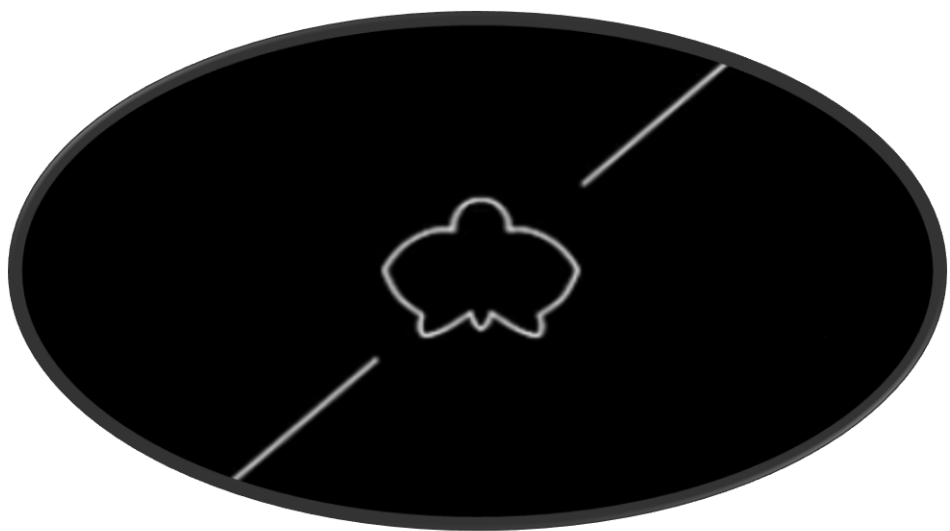




# ***RAPPORT DE STAGE***

## *ORCHID CREATION PARIS*



M2I formation  
24/10/2016 – 12/05/2017  
CDI  
Lutin Mickael



# Sommaire

---

Chapitre 1 – Présentation Général .....	5
1-1 : Mon Parcours Professionnel .....	6
1-2 : Abstract .....	7
1-3: Remerciements.....	8
1-4: Présentation de l'entreprise .....	9
Chapitre 2 - Cahier des Charges .....	11
2-1: Présentation du Projet .....	12
2-2: Déroulement du Projet .....	13
2-3: Le cahier des charges .....	14
Chapitre 3 – Analyse et Conception UML .....	15
3-1: Démarche .....	16
3-2 Analyse des besoins .....	17
3-3 Diagramme de contexte statique .....	19
3-4 Le diagramme de cas d'utilisation DCU) .....	20
3-5 Diagramme d'activité .....	21
3-5-1 exemple des pas d'authentification .....	22
3-5-2 Diagramme d'activité ( détaillé) .....	23

# Sommaire ...

---

3 -6 Le diagramme de navigation .....	23
3-7 OrchidCreation Maquettes .....	25
3-7-1 maquettes Formulaire .....	25
3 -7-2 Tableau du personnel apres validation	25
3-7-3 Maquettes Chart .....	26
3-7-4 Connexion.....	27
3-7-5 Login.....	27
3-8 Diagramme de Séquence Système (DSS) .....	28
3-8-1 DSS authentication .....	28
3-8-2 DSS projet .....	29
• <b>Chapitre 4 Conception de la Base de Données</b> .....	30
4-1 Diagramme de Classe .....	31
4-2 Model conceptuel de données MCD (MERISE) .....	32
4-3 Modèle logique de données (MLD) .....	33
4-4 Conception de la Base de Données .....	34
4-5 Modèle Physique des Données .....	35
4-6 Explication de mes Tables .....	36
4-7 Creation de Code SQL de ma Base de Données .....	38
4-8 Diagramme de déploiement .....	42
• <b>Chapitre 5 Conception de l'application</b> .....	43
5-1 L'architecture MVC .....	44
5-2 Arborescence de mon projet Symfony .....	45
5-3 Conception de Porjet Sheet .....	46
5-3-1 Exemple de code HTML .....	54
5-3-2 Code Php creation d'un formulaire .....	55
5-3-3 Page du site .....	56
• <b>Chapitre 6 La securiter</b> .....	57
6-1 La securiter de mon application.....	58

# Sommaire

---

•Chapitre 7 Conclusion .....	59
7-1 Conclusion Général .....	60
•Chapitre 8 Annexes.....	61
8-1 Webographie et Bibliographie .....	62
8-2 Correspondance Projet et Reac .....	63
8-3 Technologie Choisie .....	64

# **1 – Présentation Général**

# 1-1 Mon Parcours Professionnel

---

Après avoir obtenue mes diplômes en Restauration ( BEP , Mention complémentaire et un Bac Pro ) je me suis forgé une expérience dans le monde professionnelle qui dur déjà depuis 8 ans, j'ai décidé de me réorienter vers l'informatique . En Octobre 2016, j'ai intégré le centre de formation « M2I » afin d'obtenir mon diplôme entant que **Concepteur Développeur Informatique**.

En Mars 2017, j'ai intégré une Start Up « OrchidCreation » pour une période de stage qui a duré 2 mois. Ce stage m'a parmi d'obtenir ma première expérience professionnelle entant que Développeur Informatique dans un groupe de 8 personnes tous Graphiste. Cela m'a permis d'être autonome et de trouvé les solutions par moi-même.

## 1-2 Abstract

Durant cette période de stage je me suis avant tout de choisir la bonne méthodologie de travaille. C'était ma première expérience professionnel dans ce domaine donc je partais pas en terrain connue , j'ai du batailler pour résoudre les problèmes et essayer de rendre un travaille correct même si faute de temps j'ai pas pu finir. Mais mon tuteur savais d'où je venais a été plutôt cool et compréhensible.

Maintenant je vais pour présenter mon travaille effectuer

-----  
*During this period of internship I have above all to choose the right methodology of work. It was my first professional experience in this field so i was not in a position of strength, i had to struggle to solve my problems and try to make a work correct even if time was short I could not finish. But my tutor knew where i was coming from and was pretty cool and understandable.  
Now I go to present my work performed*

# 1- 3 Remerciements

---

Je tiens , tout particulièrement , remercier mes formateur Messieurs **BUGUET Pascal**, **MALORON Sébastien** pour leurs disponibilités , leurs enseignements durant ces 4 mois de formation et 2 mois de stage. Sans oublier **Jérémie** et **Pierre** qui sont intervenue pendant 1 moi et demi pour nous apprendre le Langage ( PHP 5) et son formidable Framework ( Symfony 2).

Je remercie aussi l'équipe d'OrchidCreation qui ont choisi de m'accueillir durant c'est deux mois de Stage.

Mais aussi a mes camarades de formation avec qui j'ai pu progresser durant ces mois

Merci à tous !



# 1-4 Présentation de l'entreprise

---

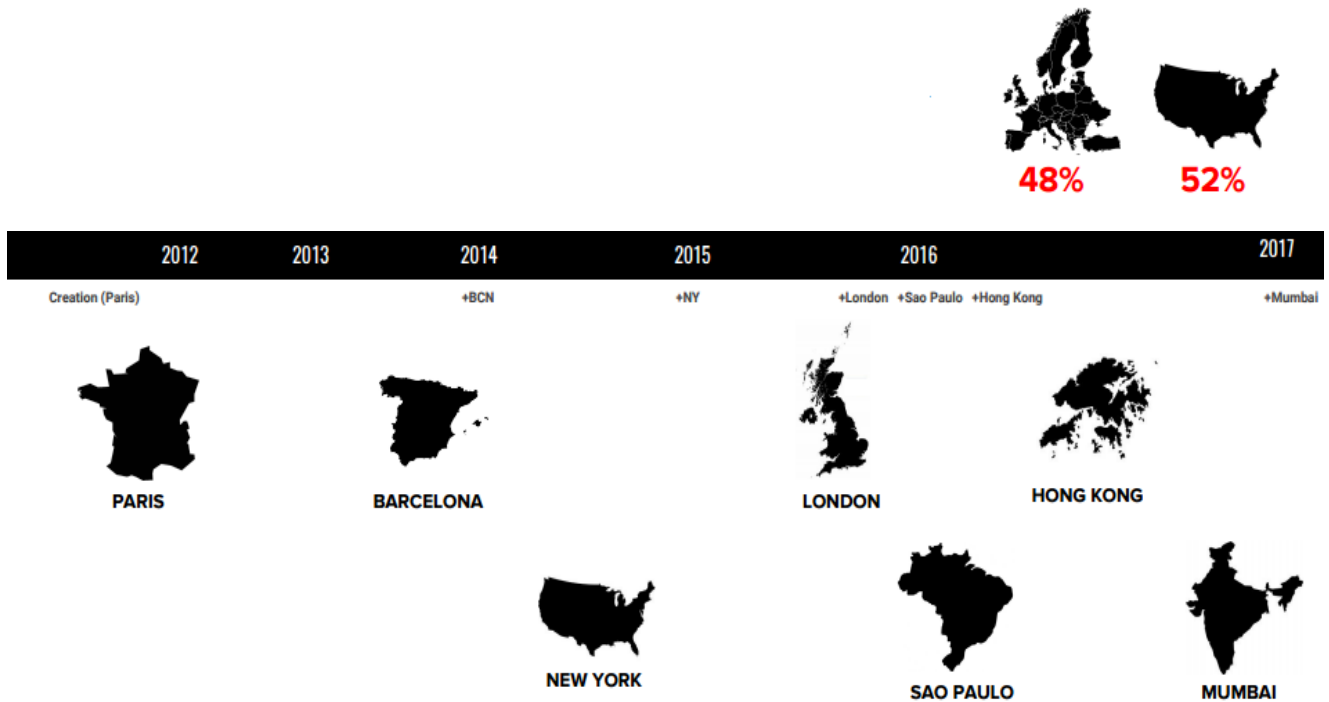


Créer en 2012 Orchid Creation est une agence de communication (de style star Up) spécialisée dans le design de contenus. C'est spécialisée dans l'enrichissement et le redesign de présentations Power Point , Keynote, Google slides, vidéos, écrans digitaux....

À partir de sources souvent peu attractives, l'agence repense l'ensemble du déroulé et anime chaque transition et slide.

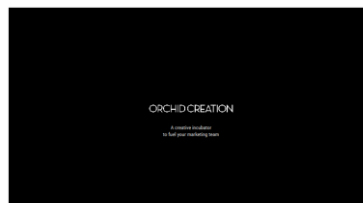
Le group est composer généralement que de graphiste 2D 3D , et motion Disigner.

Le premier OrchidCreation à avoir été créé est celui de Paris qui a été suivi de 6 autres grande Villes (Barcelone, New York, London Sao Paulo, Hong Kong et récemment Bombay) .



Le group a de nombreux clients de grande importance comme :

- LVMH
- Google
- L'Oréal
- Youtube

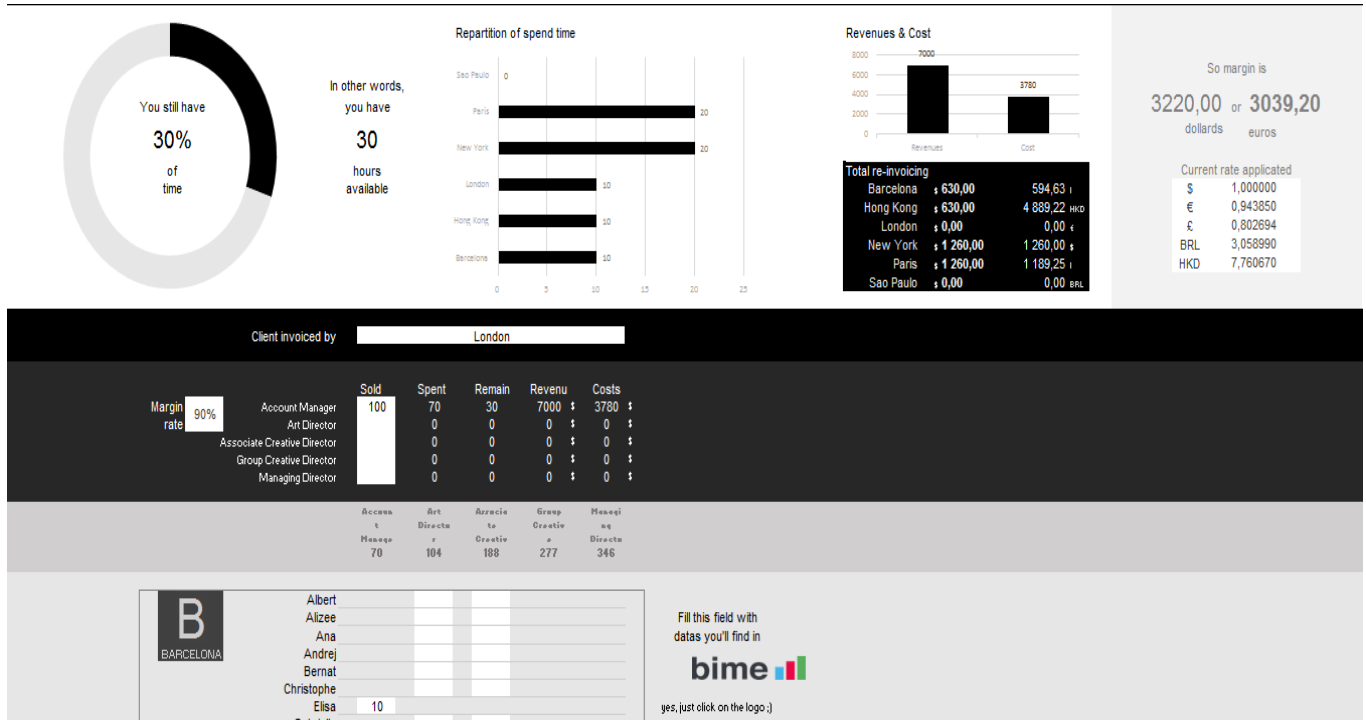


Exemple de projet format PDF

# **Chapitre 2 – Cahier des Charges**

## 2 - 1 Présentation du Projet

Mon travail au sein de cette entreprise consistait à créer une application web à partir d'un fichier Excel.

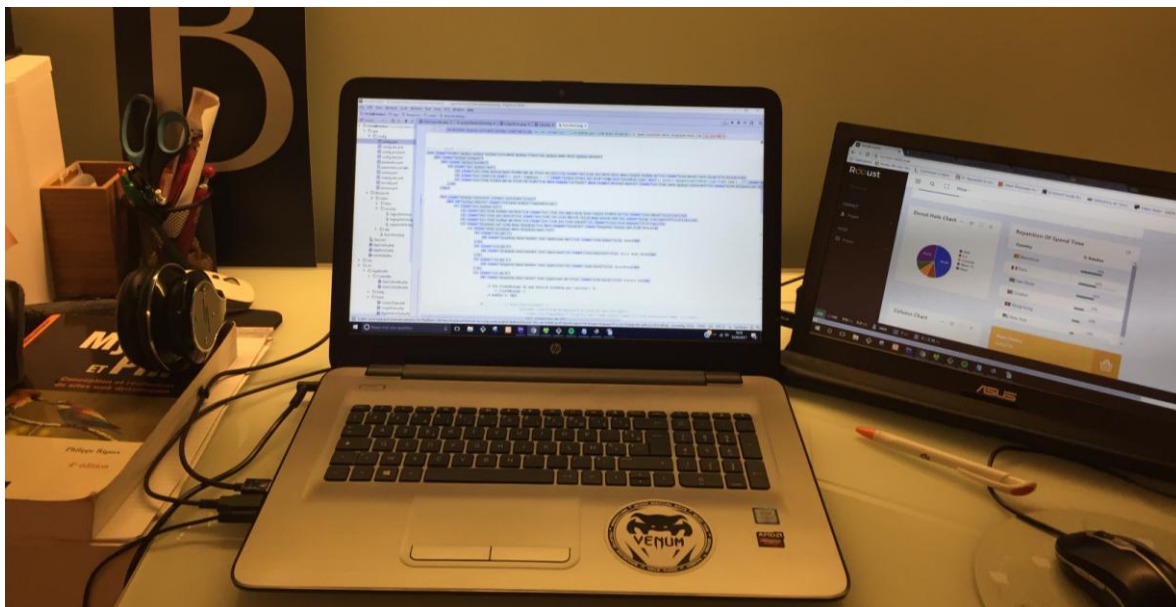


Feuille de calcul  
Microsoft Excel

Ce fichier Excel consiste à Répertorier le Temps de travail de chaque ville, et de calculé les couts. Ce fichier étant utilisé en interne par chaque responsable de chaque ville, il me fallait de créer une application qui sera consultable pour chacune d'elle en même temps via ma page web que je vais créer.

## 2 - 2 Déroulement du Projet

---



Lors de mon premier jour dans cette entreprise, on m'a assigné une place, ajouté sur « Google Hangout » pour discuter du projet avec les autres membres et donné les codes.

J'ai ensuite choisi mon environnement de travail.

J'étais libre d'utiliser les méthodologies, environnements de travail.

Mais j'avais quelques technologies imposées qui étaient :

- MySQL pour la base de données,
- HTML / JavaScript,
- PHP

Pour le reste j'avais libre choix, donc j'ai décidé de travailler sur « symfony ». Mon tuteur connaissait pas ce Framework donc après lui avoir expliqué son fonctionnement il m'a donné la patte blanche pour l'utiliser.

## 2.3 Le cahier des charges

---

-Le site disposera d'un espace pour envoyer des mails au autres employer de la boite a paris et dans les autres villes d'autres pays (Rio, Londres, New York, Hong Kong, Paris et Barcelone) .

-Un espace pour consulté les projets en cours.

-Création d'une Base de Données ( il y'avait aucune Base de Données)

-Un espace pour consulté les Statiques du Temps restant et des Couts ( Mon projet sera principalement porter sur cette partie ).

-J'ai aussi proposé a mon tuteur de mettre un accès accessible seulement au employés vu qu'a la base c'est juste une Application web, il a directement valider, donc j'ai donc créer une partie authentication.

Il y aura deux principaux acteurs l'Employer simple qui pourras juste consulté les projets , envoyer des mails. Et le Chef de Projet qui ajoutera les projet, modifiera et aura pourras modifier la page de Statistique que j'ai créer.

# **Chapitre 3 – Analyse et Conception**

### 3.1. Démarche

Pour la conception de notre application j'ai utilisé le langage de modélisation UML (Unified Modeling Language) en me basant sur l'approche de M. Pascal Roques

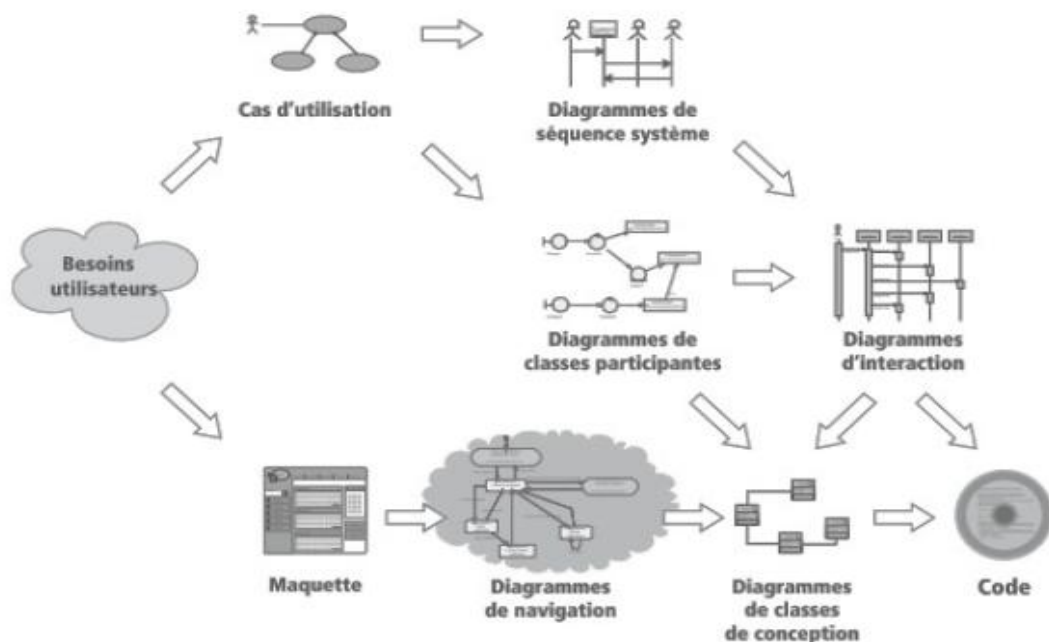


Schéma du processus de modélisation de l'application



## 3-2Analyse des besoins

---

### Exigences fonctionnelles

- Authentification
- consultation détaillée des annonces
- envoi de messages
- modération des annonces

### Exigences non fonctionnelles

- Interface simple et efficace
- Organisation claire et structurée de l'information

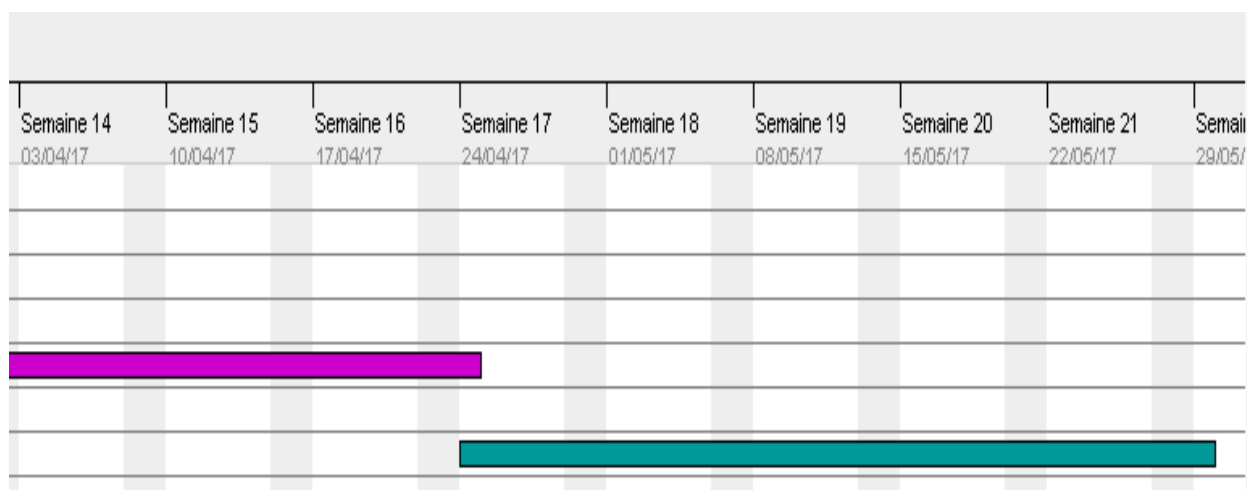
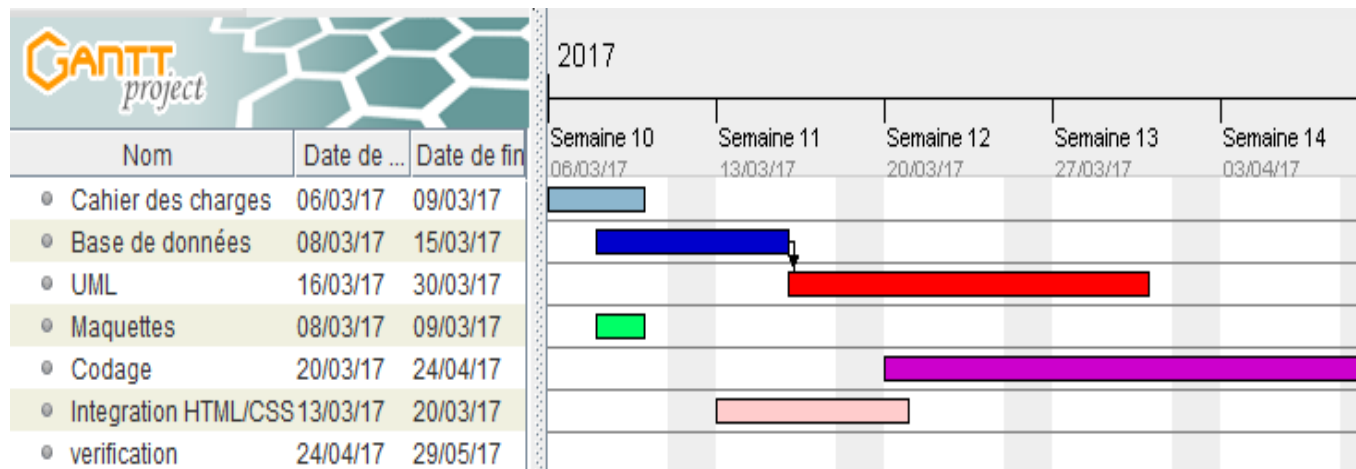
### Contraintes de conception

- Les annonces seront stockées dans une base de données

### 3-2-1 Planning 5 ( Diagramme de Gantt )

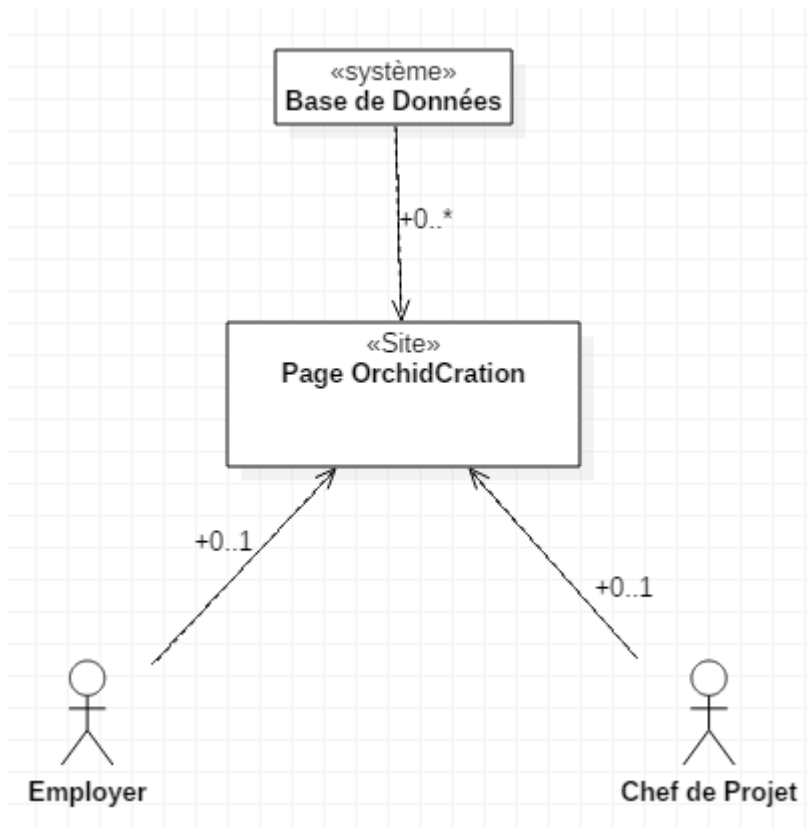
Pour réaliser mon diagramme de Gantt je me suis basé sur 7 étapes de mon projet ( Projet + rapport de stage ).

- Cahier des charges
- Base de Données
- UML
- Maquettes
- Codage
- Intégration HTML/CSS/Boostrp
- Vérification



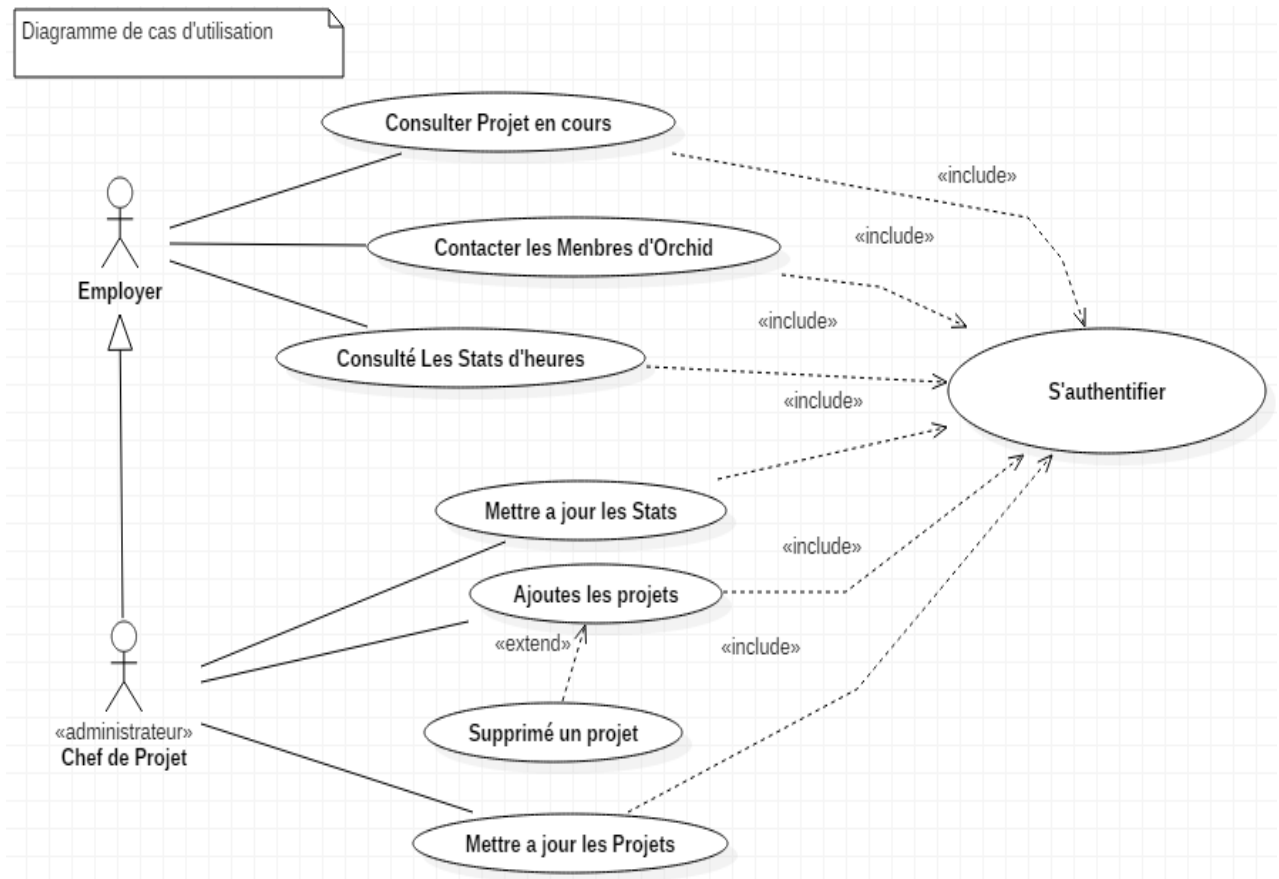
## 3-3 Diagramme de contexte statique

Ce diagramme n'a d'intérêt que lorsqu'il y a de nombreux acteurs. Les multiplicités 2 (0..\* et 0..1) font apparaître le nombre d'occurrences de certains acteurs. Par exemple dans le modèle ci-dessous, on voit qu'il n'y a qu'un seul Chef de Projet et un Employeur pour chaque villes mais plusieurs base de données.



## 3-4: Le diagramme de cas d'utilisation ( DCU )

Mon **diagramme de cas d'utilisation** représente un cas entre deux utilisateurs (Employer et Chef de Projet) .Sur ce Diagramme je montre les accès et les droits de chaque utilisateurs sur le site.



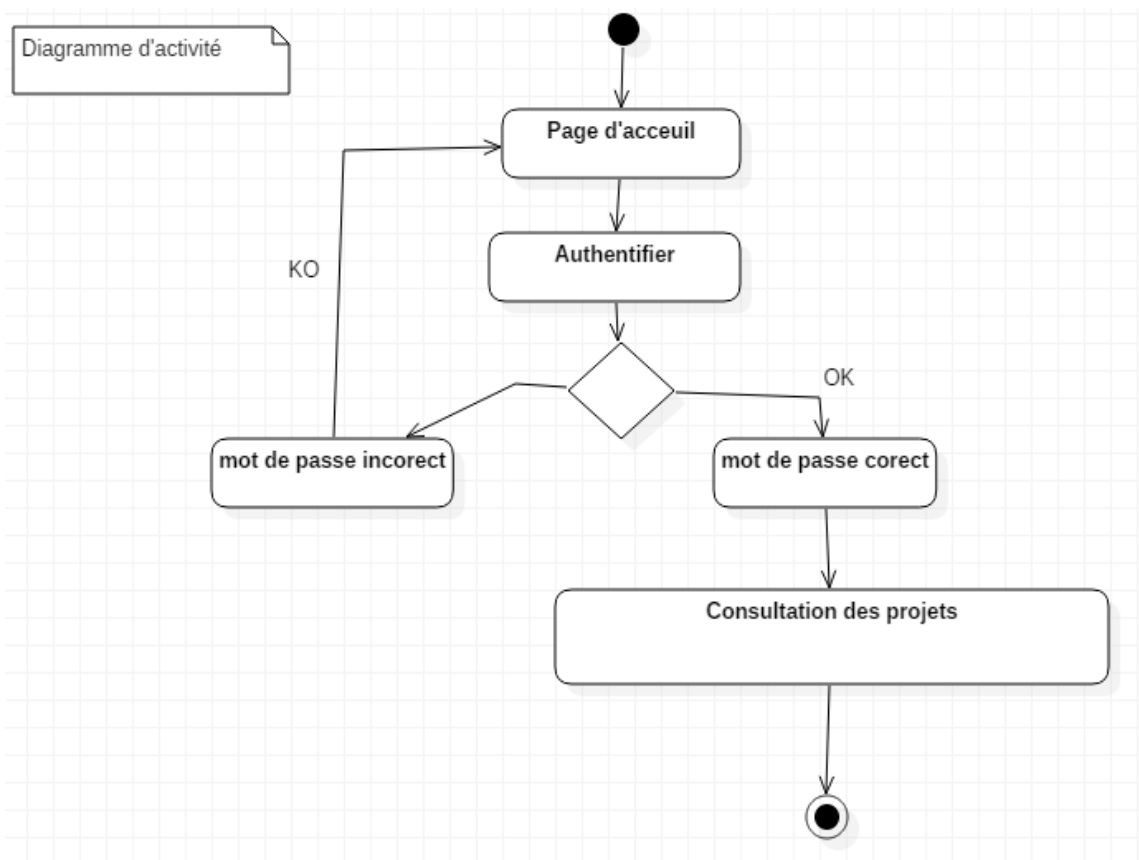
Le Chef de Projet détient les même accès droit que l'employé mais a des droit supplémentaire .

## 3 – 5 : Diagramme d'Activité

**Un Diagramme d'Activité permet** de représenter le déroulement d'un

cas d'utilisation.

Sur ce cas je représente le déroulement d'une authentification. Un User non connecté se verra rediriger sur la page de login , et User authentifier sera rediriger sur la page d'accueil.



Authentification: ( Avec FosUserBudle)

### 3-5-1: Exemple pages d'authentification

Ici vous pouvez voir 3 étapes pour accéder à la page d'accueil. Il y a la Connexion (à gauche), L'Enregistrement (à droite), et la dernière après connexion la Page d'accueil (en bas). Une fois cette étape passée, nous pouvons commencer à travailler.

S'identifier sur OrchidCreation Stats

Identifiants invalides.

Nom d'utilisateur

Mot de passe

☐ Se souvenir de moi

Connexion

Recuperer Password

Nouveau ? S'inscrire

OrchidCreation

s'inscrire

Déjà dans le groupe ? s'identifier

Recherche

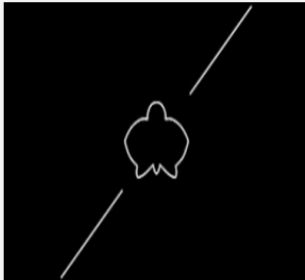
CONTACT

☐ Contact >

PAGES

☐ Project >

hello



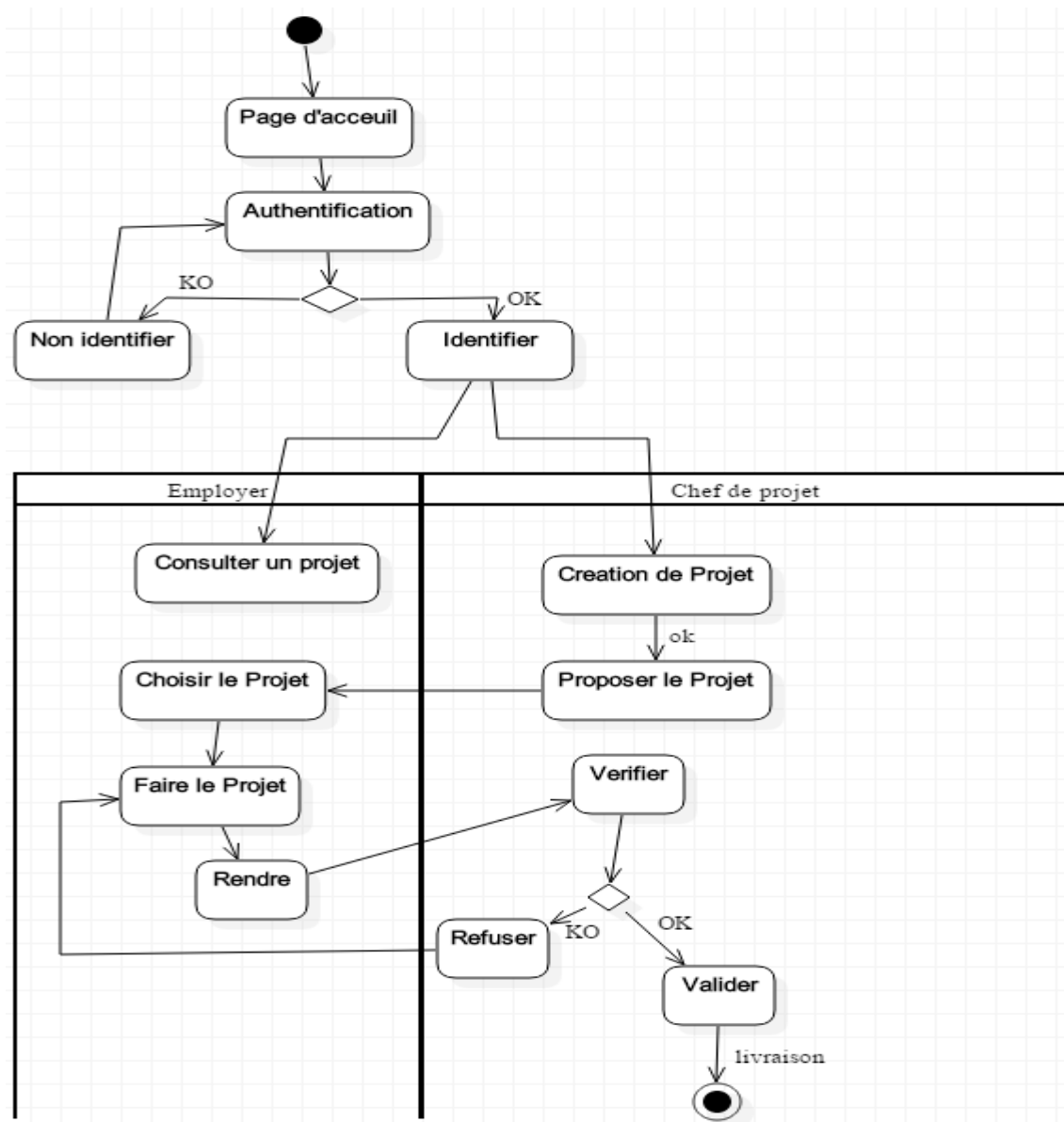
Copyright © 2017 PIXINVENT, All rights reserved.

200 @homepage 641 ms 8.0 MB mickael 6 ms 1 in 1.62 ms 32.6 X

### 3-5-2 : Diagramme d'Activité (Détailé )

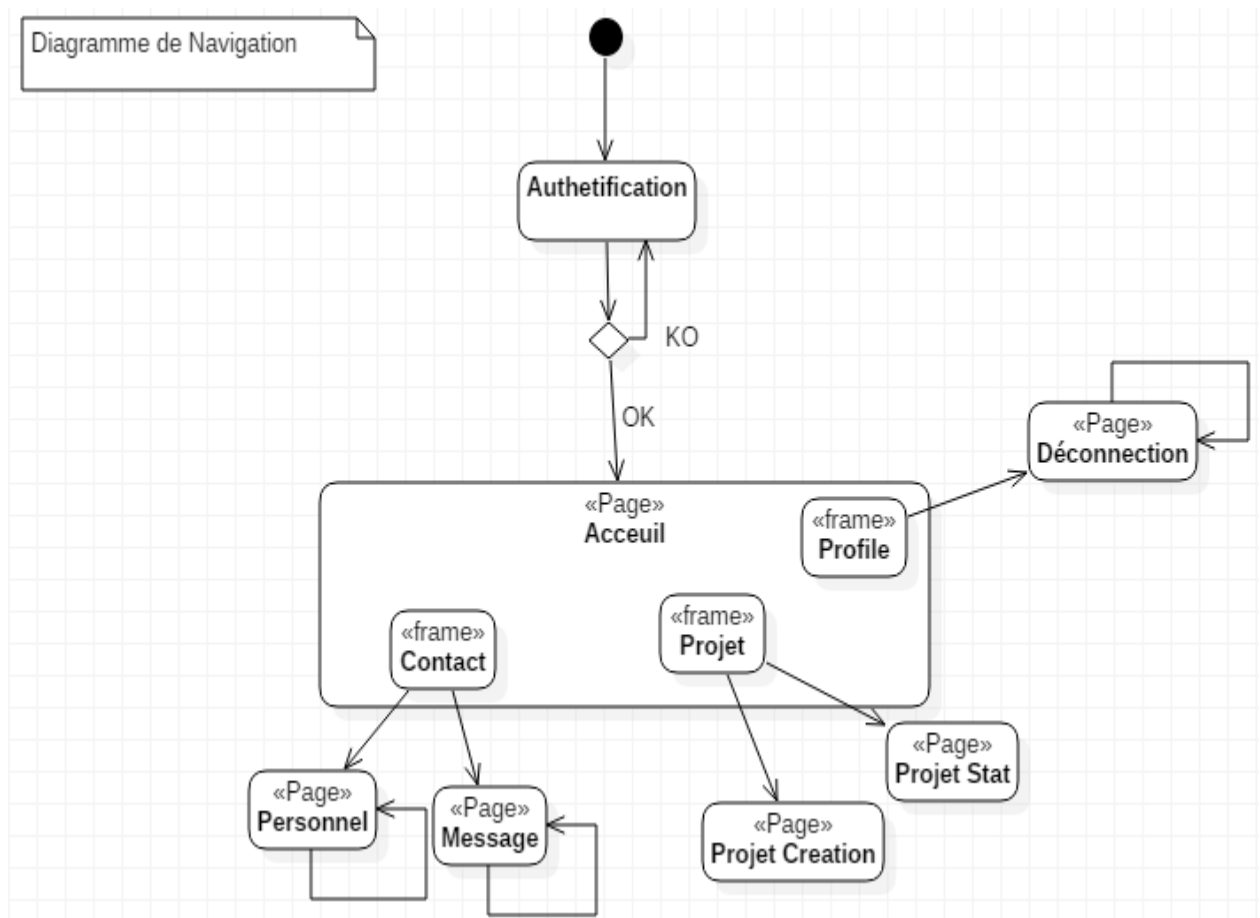
Celui la représente la l'activité entre les deux utilisateur, je montre comment ce passe le déroulement d'un projet jusqu'à sa livraison.

Le projet peut être valider mais aussi refuser donc l'employer devra le modifier pour qu'il soit conforme aux attente du Chef de Projet.



## 3 - 6 Le diagramme de navigation

Le Diagramme de Navigation sert à représenter les différentes parties ou cheminement d'une page Web ou d'une application. Ici je montre les différentes « frame » et « page » à partir de ma page principal.





# 3 – 7 OrchidCreation

## Maquettes

Les maquettes sont mes représentation de mon sites, elles vont me permettre de voir comment mes pages ou applications seront une fois terminer. Elles présentent la future interface de gestion.

Pour faire mes maquettes j'ai utiliser **Pencil Evolus**.

Mes maquettes sont en partie faite et déployer grâce au Diagramme de Navigation qui représente la cinématique de l'application.

Je fais vous présentez via ces Maquettes le travaille ou j'ai vraiment consacré la majorité de mon temps.

### Projet Principal

#### 3-7-1Maquettes Formulaire :

Name	Temps	Poste	Ville	Edit
<input type="text" value="Mickael"/>	<input type="text" value="5h"/>	<div>Art Director Manager Director Account Manager</div>	<input type="text" value="New York"/>	<div>Modifier Supprimer</div>
<input type="button" value="Valider"/>				

Name	Temps	Poste	Ville	Edit
<input type="text" value="Mickael"/>	<div>6h 7h 8h</div>	<input type="text" value="Art Director"/>	<input type="text" value="New York"/>	<div>Modifier Supprimer</div>
<input type="button" value="Valider"/>				

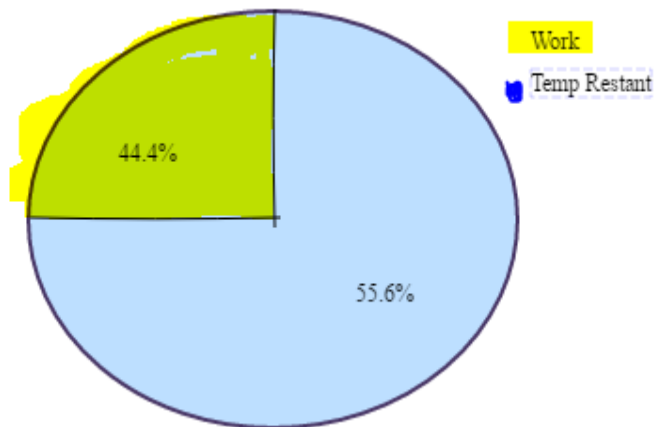
#### 3-7-2Tableau du personnel après validation :

Nom	Email	Poste	Ville
Mickael	mickael@gmail.fr	group creative director	New york
yves	yves@hotmail.fr	Account Manager	Paris
alex	alex@gmail.fr	Manager Director	London
philipe	fil@gmail.com	Assoiate Creative Director	London
antoine	anto@hormail.com	group creative director	Honk Kong
romain	roro@hotmail.fr	art director	Rio

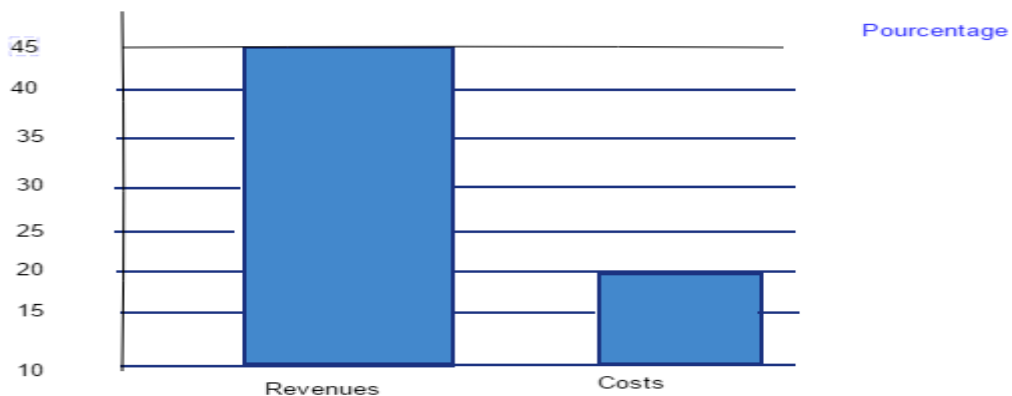
### 3-7-3Maquettes Chart:

Il représente le Temps passer de chaque employer sur les projets

#### YOU STILL IN WORK

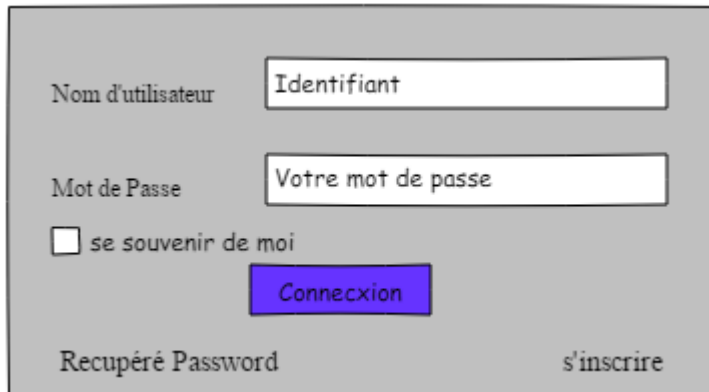


Il représente les revenus et les Frais



## Projet secondaire

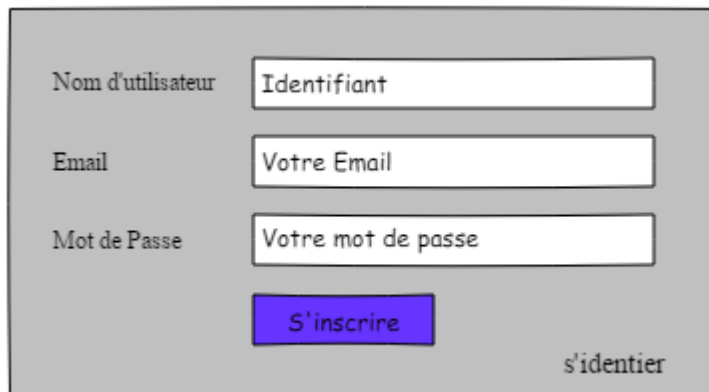
### 3-7-4Connexion:



A login form with a light gray background. It contains the following elements:

- A label "Nom d'utilisateur" followed by a text input field containing the placeholder text "Identifiant".
- A label "Mot de Passe" followed by a text input field containing the placeholder text "Votre mot de passe".
- A checkbox with the label "se souvenir de moi".
- A blue button with the text "Connexion".
- A link "Recupéré Password" on the bottom left.
- A link "s'inscrire" on the bottom right.

### 3-7-5-Longin :



A registration form with a light gray background. It contains the following elements:

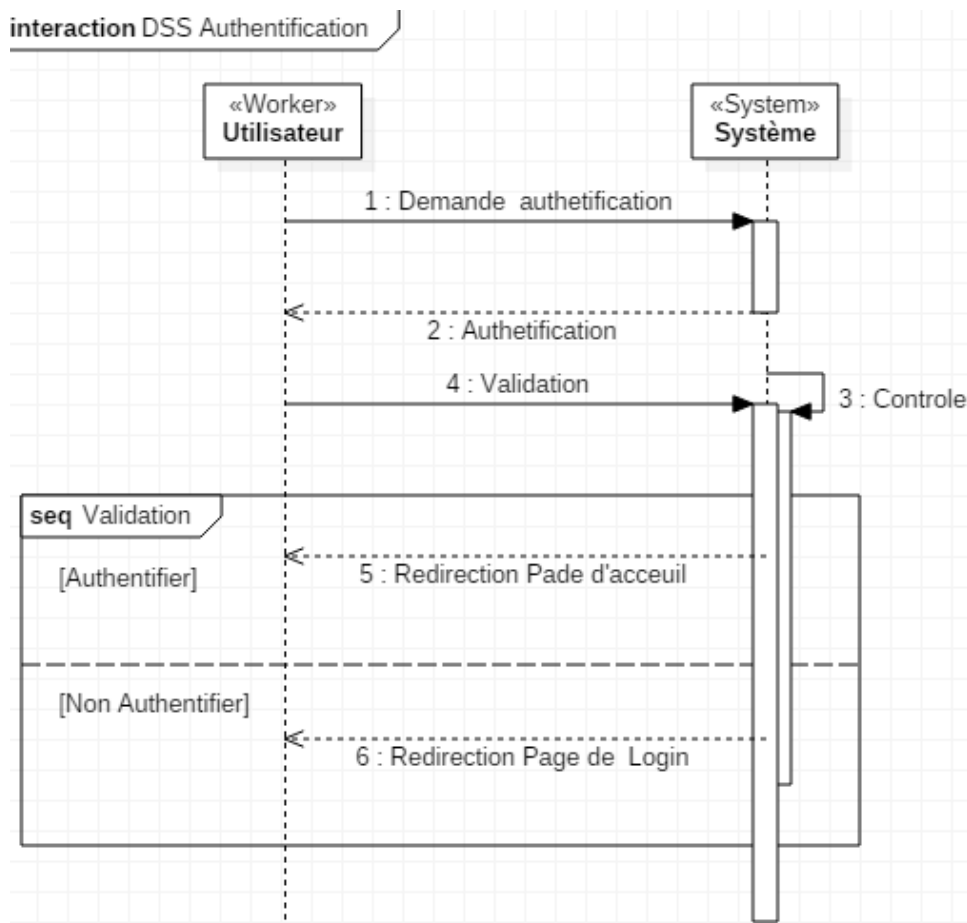
- A label "Nom d'utilisateur" followed by a text input field containing the placeholder text "Identifiant".
- A label "Email" followed by a text input field containing the placeholder text "Votre Email".
- A label "Mot de Passe" followed by a text input field containing the placeholder text "Votre mot de passe".
- A blue button with the text "S'inscrire".
- A link "s'identifier" on the bottom right.

# 3-8 Diagramme de Séquence Système (DSS)

Le **Diagramme de Séquence Système (DSS)** me permet de montrer les interactions Entre les Utilisateurs (Employer et Chef de Projet) et le Système. Les échanges entre les deux sont symbolisés sous forme de messages (représentés par une flèche) indiquant le sens du message. Les réponses du système sont eux représenté par des pointillés.

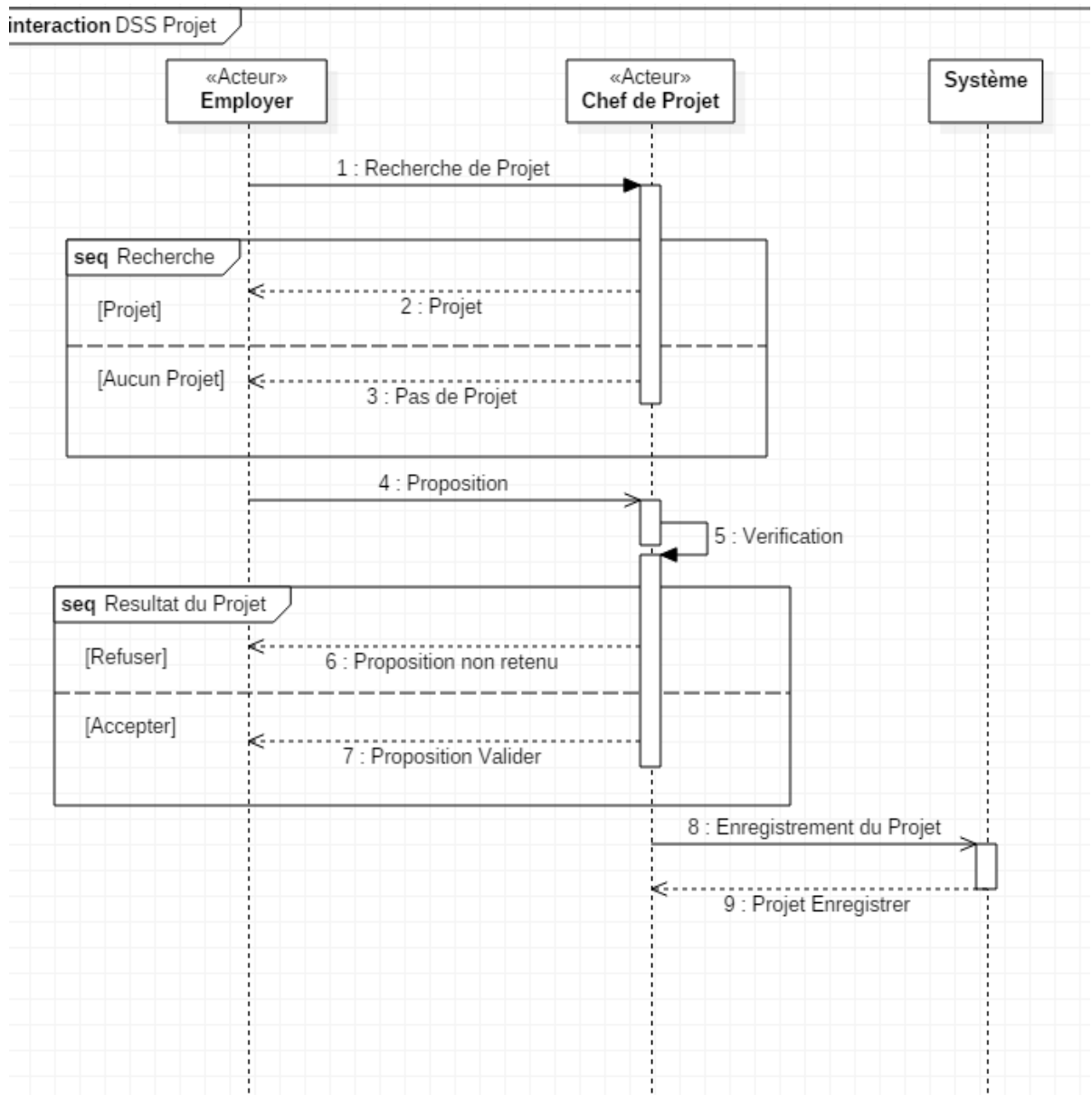
(Les Users sont représenté sur ces deux graphique par stéréotype « worker » et « Acteur » sur starUML ).

## 3-8-1 DSS : Authentification



### 3-8-2 DSS : Projet

La mon DSS représente entre 3 acteurs ( Employer, Chef de Projet et le Système) lors d'une création d'un Projet et de la finalisation.

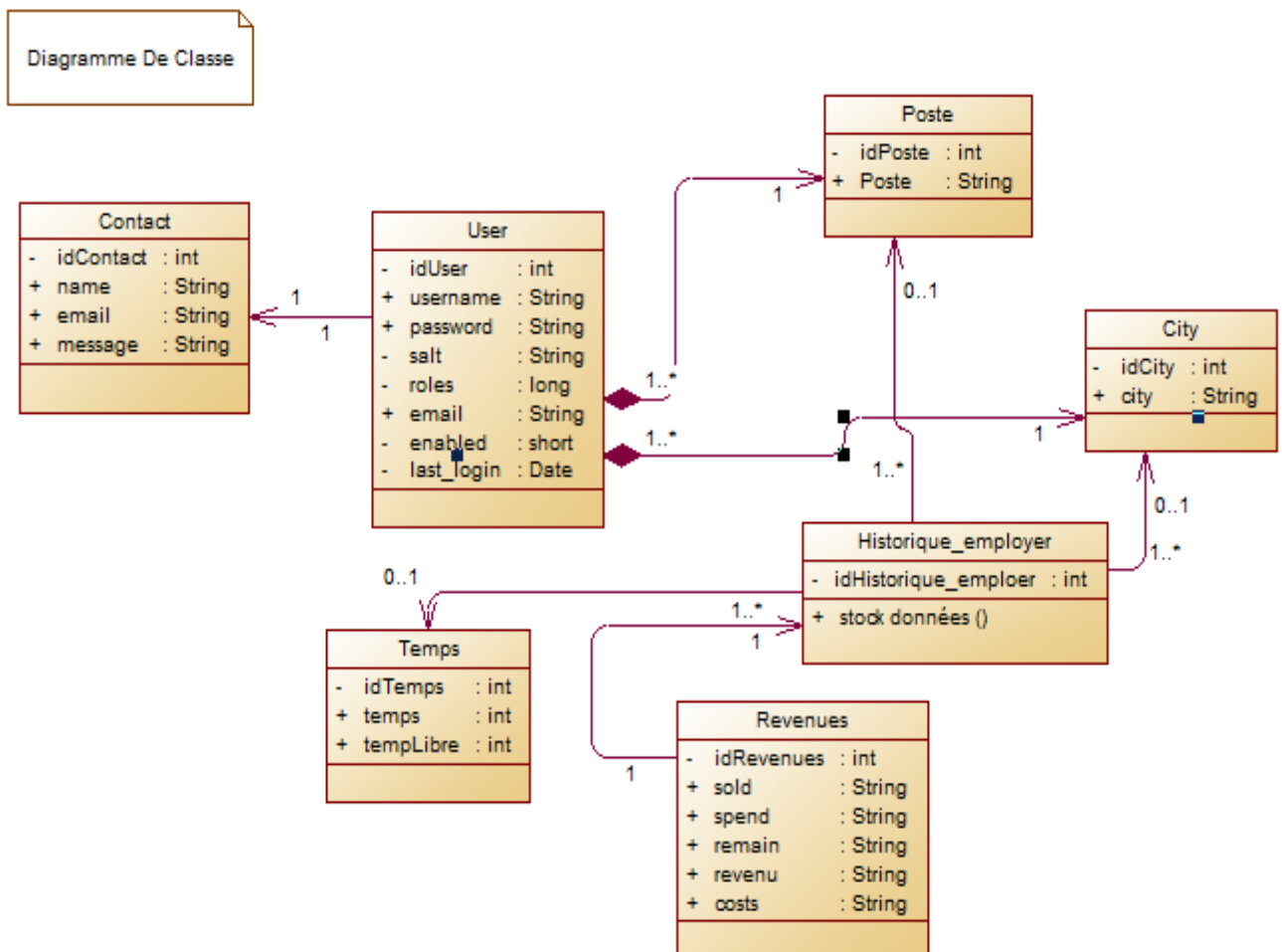


# **Chapitre 4 - Conception de la Base de Données**

# 4-1:Diagramme de Classe

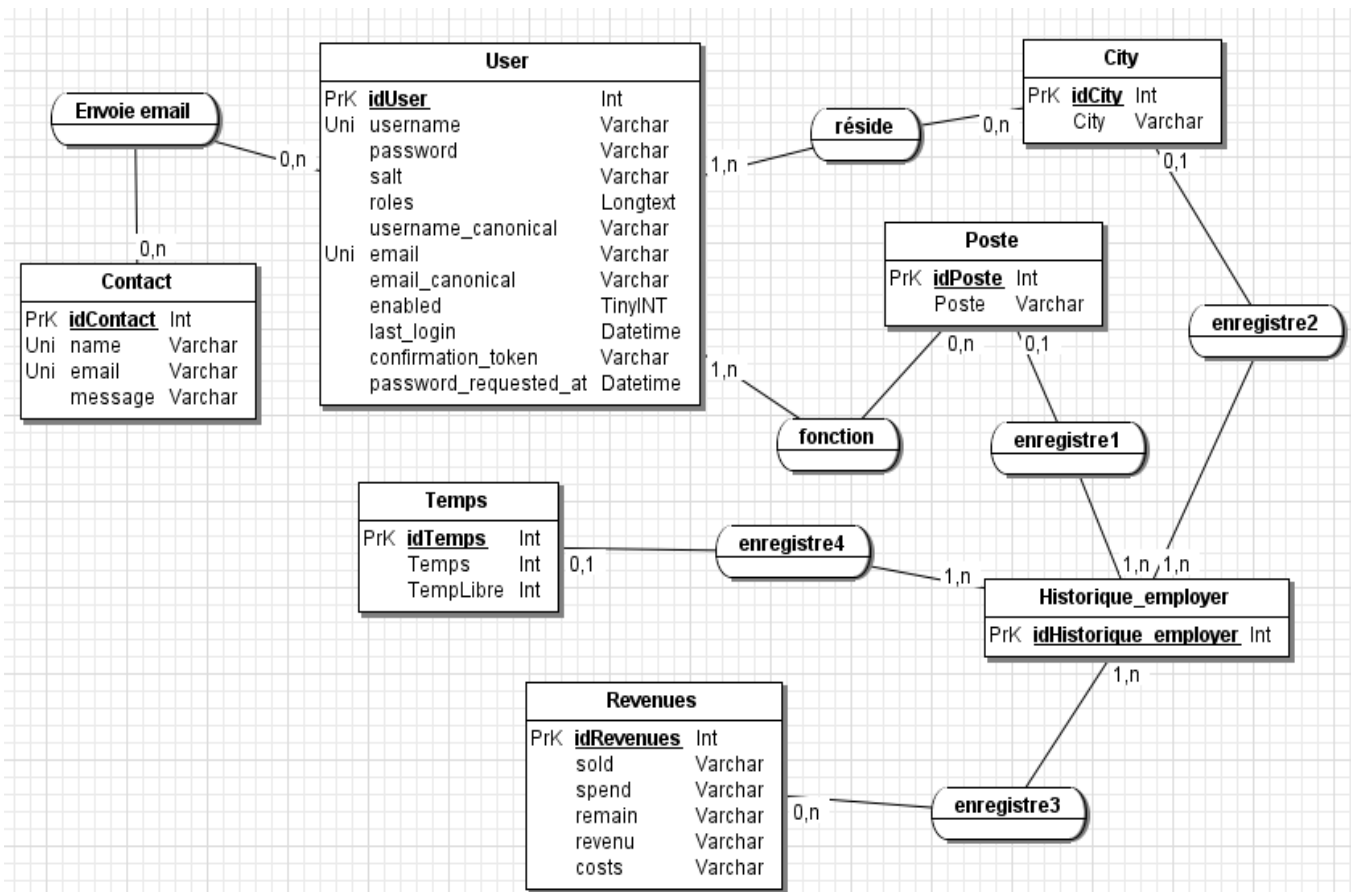
J'ai fais mon Diagramme de Classe via **PowerAMC**.

J'ai créer deux types d'associations, la première est une association de Composition sur les table (User – Poste et User-City ), elle explique que City/Poste sont fortement liés représenté par des losanges plein. La deuxième juste une association simple représenté par une petite flèche.



## 4 – 2 Model conceptuel de données MCD (MERISE)

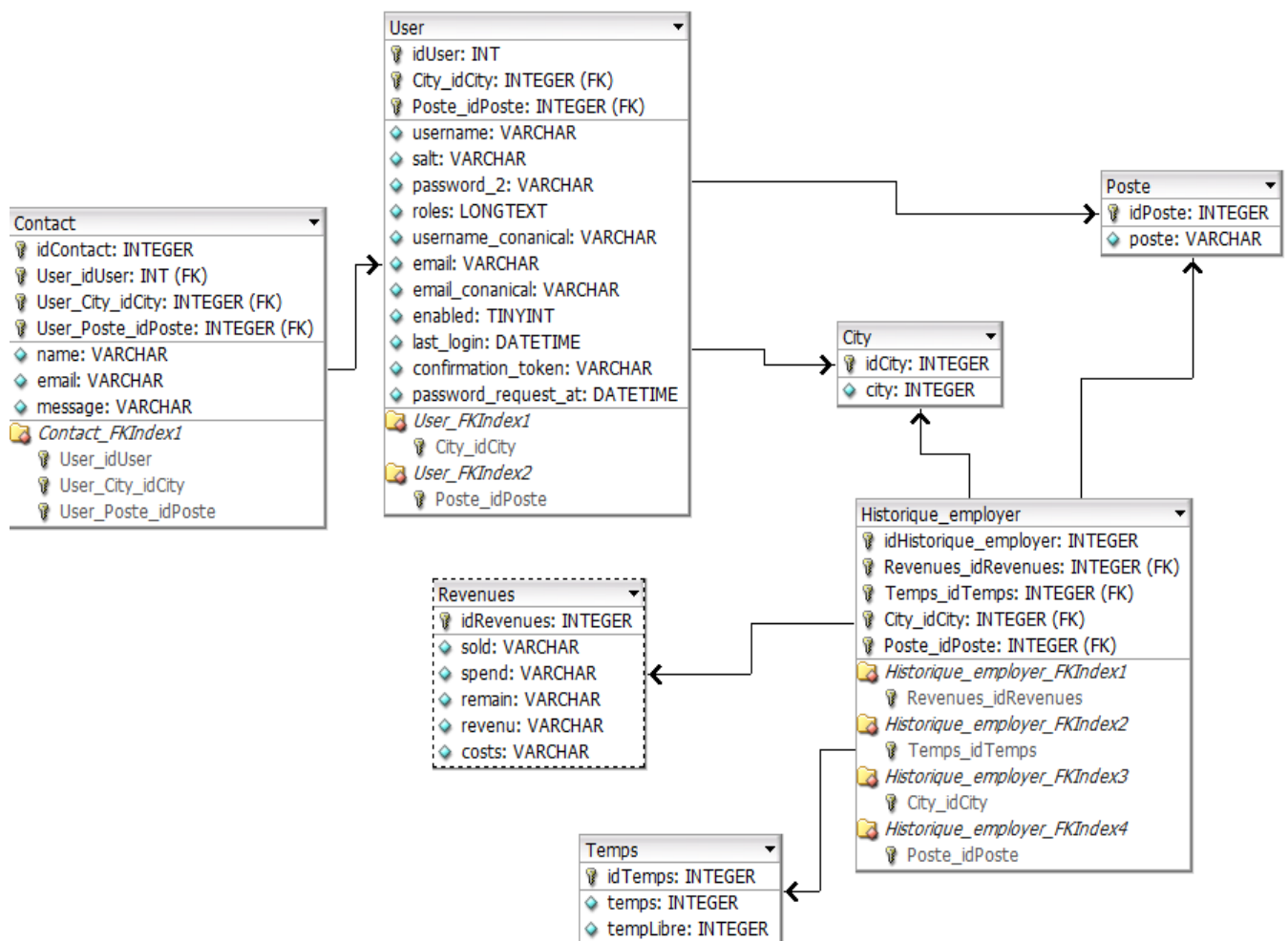
Le modèle conceptuel des données (**MCD**) d'écriture la façon les données qui seront utilisées par le système d'information. Il s'agit donc d'une représentation des données, facilement compréhensible, permettant de décrire le système d'information à l'aide d'entités. Il montre la sémantique c'est à dire le sens attaché à ces données et à leurs rapports et non à l'utilisation qui peut en être faite





# 4 - 3Modèle logique de données (MLD)

Le Modèle Logique de Données (MLD) représente la structure de ma BDD. Il tien compte du niveau organisationnel de mes données (Créer avec dg Design)



## 4 - 4 : Conception de la Base de Données

Travaillant sous le Framework PHP « Symfony », je disposais de l'ORM Doctrine2, qui c'est chargé via la console « GitBash » de généré mes

Getters /Setters et Entités (« Entité » c'est mes Tables sous Symfony ) .Je me suis aussi aidé de **WorkBench** pour mieux structuré mes Entités par

La suite mais surtout pour visualisé mes « Relations » entre différents Entités (Exemple sur la prochaine page ).

Exemple pour créer mes Entité:

```
dgmmick@LAPTOP-V4JP0I8N MINGW64 /c/xampp/htdocs/Orchid/OrchidContact (master)
$ bin/console doctrine:generate:entity
```

Welcome to the Doctrine2 entity generator

This command helps you generate Doctrine2 entities.

First, you need to give the entity name you want to generate.  
You must use the shortcut notation like **AcmeBlogBundle:Post**.

The Entity shortcut name: AppBundle:City|

-Généré les Getters/Setters toutes mes Entités:

```
dgmmick@LAPTOP-V4JP0I8N MINGW64 /c/xampp/htdocs/Orchid/OrchidContact (master)
$ bin/console doctrine:generate:entities AppBundle|
```

-Mettre a jour la Base De Données

```
dgmmick@LAPTOP-V4JP0I8N MINGW64 /c/xampp/htdocs/Orchid/OrchidContact (master)
$ bin/console doctrine:schema:update --force|
```

Résultat :

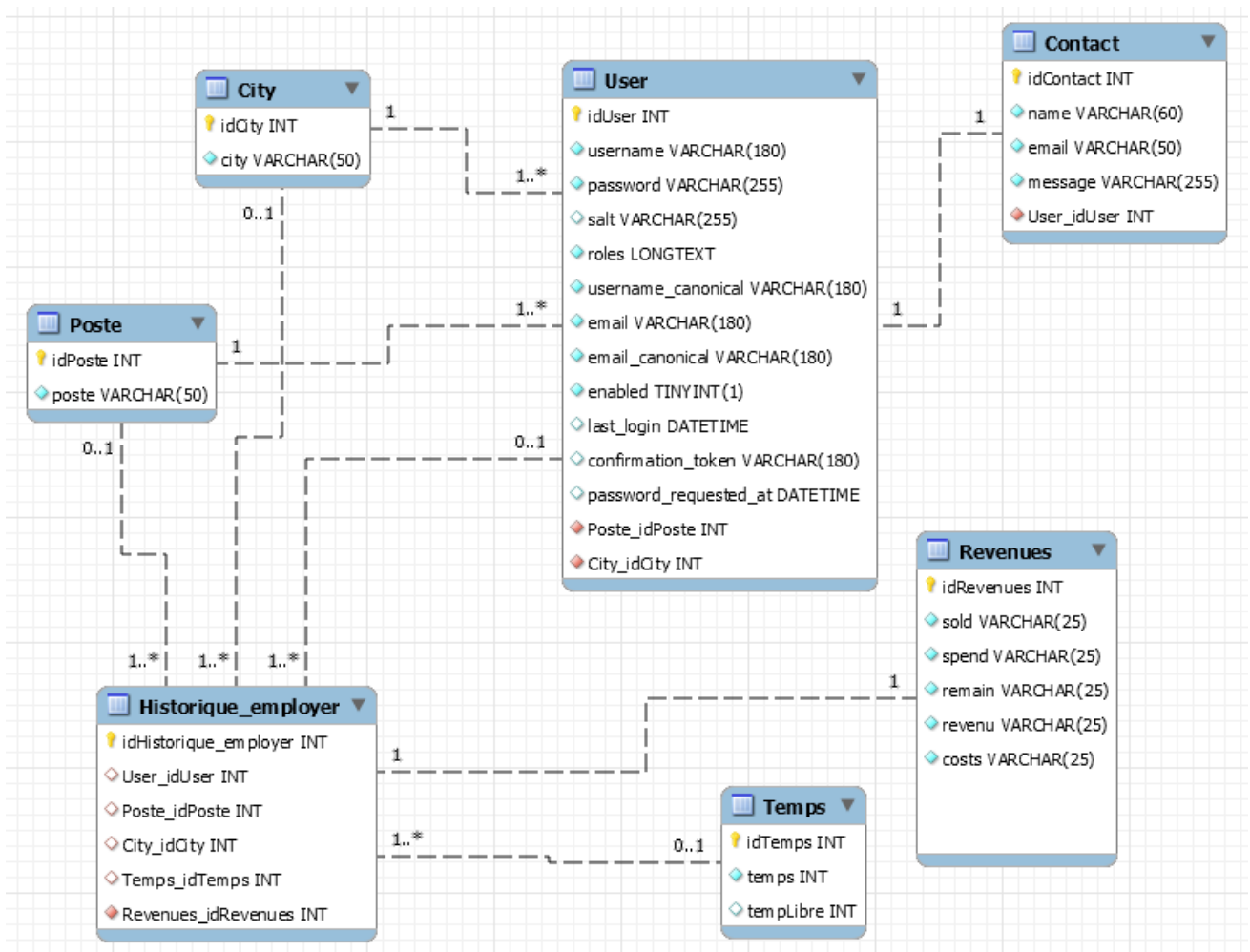
Table	Action	Lignes	Type	Interclassement	Taille	Perte
city	★ Afficher Structure Rechercher Insérer Vider Supprimer	5	InnoDB	utf8_unicode_ci	16 Kio	-
contact	★ Afficher Structure Rechercher Insérer Vider Supprimer	10	InnoDB	utf8_unicode_ci	16 Kio	-
historique_employer	★ Afficher Structure Rechercher Insérer Vider Supprimer	5	InnoDB	utf8_unicode_ci	80 Kio	-
poste	★ Afficher Structure Rechercher Insérer Vider Supprimer	5	InnoDB	utf8_unicode_ci	16 Kio	-
revenues	★ Afficher Structure Rechercher Insérer Vider Supprimer	3	InnoDB	utf8_unicode_ci	16 Kio	-
temps	★ Afficher Structure Rechercher Insérer Vider Supprimer	10	InnoDB	utf8_unicode_ci	16 Kio	-
user	★ Afficher Structure Rechercher Insérer Vider Supprimer	7	InnoDB	utf8_unicode_ci	96 Kio	-
7 table(s)	Somme	45	InnoDB	latin1_swedish_ci	256 Kio	0 0

## 4 - 5 : Modèle Physique des Données

Pour élaborer mon Modèle Physique des données (MPD), j'ai décidé d'utiliser **WorkBench**, ce logiciel m'a permis de créer un schéma réel de mes

Tables, d'ajouter mes « FOREIGN KEY », et de générer un code SQL. Vous pouvez constater que j'utilise 7 Tables avec des relations (OneToOne) et (OneToMany).

Voila le résultat :



Pour faire les relation entre Table, j'ai pensé de cette façon :

- 1 User peut avoir 1 City mais City peut avoir plusieurs User.
- 1 User peut avoir 1 Poste mais un Poste peut avoir plusieurs User .
- La Table Contact de User.

## 4-6 : Explication de mes Tables

**Table User** : Table géré avec FosUserBundle utilisé grâce a Symfony, il m'a permis d'ajouté les Utilisateurs.

COLONNE	TYPE	DESCRIPTION
idUser	int	Clé primaire de la table « User »
Username	Varchar(180)	Nom de l'utilisateur
Passesword	Varchar(255)	Mot de passe de l'utilisateur
City_idCity	int	Clé étrangère table « City »
Poste_idPoste	int	Clé étrangère table « Poste »
Email	Varchar(80)	Email de l'utilisateur

**Table Poste** : Table qui stock les différents Poste.

COLONNE	TYPE	DESCRIPTION
idPoste	int	Clé primaire de la table « Poste »
Poste	Varchar(50)	Poste de chaque utilisateurs

**Table Temps** : Table qui stock les différents Poste.

COLONNE	TYPE	DESCRIPTION
idPoste	int	Clé primaire de la table « Poste »
Temps	int *	Temps passé sur les projets
TempLibre	int *	Reste de temps

int\*: j'ai décidé en cours de mon projet de mettre les type en (int) avant c'était du (VARCHAR), a cause d'un souci dans mon projet, je vais expliqué la raison dans la partie concerner.

**Table Revenues:** Table qui stock les différents Revenues.

COLONNE	TYPE	DESCRIPTION
idRevenues	int	Clé primaire de la table « Revenues »
Sold	Varchar(25)	Projet vendu
Spend	Varchar(25)	Dépense effectuer
Remain	Varchar(25)	Reste
Revenu	Varchar(25)	Les revenus de l'entreprise
Costs	Varchar(25)	Les frais divers

**Table Contact:** Table qui stock les différents Contact.

COLONNE	TYPE	DESCRIPTION
idContact	int	Clé primaire de la table « Contact »
User_idUser	int	Clé étrangère table « User »
Name	Varchar(60)	Nom de l'utilisateur
Email	Varchar(50)	Email de l'utilisateur
message	Varchar(255)	Message a envoyer

**Table City :** Table qui stock les différentes Villes.

COLONNE	TYPE	DESCRIPTION
idCity	int	Clé primaire de la table « City »
City	Varchar(50)	Ville des différentes entreprise

**Table Historique\_employer:** Table qui stock les données du tableau des utilisateurs.

COLONNE	TYPE	DESCRIPTION
idHistorique_employer	int	Clé primaire de la table « Historique_employer »
City_idCity	int	Clé étrangère table « City »
Poste_idPoste	int	Clé étrangère table « Poste »
User_idUser	int	Clé étrangère table « User »
Temps_idTemps	int	Clé étrangère table « Temps »
Revenues_idRevenues	int	Clé étrangère table « Revenues »

#### 4 - 7: Création de Code SQL de ma Base De Données

Voici le Script SQL de ma Base De Données (BDD) qui me permettra de commencer mon travail.

J'utilisais PHPMyAdmin comme serveur local sous XAMPP.

```
-- Table `Poste`
```

```
CREATE TABLE IF NOT EXISTS `Poste` (
  `idPoste` INT NOT NULL AUTO_INCREMENT,
  `poste` VARCHAR(50) NOT NULL,
  PRIMARY KEY (`idPoste`))
ENGINE = InnoDB;
```

-----  
-- Table `City`  
-----

```
CREATE TABLE IF NOT EXISTS `City` (  
  `idCity` INT NOT NULL AUTO_INCREMENT,  
  `city` VARCHAR(50) NOT NULL,  
  PRIMARY KEY (`idCity`))  
ENGINE = InnoDB;
```

-----  
-- Table `User`  
-----

```
CREATE TABLE IF NOT EXISTS `User` (  
  `idUser` INT NOT NULL AUTO_INCREMENT,  
  `username` VARCHAR(180) NOT NULL,  
  `password` VARCHAR(255) NOT NULL,  
  `salt` VARCHAR(255) NULL,  
  `roles` LONGTEXT NOT NULL,  
  `username_canonical` VARCHAR(180) NOT NULL,  
  `email` VARCHAR(180) NOT NULL,  
  `email_canonical` VARCHAR(180) NOT NULL,  
  `enabled` TINYINT(1) NOT NULL,  
  `last_login` DATETIME NULL,  
  `confirmation_token` VARCHAR(180) NULL,  
  `password_requested_at` DATETIME NULL,  
  `Poste_idPoste` INT NOT NULL,  
  `City_idCity` INT NOT NULL,  
  PRIMARY KEY (`idUser`),  
  INDEX `fk_User_Poste1_idx` (`Poste_idPoste` ASC),  
  INDEX `fk_User_City1_idx` (`City_idCity` ASC),  
  UNIQUE INDEX `username_UNIQUE` (`username` ASC),  
  UNIQUE INDEX `email_UNIQUE` (`email` ASC),  
  CONSTRAINT `fk_User_Poste1`  
    FOREIGN KEY (`Poste_idPoste`)  
    REFERENCES `Poste` (`idPoste`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION,  
  CONSTRAINT `fk_User_City1`  
    FOREIGN KEY (`City_idCity`)  
    REFERENCES `City` (`idCity`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION)  
ENGINE = InnoDB;
```

-----  
-- Table `Contact`  
-----

```
CREATE TABLE IF NOT EXISTS `Contact` (  
  `idContact` INT NOT NULL AUTO_INCREMENT,  
  `name` VARCHAR(60) NOT NULL,  
  `email` VARCHAR(50) NOT NULL,  
  `message` VARCHAR(255) NOT NULL,  
  `User_idUser` INT NOT NULL,  
  PRIMARY KEY (`idContact`),  
  INDEX `fk_Contact_User1_idx` (`User_idUser` ASC),  
  UNIQUE INDEX `name_UNIQUE` (`name` ASC),  
  UNIQUE INDEX `email_UNIQUE` (`email` ASC),  
  CONSTRAINT `fk_Contact_User1`  
    FOREIGN KEY (`User_idUser`)  
      REFERENCES `User` (`idUser`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION)  
ENGINE = InnoDB;
```

-----  
-- Table `Revenues`  
-----

```
CREATE TABLE IF NOT EXISTS `Revenues` (  
  `idRevenues` INT NOT NULL AUTO_INCREMENT,  
  `sold` VARCHAR(25) NOT NULL,  
  `spend` VARCHAR(25) NOT NULL,  
  `remain` VARCHAR(25) NOT NULL,  
  `revenu` VARCHAR(25) NOT NULL,  
  `costs` VARCHAR(25) NOT NULL,  
  PRIMARY KEY (`idRevenues`))  
ENGINE = InnoDB;
```

-----  
-- Table `Temps`  
-----

```
CREATE TABLE IF NOT EXISTS `Temps` (  
  `idTemps` INT NOT NULL AUTO_INCREMENT,  
  `temps` INT NOT NULL,  
  `tempLibre` INT NULL,  
  PRIMARY KEY (`idTemps`))  
ENGINE = InnoDB;
```

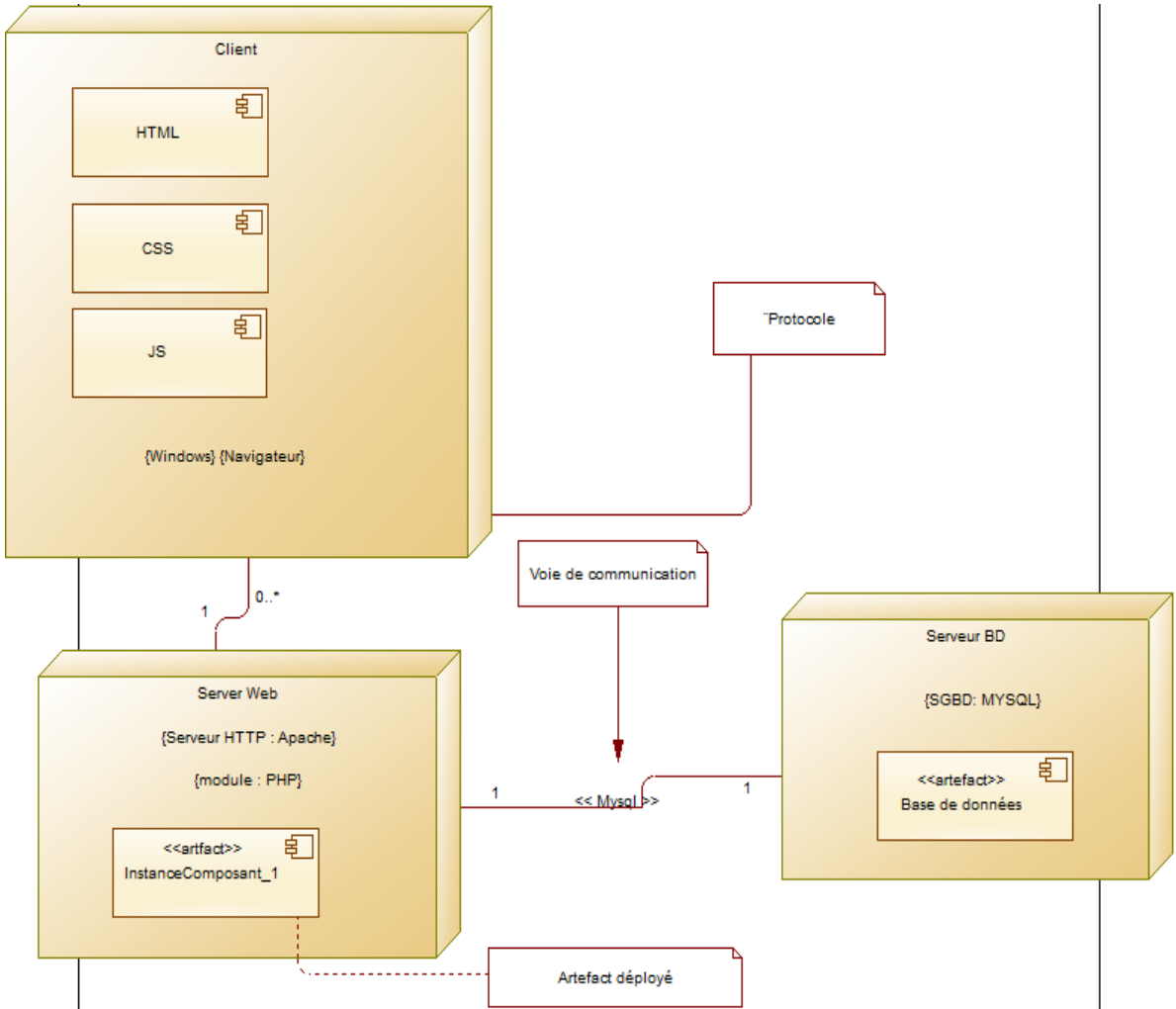


-----  
-- Table `Historique\_employer`  
-----

```
CREATE TABLE IF NOT EXISTS `Historique_employer` (  
  `idHistorique_employer` INT NOT NULL AUTO_INCREMENT,  
  `User_idUser` INT NULL,  
  `Poste_idPoste` INT NULL,  
  `City_idCity` INT NULL,  
  `Temps_idTemps` INT NULL,  
  `Revenues_idRevenues` INT NOT NULL,  
  PRIMARY KEY (`idHistorique_employer`),  
  INDEX `fk_Historique_employer_User1_idx` (`User_idUser` ASC),  
  INDEX `fk_Historique_employer_Poste1_idx` (`Poste_idPoste` ASC),  
  INDEX `fk_Historique_employer_City1_idx` (`City_idCity` ASC),  
  INDEX `fk_Historique_employer_Temps1_idx` (`Temps_idTemps` ASC),  
  INDEX `fk_Historique_employer_Revenues1_idx` (`Revenues_idRevenues` ASC),  
  CONSTRAINT `fk_Historique_employer_User1`  
    FOREIGN KEY (`User_idUser`)  
    REFERENCES `User` (`idUser`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION,  
  CONSTRAINT `fk_Historique_employer_Poste1`  
    FOREIGN KEY (`Poste_idPoste`)  
    REFERENCES `Poste` (`idPoste`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION,  
  CONSTRAINT `fk_Historique_employer_City1`  
    FOREIGN KEY (`City_idCity`)  
    REFERENCES `City` (`idCity`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION,  
  CONSTRAINT `fk_Historique_employer_Temps1`  
    FOREIGN KEY (`Temps_idTemps`)  
    REFERENCES `Temps` (`idTemps`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION,  
  CONSTRAINT `fk_Historique_employer_Revenues1`  
    FOREIGN KEY (`Revenues_idRevenues`)  
    REFERENCES `Revenues` (`idRevenues`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION)  
ENGINE = InnoDB;
```

#### 4 - 8 Diagramme de déploiement :

Le diagramme de déploiement de mon application web qui est basée sur HTML/CSS/JAVASCRIPT sur le coté « Client », PHP coté « Serveur Web » et SGBD:Mysql sue le coté « BD ». C' est une vue statique qui sert à représenter l'utilisation de l'infrastructure physique par le système et la manière dont les composants du système sont répartis ainsi que leurs relations entre eux

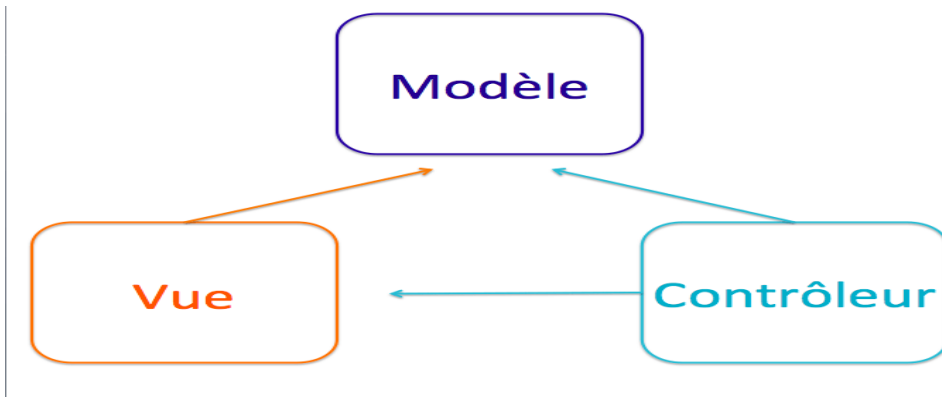


# **CHAPITRE 5 - CONCEPTION DE L'APPLICATION**

## 5 – 1 L'architecture MVC

---

L'architecture MVC qui veut dire ( Modèle-Vue-Contrôleur ) permet simplement de séparer les données, l'affichage et les traitements ou actions. En utilisant Symfony j'ai respecté cette règle. (Page suivante )



**Modèle:** Il contient les données, c'est le cœur de l'application. Il traite les données et interagit avec la BDD en ajoutant, supprimant et modifiant les valeurs.

**Vue:** c'est la plateforme d'interaction de l'utilisateur. Elle affiche les résultats renvoyés par le modèle et informations données par l'utilisateur. Elle affiche juste et ne génère pas de traitement.

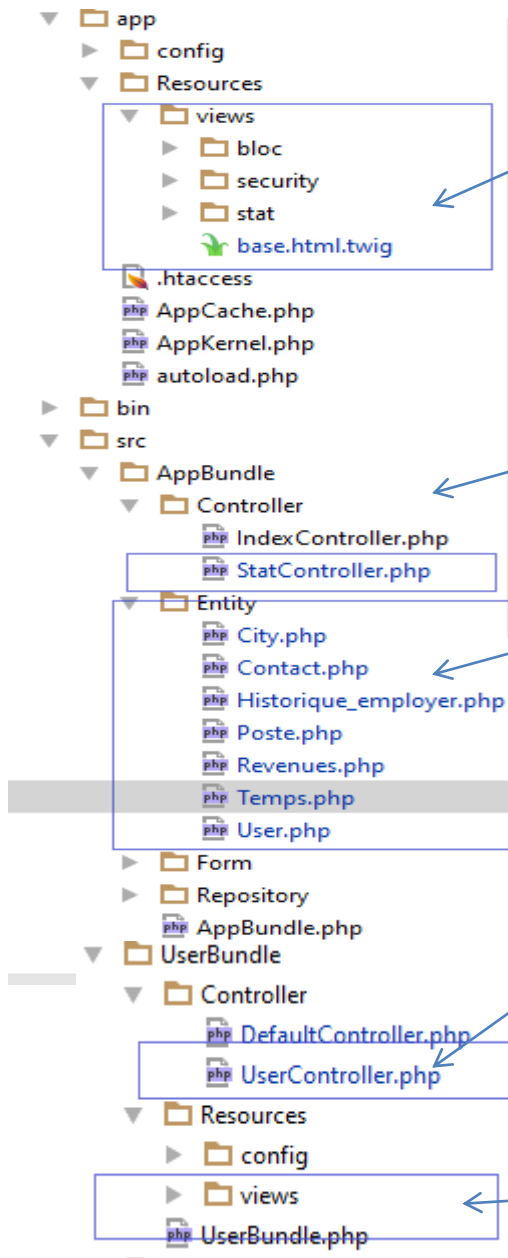
**Contrôleur:** Il gère la synchronisation, demande la modification des données au Modèle.

**Les avantages majeurs du MVC sont:**

- La propreté d'un code
- Rapidité pour le développement
- Facilité de maintenance

## 5 – 2 Arborescence de mon projet Symfony

Voila comment mon Projet Symfony sur PhpStrom était structuré.



Je placais mes Vue twig  
Directement dans un dossier view  
dans « app », je travaille ainsi depuis  
Symfony 3

Controller principal  
dans mon AppBundle

Toutes mes Entités

Mon controller pour mes  
utilisateurs dans mon UserBundle,  
Bundle comprenant mon  
FosUserBundle.

Mes vues de User et vues de  
FosUser

## 5 - 3 Conception de Projet Sheet

Mon Projet principal consistait de retranscrire mon fichier Excel ( voir au début rapport) en Application web. Je vais vous présentez une partie du projet avec son fonctionnement, il y en aura 3, le premier qui est le « Formulaire» qui va assigné chaque employés un temps passé sur un projet et la ville ou il a travaillé. Il y aura aussi les « Chart» le premier qui est un « PieChart » représente le temps global de tout les employés, le Deuxième un « ColumnChart » représente les revenus et le Costs ( ce chart a des valeurs ajouté en BDD faute de temps pour créer la partie approprier ).

J'ai réussi a créer ces Chart grâce au « GoogleChartsBundle » découvert après de long jours de recherche.

Vous trouverez les illustrations de l'application puis les Codes détaillé.

### 4-2 : Formulaire.

Personnel — ⌵

Name	Temps	Poste	Ville	Edit
<input type="text"/>	<input type="text"/>	<div><input type="text"/><ul style="list-style-type: none"><li>Account Manager</li><li>Art Director</li><li>Associate Creative Director</li><li>Group Creative Director</li><li>Manager Director</li></ul></div>	<input type="text"/>	

Envoyer

Ma class Historique\_employerType venant de mon dossier « Form »  
Pour pouvoir créer mon formulaire imbrique

```
class Historique_employerType extends AbstractType
{
    /**
     * {@inheritdoc}
     */
    public function buildForm(FormBuilderInterface $builder, array $options)
    {
        $builder
            ->add('poste')
            ->add('user')
            ->add('city')
            ->add('temps')
            ->add('envoyer', SubmitType::class);
    }
}
```

- >add('poste') venant de l'entité Poste
- >add('user') venant de l'entité User de FosUserBundle
- >add('city') venant de l'entité City
- >add('temps') venant de l'entité Temps
- >add('envoyer') Bouton Submit

```
class CityType extends AbstractType
{
    /**
     * {@inheritdoc}
     */
    public function buildForm(FormBuilderInterface $builder, array $options)
    {
        $builder->add('city', EntityType::class, array(
            'class' => 'AppBundle\Entity\City',
            'choice_label' => 'city',
            'expanded' => false,
            'multiple' => false,
            'label' => ' '
        ));
    }
}
```

Extrait de ma class CityType pour créer ma liste déroulante venant  
Directement de ma Table City.

C'est un formulaire Imbriqué a partir de plusieurs Entité ( Poste, Employé, City et Temps ), les données seront ensuite enregistré dans l'entité « Historique\_employer ». Ici 2 exemples de table en BDD (City et Poste) et de la table Historique.

id	City
1	New York
2	London
3	Paris
4	Sao Paulo
5	Hong Kong

id	Poste
1	Account Manager
2	Art Director
3	Associate Creative Director
4	Group Creative Director
5	Manager Director

	id	city_id	poste_id	user_id	temps_id
Effacer	61	1	1	1	2
Effacer	62	2	2	2	3
Effacer	63	3	3	3	4
Effacer	64	4	4	4	5
Effacer	72	3	5	4	10

Pour la création de ces animations , j'ai créer un StatController , vu que toutes mes animations sont sur la meme page j'ai decidé de toutes les coder dans mon« indexAction » vu qu'ils retournent la meme vu dans mon Render: (**stat/projectSheet.html.twig** )

```
return $this->render('stat/projectSheet.html.twig', array(
    'spendList' => $spendList,
    'cityList' => $cityList,
    'postList' => $postList,
    'historique' => $formHistorique->createView(),
    'piechart' => $pieChart,
    'col' => $col,
));
```



Maintenant le code qui ma permis de créer ce formulaire :

```
/**
 * @Route("/stat", name="stat")
 * @param Request $request
 * @return \Symfony\Component\HttpFoundation\Response
 */
public function indexAction(Request $request){

    /*****Creation du formulaire *****/

    //creation de l'entity Historique_employer
    $historique = new Historique_employer();

    $formHistorique = $this->createForm(Historique_employerType::class, $historique);

    // je verifie si ma requete est en poste et que les valeurs sont correct
    $formHistorique->handleRequest($request);

    if ($request->isMethod('POST') && $formHistorique->isSubmitted() && $formHistorique->isValid()) {
        $historiqueData = $formHistorique->getData();

        $em = $this->getDoctrine()->getManager();
        // je persiste
        $em->persist($historiqueData);
        // je declare l'enregistrement
        $em->flush();
    }
}
```

Code de ma vue Twig:

```

<form action="{{ path('/stat') }}" method="post">
    {{ form_start(historique) }}
    <table class="table table-bordered table-striped table-condensed">
        <thead>
            <tr>
                <th>Name</th>
                <th>Temps</th>
                <th>Poste</th>
                <th>Ville</th>
                <th>Edit</th>
            </tr>
            <tr>
                <th>{{ form_widget(historique.user) }}</th>
                <th>{{ form_widget(historique.temps) }}</th>
                <th>{{ form_widget(historique.poste) }}</th>
                <th>{{ form_widget(historique.city) }}</th>
                <th></th>
            </tr>
        </thead>
        <tbody>
        </tbody>
    </table>
    {{ form_widget(historique.envoyer ) }}
    {{ form_end(historique) }}
</form>

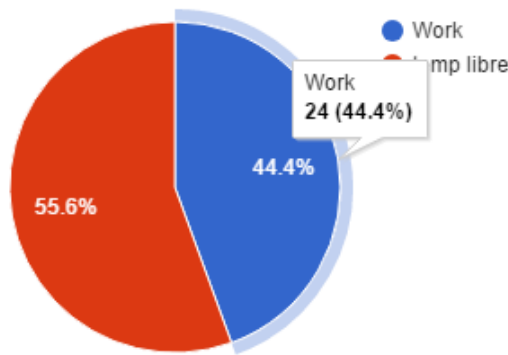
```

## Animation des Charts:

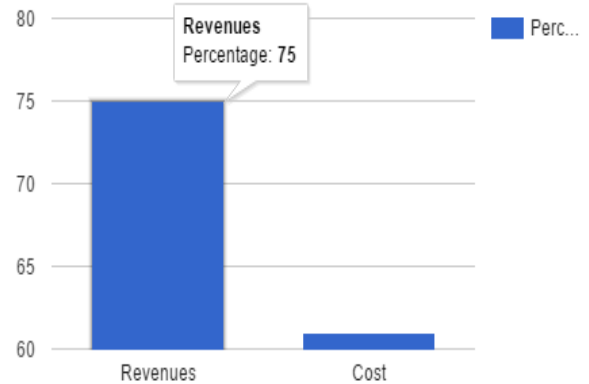
Ce formulaire en lui-même est rien, c'est juste une étape pour le fonctionnement des autres applications. Dans cette partie je présente le « PieChart » pour le premier et « ColumChart » pour Revenue & Costs pour le deuxième.

(J'ai juste intégré la partie « Temp libre » dans le premier et « Revenue & Costs », faute de temps pour créer l'application les valeurs ont été écrites directement en BDD ). Pour animer ce Chart je prend les données du temps de chaque employeur grâce au FindAll, pareil pour le Revenue&Cost.

## Vous êtes toujours au travail



## Revenue & Cost



## Code PieChart :

```

/***** Pie Chart *****/
//j'utilise un findAll pour recuperer dans ma Table "Temps " et Historique_employer
$donneesTemps = $this->getDoctrine()->getRepository(Historique_employer::class)->findAll();
$donneesTempsLibre = $this->getDoctrine()->getRepository(Temps::class)->findAll();
// je liste des données a ajouter
$temps = 0;
$tempsLibre = 0;

//j'utilise un findAll pour recuperer dans ma Table "revenue "
//toutes mes données en choisissant les colonnes :
// Temps dans l'entité Historique_employer et TempLibre dans Temps
foreach ($donneesTemps as $item) {
    $temps += (string) $item->getTemps();
}
foreach ($donneesTempsLibre as $item1){
    $tempsLibre += $item1->getTempLibre();
}
// Création de l'entité PieChart
$pieChart = new PieChart();
$pieChart->getData()->setArrayToDataTable([
    [ 'Task', 'Hours per Day' ],
    [ 'Work', $temps ],
    [ 'temp libre', $tempsLibre ]
]);
$pieChart->getOptions()->setTitle('Vous êtes toujours au travail');
$pieChart->getOptions()->getTitleTextStyle()->setFontSize(20);
$pieChart->getOptions()->setHeight(300);
$pieChart->getOptions()->setWidth(500);
    
```

Le (string) a \$temps += (string) \$item->getTemps(); a été ajouté parce que le type est en (Int) sur la colonnes (Temps). Donc pouvant pas le convertir en Int j'ai décidé de le casté en (String). Pour le fonctionnement de mes deux Charts j'ai volontairement pas modifier ces Type en String. Je vais expliquer sur le prochain chart.

## Revenue&Cost Chart :

```
//***** Column Chart controller *****/

//j'utilise un findAll pour recuperer dans ma Table "revenue "
//toutes mes données en choisissant les colonnes :
// "Reveue" et "Cost"
$item = $this->getDoctrine()->getRepository(Revenues::class)->findAll();
// je stock mes variable "revenu" et "cost"
$revenu = 0;
$cost = 0;

// je fais un foreach
foreach ($item as $item){
    $revenu += $item->getRevenu();
    $cost += $item->getCosts();
}

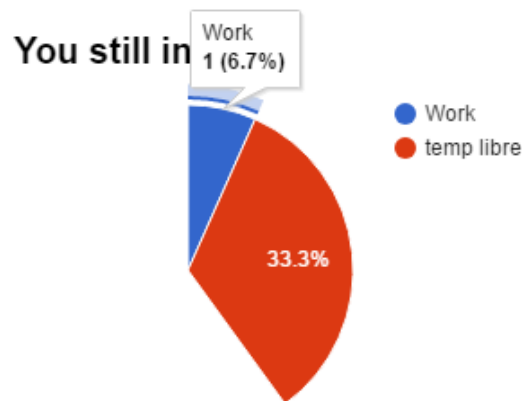
$col = new ColumnChart();
$col->getData()->setArrayToDataTable(
    [
        ['Move', 'Percentage'],
        ["Revenues", (int) $revenu],
        ["Cost", (int) $cost],
    ]
);

$col->getOptions()->setHeight(300);
$col->getOptions()->setWidth(500);
$col->getOptions()->setTitle('Revenue & Cost');
$col->getOptions()->getTitleTextStyle()->setFontSize(20);

// dump($cost);die();
```

Sur ce Code j'ai casté \$cost et \$revenu en (int) pour le fonctionnement de mon Chart. Après plusieurs heures ou jours je m'étais rendu compte que mes charts voulaient pas prendre en compte le fait que mes Types étaient en Varchar, les deux Charts rencontraient des problèmes, donc j'ai trouvé la solution en faisant cette technique. J'ai pas voulu remodifier mes Types parce qu'à mon avis ils sont mieux en VARCHAR.

ps: Probleme rencontrer avec les VARCHAR



Pour finir la vue de mon twig pour les deux Charts :

```
<div class="card-body collapse in">
  <div class="card-block">
  </div>
  <div id="div_ColumnChart"></div>
  <div id="div_PieChart"></div>

  <script type="text/javascript"
    src="https://www.gstatic.com/charts/loader.js"></script>
  <script type="text/javascript">
    {{ gc_draw([col,piechart ], [ 'div_ColumnChart','div_PieChart']) }}
  </script>
</div>
```

## 5 – 3 - 1Exemple de code HTML :

Code de mon formulaire ( login ) :

Message d'erreur .

```
{% if error %}  
    <div>{{ error.messageKey|trans(error.messageData, 'security') }}</div>  
{% endif %}
```

Formulaire avec les message de traductions pour le site francais.

```
<div class="card-body collapse in">  
    <div class="card-block">  
        {% if error %}  
            <div class="alert alert-danger">  
                {{ error.messageKey|trans(error.messageData, 'security') }}  
            </div>  
        {% endif %}  
        <form action="{{ path('/login_check') }}" method="post">  
            {% if csrf_token %}  
                <input type="hidden" name="_csrf_token" value="{{ csrf_token }}" />  
            {% endif %}  
            <fieldset class="form-group has-feedback has-icon-left mb-0">  
                <label for="username">{{ 'security.login.username'|trans }}</label>  
                <input type="text" id="username" name="_username" value="{{ last_username }}" />  
                <div class="form-control-feedback" style="position: absolute; top: 0; right: 0; font-size: 1.2em; color: #737373; width: 20px; height: 20px; border: 1px solid #ccc; border-radius: 50%; text-align: center; line-height: 20px; margin: 0 auto; display: flex; align-items: center; justify-content: center;"></div>  
                required="required"/>  
            </fieldset>  
            <fieldset class="form-group has-feedback has-icon-left">  
                <label for="password">{{ 'security.login.password'|trans }}</label>  
                <input type="password" id="password" name="_password" required="required"/>  
                <div class="form-control-feedback" style="position: absolute; top: 0; right: 0; font-size: 1.2em; color: #737373; width: 20px; height: 20px; border: 1px solid #ccc; border-radius: 50%; text-align: center; line-height: 20px; margin: 0 auto; display: flex; align-items: center; justify-content: center;"></div>  
            </fieldset>  
            <input type="checkbox" id="remember_me" name="_remember_me" value="on"/>  
            <label for="remember_me">{{ 'security.login.remember_me'|trans }}</label>  
            <input type="submit" class="btn btn-primary btn-lg btn-block" id="_submit" />  
            </form>  
    </div>  
</div>
```

## 5 – 3 - 2Code PHP création d'un formulaire

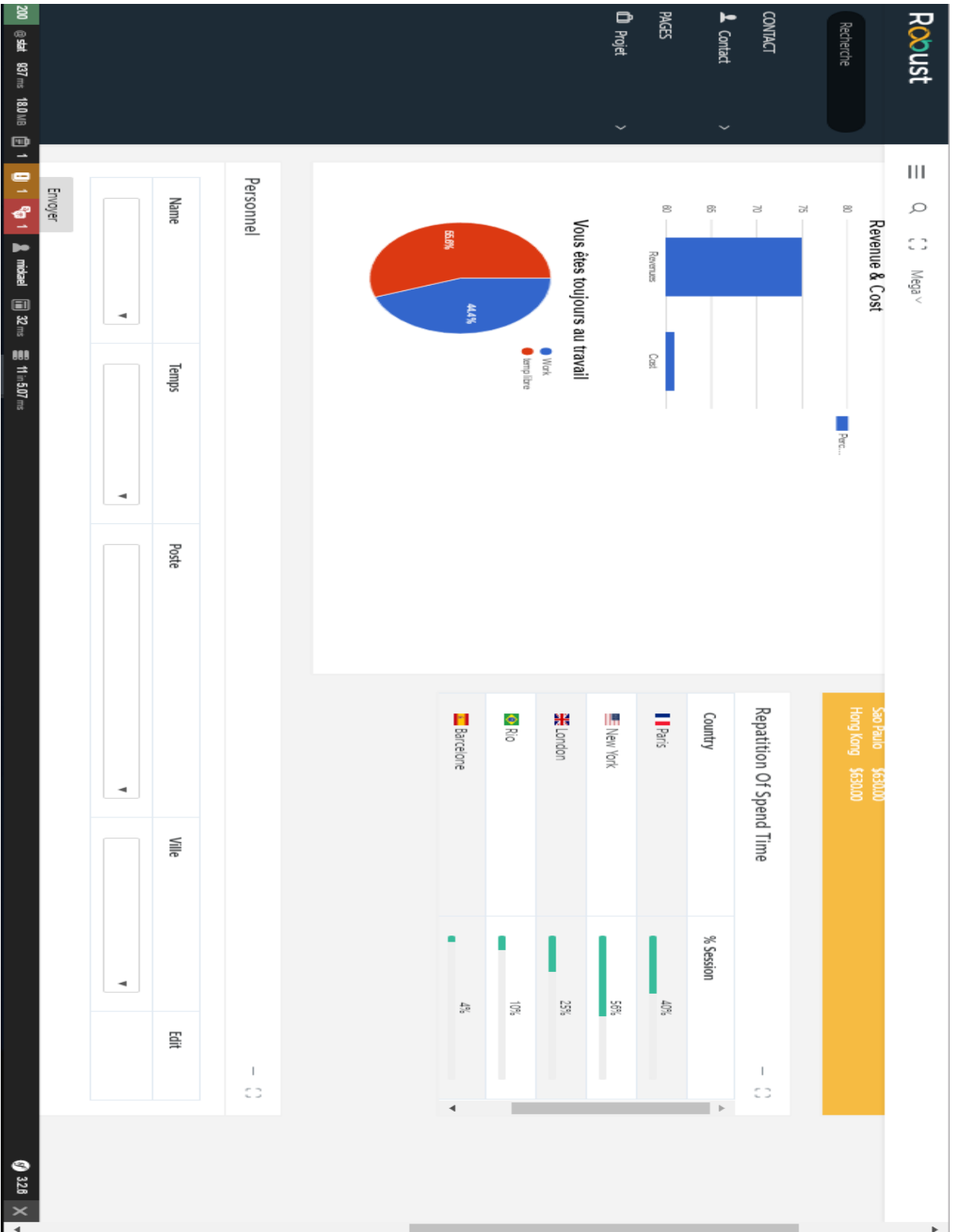
```
/**
 * @deprecated
 * L'object est de migrer toutes tes actions dans le UserBundle ce sera bcp plus simple à lire
 * et évoluer
 */
class UserController extends Controller
{
    /**
     * @Route("/message", name="message")
     * @param Request $request
     * @return \Symfony\Component\HttpFoundation\RedirectResponse|\Symfony\Component\HttpFoundation\Response
     */
    public function contactUsAction(Request $request){

        $contact = new Contact();

        $form = $this->get('form.factory')->create(ContactType::class, $contact);
        if ($form->handleRequest($request)->isValid()) {
            $em = $this->getDoctrine()->getManager();
            $em->persist($contact);
            $em->flush();
            $this->get('session')->getFlashBag()->add(
                'success',
                'Votre message a bien ete envoyé';
            );

            return $this->redirectToRoute('/message');
        }
        return $this->render('bloc/message', array(
            'contact' => $form->createView(),
        ));
    }

    /**
     * @return \Symfony\Component\HttpFoundation\Response
     * @Route("/personnel", name="personnel")
     */
    public function formationAction(){
        //Acces au service Doctrine
        $doctrine = $this->container->get('doctrine');
        $em = $doctrine->getManager();
        //Acces au repository de l'entitee
    }
}
```





# CHAPITRE 6 – LA SECURITE

## 6- 1 La sécurité de mon Application

La sécurité de mon application a été prise en compte par mon Framework Symfony mais aussi grâce à la sécurité de FosUserBundle. Le séculier de mon projet lors de ce stage se focalisait majoritairement sur les user.

security:

```
# http://symfony.com/doc/current/security.html#b-configuring-how-users-are-loaded
encoders:
    # JAMAIS INSERER LES MOTS DE PASSE EN CLAIR EN BDD!!!!
    #Symfony\Component\Security\Core\User\User: plaintext
    FOS\UserBundle\Model\UserInterface: bcrypt

role_hierarchy:
    ROLE_ADMIN:          ROLE_USER
    ROLE_SUPER_ADMIN: [ROLE_ADMIN, ROLE_ALLOWED_TO_SWITCH]

providers:
    fos_userbundle:
        id: fos_user.user_provider.username

firewalls:
    # disables authentication for assets and the profiler, adapt it according to your needs
    dev:
        pattern: ^/(_(profiler|wdt)|css|images|js)/
        security: false

    main:
        pattern: ^/
        form_login:
            provider: fos_userbundle
            csrf_token_generator: security.csrf.token_manager

        logout:      true
        anonymous:    true

access_control:
    - { path: ^/admin/users, roles: ROLE_SUPER_ADMIN }
    - { path: ^/admin, roles: ROLE_ADMIN }
    - { path: ^/login, roles: IS_AUTHENTICATED_ANONYMOUSLY }
    - { path: ^/, roles: ROLE_USER }
```

## Routing.yml

```
fos_user_security:
    resource: "@FOSUserBundle/Resources/config/routing/security.xml"
```

# **CHAPRITRE 7 - CONCLUSION**

## 7-1 :Conclusion Général.

Mon but dans ce stage était avant tout me former dans une nouvelle activité professionnel. J'ai connue des haut et des bas mais j'ai su convertir en point positif. J'ai douté de mes capacité les deux premières semaines, mais je me suis ressaisie en me disant pourquoi je me suis inscrit a cette formation et pourquoi je voulais devenir DEVELOPPEUR.

Il m'a permis de revoir un grand nombre de choses que j'avais apprises durant la formation (CSS,JAVASCRIPT, MySQL, Boodstrap etc..) mais Jai pu approfondir mes connaissant sur Symfony.

Les autres jour on été plutôt encourageant vu mes progrès a résoudre des problèmes complexe. Cette expérience ma rendu plus autonome ( seul développeur) dans mon futur métier et pour cela je suis fier des progrès parcourus.

Mon point négative est que j'ai malheureusement pas pu rendre un travaille terminer au cause du manque de temps mais aussi le fait que j'étais seul. Après ce stage je souhaite continuer de travailler dur pour encore plus me perfectionner et me consacré a fond sur PHP et Symfony. J'ai pour projet personnel « d'enfin » terminé mon CV web fait avec symfony 3.

Cela est qu'un début pour moi dans mon cheminement pour être développeur.

# CHAPITRE 8 - ANNEXES

## 8 - 1 : Webographie et Bibliographie :

Les choses qui m'a beaucoup aidé durant ma période de stage  
Sont « les documentation Officiel » comme ceux de :

- OpenClassRoom ( les forums d'entraide et les cours )
- Developpez.com
- starkoverflow.com
- PHP/Symfony ( pour certains de mes erreurs et FosUserBundle)
- MySql
- W3school ( pour Boodstrap et HTML/CSS )
- Cours de Mr Buguet Pascal (UML , PowerAMC )
- Course de Mr MALORON Sébastien ( Les supports de cours Symfony ).
- Pratique de MySQL et PHP, livre de Philippe Rigaux .

## 8 - 2Correspondance Projet et Reac

Compétence	Correspondance
Maquetter une application	UML (DCU, Maquettes, DSEQ, DNAV, DAC, DET) StarUML, Power AMC
Développer des composants d'accès aux données	PHP/Symfony
Développer des pages web en lien avec une base de données	HTML, CSS, JavaScript Php/Symfony
Préparer et exécuter le déploiement d'une application	Diagramme de deploiement (UML) Installation des codes HTML, CSS et JavaScript. Instalation du projet Symfony.
Concevoir une base de données	Modèle Conceptuel des Données (MCD) UML (Diagramme de classes) Modèle Physique des Données (MPD) avec MySQL WorkBench et ORM doctrine symfony
Mettre en place une base de données	Création de la BD, Création des tables et mise en place des contraintes (FK) avec phpMyAdmin
Développer des composants dans le langage d'une base de données	Gestion de la persistance avec un Entity (ORM)
Développer des composants métier	Entities
Concevoir une application	UML (Diagramme de Séquence Détaillé,
Collaborer a la gestion d'un projet informatique	Google Hnagout

### 8 - 3 Technologies Choisie:

- Comme éditeur de Texte j'ai choisi de travailler avec **PhpStorm** qui pour moi a pas de rivale pour travailler avec Symfony. Simple a utilisé il ma beaucoup aidé .
- Je travaillais avec un server local donc j'ai choisi **Xampp** simple d'utilisation aussi . Et donc pour la base de données j'étais avec **PhpMyAdmin**.
- Pour la modélisation des mes ScriptSql et Schema UML j'utilisais ( **WorkBench, StarUml ,PowerAMC** )
- GitBach** pour le ligne de commande .
- Pencil** pour mes maquettes.
- GanttProject** Pour mon Diagramme de Gantt

