

Concepteur · rice développeur · se d'applications



NOS VALEURS

Partage

Entraide

Solidarité

Implication

Remise en question

Confiance

L'ORGANISME DE FORMATION

Simplon.co

Cité Numérique
2 rue Marc Sangnier
33130 BEGLES

Contact :

Katy-Anna GIBBE
Chargée de projets formation
06 79 91 02 63
kgibbe@simplon.co

PUBLICS CONCERNÉS

- Toute personne inscrite comme demandeur.se d'emploi

PRÉ-REQUIS

- une expérience professionnelle en tant que développeur·se OU une formation de niveau Bac+2 en programmation informatique OU une auto-formation solide
- un goût pour le travail en équipe et la collaboration autour de projets
- une très forte motivation, à démontrer
- de la curiosité, de la créativité, une bonne expression à l'oral et à l'écrit, une représentation claire du métier... et bien sûr, l'envie de s'engager dans une formation intense !

DURÉE ET RYTHME

La formation dure 16 mois en alternance dont 679 heures en centre de formation au rythme d'1 semaine en formation pour 3 semaines en entreprise.

LE MÉTIER DE CONCEPTEUR·RICE D'APPLICATIONS

A l'issue de la formation, l'apprenant sera en capacité de mettre en œuvre les compétences du·de la concepteur·rice développeur·se d'applications dans un environnement DevOps.

Ce parcours de formation vise le titre professionnel [RNCP37873](#) de niveau 6 : **“Concepteur développeur d'applications”**.

Le/la concepteur·rice développeur·se d'applications est un·e informaticien·ne qui intervient sur l'ensemble de la chaîne de production d'une application : choix techniques, environnement de développement, conception et réalisation.

Le métier de concepteur·rice développeur·se d'applications s'articule alors autour de 3 activités principales :

- Développer une application sécurisée
- Concevoir et développer une application sécurisée organisée en couches
- Préparer le déploiement d'une application sécurisée

Pour réaliser ces activités, ce·tte professionnel·le, doit s'appuyer sur les compétences suivantes : *les compétences marquées d'un * sont déjà introduites dans le référentiel DWWM)*

- **C1.** Installer et configurer son environnement de travail en fonction du projet
- **C2.** Développer des interfaces utilisateur *
- **C3.** Développer des composants métier *
- **C4.** Contribuer à la gestion d'un projet informatique
- **C5.** Analyser les besoins et maquetter une application *
- **C6.** Définir l'architecture logicielle d'une application
- **C7.** Concevoir et mettre en place une base de données relationnelle *
- **C8.** Développer des composants d'accès aux données SQL et NoSQL
- **C9.** Préparer et exécuter les plans de tests d'une application
- **C10.** Préparer et documenter le déploiement d'une application
- **C11.** Contribuer à la mise en production dans une démarche DevOps

MOBILISER DES COMPÉTENCES TRANSVERSALES, FONDAMENTALES EN ENVIRONNEMENT PROFESSIONNEL

Les compétences transversales sont identifiées en situation d'apprentissage imitant la situation professionnelle, pour être mobilisées par les apprenants, évaluées et valorisées tout au long du parcours de formation, au même titre que les autres compétences professionnelles visées par la formation.

3 activités principales sont ainsi identifiées :

- Organiser le travail individuel et collectif en environnement professionnel.
- Adopter une démarche de résolution face à un problème.
- Communiquer à l'oral et à l'écrit en environnement professionnel

Pour réaliser ces activités, le professionnel du secteur numérique doit s'appuyer sur les compétences que nous avons identifiées dans les différentes activités professionnelles auxquelles nous formons les apprenants :

- **CT1.** Planifier le travail à effectuer individuellement et en équipe afin d'optimiser le travail nécessaire à l'atteinte de l'objectif visé, à l'aide des outils numériques adaptés.
- **CT2.** Contribuer au pilotage de l'organisation du travail individuel et collectif afin de faciliter la communication, la collaboration et la gestion des imprévus au sein de l'équipe
- **CT3.** Définir le périmètre d'un problème rencontré en adoptant une démarche inductive afin de permettre la recherche de solution
- **CT4.** Rechercher de façon méthodique des pistes de résolutions au problème rencontré afin de le résoudre
- **CT5.** Partager la solution adoptée en utilisant les moyens de partage de connaissance ou de documentation disponibles afin de contribuer au développement de la connaissance de l'entreprise.
- **CT6.** Présenter un travail réalisé en synthétisant ses résultats, sa démarche et en répondant aux questions afin de le restituer au commanditaire
- **CT7.** Se familiariser avec les codes et la culture propres à son environnement professionnel afin d'y faciliter son intégration
- **CT8.** Interagir dans un contexte professionnel de façon respectueuse et constructive pour favoriser la collaboration
- **CT9.** Faciliter un temps de travail collectif en assurant une communication constructive entre les participants dans un cadre de travail clair pour permettre l'implication de tous

DÉTAILS DU PROGRAMME DE FORMATION

Phase 0 - La prairie - 70 heures	6
Phase 1 - J'étends le périmètre fonctionnel d'une application - 161 heures	9
Phase 2 - Je pilote le développement d'une application par les tests - 140 heures	14
Phase 3 - Je fais évoluer une application multi-couches en intégration continue - 140 heures	20
Phase 4 - Projet final, préparation à la certification et clôture - 175 heures	25

Phase 0 - La prairie - 70 heures

“une période intensive pour s'outiller et s'engager dans sa formation”

Durée : 70 heures

Modalité : face à face pédagogique, en centre de formation

La Prairie est le premier temps de la formation où les apprenant-e-s prennent leurs marques. À la fin de La Prairie, les apprenant-e-s auront :

- compris le métier et surtout des compétences visées par le parcours de formation
- compris l'évaluation finale (certification) visée par le parcours de formation
- vécu une introduction à la pédagogie Simplon
- participé à l'émergence de la dynamique collective, de promotion

Les compétences visées pendant la Prairie sont :

- C1. Installer et configurer son environnement de travail en fonction du projet - *niveau 1-2*
- C2. Développer des interfaces utilisateur - *niveau 1-2*
- C3. Développer des composants métier - *niveau 1-2*
- C4. Contribuer à la gestion d'un projet informatique - *niveau 1*
- C5. Analyser les besoins et maquetter une application - *niveau 1-2*
- **CT3. Définir le périmètre d'un problème rencontré** en adoptant une démarche inductive afin de permettre la recherche de solution - *niveau 1 à 2*
- **CT4. Rechercher de façon méthodique des pistes de résolutions au problème rencontré** afin de le résoudre - *niveau 1 à 2*
- **CT5. Partager la solution adoptée** en utilisant les moyens de partage de connaissance ou de documentation disponibles afin de contribuer au développement de la connaissance de l'entreprise - *niveau 1*
- **CT6. Présenter un travail réalisé** en synthétisant ses résultats, sa démarche et en répondant aux questions afin de le restituer au commanditaire - *niveau 1*

Pour cela, la Prairie se découpe en trois temps forts :

1. Je découvre mon environnement professionnel de formation

Introduction aux modalités et aux objectifs de la prairie : durée, contenu, rythme, période intensive de découverte du programme de formation et des méthodes d'apprentissage/de formation.

Une première mise en situation professionnelle aborde la configuration des postes de travail et l'initiation aux différents outils professionnels :

- Git
- l'intégration continue
- les plateformes Cloud

- les outils pédagogiques de la formation professionnelle : Simplonline, Discord, etc.

2. Je découvre et comprends le référentiel de compétences

Je prends connaissance et je comprends le référentiel de compétences et le référentiel de certification, avec l'aide du/de la formateur-riche, une seconde mise en situation professionnelle, permettant notamment :

- l'introduction aux principes du développement et de l'algorithme
- l'initiation par la découverte des différentes activités du métier visé par la formation :
 - Conception d'applications
 - Développement web et natif
 - Intégration de tests unitaires et fonctionnels
 - Création de pipelines d'intégration continue
 - Mise en place d'environnement de développement commune avec Docker

Exemple de mise en situation professionnelle :

Développer une interface qui appelle des API web

Vous intégrez les équipes de développement de la fédération française de Surf, et vous êtes chargé de prototyper, de développer puis d'intégrer une fonctionnalité à ajouter au site : donner aux surfeurs les températures de l'air et de l'eau des spots de surf français. Cette fonctionnalité très demandée par les surfeurs internationaux sera intégrée aux endroits qui vous semblent le plus judicieux sous les deux formats (°C et K).

Votre mission est la suivante :

- planifier les tâches à réaliser
- formuler le besoin
- mettre à jour les diagrammes UML de l'application
- maquetter les pages à modifier pour ordinateur et pour mobile
- développer une reproduction responsive des pages choisies
- développer la fonctionnalité recherchée dans un nouvel embranchement
- tester votre production pour l'intégrer
- informer votre responsable de votre avancement, lui faire valider des propositions régulièrement, et lui fournir une documentation technique en fin de projet,

Ressources pédagogiques :

- votre veille préalablement réalisée sur les diagrammes UML et les outils associés
- le lien vers le dépôt du projet
- les interviews de surfeur.euses sur leur façon de consulter le site et leurs besoins liés aux informations de température
- la liste des spots de surf et des informations à faire figurer

Créer une API REST avec un CMS headless et intégrer l'API à un client web

La petite entreprise BioLegiTech vendeuse de machines d'analyses médicales, a besoin d'une solution temporaire pour aider ses équipes opérationnelles dans leurs activités. Elle fait appel à vous pour permettre aux équipes de comptabilité et de gestion client de visualiser et d'interagir avec les données dont elles ont besoin.

Votre mission est la suivante :

- formaliser les besoins fonctionnels des deux équipes (*features*)
- traduire les besoins en diagrammes UML
- créer une maquette et ses enchaînements pour chaque page proposée
- schématiser la base de données relationnelle de l'entreprise
- créer une reproduction de la base de données de l'entreprise avec un CMS headless
- développer l'interface web de chaque équipe avec un modèle en grilles
- développer les fonctions métier demandées à l'aide de l'API du CMS headless (type extraction d'un tableau, modification en batch d'informations en base de données..)
- tenir à jour chaque interlocuteur de l'avancement de sa production
- organiser des tests fonctionnels et des tests unitaires

Ressources pédagogiques :

- la veille préalablement réalisée sur les recommandations de sécurité côté navigateur
- un interview du service comptable et une liste de fonctionnalités précises du service de gestion
- des extraits de la base de données de l'entreprise : les tableaux excel des contacts client, l'inventaire excel des machines produites par l'entreprise, de leurs prix et des ventes de l'année passée, la liste des noms des employés et de leur équipe.

3. Module de contractualisation du parcours avec l'apprenant à Simplon

- la promotion réalise un "arbre de compétence" qui la suivra tout au long de la formation,
- je rédige mon projet professionnel et mon engagement - ma stratégie - pour l'atteindre,
- je restitue mon projet réalisé durant la Prairie (selon les modalités prévues par le-a formateur-riche),
- je prends acte de mon rôle d'apprenant à Simplon : projets, learning by teaching, quotidien, veilles...,
- je signe le règlement intérieur de Simplon.

A la fin de la prairie: [Atelier de sensibilisation à l'usage des IA génératives](#)

Tout au long de la formation - Valoriser les apprentissages de l'entreprise

À chaque retour en formation, les apprenants seront amenés à réaliser des retours d'expériences sur des problématiques rencontrés par les entreprises. Ils construiront ainsi une connaissance des contextes de l'entreprise, un savoir collectif et une habitude à travailler en équipe.

Ce qui implique :

- Construire un savoir partagé
- Comprendre les enjeux rencontrés en entreprise
- Savoir faire un retour d'expérience

Phase 1 - J'étends le périmètre fonctionnel d'une application - 161 heures

Durée : 161 heures, soit 4 semaines et 3 jours en centre de formation pour un total de 4 mois en entreprise

Modalité : en alternance, une semaine en centre de formation pour 3 semaines en entreprise

Maintenant que je suis capable de concevoir et de développer des applications web client, cette phase 1 me positionne dans le quotidien d'un.e concepteur.trice-développeur.euse d'applications full-stack. Je serai capable, à l'issue de cette phase, de concevoir et de développer toute la chaîne des composants liés à des fonctionnalités qui manipulent des bases de données relationnelles, en répondant à un besoin métier simple, à partir d'une demande formalisée.

Ce qui implique :

- ce qui a été vu pendant la prairie et la phase 1
- la compréhension d'un besoin formalisé en développement front-end et back-end
- la mise en place d'un environnement de test
- la conception et la mise à jour de l'architecture logiciel d'une application
- la conception et la gestion de bases de données relationnelles
- le développement d'une API et de composants d'accès aux données
- le développement de fonctionnalités simples dans un langage orienté objet
- l'intégration de ma production dans un environnement de versionnement

Compétences visées

- C1. Installer et configurer son environnement de travail en fonction du projet - *niveau 2*
- C2. Développer des interfaces utilisateur - *niveau 2*
- C3. Développer des composants métier - *niveau 2*
- C5. Analyser les besoins et maquetter une application - *niveau 1-2*
- C6. Définir l'architecture logicielle d'une application - *niveau 1-2*
- C7. Concevoir et mettre en place une base de données relationnelle - *niveau 2*
- C8. Développer des composants d'accès aux données SQL et NoSQL - *niveau 1-2*
- C9. Préparer et exécuter les plans de tests d'une application - *niveau 1*

- **CT1. Planifier le travail à effectuer individuellement et en équipe** afin d'optimiser le travail nécessaire à l'atteinte de l'objectif visé, à l'aide des outils numériques adaptés - *niveau 1 à 2*

- **CT2. Contribuer au pilotage de l'organisation du travail** individuel et collectif afin de faciliter la communication, la collaboration et la gestion des imprévus au sein de l'équipe - *niveau 1*
- **CT3. Définir le périmètre d'un problème rencontré** en adoptant une démarche inductive afin de permettre la recherche de solution - *niveau 2*
- **CT4. Rechercher de façon méthodique des pistes de résolutions au problème rencontré** afin de le résoudre - *niveau 2*
- **CT5. Partager la solution adoptée** en utilisant les moyens de partage de connaissance ou de documentation disponibles afin de contribuer au développement de la connaissance de l'entreprise - *niveau 2*
- **CT6. Présenter un travail réalisé** en synthétisant ses résultats, sa démarche et en répondant aux questions afin de le restituer au commanditaire - *niveau 2*
- **CT7. Se familiariser avec les codes et la culture propres à son environnement professionnel** afin d'y faciliter son intégration - *niveau 1*
- **CT8. Interagir dans un contexte professionnel** de façon respectueuse et constructive pour favoriser la collaboration - *niveau 2*

Activités techniques

Création et gestion de bases de données relationnelles

- Installation et configuration d'un système de gestion de bases de données (SGBD)
- Installation et configuration d'un environnement de test et d'un serveur de données en local
- Création et suppression d'une base de données
- Modification de tables et de relations en base de donnée
- Mise en place d'un processus de sauvegarde et restauration de la bases de données
- Développement de fonctions d'écriture, de modification, ou de suppression de donnés dans le langage du SGBD,
- Mise en place d'un système d'authentification pour sécuriser l'accès aux composants de base de de donnée
- Validation des entrées en base de données

Conception, création et exploitation d'API Rest

- Intégration et exécution de tests unitaires
- Configuration d'un framework de développement d'API
- Conception des points d'accès (endpoint)
- Création de points d'accès (endpoint) de l'API selon les principes REST
- Développement des modèles : fonctions d'écriture, de modification, ou de suppression de données dans un langage orienté objet
- Développement de points d'accès avancés (filtres, recherche, pagination des ressources)
- Développement d'une authentification pour l'accès aux composants de base de donnée

- Validation des entrées en base de données
- Développement d'une authentification à l'API
- Intégration et exécution de tests unitaires
- Exécution de tests de l'API

Conception et développement de composants back et front

- développement de composants métier avec les données API
- développement de composants d'interface utilisateur dynamique
- validation des entrées depuis l'interface et dans les composants serveur
- Gestion de la sécurité de l'application dans la partie serveur
- Suivi d'une procédure de démarche qualité à partir des exigences d'un cahier des charges
- Création de fichiers de persistance des données pour les échanges entre applications
- Intégration et exécution de tests unitaires

Activités transversales

- analyse d'un cahier des charges
- lecture d'un dossier technique ou de diagrammes fonctionnels
- choix de l'outil et modification d'une architecture logiciel
- choix de l'outil et modification de schémas de base de données
- résolution des problèmes et débogage de son code de manière structurée
- lecture d'un dossier technique et d'un diagramme de classes de type UML
- rédaction d'une documentation technique d'API
- suivi du modèle MVC d'architecture logicielle
- suivi de la documentation technique d'une architecture logicielle
- suivi d'un plan de tests
- rédaction d'une documentation technique
- mettre à jour un schéma de base de données
- analyse d'un cahier des charges
- maquettage d'une partie d'une application
- organisation de réunions de travail en français et en anglais
- planification de son action et information régulière à ses interlocuteurs
- versionnement de son code
- déboguer son code de manière structurée

Exemples de sujets de veille

- les failles de sécurité principales des applications web (XSS, CSRF..)
- les guide du top 10 OWASP
- L'API REST
- tests d'API
- les failles de sécurité côté serveur

Exemples d'outils envisagés

- SGBD : MySQL, MariaDB
- Git

- framework backend : Symphony, NodeJS, ExpressJS
- framework applicatif côté back-end type API Platform, Fastify, FastAPI...
- framework de mise en page : Bootstrap, Bulma, Semantic UI
- Postman pour l'API testing
- outil de gestion de projet : Trello, Jira, GitLab

Exemple de mise en situation professionnelle

Enrichir une base de données client

Vous êtes développeur fullstack pour le compte d'une agence web. Suite au cadrage avec un client pour enrichir ses données client (dans le respect du RGPD), vous êtes chargé de concevoir et mettre en place des évolutions dans sa base de données et son exploitation backend, ainsi que de faire évoluer les composants d'affichage liés à ces fonctionnalités

Votre mission est la suivante

- cadrer le besoin et concevoir une nouvelle version de la base de données
- mettre à jour les diagrammes UML et le schéma de la base de donnée
- ajouter des composants à l'interface utilisateur (formulaire p. ex) pour alimenter la base de données et pour afficher de nouvelles informations
- mettre à jour la base de données et ajouter des fonctions d'accès sécurisée aux nouvelles tables de la base de données
- tester en local les évolutions

modalités d'exécution : 3 jours en individuel

modalités d'évaluation

- Une revue de code sera réalisée en binôme
- la base de données et ses composants d'accès mis à jour seront présentés sous forme de démonstration au reste du groupe et au formateur.rice

ressources pédagogiques

- le cahier des charges des modifications à apporter, avec les diagrammes UML et les schémas de base de donnée actuels
- la base de données actuelle
- La documentation du langage de programmation utilisé et du connecteur SQL utilisé

Faire évoluer une API Rest et les fonctions qui l'utilisent

Un ancien client, travaillant dans le secteur financier, a contacté votre agence de développement web pour élargir un projet. Vous avez développé pour eux un système fusionnant plusieurs bases de données afin que les informations puissent être accessibles via une API par les différentes interfaces utilisateur qu'ils fournissent à leurs clients. Ils ont identifié un nouvel ensemble d'informations financières qu'ils aimeraient ajouter dans le système. Plus

précisément, les mesures des ratios d'efficacité, des ratios de liquidité, des ratios de levier, et des taux de rendement ainsi que l'indicateur de "moyenne mouvante".

Le client vous demande également de créer une interface de base consommant et visualisant les informations de test ajoutées au système.

Compte tenu du système actuel et des types de données des nouvelles informations à ajouter, votre mission est la suivante :

- synthétiser la demande et planifier les actions à mener
- modifier l'API pour prendre en compte les nouvelles informations
- modifier les tables de la base de données (métriques et indicateurs) afin de prendre en compte les nouveaux attributs
- modifiez les composants accédant à la base de données pour inclure les nouveaux indicateurs et métriques.
- suivre la même structure de fichiers et de composants que le système actuel.
- créer des données de test dans la base de données
- créer une interface consommant et modifiant les données
- mettre à jour la documentation : nouveaux schémas, nouvelles routes, diagramme de classes mis à jour
- tenir le client à jour de l'avancée de sa production

Modalités d'exécution : 3 jours en individuel

Modalités d'évaluation :

- la revue du code sera réalisée depuis un dépôt sur Git par le la formateur.rice
- le travail réalisé sera présenté à la promotion et au/à la formateur.rice par la présentation des schémas et des diagrammes UML mis à jour

Ressources pédagogiques :

- la base de données initiale, ainsi que les données à ajouter sont fournis par l'entreprise avec les fonctions d'accès aux données et l'API existante ainsi que trois niveaux d'accès aux nouvelles tables
- un diagramme de classes UML est fourni

Concevoir et intégrer un nouveau service

L'application de livraison de matériel de récupération ArtisansRecup souhaite faire évoluer sa plateforme et fait appel à vous pour apporter une nouvelle fonctionnalité : permettre aux artisans de s'enregistrer pour bénéficier de réductions sur les livraisons de matériaux.

Votre mission est la suivante :

- planifier les étapes de production et leur durée
- formaliser les fonctionnalités à développer
- mettre à jour un diagramme d'activité et un diagramme de classe en respectant les règles de nommage des classes existantes

- créer la/les tables à ajouter et leurs composants d'accès
- mettre à jour les contrôleurs et ajouter des contrôleurs
- maquetter et développer une page d'inscription par formulaire en validant les entrées
- maquetter puis développer l'évolution de composants d'interface pour intégrer les nouvelles données utilisateur
- intégrer des tests unitaires et fonctionnels

Modalités d'exécution :

- vous travaillerez en binômes
- vous planifiez la durée de votre production et prévoyez des temps de validation avec le client (nommages, fonctionnalité, maquettes, intégration)

Modalités d'évaluation :

- le suivi et les interactions client seront réalisées avec le/la formateur.rice
- la revue du code sera réalisée depuis un dépôt sur Git par le/la formateur.rice
- le travail réalisé sera présenté à la promotion et au/à la formateur.rice sous forme de démonstration de l'application, puis par la présentation des documents de conception.

Ressources pédagogiques :

- l'application architecturée selon un modèle MVC ou n-tiers est fourni dans un dépôt git
- le diagramme de classes, le schéma de la base de données, la documentation de l'application et le récapitulatif de la demande pour la nouvelle fonctionnalité sont fournis

Phase 2 - Je pilote le développement d'une application par les tests - 140 heures

Durée : 140 heures, soit 4 semaines en centre de formation pour un total de 4 mois en entreprise

Modalité : en alternance, une semaine en centre de formation pour 3 semaines en entreprise

Après la phase précédente, je suis en capacité de concevoir et de développer des applications web, dynamiques, responsives et qui exploitent des bases de données. Cette phase 2 me place alors dans la peau d'un.e concepteur.trice développeur.euse d'applications aux frameworks modernes dans une équipe agile. J'adopte une démarche de développement par le test et je produis des fonctionnalités architecturées et de manière itérative.

Ce qui implique :

- tout ce qui a été vu avant
- la mise en place et l'exécution d'un plan de test côté navigateur ou côté serveur
- le développement d'évolutions ou de composants d'interface
- le développement d'évolutions ou de composants de gestion des données
- la sécurisation des composants d'une application
- le suivi et l'animation de projets en mode agile

Compétences visées

- C1. Installer et configurer son environnement de travail en fonction du projet - *niveau 2*
 - C2. Développer des interfaces utilisateur - *niveau 2*
 - C3. Développer des composants métier - *niveau 2*
 - C4. Contribuer à la gestion d'un projet informatique - *niveau 2*
 - C5. Analyser les besoins et maquetter une application - *niveau 2*
 - C6. Définir l'architecture logicielle d'une application - *niveau 2-3*
 - C7. Concevoir et mettre en place une base de données relationnelle - *niveau 2-3*
 - C8. Développer des composants d'accès aux données SQL et NoSQL - *niveau 2-3*
 - C9. Préparer et exécuter les plans de tests d'une application - *niveau 1-2*
 - C11. Contribuer à la mise en production dans une démarche DevOps - *niveau 1-2*
-
- **CT1. Planifier le travail à effectuer individuellement et en équipe** afin d'optimiser le travail nécessaire à l'atteinte de l'objectif visé, à l'aide des outils numériques adaptés - *niveau 2*
 - **CT2. Contribuer au pilotage de l'organisation du travail** individuel et collectif afin de faciliter la communication, la collaboration et la gestion des imprévus au sein de l'équipe - *niveau 2*
 - **CT3. Définir le périmètre d'un problème rencontré** en adoptant une démarche inductive afin de permettre la recherche de solution - *niveau 2 à 3*
 - **CT4. Rechercher de façon méthodique des pistes de résolutions au problème rencontré** afin de le résoudre - *niveau 2 à 3*
 - **CT5. Partager la solution adoptée** en utilisant les moyens de partage de connaissance ou de documentation disponibles afin de contribuer au développement de la connaissance de l'entreprise - *niveau 2*
 - **CT6. Présenter un travail réalisé** en synthétisant ses résultats, sa démarche et en répondant aux questions afin de le restituer au commanditaire - *niveau 2*
 - **CT7. Se familiariser avec les codes et la culture propres à son environnement professionnel** afin d'y faciliter son intégration - *niveau 2*
 - **CT8. Interagir dans un contexte professionnel** de façon respectueuse et constructive pour favoriser la collaboration - *niveau 2*
 - **CT9. Faciliter un temps de travail collectif** en assurant une communication constructive entre les participants dans un cadre de travail clair pour permettre l'implication de tous - *niveau 2*

Activités techniques

Mise en place et exécution de plans de tests

- définir un plan de test : objectifs, types de tests à assurer, coverage par type de tests...
- intégration des tests en fonction du plan de test
- configuration de conteneurs pour la mise en place d'un environnement de développement et de test
- exécution de tests dans un environnement conteneurisé

Développement, amélioration et sécurisation de composants en développement test-driven

- choix et configuration d'un framework coté client
- correction de problèmes par les tests et développement de composants
- intégration des composants développés ou améliorés
- choix et configuration d'un framework côté serveur
- intégration des composants métier développés ou améliorés
- intégration d'un système multilingues à l'application
- sécurisation des composants d'interface
- amélioration de l'architecture d'une application
- choix et exploitation d'un patrons de conception (design patterns)
- intervention sur la gestion de bases de données relationnelles
 - sécurisation des accès à une API : CORS, Authentification avec JWT, limite d'usage (global et par point d'accès).
 - script de création de base de données
 - mise en place d'un processus de sauvegarde et restauration de la bases de données
 - programmation de procédures stockées avec SQL et PostgreSQL
 - programmation de déclencheurs avec SQL et PostgreSQL
- intervention sur les appels de bases de données non relationnelles
- validation de la sécurité des dépendances utilisées
- intégration de traitements sur des données transactionnelles

Activités transversales

- mise à jour d'un plan de tests
- organisation des tests de manière structurée
- documentation des tests réalisés et proposition de correctifs
- documentation de son code, et de son action de manière structurée et systématique (*commentaires, readme, comit*)
- gestion d'une fusion de branche à partir de résultats de tests
- rédaction d'un plan de test pour des composants d'interface
- planification et suivi de tâches de développement dans un outil de gestion de projet agile
- organisation et suivi de rituels agiles
- gestion des embranchements et des intégrations de composants
- respect des critères de qualité d'un cahier des charges
- Prise en compte d'un rapport d'audit et des corrections à réaliser
- intervention sur une architecture avancée : n-tiers, MVC, microservices, serverless, etc.

- rédaction d'un plan de test pour des composants coté serveur
- planification et suivi de tâches de développement dans un outil de gestion de projet agile
- organisation et suivi de rituels agiles
- gestion des embranchements et des intégrations de composants
- respect des critères de qualité d'un cahier des charges

Exemples de sujets de veille

- standard des tests fonctionnels et d'intégration
- les tests de performance avec Apache JMeter
- tests d'intégration ou end-to-end
- les guides de tests de sécurité (p. ex OWASP)
- tests fonctionnels et de qualité lié au développement côté serveur

Exemples d'outils envisagés

- Git, GitLab,
- Docker & Dockerfiles,
- Trello, Jira, GitLab ticketing
- Bibliothèque d'intégration de test : Jest, Pytest, Cypress...
- Front-end framework de mis en page type Bootstrap, Bulma, Semantic UI
- Front-end framework moderne type React, Vue.js ..
- Framework applicatif côté back-end: Laravel, Symfony, ExpressJS
- Framework de tests back-end : phpunit, pytest, Jest,...

Exemple de mise en situation professionnelle

Concevoir et développer les tests d'une application

Vous intégrez une équipe agile de l'entreprise chargée de répertorier et de maintenir les centrales nucléaires françaises. Pour bien comprendre ses fonctionnalités, vous commencez votre mission par le suivi du plan de tests de la *feature* tout juste développée.

Votre mission est la suivante :

- Récupérer la synthèse des tests d'une fonctionnalité précédente, et rédiger le plan de test de la *feature* à tester
- Mener les tests du plan de tests
- Générer une synthèse des tests et recommander ou non la fusion de la *feature* avec l'application

Modalités d'exécution :

- vous avez 1 jours pour préparer plan de tests et l'environnement de développement et de test et 2 jours pour intégrer les tests et faire un rapport de test
- vous fonctionnez en équipes de trois et échangez les plans de tests que vous rédigez avec ceux que vous exécutez

Modalités d'évaluation :

- les synthèses de chaque plan de tests seront comparés en équipe pour conclure sur la ou les approches convaincantes
- si plusieurs équipes testent des *features* différentes, une présentation comparative des plans de tests sera organisée en grand groupe

Ressources pédagogiques :

- la *feature* à tester sera placée dans une branche de tests, et le plan de tests d'une précédente *feature* sera fournie,
- le résultat de votre veille préalablement mené sur Docker et la conteneurisation

Tester et améliorer les composants d'interface d'une application

L'application EconoSoft3.0 vient de changer tout son staff de développeur.euses, elle a recruté votre équipe pour améliorer son interface web de manière rigoureuse et systématique. Elle vous confie l'application dans un framework de front moderne (Angular, VueJS ..) et les accès à son application de ticketing.

Votre mission est la suivante :

- installer les environnements de travail de chacun
- configurer des conteneurs pour l'environnement de test
- définir en équipe votre processus de développement, test, livraison,

- définir en équipe vos règles de nommage et de documentation (par exemple en créant l'UML de classes de l'application)
- planifier les tâches et les réaliser en suivant les rituels agiles
- mener un sprint avant de livrer une nouvelle version au client.

Parmi les *features* et les bugs enregistrés vous trouverez : des formulaires aux entrées non sécurisées, des cas de site trop lent lors d'une action utilisateur particulière, des enchaînements de page non cohérents, des problèmes d'affichage dans un navigateur particulier, la version responsive non lisible sur smartphone, des informations à ajouter pour un type de profil ...

Modalités d'exécution :

- vous travaillez en équipe agile, en vous répartissant les fonctionnalités et les tickets à réaliser. Chacun de vous aura :
 - développé un nouveau composant avec ses tests unitaires,
 - réglé un problème existant avec ses tests unitaires,
 - suivi le plan de tests d'une fonctionnalité,
 - mené des tests de sécurité
 - accepté la fusion d'une fonctionnalité à partir d'une synthèse de tests
- le client attend une documentation organisée, des synthèses de tests, et une nouvelle version intégrée de l'application à envoyer en production

Modalités d'évaluation :

- le.la formateur.rice organise un point d'étape à l'occasion de certains rituels agiles,
- l'application intégrée et la documentation sont partagés directement sur Git et seront passés en revue par le.la formateur.rice

Ressources pédagogiques :

- l'application à améliorer et les *features*
- le résultat de votre veille sur les frameworks front-end modernes

Tester et améliorer les composants serveur d'une application

L'application MasterCrafting vient de changer tout son staff de développeur.euses, elle a recruté votre équipe pour améliorer le covering, la sécurité de la partie serveur de son application web. Elle vous confie l'application dans un framework de backend et les accès à son logiciel de ticketing.

Votre mission est la suivante :

- installer les environnements de travail de chacun
- configurer des conteneurs de test et d'intégration
- définir en équipe votre processus de développement, test, intégration,

- définir en équipe vos règles de nommage et de documentation (par exemple en créant l'UML de classes de l'application)
- planifier les tâches et les réaliser en suivant les rituels agiles
- mener un sprint avant de livrer une nouvelle version au client.

Parmi les *features* et les bugs enregistrés vous trouverez : un utilisateur qui n'arrive pas à se connecter avec son mot de passe, l'affichage des données de quelqu'un d'autre, l'interfaçage de l'application avec des applications extérieures, l'automatisation de procédures de sauvegarde, l'ouverture du site à de nouveaux profils d'utilisateurs, la progression de la mise en place d'une nouvelle contrainte avec les anciens utilisateurs, une faille de sécurité détectée à supprimer en urgence

Modalités d'exécution :

- vous travaillez en équipe agile, en vous répartissant les fonctionnalités et les tickets à réaliser. Chacun de vous aura :
 - développé un nouveau composant avec ses tests unitaires
 - réglé un problème existant à partir du rapport d'exécution des tests unitaires
 - organisé des tests automatisés de points de terminaison d'API
 - suivre le plan de tests d'une fonctionnalité
 - mené des tests de sécurité (static, fuzzing...)
 - accepté la fusion d'une fonctionnalité à partir d'une synthèse de tests
- le client attend une documentation organisée, des synthèses de tests, et une nouvelle version intégrée de l'application à envoyer en production

Modalités d'évaluation :

- le la formateur.ice organise un point d'étape à l'occasion de certains rituels agiles,
- l'application intégrée et la documentation sont partagés directement sur Git et seront passés en revue par le la formateur.ice

Ressources pédagogiques :

- l'application à améliorer, les bases de données, et les *features*, l'entreprise vous fournit aussi le patron de conception qu'il souhaite respecter

Phase 3 - Je fais évoluer une application multi-couches en intégration continue - 140 heures

Durée : 140 heures, soit 4 semaines en centre de formation pour un total de 4 mois en entreprise

Modalité : en alternance, une semaine en centre de formation pour 3 semaines en entreprise

A l'issue de la phase précédente, je conçois, je teste et je développe les fonctionnalités d'applications en mode agile avec des langages de programmation orientés objet. Cette phase 3 me donne alors la posture de concepteur.trice développeur.euse d'applications en intégration continue. A l'issue de cette phase, je serai en mesure d'intégrer une équipe agile pour identifier,

concevoir et développer des évolutions de performance d'applications multi-couches en suivant un pipeline d'intégration continue.

Ce qui implique :

- tout ce qui a été vu avant
- la mise en place et l'exécution d'un pipeline d'intégration continue
- l'exécution de tests et l'amélioration de la sécurité des composants d'une application
- l'exécution de tests et l'amélioration de la performance des composants d'une application

Compétences visées

- C1. Installer et configurer son environnement de travail en fonction du projet - *niveau 2-3*
- C2. Développer des interfaces utilisateur - *niveau 2-3*
- C3. Développer des composants métier - *niveau 2-3*
- C4. Contribuer à la gestion d'un projet informatique - *niveau 2*
- C5. Analyser les besoins et maquetter une application - *niveau 2-3*
- C6. Définir l'architecture logicielle d'une application - *niveau 2-3*
- C7. Concevoir et mettre en place une base de données relationnelle - *niveau 2-3*
- C8. Développer des composants d'accès aux données SQL et NoSQL - *niveau 2-3*
- C9. Préparer et exécuter les plans de tests d'une application - *niveau 2-3*
- C10. Préparer et documenter le déploiement d'une application - *niveau 1-2*
- C11. Contribuer à la mise en production dans une démarche DevOps - *niveau 2-3*
- **CT1. Planifier le travail à effectuer individuellement et en équipe** afin d'optimiser le travail nécessaire à l'atteinte de l'objectif visé, à l'aide des outils numériques adaptés - *niveau 3*
- **CT2. Contribuer au pilotage de l'organisation du travail** individuel et collectif afin de faciliter la communication, la collaboration et la gestion des imprévus au sein de l'équipe - *niveau 2 à 3*
- **CT3. Définir le périmètre d'un problème rencontré** en adoptant une démarche inductive afin de permettre la recherche de solution - *niveau 3*
- **CT4. Rechercher de façon méthodique des pistes de résolutions au problème rencontré** afin de le résoudre - *niveau 3*
- **CT5. Partager la solution adoptée** en utilisant les moyens de partage de connaissance ou de documentation disponibles afin de contribuer au développement de la connaissance de l'entreprise - *niveau 3*
- **CT6. Présenter un travail réalisé** en synthétisant ses résultats, sa démarche et en répondant aux questions afin de le restituer au commanditaire - *niveau 3*
- **CT7. Se familiariser avec les codes et la culture propres à son environnement professionnel** afin d'y faciliter son intégration - *niveau 3*

- **CT8. Interagir dans un contexte professionnel** de façon respectueuse et constructive pour favoriser la collaboration - *niveau 3*
- **CT9. Faciliter un temps de travail collectif** en assurant une communication constructive entre les participants dans un cadre de travail clair pour permettre l'implication de tous - *niveau 3*

Activités techniques

Configuration et mise en place d'un pipeline CI

- choix et configuration de l'environnement d'intégration continue
- choix et configuration de l'outil de versionnement du code source
- configuration de conteneurs
- développement et exécution des scripts d'intégration continue
- paramétrage et exécution d'un pipeline d'intégration continue

Conception, développement, sécurisation et optimisation des composants front et back intégrés en continu

- choix et configuration d'un framework côté client
- choix et configuration d'un framework côté serveur
- développement, exécution et mises à jour de tests
- développement de composants d'interface avec les bibliothèques graphiques adaptées et les mécanismes asynchrones
- développement de composants du back-end (serveur/API)
- développement de composants d'accès aux données (modèles, entités, ...)
- mise en place de processus de sauvegarde et restauration de la bases de données
- sécurisation de l'application :
 - identification des besoins de sécurité de l'application
 - choix et exploitation de patrons de sécurité (security pattern)
 - sécurisation des composants d'interface
 - sécurisation de la liaison entre le client et les services back-end de l'application (CORS, HTTPS...)
 - sécurisation de la liaison avec le SGBD
- optimisation de l'application :
 - optimisation de la performance des composants d'interface
 - optimisation de l'architecture d'une application
 - gestion des dépendances de l'application à l'aide d'un gestionnaire de dépendances
 - création de vues
 - optimisation d'un modèle de données et d'une base de données : index, clés, types, relations, moteur de données, etc.
- mise en œuvre de procédures qualité

Activités transversales

- interprétation des rapports d'intégration continue
- gestion de fusion de branche à partir de résultats de tests
- mise à jour d'un plan de tests

- documentation de son code, et de son action de manière structurée et systématique (*commentaires, readme, comit*)
- utilisation et maintien à jour d'une documentation en anglais
- planification et suivi de tâches de développement dans un outil de gestion de projet agile
- organisation et suivi de rituels agiles
- gestion des embranchements et des intégrations de composants

Exemples de sujets de veille

- Sécurité dans le DevOps
- la mesure de performance côté navigateur
- guides de tests de sécurité côté navigateur
- les patterns de sécurité,
- les évolutions technologiques liées au déploiement

Exemples d'outils envisagés

- Git,
- Docker, Docker compose
- GitLab, Github actions,
- front-end framework moderne type React, Vue.js, Angular
- framework applicatif côté back-end: Laravel, Symfony, ExpressJS
- framework de test type Jest, Cypress et outils spécifiques au front-end framework
- Trello, Jira, GitLab ticketing,

Exemple de mise en situation professionnelle

Mettre en place un pipeline d'intégration continue (CI)

Le pôle de votre entreprise est chargé de faire évoluer la chaîne de mise en œuvre de services existants et de nouveaux services vers une nouvelle chaîne CI/CD. A partir des pipelines déjà mis en place pour d'autres services, vous avez pour mission de :

- migrer les procédures Legacy d'un nouveau service vers la nouvelle chaîne CI/CD,
- exécuter le pipeline pour vérifier son fonctionnement,
- vérifier la qualité du code
- faire des recommandations de tests et de documentation
- organiser des scans de sécurité
- documenter le pipeline et le partager avec l'équipe
- tester le pipeline avec plusieurs applications de l'équipe

Modalités d'exécution : production individuelle

Modalités d'évaluation : le pipeline partagé sera partagé une plateforme CI/CD comme GitLab ou GitHub actions et transmis avec sa documentation au.a à la formateur.rice

Optimiser la performance d'une interface

L'agence de location de voitures RentTheCars fait appel à vous pour optimiser les performances de son site.

Votre mission est la suivante :

- réaliser des tests de charges et de sécurité côté navigateur
- planifier les améliorations de structure, d'architecture et/ou de fonctionnalités côté navigateur
- mettre en place et suivre un pipeline d'intégration continue
- développer les améliorations
- documenter son action -

Modalités d'exécution : individuel, une intégration en environnement de pré-production est attendue à chaque évolution de composant passé par le pipeline d'intégration.

Modalités d'évaluation : les propositions d'amélioration ainsi que les résultats de tests seront documentées et partagées avec l'application au. a la formateur.rice

Optimiser la performance des composants serveur d'une application

L'agence de vente en ligne de vêtements d'occasion fait appel à votre équipe pour améliorer les performances et la sécurité de son application côté serveur.

Votre mission est la suivante :

- mettre en place un plan de tests fonctionnel et le réaliser
- réaliser des tests de charges et de sécurité côté serveur
- concevoir des améliorations de code, d'architecture et/ou de fonctionnalités
- mettre en place un pipeline commun d'intégration continue
- planifier les tâches et les réaliser en suivant les rituels agiles
- développer les améliorations et les intégrer en continue à l'environnement de pré-production
- documenter son action et justifier auprès du client des décisions d'optimisation
- documenter la procédure et les tests que l'entreprise devra réaliser pour déployer l'application

Modalités d'exécution :

- vous travaillez en équipe agile, en vous répartissant les tâches à réaliser.
- le client attend une documentation organisée, des synthèses de tests, et une nouvelle version intégrée de l'application à envoyer en pré-production.

Modalités d'évaluation :

- le.la formateur.rice organise un point d'étape à l'occasion de certains rituels agiles,

- l'application intégrée et la documentation sont partagés directement sur Git et seront passés en revue par le/la formateur.rice

Phase 4 - Projet final, préparation à la certification et clôture - 175 heures

Durée : **175 heures** en centre de formation, soit 5 semaines en centre de formation pour un total de 5 mois en entreprise

Modalité : **en alternance**, *une semaine en centre de formation pour 3 semaines en entreprise*

Préparation des certifications - Projets chef d'oeuvre et cas pratique

A travers le chef d'oeuvre les apprenants ont l'opportunité de mettre en pratique la plupart des compétences acquises au cours de la formation afin de les mettre en valeur à travers un projet concret de leur choix. Ce projet est aussi la modalité d'évaluation principale pour le passage du titre professionnel Concepteur.trice Développeur.se d'Applications.

Le chef d'oeuvre est un projet concret qui est imaginé, conçu et réalisé par l'apprenti. Il passe par plusieurs étapes complètes à travers lesquelles l'apprenti perfectionnera ses compétences.

Appui à la réalisation du chef d'oeuvre et passage certification

- **définition du projet chef d'oeuvre** : l'apprenant imagine et définit précisément le cadre du projet qu'il souhaite réaliser. Il s'agit de préparer un cahier des charges, de définir les fonctionnalités du projet.
- **réalisation du projet** : une fois validé, l'apprenant réalise le projet en mettant en pratique les compétences qu'il a acquises.
- **rédaction du rapport** : elle se fera au fil des avancées
- **rédaction du dossier professionnel** et passage en revue des ECF
- **préparation de la soutenance et de la démonstration pour le passage du titre professionnel** : à la fin du projet l'apprenant est amené à effectuer une présentation plus la démonstration de son chef d'oeuvre devant une assemblée, constituée de ses camarades, formateurs et potentiellement de professionnels externes.

Clôture du parcours de formation

- Bilan pédagogique individuel
- Session de retour d'expérience et d'évaluation du dispositif de formation

Résumé des exemples de situations professionnelles

Phase 0 - La prairie - 70 heures

Développer une interface qui appelle des API web

Créer une API REST avec un CMS headless et intégrer l'API à un client web

Phase 1 - J'étends le périmètre fonctionnel d'une application - 161 heures

Enrichir une base de données client

Faire évoluer une API Rest et les fonctions qui l'utilisent

Concevoir et intégrer un nouveau service

Phase 2 - Je pilote le développement d'une application par les tests - 140 heures

Concevoir et développer les tests d'une application

Tester et améliorer les composants d'interface d'une application

Tester et améliorer les composants serveur d'une application

Phase 3 - Je fais évoluer une application multi-couches en intégration continue - 140 heures

Mettre en place un pipeline d'intégration continue (CI)

Optimiser la performance d'une interface

Optimiser la performance des composants serveur d'une application

Phase 4 - Projet final, préparation à la certification et clôture - 175 heures

LA PÉRIODE D'ALTERNANCE

La pédagogie active promue par Simplon se traduit - entre autres - par l'implication de l'entreprise pendant la formation.

L'entreprise contribue à la formation des apprenants en les recevant pendant une période d'alternance. Elle est toujours suivie d'un retour en formation pour en acter les apprentissages.

On demande à l'apprenant de réaliser à petite échelle ce qu'il devra faire tout au long de sa vie professionnelle : de la recherche de son entreprise/job, à la compréhension de sa valeur pour une organisation, en passant par l'identification des rapports hiérarchiques.

Ce qui implique :

- la compréhension du contexte de l'entreprise
- la compréhension de sa mission dans l'organisation de l'entreprise
- la compréhension ce qu'on peut apporter à l'entreprise
- l'identification d'un ou des enjeu de l'entreprise
- l'identification de ses compétences

Valoriser les apprentissages en entreprise

À chaque retour en formation, les apprenants seront amenés à réaliser des retours d'expériences sur des problématiques rencontrés par les entreprises. Ils construiront ainsi une connaissance des contextes de l'entreprise, un savoir collectif et une habitude à travailler en équipe. En résumé, cela implique :

- Construire un savoir partagé
- Comprendre les enjeux rencontrés en entreprise
- Savoir faire un retour d'expérience

L'ENVIRONNEMENT TECHNIQUE DE LA FORMATION

Les environnements techniques possibles en formation sont toujours adaptés au projet (bassin d'emplois local, entreprises partenaires...).

Voici deux exemples d'environnements techniques envisageable pour ce parcours formation :

Exemple d'une stack technique #1 avec Java			
Système(s) de stockage et données	Développement mobile	Développement desktop	Développement web
<ul style="list-style-type: none"> • SQLite, PostgreSQL • SQL • Hibernate 	<ul style="list-style-type: none"> • Java • Android Studio 	<ul style="list-style-type: none"> • Java • JavaFx 	<ul style="list-style-type: none"> • Spring • Hibernate • Thymleaf

Exemple d'une stack technique #2 avec Javascript			
Système(s) de stockage et données	Développement mobile	Développement desktop	Développement web
<ul style="list-style-type: none"> • SQLite, PostgreSQL • SQL • Prisma 	<ul style="list-style-type: none"> • Javascript • React native 	<ul style="list-style-type: none"> • Javascript • Electron 	<ul style="list-style-type: none"> • Node, express.js • React

LA PÉDAGOGIE SIMPLON

Les 6 piliers de la pédagogie Simplon



Notre objectif est de former de bon·ne·s professionnel·le·s maîtrisant la culture du domaine auquel il·le·s sont formé·e·s, détenteur·rice·s des capacités techniques attendues dans le secteur et en capacité de renouveler leurs connaissances, travailler en équipe et résoudre des problèmes complexes.

À cette fin, Simplon met en œuvre une approche active dans ses méthodes d'animation et de formation. La pédagogie dite active, par sa nature, ne vise pas à transmettre de la connaissance - uniquement - mais vise - à Simplon - la montée en compétences des apprenant·e·s. En cela, l'accompagnement et le cadre pédagogique fourni par Simplon ne peut se satisfaire dans la production de contenus théoriques figés ayant pour effet envisageable la transmission d'informations et *in fine* potentiellement de connaissances.

Il s'agit alors de fournir un cadre favorable à la mise en situation des apprenants leur permettant de mobiliser des compétences professionnelles par le faire, notamment au travers de projets pédagogiques et d'activités pédagogiques reprenant des contextes d'entreprises.

Ce cadre est garanti par une méthodologie bien définie, dont les grandes lignes sont :

- un cadre de conception des projets pédagogiques formalisé et disponible sur la plateforme en ligne d'apprentissage de Simplon, Simplonline.
- une évaluation des acquis tout au long de la formation,
- l'apprentissage par la collaboration et l'entraide,
- un accompagnement vers l'autonomie et l'apprendre à apprendre.

Les apports théoriques ne sont pas dispensés à part, à côté de la pratique ; mais bien tout au long du parcours de formation, toujours en lien avec le besoin d'apprendre lui-même provenant du besoin de faire.

En synthèse, la pédagogie Simplon c'est :

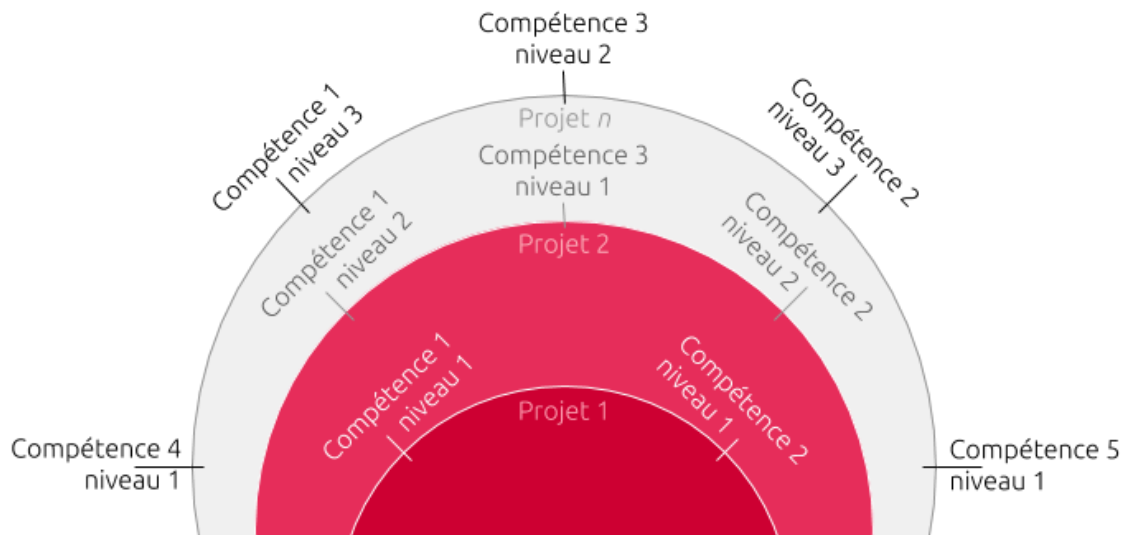
- Une pédagogie active : mise en pratique, apprentissage par projets, co-apprentissage, etc.
- Une équipe pluridisciplinaire : le formateur·rice référent·e, des expert·e·s techniques, un·e “chargé·e de médiation emploi” ainsi qu'un·e responsable de formation.
- L'entreprise au cœur du parcours : parrainage de promotion, mentorat inversé, périodes d'immersion, simulations d'entretien, job dating, meet up, etc.

Pour chaque compétence, la progression est divisée en trois niveaux :

- Niveau 1 - *imiter*
- Niveau 2 - *adapter*
- Niveau 3 - *transposer*

On considère une compétence acquise lorsque le troisième niveau est atteint.

La montée en compétence se concrétise lors de réalisations de projets d'apprentissage et de projets d'application :



Les adaptations à l'oeuvre pour les formation hybrides et distancielles

Le contexte sanitaire de 2020 oblige les entreprises mais aussi les centres de formation à réinventer leurs organisations.

Dans le cas où le projet de formation intègre du distanciel, le ou la formateur·rice donne un cadre pédagogique et accompagne les apprenant·es selon la même pédagogie Simplon assurée en présentiel : pédagogie active, par projets (pédagogiques), avec une approche par compétences.

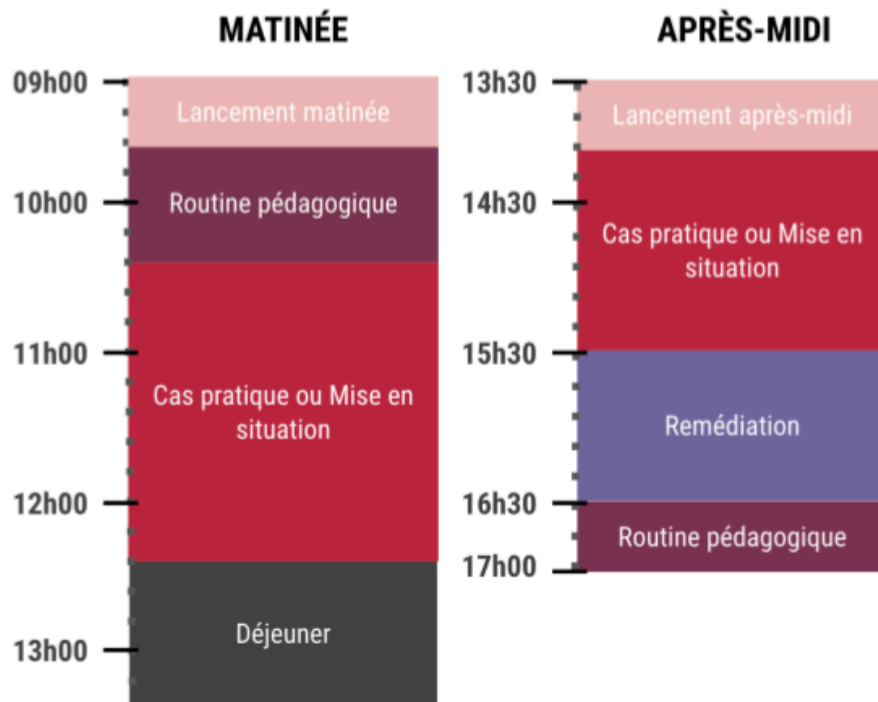
Cette pédagogie sera mise en œuvre de sorte à pouvoir accompagner au mieux les apprentissages de chaque apprenant·e en tenant compte des contraintes imposées par les conditions hybrides ou distancielles. La nature de l'adaptation de l'accompagnement reçu par chaque apprenant·e ne sera pas uniforme à l'échelle de la promotion, ni à l'échelle de la durée de la formation. C'est une forme de différenciation pédagogique.

La nature des modalités pédagogiques d'accompagnement et d'évaluation sont adaptées selon le format contraint par la situation et les besoins des apprenant·e·s. Les critères de différenciation pédagogique sont définis au préalable par l'équipe pédagogique. Les semaines types présentées plus loin dans le document sont construites à titre d'exemple selon les critères suivants, eux aussi, donnés à titre d'exemple :

- niveau d'autonomie des apprenant.e.s,
- introduction d'un nouvel outil,
- niveau de compétence des apprenant.e.s.

Finalement, la différenciation pédagogique, c'est ce qui permet de garantir, en pédagogie Simplon, un suivi régulier et de qualité des apprentissage, selon les besoins et les contraintes de chaque apprenant·e, dans un contexte de formation hybride ou distanciel.

UNE JOURNÉE TYPE À SIMPLON



Lancement

Le formateur lance les unités d'apprentissage. Il peut en profiter pour sonder le moral des apprenant-e-s, réaliser une activité de cohésion de groupe ou un réveil pédagogique.

Cas pratique ou mise en situation

Nos formations s'articulent autour de situations d'apprentissage dont les modalités pédagogiques sont précisées par le formateur. En effet, il indique aux apprenant-e-s si la situation doit être réalisée **individuellement ou en groupe, précise les contraintes de temps, les outils de collaboration à utiliser**, etc. Il peut être sollicité à tout moment de réalisation de la mise en situation par les apprenant-e-s et lever les points de blocages individuels ou collectifs selon les modalités qui lui semblent les plus adaptées (rappel théorique individuel ou en groupe, mix des groupes, activité guidée...).

Remédiation *(pas nécessairement quotidienne)*

À l'issue de certaine situation d'apprentissage, **un temps de "remédiation" en collectif** permet de faire le point sur les tâches réalisées, les ressources théoriques associées et comment elles ont été mobilisée pour dépasser l'obstacle rencontré.

Routine pédagogique

À cela s'ajoutent, des **rituels de réveil pédagogique, de veille et d'analyse réflexive** permettant de faire réfléchir le stagiaire sur ce qu'il a appris, fait, acquis grâce à la situation.

LES MOYENS PÉDAGOGIQUES ET TECHNIQUES

La formation en présentiel à lieu dans les salles de formations de Simplon.co, elles sont toutes équipées :

- de tables et de chaises
- d'une connexion internet
- d'un vidéoprojecteur

Chaque apprenant se voit attribuer un ordinateur portable pour toute la durée de la formation.

Les conditions de suivi de la formation à distance doivent être favorables aux apprentissages. Si besoin et si la situation sanitaire le permet, des solutions peuvent être proposées aux apprenant.e.s (prêt de matériel , dépannage connexion, suivi spécifique, ...).

Les outils utilisés en formation sont :

- Simplonline, notre plateforme d'apprentissage en pédagogie active et par projets permet de cadrer le travail à faire, évaluer la montée en compétence et suivre la progression collective et individuelle.
- Discord est la solution à Simplon pour les échanges au sein d'une promotion. Une promotion Simplon dispose de son salon Discord dédié dans le Discord des Alumnis, ce qui permet la continuité post-formation. À privilégier pour toute discussion en synchrone. Il y a aussi la possibilité de s'appeler à plusieurs, de streamer etc.
- Google Meet vous permet d'organiser depuis l'agenda Google des sessions de vidéoconférence avec les apprenants. Nous vous conseillons de réaliser des sessions successives par petits groupes d'apprenant.e-s pour plus d'efficacité.
- D'autres outils permettant de faciliter l'accompagnement ou l'animation de la formation à distance sont disponibles pour les formateur.rice-s, notamment : kahoot, Quizizz, trello, ...

L'ENCADREMENT PÉDAGOGIQUE

La formation est assurée par au moins un formateur sinon plusieurs, dont un formateur référent.

Chaque formateur doit suivre une formation de formateur assuré par Simplon.co.

Les apprenants sont encadrés par une équipe pédagogique pluridisciplinaire :

- Un.e formateur.rice référent.e de la formation
- Un.e ou deux formateur.rice.s en appui
- Un.e chargé.e de promo/chargé.e de médiation emploi
- L'assiduité des apprenants est suivie grâce à la signature de feuille d'émargement

L'équipe :

- Informe les prescripteurs et le public, détecte des talents et les sélectionne
- Définit les étapes de la progression des apprenants
- Organise les interventions de chacun auprès des promos
- Facilite l'acquisition des savoirs et compétences
- Garantit la mise en oeuvre des principes pédagogiques de SIMPLON
- Accompagne les apprenants dans leur insertion professionnelle
- Concourt à valoriser la marque SIMPLON auprès de tous les intervenants

De façon spécifique dans le cas où la formation devrait intégrer des temps à distance

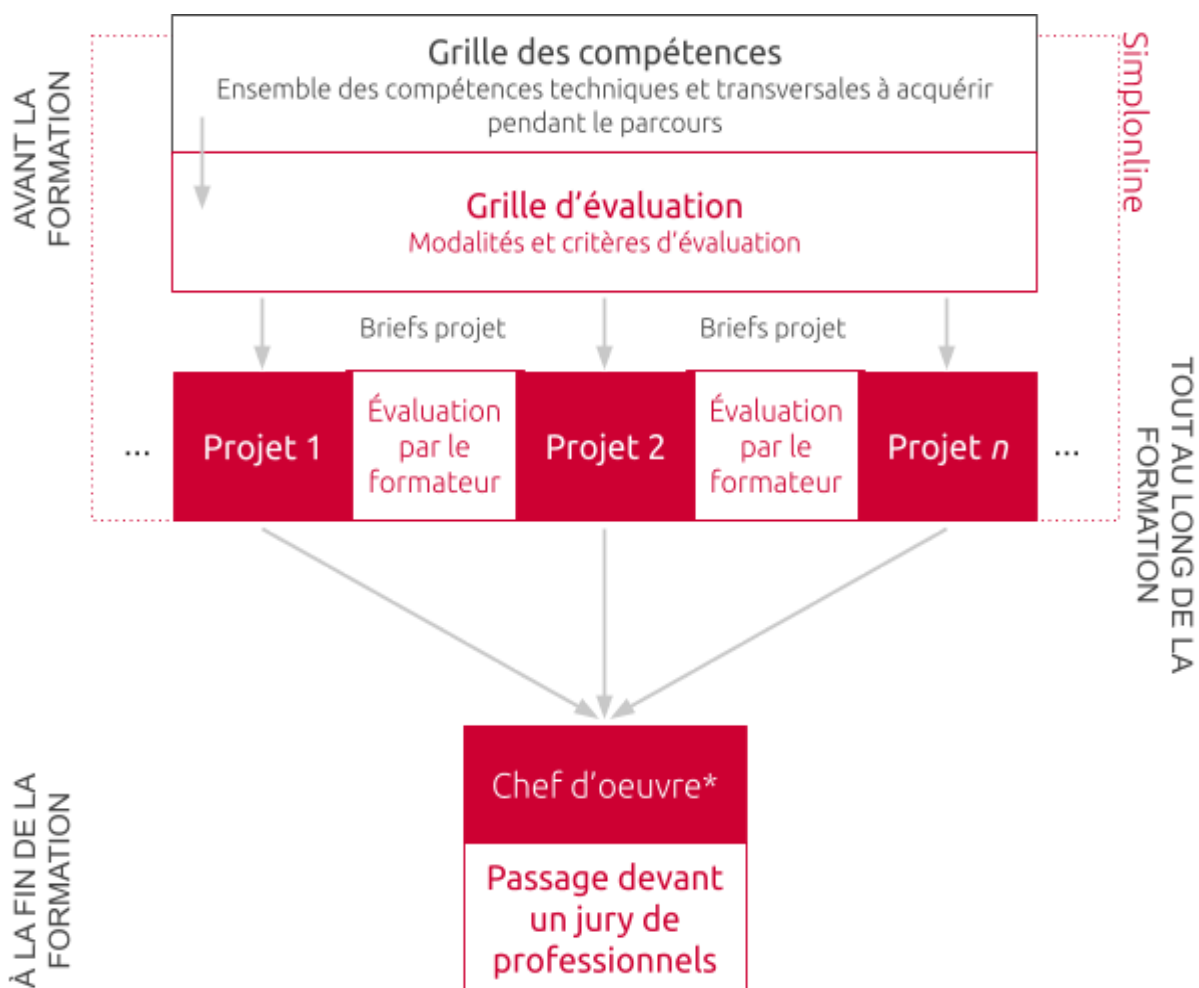
- Chaque formateur.rice doit suivre le module de formation "hybride/distanciel"
- L'assiduité des apprenant.e.s est suivi grâce à un outil dédié

L'ÉVALUATION PÉDAGOGIQUE À SIMPLON

Les apprenant·e·s sont évalué·e·s tout au long de la formation, sur leur capacité à agir en situation d'apprentissage, simulant les situations emblématiques de l'activité professionnelle visée par la formation. Toutes ces réalisations constituent leur premier "portfolio" et démontrent leur capacité à être efficient dans un environnement de travail.

Les briefs de ces situations sont construits et proposés par les formateurs depuis notre plateforme d'apprentissage *Simplonline*. Cette plateforme permet également le partage de ressources pédagogiques entre tous les membres de la plateforme (apprenants et formateurs). L'apprenant prend donc connaissance des briefs : objectifs, modalités, évaluation, etc. tout en s'aidant des ressources disponibles. Il peut par la suite y consulter sa progression, dans l'acquisition des compétences, visées par la formation.

L'évaluation des compétences se fait de façon continue et terminale :



*Le chef d'œuvre correspond au projet de fin de parcours, soutenu lors de l'évaluation finale devant un jury.