

Dossier de synthèse   
du projet « **Annu’R** ».  
Présenté pour l’obtention du titre de

**Concepteur Développeur d’Applications**

**Christophe COUCHY**Paris, Doranco, le 04/10/2019

**Remerciements**

Je remercie le centre de formation Doranco pour les moyens humains et l’infrastructure  
mis à disposition pour réaliser cet apprentissage.   
Ainsi que pour mettre à disposition un personnel intervenant pour animer les modules de formations.  
Enfin, merci de m’avoir permis de me former et de passer aujourd’hui cette certification qui me permettra, je le souhaite, de donner un nouveau départ à ma vie professionnelle.

**Résumé**

Il s’agit d’un projet personnel réalisé suite à l’expression d’un besoin dans mon ancienne entreprise. J’ai eu la chance de travailler au siège de cette société, qui est l’un des leaders européens en matériel de bureau, matériel scolaire et fournitures.

J’ai pu constater que du fait du nombre important de salariés, beaucoup ne se connaissaient pas les uns les autres, et il était fréquent de rechercher le nom d’une personne, ou son numéro de téléphone professionnel.  
L’intranet n’ayant pas été entretenu au fil des années, le manque d’un simple trombinoscope se faisait donc sentir.  
A l’époque des réseaux sociaux, il sera intéressant de créer un type de « Facebook local ».

Pour réaliser ce projet, je vais créer une application web JEE, avec   
- une partie Back-End sous technologie Java, avec le framework Spring.  
- une partie Front-End réalisé avec le framework Angular.

La finalité d’un tel projet n’est nullement lucrative. Il s’agit plus simplement de permettre de connecter les gens entre eux.

**Abstract**

It is a personal project realized following the expression of a need in my old company.   
I was fortunate enough to work at the headquarters of this company, which is one of the European leaders in office equipment, school equipment and supplies.

I found that because of the large number of employees, many did not know each other, and it was common to search for a person's name, or their business phone number.

The intranet has not been maintained over the years, the lack of a simple trombinoscope was felt.

At the time of social networks, it will be interesting to create a type of "local Facebook".

To realize this project, I will create a JEE web application, with

- a Back-End part with Java technology, with the Spring framework.

- a Front-End part realized with the Angular framework.

The purpose of such a project is not lucrative. It's more simply about connecting people to each other.

**Présentation**

Je suis Christophe COUCHY, 38 ans, célibataire sans enfants.  
Après une formation mi-artistique (diplômé en infographie print et 3D) mi-informatique (diplômé d’un BTS informatique de gestion, option administrateur réseau), j’ai pu obtenir « le meilleur des deux mondes » en devenant webdesigner (après avoir appris le HTML et le CSS en autodidacte).

J’ai ainsi pu, dans mon ancienne société (une entreprise de e-commerce), réalisé quelques projets servant les intérêts de la société (des refontes graphiques et fonctionnelles du site de vente e-commerce vendu aux clients partenaires).  
C’est ainsi qu’à commencer mon appétence pour le développement informatique. D’abord tourné exclusivement vers le développement FrontEnd (HTML, CSS, JS), la curiosité m’a poussé à découvrir le développement BackEnd.

J’ai donc pu profiter, quelques années plus tard, d’une rupture conventionnelle, pour me consacrer à mon projet de reconversion professionnelle.

Pour réaliser cette démarche, je me suis appuyé sur le site « défi-métier » ou j’y ai trouvé une  
formation conventionnée dispensée par le centre de formation « Doranco », et pour aller jusqu’au  
bout de mon projet de formation.

Aujourd’hui, mon objectif est bien sûr de décrocher un emploi en tant que développeur informatique, et, à plus ou moins long terme, évoluer en tant que développeur FullStack.

**Présentation du projet**

**Introduction**

Travaillant dans la société d’eCommerce où j’ai été employé durant 6 ans, j’ai été confronté au problème suivant : il n’y avait pas, dans cette société, de trombinoscope, ni de quelconque annuaire en ligne.  
Pour une société dont le siège emploi plus de 300 personnes, chacune ayant un bureau et un numéro de téléphone correspondant, c’était un manque pourtant évident.

Cette réflexion de ma part s’était fait à l’époque où je venais de terminer mon projet précédent de refonte du site e-commerce, à grand renfort de CSS et de JavaScript, voire de jQuery.  
L’idée de démarrer un nouveau projet était donc très motivante.

Rapidement, des idées vinrent se greffer à cette ébauche de projet : la possibilité de se connecter, de se créer un véritable profil, de discuter avec d’autres….. Très rapidement, ce simple annuaire en ligne devenait une sorte de Facebook d’entreprise….

**Mais alors, pourquoi réinventer la roue ?**

En effet, il serait facile de se demander pourquoi refaire ce qui existe déjà ?  
La première des raisons concerne bien sûr le public visé. Mon objectif n’est pas que chaque employé de la société passe ses journées sur Facebook, (je le fais déjà bien assez moi-même ^^) mais de cibler avant tout la communication entre les employés de la société uniquement.  
Cela peut inclure des moyens de communications entre les employés, mais davantage pour renforcer l’esprit de grande famille, si cher à beaucoup de sociétés.

Cependant, je me suis heurté très rapidement à un problème de taille : Impossible de réaliser un tel projet avec mes seules connaissances en programmation front end. Une couche Back End s’avérait alors nécessaire (notamment une base de données pour stocker toutes les informations telles que les noms des contacts).

**Recueil des besoins de l’interface**

L’application va se présenter comme un site internet relativement simple, accessible depuis un navigateur internet, avec un menu pour voir accès aux différentes sections du site (A propos, liste des contacts, etc….).

La navigation, possible sur desktop et mobile, sera simple, logique et intuitive.   
Dès la première page, l’utilisateur sera invité à se connecter. Il aura ainsi accès à son espace, et à sa liste de contacts.   
Chaque utilisateur aura en effet, en plus de la liste globale, accès à une liste de contacts « amis ».

L’utilisation se veut principalement depuis une machine de bureau, mais l’accès à un portail mobile sera un plus indéniable

Les besoins orientés métier :

* L’utilisateur doit pouvoir se connecter (l’accès à l’application est réservé à des utilisateurs authentifiés.
* S’il n’a pas déjà de compte, le visiteur (pas encore utilisateur donc) doit pouvoir se créer un compte. Il sera alors redirigé vers un formulaire de création de compte.
* Une fois connecté, l’utilisateur peut consulter la liste des contacts. Un simple coup d’œil à la liste des utilisateurs lui donne accès aux informations les plus basiques d’un annuaire tels que le numéro de téléphone, le nom complet, le numéro de bureau).
* Cliquer sur un contact de la liste lui donne accès à la fiche plus détaillé s’il est amis avec ce contact.
* Si l’utilisateur possède les droits suffisants, il peut ajouter, supprimer ou modifier une fiche contact.

**Description des technologies utilisés :**

Pour ce projet, j’ai fait le choix des technologies suivantes :

**Front-End :**   
- Le Framework Angular (Google) sera utilisé. Il permet la mise en place de SPA (single page applications) assez simplement.  
Notamment, il permet d’utiliser HTML, CSS, et JS. Ce dernier est « transpilé » depuis des fichiers écrits en TypeScript.  
- Le Framework Bootstrap (Twitter) sera également utilisé pour sa patte graphique reconnaissable et surtout pour rendre le tout responsive, c’est-à-dire compatible toute taille d’écran, et notamment écran de smartphones et tablettes.

**BackEnd :**  
- La technologie Java a retenu ma préférence. C’est celle où je me sens le plus à l’aise, notamment en utilisant Spring, et les spécifications JPA (Java Persistance API).   
- Spring Data JPA : Il permet d’automatiser en grande partie la couche d’accès aux données. Il va de fait générer les requêtes pour interagir avec la base de données.

MicroService REST :

SGBD :  
- Pour ce projet, le SGBD (Système de Gestion de Base de Données) retenu sera MySQL.

**Mentions légales (RGPD)**

**Référencement, sécurité, sauvegarde**

Les besoins orientés métier :

* L’utilisateur doit pouvoir se connecter avec un login et un mot de passe
* S’il n’a pas de compte, il peut s’en créer un
* Une fois connecté, il est possible d’ajouter des contacts à la base
* Il est possible de rectifier (modifier) des données de contacts
* Il est possible de supprimer un contact
* (À voir) Il est possible d’ajouter des contacts dans sa liste de contacts pour échanger discussions privées.

**Utilisation de la méthodologie Agile**

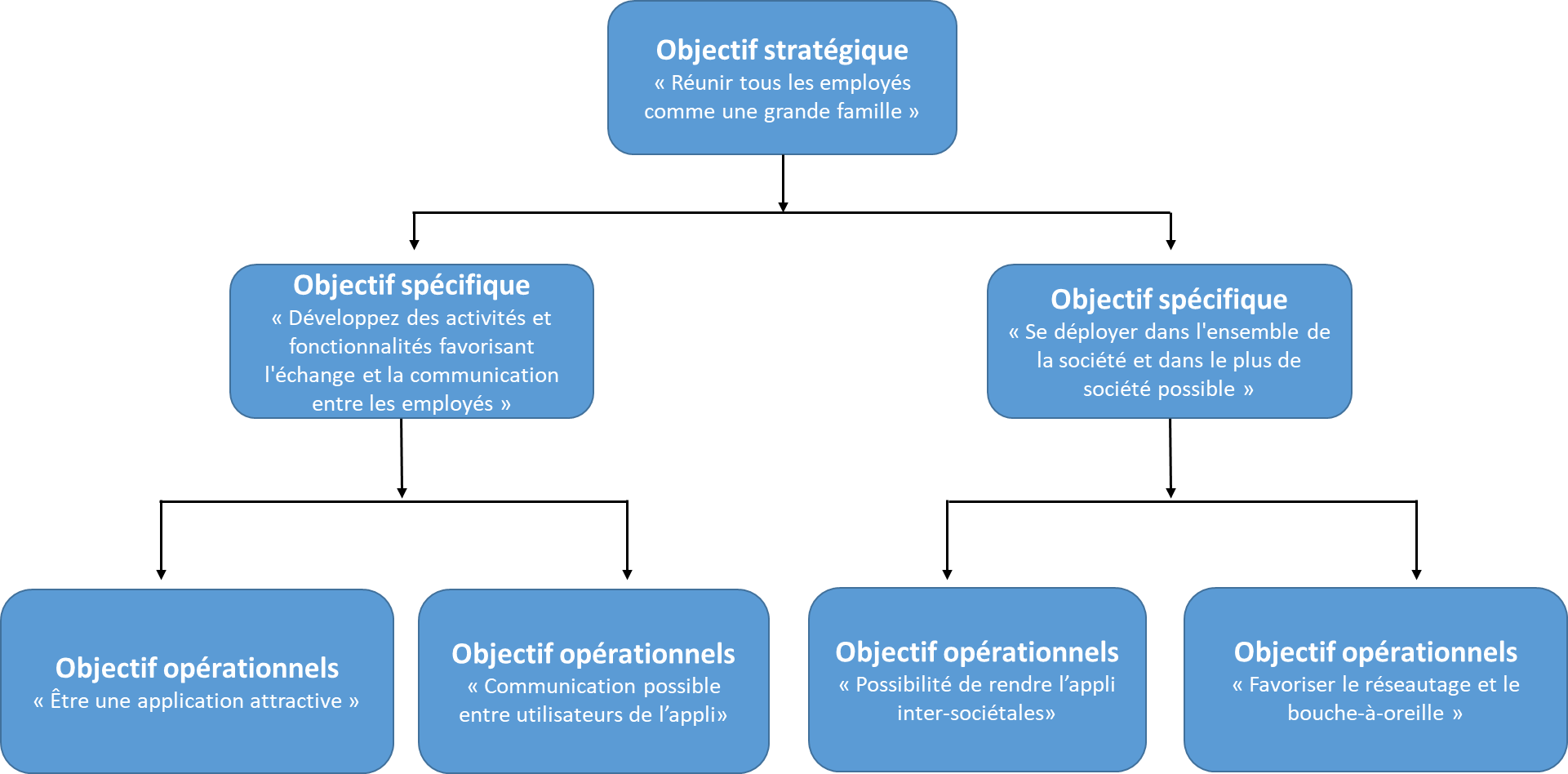
Au cours de ma formation de concepteur-développeur, j’ai notamment appris à utiliser la méthodologie AGILE pour l’organisation du travail en amont du développement.  
Je vais m’appuyer sur la méthode SCRUM pour organiser le déroulement de mon projet.

Etant donné que je suis seul sur ce projet, je vais donc jouer tous les rôles prévues de la méthode AGILE, à savoir le PO (Product Owner), le Scrum Master et l’équipe de développement.

La méthode Agile recommande de se fixer des objectifs à court terme. Le projet est donc divisé en plusieurs sous-projets. Une fois l'objectif atteint, on passe au suivant jusqu'à l'accomplissement de l'objectif final. Cette approche est plus flexible. Puisqu'il est impossible de tout prévoir et de tout anticiper, elle laisse la place aux imprévus et aux changements.

**L’arbre à objectifs**

En premier lieu, je vais utiliser un arbre à objectif afin d’offrir une vue simple des objectifs du projet :



**Backlogs**

**CONNEXION/CREATION DE COMPTE**

En tant que user, je peux créer un compte.  
En tant que user, je ne peux rien faire avant de m’être connecté.  
En tant que user connecté, je peux me connecter à l’application.  
Tester que la connexion fonctionne et que la sécurité empêche toute intrusion non autorisée.

**DECONNEXION**

En étant connecté, je peux me déconnecter, ce qui me renvoi à la page de login.

**SOCIABILITE**

En tant que user connecté, je peux apercevoir une liste des contacts inscrit (mais pas leur fiche complète).  
En tant que user connecté, j’ai la possibilité de demander une mise en relation avec d’autres personnes.  
Si ma demande est accepté, je peux voir la fiche complète du contact , et également discuter en direct (type Messenger).  
Tester que la visualisation du compte complet n’est visible qu’après validation des deux parties.

**LIEN INTER-ENTREPRISE**

L’application est par défaut liée au réseau local de l’entreprise, et de fait, non accessible en dehors de l’entreprise.  
Cependant, dans le cas où les deux parties le désire, il est possible que deux sociétés distinctes partagent le même réseau commun.  
Il est important pour la sécurité que cette fonction ne soit possible qu’après validation des deux (ou plus) sociétés.

**ATTRACTIVITE**

En tant que user connecté, je peux retrouver la vie d’entreprise via cette application (soit une utilisation similaire aux intranet, en plus moderne).  
L’application fait également office de trombinoscope. Il est possible d’imaginer une liaison avec un produit type skype, et un client mail afin de centraliser toute la communication entreprise autour du même logiciel.

**Personas et actions possibles**

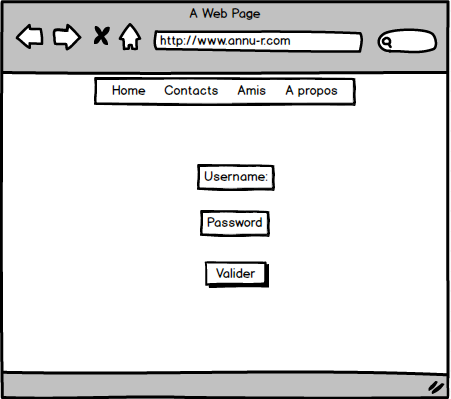
La méthode AGILE préconise la création de « personas », ce sont des acteurs virtuels chargés de représenter des personnes ou systèmes afin à décrire des interactions. En tant que Product Owner, je procède à la création de mes personas : seulement trois suffiront à décrire les situations.

* Visiteurs (non connectés)
* Utilisateurs (connecté)
* Administrateurs (possède tous les droits)

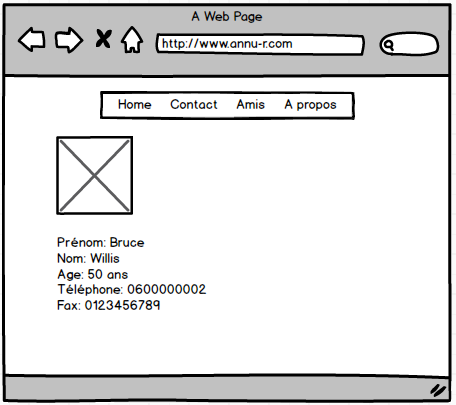
|  |  |  |  |
| --- | --- | --- | --- |
|  | Administrateur | Visiteur | Utilisateur |
| Créer un compte |  | OUI |  |
| Se connecter | OUI |  | OUI |
| Se déconnecter | OUI |  | OUI |
| Consulter liste contacts | OUI |  | OUI |
| Ajouter /modifier/ supprimer une fiche contact | OUI |  |  |
| Ajouter des contacts « amis » | OUI |  | OUI |
| Consulter fiche « details contact » | OUI |  | OUI |

**Maquettage – Structure du site**

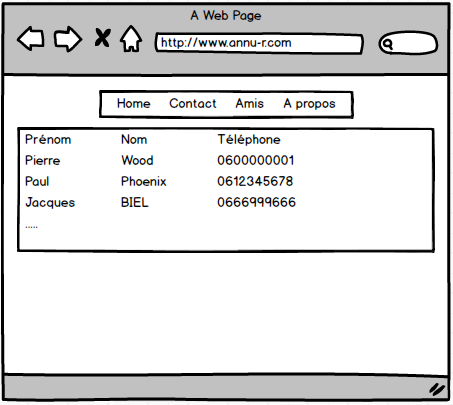
La page de login :



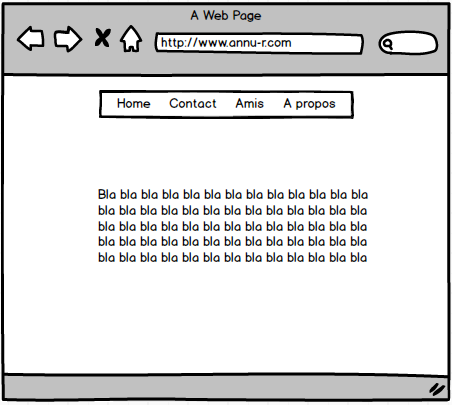
Page Détails de profil :



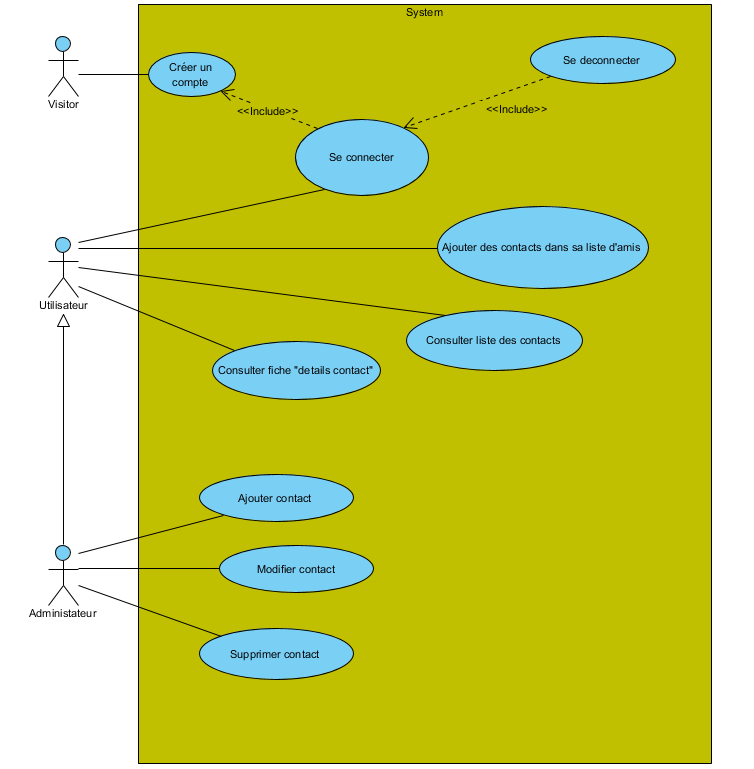
La liste des contacts (la liste d’amis aura la même forme):



La page « A propos »



**Diagramme de cas d’utilisation**



**Décrire la partie bdd**

Base de données utilisateurs  
Base photos (plusieurs photos possibles par utilisateurs)

Schéma UML (bdd ou classes)

Creation de la classe « Contact »

Cette classe(entité) sera « sérialisable » et sera défini par :  
Un ID  
Nom  
Prénom  
Date de naissance (idéalement en menu déroulant select)  
Email  
Photo (non obligatoire)  
Téléphone (non obligatoire)  
MotDePasse (devra contenir au moins 7 caractères, une majuscule, une minuscule, un charactère spécial, un chiffre)  
Sexe

Pour chaque entité créée, on va utiliser l’annotation @Entity. Cela permettra de créer facilement la bas de données correspondante.

L’annotation @Autowired permet de faire l’injection des dépendances

SECURITE

La partie sécurité sera géré grâce à l’ajout de la dépendance SPRING SECURITY