

# Projet Android : l'entreprise GestionBat

L'entreprise GestionBat a comme principale activité la vente de biens immobiliers auprès d'investisseurs qui sont le plus souvent des SCI (Société Civile Immobilière).

Trois commerciaux sont chargés de la recherche et de la vente de ce type de biens. Afin de mutualiser le portefeuille d'affaires (biens et clients), M. Gagnant, le directeur de l'agence souhaite la mise en place d'une application Web.

Une première approche a été menée afin de réaliser un premier prototype incluant l'usage de téléphones sous Android. Cette application devra permettre :

- de gérer des affaires (biens immobiliers)
- de gérer les contacts (les investisseurs)

**Vous devez respecter scrupuleusement les noms (base de données, champs, tables, classes, layout) évoqués dans le cahier des charges.**

**Vous trouverez dans `_ressource` le répertoire `projetGestionBat` contenant tous les fichiers à intégrer dans votre projet (layout, string, web service php, url)**

## Cahier des charges

### Base de données : `bdGestionBat`

`affaire(_id:int(10), datea:date,prix:bigint(10),adr:varchar(30),ville:varchar(30), typeaf:char(10))`  
`contact(_id:int(10), nomsoc:varchar(30),contact:varchar(30), tel:varchar(10),idaffaire:int(10))`

**typeaf** : Studio, F1, F2, F3, F4, F5, Maison

*Remarque* : les identifiants ont comme dénomination **`_id`**, ceci afin de pouvoir lier facilement les curseurs au `ListView`. Chaque identifiant est « auto-incrément ».

### Web Service

Côté serveur (WAMP) il va être nécessaire de créer un répertoire **GestionBat** sous **WWW**.

Dans le répertoire **GestionBat** vous y copierai le répertoire **plumwebservice** présent dans `_ressource/ ../projetGestionBat`.

Le web service peut-être testé grâce au script **testWebService.php**.

### L'adresse URL du web service sera :

["http://10.0.2.2/GestionBat/plumwebservice/plumWebServiceDataBase.php"](http://10.0.2.2/GestionBat/plumwebservice/plumWebServiceDataBase.php)

### Projet

**Vous devez respecter la casse (majuscule/minuscule) et les nommages qui suivent.**

- Project name : **Nom1Nom2GestionBat** (par exemple SoulatTorrentGestionBat)
- Application Name : **GestionBat**
- Package Name : **nom1.nom2.gestionbat** (par exemple soulat.torrent.gestionbat)
- Create Activity : ListeAffaire

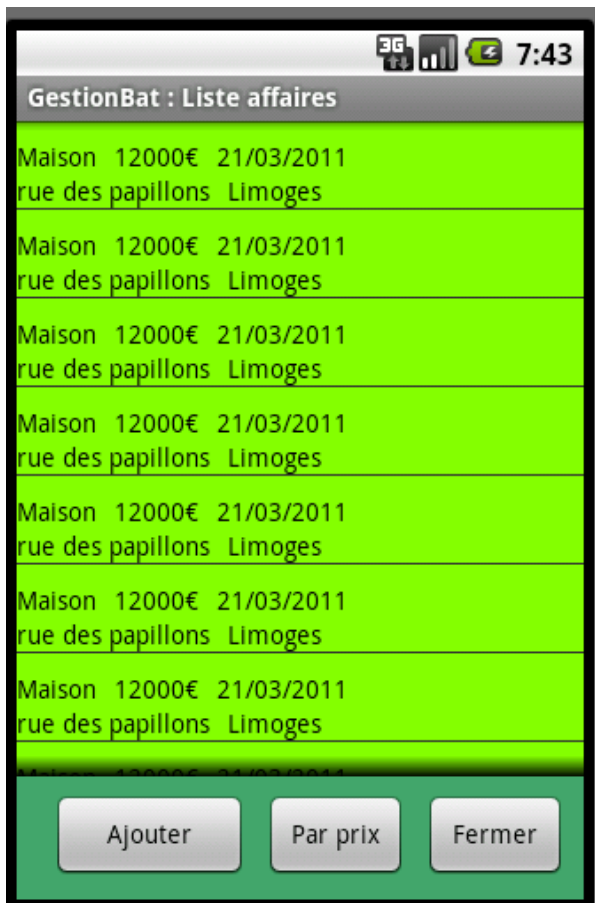
## Maquettes

Seules les maquettes sont exposées ici. Chaque maquette est liée à un contrôleur [Activity]. Chaque contrôleur gère une seule vue [Layout].

main.xml [Activity ListeAffaire.java]

### Précisions :

- les affaires sont triées par défaut sur **typeaf**. Il doit être possible de faire un affichage avec un tri sur le prix.
- Le fichier **liste\_item\_affaire.xml** décrit l'affichage de chaque ligne de la liste des affaires.
- le bouton **Ajouter** permet de démarrer l'Activity AjouterAffaire. Au retour (ré-activation) vous devez rafraichir la liste des affaires.
- le bouton **Par prix** permet d'afficher les affaires ordonnées sur le prix de manière croissante.
- Le bouton **Fermer** termine l'application.
- Le **choix d'une affaire** dans la liste démarre l'Activity **ModSupAffaire**.



### layouts

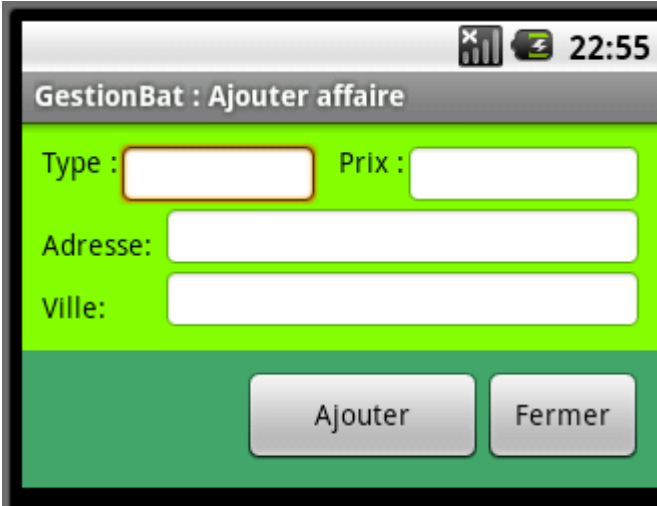
main.xml

liste\_item\_affaire.xml

## ajouter\_affaire.xml [Activity AjouterAffaire.java]

### Précisions :

- le bouton **Ajouter** permet d'insérer une affaire, une fois celle-ci insérée vous devez fermer l'Activity et revenir sur le liste des affaires.
- Le bouton **Fermer** termine l'Activity.

The screenshot shows an Android application window titled 'GestionBat : Ajouter affaire'. The status bar at the top displays the time as 22:55. The main content area has a yellow background and contains four text input fields: 'Type :', 'Prix :', 'Adresse:', and 'Ville:'. The 'Type :' field is highlighted with a red border. At the bottom, there is a green bar containing two buttons: 'Ajouter' and 'Fermer'.

### layout

ajouter\_affaire.xml

## mod\_sup\_affaire.xml [Activity ModSupAffaire.java]

### Précisions :

- sur le choix d'une affaire depuis la liste des affaires [**ListeAffaire.java**], l'Activity **ModSupAffaire** affiche les informations modifiables (toutes sauf la date et l'identifiant).
- le bouton **Modifier** permet de modifier une affaire, une fois celle-ci modifiée vous devez fermer l'Activity et revenir sur le liste des affaires.
- le bouton **Supprimer** permet de supprimer une affaire et tous ses contacts, une fois celle-ci supprimée vous devez fermer l'Activity et revenir sur le liste des affaires.
- Le bouton **Fermer** termine l'Activity.
- les contact sont triées sur **nomsoc**.
- Le fichier **liste\_item\_contact.xml** décrit l'affichage de chaque ligne de la liste des contact pour une affaire.
- Le bouton **Ajouter un contact** démarre l'Activité **AjouterContact**.
- La liste des contacts doit être actualisée au moment de la ré-activation de l'Activity **ModSupAffaire**.

GestionBat : Modifier/Supprimer affaire

Type : F4 Prix : 120000

Adresse: rue du petit cheval

Ville: Couzeix

Modifier Supprimer Fermer

SCI truc M. Jean 0555376655

SCI truc M. Jean 0555376655

SCI truc M. Jean 0555376655

Ajouter un contact

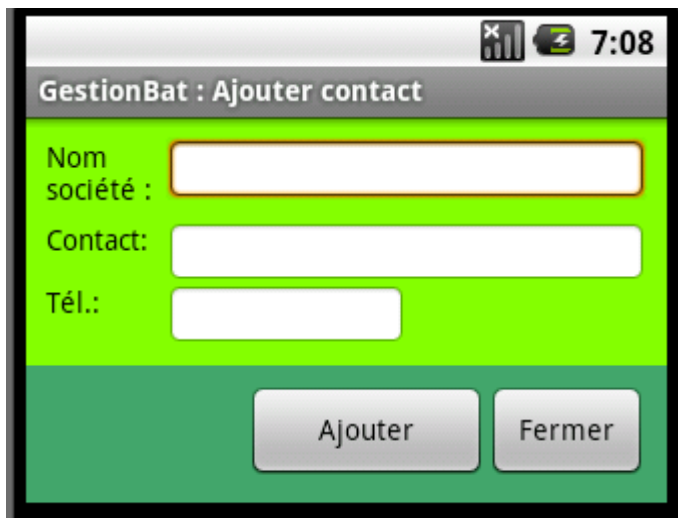
### layouts

mod\_sup\_affaire.xml  
liste\_item\_contact.xml

### ajouter\_contact.xml [Activity AjouterContact.java]

#### Précisions :

- le bouton **Ajouter** permet d'insérer un contact, une fois celui-ci inséré vous devez fermer l'Activity et revenir sur l'activity ModSupAffaire.
- Le bouton **Fermer** termine l'Activity.

The screenshot shows a mobile application interface titled 'GestionBat : Ajouter contact'. The background is a solid light blue. There are three text input fields: the first is labeled 'Nom société :', the second 'Contact:', and the third 'Tél.:'. Each field has a red border. At the bottom of the screen, there are two grey buttons with black text: 'Ajouter' on the left and 'Fermer' on the right. The top status bar shows a battery icon, a signal strength icon, and the time '7:08'.

#### layout

ajouter\_contact.xml

### mod\_sup\_contact.xml [Activity AjouterContact.java]

Proposez et implémentez une solution afin de permettre la modification ou la suppression d'un contact à partir de l'Activity **ModSupAffaire**. Dans ce cas l'utilisateur choisira un contact dans la liste afin de le modifier ou de le supprimer.

## Table des matières

Projet Android : l'entreprise GestionBat.....	1
Cahier des charges.....	1
Base de données : bdGestionBat.....	1
Web Service.....	1
Projet.....	1
Maquettes.....	2
main.xml [Activity ListeAffaire.java].....	2
ajouter_affaire.xml [Activity AjouterAffaire.java].....	3
mod_sup_affaire.xml [Activity ModSupAffaire.java].....	4
ajouter_contact.xml [Activity AjouterContact.java].....	5
mod_sup_contact.xml [Activity AjouterContact.java].....	5