


Privater Masterstudiengang Programmierung in Salesforce



the desele
active = modifier ob
(modifier_ob)) # modifi
= 0
.selected_objects[0]
cs[one.name].select = 1
please select exactly two objects,
OPERATOR CLASSES



Privater Masterstudiengang Programmierung in Salesforce

- » Modalität: online
- » Dauer: 12 Monate
- » Qualifizierung: TECH Technologische Universität
- » Zeitplan: in Ihrem eigenen Tempo
- » Prüfungen: online

Internetzugang: www.techtitute.com/de/informatik/masterstudiengang/masterstudiengang-programmierung-salesforce

Index

01

Präsentation

Seite 4

02

Ziele

Seite 8

03

Kompetenzen

Seite 14

04

Kursleitung

Seite 18

05

Struktur und Inhalt

Seite 24

06

Methodik

Seite 34

07

Qualifizierung

Seite 42

01 Präsentation

Im IT-Kontext hat sich Salesforce zu einem leistungsstarken Werkzeug für die Innovation des Kundenerlebnisses durch benutzerfreundliche, in die Cloud integrierte Unternehmensanwendungen entwickelt. Diese Technologie ermöglicht es Entwicklern, maßgeschneiderte Programme für Institutionen zu erstellen, um ihre *Customer Relationship Management*-Lösungen auf ihre spezifischen Bedürfnisse zuzuschneiden. Zu ihren Vorteilen gehört die Automatisierung einer breiten Palette von Geschäftsprozessen, vom *Lead-Management* über den Kundenservice bis hin zum Projektmanagement. In Anbetracht der zunehmenden Bedeutung dieser Plattform führt TECH einen Hochschulabschluss ein, der die modernsten Strategien für komplexe Implementierungslösungen vermittelt. Und das alles in einem bequemen 100%igen Online-Format!



“

Mit diesem 100%igen Online-Masterstudiengang werden Sie Salesforce beherrschen und fortgeschrittene Fähigkeiten erwerben, um vollständig angepasste Anwendungen zu entwickeln“

Die deklarative Programmierung in Salesforce ist ein grundlegender Prozess für die IT, da sie es den Benutzern erleichtert, die Plattform mithilfe visueller, regelbasierter Tools zu konfigurieren und anzupassen. Auf diese Weise können Administratoren Salesforce auf der Grundlage von Geschäftsanforderungen konfigurieren, ohne dass sie sich auf Entwickler verlassen müssen, die benutzerdefinierten Code schreiben. Dies wiederum ermöglicht es den Teams, Änderungen schneller zu implementieren und die Kosten zu senken, die mit der Entwicklung und Wartung von maßgeschneiderten Lösungen verbunden sind. Darüber hinaus fördert es eine größere Autonomie und die Fähigkeit der Benutzer, sich an die sich ändernden Anforderungen der Organisationen anzupassen.

Vor diesem Hintergrund hat TECH einen innovativen Privaten Masterstudiengang in Programmierung in Salesforce entwickelt. Sein Hauptziel ist es, den Studenten die besten Entwicklungspraktiken für diese Plattform sowie die ausgefeiltesten Techniken zu vermitteln, um sie zum Erfolg im Bereich der Programmierung zu führen. Zu diesem Zweck befasst sich der Studiengang mit Themen, die von den verschiedenen Modellen für die Sichtbarkeit von Protokollen über *Multitenant*-Architekturen bis hin zur Verwaltung von *Communities* für externe Kunden reichen. Außerdem werden den Programmierern eine Reihe von deklarativen Automatisierungswerkzeugen an die Hand gegeben, die eine schnelle Iteration und Wartbarkeit ermöglichen. Dementsprechend werden in den Lehrmaterialien unter anderem verschiedene Maßnahmen zur Förderung der Sicherheit in Apex, Visualforce und Aura besprochen.

Zudem zeichnet sich die Methodik dieses Universitätsabschlusses durch seinen 100%igen Online-Modus aus, der sich an die Bedürfnisse von vielbeschäftigten Fachleuten anpasst, die sich zum Ziel gesetzt haben, in ihrer Karriere voranzukommen. Außerdem wird das *Relearning*-System verwendet, das auf der Wiederholung der wichtigsten Konzepte basiert, um das Wissen zu festigen und das Lernen zu erleichtern. Die Kombination aus Flexibilität und einem soliden pädagogischen Ansatz macht das Programm somit sehr zugänglich. Darüber hinaus haben die Fachleute Zugang zu einer umfassenden virtuellen Bibliothek mit Multimedia-Ressourcen in verschiedenen Formaten, die ein dynamisches Lernen ermöglichen.

Dieser **Privater Masterstudiengang in Programmierung in Salesforce** enthält das vollständigste und aktuellste Programm auf dem Markt. Die hervorstechendsten Merkmale sind:

- ♦ Die Entwicklung von Fallstudien, die von Experten für Programmierung in Salesforce präsentiert werden
- ♦ Der anschauliche, schematische und äußerst praxisnahe Inhalt vermittelt alle für die berufliche Praxis unverzichtbaren praktischen Informationen
- ♦ Praktische Übungen, anhand derer der Selbstbewertungsprozess zur Verbesserung des Lernens verwendet werden kann
- ♦ Sein besonderer Schwerpunkt liegt auf innovativen Methoden
- ♦ Theoretische Lektionen, Fragen an den Experten, Diskussionsforen zu kontroversen Themen und individuelle Reflexionsarbeit
- ♦ Die Verfügbarkeit des Zugriffs auf die Inhalte von jedem festen oder tragbaren Gerät mit Internetanschluss



Eine Fortbildung, die Sie über die neuesten Trends und Aktualisierungen der Salesforce-Plattform auf dem Laufenden hält, um Ihre Programmierkenntnisse zu verbessern“



TECH stellt Ihnen ein breites Spektrum an Lernmaterialien in verschiedenen Formaten zur Verfügung, darunter Erklärungsvideos, ergänzende Lektüre und reale Fallstudien“

Das Dozententeam des Programms besteht aus Experten des Sektors, die ihre Berufserfahrung in diese Fortbildung einbringen, sowie aus renommierten Fachkräften von führenden Gesellschaften und angesehenen Universitäten.

Die multimedialen Inhalte, die mit der neuesten Bildungstechnologie entwickelt wurden, werden der Fachkraft ein situiertes 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.

Dank der revolutionären Relearning-Methodik werden Sie alle Konzepte effektiv integrieren, um erfolgreich die gewünschten Ergebnisse zu erzielen.

Sie werden in das OmniStudio-Framework eintauchen, um Geschäftsanwendungen in Salesforce effizient zu erstellen, anzupassen und zu verwalten.



02 Ziele

Nach Abschluss dieses privaten Masterstudiengangs verfügen die Absolventen über vertiefte Kenntnisse der Salesforce-Plattform, einschließlich ihrer Funktionalitäten und Entwicklungsmöglichkeiten. Außerdem werden die Informatiker ihre Programmierfähigkeiten verbessern, um hochgradig angepasste Anwendungen zu entwerfen, die sowohl Apex als auch Visualforce und *Lightning Web Components* auf der Grundlage von Javascript verwenden. In diesem Sinne werden sie eine effektive Datenverwaltung durch die Ausführung von SOQL- und SOSL-Abfragen durchführen. All dies wird es den Studenten ermöglichen, technische Probleme sofort zu lösen, sowie Fehler in den Codes der Sprachen zu diagnostizieren oder zu beheben. Auf diese Weise werden sie qualifiziert sein, um die Möglichkeiten zu nutzen, die die Computerindustrie bietet.





“

Sie werden agile Softwareentwicklungsmethoden in ihre tägliche Praxis einbeziehen und diese Ansätze anwenden, um die Erstellung von Anwendungen optimal zu verwalten“



Allgemeine Ziele

- ♦ Erwerben der fortgeschrittensten Konzepte für die Programmierung auf der Salesforce-Plattform
- ♦ Etablieren des Salesforce-Ökosystems als Entwicklungsplattform und Festlegen der relevanten Rahmenbedingungen
- ♦ Erwerben von Fachwissen über die Funktionsweise des in der Salesforce-Plattform verwendeten metadatengesteuerten Entwicklungsmodells
- ♦ Analysieren, wie Salesforce mit Daten und einer gebietsübergreifenden Infrastruktur arbeitet
- ♦ Entwickeln der Grundlagen der Salesforce-Datenbank
- ♦ Analysieren aller Tools und erweiterten Optionen zum Erstellen von Objekten in Salesforce
- ♦ Fördern der korrekten Verwendung von APIs und Tools zur Datenmanipulation
- ♦ Identifizieren der wichtigsten Tools und Ressourcen für die Programmierung in Apex
- ♦ Vorstellen der *Testing*-Werkzeuge, um das ordnungsgemäße Funktionieren der entwickelten Komponenten zu gewährleisten
- ♦ Fördern der Verwendung von *Best Practices* in der Entwicklung, die mit den von Salesforce definierten Standards übereinstimmen



Implementieren Sie Sicherheitssysteme in Ihren Projekten in Salesforce-Umgebungen, einschließlich der Konfiguration von Profilen, um die Einhaltung gesetzlicher Vorschriften zu gewährleisten“





Spezifische Ziele

Modul 1. Programmierung in Salesforce

- ♦ Zusammenstellen der Arten von Umgebungen, die für die Entwicklung der Salesforce-Plattform verwendet werden können
- ♦ Analysieren, wie die Datenbankarchitektur, auf der Salesforce basiert, funktioniert, um Entwicklungen innerhalb der Plattform flexibler zu gestalten
- ♦ Bewerten der Funktionsweise des Modells der Datensichtbarkeit und der gemeinsamen Nutzung von Datensätzen durch die Benutzer der Anwendung
- ♦ Bestimmen des bestehenden Berechtigungsmodells in der Plattform, um den Benutzern, die die von Salesforce vorgenommenen Entwicklungen nutzen werden, den erforderlichen Zugriff zu gewähren

Modul 2. Datenmodellierung in Salesforce

- ♦ Fortgeschrittenes Analysieren aller Salesforce-Tools für die Erstellung von Objekten und Attributen in Salesforce
- ♦ Vertiefen der *Best Practices* zur Datenmanipulation in Salesforce
- ♦ Untersuchen der besten Techniken für die Verwendung der Abfragesprachen SOQL und SOSL
- ♦ Lösen von Leistungsproblemen bei der Arbeit mit großen Datenmengen

Modul 3. Deklarative Programmierung in Salesforce

- ♦ Aufbauen von Fachwissen über kundenspezifische Formen und Funktionen auf der Plattform
- ♦ Vorschlagen von Lösungen für das Seitenlayout mit deklarativen Werkzeugen
- ♦ Entwickeln und Bewerten von Arbeitsabläufen unter Berücksichtigung von Grenzen und *Best Practices*
- ♦ Erstellen automatisierter Konfigurationen zur Verfolgung von Genehmigungsprozessen

Modul 4. Programmierung in APEX für Salesforce

- ♦ Beherrschen der Entwicklungswerkzeuge in Apex
- ♦ Bewerten der Unterschiede zwischen Java und Apex und dabei auf die Besonderheiten eingehen, die Apex von Java unterscheiden
- ♦ Analysieren der Möglichkeiten für den Datenzugriff aus Apex
- ♦ Entwickeln synchroner und asynchroner Prozesse, Kennen ihrer Grenzen und Anwenden von *Best Practices*

Modul 5. Programmierung von Benutzeroberflächen in Salesforce

- ♦ Vertiefen der verschiedenen in Salesforce verfügbaren *Frameworks* für die UI-Entwicklung
- ♦ Vertiefen der Ereignisverwaltung und wie benutzerdefinierte Komponenten innerhalb der Anwendung kommunizieren
- ♦ Identifizieren der verschiedenen Werkzeuge zur Gewährleistung der Qualität des entwickelten Codes
- ♦ Untersuchen der verschiedenen *Frameworks*, die für die Erstellung von Unit-Tests und *End-to-End* verwendet werden

Modul 6. OmniStudio-Framework

- ♦ Entwickeln von Fachwissen über die in Omnistudio enthaltenen Komponenten
- ♦ Analysieren und Spezifizieren der Verwendung der einzelnen Omnistudio-Komponenten
- ♦ Einführen von *Best Practices* für die Verwendung dieser Komponenten
- ♦ Vertiefen der Verknüpfung zwischen den verschiedenen Omnistudio-Komponenten, um die Lösung auf die effektivste und skalierbarste Weise zu entwickeln

Modul 7. APIs und Integrationen in Salesforce

- ◆ Spezifizieren der Art der Konfiguration, die in Salesforce ausführen ist, um eingehende Integrationen zu aktivieren und Sicherheitsrichtlinien einzugrenzen
- ◆ Aufzeigen der Funktionsweise der einzelnen vorhandenen API-Typen auf der Salesforce-Plattform
- ◆ Entwickeln verschiedener praktischer Fälle des Exports und Imports von Daten in großen Mengen unter Verwendung der Bulk-API
- ◆ Darstellen der verschiedenen Arten von Vorgängen und Ermitteln der Art von Informationen, die in jedem dieser Vorgänge übermittelt werden Definieren, wie wir den Event-Bus der Salesforce-Plattform abonnieren können, um die Ereignisse zu erfassen, die in dem Kanal veröffentlicht werden, an dem wir interessiert sind

Modul 8. Fortgeschrittene Programmierung in Salesforce

- ◆ Bewerten verschiedener Trigger-Frameworks mit Schwerpunkt auf den jeweiligen Merkmalen und den Vorteilen, die sie bieten
- ◆ Entwerfen eines Mechanismus zur Fehlerbehandlung, bei dem alle unkontrollierten Fehler, die während der Ausführung des angepassten Anwendungscodes auftreten, detailliert aufgezeichnet werden
- ◆ Untersuchen der SOLID-Prinzipien, die die Entwicklung effizienter, robuster und skalierbarer Software ermöglichen
- ◆ Erwerben von Fachwissen über Apex Enterprise Patterns, die definieren, wie der Apex-Code zu organisieren ist, um die Wartbarkeit, Skalierbarkeit und Robustheit zu gewährleisten





Modul 9. Sicherheit in Salesforce

- Vertiefen der *Best Practices* der gängigsten *Frameworks* zur Salesforce-Entwicklung (Apex, Visualforce, Aura, LWC)
- Analysieren der umgekehrten Sicherheitspyramide, auf der das Salesforce-Sicherheitsmodell basiert
- Zusammenstellen aller verfügbaren Maßnahmen zum Schutz vor externen Angriffen
- Korrektes Überwachen des Zugriffs auf die Anwendung, sowohl durch Benutzer als auch durch Programme

Modul 10. *Application Lifecycle Management (ALM)* in Salesforce

- Bestimmen der empfohlene Umgebungskarte für die Entwicklung, den Test und die Bereitstellung von Änderungen in produktiven Umgebungen
- Bewerten der verschiedenen Strategien zur Verzweigung des Quellcode-Repositorys, auf die sich das Team bei der Arbeit in einer kollaborativen Umgebung verlassen kann
- Untersuchen der in Salesforce DX verfügbaren Tools für den Austausch von Metadaten und die Ausführung von Transaktionen in Salesforce-Umgebungen
- Erstellen von befehlsgesteuerten Entwicklungsumgebungen auf der Grundlage der Grundlagen der Scratch Orgs

03

Kompetenzen

Dank dieses Hochschulprogramms erwerben die Teilnehmer fortgeschrittene Fähigkeiten für die Entwicklung maßgeschneiderter Anwendungen in Salesforce und beherrschen gleichzeitig die Apex-Sprache. Dies ermöglicht es IT-Fachleuten, die spezifischen Anforderungen von Organisationen zu erfüllen, einschließlich der Erstellung von individualisierten Objekten, Validierungsregeln und Workflows. Gleichzeitig werden die Fachleute in der Lage sein, Daten auf dieser Plattform effizient zu handhaben und Prozesse wie den Import, Export und die Umwandlung von Informationen durchzuführen. Sie werden sich auch durch einen problemlösenden Ansatz auszeichnen. So werden sie in der Lage sein, Apex-Code-Fehler mit Geschick zu lösen.



“

Erweitern Sie Ihre Designfähigkeiten, um die effektivsten und benutzerfreundlichsten Benutzeroberflächen unter Verwendung des Lightning-Frameworks zu erstellen“



Allgemeine Kompetenzen

- ♦ Entwickeln maßgeschneiderter und anspruchsvoller Seiten für Desktop- und mobile Anwendungen, die den Bedürfnissen der Kunden entsprechen
- ♦ Gewährleisten sicherer und skalierbarer Designs im Salesforce-Datenmodell
- ♦ Implementieren von automatisierten Workflows zur Optimierung von Geschäftsprozessen
- ♦ Bewerten der Möglichkeiten, die die deklarative Programmierung innerhalb der Plattform bietet
- ♦ Entwickeln praktischer Fähigkeiten bei der Erstellung maßgeschneiderter Lösungen
- ♦ Handhaben der Integrationsmuster, die innerhalb von Salesforce verwendet werden können, um den aktuellen Bedürfnissen gerecht zu werden
- ♦ Identifizieren fortschrittlicher Entwicklungswerkzeuge zur Erleichterung der Codierung und Fehlersuche
- ♦ Nutzen der sekundären Tools, die Salesforce für die Überwachung und detaillierte Zugriffsbeschränkung nach Art des Verfahrens bereitstellt
- ♦ Definieren des metadatengesteuerten Ansatzes, auf den sich Salesforce stützt, um Konfigurationen und Entwicklungen in verschiedenen Umgebungen zu synchronisieren
- ♦ Festlegen des Prozesses für die Konfiguration und Erstellung von Paketen, die die von uns durchgeführten Entwicklungen bündeln





Spezifische Kompetenzen

- ♦ Erwerben von Kenntnissen zur Erstellung benutzerdefinierter Benutzeroberflächen mit Lightning
- ♦ Automatisieren von Geschäftsprozessen mithilfe von APIs, Konnektoren und *Middleware*
- ♦ Korrektes Entwerfen des ERD-Datenmodells in Salesforce, wobei die *Out-of-the-Box*-Funktionen von Salesforce im Vordergrund stehen
- ♦ Identifizieren von Optionen zur Vermeidung von Doppelarbeit und zur Gewährleistung der Datenqualität
- ♦ Vorschlagen von Lösungen für das Seitenlayout mit deklarativen Werkzeugen
- ♦ Entwickeln und Bewerten von Arbeitsabläufen unter Berücksichtigung von Grenzen und *Best Practices*
- ♦ Vorschlagen von Lösungen für das Seitenlayout mit deklarativen Werkzeugen
- ♦ Entwickeln und Bewerten von Arbeitsabläufen unter Berücksichtigung von Grenzen und *Best Practices*
- ♦ Bestimmen, wie Probleme mit benutzerdefinierten Komponenten angegangen werden können und welche Tools zur Verfügung stehen, um sie zu erkennen
- ♦ Entwickeln benutzerdefinierter mobiler Anwendungen, die in Salesforce gespeicherte Informationen mithilfe des Mobile SDK nutzen
- ♦ Analysieren des Ausführungs- und Leistungsverhaltens bei der Verwendung von Omnistudio
- ♦ Entwickeln von Beispielen für den Einsatz von Omnistudio-Komponenten
- ♦ Entwickeln verschiedener praktischer Fälle des Exports und Imports von Daten in großen Mengen unter Verwendung der Bulk-API
- ♦ Darstellen der verschiedenen Arten von Vorgängen und Ermitteln der Art von Informationen, die in jedem dieser Vorgänge übermittelt werden Definieren, wie wir den Event-Bus der Salesforce-Plattform abonnieren können, um die Ereignisse zu erfassen, die in dem Kanal veröffentlicht werden, an dem wir interessiert sind
- ♦ Entwerfen eines Mechanismus zur Fehlerbehandlung, bei dem alle unkontrollierten Fehler, die während der Ausführung des angepassten Anwendungscodes auftreten, detailliert aufgezeichnet werden
- ♦ Untersuchen verschiedener Techniken zur Lösung potenzieller Probleme, die über die Grenzen der Plattform hinausgehen
- ♦ Korrektes Überwachen des Zugriffs auf die Anwendung, sowohl durch Benutzer als auch durch Programme
- ♦ Verhindern des unbefugten Zugriffs mit sekundären Mitteln oder böswilligen Handlungen (Datenlecks)
- ♦ Korrektes Überwachen des Zugriffs auf die Anwendung, sowohl durch Benutzer als auch durch Programme

04 Kursleitung

Die Philosophie der TECH beruht darauf, den Studenten erstklassige Studiengänge zu bieten, die einen Qualitätssprung in ihrer beruflichen Laufbahn ermöglichen. Aus diesem Grund wählt sie die Lehrkräfte, die jeden ihrer Studiengänge gestalten, sorgfältig aus. Für diesen privaten Masterstudiengang hat sie eine Gruppe von hochspezialisierten Fachleuten auf dem Gebiet der Salesforce-Programmierung zusammengebracht. Diese Experten verfügen über einen umfangreichen Arbeitshintergrund, in dem sie in renommierten Unternehmen tätig waren, um innovative IT-Lösungen anzubieten. Auf diese Weise kommen die Studenten in den Genuss von akademischem Material, das sie bei der Optimierung ihrer täglichen Praxis unterstützt.



“

Dank der Anleitung des Dozententeams werden Sie in der Lage sein, alle Herausforderungen, die sich bei der Implementierung von Salesforce-Lösungen ergeben, erfolgreich zu meistern“

Leitung



Hr. Tovar Barranco, Iosu Igor

- Chief Architect bei NTT Data Group
- Software Architect bei Beesion Technologies
- Systemadministrator bei Araldi
- .NET-Entwickler bei Gabinete de Gestión
- JAVA-Programmierer und J2EE-Anwendungsentwickler
- IT-Ingenieur von der Offenen Universität von Katalonien
- Spezialisiert auf SOA-Architekturen
- Diverse Salesforce-Zertifizierungen

Professoren

Hr. Sacie Alcázar, David

- ♦ Technischer Architekt und Fachberater bei NTT Data
- ♦ Leitender Programmierer bei Telefónica
- ♦ Leitender Entwickler an der Universität von Navarra
- ♦ Technologie-Analyst bei SN Arquitectura
- ♦ Technischer Leiter bei BBVA Compass
- ♦ Masterstudiengang in Telekommunikation an der Offenen Universität von Katalonien
- ♦ Zertifizierung in Salesforce Marketing Cloud
- ♦ Zertifizierung in Heroku Architect Designer

Hr. Latino Maradiaga, Mario Alfonso

- ♦ Expert Architect bei mehreren Projekten im Infrastruktursektor
- ♦ Salesforce Expert Architect in einem privaten Sicherheitsunternehmen
- ♦ Salesforce Lead Architect in der Brauereibranche
- ♦ Salesforce Architect in der Bergbau- und Energiebranche
- ♦ Salesforce Engineer in der agrochemischen Industrie
- ♦ Salesforce Engineer im Gesundheitswesen und im pharmazeutischen Vertrieb
- ♦ Zertifizierung in Java SE 7 Fundamentals, Oracle Database: Program with PL/SQL, Salesforce Service Cloud Consultant, Salesforce App Builder, Salesforce Advanced Administrator

Fr. Grao Fernández, Ester

- ♦ IT-Beraterin für die Regierung von Aragonien
- ♦ SOA/BPM-Beraterin bei Avanttic
- ♦ J2EE-Programmiererin bei Grupo Acotelsa
- ♦ Masterstudiengang in Lehramt an der Universität Jaume I
- ♦ IT-Ingenieurin von der Universität Jaume I
- ♦ Technische Ingenieurin in Verwaltungsinformatik von der Universität von Zaragoza

Hr. Santos Arias, Millán

- ♦ Chief Technical Architect - CoE Salesforce bei NTT Data
- ♦ Salesforce Lead Architect - CoE Salesforce bei Everis Spain, S.L.U.
- ♦ Senior Analyst bei Everis Spain, S.L.U.
- ♦ Salesforce Certified Administrator
- ♦ Salesforce Certified Platform App Builder
- ♦ Salesforce Certified Platform Developer
- ♦ Salesforce Certified Sharing and Visibility Architect
- ♦ Salesforce Certified Data Architecture and Management Designer
- ♦ Salesforce Certified Development Lifecycle and Deployment Architect
- ♦ Salesforce Certified Identity and Access Management Architect
- ♦ Salesforce Certified Integration Architect
- ♦ Salesforce Certified System Architect
- ♦ Salesforce Certified Application Architect
- ♦ Salesforce Certified Experience Cloud Consultant
- ♦ Hochschulabschluss in Computertechnik an der Universität von Granada

Hr. Lobato Velázquez, Juan Manuel

- ♦ Expert Engineer Salesforce bei NTT Data
- ♦ Fortgeschrittener Ingenieur bei Serces Sistemas
- ♦ Senior Solutions Analyst
- ♦ Leitender Programmierer bei Cibernos OutSourcing
- ♦ Ingenieur bei Indra Software
- ♦ JAVA-Programmierer und J2EE-Anwendungsentwickler
- ♦ Zertifizierung in: Energy and Utilities Cloud, Platform App Builder, Vlocity CPQ Developer y Vlocity Omnistudio Developer

Fr. Nebra García, Sandra

- ♦ Salesforce Expert Engineer bei NTT Data
- ♦ Salesforce Lead Engineer in einem privatem Sicherheitsunternehmen
- ♦ Salesforce Engineer in der Brauereibranche
- ♦ Salesforce Engineer im Bausektor
- ♦ Salesforce Engineer im Nahrungsmittelsektor
- ♦ Front-End Developer und Full-Stack Developer bei Hiberus Tecnología
- ♦ Salesforce Certified Administrator
- ♦ Salesforce Certified Associate
- ♦ Hochschulabschluss in Wirtschaftsingenieurwesen mit Spezialisierung auf Industrieelektronik an der Universität von Zaragoza (EINA)





Fr. Latino Guido, Katherine Elisa

- ◆ Systemadministratorin in Unternehmen der Mobil- und Festnetztelefonie
- ◆ Fehlerbeseitigung bei Computersystemen in einem E-Learning-Unternehmen
- ◆ Höhere Berufsausbildung in Multiplattform-Anwendungsentwicklung am CPIFP Los Enlaces
- ◆ Berufsausbildung in Mikrocomputersysteme und -netze

Fr. Godoy Rodríguez, Tania

- ◆ Architektin im Salesforce Center of Excellence
- ◆ Salesforce Certified Administrator
- ◆ Salesforce Certified Platform App Builder
- ◆ Salesforce Certified Platform Developer I
- ◆ Salesforce Certified Sales Cloud Consultant
- ◆ Salesforce Certified Sharing and Visibility Designer
- ◆ Salesforce Certified OmniStudio Developer
- ◆ Salesforce Certified OmniStudio Consultant
- ◆ Salesforce Certified Industries CPQ Developer
- ◆ Salesforce Certified Net Zero Cloud Accredited Professional
- ◆ Spezialisierung in Telekommunikationssystemen an der Universität von Málaga
- ◆ Masterstudiengang in Telekommunikationstechnik an der Universität Carlos III
- ◆ Telekommunikationsingenieurin von der Universität von Málaga

05

Struktur und Inhalt

Dieser Universitätsabschluss vermittelt den Studenten die besten Entwicklungspraktiken, die von der Salesforce-Plattform empfohlen werden, erarbeitet von einem Lehrkörper mit umfassender Erfahrung in diesem Ökosystem. Der Lehrplan vertieft die verschiedenen bestehenden Modelle und stellt die entsprechenden Benutzerleitfäden zur Verfügung, so dass die Absolventen sie für ihre Implementierungsprozesse nutzen können. Der Lehrplan wird auch innovative Instrumente und eine Reihe von Strategien bereitstellen, die den Fachleuten helfen, im Bereich der Programmierung und Verteilung von Vorschlägen erfolgreich zu sein. In diesem Zusammenhang bietet das Programm modernste Methoden zur Gewährleistung der Sicherheit in Salesforce.



“

Ein Lehrplan, der Ihnen die innovativsten Strategien vermittelt, um robuste Lösungen zu entwerfen, die verschiedenen Sicherheitsanforderungen gerecht werden“

Modul 1. Programmierung in Salesforce

- 1.1. Software-as-a-Service-Modell von Salesforce
 - 1.1.1. Salesforce-Lizenzierungsmodell
 - 1.1.2. Governor Limits
 - 1.1.3. Vorhandene Arten von Umgebungen
- 1.2. Modell der Salesforce-Programmierung
 - 1.2.1. Metadaten-basiertes Modell
 - 1.2.2. Rahmenbedingungen für die Umsetzung
 - 1.2.3. Salesforce RESTful API-Modell
- 1.3. Mehrsprachigkeit und Standortbestimmung in Salesforce
 - 1.3.1. Standort
 - 1.3.2. Mehrsprachige Verwaltung in Salesforce
 - 1.3.3. Verwendung von Custom Labels in Salesforce
 - 1.3.4. Translation Workbench
- 1.4. Zugriffs- und Berechtigungsmanagement in Salesforce
 - 1.4.1. Profilverwaltung in Salesforce
 - 1.4.2. Verwaltung von Permission Sets in Salesforce
 - 1.4.3. Verwaltung von Permission Sets Groups in Salesforce
- 1.5. Modell der Datensatzsichtbarkeit in Salesforce
 - 1.5.1. Arten des Datenzugriffs
 - 1.5.2. Salesforce-Sichtbarkeitsmodell
 - 1.5.3. Gemeinsame Nutzung von Datensätzen im Sichtbarkeitsmodell
- 1.6. Multitenant-Architektur
 - 1.6.1. Multitenant-Architektur in Salesforce
 - 1.6.2. Anwendungsentwicklung in einer Multitenant-Architektur
 - 1.6.3. Interne Bearbeitung der Anfragen
- 1.7. Datenbankarchitektur in Salesforce
 - 1.7.1. Platform Data Layer
 - 1.7.2. Datenbankpartitionen pro Tenant
 - 1.7.3. Struktur der Datenbanken
 - 1.7.4. Verwaltung von Verzeichnissen und Verbindungen in Datenbanken

- 1.8. Experience Cloud und Communities für externe Nutzer
 - 1.8.1. Verwaltung der Sites in Experience Cloud
 - 1.8.2. Verwaltung von Berechtigungen und Einsicht für externe Benutzer
 - 1.8.3. Digital Experience Platform (DXP)
 - 1.8.4. Lightning Web Runtime (LWR)
- 1.9. AppExchange
 - 1.9.1. Ökosystem von Partners in Salesforce
 - 1.9.2. Arten von Paketen
 - 1.9.3. Independent Software Vendor (ISV)
- 1.10. Salesforce-Infrastruktur
 - 1.10.1. Management von Domains in Salesforce
 - 1.10.2. Hyperforce-Modell
 - 1.10.3. Edge-Network-Modell

Modul 2. Datenmodellierung in Salesforce

- 2.1. Datenmodell
 - 2.1.1. Datenmodellierung
 - 2.2.2. Objekt-Entitäts-Beziehungsmodell
 - 2.2.3. Best Practices für Informationsdesign und -modellierung: Logische und physische Ebene
- 2.2. Typologie der Objekte
 - 2.2.1. Objects - Standard, benutzerdefinierte und Datensatztypen
 - 2.2.2. Ereignismodellierung in Salesforce mit Platform Events
 - 2.2.3. Salesforce-Konfigurations- und Parametrisierungsobjekte
 - 2.2.4. Sonstige besondere Objekte
- 2.3. Dokumentenverwaltung in Salesforce
 - 2.3.1. DMS und ECM
 - 2.3.2. Arten von Dokumenten in Salesforce
 - 2.3.3. Fortgeschrittenes Wissensmanagement mit Knowledge Base
 - 2.3.4. Best Practices für die Modellierung der Dokumentenverwaltung
- 2.4. Erstellen von Objekten in Salesforce
 - 2.4.1. Best Practices für die Erstellung von UI aus
 - 2.4.2. Erweiterte Verwendung von Schema Builder
 - 2.4.3. API für die Objekterstellung

- 2.5. Qualität der Daten
 - 2.5.1. *Best Practices* für die Konfiguration von Attributen und Validierungsregeln
 - 2.5.2. Kontrolle von doppelten Datensätzen
 - 2.5.3. Berichte und andere Instrumente zur Bewertung und Überwachung der Qualität
- 2.6. Datenabfrage
 - 2.6.1. *Best Practices* für die Abfrage von Daten mit SOQL
 - 2.6.2. *Best Practices* für die Suche von Daten mit SOSL
 - 2.6.3. Dynamische SOQL- und SOSL-Konfiguration
- 2.7. Verwaltung von Datenbankänderungen mit DML-Operationen
 - 2.7.1. *Data Manipulation Language*
 - 2.7.2. Dynamic DML
 - 2.7.3. *Best Practices* für Bulk DML
 - 2.7.4. DML-Ausnahmebehandlung
 - 2.7.5. Für das Datenmanagement verfügbare APIs
- 2.8. Verarbeitung großer Datenmengen (LDV)
 - 2.8.1. Verwaltung von Tabellenindizes: Standard und personalisiert
 - 2.8.2. *Skinny Tables*
 - 2.8.3. Wie man *Data Skew* vermeidet
 - 2.8.4. Erweiterte Optimierung durch *Data Archiving* und *Query Plan Tool*
- 2.9. Schutz von Informationen
 - 2.9.1. Schutz von Informationen
 - 2.9.2. Verwaltung des Datenschutzes
 - 2.9.3. *Best Practices* zur Sicherstellung der Einhaltung von Datenschutzbestimmungen
- 2.10. Backup-Verwaltung
 - 2.10.1. Backup-Verwaltung
 - 2.10.2. Sicherung von Informationen
 - 2.10.3. *Best Practices* bei der Entwicklung von Datensicherungslösungen

Modul 3. Deklarative Programmierung in Salesforce

- 3.1. Deklarative Programmierung
 - 3.1.1. Deklarative Konstruktionswerkzeuge
 - 3.1.2. Formeln und Funktionen
 - 3.1.3. Funktionalitäten für Mail-Vorlagen
- 3.2. Deklaratives Seitendesign
 - 3.2.1. *Page Layout, Lightning App Builder* und *List Views*
 - 3.2.2. Konfigurieren von Schaltflächen, Links und Aktionen
 - 3.2.3. *In-App Guidance*
- 3.3. *Flow Builder*
 - 3.3.1. Verwaltung von *Flows*
 - 3.3.2. Lebenszyklus der *Flows*
 - 3.3.3. Wiederverwendung mit *Subflows*
 - 3.3.4. *Flow Interview: Paused Flows*
 - 3.3.5. *Flow Bulkification* in Transaktionen
- 3.4. *Screen Flow*
 - 3.4.1. Felder im *Screen Flow*
 - 3.4.2. *Flow* in Etappen
 - 3.4.3. Reaktive *Screen Flows*
- 3.5. Deklarative Automatisierungswerkzeuge
 - 3.5.1. *Autolaunched Flows-No Triggered*
 - 3.5.2. *Record Triggered Flows*
 - 3.5.3. *Platform Event Triggered Flows*
- 3.6. Orchestrierung von *Flows*
 - 3.6.1. Orchestrierung von *Flows*
 - 3.6.2. *Autolaunched Orchestration-No Triggered*
 - 3.6.3. *Record-Triggered Orchestration*
- 3.7. Behandlung von Ausnahmen in *Flows*
 - 3.7.1. Testen mit *Flow Builder*
 - 3.7.2. Fehlerbehebung
 - 3.7.3. Überwachungsmöglichkeiten
 - 3.7.4. Framework für die Behandlung von Ausnahmen

- 3.8. Zulassungsprozesse
 - 3.8.1. *Approval Process Wizard* und erforderliche Konfiguration
 - 3.8.2. Konfiguration der Zulassungsphasen
 - 3.8.3. Grenzen und Überlegungen
- 3.9. *External Services* und *Outbound Message*
 - 3.9.1. Konfiguration von *External Service & Flow Action*
 - 3.9.2. *Outbound Message*: Konfiguration, Benachrichtigung und Überwachung
 - 3.9.3. *External Service* und *Outbound Message*
- 3.10. Deklarative analytische Werkzeuge
 - 3.10.1. *Custom Record Types*
 - 3.10.2. Erstellung von *Reports*
 - 3.10.3. Aufbau des *Dashboard*
 - 3.10.4. Beschränkungen von *Reports* und *Dashboard*

Modul 4. Programmierung in APEX für Salesforce

- 4.1. Entwicklungstools
 - 4.1.1. *Developer Console*
 - 4.1.2. Empfohlene IDEs für die Apex-Programmierung
 - 4.1.3. Salesforce-Erweiterungen für VS Code
 - 4.1.4. *Code Builder*
- 4.2. Programmierung mit Apex
 - 4.2.1. Apex-Programmierung
 - 4.2.2. Apex-Transaktion, Methoden und statische Variablen
 - 4.2.3. Handhabung von Ausnahmen in Apex
- 4.3. Datenzugriff von Apex aus
 - 4.3.1. SOQL-Strukturen und -Grundlagen in Apex
 - 4.3.2. Apex-Variablen in SOQL und SOSL
 - 4.3.3. DML-Deklarationen gegen Methoden der Klasse *System Database*
- 4.4. Apex Triggers
 - 4.4.1. *Triggers* und Reihenfolge der Ausführung
 - 4.4.2. Kontextbezogene Variablen
 - 4.4.3. *Bulk Triggers* und *Best Practices*
- 4.5. Asynchrones Apex
 - 4.5.1. *Future Apex*
 - 4.5.2. Queueable Apex und Ausführungsverkettung
 - 4.5.3. *Apex Scheduler*
- 4.6. *Batch Apex*
 - 4.6.1. Batch-Apex-Architektur
 - 4.6.2. Batch Jobs
 - 4.6.3. Batch-Apex-Beschränkungen
- 4.7. Sicherheit in Apex
 - 4.7.1. Apex-Anwendungen: Sicherheitsmethoden
 - 4.7.2. Digitale Erfahrungen: Methoden für Websites
 - 4.7.3. *Apex Crypto*-Schicht
- 4.8. Gemeinsame Nutzung von Datensätzen über Apex
 - 4.8.1. Von Apex verwaltete gemeinsame Nutzung
 - 4.8.2. Neuberechnung der Aufteilung mit Apex
 - 4.8.3. Zuweisung von Gebieten zu *Opportunities*
- 4.9. Apex Callouts
 - 4.9.1. SOAP-Dienste: Über WSDL generierte *Proxy*-Klassen
 - 4.9.2. HTTP-Aufrufe: Methoden der HTTP-Klasse
 - 4.9.3. Einschränkungen von *Apex Callouts*
- 4.10. Ausführung von Unit-Tests
 - 4.10.1. Muster für die Ausführung von Unit-Tests
 - 4.10.2. Isolierung der Testdaten
 - 4.10.3. Simulierte und Hilfsobjekte: *System.StubProvider*-Schnittstelle
 - 4.10.4. *Best Practices* bei der Entwicklung von Testklassen

Modul 5. Programmierung von Benutzeroberflächen in Salesforce

- 5.1. VisualForce
 - 5.1.1. Erstellung von VF-Seiten und häufigste Tags
 - 5.1.2. Standardtreiber und Standardlistentreiber
 - 5.1.3. Standardtreiber
 - 5.1.4. Fehlerbehandlung
 - 5.1.5. *Best Practices*
- 5.2. *Aura Component*
 - 5.2.1. Erstellung von Aura-Komponenten und Tags
 - 5.2.2. *Lightning Data Service*
 - 5.2.3. Kommunikation mit dem Apex-Server
 - 5.2.4. Zusammenstellung und Kommunikation zwischen Komponenten
 - 5.2.5. Fehlerbehandlung
 - 5.2.6. *Best Practices*
- 5.3. *Lightning Web Components*
 - 5.3.1. Erstellung von LWC und häufigste Tags
 - 5.3.2. Ereignisverwaltung und *Lifecycle Hooks*
 - 5.3.3. Kommunikation mit dem Apex-Server
 - 5.3.4. Seitenaufbau und Kommunikation zwischen Komponenten
 - 5.3.5. Shadow DOM
 - 5.3.6. Nutzung der Möglichkeiten von Mobiltelefonen
 - 5.3.7. Fehlerbehandlung und Fehlersuche
 - 5.3.8. *Best Practices*
- 5.4. *Lightning Message Service*
 - 5.4.1. Erstellung von *Message Channel* und Konfiguration des Scope
 - 5.4.2. Veröffentlichen einer Nachricht in einem *Message Channel*
 - 5.4.3. Abonnement für einen *Message Channel*
 - 5.4.4. Beschränkungen
- 5.5. Koexistenz von *UI-Frameworks*
 - 5.5.1. Visualforce-Interoperabilität - Aura-Komponente
 - 5.5.2. Interoperabilität Visualforce - LWC
 - 5.5.3. Interoperabilität LWC - Aura-Komponente

- 5.6. *Lightning Design System*
 - 5.6.1. Plattformen
 - 5.6.2. *Lightning Design System*
 - 5.6.3. *Best Practices*
- 5.7. *Testing UI*
 - 5.7.1. Jasmine
 - 5.7.2. Jest
 - 5.7.3. UTAM
 - 5.7.4. Selenium
- 5.8. Qualität des Codes
 - 5.8.1. Konfiguration
 - 5.8.2. Linter
 - 5.8.3. RetireJS
- 5.9. Troubleshooting
 - 5.9.1. Chrome Entwickler-Konsole
 - 5.9.2. Identifizierung von Leistungsproblemen
 - 5.9.3. Identifizierung von Netzwerkproblemen
- 5.10. *Mobile SDK*
 - 5.10.1. Modalitäten der Entwicklung
 - 5.10.2. Entwicklung hybrider Applikationen
 - 5.10.3. Entwicklung nativer Anwendungen (Android)

Modul 6. OmniStudio-Framework

- 6.1. Omnistudio-Architektur
 - 6.1.1. Omnistudio-Architektur
 - 6.1.2. Omnistudio-Komponentenebenen
 - 6.1.3. Omnistudio-Versionstypen
- 6.2. Systemverwaltung und Konfiguration
 - 6.2.1. Installieren und Aktualisieren von Omnistudio
 - 6.2.2. Lizenzen und Genehmigungen von OmniStudio
 - 6.2.3. Konfigurieren von Benutzeroberflächen und Implementierungen

- 6.3. Dataraptors
 - 6.3.1. DataRaptor
 - 6.3.2. Arten von Dataraptors
 - 6.3.3. Von Dataraptors zurückgegebene Datentypen
 - 6.3.4. Suche und Sicherheit bei Dataraptors
 - 6.3.5. Dataraptor-Aufrufmethoden
 - 6.3.6. *Best Practices* für Dataraptors
- 6.4. *Integration Procedures*
 - 6.4.1. *Integration Procedures*
 - 6.4.2. Arten von Aktionen bei *Integration Procedures*
 - 6.4.3. Suche und Sicherheit bei *Integration Procedures*
 - 6.4.4. Integrationsverfahren bei *Integration Procedures*
 - 6.4.5. Verwaltung von Fehlern bei *Integration Procedures*
 - 6.4.6. *Best Practices* für *Integration Procedures*
- 6.5. Flexcards
 - 6.5.1. Flexcards
 - 6.5.2. Elemente für Flexcards
 - 6.5.3. Verwaltung von Flexcards
 - 6.5.4. Flexcard Designer
 - 6.5.5. Fehlersuche und Tests in Flexcards
 - 6.5.6. *Best Practices* für Flexcards
- 6.6. Omniscrpts
 - 6.6.1. Omniscrpts
 - 6.6.2. Elemente für Omniscrpts
 - 6.6.3. Verwaltung von Omniscrpts
 - 6.6.4. Omniscrpts Designer
 - 6.6.5. Fehlersuche und Tests in Omniscrpts
 - 6.6.6. *Best Practices* für Omniscrpts

- 6.7. *Business Rules Engine*
 - 6.7.1. *Business Rules Engine*
 - 6.7.2. Matrix-Entscheidung
 - 6.7.3. *Decision Tables*
 - 6.7.4. *Expression Sets*
 - 6.7.5. Integrationen von Business Rules
 - 6.7.6. Migration von *Calculation-Matrix* und *Calculation Procedures*
- 6.8. Tracking Service
 - 6.8.1. Tracking Service
 - 6.8.2. Tracking-Service-Nutzungseinstellungen
 - 6.8.3. Elemente für die Nutzung des Tracking Service
- 6.9. OmniAnalytics Overview
 - 6.9.1. OmniAnalytics
 - 6.9.2. Nutzungseinstellungen für OmniAnalytics
 - 6.9.3. OmniAnalytics-Ergebnisse für Google Analytics
- 6.10. IDX Workbench (Deployment Tool)
 - 6.10.1. Installation von IDX Workbench
 - 6.10.2. Konfiguration und Migration von Metadaten
 - 6.10.3. Validierung der Metadaten-Migration

Modul 7. APIs und Integrationen in Salesforce

- 7.1. Integration in Salesforce
 - 7.1.1. Ausgehende Integration von Salesforce
 - 7.1.2. Eingehende Salesforce-Integrationen
 - 7.1.3. Integrationen zwischen Salesforce und Salesforce
 - 7.1.4. Salesforce API-Bibliotheken
- 7.2. *External Services*
 - 7.2.1. Open-API-Standardspezifikation
 - 7.2.2. *External Credentials*
 - 7.2.3. *Named Credentials*

- 7.3. *Connected Apps*
 - 7.3.1. *Connected App*
 - 7.3.2. *Integrations-Scopes*
 - 7.3.3. *Autorisierung von Integrationen und Identitätsmanagement*
 - 7.3.4. *Sicherheitsrichtlinien bei Connected Apps*
- 7.4. *SOAP API*
 - 7.4.1. *Enterprise WSDL*
 - 7.4.2. *Partner WSDL*
 - 7.4.3. *Apex WSDL*
 - 7.4.4. *Metadata WSDL*
- 7.5. *Synchrone REST-Integrationen*
 - 7.5.1. *REST API in Salesforce*
 - 7.5.2. *Connect REST API in Salesforce*
 - 7.5.3. *Dokumentenverwaltung in Salesforce über API*
- 7.6. *API für die Datenmigration*
 - 7.6.1. *Bulk API*
 - 7.6.2. *Konfiguration und Ausführung der Massendatenmigration über Bulk API*
 - 7.6.3. *Verarbeitung von Massendaten*
 - 7.6.4. *Bulk API 1.0 vs. Bulk API 2.0*
- 7.7. *Ereignisbasierte Integration*
 - 7.7.1. *Platform Event Bus*
 - 7.7.2. *Streaming API in Salesforce*
 - 7.7.3. *Pub/Sub API in Salesforce*
 - 7.7.4. *Change Data Capture*
 - 7.7.5. *Ereignisabonnement von externen Systemen mit cometD Framework*
- 7.8. *Integration zur Konfiguration und Entwicklung*
 - 7.8.1. *Tooling API*
 - 7.8.2. *Metadata API*
 - 7.8.3. *Dynamische Org-Verwaltung auf Basis von Tooling API*

- 7.9. *Datensynchronisation zwischen Orgs*
 - 7.9.1. *Datensynchronisierungsmuster zwischen Orgs*
 - 7.9.2. *Salesforce Connect mit Cross-Org-Adapter*
 - 7.9.3. *Open Data Protocol (oData)*
 - 7.9.4. *GraphQL mit oData*
- 7.10. *Integrationsmuster*
 - 7.10.1. *Remote Process Invocation-Request and Reply*
 - 7.10.2. *Remote Process Invocation-Fire and Forget*
 - 7.10.3. *Batch Data Synchronization*
 - 7.10.4. *Remote Call-In*
 - 7.10.5. *UI Update Based on Data Changes*
 - 7.10.6. *Data Virtualization*

Modul 8. Fortgeschrittene Programmierung in Salesforce

- 8.1. *Dynamisches Apex*
 - 8.1.1. *Dynamischer Zugriff auf Objekt- und Feldefinition*
 - 8.1.2. *Dynamische SOQL*
 - 8.1.3. *Dynamische DML*
- 8.2. *Platform Cache*
 - 8.2.1. *Platform Cache Nützlichkeit*
 - 8.2.2. *Org Cache*
 - 8.2.3. *Session Cache*
 - 8.2.4. *Best Practices*
- 8.3. *Platform Event Bus*
 - 8.3.1. *Platform Event Bus. Nützlichkeit*
 - 8.3.2. *Veröffentlichung eines Ereignisses*
 - 8.3.3. *Anmeldung zu einem Ereignis*
- 8.4. *SOLID-Prinzipien*
 - 8.4.1. *Single Responsibility*
 - 8.4.2. *Open-Closed*
 - 8.4.3. *Liskov Substitution*
 - 8.4.4. *Interface Segregation*
 - 8.4.5. *Dependency-Inversion*

- 8.5. *Apex Enterprise Pattern*
 - 8.5.1. *Separation of Concerns (SOC)*
 - 8.5.2. *Auswahlschicht*
 - 8.5.3. *Domänenschicht*
 - 8.5.4. *Dienstschicht*
- 8.6. *Trigger-Framework*
 - 8.6.1. *Trigger-Framework*
 - 8.6.2. *O'Hara*
 - 8.6.3. *Dan Appleman*
 - 8.6.4. *Hari Krishnan*
 - 8.6.5. *Scot Wells*
- 8.7. *Fehler-Framework*
 - 8.7.1. *Fehler-Framework*
 - 8.7.2. *Fehlererkennung*
 - 8.7.3. *Plattformereignis zur Protokollierung des Fehlers*
- 8.8. *Framework-Automatisierungen*
 - 8.8.1. *Automatisierungstools*
 - 8.8.2. *Entwurf eines Frameworks für die Automatisierung*
 - 8.8.3. *Technische Aspekte*
- 8.9. *Entwicklungstools*
 - 8.9.1. *Code Analyzer*
 - 8.9.2. *Apex Replay*
 - 8.9.3. *Apex Log Analyzer*
 - 8.9.4. *Explain Plan*
- 8.10. *Troubleshooting von Grenzwerten in Apex*
 - 8.10.1. *CPU Time*
 - 8.10.2. *Too Many SOQL*
 - 8.10.3. *Heap Size*

Modul 9. Sicherheit in Salesforce

- 9.1. *Sicherheit in Apex*
 - 9.1.1. *Sicherheit in Apex*
 - 9.1.2. *Best Practices für sicheren und angriffsgeschützten Code*
 - 9.1.3. *Datenverschlüsselung bei der Übertragung und im Ruhezustand*
- 9.2. *Sicherheit in Visuaforce*
 - 9.2.1. *Sicherheit in Visuaforce*
 - 9.2.2. *Sicherheit im Visualforce-Framework*
 - 9.2.3. *Best Practices für sicheren Code in Visualforce*
- 9.3. *Sicherheit in Aura*
 - 9.3.1. *Sicherheit im Aura-Framework*
 - 9.3.2. *Praktische Beispiele für Sicherheitsschutz und Sicherheitsbeweise in Aura*
 - 9.3.3. *Best Practices für sicheren Code in Aura*
- 9.4. *Sicherheit in Lightning Web Components*
 - 9.4.1. *Sicherheit im LWC-Framework*
 - 9.4.2. *Praktische Beispiele für Sicherheitsschutz und Sicherheitsbeweise in LWC*
 - 9.4.3. *Best Practices für sicheren Code in LWC*
- 9.5. *User Access Management*
 - 9.5.1. *Benutzerprofilerstellung und Lizenzierung (Profile & Custom permission)*
 - 9.5.2. *Rollenhierarchie und Territorialmodell*
 - 9.5.3. *Teammanagement (Fall, Konto und Verkaufschance)*
 - 9.5.4. *Warteschlangen und Benutzergruppen*
 - 9.5.5. *Best Practices für den Zugriff externer Benutzer*
- 9.6. *Sicherheit auf Registrierungsebene*
 - 9.6.1. *OWD, Sharing Rules, ARSDR y und manuelles Sharing. Best Practices*
 - 9.6.2. *Sperrung von Datensätzen*
 - 9.6.3. *Implicit Sharing und Apex Sharing Reasons*
 - 9.6.4. *Data Masking*
- 9.7. *Sicherheit vor Ort*
 - 9.7.1. *Fortgeschrittenes Profiling-Modell*
 - 9.7.2. *Best Practices für Dynamic forms*
 - 9.7.3. *FLS-Kontrolle und Datensatzsichtbarkeit in Apex und SOQL*

- 9.8. Benutzerauthentifizierung und Salesforce-Zugangsberechtigung
 - 9.8.1. Authentifizierungsmethoden (U/P, SSO und *Delegated Authentication*)
 - 9.8.2. *Best Practices* bei Login Flows
 - 9.8.3. *Best Practices* für den internen und externen Zugriff (*Experience Cloud*)
 - 9.8.4. Konfiguration des sicheren Zugriffs von externen Anwendungen auf Salesforce
 - 9.8.5. *Best Practices* in OAuth Flows für die Zugangsberechtigung
- 9.9. Überwachung und Sicherheitsrichtlinien
 - 9.9.1. *Salesforce Shield* - Erweiterte Anwendungsfälle
 - 9.9.2. *Salesforce Shield* - *Field Audit Trail*
 - 9.9.3. *Salesforce Shield* - *Transaction Security Policies*
- 9.10. Sicherheit auf Netzwerkebene
 - 9.10.1. HTTPS & TLS - Erweiterte Einstellungen in Salesforce
 - 9.10.2. 1-way & 2-way SSL Flow. Konfiguration und Anwendbarkeit
 - 9.10.3. *Best Practices* bei der Netzwerkkonfiguration

Modul 10. *Application Lifecycle Management (ALM)* in Salesforce

- 10.1. Salesforce-Entwicklungsmodelle
 - 10.1.1. Deklaratives Entwicklungsmodell
 - 10.1.2. Entwicklungsmodell basierend auf Org
 - 10.1.3. Quellcode-basiertes Entwicklungsmodell
 - 10.1.4. Paketbasiertes Entwicklungsmodell
- 10.2. Verzweigungsstrategie
 - 10.2.1. GitFlow und seine Entwicklung angepasst an Salesforce
 - 10.2.2. Verzweigungsstrategie beschränkt auf User Stories
 - 10.2.3. Verzweigungsstrategie auf der Grundlage von *Releases* und *Major Releases*
 - 10.2.4. Fokus auf das Thema
- 10.3. Quellcode-Repositoryn (VCS)
 - 10.3.1. Quellcode-Repositoryn (VCS)
 - 10.3.2. Rollen und Verantwortlichkeiten
 - 10.3.3. Git-Konfiguration und Befehlsausführung

- 10.4. Salesforce DX
 - 10.4.1. Befehlskonsole
 - 10.4.2. Verbindung und Konfiguration von Orgs
 - 10.4.3. Konfiguration und Verwaltung von *Plugins*
 - 10.4.4. Erstellen eines eigenen SFDX-*Plugins*
- 10.5. Programmierung auf der Grundlage von Metadaten
 - 10.5.1. Salesforce-Metadaten
 - 10.5.2. Abdeckung der Metadaten in Salesforce
 - 10.5.3. Einsätze auf der Grundlage einer *Manifest*-Datei
- 10.6. Scratch Orgs
 - 10.6.1. Scratch Orgs
 - 10.6.2. Konfiguration der Scratch Orgs
 - 10.6.3. Org-Shape-Generierung
 - 10.6.4. Programmierung auf der Grundlage von Scratch Orgs
- 10.7. Erstellung und Verteilung von Paketen
 - 10.7.1. Paketstruktur in Salesforce
 - 10.7.2. Verteilung der Pakete
 - 10.7.3. Pakete der ersten und zweiten Generation
- 10.8. CI/CD-Werkzeuge
 - 10.8.1. Kontinuierliche Integration in Salesforce
 - 10.8.2. Phasen eines kontinuierlichen Integrationsprozesses in Salesforce
 - 10.8.3. Tools zur statischen Codeanalyse
 - 10.8.4. Automatisierungstools
- 10.9. DevOps Center
 - 10.9.1. DevOps Center in Salesforce
 - 10.9.2. Bereitstellungsmethodik mit Salesforce DevOps Center
 - 10.9.3. Konfigurieren und Ausführen von Bereitstellungen mit Salesforce DevOps Center
- 10.10. Veröffentlichung von Paketen in AppExchange
 - 10.10.1. Management von *Namespaces* in Salesforce
 - 10.10.2. Prozess der Veröffentlichung von Paketen in AppExchange
 - 10.10.3. *Salesforce Security Review*

06 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.



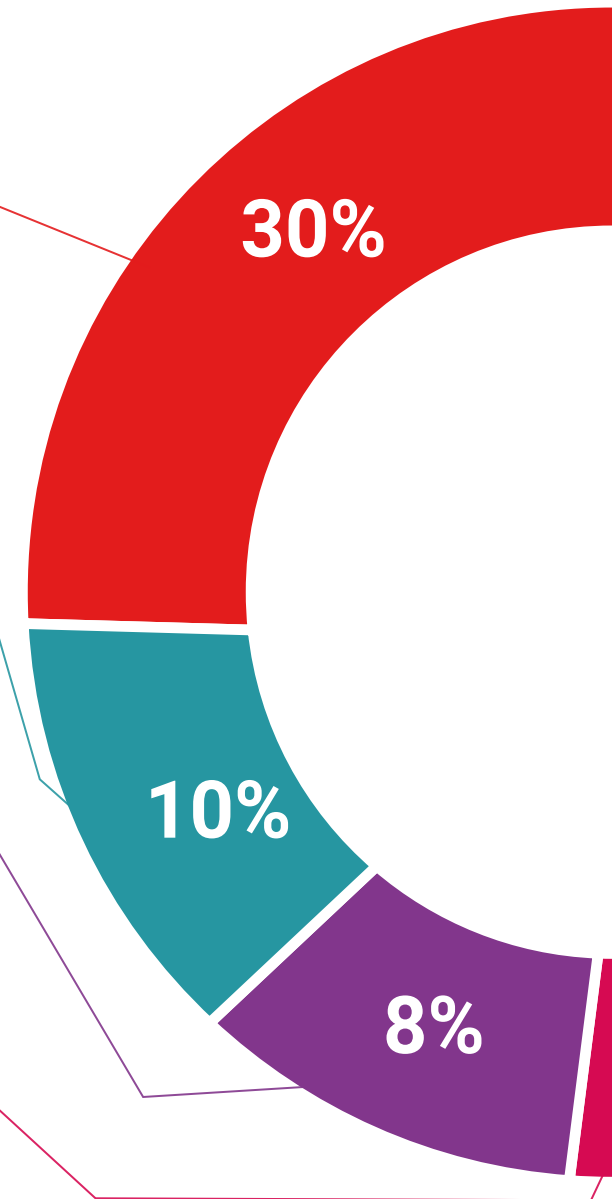
Übungen für Fertigkeiten und Kompetenzen

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.



07

Qualifizierung

Der Privater Masterstudiengang in Programmierung in Salesforce garantiert neben der präzise-
sten 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 **Privater Masterstudiengang in Programmierung in Salesforce** enthält das vollständigste und aktuellste Programm auf dem Markt.

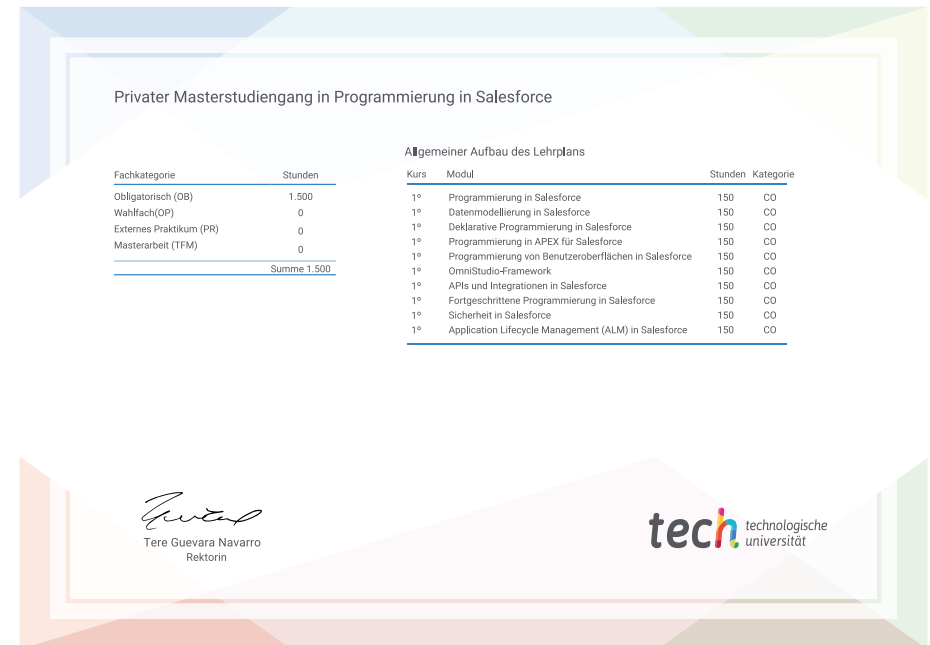
Sobald der Student die Prüfungen bestanden hat, erhält er/sie per Post* mit Empfangsbestätigung das entsprechende Diplom, ausgestellt von der **TECH Technologischen Universität**.

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: **Privater Masterstudiengang in Programmierung in Salesforce**

Modalität: **online**

Dauer: **12 Monate**



*Haager Apostille. Für den Fall, dass der Student die Haager Apostille für sein Papierdiplom beantragt, wird TECH EDUCATION die notwendigen Vorkehrungen treffen, um diese gegen eine zusätzliche Gebühr zu beschaffen.

zukunft

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

tech technologische
universität

Privater Masterstudiengang
Programmierung
in Salesforce

- » Modalität: online
- » Dauer: 12 Monate
- » Qualifizierung: TECH Technologische Universität
- » Zeitplan: in Ihrem eigenen Tempo
- » Prüfungen: online

Privater Masterstudiengang Programmierung in Salesforce