

Dédicaces

Je dédie ce modeste travail à . . .

*À la mémoire de mon père **Ali** l'être le plus cher au monde qui nous a quittés pendant la réalisation de ce travail et qui n'a jamais épargné d'effort pour me pousser à l'avant dans la vie, en témoignage de mon respect et de mon amour. Aucune dédicace ne saurait exprimer l'amour, l'estime, le dévouement que j'ai toujours eu pour toi, tu me manques,*

*À ma très chère mère **Saliha**, qui a sacrifié ses plus belles années pour embellir les miennes, aucun mot ne serait suffisant pour témoigner l'étendue des sentiments que j'éprouve à ton égard. Ce travail est le fruit de tes sacrifices que tu as consentis pour mon éducation et ma formation,*

*À ma très chère femme **Hejer**, mon trésor de bonté, de générosité et de tendresse qui n'a jamais cessé de me soutenir, m'assister et m'encourager, c'est par ton initiative que je suis là,*

*À mes adorables fils **Sami**, **Heni** et **Hakim** ainsi qu'à ma fille charmante **Linda** qui sont ma raison de vivre,*

*À mes chères sœurs **Hela** et **Sameh**, que j'aime plus que tout au monde,*

*À mes gendres **Rafik** et **Nizar** que je considère frères,*

*À mes amis les plus chers, **Ferid** et **Mourad** mes supporteurs et vrais amis, qui ne m'ont jamais laissé tomber.*

Remerciements

Je tiens tout d'abord à remercier Dieu tout puissant, qui m'a donné la force et la patience d'accomplir ce modeste travail.

Je remercie mon encadrant Monsieur Adel KHALFALLAH pour la bonne qualité de son encadrement, son aide précieuse et ses conseils.

Je voulais m'adresser aussi à mes collègues du lycée Farhat Hached pour leur disponibilité et les suggestions qu'ils m'ont faites et spécialement Madame Mounira JLASSI pour son soutien, son aide et son assistance tout au long de l'élaboration de ce travail.

Un grand remerciement pour mon cher ami Ferid HELALI pour son aide et son soutien moral qui m'a été très bénéfique pendant les moments difficiles de cette période.

Je remercie tous ceux qui ont contribué de près ou de loin à entamer ce travail et auxquels je dois être reconnaissant.

Table des matières

Introduction générale	1
Chapitre 1 Contexte du projet	3
Introduction	3
1-1 Organisme d'accueil	3
1-2 L'évaluation continue	4
1-3 Etude de l'existant	5
1-4 Critiques de l'existant	5
1-5 Proposition de solutions.....	6
Conclusion	6
Chapitre 2 Méthodologie et démarche	7
Introduction	7
2-1 Qu'est-ce que le développement agile	7
2-2 Qu'est-ce que (CI / CD) ?	7
2-3 Culture DevOps	8
2-4 Quelle relation entre eux ?	8
2-5 Adoption	9
Conclusion	11
Chapitre 3 Réalisation.....	12
Introduction	12
3-1 Planification du projet.....	12
3-2 Choix de l'environnement de développement et des outils	18
3-3 Diagramme de cas d'utilisation global	22
3-4 Raffinement du cas d'utilisation « Gérer les Quiz ».....	24
3-5 Raffinement du cas d'utilisation « Gérer les sessions »	29
3-6 Diagramme de classes.....	33
3-7 Les Sprints	34
Conclusion	43

Chapitre 4 Déploiement	44
Introduction	44
4-1 DevOps	44
4-2 IBM Cloud	46
4-3 GitHub	52
Conclusion	53
Chapitre 5 L'application	54
Introduction	54
5-1 S2H Creator	54
5-2 S2H Quiz.....	60
Conclusion générale.....	63
Bibliographie	64

Liste des Tableaux

<i>Tableau 1 : Glossaire SCRUM</i>	10
<i>Tableau 2 : L'équipe d'un projet SCRUM</i>	10
<i>Tableau 3 : Backlog du produit.....</i>	14
<i>Tableau 4 Planification des Sprints.....</i>	17
<i>Tableau 5 : Description des acteurs du diagramme de cas d'utilisation global.....</i>	23
<i>Tableau 6 : Description textuelle du cas d'utilisation « Créer un Quiz ».....</i>	25
<i>Tableau 7 : Description textuelle du cas d'utilisation « Créer une Session »</i>	31
<i>Tableau 8 : Description textuelle du cas d'utilisation « Démarrer une Session »</i>	32
<i>Tableau 9 : Backlog du Sprint 1</i>	34
<i>Tableau 10 : Backlog du Sprint 2</i>	40
<i>Tableau 11 : Backlog du Sprint 3</i>	41
<i>Tableau 12 : Backlog du Sprint 4</i>	42
<i>Tableau 13 : Backlog du Sprint 5</i>	43
<i>Tableau 14 : Les phases du cycle DevOps</i>	45

Liste des figures

<i>Figure 1 : Lycée Dour Hicher</i>	3
<i>Figure 2: Principe de la méthode SCRUM</i>	10
<i>Figure 3 : Architecture de l'application.....</i>	16
<i>Figure 4 : Diagramme de déploiement.....</i>	17
<i>Figure 5 : Logo MEAN stack.....</i>	18
<i>Figure 6 : Logo Postman</i>	19
<i>Figure 7 : Logo Visual Studio Code.....</i>	20
<i>Figure 8 : Logo Entrprise Architect</i>	20
<i>Figure 9 : Logo MongoDB Atlas</i>	21
<i>Figure 10 : Diagramme de cas d'utilisation global</i>	22
<i>Figure 11 : Diagramme de cas d'utilisation « Gérer les Quiz »</i>	25
<i>Figure 12 : Diagramme de cas d'utilisation « Gérer les sessions »</i>	29
<i>Figure 13 : Diagramme d'activités « Gérer les Sessions »</i>	30
<i>Figure 14 : Diagramme de classes</i>	33
<i>Figure 15 : Esquisse landpage.....</i>	35
<i>Figure 16 : Esquisse page d'enregistrement</i>	36
<i>Figure 17 : Esquisse Landpage Createur.....</i>	36
<i>Figure 18 : Esquisse page de création de Quiz.....</i>	37
<i>Figure 19 : Esquisse Page liste des Quiz.....</i>	37
<i>Figure 20 : Les phases du cycle DevOps.....</i>	44
<i>Figure 21 : Catalogue IBM Cloud</i>	47
<i>Figure 22 : Cloud Foundry dans le catalogue IBM Cloud</i>	48
<i>Figure 23 : Mes ressources dans IBM Cloud</i>	49
<i>Figure 24 : Détails des ressources dans IBM Cloud</i>	50
<i>Figure 25 : La chaine d'outils S2H Creator.....</i>	50
<i>Figure 26 : Delivery Pipe Line de S2H Creator.....</i>	51
<i>Figure 27 : Le projet dans GitHub</i>	53
<i>Figure 28 : CI / CD</i>	53
<i>Figure 29 : Page d'accueil</i>	54
<i>Figure 30 : Page d'accueil du Créateur</i>	55
<i>Figure 31 : Liste des Quiz</i>	56
<i>Figure 32 : Création de Quiz</i>	56
<i>Figure 33 : Formulaire d'ajout d'une question.....</i>	57
<i>Figure 34 : Création de Session</i>	58
<i>Figure 35 : Liste des Sessions</i>	58
<i>Figure 36 : Landpage S2H Quiz</i>	60

<i>Figure 37 : Page de test S2H Quiz.....</i>	61
<i>Figure 38 : Page de résultats</i>	61
<i>Figure 39 : Page d'accueil sur Smartphone.....</i>	62
<i>Figure 40 : Page de tests sur Smartphone</i>	62

Liste des abréviations

- API : Application Programming Interface
- CASE : Computer-Aided Software Engineering
- CI / CD : Continuous Integration / Continuous Delivery (ou Continuous Deployment)
- CLI : Command Line Interface
- CRUD : Create Read Update Delete
- IA : Intelligence Artificielle
- IaaS : Infrastructure as a Service
- IAM : Identity and Access Management
- IBM : International Business Machines
- IDE : Integrated Development Environment
- IoT : Internet of Things
- MEAN : MongoDB Express Angular Node.js
- NoSQL : Not only SQL
- NTIC : Nouvelles Technologies de l'Information et de la Communication
- ODM : Object Data Modeling
- PaaS : Platform as a Service
- QCM : Question à Choix Multiple
- QCU : Question à Choix Unique
- REST : REpresentational State Transfer
- SDI : Software Defined Infrastructure
- SQL : Structured Query Language
- UML : Unified Modeling Language
- URL : Uniform Resource Locator

Introduction générale

Personne ne peut nier ni sous-estimer le rôle de l'évaluation dans le processus d'apprentissage. En effet cela permet aux enseignants de valider leurs méthodes d'enseignement ou de les revoir le cas échéant ainsi qu'aux apprenants d'évaluer leurs niveaux et de se fixer de nouveaux objectifs.

L'évaluation vise divers objectifs :

- ***L'évaluation au service de l'apprentissage*** : l'évaluation informe les enseignants sur les acquis des élèves et leur permet de planifier et d'orienter l'enseignement.
- ***L'évaluation en tant qu'apprentissage*** : l'évaluation permet aux élèves d'être conscients de leurs méthodes d'apprentissage (bien que généralement, dans notre système Tunisien il est imposé par l'enseignant) et d'en assumer la responsabilité.
- ***L'évaluation de l'apprentissage*** : les renseignements recueillis à la suite de l'évaluation permettent aux élèves, aux enseignants et aux parents, ainsi qu'à tout intervenant dans le processus d'apprentissage, d'être informés sur les résultats atteints à un moment précis afin de souligner les réussites, planifier les interventions correctives et continuer à favoriser la réussite.

L'évaluation doit être planifiée en fonction de ses objectifs. L'évaluation au service de l'apprentissage, l'évaluation en tant qu'apprentissage et l'évaluation de l'apprentissage ont chacun un rôle à jouer dans le soutien et l'amélioration de l'apprentissage des élèves. L'utilisation de ces trois types d'évaluation doit être équilibrée. La partie la plus importante de l'évaluation est la façon dont on interprète et on utilise les renseignements recueillis pour l'objectif visé.

L'évaluation fait partie intégrante du processus d'apprentissage. Elle est intimement liée aux programmes d'études et à l'enseignement. En même temps que les enseignants et les élèves travaillent en vue d'atteindre les résultats d'apprentissage des programmes d'études, l'évaluation joue un rôle essentiel en fournissant des renseignements utiles pour guider l'enseignement, pour aider les élèves à atteindre les prochaines étapes et pour vérifier les progrès et les réalisations. Pour l'évaluation en classe, les enseignants recourent à toutes

sortes de stratégies et d'outils différents, et ils les adaptent de façon à ce qu'ils répondent au but prévu et aux besoins individuels des élèves.

Les recherches et l'expérience démontrent que l'apprentissage de l'élève est meilleur quand :

- l'enseignement et l'évaluation sont basés sur des objectifs clairs ;
- l'enseignement et l'évaluation doivent varier en fonction des besoins des élèves ;
- les élèves interviennent dans le processus d'apprentissage (ils comprennent les buts de l'apprentissage et les critères caractérisant un travail de bonne qualité, reçoivent et mettent à profit les rétroactions descriptives et travaillent pour ajuster leur performance) ;
- l'information recueillie au moyen de l'évaluation est utilisée pour prendre des décisions favorisant l'apprentissage continu ;
- les parents sont bien informés sur les apprentissages de leur enfant et travaillent avec l'école pour planifier et apporter le soutien nécessaire ;
- les élèves, les familles et le public en général ont confiance au système.

Chapitre 1 Contexte du projet

Introduction

Le présent projet, intitulé « Réalisation d'une application Web de sondage et d'évaluation en temps réel », est effectué en vue de l'obtention du Diplôme National de Master Professionnel en Sciences et Technologies spécialité : Logiciels OpenSource, au sein du Lycée Farhat Hached à Douar Hicher.

Ce projet de fin d'études a comme objectif de développer un outil permettant à un enseignant (ou un formateur dans le sens large) d'évaluer ses apprenants automatiquement et en temps réel via une application Web.

Cela peut être très utile pour assurer et faciliter ce qu'on appelle l'évaluation continue surtout avec l'avènement des nouvelles réformes appliquées aux méthodes d'enseignement qui valorisent l'utilisation des NTIC (Nouvelles Technologies de l'Information et de la Communication).

1-1 Organisme d'accueil



Figure 1 : Lycée Dour Hicher

Le lycée Douar Hicher rebaptisé Lycée Farhat Hached est un établissement étatique d'enseignement secondaire situé à Douar Hicher de la ville de Manouba. Le lycée accueille aux alentours de cinq cent élèves répartis sur les quatre niveaux de l'enseignement secondaire encadrés par une cinquantaine d'enseignants. Il est dirigé par une équipe d'administratifs compétents dont mon encadrant professionnel Mme Mounira JLASSI au rôle de directrice d'établissement.

1-2 L'évaluation continue

1-2-1 Définition

« *L'évaluation continue est une évaluation permanente qui s'exerce à tout moment de l'apprentissage.* » (Dictionnaire LeParisien)

1-2-2 Caractéristiques de l'évaluation continue

Il existe certaines caractéristiques de l'évaluation continue qui la distinguent des tests traditionnels. On peut citer :

- Les tests sont de nature régulière et fréquente.
- Elle se concentre principalement sur le module de cours et assure des feedbacks réguliers.
- De plus, la forme de l'évaluation continue peut être différente de celle habituelle, compte tenu des objectifs finaux.
- La technique d'évaluation continue est un instrument efficace pour déterminer et développer les compétences.
- Cette méthode est de nature globale, cumulative, diagnostique, formative, orientée et systématique.

1-2-3 Objectifs de l'évaluation continue

L'évaluation continue vise trois objectifs principaux :

1. **Amélioration de l'apprentissage de l'élève.** Une évaluation continue et complète des connaissances acquises par l'étudiant au cours du module est très importante. De plus, cela garantit que l'étudiant investit un temps considérable dans l'étude, la préparation et le renforcement des compétences académiques. Cela garantit également que les étudiants sont en contact permanent avec le programme et tout ce qu'il a à offrir. Une évaluation constante, continue et régulière des performances et de l'apprentissage des élèves est la clé pour constituer une main-d'œuvre potentielle compétente et qualifiée.
2. **Amélioration des compétences pédagogiques.** La pression d'une évaluation continue peut également garantir une performance optimale des enseignants. Disons qu'un programme particulier ne comporte qu'un examen final pour 20 points à la fin de son cours. Dans ce cas, les étudiants, ainsi que les enseignants, trouveront la nécessité de jouer de manière cohérente tout au

long du trimestre et pas seulement vers la fin du cours, ce qui en fait un excellent improvisateur de performance.

3. **Amélioration du système éducatif et d'évaluation institutionnelle.** Un système éducatif qui comprend l'importance d'une évaluation globale est formidable. Non seulement ce type de système dépeint le développement, mais montre également à quel point il est sérieux lorsqu'il s'agit d'offrir des opportunités et des performances des étudiants.

1-3 Etude de l'existant

Dans ce chapitre je vais essayer de présenter les pratiques actuelles adoptées par les enseignants lors de l'évaluation de leurs apprenants ainsi que de leurs méthodes d'apprentissage, de critiquer l'existant puis proposer des solutions en vue de remédier aux lacunes constatées.

Au moment de l'écriture de ces lignes, les enseignants utilisent des moyens d'évaluation rudimentaires, des examens sur papier avec des notes d'évaluation sans suite. Ces évaluations sont de nature sommative et effectuées afin d'attribuer une note à l'apprenant à la fin d'un module ou d'une période (Trimestre ou semestre).

1-4 Critiques de l'existant

Ces formes d'évaluation ne peuvent pas être au service des objectifs visés par l'évaluation surtout que le système éducatif actuel pousse l'apprenant à favoriser la note (résultat de l'évaluation classique) plutôt que l'apprentissage en lui-même.

Le besoin d'avoir une évaluation continue et de faire un suivi de l'évolution des compétences acquises s'avèrent indispensables pour mieux discerner les lacunes du processus d'apprentissage.

D'une autre part, et même si l'enseignant s'efforce à effectuer des évaluations tout au long de l'apprentissage, la tâche s'avère fastidieuse vu la grande quantité de copies à corriger en plus de l'espace horaire nécessaire à l'effectuer.

Il faut aussi souligner le manque de fiabilité d'une seule note attribuée d'une manière sommative qui peut ne pas refléter le niveau réel de l'apprenant.

Peu d'enseignants disposent d'outils permettant de gérer leurs groupes d'apprenants afin de suivre l'évolution de leurs compétences.

Il n'existe pratiquement pas de collaboration entre les enseignants pour échanger leurs savoir faire et leurs expériences.

1-5 Proposition de solutions

Une solution informatique doit être sollicitée pour simplifier la tâche de l'enseignant et d'assurer la qualité de l'apprentissage. Cette dite solution doit être capable de :

- Créer des Quiz en guise de tests contenant des questions sous différentes formes (QCM, QCU, Saisie, Réorganisation...).
- Effectuer des évaluations tout au long de la formation (Essentiellement, des évaluations de prérequis, des évaluations formatives ainsi que des évaluations sommatives)
- Effectuer une évaluation automatique en temps réel des réponses des apprenants aux quiz proposés.
- Gérer les apprenants.
- Gérer les groupes d'apprenants.
- Partager des quiz avec d'autres enseignants.
- Offrir un Dashboard aux enseignants afin d'avoir une vision globale sur les résultats des évaluations.

De nombreuses applications existent sur le marché, telles que Mentimeter et Kahoot permettent d'effectuer des sondages d'audience mais ne répondent pas aux besoins des enseignants tels que la gestion des groupes d'autant plus qu'elles deviennent payantes pour de fonctionnalités supplémentaires.

Il sera donc judicieux de créer une application permettant de répondre aux différents besoins des enseignants et d'assurer chacune des fonctionnalités précédemment citées.

Conclusion

Dans ce chapitre et après avoir introduit le cadre du projet, j'ai présenté l'évaluation continue qui manque à notre système éducatif actuel, critiqué ce dernier et proposé des solutions afin de bénéficier des avantages et de la valeur ajoutée que peut l'évaluation continue donner au processus d'apprentissage.

Chapitre 2 Méthodologie et démarche

Introduction

Aux débuts de la réalisation de ce travail, les besoins étaient un peu flous, je n'étais pas en mesure de définir les fonctionnalités avec précision. A partir de ce moment-là, il était judicieux de prendre en considération la « User Experience » afin de raffiner les besoins et de réorienter le projet au fil de son avancement.

De plus, je devais aussi prendre en considération la contrainte de temps ainsi que la taille de l'équipe (Formée uniquement par moi-même) et d'assurer des livrables aussi rapidement que possible (Early To Market).

Dès lors il était impératif d'opter pour l'approche CI / CD afin d'assurer des feedbacks réguliers des utilisateurs comportant essentiellement leurs remarques sur ce qui a été réalisé ainsi que des propositions pour les travaux futurs. D'une autre part, l'utilisation d'une méthodologie agile s'impose.

2-1 Qu'est-ce que le développement agile

Agile, que certains appellent désormais l'agilité, vise à éliminer les obstacles au processus et à permettre aux principales parties prenantes, telles que les développeurs et les utilisateurs, de collaborer plus étroitement pour accélérer la livraison. Agile met en évidence la constance du changement et reconnaît qu'en tant que producteurs de logiciels, nous ne savons pas souvent tout ce dont nous avons besoin pour concevoir, développer et livrer avec succès des logiciels de haute qualité dans des cycles de vie monolithiques.

Ainsi, bien que l'agilité ait fini par signifier différentes choses au cours des deux dernières décennies, ses principes fondamentaux sont les mêmes : supprimer les barrières de processus qui autonomisent les individus, produire rapidement des logiciels fonctionnels, collaborer étroitement avec les utilisateurs et réagir au changement.

2-2 Qu'est-ce que (CI / CD) ?

CI /CD rime pour Continuous Integration / Continuous Delivery (ou pour certains Continuous Deployment) traduit en français en « Intégration Continue / Livraison Continue (ou Déploiement Continu)».

(CI) est une pratique du génie logiciel où les membres d'une équipe intègrent leurs travaux de plus en plus fréquemment. Conformément à la pratique de l'intégration continue, les équipes s'efforcent d'intégrer au moins quotidiennement et même toutes les heures, une intégration proche de celle qui se produit «en continu».

Historiquement, l'intégration a été une activité d'ingénierie coûteuse. Ainsi, pour éviter les conflits, CI met l'accent sur les outils d'automatisation qui pilotent la construction et les tests, se concentrant finalement sur la réalisation d'un cycle de vie défini par logiciel. Lorsque CI réussit, les efforts de construction et d'intégration diminuent et les équipes peuvent détecter les erreurs d'intégration aussi rapidement que possible.

(CD) est à l'empaquetage et au déploiement ce que CI est de construire et de tester. Les équipes pratiquant le CD peuvent créer, configurer et empaqueter le logiciel et orchestrer son déploiement de manière à ce qu'il puisse être mis en production d'une manière définie par logiciel (faible coût, automatisation élevée et détection automatique et précoce des erreurs) à tout moment.

Les pratiques CI / CD facilitent directement le développement agile, car les changements logiciels atteignent plus fréquemment la production. En conséquence, les utilisateurs ont plus d'opportunités d'expérimenter et de fournir des commentaires sur le changement.

2-3 Culture DevOps

DevOps se concentre sur les limites de la culture et des rôles dans le processus de développement agile. L'intention de DevOps est d'éviter l'impact négatif que la surspécialisation a sur la prévention d'une réponse rapide, voire efficace, aux problèmes de production. Les organisations DevOps éliminent les barrières entre les opérations et l'ingénierie en formant chaque équipe aux compétences de l'autre. Cette approche améliore la capacité de chacun à apprécier et à participer aux tâches de chacun et conduit à une collaboration de plus haute qualité et une communication plus fréquente.

2-4 Quelle relation entre eux ?

Quel est le lien entre CI / CD, agile et DevOps dans le développement réel ? Les équipes d'ingénieurs commencent souvent par CI parce que c'est dans leur timonerie. Une approche DevOps peut aider les organisations à comprendre la configuration, le packaging et l'orchestration nécessaires pour définir encore plus le cycle de vie par le logiciel, créant

ainsi une pratique de CD plus précieuse. La pratique de CI / CD dans DevOps, à son tour, ajoute au développement agile.

En conclusion :

- Agile se concentre sur les processus mettant en évidence le changement tout en accélérant la livraison.
- CI / CD se concentre sur les cycles de vie définis par logiciel en mettant en évidence les outils qui mettent l'accent sur l'automatisation.
- DevOps se concentre sur la culture en mettant en évidence les rôles (équipe développeurs/ équipe Opérationnelle) qui mettent l'accent sur la réactivité.

2-5 Adoption

2-5-1 Méthodologie et langage de modélisation adoptés

Mon choix s'est fixé sur SCRUM comme méthodologie de gestion de projet, UML comme langage de modélisation ainsi que IBM Cloud comme infrastructure pour assurer les pratiques CI / CD.

2-5-1-1 SCRUM

La méthodologie SCRUM appartenant à la famille des méthodologies agiles a comme principe de travail la répartition du projet en périodes de temps appelés « Sprints » pouvant chacune durer 2 à 4 semaines.

Chaque jour toute l'équipe se réunit pour ce qui est appelé « la mêlée quotidienne » qui dure au maximum 15 minutes afin de mettre au point les tâches réalisées durant les dernières 24 heures. A la fin de chaque sprint, l'équipe de développement se réunit pour effectuer « la revue du sprint » qui se traduit par la réalisation du bilan du sprint, suivi d'une « rétrospective » pendant laquelle l'équipe révise le rendu du sprint pour des petites rectifications ainsi que les problèmes rencontrés pour les résoudre par la suite avant de classer la tâche comme « valide » et la livrer aux testeurs et éventuellement aux utilisateurs. En suivant le même enchaînement pour tous les sprints, on aura enfin la version « release » du travail qui est la version finale qui peut être livrée au client.

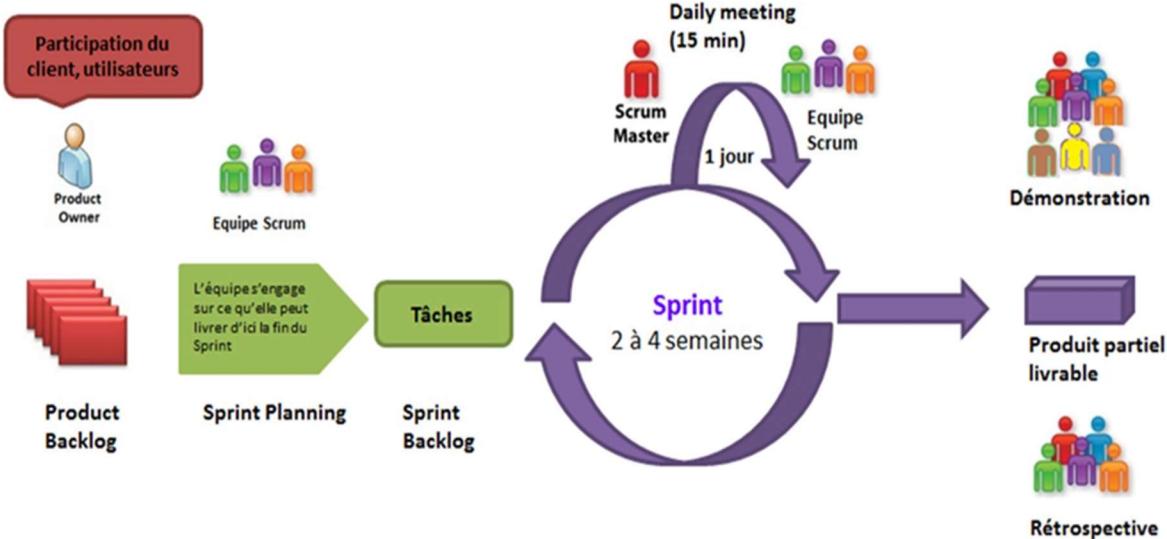


Figure 2: Principe de la méthode SCRUM

Dans un contexte SCRUM, je vais devoir utiliser quelques termes propres à cette méthodologie. En voici les plus pertinents :

Tableau 1 : Glossaire SCRUM

Terme	Définition
Backlog du produit	La définition des besoins fonctionnels sous forme de (user story)
Backlog du Sprint	la liste des tâches à implémenter dans un sprint, classées par importances et état
Produit partiel	la liste des tâches à implémenter dans un sprint, classées par importances et état
Mêlé quotidienne	le SCRUM meeting et réunion organisée tous les jours debout qui dure au plus 15 minutes

Tableau 2 : L'équipe d'un projet SCRUM

Rôle	Mission
SCRUM Master	Supervision de l'avancement du projet et des activités de l'équipe
Propriétaire du produit (Product Owner)	Présentation des caractéristiques et des fonctionnalités du produit à développer et approbation du produit à livrer
L'équipe de développeurs	Réalisation des user stories et élaboration des sprints

2-5-1-2 UML

D'après la société SPARX, UML peut être défini comme suit :

« UML est un standard de l'industrie modelant la langue avec une notation graphique riche et un ensemble complet de diagrammes et d'éléments. »

En fait, je vais devoir utiliser quelques diagrammes UML afin de modéliser des idées tout au long de la réalisation du produit. Je peux en citer le diagramme de cas d'utilisation, le diagramme de classes, le diagramme de séquences ainsi que le diagramme de déploiement.

2-5-2 IBM Cloud

IBM Cloud est une plate-forme de cloud computing ouverte et normalisée qui aide les développeurs et les entreprises à créer, déployer, exécuter et mettre à l'échelle rapidement des applications et des services.

La plateforme IBM Cloud combine Platform As A Service (PaaS) avec Infrastructure As A Service (IaaS) pour offrir une expérience intégrée. La plate-forme s'adapte et prend en charge à la fois les petites équipes et organisations de développement et les grandes entreprises. Globalement déployée dans Data Center du monde entier, la solution que vous construisez sur IBM Cloud tourne rapidement et fonctionne de manière fiable dans un environnement testé et pris en charge dans lequel vous pouvez avoir confiance. La plate-forme IBM Cloud est une plate-forme complète et de niveau entreprise spécialement conçue pour les charges de travail d'intelligence artificielle (IA) à forte intensité de données et les suites d'applications cloud natives fournies sur une infrastructure définie par logiciel (SDI).

Conclusion

Dans ce chapitre j'ai présenté la méthodologie de conception, le langage de modélisation ainsi que les pratiques que j'ai choisi pour assurer la réussite de mon projet. Ceci en donnant un aperçu sur chacun d'entre eux et justifiant mes choix.

Chapitre 3 Réalisation

Introduction

Après avoir été fixé sur la méthodologie de conception, le langage de modélisation, les pratiques logicielles ainsi que la plateforme de déploiement, l'étape suivante est la réalisation pratique de l'application.

Dans ce chapitre je vais commencer par planifier le projet utilisant la méthodologie SCRUM pour enchaîner avec mes choix des technologies de développement et terminer par la description de chacun des sprints composants la phase de réalisation. Pendant l'étape de planification je vais utiliser UML pour modéliser graphiquement quelques aspects.

3-1 Planification du projet

Dans cette partie je vais essayer d'identifier les besoins fonctionnels et non fonctionnels de l'application, de présenter le Backlog de produit pour terminer par la planification des sprints.

3-1-1 Analyse des besoins

Les besoins de l'application se répartissent en besoins fonctionnels et besoins non fonctionnels.

3-1-1-1 Les besoins fonctionnels

- **Gestion des Quiz** : L'application doit permettre aux enseignants de créer, consulter, modifier et de supprimer des quiz ainsi que de dupliquer des quiz partagés.
- **Gestion des groupes d'apprenants** : L'application doit permettre aux enseignants de créer, consulter, modifier et de supprimer des groupes d'apprenants.
- **Gérer des Sessions de Quiz** : L'application doit permettre aux enseignants de créer, démarrer et arrêter des sessions de quiz pendant lesquelles les apprenants peuvent répondre aux questions proposés dans le quiz.
- **Dashboard** : L'enseignant doit être en mesure de consulter les statistiques relatives aux réponses des apprenants.

- **Répondre aux Quiz** : L'application doit permettre aux apprenants de répondre aux questions d'un quiz dans le cadre d'une session démarrée par l'enseignant.
- **Consultation des résultats** : Les apprenants doivent être en mesure de connaître les résultats d'une session de réponses et éventuellement avoir les bonnes réponses.
- **Sécurité** : L'application doit être suffisamment sécurisée pour interdire toute opération malveillante.

3-1-1-2 Les besoins non fonctionnels

- **Ergonomie et intuitivité** : L'application doit être facile à manipuler en présentant des interfaces ergonomiques. En effet, certaine fonctionnalités peuvent être accessible à partir de plusieurs points d'entrée.
- **Disponibilité** : L'application doit fonctionner même si la demande aux ressources est importante.
- **Evolutivité** : L'application doit être extensible, c'est-à-dire facilement modifiable et maintenable, s'adaptant aux nouvelles exigences des utilisateurs.

3-1-2 Elaboration du Backlog du produit

Le Backlog du produit est l'artefact le plus important de SCRUM il s'agit d'une liste de fonctionnalités à réaliser. Ces fonctionnalités, exprimées en des besoins, sont priorisées par le Product Owner.

Le tableau qui suit présente l'ensemble des histoires utilisateur (User Story) décrites à l'aide de leur Id, Feature, User story et leur priorité (importance) :

- **Le champ ID** : Un identifiant unique, que l'on incrémente à chaque nouvelle histoire, cet identifiant permet de garder la trace des histoires même quand on les renomme.
- **Feature** : un produit peut se diviser en fonctionnalités (Features). Il englobe les histoires utilisateurs d'un même module.
- **Le champ User Story** : Met en évidence d'une manière claire les fonctionnalités désirées par l'utilisateur sous la forme d'une phrase.
- **Le champ Priority** : Définie comme son nom l'indique l'importance contractuelle de la réalisation de cette partie du projet.

Tableau 3 : Backlog du produit

ID	Feature	User Story	Priority
1	S'enregistrer	En tant qu'enseignant je veux avoir un espace de travail propre à moi.	Elevé
2	S'authentifier	En tant qu'enseignant je veux m'assurer que personne d'autre n'a accès à mes informations.	Elevé
3		En tant qu'enseignant je veux m'assurer qu'uniquement l'apprenant autorisé n'a accès aux sessions de quiz.	Elevé
4		En tant qu'enseignant je peux créer, modifier, supprimer et consulter des groupes d'apprenants.	Elevé
5	Gestion des groupes et des apprenants	En tant qu'enseignant je peux ajouter et supprimer des apprenants dans un groupe.	Elevé
6		En tant qu'enseignant je peux consulter la liste de mes apprenants.	Moyen
7		En tant qu'enseignant je préfère que les login et mots de passe des apprenants soient générés automatiquement.	Bas
8		En tant qu'enseignant je peux créer, modifier, supprimer et consulter mes Quiz.	Elevé
9		En tant qu'enseignant je peux partager mes quiz avec d'autres enseignants.	Moyen
10	Gestion des Quiz	En tant qu'enseignant je peux consulter les quiz partagés par d'autres enseignants et éventuellement les importer.	Moyen
11		En tant qu'enseignant je peux ajouter, supprimer et modifier des questions dans des quiz et y déposer les bonnes réponses qui	Elevé

		seront utilisées lors de l'évaluation des réponses des apprenants.	
12	Gestion des Session	En tant qu'enseignant je peux créer et mettre en instance une session relativement à un de mes quiz.	Elevé
13		En tant qu'enseignant je peux démarrer une session de quiz afin que les apprenants puissent se connecter et répondre aux questions.	Elevé
14		En tant qu'enseignant je veux qu'un code généré aléatoirement soit attribué à une session récemment démarrée pour qu'elle soit accessible aux apprenants.	Elevé
15		En tant qu'enseignant je peux choisir si une session soit accessible uniquement à un groupe donné ou qu'elle soit ouverte à tous ceux qui en ont le code d'accès.	Moyen
16		En tant qu'enseignant je peux arrêter une session pour qu'il n'y soit plus de possibilité aux apprenants de poster leurs réponses ni aux nouveaux venus de se connecter à la session.	Elevé
17		En tant qu'enseignant je peux consulter les notes d'évaluation des apprenants dans une session donnée.	Elevé
18	Répondre à un quiz	En tant qu'enseignant je peux consulter des statistiques de mon archive de sessions.	Moyen
19		En tant qu'apprenant je peux répondre à un quiz dont la session a été démarrée.	Elevé
20	Consulter résultat d'évaluation.	En tant qu'apprenant je peux connaître ma note d'évaluation après avoir posté mes réponses et éventuellement connaître les bonnes réponses si mon enseignant l'a permis.	Moyen

3-1-3 Architecture de l'application

J'ai choisi que l'application soit répartie en trois applications en respectant les bonnes pratiques du style d'architecture REST.

- S2H Creator : Une application Front-end accessible par les enseignants afin de gérer leurs quiz, sessions, groupes...
- S2H Quiz : Une application Front-end accessible par les apprenants afin de répondre aux quiz proposés et de consulter leurs résultats.
- S2H API : Une application Back-end sécurisée assurant la « logique métier » (Business Layer) de l'application et la persistance de son état dans la Base de données.

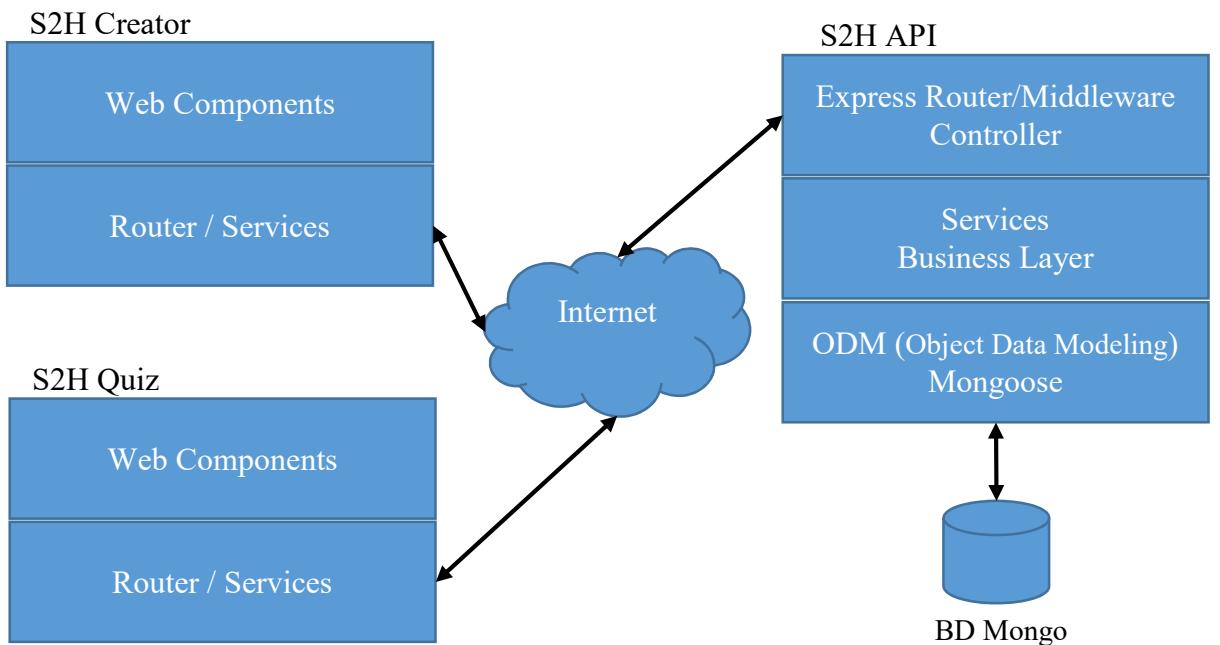


Figure 3 : Architecture de l'application

3-1-4 Diagramme de déploiement

Les diagrammes de déploiement sont utilisés pour illustrer l'architecture matérielle d'un système. En fait, les trois applications sont déployées sur le Cloud IBM et la base de données réside chez MongoDB Atlas.

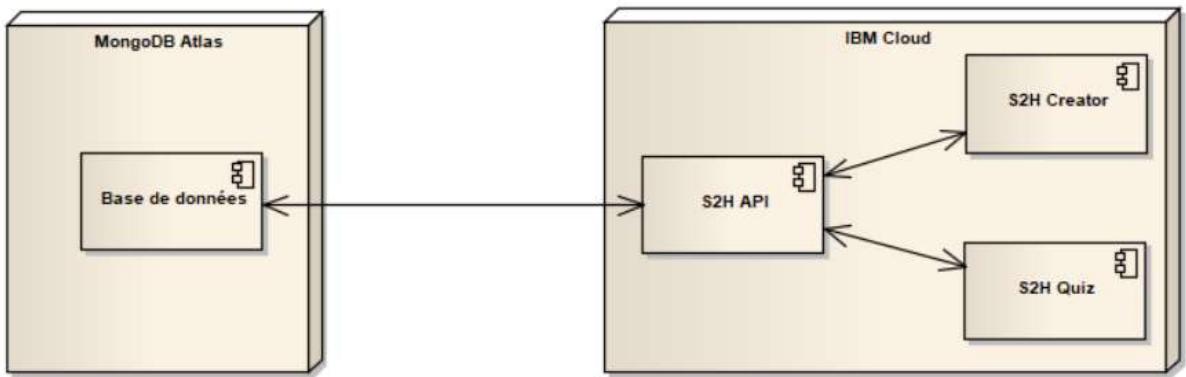


Figure 4 : Diagramme de déploiement

MongoDB Atlas offre une espace de stockage gratuit de 512 MO qui me paraît largement suffisant pour un début avant de trouver une solution de stockage radicale.

La plateforme IBM Cloud permet d'héberger des applications dans Cloud Foundry et d'assurer les pratiques CI / CD. J'expliquerai cela avec de plus amples détails dans le chapitre suivant.

3-1-5 Planification des sprints

Après l’élaboration du Backlog du produit, l’étape suivante consiste à planifier les sprints. Chaque sprint est constitué d’un ensemble de User Stories (décris dans le backlog du produit) et s’étalant sur une période donnée.

Le tableau suivant illustre cette planification :

Tableau 4 Planification des Sprints

Sprint ID	Description	User Stories	Durée (semaines)
Sprint 1	Esquisse des interfaces utilisateur. Scaffolding avec Angular de l’application Front-end S2H Creator et avec Node.js de l’application Back-end (S2H API) et avoir l’approbation des utilisateurs testeurs.		1
Sprint 2	Implémentation de la fonctionnalité « Enregistrement et Authentification » et de la gestion des groupes au niveau de S2H Creator et S2H API.	1, 2, 3, 4, 5, 6, 7	3

Sprint 3	Implémentation du design des Quiz (CRUD) et implémentation de la fonctionnalité des quiz partagés au niveau de S2H Creator et S2H API.	8, 9, 10, 11	4
Sprint 4	Implémentation de la gestion des sessions au niveau de S2H Creator et S2H API.	12, 13, 14, 15, 16	2
Sprint 5	Implémentation de S2H Quiz et implémentation du tableau de bord (Dashboard) des résultats des quiz et statistiques au niveau de S2H Creator.	17, 18, 19, 20	4

Remarque : Les durées présentées dans le tableau précédent sont de nature estimative.

3-2 Choix de l'environnement de développement et des outils

Afin de réaliser cette application, j'ai opté pour MEAN Stack comme technologies de développement, Postman pour effectuer les tests API, MongoDB Atlas comme plateforme d'hébergement de ma base de données, Enterprise Architect comme outil de modélisation UML et Visual Studio Code comme IDE.

3-2-1 La pile de développement MEAN Stack



Figure 5 : Logo MEAN stack

Le terme MEAN Stack (pile) fait référence à un ensemble de technologies basées sur JavaScript utilisées pour développer des applications Web. MEAN est un acronyme pour MongoDB, ExpressJS, Angular et Node.js. Du client au serveur en passant par la base de données, MEAN est un JavaScript full stack.

3-2-1-1 Node.js

Node.js est un environnement d'exécution JavaScript côté serveur. Il s'agit d'une plate-forme basée sur le moteur d'exécution JavaScript V8 de Google Chrome. Il aide à créer rapidement des applications hautement évolutives et simultanées.

3-2-1-2 ExpressJS

Express est un framework léger utilisé pour créer des applications Web dans Node. Il fournit un certain nombre de fonctionnalités robustes pour la création d'applications Web à une (SPA) ou plusieurs pages ainsi que des API.

3-2-1-3 MongoDB

MongoDB est un système de base de données NoSQL sans schéma (SchemaLess) contrairement aux bases de données relationnelles. MongoDB enregistre les données au format JSON binaire, ce qui facilite la transmission des données entre le client et le serveur.

3-2-1-4 Angular

Angular est un framework TypeScript développé par Google. Il fournit des fonctionnalités impressionnantes comme la liaison de données bidirectionnelle (Two Way Binding). C'est une solution complète pour un développement front-end rapide et impressionnant.

3-2-2 Autres outils utilisés :

3-2-2-1 Postman



Figure 6 : Logo Postman

Postman est une plateforme de collaboration pour le développement d'API. Les fonctionnalités de Postman simplifient chaque étape de la création d'une API et rationalisent la collaboration afin que vous puissiez créer de meilleures API, plus rapidement.

3-2-2-2 Visual Studio Code



Figure 7 : Logo Visual Studio Code

Visual Studio Code est avant tout un éditeur de texte qui a acquis au fil du temps des fonctionnalités impressionnantes.

Visual Studio Code combine la simplicité d'un éditeur de code source avec des outils de développement puissants, tels que le débogage et « IntelliSense/code completion ».

D'abord et avant tout, c'est un éditeur qui ne vous gêne pas. Le cycle d'édition-construction-débogage délicieusement fluide signifie moins de temps à manipuler votre environnement et plus de temps à exécuter vos idées.

3-2-2-3 Entreprise Architect



Figure 8 : Logo Enterprise Architect

Enterprise Architect est l'un des logiciels de modélisation et de conception UML (CASE : Computer-Aided Software Engineering) les plus connus, c'est un produit de la société Sparx Systems. Il permet de couvrir toutes les étapes de conception et de réalisation d'applications.

3-2-2-4 Mongo Atlas



Figure 9 : Logo MongoDB Atlas

MongoDB Atlas est une base de données entièrement gérée dans le cloud et développée par les mêmes personnes qui ont construit MongoDB. Elle offre gratuitement un espace de stockage plus ou moins suffisant pour gérer vos données.

3-3 Diagramme de cas d'utilisation global

En se basant sur le backlog du produit passons à l'élaboration du diagramme de cas d'utilisation décrivant schématiquement les différentes fonctionnalités de l'application réparties en cas d'utilisation (Use Cases) ainsi que les différents acteurs externes à notre système.

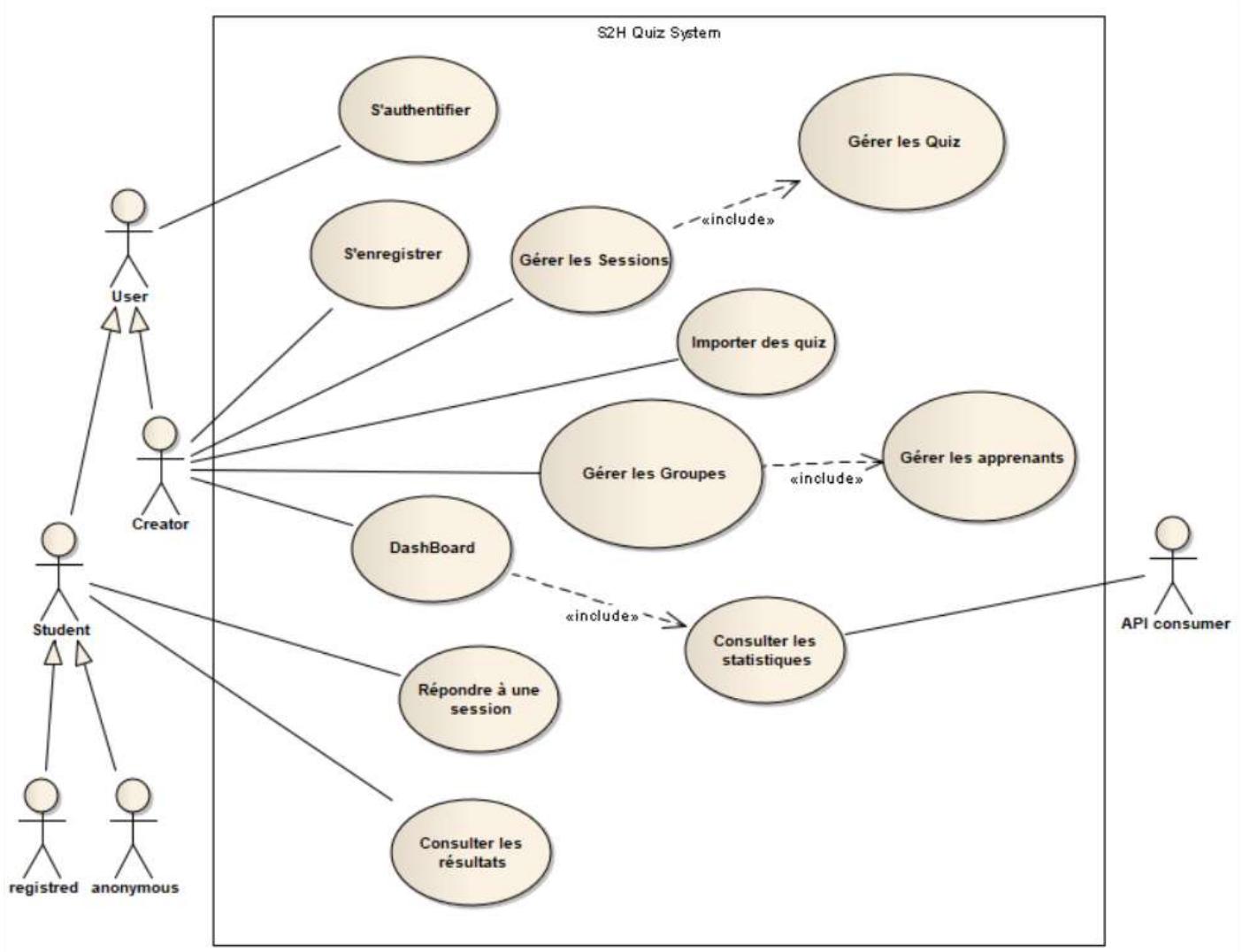


Figure 10 : Diagramme de cas d'utilisation global

3-3-1-1 Description des acteurs :

Le tableau suivant décrit chacun des acteurs présents dans le diagramme de cas d'utilisation :

Tableau 5 : Description des acteurs du diagramme de cas d'utilisation global

Acteur	Description
Creator	C'est le créateur de quiz, en d'autres termes c'est l'enseignant qui crée les quiz en vue d'être proposés comme moyen d'évaluation.
Student	C'est le consommateur de quiz, c'est l'apprenant qui est censé répondre aux questions des quiz.
API Consumer	C'est un acteur générique représentant toute application externe au système ayant accès aux statistiques via l'API.

3-3-2 Description des cas d'utilisation :

- **S'authentifier** : Ce cas d'utilisation permet au créateur de quiz (Enseignant) de se connecter en saisissant son email et mot de passe pour avoir accès à son espace de travail.
Pour un apprenant, la saisie de son login et mot de passe (générés et fournis par le créateur de quiz) lui permet d'avoir accès à une session de réponses.
- **S'enregistrer** : Ce cas d'utilisation permet au créateur futur de saisir ses informations personnelles (Nom Prénom, e-mail, genre, mot de passe) afin de créer un compte créateur lui permettant d'accéder aux différentes fonctionnalités de l'application.
- **Gérer les quiz** : Ce cas d'utilisation permet au créateur de créer, supprimer, modifier et consulter des quiz. Un quiz est formé par des questions de différentes formes (QCM, QCU, Saisie et Organisation). Lors de l'ajout d'une question à un quiz, le créateur doit spécifier les bonnes réponses. En modifiant un quiz et même au moment de sa création, le créateur peut supprimer ou modifier une question. Ce cas d'utilisation permet aussi à un créateur d'exposer certains quiz pour qu'ils puissent être partagés avec d'autres enseignants.
- **Gérer les sessions** : Ce cas d'utilisation permet à un créateur de créer à partir d'un quiz une session de réponses qui peut être démarrée à un instant voulu puis arrêtée quand le créateur décide de ne plus vouloir recevoir de réponses des apprenants.
- **Importer des Quiz** : Ce cas d'utilisation permet à un enseignant de consulter les quiz partagés par d'autres enseignants et éventuellement les importer dans ses propres quiz.

- **Gérer les apprenants** : Ce cas d'utilisation permet à un enseignant d'ajouter, supprimer ou modifier un apprenant et de pouvoir consulter la liste de tous ses apprenants. Lors de la création d'un apprenant, un login et un mot de passe sont générés automatiquement.
- **Gérer les groupes** : Ce cas d'utilisation permet à un enseignant de créer des groupes d'apprenants, de les supprimer, modifier ou consulter.
- **Consulter les statistiques** : Ce cas d'utilisation n'a pas d'interface graphique, c'est un module API permettant à la fonctionnalité « Dashboard » d'octroyer les informations dont elle a besoin et à d'autres applications tierces d'en bénéficier. Ces informations représenteront un recueil des réponses des apprenants.
- **Dashboard** : Ce cas d'utilisation permet d'afficher pour un enseignant les informations offertes par le cas d'utilisation précédent dans une interface Web.
- **Répondre à une session** : Ce cas d'utilisation permet à un élève de répondre aux questions d'un quiz dans le cadre d'une session après avoir été connecté.
- **Consulter les résultats** : Ce cas d'utilisation permet à un élève de consulter son score final après avoir répondu à un quiz et éventuellement avoir une idée sur les bonnes réponses (si l'enseignant l'a autorisé lors de la création de la session).

3-4 Raffinement du cas d'utilisation « Gérer les Quiz »

Procédons maintenant au raffinement du cas d'utilisation « Gérer les Quiz ». Cela inclura un diagramme de cas d'utilisation ainsi que la description textuelle et un diagramme de séquences du cas d'utilisation le plus pertinent qu'il inclut, à savoir « Créer un Quiz ».

3-4-1 Diagramme de cas d'utilisation de « Gérer les Quiz »

Le cas d'utilisation « Gérer les Quiz » est suffisamment complexe pour le décomposer en d'autres cas d'utilisation qui le composeront. Dans ce qui suit un diagramme de cas d'utilisation représentant cette décomposition.

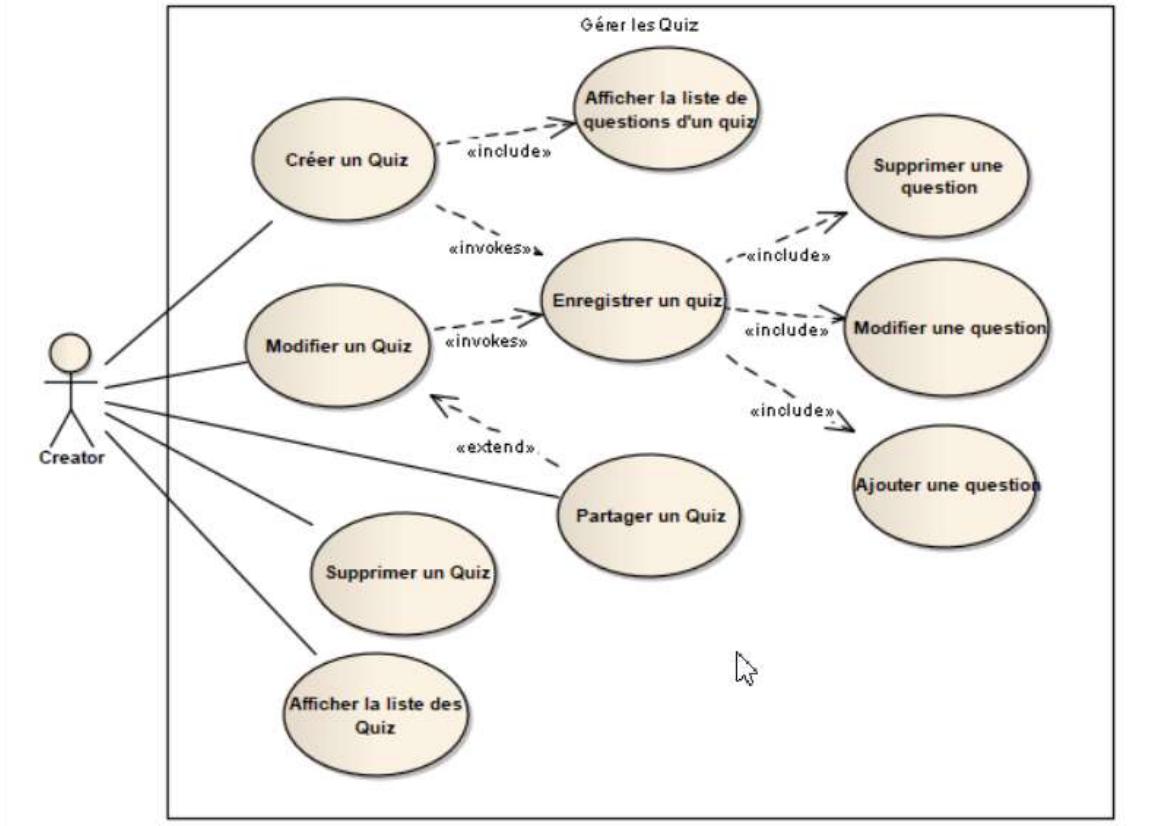


Figure 11 : Diagramme de cas d'utilisation « Gérer les Quiz »

A titre d'exemple, prenons le cas d'utilisation « Crée un Quiz » qui me paraît le plus important dans cette décomposition et essayons de l'étudier en effectuant une description textuelle et en réalisant un diagramme de séquences.

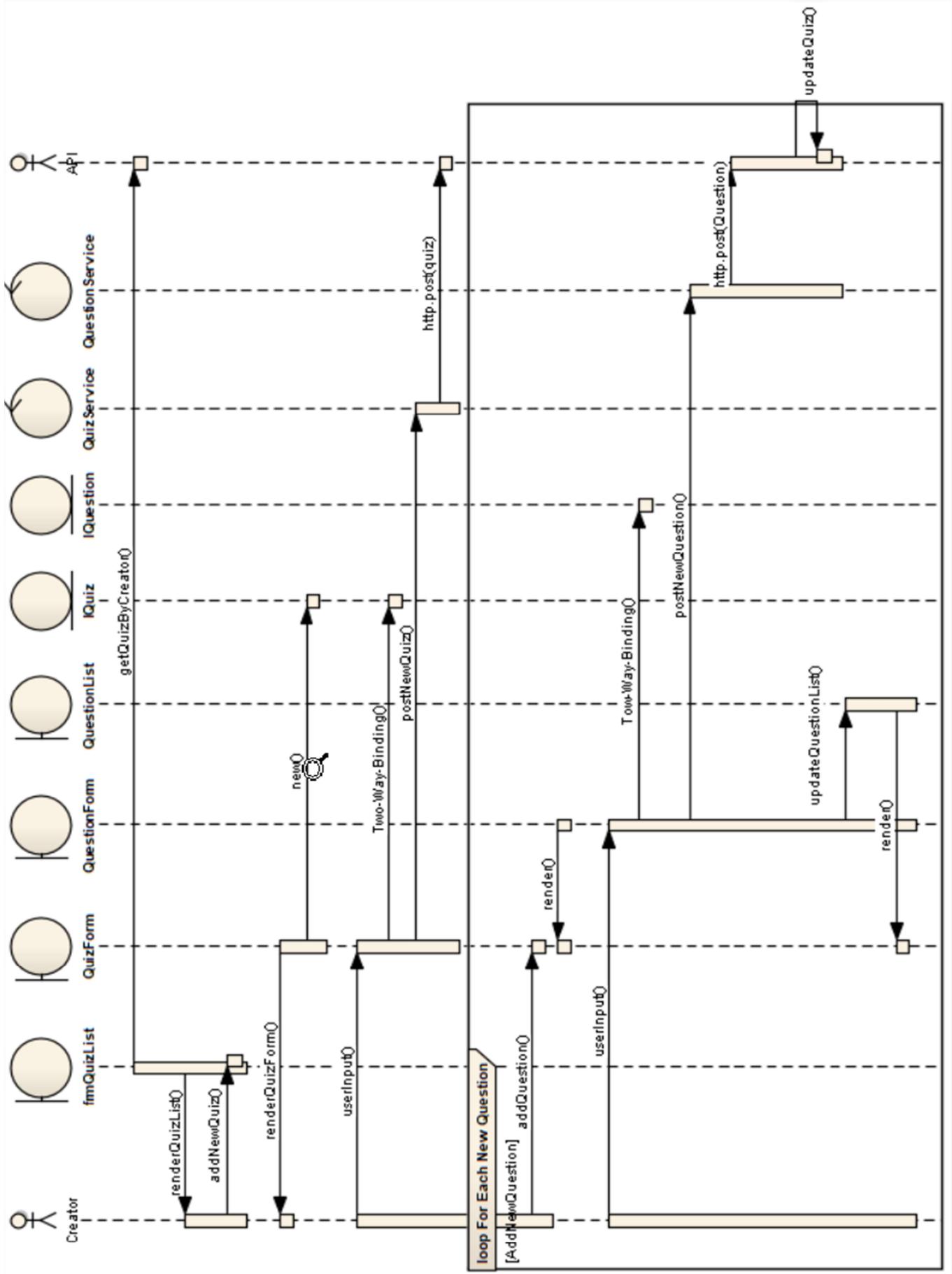
3-4-2 Description textuelle du cas d'utilisation « Crée un Quiz »

Tableau 6 : Description textuelle du cas d'utilisation « Crée un Quiz »

Titre	Crée un Quiz
Acteurs	Creator
But	Crée et sauvegarder un quiz contenant des questions.
Pré condition	Le Creator doit être authentifié.
Description	Ce cas d'utilisation doit permettre au Creator de créer un quiz, en saisissant ses informations ainsi que les questions qui lui sont attribuées.
Scénario nominal	<ol style="list-style-type: none"> 1. Creator demande d'afficher la liste de ses Quiz. 2. Le système affiche la liste des Quiz. 3. Creator demande d'ajouter un nouveau Quiz en cliquant sur le bouton « + »

	<p>4. Le système affiche l'interface de création de Quiz.</p> <p>5. Creator saisi les informations du Quiz et valide en cliquant sur le bouton « Créer le Quiz ».</p> <p>6. Le système sauvegarde le Quiz.</p> <p>7. Creator clique sur le bouton « Ajouter une question ».</p> <p>8. Le système affiche une boîte de dialogue demandant le type de question (QCM, QCU, Saisie, Ordonnancement)</p> <p>9. Creator choisit le type de question et soumet son choix.</p> <p>10. Le système affiche un formulaire de saisie de question.</p> <p>11. Creator saisi les informations de la question (Texte de la question, liste des réponses proposées, indication des bonnes réponses) puis valide l'ajout de la question.</p> <p>12. Le système met à jour la liste des questions déjà affichée.</p> <p>13. Le système fait disparaître le formulaire de saisie de questions.</p> <p>14. Creator peut ajouter une nouvelle question en cliquant sur le bouton « Ajouter une question »</p> <p>Retour à l'étape 8</p>
--	--

3-4-3 Diagramme de séquences de «Créer un Quiz»



Ce diagramme de séquences représente une description schématique du scénario nominal présenté dans description textuelle du cas d'utilisation. Il représente les interactions entre l'acteur principal (Creator) et les différents objets du système ainsi les interactions entre ces derniers.

3-4-3-1 Les objets du diagramme de séquences

Les différents objets du diagramme de séquences précédent sont les suivants :

- **Creator** : C'est l'utilisateur du système (L'enseignant voulant créer un Quiz)
- **frmQuizList** : C'est une interface graphique contenant la liste des quiz du Creator affichés sous la forme d'un tableau.
- **QuizForm** : C'est une interface graphique contenant un formulaire permettant la saisie des informations d'un Quiz.
- **QuestionForm** : C'est une interface graphique contenant un formulaire permettant la saisie des informations d'une Question et faisant partie de QuizForm.
- **QuestionList** : C'est une interface graphique affichant la liste de questions d'un Quiz sous forme d'un tableau et faisant partie de QuizForm.
- **IQuiz** : C'est un objet en mémoire permettant de sauvegarder l'état (state) du Quiz en cours de création.
- **IQuestion** : C'est un objet en mémoire permettant de sauvegarder l'état (state) de la question en cours de création.
- **QuizService** : C'est un service faisant partie la couche métier de l'application et permettant d'interagir avec l'API. Il est responsable des interactions relatives aux Quiz.
- **QuestionService** : C'est un service faisant partie la couche métier de l'application et permettant d'interagir avec l'API. Il est responsable des interactions relatives aux Questions.
- **API** : C'est l'API de l'application (S2H API).

3-5 Raffinement du cas d'utilisation « Gérer les sessions »

Procémons maintenant au raffinement du cas d'utilisation « Gérer les sessions ». Cela inclura un diagramme de cas d'utilisation, un diagramme d'activités ainsi que la description textuelle de quelques cas d'utilisation qui y sont inclus.

3-5-1 Diagramme de cas d'utilisation de « Gérer les sessions »

Le cas d'utilisation « Gérer les Sessions » est suffisamment complexe pour le décomposer en d'autres cas d'utilisation qui le composeront. Dans ce qui suit un diagramme de cas d'utilisation représentant cette décomposition.

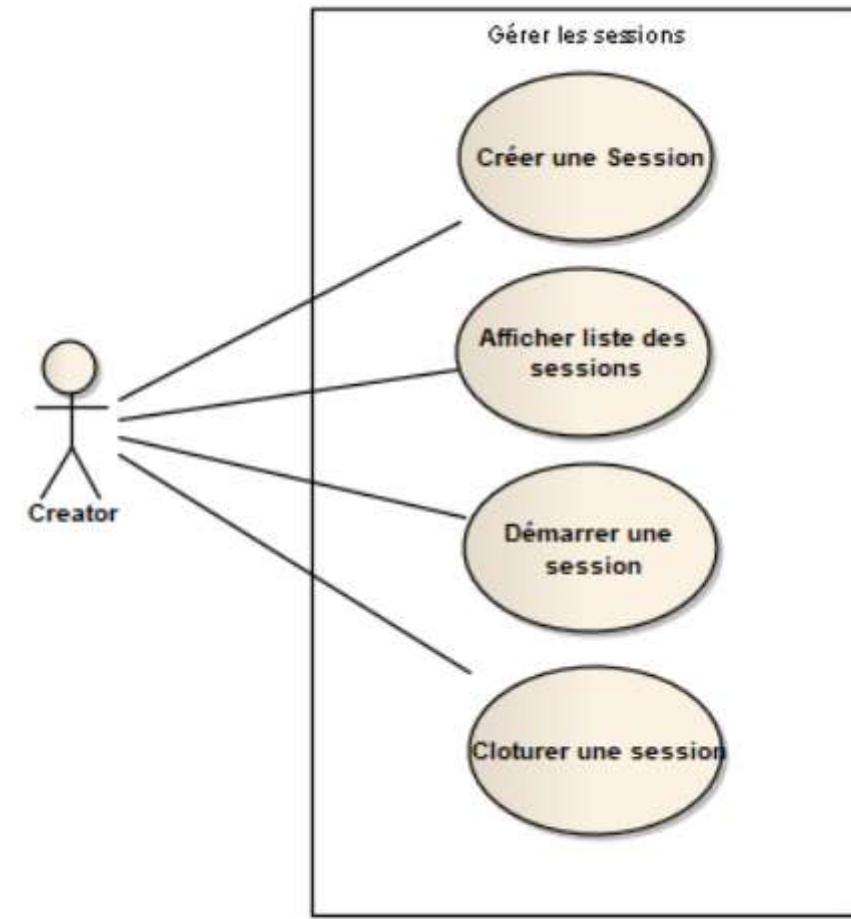


Figure 12 : Diagramme de cas d'utilisation « Gérer les sessions »

3-5-2 Diagramme d'activité de « Gérer les Sessions »

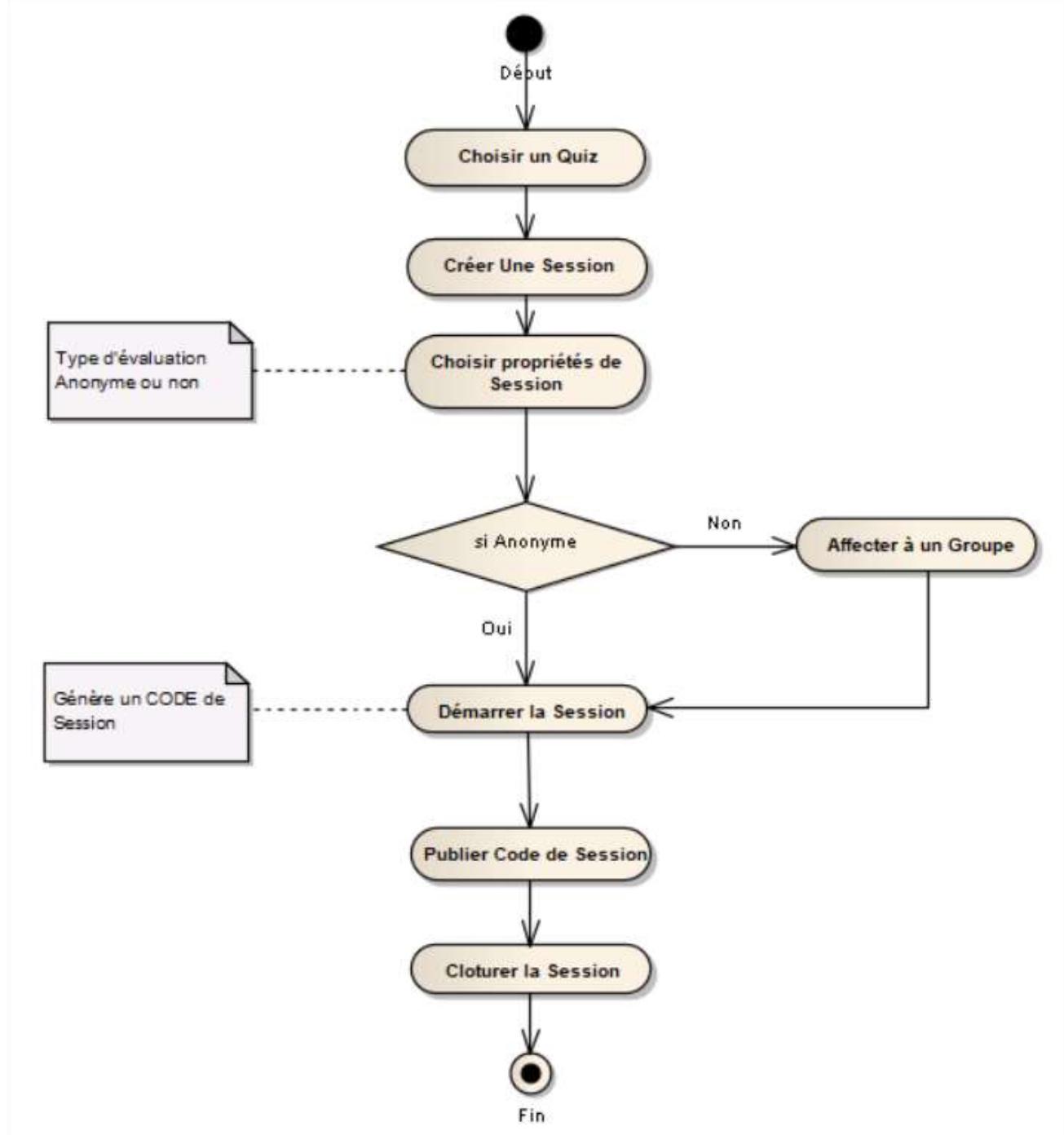


Figure 13 : Diagramme d'activités « Gérer les Sessions »

A titre d'exemple, prenons les cas d'utilisation « Créer une Session » et « Démarrer une Session » qui me paraissent les plus importants dans cette décomposition et essayons de les étudier en effectuant une description textuelle.

3-5-3 Description textuelle du cas d'utilisation « Crée une Session »

Tableau 7 : Description textuelle du cas d'utilisation « Crée une Session »

Titre	Créer une Session
Acteurs	Creator
But	Créer et sauvegarder une session en vue de la démarrer le moment voulu.
Pré condition	Le Creator doit être authentifié et le quiz sujet de la session doit être déjà créé.
Description	Ce cas d'utilisation doit permettre au Creator de créer une session de quiz, en saisissant ses informations.
Scénario nominal	<ol style="list-style-type: none"> 1. Creator demande d'afficher la liste de ses Quiz. 2. Le système affiche la liste des Quiz. 3. Creator demande de créer une nouvelle session de Quiz en cliquant sur le bouton «  » correspondant au quiz choisi. 4. Le système affiche l'interface de création de Session. 5. Creator saisi les propriétés de la session, à savoir, le type de l'évaluation (des prérequis, formative ou sommative), si la session nécessite ou non une authentification de l'apprenant et si le système doit renvoyer les bonnes réponses à l'apprenant à la fin du test ou non puis valide en cliquant sur le bouton « Crée la Session ». 6. Le système sauvegarde la session. 7. Le système affiche la liste des sessions dont celle qui vient d'être créée.

3-5-4 Description textuelle du cas d'utilisation « Démarrer une Session »

Tableau 8 : Description textuelle du cas d'utilisation « Démarrer une Session »

Titre	Démarrer une Session
Acteurs	Creator
But	Démarrer une session en vue de permettre aux apprenants de répondre aux questions du quiz concerné.
Pré condition	Le Creator doit être authentifié et la session doit être déjà créée.
Description	Ce cas d'utilisation doit permettre au Creator de démarrer une session de quiz afin de permettre aux apprenants de s'y connecter moyennant un code généré aléatoirement.
Scénario nominal	<ol style="list-style-type: none"> 1. Creator demande d'afficher la liste de ses Sessions. 2. Le système affiche la liste des Sessions. 3. Creator demande de démarrer une session de Quiz en cliquant sur le bouton «  » correspondant à la session choisie. 4. Le système rafraîchi la liste des sessions en y ajoutant un code aléatoire formé par 6 chiffres sur la ligne de la session choisie. 5. Creator communique le code généré aux apprenants pour qu'ils puissent commencer le test.

Après avoir étudié en profondeur chacun des cas d'utilisation, il m'est devenu possible d'élaborer le diagramme de classes.

3-6 Diagramme de classes

Même si j'ai utilisé une mixture de paradigme entre orienté objet et programmation fonctionnelle (JavaScript, TypeScript), un diagramme de classe permet d'avoir une vue globale sur les différentes entités que je vais utiliser dans mon application définis par leurs attributs respectifs et des relations entre-elles.

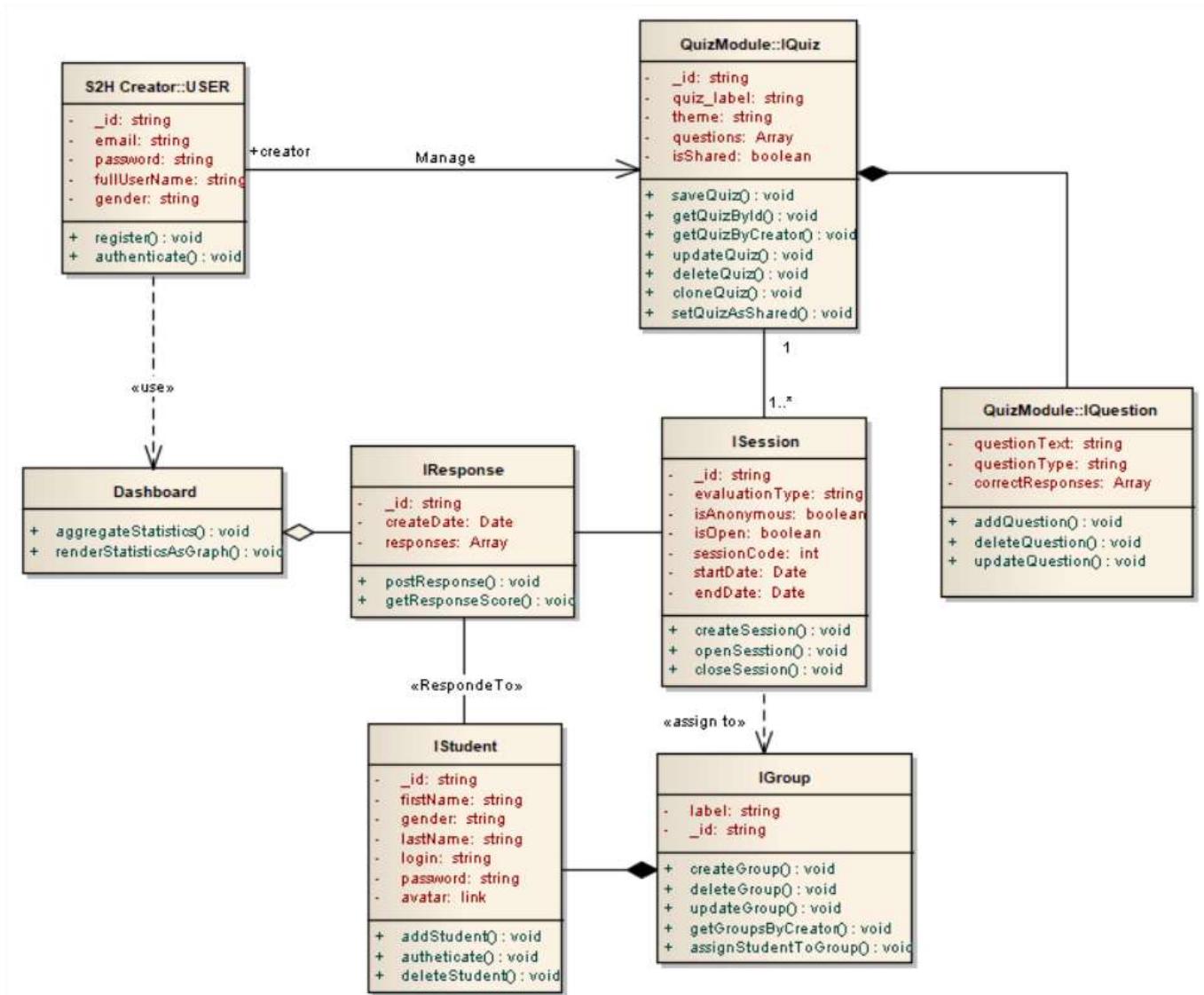


Figure 14 : Diagramme de classes

Essayons à présent d'effectuer une brève lecture de ce diagramme :

- Un utilisateur gère des Quiz (CRUD).
- Chaque Quiz est composé entre-autres par des questions.
- Un Quiz peut être attribué à une ou plusieurs sessions.
- Une session peut être attribuée à un groupe.
- Un groupe est formé par un ou plusieurs apprenants.

- Un Apprenant répond à une session dans ce qu'on appelle « Response ».
- L'ensemble des réponses des apprenants forment la base d'un dashboard.
- Un utilisateur consulte son dashboard.

3-7 Les Sprints

3-7-1 Sprint 1

3-7-1-1 Backlog du Sprint 1

Tableau 9 : Backlog du Sprint 1

Tâches à réaliser	Durée (jours)
Esquisse des interfaces utilisateur utilisant la technique Wireframe sketching	1
Scaffolding de l'application S2H Creator avec Angular	2
Scaffolding de l'application S2H API avec Node.js	2

3-7-1-2 Description du Sprint

Ce sprint consiste en la réalisation d'esquisses sur papier décrivant les différentes interfaces utilisateur, les montrer aux utilisateurs afin d'avoir leur approbation et éventuellement des suggestions puis entamer l'étape du scaffold avec Angular pour l'application S2H Creator et Node.js pour l'application S2H API.

3-7-1-3 Elaboration des esquisses papier

Pour élaborer ses esquisses j'ai utilisé la technique du Wireframe sketching permettant de créer des wireframes ou maquette fonctionnelle en français.

La maquette fonctionnelle est un schéma qui montre l'agencement des parties composant une page web. Elle permet donc la visualisation des zones de texte, l'emplacement des images, des vidéos, des liens, ainsi que des différents éléments graphiques.

En outre, pour faire une maquette fil de fer, autre appellation du wireframe, on a très peu recours aux images et aux couleurs. Et que ce soient avec des logiciels ou sur papier, le procédé reste le même.

Tout en faisant cette maquette, l'utilisateur final sera mis au centre de cette démarche.

Les wireframes sont bien utiles dans les projets web. Que ce soient lors de la refonte d'un site web ou de sa création, cet outil est performant. En effet, ils permettent de mettre à plat ses idées, de mieux communiquer avec les parties prenantes, et de faciliter le processus de création.

De mon côté, j'ai choisi d'élaborer mes wireframes sur papier afin de les montrer aux utilisateurs (Certains enseignants collègues) afin d'avoir leur approbation et de leur demander des suggestions.

Voici quelques unes des wireframes que j'ai montrés aux utilisateurs :

- Interface de la page d'atterrissement (Land Page) de l'application S2H Creator :

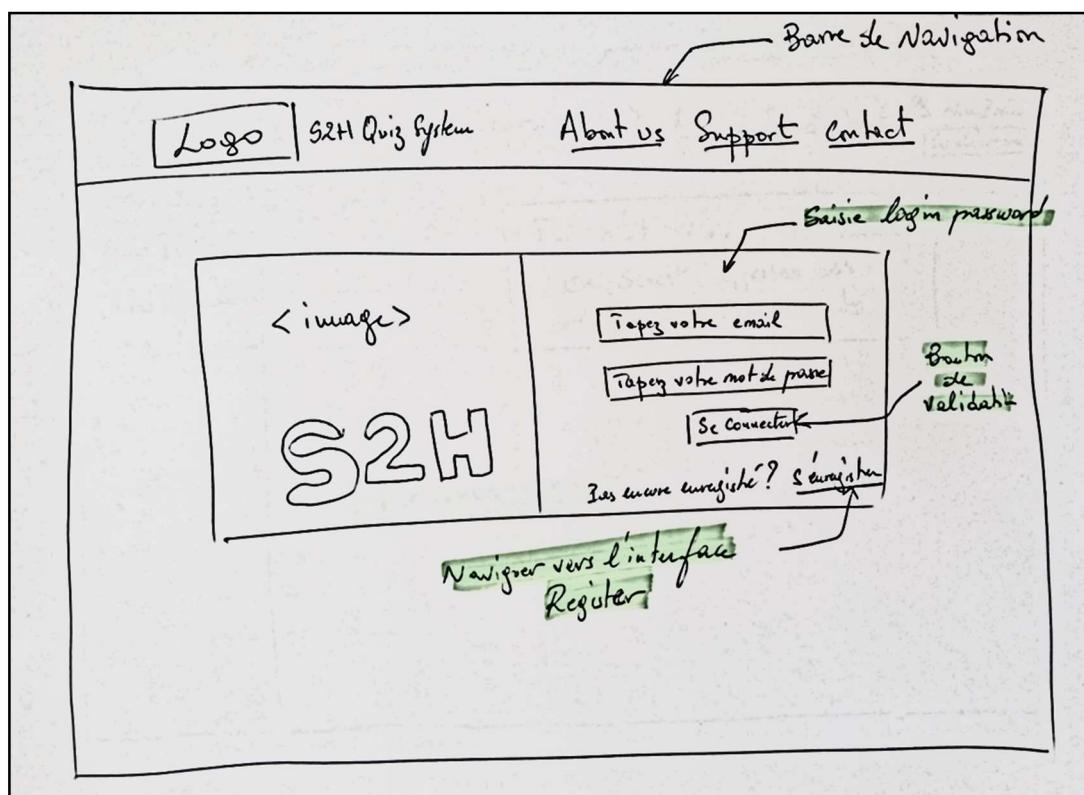


Figure 15 : Esquisse landpage

- Interface d'enregistrement d'un nouveau créateur :

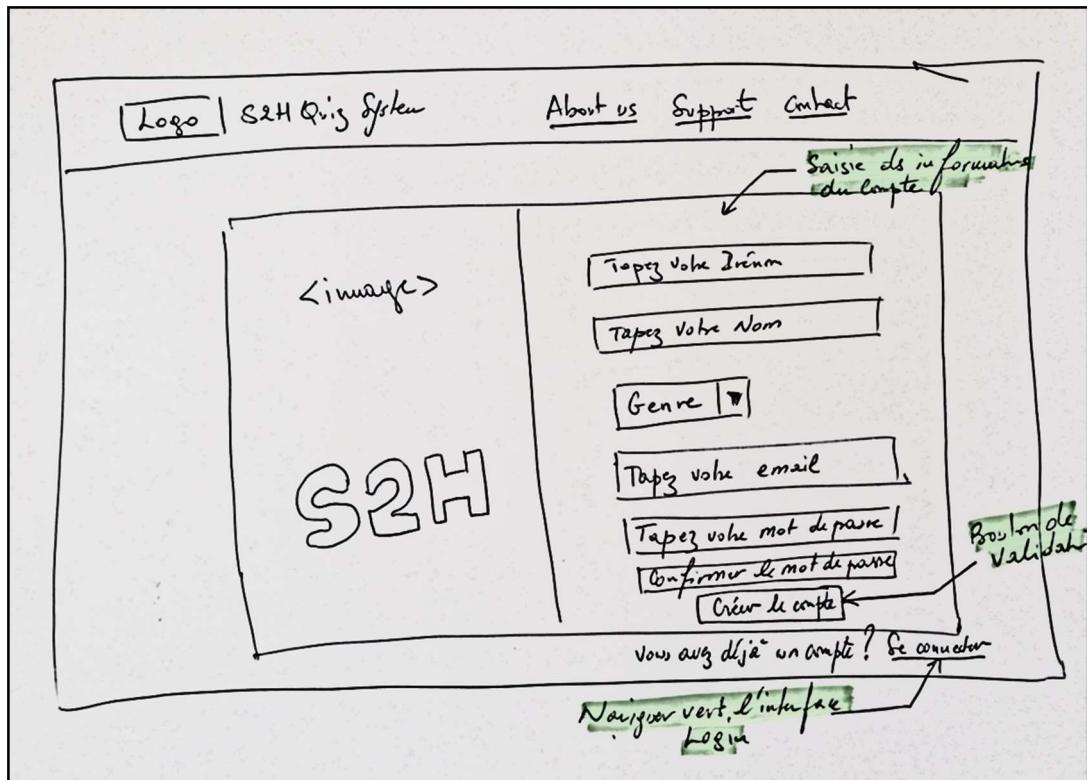


Figure 16 : Esquisse page d'enregistrement

- Interface de la page d'accueil d'un créateur :

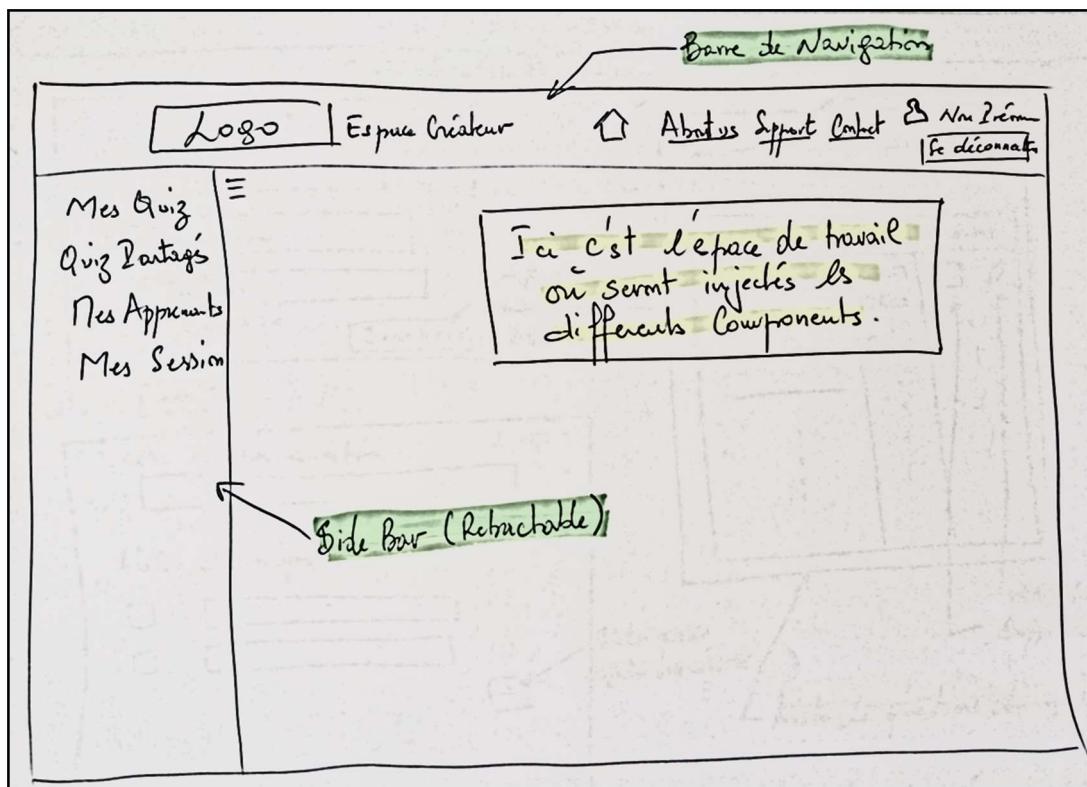


Figure 17 : Esquisse Landpage Createur

- Interface de création de Quiz :

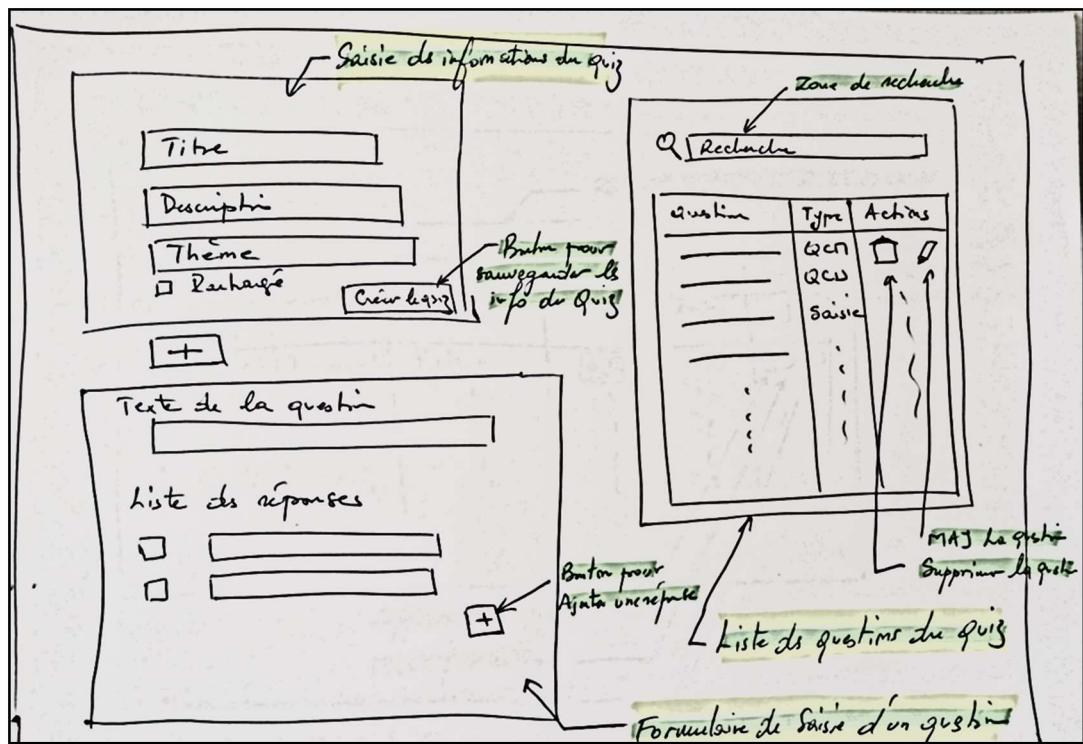


Figure 18 : Esquisse page de création de Quiz

- Interface de la liste des Quiz

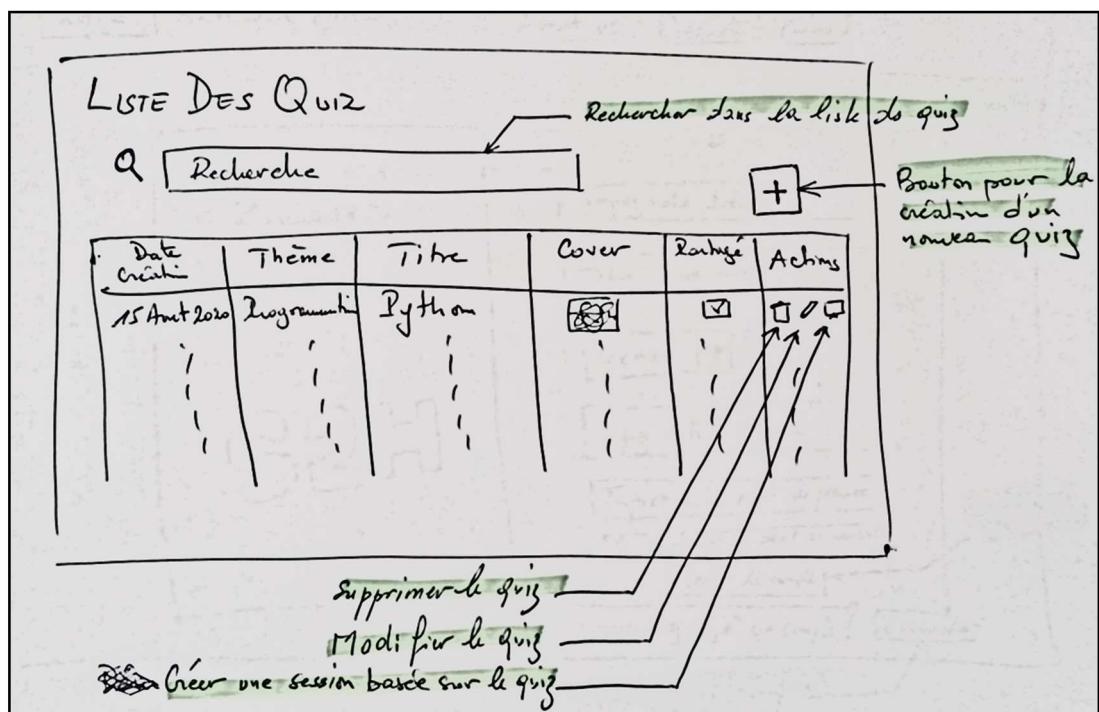


Figure 19 : Esquisse Page liste des Quiz

3-7-1-4 Scaffolding de l'application S2H Creator

Afin de scaffolder l'application S2H Creator, j'ai utilisé différentes commandes de « angular CLI », les voici :

- Pour créer l'application :

```
ng new s2h-creator
```

- Pour créer les différents modules :

```
ng generate module user
```

```
ng generate module creator
```

```
ng generate module student
```

```
ng generate module quiz
```

```
ng generate module session
```

```
ng generate module group
```

- Pour créer le module de routage :

```
ng generate module app-routing --flat --module=app
```

- Pour créer les différents services :

```
ng generate service user
```

```
ng generate service authentication
```

```
ng generate service student
```

```
ng generate service quiz
```

```
ng generate service session
```

```
ng generate service group
```

```
ng generate service question
```

- Pour créer les différents components prenons à titre d'exemples :

```
ng generate component land-page
```

```
ng generate component quiz-form
```

```
ng generate component session-form
```

```
ng generate component quiz-list
```

```
ng generate component question-list
```

3-7-1-5 Scaffolding de l'application S2H API

Il s'agit d'installer express-generator qui est un outil permettant de créer le squelette d'une application Node.js en entier, de l'utiliser puis installer les dépendances

- Pour installer express-generator :

```
npm install express-generator -g
```

- Création du squelette :

```
express s2h-api
```

- Pour installer les dépendances :

```
npm install
```

3-7-2 Sprint 2

3-7-2-1 Backlog du Sprint 2

Tableau 10 : Backlog du Sprint 2

ID	User Story	Tâches à réaliser	Durée (jours)
1	En tant qu'enseignant je veux avoir un espace de travail propre à moi.	Au niveau S2H Creator :	
		Implémenter « User Service »	3
		Implémenter « Authentication Service »	2
		Implémenter « Group Service »	1
		Implémenter « Student Service »	1
	En tant qu'enseignant je veux m'assurer que personne d'autre n'ai accès à mes informations.	Au niveau S2H API :	
		Implémenter « User Service » moyennant jwt (json web token).	3
		Implémenter « User Routes » (/authenticate, /register, /update:id...) comme contrôleur des requêtes entrantes.	2
		Implémenter « Group Service »	2
		Implémenter « Student Service »	1

Remarque :

A titre d'éclaircissement, la notion de service au niveau Front-end diffère un peu de celle au niveau Back-end.

En fait, un service au niveau Back-end un service peut être défini est représenté par une fonction qui accepte un objet mongoose schema et qui retourne un objet formé par des méthodes, tandis qu'au niveau Front un service est représenté par une classe dite injectable (implémentée par le dependency injection pattern).

3-7-3 Sprint 3

3-7-3-1 Backlog du Sprint 3

Tableau 11 : Backlog du Sprint 3

ID	User Story	Tâches à réaliser	Durée (jours)
8	En tant qu'enseignant je peux créer, modifier, supprimer et consulter mes Quiz.	Au niveau S2H Creator :	
9	En tant qu'enseignant je peux partager mes quiz avec d'autres enseignants.	Implémenter « Quiz Service »	8
10	En tant qu'enseignant je peux consulter les quiz partagés par d'autres enseignants et éventuellement les importer.	Implémenter « Quiz Form » et « Quiz List »	5
11	En tant qu'enseignant je peux ajouter, supprimer et modifier des questions dans des quiz et y déposer les bonnes réponses qui seront utilisées lors de l'évaluation des réponses des apprenants.	Au niveau S2H API : Implémenter « Quiz Service »	4
		Implémenter « Quiz Routes » (/quiz/add, /quiz/:id, /quiz/duplicate/:id, /quiz/update:id...) comme contrôleur des requêtes entrantes.	4

3-7-4 Sprint 4

3-7-4-1 Backlog du Sprint 4

Tableau 12 : Backlog du Sprint 4

ID	User Story	Tâches à réaliser	Durée (jours)
12	En tant qu'enseignant je peux créer et mettre en instance une session relativement à un de mes quiz.	Au niveau S2H Creator : Implémenter « Session Service »	3
13	En tant qu'enseignant je peux démarrer une session de quiz afin que les apprenants puissent se connecter et répondre aux questions.	Implémenter « Session Form » et « Session List »	5
14	En tant qu'enseignant je veux qu'un code généré aléatoirement soit attribué à une session récemment démarrée pour qu'elle soit accessible aux apprenants.	Au niveau S2H API :	
15	En tant qu'enseignant je peux choisir si une session soit accessible uniquement à un groupe donné ou qu'elle soit ouverte à tous ceux qui en ont le code d'accès.	Implémenter « Session Service »	2
16	En tant qu'enseignant je peux arrêter une session pour qu'il n'y soit plus de possibilité aux apprenants de poster leurs réponses ni aux nouveaux venus de se connecter à la session.	Implémenter « Session Routes » (/session/create, /session/start/:id, /session/close/:id, ...) comme contrôleur des requêtes entrantes.	4

3-7-5 Sprint 5

3-7-5-1 Backlog du Sprint 5

Tableau 13 : Backlog du Sprint 5

ID	User Story	Tâches à réaliser	Durée (jours)
17	En tant qu'enseignant je peux consulter les notes d'évaluation des apprenants dans une session donnée.	Au niveau S2H Quiz : Scaffold de l'application	1
		Implémenter « Response Service »	3
18	En tant qu'enseignant je peux consulter des statistiques de mon archive de sessions.	Implémenter « Session Dashboard » et « Statistics Dashboard »	8
		Au niveau S2H API :	
19	En tant qu'apprenant je peux répondre à un quiz dont la session a été démarrée.	Implémenter « Response Service »	4
20	En tant qu'apprenant je peux connaître ma note d'évaluation après avoir posté mes réponses et éventuellement connaître les bonnes réponses si mon enseignant l'a permis.	Implémenter « Response Routes » (/response/student/session:id, ...) comme contrôleur des requêtes entrantes.	4

Conclusion

Dans ce chapitre j'ai essayé de présenter les différentes étapes de réalisation allant de la planification à la réalisation pratique du projet.

Après avoir défini le backlog du produit, j'ai regroupé chacune des User Stories en Features (Fonctionnalités). Ces fonctionnalités sont la base de la formation des 5 sprints qui ont formé le cycle de développement de l'application.

Chapitre 4 Déploiement

Introduction

Dans ce chapitre je vais essayer de présenter la méthode et les outils avec lesquelles j'ai déployé mon application afin qu'elle soit disponible aux utilisateurs.

4-1 DevOps

DevOps est un ensemble de bonnes pratiques et de changements culturels qui permettent aux équipes chargées de développement, de contrôle de qualité et de l'exploitation de mieux répondre aux besoins des clients. Ce mode d'organisation facilite la communication, la collaboration et l'intégration entre les équipes afin de suivre l'évolution constante des besoins de l'entreprise.

DevOps permet à l'entreprise de rassembler les principales parties prenantes (l'entreprise, les applications et les opérations) en se concentrant sur la collaboration, l'automatisation et la surveillance afin de publier plus rapidement des applications de meilleure qualité.

Dans l'approche DevOps, le livrable sera créé d'une manière incrémentale et itérative. Pour cela, Devops présente un ensemble logique des phases à exécuter.

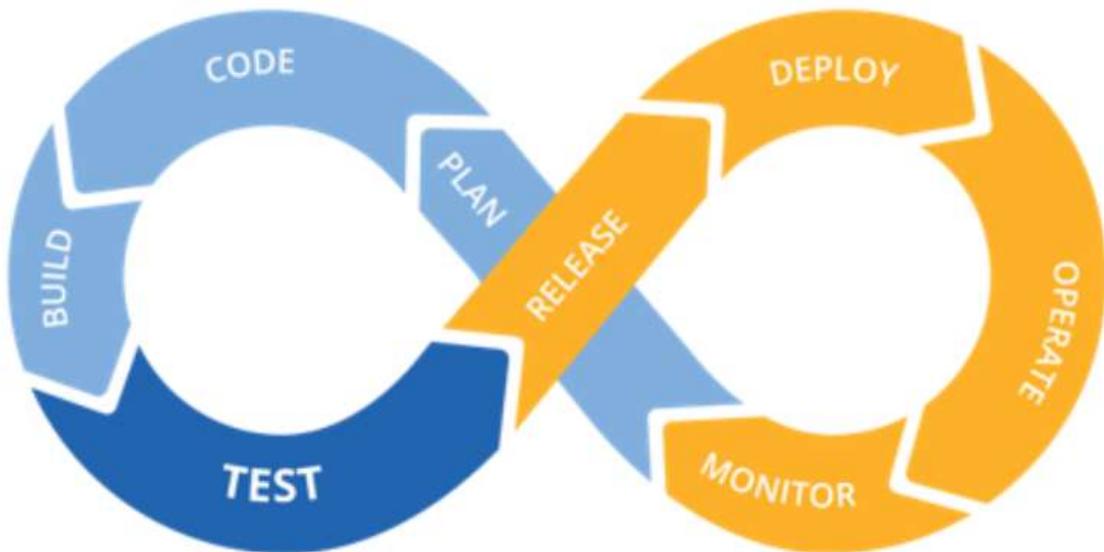


Figure 20 : Les phases du cycle DevOps

Le Tableau suivant présente une explication complète de chaque phase de cycle DevOps :

Tableau 14 : Les phases du cycle DevOps

Phase	But
PLAN	Son but est de planifier le type d'application développée. Faire le portrait approximatif du processus de développement
CODE	Son but est de coder l'application selon les besoins du client avec le plan fait dans la première étape
BUILD	C'est un processus, dont l'objectif est de générer l'application en effectuant l'intégration des différents codes réalisés à l'étape précédente
TEST	C'est le cœur de l'application. Son but est de tester l'application construite jusqu'à présent et reconstruit l'application si nécessaire (Unit testing, Integration testing, End to end testing)
RELEASE ou processus de publication	Son objectif est de déterminer si la nouvelle version du livrable répond aux exigences.
DEPLOY ou processus de déploiement	Son but est de déployer le nouveau livrable dans l'environnement de production d'une manière automatisée.
OPERATE	C'est un processus d'exploitation, son objectif est de maintenir une conformité entre le service opérationnel, les fonctionnalités ainsi que la qualité, afin d'assurer la continuité de l'activité
MONITOR	C'est un processus de surveillance dont le but est de contrôler le bon fonctionnement du service livré.

Donc, DevOps propose une démarche opérationnelle dans laquelle l'intégration et le déploiement continu seront assurés.

Les avantages de l'adoption de la philosophie DevOps sont les suivantes :

- Livraison continue et rapide d'un produit fiable.
- Adaptation rapide aux besoins des clients.

- Intégration continue.
- Automatisation du provisionnement et du déploiement. Élimination des processus manuels.
- Collaboration entre les équipes.
- Journalisation

4-2 IBM Cloud

La plateforme IBM Cloud combine PaaS (Platform As A Service) avec IaaS (Infrastructure As A Service) pour offrir une expérience intégrée. La plate-forme s'adapte et prend en charge à la fois les petites équipes et organisations de développement et les grandes entreprises. Globalement déployée dans les centres de données du monde entier, la solution construite sur IBM Cloud tourne rapidement et fonctionne de manière fiable dans un environnement testé et pris en charge dans lequel on peut avoir confiance. Il s'agit d'une plate-forme de cloud computing ouverte qui combine PaaS avec IaaS, et comprend un catalogue de divers services cloud, qui peuvent être utilisés pour créer et déployer rapidement des applications ou une infrastructure d'entreprise.

- PaaS permet aux développeurs d'accéder aux logiciels IBM pour l'intégration, la sécurité, les transactions et d'autres fonctions clés. Les types d'applications peuvent aller du Web, du mobile, du Big Data et des appareils intelligents à l'Internet des objets (IoT).
- IaaS donne aux développeurs un contrôle précis sur l'infrastructure sur laquelle leurs applications sont déployées. Les développeurs peuvent déployer des serveurs bare metal, des serveurs virtuels, des conteneurs et un stockage cloud hautes performances dans les centres de données IBM Cloud du monde entier.

La plateforme IBM Cloud est composée de plusieurs composants qui fonctionnent ensemble pour offrir une expérience cloud cohérente et fiable :

- Un catalogue composé de centaines d'offres IBM Cloud

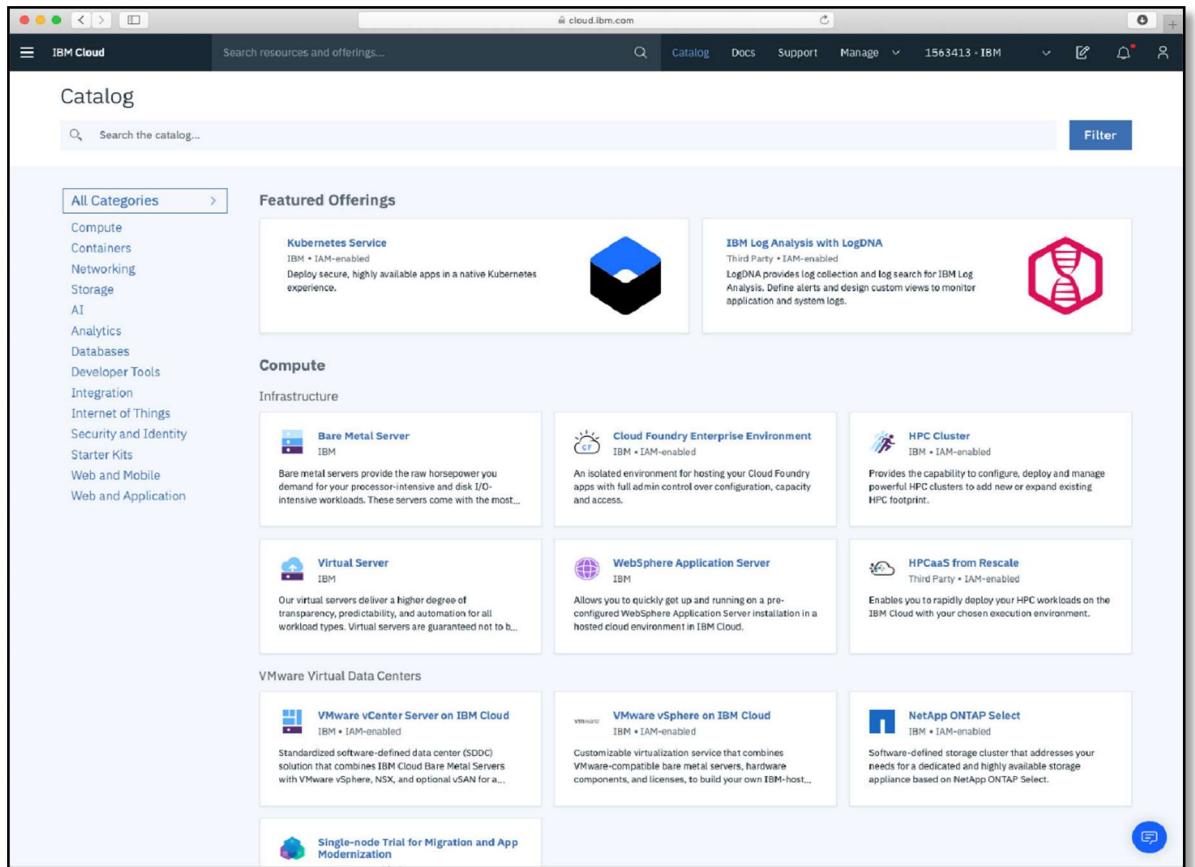


Figure 21 : Catalogue IBM Cloud

- Une console robuste qui sert d'interface pour la création, l'affichage et la gestion des ressources cloud (IBM Cloud CLI).
- Un composant IAM qui authentifie en toute sécurité les utilisateurs pour les services de la plateforme et contrôle l'accès aux ressources de manière cohérente dans IBM Cloud.
- Un système de gestion de compte et de facturation qui fournit une utilisation exacte des plans tarifaires et une protection sécurisée contre la fraude par carte de crédit.

En ce qui m'intéresse dans mon projet c'est Cloud Foundry offert par IBM Cloud comme PaaS qui fournit un Runtime pour mon application.

4-2-1 Qu'est-ce que Cloud Foundry

Cloud Foundry est la plate-forme en tant que service (PaaS) standard de l'industrie qui garantit le déploiement le plus rapide, le plus simple et le plus fiable des applications.

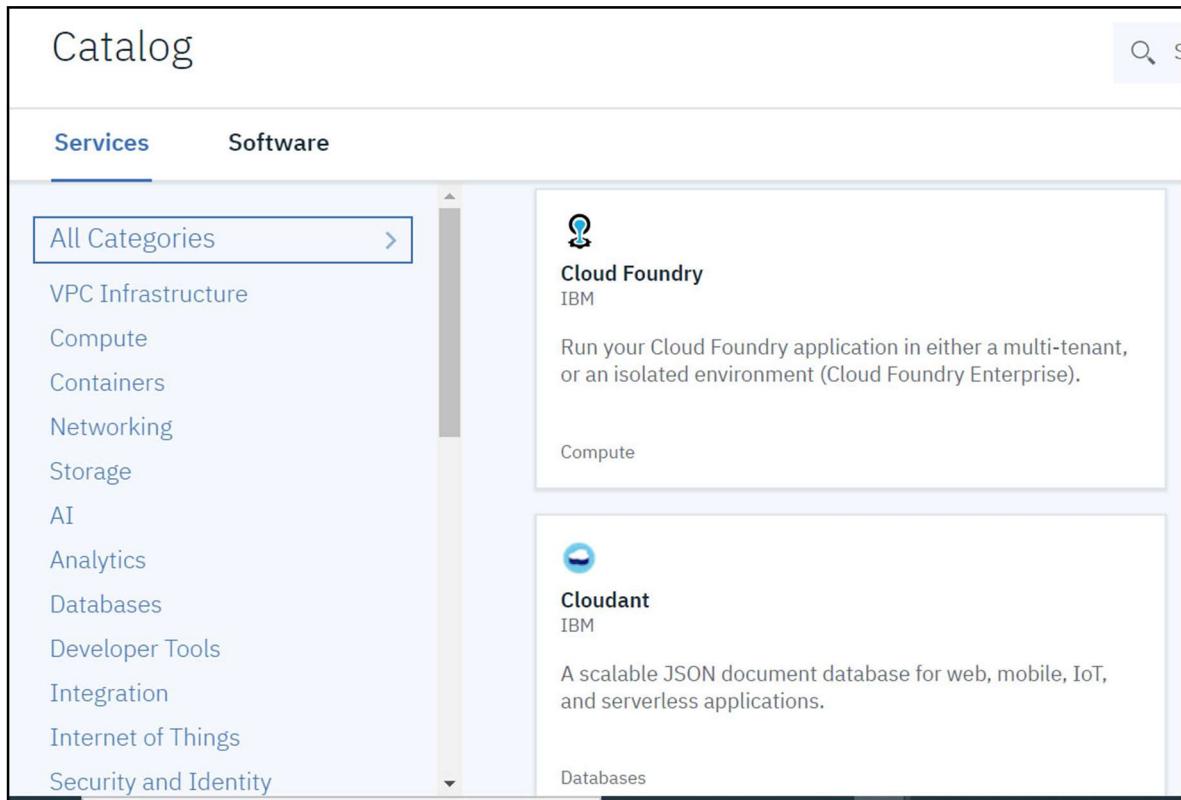


Figure 22 : Cloud Foundry dans le catalogue IBM Cloud

Je peux ici citer les principaux avantages de Cloud Foundry qui peuvent être utiles pour le projet :

- Choisir un langage : IBM Cloud Foundry inclut des Runtime pour Java, Node.js, PHP, Python, Ruby, Swift et Go. Le langage de mon application est Javascript et est déployée avec NGINX pour S2H Creator et S2H Quiz et avec Node.js pour S2H API.
- Combinés aux services DevOps, les environnements d'exécution des applications permettent un pipeline de livraison qui automatise une grande partie du processus de développement itératif.
- Tolérance aux pannes : les Runtimes facilitent le développement d'applications en tant que processus sans état (Stateless) qui démarrent et s'arrêtent rapidement, se répliquent en cas de défaillance d'une instance et se dupliquent si des performances soutenues ou accrues l'exigent.
- Gestion automatique de la santé : les applications en panne redémarrent automatiquement.

- Routage automatique : des « routes » accessibles par Internet sont automatiquement créés pour les applications.
- Mise à l'échelle automatique : le service « Auto-Scaling for IBM Cloud » permet d'augmenter ou de réduire automatiquement la capacité de calcul de l'application pour s'adapter rapidement aux besoins de chargement dynamique.

Vu ces avantages, j'ai opté pour IBM Cloud pour le déploiement de mon application.

J'ai créé trois applications Cloud Foundry permettant chacune de déployer une des applications de ma solution, à savoir, S2H Creator, S2H Quiz et S2H API.

J'ai par la suite créé pour chacune des applications Cloud Foundry une chaîne d'outils pour assurer les pratiques CI / CD.

Ici c'est une capture d'écran de mon tableau de bord IBM Cloud représentant la liste de mes ressources :

The screenshot shows the IBM Cloud dashboard with the following details:

- Header:** IBM Cloud, Rechercher des ressources et des offres..., Catalogue, Documentation, Support, Gérer, Med Khalil NAS...
- Left sidebar:** Tableau de bord, Récapitulatif des ressources (7 Ressources), Applications Cloud Foundry (3), Services (1), Developer tools (3).
- Middle section:**
 - Récapitulatif des ressources:** Shows 7 resources, with a green box highlighting the 'Applications Cloud Foundry' section.
 - Maintenance planifiée:** Shows 'Ciel dégagé ! Vous pouvez afficher ici vos événements de maintenance planifiés.'
 - Pour vous:** Watson Studio offer a suite of tools and an environment for data specialists, developers, and experts.
 - Actualités:** Optimize Cloud Costs with IBM Cloud for VMware Solutions, HashiCorp Enterprise Vault Plugins for IBM Cloud, Archive: Public Cloud Industry and Compliance.
 - Cas de demande de service récents:** Tout afficher.
 - Accès utilisateur:** Gérer les utilisateurs, Entrer des adresses électroniques ci-dessous pour accéder directement à la configuration de l'invitation d'utilisateur.
 - Statut d'IBM Cloud:** Tout afficher.

Mes ressources

Figure 23 : Mes ressources dans IBM Cloud

La capture d'écran suivante représente la liste détaillée des ressources :

- Les trois applications Cloud Foundry.
- Les trois chaînes d'outils des trois applications.

- Le service nécessaire aux chaînes d'outils d'assurer la livraison continue (Continuous Delivery).

The screenshot shows the IBM Cloud resource list interface. At the top, there's a search bar and navigation links for Catalogue, Documentation, Support, Gérer, and Med Khalil NAS... On the right, there are icons for creating a resource, filtering, and feedback.

Liste de ressources

Nom	Groupe	Emplacement	Offre	Statut	Etiquettes
s2h-API	etudiant.nasrallah.mdkhelil@uvt.tn / dev	Londres	SDK for Node.js™	Démarré	-
s2h-creator	etudiant.nasrallah.mdkhelil@uvt.tn / dev	Londres	SDK for Node.js™	Démarré	-
s2h-quiz	etudiant.nasrallah.mdkhelil@uvt.tn / dev	Londres	SDK for Node.js™	Démarré	-
Continuous Delivery-z8	Default	Londres	Continuous Delivery	Actif	-
s2h-API	Default	Londres	Toolchain	-	-
s2h-creator	Default	Londres	Toolchain	-	-
s2h-quiz	Default	Londres	Toolchain	-	-

Annotations:

- Le service assurant CI / CD** points to the "Continuous Delivery-z8" row.
- Les trois applications sur Cloud Foundry** points to the first three rows (s2h-API, s2h-creator, s2h-quiz).
- Les trois chaînes d'outils** points to the last three rows (s2h-API, s2h-creator, s2h-quiz) under the "Developer tools" section.

Figure 24 : Détails des ressources dans IBM Cloud

Prenons par exemple la chaîne d'outil S2H Creator qui est représentée dans la capture d'écran suivante :

The screenshot shows the details of the S2h-creator tool chain. At the top, there's a backlink to 'Chaines d'outils /' and a 'Add tags' button. On the right, there are buttons for 'Détails', 'Actions...', and 'Ajouter un outil'.

Présentation

Penser

- Connexions
- Gérer
- Issues** s2h-Creator
- ✓ Configuré

Coder

- GitHub** s2h-Creator
- ✓ Configuré

Distribuer

- Delivery Pipeline** s2h-creator
- ✓ Réussite

Environnement

- Eclipse Orion Web IDE**
- ✓ Configuré

Figure 25 : La chaîne d'outils S2H Creator

Elle comprend entre-autre la configuration du lien avec GitHub (que j'expliquerai plus loin dans ce chapitre) afin d'importer l'application à partir de mon dépôt GitHub à chaque fois que mon application est modifiée. Le Delivery Pipeline est déclenché à chaque PUSH dans GitHub afin d'assurer CI / CD.

La Delivery Pipeline comporte deux étapes :

- Build Stage : C'est l'étape de la construction de l'application qui vient d'être téléchargée de GitHub.
- Deploy Stage : C'est l'étape de déploiement de l'application Cloud Foundry pour qu'elle soit disponible aux utilisateurs.

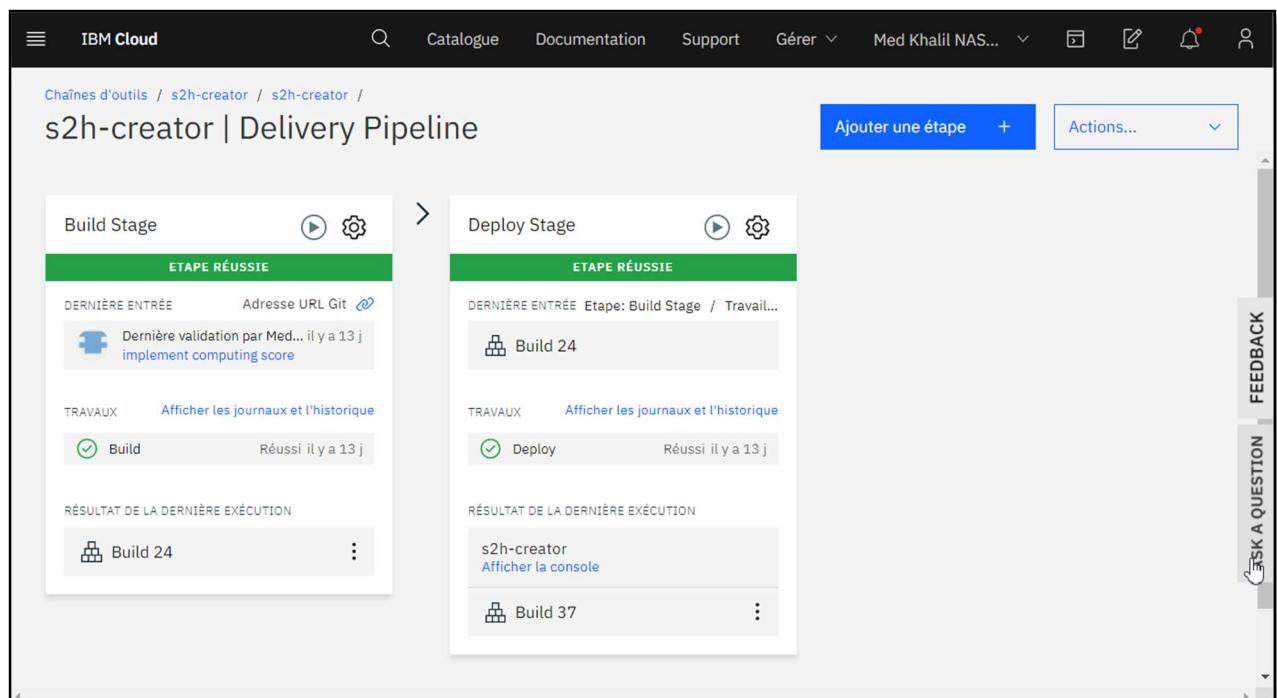


Figure 26 : Delivery Pipe Line de S2H Creator

4-3 GitHub

On ne peut pas faire du CI / CD sans un système de contrôle de versions distant qui assure une base de code (Code Base) traqué et validé pour provisionner le pipeline dans IBM Cloud. Pour mon cas, j'ai opté pour Git comme outil de gestion de versions et GitHub comme dépôt distant.

4-3-1 Qu'est-ce que Git ?

Git est un logiciel de versioning créé en 2005 par Linus Torvalds, le créateur de Linux.

Un logiciel de versioning, ou logiciel de gestion de versions est un logiciel qui permet de conserver un historique des modifications effectuées sur un projet afin de pouvoir rapidement identifier les changements effectuées et de revenir à une ancienne version en cas de problème.

Les logiciels de gestion de versions sont quasiment incontournables aujourd'hui car ils facilitent grandement la gestion de projets et car ils permettent de travailler en équipe de manière beaucoup plus efficace.

Parmi les logiciels de gestion de versions, Git est le leader incontesté et il est donc indispensable pour tout développeur de savoir utiliser Git.

4-3-2 Qu'est-ce que GitHub ?

Dans le langage des systèmes de gestion de version, la copie de l'intégralité des fichiers d'un projet et de leur version située sur le serveur central est appelé un dépôt. Git appelle également cela "repository" ou "repo" en abrégé.

GitHub est un service en ligne qui permet d'héberger des dépôts ou repo Git. C'est le plus grand hébergeur de dépôts Git du monde.

Une grande partie des dépôts hébergés sur GitHub sont publics, ce qui signifie que n'importe qui peut télécharger le code de ces dépôts et contribuer à leur développement en proposant de nouvelles fonctionnalités (Open Source).

Pour récapituler, et afin d'être bien clair sur ce point : Git est un logiciel de gestion de version tandis que GitHub est un service en ligne d'hébergement de dépôts Git qui fait office de serveur central pour ces dépôts.

4-3-3 Mon projet dans GitHub

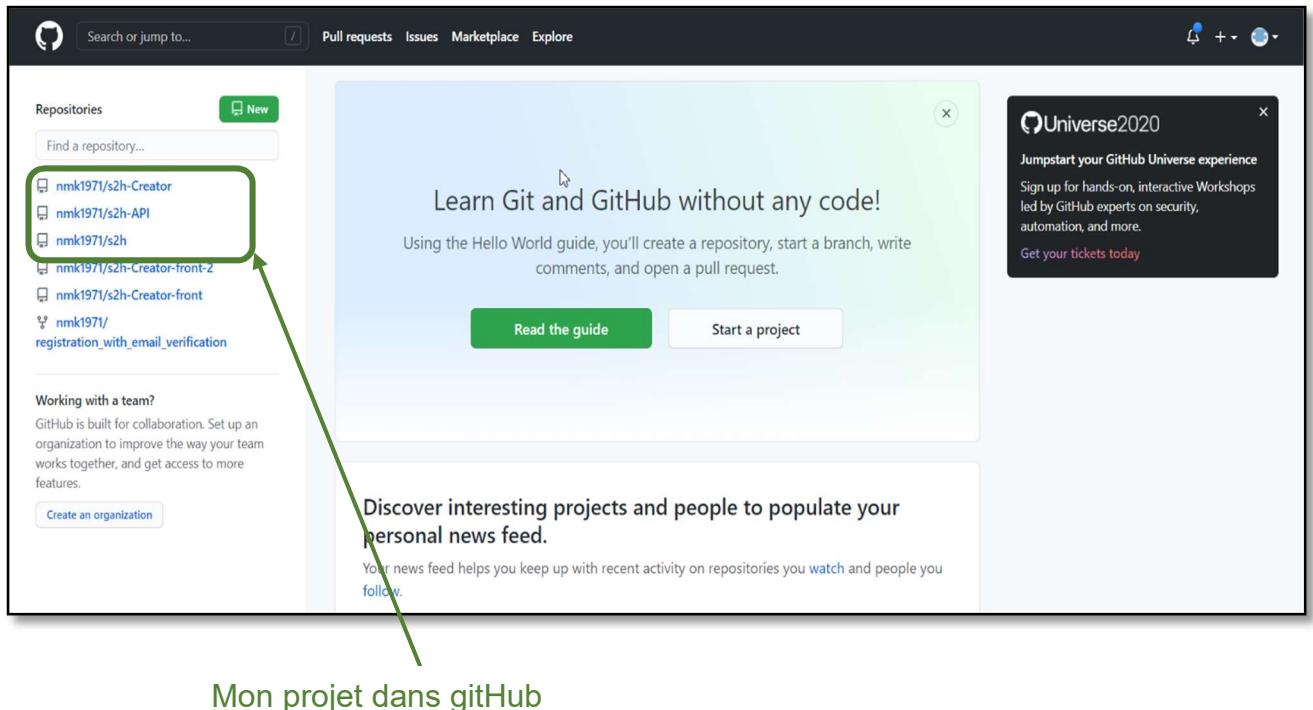


Figure 27 : Le projet dans GitHub

Conclusion

Dans ce chapitre j'ai présenté DevOps qui est un ensemble de bonnes pratiques assurant la continuité et l'automatisation du cycle de vie de mon application, IBM Cloud qui est la plateforme de déploiement ainsi que GitHub qui sert de système de gestion de versions ainsi qu'un intermédiaire entre mon ordinateur personnel est IBM Cloud.

Dès que la commande « git push » est tapée sur mon terminal, tout le reste du processus est automatique et quelques minutes après l'utilisateur aura accès à la nouvelle version via son navigateur.

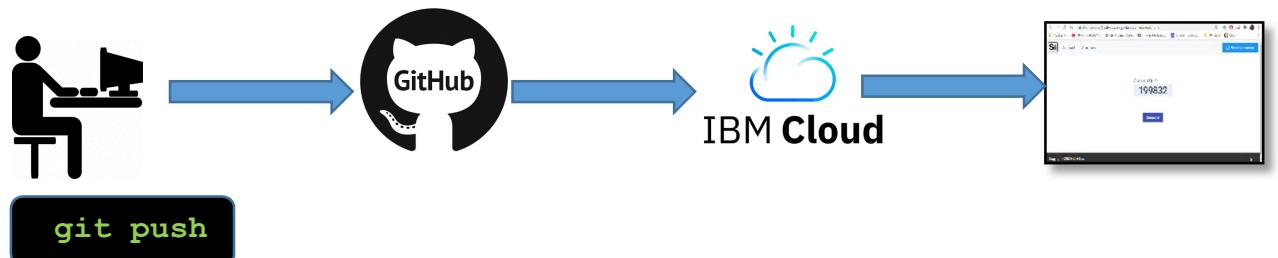


Figure 28 : CI / CD

Chapitre 5 L'application

Introduction

Dans ce chapitre je vais présenter des captures d'écran de chacune des applications en décrivant sa fonctionnalité.

5-1 S2H Creator

L'application S2H Creator permettant à un enseignant de créer ses Quiz, de gérer ses groupes d'apprenant, de gérer ses sessions de Quiz ainsi que de consulter les résultats des tests des apprenant est disponible sur l'URL <https://s2h-creator.eu-gb.cf.appdomain.cloud>

Dans la suite de cette partie je vais décrire quelques-uns des interfaces utilisateur.

5-1-1 Page de démarrage (Land Page)

A chaque fois qu'un internaute tente d'accéder à l'application et qu'il ne s'est pas authentifié dans une session précédente, il tombera sur la page Login lui permettant de se connecter à son espace de travail.

Dans le cas où l'internaute ne dispose déjà pas de compte, il pourra s'enregistrer.

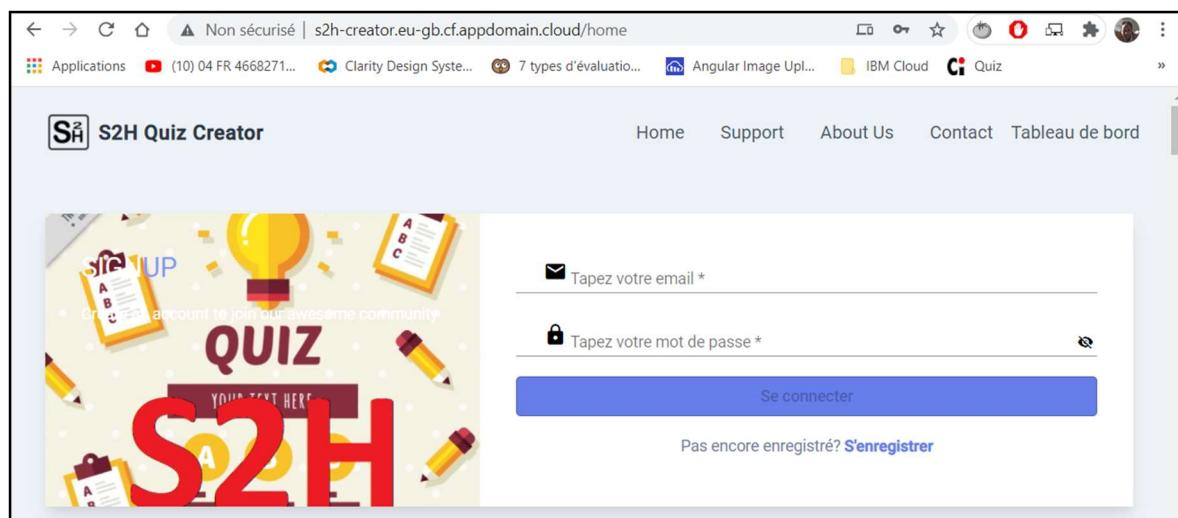


Figure 29 : Page d'accueil

5-1-2 Page d'accueil du Créateur

Une fois authentifié, l'utilisateur aura accès à son espace de travail formé essentiellement par deux barres de navigation dont l'une est rétractable et permettant d'accéder aux différentes fonctionnalités de l'application

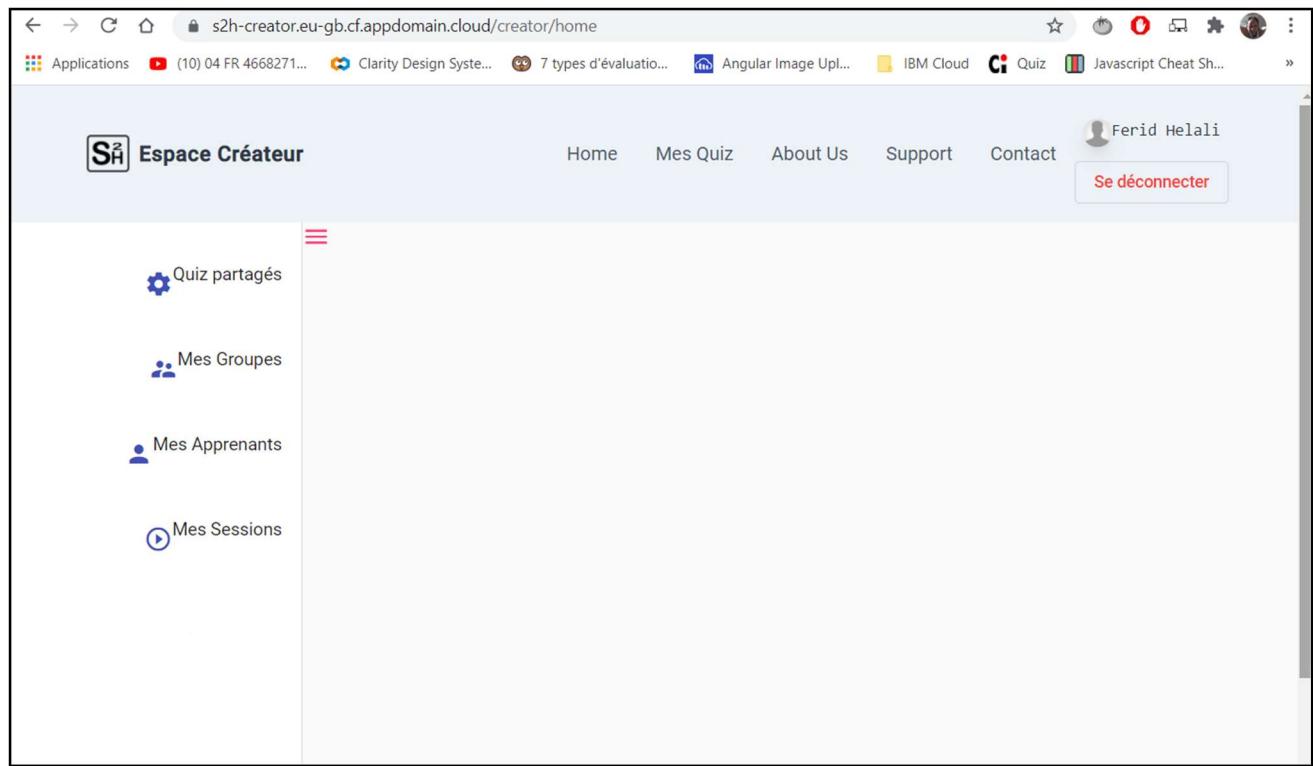


Figure 30 : Page d'accueil du Créateur

5-1-3 Liste des Quiz

En accédant au lien « Mes Quiz », l'utilisateur est redirigé vers la page représentant la liste des quiz et un bouton permettant d'ajouter un nouveau Quiz.

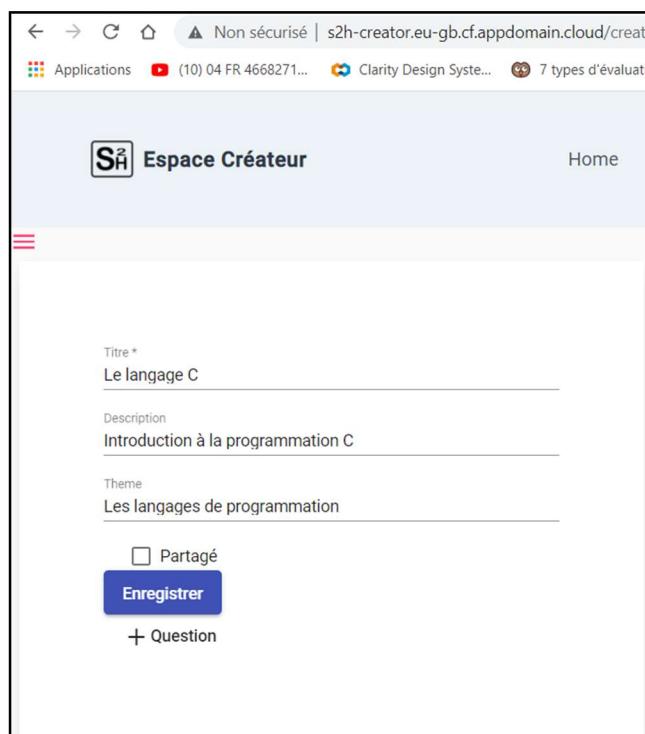
Chaque ligne du tableau contiendra les informations d'un quiz et un ensemble d'actions que l'utilisateur pourra entreprendre (Supprimer le Quiz, Modifier le Quiz ou créer une session basée sur le Quiz).

Quiz List						
Date de création	Thème	Title	Cover	Shared	Actions	
14 octobre 2020	Programmation	Python Le langage de Programmation Python		false		
14 septembre 2020	Système	Les Systèmes d'exploitation		true		
Items per page: 5 1 - 2 of 2 < >						

Figure 31 : Liste des Quiz

5-1-4 Création de Quiz

En choisissant d'ajouter un nouveau Quiz, l'utilisateur tombera sur la page suivante :



Titre *
Le langage C

Description
Introduction à la programmation C

Theme
Les langages de programmation

Partagé

Enregistrer

+ Question

Liste des Questions (2)

1- Choisir les instructions C	QCM	
2- C est un langage interprété	QCU	

Figure 32 : Création de Quiz

A gauche un formulaire permettant de saisir les informations d'un Quiz.

A droite une liste représentant la liste des questions constituant le Quiz.

En ajoutant une nouvelle question, un formulaire s'affichera en bas de la page permettant de saisir les informations d'une question.

Le formulaire permet la saisie du texte de la question ainsi que les réponses. Dans le cas d'une QCM par exemple, l'utilisateur devra saisir toutes les réponses proposées et d'indiquer celles qui sont bonnes.

QUESTION

Texte de la question
Parmi les suivantes, lesquelles ne sont pas des instructions C?

Texte de la Réponse
writeln ☒

Texte de la Réponse
scanf ☒

Texte de la Réponse
repeat ☒

Texte de la Réponse
while ☒

Figure 33 : Formulaire d'ajout d'une question

5-1-5 Créer une Session

A partir de la liste des Quiz, et si l'utilisateur choisi de démarrer une session basée sur un Quiz, il sera redirigé vers la page de création de session.

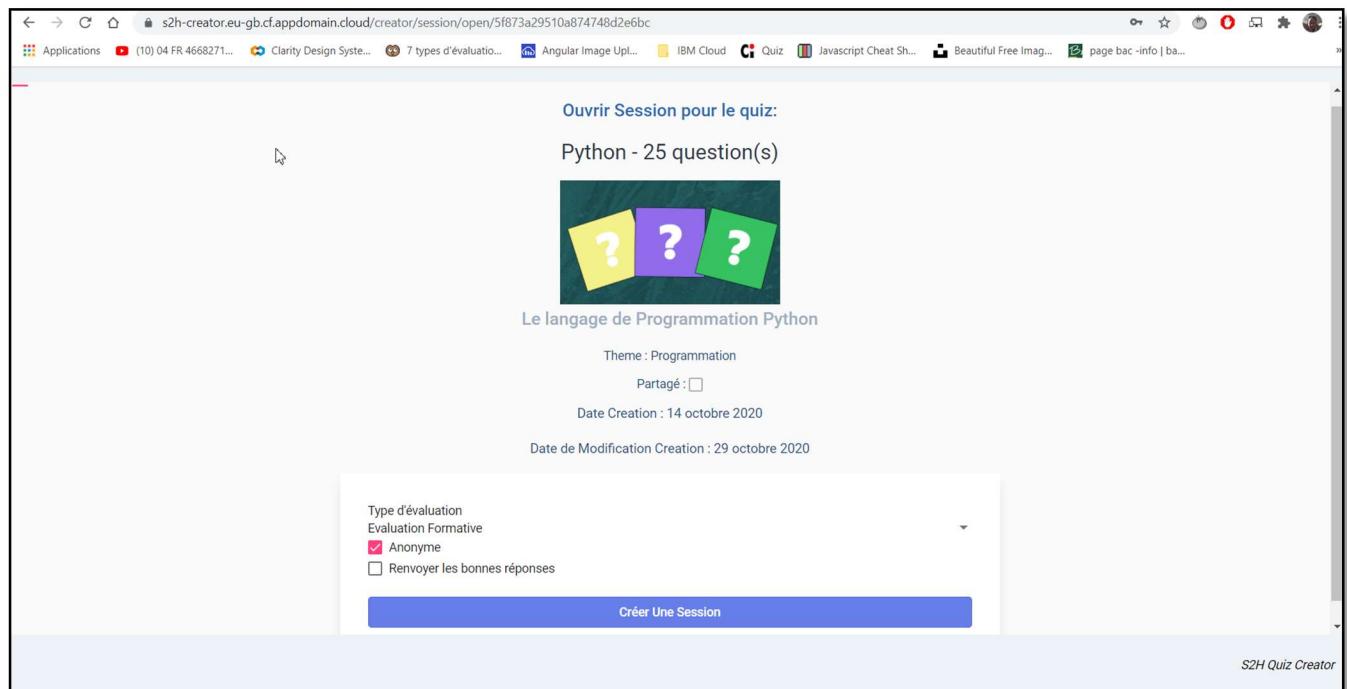


Figure 34 : Crédit de Session

Après avoir créé la session, l'utilisateur sera automatiquement redirigé vers la liste de ses sessions.

5-1-6 Liste des sessions

The screenshot shows a web browser window with the URL <https://s2h-creator.eu-gb.cf.appdomain.cloud/creator/session/list>. The page title is "Espace Créateur". On the left, there is a sidebar with links: "Quiz partagés", "Mes Groupes", "Mes Apprenants", "Mes Sessions", and "Customers". The main content area is titled "Liste des Sessions" and contains a table with the following data:

Créée le	Quiz	Anonyme	Code	Groupe	Actions
13 novembre 2020 à 22:49	Python i	<input checked="" type="checkbox"/>			(d)
8 novembre 2020 à 09:47	Les Systèmes d'exploitation i	<input checked="" type="checkbox"/>	652701		(r)
8 novembre 2020 à 09:47	Les Systèmes d'exploitation i	<input checked="" type="checkbox"/>	728728		(v)
8 novembre 2020 à 09:47	Les Systèmes d'exploitation i	<input type="checkbox"/>	403286	3ScExp Lycée Mahdia	(v)
7 novembre 2020 à 17:06	Les Systèmes d'exploitation i	<input type="checkbox"/>	556229	3ScExp Lycée Mahdia	(v)
29 octobre 2020 à 12:21	Les Systèmes d'exploitation i	<input type="checkbox"/>	757714	3ScExp Lycée Mahdia	(r)
29 octobre 2020 à 12:21	Les Systèmes d'exploitation i	<input checked="" type="checkbox"/>	892771		(v)
29 octobre 2020 à 12:21	Les Systèmes d'exploitation i	<input type="checkbox"/>	926322	3ScExp Lycée Mahdia	(v)
29 octobre 2020 à 12:21	Les Systèmes d'exploitation i	<input checked="" type="checkbox"/>	790532		(r)

Figure 35 : Liste des Sessions

Cette interface comprend la liste des sessions que l'utilisateur a créées. Chacune passe par 3 états :

- Session créée : C'est une session en instance attendant qu'elle soit démarrée en cliquant sur le bouton «  ».
- Session démarrée : C'est une session qui a été démarrée et qui attend qu'elle soit clôturée en cliquant sur le bouton «  ».
- Session Clôturée : C'est une session qui a été clôturée et est marquée par la marque «  » et il n'y aura plus d'actions à entreprendre pour cette session.

Lors du passage de la session de l'état créée à l'état démarrée, un code aléatoire est généré automatiquement afin d'être communiqué aux apprenants pour qu'ils puissent s'y connecter.

5-2 S2H Quiz

L’application S2H Quiz permettant à un apprenant de répondre à un Quiz est disponible sur l’URL <https://s2h-quiz.eu-gb.cf.appdomain.cloud>

Dans la suite de cette partie je vais décrire quelques-uns des interfaces utilisateur.

5-2-1 Page d’accueil (Land Page)

Dès qu’un apprenant accède à l’URL de l’application, il tombera sur une page demandant de saisir le code de la session à laquelle il aimerait répondre. Ce code c’est celui qui a été généré dans l’application S2H Creator puis communiqué aux apprenants.

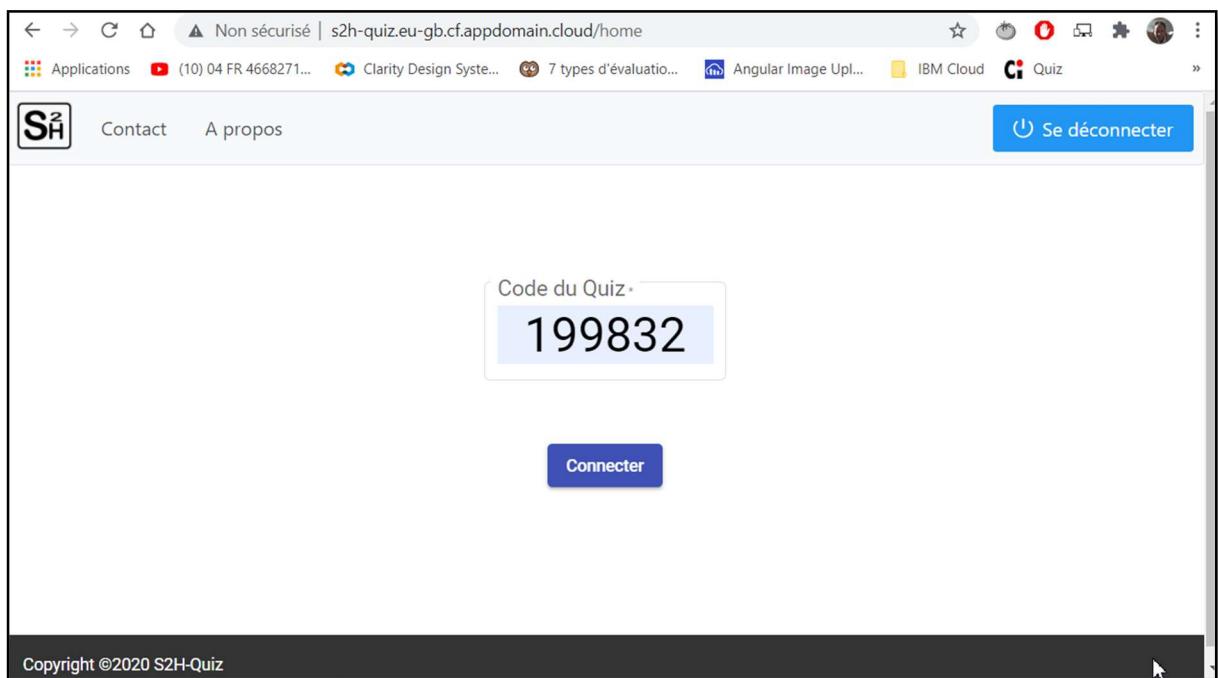


Figure 36 : Landpage S2H Quiz

Après avoir validé en cliquant sur le bouton « Connecter », deux cas se présentent :

- Si la session est anonyme, l’apprenant sera directement redirigé vers l’interface de tests.
- Sinon, il devra passer par un formulaire d’authentification avant d’aller à l’interface de test.

5-2-2 La page de tests

Cette page contient un Stepper représentant l’ensemble des questions du Quiz proposé. L’apprenant pourra naviguer entre les questions soit en cliquant directement sur le numéro de la question voulue ou moyennant les boutons « Suivant » et « Précédent ».

A la fin du test S'affichera le bouton « Terminer » permettant de poster les réponses et l'avoir le résultat de l'évaluation.

The screenshot shows a quiz interface. At the top, there's a logo 'S2H', 'Contact', 'A propos', and a 'Se déconnecter' button. Below that, it says 'Quiz : Python' and 'Description : Le langage de Programmation Python'. It shows the user's name 'Ferid Helali'. A navigation bar at the bottom has numbers 1 through 25. The current question is 'How to use math module?'. Two options are listed: 'from math import *' and 'using math.*'. A note says 'NB : Choisir la réponse valide'. The page footer shows '4/25', 'Précédent', and 'Suivant' buttons.

Figure 37 : Page de test S2H Quiz

5-2-3 Page du résultat

Après avoir terminé l'évaluation, l'application affichera le résultat du test.

The screenshot shows the results page. At the top, it has the 'S2H' logo, 'Contact', 'A propos', and 'Se déconnecter' buttons. The main message is 'Vous avez répondu correctement à 3 questions parmi 4' and 'Votre score est de 75 %'. The page footer contains the text 'Copyright ©2020 S2H-Quiz'.

Figure 38 : Page de résultats

5-2-4 Interface responsive

L'application S2H Quiz a été pensée pour qu'elle soit accessible via un smartphone, pour cela les interfaces de l'application sont très responsives.

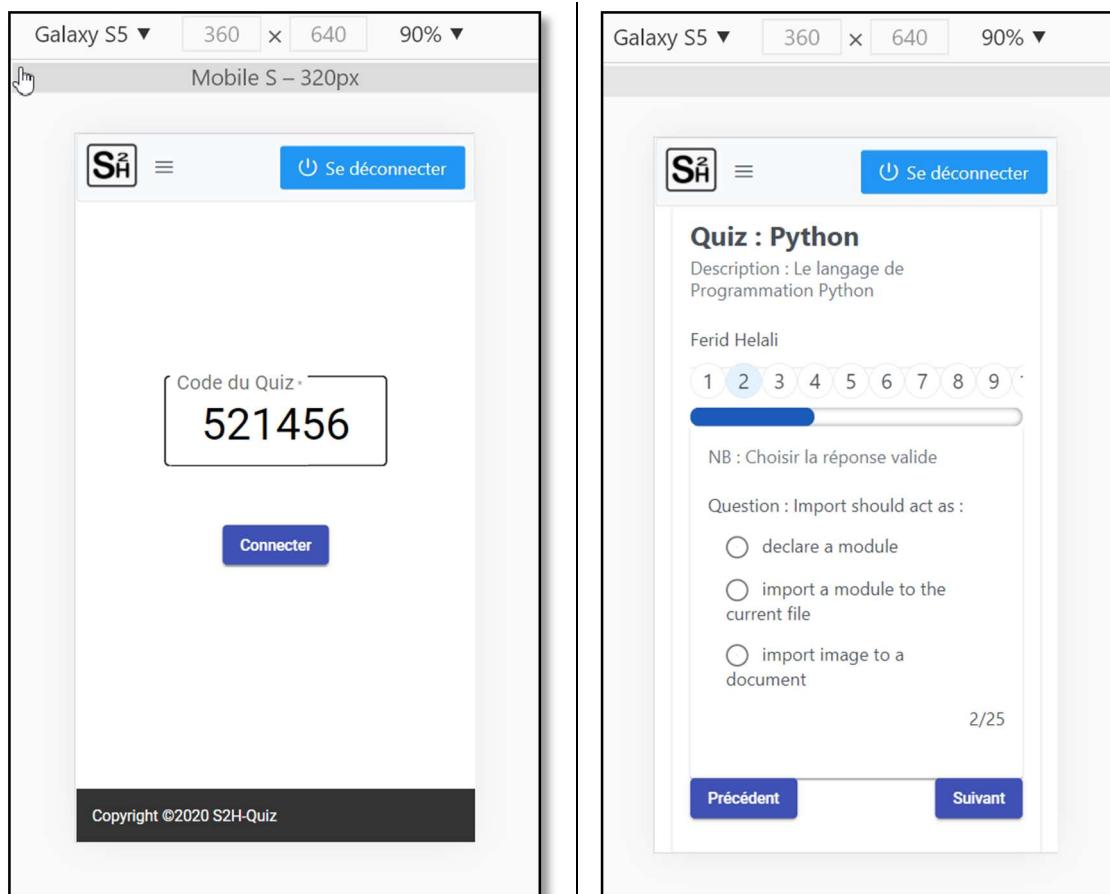


Figure 39 : Page d'accueil sur Smartphone

Figure 40 : Page de tests sur Smartphone

Conclusion générale

Les processus d'apprentissage dans le monde ne cessent de s'améliorer ou de changer cherchant à récolter des générations de professionnels plus compétents. L'évaluation est l'un des piliers les plus importants de tout processus d'apprentissage.

L'évaluation continue dans un processus d'apprentissage permet de renforcer les connaissances acquises lors de la formation. L'application réalisée dans le cadre de ce projet sert à banaliser les évaluations à tout moment de l'apprentissage en simplifiant la tâche de l'enseignant par l'utilisation d'une interface graphique ergonomie, par l'automatisation de la correction et de l'attribution de note et par offrir un tableau de bord permettant d'avoir une vue globale sur l'évolution des compétences des apprenants.

Sur le plan personnel, ce projet m'a permis de côtoyer de nouvelles technologies de développement d'applications telles que le développement Full Stack, les méthodologies agiles, le cloud computing, la culture DevOps, le système de versionning Git et d'autres. Ce qui m'a permis de rattraper un peu le temps perdu.

Comme perspective, l'application peut s'évoluer en proposant de nouvelles formes de questions autres que celles proposées, d'avoir une application mobile permettant de consommer les Quiz ou alors d'exposer quelques fonctions de l'API en vue de la consultation publique des statistiques.

Bibliographie

- [1] EasyLMS, [Accès le 13 Mai 2020], adresse :
<https://www.onlineassessmenttool.com/fr/base-connaissances/base-de-connaissances-evaluations/types-evaluations/item10637>
- [2] Manitoba, [Accès le 15 Avril 2020], adresse :
<https://www.edu.gov.mb.ca/m12/eval/role.html>
- [3] toppr, [Accès le 20 Aout 2020], adresse :
<https://www.toppr.com/bytes/continuous-assessment-features-and-purpose/>
- [4] Agile Alliance, [Accès le 25 Mai 2020], adresse :
<https://www.agilealliance.org/glossary/xp>
- [5] OMG, [Accès le 20 Septembre 2020], adresse : <https://www.omg.org/>
- [6] Slide Share, [Accès le 22 Septembre 2020], adresse :
<https://fr.slideshare.net/YounessBoukouchi/mthode-agiles-scrum-et-xp>
- [7] Le Parisien, [Accès le 11 Novembre 2020], adresse :
<http://dictionnaire.sensagent.leparisien.fr/%C3%89valuation%20continue/fr-fr/>
- [8] Synopsys, [Accès le 02 Septembre 2020], adresse :
<https://www.synopsys.com/blogs/software-security/agile-cicd-devops-difference/>
- [9] LERJEN.ME, [Accès le 02 Mai 2020], adresse : <https://lerjen.me/mean-stack/>
- [10] Sparx Systems, [Accès le 12 Novembre 2020], adresse :
<https://www.sparxsystems.fr/platforms/uml.html>
- [11] sitepoint, [Accès le 02 Mai 2020], adresse :
<https://www.sitepoint.com/introduction-mean-stack/>
- [12] anthedesign, [Accès le 12 Aout 2020], adresse :
<https://www.anthedesign.fr/webdesign-2/wireframe/>
- [13] EasyLMS, [Accès le 19 Septembre 2020], adresse : <https://www.pierre-giraud.com/git-github-apprendre-cours/presentation-git-github/>