

Esperto Universitario Ingegneria del Software



Esperto Universitario Ingegneria del Software

- » Modalità: online
- » Durata: 6 mesi
- » Titolo: **TECH** Università
Tecnologica
- » Orario: a scelta
- » Esami: online

Accesso al sito web: www.techitute.com/it/informatica/specializzazione/specializzazione-ingegneria-software

Indice

01

Presentazione

pag. 4

02

Obiettivi

pag. 8

03

Struttura e contenuti

pag. 12

04

Metodologia

pag. 22

05

Titolo

pag. 30

01

Presentazione

Il professionista potrà aggiornare le proprie conoscenze in materia di Ingegneria del Software con i migliori esperti del settore. Grazie a questa preparazione di alto livello, apprenderai le ultime tendenze e gli sviluppi del settore. Un'opportunità unica per dare alla propria carriera la spinta necessaria.





Questo Esperto Universitario ti permetterà di aggiornare le tue conoscenze in materia di Ingegneria del Software in modo pratico, 100% online, senza rinunciare al massimo rigore accademico"

Questo programma è rivolto a coloro che sono interessati a raggiungere un livello di conoscenza superiore nell'ambito dell'Ingegneria del Software. L'obiettivo principale è quello di consentire agli studenti di applicare le conoscenze acquisite in questo Esperto Universitario nel mondo reale, in un ambiente di lavoro che riproduce le condizioni che potrebbero incontrare nel loro futuro, in modo rigoroso e realistico.

Questo Esperto Universitario preparerà gli studenti alla pratica professionale dell'Ingegneria Informatica, grazie a una esperienza accademica trasversale e versatile, adeguata alle nuove tecnologie e alle innovazioni del settore. Si acquisiranno conoscenze approfondite in Ingegneria del Software da parte di professionisti del settore.

Lo studente potrà cogliere l'opportunità di seguire questa preparazione in un formato 100% online, senza dover rinunciare ai propri impegni. Si tratta di un'occasione per aggiornare le tue conoscenze, per ottenere la qualifica di Esperto Universitario e per continuare a crescere personalmente e professionalmente.

Questo **Esperto Universitario in Ingegneria del Software** possiede il programma più completo e aggiornato del mercato. Le caratteristiche principali del programma sono:

- ◆ Sviluppo di 100 scenari simulati presentati da esperti di Ingegneria del Software
- ◆ Contenuti grafici, schematici ed eminentemente pratici forniscono informazioni scientifiche e pratiche sull'Ingegneria del Software
- ◆ Notizie sugli ultimi progressi nel campo dell'Ingegneria del Software
- ◆ Esercizi pratici che offrono un processo di autovalutazione per migliorare l'apprendimento
- ◆ Sistema di apprendimento interattivo basato sul metodo dei casi e sulla loro applicazione alla pratica reale
- ◆ Lezioni teoriche, domande all'esperto, forum di discussione su questioni controverse e compiti di riflessione individuale
- ◆ Disponibilità di contenuti da qualsiasi dispositivo fisso o mobile con connessione a internet



Impara le tecniche e le strategie più recenti con questo programma e avrai successo come ingegnere informatico"

“

Questo programma ti permetterà di migliorare le tue competenze e di aggiornare le tue conoscenze in Ingegneria del Software”

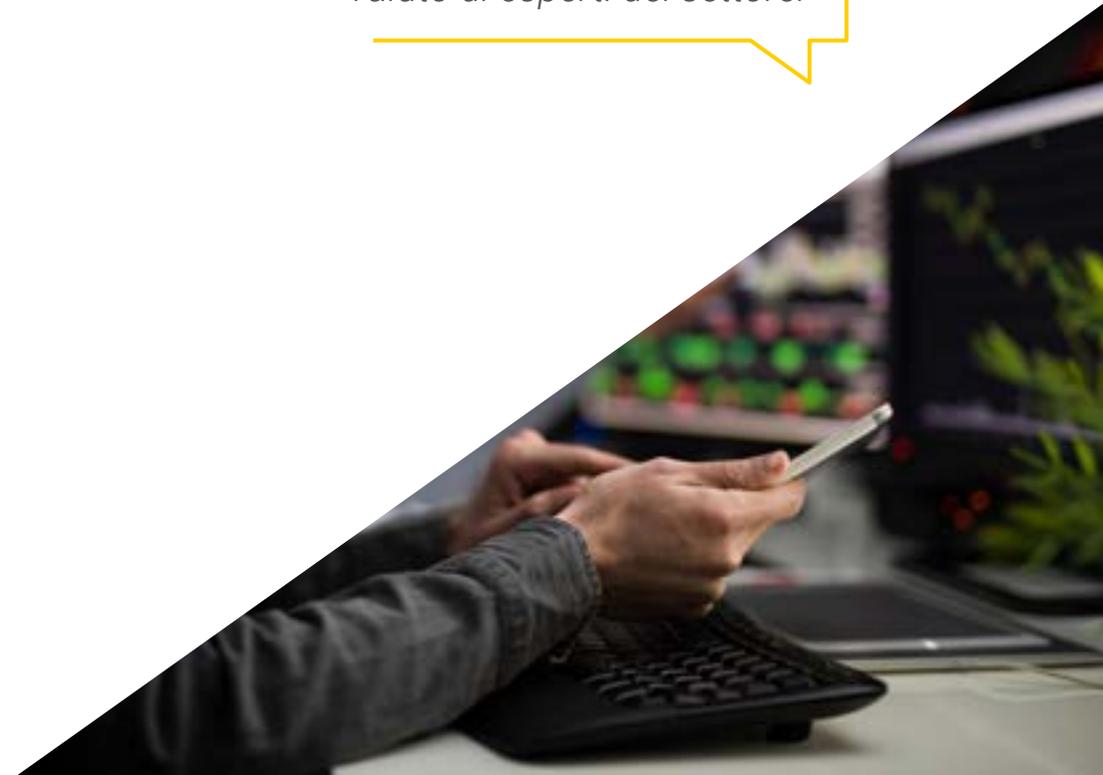
Il personale docente del programma comprende rinomati specialisti del settore e altre aree correlate, che forniscono agli studenti le competenze necessarie a intraprendere un percorso di studio eccellente.

I contenuti multimediali sviluppati in base alle ultime tecnologie educative forniranno al professionista un apprendimento coinvolgente e localizzato, ovvero inserito in un contesto reale.

La creazione di questo programma è incentrata sull'Apprendimento Basato su Problemi, mediante il quale il docente deve cercare di risolvere le diverse situazioni che gli si presentano durante il corso accademico. Lo studente potrà usufruire di un innovativo sistema di video interattivi creati da esperti di rinomata fama in Ingegneria del Software.

Approfitta delle ultime tecnologie didattiche per aggiornarti in Ingegneria del Software senza uscire di casa.

Scopri le ultime tecniche in Ingegneria del Software con l'aiuto di esperti del settore.



02

Obiettivi

L'obiettivo di questa preparazione è offrire ai professionisti di Informatica le conoscenze e le abilità necessarie per realizzare la loro attività seguendo i protocolli e le tecniche più avanzate del momento. Attraverso un approccio lavorativo completamente adattabile allo studente, questo Esperto Universitario porterà progressivamente ad acquisire le competenze che lo proietteranno a un livello professionale superiore.



“

Raggiungi il livello di conoscenza che desideri e padroneggia i concetti fondamentali dell'Ingegneria del Software, con questa preparazione di alto livello"



Obiettivi generali

- ◆ Preparare scientificamente e tecnologicamente, nonché ad esercitare la professione di ingegnere Informatico, con una conoscenza trasversale e versatile, adeguata alle nuove tecnologie e alle innovazioni del settore
- ◆ Ottenere una conoscenza approfondita nel campo del calcolo, della struttura del computer e dell'Ingegneria del Software, comprese le basi matematiche, statistiche e fisiche essenziali nell'ingegneria



Raggiungi il successo professionale come ingegnere informatico con questo programma intensivo, sviluppato da professionisti con una vasta esperienza nel settore"





Obiettivi specifici

- ◆ Porre le basi dell'ingegneria del software e della modellazione, apprendendo i principali processi e concetti
- ◆ Comprendere il processo del software e i diversi modelli di sviluppo del software, comprese le tecnologie agili
- ◆ Comprendere l'ingegneria dei requisiti, il loro sviluppo, l'elaborazione, la negoziazione e la convalida
- ◆ Imparare la modellazione dei requisiti e i diversi elementi come scenari, informazioni, classi di analisi, flussi, comportamenti e modelli
- ◆ Comprendere i concetti e i processi di progettazione del software, apprendendo anche la progettazione dell'architettura, quella a livello di componenti e quella basata su pattern
- ◆ Conoscere i principali standard relativi alla qualità del software e alla gestione dei progetti
- ◆ Acquisire una conoscenza approfondita delle diverse metodologie agili utilizzate nell'ingegneria del software
- ◆ Imparare a sviluppare utilizzando *Scrum*, la programmazione estrema e le tecniche di sviluppo software basate sul riuso
- ◆ Comprendere i diversi modelli di architetture di sistema e di progettazione del software, nonché l'architettura delle applicazioni cloud
- ◆ Imparare a realizzare prove al software, con metodologie come *Test Driven Development*, *Acceptance Test Driven Development*, *Behavior Driven Development*, BDD e *Cucumber*
- ◆ Approfondire la comprensione del miglioramento del processo di sviluppo del software e della sua qualità utilizzando gli standard ISO/IEC
- ◆ Introdurre il concetto di DevOps e le sue pratiche principali
- ◆ Comprendere l'importanza dell'ingegneria dei requisiti nel processo di sviluppo del software
- ◆ Approfondire la comprensione delle fonti dei requisiti e delle tecniche di elicitazione di questi, in quanto parte essenziale del processo
- ◆ Comprendere e applicare la prototipazione come parte essenziale del processo di sviluppo
- ◆ Impara a eseguire l'analisi dei requisiti e a documentarli correttamente
- ◆ Comprendere i processi di convalida e negoziazione dei requisiti, nonché la modellazione e la gestione dei requisiti
- ◆ Acquisire le conoscenze necessarie per la gestione dei sistemi critici e la specificazione formale dei requisiti
- ◆ Comprendere il quadro dell'ingegneria del software e lo standard ISO/IEC 12207
- ◆ Imparare le caratteristiche del processo di sviluppo software unificato e della pianificazione nel contesto dello sviluppo software agile
- ◆ Conoscere i diversi stili di progettazione del software distribuito e architetture software orientate ai servizi
- ◆ Imparare i concetti essenziali della progettazione di interfacce grafiche
- ◆ Comprendere le basi dello sviluppo di applicazioni web
- ◆ Conoscere le strategie e le tecniche di test del software, i fattori di qualità del software e le diverse metriche utilizzate

03

Struttura e contenuti

La struttura dei contenuti è stata progettata da un personale docente di ingegneria Informatica, consapevole dell'importanza della preparazione per approfondire quest'area di conoscenza. L'obiettivo è quello di arricchire umanamente lo studente e di innalzare il livello di conoscenza in Ingegneria del Software, attraverso le più recenti tecnologie didattiche disponibili.



“

*Questo Esperto Universitario in
Ingegneria del Software possiede
il programma di apprendimento più
completo e aggiornato del mercato”*

Modulo 1. Ingegneria del Software

- 1.1. Introduzione all'ingegneria del software e alla modellazione
 - 1.1.1. La natura del software
 - 1.1.2. La natura unica del *Webapps*
 - 1.1.3. Ingegneria software
 - 1.1.4. Il processo del software
 - 1.1.5. La pratica dell'ingegneria del software
 - 1.1.6. Miti del software
 - 1.1.7. Come tutto ha inizio
 - 1.1.8. Concetti orientati agli oggetti
 - 1.1.9. Introduzione a UML
- 1.2. Il processo del software
 - 1.2.1. Un modello generale di processo
 - 1.2.2. Modelli di processo prescrittivi
 - 1.2.3. Modelli di processo specializzati
 - 1.2.4. Il processo unificato
 - 1.2.5. Modelli di processo personali e di gruppo
 - 1.2.6. Che cos'è l'agilità?
 - 1.2.7. Che cos'è un processo agile?
 - 1.2.8. *Scrum*
 - 1.2.9. Kit di strumenti per i processi agili
- 1.3. Principi che guidano la pratica dell'ingegneria del software
 - 1.3.1. Principi che guidano il processo
 - 1.3.2. Principi che guidano la pratica
 - 1.3.3. Principi di comunicazione
 - 1.3.4. Principi di pianificazione
 - 1.3.5. Principi di modellazione
 - 1.3.6. Principi di costruzione
 - 1.3.7. Principi di implementazione
- 1.4. Comprendere i requisiti
 - 1.4.1. Ingegneria dei requisiti
 - 1.4.2. Porre le basi
 - 1.4.3. Indagine sui requisiti
 - 1.4.4. Sviluppo di casi d'uso
 - 1.4.5. Elaborazione del modello dei requisiti
 - 1.4.6. Negoziazione dei requisiti
 - 1.4.7. Convalida dei requisiti
- 1.5. Modellazione dei requisiti: scenari, informazioni e classi di analisi
 - 1.5.1. Analisi dei requisiti
 - 1.5.2. Modellazione basata su scenari
 - 1.5.3. Modelli UML che forniscono il caso d'uso
 - 1.5.4. Concetti di modellazione dei dati
 - 1.5.5. Modellazione basata sulle classi
 - 1.5.6. Diagrammi di classe
- 1.6. Modellazione dei requisiti: flusso, comportamento e modelli
 - 1.6.1. Strategie di definizione dei requisiti
 - 1.6.2. Modellazione orientata al flusso
 - 1.6.3. Diagrammi di stato
 - 1.6.4. Creare un modello comportamentale
 - 1.6.5. Diagrammi di sequenza
 - 1.6.6. Diagrammi di comunicazione
 - 1.6.7. Schemi per la modellazione dei requisiti
- 1.7. Concetti di design
 - 1.7.1. Il design nel contesto dell'ingegneria del software
 - 1.7.2. Processo del design
 - 1.7.3. Concetti di design
 - 1.7.4. Concetti di design orientati agli oggetti
 - 1.7.5. Il modello di design

- 1.8. Design architettonico
 - 1.8.1. Architettura del software
 - 1.8.2. Generi architettonici
 - 1.8.3. Stili architettonici
 - 1.8.4. Design architettonico
 - 1.8.5. Evoluzione dei design alternativi per l'architettura
 - 1.8.6. Mappatura dell'architettura con l'uso di flussi di dati
- 1.9. Design a livello di componente e basato su pattern
 - 1.9.1. Che cos'è un componente?
 - 1.9.2. Design dei componenti basato sulle classi
 - 1.9.3. Realizzazione del progetto a livello di componenti
 - 1.9.4. Design dei componenti tradizionali
 - 1.9.5. Sviluppo basato su componenti
 - 1.9.6. Modelli di progettazione
 - 1.9.7. Il design del software basato su modelli
 - 1.9.8. Modelli architettonici
 - 1.9.9. Modelli di design a livello di componenti
 - 1.9.10. Modelli di design dell'interfaccia utente
- 1.10. Qualità del software e gestione dei progetti
 - 1.10.1. Qualità
 - 1.10.2. Qualità del software
 - 1.10.3. Il dilemma della qualità del software
 - 1.10.4. Raggiungere la qualità del software
 - 1.10.5. Garanzia di qualità del software
 - 1.10.6. Lo spettro amministrativo
 - 1.10.7. Il personale
 - 1.10.8. Il prodotto
 - 1.10.9. Il processo
 - 1.10.10. Il progetto
 - 1.10.11. Principi e pratiche

Modulo 2. Ingegneria del Software Avanzato

- 2.1. Introduzione alle Metodologie Agili
 - 2.1.1. Modelli di processo e metodologie
 - 2.1.2. Agilità e processi agili
 - 2.1.3. Manifesto Agile
 - 2.1.4. Alcune metodologie agili
 - 2.1.5. Agile vs Tradizionale
- 2.2. *Scrum*
 - 2.2.1. Origini e filosofia di *Scrum*
 - 2.2.2. Valori di *Scrum*
 - 2.2.3. Flusso del processo *Scrum*
 - 2.2.4. Ruoli di *Scrum*
 - 2.2.5. Artefatti di *Scrum*
 - 2.2.6. Eventi di *Scrum*
 - 2.2.7. Storie degli utenti
 - 2.2.8. Estensioni *Scrum*
 - 2.2.9. Stime agili
 - 2.2.10. Scalabilità di *Scrum*
- 2.3. Programmazione estrema
 - 2.3.1. Motivazione e panoramica di XP
 - 2.3.2. Il ciclo di vita in XP
 - 2.3.3. I cinque valori fondamentali
 - 2.3.4. Le dodici pratiche di base di XP
 - 2.3.5. Ruoli dei partecipanti
 - 2.3.6. XP industriale
 - 2.3.7. Valutazione critica di XP

- 2.4. Sviluppo del software basato sul riutilizzo
 - 2.4.1. Riutilizzo del software
 - 2.4.2. Livelli di riutilizzo del codice
 - 2.4.3. Tecniche di riutilizzo specifiche
 - 2.4.4. Sviluppo basato su componenti
 - 2.4.5. Benefici e problemi del riutilizzo
 - 2.4.6. Pianificazione del riutilizzo
- 2.5. Architettura di sistema e modelli di progettazione del software
 - 2.5.1. Design architettonico
 - 2.5.2. Modelli architettonici generali
 - 2.5.3. Architetture con tolleranza ai guasti
 - 2.5.4. Architetture di sistemi distribuiti
 - 2.5.5. Modelli di progettazione
 - 2.5.6. Modelli gamma
 - 2.5.7. Modelli di progettazione dell'interazione
- 2.6. Architettura di applicazioni nel cloud
 - 2.6.1. Fondamenti di *Cloud Computing*
 - 2.6.2. Qualità delle applicazioni nel cloud
 - 2.6.3. Stili di architettura
 - 2.6.4. Modelli di progettazione
- 2.7. Test del software: TDD, ATDD e BDD
 - 2.7.1. Verifica e convalida del software
 - 2.7.2. Test del software
 - 2.7.3. *Test Driven Development* (TDD)
 - 2.7.4. *Acceptance Test Driven Development* (ATDD)
 - 2.7.5. *Behavior Driven Development* (BDD)
 - 2.7.6. BDD e *Cucumber*
- 2.8. Miglioramento dei processi software
 - 2.8.1. Miglioramento dei processi software
 - 2.8.2. Il miglioramento dei processi
 - 2.8.3. Modelli di maturità
 - 2.8.4. Il modello CMMI
 - 2.8.5. CMMI V2.0
 - 2.8.6. CMMI e Agile

```
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
```

```
echo "Photo gallery"
elseif ($_COOKIE['lang'] == 'rus')
echo "Фотопалерея";
}
else
echo "Foto galerija";
?></h3>-->
<div class="<?if($_GET[type]==1)"
<a href="foto-galerija.php?
<div id="left_sidebar">
<div id="left_ico">
<p <?if($_COOKIE['lang'] == 'rus')
echo "Wood-frame houses";
}elseif($_COOKIE['lang'] == 'rus')
echo "Деревянные каркасные
}elseif
echo "Koka karkasa mājas";
```

- 2.9. La qualità del prodotto software: Square
 - 2.9.1. La qualità del software
 - 2.9.2. Modello di qualità del prodotto software
 - 2.9.3. Famiglia ISO/IEC 25000
 - 2.9.4. ISO/IEC 25010: modello e caratteristiche di qualità
 - 2.9.5. ISO/IEC 25012: la qualità dei dati
 - 2.9.6. ISO/IEC 25020: misurazione della qualità del software
 - 2.9.7. ISO/IEC 25022, 25023 e 25024: metriche di qualità del software e dei dati
 - 2.9.8. ISO/IEC 25040: valutazione del software
 - 2.9.9. Processo di certificazione
- 2.10. Introduzione a DevOps
 - 2.10.1. Concetto di DevOps
 - 2.10.2. Pratiche principali

Modulo 3. Ingegneria dei requisiti

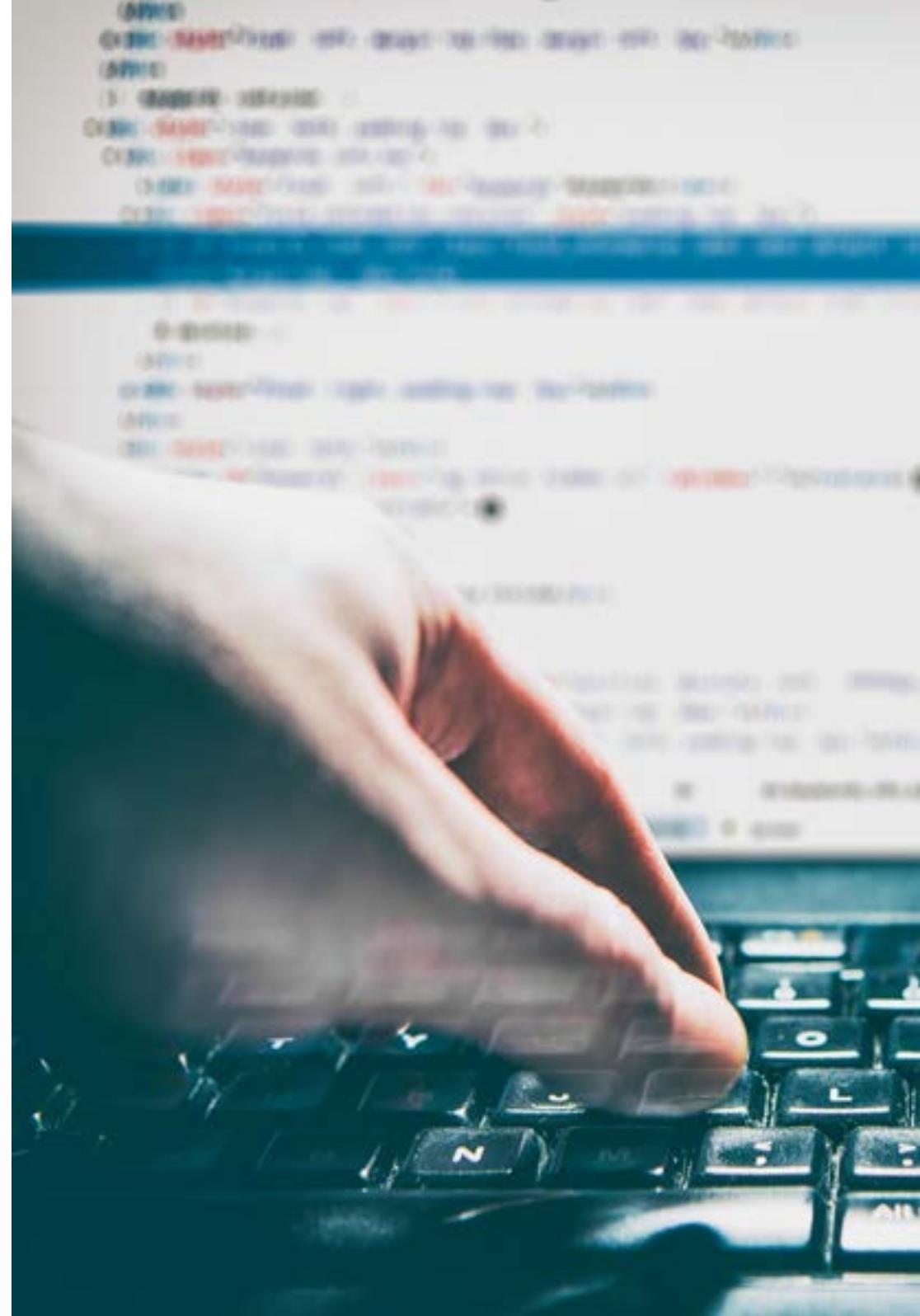
- 3.1. Introduzione all'ingegneria dei requisiti
 - 3.1.1. L'importanza dei requisiti
 - 3.1.2. Concetto di requisito
 - 3.1.3. Dimensioni dei requisiti
 - 3.1.4. Livelli e tipi di requisiti
 - 3.1.5. Caratteristiche dei requisiti
 - 3.1.6. Ingegneria dei requisiti
 - 3.1.7. Il processo di ingegneria dei requisiti
 - 3.1.8. *Frameworks* per ingegneria dei requisiti
 - 3.1.9. Pratica corretta in ingegneria dei requisiti
 - 3.1.10. Analista di business
- 3.2. Fonti dei requisiti
 - 3.2.1. Rete dei requisiti
 - 3.2.2. Gli *Stakeholders*
 - 3.2.3. Requisiti aziendali
 - 3.2.4. Documento di visione e scopo

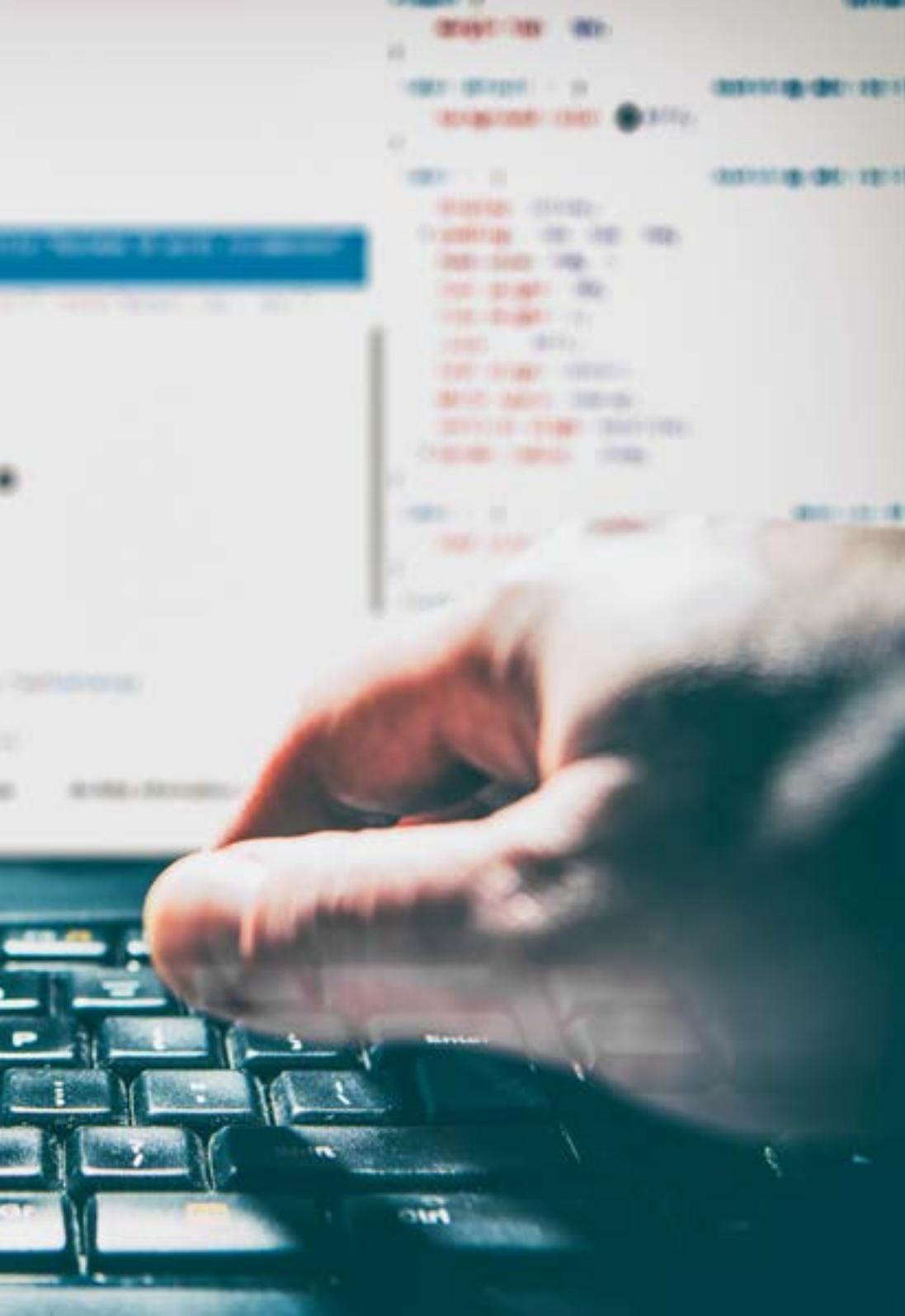
- 3.3. Tecniche di elicitazione dei requisiti
 - 3.3.1. Elicitazione dei requisiti
 - 3.3.2. Problemi di elicitazione dei requisiti
 - 3.3.3. Contesti di scoperta
 - 3.3.4. Interviste
 - 3.3.5. Osservazione e "apprendimento"
 - 3.3.6. Etnografia
 - 3.3.7. *Workshops*
 - 3.3.8. *Focus Groups*
 - 3.3.9. Questionari
 - 3.3.10. *Brainstorming* e tecniche creative
 - 3.3.11. Media del gruppo
 - 3.3.12. Analisi delle interfacce di sistema
 - 3.3.13. Analisi dei documenti e "archeologia"
 - 3.3.14. Casi d'uso e scenari
 - 3.3.15. Prototipi
 - 3.3.16. Ingegneria inversa
 - 3.3.17. Riutilizzo dei requisiti
 - 3.3.18. Pratiche corrette di elicitazione
- 3.4. Requisiti dell'utente
 - 3.4.1. Persone
 - 3.4.2. Casi di uso e storie dell'utente
 - 3.4.3. Scenari
 - 3.4.4. Tipi di scenari
 - 3.4.5. Come scoprire gli scenari
- 3.5. Tecniche di prototipazione
 - 3.5.1. Prototipazione
 - 3.5.2. Prototipi in base al percorso
 - 3.5.3. Prototipi in base alla temporalità
 - 3.5.4. La fedeltà di un prototipo
 - 3.5.5. Prototipi di interfaccia utente
 - 3.5.6. Valutazione del prototipo
- 3.6. Analisi dei requisiti
 - 3.6.1. Analisi dei requisiti
 - 3.6.2. Pratica corretta di analisi dei requisiti
 - 3.6.3. Il dizionario dei dati
 - 3.6.4. Prioritizzazione dei requisiti
- 3.7. Documentazione dei requisiti
 - 3.7.1. Il documento di specifica dei requisiti
 - 3.7.2. Struttura e contenuti di un SRS
 - 3.7.3. Documentazione in linguaggio naturale
 - 3.7.4. EARS: *Easy Approach to Requirements Syntax*
 - 3.7.5. Requisiti non funzionali
 - 3.7.6. Attributi e modelli in forma di tabella
 - 3.7.7. Pratiche corrette di specificazione
- 3.8. Convalida e negoziazione dei requisiti
 - 3.8.1. Convalida dei requisiti
 - 3.8.2. Tecniche di convalida dei requisiti
 - 3.8.3. Negoziazione dei requisiti
- 3.9. Modellazione e gestione dei requisiti
 - 3.9.1. Modellazione dei requisiti
 - 3.9.2. La prospettiva dell'utente
 - 3.9.3. La prospettiva dei dati
 - 3.9.4. La prospettiva funzionale o orientata al flusso
 - 3.9.5. La prospettiva del comportamento
 - 3.9.6. La volatilità dei requisiti
 - 3.9.7. Processo di gestione dei requisiti
 - 3.9.8. Strumenti di gestione dei requisiti
 - 3.9.9. Pratica corretta per la gestione dei requisiti
- 3.10. Sistemi critici e specifiche formali
 - 3.10.1. Sistemi critici
 - 3.10.2. Specifiche basate sul rischio
 - 3.10.3. Specifica formale

Modulo 4. Processi di ingegneria del software

- 4.1. Struttura dell'ingegneria del software
 - 4.1.1. Caratteristiche del software
 - 4.1.2. Principali processi di ingegneria del software
 - 4.1.3. Modelli di processo di sviluppo software
 - 4.1.4. Quadro di riferimento standard per il processo di sviluppo del software: lo standard ISO/IEC 12207
- 4.2. Processo unificato di sviluppo software
 - 4.2.1. Processo unificato
 - 4.2.2. Dimensioni del processo unificato
 - 4.2.3. Processo di sviluppo guidato dai casi d'uso
 - 4.2.4. Flussi di lavoro unificati fondamentali per i processi
- 4.3. Pianificazione nel contesto dello sviluppo agile del software
 - 4.3.1. Caratteristiche dello sviluppo agile del software
 - 4.3.2. Diversi orizzonti temporali di pianificazione nello sviluppo agile
 - 4.3.3. Quadro di sviluppo agile *Scrum* e pianificazione degli orizzonti temporali
 - 4.3.4. Storie dell'utente come unità di pianificazione e stima
 - 4.3.5. Tecniche comuni per ricavare una stima
 - 4.3.6. Scale di interpretazione delle stime
 - 4.3.7. *Planning Poker*
 - 4.3.8. Tipi di pianificazione comuni: pianificazione della consegna e dell'iterazione
- 4.4. Stili di progettazione del software distribuito e architetture software orientate ai servizi
 - 4.4.1. Modelli di comunicazione nei sistemi software distribuiti
 - 4.4.2. Livello intermedio o *Middleware*
 - 4.4.3. Modelli di architettura per sistemi distribuiti
 - 4.4.4. Processo generale di progettazione dei servizi software
 - 4.4.5. Aspetti di progettazione dei servizi software
 - 4.4.6. Composizione dei servizi
 - 4.4.7. Architettura dei servizi web
 - 4.4.8. Componenti infrastrutturali e SOA

- 4.5. Introduzione allo sviluppo di software guidato da modelli
 - 4.5.1. Il concetto di modello
 - 4.5.2. Sviluppo di software guidato da modelli
 - 4.5.3. Quadro di sviluppo guidato dal modello MDA
 - 4.5.4. Elementi di un modello di trasformazione
- 4.6. Progettazione dell'interfaccia grafica
 - 4.6.1. Principi di progettazione dell'interfaccia utente
 - 4.6.2. Modelli di progettazione architettonica per sistemi interattivi: Model-View-Controller (MVC)
 - 4.6.3. Esperienza dell'utente (UX *User Experience*)
 - 4.6.4. Design incentrato sull'utente
 - 4.6.5. Processo di analisi e progettazione dell'interfaccia utente
 - 4.6.6. Usabilità delle interfacce utente
 - 4.6.7. Accessibilità delle interfacce utente
- 4.7. Disegno delle applicazioni web
 - 4.7.1. Caratteristiche delle applicazioni web
 - 4.7.2. Interfaccia utente di un'applicazione web
 - 4.7.3. Design della navigazione
 - 4.7.4. Protocollo di interazione di base per le applicazioni web
 - 4.7.5. Stili di architettura per applicazioni web
- 4.8. Strategie e tecniche di test del software e fattori di qualità dello stesso
 - 4.8.1. Strategie di prova
 - 4.8.2. Progettazione dei casi di prova
 - 4.8.3. Relazione costo-qualità
 - 4.8.4. Modelli di qualità
 - 4.8.5. Famiglia di norme ISO/IEC 25000 (SQuaRE)
 - 4.8.6. Modello di qualità del prodotto (ISO 2501n)
 - 4.8.7. Modello di qualità dei dati (ISO 2501n)
 - 4.8.8. Gestione della qualità del software





- 4.9. Introduzione alle metriche nell'Ingegneria del Software
 - 4.9.1. Concetti di base: misure, metriche e indicatori
 - 4.9.2. Tipi di metriche nell'Ingegneria del Software
 - 4.9.3. Processo di misurazione
 - 4.9.4. ISO 25024. Metriche esterne e di qualità in uso
 - 4.9.5. Metrica orientata agli oggetti
- 4.10. Manutenzione e reingegnerizzazione del software
 - 4.10.1. Processo di manutenzione
 - 4.10.2. Quadro standard del processo di manutenzione: ISO/EIEC 14764
 - 4.10.3. Modello del processo di reingegnerizzazione del software
 - 4.10.4. Ingegneria inversa

“

*Un'esperienza di specializzazione
unica e decisiva per crescere
a livello professionale"*

05 Metodologia

Questo programma ti offre un modo differente di imparare. La nostra metodologia si sviluppa in una modalità di apprendimento ciclico: ***il Relearning***.

Questo sistema di insegnamento viene applicato nelle più prestigiose facoltà di medicina del mondo ed è considerato uno dei più efficaci da importanti pubblicazioni come il ***New England Journal of Medicine***.



“

Scopri il Relearning, un sistema che abbandona l'apprendimento lineare convenzionale, per guidarti attraverso dei sistemi di insegnamento ciclici: una modalità di apprendimento che ha dimostrato la sua enorme efficacia, soprattutto nelle materie che richiedono la memorizzazione”

Caso di Studio per contestualizzare tutti i contenuti

Il nostro programma offre un metodo rivoluzionario per sviluppare le abilità e le conoscenze. Il nostro obiettivo è quello di rafforzare le competenze in un contesto mutevole, competitivo e altamente esigente.

“

Con TECH potrai sperimentare un modo di imparare che sta scuotendo le fondamenta delle università tradizionali in tutto il mondo"



Avrai accesso a un sistema di apprendimento basato sulla ripetizione, con un insegnamento naturale e progressivo durante tutto il programma.



Imparerai, attraverso attività collaborative e casi reali, la risoluzione di situazioni complesse in ambienti aziendali reali.

Un metodo di apprendimento innovativo e differente

Questo programma di TECH consiste in un insegnamento intensivo, creato ex novo, che propone le sfide e le decisioni più impegnative in questo campo, sia a livello nazionale che internazionale. Grazie a questa metodologia, la crescita personale e professionale viene potenziata, effettuando un passo decisivo verso il successo. Il metodo casistico, la tecnica che sta alla base di questi contenuti, garantisce il rispetto della realtà economica, sociale e professionale più attuali.

“

Il nostro programma ti prepara ad affrontare nuove sfide in ambienti incerti e a raggiungere il successo nella tua carriera”

Il Metodo Casistico è stato il sistema di apprendimento più usato nelle migliori Scuole di Informatica del mondo da quando esistono. Sviluppato nel 1912 affinché gli studenti di Diritto non imparassero la legge solo sulla base del contenuto teorico, il metodo casistico consisteva nel presentare loro situazioni reali e complesse per prendere decisioni informate e giudizi di valore su come risolverle. Nel 1924 fu stabilito come metodo di insegnamento standard ad Harvard.

Cosa dovrebbe fare un professionista per affrontare una determinata situazione?

Questa è la domanda con cui ti confrontiamo nel metodo dei casi, un metodo di apprendimento orientato all'azione. Durante il corso, gli studenti si confronteranno con diversi casi di vita reale. Dovranno integrare tutte le loro conoscenze, effettuare ricerche, argomentare e difendere le proprie idee e decisioni.

Metodologia Relearning

TECH coniuga efficacemente la metodologia del Caso di Studio con un sistema di apprendimento 100% online basato sulla ripetizione, che combina diversi elementi didattici in ogni lezione.

Potenziamo il Caso di Studio con il miglior metodo di insegnamento 100% online: il Relearning.

Nel 2019 abbiamo ottenuto i migliori risultati di apprendimento di tutte le università online del mondo.

In TECH imparerai con una metodologia all'avanguardia progettata per formare i manager del futuro. Questo metodo, all'avanguardia della pedagogia mondiale, si chiama Relearning.

La nostra università è l'unica autorizzata a utilizzare questo metodo di successo. Nel 2019, siamo riusciti a migliorare il livello di soddisfazione generale dei nostri studenti (qualità dell'insegnamento, qualità dei materiali, struttura del corso, obiettivi...) rispetto agli indicatori della migliore università online.



Nel nostro programma, l'apprendimento non è un processo lineare, ma avviene in una spirale (impariamo, disimpariamo, dimentichiamo e re-impariamo). Pertanto, combiniamo ciascuno di questi elementi in modo concentrico. Questa metodologia ha formato più di 650.000 laureati con un successo senza precedenti in campi diversi come la biochimica, la genetica, la chirurgia, il diritto internazionale, le competenze manageriali, le scienze sportive, la filosofia, il diritto, l'ingegneria, il giornalismo, la storia, i mercati e gli strumenti finanziari. Tutto questo in un ambiente molto esigente, con un corpo di studenti universitari con un alto profilo socio-economico e un'età media di 43,5 anni.

Il Relearning ti permetterà di apprendere con meno sforzo e più performance, impegnandoti maggiormente nella tua specializzazione, sviluppando uno spirito critico, difendendo gli argomenti e contrastando le opinioni: un'equazione diretta al successo.

Dalle ultime evidenze scientifiche nel campo delle neuroscienze, non solo sappiamo come organizzare le informazioni, le idee, le immagini e i ricordi, ma sappiamo che il luogo e il contesto in cui abbiamo imparato qualcosa è fondamentale per la nostra capacità di ricordarlo e immagazzinarlo nell'ippocampo, per conservarlo nella nostra memoria a lungo termine.

In questo modo, e in quello che si chiama Neurocognitive Context-dependent E-learning, i diversi elementi del nostro programma sono collegati al contesto in cui il partecipante sviluppa la sua pratica professionale.



Questo programma offre i migliori materiali didattici, preparati appositamente per i professionisti:



Materiali di studio

Tutti i contenuti didattici sono creati appositamente per il corso dagli specialisti che lo impartiranno, per fare in modo che lo sviluppo didattico sia davvero specifico e concreto.

Questi contenuti sono poi applicati al formato audiovisivo che supporterà la modalità di lavoro online di TECH. Tutto questo, con le ultime tecniche che offrono componenti di alta qualità in ognuno dei materiali che vengono messi a disposizione dello studente.



Master class

Esistono evidenze scientifiche sull'utilità dell'osservazione di esperti terzi.

Imparare da un esperto rafforza la conoscenza e la memoria, costruisce la fiducia nelle nostre future decisioni difficili.



Pratiche di competenze e competenze

Svolgerai attività per sviluppare competenze e capacità specifiche in ogni area tematica. Pratiche e dinamiche per acquisire e sviluppare le competenze e le abilità che uno specialista deve sviluppare nel quadro della globalizzazione in cui viviamo.



Letture complementari

Articoli recenti, documenti di consenso e linee guida internazionali, tra gli altri. Nella biblioteca virtuale di TECH potrai accedere a tutto il materiale necessario per completare la tua specializzazione.





Casi di Studio

Completerai una selezione dei migliori casi di studio scelti appositamente per questo corso. Casi presentati, analizzati e monitorati dai migliori specialisti del panorama internazionale.



Riepiloghi interattivi

Il team di TECH presenta i contenuti in modo accattivante e dinamico in pillole multimediali che includono audio, video, immagini, diagrammi e mappe concettuali per consolidare la conoscenza.

Questo esclusivo sistema di specializzazione per la presentazione di contenuti multimediali è stato premiato da Microsoft come "Caso di successo in Europa".



Testing & Retesting

Valutiamo e rivalutiamo periodicamente le tue conoscenze durante tutto il programma con attività ed esercizi di valutazione e autovalutazione, affinché tu possa verificare come raggiungi progressivamente i tuoi obiettivi.



05 Titolo

L'Esperto Universitario in Ingegneria del Software garantisce, oltre alla preparazione più rigorosa e aggiornata, il conseguimento di una qualifica di Esperto Universitario rilasciata da TECH Università Tecnologica.



“

Porta a termine questo programma e ricevi la tua qualifica universitaria senza spostamenti o fastidiose formalità”

Questo **Esperto Universitario in Ingegneria del Software** possiede il programma più completo e aggiornato del mercato.

Dopo aver superato la valutazione, lo studente riceverà mediante lettera certificata* con ricevuta di ritorno, la sua corrispondente qualifica di **Esperto Universitario** rilasciata da **TECH Università Tecnologica**.

Il titolo rilasciato da **TECH Università Tecnologica** esprime la qualifica ottenuta nell'Esperto Universitario, e riunisce tutti i requisiti comunemente richiesti da borse di lavoro, concorsi e commissioni di valutazione di carriere professionali.

Titolo: **Esperto Universitario in Ingegneria del Software**

N° Ore Ufficiali: **600 o.**



*Apostille dell'Aia. Se lo studente dovesse richiedere che il suo diploma cartaceo sia provvisto di Apostille dell'Aia, TECH EDUCATION effettuerà le gestioni opportune per ottenerla pagando un costo aggiuntivo.

futuro
salute fiducia persone
educazione informazione tutor
garanzia accreditamento insegnamento
istituzioni tecnologia apprendimento
comunità impegno
attenzione personalizzata innovazione
conoscenza presente qualità
formazione online
sviluppo istituzioni
classe virtuale lingue

tech università
tecnologica

Esperto Universitario
Ingegneria del Software

- » Modalità: **online**
- » Durata: **6 mesi**
- » Titolo: **TECH Università
Tecnologica**
- » Orario: **a scelta**
- » Esami: **online**

Esperto Universitario Ingegneria del Software

