

Universitätsexperte

Programmiersprachen für Videospiele

the modifier unless its not a mesh")

ject is not None



Universitätsexperte Programmiersprachen für Videospiele

- » Modalität: online
- » Dauer: 6 Monate
- » Qualifizierung: TECH Technologische Universität
- » Aufwand: 16 Std./Woche
- » Zeitplan: in Ihrem eigenen Tempo
- » Prüfungen: online

Internetzugang: www.techtitute.com/de/informatik/spezialisierung/spezialisierung-programmiersprachen-videospiele

Index

01

Präsentation

Seite 4

02

Ziele

Seite 8

03

Struktur und Inhalt

Seite 12

04

Methodik

Seite 18

05

Qualifizierung

Seite 26

01

Präsentation

Das Gebiet der Programmierung ist sehr breit gefächert und umfasst verschiedene Arten von Sprachen, je nach dem Zweck oder der Umgebung, für die sie bestimmt ist. Videospiele weisen eine Reihe von Besonderheiten auf, die dazu führen, dass für die korrekte Entwicklung von Videospiegelprojekten Programmierer benötigt werden, die auf diesen Bereich spezialisiert sind. Diese Besonderheiten beziehen sich nicht nur auf die Art der Programmierung, sondern auch auf die verwendeten Sprachen. Daher sind entsprechende Kenntnisse in diesem Bereich erforderlich, um qualitativ hochwertigen Code zu erstellen. Dieser Studiengang vermittelt den Studenten die notwendigen Werkzeuge und Kenntnisse, um qualitativ hochwertige Videospiele zu programmieren und eröffnet ihnen zahlreiche Berufsmöglichkeiten in dieser Branche.





“

*Spezialisieren Sie sich auf die
wesentlichen Programmiersprachen für
die Entwicklung hochwertiger Videospiele”*

Auf der Welt gibt es viele Programmiersprachen. Einige sind sehr eingeschränkt und werden nur unter ganz bestimmten Umständen verwendet. Andere sind trotz ihres sehr eingeschränkten Anwendungsbereichs absolut unentbehrlich, wie z. B. Maschinensprache oder Assembler, die dazu dienen, die grundlegenden Anweisungen zwischen Hardware und Software in Betriebssystemen zu definieren.

Es gibt auch Sprachen, die nur in einem einzigen System verwendet werden, wie Visual Basic, Sprachen, die universell geworden sind, wie C, und Sprachen, die in den letzten Jahren aufgrund ihrer Nützlichkeit für statistische Aspekte an Popularität gewonnen haben, wie Python.

Es gibt also für jedes Problem oder jede Situation eine Programmiersprache. Videospiele sind da keine Ausnahme, denn sie benötigen eine bestimmte Art von Sprache, um korrekt entwickelt werden zu können. Objektorientierte Programmiersprachen wie C++ sind in diesem Prozess von entscheidender Bedeutung. Aus diesem Grund ist dieser Universitätsexperte in Programmiersprachen für Videospiele die Antwort für alle, die alles über die Entwicklung dieser Art von Werken lernen wollen.

Dank dieser Qualifikation können die Studenten zu echten Experten auf diesem Gebiet werden, was ihnen die Türen zur Industrie öffnet und ihnen dank der in diesem Kurs erworbenen Fähigkeiten Zugang zu den besten Unternehmen der Branche verschafft.

Dieser **Universitätsexperte in Programmiersprachen für Videospiele** enthält das vollständigste und aktuellste Programm auf dem Markt. Die hervorstechendsten Merkmale sind:

- ♦ Die Entwicklung von praktischen Fällen, die von Experten für die Entwicklung von Videospiele präsentiert werden
- ♦ Der anschauliche, schematische und äußerst praxisnahe Inhalt vermittelt alle für die berufliche Praxis unverzichtbaren wissenschaftlichen und praktischen Informationen
- ♦ Er enthält praktische Übungen, in denen der Selbstbewertungsprozess durchgeführt werden kann, um das Lernen zu verbessern
- ♦ Sein besonderer Schwerpunkt liegt auf innovativen Methoden
- ♦ Theoretische Vorträge, Fragen an den Experten, Diskussionsforen zu kontroversen Themen und individuelle Reflexionsarbeit
- ♦ Die Verfügbarkeit des Zugangs zu Inhalten von jedem festen oder tragbaren Gerät mit Internetanschluss



Lernen Sie die Programmiersprachen kennen, die die von Ihnen entwickelten Videospiele erfolgreich machen werden”

“

Die Programmierung ist für die Entwicklung eines Videospiele unerlässlich. Die Programmierung ist für die Entwicklung eines Videospiele unerlässlich. Schreiben Sie sich in diesen Universitätsexperten ein und fördern Sie Ihre berufliche Karriere"

Zu den Dozenten des Programms gehören Fachleute aus der Branche, die ihre Erfahrungen aus ihrer Arbeit in diese Weiterbildung einbringen, sowie anerkannte Spezialisten aus führenden Unternehmen und renommierten Universitäten.

Die multimedialen Inhalte, die mit der neuesten Bildungstechnologie entwickelt wurden, werden der Fachkraft ein situierendes und kontextbezogenes Lernen ermöglichen, d. h. eine simulierte Umgebung, die eine immersive Fortbildung bietet, die auf die Ausführung von realen Situationen ausgerichtet ist.

Das Konzept dieses Programms konzentriert sich auf problemorientiertes Lernen, bei dem die Fachkraft versuchen muss, die verschiedenen Situationen aus der beruflichen Praxis zu lösen, die während des gesamten Studiengangs gestellt werden. Zu diesem Zweck wird sie von einem innovativen interaktiven Videosystem unterstützt, das von renommierten Experten entwickelt wurde.

Die Spieleindustrie sucht talentierte Programmierer wie Sie. Lassen Sie sich diese Chance nicht entgehen und schreiben Sie sich ein.

Programmieren Sie Ihren Weg zum Erfolg mit diesem Universitätsexperten.



02 Ziele

Das Hauptziel dieses Universitätsexperten in Programmiersprachen für Videospiele ist es, Studenten zu großen Experten in der Entwicklung von Videospiele zu machen, damit sie die besten Berufschancen in einer der aufregendsten Branchen der Welt haben. Mit den in diesem Studiengang erworbenen Kenntnissen werden sie in der Lage sein, ihre Karriere voranzutreiben und mit den renommiertesten Unternehmen der Branche zusammenzuarbeiten.





“

*Sie werden der beste
Videospieleprogrammierer in Ihrer Umgebung
sein. Spezialisieren Sie sich jetzt”*



Allgemeine Ziele

- ◆ Kennenlernen der verschiedenen Programmiermethoden, die bei Videospielen angewendet werden
- ◆ Vertiefen in den Produktionsprozess von Videospielen und die Integration der Programmierung in diese Phasen
- ◆ Beherrschen der grundlegenden Programmiersprachen, die in Videospielen verwendet werden
- ◆ Anwenden von Wissen über *Software Engineering* und spezialisierte Programmierung auf Videospiele
- ◆ Verstehen, welche Rolle die Programmierung bei der Entwicklung eines Videospieles spielt
- ◆ Entwickeln von Web-Videospielen



Die Programmierung ist für Unternehmen in diesem Sektor unerlässlich. Wenn Sie diesen Abschluss machen, haben Sie die besten Karrierechancen in dieser faszinierenden Branche"





Spezifische Ziele

Modul 1. Objektorientierte Programmierung

- ◆ Verstehen der verschiedenen Entwurfsmuster für objektorientierte Probleme
- ◆ Verstehen der Bedeutung von Dokumentation und Testen in der Softwareentwicklung
- ◆ Verwalten des Einsatzes von Threading und Synchronisierung sowie das Lösen gängiger Probleme bei der gleichzeitigen Programmierung

Modul 2. 3D-Modellierung

- ◆ Bestimmen der internen Struktur einer Videospiele-Engine
- ◆ Festlegen der Elemente einer modernen Spielarchitektur
- ◆ Verstehen der Funktionen der einzelnen Komponenten eines Videospieles
- ◆ Veranschaulichen von Videospiele, die mit 2D- und 3D-Grafiken erstellt wurden

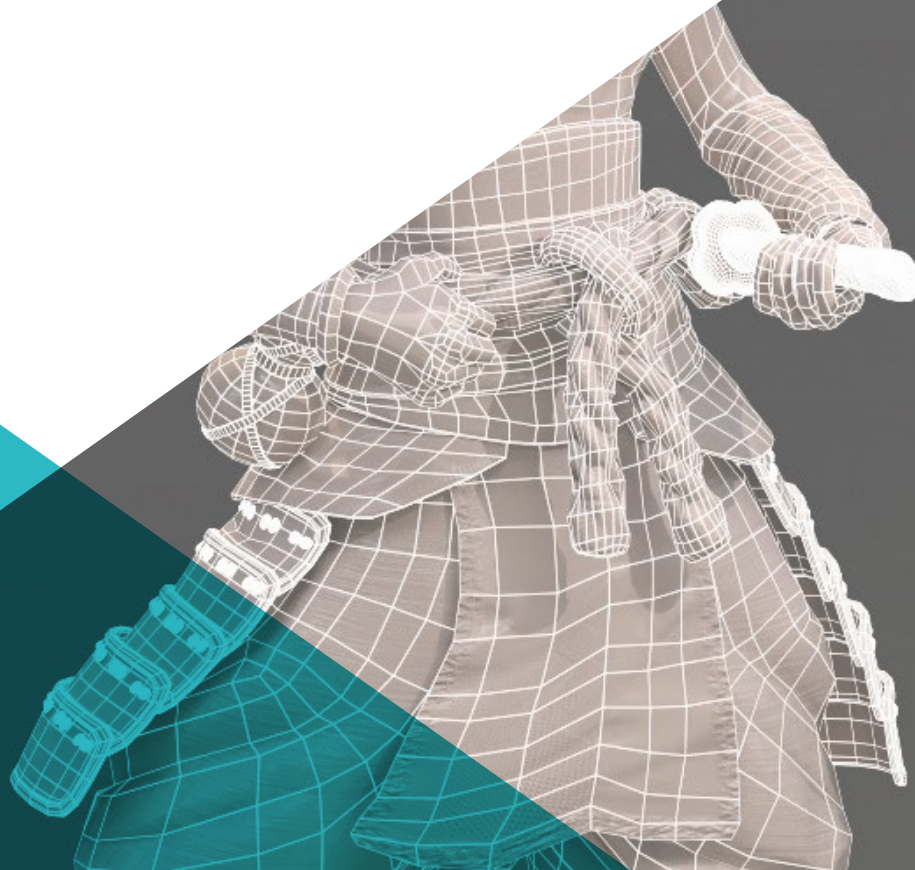
Modul 3. Design und Entwicklung von Webspielen

- ◆ Entwerfen von interaktiven Webspielen und -anwendungen mit der entsprechenden Dokumentation
- ◆ Bewerten der Hauptmerkmale von Spielen und interaktiven Webanwendungen, um professionell und korrekt zu kommunizieren

03

Struktur und Inhalt

Der Studiengang umfasst einen hochspezialisierten Lehrplan, der von führenden Experten auf diesem Gebiet entwickelt wurde. Die Studenten profitieren von Kenntnissen, die vollständig auf die Berufspraxis ausgerichtet sind, wodurch sie zu gefragten Arbeitskräften in der Branche werden. Aus diesem Grund ist dieser Universitätsexperte in Programmiersprachen für Videospiele die Lösung für all diejenigen, die ihre berufliche Laufbahn ändern möchten, um in diesem komplexen Sektor erfolgreich zu sein.





“

Sie werden keinen besseren Studiengang finden, um sich in Programmiersprachen für Videospiele zu spezialisieren”

Modul 1. Objektorientierte Programmierung

- 1.1. Einführung in die objektorientierte Programmierung
 - 1.1.1. Einführung in die objektorientierte Programmierung
 - 1.1.2. Klassen-Design
 - 1.1.3. Einführung in UML für die Modellierung von Problemen
- 1.2. Beziehungen zwischen Klassen
 - 1.2.1. Abstraktion und Vererbung
 - 1.2.2. Fortgeschrittene Konzepte der Vererbung
 - 1.2.3. Polymorphismen
 - 1.2.4. Zusammensetzung und Aggregation
- 1.3. Einführung in *Design Patterns* für objektorientierte Probleme
 - 1.3.1. Was sind Entwurfsmuster?
 - 1.3.2. *Factory*-Muster
 - 1.3.3. *Singleton*-Muster
 - 1.3.4. *Observer*-Muster
 - 1.3.5. *Composite*-Muster
- 1.4. Ausnahmen
 - 1.4.1. Was sind Ausnahmen?
 - 1.4.2. Abfangen und Behandlung von Ausnahmen
 - 1.4.3. Werfen von Ausnahmen
 - 1.4.4. Erstellung von Ausnahmen
- 1.5. Benutzeroberflächen
 - 1.5.1. Einführung in Qt
 - 1.5.2. Positionierung
 - 1.5.3. Was sind Ereignisse?
 - 1.5.4. Ereignisse: Definition und Erfassung
 - 1.5.5. Entwicklung von Benutzeroberflächen
- 1.6. Einführung in die gleichzeitige Programmierung
 - 1.6.1. Einführung in die gleichzeitige Programmierung
 - 1.6.2. Das Konzept des *Thread*-Prozesses
 - 1.6.3. Interaktion zwischen Prozessen oder *Threads*
 - 1.6.4. *Threads* in C++
 - 1.6.5. Vor- und Nachteile der gleichzeitigen Programmierung
- 1.7. *Thread*-Verwaltung und Synchronisierung
 - 1.7.1. Lebenszyklus eines *Threads*
 - 1.7.2. Die Klasse *Thread*
 - 1.7.3. Planung des *Threads*
 - 1.7.4. Gruppen von *Threads*
 - 1.7.5. Dämonen-*Threads*
 - 1.7.6. Synchronisierung
 - 1.7.7. Verriegelungsmechanismen
 - 1.7.8. Kommunikationsmechanismen
 - 1.7.9. Monitore
- 1.8. Häufige Probleme bei der gleichzeitigen Programmierung
 - 1.8.1. Das Erzeuger-Verbraucher-Problem
 - 1.8.2. Das Problem von Lesern und Schreibern
 - 1.8.3. Das Problem des Abendessens der Philosophen
- 1.9. Dokumentation und Prüfung von Software
 - 1.9.1. Warum ist es wichtig, Software zu dokumentieren?
 - 1.9.2. Design-Dokumentation
 - 1.9.3. Verwendung von Tools zur Dokumentation
- 1.10. Software-Tests
 - 1.10.1. Einführung in die Softwareprüfung
 - 1.10.2. Arten von Tests
 - 1.10.3. Einheitstest
 - 1.10.4. Integrationstests
 - 1.10.5. Validierungstest
 - 1.10.6. Systemprüfung

Modul 2. 3D-Modellierung

- 2.1. Einführung in C#
 - 2.1.1. Was ist OOP?
 - 2.1.2. Visual Studio Umgebung
 - 2.1.3. Datentypen
 - 2.1.4. Typen-Konvertierungen
 - 2.1.5. Bedingungen
 - 2.1.6. Objekte und Klassen
 - 2.1.7. Modularität und Verkapselung
 - 2.1.8. Vererbung
 - 2.1.9. Abstrakte Klassen
 - 2.1.10. Polymorphismen
- 2.2. Mathematische Grundlagen
 - 2.2.1. Mathematische Werkzeuge in der Physik: Skalare und vektorielle Größen
 - 2.2.2. Mathematische Werkzeuge in der Physik: Skalarprodukt
 - 2.2.3. Mathematische Werkzeuge in der Physik: Vektorprodukt
 - 2.2.4. Mathematische Tools in OOP
- 2.3. Physikalische Grundlagen
 - 2.3.1. Der starre Festkörper
 - 2.3.2. Kinematik
 - 2.3.3. Dynamik
 - 2.3.4. Kollisionen
 - 2.3.5. Projektile
 - 2.3.6. Flug
- 2.4. Grundlagen der Computergrafik
 - 2.4.1. Grafik-Systeme
 - 2.4.2. 2D-Grafiken
 - 2.4.3. 3D-Grafiken
 - 2.4.4. Raster-Systeme
 - 2.4.5. Geometrische Modellierung
 - 2.4.6. Eliminierung von verdeckten Teilen
 - 2.4.7. Realistische Visualisierung
 - 2.4.8. OpenGL-Grafikbibliothek
- 2.5. Unity: Einführung und Installation
 - 2.5.1. Was ist Unity?
 - 2.5.2. Warum Unity?
 - 2.5.3. Merkmale von Unity
 - 2.5.4. Installation
- 2.6. Unity: 2D und 3D
 - 2.6.1. Gameplay in 2D: Sprites und Tilemaps
 - 2.6.2. Gameplay in 2D: 2D Physics
 - 2.6.3. Beispiele für Videospiele, die mit Unity 2D entwickelt wurden
 - 2.6.4. Einleitung zu Unity 3D
- 2.7. Unity: Instanziierung und Objekterstellung
 - 2.7.1. Hinzufügen von Komponenten
 - 2.7.2. Entfernen von Komponenten
 - 2.7.3. Importieren von Assets und Texturen
 - 2.7.4. Materialien und *Maps* für Materialien
- 2.8. Unity: Interaktionen und Physik
 - 2.8.1. *Rigidbody*
 - 2.8.2. *Colliders*
 - 2.8.3. *Joints* (Gelenke)
 - 2.8.4. *Character Controllers*
 - 2.8.5. *Continous Collision Detection* (CCD)
 - 2.8.6. *Physics Debug Visualization*
- 2.9. Unity: Grundlegende künstliche Intelligenz für NPCs
 - 2.9.1. *Pathfinding* in Unity: NavMesh
 - 2.9.2. Feind mit KI
 - 2.9.3. NPC Aktionsbaum
 - 2.9.4. NPC-Hierarchie und *Scripts*
- 2.10. Unity: Grundlagen und Implementierung von Animationen
 - 2.10.1. *Animation Controller*: Charakter-Assoziation
 - 2.10.2. *Blend Trees*-Mischbaum
 - 2.10.3. Übergang zwischen Zuständen
 - 2.10.4. Ändern des Schwellenwerts für Übergänge

Modul 3. Design und Entwicklung von Webspielen

- 3.1. Ursprünge und Standards des Webs
 - 3.1.1. Die Ursprünge des Internets
 - 3.1.2. Die Entstehung des World Wide Web
 - 3.1.3. Aufkommen von Webstandards
 - 3.1.4. Der Aufstieg der Webstandards
- 3.2. HTTP und Client-Server-Struktur
 - 3.2.1. Client-Server-Rolle
 - 3.2.2. Client-Server-Kommunikation
 - 3.2.3. Jüngste Geschichte
 - 3.2.4. Zentralisierte Datenverarbeitung
- 3.3. Web-Programmierung: Einführung
 - 3.3.1. Grundlegende Konzepte
 - 3.3.2. Einrichten eines Webservers
 - 3.3.3. HTML5-Grundlagen
 - 3.3.4. HTML-Formulare
- 3.4. Einführung in HTML und Beispiele
 - 3.4.1. Geschichte von HTML5
 - 3.4.2. Elemente von HTML5
 - 3.4.3. APIS
 - 3.4.4. CCS3
- 3.5. *DDocument Object Model*
 - 3.5.1. Was ist das *Document Object Model*?
 - 3.5.2. Verwendung von DOCTYPE
 - 3.5.3. Die Bedeutung der Validierung von HTML
 - 3.5.4. Zugriff auf Elemente
 - 3.5.5. Elemente und Texte erstellen
 - 3.5.6. InnerHTML verwenden
 - 3.5.7. Ein Textelement oder einen Knoten löschen
 - 3.5.8. Lesen und Schreiben der Attribute eines Elements
 - 3.5.9. Manipulation von Elementstilen
 - 3.5.10. Mehrere Dateien auf einmal anhängen
- 3.6. Einführung in CSS und Beispiele
 - 3.6.1. CSS3-Syntax
 - 3.6.2. Stil-Blätter
 - 3.6.3. Tags
 - 3.6.4. Selektoren
 - 3.6.5. Webgestaltung mit CSS
- 3.7. Einführung in JavaScript und Beispiele
 - 3.7.1. Was ist JavaScript?
 - 3.7.2. Kurze Geschichte der Sprache
 - 3.7.3. JavaScript-Versionen
 - 3.7.4. Ein Dialogfeld anzeigen
 - 3.7.5. JavaScript-Syntax
 - 3.7.6. Skripte verstehen
 - 3.7.7. Räume
 - 3.7.8. Kommentare
 - 3.7.9. Funktionen
 - 3.7.10. Seiteninternes und externes JavaScript
- 3.8. Funktionen in JavaScript
 - 3.8.1. Funktionsdeklarationen
 - 3.8.2. Funktion Ausdrücke
 - 3.8.3. Funktionen aufrufen
 - 3.8.4. Rekursion
 - 3.8.5. Verschachtelte Funktionen und Schließungen
 - 3.8.6. Variable Konservierung
 - 3.8.7. Mehrfach verschachtelte Funktionen
 - 3.8.8. Namenskonflikte
 - 3.8.9. Schließungen
 - 3.8.10. Parameter einer Funktion



- 3.9. PlayCanvas für die Entwicklung von Webspielen
 - 3.9.1. Was ist PlayCanvas?
 - 3.9.2. Projekt-Konfiguration
 - 3.9.3. Ein Objekt erstellen
 - 3.9.4. Hinzufügen von Physiken
 - 3.9.5. Hinzufügen eines Modells
 - 3.9.6. Ändern der Schwerkraft- und Szeneneinstellungen
 - 3.9.7. *Scripts* ausführen
 - 3.9.8. Kamera-Steuerungen
- 3.10. *Phaser* für die Entwicklung von Webspielen
 - 3.10.1. Was ist *Phaser*?
 - 3.10.2. Ressourcen laden
 - 3.10.3. Die Welt bauen
 - 3.10.4. Die Plattformen
 - 3.10.5. Der Spieler
 - 3.10.6. Hinzufügen von Physiken
 - 3.10.7. Verwendung der Tastatur
 - 3.10.8. Aufnehmen von *Pickups*
 - 3.10.9. Punkte und Wertung
 - 3.10.10. *Bounce*-Pumpen



Entwickeln Sie die berühmtesten
Videospiele der Zukunft dank allem, was
Sie in diesem Studium lernen werden"

04 Methodik

Dieses Fortbildungsprogramm bietet eine andere Art des Lernens. Unsere Methodik wird durch eine zyklische Lernmethode entwickelt: **das Relearning**. Dieses Lehrsystem wird z. B. an den renommiertesten medizinischen Fakultäten der Welt angewandt und wird von wichtigen Publikationen wie dem **New England Journal of Medicine** als eines der effektivsten angesehen.



“

Entdecken Sie Relearning, ein System, das das herkömmliche lineare Lernen hinter sich lässt und Sie durch zyklische Lehrsysteme führt: eine Art des Lernens, die sich als äußerst effektiv erwiesen hat, insbesondere in Fächern, die Auswendiglernen erfordern"

Fallstudie zur Kontextualisierung aller Inhalte

Unser Programm bietet eine revolutionäre Methode zur Entwicklung von Fähigkeiten und Kenntnissen. Unser Ziel ist es, Kompetenzen in einem sich wandelnden, wettbewerbsorientierten und sehr anspruchsvollen Umfeld zu stärken.

“

Mit TECH werden Sie eine Art des Lernens erleben, die an den Grundlagen der traditionellen Universitäten auf der ganzen Welt rüttelt”



Sie werden Zugang zu einem Lernsystem haben, das auf Wiederholung basiert, mit natürlichem und progressivem Unterricht während des gesamten Lehrplans.



Der Student wird durch gemeinschaftliche Aktivitäten und reale Fälle lernen, wie man komplexe Situationen in realen Geschäftsumgebungen löst.

Eine innovative und andersartige Lernmethode

Dieses TECH-Programm ist ein von Grund auf neu entwickeltes, intensives Lehrprogramm, das die anspruchsvollsten Herausforderungen und Entscheidungen in diesem Bereich sowohl auf nationaler als auch auf internationaler Ebene vorsieht. Dank dieser Methodik wird das persönliche und berufliche Wachstum gefördert und ein entscheidender Schritt in Richtung Erfolg gemacht. Die Fallmethode, die Technik, die diesem Inhalt zugrunde liegt, gewährleistet, dass die aktuellste wirtschaftliche, soziale und berufliche Realität berücksichtigt wird.



Unser Programm bereitet Sie darauf vor, sich neuen Herausforderungen in einem unsicheren Umfeld zu stellen und in Ihrer Karriere erfolgreich zu sein"

Die Fallmethode ist das am weitesten verbreitete Lernsystem an den besten Informatikschulen der Welt, seit es sie gibt. Die Fallmethode wurde 1912 entwickelt, damit Jurastudenten das Recht nicht nur auf der Grundlage theoretischer Inhalte erlernen. Sie bestand darin, ihnen reale komplexe Situationen zu präsentieren, damit sie fundierte Entscheidungen treffen und Werturteile darüber fällen konnten, wie diese zu lösen sind. Sie wurde 1924 als Standardlehrmethode in Harvard etabliert.

Was sollte eine Fachkraft in einer bestimmten Situation tun? Mit dieser Frage konfrontieren wir Sie in der Fallmethode, einer handlungsorientierten Lernmethode.

Während des gesamten Kurses werden die Studenten mit mehreren realen Fällen konfrontiert. Sie müssen ihr gesamtes Wissen integrieren, recherchieren, argumentieren und ihre Ideen und Entscheidungen verteidigen.

Relearning Methodology

TECH kombiniert die Methodik der Fallstudien effektiv mit einem 100%igen Online-Lernsystem, das auf Wiederholung basiert und in jeder Lektion verschiedene didaktische Elemente kombiniert.

Wir ergänzen die Fallstudie mit der besten 100%igen Online-Lehrmethode: Relearning.

*Im Jahr 2019 erzielten wir die besten
Lernergebnisse aller spanischsprachigen
Online-Universitäten der Welt.*

Bei TECH lernen Sie mit einer hochmodernen Methodik, die darauf ausgerichtet ist, die Führungskräfte der Zukunft zu spezialisieren. Diese Methode, die an der Spitze der weltweiten Pädagogik steht, wird Relearning genannt.

Unsere Universität ist die einzige in der spanischsprachigen Welt, die für die Anwendung dieser erfolgreichen Methode zugelassen ist. Im Jahr 2019 ist es uns gelungen, die Gesamtzufriedenheit unserer Studenten (Qualität der Lehre, Qualität der Materialien, Kursstruktur, Ziele...) in Bezug auf die Indikatoren der besten spanischsprachigen Online-Universität zu verbessern.



In unserem Programm ist das Lernen kein linearer Prozess, sondern erfolgt in einer Spirale (lernen, verlernen, vergessen und neu lernen). Daher wird jedes dieser Elemente konzentrisch kombiniert. Mit dieser Methode wurden mehr als 650.000 Hochschulabsolventen mit beispiellosem Erfolg in so unterschiedlichen Bereichen wie Biochemie, Genetik, Chirurgie, internationales Recht, Managementfähigkeiten, Sportwissenschaft, Philosophie, Recht, Ingenieurwesen, Journalismus, Geschichte, Finanzmärkte und -instrumente fortgebildet. Dies alles in einem sehr anspruchsvollen Umfeld mit einer Studentenschaft mit hohem sozioökonomischem Profil und einem Durchschnittsalter von 43,5 Jahren.

Das Relearning ermöglicht es Ihnen, mit weniger Aufwand und mehr Leistung zu lernen, sich mehr auf Ihre Spezialisierung einzulassen, einen kritischen Geist zu entwickeln, Argumente zu verteidigen und Meinungen zu kontrastieren: eine direkte Gleichung zum Erfolg.

Nach den neuesten wissenschaftlichen Erkenntnissen der Neurowissenschaften wissen wir nicht nur, wie wir Informationen, Ideen, Bilder und Erinnerungen organisieren, sondern auch, dass der Ort und der Kontext, in dem wir etwas gelernt haben, von grundlegender Bedeutung dafür sind, dass wir uns daran erinnern und es im Hippocampus speichern können, um es in unserem Langzeitgedächtnis zu behalten.

Auf diese Weise sind die verschiedenen Elemente unseres Programms im Rahmen des so genannten Neurocognitive Context-Dependent E-Learning mit dem Kontext verbunden, in dem der Teilnehmer seine berufliche Praxis entwickelt.



Dieses Programm bietet die besten Lehrmaterialien, die sorgfältig für Fachleute aufbereitet sind:



Studienmaterial

Alle didaktischen Inhalte werden von den Fachleuten, die den Kurs unterrichten werden, speziell für den Kurs erstellt, so dass die didaktische Entwicklung wirklich spezifisch und konkret ist.

Diese Inhalte werden dann auf das audiovisuelle Format angewendet, um die Online-Arbeitsmethode von TECH zu schaffen. All dies mit den neuesten Techniken, die in jedem einzelnen der Materialien, die dem Studenten zur Verfügung gestellt werden, qualitativ hochwertige Elemente bieten.



Meisterklassen

Die Nützlichkeit der Expertenbeobachtung ist wissenschaftlich belegt.

Das sogenannte Learning from an Expert festigt das Wissen und das Gedächtnis und schafft Vertrauen für zukünftige schwierige Entscheidungen.



Übungen für Fertigkeiten und Kompetenzen

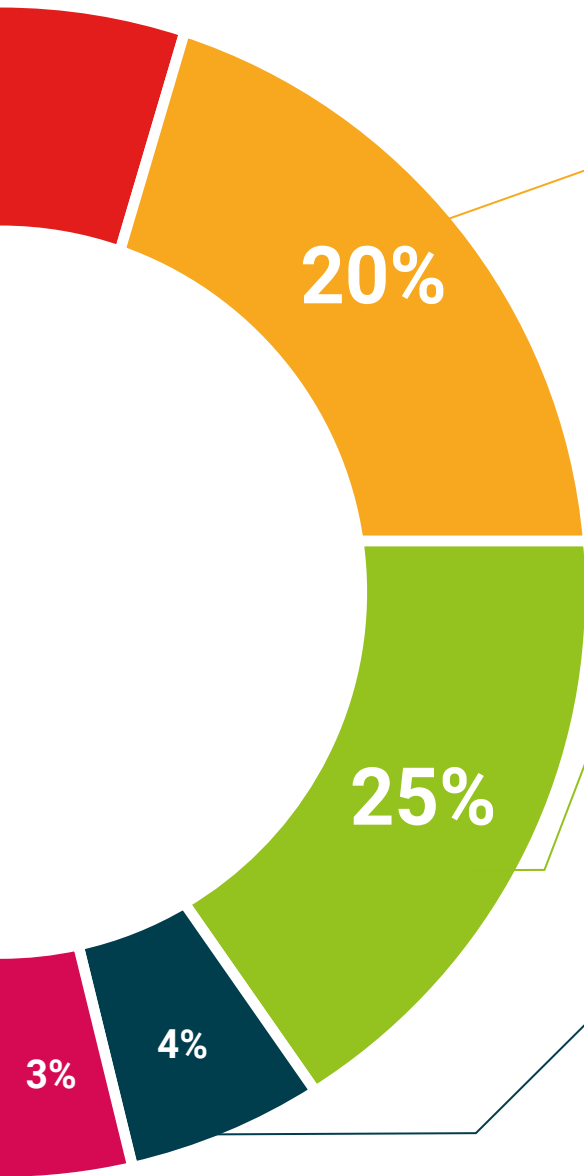
Sie werden Aktivitäten durchführen, um spezifische Kompetenzen und Fertigkeiten in jedem Fachbereich zu entwickeln. Übungen und Aktivitäten zum Erwerb und zur Entwicklung der Fähigkeiten und Fertigkeiten, die ein Spezialist im Rahmen der Globalisierung, in der wir leben, entwickeln muss.



Weitere Lektüren

Aktuelle Artikel, Konsensdokumente und internationale Leitfäden, u. a. In der virtuellen Bibliothek von TECH hat der Student Zugang zu allem, was er für seine Fortbildung benötigt.





Case Studies

Sie werden eine Auswahl der besten Fallstudien vervollständigen, die speziell für diese Qualifizierung ausgewählt wurden. Die Fälle werden von den besten Spezialisten der internationalen Szene präsentiert, analysiert und betreut.



Interaktive Zusammenfassungen

Das TECH-Team präsentiert die Inhalte auf attraktive und dynamische Weise in multimedialen Pillen, die Audios, Videos, Bilder, Diagramme und konzeptionelle Karten enthalten, um das Wissen zu vertiefen.

Dieses einzigartige Bildungssystem für die Präsentation multimedialer Inhalte wurde von Microsoft als "Europäische Erfolgsgeschichte" ausgezeichnet.



Testing & Retesting

Die Kenntnisse des Studenten werden während des gesamten Programms regelmäßig durch Bewertungs- und Selbsteinschätzungsaktivitäten und -übungen beurteilt und neu bewertet, so dass der Student überprüfen kann, wie er seine Ziele erreicht.



05

Qualifizierung

Der Universitätsexperte in Programmiersprachen für Videospiele garantiert neben der präzisesten und aktuellsten Fortbildung auch den Zugang zu einem von der TECH Technologischen Universität ausgestellten Diplom.



“

*Schließen Sie dieses Programm erfolgreich ab
und erhalten Sie Ihren Universitätsabschluss
ohne lästige Reisen oder Formalitäten"*

Dieser **Universitätsexperte in Programmiersprachen für Videospiele** enthält das vollständigste und aktuellste Programm auf dem Markt.

Sobald der Student die Prüfungen bestanden hat, erhält er/sie per Post, mit Empfangsbestätigung das entsprechende Diplom, ausgestellt von der **TECH Technologische Universität**.

Das von **TECH Technologische Universität** ausgestellte Diplom drückt die erworbene Qualifikation aus und entspricht den Anforderungen, die in der Regel von Stellenbörsen, Auswahlprüfungen und Berufsbildungsausschüssen verlangt werden.

Titel: **Universitätsexperte in Programmiersprachen für Videospiele**

Anzahl der offiziellen Arbeitsstunden: **450 Std.**



zukunft

gesundheit vertrauen menschen
erziehung information tutoren
garantie akkreditierung unterricht
institutionen technologie lernen
gemeinschaft verpflichtung
persönliche betreuung innovation
wissen gegenwart qualität
online-Ausbildung
entwicklung institutionen
virtuelles Klassenzimmer

tech technologische
universität

Universitätsexperte
Programmiersprachen
für Videospiele

- » Modalität: online
- » Dauer: 6 Monate
- » Qualifizierung: TECH Technologische Universität
- » Aufwand: 16 Std./Woche
- » Zeitplan: in Ihrem eigenen Tempo
- » Prüfungen: online

Universitätsexperte

Programmiersprachen für Videospiele

