# Introduction de chapitre

Contexte

Définition

Problématique

Plan

# Spécifications des besoins

Premièrement avant de le démarrage de projet informatique, il y a le cahier des charges. C’est un document qui retrace les fonctionnalités de l’application qui doivent être implémenter. Pour cette partie, nous allons dans la peau du maitre d’ouvrage et rédiger les spécifications de tout le travail que nous allons faire. Il n’y aura rien de complexe, juste une rédaction simplifiée des étapes auxquelles nous devons passer pour réussir à créer un modèle intelligent qui pourra prédire une analyse financière, réussir à développer un chatbot et présenter tout cela dans une interface graphique.

## Contexte de l’application

Le domaine de la finance est très vaste, il y a beaucoup de professionnels, d’ingénieurs, de chercheurs qui s’y illustre. La finance est une science qui vise à une meilleure gestion du portefeuille des entreprises, que cela soit l’investissement, le financement…

C’est ainsi que nous introduisons l’analyse financière qui nage dans cette vaste mer domaine de la finance. L’analyse financière ou le diagnostic financier nous permet de comprendre le processus de création de richesse, de voir si l’entreprise est dans une sante financière et j’en passe.

Aujourd’hui cette analyse financière se fait par des logiciels généralistes, ou même des fois sur feuille ce qui constituent un travail fastidieux. Malgré le fait de que ces logiciels peuvent être d’une grande importance, ils souffrent souvent du trop-plein information. Il y a toujours des fonctionnalités dont nous n’aurions jamais besoins si nous les utilisons pour faire une analyse financière. Autre chose, ces logiciels sont souvent très couteux et nécessitent un réinvestissement annuel, chose qui n’est pas à la portée de monsieur tout le monde.

Ceci étant posé, imaginons une application unique a vocation financière, dans cette application, il n’y aura pas de fonctionnalité qui ne soit relatif à la finance. Dans cette application, les utilisateurs pourront être en mesure de gérer les finances de l’entreprise, de faire une analyse financière, et aussi faire des prédictions. C’est cela le désir manifestée par les personnes qui travaillent dans la finance. En plus de cela nous en tant future ingénieur on IA, nous nous sommes dit qu’un petit chatbot qui permettrait de faire gagner du temps au usagers ne serai que bénéfique pour tout le monde.

Finalement, après tous ces observations nous avons conclus que développer ce genre d’application peut avoir un réel succès autant pour le concepteur que pour les utilisateurs. Cependant, avant de s’y aventurer il faut bien savoir les besoins fonctionnels du logiciel.

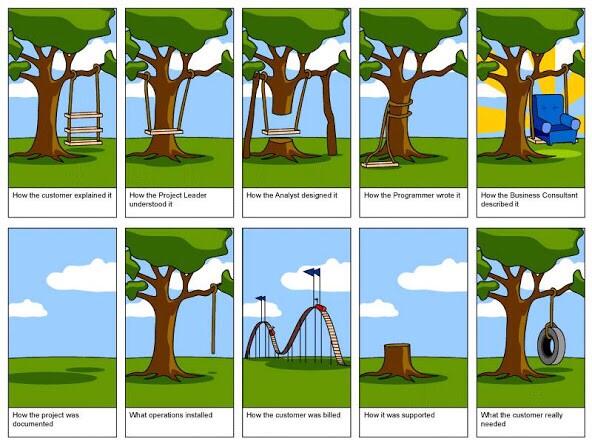
## L’expression des besoins

L’expression des besoins est une partie indéboulonnable dans la rédaction d’un cahier de charges. Dans sa forme la plus simple (ce que nous allons faire ici), l’expressions des besoins se fait en listant les éléments qui entrent dans sa composition. Nous avons deux types de besoins des fonctionnels et les non fonctionnels

* Les besoins fonctionnels
  + Création de nouveaux fichiers
  + Sauvegarde d’un fichier
  + Ouverture d’un fichier
  + Consulter le bilan
  + Consulter le compte de résultat
  + Consulter le tableau des flux de trésorerie
  + Modifier le bilan
  + Modifier le compte de résultat
  + Modifier le tableau des flux de trésorerie
  + Prédire le bilan
  + Prédire le compte de résultat
  + Prédire le tableau des flux de trésorerie
  + Faire une analyse financière complète
  + Faire une analyse prédictive financière complète
  + Avoir un chatbot
  + Faire toutes les actions citées avec le chatbot
  + Imprimer des documents
  + Rédiger un rapport d’analyse
* Les besoins non fonctionnels
  + Un temps réduit de traitement
  + Un prix abordable
  + Facile à utiliser
  + Sécurité des données
  + Assurer la maintenance
  + Déploiement seulement dans un réseau local

## Les fonctionnalités de l’application

Pour la plupart des projets informatiques, il y a déphasage entre les besoins du client et ce qu’a compris l’ingénieur, il peut même y avoir un décalage entre ce qu’a compris l’ingénieur et ce qui est vraiment réalisable comme l’illustre cette image.



C’est ce qui explique peut-être le taux de succès des projets informatiques mais cela, c’est autre débat. Bref ! il falloir tout être en communication avec le client pour ne pas faire du travail dans vent. Pour ce qui nous consterne bien que l’application finale qui aura pour vocation d’être vendu dans les entreprises doit répondre à tous les besoins cités, celle que nous allons développer ici que sera une version beta va répondre seulement aux besoins indispensables. De ce fait, les fonctionnalités que nous allons développer dans le logiciel pour ce travail de mémoire sont :

* Consulter le bilan
* Consulter le compte de résultat
* Consulter le tableau des flux de trésorerie
* Modifier le bilan
* Modifier le compte de résultat
* Modifier le tableau des flux de trésorerie
* Prédire le bilan
* Prédire le compte de résultat
* Prédire le tableau des flux de trésorerie
* Faire une analyse financière complète
* Faire une analyse prédictive financière complète
* Avoir un chatbot
* Faire toutes les actions citées avec le chatbot
* Rédiger un rapport d’analyse (limitée)

Les besoins non fonctionnels vont être plus pris au sérieux dans les versions qui vont entrer en production. Maintenant, nous savons exactement ce que nous voulons faire, il va falloir choisir les outils informatiques avec lesquelles nous allons travailler.

# Présentation des outils

Choisir un outil informatique pour implémenter un algorithme peut sembler être une chose d’évident mais nous allons voir qu’il y a des subtilités. Différents outils informatiques règles différents problème. Ce qui veut dire que ce n’est pas l’outil qui détermine le problème mais c’est le problème qui détermine l’outil. Le langage C est langage les plus rapide, mais ce n’est pas pour autant que l’on va l’utiliser pour faire du web, il y a des Framework pour cela. De ce fait comment choisis un langage ? Nous devons d’abord répondre à ces questions. Est-ce un langage utilisé pour nos problèmes posés ? Peut-il résoudre ces problèmes de manière optimiser ? Y a-t-il d’autres langage qui peuvent faire mieux le travail … Pour nos besoins cites préalablement, les langages de programmation et Framework qui répondent à ses questions sont les suivants.

## Les langages de programmation

Comme nous allons déjà dans ce document les langages de programmation sont les syntaxes qui traduisent les algorithmes d’une manière compréhensible à l’ordinateur. Apres recherche, nous avons desceller les deux meilleurs qui vont nous accompagner dans ce trajet. Que ce soit pour le développement des modèles mais aussi pour le développement une interface graphique.

### C++

Le C++ est un langage programmation crée en 1985 par l’informaticien danois Bjarne Stroustrup pour palier au manquement du langage C qui n’est orienté objet. Le C++ est le langage de programmation le plus utilisé par les développeurs, notamment en ce qui concerne les applications. Il permet d'aborder le développement sous plusieurs paradigmes : programmation générique, procédurale et orientée objet. C'est un langage compilé, ce qui signifie que le code source est traduit en code objet, ou binaire pour que la machine puisse l'exécuter. (C++ : présentation du langage de programmation, 2024)

Ce langage de programmation est un langage oriente objet ce qui veut il permet de créer des classes. C’est qu’il est si population, on peut donner l’exemple de Google qui l’utilisent pour son moteur de recherche, Microsoft qui l’utilisent World, Excel ou PowerPoint et aussi Autodesk qui l’utilise pour Maya. Pour ce qui est l’IA, avant l’avènement de Python, les ingénieurs l’utilisaient pour écrire les codes mais son impact est toujours. Car derrière presque tous les Framework de Python, qui nous dans l’AI, il y a le C++ ou C, le cas de Numpy, Pandas ou Matplotlib.

Les avantages de C++ :

* La performance, rapidité
* La popularité
* La portabilité en OS
* L’abondance de bibliothèques
* La programmation orienté objet

Les inconvénients :

* Syntaxe compliquée
* Langage pas du tout pour les débutants

### Python

Python est un langage de programmation créé par Guido Van Rossum. La première version publique du langage est sortie en 1991. Son nom provient de la troupe de comiques anglais les Monty Python.

Python est un langage de programmation dit de “très haut niveau”. Cela signifie qu’il possède un haut niveau d’abstraction par rapport au langage machine. Pour le dire très simplement : plus un langage de programmation est de “haut niveau”, plus sa syntaxe se rapproche de notre langage (l’anglais) plutôt que du langage machine. Un langage de haut niveau est donc plus facile à comprendre et à utiliser qu’un langage de plus bas niveau.

Certains langages (comme Python) utilisent un interpréteur comme traducteur tandis que d’autres utilisent un compilateur.

Un interpréteur se distingue d’un compilateur par le fait que, pour exécuter un programme, les opérations d’analyse et de traductions sont réalisées à chaque exécution du programme (par un interprète) plutôt qu’une fois pour toutes (par un compilateur). (Introduction à Python, 2024)

Les avantages de Python :

* Facile à utiliser
* Sécuriser
* Très populaire
* Compatibilité avec d’autres langages
* Possède beaucoup de bibliothèque pour le Machine Learning

Les limites de Python

* Temps d’interprétation très lent
* Mauvaise présentation des erreurs

### SQL

Structured Query Language (SQL) est un langage de gestion de données sous forme de base de données. Il est utilisé pour gérer des bases de données relationnelles avec ces quatre (4) actions principales que sont le CRUD (CREATE, RETREIVE, UPDATE ET DELETE).

De manière simple SQL va nous permettre de créer des bases de données en utilisant un système de gestion de base de données comme PostgreSQL, Oracle, Maria DB mais nous allons utiliser MySQL. Il va être créer une base données locale pour stocker les états financiers avec lesquels nous allons travailler.

## Les Framework

Un Framework est en ensemble de fonction prédéfinie dans un langage de programmation nous permettant de faire une action bien précise. Pour taches bien données, si nous avons un Framework, il n’est pas nécessaire de commencer de zéro puisque certaines fonctionnalités sont déjà implémentées. Les Framework ont été créer pour les taches complexes qui nécessite beaucoup de compétence, ainsi même les développeurs de niveau moyen peuvent créer des programmes avances, ce qui va servir à la productivité.

Il y a différents Framework pour diffèrent domaines informatique (développement web, mobile, logiciel…), mais nous allons présenter les Framework qui vont nous le développement de modèle intelligent et le développement d’interface graphique, ils sont tous liés soit à Python ou à C++.

### Scikt-learn

Scikit-learn est une bibliothèque de Python qui a commencé en 2007 avec le Google Summer of Code Project par David Carpaneau.



Ce Framework s’est spécialisé dans Machine Learning (supervisé et non supervisé) et nous donne des fonctions pratiques pour le développement de modelé. Scikit-learn supporte parfaitement des domaines comme la classification, la régression, le clustering …

Pour ce qui est de modèle, nous allons utiliser pour faire la prédiction des éléments des états financiers, la représentation de texte et bien d’autres.

### Spacy

Spacy est un Framework de python qui est utilisée dans le NLP, il nous permet tout simplement de raffiner du texte avant sa modélisation. La raison est que du texte écrit par un humains, avant de faire sa représentation en chiffre, va contenir du bruit. On entend par bruit tous les éléments, mots, vocabulaire qui ne nous sert pas. Exemple dans le langage français, il y a une conjugaison très lourde, un verbe comme « faire » peut donner fait, faite, furent etc. Spacy peut nous aider trouver la racine de ses mots. Il peut aussi nous aider à tout mettre minuscule, supprimer les ponctuation…



### Gensim

Gensim est un Framework Python avec lequel, il est possible de faire directement de la classification de texte. Dans son fonctionnement, ce Framework utilise les modèle Word2vec et aussi Fasttext. Ces derniers sont des techniques qui sont appelé Self Supervides Learning ou l’on utilise un texte pour générer les inputs et les outputs.



Avec ce Framework on calculer la similarité entre deux mots, voir faire des calculs sur des mots, oui des calculs des sur des mots.

### Tensorflow

Si Scikit-Learn est une bibliothèque de Machine Learning, Tensorflow en est une spécialisée sur le Deep Learning. Cette bibliothèque a été développe par Google dans le but de permettre aux experts mais aussi de débutant d’avoir un environnement pour travail dans le Machine Learning en générale.



De tous les Framework que nous avons présentée, Tensorflow est très probablement le plus puissant car nous permettant de faire ce que tous font. C’est un outil tout en un avec force et ses faiblesses.

En plus de tout, il nous permet de faire du Computer Vision avec les CNN, du NLP avancée avec la RNN, les LSTM, de créer des API pour le déploiement et bien d’autres.

### Pandas

Pandas est un Framework Python très pratique dans le développement de modèle. Il est utilisé dans le travail a posteriori, c’est Feature Engineering. Avec pandas, nous pouvons importer des fichiers CSV, vérifier les données manquantes, les outliers…



Avant chaque développement de modèle, Pandas va certainement intervenir, ce Framework supporte les statistiques qui pourront nous permettre de mettre les données dans formats acceptable par l’ordinateur.

### Numpy

Nativement, les structure de type tableau n’existe pas en Python, il y a des listes en Python pour le remplacer. La différence entre ces deux c’est que les tableaux acceptent un seul type de données et taille de varie pas or, les listes acceptent plusieurs types et sa taille peux varier.

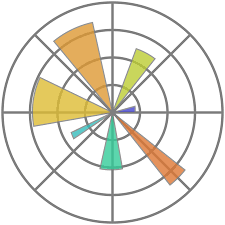


C’est là qu’intervient Numpy pour permettre d’utiliser des tableaux des Python, qui bien plus rapide à exécuter. En plus de cela, Numpy a un excellent support de l’algèbre linéaire, les matrices, les vecteurs et autres domaines mathématiques.

Il est nécessaire d’ajouter que Numpy, bien qu’utilisé en Python est écrit en langage C qui plus puissant et plus rapide que le Python.

### Matplotlib

MatplotLib est Framework de visualisation avec Python, il sert tracer des courbes en utilisant Numpy ou Pandas. La visualisation peut avant et après le modèle, soit pour les comprendre les données brutes, soit pour vérifier les résultats.



La visualisation est en train devenir une science à part entière, donc Matplotlib est utilisé dans des domaines autres le Machine Learning, notamment les développent d’interface graphique que nous allons voir.

### Qt

Qt est bibliothèque de C++ cross plateforme lance en 1995, et qui est complètement gratuit. Qt nous permet de créer des interfaces très avancé et dans un IDE et simple à comprendre et à utiliser. Avec cette bibliothèque, il n’est pas nécessaire de savoir coder pour créer des interfaces graphiques car il y a la possibilité de créer des widgets avec du glisser-déposer.



Pour ce qui nous concerne nous allons bien évidemment l’utiliser pour l’interface qui va accueillir les clients. Mais un logiciel mais pas du web.

Puisque cette application n’a pour vocation d’être déployer dans le cloud, le web n’est pas nécessaire. De plus, développer l’interface graphique de cette manière nous donne une certaine sécurité car il n’y aura pas de brèche que des personnes extérieures à l’organisation peuvent utiliser pour accéder aux données sensibles.

# La collecte de données

S’il y a bien une chose sur laquelle nous avons insisté dans ce document, c’est que l’IA travaille sur des données, beaucoup de données : pour apprendre il lui le matériel nécessaire. Les données sont pour l’IA ce que les livres, articles, les enseignements sont pour nous humains.

Cependant, après avoir compris l’importance des données, il faut comprendre qu’il n’est chose aisée de les recueillir ses données là. Très souvent c’est la partie la plus chronophage, et des fois demande beaucoup ressources humaines, financières et matériels.

Pour nos modèles, nous avons la chance que les données soit déjà disponible gratuitement. La raison est que nous allons travailler avec les données financières des entreprises, et pour les entreprises cotées en bourse c’est une obligation pour eux de les publier par souci de transparence. C’est pour cela que nous pouvons nous server sur des données des entreprises sans risque. De l’autre part, nous aurons en outre besoin de données texte pour notre chatbot, et là, c’est encore plus facile car du texte, sous n’importe quelle forme, c’est ne pas ce qui manque sur internet. Nous allons amener à voir comment il a été fait la collecte de nos données.

## Les données d’états financiers

Le travail que nous faisons va porter des entreprises cotées à la Bourse Régionale des Valeurs des Mobilières (BRVM). La BRVM est le marché financier de l’Union Economique et Monétaire Ouest Africain (UEMOA). C’est ici que l’on peut essayer des actions et obligation pour le compte des entreprises et Etats de la zone UEMOA. La BRVM regroupe les 46 entreprises et banque les performantes de la zone.

Eu égard de cela, tout entreprise cotée à la BRVM, a le devoir chaque annexe de publiée ces états financiers pour le compte des actionnaires, des Etats, des investisseurs, de potentiels investisseurs ou n’importe quelles personnes physiques ou morales. Leurs états financiers sont aussi publiés dans le [site de la BRVM](https://www.brvm.org/) ainsi, il est possible de leur télécharger et de faire notre travail, on choisit une entreprise et c’est bon.

Une fois téléchargés, les états financiers d’une seule année se présente comme suit :

* Le rapport d’activité
* Le résultat financier
* La synthèse des rapports de gestion

Mais ses données ne sont pour le moment exploitables, il va faire un certain nombre de transformation c’est-à-dire extraire les données qui nous intéresse, les mettre de fichier CSV, JSON ou autres avant de pouvoir passer au Feature Engineering. Avec des états on faire une analyse financière, vérifier la rentabilité de l’entreprise, prédire des valeurs etc…

## Les données texte

De tous les type de données que l’on va utiliser pour des modèles intelligents, bien qu’étant les plus difficile à exploiter, les données textes sont les plus facile à collecter. Nous aurons besoins des données texte pour le développement du chatbot. Notre principal objectif ici c’est de comprendre les questions que l’utilisateur du chatbot peut demander. Bien sûr, le chatbot que nous allons développer va être spécialisé dans nos états financiers, donc nous aurons besoin de texte qui traite de la finance des entreprises. Il y a plusieurs manières de collecter ce genre de texte :

* Les sites qui de finance
* Les sites d’information
* Les base de données
* Les sites de génération de textes
* Le Web Scraping (des programmes qui vont aller extraire des informations automatiquement sur les sites web)

Nous avons utilisé un condensé de tout cela pour avoir les données texte dont nous aurons besoin. Il y aura d’abord les texte qui vont nous permettre de faire le Intent classification, ensuite il va utiliser des techniques pour reconnaitre les entités d’une question posée, pour pouvoir répondre avec la plus grande des précisions. Si vous n’avez pas compris cette dernière phrase, ne vous inquiétez pas car nous allons revenir en détail sur tout ce la partie ou l’on parte du développement du chatbot.

# Prédiction des valeurs

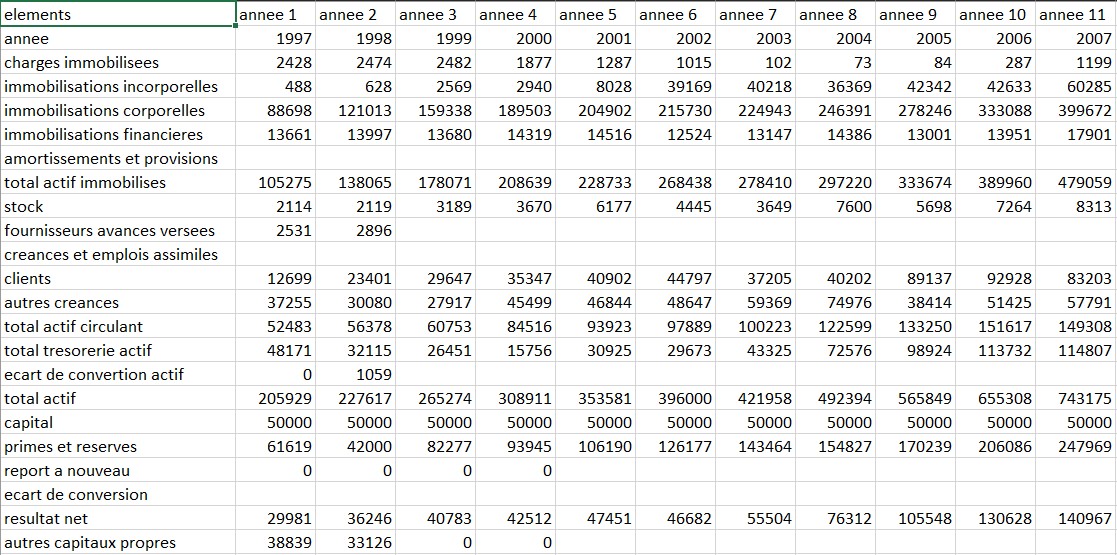
La prédiction est un vaste domaine qui ne relève seulement de l’IA. En effet beaucoup de domaine scientifique essaie de faire des prédictions sur les valeurs avec lesquelles ils travaillent, c’est notamment le cas de l’économie, la matéologie, la bourse etc.

Les intérêts de faire des prédictions dépendent du domaine ou elles faite, pour ce qui nous concerne qui est le domaine de la finance d’entreprise, cela nous permet d’approximer la future santé financière de l’entreprise en se basant sur les actuelles et passées.

Dans le chapitre passé, nous avons montré les techniques intelligentes qui permettent de prédire des valeurs ici, allons comment cela fonction en pratique.

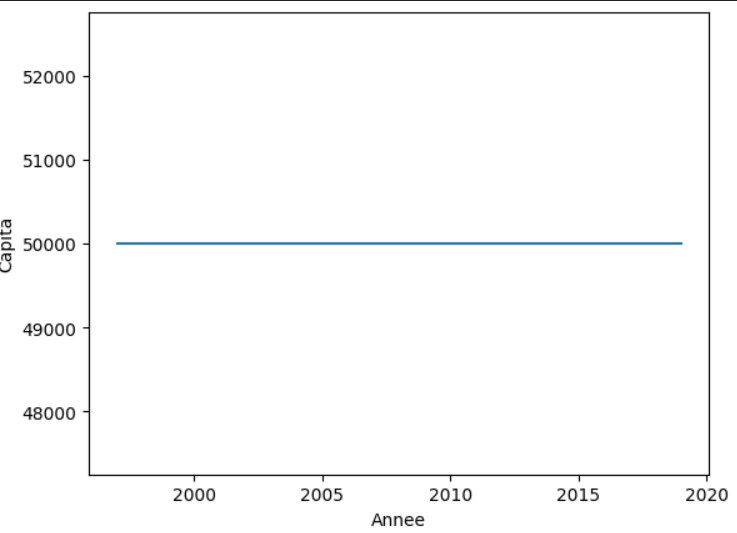
D’abord il nous faut des données, elles sont collectées, et nous avons vu comment la précédente partie. Nous faire la prédiction de chaque élément de chaque état financier. Ce qui nous fait une centaine de prédiction à faire.

Ce que nous allons faire par la suite c’est mettre toutes les valeurs dans un fichier CSV, il faut rappeler que les documents téléchargés dans le site de la BRVM, sont sous format PDF et donc non exploitables. Une fois sous le format CSV, nous aurons trois (3) fichiers à savoir les bilans, les comptes de résultats et les tableaux des flux de trésorerie.



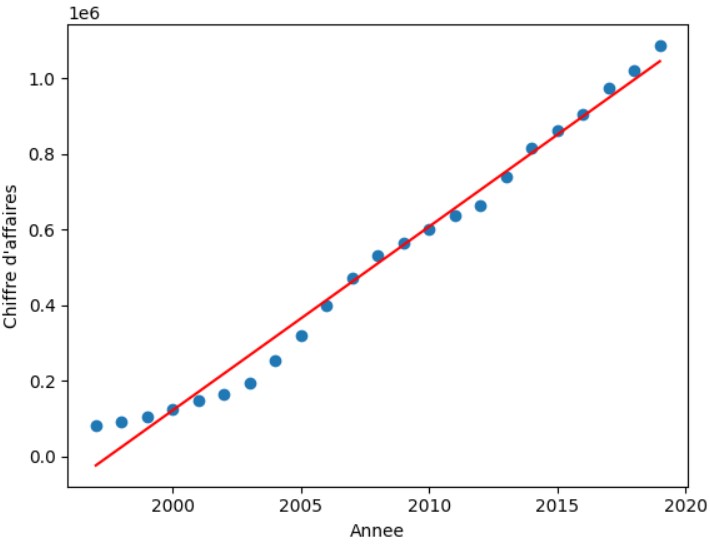
Du fait que nous avons plusieurs prédictions à faire, nous allons seulement en présenter quatre (4) du plus simple au plus intéressant.

* Le capital



Pour ce qui du capital de cette société, nous il n’a pas évolué de 1997 à 2019, donc ici il n’y a pas de prédiction à faire, puisse que nous savons que le capital de cette de société de varie pas, sur toutes cette période il reste a 50 000 000 000 de F CFA. Pour les valeurs futures et pour les calculs futurs qui vont faire intervenir le capital nous allons choisir cette même valeur.

* Le chiffre d’affaires

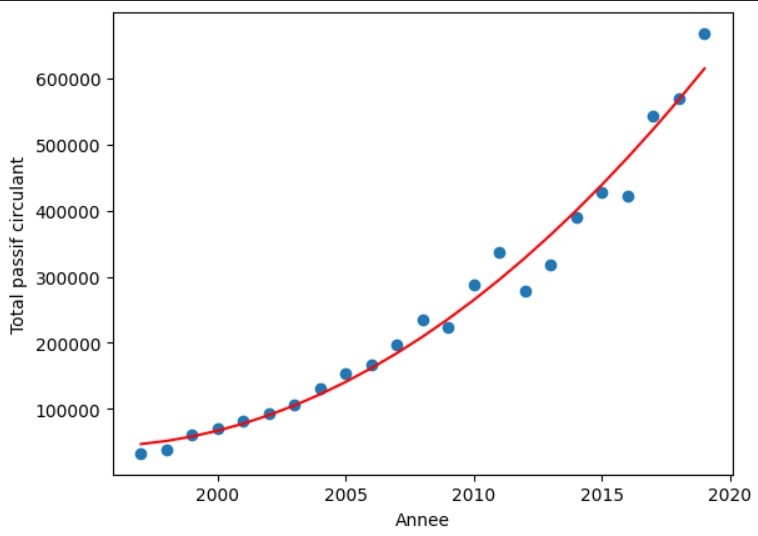


Il y a deux choses à voir ici, les points en bleu qui représentent les valeurs réelles du chiffre d’affaires en fonction des années et la droite linéaire en rouge qui est droite de régression. La régression linéaire a bien fonctionné ici puis que nous qu’elle épouse à la presque perfection les données. Cette droite va renvoyer le coefficient et la constante qui représentent nos w0 et w1 que nous avons déjà expliquer pour faire des prédictions pour les années à venir.

Le tableau ci-dessous nous donne pour la droite de régression du chiffre d’affaires son coefficient, sa constante et la prédiction de quelques valeurs futures.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Coefficient | Constante | 2025 | 2030 | 2035 |
| 48578.23616 | -97034585.22134 | 1336343.0148221 | 1579234.1956521 | 1822125.376482 |

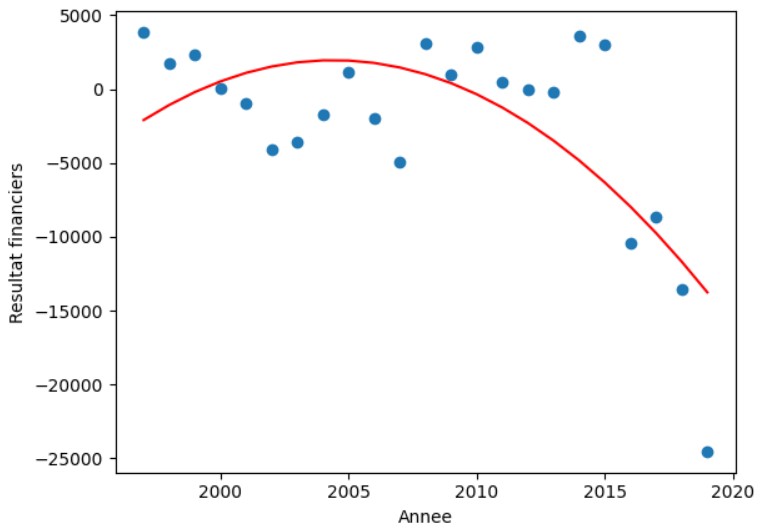
* Le passif circulant



Ce cas est un tantinet plus intéressant que le précédant les données ne suivent pas une forme linéaire, alors il nous faut autre chose. Cette autre chose c’est évidemment la régression polynomiale que nous avons aussi déjà vue. Pour ce cas de figure nous n’aurons seulement w0 et w1 mais aussi w2, puisse que le degré de notre polynôme est égal à 2.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| W0 | W1 | W2 | W3 | Année A |
| 0 | -4.00030285e+06 | 1.00251237e+03 | 3.99062354e+09 | W0+W1\*A + W2A2+W3 |

* Le résultat financier



Le modèle des résultats financier est l’un des plus intéressant pour trois (3) raisons : d’abord les valeurs sont très dispersées par rapport aux autres, ensuite les valeurs décroisent de manière exponentielle et enfin il y a des valeurs négatives. C’est normal qu’il y a des valeurs négatives pour le résultat financier dans la mesure ou charges financières sont supérieures aux produits financiers. Nous pourrions être tentée de penser que ce modèle va être plus compliquée que les autres, mais il n’en est rien. Nous allons simple le faire passer dans une régression polynomiale et la magie opérée. Le résultat pour ce modèle est la suivante.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| W0 | W1 | W2 | W3 | Année A |
| 0 | 2.95998776e+05 | -7.3836900e+01 | -2.9664942e+08 | W0+W1\*A + W2A2+W3 |

Il faut notre que pour tous les modèles, nous allons suivre exactement les mêmes procédures que ces quatre, tous autres vont se classer dans l’un de ces cas de figure.

Maintenant que nous avons prédit nos modèles, il est possible de les déployer pour utiliser dans une interface graphique. Pour ce modèle, il le déploiement peut être simple du fait qu’il y a seulement des nombres. On peut les stocker dans un fichier et appeler depuis le frontend.

# Le développement du Chatbot

# L’interface graphique

A ce moment de notre travail, tout a déjà été fait, ou presque. Oui les prédictions ont été faites, oui le chatbot a été développe mais il subsiste un grand problème : à moins être un ingénieur en IA, vous ne pouvez pas interagir ou utiliser les modèles. C’est ce problème que nous allons essayer de résoudre ici développant une interface graphique que n’importe quel utilisateur peut utiliser pour interagir avec les modelés. Pour ce faire il y aura deux phases faire de Graphical User Interface (GUI) et lier les modèles avec le GUI à travers un web service.

## Développement de l’interface graphique

L’interface graphique va être développe la bibliothèque Qt/C++, qui un Framework qui nous facile cette tache en utilisant le langage C++. Nous avons utilisé cette bibliothèque car l’application n’a pas pour vocation d’être déployer sur internet, c’est une organisation qui va utiliser une instance de celle-ci.

### Le frontend

Qt nous donne un certain nombre de fonctionnalités pour réaliser des logiciels robustes et très avancées mais allons prendre quelques-uns, qui nous ont été utile dans notre travail, pour les expliquer et faire comprendre comment nous avons utilisé dans nos modèles.

* Widget

Les Widgets font partie des éléments de base sur Qt, au tout est un widget et il y a des Widgets pour faire. Pour mieux comprendre ce concept, un Widget est élément visible que l’utilisateur peut interagir avec. Nous pouvons donner l’exemple d’un bouton, une zone de texte, un label ou voir même une page entière.

L’un des Widget les plus ce que Qt appelle QMainWidget qui est le Widget principale sur lequel tout des autres vont s’accoler. Il faut préciser chaque Widget a un parent sur le Widget principale. Ce qui nous donner une structure sous forme d’arbre.

Dans notre logiciel nous allons utiliser toute sorte de Widget commençant bien évidemment par le QMainWidget en passant par les autres qui permettent de présenter le résultat de nos modèles d’une manière simple.

* Layout

Le deuxième élément sur lequel nous allons un peu nous attarder ce sont les Layout. Ils représentent tous les manières de disposer nos Widgets. Par un Widget va empiler tous ses enfants sur son point (0, 0) en haut en gauche dans un ordinateur.

Avec l’aide des Widget nous pouvons avoir plusieurs de présenter les Widgets, ce aura pourra pour effet d’avoir un logiciel plus sérieux. Il y trois sortes de Widget sur Qt : QVBoxLayout (disposition verticale), QHBoxLayout (disposition horizontale) et QGirdLayout (disposition en grille). Ses trois héritent tous de la classe mère QLayout.

Bien que nous avons plus utiliser les dispositions verticales et horizontales dans notre logiciels, la disposition en grille peut être très pertinent par moment. C’est impressionnant de se dire quand même que tous les frontends que nous voyons dans les logiciels ne sortent pas de trois Layout.

* Signal and Slot

Les Signal and Slot sont les deux concepts de la programmation événementielle que Qt mets à notre disposition. Un Signal c’est un signal c’est-à-dire un déclencheur, cela peut être une clique, un focus, un survol un changement quelconque. Il y a deux Signal prédéfinis comme ceux que nous venons de citer, mais nous avons la possibilité d’en créer des nouveaux. Néanmoins, les prédéfinis vont nous suffire dans la plupart des cas.

Les Slot c’est les définitions que nous allons utiliser pour réagir par rapport au Slot. Mais pourquoi ce ne sont des méthodes comme les autres ? C’est que les Signal n’acceptent que les Slot comme fonction, une méthode qui ne portent pas la mention Slot ne va pas être acceptée dans la programmation évènementielle de Qt.

Nous illustrer nous cela en l’exemple de l’utilisateur qui veut ouvrir les paramètres. S’il clique sur le bouton paramètre (Signal), l’application va ouvrir la boite de dialogue paramètre (Slot). Ce qui intéressant avec Qt, c’est qu’il nous permet de faire cela en une seule ligne de code avec la fonction « connect » de la classe QObject.

### Le Web server

Une web server est un programme informatique qui nous permet une encapsulation de données avec les protocoles http ou https. Avec un web server on déployer une fonction qui peut être invoquer depuis plusieurs Endpoint.

De là on voir comment ce peut nous être utile, nous avons déjà développer nos modèles intelligents et ce qui suit c’est de les liés avec l’interface graphique. Les modèles ont été développés en langage Python et l’interface graphique avec C++, il nous faut les lier. C’est là qu’intervient la puissance des web server. Il est possible écrire un Application Programmation Interface (API) dans un langage A et de l’invoque dans une langage B avec l’aide d’un navigateur.

Il y a deux types de web server : étendu avec SOAP et REST, nous allons utiliser REST avec la bibliothèque de Python FastAPI.

Nous pouvons donner l’exemple d’un cas d’utilisateur du chabot. L’utilisateur pose sa question depuis l’application Qt, une requête REST va être invoquer avec la question comme paramètre, le web server va le prendre le traiter et renvoyer la réponse sous format JSON ou XML.

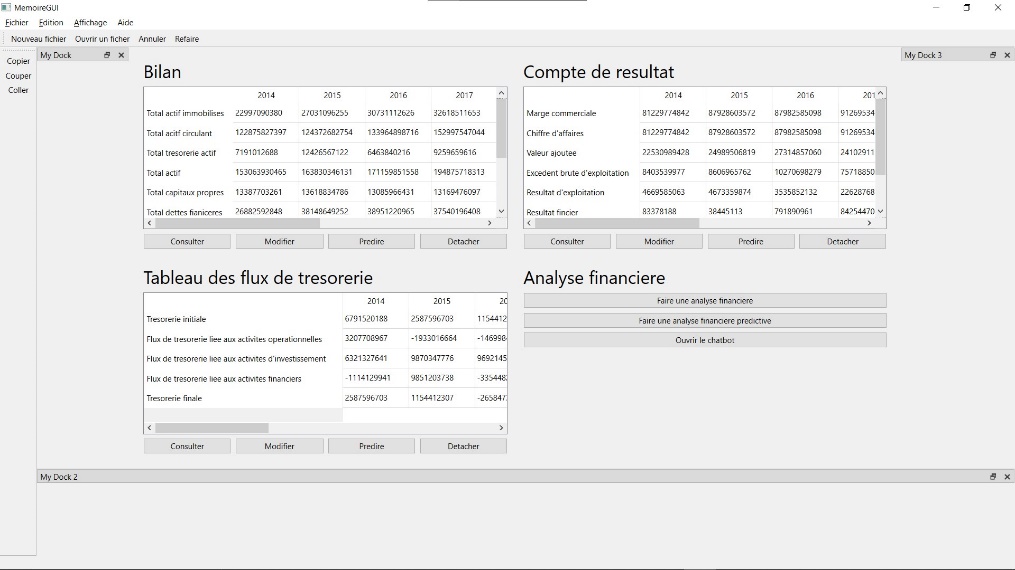
Cette méthode est aussi appelée le développement multi-tiers il y a deux programme différents (deux tiers) mais ils peuvent communiquer tout en étant indépendant l’un de l’autre. Ci-après une image qui illustre l’harchitecture d’une du processus global.

(Image illustrant l’architecture)

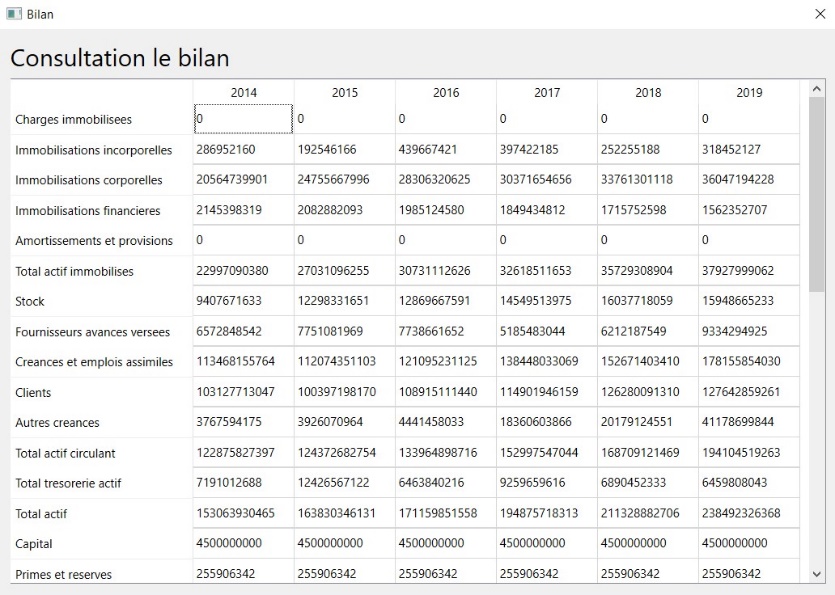
## Présentation de l’application

Toute l’explication a été faites donc ici nous allons juste montre à quoi ressemble l’application, l’interface graphique qui l’utilisateur. Toutes les pages ne seront pas montrées ici mais seulement les plus pertinent, celle l’utilisateur va plus ouvrir.

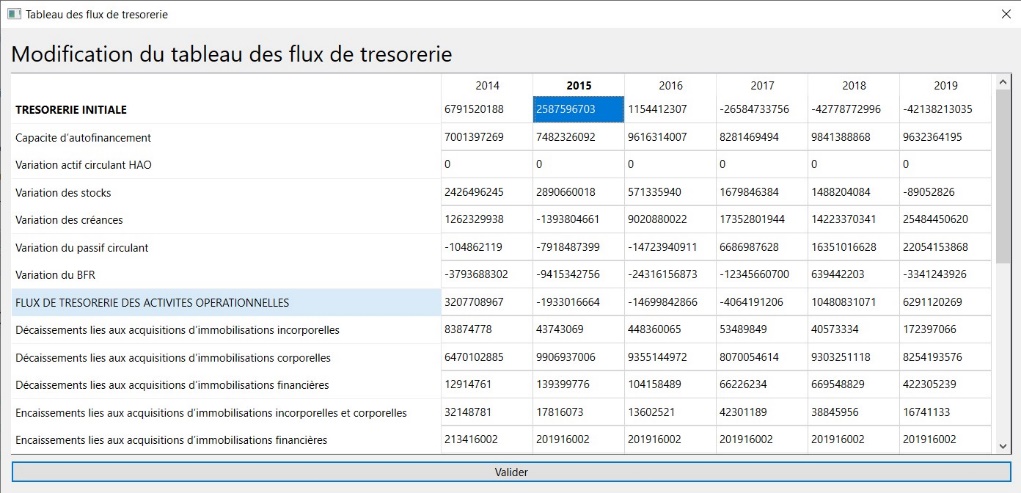
* La page d’accueil



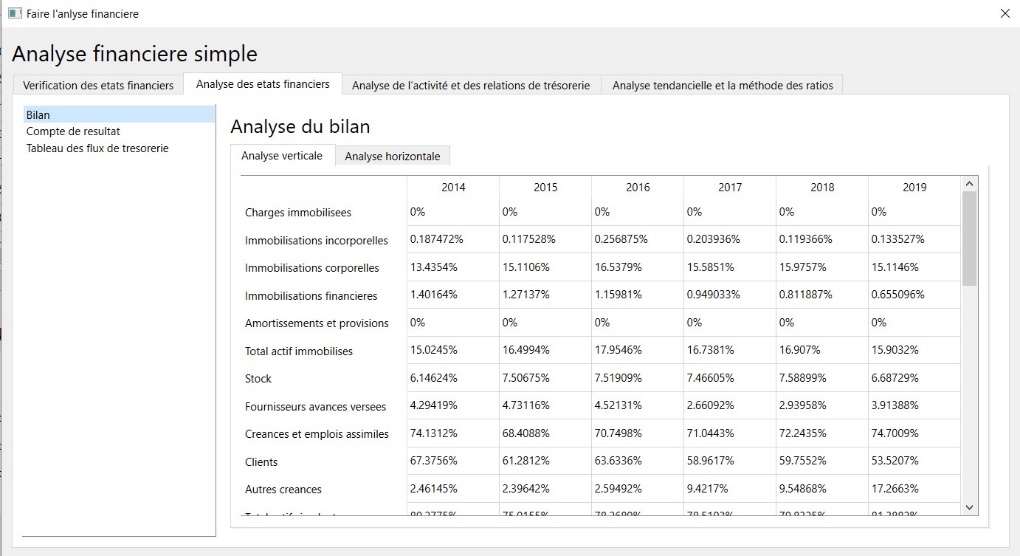
* La page de consultation du bilan



* La page de prédiction du compte de résultat sur les 3 prochaines années
* La page de modification du tableau des flux de trésorerie



* La partie analyse des états financiers



* La partie analyse prédictive de la rentabilité sur les 3 prochaines années
* Le chatbot

# Conclusion partielle

Ce qui a été fait

Les chose à retenir

Perspective

Intérêt

# Webographie

*C++ : présentation du langage de programmation*. (2024, 5 20). Retrieved from Journal du net: https://www.journaldunet.fr/web-tech/dictionnaire-du-webmastering/1445316-c-definition-et-presentation-de-ce-langage-de-programmation/

*Introduction à Python*. (2024, 5 20). Récupéré sur pierre-giraud: https://www.pierre-giraud.com/python-apprendre-programmer-cours/introduction/