

À MA FAMILLE

Remerciements

Ce mémoire est l'aboutissement d'une somme d'efforts que nous ne saurions passer sous silence, nous pensons notamment à :

- Notre superviseur **Dr ENGA Vincent** pour sa contribution à l'amélioration de la qualité de ce mémoire,
- Notre encadreur académique **M. TCHOUTA Alain Serge** pour sa disponibilité et ses conseils pendant le travail,
- Notre encadreur professionnel **M. KITIO Christian** pour son expertise technique et le suivi au cours du projet,
- Au Président Fondateur **M. GUIMEZAP Paul** pour l'accueil dans son établissement scolaire,
- Au **Dr.FOUEKENG**, pour l'expertise qu'il nous a apportée pour une meilleure compréhension de notre sujet.
- Au Directeur Général de Monglo Technology, **M. MONGLO Germain** pour sa générosité qui a permis que l'on soit accueilli au sein de son entreprise,
- Aux enseignants de l'IUC pour les connaissances apportées et leurs conseils,
- Aux amis qui n'ont cessé de m'encourager et de me motiver,

Sommaire

Remerciements	ii
Sommaire	iii
Table des figures	iv
Liste des tableaux	vi
Introduction	1
I ÉTAT DE L'ART	2
1 PRÉSENTATION DU PROJET	3
2 GÉNÉRALITÉS SUR LES OUTILS D'APPRENTISSAGE IMMERSIF BASÉ SUR LA RÉALITÉ VIRTUELLE	16
II ÉTUDE DÉTAILLÉE ET RÉALISATION	22
3 Cadrage, analyse et conception	23
4 Implémentation de la solution et résultats	57
Bibliographie	80
Table des matières	82

Table des figures

1.1	Simulateur de vol ONTOP DUO MEAC	6
1.2	Simulateur de vol Microsoft flight simulator	6
1.3	Simulateur chirurgical Osso VR	7
1.4	Simulateur scientifique Praxilab	7
1.5	Simulateur scientifique EON RealityC	8
2.1	L'oculus rift (gauche), le HTC Vive (centre) et le PlayStation VR (droite)	19
2.2	L'oculus Half Moon (gauche), le reactive Grip (centre) et le SteamVR (droite)	19
2.3	Le Virtuix Omni (gauche), le WalkMouse (centre), et le InfinaDeck (droite)	19
2.4	Le STEM (gauche), le PrioVR (centre), et le ControllVR (droite)	20
2.5	Le Leap Motion (gauche), le Myo (centre), et le Glove One (droite)	20
3.1	Diagramme de gantt du projet	30
3.2	Diagramme des cas d'utilisation	34
3.3	Diagramme des cas d'utilisation d'un administrateur	35
3.4	Diagramme des cas d'utilisation d'un enseignant	36
3.5	Diagramme des cas d'utilisation d'un apprenant	36
3.6	Diagramme de séquence du cas d'utilisation s'authentifier	40
3.7	Diagramme de séquence du cas d'utilisation création d'élément chimique	41
3.8	Diagramme de séquence du cas d'utilisation effectuer une réaction	42
3.9	Diagramme d'activité du cas d'utilisation créer une réaction	43
3.10	Diagramme d'activité du cas d'utilisation suppression d'une réaction	44
3.11	Diagramme d'état transition du cas d'utilisation changement d'état des produits d'une réaction	45
3.12	Diagramme d'état transition du cas d'utilisation changement d'état d'authentification d'un utilisateur	46
3.13	Diagramme de classe du système	48
3.14	Diagramme de déploiement du système	49
3.15	IHM de la page d'accueil	50
3.16	IHM de la page d'inscription des enseignants	51
3.17	IHM de la page de connexion des enseignants et des administrateurs	52
3.18	IHM de la page espace personnel des enseignants et des administrateurs	53
3.19	IHM du laboratoire en vue de dessus	54
3.20	IHM du laboratoire en vue de face	55
3.21	IHM du laboratoire intérieur vue du fond	55
3.22	IHM du laboratoire intérieur vue de l'apprenant	56
3.23	IHM du laboratoire intérieur vue de l'enseignant	56

4.1	IHM formulaire d'édition du profil utilisateur	59
4.2	Code backend du modèle d'utilisateur	60
4.3	Code backend du cas d'utilisation de l'édition du profil utilisateur	61
4.4	Code backend du test unitaire du cas d'utilisation de l'édition du profil utilisateur	61
4.5	Résultat du test unitaire du cas d'utilisation de l'édition du profil utilisateur	62
4.6	Code frontend du cas d'utilisation de l'édition du profil utilisateur	62
4.7	IHM formulaire de création des éléments chimiques	63
4.8	Code backend du modèle d'élément chimique	64
4.9	Code backend du cas d'utilisation de la création d'élément chimique	65
4.10	Code backend du test unitaire du cas d'utilisation de la création d'élément chimique	66
4.11	Résultat du test unitaire du cas d'utilisation de la création d'élément chimique	66
4.12	Code frontend du cas d'utilisation de la création d'élément chimique	67
4.13	IHM du listing des éléments chimiques	68
4.14	Code backend du contrôleur de listing des éléments chimiques	68
4.15	Code backend du test unitaire du cas d'utilisation de la listing des éléments chimiques	69
4.16	Résultat du test unitaire de la fonctionnalité de listing des éléments chimiques	69
4.17	Code frontend du cas d'utilisation du listing des éléments chimiques	70
4.18	IHM formulaire de création des réactions chimiques	71
4.19	Code backend du modèle de réaction	72
4.20	Code backend du contrôleur de création des réactions chimiques	73
4.21	Code backend du test unitaire du cas d'utilisation de la création d'une réaction chimique	74
4.22	Résultat du test unitaire de la fonctionnalité de création d'une réaction chimique	74
4.23	Code frontend du cas d'utilisation de la création d'une réaction chimique	75
4.24	IHM du listing des réactions chimiques	75
4.25	Code backend du contrôleur de listing des réactions chimiques	76
4.26	Code backend du test unitaire du cas d'utilisation du listing des réactions chimiques	76
4.27	Résultat du test unitaire de la fonctionnalité de listing des réactions chimiques	77
4.28	Code frontend du cas d'utilisation de listing des réactions chimiques	78

Liste des tableaux

1.1	Comparatif des solutions existantes	8
1.2	Description des solutions les plus proches de la notre	9
1.3	Comparaison entre les approches traditionnelles et agiles	11
1.4	Comparaison des méthodes agiles	12
3.1	types de projet COCOMO	26
3.2	Estimation des coûts	27
3.3	Estimation des coûts matériels	27
3.4	Estimation des coûts matériels	29
3.5	Moyens matériels	31
3.6	Le management du projet	31
3.7	Les acteurs du système et leurs cas d'utilisation	33

Résumé

Les TIC (Technologie de l'Information et de la communication) sont en constante évolution et proposent toujours de nouvelles façons plus simples et rapides d'effectuer des tâches par le biais de technologie innovante. C'est le cas de la réalité virtuelle qui permet la simulation d'environnement 3D divers afin d'apporter une expérience d'immersion aux utilisateurs. De nouvelles applications de cette technologie dans de nombreux domaines voient le jour, notamment dans l'éducation où ils permettent un enseignement immersif et sûr. L'objectif de ce projet est de créer un environnement d'apprentissage immersif de la chimie tout en limitant les risques liés à sa pratique pour des novices. D'où la problématique de savoir, Comment l'utilisation des technologies informatiques pourraient contribuer à faciliter l'enseignement de la chimie dans notre système éducatif et plus précisément dans l'enseignement secondaire Dans ce contexte, la facilitation de l'enseignement passe par une limitation non seulement des coûts liés à l'enseignement de la chimie, mais aussi des risques liés à son apprentissage par des personnes inexpérimentées. Pour répondre à cette problématique, nous avons d'une part recensé des outils existants permettant un enseignement simulé et immersif et d'autre part, nous avons défini notre méthodologie de travail grâce à une étude comparative entre plusieurs méthodes de gestion de projet (scrum, eXtreme Programing,...). Après cette étude comparative, notre choix s'est porté sur Scrum dû au type de projet (projet à grand risque de changement), à l'équipe du projet (très petite) et pour l'aspect itératif qu'il apporte à la gestion de projet. Puis une analyse a été effectuée sur la base de UML. Pour l'implémentation en terme de technologie, nous avons choisi unreal engine 5 pour le développement de l'environnement virtuel en 3D, react et react dom pour l'implémentation de l'application web, PostgreSQL comme système de gestion des bases de données et enfin Asp.net pour l'implémentation du backend. À partir de ces choix, nous avons abouti à la conception d'une plateforme qui répond aux spécifications demandées, une application web pour la gestion des réactions et une en 3D pour la pratique de ces réactions.

Mot clé — Réalité virtuelle, 3D, éducation, immersion, chimie, web

Abstract

ICT (Information and Communication Technology) is constantly evolving and always offers new, simpler and faster ways of carrying out tasks through innovative technology. This is the case of virtual reality which allows the simulation of various 3D environments in order to provide users with an immersive experience. New applications of this technology in many fields are emerging, especially in education where they allow immersive and safe teaching. The objective of this project is to create an immersive chemistry learning environment while limiting the risks associated with its practice for novices. Hence the problem of knowing, How the use of computer technologies could help facilitate the teaching of chemistry in our education system and more specifically in secondary educationIn this context, the facilitation of teaching requires a limitation not only of the costs linked to the teaching of chemistry, but also of the risks linked to its learning by inexperienced people. To answer this problem, we have on the one hand identified existing tools allowing a simulated and immersive teaching and on the other hand, we have defined our working methodology thanks to a comparative study between several project management methods (scrum, eXtreme Programming, etc.). After this comparative study, our choice fell on Scrum due to the type of project (project with a high risk of change), the project team (very small) and for the iterative aspect it brings to management. of project. Then an analysis was carried out on the basis of UML. For the implementation in terms of technology, we chose unreal engine 5 for the development of the virtual environment in 3D, react and react dom for the implementation of the web application, PostgreSQL as the database management system and finally Asp.net for the backend implementation. From these choices, we have come up with the design of a platform that meets the requested specifications, a web application for managing reactions and a 3D one for practicing these reactions.

Keyword — Virtual reality, 3D, education, immersion, chemistry, web

Introduction

Avec l'évolution constante des technologies de l'information et de la communication, de nombreux domaines de la vie courante ont changé ou sont en train de changer. Ces technologies offrent de nouvelles possibilités et permettent une nette évolution dans ces différents secteurs. Le domaine éducatif bien qu'ancien lui aussi se retrouve touché par cette évolution notamment à cause de la pandémie de la covid-19 qui a touché le monde obligeant son adaptation aux circonstances. Ainsi, l'enseignement a été délocalisé de nos salles de classe au réseau internet pour un respect des règles de distanciation sociale. De nouvelles technologies comme la réalité virtuelle ou la réalité augmentée pourraient à leur tour bouleverser le domaine éducatif en introduisant une nouvelle façon d'apprendre (en s'immergeant dans un environnement). En effet, depuis 2012 avec l'arrivée des casques oculus, la réalité virtuelle et la réalité augmentée ont subi de grandes évolutions permettant désormais de créer des environnements en trois dimensions à moindre coût et des façons réalistes ces technologies offrent l'avantage d'être virtuel ainsi détachées du monde réel et des risques qui lui sont propres. Des domaines d'enseignements où ces risques-là sont très présents pourraient bénéficier de cette technologie pour leur enseignement aux novices qui pourraient commettre des erreurs causant des accidents graves, voire mortels. La chimie est l'un des domaines où l'expérimentation dans le monde réel peut avoir de graves conséquences sur l'apprenant ou son environnement du fait de son inexpérience et de la nature des éléments manipulés qui pourraient entraîner des accidents graves.

La réalité virtuelle présente également l'avantage d'être de plus en plus abordable en matière de prix et pourrait permettre aux établissements d'enseignement de faire des économies que ce soit en matière de matériel ou de la main-d'œuvre. Dans le cas de la chimie, ces économies pourront être faites sur le local, les équipements, les éléments chimiques et la main-d'œuvre car tous ces aspects seront simulés. Dans l'optique d'apporter une solution au problème de savoir « Comment l'utilisation des technologies informatiques pourraient contribuer à faciliter l'enseignement de la chimie dans notre système éducatif et plus précisément dans l'enseignement secondaire », nous avons opté pour la réalisation d'une plateforme et pour ce faire, nous avons pour thème « Analyse, conception et réalisation d'un module d'apprentissage immersif basé sur la réalité virtuelle : cas des travaux pratiques de la chimie ». Le présent document rend compte de tout ce qui a été réalisé durant ce projet. Il s'articule autour de quatre chapitres répartis en deux parties.

La première partie porte sur l'état de l'art, divisée en deux chapitres, notamment la présentation du projet et la généralité sur les outils d'apprentissage immersif basé sur la réalité virtuelle. La deuxième partie intitulée étude et réalisation comporte deux chapitres, le premier étant l'analyse et la conception, le dernier la réalisation et les résultats.

Première partie

ÉTAT DE L'ART

Chapitre 1

PRÉSENTATION DU PROJET

Dans ce premier chapitre de la première partie de notre mémoire, nous allons présenter notre projet, du contexte à la problématique, sans bien sûr oublier les méthodes. Il présente notre projet dans son ensemble, il présente la délimitation du sujet, les problèmes à résoudre, l'intérêt du sujet et une étude de l'existant.

1.1 Compréhension du sujet

1.1.1 Contexte

L'apprentissage est un ensemble de mécanismes menant à l'acquisition de savoir-faire, de savoirs ou de connaissances afin de s'en servir dans la vie courante ou de les faire évoluer pour les générations futures. Malgré les évolutions technologiques offrant de nouvelles façons d'acquérir des connaissances, le système éducatif africain, plus précisément camerounais n'a pas beaucoup évolué, les moyens de dispenser les connaissances ne suivent pas toujours les tendances actuelles. Les personnes apprennent mieux en s'immergent dans ce qu'ils font qu'en le théorisant, cela est toujours possible dans le processus d'apprentissage, mais pas dans tous les domaines de formation du fait des risques lié à l'acquisition des connaissances en pratique.

Parmi ces domaines dans lesquels l'immersion réelle des apprenants dans les conditions réelles de pratique présente des risques réels pour l'apprenant, nous pouvons parler de la chimie, qui est un domaine de la science très expérimentale nécessitant des observations pour une compréhension du sujet étudié en vue d'y apporter des applications dans la vie courante. Une grande et bonne compréhension de ce domaine pourrait apporter de nombreuses idées de recherche qui permettront des avancées significatives dans de nombreux domaines (agriculture, industrie, mode, la mécanique, l'énergie...), avancées qui pourraient à leur tour faciliter le processus d'émergence en Afrique et plus précisément au Cameroun. Malgré l'aspect très expérimental de son apprentissage, il reste assez dangereux et couteux à enseigner en pratique, dangereux, car le manque d'expérience des apprenants pourrait les pousser à commettre des erreurs qui pourraient causer des accidents très dangereux, voire mortelle en fonction des éléments manipulés et couteux, car la maintenance du matériel nécessaire aux expérimentations (local, verrerie, éléments et personnelles de maintenance) engendre des coûts assez élevés.

VREDU est une plateforme dont l'objectif est de limiter les coûts et les risques lors des expérimentations en concevant le réalisme afin de créer un sentiment d'immersion pour une meilleure compréhension. Pour ce faire, monglo technology s'est intéressé à la réalité virtuelle qui est une expression désignant les dispositifs permettant de simuler numériquement un environnement par la machine (ordinateur), afin d'apporter ce réalisme et limiter les risques d'accidents.

1.1.2 Délimitation du sujet et hypothèse du travail

Notre travail se limitera au cas des travaux pratiques de la chimie.

- Création des réactions chimiques
- Expérimentation des réactions dans un environnement immersif

1.2 Étude de l'existant

L'étude de l'existant a pour but d'approfondir l'analyse des axes innovants d'un projet au cours d'élaboration, et avant sa mise en œuvre. Cette étude préalable sert à donner un aperçu sur la pertinence du projet, sa faisabilité ainsi que sa continuité.

1.2.1 Description de l'existant

Nous ne saurions commencer ce travail sans avoir une idée claire et précise sur l'existant quel qu'il soit. Dans la plupart des lycées, l'enseignement de la chimie suit un modèle traditionnel, à savoir cours théoriques en salle de classe au cours duquel l'apprenant découvre les principes théoriques nécessaires à la maîtrise de cette science. Ensuite, un cours pratique en laboratoire au cours duquel les apprenants发现 la réalité des réactions grâce aux expériences scientifiques.

1.2.2 Critique de l'existant et Problématique

Au cours de notre étude de l'existant, nous avons pu ressortir de nombreux problèmes liés à la réalisation des réactions dans un environnement réel, parmi lesquelles :

- Les risques d'accidents au cours d'expérimentation trop élevée dans un environnement réel, accident qui peut s'avérer mortel suivant les éléments manipulés.
- Les coûts de maintenance des équipements qui avec le temps se détériorent et nécessitent d'être renouvelé ou entretenus régulièrement ce qui entraîne des couts matériels et humains assez importants.

Ceci nous a poussé à soulever la problématique : « **Comment l'utilisation des technologies informatiques pourraient contribuer à faciliter l'enseignement de la chimie dans notre système éducatif et plus précisément dans l'enseignement secondaire ?** » Autrement dit, pourrait-on envisager une plateforme permettant la simulation d'un laboratoire de chimie en limitant les risques d'accidents au cours des expérimentations et aussi en limitant les coûts de maintenance du matériel une fois l'environnement fonctionnel ?

1.2.3 Quelques solutions existantes

Il est question ici de noter les points forts de ces dernières et leurs points faibles afin d'ajuster nos objectifs.

— ONTOP DUO MEAC

Ontop duo meac est une technique de simulation de vol utilisé dans les écoles d'aviation utilisé par les formateurs pour mettre en pratique les aspects théoriques abordés durant les formations. En effet, dans le domaine de l'aviation, les simulateurs de vol comme celui-ci sont presque incontournables, car en cas d'accident en condition réel dû au manque d'expérience des apprenants, les nombreuses vies seront menacées et des équipements très coûteux seront détruits.

Le dispositif est composé d'écrans disposés à 180° autour de l'apprenant afin de simuler le point de vue d'un pilote et d'un cockpit fidèle à leur homologue réel.



FIGURE 1.1 – Simulateur de vol ONTOP DUO MEAC

— Microsoft flight simulator

Flight Simulator est un logiciel de simulation de vol pour Microsoft Windows, vendu et souvent vu comme un jeu vidéo. Tout comme l'Ontop duo meac, il permet à l'apprenant de comprendre le domaine de l'aviation en pratiquant à moindre coût car il n'a besoin que d'une console de jeux (PlayStation, Xbox, etc) et de contrôleurs, ici des manettes des consoles ou des dispositifs spéciaux.

Cette technique est beaucoup moins rependue dans les centres de formation, car elle est plus adaptée pour les apprenants désireux de s'entraîner chez eux et ne disposant pas des moyens pour un Ontop duo meac.



FIGURE 1.2 – Simulateur de vol Microsoft flight simulator

— Osso VR

Osso VR est une plateforme de formation et d'évaluation chirurgicales qui offre aux entreprises de dispositifs médicaux et aux professionnels de la santé des moyens radicalement meilleurs de partager, de pratiquer. Tout comme dans le domaine de l'aviation,

le domaine médical, plus précisément chirurgical utilise des outils de simulation de l'aspect pratique de l'apprentissage. Ceci dû aux risques liés à l'expérimentation sur des individus vivants et le manque de cadavre.

Osso VR utilise la réalité virtuelle pour les expérimentations, afin de simuler un monde en trois dimensions représentant un laboratoire dans lequel est disposé un patient virtuel sur lequel seront effectuées les expérimentations.



FIGURE 1.3 – Simulateur chirurgical Osso VR

— **Praxilab**

Praxilab est un outil didacticiel permettant la simulation d'expériences scientifiques dans les domaines de la biologie, la physique et la chimie. Cet outil est développé pour les plateformes desktop et web. Il permet aux apprenants de suivre des instructions afin de réaliser des expériences et comprendre des phénomènes liés aux différents domaines.



FIGURE 1.4 – Simulateur scientifique Praxilab

— **EON Reality**

EON Reality est une solution de formation académique et industrielle en réalité augmentée et virtuelle. Elle permet l'expérimentation virtuelle dans plusieurs domaines de l'enseignement, à savoir la mécanique, la chimie, la biologie, l'histoire, le génie civil et plein d'autres domaines.

Cette solution est disponible sur de nombreux supports à savoir ordinateur sur Windows, casques de réalité virtuelle oculus, sous Android et ios.



FIGURE 1.5 – Simulateur scientifique EON RealityC

Tableau 1.1 – Comparatif des solutions existantes

Application	Avantages	Inconvénients
ONTOP DUO MEAC	Très immersif et très réaliste, maintenu et populaire dans le milieu de l'aviation.	Trop chère pour une utilisation à domicile, pas adapté aux sciences chimiques.
Microsoft flight simulator	Assez peu chère à l'obtention donc adapté à une utilisation à domicile et maintenu	Pas très immersif et réaliste, pas adapté aux sciences chimiques
Oss VR	Immersif, assez réaliste et maintenu	Disponible seulement en démo, pas adapté aux sciences chimiques.
Praxilab	Assez réaliste, accessible, adapté aux expériences chimiques et maintenu	Pas très immersif, impossible de créer des réactions, pas de version française
EON Reality	Immersif, très polyvalent, disponible sur de nombreuses plateformes et en plusieurs langues, et maintenu	Impossible de créer des réactions

Des outils précédemment présentés, deux ont particulièrement retenu notre attention, car étant beaucoup plus proches de la solution à implémenter. Nous parlons ici de PRAXILAB et EON Reality.

Le tableau ci-dessous fait une présentation plus détaillée de ces deux solutions.

Tableau 1.2 – Description des solutions les plus proches de la notre

Application	Fonctionnalités
Praxilab	Expériences virtuelles immersives en biologie sur des sujets allant de l'extraction d'ADN et du clonage génétique à la culture tissulaire et à l'électrophorèse des protéines, Expériences virtuelles immersives sur les domaines de la chimie générale, analytique et organique, Expériences virtuelles immersives sur le domaine de la physique.
EON Reality	Expériences virtuelles, immersives et réaliste des réactions chimiques préenregistrées sur la plateforme

1.2.4 Questions de recherche

- Comment pouvons-nous rendre l'enseignement de la chimie moins couteux ?
- Comment pouvons-nous rendre l'enseignement de la chimie moins risqué ?
- Comment conserver l'aspect réaliste et immersif des expérimentations réelles ?
- Comment permettre aux enseignants de créer des réactions pour leurs expérimentations ?

1.3 Choix et intérêt du sujet

Le contexte dans lequel nous nous trouvons et la problématique nous oriente vers un choix de thème qui est « **Analyse, conception et réalisation d'un module d'apprentissage immersif basé sur la réalité virtuelle : cas des travaux pratique de la chimie** ». La plateforme **VREDU Chemistry lab** permettra non seulement de résoudre le problème lié aux coûts de maintenance du matériel et des infrastructures en limitant aussi les risques au cours des expérimentations.

1.4 Objectif du travail

L'objectif principal est d'apporter un outil d'aide à l'enseignement secondaire, plus précisément de la chimie afin qu'un enseignant puisse créer des réactions que les apprenants suivront durant la phase pratique du cours.

Les objectifs spécifiques sont :

- Représentation des réactifs et des produits en trois dimensions et de façon réaliste et immersive
- Calcul de la quantité de matière des réactifs dans une solution
- Calcul de la concentration des réactifs dans une solution
- Calcul de la masse molaire moléculaire des réactifs dans une solution
- Calcul du ph d'une solution

1.5 Méthodologie

1.5.1 Gestion de projet

Pour le pilotage de notre projet, nous devons choisir parmi les méthodes de gestion de projets une qui convient le mieux non seulement à notre projet, mais également à notre contexte en entreprise. Pour ce faire, nous allons dans un premier lieu faire une comparaison de ces méthodes afin de faire un choix convenable.

Alors que les méthodes traditionnelles visent à traiter les différentes phases d'un projet d'une manière séquentielle (que l'on nomme aussi cycle de développement en cascade ou encore cycle en V), le principe des méthodes Agiles est de le découper en sous-parties (ou sous-projets) autonomes (on parle également de développement itératif). Les parties (itérations) forment le projet dans sa globalité.

Nous allons présenter dans un tableau une étude comparative entre l'approche de gestion de projets agiles et celle traditionnelle.

Tableau 1.3 – Comparaison entre les approches traditionnelles et agiles

Thème	Approche traditionnelle	Approche agile
Cycle de vie	En cascade ou en V, sans rétroactions possibles, phases séquentielles	Itératif et incrémental
Planification	Produite en quantité importante comme support de communication, de validation et de contractualisation	Réduite au strict nécessaire au profil d'incrément fonctionnels opérationnels pour le feedback du client
Équipe	Contrôle de qualité à la fin du cycle de développement. Le client découvre le produit fini.	Un contrôle de qualité précoce et permanent au niveau du produit et du processus. Le client visualise les résultats tôt et fréquemment
Qualité	Une équipe avec des ressources spécialisées et dirigées par un chef de projet.	Une équipe responsabilisée où l'initiative et la communication sont privilégiées et soutenues par le chef de projet
Suivi d'avancement	Mesure de la conformité aux plans initiaux. Analyse des écarts.	Un seul indicateur d'avancement : le nombre de fonctionnalités implémentées et le travail restant à faire.
Changement	Résistance, voire opposition au changement. Processus lourd de gestion des changements acceptés	Accueil favorable au changement, intégré dans le processus
Gestion des risques	Processus distinct, rigoureux de gestion des risques	Gestion de risques intégrée dans le processus global, avec responsabilisation de chacun dans l'identification et la résolution des risques. Pilotage par les risques
Mesure du succès	Respect des engagements initiaux en termes de coût, de budget et ce niveau de qualité	Satisfaction client par la livraison de valeur ajoutée

De cette comparaison, il est possible de choisir, selon le projet, l'approche qui convient le mieux.

Au vu de notre projet, il convient pour nous de choisir une méthode agile, car cette dernière s'adapte facilement aux changements et n'impose pas une planification rigide dès le début du projet. Parmi les méthodes agiles existantes, nous devons en choisir une qui nous convient encore plus. Après cette comparaison, notre choix s'oriente vers SCRUM. Car au vu de la grandeur de notre projet, de la taille de l'équipe, des préférences de l'entreprise et de l'approche orientée objet, nous pensons qu'elle est la méthode la mieux adaptée.

Tableau 1.4 – Comparaison des méthodes agiles

Méthode	Flexibilité	Itératif	Taille
Scrum	oui	oui	toute
Crystal clear	oui	Non	petite
Processus unifié Agile	oui	Non	toute
eXtreme Programming	oui	oui	petite

1.5.1.1 Qu'est-ce que la méthode SCRUM

Scrum est un cadre léger qui aide les personnes, les équipes et les organisations à générer de la valeur grâce à des solutions adaptatives pour des problèmes complexes [1].

Souvent considéré comme un framework de gestion de projets Agiles, Scrum décrit un ensemble de réunions, d'outils et de rôles qui interagissent de concert pour aider les équipes à structurer leur travail et à le gérer.

Scrum est fondé sur l'empirisme et la pensée Lean. L'empirisme affirme que la connaissance vient de l'expérience et de la prise de décisions basées sur ce qui est observé. La pensée Lean réduit le gaspillage et se concentre sur l'essentiel [1].

Scrum utilise une approche itérative et incrémentale pour optimiser la prévisibilité et contrôler les risques. Scrum engage des groupes de personnes qui ont collectivement toutes les compétences et l'expertise pour faire le travail et partager ou acquérir ses compétences selon les besoins. Scrum combine quatre événements formels pour l'inspection et l'adaptation dans un événement conteneur, le sprint. Ces événements fonctionnent parce qu'ils mettent en œuvre les piliers empiriques de Scrum que sont la transparence, l'inspection et l'adaptation.

- **La transparence** : la transparence est un facteur clé de réussite. Tout au long du développement du produit, l'équipe de développement et les parties prenantes ont accès aux informations basées sur un langage et des définitions communs. Par exemple, la définition de fini (DOD, definition of done) est obligatoire et très importante pour Scrum. La définition de prêt (DOR, definition of ready) est aussi une pratique couramment utilisée, mais non obligatoire à ce jour si on se réfère au scrum guide.
- **L'inspection et l'adaptation** : l'équipe doit se consulter quotidiennement pour déterminer rapidement d'éventuels écarts entre l'objectif de l'itération (Sprint Goal) et le travail réalisé. Cette Inspection dans le sprint a lieu principalement lors du Daily scrum, de la Sprint Review et de la Sprint Retrospective. Si des écarts sont constatés, un ajustement doit être entrepris afin d'atteindre les objectifs du sprint.
L'Inspection et l'adaptation permettent d'ajuster en permanence le développement d'un produit en fonction de l'apprentissage réalisé lors de chaque itération.

1.5.1.2 Pourquoi la méthode SCRUM

Cette méthode permet de répondre aux besoins des utilisateurs rapidement, dans les délais impartis, tout en respectant les budgets. En effet, elle canalise et modélise toutes les étapes du développement d'un logiciel. Elle ordonne aussi très clairement les différents jalons.

1.5.1.3 Avantages de la méthode SCRUM

Les équipes qui optent pour la structure Scrum gagnent en agilité et en flexibilité. Elle contribue à renforcer la collaboration au sein des équipes et les aide à atteindre leurs objectifs plus efficacement. Par ailleurs, les équipes Scrum savent en permanence sur quoi elles travaillent : elles accomplissent des tâches de leur backlog produit et ont une idée claire de leurs objectifs, car elles se sont concertées sur la définition d'un travail « terminé ».

1.5.1.4 Dans quels cas utiliser la méthode SCRUM

Offrant plus de réactivité, elle est plus adaptée que les méthodes traditionnelles pour la gestion de projets web, tel que le développement logiciel, car elle traduit et organise les projets de façon simple, transparente et pragmatique.

Ce framework, ou cadre de travail, est utile quand :

- L'ensemble d'un projet complexe ne peut être ni anticipé ni planifié entièrement.
- Son pilotage demande un minimum de flexibilité pour intégrer facilement des changements aux planifications initiales.

1.5.1.5 Principale contrainte de la méthode SCRUM

Les projets Scrum souffrent souvent de dérives des objectifs, car cette méthode accepte et encourage le changement. Cependant, il présente des risques d'itérer sans obtenir de résultats concrets si les changements sont trop nombreux ou que les retours clients sont discordants.

1.5.2 Analyse et Modélisation

Pour la réalisation du projet, nous allons procéder comme suit :

- Séparation des différents modules à déployer
- Développer les différents en utilisant l'approche de développement orienté objet
- Déployer ces modules pour la production

1.5.2.1 L'approche orientée objet

L'approche orientée objet considère le logiciel comme une collection d'objets dissociés, identifiés et définis par des propriétés. Une propriété est soit un attribut, soit une méthode. La fonctionnalité du logiciel émerge alors de l'interaction entre les différents objets qui le constituent. L'une des particularités de cette approche est qu'elle rapproche les données et leurs traitements associés au sein d'un unique objet. Un objet est caractérisé par plusieurs notions dont :

- **L'identité** : l'objet possède une identité, qui permet de le distinguer des autres objets, indépendamment de son état. On construit généralement cette identité grâce à un identifiant découlant naturellement du problème (par exemple une Banque pourra être repérée par un code, un Encaissement par un numéro identifiant, etc.)

- **Les attributs** : il s'agit des données caractérisant l'objet. Ce sont des variables stockant des informations sur l'état de l'objet.
- **Les méthodes** : les méthodes d'un objet caractérisent son comportement, c'est-à-dire l'ensemble des actions (appelées opérations) que l'objet est à même de réaliser. Ces opérations permettent de faire réagir l'objet aux sollicitations extérieures (ou d'agir sur les autres objets). De plus, les méthodes sont étroitement liées aux attributs, car leurs actions peuvent dépendre des valeurs des attributs, ou bien les modifier.

La difficulté de cette modélisation réside dans la création d'une représentation abstraite, sous forme d'objets, d'entités ayant une existence matérielle (Exemple : Banque, Guichet, Caisse, etc.) ou bien virtuelle (Exemple : Encaissement, Décaissement, Transfert de fonds, etc.). La Conception Orientée Objet (COO) est la méthode qui conduit à des architectures logicielles fondées sur les objets du système, plutôt que sur la fonction qu'il est censé réaliser.

1.5.2.2 UML et MERISE

Les différences entre l'approche objet avec UML et l'approche systémique (fonctionnelle) avec Merise sont mises en évidence dans l'étude comparative ci-dessous :

- **Points communs :**

L'approche classique et l'approche objet distinguent bien globalement trois grandes étapes dans le processus de conception et de développement d'une solution : l'analyse objet correspond au niveau conceptuel de merise, la conception objet est proche de la modélisation logique et organisationnelle de merise. Et enfin l'implémentation objet correspond à la réalisation dans merise.

Nous allons reprendre chaque grand niveau de représentation du SI et donner un certain nombre de précisions sur les points communs.

Le niveau de l'analyse objet ou le niveau conceptuel : dans les deux approches, la finalité de ses premiers niveaux de description d'un SI est d'appréhender les besoins à satisfaire et à donner une description de solutions indépendamment des considérations techniques des niveaux logiciels et physiques. Autrement dit, les préoccupations traitées sont très proches malgré des concepts par complètement identiques au niveau conceptuel et au niveau de l'analyse objet.

Le niveau conception objet ou le niveau logique organisationnel : ce niveau de description a bien pour finalité dans les deux approches de représenter la solution à implémenter sous l'angle de la logique informatique tant sur la partie des données que sur celle des traitements. Le niveau implémentation physique ou opérationnel dans les deux approches la préoccupation est la description physique et opérationnelle des données et des traitements.

- **Différences :**

Nous observons les différences entre ces deux approches au niveau des domaines d'application, de la démarche, les données et les traitements puis l'aspect évolution du système.

- **Les domaines d'application :**

Merise a pour vocation de traiter les systèmes d'information des entreprises, principalement dans le domaine de l'informatique de gestion. Le domaine de l'informatique de

gestion se caractérise en général par un grand nombre de données à gérer et à stocker avec des traitements relativement peu complexes.

Le domaine privilégié par UML est le domaine de l'informatique technique ou industrielle caractérisé par la gestion de composants physiques du monde réel (Informatisation des automates est représentative de ce domaine). Dans ce type de domaine, les aspects traitements d'états et comportements des objets, prennent le pas sur la gestion des données. En plus de cet atout, UML traite également sans difficulté majeure le domaine tel que l'informatique de gestion.

— **La démarche**

Avec merise, la démarche est structurée en étapes et phases dont l'étude préalable, l'étude détaillée, la réalisation et la mise en œuvre. Il correspond en effet au cycle de vie d'un système d'information. Et l'ensemble des résultats produits à chaque étape constitue le cycle de décision. Merise propose donc une démarche en cascade, c'est-à-dire qu'une étape ne peut être entamée que si l'étape précédente est achevée. Cela nécessite une organisation minutieuse du projet. Dans le cas contraire, l'on pourrait noter quelques blocages ou une lenteur dans le processus de modélisation du système d'information. Avec UML, la démarche est itérative, incrémentale, guidée par les besoins des utilisateurs du système, et centrée sur l'architecture logicielle. La démarche itérative permet de mieux comprendre et représenter un système complexe. Le périmètre du système à modéliser est défini par les besoins des utilisateurs (les utilisateurs définissent ce que doit être le système).

— **Choix d'une méthode d'analyse**

Suite à notre étude comparative entre l'approche systémique avec Merise et l'approche objet avec UML, Nous opterons donc pour une méthode d'analyse suivant l'approche objet dont UML pour la modélisation, dans l'étude conceptuelle de notre système. Ensuite, au vu des circonstances et les délais de notre projet, nous optons pour une démarche itérative, incrémentale, guidée par les besoins des utilisateurs du système. De plus, nous souhaiterions organiser nos programmes en rassemblant les données et les traitements en vue de former des entités cohérentes, logiques et stables. Enfin, nous aimerions faciliter les éventuelles évolutions et maintenances du système.

Chapitre 2

GÉNÉRALITÉS SUR LES OUTILS D'APPRENTISSAGE IMMERSIF BASÉ SUR LA RÉALITÉ VIRTUELLE

Afin de mener à bien un projet, il est important de comprendre les termes techniques tournant autour du sujet, notre thème porte ainsi sur la réalisation d'un outil d'apprentissage immersif basé sur la réalité virtuelle. Nous allons donc dans ce chapitre expliquer des concepts clés tels que l'apprentissage immersif et la réalité virtuelle

2.1 Apprentissage

L'apprentissage a été défini fonctionnellement comme des changements de comportement qui résultent de l'expérience ou mécaniquement comme des changements dans l'organisme qui résultent de l'expérience [2]. C'est le processus par lequel les êtres vivants acquièrent et transmettent leurs savoirs. L'apprentissage a été un sujet central dans la recherche psychologique pratiquement depuis le début de la psychologie en tant que science indépendante (par exemple, Ebbinghaus, 1885/1962 ; Thorndike, 1911). Pendant la plus grande partie du siècle précédent, c'était même le sujet le plus étudié en psychologie. De plus, aujourd'hui, les questions sur l'apprentissage sont abordées dans pratiquement tous les domaines de la psychologie et aussi de l'éducation [2]. De sa définition ressort un principe important, celui de l'expérience, la répétition d'une action qui nous amène à mieux la comprendre. L'acquisition de cette expérience est le principal défi auquel les organismes éducatifs doivent faire face aux vues du nombre grandissant de techniques et de moyen d'enseignant qui existe, moyens de plus en plus nombreux avec l'essor des nouvelles technologies telle que le web.

Comme énoncé ci-dessus, il existe un grand nombre de techniques permettant de dispenser ou d'acquérir des connaissances par le moyen de l'apprentissage, parmi lesquelles les plus communément rencontrés sont l'apprentissage traditionnel et l'apprentissage par les jeux.

2.1.1 L'apprentissage traditionnel

Par apprentissage traditionnel, on entend le processus d'apprentissage moderne le plus répandu et le plus ancien qui consiste à dispenser des notions dans un cadre très académique et très strict qui ne laisse donc que peu de liberté d'action ou d'interprétation en limitant toute forme de distraction pour une concentration optimale afin d'acquérir le maximum de connaissances sur un sujet en un temps données. Bien qu'assez ancienne comme méthode d'enseignement, elle tend à s'adapter à l'époque et aux contraintes auxquelles le monde est soumis notamment par sa digitalisation due aux avancées des TIC et à la récente pandémie de la covid qui ont créé un bouleversement dans son histoire, si bien que des écoles ou des formations complètement digitalisées ont vu le jour et offrent des diplômes qui ne présentent aucune distinction de valeur par rapport à ceux obtenus en présentiel. Cela s'explique par le fait que ces deux modes d'études sont considérés comme parfaitement égaux et qu'aucune distinction n'est faite entre eux en matière d'emploi [3].

Cette méthode d'apprentissage peut s'avérer très efficace pour l'apprentissage des petits enfants, mais des recherches récentes suggèrent que davantage de méthodes basées sur la découverte pourraient être encore plus efficaces. [4]

2.1.2 L'apprentissage par le jeu

L'apprentissage par le jeu est une approche pédagogique qui favorise le recours à des activités ludiques pour stimuler de nombreux aspects du développement et de l'apprentissage de l'enfant [5]. Au début des années 2000, il y a eu un basculement vers la recommandation de l'emploi de l'apprentissage par le jeu dans les programmes éducatifs préscolaires dans plusieurs pays, notamment le Canada, [6] la Suède, [7] la Chine, [8] les Émirats arabes unis [9] et la Nouvelle-Zélande [7]. L'environnement moins académique et moins strict permet aux apprenants de

réellement explorer le sujet soit de manière complètement autonome grâce aux jeux libre [10] dirigé par l'enfant lui-même, soit avec un degré d'encadrement grâce aux jeux dirigés [11].

Ce type d'apprentissage est plus ou moins efficace en fonction de l'expérience immersive que propose. En effet, l'opposition des méthodes d'apprentissage par le jeu immersive et non immersive a montré qu'un apprenant aura beaucoup plus de facilité à comprendre un sujet et à s'y intéresser davantage lorsqu'il y est complètement immergé [12–14].

2.2 La réalité virtuelle et l'éducation

2.2.1 La réalité virtuelle

Le virtuel peut être défini comme « étant par essence ou effet, mais pas en fait » [15] et la réalité comme « l'état ou la qualité d'être réel » [15]. De la combinaison de ces deux concepts est né un environnement généré par ordinateur avec lequel il est possible d'interagir et de faire l'expérience des sens humains ordinaires comme si l'environnement était réel [16]. La réalité virtuelle (VR) utilise des combinaisons de matériel informatique et de logiciels pour représenter différents aspects du monde physique à un individu en temps réel. Un objectif clé de la conception de la réalité virtuelle est d'instiller un sentiment de présence, ou l'illusion d'être immergé dans l'environnement, par opposition à la simple visualisation de l'environnement d'un point de vue extérieur (voir illustration).

2.2.1.1 Principe de fonctionnement

Cinquante ans se sont écoulés depuis que Sutherland a présenté sa vision de l'Ultimate Display [17] imitant le monde réel dans tous les sens disponibles. Et depuis lors, une grande quantité de technologies individuelles prenant en charge cette stimulation sensorielle ont émergé, mais ce n'est qu'en 1989 que Jaron Lanier a inventé le terme de réalité virtuelle [16]. En 2012, près d'un quart de siècle après la première vague de réalité virtuelle, un projet Kickstarter nommé Oculus Rift, dans le but de fournir au public un écran monté sur la tête abordable et de haute qualité, cherchait un financement et a atteint l'objectif de 250 000 \$ en moins de 24 heures. Ce fut l'étincelle initiale qui a lancé la deuxième vague de VR [18]. La réalité virtuelle comme tous les domaines des TIC utilisent des périphériques de deux types, à savoir périphériques d'entrée et périphériques de sortie.

Les périphériques d'entrées en réalité virtuelles sont des équipements permettant à l'utilisateur d'interagir avec l'environnement 3D. Le développement de ces périphériques est très diversifié et remplis de nombreuses catégories parmi lesquelles nous pouvons citer des casques de réalité virtuelle, des contrôleurs ou manette de réalité virtuelle, des tapis roulant omnidirectionnel, des capteurs de posture, capteur de suivi des gestes [18].

Les casques de réalité virtuelle : est le périphérique de sortie dans l'univers de la réalité virtuel, il permet la visualisation d'environnement 3D grâce à deux écrans disposés comme des lunettes au niveau des yeux de l'utilisateur et de capteur de mouvement permettant à l'utilisation d'un mouvement de tête de changer de point de vue dans l'environnement 3D. De nombreux équipements l'affiche d'environnement en 3D ont vu le jour ces dernières années parmi lesquelles les plus connus sont : le rift d'oculus, le HTC Vive et le PlayStation VR.



FIGURE 2.1 – L'oculus rift (gauche), le HTC Vive (centre) et le PlayStation VR (droite)

Les contrôleurs ou manette de réalité virtuelle, ces équipements sont très proches des manettes traditionnelles utilisées pour des consoles de jeux telles que PlayStation et Xbox mais elles sont plus adaptées à un environnement immersif du fait de leur forme et du positionnement des boutons et des joysticks. De nombreuses entreprises se sont lancées dans la conception de ces équipements parmi lesquelles nous pouvons citer : l'oculus Half Moon contrôleur pour des casques de réalité virtuelle de la marque oculus, le steamVR pour le casque HTC Vive et Reactive Grip utilisable sur plusieurs marques de cas de réalité virtuelle parmi lesquelles oculus est vive.



FIGURE 2.2 – L'oculus Half Moon (gauche), le reactive Grip (centre) et le SteamVR (droite)

Les tapis roulant omnidirectionnel ce sont des équipements permettant le déplacement dans un environnement 3D en reproduisant les mouvements du monde réel, à savoir la marche, la course et les sauts grâce à un tapis roulant multidirectionnel auquel est attaché le sujet grâce à un harnais afin de suivre et de reproduire ses mouvements. De nombreux équipements permettant la capture de ses différents mouvements existent parmi lesquelles nous avons : le Space Walker, le walkmouse et l'InfinaDeck.



FIGURE 2.3 – Le Virtuix Omni (gauche), le WalkMouse (centre), et le InfinaDeck (droite)

Les capteurs de posture ce sont des équipements permettant de reproduire les mouvements du squelette du sujet afin de le reproduire dans l'environnement. Ce type d'équipement crée un sentiment d'immersion dû à une reproduction des mimiques des utilisateurs. De nombreux équipements permettant la capture de ses différents mouvements existent parmi lesquelles nous avons : le STEM, le PrioVR et ControllVR.



FIGURE 2.4 – Le STEM (gauche), le PrioVR (centre), et le ControllVR (droite)

Les capteurs de suivi des gestes permettant de capturer le mouvement des mains, ils viennent souvent remplacer les contrôles classiques, car moins immersifs et sont souvent utilisés avec le capteur de posture. De nombreux équipements permettant la capture de ses différents mouvements existent parmi lesquelles nous avons : le Leap Motion, le Myo et le Glove One.



FIGURE 2.5 – Le Leap Motion (gauche), le Myo (centre), et le Glove One (droite)

2.2.2 La réalité virtuelle et l'éducation

La réalité virtuelle n'est pas une nouvelle technologie. Mais, plusieurs contraintes ont empêché son adoption effective. Les progrès technologiques récents ajoutés à la prolifération de matériel et de logiciels abordables ont rendu la réalité virtuelle plus viable et souhaitable dans de nombreux domaines, notamment l'éducation ; ils ont été relancés avec de nouvelles promesses jusqu'alors inimaginables. La nature de la réalité virtuelle promet de nouveaux modèles d'enseignement et d'apprentissage qui répondent mieux aux besoins de l'apprenant du 21e siècle. Nous sommes maintenant sur la voie de réinventer l'éducation.

Cette technologie permet désormais d'enseigner en toute sécurité sans risque d'impact réel dans de nombreux domaines d'enseignement telle que la chimie, la biologie, la médecine et bien des domaines où l'expérimentation est risquée soit pour l'apprenant ou son environnement.

Au cours d'une expérience dans un lycée chinois, lorsque du benzène a été ajouté à un mélange d'acide sulfurique concentré et d'acide nitrique concentré, le mélange s'est soudainement envolé du tube à essai et a éclaboussé les yeux de l'élève impliqué [19]. Les enseignants et les manuels expliquent très bien les risques liés à ce genre de réaction, mais en raison de leur manque de familiarité avec les procédures expérimentales sûres, les étudiants peuvent oublier les risques pour la sécurité lors de la réalisation d'une expérience [19]. Ces accidents surviennent également dans les établissements d'enseignement supérieur, comme les universités. Un accident récent s'est produit à l'Université Jiaotong de Pékin qui a causé la mort de trois étudiants qui participaient à l'expérience. Allant de ce constat Xiaoyun Duan [19] et une équipe de chercheur ont opté pour un dispositif utilisant la réalité mixte¹ pour simuler des réactions afin de créer un environnement sûr pour les apprenants.

1. La réalité mixte est la fusion de mondes réels et virtuels pour produire de nouveaux environnements et visualisations, où les objets physiques et numériques coexistent et interagissent en temps réel.

Au cours de la dernière décennie, l'application de la technologie VR s'est étendue de l'industrie du divertissement à la médecine clinique. Des chercheurs et des médecins ont exploré les effets de la simulation VR sur la réadaptation physique, la gestion de la douleur, la formation chirurgicale, l'éducation anatomique et le traitement des troubles psychiatriques [20–25].

Le champ d'application de cette technologie ne se limite pas qu'à la chimie et à la médecine, elle peut s'étendre à de nombreux autres domaines, car la seule limite à son utilisation est l'imagination.

Deuxième partie

ÉTUDE DÉTAILLÉE ET RÉALISATION

Chapitre 3

Cadrage, analyse et conception

Comme tout problème qui a été clairement défini, il faut le réaliser, mais nous ne pouvons parler d'implémentation sans toutefois faire une analyse et une conception. Pour ce faire, nous allons cadrer le projet en utilisant la méthode CPS, suivi d'une analyse fonctionnelle et non fonctionnelle et bien sûr terminer par une conception des architectures physique et logique.

3.1 Cadrage du projet

3.1.1 Le projet

3.1.1.1 Le nom

Le projet à réaliser consiste à pouvoir dispenser les cours (médecine, mathématiques, chimie) du système éducatif et le travail portera sur le module d'apprentissage de la chimie, d'où le nom **vredu chemistry lab**.

3.1.1.2 Définition succincte

vredu chemistry lab sera développée pour la plateforme de réalité virtuelle et les navigateurs web. Elle se devra donc d'être compatible non seulement avec tous les casques de réalité virtuelle de la marque OCULUS, mais aussi avec tous les appareils disposant d'un navigateur web. Il sera ainsi possible pour un enseignant de programmer une expérience chimique sur un navigateur web afin que l'apprenant puisse la suivre sur son casque de réalité virtuelle.

3.1.1.3 Caractéristiques essentielles

Notre application sera caractérisée par :

- Une accessibilité sur tous les formats d'équipements numériques de la marque oculus et sur les appareils disposant d'un navigateur web,
- Prise en charge des éléments permettant d'effectuer une réaction chimique (éléments du tableau périodique, éprouvettes, bêchers, centrifugeuses...).

3.1.1.4 Motifs qui sous-tendent ce projet

De nos jours, l'univers du numérique offre énormément d'opportunités en ce qui concerne l'acquisition de connaissance grâce à la grande quantité d'informations échangées. La chimie est une science dont l'apprentissage se fait de façon expérimentale, expériences entourées de contraintes auxquelles une solution digitale pourrait être la solution, à savoir :

- Annuler la contrainte du lieu des expériences.
- Annuler la contrainte liée à l'acquisition et à la maintenance du matériel d'expérimentation.
- Limiter des risques d'accidents durant les expérimentations.
- Diminuer les couts de maintenance et d'acquisition du matériel de laboratoire.

3.1.2 Les objectifs

3.1.2.1 Objectifs techniques

- **Résultats attendus** : à la fin de ce projet, nous devons avoir deux applications, une pour le web permettant à un enseignant de chimie la création de réactions chimiques auxquelles les apprenants participent par le billet de la seconde application qui elle tournera dans un environnement immersif virtuel.

- **Objectif principal** : obtenir un laboratoire de chimie virtuelle permettant la simulation des réactions chimiques.
- **Objectifs secondaires** :
 - Représenter les réactifs et les produits en trois dimensions et de façon réaliste et immersive
 - Calculer la quantité de matière des réactifs dans une solution
 - Calculer la concentration des réactifs dans une solution
 - Calculer la masse molaire moléculaire des réactifs dans une solution
 - Calculer le ph d'une solution

3.1.2.2 Objectifs de délai

L'application doit être réalisée dans un délai de sept mois à compter de la date du 10 mai 2022.

3.1.2.3 Objectifs de coûts

3.1.2.3.1 Estimation de la main d'œuvre

1. Les différents modèles d'estimation des coûts et ressources d'un projet

Au cours de nos recherches, nous avons trouvé de nombreux modèles d'estimation des coûts et ressources d'un projet. Nous pouvons entre autres parlé de :

- a) **La méthode descendante** : reste extrêmement populaire dans la gestion de projets contemporains. L'expression « top-down » (descendante) signifie que les instructions sont données en amont. Les objectifs du projet sont fixés par la direction.
- b) **La méthode ascendante** : elle se caractérise par une participation proactive de l'équipe dans le processus d'exécution du projet. Les membres de l'équipe sont invités à participer à toutes les étapes du processus de gestion. L'ensemble de l'équipe est amené à décider de la marche à suivre.
- c) **La méthode COCOMO** : (acronyme de l'anglais Constructive Cost Model) est un modèle permettant de définir une estimation de l'effort à fournir dans un développement logiciel et la durée que ce dernier aura en fonction des ressources allouées.

Sachant que la majeure partie du travail est consacrée à la production du logiciel, nous nous inspirons de la méthode COCOMO pour estimer les coûts de la ressource humaine à ce niveau, lesquels coûts seront inclus dans les coûts globaux de la solution envisagée.

2. Le modèle d'estimation des coûts COCOMO

Cocomo (Constructive Cost Model) est un modèle de régression basé sur le LOC, c'est-à-dire le nombre de lignes de code. Il s'agit d'un modèle procédural d'estimation des coûts pour les projets logiciels et souvent utilisé comme processus de prédiction fiable des divers paramètres associés à la réalisation d'un projet, tels que la taille, l'effort, le coût, le temps et la qualité. Il a été proposé par Barry Boehm en 1970 et repose sur l'étude de 63 projets, ce qui en fait l'un des modèles les mieux documentés. Différents modèles de Cocomo ont été proposés pour prédire l'estimation des coûts à différents niveaux, en fonction de la précision et de l'exactitude requises. Tous ces modèles peuvent être appliqués à une variété de projets, dont les caractéristiques déterminent la valeur de

constante à utiliser dans les calculs ultérieurs. Ces caractéristiques propres à différents types de systèmes sont mentionnées ci-dessous :

- a) **Organique** – Un projet logiciel est dit de type organique si la taille de l'équipe requise est suffisamment petite, si le problème est bien compris et a été résolu dans le passé et si les membres de l'équipe ont une expérience nominale du problème.
- b) **Semi-détaché** – Un projet logiciel est dit de type semi-détaché si les caractéristiques vitales telles que la taille de l'équipe, l'expérience, la connaissance des différents environnements de programmation se situent entre celles de l'organique et de l'Embedded. Les projets classés comme semi-détachés sont comparativement moins familiers et difficiles à développer par rapport aux projets organiques et nécessitent plus d'expérience et une meilleure orientation et créativité. Ex : Les compilateurs ou différents Systèmes Embarqués peuvent être considérés de type Semi-Détachés.
- c) **Intégré** – Un projet logiciel nécessitant le plus haut niveau de complexité, de créativité et d'expérience entre dans cette catégorie. Un tel logiciel nécessite une taille d'équipes plus importantes que les deux autres modèles et les développeurs doivent également être suffisamment expérimentés et créatives pour développer des modèles aussi complexes.

Tableau 3.1 – types de projet COCOMO

Type de projet	Effort en Homme Mois	Temps de Développement
ORGANIQUE	$2.4(KDSI)^{1.05}$	$2.5(HM)^{0.38}$
MÉDIAN	$3.0(KDSI)^{1.12}$	$2.5(HM)^{0.35}$
IMBRIQUÉ	$3.6(KDSI)^{1.20}$	$2.5(HM)^{0.32}$

Le projet est estimé à 20 KDSI (nombre de milliers d'instructions source livrées) en mode organique (Il est défini par une innovation minimale, une organisation simple en petites équipes expérimentées).

Formule :

$$E^1 = 2.4 \cdot KLOCx^{1.05}$$

$$D^2 = 2.5 \cdot E^{0.38}$$

Application numérique :

$$E = 2.4 \cdot 10^{1.05} \approx 27 \text{ Personnes/Mois}$$

$$D = 2.5 \cdot 56^{0.38} \approx 8 \text{ Mois}$$

-
1. E = Effort en Personne-Mois
 2. D = Durée en Mois.

Tableau 3.2 – Estimation des coûts

Participant	Quantité	Durée en semaine	Cout/Semaine	Cout total
Chef de projet / concepteur	1	4	200.000 XAF	800.000 XAF
Développeur	1	32	100.000 XAF	3.200.000 XAF
Testeur	2	1	50.000 XAF	50.000 XAF
Total				4.050.000 XAF

3.1.2.3.2 Estimation matérielle et logiciel

Ce tableau présente une estimation du matériel selon le besoin et la durée.

Tableau 3.3 – Estimation des coûts matériels

Ressource	Nombre de payement	Cout par payement	Total
Desktop monté avec les caractéristiques : 32 Go de RAM, Une carte graphique Nvidia GeForce GTX 1070, 250 Go de disque dur SSD, 1 To de disque dur HDD, Processeur Intel core i7 3.60GHz, Boitier et carte graphique predator, Ecran 32 pouces, Clavier HP	7	23.472 XAF <i>(Amortissement par mois)</i>	164.304 XAF
1 serveur de développement : 100 Go SSD, 4 Go Ram, Ubuntu 20	7	80.000 XAF	560.000 XAF
1 serveur git : 2 Go RAM, 30 Go HDD, Ubuntu 20	7	80.000 XAF	560.000 XAF
1 casque oculus	7	5.940 XAF <i>(Amortissement par mois)</i>	41.580 XAF
Connexion Internet haut débit	7	30.000 XAF	210.000 XAF
Rider	1	91.809 XAF	91.809 XAF
Unreal engine	7	Gratuit	Gratuit
Webstorm	1	91.809 XAF	91.809 XAF
Total			1.719.502 XAF

Le coût du matériel est ainsi estimé à : **1.719.502 XAF**

Ainsi le coût total du projet est de **5.769.502 XAF**

3.1.3 La technique

3.1.3.1 La base sur laquelle le projet s'appuie

Nous avons déjà eu à réaliser des projets en informatique et en particulier des applications Web, bien que les applications de réalité virtuelle soient pour nous une nouvelle expérience, nous pensons pouvoir y parvenir grâce à la communauté qui est nombreuse et également l'aide des professionnels en stage notamment notre encadreur. De plus, les supports de cours, les enseignants et l'internet (possibilité de documentation) sont des atouts majeurs renforçant cet engagement. Des didacticiels dans des environnements 3D existent déjà, ce qui nous fait croire encore plus à la réalisation de ce projet.

3.1.3.2 Les difficultés principales de ce projet

- Apporter du réalisme aux réactions chimiques
- Sécurité de l'application

3.1.3.3 Les solutions de repli en cas de problème

Pour le problème lié au réalisme à apporter aux expériences, nous avons pensé à confier la réalisation des assets 3D à des tiers que nous paierons.

En ce qui concerne la sécurité de l'application, vu que l'application sera hébergée en ligne, nous nous informerons sur les différents risques liés à un tel hébergement et tenterons de les éviter lors de la phase de développement.

3.1.4 Le planning

3.1.4.1 Dates clés

- Date de début : 10 mai 2022
- Date de fin : 03 décembre 2022

3.1.4.2 Grandes phases du planning

Pour la gestion du temps, il existe des méthodes de planification prévisionnelle de projet tel que le diagramme de Gantt. Ce dit diagramme regroupe toutes les tâches, les durées et les ressources ordonnées de manière graduelle permettant à toute l'équipe de suivre son évolution. Il peut être modifié au fur et à mesure en fonction des délais respectés ou pas ainsi que des imprévus. Dans le cadre de notre projet, la planification de nos tâches nous a permis de ressortir le diagramme de Gantt suivant :

Tableau 3.4 – Estimation des coûts matériels

Phase	Taches	Durée en jour	Durée en heure	Prédécesseurs
Analyse	Cadrage du projet (A)	4	36	
	Etude l'existant (B)	5	45	A
	Elaboration du diagramme des cas d'utilisation (C)	3	27	B
	Elaboration des diagrammes de séquence (D)	3	27	B
Conception	Elaboration du diagramme de classe (E)	5	45	D,C
	Elaboration du diagramme de déploiement (F)	1	9	D,C
Implémentation	Back end	Module de gestion des utilisateurs (G)	15	135
		Module de gestion des éléments chimiques (H)	12	108
		Module de gestion des équipements de laboratoire (I)	15	135
		Tests (J)	2	18
	Front end	Module de gestion des utilisateurs (K)	15	135
		Module de gestion des éléments chimiques (L)	15	135
		Module de gestion des équipements de laboratoire (M)	10	90
		Tests (N)	2	18
	Interfaces 3D	Menus et interface d'accueil (O)	10	90
		Gestion des réactions chimiques (P)	25	225
		Tests (Q)	5	45
Transition	Tests (R)	5	45	Q
Total		152	1368	

Ce tableau nous permettra de réaliser notre diagramme Gantt.

3.1.4.3 Diagramme Gant

Ce diagramme présente le planning général du projet, fait en utilisant le logiciel Gantt Project.

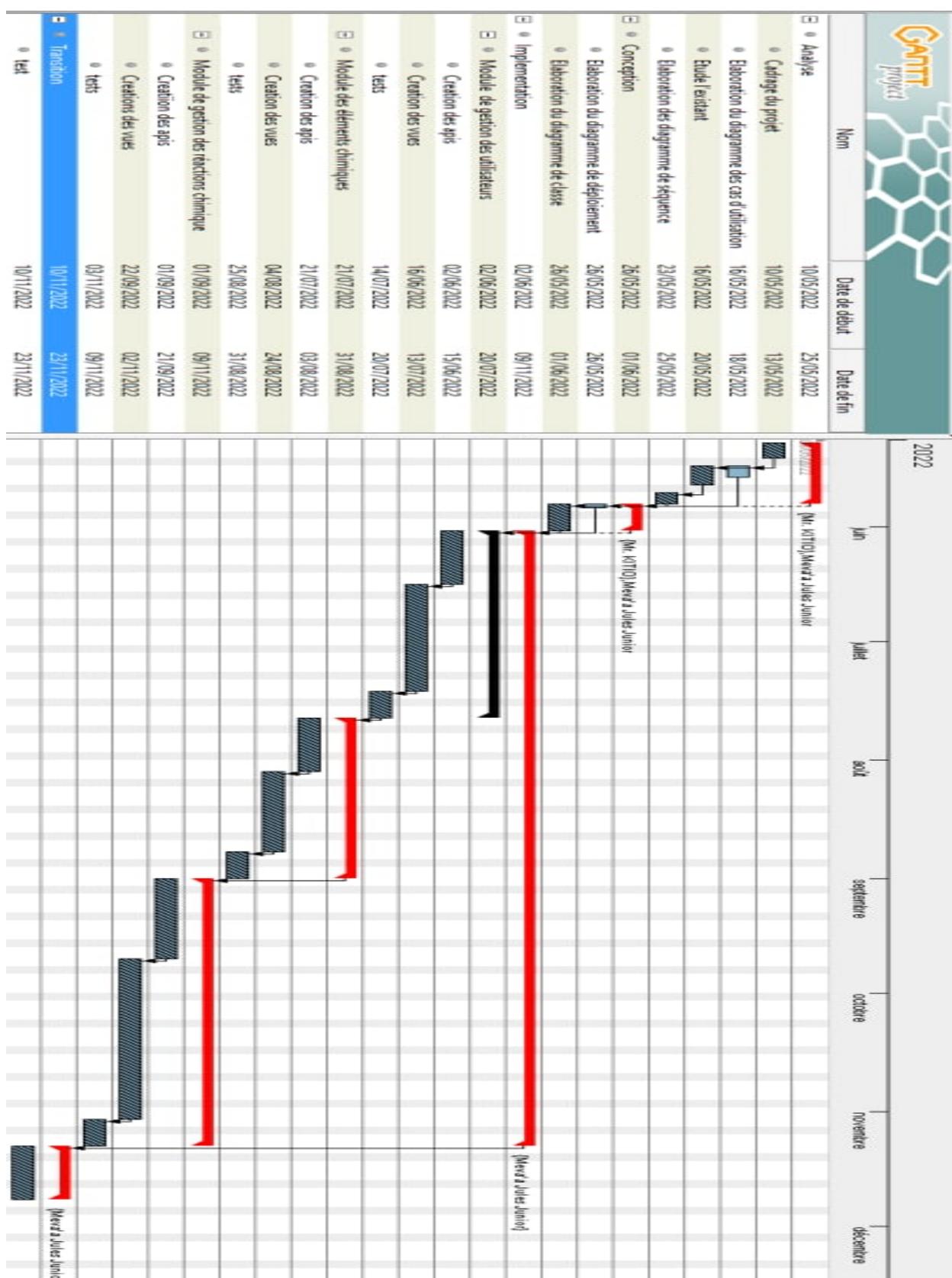


FIGURE 3.1 – Diagramme de gantt du projet

De ce diagramme, ressort la date de début du projet qui est le 10 mai 2022 et la date de fin le 03 décembre 2022. La durée du projet est de 156 jours. Le chemin critique est noté en bleu sombre sur ce diagramme.

3.1.5 Les moyens

3.1.5.1 Moyens humains

Ici nous retrouvons l'équipe chargée de la réalisation du module **M. KITIO VOUKENG Christian PATRICK** directeur technique à monglo-technologie et **MEVA'A JULES JUNIOR** stagiaire en développement et en management des solutions digitales et data à monglo-technologie.

3.1.5.2 Moyens matériels

Tableau 3.5 – Moyens matériels

Ordinateurs desktop	32 Go de RAM, Une carte graphique Nvidia GeForce GTX 1070, 250 Go de disque dur SSD, 1 To de disque dur HDD, Processeur Intel core i7 3.60GHz, Boitier et carte graphique predator, Ecran 32 pouces, Clavier HP
1 Casque de réalité virtuelle Oculus Quest	
1 serveur git	
1 serveur git	

3.1.6 Le management du projet

Pour la réalisation de ce projet, nous disposons des ressources humaines suivantes :

Tableau 3.6 – Le management du projet

Noms	Fonctions
M. KITIO VOUKENG CHRISTIAN PATRICK	Responsable du projet, analyste et testeur
MEVA'A JULES JUNIOR	Analyste, développeur et testeur

3.1.7 La communication

3.1.7.1 Communication interne du projet

- Collaboration via GitLab
- Communication via Telegram
- Réunion en présentiel ou en ligne (en fonction du contexte) pour l'évaluation de l'évolution du projet et les tests

3.1.7.2 Communication externe

- Avec les encadreurs académiques
- La communication avec l'encadreur se fera par des séances de travail en présentielle et parfois par moyens téléphoniques.
- Avec les potentiels clients
 - Information des corps administratif, professoral et étudiantin via les réseaux sociaux (Twitter, groupes WhatsApp, Facebook, Telegram) de l'existence de l'application.
 - Publication des affiches et des spots publicitaires
 - Descente dans les établissements

3.2 Analyse du système

3.2.1 Analyse fonctionnelle

Il s'agit là des fonctionnalités que doivent posséder le système :

3.2.1.1 Identification des acteurs

Un acteur représente un rôle joué par une entité externe (utilisateur humain, dispositif matériel ou autre système) qui interagit directement avec le système étudié. Un acteur peut consulter ou modifier directement l'état du système, en émettant ou en recevant des messages susceptibles d'être porteurs de données. Dans le cas de notre application, nous avons pu identifier ces différents acteurs :

- L'administrateur : il est chargé de la modération de l'application en ayant accès à toutes les fonctionnalités de la version web de l'application.
- L'enseignant : c'est celui qui crée les réactions chimiques dans le système.
- L'apprenant : c'est celui qui effectue les réactions dans un casque de réalité virtuelle.

3.2.1.2 Identification des cas d'utilisation

Il est question ici de ressortir une manière d'utiliser le système et d'en décrire les exigences fonctionnelles. Nous avons retenu les cas d'utilisation suivants :

Tableau 3.7 – Les acteurs du système et leurs cas d'utilisation

Acteurs	Cas d'utilisations
L'apprenant	Effectuer une réaction chimique
L'enseignant	S'inscrire
	Se connecter
	Consulter son compte
	Modifier son compte
	Supprimer son compte
	Créer, Lister, Modifier, activer, désactiver et supprimer des éléments chimiques
Administrateur	Effectuer tous les cas d'utilisation de l'enseignant
	Lister, activer, désactiver, modifier et supprimer un compte utilisateur.
	Lister, activer, désactiver, modifier et supprimer toutes les réactions chimiques.
	Lister, activer, désactiver, modifier et supprimer tous les éléments chimiques
	Créer, lister, modifier, activer, désactiver et supprimer du matériel de laboratoire.

3.2.1.3 Diagramme des cas d'utilisation

Les diagrammes de cas d'utilisation sont des diagrammes utilisés pour une représentation du comportement fonctionnel d'un système logiciel. Ils permettent d'avoir une vue d'ensemble entre les fonctionnalités d'un système et les acteurs qui en bénéficient.

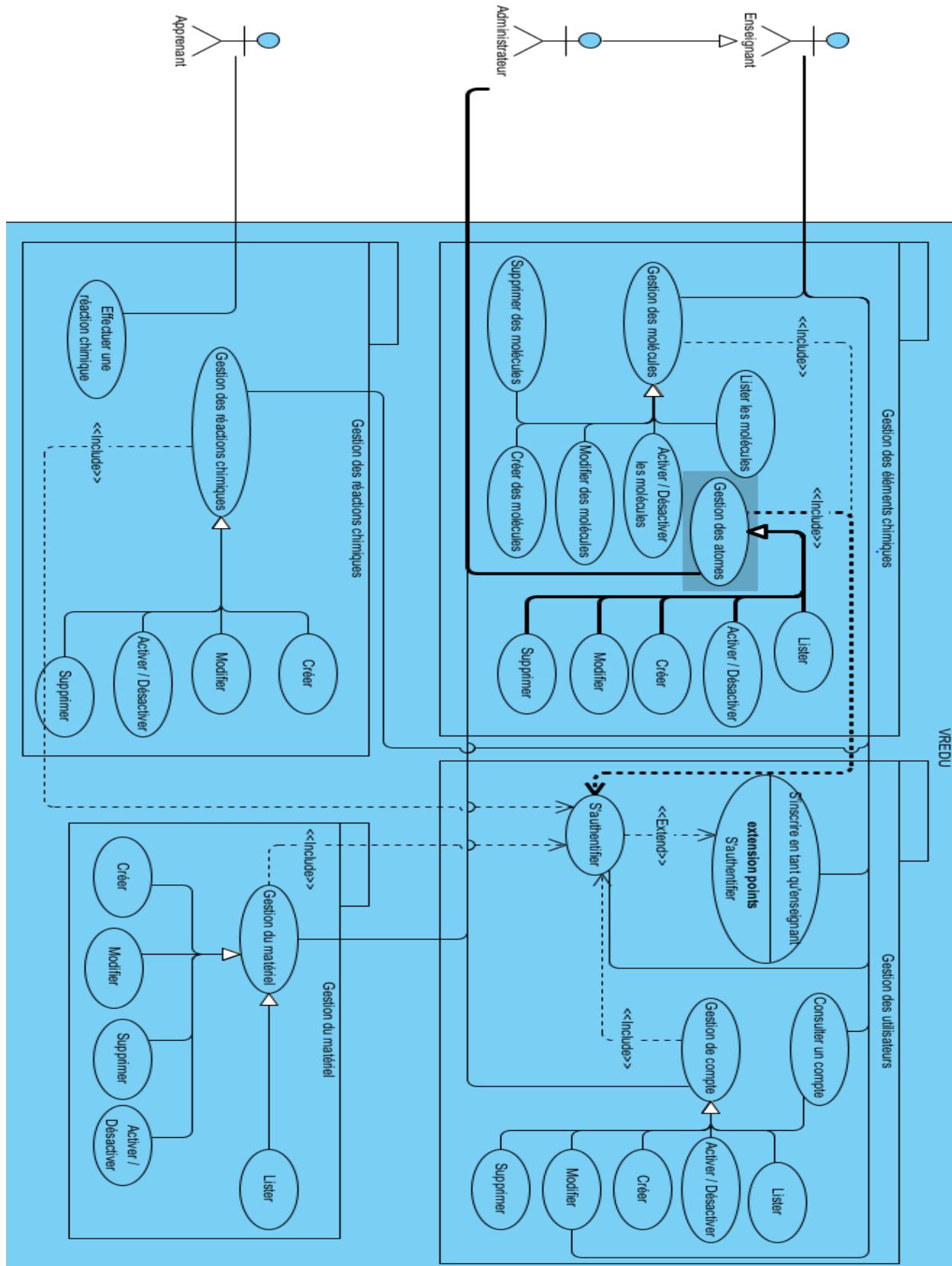


FIGURE 3.2 – Diagramme des cas d'utilisation

De ce diagramme ressortent les différents groupes de fonctionnalités du module de chimie de l'application VREDU à savoir : le module de gestion des utilisateurs qui permettra la gestion des comptes utilisateur (administrateur et enseignant), le module de gestion des éléments chimique qui permettra les opérations de création, modification. Activation et désactivation des éléments chimiques, le module de gestion des réactions chimiques et le module de gestion du matériel de laboratoire.

Chaque acteur de ce diagramme a certaines interactions bien spécifiques avec chaque module de l'application, interactions mieux illustrées dans les différents diagrammes des cas d'utilisations séparés par utilisateur suivants :

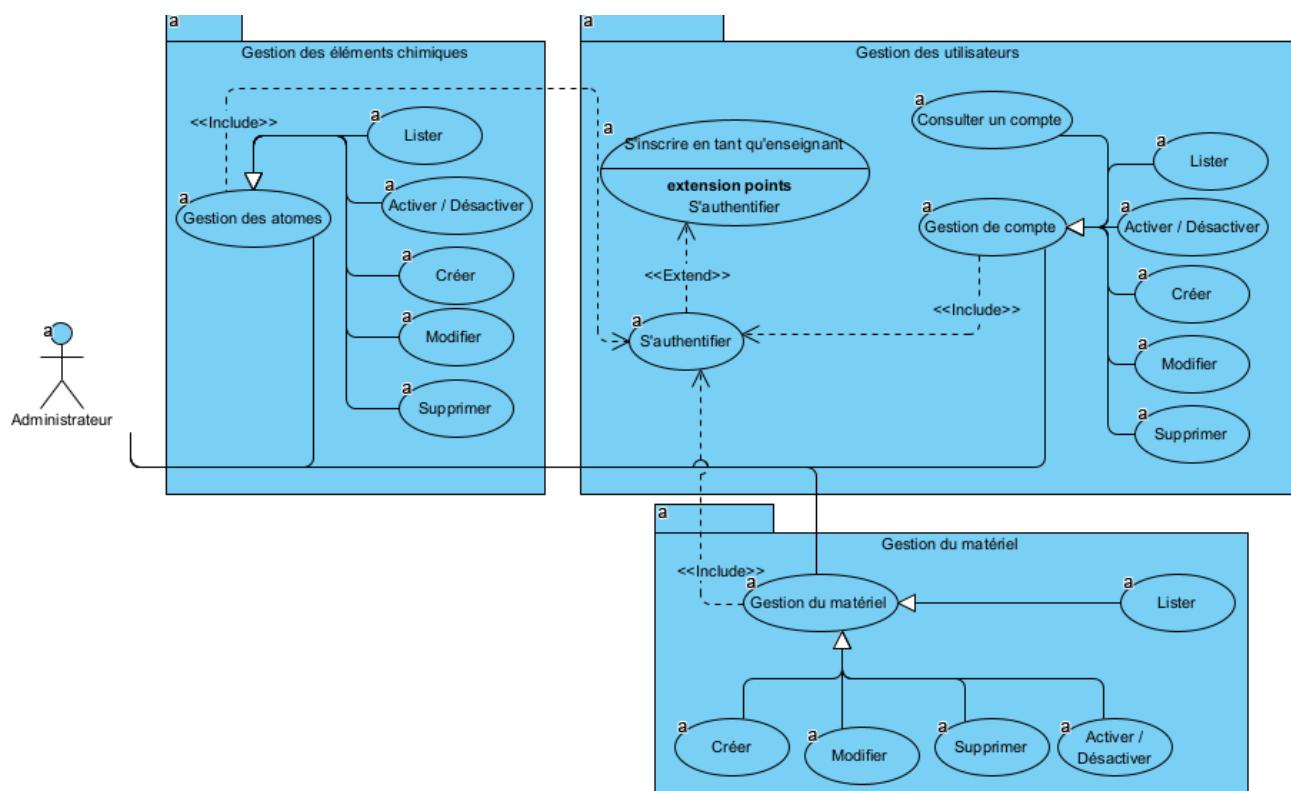


FIGURE 3.3 – Diagramme des cas d'utilisation d'un administrateur

La figure ci-dessus présente l'administrateur et tous ses cas d'utilisation de la plateforme regroupé par module. Nous constatons que le champ d'action de l'administrateur s'étend aux modules de gestion des utilisateurs, des éléments chimiques, du matériel, mais aussi aux différents cas d'utilisation liés à un enseignant.

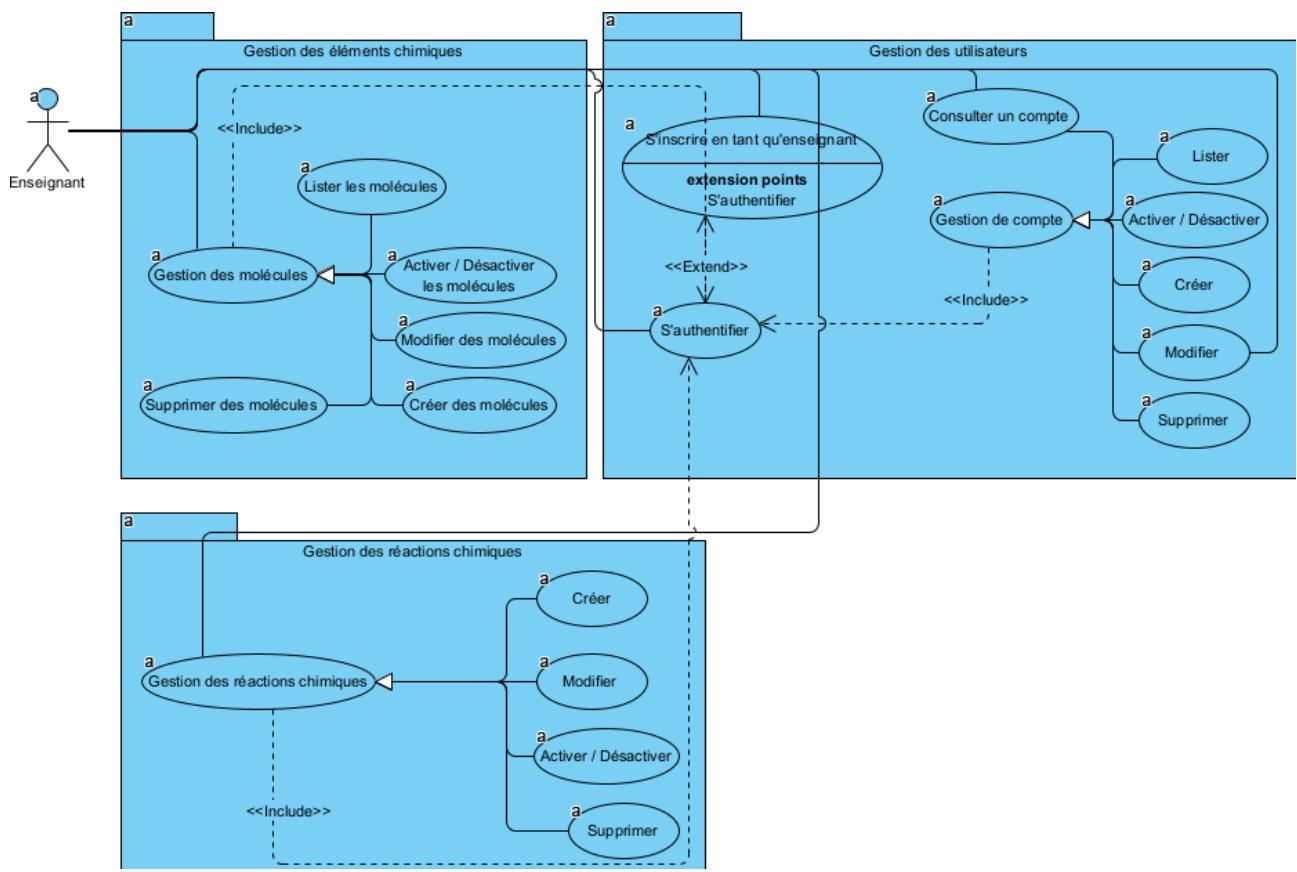


FIGURE 3.4 – Diagramme des cas d'utilisation d'un enseignant

La figure ci-dessus présente l'enseignant et tous ses cas d'utilisation de la plateforme regroupé par module. Nous constatons que le champ d'action d'un enseignant s'étend aux modules de gestion des utilisateurs, des éléments chimiques et des réactions chimiques.

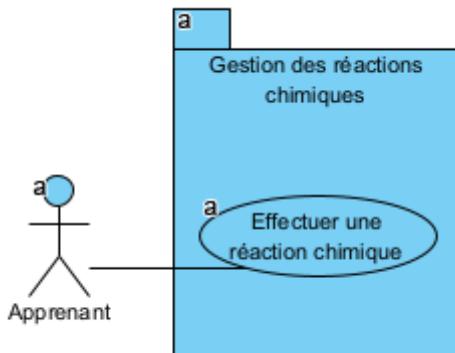


FIGURE 3.5 – Diagramme des cas d'utilisation d'un apprenant

La figure ci-dessus présente l'apprenant et tous ses cas d'utilisation de la plateforme regroupé par module. Nous constatons que le champ d'action d'un apprenant ne se limite qu'au module des réactions chimiques et uniquement au cas d'utilisation "effectuer une réaction chimique".

3.2.1.4 Description textuelle de quelques cas d'utilisation

— S'authentifier

- Sommaire d'identification
- Titre : s'authentifier
- Acteurs : Administrateur et enseignant
- Objectif : Il permet à l'acteur de s'identifier et de se connecter en saisissant son login et mot de passe.
- Description des scénarios
 - Précondition : L'acteur doit avoir ouvert l'application
 - Postcondition :
 - Acteur connecté.
 - Ouverture de l'interface utilisateur.
 - Scénario nominal
 1. L'acteur demande à ouvrir la page de connexion
 2. Le système affiche la page de connexion
 3. L'acteur saisit le nom d'utilisateur et son mot de passe
 4. Le système vérifie les données
 5. Le système connecte l'acteur et affiche l'interface utilisateur
 - Scénario alternatif
 - A. Erreur de connexion : nom d'utilisateur ou mot de passe non valide
Cet enchaînement démarre au point 4.
 - 5. Le système affiche un message d'erreur correspondant au problème d'identifiants incorrects.
 - B. Erreur de connexion : le compte est désactivé,
Cet enchaînement démarre au point 4.
 - 5. Le système affiche un message d'erreur correspondant au problème de compte désactive.

— Création d'élément chimique

- Sommaire d'identification
- Titre : Création d'élément chimique
- Acteurs : Administrateur et enseignant
- Objectif : Il permet à l'acteur de créer un élément chimique pour la plateforme.
- Description des scénarios
 - Précondition : Acteur connecté.
 - Postcondition :
 - Élément chimique crée dans la base de données.
 - Scénario nominal
 1. L'acteur demande du formulaire de création des éléments
 2. Le système affiche le formulaire de création des éléments
 3. L'acteur soumet le formulaire de création
 4. Le système vérifie les données
 5. Le système enregistre l'élément chimique et réinitialise le formulaire

- Scénario alternatif
 - A. Erreur informations soumises incorrects.
Cet enchaînement démarre au point 4.
 - 5. Le système affiche un message d'erreur pour informations soumises incorrect.
- **Effectuer une réaction**
 - Sommaire d'identification
 - Titre : Effectuer une réaction
 - Acteurs : Apprenant
 - Objectif : Il permet à l'acteur de réaliser une réaction chimique sur la plateforme.
 - Description des scénarios
 - Précondition : Application ouverte.
 - Postcondition :
 - L'interface de réalisation des réactions ouverte.
 - Scénario nominal
 1. L'acteur demande le formulaire d'identification des réactions.
 2. Le système affiche le formulaire d'identification des réactions.
 3. L'acteur saisi l'identifiant d'une réaction et soumet le formulaire.
 4. Le système recherche la réaction.
 5. Le système envoie de l'interface de réalisation de la réaction.
 6. L'acteur effectue un mélange et l'envoie.
 - Scénario alternatif
 - A. Erreur réaction introuvable.
Cet enchaînement démarre au point 4.
 - 5. Le système affiche un message d'erreur pour réaction introuvable.
 - B. Erreur réaction désactivée.
— 5. Le système affiche un message d'erreur pour réaction désactivée.
- **Création d'une réaction**
 - Sommaire d'identification
 - Titre : Création d'une réaction.
 - Acteurs : Administrateur et enseignant.
 - Objectif : Il permet à l'acteur de créer une réaction chimique pour la plateforme.
 - Description des scénarios
 - Précondition : Acteur connecté.
 - Postcondition :
 - Réaction chimique créée dans la base de données.
 - Scénario nominal
 1. L'acteur demande à créer une réaction.
 2. Le système retourne le formulaire de création.
 3. L'acteur envoie les informations sur la réaction.
 4. Le système demande une confirmation de création.
 5. L'acteur confirme la création.
 6. Le système enregistre la réaction.

- Scénario alternatif
 - A. L'acteur annule la création
Cet enchaînement démarre au point 4
 - 5. Le système annule la création.
Le scénario reprend au point 3.
- **Suppression d'une réaction**
 - Sommaire d'identification
 - Titre : Suppression d'une réaction.
 - Acteurs : Administrateur et enseignant.
 - Objectif : Il permet à l'acteur de supprimer une réaction sur la plateforme.
 - Description des scénarios
 - Précondition : Acteur connecté.
 - Postcondition :
 - La réaction est supprimée de la base de données de l'application.
 - Scénario nominal
 1. L'acteur demande de la liste des réactions.
 2. Le système retourne la liste des réactions.
 3. L'acteur sélectionne une réaction.
 4. Le système demande une confirmation de suppression.
 5. L'acteur confirme la suppression.
 6. Le système supprime la réaction.
 - Scénario alternatif
 - A. L'acteur annule la suppression
Cet enchaînement démarre au point 4
 - 5. Le système annule la suppression.
Le scénario reprend au point 3

3.2.1.5 Diagrammes de séquences

Les diagrammes de séquences sont la représentation graphique des interactions entre les acteurs et le système selon un ordre chronologique.

— S'authentifier

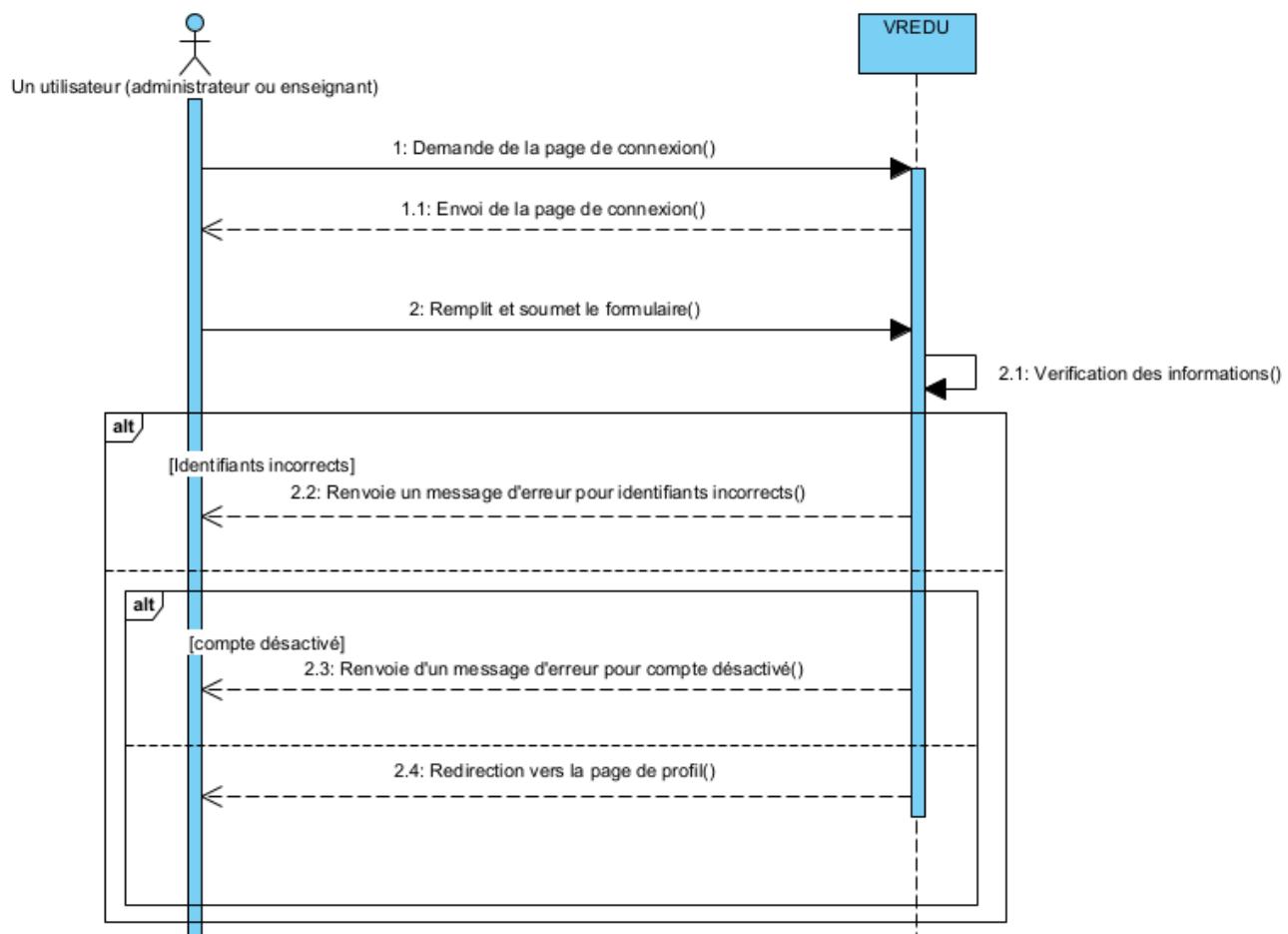


FIGURE 3.6 – Diagramme de séquence du cas d'utilisation s'authentifier

Ce diagramme fait une description détaillée du cas d'utilisation d'authentification qui permet à un utilisateur d'être reconnue par le système afin de lui accorder certains droits sur le système en fonction de son identité.

— Création d'élément chimique

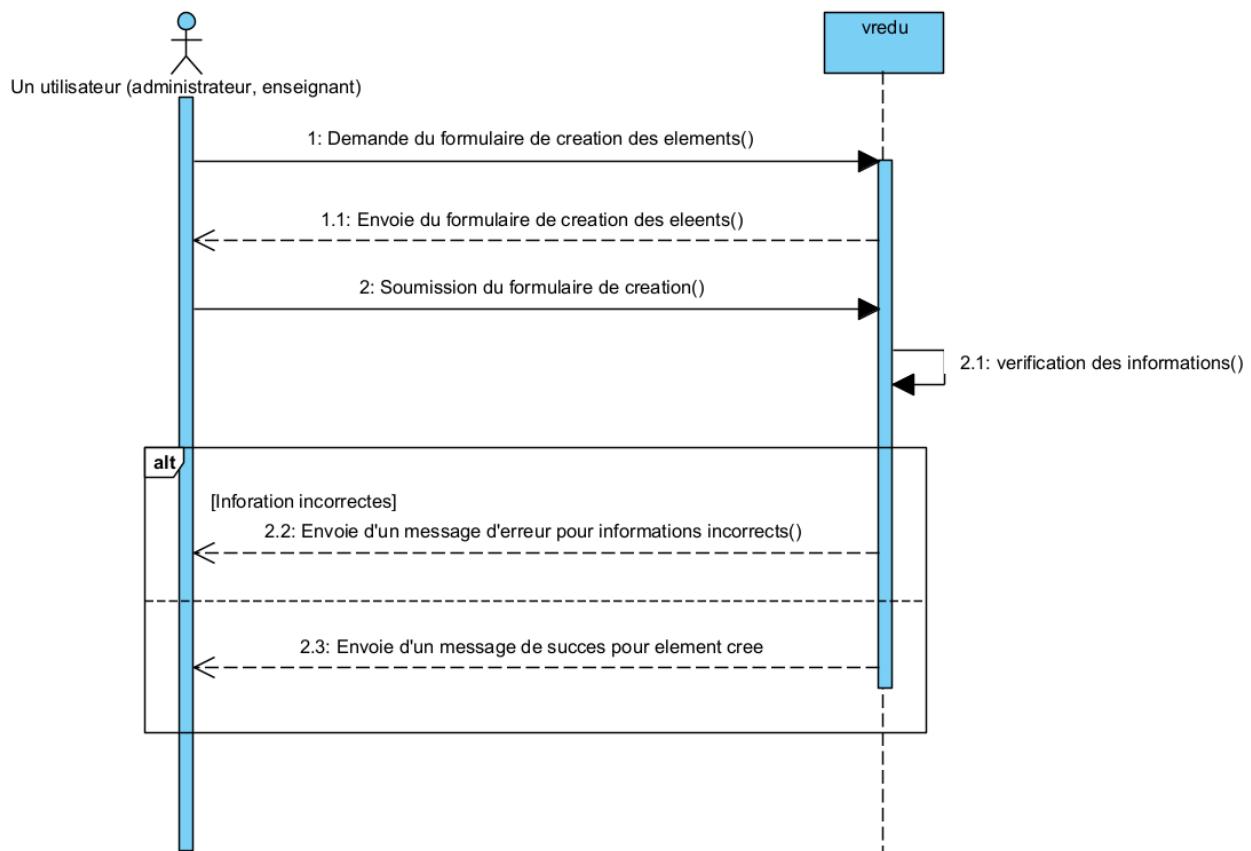


FIGURE 3.7 – Diagramme de séquence du cas d'utilisation création d'élément chimique

Ce diagramme fait une description détaillée du cas d'utilisation création d'un élément chimique, ce cas permet à un utilisateur connecté de créer un élément chimique pour une réaction chimique.

— Effectuer une réaction

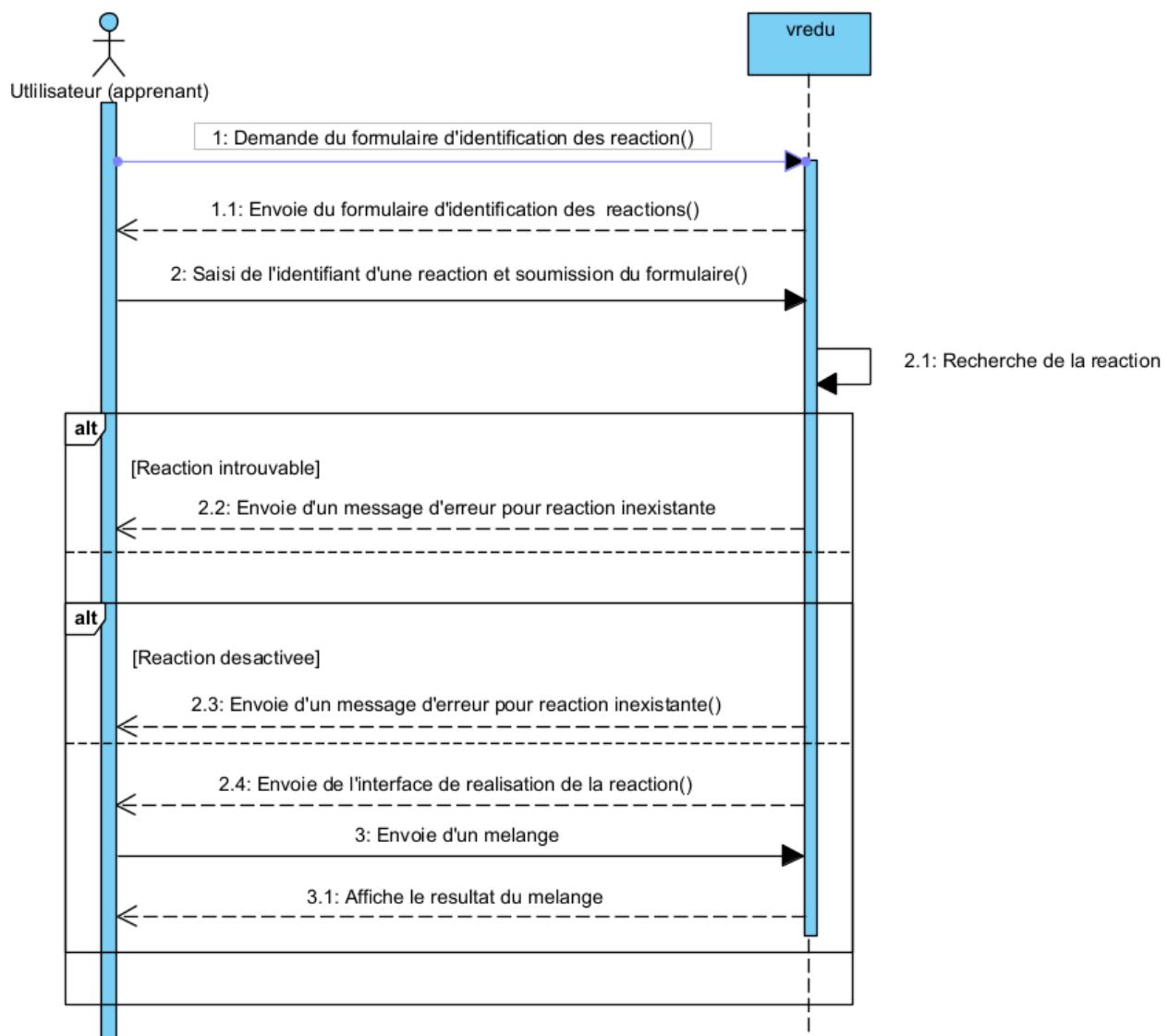


FIGURE 3.8 – Diagramme de séquence du cas d'utilisation effectuer une réaction

Ce diagramme fait une description détaillée du cas effectuer une réaction qui présente le processus de réalisation d'une réaction chimique sur le système par un apprenant.

3.2.1.6 Diagrammes d'activités

Les diagrammes d'activités permettent de mettre l'accent sur les traitements. Ils sont donc particulièrement adaptés à la modélisation du cheminement de flots de contrôle et de flots de données. Ils permettent ainsi de représenter graphiquement le comportement d'une méthode ou le déroulement d'un cas d'utilisation.

— Crée une réaction

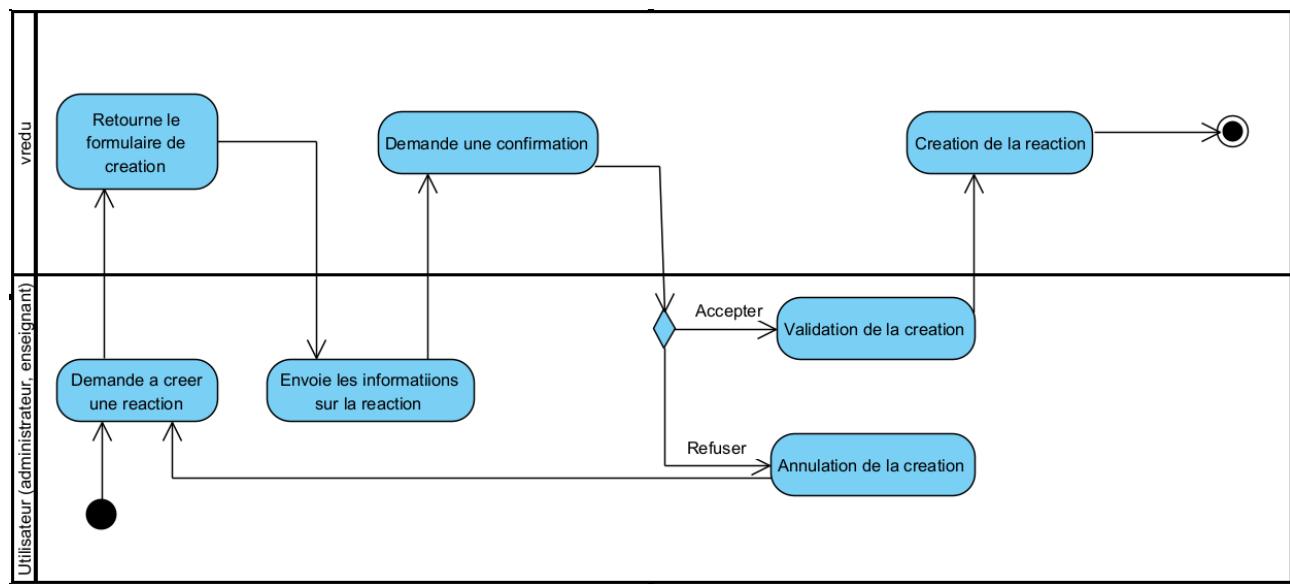


FIGURE 3.9 – Diagramme d'activité du cas d'utilisation créer une réaction

Ce diagramme fait une description détaillée du cas créer une réaction qui présente le processus de création d'une réaction chimique sur le système par un administrateur ou un enseignant.

— Suppression d'une réaction

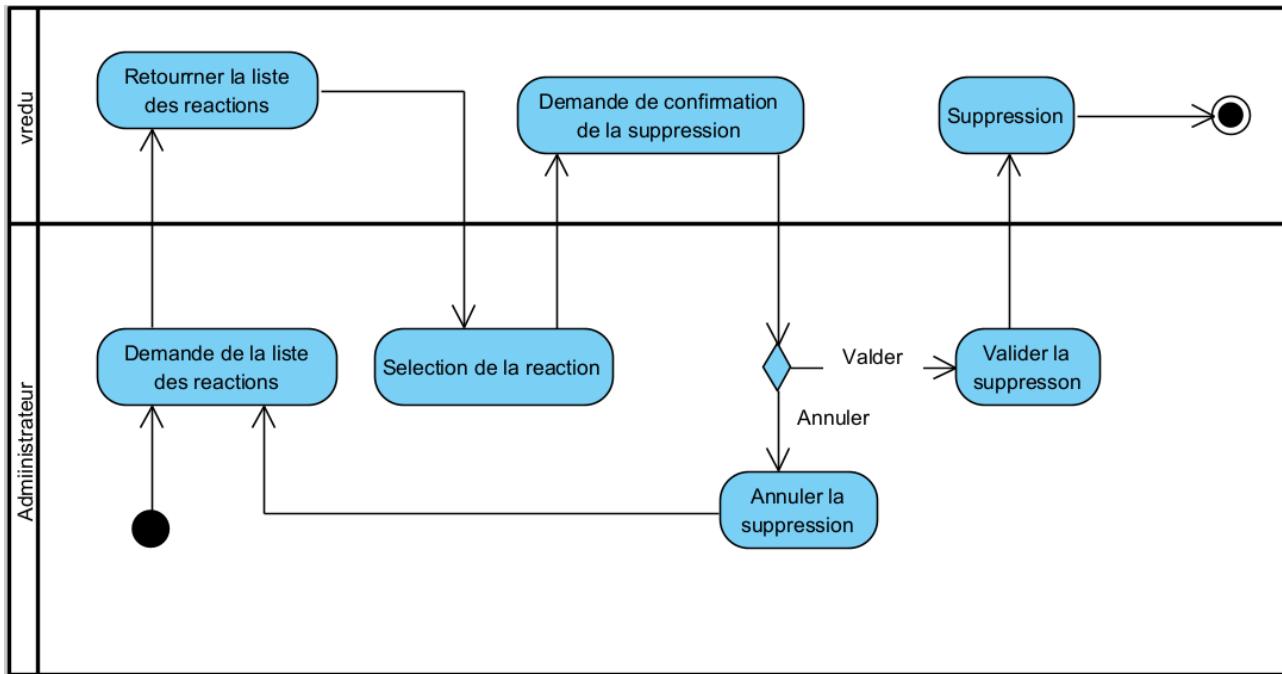


FIGURE 3.10 – Diagramme d'activité du cas d'utilisation suppression d'une réaction

Ce diagramme fait une description détaillée du cas suppression d'une réaction qui présente le processus de suppression d'une réaction chimique sur le système par un administrateur ou un enseignant.

3.2.1.7 Diagramme d'état transition

Un diagramme d'états-transitions est un type de diagramme comportemental en langage de modélisation unifié (UML) qui représente les transitions entre divers objets.

— Changement d'état des produits d'une réaction

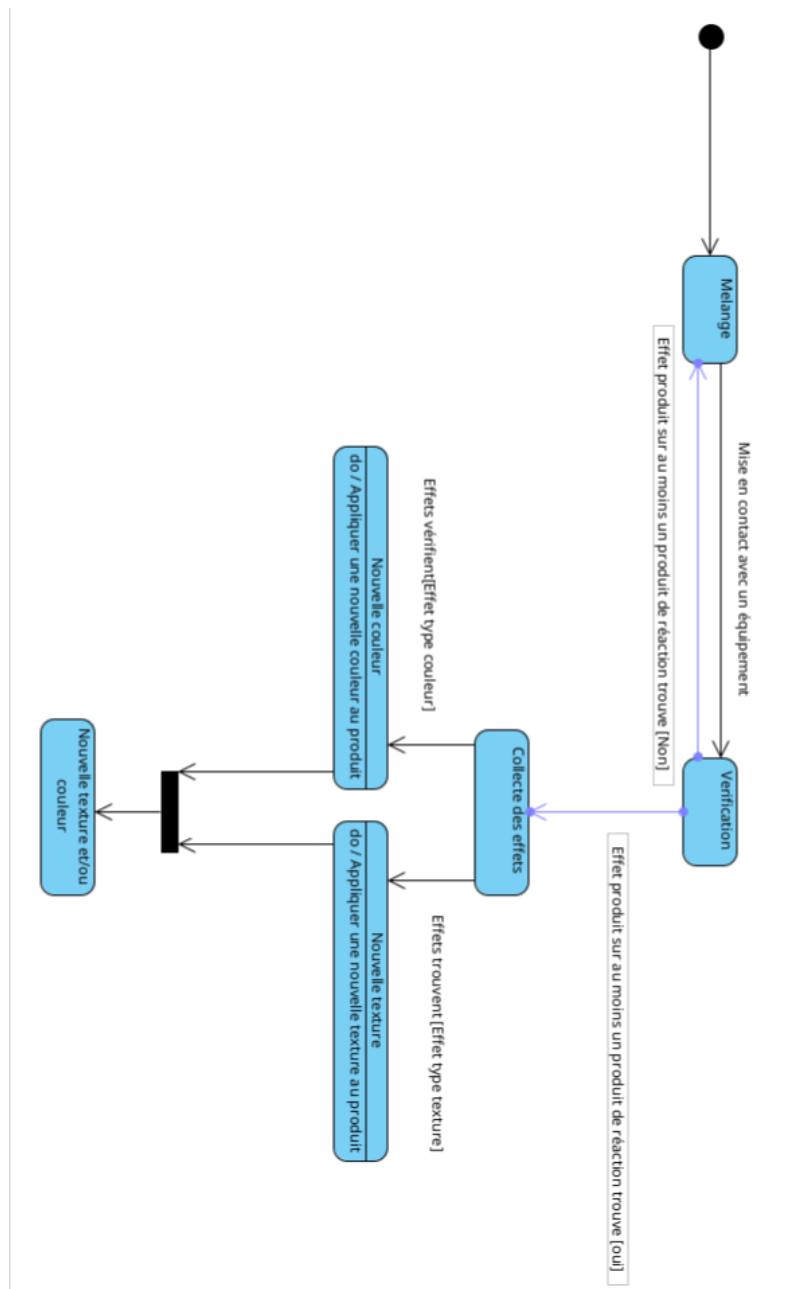


FIGURE 3.11 – Diagramme d'état transition du cas d'utilisation changement d'état des produits d'une réaction

Ce diagramme présente de façon détaillée le processus de changement d'état d'un produit de réaction lors d'une réaction chimique.

— Changement d'état d'authentification d'un utilisateur

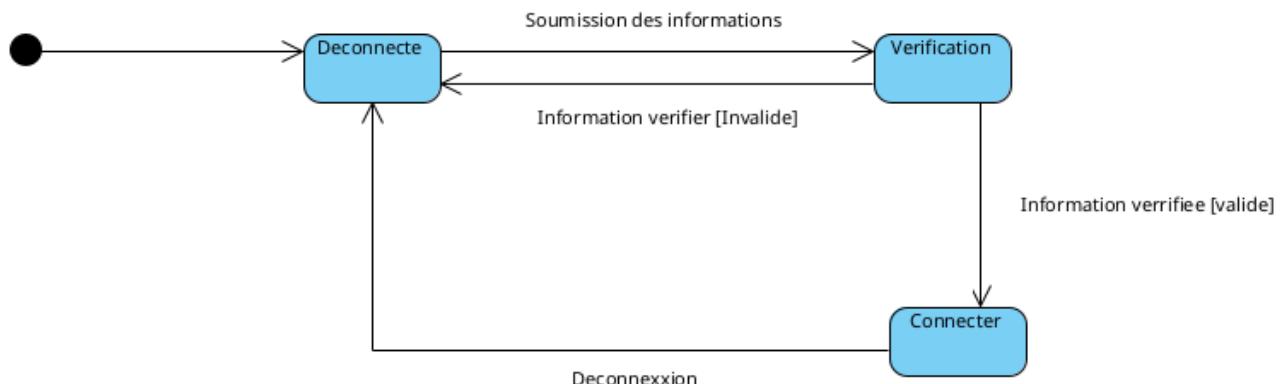


FIGURE 3.12 – Diagramme d'état transition du cas d'utilisation changement d'état d'authentification d'un utilisateur

Ce diagramme présente de façon détaillée le changement d'état d'authentification d'un utilisateur en fonction d'évènements déclenchés sur la plateforme.

3.2.2 Analyse non fonctionnelle

C'est l'ensemble des exigences qui ne concernent pas spécifiquement le comportement du système, mais plutôt identifient des contraintes internes et externes du système. Ainsi les principaux besoins non fonctionnels de notre système sont :

- Le code doit être clair, bien structuré pour permettre des futures évolutions du système ;
- L'ergonomie : la plateforme doit offrir des interfaces conviviales et faciles à utiliser ;
- La sécurité, le système doit garantir un grand niveau de protection des informations sur les utilisateurs.

3.2.3 Technique et Méthodes

3.2.3.1 Choix des techniques de développement

Plusieurs techniques peuvent être utilisées pour réaliser d'une plateforme web et 3D :

— Le code pur :

C'est ainsi que les informaticiens créent leur première application. Il est nécessaire d'avoir de bonnes connaissances en programmation. En effet, toute l'application est créée à partir de ligne de code complètement illisible pour un profane. Cette solution est longue, couteuse et rarement assez complète pour permettre une modification facile de l'application. Aujourd'hui, la plupart des développeurs ne créent plus leur application de A à Z, ils utilisent des cadres de travail prédéfinis.

— Les CMS (Content management system) pour le web :

Ce sont des aides à la création d'applications. Ils ne font pas tout à votre place, mais vous aident en grande partie. Il est très recommandé d'avoir des connaissances en informatique pour réaliser une application à partir d'un CMS. Certains sont en effet très compliqués à mettre en place. De plus, aucun hébergement n'est fourni avec le CMS. Il faudra

donc mettre en place un serveur HTTP pour héberger son application. Pour finir, la personnalisation du design d'une application sur un CMS est assez fastidieuse. Il sera presque à tous les cas obligatoires de mettre la main dans le code source de l'application. D'autres techniques, existent notamment les Outils en ligne.

— **Framework et librairies :**

Il existe également des Frameworks et des librairies (bout de code écrit permettant de faciliter le travail du développeur), qui est l'une des méthodes les plus utilisées dans le monde du développement, nous avons donc opté pour cette méthode et nous allons choisir un Framework parmi ceux existant.

3.2.3.2 Choix du Framework

ASP.net pour le **backend**, **React** pour le **frontend** et **unreal engine** pour la **3D**. Comme SGBD, nous allons utiliser **PostgreSQL** (le SGBD utilisé par l'entreprise).

3.3 Conception du système

3.3.1 Diagramme de classe

Le diagramme de classes est un schéma utilisé pour présenter les classes et les interfaces des systèmes ainsi que leurs relations.

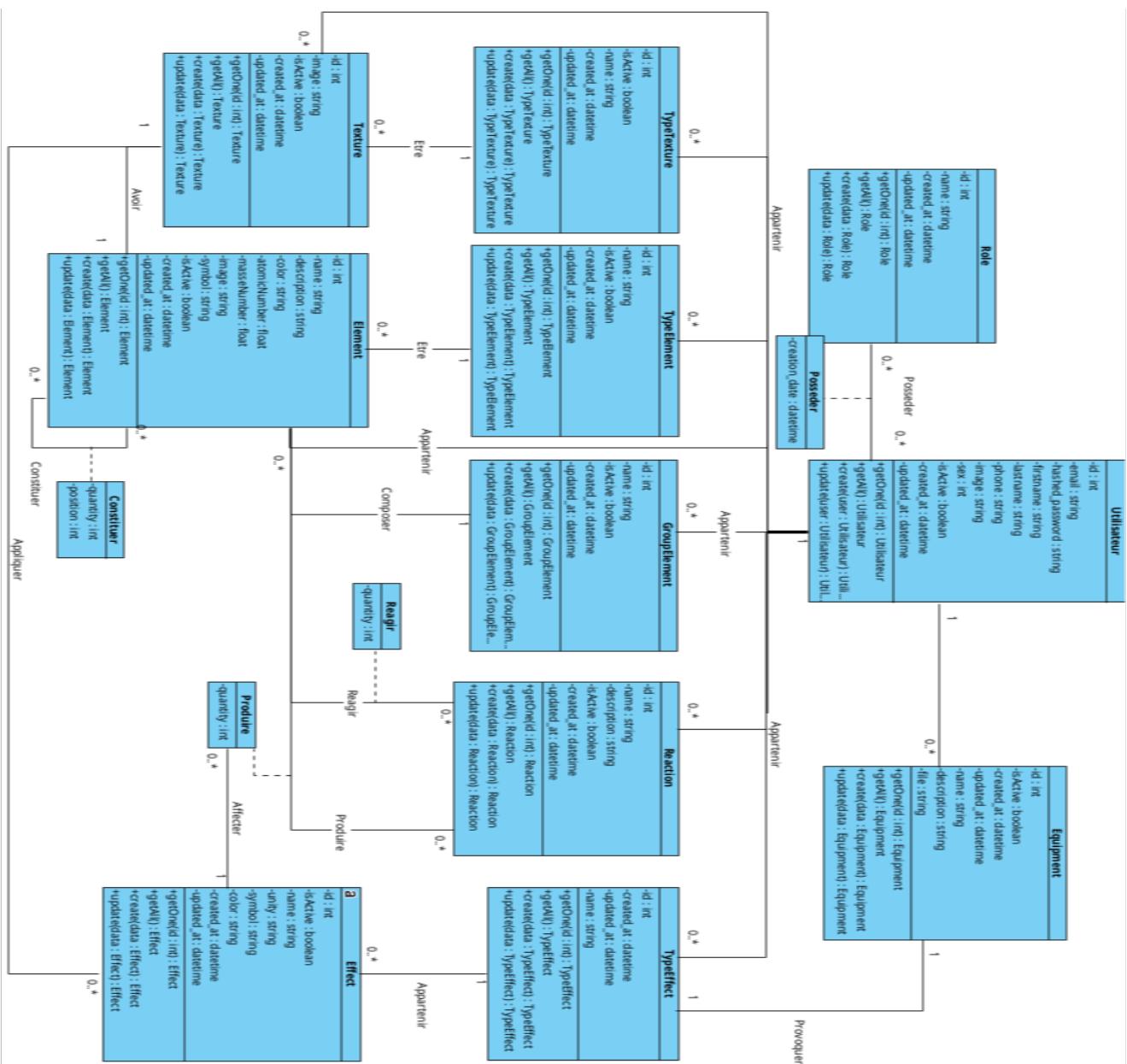


FIGURE 3.13 – Diagramme de classe du système

Ce diagramme présente les classes et leurs interactions dans le système.

3.3.2 Diagramme de déploiement

En UML, un diagramme de déploiement est une vue statique qui sert à représenter l'utilisation de l'infrastructure physique par le système et la manière dont les composants du système sont répartis ainsi que leurs relations entre eux. Les éléments utilisés par un diagramme de déploiement sont principalement les nœuds, les composants, les associations et les artefacts. Les caractéristiques des ressources matérielles physiques et des supports de communication peuvent être précisées par stéréotype.

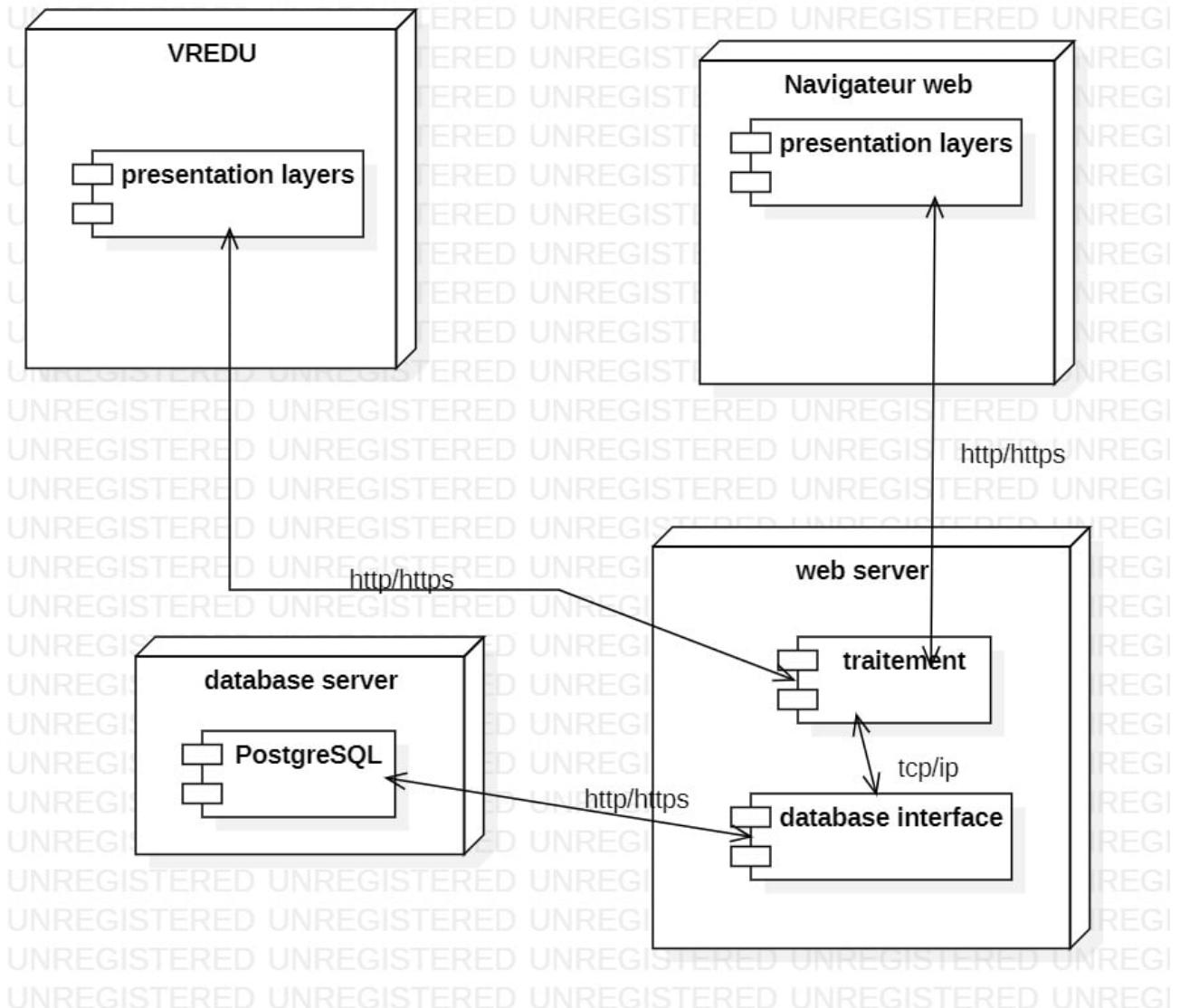
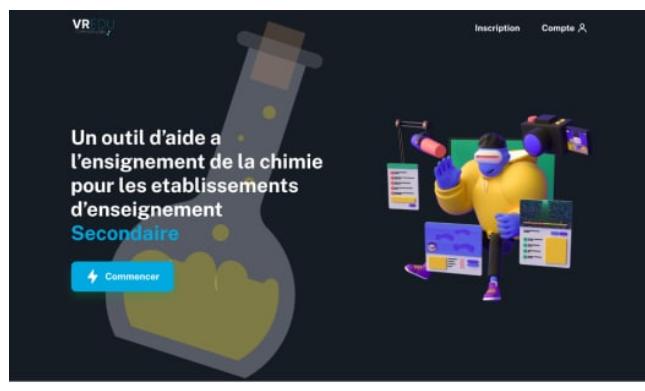


FIGURE 3.14 – Diagramme de déploiement du système

3.3.3 Quelques maquettes d'IHM (Interface Homme Machine)

3.3.3.1 Page d'accueil

Cette page permet de décrire la solution aux enseignants qui souhaitent la découvrir.



Qui sommes nous ?

Application permettant la création de réactions chimiques pour pratique dans les écoles d'enseignement secondaire permettant non seulement la création des réactions avec ses réactifs et ses produits tout en y appliquant des effets sous de nombreuses conditions que se soit de **TEMPS**, **TEMPERATURE**... Cela dans des **EQUIPEMENTS** de laboratoire virtuel.

Deux applications ?



Professeur des lycées ou collèges

Interface web permettant la création des éléments chimiques tels que Dioxyde de carbone, Eau,..., de réaction chimiques ou vous ferez interagir ces éléments pour un cours pratique délocalisé et en sécurité. [Inscrivez-vous ici.](#)



Lycéens et collégiens

Effectuez les réactions chimiques programmées par votre enseignant sur un casque de réalité virtuelle où que vous soyez et à tout moment sans risque d'accidents lors des expérimentations. À télécharger sur l'[Oculus Quest Store](#)



50 éléments chimiques modélisés en 3D sur 118

30 textures différentes à appliquer aux molécules ou aux produits de réactions

Un nombre ∞ de combinaisons entre éléments chimiques

Contacts

- Email : montgo-technologie@gmail.com
- Tel : 090961056 / 620080830

FIGURE 3.15 – IHM de la page d'accueil

Il y apparaît clairement un entête de page avec le logo et les différents liens d'inscription et de connexion. Les sections suivantes décrivent la solution, présentent ses statistiques et les contacts en cas de questions ou de problèmes.

3.3.3.2 Page d'inscription des enseignants

Cette page permet la création de compte enseignant en y renseignant les informations personnelles de l'utilisateur.

The screenshot shows the 'Inscription' (Registration) page of the VR EDU Chemistry lab application. At the top left is the VR EDU logo with the tagline 'Chemistry lab'. To the right, the title 'Creer vos travaux pratique de chimie' is displayed above a stylized illustration of a female character with red hair holding a tablet, standing next to a control panel with various knobs and buttons. The main form area is titled 'Inscription' and contains fields for personal information: 'Nom' (Name), 'Prenom' (First Name), 'Email', 'Nom d'utilisateur' (Username), 'Mot de passe' (Password), and 'Confirmation de mot de passe' (Password Confirmation). A blue 'Inscription' button is located at the bottom right of the form. A small note above the first two fields says 'Entrez vos informations personnelles ici.'

FIGURE 3.16 – IHM de la page d'inscription des enseignants

Il y apparaît le formulaire de création de compte avec les différents champs remplir avant la soumission.

3.3.3.3 Page de connexion des enseignants et administrateurs

Cette page permet à tous les utilisateurs d'accéder à l'application en utilisant un login et un mot de passe.

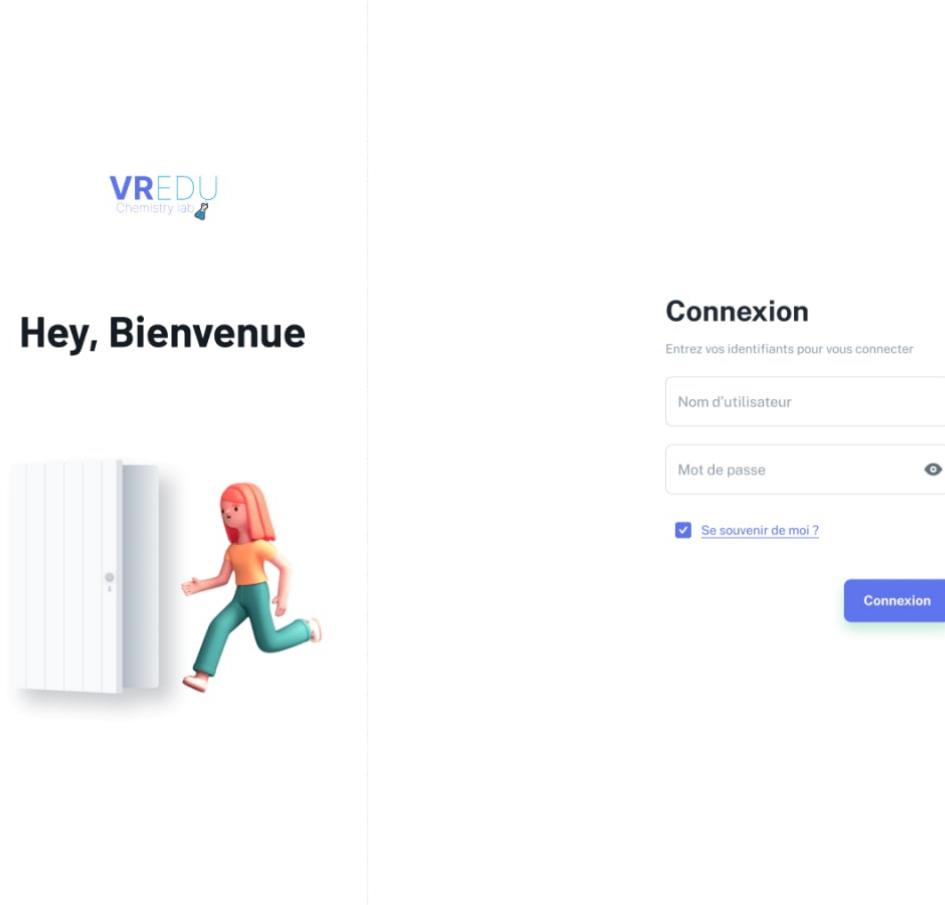


FIGURE 3.17 – IHM de la page de connexion des enseignants et des administrateurs

Il y apparaît le formulaire de connexion avec les champs pour le nom d'utilisateur et le mot de passe à remplir avant soumission. Deux catégories d'utilisateurs peuvent utiliser cette interface, à savoir les administrateurs et les enseignants.

3.3.3.4 Page espace personnel des enseignants et des administrateurs

Cette page permet à l'utilisateur d'avoir un aperçu de ses réactions.



Carlota Monteiro
Admin

Tableau périodique

1 H Hydrogène	2 He Hélium
3 Li Lithium	4 Be Béryllium
5 B Boron	6 C Carbone
7 N Nitrogen	8 O Oxygène
9 F Fluor	10 Ne Néon
11 Na Sodium	12 Mg Magnésium
13 Al Aluminium	14 Si Silicium
15 P Phosphore	16 S Soufre
17 Cl Chlore	18 Ar Argon
19 K Potassium	20 Ca Calcium
21 Sc Scandium	22 Ti Titane
23 V Vanadium	24 Cr Chromium
25 Mn Manganèse	26 Fe Fer
27 Co Cobalt	28 Ni Nickel
29 Cu Cupronickel	30 Zn Zinc
31 Ga Gallium	32 Ge Gérasme
33 As Antimoine	34 Se Sélénium
35 Br Bromure	36 Kr Krypton
37 Rb Rubidium	38 Sr Strontium
39 Y Thétium	40 Zr Zirconium
41 Nb Nioberium	42 Mo Molibdène
43 Tc Technetium	44 Ru Rhodium
45 Rh Rhodium	46 Pd Platine
47 Ag Argent	48 Cd Cadmium
49 Cd Cadmium	50 In Indium
51 Sn Stannum	52 Te Tellure
53 Cs Calcium	54 Xe Xénon
55 Ba Barium	56 Ra Radon
57 Fr Francium	58 Ac Actinium
59 Pr Protactinium	60 Nd Neodyme
61 Pm Prométhium	62 Sm Samarium
63 Eu Europium	64 Gd Gadolinium
65 Tb Thulium	66 Dy Dysprosium
67 Ho Holmium	68 Er Erbium
69 Tm Thulium	70 Yb Ytterbium
71 Lu Lutécium	
72 Hf Hafnium	73 Ta Taïtan
74 W Wolfram	75 Re Rhenium
76 Os Osmium	77 Ir Iridium
78 Pt Platine	79 Au Or
80 Hs Hassium	81 Bh Bismuth
82 Mt Moltbrium	83 Ds Darmstadtium
84 Rg Ronguéon	85 Cn Cernium
86 Cf Californium	87 Bk Bérgéinium
88 Es Einstenium	89 Fm Fermium
90 Md Mendelevium	91 No Néptunium
92 Lv Léridium	93 Ts Thénardium
94 Lv Léridium	95 Og Oganesson

GENERAL

Mon espace

Réactions

Éléments

UTILISATEUR

Compte

Déconnexion

ADMINISTRATION

Utilisateurs

Atomes

- Types

- Groupes

- Element

Textures

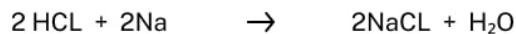
- Types de texture
- Mes textures

Effets

Equipements

Mes réactions Ajouter une réaction

Fabrication d'une molécule de NaCl



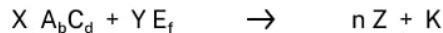
Titre de la réaction



Titre de la réaction



Titre de la réaction



Titre de la réaction



Mes éléments Ajouter un élément

- HCL
- NaCL
- ...

FIGURE 3.18 – IHM de la page espace personnel des enseignants et des administrateurs

Il y apparaît la liste des éléments chimiques du tableau périodique et la liste des réactions de l'utilisateur connecté.

3.3.3.5 Interface du laboratoire en vue de dessus

Cette interface représente une vue de dessus du laboratoire virtuelle où les apprenants effectueront leurs réactions.



FIGURE 3.19 – IHM du laboratoire en vue de dessus

Il y apparaît un local vu du dessus et présente l'environnement extérieur du laboratoire où auront lieu les réactions.

3.3.3.6 Interface du laboratoire en vue de face

Cette interface représente une vue de face du laboratoire virtuelle où les apprenants effectueront leurs réactions.

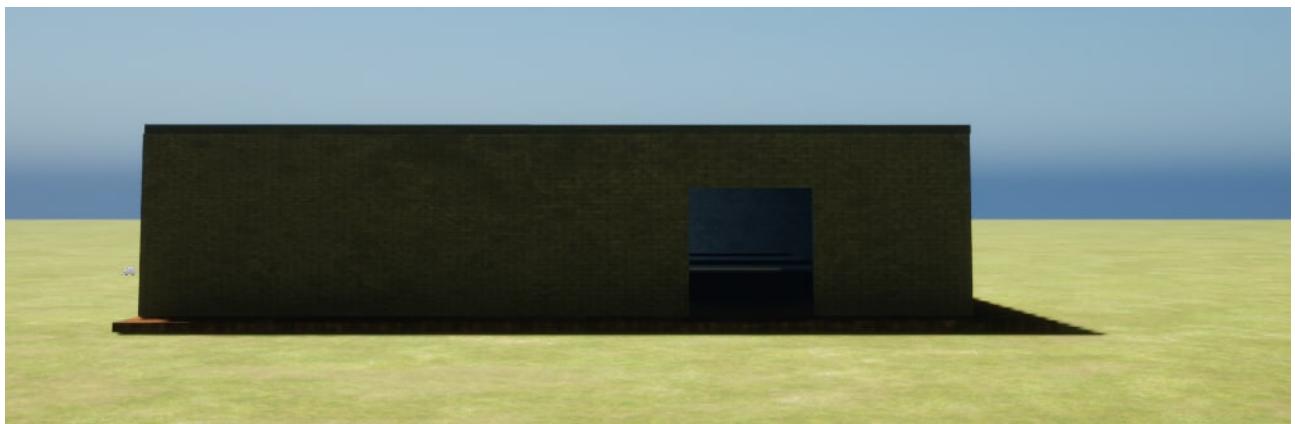


FIGURE 3.20 – IHM du laboratoire en vue de face

Il y apparaît un local vu de face et présente l'environnement extérieur du laboratoire où auront lieu les réactions.

3.3.3.7 Interface du laboratoire intérieur vue du fond

Cette interface représente une vue intérieure du laboratoire vu du fond de la salle.



FIGURE 3.21 – IHM du laboratoire intérieur vue du fond

Il y apparaît une vue du local, et est présenté l'environnement intérieur du laboratoire avec les tables où se dérouleront les réactions.

3.3.3.8 Interface du laboratoire intérieur vue de l'apprenant

Cette interface représente le point de vue d'un apprenant qui utilise l'application.

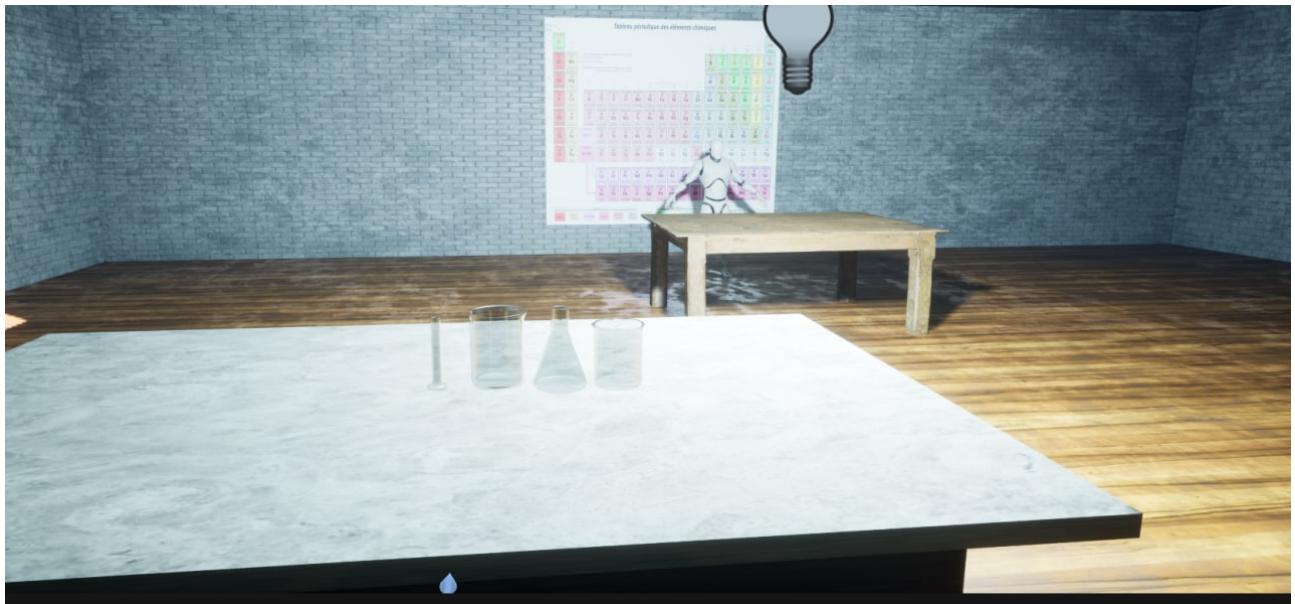


FIGURE 3.22 – IHM du laboratoire intérieur vue de l'apprenant

Il y apparaît la table de l'apprenant avec la verrerie des réactions disposée dessus.

3.3.3.9 Interface du laboratoire intérieur vue de l'enseignant

Cette interface représente le point de vue d'un enseignant qui utilise l'application.



FIGURE 3.23 – IHM du laboratoire intérieur vue de l'enseignant

Il y apparaît les tables de réaction des différents apprenants dans le laboratoire virtuel.

Chapitre 4

Implémentation de la solution et résultats

Dans ce chapitre, nous ferons un tour d'horizon sur l'ensemble des technologies utilisées afin de mettre en œuvre la solution et montrerons comment nous comptons déployer notre solution. Enfin, nous présenterons quelques résultats obtenus par implémentation.

4.1 Outils et technologies

De la conception à la réalisation de notre application, nous avons eu à utiliser les outils et technologies suivants :

- **Unreal Engine 5** : qui est un moteur de rendu 3D utilisé dans la création de jeux et la simulation d'environnement 3D utilisant le langage de programmation c++.
- **React & React Dom** : qui est un Framework JavaScript très populaire permettant la création d'interfaces utilisateur web simplement et rapidement, il offre une grande documentation et possède une communauté très active.
- **ASP.NET core web api** : est un Framework C# permettant la création d'api REST développé par Microsoft.
- **PostgreSQL** : C'est un système de gestion de base de données relationnelle et objet. C'est un outil libre disponible selon les termes d'une licence de type BSD. Ce système est concurrent d'autres systèmes de gestion de base de données, qu'ils soient libres ou propriétaires ;
- **Git & GitHub** : Git est un logiciel de gestion de versions décentralisé. C'est un logiciel libre créé par Linus Torvalds, auteur du noyau Linux, et distribué selon les termes de la licence publique générale GNU version 2. GitHub est un logiciel libre de forge basé sur git proposant les fonctionnalités de wiki, un système de suivi des bugs, l'intégration continue et la livraison continue ;
- **Rider** : C'est un environnement de développement .NET multiplateforme de chez jet-brains basé sur la plateforme IntelliJ et ReSharper.
- **Webstorm** : C'est un IDE pour les langages Web, développé par l'entreprise JetBrains et basé sur la plateforme IntelliJ IDEA ;
- **Docker** : C'est un IDE pour les langages Web, développé par l'entreprise JetBrains et basé sur la plateforme IntelliJ IDEA ;
- **Jira** : est un système de suivi de bugs, de gestion des incidents et de gestion de projets développé par Atlassian.
- **Visual paradigm** : C'est un outil UML CASE prenant en charge UML 2, SysML et la notation de modélisation de processus métier (BPMN) d'Object Management Group (OMG). Outre la modélisation, il offre des fonctionnalités de génération de rapports et d'ingénierie de code, y compris la génération de code.
- **Gantt Project** : est un logiciel libre de gestion de projet écrit en Java.

4.2 Résultats

4.2.1 Édition du profil utilisateur

4.2.1.1 Interface d'édition du profil utilisateur

La figure ci-dessous présente l'interface qui permettra à un utilisateur, en occurrence un administrateur ou un enseignant, de modifier ses informations personnelles.

The screenshot shows a user interface for managing personal information. On the left, there is a sidebar with a logo, a search bar, and a user profile for 'admin ADMIN Administrateur'. Below this are several menu items: 'Comptes' (selected), 'Reactions', 'Éléments Chimiques', 'Utilisateurs', 'Textures', and 'Equipements'. The main content area is titled 'Informations personnelles' and has a sub-section 'General'. It features a placeholder 'Modifiez vos informations' with the instruction 'Saisissez vos informations personnelles ci-dessous'. There is a placeholder image for a profile picture with the button 'Ajouter Une Photo'. The form fields include: 'Prénom' (admin), 'Nom' (admin), 'Nom d'utilisateur' (admin), 'Adresse email' (admin@mail.com), 'Numéro de téléphone' (690981056), 'Date de naissance' (09/17/2022), and 'Sexe' (Feminin). A blue 'Enregistrer' button is at the bottom right.

FIGURE 4.1 – IHM formulaire d'édition du profil utilisateur

Il y apparaît à gauche les différents menus disponibles de l'application, et à droite une zone de formulaire permettant la modification des informations d'un utilisateur.

4.2.1.2 Code source backend

Le code backend du modèle d'utilisateur permet la création d'une représentation sous forme de classe d'un utilisateur, représentation qui sera utile à notre ORM (entity Framework) pour stocker, modifier, lire et supprimer un utilisateur en base de données.

La figure ci-dessous montre la représentation sous forme de classe d'un utilisateur avec ses différents attributs, à savoir la date de création de son profil, la dernière date de modification de son profil, son prénom, son nom, sa date de naissance et plusieurs autres critères nécessaires à son rôle dans l'application.

```
public class User : IdentityUser<int>, IModelImage
{
    public DateTime CreatedDate { get; set; }

    public DateTime UpdatedDate { get; set; }
    public bool IsActive { get; set; }

    public string LastName { get; set; } = string.Empty;

    public string FirstName { get; set; } = string.Empty;

    public DateTime BirthDate { get; set; }

    public SexType Sex { get; set; }

    [JsonIgnore] public override string PasswordHash { get; set; } = string.Empty;

    public List<ElementGroup> ElementGroups { get; set; }
    public virtual List<Element> Elements { get; set; }
    public virtual List<ElementType> ElementTypes { get; set; }

    public virtual List<Texture> Textures { get; set; }
    public virtual List<Reaction> Reactions { get; set; }

    public virtual List<TypeEffect> TypeEffects { get; set; }
    public virtual List<Equipment> Equipments { get; set; }
    public string? Image { get; set; } = string.Empty;
}
```

FIGURE 4.2 – Code backend du modèle d'utilisateur

Le code backend d'un contrôleur permet de définir des réponses en fonction de requêtes effectuées par un client sur un serveur.

Le code ci-dessous présente le backend du contrôleur utilisateur, plus précisément la fonctionnalité de la modification des informations d'un utilisateur.

```
[HttpPost]
public async Task<ActionResult<User>> UserUpdate([FromForm] UserUpdateDto dto)
{
    var userId = User.FindFirstValue(ClaimTypes.NameIdentifier);
    var user = await _userManager.FindByIdAsync(userId);
    if (user is null)
        return NotFound(new Response
    {
        Message = "Utilisateur inexistant !",
        Status = "Error"
    });

    foreach (var prop in dto.GetType().GetProperties())
    {
        if (dto.OldPassword is not null && dto.NewPassword is not null)
        {
            if (dto.Image is not null) ...
            var result = await _userManager.UpdateAsync(user);
            if (!result.Succeeded) ...
            return Ok(user);
        }
    }
}
```

FIGURE 4.3 – Code backend du cas d'utilisation de l'édition du profil utilisateur

Le test unitaire est une procédure permettant de vérifier le bon fonctionnement d'une partie précise d'un logiciel ou d'une portion d'un programme.

Le code ci-dessous présente le test unitaire de la fonctionnalité d'édition du profil utilisateur.

```
[Fact]
public async Task UpdateUserTest()
{
    _client.DefaultRequestHeaders.Authorization = new AuthenticationHeaderValue(scheme: "Bearer", parameter: "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiIxMjM0NTY3ODkwIiwibmFtZSI6IkJlY2xvbmUgUGVyc29yIiwiYWRkIjoiMSIsImV4cCI6MTUwOTUwNzQwNiwidHlwZSI6InByb2ZpbGUifQ.eyJzbWFnZS5sb3dlcl1heCJ9");
    var u = new UserUpdateDto
    {
        Email = "test@test.com",
        Sex = SexType.F,
        BirthDate = DateTime.Now,
        FirstName = "Test",
        LastName = "Test",
        UserName = "Test",
        OldPassword = "admin",
        NewPassword = "test"
    };
    var serializeString = JsonSerializer.Serialize(u);
    var data = new StringContent(serializeString, Encoding.UTF8, mediaType: "application/json");
    var response = await _client.PutAsync(requestUri: "/api/user", data);
    var contentString = await response.Content.ReadAsStringAsync();
    var updatedUser = JsonSerializer.Deserialize<User>(contentString);
    Assert.Equal(expected: HttpStatusCode.OK, actual: response.StatusCode);
    Assert.NotNull(updatedUser);
    Assert.IsType<User>(updatedUser);
    Assert.Equal(expected: u.Email, actual: updatedUser?.Email);
}
```

FIGURE 4.4 – Code backend du test unitaire du cas d'utilisation de l'édition du profil utilisateur

La figure ci-dessous présente le résultat du test unitaire du cas d'utilisation de l'édition du profil utilisateur.

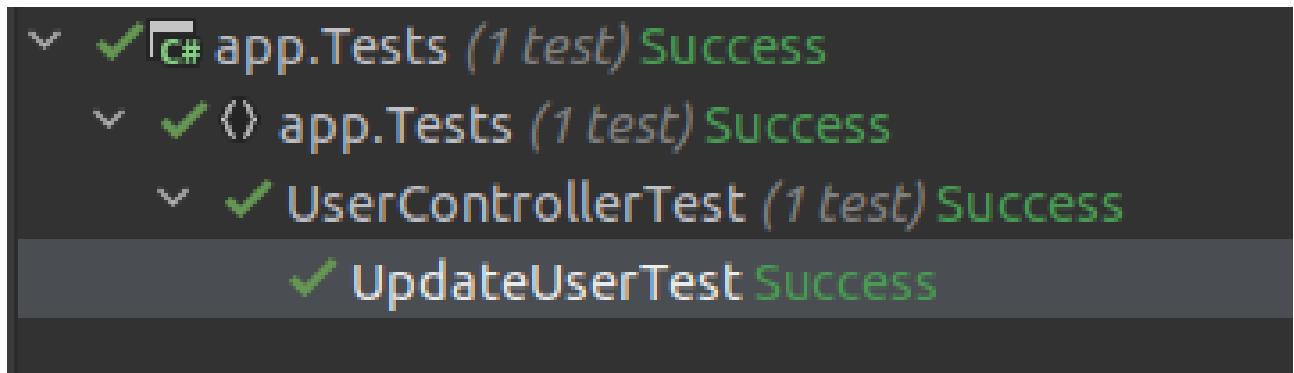


FIGURE 4.5 – Résultat du test unitaire du cas d'utilisation de l'édition du profil utilisateur

4.2.1.3 Code source frontend

Le code source frontend est celle qui permet la réalisation de l'interface que l'utilisateur verra par le billet d'un navigateur web.

La figure ci-dessous présente le code qui a permis la création de l'interface d'édition du profil utilisateur.

```
export default function Settings() {
  const [value, setValue] = useState(0);

  const tabHeaders = useMemo(() => [
    {name: "General", icon: "bx:user-rectangle"},
    {name: "Mot de passe", icon: "bx:key"},
  ], []);

  const handleChange = (event, newValue) => {
    setValue(newValue);
  };

  return <div>
    <Page title="Setting : user">
      <Container>
        <Typography variant="h4" sx={{mb: 5}}>
          Informations personnel
        </Typography>

        <Stack direction="row" flexWrap="wrap-reverse" alignItems="center" sx={{mb: 5}}>
          <Box sx={{width: "100%}}>
            <Box>
              <Tabs value={value} onChange={handleChange} aria-label="basic tabs example">
                {tabHeaders.map((item, k) => (
                  <Tab value={item.name}>
                    {item.icon}
                  </Tab>
                ))}
              </Tabs>
            </Box>
            <Card sx={{p: 3}}>
              <TabPanel value={value} index={0}>
                <GeneralForm/>
              </TabPanel>
              <TabPanel value={value} index={1}>
                <PasswordForm/>
              </TabPanel>
            </Card>
          </Box>
        </Stack>
      </Container>
    </Page>
  </div>
}
```

FIGURE 4.6 – Code frontend du cas d'utilisation de l'édition du profil utilisateur

4.2.2 Création des éléments chimiques

4.2.2.1 Interface de création des éléments chimiques

Le formulaire sur la figure ci-dessous permet la création d'un élément chimique (Atome et molécule) et cela en fonction de notre rôle, seul l'administrateur a le droit de créer des atomes tandis que la création des molécules est autorisée à tous les utilisateurs.

The screenshot shows a user interface for creating chemical elements. On the left, there's a sidebar with a user profile for 'admin ADMIN Administrateur'. The main area is titled 'Création d'une molécule ou d'un atome' with a sub-instruction 'Saisissez les informations sur l'équipement à créer ci-dessous !'. There are two tabs: 'Atome' (selected) and 'Molecule'. The 'Atome' tab contains fields for 'Nom de l'atome' (Name of the atom) and 'Description'. Below these is a placeholder image icon with the text 'Ajouter Une Photo' (Add a photo). A note says 'Aucun element trouve veillez en creer' (No element found, please create) and has a 'Creer' button. Under 'Texture', it says 'Formule chimique' (Chemical formula) and 'Aucun atome selectionne' (No atom selected). Below that is a section 'Choix des atomes' (Choice of atoms) with a dropdown menu set to 'Atomes'. At the bottom is a large blue 'Enregistrer' (Register) button.

FIGURE 4.7 – IHM formulaire de création des éléments chimiques

4.2.2.2 Code source backend

La figure ci-dessous présente la définition du modèle d'élément chimique qui sera stocké en base de données.

```
public class Element : BaseModel, IModelImage
{
    public string Name { get; set; } = string.Empty;
    public string? Symbol { get; set; }
    public int? MassNumber { get; set; }
    public int? AtomicNumber { get; set; }

    public User User { get; set; }

    public ElementType? Type { get; set; }

    public ElementGroup? Group { get; set; }

    public Texture Texture { get; set; }

    public List<Product> Products { get; set; }

    public string Description { get; set; } = string.Empty;

    [InverseProperty("Parent")]
    public List<ElementChildren> Children { get; set; }

    [InverseProperty("Children")]
    public List<ElementChildren>? Parents { get; set; }

    public string Image { get; set; } = string.Empty;
}
```

FIGURE 4.8 – Code backend du modèle d'élément chimique

La figure ci-dessous présente la partie du code backend responsable de la création d'un élément chimique.

```
[HttpPost]
[Authorize(Roles = UserRole.Admin)]
public async Task<ActionResult<Element>> Create([FromForm] RegisterElementDto dto)
{
    var userId = User.FindFirstValue(ClaimTypes.NameIdentifier);
    var user = await _context.Users.FindAsync(int.Parse(userId));

    if (user is null) return Unauthorized();

    var group = await _context.ElementGroups.FindAsync(dto.GroupId);
    if (group is null)-
        var type = await _context.ElementTypes.FindAsync(dto.TypeId);
        if (type is null)-
            var texture = await _context.Textures.FindAsync(dto.TextureId);
            if (texture is null)-
                var element = new Element
                {-
                };
                if (dto.Image is not null)
                    string? path = null;
                    try|
                    {
                        path = await FileManager.CreateFile(dto.Image, user.UserName, _env, new[] { "elements" });
                        element.Image = path ?? string.Empty;
                    }
                    catch (Exception e)
                    {
                        FileManager.DeleteFile(path ?? "", _env);
                        return BadRequest(new Response
                        {
                            Status = "Error", Errors = new Dictionary<string, string>
                            {
                                { "image", e.Message }
                            }
                        });
                    }
                }
            }
        }
    }
}
```

FIGURE 4.9 – Code backend du cas d'utilisation de la création d'élément chimique

La figure ci-dessous présente le code du test unitaire de la fonctionnalité de création d'élément chimique.

```
[Fact]
public async Task CreateElementTest()
{
    _client.DefaultRequestHeaders.Authorization = new AuthenticationHeaderValue(scheme: "Bearer", parameter: "eyJhbGciOi
    var element = new RegisterElementDto
    {
        Name = "Test",
        Symbol = "S",
        Description = "Test",
        MassNumber = 1,
        AtomicNumber = 1,
        GroupId = 1,
        TypeId = 1,
        TextureId = 1
    };
    var serializeString = JsonSerializer.Serialize(element);
    var data = new StringContent(serializeString, Encoding.UTF8, mediaType: "application/json");
    var response = await _client.PostAsync("/api/Element", data);
    var contentString = await response.Content.ReadAsStringAsync();
    var elementRs = JsonSerializer.Deserialize<Element>(contentString);
    Assert.Equal(expected: HttpStatusCode.Created, actual: response.StatusCode);
    Assert.NotNull(elementRs);
    Assert.IsType<Element>(elementRs);
}
```

FIGURE 4.10 – Code backend du test unitaire du cas d'utilisation de la création d'élément chimique

La figure ci-dessous présente le résultat du test unitaire du cas d'utilisation de la création d'élément chimique, qui s'avère concluent.

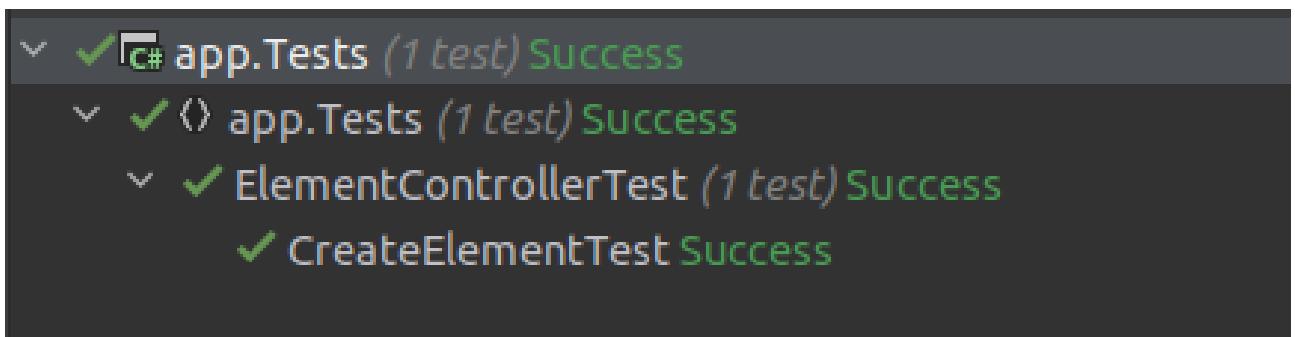


FIGURE 4.11 – Résultat du test unitaire du cas d'utilisation de la création d'élément chimique

4.2.2.3 Code source frontend

La figure ci-dessous présente le code qui a permis la création de l'interface de création d'élément chimique.

```
export default function ElementForm() {
  const [value, setValue] = useState(initialState: 1);
  const [element, setElement] = useState(initialState: null);
  const [loading, setLoading] = useState(initialState: true);

  const isAtom = useCallback(callback: () => { ... }, deps: [element]);

  const tabHeaders = useMemo(factory: () => { ... }, deps: [isAtom])
  const [searchParams] = useSearchParams();
  const getMethod = useGet()

  const getCurrentElement = useCallback(callback: async (id) => { ... }, deps: [getMethod]);
  useEffect(effect: () => { ... }, deps: [getCurrentElement, searchParams]);

  const handleChange = (event, newValue) => { ... };

  if (loading) return <Loader text="Chargement de l'élément..." />

  return <Page title="Création d'éléments chimiques">
    <Container>
      <Stack>
        <Typography variant="h4" sx={{}}>
          Création d'une molécule ou d'un atome
        </Typography>
        <Typography sx={{color: 'text.secondary', mb: 5}}>
          Saisissez les informations sur l'équipement
          à créer ci-dessous !
        </Typography>
      </Stack>
      <Stack direction="row" flexWrap="wrap-reverse" alignItems="center" sx={{mb: 5}}>
        <Box sx={{width: "100%}}>
          <Box>
            <Tabs value={value} onChange={handleChange} ... >
          </Box>
          <Card sx={{p: 3}}>
            <TabPanel value={value} index={0}>
              <AtomForm atom={isAtom() === true ? element : null}/>
            </TabPanel>
            <TabPanel value={value} index={1}>
              <MoleculeForm molecule={isAtom() === false ? element : null}/>
            </TabPanel>
          </Card>
        </Box>
      </Stack>
    </Container>
  </Page>
}
```

FIGURE 4.12 – Code frontend du cas d'utilisation de la création d'élément chimique

4.2.3 Listing des éléments chimiques

4.2.3.1 Interface de listing des éléments chimiques

La figure ci-dessous présente l'interface qui permet le listing des éléments chimiques. La première section présente la liste des atomes utilisables sur la plateforme et dans la deuxième section plus basse la liste des molécules créés par les utilisateurs de la plateforme.

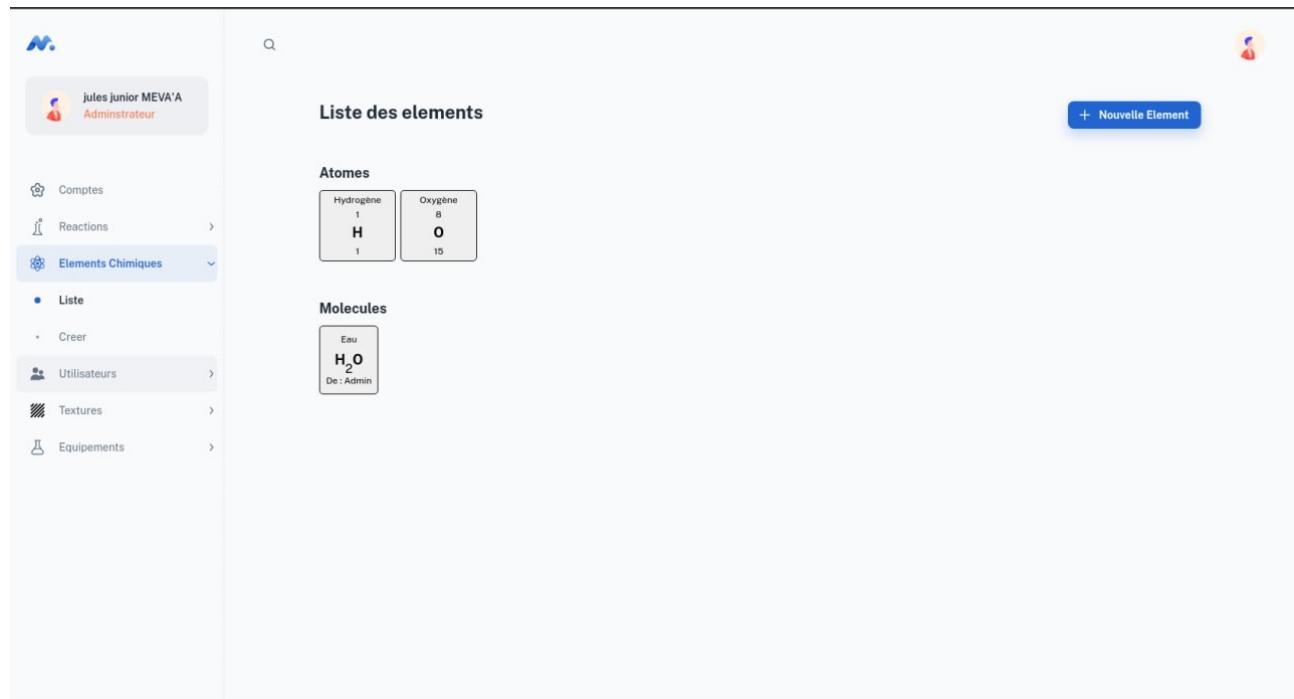


FIGURE 4.13 – IHM du listing des éléments chimiques

4.2.3.2 Code source backend

Le code ci-dessous présente le backend du contrôleur d'éléments chimiques, plus précisément la fonctionnalité de listing des éléments chimiques.

Cette fonctionnalité permet la récupération des éléments chimiques en base de données afin de les communiquer à l'interface pour un rendu à l'utilisateur (enseignants ou administrateurs).

```
[HttpGet]
[User(jules2m1998)]
public async Task<ActionResult<List<Element>>> GetAll()
{
    var all:List<Element> = await _elements.AsNoTracking().ToListAsync();
    return Ok(all);
}
```

FIGURE 4.14 – Code backend du contrôleur de listing des éléments chimiques

Le code ci-dessous présente le test unitaire de la fonctionnalité de listing des éléments chimiques.

Ce test unitaire permet de s'assurer que toutes les réponses aux requêtes de listing des éléments chimiques sur le serveur soient des réponses de type succès et que le contenu de la réponse soit une liste d'éléments chimique.

```
[Fact]
public async Task GetAllElementTest()
{
    _client.DefaultRequestHeaders.Authorization = new AuthenticationHeaderValue(
        scheme: "Bearer",
        parameter: "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiIxMjM0NTY3ODkwIi
    );
    var response = await _client.GetAsync(requestUri: "/api/Element");
    var content: string = await response.Content.ReadAsStringAsync();
    var elementRs = JsonSerializer.Deserialize<List<Element>>(content);
    Assert.Equal(expected: HttpStatusCode.OK, actual: response.StatusCode);
    Assert.NotNull(elementRs);
    Assert.IsType<List<Element>>(elementRs);
}
```

FIGURE 4.15 – Code backend du test unitaire du cas d'utilisation de la listing des éléments chimiques

La figure ci-dessous présente le résultat du test unitaire du cas d'utilisation de listing des éléments chimiques et révèle que le test est concluent.

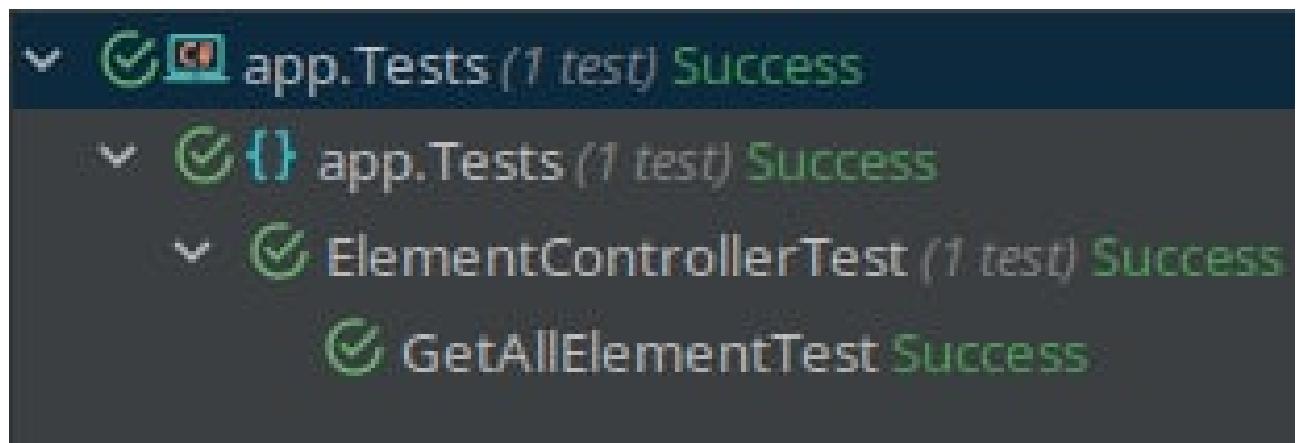


FIGURE 4.16 – Résultat du test unitaire de la fonctionnalité de listing des éléments chimiques

4.2.3.3 Code source frontend

La figure ci-dessous présente le code qui a permis la création de l'interface de listing des éléments chimiques.

```
export default function Elements() {
  const [loading, setLoading] = useState( initialState: true );
  const [element, setElement] = useState( initialState: [] );
  const getMethod = useGet();

  const fetchAll = useCallback( callback: async () => {
    const data = await getMethod("Element")
    setElement(data)
    setLoading( value: false )
  }, [deps: [getMethod]]);

  const separateElt = useMemo( factory: () => element.reduce((acc, current) => {
    if (current.children.length) return { ...acc, molecule: [...acc.molecule, current]}
    return { ...acc, atom: [...acc.atom, current]}
  }, {molecule: [], atom: []}), [deps: [element]]);

  useEffect( effect: () => {
    fetchAll();
  }, [deps: [fetchAll]]);

  if (loading) return <Loader text="Chargement des éléments ..." />

  return <Page title="Elements chimiques">
    <Container>
      <Stack direction="row" alignItems="center" justifyContent="space-between" mb={5} ... >
        <Stack spacing={6}>
          <Stack>
            <Typography variant="h6" gutterBottom>
              Atomes
            </Typography>
            <Atom>
              {
                separateElt.atom.sort( compareFn: (a:T, b:T) => a.atomicNumber - b.atomicNumber).map((i:T, k:number) => <AtomItem key={k} refresh={fetchAll} item={i}/>)
              }
            </Atom>
          </Stack>
          <Stack>
            <Typography variant="h6" gutterBottom>
              Molécules
            </Typography>
          </Stack>
        </Stack>
      </Stack>
    </Container>
  </Page>
}
```

FIGURE 4.17 – Code frontend du cas d'utilisation du listing des éléments chimiques

4.2.4 Crédation des réactions chimiques

4.2.4.1 Interface du formulaire de création des réactions chimiques

La figure ci-dessous présente l'interface qui permet la création des réactions chimiques.

Il est question ici d'un formulaire dont les champs permettent la récupération des informations pour la réaction et son enregistrement en base de données.

The screenshot shows a user interface for creating a chemical reaction. On the left, there's a sidebar with a user profile for 'admin ADMIN Adminstrateur'. The 'Reactions' option is selected in the sidebar. The main content area is titled 'Création d'une réaction chimique' and contains fields for 'Nom de la réaction' and 'Description de la réaction'. Below these, there are two sections: 'Réactifs' and 'Produits', each with a dropdown menu set to 'Atomes'. A red button labeled 'Activer' is located between these sections. At the bottom right is a blue button labeled 'Enregistrer'.

FIGURE 4.18 – IHM formulaire de création des réactions chimiques

4.2.4.2 Code source backend

La figure ci-dessous présente la définition du modèle d'une réaction chimique qui sera stocké en base de données.

```
11 usages  jules2m1998  4 exposing APIs
public class Reaction : BaseModel
{
    1 usage
    public string Name { get; set; } = string.Empty;
    1 usage
    public string? Description { get; set; } = string.Empty;
    5 usages
    public User User { get; set; }

    4 usages
    public virtual List<Product> Products { get; set; }
    4 usages
    public virtual List<Reactant> Reactants { get; set; }
}
```

FIGURE 4.19 – Code backend du modèle de réaction

La figure ci-dessous présente le code backend du contrôleur des réactions chimiques, plus précisément la fonctionnalité de création des réactions chimiques.

Cette fonctionnalité permet la création des réactions chimiques en base de données pour une pratique par les apprenants.

```
[HttpPost]
[jules2m1998]
public async Task<ActionResult<Reaction>> Create([FromBody] RegisterReactionDto dto)
{
    var userId:string? = User.FindFirstValue(ClaimTypes.NameIdentifier);
    var user = await _context.Users.FindAsync(params keyValues: int.Parse(userId));
    if (user is null) return Unauthorized();
    if (dto.ProductIds.Count == 0)
        return BadRequest(error: new Response
    {
        Errors = new Dictionary<string, string>
        {
            { "product", "Nombre de produit insuffisant pour une reaction chimique !" }
        }
    });
    if (dto.ReactantIds.Count == 0)
        return BadRequest(error: new Response
    {
        Errors = new Dictionary<string, string>
        {
            { "reactants", "Nombre de reactif insuffisant pour une reaction chimique !" }
        }
    });

    try
    {
        var reactants:IList<Reactant> = dto.ReactantIds.Select(compose =>
        {
            var elt:Element? = _context.Elements.Find(compose.Id);
            if (elt is null)
                throw new ExceptionResponse
            {
                Errors = new Dictionary<string, string>
                {
                    { "reactant", "Un ou plusieurs reactifs inexistant" }
                }
            };
            return new Reactant
            {
                Element = elt,
                Quantity = compose.Quantity
            };
        });
    }
}
```

FIGURE 4.20 – Code backend du contrôleur de création des réactions chimiques

Le code ci-dessous présente le test unitaire de la fonctionnalité de la création d'une réaction chimique.

Ce test unitaire permet de s'assurer que toutes les requêtes de création de réactions chimiques sur le serveur soient des réponses de type succès et que la réaction a bien été stockée en base de données.

```
[Fact]
public async Task CreateReactionTest()
{
    _client.DefaultRequestHeaders.Authorization = new AuthenticationHeaderValue(scheme: "Bearer", parameter: "eyJhbGciOiJIUzI1NiJ9.eyJzdWIiOiJ1c2VycmluZSIsInN1YiI6ImFkbWluIiwiaWF0IjoxNTE2MjM0NTY3fQ.vDgXyfCqLmPjwvOOGHkxRzXKuXo");
    var reaction = new RegisterReactionDto
    {
        Name = "Test",
        Description = "Test",
        ProductIds = new List<ComposeReaction>(collection: new []{new ComposeReaction{Id = 1, Quantity = 2}, new ComposeReaction{Id = 2, Quantity = 1}});
    };
    var serializeString = JsonSerializer.Serialize(reaction);
    var data = new StringContent(serializeString, Encoding.UTF8, mediaType: "application/json");
    var response = await _client.PostAsync(requestUri: "/api/Reaction", data);
    var contentString = await response.Content.ReadAsStringAsync();
    var reactionRs = JsonSerializer.Deserialize<Reaction>(contentString);
    Assert.Equal(expected: HttpStatusCode.Created, actual: response.StatusCode);
    Assert.NotNull(reactionRs);
    Assert.IsType<Reaction>(reactionRs);
    Assert.Equal(expected: reactionRs?.Name, actual: reaction.Name);
    Assert.Equal(expected: reactionRs?.Products[0].Id, actual: reaction.ProductIds[0].Id);
    Assert.Equal(expected: reactionRs?.Products[1].Id, actual: reaction.ProductIds[1].Id);
    Assert.Equal(expected: reactionRs?.Description, actual: reaction.Description);
}
```

FIGURE 4.21 – Code backend du test unitaire du cas d'utilisation de la création d'une réaction chimique

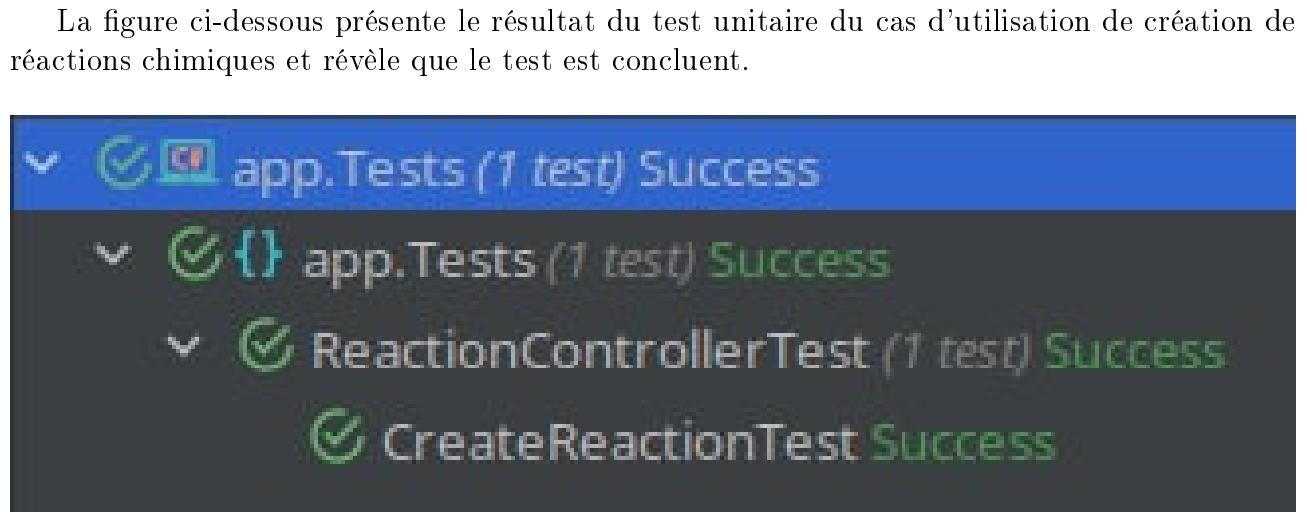


FIGURE 4.22 – Résultat du test unitaire de la fonctionnalité de création d'une réaction chimique

4.2.4.3 Code source frontend

La figure ci-dessous présente le code qui a permis la création de l'interface de création d'une réaction chimique.

```
const Formulaire = ({current}) => {
  const [elements, setElements] = useState( initialState: [] );
  const [reactants, setReactants] = useState( initialState: current?.reactants?.length > 0 ?[ ... current?.reactants?.map((v, k) => ({
    ...v, ...v.element,
    position: k + 1
}))]: [] );
  const [products, setProducts] = useState( initialState: current?.products?.length > 0 ?[ ... current?.products?.map((v, k) => ({
    ...v, ...v.element,
    position: k + 1
}))]: [] );
  const [active, setActive] = useState( initialState: current?.isActivated || false);

  const getMethod = useGet()
  const snack = useSnack()
  const postMethod = usePost()
  const putMethod = usePut()

  const reactionSchema = Yup.object().shape({
    name: Yup.string().required(REQUIRED),
    description: Yup.string().required(REQUIRED),
  })

  const defaultValues = useMemo( factory: () => ({
    name: current?.name || "",
    description: current?.description || "",
  }), [deps: [current]]);

  const methods = useForm( props: {
    resolver: yupResolver(reactionSchema),
    defaultValues,
  });

  const {
    handleSubmit,
    formState: {isSubmitting},
  } = useMemo( factory: () => methods, [deps: [methods]]);

  const onSubmit = useCallback( callback: async (e) => {
    if (!reactants.length || !products.length) return
    const data = {
      reactants: reactants.map((v) => v.element),
      products: products.map((v) => v.element),
      name: name.value,
      description: description.value
    }
    await postMethod(data)
    snack("Réaction créée")
    handleSubmit()
  }, [handleSubmit, formState, reactants, products, name, description, postMethod]);
}
```

FIGURE 4.23 – Code frontend du cas d'utilisation de la création d'une réaction chimique

4.2.5 Listing des réactions chimiques

4.2.5.1 Interface de listing des réactions chimiques

La figure ci-dessous présente l'interface qui permet le listing des réactions chimiques.

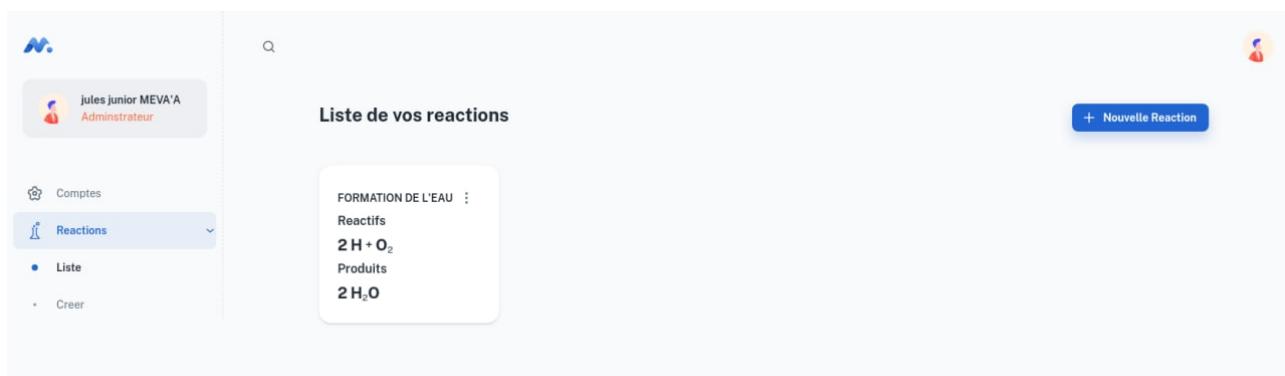


FIGURE 4.24 – IHM du listing des réactions chimiques

4.2.5.2 Code source backend

Le code ci-dessous présente le backend du contrôleur d'éléments chimiques, plus précisément la fonctionnalité de listing des réactions chimiques.

Cette fonctionnalité permet la récupération des réactions chimiques en base de données afin de les communiquer à l'interface pour un rendu à l'utilisateur (enseignants ou administrateurs).

```
[HttpGet]
[Authorize]
[Route("template: "current")]
[jules2m1998]
public ActionResult<List<Reaction>> GetByUser()
{
    var userId:string? = User.FindFirstValue(claimType:ClaimTypes.NameIdentifier);
    if (!int.TryParse(userId, out var id:int)) return Unauthorized();

    var reactions:List<Reaction> = _queryable.Where(reaction => reaction.User.Id = id).ToList();
    return Ok(reactions);
}
```

FIGURE 4.25 – Code backend du contrôleur de listing des réactions chimiques

Le code ci-dessous présente le test unitaire de la fonctionnalité de listing des réactions chimiques.

Ce test unitaire permet de s'assurer que toutes les réponses aux requêtes de listing des réactions chimiques sur le serveur soient des réponses de type succès et que le contenu de la réponse soit une liste de réaction chimique.

```
[Fact]
public async Task GetAllReactionsTest()
{
    _client.DefaultRequestHeaders.Authorization = new AuthenticationHeaderValue(scheme:"Bearer", parameter: "eyJhbGciOi
    var response = await _client.GetAsync(requestUri: "/api/Reaction/current");
    var content:string = await response.Content.ReadAsStringAsync();
    var reactions = JsonSerializer.Deserialize<List<Reaction>>(content);
    Assert.Equal(expected: HttpStatusCode.OK, actual: response.StatusCode);
    Assert.NotNull(reactions);
    Assert.IsType<List<Reaction>>(reactions);
    Assert.True(reactions?.All(reaction => reaction.User.Id == 1));
}
```

FIGURE 4.26 – Code backend du test unitaire du cas d'utilisation du listing des réactions chimiques

La figure ci-dessous présente le résultat du test unitaire du cas d'utilisation de listing des réactions chimiques et révèle que le test est concluent.

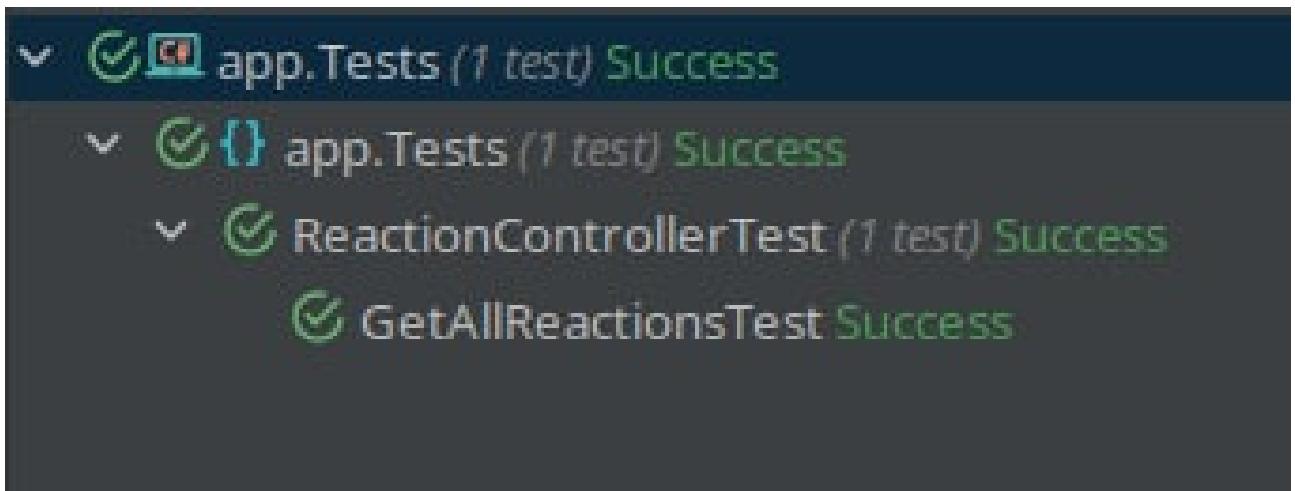


FIGURE 4.27 – Résultat du test unitaire de la fonctionnalité de listing des réactions chimiques

4.2.5.3 Code source frontend

La figure ci-dessous présente le code qui a permis la création de l'interface de listing des réactions chimiques.

```
export default function Elements() {
  const [loading, setLoading] = useState({ initialState: true });
  const [element, setElement] = useState({ initialState: [] });
  const getMethod = useGet()

  const fetchAll = useCallback(callback: async () => {
    const data = await getMethod("Element")
    setElement(data)
    setLoading({ value: false })
  }, [deps: [getMethod]]);

  const separateElts = useMemo(factory: () => element.reduce((acc, current) => {
    if (current.children.length) return { ...acc, molecule: [...acc.molecule, current] }
    return { ...acc, atom: [...acc.atom, current] }
  }, {molecule: [], atom: []}), [deps: [element]]);

  useEffect(effect: () => {
    fetchAll()
  }, [deps: [fetchAll]]);

  if (loading) return <Loader text="Chargement des éléments ..." />

  return <Page title="Elements chimiques">
    <Container>
      <Stack direction="row" alignItems="center" justifyContent="space-between" mb={5} ... >
        <Stack spacing={6}>
          <Stack>
            <Typography variant="h6" gutterBottom>
              Atomes
            </Typography>
            <Atom>
              {
                separateElts.atom.sort( compareFn: (a: T, b: T) => a.atomicNumber - b.atomicNumber).map((i: T, k: number) => <AtomItem key={k} refresh={fetchAll} item={i}/>)
              }
            </Atom>
          </Stack>
          <Stack>
            <Typography variant="h6" gutterBottom>
              Molécules
            </Typography>
          </Stack>
        </Stack>
      <Stack>
        <Image alt="Logo of the University of Paris" />
      </Stack>
    </Container>
  </Page>
}
```

FIGURE 4.28 – Code frontend du cas d'utilisation de listing des réactions chimiques

Conclusion

Il était question pour nous de concevoir un environnement virtuel permettant la simulation d'un laboratoire de chimie, afin de permettre aux apprenants d'évoluer dans un environnement sécurisé et moins onéreux qu'un laboratoire conventionnel. Et pour ce faire, nous avons d'une part fait un état de l'art afin de nous intéresser aux travaux réalisés sur le sujet, d'autre part, grâce à cet existant, nous avons effectué une étude détaillée dans laquelle nous avons analysé la solution grâce à un cadrage de pragmatique et synthétique du projet, une identification du besoin et la conception du système grâce à des diagrammes UML. Enfin, nous avons fait l'implémentation de la solution qui est passée par le choix des technologies utilisées à savoir Unreal Engine 5 pour la 3D, React & React Dom pour le web, ASP.NET core web api pour le backend, PostgreSQL comme système de gestion des bases de données.

Le résultat obtenu est satisfaisant et nous estimons le taux de réalisation à environ 75 %. Notre solution n'étant pas une fin en soi, il serait judicieux de mettre sur pied la création de réactions types suivant le programme de chaque classe afin que les apprenants effectuent des réactions en fonction de leur programme sans que l'enseignant n'ait à intervenir.

Bibliographie

- [1] S. Ken and J. Sutherland, “The scrum guide,” *Scrum Alliance*, vol. 21, no. 19, p. 1, 2011.
- [2] J. de Houwer, D. Barnes-Holmes, and A. Moors, “What is learning ? on the nature and merits of a functional definition of learning,” *Psychonomic Bulletin & Review*, vol. 20, pp. 631–642, 2013.
- [3] M. Radović-Marković *et al.*, “Advantages and disadvantages of e-learning in comparison to traditional forms of learning,” *Annals of the University of Petroşani, Economics*, vol. 10, no. 2, pp. 289–298, 2010.
- [4] W. DS and Z. JM., “How guided play promotes early childhood learning,” *Encyclopedia on Early Childhood Development*, 2018.
- [5] P. Angela Pyle and C. OISE, University of Toronto, “Apprentissage par le jeu,” *Encyclopedia on Early Childhood Development [online]*, 2018.
- [6] D. Whitebread, P. Coltman, H. Jameson, and R. Lander, “Play, cognition and self-regulation : What exactly are children learning when they learn through play ?,” *Educational and Child Psychology*, vol. 26, no. 2, p. 40, 2009.
- [7] D. EN, R. GB, and R. KH, “Play-based learning and social development,” *Encyclopedia on Early Childhood Development [online]*, 2018.
- [8] A. Pyle and E. Danniels, “A continuum of play-based learning : The role of the teacher in play-based pedagogy and the fear of hijacking play,” *Early Education and Development*, vol. 28, no. 3, pp. 274–289, 2017.
- [9] H.-D. B, Z. JM, H.-P. K, and G. RM, *Playing to Learn Mathematics*. 2018.
- [10] B. LE, *The Role of Make-Believe Play in Development of Self-Regulation*. 2018.
- [11] B. D, *Cognitive Development in Play-Based Learning*. 2018.
- [12] T. de Souza Silva, E. C. R. Marinho, G. R. E. Cabral, and K. S. da Gama, “Motivational impact of virtual reality on game-based learning : Comparative study of immersive and non-immersive approaches,” in *2017 19th Symposium on Virtual and Augmented Reality (SVR)*, pp. 155–158, IEEE, 2017.
- [13] L. Shackelford and W. D. Huang, “Relationships between cognitive loads and motivational support in a virtual reality game-based learning system for teaching introductory archaeology,” 2019.
- [14] M. A. Abdelaziz, H. M. El-Bakry, A. M. Riad, and M. B. Senousy, “The impact of using virtual reality on student’s motivation for operating systems course learning,” 2020.

- [15] J. Jerald, "What is virtual reality," 2015.
- [16] H. Rheingold, "Virtual reality," 1991.
- [17] I. Sutherland, "The ultimate display," 1965.
- [18] C. Anthes, R. J. García-Hernández, M. Wiedemann, and D. Kranzlmüller, "State of the art of virtual reality technology," in *2016 IEEE aerospace conference*, pp. 1–19, IEEE, 2016.
- [19] X. Duan, S.-J. Kang, J. I. Choi, and S. K. Kim, "Mixed reality system for virtual chemistry lab," *KSII Transactions on Internet and Information Systems (TIIS)*, vol. 14, no. 4, pp. 1673–1688, 2020.
- [20] A. Li, Z. Montaño, V. J. Chen, and J. I. Gold, "Virtual reality and pain management : current trends and future directions," *Pain management*, vol. 1, no. 2, pp. 147–157, 2011.
- [21] A. Baldominos, Y. Saez, and C. G. Del Pozo, "An approach to physical rehabilitation using state-of-the-art virtual reality and motion tracking technologies," *Procedia Computer Science*, vol. 64, pp. 10–16, 2015.
- [22] K. M. Malloy and L. S. Milling, "The effectiveness of virtual reality distraction for pain reduction : a systematic review," *Clinical psychology review*, vol. 30, no. 8, pp. 1011–1018, 2010.
- [23] M. Alaker, G. R. Wynn, and T. Arulampalam, "Virtual reality training in laparoscopic surgery : a systematic review & meta-analysis," *International Journal of Surgery*, vol. 29, pp. 85–94, 2016.
- [24] E. Yiannakopoulou, N. Nikiteas, D. Perrea, and C. Tsigris, "Virtual reality simulators and training in laparoscopic surgery," *International Journal of Surgery*, vol. 13, pp. 60–64, 2015.
- [25] M. Hackett and M. Proctor, "Three-dimensional display technologies for anatomical education : a literature review," *Journal of Science Education and Technology*, vol. 25, no. 4, pp. 641–654, 2016.

Table des matières

Remerciements	ii
Sommaire	iii
Table des figures	iv
Liste des tableaux	vi
Introduction	1
I ÉTAT DE L'ART	2
1 PRÉSENTATION DU PROJET	3
1.1 Compréhension du sujet	4
1.1.1 Contexte	4
1.1.2 Délimitation du sujet et hypothèse du travail	4
1.2 Étude de l'existant	4
1.2.1 Description de l'existant	5
1.2.2 Critique de l'existant et Problématique	5
1.2.3 Quelques solutions existantes	5
1.2.4 Questions de recherche	9
1.3 Choix et intérêt du sujet	9
1.4 Objectif du travail	9
1.5 Méthodologie	10
1.5.1 Gestion de projet	10
1.5.1.1 Qu'est-ce que la méthode SCRUM	12
1.5.1.2 Pourquoi la méthode SCRUM	12
1.5.1.3 Avantages de la méthode SCRUM	13
1.5.1.4 Dans quels cas utiliser la méthode SCRUM	13
1.5.1.5 Principale contrainte de la méthode SCRUM	13
1.5.2 Analyse et Modélisation	13
1.5.2.1 L'approche orientée objet	13
1.5.2.2 UML et MERISE	14
2 GÉNÉRALITÉS SUR LES OUTILS D'APPRENTISSAGE IMMERSIF BASÉ SUR LA RÉALITÉ VIRTUELLE	16

2.1	Apprentissage	17
2.1.1	L'apprentissage traditionnel	17
2.1.2	L'apprentissage par le jeu	17
2.2	La réalité virtuelle et l'éducation	18
2.2.1	La réalité virtuelle	18
2.2.1.1	Principe de fonctionnement	18
2.2.2	La réalité virtuelle et l'éducation	20

II ÉTUDE DÉTAILLÉE ET RÉALISATION 22

3	Cadrage, analyse et conception	23
3.1	Cadrage du projet	24
3.1.1	Le projet	24
3.1.1.1	Le nom	24
3.1.1.2	Définition succincte	24
3.1.1.3	Caractéristiques essentielles	24
3.1.1.4	Motifs qui sous-tendent ce projet	24
3.1.2	Les objectifs	24
3.1.2.1	Objectifs techniques	24
3.1.2.2	Objectifs de délai	25
3.1.2.3	Objectifs de coûts	25
3.1.3	La technique	28
3.1.3.1	La base sur laquelle le projet s'appuie	28
3.1.3.2	Les difficultés principales de ce projet	28
3.1.3.3	Les solutions de repli en cas de problème	28
3.1.4	Le planning	28
3.1.4.1	Dates clés	28
3.1.4.2	Grandes phases du planning	28
3.1.4.3	Diagramme Gantt	29
3.1.5	Les moyens	31
3.1.5.1	Moyens humains	31
3.1.5.2	Moyens matériels	31
3.1.6	Le management du projet	31
3.1.7	La communication	31
3.1.7.1	Communication interne du projet	31
3.1.7.2	Communication externe	32
3.2	Analyse du système	32
3.2.1	Analyse fonctionnelle	32
3.2.1.1	Identification des acteurs	32
3.2.1.2	Identification des cas d'utilisation	32
3.2.1.3	Diagramme des cas d'utilisation	33
3.2.1.4	Description textuelle de quelques cas d'utilisation	37
3.2.1.5	Diagrammes de séquences	40
3.2.1.6	Diagrammes d'activités	43
3.2.1.7	Diagramme d'état transition	45

3.2.2	Analyse non fonctionnelle	46
3.2.3	Technique et Méthodes	46
3.2.3.1	Choix des techniques de développement	46
3.2.3.2	Choix du Framework	47
3.3	Conception du système	47
3.3.1	Diagramme de classe	47
3.3.2	Diagramme de déploiement	48
3.3.3	Quelques maquettes d'IHM (Interface Homme Machine)	49
3.3.3.1	Page d'accueil	49
3.3.3.2	Page d'inscription des enseignants	51
3.3.3.3	Page de connexion des enseignants et administrateurs	51
3.3.3.4	Page espace personnel des enseignants et des administrateurs	52
3.3.3.5	Interface du laboratoire en vue de dessus	54
3.3.3.6	Interface du laboratoire en vue de face	54
3.3.3.7	Interface du laboratoire intérieur vue du fond	55
3.3.3.8	Interface du laboratoire intérieur vue de l'apprenant	55
3.3.3.9	Interface du laboratoire intérieur vue de l'enseignant	56
4	Implémentation de la solution et résultats	57
4.1	Outils et technologies	58
4.2	Résultats	59
4.2.1	Édition du profil utilisateur	59
4.2.1.1	Interface d'édition du profil utilisateur	59
4.2.1.2	Code source backend	59
4.2.1.3	Code source frontend	62
4.2.2	Création des éléments chimiques	63
4.2.2.1	Interface de création des éléments chimiques	63
4.2.2.2	Code source backend	63
4.2.2.3	Code source frontend	66
4.2.3	Listing des éléments chimiques	68
4.2.3.1	Interface de listing des éléments chimiques	68
4.2.3.2	Code source backend	68
4.2.3.3	Code source frontend	69
4.2.4	Création des réactions chimiques	70
4.2.4.1	Interface du formulaire de création des réactions chimiques	70
4.2.4.2	Code source backend	71
4.2.4.3	Code source frontend	74
4.2.5	Listing des réactions chimiques	75
4.2.5.1	Interface de listing des réactions chimiques	75
4.2.5.2	Code source backend	76
4.2.5.3	Code source frontend	77
Bibliographie		80
Table des matières		82