El Letche



Vitor Silva & Ricardo Delgado Miranda

ETML, Lausanne

Cindy Hardegger

Table des matières

[1 Introduction 3](#_Toc60063991)

[2 Analyse 3](#_Toc60063992)

[2.1 Pages du site 3](#_Toc60063993)

[2.2 Base de données 4](#_Toc60063994)

[2.3 Structure 4](#_Toc60063995)

[3 Réalisation 4](#_Toc60063996)

[3.1 Gestion des images 4](#_Toc60063997)

[3.2 Eco-conception 4](#_Toc60063998)

[3.3 Ajout d’une recette 4](#_Toc60063999)

[4 Tests 4](#_Toc60064000)

[5 Conclusion 4](#_Toc60064001)

[5.1 Conclusion générale 4](#_Toc60064002)

[5.2 Conclusions Personnelles 4](#_Toc60064003)

[5.2.1 Vitor Silva 4](#_Toc60064004)

[5.2.2 Ricardo Delgado Miranda 4](#_Toc60064005)

[5.3 Méthode de projet 4](#_Toc60064006)

[6 Webographie & annexes 4](#_Toc60064007)

# Introduction

Ce projet est exécuté dans le cadre des modules 133, 151 et 105 visant à acquérir les compétences nécessaires à la création d’un petit site internet dynamique récupérant ses données dans une base de données créée pour l’occasion.

Ici, le but est de reproduire un site indexant des recettes de plats. Différentes fonctionnalités sont demandées, comme l’entrée des recettes dans la base de données directement depuis le site, la connexion au site avec ou sans certains droits administrateurs ou encore la possibilité de contacter les administrateurs du site directement par un formulaire de contact.

Le site doit suivre une structure en MVC et les éléments de style sont donnés par un Framework non-imposé (ici, Bootstrap).

La méthode de travail n’est pas imposée non plus et peut être modifiée au cours de la conception du site si nécessaire.

Le projet est rendu le 4 janvier 2021 au plus tard.

# Analyse

## Pages du site

Les pages de ce site suivent une structure MVC. Les pages sont donc divisées en 3 catégories. Les Modèles, les Contrôleurs et les Vues. Chacune de ces catégories de pages ont un rôle bien défini au sein du fonctionnement du site.

1. Les Modèles vont servir à aller récupérer les données à la base de données liée au site. C’est ici que sont faites les requêtes SQL servant à récupérer ou à insérer des données dans une base. C’est également ici que sont gardées d’autres informations comme celles servant à se lier à la bonne base de données avec le bon port, etc. Ces pages sont des pages PHP et ne contiennent normalement pas de code HTML car elles ne sont pas sensées afficher les données qu’elles récupèrent.
2. Les Contrôleurs, eux, servent à faire transiter les données récupérées par le Modèle entre ce dernier et les pages de la Vue. C’est également ici que les données vont être mises en forme si c’est nécessaire, que ce soit pour les afficher à l’écran ou pour les insérer dans la base de données. Ces pages n’affichent pas de contenu à l’écran et ne contiennent donc pas de code HTML mais uniquement du PHP. Le contrôleur principal est appelé Master Controller et est une classe. La plupart des autres pages sont des extensions de cette dernière.
3. Les vues, quant à elles, servent à afficher à l’écran tout ce qui doit être affiché à l’écran de l’utilisateur. Ces pages sont quasi totalement composées de code HTML mais peuvent également contenir du PHP. C’est ici que sont gardées les pages de code du Head, Header ou encore Footer, par exemple.

Reste ensuite la page d’index, directement à la racine du site, et qui va servir de chef d’orchestre au reste des pages et permettre leur affichage dans la bonne structure.

## Base de données

Pour la base de données, nous avons dû nous y prendre à plusieurs fois. Effectivement, nous n’avions peut-être pas une vision exacte de l’entièreté du projet dès le début. Nous avons assez vite créé notre première base de données pour pouvoir commencer à intégrer certains éléments du site de manière dynamique, mais il nous a par la suite manqué certaines choses ou d’autres pouvaient être modifiées pour faciliter ou clarifier le projet. Nous avions, par exemple, au début, créé une table pour les utilisateurs et une autre pour les administrateurs. Quand il a été temps de mettre en place les connexions, il nous a tout de suite semblé beaucoup plus logique de n’avoir qu’une table pour tous les utilisateurs, à laquelle on a ajouté une colonne qui définit les droits de l’utilisateur en question.

En dehors de cela, la création de notre base de données est assez simple. Pour commencer, nous avons essayé de réfléchir à tout ce dont nous aurions besoin afin d’oublier le moins d’éléments possible. Tous ces éléments ont été notés sur une feuille de brouillon servant de schéma à notre base de données. Nous avons ensuite utilisé le logiciel *dbMain* pour mettre en place un MCD et un MLD ainsi que pour gérer les types des différentes données qui y seraient entrées.

Les données de la base de données sont ensuite entrées dans le fichier config du site afin de faciliter une modification éventuelle dans les identifiants, par exemple. Le fichier database se charge ensuite d’utiliser les données de config pour se connecter à la base de données.

A partir de là, il est possible de récupérer et d’entrer des données directement depuis le site via des fonctions contenant des requêtes SQL.

## Structure

Structure…

# Réalisation

## Gestion des images

Gestion…

## Eco-conception

Ecologie….

## Ajout d’une recette

L’ajout d’une recette se fait via un formulaire simple auquel on peut accéder via le menu en haut de chacune des autres pages du site, mais seulement si une session possédant les droits administrateurs est en cours. Nous pouvons ensuite entrer les données de la recette dans les différents champs du formulaire.

Ces dernières seront ensuite envoyées à la page recipeController, qui se chargera d’organiser les données avant la prochaine étape.

On envoie ensuite ces données dans le bon Modèle et dans la bonne fonction qui va, quant à elle, s’occuper d’entrer correctement toutes ces données dans la base de données.

# Tests

Tout au long du processus de création du site, différents tests ont été effectués.

# Conclusion

## Conclusion générale

Pour conclure, ce projet a été un de ceux qui nous aura procuré le plus de problèmes, mais également un de ceux qui nous aura appris le plus de choses et qui nous sera surement le plus important dans notre carrière. Le manque de connaissances et de temps auront été nos principaux soucis. Manque de connaissances surtout car beaucoup de facettes de ce projet nous étaient encore inconnues au début de celui-ci. Comment faire une bonne base de données qui soit utilisable mais aussi efficace ? Comment intégrer cette base de données à notre site web ? Comment y insérer des données depuis le site ? Comment structurer le projet en MVC ? Comment utiliser efficacement GitKraken afin de faciliter la mise en commun de notre travail ? Nous n’aurions certainement pas su répondre à toutes ces questions avant ce projet. Malgré les difficultés rencontrées, dans le groupe ou individuellement, nous avons aujourd’hui un site fonctionnel et qui nous plait. Ce projet a également servi à mettre en avant les lacunes de chacun de nous deux et nous permettra de connaitre nos points forts et faibles, et de consolider les sujets dans lesquels nous ne sommes pas à l’aise. Nous aurons également appris à créer un site dynamique utilisant une base de données et ce savoir pourra être utilisé lors de projets personnels ou professionnels dans nos futurs respectifs. Ces connaissances sont également des connaissances au niveau organisationnel de notre travail (Gestion du temps, utilisation d’outils comme GitKraken, etc.).

## Conclusions Personnelles

### Vitor Silva

Vitor…

### Ricardo Delgado Miranda

Dès le début, ce projet m’a beaucoup intéressé et motivé. J’ai toujours été plus intéressé par la programmation web que par la programmation c#, par exemple, et dans ce projet j’ai eu l’impression de construire un site qui pourrait réellement être utilisé. J’ai appris énormément de choses qui me seront surement très utiles dans ma vie. J’ai été un peu perdu au moment de passer le site en MVC mais Vitor qui avait bien mieux compris que moi à pu m’aider à comprendre un peu mieux cette structure. J’ai en tout cas appris énormément de choses au long de ce projet et j’ai hâte de pouvoir intégrer tout cela dans des projets personnels ou professionnels.

## Méthode de projet

Notre méthode de travail n’a pas tout de suite été la meilleure. Pour ce qui est de la répartition du travail, nous avons tout de suite réussi à faire les choses assez bien. Le travail était correctement réparti et le projet avançait bien. Le problème est surtout dans le fait que nous avons utilisé initialement Google Drive pour faire les échanges de fichiers et ça rendait les mises en commun assez compliquées à certains moments. Plus tard, nous avons appris à utiliser GitKraken et l’avons incorporé au développement du site et ça a beaucoup facilité les choses. Cependant, notre faible maitrise de l’application nous a posé quelques problèmes au moment des *push* sur le *main*. Nous avons dû gérer quelques conflits dans certains fichiers qui nous auront fait perdre un peu de temps mais ce n’est rien par rapport au gain de temps que nous a apporté GitKraken finalement. Nous avons également pas mal manqué de temps pour ce projet. En effet une partie des éléments n’étaient pas finis avant les vacances et nous avons dû nous organiser pour travailler à distance et finir le projet dans les délais impartis. Pour ce qui est de la communication au sein du groupe, tout s’est toujours plutôt bien passé. Nous avons déjà travaillé ensemble dans pas mal de projets ce qui fait qu’on connait la méthode de travail de l’autre, ce qui fluidifie les échanges et le travail.

# Webographie & annexes

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

Regex101: <https://regex101.com/>

Bootstrap: <https://getbootstrap.com/>

Font Awesome: <https://fontawesome.com/>

Section INF : <https://enseignement.section-inf.ch/>