

Master Privato

Sviluppo di Applicazioni Android



Master Privato Sviluppo di Applicazioni Android

- » Modalità: online
- » Durata: 12 mesi
- » Titolo: TECH Università Tecnologica
- » Dedizione: 16 ore/settimana
- » Orario: a scelta
- » Esami: online

Accesso al sito web: www.techitute.com/it/informatica/master/master-sviluppo-applicazioni-android

Indice

01

Presentazione

pag. 4

02

Obiettivi

pag. 8

03

Competenze

pag. 14

04

Direzione del corso

pag. 18

05

Struttura e contenuti

pag. 24

06

Metodologia

pag. 36

07

Titolo

pag. 44

01

Presentazione

Attualmente ci sono più di 3 miliardi di dispositivi Android nel mondo ed il settore dello Sviluppo di Applicazioni mobili ha registrato una crescita vertiginosa. I rapporti sul mercato del lavoro confermano la crescente domanda di profili esperti nella progettazione di applicazioni mobili, valorizzando le competenze nell'intero ciclo di vita dello sviluppo, della distribuzione e della monetizzazione dell'applicazione. Non è semplice trovare professionisti che conoscano il linguaggio di programmazione Android, le architetture e le interfacce utente. Di conseguenza, questo programma è stato concepito per fornire agli interessati tutte le conoscenze necessarie per dare slancio alla loro carriera grazie ad un profilo specializzato. In modalità completamente online e conseguendo la qualifica in pochi mesi.





“

Specializzati negli aspetti chiave della Customer Experience per sviluppare applicazioni Android in modo da avere maggiori opportunità nel mercato”

Iscrivendosi al Master Privato nello Sviluppo di Applicazioni Android, lo studente potrà specializzarsi in una delle tecnologie standard di maggiore prospettiva professionale di questi tempi. Acquisirà le conoscenze necessarie per essere autonomo nello sviluppo di un'applicazione per dispositivi che utilizzano questo sistema operativo, dalla progettazione alla produzione. Acquisirà inoltre conoscenze specialistiche sul ciclo di vita delle applicazioni Android, in modo da poter lavorare più efficientemente.

A questo scopo, TECH dispone di un team di professionisti altamente qualificati che possiedono l'esperienza necessaria per condividere casi reali che portano a un processo di apprendimento più profondo ed efficiente. Professionisti della programmazione, dello sviluppo di applicazioni e delle nuove tecnologie che accompagneranno il professionista nel processo di studio. Da una piattaforma sicura, con contenuti interattivi, distribuiti in video, grafici, esempi basati su problemi, materiale audiovisivo e con un'ampia comunità per discutere le questioni sollevate.

Pertanto, lo studente potrebbe lavorare in aree quali l'informatica, lo sviluppo dei software, lo sviluppo aziendale, il Marketing e le vendite, la consulenza, la tecnologia mobile multiplatforma, l'automotive, la domotica, l'Internet delle Cose (IoT), le banche e i droni. Sarà in grado di applicare le sue conoscenze a beneficio di uno qualsiasi di questi settori.

TECH avrà inoltre a disposizione la miglior metodologia di studio e il *Relearning*, completamente online, che elimina la necessità di frequentare le lezioni di persona e seguire un orario prestabilito. In questo modo, lo studente acquisirà le basi necessarie per intraprendere attività di consulenza in tutti gli aspetti legati alla tecnologia mobile su Android o per sviluppare una propria attività basata sullo sviluppo di applicazioni.

Questo **Master Privato in Sviluppo di Applicazioni Android** possiede il programma più completo e aggiornato del mercato. Le caratteristiche principali del programma sono:

- ◆ Lo sviluppo di casi di studio presentati da esperti di calcolo parallelo e distribuito.
- ◆ I contenuti grafici, schematici e eminentemente pratici del programma forniscono informazioni pratiche su quelle discipline che sono essenziali per l'attività professionale
- ◆ Esercizi pratici che offrono un processo di autovalutazione per migliorare l'apprendimento
- ◆ Speciale enfasi sulle metodologie innovative
- ◆ Lezioni teoriche, domande all'esperto e lavori di riflessione individuale
- ◆ Contenuti disponibili da qualsiasi dispositivo fisso o mobile dotato di connessione a internet



Con il Master Privato nello Sviluppo di Applicazioni Android potrai specializzarti in una delle tecnologie standard di maggiore prospettiva professionale al giorno d'oggi"

“

Acquisirai le conoscenze necessarie per essere autonomo nello sviluppo di un'applicazione per dispositivi Android, dalla progettazione alla produzione”

Il personale docente del programma comprende rinomati specialisti del settore 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 lo specialista deve cercare di risolvere le diverse situazioni che gli si presentano durante il corso. Lo studente potrà usufruire di un innovativo sistema di video interattivi creati da esperti di rinomata fama.

Approfitta dei vantaggi dell'ambiente tecnologico all'avanguardia che TECH mette a tua disposizione. Un programma 100% online che ti permetterà di entrare in contatto con i maggiori esperti del settore.

Fornisci soluzioni efficienti ai tuoi datori di lavoro o alla tua azienda. Con una buona padronanza dei fondamenti legati allo Sviluppo di Applicazioni per i sistemi Android.

Android Studio
Version 3.3

Android Studio project

02 Obiettivi

Il processo di crescita determinato dall'inarrestabile sviluppo di nuove tecnologie e dispositivi è ormai risaputo. Progressi che richiedono una continua preparazione dei professionisti dell'informatica e dell'ingegneria. Il mercato non si ferma e per stare al passo con le esigenze è essenziale che i professionisti siano all'avanguardia con i concetti e gli stili che vengono utilizzati. Questo Master Privato in Sviluppo di Applicazioni Android mette a disposizione tutti gli strumenti e i fondamenti necessari per sviluppare questa carriera e fornire soluzioni efficaci nell'ambiente professionale.



“

Stabilisci le tue sfide all'interno dell'ecosistema tecnologico Android. Inizia a muovere i primi passi verso una carriera di successo come sviluppatore di applicazioni”



Obiettivi generali

- ◆ Determinare gli elementi strutturali di un sistema Android
- ◆ Analizzare le differenze tra i diversi *Frameworks* di Sviluppo, i punti di forza e di debolezza
- ◆ Impostare la creazione di stili e temi, compresa la Realtà Virtuale, attraverso i *Layouts*
- ◆ Analizzare le diverse forme di gestione del flusso di codice in Kotlin
- ◆ Sviluppare capacità avanzate e best practice nello sviluppo di applicazioni con il linguaggio di programmazione Kotlin
- ◆ Sviluppo di una metodologia per la gestione ottimale dei dati nel dispositivo
- ◆ Analizzare i casi d'uso dei dispositivi Android sul mercato
- ◆ Padroneggiare gli elementi di un design responsivo e superare le sfide che esso presenta
- ◆ Riunire le diverse fasi di un ciclo di integrazione continua incentrato sullo sviluppo di Android

“

Diventa un esperto nello Sviluppo di Applicazioni Android. Iscriviti ora a questo programma e specializzati con questa modalità al 100% online”





Obiettivi specifici

Modulo 1. Linguaggio di programmazione Android

- ◆ Esaminare Linux e la macchina virtuale sulla base di Android
- ◆ Analizzare le librerie native del sistema
- ◆ Stabilire i punti di forza di Android rispetto alle altre piattaforme
- ◆ Determinare gli elementi di un'applicazione Android
- ◆ Presentare le versioni di Android e i loro miglioramenti
- ◆ Valutare il mercato delle applicazioni Android
- ◆ Gettare le basi per la futura evoluzione di Android

Modulo 2. *Frameworks* utilizzati nello sviluppo di applicazioni Android

- ◆ Analizzare il *Framework* di Android Core
- ◆ Sviluppare altri *Framework* utilizzati per lo sviluppo di applicazioni per Android
- ◆ Implementare le librerie con *Graddl*
- ◆ Stabilire i *Framework* per connettersi a un'API
- ◆ Generare conoscenze specialistiche su architetture quali MVP
- ◆ Chiarire i pro e i contro di MVP e MVVM

Modulo 3. Interfacce e *Layout* nello sviluppo di applicazioni Android

- ◆ Presentare il modello del ciclo di vita della vista in Android
- ◆ Esaminare gli attributi più importanti di un design visivo (*Layouts*)
- ◆ Analizzare i (*Layouts*) disponibili
- ◆ Generare un design (*Layouts*) riutilizzabile
- ◆ Determinare come utilizzare risorse alternative
- ◆ Identificare le differenze nell'uso di questi componenti rispetto ad altri sistemi di programmazione
- ◆ Stabilire le potenzialità e l'uso del file `AndroidManifest.xml`

Modulo 4. Linguaggio di programmazione nelle applicazioni Android. Kotlin

- ◆ Sviluppare il linguaggio di programmazione Kotlin
- ◆ Riunire le caratteristiche e le capacità differenziali del linguaggio Kotlin
- ◆ Esaminare il modello di esecuzione di base di un programma in Kotlin
- ◆ Analizzare la sintassi del linguaggio e la struttura di un programma in Kotlin
- ◆ Concretizzare il modello di tipi e variabili in Kotlin
- ◆ Stabilire le diverse forme di gestione del flusso di codice in Kotlin
- ◆ Determinare il modello di classi, collezioni e oggetti in Kotlin
- ◆ Generare conoscenze specialistiche sul modello di ereditarietà in Kotlin
- ◆ Realizzare il modello di gestione delle eccezioni e dei tipi di null in Kotlin

Modulo 5. Linguaggio di programmazione nelle applicazioni Android. Genericità avanzata di Kotlin, programmazione funzionale e parallelismo

- ◆ Esaminare il modello di genericità covariante e controvariante in Kotlin
- ◆ Analizzare il modello di programmazione funzionale con i lambda di Kotlin
- ◆ Realizzare le funzioni di ordine superiore di Kotlin
- ◆ Sviluppare le estensioni e gli oggetti companion in Kotlin
- ◆ Esaminare l'uso del *Delegation* pattern in Kotlin
- ◆ Raccogliere le annotazioni e le riflessioni in Kotlin
- ◆ Approfondire il modello di *Testing* in Kotlin
- ◆ Stabilire i diversi modelli di programmazione asincrona disponibili in Kotlin
- ◆ Determinare il modello di coroutine di Kotlin
- ◆ Riunire le diverse librerie e gli strumenti di utilità dell'ecosistema Kotlin

Modulo 6. Gestione dei dati nei dispositivi Android

- ◆ Analizzare le diverse tecniche di gestione dei dati in Android
- ◆ Proporre metodi di utilizzo ottimale dei dati nel dispositivo
- ◆ Identificare gli strumenti necessari per l'ottimizzazione dei dati
- ◆ Esaminare le caratteristiche di JSON e XML per la gestione dei dati in Android
- ◆ Valutare le problematiche generali dei sistemi distribuiti applicabili al mondo delle applicazioni per dispositivi mobili
- ◆ Determinare l'uso della libreria Room come astrazione per l'utilizzo di SQLite su Android e i suoi vantaggi e svantaggi
- ◆ Impostare le necessarie autorizzazioni di sicurezza sulla gestione dei dati in qualsiasi tecnica utilizzata in Android

Modulo 7. Strumenti dei dispositivi Android

- ◆ Riunire gli strumenti più avanzati per la gestione quotidiana
- ◆ Valutare gli strumenti di controllo dei dispositivi Android
- ◆ Dimostrare i vantaggi dell'utilizzo di Android nei Droni
- ◆ Concretare le funzionalità degli strumenti CRM
- ◆ Dimostrare i vantaggi dei dispositivi Android nelle piattaforme bancarie
- ◆ Analizzare le possibilità della piattaforma IoT e di Android
- ◆ Esaminare gli strumenti di efficienza dei processi su Android

Modulo 8. Design *Responsivo* in Android

- ◆ Analizzare gli elementi principali di un progetto
- ◆ Definire una metodologia di progettazione visiva e una metodologia di progettazione degli schermi
- ◆ Risolvere i vari problemi incontrati nello sviluppo multidispositivo
- ◆ Disporre di strumenti per generare risorse migliori e più numerose per lo sviluppo multidispositivo
- ◆ Esaminare *Framework* alternativi allo sviluppo responsivo nativo
- ◆ Generare una metodologia propria per lo sviluppo delle applicazioni utilizzando le migliori pratiche per la visualizzazione multidispositivo fin dall'inizio del progetto

Modulo 9. Marketing delle applicazioni Android

- ◆ Analizzare le nuove metodologie incentrate sull'utente
- ◆ Scoprire come l'intelligenza artificiale ha portato la CX a un livello superiore
- ◆ Stabilire l'importanza dell'accessibilità e della mobilità
- ◆ Sviluppare le tecniche più avanzate di analisi di sessione e comportamentali
- ◆ Realizzare gli obiettivi di micro-personalizzazione durante il *Journey* dell'utente
- ◆ Riunire le nuove metodologie per un ambiente mutevole e vivace
- ◆ Proporre tecniche per la creazione di prototipi

Modulo 10. Cicli di vita delle applicazioni Android. Cloud, Playstore e versioning

- ◆ Realizzare i benefici dell'adozione di un modello automatizzato di distribuzione delle versioni
- ◆ Stabilire le differenze tra integrazione continua, consegna continua e distribuzione continua
- ◆ Definire le caratteristiche principali di DevOps
- ◆ Valutare alcuni degli strumenti principali per l'implementazione di pipeline CI/CD
- ◆ Identificare i fattori essenziali per lo sviluppo di applicazioni pronte a supportare i processi CI/CD
- ◆ Esaminare la Tecnologia dei Container come pilastro fondamentale del CI/CD
- ◆ Identificare pratiche, casi d'uso, tecnologie e strumenti dell'ecosistema CI/CD

03 Competenze

Non è facile trovare profili che conoscano il linguaggio di programmazione Android, le architetture e le interfacce utente, a seconda del tipo di dispositivo nel quale viene eseguita l'applicazione. Grazie al Master Privato nello Sviluppo di Applicazioni Android, lo studente potrà specializzarsi in una delle tecnologie standard di maggiore prospettiva professionale attualmente. Acquisirà le conoscenze necessarie per essere autonomo nello sviluppo di un'applicazione per dispositivi Android, dalla progettazione alla produzione. Acquisirà conoscenze specialistiche sul ciclo di vita delle applicazioni Android, in modo da poter lavorare più efficientemente come sviluppatore di applicazioni per dispositivi mobili.



“

Grazie a questo Master Privato darai uno slancio alla tua carriera in questo settore. Sviluppando applicazioni di grand successo”



Competenze generali

- ◆ Sviluppo di applicazioni mobili per dispositivi con sistema operativo Android
- ◆ Padroneggiare gli elementi strutturali di un sistema Android
- ◆ Analizzare i vari *Framework* da utilizzare per numerose architetture
- ◆ Definire le proprietà di un'interfaccia dal punto di vista dell'usabilità
- ◆ Sviluppare una conoscenza specialistica sul linguaggio di programmazione Kotlin e sul contesto in cui è nato
- ◆ Gestire il *Content Provider* per lo scambio di dati e la loro sicurezza
- ◆ Approcciarsi agli strumenti aziendali in Android: efficienza, utilità
- ◆ Analizzare gli elementi di un design responsivo
- ◆ Esplorare le nuove metodologie incentrate sull'utente
- ◆ Adattare il sistema di Pipeline automatizzati all'ecosistema Android

“

Riunire tutte queste competenze nel tuo profilo professionale aggiungerà valore alle tue proposte e avrai maggior successo nel mercato del lavoro”





Competenze specifiche

- ◆ Utilizzare risorse alternative analizzando i design (*Layouts*) disponibili
- ◆ Sviluppare le diverse forme di gestione del flusso di codice in Kotlin e le sue capacità
- ◆ Migliorare la produttività delle applicazioni grazie alle capacità differenziali del linguaggio Kotlin
- ◆ Sviluppare le estensioni e gli oggetti companion in Kotlin
- ◆ Utilizzare database o servizi di rete per creare file in diversi formati
- ◆ Gestire la libreria Room come astrazione per l'utilizzo di SQLite su Android
- ◆ Valutare gli strumenti di controllo dei dispositivi Android, analizzando le possibilità della piattaforma IoT e Android
- ◆ Adottare un design responsivo per offrire esperienze utente più soddisfacenti
- ◆ Combinare le tecniche di prototipazione e *Wireframing* con le nuove tecnologie dirompenti come l'Intelligenza Artificiale e l'Internet delle Cose

04

Direzione del corso

Questo Master Privato in Sviluppo di Applicazioni Android è diretto da docenti che possiedono una vasta conoscenza ed esperienza nelle nuove tecnologie, nell'architettura delle soluzioni e nell'infrastruttura digitale, esperti nella programmazione Android e sviluppatori di applicazioni. I loro curriculum offrono una garanzia della qualità dei contenuti selezionati per questo programma. Concentrandosi sull'ottimizzazione del processo di apprendimento per i professionisti che cercano in questo spazio il contributo necessario per il loro successo professionale.



“

Esperti professionisti nello sviluppo di applicazioni e nell'area tecnologica ti accompagneranno durante tutto il percorso di studio, fornendoti i migliori contenuti”

Direttore Ospite Internazionale

Colin Lee è uno sviluppatore di applicazioni mobili di successo, specializzato in codice nativo Android, la cui influenza si estende a livello internazionale. L'esperto è un'autorità nell'area delle Twin Cities e nella gestione di Kotlin. Uno dei suoi contributi più recenti è stato quello di dimostrare, con codice live, come costruire rapidamente un browser utilizzando il suddetto linguaggio di programmazione e i componenti open source del browser di Mozilla per Android.

Inoltre, le sue applicazioni sono state collegate a importanti aziende globali. Ad esempio, è stato responsabile della creazione di soluzioni digitali per Pearson, uno dei maggiori editori al mondo. Ha anche sviluppato un registratore video Android di basso livello per la startup Flipgrid, in seguito acquisita da Microsoft.

Ha anche costruito una VPN Android di successo per un grande cliente di consulenza. È anche il creatore di uno strumento di gestione delle merci implementato dalla transnazionale Amazon per facilitare il lavoro dei suoi camionisti a contratto. Ha anche contribuito a costruire le versioni mobili del browser Firefox per Mozilla.

Oggi lavora come appaltatore, tra cui revisioni del codice e controlli di sicurezza. Il suo impatto sullo sviluppo di applicazioni mobili e la sua esperienza nel corso degli anni lo rendono una figura di spicco nell'arena tecnologica globale.



Dott. Lee, Colin

- Direttore presso ColinTheShots LLC
- Ingegnere software Android presso Specto Inc.
- Ingegnere Senior Android presso Mozilla
- Ingegnere Sviluppatore Software presso Amazon
- Ingegnere di Applicazioni Mobili presso Flipgrid
- Specialista di Configurazione Software per Pearson VUE
- Laurea presso l'Università della Florida

“

Grazie a TECH potrai imparare con i migliori professionisti del mondo”

Direzione



Dott. Olalla Bonal, Martín

- ◆ Attuale *Blockchain Technical Specialist* presso IBM SPGI
- ◆ Tecnico in elettronica digitale
- ◆ Architetto *Blockchain*
- ◆ Architetto di Infrastrutture nel settore Bancario
- ◆ Formazione *Hyperledger Fabric* per le aziende
- ◆ Formazione *Blockchain* orientato al business aziendale
- ◆ Gestione dei progetti e implementazione delle soluzioni
- ◆ Oltre 25 anni di esperienza nel Settore IT

Personale docente

Dott. Villot Guisán, Pablo

- ◆ Architetto Cloud di Soluzioni Esponenziali e Subject Matter Expert *Blockchain* in KPMG
- ◆ Architetto Cloud di Soluzioni Esponenziali e Subject Matter Expert *Blockchain* e di Integrazione in Everis
- ◆ Sviluppatore e Responsabile Tecnico delle applicazioni web e desktop per l'area Logistica Commerciale di Inditex, Connectis
- ◆ Laureato in Ingegneria Informatica presso l'Università di La Coruña
- ◆ Certificato Microsoft in MSCA: *Cloud Platform*

Dott. Noguera Rodríguez, Pablo

- ◆ Sviluppatore di App Native (iOS e Android) - Starman Aviation (Aviaze App)
- ◆ Sviluppatore di App Native (iOS) - Stef (Mtrack App)
- ◆ Sviluppatore di App Native (iOS e Android) - Bitnovo (Bitnovo App)
- ◆ Sviluppatore Esperto in Java: JSE, JEE e Android - Ilabora Formazione
- ◆ Programmazione di Applicazione per Android- EOI - Madrid

Dott. Guerrero Díaz-Pintado, Arturo

- ◆ Consulente di servizi professionali che lavora con organizzazioni all'avanguardia in Europa, Medio Oriente e America Latina da IBM
- ◆ Celebri collaborazioni con rinomate università e centri di istruzione superiore in materie legate alla tecnologia come l'Intelligenza Artificiale, l'Internet of Things, il Cloud, la Customer Experience e la Digital Transformation
- ◆ Ingegnere Tecnico di Prevendita attraverso il Watson Customer Engagement (soluzioni di Marketing e Customer Experience) in Spagna, Portogallo, Grecia e Israele presso IBM
- ◆ Ingegnere di rete R&D presso Telefónica
- ◆ Laureato in Ingegneria delle Telecomunicazioni presso l'Università di Alcalá e l'Istituto Danese di Ingegneria Technical University

Dott. Pérez Rico, Javier

- ◆ Attuale Responsabile tecnico Android presso Nologis
- ◆ Responsabile tecnico Android presso Seekle
- ◆ Programmatore Android presso Gowex-Ideup
- ◆ Programmatore Junior Android presso Tecnom
- ◆ Relatore al II Simposio iTest, E@tic2011
- ◆ Ingegnere informatico in tecnica dei sistemi presso l'Università Complutense di Madrid
- ◆ Master in Ricerca presso l'Università Complutense di Madrid

Dott. Jiménez Pérez, Carlos

- ◆ Sviluppatore Android Senior presso OnTheSpot - Telefónica Tech
- ◆ Ingegneria dell'Automazione e dell'Elettronica
- ◆ Master in Ingegneria dei Sistemi e delle Applicazioni Elettroniche
- ◆ Professore Aggregato presso l'Università Carlos III di Madrid

Dott. Marcano Van Grieken, Alejandro Antonio

- ◆ Responsabile del Prodotto - Vikua, a Remoto (Jira, SCRUM, Figma, Slack, Notion)
- ◆ Sviluppatore Backend - InnovativeGX
- ◆ Laurea in Ingegneria dei Sistemi, Università Metropolitana di Caracas, Venezuela
- ◆ Master in Cybersecurity presso l'Università di León Online

Dott. Arevalillo González, Emilio

- ◆ DBA Oracle BBVA
- ◆ *Assistant Project Manager Archibus Solution Center Spain*
- ◆ Backend developer presso Telefónica I+D
- ◆ Laureato in Ingegneria Informatica presso l'Università Politecnica di Madrid
- ◆ Master Universitario in Software e Sistemi presso l'Università Politecnica di Madrid

Dott. Arranz, Héctor

- ◆ Responsabile dei progetti software presso Ezenit
- ◆ Laurea in Ingegneria del Software presso Università Complutense di MADRID
- ◆ Master MBA Power Leaders per The Power MBA Esperienza Docente
- ◆ Formatore di competenze digitali presso Three Life
- ◆ Formatore di competenze digitali presso la Fondazione Esplai
- ◆ Professore aggregato per il corso di laurea in sviluppo di applicazioni multiplatforma presso MEDAC
- ◆ Sostegno al lavoro imprenditoriale presso l'Università Complutense di Madrid
- ◆ Formatore di competenze digitali presso Three Life Fondazione Esplai
- ◆ Consulente per l'imprenditorialità presso Cink Venturing
- ◆ Professore aggregato per il corso di laurea in sviluppo di applicazioni multiplatforma presso MEDAC

Dott. Gozalo Fernández, Juan Luis

- ◆ Responsabile di Prodotto Blockchain presso Open Canarias
- ◆ Direttore Blockchain DevOps presso Alastria
- ◆ Direttore della Tecnologia del Livello di Servizio presso Santander Spagna
- ◆ Direttore Sviluppo Applicazioni Mobili Tinkerlink presso Cronos Telecom
- ◆ Direttore della Tecnologia di Gestione dei Servizi IT presso Barclays Bank Spagna
- ◆ Laurea in Ingegneria Informatica presso l'UNED +
- ◆ Specializzazione in Deep Learning presso DeepLearning.ai

05

Struttura e contenuti

Questo programma è stato progettato da professionisti che comprendono la realtà attuale e le tendenze future dello sviluppo di applicazioni per il sistema operativo Android. Il programma è costituito da 10 moduli che approfondiscono perfettamente tutto ciò che è necessario sapere per diventare uno sviluppatore di applicazioni. Fin dal modulo 1, la tecnica e la conoscenza approfondita della materia sono evidenti e possono essere padroneggiate in pochi mesi. Lo studente riceverà una specializzazione al 100% online, grazie alla metodologia del *Relearning* che consente un rapido apprendimento e la memorizzazione di concetti e pratiche.



“

*Contenuti dinamici con esercizi pratici,
video e formati interattivi determineranno
una specializzazione agile e confortevole”*

Modulo 1. Linguaggio di programmazione Android

- 1.1. Piattaforma Android
 - 1.1.1. Piattaforma Android
 - 1.1.2. Sistema Operativo Android
 - 1.1.3. *Open Handset Alliance* nello sviluppo di Android
- 1.2. Architettura Android
 - 1.2.1. Elementi architettonici di un sistema Android
 - 1.2.2. Comunicazione tra elementi
 - 1.2.3. Estensibilità dell'architettura Android
 - 1.2.4. Gestione delle risorse della macchina: batteria e memoria
 - 1.2.5. Emulatori Android
- 1.3. Kernel Linux di Android
 - 1.3.1. Composizione del kernel
 - 1.3.2. Elementi strutturali del kernel
 - 1.3.3. La macchina Virtuale Dalvik
 - 1.3.4. La macchina virtuale Android RunTime (ART)
- 1.4. Librerie native di Android
 - 1.4.1. Librerie native di Android
 - 1.4.2. Librerie di compatibilità (*Support Library*)
 - 1.4.3. Librerie native ed estensibilità
- 1.5. Il sistema di file e dati in Android
 - 1.5.1. Struttura di una tipica applicazione Android
 - 1.5.2. Sistema di file YAFFS2 ed ext4
 - 1.5.3. Uso di SQLite e Room per la gestione dei dati
- 1.6. Sicurezza in Android
 - 1.6.1. Sistema di permessi
 - 1.6.2. Firme digitali negli *Application Package* di Android (apk)
 - 1.6.3. Esecuzione dei processi nel kernel
 - 1.6.4. Thread di esecuzione ed eventi

- 1.7. Componenti strutturali di un'applicazione standard
 - 1.7.1. Vista (*View*)
 - 1.7.2. Attività (*Activity*)
 - 1.7.3. Frammento (*Fragment*)
 - 1.7.4. Servizio (*Service*)
 - 1.7.5. Intenzione (*Intent*)
 - 1.7.6. *Broadcasts Receiver* e *Content Provider*
 - 1.7.7. Gestione dei dati e preferenze degli utenti
- 1.8. Versioni Android
 - 1.8.1. Versioni Android
 - 1.8.2. Distribuzione delle versioni Android
 - 1.8.3. Dispersione delle distribuzioni Android
 - 1.8.4. Android vs. Apple IOS e altri sistemi mobili
- 1.9. Android per i veicoli
 - 1.9.1. Android e il mondo automobilistico
 - 1.9.2. Elementi strutturali di un sistema Android per automobili
 - 1.9.3. Comunicazione tra dispositivi
- 1.10. Android nella Domotica, *Wearable* e nell'*Internet of Things* (IoT)
 - 1.10.1. Il mondo connesso
 - 1.10.2. Elementi strutturali in un sistema Android domotico
 - 1.10.3. Elementi di Android *Wearable*
 - 1.10.4. Android nell'*Internet of Things* (IoT)

Modulo 2. *Frameworks* utilizzati nello sviluppo di applicazioni Android

- 2.1. *Frameworks* nello Sviluppo di Applicazioni Android
 - 2.1.1. *Framework* nello Sviluppo di Applicazioni Android
 - 2.1.2. *Frameworks*. Tipologie
 - 2.1.3. Scelta dei *Frameworks* per il progetto
- 2.2. Implementazione dei *Frameworks* in Android
 - 2.2.1. *Frameworks Core* di Android per Java/Kotlin
 - 2.2.2. *Jetpack Compose*
 - 2.2.3. *Frameworks* nelle altre lingue

- 2.3. Sistemi di gestione delle librerie in fase di sviluppo
 - 2.3.1. *Graddle*
 - 2.3.2. Automatizzazione con *Graddle*
 - 2.3.3. Strumento di sviluppo Maven
- 2.4. Codice pulito
 - 2.4.1. Codice ordinato
 - 2.4.2. Preparazione del codice nelle Applicazioni Android
 - 2.4.3. *Bikeshedding* e definizione delle priorità
- 2.5. Modelli di sviluppo in Android
 - 2.5.1. Categorie dei modelli
 - 2.5.2. Differenze tra modelli
 - 2.5.3. *Factory, Observer e Singleton*
- 2.6. MVP Modello, Vista e Presentatore
 - 2.6.1. MVC. Modello, Vista e Controllo
 - 2.6.2. Modello, Vista e Presentatore
 - 2.6.3. Esempio pratico: *Pokemon Battle*
- 2.7. MVVM. Modello, Vista e View Model
 - 2.7.1. MVC vs MVVM
 - 2.7.2. Modello, Vista e *View Model*
 - 2.7.3. Esempio pratico: *Pokemon Battle II*
- 2.8. *Frameworks* e librerie più comunemente utilizzati in Android
 - 2.8.1. Librerie di interazione con API
 - 2.8.2. Librerie di conversione dei dati
 - 2.8.3. *Firebase* e *Firebase Analytics*
- 2.9. *Framework* visivo di Android
 - 2.9.1. Ciclo di vita di un'applicazione Android
 - 2.9.2. Progettazione di viste in XML
 - 2.9.3. Progettazione di elementi e animazioni in XML
- 2.10. *Frameworks* di Android nelle altre lingue
 - 2.10.1. *React Native*
 - 2.10.2. Flutter
 - 2.10.3. Ionic

Modulo 3. Interfacce e *Layout* nello sviluppo di applicazioni Android

- 3.1. Ciclo di vita di un'interfaccia in Android
 - 3.1.1. Ciclo di vita in Android
 - 3.1.2. Relazione processo-attività
 - 3.1.3. Persistenza dello stato dell'applicazione
 - 3.1.4. Architettura *Clean* applicata ad Android
- 3.2. Viste nello sviluppo di Applicazioni Android (*Views*)
 - 3.2.1. Livello di presentazione dell'architettura *Clean*
 - 3.2.2. *Recycler View*
 - 3.2.3. *Adapter View*
- 3.3. Progettazione nello Sviluppo di Applicazioni Android (*Layouts*)
 - 3.3.1. *Layouts* in Android
 - 3.3.2. *Constraint Layout*
 - 3.3.3. Creazione dei *Layouts* usando *Android Studio Layout Editor*
- 3.4. Animazioni nello Sviluppo di Applicazioni Android (*Animations*)
 - 3.4.1. Icone e immagini
 - 3.4.2. Transizioni
 - 3.4.3. Differenza tra animazione delle proprietà e animazione delle viste
- 3.5. Attività e intenzioni nello sviluppo di Applicazioni Android (*Activity e Intentions*)
 - 3.5.1. Intenzioni esplicite e implicite
 - 3.5.2. Barra delle azioni
 - 3.5.3. Comunicazione tra le attività
- 3.6. Risorse alternative e di sistema (*Material Design, Cardboard, etc.*)
 - 3.6.1. *Material Design* per Android
 - 3.6.2. Multimedialità in Android
 - 3.6.3. Realtà virtuale con *Google Cardboard per Android NDK*
- 3.7. Stili e temi nello Sviluppo di Applicazioni Android
 - 3.7.1. Stili in un progetto Android
 - 3.7.2. Temi per il progetto Android
 - 3.7.3. Riutilizzo di stili e temi

- 3.8. Grafica, touch screen e sensori
 - 3.8.1. Lavorare con grafiche avanzate
 - 3.8.2. Gestione dei dispositivi touchscreen e a tastiera
 - 3.8.3. Utilizzo dei sensori del dispositivo con Android
- 3.9. Progettazioni per la realtà aumentata
 - 3.9.1. Interfacce complesse con l'utilizzo della videocamera
 - 3.9.2. Sensori di posizione e GPS nella realtà aumentata
 - 3.9.3. Presentazione su schermi non standard
 - 3.9.4. Errori e problemi comuni
- 3.10. Configurazione avanzata dell'interfaccia con AndroidManifest.xml
 - 3.10.1. La potenza del file manifest di Android
 - 3.10.2. Progettazione programmatica contro progettazione dichiarativa
 - 3.10.3. Componenti chiave dell'archivio

Modulo 4. Linguaggio di programmazione nelle applicazioni Android. Kotlin

- 4.1. Linguaggio di programmazione Kotlin
 - 4.1.1. Linguaggio di programmazione Kotlin
 - 4.1.2. Linguaggio di programmazione Kotlin. Caratteristiche
 - 4.1.3. Esecuzione di un programma in Kotlin
- 4.2. Programmare in Kotlin
 - 4.2.1. Struttura di un programma in Kotlin
 - 4.2.2. Parole riservate e sintassi
 - 4.2.3. Digitazione della console e lettura degli input dell'utente - Hello World
- 4.3. Tipi e variabili in Kotlin
 - 4.3.1. Tipi e variabili in Kotlin
 - 4.3.2. Dichiarazione delle variabili: Var vs. Val
 - 4.3.3. Operatori
 - 4.3.4. Conversione di tipo
 - 4.3.5. Arrays
- 4.4. Controllo del flusso in Kotlin
 - 4.4.1. Controllo del flusso
 - 4.4.2. Espressioni condizionali
 - 4.4.3. Loop





- 4.5. Funzioni in Kotlin
 - 4.5.1. Funzioni in Kotlin
 - 4.5.2. Struttura di una funzione
 - 4.5.3. Funzioni di portata (*Scope Functions*)
- 4.6. Classi e oggetti in Kotlin
 - 4.6.1. Classi e oggetti in Kotlin
 - 4.6.2. Classi
 - 4.6.3. Oggetti
 - 4.6.4. Costruttori e inizializzazione delle proprietà
 - 4.6.5. Classi annidate e classi interne
 - 4.6.6. Classi di dati
- 4.7. Ereditarietà in Kotlin
 - 4.7.1. Ereditarietà
 - 4.7.2. Superclassi e sottoclassi
 - 4.7.3. Sovrascrittura di proprietà e funzioni
 - 4.7.4. Ereditarietà vs. altri tipi di relazioni tra classi
 - 4.7.5. Classi sigillate
 - 4.7.6. Elenco
- 4.8. Classi astratte e interfacce in Kotlin
 - 4.8.1. Classi astratte e interfacce
 - 4.8.2. Classi astratte
 - 4.8.3. Interfacce
 - 4.8.4. Convalida e conversione dei tipi - Operatori `Is`, `When`, `As`
- 4.9. Collezioni in Kotlin
 - 4.9.1. Collezioni in Kotlin
 - 4.9.2. Lista
 - 4.9.3. Insieme
 - 4.9.4. Mappa
- 4.10. Gestione delle eccezioni e valori nulli in Kotlin
 - 4.10.1. Gestione delle eccezioni e valori nulli
 - 4.10.2. Valore nullo, tipi *Nullable* e *Non-Nullable*
 - 4.10.3. Eccezioni

Modulo 5. Linguaggio di programmazione nelle applicazioni Android. Kotlin avanzato genericità, programmazione funzionale e parallelismo

- 5.1. Genericità in Kotlin
 - 5.1.1. Genericità in Kotlin
 - 5.1.2. Genericità di collezioni, funzioni, classi e interfacce
 - 5.1.3. Covarianza e controvarianza: *Out o In*
- 5.2. Funzioni *Lambda* in Kotlin
 - 5.2.1. Funzioni *Lambda*
 - 5.2.2. Struttura di una funzione *Lambda*
 - 5.2.3. Uso delle funzioni *Lambda*
- 5.3. Funzioni di ordine superiore in Kotlin
 - 5.3.1. Funzioni di ordine superiore
 - 5.3.2. Funzioni di ordine superiore standard di Kotlin
 - 5.3.3. Collegamento delle chiamate di funzione
- 5.4. Estensioni in Kotlin
 - 5.4.1. Estensioni in Kotlin
 - 5.4.2. Funzioni di estensione
 - 5.4.3. Proprietà di estensione
 - 5.4.4. Oggetti companion
- 5.5. Il pattern *Delegate* in Kotlin
 - 5.5.1. Pattern *Delegate*
 - 5.5.2. *Delegation* in Kotlin
 - 5.5.3. Proprietà delegate
- 5.6. Annotazioni e riflessione in Kotlin
 - 5.6.1. Annotazioni e riflessione
 - 5.6.2. Annotazioni in Kotlin
 - 5.6.3. Riflessione in Kotlin
- 5.7. *Testing* in Kotlin
 - 5.7.1. *Testing* in Kotlin
 - 5.7.2. *Frameworks* e librerie di *Testing* in Kotlin
 - 5.7.3. Kotest

- 5.8. Programmazione asincrona in Kotlin
 - 5.8.1. Programmazione asincrona
 - 5.8.2. Tecniche di programmazione asincrona in Kotlin
 - 5.8.3. Confronto tra le tecniche di programmazione
- 5.9. Coroutine in Kotlin
 - 5.9.1. Coroutine
 - 5.9.2. Canali
 - 5.9.3. Contesto e *Dispatchers*
 - 5.9.4. Stato condiviso e concomitanza
 - 5.9.5. Gestione delle eccezioni nelle coroutines
- 5.10. Ecosistema Kotlin
 - 5.10.1. Ecosistema Kotlin
 - 5.10.2. Librerie per Kotlin
 - 5.10.3. Strumenti per Kotlin

Modulo 6. Gestione dei dati nei dispositivi Android

- 6.1. Gestione dei dati. Tipologia
 - 6.1.1. La gestione dei dati nei Dispositivi mobili
 - 6.1.2. Alternative per la gestione dei dati nei dispositivi Android
 - 6.1.3. Generazione di dati per il lavoro con l'intelligenza artificiale e l'analisi d'uso
 - 6.1.4. Strumenti di misurazione delle prestazioni per una gestione ottimale dei dati
- 6.2. Gestione di preferenze dell'utente
 - 6.2.1. Tipi di dati coinvolti nei file delle preferenze
 - 6.2.2. Gestione delle preferenze dell'utente
 - 6.2.3. Esportazione delle preferenze. Gestione dei permessi
- 6.3. Sistema di archiviazione dei file
 - 6.3.1. Classificazione dei file system sui dispositivi mobili
 - 6.3.2. File system interno
 - 6.3.3. File system esterno

- 6.4. File JSON come archivio su Android
 - 6.4.1. Informazioni non strutturate in file JSON
 - 6.4.2. Librerie di gestione dei dati JSON
 - 6.4.3. Utilizzo di JSON in Android. Raccomandazioni e ottimizzazioni
- 6.5. File XML come archivio su Android
 - 6.5.1. Formato XML in Android
 - 6.5.2. XML attraverso le librerie SAX
 - 6.5.3. XML attraverso le librerie DOM
- 6.6. Database SQLite
 - 6.6.1. Database relazionale per la gestione dei dati
 - 6.6.2. Uso del database
 - 6.6.3. Metodi SQLite per la gestione dei dati
- 6.7. Uso avanzato dei database SQLite
 - 6.7.1. Recupero dei guasti utilizzando le transazioni SQLite
 - 6.7.2. Utilizzo della cache per velocizzare l'accesso ai dati
 - 6.7.3. Database mobile
- 6.8. Libreria Room
 - 6.8.1. Architettura della libreria Room
 - 6.8.2. Libreria Room. Funzionalità
 - 6.8.3. Libreria Room: vantaggi e svantaggi
- 6.9. *Content Provider* per condividere informazioni
 - 6.9.1. *Content Provider* per condividere informazioni
 - 6.9.2. *Content Provider* in Android. Tecnico in funzione
 - 6.9.3. Sicurezza dei *Content Provider*
- 6.10. Raccolta dati nel Cloud di Internet
 - 6.10.1. Android e sistemi di archiviazione Cloud
 - 6.10.2. Servizi SOAP e REST per Android
 - 6.10.3. Problemi dei sistemi distribuiti
 - 6.10.4. Internet come backup dei dati dell'applicazione

Modulo 7. Strumenti dei dispositivi Android

- 7.1. Gestione: Strumento "TO DO"
 - 7.1.1. Strumenti di mercato
 - 7.1.2. Strumenti di mercato. Confronto delle funzionalità
 - 7.1.3. Strumenti di gestione. Differenze
- 7.2. MDM: gestione aziendale dei dispositivi
 - 7.2.1. Controllo dei dispositivi aziendali
 - 7.2.2. Analisi dei principali strumenti di mercato
 - 7.2.3. Scelta dello strumento
- 7.3. CRM: strumenti di mercato
 - 7.3.1. Analisi degli strumenti di mercato con applicazione Android
 - 7.3.2. Strumenti di mercato. Efficienza
 - 7.3.3. Strumenti di mercato. Usi
- 7.4. Droni con Android
 - 7.4.1. Applicazioni per dispositivi Android per il controllo dei droni
 - 7.4.2. Controlli autonomi
 - 7.4.3. Utilizzi del Drone in Android
- 7.5. Android, valore aggiunto nelle piattaforme bancarie
 - 7.5.1. Android nelle piattaforme bancarie
 - 7.5.2. Rischi e truffe della criminalità informatica
 - 7.5.3. Utilizzo dei dispositivi mobili
- 7.6. *Brokering* nei dispositivi mobili
 - 7.6.1. Strumenti di mercato e il loro utilizzo
 - 7.6.2. Confronto tra gli strumenti
 - 7.6.3. Scelta dello strumento per ogni utilizzo
- 7.7. Strumenti di intrattenimento e formazione
 - 7.7.1. Usi
 - 7.7.2. Strumenti di mercato
 - 7.7.3. Confronto tra le caratteristiche degli strumenti di sviluppo Android

- 7.8. IoT Android
 - 7.8.1. *Framework* e piattaforme di mercato
 - 7.8.2. Rischi e considerazioni sull'IoT in Android
 - 7.8.3. Utilizzi dell'IoT in Android
- 7.9. Efficienza dei processi
 - 7.9.1. Analisi degli strumenti di mercato per la creazione di App
 - 7.9.2. Confronto tra gli strumenti per la creazione di applicazioni per Android
 - 7.9.3. *Use Case*
- 7.10. Applicazioni più scaricate attualmente
 - 7.10.1. Strumenti più scaricati attualmente
 - 7.10.2. Raggruppamento per famiglie
 - 7.10.3. Usi primari, secondari e comparativi con IOS

Modulo 8. Design Responsivo in Android

- 8.1. *Design responsivo*
 - 8.1.1. *Design Responsivo*
 - 8.1.2. Usabilità, accessibilità e UX
 - 8.1.3. *Design Responsivo* Vantaggi e svantaggi
- 8.2. *Mobile vs. Tablet vs. Web vs. Smartwatches*
 - 8.2.1. Diversi formati, diverse dimensioni, diverse esigenze
 - 8.2.2. Problemi nel design
 - 8.2.3. Adattivo vs. Responsivo
- 8.3. Guide di stili
 - 8.3.1. Guide di stili. Utilità
 - 8.3.2. *Material Design*
 - 8.3.3. Guida di stili propria
- 8.4. *Layouting* flessibile
 - 8.4.1. *Layouting* flessibile
 - 8.4.2. *Layouting* di base
 - 8.4.3. *Layouting* in *Grid*
 - 8.4.4. *Layouting* con *Relative Layout*
 - 8.4.5. *Layouting* con *Constraint Layout*

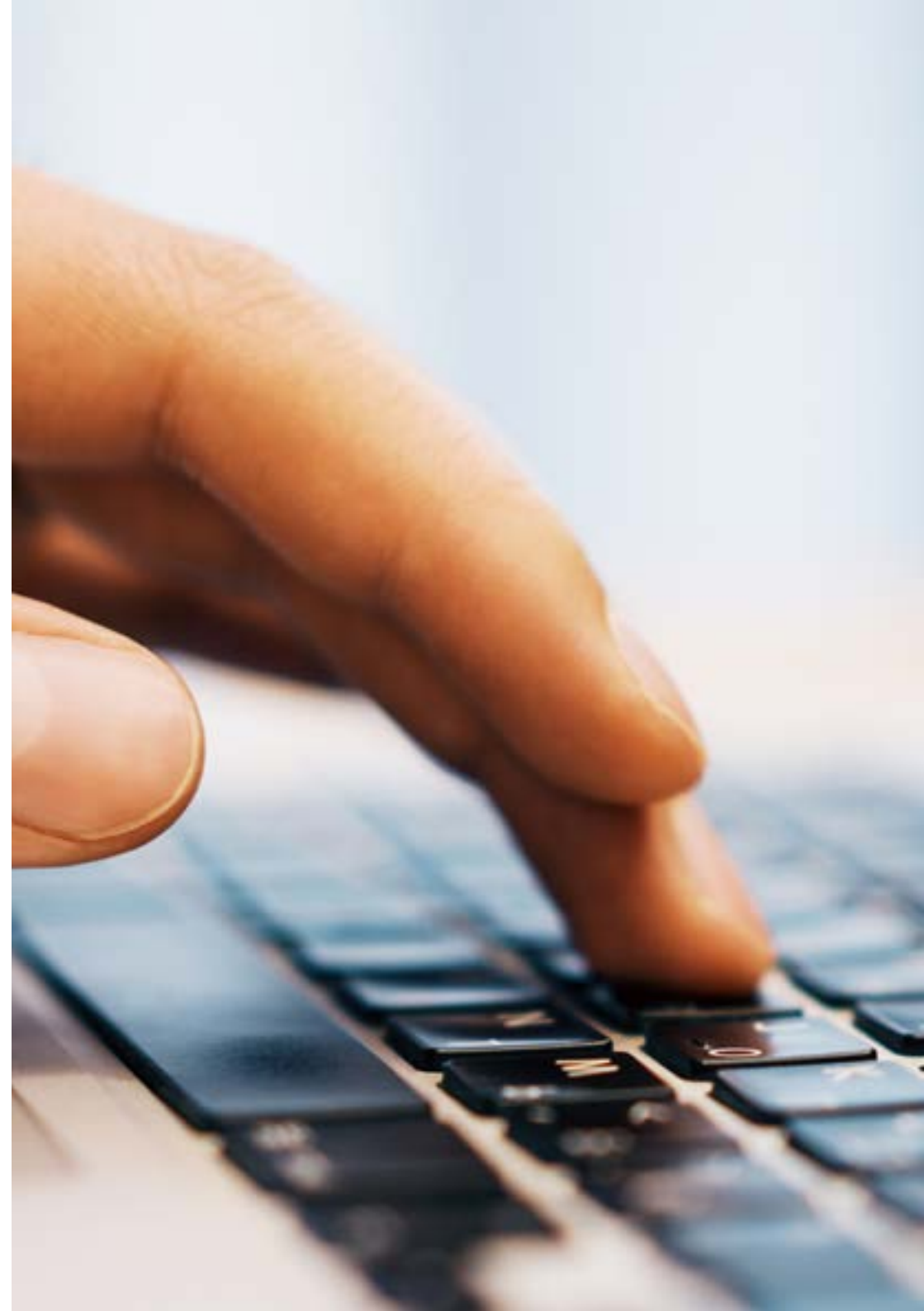




- 8.5. Risorse flessibili
 - 8.5.1. Risorse flessibili
 - 8.5.2. Immagini
 - 8.5.3. 9-Patch
 - 8.5.4. Risorse globali
- 8.6. Navigazione flessibile
 - 8.6.1. Navigazione flessibile
 - 8.6.2. Navigazione con *Activities*
 - 8.6.3. Navigazione con *Fragments*
- 8.7. Strumenti esterni
 - 8.7.1. Generatori automatici
 - 8.7.2. Strumenti di prototipazione
 - 8.7.3. Strumenti di progettazione
- 8.8. Debug e Test
 - 8.8.1. *Debug Layouts*
 - 8.8.2. *Test* automatici
 - 8.8.3. Sviluppo basato sui componenti
 - 8.8.4. Buone pratiche di *Testing e collaudi*
- 8.9. Alternative ad Android nativo I. *Web Pages*
 - 8.9.1. Progettazione in una *WebView*
 - 8.9.2. *Chrome Custom Tabs*
 - 8.9.3. *Debug e Test* nelle *web Pages*
- 8.10. Alternative ad Android nativo II Applicazioni ibride
 - 8.10.1. *React/React native*
 - 8.10.2. *Flutter*
 - 8.10.3. *Ionic*
 - 8.10.4. Apache Cordova

Modulo 9. Marketing delle applicazioni Android

- 9.1. Da *Customer Service* a *Customer Experience*
 - 9.1.1. *Customer Service*. Sviluppo del cliente attuale
 - 9.1.2. Utente con accesso alle informazioni. Esigenze e necessità
 - 9.1.3. Il Feedback come fonte di conoscenza
- 9.2. *Customer Journey*
 - 9.2.1. Percorso dell'utente alla conversione
 - 9.2.2. Micro-segmentazione
 - 9.2.3. Esperienza attraverso i canali
- 9.3. Misurazione dell'esperienza dell'utente
 - 9.3.1. Architettura web e mobile
 - 9.3.2. Analisi delle sessioni come nuovo standard
 - 9.3.3. Stato dell'arte dell'esperienza utente
- 9.4. Marketing delle Applicazioni Android
 - 9.4.1. CX+IA
 - 9.4.2. CX+Blockchain
 - 9.4.3. CX+IoT
- 9.5. Prodotti CX (Customer Experience)
 - 9.5.1. Standard dell'industria
 - 9.5.2. Telepresenza
 - 9.5.3. Esperienza del cliente per tutti gli agenti dello sviluppo
- 9.6. Lavoro incentrato sull'utente
 - 9.6.1. Attrezzatura
 - 9.6.2. Pensiero del designer
 - 9.6.3. Lavoro sul campo
- 9.7. La scienza sull'utente
 - 9.7.1. La scienza sull'utente. Regole d'oro
 - 9.7.2. Iterazione
 - 9.7.3. Errori più comuni
- 9.8. Prototipazione e Wireframing
 - 9.8.1. Prototipazione e *Wireframing*
 - 9.8.2. *Hands-On*
 - 9.8.3. Livello avanzato



- 9.9. Interfacce mobili
 - 9.9.1. Design visivo. Regole
 - 9.9.2. Interfaccia delle applicazioni. Chiavi
 - 9.9.3. Migliori pratiche nello Sviluppo di Interfacce Mobili
- 9.10. Migliori pratiche per l'esperienza utente Consigli per gli Sviluppatori
 - 9.10.1. Livello uno. Migliori pratiche di CX
 - 9.10.2. Livello Due. Migliori pratiche di UX
 - 9.10.3. Livello tre. Migliori pratiche di UI

Modulo 10. Cicli di vita delle applicazioni Android. Cloud, Playstore e versioning

- 10.1. Cicli di vita del Software
 - 10.1.1. Cicli di vita del Software
 - 10.1.2. Metodologie agili
 - 10.1.3. Il ciclo continuo agile del software
- 10.2. Sviluppo del prodotto manuale
 - 10.2.1. Integrazione manuale
 - 10.2.2. Consegna manuale
 - 10.2.3. Distribuzione manuale
- 10.3. Integrazione supervisionata
 - 10.3.1. Integrazione continua
 - 10.3.2. Revisione manuale
 - 10.3.3. Revisioni automatiche statiche
- 10.4. Test logici
 - 10.4.1. Test Unitari
 - 10.4.2. Test di integrazione
 - 10.4.3. Test di comportamento
- 10.5. Integrazione continua
 - 10.5.1. Ciclo di integrazione continua
 - 10.5.2. Dipendenze tra integrazioni
 - 10.5.3. Integrazione continua come metodologia di gestione del content repository
- 10.6. Consegna continua
 - 10.6.1. Consegna continua. Tipologia di problemi da risolvere
 - 10.6.2. Consegna continua, Risoluzione dei problemi
 - 10.6.3. Vantaggi della consegna continua
- 10.7. Distribuzione continua
 - 10.7.1. Distribuzione continua. Tipologia di problemi da risolvere
 - 10.7.2. Distribuzione continua. Risoluzione dei Problemi
- 10.8. *Firebase Test Lab*
 - 10.8.1. Configurazione da *GCloud*
 - 10.8.2. Configurazione di Jenkins
 - 10.8.3. Uso di Jenkins. Vantaggi
- 10.9. Configurazione di *Gradle*
 - 10.9.1. Sistema di automazione *Gradle*
 - 10.9.2. Componente di *Gradle Build Flavors*
 - 10.9.3. Componente di *Gradle Linteo*
- 10.10. Cicli di vita delle Applicazioni Android. Esempio
 - 10.10.1. Configurazione di *SemaphoreCI* e GitHub
 - 10.10.2. Configurazione dei blocchi di lavoro
 - 10.10.3. Promozioni e *Deployment*



Non esitare oltre e fai un passo verso il tuo futuro professionale. Inizia a progettare le migliori applicazioni mobili grazie a questo Master Privato"

06

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.



07 Titolo

Il Master Privato in Sviluppo di Applicazioni Android ti garantisce, oltre alla preparazione più rigorosa e aggiornata, l'accesso a una qualifica di Master Privato rilasciata da TECH Università Tecnologica.





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

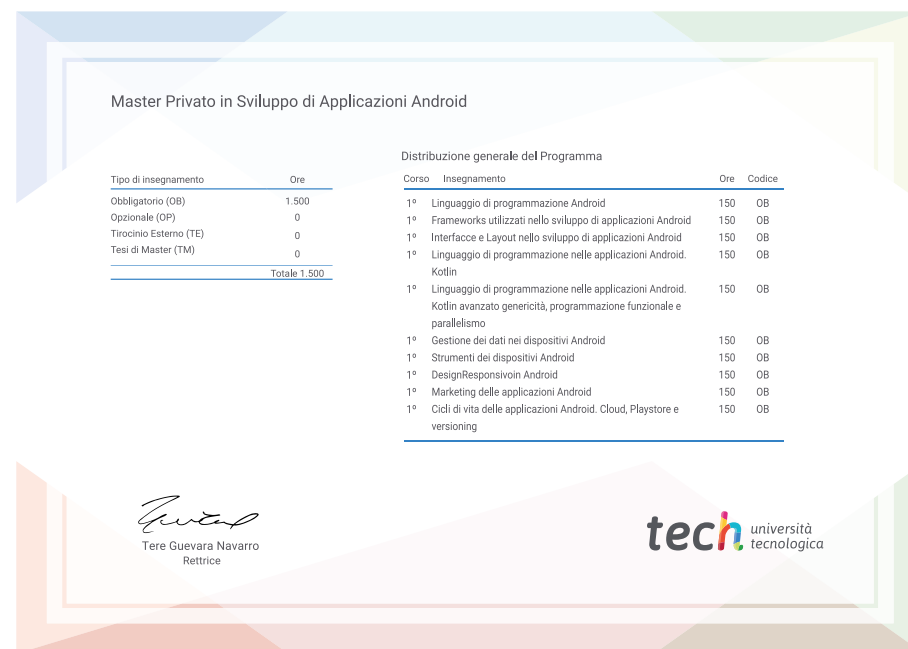
Questo **Master Privato in Sviluppo di Applicazioni Android** possiede il programma più completo e aggiornato del mercato.

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

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

Titolo: **Master Privato in Sviluppo di Applicazioni Android**

N. Ore Ufficiali: **1.500**



*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

Master Privato
Sviluppo di Applicazioni
Android

- » Modalità: **online**
- » Durata: **12 mesi**
- » Titolo: **TECH Università Tecnologica**
- » Dedizione: **16 ore/settimana**
- » Orario: **a scelta**
- » Esami: **online**

