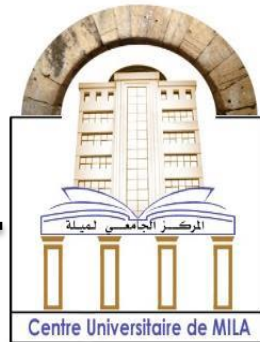


الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne Démocratique et Populaire
وزارة التعليم العالي والبحث العلمي
Ministère de l'enseignement supérieur et de la recherche scientifique



Centre Universitaire Abdelhafid Boussouf – Mila
Institut des Sciences et Technologie
Département de Mathématiques et Informatique

Mémoire préparé pour obtenir le diplôme de Master en Informatique

Option : Sciences et Technologies de l'Information et de la
Communication (STIC)

Thème :

Conception et Réalisation d'une Plateforme d'information pour le
Suivi du développement du Secteur de l'enseignement Supérieur

Réalisé par :

Boucheloukh Ghozlane.
Sekfali Aya.

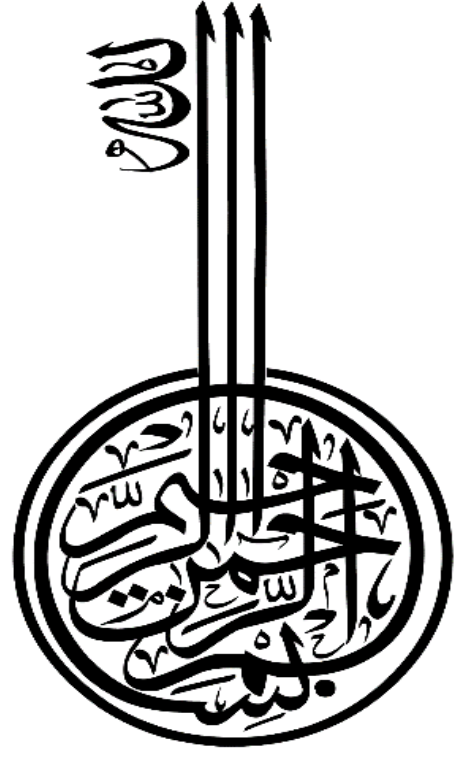
Soutenue devant le jury :

Président : Boufaghes Hamida.

Encadré par Dr. Meghzili Said.

Examineur : Boumassata Meriem.

Année Universitaire : 2022/2023



Remerciements

Avant tout, nous remercions Dieu, qui nous a donné la force et Détermination à atteindre ce jour important dans notre vie universitaire.

Avec gratitude et appréciation, Nous félicitons particulièrement notre professeur Le superviseur, Dr. **Said Meghzili**, qui a agi en tant que leader et guide lors de ce voyage De recherche grâce à sa sagesse et ses conseils, Nous ne pouvons pas oublier notre Appréciation et notre gratitude Des enseignants remarquables, qui nous ont aidés à Progresser à travers ces cinq ans, et nous remercions également les jurys pour sa Discussion et sa correction de notre mémoire de fin d'études.

En conclusion, nous remercions toujours Dieu à tout moment et à toutes les étapes vie.

Merci et nous vous souhaitons tout le meilleur avec nos meilleurs vœux.

Dédicaces

Pour tous ceux qui ont contribué au succès de notre acte.

Chers parents, les plus grands supporteurs Et à nos chers frères, pour votre influence et vos encouragements.

Nous remercions sincèrement tous ceux qui ont fourni des conseils ou des commentaires constructifs sur notre parcours scientifique.

Nous demandons à Dieu plus de succès

تلخيص

صناع القرار في قطاع التعليم العالي، بما في ذلك رؤساء المؤسسات الجامعية والأمين العام، مسؤولون عن متابعة تنفيذ خطط العمل والتطوير في قطاعهم. لذلك، من أجل تنفيذ مهامهم بشكل جيد، يحتاجون إلى قاعدة بيانات متسقة وموثوقة يتم تحديثها في الوقت الحقيقي من قبل الجهات المعنية المختلفة، وتحتوي على جميع المؤشرات اللازمة لتوفير بُعد تحليلي ودعم مرجعي لأي صانع قرار محلي أو وطني في مساره المستقبلي في عملية التنمية الاقتصادية والاجتماعية.

الهدف الرئيسي لهذا المشروع هو بناء منصة لمتابعة تطور قطاع التعليم العالي، ويجب أن يتم تطويرها باستخدام إطار تطوير ويب في الواجهة الأمامية مثل Angular ، وإطار تطوير خلفي مثل Spring Boot بناءً على هندسة microservices . ستوفر هذه المنصة معلومات موثوقة لصناع القرار، مما يمكنهم من الوصول إلى المعلومات بطريقة بسيطة وفعالة لاتخاذ القرارات اللازمة في الوقت الحقيقي ضمن عملية التنمية الاقتصادية والاجتماعية.

الكلمات المفتاحية : Spring Boot , architecture Microservice , Angular

Abstract

The decision-makers in the higher education sector, particularly the heads of academic institutions, and the Secretary-General (S.G), are responsible for monitoring the implementation of workload and development plans in their sector. Therefore, in order to successfully carry out their missions, they need a coherent and reliable database that is updated in real-time by various stakeholders. This database should contain all the necessary indices and indicators to provide a broad analytical dimension and serve as a reference for any local or national decision-maker in their prospective approach to the economic and social development process.

The main objective of this project is to build a platform for monitoring the development of the higher education sector. This platform should be developed using a web development framework on the front-end, such as **Angular**, and a back-end framework like **Spring Boot**

Résumé

based on the **Microservice** architecture. This platform will provide reliable information to decision-makers, enabling them to access information in a simple and efficient manner in order to make necessary decisions in real-time within the economic and social development process.

Keywords: Spring Boot , architecture Microservice , Angular

Résumé

Les décideurs du secteur de l'enseignement supérieur notamment les chefs des établissements universitaires, le S.G sont chargés de suivi l'exécution des plans de charge et de développement de leur secteur.

Donc, pour mener bien leurs missions, ils ont besoin d'une banque de données cohérente et fiable, mise à jour en temps réel par les différents acteurs d'activités et renferment tous les indices et indicateurs nécessaires pour offrir une large dimension analytique et un support de référence à tout décideur local ou national dans sa démarche prospective dans le processus de développement économique et social.

L'objectif principal de ce projet est de construire une plateforme permettant le suivi du développement du secteur de l'enseignement supérieur et qui doit être développé en utilisant un Framework de développement web dans le Front-end comme **Angular** et un Framework Back-end comme **Spring Boot** basé sur l'architecture **Microservice**. Cette plateforme mettra à la disposition des responsables un système d'information fiable permettant l'accès à l'information de manière simple et efficace pour prendre les décisions nécessaires dans le processus de développement économique et social en temps réel.

Les mots clé : Spring Boot , l'architecture Microservice , Angular

Table des Matières

Remerciements	i
Dédicaces	ii
تلخيص.....	iii
Abstract.....	iii
Résumé	iv
Table des Matières	v
Table des Figures	viii
Liste des Tableaux.....	x
Introduction générale	1
1. Présentation de l'organisme d'accueil	5
1.1. Introduction.....	5
1.2. Présentation de centre universitaire Abdelhafid Boussouf - Mila	5
1.2.1. Apparition.....	5
1.2.2. Les Instituts et les Départements du centre universitaire	6
1.2.3. Organisation administrative du centre universitaire	7
1.2.4. Présentation du Vice-Rectorat du développement et de la prospective	8
1.2.5. Organigramme de la Direction adjointe du développement et de la prospective.....	9
1.3. Etude de l'existant.....	10
1.3.1. Evolution Domaines, divisions et nouvelles disciplines :.....	10
1.3.2. Evolution Accompagnateur pédagogiques et administratives.....	10
1.3.3. Evolution de l'encadrement pédagogique	11
1.3.4. Evolution du nombre total d'étudiants au stade de l'obtention du diplôme.....	11
1.3.5. Evolution du nombre total d'étudiants en post-diplôme.....	12
1.4. Gestion des différents programmes.....	12
1.4.1. Catégories de programmes	12
1.4.2. L'autorisation de programme(AP).....	13
1.4.3. Clôtures des opérations.....	13
1.5. Problématique.....	14

Table des matières

1.6. Objectif	14
1.7. Conclusion	14
2. Architecture des microservices	16
2.1. Introduction.....	16
2.2 Définition	16
2.2.1. Exemple d'une architecture Microservices	17
2.3. Avantages de microservices	18
2.4. Caractéristiques de microservices.....	19
2.5. Spring Cloud.....	20
2.5.1. Définition	20
2.5.2. Avantages de Spring cloud	20
2.6. Spring Security.....	20
2.6.1. Définition	20
2.6.2. Avantages de Spring Sécurité	21
2.6.3. Fonctions de sécurité en Spring	21
2.7. Conclusion	23
3. Spécification et analyse des besoins fonctionnels	25
3.1. Introduction.....	25
3.2. Le choix de la méthodologie.....	25
3.2.1. Rappel sur le processus unifié	25
3.2.2. Cycle de vie du processus unifié.....	27
3.2.3. Rappel sur le langage UML	28
3.2.4. Architecture logicielle du système	29
3.3. Découpage en microservices de notre système.....	29
3.3.1. Détails	29
3.4. Spécification des besoins fonctionnels	30
3.4.1. Identification des acteurs principaux	30
3.4.2. Identification des cas d'utilisations	30
3.4.3. Diagramme de cas d'utilisation	32
3.5. Conclusion	36
4. Conception du système	38
4.1. Introduction.....	38
4.2. Dictionnaire de données	38
4.3. Diagramme de classes	41

Table des matières

4.3.1. Diagramme de classes du système complet	42
4.3.2. Diagramme de classes de microservices	43
4. 4. Diagramme de séquence	46
4.4.1. Gestion des projets.....	46
4.5. Diagramme d'état transition	54
4.6. Le passage du diagramme de classe au modèle relationnel.....	55
4.6.1. Les règles de passages.....	55
4.6.2. Les tables de la base de données	56
4.7. Conclusion	57
5. Implémentation du système.....	59
5.1. Introduction.....	59
5.2. Présentation des outils et de langages utilisés.....	59
5.3. Description du système	62
5.4. Les interfaces.....	64
5.4.1. L'interface de connexion	64
5.4.2. L'interface de la gestion des projets.....	64
5.4.3. L'interface de détail d'un projet.....	65
5.4.4. L'interface ajouter projet	65
5.4.5. L'interface de la liste des avancements.....	66
5.4.6. L'interface du formulaire pour ajouter un marché	66
5.4.7. L'interface de la liste des évolutions	67
5.4.8. L'interface de de La situation mensuelle des programmes.....	67
5.5 Conclusion	68
Conclusion générale.....	70
Annexe 1 : Décision de clôture d'un programme.....	71
Annexe 2 : étapes de consultation	74
Annexe 3 : Les scénarios et les diagrammes de séquence et états-transitions	75
Annexe 4: Les interfaces du système	109
Bibliographie	111
Acronymes	113

Table des Figures

Figure 1.1 Organigramme de la Sous-Direction du développement et de la prospective	9
Figure 1.2 Encadrement pédagogique du centre universitaire Mila.....	11
Figure 1.3 L'évolution du nombre total d'étudiants au stade de l'obtention du diplôme.....	11
Figure 1.4 L'évolution du nombre total d'étudiants	12
Figure 2.1 Découpage d'une application en petits services	16
Figure 2.2 Exemple d'une architecture microservices	17
Figure 3.1 Déroulement du processus unifié	26
Figure 3.2 Les deux dimensions du Processus Unifié.....	27
Figure 3.3 Diagramme de cas d'utilisation	33
Figure 3.4 Diagramme de cas d'utilisation pour le microservice gestion des utilisateurs	34
Figure 3.5 Diagramme de cas d'utilisation pour le microservice gestion des statistiques.....	34
Figure 3.6 Diagramme de cas d'utilisation pour le microservice gestion des projets.....	35
Figure 4.1 Diagramme de classes de système complet.....	42
Figure 4.2 Diagramme de classes de microservice gestion des utilisateurs	43
Figure 4.3 Diagramme de classes de microservice gestion des statistiques.....	44
Figure 4.4 Diagramme de classes de microservice gestion des projets	45
Figure 4.5 Diagramme de séquence ajouter projet.	47
Figure 4.6 Diagramme de séquence consulter projet.....	49
Figure 4.7 Diagramme de séquence modifier projet	51
Figure 4.8 Diagramme de séquence supprimer projet	53
Figure 4.9 Diagramme de séquence consulter les projets	54
Figure 4.10 Diagramme d'état transition de l'objet projet.....	54
Figure 5.1 Logo du framework spring boot	59
Figure 5.2 Logo de l'outil spring cloud	59
Figure 5.3 Logo du langage de programmation java.....	60
Figure 5.4 Logo de Maven	60
Figure 5.5 Logo de Postman	60
Figure 5.6 Logo de Framework Angular	60
Figure 5.7 : Logo du logiciel staruml.....	61

Table des figures

Figure 5.8 Logo du editeur visual studio	61
Figure 5.9 Logo du serveur Web apache Tomcat.....	61
Figure 5.10 Logo de Framework Node.js.....	62
Figure 5. 11 Logo du gestionnaire des packages NPM	62
Figure 5.12 Logo du système de la base de données	62
Figure 5.13 Architecture de l'application	63
Figure 5.14 Les microservices.....	63
Figure 5.15 Interface de connexion.....	64
Figure 5.16 Interface de la gestion des projets	64
Figure 5.17 Interface de détail d'un projet	65
Figure 5.18 Interface ajouter projet	65
Figure 5.19 Interface de la liste des avancements	66
Figure 5.20 Interface ajouter marché.....	65
Figure 5.21 Interface de la liste des évolutions.....	67
Figure 5.22 Interface de La situation mensuelle des programmes	67

Liste des Tableaux

Tableau 1.1 Domaines, divisions et nouvelles disciplines	10
Tableau 1.2 Accompagnateur pédagogiques et administratives	10
Tableau 3.1 Table des cas d'utilisation pour l'acteur opérateur	31
Tableau 3.2 Table des cas d'utilisation pour l'acteur administrateur	31
Tableau 4.1 Dictionnaire de données.....	41
Tableau 4.2 Description de cas d'utilisation ajouter projet	46
Tableau 4.3 Description de cas d'utilisation consulter projet.....	48
Tableau 4.4 Description de cas d'utilisation modifier projet	50
Tableau 4.5 Description de cas d'utilisation supprimer projet	52
Tableau 4.6 Description de cas d'utilisation Consulter la liste des projets	53

Introduction générale

Introduction générale

La plupart des administrations algériennes souffrent de manque des systèmes d'informations sécurisées permettant la gestion des différents dossiers administratives. De ce fait, l'état algérien a généralisé la numérisation de l'administration dans tous les ministères et les établissements publics cette année. L'objectif principal de la numérisation est de minimiser le cout des tâches administratives et l'effort des citoyens et des employées, de rapprocher l'administration au citoyen et d'assurer des services de bonne qualité.

Dans ce contexte, nous avons choisi la sous-direction du développement et de la prospective (SDDP) du centre universitaire Abdelhafid Boussouf – Mila comme lieu de notre stage. Après notre étude dans la SDDP, nous avons remarqué l'absence d'un système d'information ce qui rend la recherche des informations très difficiles et la génération des statistiques et le suivi des différents programmes de manière manuelle.

Dans ce travail, nous proposons un système d'information sécurisé pour le suivi du développement local au niveau du ministère de l'enseignement supérieur et la recherche scientifique basé sur l'architecture microservices. Ce système permettra aux responsables du secteur de l'enseignement supérieur de contrôler et de superviser toutes les opérations et les aident également à prendre les décisions appropriées.

Ce mémoire est organisé comme suit. Il comprend cinq (05) chapitres :

Le premier chapitre décrit la sous-direction du développement de la prospective de centre universitaire Mila, où nous avons effectué notre stage et qui permet d'expliquer les concepts de base liés au cycle de vie d'un projet.

Le deuxième chapitre présente l'architecture microservices que nous avons utilisé pour la réalisation de notre projet.

Le troisième chapitre intitulé spécification et analyse des besoins fonctionnels, il contient les diagrammes de cas d'utilisation des microservices : gestion des projets, statistiques et utilisateurs.

Introduction générale

Le quatrième chapitre est entièrement consacré à la conception du système. Il contient les diagrammes de séquence, d'états-transitions, de classe, et le modèle relationnel.

Le cinquième chapitre présente les outils, les langages de programmation et l'environnement de développement que nous avons utilisé et expose quelques interfaces de l'application.

Chapitre 01 : Présentation de l'organisme d'accueil

1. Présentation de l'organisme d'accueil

1.1. Introduction

L'étude de l'existant est la première phase du projet informatique. Son but ultime est de définir le périmètre de notre enquête ou de notre stage.

Dans ce chapitre, premièrement, nous présentons le centre universitaire Abdelhafid Boussouf - Mila, leurs instituts, départements et aussi les organisations administratives. Ensuite, nous décrivons la gestion des différents programmes du développement. Finalement, nous expliquons les problèmes que nous avons trouvés dans la gestion des programmes.

1.2. Présentation de centre universitaire Abdelhafid Boussouf - Mila

Le centre universitaire Abdelhafid Boussouf - Mila est une institution publique à caractère scientifique, culturel, moralement et financièrement indépendant. Il offre une formation scientifique qualitative aux étudiants dans différents domaines. Il a été créé par décret exécutif n°08-204 du 06 Rajab 1429 correspondant au 09 juillet 2008. Il a ouvert ses portes pendant la saison académique 2008/2009 à plus de 1000 étudiants pour être la première institution universitaire à lancer le secteur de l'enseignement supérieur et de la recherche scientifique dans l'État et selon la décision n ° 0 1/04 du 29 Dhu al-Hijjah correspondant au 23 octobre 2014 publiée par le ministère des Moudjahidines, qui comprend la dédicace des institutions universitaires, le Centre universitaire de Mila a été rebaptisé Moudjahid Abdelhafid Boussouf, le premier de novembre 2014, l'anniversaire de la Révolution de libération nationale [1].

1.2.1. Apparition

Il est situé à 5 km du centre-ville de la wilaya de Mila sur la route de zeghaia, et il occupe une superficie d'environ 87 hectare [2].

1.2.2. Les Instituts et les Départements du centre universitaire

Le centre universitaire de Mila offre de nombreuses formations dans les trois cycles : Licence, Master et Doctorat, et dans divers domaines scientifiques et technologiques. Les formations sont assurées par une sélection de compétences scientifiques dans différentes disciplines, qui supervisent la formation et l'encadrement des étudiants.

Ces formations sont réparties sur six instituts

- **Institut des Sciences et Technologie :**

Constitué dans deux Départements :

- Département de Génie Civil et Hydraulique.
- Département de Génie Mécanique et Électromécanique.

- **Institut des sciences économiques, commerciales et de gestion :**

Organisé dans Cinq départements :

- Département Sciences Économiques et Commerciales.
- Département des Sciences de Gestion.
- Département Tronc Commun SECG.
- Département Sciences Financières et Comptabilité.
- Département du droit.

- **Institut des lettres et des langues :**

Constitué de deux départements :

- Département de la langue arabe
- Département de langues étrangères.

- **Institut de Mathématiques et informatique.**
- **Institut des sciences de la nature et de la vie.**
- **Institut du droit et des sciences politiques.**

- **Laboratoires de recherche :**

Pour organiser et soutenir la recherche scientifique, l'université de Mila abrite Quatre laboratoires de recherche sous sa tutelle :

- Le Laboratoire de Mathématiques et de leurs Interactions.
- Le Laboratoire de Sciences Naturelles et des Matériaux.
- Le Laboratoire des études sur les Stratégies de Diversification Economique pour le Développement Durable.
- Laboratoire d'études critiques et littéraires.

1.2.3. Organisation administrative du centre universitaire

a) Le recteur

Le recteur est responsable du fonctionnement général de l'université.

b) Rectorat

- Cabinet Directeur.
- Cellule de statistique.

c) Vice Rectorats

- Vice-Rectorat de la formation supérieure, de la formation continue et des diplômes.
- Vice-Rectorat de la poste-graduation, de la recherche scientifique et des relations extérieurs.
- Vice-Rectorat du développement et de la prospectiviste.

d) Les instituts

- Institut des lettres et des langues.
- Institut des sciences et de la technologie.
- Institut des sciences économiques, commerciales de la gestion.

e) L'administration centrale.

- Secrétariat général.
- Sous-direction des personnels et de la formation.
- Sous-direction du budget et de la comptabilité.
- Services techniques commun de l'université.

f) Unité de la médecine préventive.

g) Bibliothèques.

1.2.4. Présentation du Vice-Rectorat du développement et de la prospective

Dans ce qui suit, on présente le Vice-Rectorat du développement et de la prospective.

a) Missions du Vice-Rectorat :

La Sous-direction du développement et de la prospective est chargée des tâches suivantes :

- Collecte des articles nécessaires.
- Préparer les plans de développement du Centre Universitaire.
- Mener des études prospectives dans le domaine du développement du recensement des étudiants et suggérer des actions pour les soutenir, en particulier dans le domaine de l'encadrement pédagogique et administratif.
- Mise à jour de la fiche statistique du Centre universitaire.
- Préparation de piliers d'information dans les parcours éducatifs garantis par le Centre Universitaire et ses instances professionnelles.
- Améliorer les activités d'information des étudiants, en particulier celles qui les aident à choisir leur orientation.

b) Les services de la sous-direction :

Assurer le suivi des programmes de construction et assurer la mise en œuvre des programmes pour équiper le centre universitaire en contact avec les intérêts concernés.

- **Département de statistique et de surveillance :**

Le Département est chargé de désigner une carte statistique pour le Centre universitaire. Elle mène également des études prospectives dans le domaine du recensement des étudiants. Il propose des mesures pour les soutenir, y compris le domaine de leur compagnie aérienne pédagogique et administrative.

- **L'intérêt des médias et des conseils :**

La mission de cet intérêt est de préparer les piliers médias dans les parcours éducatifs garantis par le Centre Universitaire et ses organes professionnels.

- **Programmes de construction et de traitement de soutien :**

Le Département assure le suivi des programmes et assure la mise en œuvre des programmes pour équiper le Centre universitaire en assurant la liaison avec les départements concernés.

1.2.5. Organigramme de la Direction adjointe du développement et de la prospective

La figure 1.1 représenter l'organigramme de la Sous-Direction du développement et de la prospective du centre universitaire Abdelhafid Boussouf - Mila

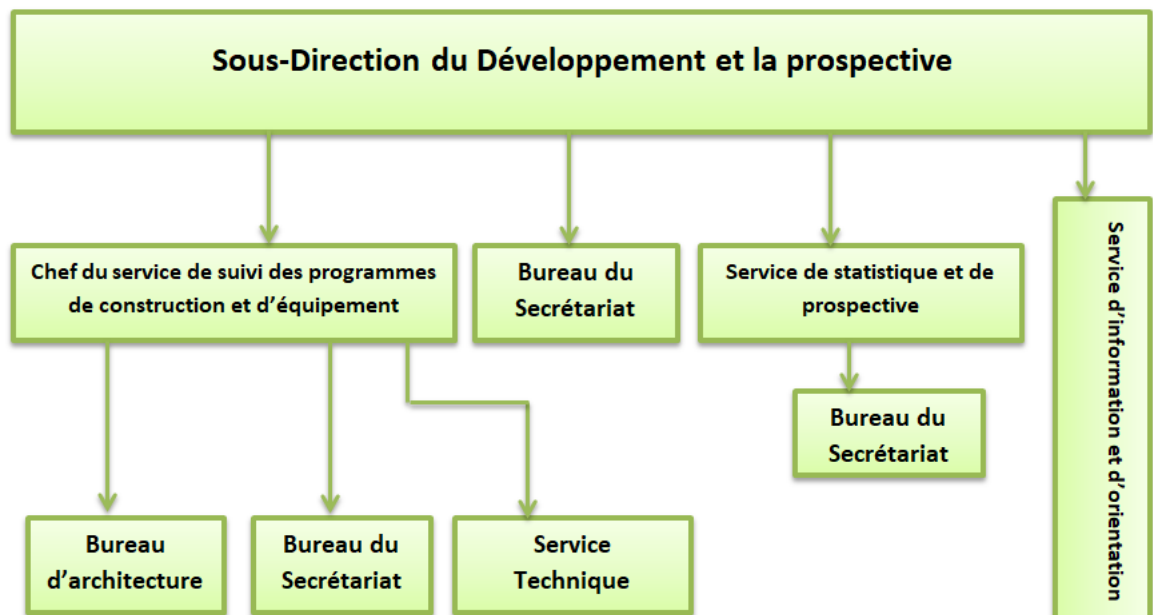


Figure 1.1 Organigramme de la Sous-Direction du développement et de la prospective

Chapitre 01 : présentation de l'organisme d'accueil

1.3. Etude de l'existant

1.3.1. Evolution Domaines, divisions et nouvelles disciplines :

domaine	filière	spécialité	type
ST	génie des procédés	génie des procédés	Licence académique
MI	mathématiques	Mathématiques appliquées	Licence académique
MI	Informatique	L'intelligence artificielle	Licence académique
SNV	les sciences	Microbiologie	Licence académique
STU	Sciences de la terre et de l'univers	géologie	Licence académique

Tableau 1.1 Domaines, divisions et nouvelles disciplines

1.3.2. Evolution Accompagnateur pédagogiques et administratives

Estimation du nombre de places pédagogiques 8000 places.

utilitaire	le nombre	Capacité
les amphi	14	4000 place
les salles de cours/TD/TP	92	4000 P
Laboratoires pédagogiques pour applications en sciences exactes	20	400
Laboratoires des technologies des médias et de la communication	10	200 P
Laboratoires d'enseignement des langues	04	80 P
Bibliothèques	04	2050 P
salle de discussion des mmoires de fin études	10	1000 P
Salle Internet	05	260 P
Administration centrale	01	/

Tableau 1.2 Accompagnateur pédagogiques et administratives

1.3.3. Evolution de l'encadrement pédagogique

Le nombre d'enseignants est de 439 professeurs.

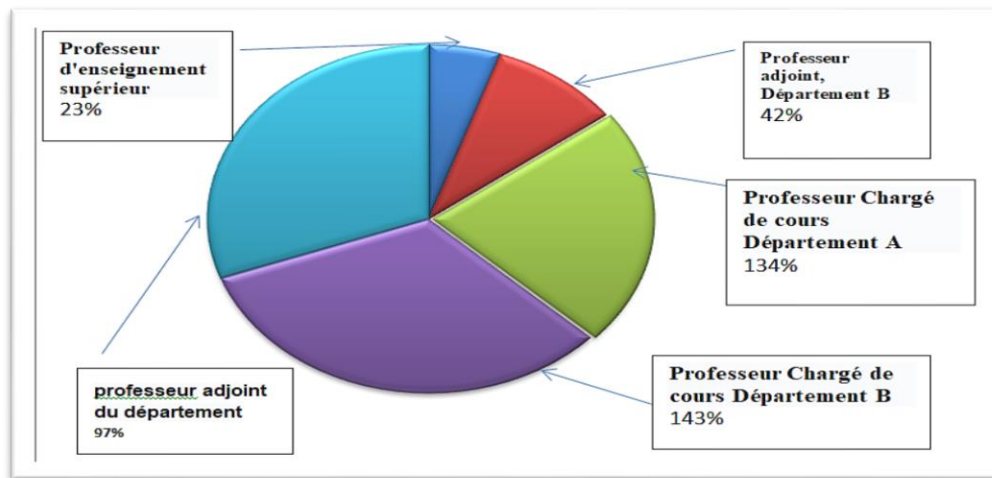


Figure 1.2 Encadrement pédagogique du centre universitaire Mila

1.3.4. Evolution du nombre total d'étudiants au stade de l'obtention du diplôme

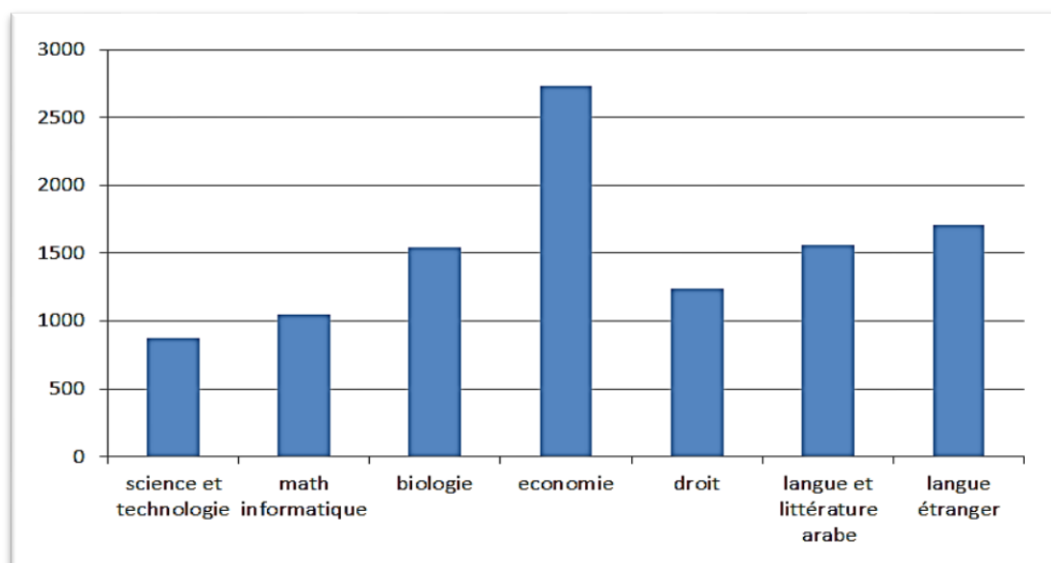


Figure 1.3 L'évolution du nombre total d'étudiants au stade de l'obtention du diplôme

1.3.5. Evolution du nombre total d'étudiants en post-diplôme

Cette saison sera le nombre d'inscriptions au doctorat 246 etudiant.

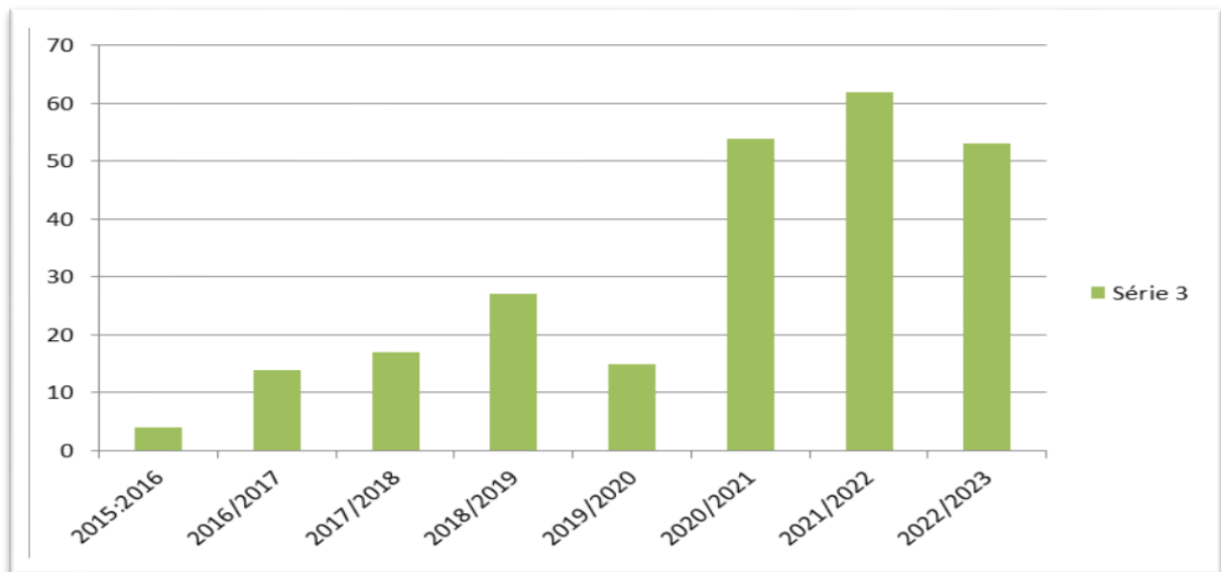


Figure 1.4 L'évolution du nombre total d'étudiants

1.4. Gestion des différents programmes

1.4.1. Catégories de programmes

La mise en œuvre de l'autorisation de projet et le décaissement des subventions dépend de la nature du programme. Les programmes se répartissent en trois grandes catégories [3]:

- Les programmes sectoriels centralisés (PSC).
- Les programmes sectoriels déconcentrés (PSD).
- Les plans communaux de développement (PCD).

- **Les programmes sectoriels centralisés (PSC)**

Le PSC est défini pour les grands projets comme le projet de la construction du barrage Beni Haroun. Ces programmes appartiennent aux administrations centrales.

- **Les programmes sectoriels déconcentrés (PSD)**

Pour un secteur donné, le PSD est constitué de l'ensemble des programmes inscrits au Catalogue National des Dépenses d'Équipement Public.

- **Les plans communaux de développement (PCD)**

Le PCD est un programme appartenant aux communes par exemple : l'approvisionnement en eau potable, l'éclairage public, les réseaux d'assainissement ect.

➤ **Les ordonnateurs**

Toute personne qui est qualifiée pour exécuter les procédures d'attestation de quittance et de liquidation, les procédures d'engagement et de liquidation, et les ordres de paiement des dépenses, dès lors que la personne est désignée pour exercer les fonctions de ces procédures, la loi lui confère la capacité de payer les ordres et cette capacité avec cette fois la fin de l'emploi et cesser d'exister.

1.4.2. L'autorisation de programme(AP)

L'autorisation de programme accompagnée d'une fiche d'engagement est prise en charge sous forme de décision d'individualisation (inscription), de réévaluation, de dévaluation, de modification des caractéristiques, de modification de la structure de coût et de clôture, par le Contrôleur Financier de Wilaya ou de la Commune selon qu'il s'agit du programme sectoriel déconcentré (PSD) ou plans communaux de développement (PSD) [3].

1.4.3. Clôtures des opérations

Les opérations sont fermées de la même manière qu'elles sont enregistrées.

La fermeture peut se produire dans les situations suivantes :

- Achèvement d'un programme ou d'un projet.
- L'arrêt définitif de la réalisation pour tout autre motif.

1.5. Problématique

Après notre étude dans la Sous-Direction du développement et de la prospective, nous avons remarqué les problèmes suivants :

- Les travailleurs utilisent juste des logiciels simples comme Word et Excel.
- Recherche manuelle des informations donc beaucoup du temps perdu.
- Suivre l'avancement des programmes manuellement c'est à dire Absence d'un système informatique pour le suivi du projet.
- Difficulté de recueillir des statistiques à partir de multiples emplacements au Centre universitaire.

1.6. Objectif

Après avoir identifié les difficultés dans la Sous-Direction du développement et de la prospective, notre solution est de concevoir et de mettre en œuvre un système qui facilite et organise le travail des employés. En plus, ce système permettra au chef de projet de suivre du projet et la connaissance de leur situation. Pour mieux développer le projet, nous avons choisi l'architecture microservices qui divise l'application en sous services indépendants.

1.7. Conclusion

Dans ce chapitre, nous avons présenté le centre universitaire Abdelhafid Boussouf - Mila et notamment le vice-rectorat de la Sous-Direction du développement et de la prospective, lieu de notre stage. Ensuite, nous avons présenté la situation de l'évolution du secteur de l'enseignement supérieur ainsi que les différents programmes du développement. Enfin, nous avons identifié les problèmes liés au suivi de l'évolution du secteur de l'enseignement supérieur et suggéré des solutions pour améliorer le suivi du développement de ce secteur.

Chapitre 2 :

Architecture des

microservices

2. Architecture des microservices

2.1. Introduction

Contrairement à l'approche homogène traditionnelle (modèle classique) qui est considéré comme une unité unifiée unique généralement basée sur une seule base de données et toute modification va affecter l'ensemble du système. Les Microservices décomposent une application pour isoler les fonctions clés. Chacune de ces fonctions est appelée « service » et ces services peuvent être développés et déployés indépendamment les uns des autres. Par conséquent, tout le monde peut travailler (ou mal fonctionner) sans affecter les autres.

Dans ce chapitre, premièrement, nous donnons un aperçu général sur l'architecture Microservices. Ensuite, nous introduisons spring cloud ainsi que ses avantages. Enfin, nous présentons spring sécurité et ses avantages.

2.2 Définition

Les Microservices sont à la fois une architecture et une approche moderne du logiciel dans laquelle le code d'application est livré en petites pièces maniables, indépendantes les unes des autres. Chaque Microservice peut utiliser sa propre base de données, et peut être développé à l'aide d'un langage [4].

Pour communiquer les uns avec les autres, les microservices d'une application se servent du modèle de communication requête-réponse. L'implémentation typique utilise des appels API REST basés sur le protocole HTTP [4].



Figure 2.1 Découpage d'une application en petits services

[5]

Chapitre 02 : Architecture des microservices

L'architecture Microservices offre une solution simple. Plus précisément, elle permet de découper une : application en petits services, appelés Microservices, parfaitement autonome. Cependant, une architecture monolithique est une application dont l'intégralité du code et de la fonctionnalité est implémentée en un seul programme [4].

2.2.1. Exemple d'une architecture Microservices

La figure 2.2 montre l'architecture d'une application de gestion d'un restaurant basé sur les Microservices. Cette application est implémentée avec Spring boot et se compose des modules suivants : vendeur de tickets, statistiques des accès au restaurant, étudiant et contrôle d'accès au restaurant. En plus, chaque module utilise sa propre base de données [4].

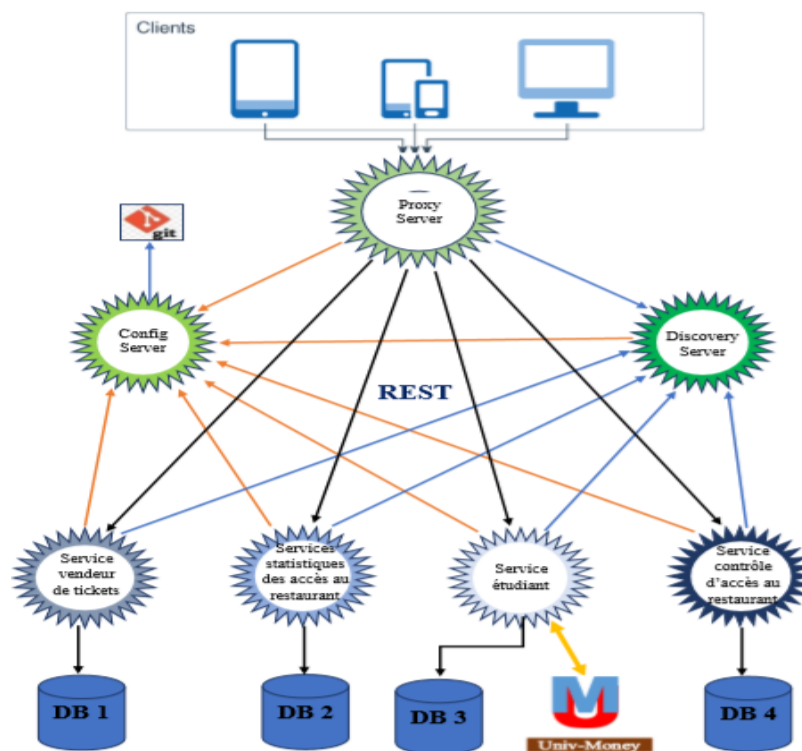


Figure 2.2 Exemple d'une architecture microservices

2.3. Avantages de microservices

Dans ce qui suit, nous présentons les avantages des Microservices :

- **Utilisation d'outils adaptés au travail :**

En raison de l'indépendance des microservices, plusieurs langages de programmation peuvent être utilisés, ce qui permet d'utiliser le bon outil pour le travail. Cela permet également de modifier les langages de mise en œuvre en fonction de l'évolution de la demande et de la technologie.

- **Sécurité :**

Tout problème ou défaillance survenant sur un microservice affecte uniquement celui-ci et n'a aucun impact sur l'ensemble de l'application.

- **Disponibilité :**

Les fonctionnalités de l'application sont réparties entre plusieurs services. Si un microservice ne fonctionne pas, les fonctionnalités des autres microservices sont toujours disponibles.

- **Flexibilité :**

Les microservices sont bien plus simples d'utilisation que les applications monolithiques. Ainsi, il est possible d'augmenter ou de réduire l'usage du Cloud en fonction de la charge de l'application. Ils facilitent aussi l'introduction de nouvelles structures, bases de données et autres ressources.

- **Scalabilité :**

Les architectures de microservice peuvent évoluer indépendamment à l'aide de clusters, de groupes...

- **Les microservices définissent les API**

Qui exposent leurs fonctions à chaque client. Il pourrait même s'agir d'autres applications.

- **Plus facile à mettre en œuvre**

Déployez par tranches pour éviter d'interférer avec d'autres services [7].

- **Plus facile à comprendre :**

Parce que la fonction est isolée et moins dépendante, le code est plus facile à suivre [7].

2.4. Caractéristiques de microservices

La plupart des systèmes de microservices partagent certaines caractéristiques. Vous trouverez ci-dessous les six caractéristiques principales des microservices [4]:

- composants multiples .
- approche métier.
- routage simple.
- Décentralisée.
- résistante aux pannes .
- évolutive.

En plus d'autres caractéristiques secondaires comme [6] :

- Petite taille.
- Messagerie activée.
- Déployables indépendamment.
- Construits et publiés avec des processus automatisés.

À grande échelle, votre application Spring se compose de plusieurs modules qui fonctionnent indépendamment les uns des autres. **Spring Cloud** et **spring Security** sont deux de ces modules

2.5. Spring Cloud

2.5.1. Définition

Spring cloud est un ensemble de bibliothèques permettant de créer des applications qui s'exécutent sur site ou dans le cloud par exemple : Cloud Foundry, AWS, etc [3].

Certaines des bibliothèques Spring Cloud pour les microservices sont :

- Spring Cloud Netflix.
- Spring Cloud Consul.

2.5.2. Avantages de Spring cloud

Spring Cloud aide les développeurs à créer rapidement des applications distribuées. Nous sommes confrontés à de nombreux problèmes lorsque nous travaillons avec des systèmes distribués, et ce framework aide avec certains des modèles communs dans ce domaine [4] :

- Découverte d'un service distribué.
- Équilibrage.
- Routage.
- Défaillances de service.
- Gestion de la configuration, etc.

2.6. Spring Security

2.6.1. Définition

Spring Sécurité est un framework qui permet à un programmeur d'utiliser des composants JEE pour définir des limitations de sécurité sur les applications Web basées sur le framework Spring. Plus précisément, Spring sécurité est une bibliothèque qui peut être utilisée et personnalisée pour répondre aux exigences du programmeur. Parce qu'il fait partie de la même famille Spring que Spring Web MVC, il fonctionne bien ensemble. Sa fonction principale est de gérer l'authentification et l'autorisation au niveau de la requête Web et de l'appel de méthode [9].

Spring Sécurité travaille sur les quatre concepts de base suivants :

- Authentification.
- Autorisation.
- Stockage des mots de passe.
- Filtres de servlet.

2.6.2. Avantages de Spring Sécurité

Voilà les principaux avantages de la sécurité Spring [9] :

- ✓ Protection contre les attaques telles que la fixation de session et le détournement de clic.
- ✓ Intégration Spring MVC.
- ✓ Prise en charge de la configuration Java.
- ✓ Portatif.
- ✓ Intégration de l'API Servlet.
- ✓ Protégez-vous contre les attaques par force brute.

2.6.3. Fonctions de sécurité en Spring

Dans ce qui suit, nous présentons certains des principales fonctions de sécurité en Spring [9] :

- **Autorisation :**

L'autorisation permet à l'utilisateur d'être autorisé avant d'accéder aux ressources. Elle permet aux développeurs de définir des contrôles d'accès pour les ressources.

- **Authentification unique :**

Cette fonctionnalité permet à un utilisateur d'utiliser un seul compte pour accéder à différentes applications (nom d'utilisateur et mot de passe).

- **Localisation de logiciels :**

Cette fonctionnalité nous permet de créer des interfaces utilisateur pour des applications dans n'importe quel langage.

- **Souviens-toi de moi :**

Avec l'aide des cookies HTTP, Spring Security fournit cette capacité. Il se souvient de l'utilisateur et l'empêche de se connecter à partir du même poste de travail jusqu'à ce qu'il se déconnecte.

- **LDAP (Lightweight Directory Access Protocol) :**

Il s'agit d'un protocole d'application ouvert pour la gestion et l'interaction avec des services d'information d'annuaire dispersés sur le protocole Internet.

- **Module de connexion JAAS (Java Authentication and Authorization Service) :**

Il s'agit d'un module d'authentification enfichable basé sur Java. Il est pris en charge par la procédure d'authentification de Spring Security.

- **Authentification par formulaire Web :**

Les formulaires Web capturent et authentifient les informations d'identification de l'utilisateur à partir du navigateur Web au cours de cette procédure. Bien que nous souhaitions créer une authentification par formulaire Web, Spring Security la prend en charge.

- **Authentification par accès Digest :**

Nous pouvons rendre la procédure d'authentification plus sûre avec cette fonctionnalité qu'avec l'authentification d'accès de base. Avant de fournir des données sensibles sur le réseau, il demande au navigateur de vérifier l'identité de l'utilisateur.

- **Autorisation http :**

En utilisant des chemins Apache Ant ou des expressions régulières, Spring fournit cette fonctionnalité pour l'autorisation HTTP des URL de requête Web.

- **Authentification d'accès de base :**

Spring Security prend en charge l'authentification d'accès de base, qui est utilisée pour donner un nom d'utilisateur et un mot de passe lors de l'exécution de demandes réseau.

2.7. Conclusion

Dans ce chapitre, nous avons donné un aperçu général sur l'architecture microservices qui est une approche de développement informatique qui est devenue très populaire ces dernières années. Ensuite, nous avons introduit Spring Cloud ainsi que ses avantages. Enfin, nous avons présenté Spring sécurité, ses avantages et ses fonctions.

Chapitre 03 :

Spécification et analyse des besoins fonctionnels

3. Spécification et analyse des besoins fonctionnels

3.1. Introduction

La spécification et l'analyse des besoins sont la première étape du cycle de développement logiciel. Dans ce chapitre, nous présentons d'abord le processus unifié (UP) et le langage de modélisation UML (*Unified Modeling Language*). Ensuite, nous spécifions les besoins que l'application doit satisfaire. Enfin, nous présentons les diagrammes de cas d'utilisation.

3.2. Le choix de la méthodologie

Plusieurs méthodes de développement logiciel sont basées sur UML comme méthode : UP, 2TUP ...

Entre autres méthodes, nous choisissons la méthode UP (processus unifié).

3.2.1. Rappel sur le processus unifié

Dans ce qui suit, nous rappelons le processus unifié UP.

3.2.1.1. Définition du processus unifié

Un processus unifié (Unified Processus) est un processus de développement logiciel construit sur UML. Il est itératif et incrémental, centré sur l'architecture, conduit par les cas d'utilisation et piloté par les risques [10].

3.2.1.2. Les caractéristiques du processus unifié

Tout processus UP répond aux caractéristiques ci-après [11] :

✓ UP est itératif et incrémental

Le projet est divisé en plusieurs étapes à court terme afin de mieux suivre l'évolution générale.

À la fin de chaque itération, une partie exécutable du système fini est créée progressivement (voir Figure. 3.1).

Chapitre 03 : Spécification et analyse des besoins fonctionnels

- **UP est centré sur l'architecture**

Tout système complexe doit être divisé en composants modulaires afin de faciliter sa maintenance et son développement. Cette structure doit être conçue en UML, pas seulement documentée dans le texte.

- **UP est guidé par les cas d'utilisation d'UML**

La réalisation du projet prend en compte les besoins et les exigences des utilisateurs. Les cas d'utilisation du futur système sont définis, correctement décrits et hiérarchisés.

- **UP est piloté par les risques**

Les principaux risques du projet doivent être identifiés le plus tôt possible mais surtout éliminés le plus rapidement possible, les actions à entreprendre dans ce cadre déterminent l'ordre des itérations.

- **UP est orienté composant**

Au niveau de la modélisation et de la production, c'est un gage de flexibilité pour le modèle lui-même et le logiciel qu'il représente. Cette pratique est le support nécessaire à la réutilisation des logiciels et offre de belles perspectives de gains [10].

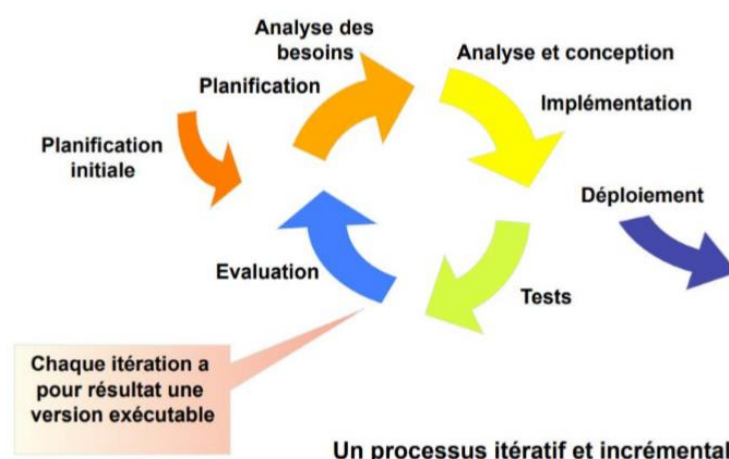


Figure 3.1 Déroulement du processus unifié

[11]

3.2.2. Cycle de vie du processus unifié

Le but du processus unifié est de maîtriser la complexité des projets et de réduire les risques, un ensemble de principes généraux adaptés à la nature spécifique des projets et de gérer le processus de développement à travers deux axes (figure 3.2).

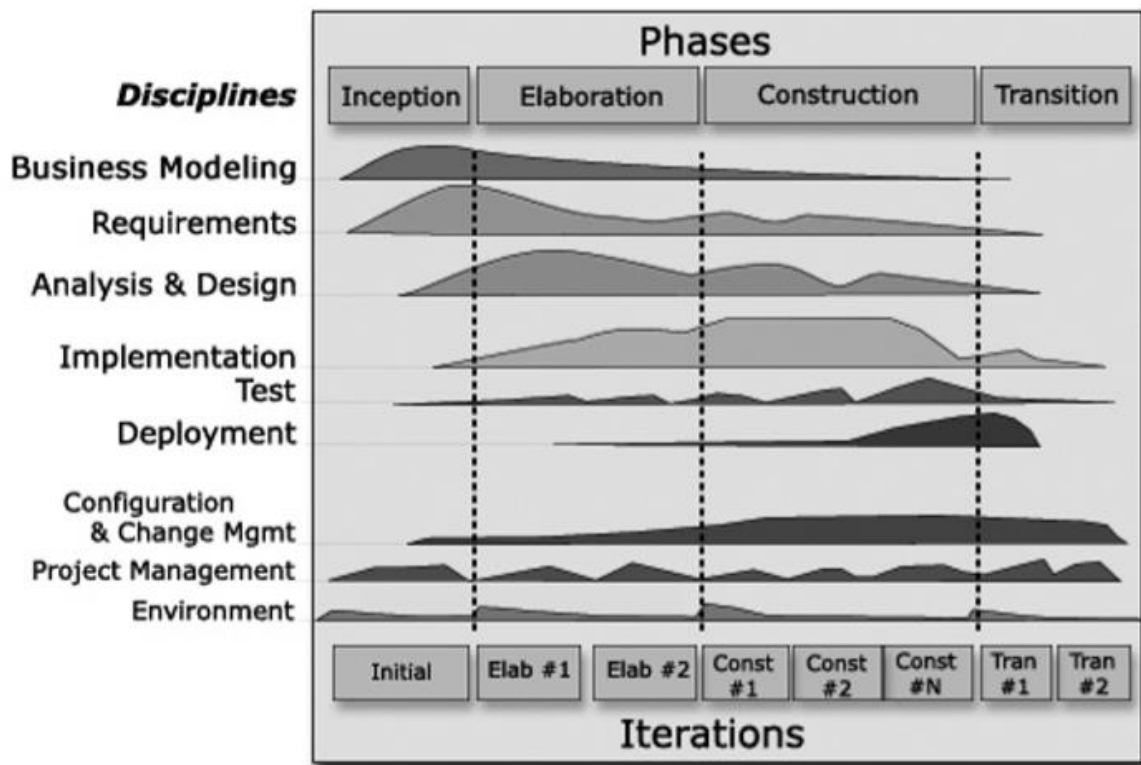


Figure 3.2 Les deux dimensions du Processus Unifié

[12]

Axe vertical : représente la séquence principale d'activités, qui regroupait les activités selon leur nature. Cette dimension représente l'aspect statique du processus qui est exprimé en termes de composants, de processus, d'activités, de séquences, d'artefacts et de travailleurs.

Axe horizontal : représente le temps et montre le chemin du cycle de vie du processus, cette dimension représente l'aspect dynamique du processus qui s'exprime en termes de cycles, d'étapes, de fréquences et de jalons

Chapitre 03 : Spécification et analyse des besoins fonctionnels

Pour réaliser efficacement un tel cycle, les développeurs ont besoin de toutes les représentations du produit logiciel.

- Un modèle de cas d'utilisation.
- Un modèle d'analyse : détailler les cas d'utilisation.
- Un modèle de conception : finissant la structure statique du système sous forme de sous-systèmes, de classes et interfaces.
- Un modèle d'implémentation : intégrant les composants.
- Un modèle de déploiement : définissant les nœuds physiques des ordinateurs.
- Un modèle de test : décrivant les cas de test vérifiant les cas d'utilisation.
- Une représentation de l'architecture.

3.2.3. Rappel sur le langage UML

3.2.3.1. Définition d'UML

UML (Unified Modeling Language) est défini comme un langage de modélisation graphique et textuelle permettant de comprendre et de décrire les besoins, d'identifier et de documenter les systèmes, de dessiner des architectures logicielles, de concevoir des solutions et de communiquer des points de vue [10].

3.2.3.2. Définition d'un diagramme

Un diagramme UML est une représentation graphique qui prend en compte un aspect particulier du modèle. Chaque type de diagramme UML dispose de sa propre structure et sémantique spécifique [12].

Pour modéliser notre projet, nous avons utilisé différents types de diagrammes (diagrammes dynamiques et diagrammes statiques) :

- **Diagramme de cas d'utilisation.**
- **Diagramme de classe.**
- **Diagramme de séquence.**
- **Diagramme d'état transition.**

3.2.4. Architecture logicielle du système

La structure du logiciel décrit de manière symbolique et schématique les différents éléments d'un ou plusieurs systèmes informatiques, leurs interrelations et interactions [4].

Pour ce projet, nous utiliserons l'architecture de microservices. Les architectures de microservices sont la « nouvelle norme ».

La création de petites applications indépendantes et prêtes à l'emploi peut offrir une grande flexibilité et augmenter la flexibilité de notre code.

3.3. Découpage en microservices de notre système

Le principe est de mettre en place un système divisé en trois (3) parties. Chaque partie représente un microservice.

- Une partie pour la gestion des statistiques.
- Une partie pour la gestion de projets.
- Une partie pour la gestion des utilisateurs.

3.3.1. Détails

a. Gestion des statistiques

Il sera constitué de :

- Page pour consulter toutes les évolutions.
- Page d'ajout des évolutions.
- Page pour modifier les évolutions dans une année.
- Page d'ajout d'une nouvelle situation des programmes.
- Page pour consulter les situations des programmes.
- Page pour modifier les informations d'une situation des programmes.

b. Gestion de projets

Il sera constitué de :

- Page pour consulter la liste des projets.
- Page d'ajout d'un nouveau projet.
- Page pour modifier les informations d'un projet.
- Page pour consulter les projets en cours et les projets clôtures.

Chapitre 03 : Spécification et analyse des besoins fonctionnels

- Page d'ajout d'un nouvel avancement pour un projet.
- Page pour consulter tous les avancements pour chaque projet.
- Page pour modifier les informations d'un avancement.
- Page pour consulter tous les lots pour chaque projet de chaque entité.
- Page d'ajout d'un nouvel lot pour un projet.
- Page pour modifier les informations d'un lot.
- Page pour consulter tous les marchés pour chaque projet.
- Page d'ajout d'un nouvel marché pour un projet.
- Page pour modifier les informations d'un marché.

c. Gestion des utilisateurs

Il sera constitué de :

- Une page pour l'authentification.
- Une page pour consulter tous les utilisateurs existant dans la base de données.
- Page de modifier les informations d'un utilisateur.

3.4. Spécification des besoins fonctionnels

Cette étape a pour but de comprendre le contexte du système. Il s'agit de déterminer les fonctionnalités et acteurs les plus pertinents, de spécifier les risques les plus critiques et d'identifier les cas d'utilisation initiaux.

3.4.1. Identification des acteurs principaux

L'identification des acteurs est une étape importante dans un projet de développement. Le système suppose deux types d'acteurs :

- **Opérateur.**
- **Administrateur.**

3.4.2. Identification des cas d'utilisations

Nous avons mené plusieurs interviews afin de déterminer au mieux les besoins de l'application, afin de répondre aux attentes des utilisateurs. Nous sommes allés chercher des informations au niveau du Vice-Rectorat du Développement et de la Prospective. Cette étape correspond à des recherches de terrain pour définir le cadre de notre système.

Chapitre 03 : Spécification et analyse des besoins fonctionnels

Les cas d'utilisations c'est un ensemble d'actions réalisées par le système qui donne un résultat observable pour chaque acteur.

Dans le tableau suivant, nous présentons l'acteur opérateur avec ses fonctionnalités.

Utilisateurs	Cas d'utilisation
Opérateur	L'application doit permettre à l'opérateur : 1. Authentification. 2. Ajouter un projet. 3. Consulter projet. 4. Modifier un projet. 5. Supprimer un projet. 6. Consulter les projets. 7. Ajouter un avancement. 8. Modifier un avancement. 9. Supprimer un avancement. 10. Consulter les avancements. 11. Ajouter un marché. 12. Modifier un marché. 13. Supprimer un marché. 14. Consulter les marchés. 15. Ajouter un lot. 16. Modifier un lot. 17. Supprimer un lot. 18. Consulter les lots. 19. Créer une évolution. 20. Modifier une évolution. 21. Supprimer une évolution. 22. Consulter les évolutions. 23. Créer une situation des programmes. 24. Modifier une situation des programmes. 25. Supprimer une situation des programmes. 26. Consulter les situations des programmes. 27. Rechercher projet.

Tableau 3.1 Table des cas d'utilisation pour l'acteur opérateur

Dans le tableau suivant, nous présentons l'acteur administrateur avec ses fonctionnalités.

Utilisateurs	Cas d'utilisation
Administrateur	L'administrateur possède les fonctionnalités de l'opérateur ainsi que les fonctionnalités suivantes : 1. Ajouter un utilisateur. 2. Modifier un utilisateur. 3. Supprimer un utilisateur. 4. Consulter la liste des utilisateurs.

Tableau 3.2 Table des cas d'utilisation pour l'acteur administrateur

3.4.3. Diagramme de cas d'utilisation

Il représente un ensemble de séquences d'actions effectuées par le système qui produisent un résultat notable d'intérêt pour un acteur particulier. Un cas d'utilisation modélise un service rendu par le système. Il exprime les interactions acteurs/système et apporte une valeur ajoutée « notable » à l'acteur concerné [12].

Pour ce travail, chaque microservice possède un diagramme de cas d'utilisation. Les figures suivantes : 3.3, 3.4, 3.5 et 3.6 représentent les diagrammes de cas d'utilisations de système complet ainsi que des différents microservices du système.

3.4.3.1. Diagramme de cas d'utilisation du système complet

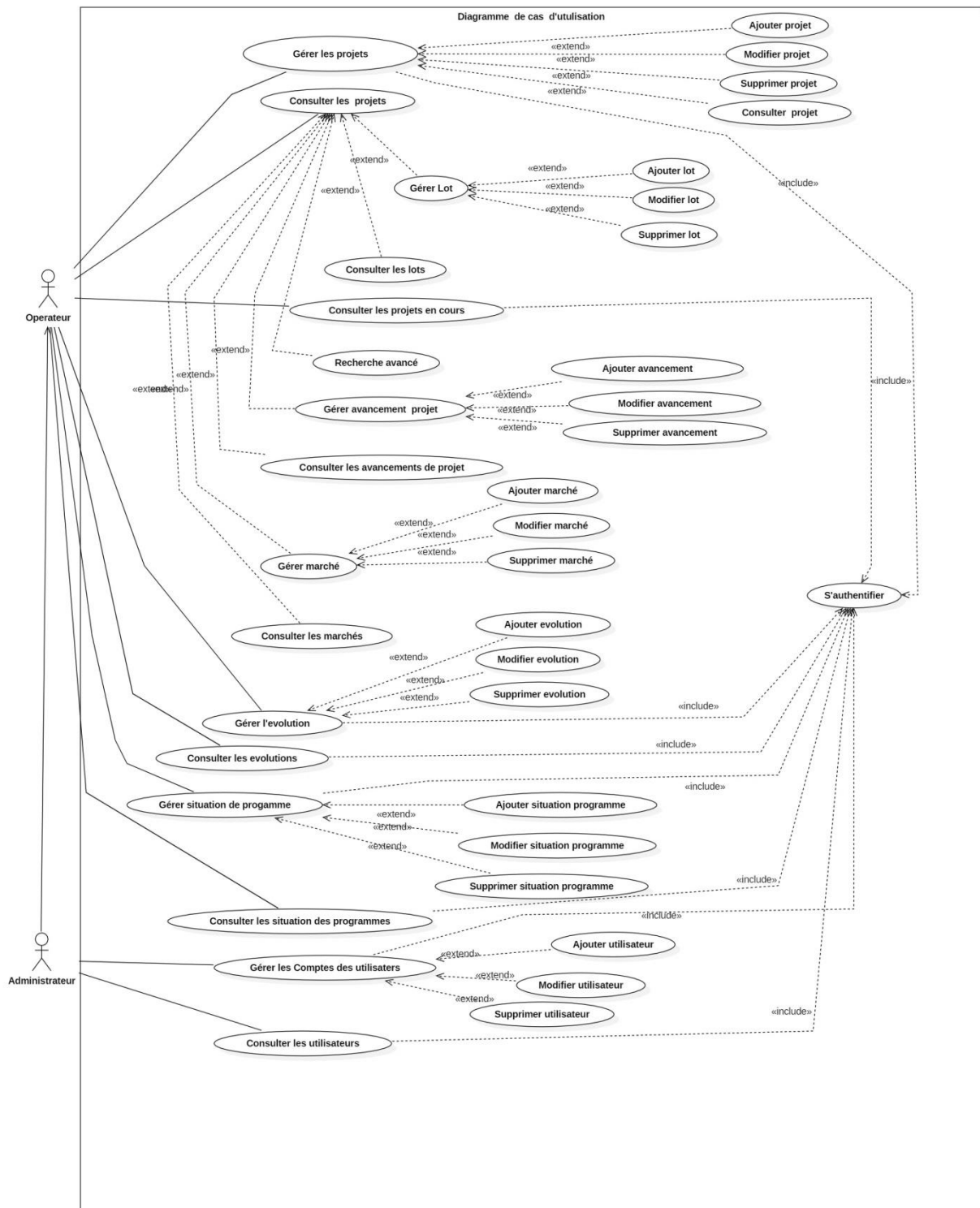


Figure 3.3 Diagramme de cas d'utilisation

3.4.3.2. Diagramme de cas d'utilisation des microservices

- Gestion d'utilisateurs

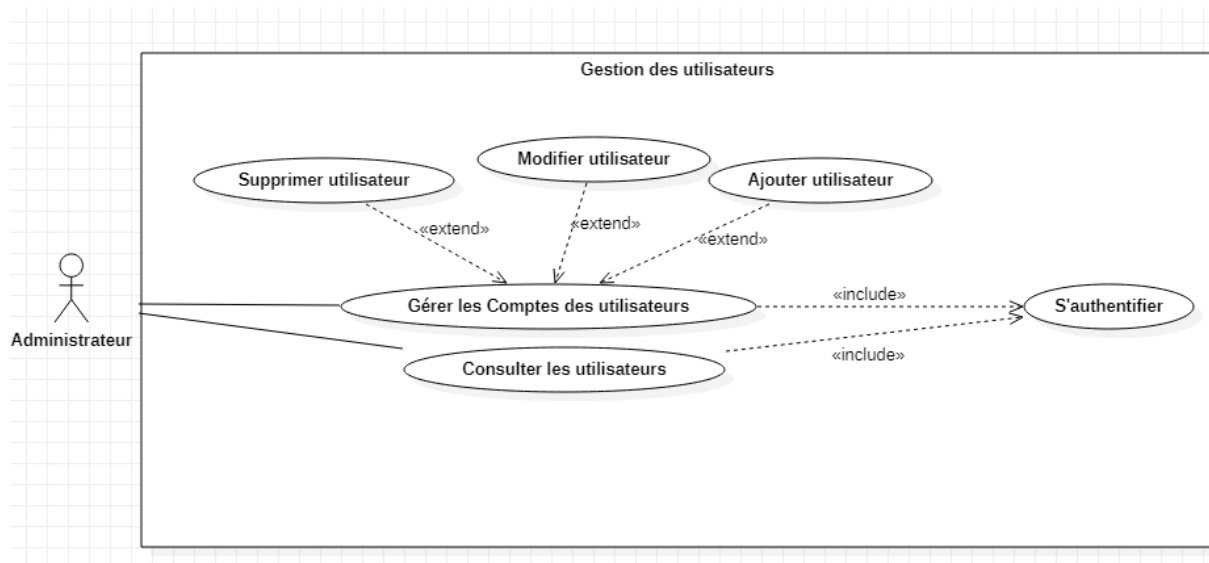


Figure 3.4 Diagramme de cas d'utilisation pour le microservice gestion des utilisateurs

- Gestion des statistiques

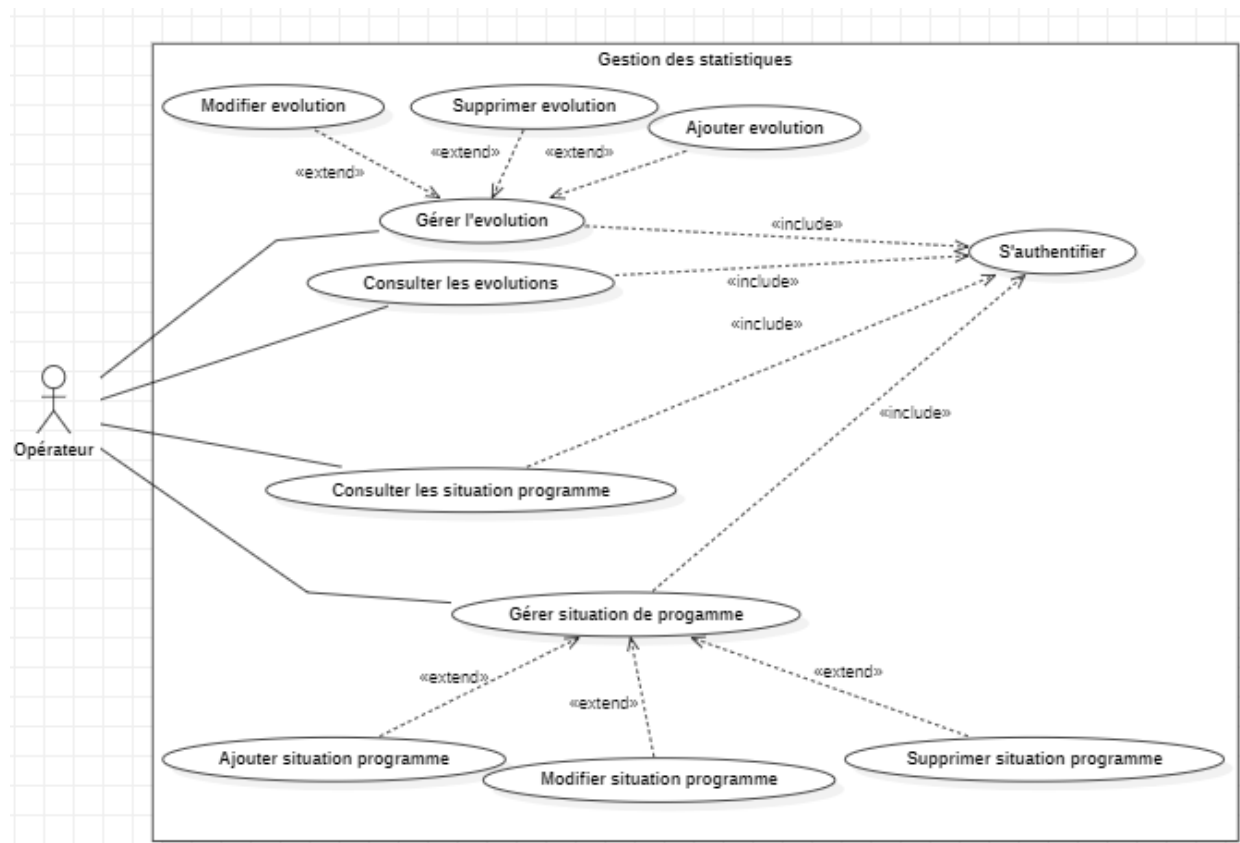


Figure 3.5 Diagramme de cas d'utilisation pour le microservice gestion des statistiques

- Gestion des projets

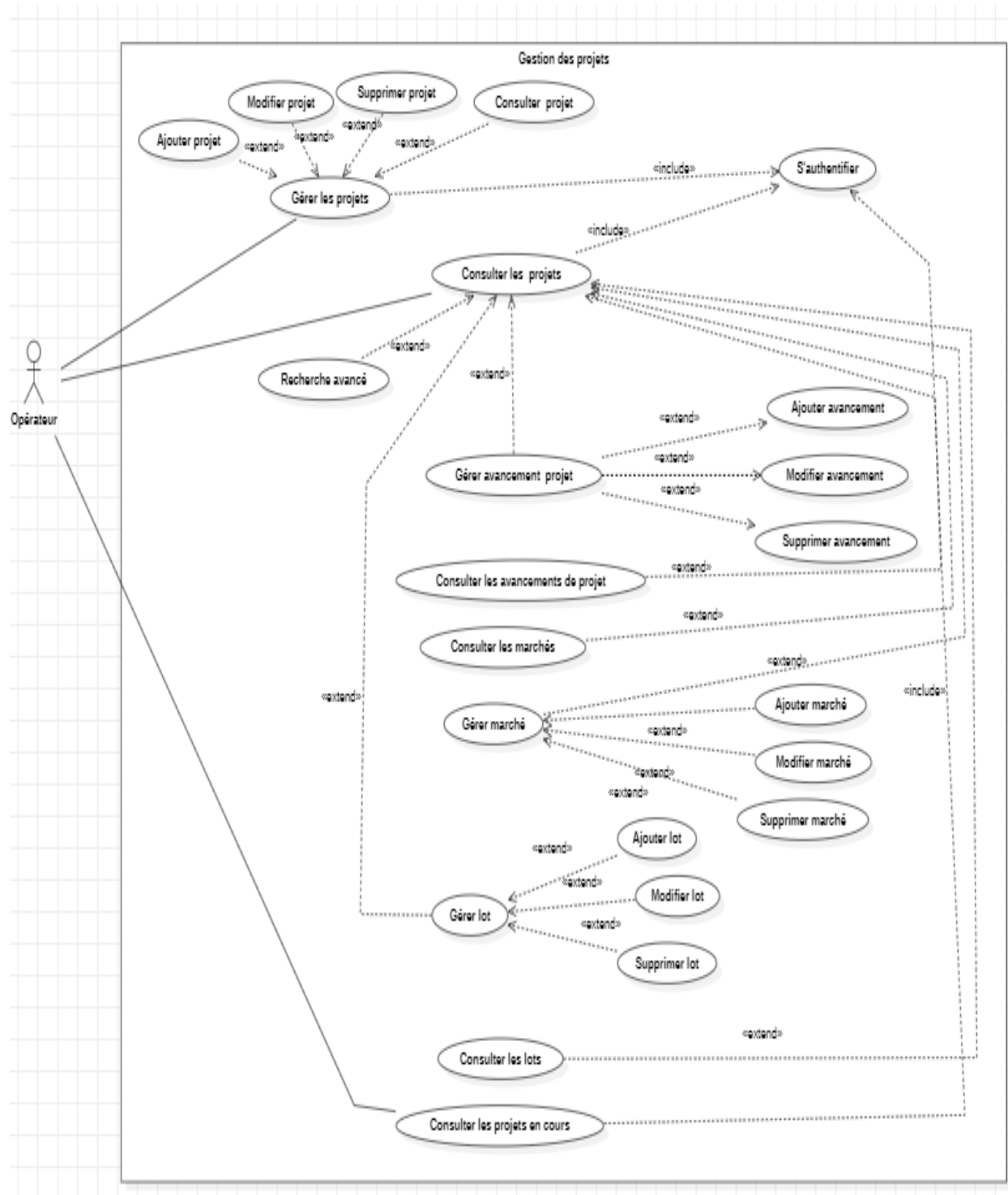


Figure 3.6 Diagramme de cas d'utilisation pour le microservice gestion des projets

3.5. Conclusion

Dans ce chapitre, nous avons présenté le diagramme de cas d'utilisation. Cette phase nous aide à préciser les objectifs de notre système et aussi à préparer la phase de conception qu'est l'objet du chapitre suivant qui consiste à la conception du système en suivant toujours le processus UP.

Chapitre 04 :

Conception du système

4. Conception du système

4.1. Introduction

Dans ce chapitre, nous présentons d'abord les diagrammes de séquence et les diagrammes d'états-transitions. Ensuite, nous décrivons le dictionnaire des données et présentons les diagrammes de classe pour chaque microservice. Enfin, nous élaborons le modèle relationnel.

4.2. Dictionnaire de données

Le dictionnaire de données est le résultat de l'étape de collecte des données « recueil d'informations ». En effet, lors de la phase de conception, les données collectées et spécifiées sont écrites dans le dictionnaire. Ce dictionnaire est un outil important, car il est une référence à toutes les études effectuées [16].

Un dictionnaire de données est un tableau qui inclut toutes les données IS, pour chaque donnée spécifique, il est nécessaire de spécifier :

- Nom symbolique (Attribut).
- Signification ou description.
- Type de retour.

Nom des classes	Attributs	Signification ou description	Type de retour
projet	numOP secteur soussecteur chapitre article gestionnaire typeProgm codeOp dateInscr localisation myfile typeBudget apAct codeSaisie dateSaisie intitule	Numéro opération Secteur Sous-secteur Chapitre Article Gestionnaire Type programme Code opération Date inscription Localisation Attachement Type budget Autorisation de programme actuelle Code Saisie Date Saisie Intitule	Entier Chaine de caractère Chaine de caractère Chaine de caractère Chaine de caractère Chaine de caractère Type Programme Chaine de caractère Date Chaine de caractère Fichier Type Budget Entier Entier Date Chaine de caractère

Chapitre 04 : conception du système

Utilisateur	userName Nom Prenom dateDeNaissance lieuDeNaissance motpasse numService numUser	User Name Nom Prénom Date De Naissance Lieu De Naissance Mot passe Numéro de Service Numéro de User	Chaîne de caractère Chaîne de caractère Chaîne de caractère Date Chaîne de caractère Chaîne de caractère Entier Entier
Entite	numEnt codeEnt libelleEnt libelleArEnt	Numéro d'Entité Code d'Entité Libelle d'Entité Libelle Arabe d'Entité	Entier Chaîne de caractère Chaîne de caractère Chaîne de caractère
Role	Id Libelle	Identifiant Libelle	Entier Chaîne de caractère
Situation programme	nbpUniv etat nbpPlace nbpHeberg nbpBiblioth nbplab nbpRest npAudito nbpRectorat nbpLogEnseign	Nombre des projets Universitaires État Nombre des projets place pédagogiques Nombre des projets Hébergés Nombre des projets de Bibliothèques Nombre des projets laboratoires Nombre des projets Restaurants Nombre des projets Auditoriums Nombre des projets des Rectorats Nombre des projets de Logements Enseignements	Entier Etat Entier Entier Entier Entier Entier Entier Entier Entier
Evolution Ens-sup	Annee nbUniv nbInst nbdep placePedag placeHeberg Biblio Restau logEnseign nbEtud nbEtudetr nbEtudHeb nbEnseign nbEmpAdm nbEtudAdm nbEtudAjo nbEtudDip	Année Nombre des Universités Nombre des Institues Nombre des départements Nombre des Places Pédagogiques Nombre des places Hébergé Nombre des Bibliothèques Nombre des Résteaux Nombre des logements d'Enseignments Nombre des Etudiants Nombre des Etudiants étrangères Nombre des Etudiants Hébergés Nombre des Enseignants Nombre des Employé Administratifs Nombre des Etudiants Admis Nombre des Etudiants Ajournées Nombre des Etudiants Diplômés	Entier Entier Entier Entier Entier Entier Entier Entier Entier Entier Entier Entier Entier Entier Entier Entier Entier

Chapitre 04 : conception du système

Marché	numMar typeMar montantMar dateOds delaisExec retenuGar myfile	Numéro de Marché Type de Marcher Montant de Marché Date d'ordre de services Délais Exception Retenu Garanté File	Entier Type marché Entier Date Chaine de caractère Entier File
Avancement projet	numAvan numOp etatOp desc consomation depenche tauxAvan dateAvan datePrev impact observation codeSaisie dateSaisie myfile	Numéro d'Avancement Numéro d'opération Etat d'Opérations Description Consommation Dépense Le taux d'Avancement La date d'Avancement La date Prévu Impact Observation Code de Saisie Date de Saisie File	Entier Entier état Chaine de caractère Entier Entier Entier date date Chaine de caractère Chaine de caractère Entier date file
Lots	idLot libelleLot montantLot	Identifiant de lot Libelle de Lot Montant de Lot	Entier Chaine de caractère Entier
Entreprise	numEntre libelleEntr gerant num Tel email degreQual siegeSoc numRegCom numFISC	Numéro d'Entreprise Libelle de l'Entreprise Gérant Numéro de Téléphone Email Degré de Qualification Siège Sociale Numéro de Registre de Commerce Identification fiscale	Entier Chaine de caractère Chaine de caractère Entier Chaine de caractère Entier Chaine de caractère Entier Entier
Structure de cout	pEtude pBati pTrav pEquip pMatér pFormat pServExt pStock autres	Prix Etude ou engineering Prix Batiment et génie-civile lie Prix Traveaux publics Prix Equipements et machines Prix Matériel de transport et de manutention Prix Formation Prix de prestation de Services Extérieur Prix Stock outils Autres	Entier Entier Entier Entier Entier Entier Entier Entier Entier

Chapitre 04 : conception du système

	pInfraEnv pTerrain pFrais	Prix Infrastructures Environnantes Prix Terrain Prix Frais Anep	Entier Entier Entier
--	---------------------------------	---	----------------------------

Tableau 4.1 Dictionnaire de données

4.3. Diagramme de classes

Les diagrammes de classes sont l'un des types de diagrammes UML les plus utiles car ils décrivent clairement la structure d'un système particulier en modélisant ses classes, ses attributs, ses processus et ses relations entre ses objets [15].

Pour ce travail, chaque microservice possède un diagramme de classes. Les figures suivantes : 4.1, 4.2, 4.3 et 4.4 représentent les diagrammes de classes de système complet ainsi des différents microservices du système.

Chapitre 04 : conception du système

4.3.1. Diagramme de classes du système complet

Le diagramme de classes de la figure 4.1 ci-dessous décrit l'ensemble des classes du système complet sans prendre en compte le découpage en microservice.

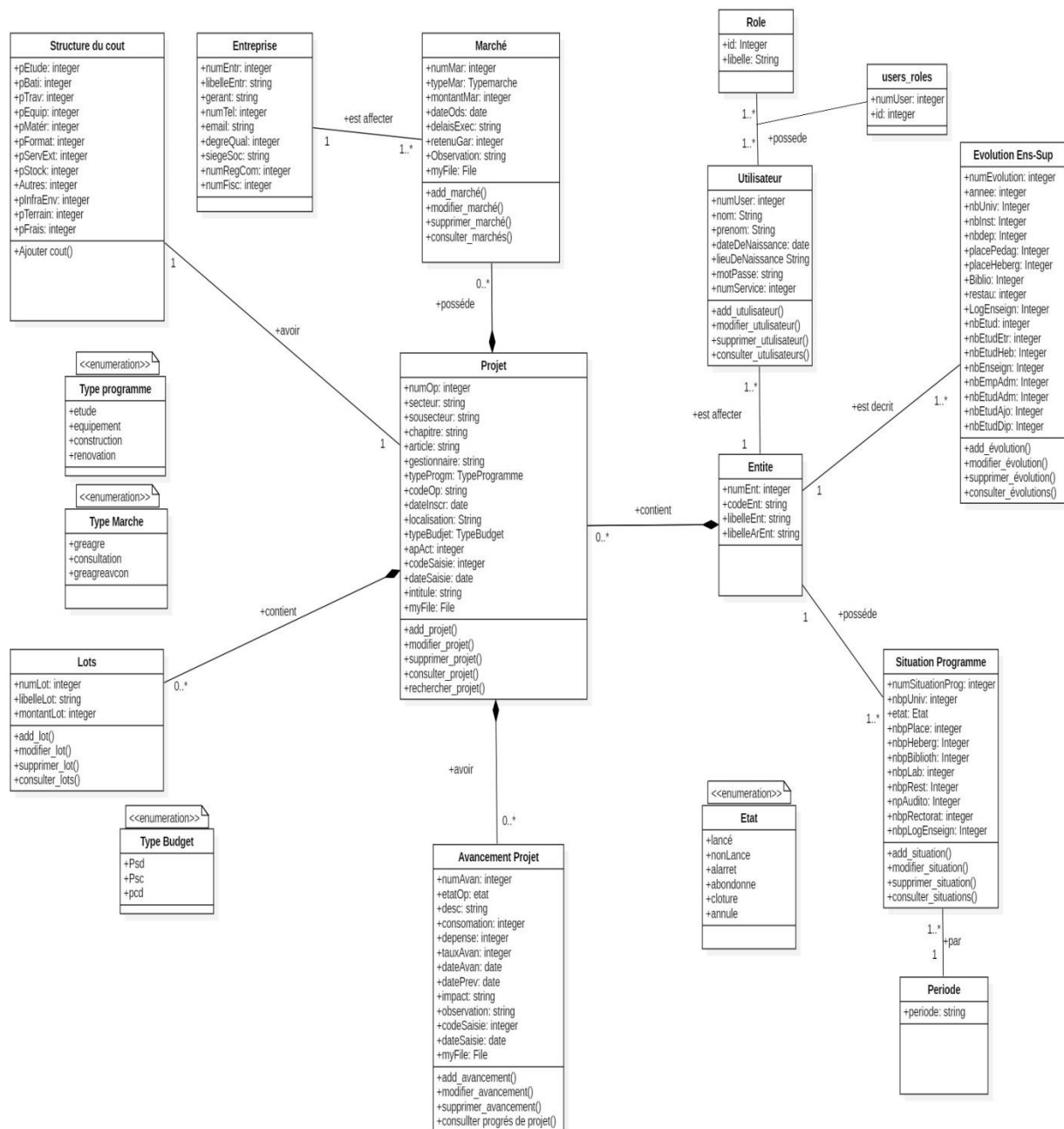


Figure 4.1 Diagramme de classes de système complet

Chapitre 04 : conception du système

La figure 4.1 montre le diagramme de classes du système si une application monolithique sera développée. Puisque notre architecture est basée sur des microservices, nous devons faire une division logique et simple de ce diagramme. Chaque microservice doit avoir sa propre base de données donc auront chacun un diagramme de classe correspondant.

4.3.2. Diagramme de classes de microservices

- **Gestion d'utilisateurs**

Le diagramme de classes de la figure 4.2 ci-dessous décrit l'ensemble des classes participantes aux fonctionnalités du microservice gestion d'utilisateurs.

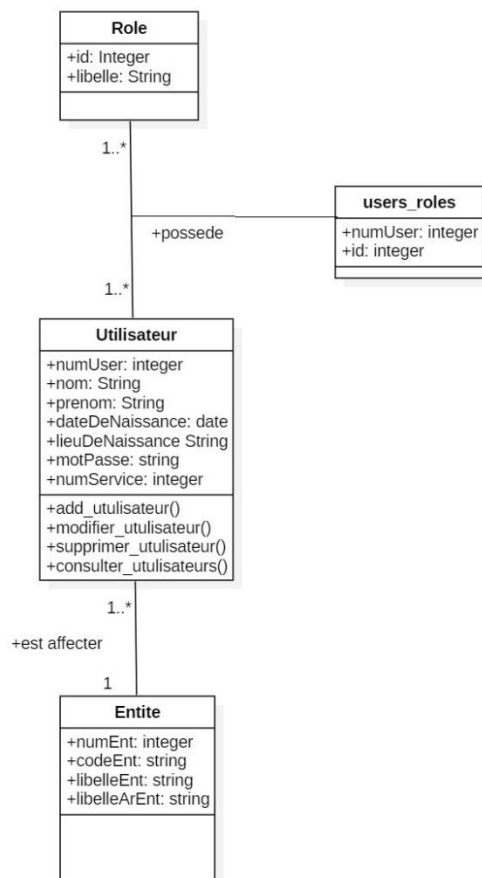


Figure 4.2 Diagramme de classes de microservice gestion des utilisateurs

• Gestion des statistiques

Le diagramme de classes de la figure 4.3 ci-dessous décrit l'ensemble des classes participantes aux fonctionnalités du microservice gestion des statistiques.

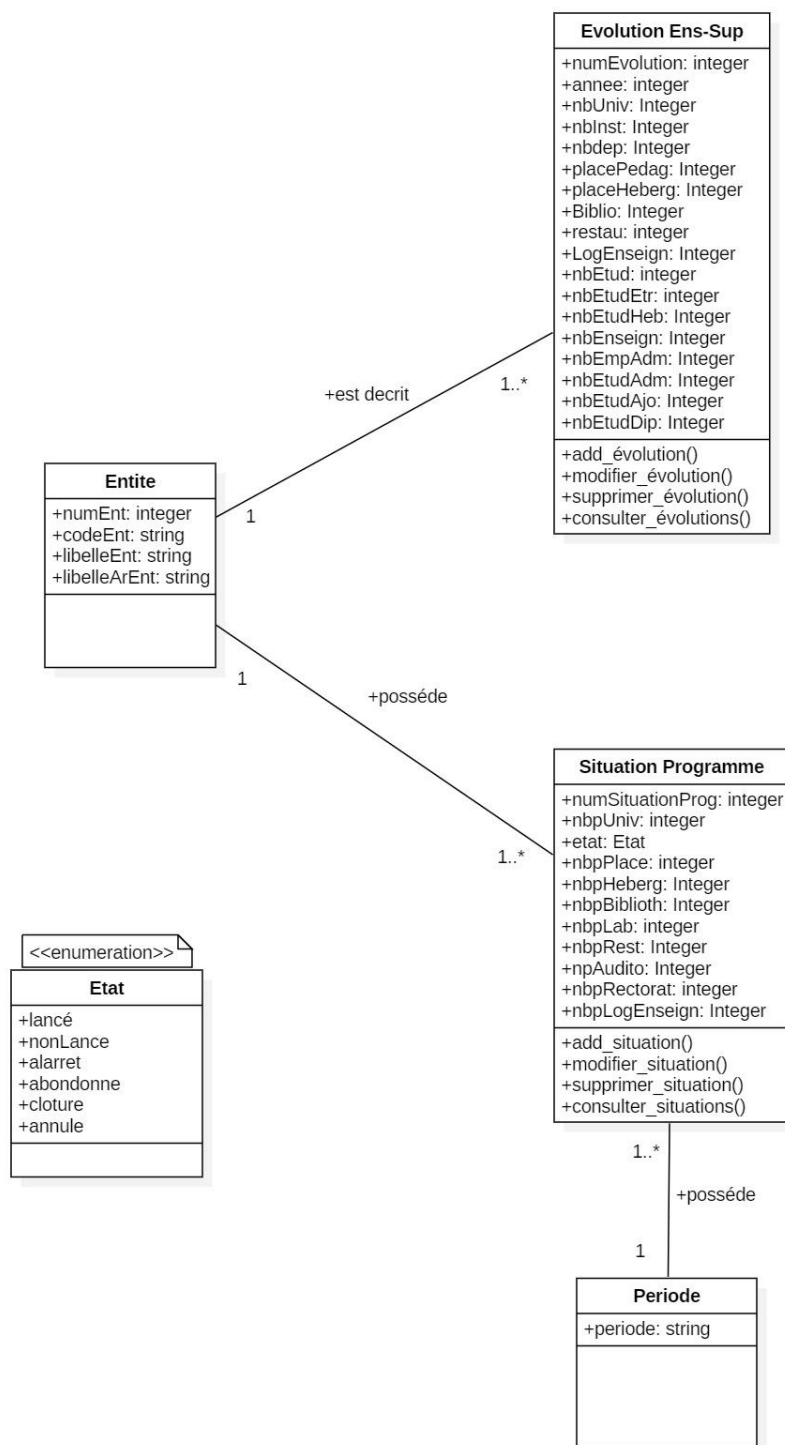


Figure 4.3 Diagramme de classes de microservice gestion des statistiques

➤ Gestion des projets

Le diagramme de classes de la figure 4.4 ci-dessous décrit l'ensemble des classes participantes aux fonctionnalités du microservice gestion des projets.

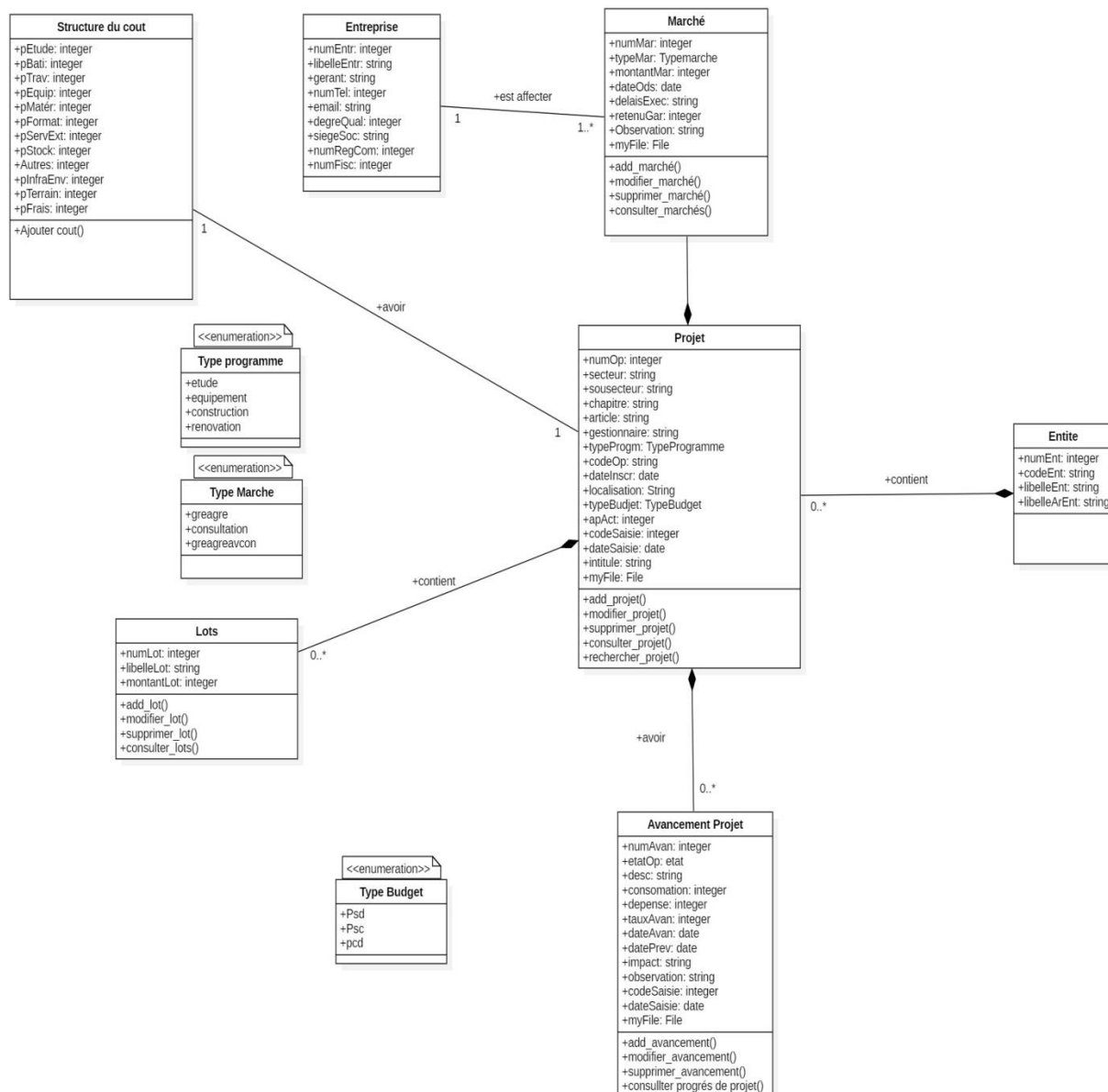


Figure 4.4 Diagramme de classes de microservice gestion des projets

Chapitre 04 : conception du système

4. 4. Diagramme de séquence

Diagramme de séquence montre l'ordre vertical des messages envoyés entre les objets dans l'interaction. Ils représentent l'échange de messages entre éléments, dans le cadre d'un certain fonctionnement du système. Les diagrammes de séquence sont d'abord utilisés pour développer des scénarios d'utilisation du système dans l'analyse. Ce dernier permet de concevoir les méthodes des classes [12].

Plusieurs scénarios seront décrits. Il y'a deux scénarios possibles : un scénario normal et un scénario d'erreur. On parle de scénario normal lorsque tout se passe bien. Un scénario d'erreur est lorsqu'on rencontre une erreur lors de l'exécution d'une tâche ou d'une fonctionnalité.

4.4.1. Gestion des projets

- Ajouter projet

a) Description textuelle

Pour la fonctionnalité ajouter projet, plusieurs scénarios peuvent être observés : scénario normale et scénario d'erreur.

Cas d'utilisation :	Ajouter projet .
Acteur :	Opérateur.
Résumé :	Ce cas d'utilisation permet à l'acteur d'ajouter un projet.
Pré condition :	L'opérateur consulte les projets.
Post condition :	Le projet est ajouté.
Description du scénario nominal :	1. L'opérateur choisit projet. 2. Le système affiche le formulaire d'ajoute. 3. L'opérateur saisit les informations du projet et enregistrer. 4. Le système lit les paramètres et vérifie les paramètres. 5. Le système affiche un message d'information.
Description du scénario alternatif :	3. a- L'opérateur n'est pas rempli des champs obligatoires, le système indique au l'opérateur que le formulaire contient des champs vide ou incorrect et lui proposer de remplir. 3. b- verrez à l'étape 3.

Tableau 4.2 Description de cas d'utilisation ajouter projet

b) Diagramme de séquence

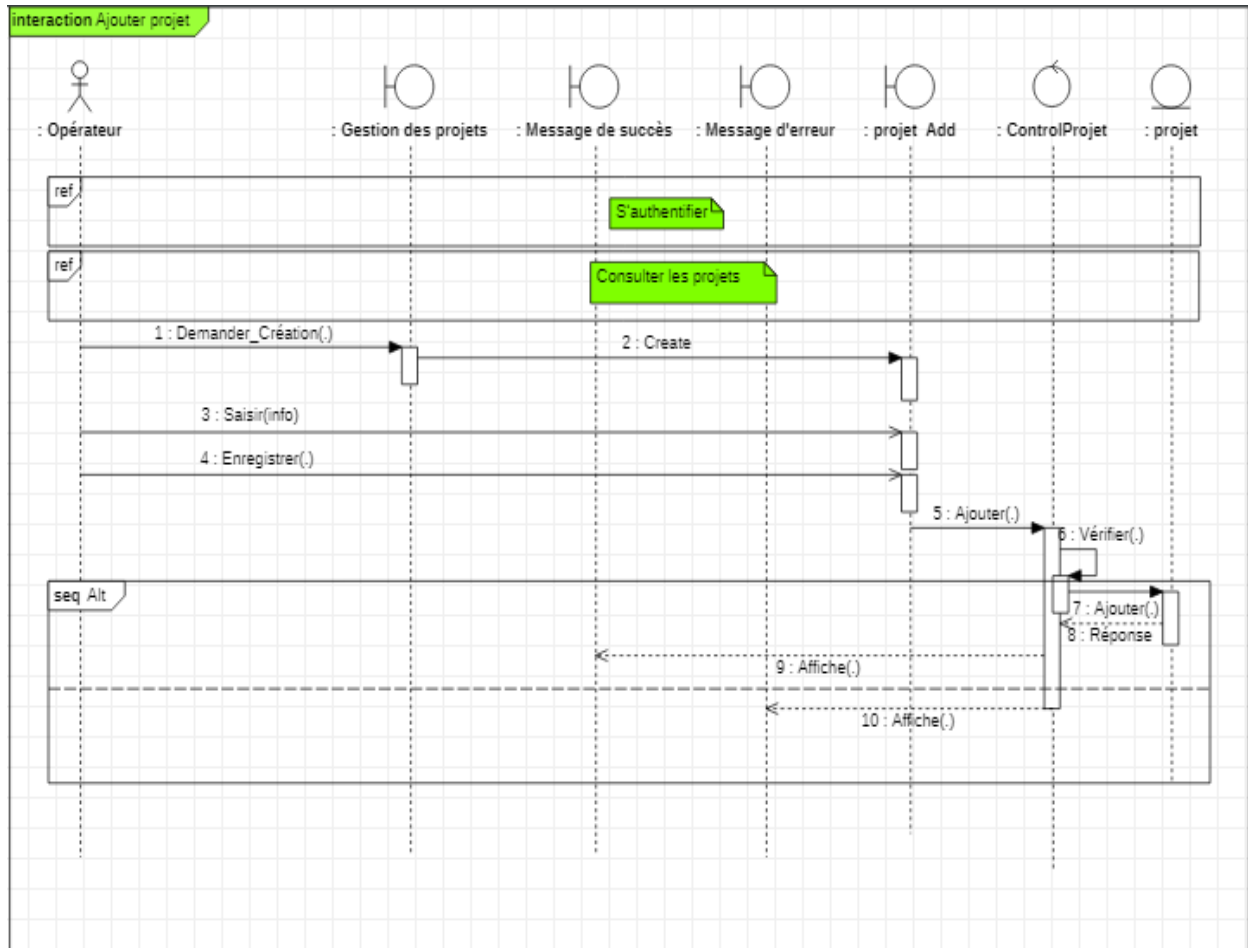


Figure 4.5 Diagramme de séquence ajouter projet.

Consulter projet Description textuelle

Cas d'utilisation :	consulter projet .
Acteur :	Opérateur.
Résumé :	Ce cas d'utilisation permet à l'acteur de modifier les informations d'un projet.
Précondition :	L'opérateur consulte les projets.
Post condition :	Les informations du projet sont modifiées.
Description du scénario nominal :	1. L'opérateur clique sur un projet qui il veut voir ses détails. 2. le système afficher un page contient tous les informations de ce projet.
Description du scenario alternatif :	//.

Tableau 4.3 Description de cas d'utilisation consulter projet

b) Diagramme de séquence

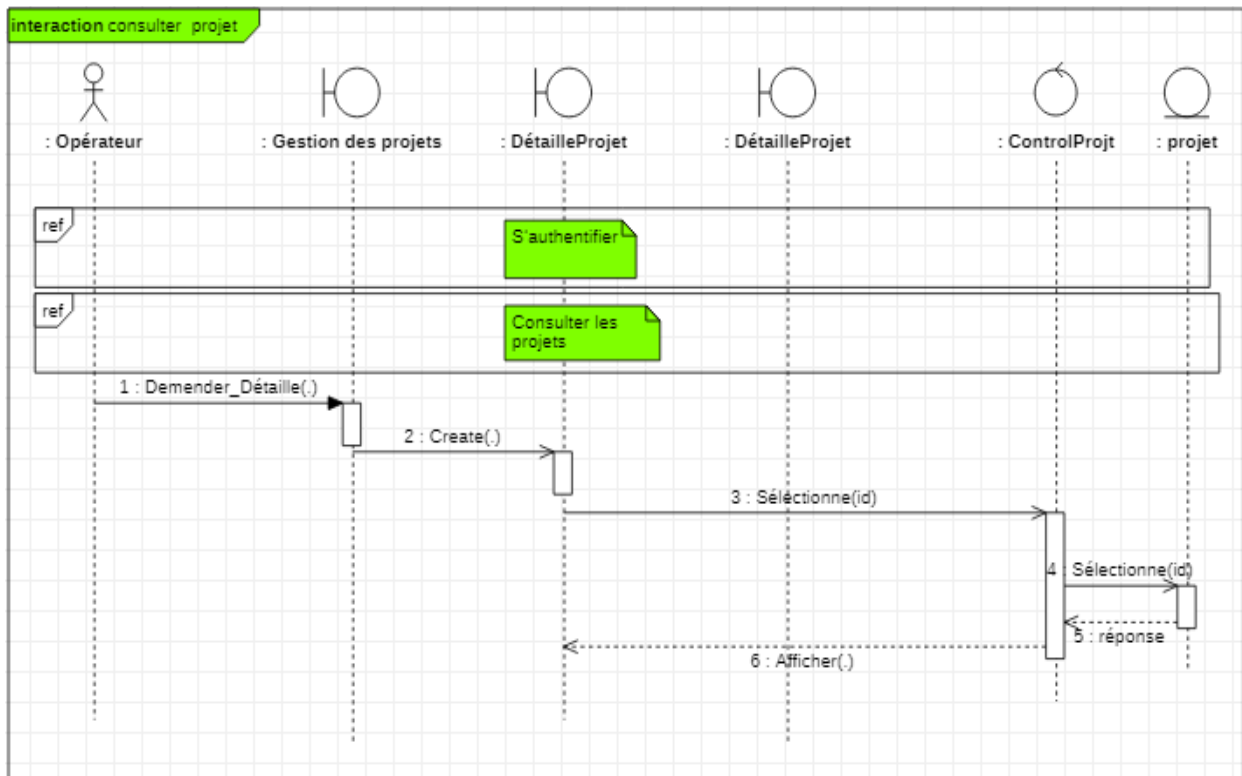


Figure 4.6 Diagramme de séquence consulter projet

- **Modifier projet**

- a) **Description textuelle**

Cas d'utilisation :	Modifier projet .
Acteur :	Opérateur.
Résume :	Ce cas d'utilisation permet à l'acteur de modifier les informations d'un projet.
Précondition :	L'opérateur consulte les projets.
Post condition :	Les informations du projet sont modifiées.
Description du scénario nominal :	1. L'opérateur choisit un projet qu'il veut modifier. 2. Le système affiche les détails de projet. 3. L'opérateur demande la modification. 4. Le système affiche un formulaire pour faire les modifications. 5. L'opérateur saisit les modifications et enregistrer. 6. Le système vérifie les paramètres et effectue la modification de projet et affiche un message d'information.
Description du scénario alternatif :	7. a -Le système affiche un message d'erreur.

Tableau 4.4 Description de cas d'utilisation modifier projet

a) Diagramme de séquence

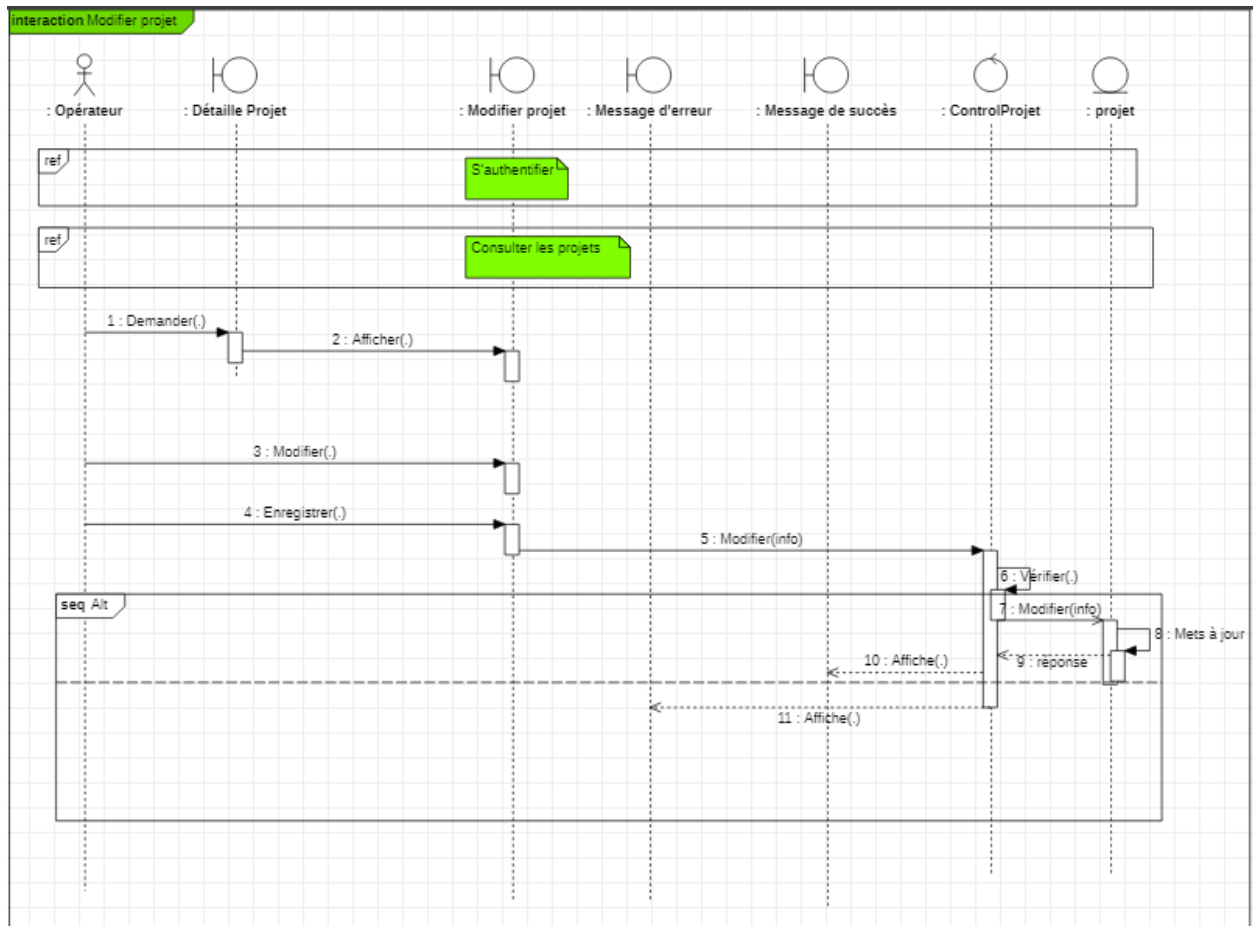


Figure 4.7 Diagramme de séquence modifier projet

- **Supprimer projet**

a) Description textuelle

Cas d'utilisation :	Supprimer projet .
Acteur :	Opérateur.
Résumé :	Ce cas d'utilisation permet à l'acteur de supprimer un projet.
Précondition :	L'opérateur consulte les projets.
Post condition :	Le projet est supprimé.
Description du scénario nominal :	1. L'opérateur choisit un projet et demande sa suppression. 2. Le système affiche un message de confirmation. 3. L'opérateur confirme la suppression de projet. 4. Le système effectue la suppression de projet et affiche un message d'information.
Description du scénario alternatif :	3. a- L'opérateur annuler la suppression de projet. .

Tableau 4.5 Description de cas d'utilisation supprimer projet

b) Diagramme de séquence

Chapitre 04 : conception du système

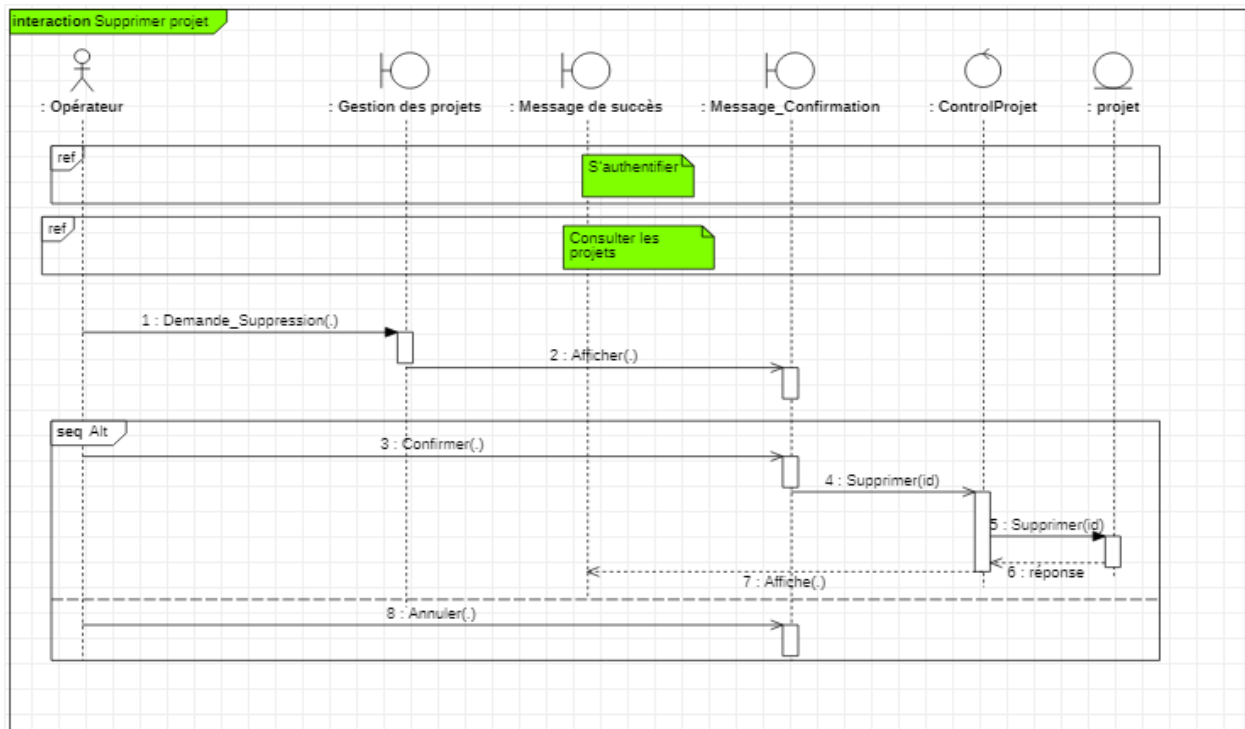


Figure 4.8 Diagramme de séquence supprimer projet

Consulter les projets

a) Description textuelle

Cas d'utilisation :	Consulter les projets.
Acteur :	Opérateur.
Résumé :	ce cas d'utilisation permet à l'acteur de consulter la liste des projets.
Précondition :	L'opérateur est authentifié.
Post condition :	La liste des projets est affichée.
Description du scénario nominal :	1. L'utilisateur demande la page de gestion des projets. 2. Le système affiche la liste des projets enregistrés.
Description du scénario alternatif :	2. a- Le système affiche liste vide.

Tableau 4.6 Description de cas d'utilisation Consulter la liste des projets

b) Diagramme de séquence

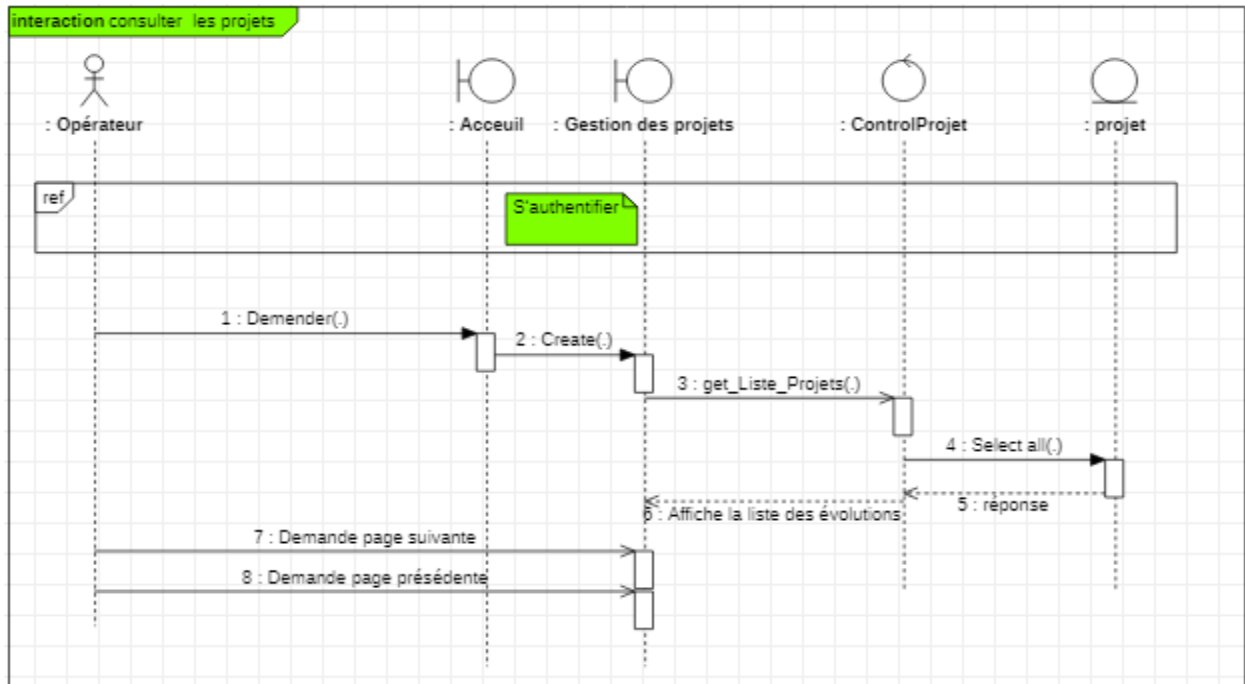


Figure 4.9 Diagramme de séquence consulter les projets

4.5. Diagramme d'état transition

Représente le cycle de vie commun aux objets d'une même classe. Ce diagramme complète la connaissance des classes en analyse et en conception [10].

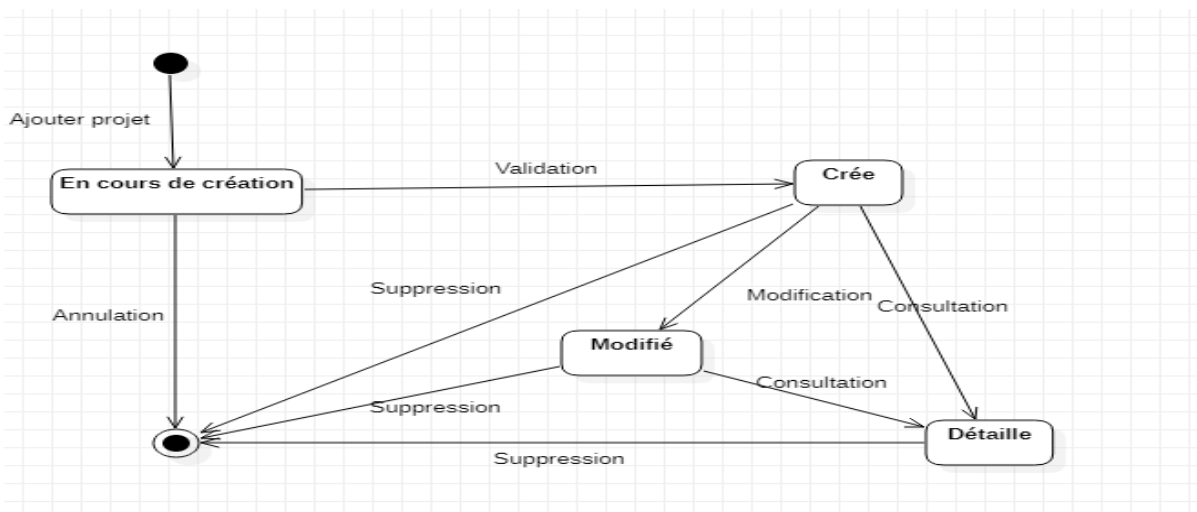


Figure 4.10 Diagramme d'état transition de l'objet projet

➤ La suite des diagrammes est dans « **Annexe 3** »

4.6. Le passage du diagramme de classe au modèle relationnel

4.6.1. Les règles de passages

La réalisation du modèle relationnel peut être faite à partir de la description conceptuelle. Ce passage de modèle doit respecter un certain nombre de règle, nous avons utilisé ces règles qui sont les plus simples et plus opérationnelles [5].

Règle1 : Toute classe devient une relation, les attributs de la classe deviennent des attributs de la relation, si la classe possède un identifiant il devient la clé primaire de la relation.

Règle2 : pour représenter une association 1 vers 1 (1..1) entre deux relations, la clé primaire de l'une des relations doit figurer comme clé étrangère dans l'autre relation.

Règle3 : pour représenter une association 1 vers plusieurs (1..*), on procède comme une association 1 vers 1, excepté que c'est forcément la relation du coté plusieurs qui reçoit comme clé étrangère la clé primaire de la relation du coté 1.

Règle4 : pour représenter une association plusieurs vers plusieurs (*..*), il faut introduire une nouvelle relation dont les attributs sont les clés primaires des relations en association, et dont la clé primaire est la concaténation de ces deux attributs. Si l'association possède des attributs, ils deviennent des attributs de la relation correspondante.

Règle5 : cas d'héritage, transformer chaque sous classe en une relation, la clé primaire de la super classe devient clé primaire de chaque sous classe.

Règle6 : cas de composition, la clé primaire de la classe composée devient clé étrangère de la classe composant.

Règle7 : cas d'agrégation, le même principe que la **Règle 3**.

Chapitre 04 : conception du système

4.6.2. Les tables de la base de données

En se basant sur les règles ci-dessus, nous avons converti les classes entités et leurs associations, à des tables dans la base donnée. Les tables générées sont :

a) microservice projet :

Projet (numOp , secteur , soussecteur , chapitre , article , typeProgm , typePudjet , codeOp , Intitule , apAct , dateInscr , entite , gestionnaire , myfile , # numEnt),

Avancement Projet (numAvan , etatOp , desc , consommation , depense , tauxAvan , dateavan , datePrev , impact , observation , myfile , # numOp),

Marché (numMar , typeMar , descMar , dateOds , montantMar , delaisExec , retenuGar , #entreprise , observation , Myfile ,#numEntr ,#numOp),

Entreprise (numEntr , libelleEntr , gerant , numTel , email , degreQual ,siegesoc , numRegCom , numFisc),

Lots (numLot , libelleLot , montantLot , # numOp)

b) microservice statistiques :

Evolution Ens-Sup (numEvolution , #numEnt , annee , nbUniv , nbInst , nbdep , placePedag , placeHeberg , biblio , restau , logEnseign , nbEtud , nbEtudEtr , nbEtudHeb , nbEnsregn , nbEmpAdm , nbEtudAjo , nbEtudDip ,# numEnt),

Situation Programme (numSituationProg , #numEntite , periode , etat , nbpUniv , nbpPlace , nbpHeberg , nbpBiblioth , nbpRest , nbpLab , nbpAudito , nbpRectorat , nbpLogEnseign ,# numEnt),

c) microservice user :

Entite (numEnt , codeEnt , libelleEnt , libelleArEnt),

Utilisateur (numUser , nom , prenom , dateDeNaissance , LieuDeNaissance , motPasse ,

numService , # numEnt),

Rôle (id , libelle),

Users_roles (#numUser , #id),

4.7. Conclusion

Dans ce chapitre, nous avons présenté la conception détaillée de notre application. Plus précisément, nous avons décrit d'abord les différents diagrammes suivants : les diagrammes de séquence et les diagrammes d'états-transitions. Ensuite, nous avons présenté le dictionnaire des données et les diagrammes de classe pour chaque microservice. Enfin, nous avons élaboré le modèle relationnel. Dans le chapitre suivant, nous allons présenter les outils de développement ainsi que la réalisation de notre système.

Chapitre 05 : Implémentation du système

5. Implémentation du système

5.1. Introduction

Dans ce chapitre, nous présentons d'abord les outils de développement utilisés pour construire notre système. Ensuite, nous décrivons l'architecture de notre implémentation. Enfin, nous donnons quelques captures d'écran des interfaces du système.

5.2. Présentation des outils et de langages utilisés

Nous avons utilisé plusieurs outils logiciels pour une meilleure réalisation du projet. Ses différents outils nous ont permis de faire l'ensemble des diagrammes présenté dans ce document et d'implémenter les différents programmes informatiques.

Spring Boot:



Figure 5.1 Logo du framework spring boot

C'est un Framework qui permet de démarrer rapidement le développement d'applications [14].

Spring Cloud :



Figure 5.2 Logo de l'outil spring cloud

Spring est une plateforme construit pour le développement d'applications web en langage Java. Il a été introduit en 2004. En 2006, des sous-projets (sub-project) sont apparus. Chaque sous-projet se concentre sur un domaine différent. Jusqu'à présent, vous pouvez voir les sous-projets listés comme l'illustration suivante. [15].

Java :



Figure 5.3 Logo du langage de programmation java

La technologie Java définit à la fois un langage de programmation orienté objet et une plateforme informatique [16].

Maven :



Figure 5.4 Logo de Maven

Maven est un outil de gestion et d'automatisation de production des projets logiciels Java en général et Java EE en particulier. Il est utilisé pour automatiser l'intégration continue lors d'un développement de logiciels [17].

Postman :



Figure 5.5 Logo de Postman

Postman est une plate-forme d'API pour la création et l'utilisation d'API. Postman simplifie chaque étape du cycle de vie des API et rationalise la collaboration afin de créer de meilleures API, plus rapidement [18].

Angular :



Figure 5.6 Logo de Framework Angular

Angular est un Framework côté client, open source, basé sur Type Script, et codirigé par l'équipe du projet Angular à Google et par une communauté de particuliers et de sociétés. Il permet la création d'applications Web et plus particulièrement de ce qu'on appelle des SPA

(*Single Page Applications*) des applications web accessibles via une page web unique qui permet de fluidifier l'expérience utilisateur et d'éviter les chargements de pages à chaque nouvelle action. Le Framework est basé sur une architecture du type MVC et permet donc de séparer les données, le visuel et les actions pour une meilleure gestion des responsabilités [19].

Star Uml:



Figure 5.7 : Logo du logiciel star uml.

Le logiciel StarUML est un logiciel open-source cédé par son ancien éditeur sous licence GNU GPL, dédié aux plateformes Windows, il est développé en Delphi.

Ses principaux avantages sont sa simplicité d'installation et de prise en main, et la possibilité de générer le squelette des classes en langages Java, C++, C#, ActionScript3.0... De plus, le logiciel a été conçu en prévoyant l'ajout de plugin supplémentaires afin de pouvoir être adapté simplement aux besoins évolutifs des ses utilisateurs [20].



Visual Paradigm for UML

Figure 5.8 Logo du editeur visual studio

Version: 6.3 Enterprise Édition Visual paradigm for UML Visual Paradigm est un logiciel permettant aux programmeurs de mettre en place des diagrammes UML [21].

Apache tomcat :



Figure 5.9 Logo du serveur Web apache Tomcat

Apache Tomcat est un serveur Web open source développé par l'Apache Software Foundation (ASF). Il est l'un des nombreux produits open source liés à Apache utilisés par les professionnels de l'informatique pour diverses tâches et objectifs [22].

Node.js :



Figure 5.10 Logo de Framework Node.js

C'est un environnement d'exécution single-thread, open source et multi plate-forme permettant de créer des applications rapides et évolutives côté serveur et en réseau [23].

Npm :



Figure 5. 11 Logo du gestionnaire des packages NPM

NPM est un gestionnaire de packages pour le langage de programmation JavaScript. Il s'agit du gestionnaire de packages par défaut pour l'environnement d'exécution JavaScript Node.js. Il se compose d'un client de ligne de commande, également appelé *npm*, et d'une base de données en ligne de packages publics et privés payants, appelée registre *npm* permettant à des milliers de développeurs d'accélérer leurs développements [24].

PostgreSQL :



Figure 5.12 Logo du système de la base de données

PostgreSQL est un SGBD relationnelle objet open source avec plus de 35 ans de développement actif qui lui a valu une solide réputation de fiabilité, de robustesse des fonctionnalités et de performance [25].

5.3. Description du système

Pour implémenter notre système, nous avons développé une application Web. L'architecture générale de cette application est montrée dans la figure 5.1 et contient les trois parties suivantes : la partie front-end, la partie back-end et les bases de données. La partie front-end est développée avec le framework Angular. La partie back-end est développée avec

Chapitre 05 : Implémentation du système

l'architecture Microservices en utilisant Spring boot, Spring cloud et Spring Security. Pour la gestion des bases de données, nous avons utilisé PostgreSQL.

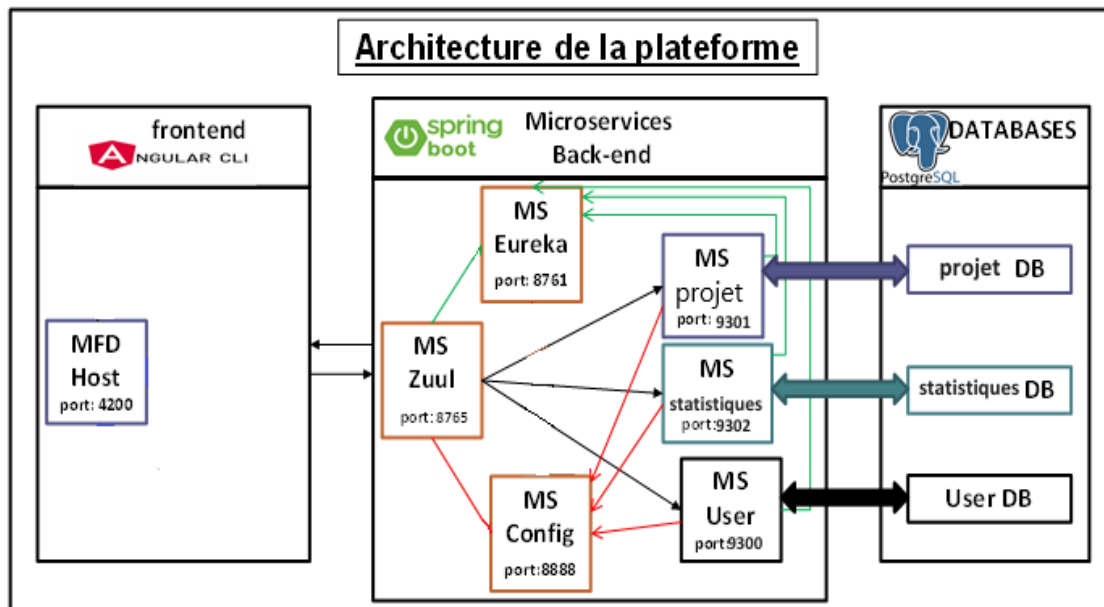


Figure 5.13 Architecture de l'application

Nous avons 6 microservices, nous en avons implémenté trois (Projet, Statistiques, Utilisateur) Pour résoudre notre problématique, et les trois autres (Config, Eureka, Zuul) pour la configuration et la communication entre les microservices.

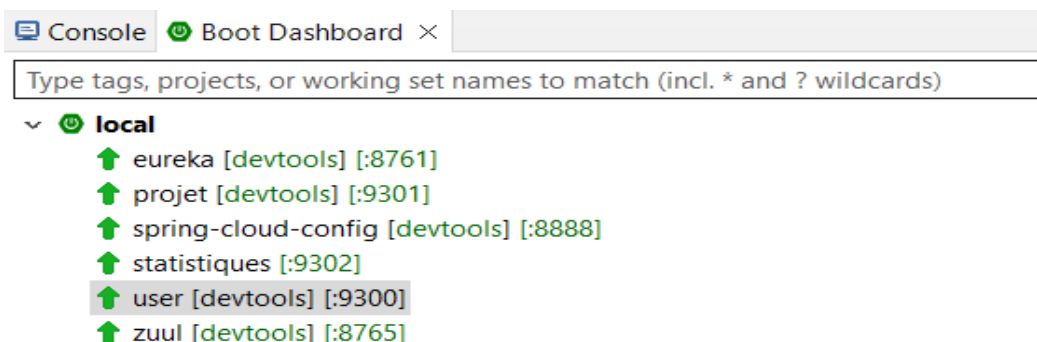


Figure 5.14 Les microservices

5.4. Les interfaces

Dans cette partie, nous présentons les interfaces graphiques de notre application.

5.4.1. L'interface de connexion

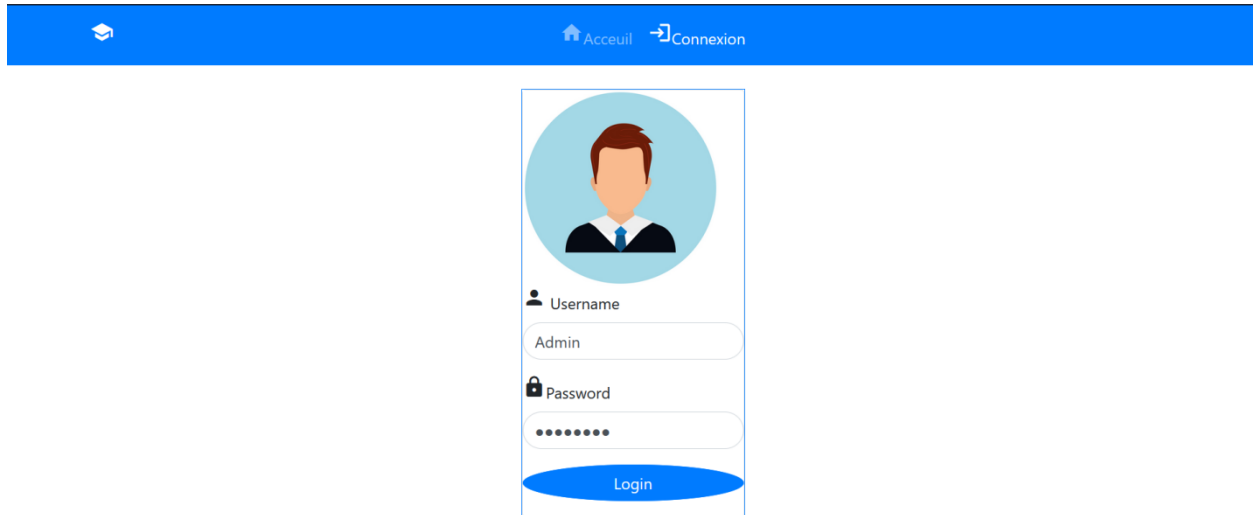


Figure 5.15 shows the login interface. It features a blue header bar with a home icon and navigation links for 'Accueil' and 'Connexion'. The main content area displays a user profile icon, a 'Username' field containing 'Admin', a 'Password' field with masked characters, and a prominent blue 'Login' button.

Figure 5.15 Interface de connexion

5.4.2. L'interface de la gestion des projets

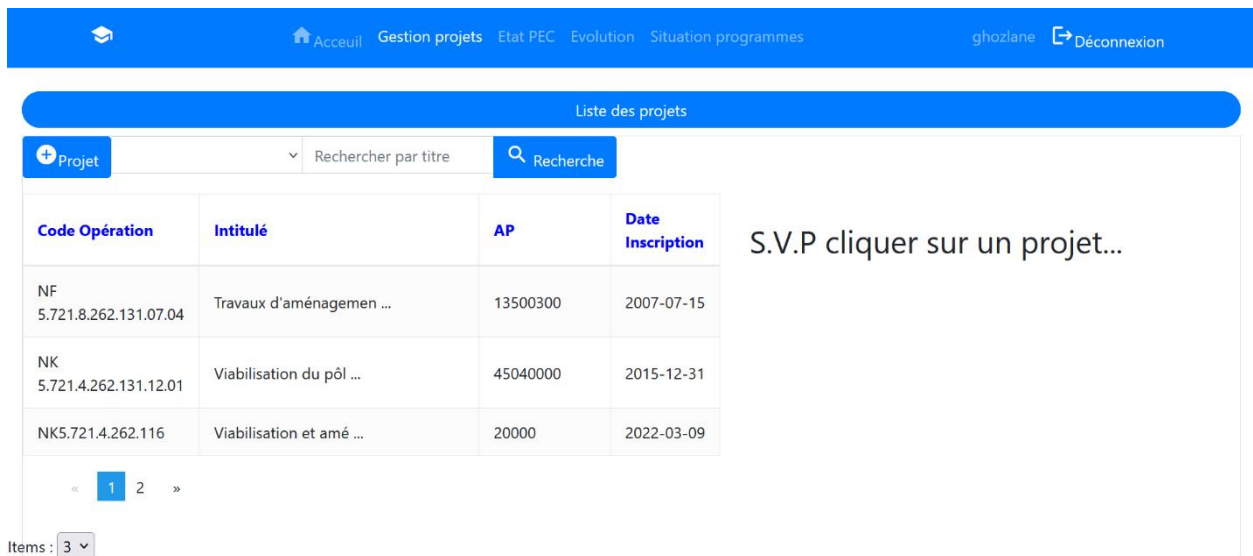


Figure 5.16 displays the project management interface. The header bar includes navigation links for 'Accueil', 'Gestion projets', 'Etat PEC', 'Evolution', and 'Situation programmes', along with a user profile and 'Déconnexion' link. The main content area features a 'Liste des projets' title, a search bar, and a table listing projects. The table has columns for 'Code Opération', 'Intitulé', 'AP', and 'Date Inscription'. The table lists three projects. A pagination bar shows '1' of 2 items, and a dropdown indicates 'Items : 3'.

Code Opération	Intitulé	AP	Date Inscription
NF 5.721.8.262.131.07.04	Travaux d'aménagemen ...	13500300	2007-07-15
NK 5.721.4.262.131.12.01	Viabilisation du pôl ...	45040000	2015-12-31
NK5.721.4.262.116	Viabilisation et amé ...	20000	2022-03-09

Figure 5.16 Interface de la gestion des projets

5.4.3. L'interface de détail d'un projet

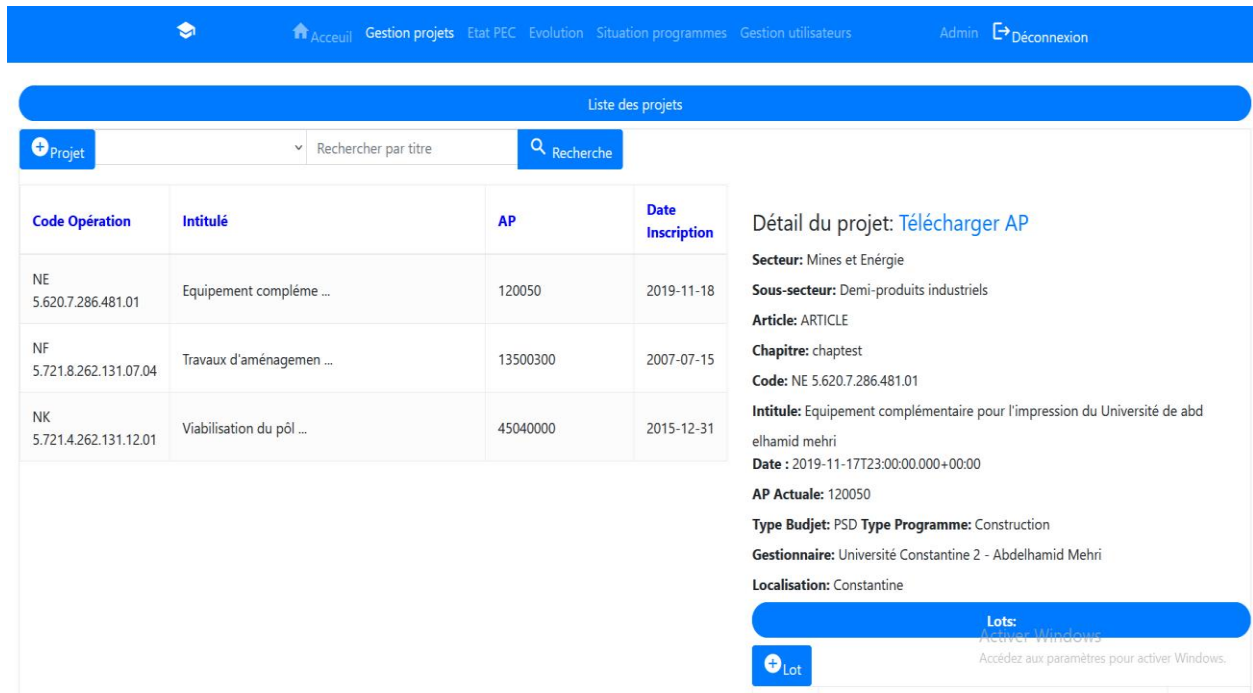


Figure 5.17 Interface de détail d'un projet

5.4.4. L'interface ajouter projet

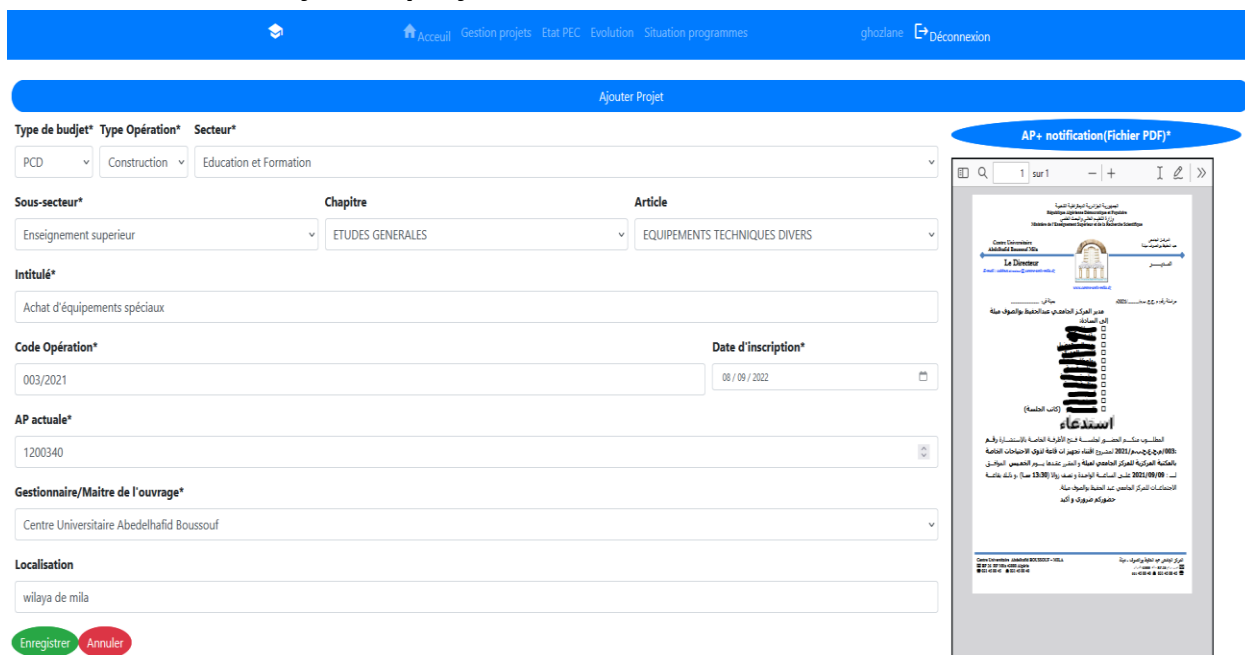


Figure 5.18 Interface ajouter projet

Chapitre 05 : Implémentation du système

5.4.5. L'interface de la liste des avancements

Liste des avancements physiques et financières										
+ Avancement										
Etat	Description	Date	Avancement %	Consommation	Dépense	Date Prévu	Impact	Observation	Attachement	Actions
Cloturé	Décision de clôture	2023-11-18	2	32200	1	2023-11-18	/	Acquisition d'équipements complémentaires pour l'imprimerie à l'Université de Constantine	file	 
Lancé	/	2019-11-18	1	30220	1	2019-11-18	/	/	file	 

Items per Page: 3

Figure 5.19 Interface de la liste des avancements

5.4.6. L'interface du formulaire pour ajouter un marché

Ajouter Marché				
Type de marché*	Désignation*	Délai exécution*	Montant Marché *	Attachement(Fichier PDF) *
Gré a Gré	Equipement de biologie 2021	1 ans	30035	
Retenu de garantie	Observation			
10000	/			
Entreprise (Soumissionnaire)	Date ODS			
sarl kourmel	15 / 04 / 2021			
				

Figure 5.20 Interface ajouter marché

Chapitre 05 : Implémentation du système

5.4.7. L'interface de la liste des évolutions

[+ Nouvelle Evolution](#)

Liste des Evolutions Physiques

Année	Nb universitaires	Nb instituts	Nb départements	Nb place Pédagogiques	Nb place Hébergé	Nb Bibliothèques	Nb Restaurant	Nb Logements Enseignements	Actions
1	1	6	8	92	2000	4	1	1	
2030	4	6	80	4000	3	2	1	1	
89	1	4	6	65	1000	3	67	5	

Liste des Evolutions Humaines

Année	Nb Etudiant	Nb Etudiant étrangères	Nb Etudiant hébergé	Nb Enseignant	Nb Employé administratifs	Nb Etudiant admis	Nb Etudiant ajournée	Nb Etudiant Diplômé	Actions
1	10606	11	5000	439	2	10002	570	10036	
2030	0	1	300	600	4300	1419	500	650	
89	10853	78	4000	300	475	1200	350	700	

« Prev 1 2 3 4 5 Next »

Figure 5.21 Interface de la liste des évolutions

5.4.8. L'interface de de La situation mensuelle des programmes

Accueil
Gestion projets Etat PEC Evolution Situation programmes
ghozlane Déconnexion

Situation mensuelle des programmes par sous-secteur

v
Recherche

Sous-secteur	Opération	Date inscription	Localisation	AP act	Etat	Consomation Milliers DA	Taux physique	OBSERVATION
EDUCATION	Entretien et réparat ... 32/2022	2022-08-22	Wilaya de mila	212000	Lancé	1	1%	/
EDUCATION	Projet des frais d'a ... 18/2023	2023-05-20	wilaya mila	8000	Cloture	2	1%	/
EDUCATION	Projet de réhabilita ... 002/	2020-12-17	wilaya de mila	10000	A l'arret	1	1%	x

« 1 »

Figure 5.22 Interface de La situation mensuelle des programmes

5.5 Conclusion

Dans ce chapitre, nous avons présenté le côté implémentation de notre projet en spécifiant les outils, les langages et l'environnement de développement ainsi que les interfaces les plus significatives de notre système.

Conclusion générale

Conclusion générale

Dans ce mémoire, nous avons proposé un système d'informations sécurisées pour le suivi du développement local au niveau du ministère de l'enseignement supérieur et la recherche scientifique basé sur l'architecture microservices. L'objectif de ce système est de fournir aux responsables du secteur de l'enseignement supérieur une base de données fiable et actualisée leur permettant de contrôler et de superviser toutes les opérations ainsi que de les aider à prendre les décisions appropriées.

Afin de réaliser notre système, nous avons commencé l'étude dans le sous-direction du développement et de la prospective (SDDP) du centre universitaire Abdelhafid Boussouf – Mila. Ensuite, nous avons élaboré une conception et une modélisation basée sur le processus UP et le langage UML. Plus précisément, nous avons décrit notre système en utilisant les diagrammes de cas d'utilisations, d'états-transitions, de séquences et de classes. Finalement, nous avons implémenté notre système en utilisant le Framework Angular dans le front-end et le framework Spring Boot, Spring cloud et Spring Security dans le back-end. En plus, nous avons utilisé le SGBD PostgreSQL.

Ce travail a été pour nous une expérience très enrichissante et très intéressante, dans la mesure où il nous a permis d'enrichir nos connaissances d'une part sur le développement d'applications basées sur l'architecture microservices et d'autre part sur la gestion des processus administratifs notamment la gestion de budget ainsi que les marchés publics.

Dans un travail future, nous planifions de déployer notre système dans le sous-direction du développement et de la prospective (SDDP) du centre universitaire Abdelhafid Boussouf – Mila. Nous planifions également de développer un nouveau module de gestion des projets conformément à Loi organique 18-15 qui est destiné aux Institutions publiques de nature managériale et institutions similaires.

Annexe 1 : Décision de clôture d'un programme

ANNEXE 04- DÉCISION DE CLÔTURE-
République Algérienne Démocratique et Populaire

ولاية

N° : رقم : le :
N° FIXE : الرقم الثابت :

Secteur :	القطاع :
Sous – Secteur :	القطاع الفرعي :
Chapitre :	الفصل :
Article :	المادة :
Gestionnaire :	المدير :
Opération n° :	عملية رقم :

- NORMALE ☐ - عادي

DECISION DE CLOTURE - CONTENTIEUSE ☐ - تنازعي

- ABANDON ☐ - ترك

مقرر الاختتام :

إن والي ولاية

- يهتفي القانون رقم 10-11 المؤرخ في 22 جوان 2011 المتضمن قانون البلدية.

- ويهتفي القانون رقم 07-12 المؤرخ في 21 فيفري 2012 المتضمن قانون الولاية.

- ويهتفي الرسوم الرئسي المؤرخ في المتضمن تعيين السيد والي ولاية ..

- ويهتفي مقرر البرنامج لسنة المؤرخ في ..

- ويهتفي مقرر برنامج إعادة التقييم لسنة المؤرخ في ..

DECIDE يقرر سيلي

المادة 1 : العملية المشار إلى رقمها فيما أعاد والي نصها قد اختلعت وسحبت من قائمة الاستثمارات المخططة.

Article 1 : L'opération, dont le numéro est indiqué ci-dessus et dont le libellé suit, est clôturée et retirée de la nomenclature des investissements planifiés

المادة 2 : الكلفة النهائية عند اختتام هذه العملية تقدر بـ ..

Article 2 : Le coût final à la clôture de cette opération est de :

المادة 3 : يوجد كل من تفصيل هيكل الكلفة ونوع التمويل الجدول الزمني وتحديد الموقع وأثر الاستثمار على التوالي في الجداول: أ، ب، ج، د، هـ التالية:

Article 3 : La structure du coût, la nature du financement, l'échéancier, la localisation est les effets de l'investissement sont détaillés respectivement dans les tableaux A,B,C,D, et E suivants :

-2-

تذكير رقم العملية : ع و 03.06.116.262.2.623.5

Rappel Opération N° :

A- STRUCTURE DU COUT (10³ DA) ١- هيكل الكلفة (10³ دج)

تفصيل الهيكلية Rubriques	الكلفة الأولى Coût initial	الكلفة الأخيرة المسجلة Dernier coût inscrit (1)	الكلفة الأخيرة Coût final (2)	الفرق Ecart (1-2)
الدراسات وأثر الهندسة 01 Etude et/ou engineering				
البناء وما ارتبط به من هندسة مدنية 02 Bâtiment et génie-Civile lie				
الأشغال العمومية 03 Travaux Publics				
الآلات والتجهيزات 04 Machines et Equipements				
عقد النقل والتوزيع 05 Matériel de Transport et de Maintenance				

الدراسات و/أو الهندسة 06 Formation				
تقديم الخدمات الخارجية 07 Prestation de Services Extérieur				
المخزون - الأتلي 08 Stock Outils				
غير ذلك 09 Autres				
المنشآت الأساسية المحيطة Infrastructures environnantes				
الأرضية Terrain				
القوائد الإضافية -Intérêts intercalaires				
حقوق الجمرک و الرسوم Droits de douanes et taxes				
Installation chauffage et équipement de la bâche à eau				
Poste transformateur				
CTC				
Raccordement électrique, eau et gaz				
Frais ANEP				
القوائد الإضافية 98 Montant opération non ventilée				
99 المجموع TOTAL				

B- NATURE DE FINANCEMENT (10, DA)

ب- نوع التمويل (10 و دج)

التمويل Financement	مساهمات ميزانية الدولة Concours budgétaires de l'Etat	قروض الخزينة Prêts du Trésor	غير ذلك Autres	المجموع Total
المبلغ السابق				
المبلغ الحالي				

-3-

تذكير رقم العملية :

الاستكمال	Achèvement	الانطلاق	Démarrage	التسجيل	Inscription	ج - الجدول الزمني للإنجاز
Tr.	An	Tr.	An	An	Mois	C- Echancier de Réalisation

الخارج Extérieur	بين الولايات Int. Wilaya	بلدية Commune	ولاية Wilaya	تحديد الموقع	د.الموقع
					D-LOCALISATION
					Initial السابق
					Final الأخير

E.1- EFFETS OU CONSISTANCE PHYSIQUE DE

ي.1- آثار الاستثمار أو المحتوى المادي

L'INVESTISSEMENT

المنتج أو نوع العمل أو العمل Produits ou nature de l'ouvrage ou l'action		وحدة القياس Unité de mesure		الفترة المنتهية أو المساحة المشيدة الأخيرة	الفترة المنتهية أو المساحة المشيدة الأخيرة	فرق
الرمز Code	بالنصوص ح En Clair	الرمز Code	بالنصوص En Clair	Capacité installée ou dimension physique initiale (1)	Capacité installée ou dimension physique finale (2)	Ecart (2-1)

ي. 2- مناصب العمل الدائمة المباشرة التي أحدثها الاستثمار بعد الإنجاز :

E-2- EMPLOIS PERMANENTS DIRECTS CREEES PAR L'INVESTISSEMENT APRES REALISATION :

Antérieur : السابق :
 ou de journées de travail : أو أيام العمل :
 Actuel : الحالي :

المادة 4 : يكلف المرسل إليهم المشار إليهم فيما يخصه ، بتنفيذ هذا المقرر.

Article 4: Les destinataires indiqués ci-après sont chargés, chacun en ce qui concerne, de l'exécution de la présente décision.

DESTINATAIRES:	المرسل إليهم السادة :
مدير التجهيزات العمومية لولاية	
أمين الخزينة لولاية	
المراقب المالي لولاية	

-4-

تذكير رقم العملية : ع.ق. 03.06.116.262.2.623.5

Rappel Opération N°:

- AUTRES INFORMATIONS SUR LA CLÔTURE DE L'OPERATION. AUTRES INFORMATIONS SUR LA CLÔTURE DE L'OPERATION.

a) Clôture contentieuse: préciser les éléments et les mesures prises en vue de son règlement, ainsi que le montant restant dû par l'opération ou, éventuellement, le trop-perçu par le réalisateur.

b) Abandon: donner les causes de cet abandon en précisant les dates de signature du contrat, d'ouverture et d'abandon du chantier ou de défaillance du fournisseur.

تذكير رقم العملية : ع.ق. 01.13.116.262.1.622.5

Rappel Opération N°:

AUTRES CARACTERISTIQUES

1-ECHEANCIER PREVISIONNEL DES PAIEMENTS (10³ DA)

المميزات الأخرى
1 - الجدول الزمني التقديري للمدفوعات (10³ دج)

المبالغ ب :						
Montants en:						
الدينارات						
Dinars						
العملة						
Devises directes						
المجموع						
Total						

MONTANT TOTAL DE L'AUTOFINANCEMENT (10³ DA)

- مبلغ إجمالي للتمويل الذاتي (10³ دج)

2- AUTRES INFORMATIONS :

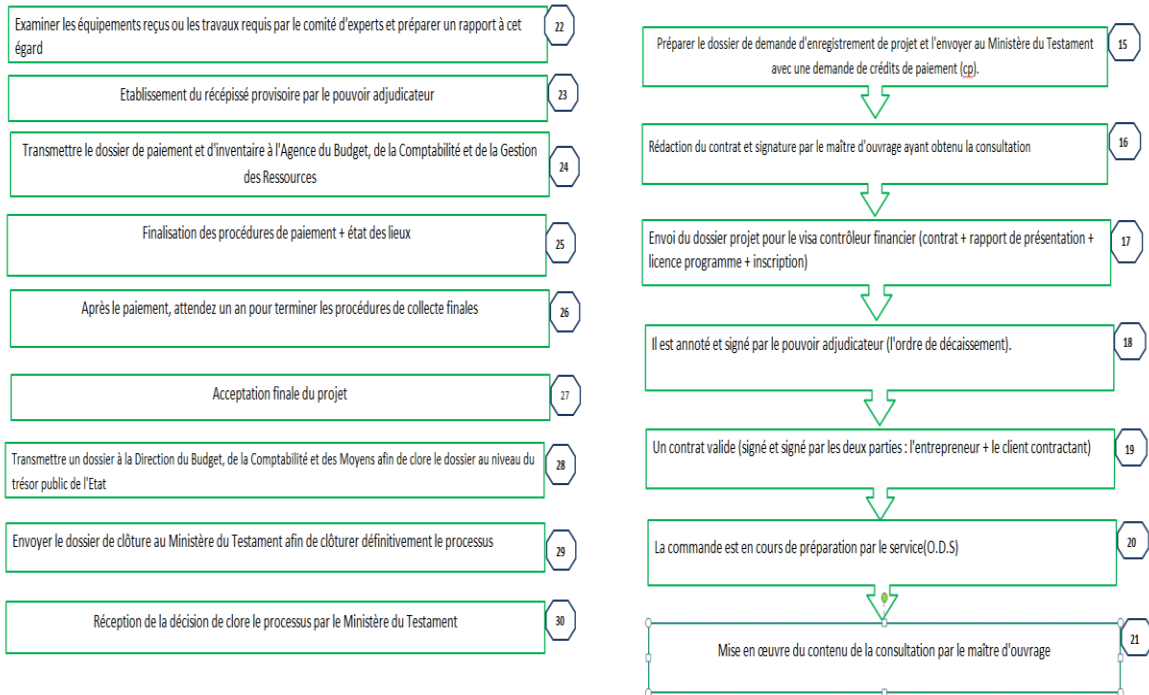
Unité en milliers de dinars

Détail de l'opération :

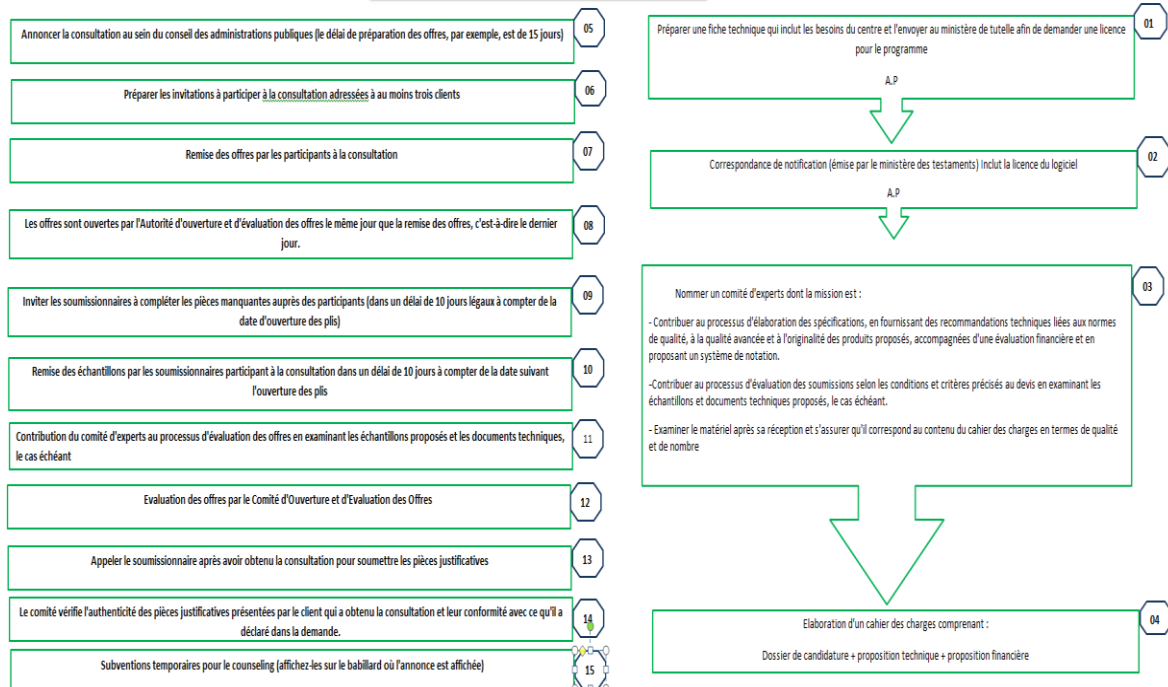
LOTS	Situation actuelle
Etude	
Sous total affecté	
Montant non ventilé	
Total	

Annexe 2 : étapes de consultation

Étapes du conseil



Étapes du conseil

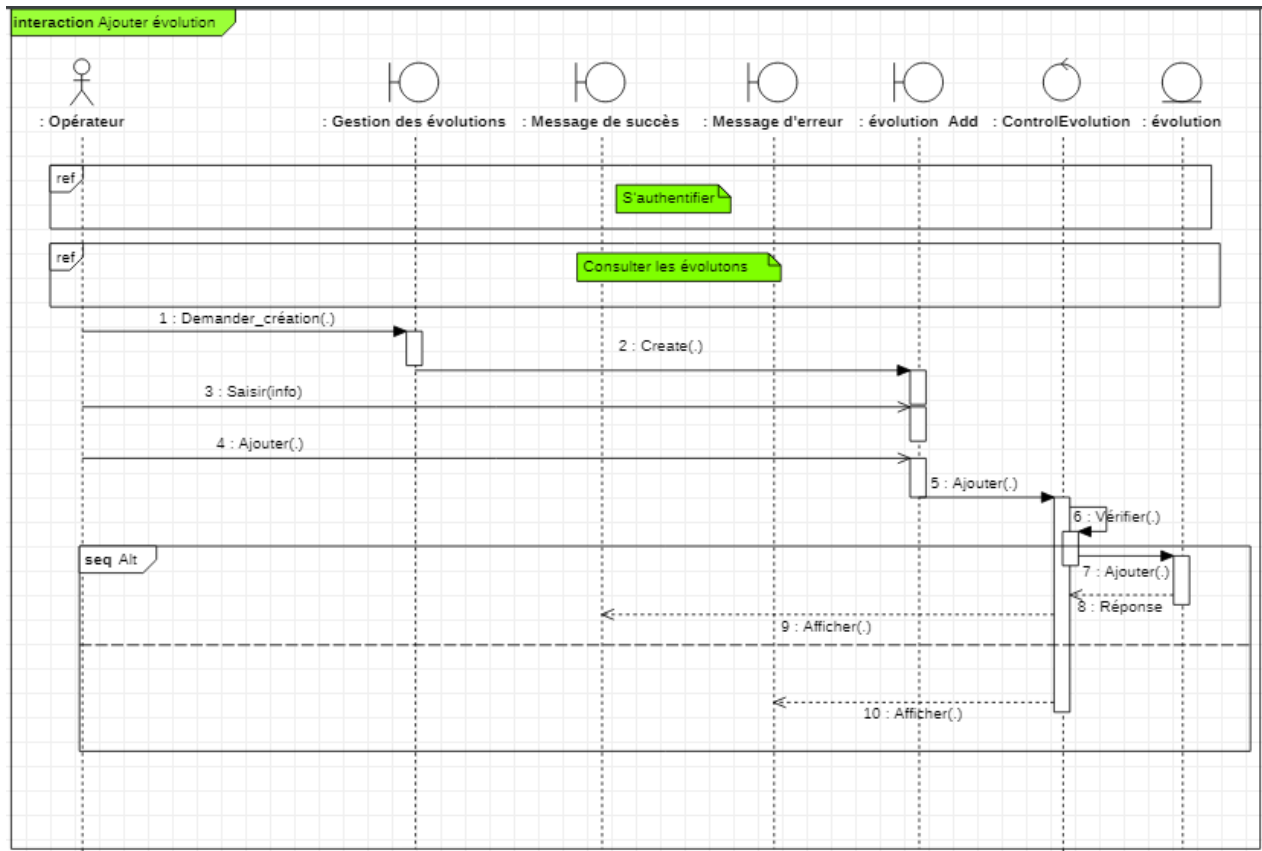


Annexe 3 : Les scénarios et les diagrammes de séquence et états-transitions➤ **Gestion des statistiques****1. Evolution**• **Ajouter évolution****a) Description textuelle**

Cas d'utilisation :	Ajouter évolution .
Acteur :	Opérateur.
Résumé :	Ce cas d'utilisation permet à l'acteur d'ajouter une évolution d'un projet.
Précondition :	L'opérateur consulte les évolutions.
Post condition :	L'évolution est ajoutée.
Description du scénario nominal :	1. L'opérateur demande la création d'évolution. 2. Le système affiche le formulaire d'ajoute. 3. L'opérateur saisit les informations de l'évolution et enregistrer. 4. Le système lit et vérifie les paramètres. 5. Le système affiche un message d'information.
Description du scénario alternatif :	3-a L'opérateur annuler l'ajoute d'évolution.

Tableau 3.19 : Description de cas d'utilisation ajouter évolution

b) Diagramme de séquence



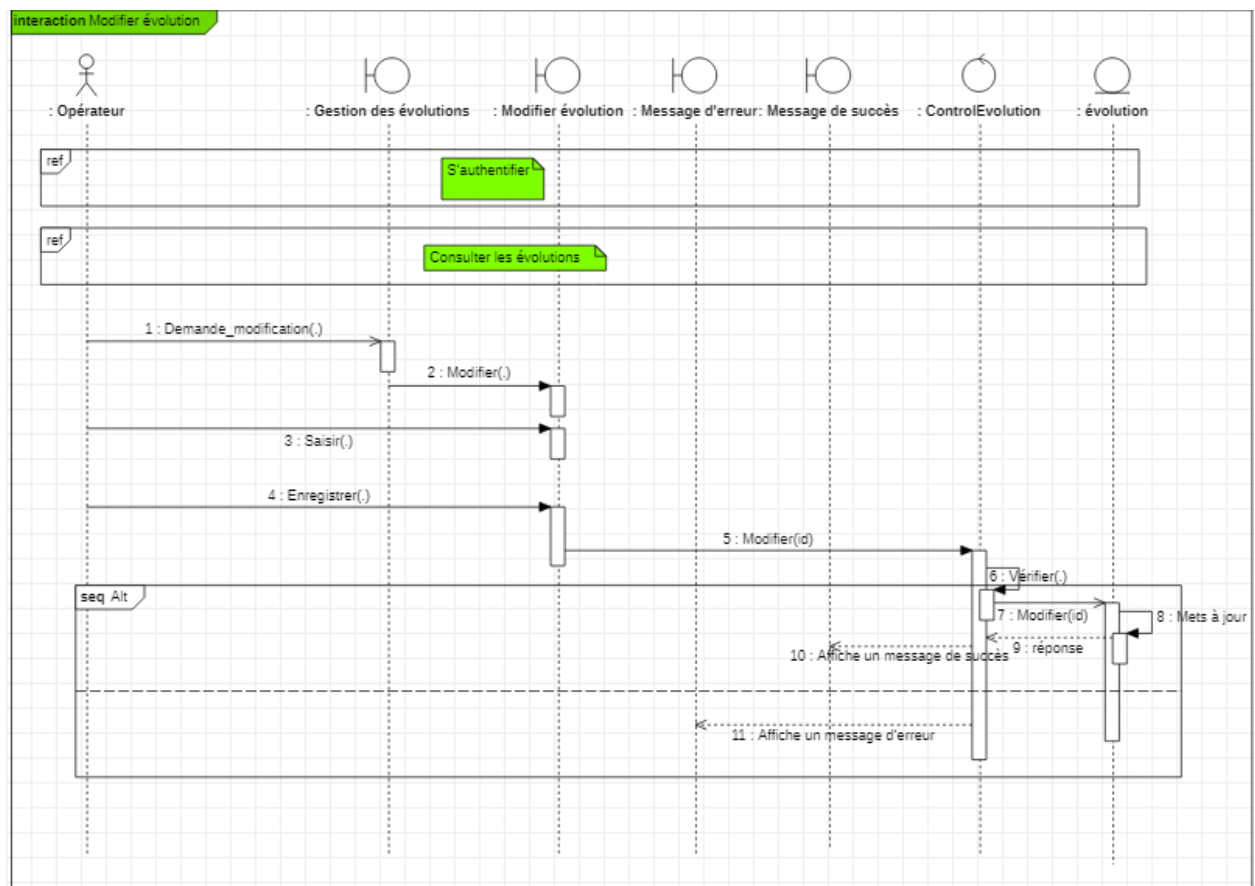
Annexe

- **Modifier évolution**

a) Description textuelle

Cas d'utilisation :	Modifier évolution.
Acteur :	Opérateur.
Résumé :	Ce cas d'utilisation permet à l'acteur de modifier l'évolution.
Précondition :	L'opérateur consulte les évolutions.
Post condition :	L'évolution est modifiée.
Description du scénario nominal :	1. L'opérateur choisit une évolution qu'il veut modifier. 2. Le système affiche les détails de projet. 3. L'opérateur saisit les modifications et enregistrer. 4. Le système vérifie les paramètres. 5. Le système effectue la modification de l'évolution et affiche un message de d'information.
Description du scénario alternatif :	5-a – le système affiche un message d'erreur.

b) Diagramme de séquence



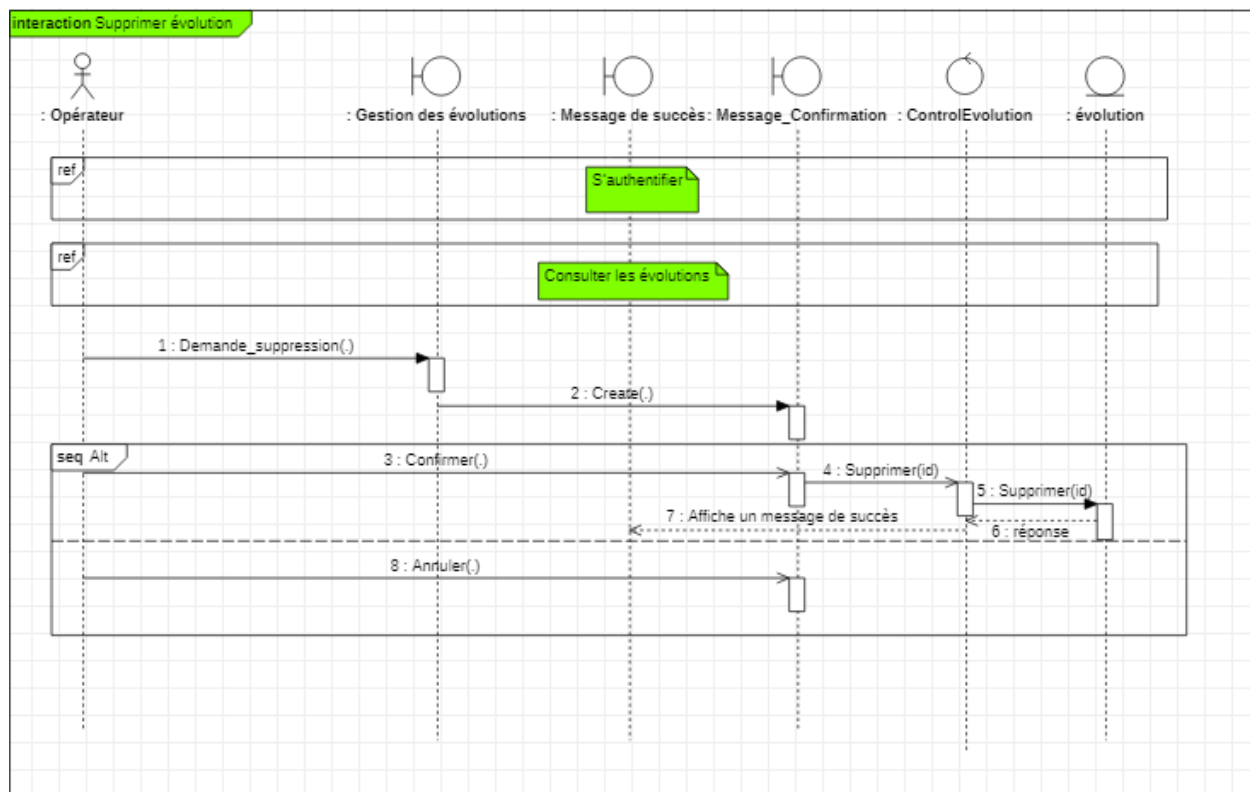
Annexe

- **Supprimer évolution**

a) Description textuelle

Cas d'utilisation :	Supprimer évolution.
Acteur :	Opérateur.
Résumé :	Ce cas d'utilisation permet à l'acteur de supprimer une évolution.
Précondition :	L'opérateur consulte les évolutions.
Post condition :	L'évolution est supprimée.
Description du scénario nominal :	1. L'opérateur choisit une évolution et demande sa suppression. 2. Le système affiche un message de confirmation. 3. L'opérateur confirme la suppression de l'évolution. 4. Le système effectue la suppression de l'évolution et affiche un message d'information.
Description du scénario alternatif :	3-a - L'opérateur annuler la suppression de l'évolution.

b) Diagramme de séquence

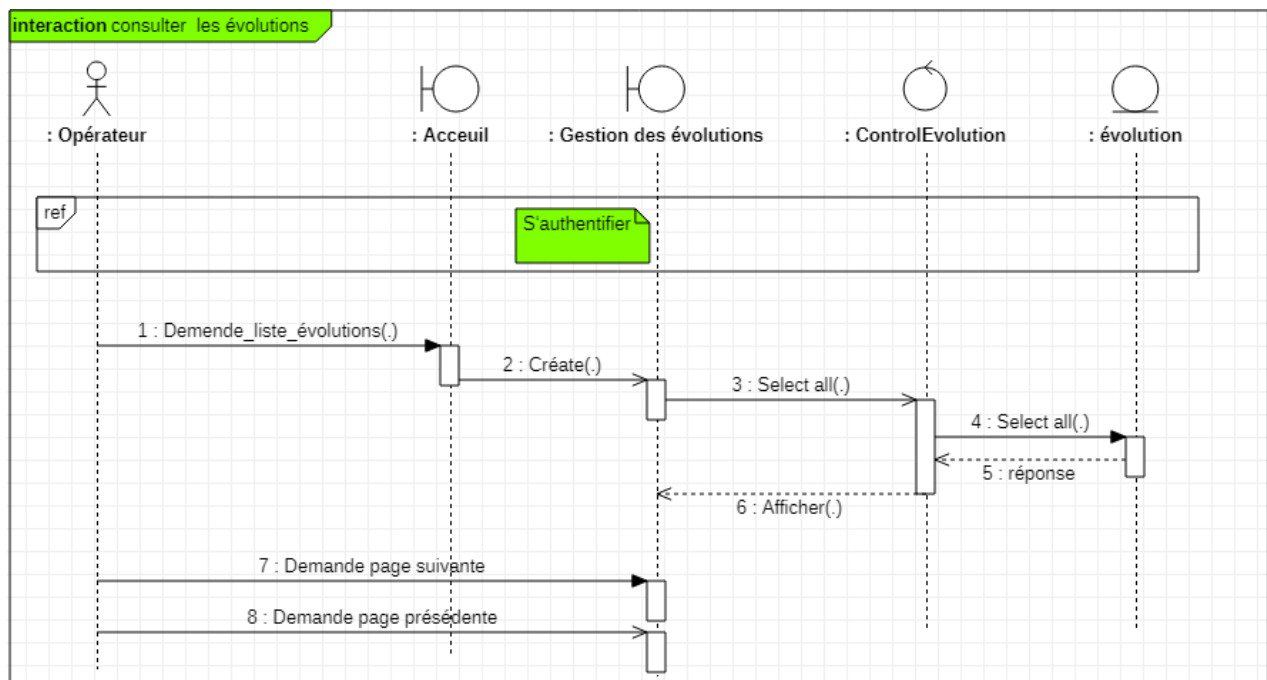


- Consulter les évolutions

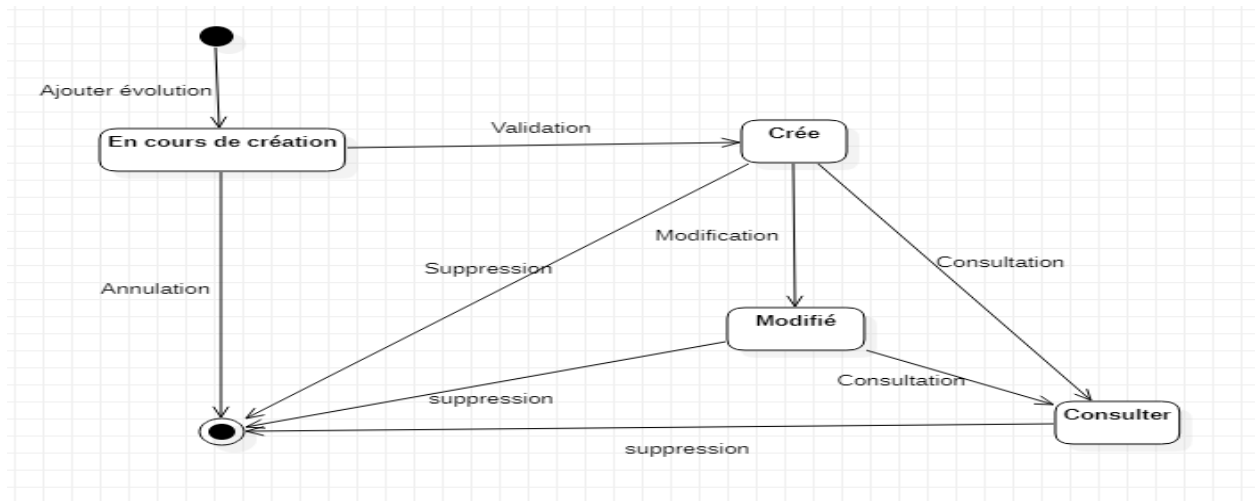
a) Description textuelle

Cas d'utilisation :	Consulter les évolutions.
Acteur :	Opérateur.
Résumé :	Ce cas d'utilisation permet à l'acteur de consulter la liste des évolutions.
Précondition :	L'opérateur est authentifié.
Post condition :	La liste des évolutions est affichée.
Description du scénario nominal :	1. L'utilisateur demande la page de gestion des évolutions. 2. Le système affiche la liste des évolutions enregistrées.
Description du scénario alternatif :	2-a – le système affiche liste vide.

b) Diagramme de séquence



- Diagramme d'état transition : Evolution



2. Situation des programmes

- Ajouter situation des programmes

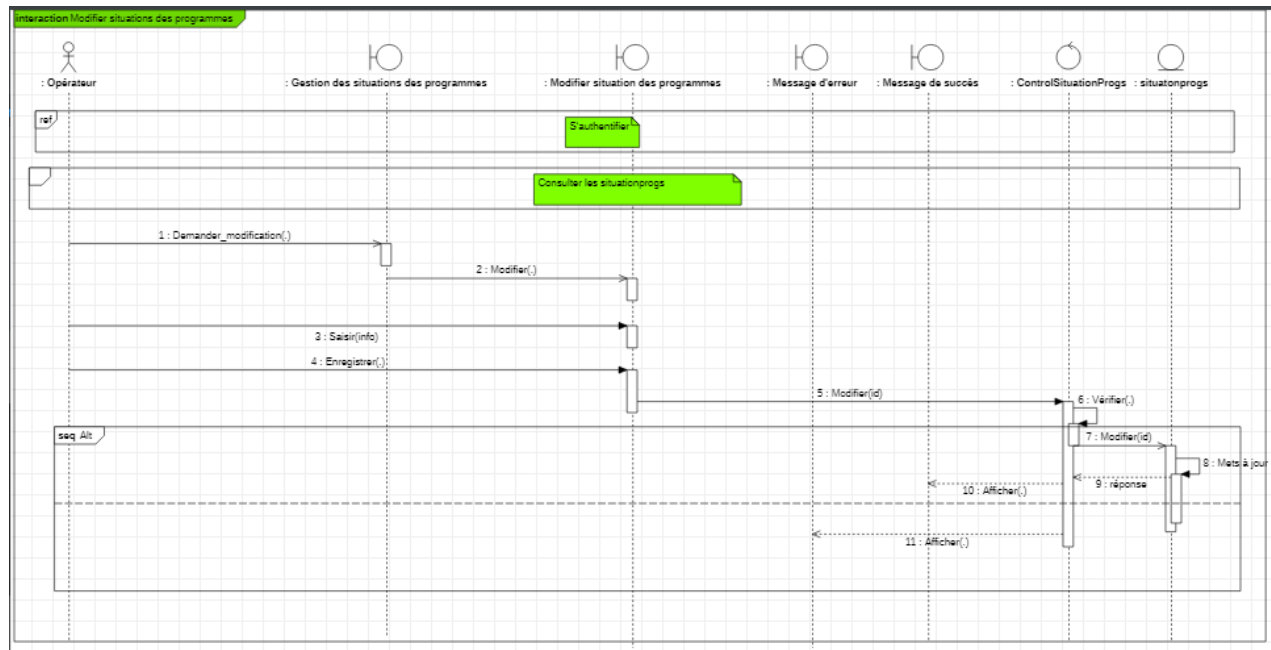
a) Description textuelle

Cas d'utilisation :	Ajouter situation des programmes .
Acteur :	Opérateur.
ORésumé :	Ce cas d'utilisation permet à l'acteur d'ajouter une situation de programme.
Précondition :	L'opérateur consulte les situationprogs.
Post condition :	La situation des programmes est ajoutée.
Description du scénario nominal :	1. L'opérateur demande la création de la situation des programmes. 2. Le système affiche le formulaire d'ajoute. 3. L'opérateur saisit les informations de la situation de programme et enregistrer. 4. Le système lit et vérifie les paramètres. 5. Le système affiche un message d'information.
Description du scenario alternatif :	3. a- L'opérateur n'est pas rempli des champs obligatoires, le système indique au l'opérateur que le formulaire contient des champs vide ou incorrect et lui proposer de remplir. 3. b- verre à l'étape 3.

c) Beschreibung textuelle

Cas d'utilisation :	Modifier situation des programmes.
Acteur :	Opérateur.
Résumé :	Ce cas d'utilisation permet à l'acteur de modifier les informations d'une situation des programmes.
Précondition :	L'opérateur consulte les situationprogrs.
Post condition :	La situation des programmes est changée.
Description du scénario nominal :	1. L'opérateur choisit une situation qu'il veut modifier faire les modifications 2. Le système affiche le formulaire pour faire les modifications 3. L'opérateur saisit les modifications et Enregistrer. 4. Le système vérifie et effectue la modification de la situation et affiche un message d'information.
Description du scenario alternatif :	4-a – le système affiche un message d'erreur

b) Diagramme de séquence

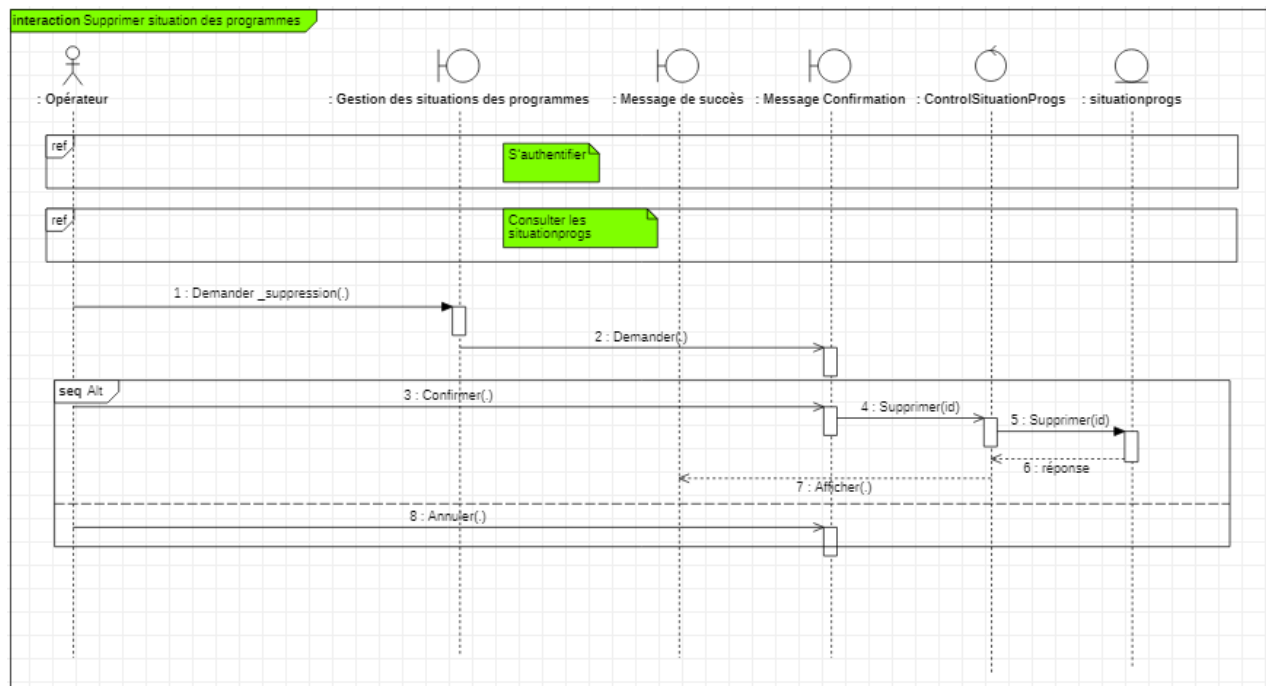


- Supprimer situation des programmes

a) Description textuelle

Cas d'utilisation :	Supprimer situation des programmes.
Acteur :	Opérateur.
Résumé :	Ce cas d'utilisation permet à l'acteur de supprimer une situation des programmes.
Précondition :	L'opérateur consulte situationprogs.
Post condition :	La situation des programmes est supprimée.
Description du scénario nominal :	1. L'opérateur choisit une situation de programme et demande sa suppression. 2. Le système affiche un message de confirmation. 3. L'opérateur confirme la suppression de situation des programmes. 4. Le système effectuer la suppression de situation et affiche un message d'information.
Description du scénario alternatif :	3-a - L'opérateur annuler la suppression de situation de programme.

b) Diagramme de séquence

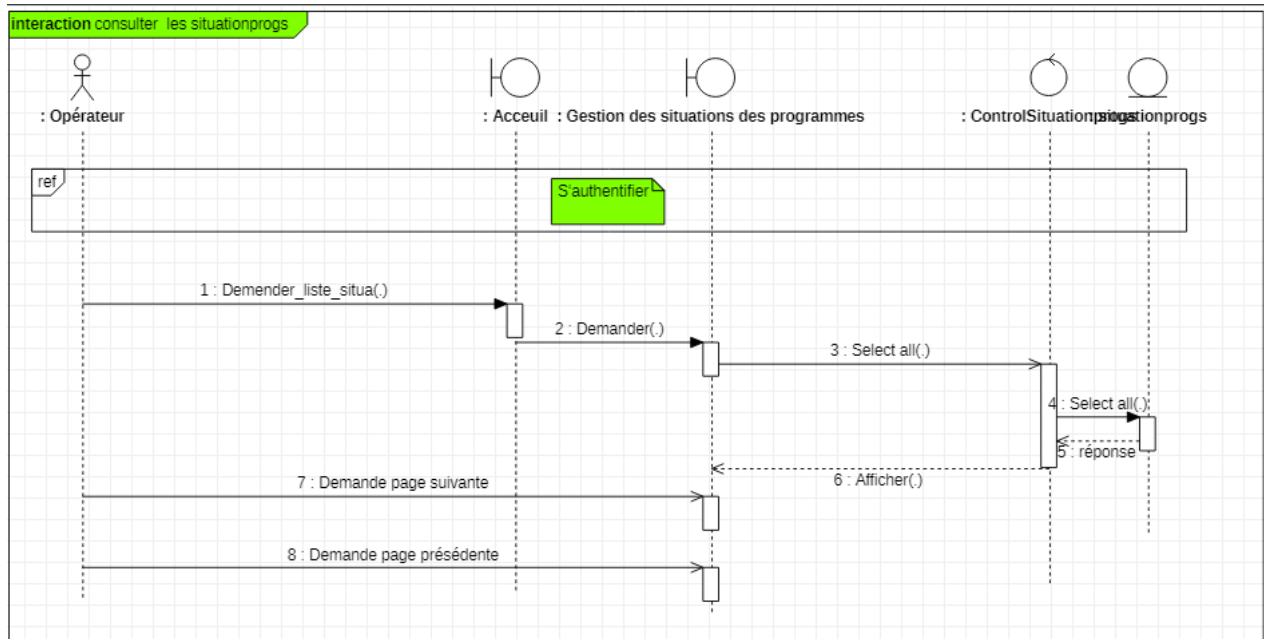


- Consulter les situations des programmes

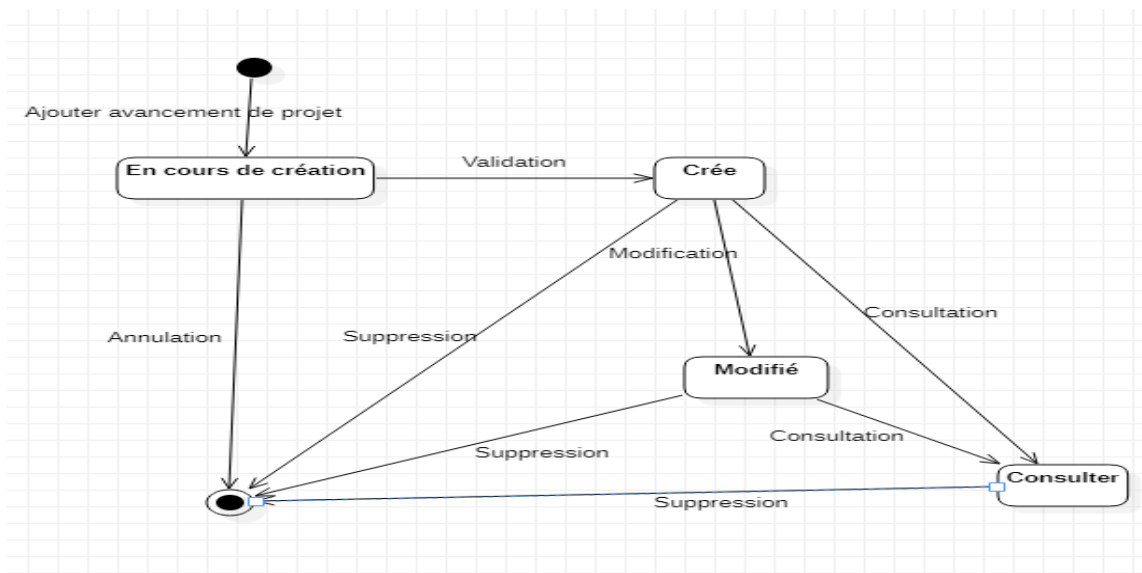
a) Description textuelle

Cas d'utilisation :	Consulter les situations des programmes.
Acteur :	Opérateur.
Résumé :	Ce cas d'utilisation permet à l'acteur de consulter la liste des situations des programmes.
Précondition :	L'opérateur est authentifié.
Post condition :	La liste des situations des programmes est affichée.
Description du scénario nominal :	1. L'utilisateur demande la page des situations des programmes. 2. Le système affiche la liste des situations des programmes enregistrées.
Description du scénario alternatif :	2-a – le système affiche liste vide.

b) Diagramme de séquence



- Diagramme d'état transition : Situation des programmes



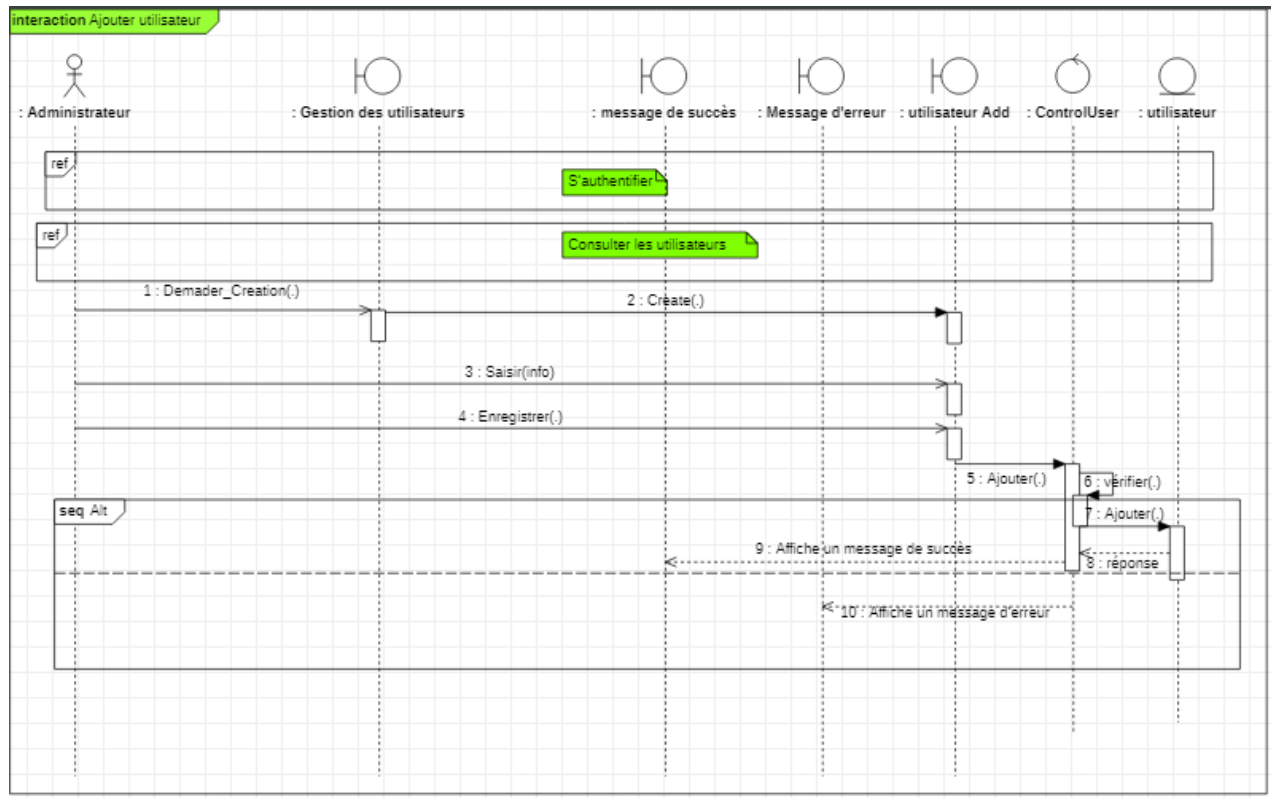
➤ Gestion des utilisateurs

- Ajouter utilisateur

a) Description textuelle

Cas d'utilisation :	Ajouter utilisateur.
Acteur :	Administrateur .
Résumé :	Ce cas d'utilisation permet à l'acteur d'ajouter un utilisateur.
Précondition :	L'administrateur consulte les utilisateurs.
Post condition :	L'utilisateur est ajouté.
Description du scénario nominal :	1. L'administrateur demande la création d'utilisateur. 2. Le système affiche le formulaire d'ajoute. 3. L'administrateur saisit les informations d'utilisateur et enregistrer. 4. Le système lit et vérifie les paramètres et affiche un message d'information.
Description du scénario alternatif :	3-a - L'administrateur n'est pas rempli des champs obligatoires, le système indique au l'opérateur que le formulaire contient des champs vide ou incorrect et lui proposer de remplir. 3- b- verre à l'étape 3.

b) Diagramme de séquence

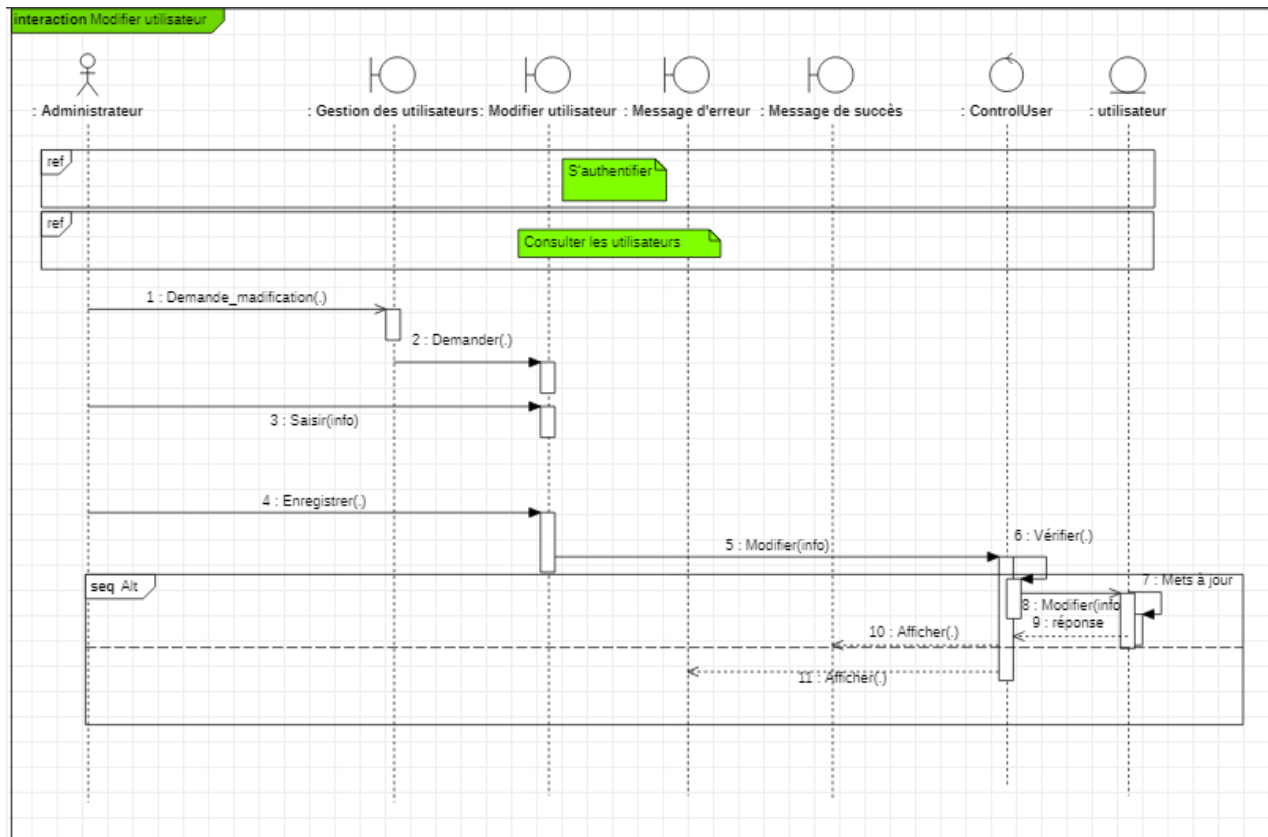


- **Modifier utilisateur**

a) Description textuelle

Cas d'utilisation :	Modifier utilisateur.
Acteur :	Administrateur.
Résumé :	Ce cas d'utilisation permet à l'acteur de modifier utilisateur.
Précondition :	L'administrateur consulte les utilisateurs.
Post condition :	Utilisateur est modifiée.
Description du scénario nominal :	1. L'administrateur choisit un utilisateur qu'il veut modifier. 2. Le système affiche les détails d'utilisateur. 3. L'administrateur saisit les modifications et enregistrer. 4. Le système vérifie les paramètres et effectue la modification d'utilisateur et affiche un message de d'information.
Description du scénario alternatif :	3-a – le système affiche un message d'erreur.

b) Diagramme de séquence

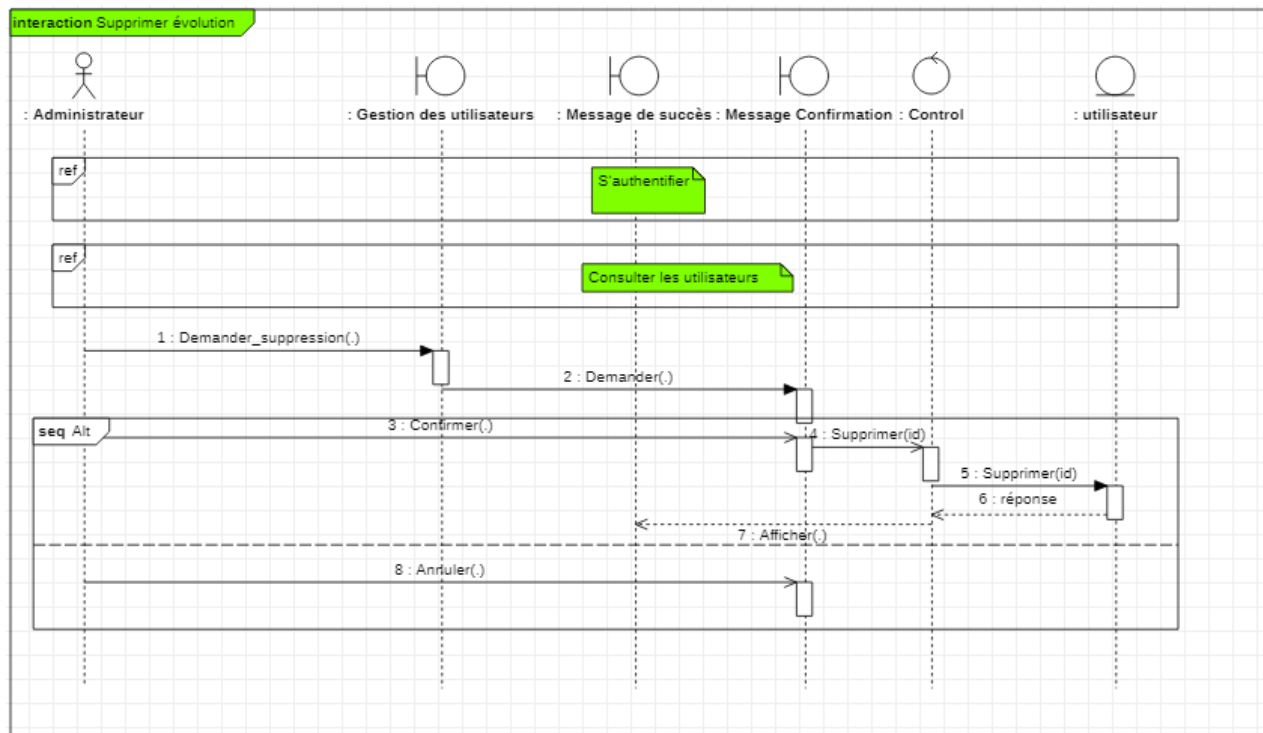


- Supprimer utilisateur

a) Description textuelle

Cas d'utilisation :	Supprimer utilisateur .
Acteur :	Administrateur.
Résumé :	Ce cas d'utilisation permet à l'acteur de supprimer un utilisateur.
Précondition :	L'administrateur consulte les utilisateurs.
Post condition :	L'utilisateur est supprimée.
Description du scénario nominal :	1. L'administrateur choisit un utilisateur et demande sa suppression. 2. Le système affiche un message de confirmation. 3. L'administrateur confirme la suppression d'utilisateur. 4. Le système effectue la suppression d'utilisateur et affiche un message d'information.
Description du scénario alternatif :	3-a - L'administrateur annule la suppression d'utilisateur.

b) Diagramme de séquence

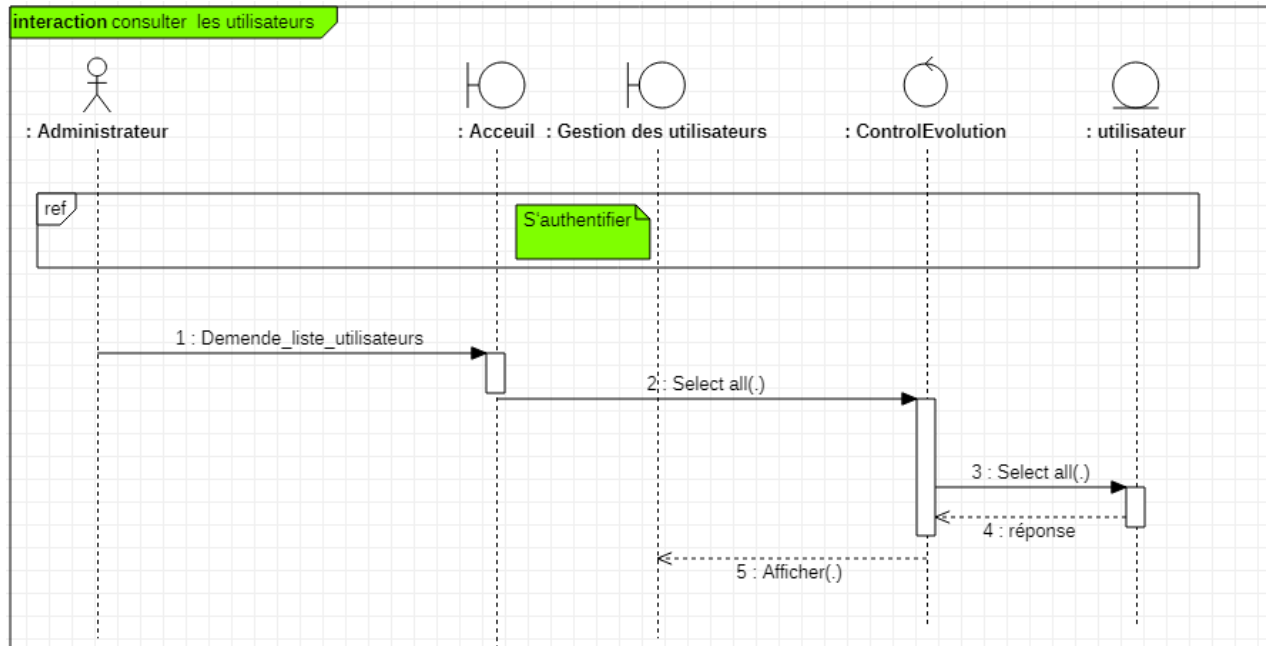


- Consulter les utilisateurs

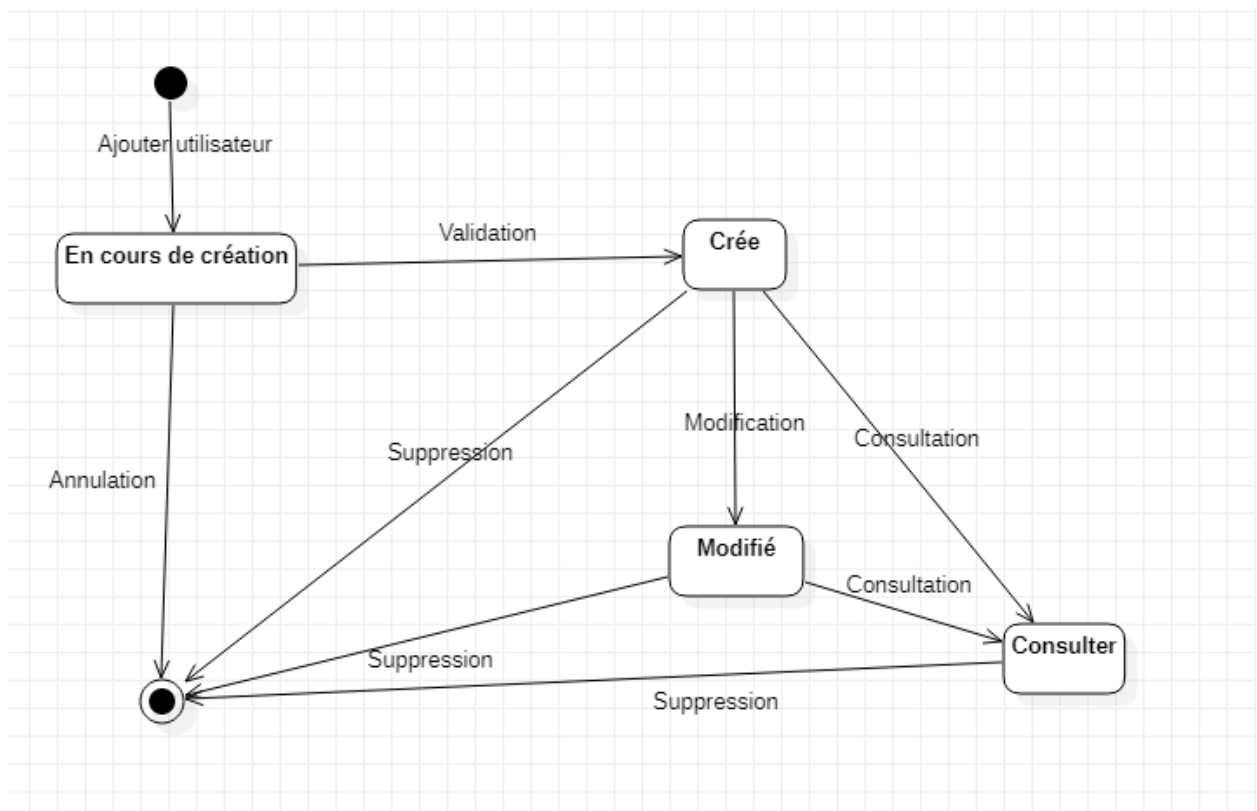
a) Description textuelle

Cas d'utilisation :	Consulter les utilisateurs.
Acteur :	Administrateur.
Résume :	Ce cas d'utilisation permet à l'acteur de consulter la liste des utilisateurs.
Précondition :	Administrateur est authentifié.
Post condition :	La liste des utilisateurs est affichée.
Description du scénario nominal :	1. L'administrateur demande la liste des Utilisateurs. 2. Le système affiche la liste des utilisateurs enregistrées.
Description du scénario alternatif :	2-a – le système affiche liste vide.

b) Diagramme de séquence



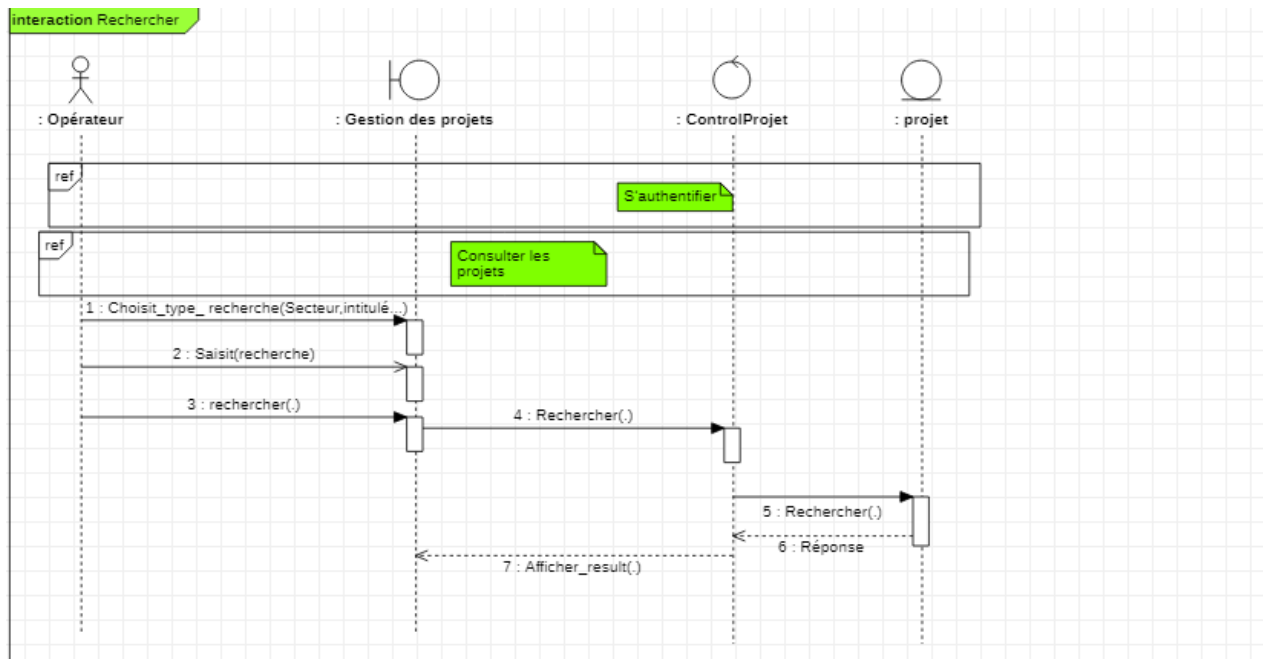
• Diagramme d'état transition : Utilisateur



- **Rechercher projet**
- a) Description textuelle**

Cas d'utilisation :	Rechercher projet.
Acteur :	Opérateur.
Résumé :	Ce cas d'utilisation permet à l'acteur de rechercher.
Précondition :	L'opérateur consulte les projets.
Post condition :	L'opérateur fait la recherche.
Description du scénario nominal :	1. L'opérateur choisit le type de recherche. 2. L'opérateur saisit la recherche et recherché. 3. Le système affiche les résultats de la recherche selon les choix et la demande de l'opérateur.
Description du scénario alternatif :	5-a – le système affiche liste vide

b) Diagramme de séquence

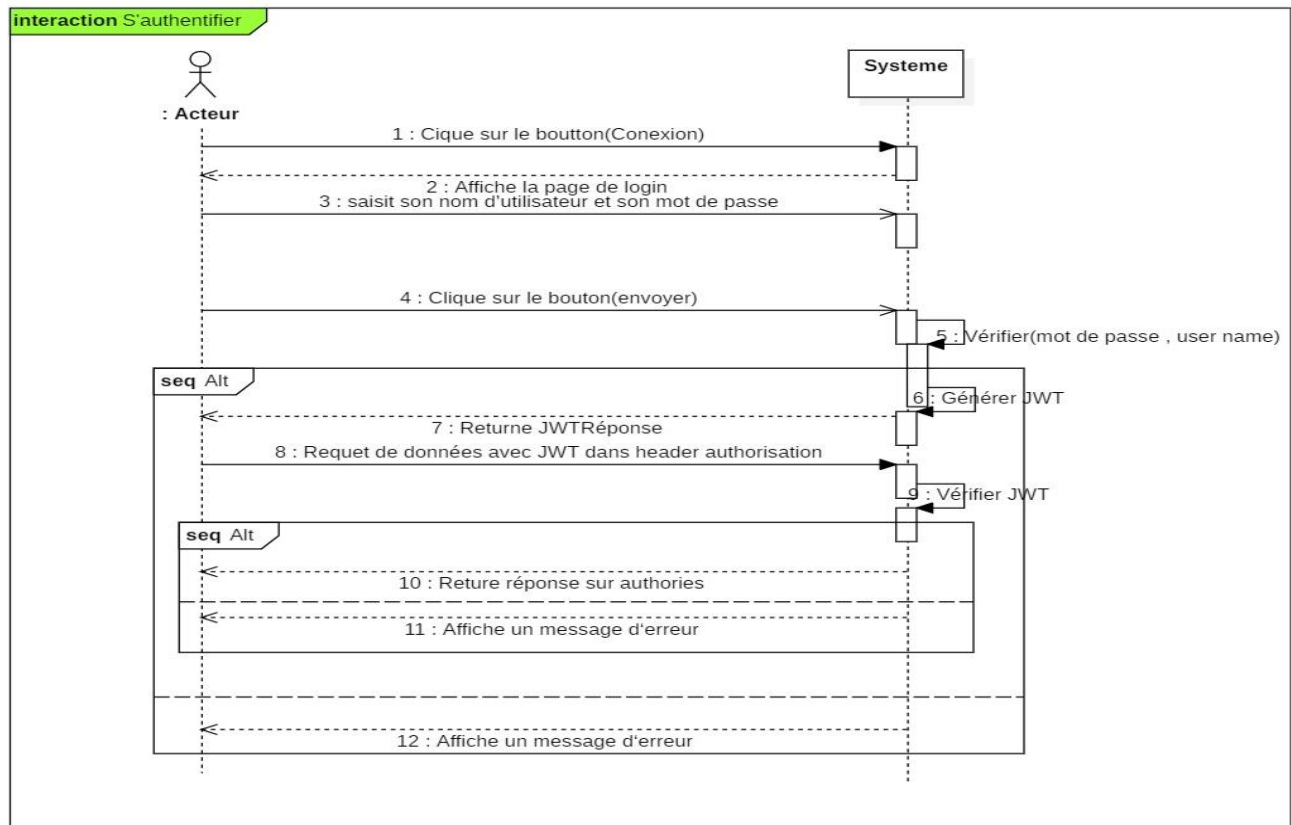


• S'authentifier

a) Diagramme de séquence

Cas d'utilisation :	S'authentifier.
Acteur :	Administrateur. Opérateur.
Résumé :	Ce cas d'utilisation permet à l'acteur de se connecter pour accéder à son espace.
Précondition :	L'utilisateur doit avoir un compte.
Post condition :	L'acteur ouverture de sa session privée.
Description du scénario nominal :	1. L'acteur demande d'accéder à son espace 2. Le système affiche l'interface de connexion. 3. L'acteur saisit les informations (user Name et mot de passe) et valide. 4. Le système vérifie l'existence de l'acteur. 5. Le système affiche l'espace de l'acteur.
Description du scénario alternatif :	5-a – le système renvoie un message d'erreur a l'acteur « il y'a pas de compte » On reprend la scénario du point 2.

b) Diagramme de séquence



➤ **Gestion des projets**

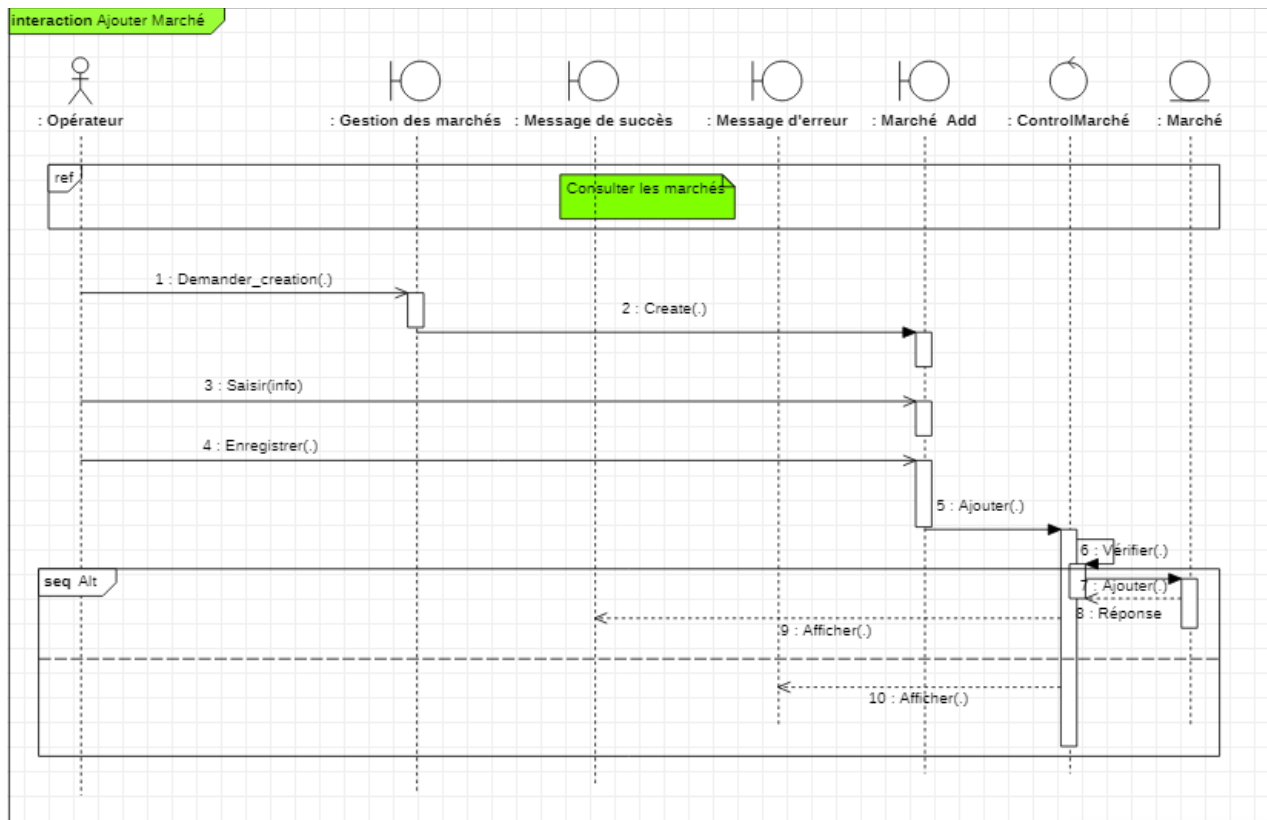
1. Marché

• **Ajouter marché**

a) **Description textuelle**

Cas d'utilisation :	Ajouter marché.
Acteur :	Opérateur .
Résumé :	Ce cas d'utilisation permet à l'acteur d'ajouter un marché pour un projet.
Précondition :	L'opérateur consulte les marchés.
Post condition :	Le marché est ajouté.
Description du scénario nominal :	1. L'opérateur demande la création de marché. 2. Le système affiche le formulaire d'ajoute. 3. L'opérateur saisit les informations de marché et enregistrer. 4. Le système lit et vérifie les paramètres. 5. Le système affiche un message d'information.
Description du scénario alternatif :	3-a - L'opérateur n'est pas rempli des champs obligatoires, le système indique au l'opérateur que le formulaire contient des champs vide ou incorrect et lui proposer de remplir. 3- b- verre à l'étape 3.

b) Diagramme de séquence

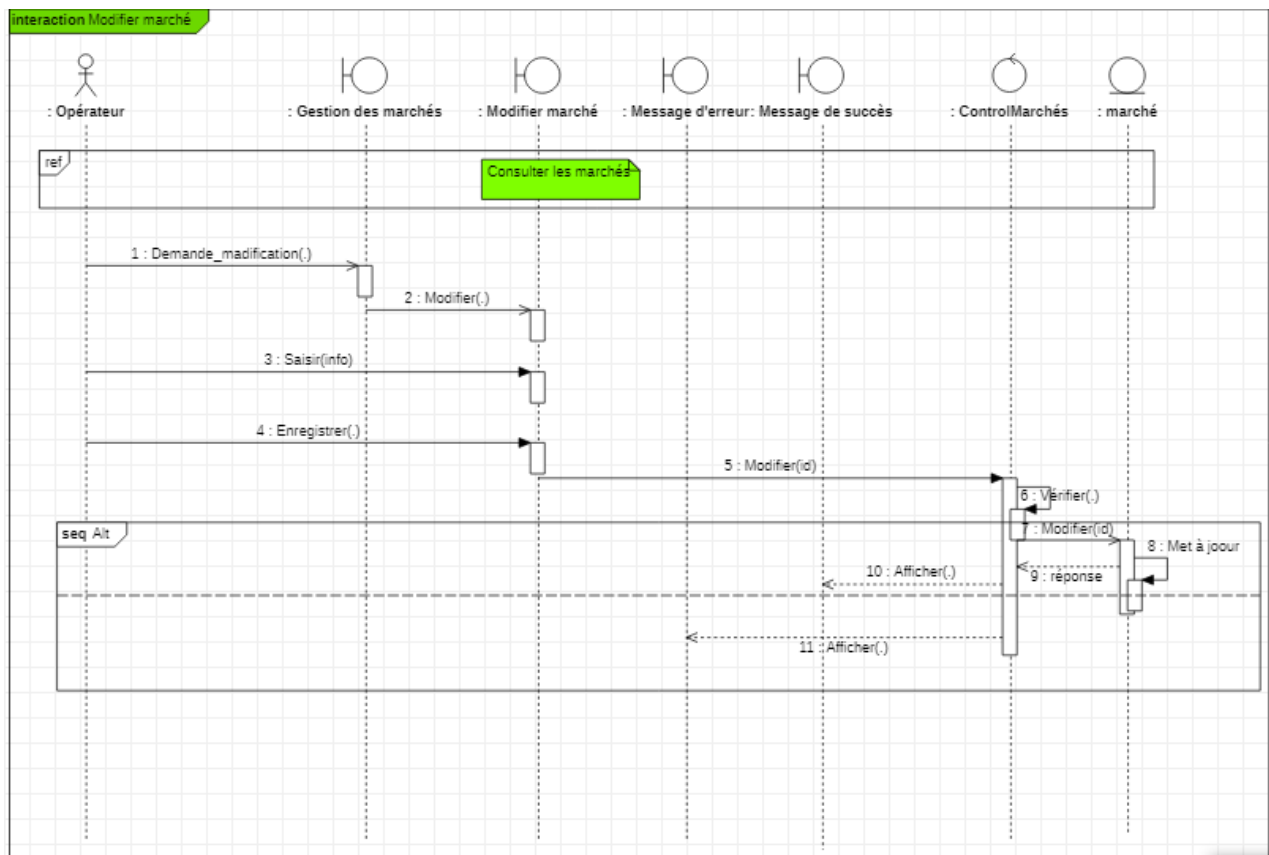


- **Modifier marché :**

a) Description textuelle

Cas d'utilisation :	Modifier marché.
Acteur :	Opérateur.
Résumé :	Ce cas d'utilisation permet à l'acteur de modifier marché .
Précondition :	L'opérateur consulte les marchés.
Post condition :	marché est modifiée.
Description du scénario nominal :	1. L'opérateur choisit un marché qu'il veut modifier. 2. Le système affiche les détails de marché. 3. L'opérateur saisit les modifications et enregistrer. 4. Le système vérifie les paramètres et effectue la modification de marché et affiche un message de d'information.

b) Diagramme de séquence

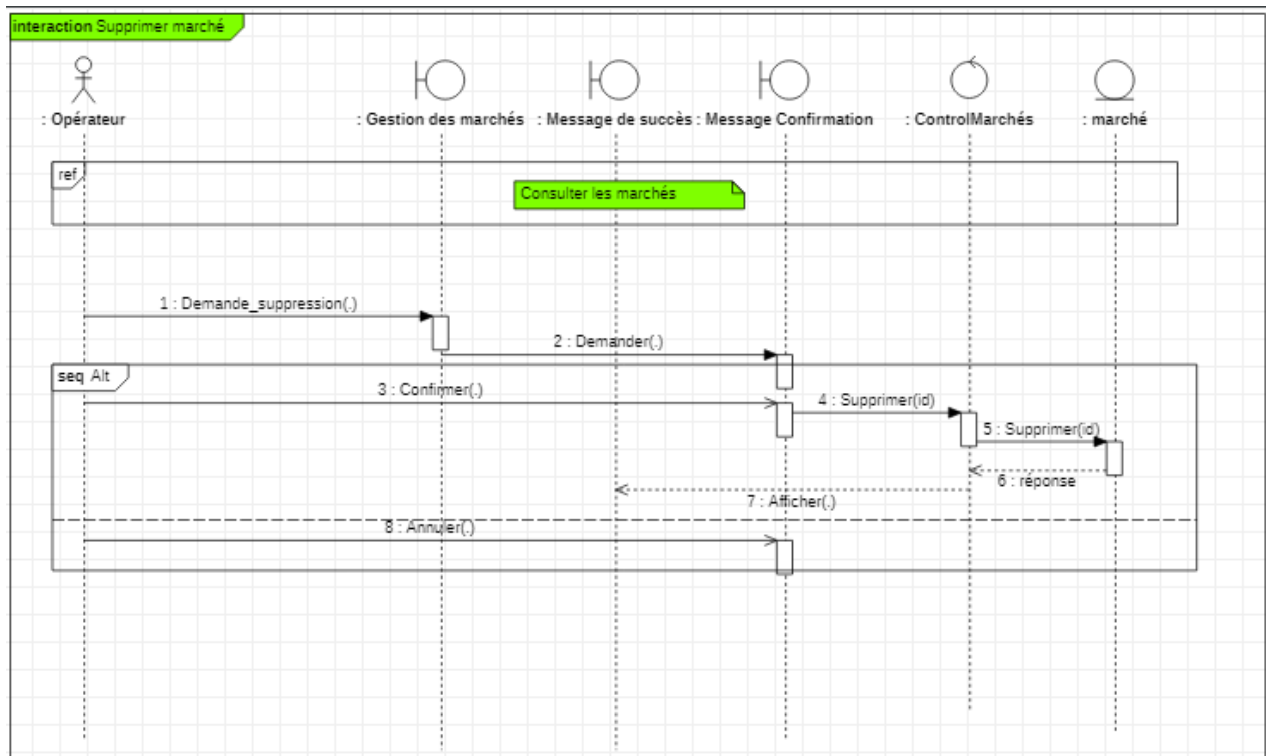


• Supprimer marché :

a) Description textuelle

Cas d'utilisation :	Supprimer marché.
Acteur :	Opérateur.
Résume :	Ce cas d'utilisation permet à l'acteur de supprimer un marché.
Précondition :	L'opérateur consulte les marchés.
Post condition :	marché est supprimée.
Description du scénario nominal :	1. L'opérateur choisit un marché et demande sa suppression. 2. Le système affiche un message de confirmation. 3. L'opérateur confirme la suppression de marché. 4. Le système effectue la suppression de marché et affiche un message d'information.

b) Diagramme de séquence

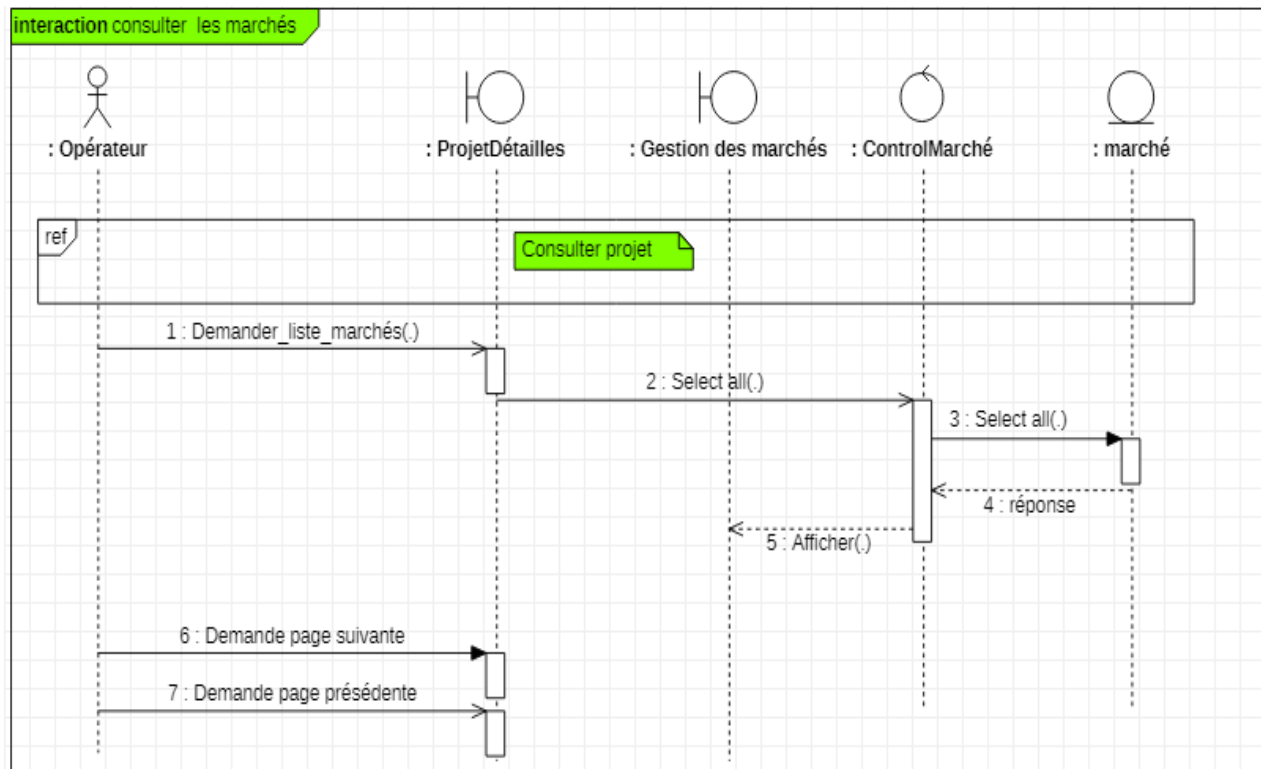


• Consulter les marchés :

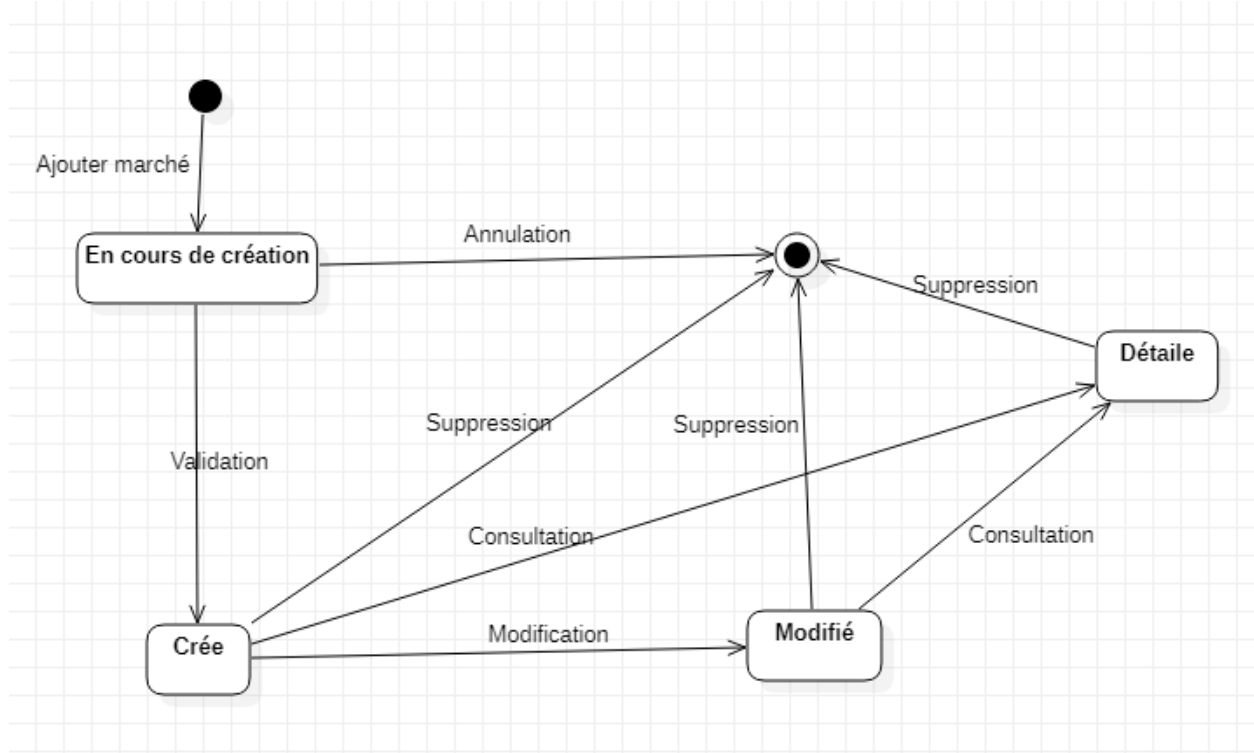
a) Description textuelle

Cas d'utilisation :	Consulter la liste des marchés.
Acteur :	Opérateur.
Résumé :	Ce cas d'utilisation permet à l'acteur de consulter la liste des marchés.
Précondition :	L'opérateur consulte projet.
Post condition :	La liste des marchés est affichée.
Description du scénario nominal :	1. L'opérateur demande la liste des marchés. 2. Le système affiche la liste des marchés enregistrées.

b) Diagramme de séquence



- Diagramme d'état transition : Marché



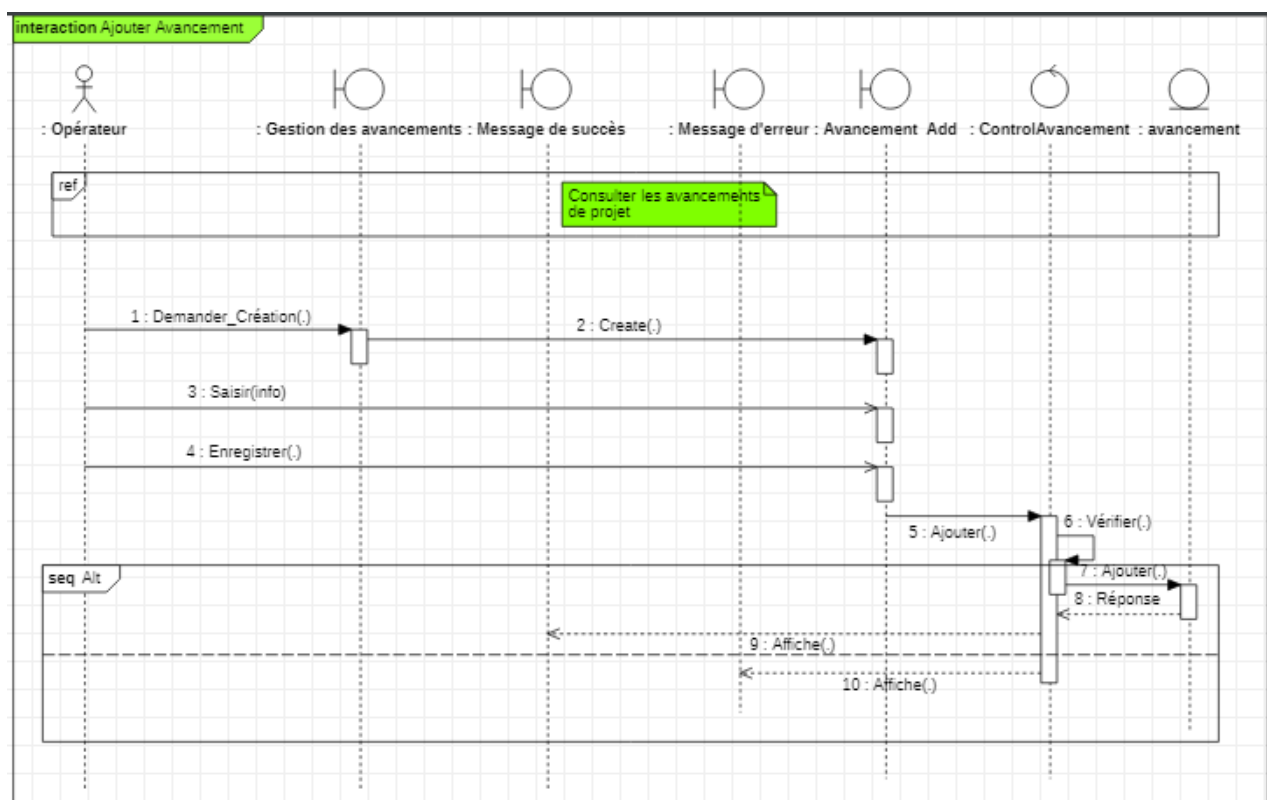
1. Avancement de projet

- Ajouter avancement de projet

a) Description textuelle

Cas d'utilisation :	Ajouter avancement de projet.
Acteur :	Opérateur .
Résumé :	Ce cas d'utilisation permet à l'acteur d'ajouter un avancement pour un projet.
Précondition :	L'opérateur consulte les avancements de projet.
Post condition :	L' avancement est ajouté.
Description du scénario nominal :	1. L'opérateur demande la création d'avancement. 2. Le système affiche le formulaire d'ajoute. 3. L'opérateur saisit les informations de avancement et enregistrer. 4. Le système lit et vérifie les paramètres. 5. Le système affiche un message d'information.
Description du scénario alternatif :	3-a - L'opérateur n'est pas remplit des champs obligatoires, le système indique au l'opérateur que le formulaire contient des champs vide ou incorrect et lui proposer de remplir. 3- b- verre à l'étape 3.

c) Diagramme de séquence



- Modifier avancement de projet

a) Description textuelle

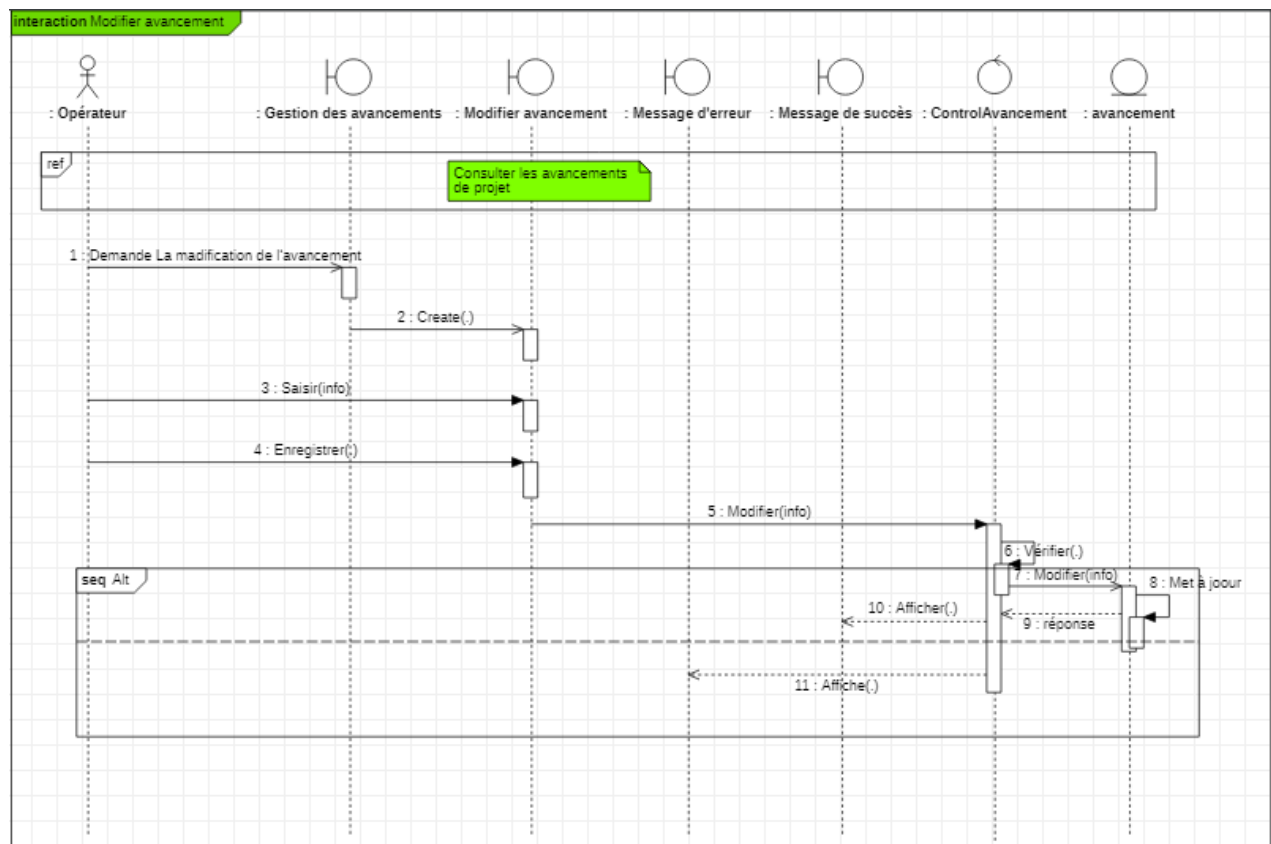
Cas d'utilisation :

Modifier avancement de projet.

Annexe

Acteur :	Opérateur.
Résumé :	Ce cas d'utilisation permet à l'acteur de modifier avancement.
Précondition :	L'opérateur consulte les avancements de projet.
Post condition :	avancement est modifiée.
Description du scénario nominal :	1. L'opérateur choisit un avancement qu'il veut modifier. 2. Le système affiche les détails de avancement. 3. L'opérateur saisit les modifications et enregistrer. 4. Le système vérifie les paramètres et effectue la modification de avancement et affiche un message de d'information.
Description du scenario alternatif :	3-a – le système affiche un message d'erreur.

b) Diagramme de séquence

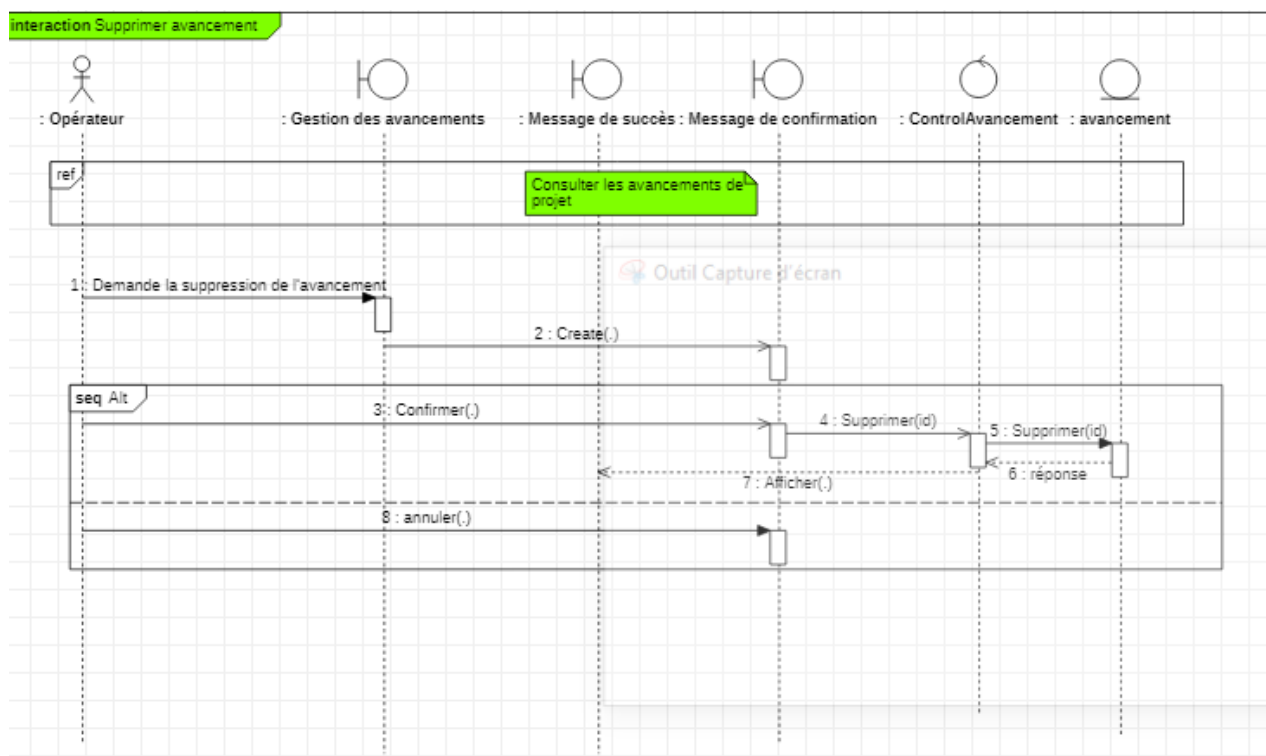


- Supprimer avancement de projet

a) Description textuelle

Cas d'utilisation :	Supprimer avancement de projet .
Acteur :	Opérateur.
Résumé :	Ce cas d'utilisation permet à l'acteur de supprimer un avancement.
Précondition :	L'opérateur consulte les avancements de projet.
Post condition :	avancement est supprimée.
Description du scénario nominal :	1. L'opérateur choisit un avancement et demande sa suppression. 2. Le système affiche un message de confirmation. 3. L'opérateur confirme la suppression d'avancement. 4. Le système effectue la suppression d'avancement et affiche un message d'information.
Description du scénario alternatif :	3-a - L'opérateur annuler la suppression de avancement.

b) Diagramme de séquence

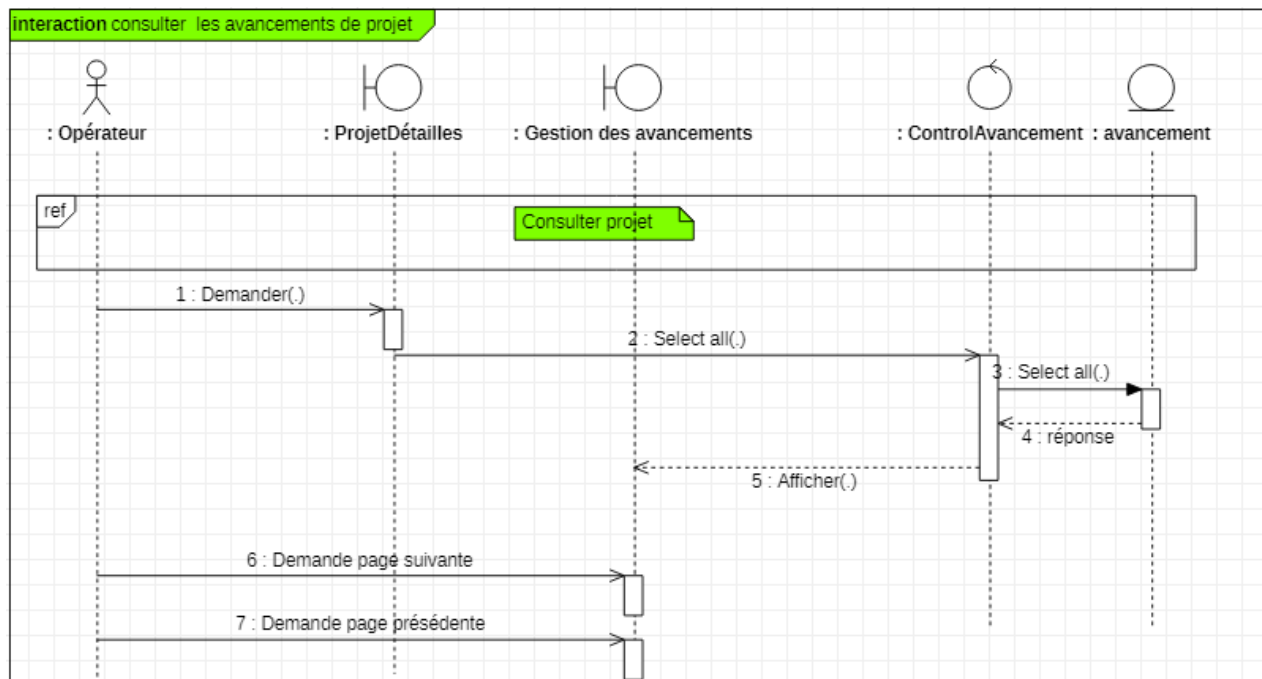


- Consulter les avancements de projet

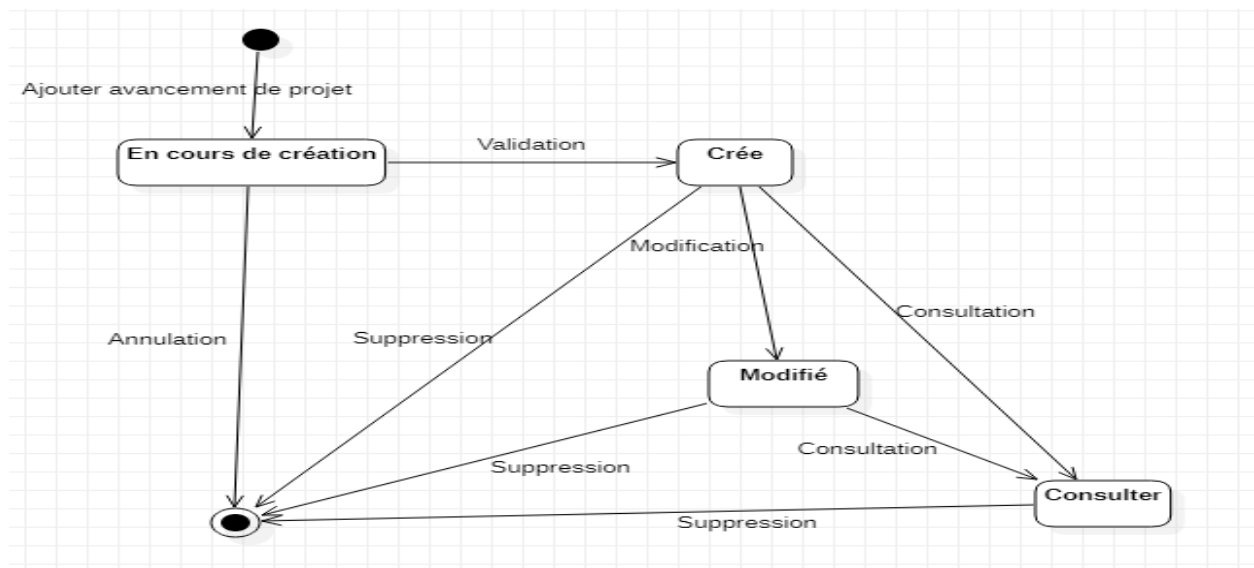
a) Description textuelle

Cas d'utilisation :	Consulter les avancements de projet.
Acteur :	Opérateur.
Résume :	Ce cas d'utilisation permet à l'acteur de consulter la liste des avancements.
Précondition :	L'opérateur consulte projet.
Post condition :	La liste des avancements est affichée.
Description du scénario nominal :	1. L'opérateur demande la liste des avancements. 2. Le système affiche la liste des avancements enregistrées.
Description du scénario alternatif :	2-a – le système affiche liste vide.

b) Diagramme de séquence



- **Diagramme d'état transition: Avancement de projet**



1. Lot

- **Ajouter lot**

a) Description textuelle

Cas d'utilisation :	Ajouter lot .
Acteur :	Opérateur .
Résumé :	Ce cas d'utilisation permet à l'acteur d'ajouter un lot pour un projet.
Précondition :	L'opérateur consulte les lots.
Post condition :	Le Lot est ajouté.
Description du scénario nominal :	1. L'opérateur demande la création de lot. 2. Le système affiche le formulaire d'ajoute. 3. L'opérateur saisit les informations de lot et enregistrer. 4. Le système lit et vérifie les paramètres. 5. Le système affiche un message d'information.

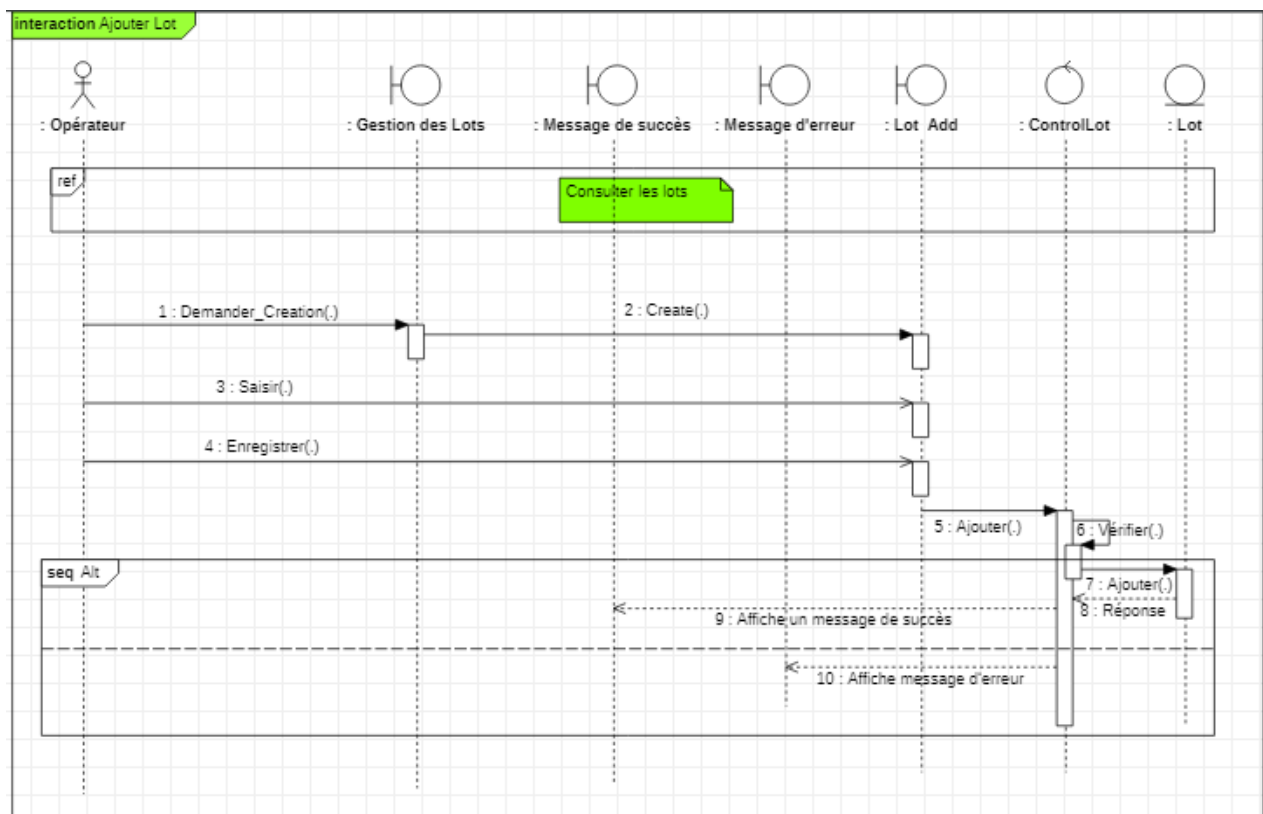
Annexe

Description du scenario alternatif :

3-a - L'opérateur n'est pas rempli des champs obligatoires, le système indique au l'opérateur que le formulaire contient des champs vide ou incorrect et lui proposer de remplir.

3- b- verre à l'étape 3.

b) Diagramme de séquence

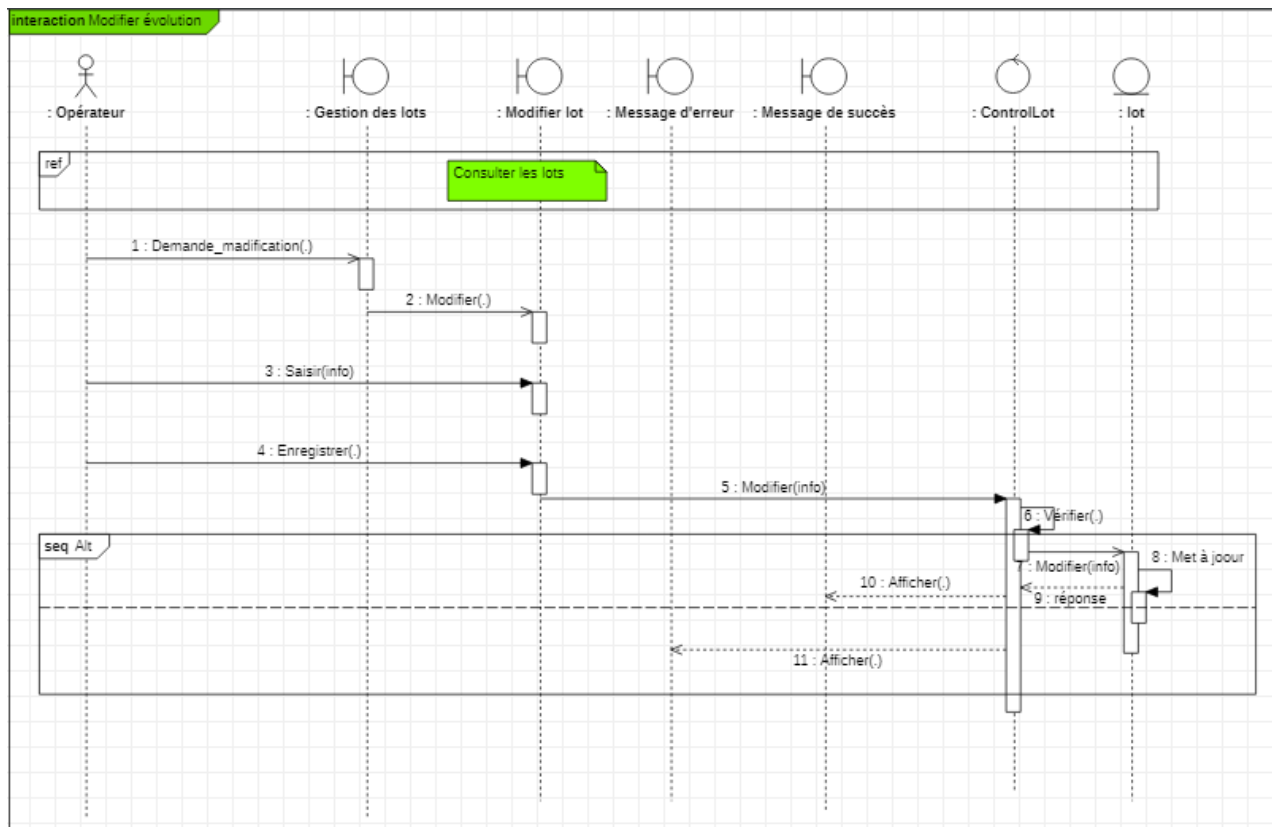


• Modifier lot

a) Description textuelle

Cas d'utilisation :	Modifier lot.
Acteur :	Opérateur.
Résume :	Ce cas d'utilisation permet à l'acteur de modifier Lot.
Précondition :	L'opérateur consulte les Lots.
Post condition :	Lot est modifiée.
Description du scénario nominal :	1. L'opérateur choisit une Lot qu'il veut modifier. 2. Le système affiche les détails de Lot. 3. L'opérateur saisit les modifications et enregistrer. 4. Le système vérifie les paramètres et effectue la modification de Lot et affiche un message de d'information.
Description du scénario alternatif :	3-a – le système affiche un message d'erreur.

b) Diagramme de séquence

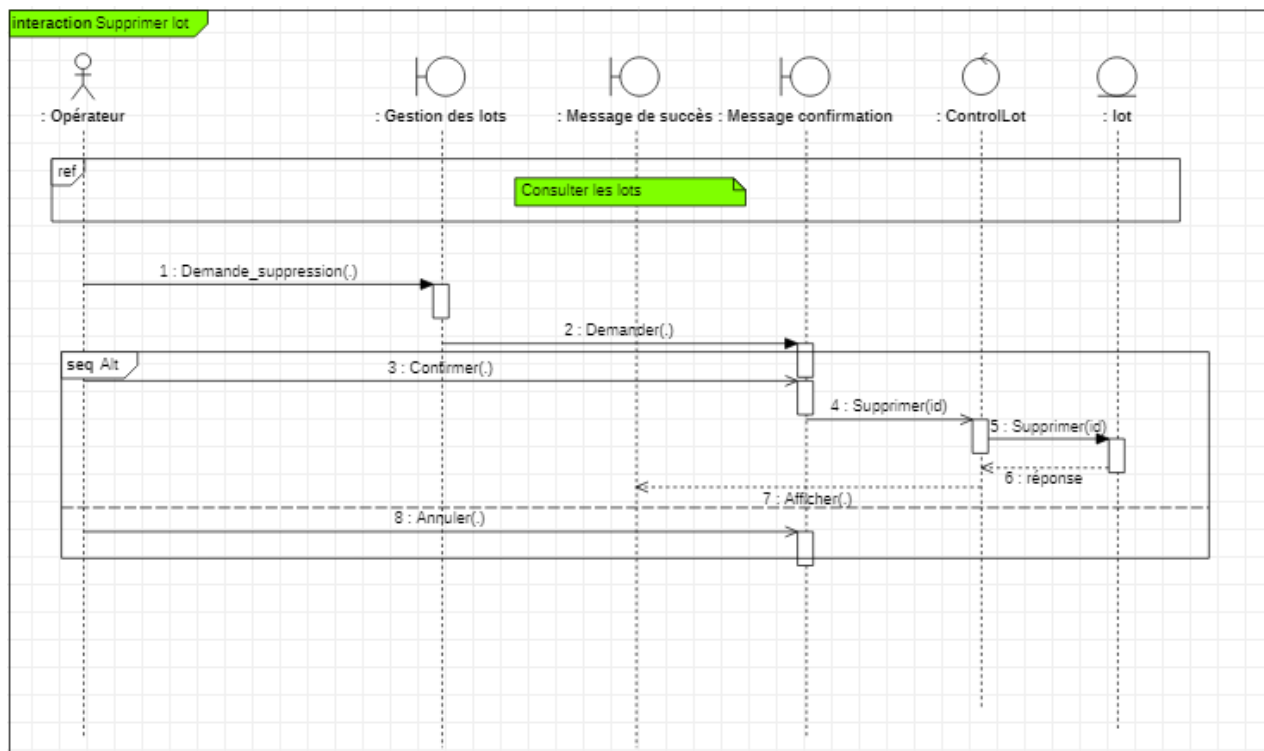


- **Supprimer lot**

a. Description textuelle

Cas d'utilisation :	Supprimer lot .
Acteur :	Opérateur.
Résumé :	Ce cas d'utilisation permet à l'acteur de supprimer un lot.
Précondition :	L'opérateur consulte les Lots.
Post condition :	L'évolution est supprimée.
Description du scénario nominal :	1. L'opérateur choisit un Lot et demande sa suppression. 2. Le système affiche un message de confirmation. 3. L'opérateur confirme la suppression de lot. 4. Le système effectue la suppression de lot et affiche un message d'information.
Description du scénario alternatif :	3-a - L'opérateur annuler la suppression de lot.

b) Diagramme de séquence

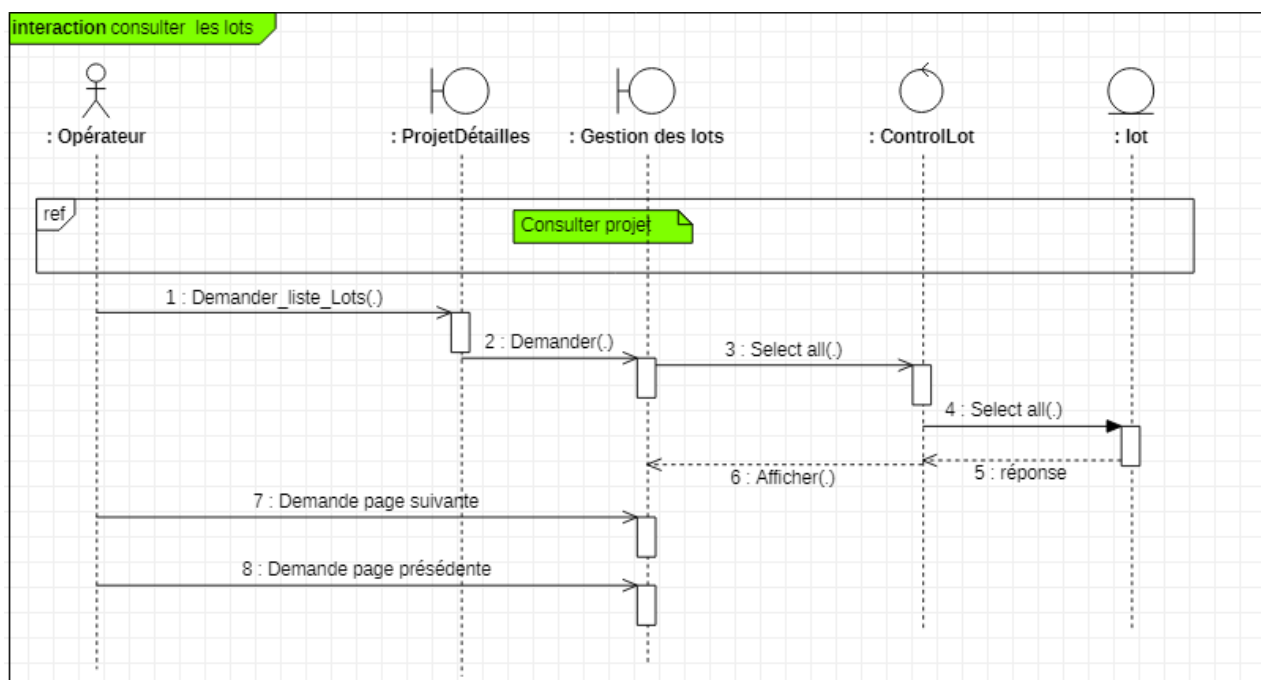


- Consulter les lots

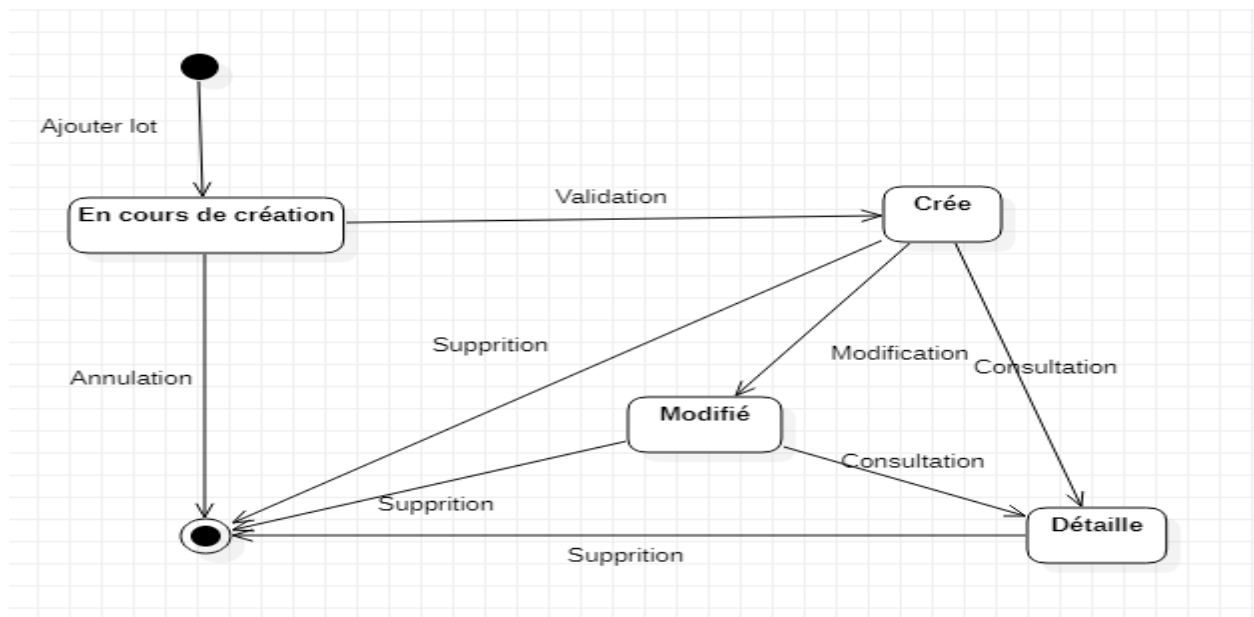
a) Description textuelle

Cas d'utilisation :	Consulter les lots.
Acteur :	Opérateur.
Résumé :	Ce cas d'utilisation permet à l'acteur de consulter la liste des lots.
Précondition :	L'opérateur est consulter projet.
Post condition :	La liste des lots est affichée.
Description du scénario nominal :	1. L'opérateur demande la liste des lots. 2. Le système affiche la liste des évolutions enregistrées.
Description du scénario alternatif :	2-a – le système affiche liste vide.

b) Diagramme de séquence



- Diagramme d'état transition : lot



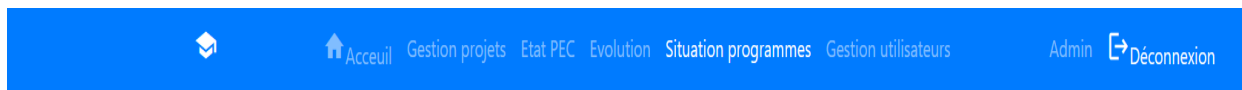
Annexe 4: Les interfaces du système

2. Interface de gestion des utilisateurs

Liste des Utilisateurs

Numero	Username	Nom	Prénom		
3	aya	sekfali	Aya	ROLE_USER	
2	ghozlane	boucheloukh	ghozlane	ROLE_USER	
1	Admin			ROLE_USER ROLE_ADMIN	

3. Interface de situation des programmes



+ Nouvelle Situation

Situation des programmes d'investissement par période

Période	Etat	Nbp Université	Nbp pédagogique	Nbp d'hébergement	Nbp Bibliothèque	Nbp Restaurant	Nbp Lab	Nbp Auditorium	Nbp Rectorat	Nbp Logement	Actions
2021	lancé	2	0	0	1	0	0	0	0	0	 
1999	En cours	0	0	0	0	0	6	1	0	2	 

« Prev 1 Next »

Bibliographie

[1]: [En ligne]. Available: http://www.centre-univ-mila.dz/?page_id=29.

[2]: [En ligne]. Available: https://fr.wikipedia.org/wiki/Wilaya_de_Mila.

[3]: Makhmoukh, S. Encadreur, Ali Maameri, and Larbi Lanseur. Les programmes sectoriels deconcentrés levier du développement local. Diss. Université de bejaia, 2016.

[4]: M.Babacar DIAGO ,2022. Automatisation du système de restauration de l'université Assane Seck de Ziguinchor, de la vente des tickets au contrôle des accès aux restaurants, mémoire de fin d'étude Master (Génie Logiciel), Université Assane Seck de Ziguinchor.

[5]: Architecture microservices. (s. d.). [linkedin.com](https://www.linkedin.com).

[6] : Brighet Nabil et Delimi Youcef ,2022. Mise en place d'une gestion sécurisée de ressources humaines et production scientifique dans un laboratoire de recherche à base de Microservices. Mémoire de Master pour obtenir le diplôme de Master en Informatique (Sciences et technologies de l'information et de la communication (STIC)). Université Abdelhamid Mehri – Constantine 2.

[7]: Microservices architecture | Characteristics, uses, and example. (s. d.). Study Abroad Blogs | All about universities, programs, tests, & ; more ! <https://ischoolconnect.com/blog/microservices-architecture-characteristics-uses-and-example>.

[8] : Saini, R. (2021, 7 janvier). How to Create Microservices Using Spring - DZone. [dzone.com. https://dzone.com/articles/how-to-create-microservices-using-spring](https://dzone.com/articles/how-to-create-microservices-using-spring)

[9]: Introduction to Spring Security and its Features - GeeksforGeeks. (s. d.). GeeksforGeeks. <https://www.geeksforgeeks.org/introduction-to-spring-security-and-its-features>

[10]: Pascal Roques; Franck Vallée édition Eyrolles; UML 2 en action: De l'analyse des besoins à la conception.

Bibliographie

[11]: Boumeshal Yahya et Meskine Mourad ,2022. Conception et Réalisation d'un Système d'Information Sécurisé basé sur la Blockchain, Mémoire préparé en vue de l'obtention du diplôme de Master, Option Informatique, Sciences et Technologies de l'Information et de la Communication (STIC), Centre Universitaire Abdellhafid Boussouf - Mila.

[12]: P. Roques, Les Cahiers du programmeur UML : Modéliser une application web, 4e éd. Paris : Eyrolles, 2002.

[13] : LEMEURS Abdelbasst et DALI Abderrachid,2022. Portail numérique de l'étudiant Université de MILA, Mémoire préparé en vue de l'obtention du diplôme de Master, Option Informatique, Sciences et Technologies de l'Information et de la Communication (STIC), Centre Universitaire Abdellhafid Boussouf - Mila.

[14]: *Spring Boot*. (s. d.). Spring Boot. <https://projects.spring.io/spring-boot/>.

[15]: Introduction à Spring Cloud | devstory.net. (s. d.). devstory. <https://devstory.net/11721/introduction-a-spring-cloud#a15351540>

[16]: « Java : définition ». JDN : E-business, FinTech, Big Data, IoT, tendances média, décideurs... <https://www.journaldunet.fr/web-tech/dictionnaire-du-webmastering/1203555-java-definition/>.

[17]: « Maven â Welcome to Apache Maven ». Maven â“ Welcome to Apache Maven. <https://maven.apache.org/>.

[18]: « Postman : comment utiliser la plateforme d'API no code ». JDN : E-business, FinTech, Big Data, IoT, tendances média, décideurs... <https://www.journaldunet.fr/web-tech/guide-de-l-entreprise-digitale/1511313-postman-comment-utiliser-la-plateforme-d-api-no-code/>.

[19]: Contributeurs aux projets Wikimedia. « Angular — Wikipédia ». Wikipédia, l'encyclopédie libre. <https://fr.wikipedia.org/wiki/Angular>.

[20]: « StarUML - air ». air. <https://air.imag.fr/index.php/StarUML>.

[21]: « Visual Paradigm - UML, Agile, PMBOK, TOGAF, BPMN and More ! » Ideal Modeling & ; Diagramming Tool for Agile Team Collaboration. <https://www.visual-paradigm.com/features/>.

[22]: « Techopedia : Educating IT Professionals To Make Smarter Decisions ». Techopedia. <https://www.techopedia.com/>.

[23]: « Node.js ». Node.js. <https://nodejs.org/en/>.

[24]: « npm Docs ». Npm Docs. <https://docs.npmjs.com/> .

[25]: « PostgreSQL ». PostgreSQL. <https://www.postgresql.org/> .

Acronymes

St : science et technologie

MI : math informatique

Snv : science de nature et vie

Stu : science de terre et l'universel

MVC : model view controller