



ÉCOLE NATIONALE D'INFORMATIQUE POUR L'INDUSTRIE ET
L'ENTREPRISE

Projet de Programmation Web
Site de Gestion d'Associations :
Apéral - Oenologiie

Auteurs :

Dorian LAUGIER
Simon LAUZERAL
Paul THIBAUD
Thomas GUBENO

22 mai 2018

Table des matières

1	Présentation du projet	1
1.1	Choix du Sujet	1
1.2	Lancement du projet	1
1.3	Répartition des tâches	2
2	Développement	3
2.1	Outils utilisés	3
2.1.1	Docker	3
2.1.2	GitLab	3
2.2	Contenu du site	4
2.3	Base de données	4
2.4	Difficultés rencontrées	5

Chapitre 1

Présentation du projet

L'objectif du projet est de développer un site web parmi plusieurs sujets proposés :

- Agenda des associations
- Accueil des nouveaux arrivants
- Twitter de l'ENSIIE
- Sujet Libre

Cependant, quel que soit le sujet choisi, le site web doit comporter :

- Un système d'authentification
- Un compte administrateur donnant les droits à certaines fonctionnalités (au choix)
- Un profil utilisateur éditable
- Une base de données relationnelle composée de :
 - Au moins 3 tables
 - Au moins une table de jointure ($n \dots n$)
 - Au moins une jointure dans une requête SQL
- Des requêtes SQL composées d'INSERT, DELETE, UPDATE, SELECT
- Un CRUD (Create Read Update Delete)

1.1 Choix du Sujet

Nous avons choisi le sujet libre afin de développer un site web nous permettant de gérer nos associations à l'école, à savoir Oenologie et Apéral.

En effet, notre groupe étant composé du président d'Oenologie (Thomas Gubeno), du vice-président et du trésorier d'Apéral (Dorian Laugier et Simon Lauzeral), et ces deux associations étant très proches dans leur fonctionnement, l'idée de réaliser un site de gestion d'association nous semblait cohérente.

De plus, ce site pourrait à terme remplacer la gestion de réunions pour Apéral et Oenologie actuelle, qui passe par une conversation de groupe sur Facebook, et manque de fonctionnalités.

Le site web devait permettre aux membres du bureau de ces deux associations de gérer les membres, les réunions, les préparatifs et recettes, et aux utilisateurs lambda d'avoir accès à toutes ces informations, en plus d'informations globales sur les associations.

1.2 Lancement du projet

La première séance de Projet Web a avant tout servi à se familiariser avec l'environnement de travail et à définir le travail à réaliser.

Pour ce faire, nous avons créé un dépôt git, et essayé de créer un conteneur pour notre site avec Docker. La marche à suivre pour créer ce conteneur a ensuite été expliquée dans un readme sur notre dépôt pour faciliter l'installation pour tout le monde :

Installation et config Docker

- Installer docker à coup de `sudo apt-get install docker et docker.io`
- Installer docker-compose en écrivant `sudo curl -L https://github.com/docker/compose/releases/download/1.21.0/docker-compose-$(uname -s)-$(uname -m) -o /usr/local/bin/docker-compose` (Installer curl si il n'est pas installé) puis `sudo chmod +x /usr/local/bin/docker-compose` (cf. ce lien : <https://docs.docker.com/compose/install/#install-compose>)
- Lancer `cp .env.dist .env` pour créer un fichier .env propre à l'utilisateur
- Changer les paramètres dans le fichier .env (vim .env dans le repo) : le DOCKER_USER (prenom.nom) et le DOCKER_USER_ID (pour recup celui la taper `$(echo id -u $USER)` dans le terminal)
- Lancer `make install` dans le repo pour installer docker sur le site
- Le site sera up sur <http://localhost:8080> et modifié en temps réel avec vos modifs
- Ne pas oublier de add, commit et push après modifications (et de pull avant)

FIGURE 1.1 – Extrait du ReadMe

Durant cette séance, nous avons également défini le squelette du site Web, à savoir toutes ses pages et leur structure, pour nous fixer un cadre de travail et faciliter le développement par la suite, en évitant de se perdre dans des pages inutiles.

Les différentes pages, après discussion sur leur pertinence avec les membres du groupe, ont été modélisées de manières très simples à l'aide de Gimp pour avoir une idée de leur aspect :

Préparatifs

Liste de courses		
Saucisson	Mardi 25 mars 2018	<input type="checkbox"/> Participer
Chips	18h00	
Tomate		
Mozarella		

Préparation	Adresse
Ecole	
Thomas GUBENO	Paul THIBAUD
Simon LAUZERAL	Dorian LAUGIER
<input type="checkbox"/>	<input type="checkbox"/>

FIGURE 1.2 – Exemple de page : La page préparatifs

1.3 Répartition des tâches

Lors de la première séance, nous en avons également profité pour répartir dans les grandes lignes les différentes tâches à réaliser.

Ainsi, Paul Thibaud devait s'occuper de créer les différentes pages Web et leur contenu, Dorian Laugier s'occupait quant à lui du CSS propre au site, Simon Lauzeral réalisait les différentes fonctions PHP et JavaScript utilisés dans le site, et Thomas Gubeno s'occupait de toute la base de données, et notamment du système d'authentification.

Par la suite, certaines tâches ont été réattribuées : Thomas Gubeno s'est occupé du CSS pour les menus sur les pages, Simon Lauzeral a géré la base de données, et Dorian Laugier a créé le système d'authentification.

Chapitre 2

Développement

Dans cette partie seront explicités les choix que nous avons décidé de prendre pour répondre aux différentes tâches du projet, comment nous les avons implémentés et quels obstacles nous avons rencontré.

2.1 Outils utilisés

Afin de pouvoir travailler efficacement en équipe, différents outils de gestion de projet ont été mis à notre disposition par les enseignants.

2.1.1 Docker

Docker est un outil logiciel permettant d'automatiser le déploiement d'applications (ici notre site web) dans des conteneurs s'appuyant sur les fonctionnalités du noyau **Linux** afin de pouvoir exécuter l'application sur n'importe quel serveur.

Dans notre cas, nous avons utilisé Docker pour héberger localement notre site web sur l'url `http://localhost:8080`, ce qui nous permettait de visualiser les modifications apportées au site en direct.

L'utilisation de Docker était grandement facilitée par le MakeFile fourni par la **Team Matters**, nous permettant de lancer `make install` pour déployer notre site web dans un conteneur puis `make start` pour lancer l'application.

2.1.2 GitLab

GitLab est une plate-forme permettant d'héberger des projets, offrant la possibilité de gérer des dépôts git et ainsi de mieux appréhender la gestion des différents codes sources composant le projet final.

Le fonctionnement du dépôt est le suivant : Le projet est pushé sur Gitlab, puis récupéré par les contributeurs à l'aide d'un pull, qui leur permet d'accéder et de modifier les différents fichiers depuis leur machine. Une fois les modifications effectuées, les utilisateurs peuvent "commit" leurs fichiers sources modifiés et les push afin de compléter le projet final.

Le projet est alors présenté sous la forme de différents dossiers et fichiers accessibles à tous, avec des informations sur les derniers changements (cf. figure suivante) et un accès à un historique retraçant tous les commits effectués.

master

▼

Projet_Web /

+

▼


History

Q Find file

Web IDE

↻


▼



Ajout table participants

Thomas GUBENO authored about a minute ago

c33b077c

















Name	Last commit	Last update
 .docker	first commit	2 months ago
 data	Ajout table participants	about a minute ago
 document	ajout des sujets	a month ago
 public	petit fixe sur la page reunion	about 16 hours ago
 src/User	Modif utilisateur	2 minutes ago
 style	oui	3 weeks ago
 test/unit/User	Modif du README	3 weeks ago
 .env.dist	remove .ne	3 weeks ago
 .gitignore	ignore	3 weeks ago
 Makefile	Modif du README	3 weeks ago
 README.md	Readme	3 weeks ago
 composer.json	first commit	2 months ago
 docker-compose.yml	first commit	2 months ago
 phpunit.xml	Modif du README	3 weeks ago

FIGURE 2.1 – Interface de GitLab

2.2 Contenu du site

Afin de répondre aux besoins de nos deux associations, le site web doit proposer plusieurs fonctionnalités dont la liste exhaustive est la suivante :

- Permettre de gérer et d'accéder aux réunions d'Apéral et d'Oenologie ayant eu lieu, à leur compte-rendu et à leurs participants, en particulier à la dernière réunion en date, correspondant à la prochaine soirée organisée par l'école.
- Permettre de gérer (organiser et participer) les courses de pré-soirée, nécessaires au bon déroulement des préparations organisées.
- Proposer aux visiteurs une liste des différentes recettes d'Apéral et des vins recommandés par Oenologie, et leur permettre de noter les recettes et vins afin de contribuer aux associations.
- Fournir un classement traduisant l'implication dans les associations : pour gagner des points dans le classement, les utilisateurs devront participer aux réunions et / ou aux courses.
- Accéder à un profil utilisateur éditabile, comme pour la majorité des sites web.
- Posséder un espace administrateur, permettant aux membres du bureau des associations d'ajouter une réunion, une recette, un vin ou une liste de courses.

Initialement, nous avons également comme idée de gérer la trésorerie des associations depuis le site-web, cependant cette idée fut abandonnée par la suite étant donné le peu de flux monétaires circulant au sein de ces associations.

2.3 Base de données

Afin de gérer les différentes interactions, le site web est accompagné d'une base de données gérant les différents éléments enregistrés.

Cette base de donnée est composée des tables suivantes :

- Une table des utilisateurs, remplie à l'inscription, composée du nom complet de l'utilisateur, son pseudo

utilisé à l'école, son adresse mail, son mot de passe, et un argument relatif à ses droits (administrateur ou simple membre de l'association).

- Une table réunion, donnant accès à l'ensemble des réunions ayant eu lieu.
- Deux tables permettant de récupérer les recettes et leur note attribuée (idem pour les vins).
- Deux tables composée des utilisateurs participant aux réunions, et de ceux participant aux courses (Cette table aurait d'ailleurs pu faire l'objet d'une jointure avec la table utilisateur, car on utilise les pseudos des participants).

L'accès à ces tables de la base de données se fait par le biais de requêtes au sein du code HTML / PHP, et nous permet d'accéder aux différents éléments dont nous avons besoin.

2.4 Difficultés rencontrées

Pendant le développement de ce projet, nous avons fait face à de nombreuses difficultés qui n'ont pas rendu le développement facile. Les difficultés majeures que nous avons rencontrées sont les suivantes :

Lors de l'explication du principe de Docker par la **Team Matters**, nous avons eu du mal à comprendre le fonctionnement de ce logiciel.

Ainsi, nous avons consacré beaucoup de temps à l'installation et la compréhension de Docker : nous n'avions toujours pas réussi à installer le logiciel correctement à la fin de la première séance, ce qui nous a complètement empêché de commencer à développer le site web.

La résolution de nombreux problèmes relatifs à la base de données a été extrêmement chronophage, ce qui est notamment dû au fait que nous ne savions pas d'où venait le problème (Notamment pour implémenter la participation aux réunions et aux courses).

Il y a également eu un problème au niveau de la mise en commun du code, en effet certaines variables et attributs de la base de données avaient le même nom, ce qui posait un problème de compréhension.

Nous avons décidé trop tardivement d'implémenter la jointure pour relier les bases de données de réunion et d'utilisateurs (comme expliqué dans la partie "**Base de données**"), ce qui nous a empêché de l'implémenter correctement, et nous avons finalement décidé d'abandonner cette idée, nous privant alors d'une jointure qui était nécessaire dans l'énoncé.