

# Gestion de Projet et Génie logiciel

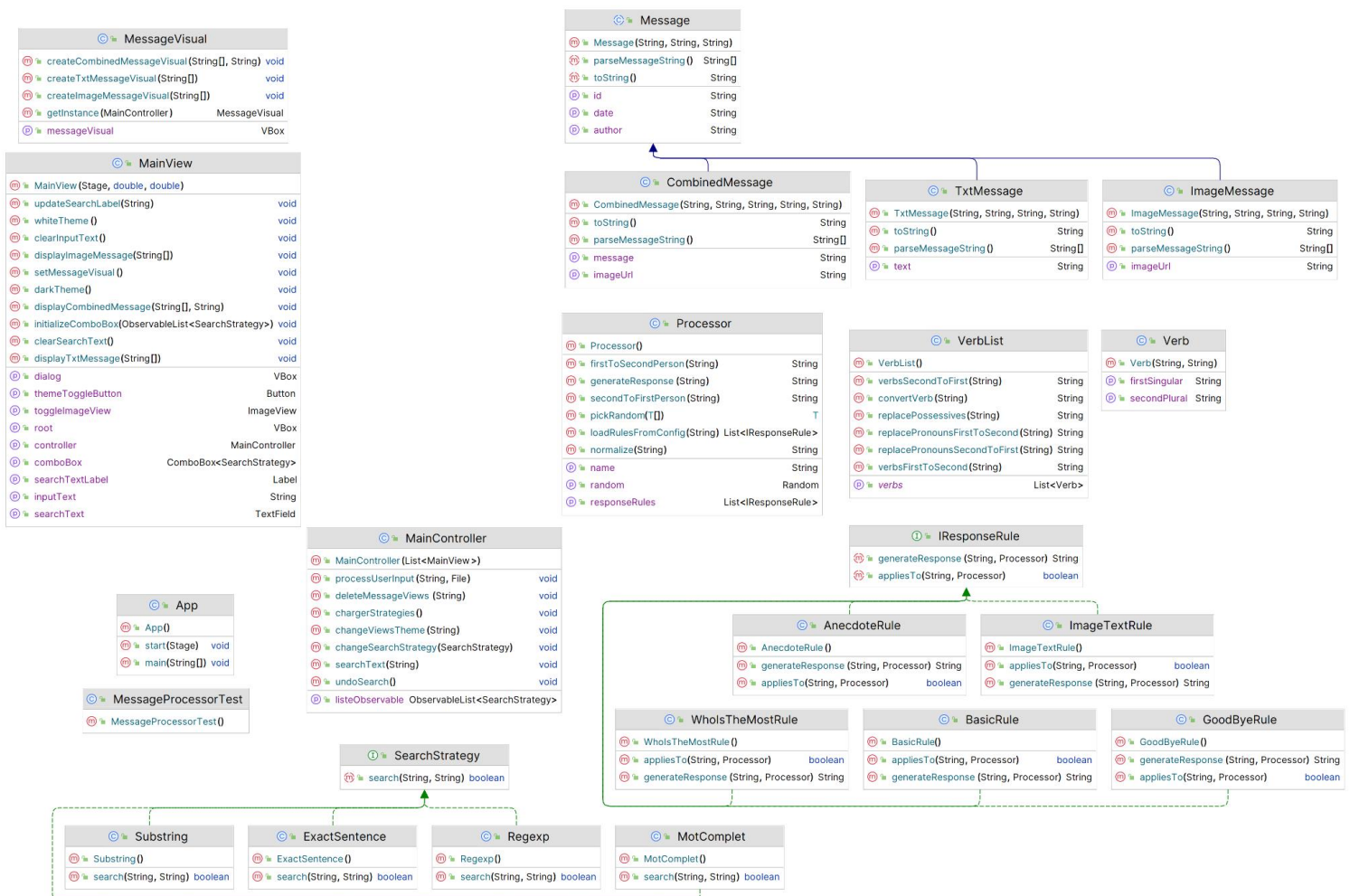
<b>I. Présentation globale du sujet .....</b>	<b>2</b>
II. Diagrammes et Design patterns .....	2
III. Éthique .....	5
IV. Tests .....	6
V. Conclusion .....	6

## I. Présentation globale du sujet

Durant ce premier semestre de Master Informatique, nous avons eu l'opportunité de réaliser un projet d'un logiciel d'intelligence artificielle similaire à l'intelligence artificielle ELIZA développé par Joseph Weizenbaum en 1966. Cette intelligence artificielle est un chatbot conversationnel. L'objectif de ce projet est de simuler un dialogue entre un humain et une intelligence artificielle qui lui répond selon sa question. L'objectif principal de ce projet est d'apprendre à gérer les étapes de développement d'un projet en binôme, d'être capable de planifier l'avancement du projet selon le respect d'une chronologie basée sur les séances de travaux pratiques. Chaque séance, ayant pour but d'aborder un aspect de la gestion d'un projet, nous permettait de mettre en œuvre les bonnes pratiques de développement. Nous avons pu ainsi utiliser plusieurs outils importants tels que Git, un gestionnaire de version qui permet de travailler en collaboration entre plusieurs développeurs. Nous avons pu apprendre le respect des règles de checkstyle pour ainsi respecter les normes du développement et ainsi pu maintenir une qualité de code élevée. Pour ce faire, il nous a été fourni un squelette de code dont l'architecture ne respectait pas le design pattern fréquemment utilisé dans l'univers du développement d'application, le MVC (Modèle-Vue-Contrôleur). Ainsi, les premières étapes du projet étaient de refactoriser le code de base pour que celui-ci respecte un design pattern que nous avons préalablement choisi en nous basant sur le cours.

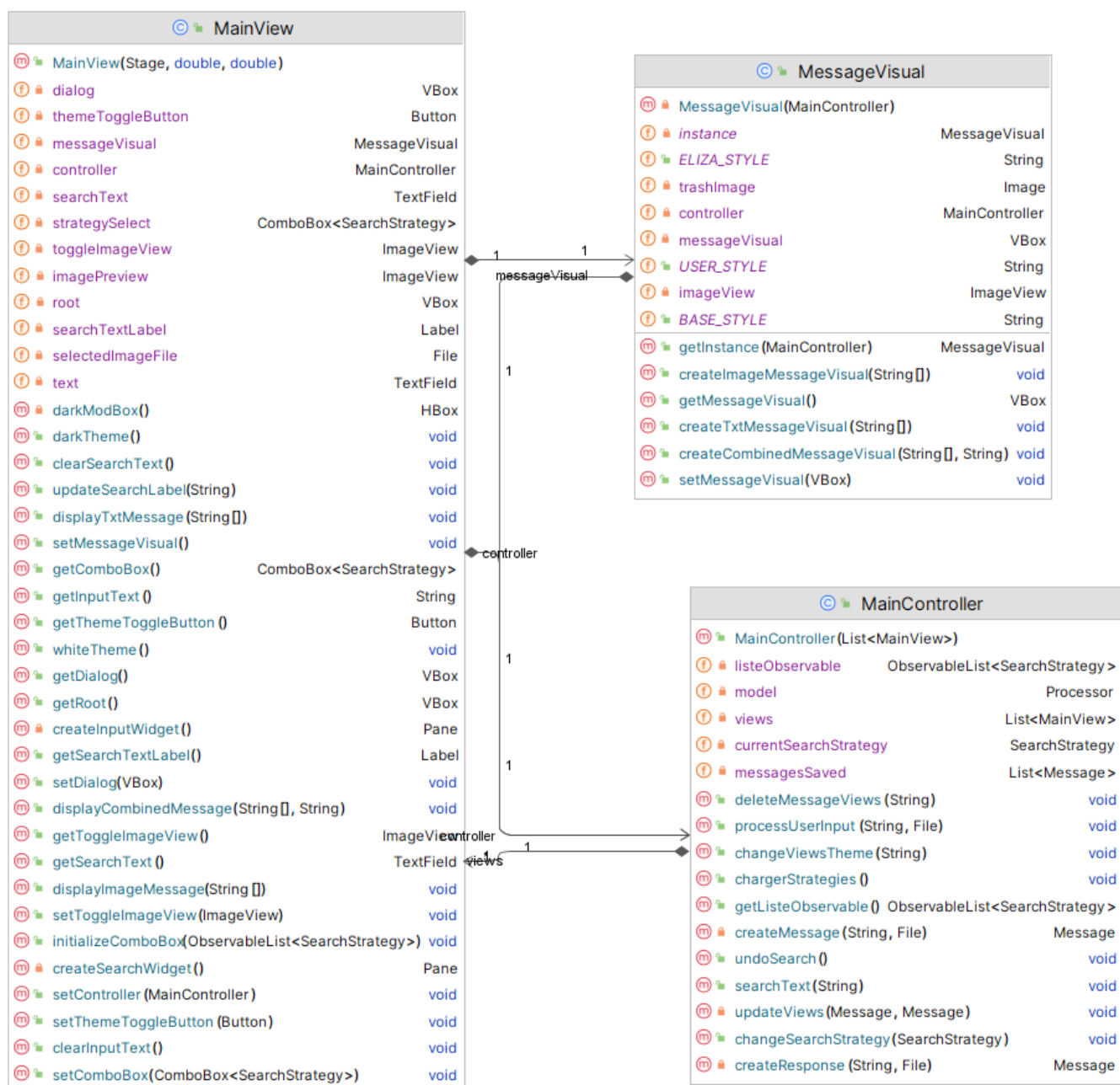
## II. Diagrammes et Design patterns

### Diagramme UML - structure globale du projet.



### Diagramme des dépendances





Le constructeur de la classe MessageVisual étant privée, on récupère l'instance grâce à la méthode getInstance depuis MainView.

### III. Éthique

Ce projet se prête bien aux tendances actuelles en ce qui concerne l'avancée et l'essor informatique dans le monde d'aujourd'hui. En effet, un des sujets principaux qui attirent l'attention et suscitent des questionnements de nombreuses personnes qu'elles soient concernées ou non par l'informatique est l'intelligence artificielle. Cet engouement est dû à la démocratisation de l'intelligence artificielle ainsi qu'à la facilité d'accès de nombreux services d'intelligence gratuitement ou payant, le plus connu étant ChatGPT, un chatbot capable de répondre à n'importe quelle question. Cet outil est devenu le service le plus utilisé du moment pour n'importe quelle utilisation, que ce soit pour le travail, pour les étudiants quel que soit leur niveau afin de faciliter les tâches. Les enjeux d'un tel service sont multiples. Nous pouvons relever la confidentialité. Les conversations avec ces intelligences artificielles peuvent contenir de nombreuses informations sensibles. Ces données sont à la portée des entreprises et des multinationales qui proposent ces services et peuvent ainsi utiliser nos données à des fins commerciales. C'est le cas de ChatGPT proposé par OpenAI qui nous informe, dans les

conditions générales d'utilisation, que nos données partagées avec l'entreprise par le biais des conversations sont stockées et utilisées par OpenAI. Cela pose un réel problème de confidentialité. C'est pourquoi de nombreux pays se penchent sur cette question comme l'Italie qui a décidé de suspendre l'utilisation du service sur le territoire national, car il est reproché l'absence d'informations quant aux données récoltées par l'intelligence artificielle. Cette suspension fait suite à un incident qui a laissé paraître publiquement certaines de ses conversations dans lesquelles apparaissent des informations personnelles sensibles.

Un autre enjeu est l'aspect social et psychologique. Certains utilisateurs utilisent ces services de discussion pour se confesser ou discuter de sujets qui sont sensibles à leurs yeux avec l'intelligence artificielle. L'intelligence artificielle, n'étant pas entraînée ou préparée pour faire face à ce cas de figure et ainsi répondre de manière éthiquement correcte, pourrait avoir un impact dramatique sur la vie des utilisateurs fragiles psychologiquement et qui nécessitent l'aide d'un professionnel de santé qui saura parfaitement répondre et réagir. Malheureusement, cela s'est déjà produit pour un utilisateur belge, anxieux face au réchauffement climatique, qui a longuement discuté avec une intelligence artificielle sur ce sujet. Au fur et à mesure de ses conversations, le chatbot ne contredit jamais et appuie les craintes de l'utilisateur. Ce programme avait été conçu dans l'idée de conforter son interlocuteur dans ses convictions. Ces nombreuses discussions ont poussé l'utilisateur à se donner la mort, encouragé par l'intelligence artificielle. Ce drame soulève une importante question sur l'éthique.

Nous pouvons nous questionner sur les risques et bénéfices de ces outils sur la société. Il est vrai que cet outil est très largement utilisé par la grande majorité des étudiants pour réaliser leurs devoirs, pour se renseigner sur des domaines scientifiques complexes, pour apprendre leurs cours ou pour effectuer le travail à leur place. L'utilisation de cet outil devrait être bénéfique pour ses utilisateurs car cela permet de simplifier de nombreuses tâches que l'utilisateur pourrait trouver pénible. Cependant, nous pouvons aussi nous questionner sur l'omniprésence de cet outil dans notre quotidien. Cela pourrait avoir un impact néfaste sur la progression et la capacité de l'utilisateur à réaliser la tâche souhaitée selon la manière dont il utilise l'intelligence artificielle. Nous pourrions enfin nous poser la question sur la manière d'utiliser cet outil afin qu'il soit réellement bénéfique.

Pour limiter les risques, quelques entreprises ont mis en place des mesures. Pour reprendre le cas tragique de l'utilisateur belge, le fondateur de la plateforme a mis en place un système d'avertissement vers les personnes qui expriment des pensées suicidaires. En ce qui concerne la question de l'âge des utilisateurs, nous pouvons prendre l'exemple de ChatGPT qui requiert d'être majeur afin d'utiliser le service. Sans quoi, l'autorisation parentale est requise pour utiliser le service. Cependant, l'entreprise ne vérifie pas réellement l'âge de ses utilisateurs et n'a pas encore mis en place de service de vérification à ce jour.

## IV. Tests

Pour tester notre application, nous avons utilisé les tests automatiques Java en ajoutant la bibliothèque Hamcrest qui permet d'écrire des tests plus proches du langage naturel. L'application étant un chatbot, nous avons fait en sorte de tester que les réponses générées par l'intelligence artificielle étaient conformes à ce qui est attendu. Dans notre choix de développement, nous avons mis en place des règles de réponse selon la forme de la question. Pour analyser la forme, nous avons utilisé les expressions régulières (regex). Au total, nous avons neuf types de règles. Selon la forme du message soumis, nous analysons s'il y a une correspondance entre les règles et le message. Par exemple, si le message contient le motif suivant : "Je m'appelle X ." (X étant une expression quelconque) reconnu par l'expression régulière suivante : ". \*Je m'appelle (.\*)\\." Alors le modèle génère une réponse cohérente. Pour le cas précédent, la réponse générée est : "Bonjour X .".

Nous avons mis en place les tests qui vérifient les règles suivantes :

L'intelligence artificielle est censée répondre un message final lorsque l'utilisateur soumet le message "Au revoir."

Lorsque l'on demande une anecdote à l'intelligence artificielle, nous vérifions bien que sa réponse est contenue dans le fichier des anecdotes. Lorsque nous donnons notre nom à l'intelligence artificielle, celle-ci doit bien le retenir et nous le retourner immédiatement dans la réponse suivante. Si le chatbot ne le connaît pas alors il doit répondre en conséquence. Dans notre

application, nous pouvons envoyer une image à notre intelligence artificielle. Cette image peut être accompagnée d'un texte. Si c'est le cas alors nous vérifions bien que celle-ci nous répond une réponse appropriée, sinon elle doit répondre d'autres réponses prédéfinies. Nous vérifions aussi que lorsque l'utilisateur pose une question, la réponse doit être celle attendue par la règle correspondante. Nous vérifions enfin que si le message entré ne correspond à aucune de nos règles définies alors elle doit nous répondre un message quelconque qui n'a pas de rapport avec notre message.

En ce qui concerne les tests manuels, ceux-ci ont été effectués sur les éléments graphiques de la vue. Nous avons testé que nous pouvions bien choisir une image, que lorsque cette dernière est envoyée alors la prévisualisation de l'image devient nulle, ce qui signifie qu'il n'y a plus d'image sélectionnée. Nous avons ensuite testé les boutons de l'interface ainsi que la liste de stratégie de recherche fonctionne correctement. Nous avons finalement testé la fonctionnalité de suppression de message suivie de la fonctionnalité de retour à l'état initial du chat à la suite d'une recherche.

## V. Conclusion

Ce projet a été une occasion enrichissante d'explorer diverses méthodes de conception, tout en nous familiarisant avec de nouveaux outils tels que Maven et Checkstyle, et en approfondissant notre maîtrise du langage Java et du système de gestion de versions Git.

Nous avons appris qu'un aspect crucial pour la réussite d'un projet de développement logiciel réside dans la qualité de sa conception structurelle, garantissant ainsi un code robuste, facile à maintenir et à étendre.

Nous avons réalisé qu'il n'existe pas une unique méthode de conceptualisation d'une application, mais plutôt que l'essentiel est de maintenir une cohésion forte au sein du code, tout en évitant des couplages trop rigides qui pourraient compromettre la flexibilité et l'évolutivité du système. En outre, la mise en place de bonnes pratiques de développement et l'adoption d'une approche modulaire se sont révélées être des stratégies clés pour atteindre une meilleure efficacité et une adaptabilité à long terme.

L'importance des tests dans le processus de développement logiciel ne saurait être sous-estimée. Tester la solidité du code est essentiel pour assurer sa fiabilité, sa sécurité et sa performance. Les tests automatisés, en particulier, jouent un rôle crucial en permettant une validation continue du code à chaque modification, garantissant ainsi une intégrité constante du système. De plus, les tests contribuent à documenter le comportement attendu du logiciel, ce qui est bénéfique pour les équipes de développement actuelles et futures.