

MÉMOIRE DE PROJET DE FIN D' ANNÉE

Présenté à

L'ÉCOLE NATIONALE DES SCIENCES APPLIQUÉES AL HOCEIMA

Filière: Génie Informatique

Titre du PFA :

**Réalisation d'un Robot pour l'automatisation des échanges de crypto-
monnaies Virtual**

Réalisé par :

- BABACHEKH Mohamed
- ELMAHALLI Zakaria
- EL MAGHRAOUI Khalil

Encadré par :

- Pr. BAHRI Abdelkhalak

Soutenu le 03/07/2021 devant le Jury :

- Pr. BAHRI Abdelkhalak
- Pr. BOUJRAF Ahmed

Avant-propos



Intitulé du Projet	Réalisation d'un Robot pour l'automatisation des échanges de crypto-monnaies Virtual
Réalisation de la part des élèves ingénieurs de l'ENSAH	BABACHEKH Mohamed ELMAHALLI Zakaria EL MAGHRAOUI Khalil
Encadrant	Le professeur BAHRI Abdelkhalak
Membres de Jury	Le professeur :BAHRI Abdelkhalak Le professeur :BOUJRAF AHMED
Délai	Du 28/04/2021 Au 30/06/2021

SOMMAIRE



Avant-propos	2
SOMMAIRE.....	3
DIDICACES	6
Remerciements.....	7
Résumé	8
Abstract.....	9
Liste des Figueres.....	10
Introduction Générale	11
Contexte de projet.....	12
1.1 Présentation de Projet	13
1.1.1Présentation de l'école nationale des sciences appliquées d'Al Hoceima (ENSAH)	13
1.1.2Notions de Base.....	13
1.1.2 1 Cryptomonnaie:	13
1.1.2 2 Qu'est-ce que le trading sur cryptomonnaies ?.....	14
1.1.2 3 Achat et vente de cryptomonnaies via une plateforme d'échange?	14
1.1.2 4 Comment fonctionne le marché des cryptomonnaies ?	15
1.1.2 5 Les caractéristiques des cryptomonnaies ?	15
1.1.3Problématique.....	16
1.1.4Contexte du projet	16
1.2 Conduite et gestion de projet :	16
1.2.1Contraintes techniques :.....	17
1.3 Conclusion :	18

1.1.	Introduction.....	20
1.2.	Déroulement du projet.....	20
1.3.	Structuration du projet.....	20
1.3.1.	Analyse fonctionnelle et définition des objectifs	20
1.3.2.	WBS (Work Breakdown Structure)	20
1.4.	Identification des risques :	21
1.5.	Le Cycle De Vie Du Projet	21
1.5.1.	La Collecte De Données	22
1.5.2.	Compréhension Des Données	22
1.5.3.	Transformation des données et traitement.....	22
1.5.4.	Planification	23
1.5.4.1.	Développement de diagramme de PERT :.....	23
1.5.4.2.	Le diagramme de PERT:.....	23
	24
1.5.5.	Développement de diagramme de GANTT :	24
	24
1.6.	Contrôle :	24
1.6.1.	Gestion de modifications :	24
1.6.2.	Gestion des risques :	25
1.7.	Cahier de charge :	25
1.8.	Introduction.....	27
1.9.	Diagramme de classes	27
1.10.	Diagramme de séquence.....	29
1.11.	Diagramme d'activité	29
1.12.	Conclusion	31
5.1	Choix logiciels	41

5.1.1 Langage Python	41
5.1.2 Numpy	42
5.1.3 Scikit-learn	43
5.1.4 google colab.....	43
4.1.5 Postman	44
5.1.6 pycharm	44
5.1.7 vs Code.....	44
5.1.8 Pandas	45
5.1.9 Matplotlib	45
5.1.10 Pickle.....	46
5.1.11 Sqlit	46
5.1.12 Git.....	46
5.1.13 Azur_computing.....	47
5.2 Application :	48
5.2.1 afficher les données de marché en direct	48
5.2.2 Afficher des messages à l'utilisateur (logging).....	49
5.2.3 Mettre en place une stratégie	49
5.2.4 Suivre les ordres /TRADES (PNL ,statues des Ordre).....	50
6 Conclusion	51
7 Conclusion générale	52
Bibliographie	53

DIDICACES



A nos chers parents,

On dédie ce modeste travail et nos profondes gratitude à nos chers parents pour l'éducation qu'ils nous ont prodigué, avec tous les moyens et au prix de tous les sacrifices qu'ils ont consentis à notre égard, pour le sens du devoir qu'ils nous ont enseigné depuis notre enfance.

A nos frères et sœurs,

Nous tenons à exprimer nos sentiments de satisfaction et d'amour pour vos encouragements permanents, vos soutiens moraux, et pour vos appuis.

A nos ami(e)s,

En souvenir de notre sincère et profonde d'amitié et des moments agréables que nous avons passés ensemble. Veuillez trouver dans ce travail l'expression de mon respect le plus profond et notre affection la plus sincère.

A nos familles,

A toute nos familles pour leur soutien tout au long.

A notre Ecole,

Nous dédions ce modeste Travail qui consiste à la réalisation d'une application de gestion des application aériennes, pour nos chers professeurs, et au cadre administratif de l'école

Remerciements



Nous aimerions de profiter de cette occasion, pour exprimer notre gratitude et remerciement à toute personne a contribué au succès de ce modeste travail pour leurs conseils et leurs confiances.

Un grand remerciement à Monsieur **BAHRI Abdelkhalek** notre encadrant, pour sa grande disponibilité, son soutien constant et l'amour du travail bien soigné qu'il nous a d'avantage inculqué.

Nous tenons à remercier également les membres de jury pour leurs efforts, afin de nous offrir des recommandations fructueuses pour une bonne orientation de prochains projets et une évaluation de notre projet.

Nous tenons également à présenter une expression de satisfaction au Génie Informatique 2 qui a fait preuve d'un esprit d'équipe admirable. Enfin, Nous tenons à remercier toute personne qui a laissé une empreinte merveilleuse lors de réalisation de ce projet et rédaction de ce rapport

Résumé



L'objectif de ce travail est le développement d'un robot pour l'automatisation des échanges de crypto-monnaies Virtual, permettant d'offrir un ensemble de fonctionnalités spécifiques aux compagnies du crypto au bien au les gens qui s'intéresse au trading et le crypto-monnaies.

Ce rapport contient des éléments qui présentent le déroulement de réalisation de notre projet, tout d'abord nous présenterons le cahier des charges, dans lequel nous analyserons les besoins de notre projet et d'offrir une description complète de projet. Après la validation de ce document, la conception c'est l'étape qui suit l'analyse, dans ce cadre nous tenons à réaliser des diagrammes à l'aide de langage UML, pour mettre en œuvre notre projet on se basera sur un langage de programmation orientée PYTHON, afin d'implémenter des diverses tâches de codage.

En outre, dans la dernière partie de ce rapport, nous avons présenté les fenêtres composées de notre projet, ainsi que l'explication de chacune. En effet, ce registre décrit le déroulement de l'exécution de notre projet qui s'inscrit dans le cadre de la réalisation de projet fin d'année (PFA) de la filière Génie Informatique au sein de l'Ecole Nationale Des Sciences Appliquées d'Al Hoceima (ENSAH).

Abstract



The objective of this work is the development of a robot for the automation of crypto-currency trading Virtual, allowing to offer a set of specific features to crypto companies also to people who are interested in trading and crypto-currencies.

This report contains elements that present the process of realization of our project, first we will present the specifications, in which we will analyze the needs of our project and offer a complete project description. After the validation of this document, the design is the step that follows the analysis, in this framework we want to make diagrams using UML, to implement our project we will be based on a programming language-oriented PYTHON, in order to implement various coding tasks.

In addition, in the last part of this report, we have presented the compound windows of our project, as well as the explanation of each one. Indeed, this register describes the course of the execution of our project which falls within the framework of the realization of end-of-year project (PFA) of the Computer Engineering Branch within the National School of Applied Sciences of Al Hoceima (ENSAH).

Liste des Figueres



Figure 1 : l'école nationale des sciences appliquées al Hoceima	13
Figure 2 work breakdown structure	21
Figure 21.5.3. Transformation des données et traitement.....	22
Figure 3 Diagramme de PERT.....	24
Figure 4 Diagramme de GANTT	24
Figure 5 Diagramme de classe	28
Figure 6 Diagramme de séquence	29
Figure 5 :python.....	42
Figure 6 : Scikit-learn	43
Figure 7 : google colab.....	43
Figure 8 Postman	44
Figure 9 pycharm	44
Figure 10 vs Code.....	45
Figure 11 Sqlit	46
Figure 12 Git.....	47
Figure 13 azur	47
Figure 14 interface d'utilisateur de l'application.....	48
Figure 15 afficher les données de marché en direct	49
Figure 16 Afficher des messages à l'utilisateur (logging)	49
Figure 17 Mettre en place une stratégie	50
Figure 18 Suivre les ordres /TRADES (PNL ,statues des Ordre)	50

Introduction Générale



Dans le programme pédagogique de l'école Nationale de Sciences Appliquées d'Al Hoceima, les étudiants sont tenus compte de préparer leur projet de fin d'année pour leur deuxième année de cycle d'ingénierie. En fait, On a eu la chance de travailler sur un sujet primordial dans le domaine informatique surtout les robots d'automatisation(en temps réel) il s'agit bien de «Réalisation d'un Robot pour l'automatisation des échanges de crypto-monnaies Virtual».

En effet ,L'objectif de ce projet est de réaliser un robot open source qui négocie la crypto-monnaie sur une bourse, telle que binance, bitmex, en utilisant des stratégies pour maximiser les profits tout en minimisant les risques tout se basent à des models de machine Learning pour automatiser les trades .

Par ailleurs, ce rapport va vous présenter les différentes phases adoptées afin d'aboutir à la réalisation du produit fini qui se présente dans l'interface graphique réalisée sur PYTHON (voir chapitre 3). En commençant par la phase présente le contexte général du projet apres ,la détermination des besoins là où on va cerner les différentes exigences de ce robot et après on va passer directement à la phase de la conception. En effet, on va décrire conceptuellement – tout en adoptant le langage de modélisation- pour la création de certains diagrammes. Ensuite, il vient l'étape qui prend une partie importante du planning global du projet, il s'agit bien, de la phase de développement (réalisation) là où on va essayer le maximum possible de répondre aux exigences déjà mentionnées. Après avoir développé toutes les fonctions, il faut bien s'assurer que celles-ci fonctionnent correctement c'est pour cela vient la phase de tests d'intégrations et une fois que tout est validé on intègre l'application toute entière. De plus, ce rapport met la lumière sur les outils qui nous ont facilité le développement et la maintenance de notre Bot.

CHAPITRE 1

Contexte de projet

Ce chapitre présente le contexte général du projet. Nous allons commencer tout d'abord par une présentation du projet où nous décrirons les objectifs ciblés ainsi que le planning de travail adopté pour ce projet.

1.1 Présentation de Projet

1.1.1 Présentation de l'école nationale des sciences appliquées d'Al Hoceima (ENSAH)

L'Ecole Nationale des Sciences Appliquées d'Al Hoceima (ENSAH) est un établissement public d'enseignement supérieur relevant de l'université Abdelmalek Essai. Créée en 2008, Sa création s'inscrit dans l'optique de favoriser la formation des ingénieurs d'Etat hautement qualifiés dans les spécialités les plus ouvertes et susceptibles de connaître d'importants développements au sein du tissu socio-économique régional et national.



Figure 1 : l'école nationale des sciences appliquées al Hoceima

L'Ecole Nationale des Sciences Appliquées Al-Hoceima dispense des formations de deux années préparatoires (AP1, AP2) et trois années de spécialités dans les domaines suivants :

- Génie Informatique
- Génie Civil
- Génie Energétique et énergies renouvelables
- Génie de l'eau et Environnement

1.1.2 Notions de Base

1.1.2.1 Cryptomonnaie:

Une cryptomonnaie, dite aussi cryptoactif, crypto devise, monnaie cryptographique ou encore cybermonnaie, est une monnaie émise de pair à pair, sans nécessité de banque centrale, utilisable au moyen d'un réseau informatique décentralisé. Elle utilise des technologies de cryptographie et associe l'utilisateur aux processus d'émission et de règlement des transactions.

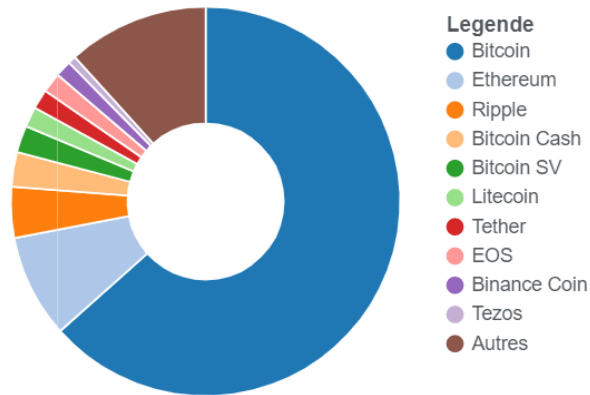


Figure 2 :Cryptomonnaies les plus utilisées en 2020.

Une cryptomonnaie est une devise numérique décentralisée, qui utilise des algorithmes cryptographiques et un protocole nommé blockchain pour assurer la fiabilité et la traçabilité des transactions. Les cryptomonnaies sont entièrement virtuelles, elles peuvent être stockées dans un portefeuille numérique protégé par un code secret appartenant à son propriétaire. Des plateformes d'échanges (Binance, Bitmex, Coinbase, Bitstamp, etc.) servent à acheter et revendre de la cryptomonnaie en ligne. Si vous souhaitez en savoir plus et comprendre la procédure pour l'achat des cryptomonnaies, vous pouvez vous rendre sur le site de Cryptoast.

La première cryptomonnaie à avoir vu le jour, et sans doute la plus célèbre d'entre elles, est le Bitcoin. Créée en 2009 par un énigmatique programmeur utilisant le pseudonyme Satoshi Nakamoto, elle a propulsé le principe de blockchain et a entraîné la création de nombreuses autres devises numériques cryptées, nommées « altcoins » par les cryptotraders.

1.1.2 2 Qu'est-ce que le trading sur cryptomonnaies ?

Le trading sur cryptomonnaies consiste à négocier sur les mouvements de cours d'une cryptomonnaie au moyen d'un compte de trading sur CFD ou à acheter et à vendre la monnaie sous-jacente via une plateforme d'échange.

1.1.2 3 Achat et vente de cryptomonnaies via une plateforme d'échange?

Lorsque vous acquérez des cryptomonnaies via une plateforme d'échange, vous achetez la monnaie elle-même. Pour négocier les monnaies numériques de cette manière, vous devrez d'abord ouvrir un compte, puis immobiliser l'intégralité de la position et enfin

conserver vos jetons sur votre portefeuille de cryptomonnaies (wallet) jusqu'à ce que vous décidiez de les revendre.

Les plateformes d'échange ne fonctionnent pas toutes de la même façon, c'est pourquoi vous devrez vous familiariser avec la technologie qu'elles utilisent et apprendre à extraire les informations qui vous intéressent. Beaucoup d'entre elles imposent des limites de dépôt, et les frais de fonctionnement du compte peuvent parfois être faramineux.

1.1.2 4 Comment fonctionne le marché des cryptomonnaies ?

Les marchés des cryptomonnaies sont décentralisés, ce qui signifie que ces monnaies ne sont pas émises par une autorité centrale (un État par exemple). Au lieu de cela, elles transitent via un réseau d'ordinateurs. Les cryptomonnaies peuvent être achetées et vendues sur des plateformes d'échange (sorte de place boursière virtuelle) et détenues sur des portefeuilles dédiés (ou wallets).

Contrairement aux devises traditionnelles, les cryptomonnaies sont uniquement une preuve numérique de propriété stockée sur une blockchain (ou chaîne de blocs en français). Lorsqu'un utilisateur souhaite vendre des unités de cryptomonnaies à un autre utilisateur, il doit les envoyer sur le portefeuille virtuel de ce dernier. La transaction est considérée comme complétée uniquement lorsqu'elle a été vérifiée et ajoutée à la blockchain au travers d'un processus de minage. C'est aussi de cette façon que sont généralement créés de nouveaux jetons.

1.1.2 5 Les caractéristiques des cryptomonnaies ?

- La volatilité des cryptomonnaies
- Les horaires de négociation des cryptomonnaies
- Une meilleure liquidité
- Possibilité de se positionner à l'achat ou à la vente

1.1.3 Problématique

La crypto-monnaie est une nouvelle forme de monnaie créée vers 2009 grâce au livre blanc présenté par Satoshi Nakamoto pour numériser et stocker la «valeur» sur Internet, connue sous le nom de Bitcoin. Depuis lors, des milliers d'autres crypto-monnaies ont été créées qui offrent une formidable opportunité pour les événements du futur facilité à l'échelle mondiale.

La valeur marchande actuelle de toutes les crypto-monnaies au moment de la rédaction de cet article est de près de 400 milliards de dollars, offrant une énorme opportunité de croissance du capital, comme en témoigne la forte hausse des prix des crypto-monnaies et des plafonds du marché en 2017.

À l'opposé de cette possibilité, le marché de la crypto-monnaie implique une forte volatilité et de nombreux risques.

Une grande partie de l'investissement actuel dans la crypto-monnaie est basée sur un environnement «buy & hodl», mais une solution automatisée est souhaitée pour maintenir et augmenter le capital investi avec une participation limitée des commerçants. Compte tenu de l'énorme opportunité (et des risques égaux), nous voulons un logiciel qui aide à la fois les investisseurs débutants et expérimentés en crypto-monnaie à investir dans la crypto-monnaie.

1.1.4 Contexte du projet

Il existe un besoin pour une solution logicielle capable d'investir dans les marchés de crypto-monnaie (tels que Binance, bitmex) qui achètent, vendent et détiennent selon différentes stratégies pour augmenter les bénéfices d'une manière facilement gérable, hautement fiable et indépendante.

Comment concevoir un bot cryptographique en Python

Les robots de trading automatisés sont très courants sur les grandes bourses, telles que la Bourse de New York, en raison de leur fonctionnalité, de leur vitesse et de leurs règles de négociation prédéfinies.

Exemple de stratégie de trading " Scalping " utilisée par Crypto Trading Bot avec entrée / sortie de prix.

1.2 Conduite et gestion de projet :

Cette section recense les contraintes ou les restrictions particulières qui limitent le projet :

1.2.1 Contraintes techniques :

- Échec technique

D'une manière générale, le processus de trading automatisé semble assez simple : installer un logiciel, créer des règles et regarder le trading se dérouler. En réalité, bien que le trading automatisé soit une méthode de trading sophistiquée, il a toujours la capacité de faire des erreurs. En raison de problèmes liés à Internet, il est possible que l'ordre ne soit pas généré. Ou peut-être que les "transactions théoriques" créées par une stratégie spécifique se transforment en transactions réelles en raison d'une anomalie de la plateforme. Il est toujours conseillé de commencer par des transactions de petite taille jusqu'à ce que le système de trading automatique "apprenne" et affine son processus.

- Contrôle manuel

Bien que le processus de trading soit automatisé, le système nécessite toujours un certain niveau de surveillance. Cette surveillance manuelle doit être effectuée afin de minimiser et de vérifier les défaillances techniques potentielles, les problèmes d'Internet ou de serveur, les pannes de système, etc. Il est possible que des "anomalies" surviennent dans un système de trading automatisé et que des ordres soient erronés, ignorés ou dupliqués. Grâce à une surveillance constante, ces problèmes peuvent être rapidement identifiés et résolus. Par exemple, en 2010, une erreur algorithmique a provoqué un effondrement massif du marché boursier de plusieurs milliards de dollars.

- Suroptimisation des algorithmes

Il est possible que les traders qui utilisent le backtesting pour affiner leurs stratégies créent des systèmes qui fonctionnent très bien en théorie, mais dont les performances sont catastrophiques dans un scénario réel. L'ajustement extrême des courbes génère un plan de trading peu fiable et inadapté au trading réel. Les traders ont souvent tendance à supposer qu'un plan de trading doit être composé de toutes les transactions rentables afin de ne jamais connaître de baisse. Un tel plan "presque parfait" peut également échouer sur le marché réel.

1.3 Conclusion :

Dans le présent chapitre, nous avons situé notre projet de fin d'année, en présentant d'une part le contexte général et la problématique, ainsi que les concepts clés de notre sujet. D'autre part nous avons présenté l'enchaînement et le déroulement de notre projet

CHAPITRE 2

Gestion de Projet Cryptocurrency Trading Bot v1.0

Dans ce chapitre, on va donner une vision globale sur le projet de Trading Bot v1.0, nous décrirons l'objectif de cette application ainsi que ses fonctionnalités dans un document appelé cahier des charges.

1.1.Introduction

Tout d'abord, il faut élaborer une description générale sur les composants et les tâches à accomplir, pour cela nous focalisons sur l'étude de projet au préalable, en utilisant des méthodes qui permettent une bonne gestion et structuration de projet.

1.2. Déroulement du projet

Le déroulement de notre projet a été basé sur le principe une méthode agile qui s'appelle SCRUM qui présente un cadre de développement dans lequel des équipes plurifonctionnelles réalisent des produits de manière itérative et incrémentale sous forme de Spins, nous avons décomposé notre projet en différentes phases, à chaque phase nous avons effectué des réunions pour discuter sur ce que nous avons fait? et sur ce que nous allons faire?, le principe de SCRUM nous permet d'accepter tout changement au cours de réalisation de notre projet.

1.3. Structuration du projet

1.3.1. Analyse fonctionnelle et définition des objectifs

Lors de cette phase préalable au démarrage du projet, les parties prenantes définissent ensemble :

- Les objectifs et la portée du projet,
- les livrables attendus,
- les délais souhaités,
- le budget alloué,
- le degré de souplesse qui pourra être accordé.

1.3.2. WBS (Work Breakdown Structure)

WBS Consiste à la définition complète des activités (série d'opérations élémentaires) nécessaires pour la production des composants et livrables du projet.

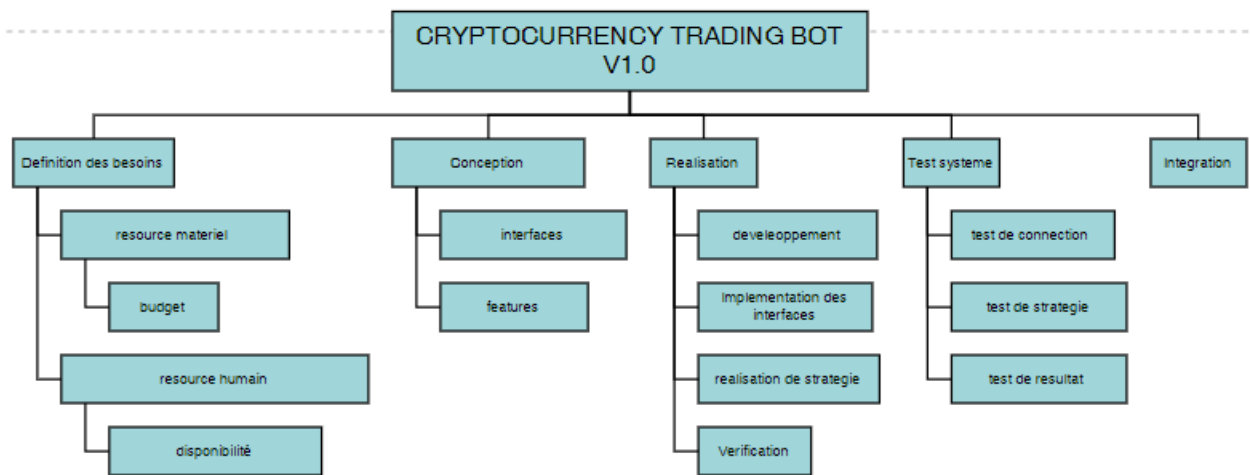


Figure 2 work breakdown structure

L'objectif pour l'équipe c'est de disposer d'une vision globale du projet en terme de tâches, dont on peut estimer la durée, les ressources requises et le coût budgétés, pour réaliser le projet

1.4. Identification des risques :

L'équipe essayé de lister les risques avant de commencer le développement pour avoir une idée globale sur le projet, voilà les principaux risques pour le projet :

- Risque concernant le fournisseur de données (Binanace.com)
- Risque concernant les limites de compte développeur (version gratuite)
- Problème de ressources matériel (pour le test du projets)
- Les deadlines de livraison

1.5. Le Cycle De Vie Du Projet

On va développer dans ce chapitre les principaux axes de connaissance ci-dessous sur lesquels nous avons travaillé pour structurer nos recherches et nécessaires pour mieux comprendre le projet et ses enjeux techniques :

- La collecte de données

- La préparation des données
- Présentation et domaines d'application du machine Learning
- Les différents types d'apprentissage
- Les modèles générés par les algorithmes d'apprentissage et leur évaluation
- Les architectures typiques de la machine Learning

1.5.1. La Collecte De Données

Dans notre projet on vise à collecter les données via un connecteur API de Binance et Bitmex.

1.5.2. Compréhension Des Données

Lorsque les données ont été collectées, il est important de découvrir les différents data sets, ce qui permet de mieux connaître les données et de déterminer le traitement à leur appliquer avant qu'elles deviennent exploitables dans l'étape de modélisation.

1.5.3. Transformation des données et traitement

Comme les algorithmes de machine Learning ne peut traiter que les variables numériques alors il est nécessaire de transformer les données catégoriques en valeurs numériques et les traiter par notre stratégie de trading.

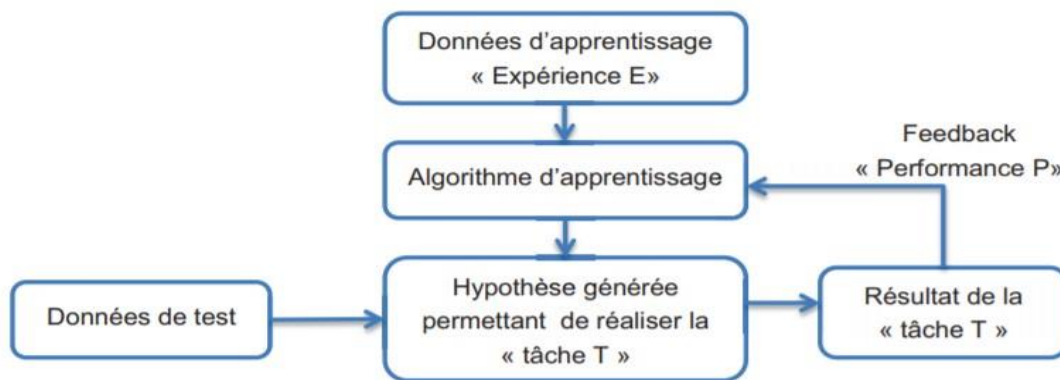


Figure 31.5.3. Transformation des données et traitement

1.5.4. Planification

1.5.4.1. Développement de diagramme de PERT :

Tache	Durée
Création de l'API	7
Conception	7
Liste des prototypes	7
Initialisation de l'interface de décision helper	10
Initialisation de l'interface de logging	10
Initialisation de l'interface de stratégie	10
Implémentation de décision helper	5
Implémentation de logging	6
Config de stratégie et implémentation	7
Intégration de l'API	8
Gérer les exception	8
Création des interfaces	8
Test de projet	6

1.5.4.2. Le diagramme de PERT:

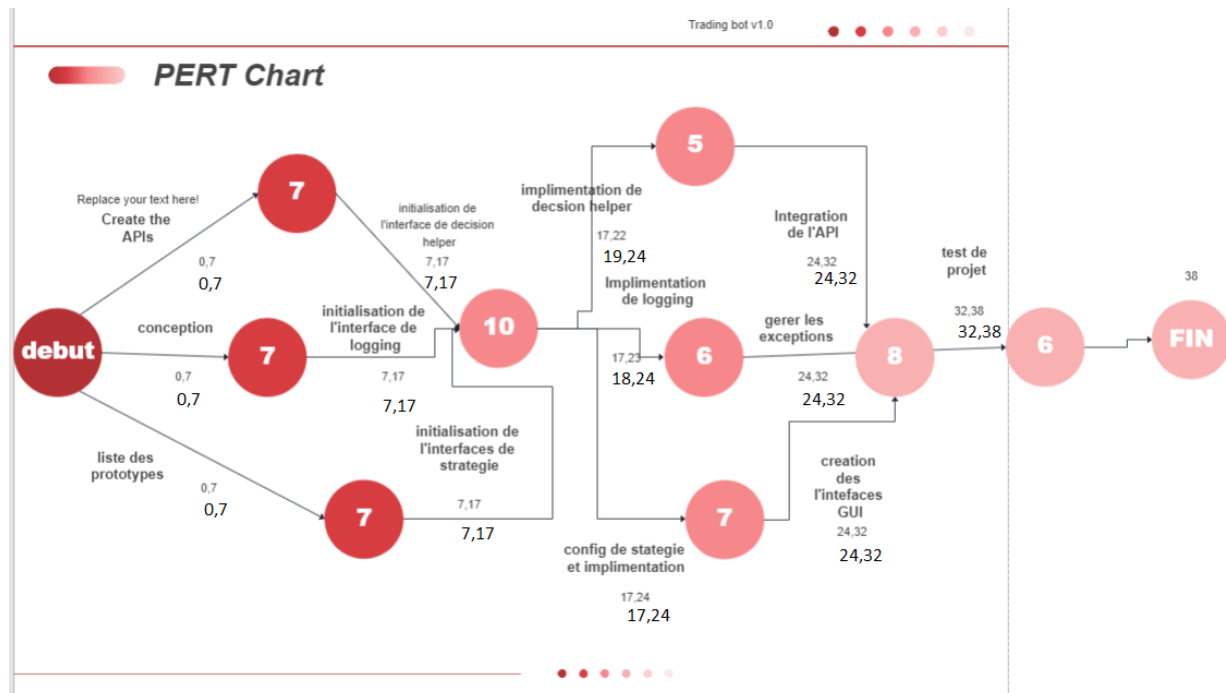


Figure 4 Diagramme de PERT

1.5.5. Développement de diagramme de GANTT :

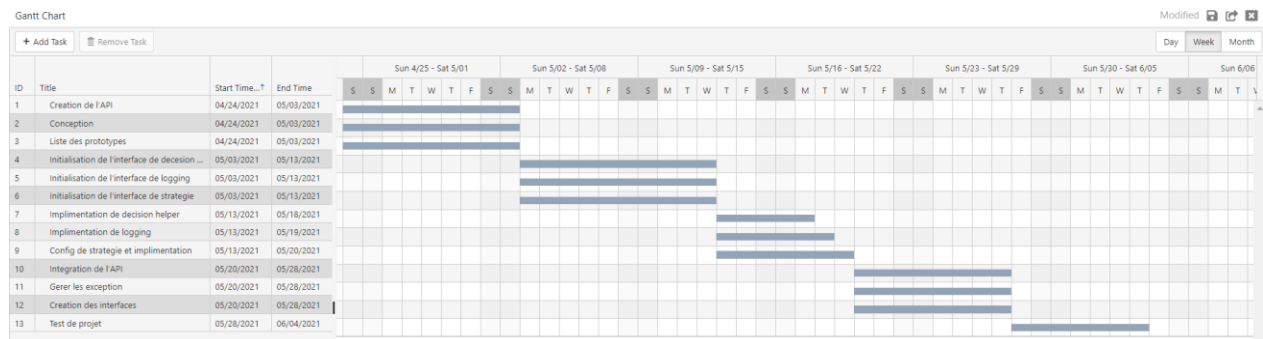


Figure 5 Diagramme de GANTT

1.6. Contrôle :

1.6.1. Gestion de modifications :

Pour la gestion de modification nous avons concentré sur GIT (un système de contrôle de version et de modifications), aussi nous avons concentré sur les meetings de SCRUM (Sprint Review) après

chaque sprint, pour bien structurer le travail, suivre le process et contrôler le workflow.

1.6.2. Gestion des risques :

Au début de projet on a plusieurs risques à gérer surtout les risque matériel et humain du cout nous avons choisir de crée plusieurs comptes de test pour profiter le maximum de l'API de Binance et Bitmex aussi de planifier le temps de disponibilité de chaque développeur, pour terminer le projet dans la durée voulu et en respectant les deadlines.

1.7. Cahier de charge :

CRYPTOCURRENCY TRADING BOT v1.0 est une application automatisée permettant de passer automatiquement des ordres, par exemple sur le marché, programmé pour suivre les instructions ou la stratégie d'un investisseur, le robot trading est opérationnel 24h/24 et 7j/7. Le système de cette application offre des fonctionnalités intéressantes come : Négocier, acheter et vendre des actions à très grande vitesse, il est destiné à réaliser des arbitrages en se substituant au trader afin d'optimiser ses transactions, Le plus souvent, le fonctionnement du robot est mimétique : il copie les positions des traders pour le compte d'un autre investisseur grâce à des algorithmes mathématiques.

Logique de fonctionnement du robot est de prendre des décisions d'investissements à la place d'un trader. Cet algorithme passe des ordres d'achat et de vente après traitement et synthèse de différents signaux émis par le marché. Le robot est opérationnel 24h/24 et procède à l'analyse des signaux en provenance des places financières en permanence. Les robots de trading les plus performants traitent 7 000 ordres en l'espace de quelques millisecondes.

Le robot offre **quatre interfaces** :

La première interface consiste à gérer les information reçus par l'API Et l'afficher sous forme d'un tableau avec les statistiques du marché, cette interface est destinée à l'utilisateur pour

l'aider à avoir une idée sur le marché et les valeurs de chaque devise.

La deuxième interface consiste à la gestion de la stratégie utilisé pour démarrer l'opération de trading, avec cette interface l'utilisateur peut choisir la stratégie le type de devise et le pourcentage de profite voulu.

La troisième interface consiste à afficher les logging c'est-à-dire le rapport des opérations effectuées durant l'opération de trading avec toutes les informations nécessaire come la date le type d'opération et le contenu.

La quatrième interface consiste a affiché les ordres en train de traiter avec des informations sur la durée, le statut, et les informations sur la devise utilisé.

L'application offre aussi des moyennes de gestion manuelle des opérations au cas des erreurs ou des futures informations sur le marché ce qui donne à l'utilisateur la flexibilité de gestion.

Pour la taille de l'équipe, il est défini comme suit : trois développeurs qui échangent les rôles des testeur, analyste et développeur, aussi pour les constraints ils ont défini comme suit :

- Contraint de budget pour le test
- Contraint de tarif de l'API
- Contraint concernant le temps de réalisation
- Contraint des ressources humaines

CHAPITRE 3

Modélisation et Conception

Ce chapitre présente l'étape de conception de projet, en utilisant le langage UML pour élaborer des diagrammes.

1.8. Introduction

Le processus de développement d'un projet informatique nécessite de passer par des différentes étapes. Dans ce chapitre nous focalisons sur l'étape de modélisation de projet qui consiste l'élaboration des diagrammes en utilisant le langage UML.

UML, c'est l'acronyme anglais pour « Unified Modeling Language », c'est un langage de modélisation orientée objet, il permet de représenter et de définir le processus d'élaboration des modèles. UML définit deux types de diagrammes, structurels (statiques) et comportementaux (dynamiques). Dans ce chapitre nous allons présenter des diverses modèles de type statique et dynamique.

1.9. Diagramme de classes

Le **diagramme de classes** est un schéma utilisé en génie logiciel pour présenter les classes et les interfaces des systèmes ainsi que les différentes relations entre elles. Ce

diagramme fait partie de la partie statique d'UML, puisqu'il fait abstraction des aspects temporels et dynamiques.

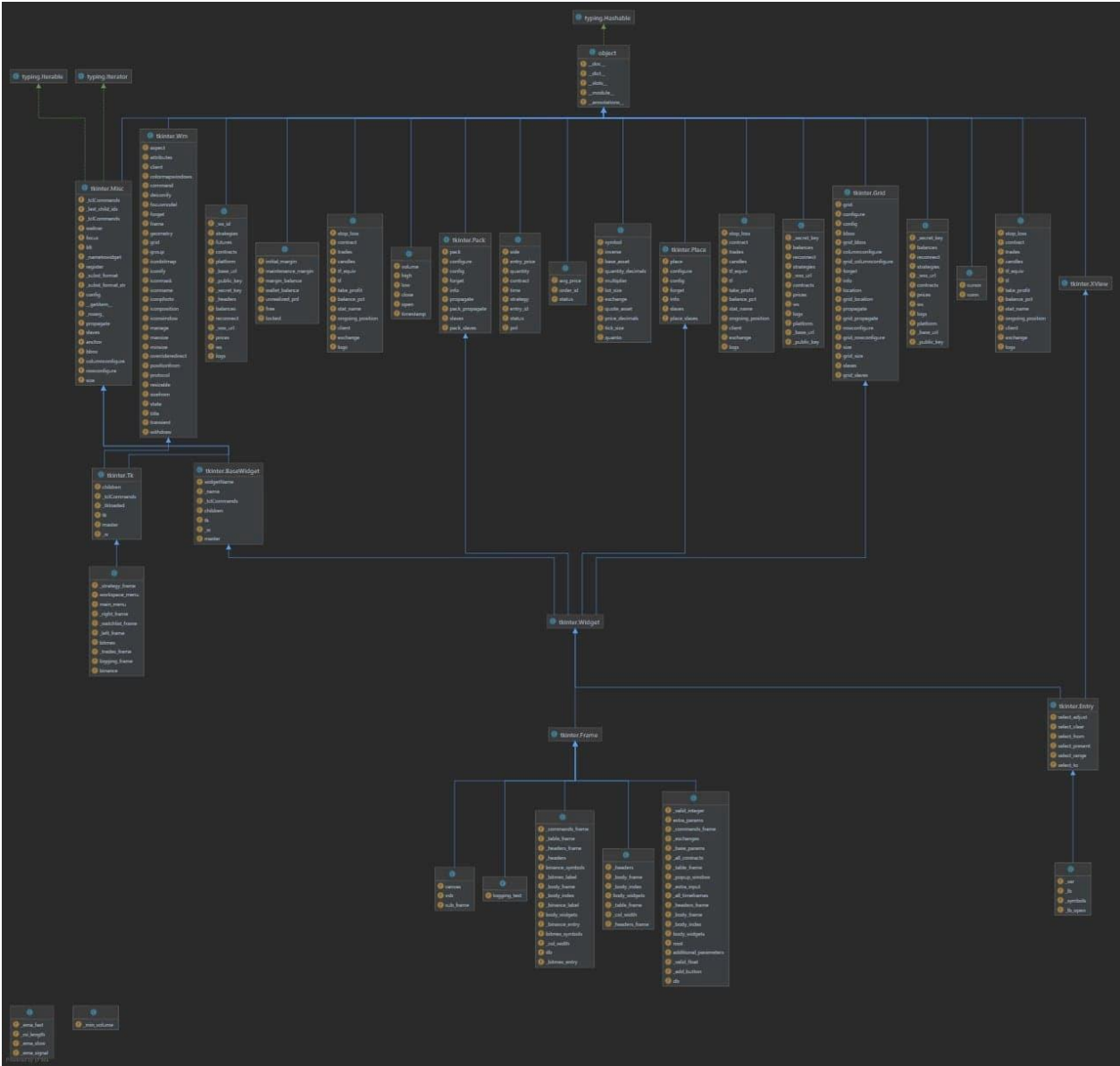


Figure 6 Diagramme de classe

1.10. Diagramme de séquence

Les **diagrammes de séquences** sont la représentation graphique des interactions entre les acteurs et le système selon un ordre chronologique dans la formulation Unified Modeling Language. ils permettent de montrer les interactions d'objets dans le cadre d'un scénario d'un diagramme des cas d'utilisation.

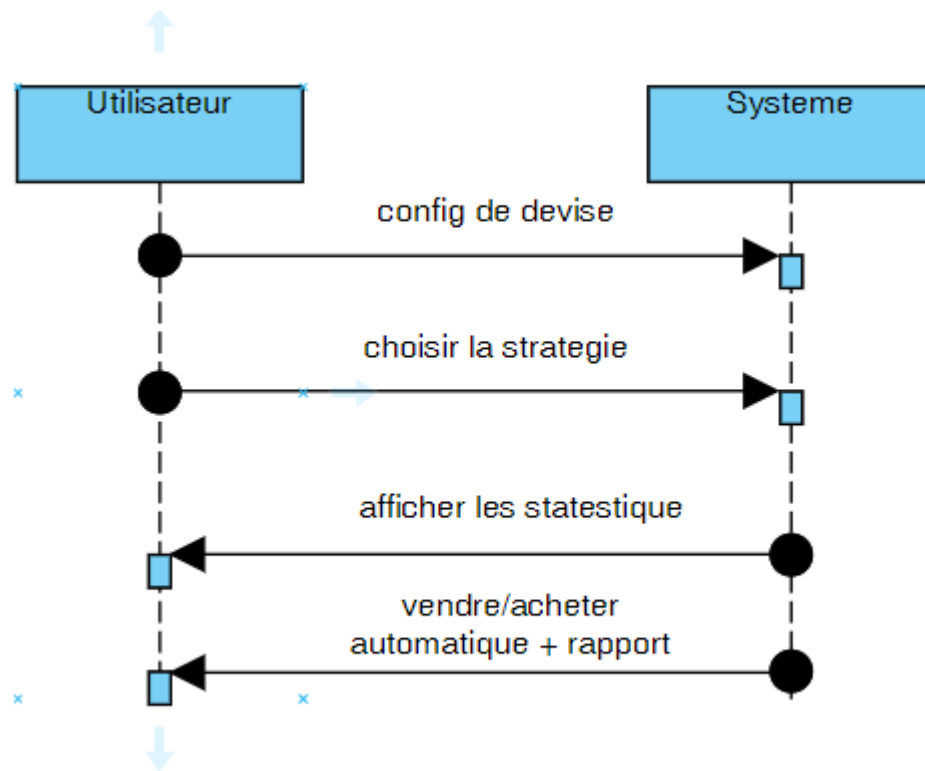


Figure 7 Diagramme de séquence

1.11. Diagramme d'activité

Un **diagramme d'activité** permet de modéliser le comportement du système, dont la séquence des actions et leurs conditions d'exécution. Les actions sont les unités de base du comportement du système.

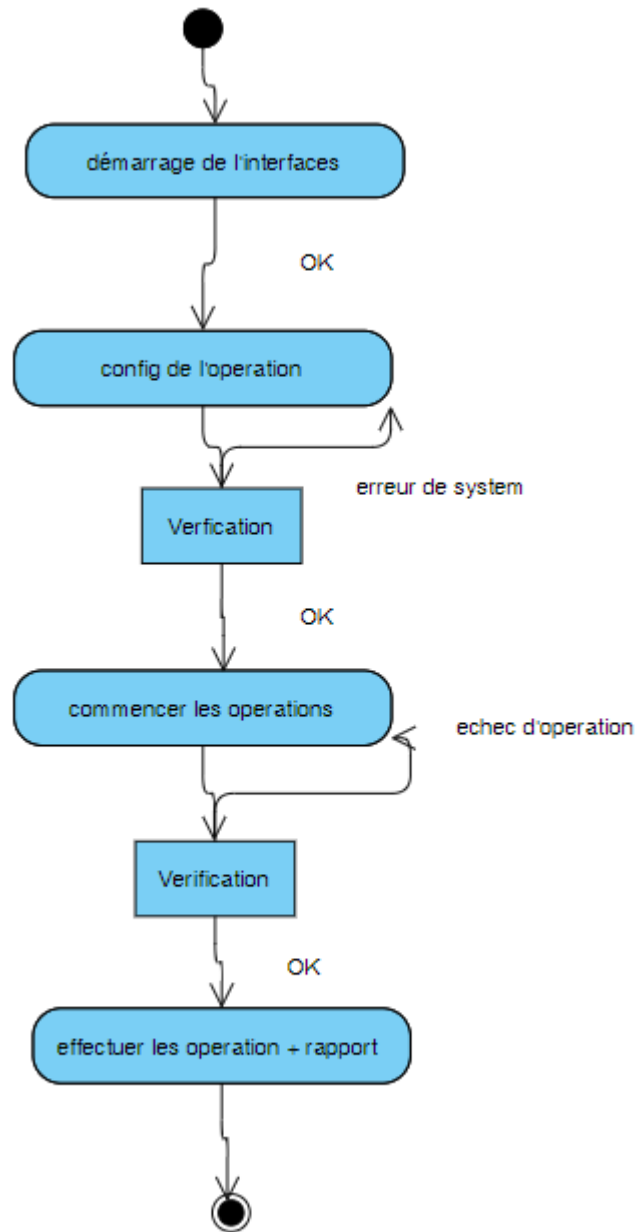


Figure 9 : Diagramme d'activité

1.12. Conclusion

L'importance de réalisation la conception de ce projet est de permettre à bien comprendre la démarche à suivre pour mettre en œuvre notre projet. Nous avons vu que UML permet une étude détaillée, grâce aux différents diagrammes qui sont très bénéfiques à la compréhension du logiciel. Il semble très efficace de se tenir à ce langage car il prend en compte la plupart des problématiques à résoudre.

CHAPITRE 4

Etude Technique

L'étude technique constitue l'une des étapes de la conception et de l'analyse de faisabilité d'un projet. Elle représente un préalable dans le sens où elle jette les premières bases du projet, c'est la phase d'adaptation de la conception à l'architecture technique retenue, tout en décrivant et documentant le fonctionnement de chaque unité logicielle.

Les étapes de la conception générale des différents modules sont considérées comme les étapes les plus critiques du projet. D'une part, elles doivent être élaborées avec attention et précision, vu que l'architecture à proposer doit rester conforme aux spécifications fonctionnelles.

D'autre part elle constitue la base de la réalisation qui concrétise les efforts fournis tout au long du projet..

3.1 Introduction

En général Un bot de trading, est un programme informatique, qui peut automatiquement passer des ordres (acheter, vendre) sur un marché, sans intervention humaine.

Crypto trading algorithmique est le processus consistant à appliquer le même ensemble de règles pour passer des ordres d'achat et de vente sur un marché de la cryptographie. Il s'agit d'appliquer les méthodes scientifiques au trading afin de développer une stratégie efficace

Dans ce projet, nous couvrons les avancées des techniques développées pour le trading algorithmique et l'analyse financière sur la base des récentes percées en apprentissage automatique. Nous tirons parti des techniques classiques largement utilisées et appliquées par les scientifiques des données financières

L'objectif principal de ce projet est d'automatiser le processus d'achat et de vente en fonction de nos signaux de modèle, tout en utilisant des connexions API à la gestion des échanges. Nous pouvons également l'utiliser comme assistant de trading à l'aide d'une interface utilisateur (UI) qui nous aiderons à suivre nos transactions.

3.2 Acquisition des données de L'API

3.2.1 Introduction

API signifie Application Programming Interface. En termes simples, les API sont un ensemble de fonctions et de procédures qui permettent la création d'applications qui accèdent aux données et fonctionnalités d'autres applications, services ou systèmes d'exploitation.

En ce qui concerne la façon dont nous créons des services Web, il existe actuellement deux principales approches concurrentes : SOAP et REST

Dans notre cas d'utilisation, nous utiliserons REST car il offre une alternative plus légère. De nombreux développeurs ont trouvé SOAP encombrant et difficile à utiliser. REST est facile à comprendre, et il est simple à écrire et à documenter.

REST prend en charge de nombreux formats de données, mais l'utilisation prédominante de JSON signifie une meilleure prise en charge des clients de navigateur.

3.2.2 Connecteurs API - Accès API aux données de marché

Il existe plusieurs options que nous pouvons utiliser pour accéder aux données de marché via une API utilisant Python. Nous présenterons tout d'abord quelques sources intégrées à la bibliothèque pandas et api binance qui facilite le téléchargement des données de marché et des données en temps réel des crypto.

Cette étape est parmi les bases de notre projet, c'est pour cela nous avons consacré pas mal de temps pour trouver les données du marché en temps réel.

Binance et Bitmex sont deux des échanges de crypto les plus populaires au monde, ils offrent des possibilités d'automatisation impressionnantes et sont parfaits pour les débutants. Vous pouvez ensuite appliquer ce que vous avez appris à votre compte Binance Spot/Margin ou à d'autres échanges.

Tout d'abord, nous commençons par créer une classe client API et commençons à analyser diverses données de Binance Futures et Bitmex.

En suite diffuser des données de marché dans notre programme à l'aide de l'API Websocket.

Améliorations des gestions des erreurs et variables, méthodes privées pour rendre notre code plus efficace et plus lisible grâce aux modèles de données et au typage des données. Et fiable avec tous les méthodes de gestion des erreurs.

3.3 Stratégies et trading - MACD and RSI

La divergence de convergence moyenne mobile (MACD) et l'indice de force relative (RSI) se classent parmi les indicateurs de dynamique les plus populaires utilisés dans le crypto trading. Lorsqu'ils sont utilisés en combinaison avec d'autres indicateurs techniques, MACD et RSI peuvent offrir une valeur dans la validation des opportunités commerciales et la synchronisation des transactions afin d'optimiser vos pratiques de gestion des risques.

Bien qu'ils représentent une approche similaire pour évaluer les transactions sur le forex, les fonctions du MACD et du RSI sont distinctes, ce qui en fait des indicateurs utiles à combiner dans l'évaluation des transactions. Voici comment utiliser MACD et RSI dans le cadre de votre analyse commerciale.

- Les bases du **MACD** et du **RSI**

Les histogrammes MACD offrent une représentation visuelle facile à la fois de la force d'une tendance des prix du forex ainsi que des signaux pouvant indiquer un changement de dynamique

Sur le graphique CHF/JPY ci-dessous, les écarts entre l'EMA de 26 jours (la ligne bleue) et la ligne de signal (la ligne rouge) sont en corrélation avec l'élán du mouvement des prix. Plus l'écart entre ces lignes est grand, plus la tendance est forte dans les deux sens.

Lorsque ces lignes convergent puis se croisent, cela peut indiquer un changement d'élán que les traders peuvent utiliser pour planifier leur sortie d'une position ou pour planifier une nouvelle position qui vise à capitaliser sur cette opportunité. Dans le graphique en CHF ci-dessous, deux de ces croisements coïncident avec des changements dans le mouvement des prix pour la paire de devises.



Figure 8 : graphique CHF/JPY

Bien que le RSI mesure également l'élán, il reflète cet élán à travers une approche analytique différente. RSI peut identifier les opportunités d'achat en identifiant les opportunités commerciales où

les conditions d'une paire de devises sont soit surachetées, soit survendues. Dans les situations où les conditions sont surachetées, RSI suggère que le prix peut être gonflé et prêt à baisser.

Dans des conditions de survente, le RSI suggère que les traders ont peut-être réagi de manière excessive et ont maintenant créé de la valeur en déprimant le prix et la demande pour une paire de devises. La fonctionnalité du RSI est évidente dans le graphique GBP/USD ci-dessous, où une plongée dans des conditions de survente - en dessous de la fourchette violette sur le graphique RSI - conduit à une augmentation rapide du prix de la paire de devises.



Figure 9 : graphique GBP/USD

Sachant que ces indicateurs de dynamique évaluent les opportunités commerciales par différentes méthodes, ils peuvent bien fonctionner ensemble pour valider les signaux que chacun fournit, donnant aux traders plus de confiance lorsque les deux indicateurs s'accordent sur une opportunité d'achat potentielle.

3.4 Modèles

Modèle : Un modèle du machine Learning peut être une représentation mathématique d'un Processus réel, L'algorithme d'apprentissage trouve des modèles dans les données d'apprentissage de sorte que les

paramètres d'entrée correspondent à la cible. Le résultat du processus de formation est un modèle du machine Learning que nous pouvons ensuite utiliser pour faire des prédictions des réponses.

Dans notre projet le principal défi de l'apprentissage automatisé est d'identifier des modèles dans les données de formation qui sont significatif lors de la généralisation de l'apprentissage du modèle à de nouvelles données. Il y a un grand nombre de modèles potentiels qu'un modèle pourrait identifier, tandis que les données d'apprentissage ne constituent qu'un échantillon du plus grand ensemble de phénomènes que l'algorithme peut rencontrer lors de l'exécution la tâche à l'avenir.

1. Le flux de travail d'apprentissage automatique

Développer une solution de ML pour une stratégie de trading algorithmique nécessite une approche pour maximiser les chances de succès tout en économisant les ressources. C'est aussi très important de rendre le processus transparent et reproductible afin de faciliter collaboration, maintenance et améliorations ultérieures.

Le tableau suivant présente les étapes clés, de la définition du problème au déploiement d'une solution prédictive :

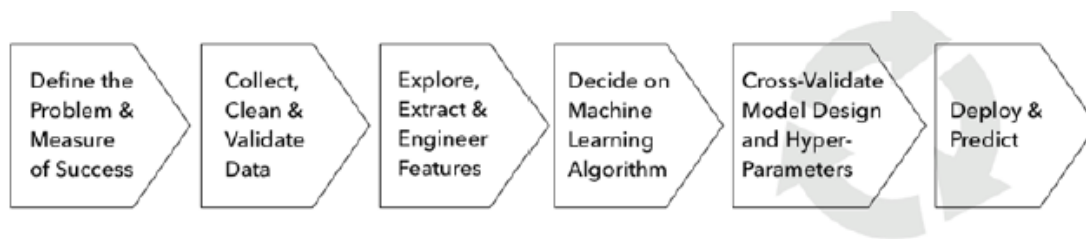


Figure 10 : flux de travail d'apprentissage automatique

Le processus est itératif tout au long, et l'effort requis à différentes étapes variera selon le projet. En général, cependant, ce processus devrait inclure les éléments suivants pas:

1. Cadrez le problème, identifiez une métrique cible et définissez le succès.
2. Sourcez, nettoyez et validez les données.
3. Comprenez vos données et générez des fonctionnalités informatives.
4. Choisissez un ou plusieurs algorithmes d'apprentissage automatique adaptés à vos données.

5. Entraînez, testez et ajustez vos modèles.

6. Utilisez votre modèle pour résoudre le problème d'origine.

Nous allons parcourir ces étapes dans les sections suivantes en utilisant un exemple simple pour illustrer certains des points clés.

3.5 Apprentissage

L'apprentissage est une phase décisionnelle, dans le cycle de vie du projet ; le projet peut avoir naissance ou passer à un autre nouveau cycle de modification.

C'est dans cette phase que les poids du modèle sont calculés. Ils sont calculés et modifiés dans des itérations de cycle d'apprentissage. Un cycle d'apprentissage est constitué de deux actions primordiales ; forward propagation (propagation vers l'avant), c'est là où un résultat ou une prédiction est calculée. Puis la propagation vers l'arrière ou back propagation, dans laquelle l'output du forward propagation est comparé avec le résultat attendu puis on se déplace en arrière pour ajuster les poids en fonction de la fonction gradient. Un point à mentionner ici, c'est que si le gradient dans un nœud est proche du 0, c'est que ce nœud n'a pas encore besoin d'apprentissage et ses poids sont les plus adéquats. Mais s'il est proche de 1, c'est qu'il a encore besoin d'apprendre pour ajuster ses poids.

Le modèle par récupération n'a pas besoin d'apprentissage, car c'est un algorithme de l'intelligence artificielle. Mais le modèle génératif en a sûrement besoin. Il est fortement conseillé que l'apprentissage doit être fait dans des machines d'hautes performances GPU ou CPU, et puisque nous n'avons pas des machines de ces caractéristiques, nous avons effectué cette phase, dans un premier temps, sur google colaboratory car c'est gratuit et il offre beaucoup d'avantages et surtout la GPU. Puis, nous avons essayé de prendre avantage des services cloud de Microsoft Azure. Ces services sont payants, mais nous avons bénéficié d'une somme de 200\$, grâce à nos emails universitaire (.edu), par laquelle nous avons pu louer des machines avec les performances qu'on cherche. Nous avons fait l'apprentissage deux fois puis nous avons comparé et évalué les résultats afin de prendre la bonne décision concernant l'architecture du modèle génératif.

Ce processus nous a pris presque trois heures d'apprentissage sur Google colab et presque deux heures et demie sur Microsoft azure.

3.6 Test

Après les 4000 itérations, nous avons atteint une perte moyenne de : 2.6038 avant qu'il était 8.9796 avec la première architecture. En plus les tests à main sont également merveilleux et excellents

3.7 Conclusion

Nous pouvons encore améliorer cette application, nous devons acquérir une solide compréhension de la terminologie et de la méthodologie financières et une expérience pratique de la conception et de la construction de modèles d'apprentissage automatique financier. et nous devrions également évaluer et valider différentes stratégies de trading algorithmiques. Nous devons essayer le backtesting qui est le Saint Graal du trading algorithmique et est une clé essentielle pour réussir le déploiement d'algorithmes fiables.

CHAPITRE 5

Les technologies, les outils utilisés et la démonstration de l'application

Ce chapitre résume toutes les techniques, les méthodes et les technologies utilisées afin d'aboutir à la réalisation de l'application ciblée. Ainsi, une capture du produit final

5.1 Choix logiciels

Le développement d'applications web et de sites Internet présente des points communs. D'où l'utilisation des Framework qui vient avec un ensemble cohérent de composants logiciels structurels, qui sert à créer les fondations ainsi que les grandes lignes d'une partie d'un logiciel (architecture). Notre projet se base totalement sur des technologies open-source.

5.1.1 Langage Python

Python est un langage de programmation interprété, multi-paradigme et multiplateformes. Il favorise la programmation impérative structurée, fonctionnelle et orientée objet. Il est doté d'un typage dynamique fort, d'une gestion automatique de la mémoire par ramasse-miettes et d'un système de gestion d'exceptions ; il est ainsi similaire à Perl, Ruby, Scheme, Smalltalk et Tcl.

À la fin des années 1980, le programmeur Guido van Rossum, participe au développement du langage de programmation ABC au Centrum voor Wiskunde en Informatica

(CWI) d'Amsterdam, aux Pays-Bas. Il travaillait alors dans l'équipe du système d'exploitation Amoeba dont les appels systèmes étaient difficilement interfaçables avec le Bourne shell utilisé comme interface utilisateur. Il estime alors qu'un langage de script inspiré d'ABC pourrait être intéressant comme interpréteur de commandes pour Amoeba.

En 1989, profitant d'une semaine de vacances durant les fêtes de Noël, il utilise son ordinateur personnel pour écrire la première version du langage, et c'est ici où python commence. Fan de la série télévisée Monty Python's Flying Circus, il décide de baptiser ce projet Python.

Le langage Python est placé sous une licence libre proche de la licence BSD et fonctionne sur la plupart des plates-formes informatiques, des smartphones aux ordinateurs centraux, de Windows à Unix avec notamment GNU/Linux en passant par macOS, ou encore Android, iOS, et peut aussi être traduit en Java ou .NET. Il est conçu pour optimiser la productivité des programmeurs en offrant des outils de haut niveau et une syntaxe simple à utiliser.

Il est également apprécié par certains pédagogues qui y trouvent un langage où la syntaxe, clairement séparée des mécanismes de bas niveau, permet une initiation aisée aux concepts de base de la programmation.

Python est un langage qui peut s'utiliser dans de nombreux contextes et s'adapter à tout type d'utilisation grâce à des bibliothèques spécialisées. Il est cependant particulièrement utilisé comme langage de script pour automatiser des tâches simples mais fastidieuses, comme un script qui récupérerait la météo sur Internet ou qui s'intégrerait dans un logiciel de conception assistée par ordinateur afin d'automatiser certains enchaînements d'actions répétitives (voir la section Adoption).

On l'utilise également comme langage de développement de prototype lorsqu'on a besoin d'une application fonctionnelle avant de l'optimiser avec un langage de plus bas niveau.

Il est particulièrement répandu dans le monde scientifique, et possède de nombreuses bibliothèques optimisées destinées au calcul numérique.



Figure 11 :python

5.1.2 Numpy

NumPy est une extension du langage de programmation Python, destinée à manipuler des matrices ou tableaux multidimensionnels ainsi que des fonctions mathématiques opérant sur ces tableaux.

Plus précisément, cette bibliothèque logicielle libre et open source fournit de multiples fonctions permettant notamment de créer directement un tableau depuis un fichier ou au contraire de sauvegarder un tableau dans un fichier, et manipuler des vecteurs, matrices et polynômes.

NumPy est la base de SciPy, regroupement de bibliothèques Python autour du calcul scientifique.

5.1.3 Scikit-learn

Scikit-learn est une bibliothèque libre Python destinée à l'apprentissage automatique. Elle est développée par de nombreux contributeurs² notamment dans le monde académique par des instituts français d'enseignement supérieur et de recherche comme Inria³. Elle comprend notamment des fonctions pour estimer des forêts aléatoires, des régressions logistiques, des algorithmes de classification, et les machines à vecteurs de support. Elle est conçue pour s'harmoniser avec d'autres bibliothèques libres Python, notamment NumPy et SciPy.



Figure 12 : Scikit-learn

5.1.4 google colab

Colaboratory, souvent raccourci en "Colab", est un produit de Google Research. Colab permet à n'importe qui d'écrire et d'exécuter le code Python de son choix par le biais du navigateur. C'est un environnement particulièrement adapté au machine learning, à l'analyse de données et à l'éducation. En termes plus techniques, Colab est un service hébergé de notebooks Jupyter qui ne nécessite aucune configuration et permet d'accéder gratuitement à des ressources informatiques, dont des GPU.

Nous l'avons utilisé comme notebook et aussi pour la phase de l'apprentissage.



Figure 13 : google colab

4.1.5 Postman

Postman est un outil de développement logiciel. Il permet aux utilisateurs de tester les appels aux API. Les utilisateurs de Postman saisissent des données. Les données sont envoyées à une adresse de serveur Web spéciale. En règle générale, les informations sont renvoyées, que Postman présente à l'utilisateur. Nous l'avons utilisé lors de la phase du développement pour tester les routes de l'api.



Figure 14 Postman

5.1.6 pycharm

PyCharm est un environnement de développement intégré utilisé pour programmer en Python. Il permet l'analyse de code et contient un débogueur graphique. Il permet également la gestion des tests unitaires, l'intégration de logiciel de gestion de versions, et supporte le développement web avec Django.



Figure 15 pycharm

5.1.7 vs Code

Visual Studio Code est un éditeur de code source conçu par Microsoft pour Windows, Linux et macOS.[9] Les fonctionnalités incluent la prise en charge du débogage, la mise en évidence de la syntaxe, la complétion intelligente du code, les extraits de code, la refactorisation du code et Git intégré.

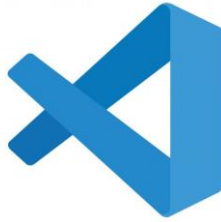


Figure 16 vs Code

5.1.8 Pandas

Pandas est une bibliothèque écrite pour le langage de programmation Python permettant la manipulation et l'analyse des données. Elle propose en particulier des structures de données et des opérations de manipulation de tableaux numériques et de séries temporelles. Pandas est un logiciel libre sous licence BSD.

Les principales structures de données sont les séries (pour stocker des données selon une dimension - grandeur en fonction d'un index), les DataFrames (pour stocker des données selon 2 dimensions - lignes et colonnes), les Panels (pour représenter des données selon 3 dimensions, les Panels4D ou les DataFrames avec des index hiérarchiques aussi nommés MultiIndex (pour représenter des données selon plus de 3 dimensions - hypercube)

5.1.9 Matplotlib

Matplotlib est une bibliothèque du langage de programmation Python destinée à tracer et visualiser des données sous formes de graphiques⁵. Elle peut être combinée avec les bibliothèques python de calcul scientifique NumPy et SciPy⁶. Matplotlib est distribuée librement et gratuitement sous une licence de style BSD⁴. Sa version stable actuelle (la 2.0.1 en 2017) est compatible avec la version 3 de Python.

Plusieurs points rendent cette bibliothèque intéressante :

- Export possible en de nombreux formats matriciels (PNG, JPEG...) et vectoriels (PDF, SVG...)
- Documentation en ligne en quantité, nombreux exemples disponibles sur internet.
- Forte communauté très active.
- Interface pylab : reproduit fidèlement la syntaxe MATLAB.

- Bibliothèque haut niveau : idéale pour le calcul interactif.

5.1.10 Pickle

Pickle est un module de python qui permet de sauvegarder une ou plusieurs variables dans un fichier et de récupérer leurs valeurs ultérieurement. Les variables peuvent être de type quelconque.

En clair, si pour une raison quelconque, dans un script **Python**, vous avez besoin de sauvegarder, temporairement ou même de façon plus pérenne, le contenu d'un objet Python comme une liste, un dictionnaire, un tuple etc etc ... au lieu d'utiliser une base de données ou un simple fichier texte, le module **pickle** est fait pour ça.

5.1.11 Sqlit

SQLite est un système de gestion de base de données relationnelle contenu dans une bibliothèque C. Contrairement à de nombreux autres systèmes de gestion de bases de données, SQLite n'est pas un moteur de base de données client-serveur. Au contraire, il est intégré dans le programme final. SQLite suit généralement la syntaxe PostgreSQL



Figure 17 Sqlit

5.1.12 Git

Git est un système de contrôle de version distribué gratuit et open source conçu pour gérer tout, des petits aux très grands projets avec rapidité et efficacité. Git est facile à apprendre et a une petite empreinte avec des performances ultra-rapides. Il surclasse les outils SCM tels que Subversion, CVS, Perforce et ClearCase avec des fonctionnalités telles que les branchements locaux, les zones de transfert pratiques et les flux de travail multiples.



Figure 18 Git

Il fonctionne comme le module JSON mais n'est pas limité à un seul format d'objet.

5.1.13 Azur_computing

Microsoft Azure, communément appelé Azure, est un service de cloud computing créé par Microsoft pour créer, tester, déployer et gérer des applications et des services via des centres de données gérés par Microsoft. Il fournit des logiciels en tant que service (SaaS), une plate-forme

en tant que service (PaaS) et une infrastructure en tant que service (IaaS) et prend en charge de nombreux langages de programmation, outils et cadres différents, y compris des logiciels et des systèmes spécifiques à Microsoft et tiers. Nous l'avons utilisé, ainsi que google colab, pour la phase de l'apprentissage.



Figure 19 azur

5.2 Application :

Trading bot avec une interface d'utilisateur avec 4 composants

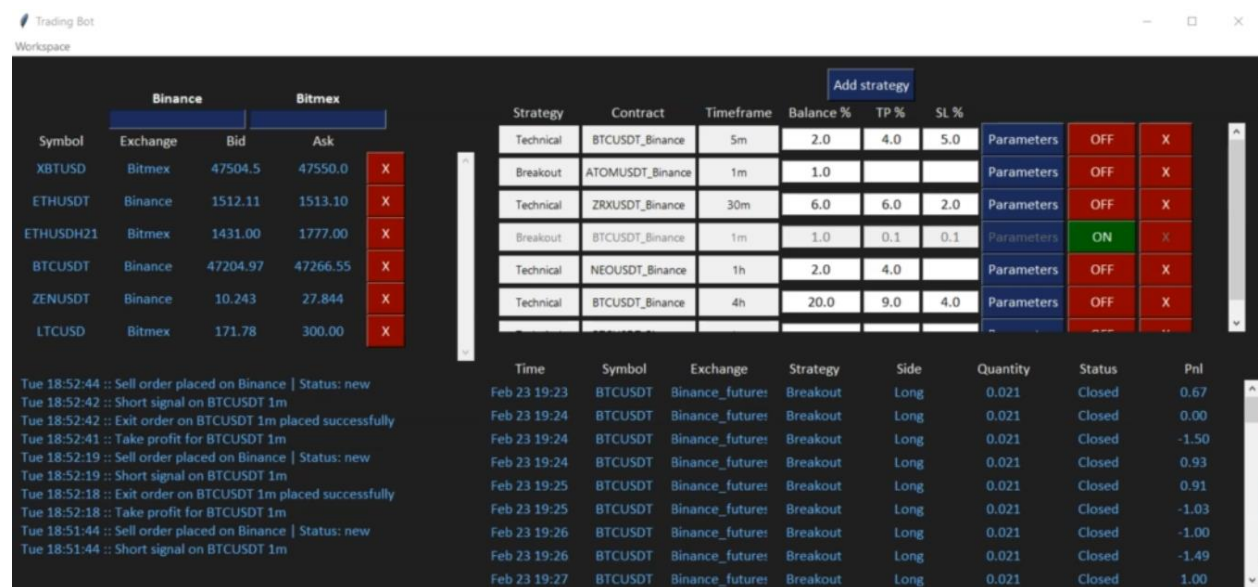


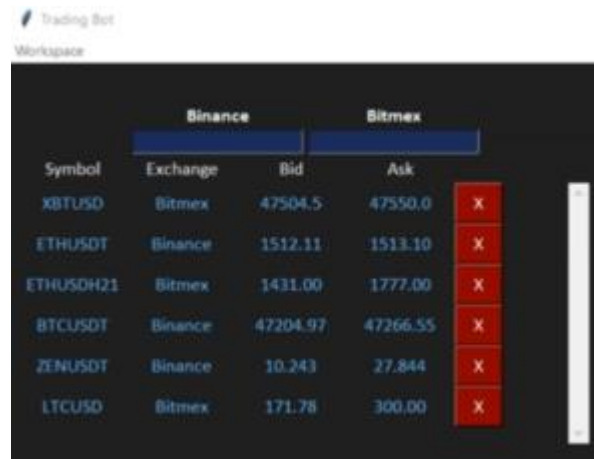
Figure 20 interface d'utilisateur de l'application

5.2.1 afficher les données de marché en direct

En particulier dans le cas où nous construisons un outil de trading semi-automatisé où le trader aime garder un certain contrôle sur le bot.

Par exemple, vous pouvez laisser le programme :

Ouvrir une position BTCUSD pour vous, mais vous préférez toujours la fermer vous-même lorsque certaines conditions du marché sont remplies, lorsque certaines nouvelles sont publiées, ou toute autre condition que vous avez définie et que vous devez vérifier manuellement



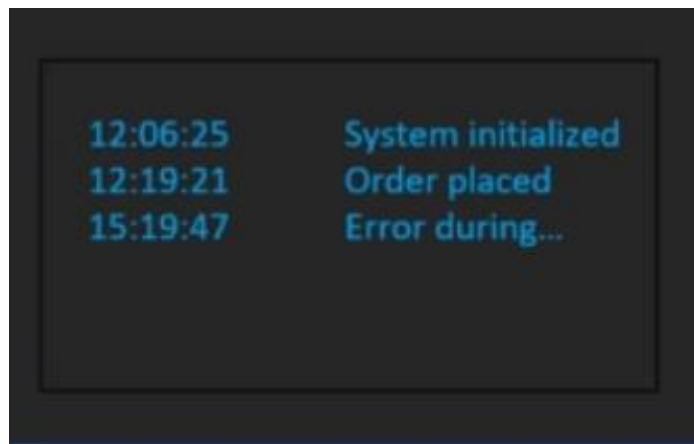
Trading Bot
Workspace

Symbol	Binance		Bitmex	
	Exchange	Bid	Ask	
XBTUSD	Bitmex	47504.5	47550.0	X
ETHUSD	Binance	1512.11	1513.10	X
ETHUSDH21	Bitmex	1431.00	1777.00	X
BTCUSD	Binance	47204.97	47266.55	X
ZENUSD	Binance	10.243	27.844	X
LTCUSD	Bitmex	171.78	300.00	X

Figure 21 afficher les données de marché en direct

5.2.2 Afficher des messages à l'utilisateur (logging)

Nous voulons informer l'utilisateur de ce qui se passe à l'intérieur de l'application C'est aussi ce qu'on appelle la journalisation(logging).



12:06:25	System initialized
12:19:21	Order placed
15:19:47	Error during...

Figure 22 Afficher des messages à l'utilisateur (logging)

5.2.3 Mettre en place une stratégie

Une fois que nous avons les données du marché, nous voulons que le programme prenne des décisions sur l'ouverture ou la fermeture d'une position en fonction d'une logique prédéfinie.

Cependant, cette logique peut nécessiter certaines entrées de la part de l'utilisateur.

Par exemple le pourcentage de stop loss, le pourcentage du solde à allouer à la stratégie ou toute autre chose dont vous auriez besoin pour votre stratégie et vous voudriez certainement pouvoir démarrer et arrêter une stratégie avec un bouton, avec un "ON" et le bouton « OFF ».

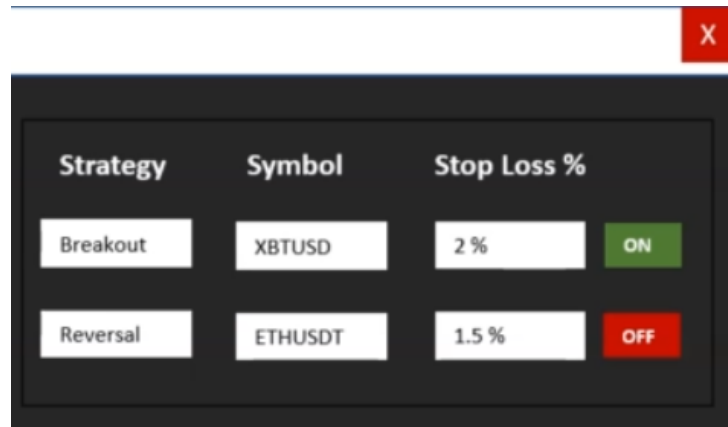


Figure 23 Mettre en place une stratégie

5.2.4 Suivre les ordres /TRADES (PNL ,statues des Ordre)

une fois qu'un ordre a été passé ou qu'une transaction a eu lieu, nous voulons pouvoir suivre ce qui arrive à la position quelle est la situation actuelle

PNL, est la position ouverte ou fermée ou toute autre chose dont vous avez besoin l'heure de l'entrée, l'heure de la sortie

Time	Symbol	PNL	Status
22:08	BTCUSDT	48 USDT	Open
19:21	BTCUSDT	98 USDT	Closed

Figure 24 Suivre les ordres /TRADES (PNL ,statues des Ordre)

6 Conclusion

Comme vous remarquez dans ce chapitre on a utilisé quelques outils qui nous ont facilité la tâche de développement et de la conception.

Le choix des outils dépend d'un développeur à un autre selon ses propres connaissances et son package d'informations qu'il maîtrise. Alors cette application peut être manipulée sur une diversité de technologies.

Après avoir essayé d'expliquer en détails le déroulement et le fonctionnement de cette application ainsi d'expliquer le rôle de chaque objet tel que bouton et champ de saisie, On mentionne que cette application peut être modélisée de plusieurs manières selon la vision de chacun. Or, dans notre cas, on a essayé le maximum possible de répondre à toutes les exigences citées dans le cahier de charges.

Comme clôture, on insiste sur le fait que cette application reste une application ouverte à toute mise à jour et qui peut être développée non seulement au niveau de fonctionnalités et de développement mais aussi au niveau de design et côté graphique.

7 Conclusion générale

Ce projet portant sur l'agent intelligent a été l'occasion pour nous d'apprendre et d'acquérir un ensemble de techniques et de connaissance dans le domaine de l'intelligence artificielle et l'apprentissage profond et aussi une grande partie de finance.

Ce projet se situe en effet, dans le cadre du projet de fin de la deuxième année du cycle ingénieur en informatique. C'était une véritable expérience de travail en collaboration, qui nous a permis de bien gérer la répartition des tâches et de renforcer l'esprit de partage de connaissance ainsi que la synchronisation de notre travail et surtout le travail à distance .Cela va beaucoup nous aidés pour le projet tutoré à venir mais également pour plus tard lorsque nous serons en entreprise

Bibliographie

- <https://fr.mathworks.com/products/image.html>consulté le 22/03/2017
- https://fr.wikipedia.org/wiki/R%C3%A9seau_de_neurones_artificielsconsulté le 01/04/2017
- https://www.amazon.com/Machine-Learning-Algorithmic-Trading-alternative/dp/1839217715/ref=sr_1_1?dchild=1&keywords=trading+machine+learning&qid=1625084713&sr=8-1
- <https://www.packtpub.com/product/machine-learning-for-algorithmic-trading-bots-with-python-video/9781789951165>
- <https://blog.usejournal.com/a-step-by-step-guide-to-building-a-trading-bot-in-any-programming-language-d202ffe91569>
- <https://towardsdatascience.com/algorithmic-trading-bot-python-ab8f42c37145>
- <https://github.com/PacktPublishing/Machine-Learning-for-Algorithmic-Trading-Bots-with-Python>
-