

Projet De Groupe

Cahier des charges – EasyGoing!

A l'attention de Dr. R. Rentsch

Thibaud Duchoud
Karim Ghozlani
Michelle Vanessa Meguep
Raphaël Racine
Miguel Santamaria



2015

TABLE DES MATIERES

Pairs.....	2
Cadre/Contexte	2
Description du projet	2
Généralités	2
Fonctionnalités	3
Liste des fonctionnalités	3
Tutoriel d'utilisation.....	4
Fil d'actualités	5
Base de données	5
Maquette ui.....	7
Sécurité.....	7
Hachage du mot de passe utilisateur.....	7
Sécurités diverses.....	7
Utilisation d'un framework.....	8
Pourquoi utiliser un Framework ?.....	8
Pourquoi Zend Framework ?.....	8
Technologies utilisées.....	8
Extensions.....	8
Planification	9

PAIRS

Miguel Santamaria (Chef de groupe)
Thibaud Duchoud
Karim Ghoslani
Michelle Vanessa Meguep
Raphaël Racine
René Rentsch (Professeur)

CADRE/CONTEXTE

Ce projet est réalisé dans le cadre du cours de PDG 2015, en 3^{ème} année de la filière TIC-IL de l'HEIG-VD à Yverdon-Les-Bains.

Après avoir rapidement présenté l'idée du projet au professeur, nous sommes dans la phase de l'élaboration du cahier des charges qui suit.

DESCRIPTION DU PROJET

GENERALITES

Notre projet « *EasyGoing!* » consiste à un outil collaboratif de gestion de projet. Cet outil est spécialement adapté au développement logiciel et sera réalisé sous forme d'un site web. Pour se distinguer des applications déjà existantes, il met un fort accent sur l'aspect ergonomique, ainsi que sur diverses fonctionnalités sociales : il offre en effet des outils d'accès à des fils d'actualités et à des historiques détaillés, ainsi qu'une utilisation intuitive à base de cliquer-glisser et un tutoriel d'utilisation simple et efficace pour les nouveaux utilisateurs.

Cet outil permettra aux membres d'une équipe d'organiser leur travail autour de boards (tableau de bord). Un board est donc une visualisation d'un certain nombre de tâches. L'aspect collaboratif réside dans le partage en temps réel des boards et l'espace de discussion mis en place autour de ces derniers.

Il s'agit donc d'une application web tournant sur un serveur Apache et possédant une base de données, le tout s'exécutant de manière sûr et sécurisé. A noter que nous avons finalement opté pour un site web pour plusieurs raisons, parmi lesquelles :









- L'ergonomie en sera grandement améliorée.
- La portabilité et la facilité d'accès nous permettra de toucher plus de personnes.
- La compatibilité avec tous les systèmes, qu'ils soient mobiles ou non.
- Les éventuelles futures mises à jour, ainsi que la maintenance en général seront facilitées.
- Il s'agit d'un challenge, car la plupart des membres de ce projet se considèrent comme novices dans les technologies web.

FONCTIONNALITES

LISTE DES FONCTIONNALITES

Voici une liste des fonctionnalités qui caractérisent cette application (certaines nécessitant des paragraphes complets plus loin) :

- Inscription / authentification.
- Les membres possèdent les attributs suivants : nom d'utilisateur, nom, prénom, mot de passe, adresse e-mail et photo. Lorsqu'ils sont ajoutés à un projet, ils ont également une spécialisation associée au projet (expert en base de données, en design, etc.).
- Lors de l'inscription, l'utilisateur peut choisir s'il désire afficher un tutoriel d'utilisation ou non. Cette partie est détaillée plus bas.
- Les membres peuvent être administrateurs d'un projet.
- Un membre peut créer un projet et en devient directement administrateur.
- L'administrateur a des privilèges : il peut ajouter et supprimer des membres, modifier et supprimer le projet. Il peut également assigner une spécialisation à un utilisateur.
- Un projet est constitué d'une description, d'un logo et d'une durée (date de début + date de fin), d'une liste de membres comprenant au moins un administrateur, d'un historique, ainsi que d'un board dont la vue est personnalisable.
- Le board comprendra en effet 2 vues principales : une vue sur les tâches et une sur les membres du projet auquel il est associé.

Tasks	Not started	In progress	Complete
A			
B			
C	C2 	C3  C1	
D	D1 		D2  D3 
E	E1 		

Members	Not started	In progress	Complete
Miguel	C1		
Karim	C3 D2		A
Raphael	C2		
Thibaud		E1	
Vanessa			D1 B

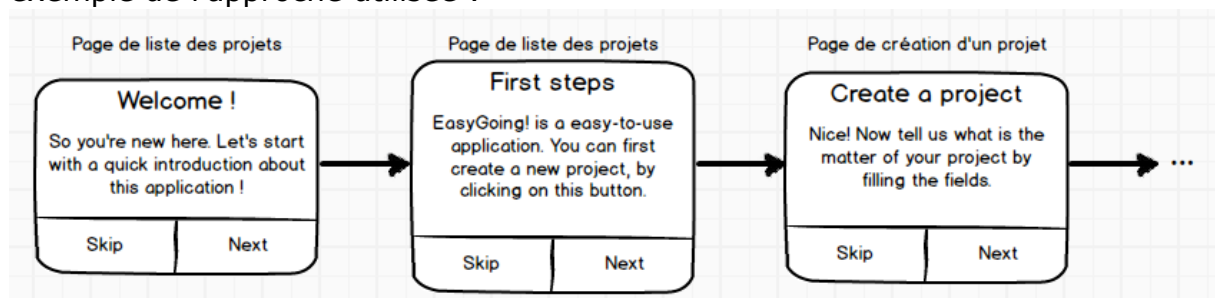
- L'historique est automatique et affiche les différentes actions que les membres du projet ont réalisé (par exemple : « <10.10.15> Miguel created the T1 task. », etc.). Il permet notamment aux membres de suivre le déroulement du projet.

- Un board comprend un ensemble de tâches. Ces tâches seront dans une « liste d'attente », avant qu'un des membres du projet les assigne à quelqu'un (un membre peut assigner une tâche à lui-même). Une fois assignées, les tâches sont affichées sur une des 4 colonnes du board (not started, In progress, Code review, Done) dépendamment de la vue choisie.
- Les membres peuvent créer des tâches, les modifier, les assigner à d'autres utilisateurs.
- Les tâches sont constituées d'une description, d'une liste de personnes assignées (qui sera vide au départ), d'un niveau de priorité, d'une durée et d'un possible ensemble de sous-tâches (un niveau maximum).
- Les sous-tâches sont liées à une seule tâche.
- Chaque tâche/sous-tâche possède un fil d'actualités qui lui est propre. Cette fonctionnalité est décrite plus en détails plus bas.
- Modification des tâches : modification d'attribut (description, priorité, membres assignés), ajouter des commentaires, attacher des liens.
- Lors de différentes actions dans notre application, nous allons également envoyer des mails (inscription, mot de passe perdu ...) et, pour les notifications de projet, ceux-ci pourront être désactivés si souhaité.

TUTORIEL D'UTILISATION

Lorsqu'un nouvel utilisateur s'inscrit, il a le choix d'accepter ou non un tutoriel d'utilisation, via une case à cocher (cochée par défaut). Celui-ci se présente sous la forme de bulles d'aide apparaissant à l'écran lorsque l'utilisateur accède pour la première fois à une page. Ce système se doit d'être le moins intrusif possible, et l'utilisateur aura le choix de le désactiver à tout moment.

Cependant, grâce à cette fonctionnalité, ainsi qu'à l'ergonomie de l'application, il comprendra facilement et très rapidement ce qu'il est censé faire. Voici un petit exemple de l'approche utilisée :



FIL D'ACTUALITÉS

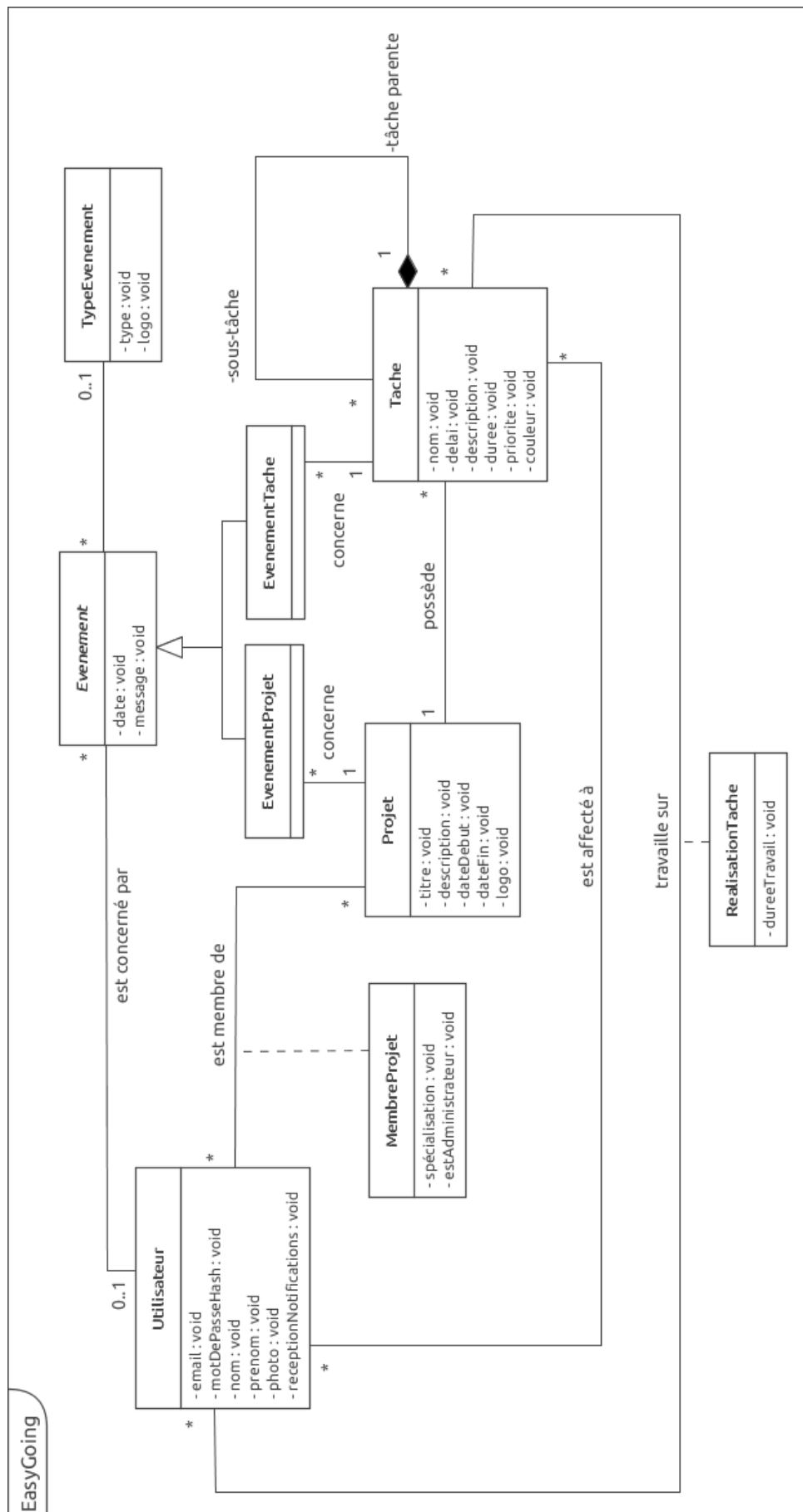
Chaque tâche/sous-tâche contient un fil d'actualités, rempli par les membres de cette tâche au fur et à mesure de l'avancement du projet. Chaque membre a la possibilité d'écrire des statuts possédant un tag (bug, help, information, warning, etc.). Cela permet aux membres de suivre l'avancement de la tâche et de communiquer diverses informations aux autres.

BUG	<10.10.15>	When you enter the '4' value, a
INFO	<11.10.15>	Please add some comments to
WARNING	<11.10.15>	The task must end

BASE DE DONNÉES

Nous utiliserons une base de données de type MySQL avec PHPMyAdmin pour la persistance des données, car il s'agit d'un SGBD qui correspond bien à ce qui est attendu du projet et qui est assez simple à prendre en main.

Elle sera constituée des éléments suivants, sous réserve de modifications si nécessaire :



Il est à noter que ce schéma est simplement un modèle de domaine et que d'autres tables seront ajoutées si nécessaire au fur et à mesure de l'avancement du projet.

Contraintes d'intégrité

Voici quelques contraintes d'intégrité de base (également sous réserve de modifications) :

- Membre projet : Un utilisateur ne peut pas être 2 fois membre d'un même projet, et il doit y avoir au moins un administrateur par projet
- Affectation de tâche : Un utilisateur ne peut pas être affecté à une tâche s'il n'est pas membre du projet dans lequel la tâche est inscrite
- Réalisation de tâche : Un utilisateur ne peut pas réaliser une tâche s'il n'y est pas affecté
- Sous tâche : Une tâche qui a déjà une tâche parente ne peut pas avoir de sous-tâche. Autrement dit, on s'arrête à un seul niveau de sous-tâche

Bien entendu, d'autres contraintes seront ajoutées lors de la phase de modélisation...

MAQUETTE UI

Une ébauche de ce à quoi ressemblera l'interface de l'application est jointe en annexe. Certains détails peuvent être sujets à des changements, mais dans l'ensemble le résultat final s'inspirera grandement de cette maquette.

A noter qu'il s'agit d'un modèle interactif sujet à modification : il est possible de cliquer sur certains boutons.

SÉCURITÉ

HACHAGE DU MOT DE PASSE UTILISATEUR

Le mot de passe utilisateur sera stocké dans la base de données sous forme de hash. Nous allons utiliser sha256 comme fonction de hachage.

SÉCURITÉS DIVERSES

Nous allons également utiliser les composants de Zend Framework pour gérer les différents aspects de la sécurité web :

- faille XSS
- injection SQ
- ...

Ainsi que les différents filtres et validateurs pour les formulaires web et qui seront appliqués sur les champs envoyés via les requêtes GET et POST.

UTILISATION D'UN FRAMEWORK

POURQUOI UTILISER UN FRAMEWORK ?

Nous avons décidé d'utiliser un framework car il nous apporte de nombreux avantages :

- avoir une application structurée et sécurisée.
- augmenter la productivité une fois le framework maîtrisé.
- améliorer la pérennité et faciliter la maintenance de l'application.

POURQUOI ZEND FRAMEWORK ?

Nous avons évidemment regardé différents framework php existant (laravel, symphony, cakePHP) mais sans expérience dessus, il est difficile de se faire une idée précise de l'efficacité d'un de ces framework dans notre projet.

Car l'un d'entre nous a déjà utilisé ce framework dans le cadre professionnel et tout c'est bien déroulé. Nous utilisons donc un framework dont on sait qu'il fonctionne et qui sera utile dans le cadre de notre projet.

TECHNOLOGIES UTILISÉES

Nous avons décidé de créer notre projet sous forme de site web et donc d'utiliser les technologies en découlant :

- **Apache v2.4** comme serveur http.
- **PHP v5.6** comme langage coté serveur.
 - o **Zend Framework** comme framework php.
- **MySQL v5.6** comme moteur de base de données.
 - o **PHPMyAdmin v4.4 et MySQL Workbench v6.3** pour gérer la base de données.
- **HTML – CSS** comme langages de structure et design des pages web.
- **Javascript/JQuery** comme langage/bibliothèque pour dynamiser les pages web.
- **Différentes bibliothèques Javascript** (comme Bootstrap) pour réaliser des actions spécifiques concernant la gestion des boards principalement (cliquer-déposer, animation, gestion d'une grille ...).

EXTENSIONS

Nous avons réfléchi à beaucoup d'idées moins prioritaires, que nous développerons si le temps nous le permet. Voici quelques-unes d'entre elles :

- Proposer une rubrique d'aide détaillée, comprenant des vidéos d'utilisation du projet.
- Pouvoir se partager des fichiers entre membres d'un même projet.
- Pouvoir personnaliser entièrement son board.
- Pouvoir personnaliser les tâches (mettre du texte en gras, italique ...).

- Améliorer l'interface du site web, afin de le rendre éventuellement plus intuitif sur les smartphones (responsive).
- Intégrer de la vidéo conférence dans un projet.
- Envoyer des messages privés.
- Améliorer la recherche des membres (cercles à Google+, origine, ...).

PLANIFICATION

La planification est jointe en annexe.

A noter que le développement sera structuré en modules (module utilisateur, module liste des projets, module projet, module aide/about).

Annexes :

- Maquette UI
- Planification