

Rapport de stage quotidien

Début du stage le 27/05/2024
Fin du stage le 05/07/2024

| | |
|---------------------|----------|
| Semaine 1 | 2 |
| Lundi 27 mai | 2 |
| Mardi 28 mai | 2 |
| Mercredi 29 mai | 2 |
| Jeudi 30 mai | 2 |
| Vendredi 01 juin | 2 |
| Semaine 2 | 3 |
| Lundi 03 juin | 3 |
| Mardi 04 juin | 3 |
| Mercredi 05 juin | 3 |
| Jeudi 06 juin | 3 |
| Vendredi 07 juin | 4 |
| Semaine 3 | 4 |
| Lundi 10 juin | 4 |
| Mardi 11 juin | 4 |
| Mercredi 12 juin | 5 |
| Jeudi 13 juin | 6 |
| Vendredi 14 juin | 6 |
| Semaine 4 | 7 |
| Lundi 17 juin | 7 |
| Mardi 18 juin | 8 |
| Mercredi 19 juin | 8 |
| Jeudi 20 juin | 8 |
| Vendredi 21 juin | 8 |
| Semaine 5 | 8 |
| Lundi 24 juin | 8 |
| Mardi 25 juin | 8 |
| Mercredi 26 juin | 8 |
| Jeudi 27 juin | 9 |
| Vendredi 28 juin | 9 |
| Semaine 6 | 9 |
| Lundi 01 juillet | 9 |
| Mardi 02 juillet | 9 |
| Mercredi 03 juillet | 9 |
| Jeudi 04 juillet | 9 |
| Vendredi 05 juillet | 9 |
| Définitions | 9 |

Semaine 1

Lundi 27 mai

J'ai commencé mon stage à 9h, à mon arrivée j'ai fait connaissance avec les employés ainsi qu'avec les lieux de l'entreprise. J'ai ensuite enchaîné avec une présentation de quelques bases et quelques règles de l'entreprise présentées par un représentant(...) Aleksis, il m'a ensuite donné un ordinateur de fonction pour travailler et nous l'avons configuré.

Pour l'après-midi j'ai principalement lu des documents et la charte de l'entreprise.

Mardi 28 mai

Durant la matinée j'ai réalisé des commandes sur un terminal de mon pc, en suivant des étapes d'un document partagé par l'entreprise nommé "Processus de développement". J'apprends à me servir d'un pc sous linux ubuntu et j'apprends de nouvelles choses. J'ai continué durant l'après-midi à suivre ce document et j'ai également suivi une réunion en vidéoconférence d'une heure.

Mercredi 29 mai

Ma matinée était dans la continuité d'hier. Et l'après midi je l'ai passé avec mon maître de stage qui m'a expliqué un bout de leur projet et surtout en partie ce que je vais faire dans l'entreprise, j'ai donc une "tâche assignée" dans leur logiciel [JIRA](#)¹ que je commencerai très prochainement.

Jeudi 30 mai

J'ai été confrontée à quelques problèmes sur mon pc que j'ai dû régler durant la matinée, et ensuite j'ai eu une réunion de 11h30 à 13h. L'après-midi j'ai réglé un second problème afin de cloner un de leur code sur mon pc et je l'ai ensuite étudié durant la fin de ma journée.

Vendredi 01 juin

J'ai étudié le code et essayé de le corriger, afin de pouvoir le mettre à jour plus tard. Toutefois, il me manque certains éléments de l'entreprise (données sur [Nexus](#)²) auxquels je n'ai pas encore accès et donc le code ne peut fonctionner.

Semaine 2

Lundi 03 juin

J'ai passé une grande partie de ma journée à étudier le code java que je devrais mettre à jour (un simulateur). Pour le moment, je note les choses que je ne comprends pas afin de demander des explications à mon maître de stage par la suite. Et j'ai eu une réunion en visio de 10h30 à 12h.

Mardi 04 juin

Avec l'aide de mon maître de stage j'ai réussi à me connecter à leur serveur [Nexus](#)² et autres éléments importants afin d'avoir l'intégralité du code "fonctionnel". J'ai commencé les tests qui étaient déjà codés pour voir leur fonction en notant tout pour pouvoir comparer ensuite. J'ai eu dans la journée 2 réunions (10h30 à 11h et 16h à 17h).

Mercredi 05 juin

Mon maître de stage étant en déplacement sur le site de Valence toute la journée, j'ai donc continué mes tests du simulateur d'une caméra thermique pour un parking. Ainsi j'en ai créé de nouveaux pour vérifier le fonctionnement. L'objectif étant de mettre à jour le simulateur, je modifie donc le code en conséquence.

Résultat du code HikControllerTest :

```
11:31:58.875 [main] INFO com.netceler.ivpark.hik.hikSimulator.web.controller.HikControllerTest - testDescriptionResponse passed - ok
```

- **Heure** : 11:31:58.875
- **Thread** : [main]
- **Niveau de journal** : INFO
- **Classe** : `com.netceler.ivpark.hik.hikSimulator.web.controller.HikControllerTest`
- **Message** : Le test `testDescriptionResponse` a réussi (passed).

Jeudi 06 juin

Mon maître de stage étant en déplacement sur Grenoble toute la journée, j'ai travaillé sur le simulateur, ajouté des commentaires et debug(=corrigé) certains problèmes. Durant la matinée mon pc a crashé à cause des programmes trop lourds, comme ce n'est pas la première fois, je suis allée voir Aleksis (Responsable service technique) et il m'a ajouté 4 Go de swap(= **Espace de stockage en plus** : 4 Go à 16 Go de RAM correspond à un minimum de 4 Go d'espace swap), ce qui a fonctionné pour le reste de la journée.

Vendredi 07 juin

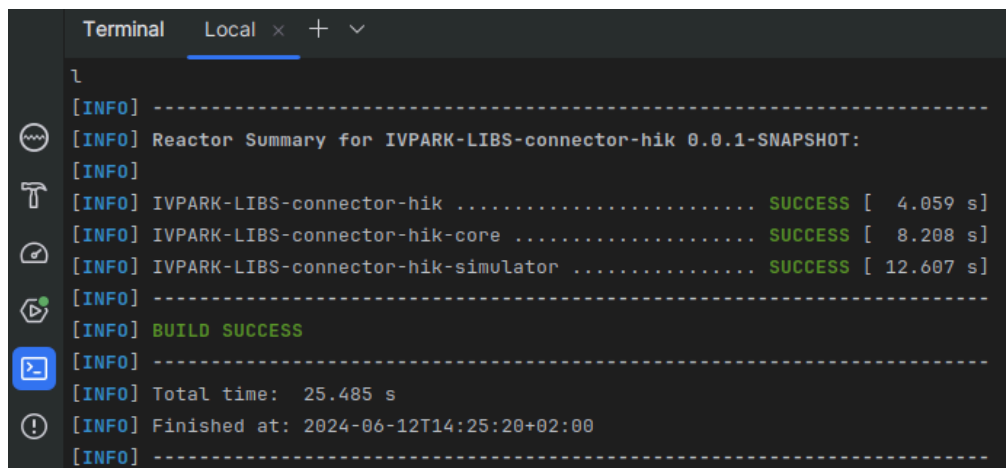
Début de la mise à jour du code du simulateur, changement de version de java de 11 à 21, puis après-midi avec Julien qui m'a aidé et m'a expliqué des éléments importants du projet et ce que je devrai faire lundi, ainsi que quelques fonctions de base qu'ont les employés dans l'entreprise. (Démonstration de logiciels, etc..).

Semaine 3

Lundi 10 juin

Aujourd'hui j'ai créé dans le simulateur, une nouvelle alerte pour une simulation d'une intrusion dans un parking(ajout d'un EventAlerte) et changement des codes en conséquence. J'ai assisté au point dev* du lundi matin à 11h15.

*point dev = chaque membre de l'équipe développement parking explique ce qu'il a fait la semaine précédente et dit ce qui lui reste à faire pour les semaines suivantes.



```
Terminal Local x + v
└─
[INFO] -----
[INFO] Reactor Summary for IVPARK-LIBS-connector-hik 0.0.1-SNAPSHOT:
[INFO]
[INFO] IVPARK-LIBS-connector-hik ..... SUCCESS [ 4.059 s]
[INFO] IVPARK-LIBS-connector-hik-core ..... SUCCESS [ 8.208 s]
[INFO] IVPARK-LIBS-connector-hik-simulator ..... SUCCESS [ 12.607 s]
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 25.485 s
[INFO] Finished at: 2024-06-12T14:25:20+02:00
[INFO] -----
```

Mardi 11 juin

Aujourd'hui j'ai travaillé sur le simulateur en essayant de créer une nouvelle alerte pour la détection de mouvement cette fois-ci, n'ayant pas réussi, j'ai arrêté le test pour me concentrer sur autre chose.

Durant toute l'après-midi, un membre de l'équipe parking Guillaume, est venu m'aider pour tenter de lancer le simulateur sur un serveur(jusqu'à présent je le lancer localement).

Nous avons dû installer premièrement [Docker](#)³ puis un logiciel [Postman](#)⁴ sauf que n'ayant pas les droits d'accès Guillaume a dû faire une demande au ST(Services Techniques) afin de me donner une authentification et des droits.

En attendant il m'a donné des explications sur le fonctionnement des applications et services depuis son pc, j'ai vu une démonstration d'un de ses projets actuels et d'autres projets qui nécessitent l'utilisation de Docker afin que je comprenne mieux son fonctionnement.

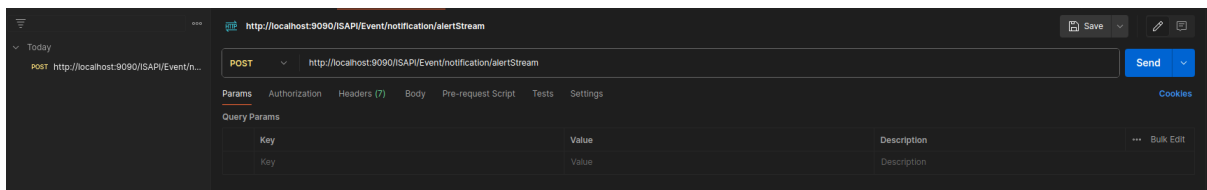
³Docker = Le logiciel Docker est une technologie de conteneurisation qui permet la création et l'utilisation de [conteneurs Linux](#)®.



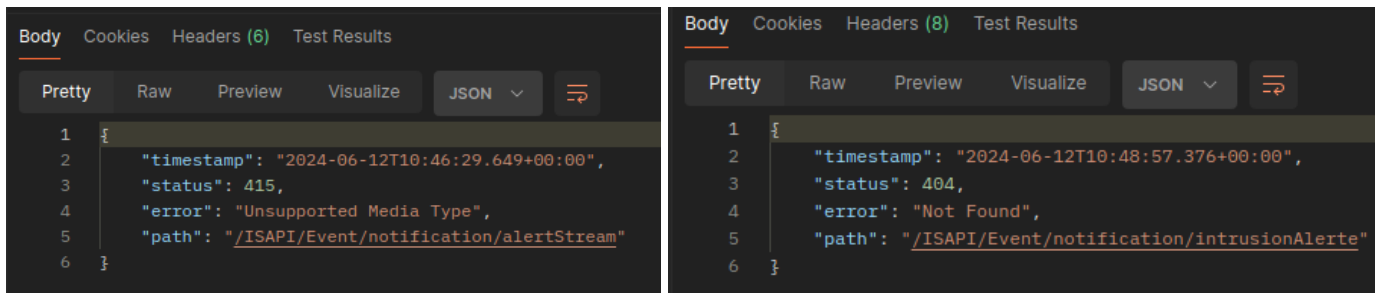
⁴Postman = Postman est un outil qui permet aux développeurs de tester et de documenter les API (Application Programming Interface).



Requête postman



Résultat requête postman (non fonctionnel)



Mercredi 12 juin

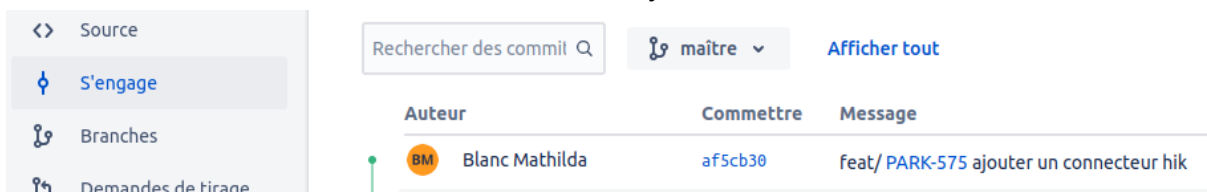
Aujourd'hui nous avons fait un [Pull Request\(PR\)](#)⁵ [d'intellij](#)⁶ vers [bitbucket](#)⁷ avec Guillaume. Le but étant que je propose mes modifications de code dans le dépôt bitbucket, où j'ajoute dans bitbucket des membres de dev parking en tant "qu'évaluateur" de mon dépôt, afin qu'ils puissent voir mes modifications et qu'ils acceptent ou non de changer avec ma version contre la leur.

Ensuite j'ai fait une documentation "utilisation" pour le simulateur en mettant quelques grandes étapes, et en expliquant comment se servir de certaines requêtes (GET ou POST).

Dans l'après-midi, suite à une demande de Guillaume qui a analysé mon code déposé dans bitbucket, j'ai commencé à faire une classe de test qui teste l'appel pour chaque [endpoint](#)⁸ grâce à une liste d'endpoints que j'ai faite.

- + Réunion présentation parking de 15h à 16h45 avec présentation des logiciels et interfaces (IVPARK), gestions des clients et autres.

⁵**pull request** consiste tout simplement à demander à l'auteur original de [merge](#) nos modifications. Ma branche ajoutée dans bitbucket :



⁷**bitbucket**: Bitbucket Cloud est un outil d'hébergement du code et de collaboration basé sur Git, qui a été conçu pour les équipes en entreprise.



⁸**Endpoint** est ce qu'on appelle une extrémité d'un canal de communication. Autrement dit, lorsqu'une API interagit avec un autre système, les points de contact de cette communication sont considérés comme des Endpoints.

Jeudi 13 juin

Durant ma matinée, j'ai donc créé des tests pour tester chaque endpoints individuellement dans le code, j'ai assisté à une réunion "présentation R&D transverse" de 10h à 11h.

L'après - midi en attendant mes droits d'accès sur [Docker](#)⁹, [Jenkins](#)¹⁰ et [Nexus](#)² j'ai donc suivi un tutoriel afin de réaliser la création d'une application [Spring](#)¹¹, puis de son test avec JUnit.

⁹**Docker** est une plateforme permettant de lancer certaines applications dans des conteneurs logiciels lancés en 2013.

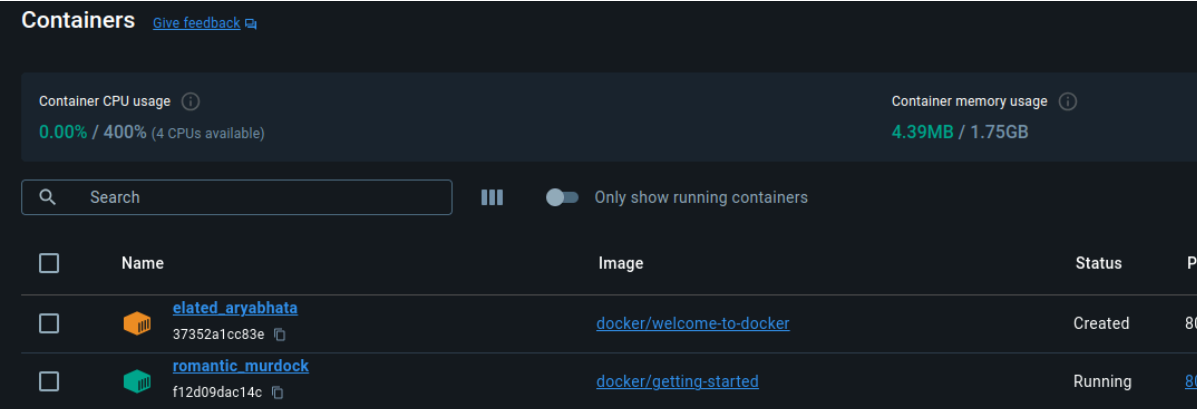
²**Nexus** est une plateforme de gestion de dépôts, permettant d'héberger des artefacts. Ces artefacts sont des composants, générés par exemple au build d'un projet, et déposés ensuite sur Nexus grâce à l'outil Maven.

¹⁰**Jenkins** est un outil open source de serveur d'automatisation. Il aide à automatiser les parties du développement logiciel liées au build, aux tests et au déploiement, et facilite l'intégration continue et la livraison continue.

¹¹**Spring Boot** est un framework Java open source utilisé pour programmer des applications Spring autonomes de qualité production avec un minimum d'effort.

Vendredi 14 juin

Aujourd'hui j'ai utilisé l'application Docker, en testant ses différentes possibilités (en dehors des serveurs Netcelar ne pouvant toujours pas y avoir accès).



The screenshot shows the Docker Desktop interface. At the top, it displays 'Containers' with a 'Give feedback' link. Below this, there are two status bars: 'Container CPU usage' at 0.00% / 400% (4 CPUs available) and 'Container memory usage' at 4.39MB / 1.75GB. A search bar and a toggle for 'Only show running containers' are present. The main part of the interface is a table listing containers.

| | Name | Image | Status | Port |
|--------------------------|--|--|---------|------|
| <input type="checkbox"/> | elated_aryabhata 37352a1cc83e | docker/welcome-to-docker | Created | 8080 |
| <input type="checkbox"/> | romantic_murdock f12d09dac14c | docker/getting-started | Running | 8080 |

J'ai ensuite travaillé sur le simulateur et j'ai corrigé et déboguer le code et ses tests avec pas mal de problèmes.

✗ Stopped. Tests failed: 3, passed: 3 of 8 tests – 190 ms

✖ Stopped. Tests passed: 6 of 8 tests – 179 ms

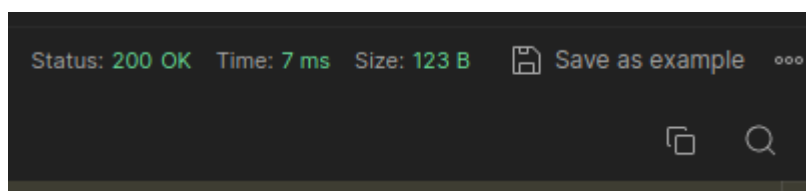
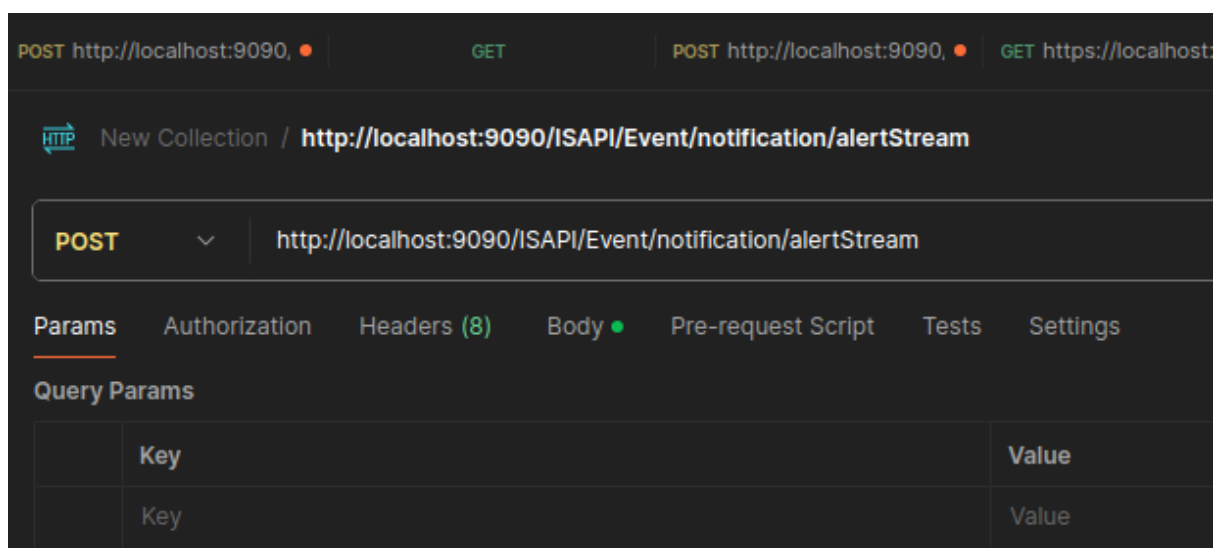
J'ai eu du mal à corriger un test, je l'ai donc mis de côté temporairement.

✔ Tests passed: 7 of 7 tests – 164 ms

Semaine 4

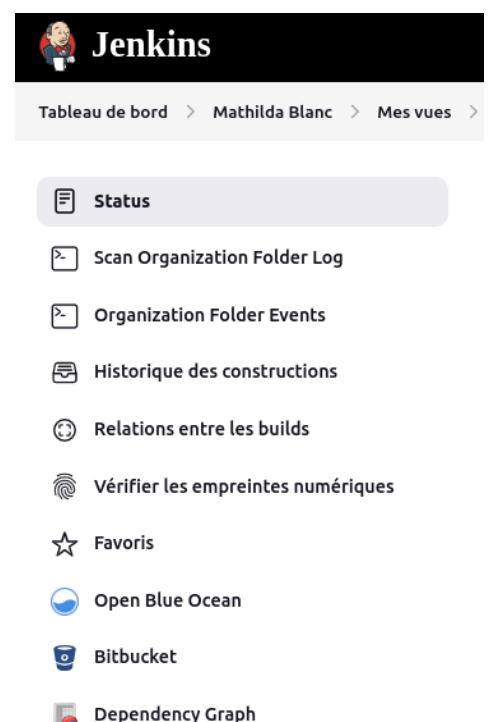
Lundi 17 juin

Aujourd'hui, comme chaque lundi j'ai participé au "[point dev](#)" de 11h15 à 12h15. J'ai pu avancer et corriger mes requêtes GET et POST afin qu'elles retournent toutes un status 200 OK.



Durant l'après-midi, j'ai enfin eu accès au serveur sur [jenkins](#)¹⁰. Apparence de jenkins :

Jenkins étant connecté en partie avec [bitbucket](#)⁷, nous y retrouvons quelque point commun, comme les branches [PR](#)⁵. Grâce à cela j'ai accès à plusieurs repositories (= lieu de stockage de paquets logiciels) de l'entreprise.



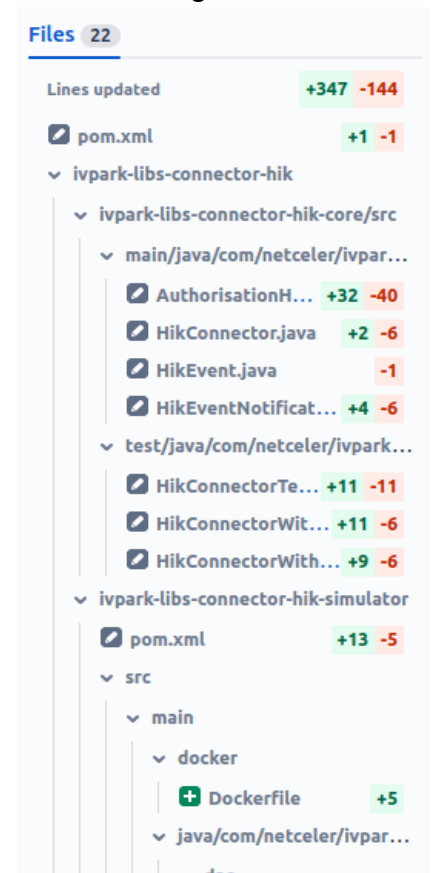
Mardi 18 juin

Pour entamer la journée, j'ai travaillé sur le simulateur et je l'ai mis à jour sur [Bitbucket](#)⁷, pour que les membres de l'équipe puissent voir tous mes changements. Cela se fait sous cette forme :

Il existe de nombreux autres détails disponibles dans Bitbucket : document modifié, ligne modifiée, mot effacé ou ajouté, etc.

Durant l'après-midi j'ai regardé des tutoriels de programmations en prenant des notes.

À l'heure actuelle, Julien souhaite que nous fassions fonctionner le simulateur sur un serveur, mais seulement les personnes nécessaires pour se connecter au serveur, ne sont pas disponibles pour le moment.



Mercredi 19 juin

La journée à commencé avec un "point de stage" accompagné de Julien et Guillaume, ce point était nécessaire car Julien étant de plus en plus occupé, il a demandé à Guillaume de prendre un peu le relais, et si j'ai des questions je peux me tourner soit vers Guillaume, soit vers Julien si il est disponible. Durant ce point nous avons vu avec Guillaume, ce que j'allais pouvoir faire les dernières semaines, ce que je préférais faire et donc ce que je peux faire en conséquence.

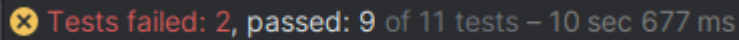
Pour le reste de la matinée j'ai corrigé le simulateur et changé des trucs, suite à la demande de Guillaume qui avait regardé mon code la veille en me laissant des commentaires, sur [bitbucket](#)⁷.

Dans le déroulement de l'après-midi Guillaume et moi avons connecté mon ordi à [docker](#)³ et donc nexus, puis nous avons parlé du code du simulateur. Il m'a bien expliqué comment fonctionne L'application [Spring boot](#)¹¹, et à quoi elle servait dans notre simulateur. Nous avons également créé un nouveau test, mais pas terminé pour que je travaille dessus et que je le finisse.

Jeudi 20 juin

La journée à commencé avec un petit point journée de quelques minutes avec Guillaume et Julien, où je leur explique ce que j'ai fait hier et ce que je vais faire aujourd'hui, afin qu'ils suivent mon travail.

Ensuite j'ai continué à travailler sur le nouveau programme test ainsi que sur un nouveau test "incomplet" fourni par Guillaume.



✖ Tests failed: 2, passed: 9 of 11 tests – 10 sec 677 ms

Beaucoup de difficultés sur un test de température qui m'a pris tout l'après-midi, au début je me suis servie d'une doc donné par Guillaume mais je n'y suis finalement pas arrivé, je demanderai de l'aide à Guillaume demain.

Vendredi 21 juin

Aujourd'hui j'ai donc continué à debug le nouveau test, Guillaume se trouvant en télétravail toute la journée, nous avons fait notre petit point de début de journée en visio, celle-ci à pris presque 2 heures au lieu de quelques minutes, car nous avons essayé de debug le code ensemble grâce à un mode où nous sommes 2 sur intellij sur le même code au même moment.

Nous n'avons finalement pas totalement réussi avant que Guillaume doive partir. J'ai donc continué dessus tout l'après midi.

De plus, en essayant de corriger un test, 2 autres ont cessé de fonctionner.

Fin d'après-midi j'ai finalement corrigé les 2 tests, et j'ai finalement trouvé le problème dans le test du début, mais je n'ai pas encore réussi à corriger le problème.

Semaine 5

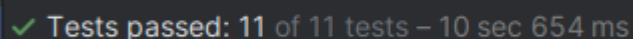
Lundi 24 juin

Pour ce début de semaine, j'ai commencé par faire un petit point avec Guillaume de mon vendredi, et je lui ai dit qu'aujourd'hui j'allais essayer de réparer l'erreur du test, il m'a dit que si j'avais besoin je pouvais lui demander.

J'ai donc travaillé sur le test en essayant de corriger l'erreur. Vers 11h j'ai eu une visio d'une présentation CSE (=Comité Sociale et Économique) de l'entreprise, je n'ai donc pas assisté au point dev du lundi.

Ensuite j'ai regardé quelques tutoriels et autres afin de m'aider à résoudre le problème, j'ai débuggé le code plusieurs fois.

Et finalement durant tout l'après-midi nous avons codé avec Guillaume afin de corriger le test ensemble, puis il m'a expliqué pleins de points que je comprenais pas grâce à des schémas sur un tableaux, etc.



✔ Tests passed: 11 of 11 tests – 10 sec 654 ms

Mardi 25 juin

Aujourd'hui Guillaume étant plus occupé, il m'a donné des trucs que je pouvais faire, de plus j'ai réalisé quelques tutoriels, ainsi qu'un autre test du code que Guillaume m'avait demandé.

Je me suis créé une fiche avec des récapitulatifs des tutoriels afin de pouvoir m'en servir directement depuis chez moi.

Mercredi 26 juin

En ce milieu de semaine, à la demande de Guillaume, j'ai travaillé sur le simulateur en ajoutant des classes en rapport avec les tests, le but étant de simplifier le code grâce à des beans, avec une classe Repository, une classe Service qui seront reliées au Controller.

J'ai fait tout un tas de changements pour simplifier le code en changeant plusieurs méthodes puis j'ai ensuite push mon code sur ma branche de bitbucket.

Jeudi 27 juin

Aujourd'hui j'ai continué à travailler sur le simulateur, en créant un test qui consiste à changer le type d'un événement (EVENT) toutes les 2 secondes en rapport avec les données insérées.

Le changement des 2 secondes est effectué grâce à une annotation spéciale de spring boot :

```
@Scheduled(fixedRate = 2000) new *
public void changeEventTypePeriodically() {
    String newType = eventTypes[currentIndex];
    hikRepositoryAlertStream.changeType(newType);
    currentIndex = (currentIndex + 1) % eventTypes.length;
}
```

Puis j'ai également créé une classe "Enum" afin d'y stocker les différents types d'événement.

```
public enum HikE
    TMA, 1 usage
    TMPA, no usag
    videoloss, r
    ipconflict,
    illaccess, r
```

Vendredi 28 juin

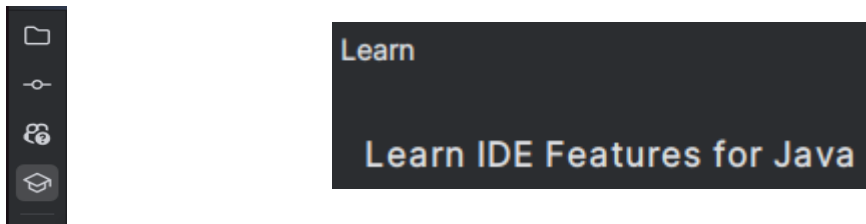
Pour finir la semaine j'ai conclu mes tests qui sont tous fonctionnels.

Durant l'après-midi j'ai pu regarder quelques tutoriels et j'ai pu avancer légèrement mon portfolio en fin de journée.

Semaine 6

Lundi 01 juillet

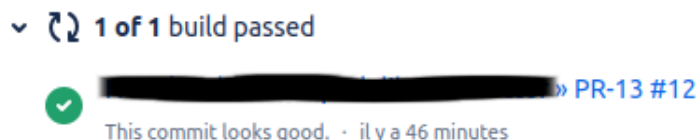
Aujourd'hui Guillaume n'étant pas trop disponible, j'ai donc profité pour avancer un peu mon portfolio, de plus j'ai regardé des tutoriels et utilisé une partie spéciale de IntelliJ grâce à un plugin qui sert d'entraînement que Guillaume m'a fait découvrir afin de progresser.



Je continuerais à travailler sur le simulateur dès demain avec Guillaume qui doit me montrer comment dockeriser mon simulateur.

Mardi 02 juillet

Aujourd'hui avec Guillaume nous avons conteneurisé le simulateur sur docker donc c'est-à-dire que nous avons commencé par push un commit sur bitbucket, puis nous avons attendu que le build passe.



Ensuite j'ai attendu que Guillaume approuve mon commits.

Puis j'ai pu le merge sur JetBrains donc le code à jour est désormais disponible depuis le cloud de l'entreprise afin qu'il puisse s'en servir le moment voulu.

[Overview](#) [Files changed 37](#) [Commits 1](#)



Enfin grâce à une commande dans mon terminal "docker pull ___ ___ ___" ce qui à créer dans mon appli docker une image et son container que l'on peut run directement dans docker.

Durant l'après-midi j'ai dû me renseigner sur ce qu'était un **docker-compose** afin de pouvoir mettre en place le simulateur dans docker-compose les prochains jours.

Mercredi 03 juillet

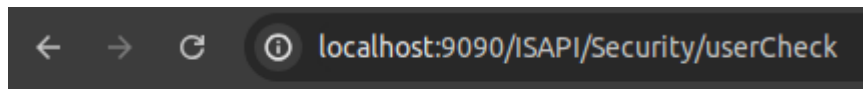
Aujourd'hui, j'ai appris à me servir de docker-compose et de podman-compose en suivant des docs d'utilisation comme :

<https://docs.oracle.com/fr/learn/podman-compose/#confirm-podman-compose-is-working> ou <https://docs.docker.com/compose/gettingstarted/>

Le but est de s'en servir pour mettre le simulateur dans le docker.

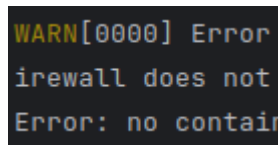
J'ai donc installé les trucs nécessaires par commande sur mon terminal, puis je m'en suis servi.

Résultat de l'appel d'un test grâce à docker-compose



User check completed successfully

Finalement je n'ai pas réussi à faire avec podman-compose, nous avons donc utilisé seulement docker-compose.



Pour terminer sur ce simulateur nous allons le faire tourner sur une VM.

Jeudi 04 juillet

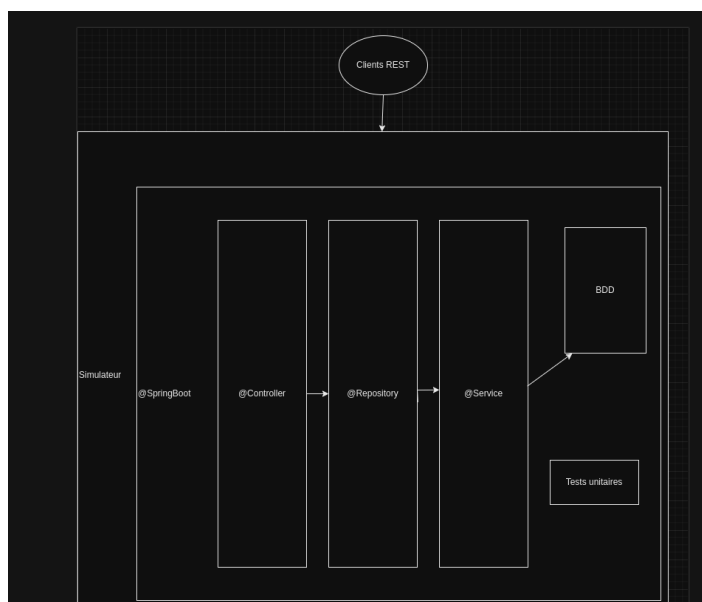
Pour terminer les 2 jours de stage, avec Guillaume on a persisté un EventNotificationAlert dans une base de données. Pour nous aider nous avons utilisé un tuto : <https://www.baeldung.com/spring-boot-h2-database> → **Spring Boot avec base de données H2.**

Guillaume a passé beaucoup de temps pour m'expliquer correctement le fonctionnement de spring boot avec la base de données car c'était le dernier jour étant donné qu'il est en télétravail le vendredi. J'ai également fait un point de fin de stage avec mon maître de stage qui est aussi en télétravail ce vendredi.

Je lui ai fait un retour du stage, lui également pour bien finir ce stage.

Vendredi 05 juillet

Pour la dernière journée j'ai rempli une documentation "utilisation du simulateur", avec plusieurs éléments demandé par Guillaume comme une liste d'endpoint, un schéma Architectural :



ou encore une doc utilisation pour le développement des test du simulateur, avec une petite conclusion de stage.

Pour finir la journée j'ai également mis mon portfolio en ligne.

Conclusion de stage

J'ai vraiment appréciée faire ce stage. Il m'a permis de me familiariser avec le travail et la vie en entreprise pendant 6 semaines.

Au cours de ce stage j'ai tout d'abord eu de vagues informations sur le travail à fournir et à réaliser en tant que stagiaire, car mon maître de stage était très occupé. Ensuite, au fil des semaines, j'ai reçu davantage d'informations et d'objectifs à accomplir avec une autre personne de l'entreprise qui m'a prise en charge.

Mes objectifs étaient :

- de mettre à jour un ancien simulateur d'une caméra thermique de sorte qu'il renvoie différentes alertes selon la situation(alerte incendie, alerte au flux de données...) ou encore que la caméra envoie des signes de vie toutes les X secondes afin de vérifier que tout soit en règle.
- de parvenir à faire fonctionner le simulateur sur une machine virtuelle (VM).

Pour mener à bien ce projet, j'ai consacré des semaines à effectuer des tests unitaires en interne afin d'explorer les différentes fonctionnalités en fonction des objectifs. J'ai visionné beaucoup de tutoriels et passé beaucoup d'heures avec une personne de l'entreprise qui m'a tout expliqué en utilisant des schémas, pour que je comprenne mieux.

Grâce à ce stage, j'ai pu améliorer mes compétences, en particulier en JAVA et en framework Java. J'ai eu l'opportunité de découvrir de nombreux logiciels et applications utilisés dans l'entreprise (comme : **docker, podman, postman, jira, bitbucket, spring boot** et bien d'autres...), j'ai appris à les manipuler et à les utiliser efficacement. De plus, j'ai découvert de nombreuses fonctionnalités sur IntelliJ en l'utilisant durant toutes ces semaines.

Définitions

¹ **JIRA** = Jira est un système de suivi de bugs, de gestion des incidents et de gestion de projets développé par Atlassian



² **Nexus** = Nexus est une plateforme de gestion de dépôts, permettant d'héberger des artefacts. Ces artefacts sont des composants, générés par exemple au build d'un projet, et déposés ensuite sur Nexus grâce à l'outil Maven.

³ **Docker** = Le logiciel Docker est une technologie de conteneurisation qui permet la création et l'utilisation de [conteneurs Linux®](#).



⁴ **Postman** = Postman est un outil qui permet aux développeurs de tester et de documenter les API (Application Programming Interface).



⁵ **pull request = (PR)** consiste tout simplement à demander à l'auteur original de [merge](#) (combiner) nos modifications.

⁶ **intellij** = IntelliJ IDEA également appelé « IntelliJ », « IDEA » ou « IDJ » est un environnement de développement destiné au développement de logiciels informatiques reposant sur la technologie Java.



⁷ **bitbucket** = Bitbucket Cloud est un outil d'hébergement du code et de collaboration basé sur Git, qui a été conçu pour les équipes en entreprise.



⁸ **Endpoint** est ce qu'on appelle une extrémité d'un canal de communication. Autrement dit, lorsqu'une API interagit avec un autre système, les points de contact de cette communication sont considérés comme des Endpoints.

⁹ **Docker** est une plateforme permettant de lancer certaines applications dans des conteneurs logiciels lancés en 2013.

¹⁰ **Jenkins** est un outil open source de serveur d'automatisation. Il aide à automatiser les parties du développement logiciel liées au build, aux tests et au déploiement, et facilite l'intégration continue et la livraison continue.

¹¹ **Spring Boot** est un framework Java open source utilisé pour programmer des applications Spring autonomes de qualité production avec un minimum d'effort.