

**L'AGILITE AU SERVICE DE E-BUSINESS : CONCEPTION D'UN SYSTEME D'E-COMMERCE DE VENTE EN LIGNE DE TICKETS DE TRAIN A LA SOCIETE NATIONALE DE CHEMIN DE FER DU CONGO (RDC).**

**AGILITY AT THE SERVICE OF E-BUSINESS: DESIGN OF AN E-BUSINESS SYSTEM FOR THE ONLINE SALE OF TRAIN TICKETS TO THE CONGO (DRC) NATIONAL RAILWAY COMPANY.**

**NYAMI NYATE Ruphin1 (Msc)**

Institut Supérieur de Commerce d'Ilebo / RDC

**BABULA NGUMBI Rene (Msc)**

Institut Supérieur de Commerce de Lubumbahi / RDC

**IMBUANGA BULEU**

Institut Supérieur Pédagogique d'Ilebo / RDC

**ISSABA LANDU Gilbert**

Institut Supérieur Médical d'Ilebo / RDC

**IBANDJI KOMBE Blaise**

Institut Supérieur de Commerce d'Ilebo / RDC

**KABWIKA Jean-Marie**

Institut Supérieur de Commerce d'Ilebo / RDC

**TWITE Grac(Msc)**

Universté de Lubumbashi/Ecole Supérieur des Ingénieurs Industriels / RDC

**BALUME Sylvain(Msc)**

Universté de Lubumbashi/Ecole Supérieur des Ingénieurs Industriels / RDC

**NTUMBA KWAKOMBE Marlène**

Institut Supérieur de Commerce d'Ilebo / RDC

**Date de soumission : 09/11/2022**

**Date d'acceptation : 28/02/2023**

**Pour citer cet article :**

R. NYAMI & al (2023) «l'agilité au service de e-business : conception d'un système d'e-commerce de vente en ligne de tickets de train a la société nationale de chemin de fer du Congo», Revue Internationale du chercheur «Volume 2 : Numéro 4 » pp : 333- 352

## Résumé

Cet article concerne la mise en œuvre d'un processus simplifié d'un système de vente en ligne de Tickets de train allant de la modélisation jusqu'à la conception détaillée. L'objectif est de concevoir un système de réservation et achat en ligne de ticket de train « PAMASOFT » afin de booster l'activité économique d'une entreprise post-COVID. Nous présentons un modèle d'exigences fonctionnelles ; suivi d'une vue structurelle des classes et objets ; ensuite une vue comportementale décrivant le dynamisme du système d'objets dans l'espace de solution. Pour le premier modèle d'exigences, il a été réalisé à partir du carnet de cas d'utilisations UML (Unified Modeling Language) ainsi que le diagramme de contexte dynamique décrivant les flux et actions du métier. L'allégeance au modèle de conception est faite par le diagramme de classes participantes. Le principe d'agilité de la démarche XP (eXtreme Programing) est matérialisé par la conception de maquettes très tôt dans la phase d'analyse de besoins. Cet article fait également les principes de la Méthode UP (Unified Process) dans la phase de conception d'un système logiciel. Les différentes phases sont couvertes : Analyse de besoins, Rédaction de scénarios et la conception en elle-même. L'outil Entreprise Architect a été choisi comme Atelier de Génie Logiciel (AGL).

**Mots clés : Agilité ; classes participantes, UP ; Vente en ligne ; Ticket Train ; e-commerce**

## Abstract

This paper concerns the implementation of a simplified process of an online train ticket sales system from modeling to detailed design. The objective is to design an online reservation and purchase system for "PAMASOFT" train tickets in order to boost the economic activity of a post-COVID company. We present a model of functional requirements; followed by a structural view of classes and objects; then a behavioral view describing the dynamism of the system of objects in the solution space. For the first requirements model, it was produced from the UML (Unified Modeling Language) use case book as well as the dynamic context diagram describing the flows and actions of the business. Allegiance to the design pattern is made by the participating class diagram. The principle of agility of the XP (eXtreme Programming) approach is materialized by the design of models very early in the needs analysis phase. This article also covers the principles of the UP Method (Unified Process) in the design phase of a software system. The different phases are covered: Needs analysis, Writing scenarios and the design itself. The Enterprise Architect tool was chosen as the Software Engineering Workshop (AGL).

**Keywords: Agility; participating classes, UP; Sold online ; train tickets; ecommerce**

## Introduction

Le monde post-COVID-19 subit une refonte totale et une mutation rapide pour les entreprises soucieuses d'épanouissement économique. Les entreprises adhérant ainsi à la nouvelle économie des affaires qui, a pour sous-bassement le commerce électronique(Rallet, 2001). L'infrastructure technologique de la nouvelle économie est soumise à de contraintes de développement informatique afin de produire de la valeur commerciale pour l'entreprise. En outre, la présence des nouvelles technologies aident à prendre des décisions rapides et efficaces(Choudhary & Pandita, 2022). Sur base d'une étude empirique, Diana cité par Setyowati et al. observent que la gestion des actifs primaires des organisations modernes passent par le commerce électronique(Setyowati et al., 2021). Ces actifs sont échangés sous forme de données informatisées (EDI) visant ainsi la dématérialisation de processus, de tableau de bords électroniques, des emails et de différents flux (financiers, commerciaux, flux liés aux technologies réseaux). Ce modèle reste un mécanisme basé sur les outils technologiques de gestion commerciale et innovatrice d'entreprise. Selon les résultats de son étude, Al-Omouh (Al-Omouh, 2020) souligne que le succès du commerce électronique implique une agilité organisationnelle et un soutien inconditionnel de décideurs de l'organisation dans la préparation d'un environnement e-business. Ainsi donc, le développement agile reste intimement lié à l'agilité organisationnelle afin que le e-business produise de la valeur pour l'entreprise. De plus, Cahyono et al. (Cahyono et al., 2022) soulignent que l'agilité au service d'e-business et la capacité d'innovation des entreprises à caractère commercial ont un effet favorable dans l'amélioration de leur performance et de l'optimisation de la croissance de l'économie numérique grâce au commerce électronique. Néanmoins, au niveau des facteurs organisationnels, *Kameni* souligne qu'outre la culture organisationnelle ou influence politique, la démocratisation de l'information, sa fluidité et la compétition dans l'axe d'affaires exercé peuvent influencer le succès d'une solution e-business (*Kameni*, 2023). Toutefois, seule, l'intelligence économique n'augmente pas la valeur commerciale de l'entreprise, mais plutôt l'agilité de pouvoir décider peut devenir une source de satisfaction et moteur de transformation compétitive(Schlömer, 2022) (Yan et al., 2019). Le résultat de l'étude empirique de Legras confirme que le développement agile apporte à l'entreprise un dynamisme qui intègre les facteurs de temps, de coût, de qualité et les ressources humaines de l'équipe informatique ainsi que les feedbacks rapides(Legras, 2014). De plus de l'agilité qui exige de bonnes manières de parties prenantes pour la réussite d'un projet informatique. Bien souvent, l'approche agile se



juxtapose avec d'autres méthodologies basées sur le génie logiciel pour atteindre l'objectif (Roques, 2008). De nombreuses recherches proposent de mécanismes de sécurités basés sur le contrat électronique ou l'architecture d'e-business d'aide à la décision de gestion basé sur les simulations informatiques(Gâteau et al., 2004), pour analyser le rendement des projets au fil du temps et les impacts des décisions de gestion stratégique sur la capacité dynamique des services de commerce électronique. D'autres solutions puissantes d'e-business existent en l'occurrence EBSME (Electronic Business for the Small and Medium Enterprises) une application dédiée au commerce de services électroniques(Gâteau et al., 2004). Néanmoins l'un des atouts des applications d'e-commerce est d'externaliser certains tâches ou de consommer les services tiers(Niranjanamurthy & Chahar, 2013). Ce mode de fonctionnement rend le contrôle fiscal difficile. A cet effet, *Rukanova* et son équipe de recherche (Rukanova et al., 2021) proposent une plateforme d'analyse de flux d'e-commerce au pouvoir publique afin d'extraire d'informations sur les systèmes d'e-commerce. Toutefois, la traçabilité de flux financiers dans un système d'e-commerce exige beaucoup de compétences de la part de 15 administrations publiques. Le système « PAMASOFT » proposé ne déroge pas à la règle à l'instar de la taxe sur la valeur ajoutée incorporée dans le ticket de voyage. En outre, le système d'e-commerce reste génialement vulnérable au regard de l'interopérabilité du Web. Pour relever ce défi, des architectes de la sécurité informatique propose une solution de sécurisation(Niranjanamurthy & Chahar, 2013). Le domaine de commerce électronique fait usage de transactions dites EDI ( Electronic Data Interchange) largement répandu. Cette technologie fait référence à l'aptitude d'une entreprise d'envoyer sous format standard des services et des produits par voie électronique à une autre société ou d'échanger des informations commerciales au sein de sa propre société ou encore avec de consommateurs(Setyowati et al., 2021). Néanmoins l'EDI suppose l'existence préalable des outils d'e-business dans le système d'information de l'entreprise. En plus, la solution EDI est naturellement complexe à déployer pour une entreprise novice. D'où l'existence de plusieurs plateformes d'e-commerce typiques et propriétaire fonctionnement comme tiers de confiance(Mercanti-Guérin, 2022). Cependant ces plateformes font l'objet de doutes pour les organisations inexpérimentés dans le commerce électronique(Dekhili 1 & Achabou 2, 2011). C'est à juste titre que cet article vise à concevoir un système typique de vente en ligne de tickets de voyage dans une entreprise frappée de plein fouet par la pandémie de COVID-19. La Société Nationale de Chemins de Fer du Congo (SNCC) a subi un blackout durant la période la pandémie de la COVID-19 et cela a révélé les limites de méthodes traditionnelles de ventes de tickets de train notamment : la priorisation du

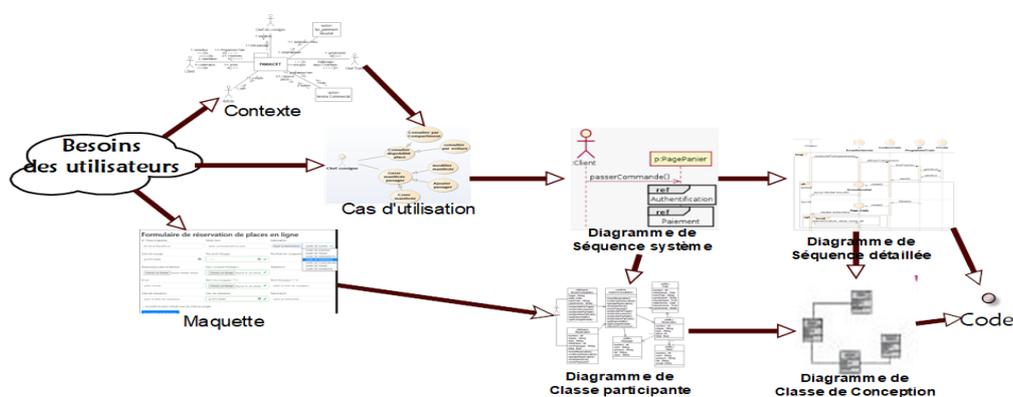
transport de marchandises, arrêt total de transport de passagers ou la fermeture des certaines gares de transit. De surcroît, les plaintes de certains passagers sur la volatilité du ticket de voyage, la lourdeur administrative lors de contrôle de passagers et le débarquement musclé des passagers par des agents de sécurité durant le trajet du train sont la preuve irréfutable de l'inefficacité de la politique de gestion mise en place. En outre, le système actuel continu à imposer un déplacement physique et une queue obligatoire devant le guichet pour réserver une place ou avoir de renseignement sur le programme de voyage. L'objectif cet article est de promouvoir la recherche en e-business et en l'ingénierie logicielle en concevant un système numérique de vente de tickets de train. La démarche simplifiée adoptée utilise trois phases à savoir : l'analyse de besoins, la conception préliminaire et la conception détaillée. Il s'agit donc de répondre à la question suivante : Comment concevoir des solutions d'e-commerce tenant de valeurs commerciales rendant ainsi l'entreprise agile pour son indice de bonheur ? Pour répondre à cette question, nous adoptons une démarche à mi-chemin entre le Processus Unifié (Unified Process) et la méthode Agile XP (eXtrem Programming). La portée de la méthode UP dans le développement des outils de commerce électronique reste générique. Il est perçu comme la décompte de développement et réputé en lourdeur vu le nombre de phase à réaliser avant d'obtenir le produit final. Nous utilisons le Processus Unifié pour concevoir notre système d'e-commerce de vente en ligne de tickets de voyage autour de 3 points à savoir : la méthodologie conception de systèmes d'e-business, l'analyse du problème et la conception de la solution d'e-commerce.

## 1. Méthodologie

Le développement d'un système e-business proposé dans cette étude exhale un processus hybride à la croisée de la démarche d'UP (Unified Process) et un ensemble de bonnes pratiques sur la capture des exigences avec le processus Agile (Roques, 2008). La démarche UP propose un agencement d'états non linéaires dans le processus de développement Unified complexe, y compris la modularité du système en tenant compte de facteurs de risques (Egholm et al., 2020). Développer un logiciel critique globalement peut s'avérer une tâche complexe et longue plus qu'il n'y paraît. Par conséquent, UP préconise un développement itératif à une durée maximale de quatre semaines afin de favoriser une livraison fréquente (Wong et al., 2020). Chaque itération passent par un ensemble d'étapes indissociables pour produire une solution exécutable à savoir : *le recueil des exigences, la construction proprement ; la construction* (la conception du système et le codage informatique, tests autres ; *la transition* où le produit construit est

transféré au près des utilisateurs finaux. Le processus utilise le langage de modélisation UML pour la représentation de facettes du système à développer. Ce langage est constitué d'un ensemble de diagrammes ayant chacun de pictogrammes distincts pour représenter les éléments du système. L'approche Agile est issue d'une entente de 17 entreprises industrielles signée dans le manifeste agile (Hamzane, 2021) (Kiv et al., 2018) avec comme base 4 valeurs à observer : « Personnes et interactions plutôt que processus et outils » « Logiciel fonctionnel plutôt que documentation complète » ; « Collaboration avec le client plutôt que négociation de contrat » : « Réagir au changement plutôt que suivre un plan ». Pour la simplicité de notre étude, la démarche de conception de notre système d'e-business suit le schéma suivant : (1) l'Analyse de besoins des utilisateurs sur l'achat de tickets de voyage en ligne, (2) la conception de maquettes décrivant le scénario d'utilisation ; (3) la modélisation du contexte du système afin de comprendre les responsabilités du système ; (4) l'identification de besoins des utilisateurs sous forme de cas d'utilisation afin de définir les responsabilités fonctionnelles du système ; (5) la modélisation des interactions entre l'acteur et le système dans un diagramme de séquences système ; (6) l'identification des objets participants à la réalisation de chaque cas d'utilisation ; (7) la modélisation des interaction dans un diagramme de séquences détaillées ; (8) affinage de classes participantes en tenant compte de contraintes techniques et des opérations issues de diagramme de séquences détaillées ; (9) la génération de structures de codes à partir des ateliers de Génie Logiciel. La figure 1 suivante illustre la démarche minimale utilisée dans cet article.

**Figure 1—schéma de la démarche utilisée**



Source : (Roques, 2008)

## 2. Analyse de besoin

### 2.1. Présentation du problème

L'étude de cas mise en œuvre dans notre démarche agile consiste en la modélisation d'un système e-business d'achat en ligne de tickets de voyage dans une société de transport ferroviaire en RDC. La réservation en ligne de tickets non seulement une facilitation pour le client qui peut économiser son temps, minimiser le coût de transport et lui éviter de longues files d'attentes sans oublier l'omniprésence du billet en ligne même en cas de perte mais aussi une réadaptation de l'organisation suite à la crise post-Covid-19. Ce cas d'étude est en effet un exemple indiqué parmi tant d'autres pour illustrer un projet d'e-commerce.

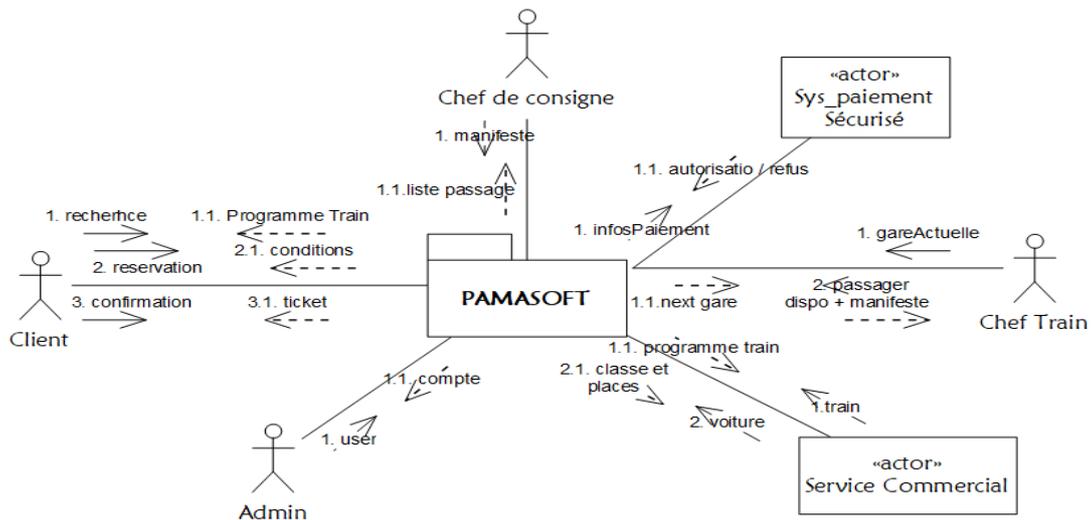
L'objectif fondamental d'un système est de permettre aux clients lointains, de rechercher des programmes de train par trajet, par période, mot-clef, etc., de se faire une réservation d'une place sans engagé une force physique ou bousculade puis de pouvoir l'acheter en ligne via un système de paiement Mobile interfacé à notre système « PAMASOFT ». Tout commence par la programmation d'un train par le Service commercial de la SNCC et qui le met à la disposition du public ; le client consulte différents trains alignés puis choisit au besoin un train avant procéder à la confirme par un paiement en ligne ; à chaque fois, le client peut à son gré annuler ou modifier avant la confirmation effective de la réservation.

Chaque train est piloté par agent « Chef Train » chargé de la supervision du trajet, la mise à jour du manifeste et de la disponibilité de places ; durant son trajet, un train passe par un certain nombre de gares qui disposent chacune un « Chef consigne » chargé de communiquer l'arrivée du train aux passagers en attente ; le système es géré par agent « Admin » ayant la charge de contrôler et maintenir le système. Dans la figure 2 suivante nous montrons la maquette de réservation de ticket dans notre système PAMASOFT ; par contre la figure 3 suivante illustre le contexte d'utilisation du système PAMASOFT où chaque acteur est relié par une association avec le rectangle représentant le système « PAMASOFT ».

Figure 2—Maquette de réservation de ticket de train

Source : Auteurs

Figure 3—Modélisation du Contexte



Source : Auteurs

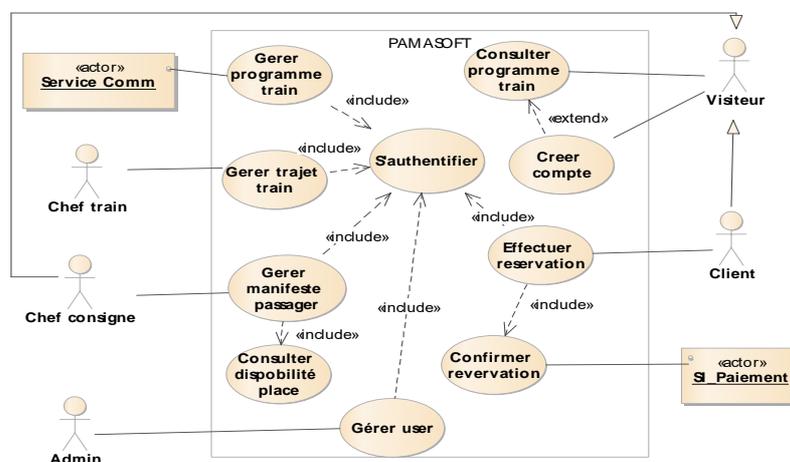
## 2.2. Identifications de cas d'utilisations du système PAMASOFT

La présente étape consiste à recenser les cas d'utilisations de chaque acteur retenu comme utilisateur final de notre système. Ainsi les besoins fonctionnelles retenus pour le système PAMASOFT sont les suivantes :

- Cas d'utilisation « **Gérer programme train** » cette fonctionnalité permet au service commerciale de mettre à jour un train donné et d'en publier en ligne ;
- Cas d'utilisation « **Consulter programme Train** » : pour effectuer une réservation, ce cas un client doit préalablement consulter un train donné pour une éventuelle réservation ;
- Cas d'utilisation « **Effectuer réservation place** » : cette fonctionnalité permet à un client faire une réservation en ligne d'une place dans le train ;
- Cas d'utilisation « **Créer compte utilisateur** » : cette fonctionnalité permet à un visiteur qui vient pour la première fois de s'inscrire dans le site ;
- Cas d'utilisation « **Confirmer réservation** » : cette fonctionnalité permet à un client de payer en ligne et confirmer sa réservation de ticket
- Cas d'utilisation « **Gérer user** » : cette fonctionnalité permet à l'administrateur de gérer le profil des utilisateurs du système ;
  - Cas d'utilisation « **Gérer Trajet train** » : ce cas d'utilisation permet au Chef train de mettre à jour le trajet du train

Ces cas d'utilisation constituent des exigences fonctionnelles de notre système et la suite du travail sera guidée par ces exigences afin de corroborer avec la démarche UP qui est conduit par les cas d'utilisation comme illustré dans la figure 4 suivante :

**Figure 4—Diagramme de cas d'utilisations du système PAMASOFT**



Source : Auteurs

Il sied de noter que pour besoin de simplicité prônée dans cet article, nous nous restreindrons à la fonctionnalité de **consultation du programme et de la réservation du ticket de voyage**.

Nous donnons ci-dessous la description textuelle détaillée de deux cas d'utilisation qui nous concerne afin d'améliorer le contenu informatif sur le déroulement de chacun.

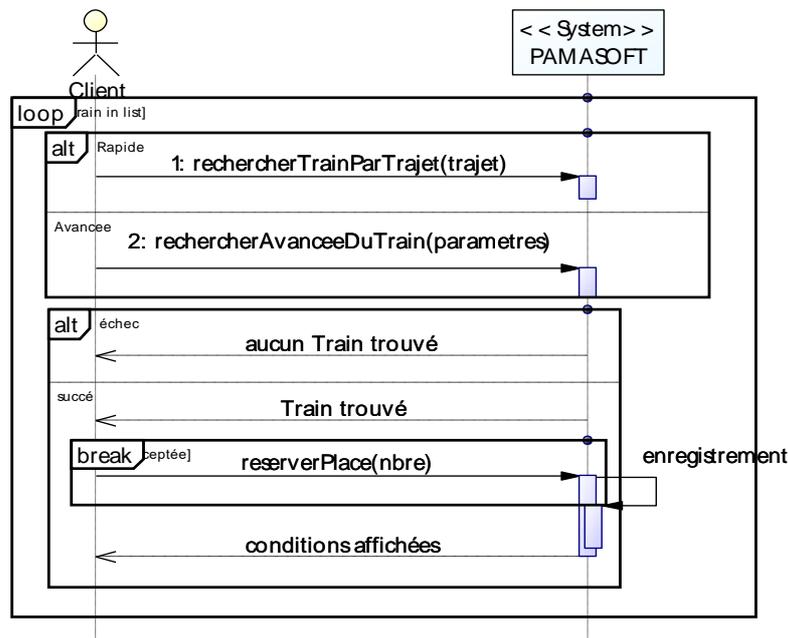
La consultation d'un catalogue du programme de train est l'apanage du rôle d'Internaute (client) qui veut trouver rapidement un train précis ; cependant un le programme de train doit être préalablement disponible. Ensuite, il réalise un scénario suivant :

**Tableau 1—Scénario de consultation de programme de Train**

Acteur	Système
1. L'internaute effectue une recherche rapide par mots-clés (le nom, le trajet ou une gare de destination du train)	1.1. La page de résultat affichée de recherche. Le résultat est affiché de manière horodatée
2. Choix du train	2.1. Affichage de détails de train choisi
3. Optionnellement un client peut parcourir le programme de train typé (diamant béton, train ordinaire ou par gare)	3.1. En cas d'échec, le système ramène le client à la recherche (scénario); sinon l'exécution du scénario nominal 2.1

La figure 5 suivante illustre le scénario de consultation de programme de train par un internaute ou un client déjà inscrit.

**Figure 5—Diagramme de séquences système de consultation « Programma train »**



Source : Auteurs

**- Processus de réservation de tickets de train**

Un client intéressé par un train quelconque, doit pouvoir accéder au formulaire de réservation, dans lequel il peut saisir ses coordonnées et les informations nécessaires au paiement et recevoir son ticket. Par ricochet, la sélection d’un train est obligatoire lors du processus de consultation du programme. De plus, le client doit prouver l’identité correcte avant toute chose. A l’issue de la réservation, le système doit pouvoir enregistrer une réservation avant sa transmission au service commercial. Ensuite, un API (Application Programming Interface) de sécurisation de paiement doit être intégrée pour le cryptage de la transaction.

Le scénario de réservation se déroule comme illustré dans le tableau suivant :

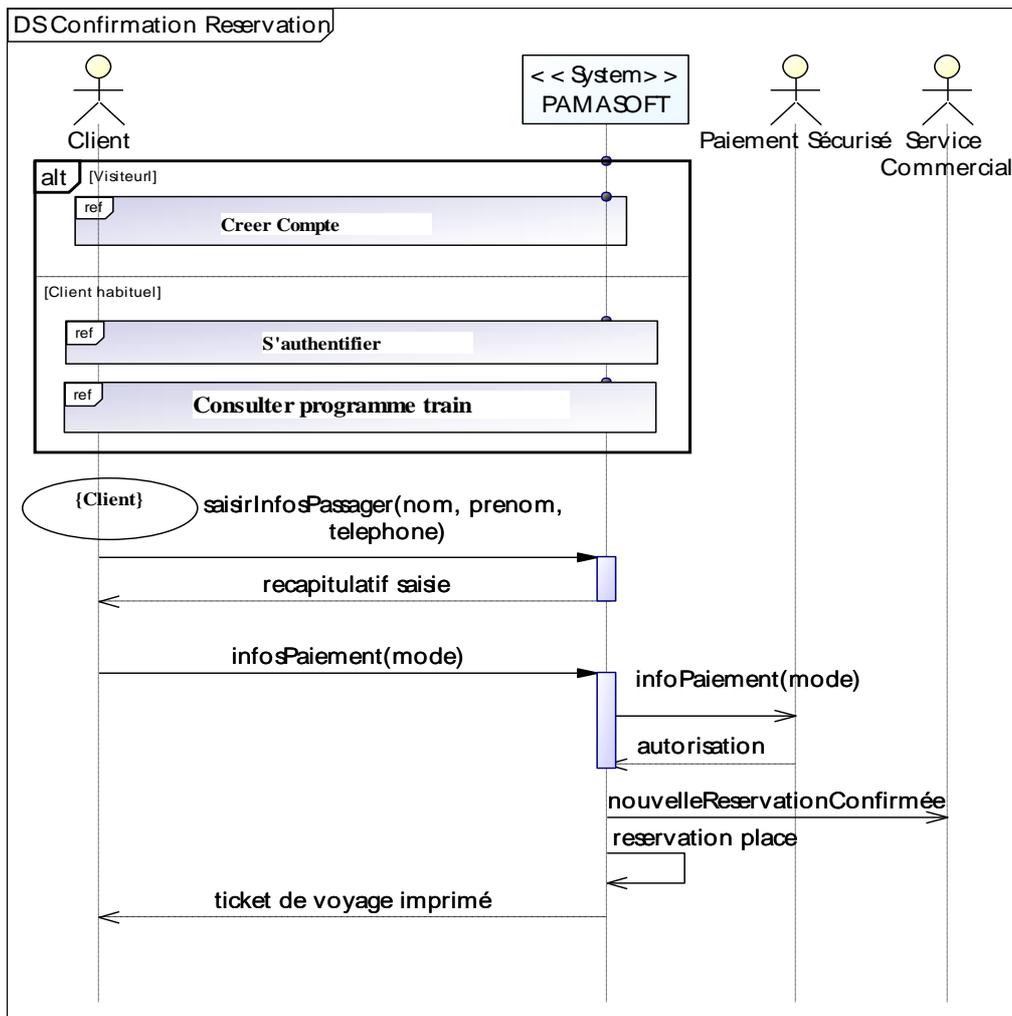
**Tableau 2—Scénario de réservation de place dans le train**

Acteur	Système
1.1.Le client décline respectivement son identité et celle du passager	1.1.Le récapitulatif de coordonnées saisis affichés avec la réservation encours
2. Le client opte un type de paiement et confirme la réservation encours	2.1. Agrégation du paiement au près du système sécurisé 2.2. En cas de succès, le système affiche la autorise le paiement; sinon une exception est levée 2.3.Transmission de la réservation validée pour la mise à jour du manifeste
3. Le client retire le ticket de train	

Le système dispose des options permettant à un client d'annuler une réservation. Le cas échéant, le processus se termine en échec.

La figure 6 suivante illustre un diagramme de séquences modélisant les échanges entre l'acteur client et le système lors de la réservation et confirmation d'une place dans un train.

**Figure 6—Diagramme de séquences « confirmer réservation »**



Source : Auteur

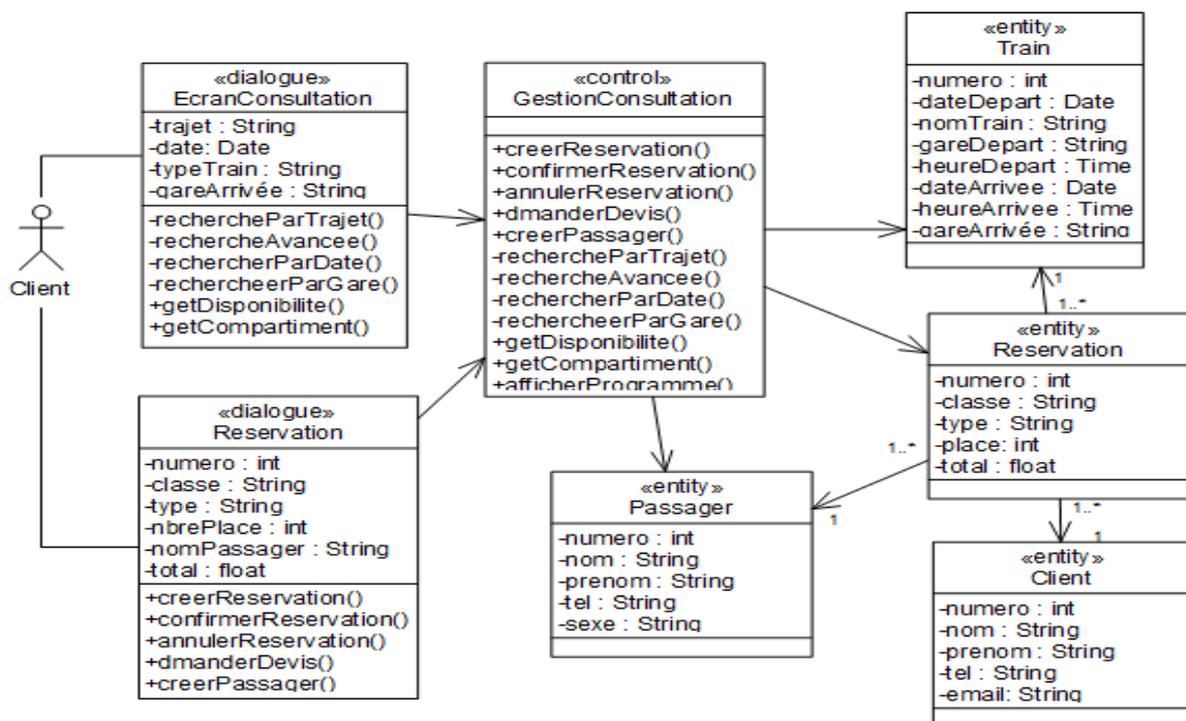
### 3 Conception du système d'e-commerce « PAMASOFT »

#### 3.1. Réalisation des cas d'utilisation : Classes participante

L'une de caractéristiques du langage UML est de pouvoir modéliser le paradigme orienté objet de classes, associations et attributs tant dans la phase d'analyse que dans celle de conception. (Djelil et al. 2020). Il y a trois types de classes à modéliser dans la phase d'analyse (Roques, 2008) (Veitaitė & Lopata, 2021) : les « dialogues » qui représentent la quasi-totalité des échanges d'information entre l'utilisateur et le système logiciel (l'entrée de données par

l'utilisateur, affichage de questions posées à l'utilisateur par le système, affichage de résultats obtenus,...) ; les « contrôles » qui constituent le canal de validation et communication entre le dialogue et la partie données, jouant ainsi le rôle l'intelligence de l'application cachée ; et les « entités » qui sont les concepts issus du métier contenant des données à persister. Par respect de protocole de communication, toute modélisation de ces trois classes peut se faire en obéissant aux règles suivantes : les classes « dialogues » dépendant de « contrôle » et les classes de contrôle dépendant des entités. Par ailleurs, les dialogues peuvent contenir les attributs (représentant les champs de saisies, liste de choix) et les opérations (actions de l'utilisateur sur les différents contrôles graphiques) ; les classes de contrôle contiennent uniquement l'interface de classes entités ; et les entités contiennent les attributs seulement et, les entités peuvent être en relation entre elles. Dans la figure 7 suivante, nous représentons l'exemple d'un diagramme de classes UML que nous appellerons diagramme de classes.

**Figure 7—Diagramme de classes participantes du cas d'utilisation « Consulter programme train »**

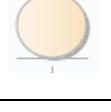


Source : Auteurs

### 3.2.Réalisation des cas d'utilisation : conception préliminaire de PAMASOFT

Dans cette section, nous allons doter à chaque objet son comportement décrivant les actions à effectuer durant son cycle de vie. L'une de principal rôle de diagramme de séquence UML est de modéliser le dynamisme d'échanges d'informations entre l'acteur et le système dans la réalisation d'une activité ou entre les objets applicatifs (Neto et al., 2021). Tout échange d'information entre objets passe par l'invocation de méthodes publiques entre les objets constituerons des opérations supplémentaires les diagrammes de classes de conception (Neto et al., 2021). Le diagramme de séquence conçu dans cette section permet de voir de l'intérieur « boîte blanche » afin de voir le concert des objets chacun avec ses responsabilités pour accomplir l'objectif (Ruphin, 2021). Comme annoncé précédemment, les types de classes doivent intervenir dans le diagramme de séquence détaillée dévoilant ainsi toute ambiguïté autour de l'acteur « système ». Le tableau 1 suivant représente le pictogramme UML utilisé pour corroborer avec les classes participantes :

**Tableau 3—Liste de pictogramme d'objets du diagramme de séquences**

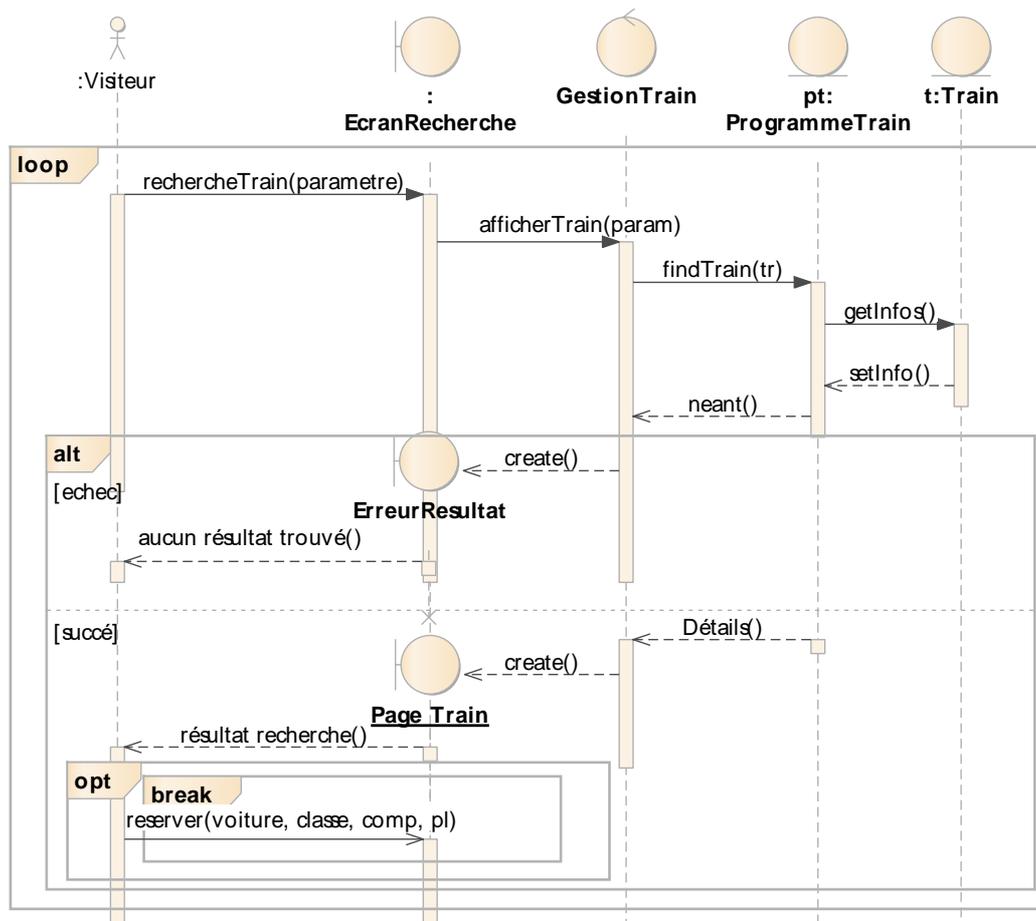
Pictogramme	Classe stéréotypée
	« dialogue ou vue » : objet de la classe d'interface graphique permettant l'interaction entre l'acteur et le système
	« contrôles » : objet de la classe contrôlant les échanges entre les dialogues et les entités.
	« entité » : un objet de la classe pouvant persister son état (ensemble de valeurs des attributs en un instant T)

**Source : Auteurs via l'Outil Entreprise Architech**

Le diagramme de séquence de la Figure 8 suivant illustre les interactions entre les 3 types d'objets précités ; où l'acteur « Visiteur » cherche à trouver un programme de train selon sa destination dans l'objet « dialogue » de deux manières à savoir par mot-clé ou par paramètre. Le dialogue invoque un contrôle responsable de la gestion de « Programme de Train ». Ce contrôle a la responsabilité de contacter l'entité « Programme Train » et « Train » afin d'avoir la collection de train correspondants à la recherche ; de créer le vue « ErreurResultat » et « **Page Train** » autant de fois que l'utilisateur le souhaite « loop » ; toutefois chaque recherche fait l'objet d'un test « alt » qui aboutit au succès ou à l'absence de résultats. En cas d'affichage du

programme de Train, l'utilisateur peut décider ou de réserver « opt » ce train. Ce scénario se consolide dans le diagramme de séquences de la figure (8).

**Figure 8—Diagramme de séquence détaillée de l'opération système « rechercher infos train»**

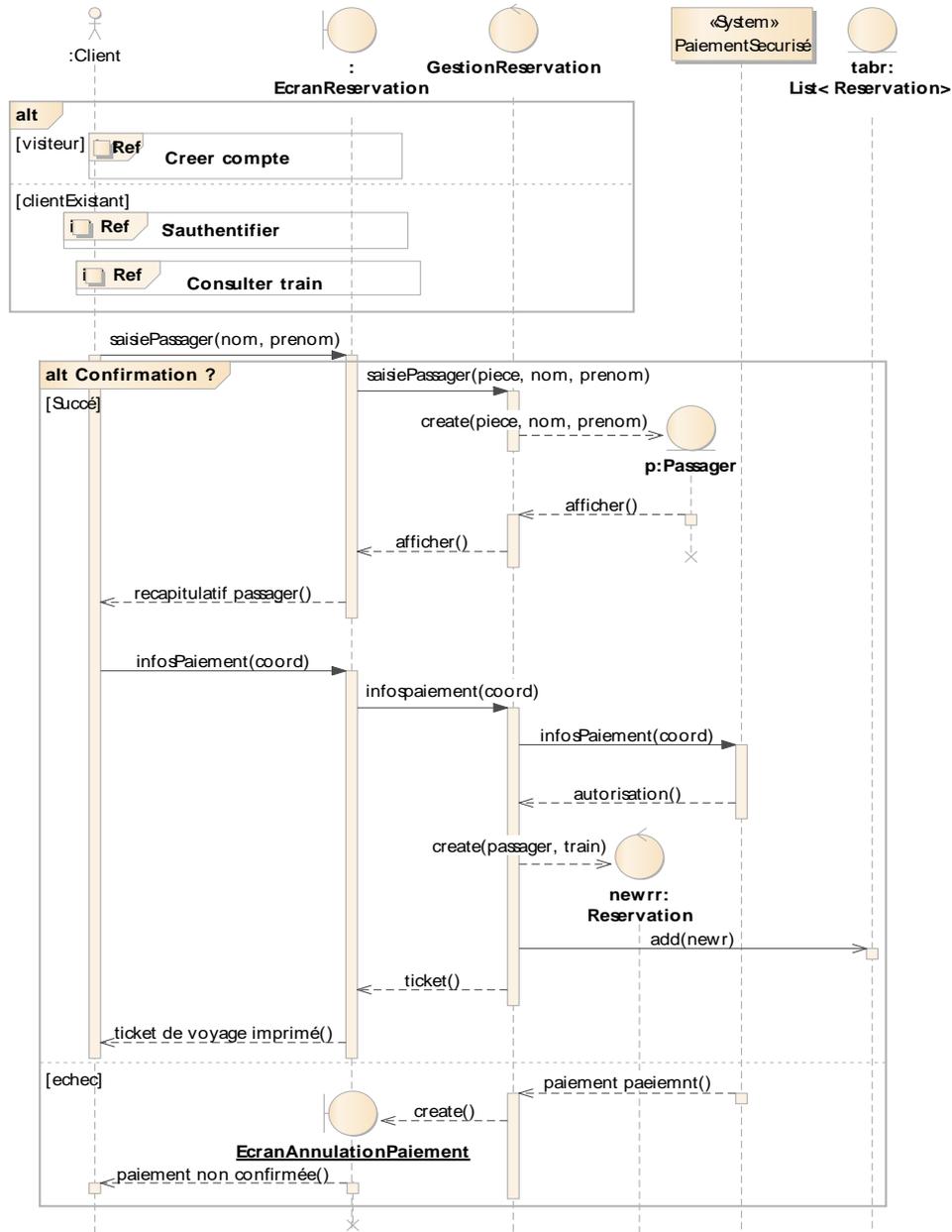


Source : Auteurs

Poursuivant le scénario de recherche de *Trains*, considérons un scénario dans lequel le Client décide de confirmer sa réservation en ligne. Premièrement le «Visiteur» doit être inscrit dans un scénario d'inscription « Ref ; s'inscrire » ou il est déjà client et qu'il s'est authentifié avec succès « Ref : s'authentifier ». Tout ceci après avoir consulté le programme de Train « Ref : consulter programme de Train ». Le contrôleur reçoit une collection d'informations sur le passager, la réservation et la passe à l'entité « Reservation ». Le contrôleur exige la confirmation du paiement et le soumet au système de paiement Mobile. Celui-ci est responsable de l'autorisation ou pas « alt ». Une fois autorise, le Contrôleur transmet un

signal aux entités de valider la réservation.

**Figure 9—Diagramme de séquence détaillée de l’opération système « créer réservation »**

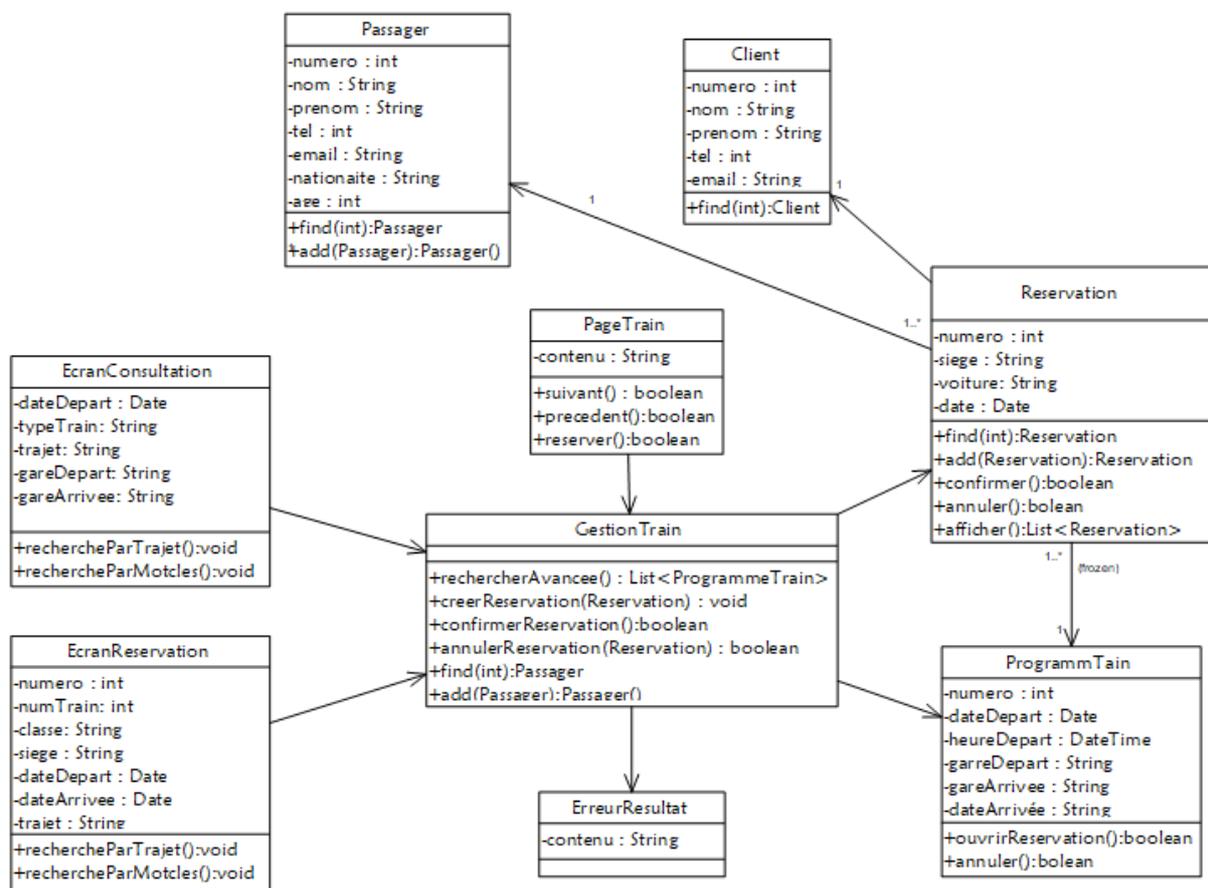


Source : Auteurs

### 3.3. Diagrammes de classes de conception

En partant des classes participantes et de différents messages issus des échanges entre objet illustrés dans le diagramme de séquences, nous complétons le diagramme de classes de conception en ajoutant les **opérations, les types** aux attributs afin de rendre les classes compatibles à l'implémentation technique (Roques, 2008). Nous avons constaté trois dialogues supplémentaires étaient absentes dans le diagramme de classes participantes : le premier correspondant à l'erreur de résultat « ErreurResultat », le second permettant d'afficher le programme de train trouvé et le dernier pour l'annulation du paiement. Le **modèle de classes de conception** obtenu est fourni par la figure 10 suivante

Figure 10—Diagramme de classes de conception



Source : Auteurs

A partir de ce diagramme, l'obtention du code source reste l'apanage de l'outil utilisé et le travail d'implémentation peut commencer.

## Conclusion

Cet article avait pour objectif la conception par approche agile d'un système e-commerce d'innovation de l'activité économique (vente de tickets de train) de la Société Nationale de Chemin de Fer du Congo. L'approche agile adoptée visait une livraison fréquente des incréments en tenant compte des urgences du service demandeur et du délai de l'itération. Nous avons montré qu'il est possible de mixer deux philosophies de développement logiciel notamment la démarche rigide d'UP et la souplesse de l'eXtrme Programming visant la prise en compte en amont des éléments de la phase de programmation notamment des maquettes ou de dialogues Homme Machine. Outre les maquettes, nous avons construit à l'aide de l'outil Entreprise Architect, un ensemble de modèles UML décrivant les différentes vues du système d'e-commerce «PAMASOFT» : le modèle de contexte dynamique pour délimiter les responsabilités et limites du système ; le diagramme de cas d'utilisation contenant les fonctionnalités du système ; le modèle de classes représentant la structure d'objets applicatifs et d'informations manipulées ; un modèle d'interaction décrivant le dynamisme lors du déroulement de différents scénarios de cas d'utilisation. Signalons aussi que notre étude s'est limitée à l'espace du problème c'est-à-dire à la conception détaillée. Le résultat de cette étude vise à promouvoir la recherche dans le domaine de Génie Logiciel qui, se focalise sur l'innovation technologique respectant la qualité, le coût et le délai. Ainsi, le diagramme de classe issu de la phase de conception pourrait également servir à une implémentation sur une cible technique différente, ce qui corrobore avec l'activité principale du domaine d'ingénierie logicielle. La réservation de titres de voyage en ligne est une solution obligée pour toute organisation ayant tiré de leçons de la période de la pandémie (COVID-19). Néanmoins, cela soulève des interrogations : « Que deviendrait le système de contrôle du métier de tickets de train ? », « Quel mécanisme de sécurisation des imprimés de valeurs capable de contrer les fléaux de piratage ? ». Ces questions nécessitent l'attention de recherches futures. Par ailleurs, certaines fonctionnalités n'ont pas fait l'objet d'analyse dans cet article, ce qui pourrait servir de base de recherches futures. Notons également que l'aspect de sécurisation de transactions et l'interopérabilité entre le système « PAMASOFT » proposé et les API (Application Programming Interface) des opérateurs de Banques Mobiles reste non résolu et cela pourrait intéresser les futurs chercheurs.

## BIBLIOGRAPHIE

- Al-Omouh, K. S. (2020). The role of top management support and organizational capabilities in achieving e-business entrepreneurship. *Kybernetes*, 50(5), 1163-1179. <https://doi.org/10.1108/K-12-2019-0851>
- Cahyono, D., Arief, M., Furinto, A., & Tjhin, V. U. (2022). STRATEGIC CHOICE OF CONSUMER PACKAGED GOODS COMPANY TO MAXIMIZE E-BUSINESS FIRM PERFORMANCE : A ROLE OF E-COMMERCE ENABLER AS COLLABORATION. *International Journal of EBusiness and EGovernment Studies*, 14(2), Art. 2.
- Choudhary, P., & Pandita, D. (2022). Agile HR : The Need for the E-Business Environment. *2022 7th International Conference on Business and Industrial Research (ICBIR)*, 274-278. <https://doi.org/10.1109/ICBIR54589.2022.9786501>
- Dekhili 1, S., & Achabou 2, M. A. (2011). La course des entreprises vers la certification environnementale : Quelles conséquences sur la crédibilité des écolabels et la confiance des consommateurs? *Revue management et avenir*, 1, 294-310.
- Djelil, F., Montesinos, M. T. S., & Gilliot, J.-M. (2020). Une approche didactique pour l'introduction de la Programmation Orientée-Objet en classe. *DIDAPRO 8– DIDASTIC*.
- Egholm, C. L., Packness, A., Stokholm, J., Rasmussen, K., Ellervik, C., Simonsen, E., Christensen, A. I., & Jepsen, R. (2020). Questionnaire development for the Lolland-Falster Health Study, Denmark : An iterative and incremental process. *BMC Medical Research Methodology*, 20(1), 52. <https://doi.org/10.1186/s12874-020-00931-1>
- Gâteau, B., Khadraoui, D., & Dubois, E. (2004). Architecture e-business sécurisée pour la gestion des contrats. *3eme Conférence sur la Sécurité et Architectures Réseaux (SAR). La Londe-France*.
- Hamzane, I. (2021). *Proposition d'une approche de transformation digitale de la composante gestion de projets de la gouvernance IT à base de modèles* [PhD Thesis]. Université Hassan II Casablanca (Maroc).
- Kameni, S. M. P. (2023). *Améliorer le processus décisionnel à travers la gestion de la relation gestionnaire-analyste de données dans un contexte d'intelligence d'affaires*.
- Kiv, S., Heng, S., Kolp, M., & Wautelet, Y. (2018). Agile manifesto and practices selection for tailoring software development : A systematic literature review. *International conference on product-focused software process improvement*, 12-30.
- Legras, S. (2014). L'agilité, nouvelle transformation pour l'entreprise. *Documentaliste-Sciences de l'Information*, Vol. 51(4), 4-6. <https://doi.org/10.3917/docs.514.0004>
- Mercanti-Guérin, M. (2022). *E-commerce et pouvoir des plateformes : Quels enjeux environnementaux et éthiques?*



- Neto, J. C., Bento, L. H. T. C., Oliveira Jr, E., & Souza, S. D. R. S. (2021). Are we teaching UML according to what IT companies need? A survey on the São Carlos-SP region. *Anais do Simpósio Brasileiro de Educação em Computação*, 34-43.
- Niranjanamurthy, M., & Chahar, D. (2013). The study of e-commerce security issues and solutions. *International Journal of Advanced Research in Computer and Communication Engineering*, 2(7), 2885-2895.
- Rallet, A. (2001). Commerce électronique ou électronique du commerce ? *Réseaux*, 106(2), 17-72.
- Roques, P. (2008). *UML 2 : Modéliser une application web*. Editions Eyrolles.
- Rukanova, B., Tan, Y.-H., Ubacht, J., Molenhuis, M., Heijmann, F., Bosch, H., Palaskas, Z., Chen, H., Männistö, T., & Ratnasari, A. (2021). ECommerce Platforms Evaluation Framework for Government. *Electronic Government*, 103-116.  
[https://doi.org/10.1007/978-3-030-84789-0\\_8](https://doi.org/10.1007/978-3-030-84789-0_8)
- Ruphin, N. N. (2021). Ingénierie Dirigée par les Modèles pour la Spécification d'un système de paiement de frais scolaire dans des écoles privées de la République Démocratique du Congo. *Revue Internationale du Chercheur*, 2(2).
- Schlömer, I. F. (2022). Agility as a Driver of Digital Transformation-a Literature Review. *Conference on e-Business, e-Services and e-Society*, 238-253.
- Setyowati, W., Widayanti, R., & Supriyanti, D. (2021). Implementation Of E-Business Information System In Indonesia : Prospects And Challenges. *International Journal of Cyber and IT Service Management*, 1(2), 180-188.  
<https://doi.org/10.34306/ijcitsm.v1i2.49>
- Veitaitė, I., & Lopata, A. (2021). Knowledge-based UML use case and UML activity models generation from enterprise model. School of languages case study. *CEUR workshop proceedings: IVUS 2021: Information society and university studies 2021: Proceedings of the 26th international conference on information society and university studies (IVUS 2021), Kaunas, Lithuania, April 23, 2021*, 2915, 112-121.
- Wong, L.-W., Leong, L.-Y., Hew, J.-J., Tan, G. W.-H., & Ooi, K.-B. (2020). Time to seize the digital evolution : Adoption of blockchain in operations and supply chain management among Malaysian SMEs. *International Journal of Information Management*, 52, 101997.
- Yan, M.-R., Tran-Danh, N., & Hong, L.-Y. (2019). Knowledge-based decision support system for improving e-business innovations and dynamic capability of IT project management. *Knowledge Management Research & Practice*, 17(2), 125-136.  
<https://doi.org/10.1080/14778238.2019.1601507>