

Certificat Avancé

Ingénierie Logicielle



Certificat Avancé Ingénierie Logicielle

- » Modalité: en ligne
- » Durée: 6 mois
- » Qualification: TECH Université Technologique
- » Intensité: 16h/semaine
- » Horaire: à votre rythme
- » Examens: en ligne

Accès au site web: www.techtitute.com/fr/informatique/diplome-universite/diplome-universite-ingenierie-logicielle

Accueil

01

Présentation

page 4

02

Objectifs

page 8

03

Structure et contenu

page 12

04

Méthodologie

page 22

05

Diplôme

page 30

01

Présentation

Le professionnel sera en mesure de mettre à jour ses connaissances en Ingénierie Logicielle avec les meilleurs professionnels de l'industrie. Grâce à cette formation de haut niveau, vous apprendrez les dernières tendances et nouveautés en la matière. Une occasion unique de donner à votre carrière l'élan dont elle a besoin.





Ce Certificat Avancé vous permettra d'actualiser vos connaissances en Ingénierie Logicielle de manière pratique et 100% en ligne, sans renoncer à la rigueur académique"

Ce programme s'adresse aux diplômés qui souhaitent atteindre un niveau supérieur de connaissances en Ingénierie Logicielle. L'objectif principal est de former les étudiants afin qu'ils puissent appliquer les connaissances acquises dans ce Certificat Avancé dans le monde réel, dans un environnement de travail qui reproduit les conditions qu'ils peuvent rencontrer dans leur futur, de manière rigoureuse et réaliste.

Ce Certificat Avancé préparera les étudiants à la pratique professionnelle de l'Ingénierie Informatique, grâce à une expérience académique transversale et polyvalente adaptée aux nouvelles technologies et aux innovations dans ce domaine. Vous acquerrez les connaissances approfondies en Ingénierie Logicielle auprès de professionnels du secteur.

Saisissez l'opportunité de suivre ce programme 100% en ligne, sans renoncer à vos obligations. Mettez à jour vos connaissances et obtenez votre qualification de Certificat Avancé pour continuer à vous développer personnellement et professionnellement.

Ce **Certificat Avancé en Ingénierie Logicielle** contient le programme académique le plus complet et le plus actuel du marché. Les principales caractéristiques sont les suivantes:

- ◆ Le développement de 100 cas simulés présentés par des experts en Ingénierie Logicielle
- ◆ Son contenu graphique, schématique et éminemment pratique fournit des informations scientifiques et pratiques en Ingénierie Logicielle
- ◆ Les récentes avancées concernant le Génie Logiciel
- ◆ Les exercices pratiques où effectuer le processus d'auto-évaluation pour améliorer l'apprentissage
- ◆ Un système d'apprentissage interactif basé sur la méthode des cas et son application dans la pratique quotidienne
- ◆ Tout cela sera complété par des cours théoriques, des questions à l'expert, des forums de discussion sur des sujets controversés et un travail de réflexion individuel
- ◆ La disponibilité du contenu à partir de tout appareil fixe ou portable doté d'une connexion à internet



Découvrez les dernières techniques et stratégies grâce à ce programme afin de réussir en tant qu'ingénieur informaticien"

“

Ce programme vous permettra d'améliorer vos compétences et mettre à jour vos connaissances en Ingénierie Logicielle"

Il comprend dans son corps enseignant une équipe de professionnels appartenant au domaine de l'Ingénierie Informatique, qui versent dans cette formation l'expérience de leur travail, ainsi que des spécialistes reconnus appartenant à des sociétés de référence et des universités prestigieuses.

Grâce à son contenu multimédia développé avec les dernières technologies éducatives les spécialistes bénéficieront d'un apprentissage situé et contextuel. Ainsi, ils se formeront dans un environnement simulé qui leur permettra d'apprendre en immersion et de s'entraîner dans des situations réelles.

Le design de ce programme est basée sur l'Apprentissage par les Problèmes. Ainsi l'étudiant devra essayer de résoudre les différentes situations de pratique professionnelle qui se présentent à lui tout au long du cursus académique. Pour ce faire, il sera assisté d'un système vidéo interactif innovant créé par des experts en Ingénierie Logicielle, ayant une grande expérience de l'enseignement.

Profitez des dernières technologies éducatives pour actualiser vos connaissances en Ingénierie Logicielle depuis votre domicile.

Découvrez les derniers développements en Ingénierie Logicielle, auprès d'experts dans le domaine.



02 Objectifs

L'objectif de cette formation est d'offrir aux professionnels les connaissances et les compétences nécessaires pour exercer leur activité en utilisant les protocoles et les techniques les plus avancés du moment. Par une démarche de travail totalement adaptable à l'étudiant, ce Certificat Avancé l'amènera progressivement à acquérir les compétences qui le propulseront vers un niveau professionnel supérieur.



“

Développez votre niveau de connaissances afin de maîtriser les concepts fondamentaux en Ingénierie Logicielle, grâce à ce programme de haut niveau”



Objectifs généraux

- ◆ Préparer les étudiants à la pratique professionnelle en Ingénierie Informatique, grâce à une formation transversale, polyvalente et adaptée aux nouvelles technologies et innovations dans ce domaine
- ◆ Obtenir une large connaissance dans le domaine de l'informatique, la structure des ordinateurs et le Génie Logiciel, incluant les bases mathématiques, statistiques et physiques essentielles à l'ingénierie

“

Réussissez professionnellement en tant qu'informaticien grâce à ce programme intensif, élaboré par des professionnels ayant une grande expérience du secteur”





Objectifs spécifiques

- ◆ Poser les bases du génie Logiciel et de la modélisation, en apprenant les principaux processus et concepts
- ◆ Comprendre le processus Logiciel et les différents modèles pour son développement, y compris les technologies agiles
- ◆ Comprendre l'ingénierie des exigences, leur développement, leur élaboration, leur négociation et leur validation
- ◆ Apprenez la modélisation des exigences et les différents éléments tels que les scénarios, les informations, les classes d'analyse, le flux, le comportement et les modèles
- ◆ Comprendre les concepts et les processus de conception logicielle, en apprenant également sur la conception de l'architecture et sur la conception au niveau des composants et basée sur des modèles
- ◆ Connaître les principales normes relatives à la qualité des Logiciels et à la gestion de projet
- ◆ Découvrez en profondeur les différentes méthodologies agiles utilisées dans l'ingénierie logicielle
- ◆ Apprendre à développer en utilisant les techniques de *Scrum*, de programmation extrême et de développement logiciel basé sur la réutilisation
- ◆ Comprendre les différents modèles d'architecture de système et de conception logicielle, ainsi que l'architecture d'applications cloud
- ◆ Apprendre à réaliser des tests de logiciels, avec des méthodologies telles que *Test Driven Development*, *Acceptance Test Driven Development*, *Behavior Driven Development*, *BDD* y *Cucumber*
- ◆ Approfondir l'amélioration du processus de développement de logiciels et de la qualité des logiciels en utilisant les normes ISO/IEC
- ◆ Introduire le concept de *DevOps* et quelles sont ses principales pratiques
- ◆ Comprendre l'importance de l'ingénierie des besoins dans le processus de développement de logiciels
- ◆ Approfondir les sources d'exigences et les techniques d'appel d'offres, car elles sont un élément essentiel du processus
- ◆ Comprendre et appliquer le prototypage comme partie essentielle du processus de développement
- ◆ Apprendre à réaliser des analyses et à documenter correctement les exigences
- ◆ Comprendre les processus de validation et de négociation des exigences, ainsi que la modélisation et la gestion des exigences
- ◆ Acquérir les connaissances nécessaires à la gestion des systèmes critiques et à la spécification formelle des exigences
- ◆ Connaître le cadre de référence de l'ingénierie logicielle et la norme ISO/IEC 12207
- ◆ Apprendre les caractéristiques du processus unifié de développement de logiciels et de planification dans le contexte du développement de logiciels agiles
- ◆ Connaître les différents styles de conception de logiciels distribués et les architectures logicielles orientées services
- ◆ Apprendre les concepts essentiels dans la conception d'interfaces graphiques utilisateur
- ◆ Comprendre les bases du développement d'applications web
- ◆ Approfondir les stratégies et techniques de test des logiciels, les facteurs de qualité des logiciels et les différentes mesures utilisées

03

Structure et contenu

La structure des contenus a été conçue par une équipe des professionnels en Ingénierie Informatique, conscients de la pertinence de l'actualité de la formation, dans le but d'enrichir les connaissances des étudiants et d'élever leur niveau en Ingénierie Logicielle, avec les dernières technologies éducatives disponibles.



“

Le Certificat Avancé en Ingénierie Logicielle contient le programme d'apprentissage le plus complet et le plus actuel du marché”

Module 1. Ingénierie Logicielle

- 1.1. Introduction au génie logiciel et à la modélisation
 - 1.1.1. La nature des logiciels
 - 1.1.2. La nature unique des *WebApps*
 - 1.1.3. Ingénierie logicielle
 - 1.1.4. Le processus logiciel
 - 1.1.5. La pratique de l'Ingénierie Logicielle
 - 1.1.6. Mythes sur les logiciels
 - 1.1.7. Comment tout commence
 - 1.1.8. Concepts orientés objet
 - 1.1.9. Introduction à UML
- 1.2. Le processus logiciel
 - 1.2.1. Un modèle général de processus
 - 1.2.2. Modèles de processus prescriptifs
 - 1.2.3. Modèles de processus spécialisés
 - 1.2.4. Le processus unifié
 - 1.2.5. Modèles de processus personnels et d'équipe
 - 1.2.6. Qu'est-ce que l'agilité?
 - 1.2.7. Qu'est-ce qu'un processus agile?
 - 1.2.8. *Scrum*
 - 1.2.9. Boîte à outils du processus Agile
- 1.3. Principes guidant la pratique de l'Ingénierie Logicielle
 - 1.3.1. Principes guidant le processus
 - 1.3.2. Principes guidant la pratique
 - 1.3.3. Principes de la communication
 - 1.3.4. Principes de planification
 - 1.3.5. Principes de modélisation
 - 1.3.6. Principes de construction
 - 1.3.7. Principes de déploiement
- 1.4. Comprendre les besoins
 - 1.4.1. Ingénierie des exigences
 - 1.4.2. Établir la base
 - 1.4.3. Détermination des besoins
 - 1.4.4. Développer des cas d'utilisation
 - 1.4.5. Élaboration du modèle d'exigences
 - 1.4.6. Négociation des exigences
 - 1.4.7. Validation des exigences
- 1.5. Modélisation des exigences: scénarios, classes d'information et d'analyse
 - 1.5.1. Analyse des besoins
 - 1.5.2. Modélisation basée sur des scénarios
 - 1.5.3. Modèles UML fournissant le cas d'utilisation
 - 1.5.4. Concepts de modélisation des données
 - 1.5.5. Modélisation basée sur les classes
 - 1.5.6. Diagrammes de classes
- 1.6. Modélisation des exigences: flux, comportement et modèles
 - 1.6.1. Stratégies de modélisation des exigences
 - 1.6.2. Modélisation orientée flux
 - 1.6.3. Diagrammes d'état
 - 1.6.4. Création d'un modèle comportemental
 - 1.6.5. Diagrammes de séquence
 - 1.6.6. Diagrammes de communication
 - 1.6.7. Modèles pour la modélisation des exigences
- 1.7. Concepts de design
 - 1.7.1. Conception dans le contexte de l'ingénierie logicielle
 - 1.7.2. Le processus de conception
 - 1.7.3. Concepts de design
 - 1.7.4. Concepts de conception orientée objet
 - 1.7.5. Le modèle de conception

- 1.8. Architecture de conception
 - 1.8.1. Architecture logicielle
 - 1.8.2. Genres architecturaux
 - 1.8.3. Styles architecturaux
 - 1.8.4. Conception architecturale
 - 1.8.5. Évolution des conceptions alternatives de l'architecture
 - 1.8.6. Cartographie de l'architecture à l'aide du flux de données
- 1.9. Conception au niveau des composants et des modèles
 - 1.9.1. Qu'est-ce qu'un composant?
 - 1.9.2. Conception de composants basée sur les classes
 - 1.9.3. Réalisation de la conception au niveau des composants
 - 1.9.4. Conception traditionnelle des composants
 - 1.9.5. Développement basé sur les composants
 - 1.9.6. Modèles de conception
 - 1.9.7. Conception de logiciels basée sur des modèles
 - 1.9.8. Modèles architecturaux
 - 1.9.9. Patrons de conception au niveau des composants
 - 1.9.10. Modèles de conception d'interface utilisateur
- 1.10. Qualité des logiciels et gestion de projet
 - 1.10.1. Qualité
 - 1.10.2. Qualité des logiciels
 - 1.10.3. Le dilemme de la qualité des logiciels
 - 1.10.4. Atteindre la qualité des logiciels
 - 1.10.5. Assurance qualité des logiciels
 - 1.10.6. Le spectre de la gestion
 - 1.10.7. Personnel
 - 1.10.8. Le produit
 - 1.10.9. Le processus
 - 1.10.10. Le projet
 - 1.10.11. Principes et pratiques

Module 2. Ingénierie Software Avancée

- 2.1. Introduction aux méthodologies agiles
 - 2.1.1. Modèles de processus et méthodologies
 - 2.1.2. Agilité et processus agiles
 - 2.1.3. Manifeste agile
 - 2.1.4. Quelques méthodologies agiles
 - 2.1.5. Agile vs. Traditionnel
- 2.2. *Scrum*
 - 2.2.1. Origine et philosophie de *Scrum*
 - 2.2.2. Valeurs de *Scrum*
 - 2.2.3. Déroulement du processus *Scrum*
 - 2.2.4. Les rôles de *Scrum*
 - 2.2.5. Les artéfacts *Scrum*
 - 2.2.6. Les événements *Scrum*
 - 2.2.7. Histoires d'utilisateur
 - 2.2.8. Les extensions de *Scrum*
 - 2.2.9. Estimations rapides
 - 2.2.10. Mise à l'échelle *Scrum*
- 2.3. Programmation extrême
 - 2.3.1. Justification et présentation de XP
 - 2.3.2. Le cycle de vie au XP
 - 2.3.3. Les cinq valeurs de base
 - 2.3.4. Les douze pratiques de base de XP
 - 2.3.5. Rôles des participants
 - 2.3.6. XP Industriel
 - 2.3.7. Évaluation critique de XP

- 2.4. Développement de logiciels basés sur la réutilisation
 - 2.4.1. Réutilisation de software
 - 2.4.2. Niveaux de réutilisation du code
 - 2.4.3. Techniques spécifiques de réutilisation
 - 2.4.4. Développement basé sur les composants
 - 2.4.5. Avantages et problèmes de réutilisation
 - 2.4.6. Planification de la réutilisation
- 2.5. Modèles d'architecture système et de conception logicielle
 - 2.5.1. Conception architecturale
 - 2.5.2. Modèles architecturaux généraux
 - 2.5.3. Architectures tolérantes aux pannes
 - 2.5.4. Architectures de systèmes distribués
 - 2.5.5. Modèles de conception
 - 2.5.6. Modèles Gamma
 - 2.5.7. Modèles de conception d'interaction
- 2.6. Architecture d'application cloud
 - 2.6.1. Principes fondamentaux du *Cloud Computing*
 - 2.6.2. Qualité d'application cloud
 - 2.6.3. Styles d'architecture
 - 2.6.4. Modèles de conception
- 2.7. Tests logiciels: TDD, ATDD et BDD
 - 2.7.1. Vérification et validation du logiciel
 - 2.7.2. Tests logiciels
 - 2.7.3. *Tests Driven Development* (TDD)
 - 2.7.4. *Acceptance Tests Driven Development* (ATDD)
 - 2.7.5. *Behavior Driven Development* (BDD)
 - 2.7.6. BDD et *Cucumber*
- 2.8. L'amélioration du processus logiciel
 - 2.8.1. L'amélioration du processus logiciel
 - 2.8.2. Le processus d'amélioration de processus
 - 2.8.3. Modèles de maturité
 - 2.8.4. Le modèle CMMI
 - 2.8.5. CMMI V2.0
 - 2.8.6. CMMI et Agile

```
63     echo "Photo gallery";
64     elseif (($_COOKIE['lang'] == 'rus')) {
65         echo "Фотогалерея";
66     }
67     else
68         echo "Foto galerija";
69     ?></h3>-->
70
71 <div class="<?if($_GET[type]==1)"
72 <a href="foto-galerija.php?t
73     <div id="left_sidebar">
74         <div id="left_ico">
75             <p <?if($_COOKIE['lang']
76 <?
77 if($_COOKIE['lang'] == 'eng') {
78     echo "Wood-frame houses";
79 }elseif($_COOKIE['lang'] == 'rus') {
80     echo "Деревянные каркасные
81 }else{
82     echo "Koka karkasa mājas";
```


- 2.9. Qualité des systèmes informatiques: Square
 - 2.9.1. Qualité des logiciels
 - 2.9.2. Modèles de qualité des produit logiciels
 - 2.9.3. Famille ISO/IEC 25000
 - 2.9.4. ISO/IEC 25010: modèle et caractéristiques de qualité
 - 2.9.5. ISO/IEC 25012: qualité des données
 - 2.9.6. ISO/IEC 25020: mesure de la qualité du logiciel
 - 2.9.7. ISO/IEC 25022, 25023 et 25024: mesures de qualité des logiciels et des données
 - 2.9.8. ISO/IEC 25040: évaluation du logiciel
 - 2.9.9. Processus de certification
- 2.10. Présentation de *DevOps*
 - 2.10.1. Concept *DevOps*
 - 2.10.2. Principales pratiques

Module 3. Ingénierie des exigences

- 3.1. Introduction à l'Ingénierie des Exigences
 - 3.1.1. L'importance des exigences
 - 3.1.2. Concept d'exigence
 - 3.1.3. Dimensions des exigences
 - 3.1.4. Niveaux et types d'exigences
 - 3.1.5. Caractéristiques des exigences
 - 3.1.6. Ingénierie des exigences
 - 3.1.7. Le processus d'Ingénierie des Exigences
 - 3.1.8. *Frameworks* pour l'ingénierie des exigences
 - 3.1.9. Meilleures pratiques en matière d'ingénierie des exigences
 - 3.1.10. L'analyste d'affaires
- 3.2. Les sources des besoins
 - 3.2.1. Le réseau des exigences
 - 3.2.2. Les *Stakeholders*
 - 3.2.3. Les exigences commerciales
 - 3.2.4. Document sur la vision et le champ d'application

- 3.3. Techniques d'élicitation des besoins
 - 3.3.1. Détermination des besoins
 - 3.3.2. Problèmes liés à la détermination des besoins
 - 3.3.3. Contextes de découverte
 - 3.3.4. Interviews
 - 3.3.5. Observation et "apprentissage"
 - 3.3.6. Ethnographie
 - 3.3.7. *Workshops*
 - 3.3.8. *Focus Groups*
 - 3.3.9. Questionnaires
 - 3.3.10. *Brainstorming* et techniques de création
 - 3.3.11. Médias du groupe
 - 3.3.12. Analyse des interfaces du système
 - 3.3.13. Analyse de documents et "archéologie"
 - 3.3.14. Cas d'utilisation et scénario
 - 3.3.15. Prototypes
 - 3.3.16. Ingénierie inverse
 - 3.3.17. Réutilisation des exigences
 - 3.3.18. Bonnes pratiques en matière d'appels d'offres
- 3.4. Besoins des utilisateurs
 - 3.4.1. Personnes
 - 3.4.2. Cas d'utilisation et histoires d'utilisateurs
 - 3.4.3. Scénario
 - 3.4.4. Types de scénarios
 - 3.4.5. Comment découvrir des scénarios
- 3.5. Techniques de prototypage
 - 3.5.1. Prototypage
 - 3.5.2. Prototypes en fonction de leur portée
 - 3.5.3. Prototypes en fonction de leur temporalité
 - 3.5.4. La fidélité d'un prototype
 - 3.5.5. Prototypes d'interface utilisateur
 - 3.5.6. Évaluation des prototypes
- 3.6. Analyse des besoins
 - 3.6.1. Analyse des besoins
 - 3.6.2. Bonnes pratiques en matière d'analyse des exigences
 - 3.6.3. Dictionnaire de données
 - 3.6.4. Hiérarchisation des besoins
- 3.7. Documentation des exigences
 - 3.7.1. Le document de spécification des exigences
 - 3.7.2. Structure et contenu d'un BSS
 - 3.7.3. Documentation en langage naturel
 - 3.7.4. EARS: *Easy Approach to Requirements Syntax*
 - 3.7.5. Exigences non fonctionnelles
 - 3.7.6. Attributs et modèles de table
 - 3.7.7. Bonne pratique de spécification
- 3.8. Validation et négociation des exigences
 - 3.8.1. Validation des exigences
 - 3.8.2. Techniques de validation des exigences
 - 3.8.3. Négociation des exigences
- 3.9. Modélisation et gestion des exigences
 - 3.9.1. Modélisation des besoins
 - 3.9.2. Le point de vue de l'utilisateur
 - 3.9.3. Perspective des données
 - 3.9.4. La perspective fonctionnelle ou axée sur le flux
 - 3.9.5. La perspective comportementale
 - 3.9.6. Volatilité des exigences
 - 3.9.7. Processus de gestion des exigences
 - 3.9.8. Outils de gestion des exigences
 - 3.9.9. Bonnes pratiques en matière de gestion des exigences
- 3.10. Systèmes critiques et spécification formelle
 - 3.10.1. Systèmes critiques
 - 3.10.2. Spécification axée sur le risque
 - 3.10.3. Spécification formelle

Module 4. Processus d'Ingénierie Logicielle

- 4.1. Cadre de l'Ingénierie logicielle
 - 4.1.1. Caractéristiques des logiciels
 - 4.1.2. Principaux processus du génie logiciel
 - 4.1.3. Modèles de processus de développement de logiciels
 - 4.1.4. Cadre de référence normalisé pour le processus de développement logiciel: la norme ISO/CEI 12207
- 4.2. Processus unifiés de développement logiciels
 - 4.2.1. Processus unifié
 - 4.2.2. Dimensions du processus unifié
 - 4.2.3. Processus de développement basé sur les cas d'utilisation
 - 4.2.4. Flux de travail fondamentaux du processus unifié
- 4.3. La planification dans le contexte du développement logiciel agile
 - 4.3.1. Caractéristiques du développement logiciel agile
 - 4.3.2. Différents horizons de planification dans le développement agile
 - 4.3.3. Cadre de développement agile *Scrum* et des horizons temporels de planification
 - 4.3.4. User stories comme unité de planification et d'estimation
 - 4.3.5. Techniques courantes de calcul d'une estimation
 - 4.3.6. Échelles d'interprétation des estimations
 - 4.3.7. *Planning Poker*
 - 4.3.8. Types courants de planification: planification des livraisons et planification des itérations
- 4.4. Styles de conception de logiciels distribués et architectures logicielles orientées services
 - 4.4.1. Modèles de communication dans les systèmes logiciels distribués
 - 4.4.2. Couche intermédiaire ou *Middleware*
 - 4.4.3. Modèles architecturaux pour les systèmes distribués
 - 4.4.4. Processus général de conception de services logiciels
 - 4.4.5. Aspects de la conception de services logiciels
 - 4.4.6. Composition des services
 - 4.4.7. Architecture des services web
 - 4.4.8. Composants d'infrastructure et SOA

- 4.5. Introduction au développement logiciel guidé par le modèle
 - 4.5.1. Le concept de modèle
 - 4.5.2. Développement logiciel guidé par le modèle
 - 4.5.3. Cadre de référence pour le développement piloté par des modèles MDA
 - 4.5.4. Éléments d'un modèle de transformation
- 4.6. Conception d'interfaces utilisateur graphiques
 - 4.6.1. Principes de conception d'interfaces utilisateur
 - 4.6.2. Modèles de conception architecturale pour les systèmes interactifs: Modèle Vue Contrôleur Vue (MVC)
 - 4.6.3. Expérience de l'utilisateur (UX *User Experience*)
 - 4.6.4. Conception centrée sur l'utilisateur
 - 4.6.5. Processus d'analyse et de conception d'une interface utilisateur graphique
 - 4.6.6. Utilisabilité d'interfaces utilisateurs
 - 4.6.7. Accessibilité des interfaces utilisateurs
- 4.7. Conception d'applications web
 - 4.7.1. Caractéristiques des applications web
 - 4.7.2. Interface utilisateur d'une application web
 - 4.7.3. Conception de la navigation
 - 4.7.4. Protocole d'interaction de base pour les applications web
 - 4.7.5. Styles d'architecture pour les applications web
- 4.8. Stratégies et techniques de test des logiciels et facteurs de qualité des logiciels
 - 4.8.1. Stratégies d'essai
 - 4.8.2. Designs de cas d'essai
 - 4.8.3. Rapport qualité-prix
 - 4.8.4. Modèles de qualité
 - 4.8.5. Famille de normes ISO/IEC 25000 (Square)
 - 4.8.6. Modèle de qualité du produit (ISO 2501n)
 - 4.8.7. Modèles de qualité des données (ISO 2501n)
 - 4.8.8. Gestion de la qualité des logiciels





- 4.9. Introduction aux métriques de l'ingénierie logicielle
 - 4.9.1. Concepts de base: mesures, métriques et indicateurs
 - 4.9.2. Types de métriques en Ingénierie Logicielle
 - 4.9.3. Le processus de mesure
 - 4.9.4. ISO 25024. Mesures externes et de qualité utilisées
 - 4.9.5. Métriques orientées objet
- 4.10. Maintenance et réingénierie des logiciels
 - 4.10.1. Processus de maintenance
 - 4.10.2. Cadre du processus de maintenance standard. ISO/EIEC 14764
 - 4.10.3. Modèle de processus de réingénierie des logiciels
 - 4.10.4. Ingénierie inverse

“

*Une expérience de formation
unique, clé et décisive pour stimuler
votre développement professionnel”*

04 Méthodologie

Ce programme de formation offre une manière différente d'apprendre. Notre méthodologie est développée à travers un mode d'apprentissage cyclique: ***le Relearning***.

Ce système d'enseignement est utilisé, par exemple, dans les écoles de médecine les plus prestigieuses du monde et a été considéré comme l'un des plus efficaces par des publications de premier plan telles que le ***New England Journal of Medicine***.



“

Découvrez Relearning, un système qui renonce à l'apprentissage linéaire conventionnel pour vous emmener à travers des systèmes d'enseignement cycliques: une façon d'apprendre qui s'est avérée extrêmement efficace, en particulier dans les matières qui exigent la mémorisation”

Étude de Cas pour mettre en contexte tout le contenu

Notre programme offre une méthode révolutionnaire de développement des compétences et des connaissances. Notre objectif est de renforcer les compétences dans un contexte changeant, compétitif et hautement exigeant.

“

Avec TECH, vous pouvez expérimenter une manière d'apprendre qui ébranle les fondations des universités traditionnelles du monde entier”



Vous bénéficierez d'un système d'apprentissage basé sur la répétition, avec un enseignement naturel et progressif sur l'ensemble du cursus.



L'étudiant apprendra, par des activités collaboratives et des cas réels, à résoudre des situations complexes dans des environnements commerciaux réels.

Une méthode d'apprentissage innovante et différente

Cette formation TECH est un programme d'enseignement intensif, créé de toutes pièces, qui propose les défis et les décisions les plus exigeants dans ce domaine, tant au niveau national qu'international. Grâce à cette méthodologie, l'épanouissement personnel et professionnel est stimulé, faisant ainsi un pas décisif vers la réussite. La méthode des cas, technique qui constitue la base de ce contenu, permet de suivre la réalité économique, sociale et professionnelle la plus actuelle.

“ Notre programme vous prépare à relever de nouveaux défis dans des environnements incertains et à réussir votre carrière ”

La méthode des cas est le système d'apprentissage le plus largement utilisé dans les meilleures écoles d'informatique du monde depuis qu'elles existent. Développée en 1912 pour que les étudiants en Droit n'apprennent pas seulement le droit sur la base d'un contenu théorique, la méthode des cas consiste à leur présenter des situations réelles complexes afin qu'ils prennent des décisions éclairées et des jugements de valeur sur la manière de les résoudre. En 1924, elle a été établie comme méthode d'enseignement standard à Harvard.

Dans une situation donnée, que doit faire un professionnel? C'est la question à laquelle nous sommes confrontés dans la méthode des cas, une méthode d'apprentissage orientée vers l'action. Tout au long du programme, les étudiants seront confrontés à de multiples cas réels. Ils devront intégrer toutes leurs connaissances, faire des recherches, argumenter et défendre leurs idées et leurs décisions.

Relearning Methodology

TECH combine efficacement la méthodologie des Études de Cas avec un système d'apprentissage 100% en ligne basé sur la répétition, qui associe différents éléments didactiques dans chaque leçon.

Nous enrichissons l'Étude de Cas avec la meilleure méthode d'enseignement 100% en ligne: le Relearning.

En 2019, nous avons obtenu les meilleurs résultats d'apprentissage de toutes les universités en ligne du monde.

À TECH, vous apprendrez avec une méthodologie de pointe conçue pour former les managers du futur. Cette méthode, à la pointe de la pédagogie mondiale, est appelée Relearning.

Notre université est la seule université autorisée à utiliser cette méthode qui a fait ses preuves. En 2019, nous avons réussi à améliorer les niveaux de satisfaction globale de nos étudiants (qualité de l'enseignement, qualité des supports, structure des cours, objectifs...) par rapport aux indicateurs de la meilleure université en ligne.



Dans notre programme, l'apprentissage n'est pas un processus linéaire, mais se déroule en spirale (apprendre, désapprendre, oublier et réapprendre). Par conséquent, chacun de ces éléments est combiné de manière concentrique. Cette méthodologie a permis de former plus de 650.000 diplômés universitaires avec un succès sans précédent dans des domaines aussi divers que la biochimie, la génétique, la chirurgie, le droit international, les compétences en gestion, les sciences du sport, la philosophie, le droit, l'ingénierie, le journalisme, l'histoire, les marchés financiers et les instruments. Tout cela dans un environnement très exigeant, avec un corps étudiant universitaire au profil socio-économique élevé et dont l'âge moyen est de 43,5 ans.

Le Relearning vous permettra d'apprendre avec moins d'efforts et plus de performance, en vous impliquant davantage dans votre formation, en développant un esprit critique, en défendant des arguments et en contrastant les opinions: une équation directe vers le succès.

À partir des dernières preuves scientifiques dans le domaine des neurosciences, non seulement nous savons comment organiser les informations, les idées, les images et les souvenirs, mais nous savons aussi que le lieu et le contexte dans lesquels nous avons appris quelque chose sont fondamentaux pour notre capacité à nous en souvenir et à le stocker dans l'hippocampe, pour le conserver dans notre mémoire à long terme.

De cette manière, et dans ce que l'on appelle Neurocognitive context-dependent e-learning, les différents éléments de notre programme sont reliés au contexte dans lequel le participant développe sa pratique professionnelle.



Ce programme offre le support matériel pédagogique, soigneusement préparé pour les professionnels:



Support d'étude

Tous les contenus didactiques sont créés par les spécialistes qui enseigneront le cours, spécifiquement pour le cours, afin que le développement didactique soit vraiment spécifique et concret.

Ces contenus sont ensuite appliqués au format audiovisuel, pour créer la méthode de travail TECH en ligne. Tout cela, avec les dernières techniques qui offrent des pièces de haute qualité dans chacun des matériaux qui sont mis à la disposition de l'étudiant.



Cours magistraux

Il existe des preuves scientifiques de l'utilité de l'observation par un tiers expert.

La méthode "Learning from an Expert" renforce les connaissances et la mémoire, et donne confiance dans les futures décisions difficiles.



Pratiques en compétences et aptitudes

Les étudiants réaliseront des activités visant à développer des compétences et des aptitudes spécifiques dans chaque domaine. Des activités pratiques et dynamiques pour acquérir et développer les compétences et aptitudes qu'un spécialiste doit développer dans le cadre de la mondialisation dans laquelle nous vivons.



Lectures complémentaires

Articles récents, documents de consensus et directives internationales, entre autres. Dans la bibliothèque virtuelle de TECH, l'étudiant aura accès à tout ce dont il a besoin pour compléter sa formation.





Case studies

Ils réaliseront une sélection des meilleures études de cas choisies spécifiquement pour ce diplôme. Des cas présentés, analysés et tutorés par les meilleurs spécialistes de la scène internationale.



Résumés interactifs

L'équipe TECH présente les contenus de manière attrayante et dynamique dans des pilules multimédia comprenant des audios, des vidéos, des images, des diagrammes et des cartes conceptuelles afin de renforcer les connaissances. Ce système éducatif unique pour la présentation de contenu multimédia a été récompensé par Microsoft en tant que "European Success Story".



Testing & Retesting

Les connaissances de l'étudiant sont périodiquement évaluées et réévaluées tout au long du programme, par le biais d'activités et d'exercices d'évaluation et d'auto-évaluation, afin que l'étudiant puisse vérifier comment il atteint ses objectifs.



05 Diplôme

Le Certificat Avancé en Ingénierie Logicielle vous garantit, en plus de la formation la plus rigoureuse et la plus actuelle, l'accès à un diplôme universitaire de Certificat Avancé délivré par TECH Université Technologique.



“

Terminez ce programme avec succès et obtenez votre diplôme universitaire sans avoir à vous déplacer ou à remplir des formalités administratives”

Ce **Certificat Avancé en Ingénierie Logicielle** contient le programme le plus complet et le plus à jour du marché.

Après avoir réussi l'évaluation, l'étudiant recevra par courrier postal* avec accusé de réception son correspondant diplôme de **Certificat Avancé** délivré par **TECH Université Technologique**.

Le diplôme délivré par **TECH Université Technologique** indiquera la note obtenue lors du Certificat Avancé, et répond aux exigences communément demandées par les bourses d'emploi, les concours et les commissions d'évaluation des carrières professionnelles.

Diplôme: **Certificat Avancé en Ingénierie Logicielle**

N.º d'heures officielles: **600 h.**



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

future
santé confiance personnes
éducation information tuteurs
garantie accréditation enseignement
institutions technologie apprentissage
communauté engagement
service personnalisé innovation
connaissance présent qualité
en ligne formation
développement institutions
classe virtuelle langues

tech université
technologique

Certificat Avancé Ingénierie Logicielle

- » Modalité: en ligne
- » Durée: 6 mois
- » Qualification: TECH Université Technologique
- » Intensité: 16h/semaine
- » Horaire: à votre rythme
- » Examens: en ligne

Certificat Avancé Ingénierie Logicielle

