



Master 1 Informatique Génie Logiciel

Angular 2sxp

Année Universitaire 2017-2018

Réalisé par :

ALOUANI Salah
BOUSTANI Sara
OLLIER Guillaume

Encadré par :

Pablo Arrighi

Résumé

Le projet que présente ce document s'inscrit dans le cadre du projet de l'UE du génie logiciel dans notre cursus en Master Informatique au sein de l'université Aix-Marseille Luminy. Notre projet consiste à améliorer le code déjà existant en Angular, en le migrant vers Angular 4 qui est la nouvelle version plus adaptable niveau navigateurs web et mobile grâce à sa nouvelle architecture et ses nouvelles fonctionnalités.

Concernant la réalisation de cette application, On a opté pour la méthode agile afin de mieux comprendre les besoins de notre client.

Abstract

The project presented in this document is part of the UE software engineering project in our Master's degree program in Computer Science at the Aix-Marseille Luminy University. Our project consists of improving the already existing code in Angular 1 by upgrading it to Angular 4 which is the new version that's more adaptable for web and mobile browsers thanks to its new architecture and new features.

Regarding the realization of this application, We opted for the agile method to better understand the needs of our client.

1. WIKI

1.1. Presentation

Dans le cadre de notre formation en Master 1 informatique, nous avons été amenés à effectuer un projet de Génie logiciel sous le thème d'Angular 1 vers Angular 4. L'objectif de ce projet est d'améliorer le code derrière, et garder la même interface et les mêmes fonctionnalités qu'avant.

1.2. Auteurs

- ALOUANI Salah :Alouani.salah9@gmail.com
- BOUSTANI Sara :boustanisara5@gmail.com
- OLLIER Guillaume :guillaume.ollier@etu.univ-amu.fr

1.3. Manuel d utilisation

Installation :

Install for Ubuntu 16.04 LTS

- **Install NPM et Node js**

```
$ sudo apt-get update  
$ sudo apt-get install nodejs npm
```

- **Lancer l'interface graphique**

Se mettre dans le dossier racine du projet
Lancer la commande : \$ Npm start

Install for MacOS

```
$sudo port install nodejs6
```

```
$sudo port install npm3
```

Install Windows

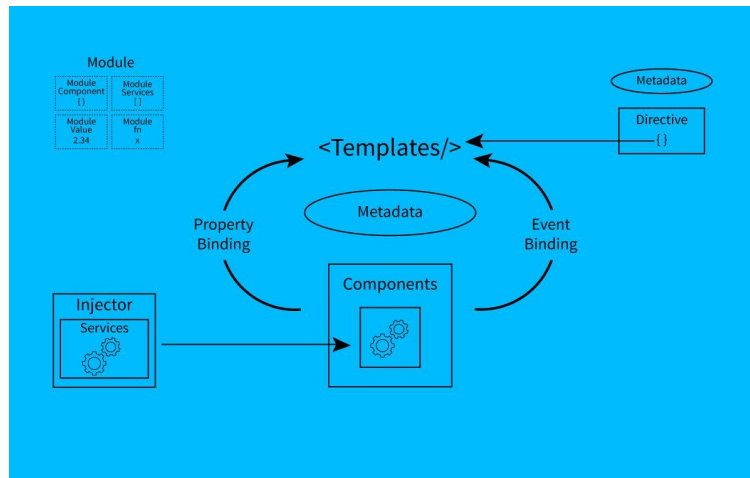
- Install jdk
- Install eclipse
- Install eclipse Angular plugin

Lancer le projet depuis eclipse et taper la commande “Npm start “ dans le terminal intégré dans eclipse

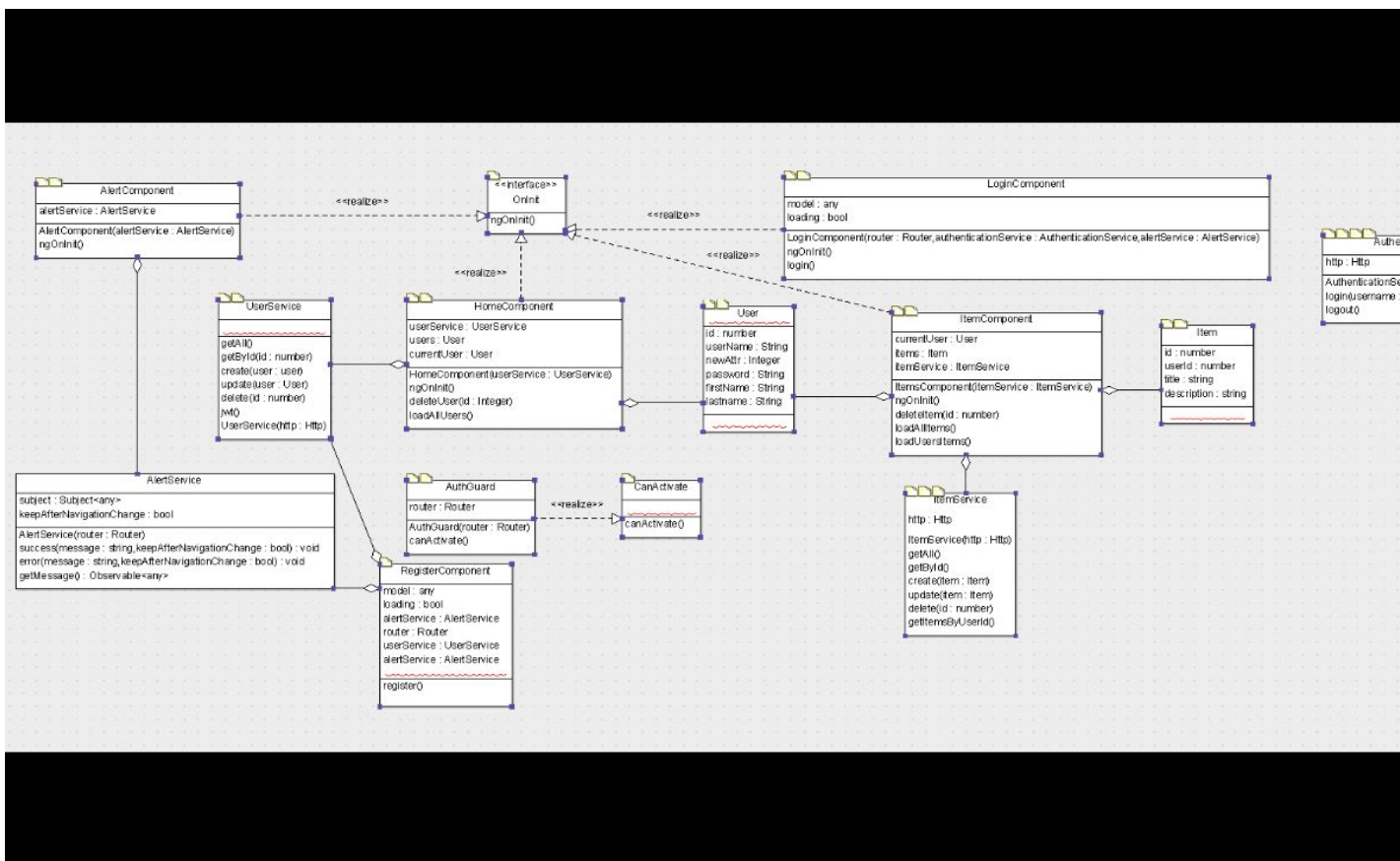
1.4. Les principaux packages

Le projet est composé de different components qui sont routé entre eux , d’une manière interactive, les principaux components sont :

- Account component
- App component
- App module
- Login component
- Home component
- Settings component
- Search component
- New message component
- New password component
- Main component
- Message component



1.5. UML



1.6. TODO

- Continuer la migration des autres composants vers angular 4
- Rendre l'interface graphique plus simple et conviviale
- Assurer la sécurité des informations
- Lier la GUI à une base de données

2. Travail réalisé

2.1. Cahier de charge

L'élaboration du cahier des charges s'avère nécessaire pour la mise au point du projet. En effet, le cahier des charges recense l'expression des besoins nécessaires, essentiels, fonctionnels et technique de la solution que le commanditaire (maître d'ouvrage) souhaite obtenir à la fin du projet. Le cahier des charges nous a été fourni par le référent de stage et sera disponible en annexes.

2.2. Méthode SCRUM

Méthode SCRUM :

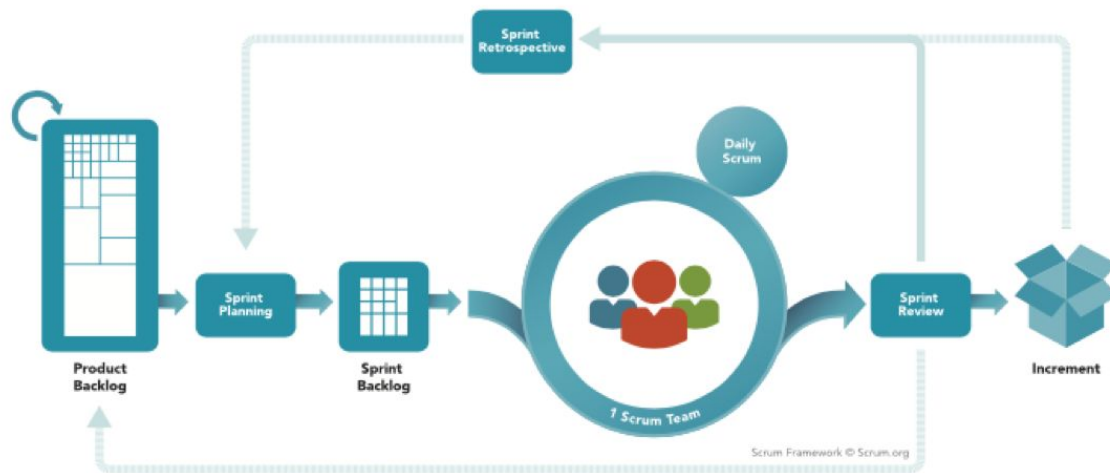
Nous avons décidé d'opter dans notre projet pour l'application de la méthode SCRUM afin de gérer notre projet.

Il s'agit d'une méthode agile dédiée à la « gestion de projet ». Cette méthode de gestion, ou plutôt ce Framework de management de projet, a pour objectif d'améliorer la productivité de son équipe dans laquelle différents rôles sont attribués à chacun.

Dans notre contexte celui qui jouera le rôle du Scrum Master sera OLLIER Guillaume et le rôle du Product Owner sera quant à lui attribué à ALOUANI Salah .

Nous avons glissé dans les annexes nos différents sprints.

SCRUM FRAMEWORK



Définition of Done :

Dans notre équipe, et donc dans ce document, nous définissons un point de charge de la manière suivante : l'unité ici sera en jour que nous multiplierons par le nombre de personne. Nous sommes 3 à travailler sur ce projet et nous disposons de 1 jour de travail par personne soit un point de charge de 3 jours par semaine. Avec une journée dédiée aux réunions avec toute l'équipe.

Définition du Backlog et des Sprints :

Afin d'utiliser cette méthode dite AGILE il est nécessaire de définir un Backlog ainsi que des Sprints. Le Backlog a été conçu à partir du cahier des charges fourni initialement à la création du projet.

Logiciels et langages implémentés :

Dans cette partie nous allons vous parler de tous les langages, logiciels et librairies que nous avons utilisés lors de notre projet :

A. TYPESCRIPT

TypeScript est un langage de programmation libre et open source développé par Microsoft qui a pour but d'améliorer et de sécuriser la production de code JavaScript. C'est un sur-ensemble de JavaScript (c'est-à-dire que tout code JavaScript correct peut être utilisé avec TypeScript). Le code TypeScript est transcompilé en JavaScript, pouvant ainsi être

interprété par n'importe quel navigateur web ou moteur JavaScript. Il a été cocréé par Anders Hejlsberg, principal inventeur de C# 1,2,3, 4,5.

TypeScript permet un typage statique optionnel des variables et des fonctions, la création de classes et d'interfaces, l'import de modules, tout en conservant l'approche non-contraignante de JavaScript. Il supporte la spécification ECMAScript 6.

B. Framework :Angular4

AngularJS3 est un framework JavaScript libre et open source4 développé par Google. Il permet de développer des pages web

AngularJS est fondé sur l'idée que la programmation déclarative doit être utilisée pour construire les interfaces utilisateurs et les composants logiciels de câblage, tandis que la programmation impérative excelle pour exprimer la logique métier5. La conception de Angular est guidée par plusieurs objectifs:

- Découpler les manipulations du DOM de la logique métier. Cela améliore la testabilité du code ;
- Considérer le test d'une application aussi important que l'écriture de l'application elle-même. La difficulté de la phase de test est considérablement impactée par la façon dont le code est structuré ;
- Découpler les côtés client et serveur d'une application. Cela permet au développement logiciel des côtés client et serveur de progresser en parallèle, et permet la réutilisabilité de chacun des côtés ;
- Guider les développeurs pendant toute la durée de la construction d'une application : de la conception de l'interface utilisateur, en passant par l'écriture de la logique métier, jusqu'au test de l'application ;
- Rendre les tâches faciles évidentes et les tâches difficiles possibles.

C. IntelliJ IDEA

IntelliJ IDEA est un IDE Java commercial développé par JetBrains. Il est fréquemment appelé par le simple nom d'« IntelliJ » ou « IDEA » .

D. Eclipse

Eclipse est un projet, décliné et organisé en un ensemble de sous-projets de développements logiciels, de la fondation Eclipse visant à développer un environnement de production de logiciels libre qui soit extensible, universel et polyvalent, en s'appuyant principalement sur Java.

Son objectif est de produire et fournir des outils pour la réalisation de logiciels, englobant les activités de programmation (notamment environnement de développement intégré et frameworks) mais aussi d'AGL recouvrant modélisation, conception, test, gestion de

configuration, reporting... Son EDI, partie intégrante du projet, vise notamment à supporter tout langage de programmation à l'instar de Microsoft Visual Studio.

Bien qu'Eclipse ait d'abord été conçu uniquement pour produire des environnements de développement, les utilisateurs et contributeurs se sont rapidement mis à réutiliser ses briques logicielles pour des applications clientes classiques. Cela a conduit à une extension du périmètre initial d'Eclipse à toute production de logiciel : c'est l'apparition du framework Eclipse RCP en 2004.

Figurant parmi les grandes réussites de l'open source, Eclipse est devenu un standard du marché des logiciels de développement, intégré par de grands éditeurs logiciels et sociétés de services. Les logiciels commerciaux Lotus Notes 8, IBM Lotus Symphony ou WebSphere Studio Application Developer sont notamment basés sur Eclipse.

Présentation du Programme

Description du travail réalisé :

Actuellement, le programme que nous avons développé nous permet de répondre aux principaux problèmes demandés lors de la création du projet.

Il est possible :

- Authentification : Pour s'authentifier, l'utilisateur saisit son login et mot de passe, le système lui renvoie la page d'accueil correspondante.
- Création compte : il s'agit d'une page où l'utilisateur peut créer un nouveau compte.
- Menu principal se compose de 9 sous menus qui sont accessibles dès que l'utilisateur s'authentifie
- Ajouter des Items
- Ajouter un contrat
- Changer le mdp
- Envoyer un message

Bilan :

Bilan Scientifique:

Au cours de ce projet, nous avons rencontré de nombreuses difficultés telles que:

- L'assimilation de nouvelles notions dans le domaine du développement web .
- Apprendre à maîtriser le framework Angular 4 et intégrer sa librairie dans notre code afin de pouvoir facilement manipuler ses méthodes.
- Absence d'un membre de notre groupe , on devrait alors partagé le travail qui a été prévu pour 4 personnes à 3 personne.

Bilan humain:

Le travail en équipe peut s'avérer parfois délicat, c'est pourquoi la mise en place d'un système de gestion de projet efficace est indispensable. Dans notre cas, l'utilisation d'une

méthode dite AGILE, nous a permis d'être plus efficace dans la réalisation de notre projet. Les réunions hebdomadaires, nous ont permis d'apprendre à gérer notre temps de travail et la pertinence des informations que nous devons présenter. En effet, chaque semaine, lors de nos réunions, nous avons été amenés à présenter le travail effectué au cours de la semaine, ainsi que les problèmes rencontrés.

3. CONCLUSION ET PERSPECTIVE

En conclusion, malgré toutes les difficultés que nous avons pu croiser tout au long du rapport, nous sommes en mesure d'affiner tous les points abordés lors de ce rapport. Et ainsi, nous estimons avoir été capables de répondre aux attentes demandées lors de la création du projet. Toutes les méthodes utilisées et présentées lors des points précédents nous ont permis de créer une interface homme-machine simple et conviviale. Nous avons été en mesure de pouvoir développer le maximum de fonctionnalités proposées lors de la création du projet dans la limite du temps qui nous a été allouée. Nous pensons que le programme est enclin à des améliorations, c'est pourquoi nous avons fait en sorte de laisser le logiciel ouvert et modulable, afin de permettre des modifications ultérieures et l'ajout de fonctionnalités supplémentaires par des utilisateurs tiers et l'optimisation de l'interface.

4. ANNEXES

4.1. Cahier de charge et Spécifications

Analyse des besoins

Dans le cadre de notre formation en Master 1 informatique, nous avons été amenés à effectuer un projet de Génie logiciel sous le thème d'Angular 1 vers Angular 4. L'objectif de ce projet est d'améliorer le code derrière, et garder la même interface et les mêmes fonctionnalités qu'avant.

Contexte du projet

Notre projet est la reprise d'un projet de l'année dernière, il était mené par anciens étudiants M1 Informatique, ce groupe avait pour objectif principal d'améliorer le code de l'interface côté client, dans le but d'utiliser les nouvelles technologies, qui facilitent beaucoup le développement comme la maintenance des applications, et sans toucher à la partie Administration de l'application.

La prise en main de ce projet était prévue en une semaine environ. Celle-ci a duré plus que prévu initialement par manque de documentation et absence de certains membres de groupe.

Development

Angular est un framework très largement utilisé aujourd'hui. Il doit faciliter le développement des applications web et met en avant, depuis sa version 2.0, le langage TypeScript de Microsoft, qui est pour rappel un sur ensemble de JavaScript adapté aux gros projets. Voilà donc Angular 4.0, qui est en fait la troisième version majeure du framework. Les nouveautés sont assez nombreuses, et bien qu'il s'agisse d'une mouture importante, elle est « compatible avec la 2.X.X pour la plupart des applications ». L'un des axes d'amélioration est de rendre les applications construites avec Angular plus légères et plus rapides. Plusieurs nouveautés vont donc en ce sens.

Pourquoi passer d'angular 2 à 4 ?

Le but principal est que toutes les bibliothèques Angular soient hébergées sur un seul dépôt GitHub avec le même numéro de version.

Angular 4 supporte désormais Typescript 2.1, passer à cette version plus récente de TypeScript signifie en effet une meilleure vérification de type dans une application et de meilleures performances pour ngc, le compilateur pour les templates Angular, donc Angular 4 n'est pas une révolution par rapport au passage de Angular 1 à 2 mais une évolution apportant des nouveautés pratiques.

Avantages et inconvénients d'Angular 4

Avantage:

- des applications Angular plus petites et plus rapides
- une réduction, de plusieurs centaines de kilo-octets, de la taille de leurs paquets en production.
- du code plus expressif un respect des nouveaux standards du web
- Compilation rapide

Inconvénient:

- migration compliquée de l'ancienne version (angular 1) vers la nouvelle
- les vieux navigateurs ne sont pas supportés

4.2. SPRINTS

Projet ANGULAR 2SXP

Master 1 Informatique

Participants

| | |
|----------------|---|
| Développeur(s) | M .ALOUANI Salah Mlle .BOUSTANI Sara M.OLLIER Guillaume |
| Scrum Master | M.OLLIER Guillaume |

Definition of Done

C'est une tâche ou une compétence que nous considérons comme acquise, que nous maîtrisons et qui est applicable directement sur le projet.

Définition du Point de charge

Dans notre équipe, et donc dans ce document, nous définissons un point de charge de la manière suivante : l'unité ici sera en jour que nous multiplierons par le nombre de personne. Nous sommes 3 à travailler sur ce projet et nous disposons de 1 jours de travail par personne soit un point de charge de 3 jours par semaine. Avec une journée dédiée aux réunions avec toute l'équipe

Semaine du : 04/10/2017
Durée de Sprint: 2 semaines
Past Sprint : 0 -- Next Sprint : 1

Next sprint backlog:

| Item |
|---------------------------------------|
| Compréhension du sujet |
| Mise en place de la gestion du projet |

Semaine du : 18/10/2017
Durée de Sprint: 2 semaine
Past Sprint : 1 -- Next Sprint : 2

Past sprint backlog

| Item | Fait(%) |
|---------------------------------------|---------|
| Compréhension du sujet | 100 |
| Mise en place de la gestion du projet | 60 |

Next Sprint backlog

| Item |
|---|
| Installation de logiciels |
| Prise en main de l'ancienne version d'angular 1 |

Semaine du : 08/11/2017
 Durée de Sprint: 2 semaine
 Past Sprint : 2 -- Next Sprint : 3

Past sprint backlog

| Item | Fait(%) |
|---|---------|
| Installation de logiciels | 100 |
| Prise en main de l'ancienne version d'angular 1 | 100 |
| Mise en place de la gestion du projet | 100 |

Next Sprint backlog

| Item |
|------------------------------------|
| Division des taches |
| Programmer les nouveaux composants |

Semaine du : 15/11/2017
Durée de Sprint: 1 semaine
Past Sprint : 3 -- Next Sprint :4

Past sprint backlog

| Item | Fait(%) |
|------------------------------------|---------|
| Division des taches | 80 |
| Programmer les nouveaux composants | 30 |

Next Sprint backlog

| Item |
|------------------------------------|
| Programmer les nouveaux composants |
| Rapport |

Semaine du : 29/11/2017
Durée de Sprint: 1 semaine
Past Sprint : 4 -- Next Sprint :5

Past sprint backlog

| Item | Fait(%) |
|------------------------------------|---------|
| Programmer les nouveaux composants | 80 |
| Rapport | 90 |

Product backlog

Valeur: 4=Indispensable, 3=Important, 2=Souhaitable, 1=Joli, 0=Inutile

Risque: 4=Impossible, 3=Fort, 2=Faisable, 1=Facile, 0=Aucun

| Item | Valeur | Charge | Risque |
|--|--------|--------|--------|
| TOMOSOFT | | | |
| Comprendre le fonctionnement de l'application | 4 | 5 | 3 |
| Apprendre Angular4 et TypeScript | 4 | 3 | 3 |
| Création du Layout globale | 3 | 3 | 3 |
| Gestion d'événements souris | 3 | 5 | 3 |
| Implémentation de fonctionnement de l'application | 3 | 4 | 3 |
| Tests | 3 | 3 | 3 |
| INTERFACE | | | |
| Interface conviviale et simple d'utilisation. | 3 | 2 | 3 |
| Affichage des résultats. | 4 | 5 | 2 |
| Optimisation | 3 | 3 | 2 |
| AUTRES | | | |
| Mettre en place de la gestion du projet (git, discussion, trello..). | 4 | 1 | 0 |
| Rapport | 4 | 5 | 2 |
| Installation des librairies. | 4 | 4 | 1 |
| Compréhension du projet | 4 | 7 | 3 |
| total | 43 | 50 | 31 |