

Table des matières

Table des matières.....	1
Remerciements.....	2
Introduction.....	3
Contexte du projet.....	4
Méthode de travail.....	5
SCRUM.....	6
2TUP.....	7
UX Design.....	8
Figure 5: UX Design Process.....	8
Planning.....	9
Gestion des tâches : GitHub projet.....	9
Branch technique.....	10
Capture des besoins techniques.....	11
Analyse technique.....	12
Autoformation.....	13
Prototype.....	14
Branch Fonctionnelle.....	15
Carte d'empathie.....	16
Formateur.....	16
Apprenant.....	17
Définir Le Problème.....	18
Ideation.....	19
Figure 9: ideation.....	19
Use Case Diagram (General).....	20
Figure 10: use case diagram (general).....	20
Use Case Diagram (Sprint 1).....	21
Figure 11: use case diagram (sprint 1).....	21
Use Case Diagram (Sprint 2).....	22
Figure 12: use case diagram (sprint 2).....	22
Conception.....	23
Diagramme de class.....	24
Maquette.....	25
A- Maquette 1:.....	26
B- Maquette 2:.....	27
c- Maquette 3:.....	28
Realisation.....	29
Conclusion.....	30

Remerciements

Je tiens à exprimer ma profonde gratitude à toutes les personnes qui ont joué un rôle essentiel dans le succès de mon projet final.

Un merci particulier à M. ES-SARRAJ FOUAD pour son enseignement exceptionnel, ses conseils précieux et son dévouement sans faille envers tous ses étudiants.

Je souhaite également adresser mes remerciements sincères à l'équipe de SOLICODE pour leur patience, leur soutien et leurs efforts inlassables pour nous offrir un cadre d'apprentissage optimal.

Un grand merci à mes collègues pour leur collaboration, leur humour et les moments inoubliables que nous avons partagés tout au long de l'année.

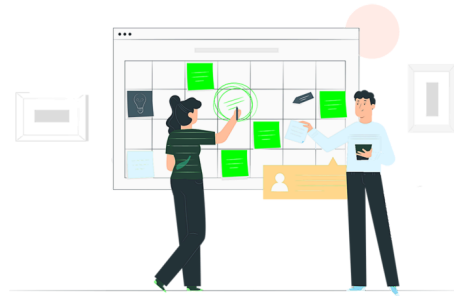
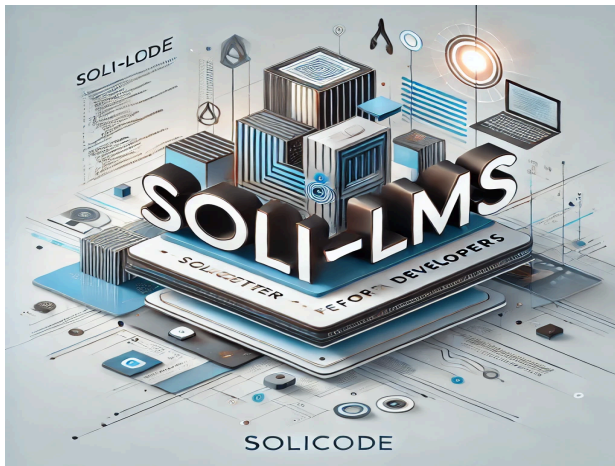
Enfin, à toutes celles et ceux qui ont contribué, directement ou indirectement, par leur expertise et leurs conseils, je vous suis profondément reconnaissant(e). Même si je ne peux pas mentionner tout le monde individuellement, sachez que votre impact n'est pas passé inaperçu.

Merci à tous pour votre soutien et vos contributions au succès de ce projet.

Introduction

Dans le cadre du projet Solicode LMS, ma responsabilité principale était de développer le module de **Suivi d'avancement des autoformations**. Ce module a pour objectif de fournir aux apprenants une plateforme intuitive pour suivre leur progression dans leurs parcours de formation en ligne. Parallèlement, il offre aux formateurs des outils leur permettant de valider les projets soumis par les apprenants et d'identifier rapidement les apprenants rencontrant des difficultés dans leur parcours d'apprentissage. Ce rapport décrit les différentes étapes de développement de cette fonctionnalité, en s'appuyant sur diverses méthodologies telles que Scrum, Design Thinking, et la méthode Y(2Tup).

Contexte du projet



Le **Suivi d'avancement des autoformations**. s'inscrit dans le cadre du projet **Soli-LMS**, qui vise à centraliser et automatiser les processus d'apprentissage. Ce module se concentre sur le suivi en temps réel des activités des apprenants, la visualisation des progrès via des tableaux de bord interactifs, et la génération de rapports détaillés pour les formateurs.

Méthode de travail

Scrum

Méthode 2tup

UX Design

SCRUM

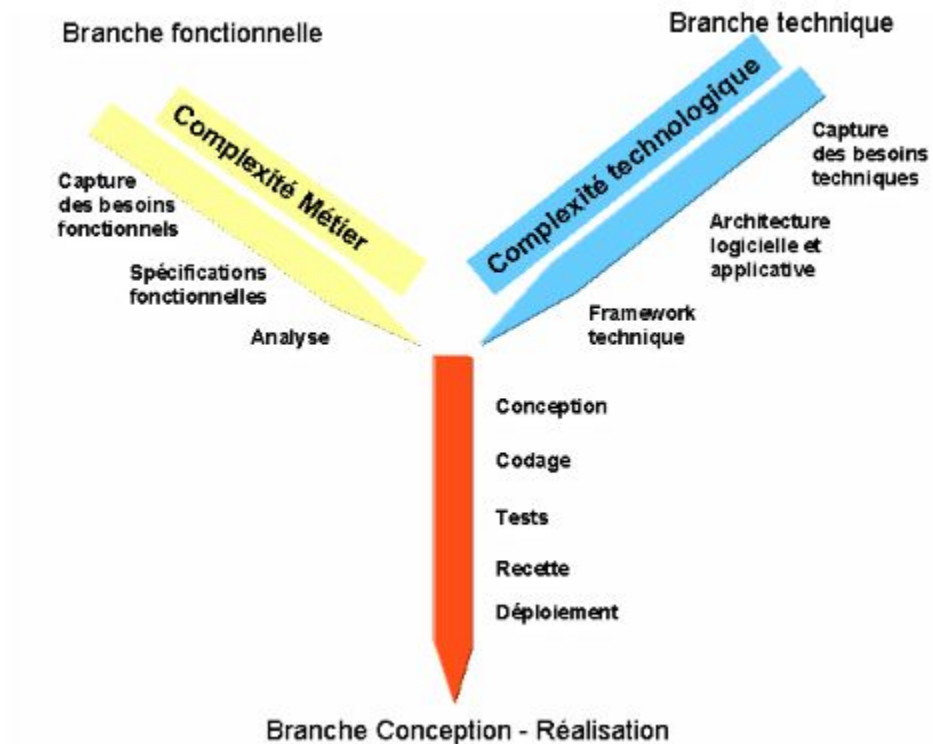
SCRUM PROCESS



Le développement du projet suit la méthodologie agile SCRUM, qui permet de gérer le projet par sprints. Chaque sprint est une période de travail focalisée sur des tâches spécifiques, permettant d'itérer rapidement sur les fonctionnalités et d'adapter le produit aux besoins des utilisateurs. SCRUM aide à optimiser la communication entre les membres de l'équipe et à livrer des versions progressives et testées du produit.

Figure 3: Scrum Methodology

2TUP



La méthode 2TUP (Two-Tiered Unified Process) permet de structurer le développement en deux phases principales : la phase d'analyse (compréhension des besoins) et la phase de développement (implémentation). Cette approche facilite la gestion des tâches en simplifiant la planification tout en mettant l'accent sur l'amélioration continue à chaque itération.

UX Design

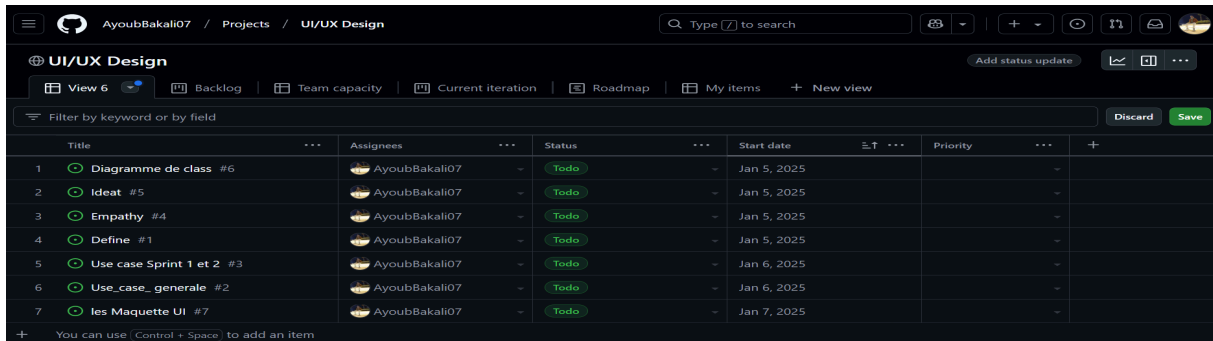


Figure 5: UX Design Process

Planning

Gestion des tâches : GitHub projet

Toutes les tâches et fonctionnalités sont organisées et suivies sur GitHub. Les issues sont utilisées pour décrire les tâches spécifiques de chaque sprint, et les pull-requests permettent d'intégrer progressivement le code développé à la branche principale du projet.



The screenshot shows a GitHub project board for the 'UI/UX Design' project. The board is titled 'UI/UX Design' and has a search bar at the top. Below the title, there are tabs for 'View 6', 'Backlog', 'Team capacity', 'Current iteration', 'Roadmap', 'My items', and 'New view'. A filter bar is present with the text 'Filter by keyword or by field'. The main table lists tasks with columns for Title, Assignees, Status, Start date, and Priority. The tasks are numbered 1 through 7 and all have a status of 'Todo'.

	Title	Assignees	Status	Start date	Priority
1	Diagramme de class #6	AyoubBakali07	Todo	Jan 5, 2025	
2	Ideat #5	AyoubBakali07	Todo	Jan 5, 2025	
3	Empathy #4	AyoubBakali07	Todo	Jan 5, 2025	
4	Define #1	AyoubBakali07	Todo	Jan 5, 2025	
5	Use case Sprint 1 et 2 #3	AyoubBakali07	Todo	Jan 6, 2025	
6	Use_case_generale #2	AyoubBakali07	Todo	Jan 6, 2025	
7	les Maquette UI #7	AyoubBakali07	Todo	Jan 7, 2025	

Branch technique

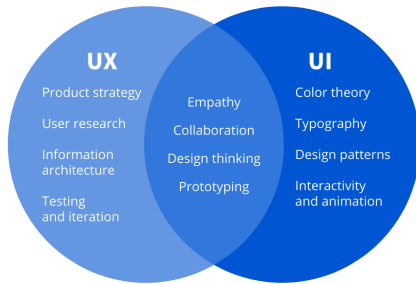
- Prototype

Le prototype initial a été développé sous forme d'une application blog, permettant de tester l'interface utilisateur avant de la déployer dans un environnement de production.

Capture des besoins techniques

- Le module nécessite des outils d'analyse en temps réel, des indicateurs de performance clé (KPI), et un tableau de bord permettant aux apprenants et formateurs de visualiser la progression.

Analyse technique



UI/UX



Ajax: (One page)

L'interface utilisateur (UI) et l'expérience utilisateur (UX) sont des aspects essentiels pour garantir que l'application soit facile à utiliser. AJAX sera utilisé pour rendre les interactions en temps réel, comme la soumission des formulaires et l'affichage des résultats des tests, sans rechargement complet de la page.

Autoformation



Composant UI

Pour garantir une maîtrise complète de l'interface, des composants de l'UI tels que les formulaires dynamiques, les graphiques de résultats et les tableaux de bord interactifs seront utilisés. Une formation continue est essentielle pour se familiariser avec les technologies front-end telles que HTML5, CSS3, JavaScript, et les frameworks comme React ou Vue.js.

Prototype

Le prototype initial a été développé sous forme d'une application blog, permettant de tester l'interface utilisateur avant de la déployer dans un environnement de production.

Branch Fonctionnelle

- Carte d'empathie
- Définir le probleme
- Ideation
- Cas d'utilisation général
- Cas d'utilisation (Sprint 1)
- Cas d'utilisation (Sprint 2)

Carte d'empathie

Formateur

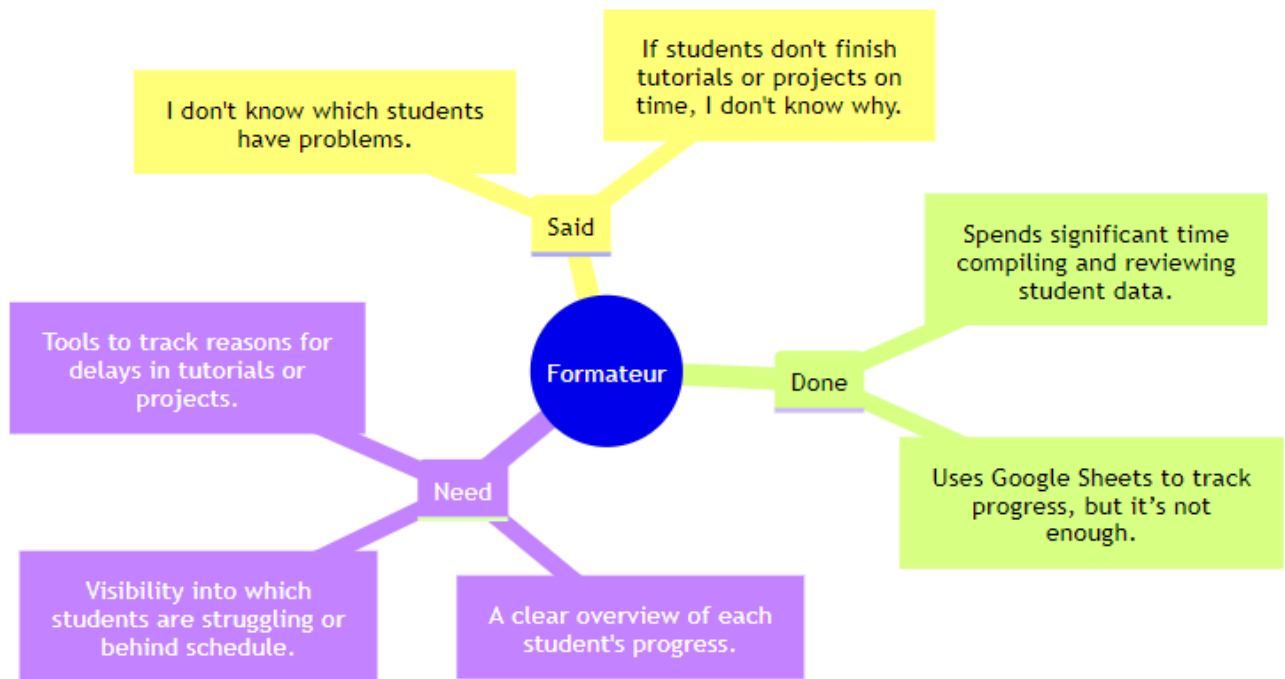


Figure 8: Empathy (responsible for learners)

Apprenant



Définir Le Problème

Les formateurs rencontrent des difficultés pour suivre efficacement les progrès des étudiants dans les programmes d'auto-formation, en particulier dans les tutoriels et les projets. Actuellement, les outils utilisés, tels que Google Sheets, sont insuffisants pour offrir une vision claire et structurée.

Ideation

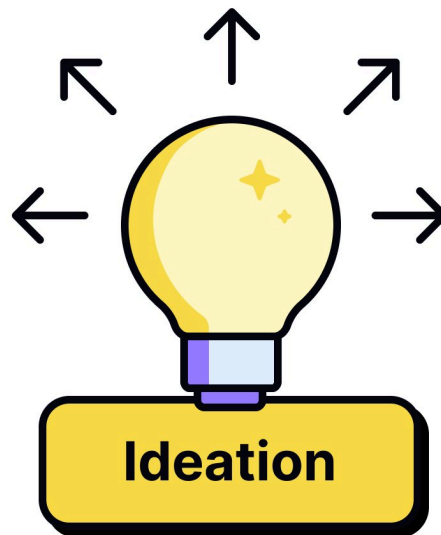


Figure 9: ideation

- 1. Develop a Dashboard for Student Progress**
 - Provide visual progress indicators for students.
 - Include filters to view students by progress status
- 2. Create an Alert System**
 - Highlight students with incomplete tutorials/projects
- 3. Implement a Reason Tracking Feature**
 - Enable Formateur to add notes or comments on student progress.
- 4. Introduce a Structured Feedback Mechanism**
 - Allow Formateur to provide feedback directly through the system.
 - Store feedback history for future reference.
- 5. Enable Offline Access to Reports**
 - Offer an export feature to download reports for offline use.
 - Ensure reports are comprehensive and easy to share with other stakeholders.

Use Case Diagram (General)

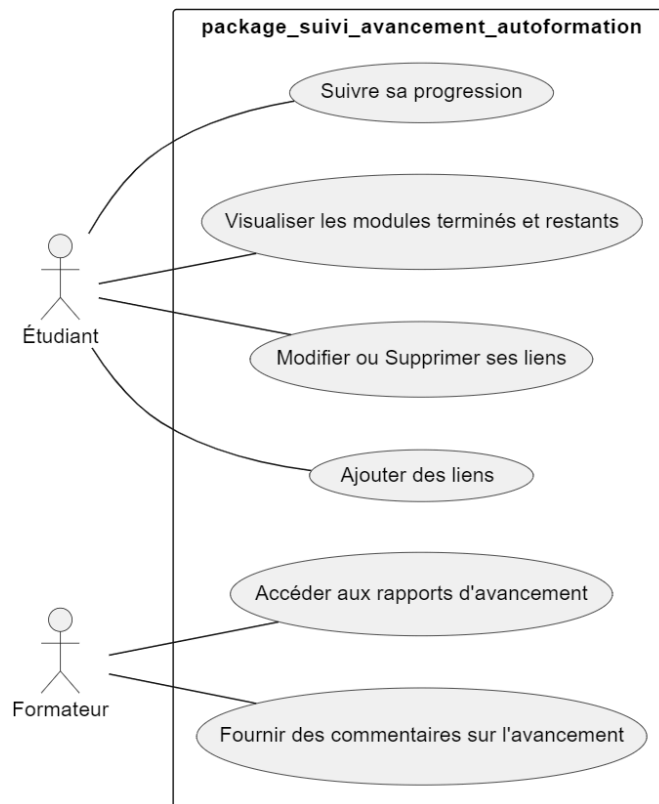


Figure 10: use case diagram (general)

Use Case Diagram (Sprint 1)

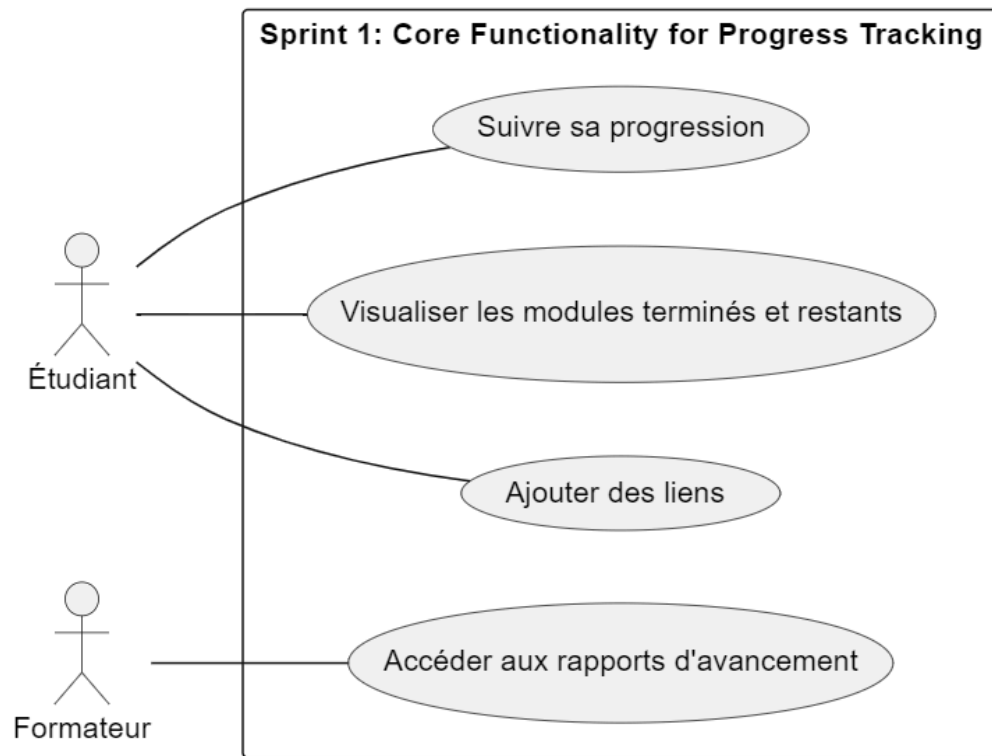


Figure 11: use case diagram (sprint 1)

Use Case Diagram (Sprint 2)

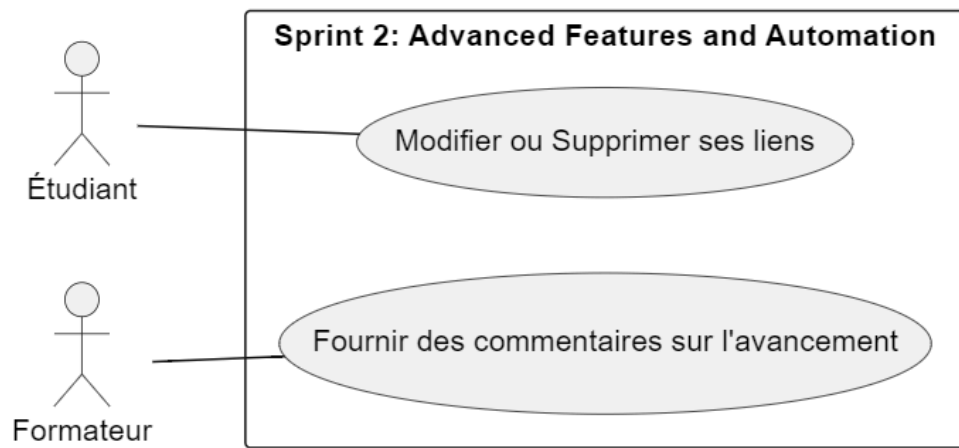
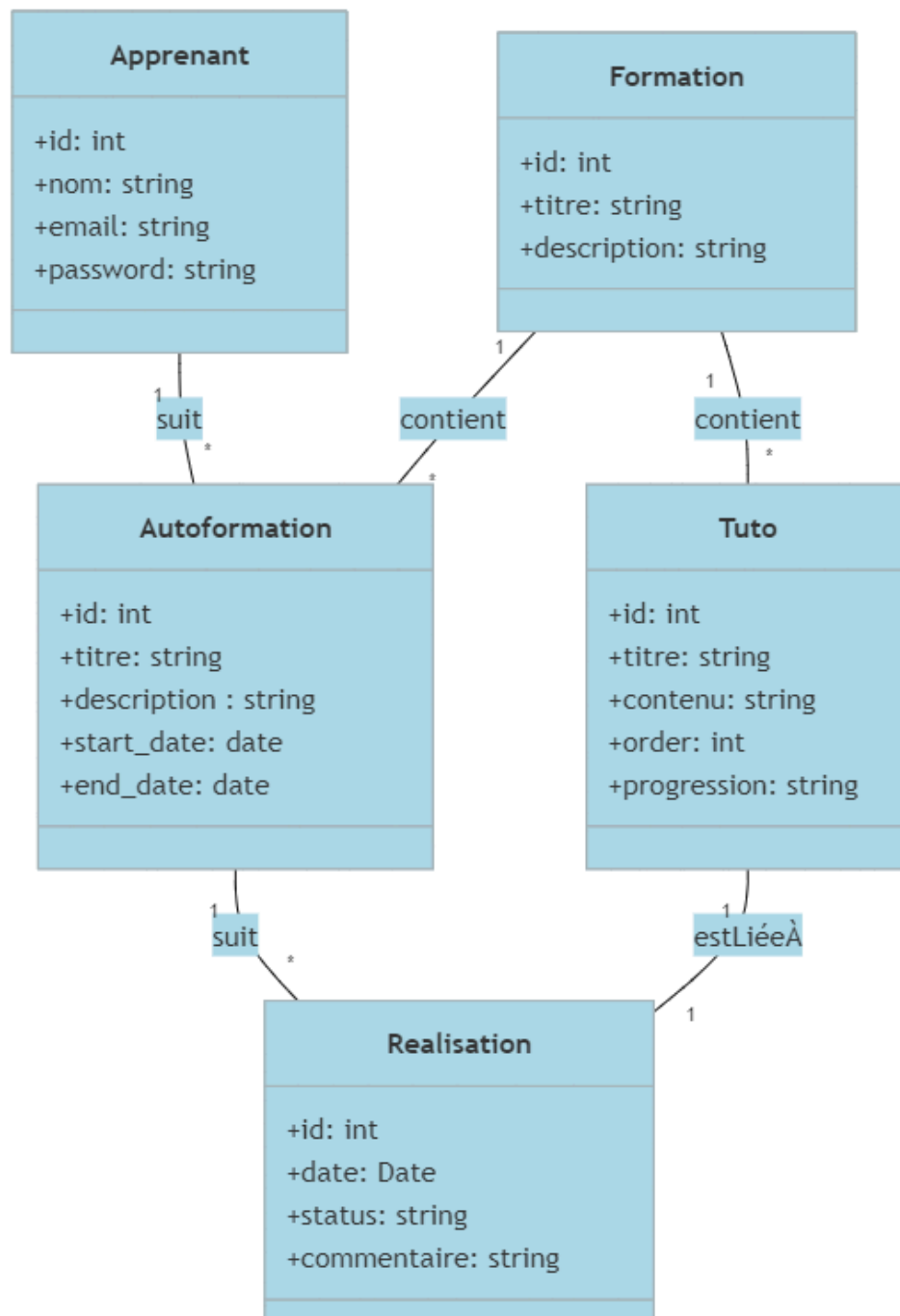


Figure 12: use case diagram (sprint 2)

Conception

- Diagramme des Classes
- Maquette

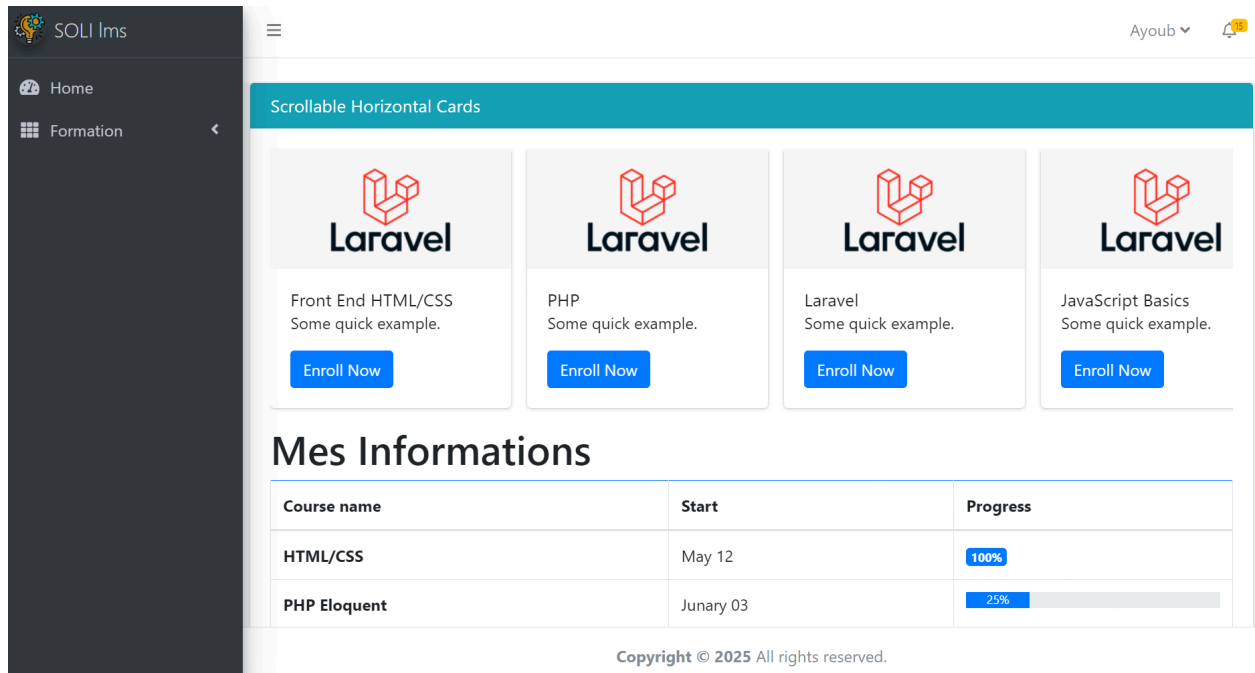
Diagramme de class



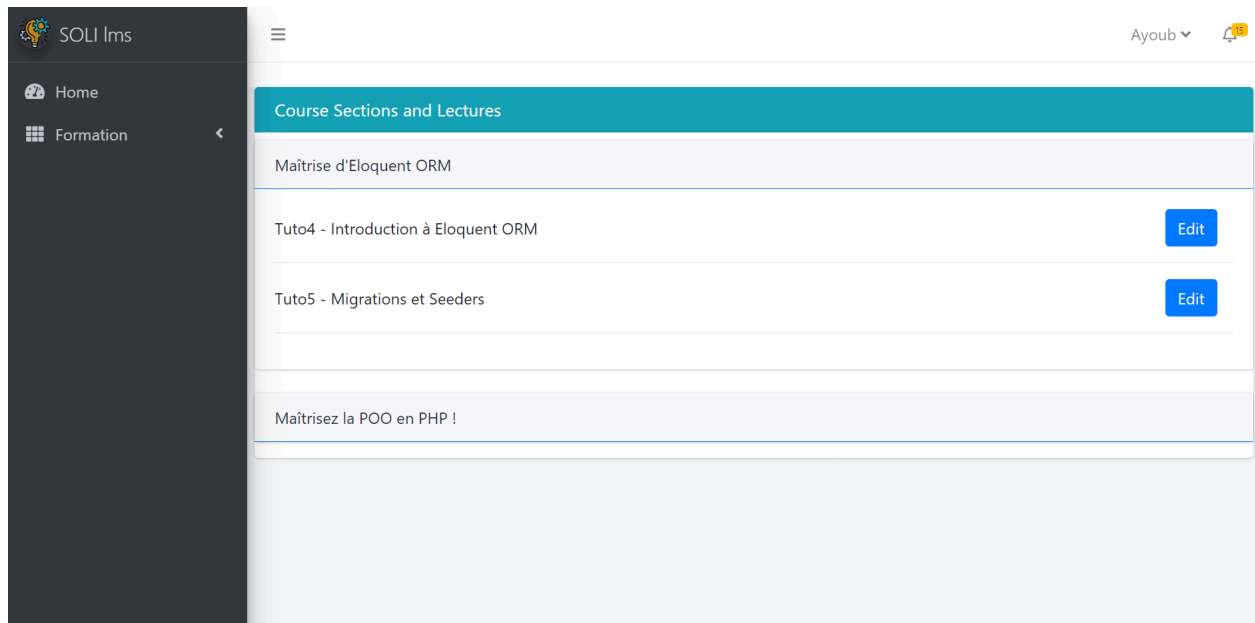
Maquette

- a Maquette 1
- b Maquette 2
- c Maquette 3

A- Maquette 1:



B- Maquette 2:



c- Maquette 3:

The screenshot displays a web application interface with a dark sidebar on the left containing 'Home' and 'Formation' links. The main content area shows a list of course sections. A modal window titled 'Edit Introduction to Eloquent ORM' is open, featuring four input fields for 'Status', 'GitHub Link', 'Project Link', and 'Slide Link'. The modal also includes 'Close' and 'Save changes' buttons at the bottom right. The background content is dimmed, showing course sections like 'Maîtrise d'Eloquent ORM' and 'Tuto4 - Introduction'.

Edit Introduction to Eloquent ORM	
Status	<input type="text" value="Enter your status"/>
GitHub Link	<input type="text" value="Enter your GitHub link"/>
Project Link	<input type="text" value="Enter your project link"/>
Slide Link	<input type="text" value="Enter your slide link"/>
<div>Close Save changes</div>	

Realisation

Conclusion