

Kereval propose des formations Inter et en Intra-entreprise dispensées par des formateurs certifiés et experts en test logiciel. Selon les besoins spécifiques, Kereval est en mesure de dispenser des formations sur-mesure, dont le contenu et la durée sont adaptés à votre contexte.

Les systèmes à base d'Intelligence Artificielle sont de plus en plus répandus, notamment dans des cas d'applications critiques, tels que la conduite autonome, l'aviation ou encore l'imagerie médicale. Cependant, les techniques classiques de test logiciel ne sont généralement pas applicables à ces systèmes. Pour répondre à ce besoin, de nouveaux processus de test pour l'Intelligence Artificielle ont émergé ces dernières années.

Cette formation permet de comprendre les enjeux du test des systèmes à base d'IA, au regard de leurs vulnérabilités, et d'identifier les méthodes de test et de vérification les plus pertinentes.

Dans le cadre d'exercices pratiques, nous vous proposons de mettre en œuvre plusieurs méthodes d'assurance de la qualité des systèmes à base d'IA, notamment en lien avec la robustesse, l'éthique ou encore l'explicabilité.

Objectifs pédagogiques

- > Comprendre le test dans le domaine de l'IA
- > Connaître les principales faiblesses d'une IA
- > Savoir choisir les méthodes de test et de vérification pertinentes

Pré-requis

Connaissances des fondamentaux en IA et du langage de programmation Python

Public formé

Chef de projet, ingénieur, développeur, chercheur

Contenu

- > Les enjeux de l'IA
 - Le test, plus qu'une évaluation des performances
 - Pourquoi tester l'IA ? L'exemple du logiciel COMPAS
 - Différences fondamentales avec les logiciels traditionnels et leur impact sur le test
 - Les défis du test des systèmes à base d'IA
- > Les propriétés d'une IA fiable
 - Quelles propriétés pour une IA fiable ?
 - L'importance de la qualité des données
 - Définition d'une IA éthique
 - Reconnaître les biais d'une IA
 - L'explicabilité, un outil pour la confiance
 - Cybersécurité de l'IA : des vulnérabilités très spécifiques
 - Sûreté et robustesse en pratique
- > Les méthodes et outils pour le test

- Les oracles de test ou comment savoir que le système se trompe
- Générateurs de cas de test :
 - Tester les propriétés du système avec les tests métamorphiques
 - Adversarial examples : tester la résistance de l'IA aux attaques

> Les méthodes de vérification formelle

- La vérification formelle pour apporter une preuve du comportement du système
 - Définitions mathématiques et concepts
 - Méthodes par approximation des modèles (interprétation abstraite, approximation linéaire)
 - Utilisation de solveurs pour l'obtention de garanties (solveur SMT, branch-and-bound)
- Exemple pratique : mise en œuvre des méthodes d'interprétation abstraite et de branch-and-bound

> Le développement d'une IA fiable

- Méthodes de défense contre les attaques : protections en amont et défense « by design »
- Réglementation européenne, normes et bonnes pratiques

Méthodes pédagogiques

- > Formation en présentiel ou en distanciel synchrone (Teams)
- > Exposé des concepts avec support powerpoint©
- > Exercices pratiques : génération d'adversarial examples, vérification formelle d'un réseau, méthodes d'explicabilité, tests métamorphiques

Modalités d'accès et délais

Modalités d'inscription : nous contacter

Délais d'accès : de 1 à 6 mois

Les délais d'accès sont dépendants de la disponibilité de nos formateurs.

Accessible aux personnes en situation de handicap. Nous contacter pour tout aménagement spécifique.

Modalités d'évaluation

Le formateur évalue la progression pédagogique du participant tout au long de la formation au moyen de QCM, mises en situation, travaux pratiques...

Le participant complète également un test de positionnement en amont et en aval pour valider les compétences acquises.

Durée

Une journée (8h)

Prix intra – entreprise	Prix inter – entreprise
Nous contacter pour obtenir un devis.	950€ HT / stagiaire tout inclus (Pause-café et déjeuner compris)

Chiffres clés

Nous ne possédons pas encore d'indicateurs sur cette formation.