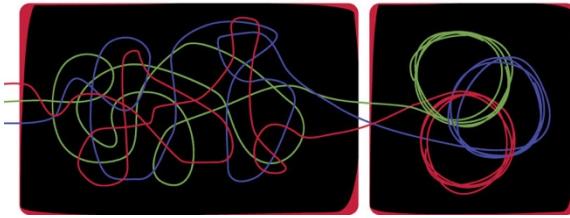


transformer vos stratégies
en logiques informatiques

norsys
EASymakers



RAPPORT DE STAGE

NORSYS

22 mai 2017

Sujet : Développement Mobile et Back Office

Enseignant : Sylvain Lagrue

Tuteur : Marine Fromentin – Grégoire Doussot



TABLE DES MATIÈRES

Table des matières

Table des Annexes	2
Remerciements	1
Introduction	2
L'Entreprise	3
Norsys	4
Organigramme	5
Contact	7
Les Formations	8
AnimaBox	12
Conclusion	18
Glossaire	19
Annexe	20

TABLE DES MATIÈRES

Table des Annexes

Annexe 1 : Cycle de vie IOS	20
Annexe 2 : Cycle de vie Android	21
Annexe 3: Fichier pod	22
Annexe 4: Authentification Firebase	23
Annexe 5: Creation d'un observateur	24

REMERCIEMENTS

Remerciements

Avant tout développement sur l'entreprise et sur les missions il me paraît juste de commencer ce rapport par quelques remerciements.

Premièrement, je voudrais remercier le corps enseignant de la Faculté de Lens qui m'a permis de développer mes compétences et d'avoir ainsi des bases théoriques pour pouvoir évoluer dans une entreprise.

Secondement je voudrais remercier Marine FROMENTIN et Grégoire DOUSSOT pour m'avoir accompagné tout au long de mon stage et d'avoir été là pour me donner des missions très intéressantes et m'aider lorsque certains problèmes survenaient.

Ensuite, j'aimerais remercier Laurent Payent, Benoît Hermant, Florian Duhen et Aurélien Delrue pour leurs formations dans les différents domaines que j'ai rencontré durant mon stage et qui m'ont été indispensables.

Et pour finir, je voudrais remercier toutes les personnes de Norsys qui m'ont permis de bien m'intégrer au sein de la société ainsi que de faire évoluer mes compétences tout au long de ce stage.

INTRODUCTION

Introduction

Passionné d'informatique depuis mon très jeune âge je me suis orienté dans cette branche pour que cette passion devienne mon métier. Arrivé au lycée j'ai commencé à découvrir tous les métiers et c'est suite à mon apprentissage autodidacte du C que j'ai décidé de devenir développeur à court terme puis chef de projet à long terme.

Pour réaliser ce projet j'ai donc décider de faire un DUT informatique à l'institut Universitaire et technologique de Lens. Celle-ci grâce à son enseignement m'a permis de développer multiples connaissances dans le développement d'application et de site web ainsi que dans la gestion de projet.

Après L'IUT dans le but d'approfondir mes connaissances j'ai décidé d'aller à la Faculté Jean-Perrin qui m'a permis de consolider et d'étendre les capacités obtenues durant mon précédent diplôme.

Pour valider mon semestre et continuer mon cursus, Norsys a accepté de m'accompagner durant 16 semaines sur le site de Ennevelin dans le rôle de développeur.

Norsys m'a séduit en me proposant des missions techniques me permettant d'étendre mes compétences en me faisant découvrir des choses nouvelles ainsi qu'une formation sur les technologies utilisées au sein de cette structure.

Norsys a su se démarquer grâce à son côté hors du temps et sa philosophie innovante en ce qui concerne la vie des collaborateurs au sein de sa structure ainsi que son engagement envers les clients.

Tout au long de ce rapport je vais vous présenter l'entreprise dans un premier temps puis mes missions et pour finir mon ressentis sur ce stage et ce qu'il m'a apporté d'un point de vue personnel et professionnel.

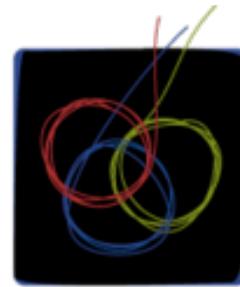
L'ENTREPRISE

L'Entreprise

Norsys est une société composée de cinq divisions :

- **Norsys Groupe :**

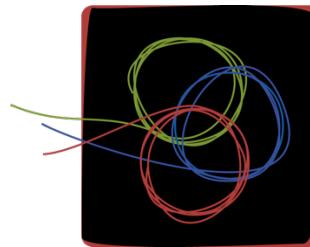
Regroupe l'ensemble des entités de Norsys qu'elles soient géographiques ou fonctionnelles.



norsys
GROUPE

- **Norsys EasyMakers :**

Rassemble l'ensemble des compétences et les méthodes que Norsys propose à ses clients.



norsys
EASYMAKERS

- **Norsys Université :**

Ensemble des formations permanentes misent en places dans le but d'avoir une démarche qualitative, de recherche et de développement.



norsys
FONDATION

- **Norsys Ways :**

Permet une progression des collaborateurs suivant deux axes : la progression et l'évolution.

- **Norsys Fondation :**

Facilite l'accès à la connaissance pour mieux vivre.

Norsys



Site de Ennevelin

Norsys est une société française d'informatique née en 1994 dont le siège social est à Ennevelin. Son PDG (Président Directeur Général) est Monsieur Sylvain Breuzzard , président de Green Peace qui mise sur une entreprise se préoccupant au maximum du bien-être de ses collaborateurs et de la vie en société.

Norsys se décompose en sept secteurs d'activité :

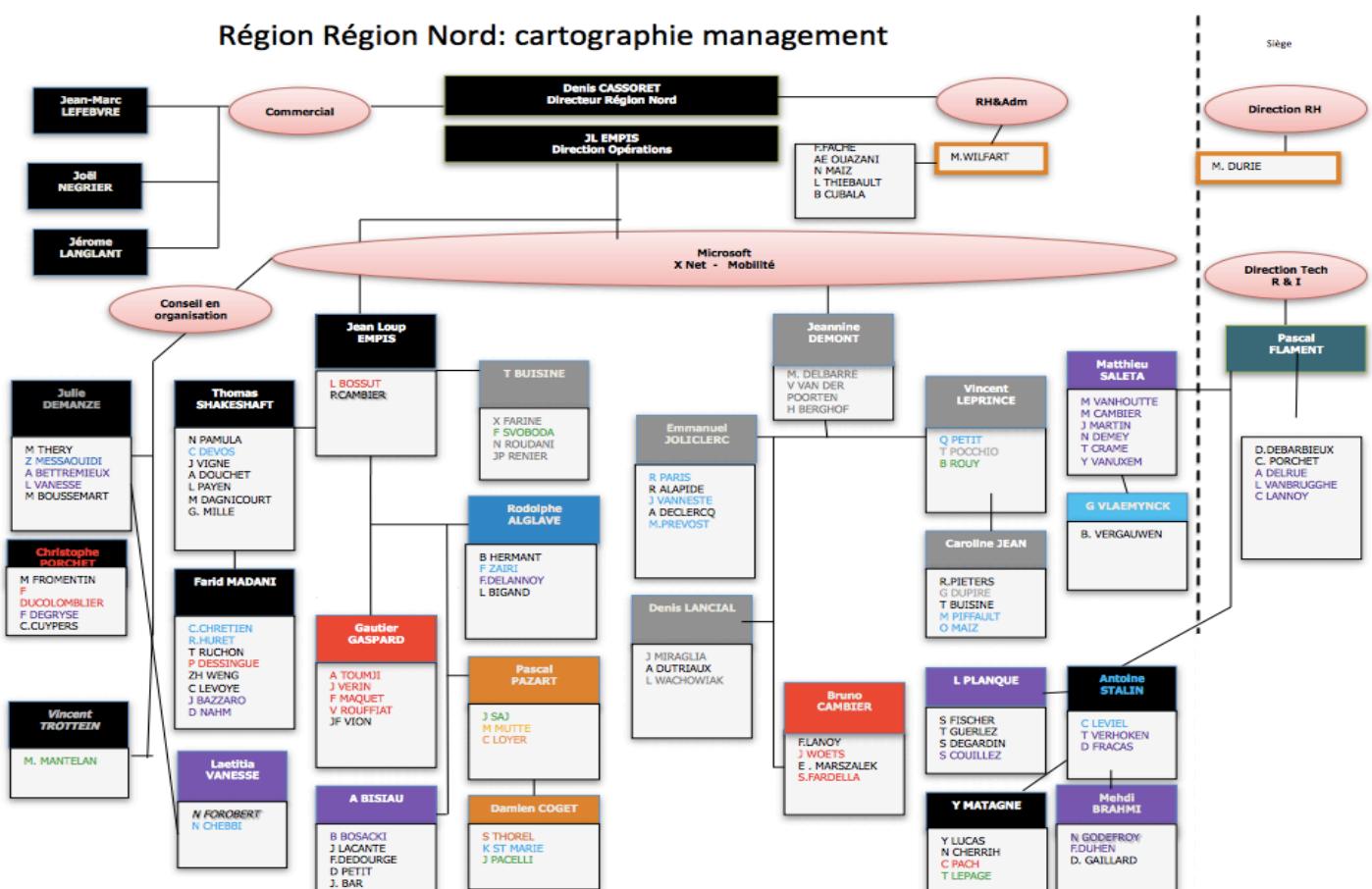
- Santé / Social
- Commerce
- Assurance / Banque
- Commerce
- Service
- Etat et collectivité
- Autres



ORGANIGRAMME

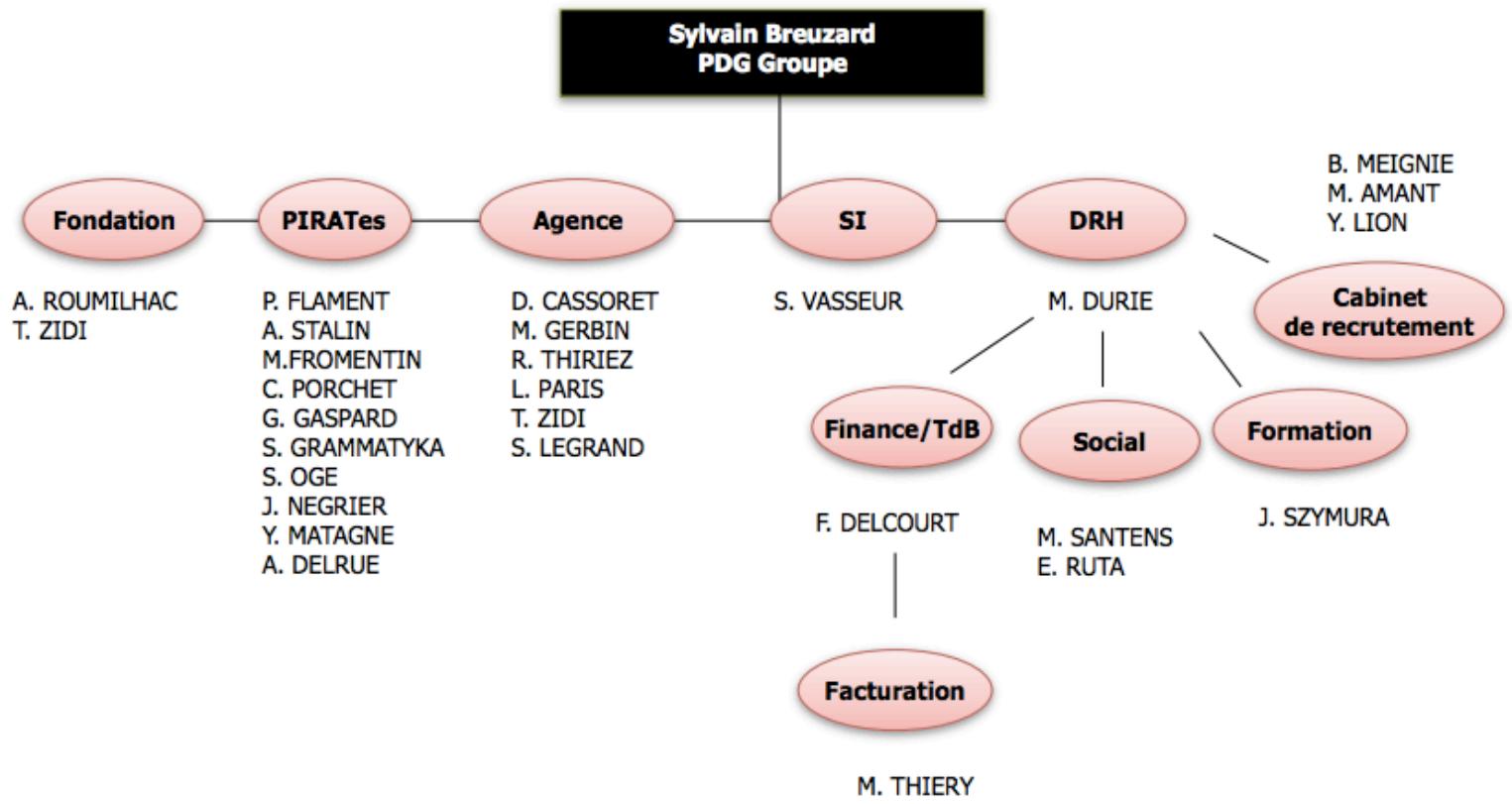
Organigramme

Région Région Nord: cartographie management



ORGANIGRAMME

Groupe norsys : cartographie management 2017



CONTACT

Contact

Siège :

1 rue de la Cense des Raines
59710 Ennevelin

Adresse du Stage :

1 rue de la Cense des Raines
59710 Ennevelin

Téléphone :

03.28.76.56.70

LES FORMATIONS

Les Formations

OBJECTIFS

Le But de la formation était de comprendre les méthodologies de Norsys ainsi que les technologies (Framework, langages etc....) que Norsys utilisent.

Pour cela avec 12 autres stagiaires nous avions pour but de faire une API REST permettant de suivre l'humeur et la motivation de chaque salarié pour pouvoir faire différentes actions au cas où celle-ci ne serait pas bonne.

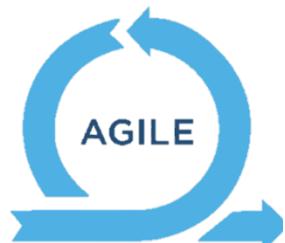
Cette application s'est vu ajouté une VDM (Vie de merde) pour la rendre plus attractive et ainsi permettre une attractivité sur celle-ci.

METHODE DE TRAVAIL

Les méthodes agiles reposent sur un cycle de développement itératif, incrémental et adaptatif. Elles doivent respecter quatre valeurs fondamentales déclinées en douze principes desquels découlent une base de pratiques, soit communes, soit complémentaires.

Les 4 valeurs sont :

- Individus et interactions plutôt que processus et outils (collaborations)
- Fonctionnalités opérationnelles plutôt que documentation exhaustive (pas de cahier des charges , etc...)
- Collaboration avec le client plutôt que contractualisation des relations (proximité)
- Acceptation du changement plutôt que conformité aux plans (s'adapter aux besoins)



LES BONNES PRATIQUES

INTELLIJ IDEA :

Lors de la formation, j'ai utilisé l'environnement de développement « intelliJ IDEA ». Celui-ci à de nombreux avantages. Il est compatible avec de multiples langages (comme Java, Scala, Groovy, Clojure, Javascript, ...), intègres des outils de versionning (Git, Svn, Mercurial) et des outils de compilation tels que Gradle, et Maven.

LES FORMATIONS

GIT :

Afin de « versionner » notre projet, l'outil GIT a été utilisé.

Pour plus de lisibilité sur les « commits » effectués par toute l'équipe et enfin de ne pas se perdre, le format « karma commit » a été utilisé. Ce format ressemble à ceci :

<Type> (<Scope>) : <Subject>

- <Type> : le type, par exemple documentation, fix, test, ...
- <Scope> : la portée, par exemple configuration, proxy, ...
- <Subject> : et enfin le pourquoi du « commit »

TECHNOLOGIES

Spring :

Spring est un Framework qui permet d'écrire du code clean et testable facilement. En effet celui-ci a la particularité de baser le développement sur les tests (Méthode TDD).

Il permet de définir l'infrastructure d'une application java facilement et est dit « léger » c'est à dire qu'il prend en charge lui-même la création des différents objets grâce aux fichiers de configuration.

Le gros avantage de ce Framework est que les classes n'ont pas besoin d'implémenter une quelconque interface pour être prises en charge par le Framework



Spring Boot :

Spring Boot est lui aussi un Framework basé sur Spring pour créer des applications Spring de qualité indépendante.

Il prend toutes les fonctionnalités de Spring et des bibliothèques tierces afin de pouvoir commencer avec un minimum d'agitation. La plupart des applications Spring Boot nécessitent très peu de configuration Spring.

Celui-ci intègre un serveur TOMCAT (implémente les servlets JEE) et utilise un système simple de configuration (pom.xml)



LES FORMATIONS

MES TACHES

- Création de la partie Service et de la partie Controller pour une création d'humeur.
- Création de la partie Service et de la partie Controller pour une création d'anecdote.
- Démonstration au client du travail réalisé lors du 1^{er} sprint
- Rôle de Pilote de projet pendant 1 journée

LA FORMATION MOBILITE

Android :

Android, est un système d'exploitation mobile basé sur le noyau Linux et développé par Google. Le système a d'abord été conçu pour les smartphones et tablettes tactiles, puis s'est diversifié dans les objets connectés et ordinateurs.

Une application Android a un certain cycle de vie (**Annexe 2 : Cycle de vie Android**) qui se compose en différentes phases.

De plus le langage utilisé par une application Android est le Java ce qui est facilement accessible lorsqu'on a suivi une licence 3.

iOS :

iOS, autrefois iPhone OS, est le système d'exploitation mobile développé par Apple pour plusieurs de ses appareils. Il est dérivé de macOS dont il partage les fondations.

Une application iOS a elle aussi un cycle de vie qui est proche de celui d'une application Android comme nous pouvons le constater (**Annexe 1 : Cycle de vie iOS**).

De plus le langage utilisé par une application iOS est le Swift qui coexiste avec l'objectif-c depuis 2014.

LES FORMATIONS

CONCLUSION

La formation a été une réelle opportunité pour moi de découvrir le fonctionnement, l'esprit et les technologies de Norsys.

Premièrement cela m'a permis de découvrir les différentes technologies que j'allais côtoyer pendant toute la durée de mon stage et dans mon avenir.

D'autre par cette formation m'a fait découvrir une nouvelle gestion de projet qui est une très bonne selon moi puisque cela permet d'avoir de la créativité et de l'indépendance.

Pour finir elle m'a permis de découvrir l'esprit de Norsys qui est très différent des autres entreprises puisqu'il n'y a pas la notion de hiérarchie ce qui permets une meilleur collaboration et une ambiance différente au sein de l'équipe.

AnimaBox

OBJECTIF

L'objectif de mission était de créer une application iOS MVP en 3 semaines pour qu'elle puisse être présentée à Marrakech et ensuite de l'améliorer après cette dead line ainsi que la développer sur Android.

Cette application a pour but d'aider les animateurs de réunion à réaliser de bonne réunion grâce à de petit outils.

Dans ces outils nous pouvons trouver des fiches méthodes, des timers programmables, un planning Poker (local et à distance), une médiatique de son, des Dés configurables et un système de cartons lors de réunion.

RESSOURCES

Xcode :

Xcode est un environnement de développement pour macOS, ainsi que pour iOS, watchOS et tvOS.
API de programmation :

- Carbon permet de programmer avec les langages suivants :
 - C
 - C++
- Cocoa permet de programmer avec les langages suivants :
 - Objective-C (version 2.0 à partir de Xcode 3.0)
 - AppleScript (à travers l'interface AppleScript Studio)
 - Java
 - ***Swift***

DÉROULEMENT

Afin de bien commencer le projet nous avons mis en place les méthodes de gestion de projet pour estimer les temps des différentes tâches au moyen d'un poker planning.

ANIMABOX

Après cette étape , nous avons pu mettre en place notre planning avec les ordres de priorités .C'est alors qu'avec l'équipe nous avons convenu des conventions et fait une bonne base du projet pour commencer le développement de l'application.

La mission se déroule très bien il y a une bonne entente et une bonne cohésion avec Elodie qui est la seule personne avec qui je travaille.

Niveau technique au vu de la nouveauté du langage pour moi le début fut difficile mais à l'heure actuelle je me sens de plus en plus alaise et de nouvelle technologie ont été mise en place un peu difficilement mais à la fin le résultat attendu est là.

LA PARTIE TECHNIQUE

Firebase :

Présentation :

Firebase est un ensemble de services d'hébergement pour n'importe quel type d'application (Android, iOS, Javascript, Node.js, Java, Unity, PHP, C++ ...). Il propose d'héberger en NoSQL et en temps réel des bases de données, du contenu, de l'authentification sociale (Google, Facebook, Twitter et Github), et des notifications, ou encore des services, tel que par exemple un serveur de communication temps réel.

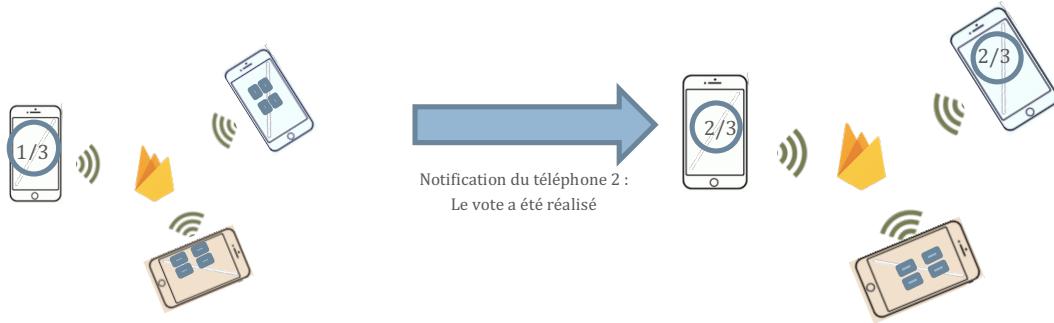
Objectif :

Firebase a été utilisé dans le but de faire un poker planning à distance. Pour réaliser cette partie nous avons créé des salons à la manière de *discord* dans lequel nous attendons que les personnes se connectent dessus. Lorsque tout le monde est présent l'administrateur du salon lance le vote.

Cet outil permet 2 fonctions très intéressantes. En effet il permet premièrement d'avoir un module d'authentification qui va permettre de se connecter à des salons.

Deuxièmement il permet d'avoir les changements et les informations du salons en directe par exemple un écran d'attente est réalisé et on peut voir le nombre de personne qui s'incrémentent au fur et à mesure que les personnes votent.

ANIMABOX



Mise en place :

Pour mettre en place Firebase deux étapes sont nécessaires :

Tout d'abord il y a la partie configuration et création de la base par le biais du site internet Firebase, grâce à cette configuration nous obtenons un fichiers "GoogleService-Info.plist" que l'on va mettre dans le projet.

Pour finir par le biais de cocoa et son fichier pod nous stipulons qu'on utilise Firebase (**Annexe 3 : Fichier pod**) puis installons les extensions avec un '*pod install*'. Maintenant que les extensions sont installées nous pouvons les utiliser dans le projet.

Utilisation :

- Pour l'authentification nous avons utilisé une authentification anonyme de Firebase qui permets d'avoir un UserID. Cet ID aura pour fonction de clé pour actualiser les informations de la personne. C'est aussi avec cette clé que l'on va vérifier et détecter si la personne qui est connectée sur le salon est l'administrateur de celui-ci.

Dans le code cette opération se fait de la même façon qu'en annexe (**Annexe 4 : Authentification Firebase**) c'est à dire une partie où l'on vérifie si l'User ID (identifiant unique) est présent sur l'appareil, ensuite on se connecte puis grâce à celui-ci nous pouvons vérifier si l'utilisateur est l'administrateur, si c'est lui qui a été éjecté etc

ANIMABOX

- Pour la partie synchronisation en direct des données dans un salon, le principe de Firebase est de créer un observateur sur le nœud souhaité et ensuite de spécifier le nom du fils de celui-ci.

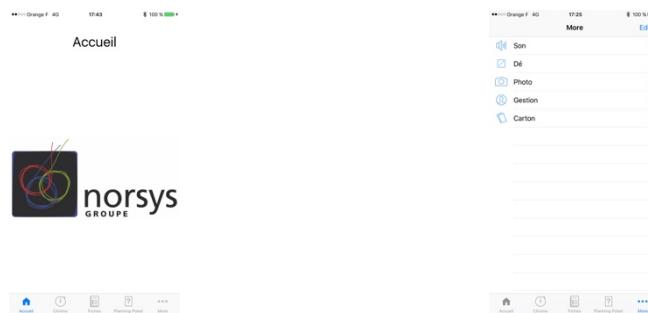
Par exemple si on a le type de données ci-dessous nous aurons un observateur sur le nœud dont le nom est Salon et ensuite préciser le nom du nœud fils souhaité par exemple state :



L'implémentation de ce principe est relativement simple dès lors qu'on a compris le principe de Firebase. Nous pouvons voir ce principe en annexe (**Annexe 5 : Creation d'un observateur**)

L'APPLICATION EN IMAGES

- L'application a comme navigation une navigation via TabBar qui est tendance dans les applications actuellement sur le marché (Facebook, Application Timer d'origine d'iOS ...)



- Une section Timer a été réalisé dans le but de créer des séquences durant toute la durée de l'animation et qui émet une sonnerie à chaque fin de séquences.

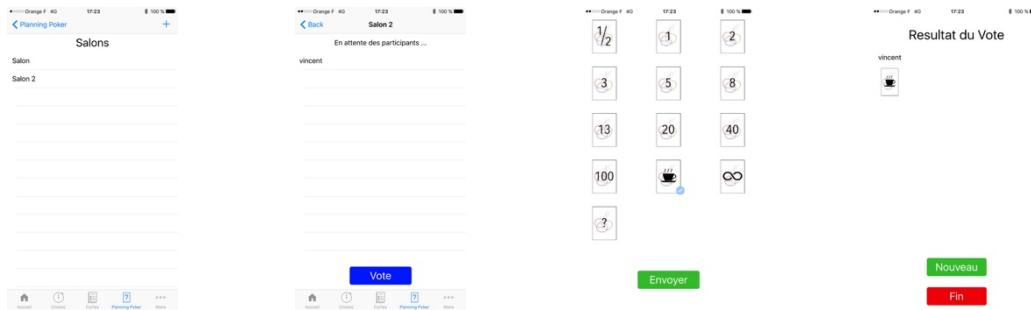
ANIMABOX



- Une section Fiches, une des demandes principales est implémentée dans l'application. Celle-ci permet la mise à disposition de fiches méthodes pour toutes les personnes devant organiser des réunions.

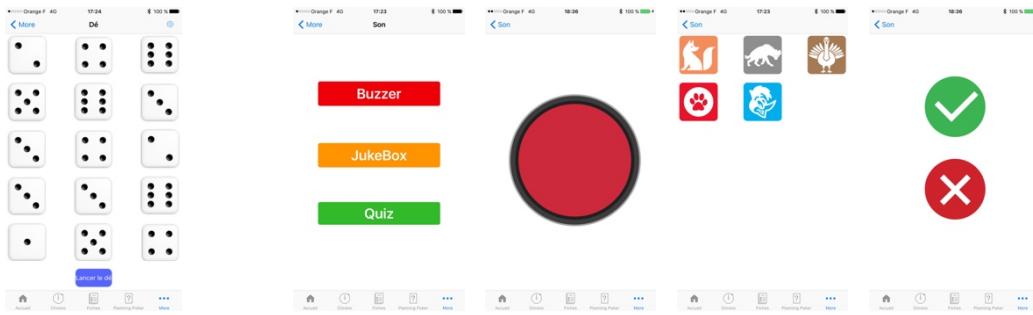


- Une section Planning Poker deuxième fonctionnalité la plus demandée a vu le jour. Celle-ci a été la partie la plus techniques à mettre en place pour l'instant. Celle-ci est fonctionnelle, répond exactement aux besoins et est très intuitive.



ANIMABOX

- En ce qui concerne le reste de l'application, de petits outils ont été mis en place tels qu'une bibliothèque de son, un buzzer, des sons de victoire ou défaite, des Dés qui peuvent être configurable en nombre et en nombre de face etc....



Conclusion

En Conclusion ce projet m'a permis de découvrir et de développer de nouvelles capacités.

Tout d'abord d'un point de vue développement, le fait de voir un nouveau langage et de nouvelles technologies m'a permis de développer et de découvrir une autre facette du travail de développeur.

Ensuite le fait de travailler sur ce projet m'a permis de voir de voir un peu le travail en équipe et la répartition des tâches dans une entreprise qui s'est très bien déroulé.

De plus le côté Analyse était présent durant toute la continuité du projet. Cette capacité est essentielle pour un développeur et un chef de projet.

Enfin le fait d'avoir carte blanche et de laisser place à ma personnalité est un très bon exercice pour la créativité qui est important et donnant une bonne valeur ajoutée.

Et pour finir le fait de voir que le travail de développeur propose des missions variées dans un autre registre me pousse à continuer dans cette branche.

CONCLUSION

Conclusion

La formation que m'a donnée l'université de L'Artois m'a été utile chaque jour chez Norsys.

Tout d'abord, les cours de programmation m'ont permis d'avoir de bons réflexes niveau code et de bonnes bases algorithmiques.

Ensuite les cours d'Anglais ont été utile tous les jours puisque toutes les documentations iOS, Firebase etc ... étaient en Anglais et avec mon souhait de progresser dans cette langue cela m'a été une bonne opportunité de l'exercer.

Enfin Norsys m'a permis d'avoir un stage très enrichissant dans le monde du travail. Les missions qui m'ont été attribuées m'ont apprise à approfondir les différents secteurs enseignés pendant ma formation.

En plus du point de vue technique, Norsys m'a permis de découvrir le monde professionnel d'une grosse entreprise dans une ambiance sympathique, conviviale avec une intégration rapide dans l'équipe et des « ateliers » techniques variés (Docker, JHipster, JUnit 5).

En conclusion ce stage a été une bonne expérience et je remercie Norsys qui m'a permis de développer mes compétences dans des secteurs multiples.

GLOSSAIRE

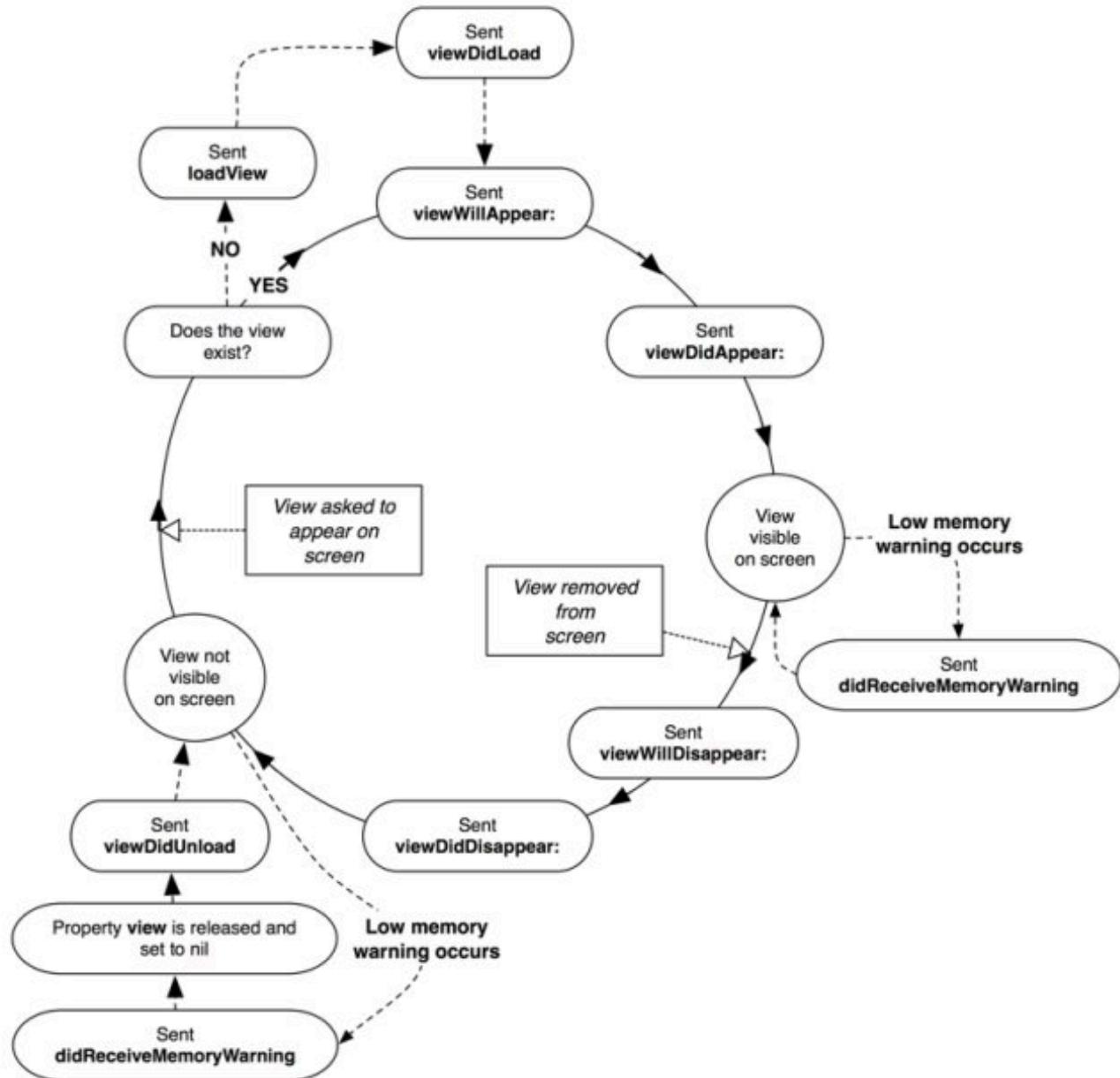
Glossaire

MVP : Une application mobile MVP correspond à une pratique d'innovation MVP spécifiquement appliquée aux domaines des applications mobiles. Il s'agit donc d'une application mobile pour laquelle on a fait le choix de « sacrifier » des fonctionnalités jugées comme non indispensables afin d'obtenir un lancement plus rapide.

ANNEXE

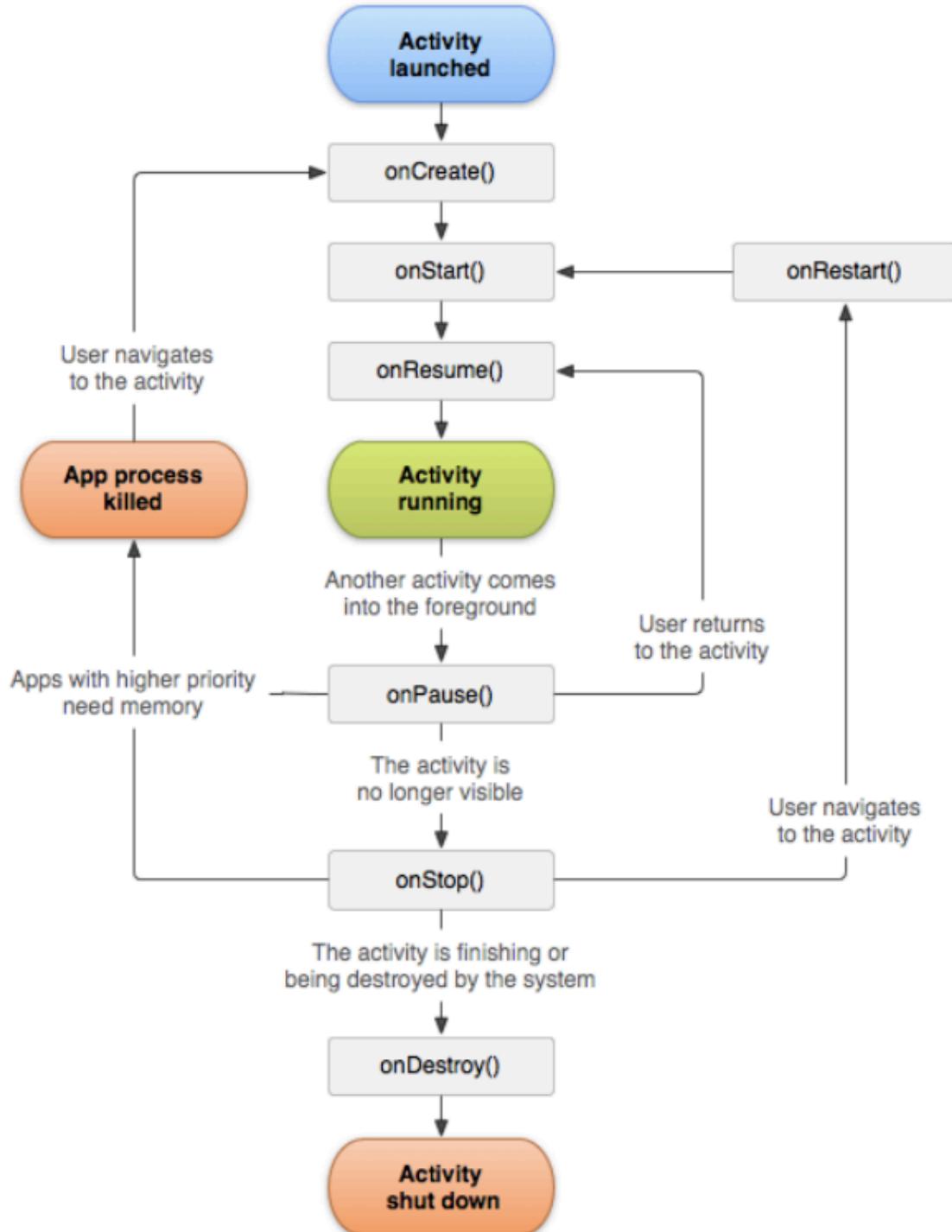
Annexe

Annexe 1 : Cycle de vie iOS



ANNEXE

Annexe 2 : Cycle de vie Android



ANNEXE

Annexe 3: Fichier pod

```
# Uncomment this line to define a global platform for your
# project
# platform :ios, '9.0'

target 'Animation' do
    # Comment this line if you're not using Swift and don't
    # want to use dynamic frameworks
    use_frameworks!

    # Pods for Animation
    pod 'Firebase/Database'
    pod 'Firebase/Core'
    pod 'Firebase/Auth'

    target 'AnimationTests' do
        inherit!: search_paths
        # Pods for testing
    end

    target 'AnimationUITests' do
        inherit!: search_paths
        # Pods for testing
    end
end
```

ANNEXE

Annexe 4: Authentification Firebase

```
#-----Creation du UserID-----
if FIRAuth.auth()?.currentUser == nil {
    FIRAuth.auth()?.signInAnonymously(){ (user, error) in
        if error != nil {
            .....
        }
    }
    else {
        self.performSegue(withIdentifier: segueId, sender: nil)
    }
}

#-----Connexion au Salon-----
if password == room.password{
    self.checkNameUser( nameUser : name, nameRoom : room.name,
completion: { (valid) in
    if valid{
        let user = User(name: name, uid: self.user.uid, roomName: room.name)
        let userRef = self.usersRef.child(user.uid)
        userRef.setValue(user.toAnyObject())
        self.performSegue(withIdentifier: "showJoinRoomSegue", sender: room)
    } else{
        self.showAlert(message:"Nom Utilisateur déjà utilisé")
        return
    }})
}
#-----Acces au UserID-----
if (user.uid == self.user.uid) {
    .....
}
```

ANNEXE

Annexe 5: Creation d'un observeur

#-----CREATION DE LA REFERENCE-----

```
roomRef = FIRDatabase.database().reference(withPath: "Rooms").child(room.name)
usersRef = FIRDatabase.database().reference(withPath: "Users")
```

#-----CREATION DE L'OBSERVEUR-----

```
valueVotersHandle = usersRef.queryOrdered(byChild: "roomName")
    .queryEqual(toValue: room.name)
    .observe(.value, with: { snapshot in

        self.nbMembers.text = String(snapshot.childrenCount)
        var nbVote = 0
        for userRef in snapshot.children{
            let user = User(snapshot: userRef as! FIRDataSnapshot)
            if user.voteSuccess {
                nbVote += 1
            }
        }
        nbVoters.text = String(nbVote)
        if self.nbVoters.text == self.nbMembers.text {
            self.performSegue(withIdentifier: "showResultVoteSegue", sender: self.user)
        }
    })
}
```

#-----SUPPRESSION DE L'OBSERVEUR-----

```
usersRef.removeObserver(withHandle: valueVotersHandle)
```

ANNEXE

