

Master Semipresenziale

Creazione di Interfacce e Applicazioni in Rete





tech *universidad
tecnológica*

Master Semipresenziale Creazione di Interfacce e Applicazioni in Rete

Modalità: Semipresenziale (Online + Tirocinio)

Durata: 12 mesi

Titolo: TECH Università Tecnologica

Ore teoriche: 1.620

Accesso al sito web: www.techtute.com/it/informatica/master-semipresenziale/master-semipresenziale-creazione-interfacce-applicazioni-rete

Indice

01

Presentazione

pag. 4

02

Perché iscriversi a questo
Master Semipresenziale?

pag. 8

03

Obiettivi

pag. 12

04

Competenze

pag. 18

05

Struttura e contenuti

pag. 22

06

Tirocinio

pag. 34

07

Dove posso svolgere il
Tirocinio?

pag. 40

08

Metodologia

pag. 44

09

Titolo

pag. 52

01

Presentazione

Le interfacce e le applicazioni di rete forniscono una piattaforma per condividere informazioni in modo efficiente e sicuro, promuovendo la produttività e l'innovazione in vari settori, dall'istruzione all'eCommerce. Fornendo interfacce intuitive e un'esperienza utente ottimizzata, queste applicazioni di rete migliorano l'esperienza utente e aumentano la soddisfazione, con conseguente maggiore ritenzione e fidelizzazione dei clienti. Per questo TECH ha sviluppato questo programma completo, in modo che gli informatici si specializzino in un campo molto richiesto nelle organizzazioni. Attraverso un formato che combina lo studio teorico 100% online e un tirocinio di 3 settimane in una prestigiosa azienda informatica.



A close-up, shallow depth-of-field photograph of a person's hands typing on a white laptop keyboard. The focus is on the keys and the fingers, with the background being a bright, out-of-focus white. The image is positioned on the left side of the slide, partially overlapping a diagonal teal-colored section that extends from the top right corner.

“

Scegli TECH! Avrai l'opportunità di lavorare su progetti pratici e collaborativi, incoraggiando lo sviluppo delle tue capacità di lavoro di squadra e risoluzione dei problemi”

Le interfacce e le applicazioni di rete facilitano la comunicazione e la collaborazione tra utenti geograficamente sparsi, consentendo loro di condividere informazioni in modo efficiente e in tempo reale. Inoltre, migliorano l'accessibilità a servizi e risorse, incoraggiando l'inclusione e le pari opportunità, e aiutano a raccogliere e analizzare i dati, consentendo di prendere decisioni informate e migliorare continuamente l'esperienza utente.

Nasce così questo Master Semipresenziale, la cui prima fase immergerà l'informatico nella teoria sulla Creazione di Interfacce e Applicazioni in Rete. In questo modo, il programma si concentrerà sulla fornitura di solide conoscenze sulla progettazione di interfacce, usabilità e adattabilità alla diversità umana. Inoltre, gli studenti approfondiranno il funzionamento dei sistemi, il modello relazionale e la manipolazione dei dati tramite SQL.

Inoltre, il piano di studi introdurrà i professionisti nell'uso di HTML, CSS e JavaScript per la creazione web, nonché nell'architettura MVC e nella programmazione orientata ai componenti. Impareranno anche a conoscere il software libero e i vari strumenti disponibili, tra cui sistemi operativi, gestione aziendale e CMS come WordPress.

Infine, verranno analizzate la strategia di riutilizzo del software, i modelli di progettazione e l'uso di *framework* per la creazione di interfacce utente grafiche e lo sviluppo web, approfondendo il Model-view-controller (MVC). Verranno inoltre analizzate metodologie agili come Scrum, programmazione estrema e sviluppo basato sul riutilizzo.

A questo va aggiunto che questo titolo accademico di TECH consentirà agli studenti di sviluppare le loro capacità in un tirocinio pratico completo, in cui dedicheranno 3 settimane a lavorare in un'azienda informatica di alto livello, imparando dai migliori esperti nel campo della Creazione di Interfacce e Applicazioni in Rete. Così, acquisiranno le conoscenze e le competenze necessarie per eccellere in un settore altamente competitivo.

Questo **Master Semipresenziale in Creazione di Interfacce e Applicazioni in Rete** possiede il programma più completo e aggiornato del mercato. Le caratteristiche principali del programma sono:

- ♦ Sviluppo di oltre 100 casi pratici presentati da professionisti del settore dell'Informatica, esperti di creazione di interfacce e web, oltre professori universitari con una vasta esperienza nel paziente critico
- ♦ Contenuti grafici, schematici ed eminentemente pratici che forniscono informazione tecnica riguardo alle discipline essenziali per l'esercizio della professione
- ♦ Sistema di apprendimento interattivo basato su algoritmi per prendere decisioni su situazioni date
- ♦ Lezioni teoriche, domande all'esperto, forum di discussione su argomenti controversi e lavori di riflessione individuale
- ♦ Contenuti disponibili da qualsiasi dispositivo fisso o mobile dotato di connessione a internet
- ♦ Possibilità di svolgere un tirocinio presso una delle migliori aziende del settore



Aggiungerai al tuo studio online un tirocinio presso una società informatica d'élite, dotata del più alto livello tecnologico"

“

La creazione di interfacce e applicazioni in rete rappresenta un potente strumento per promuovere il progresso tecnologico e l'interconnessione globale”

Il programma di questo Master di specializzazione semipresenziale è orientato all'aggiornamento dei professionisti nella creazione di interfacce e applicazioni web, che richiedono un alto livello di qualifica. I contenuti sono basati sulle ultime evidenze scientifiche, orientati in modo didattico per integrare le conoscenze teoriche nella pratica Informatica, e gli elementi teorico-pratici faciliteranno l'aggiornamento delle conoscenze e permetteranno di prendere decisioni in situazioni complesse.

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 sui Problemi, mediante il quale lo specialista deve cercare di risolvere le diverse situazioni di pratica professionale che gli si presentano durante il corso. Lo studente potrà usufruire di un innovativo sistema di video interattivi creati da esperti di rinomata fama.

Approfondirai i tuoi sistemi di banche dati avanzate, XML, banche dati parallele e distribuite, il tutto attraverso una vasta biblioteca con le risorse multimediali più innovative.

Affronterai architetture di sistema, test software e standard ISO/IEC, oltre a approfondire il concetto di DevOps e le sue pratiche principali.



02

Perché iscriversi a questo Master Semipresenziale?

In molti campi del lavoro e della conoscenza, la teoria da sola non è sufficiente per ottenere uno sviluppo o un progresso reale. In questo modo, lo studente otterrà una visione completa del panorama attuale in materia, guidato durante l'intero processo da veri esperti del settore.



“

Ottieni opportunità di lavoro ben retribuite in aziende tecnologiche leader, start-up innovative o intraprendi i tuoi progetti aziendali”

1. Aggiornarsi sulla base delle più recenti tecnologie disponibili

Nel campo della creazione di interfacce e applicazioni di rete, l'integrazione di sistemi di Intelligenza Artificiale e Machine Learning consente di sviluppare interfacce più intuitive e adattive, in grado di anticipare le esigenze dell'utente e personalizzare l'esperienza in modo dinamico. Inoltre, il cloud computing fornisce un'infrastruttura scalabile e flessibile per lo sviluppo e la distribuzione di applicazioni web, consentendo agli sviluppatori di creare esperienze multiplatforma e ad alte prestazioni che si adattano alle esigenze del mercato di oggi.

2. Approfondire nuove competenze dall'esperienza dei migliori specialisti

Il personale docente che accompagnerà l'informatico durante l'intero tirocinio rappresenta un'approvazione di prim'ordine e una garanzia di aggiornamento senza precedenti. Con un tutor appositamente assegnato, lo studente sarà in grado di lavorare in progetti reali in un ambiente all'avanguardia, che gli consentirà di incorporare nella sua pratica quotidiana le procedure e gli strumenti più efficaci per la Creazione di Interfacce e Applicazioni in Rete.

3. Accedere ad ambienti professionali di prim'ordine

TECH seleziona con cura tutti le aziende disponibili per lo svolgimento del Tirocinio. Grazie a ciò, lo specialista avrà accesso garantito a un'organizzazione prestigiosa nel settore dello sviluppo di interfacce e applicazioni web. In questo modo, potrà vedere il lavoro quotidiano di un'area di lavoro esigente, rigorosa ed esaustiva, che applica sempre i progressi più recenti nella sua metodologia di lavoro.





4. Combinare la migliore teoria con la pratica più avanzata

Il mercato accademico è pieno di programmi educativi poco adattabili alle attività quotidiane dello specialista e che richiedono lunghe ore di carico didattico, spesso poco compatibili con la vita personale e professionale. TECH offre un nuovo modello di apprendimento, 100% pratico, che consente di conoscere le procedure più avanzate nel campo della Creazione di Interfacce e Applicazioni in Rete e, soprattutto, di metterle in pratica in sole 3 settimane.

5. Accedere a nuove opportunità

Con il progresso della tecnologia e la crescente domanda di esperienze digitali innovative, gli informatici hanno l'opportunità di esplorare e sfruttare strumenti emergenti come l'Intelligenza Artificiale e il cloud computing. Inoltre, la globalizzazione e la crescente interconnessione di dispositivi e piattaforme digitali creano un panorama favorevole alla collaborazione internazionale e alla creazione di reti professionali, consentendo agli esperti di creare interfacce e applicazioni di rete di espandere la loro portata e contribuire allo sviluppo di soluzioni innovative su scala globale.

“

Avrai l'opportunità di svolgere un tirocinio all'interno di un centro a tua scelta”

03

Obiettivi

Gli obiettivi di questo Master Semipresenziale saranno basati sulla fornitura agli informatici di una formazione completa e specializzata in progettazione, sviluppo e gestione di interfacce e applicazioni digitali. I professionisti saranno così preparati a comprendere le ultime tendenze e tecnologie nel campo dell'informatica e della tecnologia dell'informazione, nonché a padroneggiare strumenti e metodologie di progettazione e programmazione. Inoltre, verrà incoraggiato il pensiero critico e la risoluzione dei problemi, promuovendo l'innovazione e la creatività nello sviluppo di soluzioni digitali efficienti, intuitive e accessibili a un'ampia varietà di utenti.



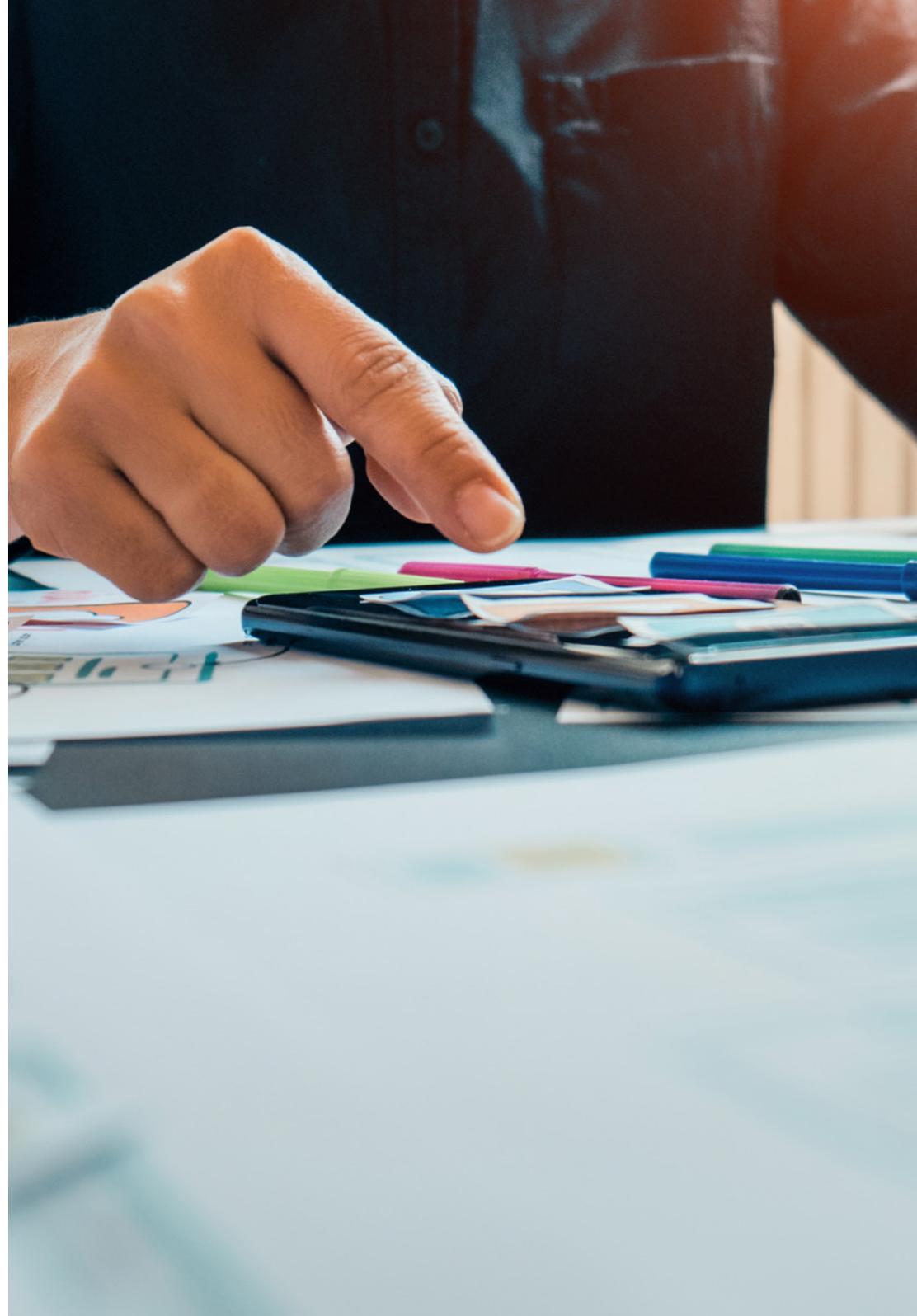
“

Questo Master Semipresenziale ti trasformerà in un professionista altamente qualificato e con una visione strategica, per guidare progetti digitali innovativi in un ambiente tecnologico in continua evoluzione”



Obiettivo generale

- ♦ L'obiettivo del Master Semipresenziale in Creazione di Interfacce e Applicazioni in Rete è quello di formare scientificamente e tecnologicamente gli studenti per esercitare l'ingegneria informatica, fornendo loro una solida base di conoscenze in settori chiave. In questo modo, i professionisti acquisiranno conoscenze approfondite nel vasto campo del calcolo, che spaziano dagli aspetti teorici alle applicazioni pratiche nella progettazione e nello sviluppo di interfacce e applicazioni di rete. Inoltre, verrà enfatizzata la comprensione della struttura dei computer, consentendo di comprendere i componenti e il funzionamento interno dei sistemi informatici. Si approfondirà anche l'ingegneria del software, fornendo agli informatici le competenze necessarie per pianificare, progettare, implementare e mantenere software di alta qualità.





Obiettivi specifici

Modulo 1. Interazione Uomo-Macchina

- ♦ Acquisire solide conoscenze relative all'interazione uomo-macchina e alla creazione di interfacce utilizzabili
- ♦ Capire l'importanza dell'usabilità nelle applicazioni e perché bisogna tenerne conto quando si progetta il software
- ♦ Comprendere i diversi tipi di diversità umana, le limitazioni che comportano e come adattare le interfacce in base alle esigenze specifiche di ciascuno di essi
- ♦ Imparare il processo di progettazione di un'interfaccia, dall'analisi dei requisiti alla valutazione, passando per le diverse fasi intermedie necessarie a creare un'interfaccia adeguata
- ♦ Conoscere le diverse linee guida sull'accessibilità, gli standard che le stabiliscono e gli strumenti che ci permettono di valutarle
- ♦ Comprendere i diversi metodi di interazione con il computer, utilizzando periferiche e dispositivi

Modulo 2. Database

- ♦ Imparare le diverse applicazioni e finalità dei sistemi di database, nonché il loro funzionamento e la loro architettura
- ♦ Comprendere il modello relazionale, dalla sua struttura e operazioni all'algebra relazionale estesa
- ♦ Approfondire cosa sono i database SQL, come funzionano, la definizione dei dati e la creazione di query, dalle più elementari alle più avanzate
- ♦ Imparare a progettare database utilizzando il modello entità-relazionale, a creare diagrammi e a conoscere le caratteristiche del modello E-R esteso
- ♦ Approfondire la progettazione di database relazionali, analizzando le diverse forme normali e gli algoritmi di decomposizione
- ♦ Porre le basi per comprendere il funzionamento dei database NoSQL e introdurre il database Mongo DB

Modulo 3. Sviluppo delle Applicazioni Web

- ♦ Imparare le caratteristiche del linguaggio di markup HTML e il suo utilizzo nella creazione di siti web insieme ai fogli di stile CSS
- ♦ Imparare a utilizzare il linguaggio di programmazione orientato al browser JavaScript e alcune delle sue caratteristiche principali
- ♦ Comprendere i concetti di programmazione orientata ai componenti e di architettura dei componenti
- ♦ Imparare a utilizzare il *Framework* per *Front-End* Bootstrap per la progettazione di siti web
- ♦ Comprendere la struttura del Model-View-Controller nello sviluppo di siti web dinamici
- ♦ Conoscere l'architettura orientata ai servizi e le basi del protocollo HTTP

Modulo 4. Software libero e conoscenza aperta

- ♦ Imparare i concetti di Software Libero e Conoscenza Aperta, nonché i diversi tipi di licenze associate
- ♦ Conoscere i principali strumenti gratuiti disponibili in diversi settori come i sistemi operativi, la gestione aziendale, i gestori di contenuti, la creazione di contenuti multimediali
- ♦ Comprendere l'importanza e i vantaggi del software libero nel mondo degli affari, sia in termini di caratteristiche che di costi
- ♦ Approfondire la conoscenza del sistema operativo GNU/Linux, delle diverse distribuzioni esistenti e di come realizzarne adattamenti personalizzati
- ♦ Conoscere il funzionamento e lo sviluppo di WordPress, dato che questo CMS rappresenta più del 35% dei siti web attivi nel mondo, e più del 60% nel caso particolare dei CMS
- ♦ Comprendere il funzionamento del sistema operativo dei dispositivi mobili Android e le basi dello sviluppo di applicazioni mobili, sia in modo nativo che con *Framework* multiplatforma

Modulo 5. Database avanzati

- ♦ Introdurre i diversi sistemi di database attualmente disponibili sul mercato
- ♦ Apprendere l'uso di XML e dei database per il web
- ♦ Comprendere il funzionamento di database avanzati come i database paralleli e distribuiti
- ♦ Comprendere l'importanza dell'indicizzazione e dell'associazione nei sistemi di database
- ♦ Comprendere il funzionamento dei sistemi di elaborazione transazionale e i sistemi di recupero
- ♦ Acquisire conoscenze relative ai database non relazionali e al data mining

Modulo 6. Ingegneria del software

- ♦ 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

Modulo 7. Programmazione Avanzata

- ♦ Approfondire la conoscenza della programmazione, soprattutto in relazione a quella orientata agli oggetti, e dei diversi tipi di relazioni tra classi
- ♦ Conoscere i diversi modelli di progettazione per i problemi orientati agli oggetti
- ♦ Imparare la programmazione guidata dagli eventi e lo sviluppo di interfacce utente con Qt
- ♦ Acquisire le conoscenze essenziali della programmazione concorrente, dei processi e dei thread
- ♦ Imparare a gestire l'uso dei thread e della sincronizzazione, nonché a risolvere i problemi più comuni della programmazione concorrente
- ♦ Comprendere l'importanza della documentazione e dei test nello sviluppo del software

Modulo 8. Riutilizzo di software

- ♦ Conoscere il panorama del riutilizzo di software
- ♦ Apprendere i diversi modelli relativi al riutilizzo del software, in termini di progettazione, creazione, struttura e comportamento
- ♦ Introdurre il concetto di *Framework*, nonché conoscerne le principali tipologie come quelle destinate alla progettazione di interfacce grafiche, allo sviluppo di applicazioni web e alla gestione della persistenza degli oggetti nei database
- ♦ Comprendere il funzionamento del pattern Model-View-Controller (MVC), ampiamente utilizzato

Modulo 9. Intelligenza Artificiale e Ingegneria della Conoscenza

- ♦ Porre le basi dell'Intelligenza Artificiale e dell'Ingegneria della Conoscenza, fornendo un breve sguardo alla storia dell'Intelligenza Artificiale fino ai giorni nostri
- ♦ Comprendere i concetti essenziali della ricerca nell'Intelligenza Artificiale, sia la ricerca informata che quella non
- ♦ Capire come funziona l'Intelligenza Artificiale nei giochi
- ♦ Apprendere i concetti fondamentali delle reti neurali e l'uso degli algoritmi genetici
- ♦ Acquisire i meccanismi appropriati per rappresentare la conoscenza, soprattutto in vista del web semantico
- ♦ Comprendere il funzionamento dei sistemi esperti e dei sistemi di supporto alle decisioni

Modulo 10. Ingegneria del Software Avanzato

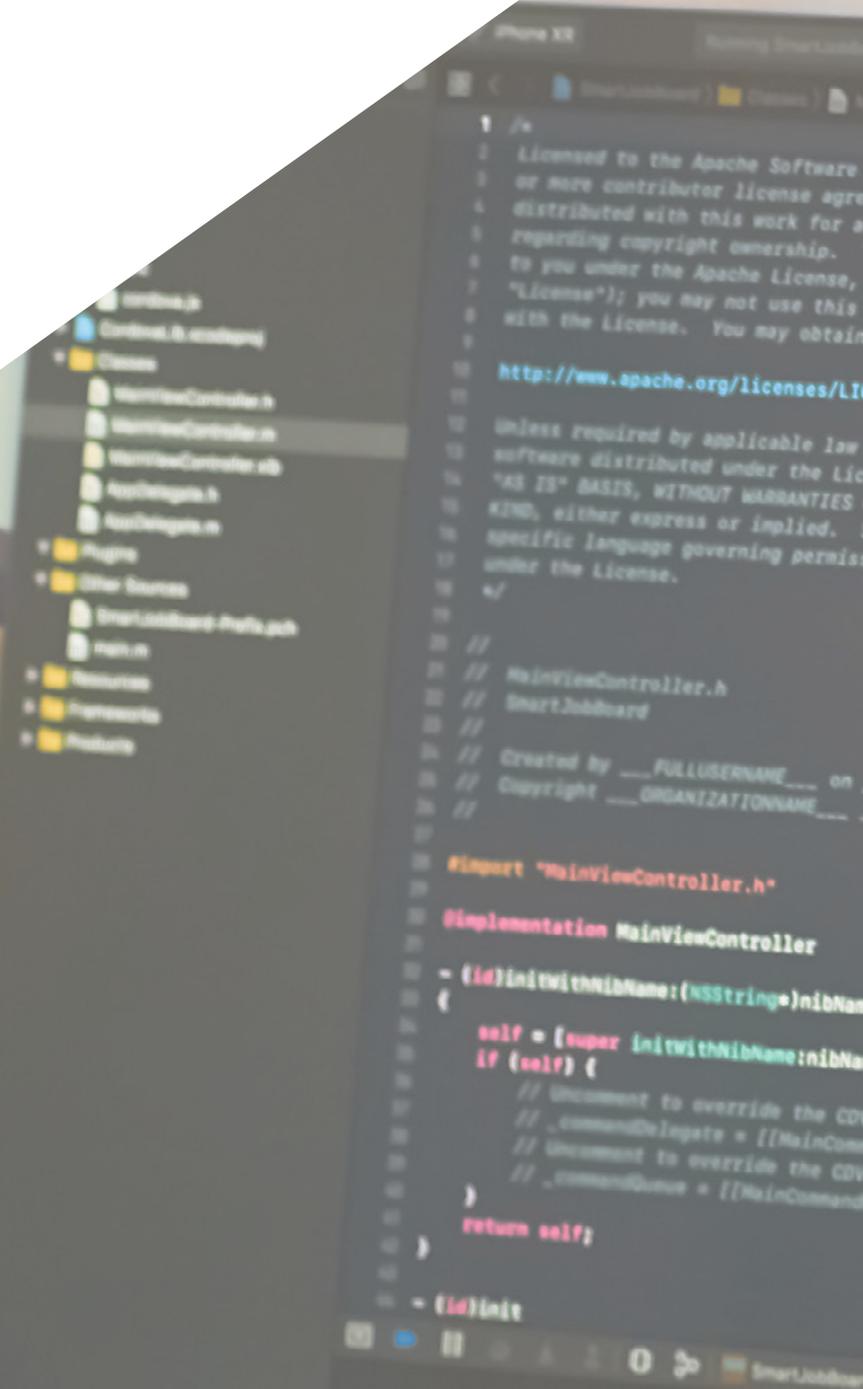
- ♦ Acquisire una conoscenza approfondita delle diverse metodologie agili utilizzate nell'ingegneria del software
- ♦ Imparare a sviluppare utilizzando *Scrum*, la programmazione estrema e le tecniche di sviluppo software basate sul riuso
- ♦ Comprendere i diversi modelli di architetture di sistema e 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



Sarai dotato di competenze tecniche, conoscenze teoriche e competenze professionali per distinguerti nei ruoli relativi alla progettazione di interfacce e applicazioni di rete”

04 Competenze

Tra le competenze acquisite in questo programma universitario sarà compresa la capacità di analizzare e comprendere le esigenze dell'utente e progettare interfacce intuitive e funzionali che migliorano la sua esperienza. Inoltre, gli studenti svilupperanno competenze tecniche in settori come la programmazione web, la progettazione di database e l'implementazione di sistemi distribuiti. Sarà inoltre incoraggiato il lavoro di squadra e la comunicazione con altri professionisti del settore, e la capacità di adattamento ai cambiamenti tecnologici e la capacità di apprendimento continuo, per rimanere aggiornati in un ambiente digitale in continua evoluzione.





“

Sarai in grado di progettare interfacce e applicazioni interessanti e funzionali, garantendo un'esperienza utente ottimale, grazie alla rivoluzionaria metodologia Relearning”

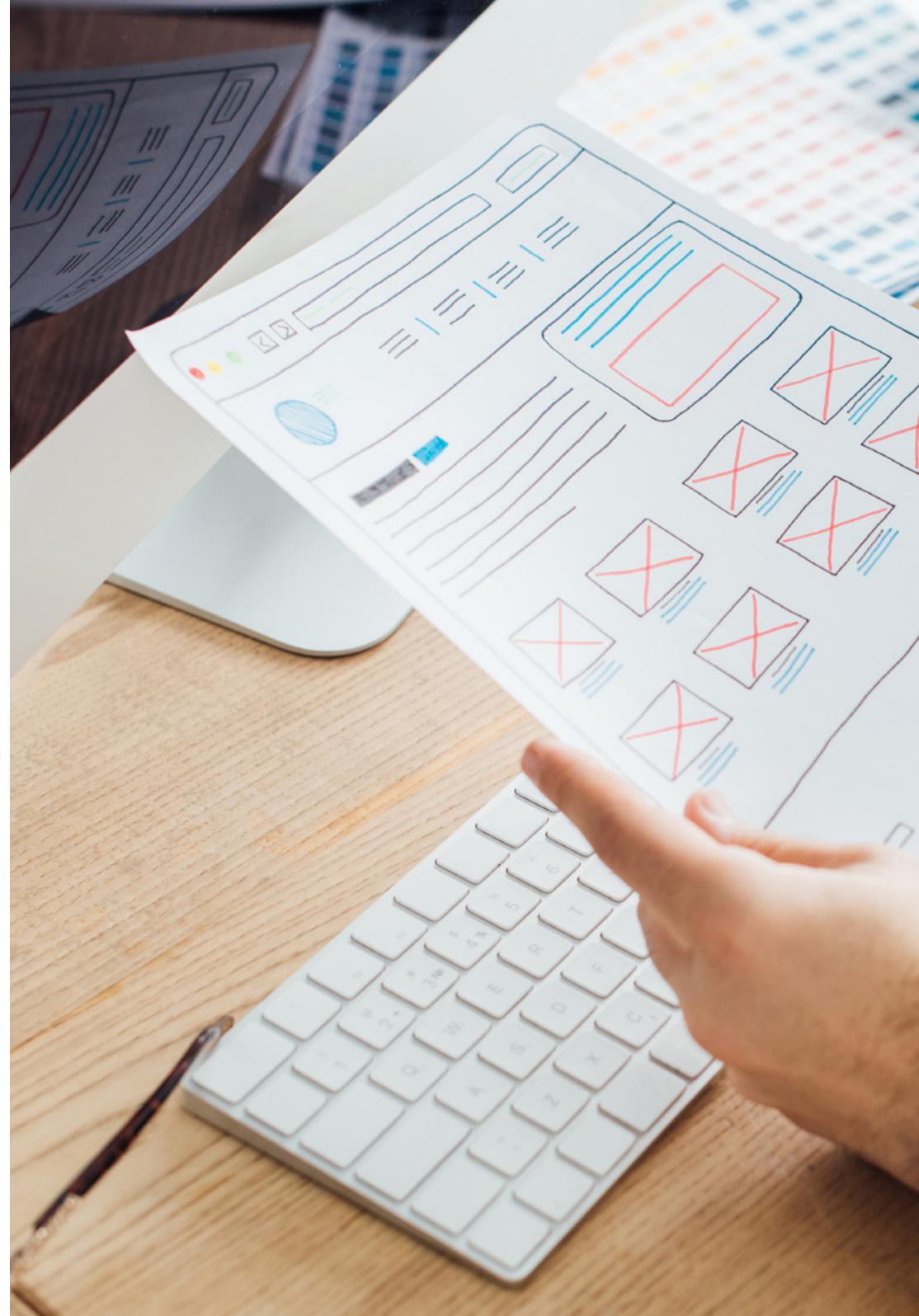


Competenza generale

- ◆ Acquisire le competenze necessarie per la pratica professionale dell'ingegneria informatica con la conoscenza di tutti i fattori necessari per svolgerla con qualità e solvibilità

“

Acquisirai competenze tecniche nello sviluppo del software, padroneggiando vari linguaggi di programmazione, framework e strumenti di sviluppo web e mobile”





Competenze specifiche

- ◆ Approfondire tutte le sfaccettature dell'interazione uomo-macchina e di come queste coinvolgano gli sviluppi informatici
- ◆ Essere abili nell'uso delle banche dati
- ◆ Sviluppare diversi tipi di applicazioni di rete
- ◆ Descrivere e sfruttare il software libero e la conoscenza aperta sul web
- ◆ Lavorare come ingegnere del software
- ◆ Controllare l'uso di database avanzati
- ◆ Eseguire una programmazione avanzata
- ◆ Saper riutilizzare il software
- ◆ Creare interfacce e applicazioni di rete
- ◆ Padroneggiare i diversi sistemi di lavoro dell'ingegneria del software avanzata

05

Struttura e contenuti

Dai fondamenti dell'interazione uomo-computer, alle tecnologie più avanzate nel campo dell'Intelligenza Artificiale e del cloud computing, questo programma coprirà aspetti essenziali come l'analisi dei requisiti, la progettazione di interfacce utilizzabili, la programmazione web e mobile, l'architettura dei sistemi di rete e la sicurezza informatica. Inoltre, i professionisti affronteranno argomenti relativi ai database, all'ingegneria del software e al software libero, essendo preparati ad affrontare le sfide e a cogliere le opportunità nel settore della tecnologia digitale.





“

Sarai dotato di competenze nell'uso di strumenti e tecnologie moderne, nonché nell'applicazione di metodologie agili e buone pratiche di sviluppo software”

Modulo 1. Interazione Uomo-Macchina

- 1.1. Introduzione all'interazione uomo-macchina
 - 1.1.1. Che cos'è l'interazione uomo-macchina
 - 1.1.2. Rapporto dell'interazione uomo-macchina con altre discipline
 - 1.1.3. L'interfaccia utente
 - 1.1.4. Usabilità e accessibilità
 - 1.1.5. Esperienza utente e design incentrato sull'utente
- 1.2. Il computer e l'interazione: Interfaccia utente ed paradigma di interazione
 - 1.2.1. L'interazione
 - 1.2.2. Paradigmi di interazione e stili di interazione
 - 1.2.3. Evoluzione delle interfacce utente
 - 1.2.4. Interfacce utente classica: WIMP/GUI, comandi, voce, realtà virtuale
 - 1.2.5. Interfacce utente innovative: Cellulari, portatili, collaborativi, BCI
- 1.3. Il fattore umano: Aspetti psicologici e cognitivi
 - 1.3.1. L'importanza del fattore umano nell'interazione
 - 1.3.2. Elaborazione dell'informazione umana
 - 1.3.3. Input e output di informazioni: Visivo, uditivo e tattile
 - 1.3.4. Percezione e attenzione
 - 1.3.5. Conoscenza e modelli mentali: Rappresentazione, organizzazione e acquisizione
- 1.4. Il fattore umano: Limitazioni sensoriali e fisiche
 - 1.4.1. Diversità funzionale, disabilità e deficit
 - 1.4.2. Diversità visiva
 - 1.4.3. Diversità dell'udito
 - 1.4.4. Diversità cognitiva
 - 1.4.5. Diversità motoria
 - 1.4.6. Il caso degli immigrati digitali
- 1.5. Processo di progettazione (I): analisi dei requisiti per la progettazione dell'interfaccia utente
 - 1.5.1. Progettazione incentrata sull'utente
 - 1.5.2. Che cos'è l'analisi dei requisiti?
 - 1.5.3. Raccolta di informazioni
 - 1.5.4. Analisi e interpretazione delle informazioni
 - 1.5.5. Analisi di usabilità e accessibilità
- 1.6. Processo di progettazione (II): Prototipazione e analisi delle attività
 - 1.6.1. Progetto concettuale
 - 1.6.2. Prototipazione
 - 1.6.3. Analisi gerarchica dei compiti
- 1.7. Processo di progettazione (III): La valutazione
 - 1.7.1. La valutazione nel processo di progettazione: Obiettivi e metodi
 - 1.7.2. Metodi di valutazione senza utenti
 - 1.7.3. Metodi di valutazione con gli utenti
 - 1.7.4. Standard e norme di valutazione
- 1.8. Accessibilità: Definizione e linee guida
 - 1.8.1. Accessibilità e design universale
 - 1.8.2. L'iniziativa WAI e le linee guida WCAG
 - 1.8.3. Linee guida WCAG 2.0 e 2.1
- 1.9. Accessibilità: Valutazione e diversità funzionale
 - 1.9.1. Strumenti di valutazione dell'accessibilità del web
 - 1.9.2. Accessibilità e diversità funzionale
- 1.10. Il computer e l'interazione: Unità periferiche e dispositivi
 - 1.10.1. Dispositivi e periferiche tradizionali
 - 1.10.2. Dispositivi e periferiche alternative
 - 1.10.3. Telefoni cellulari e tablet
 - 1.10.4. Diversità funzionale, interazione e periferiche

Modulo 2. Database

- 2.1. Applicazioni e scopi dei sistemi di database
 - 2.1.1. Applicazioni di diversi sistemi di database
 - 2.1.2. Scopo dei diversi sistemi di database
 - 2.1.3. Visione dei dati
- 2.2. Database e architettura
 - 2.2.1. Database relazionale
 - 2.2.2. Progettazione di database
 - 2.2.3. Database a oggetti e semi-strutturati
 - 2.2.4. Memorizzazione dei dati e interrogazioni
 - 2.2.5. Gestione delle transazioni
 - 2.2.6. Estrazione e analisi dei dati
 - 2.2.7. Architettura del database
- 2.3. Il modello relazionale: struttura, operazioni e algebra relazionale estesa
 - 2.3.1. La struttura dei database relazionali
 - 2.3.2. Operazioni fondamentali dell'algebra relazionale
 - 2.3.3. Altre operazioni dell'algebra relazionale
 - 2.3.4. Operazioni di algebra relazionale estesa
 - 2.3.5. Valori nulli
 - 2.3.6. Modifica del database
- 2.4. SQL (I)
 - 2.4.1. Cos'è SQL?
 - 2.4.2. La definizione di dati
 - 2.4.3. La struttura di base delle query SQL
 - 2.4.4. Operazioni sugli insiemi
 - 2.4.5. Funzioni di aggregazione
 - 2.4.6. Valori nulli
- 2.5. SQL (II)
 - 2.5.1. Subquery annidate
 - 2.5.2. Query complesse
 - 2.5.3. Visualizzazioni
 - 2.5.4. Cursori
 - 2.5.5. Query complesse
 - 2.5.6. Trigger
- 2.6. Progettazione di database e modello E-R
 - 2.6.1. Panoramica del processo di progettazione
 - 2.6.2. Modello entità-relazione
 - 2.6.3. Restrizioni
- 2.7. Diagrammi entità-relazione
 - 2.7.1. Diagrammi entità-relazione
 - 2.7.2. Aspetti di progettazione delle entità e delle relazioni
 - 2.7.3. Insiemi di entità deboli
- 2.8. Il modello esteso di entità-relazione
 - 2.8.1. Caratteristiche del modello E-R esteso
 - 2.8.2. Progettazione del database
 - 2.8.3. Riduzione a schemi relazionali
- 2.9. Progettazione di database relazionali
 - 2.9.1. Caratteristiche di un buon progetto relazionale
 - 2.9.2. Domini atomici e prima forma normale (1FN)
 - 2.9.3. Decomposizione mediante dipendenze funzionali
 - 2.9.4. Teoria della dipendenza funzionale
 - 2.9.5. Algoritmi di decomposizione
 - 2.9.6. Decomposizione con dipendenze multivariate
 - 2.9.7. Altre forme normali
 - 2.9.8. Processo di progettazione del database
- 2.10. Database NoSQL
 - 2.10.1. Cosa sono i database NoSQL?
 - 2.10.2. Analisi delle diverse opzioni NoSQL e delle loro caratteristiche
 - 2.10.3. MongoDB

Modulo 3. Sviluppo delle Applicazioni Web

- 3.1. Linguaggi di mercato HTML5
 - 3.1.1. Nozioni di base sulla HTML
 - 3.1.2. Nuovi elementi HTML 5
 - 3.1.3. Moduli: nuovi controlli
- 3.2. Introduzione ai fogli di stile CSS
 - 3.2.1. Primi passi con CSS
 - 3.2.2. Introduzione ai CSS3
- 3.3. Linguaggio di scripting del browser: JavaScript
 - 3.3.1. Nozioni di base di JavaScript
 - 3.3.2. DOM
 - 3.3.3. Eventi
 - 3.3.4. JQuery
 - 3.3.5. Ajax
- 3.4. Concetto di programmazione orientata ai componenti
 - 3.4.1. Contesto
 - 3.4.2. Componenti e interfacce
 - 3.4.3. Stati di un componente
- 3.5. Architettura dei componenti
 - 3.5.1. Architetture attuali
 - 3.5.2. Integrazione e distribuzione dei componenti
- 3.6. *Framework Front-End*: Bootstrap
 - 3.6.1. Design con la rete
 - 3.6.2. Formolari
 - 3.6.3. Componenti
- 3.7. Controllore della vista del modello
 - 3.7.1. Metodi di sviluppo web
 - 3.7.2. Pattern di progettazione: MVC
- 3.8. Tecnologie Grid informative
 - 3.8.1. Aumento delle risorse informatiche
 - 3.8.2. Concetto di tecnologia Grid

- 3.9. Architetture orientate ai servizi
 - 3.9.1. SOA e servizi web
 - 3.9.2. Topologia del servizio web
 - 3.9.3. Piattaforme di servizi web
- 3.10. Protocollo HTTP
 - 3.10.1. Messaggi
 - 3.10.2. Sessioni persistenti
 - 3.10.3. Sistema crittografico
 - 3.10.4. Funzionamento del protocollo HTTPS

Modulo 4. Software libero e conoscenza aperta

- 4.1. Introduzione al software libero
 - 4.1.1. Storia del software libero
 - 4.1.2. "Libertà" del software
 - 4.1.3. Licenze per l'uso di strumenti software
 - 4.1.4. Proprietà intellettuale del software
 - 4.1.5. Qual è la motivazione per l'utilizzo del software libero?
 - 4.1.6. Miti del software libero
 - 4.1.7. Top500
- 4.2. Conoscenza aperta e licenze CC
 - 4.2.1. Concetti di base
 - 4.2.2. Licenze *Creative Commons*
 - 4.2.3. Altre licenze di contenuto
 - 4.2.4. Wikipedia e altri progetti di conoscenza aperta
- 4.3. Principali strumenti di software libero
 - 4.3.1. Sistemi operativi
 - 4.3.2. Applicazioni per ufficio
 - 4.3.3. Applicazioni di gestione aziendale
 - 4.3.4. Gestione dei contenuti web
 - 4.3.5. Strumenti per la creazione di contenuti multimediali
 - 4.3.6. Altre applicazioni

- 4.4. L'azienda: il software libero e i suoi costi
 - 4.4.1. Software libero: Sì o no?
 - 4.4.2. Verità e bugie sul software libero
 - 4.4.3. Software aziendale basato su software libero
 - 4.4.4. Software aziendale basato su software libero
 - 4.4.5. Modelli di software liberi
- 4.5. Il sistema operativo GNU/Linux
 - 4.5.1. Architettura
 - 4.5.2. Struttura della directory base
 - 4.5.3. Interfaccia e implementazione del file system
 - 4.5.4. Rappresentazione interna dei file
- 4.6. Il sistema operativo mobile Android
 - 4.6.1. Storia
 - 4.6.2. Architettura
 - 4.6.3. *Fork* di Android
 - 4.6.4. Introduzione allo sviluppo di Android
 - 4.6.5. *Framework* per lo sviluppo di applicazioni mobili
- 4.7. Creazione di siti web con WordPress
 - 4.7.1. Caratteristiche e struttura di WordPress
 - 4.7.2. Creazione di siti su wordpress.com
 - 4.7.3. Installazione e configurazione di WordPress in un server proprio
 - 4.7.4. Installazione di *Plugin* ed estensione di WordPress
 - 4.7.5. Creazione di *Plugin* per WordPress
 - 4.7.6. Creazione di temi per WordPress
- 4.8. Tendenze del software libero
 - 4.8.1. Ambienti nel cloud
 - 4.8.2. Strumenti di monitoraggio
 - 4.8.3. Sistemi operativi
 - 4.8.4. *Big Data* e *Open Data* 2.0
 - 4.8.5. Informatica quantistica

- 4.9. Controllo delle versioni
 - 4.9.1. Concetti di base
 - 4.9.2. Git
 - 4.9.3. Servizi Git in cloud e self-hosted
 - 4.9.4. Altri sistemi di controllo della versione
- 4.10. Distribuzione di GNU/Linux personalizzati
 - 4.10.1. Principali distribuzioni
 - 4.10.2. Distribuzioni derivate da Debian
 - 4.10.3. Creazione pacchetti deb
 - 4.10.4. Modifica della distribuzione
 - 4.10.5. Creazione di immagini ISO

Modulo 5. Database avanzati

- 5.1. Scopo dei diversi sistemi di database
 - 5.1.1. Rassegna storica
 - 5.1.2. Database gerarchici
 - 5.1.3. Database di rete
 - 5.1.4. Database relazionali
 - 5.1.5. Database non relazionali
- 5.2. XML e database per il web
 - 5.2.1. Convalida dei documenti XML
 - 5.2.2. Trasformazioni dei documenti XML
 - 5.2.3. Memorizzazione di dati XML
 - 5.2.4. Database relazionali XML
 - 5.2.5. SQL/XML
 - 5.2.6. Database XML nativi
- 5.3. Basi di dati parallele
 - 5.3.1. Sistemi paralleli
 - 5.3.2. Architetture di database paralleli
 - 5.3.3. Parallelismo delle query
 - 5.3.4. Parallelismo tra query
 - 5.3.5. Progettazione di sistemi paralleli
 - 5.3.6. Elaborazione parallela in SQL

- 5.4. Database distribuiti
 - 5.4.1. Sistemi distribuiti
 - 5.4.2. Archiviazione distribuita
 - 5.4.3. Disponibilità
 - 5.4.4. Elaborazione distribuita delle query
 - 5.4.5. Fornitori di database distribuiti
- 5.5. Indicizzazione e associazione
 - 5.5.1. Indici ordinati
 - 5.5.2. Indici densi e radi
 - 5.5.3. Indici multilivello
 - 5.5.4. Aggiornamento dell'indice
 - 5.5.5. Associazione statica
 - 5.5.6. Come utilizzare gli indici nei database
- 5.6. Introduzione all'elaborazione transazionale
 - 5.6.1. Stati di una transazione
 - 5.6.2. Implementazione dell'atomicità e della durata
 - 5.6.3. Sequenzialità
 - 5.6.4. Recuperabilità
 - 5.6.5. Implementazione dell'isolamento
- 5.7. Sistemi di recupero
 - 5.7.1. Classificazione dei guasti
 - 5.7.2. Strutture di archiviazione
 - 5.7.3. Recupero e atomicità
 - 5.7.4. Recupero basato sul record storico
 - 5.7.5. Transazioni e recupero concorrenti
 - 5.7.6. Alta disponibilità nei database
- 5.8. Esecuzione ed elaborazione di query
 - 5.8.1. Costo di una query
 - 5.8.2. Operazione di selezione
 - 5.8.3. Ordinamento
 - 5.8.4. Introduzione all'ottimizzazione delle query
 - 5.8.5. Monitoraggio delle prestazioni



- 5.9. Database non relazionali
 - 5.9.1. Database orientati ai documenti
 - 5.9.2. Database orientati alla rete
 - 5.9.3. Database chiave-valore
- 5.10. *Data Warehouse*, OLAP e estrazione dei dati
 - 5.10.1. Componenti di un data warehouse
 - 5.10.2. Architettura di un *Data Warehouse*
 - 5.10.3. OLAP
 - 5.10.4. Funzionalità di data mining
 - 5.10.5. Altri tipi di estrazione mineraria

Modulo 6. Ingegneria del software

- 6.1. Struttura dell'ingegneria del software
 - 6.1.1. Caratteristiche del software
 - 6.1.2. Principali processi di ingegneria del software
 - 6.1.3. Modelli di processo di sviluppo software
 - 6.1.4. Quadro di riferimento standard per il processo di sviluppo del software: Standard ISO/IEC 12207
- 6.2. Processo unificato di sviluppo software
 - 6.2.1. Processo unificato
 - 6.2.2. Dimensioni del processo unificato
 - 6.2.3. Processo di sviluppo guidato dai casi d'uso
 - 6.2.4. Flussi di lavoro unificati fondamentali per i processi
- 6.3. Pianificazione nel contesto dello sviluppo agile del software
 - 6.3.1. Caratteristiche dello sviluppo agile del software
 - 6.3.2. Diversi orizzonti temporali di pianificazione nello sviluppo agile
 - 6.3.3. Quadro di sviluppo Agile Scrum e pianificazione degli orizzonti temporali
 - 6.3.4. Storie dell'utente come unità di pianificazione e stima
 - 6.3.5. Tecniche comuni per ricavare una stima
 - 6.3.6. Scale di interpretazione delle stime
 - 6.3.7. *Planning Poker*
 - 6.3.8. Tipi di pianificazione comuni: Pianificazione delle consegne e iterazione
- 6.4. Stili di progettazione del software distribuito e architetture software orientate ai servizi
 - 6.4.1. Modelli di comunicazione nei sistemi software distribuiti
 - 6.4.2. Livello intermedio o *middleware*
 - 6.4.3. Modelli di architettura per sistemi distribuiti
 - 6.4.4. Processo generale di progettazione dei servizi software
 - 6.4.5. Aspetti di progettazione dei servizi software
 - 6.4.6. Composizione dei servizi
 - 6.4.7. Architettura dei servizi web
 - 6.4.8. Componenti infrastrutturali e SOA
- 6.5. Introduzione allo sviluppo di software guidato da modelli
 - 6.5.1. Il concetto di modello
 - 6.5.2. Sviluppo di software guidato da modelli
 - 6.5.3. Quadro di sviluppo guidato dal modello MDA
 - 6.5.4. Elementi di un modello di trasformazione
- 6.6. Progettazione dell'interfaccia grafica
 - 6.6.1. Principi di progettazione dell'interfaccia utente
 - 6.6.2. Modelli di progettazione architettonica per sistemi interattivi: Model-View-Controller (MVC)
 - 6.6.3. Esperienza dell'utente (UX *User Experience*)
 - 6.6.4. Progettazione incentrata sull'utente
 - 6.6.5. Processo di analisi e progettazione dell'interfaccia utente
 - 6.6.6. Usabilità delle interfacce utente
 - 6.6.7. Accessibilità delle interfacce utente
- 6.7. Disegno delle applicazioni web
 - 6.7.1. Caratteristiche delle applicazioni web
 - 6.7.2. Interfaccia utente di un'applicazione web
 - 6.7.3. Design della navigazione
 - 6.7.4. Protocollo di interazione di base per le applicazioni web
 - 6.7.5. Stili di architettura per applicazioni web

- 6.8. Strategie e tecniche di test del software e fattori di qualità dello stesso
 - 6.8.1. Strategie di prova
 - 6.8.2. Progettazione dei casi di prova
 - 6.8.3. Relazione costo-qualità
 - 6.8.4. Modelli di qualità
 - 6.8.5. Famiglia di norme ISO/IEC 25000 (SQuaRE)
 - 6.8.6. Modello di qualità del prodotto (ISO 2501n)
 - 6.8.7. Modello di qualità dei dati (ISO 2501n)
 - 6.8.8. Gestione della qualità del software
- 6.9. Introduzione alle metriche nell'ingegneria del software
 - 6.9.1. Concetti di base: Misure, metriche e indicatori
 - 6.9.2. Tipi di metriche nell'ingegneria del software
 - 6.9.3. Processo di misurazione
 - 6.9.4. ISO 25024: Metriche esterne e di qualità in uso
 - 6.9.5. Metrica orientata agli oggetti
- 6.10. Manutenzione e reingegnerizzazione del software
 - 6.10.1. Processo di manutenzione
 - 6.10.2. Quadro standard del processo di manutenzione: ISO/EIEC 14764
 - 6.10.3. Modello del processo di reingegnerizzazione del software
 - 6.10.4. Ingegneria inversa

Modulo 7. Programmazione Avanzata

- 7.1. Introduzione alla programmazione orientata agli oggetti
 - 7.1.1. Introduzione alla programmazione orientata agli oggetti
 - 7.1.2. Progettazione delle lezioni
 - 7.1.3. Introduzione a UML per la modellazione dei problemi
- 7.2. Relazioni tra lezioni
 - 7.2.1. Astrazione ed ereditarietà
 - 7.2.2. Concetti avanzati di ereditarietà
 - 7.2.3. Polimorfismo
 - 7.2.4. Composizione e aggregazione

- 7.3. Introduzione ai design pattern per i problemi orientati agli oggetti
 - 7.3.1. Cosa sono i design pattern
 - 7.3.2. Pattern *Factory*
 - 7.3.4. Pattern *Singleton*
 - 7.3.5. Pattern *Observer*
 - 7.3.6. Pattern *Composite*
- 7.4. Eccezioni
 - 7.4.1. Quali sono le eccezioni?
 - 7.4.2. Gestione e acquisizione delle eccezioni
 - 7.4.3. Avvio delle eccezioni
 - 7.4.4. Creazione di eccezioni
- 7.5. Interfacce utente
 - 7.5.1. Introduzione a Qt
 - 7.5.2. Posizionamento
 - 7.5.3. Cosa sono gli eventi?
 - 7.5.4. Eventi: Definizione e cattura
 - 7.5.5. Sviluppo di interfacce utente
- 7.6. Introduzione alla programmazione concorrente
 - 7.6.1. Introduzione alla programmazione concorrente
 - 7.6.2. Il concetto di processo e di thread
 - 7.6.3. Interazione tra processi o thread
 - 7.6.4. Thread in C++
 - 7.6.5. Vantaggi e svantaggi della programmazione concorrente
- 7.7. Gestione e sincronizzazione dei thread
 - 7.7.1. Ciclo di vita di un thread
 - 7.7.2. La classe *Thread*
 - 7.7.3. Pianificazione del thread
 - 7.7.4. Gruppi di thread
 - 7.7.5. Thread di tipo demoniaco
 - 7.7.6. Sincronizzazione
 - 7.7.7. Meccanismi di bloccaggio
 - 7.7.8. Meccanismi di comunicazione
 - 7.7.9. Monitor

- 7.8. Problemi comuni nella programmazione concorrente
 - 7.8.1. Il problema dei produttori-consumatori
 - 7.8.2. Il problema dei lettori e degli scrittori
 - 7.8.3. Il problema della cena dei filosofi
- 7.9. Documentazione e test del software
 - 7.9.1. Perché è importante documentare il software?
 - 7.9.2. Documentazione di progettazione
 - 7.9.3. Utilizzo di strumenti per la documentazione
- 7.10. Test di software
 - 7.10.1. Introduzione al test del software
 - 7.10.2. Tipi di test
 - 7.10.3. Test dell'unità
 - 7.10.4. Test di integrità
 - 7.10.5. Test di convalida
 - 7.10.6. Test del sistema

Modulo 8. Riutilizzo di software

- 8.1. Panoramica del riutilizzo di software
 - 8.1.1. In che consiste il riutilizzo di software
 - 8.1.2. Vantaggi e svantaggi del riutilizzo di software
 - 8.1.3. Principali tecniche di riutilizzo di software
- 8.2. Introduzione ai modelli di progettazione
 - 8.2.1. Che cos'è un modello di progettazione?
 - 8.2.2. Catalogo dei principali modelli di progettazione
 - 8.2.3. Come utilizzare i modelli per risolvere i problemi di progettazione
 - 8.2.4. Come selezionare il miglior modello di progettazione
- 8.3. Pattern di creazione (I)
 - 8.3.1. Pattern di creazione
 - 8.3.2. Pattern *Abstract Factory*
 - 8.3.3. Esempio di implementazione del Pattern *Abstract Factory*
 - 8.3.4. Pattern *Builder*
 - 8.3.5. Esempio di implementazione del metodo *Builder*
 - 8.3.6. Pattern *Abstract Factory* vs *Builder*
- 8.4. Pattern di creazione (II)
 - 8.4.1. Pattern *Factory Method*
 - 8.4.2. *Factory Method* vs *Abstract Factory*
 - 8.4.3. Pattern *Singleton*
- 8.5. Pattern strutturali (I)
 - 8.5.1. Pattern strutturali
 - 8.5.2. Pattern *Adapter*
 - 8.5.3. Pattern *Bridge*
- 8.6. Pattern strutturali (II)
 - 8.6.1. Pattern *Composite*
 - 8.6.2. Pattern *Decorador*
- 8.7. Pattern strutturali (III)
 - 8.7.1. Pattern *Facade*
 - 8.7.2. Pattern *Proxy*
- 8.8. Pattern di comportamento (I)
 - 8.8.1. Concetto di pattern di comportamento
 - 8.8.2. Pattern di comportamento: Catena di responsabilità
 - 8.8.3. Pattern di comportamento Ordine
- 8.9. Pattern di comportamento (II)
 - 8.9.1. Pattern *Interprete* o *Interpreter*
 - 8.9.2. Pattern di iterazione
 - 8.9.3. Pattern di osservazione
 - 8.9.4. Pattern di strategia
- 8.10. *Framework*
 - 8.10.1. Concetto di *Framework*
 - 8.10.2. Sviluppo mediante *Framework*
 - 8.10.3. Pattern *Model View Controller*
 - 8.10.4. *Framework* per la progettazione di interfacce grafiche
 - 8.10.5. *Framework* per lo sviluppo di applicazioni web
 - 8.10.6. *Framework* per la gestione della persistenza degli oggetti nei database

Modulo 9. Intelligenza Artificiale e Ingegneria della Conoscenza

- 9.1. Introduzione all'intelligenza artificiale e all'ingegneria della conoscenza
 - 9.1.1. Breve storia dell'intelligenza artificiale
 - 9.1.2. L'intelligenza Artificiale oggi
 - 9.1.3. Ingegneria della Conoscenza
- 9.2. Ricerca
 - 9.2.1. Concetti di ricerca comuni
 - 9.2.2. Ricerca non informata
 - 9.2.3. Ricerca informata
- 9.3. Soddisfacibilità booleana, soddisfacibilità dei vincoli e pianificazione automatica
 - 9.3.1. Soddisfacibilità booleana
 - 9.3.2. Problemi di soddisfazione dei vincoli
 - 9.3.3. Pianificazione automatica e PDDL
 - 9.3.4. Pianificazione come ricerca Euristica
 - 9.3.5. Pianificazione con il SAT
- 9.4. Intelligenza artificiale nei giochi
 - 9.4.1. Teoria dei giochi
 - 9.4.2. Potenziamiento Minimax e Alfa-Beta
 - 9.4.3. Simulazione: Monte Carlo
- 9.5. Apprendimento supervisionato e non
 - 9.5.1. Introduzione all'Apprendimento Automatico
 - 9.5.2. Classificazione
 - 9.5.3. Regressione
 - 9.5.4. Convalida dei risultati
 - 9.5.5. Raggruppamento (*Clustering*)
- 9.6. Reti neurali
 - 9.6.1. Basi Biologiche
 - 9.6.2. Modello Computazionale
 - 9.6.3. Reti Neurali Supervisionate e non
 - 9.6.4. Percettrone Semplice
 - 9.6.5. Percettrone Multistrato

- 9.7. Algoritmi genetici
 - 9.7.1. Storia
 - 9.7.2. Base biologica
 - 9.7.3. Codifica dei problemi
 - 9.7.4. Generazione della popolazione iniziale
 - 9.7.5. Algoritmo principale e operatori genetici
 - 9.7.6. Valutazione degli individui: *Fitness*
- 9.8. Thesauri, vocabolari, tassonomie
 - 9.8.1. Vocabolari
 - 9.8.2. Tassonomie
 - 9.8.3. Thesauri
 - 9.8.4. Ontologie
- 9.9. Rappresentazione della conoscenza: Web semantico
 - 9.9.1. Web semantico
 - 9.9.2. Specifiche: RDF, RDFS e OWL
 - 9.9.3. Inferenza/ragionamento
 - 9.9.4. *Linked Data*
- 9.10. Sistemi esperti e DSS
 - 9.10.1. Sistemi esperti
 - 9.10.2. Sistemi di supporto decisionale

Modulo 10. Ingegneria del Software Avanzato

- 10.1. Introduzione alle Metodologie Agili
 - 10.1.1. Modelli di processo e metodologie
 - 10.1.2. Agilità e processi agili
 - 10.1.3. Manifesto Agile
 - 10.1.4. Alcune metodologie agili
 - 10.1.5. Agile vs tradizionale

- 10.2. *Scrum*
 - 10.2.1. Origini e filosofia di *Scrum*
 - 10.2.2. Valori di *Scrum*
 - 10.2.3. Flusso del processo *Scrum*
 - 10.2.4. Ruoli di *Scrum*
 - 10.2.5. Artefatti di *Scrum*
 - 10.2.6. Eventi di *Scrum*
 - 10.2.7. Storie degli utenti
 - 10.2.8. Estensioni *Scrum*
 - 10.2.9. Stime agili
 - 10.2.10. Scalabilità di *Scrum*
- 10.3. Programmazione estrema
 - 10.3.1. Motivazione e panoramica di XP
 - 10.3.2. Il ciclo di vita in XP
 - 10.3.3. I cinque valori fondamentali
 - 10.3.4. Le dodici pratiche di base di XP
 - 10.3.5. Ruoli dei partecipanti
 - 10.3.6. XP industriale
 - 10.3.7. Valutazione critica di XP
- 10.4. Sviluppo del software basato sul riutilizzo
 - 10.4.1. Riutilizzo del software
 - 10.4.2. Livelli di riutilizzo del codice
 - 10.4.3. Tecniche di riutilizzo specifiche
 - 10.4.4. Sviluppo basato su componenti
 - 10.4.5. Benefici e problemi del riutilizzo
 - 10.4.6. Pianificazione del riutilizzo
- 10.5. Architettura di sistema e modelli di progettazione del software
 - 10.5.1. Design architettonico
 - 10.5.2. Modelli architettonici generali
 - 10.5.3. Architetture con tolleranza ai guasti
 - 10.5.4. Architetture di sistemi distribuiti
 - 10.5.5. Pattern di progettazione
 - 10.5.6. Pattern gamma
 - 10.5.7. Pattern di progettazione dell'interazione
- 10.6. Architettura di applicazioni nel cloud
 - 10.6.1. Fondamenti di *Cloud Computing*
 - 10.6.2. Qualità delle applicazioni nel cloud
 - 10.6.3. Stili di architettura
 - 10.6.4. Pattern di progettazione
- 10.7. Test del software: TDD, ATDD e BDD
 - 10.7.1. Verifica e convalida del software
 - 10.7.2. Test del software
 - 10.7.3. *Test Driven Development* (TDD)
 - 10.7.4. *Acceptance Test Driven Development* (ATDD)
 - 10.7.5. *Behavior Driven Development* (BDD)
 - 10.7.6. BDD e *Cucumber*
- 10.8. Miglioramento dei processi software
 - 10.8.1. Miglioramento dei processi software
 - 10.8.2. Il miglioramento dei processi
 - 10.8.3. Modelli di maturità
 - 10.8.4. Il modello CMMI
 - 10.8.5. CMMI V2.0
 - 10.8.6. CMMI e Agile
- 10.9. La qualità del prodotto software: SQuaRE
 - 10.9.1. La qualità del software
 - 10.9.2. Modello di qualità del prodotto software
 - 10.9.3. Famiglia ISO/IEC 25000
 - 10.9.4. ISO/IEC 25010: Modello e caratteristiche di qualità
 - 10.9.5. ISO/IEC 25012: La qualità dei dati
 - 10.9.6. ISO/IEC 25020: Misurazione della qualità del software
 - 10.9.7. ISO/IEC 25022, 25023 e 25024: Parametri di qualità del software e dei dati
 - 10.9.8. ISO/IEC 25040: Valutazione del software
 - 10.9.9. Processo di certificazione
- 10.10. Introduzione a DevOps
 - 10.10.1. Concetto di DevOps
 - 10.10.2. Pratiche principali

06 Tirocinio

Al termine del percorso teorico online, il programma prevede un periodo di Tirocinio presso un'azienda informatica di riferimento. Lo studente disporrà del supporto di un tutor personale che lo accompagnerà durante tutto il percorso, sia nella preparazione che nello svolgimento del tirocinio.





“

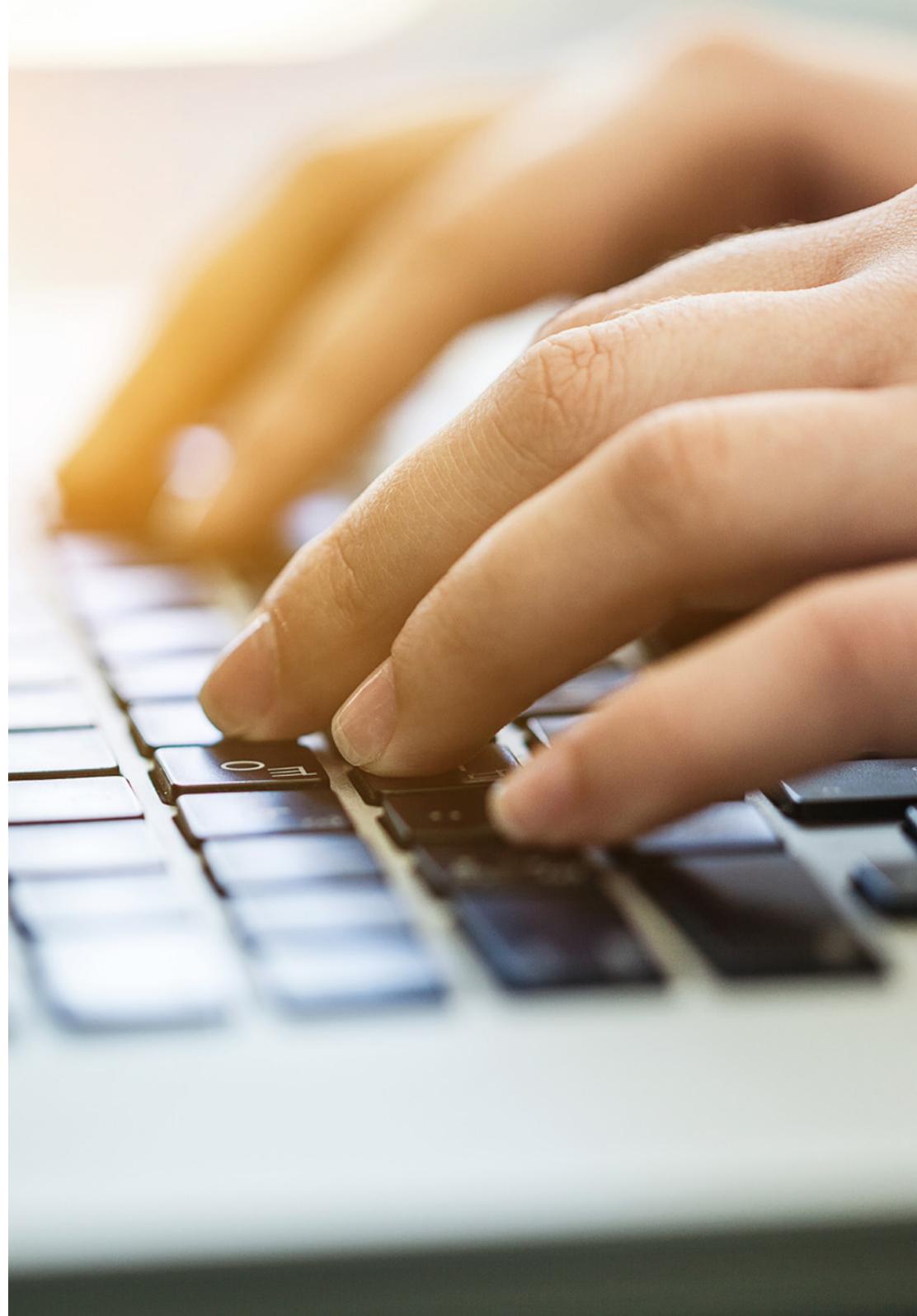
Parteciperai a progetti pratici che richiedono l'applicazione dei concetti teorici appresi, permettendoti di sviluppare soluzioni innovative e funzionali”

Il periodo di Tirocinio di questo programma in Creazione di Interfacce e Applicazioni in Rete è costituito da uno stage presso una prestigiosa azienda informatica, di 3 settimane, dal lunedì al venerdì e con giornate di 8 ore consecutive di formazione pratica accanto a uno specialista. Durante questo tirocinio, lo studente sarà in grado di applicare le conoscenze teoriche acquisite a situazioni reali, consolidando la loro comprensione e abilità nel design e nello sviluppo di interfacce e applicazioni web.

Questa proposta di formazione, completamente pratica, mira a coinvolgere gli studenti in progetti reali di sviluppo di applicazioni web, dalla fase di concettualizzazione alla loro implementazione e diffusione. Ciò consentirà di affrontare sfide tecniche e creative, oltre a imparare a lavorare in team, gestire le risorse e rispettare scadenze impegnative.

Inoltre, si tratta di una grande opportunità per imparare lavorando, sperimentando vari strumenti e tecnologie all'avanguardia, preparando i professionisti ad affrontare le sfide del mondo del lavoro in continua evoluzione. Il tutto lavorando in un ambiente collaborativo con esperti del settore, fornendo una visione più ampia e aggiornata delle tendenze e delle pratiche emergenti nel campo della tecnologia dell'informazione e della comunicazione.

La fase pratica prevede la partecipazione attiva dello studente che svolgerà le attività e le procedure di ogni area di competenza (imparare a imparare e imparare a fare), con l'accompagnamento e la guida del personale docente e degli altri compagni di corso che facilitano il lavoro di squadra e l'integrazione multidisciplinare come competenze trasversali per la Creazione di Interfacce e Applicazioni in Rete (imparare a essere e imparare a relazionarsi).



Le procedure descritte di seguito costituiranno la base della parte pratica del corso e la loro attuazione sarà soggetta alla disponibilità e al carico di lavoro del centro stesso; le attività proposte sono le seguenti:

| Modulo | Attività Pratica |
|---------------------------------------|---|
| Progettazione dell'interfaccia | Progettare interfacce utente intuitive e coinvolgenti |
| | Eseguire test di usabilità per migliorare l'esperienza utente |
| | Implementare misure di sicurezza per proteggere le sue informazioni sensibile |
| | Implementare sistemi di analisi dei dati per migliorare il processo decisionale |
| Applicazioni in rete | Sviluppare applicazioni web reattive e ad alte prestazioni |
| | Integrare i sistemi di gestione dei contenuti (CMS) per facilitare la gestione di contenuti |
| | Ottimizzare la velocità e l'efficienza delle applicazioni di rete |
| | Collaborare alla strategia di Digital Marketing per promuovere le applicazioni di rete |
| Accessibilità | Adattare le interfacce e le applicazioni per diversi dispositivi e piattaforme |
| | Incorporare tecniche di accessibilità per garantire l'inclusione di tutti gli utenti |
| | Consentire agli utenti finali di utilizzare le applicazioni sviluppate |
| | Adattare interfacce e applicazioni a diverse lingue e culture |

| Modulo | Attività Pratica |
|--------------------------------------|--|
| Documentazione e manutenzione | Ricerare e risolvere problemi tecnici relativi all'interfaccia e all'applicazione |
| | Documentare il processo di sviluppo e mantenere aggiornata la documentazione tecnica |
| | Gestire l'intero ciclo di vita delle applicazioni, dallo sviluppo alla manutenzione e gli aggiornamenti |
| | Rimanere aggiornati su normative e regolamenti correlati con la privacy e la sicurezza dei dati |
| Ricerca e collaborazione | Rimanere al passo con le ultime tendenze e tecnologie nel web design e sviluppo di applicazioni |
| | Collaborare con team multidisciplinari per garantire la coerenza tra progettazione e funzionalità |
| | Creare prototipi ed eseguire test di concetto per nuove funzionalità |
| | Analizzare i dati degli utenti per identificare le aree di miglioramento dell'interfaccia e delle funzionalità |



Lavorerai a stretto contatto con professionisti esperti del settore, comprendendo i principi fondamentali della progettazione di interfacce e dello sviluppo di applicazioni web”

Assicurazione di responsabilità civile

La preoccupazione principale di questa istituzione è quella di garantire la sicurezza sia dei tirocinanti e degli altri agenti che collaborano ai processi di tirocinio in azienda. All'interno delle misure rivolte a questo fine ultimo, esiste la risposta a qualsiasi incidente che possa verificarsi durante il processo di insegnamento-apprendimento.

A tal fine, questa istituzione educativa si impegna a stipulare un'assicurazione di responsabilità civile per coprire qualsiasi eventualità che possa insorgere durante la permanenza presso il centro di tirocinio.

La polizza di responsabilità civile per i tirocinanti deve garantire una copertura assicurativa completa e deve essere stipulata prima dell'inizio del periodo di tirocinio. Grazie a questa garanzia, il professionista si sentirà privo di ogni tipo di preoccupazione nel caso di eventuali situazioni impreviste che possano sorgere durante il tirocinio e potrà godere di una copertura assicurativa fino al termine dello stesso.



Condizioni generali del tirocinio

Le condizioni generali dell'accordo di tirocinio per il programma sono le seguenti:

1. TUTORAGGIO: durante il Master Semipresenziale agli studenti verranno assegnati due tutor che li seguiranno durante tutto il percorso, risolvendo eventuali dubbi e domande. Da un lato, lo studente disporrà di un tutor professionale appartenente al centro di inserimento lavorativo che lo guiderà e lo supporterà in ogni momento. Dall'altro lato, allo studente verrà assegnato anche un tutor accademico che avrà il compito di coordinare e aiutare lo studente durante l'intero processo, risolvendo i dubbi e fornendogli tutto ciò di cui potrebbe aver bisogno. In questo modo, il professionista sarà accompagnato in ogni momento e potrà risolvere tutti gli eventuali dubbi, sia di natura pratica che accademica.

2. DURATA: il programma del tirocinio avrà una durata di tre settimane consecutive di preparazione pratica, distribuite in giornate di 8 ore lavorative, per cinque giorni alla settimana. I giorni di frequenza e l'orario saranno di competenza del centro, che informerà debitamente e preventivamente il professionista, con un sufficiente anticipo per facilitarne l'organizzazione.

3. MANCATA PRESENTAZIONE: in caso di mancata presentazione il giorno di inizio del Master Semipresenziale, lo studente perderà il diritto allo stesso senza possibilità di rimborso o di modifica di date. L'assenza per più di due giorni senza un giustificato motivo/certificato medico comporterà la rinuncia dello studente al tirocinio e, pertanto, la relativa automatica cessazione. In caso di ulteriori problemi durante lo svolgimento del tirocinio, essi dovranno essere debitamente e urgentemente segnalati al tutor accademico.

4. CERTIFICAZIONE: lo studente che supererà il Master Semipresenziale riceverà un certificato che attesterà il tirocinio svolto presso il centro in questione.

5. RAPPORTO DI LAVORO: il Master Semipresenziale non costituisce alcun tipo di rapporto lavorativo.

6. STUDI PRECEDENTI: alcuni centri potranno richiedere un certificato di studi precedenti per la partecipazione al Master Semipresenziale. In tal caso, sarà necessario esibirlo al dipartimento tirocini di TECH affinché venga confermata l'assegnazione del centro prescelto.

7. NON INCLUDE: il Master Semipresenziale non includerà nessun elemento non menzionato all'interno delle presenti condizioni. Pertanto, non sono inclusi alloggio, trasporto verso la città in cui si svolge il tirocinio, visti o qualsiasi altro servizio non menzionato.

Tuttavia, gli studenti potranno consultare il proprio tutor accademico per qualsiasi dubbio o raccomandazione in merito. Egli fornirà tutte le informazioni necessarie per semplificare le procedure.

07

Dove posso svolgere il Tirocinio?

Questo programma di Master Semipresenziale include un tirocinio presso un prestigioso un'azienda informatica, dove gli studenti metteranno in pratica tutto ciò che hanno imparato in materia di Creazione di Interfacce e Applicazioni in Rete. In questo senso, e per avvicinare questo titolo di studio a un maggior numero di professionisti, TECH offre l'opportunità di studiarlo in centri diversi e prestigiosi. In questo modo, l'istituzione rafforza il suo impegno per un'istruzione di qualità e a prezzi accessibili per tutti.



“

Completa la tua preparazione teorica grazie al miglior tirocinio del mercato. Raggiungi il successo nella tua carriera in modo agile e semplice grazie a TECH!”

tech 42 | Dove posso svolgere il Tirocinio?



Gli studenti potranno svolgere il tirocinio di questo Master Semipresenziale presso i seguenti centri:



Informatica

Ogilvy Barcelona

| | |
|--------|------------|
| Paese | Città |
| Spagna | Barcellona |

Indirizzo: Calle Bolivia 68-70, 08018, Barcelona

Ogilvy è pioniera nella pubblicità onnipresente, marketing e comunicazione aziendale

Tirocini correlati:

- Intelligenza Artificiale nel Design
- Costruzione del Marchio Personale





“

Potenzia la tua carriera professionale con un insegnamento olistico, che ti consenta di progredire sia dal punto di vista teorico che pratico”

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

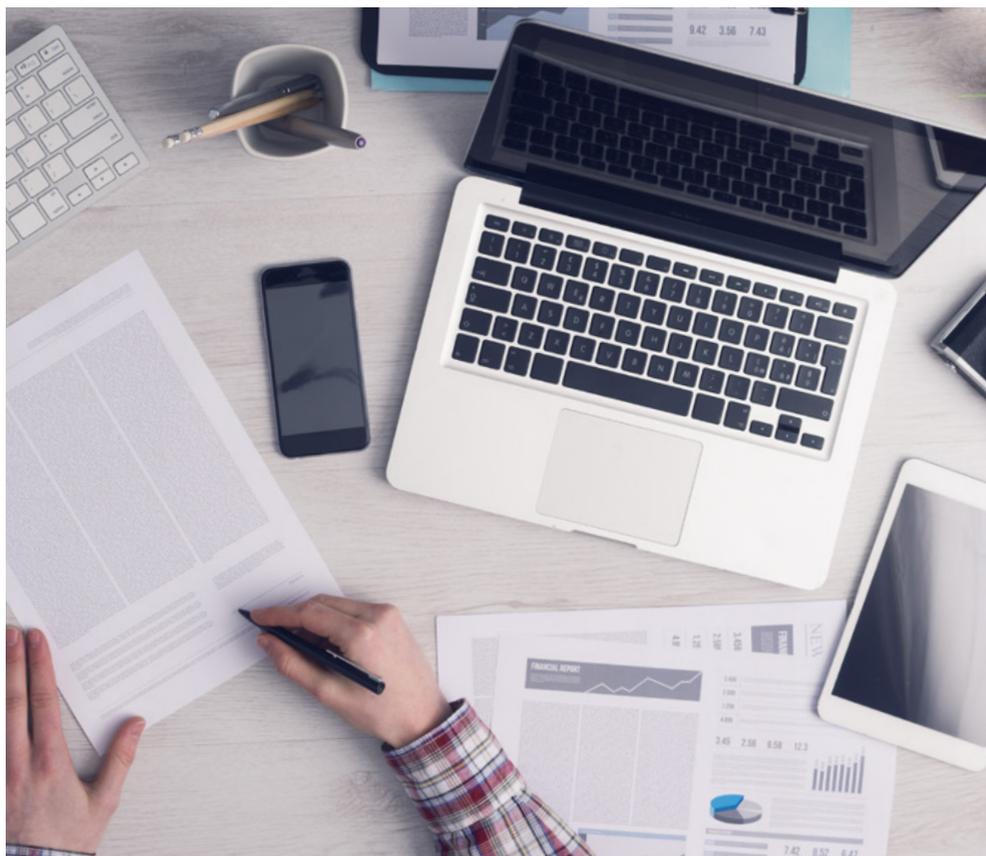
Il nostro programma offre un metodo rivoluzionario per sviluppare le abilità e le conoscenze. Il nostro obiettivo è quello di rafforzare le competenze in un contesto mutevole, competitivo e altamente esigente.

“

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.



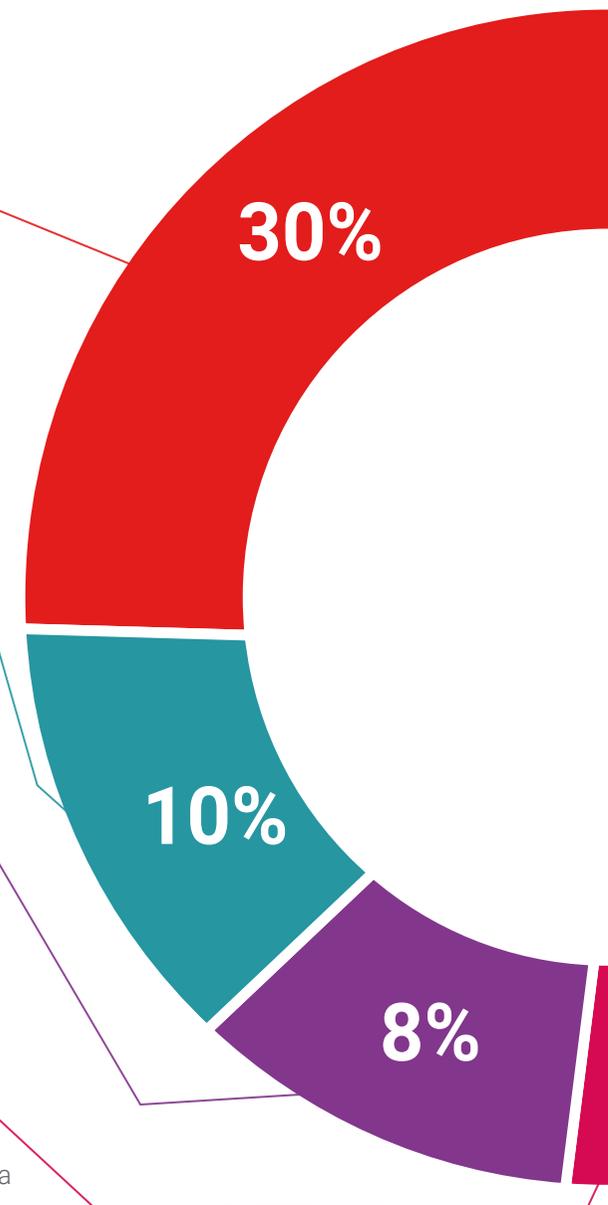
Pratiche di competenze e competenze

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.



09

Titolo

Il titolo di Master Semipresenziale in Creazione di Interfacce e Applicazioni in Rete garantisce, oltre alla specializzazione più rigorosa e aggiornata, l'accesso ad una qualifica di Master Semipresenziale rilasciata da TECH Università Tecnologica



“

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

Questo **Master Semipresenziale in Creazione di Interfacce e Applicazioni in Rete** possiede il programma più completo e aggiornato del panorama professionale e accademico.

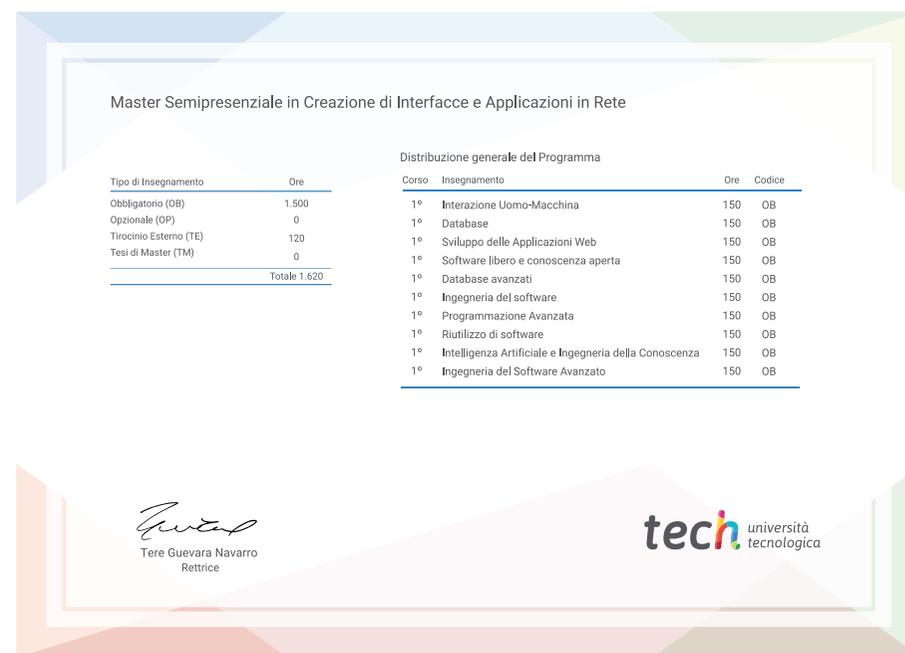
Dopo aver superato le valutazioni, lo studente riceverà mediante lettera certificata, con ricevuta di ritorno, la corrispondente qualifica di **Master Semipresenziale** rilasciata da **TECH Università Tecnologica**, che accrediterà il superamento delle valutazioni e l'acquisizione delle competenze del programma.

Oltre alla qualifica, sarà possibile ottenere un certificato e un attestato dei contenuti del programma. A tal fine, sarà necessario contattare il proprio consulente accademico, che fornirà tutte le informazioni necessarie.

Titolo: **Master Semipresenziale in Creazione di Interfacce e Applicazioni in Rete**

Modalità: **Semipresenziale (Online + Tirocinio)**

Durata: **12 mesi**



*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 lingu

tech  universidad
tecnológica

Master Semipresenziale Creazione di Interfacce e Applicazioni in Rete

Modalità: Semipresenziale (Online + Tirocinio)

Durata: 12 mesi

Titolo: TECH Università Tecnologica

Ore teoriche: 1.620

Master Semipresenziale

Creazione di Interfacce e Applicazioni in Rete