|  |  |  |
| --- | --- | --- |
| TopRecettes | TPI 2015 | |
| Documentation Technique | | Cedric Dos Reis |

Table des matières

[Introduction 2](#_Toc419217209)

[Etude d’opportunité 2](#_Toc419217210)

[Analyse fonctionnelle 3](#_Toc419217211)

[Généralité 3](#_Toc419217212)

[Fonctions globales 3](#_Toc419217213)

[Connexion au site 4](#_Toc419217214)

[Inscription au site 4](#_Toc419217215)

[Afficher la liste des recettes 4](#_Toc419217216)

[Afficher les détails d’une recette 4](#_Toc419217217)

[Ajouter une nouvelle recette 5](#_Toc419217218)

[Modifier une recette 5](#_Toc419217219)

[Supprimer une recette 5](#_Toc419217220)

[Ajouter un commentaire / note à une recette 5](#_Toc419217221)

[Supprimer un commentaire / note 5](#_Toc419217222)

[Modifier son mot de passe 5](#_Toc419217223)

[Modifier un utilisateur 6](#_Toc419217224)

[Supprimer un utilisateur 6](#_Toc419217225)

[Interfaces 7](#_Toc419217226)

[Fonctions particulières 14](#_Toc419217227)

[Eléments de sécurité 14](#_Toc419217228)

[Analyse Organique 14](#_Toc419217229)

[Généralité 14](#_Toc419217230)

[Description des méthodes (algorithmes) de résolution 15](#_Toc419217231)

[Description des actions 15](#_Toc419217232)

[Description des fonctions 15](#_Toc419217233)

[Tests 16](#_Toc419217234)

[Plan de test 16](#_Toc419217235)

[Rapport de test 17](#_Toc419217236)

[Conclusion 18](#_Toc419217237)

[Améliorations 18](#_Toc419217238)

[Comparaison énoncé avec projet final 18](#_Toc419217239)

# Introduction

Ce projet est réalisé dans le cadre du TPI. Le but du projet est de réaliser un site web qui permet de gérer des recettes de cuisine. Les recettes appartiennent à une catégorie de plat. Elles sont composées d’ingrédients avec différentes quantités

# Etude d’opportunité

Les sites de recettes en ligne existent dans tous les pays et sont très nombreux sur le net.

Mon site étant un gestionnaire de recettes, je vais m’intéresser aux lieders français des recettes de cuisine en ligne

Marmiton propose aux visiteurs plus de 60'000 recettes. On y retrouve tous types de recettes comme par exemple les entrées, les plats principaux, les desserts et autres.

Ses différentes options de recherche permettent aux utilisateurs de trouver rapidement les recettes qu’ils recherchent. Il y a notamment la difficulté de la recette, son cout, le type de plat et d’autres tris.

Un utilisateur inscrit peut proposer une recette qui doit être validée par les administrateurs avant d’être affichée sur le site

Une recette sur marmiton contient un titre, la liste des ingrédients nécessaires et leur quantité, la préparation de la recette, le temps de préparation, le temps de cuisson, le nombre de portions et il y a parfois une vidéo en rapport avec la recette que vous consultez.

Marmiton propose aussi pour ceux qui le souhaitent d’afficher la recette en version imprimable.



# Analyse fonctionnelle

## Généralité

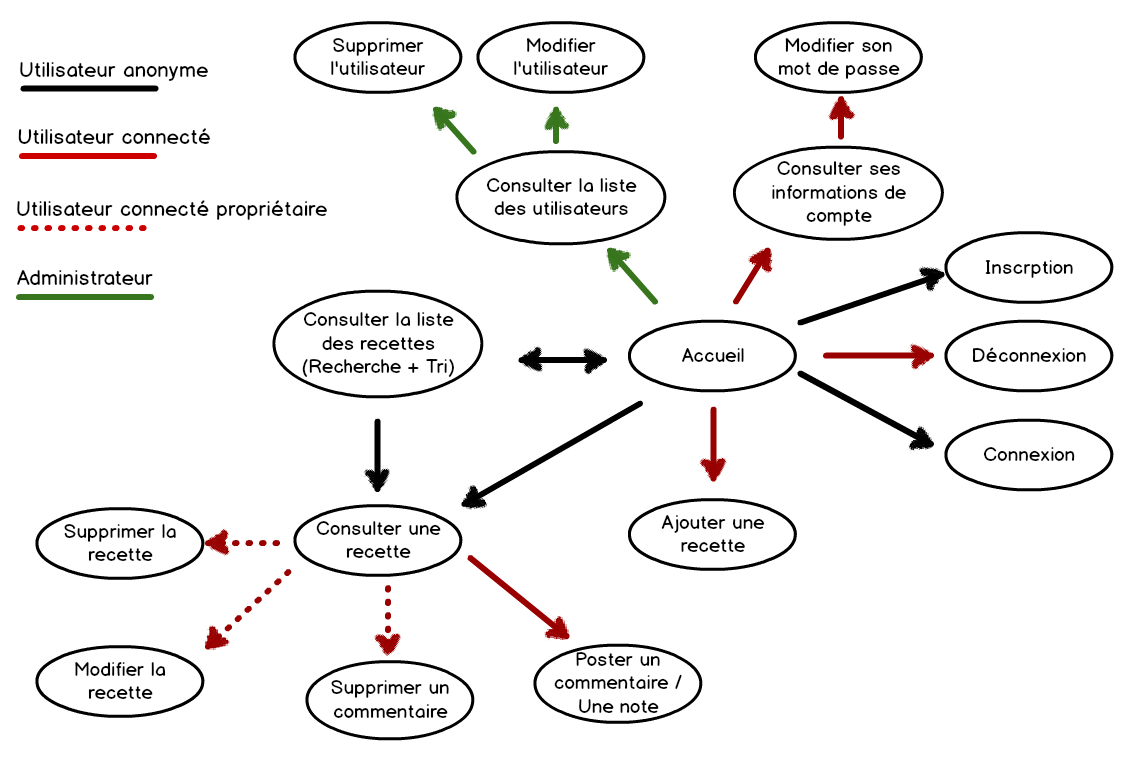


Figure 1 : Schéma de fonctionnalités

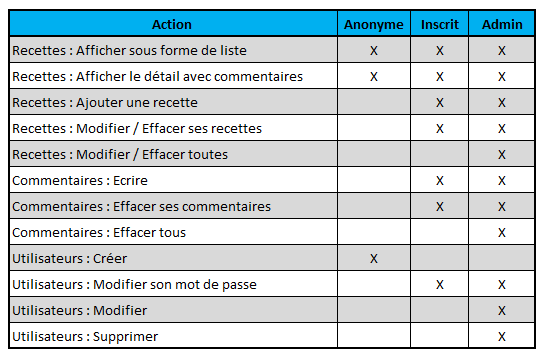


Figure 2 : Droits des utilisateurs

## Fonctions globales

### Connexion au site

Un utilisateur anonyme peut se connecter au site s’il possède un déjà un compte.

Les informations nécessaires à la connexion sont :

* L’email
* Le mot de passe

### Inscription au site

Un utilisateur anonyme peut s’inscrire au site s’il ne possède pas de compte.

Les informations nécessaires à la connexion sont :

* Un pseudo non utilisé
* Un email non utilisé
* Un mot de passe
* Une confirmation du mot de passe

### Afficher la liste des recettes

Tout le monde peut afficher la liste des recettes, celles-ci peuvent être triées par date ou par note, il est également possible de rechercher une recette par son nom.

Pour un utilisateur connecté, il est également possible d’afficher uniquement ses recettes.

Les informations récupérées sont les suivantes :

* Le titre de la recette
* L’image représentant la recette
* L’auteur de la recette
* La date de publication de la recette
* La moyenne des notes attribuées à la recette

### Afficher les détails d’une recette

Tout le monde peut consulter les détails d’une recette à partir de la liste.

Les informations récupérées sont les suivantes :

* Le titre de la recette
* L’image représentant la recette
* Les ingrédients qui composent la recette
* Les quantités d’ingrédients nécessaires à la réalisation de la recette
* L’explication pour la réalisation de la recette
* L’auteur de la recette
* La date de publication de la recette
* La moyenne des notes attribuées à la recette
* Les commentaires postés sur cette recette

### Ajouter une nouvelle recette

Un utilisateur connecté peut ajouter une nouvelle recette

L’ajout d’une nouvelle recette se déroule en 2 étapes.

Les informations nécessaires à l‘ajout sont :

1ère étape :

* Un titre ->1ere étape
* Le type de plat
* Le nombre d’ingrédients nécessaires
* L’origine de la recette (Non obligatoire)

2ème étape

* Les ingrédients et leur quantité
* La préparation de la recette
* Une image (Non obligatoire)

### Modifier une recette

L’auteur de la recette et les administrateurs ont la possibilité de modifier une recette.

La modification d’une recette se déroule comment l’ajout

### Supprimer une recette

L’auteur de la recette et les administrateurs ont la possibilité de supprimer une recette.

### Ajouter un commentaire / note à une recette

Un utilisateur connecté peut commenter une recette à partir de la page qui en affiche les détails.

Les informations nécessaires à l’ajout d’un commentaire sont :

* Un commentaire
* Une note pour la recette (Non obligatoire)

### Supprimer un commentaire / note

L’auteur du commentaire et les administrateurs ont la possibilité de supprimer un commentaire

Si une note a été attribuée à cette recette

### Modifier son mot de passe

Un utilisateur connecté à la possibilité de modifier son mot de passe

Les informations nécessaires à la modification du mot de passe sont :

* Le mot de passe actuel
* Le nouveau mot de passe
* Confirmation du nouveau mot de passe

### Modifier un utilisateur

Un administrateur à la possibilité de modifier tous les utilisateurs/administrateurs

Les informations qu’il peut modifier sont :

* Le pseudo de l’utilisateur
* L’email de l’utilisateur
* Le type de l’utilisateur
* Le mot de passe de l’utilisateur

### Supprimer un utilisateur

Un administrateur à la possibilité de supprimer tous les utilisateurs/administrateurs

## Interfaces

L’interface du site web est très simple, elle est constituée d’un menu en haut toujours visible par l’utilisateur. Juste en dessous du menu se trouve le titre et le contenu de la page et pour finir tout à la fin se trouve le pied de page.

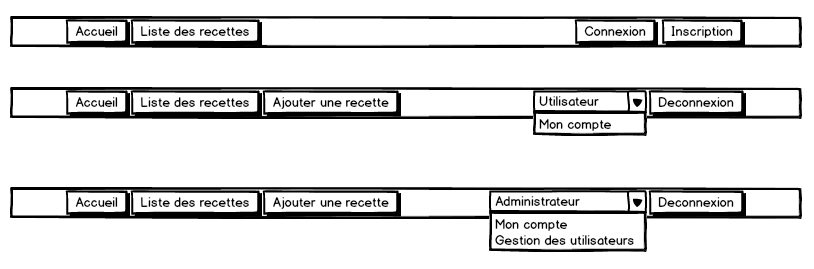
**Le menu change en fonction du type d’utilisateur.**

Un utilisateur anonyme a accès aux liens pour accéder à la fenêtre de connexion, d’inscription et la liste des recettes. Il n’y que lui qui peut accéder à la fenêtre de connexion et d’inscription.

Un utilisateur connecté a accès aux liens pour se déconnecter, pour ajouter une nouvelle recette, pour gérer son compte, pour afficher ses recettes et aussi la liste des recettes.

Un administrateur, lui à accès à toutes les pages disponible pour un utilisateur connecté et plus de celle pour la gestion des utilisateurs.

Figure 3 : Différentes interfaces du menu utilisateur



**La page d’accueil** affiche quelques-unes de meilleures recettes présentes sur le site ainsi que les dernières ajoutées.

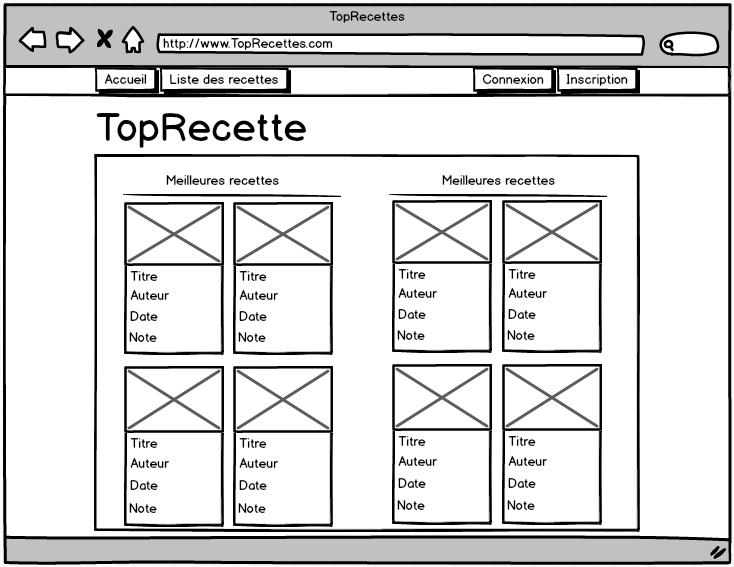


Figure 4 : Interface - Accueil

**La page de connexion** contient un formulaire qui nécessite un email et un mot de passe.

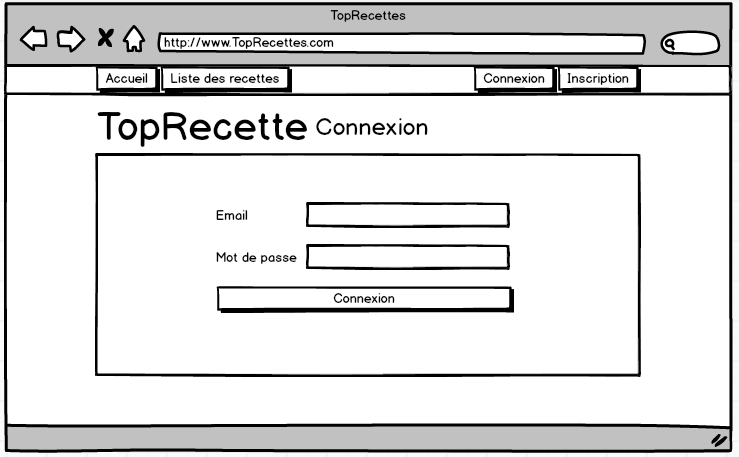


Figure 5 : Interface - Connexion

**La page d’inscription** contient un formulaire qui nécessite un pseudo, un email, un mot de passe et une confirmation du mot de passe.

La page de modification d’un utilisateur ressemble fortement à celle-ci, si ce n’est que le page de modification permet de choisir le type d’utilisateur

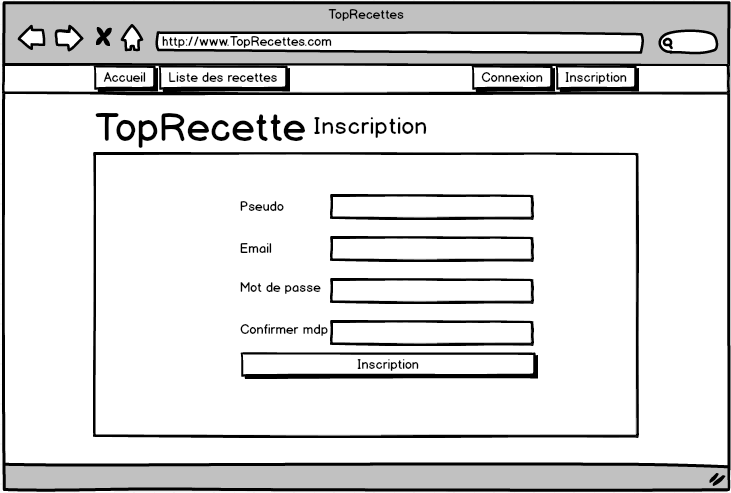


Figure 6 : Interface - Inscription

**La page de consultation de la liste** **des recettes** permet d’afficher un aperçu des recettes disponibles sur le site. Elle permet à l’utilisateur de trier les recettes par date, note ou catégories.

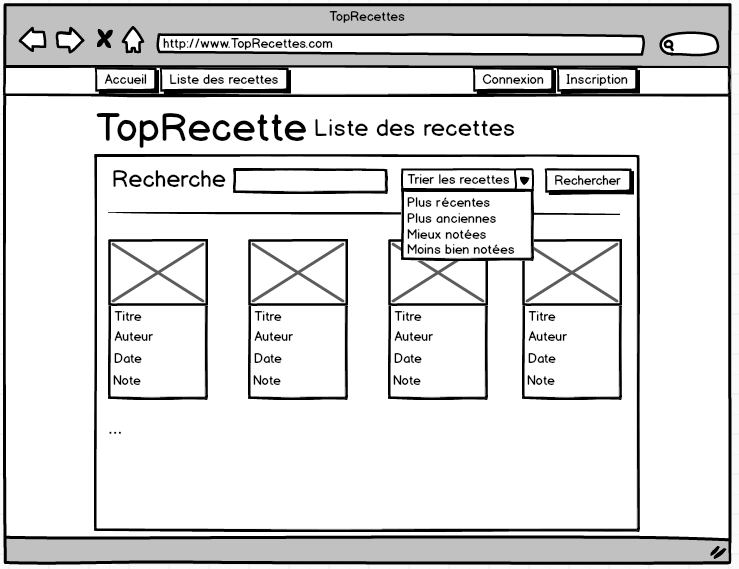
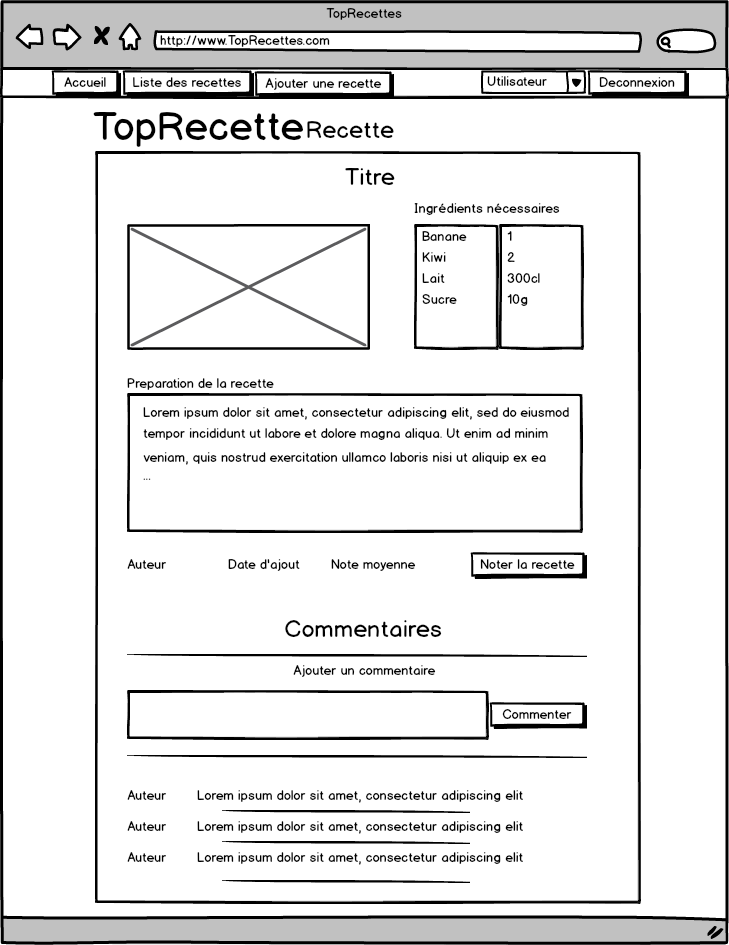


Figure 7 : Interface – Consultation de la liste des recettes

**La page de consultation d’une recette** permet d’afficher toutes les informations nécessaire à la réalisation de la recette, elle affiche également les commentaires poster par les autres utilisateurs. Le propriétaire de la recette et l’administrateur ont la possibilité de supprimer la recette depuis cette page et également d’accéder à la page de modification.

Figure 8 : Interface - Consultation d'une recette



**La page d’édition de recette** permet à un utilisateur connecter de rédiger une recette et l’envoyer. Elle permet également de modifier une recette pour le propriétaire ou l’admin.

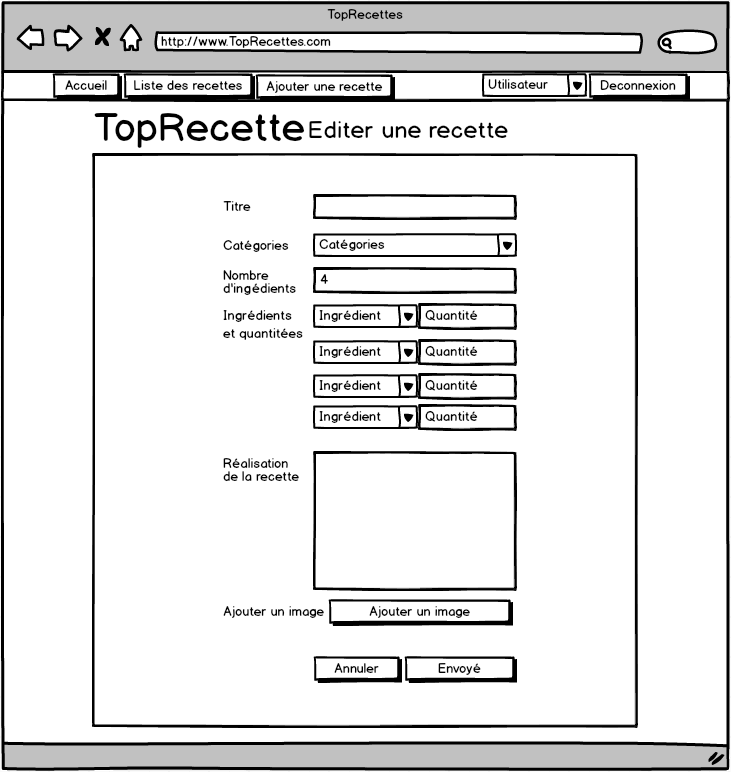
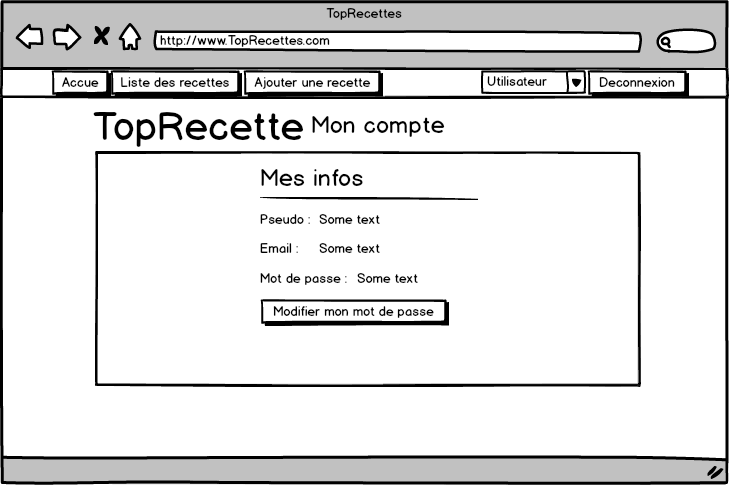


Figure 9 : Interface - édition d’une recette

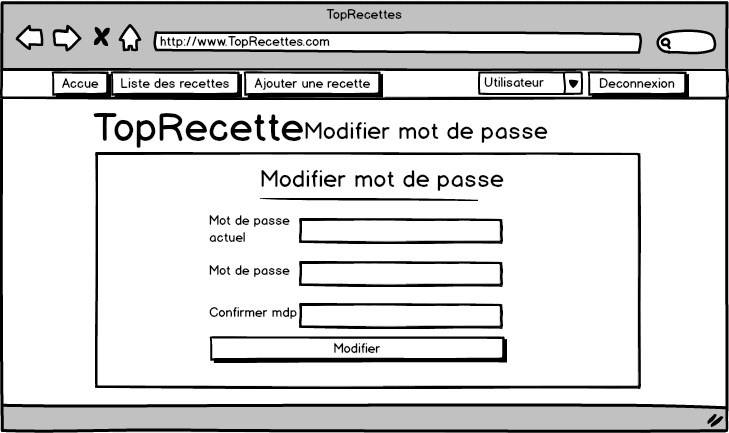
**La page de consultation d’un utilisateur** permet à un utilisateur connecté de consulter ses informations de compte.

Figure 10 : Interface - Consultation d'un utilisateur



**La page de modification de mot de passe** permet à un utilisateur connecté de modifier son mot de passe

Figure 11 : Interface - Modification du mot de passe



**La page de consultation de la liste** **des utilisateurs** permet à un administrateur d’afficher tous les utilisateurs qui se sont inscrits sur le site.

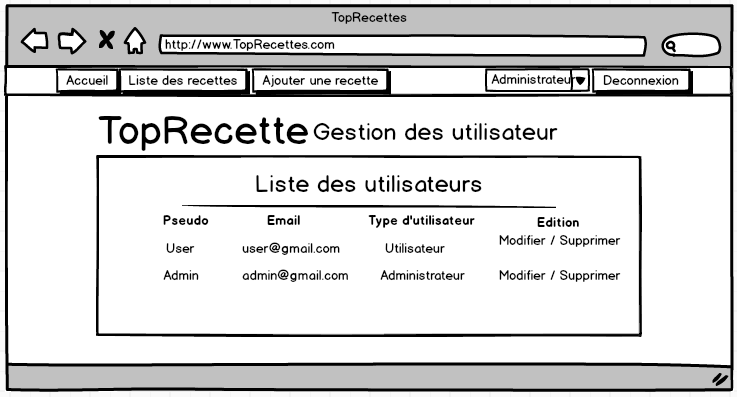


Figure 12 : Interface - Consultation de la liste des utilisateurs

**La page de modification d’un utilisateur** permet à un administrateur de modifier toutes les informations d’un utilisateur

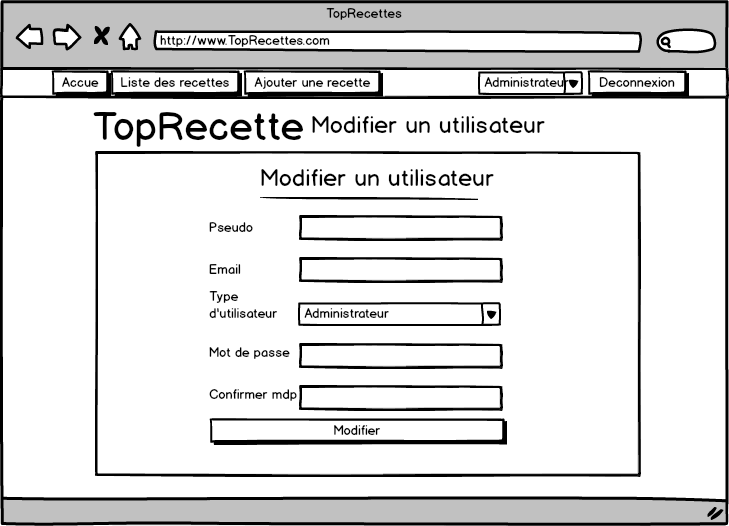


Figure 13 : Interface - Modification d'un utilisateur

## Fonctions particulières

## Eléments de sécurité

Pour assurer la sécurité des utilisateurs, les mots de passe son cryptés grâce à la fonction de hachage *Secure Hash Algorithm* « SHA1 ».

# Analyse Organique

## Généralité

### Model conceptuel de données

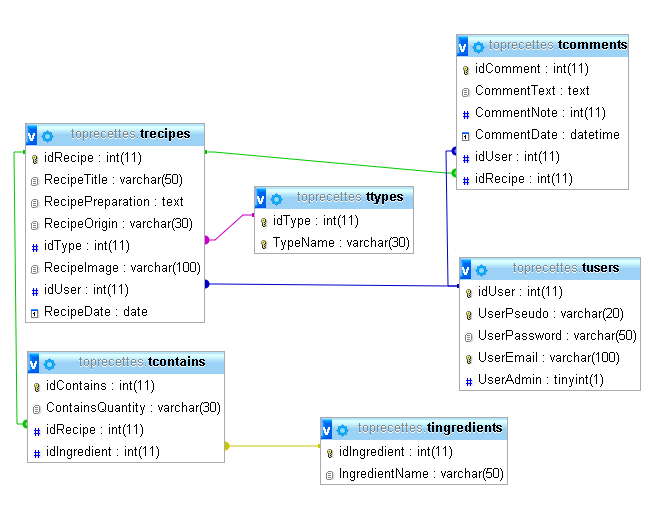


Figure 14 : Model conceptuel de données

Voici mon model conceptuel de données. Il contient 6 tables

## Description des méthodes (algorithmes) de résolution

### Description des actions

### Description des fonctions

|  |  |  |
| --- | --- | --- |
| **Fonction** | **Paramètres** | **Description** |
| connectDB |  | Ouvre la connexion à la base de données |
| Select | $name  $liste  $courant  $required | Permet de créer un Select (formulaire) à partir d'un tableau associatif ($liste)  Sélectionne l’option définit dans $courant |
| DataList | $name  $liste | Construit une série de cases à cocher (formulaire) basée sur le tableau associatif $liste,  Coche l'ensemble de éléments cochés signalé comme tel dans le tableaus $checked  et ajoute les éventuels attributs |
| ShowError | $Message | Affiche un message dans un block rouge (Message d’erreur en paramètre). |
| ShowSuccess | $Message | Affiche un message dans un block vert (Message de succès en paramètre). |
| login | $UserEmail  $UserPassword | Connexion au site.  Récupère l’utilisateur dont les données sont reçues en paramètres correspondent.  Si un utilisateur est récupéré, renvoie vrai et l’utilisateur est connecté.  Si aucun utilisateur n’est récupéré, renvoie faux. |
| register | $UserPassword  $UserEmail  $UserPassword | Inscription au site.  Insert les données en en paramètre dans la table tusers. |
| get\_users |  | Récupère tous les utilisateurs présents dans la base.  Renvoie la liste des utilisateurs. |
| get\_user | $idUser | Récupère les données (Email, Pseudo, Type d’utilisateur) de l'utilisateur dont l’identifiant est en paramètre  Renvoie les données de l’utilisateur. |
| get\_user\_pseudo | $IdUser | Récupère le pseudo de l'utilisateur dont l’identifiant est en paramètre.  Renvoie le pseudo. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Fonction** | **Paramètres** | **Description** | |
| check\_password | $idUser  $Password | Récupère le mot de passe de l’utilisateur dont l’identifiant est en paramètre.  Vérifie que le mot de passe récupéré correspond au mot de passe en paramètre.  Renvoie vrai si les mots de passe correspondent.  Renvoie faux s’ils ne correspondent pas. | |
| CheckExist\_Pseudo | $UserPseudo | Vérifie si le pseudo en paramètre est déjà utilisé.  Récupère le pseudo qui correspond au pseudo en paramètre.  Si un pseudo est récupéré, renvoie vrai car il existe déjà.  Si aucun pseudo n’est récupéré, renvoie faut car il n’existe pas. | |
| CheckExist\_Email | $UserEmail | Vérifie si l’adresse email reçu en paramètre est déjà utilisés Récupère l’email qui correspond à l’email reçu.  Si un email est récupéré renvoie vrai car il existe déjà.  Si aucun email n’est récupéré, renvoie faut car elle n’existe pas. | |
| CheckAdmin | $idUser | Test le type d’utilisateur.  Récupère le type de l’utilisateur dont l’identifiant est en paramètre.  Si le type est 0, renvoie faux car l’utilisateur n’est pas administrateur.  Si le type est 1, renvoie vrai car l’utilisateur est administrateur. | |
| edit\_user | $UserPseudo  $UserEmail  $idUser | Modifie les données (Pseudo et Email) de l'utilisateur dont l’identifiant est en paramètre.  Si le pseudo en paramètre est défini et l’email paramètre est vide alors modifie uniquement le pseudo de l’utilisateur.  Si le pseudo est vide et l’email est défini alors modifie uniquement l’email de l’utilisateur.  Si le pseudo et l’email sont définis alors modifie le pseudo et l’email de l’utilisateur. |

|  |  |  |
| --- | --- | --- |
| **Fonction** | **Paramètres** | **Description** |
| set\_administrateur | $UserAdmin  $idUser | Modifie le type de l'utilisateur en fonction du type d’utilisateur qui est en paramètre. |
| edit\_user\_password | $idUser  $NewPassword | Modifie le mot de passe de l'utilisateur dont l’identifiant est en paramètre. |
| delete\_user | $idUser | Supprime l'utilisateur dont l’identifiant est en paramètre. |
| get\_recipes | $sort  $search  $idUser  $limit | Récupère le la liste de des recettes en fonction des paramètres reçu.  Si tous les paramètres sont vides alors elle récupère toutes les recettes de la base.  Si le tri ($sort) est défini alors elle récupère toutes les recettes en les triant selon le tri reçu (par date d’ajout, par note).  Si la recherche ($search) est définie alors elle récupère les recettes qui correspondent à la recherche.  Si l’identifiant d’un utilisateur est reçu alors elle récupère les recettes postées par cet utilisateur.  Si une limite est définie alors elle récupère un nombre de recettes défini par la limite  Un tri et une recherche peuvent être effectués en même temps  Renvoie en tableau de données contenant les recettes récupérées |
| get\_recipe | $idRecipe | Récupère les données (Titre, préparation, origine, type, moyenne des notes, image, l’auteur) d’une recette dont l’identifiant est en paramètre.  Renvoie un tableau de données. |

|  |  |  |
| --- | --- | --- |
| **Fonction** | **Paramètres** | **Description** |
| add\_recipe | $RecipeInfos  $idUser | Insert une nouvelle recette dans la table trecipe (Titre, préparation, origine, type, moyenne des notes, image, l’auteur).  Si une image est reçue alors elle upload et ajoute la recette avec l’image reçu en paramètre.  Su aucune image n’est reçue alors elle ajoute la recette sans image.  Renvoie l’identifiant de la recette qui vient d’être ajouté. |
| edit\_recipe | $idRecipe  $RecipeInfos | Modifie la recette dans la table trecipes dont l’identifiant est reçu en paramètre.  Si une nouvelle image est reçue alors elle vérifie l’image actuelle de la recette.  Si l’image actuelle de la recette est celle par défaut alors elle upload la nouvelle image et modifie la recette avec la nouvelle image.  Si l’image actuelle est différente de celle par défaut alors l’image actuelle est supprimée du dossier, la nouvelle image est uploadée et la recette est modifiée avec la nouvelle image.  Si aucune image n’est reçue alors elle modifie la recette sans changer l’image actuelle. |
| delete\_recipe | $idRecipe | Supprime la recette dont l’identifiant est en paramètre de la table trecipes |
| get\_recipe\_image | $idRecipe | Récupère le chemin d'accès à l'image de la recette dont l'identifiant est en paramètre.  Si le chemin récupéré correspond au chemin de l'image par défaut alors elle renvoie false.  Si le chemin récupéré ne correspond pas au chemin de l'image par défaut alors elle renvoie le chemin récupéré. |
| get\_ingredients |  | Récupère tous les ingrédients de la base.  Renvoie un tableau de données avec tous les ingrédients de la base |

|  |  |  |
| --- | --- | --- |
| **Fonction** | **Paramètres** | **Description** |
| count\_ingredient\_recipe | $idRecipe | Récupère le nombre d'ingrédients qui composent la recette dont l'identifiant est en paramètre  Renvoie le nombre d’ingrédients |
| checkExist\_ingredient | $IngredientName | Vérifie si l’ingrédient en paramètre existe dans la base.  Récupère l’identifiant de l’ingrédient en paramètre.  Si aucun identifiant n’est récupéré alors elle renvois faux car l’ingrédient n’existe pas.  Si un identifiant est récupéré alors elle le renvoie car l’ingrédient existe ; |
| add\_ingredient | $IngredientName | Insert un nouvel ingrédient dans la table ingrédients. |
| ingredients\_associate |  | Transforme un tableau de données en un tableau associatif.  Récupère un tableau de données qui contient la liste des ingrédients.  Transforme les valeurs du tableau dans un nouveau tableau avec comme valeur le nom de l’ingrédient et en index l’identifiant de l’ingrédient.  Renvoie le tableau associatif |
| get\_recipe\_types |  | Récupère tous les types de recettes.  Renvoie un tableau de données |
| recipe\_types\_associate |  | Transforme un tableau de données en un tableau associatif.  Récupère un tableau de données qui contient la liste des types de recettes.  Transforme les valeurs du tableau dans un nouveau tableau avec comme valeur le nom de l’ingrédient et en index l’identifiant de l’ingrédient.  Renvoie le tableau associatif |
| add\_contains | $idIngredient  $IngredientQuantity  $idRecipe | Insert un contenu (un ingrédient et une quantité) à la recette en paramètre dans la table tcontains |

|  |  |  |
| --- | --- | --- |
| **Fonction** | **Paramètres** | **Description** |
| get\_ingredients\_recipe | idRecipe | Récupère les contenus de la recette dont l’identifiant est en paramètre.  Renvoie un tableau de données. |
| get\_comments\_recipe | idRecipe | Récupère les commentaires postés sur la recette dont l’identifiant est en paramètre. |
| add\_comment | idUser  idRecipe  comment  note | Insert un nouveau commentaire dans la table tcomments. |
| delete\_user\_comments | idUser | Supprime les commentaires postés par l’utilisateur dont l’identifiant est en paramètre. |
| delete\_link\_user\_recipes | idUser | Supprime les liens entre l'utilisateur dont l’identifiant est en paramètre et les commentaires qu'il a posté  Remplace l’identifiant de l’utilisateur par un 0. |
| check\_owner\_recipe | idUser  idRecipe | Vérifie si l'utilisateur est le propriétaire de la recette.  Récupère l’identifiant de l’utilisateur qui représente le propriétaire de la recette dont l’identifiant est en paramètre.  Si l’identifiant récupéré correspond à l’identifiant en paramètre alors elle renvois vrai car l’utilisateur est le propriétaire de la recette.  Si l’identifiant récupéré ne correspond à l’identifiant en paramètre alors elle renvois faux car l’utilisateur n’est pas le propriétaire de la recette. |
| check\_owner\_comment | idUser  idComment | Vérifie si l'utilisateur est le propriétaire du commentaire.  Récupère l’identifiant de l’utilisateur qui représente le propriétaire du commentaire dont l’identifiant est en paramètre.  Si l’identifiant récupéré correspond à l’identifiant en paramètre alors elle renvois vrai car l’utilisateur est le propriétaire du commentaire.  Si l’identifiant récupéré ne correspond à l’identifiant en paramètre alors elle renvois faux car l’utilisateur n’est pas le propriétaire du commentaire. |
| **Fonction** | **Paramètres** | **Description** |
| delete\_comment | idComment | Supprime le commentaire dont l’identifiant est en paramètre |
| delete\_comment\_recipe | Recipe | Supprime les commentaires, notes qui ont été postés sur la recette dont l’identifiant est en paramètre |
| delete\_contains\_recipe | idRecipe | Supprime les contenus de la recette dont l’identifiant est en paramètre |
| show\_avg\_note\_recipe | AvgNote | Affiche 5 étoiles vides et/ou remplies (Affichage de la note)  L’étoile est remplie si la note reçue en paramètre est plus grande |
| upload | file | upload l’image en paramètre dans le dossier imgRecettes.  L'image est traitée afin de changer son nom.  Son nouveau nom est définit par la fonction uniqid(). |

# Tests

## Plan de test

Rapport de test

# Conclusion

## Améliorations

## Comparaison énoncé avec projet final