IUT Nantes
Pôle Sciences et technologie
Nantes Université

SAÉ 401 : Développement d'une application complexe

R4.09 - Management avancé des systèmes d'information : Modélisation BPMN d'une API

Groupe_1_1 : Kyllian Arnaud, Jauzua Destain, Pol Lamothe, Brieuc Le Carluer, Thomas Souchet

14 avril 2025

BUT2 Informatique, année académique 2024-2025

Table des matières

1 Contexte

Dans le cadre de la SAE du semestre 4, il nous a été demandé dans la ressource R4.01, Architecture logicielle, de développer trois micro-services chacun sous la forme d'une API REST. Une API REST (Representational State Transfer) est une architecture d'API qui repose sur le protocole HTTP pour permettre la communication entre clients et serveurs. Elle est basée sur la notion de ressources, identifiées par des URLs, et manipulées à l'aide des méthodes HTTP standard. Une API REST est dite *stateless*, c'est à dire que chaque requête contient toutes les informations nécessaires, et le serveur ne garde pas de mémoire de l'état du client entre les appels.

En tenant compte du cahier des charges de la ressource R4.01, nous avons décidé de développer les trois micro-services suivants :

— Un service de consultation et d'opérations sur des cartes Pokémon reposant sur l'API externe : Pokémon TCG API [1]

— Un service de gestion des utilisateurs et de leur collection de cartes Pokémon

— Un service qui fait office de proxy pour les deux autres, ce service est le point d'entrée de notre application

A partir de ces choix, dans le cadre de la ressource R4.09, Management avancé des systèmes d'information, nous avons modélisé la logique métier des principales fonctionnalités de notre application en utilisant le langage BPMN (Business Process Model and Notation).

2 Modélisation des logiques métier

Dans chacune des fonctionnalités suivantes intervient le concept de jeton d'authentification. En effet, dû au fait qu'une API REST est par définition sans état (*stateless*), pour chaque fonctionnalité l'application a besoin de savoir quel est l'utilisateur qui la contacte. C'est donc à l'utilisateur de fournir cette information, pour cela il va devoir s'authentifier auprès de l'application qui lui renvoie un jeton d'authentification, ce jeton est une chaîne de caractères que l'utilisateur devra fournir pour les requêtes suivantes afin que l'application le reconnaisse.

2.1 Création d'un compte

Ce processus permet à un utilisateur de se créer un compte sur l'application. Pour cela, il doit fournir un identifiant (*login*) qui devra être unique, un pseudo et un mot de passe. L'application vérifie seulement que l'identifiant n'est pas déjà pris et renvoie un jeton d'authentification si le processus est réussi.

1. https://pokemontcg.io/

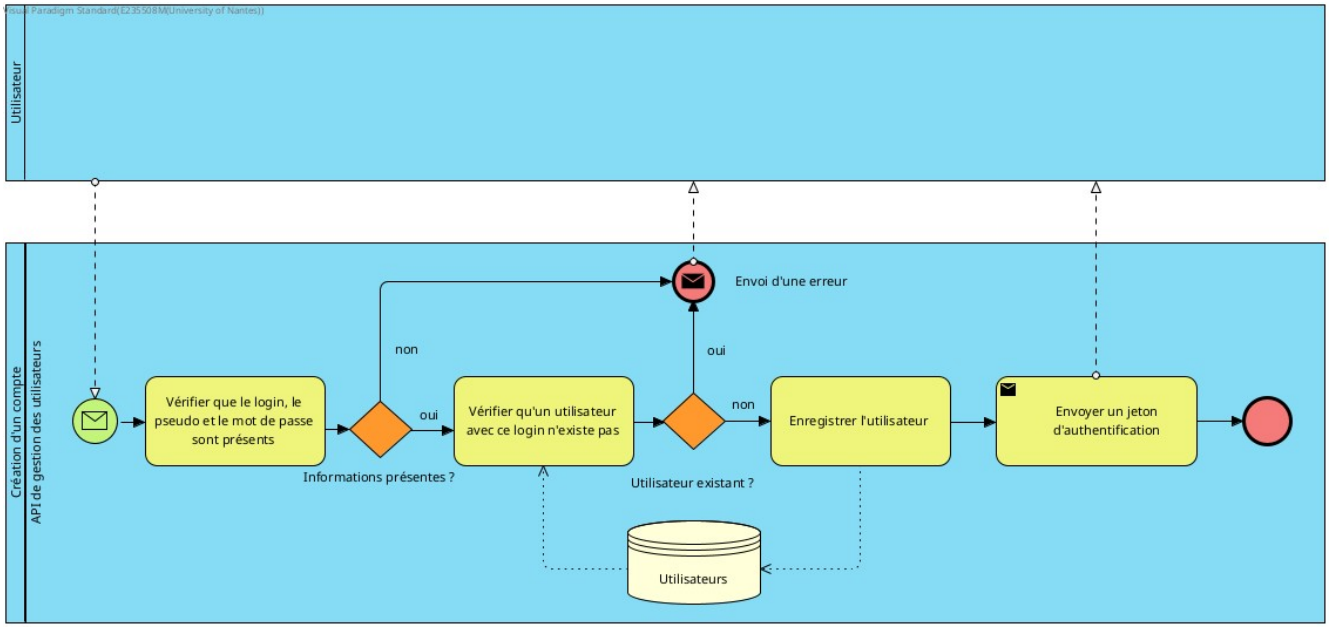

FIGURE 1 – Processus de création d'un compte

2.2 Connexion à un compte

Ce processus permet à un utilisateur possédant un compte de se connecter avec ses informations de connexion (identifiant et mot de passe). L'application lui répond soit avec une erreur si les informations sont invalides soit avec un jeton d'authentification. Ce processus permet aussi à l'utilisateur de fournir simplement un jeton d'authentification précédemment généré pour que l'application l'actualise.

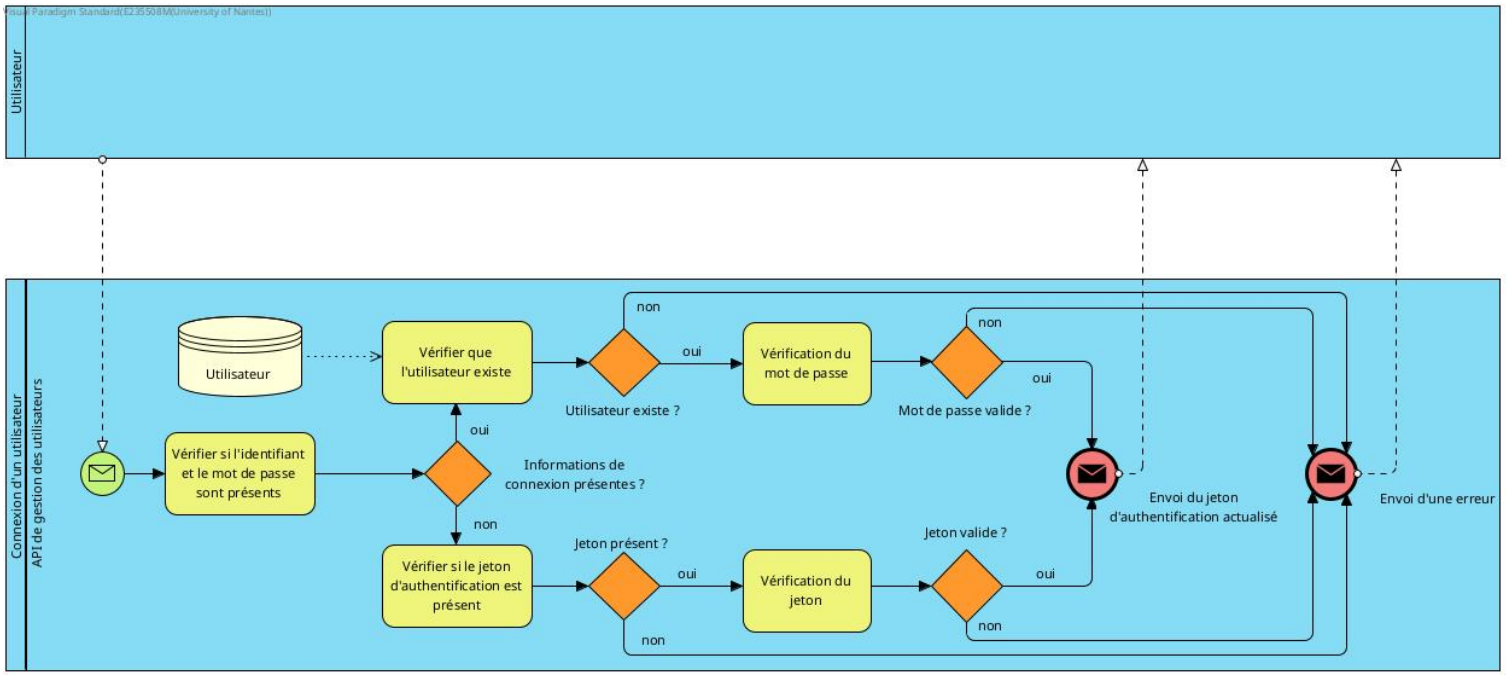

FIGURE 2 – Processus de connexion à un compte

2.3 Voir la collection de cartes

Ce cas d'usage permet à l'utilisateur de voir l'ensemble des cartes Pokémon qu'il possède dans sa collection, pour faire cette demande il suffit qu'il fournisse un jeton d'authentification valide.

L'application commence par vérifier que l'utilisateur est bien connecté grâce au jeton puis elle récupère sa collection de cartes puis elle demande à un deuxième micro-service les informations sur ces cartes (nom, date, image, etc). Ce deuxième micro-service vérifie s'il dispose des informations sur ces cartes sinon il demande à l'API Pokémon externe. Enfin, le processus envoie l'ensemble des informations sur chaque carte de sa collection à l'utilisateur.

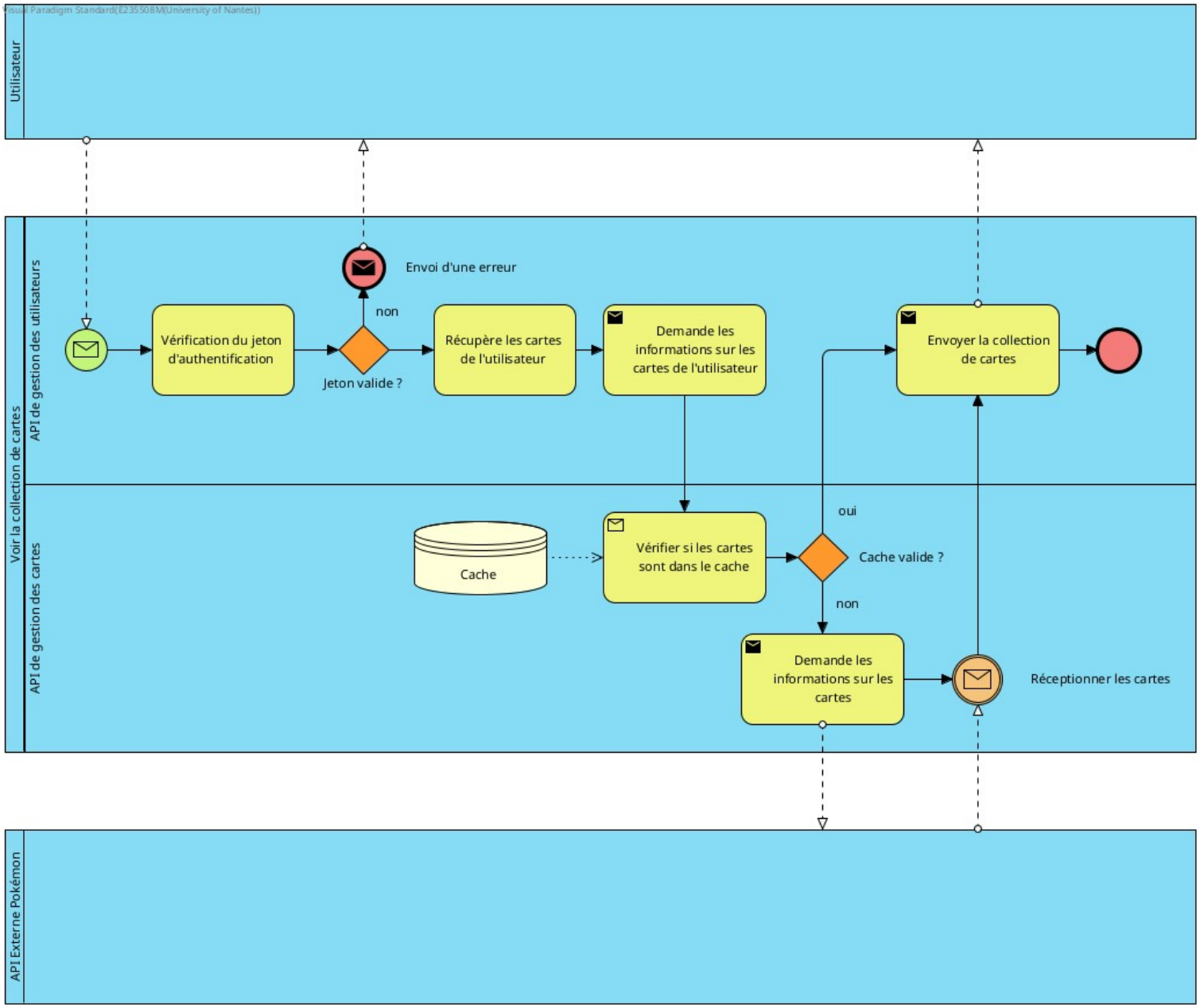

FIGURE 3 – Processus de consultation de la collection de cartes de l'utilisateur

2.4 Ouverture d'un booster de cartes Pokémon

Ce processus permet à un utilisateur d'ajouter des cartes à sa collection. Cela se fait avec l'ouverture de *booster* de cartes Pokémon. Un *booster* est un paquet scellé qui contient plusieurs cartes aléatoires (dans notre cas 5) qui appartiennent à une même série (*set* en anglais).

L'utilisateur commence par choisir la série dans laquelle il veut ouvrir un *booster* en demandant la liste à l'application. Une fois qu'il a choisi la série il demande à l'application d'ouvrir un *booster*, elle lui répond avec les cartes qui étaient contenues dans le paquet.

De son côté, l'application doit à plusieurs moments gérer un système de mise en cache qui permet de limiter le nombre d'appels à l'API externe. Lors de l'ouverture du *booster*, l'application se contente de choisir cinq cartes aléatoirement dans une série et elle les sauvegarde dans la collection de l'utilisateur.

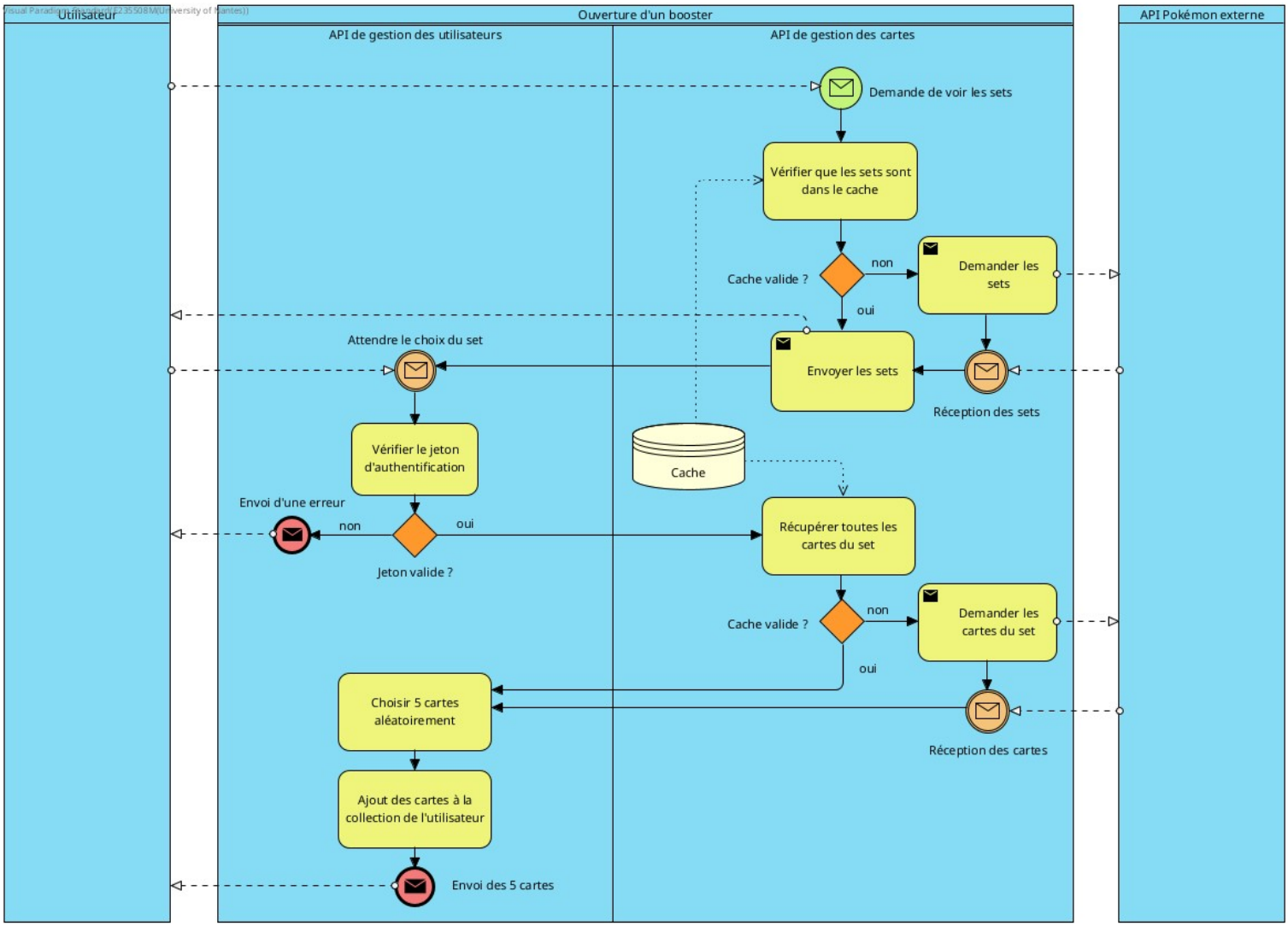

FIGURE 4 – Processus d'ouverture d'un booster

2.5 Mettre à jour les informations d'un compte

Ce cas d'usage permet à l'utilisateur de mettre à jour son pseudo et/ou son mot de passe, il doit fournir un jeton d'authentification et un nouveau pseudo et/ou un nouveau

mot de passe. Si le processus est réussi l'application lui renvoie un jeton d'authentification actualisé.

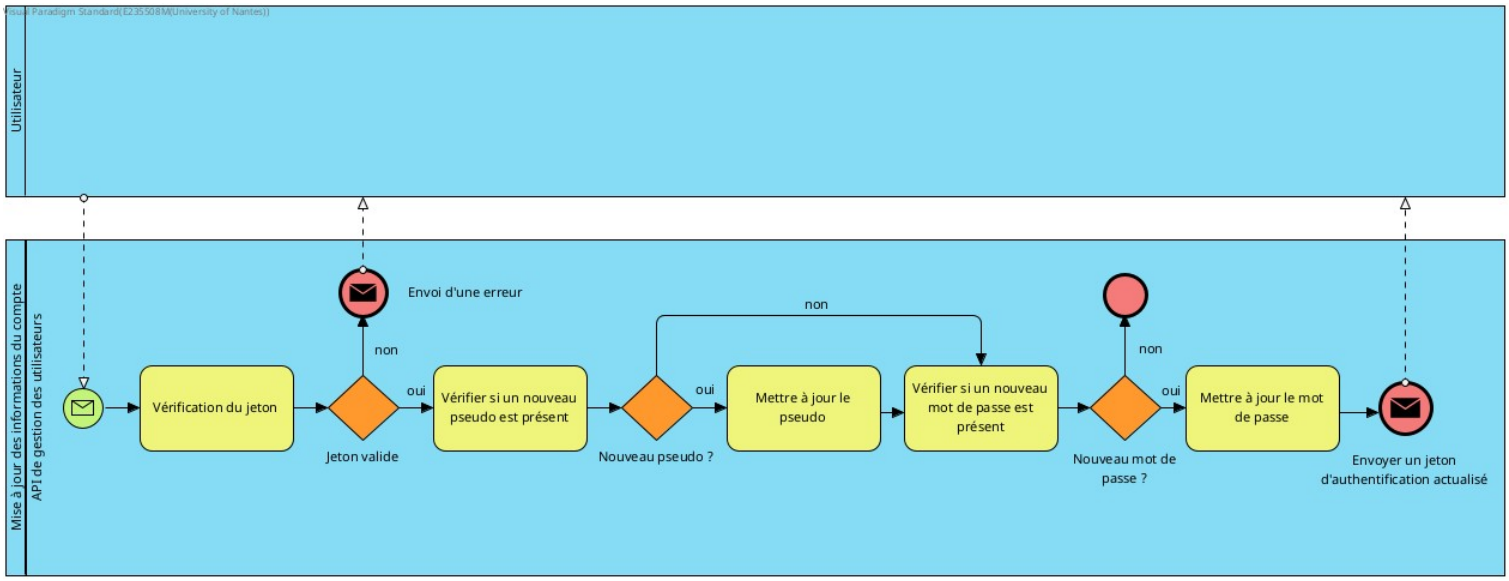

FIGURE 5 – Processus de mise à jour d'un compte