



## C2 - Projet de fin de formation

C-COD-260

# ETP

---

Mon propre projet

1,44



# ETP

nom du dépôt: codac-etp



- L'intégralité de vos fichiers sources, à l'exception de tous les fichiers inutiles (fichiers binaires, fichiers temporaires, fichiers obj,...), doit être incluse dans votre livraison.
- Tous les fichiers bonus (y compris un éventuel Makefile spécifique) doivent se trouver dans un répertoire nommé *prime*.
- Les messages d'erreur doivent être écrits sur la sortie d'erreur, et le programme doit ensuite se terminer avec le code d'erreur 84 (0 s'il n'y a pas d'erreur).

## INTRODUCTION

La pédagogie d'EPITECH privilégie un enseignement actif, est axée sur les projets et s'appuie sur l'expérimentation. Le projet est au cœur du processus d'apprentissage. C'est en recherchant des solutions à de nouveaux problèmes que vous avez acquis de nouvelles connaissances et une expérience enrichissante. Dans ce contexte, vous avez pu réaliser un travail personnel important et vous êtes aujourd'hui capable d'acquérir des compétences techniques de manière autonome et de créer un projet de A à Z. L'agilité et l'ouverture sur le monde professionnel sont au cœur de cette pédagogie. **Projet de formation (ETP)**. Vous devrez faire preuve de proactivité dans le choix du projet à réaliser en fonction des technologies proposées par le sujet et de ses contraintes imposées.

Vous devrez travailler en groupes de 2 à 4 de vos camarades de classe et de votre groupe (par exemple, groupe [nom du groupe]). **propriétaire** ou **client** sera associé à un autre groupe (appelé groupe **promoteur** ou **fournisseur**).

## PRÉSENTATION DU PROJET

Si vous êtes le chef de groupe, vous devrez rédiger les spécifications techniques du projet choisi et vous positionner comme responsable produit pour le groupe 2, qui réalisera votre projet en tant que prestataire de services. Et inversement.

### PLANIFICATION

- J1 matin : Spécifications fonctionnelles du responsable du groupe et validation du projet par l'équipe pédagogique
- Après-midi J1 : Examen des spécifications fonctionnelles et du choix des technologies par l'équipe de développement
- J2-J5 : Sprint 1
- J6-10 : Sprint 2



- J11 : Tests d'acceptation 1
- J11 : Réévaluation des spécifications fonctionnelles
- J11-J14 : Sprint 3
- J14 matin : Déploiement
- J14 après-midi : Tests d'acceptation 2
- J15 : Présentation de l'ETP

Toutes ces étapes doivent figurer sur votre outil d'organisation de groupe (par exemple, chaque étape devra être représentée par une carte des commentaires ajoutés pour les décisions prises lors de la révision).

Vous devez respecter les délais imposés par votre groupe de clients et communiquer avec lui selon le calendrier établi. Ce projet, solidement ancré dans le monde professionnel, vous permettra de mettre en valeur, lors de votre stage, votre expérience de développeur et de chef de projet, mais aussi de comprendre les besoins et les attentes de vos futurs clients. C'est pourquoi il est essentiel de vous investir pleinement dans ce projet professionnel et opérationnel.

Pour rappel, un sprint est composé de :

- \* Une revue de sprint au début du sprint en présence du responsable produit
- \* L'élection d'un Scrum Master à la fin de la revue de sprint pour le sprint à venir
- \* Un planning de sprints après l'élection du Scrum Master avec l'équipe de développement

\* Une réunion quotidienne, chaque journée commence avec l'équipe de développement

## VALIDATION DU PROJET

À la fin de la matinée du premier jour, votre travail doit être validé par l'équipe pédagogique et vous devez présenter :

- \* Les spécifications fonctionnelles de votre projet
- \* Une présentation de votre équipe de développement avec un nom d'équipe
- \* Une présentation de votre projet en 50 mots maximum (en français) est requise. Il s'agit de décrire votre projet de manière synthétique. Votre texte ne doit pas être technique ; il doit être rédigé dans un langage accessible à un public non spécialiste en informatique.

À la fin de la première journée, vous devez faire valider par l'équipe pédagogique :

- \* Les spécifications fonctionnelles de votre projet après discussion avec votre responsable produit
- \* Vos choix technologiques justifiés



Lors de la rédaction de ces documents, mettez-vous toujours à la place de ceux à qui ils sont destinés.



Vous accorderez une attention particulière à la qualité des documents soumis, tant du point de vue ergonomique qu'orthographique.



Tous vos documents doivent être soumis au format PDF et figurer dans votre dossier de travail lors de la présentation finale.



Seules vos spécifications seront utilisées pour évaluer le travail final de votre groupe de binômes.



L'équipe pédagogique est habilitée à modifier vos spécifications fonctionnelles jusqu'à la fin du premier jour si elle le juge nécessaire. De plus, l'équipe pédagogique est en mesure d'arbitrer vos désaccords avec votre groupe de travail.

## CONTRAINTES TECHNIQUES

---

- Vous devez uniquement choisir les technologies qui ont été abordées lors de la formation.
- Vous devez créer un backend avec une API

## SPÉCIFICATIONS FONCTIONNELLES

---

La spécification fonctionnelle décrit les fonctions d'un logiciel nécessaires à sa mise en œuvre. Elle décrit en détail comment les exigences seront prises en compte.

En plus de vos spécifications fonctionnelles, vous devez fournir :

- \* Un backlog produit
- \* Un schéma de conception (type UML, MySQL Workbench, etc.)
- \* Documentation API



## ORGANISATION DE GROUPE

---

Durant ce projet, vous devrez mettre en œuvre et respecter l'agilité telle que vous l'avez abordée lors de votre formation.

Vous devrez ensuite mettre en place et justifier un flux Git qui reflète l'organisation agile de votre groupe. Nous attendons au minimum :

- \* Une méthodologie KANBAN (trello)
- \* Un backlog produit avec planification de type poker
- \* 3 sprints par semaine

## DOCUMENTATION

---

Durant le projet, vous devrez créer une documentation destinée à l'utilisateur final, ainsi qu'une documentation technique pour les futurs développeurs qui reprendront le projet, c'est-à-dire dans ce cas précis, votre groupe de clients.

### DOCUMENTATION UTILISATEUR

---

La documentation utilisateur a pour but de fournir un guide du produit permettant aux utilisateurs de l'utiliser pleinement et en toute connaissance de cause. Son format dépend du projet et du type d'utilisateur. Veillez à ce qu'elle soit claire et accessible au plus grand nombre.



Le manuel de votre four à micro-ondes est un bon exemple de documentation utilisateur, même si nous attendons de votre part une réflexion plus approfondie sur l'ergonomie de l'interface.

### DOCUMENTATION TECHNIQUE

---

La documentation technique a pour but de décrire les blocs fonctionnels de votre code. L'objectif est de créer un guide du développeur permettant à tout nouveau membre de l'équipe de contribuer au projet en comprenant son état, sa structure et la procédure de contribution, sans avoir à demander les informations nécessaires. Cette documentation technique doit se présenter sous la forme d'annotations de votre code et d'un fichier README.md complet pour chaque partie de votre projet.



Avez-vous entendu parler de Doxygen ?



## INTÉGRATION CONTINUE

---

Au cours du développement du projet, vous devrez choisir et configurer une pile d'intégration continue comprenant au minimum :

- \* Tests unitaires
- \* Déploiement automatisé
- \* Tests d'acceptation



Pour l'intégration continue, connaissez-vous Travis, Docker, Jenkins et SonarQube ?



Pour les tests d'intégration, savez-vous comment utiliser les linters (type ESLint ou Credo en ligne de commande) et les formateurs (type PHPSniffer) ?

## CONCEPTION

---

L'ergonomie de votre site web est un aspect important à prendre en compte. Combien de boutons intégrer à vos pages ? Comment utiliser les espaces blancs ?

Vous disposez de différents outils pour garantir une expérience à la hauteur de vos attentes.

Pour ce projet, vous devrez justifier vos choix de conception.



Vous pouvez trouver des outils et des méthodes pour travailler ici : <https://www.designkit.org/methods>.



## PRIME

---

- Modèles d'applications (Adobe XD, Photoshop, etc.)
- Tests de qualité (Google Lighthouse, Dareboost, etc.)
- Utiliser une API externe (connexion Facebook, Google, etc.)
- Tests fonctionnels