**UNIVERSITE MOHAMMED V DE RABAT**

**ECOLE MOHAMMADIA D’INGENIEURS**

****

**Année universitaire : 2019 - 2020**

**Mémoire de Projet de Fin d’Etudes  
 N° INF**

**Dirigé par :**

Mme. KETTANI Bochra

Mr. EL IDRISSI Abdelhamid

**Implémentation d’un chatbot intelligent sur l’outil workflow SOGEKYC de SGMA**

**Soutenu par :**

ALAMI Malak

Filière : Génie Modélisation et Informatique Scientifique (MIS)

**UNIVERSITE MOHAMMED V DE RABAT**

**ECOLE MOHAMMADIA D’INGENIEURS**

****

**Année universitaire : 2019 - 2020**

**Membres de Jury :**

**Soutenu par :**

ALAMI Malak

**Implémentation d’un chatbot intelligent sur l’outil workflow SOGEKYC de SGMA**

**Mémoire de Projet de Fin d’Etudes  
 N° INF**

Filière : Génie Modélisation et Informatique Scientifique (MIS)

# Dédicace :

**A ma jolie maman, A mon cher papa,**

Je ne saurai exprimer mon respect, mon amour éternel et ma considération pour les sacrifices que vous avez consenti pour mon instruction et mon bien être. Puisse Dieu, le Très Haut, vous accorder santé, bonheur et longue vie.  
Que ce travail soit le fruit de vos sacrifices. Merci !

**A mon frère d’amour,**

Je t’aime au nombre de fois que je respire, que le Grand Dieu nous réunisse pour toujours et t’accorde tout le bonheur du monde.  
Très bonne chance pour ton bac !

**A ma chère tante Souad, A mon oncle Abdelatif,**

4 mois passés, et même pas un instant je me suis sentie loin de mes parents … Je vous remercie pour tout votre amour et votre soutien. Que ce travail soit l’exaucement de vos vœux sincères.

**A toutes mes tantes, mes oncles, mes cousins et cousines qui m’aiment,**

Je vous remercie pour votre amabilité et votre générosité. J’ai de la chance de vous avoir auprès de moi et j’espère que nous aurons plus de moments à vivre ensemble.

**A ma cousine Kenza, A ma grande sœur,**

Je ne saurai exprimer ma gratitude envers tout ce que tu fais pour moi ! Tu me manques et tu me manqueras au nombre de kilomètres qui nous séparent. Que Dieu nous réunisse pour toujours et t’accorde tout le bonheur du monde.

**A mes chers amis …**

Vous êtes simplement ma deuxième famille ! A chaque fois, nos heures passées ensemble semblent être des minutes à mes yeux, que Dieu préserve notre amitié pour toujours. Je vous aime de tout mon cœur !

# Remerciements :

*Tout d’abord, je remercie Dieu, le Tout Puissant de m’avoir donné le courage et la santé pour pouvoir accomplir ce projet de fin d’études.*

*الحمد لله*

**A Abdelhamid EL IDRISSI, A Kaoutar DOUMIRI, à toute l’équipe COO Solutions – Programme KYC :**

Je vous remercie tout d’abord pour votre gentillesse et votre sympathie et bienveillance. Quoiqu’on n’ait pas passé beaucoup de semaines ensemble, j’ai eu le plaisir de partager ces quelques moments avec vous. Merci pour votre aide sur tous les plans.

A Ouadie,

**A mon encadrante à l’EMI, Mme Bochra EL KETTANI :**

Je vous remercie pour votre aide et votre encadrement plein d’encouragement tout au long de mon PFE. Merci pour chaque instant que vous avez pris pour répondre à mes mails et à me faire part de vos conseils de valeur.

**A tous les collaborateurs de SGMA que j’ai dû rencontrer au long de mon stage,**

**MERCI !**

# Résumé :

Know Your Customer (KYC) – traduite littéralement par – Connaître son client, est le nom donné au processus permettant de vérifier l’identité des clients d’une entreprise, afin d’anticiper les fraudes et de lutter contre le blanchiment d’argent, dans le cas des banques.  
A Société Générale Maroc, l’efficacité et la qualité du processus KYC sont assurées à travers l’outil workflow SOGEKYC, constituant ainsi une préoccupation constante au sein de la banque, ce qui prouve bien l'adage selon lequel *le client est roi !*

*Mais comment en être sûr que les différents cas d’usages sont traités correctement sur l’outil ?*

Pour aider tous les acteurs KYC à réaliser, suivre et valider les exigences de la connaissance client, une aide digitale va devoir les accompagner dans leur voyage !

Mon projet de fin d’études consiste en la conception d’un agent conversationnel, appelé aussi « Chatbot » qui permettrait de répondre à la majorité des questions des gestionnaires et simuler ainsi une conversation humaine sur une interface de messagerie, disponible 24h/24 et 7j/7.

**Attention :** Avec les progrès de l’Intelligence Artificielle dans une société hyper connectée, les chatbots ont le droit à beaucoup d’attention !

# Abstract:

Know Your Customer (KYC) is the name given to the process of verifying the identity of a company's customers in order to anticipate fraud and combat money laundering, in the case of banks.

To Société Générale Maroc, the workflow tool SOGEKYC represents her top key concern and priority as long as it ensures the efficiency and the quality of KYC process, which highly proves the adage that *the customer is king!*

*However, how can we be sure that the different use-cases are handled correctly on the tool?*

In order to help all KYC stakeholders to achieve, monitor and validate customer requirements, a digital solution will have to accompany them on their journey!

My end-of-study project consists in designing a conversational agent, also called "Chatbot", which would answer the majority of managers' questions and thus simulate a human conversation on a messaging interface, available 24 hours a day, 7 days a week.

**Warning:** With the progress of Artificial Intelligence in a hyper-connected society, Chatbots deserve a lot of attention!

**ملخص:**

(KYC) - Know Your Customerهو الاسم المعطى لعملية التحقق من هوية زبائن شركة ما، من أجل مكافحة الاحتيال وغسيل الأموال، في حالة البنوك. في الشركة العامة، يتم ضمان كفاءة وجودة عملية KYC من خلال أداة سير العملSOGEKYC ، والتي تشكل مصدر اهتمام مستمر داخل البنك.

*ولكن كيف يمكننا التأكد من أن العمليات المختلفة تعالج بشكل صحيح على الأداة؟*

لمساعدة جميع مستخدمي الشركة على التحقق من كل معلومات الزبائن واستخدام هذه الأداة على أحسن وجه، ستتعين لهم مساعدة رقمية ترافقهم في رحلتهم!

فالهدف من مشروع نهاية دراستي هو برمجة روبوت محادثة، يُطلق عليه اسم "Chatbot" بالإنجليزية والذي سيجيب على معظم أسئلة المستخدمين على مدار الساعة، 7 أيام في الأسبوع.

مع تقدم الذكاء الاصطناعي في زمن العولمة، يحق لروبوتات الدردشة الحصول على الكثير من الاهتمام!

# Table des matières

[Dédicace : 2](file:///C:\Users\hp\Desktop\PFE\Rapport\Rapport%20(Nouvelle%20version).docx#_Toc42288448)

[Remerciements : 3](#_Toc42288449)

[Résumé : 4](#_Toc42288450)

[Abstract : 5](#_Toc42288451)

[Résumé en arabe : 6](#_Toc42288452)

[Table des matières 7](#_Toc42288453)

[Liste des figures 9](#_Toc42288454)

[Liste des tableaux 10](#_Toc42288455)

[Liste des abréviations : 11](#_Toc42288456)

[INTRODUCTION : 12](file:///C:\Users\hp\Desktop\PFE\Rapport\Rapport%20(Nouvelle%20version).docx#_Toc42288457)

[Chapitre 1 : 14](#_Toc42288458)

[Contexte général du projet 14](#_Toc42288459)

[1) Présentation de l’organisme d’accueil 15](#_Toc42288460)

[1.1) Groupe Société Générale 15](#_Toc42288461)

[1.2) Société Générale Maroc (SGMA) 15](#_Toc42288462)

[1.2.1) Présentation : 15](#_Toc42288463)

[1.2.2) Organisation de SGMA 16](#_Toc42288464)

[1.2.3) Organisation de l’équipe Solutions – Programme KYC 16](#_Toc42288465)

[2) Présentation du projet 17](#_Toc42288466)

[2.1) Introduction et Objectif 17](#_Toc42288467)

[2.2) Présentation de l’existant 18](#_Toc42288468)

[2.2.1) Programme KYC 18](#_Toc42288469)

[2.2.2) Outil SOGEKYC 18](#_Toc42288470)

[2.3) Problématique 18](#_Toc42288471)

[2.3.1) Au niveau du programme KYC 18](#_Toc42288472)

[2.3.2) Au niveau de l’outil SOGEYC 18](#_Toc42288473)

[2.4) Présentation de la solution proposée 19](#_Toc42288474)

[2.5) Conduite du projet 20](#_Toc42288475)

[2.5.1) Processus de développement choisi 20](#_Toc42288476)

[2.5.2) Planification du projet : Diagramme de GANTT 21](#_Toc42288477)

[Chapitre 2 : 22](#_Toc42288478)

[Etude fonctionnelle 22](#_Toc42288479)

[1) Cadre du projet 23](#_Toc42288480)

[1.1. Résumé du projet 23](#_Toc42288481)

[1.2. Objectif du projet 23](#_Toc42288482)

[1.3. Idée sur la solution 23](#_Toc42288483)

[2) Benchmark 24](#_Toc42288484)

[3) Spécifications fonctionnelles 25](#_Toc42288485)

[3.1. Périmètre fonctionnel 26](#_Toc42288486)

[3.2. Aperçu du chatbot 26](#_Toc42288487)

[4) Spécifications techniques 27](#_Toc42288488)

[4.1. Choix technologiques 27](#_Toc42288489)

[4.2. Hébergement 28](#_Toc42288490)

[4.3. Maintenance et évolution 28](#_Toc42288491)

[Chapitre 3 : 29](#_Toc42288492)

[Conception du chatbot 29](#_Toc42288493)

[1) La technologie RASA 30](#_Toc42288494)

[1.1. Présentation 30](#_Toc42288495)

[1.2. Fonctionnement global de RASA 30](#_Toc42288496)

[1.3. Installation de RASA Open Source et de RASA X 31](#_Toc42288497)

[2) Test de RASA sur une réelle conversation : 32](#_Toc42288498)

[2.1. A partir du terminal : 32](#_Toc42288499)

[2.2. Sur RASA X 37](#_Toc42288500)

[Chapitre 4 : 40](#_Toc42288501)

[Production du chatbot 40](#_Toc42288502)

[1) Structuration des questions KYC 41](#_Toc42288503)

[2) Implémentation du contenu sous RASA 43](#_Toc42288504)

[3) Test du chatbot sous RASA Open Source 47](#_Toc42288505)

[4) Plateforme du chatbot sous RASA X 50](#_Toc42288506)

# Liste des figures

**Figure 1** : 8 filiales de SG Maroc  
**Figure 2 :** Illustration d’une conversation avec un Script bot  
**Figure 3 :** Processus de développement en cascade  
**Figure 4 :** Diagramme de GANTT (Vue générale)   
**Figure 5 :** Diagramme de GANTT détaillé  
**Figure 6 :** Architecture du chatbot  
**Figure 7 :** Spécifications fonctionnelles du chatbot  
**Figure 8 :** Maquette simple du chatbot  
**Figure 9 :** Choix technologiques pour la réalisation du chatbot  
**Figure 10 :** Installation de Python et de la bibliothèque Pip  
**Figure 11 :** Plateforme de RASA X  
**Figure 12 :** Environnement virtuel Python **Figure 13 :** Création de nouveau dossier en ligne de commande **Figure 14 :** Préparation du dossier de test sous RASA **Figure 15 :** Aperçu des fichiers de test **Figure 16 :** Configuration du chatbot **Fichier 17 :** Fichier des stories de test **Fichier 18 :** Fichier des intentions de test  
**Figure 19 :** Fichier des réponses aux intentions de test  
**Figure 20 :** Entraînement du chatbot en ligne de commande  
**Figure 21 :** Test du chatbot  
**Figure 22 :** Importation des fichiers sur github  
**Figure 23 :** Connexion du chatbot sur github avec RASA X   
**Figure 24 :** Test du chatbot sur l’interface de conversation   
**Figure 25 :** Questions ouvertes à implémenter sur le chatbot   
**Figure 26 :** Use-cases à implémenter sur le chatbot  
**Figure 28 :** Intentions du chatbot implémentées   
**Figure 29 :** Stories du chatbot implémentées   
**Figure 30 :** Réponses aux intentions implémentées   
**Figure 31 :** Test du chatbot à partir du Terminal   
**Figure 32 :** Niveau de confiance associé aux intentions testé  
**Figure 33 :** Test final du chatbot sur l’interface de conversation

# Liste des abréviations:

**KYC:** Know Your Customer

**AML:** Anti-Money Laundering

**GAFI:** Groupe d’Action Financière

**CFT:** Combating Financing of Terrorism

**SGMA :** Société Générale Maroc

**SG :** Société Générale

**ML:** Machine Learning

**AI:** Artificial Intelligence

**DL:** Deep Learning

**NLP:** Natural Language Processing

**NLU:** Natural Language Understanding

**VM:** Virtual Machine

**GCP:** Google Cloud Platform

**AWS:** Amazon Web Services

**CSML:** Conversational Standard Meta Language

**OS:** Operating System

# Introduction :

Echapper au fisc, blanchir de l’argent sale ou contourner les sanctions internationales sont des opérations frauduleuses interdites par la loi. Par conséquent, les entreprises accusées de ces infractions courent un sérieux danger pour leur pérennité, pour l’économie et aussi pour le bien-être de toute la nation**1**, puisque les personnes et organismes impliqués dans ces violations passent généralement par un long parcours judiciaire et risquent d’être condamnées à de graves peines d’emprisonnement.   
 *Et tout comme n’importe quelle entreprise, les banques n’échappent pas à ces obligations !*

La réglementation oblige les [banques](https://www.lesclesdelabanque.com/Web/Cdb/Particuliers/Content.nsf/LexiqueByTitleWeb/banque) à avoir une **connaissance actualisée** de tous leurs clients y compris des revenus et du [patrimoine](https://www.lesclesdelabanque.com/Web/Cdb/Particuliers/Content.nsf/LexiqueByTitleWeb/patrimoine) et à suivre leurs opérations.  
A l’ouverture d’un compte bancaire à titre d’exemple, appelée aussi « **l’entrée en relation avec le client** », le gestionnaire est amené à lui poser plusieurs questions et à lui demander plusieurs justificatifs permettant à la banque de mieux le connaître, de mieux comprendre ses motivations et de lever l'éventuel doute sur les conditions dans lesquelles il souhaite réaliser une opération**2**.

**K**now **Y**our **C**ustomer ou processus **KYC**, faisant l’objet principal de ce projet, est une réglementation imposée et visitée régulièrement par le groupe d’action financière (GAFI), qui permet de vérifier l’identité des clients afin de prévenir les opérations de blanchiment d’argent (AML *standards*) et le financement du terrorisme (CFT)**3**.

L’application du **KYC** est alors un enjeu économique de premier ordre, puisque les informations complémentaires peuvent servir d’une part à l’optimisation de l’expérience client qu’il serait regrettable de ne pas exploiter, et reste d’une autre part un défi dans la mesure où il recouvre un nombre important d’obligations plus ou moins contraignantes ; car plus le risque présenté par l’opération ou par le client est élevé, plus l’obligation de vigilance et le travail de l’équipe de conformité sont exigeants**4**.

Dans ce cadre, afin de gérer les données des clients de façon performante, l’un des enjeux stratégiques de Société Générale Maroc est de dématérialiser le processus **KYC** et de faciliter le traitement du flux de documents et de données client par le nouveau dispositif workflow **SOGEKYC** et de le mettre à la disposition de tous les collaborateurs de la banque.   
  
Aujourd’hui, la maîtrise des processus et de tous les outils liés à la réglementation **KYC** n’est pas chose facile ! Ainsi, toutes les équipes se mobilisent pour rendre le support de cette instruction encore plus accessible et encore plus rapide.

**Chronologie selon la table des matières**

# 

# Chapitre 1 :

# Contexte général du projet

*Les premières pages de ce rapport ont pour rôle d’offrir une vision générale du projet en délimitant ses contours.*

Description du chapitre

## 

## 1) Présentation de l’organisme d’accueil

### 1.1) Groupe Société Générale

**Société Générale est l’un des tout premiers groupes européens de services financiers**. S’appuyant sur un modèle diversifié et intégré, le Groupe allie solidité financière, dynamique d’innovation et stratégie de croissance durable afin d’être le partenaire de confiance de ses clients, engagé dans les transformations positives du monde.

**Acteur important de l’économie depuis plus de 150 an**s avec un ancrage solide en Europe et connecté au reste du monde, **Société Générale** emploie plus de 149 000 collaborateurs dans 67 pays **(Le Maroc y compris)** et accompagne au quotidien 31 millions de clients particuliers, entreprises et investisseurs institutionnels à travers le monde, en offrant une large palette de conseils et de solutions financières sur mesure afin de sécuriser leurs échanges, protéger et gérer leur épargne, les aider à financer leurs projets, les protéger dans leur vie et leurs activités et leur offrir des solutions et des services innovants**5**.

### 1.2) Société Générale Maroc (SGMA)

#### 1.2.1) Présentation :

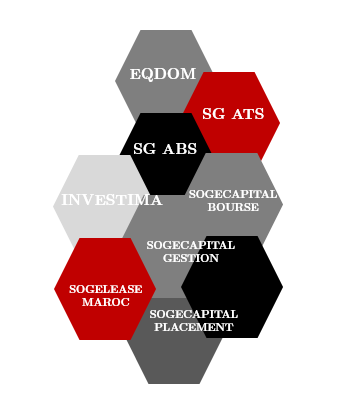
C’est en 1913 que la première agence Société Générale a ouvert ses portes au Maroc.

Depuis, **Société Générale Maroc** est devenu un groupe financier de premier plan, s’appuyant sur un modèle éprouvé de banque universelle et de synergies avec ses nombreuses filiales spécialisées.

Tout au long du siècle écoulé, Société Générale Maroc a su adapter progressivement son organisation et son dispositif afin de répondre au mieux aux attentes de tous ses clients : Grandes Entreprises, Investisseurs Institutionnels, PME/PMI, Particuliers et Professionnels.

Se dotant d’un esprit d’équipe pointu, les collaborateurs de Société Générale Maroc et de toutes les filiales sont innovants et responsables de satisfaire tous les besoins des clients par un esprit de service nourri de la diversité des expertises et connaissances de chacun.

Le groupe Société Générale étant le premier groupe financier international du pays renforce sa relation avec ses clients en apportant une expertise sur divers métiers financiers spécialisés à travers ses filiales**6** :



*Source :* [*https://www.sgmaroc.com/*](https://www.sgmaroc.com/)

**Figure 1** : 8 filiales de SG Maroc

#### 1.2.2) Organisation de SGMA

#### 1.2.3) Organisation de l’équipe Solutions – Programme KYC

## 2) Présentation du projet

### 2.1) Introduction et Objectif

En plus de gérer la concurrence et l’environnement économique instable, toutes les banques du monde sont confrontées à des menaces qu’elles doivent efficacement gérer, comme les cas de blanchiment d’argent et les fraudes fiscales.

Quand une banque tire des profits directs ou indirects d’activités criminelles, en justifiant ou en dissimulant l’origine des biens ou des revenus de l'auteur du crime ou du délit, elle est amenée à payer des amendes très lourdes et lui fait perdre beaucoup de sa crédibilité et de sa transparence**7**.

Afin d'assurer donc la sécurité des placements et des investissements et de permettre la transparence et la traçabilité des opérations bancaires, plusieurs normes sont mises en place.   
Dans le cadre de la **lutte contre le blanchiment de capitaux et le financement du terrorisme et de la prolifération :**



*Le Groupe d’action financière (GAFI) est un organisme intergouvernemental d’élaboration de politiques dont l’objectif est d’établir des normes internationales, et de développer et promouvoir les politiques nationales et internationales de lutte contre le blanchiment de capitaux, le financement du terrorisme et le financement de la prolifération des armes de destruction massive***8***.*

**KYC** est l’instruction (*mise en place par le GAFI et revisitée par d’autres organismes législatifs)* qui consiste à identifier le client et à vérifier toute information collectée avant d’entrer en relation bancaire avec lui.

Afin de se conformer aux exigences internationales en la matière, **SGMA** a également poursuivi la mise à niveau de sa réglementation interne spécifique à la lutte contre le blanchiment de capitaux et le financement du terrorisme, sur la base d’un recensement exhaustif des textes nécessitant un alignement par rapport aux standards du GAFI.

*L’objectif de ce projet de fin d’études est de garder un tracé de toute l’instruction KYC, disponible 24h/24, 7j/7 et facilement accessible de la part de tous les gestionnaires de la banque.*

### 2.2) Présentation de l’existant

#### 

#### 2.2.1) Programme KYC

#### 2.2.2) Outil SOGEKYC

### 2.3) Problématique

#### 2.3.1) Au niveau du programme KYC

**Le tracé de l’instruction KYC est manuscrit !** S’il faut revisiter une notion au siège de **SGMA**, il faut chercher ces dossiers au bout de l’étagère afin d’y répondre. A l’ère de la digitalisation, ceci devenu inacceptable !

Par contre, si le besoin est réclamé au niveau de l’agence ou de la banque de détail, il faut poser la question au siège !

*La question est peut-être classique et c’est très lassant d’y répondre à chaque fois ?  
La question est très basique et met en cause l’assimilation des gestionnaires ?  
La question prend du temps à répondre car le responsable est parti prendre un café ?*

Pour répondre à ces problèmes, il est donc très légitime de penser à une solution digitale efficace, facilement accessible et disponible 24h/24.

*Mais quelle solution au juste ?*

#### 2.3.2) Au niveau de l’outil SOGEYC

La manipulation de l’outil SOGEKYC est encore plus primordiale que la connaissance de l’instruction KYC ! L’outil workflow est utilisé dans les 4 processus : Entrée en relation, Remédiation, Contrôle LOD1 et Revue périodique.

A titre d’exemple, une entrée en relation ne peut jamais attendre ! Si un Chargé de Clientèle (CC) a besoin d’appeler le siège pour un cas d’usage, le client se lassera et partira chercher une autre banque à la minute qui suit !

*Quelle solution digitale pour remédier donc à ça ?*

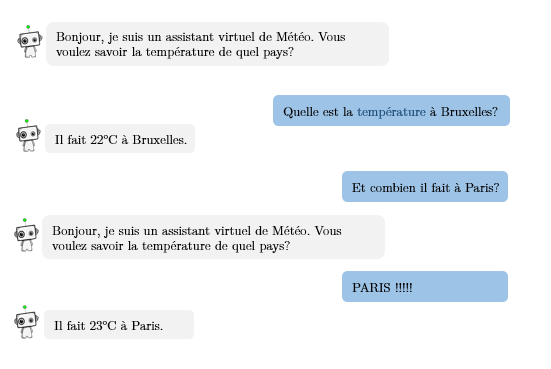
### 2.4) Présentation de la solution proposée

Si l’idée de poser la question à un humain est géniale et bénéfique, **autant le simuler** !

Un **chatbot**, appelé également agent conversationnel, est un bot informatiquecapable d’interagir avec le client et de simuler une conversation comme un être humain.

Les bots informatiques semblent être des phénomènes très récents, mais en réalité, ils reposent sur plus de 70 ans de travail et l’histoire en a beaucoup connu. *(Test de Turing en 1950, ELIZA en 1960, Freddy en 1980, Assistants vocaux en 2011 et au-delà)*.**9**

Au départ, on avait le plus affaire à des bots, appelés *(scriptés)*, qui répondaient à des questions basiques et non compliquées, ne relevant pas du langage naturel humain, car ils suivaient simplement un enchaînement de règles et d’instructions.



**Figure 2 :** Illustration d’une conversation avec un Script bot

Et les bots *scriptés* ont très tendance à frustrer le client !

Au fur et à mesure que l’Intelligence Artificielle s’est développée, les nouveaux bots commencèrent à avoir une forte composante de traitement côté serveur à l’aide des bibliothèques Open Source développées comme Theano, Pytorch et Tensorflow qui leur permet d'accéder à une puissance de calcul massive pour comprendre et répondre aux requêtes reformulées en langage humain.**10**

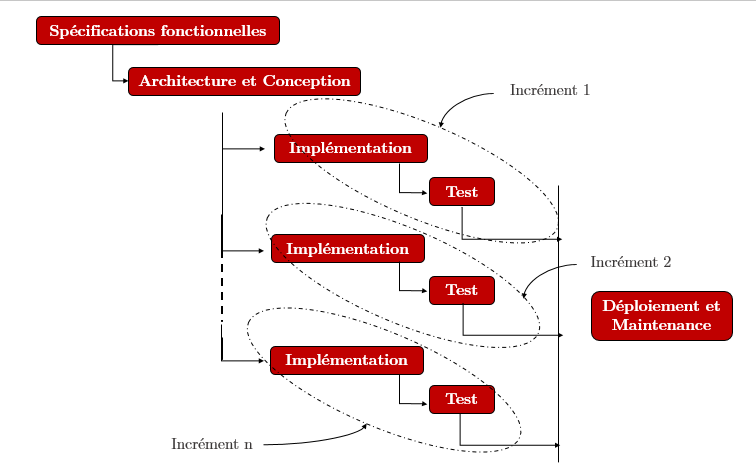
Pour ce projet de fin d’études, l’objectif est donc de réaliser un **chatbot intelligent**, capable de comprendre le langage humain des gestionnaires de la banque, et de leur apporter des réponses adéquates à leurs questions **KYC**.

### 

### 2.5) Conduite du projet

#### 2.5.1) Processus de développement choisi

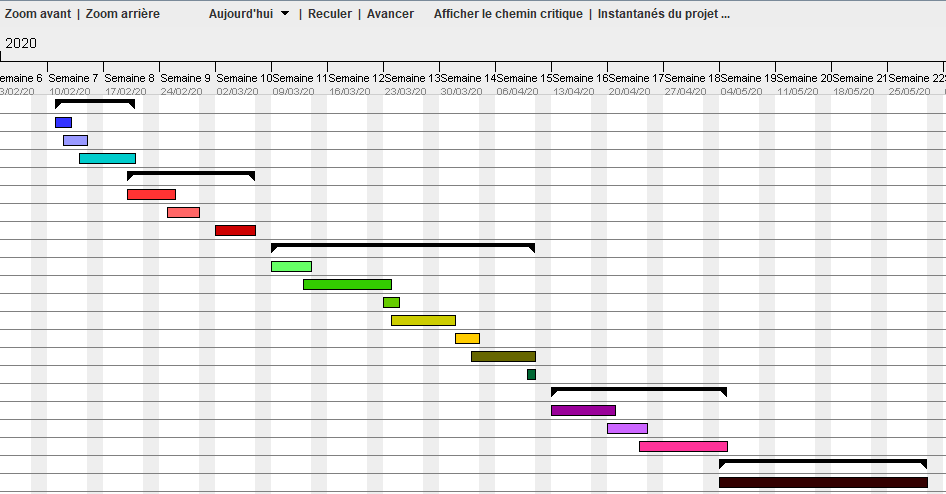
Quoique l’agilité ait le vent en poupe, pour réaliser ce chatbot, le processus de développement choisi est le processus traditionnel et précisément le **modèle cascade** tout en ajoutant une touche **itérative et incrémentale.** Et cecipour les raisons suivantes :

* Les spécifications fonctionnelles *(Questions sur l’instruction KYC et l’outil SOGEKYC)* sont déterminées et ne risquent pas de changer au fur et à mesure que l’application est en développement.
* La problématique est claire et la solution **RASA** est connue et bien déterminée (*à détailler dans les paragraphes qui suivent)*
* Le design restera intact et les modifications ne seront faites qu’au niveau de l’implémentation.
* Pour la banque, il est possible de délivrer une partie du chatbot seulement, et compléter les questions/réponses avec le temps.
* Le risque technologique est quasi-nul.

**Figure 3 :** Processus de développement en cascade

#### 2.5.2) Planification du projet : Diagramme de GANTT

Pour gérer ce projet de fin d’études et visualiser dans le temps les différentes tâches, je propose le diagramme de GANTT suivant :

*Il est à préciser que le travail sur le rapport se fera au fur et à mesure avec le développement du chatbot.*

**Figure 4**: Diagramme de GANTT (Vue générale)



**Figure 5 :** Diagramme de GANTT détaillé

# Chapitre 2 :

# Etude fonctionnelle

Description du chapitre 2

## Cadre du projet

### Résumé du projet

Dans le cadre de l’instruction KYC, afin de mettre tous les collaborateurs sur la même longueur d’onde, l’équipe COO/Solutions – Programme KYC a exprimé les besoins explicites suivants :

* Répondre aux questions des gestionnaires sur l’instruction KYC et sur l’outil SOGEKYC.
* Diriger le gestionnaire vers une personne physique que dans les cas compliqués.

En précisant que l’application souhaitée ne doit pas être scriptée et ne doit pas répondre bêtement aux questions posées, elle doit :

* Comprendre et décrypter le langage humain.
* Répondre sympathiquement et donner l’impression d’une conversation physique réelle.

Afin de subvenir efficacement aux besoins de l’équipe, le livrable potentiel proposé est un chatbot intelligent axé sur l’instruction KYC et également sur l’outil SOGEKYC.

Date de livraison du projet : Ouverte (1ère release : 10/04/2020)  
Statut : Projet concret

### Objectif du projet

L’objectif de ce projet est de constituer un recueil important de l’instruction KYC pour permettre aux gestionnaires de la banque de répondre à leurs questions, et ainsi de leur apporter une aide en cas de besoin.

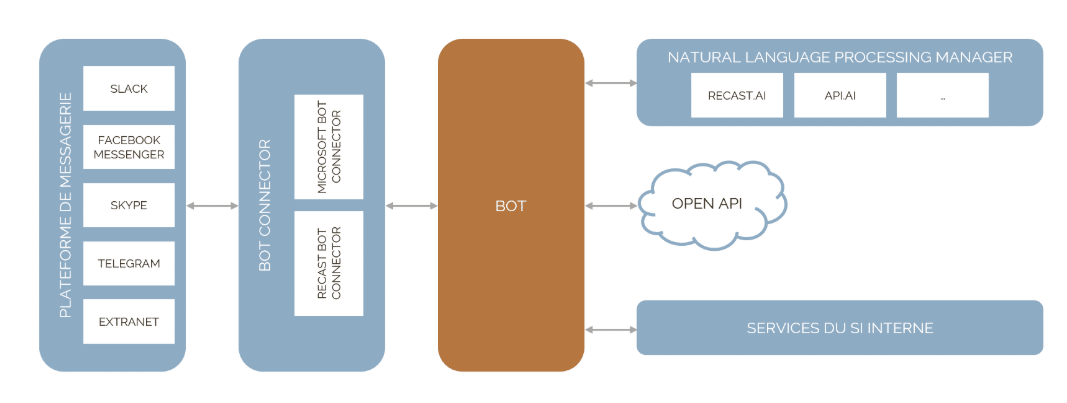
Pour réaliser ce projet, il est important de :

* Comprendre l’architecture d’un chatbot
* Benchmarker sur les technologies Chatbot du marché (Comparaison des coûts des éventuelles licences)
* Avoir une expertise en matière de développement informatique
* Réaliser une bonne modélisation des scénarios

### Idée sur la solution

Avec l’émergence de l’intelligence artificielle, les chatbots ont réussi à prendre un tout nouvel essor du fait de leur capacité à interpréter finement l’intention et l’humeur de l’utilisateur.  
L’interface de messagerie avec laquelle l’utilisateur communique avec le bot ne représente en réalité que son côté front car au centre, le chatbot dispose de plusieurs arbres de décision conversationnels préconfigurés lui permettant de répondre aux requêtes de l’utilisateur.

Et c’est le côté back du chatbot, composé principalement d’un analyseur de langage NLP qui constitue l’intelligence du chatbot, capable d’interpréter la demande et d’en déduire les intentions de l’utilisateur, de plus d’une couche « métier exposée à travers des API.

Requête après requête, cette couche d’intelligence « apprend » et améliore sa compréhension fine et précise de la demande de l’utilisateur.**11**

**Figure 6 :** Architecture du chatbot

*Source :* [*https://www.suricats-consulting.com/lab4us-chatbot/*](https://www.suricats-consulting.com/lab4us-chatbot/)

Certes, un projet chatbot est plus ou moins compliqué et nécessite beaucoup d’expertise dans le domaine.

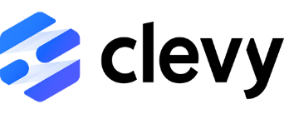
Cependant, il est possible de lancer un bot efficace et rapide en utilisant une **technologie assurant un bon niveau d’abstraction**, sur une plateforme existante, permettant de faire évoluer le bot de manière centralisée.

*En informatique, le concept d'abstraction identifie et regroupe des caractéristiques et traitements communs applicables à des entités ou concepts variés ; une représentation abstraite commune de tels objets permet d'en simplifier et d'en unifier la manipulation.***12**

## Benchmark

****

**Watson** est un programme informatique d'[intelligence artificielle](https://fr.wikipedia.org/wiki/Intelligence_artificielle) conçu par [IBM](https://fr.wikipedia.org/wiki/IBM) dans le but de répondre à des questions formulées en [langage naturel](https://fr.wikipedia.org/wiki/Langage_naturel). Le programme a notamment acquis une notoriété mondiale en devenant en 2011 champion du jeu télévisé américain [Jeopardy!](https://fr.wikipedia.org/wiki/Jeopardy!)**14**

****

Clevy est une start-up spécialisée dans les chatbots conversationnels à destination des entreprises. Récemment, elle a annoncé la sortie officielle du langage open-source **CSML**, destiné principalement au développement des chatbots.

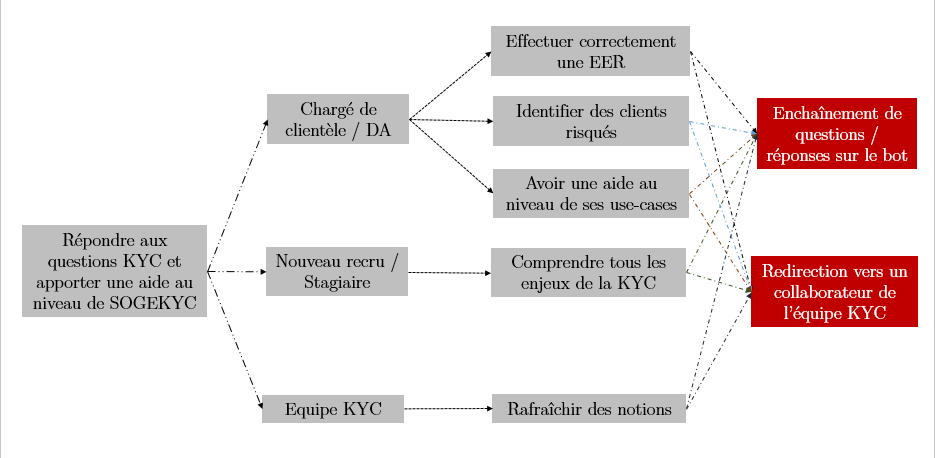
Clevy possède déjà plusieurs clients qui lui font confiance :

Comme mentionné, toutes ces entreprises offrent des chatbots personnalisés, et suivent leur maintenance mais proposent un coût assez élevé.

**L’idée de ce projet de fin d’étude** est de s’inspirer des solutions existantes ou d’utiliser une technologie qui permet une implémentation facile du chatbot et assure sa maintenance avec un coût optimisé.

## Spécifications fonctionnelles

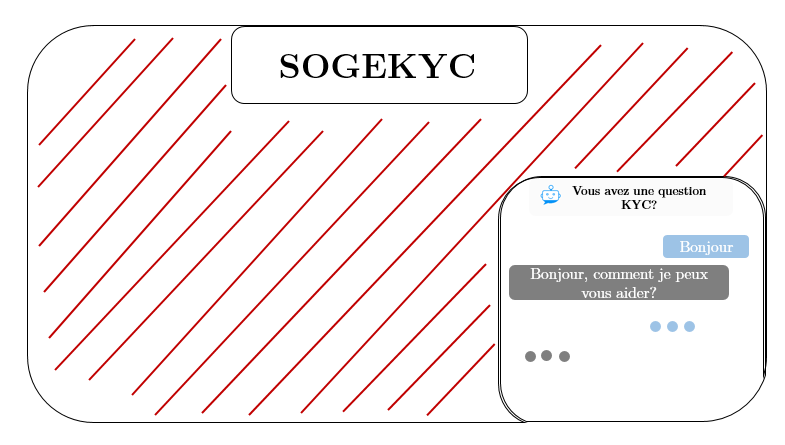
## 3.1. Périmètre fonctionnel



**Figure 7 :** Spécifications fonctionnelles du chatbot

### 3.2. Aperçu du chatbot

**Figure 8 :** Maquette simple du chatbot



**Figure 8 :** Maquette simple du chatbot

## Spécifications techniques

### 4.1. Choix technologiques

|  |  |  |
| --- | --- | --- |
| Besoins | Contraintes | Solutions |
| * Répondre aux questions intelligemment * Avoir un chatbot personnalisable | * Contraintes temporelles et budgétaires * La maintenance du chatbot doit être facilement assurée | * Solution chatbot open-source, facilement intégrable et maintenable, avec des fonctionnalités préconfigurées plutôt que de développer le tout à zéro. |

En fonction des critères présentés dans la score-card suivante, il convient la solution RASA pour réaliser le chatbot intelligent.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Facilité de l’implémentation (0,3)** | **Maintenance (0,3)** | **Intelligence (0,4)** | **Total  / 1** |
| **CSML** | 0.7 | 0.5 | 0.2 | 0,44 |
| **RASA** | 0,8 | 0,8 | 0,8 | **0,8** |
| **Web API (ex : IBM Watson)** | 0,2 | 0,2 | 0,8 | 0,44 |

**Figure 9 :** Choix technologiques pour la réalisation du chatbot

### 4.2. Hébergement

Pour le test, la plateforme sera hébergée sur une machine virtuelle (VM) sur Azure, Google Cloud (GCP) ou sur Amazone (AWS).   
Le nom de domaine n’est pas obligatoire.

### 4.3. Maintenance et évolution

La maintenance du chatbot est facilement accessible de la plateforme qui sera hébergée plus tard sur le serveur de la banque.   
Presque aucune connaissance de développement n’est requise.

# Chapitre 3 :

# Conception du chatbot

Description du chapitre 3

## La technologie RASA

### Présentation

RASA est une entreprise allemande fournissant une technologie pour la création de chatbots. Elle propose une plateforme pour l’intégration de son agent conversationnel, notamment en fournissant des conteneurs Docker qui permettent de le déployer rapidement et de passer à l’échelle industrielle.**15**

Pour construire ces chatbots, RASA a développé deux Frameworks d’Intelligence Artificielle, basés sur du Machine Learning et a décidé de les rendre open-source afin de les améliorer constamment.

* RASA NLU**:** Gère la partie compréhension du langage de l’agent
* RASA CORE **:** Gère la partie stockage d’information, et la connexion avec l’interface de discussion.

### Fonctionnement global de RASA

Pour permettre à un chatbot de communiquer avec des utilisateurs en utilisant le langage naturel, tout en répondant de manière pertinente, il doit extraire des informations, analyser des sentiments, traduire et analyser des contextes.

Le **NLU** est la sous-entité du **NLP** qui est axée sur la compréhension de phrases. Le système basé sur le **NLU** effectue un « mapping » de langage naturel en une **représentation intention/entités**, de la même façon qu’une classification.

*Les intentions sont des classes d’énoncés qui correspondent souvent au verbe utilisé par l’utilisateur pour écrire l’action qu’il souhaite faire.*

*Les entités sont des sous-unités de la phrase écrite contenant des informations clés utilisables dans les applications.*

Sur **RASA,** ces deux principaux concepts **intention/entités** sont présents sous leurs terminologies anglaises « Intent » et « Entity ».

Il existe également des concepts comme **les slots, les actions et les stories.**

*Les slots sont les dépôts de valeurs importantes dont le chatbot se rappellent au long de la conversation.*

*Les actions sont les opérations qu’effectue le chatbot afin de répondre aux requêtes des utilisateurs. Elles peuvent être très basiques comme elles peuvent être compliquées.*

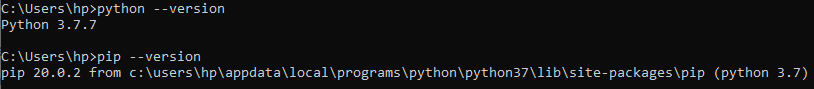
*Les stories sont les exemples de discussions créées à partir de l’enchaînement d’intentions de l’utilisateur et d’actions prises par l’agent selon les intentions reçues.*

### Installation de RASA Open Source et de RASA X

L’installation de RASA est disponible pour 3 OS : **Linux Ubuntu – Mac OS – Windows.**

**Dans ce projet de fin d’études,** RASA Open Source a été installée sous **Windows**. Par contre, pour l’installation de RASA X, le système d’exploitation utilisé sur la VM est **Linux Ubuntu.**

RASA Open Source :

Pour RASA Open Source, il faut d’abord mettre à jour le gestionnaire de paquets **pip** qui permet de gérer les paquets écrits en **Python**, puis installer une version récente de **Python** compatible avec les composantes de **RASA**. Dans ce cas, la version installée est **3.7.7**.

**Figure 10 :** Installation de Python et de la bibliothèque Pip

*Il est intéressant de créer un environnement virtuel pour l’installation afin d’empêcher les erreurs de configuration.*

Ensuite, avec la simple commande de **pip install rasa**, RASA Open Source est installée et prête à être utilisée.

*Autrement, si l’une des commandes ne marche pas, il est possible de « pull » toute l’installation de RASA à partir de github.*

RASA X :

Après RASA Open Source, vient l’étape supérieure RASA X qui facilite le déploiement du chatbot sur la plateforme offerte par RASA.

Afin de faciliter l’installation de RASA X, il faut disposer d’une machine avec Linux Ubuntu ou Debian comme système d’exploitation, en plus de :

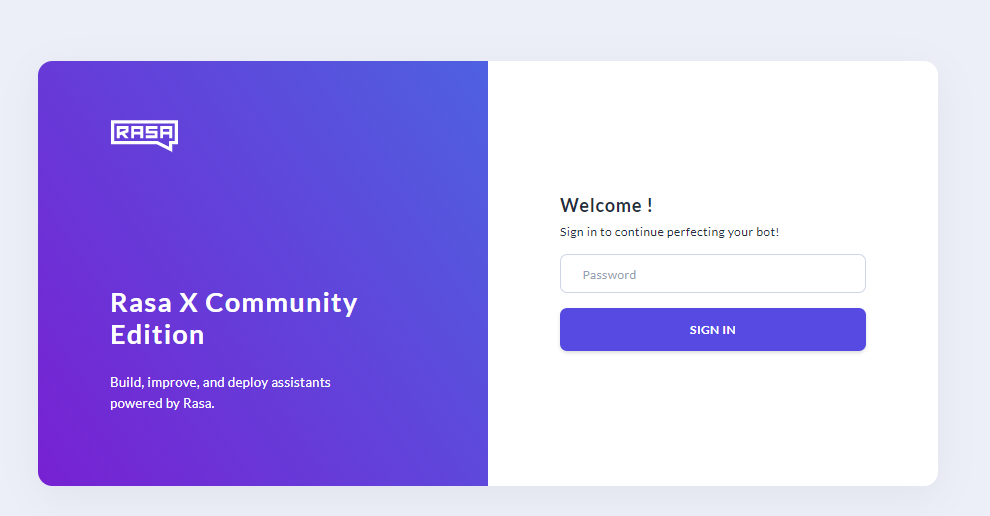
* 2 unités centrales virtuelles
* 4 GB de RAM
* 100 GB d’espace disque dur.

Au minimum.

Dans notre cas, la machine virtuelle hébergée sur Microsoft Azure tient en compte toutes ces spécifications et fonctionne avec le système d’exploitation Linux Ubuntu.

RASA X est déployée sous forme de plusieurs conteneurs Docker, et est en plus gérée par plusieurs APIs stockées sur Google Cloud Storage.

L’installation est un peu fastidieuse. Toutes les étapes sont disponibles sur <https://rasa.com/docs/rasa-x/installation-and-setup/docker-compose-script/>

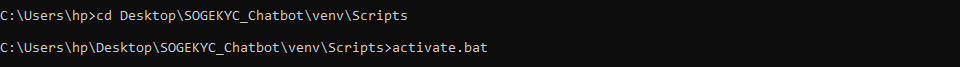
Après avoir installé RASA X sur serveur, on obtient la plateforme suivante :

**Figure 11 :** Plateforme de RASA X

## Test de RASA sur une réelle conversation :

### 2.1. A partir du terminal :

Afin de tester un chatbot très basique à partir du terminal de l’ordinateur, il faut :

1. **Activer l’environnement virtuel**

**Figure 12 :** Environnement virtuel Python

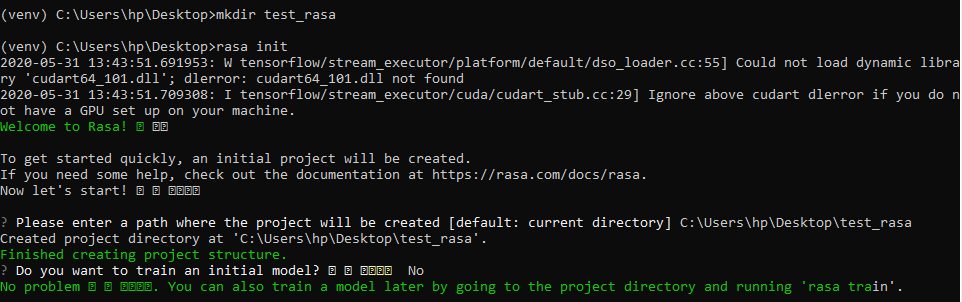
Dans ce projet, l’environnement virtuel est créé sous **venv.**

1. **Créer un dossier pour le test**

Dans le dossier test\_rasa, nous allons intégrer tous les fichiers relatifs à l’implémentation du chatbot de test.

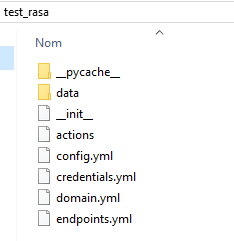
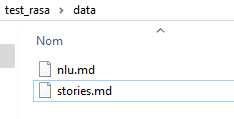
**Figure 13 :** Création de nouveau dossier en ligne de commande

1. **Insérer les fichiers d’implémentation dans le dossier**

Pour insérer tous les fichiers, il suffit de taper la commande **rasa init.**

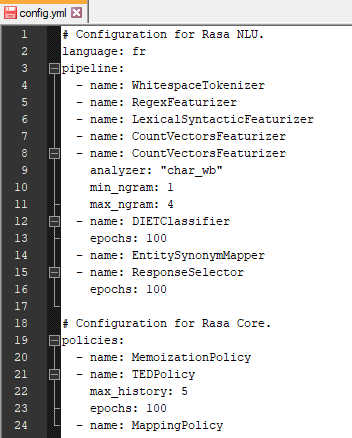
**Figure 14 :** Préparation du dossier de test sous RASA

Tous les fichiers sont ainsi créés.



**Figure 15 :** Aperçu des fichiers de test

1. **Configurer le chatbot**



**Figure 16 :** Configuration du chatbot

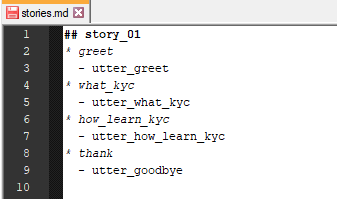
Pour comprendre les intentions des utilisateurs, pouvoir les classifier, et les relier avec leurs réponses adéquates, plusieurs options préconfigurées sont offertes par RASA comme les **Word Vector Sources** qui prédisent le contexte des mots, les **tokenizers** qui utilisent le principe de la segmentation pour convertir les chaînes de caractère en une liste de symboles ou encore les **Intent Classifiers** qui permettent de classifier les intentions des utilisateurs etc.

Toutes ces options regroupent des éléments préconfigurées qu’il suffit seulement de comprendre pour suivre le raisonnement du chatbot au cours de ses réponses et lui corriger en cas d’erreur, afin de maximiser sa performance.

Comme les questions de notre chatbot seront posées en français, il suffit de changer le terme « en » en « fr ».

1. **Implémenter une story**

Tous les fichiers sont reliés, et chaque fichier dépend de l’autre. Cependant, l’ordre d’implémentation des fichiers n’est pas important et ne change rien au niveau de la performance du chatbot, mais commencer par implémenter les stories est une meilleure pratique afin de ne rien oublier sur les autres fichiers et entretenir un fil conducteur quand les choses deviennent plus compliquées.



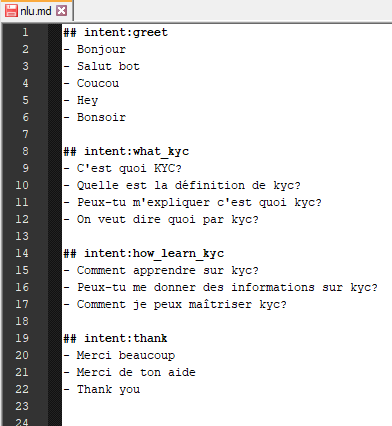
L’implémentation des stories sur RASA est faite par défaut en anglais. Il est important d’utiliser des mots clés qui relèvent les intentions des utilisateurs à partir de leurs questions.

* Greet : Bonjour
* What\_kyc : C’est quoi KYC ?
* How\_learn\_kyc : Comment je peux apprendre sur KYC ?
* Thank : Merci

**Figure 17 :** Fichier des stories de test

1. **Implémenter les intentions**

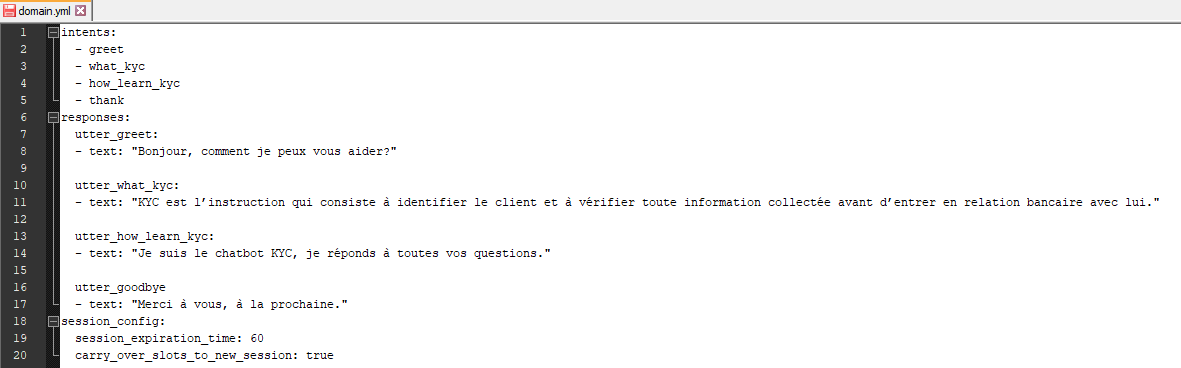
Toutes les intentions citées dans les stories doivent exister dans le fichier nlu.md sous plusieurs exemples pour apprendre à la machine les différentes formulations.  
Plus on ajoute d’exemples plus le chatbot devienne plus intelligent.



Cependant, au fur et à mesure que les choses deviennent compliquées, le chatbot pourrait mélanger quelques intentions. Il faut être très prudent !

**Figure 18 :** Fichier des intentions de test

1. **Implémenter le domain**

Le fichier domain.yml contient toutes les réponses et les leurs intentions correspondantes.

**Figure 19 :** Fichier des réponses aux intentions de test

Conclusion :

Plus on ajoute de stories, plus on ajoute d’intentions, plus le chatbot sait répondre.

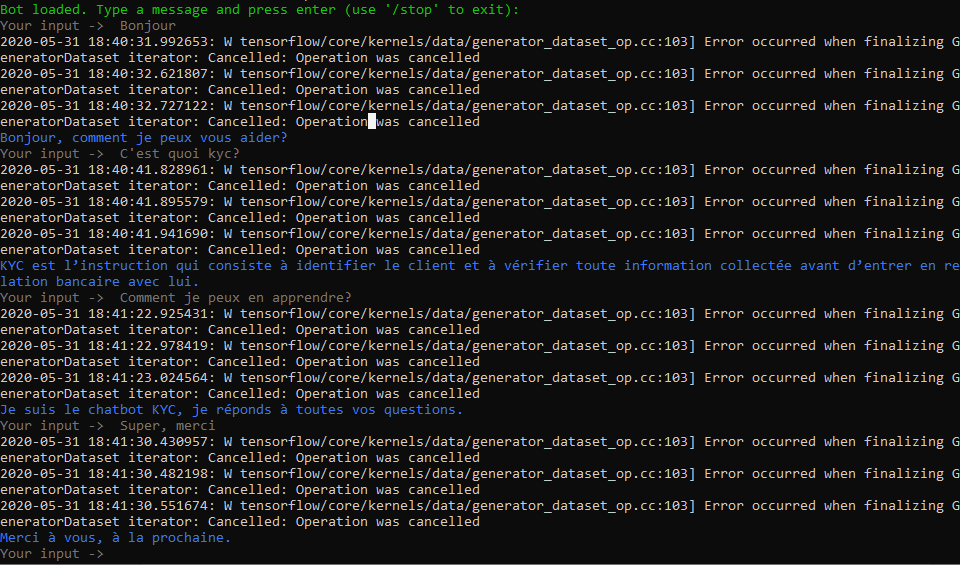
*L’implémentation ci-dessus est très basique par rapport à toutes les fonctionnalités qu’offre RASA.*

1. **Entraîner le chatbot**

Après avoir fini l’implémentation basique de tous les fichiers, il faut entraîner le chatbot et lui donner un peu de temps pour apprendre ses données afin de savoir nous répondre.  
  
Pour ceci, il suffit de taper la commande **rasa train,** puis **rasa shell** pour commencer la discussion.

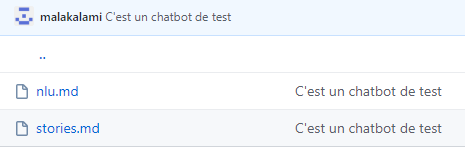
**Figure 20 :** Entraînement du chatbot en ligne de commande

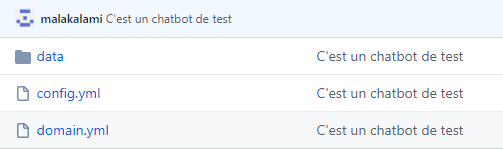
1. **Essayer le chatbot**



**Figure 21 :** Test du chatbot

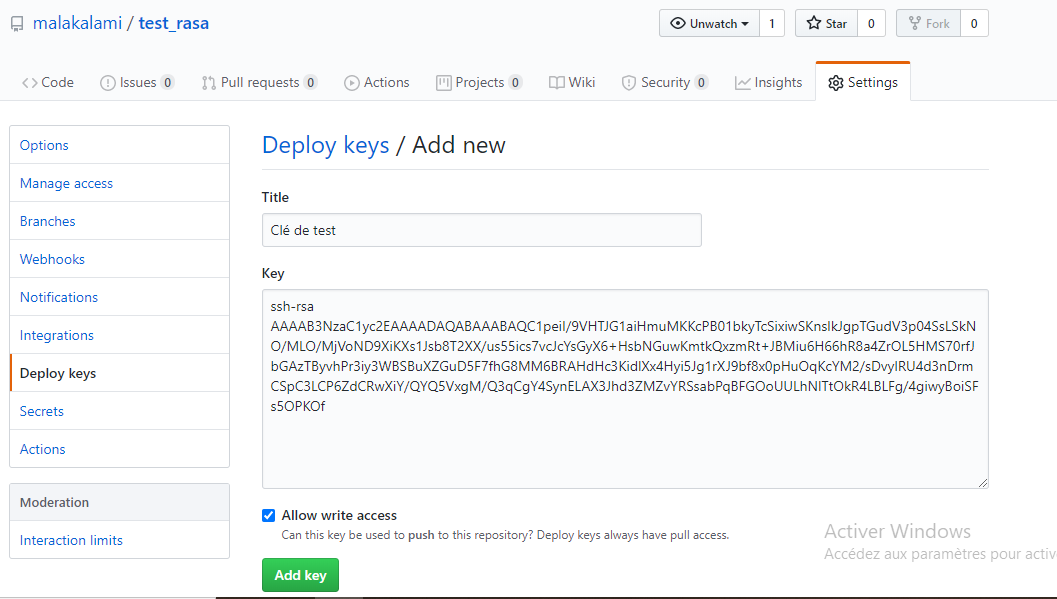
### 2.2. Sur RASA X

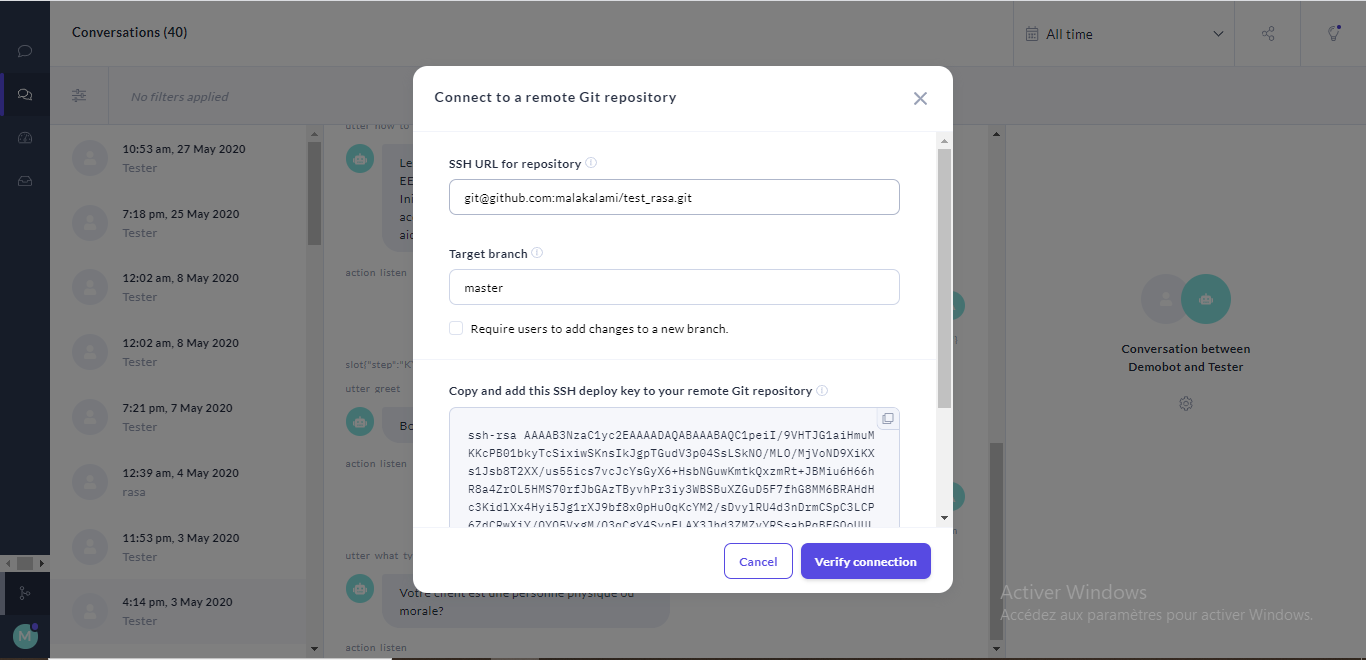
  
Pour essayer le chatbot sur RASA X, il faut d’abord déposer tous les fichiers qu’on a utilisés sur **github** puis le connecter à RASA X.



**Figure 22 :** Importation des fichiers sur github

Afin de connecter le chatbot sur RASA X, on saisit d’abord l’adresse IP du serveur qui est dans notre cas <http://20.188.40.162/>, et on y saisit le mot de passe pour entrer à la plateforme.

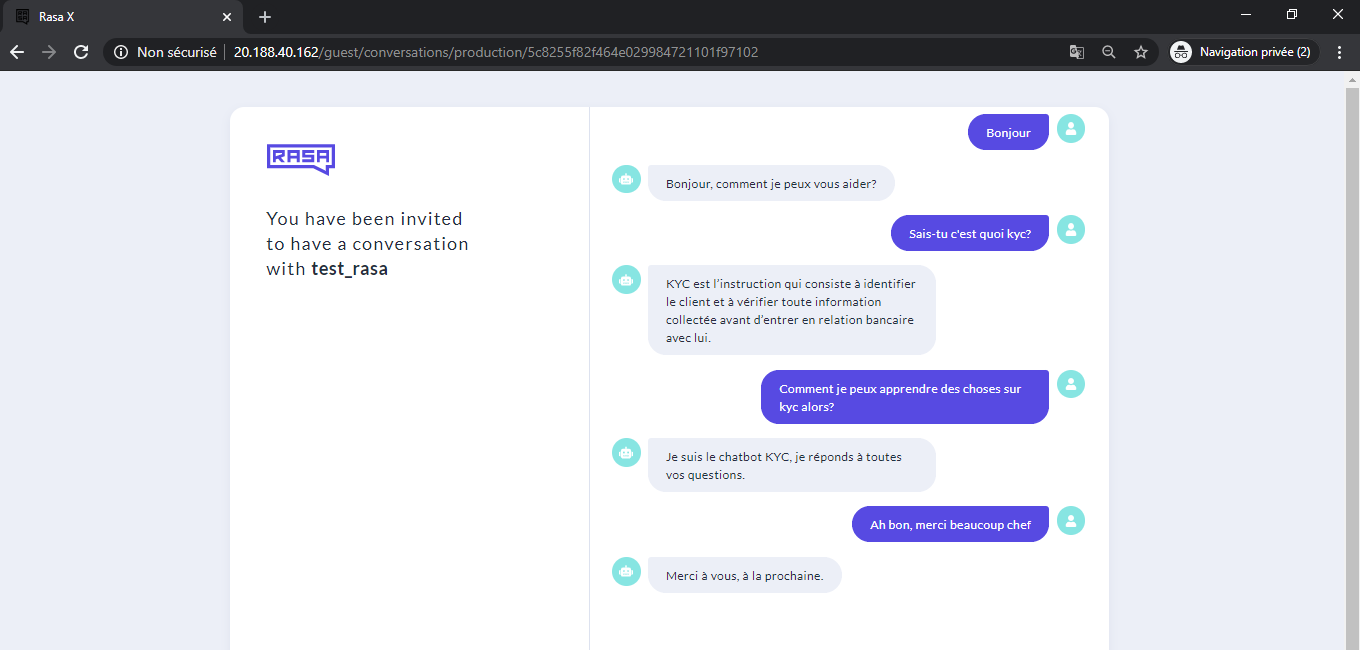




**Figure 23 :** Connexion du chatbot sur github avec RASA X

**Figure 23 :** Connexion du chatbot sur github avec RASA X

La prochaine étape est d’entraîner le chatbot en cliquant sur **train** puis d’activer le modèle correspondant.

Finalement, le chatbot est prêt à être essayé !

**Figure 24 :** Test du chatbot sur l’interface de conversation

Et le chatbot répond effectivement et sans erreur aux questions qu’il a apprises.

# Chapitre 4 :

# Production du chatbot

Description du chapitre 4

## Structuration des questions KYC

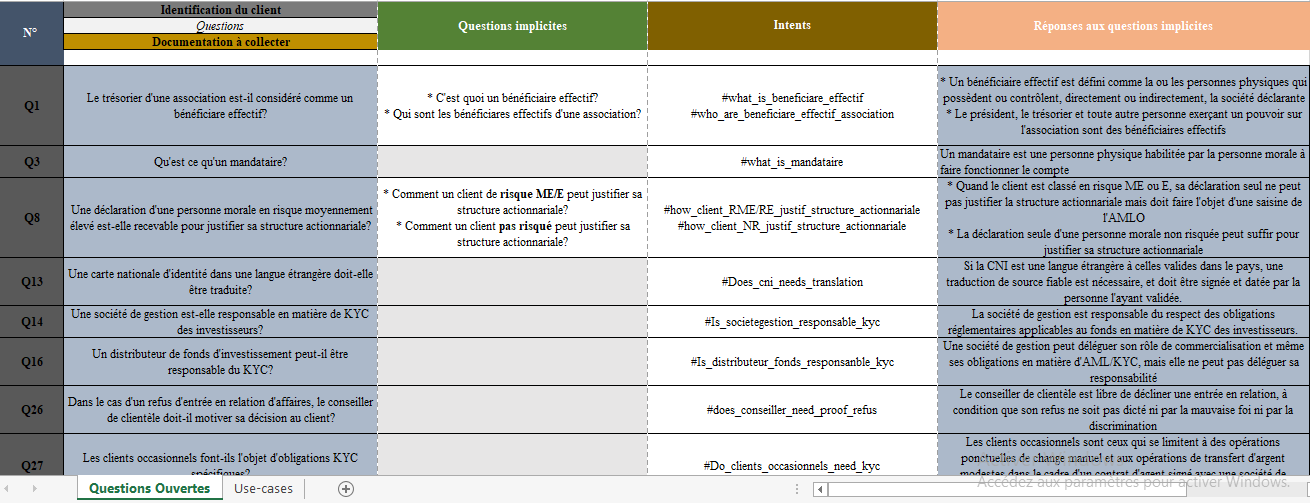
Tout d’abord, afin de gérer les différents jets du chatbot, il est intéressant de ne pas mélanger les questions KYC des **4 processus** et d’en saisir au hasard.  
  
Vu les circonstances de ce projet de fin d’études et le temps qui lui est accordé, l’Entrée en relation est le seul processus géré. De la même manière, avec plus de temps, on peut implémenter les autres processus et ajouter autant de questions que l’on veut.

En plus, il est très important d’instaurer une bonne logique entre les questions de manière à garantir la bonne performance de ce chatbot, ainsi que la lisibilité du contenu à la prochaine maintenance.

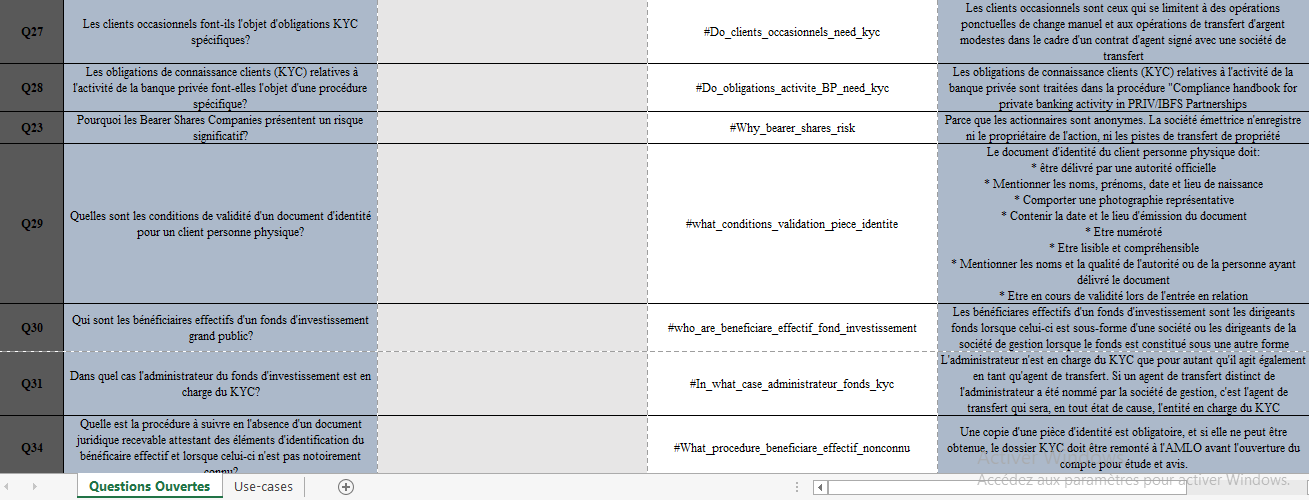
Sous ce fichier Excel, par le simple souci d’optimisation, les questions du premier processus ont été séparées en 2 catégories :

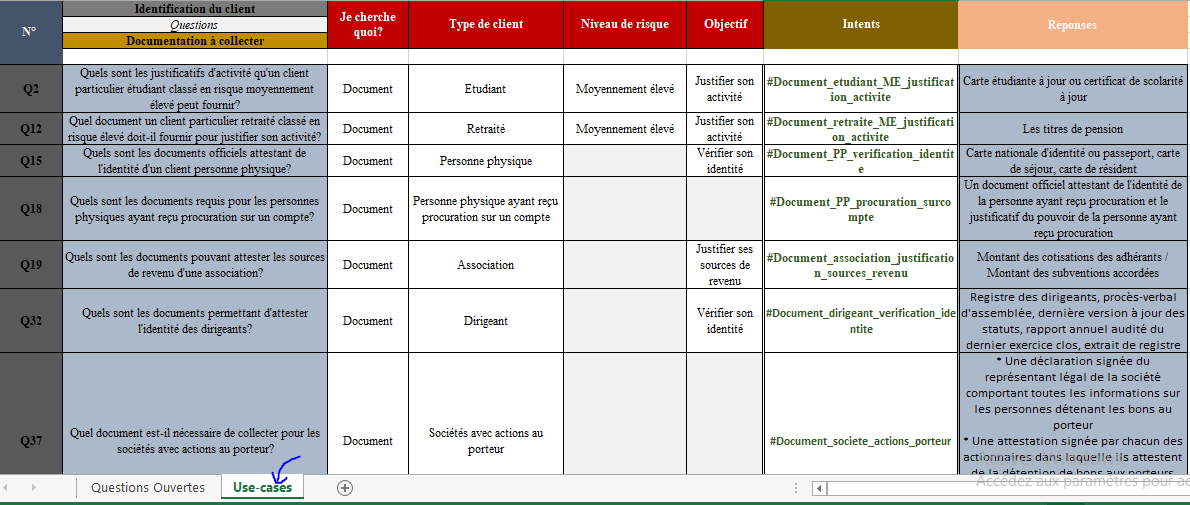
**Questions Ouvertes** **:** comme les définitions et les questions classiques …  
**Use-cases :** Ce sont les différents cas que les gestionnaires peuvent rencontrer au cours de leurs entrées en relation, et elles sont innombrables.

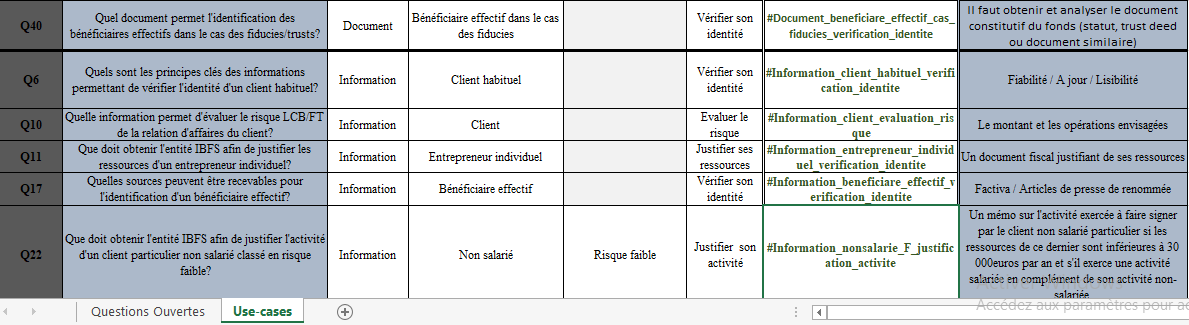
*Quand le nombre des use-cases deviendrait énorme, on pourrait fusionner des cas et c’est à ce moment là où cette catégorisation serait très bénéfique.*



**Figure 25 :** Questions ouvertes à implémenter sur le chatbot



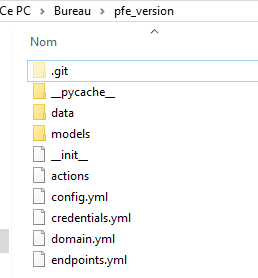
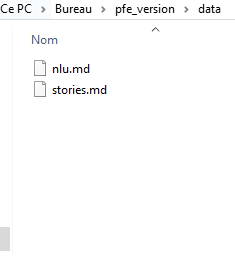




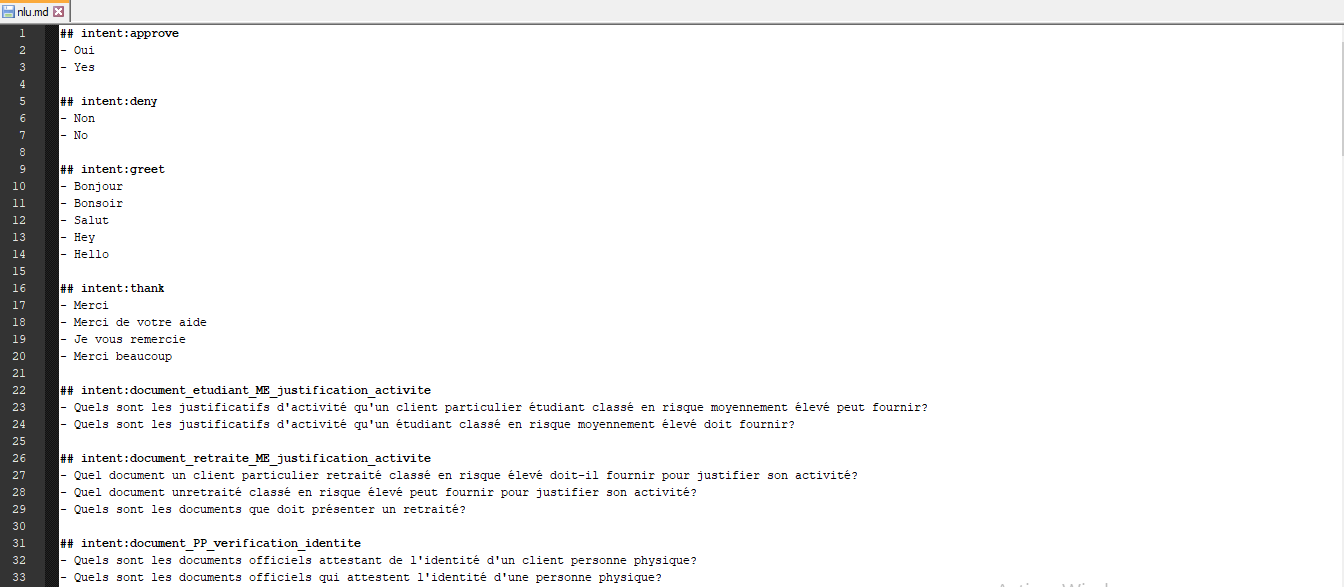
**Figure 26 :** Use-cases à implémenter sur le chatbot

## Implémentation du contenu sous RASA

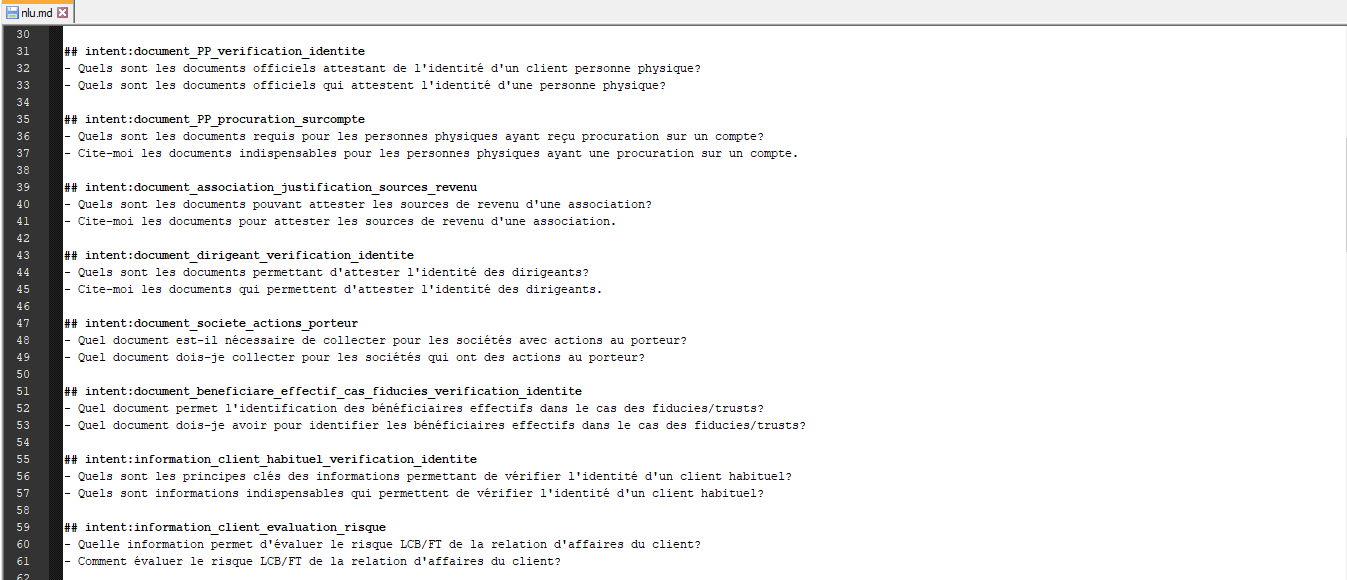
Exactement comme on a fait dans la phase de test, le dossier hébergeant notre chatbot se compose des fichiers suivants :



**Figure 27 :** Dossier de travail

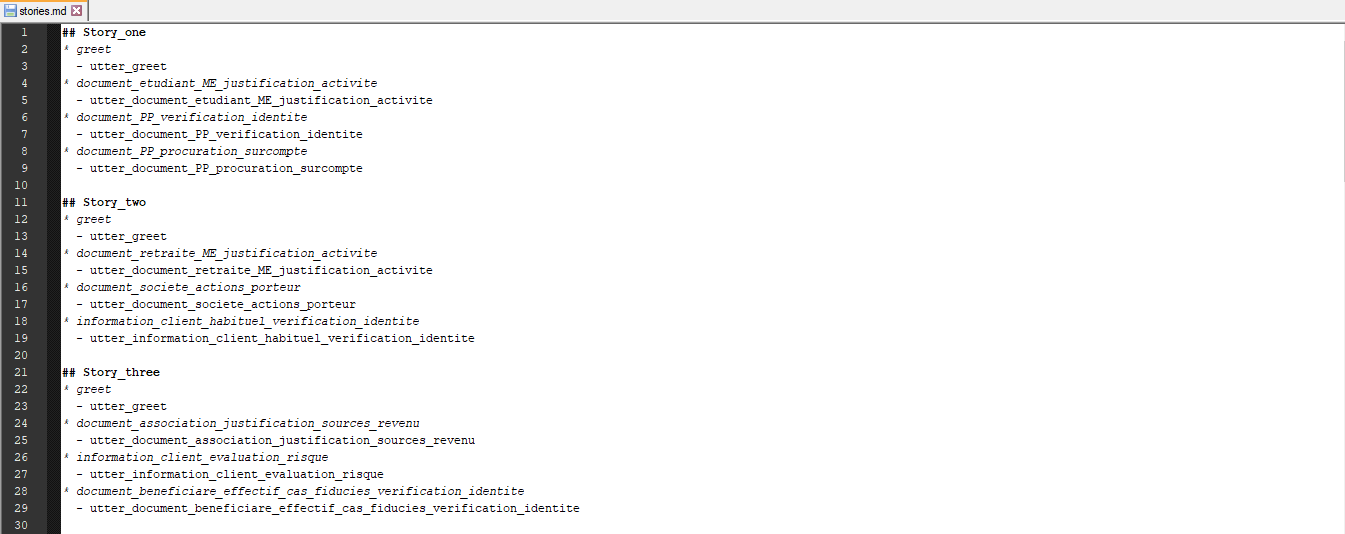
A l’intérieur du fichier **nlu.md,** sont regroupées toutes les intentions déjà citées sur le fichier Excel avec 2 reformulations de chaque question. *(Il est possible d’ajouter d’autres reformulations)*

**Figure 28 :** Intentions du chatbot implémentées

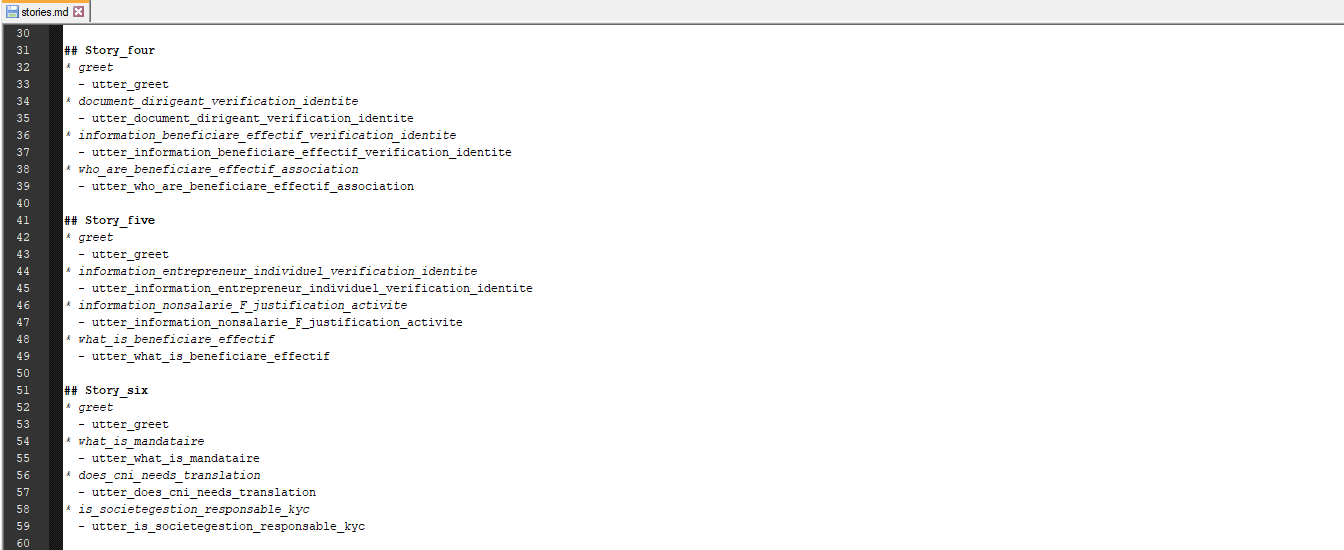


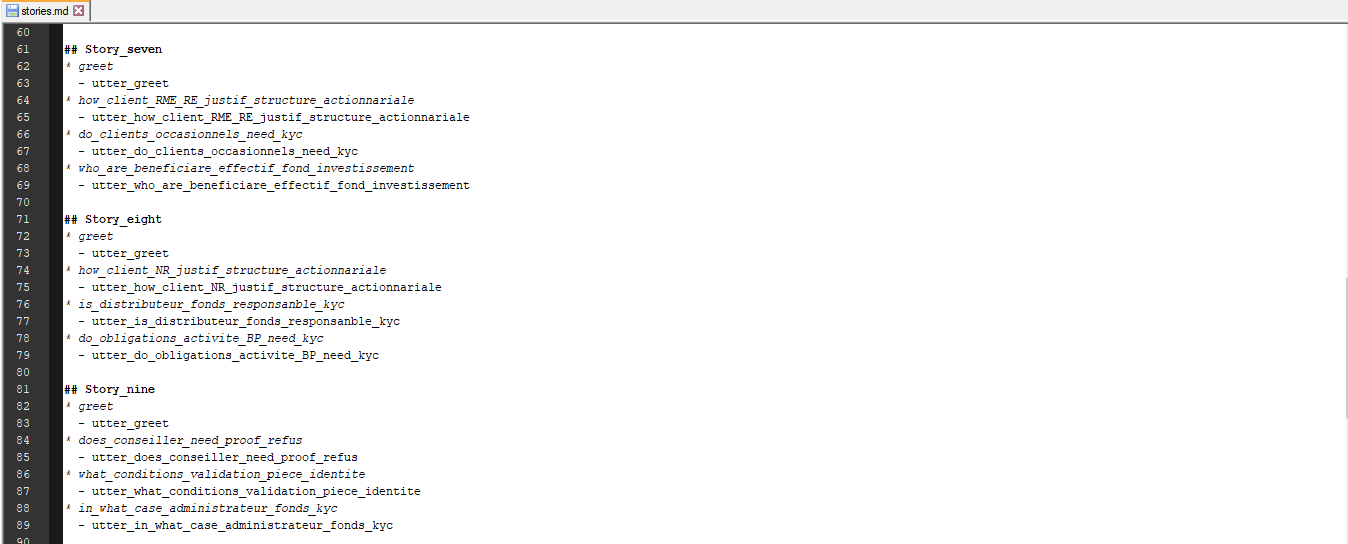
Et encore plus …

A l’intérieur du fichier **stories.md** sont regroupées les différentes conversations possibles, regroupées aléatoirement puisque les questions ne sont pas liées.

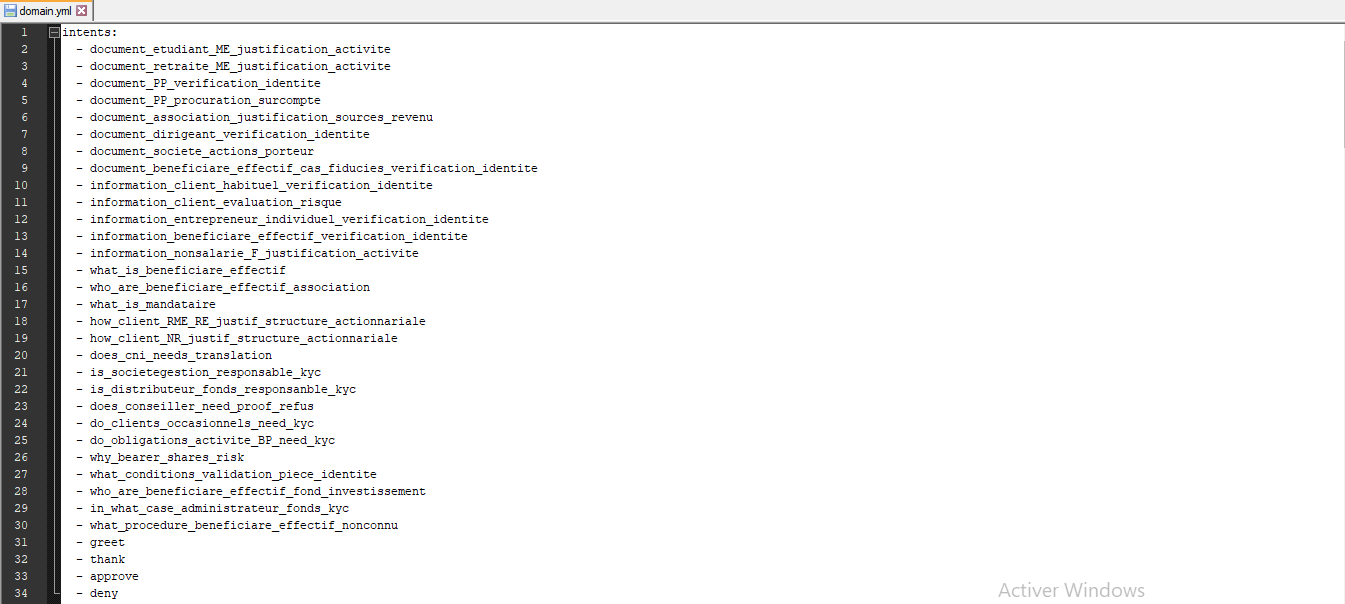


**Figure 29 :** Stories du chatbot implémentées

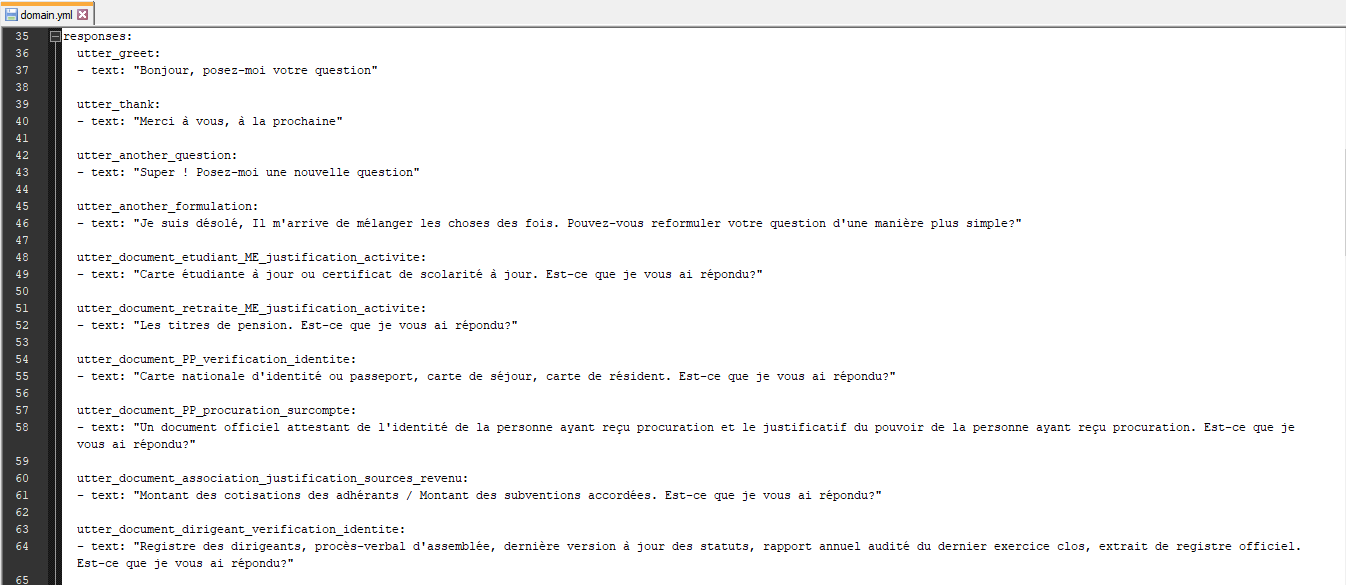


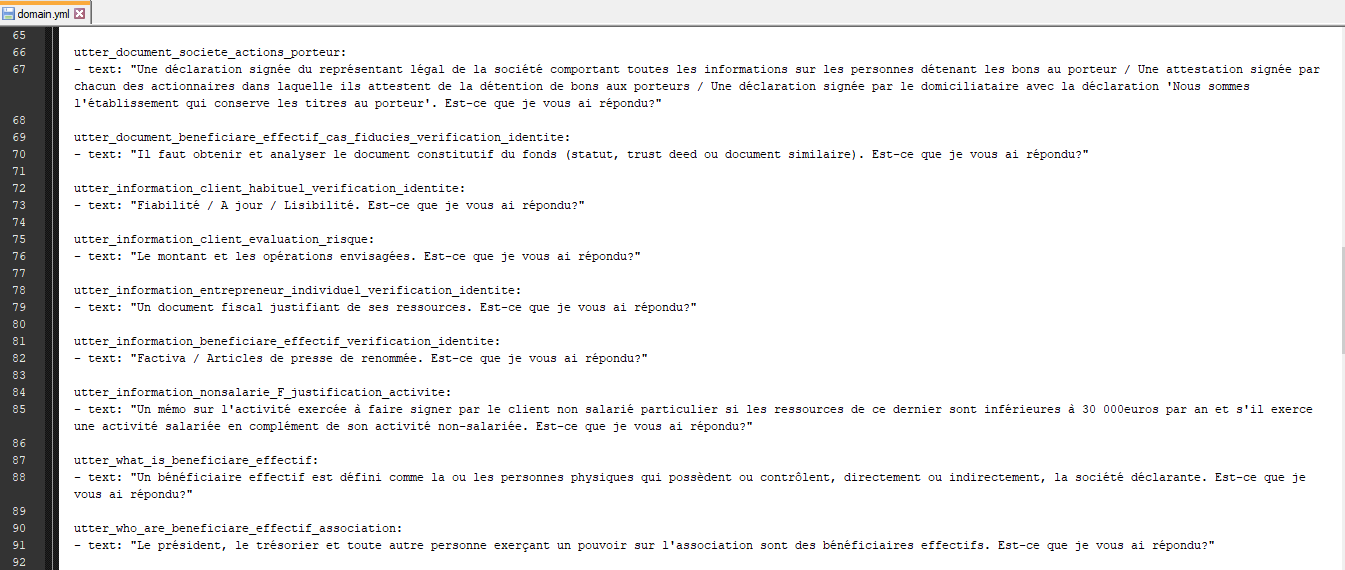


Et encore plus …

A l’intérieur du fichier **domain.yml,** on retrouve toutes les intentions listées mais non explicitées cette fois-ci en plus des réponses associées à chaque intention.

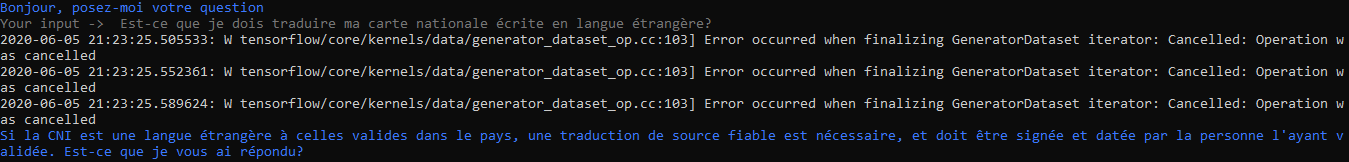
**Figure 30 :** Réponses aux intentions implémentées





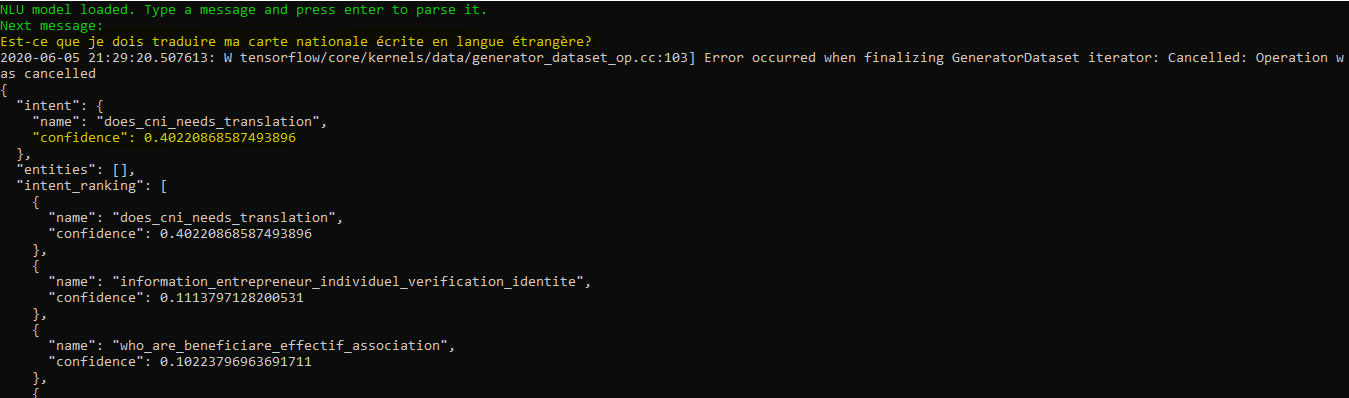
Et encore plus …

## Test du chatbot sous RASA Open Source

  
Quelques instants après l’entraînement du chatbot, le test a relevé des résultats positifs en répondant correctement à toutes les questions qu’il a apprises.

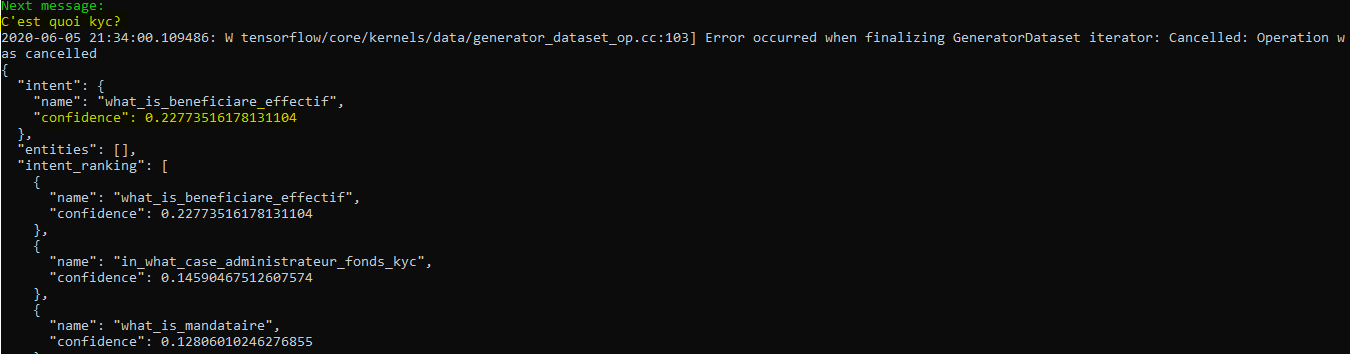
**Figure 31 :** Test du chatbot à partir du Terminal

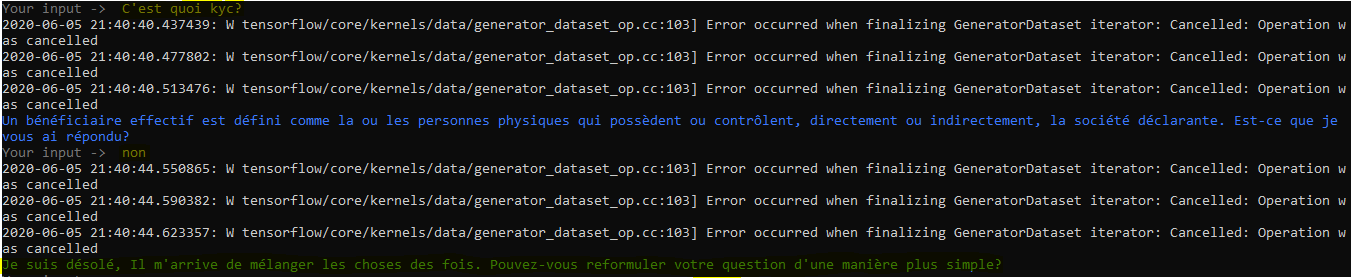
Tout d’abord, on peut remarquer que la syntaxe de la question *« Est-ce que je dois traduire ma carte nationale écrite en langue étrangère ? »* est un peu différente de celle dictée dans le fichier des intentions *« Une carte nationale d'identité dans une langue étrangère doit-elle être traduite ? ».*Pourtant le chatbot en a deviné la réponse avec un **niveau de confiance égal à 40.2%,** bien plus que toutes les autres intentions.

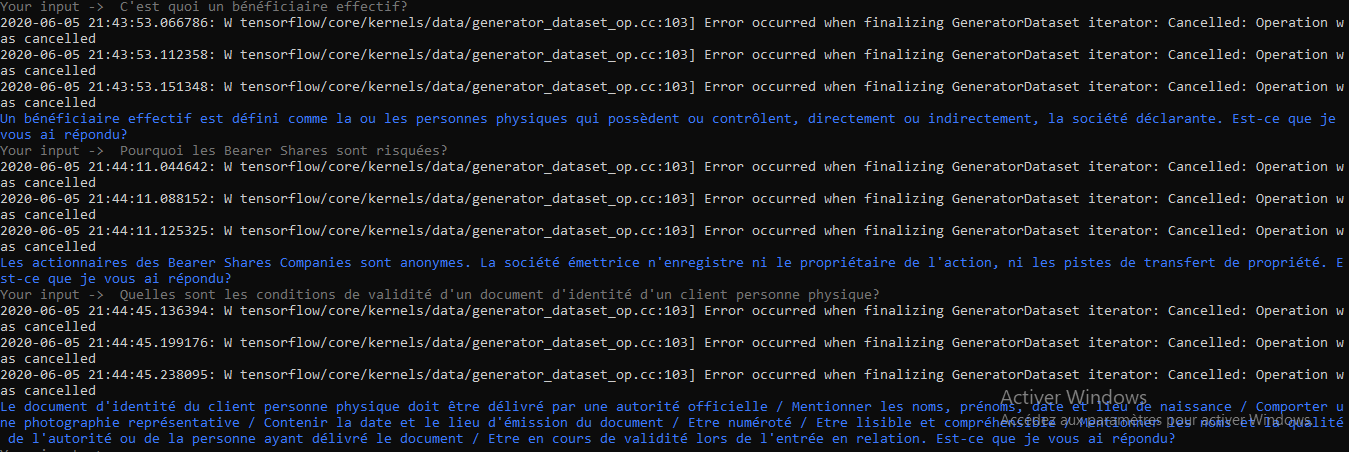
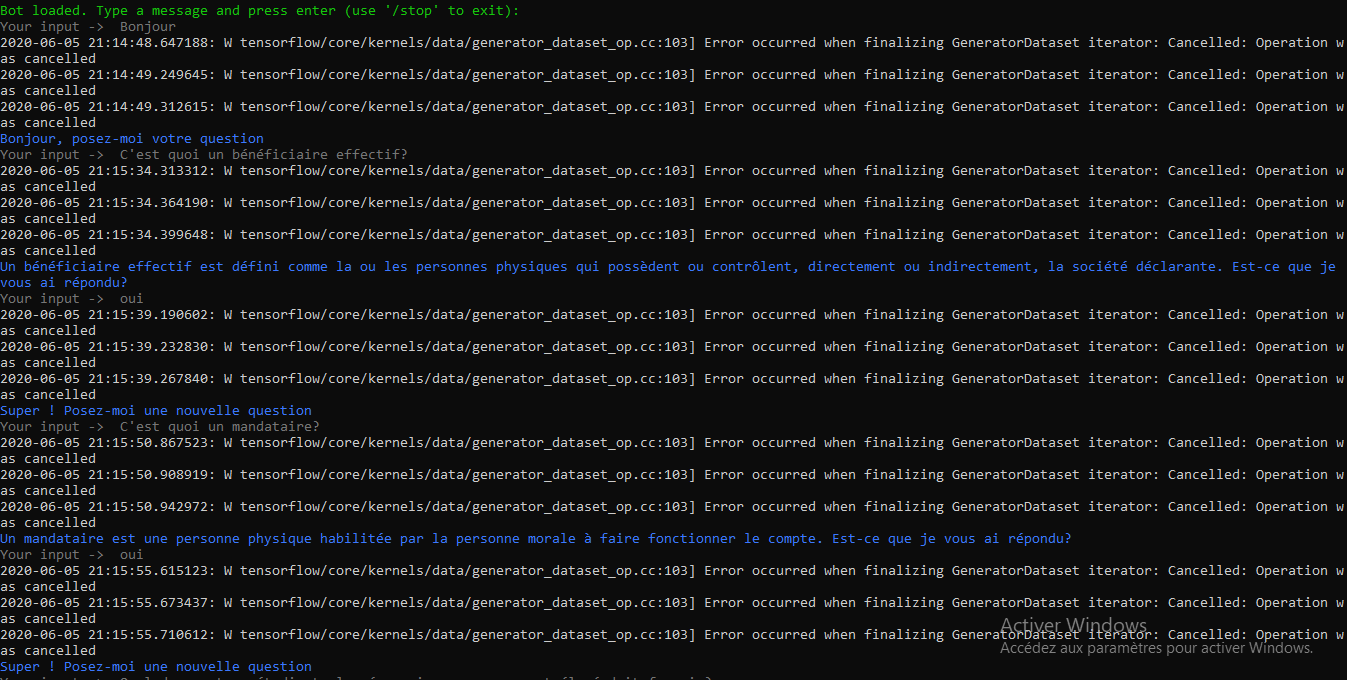


**Figure 32 :** Niveau de confiance associé aux intentions testées

Plus on ajoute de reformulations des questions, plus le chatbot enrichit son dictionnaire et augmente son niveau de confiance par rapport aux réponses.

Parallèlement, on peut remarquer aussi que la question *« C’est quoi KYC ? »* n’existe pas sur le fichier Excel, et pour remédier à ça, le chatbot devine sa réponse en l’approchant à une autre question qui existe déjà avec un **niveau de confiance égal à 22.7%** cette fois-ci.

Ce qui a conduit à la réponse suivante :

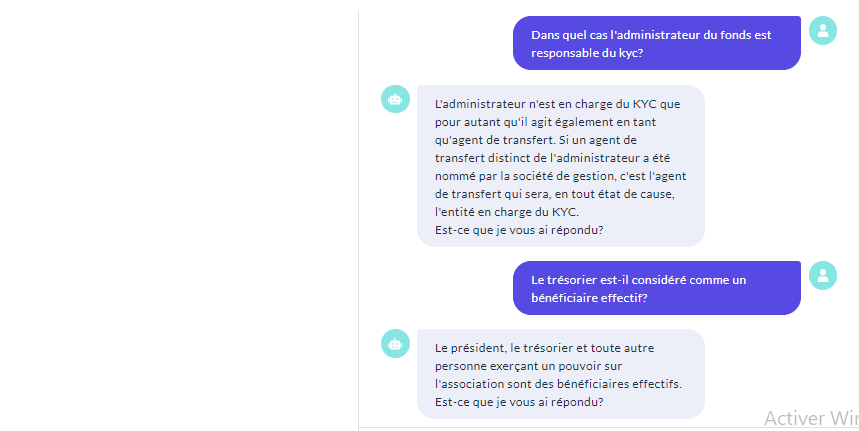
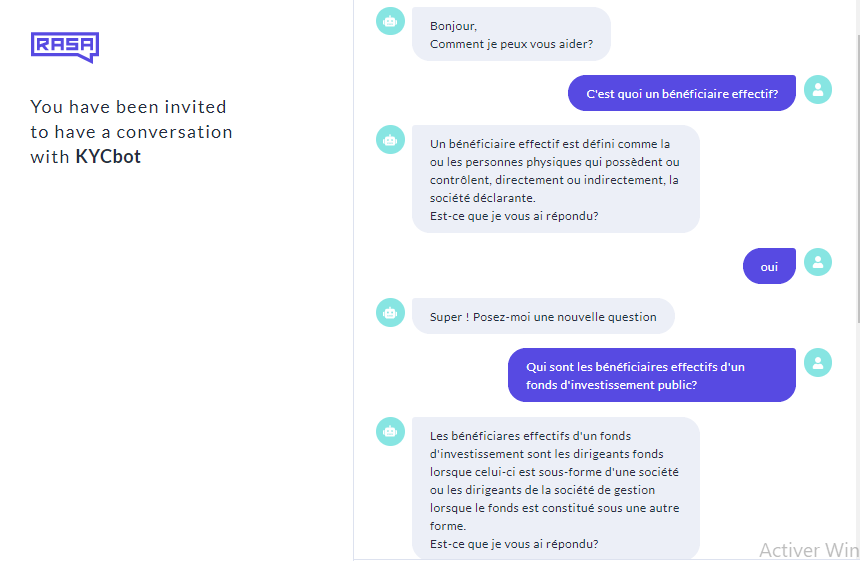
Mises à part toutes ces exceptions, le chatbot répond à toutes les autres questions décrites sur le fichier Excel, correctement et sans faute.

**Figure 32 :** Réponses du chatbot aux différentes intentions

Et encore plus …

## Plateforme du chatbot sous RASA X

Comme convenu dans la phase de test, les fichiers sont d’abord enregistrés sur **github.com** puis le chatbot est disponible avec une vraie plateforme et répond correctement aux questions !



**Figure 33 :** Test final du chatbot sur l’interface de conversation

# Chapitre 5 :

# Exploitation et Maintenance du chatbot

Description du chapitre 4

1. Installation de RASA et de RASA X sur le serveur de la banque
2. Code JavaScript
3. Maintenance du chatbot à partir de la plateforme de RASA X