|  |  |
| --- | --- |
| 14/03/2018 | Documentation technique SP2 |
|  | Documentation technique du logiciel Alume Gestion pour l’entreprise Alume Menuiserie. |



SOMMAIRE

1. INFORMATIONS
   1. Descriptif gÉnÉral
   2. fonctionnement
2. modele conceptuel de donnees
3. diagramme de classe
4. specificites techniques
   1. architecture
   2. developpement du LOGICIEL Alume gestion

I) Information

Les spécifications fonctionnelles et techniques présenté ci-après ont pour but de fournir à l’entreprise Alume Menuiserie les informations nécessaires à la compréhension du fonctionnement du logiciel Alume Gestion en vue de son utilisation, sa maintenance et mise à jour ultérieurement.

Le dossier sera décomposé en plusieurs parties, vous trouverez en fin de dossier les annexes.

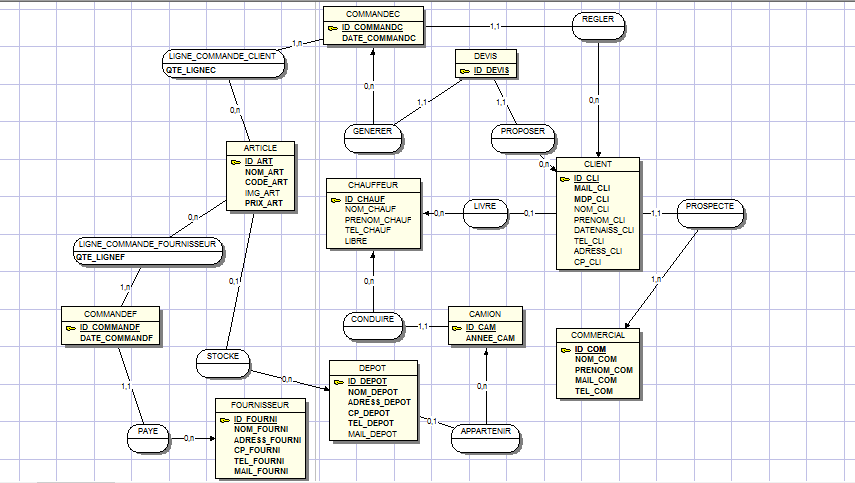
* + - 1. DESCRIPTIF GÉNÉRAL

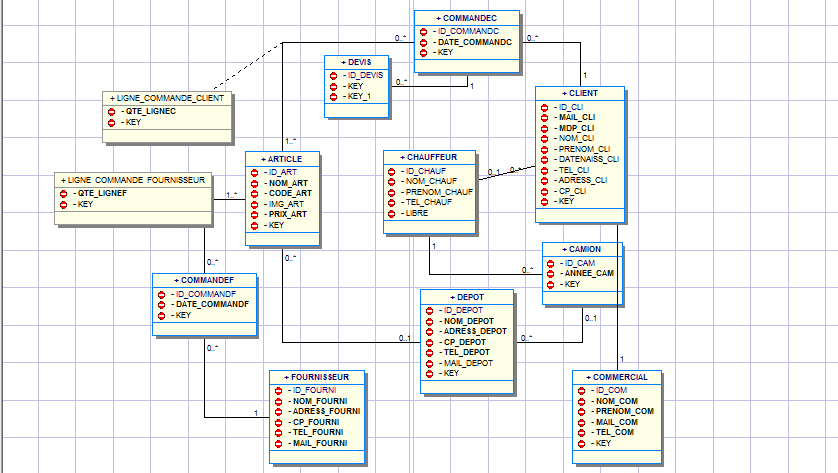
La mission principale du logiciel Alume Gestion est d’aider les commerciaux à mieux s’organiser et centraliser le flux d’informations pour faciliter le traitement de ceux-là. Le logiciel permet aux commerciaux de pouvoir accéder aux informations de leurs clients, de pouvoir gérer les taches logistiques tel que la gestion des stocks ou des chauffeurs disponibles pour la livraison. Le projet a été initié en 2018 par l’équipe développement de MAXCORP : GUVENDI Mumin, CHAT Tristan et JACQUART Anthony.

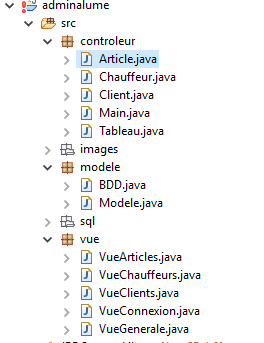
* + - 1. FONCTIONNEMENT

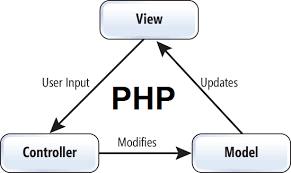
Les applications sont de type client / serveur et développées sous plusieurs langages, Java pour la partie applicatif et MySQL pour la gestion de base de données.

1. MODELE CONCEPTUEL DE DONNÉES



1. Idiagramme de classe
2. SPÉCIFICITÉS TECHNIQUES
3. architecture

Les applications tournent sous un serveur HTTP (dans l'exemple présenté ci-dessous, Apache). Elles ont été développées sous une architecture MVC qui veut dire Model View Controller en Anglais et Modèle Vue Contrôleur en Français.



Cette architecture permet une meilleure organisation et un développement structuré de l’application.

Un modèle contient les données à afficher.

Une vue contient la présentation de l'interface graphique.

Un contrôleur contient la logique concernant les actions effectuées par l'utilisateur.

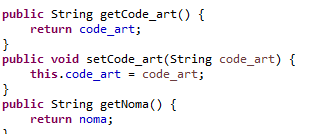
1. Développement Logiciel Alume Gestion

Nous allons étudier une partie de code expliquant le fonctionnement de l’application.

Nous allons analyser le code de la fenêtre client.

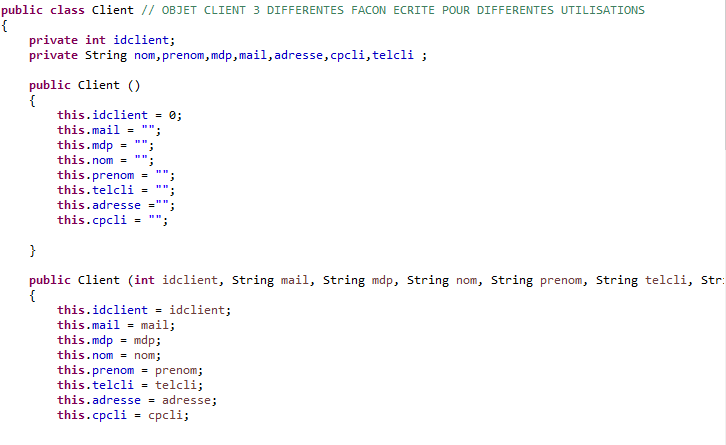
*Attentions Les getters et setters de l’application ne seront pas expliqués : ils ont pour but de récupérer les valeurs des pages pour les utiliser dans d’autres.*

*Ex :*

**

*Ils permettent d’utiliser le code\_article dans la page vue article.*

La classe Client



La classe client comporte tous les éléments relatifs à la base de données :

- L’id qui identifie le client

- son mail

- son mot de passe

- le nom

- Le prénom

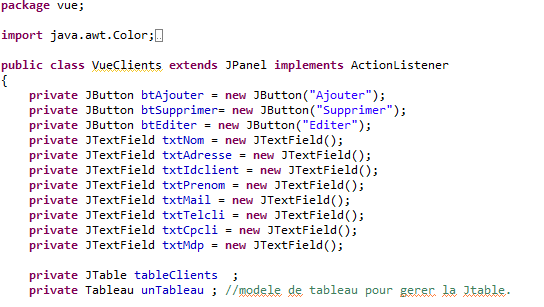
-le téléphone

- l’adresse et le code postal

Il faut savoir que cette base client et reliée à la même base de données que le site WEB <http://alumeproject.fr/> et les données sont mise à jour en direct.

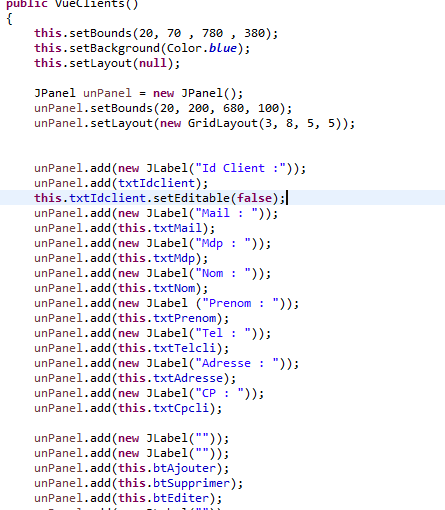
Nous avons 2 différents objets « Client » car ils peuvent être appelé de différentes façon dans les autres pages ( par exemple un avec l’id client à 0 et l’autre comme tel.)

La vue client



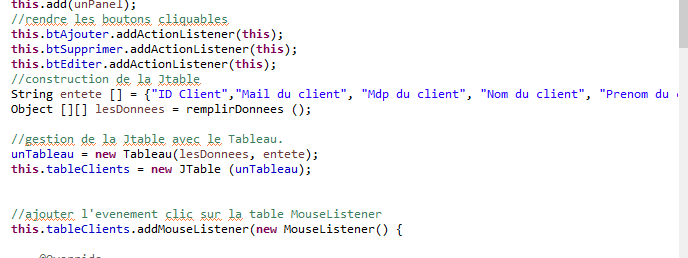
La vue du client est d’abord composé des lignes qui déclare l’utilisation des boutons Ajouter/supprimer/éditer. Ces codes sont nécessaires pour le bon fonctionnement et l’interactivité de ces derniers.

Ensuite on déclare les champs qui seront à compléter pour l’ajout ou la modification d’un client.

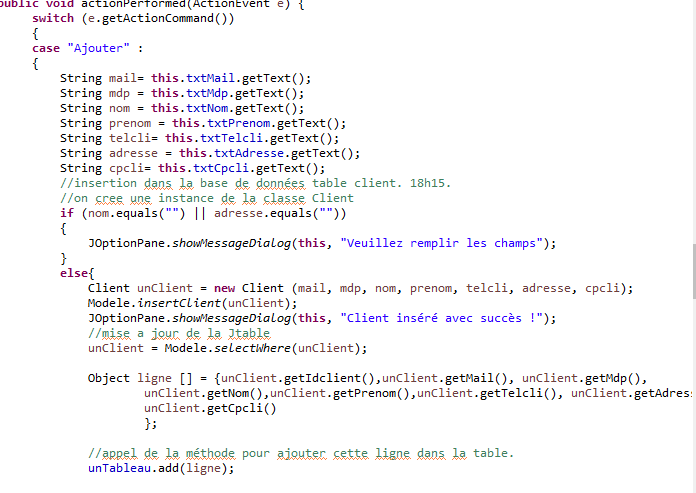


Ceci correspond à l’ensemble des éléments visuels ( Le background correspond au fond de l’application, (pour la page client la couleur bleu a été choisie )

Les panels correspondent aux cases /champs de l’application, le mot clé add permet d’en ajouter .



Cette screenshot parle d’elle-même elle est commenté pour comprendre au mieux son fonctionnement.

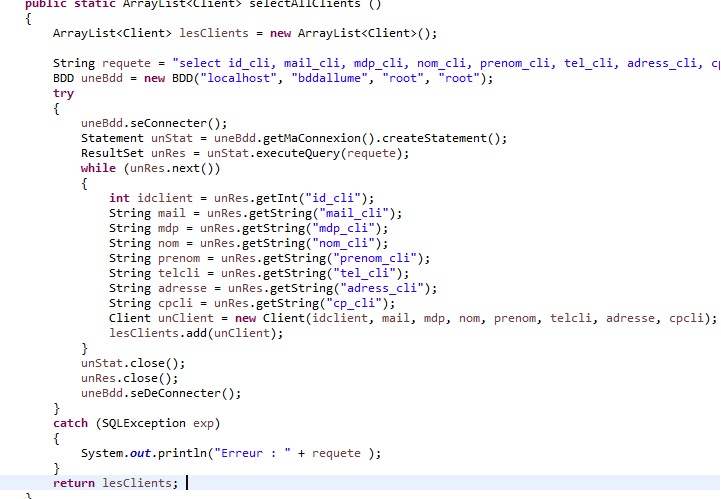


Cet extrait de code permet au logiciel de récupérer les valeurs saisie ^par le client et les insérer à la BDD en appelant la méthode insertclient.

On peut apercevoir la méthode selectwhere(unclient), celle-ci permet de « rafraîchir » la page afin de mettre à jour l’encadré des clients avec ce dernier récemment ajouté Pour supprimer ou éditer le procédé et similaire la vue appel juste les autres méthode deleteclient, editclient. Toutes ces méthodes précédemment citées seront expliquées juste ensuite.

Le plugin JDBC à été installer pour pouvoir connecter le client lourd à la base de donnée MYSQL Bddallulme : sans lui aucune action serai possible et il serai même impossible de se connecter !

Afficher les clients :



La fonction selectallClients() permet d’afficher les clients dans le logiciel, il effectue une recette SQL permettant d’afficher toutes les informations utile de la base de donnée concernant le client .

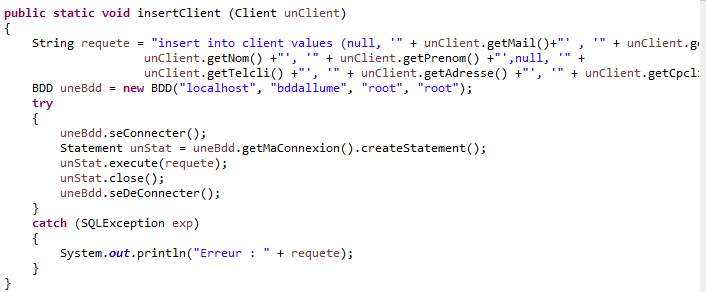
Les termes utilisés dans la requête SQL sont lié à l’objet CLIENT (Client.java dans le contrôleur).

Afin de manipuler au mieux l’objet.



Cette ligne permet la connexion à la base de donnée (ici en localhost pour la démonstration, c’est l’adresse du serveur qui sera web et les identifiants relatifs qui seront renseignés lors de la mise en productions de ces services.

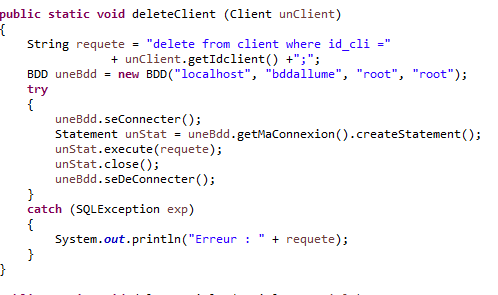
L’ajout de client



La requête SQL ici change il s’agit du terme INSERT INTO plutot que de SELECT.

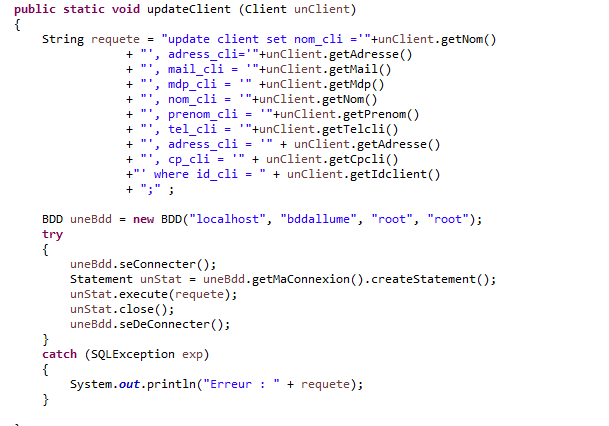
Les getter sont spécifiés afin de récupérée les valeurs renseignées par l’utilisateur dans les Textfield développés plus tôt. Une fois récupérées il les insert dans la requête SQL et la transforme avec les valeurs fournies.

LA suppression de client



La requête pour supprimer un client et beaucoup plus simple : au clique du client qu’on souhaite supprimé, il récupère l’id de celui-ci : et le getter se fait par l’id choisi on supprime donc le client avec l’id sélectionné.

L’édition du client :



Pour mettre à jour les clients de la base, les valeurs sont récupérées et remplacées par les celles saisies par le client. Ensuite intégrés à la requête SQL.

1. OUTILS UTILISÉS

Plusieurs outils ont été utilisés pendant le développement, GitHub a été utilisé en tant qu'outil de versioning. Les outils de versioning permettent d'avoir une vue sur les modifications effectués par les développeurs en charge du projet, GitHub est un outil libre et gratuit, avec une communauté de plus de 9 millions d'utilisateurs, c'est l'un des plus utilisé.



Pour le développement de l’application Alume Gestion, le logiciel Eclipse a été utilisé du fait qu’il est simple d’utilisation et intuitif.

