

Formation & Certification • App Development with Swift
Certified User

App Development with Swift – Certified User

Formation en distanciel (présentiel sur demande).
Développez une maîtrise plus avancée de Swift, de Xcode
et de UIKit afin de créer des applications iOS complètes qui
respectent les bonnes pratiques en termes d'interface, de
navigation et de qualité de code.

Distributeur officiel Certiport

Centre d'examen Certiport

Learn • Practice • Certify

Durée 24–30 h	Examen App Development with Swift Certified User
Modalité Distanciel	Niveau Certified User – Intermédiaire

INSCRIPTION / RÉSERVATION



Je m'inscris
maintenant



- **Learn** : ressources officielles Apple & Certiport couvrant tous les domaines d'objectifs de la certification.
- **Practice** : ateliers pratiques, projets Xcode, exercices guidés sur Swift et UIKit, et simulations d'examen.
- **Certify** : voucher d'examen App Development with Swift Certified User inclus pour passer la certification dans notre centre.

OBJECTIFS PÉDAGOGIQUES

- Ce programme apprend à utiliser Xcode et ses principaux outils de développement.
- Il enseigne à écrire du code Swift solide avec types, collections, fonctions et Optionals.
- Il permet de créer des interfaces iOS professionnelles et des applications multi-vues avec navigation.
- Il prépare au débogage efficace et à l'examen App Development with Swift Certified User.

PUBLIC CIBLE

- Ce programme s'adresse aux étudiants et développeurs ayant déjà une première expérience en Swift ou développement iOS.
- Il est conçu pour les enseignants souhaitant intégrer des projets iOS avancés dans leurs cours.
- Il cible aussi les personnes en reconversion cherchant une certification reconnue pour valider leurs compétences.

PRÉREQUIS

- Ce programme requiert une compréhension de base de Swift ou le niveau Associate équivalent, ainsi que des notions de logique et de structures de contrôle.
- Un Mac avec Xcode installé est nécessaire pour les travaux pratiques.

PROGRAMME DE LA FORMATION – DÉTAILLÉ

Xcode Developer Tools

Identifier et utiliser les fonctionnalités de l'interface Xcode

- Naviguer dans Xcode et ses zones principales.
- Créer et modifier des vues avec Interface Builder.
- Accéder à la documentation et à l'aide intégrée.

Construire et exécuter une application

- Lancer l'app sur le simulateur iOS.
- Déployer et exécuter l'app sur un appareil iOS réel.

Utiliser des techniques de débogage pour résoudre les erreurs

- Placer des breakpoints et exécuter le code étape par étape.

Positionner et mettre en page des objets UIKit

- Utiliser l'auto layout.
- Intégrer des objets dans un Stack View.
- Gérer alignements et contraintes.
- Naviguer dans les composants UI via le Document Outline.
- Mettre en œuvre la "personnalité" de l'app (cohérence UI, style).

Swift Programming Language

Déclarer et utiliser les types Swift de base

- Décrire et utiliser les types de données et les opérateurs.
- Mettre en œuvre le type casting (sécurisé et non sécurisé).
- Différencier et utiliser constantes et variables.
- Interpréter et utiliser les types de base.

Gérer les données avec les types de collection

- Utiliser les Arrays (tableaux).
- Utiliser les Dictionaries (dictionnaires clé/valeur).

Savoir quand et comment appliquer les structures de contrôle et les boucles

- Utiliser les opérateurs logiques.
- Utiliser guard pour la validation.
- Utiliser les opérateurs de plage (range operators).



PROGRAMME DE LA FORMATION – DÉTAILLÉ

Utiliser les fonctions

- Organiser et structurer le code avec des fonctions.
- Créer et appeler une fonction.
- Utiliser et interpréter la valeur de retour d'une fonction.
- Personnaliser les noms de paramètres (internes, externes, anonymes).
- Implémenter des valeurs par défaut pour les paramètres.

Utilisation correcte des structs, classes et enums

- Définir et utiliser propriétés et méthodes.
- Différencier structures et classes.
- Différencier les divers initializers.
- Définir et utiliser des observateurs de propriété.

Utiliser les types optionnels (Optionals)

- Démontrer comment unwrap des Optionals de manière sûre.
- Appliquer l'Optional binding et l'Optional chaining (if let, guard let, etc.).

Évaluer la portée des variables (scope) et le shadowing

- Comprendre comment la portée affecte l'accès aux variables et éviter le shadowing non intentionnel.

iOS UIKit

Créer des View Controllers pour implémenter la logique de l'app

- Configurer des View Controllers pour gérer les écrans et la logique métier.

Décrire le cycle de vie d'un View Controller

- Comprendre les méthodes clés (viewDidLoad, viewWillAppear, etc.).

Utiliser les segues pour lier les View Controllers, préparer, passer des données et gérer les unwind segues

- Configurer des segues pour la navigation entre écrans.
- Différencier les différents types de segues.



PROGRAMME DE LA FORMATION – DÉTAILLÉ

Créer une application multi-vues avec hiérarchie de navigation

- Créer et utiliser un Navigation Controller.
- Créer et utiliser un Tab Bar Controller.

Créer et manipuler des objets UIKit

- Utiliser des objets de vue courants (labels, image views, etc.).
- Utiliser des contrôles courants (boutons, text views, etc.).
- 3Utiliser IBOutlet et IBAction pour connecter les éléments du storyboard au code.

