

Compte Rendu Hackathon Equipe 3:

<https://github.com/placaes/ctp-journal>

https://github.com/GastonDeseinelG2I/LA2-HACKATHON-POO_BDD

I] Présentation de l'équipe	1
II] Introduction	2
III] Mise en place d'outils de gestion de projet	3
IV] Notre choix fonctionnel	5
V] Nos choix techniques	7
VI] Installation de la solution	8
VII] Conclusion - Retour sur expérience	9

I] Présentation de l'équipe

L'équipe 3 est composé de :

- Baptiste Pautrat : Réalisation de la partie "front", ainsi que de la mise en place d'outils permettant la gestion de ce projet.
- Paul Lacaes: Réalisation de la partie "back" ainsi que de l'API REST.
- Valentin Poitrot: Réalisation de l'UML, de la Base de données ainsi que réalisation de la partie "back".
- Gaston Deseine: Réalisation de la partie "front" et mise en relation du "Back" et du "Front"
- Hadrien Belcour: Réalisation de la partie "front", réalisation de la présentation (diaporama).
- Loïc Coquerelle: Réalisation de la partie "back" et de la documentation ci présente.



Compte Rendu Hackathon Equipe 3:

<https://github.com/placaes/ctp-journal>

https://github.com/GastonDeseinelG2I/LA2-HACKATHON-POO_BDD

II] Introduction

Durant la journée du 23 mars 2021, nous devons réaliser, par équipe, lors d'un hackathon, une solution respectant un cahier des charges qui nous est fourni au début de cette journée. Le but de cet hackathon, en plus de la solution fournie, est de réfléchir aux divers choix techniques, à l'architecture mais également à l'organisation de l'équipe, qui sera tirée au sort le matin même. Toutes ces contraintes seront présentées à la fin de la journée lors d'une présentation de la solution devant les autres équipes de manière synthétique, de plus elle sera chaque choix sera détaillé dans cette présente documentation.

Le sujet de ce hackathon consiste à la réalisation d'un "journal personnel", ce "journal" permettra de saisir des indicateurs (ou des informations) sur des sujets qui seront paramétrables par l'utilisateur, et d'intégrer par la suite des informations issues d'objets connectés (frigo, montre...). Cette deuxième partie ne sera pas réalisée aujourd'hui, cependant la solution mise en place doit pouvoir prévoir la possibilité d'inclure cette dernière.

Cette solution se doit d'être, comme son nom l'indique, un "journal personnel", ainsi l'utilisateur doit avoir la possibilité de paramétrer l'application en fonction de son usage, et donc de la personnaliser comme bon lui semble. Il pourra donc ajouter un événement à sa journée selon un type d'activité, ces types seront ajoutés par l'utilisateur en question et seront donc personnels. L'utilisateur peut ensuite comme dit précédemment ajouter des types d'activités, il pourra également consulter l'historique de toutes ses activités.

A titre d'exemple, ce journal peut être utilisé par :

- une personne qui souhaite avoir une trace de toutes ses lectures..
- une personne souhaitant réduire son empreinte carbone et qui notera ses déplacements de la journée..
- une personne souhaitant intégrer les données de sa montre connectée
- une personne souhaitant intégrer les données collectées par son frigo connecté

Ces exemples, comme leurs noms l'indiquent ne sont que des exemples, l'utilisation de cette solution étant le plus personnalisable possible.

Compte Rendu Hackathon Equipe 3:

<https://github.com/placaes/ctp-journal>

https://github.com/GastonDeseinelG2I/LA2-HACKATHON-POO_BDD

III] Mise en place d'outils de gestion de projet

Naturellement, au sein de l'équipe, Baptiste a pris en main l'installation d'outils qui nous permettrait d'optimiser cette journée. Nous avons donc réalisé dans un premier temps une matrice de compétences, ainsi que d'intéressement dans les divers domaines que nous devons traiter durant cette journée, afin d'optimiser les compétences de chacun.

Il en a résulté la matrice suivante:

	Baptiste		Paul		Loïc		Valentin		Gaston		Hadrien	
JAVA	1		3		2		1		2		1	
Angular	2		1		1		2		3		1	
SQL	3		3		2		3		2		2	
Diapo	3		1		3		3		3		3	
Doc	2		2		3		3		2		3	
UML	1		3		3		3		3		3	

Notes de 1 à 3

→ 1 = pas à

l'aise

→ 2 = à l'aise

→ 3 = très à

l'aise



Très intéressé



Intéressé



Pas intéressé

Par exemple nous avons donc décidé que Paul serait donc responsable de la partie "Backend" qui sera détaillée dans la partie "Architecture Technique". Gaston prend la responsabilité de la partie "Frontend" au vu des différentes capacités de chacun, et au vu de l'intéressement de l'implémentation des diverses parties.

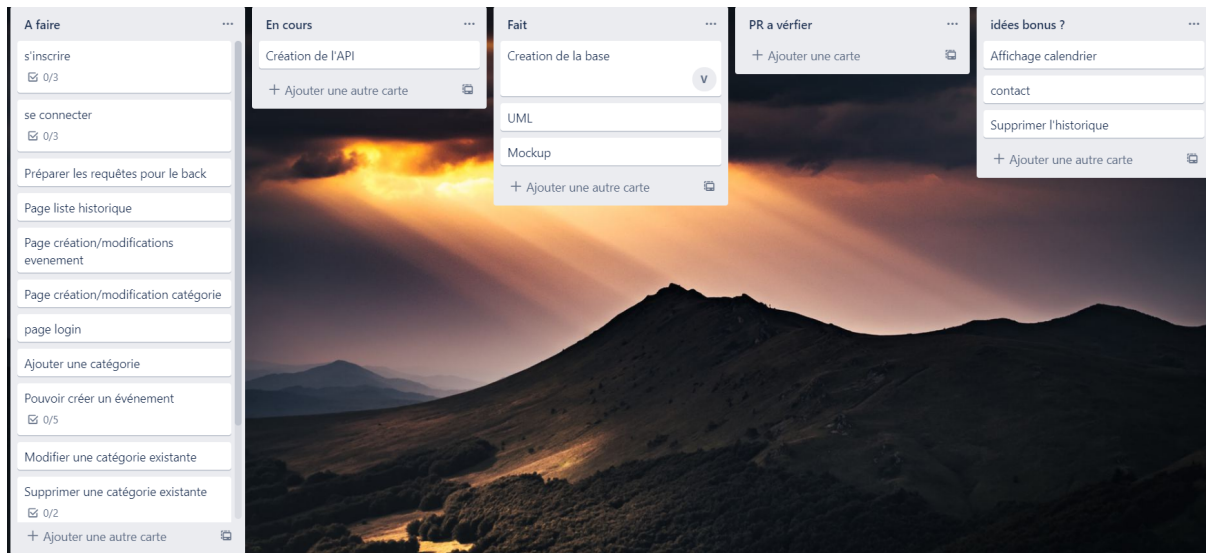
Compte Rendu Hackathon Equipe 3:

<https://github.com/placaes/ctp-journal>

https://github.com/GastonDeseinelG2I/LA2-HACKATHON-POO_BDD

Nous avons ensuite réfléchi aux différentes vues de manière fonctionnelle, afin d'en extraire les diverses informations nécessaires pour la création de la base de données et du modèle relationnel.

Par la suite, nous avons listé les diverses tâches que nous devons réaliser durant cette journée en fonction des différentes “parties”. Nous avons mis en place un trello afin de suivre l'état des tâches et l'avancement de ces dernières.



Les différentes vues fonctionnelles sur lesquelles nous avons réfléchi étaient cependant encore “au brouillon” sur le tableau. Nous avons donc réalisé de rapides mockups afin d’avoir ces vues de manière plus propres et sur lesquelles se rattacher plus facilement. Ils sont disponibles suivant ce lien : <https://balsamiq.cloud/slyj28q/pnk8fdm>

Compte Rendu Hackathon Equipe 3:

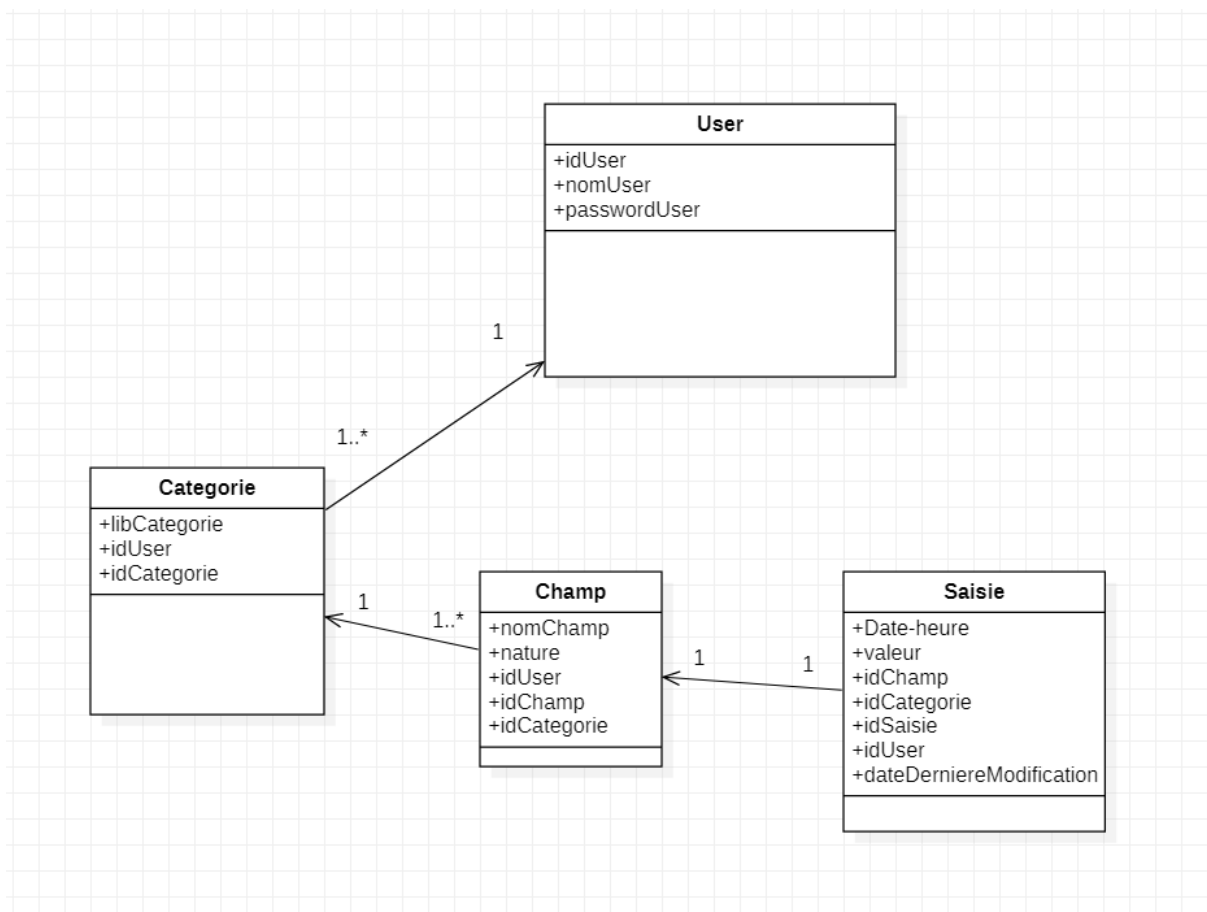
<https://github.com/placaes/ctp-journal>

https://github.com/GastonDeseinelG2I/LA2-HACKATHON-POO_BDD

IV] Notre choix fonctionnel

Afin de respecter le cahier des charges et la perspective de “personnaliser” au maximum le journal, nous avons décidé que chaque utilisateur aurait ses propres “catégories” et il pourrait en ajouter selon sa guise. Il pourra également les modifier après création.

Nous avons donc réalisé le modèle relationnel (diagramme UML) suivant:



Ce diagramme se compose de 4 entités:

- Saisie: qui représente la saisie des différents “champs”
- Champ: qui représente le champ
- Catégorie: représentant le type personnalisable par l'utilisateur

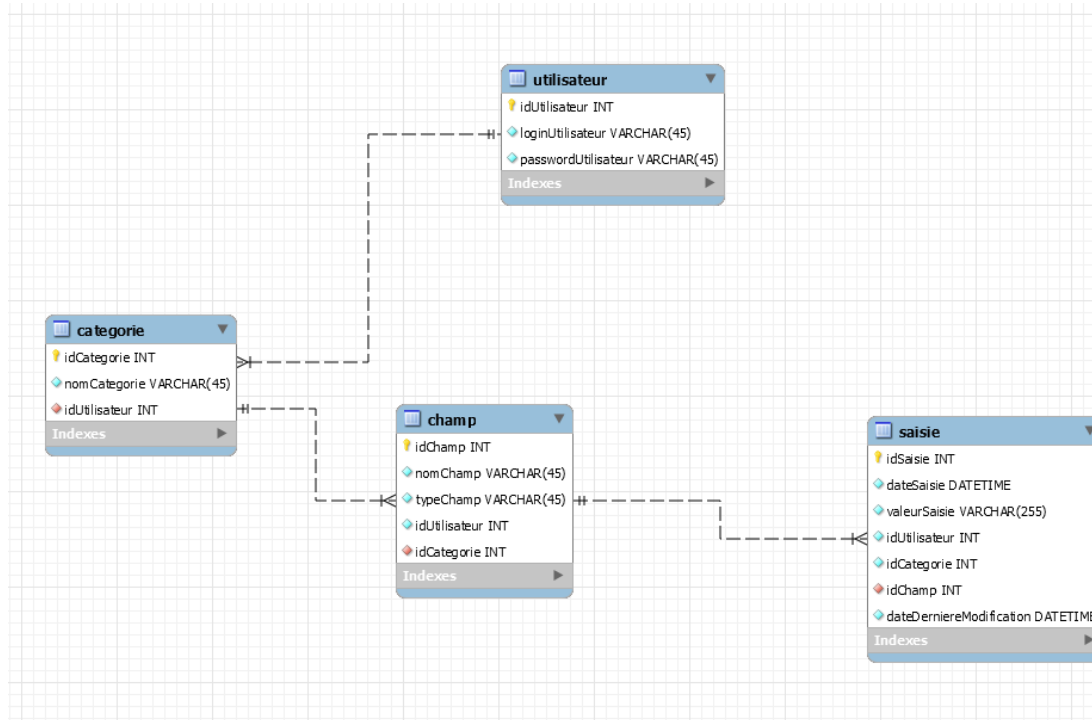
Compte Rendu Hackathon Equipe 3:

<https://github.com/placaes/ctp-journal>

https://github.com/GastonDeseinelG2I/LA2-HACKATHON-POO_BDD

- User: représentant les utilisateurs

De ce diagramme découle le schéma de notre base de données qui sera le suivant:



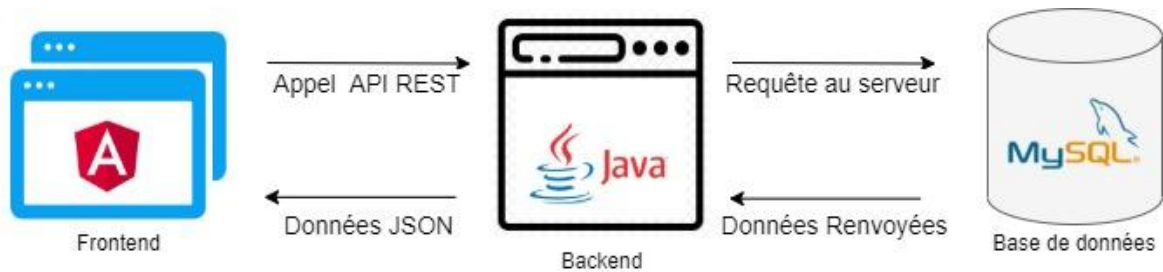
Compte Rendu Hackathon Equipe 3:

<https://github.com/placaes/ctp-journal>

https://github.com/GastonDeseinelG2I/LA2-HACKATHON-POO_BDD

V] Nos choix techniques

Pour réaliser ce projet nous étions totalement libres dans les divers choix techniques. Nous devons cependant réaliser un modèle DAO pour la partie Back. Nous nous sommes donc orientés vers le JAVA pour cette partie, puisque nous venions de suivre l'enseignement POO sur cette partie. Pour la partie Front nous étions toujours libres sur le choix, nous nous sommes orientés vers Angular, puisque nous venions d'avoir un enseignement sur cette technologie quelques semaines auparavant. Ainsi notre projet s'organisera de la façon suivante.



Compte Rendu Hackathon Equipe 3:

<https://github.com/placaes/ctp-journal>

https://github.com/GastonDeseinelG2I/LA2-HACKATHON-POO_BDD

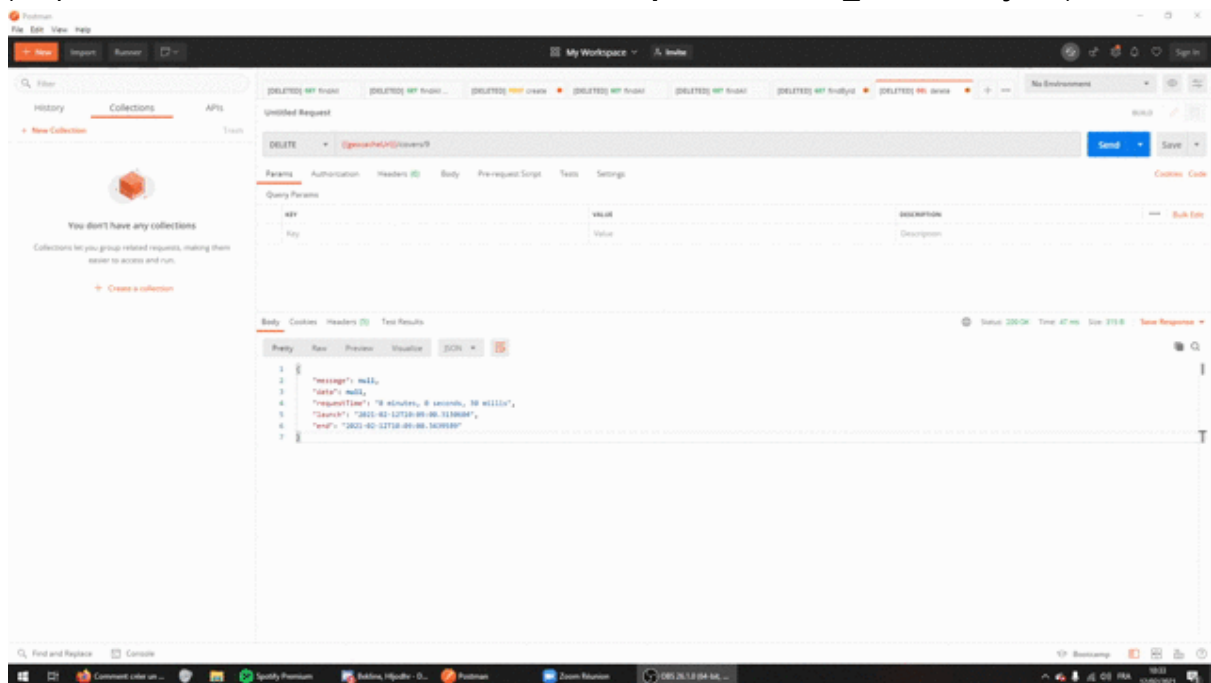
VI] Installation de la solution

A] Installation de la partie Back:

Postman

Pour l'installation de la solution, il faut dans un premier temps importer la collection Postman dans Postman. Pour ce faire, il suffit de cliquer sur le "IMPORT", puis cliquez sur la nouvelle fenêtre et choisissez le fichier JSON. La collection est maintenant importée.

(Disponible ici : [/src/main/resources/hackathon.postman-final_collection.json](#))



Cette collection va permettre de faire des appels à nos différents services et de récupérer les informations voulues. Dedans, on pourra remarquer deux dossiers un mongodb et un mysql pour chaque mode de fonctionnement.

Mysql

Il faut ensuite importer la base de données dont le script est disponible dans **[/src/main/resources/bdd/hackathon_v2.sql](#)**, nous avons utilisé l'utilisateur root et le mot de passe root. Si la bdd de données possède un nom d'utilisateur et/ou mot de passe particulier, il est possible de les modifier dans le fichier [/src/main/resources/application.properties](#) pour mysql

Compte Rendu Hackathon Equipe 3:

<https://github.com/placaes/ctp-journal>

https://github.com/GastonDeseinelG2I/LA2-HACKATHON-POO_BDD

Lancement du projet via IntelliJ

Pour l'installation de la solution, il faut simplement faire à la racine du projet : mvn clean install Puis il suffit de lancer la configuration CtpJournalApplication

Maintenant, il est possible de tester l'application via Postman

Lancement via terminal

java -jar target/*.jar

B] Installation de la partie Front

Il est développé en Angular et communique avec le back-end via des requêtes HTTP à l'API en Java.

Pour installer les dépendances:

npm install

Pour démarrer le serveur de développement

npm start -- -o

On retrouve ensuite le projet à l'adresse **http://localhost:4200**

Pour que le front puisse communiquer avec le back, il faut que le projet Java soit démarré au préalable.

VII] Conclusion - Retour sur expérience

Afin de conclure sur ce hackathon nous avons décidé de faire un bref retour sur expérience de cette journée. Dans un premier temps nous sommes tous d'accord pour dire que le format de la journée était bon, cependant les délais étaient peut être trop bref pour le travail demandé. De plus les groupes étant choisis aléatoirement, le déséquilibre de compétences au sein d'un même groupe pouvait se faire ressentir facilement (Cf: matrice de compétences). Nous avons toutefois fortement apprécié participer à ce hackathon, malgré une solution qui ne fut malheureusement pas choisie comme grande gagnante de ce mini concours.