****

**Ministère de l’Enseignement Supérieur et de la Recherche Scientifique**

**Direction Générale des Etudes Technologiques**

**Institut Supérieur des Etudes Technologiques de Bizerte**

**Département Technologies de l'Informatique**

|  |  |  |
| --- | --- | --- |
| Référence | Dép. | TI |
| **AN** | **2017** |
| **N°** | **DSI 20** |

**Rapport de**

**PROJET DE FIN D’ETUDES**

**En vue de l’obtention de :**

**Licence Appliquée en Développement des Systèmes d’Information « DSI »**

Application Android de Gestion du centre de formation

"Forma Pro"

Elaboré par :

**Jihene ABBESSI**

**&**

**Mohamed ABBES**

Encadré par :

**Mme Aïda BEN SALEM**

**Mme Yosra BEN SALEM**

Effectué à :

**Entreprise : Forma Pro**

**Adresse : 31, rue Thàalbi, Bizerte 7000**

**Tel : 29 215 715**

**Mail :** [**formapro@topnet.tn**](mailto:formapro@topnet.tn)

**Année universitaire : 2016/2017**

***A mes chers parents,***

*Que nulle dédicace ne puisse exprimer ce que je leurs dois, pour leur bienveillance, leur affection et leur soutien... Trésors de bonté, de générosité et de tendresse, en témoignage de mon profond amour et ma grande reconnaissance « Que Dieu vous garde ».*

***A mes chères sœurs,***

*En témoignage de mes sincères reconnaissances pour les efforts qu’elles ont consenti pour l’accomplissement de mes études. Je leur dédie ce modeste travail en témoignage de mon grand amour et ma gratitude infinie.*

***A tous mes amis,***

*Pour leur aide et leur soutien moral durant l’élaboration du projet de fin d’études.*

*Jihene*

***A mes parents***

*Pour tous les sacrifices qu’ils ont faits et pour tout le soutien qu’ils ont offert tout au long de mes études.*

*J’espère qu’ils puissent trouver dans ce modeste travail un témoignage d’amour et d’affection envers eux.*

***A toute ma famille***

*A ma sœur Je te remercie pour tes encouragements et*

*pour le soutien que tu m’as apporté.*

***A mes amis et mes collègues***

*Pour leurs encouragements et pour tous les bons moments qu’on a vécu*

*ensemble. J’espère que notre amitié durera éternellement.*

Mouhamed

1. Remerciements

*Nos remerciements les plus sincères à tous ceux et celles qui nous ont*

*apporté leur aide, leurs encouragements et leur soutien pour nous*

*permettre de mener à bien ce projet.*

*Nos remerciements s’adressent à notre encadreur, Mme Aida Ben Salem, pour*

*son assistance, son soutien, sa disponibilité et ses précieux conseils durant*

*la période de ce stage.*

*Nos remerciements à notre encadreur à FORMA PRO, Mme Yosra Ben Salem,*

*pour son entière disponibilité à nous fournir son assistance et sa*

*confiance la plus attentionnée.*

*Nos remerciements à tous ceux et celles qui nous ont apporté leur*

*contribution.*

1. Tables des matières

[Remerciements 2](#_Toc483751079)

[Tables des matières 3](#_Toc483751080)

[Liste des figures 6](#_Toc483751081)

[Introduction generale 8](#_Toc483751082)

[Chapitre. I Etude préalable 9](#_Toc483751083)

[Introduction 9](#_Toc483751084)

[I.1 Présentation du cadre du projet 9](#_Toc483751085)

[I.2 Etude de l’existant 10](#_Toc483751086)

[I.3 Critique de l’existant 12](#_Toc483751087)

[I.4 Solution proposée 13](#_Toc483751088)

[I.5 Etat de l’art 13](#_Toc483751089)

[I.5.1 Technologie Android 14](#_Toc483751090)

[I.5.2 Les web services 15](#_Toc483751091)

[I.5.3 L’architecture MVC 16](#_Toc483751092)

[I.6 Framework adopté 17](#_Toc483751093)

[I.6.1 Les méthodes agiles 17](#_Toc483751094)

[I.6.2 Le Framework SCRUM 18](#_Toc483751095)

[I.6.3 Le langage UML 19](#_Toc483751096)

[I.7 Etude Agile 19](#_Toc483751097)

[I.7.1 Scrum Team 20](#_Toc483751098)

[I.7.2 Les cérémonies 21](#_Toc483751099)

[I.7.3 Les artéfacts 22](#_Toc483751100)

[I.7.4 La définition du Done 26](#_Toc483751101)

[I.8 Spécification des besoins 26](#_Toc483751102)

[I.8.1 Solution retenue 26](#_Toc483751103)

[I.8.1 Identifications des acteurs 28](#_Toc483751104)

[I.8.2 Spécifications graphique 28](#_Toc483751105)

[I.8.3 Spécifications des besoins en sécurité 29](#_Toc483751106)

[I.9 Etude technique 29](#_Toc483751107)

[I.9.1 L’architecture adoptée 29](#_Toc483751108)

[I.9.2 Environnement du travail 32](#_Toc483751109)

[Conclusion 35](#_Toc483751110)

[Chapitre. II Release 1 36](#_Toc483751111)

[Introduction 36](#_Toc483751112)

[II.1 Organisation des Sprints 36](#_Toc483751113)

[II.2 Sprint 1 37](#_Toc483751114)

[II.2.1 Sprint Planning 37](#_Toc483751115)

[II.2.2 Conception du Sprint 1 38](#_Toc483751116)

[II.2.3 Réalisation et Test de l’incrément 44](#_Toc483751117)

[II.2.4 Sprint Review & Rétrospective 45](#_Toc483751118)

[II.3 Sprint 2 46](#_Toc483751119)

[II.3.1 Sprint Planning 46](#_Toc483751120)

[II.3.2 Conception du Sprint 2 47](#_Toc483751121)

[II.3.3 Réalisation de l’incrément 50](#_Toc483751122)

[II.3.4 Sprint Review & Rétrospective 51](#_Toc483751123)

[Conclusion 51](#_Toc483751124)

[Chapitre. III Release 2 52](#_Toc483751125)

[Introduction 52](#_Toc483751126)

[III.1 Organisation des Sprints 52](#_Toc483751127)

[III.2 Sprint 3 53](#_Toc483751128)

[III.2.1 Sprint Planning 53](#_Toc483751129)

[III.2.2 Conception du Sprint 3 54](#_Toc483751130)

[III.2.3 Réalisation et Test de l’incrément 56](#_Toc483751131)

[III.2.4 Sprint Review & Rétrospective 56](#_Toc483751132)

[III.3 Sprint 4 56](#_Toc483751133)

[III.3.1 Sprint Planning 56](#_Toc483751134)

[III.3.2 Conception du Sprint 4 57](#_Toc483751135)

[III.3.3 Sprint Review & Rétrospective 58](#_Toc483751136)

[Conclusion 58](#_Toc483751137)

[Chapitre. IV Release 3 59](#_Toc483751138)

[Introduction 59](#_Toc483751139)

[IV.1 Organisation des Sprints 59](#_Toc483751140)

[IV.2 Sprint 5 60](#_Toc483751141)

[IV.2.1 Sprint Planning 60](#_Toc483751142)

[IV.2.2 Conception du Sprint 5 61](#_Toc483751143)

[IV.2.3 Réalisation et Test de l’incrément 63](#_Toc483751144)

[IV.2.4 Sprint Review & Rétrospective 63](#_Toc483751145)

[IV.3 Sprint 6 63](#_Toc483751146)

[IV.3.1 Sprint Planning 63](#_Toc483751147)

[IV.3.2 Conception du Sprint 64](#_Toc483751148)

[IV.3.1 Sprint Review 65](#_Toc483751149)

[Conclusion 65](#_Toc483751150)

[Conclusion 66](#_Toc483751151)

[Netographie 67](#_Toc483751152)

[Annexes 68](#_Toc483751153)

1. Liste des figures

[Figure 1 Organigramme de l'entreprise 10](#_Toc483749902)

[Figure 2: Liste des clients 11](#_Toc483749903)

[Figure 3: Ajout d'un client 12](#_Toc483749904)

[Figure 4: Listes des clients inscrits 12](#_Toc483749905)

[Figure 5 Nougat 15](#_Toc483749906)

[Figure 6 Architecture d'un web service SOAP 16](#_Toc483749907)

[Figure 7: L'architecture MVC 17](#_Toc483749908)

[Figure 8: Principe du Scrum 19](#_Toc483749909)

[Figure 9 : Enchainement des Releases 25](file:///C:\Users\jiji\Desktop\rapport%20finale\RapportVFpfejihene.docx#_Toc483749910)

[Figure 10: Catalogue des formations Figure 11: Authentification 26](#_Toc483749911)

[Figure 12: Inscription Figure 13: Contact 27](#_Toc483749912)

[Figure 14: Profil Figure 15: Messagerie Figure 16: Les activités 27](#_Toc483749913)

[Figure 17 Login Figure 18 Dashboard Figure 19 Gestion des formateurs 28](#_Toc483749914)

[Figure 20 Architecture adoptée 29](#_Toc483749915)

[Figure 21 Diagramme de deploiement 30](#_Toc483749916)

[Figure 22 Diagramme des classes 31](#_Toc483749917)

[Figure 23: Android Studio 32](file:///C:\Users\jiji\Desktop\rapport%20finale\RapportVFpfejihene.docx#_Toc483749918)

[Figure 24 : Just In Mind 32](file:///C:\Users\jiji\Desktop\rapport%20finale\RapportVFpfejihene.docx#_Toc483749919)

[Figure 25 Easy PHP 33](#_Toc483749920)

[Figure 26: StarUML 33](#_Toc483749921)

[Figure 27 Adobe Dreamweaver CS6 34](#_Toc483749922)

[Figure 28 : JAVA 34](file:///C:\Users\jiji\Desktop\rapport%20finale\RapportVFpfejihene.docx#_Toc483749923)

[Figure 29 php 34](#_Toc483749924)

[Figure 30 : RELEASE 1 37](#_Toc483749925)

[Figure 31: Diagramme de cas d'utilisation du sprint1 39](#_Toc483749926)

[Figure 32: Diagramme de séquence « S'inscrire » 41](#_Toc483749927)

[Figure 33: Diagramme de séquence « S'authentifier » 42](#_Toc483749928)

[Figure 34: Diagramme de séquence « Consulter le catalogue 43](#_Toc483749929)

[Figure 35 Connexion Figure 36 Inscription Figure 37 Localisation du centre 44](#_Toc483749930)

[Figure 38 A propos Figure 39 Catalogue Figure 40 Catégorie informatique 44](#_Toc483749931)

[Figure 41 : Schéma du Sprint Review 45](file:///C:\Users\jiji\Desktop\rapport%20finale\RapportVFpfejihene.docx#_Toc483749932)

[Figure 42: Diagramme de cas d'utilisation du sprint 2 47](#_Toc483749933)

[Figure 43 A Diagramme de séquence « suivre mes activités» 50](#_Toc483749934)

[Figure 44 Espace du client Figure 45 Les activités Figure 46 Profil 51](#_Toc483749935)

[Figure 47 : Release 2 53](file:///C:\Users\jiji\Desktop\rapport%20finale\RapportVFpfejihene.docx#_Toc483749936)

[Figure 48 : Diagramme de cas d'utilisation du Sprint 3 54](file:///C:\Users\jiji\Desktop\rapport%20finale\RapportVFpfejihene.docx#_Toc483749937)

[Figure 49 Ajout d'un client Figure 50 Gestion des formateurs 56](#_Toc483749938)

[Figure 51 : Diagramme de cas d'utilisation du Sprint 4 57](file:///C:\Users\jiji\Desktop\rapport%20finale\RapportVFpfejihene.docx#_Toc483749939)

[Figure 52 Release 3 60](#_Toc483749940)

[Figure 53 : Diagramme de cas d'utilisation DU << Sprint 5 >> 61](#_Toc483749941)

[Figure 54 Messagerie 63](#_Toc483749942)

[Figure 55 Diagramme de cas d'utilisation du << Sprint 6 >> 65](file:///C:\Users\jiji\Desktop\rapport%20finale\RapportVFpfejihene.docx#_Toc483749943)

[Figure 56 Cycle de vie d’une application Android 68](#_Toc483749944)

**Liste des tableaux**

[Tableau 1Les versions d'Android 14](#_Toc483751154)

[Tableau 2 : Product Backlog 24](#_Toc483751155)

[Tableau 3 : Matériels utilisés 32](#_Toc483751156)

[Tableau 4 Backlog du «Sprint 1 » 38](#_Toc483751157)

[Tableau 5 Description textuelle du diagramme de cas d’utilisation «S’inscrire» 39](#_Toc483751158)

[Tableau 6 Description textuelle du diagramme de cas d’utilisation «S’authentifier» 40](#_Toc483751159)

[Tableau 7 Backlog du « Sprint 2 » 46](#_Toc483751160)

[Tableau 8 Description textuelle du diagramme de cas d’utilisation « Suivre mes activités » 48](#_Toc483751161)

[Tableau 9 Description textuelle du diagramme de cas d’utilisation «Consulter les détails d’une formation» 48](#_Toc483751162)

[Tableau 10 Description textuelle du diagramme de cas d’utilisation «S’inscrire à une formation» 49](#_Toc483751163)

[Tableau 11 Sprint Backlog du Sprint 3 53](#_Toc483751164)

[Tableau 12 DESCRIPTION TEXTUELLE DU DIAGRAMME DE CAS D'UTILISATION << Gérer le catalogue >> 55](#_Toc483751165)

[Tableau 13 DESCRIPTION TEXTUELLE DU DIAGRAMME DE CAS D'UTILISATION « Gérer les clients » 55](#_Toc483751166)

[Tableau 14 DESCRIPTION TEXTUELLE DU DIAGRAMME DE CAS D'UTILISATION « Gérer les formateurs » 55](#_Toc483751167)

[Tableau 15 Sprint Backlog du Sprint 4 57](#_Toc483751168)

[Tableau 16 DESCRIPTION TEXTUELLE DU DIAGRAMME DE CAS D'UTILISATION <<GERER LES SALLES>> 58](#_Toc483751169)

[Tableau 17 DESCRIPTION TEXTUELLE DU DIAGRAMME DE CAS D'UTILISATION << Réserver une salle >> 58](#_Toc483751170)

[Tableau 18 Sprint Backlog du Sprint 5 61](#_Toc483751171)

[Tableau 19 DESCRIPTION TEXTUELLE DU DIAGRAMME DE CAS D’UTILISATION «Consulter la boite de réception» 62](#_Toc483751172)

[Tableau 20 DESCRIPTION TEXTUELLE DU DIAGRAMME DE CAS D’UTILISATION «Envoyer un message» 62](#_Toc483751173)

[Tableau 21 TABLEAU 21 DESCRIPTION TEXTUELLE DU DIAGRAMME DE CAS D’UTILISATION «Recevoir un message » 62](#_Toc483751174)

[Tableau 22 Sprint Backlog du Sprint 6 64](#_Toc483751175)

1. Introduction generale

L’avantage compétitif des centres de formation repose sur leur capacité à répondre rapidement aux changements des besoins du marché, et à augmenter leurs bénéfices en produisant des produits, des biens et des services qui satisfait les souhaits et les attentes de la clientèle.

Dans un monde en constante évolution et un environnement concurrentiel, un avantage compétitif pour les centres de formation ainsi que leurs clients potentiels, est de bénéficier des nouvelles technologies de l’information et de communication (NTIC) pour fournir le meilleur service, au moment opportun et au bon endroit.

La Forma Pro, visant cibler une clientèle variée, se trouve orientée vers le développement d’une application mobile pour satisfaire le besoin de mobilité et de la disponibilité du service quel que soit le lieu et le temps.

Dans ce contexte, s’introduit notre projet de fin d’études intitulé : conception et développement d’une application Android « Forma Pro ».

Pour mener à bien notre projet de fin d’études et atteindre les objectifs visés, nous commençons par une étude de l’existant de la Forma Pro. Ensuite, nous choisissons une architecture Client/Serveur afin d’implémenter les différents modules logiciels de l’application, et enfin, nous procédons avec le Framework SCRUM afin de garantir la satisfaction de notre client par chaque incrément développé.

Le présent rapport se décompose en trois chapitres. Tout d’abord, le premier chapitre qui est consacré à l’étude préalable du projet, comportant la présentation du cadre de travail, la problématique posée, les objectifs visés et la solution choisie ainsi que le Framework adopté SCRUM et l’architecture globale de l’application. Ensuite, le deuxième chapitre, intitulé « Release 1 » et qui permet de clarifier les détails du premier livrable, il regroupe les deux premiers sprints 1 et 2. Et enfin, le troisième chapitre « Release 2 » présente en détail les sprints 3 et 4. Et le quatrième chapitre, intitulé, « Release 3 » est consacré pour les sprints 6 et 7.

* 1. Etude préalable
     1. Introduction

Dans ce chapitre, nous nous intéressons à la présentation de la société Forma Pro au sein de laquelle nous avons effectué notre stage. Ensuite, nous entamerons l’étude et la la critique de l’existant. Après, nous passerons à une étude agile où nous explicitons l’adaptation du Framework SCRUM dans notre projet.

* + - 1. Présentation du cadre du projet

Forma Pro est un centre de formation professionnelle privé situé en Tunisie crée en 2 Mai 2013 et agree par le Ministère de la Formation professionnelle et de l’Emploi

* **Organigramme**

Figure 1 Organigramme de l'entreprise

* + - 1. Etude de l’existant

Forma Pro dispose actuellement d’une application web disponible pour les clients, nous pouvons la voir dans les figures suivantes :

|  |  |
| --- | --- |
| Figure 2: Liste des clients |  |
|  |  |
| Figure 3: Ajout d'un client |  |
| Figure 4: Listes des clients inscrits |  |

* + - 1. Critique de l’existant

Après une étude approfondie de l’existant nous avons dégagé un certain nombre de lacunes :

* L’application est lente
* Elle n’exploite pas la puissance hardware dont dispose les Smartphones. (La géolocalisation par GPS, l’Accéléromètre, la possibilité de sauvegarder des données en local etc.)
* Pas de services si le terminal ne dispose pas d’accès internet.
* Des bugs liés au développement de l’application.
* Une interface non ergonomique
* Manque de fluidité

L’application existante au sein du centre ne permet pas une interaction entre l’administrateur et le client.

* + - 1. Solution proposée

Le marché des centres de formation, comme toute autre entreprise, suit les dernières évolutions technologiques afin de séduire leurs clients. La solution proposée est une application mobile permettant de:

* *Renforcer la relation avec les clients* :Le téléphone mobile est un objet disponible et personnel, il permet d’établir une meilleure communication et un lien fort de proximité avec les formateurs d’un côté et avec les clients de l’autre côté.
* *Optimiser les coûts* :Une application mobile permet d’une part, à un client distant de consulter le catalogue à travers son téléphone mobile. Il peut également s’inscrire à une formation, il peut également suivre ses formations. D’autre part, cette application permettra au formateur de réserver une salle, de consulter les formations qu’il doit assurer. Cette application mobile permettra aussi à l’administrateur de suivre les clients, les formateurs et les formations.
* *Accroitre son Chiffre d’Affaires***:** L’application mobile facilitera la consultation du catalogue des formations et accélèrera le processus d’inscription.
  + - 1. Etat de l’art

Dans cette partie nous allons présenter les technologies existantes, leurs points forts et faibles.

* + - * 1. Technologie Android
* Présentation

Android est un système d’exploitation Open Source pensé pour les téléphones mobiles et développé par **Handset Alliance (OHA1)** sous l’autorité de Google.

Android est basé sur le noyau linux et possède une **licence Apache 2.0**. Il sépare la couche logicielle de la couche matérielle, c’est-à-dire que n’importe quel téléphone Android peut faire fonctionner la même application.

Android est aussi un **framework libre** et en tant que développeur ; on a accès au SDK et à tout le code source de la plateforme.

Plusieurs appareils tournant sous Android ont vu le jour, tels que, autoradio, montre, casque de ski…

* Historique des versions

Android est apparu en octobre **2003,** lorsque Andy Rubin, Rich Miner, Nick Sears et Chris White créent la société Android à Palto Alto.

Google a racheté la société en **2005**. En **2007**, OHA est apparu et Android devient Open Source.

La première version du SDK Android 1.0 sort en **2008** avec le premier Téléphone Android HTC Dream.

Les versions d’Android se sont suivi comme suit :

|  |  |  |
| --- | --- | --- |
| **Nom** | **Version** | **Date** |
| **Android** | 1.0 | 09/2008 |
| **Petit Four** | 1.1 | 02/2009 |
| **Cupcake** | 1.5 | 04/2009 |
| **Donut** | 1.6 | 09/2009 |
| **Gingerbread** | 2.3 | 12/2010 |
| **Honeycomb** | 3.0 | 02/2011 |
| **Ice Cream Sandwich** | 4.0.1 | 10/2011 |
| **Jelly Bean** | 4.1 | 07/2012 |
| **KitKat** | 4.4 | 10/2013 |
| **Lollipop** | 5.0 | 10/2014 |
| **Marshmallow** | 6.0 | 05/2015 |
| **Nougat** | 7.0 | 09/2016 |

Tableau 1Les versions d'Android



Figure 5 Nougat

* + - * 1. Les web services

Avec l'avènement de l'informatique et surtout l'émergence de l'Internet, les entreprises optent pour des solutions dans le but de faire fonctionner des applications de façon distantes, ou celles-ci sont généralement liées à une base de données déployée sur un serveur.

Ce concept la nécessite alors un composant gérant l'interaction entre le serveur et ces applications ; d'où le besoins des WS.

* **Définition d’un WS**

Un WS est déni comme "une technologie émergente permettant à des applications de dialoguer à distance via Internet et ceci, indépendamment des plateformes sur lesquelles elles reposent." Maguelonne Héritier

En effet, nous comprenons les origines de l'appellation Web Service : c'est une application interagissant avec d'autres applications distantes et hétérogènes en leur fournissant les services dont elles sont besoin à travers le réseau Internet. Ce qui implique qu'un WS doit être identifie par une adresse représentant son identité sur le web : c'est son URI, il doit aussi faire intervenir un protocole garantissant l'échange de données.

* **Protocoles d'implémentation de WS : SOAP**

C'est un protocole léger assurant l'échange des messages entre un client et un serveur dans les architectures dites Orientées Services (SOA). Il repose sur le standard XML pour la structure des messages et aussi sur le standard HTTP pour le transport des messages. Ses principaux avantages sont : la portabilité, l'utilisation de standards ouverts et l'interopérabilité.

Les services sont déployés par le serveur dans un annuaire UDDI et y sont enregistrés. Un client venant chercher un service dans l'annuaire, est dirigé vers le serveur qui l'offre. Le client demande au serveur l'interface du service, l'analyse et la comprend puis envoie sa requête. Le serveur exécute la requête et lui retourne le résultat. Les \_échanges sont effectués en langage XML en faisant intervenir un contrat SOAP.

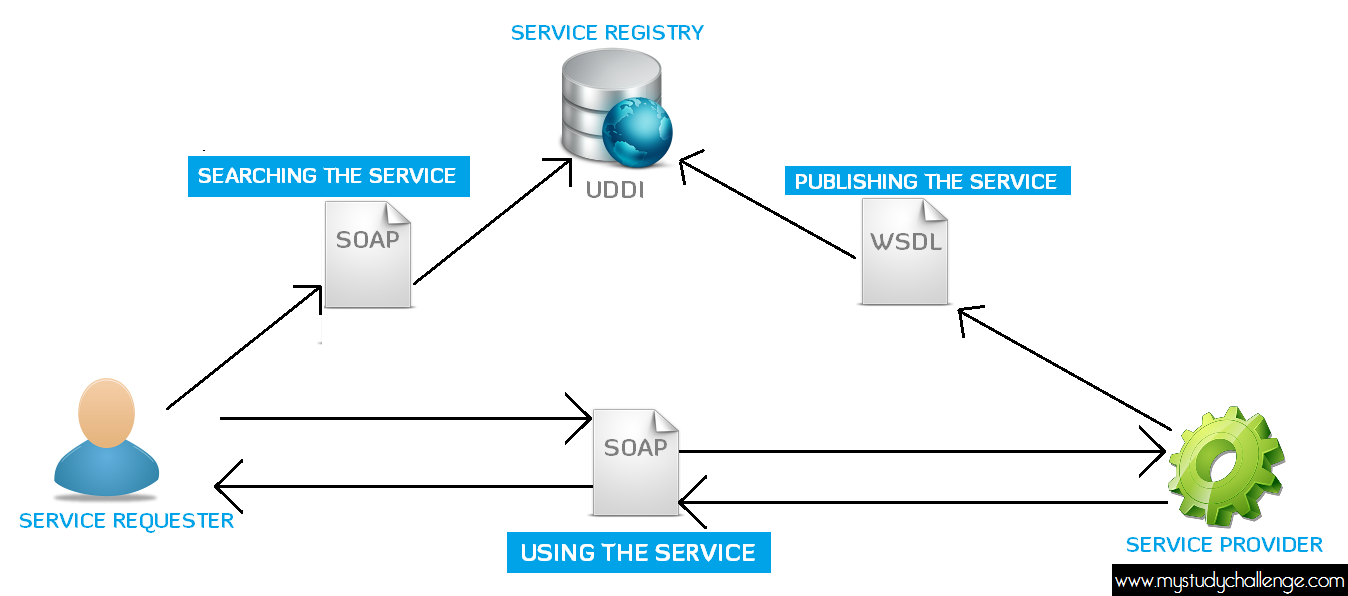


Figure 6 Architecture d'un web service SOAP

Notre service web est implémenté dans un fichier d'extension PHP de façon incrémentale. Sa description dans le langage WSDL est générée automatiquement à partir du script PHP par le biais de la bibliothèque de classe "NuSOAP". Pour pouvoir consommer ce service, le client mobile y fait référence.

Nous allons nous intéresser plus précisément aux services retournant des données en JSON (JavaScript Object Notation) qui est un format léger d'échange de données.

* **Pourquoi le JSON ?**

Il est possible d’utiliser XML, mais JSON a la particularité d'être plus léger et un gain de poids n'est pas anodin dans le domaine de la mobilité.

* + - * 1. L’architecture MVC

L'architecture **MVC** (Modèle, Vue et Contrôleur) est un concept très puissant qui intervient dans la réalisation d'une application. Son principal intérêt est la séparation des données (modèle), de l'affichage (vue) et des actions (contrôleur).

**Le Modèle** représente le comportement de l'application : traitements des données, interactions avec la base de données, etc. Il décrit les données manipulées par l'application et définit les méthodes d'accès.

**La Vue** correspond à l'interface avec laquelle l'utilisateur interagit. Les résultats renvoyés par le modèle sont dénués de toute présentation mais sont présentés par les vues. Plusieurs vues peuvent afficher les informations d'un même modèle. Elle peut être conçue en html, ou tout autre " langage " de présentation. La vue n'effectue aucun traitement, elle se contente d'afficher les résultats des traitements effectués par le modèle, et de permettre à l'utilisateur d'interagir avec elle.

**Le Contrôleur** prend en charge la gestion des évènements de synchronisation pour mettre à jour la vue ou le modèle. Il n'effectue aucun traitement, ne modifie aucune donnée, il analyse la requête du client et se contente d'appeler le modèle adéquat et de renvoyer la vue correspondant à la demande.



Figure 7: L'architecture MVC

* + - 1. Framework adopté

Nous allons maintenant aborder les méthodes agiles ainsi que le Framework SCRUM, le langage de conception UML et enfin la démarche de réalisation.

* + - * 1. Les méthodes agiles

« Une méthode agile est une approche itérative et incrémentale, qui est menée dans un esprit collaboratif avec juste ce qu’il faut de formalisme. Elle génère un produit de haute qualité tout en prenant en compte l’évolution des besoins des clients » (Véronique Messager Rota, Gestion de projet : Vers les méthodes agiles)

Les principes de l’agilité:

* La satisfaction client
* L’excellence technique
* Les individus
* L’amélioration continue
  + - * 1. Le Framework SCRUM

SCRUM est un Framework agile qui suit un processus itératif incrémental, en effet un projet SCRUM passe toujours par une séquence d’évènements bien connus.

Il se compose d’un SCRUM Team qui dispose de trois rôles SCRUM Master, Product Owner et Development Team que nous allons détailler dans le chapitre suivant.

Le cœur de SCRUM est le Sprint dont l’objectif est de réaliser un incrément potentiellement livrable et de plus haute valeur ajouté.

Il se base sur trois piliers :

* La transparence
* L’inspection
* L’adaptation

La figure suivante illustre le parcours SCRUM ainsi que ses différents composants.

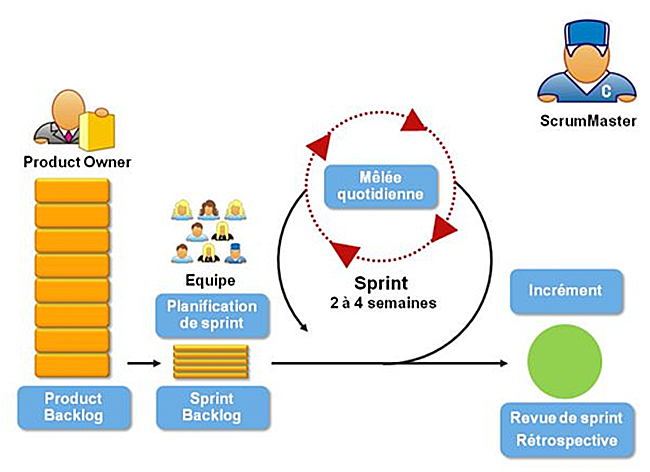


Figure 8: Principe du Scrum

* + - * 1. Le langage UML

Pour faciliter la mise en place des Sprints et mieux documenter les incréments nous avons eu recours à l’utilisation du langage de modélisation UML dans sa version 2.0.

UML (Unified Modeling Language) est un langage de modélisation objet permettant de représenter n’importe quel système d’informations sous forme de modèles autour de trois axes ; un axe statique, un axe dynamique et un axe fonctionnel.

Nous nous engageons dans la partie conception des sprints de réaliser des modèles de différents axes afin d’avoir une vue complète et de clarifier au mieux le développement des sprints.

* + - 1. Etude Agile

« L’agilité est la capacité à favoriser le changement et y répondre en vue de s’adapter au mieux à un environnement turbulent » (Jim Highsmith)

Dans le contexte de notre application, les changements et les mises à jour des besoins sont fréquents et cruciaux pour l’élaboration d’une application qui satisfait au mieux son utilisateur final. C’est pour cette raison qu’une étude agile est mise en jeux pour déterminer notre aptitude à réaliser notre projet.

* + - * 1. Scrum Team

L’équipe SCRUM est une équipe composée de trois rôles, qui sont :

* Scrum Master
* Product Owner
* Development Team

Ces trois rôles seront détaillés dans la partie ci-après.

* Scrum Master

Il est le responsable de la bonne compréhension de SCRUM et de sa mise en œuvre. Le Scrum Master de notre projet est notre encadrant à l’ISET de Bizerte qui nous aide à développer un côté d’auto-organisation avantageux et intéressant pour notre savoir être et savoir-faire. Notre Scrum Master nous encourage et nous pousse à prendre les décisions qui nous concerne par nous-même.

Il nous facilite l’avancement dans le projet en essayant d’éliminer quelques obstacles rencontrés au cours de la réalisation. Il nous responsabilise aussi envers le développement et nous motive continuellement à l’accomplissement de l’application dans les délais et selon les règles et les normes SCRUM.

* Product Owner

Il est le responsable qui établit la vision produit et en même temps c’est le représentant du client. C’est à lui qu’ont été confiées : la collecte et la négociation des exigences et des priorités du client.

Pour notre projet, c’est notre encadrant à la Forma Pro qui maintient le Product Backlog. Il exprime clairement les cas d’utilisation, nous donne la priorité de chacune et planifie aussi leurs réalisations.

Notre Product Owner s’assure toujours que les cas d’utilisation sont clairs et compréhensibles. Il est à l’écoute de nos interrogations, nos réclamations et nos propositions.

C’est à lui aussi de décider les livraisons des incréments développés tout en tenant en considération l’objectif de maximiser le Business Value de l’application.

* Development Team

L’équipe de développement est responsable de livrer à chaque sprint un incrément Done et potentiellement livrable du produit. Nous, membres de l’équipe de développement, nous développons à travers ce projet une auto-organisation et des compétences variées.

Nous avons appris le vrai sens de l’engagement et la discipline. Etre membre d’une équipe de développement SCRUM est une responsabilité à tenir.

Nous formons Jihene ABBESSI et Mouhamed BEN ABBES les deux membres de l’équipe Scrum.

* + - * 1. Les cérémonies
* Sprint Planning

C’est un événement qui démarre chaque sprint et qui réunit la Scrum Team pendant 4 heures vu que nos sprints s’étalent seulement sur deux semaines.

Au cours de cette cérémonie nous avons fixé les deux objectifs suivants :

* Définir le sprint goal
* Elaborer le Sprint Backlog
* Daily Meeting

Nous ; équipe de développement, nous avons consacré 15 minutes chaque matin, au Forma Pro où nous effectuons notre projet pour faire le point sur :

* Ce que nous avons réalisé la veille pour atteindre l’objectif du sprint.
* Ce que nous réaliserons le jour même pour atteindre l’objectif du sprint.
* Les obstacles qui nous empêchent d’atteindre l’objectif du sprint.
* Sprint Review

Cette cérémonie est tenue à la fin du sprint pour inspecter l’incrément résultant du sprint. C’est aussi une occasion pour adapter le Product Backlog si nécessaire. La Scrum Team participe à cet événement.

Nous avons réalisé cet événement au cours de notre projet selon la disponibilité de notre Scrum Master, de notre Product Owner et d’autres équipes de la Forma Pro. La présence de ces trois profils de professionnels nous permettait d’avoir un avis critique sur l’incrément réalisé.

* Sprint Retrospective

Cette cérémonie représente une opportunité pour la Scrum Team pour s’inspecter et créer un plan d’amélioration qui sera mis en place au cours du Sprint qui suit. Cette réunion aura lieu après le Sprint Review et avant le prochain Sprint planning.

Notre objectif à travers chaque Sprint Retrospective est de :

* Revoir la manière dont le sprint s’est déroulé en ce qui concerne nos interactions, nos relations et la fluidité de passage de l’information au sein de toute l’équipe.
* Déterminer les points forts qui nous ont amené à réussir le sprint encours et de les consolider dans les prochains sprints.
  + - * 1. Les artéfacts

Les artéfacts de Scrum sont en nombre de trois et ils seront détaillés dans les sections ci-dessous. En effet, ce sont des outils nécessaires pour l’accomplissement du travail demandé.

Nous avons gardé toujours la transparence, l’inspection et l’adaptation lors de l’élaboration ou l’utilisation de ces artéfacts.

* Le Product Backlog

C’est l’ensemble des cas d’utilisation exprimé par l’utilisateur final de notre application et rédigé et priorisé par le Product Owner. C’est l’une des manifestations agiles de SCRUM. En effet, c’est un artéfact vivant et dynamique vu qu’il s’adapte convenablement aux changements de tout contexte : changement des besoins, de la technologie, du marché, etc.

Comme équipe de développement, nous avons toujours recours au Product Backlog et à son rédacteur le Product Owner.

* Présentation Product Backlog de l’application

Dans ce qui suit le Product Backlog de notre application.

Nous choisissons volontairement les user stories qui conviennent avec notre effort personnel de développement et nous construisons seuls nos sprints à partir du Product Backlog.

|  |  |  |  |
| --- | --- | --- | --- |
| Nom | Description | Complexité | Priorité |
| Authentification | En tant qu’utilisateur (client, administrateur ou formateur) je veux m’authentifier. | moyenne | Élevée |
| Inscription à l’application | En tant qu’utilisateur je dois pouvoir m’inscrire. | moyenne | Élevée |
| Consultation du catalogue des formations | En tant qu’utilisateur je veux consulter le catalogue de formation. | moyenne | Élevée |
| Inscription a une formation | En tant que client je dois pouvoir m’inscrire à une formation. | moyenne | Élevée |
| Consultation des informations du centre | En tant qu’utilisateur je veux consulter les informations du centre. | moyenne | Élevée |
| Consultation de page contact | En tant qu’utilisateur je veux accéder au contact. | moyenne | Élevée |
| Profil | En tant qu’utilisateur inscrit je dois gérer mon compte. | moyenne | Élevée |
| Suivi des activités | En tant qu’un client je veux suivre mes activités. | moyenne | Élevée |
| Consultation des formations | En tant que formateur je dois pouvoir consulter mes formations (assurées et a venir). | moyenne | Élevée |
| Détails d’une formation | En tant que client je veux voir les détails d’une formation. | moyenne | Élevée |
| Gestion de catalogue | En tant qu’administrateur je dois gérer le catalogue de formation. | moyenne | Élevée |
| Gestion des clients | En tant qu’administrateur je dois gérer les clients, dans le but d’ajouter ou modifier un client. | moyenne | Élevée |
| Gestion des formateurs | En tant qu’administrateur je dois gérer les formateurs, dans le but d’ajouter un formateur ou l’affecter à une formation. | moyenne | Élevée |
| Gestion des salles | En tant qu’administrateur je dois gérer les salles. | moyenne | Élevée |
| Réservation d’une sale | En tant que formateur je dois pouvoir réserver une salle. | moyenne | Élevée |
| Gestion des paiements | En tant qu’administrateur je dois gérer les paiements des clients et formations. | moyenne | Élevée |
| Demande d’attestation/diploid | En tant que client je dois commander une attestation ou un diplôme. | moyenne | moyenne |
| Messagerie | En tant qu’administrateur/client je dois pouvoir envoyer ou recevoir des messages privés. | complexe | moyenne |
| Statistiques | En tant qu’administrateur je dois consulter les statistiques. | complexe | moyenne |
| Noter une formation | En tant que client je dois pouvoir noter une formation après l’avoir terminés. | moyenne | Faible |
| Proposition d’une formation | En tant que client je veux proposer une formation. | moyenne | Faible |
| Commenter une formation | En tant que client je dois pouvoir commenter (d’une façon anonyme) une formation. | moyenne | Faible |

Tableau 2 : Product Backlog

* Enchainement de la livraison

Nous optons dans le présent rapport d’organiser les chapitres selon la fréquence de la livraison, raison pour laquelle nous clarifions les releases et leurs structurations en sprints.

La figure ci-dessous montre l’agencement des releases et par la suite l’agencement des chapitres du rapport.

Figure 9 : Enchainement des Releases

* Le Sprint Backlog

Le Sprint Backlog est un sous ensemble du Product Backlog. Il contient seulement les cas d’utilisation qui permettent de réaliser l’objectif du sprint.

Pour l’élaboration de cet artéfact, nous avons détaillé les tâches à effectuer pour chaque cas d’utilisation. Nous avons tout droit sur le sprint Backlog car nous sommes les seuls responsables de son maintien et sa réalisation vu que c’est un document qui nous appartient.

Outre ces tâches, nous avons consacré une partie dans le sprint Backlog pour l’estimation de la charge de travail (le degré de difficulté, la priorité et la durée).

* L’incrément

« L’incrément est constitué des éléments du Product Backlog terminés pendant ainsi que la valeur cumulative des incréments livrés dans les sprints précédents. »

Les incréments que nous avons réalisé sont tous utilisables, potentiellement livrables et respectent nécessairement la définition du Done.

Dans le présent rapport nous nous sommes engagés de montrer dans chaque sprint l’incrément qui en résulte.

* + - * 1. La définition du Done

Afin d’améliorer la transparence dans l’équipe SCRUM et garantir une inspection efficace et par la suite une meilleure adaptation, nous avons défini le Done de la manière suivante :

Un incrément est dit Done, si et seulement si, tout effort de conception, de développement et de test sont totalement terminés.

Nous devons respecter cette définition tout au long du projet. C’est un auto-engagement que nous avons pris en considération durant tous les sprints.

* + - 1. Spécification des besoins
         1. Solution retenue
* **Pour les clients**

La Forma Pro a opté pour la conception et le développement d’une application Android « Forma Pro » destinés à ses clients qui dispose de deux modes :

* Mode Offline : Offre à tout utilisateur (qu’il soit un client ou un simple internaute) des services qui ne demande pas un compte comme :
  + L’accès au catalogue de formations
  + L’affichage de la localisation du centre sur la carte géographique
  + L’accès aux informations concernant le centre
  + Faire passer un appel vers le centre

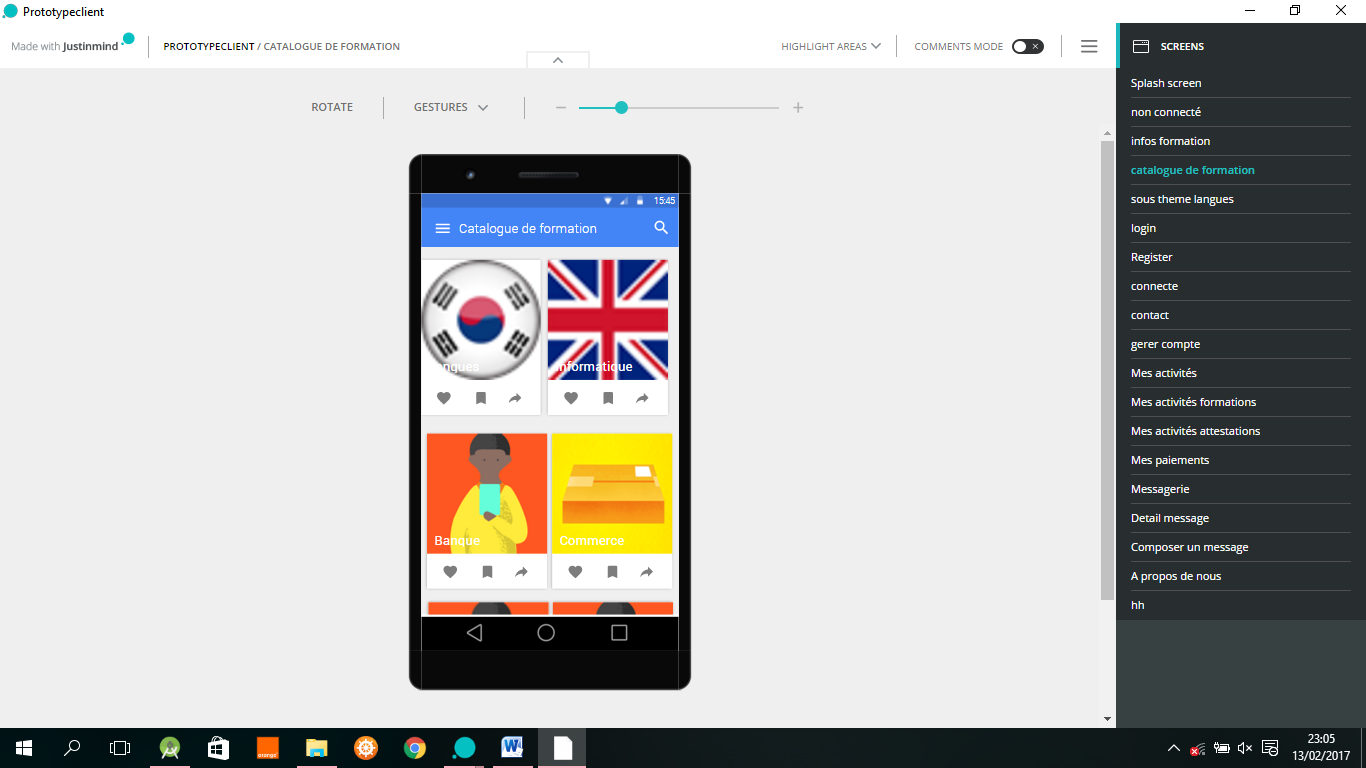
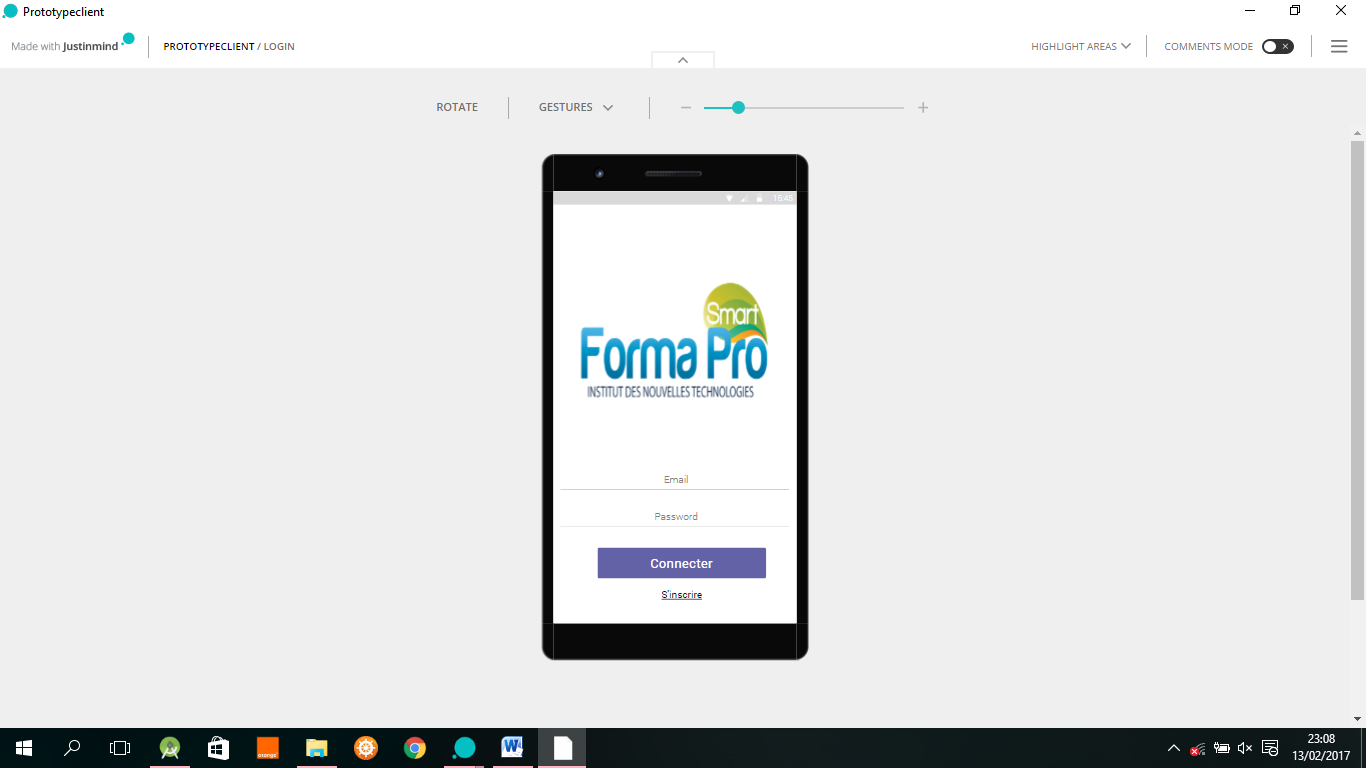
 

Figure 10: Catalogue des formations Figure 11: Authentification

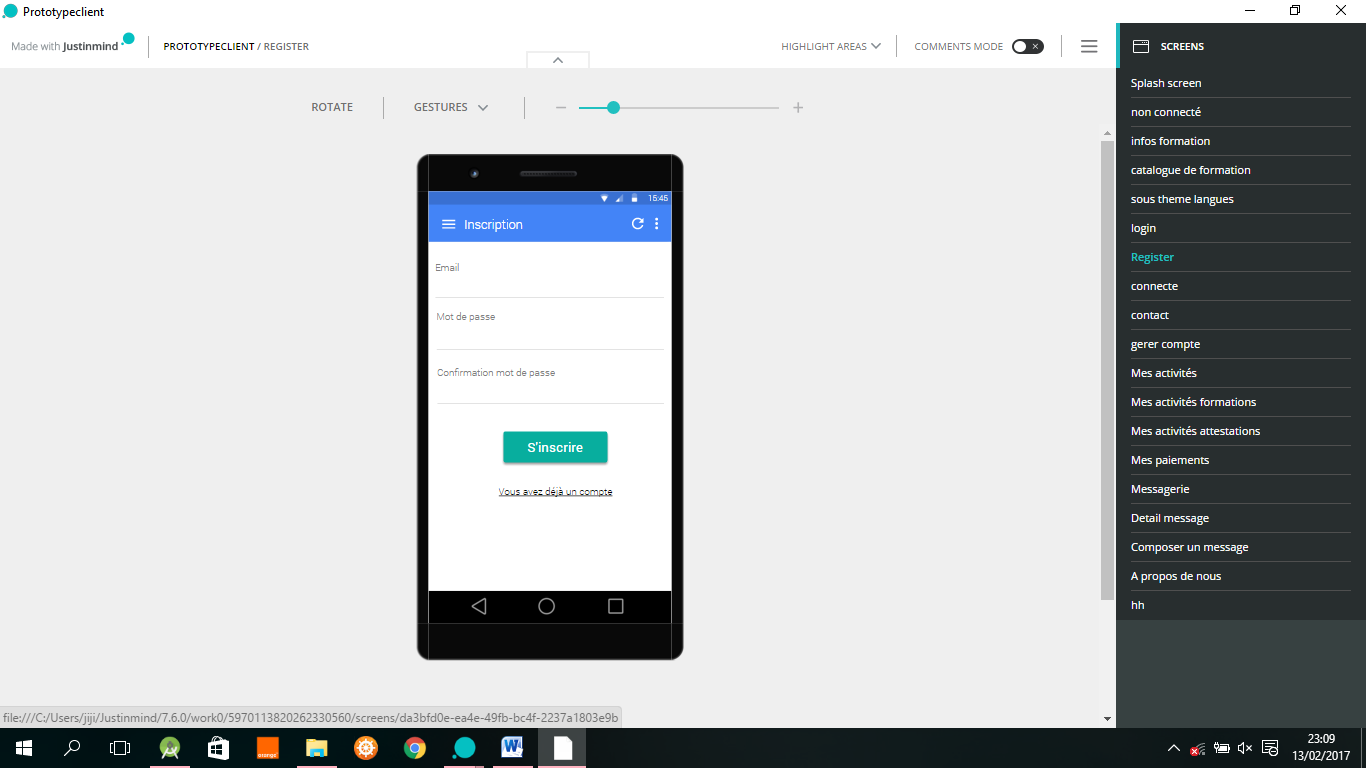
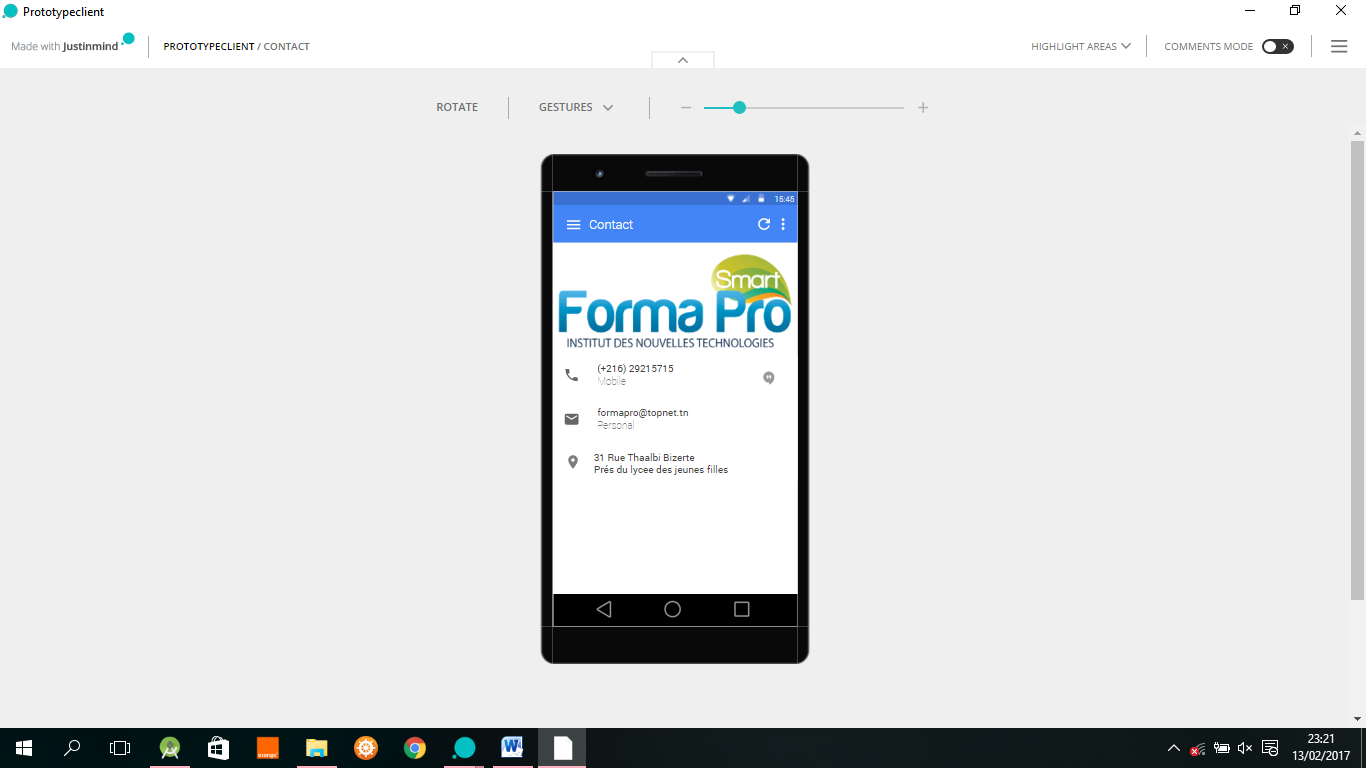
 

Figure 12: Inscription Figure 13: Contact

* Mode Online : Offre aux clients des services tels que :
  + Un espace de messagerie avec l’administrateur
  + Mise à jour de son profil
  + Inscription à une formation
  + Commande d’ une attestation
  + Gestion de son espace (historique des formations, historiques des attestations, historique des paiements, ses attestations)

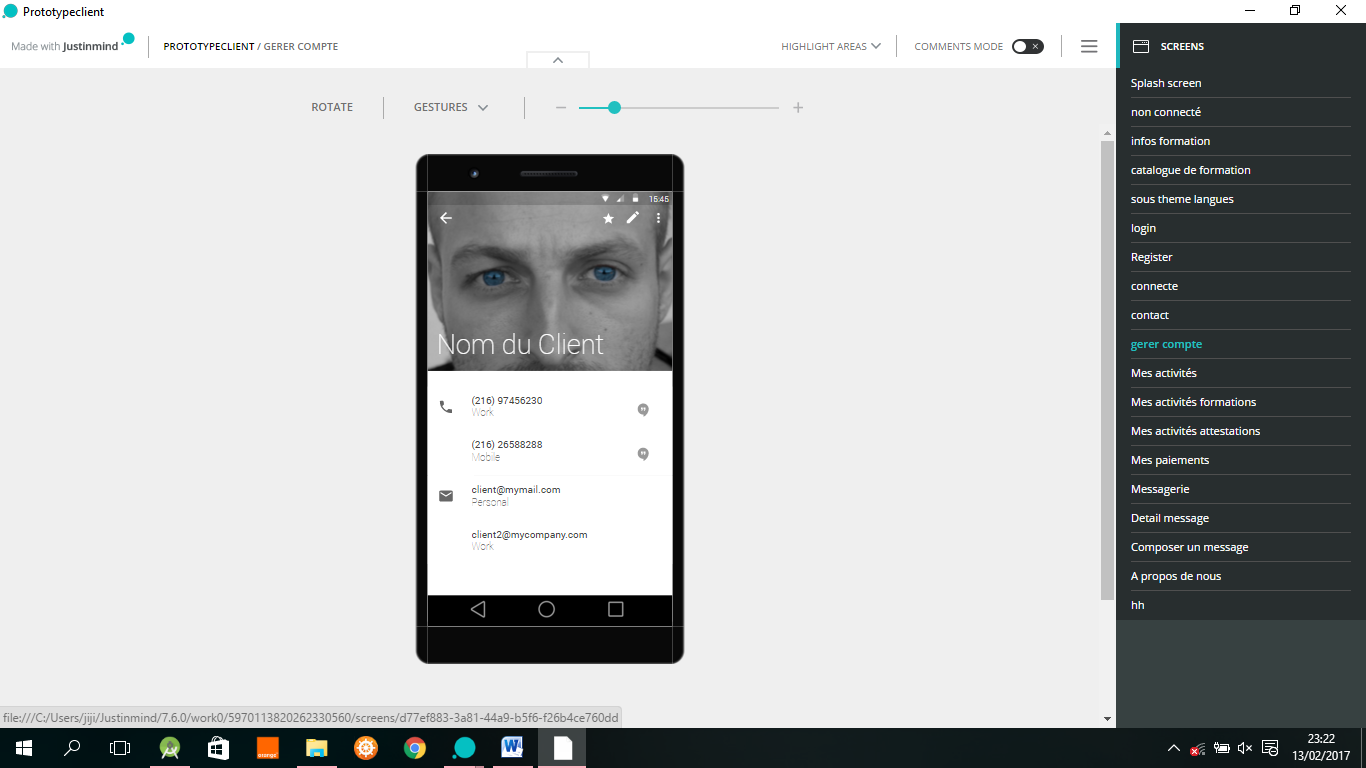
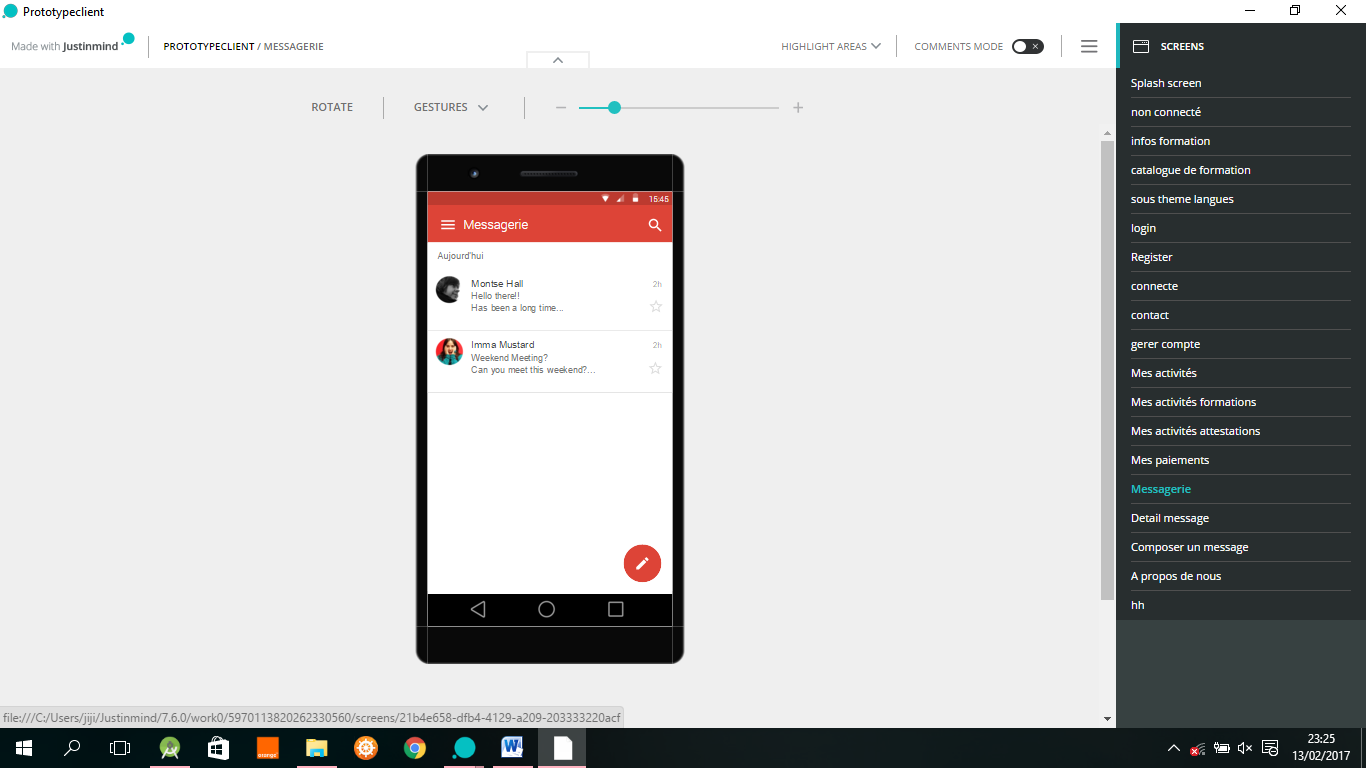
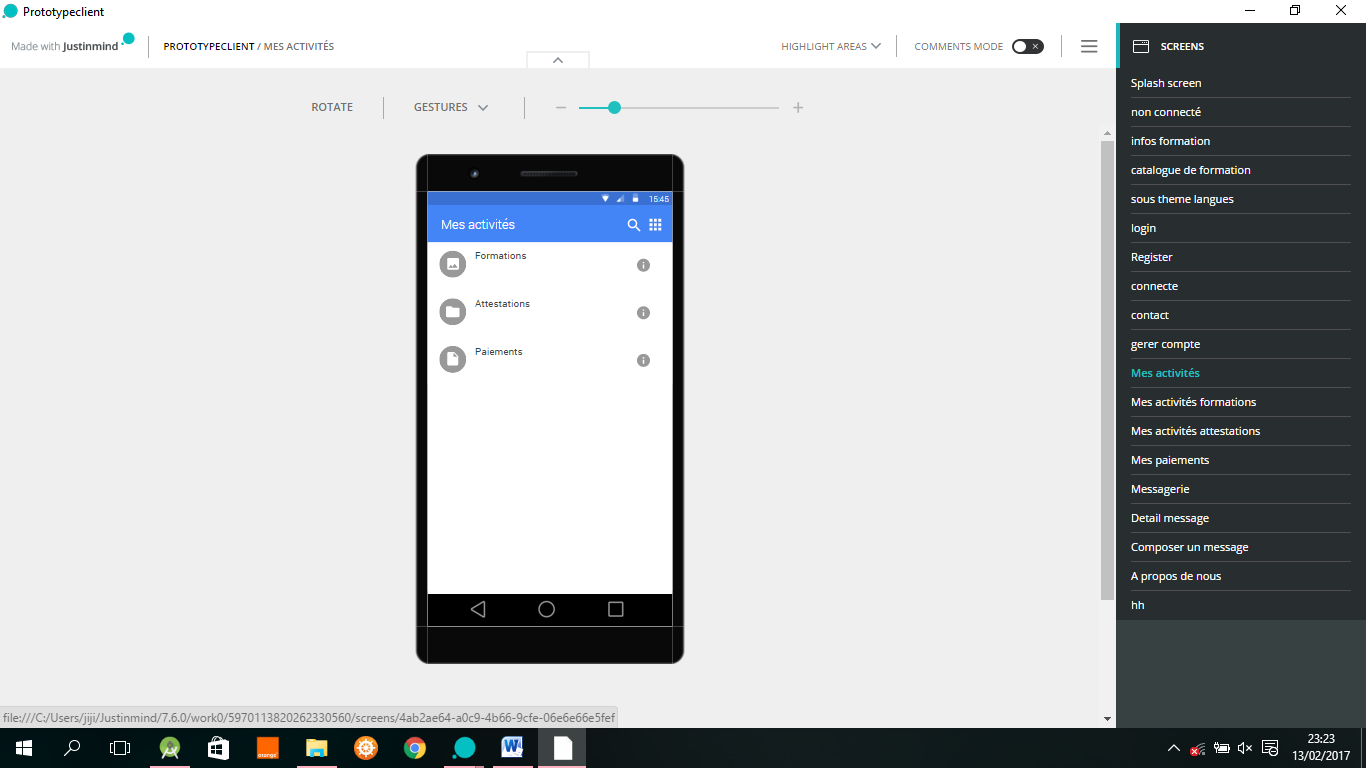
  

Figure 14: Profil Figure 15: Messagerie Figure 16: Les activités

* **Pour l’administration**

Une application pour l’administrateur qui lui permet de gérer les clients, les formateurs, les salles … et pour les formateurs qui lui offre la possibilité de réserver une salle d’avance et de consulter ses planning de formations.

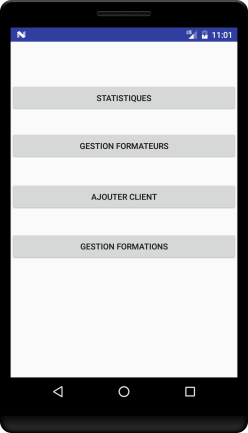
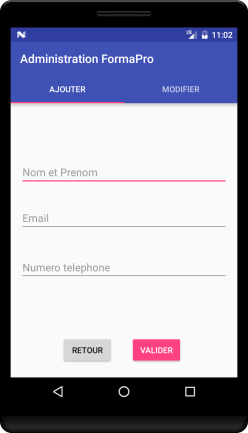
  

Figure 17 Login Figure 18 Dashboard Figure 19 Gestion des formateurs

* + - * 1. Identifications des acteurs

Un acteur est l’idéalisation d’un rôle joué par une personne externe, un processus ou une chose qui interagit avec un système. Dans le cas de notre projet nous présenterons les acteurs suivant :

* **Le Mobinaute :** toute personne qui installe l'application sur son Smartphone Android. Il a la possibilité de consulter le catalogue de formations et les nouveautés du centre en hors ligne**.**
* **Le Client :** toute personne qui s’inscrit dans l'application et peut, par la suite, réserver des formations, gérer son profil, demander une formation spécifique en ligne…
* **Le formateur :** c’est un client particulier de l’application. C’est la personne assurant des formations au centre.
* **L’administrateur :** c'est le superviseur qui contrôle et rectifie l’application pour assurer le bon fonctionnement.
  + - * 1. Spécifications graphique

La charte graphique est l’ensemble des codes graphiques, colorés et formels crées pour la communication visuelle d’une entreprise afin de séduire les clients.

Les couleurs occupent une place précieuse, ils ont été inspirés du logo du centre.

Pour attirer les clients nous proposons d’adopter un style d’apparition des prix tenant compte de leurs psychologies. Selon la psychologie comportementale et la psychologie cognitive, les prix importants doivent apparaitre en petite taille. En effet, l’attention du client sera focalisée plus sur la description du produit que sur son prix.

* + - * 1. Spécifications des besoins en sécurité
* Cryptage des mots de passe

Pour augmenter la sécurité des mots de passe, on a recourt à utiliser l’algorithme de cryptage **MD5** qui transforme une chaine de caractères initiale en une nouvelle chaine de 32 caractères (128 bits) supposée unique.

Ainsi, avec ce système, si un individu malintentionné arrivait à accéder à la base de données, il arriverait seulement à récupérer le mot de passe crypté et non le précieux sésame.

* Le format des id

Les identifiants sont créés par une méthode (**format\_ref**) qu’on a créé spécifiquement pour formaterla référence d’un client en exploitant la date et l’heure du système. Cette référence sera unique pour chaque client vu que nous estimons qu’il est très rare d’insérer deux clients simultanément à une seconde prés.

* + - 1. Etude technique

L’objectif de cette section est de mettre l’accent sur l’architecture globale de notre application ainsi que la définition des outils de réalisation.

* + - * 1. L’architecture adoptée

Avant d’entamer tout projet, une phase de définition architecturale est cruciale pour assurer le bon fonctionnement de tout système et l’adéquation avec son environnement matériel et logiciel c’est pourquoi nous avons choisi d’adopter une architecture orienté service qui va assurer le bon déploiement de notre solution.

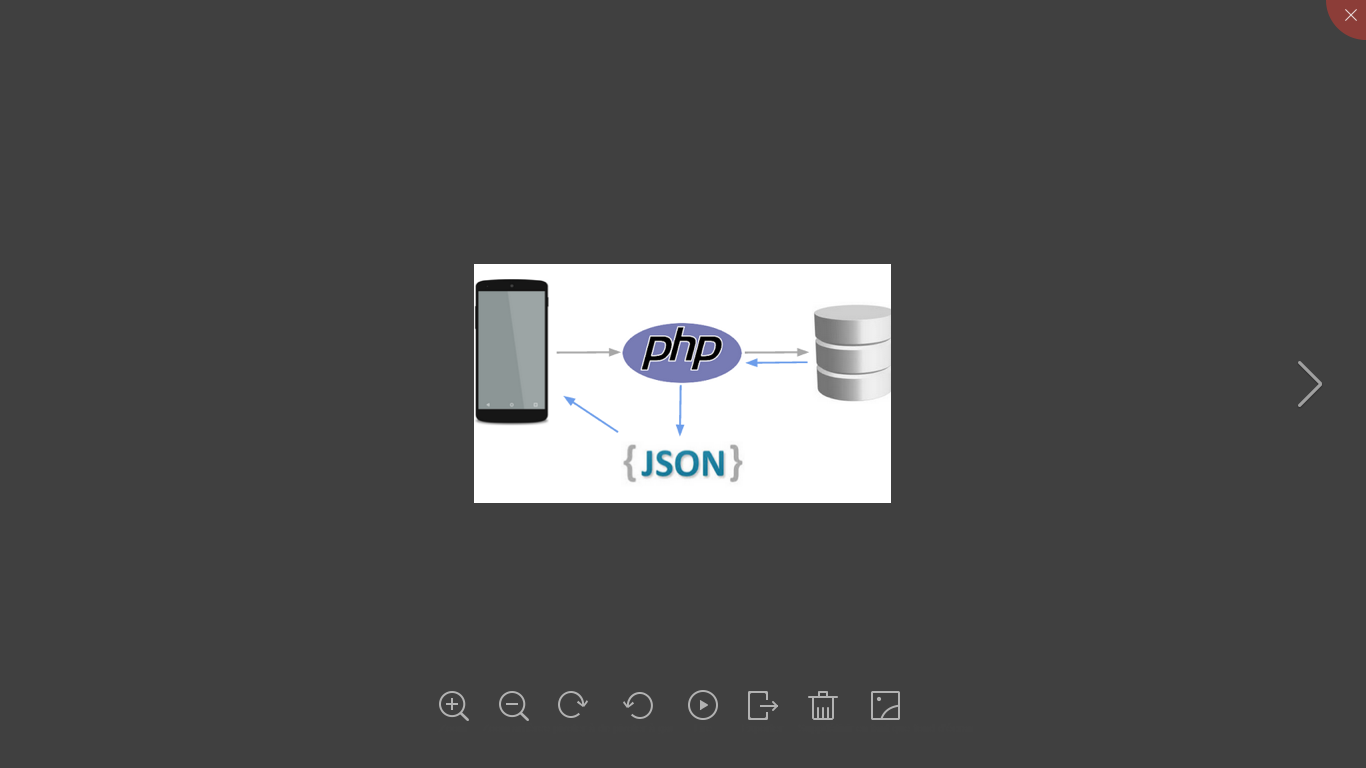


Figure 20 Architecture adoptée

* **Diagramme de déploiement**

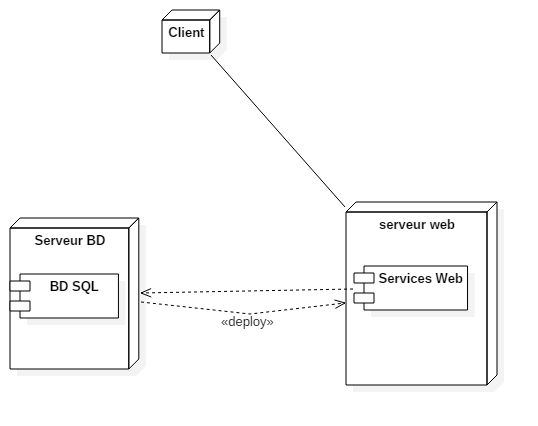


Figure 21 Diagramme de deploiement

* **Diagramme des classes**

Le diagramme de classe est un des diagrammes statiques d’UML. Il permet de décrire la structure interne d’un système tout en montrant ses différents objets et les relations entre eux.

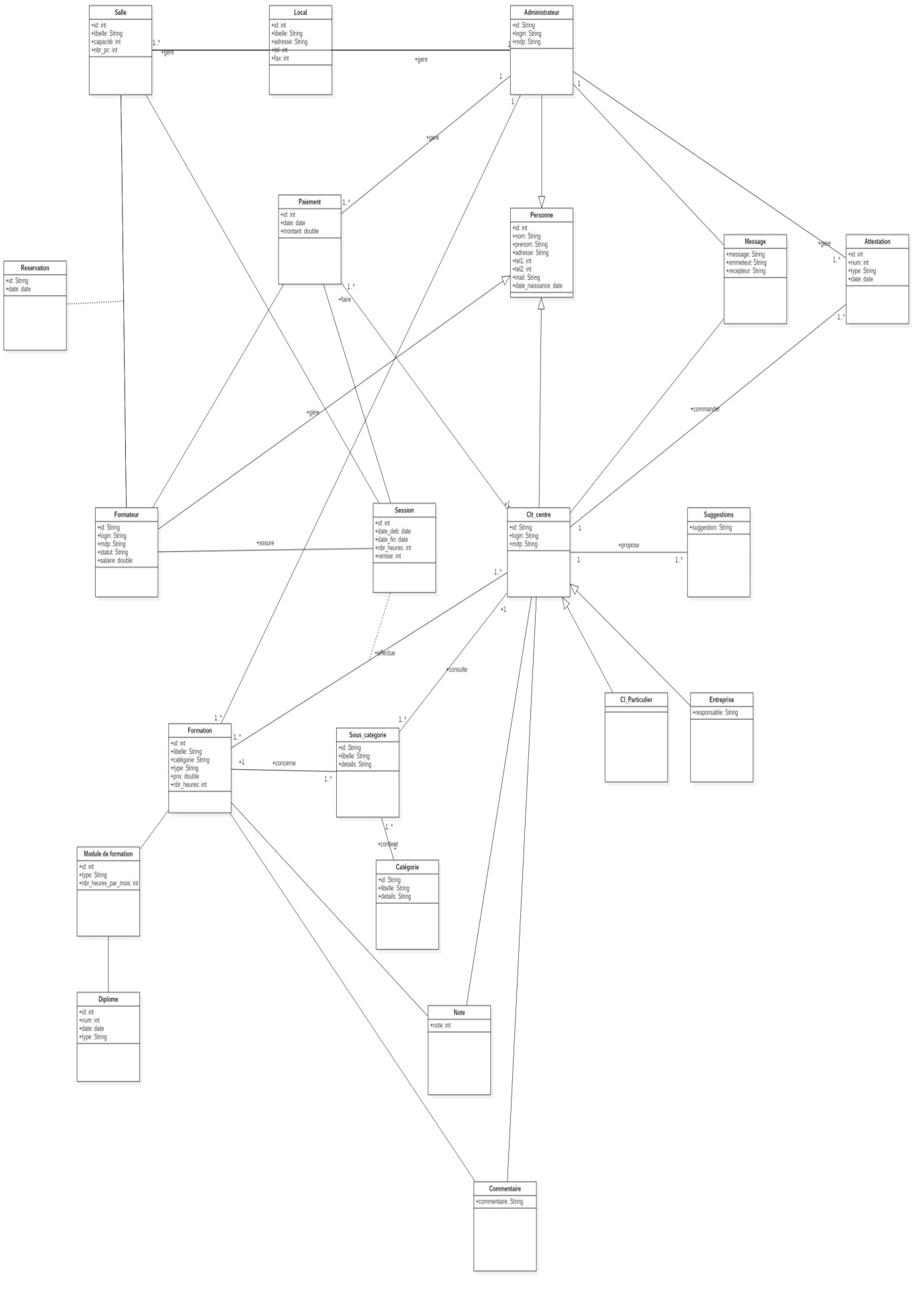


Figure 22 Diagramme des classes

* + - * 1. Environnement du travail
* Environnement matériel

Pour le développement de notre application, nous avons utilisé :

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Modèle | Processeur | RAM | Disque Dur | Système d’exploitation |
| HP | Intel I3 | 4 Go | 500 Go | Windows 10 |
| Huawei Y3c |  | 2 Go | 2 Go | Android 4.4 (Kitkat) |
| Dell | Intel i5 | 6 Go | 500 Go | Windows 8.1 |

Tableau 3 : Matériels utilisés

* Environnement logiciel

Les logiciels utilisés pour l’implémentation de notre solution sont les suivants

Android Studio

Figure : Android Studio

Android Studio est un environnement de développement pour développer des applications Android. Il est basé sur IntelliJ IDEA.

Just In mind

Figure 24 : Just In Mind

Just in mind est un logiciel de maquettage qui nous a aidés à préparer les prototypes.

Easy PHP DevServer 14.1 VC11 :



Figure 25 Easy PHP

Easy PHP est une plateforme de développement Web, permettant de faire fonctionner localement (sans se connecter à un serveur externe) des scripts PHP. Easy PHP n'est pas en soi un logiciel, mais un environnement comprenant : deux serveurs (un serveur web Apache et un serveur de bases de données MySQL), un interpréteur de script (PHP), ainsi qu'une administration SQL PHPMyAdmin. Il dispose d'une interface d'administration permettant de gérer les alias (dossiers virtuels disponibles sous Apache), et le démarrage/arrêt des serveurs. Il permet donc d'installer en une seule fois tout le nécessaire au développement local du PHP. Par défaut, le serveur Apache crée un nom de domaine virtuel (en local) 127.0.0.1 ou localhost. Ainsi, quand on choisit « Web local » dans le menu d'EasyPHP, le navigateur s'ouvre sur cette URL et affiche la page index.php de ce site qui correspond en fait au contenu du dossier www d'EasyPHP.

StarUML



Figure 26: StarUML

StarUML est un logiciel de modélisation UML (Unified Modeling Language) open source qui peut remplacer dans bien des situations des logiciels commerciaux et coûteux comme Rational Rose ou Together. Étant simple d’utilisation, nécessitant peu de ressources système, supportant UML 2, ce logiciel constitue une excellente option pour une familiarisation à la modélisation. Cependant, seule une version Windows est disponible.

Adobe Dreamweaver CS6



Figure 27 Adobe Dreamweaver CS6

Adobe Dreamweaver est un éditeur de site web de type WYSIWYG.

Nous avons l’utilisé dans notre projet afin de développer le web service qui assure la transmission des données entre l’application mobile et la base de données.

* Langages utilisés

Les langages de programmations utilisés pour le développement de notre application sont les suivants :

Java

Figure 28 : JAVA

Le langage Java est un langage de programmation informatique orienté objet créé par James Gosling et Patrick Naughton, employés de Sun Microsystems, avec le soutien de Bill Joy (Cofondateur de Sun Microsystems en 1982), présenté officiellement le 23 mai 1995 au SunWorld.

Php



Figure 29 php

Le **PHP** est un langage informatique utilisé sur l'internet. Le terme PHP est un acronyme récursif de "*PHP: Hypertext Preprocessor*".

Nous avons utilisé ce langage pour produire les scripts de web service et pour tester son bon fonctionnement.

Conclusion

Dans ce chapitre nous avons décrit l’organisme d’accueil, le contexte du projet et le Framework adopté. Ensuite, nous avons effectué une étude agile qui se voit essentielle dans notre application du Framework SCRUM. Puis, une étude technique où nous avons abordé l’architecture adoptée et enfin une description de l’environnement matériel et logiciel. Dans le chapitre suivant, nous allons détailler le premier lot de livraison qui est la Release 1.

* 1. Release 1
     1. Introduction

Suite au recensement des besoins sous forme de cas d’utilisations bien structurés au niveau du Product Backlog, et comme convenu dans le chapitre précédent, il est temps d’expliciter chaque release.

Dans le présent chapitre, nous allons détailler la Release 1 tout en montrant les différentes phases de sa réalisation ainsi que les tests effectués.

* + - 1. Organisation des Sprints

Cette Release contient les deux sprints :

* Sprint 1 :
* Authentification
* Inscription
* Consultation du catalogue des formations
* Inscription a une formation
* Consultation des informations du centre
* Consultation de page contact
* Sprint 2 :
* Profil
* Suivi des activités
* Consultation des formations
* Détails d’une formation
* Inscription à une formation

Figure 30 : RELEASE 1

* + - 1. Sprint 1

Au sein de Scrum, le Sprint a une durée limitée de 2 à 4 semaines au cours desquels une version utilisable et livrable du logiciel est créée. Il est préférable que les Sprints gardent une durée constante tout au long du développement et chaque nouveau Sprint n’est lancé qu’après la fin du prédécesseur afin d’éviter le chevauchement des Sprints.

« Les Sprints sont constitués de la planification du Sprint (Sprint Planning), des mêlées quotidiennes (Daily Scrums), des activités de développement, de la revue du Sprint (Sprint Review) et de la rétrospective du Sprint (Sprint Retrospective). »

Avant de commencer le Sprint nous avons défini son but en répondant à la question « pourquoi faisons-nous ce Sprint ? ».

* + - * 1. Sprint Planning

Le Sprint Planning est un travail collaboratif de la Scrum Team. Il est l’étape initiale de tout Sprint et il a duré 4 heures, au cours desquelles nous avons fixé le Sprint Goal et le Sprint Backlog.

* Sprint Goal

Le but de ce Sprint est de réaliser un espace pour les utilisateurs non connectés, d’avoir la possibilité d’en afficher un catalogue pour les formations assurées et une carte géographique qui localise le centre et la possibilité d’effectuer un appel vers le centre.

* Sprint Backlog

Une fois le but de notre Sprint défini, il est temps de décider quelles sont ses fonctionnalités et leurs degrés d’importance. Le tableau ci-dessous résume le Backlog de notre premier sprint :

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nom du sprint | Taches | Degré de difficulté | Priorité | Semaine |
| Sprint 1 | Authentification | 1 | Élevée | 1 |
| Inscription | 1 | Élevée | 1 |
| Consultation des informations du centre | 1 | Élevée | 1 |
| Consultation de page contact | 1 | Élevée | 1 |
| Consultation du catalogue des formations | 1 | Élevée | 1 |

Tableau 4 Backlog du «Sprint 1 »

* + - * 1. Conception du Sprint 1

Nous allons, par la suite, présenter la partie analyse et conception afin de mieux détailler le fonctionnement de notre système.

* Diagramme de cas d’utilisation du Sprint 1

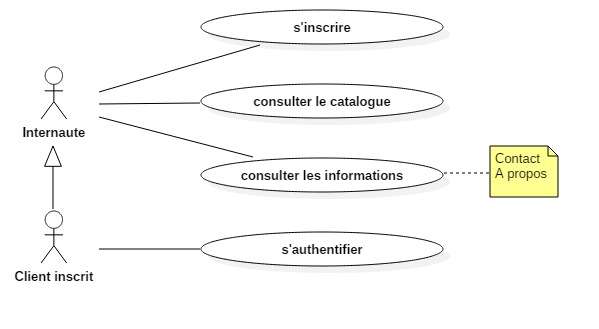


Figure 31: Diagramme de cas d'utilisation du sprint1

* Description textuelle du diagramme de cas d’utilisation «S’inscrire »

|  |  |
| --- | --- |
| Cas d’utilisation : S’inscrire | |
| Acteur | Internaute |
| Déclencheur | L’internaute demande de s’inscrire |
| Flot d’information | |  | | --- | | 1. L’utilisateur entre dans la page d’inscription  2. L’utilisateur entre son e-mail et le mot de passe confirmé  3. Le système vérifie la validité des informations entrées  4. Le système ajoute l’utilisateur à la base de données et lui confirme la création de son compte | |
| Exception | |  | | --- | | Si les informations entrées par l’utilisateur ne sont pas valide ou vide, le système lui affiche un message d’erreur lui informant qu’il doit vérifier ses informations. | |

Tableau 5 Description textuelle du diagramme de cas d’utilisation «S’inscrire»

* Description textuelle du diagramme de cas d’utilisation «S’authentifier »

|  |  |
| --- | --- |
| Cas d’utilisation : S’authentifier | |
| Acteur | Client inscrit |
| Déclencheur | Le client demande de s’authentifier |
| Flot d’information | |  | | --- | | 1. L’utilisateur entre dans la page d’authentification  2. L’utilisateur remplit le formulaire avec l’ensemble des informations nécessaires à son identification  3. Le système vérifie la validité des informations entrées | |
| Extension | 1. L’utilisateur n’as pas saisie les bons identifiants 2. Le système renvoie un message d’erreur et signale à l’utilisateur de recommencer 3. L’utilisateur n’existe pas dans la base de données |

Tableau 6 Description textuelle du diagramme de cas d’utilisation «S’authentifier»

* Diagramme des séquences de Sprint 1

En nous référant aux descriptions textuelles dans la section précédente, nous présentons les diagrammes des séquences adéquats.

Diagramme de séquence « s’inscrire » :

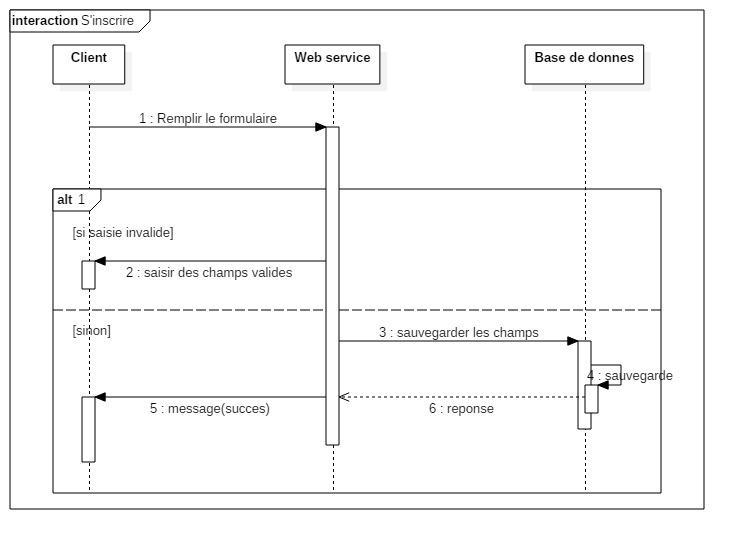


Figure 32: Diagramme de séquence « S'inscrire »

Diagramme de séquence « s’authentifier » :

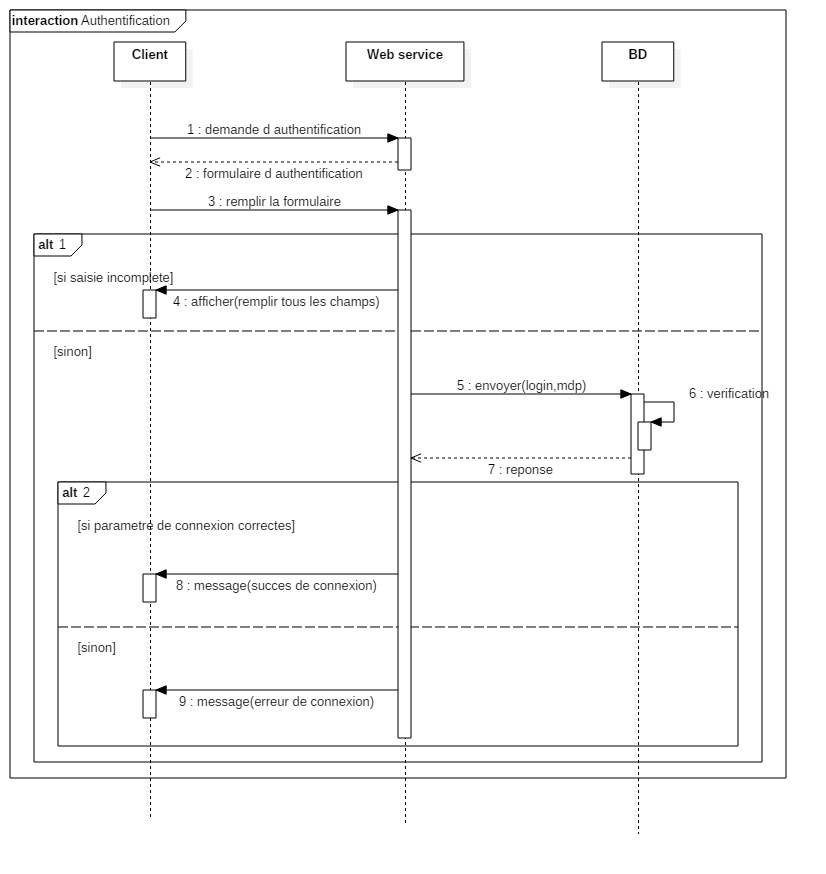


Figure 33: Diagramme de séquence « S'authentifier »

Diagramme de séquence « consulter le catalogue» :

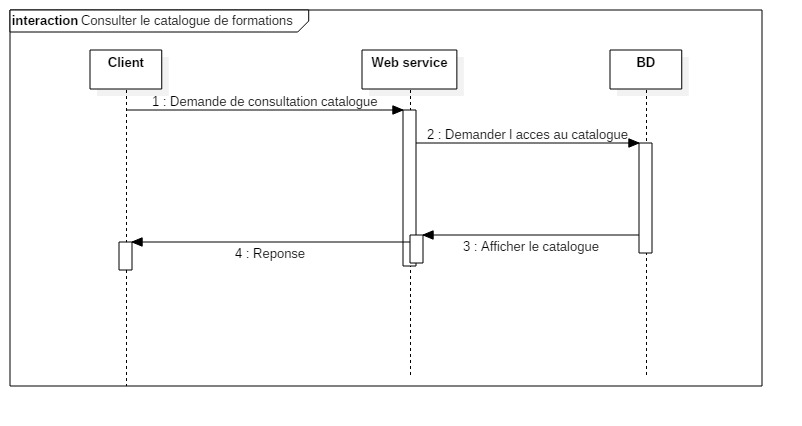


Figure 34: Diagramme de séquence « Consulter le catalogue

* + - * 1. Réalisation et Test de l’incrément

A ce stade, nous exposons quelques interfaces de notre application.

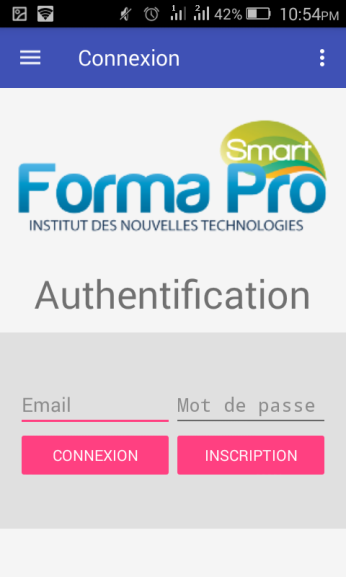
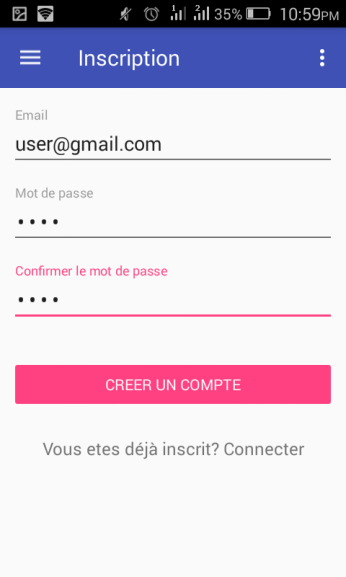
  

Figure 35 Connexion Figure 36 Inscription Figure 37 Localisation du centre

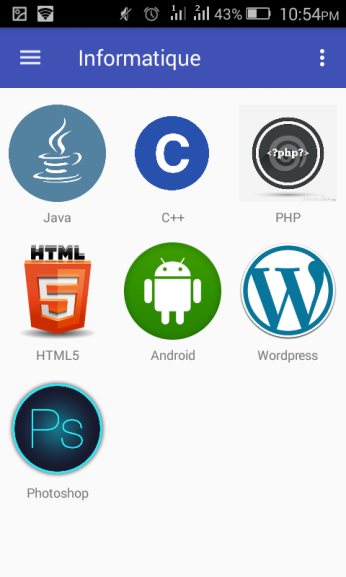
  

Figure 38 A propos Figure 39 Catalogue Figure 40 Catégorie informatique

* + - * 1. Sprint Review & Rétrospective
* Sprint Review

Le but de la revue de Sprint (Sprint Review) est de montrer ce qui a été réalisé pendant le Sprint afin d’en tirer les conséquences pour la suite du projet.

Le schéma présente les étapes du Sprint Review :

Figure 41 : Schéma du Sprint Review

Le Sprint Review se conclu avec le Sprint Rétrospective.

* Sprint Rétrospective

Après avoir effectué le Sprint Review, l’équipe SCRUM se réunie une dernière fois dans le Sprint afin de dégager les points à faire évoluer.

Nous allons fournir pour chaque Sprint un plan d’amélioration.

Pour ce Sprint, nous avons choisi d’améliorer ces points :

* Amélioration au niveau de la relation avec le Product Owner et le Scrum Master.
* Amélioration au niveau de la communication dans l’équipe SCRUM.
  + - 1. Sprint 2

En partant sur le même principe que le Sprint précédent, nous commençons par définir le but de notre second sprint.

* + - * 1. Sprint Planning

Tout comme le Sprint précédant, nous avons démarré ce Sprint avec une réunion du Sprint Planning qui a duré 4 heures au cours desquelles nous avons réalisé le Sprint Goal et le Sprint Backlog.

* Sprint Goal

L’objectif de ce Sprint est de réaliser un module permettant à l’utilisateur d’accéder à son profil, consulter les détails d’une formation et d’avoir la possibilité de s’inscrire à cette formation.

* Sprint Backlog

Une fois que le but de notre Sprint est défini, il est temps de décider quel item /histoire inclure dans ce dernier.

Le tableau ci-dessous résume le Backlog de notre second Sprint.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nom | Description | Complexité | Priorité | Semaine |
| Profil | En tant qu’un utilisateur inscrit je dois gérer mon compte. | moyenne | Élevée | 2 |
| Suivi des activités | En tant que client je veux suivre mes activités. | moyenne | Élevée | 2 |
| Consultation des formations | En tant que formateur je dois pouvoir consulter mes formations (assurées et à venir). | moyenne | Élevée | 2 |
| Détails d’une formation | En tant que client je veux voir les détails d’une formation. | moyenne | Élevée | 2 |
| Inscription a une formation | En tant que client je dois pouvoir m’inscrire à une formation. | moyenne | Élevée | 2 |

Tableau 7 Backlog du « Sprint 2 »

* + - * 1. Conception du Sprint 2

Pour la spécification fonctionnelle de ce Sprint, nous commençons par le diagramme de cas d’utilisation.

* Diagramme de cas d’utilisation du Sprint 2

Dans la figure suivante nous illustrons le diagramme de cas d’utilisation global du second Sprint.

En respectant toujours le même principe que le Sprint précédent, nous avons découpé certaines histoires/items en un ensemble de tâches.

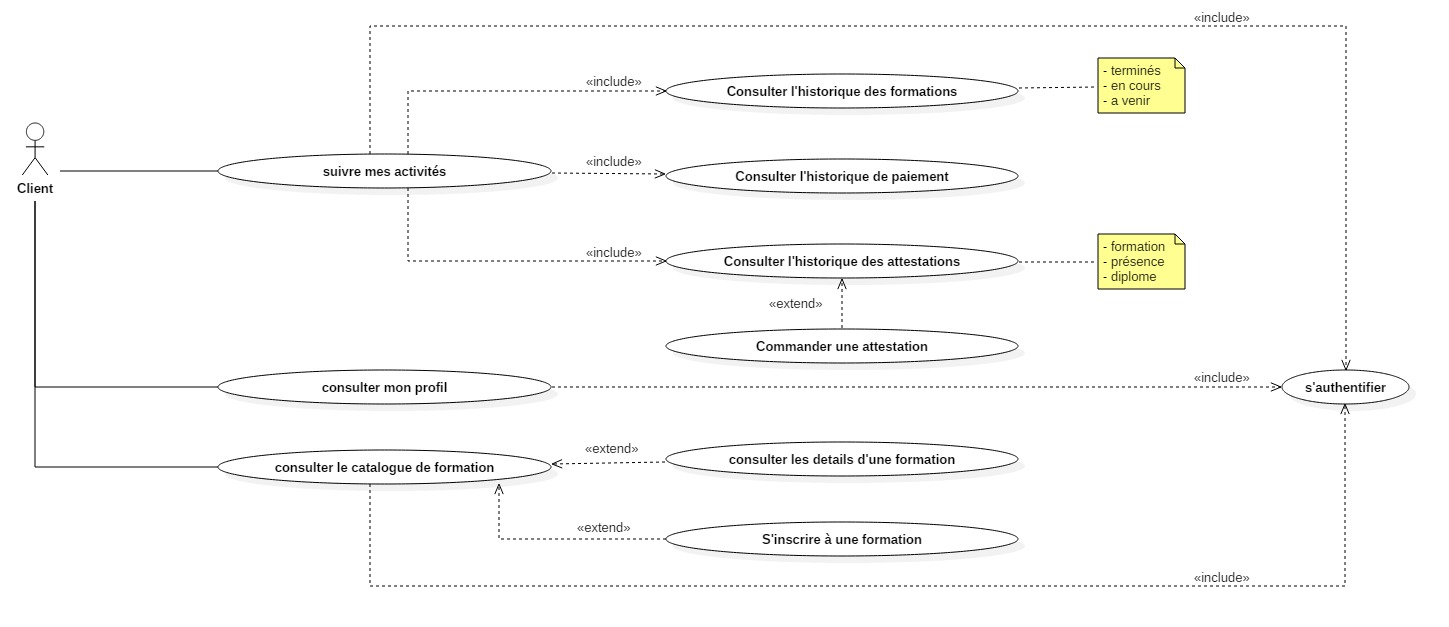


Figure 42: Diagramme de cas d'utilisation du sprint 2

|  |  |
| --- | --- |
| Cas d’utilisation : Suivre mes activités | |
| Acteur | Client |
| Déclencheur | Le client demande la page « mes activités » |
| Flot d’information | |  | | --- | | 1. L’utilisateur a plusieurs choix :  * Suivi des formations terminés, en cours ou à venir. * Suivi de paiement * Suivi des attestations de présence, de formation ou diplôme, comme il peut aussi commander une attestation | |

Tableau 8 Description textuelle du diagramme de cas d’utilisation « Suivre mes activités »

|  |  |
| --- | --- |
| Cas d’utilisation : Consulter les détails d’une formation | |
| Acteur | Client |
| Déclencheur | Le client consulte le catalogue des formations et choisit la catégorie qui l’intéresse. Ensuite, il clique sur la formation dont il veut consulter les détails |
| Flot d’information | |  | | --- | | 1. L’utilisateur a plusieurs choix :  * Suivi des formations terminés, en cours ou à venir. * Suivi de paiement * Suivi des attestations de présence, de formation ou diplôme | |

Tableau 9 Description textuelle du diagramme de cas d’utilisation «Consulter les détails d’une formation»

|  |  |
| --- | --- |
| Cas d’utilisation : S’inscrire à une formation | |
| Acteur | Client |
| Déclencheur | Le client consulte le catalogue de formation et choisit la catégorie qui l’intéresse. Après, il clique sur la formation à laquelle il veut s’inscrire. |
| Flot d’information | |  | | --- | | - Lorsque le client clique sur le bouton «s’inscrire » un test d'authentification sera effectué :  Si le client est déjà authentifié le couple (id\_client/id\_formation) sera ajouté pour remplir une ligne dans la table « Réservation »  Sinon, si le client n'est pas déjà authentifié il sera redirigé vers la page d'authentification et après il pourra s’inscrire à une formation. | |

Tableau 10 Description textuelle du diagramme de cas d’utilisation «S’inscrire à une formation»

* Diagramme des séquences du Sprint 2

En nous référant aux descriptions textuelles dans la section précédente, nous présentons les diagrammes de séquences adéquats.

Diagramme de séquence « suivre mes activités» :

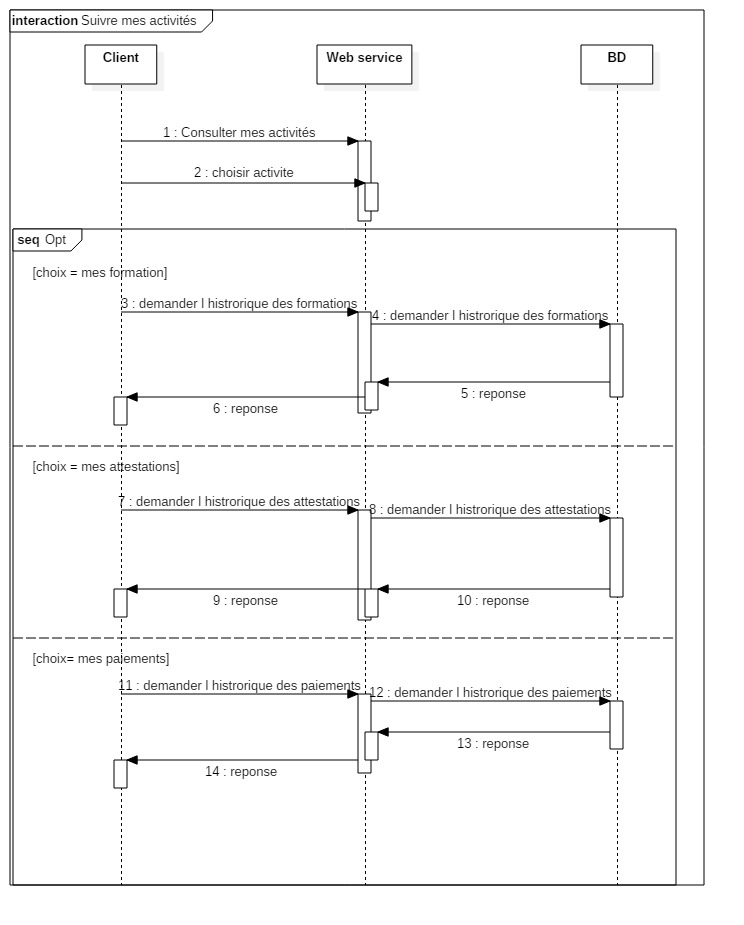


Figure 43 A Diagramme de séquence « suivre mes activités»

* + - * 1. Réalisation de l’incrément

Nous passons à exposer quelques interfaces de notre application.

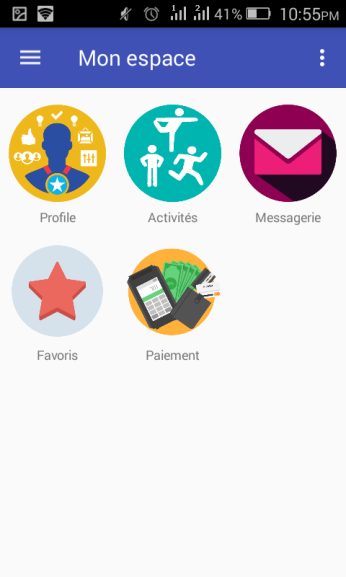
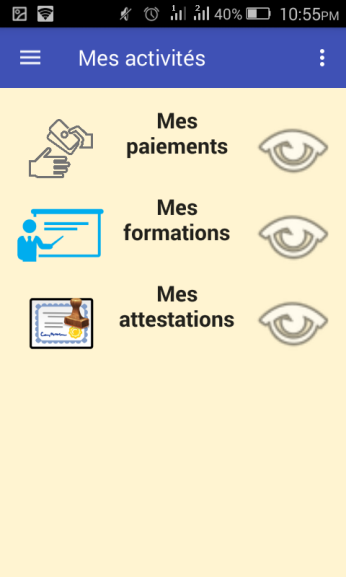
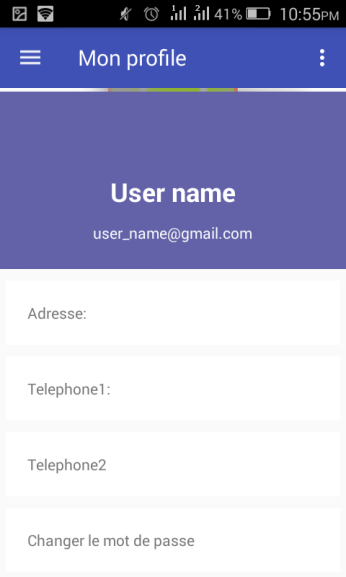
  

Figure 44 Espace du client Figure 45 Les activités Figure 46 Profil

* + - * 1. Sprint Review & Rétrospective
* Sprint Review

Durant le Sprint Review, nous avons présenté l’incrément réalisé avec toute l’équipe Scrum.

* Sprint Rétrospective

Pour ce Sprint, nous avons choisi d’améliorer ces points :

* Amélioration au niveau de la communication.
* Amélioration des techniques de développement

Conclusion

Au cours de ce chapitre, nous avons mis comme objectif la réalisation des Sprints 1 et 2 qui ont aboutis à la livraison de la première Release. Nous allons enchaîner avec les Sprints suivants.

* 1. Release 2

* + 1. Introduction

Après avoir entamé le premier release de notre système, nous pouvons maintenant passer à la seconde. En effet, les Framework Agiles, et SCRUM en particulier, sont caractérisés par un rythme régulier. Tout au long du chapitre nous aurons trois Sprints ayant la même vélocité que les sprints précédents, et nous allons traiter les histoires utilisateurs de ces derniers pour avoir à la fin de cette Release une solution complète, livrable et fonctionnelle.

* + - 1. Organisation des Sprints

Cette Release contient les deux sprints :

* Sprint 3 :
* Gestion du catalogue
* Gestion des clients
* Gestion des formateurs
* Sprint 4 :
* Gestion des salles
* Réservation d’une salle

Figure 47 : Release 2

* + - 1. Sprint 3

En partant du même principe que les Sprints précédents, nous commençons par définir le but de notre premier Sprint pour ce release.

* + - * 1. Sprint Planning

Au cours du Sprint Planning, nous avons fixé le Sprint Goal et le Sprint Backlog.

* Sprint Goal

L’objectif de ce Sprint est de réaliser un module permettant à l’administrateur de gérer le catalogue, les clients et les formateurs

* Sprint Backlog

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nom du sprint | Taches | Degré de difficulté | Priorité | Semaine |
| Sprint 3 | Gestion du catalogue | 1 | Élevée | 3 |
| Gestion des clients | 1 | Élevée | 3 |
| Gestion des formateurs | 1 | Élevée | 3 |

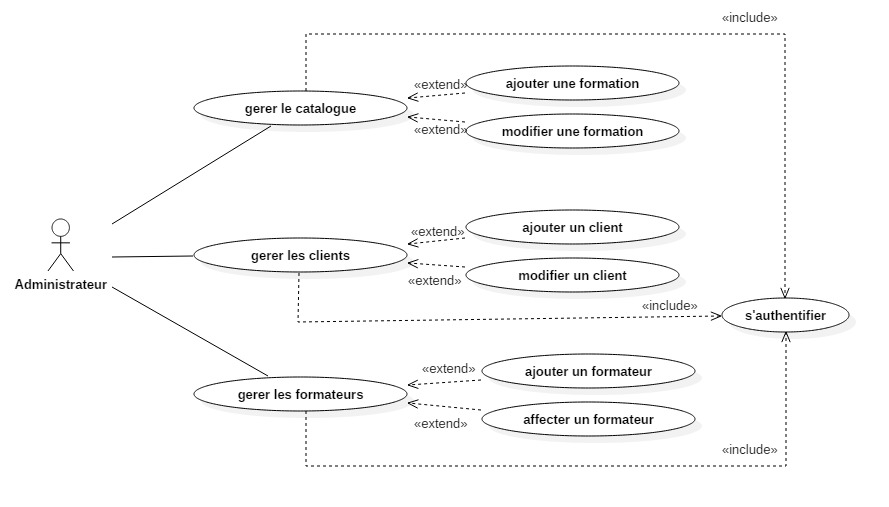
Tableau 11 Sprint Backlog du Sprint 3

* + - * 1. Conception du Sprint 3

Pour la spécification fonctionnelle de ce Sprint, nous commençons par le diagramme des cas d’utilisation.

* Diagramme de cas d’utilisation du Sprint 3

Dans la figure suivante, nous illustrons le diagramme des cas d’utilisation global du troisième Sprint.

En respectant toujours le même principe que les Sprints précédents, nous avons découpé certaines histoires en un ensemble de tâches.

Nous allons maintenant décrire ces cas d’utilisation en identifiant les acteurs et les différents scénarios possibles.

Figure : Diagramme de cas d'utilisation du Sprint 3

* Description textuelle du diagramme de cas d’utilisation «Gérer le catalogue »

|  |  |
| --- | --- |
| Cas d’utilisation : Gérer le catalogue | |
| Acteur | Administrateur |
| Déclencheur | L’administrateur demande de gérer le catalogue |
| Flot d’information | |  | | --- | | 1. L’administrateur accède à la page de gestion du catalogue  2. L’administrateur a deux choix à faire:   * Modifier une formation * Ajouter une nouvelle formation   3. Le système vérifie la validité des informations entrées | |

Tableau 12 DESCRIPTION TEXTUELLE DU DIAGRAMME DE CAS D'UTILISATION << Gérer le catalogue >>

* Description textuelle du diagramme de cas d’utilisation « Gérer les clients »

|  |  |
| --- | --- |
| Cas d’utilisation : Gérer les clients | |
| Acteur | Administrateur |
| Déclencheur | L’administrateur demande de gérer les clients |
| Flot d’information | |  | | --- | | 1. L’administrateur accède à la page de gestion des clients  2. L’administrateur peut :   * Modifier un client * Ajouter un nouveau client   3. Le système vérifie la validité des informations entrées | |

Tableau 13 DESCRIPTION TEXTUELLE DU DIAGRAMME DE CAS D'UTILISATION « Gérer les clients »

* Description textuelle du diagramme de cas d’utilisation « Gérer les formateurs »

|  |  |
| --- | --- |
| Cas d’utilisation : Gérer les formateurs | |
| Acteur | Administrateur |
| Déclencheur | L’administrateur demande de gérer les formateurs |
| Flot d’information | |  | | --- | | 1. L’administrateur accède à la page de gestion des formateurs  2. L’administrateur peut :   * Ajouter un nouveau formateur * Affecter un formateur à une formation   3. Le système vérifie la validité des informations entrées | |

Tableau 14 DESCRIPTION TEXTUELLE DU DIAGRAMME DE CAS D'UTILISATION « Gérer les formateurs »

* + - * 1. Réalisation et Test de l’incrément

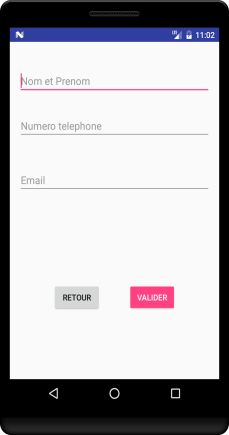
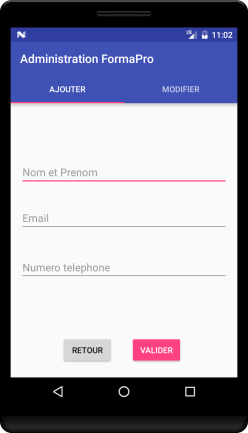
 

Figure 49 Ajout d'un client Figure 50 Gestion des formateurs

* + - * 1. Sprint Review & Rétrospective
* Sprint Review

Lors de ce Sprint Review l’équipe de développement a discuté les tâches accomplies, les problèmes rencontrés et la procédure de résolution de ces problèmes.

* Sprint Rétrospective

Pour ce Sprint, nous avons choisi d’améliorer ces points :

* Amélioration au niveau de la communication.
* Amélioration des techniques de développement
  + - 1. Sprint 4

En partant du même principe que les Sprints précédents, nous commençons par définir le but de notre deuxième Sprint pour ce release.

* + - * 1. Sprint Planning
* Sprint Goal

L’objectif de ce Sprint est de réaliser un module permettant à l’administrateur de s’authentifier et de gérer les salles.

* Sprint Backlog

Une fois que le but de notre Sprint est défini, il est temps de décider quelles histoires inclure dans ce dernier.

Le tableau ci-dessous résume le Backlog de notre quatrième Sprint.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nom du sprint | Taches | Degré de difficulté | Priorité | Semaine |
| Sprint 4 | Gestion des salles | 1 | Élevée | 4 |
| Réservation d’une salle | 1 | Élevée | 4 |

Tableau 15 Sprint Backlog du Sprint 4

* + - * 1. Conception du Sprint 4

Pour la spécification fonctionnelle de ce Sprint, nous commençons par le diagramme des cas d’utilisation.

* Diagramme de cas d’utilisation du Sprint 4

Dans la figure suivante nous illustrons le diagramme des cas d’utilisation global du quatrième Sprint.

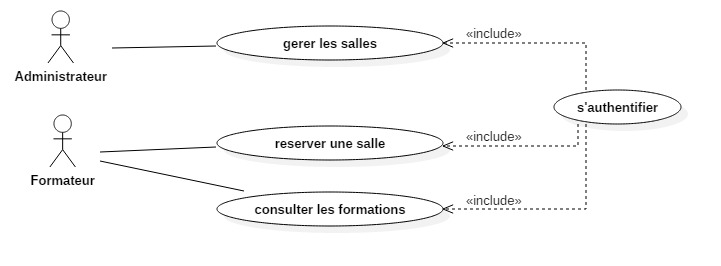
En respectant toujours le même principe que les Sprints précédents, nous avons découpé certaines histoires en un ensemble de tâches.

Figure 51 : Diagramme de cas d'utilisation du Sprint 4

Nous passons maintenant à la description des cas d’utilisations en identifiant les acteurs et les différents scénarios possibles.

* Description textuelle du cas d’utilisation « Gérer les salles»

|  |  |
| --- | --- |
| Cas d’utilisation : Gérer les salles | |
| Acteur | Administrateur |
| Déclencheur | L’administrateur demande de gérer les salles |
| Flot d’information | |  | | --- | | 1. L’administrateur accède à la page de gestion des salles 2. Le système vérifie la validité des informations entrées | |

Tableau 16 DESCRIPTION TEXTUELLE DU DIAGRAMME DE CAS D'UTILISATION <<GERER LES SALLES>>

* Description textuelle du cas d’utilisation « Réserver une salle »

|  |  |
| --- | --- |
| Cas d’utilisation : Réserver une salle | |
| Acteur | Formateur |
| Déclencheur | Le formateur demande de réserver une salle d’avance |
| Flot d’information | |  | | --- | | 1. Le formateur accède à la page de réservation des salles 2. Le système vérifie la validité des informations entrées | |

Tableau 17 DESCRIPTION TEXTUELLE DU DIAGRAMME DE CAS D'UTILISATION << Réserver une salle >>

* + - * 1. Sprint Review & Rétrospective
* Sprint Review

Après avoir démontré la réalisation de ce Sprint, nous passons maintenant au Sprint Review comme pour les Sprints précédents, durant ce dernier nous, équipe de développement, avons démontré le travail réalisé durant ce sprint et nous avons répondu aux questions sur l’incrément.

* Sprint Rétrospective

Durant le Sprint Retrospective de ce Sprint, nous avons décidé d’améliorer l’auto-organisation de l’équipe.

Conclusion

Tout au long de cette Release, nous avons réalisé les Sprints 3 et 4 pour aboutir à la fin aux incréments potentiellement fonctionnels qui satisfont parfaitement les demandes du Client.

* 1. Release 3

* + 1. Introduction

Après avoir terminé le deuxième release de notre système, nous passons au troisième et dernier release. Tout comme les deux releases précédents, nous allons traiter les histoires utilisateurs de ces derniers pour arriver à la fin à une solution complète, livrable et fonctionnelle.

* + - 1. Organisation des Sprints

Ce release contient les deux Sprints suivants :

* Sprint 5
* Messagerie
* Statistiques
* Sprint 6
* Noter une formation
* Proposer une formation
* Commenter une formation

Figure 52 Release 3

* + - 1. Sprint 5

En partant sur le même principe que les Sprints précédents, nous commençons par définir le but de notre premier Sprint pour ce release.

* + - * 1. Sprint Planning

Au cours du Sprint Planning de ce Sprint, nous avons fixé le Sprint Goal et le Sprint Backlog.

* Sprint Goal

L’objectif de ce Sprint est de réaliser un module permettant pour les deux acteurs client et administrateur de communiquer entre eux grâce à la messagerie. Et permettre à l’administrateur de consulter les statistiques.

* Sprint Backlog

Une fois que le but de notre sprint est défini, il est temps de décider quelles histoires inclure dans ce dernier.

Le tableau ci-dessous résume le backlog de notre cinquième sprint.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nom du sprint | Taches | Degré de difficulté | Priorité | Semaine |
| Sprint 5 | Messagerie | 1 | Élevée | 5 |
| Statistiques | 1 | Élevée | 5 |

Tableau 18 Sprint Backlog du Sprint 5

* + - * 1. Conception du Sprint 5

Nous commençons la conception de ce Sprint, comme chaque Sprint, par le diagramme de cas d’utilisation.

* Diagramme du cas d’utilisation du Sprint 5

Dans la figure suivante nous illustrons le diagramme des cas d’utilisation global du cinquième Sprint.

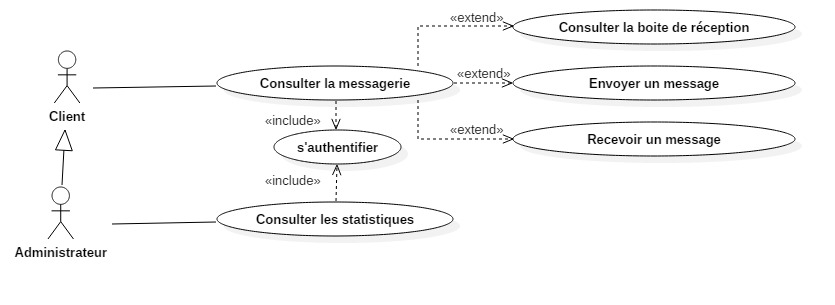
En respectant toujours le même principe que les Sprints précédents, nous avons découpé les histoires (items) choisies en un ensemble de tâches.

Figure 53 : Diagramme de cas d'utilisation DU << Sprint 5 >>

Nous allons maintenant décrire ces cas d’utilisation en identifiant les acteurs et les différents scénarios possibles.

* Description textuelle du diagramme de cas d’utilisation « Consulter la boîte de réception»

|  |  |
| --- | --- |
| Cas d’utilisation : Consulter la boite de réception | |
| Acteur | Administrateur / Client |
| Déclencheur | L’administrateur ou le client consulte sa boîte de réception |
| Flot d’information | |  | | --- | | 1. L’administrateur ou le client consulte sa boîte de réception | |
| Exception | L’administrateur ou le client quitte l’application |

Tableau 19 DESCRIPTION TEXTUELLE DU DIAGRAMME DE CAS D’UTILISATION «Consulter la boite de réception»

* Description textuelle du diagramme de cas d’utilisation « Envoyer un message»

|  |  |
| --- | --- |
| Cas d’utilisation : Envoyer un message | |
| Acteur | Administrateur / Client |
| Déclencheur | L’administrateur ou le client consulte sa boîte de réception |
| Flot d’information | |  | | --- | | 1. L’administrateur ou le client écrit un message 2. L’administrateur ou le client remplit le formulaire d’envoi qui contient l’adresse du récepteur 3. L’administrateur ou le client envoie le message | |
| Exception | L’administrateur ou le client n’a pas rempli tous les champs |

Tableau 20 DESCRIPTION TEXTUELLE DU DIAGRAMME DE CAS D’UTILISATION «Envoyer un message»

* Description textuelle du diagramme de cas d’utilisation « Recevoir un message »

|  |  |
| --- | --- |
| Cas d’utilisation : Recevoir un message | |
| Acteur | Administrateur / Client |
| Déclencheur | L’administrateur ou le client a reçu un message |
| Flot d’information | |  | | --- | | 1. L’administrateur ou le client ouvre le message reçu | |
| Exception | L’administrateur ou le client a quitté l’application |

Tableau 21 TABLEAU 21 DESCRIPTION TEXTUELLE DU DIAGRAMME DE CAS D’UTILISATION «Recevoir un message »

* + - * 1. Réalisation et Test de l’incrément



Figure 54 Messagerie

* + - * 1. Sprint Review & Rétrospective
* Sprint Review

Tout comme durant les Sprint précédents, le Sprint Review s’est conclu avec une démonstration de l’incrément terminé ainsi qu’à la réponse aux questions des différents intervenants dans notre application.

* Sprint Rétrospective

Pour ce Sprint, nous avons choisi d’améliorer ces points :

* Améliorer le langage utilisé entre les membres de l’équipe pour créer une entente entre eux.
* Chercher les moyens possibles pour régner l’harmonie dans l’équipe
  + - 1. Sprint 6

En partant du même principe que les Sprints précédents, nous commençons par définir le but de notre deuxième Sprint pour ce release.

* + - * 1. Sprint Planning

Au cours du Sprint Planning de ce Sprint, qui a duré 2 heures, nous avons fixé le Sprint Goal et le Sprint Backlog.

* Sprint Goal

L’objectif de ce Sprint est de permettre au client un ensemble de taches pour qu’il puisse noter, proposer ou commenter une formation d’une façon anonyme.

* Sprint Backlog

Une fois que le but de notre Sprint est défini, il est temps de décider quelles histoires inclure dans ce dernier.

Le tableau ci-dessous résume le Backlog de notre sixième Sprint.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nom du sprint | Taches | Degré de difficulté | Priorité | Semaine |
| Sprint 6 | Noter une formation | 1 | Élevée | 6 |
| Proposer une formation | 1 | Élevée | 6 |
| Commenter une formation | 1 | Élevée | 6 |

Tableau 22 Sprint Backlog du Sprint 6

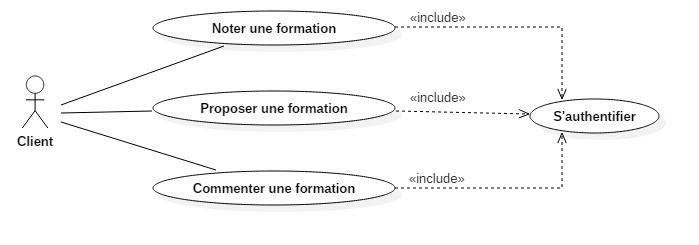
* + - * 1. Conception du Sprint

Nous commençons la conception de ce Sprint, comme chaque Sprint, par le diagramme de cas d’utilisation

* Diagramme de cas d’utilisation du Sprint 6

Dans la figure suivante nous illustrons le digramme des cas d’utilisation global du sixième Sprint.

En respectant toujours le même principe que les Sprints précédents, nous avons découpé certaines histoires en un ensemble de tâches.



Nous allons maintenant décrire ces cas d’utilisation en identifiant les acteurs et les différents scénarios possibles.

* + - * 1. Sprint Review

Tout comme durant les Sprint précédents, nous nous sommes réunis avec l’équipe Scrum et les parties prenantes afin de discuter les incréments résultants des Sprint 5 et 6 et ils ont été acceptés en totalité.

Conclusion

Dans de ce chapitre, nous avons traité les derniers Sprints qui ont aboutis à la dernière Release et donc la finalisation de notre projet.

Conclusion

L’objectif de notre projet de fin d’études était de concevoir et implémenter une application Android de gestion de centre de formation Forma Pro.

Le point de départ de la réalisation de ce projet était une récolte des informations nécessaires

pour dresser un étude de l’existant et présenter un aperçu sur la problématique.

Par la suite, nous nous sommes intéressés à l’analyse et la spécification des besoins qui nous a permis de distinguer les différents acteurs interagissant avec l’application visée.

L’objectif de la partie suivante était la conception détaillée, dans laquelle nous avons fixé la structure globale de l’application.

Le dernier volet de notre projet était la partie réalisation qui a été consacrée à la présentation des outils de travail et les interfaces les plus significatives de notre application.

L’apport de ce travail a été d’une importance très considérable, en effet, il nous a permis : de suivre une méthodologie de travail bien étudiée, d’approfondir nos connaissances dans le monde de développement des applications et de bien nous exercer sur la méthodologie SCRUM et l’Android.

La réalisation d’un tel projet, nous a permis d’apprendre et de toucher de près à une partie de divers aspects du métier de développeur et de celui de concepteur.

Netographie

|  |  |  |
| --- | --- | --- |
| [www.developers.google.com/](http://www.developers.google.com/) | | (visité en Février 2017) |
| [www.stackoverflow.com/](http://www.stackoverflow.com/) | | (visité en Février 2017) |
| [www.github.com/](http://www.github.com/) | | (visité en Février 2017) |
| [www.youtube.com/](http://www.youtube.com/) | | (visité en Février 2017) |
| [www.openclassrooms.com](http://www.openclassrooms.com) | | (visité en Mars 2017) |
| [www.tutos-android.com/](http://www.tutos-android.com/) | | (visité en Avril 2017 ) |
| [www.staruml.sourceforge.net](http://www.staruml.sourceforge.net) | | (visité en Mars 2017) |
| [www.play.google.com/store](file:///C:\Users\jiji\Downloads\www.play.google.com\store) | | (visité en Mars 2017) |
| [www.json.org](http://www.json.org) | | (visité en Mai 2017 ) |
| [fr.wikipedia.org](file:///C:\Users\jiji\Downloads\fr.wikipedia.org) | | (visité en Mai 2017) |
| [www.vogella.com](file:///C:\Users\jiji\Downloads\www.vogella.com) | | (visité en Mai 2017) |
| [www.ibm.com](file:///C:\Users\jiji\Downloads\www.ibm.com) |  | (visité en Mai 2017) |
| [www.java2s.com](file:///C:\Users\jiji\Downloads\www.java2s.com) | | (visité en Mai 2017) |
| [www.php.net](http://www.php.net) | | (visité en Mai 2017) |
|  | | |

Annexes

Cycle de vie d’une application android

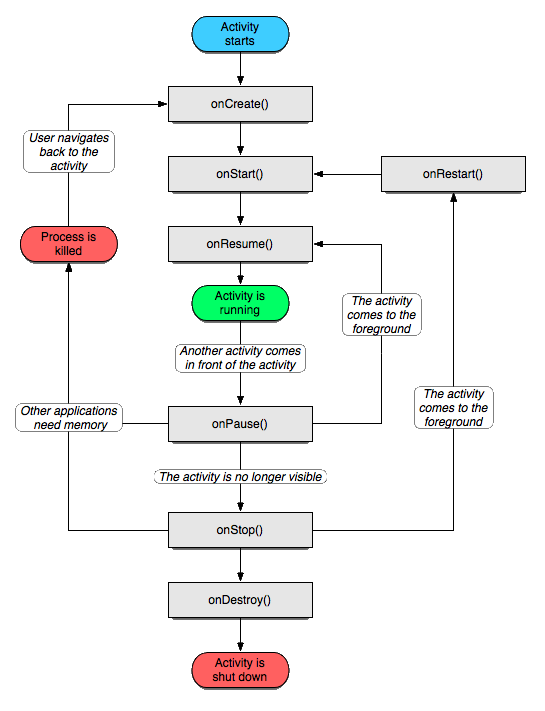


Figure 56 Cycle de vie d’une application Android