Présentiel/distanciel

NOUVEAUTÉ

Design Patterns et conception logicielle avancée

Objectifs pédagogiques

- Comprendre l'importance des design patterns dans la conception logicielle.
- Maîtriser les principes SOLID ainsi que les bonnes pratiques de développement logiciel.
- Mettre en oeuvre les principaux design patterns afin d'améliorer la maintenabilité et l'évolutivité d'un projet logiciel.
- Identifier les situations nécessitant l'utilisation d'un design pattern et justifier le choix de la solution retenue.
- Analyser, refactoriser et optimiser un code existant en intégrant les design patterns les plus adaptés.

Prérequis

Personnes ayant des prérequis techniques orientés études ou développement (avec diplôme de niveau Bac+3, tel qu'une licence, un BUT ou un titre de niveau 6) ou des personnes infomaticiennes expérimentées.

Public concerné

La formation en Design Patterns s'adresse principalement aux architectes logiciels et développeurs expérimentés en programmation orientée objet.

Programme détaillé

Fondamentaux des Design Patterns et principes SOLID (3h30)

- Expliquer le rôle des Design Patterns dans la conception d'applications logicielles.
- Identifier les situations dans lesquelles les principes SOLID permettent d'améliorer la qualité d'un code.
- Appliquer les cinq principes SOLID lors de la conception ou du refactoring d'une application.
- Sélectionner un Design Patterns adapté à un besoin de développement courant.
- Refactoriser un code existant afin d'en améliorer la maintenabilité, la lisibilité et l'évolutivité.
- Démonstration appliquée :
  - Analyser un code non optimisé et refactoring avec SOLID.
- Travaux pratiques :
  - Mettre en oeuvre les principes SOLID dans un projet simple.

Les Design Patterns de Création (7h00)

- Identifier les principaux Design Patterns de création et leur domaine d'application.
- Choisir le Design Patterns de création le plus adapté en fonction d'un besoin métier.
- Implémenter les Design Patterns de création dans un projet de développement logiciel.
- Évaluer les avantages et les limites de chaque Patterns afin d'améliorer la maintenabilité et l'évolutivité d'une application.
- Démonstration appliquée :
  - Mise en oeuvre d'un Pattern Factory au sein d'un projet simple.
- Travaux pratiques :
  - Appliquer des Patterns de création dans un contexte métier.

Les Design Patterns Structurels (7h00)

- Identifier les principaux Patterns structurels et leurs cas d'utilisation.
- Expliquer le rôle des Patterns Adapter, Decorator, Proxy, Composite, Bridge et Facade.
- Réduire le couplage entre composants logiciels en appliquant le Pattern Adapter.
- Refactoriser une application existante en intégrant un Pattern structurel adapté.
- Démonstration appliquée :
  - Mise en oeuvre du Pattern Adapter afin de réduire le couplage entre les composants d'une application.
- Travaux pratiques :
  - Refactorisation d'un projet existant par l'intégration d'un Pattern structurel adapté aux besoins identifiés.

Les Design Patterns Comportementaux (7h00)

- Identifier les principaux Patterns comportementaux.
- Mettre en oeuvre des Patterns comportementaux afin d'optimiser la gestion des interactions et des comportements des objets dans une application.
- Démonstration appliquée :
  - Démonstration de la mise en oeuvre du Pattern Observer pour la gestion d'événements.
- Travaux pratiques :
  - Intégrer et utiliser des Patterns comportementaux au sein d'un projet collaboratif.

Validation et refactoring avec les Design Patterns (7h00)

- Évaluer la pertinence de l'utilisation d'un Design Pattern.
- Améliorer la qualité du code en appliquant les bonnes pratiques de conception et de refactoring.
- Démonstration appliquée :
  - Étude de cas sur une refonte de code avec Patterns.
- Travaux pratiques :
  - Analyser un projet et refactoring progressif en intégrant des Patterns

Délais d'accès à la formation

Les inscriptions sont possibles jusqu'à 48 heures avant le début de la formation.
Dans le cas d'une formation financée par le CPF, ENI Service est tenu de respecter un délai minimum obligatoire de 11 jours ouvrés entre la date d'envoi de sa proposition et la date de début de la formation.

Modalités et moyens pédagogiques, techniques et d'encadrement

La formation peut être suivie uniquement en présentiel dans l'un des campus de l'ENI Ecole Informatique (Nantes, Rennes, Niort, Quimper). Le stagiaire intègre une promotion de l'ENI Ecole Informatique d'environ 25 à 28 stagiaires max par session.
Chaque stagiaire dispose d'un poste de travail adapté aux besoins de la formation et des supports de cours au format numérique.
Il alterne apports théoriques, démonstrations et mise en pratique au travers d'exercices, de travaux pratiques et de cas concrets, réalisés individuellement ou en groupe.

Modalités d'évaluation des acquis

Une évaluation des acquis du stagiaire est réalisée en fin de formation.

Moyens de suivi d'exécution et appréciation des résultats

Feuille de présence, émargée par demi-journée par chaque stagiaire et le formateur.
Évaluation qualitative de fin de formation, qui est ensuite analysée par l'équipe pédagogique de l'ENI Ecole Informatique.
Attestation de fin de formation remise au stagiaire.

Qualification du formateur

La formation est animée par un professionnel de l'informatique et de la pédagogie, recruté sur la base de son expérience professionnelle dans le domaine enseigné.
Nos formateurs sont pour la plupart ENI Certified Trainers. Ils assurent une veille technologique permanente.

Accessibilité de la formation

ENI Service s'engage en faveur de l'accessibilité pour les personnes en situation de handicap (PSH). Toutes nos formations sont ainsi accessibles aux PSH. Pour en savoir plus, nous vous invitons à consulter la page "Accueil des personnes en situation de handicap" de notre site internet.