



# Rapport de Projet file rouge

## Gestion de projet : Développement d'un système LMS

### Formation développement mobile mode bootcamp

**Réalisé par :** Bouik Hussein

**Encadré par :** Essarraïj Fouad

Année de formation 2023/2024

# Table des matières

<b>Table des matières</b>	<b>2</b>
<b>Listes des figures</b>	<b>2</b>
<b>Remerciement</b>	<b>4</b>
<b>Introduction</b>	<b>5</b>
<b>Cahier de charge</b>	<b>6</b>
Contexte du projet	6
Objectifs de formation	7
<b>Processus de développement et méthodes</b>	<b>8</b>
2TUP (Two Tracks Unified Process) : Un développement en deux branches	8
Design Thinking : l'humain au cœur du processus	10
Le Développement Agile : Une Approche Adaptative	11
Scrum : Une Méthodologie Agile Populaire	12
Gestion des tâches	13
Diagramme de Gantt	13
<b>Branche fonctionnelle</b>	<b>14</b>
Définir le problème	16
Idéation	17
Diagramme de cas d'utilisation	18
Sprints backlog	19
<b>Branche technique</b>	<b>20</b>
Architecture de l'application	20
Analyse technique	21
Conception générique	22
Prototype	22
Gestion des Projets (Sprint 1):	22
<b>Conception</b>	<b>23</b>
Diagramme de classes	24
Diagramme des packages	25
Maquettes	26
<b>Réalisation</b>	<b>27</b>
Technologies	27
Outils	28
<b>Conclusion</b>	<b>29</b>
Annexe 1: Maquette	30
Annexe 2: Planification	31

# Listes des figures

Figure 1 : Introduction	5
Figure 2 : Schéma cahier de charge	6
Figure 3 : Schéma objectif de formation	7
Figure 4 : 2TUP	8
Figure 5 : Design thinking	10
Figure 6: Méthode agile	11
Figure 7 : Les phases d'une méthode Scrum	12
Figure 8 : Diagramme de Gantt	13
Figure 9 : Carte d'empathie Formateur	14
Figure 10 : Schéma définir le problème	16
Figure 11 : Idéation	17
Figure 12 : Diagramme de cas d'utilisation	18
Figure 13 : Architecture de l'application	21
Figure 14 : Diagramme de classe pkg_creation_projets	24
Figure 15 : Diagramme des packages	25
Figure 16 : Maquette de côté formateur	26
Figure 17 : Maquette de côté apprenant	26
Figure 18 : Création projet phase 1 (projet-livrables-ressources)	30
Figure 19: Création projet phase 3 (affectation du projet)	31
Figure 20: Backlog du pkg_creation_projets	31

# Remerciement

Ce projet n'aurait pu voir le jour sans le soutien précieux et l'aide bienveillante de plusieurs personnes que je tiens à remercier sincèrement, sans oublier ma propre persévérance et mon engagement tout au long de cette aventure.

Tout d'abord, je souhaite exprimer ma profonde gratitude à Monsieur Fouad Essaraj, mon encadrant de projet, pour sa disponibilité, ses précieux conseils et son accompagnement constant. Son expertise m'a été d'une aide précieuse tout au long de ce projet.

Je remercie également ma famille, mes amis, mes collègues et les employés de l'établissement pour leur soutien indéfectible, leurs encouragements et leur aide précieuse tout au long de ce projet.

Enfin, je tiens à remercier tous ceux qui, de près ou de loin, ont contribué à la réalisation de ce travail, et surtout, je me remercie moi-même pour ma détermination, mon travail acharné et ma capacité à ne jamais abandonner malgré les défis rencontrés.

## Introduction

Le Centre Solidaires Digital SOLICODE s'engage à offrir des formations de qualité supérieure, accessibles à tous. Pour concrétiser cette ambition, le centre a mis en place Soli-LMS, un système de gestion d'apprentissage (LMS) complet et intuitif, destiné à révolutionner la gestion des formations.

Le cœur de ma mission réside dans le package de création de projets de Soli-LMS. Cet outil est un élément fondamental pour garantir la qualité des projets de formation. Il permet aux formateurs de créer des projets pertinents et alignés sur les compétences recherchées, assurant ainsi une expérience d'apprentissage optimale pour les étudiants.

Mon travail au sein de ce projet a consisté à développer et à améliorer ce package. L'objectif principal était de simplifier la création de projets pour les formateurs, tout en assurant une flexibilité maximale et une intégration transparente avec les compétences visées.

*Figure 1 : Introduction*



# Cahier de charge

## Contexte du projet

Le Centre Solidaires Digital **SOLICODE** aspire à offrir des formations de qualité supérieure, accessibles à tous. Pour y parvenir, le centre a besoin d'une solution de gestion de formation intuitive et complète. Soli-LMS est la réponse idéale. Cette plateforme, conçue pour simplifier les processus et optimiser la gestion des formations, permettra au centre de :

- **Améliorer l'efficacité des formations:** en centralisant la gestion des projets et en offrant un suivi personnalisé des apprenants.
- **Accroître la qualité des formations:** en facilitant la création de contenus et l'évaluation des compétences.
- **Renforcer la satisfaction des apprenants:** en leur offrant une expérience d'apprentissage fluide et intuitive.



Figure 2 : Schéma cahier de charge

## Objectifs de formation

Soli-LMS est développé en mettant en avant les compétences en développement web, conception et gestion de projet. La plateforme offrira une expérience utilisateur optimale, accessible sur tous les appareils. En utilisant des technologies performantes comme Laravel et MySQL, Soli-LMS permettra de simplifier la gestion des données et d'améliorer les processus de formation.

### Fonctionnalités clés de la plateforme:

- **Interface intuitive et responsive:** facilitant l'accès et la navigation pour les clients.
- **Communication fluide:** permettant aux apprenants et aux formateurs d'échanger facilement.
- **Évaluation précise des projets:** permettant d'évaluer le progrès et les compétences des apprenants.

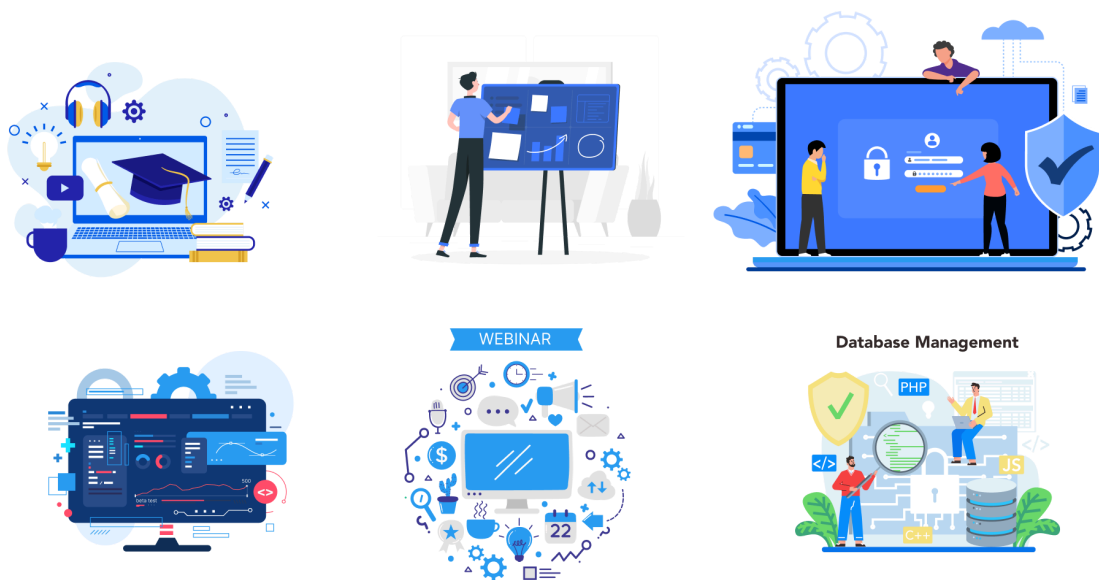


Figure 3 : Schéma objectif de formation

# Processus de développement et méthodes

Le développement de notre plateforme repose sur une approche rigoureuse et structurée, combinant le meilleur des méthodes agiles 2TUP et du Design Thinking. Cette combinaison nous permet de garantir une solution à la fois fonctionnelle, ergonomique et parfaitement adaptée aux besoins réels des utilisateurs.

## 2TUP (Two Tracks Unified Process) : Un développement en deux branches

2TUP est comme un plan d'architecte raffiné, une méthode basée sur le Processus Unifié, éprouvé et fiable. Tout commence par une analyse approfondie, une exploration méticuleuse des acteurs impliqués, de leurs besoins et de l'environnement dans lequel le système évoluera.

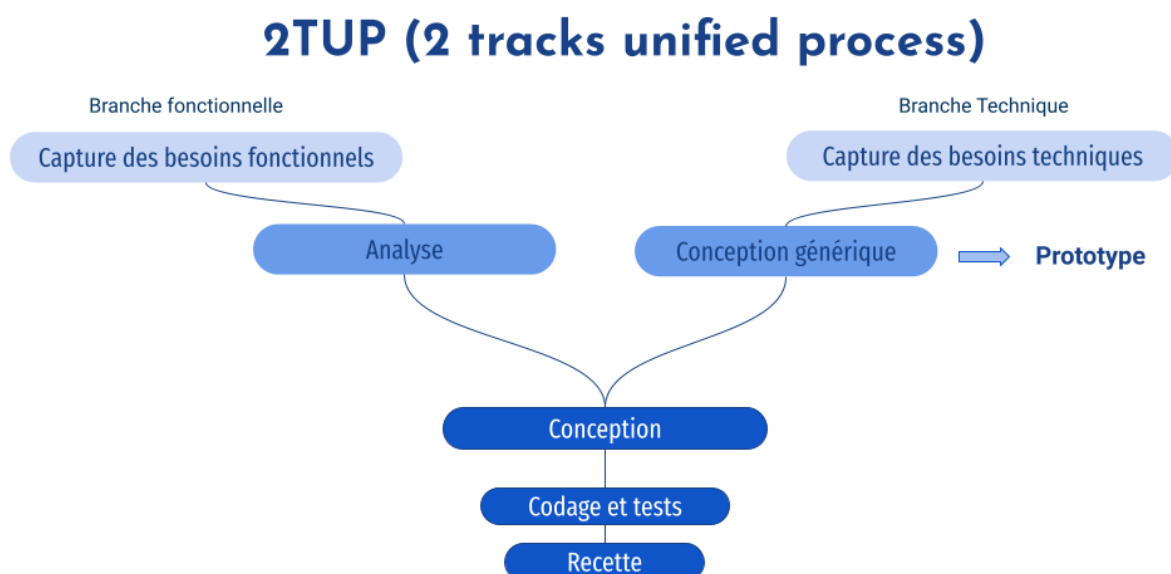


Figure 4 : 2TUP



Ensuite, la méthode se divise en deux voies parallèles :

- **La Voie Fonctionnelle** : C'est là que nous cartographions les capacités du système et élaborons des maquettes pour obtenir une vision claire.
- **La Voie Technique** : Ici, nous définissons toutes les contraintes techniques, en identifiant les briques qui composeront l'architecture.

Enfin, nous fusionnons ces voies, rassemblant le tout pour créer une solution sur mesure. C'est un processus systématique, qui culmine en codage et en tests rigoureux pour garantir que le système répond à toutes les exigences.

### **La Voie Fonctionnelle : Comprendre les besoins**

- Identification des acteurs (formateurs, apprenants, administrateurs) et de leurs interactions avec la plateforme.
- Formalisation des besoins et des fonctionnalités attendues.
- Modélisation des processus clés (création de projets, attribution de compétences, validation, etc.).
- Création de maquettes d'interface (wireframes) pour visualiser l'expérience utilisateur.

### **La Voie Technique : Bâtir les fondations**

- Identification des contraintes techniques (environnement d'hébergement, sécurité, performance).
- Choix des technologies les plus adaptées (PHP, Laravel, MySQL, etc.).
- Définition de l'architecture technique de la plateforme (MVC).
- Mise en place des outils de développement et de test.

### **La phase de réalisation : Convergence et concrétisation**

- Fusion des éléments des branches fonctionnelle et technique.
- Développement itératif et incrémental de la plateforme, en sprints.
- Tests et validation des fonctionnalités à chaque étape du développement.
- Livraison d'une solution fonctionnelle et conforme aux besoins.

## Design Thinking : l'humain au cœur du processus

Le Design Thinking est une approche centrée sur l'humain pour la résolution de problèmes qui utilise un processus en cinq phases pour créer des solutions innovantes. Il met l'accent sur l'empathie, l'idéation, l'expérimentation et l'itération.

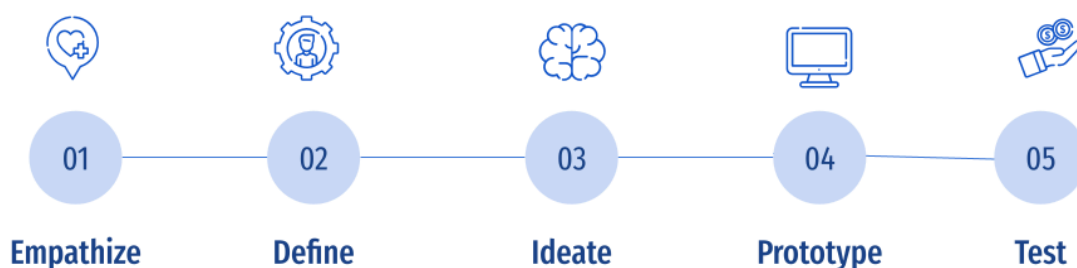


Figure 5 : Design thinking

Le Design Thinking, approche centrée sur l'utilisateur, nourrit chaque étape de notre développement :

- **Empathie** : Comprendre en profondeur les besoins, les frustrations et les motivations des formateurs.
- **Définition** : Définir clairement le problème à résoudre et les objectifs à atteindre.
- **Idéation** : Générer un maximum d'idées et de solutions possibles.
- **Prototypage** : Créer des prototypes (maquettes, versions simplifiées) pour tester les idées et recueillir des retours.
- **Test** : Faire tester les prototypes par les utilisateurs finaux pour identifier les points forts et les axes d'amélioration.

En combinant 2TUP et Design Thinking, nous nous assurons de développer une plateforme qui répond non seulement aux besoins fonctionnels, mais aussi aux attentes et aux préférences des utilisateurs, offrant ainsi une expérience optimale.

## Le Développement Agile : Une Approche Adaptative

Le développement Agile est comme une danse de flexibilité et d'itération. Il s'agit de livrer des versions fonctionnelles de logiciels à intervalles réguliers, en s'adaptant en permanence aux besoins changeants des utilisateurs.



*Figure 6: Méthode agile*

## Scrum : Une Méthodologie Agile Populaire

Scrum est une méthodologie de gestion de projet agile qui se base sur des cycles de travail itératifs appelés « sprints ». Il implique un Product Owner, un Scrum Master, une équipe de développement et utilise des événements tels que les réunions quotidiennes, les revues de sprint et les rétrospectives de sprint pour favoriser la collaboration, la transparence et la livraison régulière de fonctionnalités de haute qualité. Le Scrum permet aux équipes de répondre rapidement aux changements et de livrer de la valeur plus rapidement.

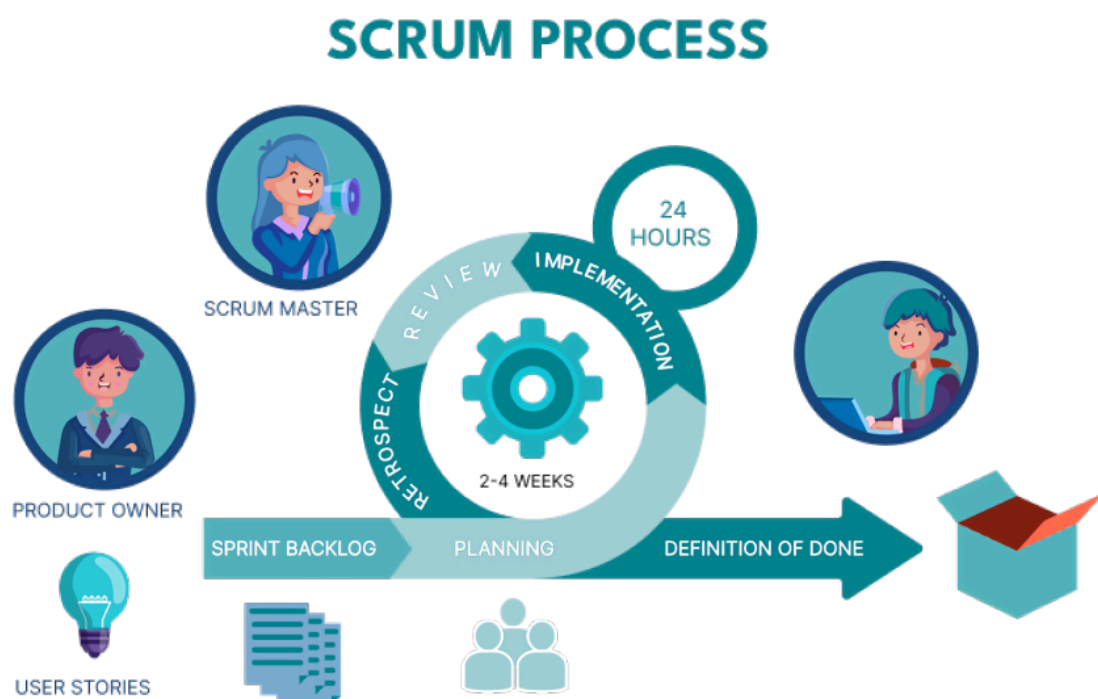


Figure 7 : Les phases d'une méthode Scrum

## Planification

Avant d'entamer notre projet, il est essentiel d'élaborer un plan détaillé de toutes les tâches à accomplir en utilisant les méthodologies de travail que nous avons présentées précédemment. Cette planification permet de définir les objectifs, les exigences, les ressources et les délais nécessaires pour mener à bien le projet.

## Diagramme de Gantt

Un diagramme de Gantt est un outil de gestion de projet qui montre la planification de toutes les tâches d'un projet sur une ligne de temps.

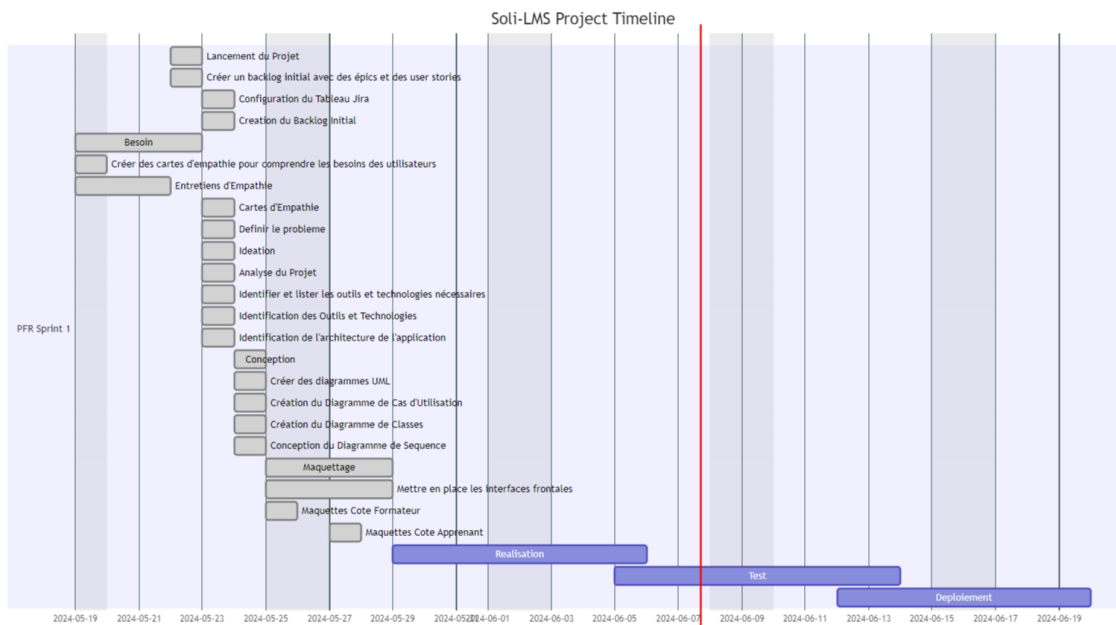


Figure 8 : Diagramme de Gantt

## Branche fonctionnelle

Dans cette phase de capture des besoins fonctionnels de notre plateforme de gestion de projets et d'apprentissage, nous utilisons une approche centrée sur l'utilisateur. En nous appuyant sur des outils tels que la carte d'empathie et la définition du problème, ainsi que sur des techniques d'idéation, nous identifions les besoins spécifiques des formateurs. La modélisation UML, à travers des diagrammes de cas d'utilisation et de classes, vient ensuite formaliser et structurer ces besoins.

### Empathie : Formateur

Une séance d'empathie a été réalisée avec les formateurs de centre solicode le 23 février 2024.

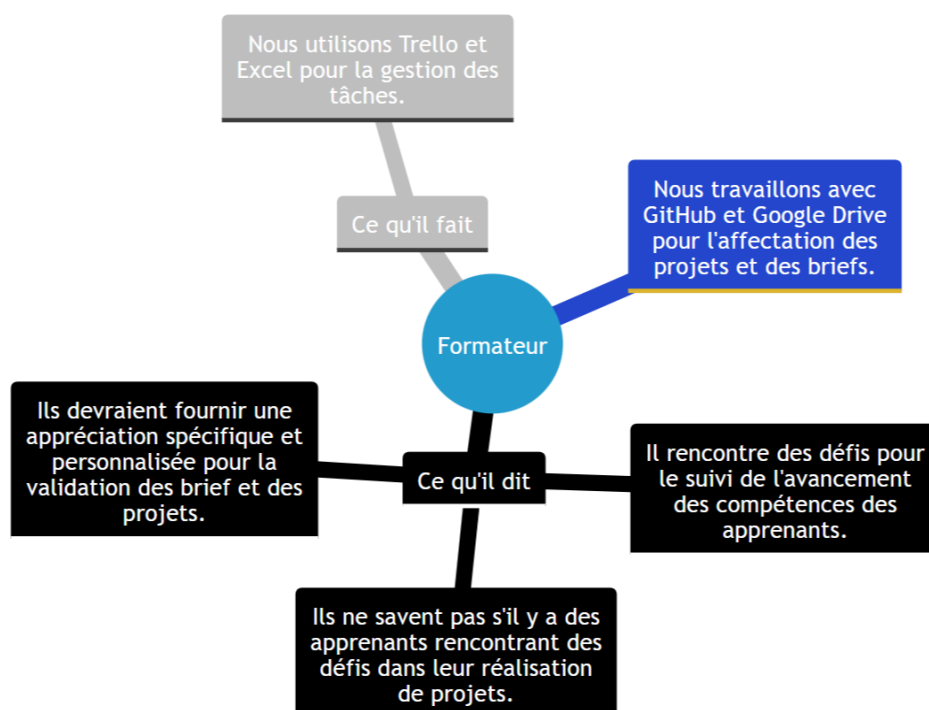


Figure 9 : Carte d'empathie Formateur

### **Que dit-il ?**

- Ils devraient fournir une appréciation spécifique et personnalisée pour la validation des brief et des projets.
- Ils ne savent pas s'il y a des apprenants rencontrant des défis dans leur réalisation de projets.
- Il rencontre des défis pour le suivi de l'avancement des compétences des apprenants.

### **Que fait-il ?**

- Nous utilisons Trello et Excel pour la gestion des tâches.
- Nous travaillons avec GitHub et Google Drive pour l'affectation des projets et des briefs.

## Définir le problème



Figure 10 : Schéma définir le problème

- **Manque d'harmonisation des critères d'évaluation** : Les formateurs soulignent la nécessité de critères clairs et uniformes pour garantir une évaluation objective et équitable des projets.
- **Dispersion des outils de suivi et de communication** : L'utilisation d'outils disparates (GitHub, Google Drive, Trello, Excel) complexifie la gestion des projets et la communication avec les apprenants.
- **Incohérence des niveaux de validation** : La multitude de niveaux de validation ("Basic", "Standard", "Premium", "Live coding", etc.) engendre confusion et manque de cohérence dans l'évaluation des compétences.
- **Difficultés à identifier les apprenants en difficulté** : Les formateurs manquent d'outils performants pour détecter rapidement les apprenants en besoin de soutien et suivre leur progression.



## Idéation



Figure 11 : Idéation

Créer une application web conviviale pour le centre Solicode qui permettra aux formateurs de :

- **Créer des projets de formation personnalisés:** Faciliter la création de projets adaptés aux besoins des apprenants, intégrant les compétences ciblées et permettant de gérer les ressources pédagogiques.
- **Affecter les projets aux apprenants:** Simplifier l'attribution des projets aux apprenants de manière efficace.
- **Suivre l'avancement des projets et la progression des apprenants:** Offrir un outil de suivi précis et un système de retour d'information clair et personnalisé.
- **Communiquer facilement avec les apprenants:** Permettre aux formateurs d'échanger des messages et des documents avec les apprenants.

## Diagramme de cas d'utilisation

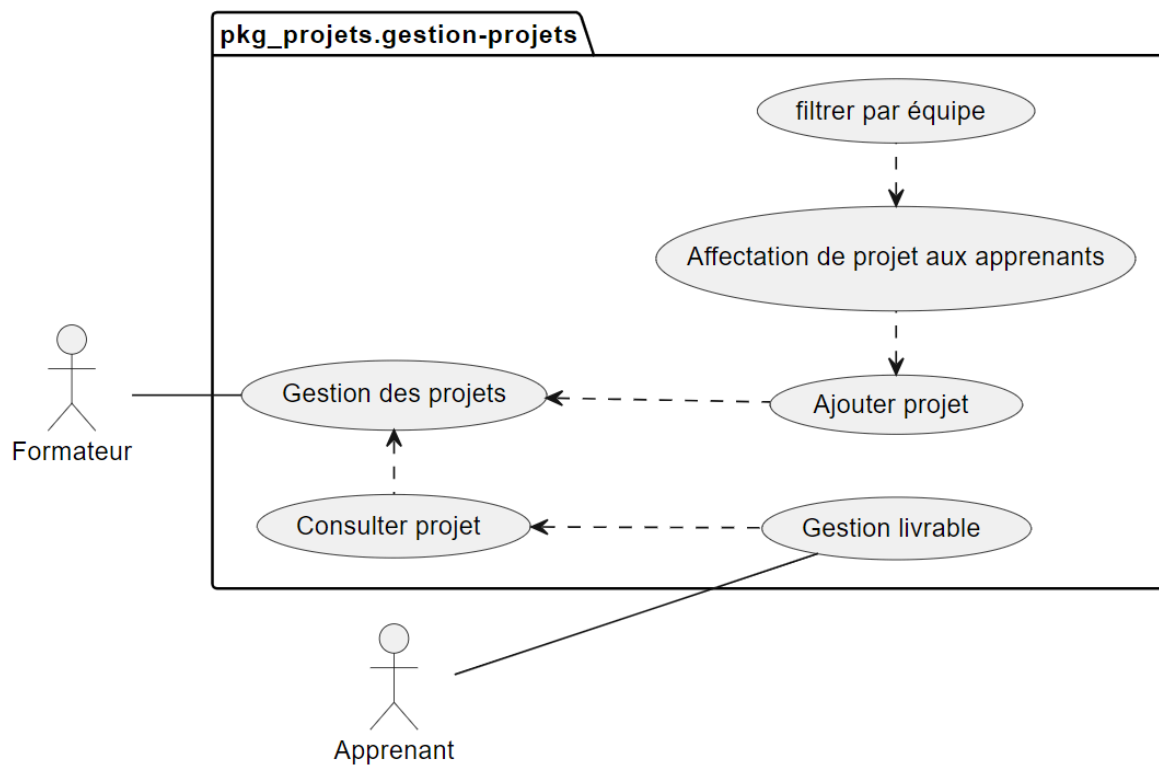


Figure 12 : Diagramme de cas d'utilisation

## Sprints backlog

### Sprint 1: Gestion de SOLI-LMS

- Gestion des projets.
- Gestion des réalisations .
- Gestion des validations.
- Gestion des ressources humaines .
- Gestion des compétences .
- Gestion des formations.
- Suivi d'avancement des apprenants

### Sprint 2: Évaluation en Temps Réel

- Fonctionnalités de suivi en temps réel des évaluations en cours.
- Envoi de notifications instantanées pour les mises à jour des évaluations.
- Possibilité d'ajuster les critères d'évaluation en cours d'évaluation.
- Personnalisation du barème de validation

## Branche technique

La branche technique fait référence à la partie du projet qui est axée sur les aspects techniques et de développement. Cette branche englobe les activités liées à la conception, la programmation, et la mise en œuvre des fonctionnalités et des composants du projet. Elle implique généralement l'utilisation d'outils de développement, la gestion des versions du code source, et la collaboration entre les développeurs. La branche technique joue un rôle essentiel dans la création d'un produit final de haute qualité, en veillant à ce que les spécifications techniques soient respectées et que les fonctionnalités soient correctement implémentées.

## Architecture de l'application

L'architecture de notre application suit le modèle Modèle-Vue-Contrôleur (MVC) avec une couche supplémentaire pour le Repository Pattern.

Le Modèle-Vue-Contrôleur est un modèle d'architecture logicielle qui divise une application en trois parties principales : le modèle (qui représente les données et la logique de l'application), la vue (qui affiche les données au client) et le contrôleur (qui gère les entrées utilisateur et coordonne les interactions entre le modèle et la vue).

Le Repository Pattern ajoute une couche supplémentaire d'abstraction qui permet de séparer la logique de stockage des données de la logique métier de l'application. Cette couche est responsable de l'accès aux données, de leur persistance et de leur récupération. Cette abstraction permet une meilleure maintenabilité et évolutivité de l'application.

En utilisant cette architecture, nous avons pu séparer les différentes responsabilités de l'application, ce qui facilite la maintenance et l'évolution de celle-ci.

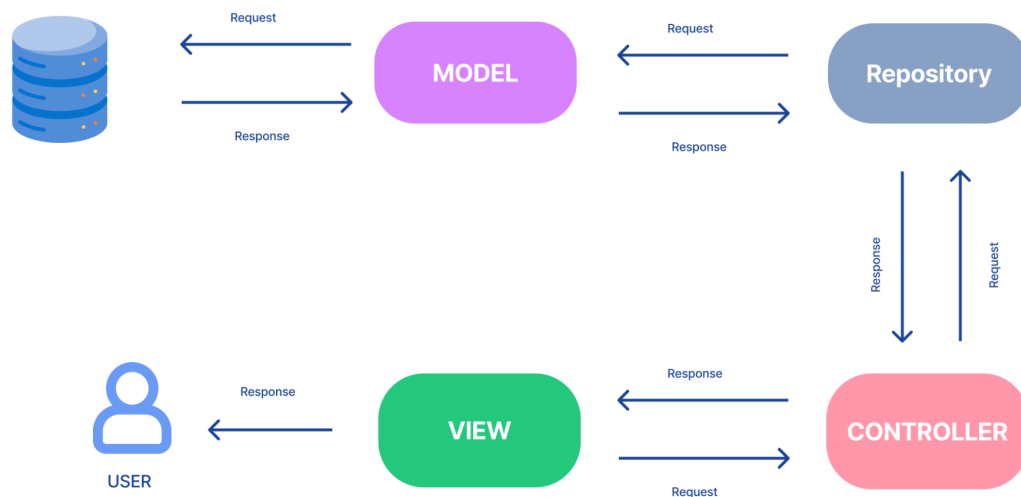


Figure 13 : Architecture de l'application

## Analyse technique

Pour le développement de l'application d'évaluation de SoliCode, nous avons opté pour des technologies clés afin d'optimiser le processus de création.

**Laravel** : a été choisi comme framework backend, exploitant ses fonctionnalités avancées pour faciliter le développement collaboratif. Son architecture MVC a permis une organisation claire du code, contribuant à l'efficacité de l'équipe.

**AdminLTE** : en tant que modèle d'application web, a été intégré pour la conception du tableau de bord d'administration. Son interface conviviale et ses fonctionnalités avancées ont joué un rôle essentiel dans la création d'une expérience utilisateur fluide et intuitive.

Ces technologies ont été déterminantes dans le succès de l'application d'évaluation de SoliCode, accélérant le processus de développement et assurant la création d'une interface utilisateur riche en fonctionnalités.

## Conception générique

### Prototype

#### Gestion des Projets (Sprint 1):

Le sprint 1 a permis de poser les bases de la gestion des projets. Nous avons implémenté les opérations CRUD (Créer, Lire, Mettre à Jour, Supprimer) de base pour les projets, permettant aux utilisateurs de créer, modifier et supprimer des projets. L'intégration des utilisateurs et des tâches aux projets a été réalisée, ainsi que la possibilité d'importer et d'exporter des données pour une meilleure manipulation. Des recherches dynamiques via Ajax ont été implémentées pour faciliter la recherche d'informations.

#### Gestion des Tâches (Sprint 2):

Le sprint 2, bien que prévu dans le planning initial, n'a pas encore été réalisé. Ce sprint sera dédié à la gestion des tâches au sein des projets. Il permettra la création, l'assignation, le suivi et la collaboration sur les différentes tâches.

## Conception

Ce chapitre présente le diagramme de classe de l'application ainsi que des maquettes représentant son interface utilisateur. Le diagramme de classe offre une visualisation structurée des différentes classes et de leurs relations au sein de l'application. Les maquettes, quant à elles, permettent de visualiser de manière concrète l'apparence et l'agencement de l'interface utilisateur, en mettant en évidence les fonctionnalités clés de l'application. Ces éléments sont essentiels pour comprendre l'architecture globale de l'application et pour guider le processus de développement en garantissant une conception cohérente et conviviale.

## Diagramme de classes

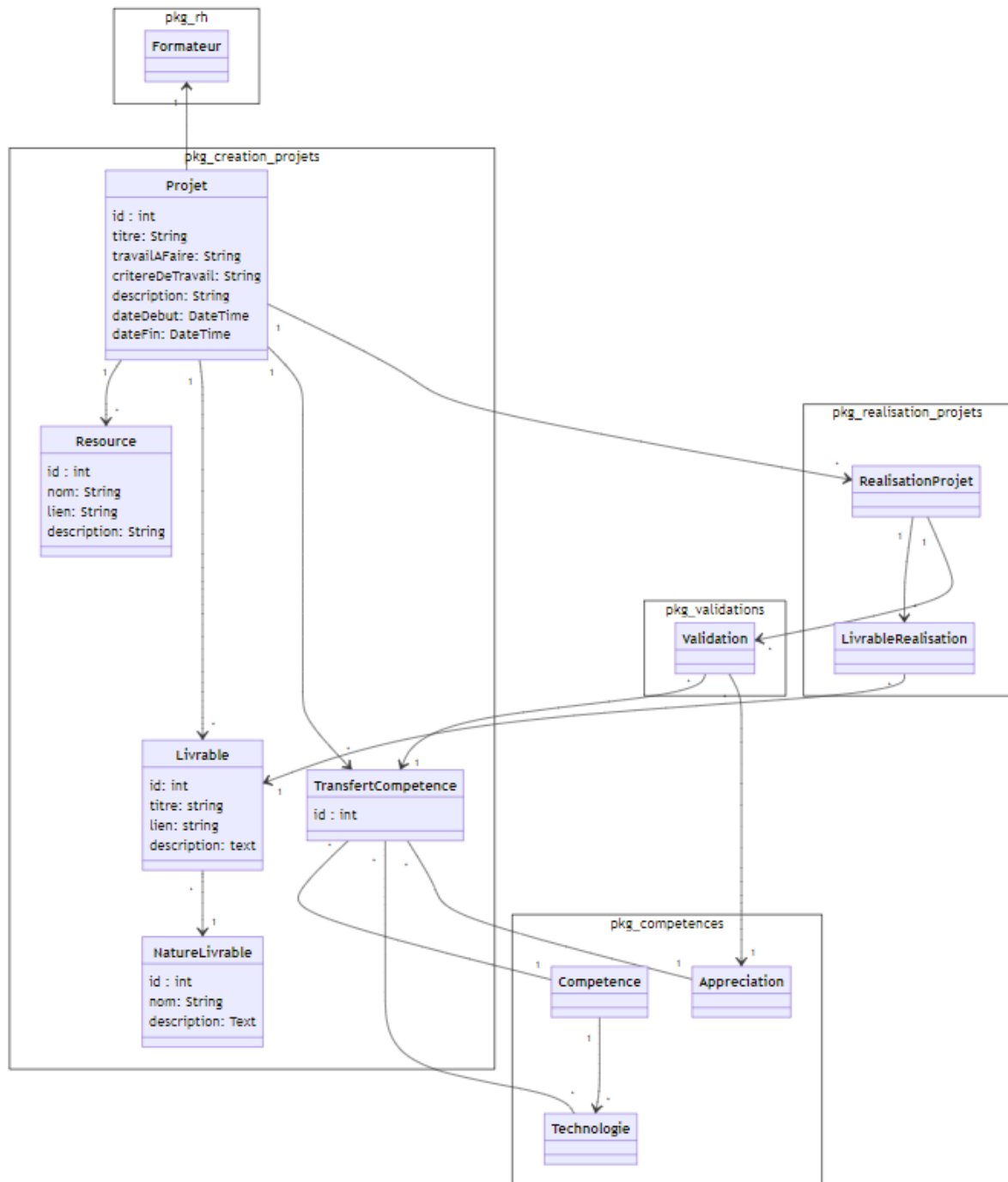


Figure 14 : Diagramme de classe pkg\_creation\_projets



## Diagramme des packages

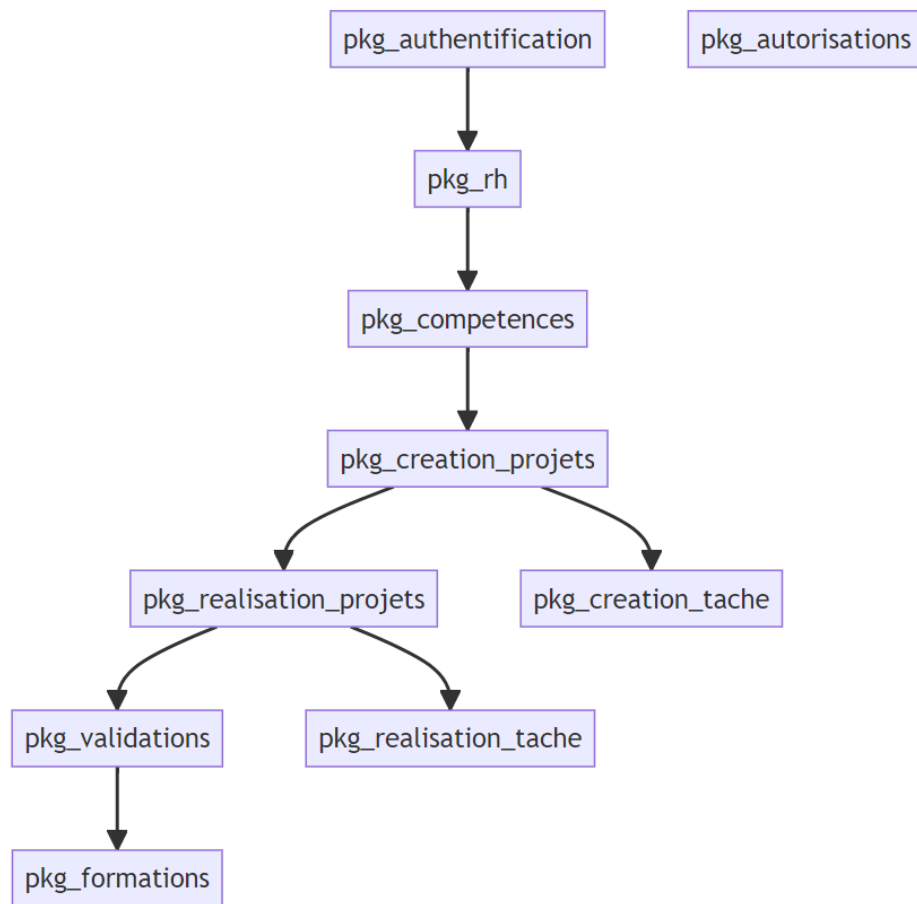


Figure 15 : Diagramme des packages

# Maquettes

## Maquette côté formateur

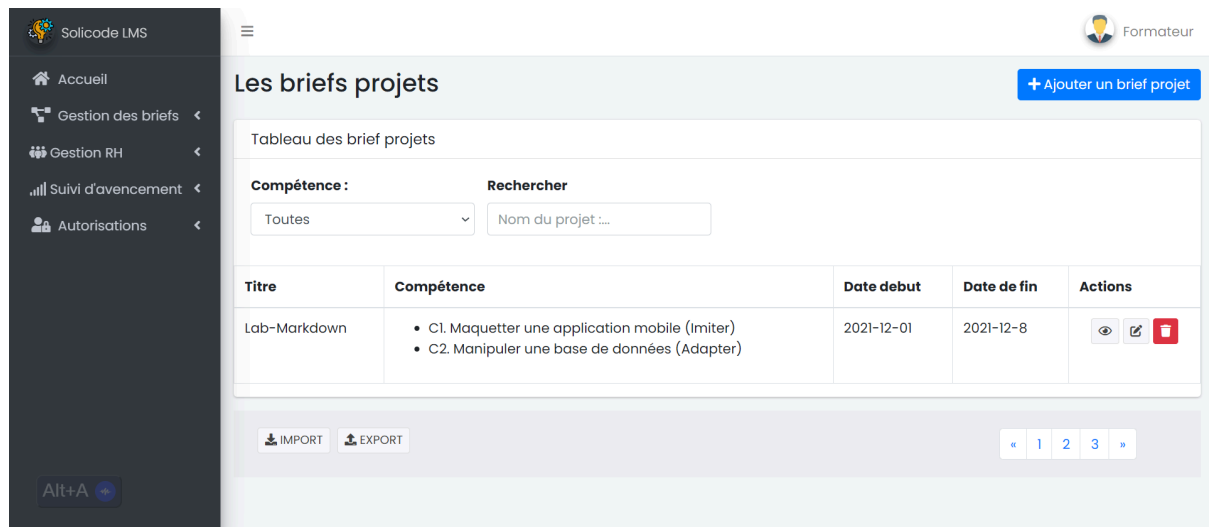


Figure 16 : Maquette de côté formateur

## Maquette côté apprenant

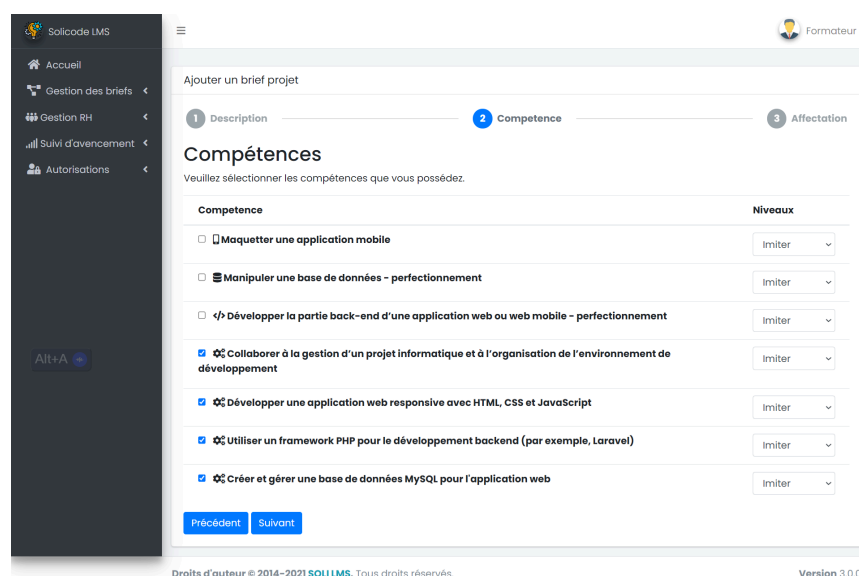



Figure 17 : Maquette de côté apprenant


Pour voir les autres maquettes, veuillez consulter l'annexe 1

# Réalisation

Ce chapitre met en valeur les technologies et outils utilisés dans la réalisation du projet gestion des évaluations. Il comprend également des captures d'écran de la réalisation du projet. Il permet de présenter les différentes technologies mises en œuvre et de visualiser le résultat obtenu.

## Technologies

	<b>HTML</b> signifie « <i>HyperText Markup Language</i> » que l'on peut traduire par « langage de balises pour l'hypertexte ». Il est utilisé afin de créer et de représenter le contenu d'une page web et sa structure
	<b>CSS</b> pour Cascading Style Sheets est un langage informatique utilisé sur Internet pour la mise en forme de fichiers et de pages <u>HTML</u> . On le traduit en français par feuilles de style en cascade.
	<b>JavaScript</b> est un langage de programmation de scripts essentiellement employé dans les pages web interactives et à ce titre est une partie essentielle des applications web.
	<b>PHP</b> , pour Hypertext Preprocessor, désigne un langage informatique, ou un langage de script, utilisé principalement pour la conception de sites web dynamiques.
	<b>MySQL</b> est un système de gestion de bases de données relationnelles. Il est distribué sous une double licence GPL et propriétaire.
	<b>L'UML</b> (Unified Modeling Language ou Langage de modélisation unifiée en français) est un langage graphique de modélisation informatique.
	<b>Laravel</b> est un framework web open-source écrit en PHP1 respectant le principe modèle-vue-contrôleur et entièrement développé en programmation orientée objet
<b>AdminLTE</b>	<b>AdminLTE</b> est un framework de conception d'interface utilisateur (UI) open source, basé sur Bootstrap. Il offre des modèles de tableau de bord et des composants prêts à l'emploi pour le développement rapide d'applications web

	<b>Bootstrap</b> est une collection d'outils utiles à la création du design de sites et d'applications web.
---	---

## Outils

	<b>Visual Studio Code</b> est un éditeur de code extensible développé par Microsoft pour Windows, Linux et macOS.
	<b>Github</b> est une entreprise de développement et services logiciels sise aux États-Unis. Github développe notamment la plateforme Github, l'éditeur de texte Atom ou encore la structure Electron
	<b>Git</b> est un logiciel de gestion de versions décentralisé. C'est un logiciel libre créé par Linus Torvalds, auteur du noyau Linux, et distribué selon les termes de la licence publique générale GNU version 2.
	<b>Apache</b> le serveur web open source de la Fondation Apache Software, est largement utilisé pour son agilité et sa stabilité.
	<b>Jira</b> est une plateforme logicielle de gestion de projet développée par Atlassian.
	<b>Jekyll</b> est un générateur de site statique open-source. Il convertit des fichiers texte simples en sites web statiques et blogs, offrant une solution rapide et sécurisée pour créer des sites sans la complexité des CMS dynamiques.
	<b>Mermaid</b> est un langage similaire au markdown pour générer des diagrammes, offrant simplicité et polyvalence. Il est largement utilisé pour créer différents types de graphiques et de diagrammes, y compris des organigrammes et des diagrammes de séquence.
	<b>PlantUML</b> est un outil basé sur du texte pour générer des diagrammes UML. Avec sa syntaxe intuitive, il permet de créer des diagrammes de classes, des diagrammes de cas d'utilisation, et plus encore, en faisant un choix populaire en génie logiciel pour visualiser des systèmes complexes.

## Conclusion

Le développement du package de création de projets pour Soli-LMS a été une expérience enrichissante. En travaillant sur cet outil, j'ai pu approfondir ma compréhension des besoins spécifiques du SoliCode Center et contribuer à la réalisation de sa mission d'offrir des formations de qualité supérieure, accessibles à tous.

L'objectif principal de ce projet était de simplifier la création de projets pour les formateurs, tout en assurant une flexibilité maximale et une intégration transparente avec les compétences visées. Grâce à ce travail, nous avons pu contribuer à améliorer l'efficacité des formations, accroître la qualité des contenus et renforcer la satisfaction des apprenants.

Ce projet a été l'occasion de développer des compétences techniques et de collaborer efficacement avec l'équipe du SoliCode Center. Je suis convaincu que ce package de création de projets constituera un outil précieux pour le centre et contribuera à l'atteinte de ses objectifs pédagogiques.

# Annexe 1: Maquette

Solicode LMS

Accueil

Gestion des briefs <

Gestion RH <

Suivi d'avancement <

Autorisations <

Ajouter un brief projet

1 Description

2 Competence

3 Affectation

Titre du brief

Prototype

Description

Paragraph

**B** *I*

Description du projet

Travail à faire

Paragraph

**B** *I*

Concevoir et développer un site Web responsive pour le client

Critère de validation

Paragraph

**B** *I*

Le site Web doit être entièrement responsive, respecter les meilleures pratiques en développement Web et répondre aux exigences du client.

Nom de livrable

Presentation

Ajouter un autre livrable

Date de début

01/01/2022

Date de fin

01/31/2022

Resources

Enter a link

Ajouter un lien

https://grafikart.fr/formations/laravel

Remove

https://laracasts.com/series/phpunit-testing-in-laravel

Remove

Reference

Enter a link

Ajouter un lien

https://www.figma.com/file/Aciw4FSMe0RsC3x1LH3R1/biblioth%C3%A8que-website?type=design&node-id=55%3A344&mode=design&t=bXJnh73IQoHkdKdj-1

Remove

https://m3.material.io/

Remove

Suivant

Droits d'auteur © 2014–2021 SOLI LMS. Tous droits réservés.

Version 3.0.0

Figure 18 : Création projet phase 1 (projet-livrables-ressources)

**Ajouter un brief projet**

1 Description — 2 Competence — 3 Affectation

☐ Tout cocher

- ☒ sarsri imrane
- ☒ Assaid Amina
- ☒ El ajoumi Mohammed aymane
- ☒ BEN NASAR ADNAN
- ☒ lamchatab amine
- ☒ Grain Reda
- ☒ zaani hamza
- ☒ Lharrak Adnan
- ☒ Achaou Hamid
- ☒ Boukhar Soufiane
- ☒ Bouik Hussein
- ☒ FAIZ SAFAA
- ☒ YASMINE DAIFANE
- ☒ Betroji Jalil

[Précédent](#) [Ajouter](#)

Droits d'auteur © 2014–2021 SOLI LMS. Tous droits réservés. Version 3.0.0

Figure 19: Création projet phase 3 (affectation du projet)

## Annexe 2: Planification

**pkg\_creation\_projets\_SoliLMS**

Backlog | Gantt | New view

Filter by keyword or by field

Title	Assignees	Status	Priorité	date debut	date fin
1 init-app #7	husseinbouik	Terminée	P0	Mar 26, 2024	Mar 27, 2024
2 CreationBriefProjet_base-donnees #28	husseinbouik	Terminée	P0	Apr 21, 2024	Apr 24, 2024
3 GestionRH_base-donnees #27	husseinbouik	Terminée	P0	Apr 21, 2024	Apr 25, 2024
4 Global_diagramme-classe-global #29	husseinbouik	Terminée	P0	Apr 21, 2024	Apr 26, 2024
5 CreationBriefProjet_diagramme-classe #92	husseinbouik	Terminée	P0	Apr 26, 2024	May 2, 2024
6 CreationBriefProjet_maquettage-spri... #30	husseinbouik	Terminée	P1	Apr 19, 2024	May 2, 2024
7 RealisationBriefProjet_maquettage-s... #31	husseinbouik	Terminée	P1	Apr 19, 2024	May 2, 2024

You can use **Control + Space** to add an item

Figure 20: Backlog du pkg\_creation\_projets