

Rapport final TER

*Projet Quick Recipes*



ROMDAN Elias  
M1DSC

1. Introduction

-n/a-

2. Description de l’application

2.1 Expression du besoin

Nous vivons dans une époque où la majorité des échanges d’informations se passe par le réseau Internet. L’accès à l’information est devenu possible à n’importe quel individu ayant une connexion Internet et inversement, cet individu peut créer lui-même cette information et la rendre accessible au grand public. En ayant plusieurs acteurs participant à la création de l’information, ce flux ne cessera d’augmenter avec le temps.

Plus particulièrement dans le domaine de la cuisine, il est devenu facile d’avoir accès aux recettes cuisinières et d’essayer de les appliquer chez soi, sans même la nécessité d’être un chef cuisiner ou d’avoir suivi des cours en cuisine. La plupart des solutions existant actuellement sur le marché se ressemblent toutes et sont trop focalisées sur l’aspect préparation que les autres points, comme l’historique d’un plat, sont négligés, de plus la grande quantité d’informations fournie au même moment n’est pas toujours facile à exploiter.

Le besoin est toujours présent d’une solution simple et rapide à naviguer, qui centralise les différentes informations sur un plat cuisiner, que ce soit l’aspect préparation, historique ou sanitaire.

2.2 Objectifs

L’objectif de ce projet est de développer une solution qui permettra de répondre aux besoins cités dans l’expression du besoin et de faire en sorte que la source d’informations de cette solution soit alimentée principalement par les spécialistes de la cuisine, ainsi que de l’ajout d’un système de recommandation géré par une IA.

2.3 Solution

La solution est une application au nom de « Quick Recipes » développée dans un environnement web.

3. Architecture du projet

3.1 Langages de programmation

Le choix des langages de programmation pour le développement du logiciel « Quick Recipes » était fait en fonction des plateformes où l’application sera disponible, de mes acquis pendant le cursus de Master 1 et des avantages que les technologies actuelles apporteront à l’application :

* Le côté client sera codé avec le Framework Angular qui sera composé du langage HTML et TypeScript. Le design sera effectué en langage CSS.
* Le côté serveur sera codé en Java EE avec le Framework Spring. L’interface de programmation JPA sera utilisée pour les requêtes. Dans le cas où ces requêtes seront complexes, le langage SQL sera utilisé à la place.
* La base de données utilisée sera MySQL pour ses performances mais aussi son accès libre.
* Le style d’architecture logicielle REST sera utilisé pour établir la connexion entre le côté client et serveur à travers des API.

3.2 Environnement de développement

L’environnement utilisé dans le développement du logiciel « Quick Recipes » sera un ensemble d’outils, qui seront choisis en fonction des besoins de l’application et de mon expérience à les utiliser :

OS : Windows.

IDE : Visual Code Studio.

3.3 Normes et standards

Pendant le développement du logiciel « Quick Recipes », les règles ci-dessous doivent être respectées pour éviter toute anomalie dans la compréhension et l’utilisation du code :

* Utilisation de Camel Case lors du nommage des variables et des méthodes.
* Utilisation de JavaDoc pour la documentation du code source Java.
* Utilisation des conventions de codage Java.
* Indentation de code cohérente en utilisant des espaces.
* Écriture du code et des commentaires en Anglais.
* Les ressources utilisées doivent être en open source.

4. Fonctionnalités de l’application

4.1 Fonctionnalités mises en place

Composant 1 : visualisation d’un plat

- Visualiser le nom d’un plat

- Visualiser les ingrédients d’un plat

- Visualiser la préparation d’un plat

- Visualiser les images d’un plat

- Visualiser l’historique d’un plat

- Visualiser le bilan nutritif d’un plat

- Visualiser les mots clés (catégorie et compagnie) d’un plat

- Système de recommandation par l’IA (à détailler)

Composant 2 : liste des plats

- Barre de recherche

- Visualiser la liste des plats (image de couverture + nom plat + région)

Composant 3 : page principale

- Barre de recherche (par nom plat / ingrédient / mots clés (catégorie/région))

Composant 4 : rentrer des fichiers avec infos sur les plats

4.2 Fonctionnalités prévues dans le futur

-n/a-

5. Modélisation

5.1 Diagramme de cas d’utilisation

-n/a-

5.2 Diagramme d’activité

-n/a-

5.3 Diagramme de séquence

-n/a-

5.4 Diagramme de classe

-n/a-

6. Interfaces Homme-Machine

6.1 Interface A

-n/a-

6.2 Interface B

-n/a-

7. Synthèse

7.1 Auto-critique

-n/a-

7.2 Difficultés rencontrées

-n/a-

7.3 Acquis

-n/a-

7.4 Quantité de code

-n/a-

8. Conclusion

-n/a-