

Rapport de stage Assistant ingénieur

Semestre 8

Mise en place d'un environnement de
réalité virtuelle et/ou de réalité
augmentée permettant la mise en
œuvre des protocoles de traçabilité et
d'hygiène.

Le Gall Marine

28/05/23

contact@emansk.fr
SIRET 820 843 001 00026 APE 7490B
29480 LE RELECO KERHON
ZI DE KERSCAO
e-ManRisk
Taigry DANIEL.

Remerciements

Je tiens à remercier toutes les personnes qui ont contribué au succès de mon stage et qui m'ont aidé lors de la rédaction de ce rapport.

Tout d'abord je tiens à remercier Tanguy Daniel, mon maitre de stage, pour m'avoir permis d'effectuer ce stage dans l'entreprise d'e-ManRisk ainsi que pour son accueil, le temps passé ensemble et le partage de son expertise.

Je tiens à remercier Ronan Querrec, Kevin Le Bars, Nicolas Simon et les étudiants de 5^{ème} année qui ont travaillé avec moi pour leur aide au quotidien dans leur domaine de compétence. Grace à vous j'ai pu résoudre les problèmes techniques rencontrés.

Je remercie également Mr Le Magueresse pour avoir été mon tuteur de stage.

Enfin je remercie Patrick, Romain et Timothée pour la relecture et l'aide apportée à ce rapport.

Résumé

Dans ce rapport de stage je vais vous décrire ce que j'ai effectué pendant les trois premiers mois de mon stage assistant ingénieur dans l'entreprise e-ManRisk. Voici un petit condensé de ce que je vais décrire par la suite.

J'ai commencé mon stage par faire la visite d'un E.Leclerc afin de pouvoir observer dans quel environnement le personnel travaille. Cela était nécessaire afin de pouvoir ensuite modéliser au mieux l'environnement virtuel. J'ai ensuite fait une phase d'autoformation et d'analyse des outils à travers quelques tutoriels Unity pour prendre en main le logiciel et réviser mes bases en C#. Puis je suis passé à la phase de conception en réalisant les diagrammes UML pour mettre en forme le modèle de l'API de l'entreprise et le modèle que j'allais manipuler par la suite. Enfin je suis passé à la phase de réalisation en commençant par faire des requêtes HTTP sur l'API du réseau Bibus pour m'entraîner puis sur l'API de l'entreprise. Après avoir réussi à faire fonctionner toutes les requêtes dans l'environnement de Unity j'ai réalisé avec les étudiants de cinquième année à l'ENIB présents sur ce projet, les objets 3D nécessaire pour créer l'environnement d'un E.Leclerc comme par les pièces et les objets utiles comme le distributeur de savon, le lavabo, le four, etc. Enfin j'ai réalisé le scénario décrit par l'entreprise qui consiste à se laver les mains soigneusement, puis à prendre une sonde de thermomètre et de contrôler la température d'un produit pendant sa phase de cuisson puis de refroidissement et enfin de le stocker en chambre froide.

Introduction

Voici le rapport de stage effectué dans le cadre du stage S8 assistant ingénieur à l'ENIB du 24 février au 9 juillet 2021. L'objectif de ce stage est la mise en place d'un environnement de réalité virtuelle et/ou de réalité augmentée permettant la mise en œuvre des protocoles de traçabilité et d'hygiène pour se faire j'ai mis en place toute une procédure que je vais vous détailler point par point, toutefois n'ayant pas encore terminé mon stage au moment de l'écriture de ce rapport, il se peut que des choses soient amenées à évoluer dans le futur et que certaines informations soient obsolètes. L'utilisateur final sera un employé d'un magasin E.Leclerc se formant sur les protocoles de sécurité alimentaires via une application de réalité virtuelle et/ou augmentée. Le stage s'est déroulé dans l'entreprise e-ManRisk. Dans ce rapport je vais commencer par un glossaire pour définir les termes importants, puis je ferai une présentation de l'entreprise dans laquelle j'ai effectué mon stage. Je développerai ensuite sur le déroulé du stage en parlant de la visite des locaux, de la découverte de Unity, des diagrammes UML réalisés, des requêtes HTTP envoyées, des objets 3D créés et de la réalisation du scénario. Je terminerai ensuite sur un bilan des compétences acquises et sur les perspectives pour la fin du stage et pour la suite de mes études.

Sommaire

Remerciements	2
Résumé	3
Introduction	4
Sommaire	5
I. Glossaire	6
II. Liste des figures	7
III. Présentation de l'entreprise	8
IV. Développement du sujet de stage	11
1. Visite de l'entreprise	12
2. Découverte de Unity	15
3. Diagramme UML	18
4. Requêtes http	23
5. Eléments 3D	27
6. Réalisation du scénario	30
a. Le lavage des mains	30
b. La cuisson d'un produit	31
c. Le refroidissement d'un produit	32
d. Le stockage en chambre froide	32
V. Problèmes rencontrés	33
VI. Bilan et perspectives	34
VII. Conclusion	35
VIII. Bibliographie	36
IX. Annexes	37

I. Glossaire

API : (Application Programming Interface) Interface de programmation d'application

Unity : Unity est un moteur de jeu multiplateforme (smartphone, ordinateur, consoles de jeux vidéo et Web) développé par Unity Technologies. Il permet de créer des environnements virtuels et de les tester dans un casque très facilement.

Requête HTTP : L'Hypertext Transfer Protocol (littéralement « protocole de transfert hypertexte ») est un protocole de communication client-serveur développé pour le *World Wide Web*.

cURL : cURL est une interface en ligne de commande, destinée à récupérer le contenu d'une ressource accessible par un réseau informatique. Il permet de faire des requête http.

Postman : Postman est un logiciel qui va vous permettre d'appeler / tester une API.

C# : le C# est un langage de programmation orientée objet. C'est le langage utilisé sur Unity.

Laboratoires : Un laboratoire est l'endroit, dans un magasin, où les produits sont transformé et/ou fabriquer. Par exemple la boulangerie ou la boucherie dans un supermarché E.Leclerc sont des laboratoires.

Scène (Unity) : Les scènes sont l'endroit où sont placé tout les objets 3D sur Unity

UML : C'est un langage de modélisation graphique à base de pictogrammes.

II. Liste des figures

Figure 1 - logo de Sequoia.....	9
Figure 2 - Logo e-ManRisk.....	10
Figure 3 - exemple de laboratoire	13
Figure 4 - exemple de chambre froide.....	13
Figure 6 - Schéma réalisation du projet	14
Figure 5 - exemple de stockage.....	14
Figure 7 - jeu de plateforme	15
Figure 8 - import d'un modèle d'une maison	16
Figure 9 - création d'un terrain	17
Figure 10 - diagramme UML base de données.....	18
Figure 11 - exemple de réponse de l'API	19
Figure 12 - modèle UML métadonnées	20
Figure 13 - Diagramme UML modèle Métier	21
Figure 14 - exemple requête cURL.....	23
Figure 15 - interface Postman	24
Figure 16 - requête http en C#.....	24
Figure 17 - requête post authentification	25
Figure 18 - comparaison format JSON et C#	26
Figure 19 - objets 3D produits	27
Figure 20 - environnement global	28
Figure 21 - lavabo.....	28
Figure 22 - distributeur de savon et de serviette.....	28
Figure 23 - thermomètre et sonde	29
Figure 24 - four et frigidaire	29
Figure 25 - mouvements de mains à réaliser	30

III. Présentation de l'entreprise

Dans le cadre de ma scolarité à l'École Nationale d'Ingénieurs de Brest, j'ai effectué mon stage assistant ingénieur à e-ManRisk. C'est une filiale de E.Leclerc spécialisée dans le domaine de la gestion des risques et du développement logiciel. Cette entreprise propose un service complet de management des risques en entreprise. Ils agissent à tous les niveaux pour les sociétés du Mouvement E.Leclerc. Ils conçoivent et développent des solutions logicielles qui viennent parfaitement compléter le système de management des risques.

E-ManRisk appartient à l'enseigne de grande distribution E.Leclerc. E.Leclerc est un organisme qui ne se qualifie pas comme un groupe, du fait que chaque magasin est indépendant. C'est pour cela qu'on parle de Mouvement E.Leclerc, et c'est ce terme que j'utiliserai dans la suite de mon document. Le Mouvement E.Leclerc est un ensemble de 640 magasins tous dirigés par des adhérents. Ils sont tous indépendants, mais en adhérant au Mouvement, ils se soumettent aux règles dictées par l'Association des Centres Distributeurs E.Leclerc (ACDLec), qui est l'organe politique du Mouvement. C'est elle qui donne le droit aux adhérents d'utiliser les marques E.Leclerc, qui s'occupe de la communication, des finances, qui détermine les orientations stratégiques du Mouvement et s'assure que ses valeurs soient respectées par les adhérents et affiliés. Les magasins sont approvisionnés par des centrales d'achats sous statut coopératif. Par exemple, pour les départements du Finistère, des Côtes-d'Armor et du Morbihan c'est la Scarmor (ce qui signifie Société Coopérative d'Achats de l'Armor) qui approvisionne les magasins.

Le Mouvement E.Leclerc comprend aussi 15 entités spécialisées dont l'ACDLec dont j'ai parlé plus tôt. Le Groupement d'Achat E.Leclerc (Galec) est la coopérative d'achats nationale. Elle s'occupe du référencement des produits nationaux et de l'aspect marketing du Mouvement.

C'est cette entité qui finance le projet à ses débuts. En 2011, le déploiement de Séquoia est fait à tous les magasins de la Scarmor et en 2014, de nouvelles centrales commencent à s'intéresser au logiciel et souhaitent pouvoir l'utiliser. C'est ainsi qu'en 2016, e-ManRisk, une SAS (Société par Action Simplifiées à capital variable) dont David Constans et Tanguy Daniel sont les premiers membres, est créée. Cela va permettre de pouvoir proposer le logiciel à d'autres magasins externes à la Scarmor, à la condition qu'ils appartiennent au Mouvement E.Leclerc, puisque e-ManRisk est créée au sein du Mouvement et appartient aux centrales d'achats qui utilisent ses services. Aujourd'hui, plus de 60%, soit près de 380 magasins de E.Leclerc utilisent le logiciel, l'objectif premier étant de pouvoir le proposer à tous les magasins du Mouvement en France.



Figure 1 - logo de Séquoia

Séquoia est le logiciel à l'origine de la création d'e-ManRisk. C'est un logiciel dont l'interface client est gérée en mode SaaS, Software as a Service. C'est à dire qu'il permet à l'entreprise cliente de l'utiliser via Internet au lieu de devoir l'installer sur leur serveur. C'est un outil couvrant les besoins opérationnels de la gestion des risques. C'est-à-dire qu'il permet d'écrire et de centraliser les documents concernant les domaines de l'administration, ressources humaines, sécurité, environnement, maintenance et qualité. Il permet une entrée simple de toutes ces données et de les garder en mémoire. Il permet aussi d'envoyer des alertes automatiques si nécessaire et selon l'organisation de l'entreprise. Séquoia permet d'extraire des analyses graphiques des données entrées et de mesurer la performance d'une entreprise au cours d'une période donnée. Il existe une partie interne à l'entreprise, appelée le back-office. Elle est utilisée uniquement par le personnel d'e-ManRisk. En parallèle de Séquoia, e-ManRisk développe d'autres logiciels à la demande du Mouvement.

La société est encore une petite entreprise qui compte à ce jour neufs employés et deux alternants mais qui est en constante évolution.



Figure 2 - Logo e-ManRisk

IV. Développement du sujet de stage

Durant ce stage je dois mettre en place un environnement de réalité virtuelle et/ou de réalité augmentée permettant la mise en œuvre des protocoles de traçabilité et d'hygiène. L'utilisateur final sera un employé faisant sa formation sur ces protocoles. Il devra les appliquer suivant plusieurs scénarios. L'environnement virtuel sera représentatif d'un magasin Leclerc avec les différents lieux utiles (labos, rayon, chambre froide, etc...). L'objectif de ce stage est aussi de montrer l'intérêt d'avoir un environnement virtuel pour effectuer cette formation.

Pour ce stage je serai aidé de quatre étudiants ENIB de 5^{ème} année qui, pour leur projet informatique, se chargeront de mettre en place l'environnement virtuel. En effet il faut recréer le modèle d'un magasin E.Leclerc avec tous les éléments qu'un opérateur a besoin pour sa formation que ce soit au niveau des petits objets comme le distributeur de savon ou la sonde de température. Ils vont aussi modéliser toutes les pièces utiles comme les laboratoires où sont transformés les produits ou la chambre froide. Ils seront aussi chargés de s'occuper de l'intégration du logiciel sequoia et de la tablette dans l'environnement virtuel.

Pour cela j'ai commencé par faire des tutoriels de prise en main d'Unity et du C# puis j'ai codé les différentes requêtes http dans l'environnement d'Unity. Ensuite j'ai réalisé le diagramme UML correspondant au modèle métier pour instancier toutes les données présentes dans la base de données. Enfin j'ai travaillé sur le scénario définis à l'avance par e-ManRisk.

1. Visite de l'entreprise

Le premier jour de mon stage je me suis rendue dans les locaux de e-ManRisk avec les étudiants de l'ENIB de cinquième année pour définir les attentes du stage. Pour commencer nous avons pu voir le fonctionnement de leur application Sequoia. En effet dans les magasins E.Leclerc il faut réaliser des contrôles assez régulièrement sur les produits notamment au niveau de la température. Il faut contrôler régulièrement la température du produit en lui-même tout comme la température des chambres froide ou des bacs de refroidissement. Pour cela actuellement les employés ont des fiches papier à remplir régulièrement et ces dernières sont ensuite stockées et archivées. E-ManRisk a donc créé une application qui permet de faire ces contrôles de façon informatique avec une dimension de rappel pour ne pas oublier de faire certains contrôles. Cette application permet aussi d'apporter plus de précision et de solutions si la température relevée n'est pas conforme.

Nous avons aussi pu discuter plus en détails des modalités et du but du stage. En effet un scénario idéal a été défini par Nicolas Simon et nous avons pu avoir toutes les précisions nécessaires sur le fonctionnement d'un magasin E.Leclerc dans les coulisses. Le but du stage est aussi d'avoir un premier rendu sur ce qu'on peut réussir à faire en réalité virtuelle et/ou augmentée afin de pouvoir le développer plus tard si le test a été convaincant.

Enfin nous avons visité le magasin E.Leclerc du Relecq-Kerhuon afin de voir par nous même les différents lieux dans lesquelles travaillent les employés du E.Leclerc. Nous avons pu prendre quelques photos afin de s'en servir comme modèle pour créer l'environnement en 3D sur Unity.



Figure 3 - exemple de laboratoire

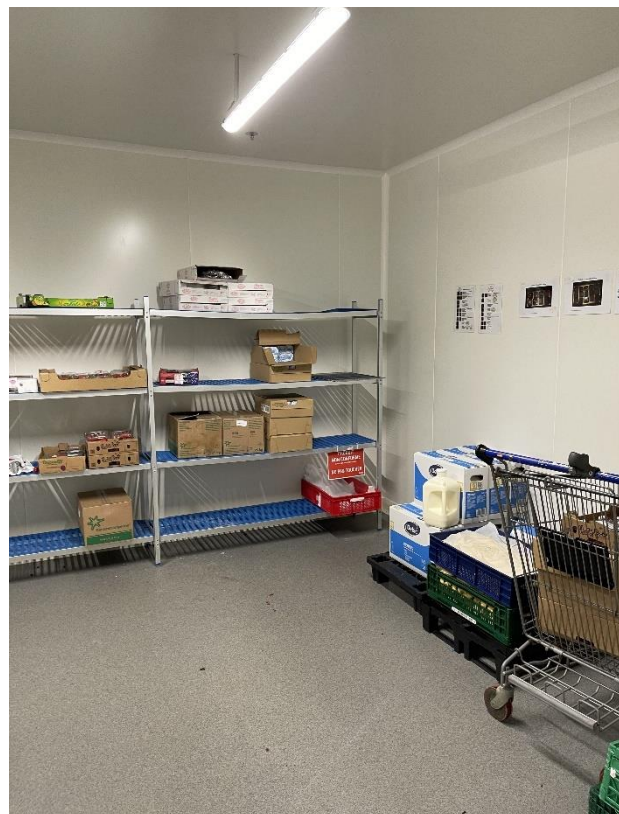


Figure 4 - exemple de chambre froide



Figure 5 - exemple de stockage

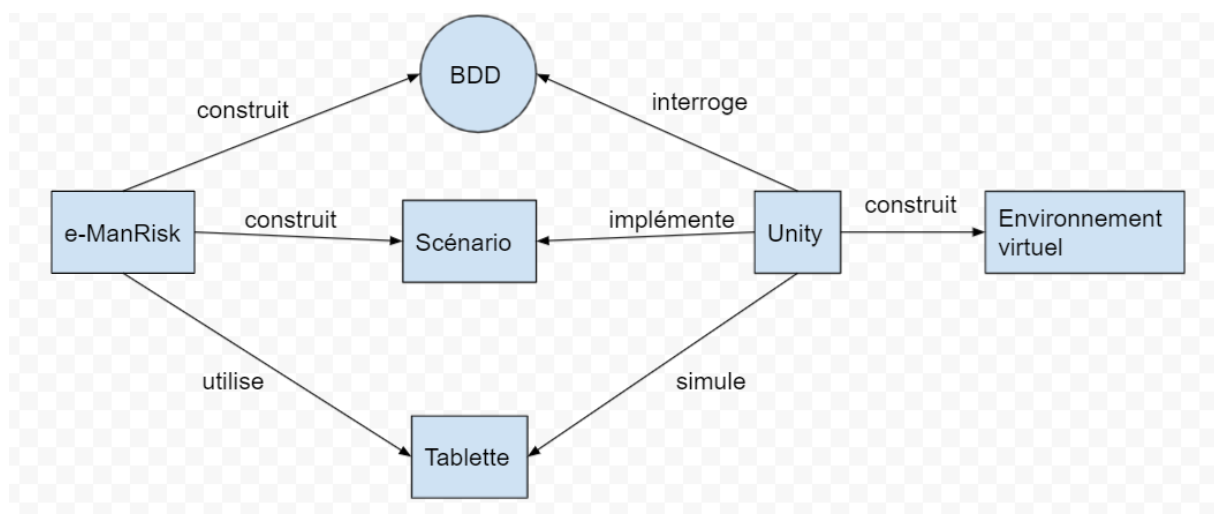


Figure 6 - Schéma réalisation du projet

Voici le schéma de ce qu'on souhaite mettre en place à travers le projet. Chaque bloc correspond à une brique qu'on doit analyser pour vérifier la faisabilité. Cela permet de se rendre compte des problèmes techniques et des outils disponibles pour réaliser ces briques.

2. Découverte de Unity

La première partie de mon stage a été une partie d'auto-formation. La base de données utilisée n'étant pas publique j'ai eu besoin d'installer un antivirus et un VPN. Donc en attendant d'avoir accès à la base de données en entier j'ai fait quelques tutoriels pour découvrir Unity. Unity est un moteur de jeu multiplateforme qui permet de créer des environnements virtuels ainsi que des jeux vidéo. Il permet aussi de créer ces environnements dans le but de formations.

Le but de ces premiers jours était de prendre en main le logiciel. J'ai donc pu réaliser quelques petits projets pour utiliser quelques fonctionnalités de Unity. J'ai commencé simplement en ajoutant des objets 3D à la scène comme des cubes, en leur donnant des caractéristiques physiques comme une masse pour que cet objet ne flotte pas ou qu'il ait un « *boxcollider* » (*boîte de collision*) pour qu'on ne puisse pas passer à travers.

Le deuxième petit projet a été de réaliser un petit jeu de plateforme mais sans ligne de code. En effet en utilisant le menu « Animation » de Unity on peut faire de petites animations très simplement. Par exemple dans le jeu de plateforme si dessous la première plateforme bouge verticalement pour faire monter le personnage puis la deuxième plateforme bouge horizontalement pour déplacer le personnage et lui permettre d'accéder à la fin du parcours.

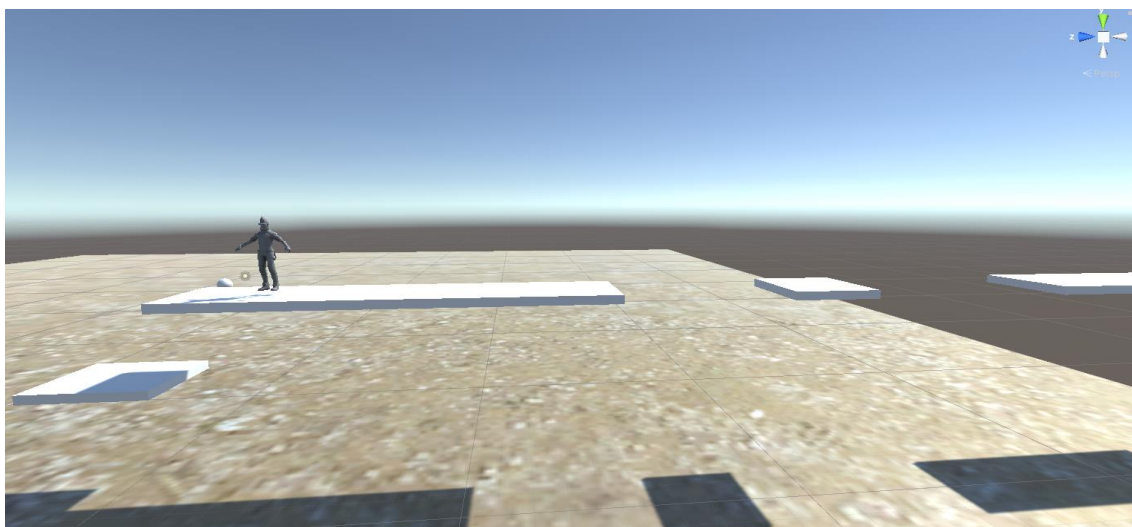


Figure 7 - jeu de plateforme

Le troisième projet consistait à importer des objets 3D. En effet dans le contexte du stage il me sera demandé d'importer les objets créés en dehors de Unity, comme sur Blender qui est un logiciel de modélisation. Il existe aussi des sites comme 3dwarehouse ou turbosquid qui proposent des objets 3D déjà créés gratuitement. Il suffit ensuite de les importer dans Unity. Il existe plusieurs façons de créer un objet 3D de cette façon et la différence est surtout au niveau du nombre de polygones que contient l'objet. En effet plus il contient de polygones et plus l'objet va pouvoir être séparé en de plus petits objets qui le composent. Par exemple, sur l'image ci-dessous, les murs, la table, les portes, etc..., sont indépendants et peuvent donc être bougés indépendamment les uns des autres. Cela est aussi utile pour mettre en place les « *boxcolliders* » afin que l'on ne puisse pas passer au travers des objets. Mais si l'objet ne contient pas assez de polygones alors le « *boxcollider* » sera juste une grande boîte et on ne pourra pas rentrer dans la maison par exemple.

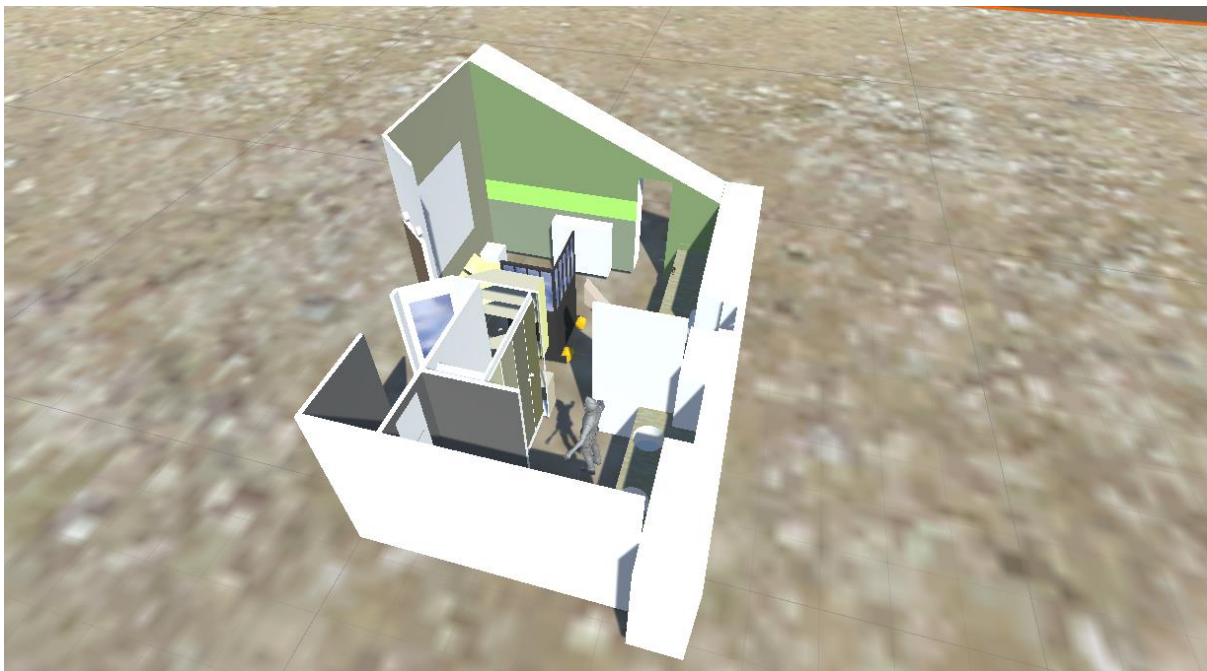


Figure 8 - import d'un modèle d'une maison

Le dernier projet est la création d'un terrain. En effet Unity dispose d'outils pour créer des montagnes ou des vallées et des creux à partir d'un terrain plat. On peut ainsi

facilement rajouter différents types d'arbres en éléments 3D ainsi qu'une texture d'herbes ou de sable selon l'endroit. On peut aussi rajouter un deuxième plan avec la texture de l'eau pour créer un lac. Cet outil est très simple à prendre en main et pourtant assez puissant car les terrains sont lisses et semblent très facilement réels en peu de temps et avec suffisamment de détails.



Figure 9 - création d'un terrain

3. Diagramme UML

Avant de pouvoir commencer à créer mes objets et à récupérer les informations disponibles de la base de données. J'ai créé deux modèles de données : un qui est représentatif des réponses données par l'API et un modèle métier. C'est ce dernier que je manipule dans le code ensuite.

Voici tout d'abord le modèle de la base de données :

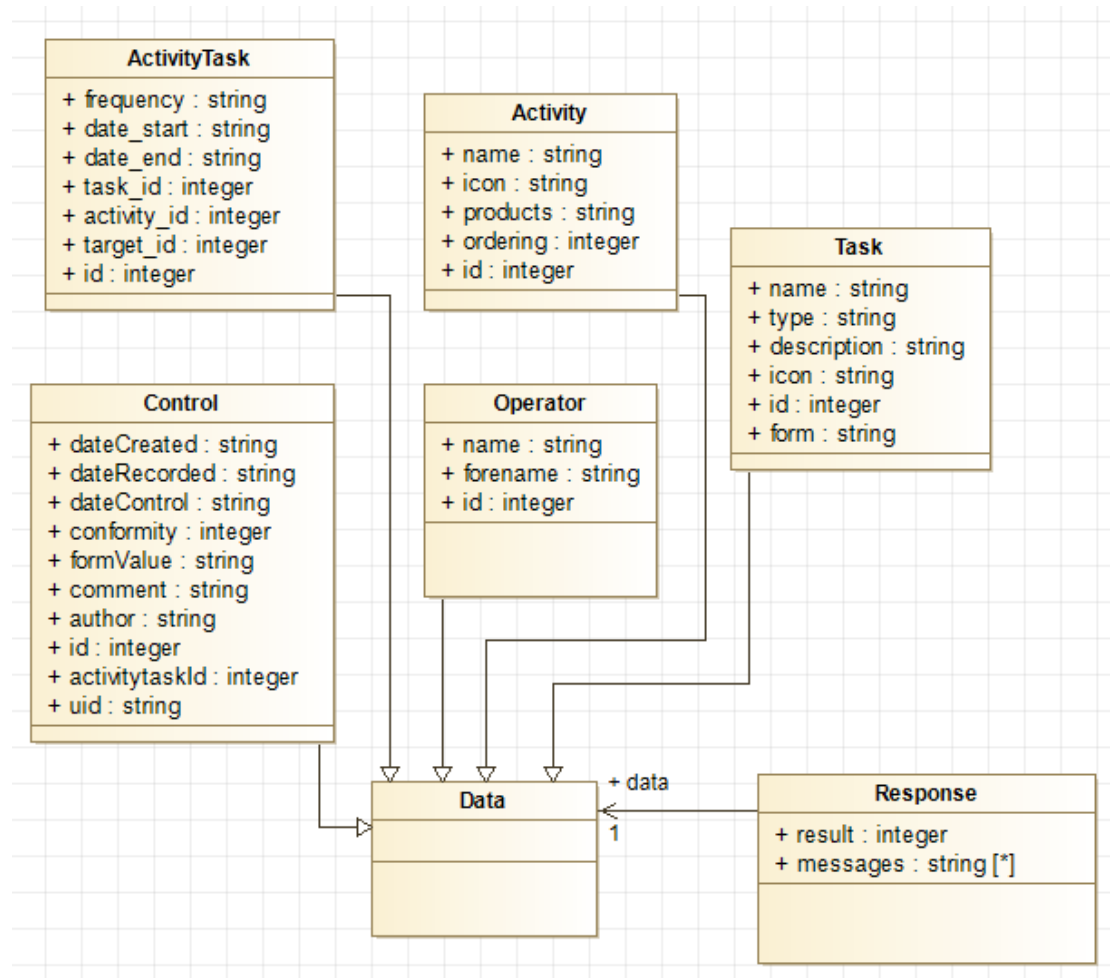


Figure 10 - diagramme UML base de données

Ce modèle de données est similaire aux réponses obtenues aux requêtes envoyé à l'api. La réponse est reçue dans le format JSON qui est le format de données textuelles dérivé de la notation des objets du langage JavaScript et utilisé de manière standard dans les échanges clients serveur.

Chaque réponse est reçue sous le même format : un champ « result » qui est à zéro lorsque la requête a montré une erreur et à un lorsque tout s'est bien passé, un champ « message » pour indiquer le motif de l'erreur s'il y en a une et un champ « data » qui contient les informations voulues.

```
{
  "result": 0,
  "messages": {
    "error": "string"
  },
  "data": {
  }
}
```

Figure 11 - exemple de réponse de l'API

Il y a ensuite 5 classes importante :

■ Activity

Une activité correspond à un laboratoire. Par exemple cela va désigner la boucherie ou la boulangerie. Chaque activité a un identifiant unique, un nom, une icône pour la représenter sur la tablette, une liste de produits associés et un ordre pour lire les activités.

■ Task

La classe Task correspond à toutes les tâches à faire mais avant la planification. Chaque tâche a, comme pour une activité, un identifiant unique, un nom et une icône pour l'identifier, une description pour décrire la tâche à effectuer et un type.

■ ActivityTask

La classe ActivityTask est une classe qui permet la réunion entre un laboratoire et une tâche. Cela concerne les tâches planifiées et donc les contrôles qu'il faut réaliser. Chaque objet de cette classe possède un identifiant unique, l'identifiant de la tâche et l'identifiant de l'activité pour qu'elles soient liées, l'identifiant de la cible (comme le bac de congélation où il faut contrôler la température), une fréquence de contrôle, et une date de début et de fin. La date de début et de fin ou la fréquence peuvent être nuls si la tâche planifiée est seulement ponctuelle.

■ Control

La classe Control concerne tous les contrôles qui ont été réalisés. Elle possède, de la même façon, un identifiant unique, un identifiant lié à l'utilisateur, l'identifiant de la ActivityTask liée au control, la date de création du contrôle, la date du contrôle en lui-même et la date où le control est enregistré dans la base de données. Elle possède aussi un attribut pour dire si le contrôle est conforme ou non, le formulaire, contenant les données comme la température mesurée, rempli, un commentaire et l'auteur du control.

■ Operator

Les opérateurs sont les utilisateurs de la tablette. Ils sont identifiés par un identifiant unique, un nom et un prénom.

Une autre requête importante concerne celle sur les métadonnées. Cette requête concerne toutes les données sur les produits et sur les fréquences. Il est donc très important de récupérer toutes ces données pour pouvoir réaliser le scénario. Cependant le modèle de donnée est un peu plus compliqué que celui des classes d'avant et posait des problèmes au niveau du nom des attributs.

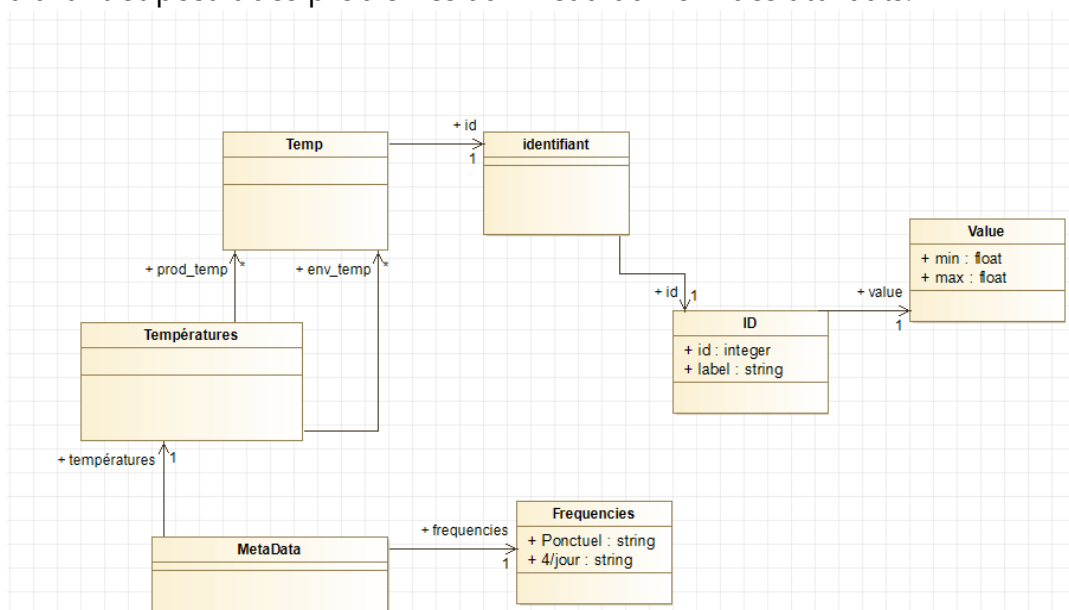


Figure 12 - modèle UML métadonnées

A la place de coder le modèle de donnée tel quel j'ai donc décidé d'utiliser les fonctions que proposent le module me permettant de manipuler le fichier JSON et donc d'utiliser la méthode `selectToken` qui permet de chercher directement l'attribut voulu et donc de ne pas passer par des classes. Cette façon de faire est un peu plus longue que de passer par les classes donc je n'ai utilisé cette méthode que pour collecter les métadonnées.

Ce modèle de donnée est simple mais moins facilement manipulable. En effet il y a des répétitions comme l'identifiant qui peut être regroupé et à la place d'avoir les identifiants dans `ActivityTask` par exemple, il est plus simple d'avoir directement une référence sur la tâche. J'ai donc fait le deuxième diagramme UML qui lui est un modèle métier et donc plus facilement manipulable.

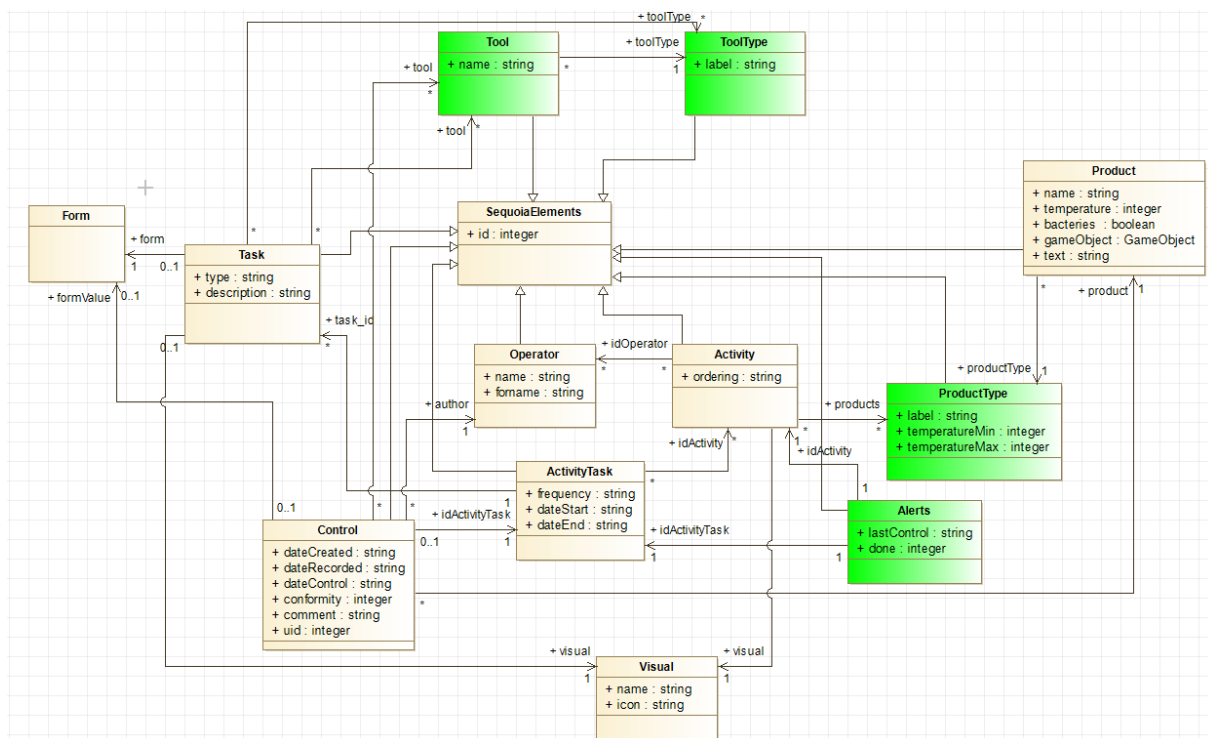


Figure 13 - Diagramme UML modèle Métier

Ce diagramme reprend les mêmes éléments que le diagramme précédent avec les tâches, les contrôles, les activités et les opérateurs mais en mettant des références aux autres classes plutôt que d'avoir seulement les identifiants. Il y a aussi la classe `Product` qui concerne tous les produits qui a été séparé des activités et qui contient désormais

un nom, une température et un indicateur pour savoir s'il présente des bactéries. Il contient aussi la référence vers le GameObject (l'objet dans la scène Unity) et un texte à afficher. Les classes qui sont affichées en vert sont les classes qui n'étaient pas présente dans le modèle de données initial. En effet pour différencier le type d'un produit et le produit en lui-même j'ai créé une classe ProductType qui contient le label et les températures maximales et minimales entre lesquelles le produit doit être. J'ai procédé de la même façon avec les outils en créant les classes Tool et ToolType. La difficulté avec ces deux classes c'est qu'elles ne sont pas présentes dans la base de données donc c'est à moi de fournir les éléments nécessaires pour le scénario.

Toutes les classes héritent de SequoiaElements car chaque classe à un identifiant propre.

J'ai aussi intégré la classe Alerts qui n'est pas une table dans la base de données mais j'ai choisi de l'intégrer dans le diagramme de classe. Les alertes sont la liste des contrôles à effectuer dans la journée. Cette liste contient uniquement les contrôles programmés avec une fréquence et, par exemple, un même contrôle apparaîtra 2 fois s'il faut faire le contrôle 2 fois par jour.

4. Requêtes http

L'application Sequoia demandant de faire des requêtes HTTP j'ai dû me familiariser avec ces requêtes dans Unity. Une requête http, acronyme de « Hypertext Transfer Protocol », permet d'envoyer un message à la base de données et de recevoir sa réponse. Il existe différents types de requêtes en fonction de ce que l'on souhaite demander. En effet les requêtes que je dois utiliser sont les requêtes « get » et « post ». Les requêtes « get » sont des requêtes pour récupérer uniquement des informations contrairement aux requêtes « post » où on envoie des informations à la base de données. Pour effectuer ces requêtes j'ai pu utiliser cURL qui est un outil en ligne de commande qui permet de faire des requêtes http, Postman qui est un logiciel permettant de tester une API et UnityWebRequest qui est un module Unity nécessaire pour faire des requêtes HTTP. J'ai commencé mes requêtes sur l'API de bibus qui est libre. Cela m'a permis de comprendre comment les outils et les requêtes fonctionnaient.

■ Requête GET

Pour faire une requête cURL il suffit de mettre la commande voulue (ici GET) avec l'URL et les paramètres souhaités (ici -k pour ignorer la validation du certificat)

```
C:\Users\cerv>curl -GET https://sequoia-labo.emanrisk.lan/ping -k  
{"result":1,"messages":[],"data":{"result":"Pong !","version":"API 0.95 b"}}
```

Figure 14 - exemple requête cURL

Ici le résultat est reçu sous la forme JSON. Je faisais en premier les requêtes sur cURL pour vérifier tout d'abord que l'URL était bien fonctionnelle. Mon maître de stage m'a aussi fourni un fichier JSON à importer dans Postman contenant toutes les informations pour faire fonctionner les requêtes.

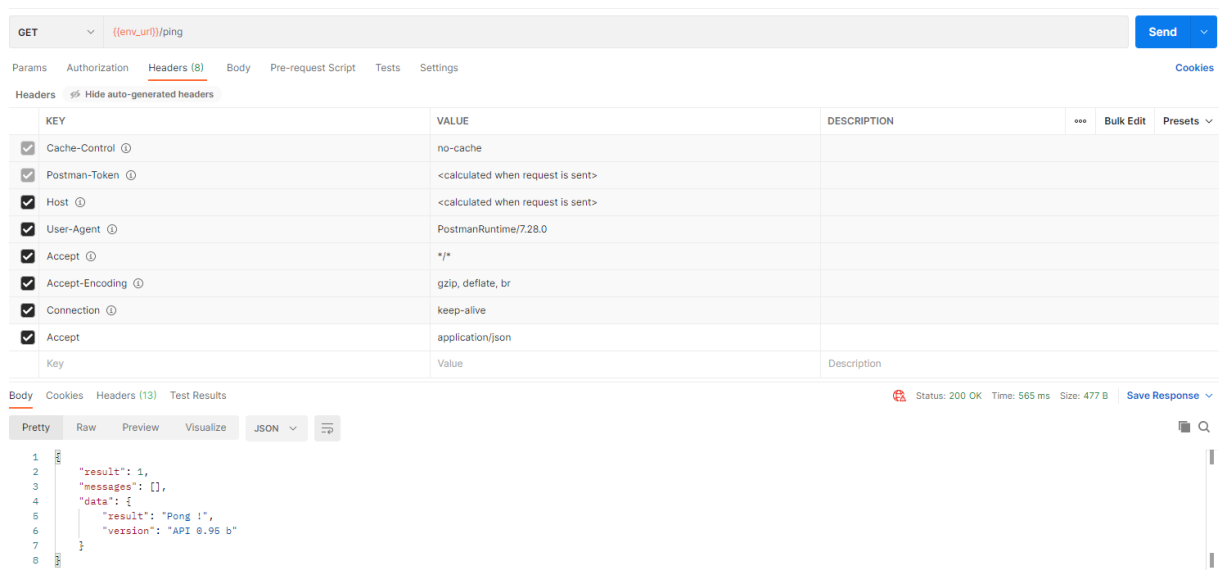


Figure 15 - interface Postman

J'ai utilisé ces outils en compléments du module UnityWebRequest afin d'identifier les différentes erreurs que je pouvais avoir. En effet faire la requête sur Unity en C# demande plus d'étapes et est donc plus sources d'erreurs. Je pouvais donc vérifier que si une même requête fonctionnait sur Postman ou sur cURL mais pas sur Unity que le problème venait bien de moi et donc corriger en fonction. Cela m'a aussi permis d'aller demander de l'aide quand une requête ne fonctionnait pas sur tous les différents outils.

```

IEnumerator GetPing()
{
    Debug.Log("getPing");
    UnityWebRequest www = UnityWebRequest.Get(url + "ping"); //méthode Get avec l'url voulue
    www.SetRequestHeader("Content-type", "application/json"); //Header de la requête

    www.certificateHandler = new WebRequestCert(); //certificat de validation (-k sur curl)

    yield return www.SendWebRequest(); //envoi de la requête
    if (www.isNetworkError || www.isHttpError) // on regarde s'il y a une erreur
    {
        Debug.Log(" ----> ERREUR !");
        Debug.Log(www.error); // on affiche l'erreur
    }
    else
    {
        Debug.Log(www.downloadHandler.text); // on affiche la réponse
        // ensuite on peut manipuler la reponse reçue
    }
}

```

Figure 16 - requête http en C#

■ Requête POST

Les requêtes POST sont plus compliquées à mettre en place car ces requêtes demandent des informations qui doivent être bien mise en forme. J'ai eu plusieurs types de problèmes avec la requête contro1. Le problème principal est la mise en forme des données à envoyer. En effet la première requête que j'ai fait sur l'API Sequoia concerne l'authentification. Il faut donc mettre comme paramètre au format JSON le nom d'utilisateur et le mot de passe. Sur Unity pour obtenir ce format il suffit de créer une classe contenant tous les paramètres nécessaire et de sérialiser ensuite cet objet en JSON. Il suffit ensuite de transmettre cette chaine de caractère en paramètre de la fonction Post pour configurer la requête.

```
Authentication auth1 = new Authentication();  
auth1.username = "username";  
auth1.password = "password";  
string rep = JsonConvert.SerializeObject(auth1);  
  
UnityWebRequest www = UnityWebRequest.Post(url+ "login_check", rep);
```

Figure 17 - requête post authentication

La deuxième requête importante en envoyant des informations à la base de données est la requête permettant de faire un contrôle. En effet les informations à envoyer sont plus développées car, par exemple, il faut envoyer un formulaire qui est lui-même en format JSON, ce qui implique de créer 2 classes, avec la première classe ayant un attribut du type de la deuxième. La façon de faire sur unity est un peu plus longue car elle implique de créer plus d'objets temporaires

```

{
  "uid": "01234567",
  "date_created": "2021-05-03 10:10:10",
  "activitytask_id": 281,
  "date_control": "2021-05-03 10:00:00",
  "form_value": [
    {
      "uid": "57b352ebe3121fed",
      "value": "5",
      "metadata": {
        "id": "101",
        "label": "test",
        "value": {
          "min": "10",
          "max": "20"
        }
      }
    },
    {
      "uid": "7249672e90c84455",
      "value": "8"
    },
    {
      "uid": "b08a2bed0998dfb2",
      "value": "yes"
    }
  ],
  "comment": "Lorem ipsum dolor sit amet...",
  "author": "TEST API"
}

```

```

ControlAPI control1 = new ControlAPI();
control1.uid = "01234567";
control1.date_created = "2021-04-27 10:10:10";
control1.activitytask_id = 281;
control1.date_control = "2021-05-11 10:00:00";
control1.comment = "Lorem ipsum dolor sit amet ...";
control1.author = "TEST API";
FormValueAPI valueform1 = new FormValueAPI();
FormValueAPI valueform2 = new FormValueAPI();
FormValueAPI valueform3 = new FormValueAPI();

valueform1.uid = "57b352ebe3121fed";
valueform1.value = "5";
valueform1.metadata.id = "101";
valueform1.metadata.label = "test";
valueform1.metadata.value.min = "10";
valueform1.metadata.value.max = "20";

valueform2.uid = "7249672e90c84455";
valueform2.value = "8";

valueform3.uid = "b08a2bed0998dfb2";
valueform3.value = "yes";

List<FormValueAPI> listform = new List<FormValueAPI>();
listform.Add(valueform1);
listform.Add(valueform2);
listform.Add(valueform3);

control1.form_value = listform;

string rep = JsonConvert.SerializeObject(control1);

```

Figure 18 - comparaison format JSON et C#

Cette mise en forme est aussi totalement dépendante du format de la base de données car si de ce côté les paramètres changent et que je n'ai pas actualisé de mon côté la requête n'aboutira pas et cela peut être long pour trouver l'erreur au début.

5. Eléments 3D

Après avoir créé les classes et effectuer toutes les requêtes j'ai pu instancier mes objets. J'ai commencé par créer un dictionnaire contenant tous mes productsType à partir des informations récoltées dans les métadonnées. Ensuite j'ai instancié toutes les informations nécessaires pour les tâches, les activités, les opérateurs, les activityTask, les alertes, et les contrôles. J'ai aussi créé des fichiers XML, qui est un langage de structuration de données un peu comme le JSON, afin d'instancier proprement toutes les informations que je dois créer pour réaliser le scénario.

Ensuite j'ai pu utiliser les objets 3D déjà créer pour les associer aux produits créer. Cette association permet d'afficher sur la scène les métadonnées et de détecter si l'objet est touché plus facilement.



Figure 19 - objets 3D produits

Les étudiants de 5^{ème} année faisant leur module de projet informatique sur ce projet ont modélisé tous les objets nécessaires à la réalisation du scénario. Ils ont d'abord commencé par faire l'environnement global du E.Leclerc, c'est-à-dire les murs, le sol et le mobilier fixe et simple à réaliser comme les tables ou les étagères.

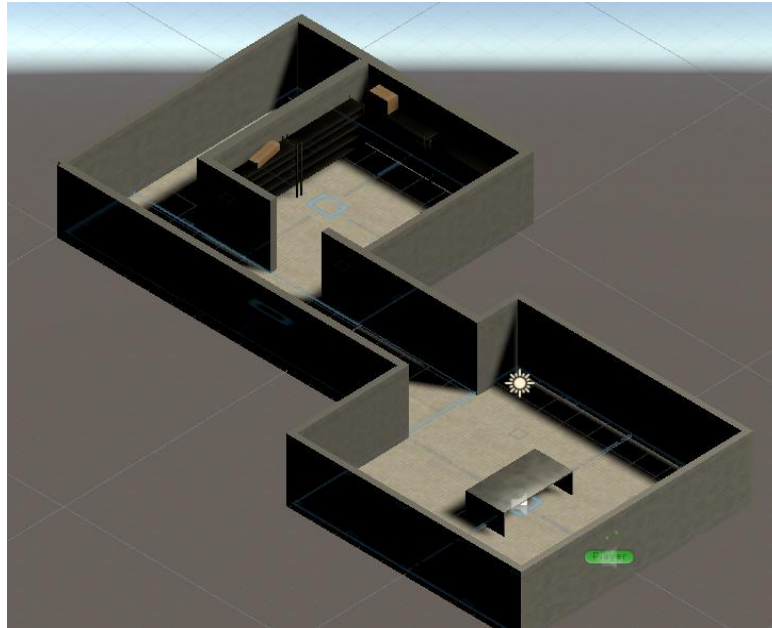


Figure 20 - environnement global

Ensuite les étudiants ont créé tous les objets mobile ou plus compliqué comme le distributeur de savon ou de serviette, ou le lavabo.



Figure 22 - distributeur de savon et de serviette



Figure 21 - lavabo



Figure 24 - four et frigidaire



Figure 23 - thermomètre et sonde

Le but est aussi de rendre ces outils utilisables avec, par exemple, la possibilité d'ouvrir la porte du four ou de placer des produits dans le frigo. Il y a aussi des animations pour faire couler l'eau du robinet ou donner du savon au distributeur de savon.

6. Réalisation du scénario

Après avoir récupéré tous les objets 3D j'ai pu commencer à réaliser le scénario.

Le scénario donné en annexe est le scénario idéal donné par e-ManRisk, il doit donc être adapté à ce qui est possible de faire en réalité virtuelle et/ou augmentée. Dans le temps imparti le but n'est pas d'avoir le scénario parfait mais de mettre en place des actions à réaliser similaires à ce que ferait une personne réalisant cette formation. Le scénario que j'ai réalisé est donc une version simplifiée du scénario donné mais en reprenant les éléments clés.

a. Le lavage des mains

Le lavage des mains est la tâche la plus compliqué à réaliser. En effet le scénario demande de vérifier que la personne fait bien les bons gestes pendant un temps mesuré. Or il est difficile pour l'utilisateur du casque de faire les mouvements appropriés avec les manettes dans les mains et il est difficile de capter les bons mouvements sans avoir de capteurs précis.



Figure 25 - mouvements de mains à réaliser

Il est aussi précisé dans le scénario que les objets ne sont pas à manipuler avec les mains mais par exemple il faut activer le lavabo avec le genou ou ouvrir la poubelle la poubelle avec le pied. Cependant sans capteur adapté cela est très difficile et dans les environnements virtuels généralement tout est manipulable par les mains car elles sont représentées par les manettes. Pour l'instant cette partie du scénario n'a pas été programmer mais les pistes envisagées sont de faire couler l'eau ou ouvrir la poubelle quand le joueur s'approche de l'objet mais il y aura aussi la possibilité de les ouvrir avec les mains afin de pouvoir montrer que c'est une erreur comme précisé sur le scénario. Les mains seront lavées de façon plus simple avec un temps à rester sous l'eau avec du savon et prendre une serviette pour s'essuyer les mains ensuite. Il doit aussi y avoir la notion de bactéries qui doit entrer en jeu. En effet tant que les mains ne sont pas lavées il y aura un effet visuel montrant les bactéries dessus et sur toutes les surfaces non stériles. Le lavage des mains (s'il est bien fait) aura donc pour but d'éliminer toutes les bactéries présentes, et si l'utilisateur touche des surfaces contaminées avec ses mains alors l'effet avec les bactéries reviendra. Cela permettra à l'utilisateur de bien montrer l'importance de l'hygiène des mains avant d'effectuer ses tâches.

b. La cuisson d'un produit

La deuxième partie du scénario est la cuisson du produit. Cette étape est très importante car elle permet l'élimination de toutes les bactéries à partir d'une certaine température, ici 63°C. Cette température est importante car c'est la température à laquelle la salmonelle, qui est la bactérie la plus thermo résistante, est détruite. Le but de cette partie du scénario est donc de désinfecter la sonde, prendre la température d'un produit, de le mettre à chauffer et de surveiller sa température jusqu'à ce qu'il atteigne les 63°C et désinfecter la sonde après l'avoir enlevé. La phase de désinfection sera la plus compliqué à mettre en œuvre car il faut pouvoir manipuler un objet et le rendre souple ce qui est plus compliqué à modéliser que de faire un objet « dur ». Pour l'instant toute la programmation de prise de température est déjà réalisée et il manque

la modélisation de la lingette de désinfection et le programme pour faire cuire le produit quand on le place dans le four. Dans cette partie aussi il y aura des effets pour simuler les bactéries sur le produit pas encore cuit ou sur la sonde lorsqu'elle n'est pas désinfectée.

c. Le refroidissement d'un produit

La troisième partie du scénario concerne le refroidissement d'un produit. Il va se passer sensiblement de la même façon que la cuisson du produit mais avec des protocoles plus stricts car il faut que la température du produit passe de 63°C à 10°C en moins de 2 heures sinon le produit sera considéré comme contaminé et il sera jeté. Le produit devra donc être placé dans une cellule de refroidissement, puis on devra surveiller sa température et attendre qu'il arrive à 10°C et le noter sur la tablette. Il y aura aussi la notion du temps à surveiller même si pour les besoins de la simulation il sera accéléré. Dans cette partie la prise de température par la sonde est déjà programmée mais il manque encore l'intégration de la tablette qui devrait être faite par les étudiants de 5^{ème} année. A partir de la tablette on pourra faire un contrôle de la température du produit et la rentrer dans la base de données. Il manque aussi toute la gestion du temps pour calculer si les deux heures sont passées et comment on les symbolise dans l'environnement virtuel.

d. Le stockage en chambre froide

La dernière partie du scénario concerne le stockage du produit en chambre froide. Ce dernier est assez court car il suffit de protéger le produit qui est sur une échelle, puis de lui mettre une étiquette avec toutes les informations et de mettre cette échelle dans la chambre froide. La partie la plus difficile à mettre en œuvre sera de simuler qu'on a mis une bâche sur l'échelle car la bâche étant un objet souple il sera plus difficile à modéliser. Cette partie ne sera effectuée que si le produit n'est pas contaminé et n'a donc pas été jeté.

V. Problèmes rencontrés

Pendant ce stage j'ai rencontré quelques problèmes. Premièrement les problèmes se sont posés au niveau de la réalisation pratique du scénario. En effet certains éléments, comme je l'ai détaillé dans le scénario, comme le lavage des mains sont très compliqué à mettre en place et demandent des compromis. Dans ces cas-là il est bien de dire que la réalité virtuelle n'est pas la plus adapté pour apprendre les gestes d'hygiène comme le lavage de mains. Certains objets sont aussi compliqués à rendre réaliste dans l'environnement virtuel du fait de leur caractéristiques physiques. En effet le câble reliant la sonde au thermomètre n'est pas présent car il est difficile de modéliser le câble souple et de prévoir ses mouvements.

Un autre problème du stage a été le télétravail. En effet au vu de la situation sanitaire j'ai passé une bonne partie de mon stage à travailler de chez moi. J'avais les outils nécessaires pour travailler dans de bonnes conditions mais il manquait le contact humain et parfois le cadre pour travailler efficacement. Je suis quand même présente sur le Chat de e-ManRisk et cela m'a permis de profiter de la bonne ambiance de l'entreprise.

Comme je travaillais en collaboration avec des étudiants de 5^{ème} année parfois les périodes de vacances pour eux était un peu plus compliqué car ils n'étaient pas sur le projet et j'avais donc parfois du mal à avancer notamment tant que les différents objets 3D n'avaient pas été créés.

VI. Bilan et perspectives

Ce stage m'aura beaucoup apporté sur les plans techniques et les plans personnels et il me reste encore un mois à travailler dans l'entreprise. Premièrement cela m'a permis de pouvoir apprendre à travailler en entreprise et voir l'ambiance qu'il peut y avoir avec des collègues. J'ai aussi pu avoir une première vision de ce que cela pourrait être de travailler dans la réalité virtuelle qui est mon objectif professionnel et cela me conforte dans cette idée car j'ai beaucoup aimé faire ce stage. J'ai pu me familiariser avec des logiciels qui me seront utiles dans le monde du travail ensuite si je continue dans cette voie et j'ai participé à un projet vraiment concret et très intéressant. Cela m'a aussi permis de voir que je préfère créer des applications en réalité virtuelle plus dans un but de formation et d'apprentissage que pour faire des jeux vidéo, qui était ma motivation principale pour travailler dans la réalité virtuelle par le passé.

J'ai aussi acquis des compétences dans le domaine des requêtes HTTP ou de l'utilisation d'Unity.

Pour les perspectives j'aimerais pouvoir finir le scénario le mieux possible avec tous les éléments que je peux mettre en place afin qu'il serve de base pour un prochain projet. En effet, ce projet a atteint ses objectifs en termes de preuve de concept et a permis de définir le contour des briques techniques pour l'avancée du projet. Je dois aussi faire une présentation à la fin de mon stage dans les locaux de e-ManRisk afin de montrer à l'équipe mon travail et qu'ils puissent tester par eux même le scénario ainsi créé.

VII. Conclusion

J'ai pu vous montrer le travail effectué pendant les trois premiers mois du stage. Certaines parties comme la réalisation du scénario sont encore assez peu développées mais j'ai mis en place tous les briques nécessaires à la réalisation du scénario et dans les prochains mois je vais les développer afin de montrer les possibilités que l'on peut obtenir. E-ManRisk aura tous les éléments nécessaires pour la réalisation de ce projet.

Je pense que ce stage m'apportera aussi beaucoup de choses pour ma connaissance du monde du travail car, n'ayant pas pu faire le stage technicien, c'est ma première expérience de plusieurs mois en entreprise. Je saurai aussi surement mieux appréhender le télétravail qui sera certainement présent dans mon futur emploi.

VIII. Bibliographie

Tutoriel Youtube pour la création de textures sur Unity :

<https://www.youtube.com/watch?v=2oV-YN2IQGg&list=PLDQKHJxRCgUlpRYCPdQgmb0qFIFGJSjOI&index=3>

(consulté le 1 mars 2021)

Tutoriel Youtube pour la création d'animations sur Unity :

<https://www.youtube.com/watch?v=zGyj9d1L9so&list=PLDQKHJxRCgUlpRYCPdQgmb0qFIFGJSjOI&index=5> consulté le 2 mars 2021

Tutoriel Youtube pour la création de terrain sur Unity :

<https://www.youtube.com/watch?v=aG8SSD4PQ10&list=PLDQKHJxRCgUlpRYCPdQgmb0qFIFGJSjOI&index=4> consulté le 3 mars 2021

Site Web contenant des objets modéliser en 3D à télécharger :

<https://www.turbosquid.com/> consulté le 10 mars 2021

Site Web contenant des objets modéliser en 3D à télécharger :

<https://3dwarehouse.sketchup.com/> consulté le 10 mars 2021

Site Web Postman :

<https://web.postman.co/home> consulté le 26 mars 2021

Documentation officielle de Unity :

<https://docs.unity3d.com/Manual/index.html> consulté le 29 mars 2021

IX. Annexes

Objet	Description d'un scénario dans le cadre du projet RV
Date	Le 11 mars 2021
Participants	S/O

Version	Auteur	Date	Description
1.0	Nicolas SIMON	11/03/2021	Création initiale

Résumé

Le projet consiste à construire un environnement virtuel qui vise à proposer à nos clients un support original de formation à l'hygiène alimentaire en mode Réalité virtuelle.

Le but de ce document est de rédiger un premier scénario en décrivant l'environnement ainsi que son déroulement qui sera proposé aux personnes formées.

Il s'agit d'intégrer le salarié dans un environnement de travail similaire au sien en lui proposant une série d'actions et leur conséquence (conforme ou non-conforme) dans le cadre de la maîtrise de la sécurité alimentaire.

Pour ce premier scénario, je propose d'amener l'opérateur dans une opération de cuisson, de refroidissement rapide et de stockage.

Détail du scénario

Tâche : l'opérateur se verra confier quatre missions principales :

- Le lavage des mains
- La cuisson
- Le refroidissement d'un plat
- Le stockage de ce plat dans une chambre froide

Locaux visités :

- Laboratoire de fabrication
- Couloir
- Chambre de stockage

Matériels nécessaires :

- Lave main à commande non manuelle – distributeur à savon et à essuie main
- Boîte de gants
- Bac gastro et son couvercle
- Poubelle à pédale
- Thermomètre à sonde, bouton ON/OFF, afficheur
- Echelle de stockage
- Armoire de refroidissement
- Housse de protection
- Etiquette balance vierge
- Marqueur

Etape 1 : le lavage des mains

Avant toute opération de fabrication, l'opérateur doit se laver les mains selon des règles très précises.

Matériel nécessaire :



- ✓ **Scénario idéal** : utilisation du lave-main à commande non manuelle (allumage de l'arrivée d'eau à l'aide du genou par pression sur plaque frontale), une pression sur le distributeur de savon pour une dose puis lavage des mains sur une durée minimale de 30 secondes en respectant le protocole ci-dessous :



Après le lavage, le rinçage, encore une fois à l'aide de la commande non manuelle. Enfin le séchage par **tapotage** d'un papier à usage unique sur toute la surface lavée. Le papier à usage unique est ensuite jeté à la poubelle qui sera ouverte avec le pied.



Tous les autres scénarii possibles seront non conformes avec des degrés de gravité variés selon l'erreur commise. Suggestion : signifier l'erreur ou le manquement par un visuel fort (croix rouge ?) et faire comprendre l'erreur par une vision virtuelle du danger.

Ex : si l'opérateur se saisit d'un torchon que le scénario prévoit de laisser à disposition pour s'essuyer les mains (piège), prévoir une visualisation virtuelle des colonies de bactéries présentes sur le support interdit.

- ✓ **Scénario 1** : manipuler la commande d'eau non manuelle à la main
- ✓ **Scénario 2** : manquer l'une ou l'autre (ou plusieurs) des étapes du protocole de lavage des mains
- ✓ **Scénario 3** : se laver les moins en moins de 30 secondes
- ✓ **Scénario 4** : s'essuyer les mains avec un torchon
- ✓ **Scénario 5** : s'essuyer les mains (au lieu du tapotage)
- ✓ **Scénario 6** : ouvrir la poubelle pour jeter l'essui main avec les mains (le couvercle) sans utiliser la pédale.

Etape 2 : cuisson

La cuisson a pour importance microbiologique de détruire les bactéries pathogènes.

Parmi celles-ci, la salmonelle est la plus thermo résistante. Mais elle-même, à 63°C est détruite. L'objectif de cette première étape est donc de s'assurer que le thermomètre affiche au minimum la température de +63°C.

Le plat est mis en cuisson dans une gamelle sur un piano. Le thermomètre est disposé dans l'aliment pour vérifier sa température de cuisson.

- ✓ **Scénario idéal** : l'opérateur doit
 - Enfiler des gants
 - Allumer le thermomètre
 - Désinfecter la sonde du thermomètre (avec une lingette désinfectante)
 - Plonger la sonde à cœur du produit
 - A l'issue de la cuisson, l'opérateur doit penser à désinfecter la sonde du thermomètre.
 - Jeter la lingette à la poubelle en utilisant la pédale (sans toucher le couvercle).

Matériel nécessaire (de gauche à droite : gants, lingettes désinfectantes, thermomètre à sonde).



Option virtuelle : symboliser la montée en température avec un thermomètre graphique qui symbolise le passage en zone saine (au-dessus de $+63^{\circ}\text{C}$).

Symboliser la présence de bactéries en grand nombre sur certaines surfaces : la sonde du thermomètre, la commande non manuelle du lave main, le couvercle de la poubelle (présence +++).

Les scénarii

- ✓ Scénario 1 : arrêt de la cuisson avant la température cible, représentation virtuelle de la présence microbienne.
- ✓ Scénario 2 : pas de désinfection de la sonde du thermomètre
- ✓ Scénario 3 : pas de port de gants.

Etape 3 : le refroidissement

Théorie microbiologique : obligation réglementaire de passer d'une température supérieure à $+63^{\circ}\text{C}$ à une température inférieure à $+10^{\circ}\text{C}$ en moins de 2 heures.

La zone thermique $+63$ à $+10^{\circ}\text{C}$ est critique en microbiologie et doit être franchie le plus rapidement possible pour minimiser le risque de contamination et de multiplication microbienne.

Une représentation virtuelle d'un thermomètre et la descente du mercure est envisageable avec un changement de couleur selon la tranche de température : $> +63^{\circ}\text{C}$ (vert), entre $+10$ et $+63^{\circ}\text{C}$ (rouge), inférieure à $+10^{\circ}\text{C}$ (vert).

- ✓ **Scénario idéal** : l'opérateur doit
 - Placer un bac gastro (avec l'aliment) sur une échelle
 - Placer l'échelle dans l'armoire de refroidissement
 - Fermer la porte de l'armoire de refroidissement
 - Lancer le programme
 - Attendre maximum deux heures
 - Désinfecter la sonde du thermomètre de la cellule de refroidissement
 - Planter la sonde à cœur du produit
 - Attendre que la température se stabilise sur l'afficheur.
 - Si cette température est inférieure à $+10^{\circ}\text{C}$, la noter (sur tablette ?)
 - En cas de présentation de l'outil tablette, présenter un formulaire Refroidissement rapide.
 - Si la température est supérieure à $+10^{\circ}\text{C}$ et que les 2 heures ne sont pas écoulées, poursuivre le refroidissement.

- Si la température est supérieure à $+10^{\circ}\text{C}$ et que les 2 heures sont écoulées, jeter le produit (potentiellement contaminé).
- Récupérer l'échelle, noter la température affichée sur le thermomètre
- L'étape suivante (stockage) ne peut être réalisée que si $T^{\circ}\text{C}$ inférieure à $+10^{\circ}\text{C}$ en moins de 2 Heures.

Matériel nécessaire : (de gauche à droite : échelle, bac gastro, cellule de refroidissement).



- ✓ Scénario 1 : pas de désinfection de la sonde du thermomètre
- ✓ Scénario 2 : durée supérieure à 2H
- ✓ Scénario 3 : température supérieure à $+10^{\circ}\text{C}$ et stockage de la marchandise
- ✓ Scénario 4 : température et durée non saisies sur support Tablette.

Etape 4 : le stockage en chambre froide.

Les choses importantes à savoir pour cette quatrième étape :

- Le froid conserve car limite le développement microbien sans le stopper.
- Comme il y a toujours multiplication bactérienne, la salubrité du produit a une limite de temps : c'est sa DLC à indiquer au marqueur sur une étiquette balance par exemple.
- Tout produit présent en chambre froide doit être identifié (quoi, fabriqué quand et DLC) et protégé (film, housse d'échelle).

Matériel nécessaire : (de gauche à droite : étagères de chambre froide, housse d'échelle, rouleau d'étiquettes vierges de balance).



- ✓ **Scénario idéal** : l'opérateur doit
 - Protéger le bac ou l'échelle

- Indiquer sur une étiquette nom, date de fabrication et/ou DLC
 - La stocker dans la chambre froide.
-
- ✓ Scénario 1 : stockage à nu
 - ✓ Scénario 2 : une ou plusieurs indications manquantes sur le bac
 - ✓ Scénario 3 : pas de stockage en CF.

Outils utilisés :

