


# Master's Degree Salesforce Programming



the developer  
active = modifier  
(modifier\_ob)) # modifier  
= 0  
.selected\_objects[0]  
cs[one.name].select = 1  
please select exactly two objects,  
OPERATOR CLASSES



## Master's Degree Salesforce Programming

- » Modality: online
- » Duration: 12 months
- » Certificate: TECH Global University
- » Credits: 60 ECTS
- » Schedule: at your own pace
- » Exams: online

Website: [www.techtute.com/us/information-technology/master-degree/master-salesforce-programming](http://www.techtute.com/us/information-technology/master-degree/master-salesforce-programming)

# Index

01

Introduction

---

*p. 4*

02

Objectives

---

*p. 8*

03

Skills

---

*p. 14*

04

Course Management

---

*p. 18*

05

Structure and Content

---

*p. 22*

06

Methodology

---

*p. 32*

07

Certificate

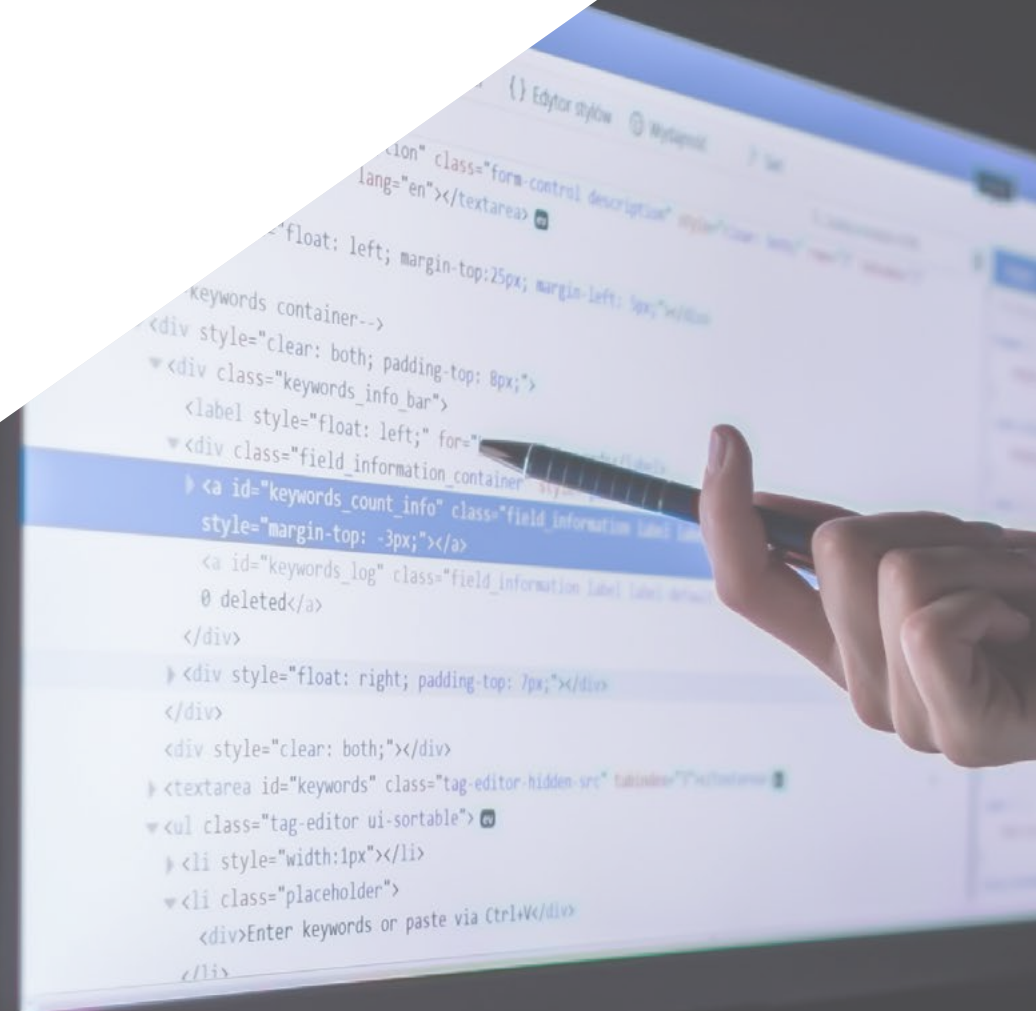
---

*p. 40*

# 01

# Introduction

In the context of IT, Salesforce has become a powerful tool for innovating the customer experience through easy-to-use, cloud-integrated enterprise applications. This technology allows developers to create customized programs for institutions to adapt their Customer Relationship Management solutions according to their specific needs. Among its advantages is the automation of a wide range of business processes, from lead management to customer service and project management. Given the growing importance of this platform, TECH is launching a university degree that will provide the most cutting-edge strategies for complex implementation solutions. And all in a convenient 100% online format!



“

*With this Master's Degree, 100% online, you will master Salesforce and get advanced skills to develop fully customized applications"*

Declarative Programming in Salesforce is a fundamental process for IT, as it makes it easy for users to configure and customize the platform using visual, rules-based tools. In this way, administrators configure Salesforce based on business requirements without having to rely on developers to write custom code. This, in turn, enables teams to implement changes more quickly and reduces the costs associated with both developing and maintaining individualized solutions. In addition, it fosters greater autonomy and capacity on the part of users to adapt to the changing needs of organizations.

In this scenario, TECH has created an innovative Master's Degree in Salesforce Programming. Its main objective is to provide students with the best development practices in this platform, as well as the most sophisticated techniques to lead them to success in the field of programming. To this end, the academic path will delve into issues ranging from different log visibility models or multitenant architectures to community management for external consumers. The curriculum will also provide programmers with a set of declarative automation tools to facilitate rapid iteration and maintainability. In line with this, the didactic materials will analyze different measures to promote security in Apex, Visualforce and Aura, among others.

On the other hand, the methodology of this university degree stands out for its 100% online mode, adapting to the needs of busy professionals who set the goal of advancing in their careers. In addition, it uses the Relearning system, based on the repetition of key concepts to fix knowledge and facilitate learning. In this way, the combination of flexibility and a robust pedagogical approach makes the program highly accessible. In addition, professionals will have access to a comprehensive virtual library full of multimedia resources in different formats to ensure dynamic learning.

This **Master's Degree in Salesforce Programming** contains the most complete and up-to-date program on the market. The most important features include:

- ♦ Development of practical cases presented by experts in Salesforce Programming
- ♦ The graphic, schematic and practical contents of the program provide complete and practical information on those disciplines that are essential for professional practice
- ♦ Practical exercises where the self-assessment process can be carried out to improve learning
- ♦ Its special emphasis on innovative methodologies
- ♦ Theoretical lessons, questions to the expert, debate forums on controversial topics, and individual reflection assignments
- ♦ Content that is accessible from any fixed or portable device with an Internet connection



*A degree that will keep you abreast of the latest trends and updates in the Salesforce platform to elevate your programming skills"*

“

*TECH provides you with a wide variety of learning materials in a variety of formats including how-to videos, supplemental readings, and case studies”*

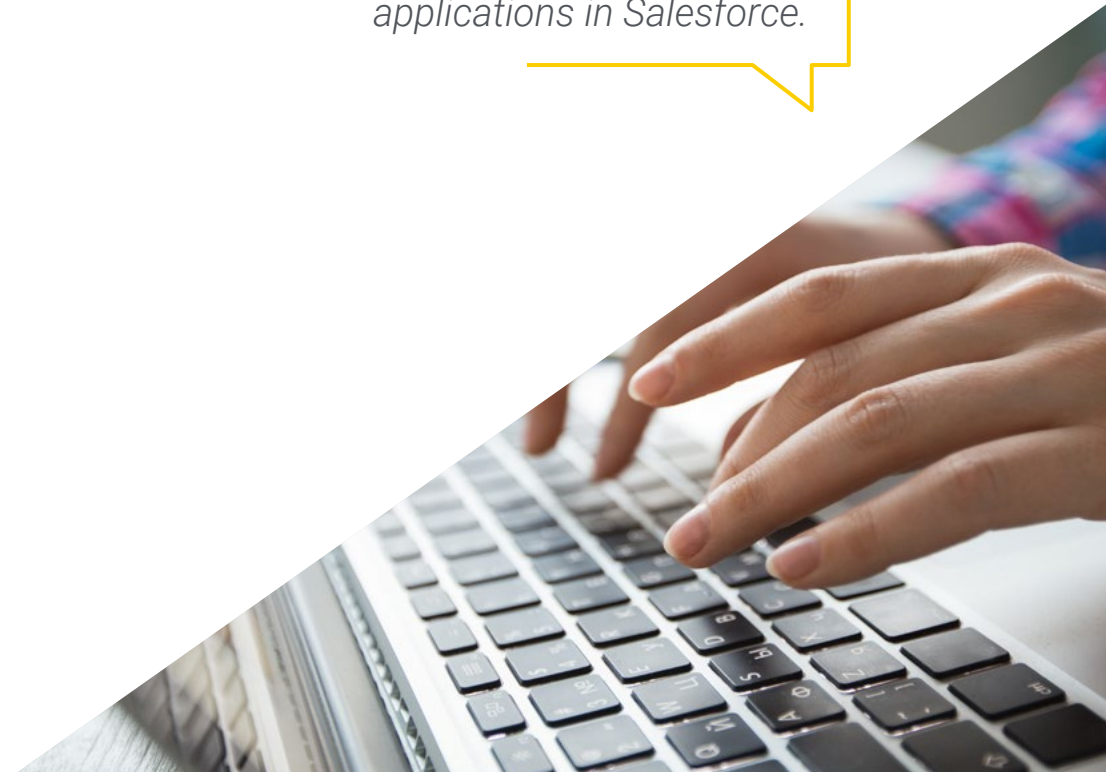
The program’s teaching staff includes professionals from the field who contribute their work experience to this educational program, as well as renowned specialists from leading societies and prestigious universities.

The multimedia content, developed with the latest educational technology, will provide the professional with situated and contextual learning, i.e., a simulated environment that will provide immersive education programmed to learn in real situations.

This program is designed around Problem-Based Learning, whereby the professional must try to solve the different professional practice situations that arise during the academic year. For this purpose, the students will be assisted by an innovative interactive video system created by renowned and experienced experts.

*Using the revolutionary Relearning methodology, you will effectively integrate all concepts to successfully achieve the results you want.*

*You'll delve into the OmniStudio Framework to efficiently build, customize and manage business applications in Salesforce.*



# 02 Objectives

Upon completion of this Master's Degree, graduates will have gained in-depth knowledge of the Salesforce platform, including both its functionality and development capabilities. In turn, computer scientists will improve their programming skills to design highly customized applications using both Apex and Visualforce. In this regard, they will perform effective data management by executing SOQL and SOSL queries. All this will allow students to solve technical problems with immediacy, as well as to diagnose or solve bugs in the language codes. In this way, they will be qualified to take advantage of the opportunities offered by the computer industry.







“

*You will incorporate agile software development methodologies into your daily practice and apply these approaches to optimally manage the creation of applications”*



## General Objectives

---

- ◆ Lay the foundation for the most advanced concepts focused on programming on the Salesforce platform
- ◆ Establish the Salesforce ecosystem as a development platform and determine the relevant boundaries
- ◆ Generate specialized knowledge of how the metadata-driven development model used on the Salesforce platform works
- ◆ Analyze how Salesforce works with data and multitenant architecture
- ◆ Develop Salesforce database fundamentals
- ◆ Analyze all the tools and advanced options for creating objects in Salesforce
- ◆ Promote the correct use of APIs and data manipulation tools
- ◆ Identify the key tools and resources for programming in Apex
- ◆ Introduce testing tools to ensure the proper functioning of developed components
- ◆ Promote the use of good development practices aligned with the standards defined by Salesforce



*You will implement in your projects security systems in Salesforce environments, including the configuration of profiles, to ensure regulatory compliance"*





## Specific Objectives

---

### Module 1. Salesforce Programming

- ◆ Compile the types of environments that can be used for development on the Salesforce platform
- ◆ Analyze how the database architecture on which Salesforce is based works in order to make developments within the platform more flexible
- ◆ Evaluate how the record visibility model and record sharing between application users works
- ◆ Establish the existing permissions model in the platform to grant the necessary accesses to the users that will use the developments that we make in Salesforce

### Module 2. Data Modeling in Salesforce

- ◆ Analyze, in an advanced way, all the Salesforce tools for the creation of objects and attributes in Salesforce
- ◆ Delve into the best practices for data manipulation in Salesforce
- ◆ Examine, at a specialized level, best techniques in the use of SOQL and SOSL query languages
- ◆ Solve performance issues when working with large volumes of data

### Module 3. Declarative Programming in Salesforce

- ◆ Generate specialized knowledge on formulas and custom functions on the platform
- ◆ Propose page layout solutions using declarative tools
- ◆ Develop and evaluate workflows taking into account limits and best practices
- ◆ Generate automated configurations to track approval processes

### Module 4. Programming in APEX for Salesforce

- ◆ Master Apex development tools
- ◆ Evaluate the differences between Java and Apex, delving into the features that distinguish Apex
- ◆ Analyze the possibilities of accessing data from Apex
- ◆ Develop synchronous and asynchronous processes, understanding their limitations and applying best practices

### Module 5. UI Programming in Salesforce

- ◆ Delve into the various user interface (UI) development frameworks available in Salesforce
- ◆ Delve into event management and how custom components communicate within the application
- ◆ Identify the various tools dedicated to ensuring the quality of the developed code
- ◆ Explore the different frameworks used for unit and end-to-end test generation

### Module 6. OmniStudio Framework

- ◆ Develop specialized knowledge on the components included in Omnistudio
- ◆ Analyze and specify the use of each of Omnistudio's components
- ◆ Establish good practices for the use of these components
- ◆ Delve into the connection between the different Omnistudio components to develop the solution in the most effective and scalable way

### Module 7. API and Integrations in Salesforce

- ◆ Specify the type of configuration we need to run in Salesforce to enable inbound integrations and to narrow down your security policies
- ◆ Demonstrate how each of the API types in the Salesforce platform works
- ◆ Develop different practical cases of exporting and importing data in bulk using the Bulk API
- ◆ Present the different types of existing events and identify the type of information that is sent in each of them. In turn, define how we can subscribe to the Salesforce platform event bus to capture the events published in the channel we are interested in

### Module 8. Advanced Programming in Salesforce

- ◆ Evaluate different Trigger Frameworks with emphasis on the characteristics of each one and the benefits they provide
- ◆ Design an error handling mechanism, in which any uncontrolled error that occurs during the execution of the application's custom code is recorded in detail
- ◆ Examine the SOLID principles that enable the creation of efficient, robust and scalable software
- ◆ Generate expertise in Apex Enterprise Patterns that define how to organize Apex code to ensure maintainability, scalability and robustness





### Module 9. Salesforce Security

- ◆ Delve into the best practices of the most popular Salesforce development frameworks (Apex, Visualforce, Aura, LWC)
- ◆ Analyze the inverted security pyramid on which the Salesforce security model is based
- ◆ Compile all available measures to protect the solution against external attacks
- ◆ Correctly monitor application access operations for both users and applications

### Module 10. Application Lifecycle Management (ALM) in Salesforce

- ◆ Determine the recommended environment roadmap for developing, testing, and deploying changes to live environments
- ◆ Evaluate the different source code repository branching strategies on which the team can rely to work in a collaborative environment
- ◆ Examine the tools available in Salesforce DX for exchanging metadata and executing operations against Salesforce environments
- ◆ Generate command-driven development environments based on the fundamentals of Scratch Orgs

# 03 Skills

Thanks to this university program, graduates will acquire advanced skills for the development of customized applications in Salesforce, while mastering the Apex language. This will enable IT professionals to meet the specific needs of organizations, including the creation of individualized objects, validation rules and workflows. At the same time, professionals will effectively manage data on this platform and will be able to develop processes such as importing, exporting and transforming information. They will also be characterized by a problem-solving approach. You will be able to solve Visualforce code errors with dexterity.



“

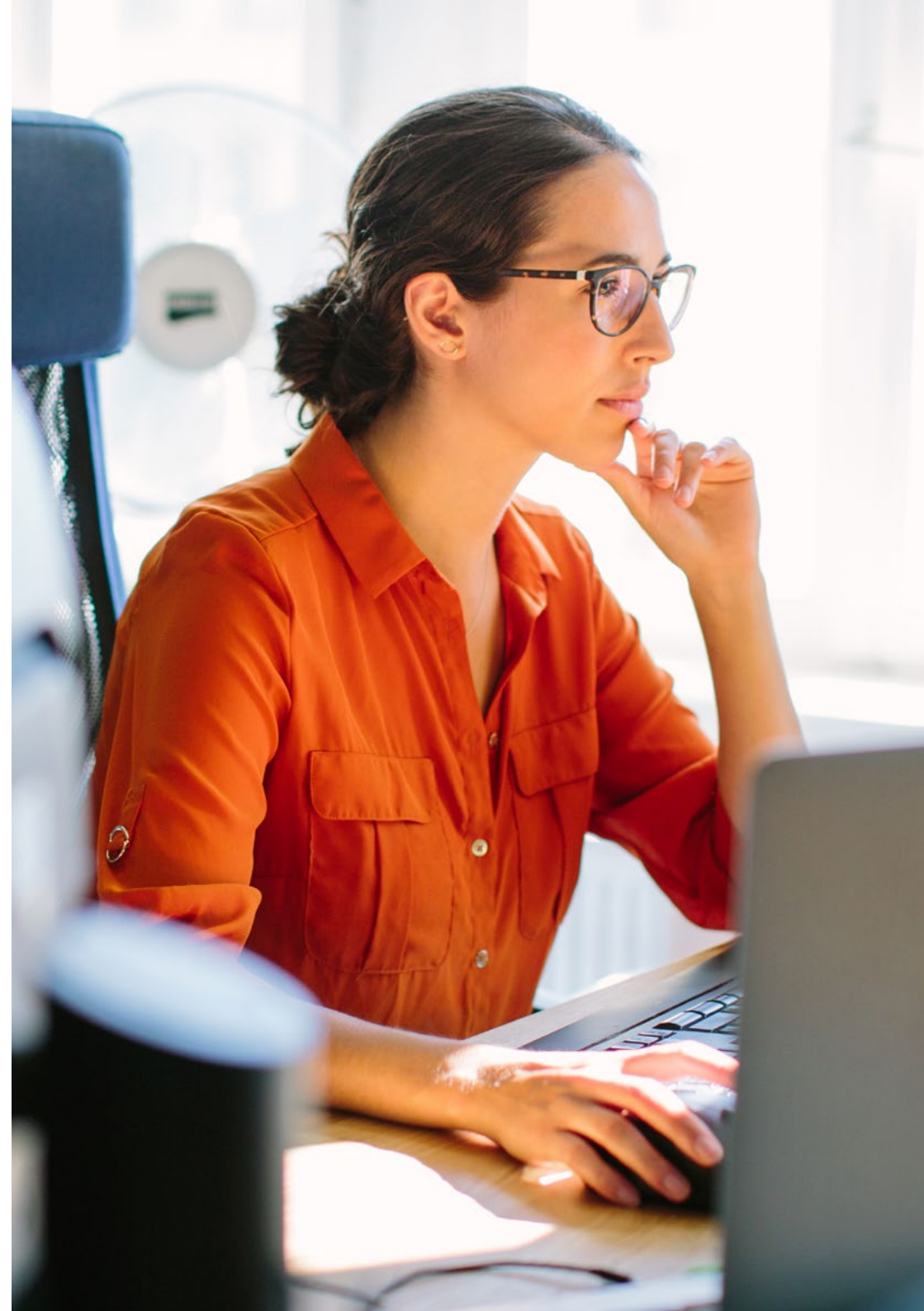
*You will enhance your design skills to create the most effective and user-friendly user interfaces using Visualforce”*



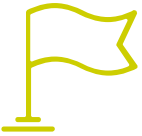
## General Skills

---

- ◆ Develop sophisticated custom pages for desktop and mobile apps that meet customer needs
- ◆ Ensure secure, scalable designs in the Salesforce data model
- ◆ Implement automated workflows to optimize business processes
- ◆ Evaluate the possibilities offered by declarative programming within the platform
- ◆ Develop practical skills in creating custom solutions
- ◆ Address integration patterns that can be used within Salesforce to suit the need of the moment
- ◆ Identify advanced development tools to make coding and debugging easier
- ◆ Take full advantage of the secondary tools provided by Salesforce for granular access monitoring and restriction by transaction type
- ◆ Define the metadata-driven approach that Salesforce relies on to synchronize configurations and developments across environments
- ◆ Establish the process to be followed for the configuration and generation of packages that group together the developments we make







## Specific Skills

---

- ◆ Acquire the skills to create custom user interfaces using Visualforce
- ◆ Automate business processes using APIs, connectors, and middleware
- ◆ Properly design the ERD data model in Salesforce, prioritizing Salesforce's Out-of-the-Box capabilities
- ◆ Identify options to avoid duplication and ensure data quality
- ◆ Propose page layout solutions using declarative tools
- ◆ Develop and evaluate workflows taking into account limits and best practices
- ◆ Propose page layout solutions using declarative tools
- ◆ Develop and evaluate workflows taking into account limits and best practices
- ◆ Determine how to address custom component issues and the tools available to detect them
- ◆ Develop custom mobile applications using information stored in Salesforce using the Mobile SDK
- ◆ Analyze execution and performance behavior of Omnistudio usage
- ◆ Develop examples for the use of Omnistudio components
- ◆ Develop different practical cases of exporting and importing data in bulk using the Bulk API
- ◆ Present the different types of existing events and identify the type of information that is sent in each of them. In turn, define how we can subscribe to the Salesforce platform event bus to capture the events published in the channel we are interested in.
- ◆ Design an error handling mechanism, in which any uncontrolled error that occurs during the execution of the application's custom code is recorded in detail
- ◆ Review different techniques to solve possible problems that exceed the limits of the platform
- ◆ Correctly monitor application access operations for both users and applications.
- ◆ Prevent unauthorized access using secondary means or malicious operations (Data Leakage)
- ◆ Correctly monitor application access operations for both users and applications.
- ◆ Prevent unauthorized access using secondary means or malicious operations (Data Leakage)

# 04

# Course Management

TECH's philosophy is based on providing students with first-rate academic itineraries that promote a leap in quality in their professional careers. For this reason, TECH carefully selects the teaching staff that makes up each of its university degrees. For this Master's Degree, it brings together a group of highly specialized professionals in the field of Salesforce Programming. These experts have an extensive work background, where they have been part of renowned companies to offer innovative IT solutions. Therefore, students will enjoy academic materials of full applicability that will help them to optimize their daily practice.



“

*Thanks to the guidance of the teaching team, you will be able to successfully overcome any challenge that arises during the implementation of Salesforce solutions”*

## Management



### Mr. Tovar Barranco, Iosu Igor

- ♦ Chief Architect at NTT Data Group
- ♦ Software Architect at Beesion Technologies
- ♦ Systems Administrator at Araldi
- ♦ .NET Developer at Gabinete de Gestión
- ♦ JAVA Programmer and J2EE Application Developer
- ♦ Senior Technician in Computer Applications Development at Centro de Estudios AEG
- ♦ Specialized in SOA Architectures
- ♦ Various Salesforce certifications

## Professors

### Mr. Sacie Alcázar, David

- ♦ Technical Architect and Expert Consultant at NTT Data
- ♦ Senior Programmer at Telefónica
- ♦ Senior Developer at Universidad de Navarra
- ♦ Technology Analyst at SN Arquitectura
- ♦ Technical Leader at BBVA Compass
- ♦ Master's Degree in Telecommunications at the Open University of Catalonia
- ♦ Salesforce Marketing Cloud Certification
- ♦ Certification in Heroku Architect Designer

### Mr. Latino Maradiaga, Mario Alfonso

- ♦ Expert Architect in the Infrastructure Sector
- ♦ Salesforce Expert Architect in Private Security Company
- ♦ Salesforce Lead Architect in the Brewing Industry
- ♦ Salesforce Architect in the Mining and Energy Sector
- ♦ Salesforce Engineer in the Agrochemical Industry
- ♦ Salesforce Engineer in the Healthcare and Pharmaceutical Distribution Sector
- ♦ Certification in: Java SE 7 Fundamentals, Oracle Database: Program with PL/SQL, Salesforce Service Cloud Consultant, Salesforce App Builder, and Salesforce Advanced Administrator

**Ms. Grao Fernández, Ester**

- ♦ IT Specialist for the Government of Aragon
- ♦ SOA/BPM Consultant at Avanttic
- ♦ J2EE Programmer at Grupo Acotelsa
- ♦ Master's Degree in Teaching Staff at Universitat Jaume I
- ♦ Computer Engineer from the Universitat Jaume I
- ♦ Technical Engineer in Computer Science Management at the University of Zaragoza

**Mr. Santos Arias, Millán**

- ♦ Chief Technical Architect - CoE Salesforce at NTT Data
- ♦ Chief Salesforce Architect - CoE Salesforce at Everis Spain, S.L.U.
- ♦ Senior Analyst at 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
- ♦ Graduate in Computer Engineering from the University of Granada

**Mr. Lobato Velázquez, Juan Manuel**

- ♦ Expert Engineer Salesforce at NTT Data
- ♦ Advanced Engineer at Serces Sistemas
- ♦ Senior Solutions Analyst
- ♦ Senior Programmer at Cibernos OutSourcing
- ♦ Engineer at Indra Software
- ♦ JAVA Programmer and J2EE Application Developer
- ♦ Certification in: Energy and Utilities Cloud, Platform App Builder, Vlocity CPQ Developer and Vlocity Omnistudio Developer

**Ms. Nebra García, Sandra**

- ♦ Salesforce Expert Engineer at NTT Data
- ♦ Salesforce Lead Engineer at a Private Security Company
- ♦ Salesforce Engineer in the Brewing Industry
- ♦ Salesforce Engineer in the Construction Industry
- ♦ Salesforce Engineer in the Food Industry
- ♦ Front-End Developer and Full-Stack Developer at Hiberus Tecnología
- ♦ Salesforce Certified Administrator
- ♦ Salesforce Certified Associate
- ♦ Graduated in Industrial Technical Engineering with specialization in Industrial Electronics by the University of Zaragoza (EINA)

**Ms. Latino Guido, Katherine Elisa**

- ♦ Systems Administrator
- ♦ Troubleshooting with Computer Systems in an Online Teaching Company
- ♦ Higher Degree in Development of Multiplatform Applications by CPIFP Los Enlaces
- ♦ Medium Degree in Microcomputer Systems and Networks

# 05

# Structure and Content

This university degree will provide students with the best development practices recommended by the Salesforce Platform, developed by a teaching staff with extensive experience in this ecosystem. The curriculum will delve into the different existing models, providing their corresponding user guides, so that graduates can use them to address the implementation processes. Likewise, the syllabus will provide innovative tools and a set of strategies that will help experts achieve success in the field of programming and distribution of proposals. In this sense, the program will offer cutting-edge methods to ensure security in Salesforce.



“

*A curriculum that will provide you with the most innovative strategies to design robust solutions that meet different security requirements”*

## Module 1. Salesforce Programming

- 1.1. Salesforce Software-as-a-Service Model
  - 1.1.1. Salesforce Licensing Model
  - 1.1.2. Governor Limits
  - 1.1.3. Types of Existing Environments
- 1.2. Salesforce Programming Model
  - 1.2.1. Metadata-based Model
  - 1.2.2. Execution Contexts
  - 1.2.3. Salesforce RESTful API Model
- 1.3. Multilanguage and Localization in Salesforce
  - 1.3.1. Localization
  - 1.3.2. Multilanguage Management in Salesforce
  - 1.3.3. Using Custom Labels in Salesforce
  - 1.3.4. Translation Workbench
- 1.4. Access and Permission Management in Salesforce
  - 1.4.1. Profile Management in Salesforce
  - 1.4.2. Permission Sets Management in Salesforce
  - 1.4.3. Permission Sets Group Management in Salesforce
- 1.5. Record Visibility Model in Salesforce
  - 1.5.1. Data Access Types
  - 1.5.2. Salesforce Visibility Model
  - 1.5.3. Record Sharing in the Visibility Model
- 1.6. Multitenant Architecture
  - 1.6.1. Multitenant Architecture in Salesforce
  - 1.6.2. Application Development on Multitenant Architecture
  - 1.6.3. Internal Request Processing
- 1.7. Database Architecture in Salesforce
  - 1.7.1. Platform Data Layer
  - 1.7.2. Database Partitioning by Tenant
  - 1.7.3. Structure of Databases
  - 1.7.4. Management of Indexes and Relations in Databases

- 1.8. Experience Cloud and Communities for External Users
  - 1.8.1. Experience Cloud Site Administration
  - 1.8.2. Permissions and Visibility Management for External Users
  - 1.8.3. Digital Experience Platform (DXP)
  - 1.8.4. Lightning Web Runtime (LWR)
- 1.9. AppExchange
  - 1.9.1. Salesforce Partner Ecosystem
  - 1.9.2. Types of Packages
  - 1.9.3. Independent Software Vendor (ISV)
- 1.10. Salesforce Infrastructure
  - 1.10.1. Domain Management in Salesforce
  - 1.10.2. Hyperforce Model
  - 1.10.3. Edge Network Model

## Module 2. Data Modeling in Salesforce

- 2.1. Data Models
  - 2.1.1. Data Modeling
  - 2.2.2. Object-entity-relationship Model
  - 2.2.3. Good Practices in Information Design and Modeling: Logical and Physical Levels
- 2.2. Object Typology
  - 2.2.1. Objects-Standard, Custom and Record Types
  - 2.2.2. Event Modeling in Salesforce Using Platform Events
  - 2.2.3. Salesforce Configuration and Parameterization Objects
  - 2.2.4. Other Special Objects
- 2.3. Document Management in Salesforce
  - 2.3.1. DMS and ECM
  - 2.3.2. Types of Documents in Salesforce
  - 2.3.3. Advanced Knowledge Management with Knowledge Base
  - 2.3.4. Best Practices in Document Management Modeling
- 2.4. Creating Objects in Salesforce
  - 2.4.1. Best Practices for Generation from UI
  - 2.4.2. Advanced Use of Schema Builder
  - 2.4.3. API for Object Creation



- 2.5. Data Quality
  - 2.5.1. Best Practices for Attribute Configuration and Validation Rules
  - 2.5.2. Control of Duplicate Records
  - 2.5.3. Reports and Other Tools for Measuring and Monitoring Quality
- 2.6. Data Query
  - 2.6.1. Best Practices for Data Querying with SOQL
  - 2.6.2. Best Practices for Data Searching with SOSL
  - 2.6.3. Dynamic SOQL & SOSL Configuration
- 2.7. Database Change Management Using DML Operations
  - 2.7.1. Data Manipulation Language
  - 2.7.2. Dynamic DML
  - 2.7.3. Best Practices for Bulk DML
  - 2.7.4. DML Exception Handling
  - 2.7.5. APIs Available for Data Management
- 2.8. Processing of Large Data Volumes (LDV)
  - 2.8.1. Table Index Management: Standard and Customized
  - 2.8.2. Skinny Tables
  - 2.8.3. How to Avoid Data Skew
  - 2.8.4. Advanced Optimization Using Data Archiving and the Query Plan Tool
- 2.9. Information Privacy
  - 2.9.1. Information Privacy
  - 2.9.2. Information Privacy Management
  - 2.9.3. Best Practices to Ensure Data Protection Regulations Compliance
- 2.10. Security Copy Management
  - 2.10.1. Security Copy Management
  - 2.10.2. Information Backup
  - 2.10.3. Best Practices in Data Backup Solution Design

### Module 3. Declarative Programming in Salesforce

- 3.1. Declarative Programming
  - 3.1.1. Declarative Construction Tools
  - 3.1.2. Formulas and Functions
  - 3.1.3. Functionalities for Mail Templates
- 3.2. Declarative Page Design
  - 3.2.1. Page Layout, Lightning App Builder and List Views
  - 3.2.2. Buttons, Links and Actions Configuration
  - 3.2.3. In-App Guidance
- 3.3. Flow Builder
  - 3.3.1. Flow Management
  - 3.3.2. Flow Life Cycle
  - 3.3.3. Reuse with Subflows
  - 3.3.4. Flow Interview: Paused Flows
  - 3.3.5. Flow Bulkification in Transactions
- 3.4. Screen Flow
  - 3.4.1. Fields in Screen Flow
  - 3.4.2. Flow with Stages
  - 3.4.3. Reactive Screen Flows
- 3.5. Declarative Automation Tools
  - 3.5.1. Autolaunched Flows-No Triggered
  - 3.5.2. Record Triggered Flows
  - 3.5.3. Platform Event Triggered Flows
- 3.6. Orchestration of Flows
  - 3.6.1. Orchestration of Flows
  - 3.6.2. Autolaunched Orchestration-No Triggered
  - 3.6.3. Record-Triggered Orchestration
- 3.7. Exception Handling in Flows
  - 3.7.1. Testing with Flow Builder
  - 3.7.2. Debugging Errors
  - 3.7.3. Monitoring Capabilities
  - 3.7.4. Exception Handling Framework

- 3.8. Approval Processes
  - 3.8.1. Approval Process Wizard and Required Configuration
  - 3.8.2. Configuration of the Approval Phases
  - 3.8.3. Limits and Considerations
- 3.9. External Services and Outbound Message
  - 3.9.1. External Service & Flow Action Configuration
  - 3.9.2. Outbound Message: Configuration, Notification & Monitoring
  - 3.9.3. External Services and Outbound Message
- 3.10. Declarative Analytical Tools
  - 3.10.1. Custom Record Types
  - 3.10.2. Report Construction
  - 3.10.3. Dashboard Construction
  - 3.10.4. Limitations of Reports and Dashboard

## Module 4. Programming in APEX for Salesforce

- 4.1. Development Tools
  - 4.1.1. Developer Console
  - 4.1.2. Recommended IDE for Apex Programming
  - 4.1.3. Salesforce Extensions for VS Code
  - 4.1.4. Code Builder
- 4.2. Programming with Apex
  - 4.2.1. Apex Programming
  - 4.2.2. Apex Transaction, Methods and Static Variables
  - 4.2.3. Management of Exceptions in Apex
- 4.3. Data Access from Apex
  - 4.3.1. SOQL Structures and Fundamentals in Apex
  - 4.3.2. Apex Variables in SOQL and SOSL
  - 4.3.3. DML Statements vs. Methods of the System Database Class
- 4.4. Apex Triggers
  - 4.4.1. Triggers and Execution Order
  - 4.4.2. Context Variables
  - 4.4.3. Bulk Triggers and Best Practices
- 4.5. Asynchronous Apex
  - 4.5.1. Future Apex
  - 4.5.2. Queueable Apex and Execution Chaining
  - 4.5.3. Apex Scheduler
- 4.6. Batch Apex
  - 4.6.1. Batch Apex Architecture
  - 4.6.2. Batch Jobs
  - 4.6.3. Batch Apex Limitations
- 4.7. Apex Security
  - 4.7.1. Apex Applications: Security Methods
  - 4.7.2. Digital Experiences: Methods for Websites
  - 4.7.3. Crypto Layer of Apex
- 4.8. Record Sharing Using Apex
  - 4.8.1. Sharing Managed by Apex
  - 4.8.2. Recalculation of Sharing with Apex
  - 4.8.3. Assignment of Territories to Opportunities
- 4.9. Apex Callouts
  - 4.9.1. SOAP Services: Proxy Classes Generated Using WSDL
  - 4.9.2. HTTP Calls: HTTP Class Methods
  - 4.9.3. Limitations of Apex Callouts
- 4.10. Execution of Unit Tests
  - 4.10.1. Unit Test Execution Patterns
  - 4.10.2. Test Data Isolation
  - 4.10.3. Simulated and Auxiliary Objects: System.StubProvider Interface
  - 4.10.4. Best Practices in the Development of Test Classes

## Module 5. UI Programming in Salesforce

- 5.1. VisualForce
  - 5.1.1. VF Page Creation and Most Common Tags
  - 5.1.2. Standard Controller and Standard List Controller
  - 5.1.3. Custom Controller
  - 5.1.4. Error Handling
  - 5.1.5. Best Practices
- 5.2. Aura Component
  - 5.2.1. Creating Aura Components and Tags
  - 5.2.2. Lightning Data Service
  - 5.2.3. Communication with Apex Server
  - 5.2.4. Composition and Communication between Components
  - 5.2.5. Error Handling
  - 5.2.6. Best Practices
- 5.3. Lightning Web Components
  - 5.3.1. LWC Creation and Most Common Tags
  - 5.3.2. Event Management and Lifecycle Hooks
  - 5.3.3. Communication with Apex Server
  - 5.3.4. Page Composition and Communication between Components
  - 5.3.5. Shadow DOM
  - 5.3.6. Use of Cell Phone Capabilities
  - 5.3.7. Error Handling and Debugging
  - 5.3.8. Best Practices
- 5.4. Lightning Message Service
  - 5.4.1. Message Channel Creation and Scope Configuration
  - 5.4.2. Publishing a Message in a Message Channel
  - 5.4.3. Subscription to a Message Channel
  - 5.4.5. Limitations
- 5.5. UI Frameworks Coexistence
  - 5.5.1. Visualforce - Aura Component Interoperability
  - 5.5.2. Visualforce - LWC Interaoperability
  - 5.5.3. LWC - Aura Component Interoperability

- 5.6. Lightning Design System
  - 5.6.1. Platforms
  - 5.6.2. Lightning Design System
  - 5.6.3. Best Practices
- 5.7. UI Testing
  - 5.7.1. Jasmine
  - 5.7.2. Jest
  - 5.7.3. UTAM
  - 5.7.4. Selenium
- 5.8. Code Quality
  - 5.8.1. Settings
  - 5.8.2. Linter
  - 5.8.3. RetireJS
- 5.9. Troubleshooting
  - 5.9.1. Chrome Developer Console
  - 5.9.2. Identify Performance Issues
  - 5.9.3. Identify Networks Issues
- 5.10. Mobile SDK
  - 5.10.1. Development Modes
  - 5.10.2. Hybrid Application Development
  - 5.10.3. Native Application Development (Android)

## Module 6. OmniStudio Framework

- 6.1. Omnistudio Architecture
  - 6.1.1. Omnistudio Architecture
  - 6.1.2. Omnistudio Component Layers
  - 6.1.3. Omnistudio Version Types
- 6.2. System Administration and Configuration
  - 6.2.1. Installing and Upgrading Omnistudio
  - 6.2.2. OmniStudio Licenses and Permissions
  - 6.2.3. Configuring Interfaces and Deployments

- 6.3. Dataraptors
  - 6.3.1. DataRaptor
  - 6.3.2. Types of Dataraptors
  - 6.3.3. Types of Data Returned by Dataraptors
  - 6.3.4. Caching and Security in Dataraptors
  - 6.3.5. Dataraptors Invocation Methods
  - 6.3.6. Best Practices for Dataraptors
- 6.4. Integration Procedures
  - 6.4.1. Integration Procedures
  - 6.4.2. Types of Actions in Integration Procedures
  - 6.4.3. Caching and Security in Integration Procedures
  - 6.4.4. Integration Procedures Invocation Methods
  - 6.4.5. Error Handling in Integration Procedures
  - 6.4.6. Best Practices for Integration Procedures
- 6.5. Flexcards
  - 6.5.1. Flexcards
  - 6.5.2. Elements for Flexcards
  - 6.5.3. Flexcard Management
  - 6.5.4. Flexcard Designer
  - 6.5.5. Debugging and Testing in Flexcards
  - 6.5.6. Best Practices for Flexcards
- 6.6. Omniscripts
  - 6.6.1. Omniscripts
  - 6.6.2. Omniscript Elements
  - 6.6.3. Omniscripts Management
  - 6.6.4. Omniscripts Designer
  - 6.6.5. Debugging and Testing in Omniscripts
  - 6.6.6. Best Practices for Omniscripts

- 6.7. Business Rules Engine
  - 6.7.1. Business Rules Engine
  - 6.7.2. Decision Matrix
  - 6.7.3. Decision Tables
  - 6.7.4. Expression Sets
  - 6.7.5. Business Rules Integrations
  - 6.7.6. Migration of Calculation Matrix and Calculation Procedures
- 6.8. Tracking Service
  - 6.8.1. Tracking Service
  - 6.8.2. Configuration of Tracking Service Usage
  - 6.8.3. Elements for Using Tracking Service
- 6.9. OmniAnalytics Overview
  - 6.9.1. OmniAnalytics
  - 6.9.2. Usage Settings for OmniAnalytics
  - 6.9.3. OmniAnalytics Results for Google Analytics
- 6.10. IDX Workbench (Deployment Tool)
  - 6.10.1. Installing IDX Workbench
  - 6.10.2. Metadata Configuration and Migration
  - 6.10.3. Metadata Migration Validation

## Module 7. API and Integrations in Salesforce

- 7.1. Salesforce Integration
  - 7.1.1. Outbound Integrations from Salesforce
  - 7.1.2. Inbound Integrations to Salesforce
  - 7.1.3. Salesforce to Salesforce Integrations
  - 7.1.4. Salesforce API Libraries
- 7.2. External Services
  - 7.2.1. Open AP Standard Specification
  - 7.2.2. External Credentials
  - 7.2.3. Named Credentials

- 7.3. Connected Apps
  - 7.3.1. Connected App
  - 7.3.2. Integration Scopes
  - 7.3.3. Integration Authorization and Identity Management
  - 7.3.4. Connected Apps Security Policies
- 7.4. SOAP API
  - 7.4.1. Enterprise WSDL
  - 7.4.2. WSDL Partner
  - 7.4.3. Apex WSDL
  - 7.4.4. WSDL Metadata
- 7.5. Synchronous REST Integrations
  - 7.5.1. REST API in Salesforce
  - 7.5.2. Connect REST API in Salesforce
  - 7.5.3. Document Management in Salesforce via API
- 7.6. API for Data Migration
  - 7.6.1. Bulk API
  - 7.6.2. Configuration and Execution of Bulk Data Migration by Bulk API
  - 7.6.3. Data Processing in Bulk Loads
  - 7.6.4. Bulk API 1.0 vs. Bulk API 2.0
- 7.7. Event-driven 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. Event Subscription from External Systems with the cometD Framework
- 7.8. Integration for Configuration and Development
  - 7.8.1. Tooling API
  - 7.8.2. API Metadata
  - 7.8.3. Dynamic Org Management Based on Tooling API

- 7.9. Data Synchronization between Orgs
  - 7.9.1. Data Synchronization Patterns Between Orgs
  - 7.9.2. Salesforce Connect with Cross-Org Adapter
  - 7.9.3. Open Data Protocol (oData)
  - 7.9.4. GraphQL with oData
- 7.10. Integration Patterns
  - 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

## Module 8. Advanced Programming in Salesforce

- 8.1. Dynamic Apex
  - 8.1.1. Dynamic Access to Object and Field Definitions
  - 8.1.2. Dynamic SOQL
  - 8.1.3. Dynamic DML
- 8.2. Platform Cache
  - 8.2.1. Platform Cache. Uses
  - 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. Uses
  - 8.3.2. Publication of an Event
  - 8.3.3. Subscription to an Event
- 8.4. SOLID Principles
  - 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. Selector Layer
  - 8.5.3. Domain Layer
  - 8.5.4. Service Layer
- 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. Error Framework
  - 8.7.1. Error Framework
  - 8.7.2. Error Capture
  - 8.7.3. Platform Event to Register the Error
- 8.8. Framework Automations
  - 8.8.1. Automation Tools
  - 8.8.2. Design of Automation Framework
  - 8.8.3. Technical Aspects
- 8.9. Development Tools
  - 8.9.1. Code Analyzer
  - 8.9.2. Apex Replay
  - 8.9.3. Apex Log Analyzer
  - 8.9.4. Explain Plan
- 8.10. Troubleshooting Limits in Apex
  - 8.10.1. CPU Time
  - 8.10.2. Too Many SOQL
  - 8.10.3. Heap Size

## Module 9. Salesforce Security

- 9.1. Apex Security
  - 9.1.1. Apex Security
  - 9.1.2. Best Practices for Secure and Attack Protected Code
  - 9.1.3. Encryption of Data in Transit and at Rest
- 9.2. Visualforce Security
  - 9.2.1. Visualforce Security
  - 9.2.2. Security in Visualforce Framework
  - 9.2.3. Best Practices for Secure Visualforce Code
- 9.3. Security in Aura
  - 9.3.1. Security in Aura Framework
  - 9.3.2. Practical Examples of Protection and Security Evidences in Aura
  - 9.3.3. Best Practices for Secure Code in Aura
- 9.4. Security in Lightning Web Components
  - 9.4.1. Security in LWC Framework
  - 9.4.2. Practical Examples of Protection and Security Evidences in LWC
  - 9.4.3. Best Practices for Secure Code in LWC
- 9.5. User Access Management
  - 9.5.1. User Profiling and Licensing (Profiles & Custom Permission)
  - 9.5.2. Role Hierarchy and Territory Model
  - 9.5.3. Team Management (Case, Account and Opportunity)
  - 9.5.4. Queues and User Groups
  - 9.5.5. Best Practices for External User Access
- 9.6. Security at the Registry Level
  - 9.6.1. OWD, Sharing Rules, ARSDR and Manual Sharing. Best Practices
  - 9.6.2. Registry Locking
  - 9.6.3. Implicit Sharing and Apex Sharing Reasons
  - 9.6.4. Data Masking
- 9.7. Security at the Field Level
  - 9.7.1. Advanced Profiling Model
  - 9.7.2. Best Practices for Dynamic Forms
  - 9.7.3. FLS Control and Record Visibility in Apex and SOQL

- 9.8. User Authentication and Salesforce Access Authorization
  - 9.8.1. Authentication Methods (U/P, SSO and Delegated Authentication)
  - 9.8.2. Best Practices in Login Flows
  - 9.8.3. Best Practices for Internal and External Access Best Practices (Experience Cloud)
  - 9.8.4. Configuring Secure Access for External Salesforce Applications
  - 9.8.5. Best Practices in OAuth Flows for Access Authorization
- 9.9. Monitoring and Security Policies
  - 9.9.1. Salesforce Shield-Advanced Use Cases
  - 9.9.2. Salesforce Shield-Field Audit Trail
  - 9.9.3. Salesforce Shield-Transaction Security Policies
- 9.10. Security at the Network Level
  - 9.10.1. HTTPS & TLS - Advanced Configuration in Salesforce
  - 9.10.2. 1-way & 2-way SSL Flow. Configuration and Applicability
  - 9.10.3. Network Configuration Best Practices

## Module 10. Application Lifecycle Management (ALM) in Salesforce

- 10.1. Salesforce Development Models
  - 10.1.1. Declarative Development Model
  - 10.1.2. Org-based Development Model
  - 10.1.3. Source Code Based Development Model
  - 10.1.4. Package-based Development Model
- 10.2. Branching Strategy
  - 10.2.1. GitFlow and Its Evolution Adapted to Salesforce
  - 10.2.2. Branching Strategy Limited to User Stories
  - 10.2.3. Branch Strategy Based on Releases and Major Releases
  - 10.2.4. Coded Approach
- 10.3. Version Control Systems (VCS)
  - 10.3.1. Version Control Systems (VCS)
  - 10.3.2. Roles and Responsibilities
  - 10.3.3. Git Configuration and Command Execution

- 10.4. Salesforce DX
  - 10.4.1. Command Console
  - 10.4.2. Orgs Connection and Configuration
  - 10.4.3. Plugins Configuration and Management
  - 10.4.4. Creating Your Own SFDX Plugin
- 10.5. Programming Based on Metadata
  - 10.5.1. Salesforce Metadata
  - 10.5.2. Metadata Coverage in Salesforce
  - 10.5.3. Deployments Based on Manifest File
- 10.6. Scratch Orgs
  - 10.6.1. Scratch Orgs
  - 10.6.2. Scratch Orgs Configuration
  - 10.6.3. Org Shapes Generation
  - 10.6.4. Programming Based on Scratch Orgs
- 10.7. Package Generation and Distribution
  - 10.7.1. Package Structure in Salesforce
  - 10.7.2. Package Distribution
  - 10.7.3. First and Second Generation Packages
- 10.8. CI/CD Tools
  - 10.8.1. Continuous Integration in Salesforce
  - 10.8.2. Phases of a Continuous Integration Process in Salesforce
  - 10.8.3. Static Code Analysis Tools
  - 10.8.4. Automation Tools
- 10.9. DevOps Center
  - 10.9.1. DevOps Center in Salesforce
  - 10.9.2. Salesforce DevOps Center Deployment Methodology
  - 10.9.3. Configuring and Executing Deployments with Salesforce DevOps Center
- 10.10. Publishing Packages on the AppExchange
  - 10.10.1. Managing Namespaces in Salesforce
  - 10.10.2. Package Publishing Process on the AppExchange
  - 10.10.3. Salesforce Security Review

06

# Methodology

This academic program offers students a different way of learning. Our methodology uses a cyclical learning approach: **Relearning**.

This teaching system is used, for example, in the most prestigious medical schools in the world, and major publications such as the **New England Journal of Medicine** have considered it to be one of the most effective.







“

*Discover Relearning, a system that abandons conventional linear learning, to take you through cyclical teaching systems: a way of learning that has proven to be extremely effective, especially in subjects that require memorization"*

## Case Study to contextualize all content

Our program offers a revolutionary approach to developing skills and knowledge. Our goal is to strengthen skills in a changing, competitive, and highly demanding environment.

“

*At TECH, you will experience a learning methodology that is shaking the foundations of traditional universities around the world”*



*You will have access to a learning system based on repetition, with natural and progressive teaching throughout the entire syllabus.*



*The student will learn to solve complex situations in real business environments through collaborative activities and real cases.*

### A learning method that is different and innovative

This TECH program is an intensive educational program, created from scratch, which presents the most demanding challenges and decisions in this field, both nationally and internationally. This methodology promotes personal and professional growth, representing a significant step towards success. The case method, a technique that lays the foundation for this content, ensures that the most current economic, social and professional reality is taken into account.

“*Our program prepares you to face new challenges in uncertain environments and achieve success in your career”*

The case method has been the most widely used learning system among the world's leading Information Technology schools for as long as they have existed. The case method was developed in 1912 so that law students would not only learn the law based on theoretical content. It consisted of presenting students with real-life, complex situations for them to make informed decisions and value judgments on how to resolve them. In 1924, Harvard adopted it as a standard teaching method.

What should a professional do in a given situation? This is the question that you are presented with in the case method, an action-oriented learning method. Throughout the course, students will be presented with multiple real cases. They will have to combine all their knowledge and research, and argue and defend their ideas and decisions.

## Relearning Methodology

TECH effectively combines the Case Study methodology with a 100% online learning system based on repetition, which combines different teaching elements in each lesson.

We enhance the Case Study with the best 100% online teaching method: Relearning.

*In 2019, we obtained the best learning results of all online universities in the world.*

At TECH you will learn using a cutting-edge methodology designed to train the executives of the future. This method, at the forefront of international teaching, is called Relearning.

Our university is the only one in the world authorized to employ this successful method. In 2019, we managed to improve our students' overall satisfaction levels (teaching quality, quality of materials, course structure, objectives...) based on the best online university indicators.



In our program, learning is not a linear process, but rather a spiral (learn, unlearn, forget, and re-learn). Therefore, we combine each of these elements concentrically.

This methodology has trained more than 650,000 university graduates with unprecedented success in fields as diverse as biochemistry, genetics, surgery, international law, management skills, sports science, philosophy, law, engineering, journalism, history, and financial markets and instruments. All this in a highly demanding environment, where the students have a strong socio-economic profile and an average age of 43.5 years.

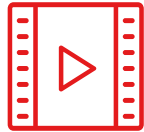
*Relearning will allow you to learn with less effort and better performance, involving you more in your training, developing a critical mindset, defending arguments, and contrasting opinions: a direct equation for success.*

From the latest scientific evidence in the field of neuroscience, not only do we know how to organize information, ideas, images and memories, but we know that the place and context where we have learned something is fundamental for us to be able to remember it and store it in the hippocampus, to retain it in our long-term memory.

In this way, and in what is called neurocognitive context-dependent e-learning, the different elements in our program are connected to the context where the individual carries out their professional activity.



This program offers the best educational material, prepared with professionals in mind:



#### Study Material

All teaching material is produced by the specialists who teach the course, specifically for the course, so that the teaching content is highly specific and precise.

These contents are then applied to the audiovisual format, to create the TECH online working method. All this, with the latest techniques that offer high quality pieces in each and every one of the materials that are made available to the student.



#### Classes

There is scientific evidence suggesting that observing third-party experts can be useful.

Learning from an Expert strengthens knowledge and memory, and generates confidence in future difficult decisions.



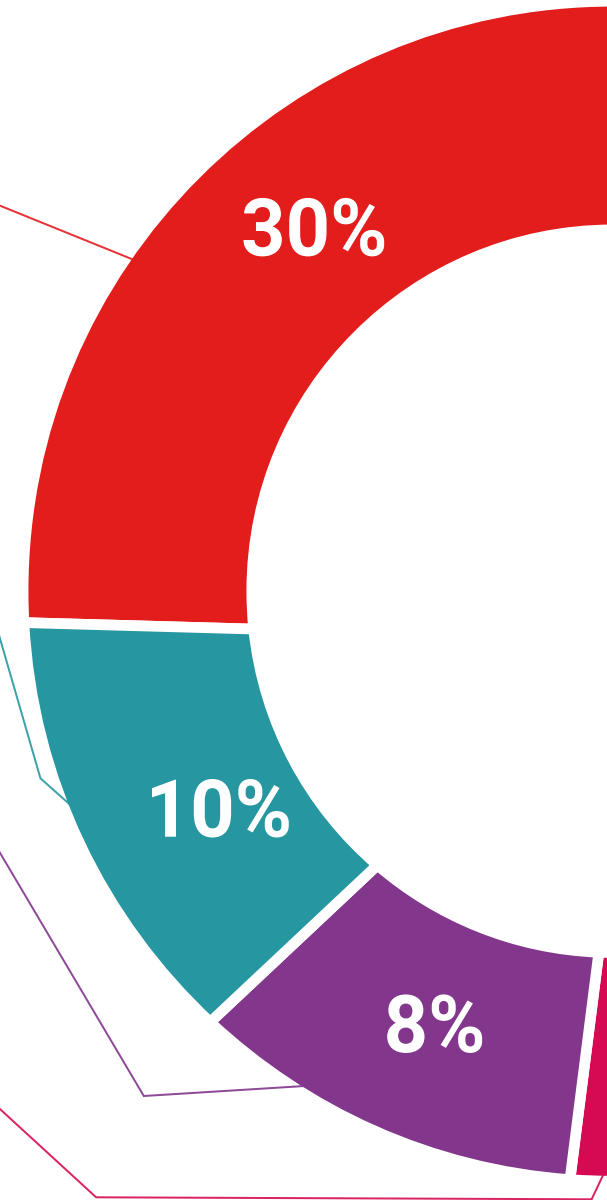
#### Practising Skills and Abilities

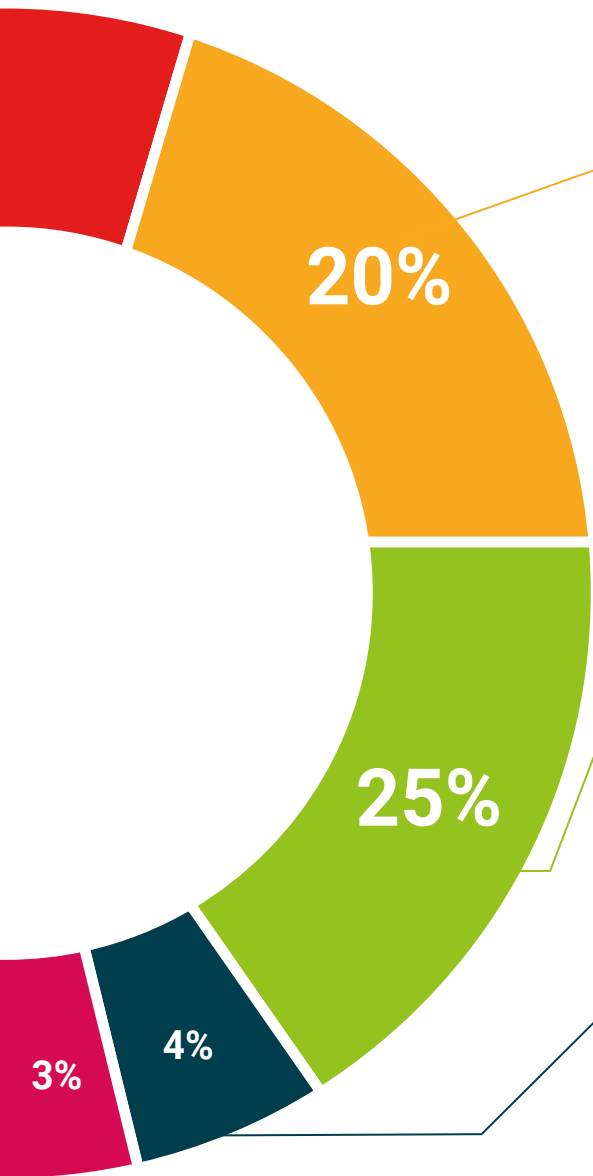
They will carry out activities to develop specific skills and abilities in each subject area. Exercises and activities to acquire and develop the skills and abilities that a specialist needs to develop in the context of the globalization that we are experiencing.



#### Additional Reading

Recent articles, consensus documents and international guidelines, among others. In TECH's virtual library, students will have access to everything they need to complete their course.





#### Case Studies

Students will complete a selection of the best case studies chosen specifically for this program. Cases that are presented, analyzed, and supervised by the best specialists in the world.



#### Interactive Summaries

The TECH team presents the contents attractively and dynamically in multimedia lessons that include audio, videos, images, diagrams, and concept maps in order to reinforce knowledge.

This exclusive educational system for presenting multimedia content was awarded by Microsoft as a "European Success Story".



#### Testing & Retesting

We periodically evaluate and re-evaluate students' knowledge throughout the program, through assessment and self-assessment activities and exercises, so that they can see how they are achieving their goals.



# 07 Certificate

The Master's Degree in Salesforce Programming guarantees students, in addition to the most rigorous and up-to-date education, access to a Master's Degree diploma issued by TECH Technological University.





“

*Successfully complete this program and receive your university qualification without having to travel or fill out laborious paperwork"*

This program will allow you to obtain your **Master's Degree diploma in Salesforce Programming** endorsed by **TECH Global University**, the world's largest online university.

**TECH Global University** is an official European University publicly recognized by the Government of Andorra ([official bulletin](#)). Andorra is part of the European Higher Education Area (EHEA) since 2003. The EHEA is an initiative promoted by the European Union that aims to organize the international training framework and harmonize the higher education systems of the member countries of this space. The project promotes common values, the implementation of collaborative tools and strengthening its quality assurance mechanisms to enhance collaboration and mobility among students, researchers and academics.

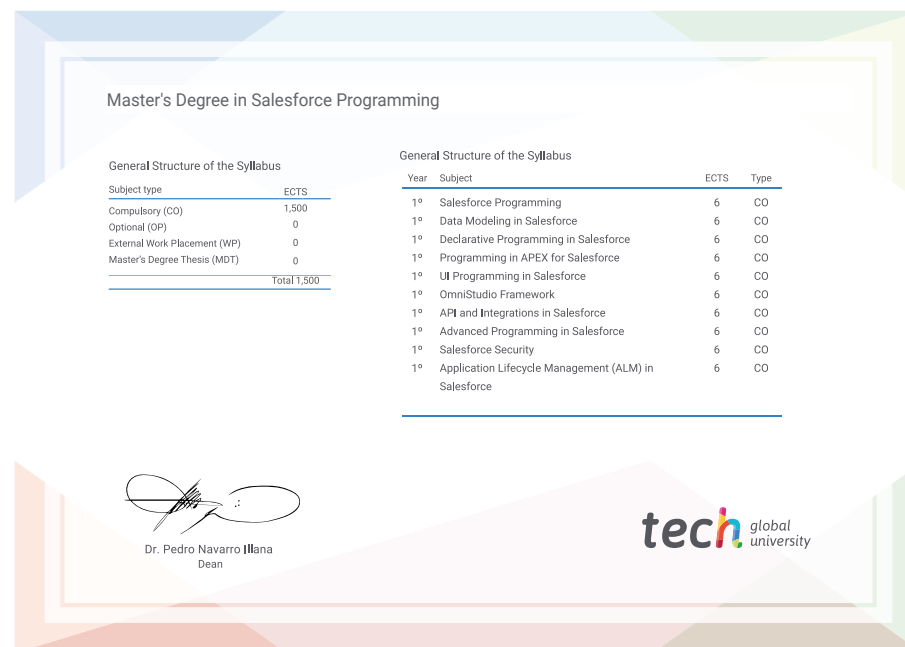
This **TECH Global University** title is a European program of continuing education and professional updating that guarantees the acquisition of competencies in its area of knowledge, providing a high curricular value to the student who completes the program.

Title: **Master's Degree in Salesforce Programming**

Modality: **online**

Duration: **12 months**

Accreditation: **60 ECTS**



\*Apostille Convention. In the event that the student wishes to have their paper diploma issued with an apostille, TECH Global University will make the necessary arrangements to obtain it, at an additional cost.

future  
health confidence people  
education information tutors  
guarantee accreditation teaching  
institutions technology learning  
community commitment  
personalized service innovation  
knowledge present quality  
development languages  
virtual classroom



## Master's Degree Salesforce Programming

- » Modality: online
- » Duration: 12 months
- » Certificate: TECH Global University
- » Credits: 60 ECTS
- » Schedule: at your own pace
- » Exams: online

# Master's Degree Salesforce Programming

