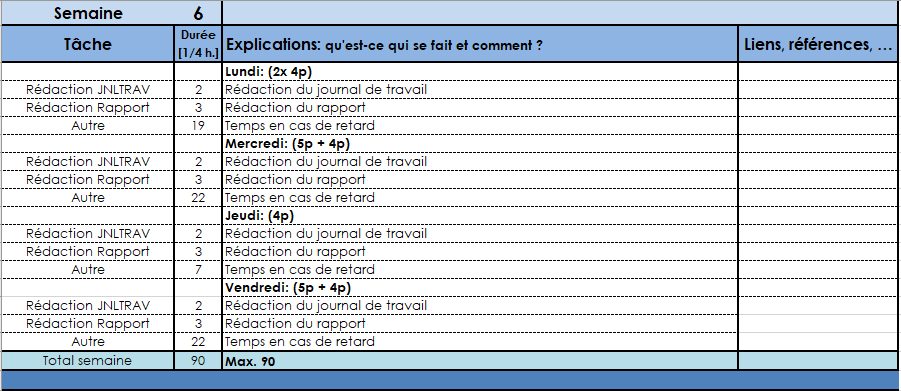
## Journal de travail

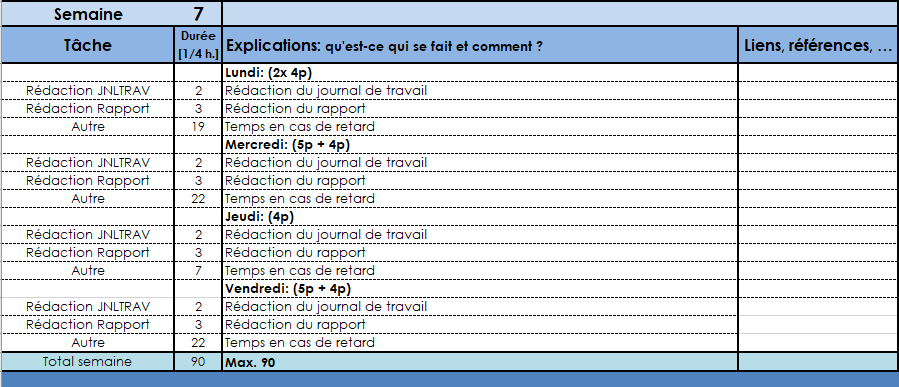
|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Durée** | **Activité** | **Remarques** |
|  |  |  |  |
|  |  |  |  |

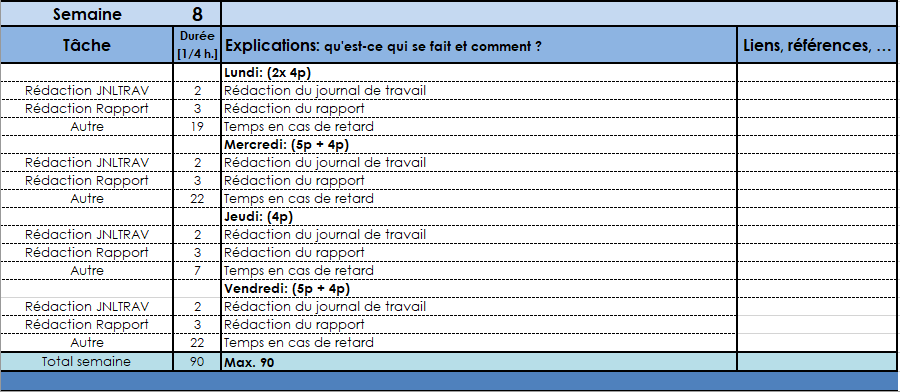
## Planification détaillée

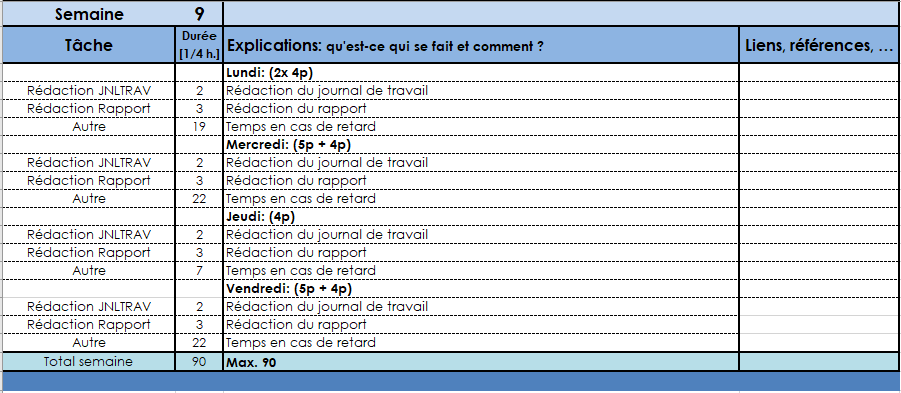




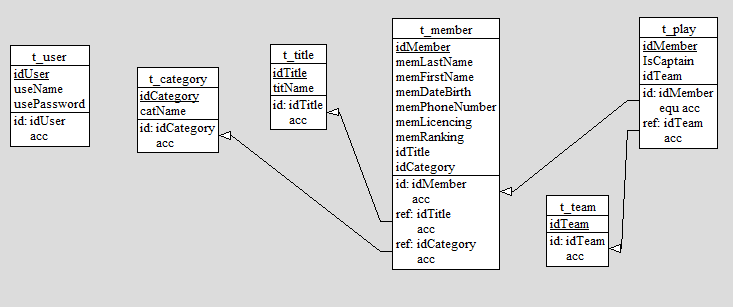








## MCD et MLD

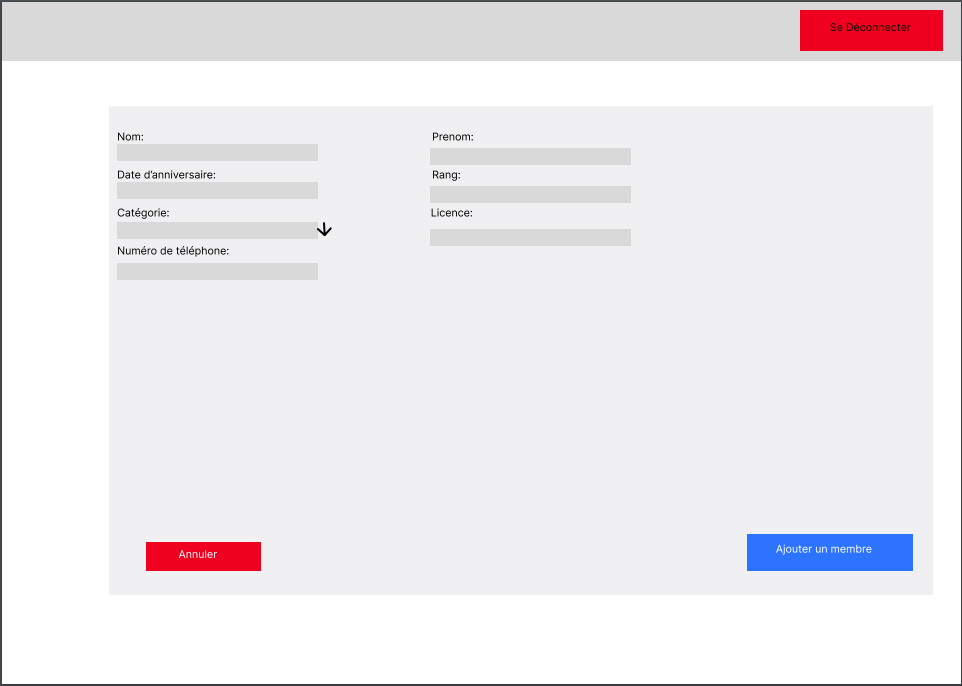


## Maquette des pages du site web

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*

1. Source 1 [↑](#footnote-ref-1)
2. Source 1 [↑](#footnote-ref-2)
3. Source 1 [↑](#footnote-ref-3)