

UNIVERSITÉ DU QUÉBEC À MONTRÉAL

VÉRIFICATION DE DOCUMENTS PAR RPA :
ÉTUDE DE CAS DANS LE CADRE DE L'INDUSTRIE BANCAIRE.

RAPPORT D'ACTIVITÉ DE SYNTHÈSE
PRÉSENTÉ COMME EXIGENCE PARTIELLE
DE LA MAÎTRISE EN INFORMATIQUE DE GESTION

PAR
OUSSAMA CHOUK

AVRIL 2022

REMERCIEMENTS

Je voudrais remercier toutes les personnes qui m'ont supporté pour réaliser ce projet.

Je voudrais remercier ma directrice de recherche, Mme Sylvie Trudel, d'avoir cru dans cette recherche et pour son appui continu. Son expertise dans le domaine m'a permis de poser les bonnes questions et d'analyser les problèmes différemment.

Je tiens à remercier mon codirecteur de recherche, Mr Abderrahmane Lashhob. Ses perspectives de la technologie étaient un enjeu majeur pour l'accomplissement du projet.

Je remercie mes collègues qui ont été curieux de voir ce projet aboutir et qui m'ont supporté avec leurs calculs et rétroactions.

Finalement, je remercie ma famille qui m'a toujours soutenu dans l'atteinte de mes buts dans la vie.

DÉDICACE

*Je dédie cette recherche à ma femme et à mes enfants
qui ont été ma source continue d'encouragement et de motivation
pour surmonter tous les défis.*

TABLE DES MATIÈRES

Remerciements	iii
Dédicace	v
Liste des figures	ix
Liste des tableaux	xi
Listings	xiii
Liste des abréviations, sigles et acronymes	xv
INTRODUCTION	1
1. PROBLÉMATIQUE ET OBJECTIF DE RECHERCHE	3
1.1. Mise en contexte	3
1.2. Présentation de la problématique	4
1.3. Questions de recherche	7
1.4. Objectifs de recherche	7
2. REVUE DE LITTÉRATURE	9
2.1. La transformation numérique	9
2.1.1. Vitesse	9
2.1.2. Ampleur	10
2.1.3. Complexité de l'impact sociétal	10
2.2. L'automatisation robotisée des processus	11
2.3. L'automatisation intelligente des processus ou RPA 2.0	14
2.4. Contribution de la recherche et conclusion	15
3. CADRE MÉTHODOLOGIQUE DE RECHERCHE	17
3.1. Étapes de notre démarche	18
3.1.1. Planification de notre recherche	19
3.1.2. Raffinement méthodologique	21
3.1.3. Conception de la solution RPA	21
3.1.4. Communication de la recherche	22
3.2. Sources de connaissances	22
3.3. Conclusion	23

4. SOLUTION PROPOSÉE	25
4.1. Simulation des données bancaires	26
4.2. Le choix de l'outil et le développement de la solution automatisée	28
4.2.1. Chercher les documents relatifs à un dossier	31
4.2.2. Compréhension intelligente des documents	32
4.2.3. Chercher les informations de l'API	33
4.2.4. La Comparaison des données	34
5. RÉSULTATS	37
6. DESCRIPTION DES OBSTACLES RENCONTRÉS ET SOLUTIONS AP- PLIQUÉES	41
6.1. L'environnement de développement et la création de l'API	41
6.2. La lecture des documents de revenus	42
6.3. La comparaison des valeurs	45
7. OBSERVATIONS	47
7.1. L'état actuel de la technologie	47
7.1.1. RPA	47
7.1.2. IPA	48
7.2. L'expérience utilisateur	49
7.3. Pistes d'améliorations	49
CONCLUSION	51
A. Simulation des données bancaires	53
A.1. Création de l'environnement de développement	53
A.2. Utilisation de Django REST framework.	53
A.2.1. Modèle de données	54
A.2.2. Exemple de <i>Serializer</i>	55
B. Développement de la solution avec UIPath	57
B.1. Développement du robot	57
B.2. Connexion à l'API	61
B.3. Téléchargement des documents de revenus	63
B.4. Extraction intelligente des données	66
B.5. Comparaison des valeurs	69
Bibliographie	71

LISTE DES FIGURES

1.1. Le processus d'analyse de dossier de crédit dans la compagnie XYZ . . .	5
2.1. La bande automatisable (Lacity et al., 2015)	13
3.1. Aperçu méthodologique	18
3.2. Sous-processus visé	21
4.1. Sous-processus souhaité	25
4.2. Liste des clients	27
4.3. Organigramme de séquences à exécuter	30
4.4. Chercher les documents relatifs à un dossier	31
4.5. La compréhension intelligente des documents	32
4.6. Exemple de comparaison dans la table de jointure	35
5.1. Gains potentiels avec la robotisation du processus de vérification des documents	39
5.2. Exemple de comparaisons effectuées par le robot	40
B.1. Solution complète Main	58
B.2. Chercher les infos des clients	61
B.3. Chercher les infos des clients	63
B.4. Extraction intelligente des documents 1/2	66
B.5. Extraction intelligente des documents 2/2	67
B.6. Comparaison des valeurs	69

LISTE DES TABLEAUX

3.1.	Temps alloué pour la confirmation des revenus	20
4.1.	Tableau comparatif entre <i>UI Path et Automation Anywhere</i>	29
5.1.	Tableau des résultats de la recherche	38
6.1.	Tableau de contraintes du développement de l'API	42
6.2.	Tableau de contraintes de UiPath <i>Document Understanding</i>	44

LISTINGS

6.1. Exemple de code en VB pour traiter les données	45
A.1. Création des <i>data models</i>	54
A.2. Création des <i>serializers</i>	55
B.1. Exemple de code en VB pour comparer le salaire	70

LISTE DES ABRÉVIATIONS, SIGLES ET ACRONYMES

ADC	Avis de cotisation
DSR	<i>Design science research</i>
IA	Intelligence artificielle
IPA	<i>Intelligent process automation</i>
MIG	Maîtrise en informatique de gestion
NAS	Numéro d'assurance social
RDA	<i>Robotic desktop automation</i>
RPA	<i>Robotic process automation</i>
ROC	Reconnaissance Optique de Caractères
SI	Systèmes d'information
UQAM	Université du Québec à Montréal

INTRODUCTION

La transformation numérique pousse les entreprises à innover continuellement pour pouvoir s'adapter à un monde hyper-connecté et en constante mutation. Ces innovations touchent essentiellement le modèle commercial, les processus d'affaires et la culture organisationnelle des entreprises (Mignot, 2019). Cette révolution technologique nécessite souvent l'introduction des nouvelles technologies, mais aussi des améliorations logicielles et l'adoption de nouveaux procédés émergents. Ce qui implique des enjeux considérables liés à la gestion de changement (Herberger, Dötsch, 2021).

Les projets de robotisation occupent une place à part au sein des projets de transformation digitale. L'automatisation robotisée des processus¹ est une nouvelle technologie révolutionnaire conçue pour faciliter l'exécution rapide d'un grand nombre de tâches à faible degré de complexité. L'idée principale est de créer et de déployer des robots qui vont imiter des agents humains dans les tâches répétitives avec une précision accrue et des risques d'erreurs quasi nuls (Leshob et al., 2020).

La RPA est aussi une approche à faible coût qui est très attrayante et qui progresse rapidement dans les industries de services (assurances, banques, etc.) (Lacity et al., 2015; Leshob et al., 2018), car elle facilite l'automatisation des processus d'affaires, en opérant sur les logiciels légataires de l'entreprise. (Rizk et al., 2020). Contrairement à l'automatisation *backend*, la RPA exécute les tâches au sein même des interfaces utilisateurs en s'appropriant le processus d'affaires existant (Rizk et al., 2020).

Les cas d'utilisations des robots logicielles sont aussi en plein essor à cause de la maturité de cette technologie. C'est notamment le cas dans le secteur bancaire où l'automatisation robotisée des processus implique des transferts de données entre plusieurs bases de données, différents systèmes et applications. Ces dernières années, l'approbation automatique des demandes de crédit personnel, la surveillance des activités suspectes et la vérification robotisée de l'identité du client sont devenues possibles avec l'usage des RPA (Gogineni, 2021).

1. En anglais Robotic Process Automation. Nous allons utiliser l'acronyme (RPA), car, c'est le plus utilisé dans la littérature.

Ce document est un rapport d'activité de synthèse à la maîtrise en informatique de gestion de l'UQAM, se composant éventuellement de sept chapitres : Nous allons établir dans le premier chapitre le contexte et l'objectif de cette recherche. Le second chapitre fera le survol de la littérature pour mieux éclaircir les théories et les concepts de la robotisation des processus. Dans le troisième chapitre, nous allons justifier l'approche méthodologique de la science de la conception choisie pour cette recherche, ensuite, nous présenterons les étapes que nous estimons nécessaires pour l'atteinte de nos objectifs. Le quatrième chapitre présentera la solution proposée pour relever le défi de cette recherche. Les chapitres cinq et six présenteront consécutivement les résultats obtenus par le robot ainsi que les obstacles rencontrés lors du développement de notre solution. Finalement dans le chapitre sept nous allons étaler nos observations sur les éléments clés de cette approche pour donner nos recommandations.

CHAPITRE 1.

PROBLÉMATIQUE ET OBJECTIF DE RECHERCHE

Dans ce chapitre, on va présenter le contexte général de notre recherche dans la partie 1.1. Ensuite, une description de la problématique de recherche est exposée dans la partie 1.2. Les questions de la recherche sont définies à la partie 1.3. Ce chapitre se termine avec l'objectif de recherche avec la partie 1.4.

1.1. Mise en contexte

Le marché canadien de la dette a des impacts considérables sur la croissance économique, donc il influence directement la politique monétaire du pays. Une étude récente de **statistique Canada** montre que les ménages canadiens doivent 1,76 \$ pour chaque dollar de revenu net pour un total de 2,3 trillions de dollars, dont 65 % sont liés au crédit hypothécaire¹.

Un marché de cette envergure exige une réglementation rigide et en constante évolution notamment après la « récente » crise financière de 2008, pendant laquelle plusieurs questions ont été soulevées, car les prêts hypothécaires américains ont été, à l'été 2007, le catalyseur de la crise financière. En effet, les banques ont octroyé des prêts à des taux raisonnables à des acheteurs n'ayant pas nécessairement le salaire approprié pour payer les taux d'intérêt².

Par conséquent, la bonne compréhension des sources de revenus des demandeurs est primordiale pour l'analyse des demandes de crédit, pour cerner le risque de défaut de paiement et de la solvabilité des clients. Une diligence raisonnable de la part des institutions est requise d'une façon régulière lors de la confirmation des revenus, car

1. www150.statcan.gc.ca/t1/tbl1/en/cv.action?pid=3810023801

2. ici.radio-canada.ca/nouvelle/1123866/crise-financiere-2008-faillite-lehman-brothers-subprimes-bulle-immobiliere-etats-unis-dette-surendettement

cela a des implications importantes, dont la détection de fraudes et la lutte contre le blanchiment d'argent.

Cette constante évolution du marché de la dette exige des institutions financières une constante réflexion d'ordre managérial, produit et opérationnel sur l'innovation. Au cours des dernières années, diverses acquisitions et fusions ont eu lieu, ce qui a entraîné une augmentation de la concurrence dans les services financiers.

L'adoption de l'automatisation robotisée des processus(RPA) se fait de plus en plus sentir dans les opérations bancaires. En effet, les dépenses de RPA dans les services bancaires et financiers étaient estimées à 200 millions de dollars en 2018 et on estime une augmentation à 1,2 milliard de dollars en 2023 (Gregory, 2019).

1.2. Présentation de la problématique

L'analyse des dossiers de crédit hypothécaire est un travail complexe, régi par une réglementation rigoureuse, une concurrence ardente et un besoin incessant d'innovation. Durant cette décennie, on a vu l'adoption de RPA dans plusieurs aspects du service financier. Présentement, la majorité des dossiers soumis par les canaux de ventes est auto traitée par le système. Cela a permis aux banques d'améliorer leurs services aux clients avec une considérable réduction des délais d'attente. Le processus de révisions qui prenait plusieurs jours, mêmes semaines, se fait maintenant approuver instantanément. Les Systèmes d'Information (SI) derrière tout ça, peuvent évaluer le risque de crédit en se basant sur l'information fournie.

Dans le monde des affaires, il ne faut pas faire de compromis sur la vitesse, l'efficience et la qualité. L'automatisation ne peut avoir lieu sans numérisation. En termes simples, la numérisation est la pierre angulaire de notre parcours vers la transformation numérique. Cependant, la confirmation des revenus, à ce jour, est traitée d'une façon manuelle. La révision des documents historiques (avis de cotisations, T4, etc.) et des documents courants (talons de paies, dépôts directs, lettre d'emploi, etc.) demande une très bonne compréhension des lois fiscales et des procédés qui règlent le marché.

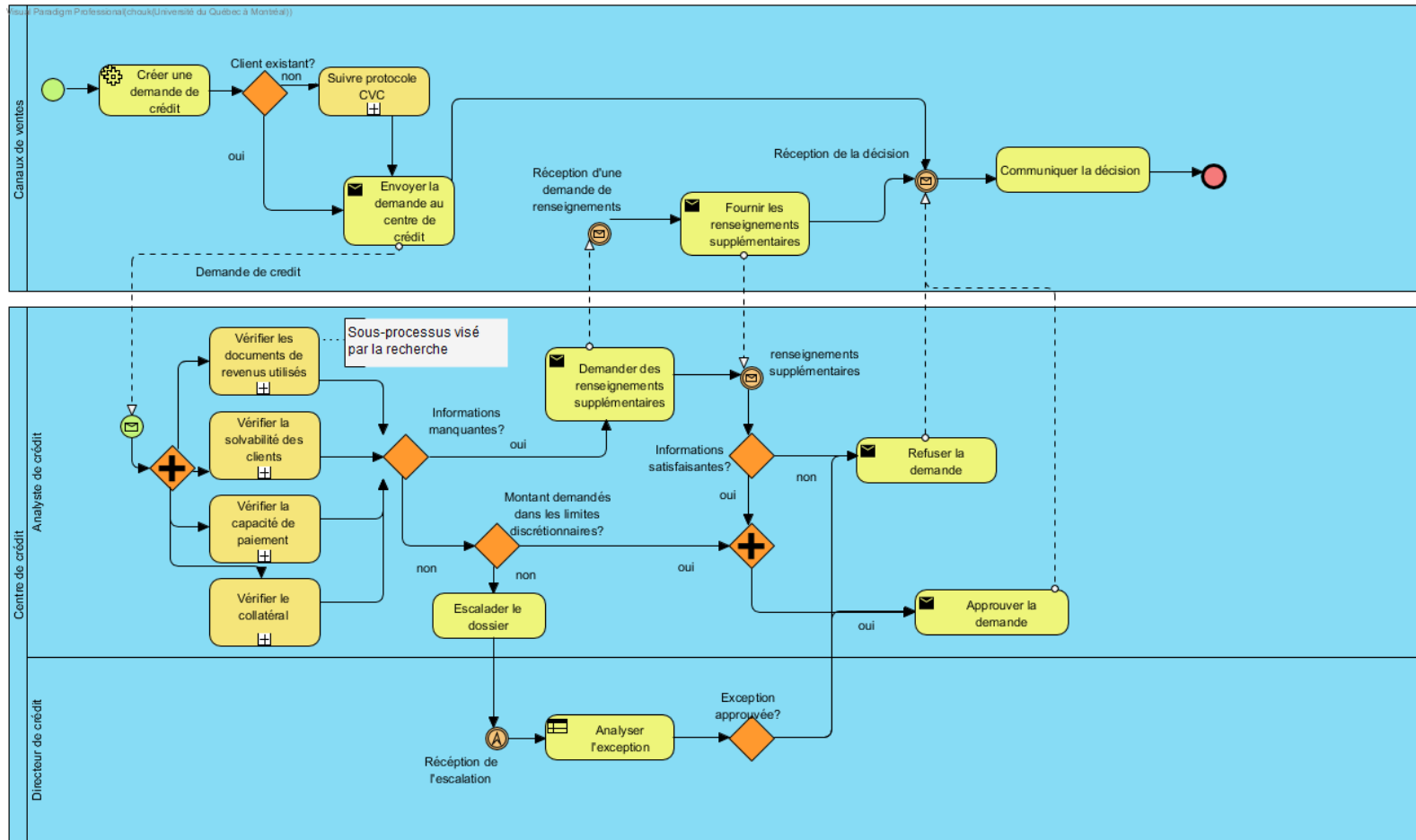


FIGURE 1.1. – Le processus d'analyse de dossier de crédit dans la compagnie XYZ

Le processus BPMN illustré dans la figure 1.1 montre les tâches routinières à effectuer pour s'assurer en premier lieu de l'exactitude des informations (exemples : même numéro d'assurance sociale, même adresse...). L'analyste de crédit doit faire attention aux signes potentiels de fraudes dans ces documents : l'usage de différentes polices de caractères, fautes d'orthographe, dépôts suspects aux guichets, etc.

En deuxième étape, l'analyste de crédit doit s'assurer que les sources de revenus courantes sont vérifiables et s'alignent avec les revenus historiques. Cette tâche va permettre de dresser le profil général du client et d'analyser le caractère raisonnable de la demande. Exemple : S'il existe un écart considérable entre le revenu d'emploi courant et celui des années précédentes, est-ce qu'il existe une justification à cela (exemple : prestations d'invalidité temporaire).

On essaye ainsi de comprendre la situation financière des clients et de s'assurer qu'ils ont une capacité financière suffisante à servir la dette. Ces tâches sont la partie la plus fastidieuse dans l'analyse d'une application de crédit hypothécaire. La demande est généralement soumise par un à quatre codemandeurs avec des types de revenus différents (Salarié, travailleur autonome, retraité, etc.).

Une étude faite en recensant les données opérationnelles chez 5 analystes de crédit à l'institution financière³ a montré qu'en moyenne ça prend 45 minutes pour la révision d'un dossier de crédit et 40 % de ce temps alloué pour la révision des sources de revenus. Ce qui représente :

$$45min * 40\% = \frac{18 \text{ minutes}}{\text{application}}$$

L'automatisation de cette étape de traitement aura un effet domino sur tout le processus. Un analyste fait en moyenne 8 dossiers par jour, aura un gain total de :

$$8 * 18 \text{ minutes} = 144 \text{ minutes ou bien } \frac{144}{45} = 3,2 \text{ dossiers supplémentaires}$$

Ainsi, pour un centre de 60 analystes, ceci représente une capacité supplémentaire de 192 dossiers traités/jour. La confirmation manuelle pose aussi des problèmes de qualité et représente une importante partie des défaillances lors des audits interne et externe. Cela est dû au gros nombre de documents qu'on reçoit (Exemple : les avis de cotisations fédéraux des deux dernières années représentent 10 pages à vérifier). Finalement la confirmation manuelle demande, pour certains, beaucoup d'impressions de documents physiques, ce qui est à l'encontre de la stratégie d'optimisation de flux de

3. Pour des raisons de confidentialité, nous allons omettre de mentionner le nom de cette banque canadienne.

l'information de l'industrie et l'image de marque « 0 papier ».

1.3. Questions de recherche

Pour réaliser ce projet, nous posons la question suivante : Comment intégrer l'approche RPA pour réduire les tâches routinières de confirmation de revenus afin d'améliorer l'efficacité et la qualité de ce processus ? Cette question est indispensable à l'amélioration de l'expérience client. En effet, les clients veulent que leurs demandes soient traitées toujours plus rapidement et exigent un service de plus en plus optimal. En plus, cela améliorera la gestion des tâches routinières. Pour nous aider à développer un artéfact qui s'aligne avec cette question, on va explorer les innovations acquises en « Traitement intelligent des documents » et son application en RPA. Les sous-questions suivantes ont été développées :

- I. Quelles sont les options d'automatisations offertes ?
- II. Comment intégrer les robots intelligents dans le processus courant ?

1.4. Objectifs de recherche

L'objectif de ce travail de recherche est de concevoir et développer un artéfact pour le traitement intelligent des documents qui sera adapté au système bancaire canadien et qui aidera à réinventer le processus de traitement des applications de crédit. Cela relève d'une bonne compréhension de l'approche RPA d'une manière générale et, en particulier, l'application du *machine learning* pour le traitement automatisé des documents de revenus.

Pour atteindre cet objectif, on va commencer par définir un cadre de travail qui favorise l'implémentation de l'artéfact. Une recherche sur les facteurs de succès pour l'implémentation d'un RPA sera ainsi définie. Ensuite, on regardera les options offertes actuellement pour le traitement de documents, où une approche coûts-bénéfices sera dressée. On développera par la suite un artéfact qui sera soutenable à l'industrie bancaire canadienne et qui prendra en charge l'implémentation de l'expertise d'un analyste de crédit pour automatiser les tâches routinières de la confirmation des revenus. La démarche employée est le *design science* selon les trois boucles d'activités : rigueur, pertinence et design (Hevner, 2007).

CHAPITRE 2.

REVUE DE LITTÉRATURE

Dans ce chapitre, on va présenter le concept de la transformation numérique comme garant de survie des entreprises. Ensuite, on va établir les caractéristiques distinctives de cette transformation. Enfin, on abordera une facette de cette transformation que sont les RPA, on établira la liste des avantages et des inconvénients pour finir avec les opportunités futures que l'intégration de l'intelligence artificielle (IA) peut apporter.

2.1. La transformation numérique

La technologie fusionne de plus en plus avec le quotidien des humains. Les frontières s'estompent entre l'ensemble physique, numérique et biologique (Schwab et al., 2014) comme les voitures autonomes, la reconnaissance biométrique ou le diagnostic médical assisté par IA. Ces innovations technologiques prennent de plus en plus d'importance dans nos vies. Schwab, K (2014) considère la transformation numérique comme la quatrième révolution industrielle. Il identifie trois axes (La vitesse, l'ampleur et l'impact sociétal) qui caractérisent cette révolution.

2.1.1. Vitesse

Contrairement aux évolutions linéaires vécues dans le passé, on passe à une vitesse exponentielle. En effet, il a fallu 75 ans pour que 100 millions d'utilisateurs adoptent le téléphone. Par ailleurs, en 15 mois seulement, le jeu Candy Crush a atteint le même nombre d'utilisateurs (Insider, 2015). La transformation numérique est en plein essor et continue de s'accélérer, puisque les statistiques démontrent que le nombre de demandes de brevet incluant l'expression « intelligence artificielle » ou « apprentissage machine » publiées par l'Office américain des brevets et des marques (USPTO) est passé de 406 publications en 2009 à 4 091 publications en 2019 (projection à partir des statistiques

disponibles le 11 septembre 2019) . En d’autres termes, ce n’est pas qu’une mode passagère, mais plutôt une réalité à laquelle les entreprises doivent s’adapter.

2.1.2. Ampleur

Cette révolution industrielle est en train de modifier en profondeur le monde tel qu’on le connaît. Elle modifie le comportement des consommateurs, l’organisation des entreprises et le rôle des collaborateurs (Zaki, 2019). Une nouvelle compréhension des modèles d’affaires est indispensable pour adopter ces nouveaux paradigmes (Kotarba, 2018). Les entreprises adoptent de nouvelles technologies pour rendre leurs activités plus efficaces comme elles ont adopté la machine à vapeur lors de la première révolution industrielle. La survie des compagnies modernes dépend essentiellement des bonnes décisions qui ont été prises (Berthold et al., 2010), cela explique l’émergence des logiciels d’aide à la décision et de l’intelligence d’affaires, ce qui va favoriser l’établissement d’un écosystème de coopération, de coordination et de collaboration (Pappas et al., 2018).

2.1.3. Complexité de l’impact sociétal

Plusieurs technologies imposent une réflexion sur les enjeux éthiques et sociaux de cette révolution. Exemple : la robotisation des processus, l’intelligence artificielle ou l’apprentissage automatique. Parmi ces enjeux les plus importants, on peut citer :

- La vie privée : que ce soit l’industrie alimentaire, les banques ou les épiceries, ces entités se transforment progressivement en entreprises technologiques. Passant au numérique, toutes ces compagnies génèrent un volume considérable de données (Pappas et al., 2018). Les utilisateurs s’interrogent de plus en plus sur l’usage de leurs données privées et si ces compagnies n’en savent pas un peu trop sur eux (Reinartz, 2019).
- Employabilité : de nombreux spécialistes se préoccupent du fait que l’IA pourrait engendrer un chômage de masse (en raison d’une automatisation croissante des emplois), même s’il faut quelque peu tempérer ces craintes. Selon (Frey, Osborne, 2017) 47 % des tâches seront automatisées. Les analystes sont d’accord sur les changements structuraux du marché de travail. La polarisation de l’emploi décrit un changement dans le marché de travail, où les emplois qualifiés sont évidés et une augmentation d’une manière disproportionnée des emplois à faibles ou très hautes qualifications (Hirschi, 2018).

Même si cela semble inquiétant pour certains, plusieurs exemples récents d'entreprises qui investissent massivement dans l'IA et l'automatisation constatent que les innovations, les plus efficaces et les plus rentables sont les résultats d'une collaboration humains-machines¹. Cela démontre l'intérêt et l'émergence des RPA dans cette dernière décennie.

2.2. L'automatisation robotisée des processus

Parmi les multiples facettes de la transformation numérique on va se concentrer sur le RPA car tel qu'était établi précédemment on ne peut parler de transformation numérique sans la numérisation des données, d'où viennent les besoins des RPA (Leopold et al., 2018).

Dans son livre *The Robotic Process Automation Handbook* (Taulli, 2020), Tom Taulli énumère trois versions de RPA, ce qui représente trois grandes évolutions dans l'histoire de cette technologie :

1. RPA supervisé : représente la première génération (2003) de RPA que l'on connaît également sous le nom de *robotic desktop automation* (RDA) et qui est une collaboration entre le logiciel et l'utilisateur pour l'exécution de certaines tâches. Exemple : un RPA qui vérifie les informations du client quand le représentant est en train de prendre l'appel téléphonique.
2. RPA non supervisé : représente la deuxième génération de RPA. Celle-ci peut effectuer des tâches d'une façon autonome. Les robots sont déclenchés avec des événements et leurs exécutions sont indépendantes de toute intervention humaine.
3. RPA intelligent : *Intelligent process automation* (IPA) est la plus récente génération de RPA. Le logiciel utilise sa propre intelligence pour prendre des décisions sur l'exécution des tâches. Exemple : l'apparition de la lecture intelligente des documents.

Le RPA a suscité énormément d'intérêts ces deux dernières années. Cette approche est en train de révolutionner notre milieu de travail. Dans sa forme la plus simple, c'est l'implémentation de robots logiciels pour interagir avec multiples systèmes d'informations d'une entreprise. Avec une série d'instructions, on peut déléguer les tâches fastidieuses et répétitives à un robot et libérer ainsi les ressources humaines (les travailleurs) pour

1. <https://www.businessmodelsinc.com/machines/>

consacrer leurs efforts aux tâches à valeur ajoutée. Exemple : extraire des données, remplir des formulaires ou contrôler des processus (Anagnoste, 2017).

Les RPA sont là pour compléter les systèmes de gestion des processus d'affaires (BPMS, *Business process management system*) et non pour les remplacer affirme le directeur marketing de *Blue prisme* (un leader de solutions RPA) « ... Les humains peuvent être redéployés vers des tâches décisionnelles. » Dans un milieu de travail transactionnel à haut volume, la vitesse, la fiabilité, la cohérence et la précision sont des exigences essentielles. Les RPA permettent aux entreprises de sauver en effectif de remplacements *offshore* (Lacity et al., 2015), d'exécuter les tâches répétitives d'une façon plus efficiente, d'améliorer la qualité du service, de réduire le délai d'exécution et d'augmenter la conformité. Le secteur bancaire, par exemple, est sous pression pour fournir des services ininterrompus et leurs marges bénéficiaires sont en baisse. Alors, la RPA est aujourd'hui la version de l'externalisation technologique idéale. La RPA est rapide et rentable avec un retour sur investissement (ROI) tangible pour les banques (Kumar, Balaramachandran, 2018).

Bien qu'il soit récent, ce type d'automatisation est progressivement utilisé, mais à mesure qu'il se généralise, il y a des défis que les entreprises doivent relever et dont ils doivent tenir compte afin d'automatiser avec succès leurs processus. L'un des défis est l'identification des processus adaptés à l'automatisation de RPA (Leopold et al., 2018). Dans leurs publications *Identifying Candidate Tasks for Robotic Process Automation in Textual Process Descriptions*, les auteurs présentent une approche révolutionnaire pour classer les tâches d'un processus d'affaires. En se basant sur la description textuelle des tâches et de « Machine learning » et le traitement du langage naturel, ils ont pu classer correctement 342/424 des processus.

Lorsqu'elle est implémentée de manière appropriée (c.-à-d. des règles bien définies qui s'alignent avec le processus d'affaires), la RPA présente aussi plusieurs avantages aux collaborateurs : En remplaçant les humains par des robots pour effectuer des travaux répétitifs, les travailleurs peuvent se concentrer sur les tâches les plus importantes qui impliquent de l'analyse critique et la gestion des exceptions, la satisfaction au travail ainsi que la rétention des employés augmentent (Slaby, 2012). Le maintien et le contrôle de ces robots permettent aussi de créer de nouvelles opportunités d'emploi (Asatiani, Penttinen, 2016). L'étude de cas élaborée par Lacity et al. (2015) chez *Telefónica* a présenté un modèle très intéressant (Figure 2.1) pour mettre en contexte les besoins d'automatisation et qui s'aligne avec le modèle établi par Frey, Osborne(2017).

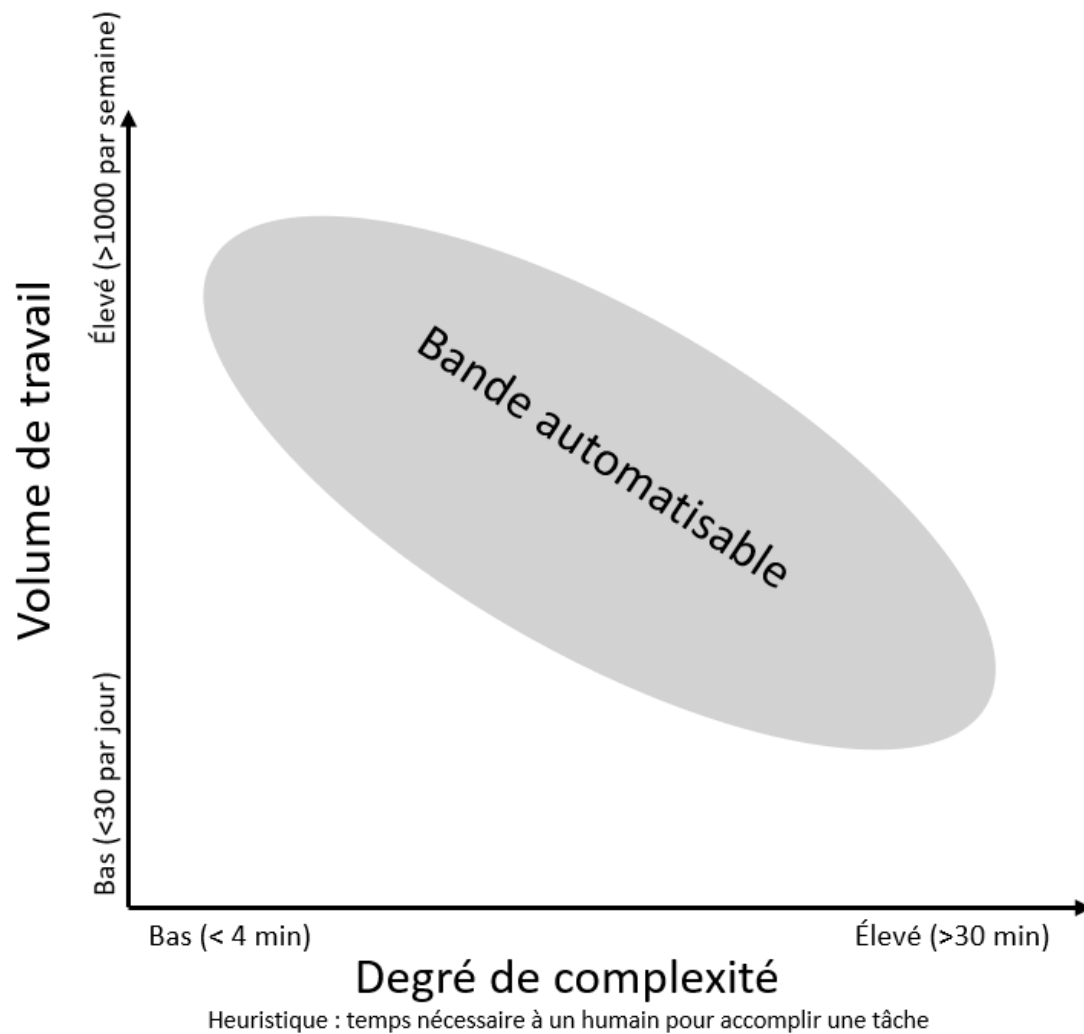


FIGURE 2.1. – La bande automatisable (Lacity et al., 2015)
Interprété de l'étude de cas chez *Telefónica*

L'expérience démontre qu'il est plus approprié d'automatiser les processus routiniers simples à grand volume, plus la complexité des tâches augmente plus ça devient difficile d'implémenter une RPA (Leshob et al., 2020 ; Lacity et al., 2015). L'humain est capable de prendre de petites décisions qui se basent sur le gros bon sens. Dans le cas des automates, il faut explicitement définir les règles de contrôle. En d'autres termes, le manque de compétences cognitives nécessite des règles plus rigoureuses afin d'exécuter ces tâches avec succès. Si le processus contient beaucoup d'exceptions, il doit être remis aux travailleurs (Santos et al., 2020). Pour réussir l'implémentation d'automatisations, une bonne compréhension des règles et du *workflow* est donc indispensable.

La RPA est aussi une solution temporaire pour automatiser les processus basés sur les systèmes légataires. À long terme, il serait peut-être plus approprié de mettre fin aux SI légataires et d'en reconstruire de nouveaux (Asatiani, Penttinen, 2016).

2.3. L'automatisation intelligente des processus ou RPA 2.0

Présentement, il existe plusieurs approches et critères pour déterminer les tâches automatisables, des tâches à très grands volumes avec un faible degré de complexité (Asatiani, Penttinen, 2016 ; Leshob et al., 2018), selon le niveau de maturation du processus, selon les fonctions et l'industrie (Madakam et al., 2019) etc.

Mais la littérature scientifique n'est pas encore décidée quant à l'intégration de RPA avec l'apprentissage machine ou l'IA. L'identité même de ces IPA n'est pas encore claire, car tout simplement, il n'existe pas assez de recherches dans ce domaine. Les IPA peuvent effectuer des automatisations plus complexes en utilisant l'IA pour ajouter cet aspect décisionnel (Chakraborti et al., 2020). Les tâches les plus complexes peuvent être traitées par des machines (Viehhauser, 2020). L'utilisation des capacités cognitives de l'IA sème aussi de la confusion dans les définitions. Il faut comprendre que l'IPA est tout d'abord un robot axé sur l'automatisation des processus auxquels on ajoute une couche d'intelligence sous forme d'apprentissage machine ou traitement naturel de langages (Anagnoste, 2018).

L'automatisation intelligente peut aider à gérer des règles d'affaires plus complexes avec une meilleure flexibilité de surmonter les limitations des RPA (Anagnoste, 2018). On peut extraire et traiter intelligemment du texte, cela permettra une meilleure faculté à gérer un contenu non structuré comme des documents et d'extraire des caractéristiques comme les intentions et les sentiments des utilisateurs (Le Clair et al., 2017). Les opportunités pour l'automatisation intelligente peuvent affecter tous les niveaux de l'entreprise (Anagnoste, 2018).

Dans l'étude de (Viehhauser, 2020), l'auteur explique que même si les robots commencent à obtenir des fonctionnalités « intelligentes », telles que la reconnaissance d'images, les recherches révèlent que la mesure dans laquelle la reconnaissance optique de caractères (ROC) fait partie de la RPA est très limitée et la majorité des fournisseurs de logiciels RPA ne considèrent pas la ROC comme une partie essentielle de la RPA. De plus, aucune des solutions présentes sur le marché n'est capable de capturer des données complexes et non structurées provenant de sources telles que la voix ou le son. Côté traitement, aucun des fournisseurs RPA ne fournit du traitement du langage naturel (NLP)

ou de raisonnement automatisé. Ils sont considérés comme des technologies complexes et non essentielles (Viehhauser, 2020).

Il faut aussi souligné que contrairement aux RPA dont la potentielle réduction de temps de cycle et des économies de coûts va jusqu'à 30 % (Lacity et al., 2015), il y a un coût plus important pour le développement d'IPA, car ça nécessite une préparation assidue de données (identification des données pertinentes, nettoyage et transformation) et l'ingénierie des fonctionnalités (extraction des fonctionnalités appropriées), avant de construire et de valider les capacités cognitives. (Bosco et al., 2019)

2.4. Contribution de la recherche et conclusion

L'automatisation intelligente est un domaine qui est très peu exploité, mais les preuves de concepts récentes prouvent un futur impact imminent sur l'automatisation des processus (Anagnoste, 2018). Si on combine la performance des RPA pour gérer des tâches routinières avec la force cognitive des IA, on pourra perfectionner l'utilité de l'automatisation à tous les niveaux des entreprises et on augmentera ainsi sa compétitivité. Cette revue de littérature nous encourage à améliorer nos recherches dans ce domaine. Car dans le contexte bancaire québécois, il est nécessaire de développer un plan d'implémentation infaillible pour susciter l'intérêt d'adoption.

CHAPITRE 3.

CADRE MÉTHODOLOGIQUE DE RECHERCHE

Dans ce chapitre, nous allons présenter le cadre méthodologique de l'ensemble de la démarche. On se basera sur le travail de (Hevner, 2007) comme source de référence pour atteindre nos objectifs escomptés. Nous allons tout d'abord présenter notre tableau méthodologique pour donner une idée sommaire sur les activités de notre recherche dans le contexte d'une activité de *design science*. On expliquera brièvement le processus actuel de vérification de revenus et les besoins d'automatisations de ce dernier. On dévoilera ensuite notre solution proposée, les obstacles courus, lors de la réalisation de cette solution et les approches prises pour les contourner.

3.1. Étapes de notre démarche

Phases	Intrants	Activités	Extrants
1 - Planification de la recherche	<ul style="list-style-type: none"> • Besoins d'affaires complexes • Théories scientifiques et méthodes 	<ul style="list-style-type: none"> • Établir la problématique. (Ch1) • Effectuer une revue de la littérature (Ch2) 	<ul style="list-style-type: none"> • Liste des objectifs fixés par la recherche.
2 – Raffinement méthodologique	<ul style="list-style-type: none"> • Listes des fournisseurs RPA disponibles sur le marché. • Listes des RPA « logiciels libres » 	<ul style="list-style-type: none"> • Explorer différentes solutions RPA existantes • Tester des solutions. • Établir un système de mesure. • Créer un cadre du travail. 	<ul style="list-style-type: none"> • Liste de solutions potentielles • Outils nécessaires au déploiement de la solution RPA
3a - Conception de la solution RPA	<ul style="list-style-type: none"> • Liste des objectifs fixés par la recherche. • Solution optimale choisie et les outils nécessaires au déploiement de l'artéfact. 	<ul style="list-style-type: none"> • Concevoir des scripts pour le traitement des données non structurées (PDF). • Implémenter la solution RPA choisie • Tester l'efficacité des résultats dans notre contexte. 	<ul style="list-style-type: none"> • Solution pour la vérification des revenus fictifs.
3b - Évaluation de la solution RPA	<ul style="list-style-type: none"> • Solution de Vérification de revenus fictifs 	<ul style="list-style-type: none"> • Évaluer la solution avec des analystes de crédit et analyser la rétroaction. 	<ul style="list-style-type: none"> • Rétroaction des analystes de crédits hypothécaires
4 - Communication de la recherche	<ul style="list-style-type: none"> • Résultats de la recherche 	<ul style="list-style-type: none"> • Rédiger le rapport de l'activité. 	<ul style="list-style-type: none"> • Communiquer les résultats à UQAM et à l'institution financière

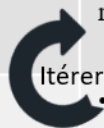


FIGURE 3.1. – Aperçu méthodologique

3.1.1. Planification de notre recherche

Nous avons défini au premier chapitre le contexte et les besoins qui nous poussent à innover et à résoudre des problèmes complexes. On a aussi effectué au chapitre 2 une revue de la littérature qui nous a éclaircis sur fondements des RPA ainsi que l'expertise existante dans ce domaine.

Nous voulons réaliser une preuve de concept pour démontrer l'utilité d'un RPA dans la vérification de la cohérence des déclarations de revenus du demandeur de crédit hypothécaire, à partir des documents historiques fournis. Le but de ce travail est de réduire au maximum les délais de traitements des demandes de crédit tout en assurant une qualité d'analyse hors pair. Ces deux axes sont d'une importance majeure considérant la marge compétitive assez mince du marché hypothécaire. Les délais de traitement influencent directement la satisfaction des clients, la qualité des souscriptions et la tolérance au risque des demandes de crédit.

Afin de quantifier la durée moyenne allouée pour le processus de vérification de l'exactitude des revenus, nous avons demandé un échantillon de cinq dossiers aléatoires par trois analystes.

Nous avons obtenu les résultats décrits dans le tableau 3.1 :

Tableau 3.1. – Temps alloué pour la confirmation des revenus

Applications	Nombre de demandeurs	Temps alloué
1	2	28
2	2	12
3	3	24
4	1	9
5	1	5
6	2	15
7	2	14
8	2	16
9	1	10
10	4	26
11	2	13
12	2	17
Moyenne		15.75

Nous savons déjà que tout le processus de confirmation de revenus en moyenne est de 18 minutes donc on peut conclure que la vérification manuelle des documents représente environ 85 % du sous-processus. Ces données ont été collectées auprès de 3 de nos 60 collègues, le traitement peut grandement varier d'un analyste de crédit à l'autre de par son expérience. Nous espérons réduire ce temps de traitement de moitié dans 90% des cas afin d'obtenir le sous-processus souhaité, illustré à la Figure 3.2

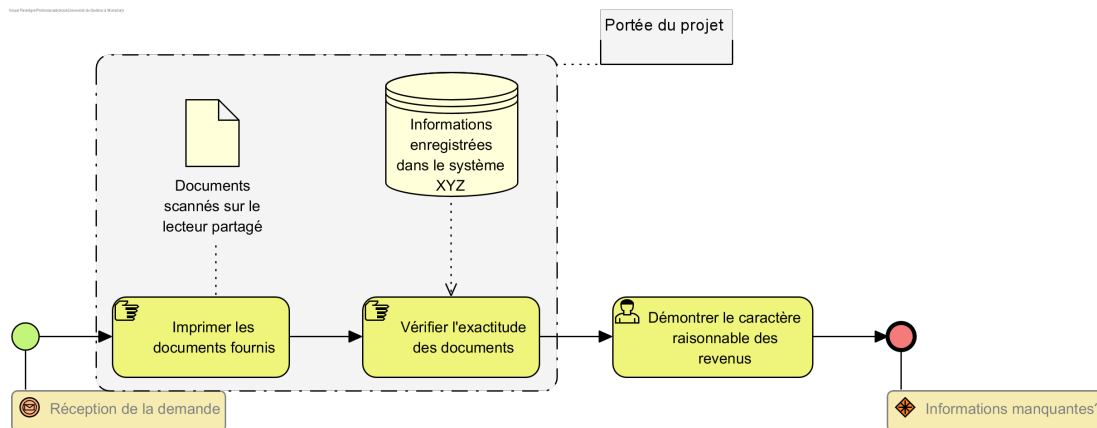


FIGURE 3.2. – Sous-processus visé

3.1.2. Raffinement méthodologique

Les trois plus grands fournisseurs de système RPA sont *Blue Prism*, *Automation Anywhere* et *UiPath*. Il existe aussi des compagnies récentes sur le marché, de sérieux concurrents, comme NICE et Pegasystems, sans oublier les solutions de logiciels libres comme Robot Framework, Robocorp, TagUI etc. Chaque solution vient avec sa liste d'avantages et inconvénients. Certaines requièrent de bonnes connaissances de langages de programmation, exemple : C#, d'autres requièrent un minimum de code possible. À cette étape, nous allons comparer les solutions RPA existantes pour définir une liste d'avantages et inconvénients, puis nous sélectionnerons les solutions qui conviennent au mieux à nos objectifs de recherches. Nous nous limiterons dans le cadre de cette recherche aux trois premières compagnies du classement de Gartner comme solutions payantes et Robot Framework comme solution de logiciel libre. Considérant les enjeux de sécurité importants pour les documents de revenus personnels des clients, nous allons créer un cadre de développement virtuel *sandbox*. Le but est de créer un répertoire infonuagique dans lequel on va stocker des preuves de revenus fictives. Enfin nous allons concevoir une application web qui va simuler le point d'accès aux profils des clients. Ces informations sont les données collectées lors de l'ouverture d'un compte bancaire par les canaux de ventes.

3.1.3. Conception de la solution RPA

Nous allons utiliser la liste de solutions potentielles et outils nécessaires établie dans la phase de raffinement méthodologique comme un point de dé-

part du projet. On va commencer par développer des scripts qui peuvent déterminer quels types de documents sont fournis par le client (Avis de cotisation ou T4), car chacun présente des paramètres différents à considérer. Ensuite, nous allons extraire l'information nécessaire aux traitements (Nom, prénom, numéro d'assurance sociale, salaire annuel, etc.). Ces scripts vont nous permettre de paramétrer la solution RPA pour automatiser l'analyse de ces documents afin de détecter s'il y a un écart entre le profil existant et les informations qui se trouvent sur les documents sous forme d'un rapport. Nous allons présenter les résultats à un comité qui se compose de trois analystes hypothécaires pour solliciter une rétroaction sur l'exactitude de la solution. La démonstration sera effectuée dans le même environnement *sandbox* avec des revenus fictifs. Cette rétroaction d'experts est d'une grande importance pour nous permettre d'améliorer notre solution.

Nous allons itérer jusqu'à l'obtention d'un résultat majoritairement favorable. Dans chaque itération, on va raffiner les données (fichiers PDF pour les avis de cotisations et les T4) pour les adapter aux besoins de notre Solution et améliorer les scripts afin de générer un rapport de vérification automatisé pertinent.

3.1.4. Communication de la recherche

Pour conclure cette activité, nous allons communiquer les résultats au cadre de gestionnaires de l'institution financière en question sous forme de recommandations sur les solutions RPA existantes. Ce projet s'aligne avec la stratégie d'amélioration continue des processus d'affaires de la banque. Les résultats seront également relatés dans un rapport d'activité qui sera fourni à un jury de l'UQAM dans le cadre de soutenance pour accéder au diplôme de maîtrise.

3.2. Sources de connaissances

Pour réaliser ce projet, j'ai dû faire plusieurs tests qui ont abouti à d'innombrables échecs et qui m'ont permis de me développer tout au long du parcours. L'expertise qui m'a été nécessaire pour finir ce projet a été acquise en utilisant plusieurs sources de connaissances. La littérature regorge d'études de cas et d'articles scientifiques sur l'état de l'art des RPA, mais dans cette partie, je voulais recommander quelques sources supplémentaires que j'ai utilisées en mettant l'accent sur les livres, les tutoriels et les blogues qui ont été les plus pertinents pour mon avancement dans ce projet.

Tout d'abord, (Taulli, 2020) m'a permis d'avoir une vue d'ensemble de l'état actuel de cette technologie. C'était une source très riche en connaissances pour m'aider à planifier et à structurer ce projet. Ensuite, lors du développement de l'API, la documentation de Django REST framework¹ était extrêmement utile et complète pour développer l'environnement virtuel des données bancaires.

Pour le développement de la solution RPA, le site web de UiPath était ma première source d'information. UiPath offre aussi des formations gratuites² assez complètes et pour tous les niveaux. Enfin, toutes ces sources de connaissances n'étaient pas infaillibles et sans obstacle. Plusieurs difficultés ont été rencontrées lors du développement de la solution, surtout des problèmes liés à des mises à jour des modules et des problèmes d'incompatibilité.

3.3. Conclusion

Ce chapitre avait pour but la présentation du cadre méthodologique de la recherche basée sur le *design science research*(DSR). Cette démarche est idéale pour la conception et le développement d'une solution nouvelle au problème qu'on a défini au premier chapitre.

1. <https://www.django-rest-framework.org/tutorial/quickstart/>

2. <https://academy.uipath.com/activity-dashboard>

CHAPITRE 4.

SOLUTION PROPOSÉE

Dans ce chapitre, nous allons présenter la solution proposée afin de répondre à nos objectifs de recherche. En premier lieu, nous allons expliquer les contraintes technologiques qu'on a dû contourner pour la continuité du projet. Ensuite, nous allons dévoiler nos critères de sélection de la solution RPA. Finalement, nous allons présenter les séquences nécessaires pour appliquer l'approche RPA à notre preuve de concept. Nous avons déjà ciblé le sous-processus à automatiser dans notre phase de planification. Nous estimons que nous pouvons éliminer une grande partie de l'intervention manuelle dans la vérification des documents de revenus, pour obtenir le sous-processus souhaité suivant :

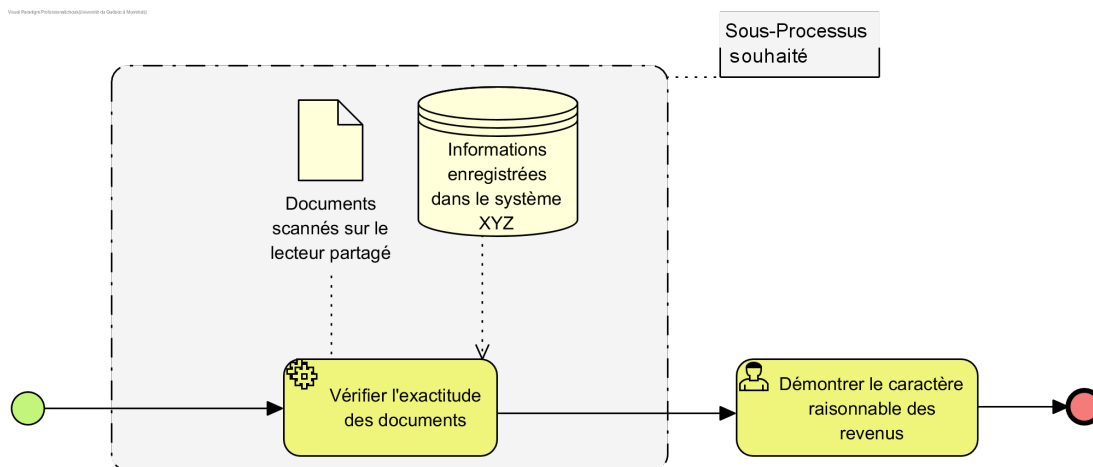


FIGURE 4.1. – Sous-processus souhaité

Nous savons déjà que l'un des avantages majeurs des RPA c'est la possibilité de déploiement des robots sans impact sur l'infrastructure technologique des institutions. Cela représente le premier grand défi de ce projet. Ce défi consiste à développer des robots dans un environnement complètement isolé des systèmes TI d'une banque, mais lors du déploiement nous allons n'avoir aucun

problème à les intégrer et à les faire fonctionner. En d'autres termes comment peut-on tester les robots sans avoir accès aux systèmes d'informations bancaires ?

Pour contourner ce défi, nous avons développé une (Interface Programmable) API qui va fournir les données bancaires nécessaires pour nos tests. Avec cette API, nous allons créer un environnement virtuel où les robots peuvent agir et effectuer les tâches routinières d'analyse des preuves de revenus.

4.1. Simulation des données bancaires

Toutes les solutions testées lors de la phase de planification offraient des modules ou des connecteurs à des bases de données. Cependant, dans un scénario réel, un analyste de crédit interroge les informations d'un client par l'intermédiaire d'une application web. Donc pour simuler cet environnement, nous avons développé une API. Il faut souligner que le but initial de cette démarche est de créer un environnement qui nous permettra de tester les robots. Lors du déploiement, ces derniers vont devoir interagir avec l'application web actuelle de l'entreprise. Selon les besoins d'affaires, une application de crédit peut avoir jusqu'à 4 clients : un demandeur principal, un conjoint et deux codemandeurs pour supporter la demande.

Considérant nos connaissances académiques du langage Python, mais aussi les modules clés en main reconnus de Django, ce dernier fut notre outil de préférence pour cette étape. Django utilise une architecture MVC (Modèle-vue-contrôleur). Voir annexe A.1. Django REST Framework est un module supplémentaire à installer et qui nous permettra de chercher facilement les données et les transformer sous format JSON *Serializer* (Annexe A.2). Voir le résultat final de la vue de la liste des clients obtenu illustré dans la figure 4.2

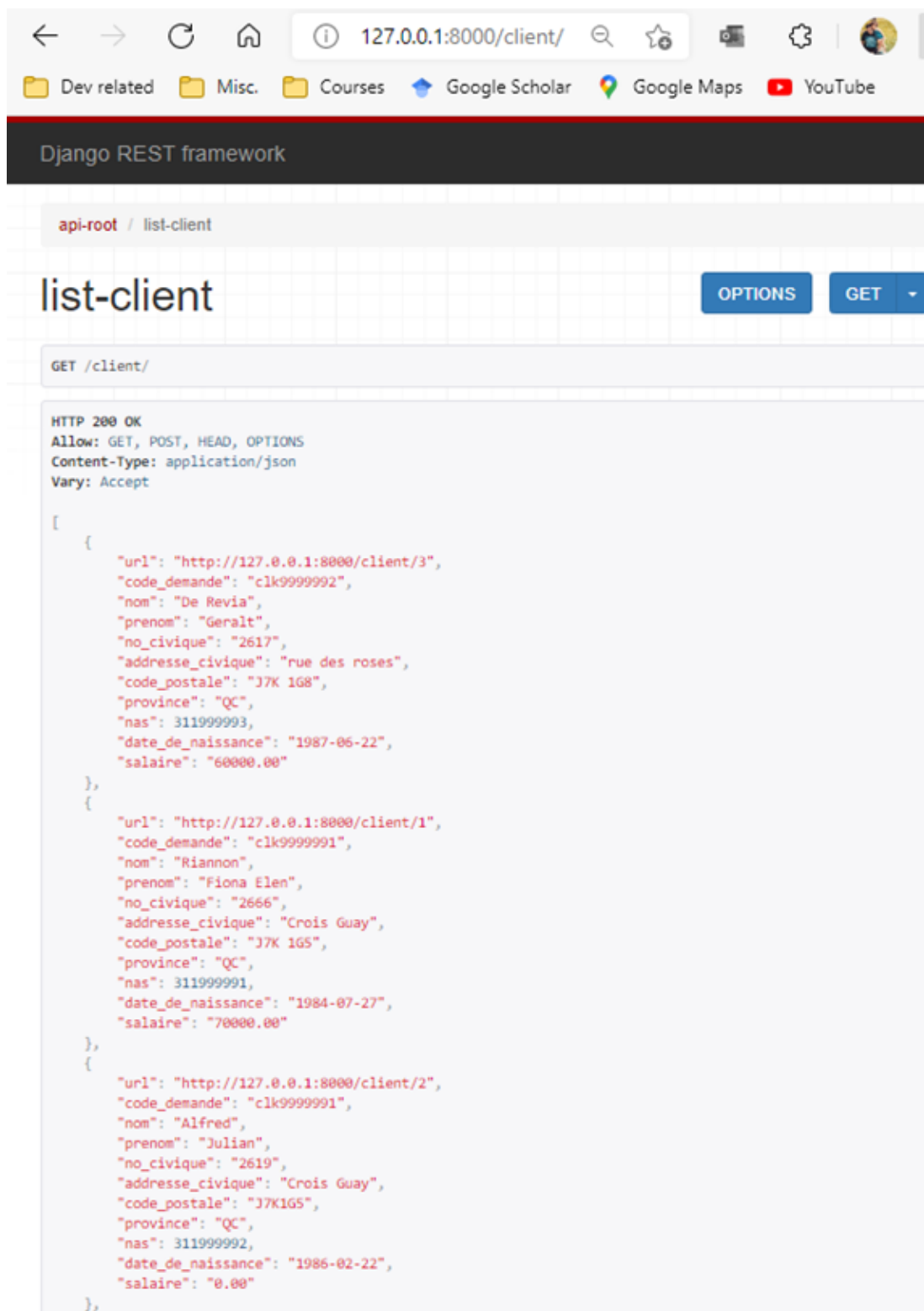


FIGURE 4.2. – Liste des clients

4.2. Le choix de l'outil et le développement de la solution automatisée

Comme mentionné au début du raffinement méthodologique, plusieurs solutions RPA sont offertes et la concurrence ne fait qu'accroître. À la suite des tests préliminaires effectués pour déterminer les outils nécessaires à la création de notre solution, nous avons choisi de démarrer notre projet avec *Automation Anywhere*, car premièrement c'est un leader mondial de cette industrie (Gartner, 2021), deuxièmement pour sa plateforme infonuagique qui requiert un minimum de configuration et un maximum de flexibilité avec sa version communautaire.

Cette solution semblait être parfaite au tout début, l'utilisation de la plateforme est très intuitive jusqu'à la lecture des fichiers PDF. Pour les fichiers PDF de format image (cela représente un bon nombre des documents fournis par les clients), plusieurs difficultés se manifestaient avec la reconnaissance optique de caractères (ROC). Nous avons essayé de créer un script Python qui importait la bibliothèque Tesseract pour la lecture des documents scannés et comme une solution de contournement.

Le résultat obtenu ne semble pas répondre aux attentes escomptées, surtout quand le texte est imprimé au-dessus d'un arrière-plan coloré. Le IQ-bot est un nouveau module chez *Automation Anywhere* et la seule solution infonuagique qui combine les capacités RPA et le *machine learning* pour la lecture intelligente des documents, ce qui représentait une meilleure alternative, mais cette version ne répondait pas à nos besoins de classification des documents. Le tableau 4.1 est un sommaire comparatif entre les deux solutions choisis :

Tableau 4.1. – Tableau comparatif entre *UI Path* et *Automation Anywhere*

Critères	UI Path	Automation Anywhere
Version	Version d’essai, valide gratuitement pour 60 jours	Community Edition gratuite pour les étudiants. Aucune limite de temps
Architecture de développement	Locale avec UiPath Studio	Infonuagique
Modules nécessaires	Logique conditionnelle Services Web REST PDF Document Understanding Script Csharp ou VB.net Service Fichier	Logique conditionnelle Services Web REST PDF IQ-Bot Script python Service Fichier

UIPath offrait un module de lecture intelligente de fichier qui nous permettra de contourner cette lacune chez *Automation Anywhere*. Pour cette raison nous avons décidé de continuer le projet avec *UIPath*. Cette solution permet de découper un projet complexe en petits projets indépendants qui s’appellent *Sequence*. Nous pouvons ainsi construire, tester et valider ces petits projets avant de les réintégrer dans le flux principal *main*.

Au début de notre approche, nous avons structuré la problématique comme une suite de séquences d'activités (voir la Figure 4.3).

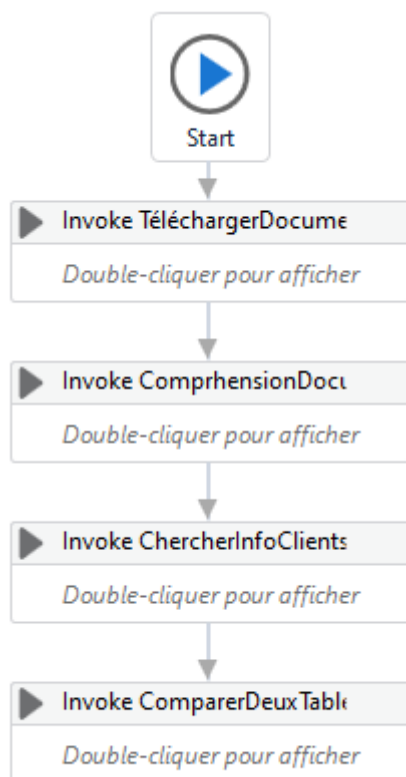


FIGURE 4.3. – Organigramme de séquences à exécuter

Pour résumer, la solution va tout d'abord télécharger les documents de revenus reliés à une application de crédit donnée. Ensuite, elle va faire appel à la compréhension intelligente des documents pour pouvoir distinguer entre les différents types (avis de cotisations et T4). Cela nous permettra d'extraire certaines informations ciblées dans ces documents et les stocker dans un tableau temporaire. Par la suite, nous allons chercher les informations des clients de l'API avec le numéro de dossier. La RPA devra sauvegarder les informations dans un deuxième tableau temporaire. Enfin, l'automate va faire la comparaison de l'information entre ces deux tableaux pour construire un petit rapport de résultats.

4.2.1. Chercher les documents relatifs à un dossier

Premièrement, il fallait chercher les documents fournis par les clients pour les préparer à la lecture. Dans un scénario réel, un analyste accède au répertoire infonuagique où les documents sont stockés par un numéro d'application unique. Nous avons ainsi créé un répertoire infonuagique où on a organisé des documents fictifs d'une façon similaire pour l'extraction. On a créé une séquence spécifique pour aller chercher tous les documents disponibles dans un répertoire avec un numéro de dossier donné. On a dû importer le module **UiPath.MicrosoftOffice365.Activities** pour pouvoir manipuler le service OneDrive. Nous demandons à l'utilisateur d'entrer manuellement le numéro de dossier et la séquence télécharge tous les documents de cette demande de crédit dans un répertoire local (voir la Figure 4.4).

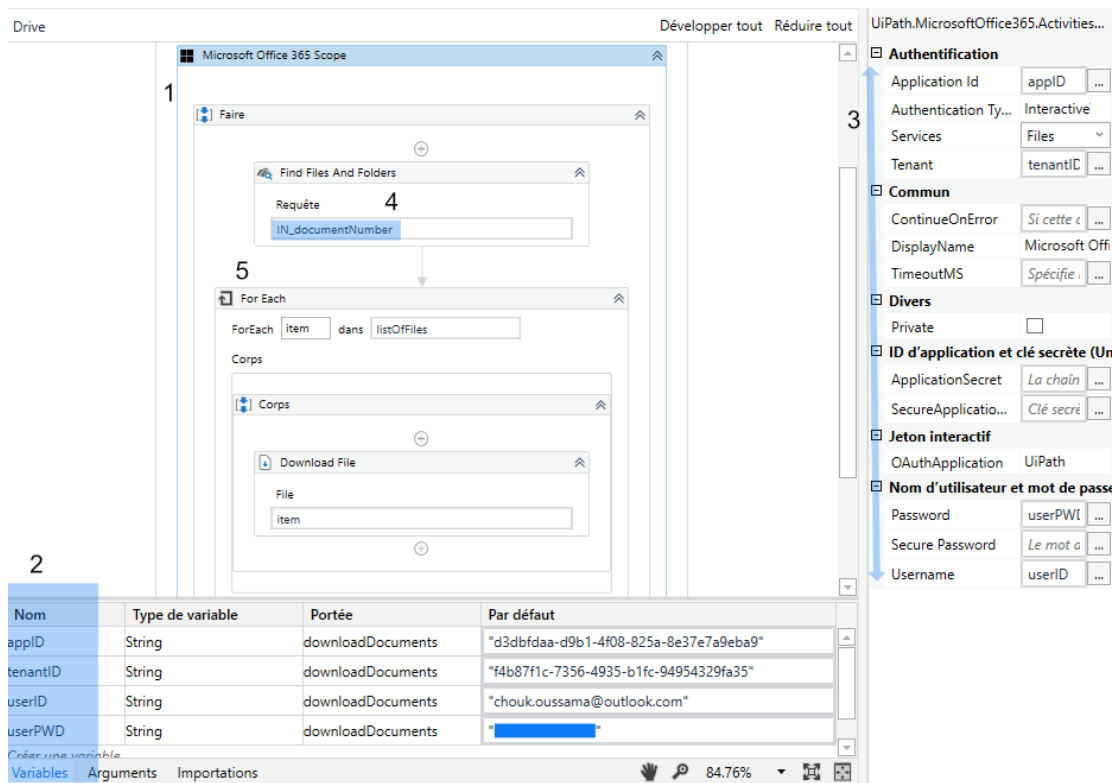


FIGURE 4.4. – Chercher les documents relatifs à un dossier

1. Nous allons travailler dans un *scope* Microsoft Office 365. Cela permettra le dialogue avec la plateforme OneDrive en toute sécurité.
2. Nous avons créé des variables de connexion pour accéder aux services tenantID et appID qui ont été configurés dans *Azure active directory*.

3. Nous avons utilisé ces variables pour paramétrer le *scope*.
4. Nous avons accès aux activités fournies par ce module dont *Find files and folders*. Nous avons utilisé un argument qui représente le numéro de dossier (il va être fourni dans la séquence *main*).
5. Si la recherche est un succès, nous allons télécharger chaque document dans ce répertoire.

4.2.2. Compréhension intelligente des documents

Le module **UiPath.DocumentUnderstanding.ML.Activities** associe RPA et IA pour traiter automatiquement les documents de plusieurs formats. Il permet un traitement intelligent des documents non structurés dans les flux de processus métier, permettant ainsi l'automatisation de processus complexes et cognitifs qui sont jusque-là très manuels. Ce module qui a été introduit à *UiPath* en mai 2020, représente une grande avancée des IPA. C'est même une véritable opportunité pour la bonne continuité de ce projet, car les tests préliminaires effectués avec des ROC traditionnels n'étaient pas concluants et présentaient plusieurs limitations.

Le besoin pour la lecture intelligente des documents dans notre recherche, découle de la multitude de documents fournis par les clients. Dans cette étape, nous voulons lire des documents historiques des revenus. Pour ce faire, nous avons besoin de les classer automatiquement en différentes catégories. Les données relatives aux clients sont répandues différemment entre une page de T4 ou cinq pages d'un avis de cotisation.

Pour atteindre cet objectif assez complexe, *UiPath* a développé ce module avec 6 composants majeurs.

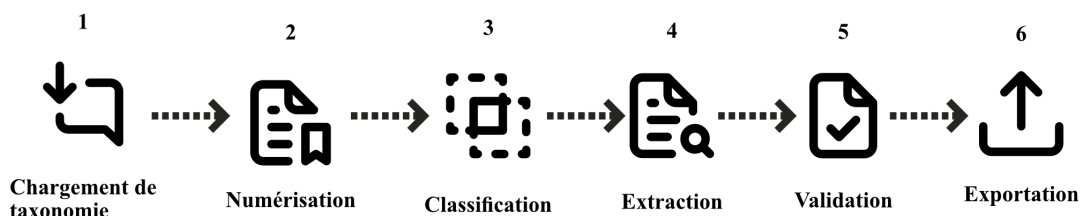


FIGURE 4.5. – La compréhension intelligente des documents

1. Chargement de taxonomie : après l'implémentation du module, l'onglet **Gestionnaire de taxonomie** apparaît dans la barre de menu et cela permet de définir le type de documents et de données vont être traités.

2. Numérisation des documents : cela permet d'utiliser la reconnaissance optique des caractères pour numériser le texte. *UiPath* donne le choix d'utiliser plusieurs fournisseurs de ROC. Dans le cadre de cette recherche et au moment du développement, nous avons constaté que **OmniPage OCR** semble être le meilleur choix pour la lecture d'un texte français.
3. Classification des documents : On utilise ici la taxonomie déjà définie pour classer les documents selon nos besoins. *UiPath* offre ici plusieurs types de classificateurs. Pour ce projet, nous avons choisi d'utiliser un classificateur basé sur des mots-clés, car nous voulons lire des T4 et des avis de cotisations qui ont une structure statique. Mais il faut savoir qu'il est possible d'utiliser un classificateur avec des mots-clés intelligents ou un classificateur basé sur l'apprentissage machine.
4. Extraction des données : pour l'extraction des données, nous avons aussi utilisé la taxonomie déjà définie dans la première étape. Pour les fins de ce projet, on a utilisé un extracteur intelligent de formulaire qui va utiliser un modèle déjà paramétré. C'est une API offerte par *UiPath* qui permet de chercher des mots dans un formulaire spécifique selon leur emplacement.
5. La validation des données : cette étape est optionnelle ; elle permet à l'utilisateur de valider les mots extraits ou de pouvoir les modifier selon le besoin.
6. L'exportation des données : cette étape permet d'exporter les données qui ont été déjà validées dans un format de jeux de données.

Nous allons par la suite enregistrer le jeu de données dans un tableau pour l'exporter et le comparer avec les données des clients qui ont été extraites de l'API.

4.2.3. Chercher les informations de l'API

Pour atteindre cet objectif, il fallait importer le module **WebAPI** dans notre projet. Ce module nous permettait de faire des demandes *HTTP Request* pour un point d'entrée de données. Le point d'entrée se composait d'un lien URL avec une variable qui nous permettait d'accéder aux documents d'un dossier donné. La réponse étant une chaîne JSON, nous avons dû la convertir en objet JSON pour pouvoir la manipuler et extraire les liens vers les clients relatifs à ce dossier.

Ensuite, nous allons refaire les mêmes étapes décrites pour chercher les informations relatives aux clients. Car, pour chaque dossier de crédit, le nombre de clients est variable. Quand les objets JSON relatifs aux clients sont récupérés, nous allons assigner chaque

valeur récupérée à une cellule spécifique d'un tableau de données. Cette séquence retourne ce tableau à la séquence principale comme argument pour la suite du processus (Voir annexe pour le détail de la séquence).

4.2.4. La Comparaison des données

À ce jour, nous nous sommes retrouvés avec deux tableaux récupérés des séquences précédentes (celui de l'API et l'autre des documents). Pour réussir cette séquence d'activité, nous avons tout d'abord pensé à utiliser un fichier Excel pour enregistrer les 2 tableaux et ensuite comparer les valeurs l'une après l'autre. Mais cette approche ne s'aligne pas avec notre objectif final, car nous ne voulons pas avoir des fichiers intermédiaires ou des logiciels supplémentaires afin de réussir ce projet. Il fallait donc trouver une meilleure solution qui utilise les variables *System.Data.DataTable* fournies par UiPath. Or cette piste de solution venait avec un autre degré de complexité. Comment s'assurer que l'appliquant principal enregistré dans l'API, par exemple, est bel et bien l'appliquant principal sur les documents. Autrement dit, nous voulons nous assurer de comparer les bons documents aux bonnes données bancaires. Nous pouvons utiliser le numéro d'assurance sociale (NAS) comme clé de recherche sur les deux tableaux. Mais pour simplifier la solution RPA, nous avons créé une table de jointure avec cette clé primaire. Une fois cette table créée, nous avons fait une comparaison littérale nom pour nom, prénom pour prénom, salaire pour salaire, etc (voir la Figure 4.6). Si la comparaison retourne des valeurs différentes, nous enregistrons un message explicatif dans un fichier de résultats.

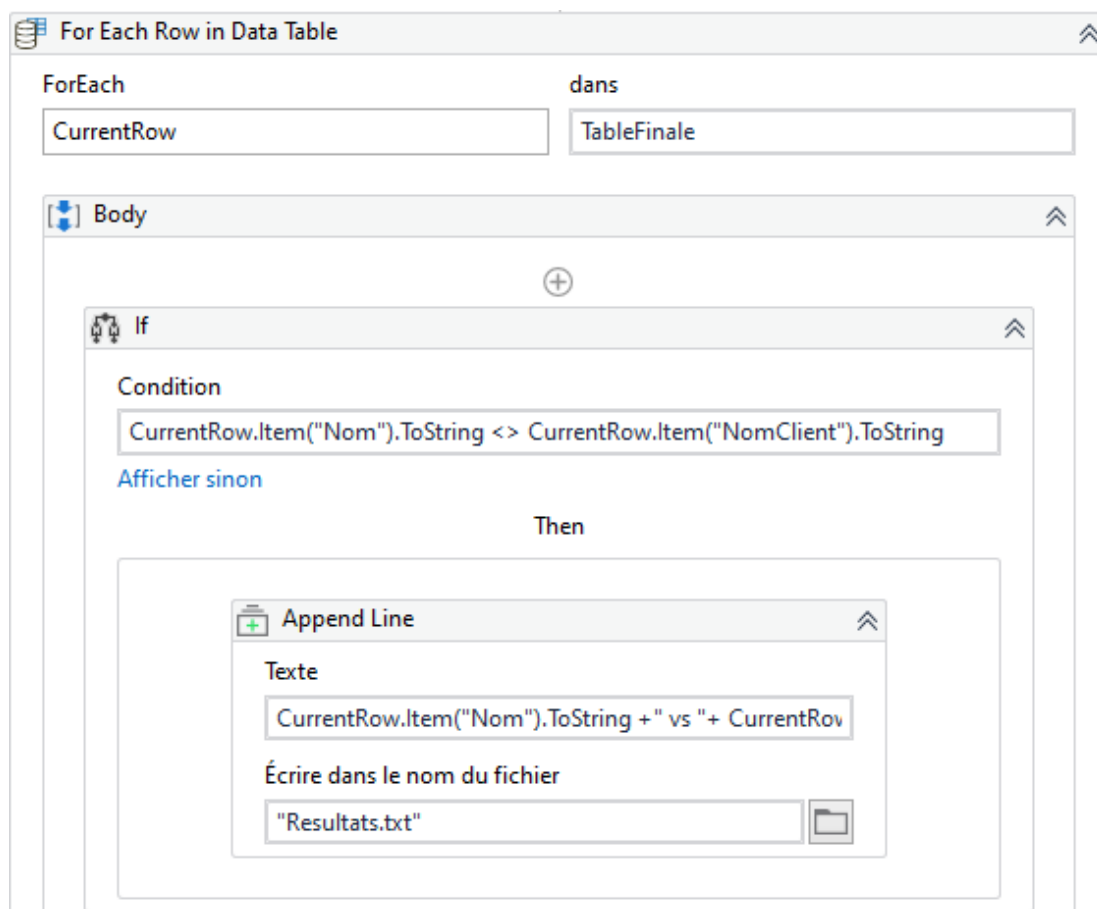


FIGURE 4.6. – Exemple de comparaison dans la table de jointure

CHAPITRE 5.

RÉSULTATS

Dans ce chapitre, nous allons reprendre les exemples déjà mesurés avec les analystes de crédit et nous allons automatiser la vérification de ces exemples pour démontrer la pertinence de cette recherche. Nous ferons des mesures de gains basés sur nos résultats pour illustrer l'importance d'automatiser la vérification des revenus. Nous avons créé dix exemples inspirés des cas déjà calculés par les analystes dans la phase de planification. Nous avons mesuré dans chaque cas le temps d'exécution du robot. Pour chaque client nous avons vérifié six éléments (Nom, Prénom, No Civique, Code postal, Revenu, NAS). Nous avons introduit quelques erreurs dans chaque document.

$$Erreurs\ Detectees = \frac{\sum Valeurs\ Correctement\ Detectees}{\sum Elements\ Verifies}.$$

Le tableau ci-dessous représente les résultats des tests :

Tableau 5.1. – Tableau des résultats de la recherche

Cas	Nombre de clients	Type	Temps d'exécution	Erreurs détectés
1	2	1. T4 2. T4	00 :17s	91.6%
2	2	1. T4 2. T4	00 :15s	100%
3	1	1. ADC	00 :13s	100%
4	1	1. T4	00 :12s	100%
5	4	1. ADC 2. T4 3. T4 4. T4	00 :23s	95.83 %
6	2	1. ADC 2. ADC	00 :16s	100%
7	1	1. ADC	00 :13s	100%
8	3	1. ADC 2. T4 3. T4	00 :19s	100%
9	3	1. ADC 2. ADC 3. ADC	00 :18s	100%
10	4	1. ADC 2. T4 3. ADC 4. ADC	00 :21s	100%

Le premier élément à constater est que la durée de traitement automatisé des documents était entre douze et vingt-trois secondes. Ce qui est négligeable considérant les chiffres fournis par les analystes de crédit. En d'autres termes, l'exécution de ce robot nous permettra de gagner la quasi-totalité du temps alloué au traitement manuel des documents.

Pour illustrer ces gains, nous avons créé le graphique sommaire suivant :

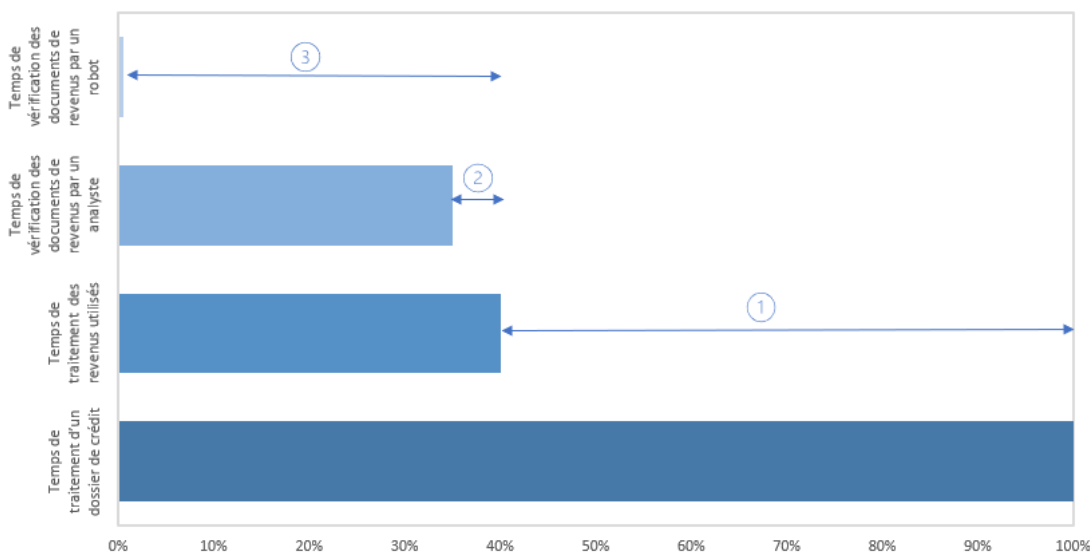


FIGURE 5.1. – Gains potentiels avec la robotisation du processus de vérification des documents

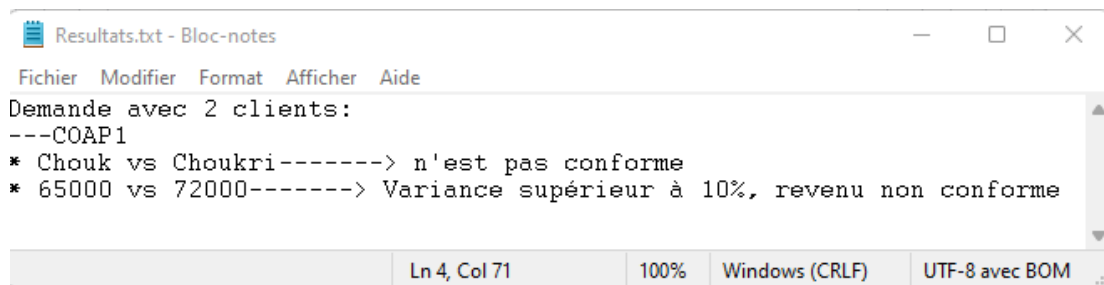
1. Un dossier de crédit prend en moyenne 45 minutes pour être analysé dont 40% (18 minutes) pour l'analyse de revenus et 60% pour toutes les autres étapes.
2. L'analyse des revenus consiste à vérifier les documents fournis (16 minutes) et deux minutes pour l'étude du caractère raisonnable de ces revenus.
3. La moyenne de traitement des robots est de 16 secondes ; en d'autres termes, on a éliminé $16 \text{ m} - 16 \text{ s} = 15 \text{ m} : 44 \text{ s}$ du processus initial :

$$Gains = \frac{Verification \text{ Manuelle}}{Duree \text{ Processus}} = \frac{15 : 44}{45 : 00} = 35\%$$

Le robot fournit à la fin de la séquence un fichier résultats.txt avec les résultats de comparaison. Les résultats peuvent être exportés vers plusieurs formats ou bien peuvent

apparaître dans un message console. Nous avons décidé de les exporter dans un fichier texte pour des raisons d'efficacité.

Les cas 2 et 5 n'ont pas eu un score parfait lors de l'exécution. Cela est dû à une modification dans la taille de police utilisé pour fabriquer les T4. Le module de lecture intelligent ne permettait pas une bonne lecture des chiffres avec une police inférieure à 7 points. Cependant, nous sommes satisfaits des résultats, car le robot a laissé ces valeurs vides. Donc en regardant le document résultats.txt, il était clair qu'une vérification supplémentaire était nécessaire pour pouvoir continuer.



```
Resultats.txt - Bloc-notes
Fichier  Modifier  Format  Afficher  Aide
Demande avec 2 clients:
---COAP1
* Chouk vs Choukri-----> n'est pas conforme
* 65000 vs 72000-----> Variance supérieur à 10%, revenu non conforme
Ln 4, Col 71  100%  Windows (CRLF)  UTF-8 avec BOM
```

FIGURE 5.2. – Exemple de comparaisons effectuées par le robot

CHAPITRE 6.

DESCRIPTION DES OBSTACLES RENCONTRÉS ET SOLUTIONS APPLIQUÉES

Dans ce chapitre, nous allons présenter trois principaux obstacles lors du développement de notre robot ainsi que les solutions appliquées pour les contourner. Premièrement, nous allons présenter les défis liés à l'environnement de développement. Ensuite, nous allons donner une vue d'ensemble sur les problématiques liées à la lecture des documents pour finir avec les obstacles dans la comparaison des valeurs.

6.1. L'environnement de développement et la création de l'API

Le premier obstacle rencontré lors de la création de cette solution est l'environnement du développement lui-même. Ce n'est pas un secret que l'industrie bancaire en général est extrêmement exigeante sur les enjeux de sécurité. Ajoutons à cela les multiples fuites de données qui ont été commises ces dernières années, rendant la tâche de plus en plus difficile. Pour cette raison, nous avons créé un environnement complètement indépendant pour tester les fonctionnalités de notre solution. Ces robots doivent imiter le travail des analystes de crédit sans être liés à l'architecture existante du processus et sans la modifier. Nos tests nous ont permis de délimiter les contraintes et les limites de notre robot et d'imaginer une nouvelle façon de faire les choses pour une meilleure expérience employé.

Le tableau suivant est un sommaire des difficultés rencontrées lors de cette étape ainsi que les solutions fournies :

Tableau 6.1. – Tableau de contraintes du développement de l’API

Problématique	Description	Solution
Compatibilité avec Django REST FRAMEWORK (DRF)	La version Django 3.1 présente des problèmes de compatibilité avec DRF 3.11	Au début, nous avons dû revenir à une version plus ancienne pour résoudre ce problème mais avec la nouvelle mise à jour 3.12.4, cela a été résolu (Voir annexe A.1. pour la liste des modules)
Compatibilité avec le module Psycopg2	La migration vers la base de données génère des problèmes d’incompatibilité dus à l’adaptateur Psycopg2, conçu pour faire le lien entre notre modèle de données et une base POSTGRES	Utilisation de SQLite3 en réseau local, puis utiliser l’adaptateur psycopg2-binary pour utiliser une base de données PostgreSQL en nuage
API présente les données sous format de JARRAY	La conversion de Jarray vers un JObject dans la RPA a présenté un défi supplémentaire pour l’extraction des données.	Utilisation du Hyperlinked-Model comme solution de contournement pour représenter les données de n clients dans une même demande de crédit.

6.2. La lecture des documents de revenus

Nous estimons que le plus grand obstacle rencontré dans cette recherche a été la lecture des documents des clients. Les canaux de ventes de l’institution XYZ envoient les documents récupérés chez les clients, soit par courriels (donc des fichiers PDF standard)

soit scannés (donc format PDF image). Pour le premier format, le module PDF de UiPath était suffisant pour automatiser notre solution. Ce module permettait l'accès aux valeurs nécessaires au traitement de la même façon qu'on accède aux éléments d'une page HTML.

En revanche, il fallait trouver une solution pour les PDF scannés. La solution ROC permettait une bonne reconnaissance des caractères, mais nous avons rencontré plusieurs difficultés liées à la qualité du PDF. En plus, on ne pouvait pas détecter les caractères imprimés sur un fond coloré. Nous avons donc utilisé un script Python pour nous permettre de lire les documents par l'intermédiaire de la bibliothèque Tesseract. Toutefois, le résultat n'était pas encourageant, il y avait plusieurs erreurs dans le texte extrait et techniquement on ne devrait pas changer l'infrastructure où les robots vont être déployés. Nous avons donc commencé à utiliser UiPath *document Understanding* même si la technologie était très récente.

Ce module ajoutait à notre projet une nouvelle couche de complexité. Après plusieurs essais et corrections, nous avons pu avoir des résultats convaincants avec les 2 types de formats PDF. Le présent tableau est un sommaire des difficultés rencontrées lors de cette étape ainsi que les solutions fournies :

Tableau 6.2. – Tableau de contraintes de UiPath *Document Understanding*

Problématique	Description	Solution
<i>UiPath Document OCR</i>	Dû à un problème de versions, la numérisation des documents français n'était pas convaincante	L'utilisation de OmniPage ROC était une meilleure alternative pour nos besoins spécifiques
<i>Intelligent Form Extractor</i> limité à deux pages	L'API de la version communautaire nous permettait de traiter deux pages à la fois	Pour les T4 cela ne posait pas de problème. Nous nous contentons de charger uniquement les deux premières pages des ADC. Toute l'information pertinente est disponible dans ces pages.
<i>Classifier</i> n'arrive plus à détecter le bon type de documents	Nous avons changé certains types de valeurs pour les besoins d'analyse. Cela a causé des problèmes avec la classification	Nous avons dû changer le type de classificateur de <i>Intelligent Keyword Classifier</i> à <i>Keyword Based Classifier</i> et de recréer le fichier classifier.json dans le <i>Classify document scope</i> (Voir annexe B.4)
Module très grand en fonctionnalité avec une documentation restreinte	Nous avons eu plusieurs messages d'erreur que nous ne pouvions pas déboguer et dont la documentation était très large	Nous avons dû adapter des exemples testés sur des chaînes YouTube et les adapter à nos besoins

6.3. La comparaison des valeurs

Cette étape semblait être facile au début. Nous avions déjà fait les plus grandes étapes de notre projet. On a bien récupéré les données de l'API et de nos documents. Nous avons joint ces données dans une table finale en nous basant sur le numéro d'assurance sociale des clients. Nous avons constaté que le tableau sommaire des valeurs n'était pas homogène.

Nous avons aussi dû changer les types des valeurs enregistrées pour pouvoir faire des calculs ensuite. Ceci représente les traitements qui ont été ajoutés dans une phase ultérieure lors de la lecture des documents.

```
System.Text.RegularExpressions.Regex.Match(CurrentRow.Item("adresse").ToString.Replace("
  ↪ ", ""), "[A-Za-z] [0-9] [A-Za-z] [0-9] [A-Za-z] [0-9]").Value
```

Listing 6.1 – Exemple de code en VB pour traiter les données

Plusieurs fois, je me suis retrouvé dans les forums de UiPath et de *Stack Overflow* (principalement pour le code VBA) pour essayer de trouver une solution ou des pistes de solutions, mais plusieurs fois je ne trouvais que des solutions de contournement qui ne répondaient pas spécifiquement à mes besoins.

CHAPITRE 7.

OBSERVATIONS

Dans ce chapitre, nous allons présenter nos observations sur l'état actuel de la technologie. Ensuite, considérant les obstacles rencontrés, nous allons explorer les défis d'adoption de cette technologie. En se basant sur la rétroaction des analystes, nous allons étaler notre point de vue sur la place des RPA dans les entreprises et finir par des pistes d'améliorations dans ce fascinant domaine de la robotisation.

7.1. L'état actuel de la technologie

7.1.1. RPA

Le résultat obtenu avec ce projet est encourageant et démontre une bonne maturité du côté RPA. La littérature scientifique semble aussi être d'accord sur un bon niveau de maturité des RPA classiques pour robotiser les tâches humaines à faible complexité (Enríquez et al., 2020). Nous avons rencontré peu de difficultés à reproduire les tâches simples et répétitives d'un analyste de crédit tels que : accéder au dossier, télécharger les documents et les comparer à l'information fournie par l'API.

UiPath nous permettait de configurer une séquence de tâches en nous basant sur des modules officiels ou de tierces parties (comme le module Microsoft 365). Cela nous permettait d'atteindre nos objectifs efficacement dans chaque itération. Même si le RPA pouvait reproduire aisément les tâches humaines monotones, le développement de ces séquences requiert beaucoup de tests et de gestion d'erreurs. L'effort fourni pour apprendre à déboguer et à gérer ces erreurs est considérable. Nous avons constamment regardé la documentation et les forums afin de trouver une solution.

Tel qu'il était mentionné dans les obstacles rencontrés, nous pensons que la documentation pour UiPath présentait certaines lacunes pour des besoins un peu plus avancés (Exemples : la documentation des manipulations d'une *DataTable*, manipulation des

arguments entre les séquences, etc.). Nous avons constaté que plusieurs solutions discutées dans ces forums sont des solutions de contournement de problèmes plutôt que des réponses à nos besoins spécifiques.

Nous avons constaté avec UiPath que c'est une solution qui n'arrête pas d'évoluer pour répondre à des besoins plus spécifiques. Nous avons vécu plusieurs améliorations et mises à jour dans les modules lors du développement de notre solution. Ce qui est raisonnable, considérant la pleine expansion de cette technologie. Mais une réelle estimation de la complexité des processus, la bonne définition de la portée du processus, les étapes impliquées et les attentes des utilisateurs sont primordiales pour le bon aboutissement aux projets de robotisations.

7.1.2. IPA

Côtés automatisation intelligente du processus, quelques lacunes ont été observées et cela reflète un bas niveau de maturité de cet aspect des IPA. Pour donner un exemple, nous avons voulu déterminer si le type et la taille de police utilisée changent dans les documents scannés pour prédire des probabilités de fraude. Cela peut être atteignable avec beaucoup de difficultés (en ouvrant les documents avec l'extraction de données d'écran), ce qui ne peut pas être considéré comme une solution fiable pour nos besoins. Le module *Document Understanding* ne permet pas à ce jour de réaliser cette tâche. Nous avons aussi testé ce module pour des documents de revenus officiels (T4 et ADC). Le robot permet de classer ces documents en se basant sur des indices bien définis et de chercher à extraire des valeurs bien définies dans un gabarit préalable.

Dans certains cas, un analyste requiert des documents de revenus courants (talons de paie ou lettre d'emploi) la technologie utilisée ne nous permet pas de répondre dans l'immédiat à toutes les variétés de documents qui peuvent être fournis par les clients. À notre connaissance, l'analyse du sens d'un texte reste encore non atteignable avec ce module.

Il est vrai que l'intégration de l'IA avec les RPA classiques nous permettra de traiter ces tâches assez complexes comme comparer les documents fournis à d'innombrables exemples de lettre d'emploi ou des talons de paie pour pouvoir y soustraire automatiquement des informations convaincantes, mais il est certain que l'efficacité de la solution sera affectée. Ce qui nous ramène au débat courant de la littérature, celui de trouver les tâches candidates à l'automatisation est primordial pour atteindre ce gain

en capacité et en coût de processus (Lacity et al., 2015) car l'intégration de l'IA vient avec un coût additionnel à prendre en considération.

7.2. L'expérience utilisateur

La rétroaction des analystes qui ont testé notre solution était majoritairement positive. Les tâches que nous avons ciblées avec le robot représentaient une étape fastidieuse dans le processus, qui n'exigent pas une analyse approfondie du dossier, juste la vérification de l'identité du client et la cohérence des sources de revenus. Les risques d'échecs dans la conformité du dossier étaient élevés et assez frustrants, car c'est une étape monotone, mais théoriquement simple à achever.

Comme mentionné dans la revue de la littérature, de plus en plus, les compagnies s'aperçoivent de l'impact favorable des robots dans l'expérience employée globale. Ceci semble vrai dans notre cas, en éliminant la partie de vérification des documents des revenus, les analystes gagnent en efficience et en qualité de travail tout en réduisant la frustration et les erreurs d'inattention. Cependant, les résultats montrent que certaines améliorations sont nécessaires pour les différentes tailles de caractères. Pour bâtir la confiance envers cette technologie, il est primordial que cette partie-là soit infaillible.

7.3. Pistes d'améliorations

Cette recherche nous a permis de tester les capacités des RPA. La présente technologie n'est pas infaillible, surtout en ce qui concerne l'interprétation robotisée de l'information. On espère voir une amélioration future des IPA, pour avoir des robots qui répondent à 100 % aux besoins actuels de l'industrie. Mais nous sommes conscients de l'impact qu'il peut y avoir sur la performance et sur l'entretien de cette technologie.

Pour donner un exemple concret, imaginons un robot qui peut comprendre et analyser une lettre d'emploi, ou bien un talon de paie fournis par un client. Est-ce que ce robot sera capable de comparer ce genre de documents à des millions d'exemples pour pouvoir y déceler les informations nécessaires aux analystes de crédit ? Est-ce que l'impact de cette approche nous fera aussi gagner en temps de traitement et en qualité ?

Le deuxième exemple, que nous aurions bien aimé avoir est la capacité de détecter les anomalies dans un document , comme par exemple la taille ou la famille de polices diff-

rentes. Selon nos connaissances à date, aucun fournisseur n'offre ce service. Cela aurait aidé à faire une meilleure détection des cas de fraude pour les documents scannés.

Finalement, nous avons éliminé 35 % du temps de processus en restructurant le sous-processus de vérification de revenus. Il sera intéressant pour des recherches futures de voir si on peut améliorer le temps de cycle du reste du processus avec des séquences qui gèrent par exemple la vérification de la solvabilité ou la vérification du collatéral. L'autre piste d'amélioration qui reste à explorer est de réorganiser les séquences de tâches d'une façon parallèle et de mesurer les gains qui découlent d'une collaboration entre les différents robots.

CONCLUSION

Pour conclure ce rapport, cette recherche nous a permis de nous familiariser tout d'abord avec les RPA et avoir une bonne vue d'ensemble sur les plus grands joueurs de l'industrie. Nous avons testé aussi un aspect de la robotisation intelligente dans un domaine qui est encore en pleine croissance et avons exploré plusieurs pistes d'amélioration.

Notre solution nous a permis aussi de démontrer qu'il y a un gain considérable en automatisant la procédure de vérifications des documents de revenus. L'objectif initial était de pouvoir gagner au moins 50 % du temps de traitement des documents (environ 9 minutes). Nous nous sommes mis à l'évidence qu'on pouvait gagner la totalité du traitement des documents dans un temps considérablement court, peu importe le nombre de clients à traiter dans une demande de crédit.

L'objectif de cette recherche a bien été atteint, cependant la lecture intelligente des documents n'est pas encore fiable à 100%. Il existe donc certaines contraintes pour chaque projet RPA à prendre en considération comme la taille de police et la qualité des documents scannés.

En nous basant sur l'état présent de l'industrie, et les résultats obtenus lors de cette recherche, nous sommes très optimistes envers le futur des RPA et le développement des solutions intelligentes. Car ce qu'on a pu construire dans ce projet n'est qu'une infime partie du plein potentiel de cette technologie.

ANNEXE A.

SIMULATION DES DONNÉES BANCAIRES

Vous pouvez trouver ici le code source de l'API : [Github](#)

A.1. Création de l'environnement de développement

Modules utilisés pour la création de l'API

```
asgiref==3.4.1
dj-database-url==0.5.0
Django==3.2.6
django-heroku==0.3.1
djangorestframework==3.12.4
gunicorn==20.1.0
psycopg2==2.9.1
pytz==2021.1
sqlparse==0.4.1
whitenoise==5.3.0
```

A.2. Utilisation de Django REST framework.

Nous avons choisi Django REST framework pour sa flexibilité et sa facilité d'usage.

A.2.1. Modèle de données

```

from django.db import models

class Application(models.Model):
    code = models.CharField(max_length=10)
    date_soumise = models.DateField(auto_now=True)

    class Meta:
        ordering = ('-date_soumise',)

    def __str__(self):
        return self.code

class Client(models.Model):
    provinces = (
        ('NB', 'New Brunswick'),
        ('NL', 'Newfoundland and Labrador'),
        ('NS', 'Nova Scotia'),
        ('ON', 'Ontario'),
        ('PE', 'Prince Edward Island'),
        ('QC', 'Québec'),
    )
    types_client = (
        ('MAPL', 'Appllicant principale'),
        ('JAPL', 'Appllicant conjoint'),
        ('COAP1', 'Co-Appllicant1'),
        ('COAP2', 'Co-Appllicant2'),)

    code_demande = models.ForeignKey(Application, related_name='clients',
        ↪ on_delete=models.CASCADE)
    type_client = models.CharField( "Type: ", max_length=5, choices=types_client,
        ↪ default='MAPL')
    nom = models.CharField( "Nom: ", max_length=100)
    prenom = models.CharField("Prénom: ", max_length=100)
    no_civique = models.CharField("Numéro civique: ", max_length=20)
    adresse_civique = models.CharField("Adresse: ", max_length=100)
    code_postale = models.CharField("Code postale: ", max_length=7)
    province = models.CharField( "Province: ", max_length=2, choices=provinces)
    nas = models.IntegerField("Numéro d'assurance sociale: ")

```

```

date_de_naissance = models.DateField("Date de naissance: ", auto_now=False,
    ↪ auto_now_add=False)
employeur = models.CharField("Employeur: ", max_length=100)
salaire = models.DecimalField("Salaire: ", max_digits=8, decimal_places=2)

class Meta:
    unique_together = ['code_demande', 'type_client']
def __str__(self):
    return self.nom + ' ' + self.prenom

```

Listing A.1 – Création des *data models*

A.2.2. Exemple de *Serializer*

```

from rest_framework import serializers
from .models import Application
from .models import Client
import clients.views

class ApplicationSerializer(serializers.HyperlinkedModelSerializer):
    clients = serializers.HyperlinkedRelatedField(
        many=True,
        read_only=True,
        view_name='client-detail'
    )

    class Meta:
        model = Application
        fields = (
            'url',
            'pk',
            'code',
            'clients')

class ClientSerializer(serializers.HyperlinkedModelSerializer):
    code_demande =
        ↪ serializers.SlugRelatedField(queryset=Application.objects.all(),
        ↪ slug_field='code')

```

```
class Meta:
    model = Client
    fields = (
        'url',
        'code_demande',
        'type_client',
        'nom',
        'prenom',
        'no_civique',
        'adresse_civique',
        'code_postale',
        'province',
        'nas',
        'date_de_naissance',
        'salaire',)
```

Listing A.2 – Création des *serializers*

ANNEXE B.

DÉVELOPPEMENT DE LA SOLUTION AVEC UIPATH

B.1. Développement du robot

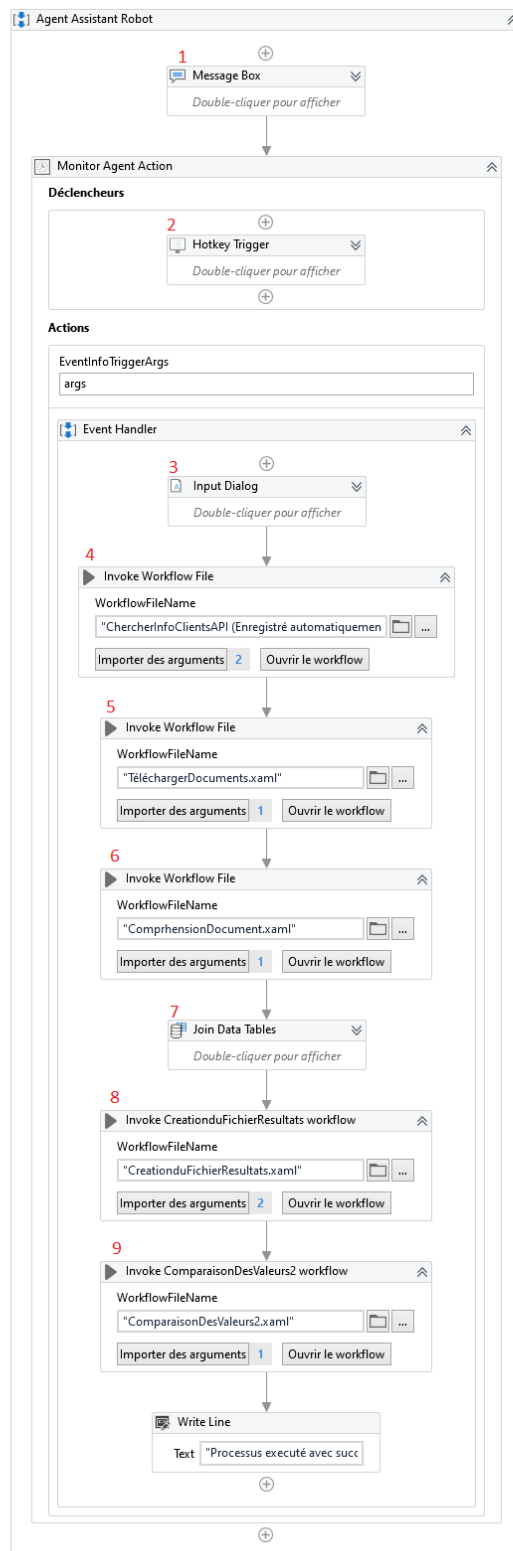


FIGURE B.1. – Solution complète Main

- (1) et (2) nous permettra de déclencher le robot par la composition de *hotkey*. (3) L'utilisateur est invité à saisir le numéro de dossier. (4) la séquence **ChercherInfoClientsAPI** est invoquée pour récupérer les données clients d'un dossier de crédit. Cette séquence utilise deux arguments (entrant : numéro de dossier ; sortant : table de données des clients). (5) la séquence **TéléchargerDocuments** est invoquée pour télécharger les documents de *OneDrive*. Cette séquence a besoin d'un argument (entrant : numéro de dossier). Une fois les documents sont téléchargés (6) la séquence **ComprehensionDocument** est lancé pour extraire les données de ces documents. Cette séquence retourne un argument (table de données extraites). (7) Nous faisons une jointure entre les deux tables retournées sur le NAS des clients. Cela nous permettra de créer une seule table avec toutes les données à vérifier. (8) Permettra de créer un fichier résultats. (9) Finalement, la séquence **ComparaisondesValeurs** nous permettra d'itérer sur la table de données et de comparer les valeurs ligne par ligne. Toute différence sera écrite dans le fichier résultat.

B.2. Connexion à l'API

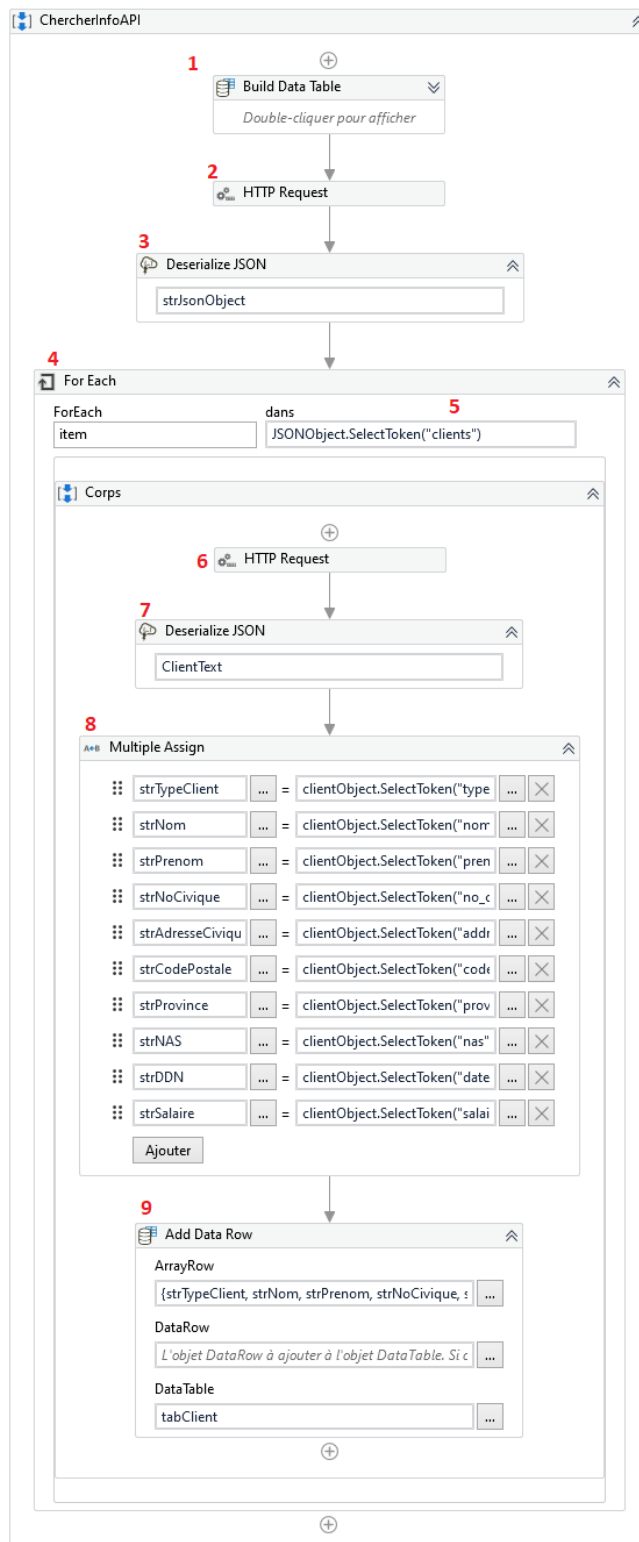


FIGURE B.2. – Chercher les infos des clients

Pour réaliser cette séquence, nous avons créé un tableau (1) pour stocker les informations retournées par l'API. (2) On fait ici un premier appel à l'API pour accéder à un dossier de crédit précis (fournis comme paramètre). Nous décodons l'objet JSON (3) retourné par ce dernier pour obtenir une chaîne de caractère (**Deserialize JSON**) qui contient la liste des clients (**Endpoint**). On va itérer chaque client dans une demande de crédit (4) pour extraire le lien (5) qui va permettre un deuxième appel à l'API (6) pour accéder aux données spécifiques d'un client (7) Enfin, nous allons assigner chaque valeur soustraite (8) dans une variable temporaire pour finalement la stocker dans le premier tableau (9). L'extrait de cette séquence est ce tableau.

B.3. Téléchargement des documents de revenus

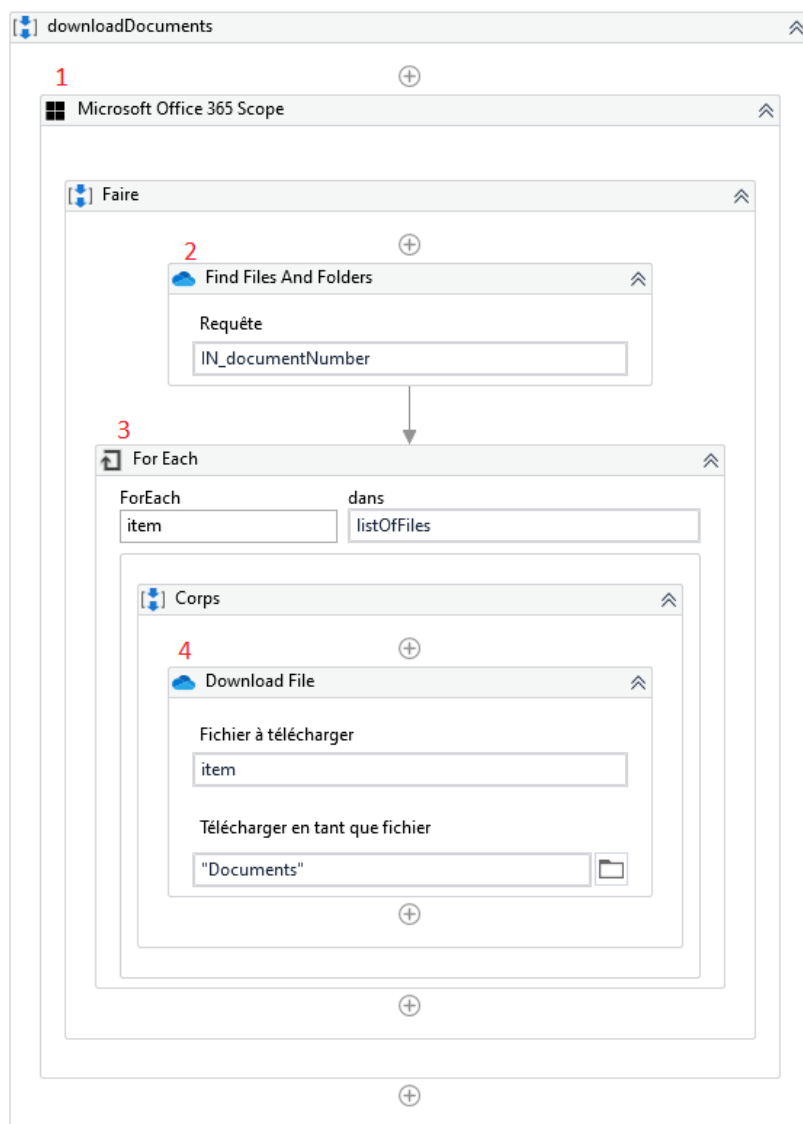


FIGURE B.3. – Chercher les infos des clients

Pour réaliser cette séquence, nous avons créé un tableau (1) pour stocker les informations retournées par l'API. (2) On fait ici un premier appel à l'API pour Dans **** Microsoft Office 365 Scope**** (1), on a défini les accès pour pouvoir utiliser le service *OneDrive*. (2) Nous permet de trouver un dossier de crédit existant avec l'argument saisi par

l'utilisateur. (3) Pour chaque fichier dans ce dossier on lance un téléchargement (4) dans le dossier local **Documents**.

B.4. Extraction intelligente des données

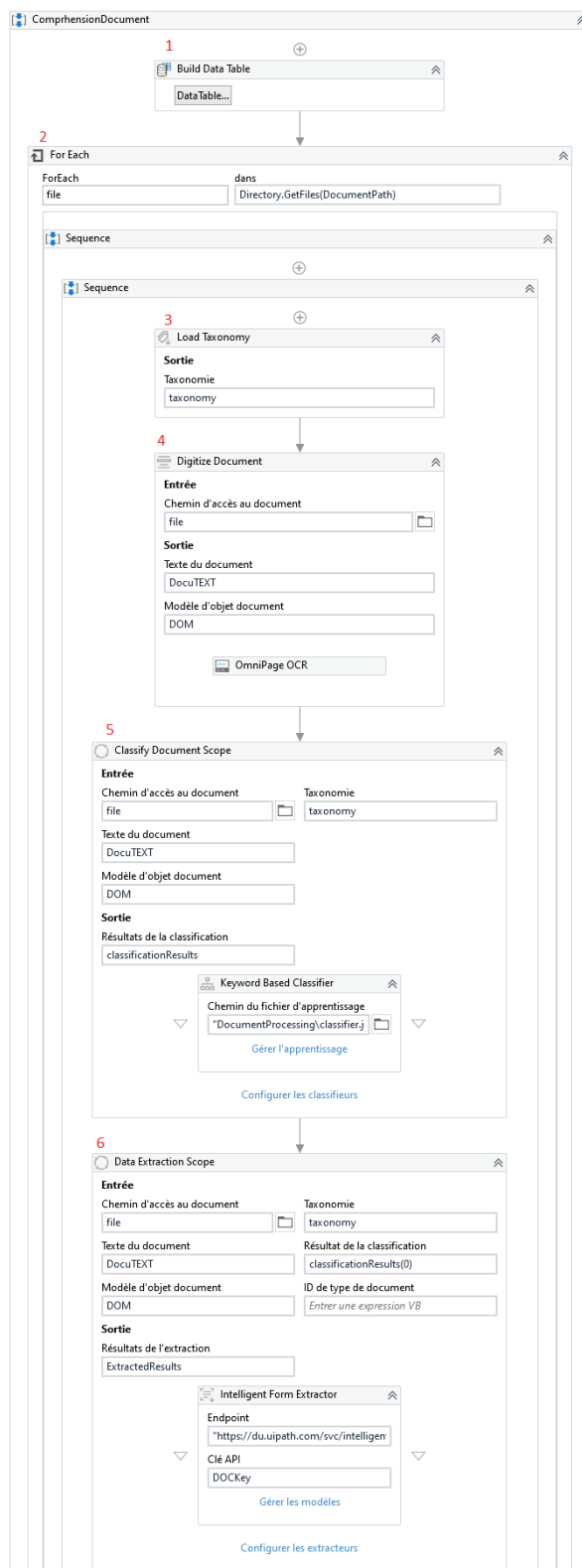


FIGURE B.4. – Extraction intelligente des documents 1/2

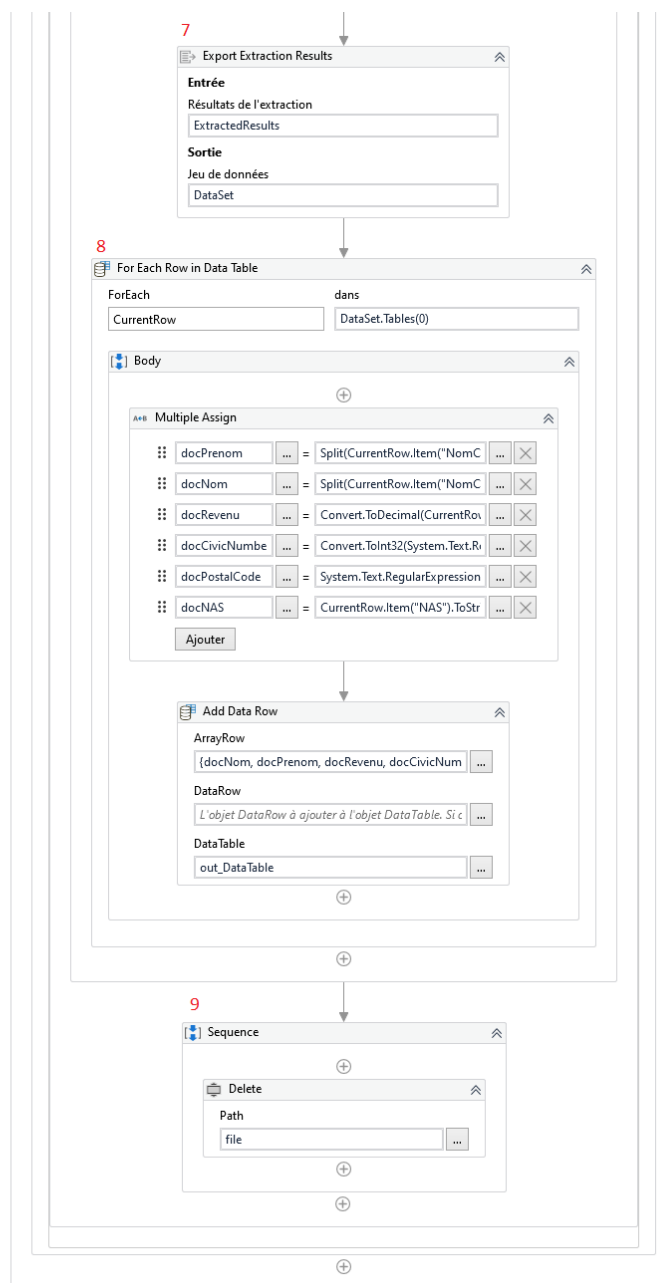


FIGURE B.5. – Extraction intelligente des documents 2/2

Dans cette séquence (1), nous allons tout d'abord créer un tableau vide pour stocker les données extraites. Pour chaque fichier dans Documents (2), nous allons télécharger la taxonomie utilisée pour définir les valeurs qu'on veut récupérer des documents (3). Ensuite, nous allons numériser (4) les documents en utilisant **OmniPageOCR**. Nous

allons classifier (5) les documents dans deux catégories (ADC et T4) définies dans le fichier **classifier.json**. Nous allons utiliser **Keyword Based Classifier**. L'étape (6) nous permet d'extraire les données. Pour ce faire, nous utilisons ici **Intelligent Form Extractor**. Les données sont extraites (7) sous forme d'un **DataSet**. Pour chaque élément (8) de ce jeu de données, nous allons l'attribuer à une variable (après transformation et validation). Les données validées seront ensuite stockées dans une table qui va être retournée à la séquence *main*. Finalement, on supprime le fichier traité (9).

B.5. Comparaison des valeurs

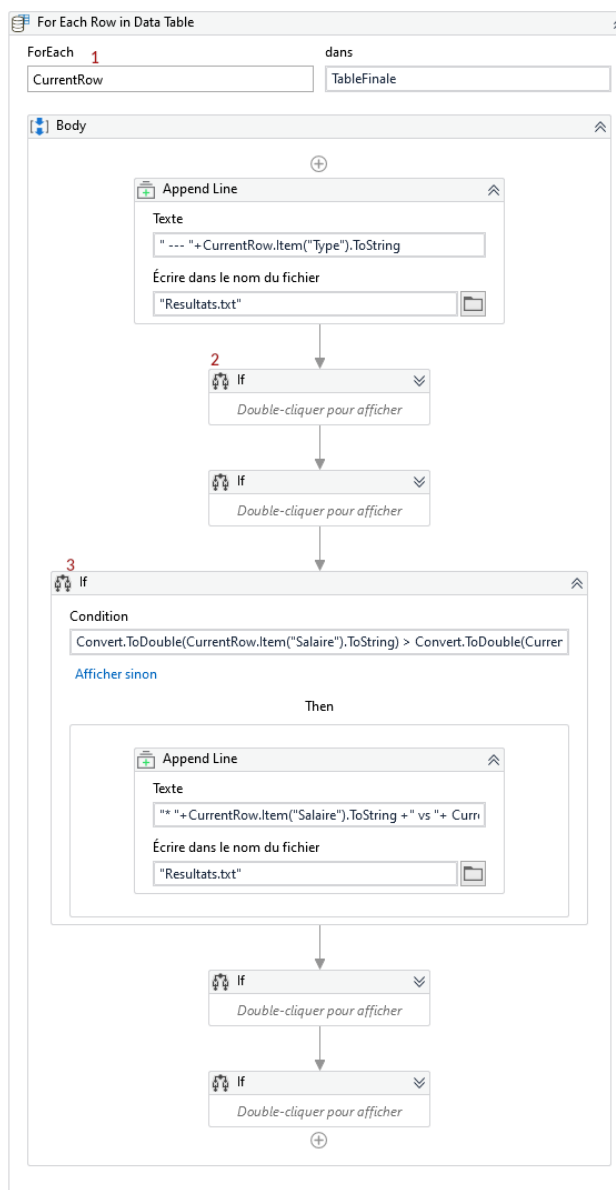


FIGURE B.6. – Comparaison des valeurs

Cette séquence (1) va prendre pour argument (entrant : table finale avec jointure faite dans la séquence *main*). Pour chaque ligne de cette table, nous allons appliquer une logique conditionnelle (2) pour s'assurer que les valeurs cherchées par l'API et celles

extraites des documents sont pareilles. Ex. (3) est un code pour vérifier si le salaire présente une variance de $\pm 10\%$.

```
Convert.ToDouble(CurrentRow.Item("Salaire").ToString) >  
    ↪ Convert.ToDouble(CurrentRow.Item("docRevenu").ToString) * 1.1 Or  
    ↪ Convert.ToDouble(CurrentRow.Item("Salaire").ToString) * 1.1 <  
    ↪ Convert.ToDouble(CurrentRow.Item("docRevenu").ToString)
```

Listing B.1 – Exemple de code en VB pour comparer le salaire

BIBLIOGRAPHIE

- ANAGNOSTE, Sorin, 2017. Robotic Automation Process-The next major revolution in terms of back office operations improvement. In : *Proceedings of the International Conference on Business Excellence*. Conference Proceedings. Sciendo. 2017. p. 676-686.
- ANAGNOSTE, Sorin, 2018. The Road to Intelligent Automation in the Energy Sector. In : *Management Dynamics in the Knowledge Economy*. 2018. Vol. 6, n° 3, p. 489-502.
- ASATIANI, Aleksandre et PENTTINEN, Esko, 2016. Turning robotic process automation into commercial success—Case OpusCapita. In : *Journal of Information Technology Teaching Cases*. 2016. Vol. 6, n° 2, p. 67-74.
- BERTHOLD, Michael R, BORGELT, Christian, HÖPPNER, Frank et KLAUONN, Frank, 2010. *Guide to intelligent data analysis : how to intelligently make sense of real data*. Springer Science & Business Media.
- BOSCO, Antonio, AUGUSTO, Adriano, DUMAS, Marlon, LA ROSA, Marcello et FORTINO, Giancarlo, 2019. Discovering automatable routines from user interaction logs. In : *International Conference on Business Process Management*. Springer. 2019. p. 144-162.
- CHAKRABORTI, Tathagata, ISAHAGIAN, Vatche, KHALAF, Rania, KHAZAENI, Yasaman, MUTHUSAMY, Vinod, RIZK, Yara et UNUVAR, Merve, 2020. From Robotic Process Automation to Intelligent Process Automation. In : *International Conference on Business Process Management*. Springer. 2020. p. 215-228.
- ENRÍQUEZ, José Gonzalez, JIMÉNEZ-RAMÍREZ, A, DOMÍNGUEZ-MAYO, FJ et GARCÍA-GARCÍA, JA, 2020. Robotic process automation : a scientific and industrial systematic mapping study. In : *IEEE Access*. 2020. Vol. 8, p. 39113-39129.
- FREY, Carl Benedikt et OSBORNE, Michael A, 2017. The future of employment : How susceptible are jobs to computerisation ? In : *Technological forecasting and social change*. 2017. Vol. 114, p. 254-280.
- GARTNER, 2021. Magic Quadrant Leader for RPA | Automation Anywhere. In : [en ligne]. 2021. Disponible à l'adresse : <https://www.automationanywhere.com/lp/gartner-magic-quadrant>.
- GOGINENI, Chandra, 2021. 10 Use Cases of RPA in Banking Industry. In : [en ligne]. 2021. Disponible à l'adresse : <https://www.comtecinfo.com/rpa/use-cases-of-rpa-in-banking-industry/>.

- GREGORY, Scott, 2019. Ten robotic process automation strategies for retail bank success. In : *Banking Strategies Daily*. 2019. p. 1-1.
- HERBERGER, Tim A et DÖTSCH, Jörg J, 2021. The Means Justifies the End ? Digitalization and Sustainability as a Social Challenge. A Plea for an Integrative View. In : *Digitalization, Digital Transformation and Sustainability in the Global Economy*. Springer. p. 1-8.
- HEVNER, Alan R, 2007. A three cycle view of design science research. In : *Scandinavian journal of information systems*. 2007. Vol. 19, n° 2, p. 4.
- HIRSCHI, Andreas, 2018. The Fourth Industrial Revolution : Issues and Implications for Career Research and Practice. In : *The Career Development Quarterly*. 2018. Vol. 66, n° 3, p. 192-204.
- INSIDER, Business, 2015. 2015. Disponible à l'adresse : <https://www.businessinsider.com/it-took-75-years-for-the-telephone-to-reach-100-million-users-and-it-took-candy-crush-15-months>.
- KOTARBA, Marcin, 2018. Digital Transformation of Business Models. In : *Foundations of Management*. 2018. Vol. 10, n° 1, p. 123-142.
- KUMAR, Karippur Nanda et BALARAMACHANDRAN, Pushpa Rani, 2018. Robotic process automation-a study of the impact on customer experience in retail banking industry. In : *Journal of Internet Banking and Commerce*. 2018. Vol. 23, n° 3, p. 1-27.
- LACITY, Mary, WILLCOCKS, Leslie P et CRAIG, Andrew, 2015. Robotic process automation at Telefonica O2. In : 2015.
- LE CLAIR, Craig, CULLEN, A et KING, M, 2017. The Forrester Wave™ : Robotic Process Automation, Q1 2017. In : *Forrester Research*. 2017.
- LEOPOLD, Henrik, AA, Han van der et REIJERS, Hajo A, 2018. Identifying candidate tasks for robotic process automation in textual process descriptions. In : *Enterprise, business-process and information systems modeling*. Springer. p. 67-81.
- LESHOB, Abderrahmane, BÉDARD, Maxime et MILI, Hafedh, 2020. Robotic Process Automation and Business Rules : A Perfect Match. In : *ICETE (3)*. 2020. p. 119-126.
- LESHOB, Abderrahmane, BOURGOIN, Audrey et RENARD, Laurent, 2018. Towards a process analysis approach to adopt robotic process automation. In : *2018 IEEE 15th International Conference on e-Business Engineering (ICEBE)*. IEEE. 2018. p. 46-53.
- MADAKAM, Somayya, HOLMUKHE, Rajesh M et JAISWAL, Durgesh Kumar, 2019. The future digital work force : robotic process automation (RPA). In : *JISTEM-Journal of Information Systems and Technology Management*. 2019. Vol. 16.
- MIGNOT, Océane, 2019. *La transformation digitale des entreprises : Principes, exemples, mise en oeuvre, impact social*. Maxima.

- PAPPAS, Ilias O., MIKALEF, Patrick, GIANNAKOS, Michail N., KROGSTIE, John et LEKAKOS, George, 2018. Big data and business analytics ecosystems : paving the way towards digital transformation and sustainable societies. In : *Information Systems and eBusiness Management*. 2018. Vol. 16, n° 3, p. 479-491.
- REINARTZ, Werner, 2019. Reinventing the Retailer : Retaining Relevance and Customer Access. In : *NIM Marketing Intelligence Review*. 2019. Vol. 11, n° 1, p. 10-17.
- RIZK, Yara, ISAHAGIAN, Vatche, BOAG, Scott, KHAZAENI, Yasaman, UNUVAR, Merve, MUTHUSAMY, Vinod et KHALAF, Rania, 2020. A conversational digital assistant for intelligent process automation. In : *International Conference on Business Process Management*. Springer. 2020. p. 85-100.
- SANTOS, Filipa, PEREIRA, Rúben et VASCONCELOS, José Braga, 2020. Toward robotic process automation implementation : an end-to-end perspective. In : *Business Process Management Journal*. 2020. Vol. 26, p. 405-420.
- SCHWAB, Klaus, SALA-I-MARTIN, Xavier et OTHERS, 2014. The global competitiveness report 2014-2015. In : 2014. Vol. 549, p. 36-38.
- SLABY, James R, 2012. Robotic automation emerges as a threat to traditional low-cost outsourcing. In : *HfS Research Ltd*. 2012. Vol. 1, n° 1, p. 3-3.
- TAULLI, Tom, 2020. *The Robotic Process Automation Handbook : A Guide to Implementing RPA Systems*. Springer.
- VIEHHAUSER, Johannes, 2020. Is robotic process automation becoming intelligent ? early evidence of influences of artificial intelligence on robotic process automation. In : *International Conference on Business Process Management*. Springer. 2020. p. 101-115.
- ZAKI, Mohamed, 2019. Digital transformation : harnessing digital technologies for the next generation of services. In : *The Journal of Services Marketing*. 2019. Vol. 33, n° 4, p. 429-435.