Furnigo Documentation Technique

**CFPT**

**Centre de Formation Professionnelle Technique Route du Pont-Butin 43 CP 548 1213 Petit-Lancy**

**078 741 12 14**

**I.IN-P4B**

**2016-2017**

Maurice Dinh

Document technique de développement du site web Furnigo

Table des matières

[Introduction 3](#_Toc483407530)

[Description 3](#_Toc483407531)

[Pourquoi ce sujet ? 3](#_Toc483407532)

[Le PHP 3](#_Toc483407533)

[Etude d’opportunité 4](#_Toc483407534)

[Rappel du cahier des charges 6](#_Toc483407535)

[But 6](#_Toc483407536)

[Fonctionnalités 6](#_Toc483407537)

[Matériel et logiciels nécessaires 6](#_Toc483407538)

[Travail à rendre 6](#_Toc483407539)

[Analyse fonctionnelle 7](#_Toc483407540)

[Fonctionnalités des utilisateurs 7](#_Toc483407541)

[Fonctionnalités de l’utilisateur anonyme (non connecté): 7](#_Toc483407542)

[Fonctionnalités d’un utilisateur connecté : 7](#_Toc483407543)

[Fonctionnalités de l’administrateur : 7](#_Toc483407544)

[Maquettes de l’interface visuelle 8](#_Toc483407545)

[Analyse organique 12](#_Toc483407546)

[Manipulation de devis 12](#_Toc483407547)

[Création de devis 12](#_Toc483407548)

[Suppression de devis 12](#_Toc483407549)

[Modification d’un devis 13](#_Toc483407550)

[Affichage des devis 13](#_Toc483407551)

[Manipulation des options 13](#_Toc483407552)

[Création d’une option 13](#_Toc483407553)

[Affichage d’option 14](#_Toc483407554)

[Modification d’option 14](#_Toc483407555)

[Suppression d’une option 15](#_Toc483407556)

[Architecture du site 15](#_Toc483407557)

[Description des fonctions 16](#_Toc483407558)

[Planning finale 0](#_Toc483407559)

[0](#_Toc483407560)

[Tests et protocole de tests 0](#_Toc483407561)

[Conclusion 3](#_Toc483407562)

[Amélioration possible 3](#_Toc483407563)

[Remerciement 3](#_Toc483407564)

[Figure 1 Site web Balestrafic en responsive 4](#_Toc483403016)

[Figure 2 Interface du calculateur de pré-devis du site Balestrafic 5](#_Toc483403017)

[Figure 3 Page d'accueil du site L&L Déménagement 5](#_Toc483403018)

[Figure 4 Interface du devis en ligne de L&L Déménagement 6](#_Toc483403019)

[Figure 5 Maquette de la page d'accueil (index) 8](#_Toc483403020)

[Figure 6 Maquette de la page de connexion 8](#_Toc483403021)

[Figure 7 Maquette de la page d'inscription 9](#_Toc483403022)

[Figure 8 Maquette de l'interface de gestion de devis 9](#_Toc483403023)

[Figure 9 Maquette de l'interface du calculateur de devis 10](#_Toc483403024)

[Figure 10 Maquette de l'interface de gestion d'utilisateurs 10](#_Toc483403025)

[Figure 11 Maquette de l'interface de gestion d'option 11](#_Toc483403026)

[Figure 12 Maquette de l'interface de modification d'option 11](#_Toc483403027)

[Figure 13 Schéma logique de la création d'un devis 12](#_Toc483403028)

[Figure 14 Schéma logique de la suppression d'un devis 13](#_Toc483403029)

[Figure 15 Schéma logique de la modification d'un devis 13](#_Toc483403030)

[Figure 16 Schéma logique de l'affichage des devis 13](#_Toc483403031)

[Figure 17 Schéma logique de la création d'une option 14](#_Toc483403032)

[Figure 18 Schéma de la logique d'affichage des options 14](#_Toc483403033)

[Figure 19 Schéma de la logique de la modification de données 14](#_Toc483403034)

[Figure 20 Schéma logique de la suppression d'une option 15](#_Toc483403035)

# Introduction

## Description

Dans le cadre de mon TPI 2017 du CFPT-I, j’ai choisi de réaliser un site internet de déménagement pour pouvoir appliquer mes connaissances en web .Ce document décrit les étapes de la réalisation de ce site.

De nos jours, le moyen le plus utilisé pour contacter une entreprise est à travers son portail web, c’est-à-dire son site internet. Le site doit offrir des fonctionnalités utiles à ses visiteurs. Mon site propose donc à ses utilisateurs la possibilité de pouvoir faire un devis gratuit en ligne pour avoir un aperçu du coût de son déménagement. Le plus gros du travail de ce projet est la mise en place du calculateur de devis.

## Pourquoi ce sujet ?

J’ai choisi ce sujet pour pouvoir appliquer mes connaissances en web plus précisément le PHP parce-que j’ai de la satisfaction à voir mon site fonctionner avec des fonctionnalités PHP et de pouvoir montrer mes connaissances acquises lors de ma formation pendant 4 ans. Le sujet à un deuxième intérêt, car il a un rapport avec le métier de développeur web que j’aimerais avoir plus tard et il pourra donc me servir comme projet de démonstration par la suite.

## Le PHP

Durant les 3 dernières années, nous avons travaillé le web avec du PHP. C’est pourquoi j’ai choisi de réaliser mon site était le PHP.

# Etude d’opportunité

Il existe des centaines de sites de déménagement, en voici un exemple :   
Balestrafic, c’est une entreprise de déménagement situé à Carouge.



Figure Site web Balestrafic en responsive

Leur site, est responsive. Mon site, lui, n’aura que le stricte minimum pour la partie esthétique, car, conformément à ce qui a été convenu avec les experts lors de la définition de ce travail, je me suis concentré sur les fonctionnalités et pas les aspects visuels. Leur site possède plusieurs pages présentant l’entreprise et les services disponibles. Mon site, au contraire n’en possèdera pas.

Par contre, leur site possède aussi un calculateur de devis tout comme celui que je vais réaliser

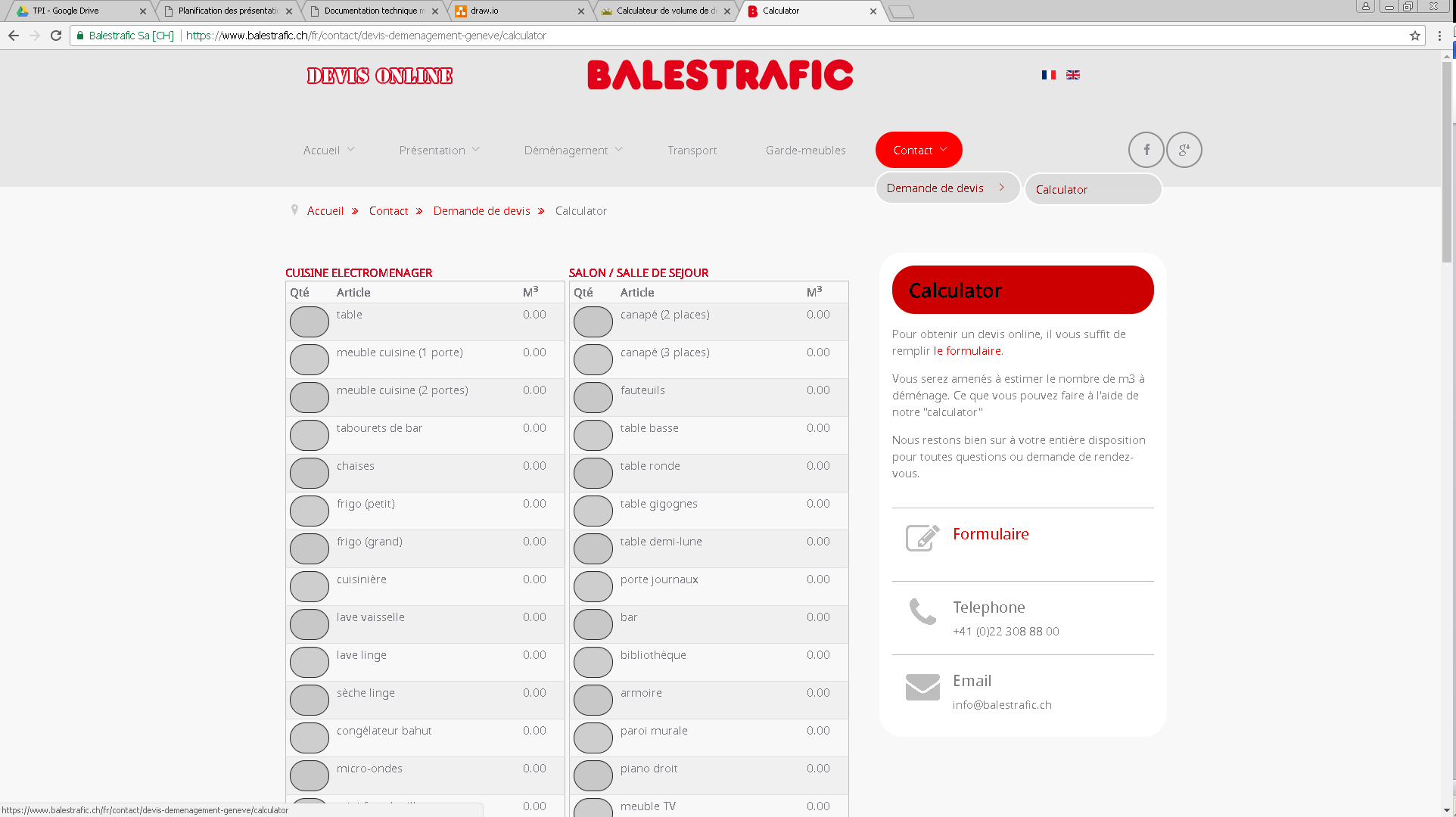


Figure Interface du calculateur de pré-devis du site Balestrafic

Un deuxième exemple serait l’entreprise L&L Déménagement S.A. C’est une entreprise de déménagement située à Genève. Ils proposent des options basiques de déménagement tel que la manutention, l’emballage et le stockage.

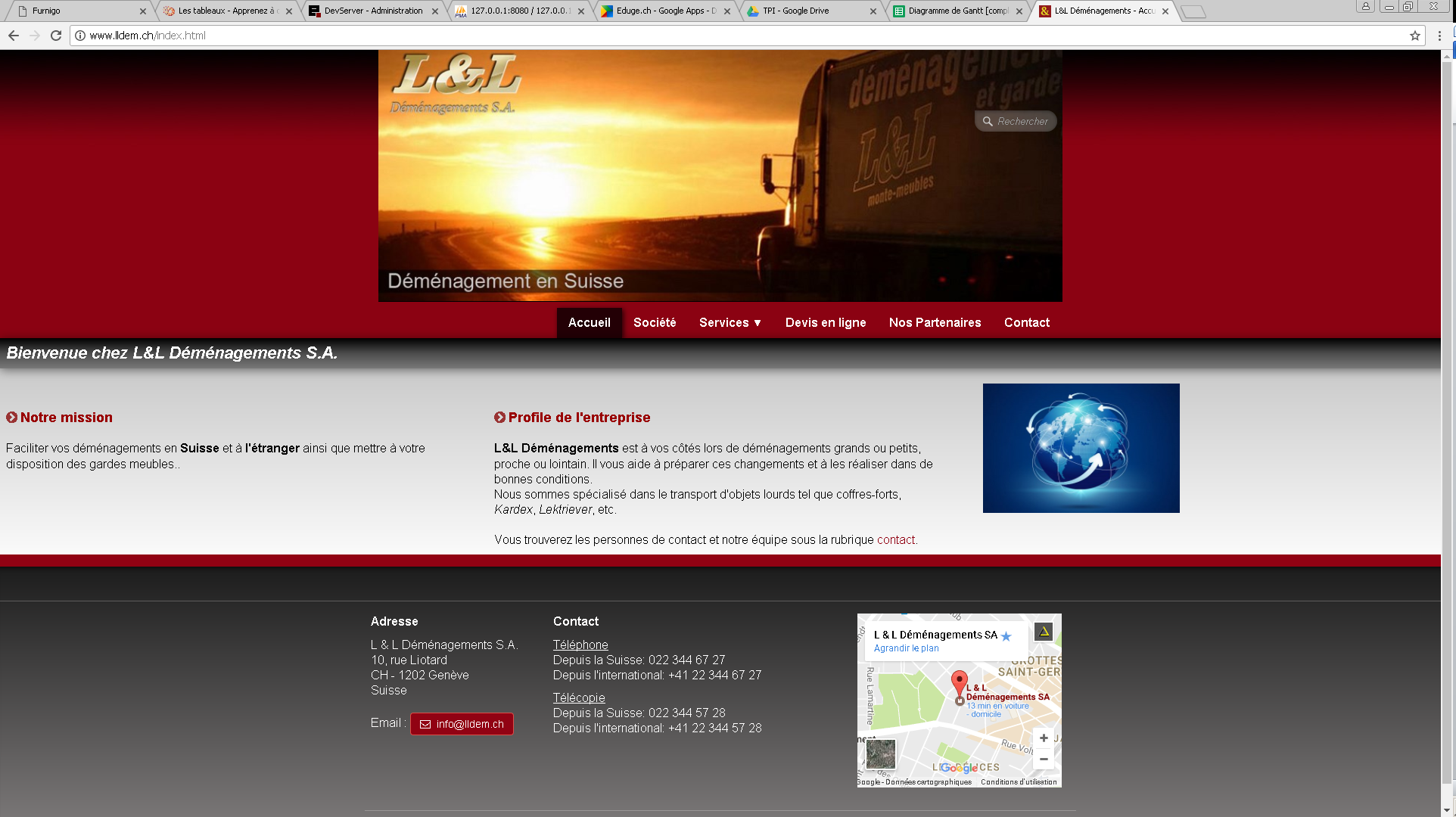


Figure Page d'accueil du site L&L Déménagement

Ils proposent également un devis en ligne mais moins complexe que celui que je vais réaliser.

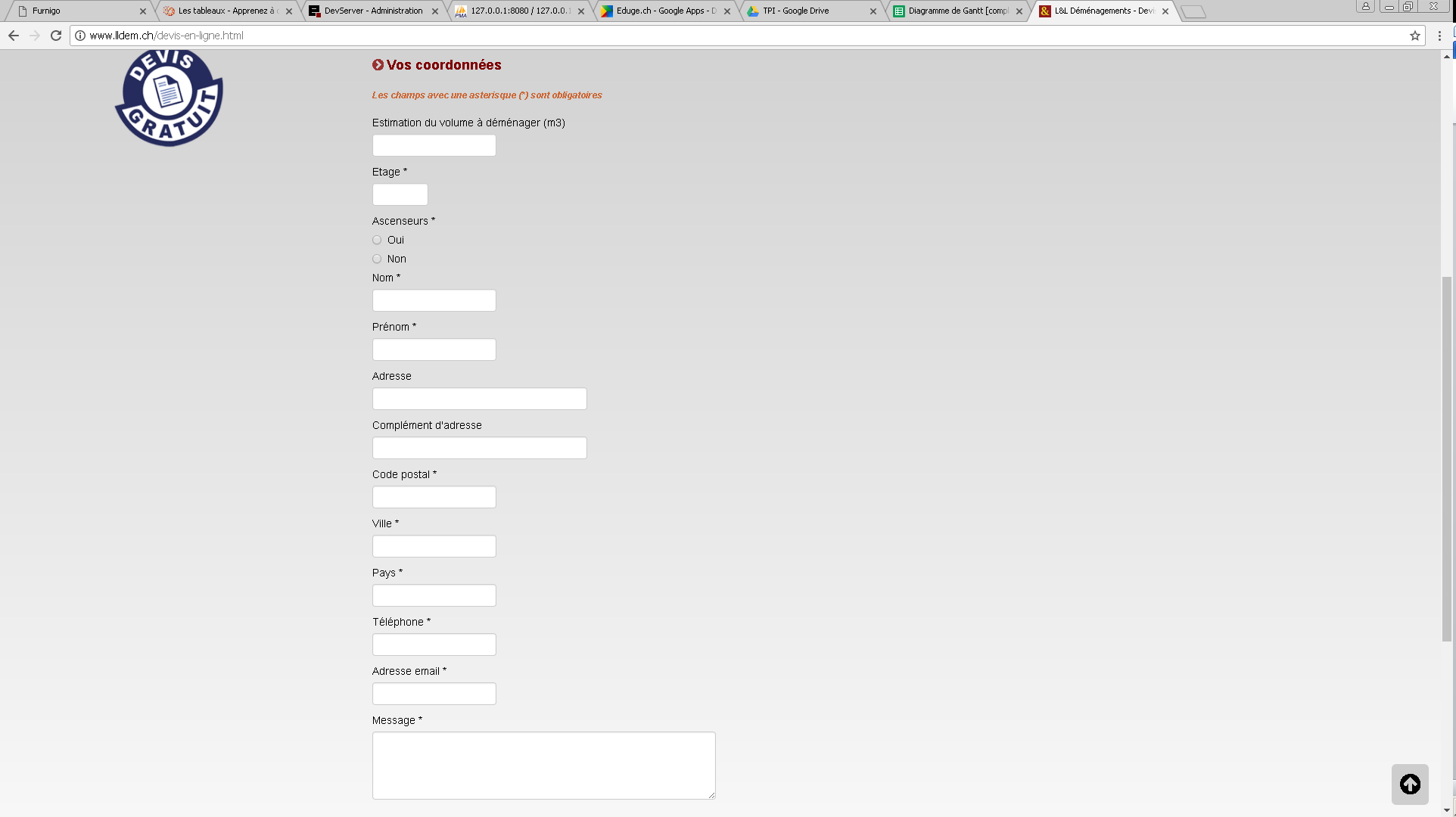


Figure Interface du devis en ligne de L&L Déménagement

# Rappel du cahier des charges

## But

Créer un site de déménagement fonctionnel permettant aux visiteurs d’utiliser un calculateur de devis.

## Fonctionnalités

Le calculateur de devis se base sur les informations des meubles, du lieu et propose des options additionnelles à partir de quoi il calcule le coût total du déménagement pour le client potentiel.

## Matériel et logiciels nécessaires

* Ordinateur
* NetBeans
* EasyPHP

## Travail à rendre

* Documentation Technique
* Manuel utilisateur
* Journal de bord
* Site web

# Analyse fonctionnelle

Cette partie définie les fonctionnalités du site web pour les différents profils d’utilisateurs : anonyme, connecté et administrateur.

## Fonctionnalités des utilisateurs

### Fonctionnalités de l’utilisateur anonyme (non connecté):

* S’inscrire
* Se connecter
* Calculer un devis
  + Sélectionner des options
  + Entrer les tailles des fournitures

### Fonctionnalités d’un utilisateur connecté :

* Se déconnecter
* Créer un pré-devis
* Modifier un pré-devis
* Modifier ses données personnelles

## Fonctionnalités de l’administrateur :

* Gérer les utilisateurs
* Éditer les options proposées

Attention ! L’administrateur ne peut pas éditer les pré-devis, cependant s’il supprime un membre du site, les devis du membre seront supprimés également de la base.

## Maquettes de l’interface visuelle

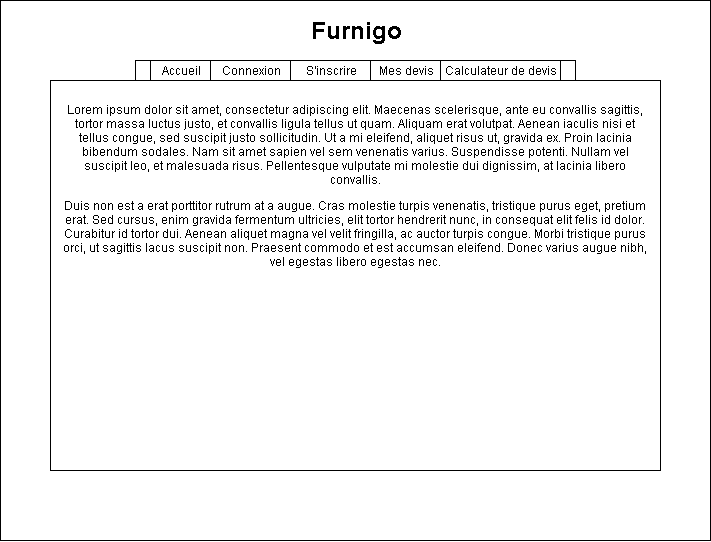


Figure Maquette de la page d'accueil (index)

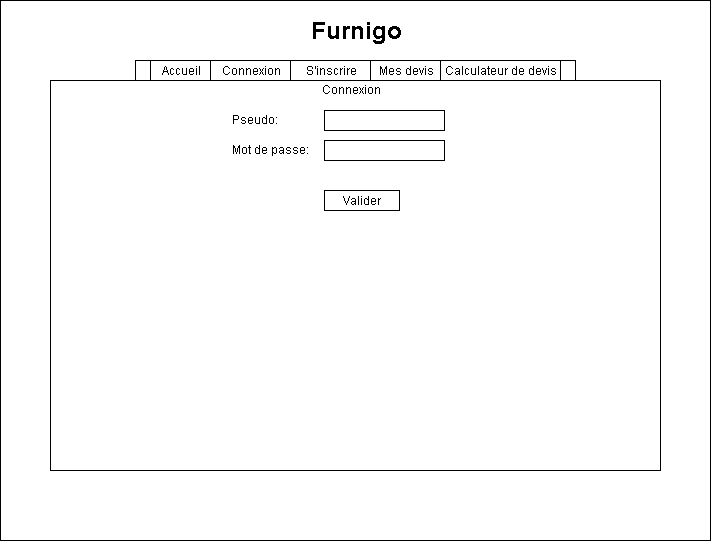


Figure Maquette de la page de connexion

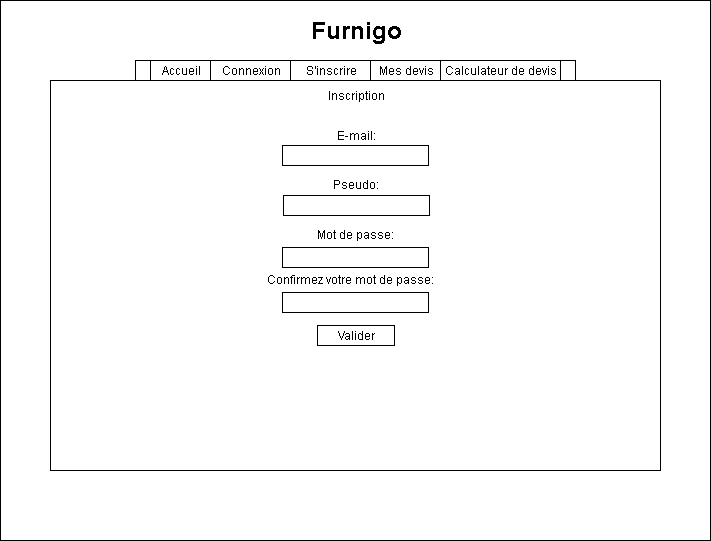


Figure Maquette de la page d'inscription

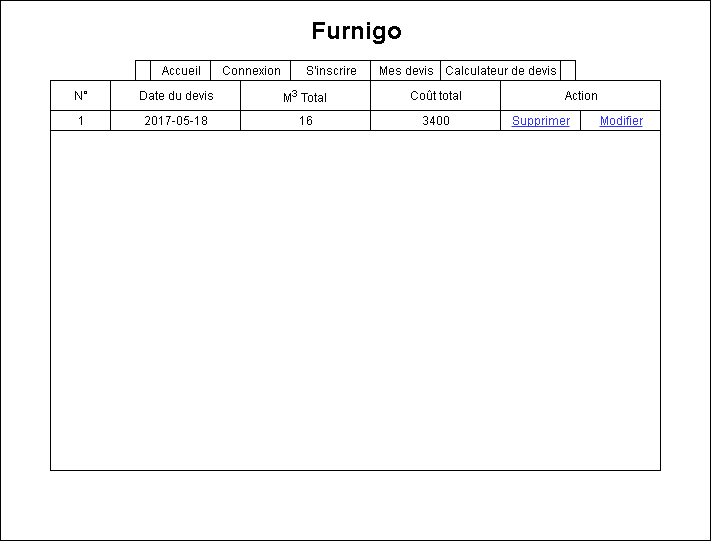


Figure Maquette de l'interface de gestion de devis

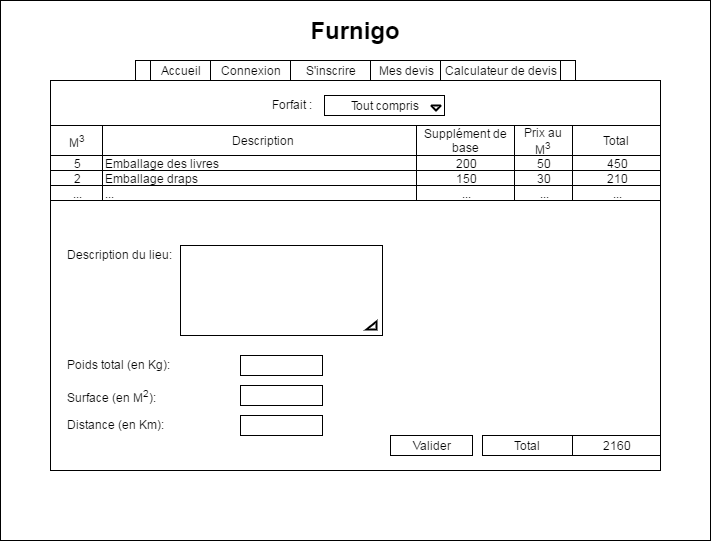


Figure Maquette de l'interface du calculateur de devis

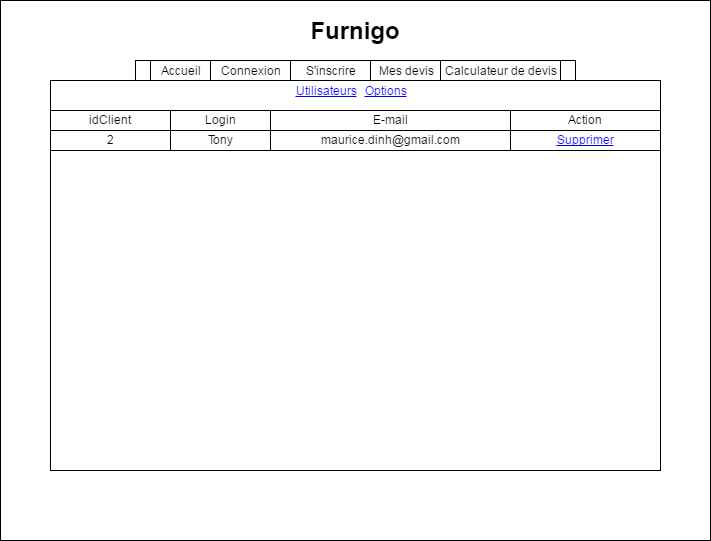


Figure Maquette de l'interface de gestion d'utilisateurs

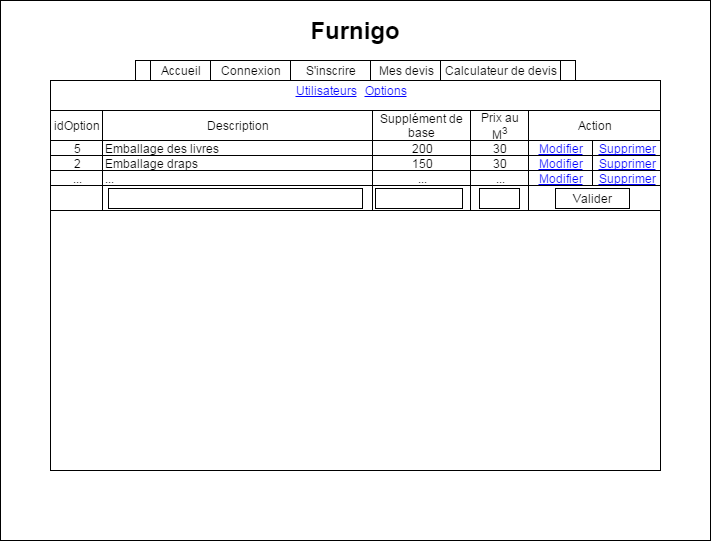


Figure Maquette de l'interface de gestion d'option

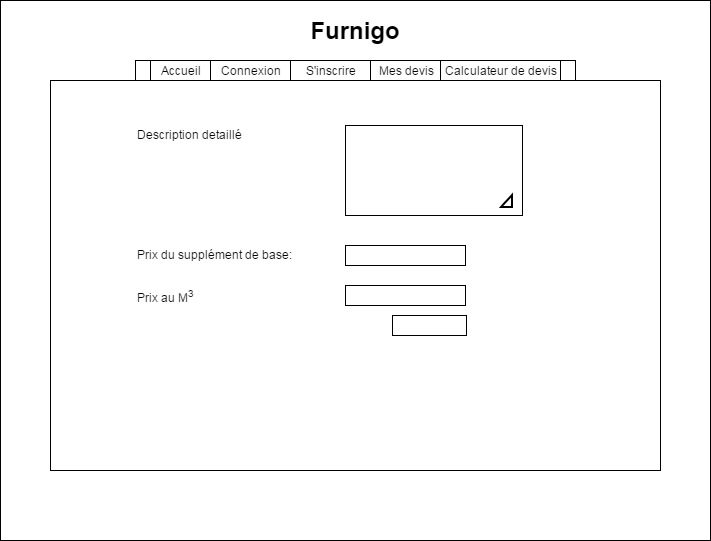


Figure Maquette de l'interface de modification d'option

# Analyse organique

Cette section sera séparée en deux parties :

* Les échanges de données
* Description des fonctions

Cette partie montre l’architecture, la logique de mon projet et les méthodes de manipulation des données.

## Manipulation de devis

### Création de devis

Le calculateur envois les données entrées par l’utilisateur à la page registerDevis.php, celui-ci exécute la fonction CreateQuotation et insère dans la base de donnée les informations donné en paramètre de la fonction. L’utilisateur est ensuite redirigé vers la page devis.php si la création s’est bien déroulée.

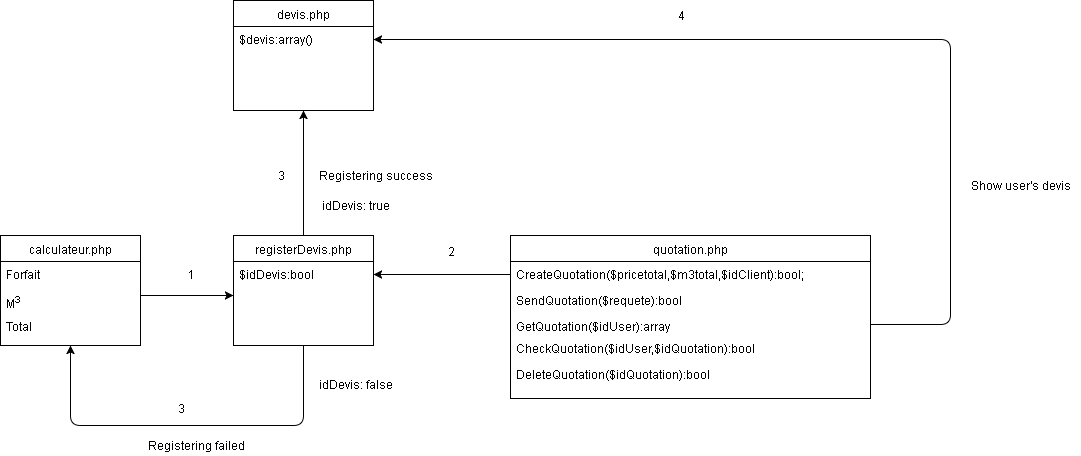


Figure Schéma logique de la création d'un devis

### Suppression de devis

Requête de suppression d’un devis par un utilisateur l’utilisateur clique sur le lien « Supprimer ». Celui-ci est redirigé vers le fichier delete.php en passant dans le GET l’id du devis à supprimer. Le fichier delete.php appel ensuite la fonction DeleteQuotation en passant l’id reçu en GET en paramètre, qui envoie une requête de suppression pour la table r\_ajouter, t\_detail et t\_devis, car la suppression d’un devis doit également supprimer les options sélectionné pour le devis mais également les détails du devis. Une fois la fonction exécutée, le client retourne sur la page devis.php avec un message selon le résultat de la fonction.

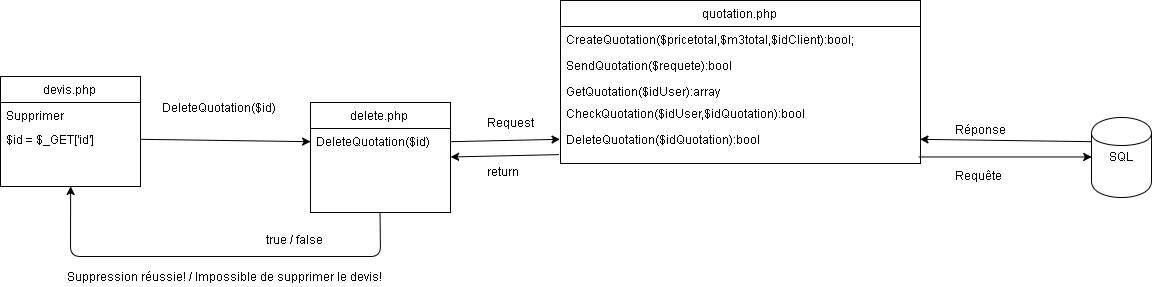


Figure Schéma logique de la suppression d'un devis

### Modification d’un devis

Requête de modification d’un devis. L’utilisateur commence par choisir le devis à modifier en cliquant sur le lien « Modifier » correspondant au devis à modifier. Il est ensuite redirigé sur la page modification qui est exactement la même que la page calculateur, sauf qu’elle contient les paramètres précédent des options du devis.

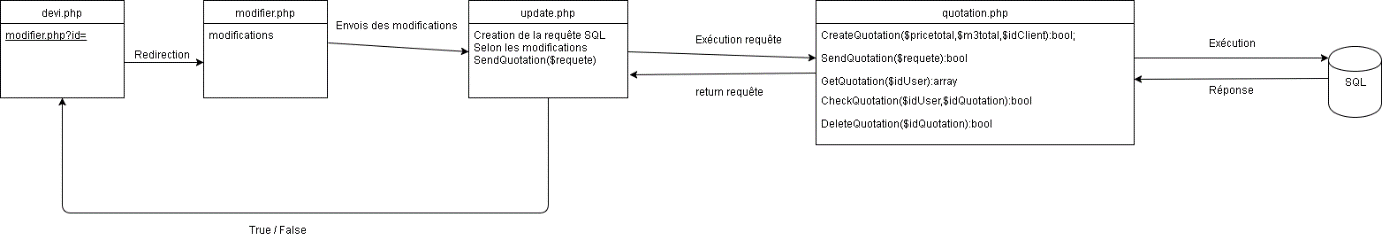


Figure Schéma logique de la modification d'un devis

### Affichage des devis

Requête pour l’affichage des devis de l’utilisateur. Le navigateur envois une requête à la bdd avec son id d’utilisateur. La bdd retourne la liste des requêtes selon l’id de l’utilisateur lié au devis.

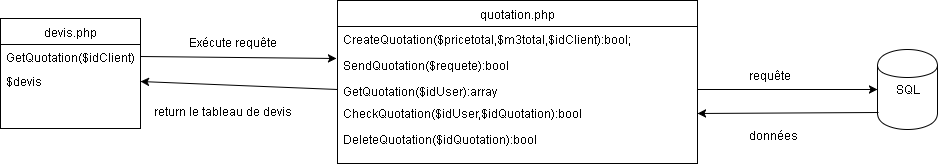


Figure Schéma logique de l'affichage des devis

## Manipulation des options

### Création d’une option

Requête de création d’une option par l’administrateur. L’administrateur remplit le formulaire au pied du tableau des options et l’envoi ensuite. Les donnée sont envoyé au fichier addoption.php qui valide les données fournies et les insère en paramètre dans la fonction CreateOption en paramètre qui s’exécute ensuite. La fonction envois sa requête à la base de donnée qui lui renvois une réponse. La fonction, elle retourne un boolean qui est ensuite traité par le fichier addoption.php et est interprété comme un message qui est ensuite renvoyé vers la page adminoptions.php.

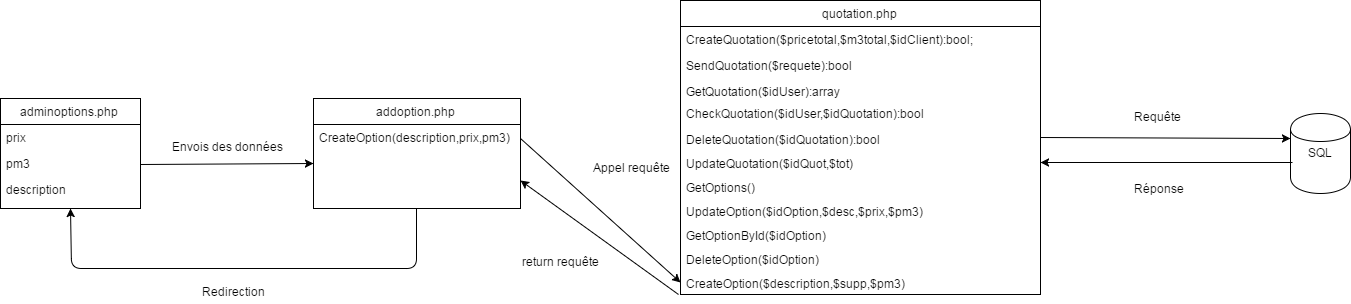


Figure Schéma logique de la création d'une option

### Affichage d’option

Requête d’affichage des options contenues dans la base de données. La fonction GetOptions() est appelé dans la page ou l’on doit afficher les options et stocké dans une variable. La fonction s’exécute, envoi la requête à la base de données. Celle-ci retourne les options à la fonction qui celle-ci à son tour retourne le résultat dans la fonction $options à la vue.

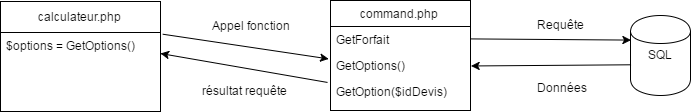


Figure Schéma de la logique d'affichage des options

### Modification d’option

La requête de modification commence à la page adminoption.php. L’utilisateur clique sur le lien « Modifier » ce qui le redirige vers la page modifieroption.php avec les données précédente de l’option déjà dans la valeur des champs. Une fois les modifications effectué, l’utilisateur valide le formulaire et les données sont envoyées sous forme de $\_POST à la page modOption.php. Cette page appelle ensuite la fonction UpdateOption en passant les données du POST en paramètres, la fonction envois ensuite une requête à la base de données qui renvois une réponse suite à la requête. La fonction return ensuite un booléen qui définit le message à afficher et redirige ensuite le client vers la page de gestion d’options.

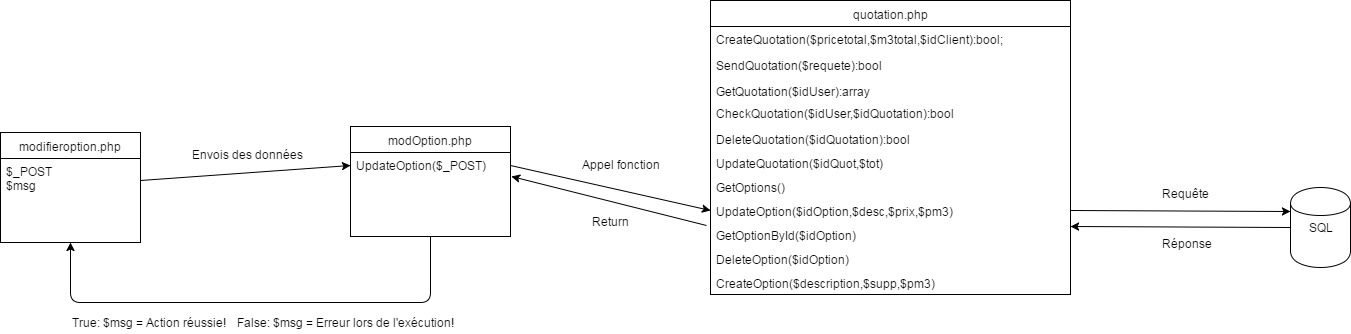


Figure Schéma de la logique de la modification de données

### Suppression d’une option

La suppression d’une option se fait depuis l’interface de gestion d’option à la page modifieroption.php. Le lien « Supprimer » récupère l’id de l’option à supprimer et l’envoi à la au script deleteoption.php. Celui-ci appel la fonction DeleteOption dans le script quotation.php en passant en paramètre l’id de l’option. La fonction envois une requête à la base de donnée qui renvois une réponse que la fonction interprète et revois un return. Selon la réponse du return, la page deleteoption.php redirige ensuite l’utilisateur vers la page modifieroption.php avec un message.

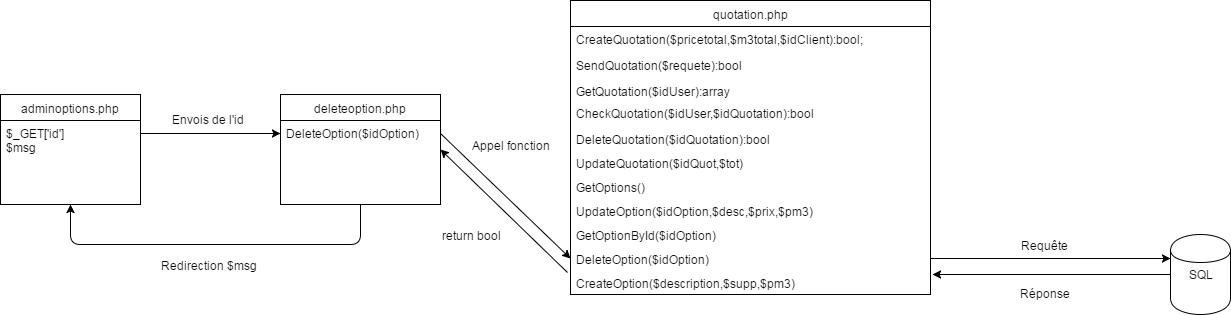
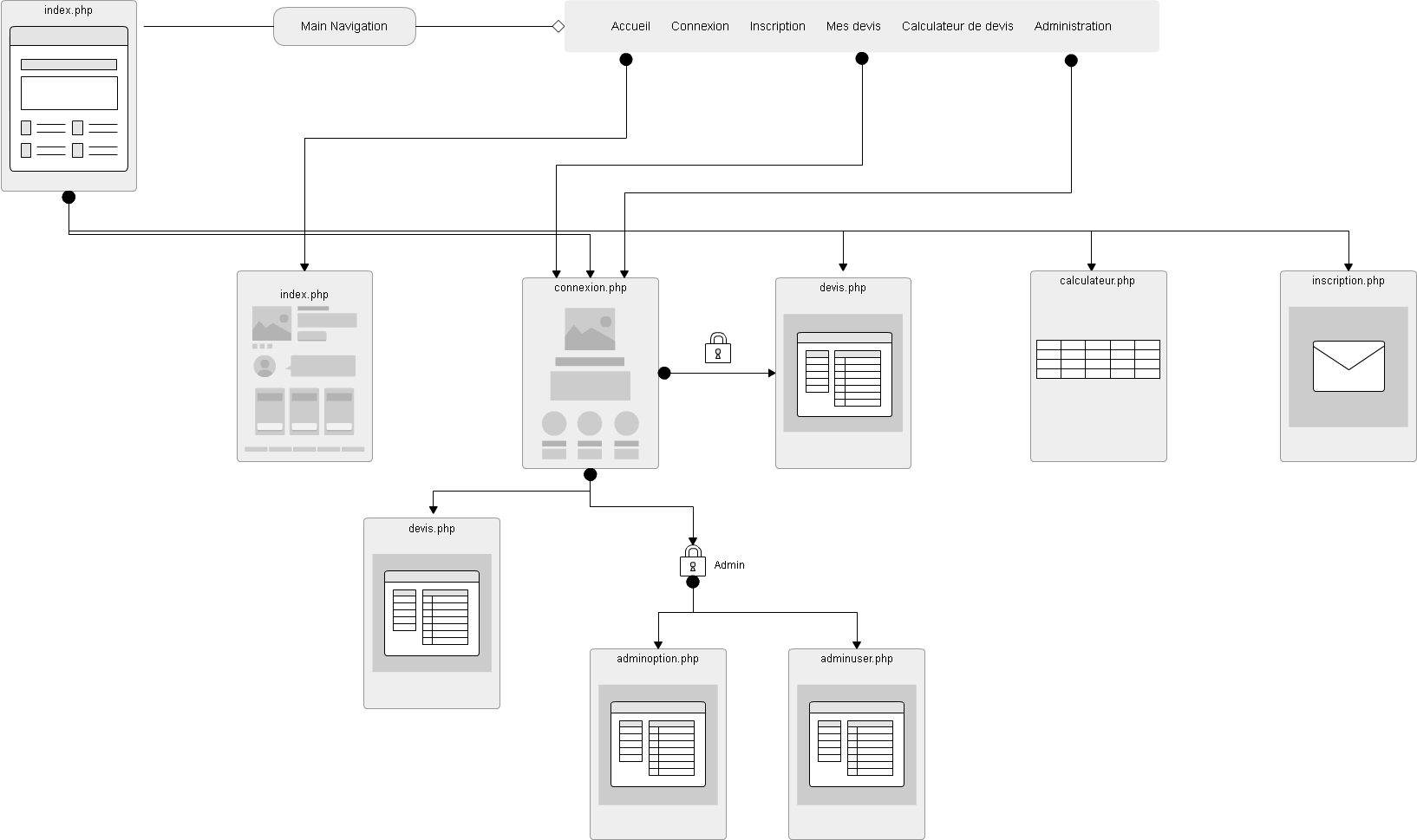


Figure Schéma logique de la suppression d'une option

## Architecture du site



## Description des fonctions

Cette partie décrit les fonctions et leurs propos.

Calculateur.php (javascript) :

* CalculTotal() :
  + Cette fonction parcourt chaque cellule total d’option et les additionnes, elle retourne ensuite le résultat final.
* ShowTotal() :
  + Cette fonction récupère le prix total du devis avec CalculTotal() et le stocke dans la cellule totale du tableau et dans un input « hidden ».
* QtTotal(qt, unit, supp) :
  + Cette fonction prend trois int en paramètre et les calculs. Le calcul est le suivant : qt \* unit + supp = total   
    La fonction retourne ensuite le total
* GetTotalM3() :
  + Cette fonction parcourt les inputs number de la page et additionne leur valeur. Elle return ensuite le total.
* ShowResult(id) :
  + Cette fonction s’occupe de l’affichage du prix total par option et du prix total du devis également. L’id en paramètre est utilisé pour déterminer quelle option à calculer.

Quotation.php (php) :

* CreateQuotation($pricetotal, $m3total, $idClient) :
  + Cette fonction sert à créer un devis dans la table devis de notre base de données. Elle prend en paramètre le prix total du devis, le volume total des meubliers et l’id du client lié au devis.
* SendQuotation($requete) :
  + Cette fonction sert à envoyer une requête à la base de données avec la requête passée en paramètre. Cette fonction est plutôt utilisé dans des boucles quand certain paramètre ont besoin d’être indenté.
* GetQuotation($idUser) :
  + Cette requête récupère les devis lié à un utilisateur grâce à l’id de celui-ci. Elle est utilisée dans la page mesdevis.php pour afficher les requêtes appartenant à l’utilisateur connecté.
* CheckQuotation($idUser, $idQuotation) :
  + Cette requête fait un SELECT de tous les devis dont l’id du client est lié à l’id du devis. Elle est utilisé pour vérifier si le devis qu’un utilisateur connecté essaie de supprimer lui appartient ou pas et il retourne un booléen selon la réponse.
* DeleteQuotation($idQuotation) :
  + Cette fonction sert à supprimer un devis. Pour cela, cette fonction envois 3 requêtes.
    - La première est pour supprimer les options de ma base liée au devis à supprimer.
    - La deuxième est utilisée pour supprimer les détails liés au devis
    - Finalement le troisième est pour supprimer le devis en question.
  + Ces requêtes doivent être exécutées dans cet ordre pour question de liaison.
* UpdateQuotation($idQuot, $tot) :
  + Cette fonction sert à modifier les données d’un devis en prenant l’id du devis et la nouvelle somme du devis en paramètre.
* GetOptions() :
  + Cette fonction envoie une requête à la base de donnée et récupère les données d’un select. La fonction renvoi ensuite les données dans le return.
* UpdateOption($idOption, $desc, $prix, $pm3) :
  + Cette fonction envoie une requête d’update à la base de donnée. Elle sert à modifier les informations d’une option en passant en paramètre l’id de l’option à modifier, la nouvelle description de l’option, le prix en supplément et le prix par M3.
* GetOptionById($idOption) :
  + Cette fonction envoie une requête select à la base de donnée en passant en précisant l’id de l’option à afficher. Cette fonction est utilisée pour la modification de cette option. On récupère les données de l’option pour afficher les données précédentes dans les champs de modification.
* DeleteOption($idOption) :
  + Cette fonction envoie une requête delete à la base de données en précisant l’id de l’option à supprimer. Cette fonction est utilisée par l’administrateur pour supprimer uniquement lorsqu’il gère les options.
* CreateOption($description, $supp, $pm3) :
  + Cette fonction envoie une requête insert into à la base de données, en passant les données de l’insert en paramètre. La fonction prend en paramètre la description de l’option, son prix supplément et le prix par M3. Cette fonction est utilisée par l’administrateur lors de la création d’une nouvelle option.
* AddDetails($lieu, $volume, $surface, $poids, $distance, $idDevis) :
  + Cette fonction envoie une requête insert into à la base de données en passant en paramètre le lieu du déménagement, le volume total des meubles à transporter, la surface total de l’endroit, le poids total des meubles, la distance à parcourir jusqu’à la nouvelle adresse (en km) et l’id du devis auquel les détails doivent être ajoutés. Cette fonction est appelée lors de la création d’un devis pour ajouter les détails au devis.
* UpdateDetails($lieu, $totalm3, $surface, $poids, $distance, $idDevis) :
  + Cette fonction envoie une requête update à la base de données en envoyant les nouvelles informations précisées en paramètre de la fonction et en choisissant le devis grâce à son devis. Cette fonction est utilisée par un utilisateur lorsqu’il veut modifier les détails de son devis.

Command.php (php) :

* GetForfait() :
  + Cette fonction récupère les forfaits disponible dans la base données grâce à un select. Cette fonction est utilisé quand on veut afficher les forfaits disponible dans un select.
* GetOption($idDevis) :
  + Cette fonction récupère les options d’un devis existant en passant en paramètre l’id du devis. Cette fonction est utilisée pour la modification d’un devis existant. On récupère les options précédentes de l’option et on les affiches dans le formulaire.
* GetDetail($idDevis) :
  + Cette fonction récupère les détails d’un devis désigné par son id en paramètre. Cette fonction est utilisée à la modification d’un devis pour afficher les détails précédent dans le formulaire.

Users.php (php) :

* CheckUserId($Login, $pwd) :
  + Cette fonction envois une requête à la base de données en recherchant à l’aide d’un select l’existence d’un utilisateur passé en paramètre possédant un mot de passe passé en paramètre. Cette fonction est utilisée lors de la connexion d’un utilisateur.
* AddUser($pseudo,$mail,$pwd) :
  + Cette fonction ajoute un utilisateur dans la base de données avec son pseudo, son adresse email son mot de passe hashé passé en paramètre. Cette fonction est bien sur utilisée lors de l’inscription d’un utilisateur.
* GetUsers() :
  + Cette fonction envoie une requête à la base de données pour récupérer tous les utilisateurs qui ne sont pas administrateur. Cette fonction est utilisée lors de l’affichage des utilisateurs pour la gestion d’utilisateurs par l’administrateur.
* DeleteUser($idUser) :
  + Cette fonction envois une requête de suppression à la base de données grâce à l’id de l’utilisateur passé en paramètre pour d’abord supprimer les options lié au devis. Puis viens ensuite les détails du devis, suivis du devis en lui-même et finalement l’utilisateur.

# Planning finale

# 

# Tests et protocole de tests

Cette partie fournie une description détaillé des scénarios de toutes les actions possibles et leurs conclusion.

|  |  |  |
| --- | --- | --- |
| N° Test | Scénario | Résultat attendu |
| 1 | Un utilisateur anonyme arrive sur le site et essaie d’accéder aux pages devis.php, adminoptions.php ou encore adminuser.php. | L’utilisateur anonyme ne peut pas accéder à cette page sans être connecté et est redirigé vers la page connexion.php |
| 1.1 | Un utilisateur anonyme va dans le calculateur de devis et essaie de créer un devis | Il est redirigé vers la page connxion.php avec un message d’erreur |
| 2 | Un utilisateur se connecte | Il est redirigé vers la page index.php et le lien vers connexion.php est retiré de la barre de navigation |
| 2.1 | Un utilisateur se connecte avec de mauvais identifiant | Il est redirigé vers la page connexion.php avec un message d’erreur de connexion |
| 2.2 | Un utilisateur connecté clique sur le lien Mes devis | Il est redirigé vers la page devis.php et ses devis sont affichés |
| 2.3 | Un utilisateur connecté tente de créer un nouveau devis | Il est redirigé vers la page devis.php et ses devis sont affichés dont celui qui vient d’être créé |
| 2.4 | Un utilisateur normal choisit de modifier un de ses devis existant | Il est redirigé vers la page modifier.php et les données du devis à modifier sont affichées dans le formulaire |
| 2.5 | Un utilisateur normal choisit de modifier un devis avec un id qui ne lui appartient pas via l’url avec un get | Il est redirigé vers la page devis avec le message « Ce devis ne vous appartient pas! » |
| 2.6 | Un utilisateur normale envois en get un id d’un devis qui ne lui appartient pas pour le supprimer | Il est redirigé sur la page devis.php avec le message d’erreur « Ce devis ne vous appartient pas! » |
| 2.7 | Un utilisateur normale essaie d’accéder aux pages adminuser.php et adminoptions.php | Il est redirigé vers la page connexion.php avec le code d’erreur n°5 (Seul l’admin peut accéder à cette page !) |
| 2.8 | Un utilisateur connecté supprime un de ses devis | Il est redirigé vers la page devis.php avec le message « Suppression réussie ! » |
| 2.9 | L’utilisateur modifie un devis et envoi son formulaire | Il est redirigé vers la page devis.php avec le message "Modification réussie!" |
| 3 | L’administrateur se connecte | Il est redirigé vers la page accueil et le lien administration s’affiche dans la barre de navigation. |
| 3.1 | L’utilisateur clique sur administration | Il est redirigé vers la page adminuser.php et la liste des utilisateurs de son site s’affiche. |
| 3.2 | L’utilisateur clique sur le lien Options de la page administration. | Il est redirigé vers la page adminoptions.php et la liste des options à modifier s’affiche |
| 3.3 | L’administrateur clique sur le lien « Supprimer » d’un utilisateur | Il est redirigé vers le fichier de redirection delete.php qui appelle la fonction DeleteUser() et redirige l’admin vers la page adminoptions.php avec un code de message. |
| 3.4 | L’administrateur clique sur le le lien « Supprimer » d’une option | L’administrateur est redirigé vers la page deleteoption.php avec qui appelle la fonction DeleteOption() et qui redirige ensuite l’administrateur vers la page adminoptions.php avec un code de message en GET. |
| 3.5 | L’administrateur clique sur le lien « Modifier » d’une option | Il est redirigé vers la page modifieroption.php ou les données de l’option sont affichées dans un formulaire |
| 3.6 | L’administrateur envoi le formulaire de modification | Il est redirigé vers la page modOption.php qui appelle une fonction de modification et l’admin est redirigé vers la page adminoptions.php |
| 4 | Un utilisateur connecté clique sur le lien Déconnexion | Le script de déconnexion s’exécute et l’utilisateur est redirigé vers la page index.php |
| 4.1 | Un utilisateur anonyme accède à la page logout.php | Il est redirigé vers la page connexion.php avec le message d’erreur « Vous n’êtes pas connecté ! » |
| 5 | Un utilisateur anonyme s’inscrit et confirme son formulaire d’inscription | Il est redirigé vers la page index.php |
| 5.1 | Un utilisateur anonyme s’inscrit et entre une mauvaise adresse email | Un message d’erreur html5 s’affiche et l’envoi du formulaire ne se fait pas |
| 5.2 | Un utilisateur anonyme qui s’inscrit mais qui n’entre pas le même mot de passe dans la confirmation | Un message d’erreur html5 s’affiche et l’envoi du formulaire ne se fait pas |

# Conclusion

La réalisation de ce projet m’a permis d’appliquer mes connaissances en PHP en un projet concret qui peut se rendre utile. J’ai eu un certain nombre d’obstacle à surmonter lors du développement du projet, mais cela m’a rendu plus à l’aise avec le langage et les différentes manières de résoudre un problème (débogage).

Ce projet a été mon plus gros projet de ma formation. J’ai appliqué toutes mes notions acquise lors de ces 4 ans. C’est pour moi une autosatisfaction d’avoir réalisé ce site en PHP. C’est ainsi que je vois mon futur métier, mais avec moins de documentation.

## Amélioration possible

En bilan, les fonctionnalités principales du site sont fonctionnelles mais malgré cela, il y a des améliorations possibles qui ont été découvert pendant le développement du site.

En voici quelques-uns :

* Passer l’architecture de mon site en MVC
* Lors de l’inscription, accepter un email en unique
* Ajouter la description des forfaits et afficher leur prix

## Remerciement

Je tiens à vous remercier, vous experts mais également mon professeur M.Blanvillain, pour votre participation à la réalisation de ce projet.