

Mastère Spécialisé

Programmation Front End Full Stack Developer



Mastère Spécialisé Programmation Front End Full Stack Developer

- » Modalité : en ligne
- » Durée : 12 mois
- » Diplôme : TECH Euromed University
- » Accréditation : 60 ECTS
- » Horaire : à votre rythme
- » Examens : en ligne

Accès au site web : www.techtitute.com/fr/informatique/mastere/mastere-programmation-front-end-full-stack-developer

Sommaire

01

Présentation du programme

page 4

02

Pourquoi étudier à TECH?

page 8

03

Programme d'études

page 12

04

Objectifs pédagogiques

page 24

05

Opportunités de carrière

page 30

06

Méthodologie d'étude

page 34

07

Corps Enseignant

page 44

08

Diplôme

page 48

01

Présentation du programme

La Programmation *Front End Full Stack Developer* est l'une des compétences les plus recherchées dans la sphère numérique aujourd'hui, dans un environnement où la conception visuelle et la fonctionnalité des serveurs convergent pour donner vie à des plateformes dynamiques. Selon les données de l'Organisation des Nations Unies, les profils liés au développement web sont parmi les plus demandés dans le secteur technologique, avec une tendance à la croissance soutenue ces dernières années. Dans ce contexte, ce programme universitaire de TECH est apparu comme une réponse académique. Grâce à une méthodologie 100 % en ligne et à l'utilisation d'un matériel didactique actualisé, une expérience autonome et interactive sera fournie, alignée sur les défis actuels du marché.



“

Grâce à ce programme 100% en ligne, vous maîtriserez les outils et les langages les plus innovants en Programmation Front End Full Stack Developer pour développer des solutions web fonctionnelles”

Dans le contexte numérique actuel, la maîtrise des fondamentaux du développement web est devenue indispensable pour construire des solutions dynamiques, intuitives et adaptées aux exigences de l'environnement technologique. Pour cette raison, la spécialisation en *Front End Full Stack Developer* représente une voie stratégique pour ceux qui cherchent à créer des expériences interactives de l'interface à la logique du serveur. En fait, ce domaine permet non seulement de développer des produits numériques fonctionnels, mais aussi de garantir leur performance, leur accessibilité et leur évolutivité, des aspects essentiels sur le marché actuel, compétitif et orienté vers l'utilisateur.

Par conséquent, TECH a conçu un programme d'études innovant qui approfondira des aspects essentiels tels que les rôles impliqués dans les environnements de programmation, la conception et l'interprétation d'algorithmes efficaces et l'utilisation de référentiels logiciels pour la gestion collaborative du code. Grâce à une approche rigoureuse, ces contenus seront articulés avec les outils actuels du secteur, favorisant une compréhension globale des processus impliqués dans le développement complet des applications web.

Par la suite, ce programme universitaire offrira aux professionnels une occasion unique de perfectionner leurs compétences dans les environnements web, en consolidant les aptitudes qui leur permettent de construire des projets technologiques à partir de zéro et de mener des initiatives numériques avec autonomie. En outre, ils renforceront la prise de décision technique, la pensée logique et la capacité d'adapter des solutions à différents contextes d'entreprise. Grâce à une préparation alignée sur les exigences du marché, une solide projection professionnelle sera favorisée dans un domaine où la demande de main-d'œuvre est importante au niveau mondial.

Enfin, la méthodologie de TECH se caractérise par sa flexibilité, permettant d'accéder à tous les contenus en ligne à tout moment de la journée, de n'importe où et par le biais de multiples appareils connectés à Internet. Ce modèle est soutenu par le système innovant *Relearning*, qui favorisera une assimilation progressive et durable des connaissances grâce à la répétition intelligente de concepts clés, renforçant ainsi une formation autonome et dynamique.

Ce **Mastère Spécialisé en Programmation Front End Full Stack Developer** contient le programme universitaire plus complet et le plus actualisé du marché. Ses caractéristiques sont les suivantes :

- ♦ Développement de cas pratiques présentés par des experts en Programmation Front End Full Stack Developer
- ♦ Les contenus graphiques, schématiques et éminemment pratiques de l'ouvrage fournissent des informations scientifiques et pratiques sur les disciplines essentielles à la pratique professionnelle
- ♦ Les exercices pratiques où effectuer le processus d'auto-évaluation pour améliorer l'apprentissage
- ♦ L'accent est mis sur les méthodologies innovantes en matière de systèmes informatiques
- ♦ Cours théoriques, questions à l'expert, forums de discussion sur des sujets controversés et travail de réflexion individuel
- ♦ La possibilité d'accéder aux contenus depuis n'importe quel appareil fixe ou portable doté d'une connexion internet



Un cursus basé sur le système disruptif Relearning de TECH, qui facilitera l'assimilation de concepts complexes de manière rapide et flexible”



Vous approfondirez les rôles impliqués dans les environnements de programmation, en adaptant votre compréhension aux différents contextes de développement numérique”

Le corps enseignant comprend des professionnels appartenant au domaine de la Programmation Front End Full Stack Developer, qui apportent l'expérience de leur travail à ce programme, ainsi que des spécialistes reconnus issus d'entreprises de premier plan et d'universités prestigieuses.

Son contenu multimédia, développé avec les dernières technologies éducatives, permettra au professionnel un apprentissage situé et contextuel, c'est-à-dire un environnement simulé qui fournira un étude immersif programmé pour s'entraîner dans des situations réelles.

La conception de ce programme est axée sur l'Apprentissage par les Problèmes, grâce auquel l'étudiant doit essayer de résoudre les différentes situations de la pratique professionnelle qui se présentent tout au long du programme académique. Pour ce faire, le professionnel aura l'aide d'un système vidéo interactif innovant créé par des experts reconnus.

Vous serez en mesure d'automatiser les processus de développement et de déploiement à l'aide d'outils DevOps.

Vous maîtriserez l'utilisation des référentiels logiciels, en optimisant la gestion collaborative du code.



02

Pourquoi étudier à TECH?

TECH est la plus grande Université numérique du monde. Avec un catalogue impressionnant de plus de 14 000 programmes universitaires, disponibles en 11 langues, elle se positionne comme un leader en matière d'employabilité, avec un taux de placement de 99 %. En outre, elle dispose d'un vaste corps professoral composé de plus de 6 000 professeurs de renommée internationale.



“

*Étudiez dans la plus grande université
numérique du monde et assurez
votre réussite professionnelle. L'avenir
commence à TECH”*

La meilleure université en ligne du monde, selon FORBES

Le prestigieux magazine Forbes, spécialisé dans les affaires et la finance, a désigné TECH comme "la meilleure université en ligne du monde". C'est ce qu'ils ont récemment déclaré dans un article de leur édition numérique dans lequel ils se font l'écho de la réussite de cette institution, "grâce à l'offre académique qu'elle propose, à la sélection de son corps enseignant et à une méthode d'apprentissage innovante visant à former les professionnels du futur".

Le meilleur personnel enseignant top international

Le corps enseignant de TECH se compose de plus de 6 000 professeurs jouissant du plus grand prestige international. Des professeurs, des chercheurs et des hauts responsables de multinationales, parmi lesquels figurent Isaiah Covington, entraîneur des Boston Celtics, Magda Romanska, chercheuse principale au Harvard MetaLAB, Ignacio Wistumba, président du département de pathologie moléculaire translationnelle au MD Anderson Cancer Center, et D.W. Pine, directeur de la création du magazine TIME, entre autres.

La plus grande université numérique du monde

TECH est la plus grande université numérique du monde. Nous sommes la plus grande institution éducative, avec le meilleur et le plus vaste catalogue éducatif numérique, cent pour cent en ligne et couvrant la grande majorité des domaines de la connaissance. Nous proposons le plus grand nombre de diplômes propres, de diplômes officiels de troisième cycle et de premier cycle au monde. Au total, plus de 14 000 diplômes universitaires, dans dix langues différentes, font de nous la plus grande institution éducative au monde.



Les programmes d'études les plus complets sur la scène universitaire

TECH offre les programmes d'études les plus complets sur la scène universitaire, avec des programmes qui couvrent les concepts fondamentaux et, en même temps, les principales avancées scientifiques dans leurs domaines scientifiques spécifiques. En outre, ces programmes sont continuellement mis à jour afin de garantir que les étudiants sont à la pointe du monde universitaire et qu'ils possèdent les compétences professionnelles les plus recherchées. De cette manière, les diplômes de l'université offrent à ses diplômés un avantage significatif pour propulser leur carrière vers le succès.

Une méthode d'apprentissage unique

TECH est la première université à utiliser *Relearning* dans tous ses formations. Il s'agit de la meilleure méthodologie d'apprentissage en ligne, accréditée par des certifications internationales de qualité de l'enseignement, fournies par des agences éducatives prestigieuses. En outre, ce modèle académique perturbateur est complété par la « Méthode des Cas », configurant ainsi une stratégie d'enseignement en ligne unique. Des ressources pédagogiques innovantes sont également mises en œuvre, notamment des vidéos détaillées, des infographies et des résumés interactifs.

L'université en ligne officielle de la NBA

TECH est l'université en ligne officielle de la NBA. Grâce à un accord avec la grande ligue de basket-ball, elle offre à ses étudiants des programmes universitaires exclusifs ainsi qu'un large éventail de ressources pédagogiques axées sur les activités de la ligue et d'autres domaines de l'industrie du sport. Chaque programme est conçu de manière unique et comprend des conférenciers exceptionnels : des professionnels ayant un passé sportif distingué qui apporteront leur expertise sur les sujets les plus pertinents.

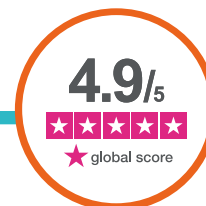
Leaders en matière d'employabilité

TECH a réussi à devenir l'université leader en matière d'employabilité. 99 % de ses étudiants obtiennent un emploi dans le domaine qu'ils ont étudié dans l'année qui suit la fin de l'un des programmes de l'université. Un nombre similaire parvient à améliorer immédiatement sa carrière. Tout cela grâce à une méthodologie d'étude qui fonde son efficacité sur l'acquisition de compétences pratiques, absolument nécessaires au développement professionnel.



Google Partner Premier

Le géant américain de la technologie a décerné à TECH le badge Google Partner Premier. Ce prix, qui n'est décerné qu'à 3 % des entreprises dans le monde, souligne l'expérience efficace, flexible et adaptée que cette université offre aux étudiants. Cette reconnaissance atteste non seulement de la rigueur, de la performance et de l'investissement maximaux dans les infrastructures numériques de TECH, mais positionne également TECH comme l'une des principales entreprises technologiques au monde.



L'université la mieux évaluée par ses étudiants

Les étudiants ont positionné TECH comme l'université la mieux évaluée du monde dans les principaux portails d'opinion, soulignant sa note la plus élevée de 4,9 sur 5, obtenue à partir de plus de 1 000 évaluations. Ces résultats consolident TECH en tant qu'institution universitaire de référence internationale, reflétant l'excellence et l'impact positif de son modèle éducatif.



03

Programme d'études

Cet itinéraire académique exclusif approfondira des sujets très pertinents, qui complètent la Programmation *Front End Full Stack Developer*. Ainsi, vous vous plongerez progressivement dans des outils tels que le langage HTML, essentiel pour structurer correctement l'information dans les environnements numériques. Dans le même temps, vous approfondirez la conception de *layouts* modernes, en mettant l'accent sur l'adaptabilité et l'expérience utilisateur, éléments cruciaux dans le développement d'interfaces efficaces. De manière complémentaire, la maîtrise du langage JavaScript, indispensable à la mise en œuvre de fonctionnalités dynamiques, sera intégrée. Cette combinaison de connaissances favorise une connaissance technique rigoureuse, alignée sur les besoins du secteur technologique.



“

Vous approfondirez les principes structurels les plus récents du langage HTML, appliqués au développement d'interfaces fonctionnelles”

Module 1. Développeur *Full Stack Developer*

- 1.1. Développement *Full Stack Developer* I. Programmation et langages
 - 1.1.1. Programmation
 - 1.1.2. Rôles de planification
 - 1.1.3. Langages et *framework*
 - 1.1.4. Algorithmes
 - 1.1.5. Caractéristiques d'un algorithme
- 1.2. Développement *Full Stack Developer* II. Typologie
 - 1.2.1. Variables et constantes
 - 1.2.2. Types
 - 1.2.3. Opérateurs
 - 1.2.4. Déclarations
 - 1.2.5. Boucles
 - 1.2.6. Fonctions et objectifs
- 1.3. Structure de données dans le développement
 - 1.3.1. Types de structure linéaire
 - 1.3.2. Types de structures fonctionnelles
 - 1.3.3. Types de structures d'arborescence
- 1.4. Conception et interprétation d'algorithmes
 - 1.4.1. Parallélisme en développement. Diviser pour régner
 - 1.4.2. Algorithmes voraces
 - 1.4.3. Programmation dynamique
- 1.5. Environnement et outils de développement orienté *Full Stack Developer*
 - 1.5.1. Préparation de l'environnement pour Mac OS
 - 1.5.2. Préparation de l'environnement Linux
 - 1.5.3. Préparation de l'environnement Windows
- 1.6. Ligne de commande. Typologie et fonctionnement
 - 1.6.1. Le terminal
 - 1.6.2. Émulateurs
 - 1.6.3. Interprète de commande
 - 1.6.4. Premières commandes
 - 1.6.5. Navigation
 - 1.6.6. Gestion des fichiers et des dossiers à l'aide de l'interface de ligne de commande
 - 1.6.7. *Secure shell*. SSH
 - 1.6.8. Commandes avancées
- 1.7. Git. Dépôt de *logiciels*
 - 1.7.1. Git Dépôt de *logiciels*
 - 1.7.2. Utilisation de Git
 - 1.7.3. Dépôt de *logiciels*
 - 1.7.4. Branches
 - 1.7.5. Cycle d'utilisation
 - 1.7.6. Commandes
- 1.8. Service d'hébergement de versions de code
 - 1.8.1. Service d'hébergement de versions de code
 - 1.8.2. Fournisseurs
 - 1.8.3. Référentiel
- 1.9. Internet
 - 1.9.1. Internet
 - 1.9.2. Protocoles utilisés dans les WWW
 - 1.9.3. Protocole HTTP
- 1.10. Méthodologies dans le développement *Full Stack*
 - 1.10.1. *Scrum*
 - 1.10.2. XP
 - 1.10.3. *Design Sprint*

Module 2. *Front-end* dans Programmation

- 2.1. Langage HTML
 - 2.1.1. Document HTML
 - 2.1.2. Élément *head*
 - 2.1.3. Élément *body*
 - 2.1.4. Texte
 - 2.1.5. Hyperliens
 - 2.1.6. Images
 - 2.1.7. *Primer Site*
- 2.2. Langage HTML *Layouts*
 - 2.2.1. Langage HTML Éléments
 - 2.2.2. *Layout* traditionnel
 - 2.2.3. *Layout* sémantique
- 2.3. Feuilles de style en cascade CSS (*Cascading Style Sheets*)
 - 2.3.1. Inclusion de CSS dans un document HTML
 - 2.3.2. Commentaires
 - 2.3.3. Sélecteurs
 - 2.3.4. Sélecteurs avancés
- 2.4. Propriétés CSS (*cascading style sheets*)
 - 2.4.1. Couleur
 - 2.4.2. Texte
 - 2.4.3. Pseudo classes
 - 2.4.4. Transitions
 - 2.4.5. Animations
 - 2.4.6. Animation des éléments
 - 2.4.7. Animation avancée
- 2.5. Modèle de boîte
 - 2.5.1. Hauteur et largeur
 - 2.5.2. Marge
 - 2.5.3. Remplissages
- 2.6. Positionnement
 - 2.6.1. Positionnement statique
 - 2.6.2. Positionnement relatif
 - 2.6.3. Positionnement absolu
 - 2.6.4. Positionnement fixe
 - 2.6.5. *Flotteurs*
- 2.7. Conception adaptative
 - 2.7.1. *Viewport*
 - 2.7.2. *Media Queries*
 - 2.7.3. Unités CSS
 - 2.7.4. *Images*
 - 2.7.5. *Frameworks*
- 2.8. Layout moderne
 - 2.8.1. *Flex*
 - 2.8.2. *Grid*
 - 2.8.3. *Flex vs Grid*
- 2.9. Préprocesseur
 - 2.9.1. Sass
 - 2.9.2. Variables
 - 2.9.3. Mixins
 - 2.9.4. Boucles
 - 2.9.5. Fonctions
- 2.10. Système de conception
 - 2.10.1. *Bootstrap*
 - 2.10.2. Grille de *bootstrap*
 - 2.10.3. *Header* et *Footer* de notre site
 - 2.10.4. Formulaires
 - 2.10.5. *Cards*
 - 2.10.6. Modals

Module 3. Le langage JAVASCRIPT appliqué au *Full Stack Developer*

- 3.1. Types primitifs et opérateurs
 - 3.1.1. Langage JavaScript
 - 3.1.2. Les nombres et leurs opérateurs
 - 3.1.3. Chaînes de texte et leurs opérateurs
 - 3.1.4. Valeurs booléennes
 - 3.1.5. Conversion entre les types
- 3.2. Régulateurs de débit et structure
 - 3.2.1. Expressions et déclarations
 - 3.2.2. Variables et constantes
 - 3.2.3. Déclaration *if*
 - 3.2.4. Déclarations *for*, *while*
- 3.3. Fonctions
 - 3.3.1. Fonctions
 - 3.3.2. Paramètres
 - 3.3.3. Fonctions en tant que paramètres
 - 3.3.4. Champ d'application des variables
 - 3.3.5. *Scopes* imbriquées
 - 3.3.6. *Hoisting*
 - 3.3.7. *Fermetures*
 - 3.3.8. Récursion
- 3.4. Structures de données : objets
 - 3.4.1. Type *object*
 - 3.4.2. Création d'objets
 - 3.4.3. Accès aux valeurs d'un objet
 - 3.4.4. Ajout ou suppression de propriétés
 - 3.4.5. Objets imbriqués
 - 3.4.6. *Destructuring* d'objets
 - 3.4.7. Méthodes du type *object*
 - 3.4.8. *Spread Operator*
 - 3.4.9. Immutabilité
- 3.5. Structure des données : *Array*
 - 3.5.1. Structure des données. *Array*
 - 3.5.2. *Array*. Typologie
 - 3.5.3. *Arrays* emboîté
 - 3.5.4. Méthodes d'un *array*
- 3.6. POO: Prototype et classe
 - 3.6.1. POO. Programmation orientée objet
 - 3.6.2. Prototypes
 - 3.6.3. Classe
 - 3.6.4. Données privées
 - 3.6.5. Sous-classes
 - 3.6.6. *Call* et *apply*
- 3.7. Types de JavaScript
 - 3.7.1. *Set*
 - 3.7.2. *WeakSet*
 - 3.7.3. *Map*
 - 3.7.4. *WeakMap*
 - 3.7.5. Expressions régulières
- 3.8. Utilitaires JavaScript
 - 3.8.1. *Date*
 - 3.8.2. *Math*
 - 3.8.3. *Symbol*
 - 3.8.4. JSON
- 3.9. JavaScript dans le *browser*
 - 3.9.1. Inclusion de JavaScript dans un site web
 - 3.9.2. DOM
 - 3.9.3. Événements
 - 3.9.4. *Storage* du navigateur

- 3.10. Programmation asynchrone
 - 3.10.1. Programmation asynchrone
 - 3.10.2. *Event loop*
 - 3.10.3. *Callbacks*
 - 3.10.4. *Promises*
 - 3.10.5. *Async/await*

Module 4. Mise en page Web Appliqué à Développeur Full Stack

- 4.1. CSS et mise en page
 - 4.1.1. Mise en page avec des tableaux
 - 4.1.2. Conception des fluides
 - 4.1.3. L'ère du *Responsive*
 - 4.1.4. *Mobile First* vs *desktop First*
- 4.2. CSS et les règles de la conception web
 - 4.2.1. Sélecteurs
 - 4.2.2. Pseudo classes
 - 4.2.3. Pseudo-éléments
- 4.3. Mise en page avec CSS
 - 4.3.1. Règles du *box model*
 - 4.3.2. Typographie
 - 4.3.3. Couleurs
 - 4.3.4. Images
 - 4.3.5. Fonds
 - 4.3.6. Tableaux
 - 4.3.7. Formulaire
 - 4.3.8. Afficher et masquer des éléments
 - 4.3.9. Variables CSS
- 4.4. Design *responsive* et design fluide
 - 4.4.1. Éléments flottants
 - 4.4.2. *Grid* CSS
 - 4.4.3. *Media Queries*
 - 4.4.4. Flex Box
- 4.5. La cascade CSS
 - 4.5.1. Priorité des règles CSS
 - 4.5.2. Règles dérogatoires
 - 4.5.3. Classes vs identifiants
- 4.6. SaSS
 - 4.6.1. Logiciel en tant que service (SaSS)
 - 4.6.2. Installation de SaSS
 - 4.6.3. Exécution et compilation de SaSS
 - 4.6.4. Structure d'un répertoire SaSS
- 4.7. Utilisation de SaSS
 - 4.7.1. Variables dans Sass
 - 4.7.2. Modulaire de notre projet
 - 4.7.3. Syntaxe SaSS
- 4.8. Logique SaSS
 - 4.8.1. Mixins
 - 4.8.2. Maps
 - 4.8.3. Fonctions et structures de contrôle
- 4.9. Composition avec *bootstrap*
 - 4.9.1. *Bootstrap*
 - 4.9.2. *Layout* de *bootstrap*
 - 4.9.3. Formulaire
 - 4.9.4. *Box model* avec *bootstrap*
 - 4.9.5. Couleurs et polices
 - 4.9.6. Liens et boutons
 - 4.9.7. Afficher et masquer les éléments avec *bootstrap*
 - 4.9.8. *Flex box* avec *bootstrap*
 - 4.9.9. Composants
- 4.10. *Theming bootstrap*
 - 4.10.1. Réécriture de *bootstrap* avec SaSS (*Software as a Service*)
 - 4.10.2. Structure des fichiers
 - 4.10.3. Création de notre propre *framework* CSS (*Cascading style sheets*)

Module 5. Outils de JavaScript. Bibliothèque ReactJs

- 5.1. Outil Javascript ReactJS
 - 5.1.1. L'outil ReactJS
 - 5.1.2. Create React App
 - 5.1.3. JavaScript *syntax extension*
- 5.2. Composants ReactJS
 - 5.2.1. Composants
 - 5.2.2. *Props*
 - 5.2.3. Rendering
- 5.3. Événements dans la bibliothèque ReactJS
 - 5.3.1. Gestion des événements
 - 5.3.2. Gestion des événements en ligne
 - 5.3.3. Événements dans la bibliothèque ReactJS
- 5.4. Configuration *hooks* de ReactJS
 - 5.4.1. États d'un composant
 - 5.4.2. *Hook* d'état
 - 5.4.3. *Hook* à effet
 - 5.4.4. *Custom Hooks*
 - 5.4.5. Autres *hooks*
- 5.5. *Component context* dans ReactJS
 - 5.5.1. *Component context* dans ReactJS
 - 5.5.2. Utilisation du *context*
 - 5.5.3. Structure du *context*
 - 5.5.4. *React. createContext*
 - 5.5.5. *Context .Provider*
 - 5.5.6. *Class. Context type*
 - 5.5.7. *Context. Consumer*
 - 5.5.8. *Context.displayName*
 - 5.5.9. Application pratique d'utilisation du *context*



- 5.6. Routage dans ReactJs
 - 5.6.1. *Routeur*
 - 5.6.2. *React router*
 - 5.6.3. Installation
 - 5.6.4. Routage de base
 - 5.6.5. Routage dynamique
 - 5.6.6. Composants primaires
 - 5.6.7. *React router hooks*
- 5.7. Utilisation de listes et de formulaires avec ReactJS
 - 5.7.1. Listes et boucles
 - 5.7.2. Formulaires et validations
 - 5.7.3. *React hook forms*
- 5.8. Utilisation des styles dans ReactJS
 - 5.8.1. Stylisé traditionnel
 - 5.8.2. Styles en ligne
 - 5.8.3. Agrégat de bibliothèque système de conception
- 5.9. Test en Javascript. Outils
 - 5.9.1. *Testing*
 - 5.9.2. Jest JavaScript *testing framework*
 - 5.9.3. *Visual testing and documentation*
- 5.10. Déploiement de code avec ReactJS
 - 5.10.1. *Hosting*
 - 5.10.2. Fournisseurs
 - 5.10.3. Préparation du projet
 - 5.10.4. Déploiement à Heroku

Module 6. Framework de JavaScript. Angular

- 6.1. Le *framework* angulaire et son architecture
 - 6.1.1. Angular CLI
 - 6.1.2. Architecture
 - 6.1.3. *Workspace* et structure
 - 6.1.4. Environnement
- 6.2. Composants du *Framework* Angular
 - 6.2.1. Cycle de vie
 - 6.2.2. Encapsulation de la vue
 - 6.2.3. Interaction entre les composants
 - 6.2.4. Projection de contenu
- 6.3. Modèles de *framework* angular
 - 6.3.1. Interpolation de texte
 - 6.3.2. Déclarations
 - 6.3.3. *Property binding*
 - 6.3.4. *Class, style* et *attribute binding*
 - 6.3.5. *Event binding* et *two - way binding*
 - 6.3.6. *Pipes*
- 6.4. Directive de *framework* angular
 - 6.4.1. Stratégies angular
 - 6.4.2. Stratégies d'attribut
 - 6.4.3. Stratégies de structure
- 6.5. Services d'injection de dépendances
 - 6.5.1. Services
 - 6.5.2. Injection de dépendances
 - 6.5.3. *Service Providers*

- 6.6. *Routing* et navigation
 - 6.6.1. Application avec *Routing*
 - 6.6.2. Routage de base
 - 6.6.3. Itinéraires imbriqués
 - 6.6.4. Paramètres
 - 6.6.5. Accès et autorisation
 - 6.6.6. *Lazy Loading* de modules
- 6.7. RxJS
 - 6.7.1. Observables
 - 6.7.2. Observers
 - 6.7.3. Abonnements
 - 6.7.4. Opérateurs
- 6.8. Formulaires et HTTP
 - 6.8.1. Formulaires réactifs
 - 6.8.2. Validation des champs
 - 6.8.3. Formulaires dynamiques
 - 6.8.4. Pétitions
 - 6.8.5. *Intercepteurs*
 - 6.8.6. Sécurité
- 6.9. Animations
 - 6.9.1. Transitions et *triggers*
 - 6.9.2. Transitions de gamme
 - 6.9.3. Différences entre les transitions
- 6.10. *Testing* dans le *framework* angular
 - 6.10.1. Testage des services
 - 6.10.2. Testage des composants
 - 6.10.3. Testage des stratégies et des *pipes*

Module 7. Programmation en langage NodeJS

- 7.1. NodeJS et son architecture
 - 7.1.1. NPM et gestion des paquets
 - 7.1.2. Exécution d'un programme
 - 7.1.3. Modules
 - 7.1.4. Création d'un module
 - 7.1.5. *Loop* des événements
- 7.2. Servidor *backend*, HTTP, *express* et *sockets*
 - 7.2.1. Module HTTP
 - 7.2.2. Express
 - 7.2.3. Socket.io
- 7.3. Bases de données et cache
 - 7.3.1. MongoDB
 - 7.3.2. Mongoose
 - 7.3.3. SQL
 - 7.3.4. Sequelize
 - 7.3.5. Redis
- 7.4. Système de fichiers et Os
 - 7.4.1. *File system module*
 - 7.4.2. *Os module*
 - 7.4.3. *Cluster module*
- 7.5. Événements, *buffers* et *streams*
 - 7.5.1. Événements
 - 7.5.2. *Buffers*
 - 7.5.3. *Streams*
- 7.6. *Testing*
 - 7.6.1. Jest
 - 7.6.2. Mocha
 - 7.6.3. TDD - *Cucumber*

- 7.7. Architecture et bonnes pratiques
 - 7.7.1. DRY
 - 7.7.2. SOLID
 - 7.7.3. CRUD
 - 7.7.4. MVC
 - 7.7.5. Monolithes
 - 7.7.6. Microservices
 - 7.7.7. Architectures hexagonales
- 7.8. Typescript
 - 7.8.1. Types, interfaces et classes
 - 7.8.2. Fonctions et modules
 - 7.8.3. Génériques
 - 7.8.4. *Namespaces*
 - 7.8.5. Décorateurs
- 7.9. API REST
 - 7.9.1. Get
 - 7.9.2. Post
 - 7.9.3. Put
 - 7.9.4. Delete
 - 7.9.5. *Swagger*
 - 7.9.6. Construction d'une API REST avec express
- 7.10. Création et conteneurisation d'une application avec NestJS
 - 7.10.1. Nest CLI
 - 7.10.2. *Docker*
 - 7.10.3. Construction d'une application

Module 8. Bases de données pour développeurs *Full Stack*

- 8.1. Bases de données pour développeurs *Full Stack*
 - 8.1.1. Base de données dans le développement d'applications
 - 8.1.2. Capacités de bases de données
 - 8.1.3. SQL (*Structured query language*)
- 8.2. Choix de la base de données
 - 8.2.1. Application ou service à considérer
 - 8.2.2. Catégories de base de données
 - 8.2.3. Panorama de base de données
- 8.3. Développement avec MySQL
 - 8.3.1. Développement avec MySQL
 - 8.3.2. Déploiement de modèle relationnel avec MySQL
 - 8.3.3. Connexion à MySQL
- 8.4. Développement avec Oracle Database
 - 8.4.1. Développement avec Oracle DB
 - 8.4.2. Déploiement du modèle
 - 8.4.3. Connexion à Oracle Database
- 8.5. Développement avec Oracle SQL *server*
 - 8.5.1. Oracle SQL *server*
 - 8.5.2. Déploiement du modèle
 - 8.5.3. Connexion à SQL *server*
- 8.6. Développement avec NoSQL
 - 8.6.1. Comparaison avec les bases de données SQL
 - 8.6.2. Création de base de données dans MongoDB
 - 8.6.3. Connexion à MongoDB

- 8.7. Développement avec graphes
 - 8.7.1. Développement avec graphes
 - 8.7.2. Création de base de données avec Neo4j
 - 8.7.3. Connexion à Neo4j
- 8.8. Développement avec base de données clé-valeur
 - 8.8.1. Développement avec base de données k-v
 - 8.8.2. Création d'une base de données Redis
 - 8.8.3. Connexion Redis
- 8.9. Bases de données avec autres types de données
 - 8.9.1. *Elastic search*
 - 8.9.2. *Inmemory Database*
 - 8.9.3. Développement des données spatiales
- 8.10. Bases de données. Aspects avancés
 - 8.10.1. Bases de données dans les développements *cloud* natif
 - 8.10.2. Bases de données d'architecture microservices
 - 8.10.3. CI/CD et bases de données

Module 9. UX CX. Expérience client

- 9.1. *Expérience client*
 - 9.1.1. *Customer experience (CX)*
 - 9.1.2. Nouveaux besoins de consommation
 - 9.1.3. *Feedback* dans *customer experience*
- 9.2. Technologies novatrices
 - 9.2.1. Machines à penser
 - 9.2.2. Nouvelles façons de partager des informations
 - 9.2.3. Mesure de ce qui ne peut pas être mesuré
- 9.3. Canaux d'interaction utilisateur
 - 9.3.1. Analyse client
 - 9.3.2. Personnalisation
 - 9.3.3. Plusieurs canaux d'interaction utilisateur
- 9.4. Analyse utilisateur
 - 9.4.1. Structure du site web
 - 9.4.2. Analyse utilisateur
 - 9.4.3. Analyse utilisateur avancée

- 9.5. Nielsen et son impact sur la CX
 - 9.5.1. Nielsen et son impact sur la CX
 - 9.5.2. Techniques de *user testing*
- 9.6. Outils de *customer experience*
 - 9.6.1. Outils avancés
 - 9.6.2. Mobilité
 - 9.6.3. Accessibilité
- 9.7. Nouvelles méthodes
 - 9.7.1. Le défi de l'utilisateur
 - 9.7.2. Processus UX
 - 9.7.3. Recherche utilisateur
- 9.8. Communication d'une conception
 - 9.8.1. *Wireframing*
 - 9.8.2. Outils de communication d'une conception
 - 9.8.3. Outils de communication avancés d'une conception
- 9.9. Conception d'UI
 - 9.9.1. Conception d'UI
 - 9.9.2. Interfaces web et mobiles
 - 9.9.3. Composants web et mobiles
- 9.10. Élaboration d'une CX
 - 9.10.1. Élaboration d'une CX
 - 9.10.2. Conception de nouvelles expériences
 - 9.10.3. Interfaces

Module 10. Intégration continue et déploiement d'applications

- 10.1. Intégration continue et déploiement continu : CI/CD
 - 10.1.1. Utilisation de l'intégration continue et du déploiement continu (CI/CD)
 - 10.1.2. Différences entre intégration continue et déploiement continu (CI/CD)
 - 10.1.3. Intégration continue et déploiement continu. Avantages du CI/CD
- 10.2. Les nouveaux paradigmes de développement
 - 10.2.1. De *waterfall* à DevOps
 - 10.2.2. Guide de style : les 12 facteurs
 - 10.2.3. *Cloud native*, microservices et serverless

- 10.3. DevOps, au-delà de CI/CD
 - 10.3.1. DevOps
 - 10.3.2. DevOps. *Continuous everything*
 - 10.3.3. DevOps vs SRE
- 10.4. Technologie de conteneurs I - Docker
 - 10.4.1. Les conteneurs. Contributions
 - 10.4.2. Docker. Architecture
 - 10.4.3. Processus de déploiement avec Docker
- 10.5. Technologie de conteneurs II - Kubernetes
 - 10.5.1. Orchestration
 - 10.5.2. Kubernetes
 - 10.5.3. L'écosystème de Kubernetes
- 10.6. Configuration de l'infrastructure avec GitOps
 - 10.6.1. Infrastructure immuable
 - 10.6.2. GitOps
 - 10.6.3. Outils GitOps
- 10.7. Pipelines et automatisation. Cas d'Utilisation de CI/CD
 - 10.7.1. Intégration continue
 - 10.7.2. Déploiement et livraison continue
 - 10.7.3. Validations automatiques
 - 10.7.4. Bonnes pratiques en matière de CI/CD
- 10.8. CI/CD avec Jenkins. Référents
 - 10.8.1. CI/CD avec Jenkins
 - 10.8.2. Pipelines Jenkins
 - 10.8.3. Bonnes pratiques avec Jenkins
- 10.9. Écosystème CI/CD
 - 10.9.1. Organisation des écosystèmes
 - 10.9.2. Outils avancés
 - 10.9.3. *Dagger*. Le futur
- 10.10. Phases finales du cycle du *logiciel* orienté CI/CD
 - 10.10.1. Application de l'IA au processus CI/CD
 - 10.10.2. DevSecOps
 - 10.10.3. *Chaos engineering*



Se mettre à jour dans l'utilisation des bases de données orientées vers les développeurs Full Stack, clé pour gérer efficacement l'information dans les environnements web

04

Objectifs pédagogiques

L'objectif principal de ce programme universitaire est de renforcer les compétences techniques qui permettront aux professionnels de développer des solutions numériques. À cette fin, la priorité sera donnée à la maîtrise des langages et des *frameworks* de programmation actuels, ainsi qu'à une compréhension approfondie des algorithmes et des structures de données appliqués à des environnements réels. Grâce à cette approche, la capacité à concevoir des systèmes fonctionnels alignés sur les besoins du secteur technologique sera favorisée.



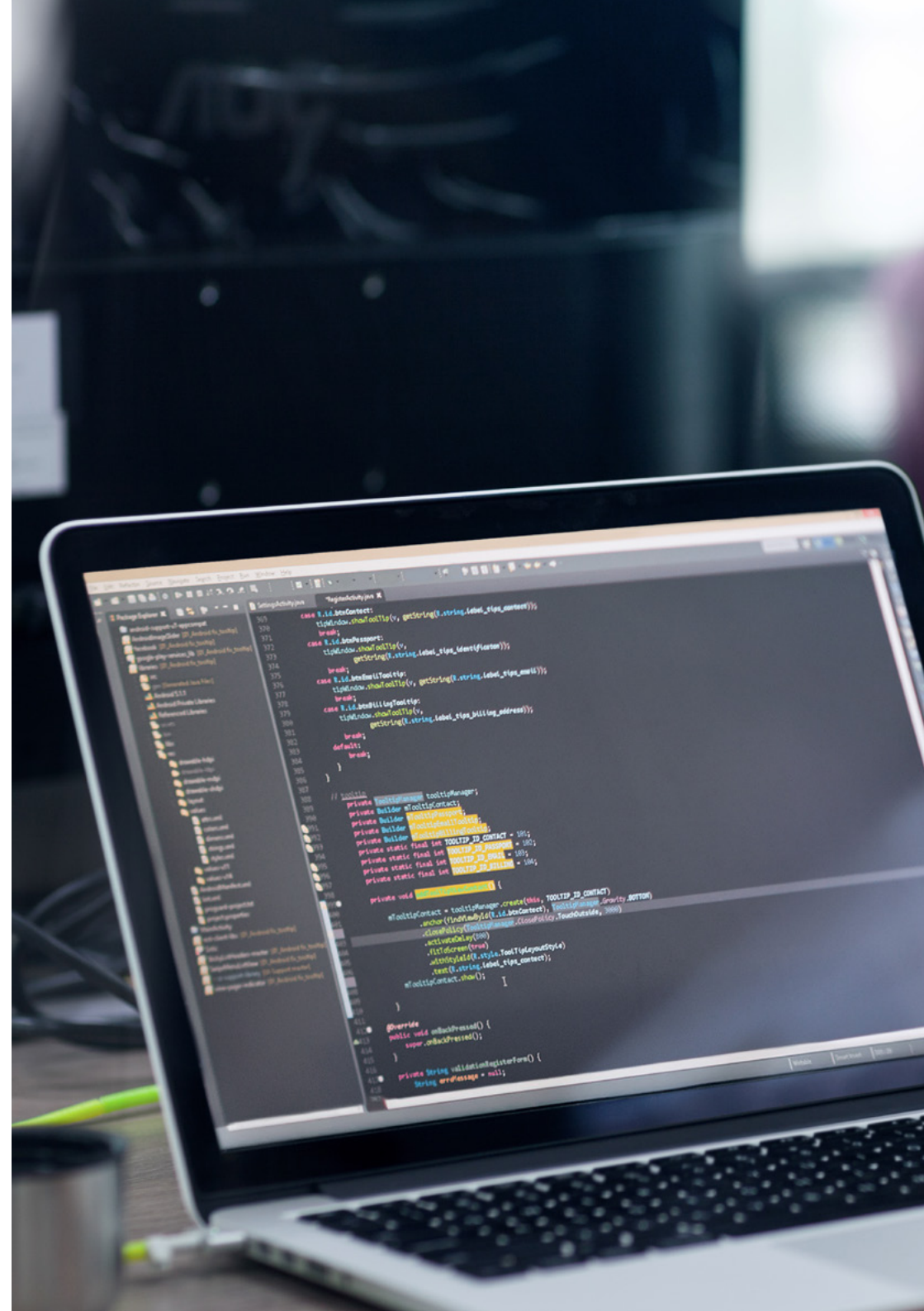
“

Vous évalueriez l'impact de l'utilisation des frameworks actuels dans les environnements de programmation, en identifiant leurs avantages opérationnels”



Objectifs généraux

- Développer des compétences intégrales dans l'environnement *Full Stack*, couvrant la conception de l'interface ainsi que la logique du serveur et la gestion des données
- Consolider la gestion des structures *front-end*, en intégrant les bonnes pratiques de programmation pour garantir des interfaces fonctionnelles et efficaces
- Appliquer le langage JavaScript dans la construction d'applications dynamiques, en intégrant une logique avancée et des composants réutilisables
- Mettre en œuvre des techniques de mise en page web orientées vers l'adaptabilité, l'expérience utilisateur et la cohérence visuelle du projet
- Intégrer des bibliothèques telles que ReactJS pour optimiser les performances et l'évolutivité des applications développées
- Utiliser des *frameworks* tels qu'Angular pour structurer des applications complexes, en maintenant un flux de travail organisé et durable
- Utiliser l'environnement NodeJS pour la programmation côté serveur, en assurant une communication efficace entre le client et la base de données
- Gérer les bases de données et les processus d'intégration continue, en assurant l'automatisation du déploiement et la qualité du développement *Full Stack*





Objectifs spécifiques

Module 1. Développeur *Full Stack Developer*

- ♦ Utiliser des algorithmes et des structures de données pour résoudre des problèmes dans des environnements de développement *Full Stack*
- ♦ Reconnaître les différents rôles au sein d'une équipe de programmation et leur impact sur le développement du projet
- ♦ Gérer les versions du code à l'aide de Git et de services d'hébergement collaboratifs
- ♦ Mettre en œuvre des méthodologies agiles pour optimiser l'organisation et l'exécution des tâches dans les projets technologiques

Module 2. *Front-end* dans Programmation

- ♦ Structurer le contenu web à l'aide des éléments HTML fondamentaux et des principes de conception sémantique
- ♦ Appliquer les feuilles de style en cascade (CSS) pour définir l'aspect visuel des sites web, y compris les animations et les transitions
- ♦ Mettre en œuvre des modèles de boîtes et des techniques de positionnement pour construire des interfaces soignées et fonctionnelles
- ♦ Utiliser les principes de la conception réactive et les Media Queries pour garantir un affichage correct sur différents appareils
- ♦ Différencier et appliquer les systèmes de mise en page modernes tels que Flexbox et Grid, en optimisant l'organisation du contenu
- ♦ Intégrer des préprocesseurs tels que Sass et des *frameworks* tels que Bootstrap pour rationaliser le développement et maintenir la cohérence visuelle

Module 3. Le langage JAVASCRIPT appliqué au *Full Stack Developer*

- ♦ Maîtriser les types primitifs JavaScript et leurs opérateurs pour établir une base solide dans le traitement des données et les structures logiques
- ♦ Concevoir des fonctions avancées incluant les *closures*, la récursivité et la gestion de l'étendue, en renforçant le contrôle sur le flux d'exécution
- ♦ Manipuler des structures complexes telles que les objets, les *arrays*, les *sets* et les *maps*, en appliquant des méthodes modernes d'accès, de transformation et d'immutabilité
- ♦ Gérer l'ordonnancement asynchrone à l'aide de *callbacks*, de promesses et de fonctions *async/await*, en optimisant la réponse et l'efficacité dans des environnements dynamiques

Module 4. Mise en page web appliquée au *Full Stack Developer*

- ♦ Mettre en œuvre des conceptions réactives à l'aide de CSS Grid, Flexbox et Media Queries, en adaptant les interfaces web à plusieurs appareils et résolutions
- ♦ Construire des structures visuelles organisées en appliquant les principes de mise en page, la conception fluide et les règles du modèle de boîte
- ♦ Utiliser des préprocesseurs tels que Sass pour optimiser le code CSS à l'aide de variables, de mixins, de fonctions et de la modularisation du projet
- ♦ Intégrer Bootstrap dans des projets web, en personnalisant leur *theming* avec Sass pour développer des composants réutilisables et des systèmes visuels cohérents

Module 5. Outils de JavaScript. Bibliothèque ReactJs

- ♦ Construire des interfaces dynamiques à l'aide de composants réutilisables, en gérant leur état et leur cycle de vie à l'aide de *Hooks* intégrés et personnalisés
- ♦ Mettre en œuvre une navigation transparente dans les applications à page unique à l'aide de React Router et de ses fonctions de routage dynamique
- ♦ Gérer les flux de données partagés entre les composants grâce à une utilisation efficace de l'API Contexte et de ses méthodes associées
- ♦ Préparer, tester et déployer des projets développés avec ReactJS en utilisant des outils tels que Jest et des plateformes telles que Heroku

Module 6. Framework de JavaScript. Angular

- ♦ Concevoir des applications modulaires et évolutives en Angular, en comprenant son architecture, son CLI et les principes d'organisation du projet
- ♦ Mettre en œuvre une interaction avancée entre les composants, les directives et les modèles en utilisant des techniques telles que le data *binding*, la projection de contenu et les pipes personnalisés
- ♦ Gérer les services, les dépendances et l'intégration avec les formulaires et les requêtes HTTP grâce à l'injection de dépendances et à l'utilisation d'intercepteurs
- ♦ Intégration de la navigation dynamique, de la sécurité, du *Lazy Loading* et des animations, optimisation de l'expérience de l'utilisateur dans les applications à page unique

Module 7. Programmation en langage NodeJS

- ♦ Comprendre l'architecture NodeJS et son fonctionnement à l'aide des modules, de NPM et de la boucle d'événements
- ♦ Créer des serveurs avec Express et Socket.io, en intégrant la gestion des requêtes HTTP et la communication en temps réel
- ♦ Connecter les applications aux bases de données avec Mongoose et Sequelize, en gérant efficacement les données
- ♦ Concevoir des API RESTful et les déployer avec NestJS et Docker, en appliquant les principes de l'architecture moderne

Module 8. Bases de données pour développeurs *Full Stack*

- ♦ Distinguer les différents types de bases de données et sélectionner l'option la plus appropriée en fonction des besoins de développement
- ♦ Mettre en œuvre des modèles relationnels et connecter des applications à des systèmes tels que MySQL, Oracle et SQL Server
- ♦ Développer des projets avec des bases de données NoSQL, graphiques et key-value, en intégrant des solutions telles que MongoDB, Neo4j et Redis
- ♦ Intégrer les bases de données dans les environnements *cloud-native*, les microservices et les flux CI/CD avec une approche avancée

Module 9. UX CX. Expérience client

- ♦ Comprendre l'évolution du *Customer Experience* et son impact sur les nouvelles habitudes des consommateurs
- ♦ Intégrer des technologies innovantes pour améliorer l'interaction avec l'utilisateur et personnaliser l'expérience
- ♦ Analyser le comportement des utilisateurs à l'aide d'outils d'analyse et de tests de convivialité avancés
- ♦ Concevoir des expériences centrées sur l'utilisateur en appliquant des méthodologies UX, UI et de communication visuelle efficace

Module 10. Intégration continue et déploiement d'applications

- ♦ Distinguer les phases CI/CD et leur intégration dans les nouveaux modèles de développement orientés DevOps
- ♦ Utiliser des outils tels que Docker, Kubernetes et Jenkins pour automatiser les processus de déploiement et d'orchestration
- ♦ Appliquer les meilleures pratiques d'infrastructure telles que le code via GitOps et les validations automatisées
- ♦ Incorporer des approches avancées telles que DevSecOps, Chaos Engineering et AI dans les environnements d'intégration continue

05

Opportunités de carrière

Cette formation universitaire vous apportera l'expérience académique nécessaire pour accéder à des postes tels qu'architecte *logiciel*, responsable de l'automatisation ou spécialiste de l'infrastructure *cloud*. Grâce à vos compétences, vous pourrez rejoindre des centres de développement technologique, des entreprises de solutions numériques ou des projets dans des entreprises mondiales. En outre, cela facilitera l'adaptation à des environnements où l'efficacité opérationnelle, l'intégration continue et le déploiement automatisé représentent des avantages concurrentiels. Cela ouvrira des possibilités concrètes de mener des initiatives stratégiques, d'optimiser les processus techniques et de participer activement à la conception de systèmes évolutifs et sûrs.



“

*Vous souhaitez travailler en tant que
Développeur d'Applications Web Réactives ?
Réalisez-le avec cette formation universitaire
en seulement 12 mois”*

Profil des diplômés

Le diplômé de ce programme se caractérisera par une grande capacité à développer des applications web complètes, tant du côté client que du côté serveur. Tout au long du programme universitaire, ils intégreront des techniques modernes de mise en page, de gestion des API et de manipulation des bases de données. Vous serez ainsi en mesure de mettre en œuvre des solutions interactives, optimisées et adaptables à de multiples appareils. D'autre part, vous maîtriserez les outils de contrôle de version, les tests automatisés et le déploiement continu. Enfin, vous serez prêt à vous intégrer dans des projets réels, à mener des développements évolutifs et à vous adapter rapidement aux nouvelles technologies de l'environnement numérique.

Vous assumerez des rôles clés dans la manipulation des bases de données, en optimisant les requêtes et les structures avec précision.

- ♦ **Pensée logique** : Capacité à structurer des solutions efficaces et cohérentes à des problèmes complexes, facilitant le développement d'un code propre, fonctionnel et évolutif
- ♦ **Communication efficace** : Aptitude qui favorise la collaboration avec des équipes pluridisciplinaires, permettant de transmettre clairement des idées techniques, tant en face à face que dans des environnements distants
- ♦ **Adaptabilité technologique** : Capacité à faciliter l'intégration rapide de nouveaux outils, *frameworks* et environnements numériques en constante évolution dans l'écosystème du développement web
- ♦ **Gestion du temps** : Compétence pour organiser les tâches et respecter les délais dans le cadre de projets agiles, en optimisant les ressources et en améliorant la productivité dans des environnements exigeants





À l'issue de ce programme, vous serez en mesure d'utiliser vos connaissances et vos compétences dans les postes suivants :

- 1. Développeur Full Stack** : Responsable de la conception et de la réalisation d'applications complètes, de l'interface visuelle à la logique du serveur et à la gestion de la base de données.
- 2. Front End Developer** : Il se consacre à la création d'interfaces interactives, accessibles et adaptables, en optimisant l'expérience de l'utilisateur sur les navigateurs et les appareils.
- 3. Back End Developer** : Chargé de la logique du serveur, de la gestion de la base de données et du développement de l'API, il garantit le bon fonctionnement du système.
- 4. Ingénieur Logiciel Web** : Responsable du développement et de la maintenance d'applications web complexes, en mettant l'accent sur l'évolutivité, la sécurité et les performances.
- 5. Développeur d'Applications Web** : Il se consacre à la création de plateformes numériques fonctionnelles grâce à des technologies modernes et à des *frameworks* de travail actualisés tels que React, Angular ou Vue.
- 6. Architecte Logiciel** : Chargé de définir la structure technique des projets, de sélectionner les outils, les langages et les modèles qui garantissent la qualité et la durabilité.
- 7. Spécialiste DevSecOps** : Responsable de la mise en œuvre des pratiques de sécurité tout au long du cycle de développement, intégrant la protection depuis l'écriture du code jusqu'au déploiement.
- 8. Administrateur de Base de Données** : Gère l'intégrité des données, facilite la disponibilité des données et le basculement.
- 9. Concepteur d'Interface** : Il est chargé de concevoir des composants visuels intuitifs, conformes à la convivialité et à l'identité visuelle du produit numérique.
- 10. Consultant en Transformation Numérique** : Chargé de conseiller les entreprises sur l'adoption de solutions numériques, en améliorant les processus et les services grâce à des outils technologiques innovants.

06

Méthodologie d'étude

TECH est la première université au monde à combiner la méthodologie des **case studies** avec **Relearning**, un système d'apprentissage 100 % en ligne basé sur la répétition guidée.

Cette stratégie d'enseignement innovante est conçue pour offrir aux professionnels la possibilité d'actualiser leurs connaissances et de développer leurs compétences de manière intensive et rigoureuse. Un modèle d'apprentissage qui place l'étudiant au centre du processus académique et lui donne le rôle principal, en s'adaptant à ses besoins et en laissant de côté les méthodologies plus conventionnelles.



“

*TECH vous prépare à relever de nouveaux défis
dans des environnements incertains et à réussir
votre carrière”*

L'étudiant : la priorité de tous les programmes de TECH

Dans la méthodologie d'étude de TECH, l'étudiant est le protagoniste absolu. Les outils pédagogiques de chaque programme ont été sélectionnés en tenant compte des exigences de temps, de disponibilité et de rigueur académique que demandent les étudiants d'aujourd'hui et les emplois les plus compétitifs du marché.

Avec le modèle éducatif asynchrone de TECH, c'est l'étudiant qui choisit le temps qu'il consacre à l'étude, la manière dont il décide d'établir ses routines et tout cela dans le confort de l'appareil électronique de son choix. L'étudiant n'a pas besoin d'assister à des cours en direct, auxquels il ne peut souvent pas assister. Les activités d'apprentissage se dérouleront à votre convenance. Vous pouvez toujours décider quand et où étudier.

“

*À TECH, vous n'aurez PAS de cours en direct
(auxquelles vous ne pourrez jamais assister)”*



Les programmes d'études les plus complets au niveau international

TECH se caractérise par l'offre des itinéraires académiques les plus complets dans l'environnement universitaire. Cette exhaustivité est obtenue grâce à la création de programmes d'études qui couvrent non seulement les connaissances essentielles, mais aussi les dernières innovations dans chaque domaine.

Grâce à une mise à jour constante, ces programmes permettent aux étudiants de suivre les évolutions du marché et d'acquérir les compétences les plus appréciées par les employeurs. Ainsi, les diplômés de TECH reçoivent une préparation complète qui leur donne un avantage concurrentiel significatif pour progresser dans leur carrière.

De plus, ils peuvent le faire à partir de n'importe quel appareil, PC, tablette ou smartphone.

“

Le modèle de TECH est asynchrone, de sorte que vous pouvez étudier sur votre PC, votre tablette ou votre smartphone où vous voulez, quand vous voulez et aussi longtemps que vous le voulez”

Case studies ou Méthode des cas

La méthode des cas est le système d'apprentissage le plus utilisé par les meilleures écoles de commerce du monde. Développée en 1912 pour que les étudiants en Droit n'apprennent pas seulement le droit sur la base d'un contenu théorique, sa fonction était également de leur présenter des situations réelles et complexes. De cette manière, ils pouvaient prendre des décisions en connaissance de cause et porter des jugements de valeur sur la manière de les résoudre. Elle a été établie comme méthode d'enseignement standard à Harvard en 1924.

Avec ce modèle d'enseignement, ce sont les étudiants eux-mêmes qui construisent leurs compétences professionnelles grâce à des stratégies telles que *Learning by doing* ou le *Design Thinking*, utilisées par d'autres institutions renommées telles que Yale ou Stanford.

Cette méthode orientée vers l'action sera appliquée tout au long du parcours académique de l'étudiant avec TECH. Vous serez ainsi confronté à de multiples situations de la vie réelle et devrez intégrer des connaissances, faire des recherches, argumenter et défendre vos idées et vos décisions. Il s'agissait de répondre à la question de savoir comment ils agiraient lorsqu'ils seraient confrontés à des événements spécifiques complexes dans le cadre de leur travail quotidien.



Méthode Relearning

Chez TECH, les *case studies* sont complétées par la meilleure méthode d'enseignement 100% en ligne: le *Relearning*.

Cette méthode s'écarte des techniques d'enseignement traditionnelles pour placer l'apprenant au centre de l'équation, en lui fournissant le meilleur contenu sous différents formats. De cette façon, il est en mesure de revoir et de répéter les concepts clés de chaque matière et d'apprendre à les appliquer dans un environnement réel.

Dans le même ordre d'idées, et selon de multiples recherches scientifiques, la répétition est le meilleur moyen d'apprendre. C'est pourquoi TECH propose entre 8 et 16 répétitions de chaque concept clé au sein d'une même leçon, présentées d'une manière différente, afin de garantir que les connaissances sont pleinement intégrées au cours du processus d'étude.

Le Relearning vous permettra d'apprendre plus facilement et de manière plus productive tout en développant un esprit critique, en défendant des arguments et en contrastant des opinions : une équation directe vers le succès.



Un Campus Virtuel 100% en ligne avec les meilleures ressources didactiques

Pour appliquer efficacement sa méthodologie, TECH se concentre à fournir aux diplômés du matériel pédagogique sous différents formats: textes, vidéos interactives, illustrations et cartes de connaissances, entre autres. Tous ces supports sont conçus par des enseignants qualifiés qui axent leur travail sur la combinaison de cas réels avec la résolution de situations complexes par la simulation, l'étude de contextes appliqués à chaque carrière professionnelle et l'apprentissage basé sur la répétition, par le biais d'audios, de présentations, d'animations, d'images, etc.

Les dernières données scientifiques dans le domaine des Neurosciences soulignent l'importance de prendre en compte le lieu et le contexte d'accès au contenu avant d'entamer un nouveau processus d'apprentissage. La possibilité d'ajuster ces variables de manière personnalisée aide les gens à se souvenir et à stocker les connaissances dans l'hippocampe pour une rétention à long terme. Il s'agit d'un modèle intitulé *Neurocognitive context-dependent e-learning* qui est sciemment appliqué dans le cadre de ce diplôme universitaire.

D'autre part, toujours dans le but de favoriser au maximum les contacts entre mentors et mentorés, un large éventail de possibilités de communication est offert, en temps réel et en différé (messagerie interne, forums de discussion, service téléphonique, contact par courrier électronique avec le secrétariat technique, chat et vidéoconférence).

De même, ce Campus Virtuel très complet permettra aux étudiants TECH d'organiser leurs horaires d'études en fonction de leurs disponibilités personnelles ou de leurs obligations professionnelles. De cette manière, ils auront un contrôle global des contenus académiques et de leurs outils didactiques, mis en fonction de leur mise à jour professionnelle accélérée.



Le mode d'étude en ligne de ce programme vous permettra d'organiser votre temps et votre rythme d'apprentissage, en l'adaptant à votre emploi du temps"

L'efficacité de la méthode est justifiée par quatre acquis fondamentaux:

1. Les étudiants qui suivent cette méthode parviennent non seulement à assimiler les concepts, mais aussi à développer leur capacité mentale au moyen d'exercices pour évaluer des situations réelles et appliquer leurs connaissances.
2. L'apprentissage est solidement traduit en compétences pratiques ce qui permet à l'étudiant de mieux s'intégrer dans le monde réel.
3. L'assimilation des idées et des concepts est rendue plus facile et plus efficace, grâce à l'utilisation de situations issues de la réalité.
4. Le sentiment d'efficacité de l'effort investi devient un stimulus très important pour les étudiants, qui se traduit par un plus grand intérêt pour l'apprentissage et une augmentation du temps passé à travailler sur le cours.

La méthodologie universitaire la mieux évaluée par ses étudiants

Les résultats de ce modèle académique innovant sont visibles dans les niveaux de satisfaction générale des diplômés de TECH.

L'évaluation par les étudiants de la qualité de l'enseignement, de la qualité du matériel, de la structure du cours et des objectifs est excellente. Il n'est pas surprenant que l'institution soit devenue l'université la mieux évaluée par ses étudiants selon l'indice global score, obtenant une note de 4,9 sur 5.

Accédez aux contenus de l'étude depuis n'importe quel appareil disposant d'une connexion Internet (ordinateur, tablette, smartphone) grâce au fait que TECH est à la pointe de la technologie et de l'enseignement.

Vous pourrez apprendre grâce aux avantages offerts par les environnements d'apprentissage simulés et à l'approche de l'apprentissage par observation: le Learning from an expert.

Ainsi, le meilleur matériel pédagogique, minutieusement préparé, sera disponible dans le cadre de ce programme:



Matériel didactique

Tous les contenus didactiques sont créés par les spécialistes qui enseignent les cours. Ils ont été conçus en exclusivité pour le programme afin que le développement didactique soit vraiment spécifique et concret.

Ces contenus sont ensuite appliqués au format audiovisuel afin de mettre en place notre mode de travail en ligne, avec les dernières techniques qui nous permettent de vous offrir une grande qualité dans chacune des pièces que nous mettrons à votre service.



Pratique des aptitudes et des compétences

Vous effectuerez des activités visant à développer des compétences et des aptitudes spécifiques dans chaque domaine. Pratiques et dynamiques permettant d'acquérir et de développer les compétences et les capacités qu'un spécialiste doit acquérir dans le cadre de la mondialisation dans laquelle nous vivons.



Résumés interactifs

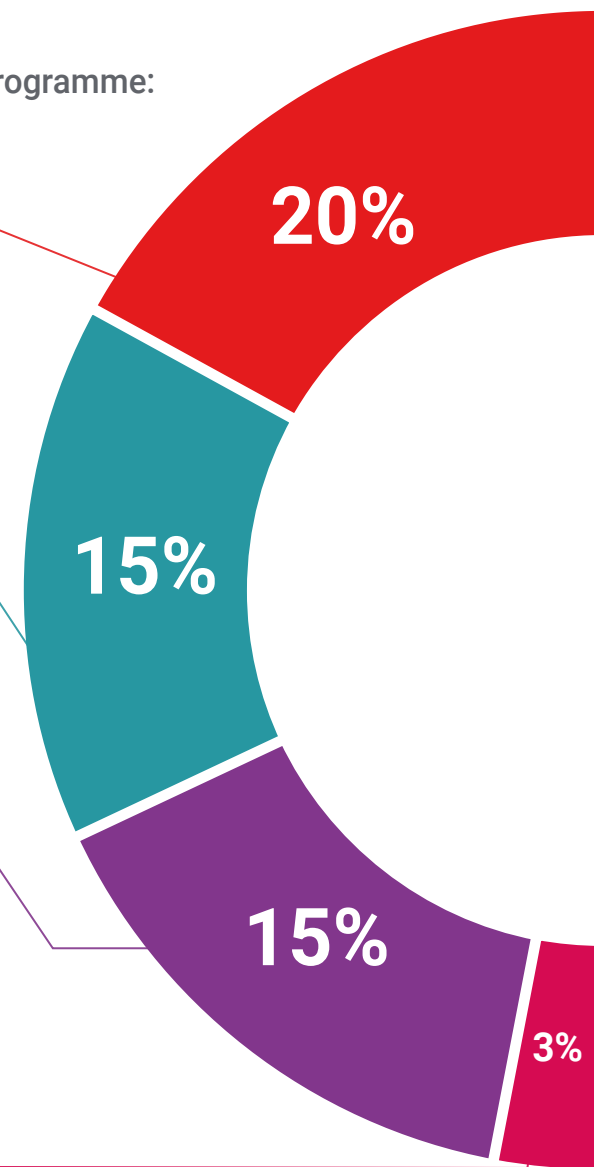
Nous présentons les contenus de manière attrayante et dynamique dans des dossiers multimédias qui incluent de l'audio, des vidéos, des images, des diagrammes et des cartes conceptuelles afin de consolider les connaissances.

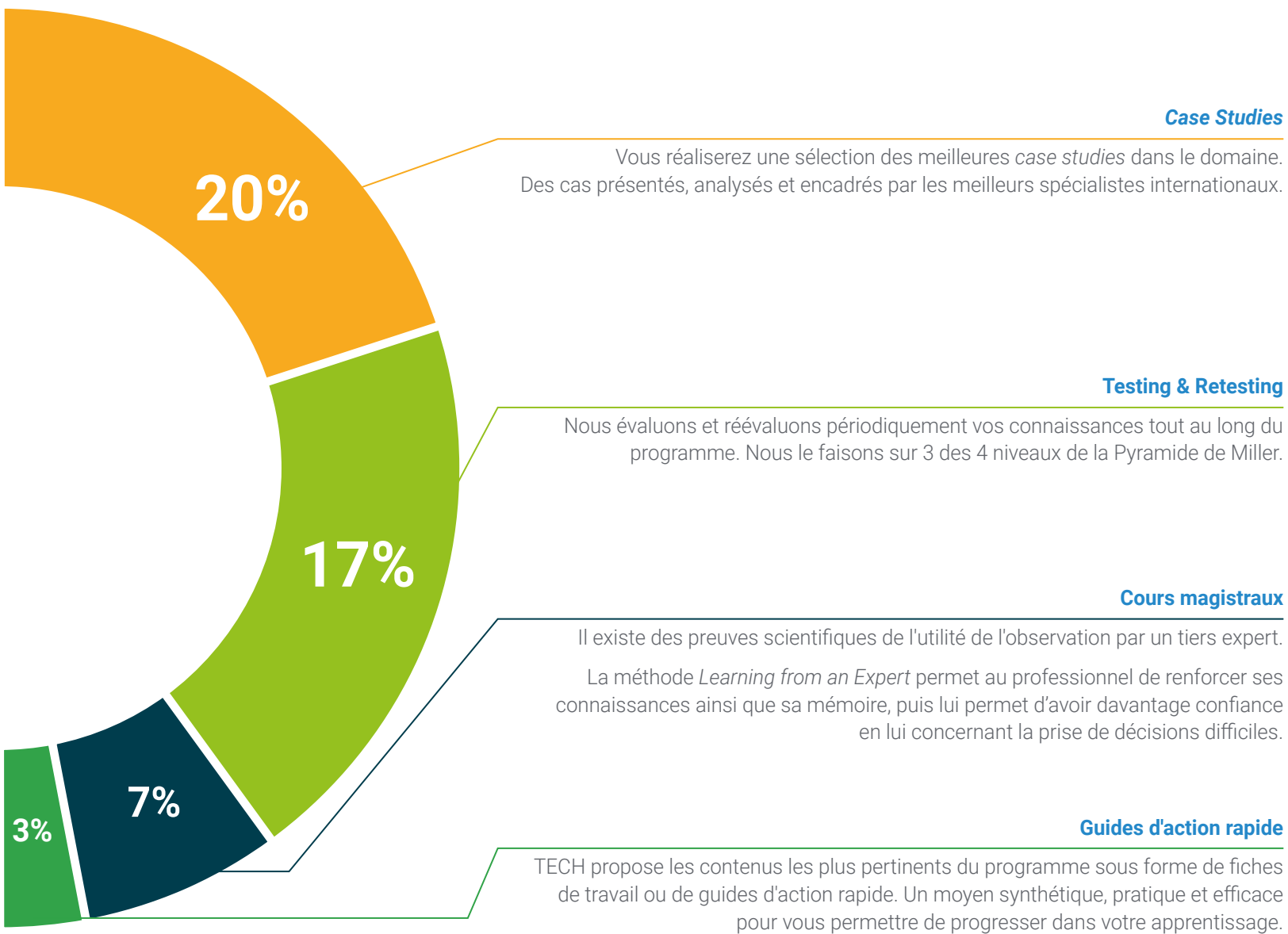
Ce système éducatif unique de présentation de contenu multimédia a été récompensé par Microsoft en tant que «European Success Story».



Lectures complémentaires

Articles récents, documents de consensus, guides internationaux, etc... Dans notre bibliothèque virtuelle, vous aurez accès à tout ce dont vous avez besoin pour compléter votre formation.





Case Studies

Vous réaliserez une sélection des meilleures *case studies* dans le domaine. Des cas présentés, analysés et encadrés par les meilleurs spécialistes internationaux.



Testing & Retesting

Nous évaluons et réévaluons périodiquement vos connaissances tout au long du programme. Nous le faisons sur 3 des 4 niveaux de la Pyramide de Miller.



Cours magistraux

Il existe des preuves scientifiques de l'utilité de l'observation par un tiers expert. La méthode *Learning from an Expert* permet au professionnel de renforcer ses connaissances ainsi que sa mémoire, puis lui permet d'avoir davantage confiance en lui concernant la prise de décisions difficiles.



Guides d'action rapide

TECH propose les contenus les plus pertinents du programme sous forme de fiches de travail ou de guides d'action rapide. Un moyen synthétique, pratique et efficace pour vous permettre de progresser dans votre apprentissage.



07

Corps Enseignant

TECH applique un processus rigoureux dans la sélection du personnel enseignant responsable de ses diplômes. Grâce à ce processus exigeant, des équipes académiques de haut niveau sont formées, engagées dans l'excellence et avec une vision orientée vers le développement professionnel des informaticiens. En conséquence, non seulement la solidité et l'actualité du contenu sont garanties, mais aussi une expérience académique soutenue par du matériel pédagogique de qualité, des ressources complémentaires et une approche personnalisée. Ainsi, les diplômés trouveront dans TECH Global University le soutien de la plus grande institution en ligne au monde, engagée dans leur croissance et leur projection professionnelle.



“

L'équipe enseignante, spécialisée dans la Programmation Front End Full Stack Developer, a conçu pour vous des heures de contenu additionnel afin d'approfondir chaque section du syllabus de manière personnalisée”

Direction



M. Olalla Bonal, Martín

- ♦ Responsable de la Pratique *Blockchain* chez EY
- ♦ Spécialiste Technique Client *Blockchain* pour IBM
- ♦ Directeur de l'Architecture de Blocknitive
- ♦ Coordinateur de l'Équipe Bases de Données Distribuées non Relationnelles pour wedoIT, Filiale d'IBM
- ♦ Architecte d'Infrastructure chez Bankia
- ♦ Chef du Département Mise en Page chez T-Systems
- ♦ Coordinateur de Département pour Bing Data España SL.

Professeurs

M. Guerrero Díaz-Pintado, Arturo

- ♦ Directeur de l'Expérience Client chez IBM
- ♦ Ingénieur Technique Avant-vente pour *Watson Customer Engagement* portefeuille
- ♦ Ingénieur réseau R&D chez Telefonica
- ♦ Licence en Ingénierie Supérieure des Télécommunications de l'Université d'Alcalá et de la *Danish Technical University*

M. Frias Favero, Pedro Luis

- ♦ Architecte Principal de la Blockchain chez EY
- ♦ Cofondateur et Directeur Technique de Swear IT Technologies
- ♦ Directeur de l'Assistance Informatique pour Key Business One. Mexique, Colombie et Espagne
- ♦ Diplôme d'Ingénieur Industriel de l'Université Yacambú
- ♦ Expert en Blockchain et Applications Décentralisées à l'Université d'Alcalá

M. Gómez Rodríguez, Antonio

- ♦ Ingénieur Principal de Solutions Cloud chez Oracle
- ♦ Co-organisateur de Malaga Developer Meetup
- ♦ Consultant Spécialisé pour Sopra Group et Everis
- ♦ Chef d'équipe chez System Dynamics
- ♦ Développeur de Logiciels chez SGO Software
- ♦ Master en E-Business de l'École de Commerce de La Salle
- ♦ Diplôme en Technologies et Systèmes d'Information, Institut Catalan de Technologie
- ♦ Licence en Génie Supérieur des Télécommunications de l'Université Polytechnique de Catalogne

M. Pintado San Claudio, Bruno

- ♦ Coordinateur du Développement chez IDavinci
- ♦ Développeur Java à la Bibliothèque Nationale d'Espagne
- ♦ Développeur de Support et Technicien Réseau N1 à Sanitas
- ♦ Technicien d'Appui Système à la Mairie d'Alcobendas
- ♦ Technicien de communication N1 pour ADIF au Centre de Télécommunications d'Atocha
- ♦ Diplôme en Ingénierie Technique des Télécommunications avec une Spécialisation en Systèmes
- ♦ Électroniques de l'Université Polytechnique de Valence
- ♦ Diplôme en Ingénierie Électronique des Communications à l'Université Polytechnique de Madrid

M. Reyes Oliva, Luis

- ♦ Architecte de *Logiciel* et *Cloud*
- ♦ Développeur et Architecte *Cloud* chez IBM
- ♦ Responsable Technique Client pour les comptes intégrés de BBVA chez IBM
- ♦ Architecte *Cloud* et *DevOps* chez IBM
- ♦ Architecte *Logiciel* client chez Telefónica
- ♦ Architecte de solutions techniques chez Rational
- ♦ *Software Engineering Manager* chez Borland
- ♦ *Project Manager* chez Altana Consulting
- ♦ Licence en Ingénierie Informatique de l'Université Pontificia de Salamanca

M. Calzada Martínez, Jesús

- ♦ Ingénieur de Software Senior en Devo
- ♦ Développeur Full *Full Stack* chez Blocknitive
- ♦ Responsable *Frontend* chez Infinia
- ♦ Développeur *Full Stack* chez Resem
- ♦ Développeur Java chez Hitec
- ♦ Diplôme en Ingénierie Informatique de l'Université Complutense de Madrid

08 Diplôme

Le Mastère Spécialisé en Programmation Front End Full Stack Developer garantit, outre la formation la plus rigoureuse et la plus actualisée, l'accès à un diplôme de Mastère Spécialisé délivré par TECH Global University.



“

*Terminez ce programme avec succès
et recevez votre diplôme sans avoir à
vous soucier des déplacements ou des
formalités administratives”*

Ce programme vous permettra d'obtenir votre diplôme propre de **Mastère Spécialisé en Programmation Front End Full Stack Developer** est le programme le plus complet sur la scène académique actuelle. Après avoir obtenu leur diplôme, les étudiants recevront un diplôme d'université délivré par TECH Global University et un autre par Université Euromed de Fès.

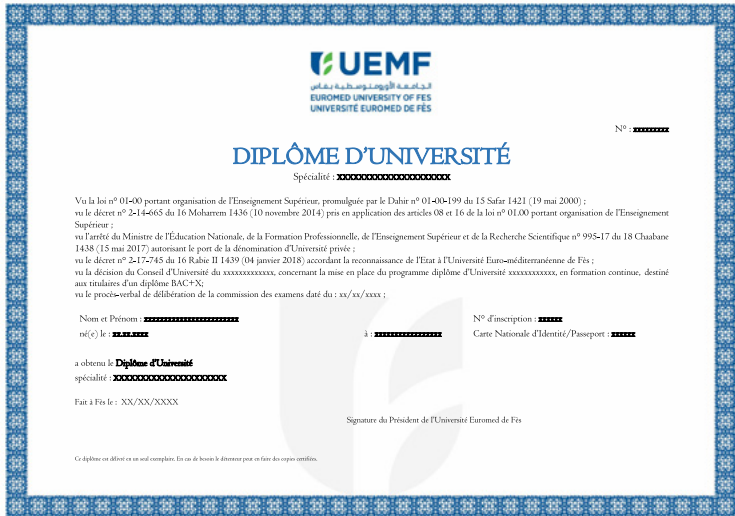
Ces diplômes de formation continue et d'actualisation professionnelle de TECH Global University et d'Université Euromed de Fès garantissent l'acquisition de compétences dans le domaine de la connaissance, en accordant une grande valeur curriculaire à l'étudiant qui réussit les évaluations et accrédite le programme après l'avoir suivi dans son intégralité. Ce double certificat, de la part de deux institutions universitaires de premier plan, représente une double récompense pour une formation complète et de qualité, assurant à l'étudiant l'obtention d'une certification reconnue au niveau national et international. Ce mérite académique vous positionnera comme un professionnel hautement qualifié, prêt à relever les défis et à répondre aux exigences de votre secteur professionnel.

Diplôme : **Mastère Spécialisé en Programmation Front End Full Stack Developer**

Modalité : **en ligne**

Durée : **12 mois**

Accréditation : **60 ECTS**



*Si l'étudiant souhaite que son diplôme version papier possède l'Apostille de La Haye, TECH Euromed University fera les démarches nécessaires pour son obtention moyennant un coût supplémentaire.



Mastère Spécialisé Programmation Front End Full Stack Developer

- » Modalité : en ligne
- » Durée : 12 mois
- » Diplôme : TECH Euromed University
- » Accréditation : 60 ECTS
- » Horaire : à votre rythme
- » Examens : en ligne

Mastère Spécialisé

Programmation Front End Full Stack Developer