



Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université de Carthage

Institut National des Sciences Appliquées et de Technologie



Spécialité : Réseaux Informatiques et Télécommunications 3

Rapport Projet

Analyse et Conception

Conception de Plateforme Éducative "EduFamily Connect"

Encadré par :

Mme Sonia Bouzid

Elaboré par :

Ouertani Mariem

Belabed Youssef

Meryem Karoui

Ashouri Mohamed

Période : Du 01/10/2025 au 15/12/2025

Soumis le : 15/12/2025

Année Académique : 2025– 2026

REMERCIEMENT

Nous tenons à exprimer nos sincère **gratitude** envers notre professeur pour l'opportunité d'étudier le cas **EduFamily Connect**.

Ce projet, bien que théorique, a constitué un terrain d'exercice extrêmement formateur pour l'application des méthodologies d'**Analyse** et de **Conception Logicielle** basées sur le langage **UML**.

RESUME

Ce rapport de conception présente l'étude approfondie du projet de plateforme éducative **EduFamily Connect**, visant à créer un écosystème numérique complet reliant élèves, parents et enseignants.

La première partie du rapport expose l'**analyse des besoins** fonctionnels, non fonctionnels et techniques, servant de fondement à la conception.

La phase de modélisation (UML) a ensuite été structurée autour de trois axes majeurs :

1. **Analyse des Besoins (Phase 1)** : Identification des acteurs et de leurs interactions via le **Diagramme de Cas d'Utilisation**, et structuration des concepts métier avec le **Diagramme de Classes du Domaine**.
2. **Conception Détailée (Phase 2)** : Modélisation du comportement dynamique du système à travers les **Diagrammes de Séquence** (interactions clés), les **Diagrammes d'État-Transition** (cycle de vie des objets) et les **Diagrammes d'Activité** (processus métier).
3. **Architecture Avancée (Phase 3)** : Définition de l'organisation logique (**Diagramme de Packages**), structurelle (**Diagramme de Composants**) et physique (**Diagramme de Déploiement**) du système.

Ce travail aboutit à une documentation de conception détaillée et structurée, prête à servir de base pour la phase de développement d'**EduFamily Connect**, tout en respectant les **bonnes pratiques** d'ingénierie logicielle.

TABLE DES MATIERES

LISTE DES FIGURES

Figure 1 : Les acteurs

Figure 2 : Diagramme de Cas d'Utilisation d'User

Figure 3 : Diagramme de Cas d'Utilisation d'Enseignant

Figure 4 : Diagramme de Cas d'Utilisation d'Administrateur

Figure 5 : Diagramme de Cas d'Utilisation d'élève

Figure 6 : Diagramme de Cas d'Utilisation de parent

Figure 7 : Diagramme de classes

Figure 8 : Diagramme de soumission et correction par enseignant

Figure 9 : Diagramme d'interaction parent-enseignant

Figure 10 : Diagramme de création d'évaluation

Figure 11 : Diagramme de consultation progression pour parent

Figure 12 : Diagramme de passation d'évaluation

Figure 13 : Diagramme d'interaction utilisateur – forum

Figure 14 : Diagramme de cycle de vie du cours

Figure 15 : Diagramme de cycle de vie d'un devoir

Figure 16 : Diagramme de cycle de vie d'un message

Figure 17 : Diagramme de cycle de vie d'une alerte

Figure 18 : Diagramme de processus d'évaluation

Figure 19 : Diagramme de processus résolution d'exercice

Figure 20 : Diagramme de processus ajout ressources à la banque

Figure 21 : Diagramme de processus de consultation progression

Figure 22 : Architecture logique de projet

Figure 23 : Architecture déploiement de projet

Table des Matières

Partie 1 : Analyse et Spécifications (Cahier de Charge)

1. Introduction et Contexte du Projet	7
2. Présentation et Description de la Plateforme "EduFamily Connect"	7
3. Objectifs du Projet	7
4. Analyse des Besoins Fonctionnels Détaillés (Fiches de Fonctionnalités)	7
5. Analyse des Besoins Non Fonctionnels (BNF)	8
6. Exigences Techniques et Contraintes	9

Partie 2 : Modélisation UML - Analyse des Besoins (Phase 3)

1. Modélisation des Exigences.....	9
1.1 Identification des Acteurs	9
1.2 digrammes des cas d'utilisation légendés selon les acteurs	10
A- User	
B- Enseignant	
C- Administrateur	
D- Elève	
E- Parent	
1.3 description textuelle des 5 cas d'utilisation les plus complexes	20
A- login (user)	
B- Créer des Exercices (enseignants)	
C- Effectuer des exercices (élève)	
D- Suivre progression (parent)	
E- Planifier (Administrateur)	
2. Modèle du Domaine : Diagramme de Classes.....	27
2.1 Identification des Concepts Métier	27
2.2 Définition des Éléments du Modèle	28
• Définir les attributs, opérations et relations	
• Spécifier les cardinalités et contraintes	

2.3 Justification du Modèle et des Relations	29
--	----

Partie 3 : Modélisation UML - Conception Détailée (Phase 2)

1. Modélisation des Interactions : Diagrammes de Séquence Système	29
2. Modélisation des Processus : Diagrammes d'état transition et d'activité	35

Partie 4 : Architecture et Conception Avancée (Phase 3)

1. Architecture Logique : Diagramme de Packages	45
2. Architecture Physique : Diagramme de Composants	47
3. Architecture de Déploiement	47

Partie 5 : Conclusion et Perspectives

1. Conclusion et Bilan de la Conception	48
2. Perspectives d'Évolution et Fonctionnalités Futures	49

Partie 1 : Analyse et Spécifications (Cahier de Charge)

1. Introduction et Contexte du Projet

Le secteur éducatif moderne fait face à un double défi : **maintenir la qualité pédagogique** tout en **digitalisant les interactions** entre les établissements scolaires, les élèves et les familles. Le projet **EduFamily Connect** répond à ce besoin en proposant une **plateforme éducative intégrée**. Elle vise à combler le fossé numérique et communicationnel, en créant un **écosystème unifié** où l'information, le suivi pédagogique et la communication circulent de manière fluide et sécurisée.

2. Présentation et Description de la Plateforme "EduFamily Connect"

EduFamily Connect est une solution logicielle innovante destinée aux écoles, collèges et lycées. Elle se positionne comme le **point central de gestion** de l'ensemble des activités pédagogiques et administratives liées au cursus des élèves. La plateforme est conçue pour être **multi-profil** (Elève, Parent, Enseignant, Administrateur) et s'articule autour de quatre modules principaux : la gestion des cours, les exercices/évaluations, le suivi de progression et l'espace de communication.

3. Objectifs du Projet

- **Digitaliser l'Écosystème** : Transformer les processus d'enseignement, d'apprentissage et de gestion administrative de l'école vers une solution 100% numérique.
- **Faciliter la Communication** : Améliorer significativement la transparence et la rapidité des échanges entre tous les acteurs éducatifs.
- **Assurer un Suivi Personnalisé** : Offrir des outils détaillés pour monitorer la progression individuelle des élèves, identifier les difficultés et adapter l'enseignement.
- **Qualité des données** : Garantir une **fiabilité de 99,9%** dans le calcul du carnet de notes et des moyennes pondérées.

4. Analyse des Besoins Fonctionnels Détaillés (Fiches de Fonctionnalités)

Les besoins fonctionnels se structurent autour des fonctionnalités clés de la plateforme.

- **Gestion des Cours :**
 - Création, modification et archivage de cours structurés en chapitres et leçons.
 - Intégration de ressources multimédias (vidéos, documents, liens externes).
 - Gestion d'un **calendrier interactif** pour les plannings de cours et les échéances.
- **Exercices et Évaluations :**
 - Développement d'un outil de création d'exercices variés (QCM, appariement, questions ouvertes, etc.).
 - Mise en place de la **notation automatique** pour les exercices fermés.
 - Gestion d'une **banque d'exercices** réutilisables et filtrables par compétence.
 - Fonctionnalités pour les évaluations **chronométrées et sécurisées**.
- **Système d'Évaluation et Suivi :**

- Gestion d'un **carnet de notes numérique** détaillé avec application des coefficients.
- Association des notes aux **compétences et objectifs d'apprentissage** (approche par compétences).
- Affichage de **tableaux de bord** de progression personnalisés (pour élèves, parents et enseignants).
- Génération d'**alertes automatiques** en cas de difficulté académique, d'absence ou de retard.
- **Espace de Communication :**
 - Mise à disposition d'une **messagerie interne** sécurisée entre tous les profils utilisateurs.
 - Création de **forums de discussion** spécifiques par matière ou par classe.
 - Diffusion d'**annonces** générales ou ciblées avec notifications *push*.
 - Fonctionnalité de **partage de ressources et documents** entre enseignants et élèves/parents.

5. Analyse des Besoins Non Fonctionnels (BNF)

Les besoins non fonctionnels sont essentiels pour garantir la qualité, la fiabilité et la sécurité de la plateforme.

Performance et Montée en Charge :

- Le temps de réponse de l'application ne doit pas dépasser **2 secondes** pour les interactions courantes.
- Le système doit supporter un nombre élevé d'**utilisateurs simultanés** (ex: lors de l'accès aux notes ou aux devoirs) sans dégradation notable.

Sécurité et Confidentialité :

- Conformité totale avec les réglementations sur la **protection des données personnelles (RGPD/lois locales)**.
- Mise en œuvre d'une **gestion fine des droits d'accès** basée sur le rôle de l'utilisateur (ACL).
- Utilisation du protocole **HTTPS** et chiffrement des données sensibles au repos et en transit.

Disponibilité et Fiabilité :

- Le système doit garantir un taux de **disponibilité de 99,5%** en heures scolaires.
- Mécanismes robustes de **sauvegarde** et de **restauration** des données pour assurer la continuité du service.

Ergonomie et Accessibilité :

- L'interface utilisateur doit être **intuitive, claire et homogène** sur tous les modules.
- Conception **responsive** garantissant une expérience optimale sur ordinateur, tablette et smartphone.
- Conformité aux normes d'accessibilité (ex: WCAG) pour les utilisateurs ayant des besoins spécifiques.

Évolutivité et Maintenabilité :

- L'architecture doit être **modulaire** pour faciliter l'ajout de nouvelles fonctionnalités sans affecter les modules existants.
- Le code doit être documenté et maintenable par une équipe de développement tierce.

6. Exigences Techniques et Contraintes

L'architecture du système devra être **évolutive et robuste**, idéalement basée sur une architecture **microservices** pour faciliter l'ajout de modules futurs.

- **Technologies** : Privilégier un langage de backend moderne et performant (ex: Python/Django ou Java/Spring) et un framework frontend dynamique (ex: React ou Angular).
- **Base de Données** : Utilisation d'une base de données relationnelle (PostgreSQL ou MySQL) pour la fiabilité transactionnelle des données de notes et d'une base NoSQL pour la gestion des ressources multimédias.
- **Déploiement** : Solution hébergée sur un environnement Cloud (ex: AWS, Azure ou GCP) pour garantir la scalabilité et la haute disponibilité.
- **Contrainte réglementaire** : Conformité obligatoire aux lois nationales sur la protection des données personnelles (équivalent du RGPD pour les données éducatives).
- **Fournir des outils d'évaluation diversifiés et adaptatifs**
-

Partie 2 : Modélisation UML - Analyse des Besoins

1. Modélisation des Exigences : Diagramme de Cas d'Utilisation

1.1 Identification des Acteurs

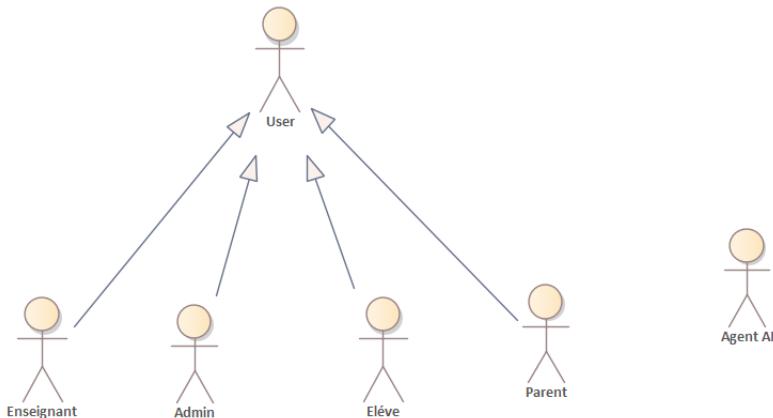


Figure1

Le diagramme présente deux acteurs principaux distincts et un acteur généralisé avec des acteurs spécialisés :

L'acteur principal au sommet est **User** (Utilisateur). C'est un **acteur généralisé** qui représente un rôle commun ou de base.

Les quatre acteurs situés en dessous (**Enseignant**, **Admin**, **Élève**, **Parent**) sont des **acteurs spécialisés** qui héritent des caractéristiques et des interactions de l'acteur **User**.

- **User (Utilisateur)** : Représente l'entité générale qui interagit avec le système.

- **Enseignant (Professeur)** : Est un type d'utilisateur.
- **Admin (Administrateur)** : Est un type d'utilisateur, probablement avec des droits et fonctions de gestion du système.
- **Élève (Étudiant)** : Est un type d'utilisateur.
- **Parent** : Est un type d'utilisateur.

La relation représentée par les flèches pleines allant des acteurs spécialisés vers l'acteur généralisé (**User**) signifie :

- "est un type de" (relation de généralisation/spécialisation ou héritage).
- Cela implique que tout ce qu'un **User** peut faire dans le système, un **Enseignant**, un **Admin**, un **Élève**, ou un **Parent** peut aussi le faire, en plus de leurs actions spécifiques.

L'acteur **Agent AI** (Agent d'Intelligence Artificielle) est positionné séparément. **Agent AI** : Représente un système externe (non humain) qui interagit avec le système modélisé en tant qu'**acteur secondaire**.

1.2 diagrammes des cas d'utilisation légendés

A- Acteur User

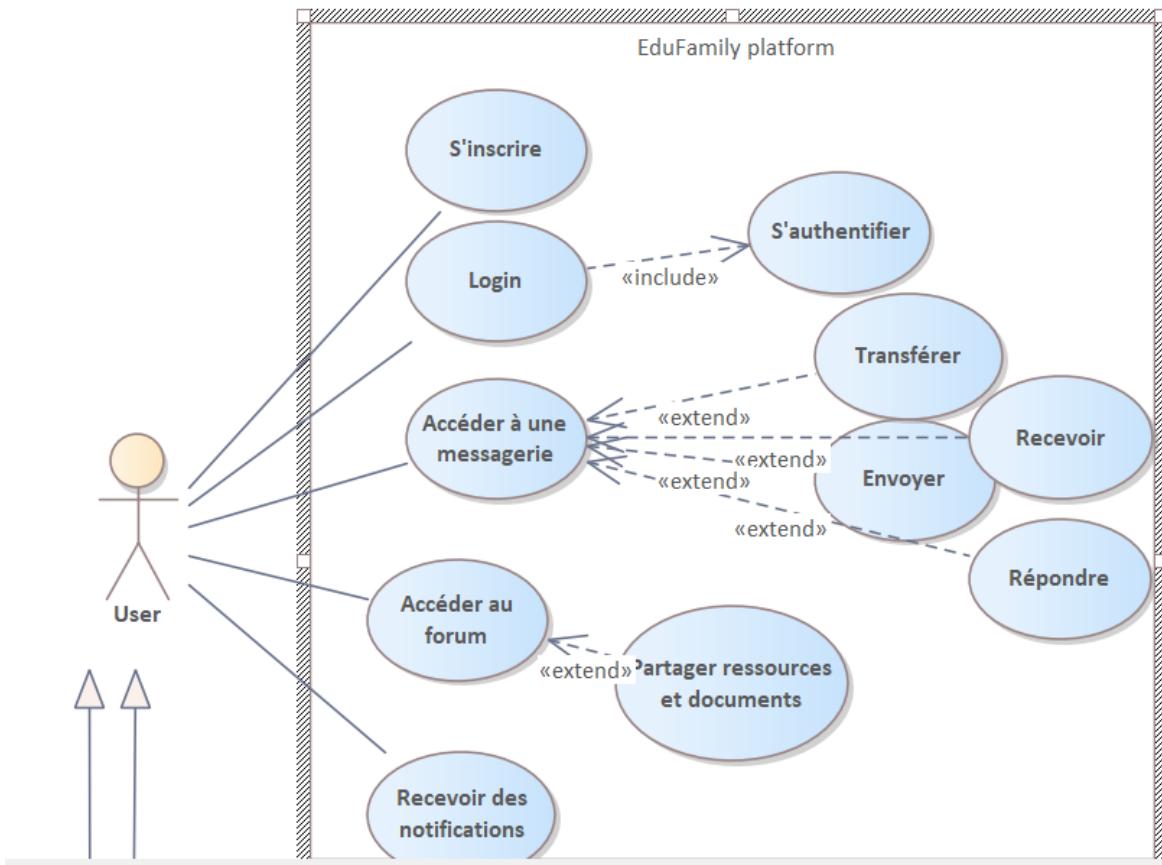


Figure 2 : Diagramme de Cas d'Utilisation d'User

Description du Diagramme de Cas d'Utilisation de User

Acteur Principal

L'acteur unique, **User (Utilisateur)**, représente la généralisation de tous les utilisateurs humains de la plateforme (Enseignant, Élève, Parent, Admin). Cet acteur peut réaliser toutes les actions de base modélisées dans le diagramme.

Fonctionnalités de Gestion de Compte

- **S'inscrire** : Permet au nouvel utilisateur de s'inscrire sur la plateforme.
- **Login** : Permet à l'utilisateur de se connecter au système. Ce cas d'utilisation **inclus** (<<include>>) le cas d'utilisation **S'authentifier**, signifiant que l'authentification est une étape obligatoire et systématique du processus de connexion.
- **S'authentifier** : Processus vérifiant les identifiants de l'utilisateur.

Fonctionnalités de Messagerie et Communication

- **Accéder à une messagerie** : Le cas d'utilisation principal pour interagir avec le module de communication.
Ce cas d'utilisation est **étendu** (<<extend>>) par plusieurs actions spécifiques, qui ne sont pas toujours exécutées :
 - **Envoyer** : Action d'initier et d'expédier un nouveau message.
 - **Recevoir** : Action de consulter les messages entrants.
 - **Répondre** : Action de rédiger une réponse à un message reçu.
 - **Transférer** : Action de renvoyer un message existant à un autre destinataire.
- **Accéder au forum** : Permet à l'utilisateur de consulter et de participer aux forums de discussion.
Ce cas d'utilisation est **étendu** par **Partager ressources et documents**, indiquant que le partage est une action optionnelle effectuée dans le contexte de l'accès au forum (ou potentiellement à la messagerie, bien que la flèche pointe spécifiquement vers le forum).

Fonctionnalités de Notification

- **Recevoir des notifications** : Permet à l'utilisateur d'être informé des événements importants du système (alertes, annonces, nouveaux messages, etc.).

B- Acteur Enseignant

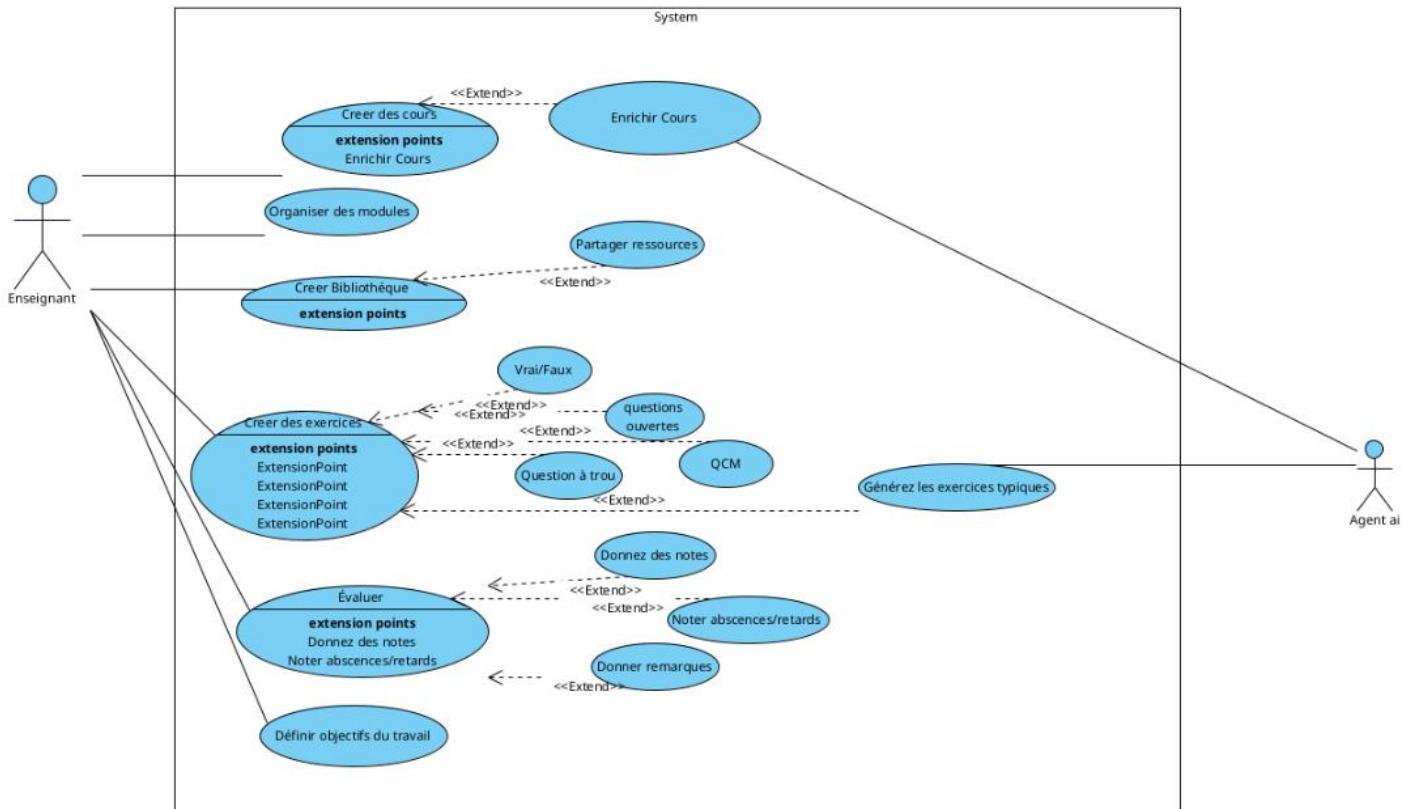


Figure 3 : Diagramme de cas d'utilisation Enseignant

Description du Diagramme de Cas d'Utilisation :

Rôle de l'Enseignant Ce diagramme de cas d'utilisation modélise les interactions spécifiques de l'acteur Enseignant avec le système, se concentrant sur les fonctionnalités de gestion de contenu pédagogique, de création d'évaluations et de suivi des élèves. L'interaction avec un Agent IA pour l'assistance dans la génération d'exercices est également modélisée.

Acteurs

Acteur	Rôle
Enseignant	L'acteur principal. Il est responsable de la création et de l'organisation du contenu, de la conception des évaluations, de l'évaluation des élèves (notation, suivi de présence) et de la planification des objectifs.
Agent IA	Un acteur secondaire (entité logicielle) dont le rôle est d'interagir avec le système pour assister l'enseignant , notamment en automatisant ou en facilitant la génération de certains types d'exercices.

Fonctionnalités de Gestion du Contenu et des Ressources

- L'acteur **Enseignant** dispose de plusieurs cas d'utilisation principaux pour organiser et partager le matériel pédagogique :
- Créer des cours** : Action fondamentale pour mettre en place une nouvelle matière ou un sujet d'étude.

- **Relation d'extension (<<extend>>)** : Peut être étendu par **Enrichir Cours**, ce qui signifie que l'enrichissement est une fonctionnalité **optionnelle** qui peut être déclenchée après ou pendant la création du cours.
- **Organiser des modules** : Permet de structurer le contenu d'un cours en unités logiques (modules, chapitres, leçons).
- **Créer bibliothèque** : Permet de constituer un répertoire de ressources et de documents.
 - **Relation d'extension (<<extend>>)** : Est étendu par **Partager ressources**, ce qui implique que l'action de partager est une étape **optionnelle** qui peut suivre la création de la bibliothèque.

Fonctionnalités de Création d'Évaluations

- L'**Enseignant** utilise le cas d'utilisation principal **Créer des exercices** pour concevoir des évaluations. Ce CU est étendu (<<extend>>) par les formats d'exercices, indiquant qu'ils sont des ajouts **optionnels** et spécifiques au processus de création :

Extension Optionnelle	Description
Vrai/Faux	Format d'exercice binaire.
Questions ouvertes	Format d'exercice nécessitant une réponse textuelle ou développée.
Question à trou	Format d'exercice avec des espaces à compléter.
QCM	Format d'exercice à choix multiples.

- **Interaction avec l'Agent IA** : Le cas d'utilisation **Générez les exercices typiques** est relié aux formats d'exercices **Question à trou** et **QCM** par une relation d'inclusion (<<include>>), et cet CU est réalisé par l'acteur **Agent IA**. Cela suggère que l'Agent IA est responsable de l'**automatisation de la génération** de ces types d'exercices (Question à trou et QCM) pour l'Enseignant.

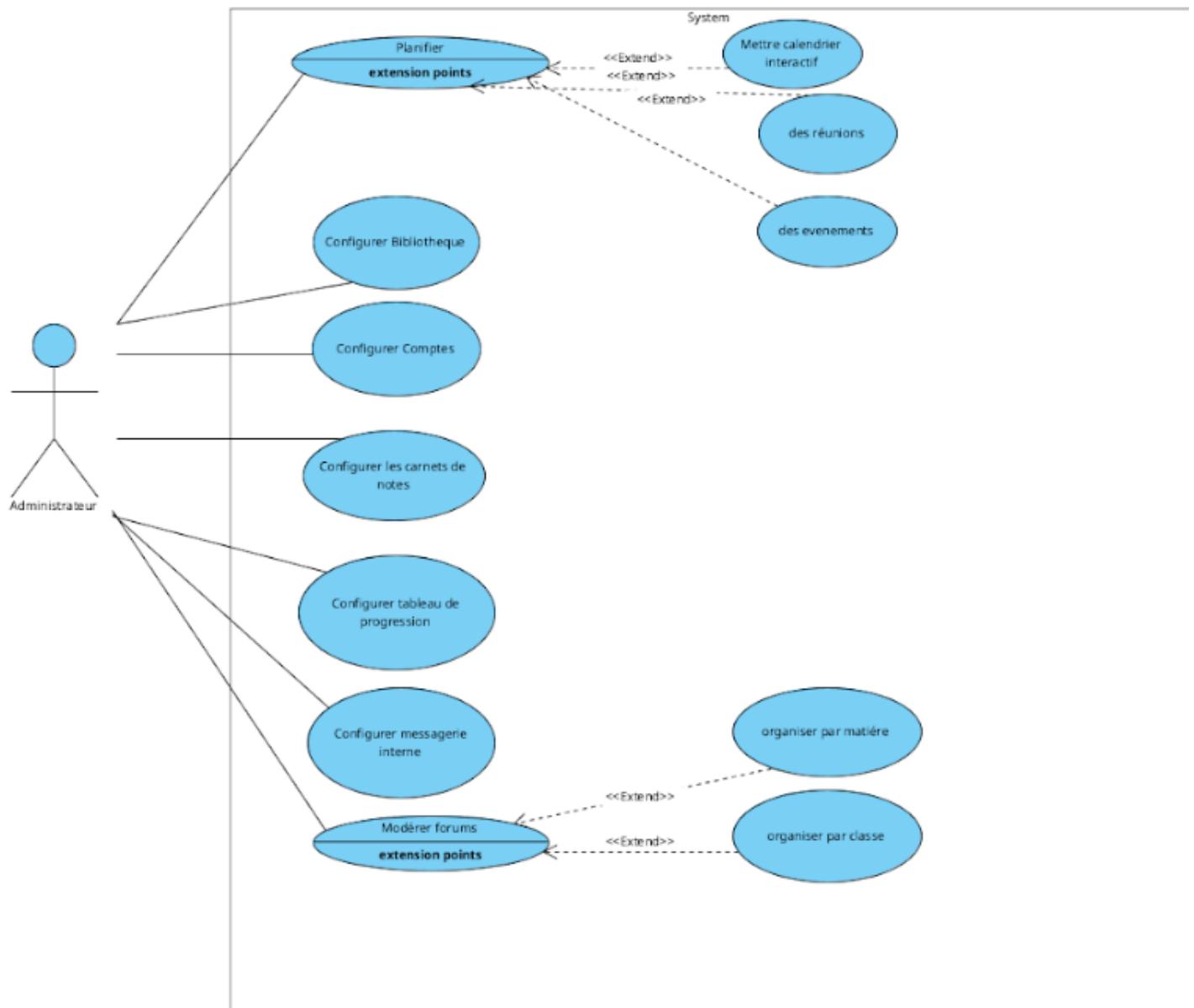
Fonctionnalités d'Évaluation et de Suivi

Le cas d'utilisation principal **Évaluer** regroupe les tâches de notation et de suivi des élèves. Il est étendu (<<extend>>) par des actions spécifiques, qui sont des étapes **optionnelles** du processus d'évaluation :

Extension Optionnelle	Description
Donner des notes	Attribution de scores ou de notes à un travail ou une performance.
Noter absences/retards	Enregistrement des informations de présence et de ponctualité.
Donner remarques	Ajout de commentaires ou de feedback qualitatifs.

- **Planification Pédagogique** : Le CU **Définir objectifs du travail** est une action distincte pour l'Enseignant, relative à l'association des évaluations à des objectifs pédagogiques.

Le diagramme est complet et bien structuré pour représenter l'éventail des interactions de l'Enseignant avec le système éducatif.



C- Acteur Administrateur

Figure 4 : Diagramme de cas d'utilisation d'Administrateur

Description du Diagramme de Cas d'Utilisation : Rôle de l'Administrateur

Ce diagramme de cas d'utilisation modélise les interactions spécifiques de l'acteur **Administrateur** (Admin) avec le système, se concentrant sur les fonctions de **supervision**, de **configuration** et de **gestion globale** des outils et de l'infrastructure de la plateforme.

Acteurs

Acteur	Rôle
--------	------

Administrateur	L'acteur principal. Ce rôle possède les permissions nécessaires pour configurer l'infrastructure de la plateforme, gérer les outils de suivi, planifier les événements et superviser les communications.
-----------------------	---

Fonctionnalités de Configuration et de Planification

L'Administrateur gère l'organisation structurelle et temporelle de la plateforme :

- **Planifier** : Cas d'utilisation principal pour la gestion des agendas et des événements.
 - **Relations d'extension (<<extend>>)** : Les actions suivantes sont des fonctionnalités **optionnelles** du processus de planification :
 - **Mettre calendrier interactif** : Mettre en place et gérer l'outil de calendrier central.
 - **des réunions** : Planifier et enregistrer les réunions.
 - **des événements** : Planifier et enregistrer des événements divers.
- **Configurer Bibliothèque** : Gérer les paramètres et l'organisation de la bibliothèque de ressources globale de la plateforme.

Fonctionnalités de Gestion des Utilisateurs et des Données

- L'Admin est responsable de la mise en place et de la maintenance des éléments fondamentaux du système de gestion pédagogique :
- **Configurer Comptes** : Gérer les comptes utilisateurs (création, modification, droits d'accès) pour tous les acteurs du système (enseignants, élèves, etc.).
- **Configurer les carnets de notes** : Définir la structure, les paramètres et les règles du système de notation.
- **Configurer tableau de progression** : Mettre en place et paramétrier les tableaux de bord et les outils de suivi de la progression des élèves.

Fonctionnalités de Gestion des Outils de Communication

- L'Administrateur supervise les outils d'échange et de discussion pour garantir le bon fonctionnement :
- **Configurer messagerie interne** : Paramétrier et maintenir le système de messagerie entre les utilisateurs.
- **Modérer forums** : Cas d'utilisation principal pour la gestion des espaces de discussion.
 - **Relations d'extension (<<extend>>)** : Les options d'organisation suivantes sont des fonctionnalités **optionnelles** de la modération (ou de la configuration initiale) des forums :
 - **organiser par matière** : Structurer les forums selon les sujets ou les disciplines.
 - **organiser par classe** : Structurer les forums selon les groupes d'élèves.

D- Acteur Elève

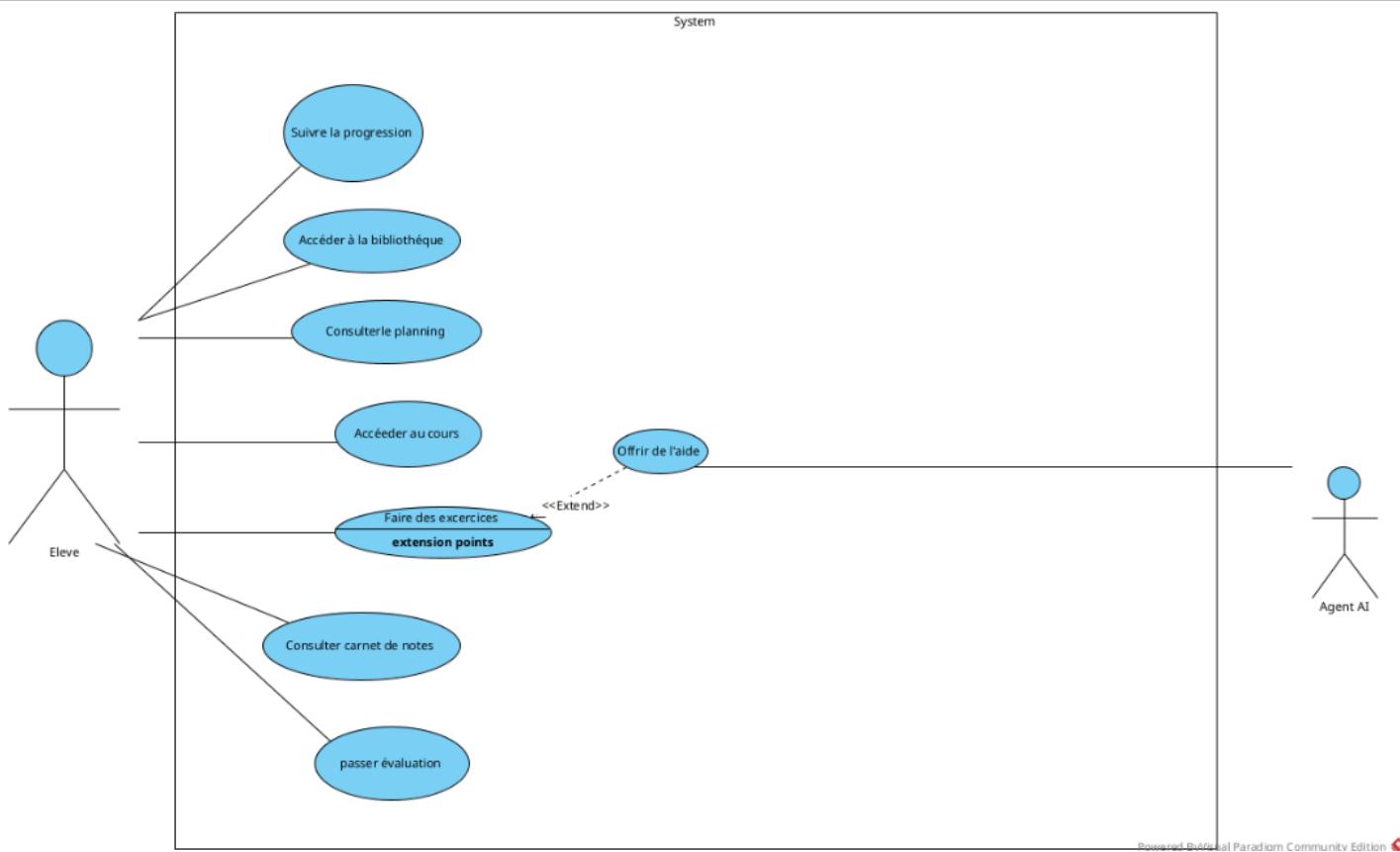


Figure 5 : Diagramme de Cas d'Utilisation d'élève

Powered by UMLparadigm Community Edition

Description du Diagramme de Cas d'Utilisation : Rôle de l'Élève

Ce diagramme de cas d'utilisation illustre les interactions spécifiques de l'acteur **Élève** avec la plateforme éducative, axées sur **l'apprentissage**, la **consultation des ressources**, le **passage d'évaluations** et la **réception d'assistance** grâce à un Agent IA.

Acteurs

Acteur	Rôle
Élève	L'acteur principal. Il représente l'utilisateur apprenant qui interagit directement avec le contenu pédagogique, consulte les informations de suivi (notes, planning), et réalise les activités et les évaluations.
Agent AI	Acteur secondaire. Il intervient pour automatiser ou assister l'acteur principal, fournissant ici une fonctionnalité d'aide contextuelle pour les exercices.

Fonctionnalités d'Apprentissage et d'Accès aux Ressources

L'**Élève** utilise la plateforme pour accéder aux outils d'apprentissage et aux ressources mis à sa disposition :

- **Accéder aux cours** : Action de consulter les supports de cours (leçons, documents) mis en ligne par les enseignants.
- **Accéder à la bibliothèque** : Consulter et utiliser les ressources documentaires et supplémentaires disponibles sur la plateforme.
- **Faire des exercices** : Réaliser les activités d'entraînement et les devoirs assignés.
 - **Relation d'extension (<<extend>>)** : Ce CU est étendu par **Offrir de l'aide**, ce qui signifie que l'aide est une fonctionnalité **optionnelle** que l'élève peut demander pendant la réalisation des exercices.
- **Passer évaluation** : Réaliser des tests formels, des examens ou des quizz pour être noté.

Fonctionnalités de Suivi et de Consultation

Ces cas d'utilisation permettent à l'élève de gérer son apprentissage et de suivre ses performances :

- **Consulter le planning** : Visualiser les horaires de cours, examens, et événements scolaires.
- **Consulter carnet de notes** : Accéder à ses résultats, moyennes et remarques des enseignants.
- **Suivre la progression des études** : Afficher une vue synthétique de son évolution académique (notes, compétences acquises, taux de réussite, etc.).

Fonctionnalités de Communication et d'Alerte

Ces cas d'utilisation permettent à l'élève de suivre son parcours académique, son emploi du temps et ses performances :

- **Suivre la progression** : Consulter un tableau de bord ou un rapport pour visualiser son évolution académique globale (compétences maîtrisées, taux de réussite).
- **Consulter le planning** : Visualiser l'emploi du temps, les échéances des devoirs, les dates d'exams et autres événements importants.
- **Consulter carnet de notes** : Accéder à ses résultats détaillés, aux notes obtenues et aux remarques associées.

Interaction avec l'Agent AI

L'**Agent AI** intervient pour fournir une assistance intelligente à l'**Élève** :

- **Offrir de l'aide** : L'**Agent AI** est directement associé à ce cas d'utilisation, qui est une extension du CU **Faire des exercices**. Cela indique que l'**Agent AI** est responsable de fournir l'aide contextuelle

demandée, que ce soit des indices, des explications, ou des rappels de cours, pour l'assister pendant ses exercices.

Cette intégration de l'Agent AI met en évidence une approche d'apprentissage **assisté**, où un support est disponible pour l'Élève au moment précis où il en a besoin pour surmonter une difficulté pendant son entraînement.

E- Acteur Parent

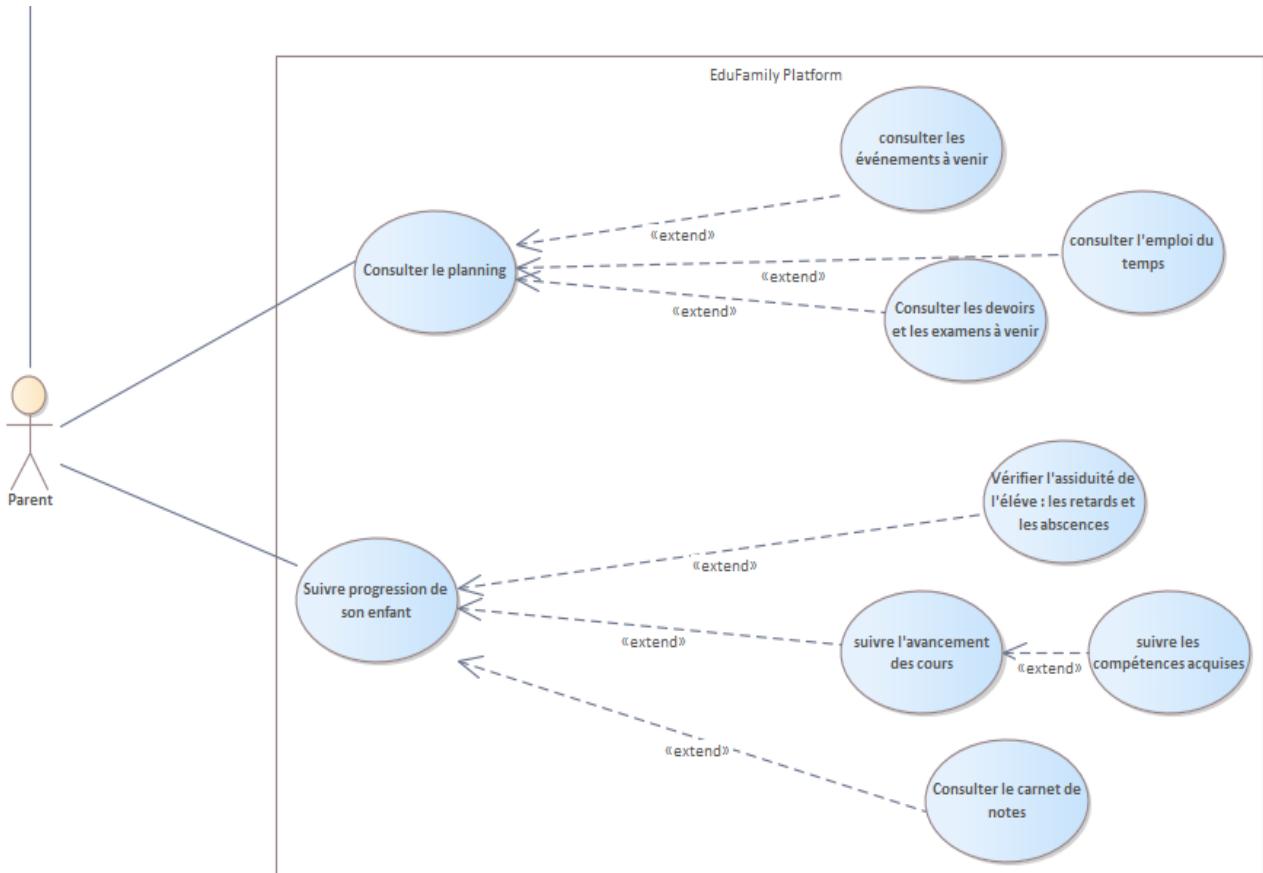


Figure 6 : Diagramme de Cas d'Utilisation de parent

Description du Diagramme de Cas d'Utilisation : Rôle du Parent

Ce diagramme de cas d'utilisation illustre les interactions de l'acteur **Parent** avec la plateforme éducative **EduFamily Platform**.

Le parent y joue un rôle essentiel de **suivi, d'encadrement et de consultation** des informations liées à la scolarité de son enfant.

Les fonctionnalités sont principalement orientées vers la supervision du parcours académique et la gestion de l'emploi du temps de l'élève.

Acteurs

Acteur	Rôle
Parent	Acteur principal. Il représente le tuteur ou responsable légal de l'élève. Son rôle est de consulter les informations relatives à la progression scolaire, à la présence et à la planification des activités de son enfant.

Fonctionnalités de Consultation et de Planification

Le parent dispose d'un accès aux éléments organisationnels de la scolarité de l'élève :

- **Consulter le planning** : Cas d'utilisation principal permettant d'afficher l'ensemble des activités et emplois du temps.
 - « **extend** » **Consulter les devoirs et les examens à venir** : permet d'anticiper les échéances scolaires.
 - « **extend** » **Consulter les événements à venir** : donne accès aux activités extrascolaires et réunions importantes.
 - « **extend** » **Consulter l'emploi du temps** : visualise la répartition quotidienne et hebdomadaire des cours.

Fonctionnalités de Suivi et de Progression de l'Enfant

Le parent a la possibilité de suivre en détail le parcours éducatif de son enfant :

- **Suivre la progression de son enfant** : Cas d'utilisation central regroupant plusieurs sous-fonctions de suivi.
 - « **extend** » **Vérifier l'assiduité de l'élève : retards et absences** : consulter les présences et la ponctualité.
 - « **extend** » **Suivre l'avancement des cours** : connaître les chapitres et le contenu pédagogique abordés.
 - « **extend** » **Suivre les compétences acquises** : évaluer les connaissances déjà maîtrisées par l'élève.
 - « **extend** » **Consulter le carnet de notes** : visualiser les résultats, appréciations et remarques des enseignants.

Le **Parent** agit comme un **observateur actif** du parcours scolaire de son enfant à travers :

- la **consultation du planning** (cours, examens, événements),
- le **suivi des résultats et compétences**,
- et la **vérification de l'assiduité et de la progression**.

Ces fonctionnalités offrent une visibilité complète sur la scolarité et facilitent la communication entre la famille et l'établissement éducatif.

1.3 description textuelle des 5 cas d'utilisation les plus complexes

Cas d'Utilisation : Login (S'authentifier)

1. Identification du Cas d'Utilisation

- **Nom du CU Principal :** Login
- **CU Inclus :** S'authentifier (<<include>>)
- **Acteur Primaire :** User (Utilisateur général : Enseignant, Élève, Parent, Admin).

2. Objectif

Permettre à un utilisateur enregistré de prouver son identité, d'obtenir l'accès aux fonctionnalités de la **EduFamily Platform** correspondant à son rôle, et d'établir une session sécurisée.

3. Pré-conditions

- L'utilisateur doit disposer d'un **compte actif** (i.e., il doit avoir effectué **Créer Compte** et son compte ne doit pas être désactivé par l'Administrateur).

4. Scénario Principal (Login Réussi)

1. L'**Utilisateur** accède à la page de connexion de la plateforme.
2. Le **Système** affiche le formulaire de connexion (identifiant / mot de passe).
3. L'**Utilisateur** saisit son identifiant et son mot de passe, puis valide la connexion.
4. Le **Système** appelle le cas d'utilisation **S'authentifier**.
5. Le **Système** vérifie si les informations saisies correspondent à un compte actif dans la base de données.
6. Si la vérification est réussie, le **Système** crée un **jeton de session sécurisé**.
7. Le **Système** redirige l'**Utilisateur** vers son **tableau de bord personnalisé** en fonction de son rôle (Enseignant, Élève, etc.).
8. Le **Système** enregistre l'heure et la date de la connexion.

5. Scénarios Alternatifs

A. Mot de Passe Oublié (Point 2)

- 2a. L'**Utilisateur** clique sur le lien "Mot de passe oublié".
 - 2a1. Le **Système** demande à l'**Utilisateur** son adresse e-mail ou son identifiant.
 - 2a2. Le **Système** envoie un lien sécurisé et temporaire à l'adresse e-mail enregistrée.
 - 2a3. L'**Utilisateur** clique sur le lien, et le **Système** lui permet de définir un nouveau mot de passe.
 - 2a4. L'utilisateur est invité à se connecter à nouveau (retour au point 1).

B. Identifiants Invalides (Point 5)

- 5a. Si les identifiants ne correspondent à aucun compte actif :

- 5a1. Le **Système** affiche un message d'erreur générique : "Identifiant ou mot de passe invalide."
- 5a2. L'**Utilisateur** est invité à réessayer (retour au point 3) ou à utiliser la fonction "Mot de passe oublié".
- 5a3. Le **Système** incrémente le compteur de tentatives de connexion échouées.

6. Post-conditions

- Une **session sécurisée** est établie.
- L'Utilisateur est **authentifié** et a accès aux fonctions de son rôle.
- L'Utilisateur est redirigé vers l'interface appropriée.

Cas d'Utilisation : Créer des Exercices

1. Identification du Cas d'Utilisation

Nom du CU Principal	Créer des Exercices
CU Étendus (<<extend>>)	Questions ouvertes, QCM, Vrai/Faux, Questions à trous
Acteur Primaire	Enseignant

2. Objectif

Permettre à l'**Enseignant** de concevoir et de publier une évaluation ou un travail pour ses cours. Le système doit offrir la flexibilité de choisir parmi différents formats de questions pour construire l'exercice.

3. Pré-conditions

- L'Enseignant doit être **authentifié** (avoir effectué le Login).
- L'Enseignant doit avoir préalablement **Créer des cours** et potentiellement **Organiser des modules** pour y rattacher l'exercice.
- L'Enseignant doit avoir les droits d'écriture sur le cours sélectionné.

4. Scénario Principal (Création d'Exercice Réussie)

1. L'Enseignant accède à l'interface de gestion de ses cours.
2. L'Enseignant sélectionne un cours et choisit l'option "**Créer des Exercices**".
3. Le Système demande à l'Enseignant de définir les paramètres généraux de l'exercice (Titre, Instructions, Poids/Coefficient, Date limite, Visibilité).
4. L'Enseignant valide les paramètres et entre dans l'interface de conception des questions.
5. Pour chaque question, l'Enseignant sélectionne un format de question (ceci déclenche un des CU étendus, par exemple, le CU **QCM**).
6. Le Système affiche le formulaire spécifique au format choisi (ex : pour QCM, champs pour l'énoncé, les options de réponse, et la/les bonne(s) réponse(s)).
7. L'Enseignant saisit les informations de la question et les valide.
8. L'Enseignant ajoute d'autres questions (retour au point 5) ou termine la conception.

9. L'Enseignant choisit de sauvegarder l'exercice comme Brouillon ou de le **Publier**.
10. Le Système enregistre l'exercice dans la base de données et, si l'exercice est publié, il le rend visible aux élèves du cours concerné.

5. Scénarios Alternatifs

A. Choix du Format QCM (Point 5)

- 5a. L'Enseignant sélectionne le format **QCM**.
 - 5a1. Le Système exige l'ajout d'au moins deux options de réponse.
 - 5a2. Le Système exige l'identification d'au moins une bonne réponse.
 - 5a3. Le Système permet de configurer le mode de correction (choix unique ou multiple).
 - 5a4. Si l'Enseignant ne définit aucune bonne réponse, le Système affiche un message d'erreur et bloque la sauvegarde de la question.

B. Choix du Format Questions Ouvertes (Point 5)

- 5b. L'Enseignant sélectionne le format **Questions ouvertes**.
 - 5b1. Le Système propose un champ pour une grille de notation ou des critères d'évaluation (associé à Définir les objectifs du travail).
 - 5b2. Le Système marque la question comme nécessitant une **correction manuelle** par l'Enseignant (non corrigible par l'Agent IA).

6. Post-conditions

- Un nouvel enregistrement d'exercice est créé dans le système.
- L'exercice est soit à l'état **Brouillon** (invisible aux élèves), soit à l'état **Publié** (accessible selon la date de début).
- L'exercice est rattaché au cours sélectionné et est prêt pour la phase d'**Évaluation**.

Cas d'Utilisation : Faire des exercices

1. Identification du Cas d'Utilisation

- **Nom du CU** : Faire des exercices
- **Acteur Principal** : Élève
- **Acteurs Secondaires** : Agent AI (optionnel – aide à la correction ou feedback)

2. Objectif

Permettre à l'élève d'accéder aux exercices disponibles sur la plateforme **EduFamily**, de les résoudre et d'obtenir une correction automatique ou assistée par l'Agent AI.

3. Préconditions

- L'élève doit être connecté à la plateforme.
- Des exercices doivent être disponibles pour son niveau et sa matière

4. Scénario Principal (Chemin nominal)

1. L'élève se connecte à son espace personnel.
2. Il choisit la section « **Exercices** » dans le menu principal.
3. Le système affiche la liste des exercices disponibles (par matière, chapitre, ou enseignant).
4. L'élève sélectionne un exercice à effectuer.
5. Le système affiche les consignes et les questions correspondantes.
6. L'élève saisit ses réponses et clique sur « **Soumettre** ».
7. Le système enregistre la soumission.
8. L'Agent AI (ou le système) corrige automatiquement les réponses.
9. Le système affiche la **note obtenue**, les **réponses correctes et incorrectes**, ainsi que les **explications**.
10. Le système enregistre le résultat dans le profil de l'élève.

5. Scénarios Alternatifs

A. Aucun exercice disponible (étape 3)

- Le système affiche un message : « Aucun exercice n'est disponible pour le moment. »
- L'élève peut choisir une autre matière ou quitter la section.

B. Soumission incomplète (étape 6)

- Le système détecte des réponses manquantes et demande confirmation avant de soumettre.

C. Problème de connexion (étape 7)

- Si la connexion est perdue, les réponses sont sauvegardées localement et pourront être soumises plus tard.

6. Post-conditions

- Les résultats et commentaires sont enregistrés dans le profil de l'élève.
- L'élève peut consulter ses performances et ses corrections.

Cas d'utilisation : suivre progression

1. Identification du Cas d'Utilisation

Nom du CU Principal : Suivre progression de son enfant

CU Étendus :

- Vérifier l'assiduité de l'élève : les retards et les absences (<<extend>>)
- Suivre l'avancement des cours (<<extend>>)
- Consulter le carnet de notes (<<extend>>)
- Suivre les compétences acquises (<<extend>> de "Suivre l'avancement des cours")

Acteur Primaire : Parent

2. Objectif

Permettre au **Parent** de disposer d'une vue synthétique et détaillée de l'évolution académique et comportementale de son enfant au sein de l'établissement, en s'appuyant sur les données de l'ensemble de la plateforme (notes, assiduité, couverture du programme).

3. Pré-conditions

- Le **Parent** doit être connecté (CU Login réussi).
- Le **Parent** doit être correctement lié à l'élève dans le système.
- Des données de progression (notes, absences, avancement des cours) doivent être enregistrées et publiées par les enseignants/l'administration dans la plateforme.

4. Scénario Principal (Consultation réussie)

1. Le **Parent** se connecte à la plateforme et accède à son tableau de bord.
2. Le Système affiche le tableau de bord personnalisé du **Parent**.
3. Le **Parent** clique sur l'option "Suivre progression de son enfant" (ou accède au tableau de bord de progression dédié).
4. Le Système présente un tableau de bord global de la progression de l'enfant.
5. Le **Parent** sélectionne l'un des indicateurs ou modules de suivi pour obtenir plus de détails (ex: notes, absences, avancement).
 - *Option 1* : Le **Parent** choisit de **Consulter le carnet de notes**. Le Système affiche le carnet de notes détaillé (avec coefficients et moyennes).
 - *Option 2* : Le **Parent** choisit de **Vérifier l'assiduité de l'élève : les retards et les absences**. Le Système affiche un historique détaillé des absences et retards.
 - *Option 3* : Le **Parent** choisit de **Suivre l'avancement des cours**. Le Système affiche la couverture du programme par matière.
Le **Parent** peut ensuite choisir de **Suivre les compétences acquises**. Le Système affiche la grille des compétences validées pour l'enfant.
6. Le Système affiche les données demandées.
7. Le **Parent** consulte les informations et peut revenir à la vue globale ou se déconnecter.

5. Scénarios Alternatifs

A. Données indisponibles

5a. Si certaines données (ex : note d'une évaluation) ne sont pas encore publiées par l'enseignant :

5a1. Le Système affiche un message indiquant que les données sont "En attente de publication" ou "Non disponibles pour le moment".

5a2. Le **Parent** est invité à consulter d'autres indicateurs.

B. Alerte de difficulté

4a. Lors de l'affichage du tableau de bord global (Point 4), si des alertes automatiques sont configurées (ex : moyenne sous un seuil, nombre d'absences excessif) :

4a1. Le Système affiche un **message d'alerte visuel** (ex: notification push, bandeau) sur le tableau de bord.

4a2. L'alerte redirige le **Parent** directement vers la section détaillée concernée (ex: **Consultation de la progression par un parent** ou **Vérifier l'assiduité**).

6. Post-conditions

- Le **Parent** dispose d'une vision actualisée de la progression de son enfant.
- Le Système a enregistré la consultation (à des fins de suivi interne, non visible par le **Parent**).
- Si le Parent a accédé à des alertes (ex: difficultés), il peut être incité à utiliser la Messagerie interne (**CU Communication parent-enseignant**).

Cas d'utilisation : Planifier

1. Identification du Cas d'Utilisation

Nom du CU Principal : Planifier

CU Étendus :

- Mettre calendrier interactif (« extend »)
- Planifier des réunions (« extend »)
- Planifier des événements (« extend »)

Acteur Primaire : Administrateur

2. Objectif

Permettre à l'administrateur de planifier et d'organiser les activités de la plateforme **EduFamily**, telles que les réunions, les événements scolaires et les tâches importantes, à l'aide d'un calendrier interactif.

Ce cas d'utilisation vise à centraliser la gestion du temps et des événements afin d'améliorer la coordination entre les enseignants, les élèves et les parents.

3. Pré-conditions

- L'administrateur doit être authentifié sur la plateforme.
- Les modules nécessaires (calendrier, base de données des utilisateurs, événements, etc.) doivent être opérationnels.
- Les droits de planification doivent être attribués à l'administrateur.

4. Scénario Principal

1. L'administrateur accède à la section **Planification** de la plateforme.
2. Le système affiche l'interface du calendrier interactif.
3. L'administrateur sélectionne une date et une heure pour créer un nouvel élément de planification.
4. Le système affiche un formulaire de création d'événement.
5. L'administrateur saisit les informations nécessaires (titre, description, type : réunion / événement / tâche, participants, durée, etc.).
6. Le système enregistre les données et les associe au calendrier.
7. Le système notifie les utilisateurs concernés via la messagerie interne ou des alertes.
8. Le système met à jour automatiquement le calendrier interactif.
9. L'administrateur peut visualiser, modifier ou supprimer les éléments planifiés.

5. Scénarios Alternatifs

A. Planification d'une réunion (extension)

- 5a1. L'administrateur choisit le type : *Réunion*.
- 5a2. Le système propose la liste des participants possibles (enseignants, parents, direction).
- 5a3. L'administrateur sélectionne les participants et valide.
- 5a4. Le système envoie automatiquement les invitations et ajoute la réunion au calendrier.

B. Planification d'un événement (extension)

- 5b1. L'administrateur choisit le type : *Événement*.
- 5b2. Le système permet de définir la localisation, le public concerné et la durée.
- 5b3. Le système envoie une annonce aux utilisateurs concernés.

C. Échec d'enregistrement (erreur système)

6a1. Si le serveur ou la base de données est indisponible, le système affiche un message :

« Erreur lors de la sauvegarde de l'événement. Veuillez réessayer plus tard. »

6a2. L'administrateur peut retenter l'opération une fois le service rétabli.

6. Post-conditions

- Le nouvel élément de planification (réunion, événement ou tâche) est enregistré dans le calendrier.
- Les utilisateurs concernés reçoivent les notifications correspondantes.
- Le calendrier interactif est mis à jour en temps réel.

2. Modèle du Domaine : Diagramme de Classes

Le diagramme de classes ci-dessous présente la **structure statique** des informations et des entités clés de la plateforme "EduFamily Connect". Il sert de base pour l'implémentation du système.

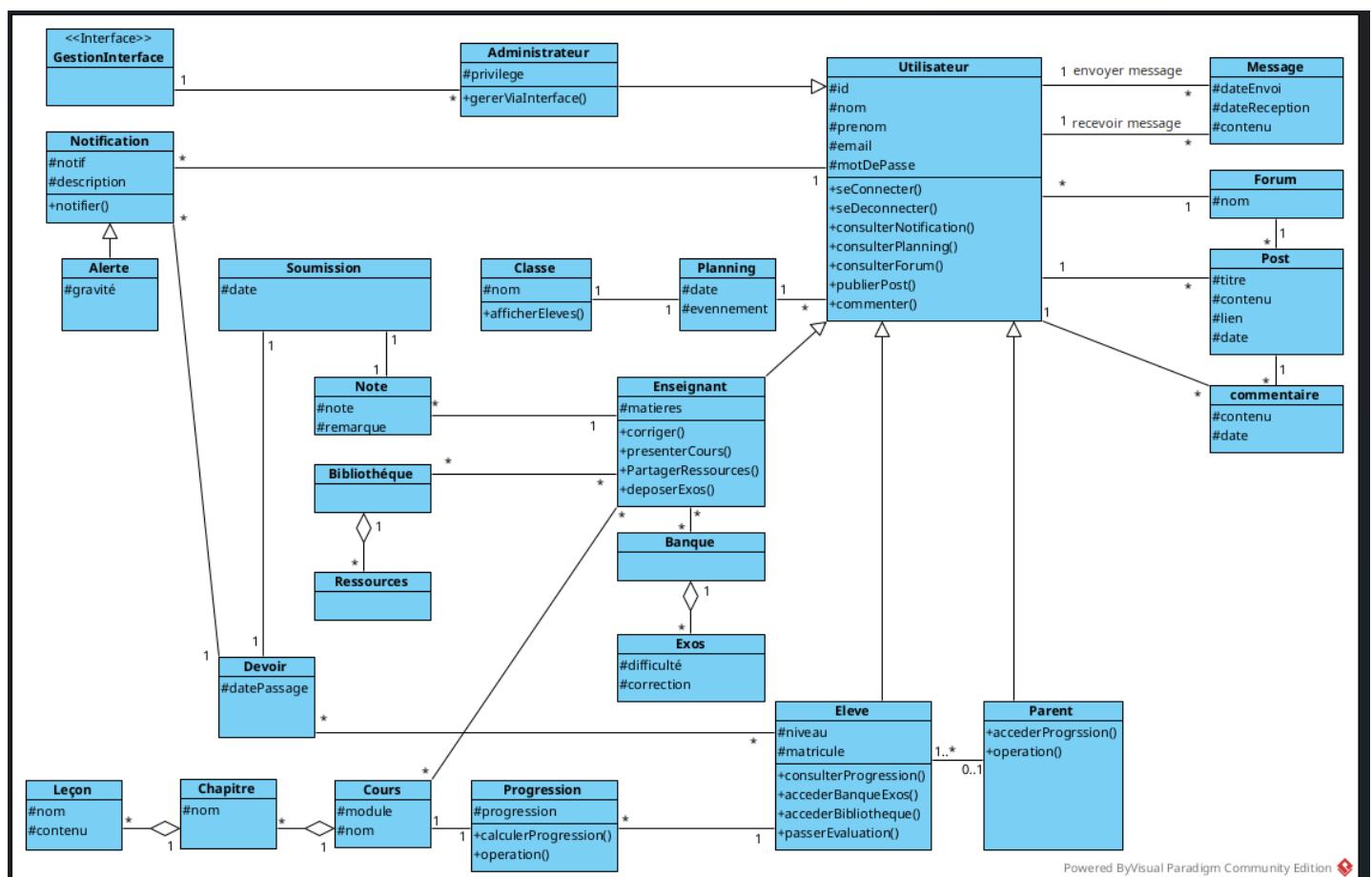


Figure 7 : Diagramme de Classe

2.1 Identification des Concepts Métier Les concepts métier principaux (classes) sont identifiés directement à partir des exigences du projet (gestion des utilisateurs, cours, évaluations, communication, etc.). Ils représentent les entités fondamentales du système.

Catégorie	Concepts Métier Principaux	Rôle dans le Système
Acteurs	Utilisateur, Administrateur, Enseignant, Élève, Parent	Gèrent les accès et les interactions spécifiques au rôle (connexion, consultation, gestion).
Contenu Pédagogique	Cours, Chapitre, Leçon, Ressource, Bibliothèque	Structuration et mise à disposition du matériel d'apprentissage.
Évaluation & Suivi	Devoir, Exos (Exercice), Banque, Soumission, Note, Progression	Gestion des travaux, des tests et du suivi de performance des élèves.
Organisation & Logistique	Classe, Planning, Notification, Alerta, GestionInterface (Interface)	Organisation des groupes, des événements et des communications automatiques/générales.
Communication	Message, Forum, Post, Commentaire	Outils facilitant les échanges (privés et publics) entre les différents acteurs.

2.2 Définition des Éléments du Modèle

- **Définition des Attributs, Opérations et Relations**

Définition des Attributs, Opérations et Relations

- **Attributs & Opérations :**
 - La classe **Utilisateur** centralise les attributs de base (ex: #id, #nom, #email, #motDePasse) et les opérations communes (ex: +seConnecter(), +seDeconnecter(), +consulterNotification()).
 - Les sous-classes ajoutent des opérations spécifiques : **Enseignant** (+corriger(), +presenterCours(), +deposerExos()), **Élève** (+accéderBanqueExos(), +passerEvaluation()), **Administrateur** (+gererVrailInterface()).
 - Les classes de données décrivent leurs propriétés : **Note** (#note, #remarque), **Message** (#dateEnvoi, #contenu), **Post** (#titre, #contenu).
- **Héritage (Généralisation) :**
 - Les classes **Administrateur**, **Enseignant**, **Élève**, et **Parent** héritent de la classe générale **Utilisateur**.
- **Agrégation/Composition (Dépendances Fortes) :**
 - Une **Bibliothèque** contient 1 à plusieurs **Ressources** (Agrégation).

- Un **Cours** est composé d'un ou plusieurs **Chapitres** qui, à leur tour, contiennent 1 à plusieurs **Leçons** (Agrégation/Composition).
 - Une **Soumission** (la copie d'un Devoir) est associée à une **Note**.
- **Relations d'Association Clés :**
 - Un **Utilisateur** peut envoyer et recevoir des **Messages** (1 envoyer message, 1 recevoir message).
 - Un **Enseignant** est associé à des **Cours** (**), des **Classes** (1), et une **Banque d'Exos** (1).
 - Un **Devoir** est associé à 1 ou plusieurs **Soumissions** (1..).
 - Un **Élève** appartient strictement à **une seule Classe** (1).

Spécification des Cardinalités et Contraintes

Les cardinalités (multiplicités) sont spécifiées sur le diagramme et définissent les contraintes structurelles du domaine.

Relation	Cardinalité	Signification
Cours / Chapitre	1..*	Un Cours doit contenir au moins un Chapitre .
Devoir / Soumission	1..*	Un Devoir peut être lié à plusieurs Soumissions (une par élève l'ayant soumis).
Élève / Classe	1	Un Élève appartient strictement à une seule Classe .
Parent / Élève	0..1 / 1..*	Un Parent est associé à un ou plusieurs Élèves (les siens), mais un Élève peut n'avoir qu'un Parent associé (ou aucun, selon le contexte \$0..1\$).
Utilisateur / Post	*	Un Utilisateur peut publier un nombre illimité de Posts sur le Forum .

2.3 Justification du Modèle et des Relations

1. Le modèle proposé est justifié par la nécessité de répondre aux objectifs et fonctionnalités d'une plateforme d'apprentissage structurée :
2. **Flexibilité et Réutilisation (Héritage)** : La classe **Utilisateur** généralise les fonctionnalités de connexion et de communication (*Message*, *Forum*), réduisant la duplication de code. L'héritage permet aux rôles spécifiques (Enseignant, Élève, Parent) d'ajouter des opérations ciblées sans alourdir la classe mère.
3. **Gestion Structurée du Contenu (Agrégation)** : La décomposition du **Cours** en **Chapitres** puis en **Leçons** assure une organisation pédagogique hiérarchique et modulaire. La **Bibliothèque** centralise les **Ressources** pour une réutilisation facile par les **Enseignants**.

4. **Traçabilité des Évaluations (Devoir, Soumission, Note)** : La distinction entre un **Devoir** (le sujet) et sa **Soumission** (la copie par l'Élève) liée à une **Note** permet une traçabilité complète du cycle d'évaluation, du rendu à la notation par l'**Enseignant**.
5. **Suivi Personnalisé (Progression)** : La classe **Progression**, associée à l'**Élève** et enrichie par l'opération `+calculerProgression()`, est essentielle pour fournir des tableaux de bord et un suivi de performance qui agrège les résultats et non pas seulement des notes brutes.
6. **Communication et Organisation (Alerte, Notification, Planning)** : Ces classes garantissent que le système peut informer activement ses acteurs. **Alerte** est spécialisée à partir de **Notification** par sa `#gravité`, permettant de distinguer les messages importants des communications courantes. Le **Planning** organise les **Événements** pour les **Classes**.

Partie 3 : Modélisation UML - Conception Détailée (Phase 2)

1. Modélisation des Interactions : Diagrammes de Séquence Système

1– Soumission et Correction des soumissions pour un enseignant

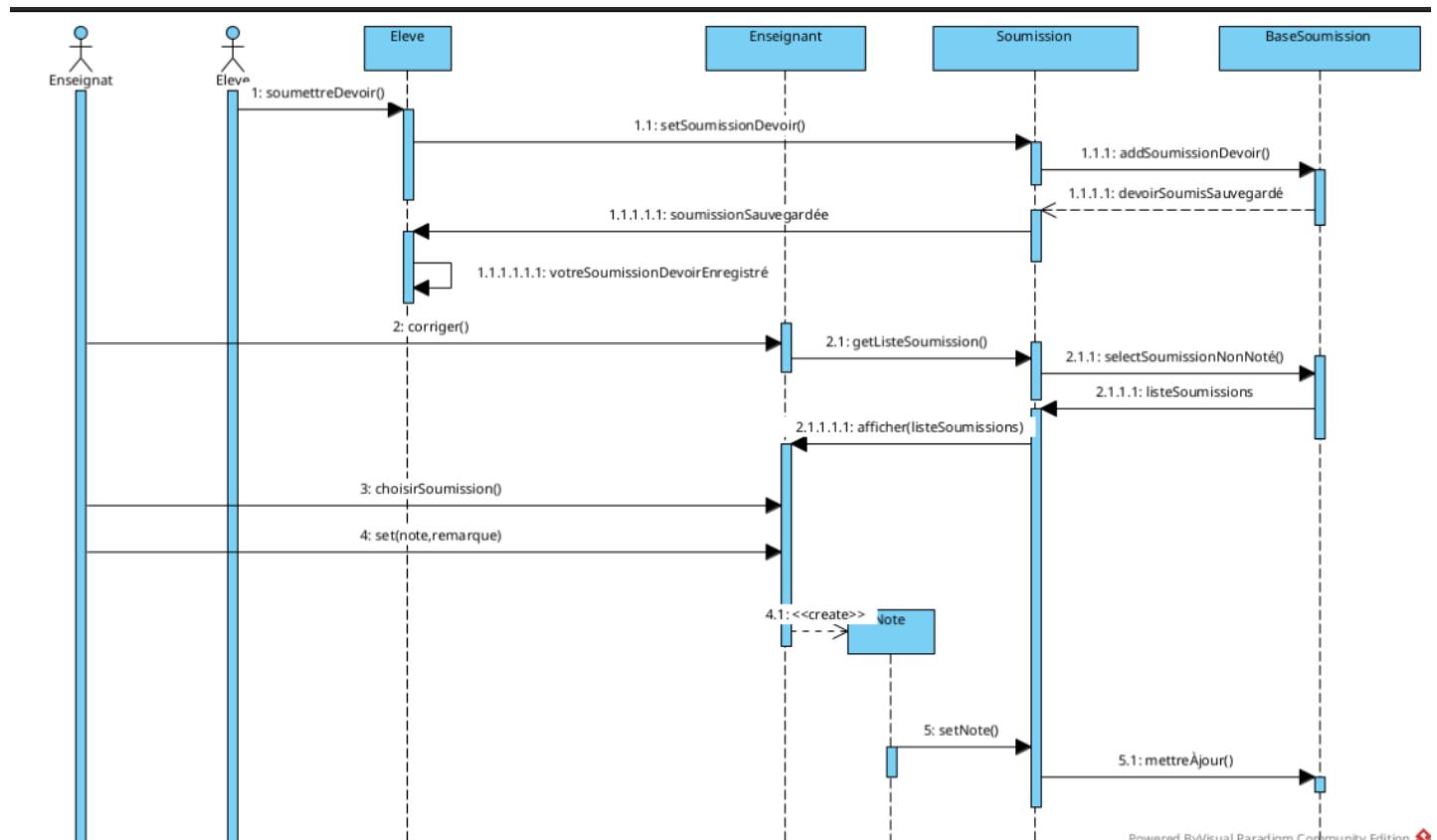


Figure 8 : Diagramme de soumission et correction soumission par enseignant

Powered ByVisual Paradigm Community Edition

Description : Ce scénario modélise la séquence critique de soumission devoir et correction manuelle. Il montre le processus par lequel l'**Enseignant** accède à une liste de soumissions d'élèves pour un devoir donné, sélectionne un travail, saisit la **note** et le feedback, et déclenche l'enregistrement de cette note dans l'entité Note

2 – Use Case : Communication parent enseignant

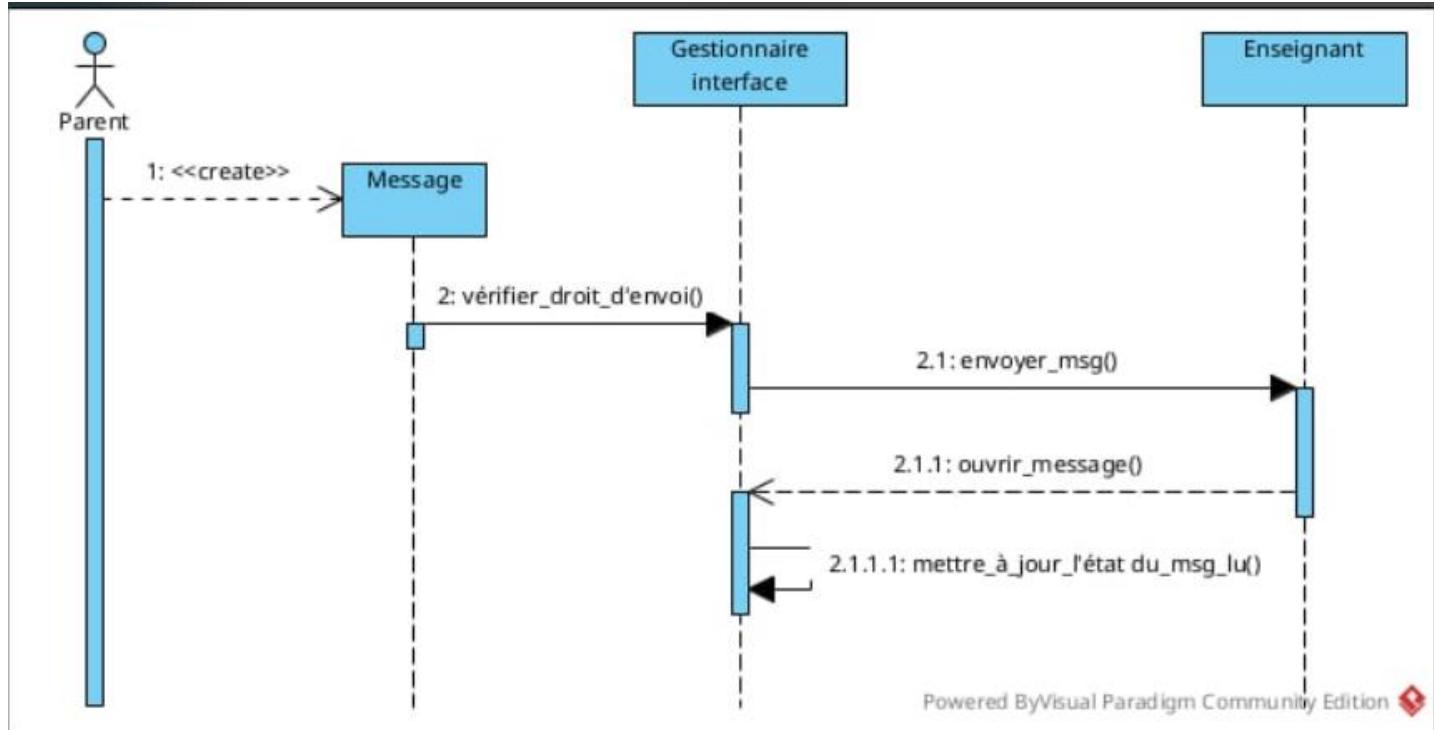


Figure 9 : Diagramme d 'interaction parent - enseignant

Ce scénario modélise l'échange sécurisé d'un message entre un Parent et un Enseignant via la messagerie interne de la plateforme. Le Parent initie le processus en rédigeant et soumettant son message. L'API vérifie les droits, enregistre le message dans la base de données, et déclenche l'envoi d'une notification push ou SMS à l'Enseignant. Lorsque l'Enseignant ouvre le message, l'état de celui-ci est mis à jour dans le système pour indiquer qu'il a été **Lu**.

3 – Use Case :Création d'une évaluation par un enseignant

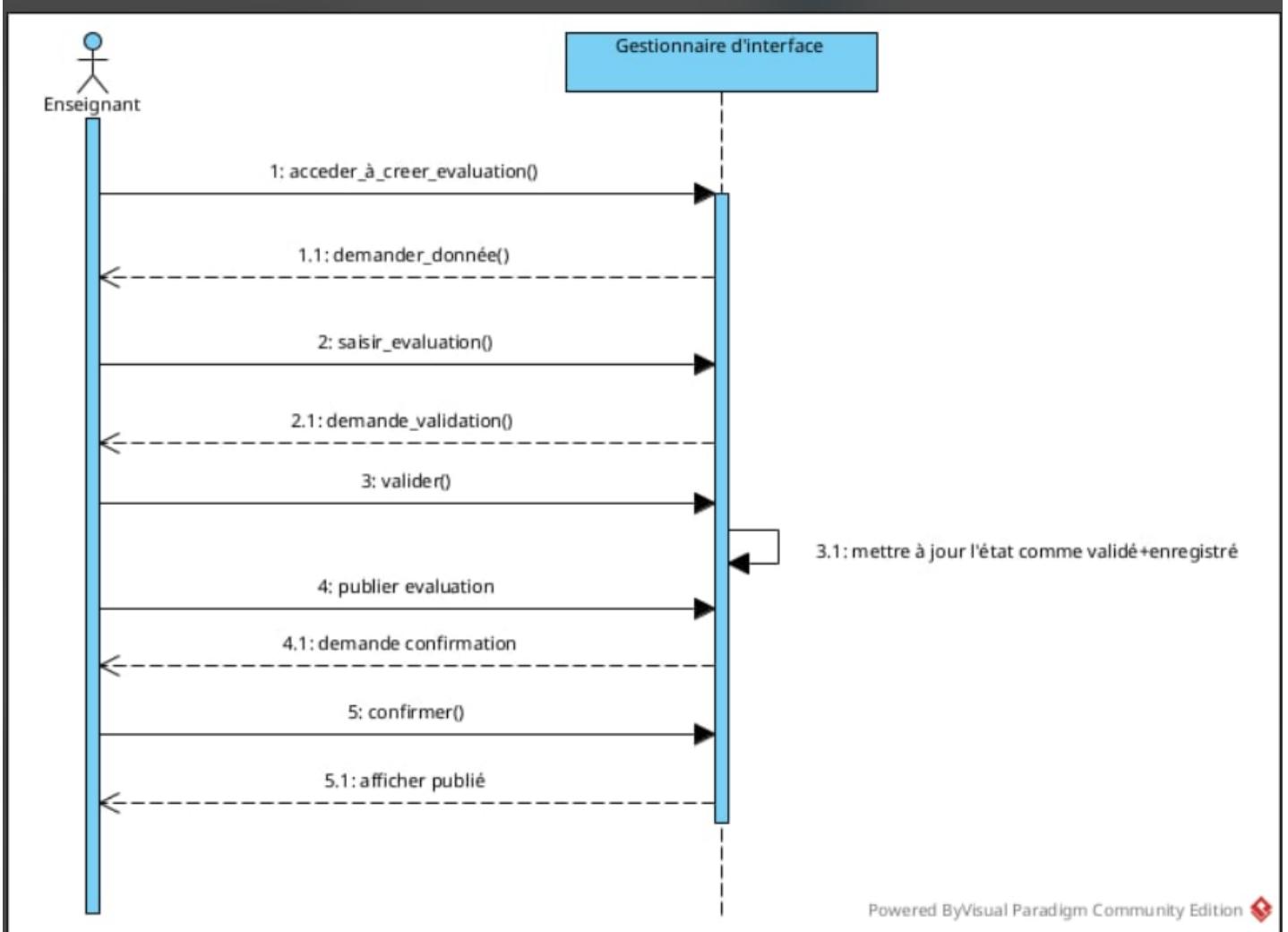


Figure 10 : Diagramme de création d'évaluation

Ce scénario modélise la séquence de création et de publication d'une nouvelle évaluation ou d'un devoir par un enseignant. Il commence par l'accès au formulaire et la récupération des données de contexte nécessaires (cours, classes) auprès de l'API et de la base de données. L'enseignant configure ensuite l'évaluation (questions, barème, etc.) et l'enregistre initialement dans l'état **Brouillon**. L'action finale, souvent distincte, consiste à **Publier** l'évaluation, ce qui met à jour son état et la rend visible aux élèves ciblés.

4 – Use case : Consulter progression pour parent

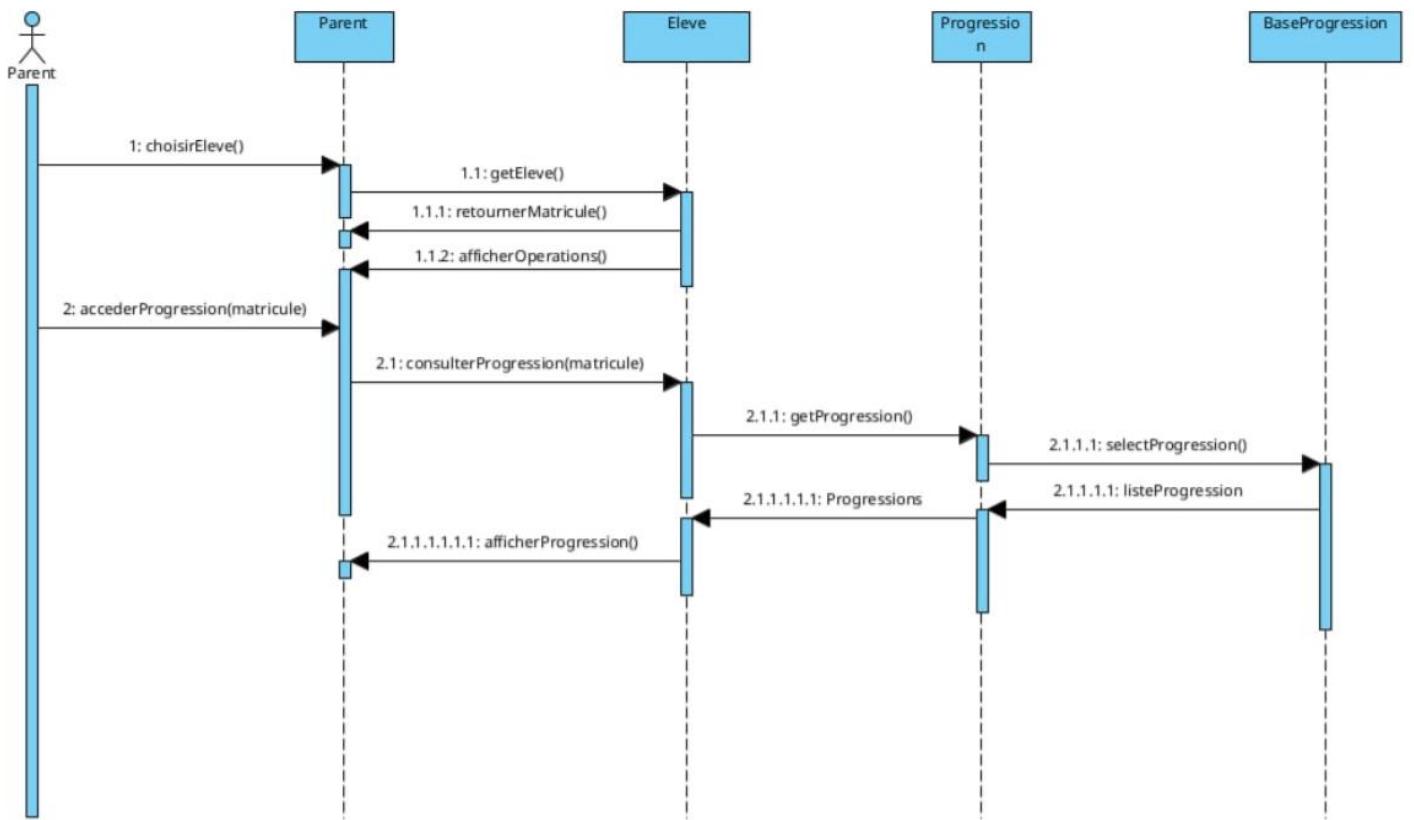


Figure 11 : Diagramme de consultation progression pour parent

Description : Ce diagramme capture l'interaction clé de **suivi parental**. Il décrit la séquence où un **Parent** s'authentifie, sélectionne l'enfant concerné puis accède au tableau de progression.

5– Use Case : Passer Evaluation pour élève

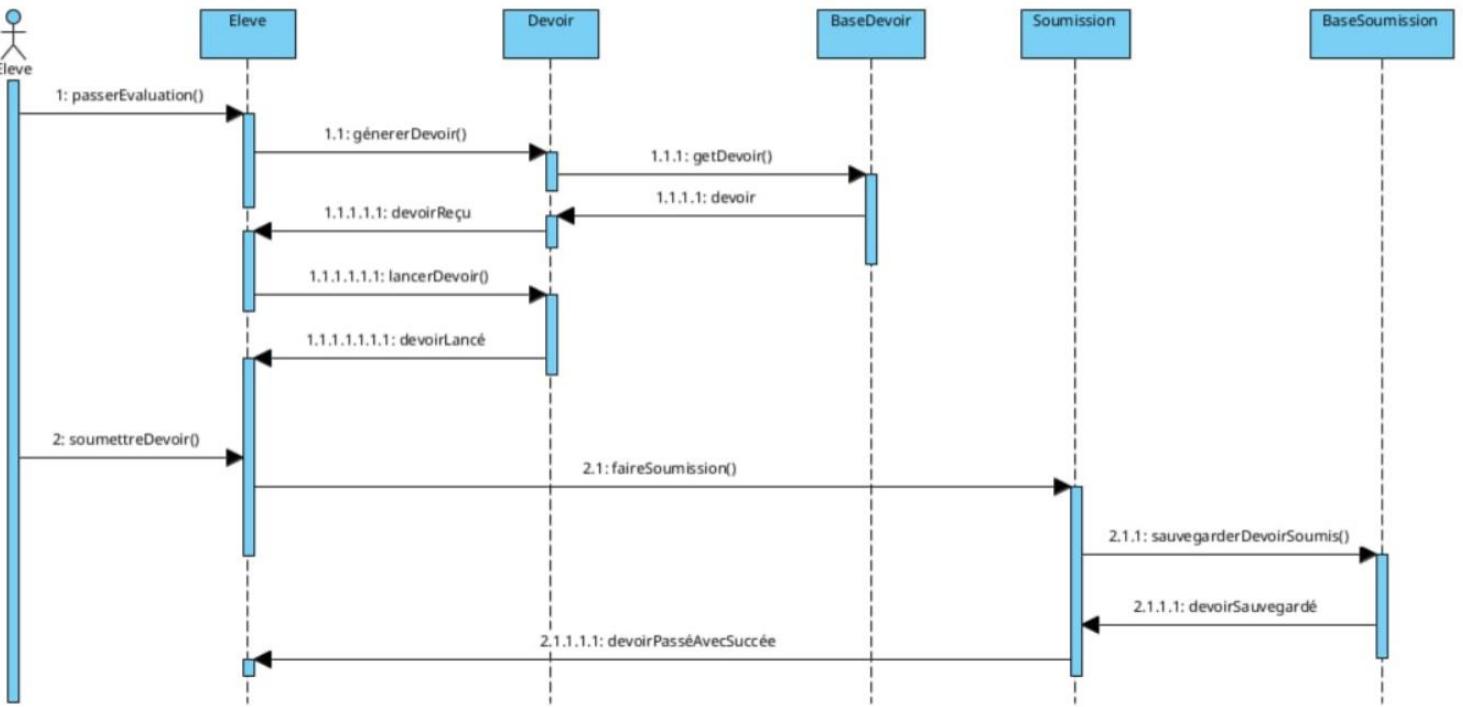


Figure 12 : Diagramme de passation évaluation

Description : Ce diagramme modélise le flux complet des interactions qui se produisent lorsqu'un **Élève** décide de passer une évaluation ou un examen en ligne. La séquence capture le processus allant de la requête initiale de l'élève à la génération du devoir, la soumission des réponses et l'enregistrement dans le système.

6 – Use case : Commenter un post du Forum pour un user

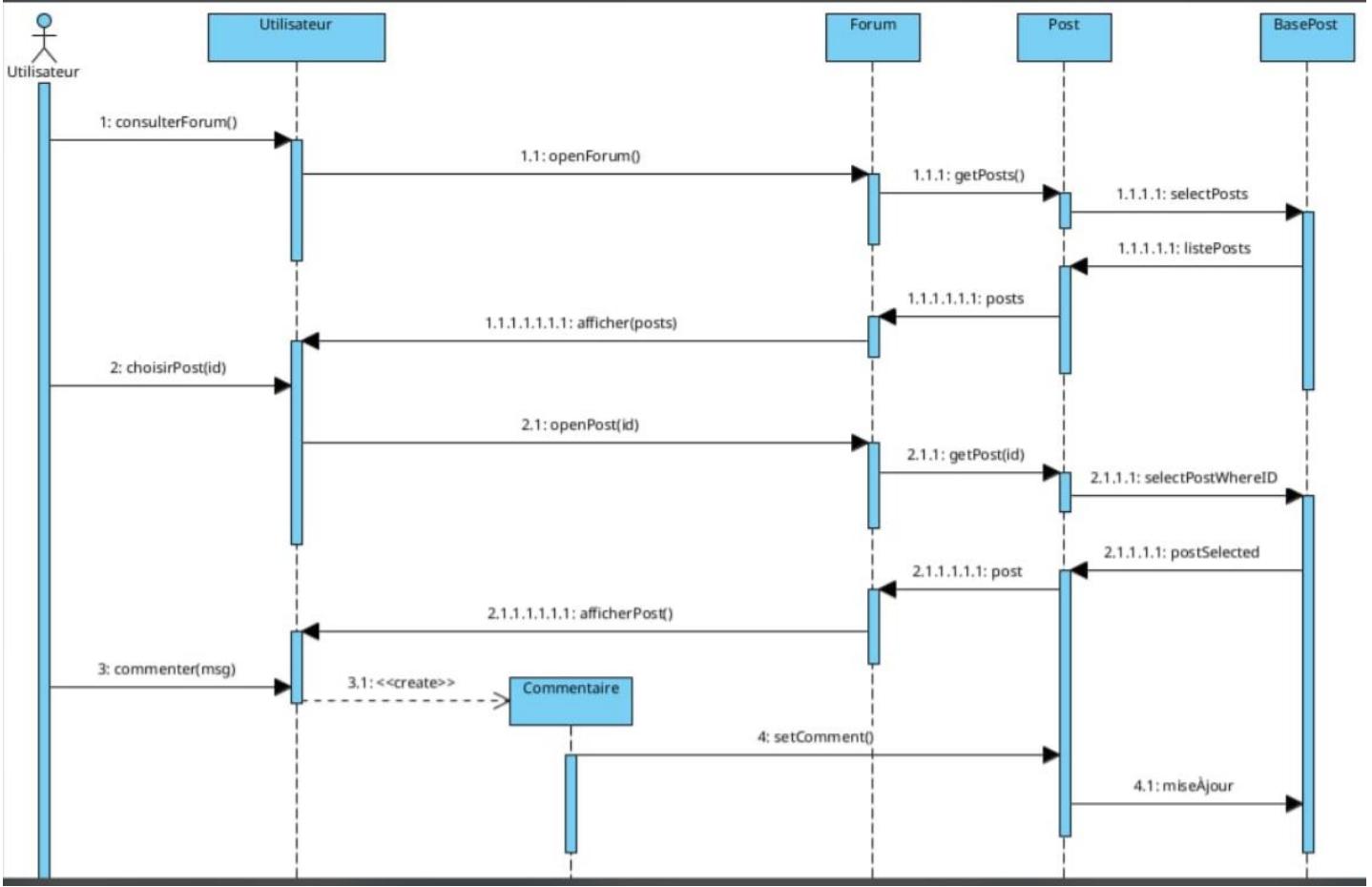


Figure 13 : Diagramme d ‘interaction utilisateur - forum

Description : Ce modèle représente le scénario de communication asynchrone sur le forum de la plateforme. Il détaille les étapes par lesquelles un utilisateur (que ce soit un Parent, un Élève ou un Enseignant, administrateur) interagit avec l’interface pour soumettre un **commentaire** sur un post existant, incluant l’accès au forum, l’affichage et la sélection du post, commenter et l’enregistrement du nouveau commentaire par le module de communication du système

2. Modélisation des Processus : Diagrammes d'état transition et d'Activité

Cycle de vie d'un cours

cycle de vie dun cours:

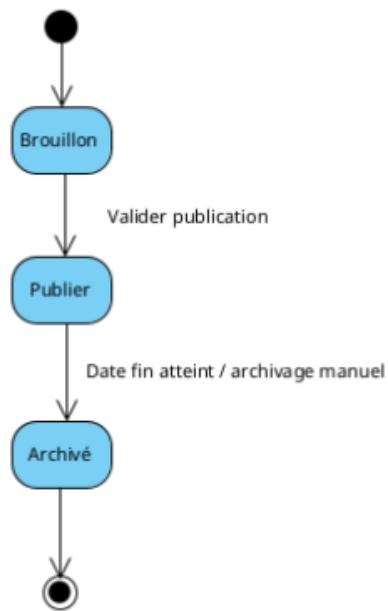


Figure 14 : Diagramme de cycle de vie du cours

Ce diagramme montre les étapes et les conditions (transitions) par lesquelles passe un cours, de sa création à son archivage.

Élément	Description
État Initial (Début)	Le cercle noir plein indique le point de départ du cycle.
État 1 : Brouillon	C'est l'état initial du cours. Le cours est en cours de création ou de modification par l'enseignant et n'est pas visible par les élèves ou les parents.
Transition 1 : Valider publication	C'est l'action ou la condition qui fait passer le cours de l'état "Brouillon" à l'état "Publier". Cette action est généralement effectuée manuellement par l'enseignant une fois que le contenu est prêt.
État 2 : Publier	Le cours est actif et visible par les élèves concernés. Les activités (leçons, exercices) sont accessibles pour l'enseignement et l'apprentissage.
Transition 2 : Date fin atteint / archivage manuel	Deux conditions peuvent mettre fin à la publication : 1) La date de fin planifiée pour le cours est atteinte (transition automatique). 2) L'enseignant décide de l' archiver manuellement avant cette date.
État 3 : Archivé	C'est l'état final du cours. Le cours n'est plus actif pour l'apprentissage régulier, mais son contenu et les notes associées sont conservés pour consultation et historique .
État Final (Fin)	Le cercle noir double indique la fin du cycle de vie du cours.

Ce cycle assure un suivi précis du statut des communications (Envoyé/Lu) et offre à l'utilisateur deux options terminales pour la gestion de sa boîte de réception : la **conservation** (Archivé) ou l'**élimination** (Supprimé).

Cycle de vie d'un devoir

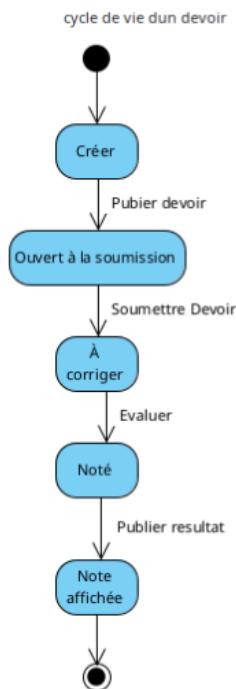


Figure 15 : Diagramme de cycle de vie d'un devoir

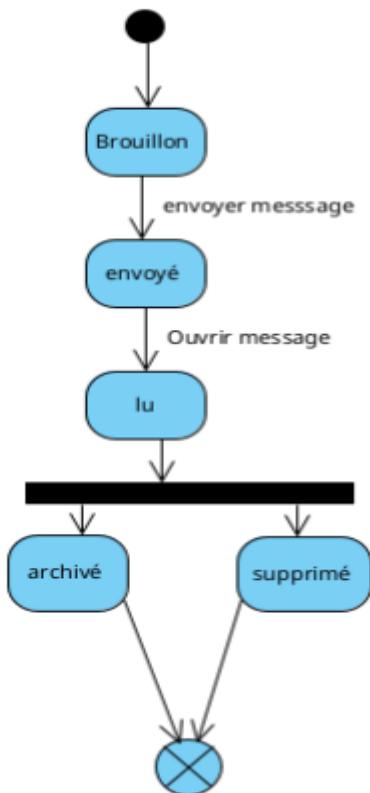
Ce diagramme d'état-transition modélise le processus complet que suit un devoir ou une évaluation, de sa création par l'enseignant jusqu'à l'affichage de la note à l'élève.

État	Transition (Action)	Rôle dans le Processus
Créer		Le devoir est défini par l'enseignant (consignes, barème, etc.) mais n'est pas encore accessible aux élèves.
Ouvert à la soumission	Publier devoir	L'enseignant rend le devoir actif et accessible aux élèves. Ces derniers peuvent commencer à travailler et soumettre leurs réponses.
À corriger	Soumettre Devoir	L'élève a soumis sa copie. Le devoir est maintenant dans la file d'attente de l'enseignant, attendant d'être évalué.
Noté	Évaluer	L'enseignant a corrigé et attribué une note (ou le système l'a fait automatiquement). La note est enregistrée mais n'est pas encore visible par l'élève.
Note affichée	Publier résultat	L'enseignant (ou le système) rend la note et le feedback associés officiellement visibles pour l'élève et les parents. C'est l'état final du processus d'évaluation.
Fin		Le cercle noir double indique la fin du cycle de vie du devoir.

Ce cycle est crucial pour la fonction de **suivi et alertes automatiques**. Il garantit non seulement que l'information est envoyée, mais aussi qu'il y a une **trace de sa reconnaissance** par l'utilisateur (Reconnue / Traitée), ce qui est essentiel pour un suivi pédagogique efficace.

Cycle de vie d'un message

cycle de vie d'un message



Powered ByVisual Paradigm Community Edition **Figure 16 : Diagramme de cycle de vie d'un message**

Ce diagramme d'état-transition modélise les états successifs par lesquels passe un message dans la messagerie interne, depuis sa création jusqu'à sa fin de vie (archivage ou suppression).

Élément	Description
État Initial (Début)	Le cercle noir plein marque le début de la création du message.
État 1 : Brouillon	Le message est en cours de rédaction par l'émetteur. Il est sauvegardé localement et n'est pas encore envoyé au destinataire.
Transition 1 : Envoyer message	L'émetteur valide et envoie le message.
État 2 : Envoyé	Le message a quitté la boîte d'envoi de l'émetteur et est arrivé dans la boîte de réception du destinataire (non encore lu).
Transition 2 : Ouvrir message	Le destinataire clique pour ouvrir et consulter le contenu du message.
État 3 : Lu	Le message a été ouvert et lu par le destinataire.
Split (Barre Noire)	Indique une décision ou un fork : après avoir été lu, le message peut suivre deux chemins indépendants.
État 4a : Archivé	Le message est déplacé dans un dossier d'archive pour être conservé en historique par l'utilisateur (émetteur ou destinataire).
État 4b : Supprimé	Le message est définitivement retiré (ou marqué pour suppression) des vues de l'utilisateur.
Merge (Cercle avec X)	Le symbole final (cercle avec un X) est une Fin d'Activité (dans un Diagramme d'Activité) ou une convention pour indiquer que les chemins d'exécution distincts convergent vers la fin du cycle de vie du message.

Ce cycle garantit que le processus d'évaluation est **structuré et séquentiel**, assurant que la note n'est rendue publique qu'après l'étape de correction et de notation par l'enseignant, ce qui est essentiel pour la fiabilité du suivi pédagogique.

Cycle de vie d'une alerte

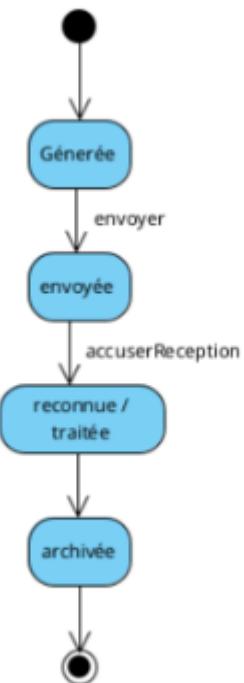


Figure 17 : Diagramme de cycle de vie d'une alerte

Ce diagramme modélise les étapes qu'une alerte (difficulté, absence, retard, etc., comme mentionné dans les fonctionnalités) traverse dans le système, de sa détection à son archivage.

Élément	Description
État Initial (Début)	Le point de départ du cycle.
État 1 : Générée	L'alerte est crée par le système (suite à une condition détectée, ex: note basse, absence enregistrée) et est prête à être notifiée.
Transition 1 : envoyer	Le système effectue l'action d'envoi de l'alerte aux destinataires concernés (parents et/ou enseignants).
État 2 : Envoyée	L'alerte a été transmise aux destinataires via la messagerie ou les notifications push.
Transition 2 : accuserReception	Le destinataire (parent ou enseignant) ouvre l'alerte ou la marque comme lue/traitée dans l'application.
État 3 : Reconnue / Traitée	L'alerte a été portée à la connaissance du destinataire et a potentiellement fait l'objet d'une action.
État 4 : Archivée	L'alerte est déplacée vers un historique, son rôle actif est terminé.
État Final (Fin)	La fin du cycle de vie de l'alerte.

Ce processus est bien structuré et met en évidence la **responsabilité partagée** : l'Enseignant fournit le contenu et les informations métier (métadonnées, droits d'accès), tandis que le Système gère la **technicité** (vérification, encodage, stockage). La saisie des métadonnées est essentielle pour l'organisation et la réutilisation de la ressource, un objectif clé de la plateforme.

[Passer evaluation](#)

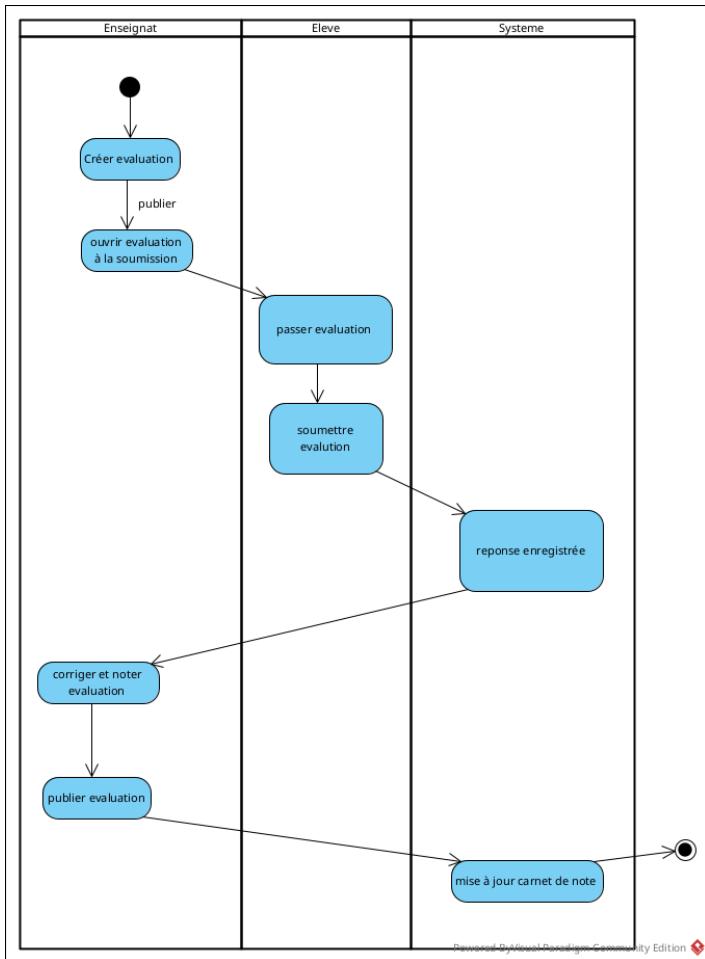


Figure 18 : Diagramme de processus évaluation

Ce diagramme d'activité modélise le flux de travail impliquant l'**Élève**, l'**Enseignant**, et le **Système** lors du déroulement d'une évaluation ou d'un devoir.

Couloir (Swimlane)	Action	Rôle dans le Processus
Enseignant	Créer évaluation	L'Enseignant prépare le devoir ou l'examen.
Enseignant	Publier (Transition)	L'Enseignant rend l'évaluation disponible.
Élève	Passer évaluation	L'Élève accède au devoir et fournit ses réponses.
Élève	Soumettre la réponse	L'Élève finalise et envoie sa copie au système.
Système	Injecter correction (automatique ou manuelle)	Le système ou l'enseignant prend en charge la correction des réponses soumises.
Enseignant	Corriger et noter évaluation	L'Enseignant effectue la notation finale et ajoute les commentaires.
Enseignant	Publier évaluation	L'Enseignant rend la note publique pour l'Élève et le Parent.
Système	Mettre à jour le carnet de notes	Le système enregistre officiellement la note dans le carnet de l'Élève et met à jour les indicateurs de progression. (Étape cruciale du suivi)
Final		Fin du processus d'évaluation/soumission.

Ce processus est central à la plateforme, car il intègre les trois acteurs dans le cycle d'apprentissage et d'évaluation. Il montre la dépendance de l'Élève vis-à-vis de la publication par l'Enseignant, et surtout, la finalisation par la **mise à jour automatique du carnet de notes**, assurant la fiabilité du suivi.

Resolution Exercice

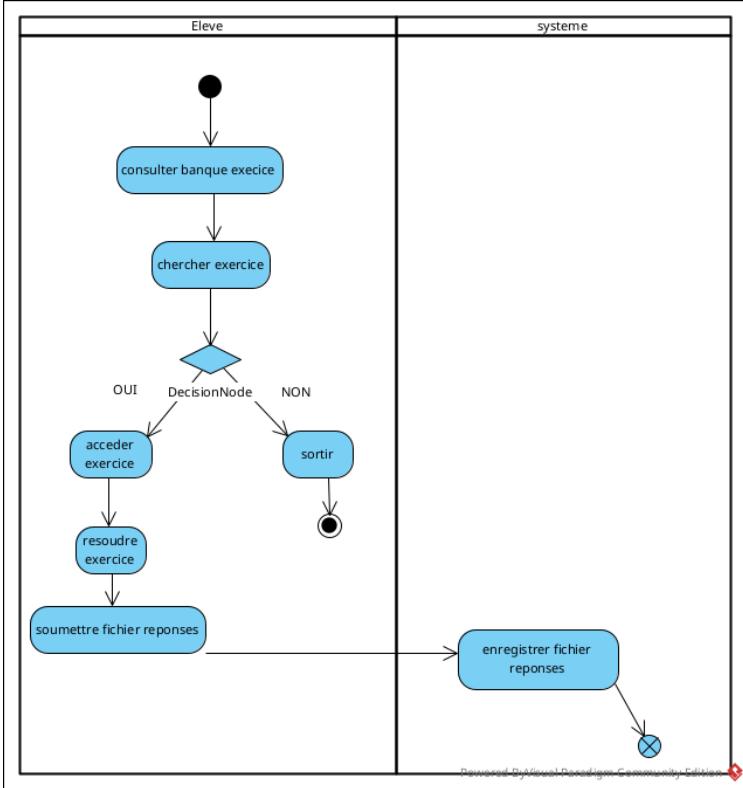


Figure 19 : Diagramme de processus résolution exercice

exercie

Ce diagramme d'activité se concentre sur les actions de l'**Élève** et l'interaction avec le **Système** pour la recherche et la soumission d'un exercice.

Couloir (Swimlane)	Action	Rôle dans le Processus
Élève	consulter banque exercice	L'Élève initie l'action en accédant à la liste des exercices disponibles.
Élève	chercher exercice	L'Élève utilise des filtres ou une recherche pour localiser l'exercice souhaité.
Nœud de Décision (Losange)	DecisionNode (OUI/NON)	Le système vérifie si l'exercice a été trouvé.
Flux 'NON'	sortir	Si l'exercice n'est pas trouvé, le processus se termine immédiatement.
Flux 'OUI'	accéder exercice	Si l'exercice est trouvé, l'Élève ouvre la page de l'exercice.
Élève	résoudre exercice	L'Élève travaille sur l'exercice (peut être hors ligne ou directement dans l'interface).
Élève	soumettre fichier réponses	L'Élève valide et envoie sa réponse ou son fichier de réponse au système.
Système	enregistrer fichier réponses	Le Système reçoit et stocke la soumission de l'Élève. (Point d'enregistrement clé)
Final		Fin du processus de soumission.

Ce processus est simple et direct. Il est essentiel pour la fonctionnalité de **gestion des exercices**. Le nœud de décision assure que la suite du processus ne s'exécute que si l'exercice est localisé, et la transition entre l'élève et le système (soumettre \ enregistrer) garantit la trace de la soumission.

Ajout de ressource a la banque

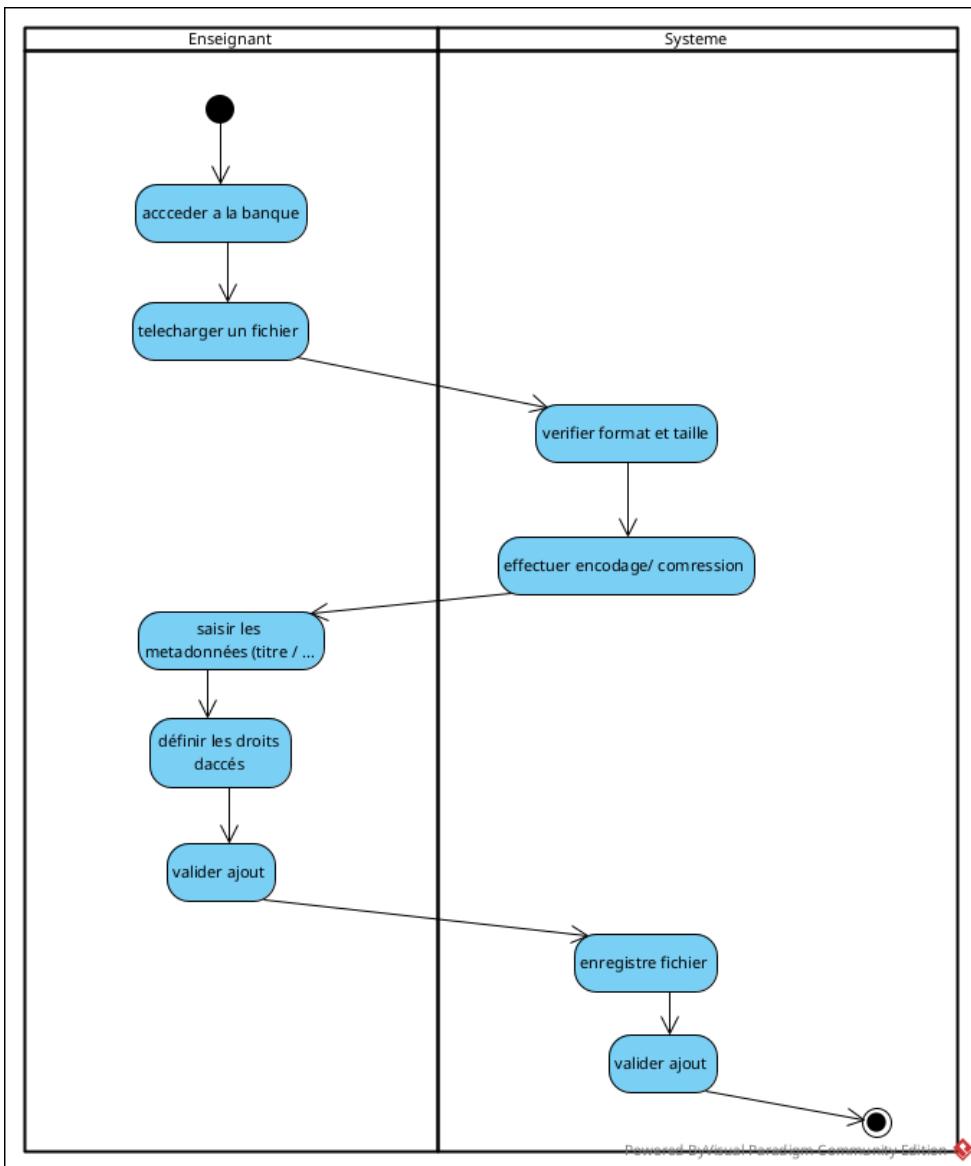


Figure 20 : Diagramme d'ajout ressources à la banque

Ce diagramme d'activité modélise le processus détaillé par lequel un **Enseignant** ajoute une ressource à la **Bibliothèque de ressources partagées** d'EduFamily Connect. Il montre l'interaction séquentielle entre l'acteur (l'Enseignant) et le système (la Plateforme).

Couloir (Swimlane)	Action	Rôle dans le Processus
Enseignant	Accéder à la banque	L'enseignant initie l'opération en ouvrant la bibliothèque de ressources.
Enseignant	Télécharger un fichier	L'enseignant sélectionne et envoie le document (vidéo, PDF, etc.) à la plateforme.
Système	Vérifier format et taille	Le système effectue des contrôles techniques immédiats (sécurité, compatibilité).
Système	Effectuer encodage / compression	Le système prépare la ressource pour le stockage et la diffusion (optimisation).
Enseignant	Saisir les métadonnées (titre /...)	L'enseignant ajoute les informations descriptives (titre, matière, mots-clés) qui faciliteront la recherche. (Étape cruciale)
Enseignant	Définir les droits d'accès	L'enseignant spécifie qui peut voir et utiliser la ressource (ex: une seule classe, tous les enseignants, tous les élèves).
Enseignant	Valider ajout	L'enseignant donne l'approbation finale pour l'enregistrement.

Système	Enregistrer fichier	Le système stocke la ressource optimisée et ses métadonnées dans la base de données.
Système	Valider ajout	Confirmation de l'enregistrement et fin du processus.
Final		Le cercle avec le point au centre indique la fin du processus.

Ce processus est bien structuré et met en évidence la **responsabilité partagée** : l'Enseignant fournit le contenu et les informations métier (métadonnées, droits d'accès), tandis que le Système gère la **technicité** (vérification, encodage, stockage). La saisie des métadonnées est essentielle pour l'organisation et la réutilisation de la ressource, un objectif clé de la plateforme.

Consulter Progression

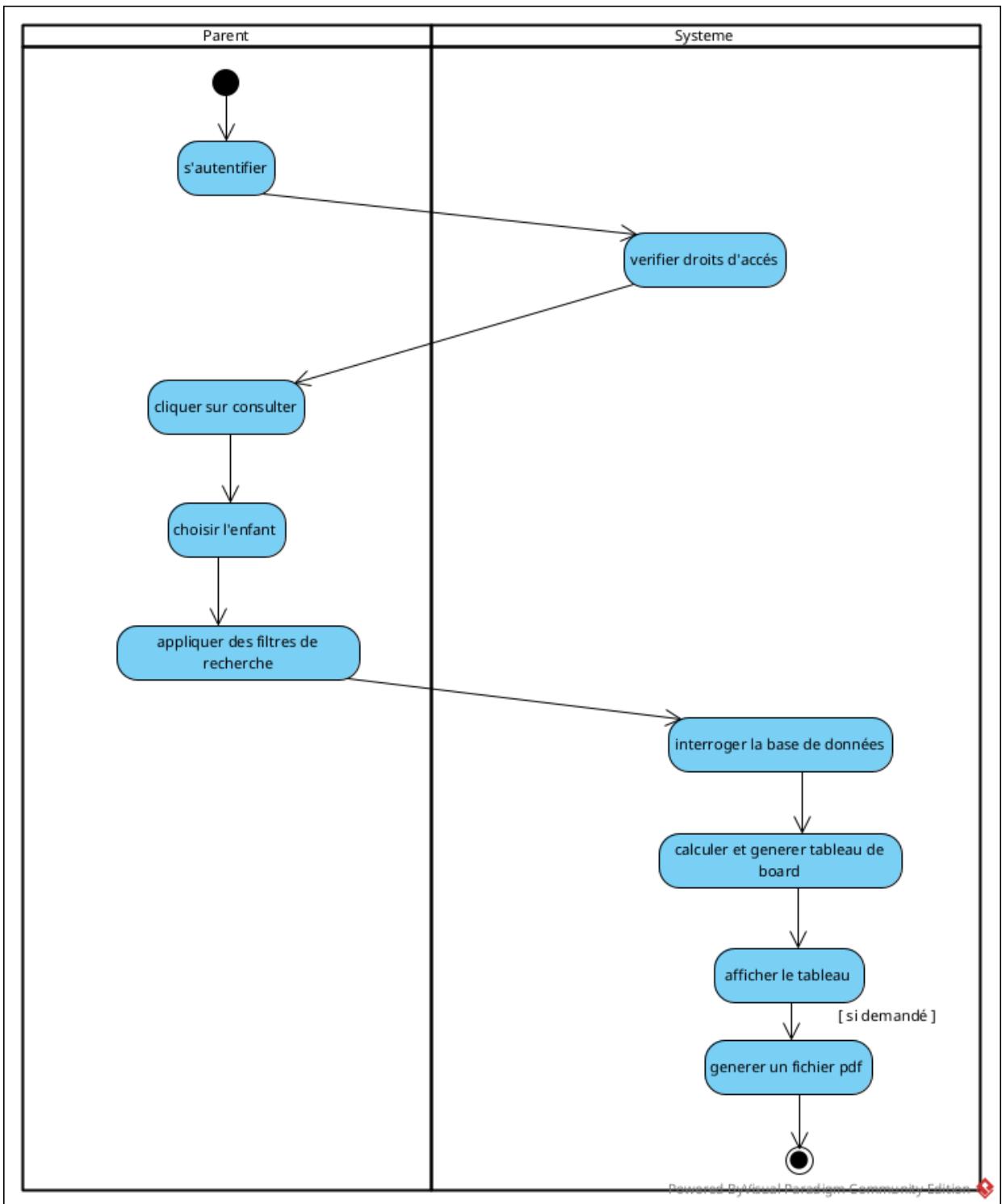


Figure 21 : Diagramme de processus de consultation de progression

Ce diagramme d'activité modélise le processus complet par lequel un **Parent** accède et visualise le tableau de bord de progression de son enfant. Il détaille l'interaction séquentielle entre le Parent et le Système.

Couloir (Swimlane)	Action	Rôle dans le Processus
Parent	s'authentifier	Le Parent se connecte à la plateforme.
Système	vérifier droits d'accès	Le système valide l'identité du Parent et vérifie à quels enfants il est lié.

Parent	cliquer sur consulter	Le Parent initie la consultation du suivi.
Parent	choisir l'enfant	Si le Parent gère plusieurs enfants, il sélectionne celui dont il veut voir la progression.
Parent	appliquer des filtres de recherche	Le Parent peut affiner sa demande (ex: progression en mathématiques, sur le dernier trimestre).
Système	interroger la base de données	Le système récupère les données brutes (notes, compétences validées, absences) selon les filtres.
Système	calculer et générer tableau de bord	Le système agrège les données, effectue les calculs (moyennes, coefficients) et met en forme le tableau de bord.
Système	afficher le tableau	Le Parent visualise la progression à l'écran.
Système	[si demandé] générer un fichier pdf	Le Parent a l'option d'exporter le tableau de bord pour l'imprimer ou le sauvegarder.
Final		Le cercle avec le point au centre indique la fin du processus.

Ce workflow est fondamental pour l'objectif "**Offrir un suivi personnalisé de la progression des élèves**". Il garantit la **sécurité** (vérification des droits), la **flexibilité** (application de filtres) et fournit une information **synthétisée** (calcul et génération du tableau de bord) ainsi qu'une option d'exportation.

Partie 4 : Architecture et Conception Avancée (Phase 3)

1. Architecture Logique : Diagramme de Packages

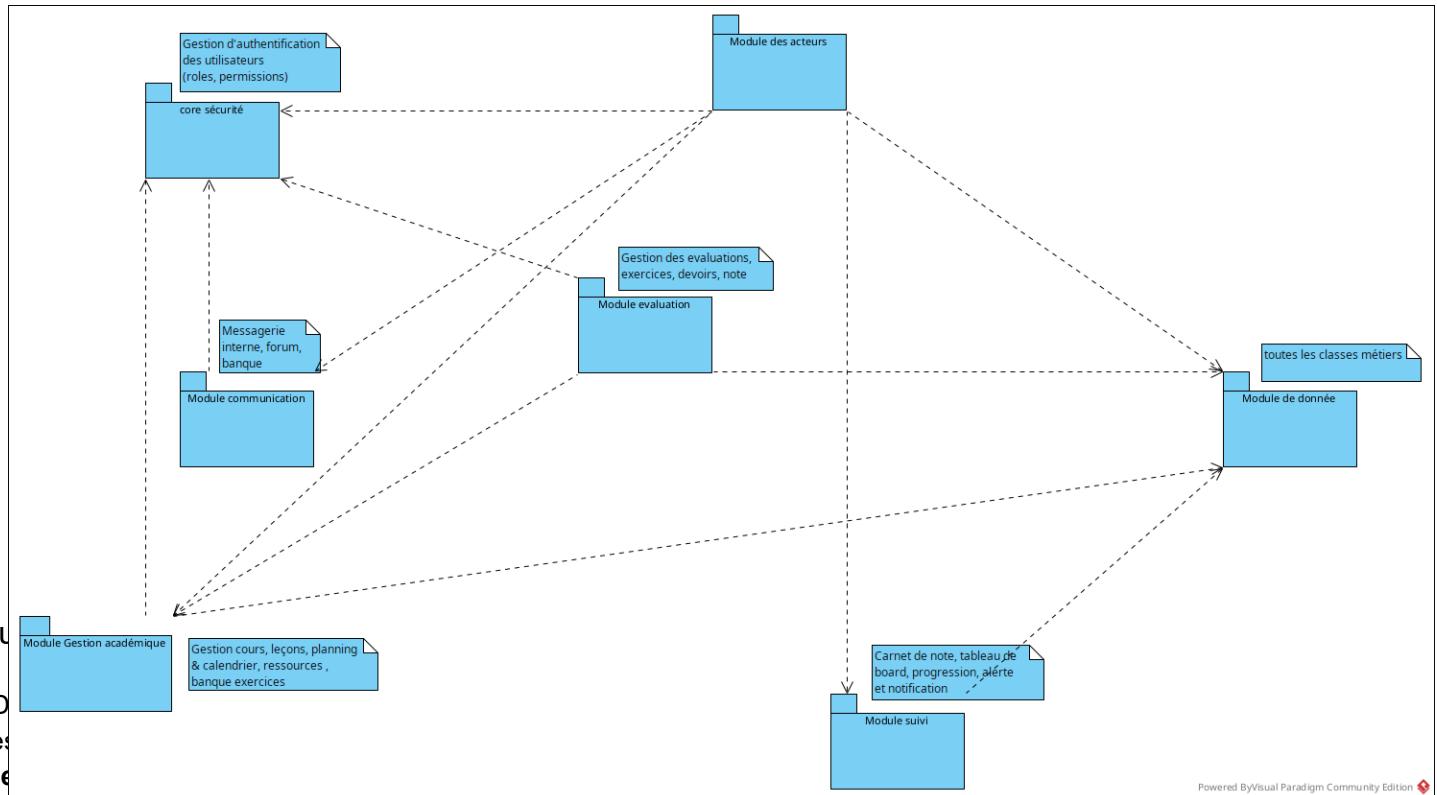


Figure 4

Le D
et les
dépe

Organisation des Packages

Le système est décomposé en sept packages principaux, reflétant la structure fonctionnelle de la plateforme :

Package	Contenu et Rôle
Module Sécurité	Gère l' authentification des utilisateurs, les rôles (Élève, Enseignant, Parent) et les permissions (droits d'accès). C'est le cœur de la protection du système.
Module Utilisateurs	Gère les profils des acteurs (Élèves, Enseignants, Parents). Contient les informations de base des utilisateurs.
Module Gestion Académique	Gère la structure des cours : cours, leçons, planning, calendrier, ressources et la banque d'exercices . C'est le cœur du contenu pédagogique.
Module Évaluation	Gère les processus d'évaluation : évaluations, exercices, devoirs et l'enregistrement des notes .
Module Suivi	Gère le retour d'information et la performance : carnet de notes, tableau de bord, progression et les alertes/notifications.
Module Communication	Gère les outils d'échange : messagerie interne, forums et la gestion des documents partagés (banque).
Module de Données	Représente le niveau le plus basique. Contient les classes métiers (Cours, Note, Utilisateur, etc.) qui sont persistantes (stockées en base de données).

Analyse des Dépendances Clés

Les flèches indiquent qu'un package a besoin des services ou des classes d'un autre. Les dépendances sont logiques et montrent la hiérarchie de l'information :

1. **Dépendance forte vers le Module de Données** : Presque tous les packages fonctionnels (**Sécurité, Utilisateurs, Gestion Académique, Évaluation, Suivi, Communication**) dépendent du **Module de Données**. C'est logique : ces modules ont tous besoin de manipuler (lire/écrire) les objets métier de base (Cours, Note, Utilisateur, etc.).
2. **Sécurité et Utilisateurs** : Le **Module Sécurité** dépend du **Module Utilisateurs** (pour connaître les informations de profil) et inversement, le **Module Utilisateurs** dépend du **Module Sécurité** pour les permissions.
3. **Gestion Académique et Évaluation** : Le **Module Évaluation** dépend du **Module Gestion Académique** (on ne peut évaluer que des éléments définis dans les cours et les banques d'exercices).
4. **Suivi** : Le **Module Suivi** dépend de l'**Évaluation** (pour les notes) et de la **Gestion Académique** (pour connaître la structure du cours à suivre).
5. **Communication et Gestion Académique** : Le **Module Communication** dépend de la **Gestion Académique** car les forums ou la messagerie sont souvent structurés par matière ou par classe (définies dans l'Académique).

Ce diagramme établit une **architecture logique claire et modulaire**. La séparation des fonctionnalités (Sécurité, Académique, Évaluation, Suivi, Communication) facilite le développement, la maintenance et l'évolution de la plateforme, avec une couche de **Données** centralisée et une couche de **Sécurité** transversale, ce qui est une bonne pratique d'architecture logicielle.

2. Architecture Physique : Diagramme de Composants

3. Architecture de Déploiement

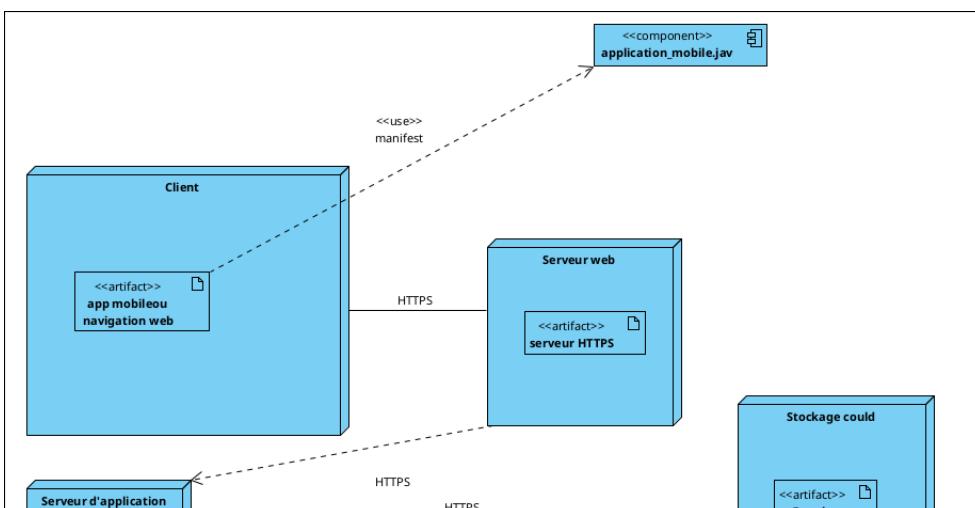


Figure 22 : Architecture déploiement

Ce diagramme modélise l'architecture physique du système, montrant les **nœuds** (serveurs, dispositifs clients) et la manière dont les **composants logiciels** (artefacts) sont installés et communiquent entre eux.

Nœuds et Composants (Artefacts)

Nœud (Matériel / Localisation)	Rôle et Composants Clés
Client	Représente les appareils utilisateurs (téléphones mobiles, ordinateurs). Il contient l'artefact app mobile ou navigation web, permettant l'accès à la plateforme.
Serveur web	Le point d'entrée du système. Il héberge l'serveur HTTPS, chargé de la sécurité (certificats SSL/TLS) et de la distribution des requêtes.
Serveur d'application	Cœur de la logique métier. Il héberge l'artefact API, qui gère toutes les fonctionnalités (gestion académique, évaluation, suivi) et les interactions entre les autres serveurs.
Serveur base de données	Stocke l'ensemble des données de la plateforme (toutes les classes métiers). Il utilise une base de données PostgreSQL .
Stockage cloud (Google)	Utilisé pour le stockage de fichiers lourds, principalement les ressources multimédia (vidéos, documents, images) de la banque de ressources.
Service externe (Service de notification)	Représente un service tiers externe, essentiel pour les fonctionnalités d' alertes (notifications push/SMS) mentionnées dans le cahier des charges.

Analyse des Connexions et Protocoles

Le diagramme montre une architecture moderne, distribuée et sécurisée :

Connexion	Rôle / Protocole	Justification
Client \ Serveur web	HTTPS	Connexion sécurisée (chiffrée) standard pour l'accès web et mobile.
Serveur web \ Serveur d'application	HTTPS	Le Serveur web relaie les requêtes sécurisées à l'API pour traitement.
Serveur d'application \ Base de données	TCP/IP	Connexion interne optimisée entre la couche logique et la couche de persistance.
Serveur d'application \ Stockage cloud	HTTPS	Connexion sécurisée pour l'envoi et la récupération des fichiers stockés dans le cloud.

Serveur d'application \ Service externe	(Non spécifié clairement, implicitement HTTPS/API)	L'API déclenche l'envoi des notifications et alertes via ce service tiers.
--	--	--

Cette architecture est un modèle à trois niveaux (**Client, Web, Application**), ce qui est robuste et évolutif. L'utilisation d'**HTTPS** sur toutes les communications externes et le recours à un **stockage cloud** et à un **service externe de notification** confirment que la plateforme est conçue pour gérer la charge, la sécurité et les fonctionnalités modernes requises par EduFamily Connect (notamment les ressources multimédia et les alertes instantanées).

Partie 5 : Conclusion et Perspectives

1. Conclusion et Bilan de la Conception

Le projet de plateforme éducative **EduFamily Connect** a été entièrement modélisé selon une approche rigoureuse basée sur le langage UML. La conception détaillée confirme que la plateforme est capable d'atteindre ses objectifs fondamentaux : **digitaliser les processus d'enseignement, faciliter la communication trilatérale, et offrir un suivi personnalisé et détaillé de la progression des élèves.**

L'analyse des diagrammes d'état et d'activité a permis de valider la fluidité et la robustesse des processus métier clés, notamment le cycle complet d'évaluation (création, soumission, correction, affichage) et la gestion des communications et des alertes.

D'un point de vue architectural, le choix d'une architecture modulaire (Diagramme de Packages) avec des modules distincts pour la Sécurité, la Gestion Académique, l'Évaluation et le Suivi garantit une excellente **maintenabilité** et une **évolutivité** future. Le Diagramme de Déploiement confirme une architecture technique moderne à trois niveaux, s'appuyant sur des protocoles sécurisés (**HTTPS**) et intégrant des services externes cruciaux (notifications push/SMS et stockage Cloud), assurant ainsi une performance et une sécurité à la hauteur des enjeux du domaine éducatif. La plateforme est désormais prête pour la phase de développement.

2. Perspectives d'Évolution et Fonctionnalités Futures

Afin d'assurer la pérennité et l'excellence de la plateforme EduFamily Connect, plusieurs axes d'évolution fonctionnels et techniques sont envisagés :

- Amélioration de l'Engagement (Gamification)** : Intégrer des mécanismes de gamification pour motiver les élèves. Cela pourrait inclure des systèmes de badges, de points d'expérience ou de classements virtuels pour la complétion des leçons et la réussite des exercices.
- Intelligence Artificielle et Prédiction** : Utiliser des algorithmes d'apprentissage automatique (**Machine Learning**) sur les données de progression (Module Suivi) pour prédire les difficultés d'apprentissage. Le système pourrait ainsi générer des **alertes proactives** pour les enseignants avant même qu'une situation de décrochage ne se manifeste.
- Mode Hors-Ligne (Mobile)** : Développer une fonctionnalité de synchronisation pour l'application mobile, permettant aux élèves de télécharger certains cours et exercices pour y travailler en **mode déconnecté** (offline), et de synchroniser leurs progrès lors de la reconnexion.
- Interopérabilité et Intégration** : Créer des interfaces et des connecteurs (APIs dédiées) pour permettre à EduFamily Connect de communiquer facilement avec d'autres systèmes de gestion scolaire existants (logiciels administratifs, ENT), renforçant ainsi son rôle d'écosystème central.

5. **Microservices et Scalabilité** : Pour la haute disponibilité et la gestion d'une base d'utilisateurs croissante, envisager une refonte de l'API en une architecture de **microservices**, permettant de gérer l'échelle des modules critiques (comme l'Évaluation en période d'examen ou la Communication instantanée) de manière indépendante et plus résiliente.