

**EQUIPE :**

L3 – 2017-2018 – Groupe 2

Anthony CADET - Kevin MARSAL - Grégory NIELLEZ - Jordan OLIVIER

**DATE :**

Dernière modification : 09/07/2018

|  |  |
| --- | --- |
|  | Journal du projet  ComÉdu |
|  |  |

Table des matières

[I. INITIALISATION du projet 3](#_Toc519434656)

[I.1 Organisation équipe 3](#_Toc519434657)

[I.2 L‘idée de départ 3](#_Toc519434658)

[I.3 Motifs (justifications de l'idée) 3](#_Toc519434659)

[I.4 Validation via veille concurrentielle 4](#_Toc519434660)

[I.5 Estimation de la complexité technique 4](#_Toc519434661)

[II.1 Description du projet 6](#_Toc519434662)

[II.2 Intégration 9](#_Toc519434663)

[II.3 Conditions de réussite 9](#_Toc519434664)

[II.4 Echéancier (macro-planning) 10](#_Toc519434665)

[II.5 Gestion des risques 10](#_Toc519434666)

[II.6 Organisation 11](#_Toc519434667)

[II.7 Représentation des contraintes techniques 11](#_Toc519434668)

[II.8 Description des environnements 13](#_Toc519434669)

[II.9 Les livrables du projet 13](#_Toc519434670)

[Document de déploiement du Projet 14](#_Toc519434671)

[III. PLANNING initial du projet 17](#_Toc519434672)

[III.1 Votre planning de référence 17](#_Toc519434673)

[III.2 Vos plans de tests (tests unitaires & tests de charge) 19](#_Toc519434674)

[III.3 Vos outils 19](#_Toc519434675)

[Votre outil de traçabilité 19](#_Toc519434676)

[Votre mécanisme de build 19](#_Toc519434677)

[Votre outil d'analyse et suivi qualité 19](#_Toc519434678)

[Votre outil de versionning 20](#_Toc519434679)

[IV. PILOTAGE du projet 21](#_Toc519434680)

[IV.1 Période concernée 21](#_Toc519434681)

[IV.2 Planning de référence vs. Planning de suivi, pour la période concernée 21](#_Toc519434682)

[IV.3 Analyse et explication des écarts 21](#_Toc519434683)

[IV.4 Trace des risques avérés et/ou des changements opérés 21](#_Toc519434684)

[IV.5 Métriques 22](#_Toc519434685)

[Sur l'utilisation de l'outil de versionning 22](#_Toc519434686)

[Sur l'utilisation de l'outil de d'analyse qualité 22](#_Toc519434687)

[Sur les résultats de vos tests 22](#_Toc519434688)

[Sur la tenue de charge 22](#_Toc519434689)

[Sur l'outil de traçabilité 22](#_Toc519434690)

[V. Bilan du projet 22](#_Toc519434691)

# I. INITIALISATION du projet

## I.1 Organisation équipe

Le projet ComÉdu est composé de 4 acteurs : Anthony CADET, Kevin MARSAL, Gregory NIELLEZ et Jordan OLIVIER. Nous sommes tous en L3 M2I à ITESCIA.

Le client de notre application est le personnel de l’administration de l’école ITESCIA et plus particulièrement les personnes chargées du recrutement.

ITESCIA est une école de la Chambre de Commerce et d’Industrie à prédominance informatique en apprentissage. Elle est le fruit de la fusion de l'ITIN et de l'ESCIA en 2013.

## I.2 L‘idée de départ

Tout est venu d’un simple constat : il est difficile pour les professeurs de recevoir des retours, aussi bien sur la qualité et que sur l’adaptation des cours en fonctions de la classe. C’est davantage le cas lorsque que l’audience est trop nombreuse. En effet, la plupart des étudiants n’osent ni poser de questions, ni intervenir durant les cours par peur d’être jugé. Cela peut provoquer un désintérêt des étudiants envers les cours.

Au départ, le projet ComÉdu souhaitait impacter la relation professeur - étudiants. Le but était d’avoir une interface à travers laquelle le professeur pourra poser des questions. Cela pouvait lui permettre de comprendre où en sont ses étudiants, de savoir s’ils ont compris ou s’ils ont des difficultés sur un sujet.

L’interface réservée aux étudiants devait permettre de garantir leur anonymat auprès des autres élèves tout en leurs permettant de répondre aux différentes questions et d’en poser. Ceci devait également moderniser les cours tout en augmentant l’intérêt de ces derniers.

Suite à la soutenance du 16 février 2018, ITESCIA a porté un intérêt tout particulier à notre projet ComÉdu afin de l’adapter à la modernisation des questionnaires des Journées Portes Ouvertes.

## I.3 Motifs (justifications de l'idée)

Notre client ITESCIA utilise toujours le format papier pour les questionnaires des Journées Portes Ouvertes, ce qui est en contradiction avec l’image dynamique qu’elle émane. Plus de 500 étudiants se rendent aux JPO réparties sur les trois sites de l’ITESCIA par an, d’où l’importance de ce projet.

Celui-ci a donc pour objectif de moderniser ce système tout en y ajoutant de nouvelles fonctionnalités. L’application permettra de récupérer les données des questionnaires de manières automatique et simplifiée.

La valeur ajoutée par rapport à la solution existante est que cette application permettra une rapidité de traitement pour le personnel pour faire des statistiques puisque tout est dématérialisé. Cette application mobile va promouvoir les capacités de ses étudiants et va également donner une idée aux futurs élèves de ce qu’il est possible de faire.

Enfin, cette application va permettre d’améliorer l’image de l’école, notamment en montrant le côté numérique et moderne de l’ITESCIA.

## I.4 Validation via veille concurrentielle

Notre projet fait face à plusieurs concurrents, dont voici les principaux :

**Klaxoon** est une solution en ligne permettant à un formateur de créer des activités pour animer ses formations, réunions et séminaires. Elle propose également des quizz, des sondages ou encore des défis collectifs. Cette solution est ergonomique, intuitive et dispose d’une interface épurée et claire.   
Les utilisateurs accèdent aux sessions en se connectant sur le site officiel.

Cependant, l’application est assez chère. Il faut compter 50€ par mois pour la version en Cloud acceptant 50 connexions simultanées ou 99€ pour la box qui accepte 20 connexions simultanées en local.   
Klaxoon est généralement confrontés à des problèmes de connectivité. Il lui est par ailleurs reproché d’avoir un manque de souplesse et de ne pas pouvoir intégrer de Powerpoint ou de Prezi.

**PEW (Poll EveryWhere)** est une solution de vote en live capable d’intégrer les résultats dans un Powerpoint en temps réel. Cependant, le modèle gratuit de l’application est assez limité et il ne propose pas, par exemple, de personnalisation. Une solution payante existe sous forme d’abonnement moyennant 65$ par mois apportant un meilleur support et plus de diversités.

Notre solution **ComÉdu** a l’avantage de répondre aux besoins très spécifique de l’ITESCIA, c’est-à-dire d’adapter le questionnaire Journées Portes Ouvertes au format numérique sur tablette.

Elle est intuitive et ergonomique et elle permet également de garantir l’anonymat des participants. En effet, à aucun moment l’application ne récolte les données personnelles. Tout ceci est fait en amont sur le site de l’ITESCIA. Les étudiants doivent s’inscrire afin de pouvoir participer au Journées Portes Ouvertes.

Enfin, cette solution est gratuite, offerte par les étudiants à l’école et elle pourra s’adapter pour d'autres besoins tels que les réunions des maîtres d’apprentissages.

## I.5 Estimation de la complexité technique

La complexité technique de ce projet n’est pas très importante. En effet, réaliser une interface permettant de remplir un formulaire ne comporte pas de difficultés particulières. Le défi de ce projet se porte sur la simplification des vues de l’utilisateur pour obtenir une application facile à prendre en main. Cela nous permettra de nous différencier de nos concurrents.

Le seul point sur lequel nous aurions effectivement du mal est la relation application – base de données.

Nous utiliserons principalement le langage de programmation JAVA pour réaliser notre application ComÉdu puisque nous développerons sous Android Studio. Le langage XML sera utilisé pour la modélisation des interfaces et le langage SQL utilisé pour la base de données.

Le chef de projet, Jordan OLIVIER, a déjà utilisé et pratiqué les trois langages de programmation cités précédemment. Il pourra donc nous transmettre son expérience et ses connaissances.

Nous comptons également sur l’intervention de Virginie SANS pour progresser sur le langage de programmation JAVA. Enfin, nous nous formerons sur les cours mis à disposition sur le site [openclassrooms](https://openclassrooms.com). Celui-ci nous permettra d’acquérir de l’expérience sur certains points spécifiques liés à notre projet, comme la relation entre base de données et l’application.

Concernant les technologies, nous utiliserons l’environnement de programmation Android Studio, l’outil de mécanisme de build associé à ce dernier qui est Gradle, l’outil de traçabilité Mantis Bug Tracker, l’outil d’analyse et suivi qualité SonarQube, Jenkins pour l’intégration continue, et enfin Github comme outil de versionning. De plus, notre base de données sera hébergée sur un serveur Ubuntu 16.04 virtualisé.

Fort de son expérience dans la programmation, Jordan OLIVIER a déjà utilisé à plusieurs reprises le logiciel Android Studio et l’outil de mécanisme de build Gradle. Nous suivrons ses conseils afin de l’utiliser correctement. De plus, nous ferons la demande auprès de l’école afin de bénéficier de cours d’Android. Concernant les autres outils, nous nous formerons sur internet. Il existe, en plus des guides d’utilisation sur les sites des développeurs, plusieurs tutoriels fait par des utilisateurs pouvant nous aider et nous suivront également des tutoriels d’installation et d’utilisation sur Youtube.

II. PRÉPARATION du projet

## II.1 Description du projet

Le projet est composé de plusieurs acteurs. En premier lieu, nous avons l’équipe projet constituée de 4 étudiants, qui sont Anthony CADET, Kevin MARSAL, Grégory NIELLEZ et Jordan OLIVIER.

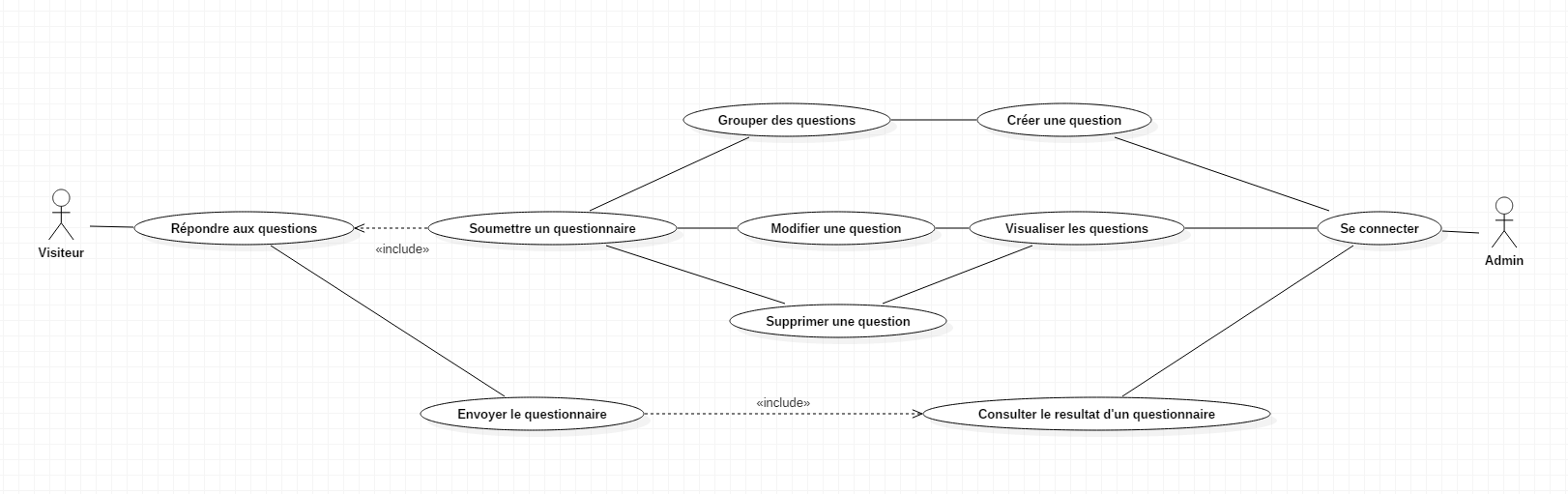
Puis, nous avons comme client l’école ITESCIA, et plus précisément l’administration et les personnes chargées du recrutement.

Enfin, les utilisateurs finaux seront aussi bien les étudiants participants aux JPO que les organisateurs d’évènement de l’ITESCIA.

L’application sera principalement utilisée de Décembre à Mars aussi bien par le client afin d’élaborer les questionnaires, de récolter et d’analyser les résultats que par les étudiants qui répondront aux questionnaires mis en place.

Le projet a pour but de livrer une application Android pour remplacer le formulaire papier des Journées Portes Ouvertes. Les étudiants présents lors des JPO pourront répondre aux questionnaires sur les tablettes mises à disposition lors de ces événements.

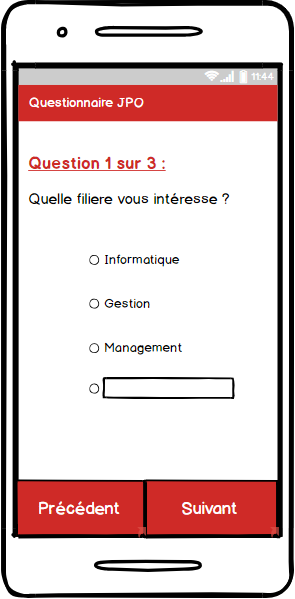
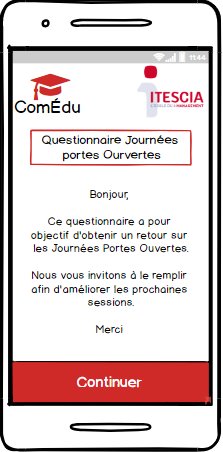
Pour mieux visualiser les différentes possibilités de l’application, nous avons réalisé un diagramme de cas d’utilisation :



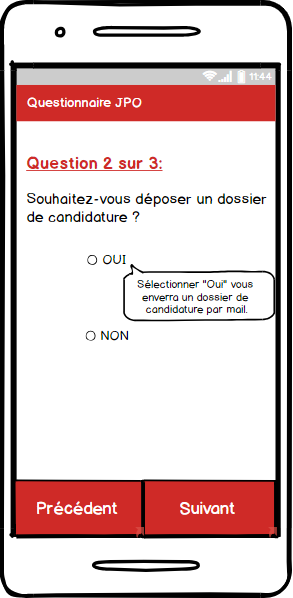
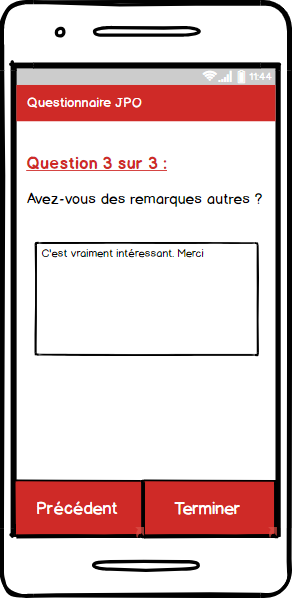
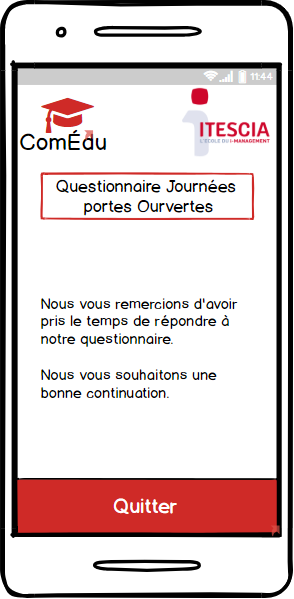
Nous avons également réalisé des maquettes en cours de UX design :

Scénario 1 : Un visiteur répond à un questionnaire JPO.

Visiteur Continuer Question 1

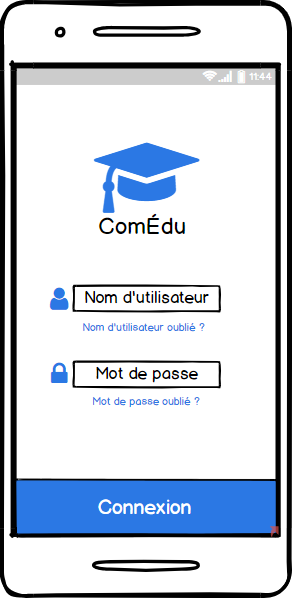
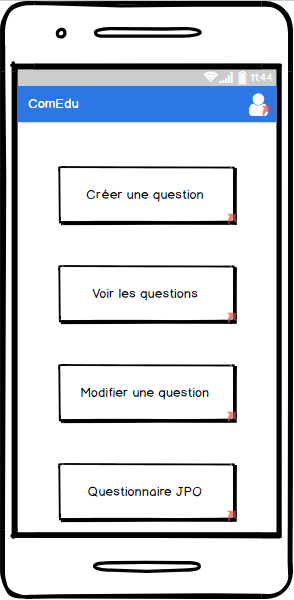


Question 2 Question 3 Remerciement

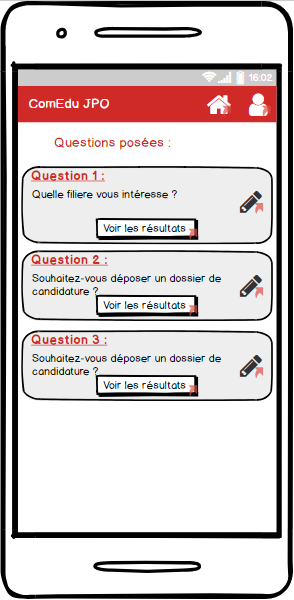
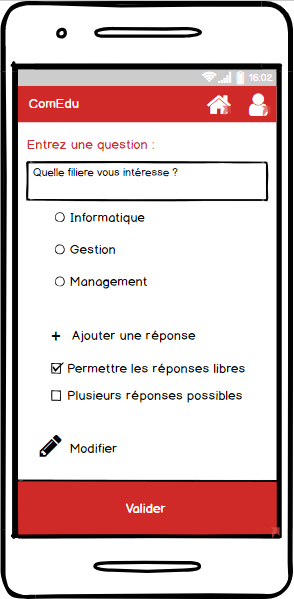
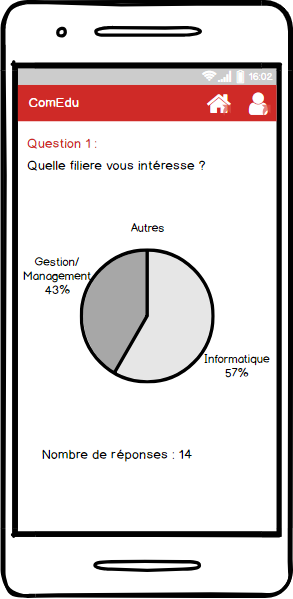


Scénario 2 : Création, modification d’un questionnaire JPO et visualisation des résultats.

 Se connecter Connexion Questionnaire JPO



Questions Modification Résultats



## II.2 Intégration

L’application devra se connecter à une base de données, présente sur serveur Ubuntu 16.04 virtualisé et hébergé sur les serveurs de l’ITESCIA, pour une gestion des identifiants. Elle sera indépendante à tout autre système d’informations et sera référencée par des informations entrées directement dans l’outil.

Étant donné que nous développons une application pour tablettes, nous avons choisi l’interface graphique comme type d’interface utilisateur. L'interface graphique désigne la manière dont est présenté un logiciel à l'écran pour l'utilisateur. Les [objets](https://fr.wikipedia.org/wiki/Objet_(informatique)) à manipuler sont présentés sous forme de pictogrammes sur lesquels l'usager peut imiter des manipulations physiques avec les doigts. C’est pour cette raison que notre interface graphique sera ergonomique et intuitive.

Au niveau de la confidentialité et de la sécurité, nous optons pour une sauvegarde des données (tel que les questionnaires) en local. De plus, les administrateurs et l’administration auront à leur disposition un identifiant et un mot de passe pour pouvoir modifier les questionnaires et visualiser les résultats.

## II.3 Conditions de réussite

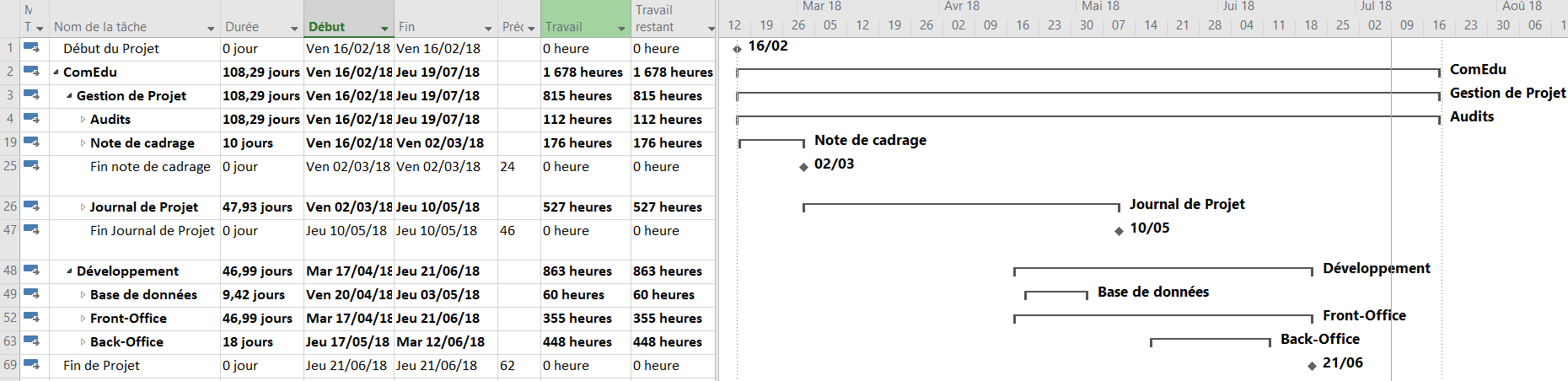
Afin de considérer le projet comme étant réussi nous devons, en priorité, développer une application Android qui permettra de répondre aux exigences de l'école. C’est à dire une application permettant d’une part, pour l’administration, de créer les questionnaires et visualiser les résultats et d’autre part, pour les visiteurs, de répondre aux questions et de donner leur avis sur les Journées Portes Ouvertes.

Nous avons envisagé d’exporter les résultats sous format Excel avec diagramme afin de faciliter le traitement de ces derniers par le personnel de l’administration. De plus, nous voulons également adapter l’application ComÉdu sous IOS.

Ensuite, on pourra l’adapter afin de revenir sur notre projet d’origine qui consistait à impacter la relation professeur-étudiants. Le but était d’avoir une interface à travers laquelle le professeur pourrait poser des questions afin de comprendre où en sont ses étudiants, de savoir s’ils ont compris ou s’ils ont des difficultés sur un sujet.

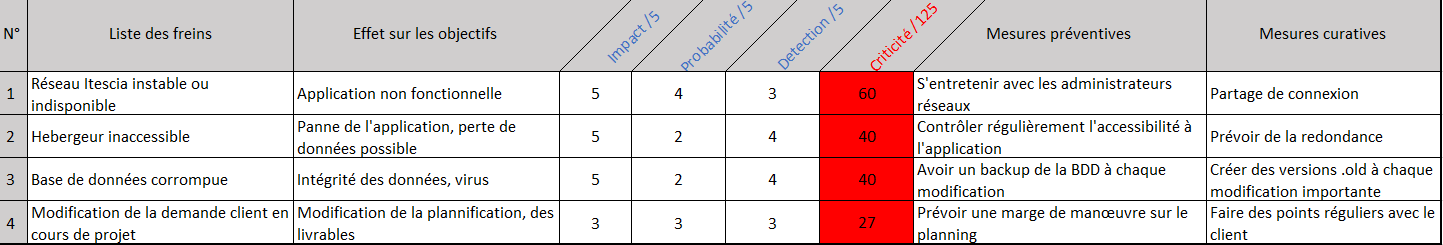
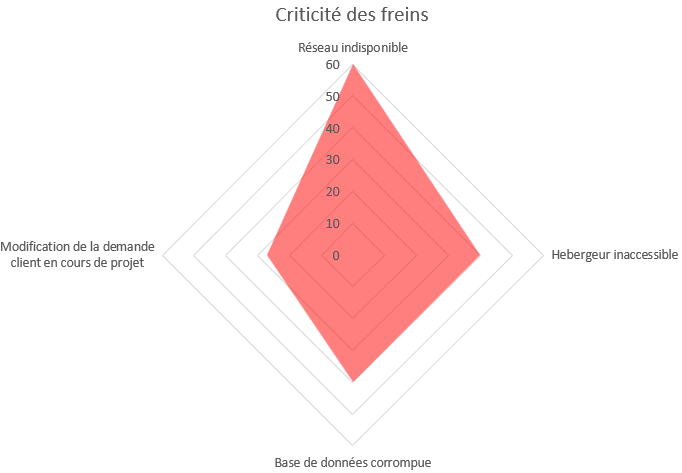
Enfin, nous avions pensé créer une interface web afin de permettre aux intervenants de mettre à disposition des cours et des exercices pour les élèves. Nous pensions exporter cette application dans d’autres écoles et pour d’autres évènements tels que les réunions des maîtres d'apprentissages et les séminaires.

## II.4 Echéancier (macro-planning)

L’échéancier est un planning assez général qui permet d’indiquer les grandes phases que nous envisageons d’organiser. Nous nous engageons donc à livrer l’application le 21/06/2018, ce qui nous laisse une marge de manœuvre.

## II.5 Gestion des risques

Voici dans le tableau ci-dessous, notre première analyse des risques afin de déterminer d’éventuelles mesures préventives avec :

* La liste des freins ;
* L’effet sur les objectifs ;
* L’impact par rapport aux objectifs : de 1 (impact faible) à 5 (impact très fort) ;
* La probabilité d’apparition du risque : de 1 (probabilité faible) à 5 (probabilité forte) ;
* La difficulté de détection : de 1 (peu de difficulté) à 5 (forte difficulté) ;
* La criticité du risque sur 125 (impact \* probabilité \* détection) ;
* Les mesures préventives ;
* Et les mesures créatives ;

Nous comptons vérifier qu’aucun des risques identifiés n’est venu freiner régulièrement notre projet. Par ailleurs nous rechercherons d’éventuels nouveaux risques pouvant survenir selon l’évolution du projet.

Afin de retracer un risque avéré, il nous faudra évaluer les effets causés au projet afin d’en trouver la raison et donc de trouver et d’appliquer une mesure créative pour y remédier. Ainsi, les risques seront identifiés par une personne afin d’en assurer le suivi et ils seront associés à de possibles mesures curatives avec une date d’implémentation de l’action retenue.

## II.6 Organisation

Concernant l’organisation de l’équipe, nous nous sommes répartis les rôles selon nos capacités, nos préférences et nos compétences.

Nous nous sommes mis d’accord sur le fait que Jordan Olivier soit le chef de projet puisqu’il a déjà mené un projet et c’est aussi un bon communicant.

Grégory Niellez est assistant chef de projet puisqu’il apprécie la gestion de projet et exerce ce métier dans son entreprise en alternance.

Enfin, Anthony Cadet et Kevin Marsal sont les développeurs de l’application puisqu’ils veulent améliorer leurs compétences.

Pour la gestion de la documentation nous avons utilisé « **Google Drive** ».

Google Drive est un cloud, cela permet à tous les membre de l’équipe d’avoir un historique des activités sur les fichiers. Il est possible de mettre des commentaires sur un point spécifique du document ou via la messagerie instantanée.

Une autre fonctionnalité de Google Drive est de pouvoir effectuer plusieurs modifications sur un fichier simultanément. En effet, Google Drive utilise les technologies de Google Docs, Google Sheets, et Google Slides, ce qui permet d’effectuer les modifications en ligne.

## II.7 Représentation des contraintes techniques

Nous avons réalisé un diagramme de déploiement :

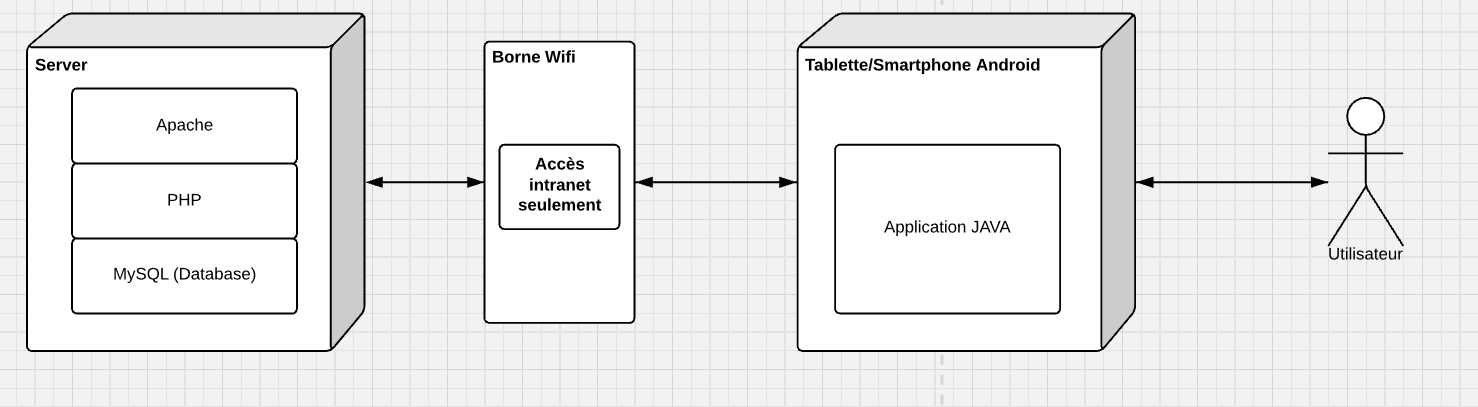
**

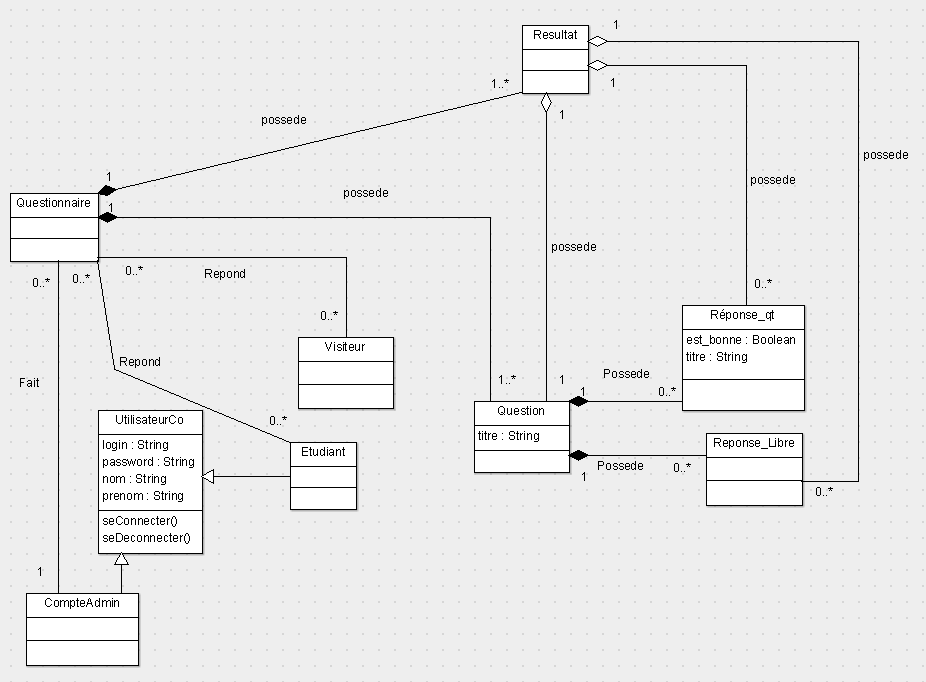
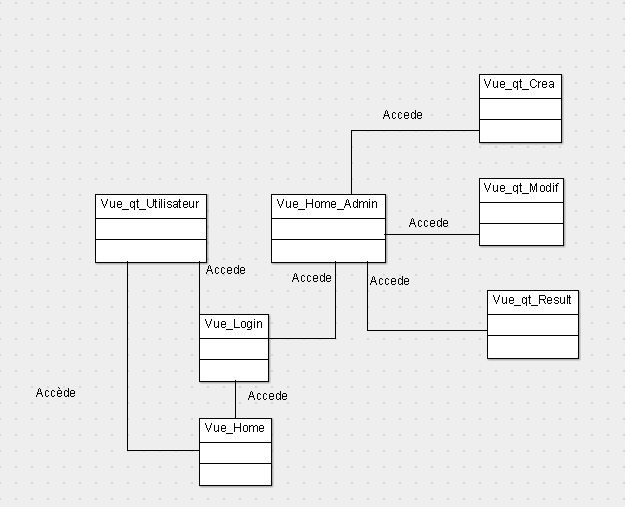
Diagramme de classes contrôleur :

Diagramme de classes des vues :

**

## II.8 Description des environnements

Pour notre application Android ComÉdu, nous avons choisi d’utiliser Android Studio 3.1.3 comme environnement de développement. Celui-ci intègre un émulateur Android et est relié avec notre Github https://github.com/JordanOlivier78700/Comedu. Les quatre membre du projet ont ainsi installé Android Studio sur leur ordinateur personnel ce qui nous permet de développer sur le même environnement partagé.

Tout d’abord, nous utilisons l’émulateur d’Android Studio comme environnement de test. Celui-ci permet d’exécuter l’application rapidement juste après avoir fait des modifications sur le code. Ensuite, pour l’environnements de préproduction, nous allons utiliser nos tablettes Android. Enfin, pour l’environnement de production, ITESCIA nous mettra à disposition des tablettes qui serviront pour les JPO.

La cible de production de notre application ComÉdu est le système d’exploitation Android.

## II.9 Les livrables du projet

*Au final vous fournissez quoi, un zip, un Jar, un war, un dossier …*

*Des notices ou documents techniques, …*

* *Un document de 5/10 pages maximum sur les contraintes techniques associés à votre projet*
  + *J’utiliser quel langage, plateforme, os … ?*
  + *Il faut une base de données, laquelle, quelle version, les utilisateurs associées ?*
  + *Il faut un serveur web, lequel, quelle version, quel paramétrage ?*
  + *Contraintes de licences associées (indiquez les licences associées à chaque élément du projet)*
  + *…*

Au final, nous allons livrer :

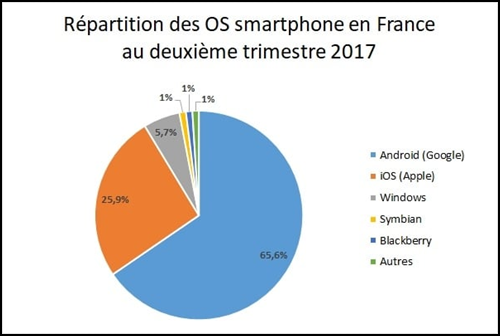
* Une application Android en .APK.
* Les documents d’installations des outils.
* Une procédure d’utilisation pour les différents types d’utilisateurs.
* Document de déploiement du projet (langages, où est le code, où est livrable, droits sur fichiers).
* Document de contraintes techniques (infos BDD, serveur web, licences, langages).

## Document de déploiement du Projet

I. Langages et la plateforme utilisés

Pour réaliser notre projet plusieurs possibilités s’offraient à nous : développer une application WEB ou bien une application mobile. Nous avons opté pour une application mobile pour que l’application soit disponible par un plus grand nombre de personnes même en cas de coupure internet. Une application est souvent plus simple à comprendre et à utiliser qu’un site web.

D’après le diagramme de répartition des systèmes d’exploitation mobiles ci-dessous, Android représente à lui seul plus de 65 % des utilisateurs. C’est pourquoi nous avons choisi ce système d’exploitation.



Nous sommes conscients de passer à côté de plus de 30% des utilisateurs mais notre choix ne sera pas impactant car l’application sera mise à disposition par l’école, qui utilise des tablettes Android.

Afin de développer notre application Android, nous allons devoir utiliser les langages suivants :

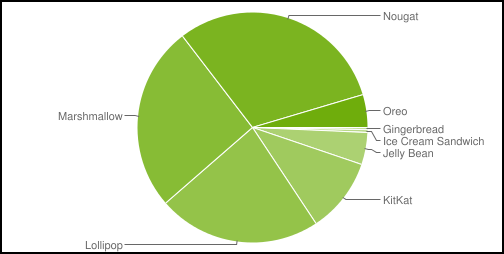
* Java pour le code qui va faire fonctionner l’application ;
* XML pour créer les visuels de l’application ;
* SQL pour créer la base de données.

La partie la plus importante de l’application Android est développée en Java. Les fichiers XML sont la partie visible de l’application, la partie qui affiche les différents boutons, champs, etc…

En ce qui concerne la base de données de l’application, nous avons besoin d’utiliser les requêtes SQL qui vont nous permettre de stocker et utiliser des données.

Une fois le choix du support fait nous devions choisir la version de l’API sur laquelle nous allions travailler.

D’après le ce diagramme ci-dessous montrant la répartition des API Android en France en Avril 2018, Kitkat a une part de marché à 12 %, Lollipop représente 22%, Marshmallow est présent sur 26 % des terminaux Android, Nougat est à 30 % et enfin le plus récent, Oreo, représente 5%.



Notre choix s’est donc porté sur l’API KitKat 4.4 sortie en octobre 2013, puisqu’il est compatible avec les versions ultérieures et c’est un compromis entre les fonctionnalités mobiles, l’ergonomie et le nombre d’utilisateurs.

II. Emplacement du code

Cette application est développée par les quatre membre du Projet. Pour que tout le monde puisse avoir accès à l’application et puisse travailler en même temps sur le projet, nous utilisons GitHub.

Nous utilisons chacun une branche du projet ce qui permet de pouvoir développer la partie du projet qui nous est assignée sans pour autant gêner les autres si une partie du code développé pose problème.

GitHub nous permet d’introduire les fonctionnalités au fur et à mesure du développement en ajoutant le code qui est développé sur notre branche directement sur la branche commune, dite “master”. Les fichiers de l’application ajouté au projet sur GitHub sont accessibles à tout le monde à l’adresse : <https://github.com/JordanOlivier78700/Comedu>

III. Où se trouvent les livrables ?

Les livrables se trouvent à plusieurs endroits.

En ce qui concerne les documents comme le planning de référence et de suivi, le journal de projet, les différentes procédures, documents techniques et autres, ils se trouvent sur notre Google Drive : <https://drive.google.com/drive/u/0/folders/10i8mXkiVUS6g3fsF7PA8r9kJcXi7KP2C>

L’application sera livrée en .APK. Cet APK sera généré par Android Studio à partir du code se trouvant sur GitHub. Il se trouvera également sur le Google Drive une fois le projet terminé.

Pour pouvoir avoir accès au Drive il faut demander l’accès à un des membres de l’équipe ComEdu.

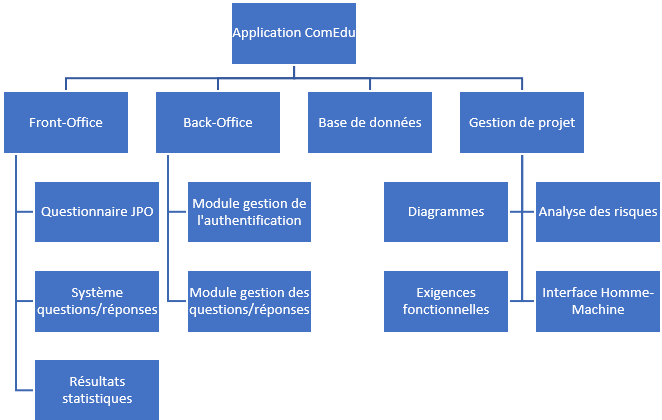
Suite aux problèmes que nous avons rencontrés avec le réseau ITESCIA ainsi que les problèmes lors du développement de l’application, les outils mis en place se trouvent sur les postes des membres de l’équipe ComEdu (Base de données, SonarQube, …).

# III. PLANNING initial du projet

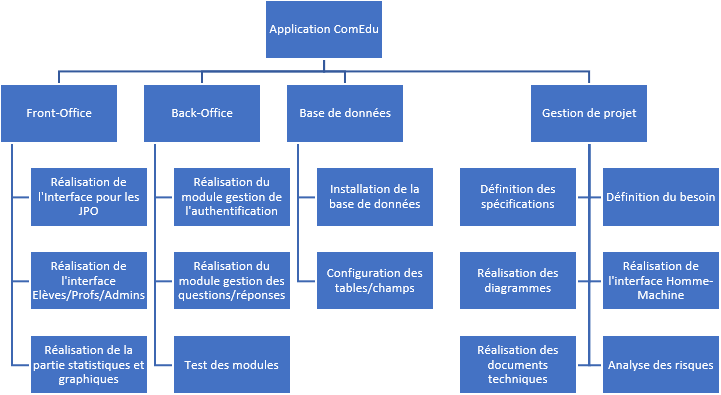
## III.1 Votre planning de référence

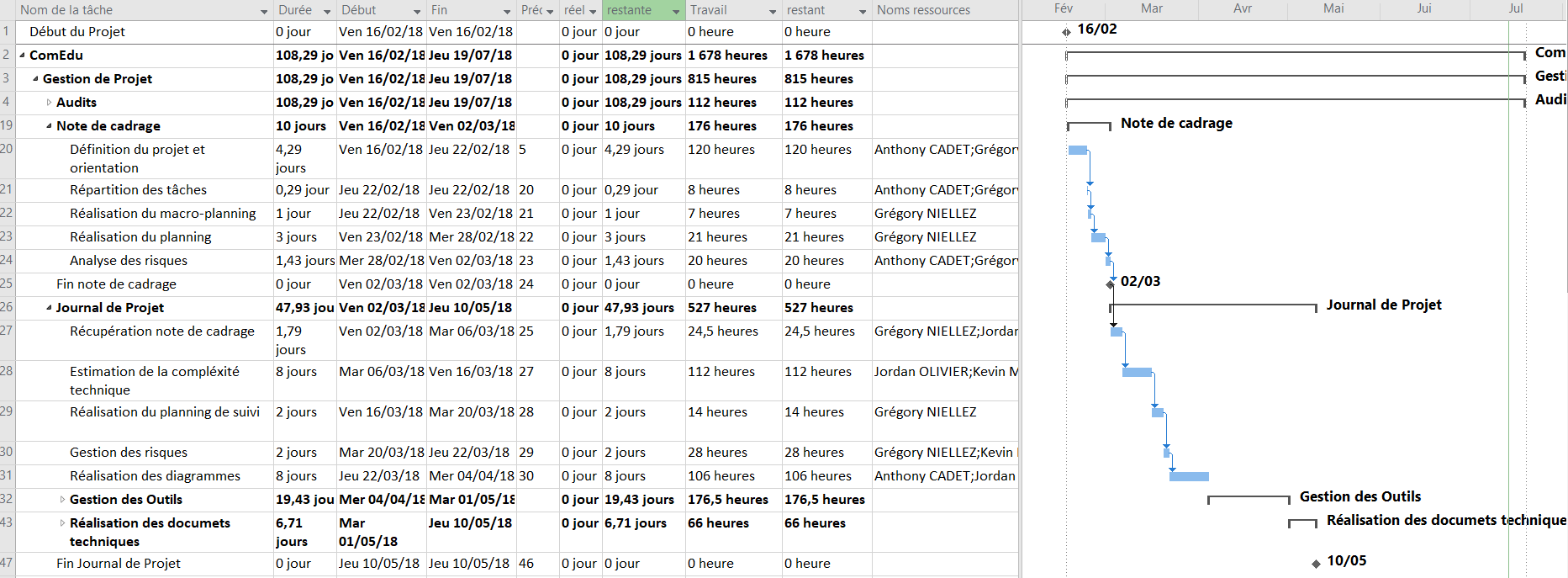
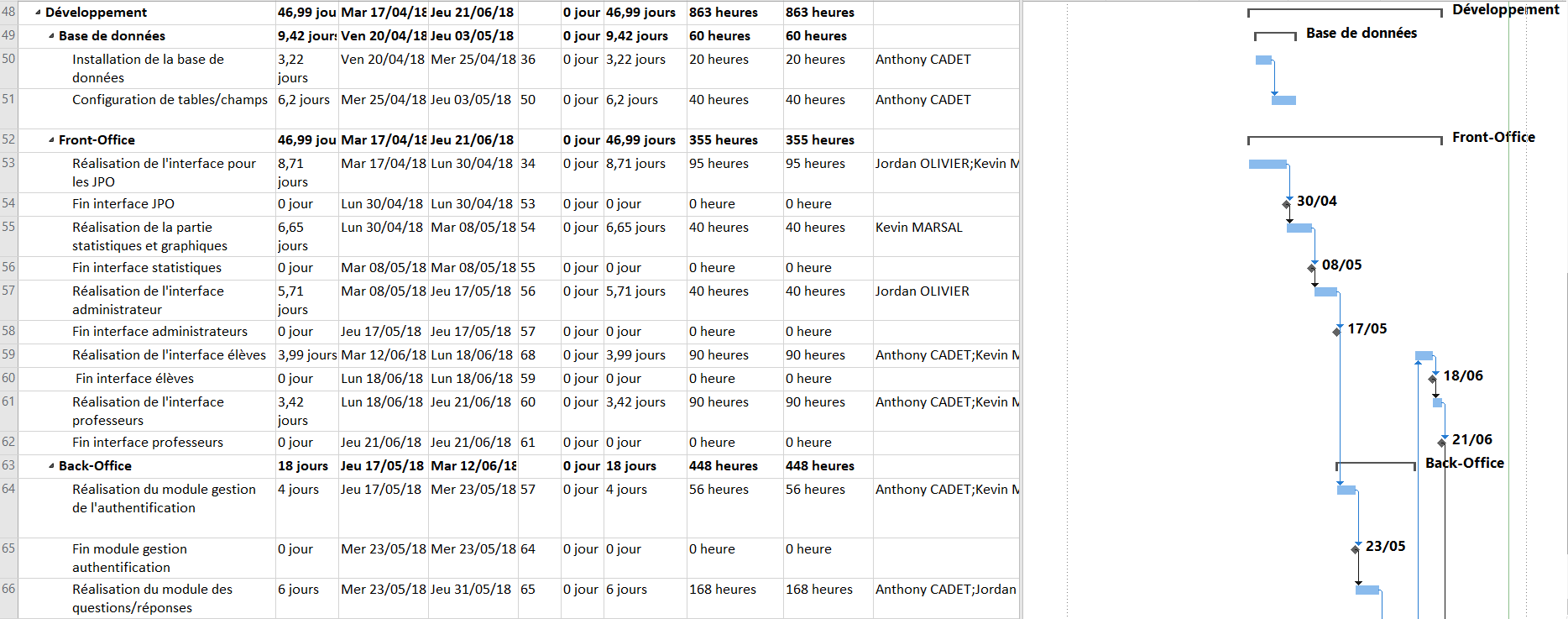
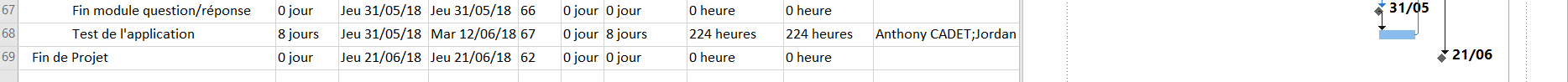
Afin de réaliser le planning de référence, nous avons :

* Réalisé le **PBS** (Product Breakdown Structure) qui est la décomposition du produit à livrer en composants, pouvant à leur tour être décomposés.



* Réalisé le **WBS** (Work Breakdown Structure) qui est la décomposition du projet en phases, étapes, lots, tâches, etc.



Nous avons choisi d’utiliser Microsoft Project pour réaliser notre planning de référence puisqu’il permet d’indiquer les durées et charges de travail.

## III.2 Vos plans de tests (tests unitaires & tests de charge)

*Comment, techniquement allez-vous mettre en place vos tests*

*PTV : Exprimé sous forme de phrases courtes (petits scénarios), quels sont les tests que doit passer votre application pour être considérée comme valide / utilisable*

*Quels sont vos framework pour tests Unitaire : JUnit, PHPUnit, QUnit …*

Les tests seront mis en place directement à l’aide de JUnit 5.

Nous considérons l’application viable à partir du moment où :

* Un utilisateur A peut réaliser un questionnaire et le sauvegarder.
* Un utilisateur B peut répondre à un questionnaire.
* Un utilisateur A peut consulter les résultats d’un questionnaire.

*Quels sont vos outils pour tester la charge (comportement de votre application quand elle est utilisée par 500 personnes, voir plus, en même temps) : JMeter, Gathling, Selenium, Fitness …*

Au départ, nous voulions utiliser le logiciel libre JMeter pour tester la charge de notre application ComÉdu. Cependant, étant donné que le nombre de tablettes prêtées par ITESCIA est de cinq pour les Journées Portes Ouvertes, nous avons décidé de ne pas faire de test de charge. C’est également ce que l’on nous a conseillé lors du premier audit technique.

## III.3 Vos outils

### Votre outil de traçabilité

Nous avons installé **Mantis Bug Tracker 2.13.1** sur notre serveur virtualisé Ubuntu 16.04. C’est un outil open source de suivi d’anomalies logicielles basé sur une interface web. Il est écrit en PHP et requiert une base de données (MySQL) et un serveur web (Apache).

Son principe de base est simple : il consiste à enregistrer la déclaration d'un bug informatique via un système de tickets, puis les techniciens de maintenance informatique mettent à jour l'avancement de sa résolution, jusqu'à la clôture du ticket en question.

Grégory NIELLEZ est responsable de l’outil.

### Votre mécanisme de build

Etant donné que nous utilisons Android Studio pour le développement de l’application, nous utiliserons le mécanisme de build qui est intégré à ce dernier, c’est-à-dire **Gradle 4.4**. C’est le digne successeur de Maven et de Ant, alliant ces deux outils afin de créer une plateforme de production Java simple à utiliser, et bien adaptée pour les projets Android.

En complément, nous utilisons l’outil d’intégration continu **Jenkins 2.121.1** afin d’automatiser les builds.

### Votre outil d'analyse et suivi qualité

Nous utilisons le logiciel open source SonarQube qui permet d’analyser et de mesurer la qualité du code source de façon continue d'un projet informatique. Il peut analyser plus de vingt-cinq langages tels que JAVA, C#, PHP et tant d’autres.

En dehors de cela, des commentaires seront mis en place régulièrement pour faciliter la compréhension du code. L’analyse avec SonarQube peut être entièrement automatisée avec l’outil d’intégration continue Jenkins.

Kevin MARSAL est responsable de l’outil et a installé **SonarQube Server 7.2** 2 et **Sonar Scanner 2.6.1** sur son ordinateur personnel Windows 10.

### Votre outil de versionning

Nous avons choisi l’outil GitHub afin de gérer le versionning de notre application. De plus, celui-ci dispose d’une interface graphique sous Windows **Github Deskstop 1.2.6** et est relié avec Android Studio. Les versions sont gérées par un système de branches : il y a une branche master et une branche par membres.

Jordan Olivier, le chef de projet, en a la charge et il vérifiera les merges avant de brancher et de fusionner (merger) les branches vers le master. Le fait de centraliser les merges permet d’éviter les problèmes de conflits.

# IV. PILOTAGE du projet

## IV.1 Période concernée

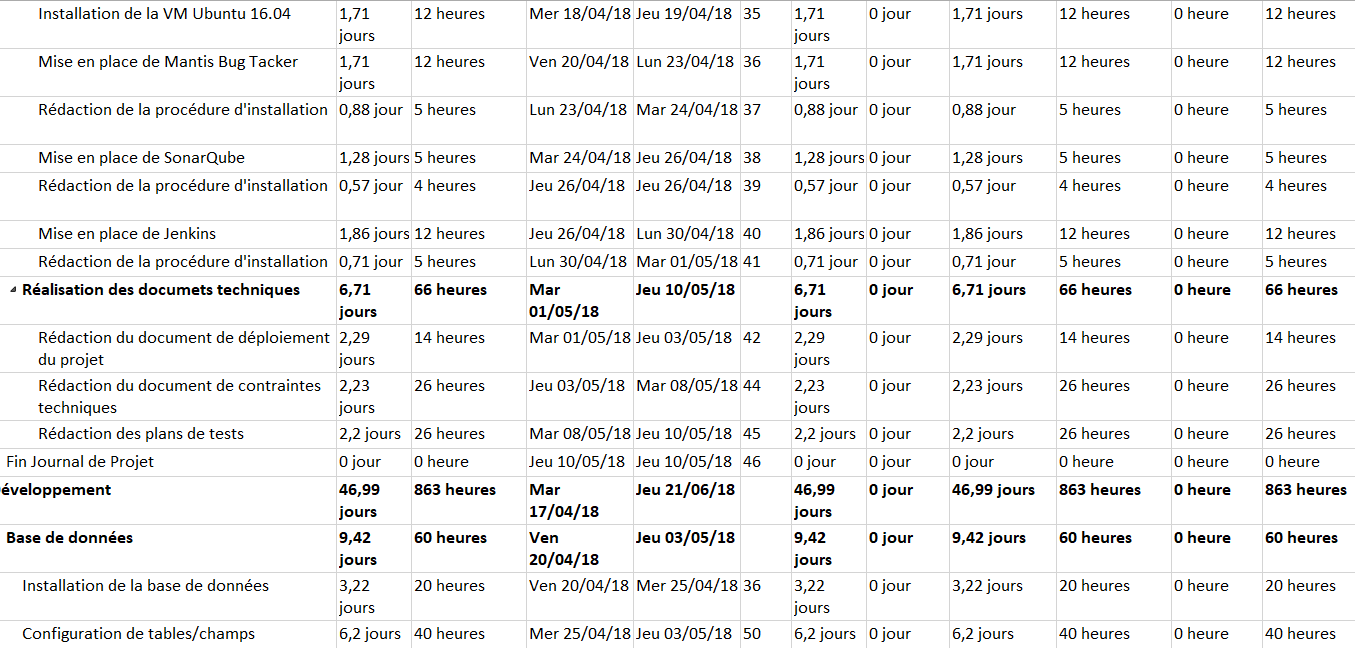
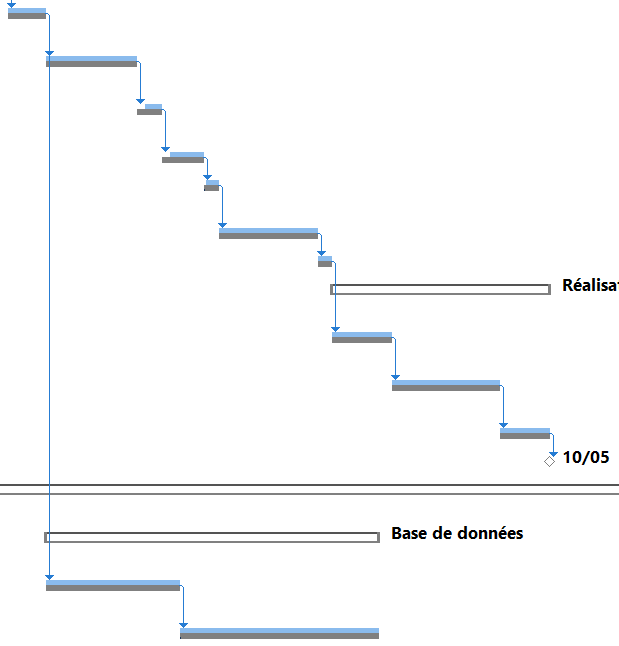
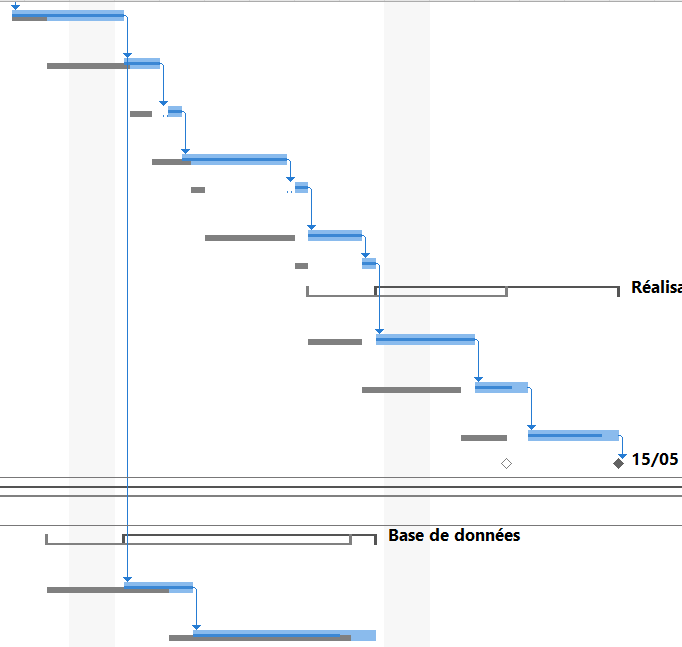
La période concernée commence à l’installation de la machine virtuelle Ubuntu 16.04, le mercredi 18 Avril 2018 jusqu’à la configuration des tables et des champs dans la base de données, le jeudi 03 Mai 2018.

## IV.2 Planning de référence vs. Planning de suivi, pour la période concernée

Pour définir un planning de référence sous Microsoft Project, il faut tout d’abord ajouter les durées et les charges de travail de référence, réelles et restantes.

Ensuite, il faut se rendre dans « Projet » puis cliquer sur « Définir le planning de référence ». Ainsi, la durée et le travail de référence prendront les mêmes valeurs que la durée et le travail.

Planning de suivi Planning de référence



## IV.3 Analyse et explication des écarts

Sur la première partie de la période, allant de l’installation de la machine virtuelle le 18 Avril à la fin du journal projet le jeudi 10 Mai, nous pouvons constater un écart de 5 jours en comptant le week-end par rapport au planning de référence.

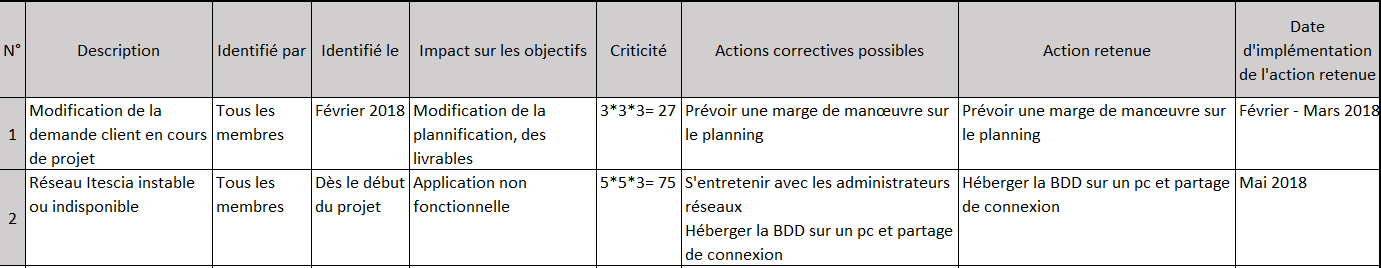
En effet, sur le planning de suivi, cette période se termine le mardi 15 Mai. Ceci s’explique tout d’abord par un problème d’installation de notre machine virtuelle Ubuntu qui a pris 20 heures au lieu de 12. Ensuite, nous avons rencontré des difficultés quant à la mise en place et à la configuration de l’outil d’analyse et de suivi qualité SonarQube et de l’outil d’intégration continue Jenkins qui ont pris 15 heures chacun au lieu de respectivement 5 et 12 heures.

Les problèmes d’installation de la machine virtuelle ont aussi impacté la seconde partie de la période, débutant à l’installation de la base de données le lundi 23 Avril à la place du vendredi 20 et la configuration des champs ce qui fait terminer cette seconde période le vendredi 4 Mai, initialement prévu le jeudi 3 Mai.

Enfin, le retard sur toute la période s’explique également par notre sous-estimation de temps de travail alloué pendant la période entreprise.

Ainsi, afin de limiter au maximum les retards, nous avons décidé de réduire le temps de travail de certaines taches dès lors qu’il était possible. Nous avons donc mis 22 heures pour réaliser les taches de rédaction du document de contraintes techniques et de la rédaction des plans de test chacune. Nous avons également réduit le temps de travail alloué à la configuration des tables et champs de la base de données, passant de 40 à 30 heures.

## IV.4 Trace des risques avérés et/ou des changements opérés

Voici le journal des risques :

Pendant la réalisation de notre projet, nous avons rencontré seulement deux risques : la modification de la demande client au début du projet et le réseau ITESCIA instable ou indisponible.

En effet, au départ le projet ComÉdu devait impacter la relation professeur – étudiants sous forme de questions réponses. Cependant, le client ITESCIA a vu une opportunité d’améliorer le fonctionnement du questionnaire JPO. C’est pour cette raison que nous avons changé de direction quelque temps après la présentation du projet du 16 Février 2017.

Pour le dernier risque, nous avons trouvé une solution temporaire en attendant la résolution des problèmes réseaux de l’ITESCIA. En effet, nous avons attendu le mois de Mai pour nous décider d’héberger la base de données sur notre ordinateur personnel et d’utiliser le partage de connexion.

## IV.5 Métriques

### Sur l'utilisation de l'outil de versionning

Nous avons la branche master, puis nous avons créé une branche par membres et une branche spécifique pour les interfaces, ce qui en fait six. Cependant, vers la fin du projet, nous avons supprimé les branches des membres afin de réduire le nombre de versions pour garder l’essentiel et pour gagner en visibilité.

*commit/utilisateur*

### Sur l'utilisation de l'outil de d'analyse qualité

*Donnez un résumé chiffré en termes d’évolution sur la qualité de votre code*

*La qualité a augmenté ou baissé tout au long du projet*

### Sur les résultats de vos tests

*% de succès / echec*

*% de la couverture de vos tests*

### Sur la tenue de charge

### Sur l'outil de traçabilité

*Donnez un résumé chiffré en termes d’évolution sur le nombre d’anomalies / d’évolution*

*Qui a fermé le plus d’anomalie*

*Reste-il des anomalies ouvertes*

Nous avons rencontré des problèmes pour lier notre base de données à notre application ComÉdu et pour réaliser les premières interfaces. Nous avons donc été contraints de ne pas lier notre base de données à l’application pour s’authentifier et de réduire le nombre d’interfaces telles les interfaces élèves, professeurs et une partie du module statistique afin d’avoir au minimum les interfaces pour les Journées Portes Ouvertes. Ceci correspond au must have des conditions de réussites de notre projet.

# V. Bilan du projet

Ce projet a été très instructif et enrichissant pour tous les membres du projet. Certains d’entre nous n’ont même jamais pu faire de la gestion de projet durant leur présence en entreprise.

Tout d’abord, nous avions dû trouver une idée de projet intéressante, aussi bien au niveau de l’innovation que de la complexité technique. Jordan OLIVIER a pu proposer une idée qui a plu aux autres membres du projet et aux membres du jury lors de la soutenance du 16 Février.

Ensuite, nous nous sommes repartis les taches en fonctions de nos capacités et nos compétences. A la suite de cela nous avons dû réaliser le planning de référence en utilisant un outil que nous n’avions jamais utilisé auparavant, qui est Microsoft Project.

De plus, l’installation et la configuration des différents outils nous ont permis de gagner en expérience et en aptitudes. Savoir mettre en place ces outils nécessaires au développement pourront nous servir dans notre vie professionnelle.

Nous avons également tous participé, à des échelles différentes, au développement de l’application et à la création des différentes interfaces, ce qui nous a permis d’augmenter nos compétences en développement et d’en acquérir de nouvelles.

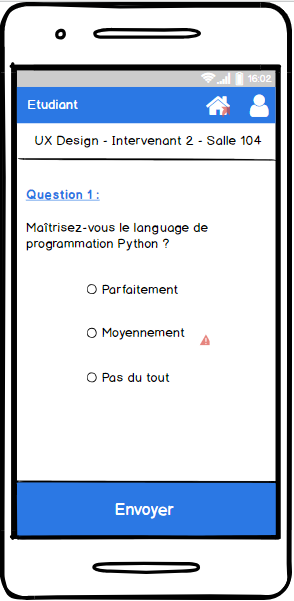
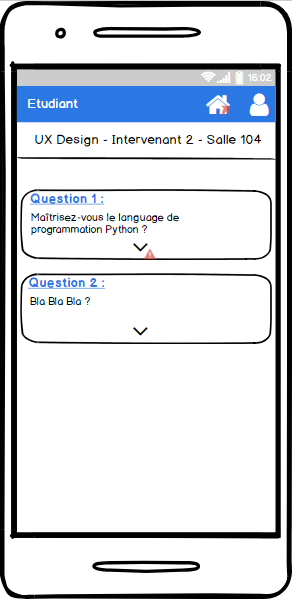
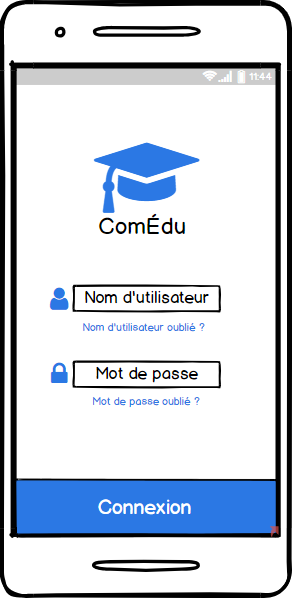
Cependant, nous avons tout de même rencontrés des difficultés et des échecs concernant la liaison entre la base de données et l’application. Nous avons également eu des problèmes sur la configuration de certains outils et sur la réalisation des certaines interfaces.

Nous avons fait de notre mieux pour respecter notre planning de référence. Mais au vu des difficultés rencontrées, nous avons été obligés de retarder certaines tâches et même de ne pas en réaliser telles que la réalisation des interfaces élèves et professeurs.

Pour conclure, même si l’application livrée n’est pas celle à laquelle nous nous attendions, nous avons appris des choses fondamentales telles que l’aboutissement d’une idée, la gestion d’équipe, la gestion des risques, le maintien du planning et la réalisation du journal de projet.

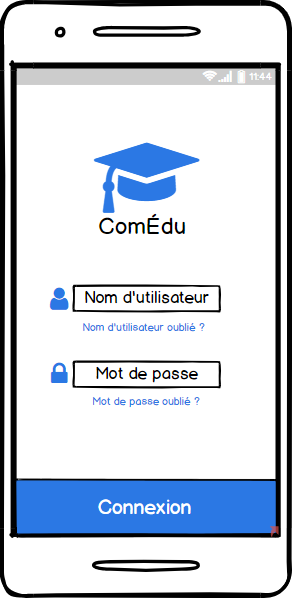
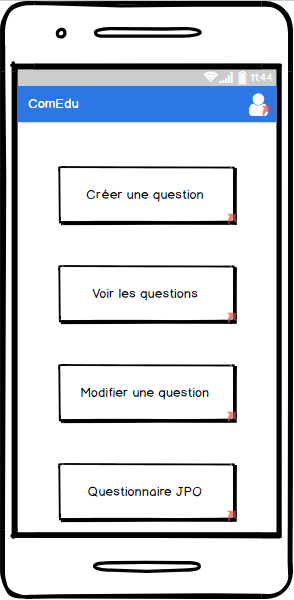
Afin d’améliorer notre démarche, nous avons appris une chose essentielle : faire des réunions hebdomadaires afin de suivre au mieux l’avancement du projet tout en apportant rapidement des solutions concrètes ce qui nous permettra de réduire les éventuels retards.

Scénario 3 : Un étudiant répond à une question posée par un professeur

 Se connecter Connexion Questions Répondre

Scénario 4 : Création de questions et visualisation des résultats

Se connecter Connexion Questions



Questions Modification Résultats

