

Rapport de stage pour le diplôme de
Développeur Logiciel

Mabiala Alban accueilli par SOZO unltd
du 20/06/2016 au 19/08/2016

Document confidentiel

SŌZŌ UNLTD	Rapport de Stage Développeur logiciel	page 2/57
----------------------	--	-----------

Remerciements

Je souhaiterais exprimer toute ma gratitude à M. Sothera KHEM, mon maître de stage et dirigeant de SOZO unltd. Je le remercie pour son attitude, ses conseils et sa bienveillance pour la réalisation de ce projet.

Je remercie également l'équipe de ETERNAL de m'avoir accueilli au sein de ses locaux. Je remercie toutes les personnes qui m'ont encouragé dans la réalisation de mon projet de formation, celles m'ont soutenu et motivé pour l'obtention de ce diplôme ; ma compagne, ma famille, mes amis, mes formateurs, mes confrères stagiaires.

Avec une attention particulière pour mon formateur et tuteur M. Pascal BUGUET, dont les l'encadrement, les conseils et la patience ont été d'une aide plus que précieuse.

Sans oublier,



Sommaire

1. Introduction	5
1.1.1. La société	6
1.1.2. L'équipe :	6
1.1.3. Concept:	7
1.1.4. Services proposés :	7
1.1.5. Références :	7
1.1.6. Réalisations :	7
1.1.7. Environnement techniques :.....	8
1.1.7.1. Environnement de développement.....	8
1.1.7.2. Plate forme locale s.....	8
1.1.7.3. Serveur de production.....	8
2. Étude de cas : conception d'une application mobile.....	9
2.1.1. Contexte.....	9
2.1.2. L'application.....	10
2.1.3. Contraintes et environnement	10
2.1.4. Objectif :	11
2.1.5. La solution.....	11
2.1.5.1. Description.....	11
2.1.5.2. Avantages	13
2.1.5.3. Limites	13
2.1.6. Développement	13
2.1.7. Langages et outils utilisés	13
3. Conception.....	14
3.1. Démarche	14
3.2. Analyse des besoins	15
3.2.1. Exigences fonctionnelles.....	15
3.3. Modélisation.....	16
3.3.1. Diagramme de contexte statique.....	16
3.3.2. Diagramme de cas d'utilisation.....	17
3.3.3. Fiche descriptive des cas d'utilisation (exemple)	18
3.3.4. Diagrammes de séquence	19
3.3.5. Diagramme de navigation.....	21
3.3.6. Maquettes	22
3.3.7. Classement des cas d'utilisation.....	24
3.3.8. Planning.....	25

3.3.9.Charte graphique	26
4.Analyse et conception des données.....	27
4.1.Les règles de gestion :.....	27
4.2.Dictionnaire des données	28
4.3.Model conceptuel de données (MCD).....	29
4.4.Modèle logique des données (MLD).....	30
4.5.Script de création de la BD.....	30
4.6.Modèle physique de données (MPD).....	31
4.7.Diagramme de déploiement.....	32
5.Code.....	33
5.1.backoffice.....	33
5.2.Mobile.....	44
6.Sécurité.....	47
7.Conclusion.....	48
8 Annexes.....	49
8.1.1. Script de création de la base de données.....	49
8.1.2.Les langages (description).....	52
8.1.3. Les outils (détails).....	53
8.1.4.Glossaire.....	54
8.1.5.Webographie.....	54
8.1.6.Bibliographie.....	54
8.1.7.Diagramme de Gantt.....	55

SŌZO UNLTD	Rapport de Stage Développeur logiciel	page 5/57 <i>Introduction</i>
----------------------	--	----------------------------------

1. Introduction

Le marché du mobile est en évolution constante depuis plus de 15 ans !

Longtemps réservés aux milieux professionnels, les téléphones portables se sont peu à peu démocratisés pour devenir de formidables outils de communication à la pointe de la technologie.

Aujourd'hui avec un « Smartphone », il nous est possible de télécharger des applications pour se divertir ou travailler, de se géolocaliser, de communiquer également par courriels ou de créer et partager ses contenus multimédias : images, vidéos et musiques... et les usages ne cessent d'évoluer !

Notre téléphone fait désormais partie de notre quotidien, il a modifié notre rapport à la technologie. Le fait de pouvoir installer des applications sur nos « Smartphones » nous a permis de prolonger cette expérience et nous a ouvert de nouvelles voies en terme de services.

Cette évolution a permis l'émergence de nouveaux services et nouveaux concepts adaptés à une clientèle de plus en plus large et hétérogène. De plus, avec le développement des infrastructures réseau (WiFi, 4G, hotspots) proposées par les opérateurs téléphoniques, il nous est possible d'utiliser nos applications de façon permanente partout dans le monde. C'est dans ce contexte qu'intervient mon projet de stage de fin de cycle.

Ce stage au a été entrepris à l'issu de ma formation de (?) heures dispensée par l'organisme Formation M2i pour l'obtention du titre développeur logiciel. Ce stage a été effectué sur une période de 2 mois.

Durant mon stage j'ai été amené à concevoir et participer au développement d'une application mobile d'entreprise pour le compte de la société SOZO unltd. Ce document constitue, le rapport de stage rédigé pour présenter la démarche de conception ainsi que la réalisation technique utilisée pour mener à bien ce projet.
Bonne lecture !

SŌZŌ UNLTD	Rapport de Stage Développeur logiciel	page 6/57 <i>La société</i>
----------------------	--	--------------------------------

1.1.1. La société



SOZO unltd est une agence de communication visuelle spécialisée dans la conception graphique, le design et le développement de solutions interactives.

Parce que ceux sont les gens qui font évoluer le monde, elle attache une attention particulière à chacun de leurs projets pour répondre aux attentes de la révolution numérique.

Par la créativité, elle ajoute de la valeur à leur activité afin de toucher le plus de gens, car ceux sont eux qui nous font évoluer.

Le monde est en constante évolution, et vous participez activement à son changement.
Par votre esprit d'entreprise et de création, vous voulez apporter de nouvelles valeurs.
Vous avez besoin d'un site internet PC, tablette et mobile.
Vous recherchez une identité visuelle à votre image, ou créer des supports de communications print ou vidéos.
Besoin de main d'œuvre pour répondre à des commandes.
Grâce à notre créativité et nos compétences, nous pouvons trouver ensemble des solutions pour apprendre au monde à vous connaître, à vous aimer.

1.1.2. L'équipe :

SOZO unltd est gérée par une personne qui selon la taille des projets s'entoure d'intervenants, afin de répondre au mieux au cahier des charges et délais de réalisations.

L'expérience de la première société NEKO entertainment, a permis au gérant d'optimiser la gestion des projets tout en gérant des équipes d'une quinzaine de personnes dans un soucis d'exigence des gens.

1.1.3. Concept:

Pour SOZO unltd les clients ne sont pas des clients. Ceux sont des personnes. Le concept du studio est d'apporter de la créativité dans ses projets afin de les rendre unique et à la portée de tous.

1.1.4. Services proposés :

- Création d'identités visuelles
- Design et rendus pré-calculer 3D
- Développement de jeux vidéo
- Conception et développement de sites internet
- Conception, développement et d'applications Web

1.1.5. Références :



1.1.6. Réalisations :



1.1.7. Environnement techniques :

Afin de répondre efficacement aux besoins graphiques de chacun de ses clients, SOZO unltd dispose de tous les outils de la chaîne de production nécessaires à la réalisation de travaux techniques complexes (photomontages, illustrations, impression 3D, VR). SOZO unltd a mis en place une organisation spécifique pour ses solutions Web..

1.1.7.1. Environnement de développement

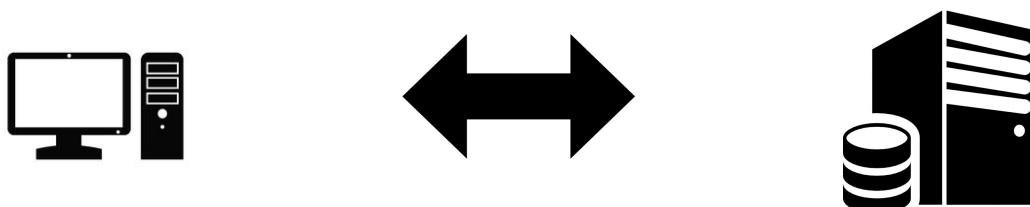


Plate forme locale

Serveur de développement :
XAMPP, PhpMyadmin, Git...

Transfert et synchronisation

FTP / rsync et git (SSH)

Serveur de Production

VPS Linux (OVH) : LAMP,
PhpMyadmin, Git....

1.1.7.2. Plate forme locale s

Système	Windows 7 Pro 32 Bits
Processeur	AMD Athlon ,4 Go de RAM
Logiciels installés	Chromes, Firefox
Connexion Internet	Connexion internet Haut débit (fibre),

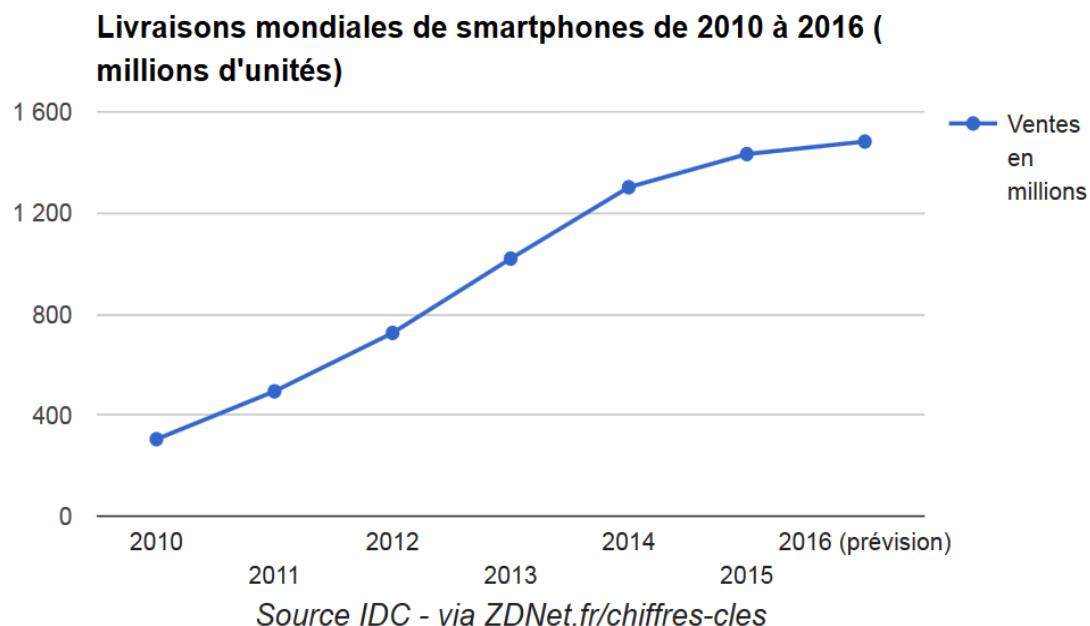
1.1.7.3. Serveur de production

Technologie	Serveurs VPS Linux® vCORE
Stockage	250 Go,
Web	CDN, Accès SSH , Accès FTP , Géolocalisation IP, Statistiques, Logs
Développement	HTML, CSS, Javascript, PHP5.6 / 5.5, Ioncube, PHP/FPM, Tâches planifiée (cron), Perl (cgi), C (cgi), Python (cgi), Imagick(PHP extension) ,Gestionnaire de version (svn + git)
Base de données :	4 BD Mysql (3x 400Mo + 1xGo), Serveur SQL Privé, Sauvegarde / restauration à la demande
Sécurité	Protection Anti-DDoS, SSL Let's Encrypt ,SSL DV / OV / EV,.htaccess ,Mod Security

2. Étude de cas : conception d'une application mobile

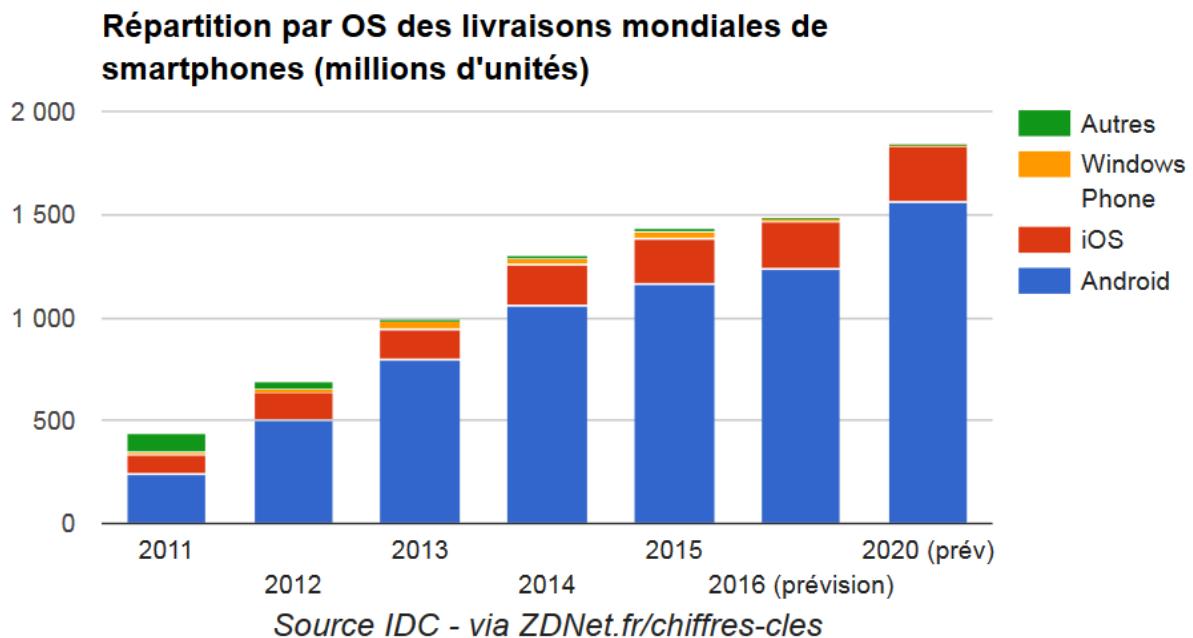
2.1.1. Contexte

L'utilisation quotidienne des Smartphones a modifié notre rapport avec la technologie. À présent nous utilisons nos téléphones portables quotidiennement, pour régler des achats sur internet ou bénéficier de services d'informations. Et même si le marché est très diversifié en terme de modèles d'appareils et de composants nous assistons une généralisation des systèmes d'exploitation.



Ces tendances ont permis l'émergence de nouveaux acteurs et nouveaux de services plus adaptés aux besoins des utilisateurs de Smartphones. L'utilisation croissante de la géolocalisation et le souci de consommer de façon plus responsable ont laissé entrevoir de nouvelles perspectives en terme d'applications.

C'est dans ce contexte que SOZO unltd a décidé de réaliser une l'application pour les commerces de proximité. : SnapAds



2.1.2. L'application



SnapAds est une application mobile.

- une application qui permet de localiser de bons plans (services ou produits)
- une application s'adresse aux particuliers et aux commerçants
- une application qui facilite l'échange de bons plans entre amis
- une application multiplateforme (Android et IOS) pour smartphones et tablettes

2.1.3. Contraintes et environnement

Par souci esthétique l'application mobile devra être épurée et simple à utiliser. Pour que l'expérience utilisateur soit complète, les utilisateurs devront activer la géolocalisation et accepter de recevoir les notifications liées aux annonces .

Pour des raisons de coût, l'application devra être multiplateforme (ios & Android), consultable sur smartphone et tablette et devra être disponible en téléchargement depuis les stores respectifs. Important :

L'application sera gérée depuis un back-office WEB.

SŌZŌ UNLTD	Rapport de Stage Développeur logiciel	page 11/57 <i>Objectif :</i>
----------------------	--	---------------------------------

2.1.4. Objectif :

L'objectif du projet qui m'a été confié est de développer une application mobile multiplate-formes qui permettrait aux commerces de diffuser des annonces et aux utilisateurs d'en être informés de bons plans dans un périmètre défini.

Cette application sera gérée depuis un back-office consultable depuis internet.

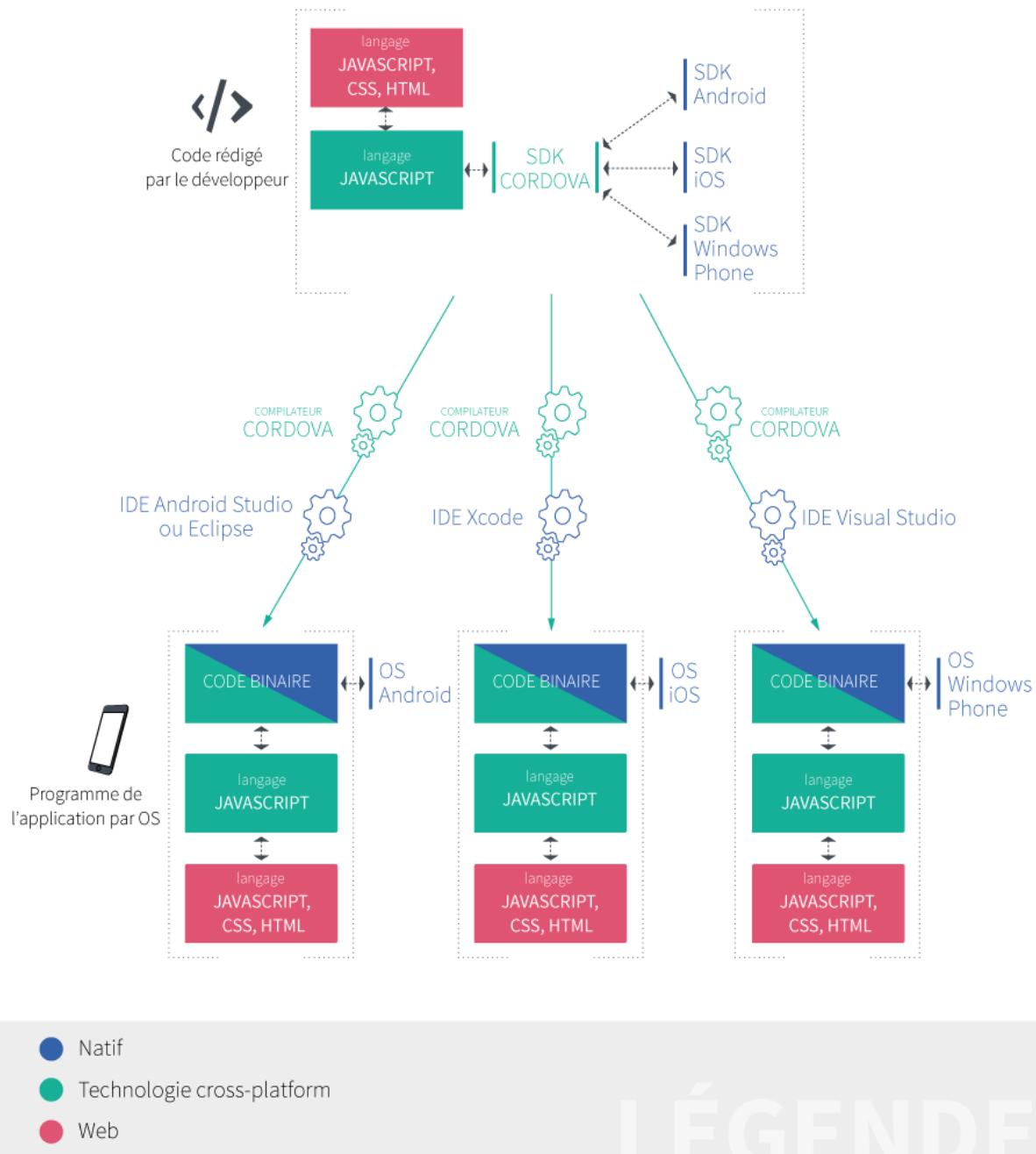
2.1.5. La solution

Afin de répondre au mieux à la demande formulée nous avons choisi de Développer de l'application en se basant sur le Framework CORDOVA.



Cordova est une solution de développement cross-platform qui permet développer une application mobile une fois et de la faire fonctionner sur toutes les plates-formes mobiles du marché.

2.1.5.1. Description



2.1.5.2. Avantages

- OpenSource
- basé sur les standards du Web.
- n'inclut pas d'interpréteur de code
- supporté la plupart des plate-formes mobiles du marché
- possède de nombreux *plugins* pour d'accéder aux ressources natives des terminaux mobiles.

2.1.5.3. Limites

L'inconvénient de Cordova est adaptée à des projets mobiles avec une faible logique métier .car doit être développée en Javascript.

2.1.6. Développement

Quoi	Avec Quoi	Outils
Maquettes	Mockups	Adobe Fireworks
Base de Données	MySQL	PhpMyAdmin, Msqlworkbench
Backoffice	PHP, HTML, CSS	Netbeans
Application mobile	HTML, CSS,Javascript, (Jquery, Ajax)	Netbeans
Compilation	Node js, Cordova,	CLI, Android Studio,

2.1.7. Langages et outils utilisés



3. Conception

3.1. Démarche

Pour la conception de notre application nous utiliserons le langage de modélisation UML (*Unified Modeling Language*) en se basant sur l'approche de M. Pascal Roques (*dans son ouvrage : UML2 :Modéliser une application web, Eyrolles EAN13 : 9782212123890*)

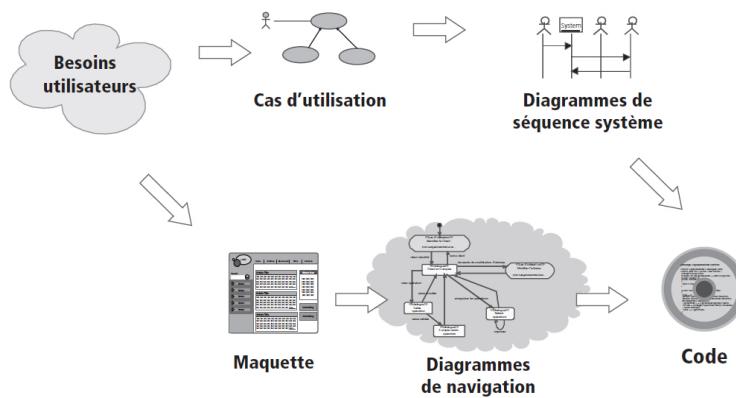


Schéma du processus de modélisation de l'application

SŌZŌ UNLTD	Rapport de Stage Développeur logiciel	page 15/57 <i>Analyse des besoins</i>
----------------------	--	--

3.2. Analyse des besoins

3.2.1. Exigences fonctionnelles

Application mobile

- Authentification
- la création de compte
- la création d'annonce
- l'affichage des annonces par catégorie
- consultation détaillée des annonces
- gestion des paramètres personnels
- modération des annonces
- envoi de mail

Back-office en ligne

- Authentification
- gestion des utilisateurs
- gérer des annonces
- gestions des modérateurs
- modération des annonces

Exigences non fonctionnelles

- Interface simple et efficace
- Accès rapide à l'information
- Organisation claire et structurée de l'information
- Respect des contraintes d'accessibilité et d'ergonomie

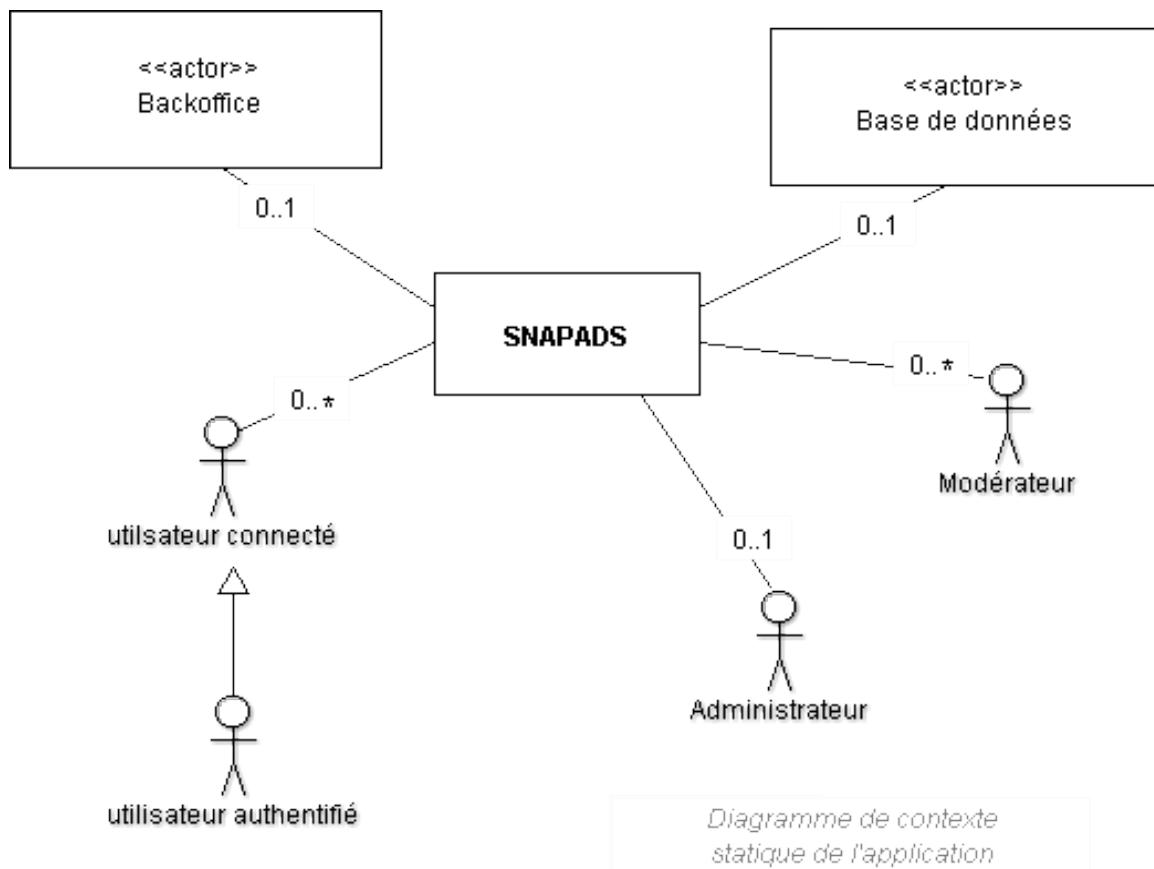
Contraintes de conception

- Seul les profils type commerçant pourront créer des annonces
- Les annonces seront stockées dans une base de données (SGBDR)
- L'utilisateur devra activer les notifications sur son téléphone
- L'utilisateur devra activer mes services liés à la géolocalisation
- La gestion des utilisateurs et des annonces se fera depuis le back-office Web.

3.3. Modélisation

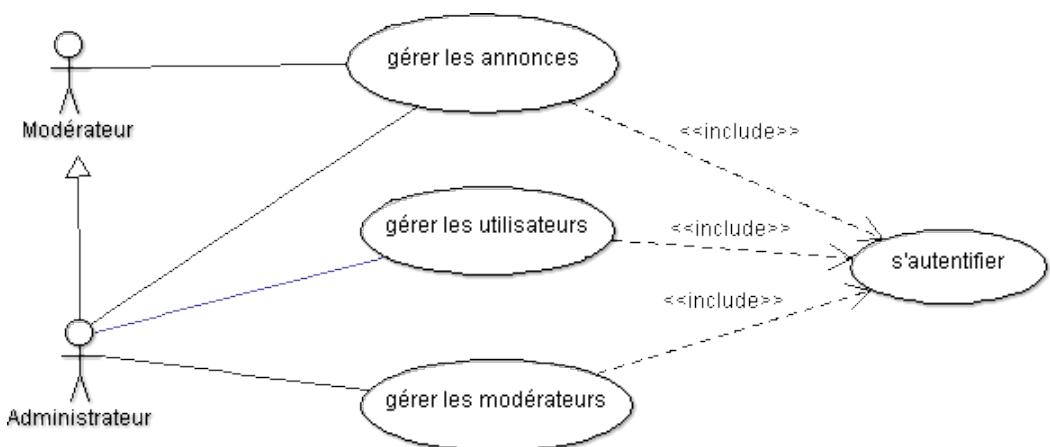
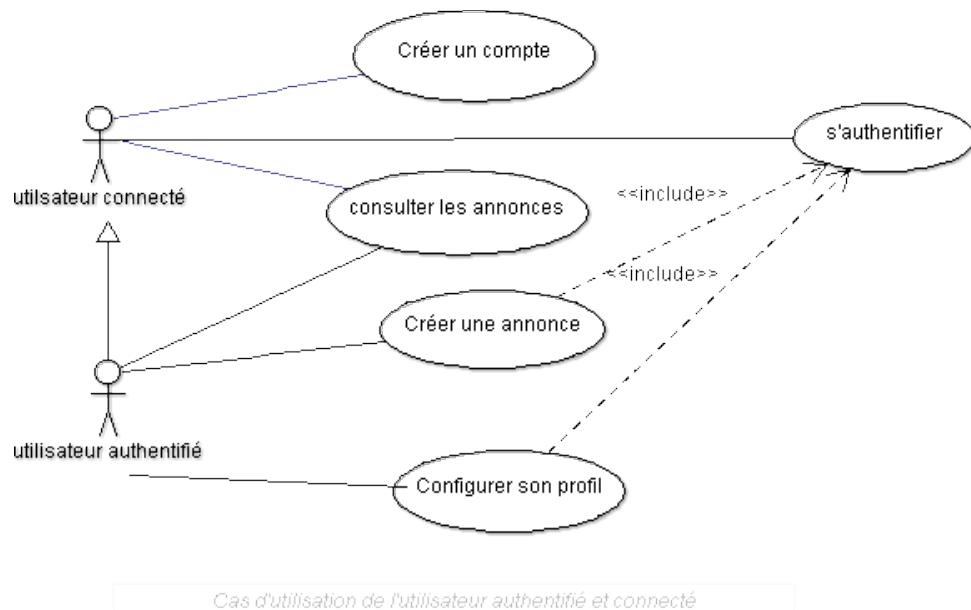
3.3.1. Diagramme de contexte statique

Le diagramme de contexte statique sert à présenter le contexte métier ou fonctionnel de l'application. Il nous permet de clarifier notre étude en début de projet ..



3.3.2. Diagramme de cas d'utilisation

Les cas d'utilisation constituent une vision orientée utilisateur . Ils permettent de capturer le comportement d'un système perçu par un utilisateur ou à exprimer les besoins de cet utilisateur vis à vis de ce système.

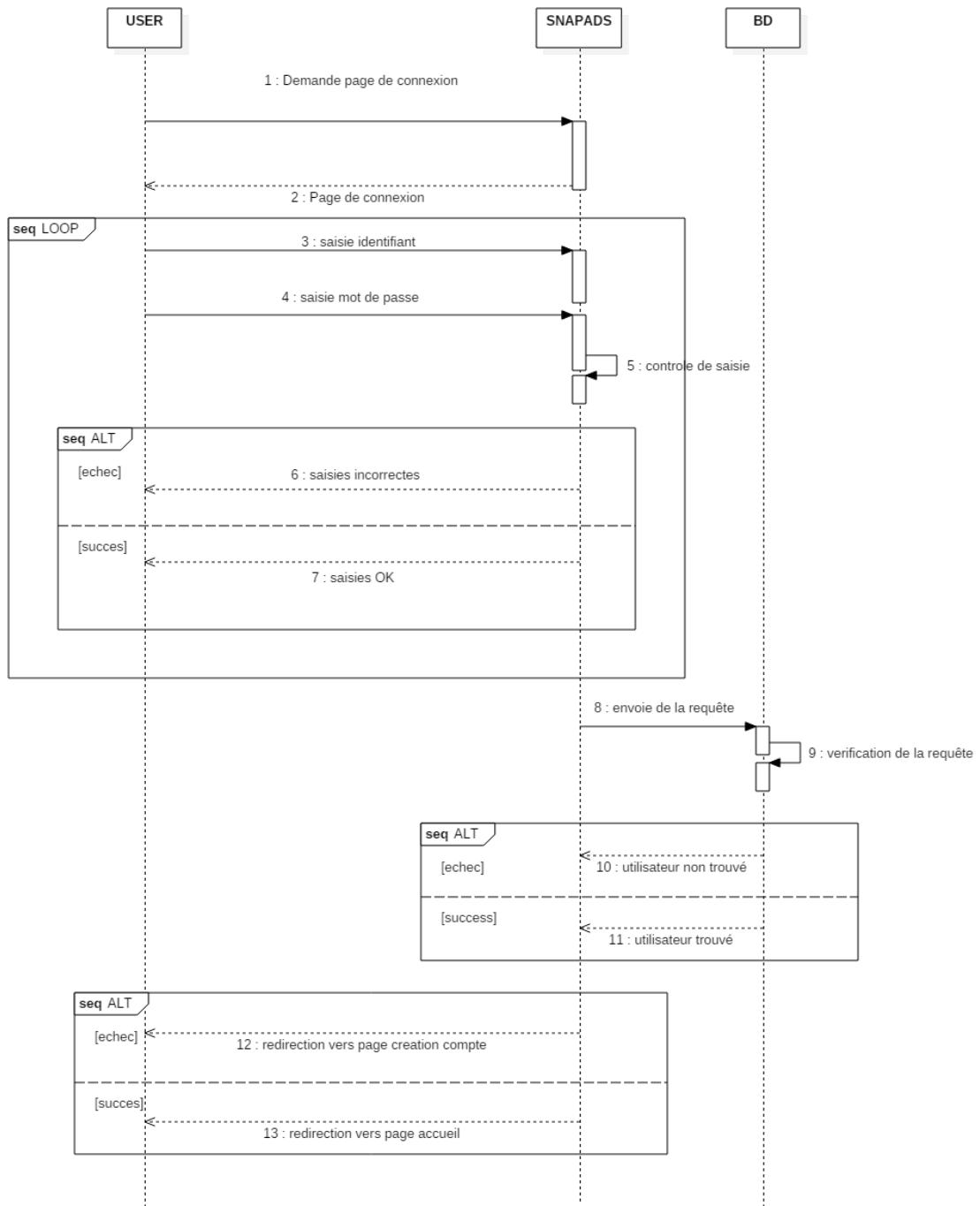


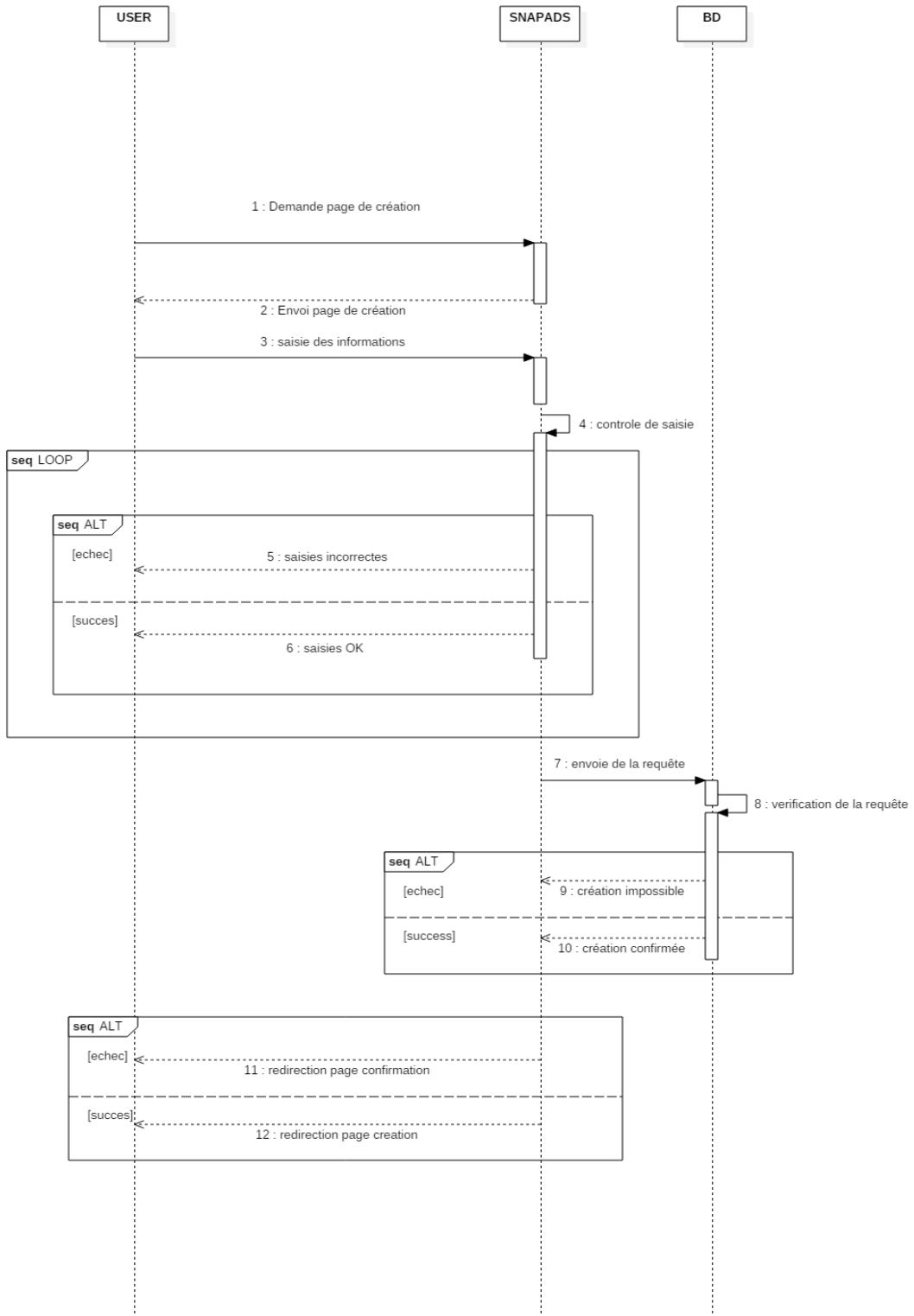
	Rapport de Stage Développeur logiciel	page 18/57 <i>Fiche descriptive des cas d'utilisation (exemple)</i>
---	--	--

3.3.3. Fiche descriptive des cas d'utilisation (exemple)

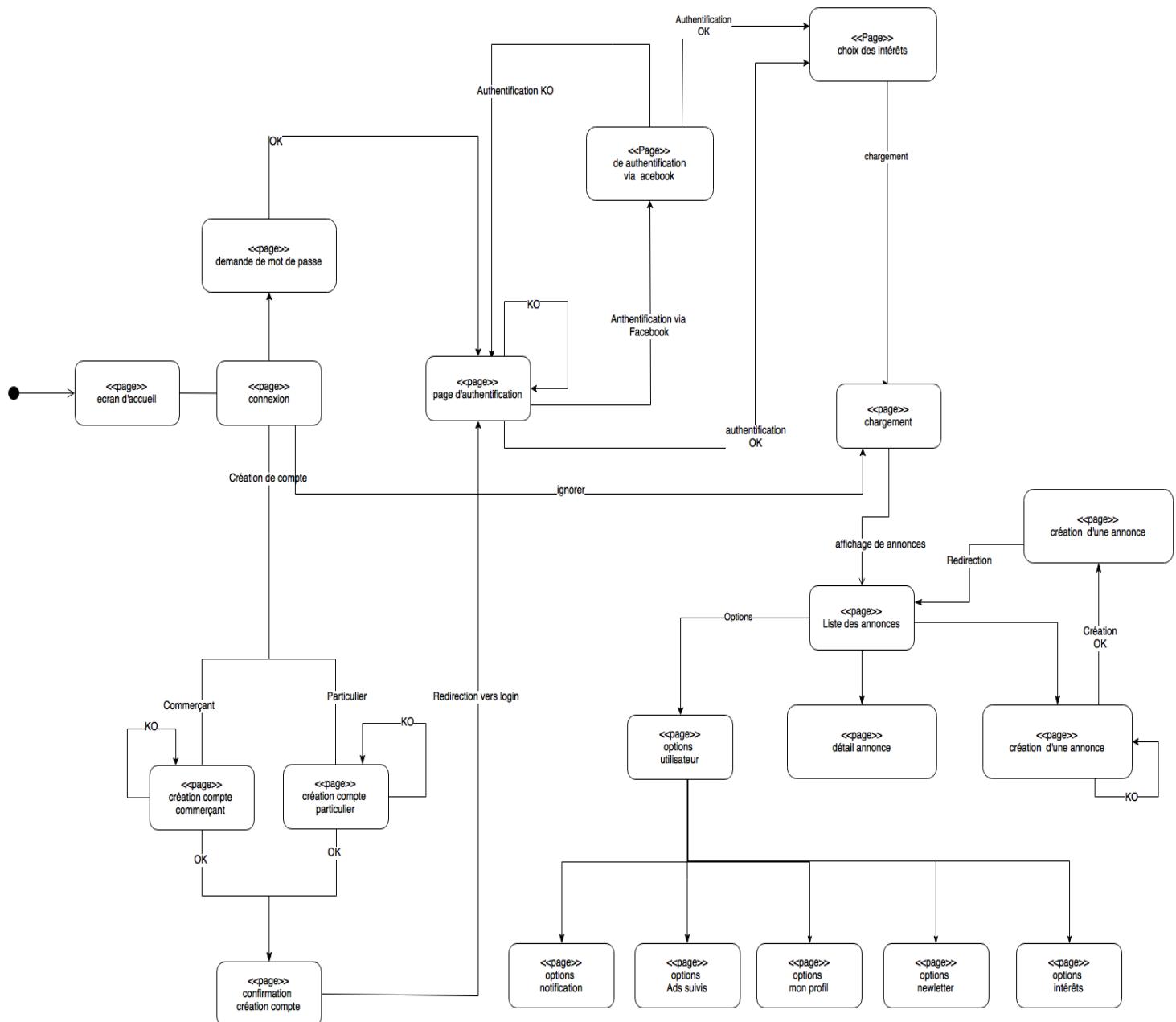
Étapes	Description
Créer une annonce	<p>Titre : Crée une annonce</p> <p>Résumé : ce cas d'utilisation permet à un utilisateur authentifié de créer une annonce</p> <p>Acteurs : utilisateur connecté (principal), Base de données (secondaire).</p> <p>Date de création : 09/07/16</p> <p>Date de modification : 05/05/06</p> <p>Version : 1.0 Responsable : Alban MABIALA</p>
Pré-conditions	<ul style="list-style-type: none"> • connexion à la bd disponible, • utilisateur authentifié • géolocalisation activée
Scénario nominal	<ul style="list-style-type: none"> • accès à la page de création d'annonce • saisie des champs • contrôle de surface • envoie requête serveur • Contrôle des informations cote serveur • Inscription dans la BD • affichage page de confirmation • Redirection vers la liste des annonces.
Scénarios alternatifs	<ul style="list-style-type: none"> • les zones saisies renseignées (vides ou non-conformes) • l'utilisateur décide de quitter la création de l'annonce • l'utilisateur décide de quitter l'application
Scénarios d'erreur	<ul style="list-style-type: none"> • La création d'annonce échoue lors de la soumission
Post-conditions	<ul style="list-style-type: none"> • une page de confirmation création • une annonce est créée
Exigences non fonctionnelles	<ul style="list-style-type: none"> • Saisie des informations simplifiée
Besoins d'IHM	<ul style="list-style-type: none"> • page de création d'annonce, • page détails d'une annonce, • page de confirmation de création

3.3.4. Diagrammes de séquence





3.3.5. Diagramme de navigation



SŌZŌ UNLTD	Rapport de Stage	page 22/57
	Développeur logiciel	<i>Maquettes</i>

3.3.6. Maquettes

Les maquettes sont de représentations des futures vues de l'interfaces de l'application. Elles ne sont pas définitives mais prennent en compte tous les éléments de navigation afin d'avoir un aperçu de la logique d'enchaînement d'écrans. Les fonctionnalités qu'elles restent fictives.

Page de connexion

Page d'authentification

Page de création de compte (Particulier)

Page de choix des intérêts

3.3.7. Classement des cas d'utilisation

Après avoir identifié les cas d'utilisation nous avons les avons classer en tenant compte des aspect marketing et des risques techniques. Par la suite nous avons pu les découper en itérations.

Cas d'utilisation	Priorité	Risque	itération
créer une annonce	Haute	Haut	2
consulter une annonce	Haute	Moyenne	3
gérer ses paramètres	Moyenne	bas	8
s'identifier	Haute	Haut	1
signaler une annonce	Basse	bas	9
suivre une annonce	Basse	bas	10
valider une annonce	Moyenne	Haut	5
s'inscrire à la newsletter	Basse	bas	13
choisir ses catégories	Moyenne	bas	11
activer les notifications	Basse	bas	12
gérer les utilisateurs	Moyenne	moyen	6
gérer les modérateurs	Moyenne	moyen	7
gérer les annonces	haute	moyen	4

3.3.8. Planning

		Nom	Durée	Début	Fin	Prédécesseurs	Ressources
1		✉ Cahier des charges	4j	20/06/2016	23/06/2016		
2		Expression des besoins	3j	20/06/2016	22/06/2016		
3		Redaction	2j	22/06/2016	23/06/2016		
4		✉ Analyse	5j	24/06/2016	30/06/2016	1	
5		Maquettage	3j	24/06/2016	28/06/2016		
6		Diagrammes UML	5j	24/06/2016	30/06/2016		
7		✉ Graphisme	3j	01/07/2016	05/07/2016	4	
8		Charte graphique	3j	01/07/2016	05/07/2016		
9		✉ Conception de la base de données	10j	05/07/2016	18/07/2016	4	
10		Modélisation	10j	05/07/2016	18/07/2016		
11		✉ Developpement et codage	24j	19/07/2016	19/08/2016	9	
12		Test	5j	25/07/2016	29/07/2016	9	
13		Codage de l'interface mobile	5j	19/07/2016	25/07/2016		
14		✉ Codage des fonctionnalités	20j	25/07/2016	19/08/2016		
15		Connexion BD	1.5j	25/07/2016	26/07/2016	9	
16		Création annonce / user	2j	26/07/2016	28/07/2016	15	
17		Modification Annonce / user	2j	28/07/2016	01/08/2016	16	
18		Affichage des annonces	3j	01/08/2016	03/08/2016	16	
19		Authentification web / mobile	2j	04/08/2016	05/08/2016	15	
20		Geolocalisation	2j	12/08/2016	15/08/2016	16	
21		Fonctionnalités photo	3j	05/08/2016	09/08/2016	16	
22		Gestion des paramètres	3j	17/08/2016	19/08/2016	16	
23		✉ Sécurité	3j	22/08/2016	24/08/2016	11	
25		✉ Test	2j?	30/08/2016	31/08/2016	23	
27		✉ Compilation	1.5j	01/09/2016	02/09/2016	25	
29		✉ deploiemt	1j?	01/09/2016	01/09/2016	25	
31		Redaction rapport de stage	37j?	20/06/2016	09/08/2016		

SŌZŌ UNLTD	Rapport de Stage Développeur logiciel	page 26/57 Charte graphique
----------------------	--	--------------------------------

3.3.9. Charte graphique

SNAP ADS Charte Graphique

01. PALETTE DE COULEUR

#2EABE3
#96D6F1
#095DB1
#8B9EB6
#BDC7D4

02. TYPOGRAPHIE

Roboto

Title	Bold 72px
Subheader	Regular 36px
Body 2 / Menu	Medium 16px
Body 1	Regular 15px
Caption	Regular 12px
Button	Medium 13-16px

03. BUTTONS

FILLED BUTTONS

Normal	Normal
Hover	Hover

Onglets

04. FORMULAIRES

05. LISTE DES ANNONCES

06. ICÔNES / SOCIAUX

	Rapport de Stage Développeur logiciel	page 27/57 <i>Analyse et conception des données</i>
---	---	--

4. Analyse et conception des données

4.1. Les règles de gestion :

Les règles de gestion dépendent du contexte "métier". Elles décrivent les actions effectuées par les différents acteurs au sein d'un processus métier.

- 1 une annonce est constituée d'une catégorie, d'un ancien prix, d'un nouveau prix, d'une date de début, d'une date de fin, d'un statut, d'une adresse , d'une photo, d'une description, d'une position GPS
- 2 une annonce peut être signalée par un ou plusieurs utilisateurs
- 3 les annonces sont consultables par tous types d'utilisateurs (authentifiés ou non)
- 4 les utilisateurs peuvent être anonymes ou authentifiées
- 5 les utilisateurs anonymes peuvent consulter les annonces
- 6 les utilisateurs authentifiés peuvent créer des annonces
- 7 les utilisateurs authentifiés peuvent suivre une ou plusieurs annonces
- 8 les utilisateurs authentifiés ont accès à des paramètres
- 9 les utilisateurs authentifiés peuvent être des commerçants ou des particuliers
- 10 les commerçants sont définis par une raison sociale, un login, un mot de passe, un numéro de siret, une adresse mail, un numéro de téléphone, une adresse postale, une position GPS
- 11 les particuliers sont définis par un pseudo, un nom, un prénom, un login, un mot de passe, une adresse mail, un numéro de téléphone, une adresse postale, une position GPS
- 12 les paramètres disponibles sont : la liste des intérêts, le rayon de détection des annonces, la gestion des notifications, l'inscription à la newsletter

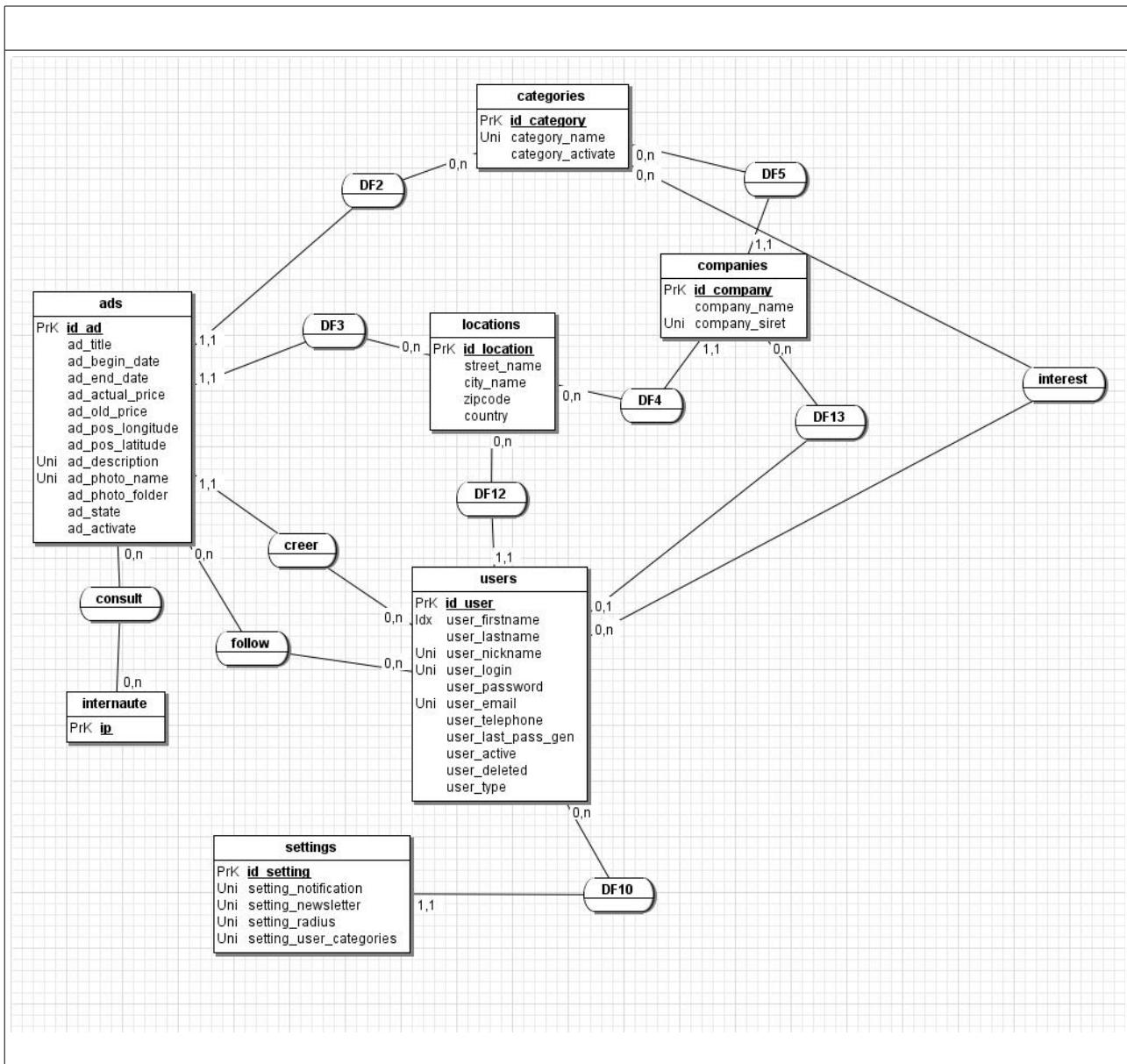
	Rapport de Stage Développeur logiciel	page 28/57 <i>Dictionnaire des données</i>
---	---	---

4.2. Dictionnaire des données

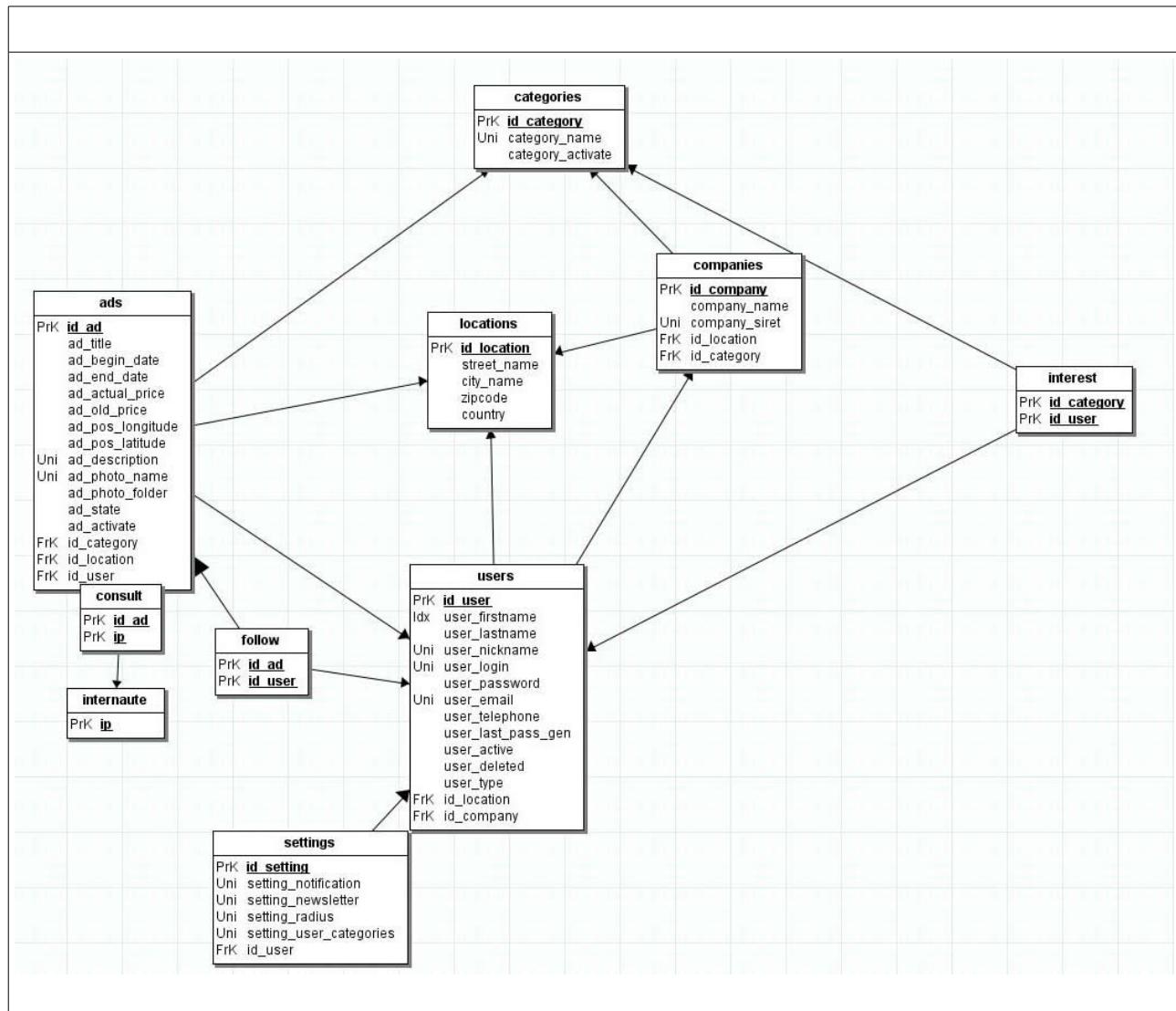
Après l'étude des règles de gestion, nous pouvons établir le dictionnaire des données suivant

Code mnémonique	Désignation	Type	Taille	Remarque
<u>id_photo</u>	Identifiant de la photo	N	10	
photo_name	nom de la photo	A	45	
photo_folder	repertoire de la photo	A	45	
<u>id_location</u>	Identifiant de l'adresse	N	10	
street_name	Nom de la rue	AN	45	
city_name	Nom de la ville	A	64	
zipcode	Code postal de la ville	N	45	
country	Pays	A	12	
<u>id_user</u>	Identifiant de l'utilisateur	N	10	
user_firstname	nom de l'utilisateur	A	32	
user_lastname	prénom de l'utilisateur	A	32	
user_login	login de l'utilisateur	AN	32	
user_password	mot de passe de l'utilisateur	AN	32	
user_email	adresse mail de l'utilisateur	AN	128	
user_telephone	n° de téléphone de l'utilisateur	N	16	
userNickname	pseudonyme de l'utilisateur	AN	32	
user_type	type de l'utilisateur	N	1	
company_name	nom de l'entreprise	AN	128	
company_siret	n° de siret de l'entreprise	N	?	
<u>id_company</u>	identifiant de l'entreprise	N	10	
<u>id_settings</u>	Identifiant du paramètre	N	10	
setting_notification	Paramètre de notification	N	1	
setting_newsletter	Paramètre de newsletter	N	1	
setting_radius	Parametre de zone de localisation	N	1	
setting_user_categories	paramètre d'intérêt de l'utilisateur	N	1	
<u>id_categories</u>	Identifiant de la catégorie	N	10	
category_name	nom de la catégorie	A	45	
<u>id_ad</u>	identifiant de la l'annonce	N	1	
ad_title	Titre de l'annonce	AN	80	
ad_begin_date	Date de début de l'annonce	date		Au format AAAA-JJ-MM
ad_end_date	Date de fin de l'annonce	date		Au format AAAA-JJ-MM
ad_actual_price	Prix actuel de l'annonce	D	-20,2	
ad_old_price	Ancien prix de l'annonce	D	-20,2	
ad_adresse	Adresse de l'annonce	AN	80	
ad_pos_longitude	Longitude de l'annonce	D	-11,8	calculées
ad_pos_latitude	Latitude de l'annonce	D	-11,8	calculées
ad_description	Description de l'annonce	AN	256	
ad_category	Catégorie de l'annonce	A	45	

4.3. Model conceptuel de données (MCD)



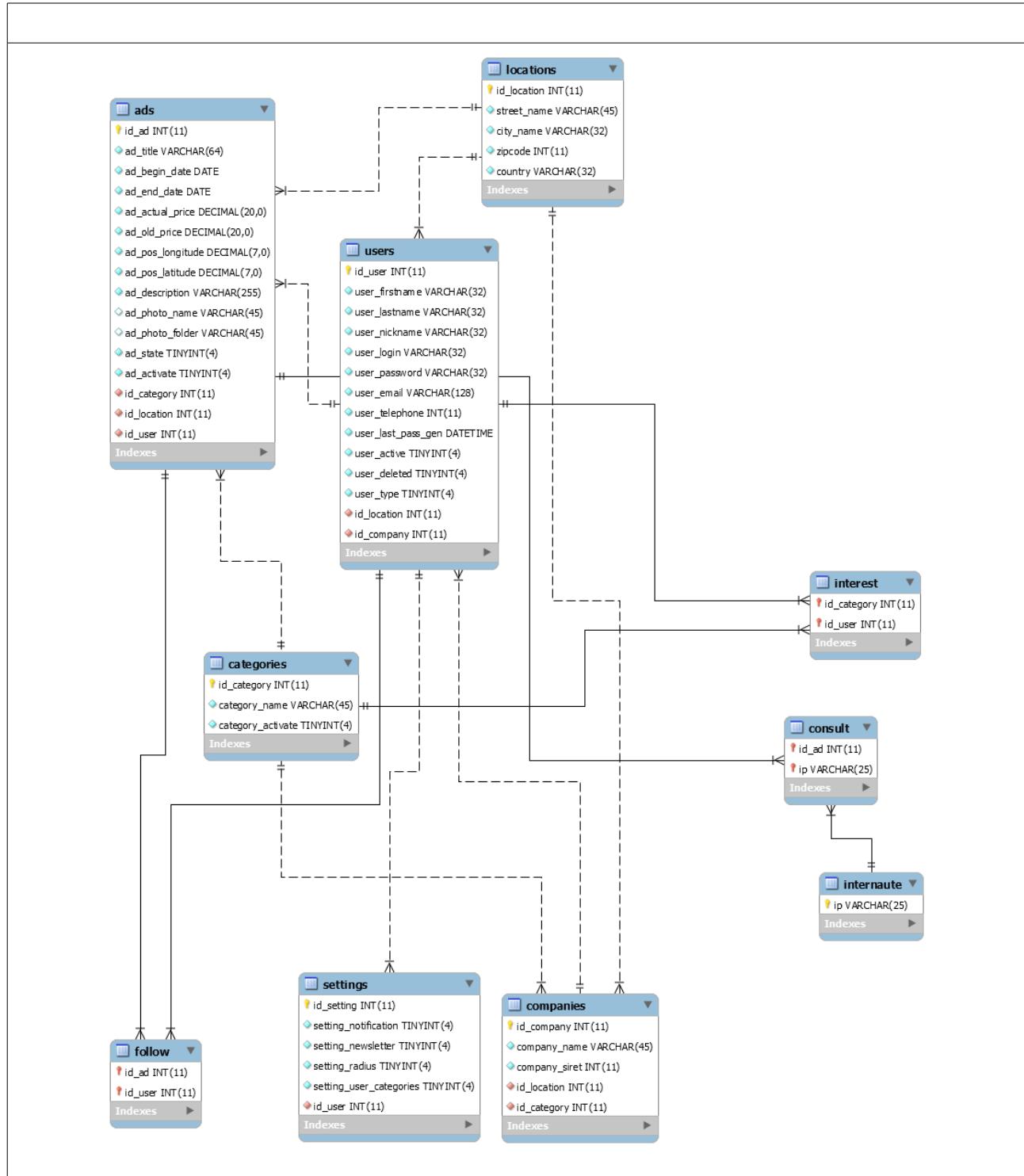
4.4. Modèle logique des données (MLD)



4.5. Script de création de la BD

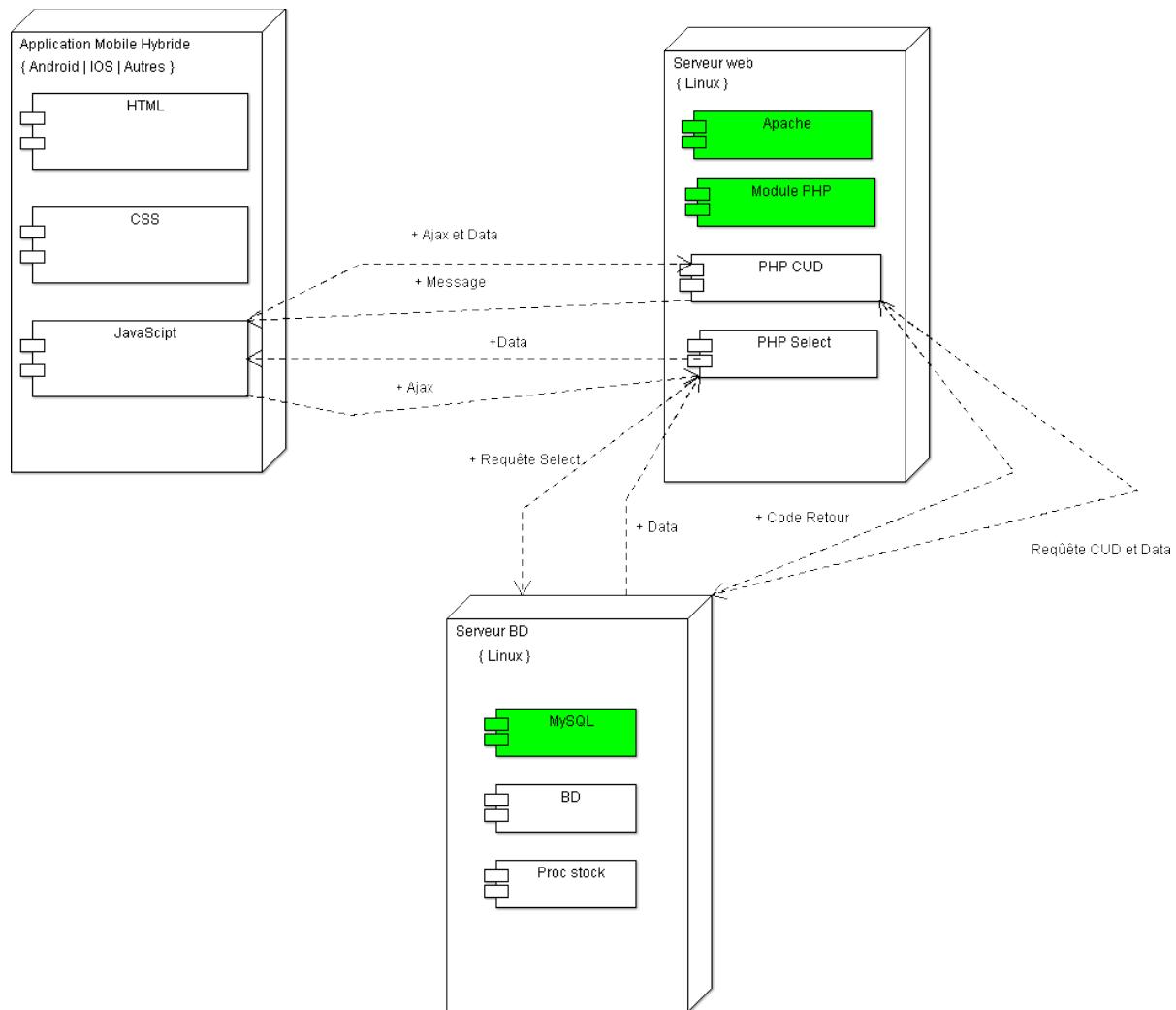
(voir annexe)

4.6. Modèle physique de données (MPD)



4.7. Diagramme de déploiement

Le diagramme de déploiement de notre application mobile hybride basée sur de l'HTML/CSS/Javascript côté client et PHP et MySQL coté serveur



5. Code

5.1. back-office

snap_ads.properties

```
-- snap_ads.properties
[section_connexion]
protocole=mysql
serveur=localhost
port=3306
bd=karrington
user=root
mdp=
```

saisieAnnonce.php

```
<html>
    <head>
        <meta charset="utf-8">
        <meta http-equiv="X-UA-Compatible" content="IE=edge">
        <meta name="viewport" content="width=device-width, initial-scale=1">
        <title>Snapads : créer une annonce</title>
        <script src="js/jquery.js"></script>
        <link rel="stylesheet"
            href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.1/css/bootstrap.min.css">
            <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.1/css/bootstrap-theme.min.css">

    </head>
    <body>

        //header
        <?php require_once 'header.php'; ?>';
        <p id="Resultat"></p>
        <div id="message"></div>
        <div class="container mainContainer col-md-6 col-xs-12s col-md-offset-3" >
            <div class="row">
                <div class="col-md-12">
                    <h1>Saisie de votre annonce !</h1>
                </div>
            </div>
            <div class="row">
                <div class="col-md-12">
                    <p class="textend" >Merci de bien vouloir saisir les informations relatives
à votre annonce</p>
                </div>
            </div>
            <form action="saveAnnonce.php" id="contact" method="POST" name="#">
                <div class="row form-group">
                    <div class="col-md-2">
                        <label for="adCategory" class="control-label">Catégorie du
produit</label>
                    </div>
                    <div class="col-md-10">
                        <select id="adCategory" name="adCategory" class="form-control" size="1">
```

```
        <option value="2">loisir</option>
        <option value="1">automobile</option>
    </select>
</div>
</div>
<div class="row form-group">
    <div class="col-md-2">
        <label for="location" class="control-label">ID adresse</label>
    </div>
    <div class="col-md-10">
        <div class="error-message">erreur</div>
        <input type="text" name="location" class="form-control" id="location"
value="1">
    </div>
</div>

<div class="row form-group">
    <div class="col-md-2">
        <label for="user" class="control-label">ID utilisateur</label>
    </div>
    <div class="col-md-10">
        <div class="error-message">erreur</div>
        <input type="text" name="user" class="form-control" id="user" value="1">
    </div>
</div>
<div class="row form-group">
    <div class="col-md-2">
        <label for="adTitle" class="control-label">Titre de l'annonce</label>
    </div>
    <div class="col-md-10">
        <div class="error-message">erreur</div>
        <input type="text" name="adTitle" class="form-control" id="adTitle"
value="Annonce desss dernière minute">
    </div>
</div>
<div class="row form-group">
    <div class="col-md-2">
        <label for="adBeginDate" class="control-label">Date de debut</label>
    </div>
    <div class="col-md-10">
        <div class="error-message">erreur</div>
        <input type="date" name="adBeginDate" class="form-control" id="adBeginDate"
value="2017-07-14">
    </div>
</div>
<div class="row form-group">
    <div class="col-md-2">
        <label for="adEndDate" class="control-label ">Date de fin</label>
    </div>
    <div class="col-md-10">
        <div class="error-message">erreur</div>
        <input type="date" max="2017-07-15" min="2017-07-15" name="adEndDate"
class="form-control" id="AdEndDate" value="2017-07-15">
    </div>
</div>
<div class="row form-group">
    <div class="col-md-2">
        <label for="adLong" class="control-label">longitude</label>
    </div>
    <div class="col-md-10">
        <div class="error-message">erreur</div>
```

```
<input type="text" name="adLong" class="form-control" id="adLong"
value="25555">
    </div>
</div>
<div class="row form-group">
    <div class="col-md-2">
        <label for="adLat" class="control-label">latitude</label>
    </div>
    <div class="col-md-10">
        <div class="error-message">erreur</div>
        <input type="text" name="adLat" class="form-control" id="adLat" value="000">
    </div>
</div>
<div class="row form-group">
    <div class="col-md-2">
        <label for="newPrice" class="control-label">Nouveau prix (€)</label>
    </div>
    <div class="col-md-10">
        <div class="error-message">erreur</div>
        <input type="text" name="newPrice" class="form-control" id="newPrice" value="12">
    </div>
</div>

<div class="row form-group">
    <div class="col-md-2">
        <label for="oldPrice" class="control-label">Ancien prix (€)</label>
    </div>
    <div class="col-md-10">
        <div class="error-message">erreur</div>
        <input type="text" name="oldPrice" class="form-control" id="oldPrice" value="15">
    </div>
</div>
<div class="row form-group">
    <div class="col-md-2">
        <label for="desc" class="control-label">Description de l'offre</label>
    </div>
    <div class="col-md-10">
        <div class="error-message">erreur</div>
        <textarea id="desc" class="form-control" name="desc" >lorem ipsum sit dolor
amet</textarea>
    </div>
</div>
<div class="row form-group">
    <div class="col-md-2">
        <label for="adPhotoName" class="control-label">nom de la photo</label>
    </div>
    <div class="col-md-10">
        <div class="error-message">erreur</div>
        <input type="text" name="adPhotoName" class="form-control" id="adPhotoName" value="exemple">
    </div>
</div>
<div class="row form-group">
    <div class="col-md-2">
        <label for="adPhotoFolder" class="control-label">repertoire de la
photo</label>
    </div>
    <div class="col-md-10">
        <div class="error-message">erreur</div>
        <input type="text" name="adPhotoFolder" class="form-control"
```

```
        id="adPhotoFolder" value="/photos">
            </div>
        </div>
        <div class="row form-group">
            <div class="col-md-2">
                <label for="adState" class="control-label">Etat</label>
            </div>
            <div class="col-md-10">
                <div class="error-message">erreur</div>
                <select id="adState" name="adState" class="form-control" size="1">
                    <option value="1">validé</option>
                    <option value="2">En cours de validation</option>
                    <option value="3">non validé</option>
                </select>
            </div>
        </div>
        <div class="row form-group">
            <div class="col-md-2">
                <label for="photofile" class="control-label">Etat</label>
            </div>
            <div class="col-md-10">
                <div class="error-message">erreur</div>
                <input type="file" name ="photofile" >
            </div>
        </div>
        <div class="row form-group">
            <div class="col-md-2">
                <label for="adActivation" class="control-label">activation</label>
            </div>
            <div class="col-md-10">
                <div class="error-message">erreur</div>
                <select id="adActivation" name="adActivation" class="form-control" size="1">
                    <option value="1">activé</option>
                    <option value="0">désactivé</option>
                </select>
            </div>
        </div>
        <hr>
        <div class="form-group form-group">
            <div class="col-md-offset-2 col-md-10">
                <input type="submit" class="btn btn-success" id="envoyer" value="save">
                <input type="button" class="btn btn-danger" value="Annuler">
            </div>
        </div>
    </body>
</html>
```

saveAnnonce.php

```
<?php
#-----
# Fichier de sauvegarde d'une annonce dans la BD
#-----

//Fichier de paramètres
$Proprietes = parse_ini_file("snap_ads.properties");
```

```
//Paramètres de connexion
$dbhost = $Proprietes ['serveur'];
$dbport = $Proprietes['port'];
$dbuser = $Proprietes['user'];
$dbuserPass = $Proprietes['mdp'];
$dbname = $Proprietes['bd'];
$dbpass = '';

//Je récupère les saisies utilisateurs
$adtitle = filter_input(INPUT_POST, 'adTitle');
$adbegindate = filter_input(INPUT_POST, 'adBeginDate');
$adenddate = filter_input(INPUT_POST, 'adEndDate');
$newprice = filter_input(INPUT_POST, 'newPrice');
$oldprice = filter_input(INPUT_POST, 'oldPrice');
$adlong = filter_input(INPUT_POST, 'adLong');
$adlat = filter_input(INPUT_POST, 'adLat');
$addesc = filter_input(INPUT_POST, 'desc');
$adphotoname = filter_input(INPUT_POST, 'adPhotoName');
$adphotofolder = filter_input(INPUT_POST, 'adPhotoFolder');
$adstate = filter_input(INPUT_POST, 'adState');
$adactivation = filter_input(INPUT_POST, 'adActivation');
$category = filter_input(INPUT_POST, "adCategory");
$location = filter_input(INPUT_POST, 'location');
$user = filter_input(INPUT_POST, 'user');

//requête
$sql= "INSERT INTO ads (ad_title, ad_begin_date, ad_end_date, ad_actual_price, ad_old_price,
ad_pos_longitude, ad_pos_latitude, ad_description, ad_photo_name, ad_photo_folder, ad_state,
ad_activate, id_category, id_location, id_user)
VALUES (?,?,?,?,?,?,?,?,?,?,?,?,?,?)";

//Connexion
try {

$pdo = new PDO("mysql:host=$dbhost;port=$dbport;dbname=$dbname", $dbuser, $dbuserPass);
$ps=$pdo->prepare($sql);

//tableau de paramètres
$params = array($adtitle, $adbegindate, $adenddate, $newprice, $oldprice, $adlong, $adlat, $addesc,
$adphotoname, $adphotofolder, $adstate,$adactivation, $category, $location,$user);

//j'exécute ma requête en lui passant un tableau de paramètres
$ps->execute($params);

//redirection
header("location:annonces.php");

//je lève une exception
} catch (PDOException $e) {
    $messsage = $e->getMessage();
}

?>
```

editAnnonce.php

```
<?php

#-----
# Page d'édition d'une annonce
#-----

$Proprietes = parse_ini_file("snap_ads.properties");

$dbhost = $Proprietes ['serveur'];
$dbport = $Proprietes['port'];
$dbuser = $Proprietes['user'];
$dbuserPass = $Proprietes['mdp'];
$dbname = $Proprietes['bd'];
$dbpass = '';
$code = $_GET['code'];

//je choisis toutes les annonces
$sql = "SELECT * FROM ads WHERE id_ad=?";

//nouvel objet de connexion
$pdo = new PDO("mysql:host=$dbhost;port=$dbport;dbname=$dbname", $dbuser, $dbuserPass);

//tableau de paramètres
$params = array($code);

//Préparation de la requête
$ps = $pdo->prepare($sql);

//exécution de la requête
$ps->execute($params);

//Je récupère une ligne du jeu d'enregistrement
$annonce = $ps->fetch();
?>

<html>
    <head>
        <meta charset="utf-8">
        <meta http-equiv="X-UA-Compatible" content="IE=edge">
        <meta name="viewport" content="width=device-width, initial-scale=1">
        <title>Snapads : créer une annonce</title>
        <script src="js/jquery.js"></script>
        <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.1/css/bootstrap.min.css">
        <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/
theme.min.css">
        <script type="text/javascript" src="js/AdsInsert.js"></script>
    </head>
    <body>
        <?php require_once 'header.php'; ?>';
        <p id="Resultat"></p>
        <div id="message"></div>
        <div class="container mainContainer col-md-6 col-xs-12 col-md-offset-3" >
            <div class="row">
                <div class="col-md-12">
                    <h1>Modifier votre annonce</h1>
                </div>
            </div>
        </div>
```

```
<div class="row">
    <div class="col-md-12">
        <p class="textend" >Merci de bien vouloir saisir les informations relatives à
votre annonce</p>
    </div>
</div>
<form action="updateAnnonce.php" id="contact" method="POST" name="#">
    <div class="row form-group">
        <div class="col-md-2">
            <label for="code" class="control-label">code de l'annonce</label>
        </div>
        <div class="col-md-10">
            <div class="error-message">erreur</div>
            <input type="h" name="code" class="form-control" id="code" value="php
echo($annonce['id_ad']) ?&gt;"&gt;
        &lt;/div&gt;
    &lt;/div&gt;
    &lt;div class="row form-group"&gt;
        &lt;div class="col-md-2"&gt;
            &lt;label for="adCategory" class="control-label"&gt;Catégorie du produit&lt;/label&gt;
        &lt;/div&gt;
        &lt;div class="col-md-10"&gt;
            &lt;div class="error-message"&gt;erreur&lt;/div&gt;
            &lt;input type="text" name="adCategory" class="form-control" id="adCategory"
value="<?php echo($annonce['id_category']) ?&gt;"&gt;
        &lt;/div&gt;
    &lt;/div&gt;
    &lt;div class="row form-group"&gt;
        &lt;div class="col-md-2"&gt;
            &lt;label for="location" class="control-label"&gt;ID adresse&lt;/label&gt;
        &lt;/div&gt;
        &lt;div class="col-md-10"&gt;
            &lt;div class="error-message"&gt;erreur&lt;/div&gt;
            &lt;input type="text" name="location" class="form-control" id="location"
value="<?php echo($annonce['id_location']) ?&gt;"&gt;
        &lt;/div&gt;
    &lt;/div&gt;
    &lt;div class="row form-group"&gt;
        &lt;div class="col-md-2"&gt;
            &lt;label for="user" class="control-label"&gt;ID utilisateur&lt;/label&gt;
        &lt;/div&gt;
        &lt;div class="col-md-10"&gt;
            &lt;div class="error-message"&gt;erreur&lt;/div&gt;
            &lt;input type="text" name="user" class="form-control" id="user" value="<?php
echo($annonce['id_user']) ?&gt;"&gt;
        &lt;/div&gt;
    &lt;/div&gt;
    &lt;div class="row form-group"&gt;
        &lt;div class="col-md-2"&gt;
            &lt;label for="adTitle" class="control-label"&gt;Titre de l'annonce&lt;/label&gt;
        &lt;/div&gt;
        &lt;div class="col-md-10"&gt;
            &lt;div class="error-message"&gt;erreur&lt;/div&gt;
            &lt;input type="text" name="adTitle" class="form-control" id="adTitle"
value="<?php echo($annonce['ad_title']) ?&gt;"&gt;
        &lt;/div&gt;
    &lt;/div&gt;
    &lt;div class="row form-group"&gt;
        &lt;div class="col-md-2"&gt;
            &lt;label for="adBeginDate" class="control-label"&gt;Date de debut&lt;/label&gt;
</pre
```

```
</div>
<div class="col-md-10">
    <div class="error-message">erreur</div>
    <input type="date" name="adBeginDate" class="form-control" id="adBeginDate"
value="<?php echo($annonce['ad_begin_date']) ?>">
    </div>
</div>
<div class="row form-group">
    <div class="col-md-2">
        <label for="adEndDate" class="control-label">Date de fin</label>
    </div>
    <div class="col-md-10">
        <div class="error-message">erreur</div>
        <input type="date" max="2017-07-15" min="2017-07-15" name="adEndDate"
class="form-control" id="AdEndDate" value="<?php echo($annonce['ad_end_date']) ?>">
    </div>
</div>
<div class="row form-group">
    <div class="col-md-2">
        <label for="adLong" class="control-label">longitude</label>
    </div>
    <div class="col-md-10">
        <div class="error-message">erreur</div>
        <input type="text" name="adLong" class="form-control" id="adLong" value="<?
php echo($annonce['ad_pos_longitude']) ?>">
    </div>
</div>
<div class="row form-group">
    <div class="col-md-2">
        <label for="adLat" class="control-label">latitude</label>
    </div>
    <div class="col-md-10">
        <div class="error-message">erreur</div>
        <input type="text" name="adLat" class="form-control" id="adLat" value="<?php
echo($annonce['ad_pos_latitude']) ?>">
    </div>
</div>

<div class="row form-group">
    <div class="col-md-2">
        <label for="newPrice" class="control-label">Nouveau prix (€)</label>
    </div>
    <div class="col-md-10">
        <div class="error-message">erreur</div>
        <input type="text" name="newPrice" class="form-control" id="<?php
echo($annonce['ad_actual_price']) ?>" value="12">
    </div>
</div>

<div class="row form-group">
    <div class="col-md-2">
        <label for="oldPrice" class="control-label">Ancien prix (€)</label>
    </div>
    <div class="col-md-10">
        <div class="error-message">erreur</div>
        <input type="text" name="oldPrice" class="form-control" id="oldPrice"
value="<?php echo($annonce['ad_old_price']) ?>">
    </div>
</div>
```

```
<div class="row form-group">
    <div class="col-md-2">
        <label for="desc" class="control-label">Description de l'offre</label>
    </div>
    <div class="col-md-10">
        <div class="error-message">erreur</div>
        <textarea id="desc" class="form-control" name="desc" ><?php
echo($annonce['ad_description']) ?></textarea>
    </div>
</div>
<div class="row form-group">
    <div class="col-md-2">
        <label for="adPhotoName" class="control-label">nom de la photo</label>
    </div>
    <div class="col-md-10">
        <div class="error-message">erreur</div>
        <input type="text" name="adPhotoName" class="form-control" id="adPhotoName"
value="<?php echo($annonce['ad_photo_name']) ?>">
    </div>
</div>

<div class="row form-group">
    <div class="col-md-2">
        <label for="adPhotoFolder" class="control-label">repertoire de la
photo</label>
    </div>
    <div class="col-md-10">
        <div class="error-message">erreur</div>
        <input type="text" name="adPhotoFolder" class="form-control"
id="adPhotoFolder" value="<?php echo($annonce['ad_photo_folder']) ?>">
    </div>
</div>

<div class="row form-group">
    <div class="col-md-2">
        <label for="adState" class="control-label">Estat</label>
    </div>
    <div class="col-md-10">
        <div class="error-message">erreur</div>
        <input type="text" name="adState" class="form-control" id="adState"
value="<?php echo($annonce['ad_state']) ?>">
    </div>
</div>

<div class="row form-group">
    <div class="col-md-2">
        <label for="adActivation" class="control-label">activation</label>
    </div>
    <div class="col-md-10">
        <div class="error-message">erreur</div>
        <input type="text" name="adActivation" class="form-control" id="adActivation"
value="<?php echo($annonce['ad_activate']) ?>">
    </div>
</div>
<hr>
<div class="form-group form-group">
    <div class="col-md-offset-2 col-md-10">
        <input type="submit" class="btn btn-success" id="envoyer" value="Mettre à
jour">
        <input type="button" class="btn btn-danger" value="Annuler">
    </div>
</div>
```

```
        </div></div>
    </div>
</div>
</form>
</body>
</html>
```

updateAnnonce.php

```
<?php

#-----
# Mise à jour de l'annonce
#-----

////fichier de sécurité
//require_once ("securite.php");

//Fichier de paramètres
$Proprietes = parse_ini_file("snap_ads.properties");

//Paramètres de connexion
$dbhost = $Proprietes ['serveur'];
$dbport = $Proprietes['port'];
$dbuser = $Proprietes['user'];
$dbuserPass = $Proprietes['mdp'];
$dbname = $Proprietes['bd'];
$dbpass = '';

//Je recuperer la valeur des champs saisis
$code = filter_input(INPUT_POST, 'code');
$adtitle = filter_input(INPUT_POST, 'adTitle');
$adbegindate = filter_input(INPUT_POST, 'adBeginDate');
$adenddate = filter_input(INPUT_POST, 'adEndDate');
$newprice = filter_input(INPUT_POST, 'newPrice');
$oldprice = filter_input(INPUT_POST, 'oldPrice');
$adlong = filter_input(INPUT_POST, 'adLong');
$adlat = filter_input(INPUT_POST, 'adLat');
$addesc = filter_input(INPUT_POST, 'desc');
$adphotoname = filter_input(INPUT_POST, 'adPhotoName');
$adphotofolder = filter_input(INPUT_POST, 'adPhotoFolder');
$adstate = filter_input(INPUT_POST, 'adState');
$adactivation = filter_input(INPUT_POST, 'adActivation');
$category = filter_input(INPUT_POST, "adCategory");
$location = filter_input(INPUT_POST, 'location');
$user = filter_input(INPUT_POST, 'user');

// la requête
$sql= "UPDATE ads SET ad_title=?, ad_begin_date=?, ad_end_date=?, ad_actual_price=?, ad_old_price=?,
ad_pos_longitude=?, ad_pos_latitude=?, ad_description=?, ad_photo_name=?, ad_photo_folder=?,
ad_state=?, ad_activate=?, id_category=?, id_location=?, id_user=? WHERE id_ad=?"; 

//Je me connecte
try {
//création d'un objet connection
```

	Rapport de Stage Développeur logiciel	page 43/57 <i>back-office</i>
---	---	----------------------------------

```

$pdo = new PDO("mysql:host=$dbhost;port=$dbport;dbname=$dbname", $dbuser, $dbuserPass);

//préparation de la requête
$ps=$pdo->prepare($sql);

//je crée un tableau de paramètres
$params = array($adtitle, $adbegindate, $adenddate, $newprice, $oldprice, $adlong, $adlat, $addesc,
$adphotename, $adphotofolder, $adstate, $adactivation, $category, $location, $user, $code);

//je passe les paramètres à l'exécution
$ps->execute($params);
echo ('Tout es OK');

//redirection
header("location:annonces.php");

//en cas d'erreur je lève une exception
} catch (PDOException $e) {
    $messsage = $e->getMessage();
}

//print_r($messsage);
?>

```

supprimerAnnonce.php

```

<?php

#-----
# Suppression d'une annonce
#-----

//Fichier de paramètres
$Proprietes = parse_ini_file("snap_ads.properties");

//Paramètres de connexion
$dbhost = $Proprietes ['serveur'];
$dbport = $Proprietes['port'];
$dbuser = $Proprietes['user'];
$dbuserPass = $Proprietes['mdp'];
$dbname = $Proprietes['bd'];
$dbpass = '';

//Je récupère la valeur de l'ID de l'annonce
$code=$_GET['code'];

//requête
$sql = "DELETE FROM ads WHERE id_ad=?";

//nouvel objet de connexion
$pdo = new PDO("mysql:host=$dbhost;port=$dbport;dbname=$dbname", $dbuser, $dbuserPass);

//tableau de paramètres
$params = array($code);

//Préparation de la requete
$ps = $pdo->prepare($sql);

//exécution de la requete
$ps->execute($params);

//Parcours des enregistrements

```

SŌZŌ UNLTD	Rapport de Stage Développeur logiciel	page 44/57 <i>back-office</i>
----------------------	--	----------------------------------

```
$ad=$ps->fetch();

//Redirection
header("location:annonces.php");

?>
```

5.2. Mobile

selectallads.html

```
<!DOCTYPE html>

<html>

    <title></title>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.1/css/bootstrap.min.css">
        <!--      <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.1/css/bootstrap-theme.min.css">-->
    <script src="js/jquery.js"></script>
    <script type="text/javascript" src="js>SelectAllAds.js"></script>

</head>
<body>

    <ul id="ads" class="list-group"></ul>

</body>
</html>
```

selectallads.js

```
function init() {
    selecAllAds();
}

var ads;

function selecAllAds() {
    var url = "php/selectallads.php";
    console.log(url);

//création d'un objet de connexion de type Jquery XMLHttpRequest
    var jqXHR = $.post
        (url,
        {
        })
```

```
);

jqXHR.done(function (data) {
    console.log(data);

    //on parcours jeux d'enregistrement au format
    var json = $.parseJSON(data);

    //Recupération des éléments avec un boucle
    for (var i = 0; i < json.length; ++i)
    {
        //insertion des éléments pour
        $('#ads').append('<li class="list-group-item"><a class="ad" href="addetails.html?id=' +
        json[i].id + '"><span class="title">' +
        json[i].ad_title +
        '</span> <span class="category">' +
        json[i].id_category +
        '</span><span class="ad_actual_price">' +
        json[i].ad_actual_price +
        '</span> <span class="ad_old_price">' +
        json[i].ad_old_price +
        '</span> <span class="ad_end_date">' +
        json[i].ad_end_date +
        '</span></a></li>');
    }
});

//en cas d'erreur
jqXHR.fail(function (xhr, statut, erreur) {
    console.log("Erreur : " + xhr.status + "-" + xhr.statusText);
    $('#Resultat').html("Erreur : " + xhr.status + "-" + xhr.statusText);
    alert("Failed");
});

}

//au chargement de mon document
$(document).ready(init);

selectallads.php
```

<?php

```
$Proprietes = parse_ini_file("snap_ads.properties");
$dbhost = $Proprietes ['serveur'];
$dbport = $Proprietes['port'];
$dbuser = $Proprietes['user'];
$dbuserPass = $Proprietes['mdp'];
$dbname = $Proprietes['bd'];
$dbpass = '';

//echo 'On est ICI';
```

| | | |
|---|---|-----------------------------|
|  | Rapport de Stage
Développeur logiciel | page 46/57
<i>Mobile</i> |
|---|---|-----------------------------|

```

$sql = "SELECT * FROM ads";

try {
  //Instanciation d'un objet connexion
  $pdo = new PDO("mysql:host=$dbhost;port=$dbport;dbname=$dbname", $dbuser,
$dbuserPass);

  // Gestion des erreurs en mode Exception
  $pdo->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
  $pdo->exec("SET NAMES 'UTF8'");

  //préparation de la requete
  $stmt = $pdo->prepare($sql);
  //exécutiond de la requête
  $stmt->execute();

  //Parcours du jeu de résultats
  $datas = $stmt->fetchAll(PDO::FETCH_ASSOC);
  //fermeture de la connection
  $lcn = null;
  //convertit un tableau PHP en un objet JSON que pourra lire Javascript.
  $lsResultas = json_encode($datas);

} catch (PDOException $ex) {
  echo 'Il y a un problème';
  $messsage = $ex->getMessage();
}

print_r($lsResultas);

```

6. Sécurité

7. Conclusion

8. Annexes

8.1.1. Script de création de la base de données

```
-- Schema snapads
-----
CREATE SCHEMA IF NOT EXISTS `snapads` DEFAULT CHARACTER SET utf8 ;
USE `snapads` ;

-- Table `snapads`.`categories`
-----
DROP TABLE IF EXISTS `snapads`.`categories` ;

CREATE TABLE IF NOT EXISTS `snapads`.`categories` (
  `id_category` INT(11) NOT NULL AUTO_INCREMENT,
  `category_name` VARCHAR(45) NOT NULL,
  `category_activate` TINYINT(4) NOT NULL,
  PRIMARY KEY (`id_category`),
  UNIQUE INDEX `category_name` (`category_name` ASC)
ENGINE = InnoDB
DEFAULT CHARACTER SET = latin1;

-- Table `snapads`.`locations`
-----
DROP TABLE IF EXISTS `snapads`.`locations` ;

CREATE TABLE IF NOT EXISTS `snapads`.`locations` (
  `id_location` INT(11) NOT NULL,
  `street_name` VARCHAR(45) NOT NULL,
  `city_name` VARCHAR(32) NOT NULL,
  `zipcode` INT(11) NOT NULL,
  `country` VARCHAR(32) NOT NULL,
  PRIMARY KEY (`id_location`)
ENGINE = InnoDB
DEFAULT CHARACTER SET = latin1;

-- Table `snapads`.`companies`
-----
DROP TABLE IF EXISTS `snapads`.`companies` ;

CREATE TABLE IF NOT EXISTS `snapads`.`companies` (
  `id_company` INT(11) NOT NULL AUTO_INCREMENT,
  `company_name` VARCHAR(45) NOT NULL,
  `company_siret` INT(11) NOT NULL,
  `id_location` INT(11) NOT NULL,
  `id_category` INT(11) NOT NULL,
  PRIMARY KEY (`id_company`),
  UNIQUE INDEX `company_siret` (`company_siret` ASC),
  INDEX `FK_companies_id_location` (`id_location` ASC),
  INDEX `FK_companies_id_category` (`id_category` ASC),
  CONSTRAINT `FK_companies_id_category`
    FOREIGN KEY (`id_category`)
    REFERENCES `snapads`.`categories` (`id_category`),
  CONSTRAINT `FK_companies_id_location`
    FOREIGN KEY (`id_location`)
    REFERENCES `snapads`.`locations` (`id_location`)
ENGINE = InnoDB
DEFAULT CHARACTER SET = latin1;
```

```
-- Table `snapads`.`users`
-----
DROP TABLE IF EXISTS `snapads`.`users` ;

CREATE TABLE IF NOT EXISTS `snapads`.`users` (
  `id_user` INT(11) NOT NULL AUTO_INCREMENT,
  `user_firstname` VARCHAR(32) NOT NULL,
  `user_lastname` VARCHAR(32) NOT NULL,
  `user_nickname` VARCHAR(32) NOT NULL,
  `user_login` VARCHAR(32) NOT NULL,
  `user_password` VARCHAR(32) NOT NULL,
  `user_email` VARCHAR(128) NOT NULL,
  `user_telephone` INT(11) NOT NULL,
  `user_last_pass_gen` DATETIME NOT NULL,
  `user_active` TINYINT(4) NOT NULL,
  `user_deleted` TINYINT(4) NOT NULL,
  `user_type` TINYINT(4) NOT NULL,
  `id_location` INT(11) NOT NULL,
  `id_company` INT(11) NOT NULL,
  PRIMARY KEY (`id_user`),
  UNIQUE INDEX `user_nickname` (`user_nickname` ASC, `user_login` ASC, `user_email` ASC),
  INDEX `user_firstname` (`user_firstname` ASC),
  INDEX `FK_users_id_location` (`id_location` ASC),
  INDEX `FK_users_id_company` (`id_company` ASC),
  CONSTRAINT `FK_users_id_company`
    FOREIGN KEY (`id_company`)
      REFERENCES `snapads`.`companies` (`id_company`),
  CONSTRAINT `FK_users_id_location`
    FOREIGN KEY (`id_location`)
      REFERENCES `snapads`.`locations` (`id_location`))
ENGINE = InnoDB
DEFAULT CHARACTER SET = latin1;

-----
```

```
-- Table `snapads`.`ads`
-----
DROP TABLE IF EXISTS `snapads`.`ads` ;

CREATE TABLE IF NOT EXISTS `snapads`.`ads` (
  `id_ad` INT(11) NOT NULL AUTO_INCREMENT,
  `ad_title` VARCHAR(64) NOT NULL,
  `ad_begin_date` DATE NOT NULL,
  `ad_end_date` DATE NOT NULL,
  `ad_actual_price` DECIMAL(20,0) NOT NULL,
  `ad_old_price` DECIMAL(20,0) NOT NULL,
  `ad_pos_longitude` DECIMAL(7,0) NOT NULL,
  `ad_pos_latitude` DECIMAL(7,0) NOT NULL,
  `ad_description` VARCHAR(255) NOT NULL,
  `ad_photo_name` VARCHAR(45) NULL DEFAULT NULL,
  `ad_photo_folder` VARCHAR(45) NULL DEFAULT NULL,
  `ad_state` TINYINT(4) NOT NULL,
  `ad_activate` TINYINT(4) NOT NULL,
  `id_category` INT(11) NOT NULL,
  `id_location` INT(11) NOT NULL,
  `id_user` INT(11) NOT NULL,
  PRIMARY KEY (`id_ad`),
  UNIQUE INDEX `ad_description` (`ad_description` ASC, `ad_photo_name` ASC),
  INDEX `FK_ads_id_category` (`id_category` ASC),
  INDEX `FK_ads_id_location` (`id_location` ASC),
  INDEX `FK_ads_id_user` (`id_user` ASC),
  CONSTRAINT `FK_ads_id_category`
    FOREIGN KEY (`id_category`)
      REFERENCES `snapads`.`categories` (`id_category`),
  CONSTRAINT `FK_ads_id_location`
    FOREIGN KEY (`id_location`)
```

```

    REFERENCES `snapads`.`locations` (`id_location`),
CONSTRAINT `FK_ads_id_user`
    FOREIGN KEY (`id_user`)
        REFERENCES `snapads`.`users` (`id_user`)
ENGINE = InnoDB
DEFAULT CHARACTER SET = latin1;

-- -----
-- Table `snapads`.`internaute`
-- -----
DROP TABLE IF EXISTS `snapads`.`internaute` ;

CREATE TABLE IF NOT EXISTS `snapads`.`internaute` (
`ip` VARCHAR(25) NOT NULL,
PRIMARY KEY (`ip`))
ENGINE = InnoDB
DEFAULT CHARACTER SET = latin1;

-- -----
-- Table `snapads`.`consult`
-- -----
DROP TABLE IF EXISTS `snapads`.`consult` ;

CREATE TABLE IF NOT EXISTS `snapads`.`consult` (
`id_ad` INT(11) NOT NULL,
`ip` VARCHAR(25) NOT NULL,
PRIMARY KEY (`id_ad`, `ip`),
INDEX `FK_consult_ip` (`ip` ASC),
CONSTRAINT `FK_consult_id_ad`
    FOREIGN KEY (`id_ad`)
        REFERENCES `snapads`.`ads` (`id_ad`),
CONSTRAINT `FK_consult_ip`
    FOREIGN KEY (`ip`)
        REFERENCES `snapads`.`internaute` (`ip`))
ENGINE = InnoDB
DEFAULT CHARACTER SET = latin1;

-- -----
-- Table `snapads`.`follow`
-- -----
DROP TABLE IF EXISTS `snapads`.`follow` ;

CREATE TABLE IF NOT EXISTS `snapads`.`follow` (
`id_ad` INT(11) NOT NULL,
`id_user` INT(11) NOT NULL,
PRIMARY KEY (`id_ad`, `id_user`),
INDEX `FK_follow_id_user` (`id_user` ASC),
CONSTRAINT `FK_follow_id_ad`
    FOREIGN KEY (`id_ad`)
        REFERENCES `snapads`.`ads` (`id_ad`),
CONSTRAINT `FK_follow_id_user`
    FOREIGN KEY (`id_user`)
        REFERENCES `snapads`.`users` (`id_user`))
ENGINE = InnoDB
DEFAULT CHARACTER SET = latin1;

-- -----
-- Table `snapads`.`interest`
-- -----
DROP TABLE IF EXISTS `snapads`.`interest` ;

CREATE TABLE IF NOT EXISTS `snapads`.`interest` (
`id_category` INT(11) NOT NULL,
`id_user` INT(11) NOT NULL,
PRIMARY KEY (`id_category`, `id_user`),

```

| | | |
|----------------------|----------------------|---|
| SŌZŌ
UNLTD | Rapport de Stage | page 52/57 |
| | Développeur logiciel | <i>Script de création de la base de données</i> |

```

INDEX `FK_interest_id_user` (`id_user` ASC),
CONSTRAINT `FK_interest_id_category`
  FOREIGN KEY (`id_category`)
    REFERENCES `snapads`.`categories` (`id_category`),
CONSTRAINT `FK_interest_id_user`
  FOREIGN KEY (`id_user`)
    REFERENCES `snapads`.`users` (`id_user`))

ENGINE = InnoDB
DEFAULT CHARACTER SET = latin1;

-----  

-- Table `snapads`.`settings`  

-----  

DROP TABLE IF EXISTS `snapads`.`settings` ;  

CREATE TABLE IF NOT EXISTS `snapads`.`settings` (
  `id_setting` INT(11) NOT NULL AUTO_INCREMENT,
  `setting_notification` TINYINT(4) NOT NULL,
  `setting_newsletter` TINYINT(4) NOT NULL,
  `setting_radius` TINYINT(4) NOT NULL,
  `setting_user_categories` TINYINT(4) NOT NULL,
  `id_user` INT(11) NOT NULL,
  PRIMARY KEY (`id_setting`),
  UNIQUE INDEX `setting_notification` (`setting_notification` ASC, `setting_newsletter` ASC, `setting_radius` ASC, `setting_user_categories` ASC),
  INDEX `FK_settings_id_user` (`id_user` ASC),
  CONSTRAINT `FK_settings_id_user`
    FOREIGN KEY (`id_user`)
      REFERENCES `snapads`.`users` (`id_user`))

ENGINE = InnoDB
DEFAULT CHARACTER SET = latin1;

```

8.1.2. Les langages (description)



HTML (*HyperText Mark-Up Language*) est un langage dit de « marquage » (de « structuration » ou de « balisage ») dont le rôle est de formaliser l'écriture d'un document avec des balises de formatage. Les balises permettent d'indiquer la façon dont doit être présenté le document et les liens qu'il établit avec d'autres documents.



CSS (*Cascading Style Sheets*) sont utilisés pour définir la présentation des pages Web. Couplé au langage HTML les CSS vous permettront de créer vos styles et de définir le design général de votre site Internet.



PHP: Hypertext Preprocessor est un langage de scripts coté serveur qui permet de générer dynamiquement des pages (réécrites à chaque exécution) en HTML/XHTML permettant de créer des applications web.



Javascript est un langage de script coté client, c'est à dire qu'il est exécuté sur le navigateur. Il complète le HTML et le CSS en y ajoutant une couche d'interactivité. C'est un langage orienté objet à prototype (sans classe).

| | | |
|----------------------|----------------------|-----------------------------|
| SŌZŌ
UNLTD | Rapport de Stage | page 53/57 |
| | Développeur logiciel | <i>Les outils (détails)</i> |

8.1.3. Les outils (détails)



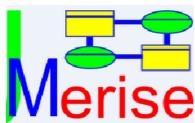
XAMPP est un ensemble de logiciels qui permet de mettre en place facilement un serveur Web , un serveur FTP et un serveur de messagerie électronique. Il s'agit d'une distribution de logiciels libres (X (cross) Apache MariaDB Perl PHP).



Github est un logiciel de gestion de versions qui permet de stocker un ensemble de fichiers en conservant la chronologie de toutes les modifications qui ont été effectuées dessus. Il permet notamment de retrouver les différentes versions d'un lot de fichiers connexes.



C'est un l'environnement de Développement Intégré (EDI) sous licence libre qui propose un éditeur de code intelligent multi-langage, différent des modèles de projets, un créateur d'interfaces graphiques, un simulateur,un compilateur, et un débogueur.



JMerise : est un logiciel dédié à la modélisation des modèles conceptuels de données (MCD) pour Merise il permet la généralisation et la spécialisation des entités, la création des relations et des cardinalités ainsi que la généralisation des modèles logiques de données(MLD) et des script SQL.



MySQL Workbench est un logiciel de gestion et d'administration de bases de données MySQL qui permet via une interface graphique , de créer, modifier ou supprimer des tables, des comptes utilisateurs, et d'effectuer toutes les opérations inhérentes à la gestion d'une base de données.



phpMyAdmin est une application Web développée en PHP , offrant une interface pour la gestion des bases de données MySQL. Il est ainsi possible, grâce à un simple navigateur Internet de procéder à des opérations permettant de modifier / supprimer des enregistrements, sauvegarder sa BD.



Le CLI (Command Line Interface)CMD est une interface qui permet une communication directe entre l'utilisateur et le système d'exploitation. Elle permet d'exécuter divers programmes en ligne de commande en offrant la possibilité de passer des paramètres appropriés à chaque exécution d'un programme. Les communication s'effectuent en mode texte.

Apache Cordova (ex phonegap 1) est un framework open-source développé par la Fondation Apache. Il permet de créer des applications pour différentes plateformes (Android,Firefox

| | | |
|----------------------|--|---|
| SŌZŌ
UNLTD | Rapport de Stage
Développeur logiciel | page 54/57
<i>Les outils (détails)</i> |
|----------------------|--|---|

OS, iOS, Ubuntu, Windows 8...) en HTML, CSS et JavaScript.



Présenté par Google comme l'IDE officiel comme Android Studio est IDE pour développer des applications Android. Il embarque en autre un AVD (Android Virtual Device) qui un est un émulateur capable de n'imorte quel plateforeme Android .



Le JDK (Java Development Kit), est le kit de développement de base que propose gratuitement la société Oracle. un ensemble d'outils pour compiler et déboguer le code JAVA (il embarque le L'environnement d'exécution de Java (JRE)



Node.js est une plateforme de développement Javascript. Node Js le langage Javascript avec des bibliothèques permettant de réaliser des actions comme écrire sur la sortie standard, ouvrir/fermer des connections réseau ou encore créer un fichier.

8.1.4. Glossaire

8.1.5. Webographie

- fr.wikipedia.org
- <https://www.grafikart.fr/>
- <http://php.net/manual/fr/>
- <http://www.developpez.com/>

8.1.6. Bibliographie

Pascal Roques, 2006, UML2 modeliser une application web, ed. Eyrolles, 4eme edition

8.1.7. Diagramme de Gantt

