

---

# PROJET TUTORÉ

## LE BON TALENT

UPMC - LICENCE DÉVELOPPEMENT D'APPLICATION AUX NOUVELLES  
TECHNOLOGIE

---



---

TUTEUR ACADEMIQUE : FABIAN PALACIOS  
ETUDIANT : ZAKARIA AMRI  
ANNEE : 2015 - 2016

---

---

## Remerciement

Tout d'abord je tiens à remercier mon professeur encadrant monsieur Fabian Palacios pour avoir su se montrer disponible dès que je lui ai demandais de l'aide dans le cadre de cette unité d'enseignement. Et de m'avoir fait une suivie permanente tout au long du projet pour le bon déroulement du projet.

Ensuite, je souhaite remercier tous les professeur m'ayant permis d'avoir les compétences nécessaires pour réaliser ce projet.

---

## Sommaire

**Introduction .P4**

**Application : Le bon talent .P5**

**Présentation .P5**

**Fonctionnalités .P5**

**Diagramme d'activité .P6**

**Les besoins de l'applications .P7**

**Les technologies .P7**

**Qu'est ce que le cloud computing .P7**

**Google Cloud .P7**

**Choix du langage .P8**

**Structure de l'application .P9**

**Architecture REST .P9**

**Structure de la base de donnée .P10**

**Organisation des fichiers .P11**

**Application mobile IOS .P12**

**Conclusion .P13**

**Annexe .P14**

---

# Introduction

Dans le cadre de la licence « développement des applications aux nouvelles technologies », j'ai dû réaliser un projet personnel de la conception jusqu'au développement de celle-ci.

J'ai décidée de réaliser une application web/mobile, indépendante des tâches que j'effectue en entreprise, qui concerne plus de la gestion de projet que du développement d'application. L'objectif final de cette application est de créer une communauté.

Dans un premier temps, je présenterais mon application de manière générale, c'est à dire, l'objectif de celle-ci, les fonctionnalités et les besoin qu'elle nécessite.

Je définirais ce qu'est le cloud computing et j'expliquerais pourquoi j'ai décider de choisir Google App Engine .

Puis, je présenterais comment mon application fonctionne, autrement dit la structure de l'application et de la compatibilité avec les mobiles.

En Conclusion, je reviendrais sur ce que cette expérience m'a apporté.

# Application : Le bon talent

## Présentations

L'application « Le bon Talent » est une application gratuite qui permet à des personnes de rendre un service à un tiers (gratuit ou non) dans son domaine de prédilection. Ou bien effectuer une demande pour qu'on l'aide sur divers problèmes.

A ce jour, l'application présente 5 domaines :

- informatique : pour tous les problèmes hardware/software
- aide à domicile : pour tous ce qui aide à la personne
- bricolage : pour les problèmes matérielles autre qu'informatiques
- éducation : pour tous ce qui est de l'acquisition de connaissance
- autre : pour le reste

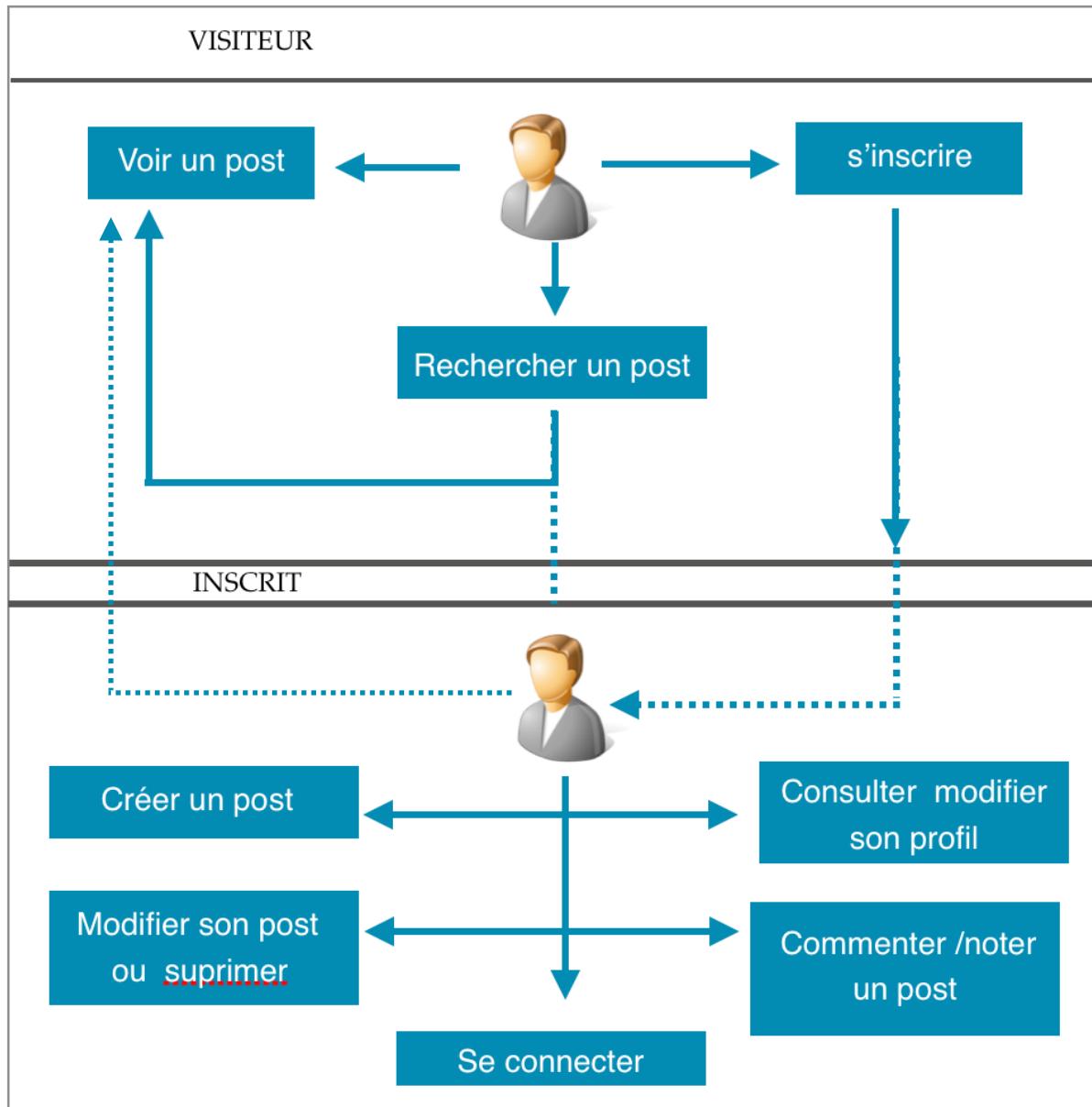
A fin d'avoir un aspect communautaire, les utilisateurs pourront laisser un commentaire au post ou le contacter indirectement par mail.

## Fonctionnalités

L'application offre à ces utilisateurs les fonctions suivantes :

FONCTIONNALITÉS	
NOM DE LA FONCTION	DESCRIPTION DE LA FONCTION
Inscription	Enregistre un utilisateur dans la base de donnée
Connexion	Permet à un utilisateur de se connecter via son mail et son mot de passe
Déconnexion	Ferme la session d'un utilisateur
Consulter son profil	L'utilisateur pourra voir toutes ces demandes / annonces et ces données personnelles
Créer un post	Une fois connecté l'utilisateur pourra créer une annonce ou une demande
Laisser un commentaire sur un post	L'utilisateur pourra ajouter un commentaire à une annonce ou une demande si il est connecté
Noter un post	L'utilisateur pourra ajouter une note à une annonce ou une demande si il est connecté
Modifier ou supprimer son post	Une fois connecté il peut modifier ou supprimer une de ces demandes ou annonces
Rechercher un post	Tout le monde peut rechercher une demande via certains critères

## Diagramme d'activité



Sur le diagramme ci-dessus, on peut voir les différentes actions que faire un utilisateur selon son statut.

---

## Les besoins techniques de l'application

Pour réaliser cette application, il faut :

- une base de données capable d'enregistrer, traiter, récupérer et modifier des informations de manière rapide
  - une technologie qui puisse traiter le côté serveur
  - une architecture REST pour gérer le côté client
  - le site doit avoir un design capable de s'adapter à n'importe quelle plateforme.
- J'ai décidé d'opter pour le « cloud computing » pour créer l'application

## Les technologies

### Qu'est ce que le cloud computing

Voici la définition du cloud computing :

« Le cloud computing ou informatique en nuage est une infrastructure dans laquelle la puissance de calcul et le stockage sont gérés par des serveurs distants auxquels les usagers se connectent via une liaison Internet sécurisée. L'ordinateur de bureau ou portable, le téléphone mobile, la tablette tactile et autres objets connectés deviennent des points d'accès pour exécuter des applications ou consulter des données qui sont hébergées sur les serveurs. Le cloud se caractérise également par sa souplesse qui permet aux fournisseurs d'adapter automatiquement la capacité de stockage et la puissance de calcul aux besoins des utilisateurs. » Le cloud permet donc d'héberger des applications (entre autres web/mobile) et de stocker/traiter une quantité de données rapidement. Le cloud permet au client de récupérer des données.

Les trois leaders dans ce domaine sont :

- Google
- Amazon
- Microsoft

J'ai opté pour le cloud de Google, étant donné que l'on me l'a conseillé, qu'il y avait pas mal de documentations et que le langage Java est supporté.

### Google Cloud

Il faut savoir que Google propose plusieurs types de services cloud :

- le service PaaS (Google App Engine qui permet d'avoir accès aux serveurs de Google pour stocker votre site, mais aussi des fonctionnalités très puissantes de gestion de la charge, un accès aux bases de données Google, aux serveurs d'envoi d'e-mails, etc.
- le service IaaS (Google Compute Engine) où Google fournit un accès à tout ou une partie de son infrastructure technique, c'est-à-dire à ses serveurs.
- des services de gestion de base de données SQL ou NoSQL avec Google DataStore, Cloud Storage et Cloud SQL.

---

## Choix du langage

Le langage de programmation choisi pour la réalisation du site est le langage Java

1.8. Pour les raisons suivantes :

- c'est un langage orienté objet
- il est optimisé
- il est portable
- il présente beaucoup d'API et de framework

## Gestion du projet

### Répartition des tâches

La réalisation du projet est répartie en quatre parties distinctes :

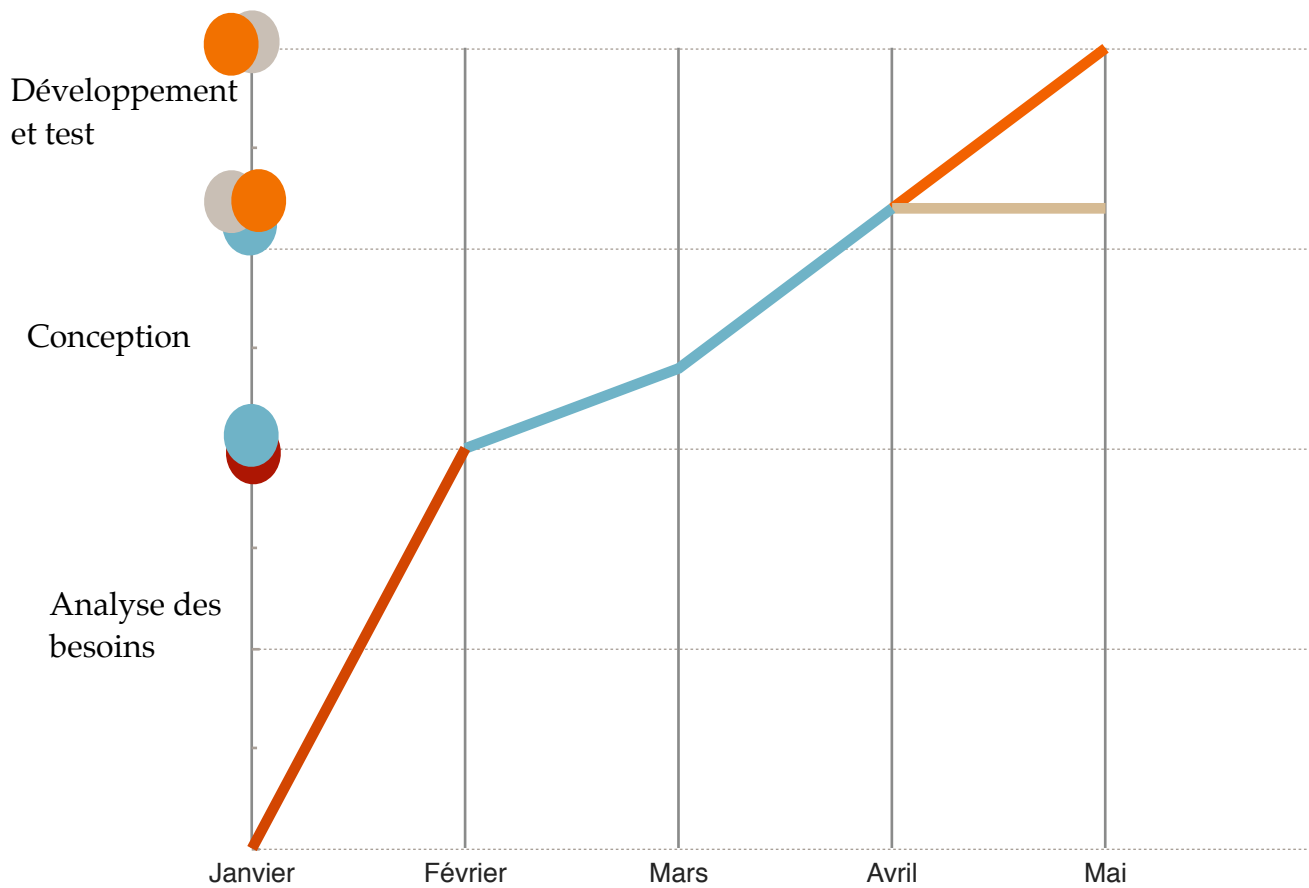
- l'analyse des besoins : chercher ce qui est attendu et demandé
- la conception : la partie la plus importante, il faut choisir la bonne technologies, architecture, langage et outils avec lesquelles on pourra réaliser le projet.
- le développement: coder le projet.
- le test : réaliser les différents cas de figure des fonctionnalités.

Ayant eu quelque problème de santé la dernière partie n'a pas pu être exécutée.

### Diagramme de Gant

Le diagramme de Gantt est un outil utilisé en gestion de projet et permet de représenter graphiquement l'avancement réel et/ou théorique du projet. Ici, il représente l'avancement réel du projet.





## Structure de l'application

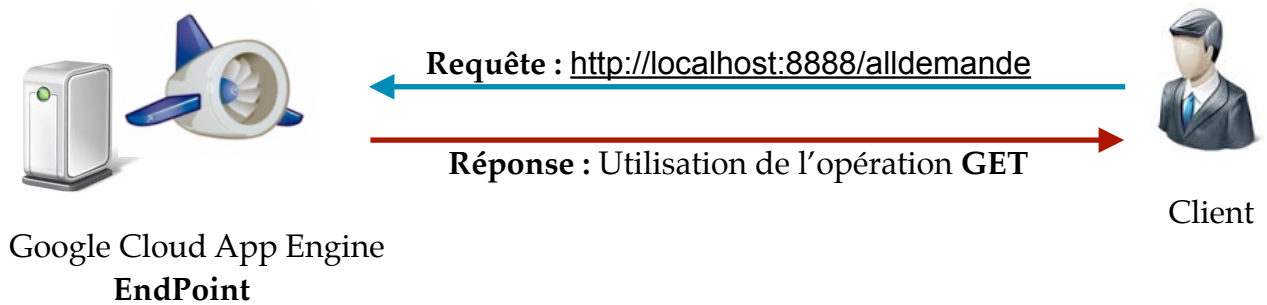
### Architecture REST

L'architecture de l'application est une architecture REST qui permet de construire des application web. Elle se base sur les URI afin d'identifier une ressource. Par exemple, <http://localhost:8888/alldemande> qui permet de récupérer tout les demande présente dans la base de donnée.

Quatre opération sont disponibles:

- **POST** : pour crée une ressource
- **GET** : pour récupérer une ressource
- **PUT** : pour modifier une ressource
- **DELETE** : pour supprimer une ressource

Voici un exemple d'interaction :



## Structure de la base de donnée

Etant donnée que mon application stocke beaucoup d'objets, j'ai décidé d'utiliser le Google Datastore pour gérer les données (de plus Google recommande l'utilisation de ce service). Le Datastore fonctionne de façon très simple : il accepte des clés et des valeurs, exactement comme une HashMap en Java. Pour une clé, on a une valeur. De plus, on peut regrouper plusieurs clés dans une seule entité et avoir un lien de parenté entre elle.

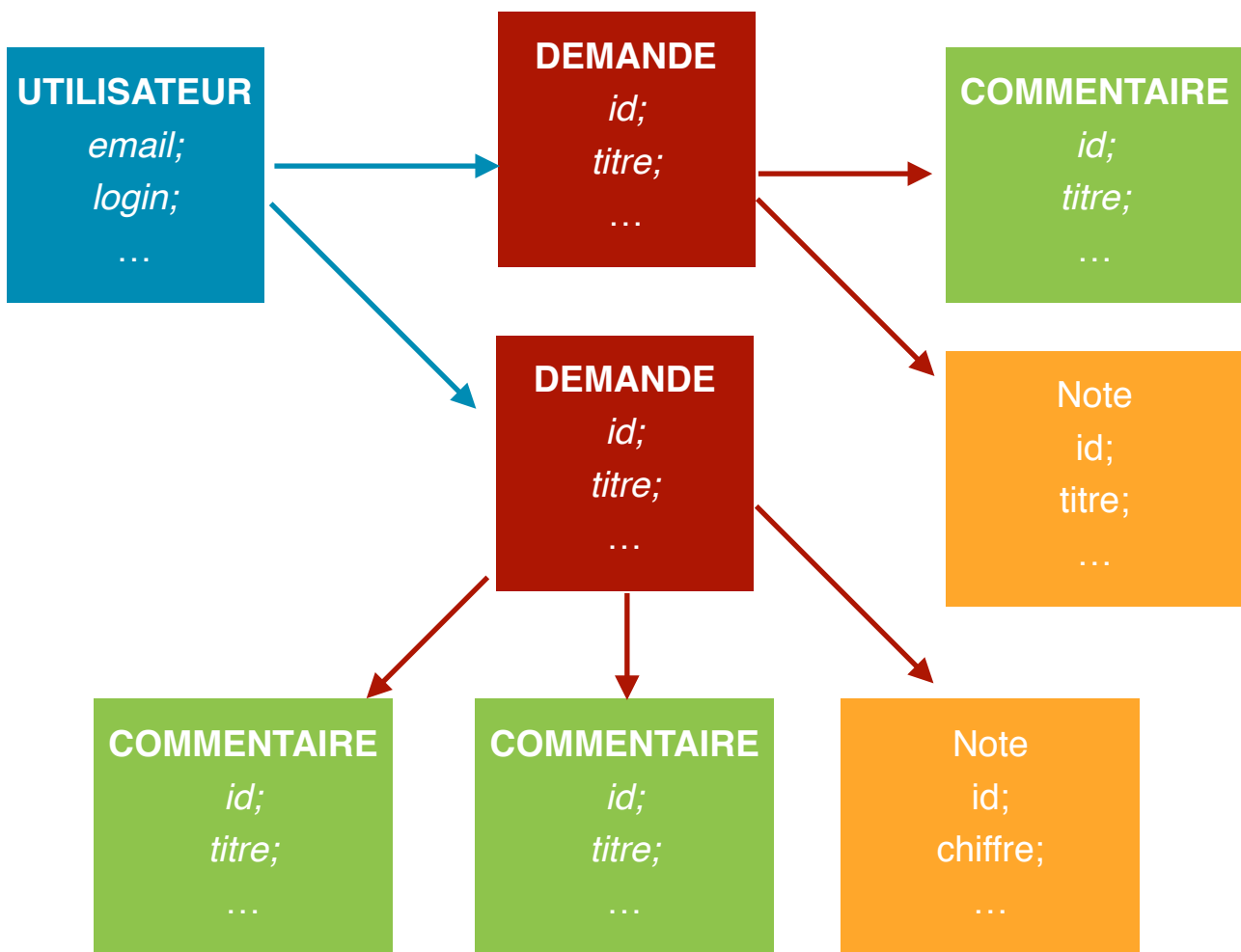
J'ai décidé d'organiser ma base de données de la manière suivante :

Il y aura 4 types d'entités:

- utilisateur
- demande
- commentaire
- note

Les entités demande sont les fils des entités utilisateurs. Les entités commentaire et note sont les fils de l'entité utilisateur.

On peut représenter ma base de donnée de la manière suivante :



## Organisation des fichiers java

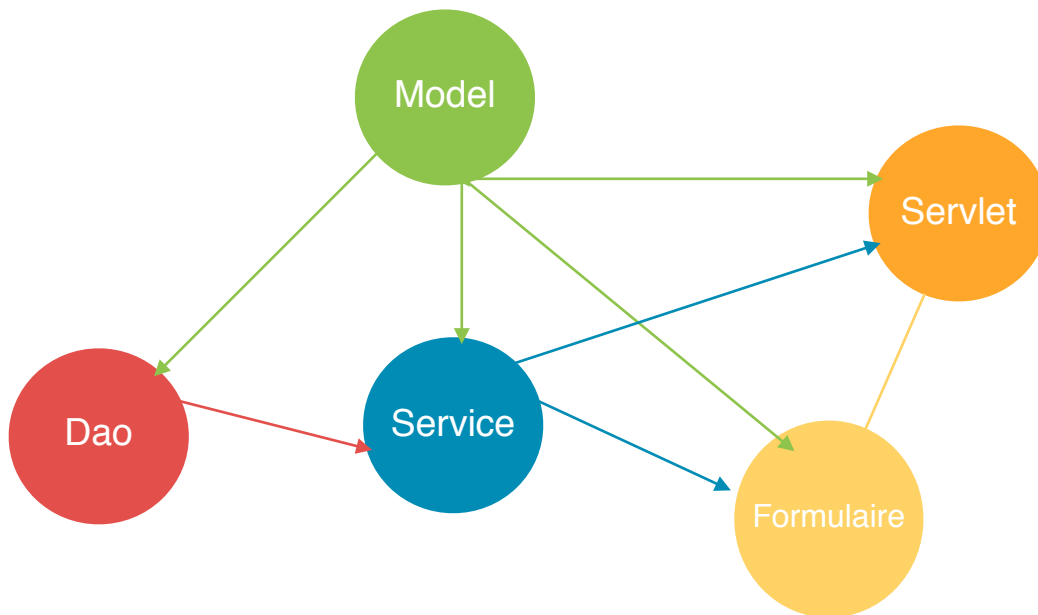
J'ai décidé de mettre en place un package par type d'action.

On a 5 packages:

- model : un template pour chaque objet que l'on va utiliser
- dao : interaction avec la base
- service : lien entre la dao et les autres packages
- servlet : génère les requêtes HTTP
- formulaire : récupère les données de certaines requêtes

---

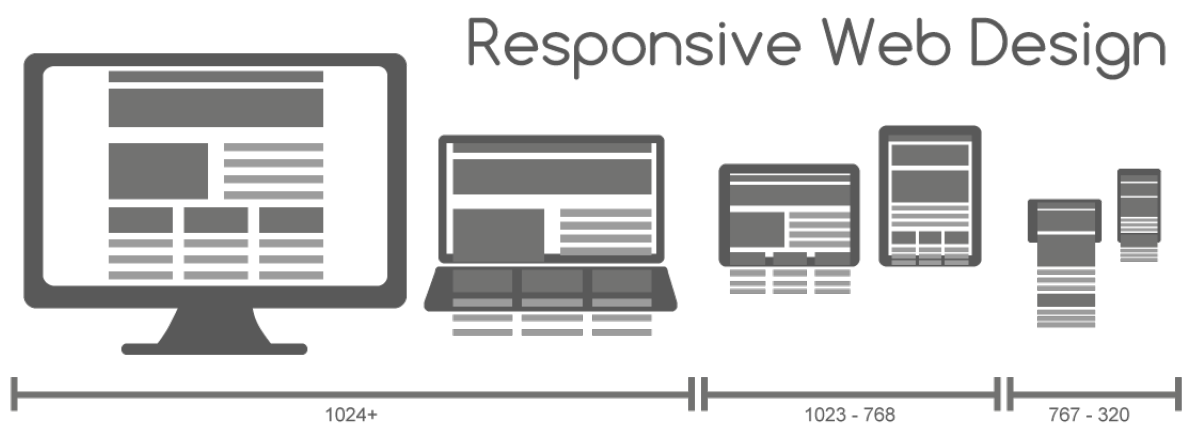
On peut Représenter graphiquement tout ça de la manière suivante :



Cette organisation m'a permis de résoudre pas mal de bug. Car on sait d'où vient l'erreur.

### Application mobile IOS

J'ai développé l'application en Responsive Design parfaitement lisible sur toute les plateformes. Les textes et les images d'adaptent à la taille de l'écran.



Grâce à cela j'ai pu développer une application ios, de manière simple et rapide. L'application ios, contient uniquement une UIWebView qui fait référence à mon application web.

---

## Conclusion

Ce projet a été très intéressant ,au tant sur le plan de la gestion que sur le développement. .

J'ai pu réaliser un site dynamique à l'aide du cloud de google, et approfondir mes connaissance dans des langages.

Enfin, j'ai pu comprendre le fonctionnement d'une base de donnée NoSQL et mettre en pratique ces connaissances.

# Annexe

## Quelques interfaces

