

IUT d'Arles

Année 2010-2011

Université de Provence

Section Imagerie Numérique 2<sup>ème</sup> année

SAHUC Cyril

# Rapport de Stage

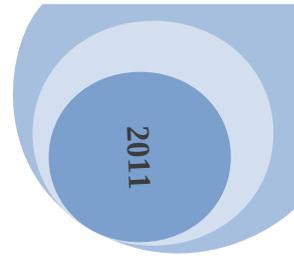
## OpenDébitBoisson

Réalisé à la Mairie d'Arles,

Au service de la DSIT

Tuteur : M. François RAYNAUD

Enseignant tuteur : M. Gilles GESQUIERE



# Rapport de Stage

## OpenDébitBoisson

## RESUME

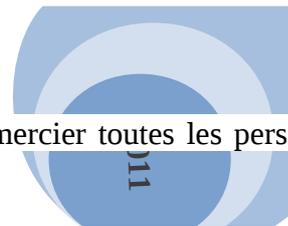
Ce rapport présente un résumé de l'expérience qu'a été pour moi mon stage de fin de 2ème année de DUT informatique. J'ai réalisé ce stage au sein de la mairie d'Arles dans le service de la Direction des Systèmes Informatique et Télécom, sur un poste de développeur. Ce stage, d'une durée de 10 semaines, m'a permis d'avoir une première approche dans le milieu du travail en tant que développeur. Durant celui-ci j'ai réalisé un travail sur un *logiciel open source\** dédié à faciliter les saisies de demandes de débit de boissons pour les collectivités. Ce poste m'amène à communiquer simultanément avec plusieurs services de la mairie d'Arles. D'un coté, les futurs utilisateurs afin de pouvoir définir au mieux leurs attentes, et de l'autre, différents services de la DSIT (en particulier le service développement et le service des réseaux), afin d'obtenir les connaissances nécessaires à la réalisation de ce logiciel. De plus, ce stage m'a permis d'acquérir une bonne expérience pratique dans le domaine de *l'open source\**, que je ne connaissais que de manière théorique.

**Mots clés** : *OpenMairie ~ Open Source ~ Framework ~ PHP ~ Développement orienté objet géolocalisation ~ SVN ~ OpenLayers ~ SQL ~ FPDF*

## SUMMARY

In this report, I will share with you the experience would have been for me my end of the second year of DUT. I made this internship at the mayor of Arles in the service of DSIT as a developer. This internship, lasting 10 weeks, I allowed a first approach the workplace as a developer. In this one I had to make a work dedicated to OpenSource facilitate seizures Request drinking place for communities. This post led me to communicate simultaneously with several departments of the mayor of Arles. On the one hand, the future users in order to best achieve their expectations, and other various services DSIT (especially the service development and service networks) to acquire the knowledge necessary to achieve this software. In addition, this internship allowed me to gain good experience in the field of open source, I knew only theoretically.

## REMERCIEMENTS

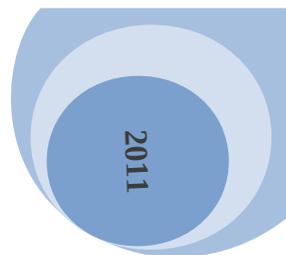


Pour débiter ce rapport de stage je tiens avant tout à remercier toutes les personnes sans lesquelles ce stage n'aurait jamais eu lieu :

- M. Hervé SCHIAVETTI, Maire de la Ville d'Arles et M. Patrick HAUTBOUT, Directeur Général des Services qui m'ont permis d'effectuer mon stage à la Direction des Systèmes Informatique et Télécom.
- M. Nicolas ISSART, Directeur de la DSIT, pour avoir accepté de me prendre en stage et de m'avoir mis en relation avec mon tuteur.
- M. François RAYNAUD, mon tuteur, qui m'a encadré tout au long de ce stage, s'est montré très pédagogique et m'a énormément appris.
- M. Jean-Louis BASTIDE, qui a secondé mon tuteur, m'a été souvent d'une grande aide, et a agréablement participé à la bonne ambiance dans laquelle a pu se dérouler le stage.
- Le personnel enseignant de l'IUT d'Arles qui m'a encadré tout au long de ma deuxième année de DUT, en particulier M. Gilles GESQUIERE, mon enseignant tuteur de l'IUT pour ce stage.

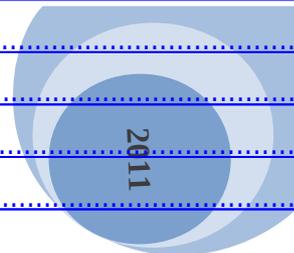
Je tiens aussi à remercier beaucoup d'autres personnes, qui, elles, sans avoir directement été liées au déroulement de mon stage, m'ont été d'une grande aide. Je remercie le personnel du service Infrastructure, qui m'a aidé pour la partie de mise en réseau de mon application. Je remercie aussi le personnel du service de Gestion Administrative, de qui j'ai partagé le bureau pendant ces dix semaines, pour m'avoir donné une atmosphère de travail toujours agréable. Je remercie l'ensemble de la DSIT pour leur accueil très chaleureux. Je remercie enfin Mme. MILCENT, testeuse et maintenant utilisatrice de mon logiciel, qui m'a guidé sur les points de vue juridiques et ergonomique de l'application.

## SOMMAIRE



<u>RESUME.....</u>	<u>3</u>
<u>SUMMARY.....</u>	<u>3</u>
<u>REMERCIEMENTS.....</u>	<u>4</u>
<u>SOMMAIRE.....</u>	<u>5</u>
<u>INTRODUCTION.....</u>	<u>7</u>
<u>LE STAGE.....</u>	<u>8</u>
<u>1) La DSIT.....</u>	<u>8</u>
<u>a) Description.....</u>	<u>8</u>
<u>b) Organisation.....</u>	<u>9</u>
<u>c) Encadrement du stage.....</u>	<u>9</u>
<u>2) OpenMairie.....</u>	<u>10</u>
<u>a) Caractéristiques :.....</u>	<u>10</u>
<u>b) Exemples d'applications utilisant OpenMairie :.....</u>	<u>10</u>
<u>3) OpenDebitBoisson.....</u>	<u>11</u>
<u>a) Principe.....</u>	<u>11</u>
<u>b) Méthodes :.....</u>	<u>11</u>
<u>c) Détail des principes :.....</u>	<u>11</u>
<u>DEROULEMENT DU STAGE.....</u>	<u>13</u>
<u>Etape 1 : Analyse du besoin .....</u>	<u>13</u>
<u>Etape 2 : Prise en main des ressources utilisées.....</u>	<u>14</u>
<u>Etape 3 : Conception, en php/sql et avec openLayers.....</u>	<u>15</u>
<u>Etape 4 : Première réunion avec l'utilisateur et présentation de la première maquette</u>	<u>16</u>
<u>Etape 5 : Améliorations avec les attentes précise de l'utilisateur.....</u>	<u>17</u>
<u>Etape 6 : Deuxième réunion avec l'utilisateur et présentation de la deuxième maquette</u>	<u>20</u>
<u>Etape 7 : Finalisation de la maquette et première version beta version 1.0.0beta.....</u>	<u>21</u>
<u>Etape 8 : Mise en place de la Version Béta sur la forge et installation en réseau pour</u>	
<u>phase de test de la version beta par l'utilisateur.....</u>	<u>22</u>
<u>Etape 9 : Dernières modifications, formation et implémentation de la version 1.0.0 sur</u>	
<u>le réseau de la ville, sur demo.openmairie.org et sur la forge.....</u>	<u>23</u>
<u>RENDU FINAL.....</u>	<u>24</u>

<u>ANALYSES DU TRAVAIL.....</u>	<u>26</u>
<u>1) Technique de travail.....</u>	<u>26</u>
<u>2) Compétences.....</u>	<u>27</u>
<u>a) Les compétences qui m'ont été utiles :.....</u>	<u>27</u>
<u>b) Les compétences que j'ai acquises ou perfectionnées.....</u>	<u>27</u>
<u>CONCLUSION.....</u>	<u>28</u>
<u>GLOSSAIRE.....</u>	<u>29</u>
<u>ATTESTATION DE PRISE DE CONNAISSANCE DU RAPPORT DE STAGE.....</u>	<u>30</u>



## INTRODUCTION

Il m'est donné pour la fin de mon DUT informatique à l'IUT d'Arles de réaliser un stage en entreprise d'une durée de 10 semaines. Ce stage a trois objectifs : valider les compétences acquises lors de mes deux années de DUT ; approfondir ces connaissances dans une mise en situation réelle sur un projet lié à ma formation ; se confronter, dans notre spécialisation, au milieu du travail.

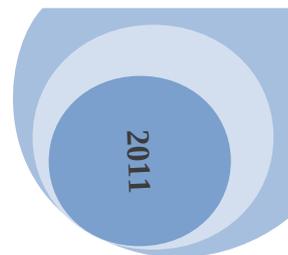
Je choisis de réaliser ce stage au sein du service de la DSIT (Direction des Systèmes Informatique et Télécom) de la mairie d'Arles. Ce service travaille en particulier sur des *logiciels open sources*\* destinés à être en libre accès pour toutes les mairies de France, voir même de l'étranger.

C'est ainsi que du 3 Avril 2011 au 10 juin 2011, je travaille auprès de mon tuteur de stage M. RAYNAUD François sur un logiciel nommé OpenDebitBoisson. Ce logiciel permet de faciliter les saisies et les vérifications de demande de licence pour les débits de boisson. Ce projet implique l'utilisation de leur *framework*\* OpenMairie (qui facilite la conception d'un logiciel de ce type, même lorsqu'on ne l'exploite que pour la première fois) ainsi que d'OpenLayer (permet de gérer des cartes de *géolocalisation*\*) et FPDF (génération de fichier au format PDF). Je travaille aussi par moment au côté de M.BASTIDE Jean-Louis, en particulier lors des utilisations de FPDF ou des développements de fonctionnalités sous OpenLayer.

Pour ce projet, un ordinateur et diverses documentations sur le *framework*\* d'OpenMairie me sont attribués. Ma première approche sur ce projet a été de consacrer une première partie brève du stage à prendre en main le *framework*\* utilisé (au côté de M.RAYNAUD, qui m'a consacré beaucoup de temps la première semaine) tout en faisant les diverses installations qui me seront nécessaires pour la conception sur mon poste (postgreSQL, postgis, pgAdmin, WampServer,...). Ensuite je prévois une première ébauche de conception me permettant mieux d'assimiler le *framework*\*, suivi d'une étude approfondie du besoin qui précède la conception spécifique d'OpenDebitBoisson.

Une annexe à ce rapport vous est fournie. Elle regroupe plusieurs impressions d'écran effectuées sur le logiciel OpenDebitBoisson, ce afin de vous permettre de mieux visualiser l'application. Cette annexe est principalement dédiée à la rubrique "RENDU FINAL".

## LE STAGE



### 1) La DSIT

#### a) Description

La DSIT, Direction des Systèmes Informatiques et Télécom, a été implanté en Arles en 1998. Son schéma directeur avait deux objectifs :

- créer un réseau haut débit unique pour la téléphonie et l'informatique de la mairie d'Arles.
- mettre en place des applications client serveur dans une ergonomie graphique permettant une plus grande autonomie des utilisateurs.

Son organisation subit deux grandes modifications au cours du temps, à savoir en septembre 2003 et en mai 2006, afin de mieux s'adapter aux nouveaux besoins et aux nouvelles techniques. Par exemple, pour mai 2006, réorienter le développement du Système d'information vers les technologies Web 2.0 et *l'open source\**.

En 2009 cette direction se caractérise par :

- 5 organisations
- 14 personnes
- 780 PCs Réseau + 300 PCs Education
- 8 serveurs de fichier
- 8 types de bases de données gérées
- 1 parefeu serveur + 750 parefeu clients
- 2 relais de messagerie
- 1.3 To d'espace disque utilisés pour la bureautique
- 3 logiciels de messagerie
- 46 sites connectés
- 1 réseau IP Backbone à 1 Gb
- 9 Systèmes d'exploitation
- 50 Applications métier

## b) Organisation

L'organisation de la DSIT se décompose en trois services :

- Le service Infrastructure (Réseau / Télécom, Système / base de donnée, et Support) composé de 9 personnes.
- Le service Etudes, Projets, Développement (Applications OpenMairie, Développement, ...) composé de 2 personnes.
- Le service Gestion Administrative (suit des contrats et des marchés, contrôle de gestion, suit des licences, ...) composé de 2 personnes.

M Nicolas ISSART, Directeur de la DSIT, coordonne ces trois services.

## c) Encadrement du stage

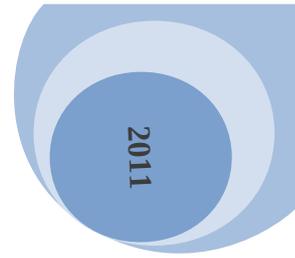
Je travaille pour mon stage de fin de DUT avec deux Techniciens Supérieurs du service Etude, Projet, Développement : M. François RAYNAUD et M. Jean-Louis BASTIDE. Ces deux personnes m'ont encadré tout au long de mon stage, en m'expliquant le fonctionnement d'OpenMairie et en m'orientant dans ma conception. Mon bureau se situe dans le service de Gestion Administrative, avec les deux personnes le composant. Mon travail m'a aussi amené à faire appel à d'autres membres de la DSIT comme M. Philippe ANTON, du service Infrastructure, pour les mises en réseau. J'ai pu donc être en relation avec l'ensemble des différents services de la DSIT.

De plus, mon sujet de stage m'a aussi amené à rencontrer le personnel de la mairie se chargeant des demandes de licence, comme Mme R.MILCENT, n'appartenant pas à la DSIT.

Un ordinateur de la mairie m'a été attribué pour ce stage. Celui-ci étant connecté au réseau de la mairie, j'avais accès à tout le matériel dont je pouvais avoir besoin, comme les imprimantes ou les accès au serveur de test.

## 2) OpenMairie

### a) Caractéristiques :



OpenMaire est un *framework*\*.

Son objectif est d'être une base commune aux applications métiers :

- Compatible avec toute les bases de données
- Customisable
- Compatible php 4 et 5
- *Orienté Objet*\*
- Permettant de lire le triple contrôle : client (javascript), serveur (php), et base de données.
- Permettant une ergonomie réactive est dynamique.

Il utilise des composants déjà existant.

- dbpear : connecteur de bases de données
- fpdf : génération de pdf
- OpenLayer : pour la gestion de cartes de *géolocalisation*\*

OpenMairie apporte une compatibilité complète entre diverses bases en stockant les requêtes dans des fichiers individualisés suivant chaque base de donnée et permet d'implémenter facilement des objets métiers définis dans l'analyse UML.

### b) Exemples d'applications utilisant OpenMairie :

- OpenElec : Gestion des listes électorales.
- OpenCimetière : Gestion des concessions d'un cimetière.
- OpenRésultat : Gestion des résultats électoraux.
- OpenCourrier : Gestion du courrier.
- OpenTel : Gestion des consommations téléphoniques.

- OpenPress : Gestion de la revue de presse.

### 3) OpenDebitBoisson

#### a ) Principe

Le projet OpenDebitBoisson a été demandé par le personnel enregistrant les demandes de débit de boisson. Son but est d'apporter une solution informatique pour ces demandes. Son principe repose sur trois critères :

- Pouvoir savoir de manière rapide et efficace qu'il n'y ait pas dans une demande de conflit de localisation ou de dénombrement.
- Permettre une génération automatisée des récépissés de demande de débit de boisson.
- Pouvoir donner un aperçu géographique des demandes de débit de boisson ainsi que des périmètres dans lesquels les demandes sont interdites.

#### b ) Méthodes :

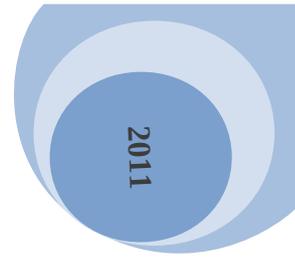
OpenDebitBoisson utilise le *framework*\* OpenMairie. Les différentes classes doivent être posées sur un diagramme de classe UML. Ensuite la base de donnée devra être créée et les procédures habituelles de codage par OpenMairie devront être suivies. Une fois ces procédures suivies, nous disposons d'un logiciel permettant de pouvoir accéder et modifier librement les éléments rentrés dans chaque table de cette base.

Il restera ensuite à intégrer les fonctions propres à cette application dans les différents fichiers générés par OpenMairie. Les langages principaux abordés seront le PHP, le javascript et le SQL.

#### c ) Détail des principes :

##### 1. Illégitimité d'une demande.

Les trois classes principales d'OpenDebitBoisson sont les établissements, les demandes, et les périmètres d'exclusion (exemple : église, cimetière, école, gendarmerie... ).



Il peut y avoir illégitimité de localisation uniquement dans les cas de demande de licence permanente, et qui ne sont pas simplement des transferts (renouvellement) si :

- L'établissement se situe dans un périmètre d'exclusion.
- l'établissement se situe à moins de 50 mètres d'un autre établissement possédant une licence permanente de catégorie 2, 3 ou 4.

Ensuite, toute demande temporaire confondue (géolocalisée ou non) admet une limite dans le nombre annuelle de demande. La limite est fixée à 5, mais peut monter jusqu'à 10 si elle concerne une demande temporaire liée à un terrain de sport (exemple : club de foot d'Arles).

## 2. Génération de récépissé

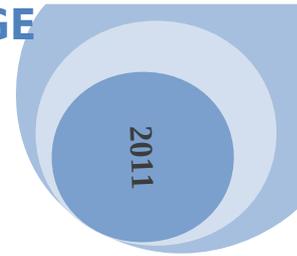
OpenDebitBoisson doit pouvoir se servir de toutes les caractéristiques rentrées dans une demande de licence ainsi que de toutes celles de son établissement demandeur afin de générer un PDF complet de récépissé de ladite demande. Pour les demandes de licence temporaire un autre PDF doit aussi être généré.

## 3. Aperçu géographique

Avec les méthodes implémentables sur OpenMairie il est simple de ressortir une carte géographique de représentation d'un type d'élément (comme les établissements) possédant une *géolocalisation*\*. Mais le système devrait être amélioré afin de pouvoir :

- confronter différents types d'éléments géolocalisables (établissements et périmètres d'exclusion).
- tracer des cercles dont le rayon correspondra parfaitement à ceux des périmètres d'exclusion quelque soit le niveau de zoom.

## DEROULEMENT DU STAGE



### **Etape 1 : Analyse du besoin**

Je prends connaissance de la note détaillant le sujet de stage. Ensuite j'évalue rapidement une approche du besoin de l'utilisateur. Ici, contrairement à ce que l'on rencontre fréquemment dans beaucoup d'autres projets, il ne faut commencer que par une approche du besoin. Le *framework*\* rend toute manipulation dans la base de données assez facile et rapide. Les contraintes des demandes de licence ne sont pas encore détaillées à ce moment là. Je fais donc un premier diagramme de classe temporaire qui résume les bases devant être implémentées pour le fonctionnement de ce logiciel. Je ne m'arrête pas à ce moment là sur les points précis à aborder. En effet, pour ce projet il m'est demandé d'utiliser une méthode de maquetage. Cette méthode consiste tout d'abord à raccourcir le temps généralement prévu pour la phase d'analyse afin de concevoir rapidement une première maquette. Sur cette première maquette l'utilisateur pourra déterminer les points à approfondir. Cette méthode repose sur un enchaînement de plusieurs cycles de maquette, desquelles s'ensuivent des tests effectués par l'utilisateur, qui lui nous amènera à faire des mises à jour afin d'aboutir à une maquette suivante, et ce jusqu'à arriver à la version finale.

## Etape 2 : Prise en main des ressources utilisées

J'installe sur mon poste tous les logiciels dont je vais avoir besoin pour le bon déroulement de mon projet. Suite à cela, M. RAYNAUD m'explique le fonctionnement du *framework*\* OpenMairie. Il me montre comment l'on procède pour implémenter des éléments qui vont être nécessaire au projet. Je commence à prendre en main le *framework*\* par une simple mise en place de la base de donnée initiale. Il faut l'implémenter de façon à ce qu'elle fonctionne à partir d' OpenMairie (à ce moment là, aucune fonctionnalité particulière ni de vérification d'entrée de donnée. Simplement la possibilité de rentrer des données depuis le logiciel sur la base de donnée et de les visualiser par des tableaux).

La gestion des fonctionnalités et des surcharges de fonction d'OpenMairie m'est devenue familière. J'ai à apprendre comment implémenter (toujours avec l'aide de mon tuteur) des fonds de cartes géographiques. Il faut aussi permettre au logiciel de récupérer, grâce à une base de données chargée au préalable, une *géolocalisation*\* à partir d'une adresse postale.

Il faut modifier certains de ces fichiers auto-générés afin de pouvoir ne visualiser que les données dont l'utilisateur à besoin. D'autres modifications de fichiers auto-générés servent à changer les formulaires de saisies de données et d'intégrer leur accès par le menu principal.

Des champs de type « géométrie » sont rajoutés afin de permettre la *géolocalisation*\* via postgre, pour les établissements et les périmètres d'exclusion. Certains champs non-visibles par l'utilisateur permettent, à l'aide de fonction dans le fichier de classe, de remplir cette *géolocalisation*\* lors de la saisie de formulaire.



### Quelques points de vue techniques :

Je commence par installer les logiciels suivant sur mon ordinateur : pgAdmin, PostgreSQL, PostGis, WampServer, Notepad++. Une fois cette action effectuer j'intègre la base de donnée d'OpenMairie sur pgAdmin, et le *framework*\* sur wamp.

Une fois ces actions effectuées il suffit de créer mes requêtes SQL afin de créer les tables de mon diagramme UML et leurs contraintes sous pgAdmin. Puis de générer ensuite grâce à OpenMairie les fichiers qui contiendront les fonctionnalités des classes associées, ainsi que bon nombre d'autres éléments dont se servira mon logiciel pour fonctionner.

### ***Etape 3 : Conception, en php/sql et avec openLayers.***

Je conçois sur les fonds de cartes générées la possibilité de visualiser des périmètres. Je choisis un cercle bleu transparent afin de représenter ceux-ci. Ces périmètres tracés serviront à représenter les périmètres d'exclusion dans lesquelles les établissements ne sont pas autorisés à faire de demande de licence. Par exemple, un établissement ne peut pas faire de demande de licence à moins de cent cinquante mètres d'une école ou d'un cimetière.

Sur ces mêmes fonds de carte, de petites gouttes d'eau représentent l'élément géolocalisé sélectionné. Lorsqu'on clique sur ces gouttes d'eau une vignette apparaît renseignant sur les éléments principaux de cet élément. Cette vignette contient aussi un lien ouvrant sur une page plus détaillée sur l'élément en question. Pour pouvoir faire une carte comparative entre les établissements et les périmètres d'exclusion, tous deux géolocalisés, il faut rajouter la possibilité que les fonds de carte puissent gérer plusieurs éléments simultanément. Et qu'ainsi les pages de détail de ces éléments s'ouvrent correctement sur le bon lien.

Au niveau des éléments de la base de donnée, les types d'établissements et les établissements d'exclusion -dont dépendaient les périmètres d'exclusion- sont abandonnés. Le périmètre d'un lieu dit et d'un rayon ne dépend pas forcément en soi d'un établissement.



#### Quelques points de vue techniques :

Au niveau des modifications les plus importantes sur le script d'affichage des cartes, codés en php et javascript et utilisant OpenLayer, un nouveau Layer est rajouté lorsqu'on indique à l'ouverture de la page que la variable `$GET['type']=zone`. Ce layer permet de faire apparaître les cercles bleu transparent délimitant les périmètres d'exclusion.

Ce script va lire un autre fichier qui va lui retourner l'ensemble des points à afficher et leurs données. Dans ces données ont été rajouté les rayons de chaque élément à afficher. Le script se sert de toutes ces caractéristiques afin de pouvoir faire apparaître, à l'aide d'un style défini, chaque élément par des cercles bleus dont le rayon correspond à leur rayon d'exclusion rentré en paramètre.

Dans le fichier retournant les données des éléments à afficher a aussi été rajouté l'identité de la classe auquel appartient cet élément. Ceci afin de permettre faire correspondre le bon lien vers la page détaillée lors de la sélection d'un élément, lors de l'affichage d'établissements et de périmètres d'exclusion simultanément.

## **Etape 4 : Première réunion avec l'utilisateur et présentation de la première maquette**



Une première rencontre avec Mme MILCENT a lieu au alentour de la fin de ma deuxième semaine de stage. Durant celle-ci j'ai pu être renseigné avec plus de précisions au niveau de la législation d'une demande de licence. Par exemple sur le fait qu'un établissement ne puisse pas non plus demander de demande de licence si les trois faits suivant sont réunis :

- Il s'agit d'une demande permanente d'ouverture ou de mutation
- Il s'agit d'une demande pour une licence de type 2,3 ou 4.
- Un établissement possédant une licence permanente de type 2,3 ou 4 se situe à moins de cinquante mètres de cet établissement.

J'apprends de plus, l'existence de type de demande (ouverture, transfert ou mutation) en plus des type de licences.

Durant cette réunion on parle aussi de la génération des pdf de récépissé de demande de licence permanente et temporaire, qui était prévue comme l'étape suivant les réalisations de fonds de carte adaptées à ce logiciel. Des exemples de ces pdf me seront envoyés quelques jours suivants la réunion.

Une erreur fit aussi surface pendant la cette présentation de maquette. Il faut savoir que trois fonds de carte différents peuvent être utilisés :

- avec OSM (OpenStreetMap, en local)
- avec Google Hybride (nécessite un accès à Internet)
- avec Bing (Nécessite un accès à Internet)

L'erreur en question concerne le tracé de périmètre. Celui-ci ne s'affiche pas lorsque l'on utilise un autre fond de carte que OSM.

Les demandes de licence non permanente peuvent être temporaires ou liées à un terrain de sport. Il n'y a aucune réglementation d'un point de vue localisation pour ces demandes de licence mais il y a une limite du nombre de ces demandes dans l'année pour un établissement donné : cinq par an pour les demandes temporaires, dix par an pour les demandes liées à un terrain de sport.

## **Etape 5 : Améliorations avec les attentes précise de l'utilisateur**

La première amélioration que je choisis de traiter est un affichage correct des périmètres quelque soit le type de fond de carte. Aussi je remarque à ce moment là que les distances tracées sous OSM n'étaient pas si précises que cela. J'adapte un coefficient multiplicateur à ce rayon afin que, pour une distance donnée de trois milles mètres, l'imprécision soit d'un mètre. Ce logiciel ne devant traiter que des périmètres d'exclusion de maximum cent cinquante mètre, une imprécision de cinq centimètres ne sera pas visible. Pour ce qui en est du pourquoi les autres fonds de cartes n'affichaient pas les périmètres, le problème vient d'une variable appelée « zoom », dont je me servais afin que les tracés de périmètres correspondent à l'échelle de la carte quelque soit le niveau de zoom. Mais OSM utilise une échelle de zoom différente de Google Hybrid et Bing. Ce souci ne faisait pas disparaître les périmètres mais les rendait tellement petits qu'il n'étaient pas visibles sur les cartes. Pour répondre à ce problème il suffit de créer une fonction permettant de retourner toujours la correspondance en « zoom d'OSM » quelque soit le fond de carte utilisé.

Aux cartes ont été ajouté la possibilité d'afficher plusieurs images différentes pour mieux différencier les différents types d'éléments. Je conserve la gouttelette d'eau pour les établissements et intègre un point d'exclamation pour les périmètres d'exclusion.

Les contraintes aux niveau des demandes de licence temporaires ont été assez simplement ajoutées.

L'élément « type de demande » est ajouté, ainsi que la possibilité que le « terme » d'une demande de licence soit liée à un terrain de sport (équivalent au terme temporaire, mais avec quelques variations).

De la même façon que l'on vérifiait qu'une demande de licence ne concernait pas un établissement inclus dans un périmètre d'exclusion, l'on vérifie en plus si les conditions sont réunis (demande de licence 2,3 ou 4, permanente, d'ouverture ou de mutation) qu'il ne se situe pas à moins de cinquante mètre d'un autre établissement ayant déjà une licence de type 2,3 ou 4.

Durant la quatrième semaine je m'occupe des générations automatiques de récépissé au format PDF. Jean-Louis BASTIDE, travaillant lui aussi au service de la DSIT de la mairie d'Arles et ayant travaillé sur FPDF m'explique les méthodes de génération. Une grande partie de ces générations sont maintenant automatisées mais il me reste un souci : certain PDF

devraient avoir deux pages, les générations de PDF avec les éléments dont je dispose ne me permette pas de rendre contrôlable l'endroit de la coupure entre ces deux pages. Après quelques recherches sur Internet je trouve une méthode utilisant FPDI, une extension de FPDF permettant la concaténation de deux PDF de façon assez simple. A ce moment là je choisis donc de lancer lors du click de l'utilisateur sur la génération automatique de PDF, deux générations, et de l'envoyer sur une page contenant la concaténation des deux. Cette méthode, même si opérationnelle, a malgré tout quelques inconvénients et mon tuteur me lance sur une autre piste plus simple utilisée dans d'autres logiciels d'OpenMairie utilisant '\p' pour faire un saut de page. Je reprends cette méthode en l'intégrant dans le code source d'OpenDebitBoisson afin que l'expression '\p' fonctionne aussi, et pouvoir ainsi l'utiliser pour mes propres générations de récépissé.



#### Quelques points de vue techniques :

##### 1. Pour régler le rayon tracé :

Je me suis servi de deux périmètres que je créé moi-même, très distant l'un de l'autre (environ trois kilomètres). Je lance manuellement par pgAdmin une requête SQL permettant de savoir la distance exacte entre ces deux points. Je rentre en paramètre dans l'un des deux périmètres d'exclusion cette distance en tant que longueur d'exclusion. Je coefficiente correctement (par tâtonnement) le rayon de cercle tracé afin que celui-ci, ayant pour centre un des deux périmètres, passe par le second.

##### 2. Pour le zoom :

La différence du niveau de zoom entre le fond de cartes OSM et les autres est de 10. Une valeur de zoom sur OSM de 18 correspond parfaitement à une valeur de zoom sur Hybrid ou sur Bing de 8. Je fais donc simplement une fonction qui retourne la variable zoom si l'on utilise OSM, ou retourne zoom+10 sinon, et mes réglages effectués au préalable pour les distance de rayons sont toujours corrects.

3. *Pour les images représentatives des éléments différents :*

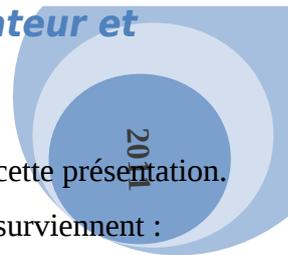
De la même façon qu'ont été implémentés les rayons lors de l'étape 3 dans le fichier qui retourne les points à dessiner sur la carte, je rajoute une nouvelle variable qui retournera le nom de l'image correspondante à cet élément. Ce fichier se sert d'une requête SQL faisant la jointure entre certaines données des tables « etablissement » et « perimetre ». Il suffit de rajouter à cette jointure une colonne prenant le nom d'une image pour la partie établissement et un nom différent d'image pour la partie périmètre jointe.



4. *Au niveau de la concaténation PDF :*

L'inconvénient majeur qui fait que cette solution n'est pas très envisageable est que les PDF devaient se créer une première fois physiquement sur le serveur avant d'être concaténés. Cela poserait beaucoup de soucis complexes, comme veiller à la suppression de ces fichiers temporaires, les cas où deux personnes génèrent des PDF simultanément, les droits de suppressions lorsque le PDF lu et concaténé n'est pas fermé, etc... La solution visant à inclure un '\p' comme étant un symbole de saut de page est beaucoup plus simple.

## ***Etape 6 : Deuxième réunion avec l'utilisateur et présentation de la deuxième maquette***



Les modifications apportées lors de l'étape 5 sont validées lors de cette présentation.

D'autres problèmes, ou plutôt possibilités d'améliorer le logiciel, surviennent :

- Les établissements ne demandant que des licences temporaires, ne pouvant pas vraiment être qualifiés d'établissements en soi, n'ont nul besoin d'être géolocalisables. Ainsi il serait intéressant de pouvoir différencier les établissements sujet à des demandes permanentes de ceux ne fixant que des licences temporaires non-géolocalisées.
- Une classe « courrier » serait intéressante afin de pouvoir saisir manuellement et enregistrer des courriers particuliers pour les établissements, autres que les récépissés ou demandes de licence temporaire habituels.
- Les demandes temporaires s'effectuent parfois pour une seule journée suivant un créneau horaire bien délimité. Aucune zone permettant de rentrer les heures n'est disponible dans les demandes de licence, il faudra donc en rajouter une et intégrer ce champs dans les générations de récépissé de demande de licence temporaire.

## ***Etape 7 : Finalisation de la maquette et première version beta version 1.0.0beta***

L'élément établissement admet à présent une sous-catégorie « établissement temporaire » qui définit un établissement non-géolocalisé ne pouvant effectuer que des demandes de licence temporaire. Lors de la sélection de celui-ci, l'ancien onglet « demande de licence » devient « demande de licence temporaire » qui lui-même est aussi une sous-catégorie du précédent. Dans ces deux sous-catégories, certains champs n'apparaissent pas car inutiles pour ces cas-ci, et certaines sélections sont rendues impossibles (exemple : la saisie d'une demande contient une zone de sélection pour rentrer le « terme » de la demande de licence. Lors d'une demande de licence temporaire, le choix « permanente » n'est pas disponible.).

L'élément courrier, appartenant à un établissement donné, est créé. Il dispose d'une zone de saisie de texte et d'un objet. Il faut intégrer une génération de PDF pour pouvoir visualiser les courriers saisis. Le courrier prend directement les données de l'établissement comme adresse.

Pour les demandes de licence apparaît la nouvelle zone de saisie « heure de début » et « heure de fin ». Ces deux zones, si non-remplies, se remplissent automatiquement par de 00 :00 à 23 :59. Elles n'apparaissent que sur les récépissés de demande de licence temporaire.

Suite à ces modifications, ainsi qu'à une réparation de petits soucis moins significatifs qui pouvaient survenir sur le logiciel, on valide la version bêta.



### *Quelques points de vue techniques :*

Au niveau de la base de donnée une seule classe « établissement » existe. Une colonne « nature » pouvant contenir soit le terme « permanente » soit le terme « temporaire » a été créée. L'utilisateur lui de son côté ne peut accéder qu'à l'une ou l'autre comme deux classes distinctes mais il ne s'agit au niveau du code proprement dit que d'une surcharge de classes afin que ces deux-ci soient différentes, dans leurs saisies et leurs présentations.

## ***Etape 8 : Mise en place de la Version Béta sur la forge et installation en réseau pour phase de test de la version beta par l'utilisateur***



La version bêta est lancée sur le site <http://www.adullact.net>.

Un serveur de la mairie est mis à ma disposition afin de pouvoir tester mon application en réseau. Une fois ce serveur en ma possession je réplique le même procédé que pour l'étape 1, à savoir l'installation nécessaire au fonctionnement de mon logiciel. Seulement un petit souci survient quant à l'une de ces installations nécessaires. WampServer, un logiciel essentiel à la mise en réseau d'OpenDebitBoisson, ne veut pas fonctionner correctement. C'est sûrement dû à une incompatibilité avec Windows Server 2000. Après beaucoup d'essais non concluants, j'abandonne l'installation de WampServer et essaie avec un logiciel permettant les mêmes fonctionnalités : EasyPHP. Je fais ensuite plusieurs essais et corrections sur ce même serveur pendant quelques jours afin de détecter les moindres erreurs qui resteraient encore présentes sur le logiciel. Quelques modifications d'ordre mineur ont été faites durant cette période.



### Quelques points de vue techniques :

Problème de WampServer : WampServer en tant que tel fonctionnait. Mais après l'ajout de la ligne « extension=php\_pgsql » du php.ini contenu dans le document d'apache pour utiliser postgres, le redémarrage d'apache ne se lance pas. Le service apache reste sur « Démarrage » sans jamais s'activer (lorsque le service est en fonction il affiche « démarré »). Un conflit doit avoir eu lieu entre WindowsServeur2000 et l'ajout de l'utilisation de postgres à l'utilisation du service apache car en employant exactement les mêmes procédés hors serveur tout fonctionne parfaitement.

## ***Etape 9 : Dernières modifications, formation et implémentation de la version 1.0.0 sur le réseau de la ville, sur [demo.openmairie.org](http://demo.openmairie.org) et sur la forge***

2011

La version mise en réseau est mise à disposition du personnel de la ville d'Arles. Je leur fourni le manuel de l'utilisateur que j'ai rédigé, et passe une après midi à les former à l'utilisation de ce programme. Cet après midi de formation m'a permis de pouvoir voir lors de l'utilisation réel du logiciel certaine préférence des utilisateurs auxquelles je pouvais facilement répondre. Le lendemain donc je me charge de faire rapidement ces dernières modifications sur le serveur afin qu'ils puissent au plus tôt disposer de toutes les fonctionnalités souhaitées pour ce logiciel. Certains changements très mineurs surviennent encore de temps à autre comme l'agrandissement d'un champ légèrement trop petit.

Aujourd'hui la version 1.0.0 d'OpenDebitBoisson semble stable et parfaitement convenir aux attentes de l'utilisateur. Cette version va être validée comme n'étant plus une version bêta sur la forge d'addulact, et sera mise sur le serveur de démo en lien avec cette forge.

## RENDU FINAL

Cette partie du rapport explique brièvement le fonctionnement général, d'un point de vue utilisateur, d'OpenDebitBoisson. Cette explication se limite à l'essentiel du fonctionnement du logiciel. Les fonctionnalités les moins importantes ne seront pas vues. Pour cette partie je vous conseille d'ouvrir le livre d'annexes fournis avec ce rapport de stage afin d'avoir la meilleure visualisation possible quant au fonctionnement de cette application.

L'utilisateur demande tout d'abord à un administrateur de lui créer un compte pour pouvoir se connecter. Une fois connecté il accède directement à un tableau de bord, vierge tout d'abord, auquel il pourra intégrer diverses petites fenêtres de renseignements ou de raccourcis vers d'autres pages.

Il pourra à ce moment là, à l'aide du menu de gauche, accéder aux saisies d'établissements ou de périmètres d'exclusion. Ces onglets le mèneront vers une page regroupant dans un tableau tout les éléments déjà enregistré, et lui permettra de les sélectionner, d'en ajouter, d'en supprimer ou même d'avoir un aperçu géographique sur leur *géolocalisation\**.

La sélection d'un établissement, tout comme celle d'un périmètre, amène l'utilisateur vers une page regroupant toute les informations dédiées à cet élément, nous permet de les modifier. Pour ces deux types d'éléments, on permet à l'utilisateur de saisir un début d'adresse, voir d'une adresse exacte, et à l'aide d'un système d'adresse postale la *géolocalisation\** de l'établissement ou du périmètre est retrouvée, ou dans le cas où plusieurs adresses sont possibles il lui est amené à choisir la bonne parmi toutes les adresses possibles.

De plus, la sélection d'un établissement ou d'un périmètre laisse aussi apparaître de nouveaux onglets sur la partie supérieure. Ces onglets correspondent aux différents types d'éléments dépendants de celui sélectionné. Lors de la sélection d'un établissement apparaîtra les onglets "demande de licence", "dossier" et "courrier".

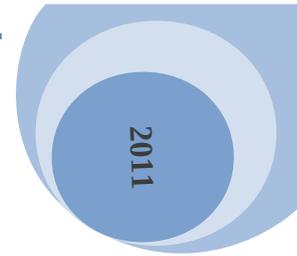
C'est dans l'onglet "demande de licence" que l'utilisateur pourra saisir une demande de licence pour un établissement donné. L'onglet "demande de licence", comme les autres onglets d'ailleurs, nous amène, comme lors d'un click sur le menu "Etablissement" à un tableau récapitulatif des éléments contenus dans la base de donnée. Avec la différence que seuls les éléments (par exemple les demandes de licence) ayant pour établissement l'établissement sélectionné, apparaîtront.

Les actions permises à l'utilisateur dans chaque onglet ou barre de menu ne sont pas toujours identiques. Dans toutes les catégories ici citées, pour un utilisateur, il sera possible de modifier les éléments, les supprimer ou en ajouter. En particulier dans la catégorie "demande de licence" il sera possible en plus de générer les PDF de récépissé ainsi que les PDF de demande temporaire. De même pour la catégorie courrier il sera possible de générer un PDF du courrier en question.

A l'ajout ou à la modification d'une demande de licence, les vérifications de localisation ainsi que d'énumération se font directement et si un conflit survient, un message prévient l'utilisateur et le renseigne sur l'exactitude de la source du conflit. Par exemple s'il entre en conflit avec un établissement ayant une licence permanente trop proche de lui, le nom de l'établissement, sa catégorie de licence, la distance exacte séparant les deux établissements et un lien vers les caractéristique de cet établissement posant problème lui seront retourné. La demande de licence par contre ne sera pas interdite et s'enregistrera malgré tout. Après étude du conflit en question si l'utilisateur souhaite annuler la demande il devra la supprimer.

Sur le tableau de bord un lien permet d'avoir directement accès à une carte géographique regroupant tous les établissements ainsi que tous les tracés de périmètres d'exclusion afin d'avoir un meilleur visuel sur la situation. On peut accéder de la même façon à ces cartes géographiques par un lien contenu dans les tableaux regroupants les éléments géolocalisable (ceux accéder par le menu, les établissements et les périmètres d'exclusion, comme expliqué précédemment.). Lorsqu'on accède à la carte par ces liens là par contre seul le type de l'élément sélectionné sera représenté, à savoir soit les établissements, soit les périmètres. Par contre, par cette même carte, l'élément sélectionné pourra être manuellement déplacée, si la *géolocalisation\** automatique par adresse postale n'a pas été assez précise, ou pour faire une *géolocalisation\** manuelle.

## ANALYSES DU TRAVAIL



### 1) Technique de travail

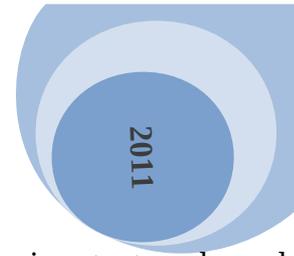
L'évolution de la conception se faisait avec tortoise SVN afin de partager toujours les mêmes fichiers sources avec mon tuteur M. RAYNAUD François.

Pour ce projet il n'y eu pas par contre de longues analyses avant la conception comme dans beaucoup de projets, simplement un bref diagramme de classe UML afin de ne pas prendre de mauvaises orientations au départ de la programmation. Ce fut un choix. Voici quelles en sont les raisons :

- Gain de temps très conséquent. Une analyse du besoin aurait été trop longue par rapport à la durée du stage.
- L'utilisation d'OpenMairie permet une certaine aisance à pouvoir modifier les tables de la base de donnée sans trop de soucis dans le cas où l'on devrait revenir sur un choix effectué. Entre le début et la fin de la conception, les tables (classes d'élément) ne sont plus vraiment les mêmes, deux ont été supprimées et deux nouvelles sont apparues, et cela sans posé de soucis.
- Allez retour développeur / utilisateur.

## 2) Compétences

### a) Les compétences qui m'ont été utiles :



- Mes connaissances en *programmation orienté objet\** acquises tout au long de mes deux années d'iut m'ont été très utiles pour ce projet. En effet même si l'utilisation de *framework\** m'était auparavant inconnue je n'ai pas eu de mal à l'assimiler, ne reposant au final, en grande partie que sur des surcharges de fonction.
- Le php vu durant cette deuxième année, dont en particulier un projet semblable à un "facebook" simplifié, m'a aussi été d'une grande aide. J'ai donc travaillé ici dans un langage où je me sentais à mon aise, sans avoir vraiment d'assimilation nouvelle à faire au niveau du code. De même pour le langage SQL qui lui a été présent durant les deux années.

### b) Les compétences que j'ai acquises ou perfectionnées.

- l'utilisation d'un SVN, utilisé rapidement durant cette année pour mon projet tuteuré mais pas réellement exploité au maximum. Le travail par SVN avec M. RAYNAUD François pour partager nos ressources m'a vraiment montré l'importance de l'utilisation d'un tel procédé.
- Même si ce langage m'était déjà familier, le fait d'avoir pratiqué du php pendant 10 semaines, ainsi que de l'avoir ici utilisé de façon *orienté objet\**, m'a permis d'approfondir vraiment mes connaissances dans ce langage.
- OpenLayers, qui m'était totalement inconnu à mon arrivé à la mairie d'Arles. Je ne maîtrise pas vraiment cette bibliothèque encore mais j'arrive à lire, comprendre et modifier du code source l'utilisant.
- Je n'ai pas eu à faire d'installation de serveur mais quelques problèmes que j'ai eu en fin de stage lors de l'implémentation du logiciel sur le réseau de la mairie d'Arles m'ont fait apprendre quelques astuces dans ce domaine qui m'étaient inconnues.

## CONCLUSION

Ce stage s'est donc déroulé dans de bonnes conditions. Le travail fut vraiment plaisant dans ce service. J'ai été très bien suivi tout au long de mon stage et les personnes qui m'ont encadré ont pu me fournir toutes les réponses techniques nécessaires à mon projet. J'ai pu approfondir mon savoir faire dans le domaine du php et du sql, ainsi qu'apprendre de nombreuses notions comme l'utilisation de *framework\** (ici OpenMairie), d'OpenLayer, et de FPDF. Aussi ai-je pu, en cette fin de DUT informatique, grâce à ce stage, comprendre l'importance du travail en équipe pour aboutir à un bon résultat.

Le service de la DSIT m'a permis de réaliser un logiciel libre, où j'apparais en tant que développeur. Ce logiciel pourrait très bien être utilisé à l'avenir par bon nombre de mairie française.

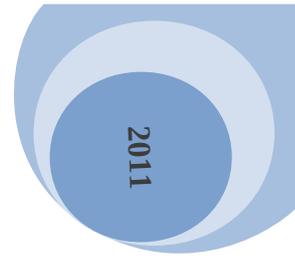
Il est vrai que la création et l'utilisation de *framework\**, facilite énormément la phase de conception. Quelques inconvénients pourraient en découler : la création du *framework\** lui-même, qui fut réalisé par une équipe de développeur dont M. RAYNAUD, et le choix à chaque nouvelle intégration de fonctionnalité de l'intégrer de façon spécifique au logiciel conçu sans toucher à la partie « mère » ou de l'intégrer dans le *framework\** d'origine. Lorsqu'on réalise plusieurs logiciels semblables, même si ayant des fonctionnalités différentes, comme c'est le cas au service de la DSIT, la perte de temps engendré par l'apprentissage du *framework\** est largement rattrapée par le gain de temps que nous apporte celui-ci lors des conceptions de logiciel. Il est donc très intéressant d'y avoir recourt dans ce cas présent.

Ainsi, OpenDebitBoisson est opérationnel à la mairie d'Arles, plus précisément utilisé par Mme MILCENT et Mme REMI. Plusieurs mises à jour auront été nécessaires même après son lancement sur serveur. Actuellement la version utilisée répond au besoin.

De mon point de vue personnel ce stage m'a pleinement satisfait et a été pour moi une réussite, tant au niveau de l'apprentissage qu'il m'a apporté que par la satisfaction d'avoir pu apporter une solution informatique efficace à un problème dans un domaine m'étant assez étranger.

La mission est remplie.

## GLOSSAIRE



### \*Framework :

Un framework est un kit de composants logiciels qui servent à créer les fondations ainsi que les grandes lignes de tout ou d'une partie d'un logiciel (architecture). En *programmation orientée objet\**, un framework est typiquement composé de classes mères qui seront dérivées et étendues par héritage en fonction des besoins spécifiques à chaque logiciel qui utilise le framework.

### \*Géolocalisation :

Technique de détermination de la localisation géographique précise d'un lieu.

### \*Logiciel open source :

Logiciel libre, que chacun peut utiliser, copier, modifier ou distribuer à sa guise.

### \*Programmation orientée objet :

Mode de programmation dans lequel les données et les procédures qui les manipulent sont regroupées en entités appelées objet.

## ATTESTATION DE PRISE DE CONNAISSANCE DU RAPPORT DE STAGE



Je soussigné M. François RAYNAUD atteste avoir pris connaissance de ce rapport de stage réalisé par l'élève Cyril SAHUC, dépendant de l'IUT d'Arles, qui fut sous ma tutelle pendant tout le déroulement du stage.

Signature :