

# Hybrid Master's Degree

## Creation of Network Interfaces and Applications





## Hybrid Master's Degree

### Creation of Network Interfaces and Applications

Modality: Hybrid (Online + Internship)

Duration: 12 months

Certificate: TECH Global University

Credits: 60 + 4 ECTS

Website: [www.techtitude.com/us/information-technology/hybrid-master-degree/hybrid-master-degree-creation-network interfaces-applications](http://www.techtitude.com/us/information-technology/hybrid-master-degree/hybrid-master-degree-creation-network-interfaces-applications)

# Index

01

Introduction

---

p. 4

02

Why Study this Hybrid Master's Degree?

---

p. 8

03

Objectives

---

p. 12

04

Skills

---

p. 18

05

Educational Plan

---

p. 22

06

Clinical Internship

---

p. 34

07

Where Can I Do the Internship?

---

p. 40

08

Methodology

---

p. 44

09

Certificate

---

p. 52

# 01

# Introduction

Networked interfaces and applications provide a platform for efficient and secure information sharing, promoting productivity and innovation in everything from education to e-commerce. By providing intuitive interfaces and an optimized user experience, these networked applications enhance the user experience and increase satisfaction, resulting in increased customer retention and loyalty. For this reason, TECH has developed this comprehensive program for computer scientists to specialize in a field that is in high demand in organizations. Through a format that combines theoretical study 100% online and a practical stay of 3 weeks in a prestigious IT company.



“

*Bet on TECH! You will have the opportunity to work on practical and collaborative projects, fostering the development of your teamwork and problem solving skills"*

Networked interfaces and applications facilitate communication and collaboration between geographically dispersed users, enabling them to share information efficiently and in real time. They also improve accessibility to services and resources, promoting inclusion and equal opportunities, and help to collect and analyze data, enabling informed decision-making and continuous improvement of the user experience.

In this way this Hybrid Master's Degree was created, whose first phase will immerse the computer scientist in the theory on the Creation of Network Interfaces and Applications. In this way, the program will focus on providing solid knowledge on interface design, usability and adaptability to human diversity. In addition, graduates will delve into systems operation, relational modeling and data manipulation using SQL.

The curriculum will also introduce professionals to the use of HTML, CSS and JavaScript for web creation, as well as MVC architecture and component-oriented programming. They will also become familiar with open source software and the various tools available, including operating systems, business management and CMS such as WordPress.

Finally, we will analyze the software reuse strategy, design patterns and the use of *frameworks* for the creation of graphical user interfaces and web development, delving into the Model View Controller (MVC) pattern. Likewise, agile methodologies such as Scrum, extreme programming and development based on reuse will be explored.

In addition, this TECH academic degree will allow students to develop their skills in an exhaustive practical stay, in which they will spend 3 weeks working in a high-level IT company, learning from the best experts in the field of the Creation of Interfaces and Web Applications. They will acquire the knowledge and skills necessary to excel in a highly competitive field.

This **Hybrid Master's Degree in Creation of Interfaces and Network Applications** contains the most complete and up-to-date program on the market. The most important features include:

- ♦ Development of more than 100 case studies presented by IT professionals, experts in creation of interfaces and university professors with extensive experience in this field
- ♦ Their graphic, schematic and practical contents provide essential information on those disciplines that are indispensable for professional practice
- ♦ Algorithm-based interactive learning system for decision-making in the situations that are presented to the student
- ♦ All of this will be complemented by 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
- ♦ Furthermore, you will be able to carry out a internship in one of the best Companies



*You will add to your online study an internship in an elite IT company, equipped with the highest level of technology"*

“

*The Creation of Network Interfaces and Applications represents a powerful tool to drive technological progress and global interconnection"*

In this proposed Master's Degree, of a professionalizing nature and blended learning modality, the program is aimed at updating IT professionals who are dedicated to the creation of web interfaces and applications, and who require a high level of qualification. The contents are based on the latest scientific evidence, and oriented in an educational way to integrate theoretical knowledge into practice, and the theoretical-practical elements will facilitate knowledge update and decision-making in patient situations posed.

Thanks to its multimedia content elaborated with the latest educational technology, they will allow the IT professional a situated and contextual learning, that is to say, a simulated environment that will provide an immersive learning programmed to specialize 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 throughout the program. For this purpose, the students will be assisted by an innovative interactive video system created by renowned and experienced experts.

*You will delve into the advanced database systems, XML, and parallel and distributed databases, all through an extensive library of the most innovative multimedia resources.*

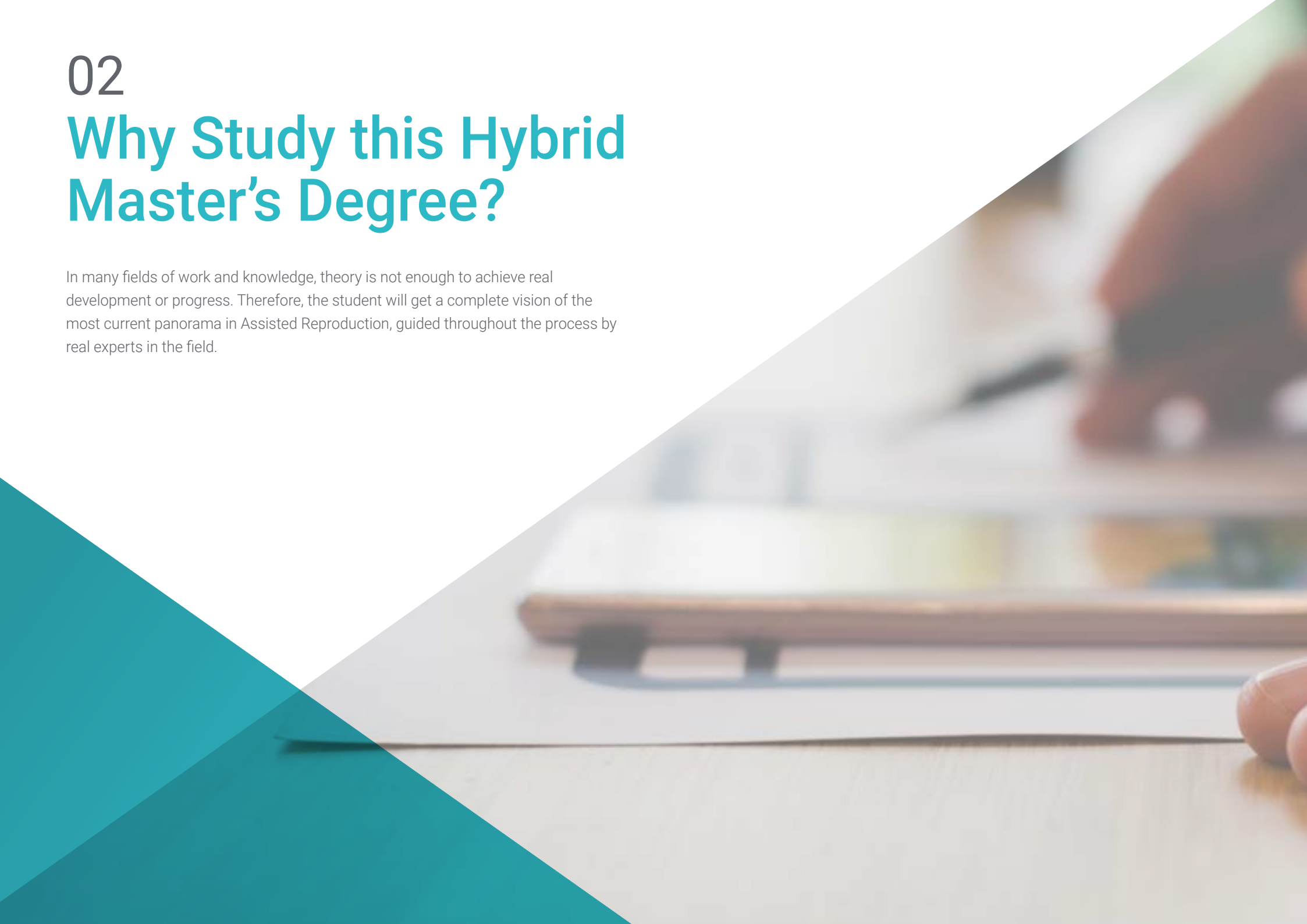
*You will cover system architectures, software testing and ISO/IEC standards, as well as delve into the concept of DevOps and its core practices.*



02

# Why Study this Hybrid Master's Degree?

In many fields of work and knowledge, theory is not enough to achieve real development or progress. Therefore, the student will get a complete vision of the most current panorama in Assisted Reproduction, guided throughout the process by real experts in the field.



“

*Access to well-paid job opportunities in leading technology companies, innovative startups or even start your own business projects"*

### 1. Updating from the Latest Technology Available

In the area of interface creation and networked applications, the integration of Artificial Intelligence and machine learning systems enables the development of more intuitive and adaptive interfaces, capable of anticipating user needs and dynamically personalizing the experience. In addition, cloud computing provides a scalable and flexible infrastructure for the development and deployment of web applications, allowing developers to create multiplatform and high-performance experiences that adapt to the demands of today's market.

### 2. Gaining In-depth Knowledge from the Experience of Top Specialists

The large team of professionals who will accompany the computer scientist throughout the entire practical period is a first-rate endorsement and a guarantee of unprecedented updating. With a specifically designated tutor, the graduate will be able to work on real projects in a state-of-the-art environment, which will allow them to incorporate into their daily practice the most effective procedures and tools for the Creation of Network Interfaces and Applications.

### 3. Entering First-Class Professional Environments

TECH carefully selects all the companies available for Internship Programs. Thanks to this, the specialist will have guaranteed access to a prestigious IT organization in the area of interface and web application development. In this way, they will be able to experience the day-to-day of a demanding, rigorous and exhaustive area of work, always applying the latest advances in their work methodology.



#### 4. Combining the Best Theory with State-of-the-Art Practice

The academic market is plagued by teaching programs that are poorly adapted to the daily work of the specialist and that require long teaching hours, often not very compatible with personal and professional life. For this reason, TECH offers a new learning model, 100% practical, that allows you to get in front of state-of-the-art procedures in the field of the Creation of Interfaces and Network Applications and, best of all, to put it into professional practice in only 3 weeks.

#### 5. Opening the Door to New Opportunities

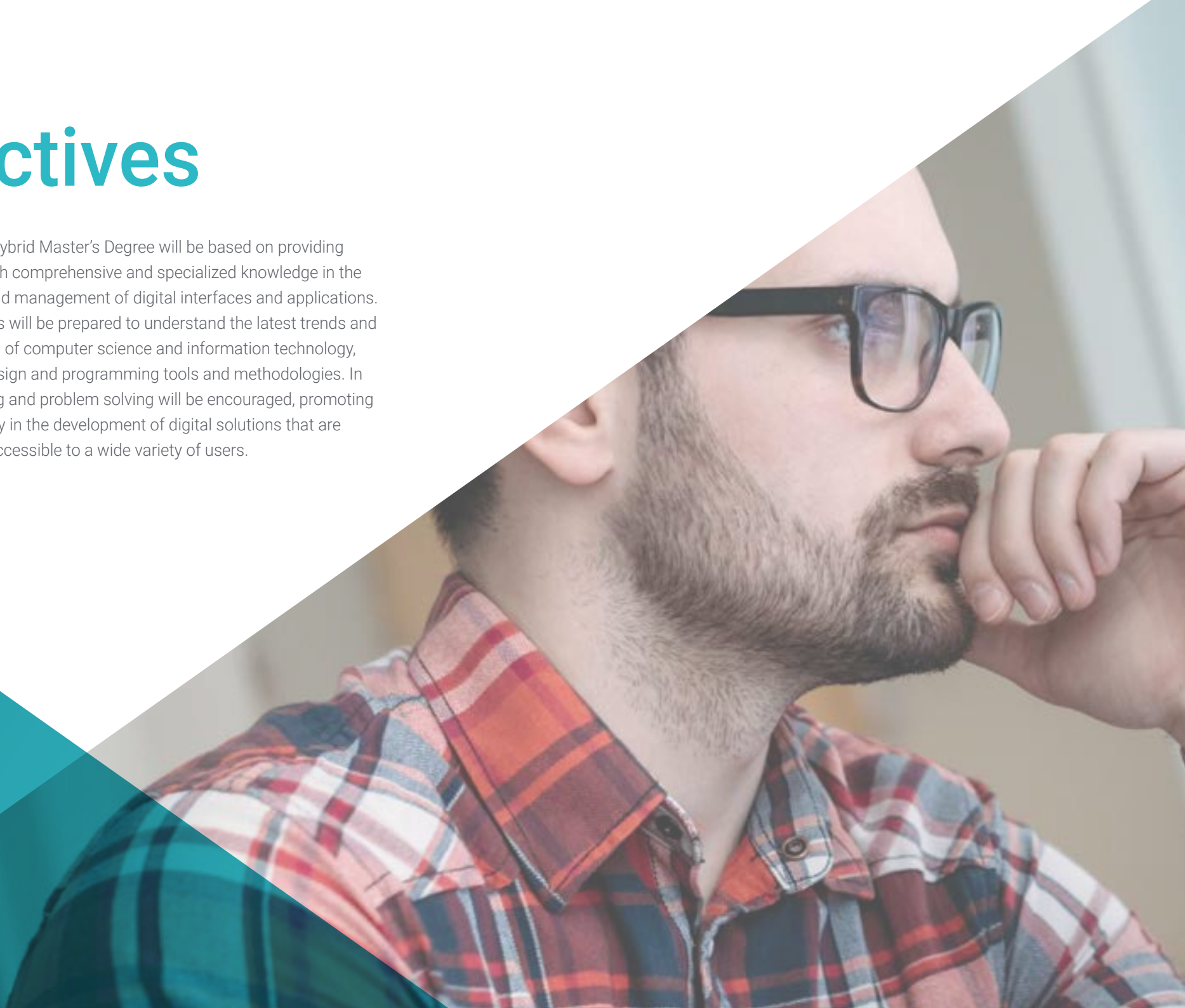
With the advancement of technology and the growing demand for innovative digital experiences, computer scientists have the opportunity to explore and leverage emerging tools, such as Artificial Intelligence and cloud computing. In addition, globalization and the increasing interconnectedness of digital devices and platforms create a landscape conducive to international collaboration and professional networking, allowing experts in creating networked interfaces and applications to expand their reach and contribute to the development of innovative solutions on a global scale.

“

*You will have full practical immersion  
at the center of your choice”*

# 03 Objectives

The objectives of this Hybrid Master's Degree will be based on providing computer scientists with comprehensive and specialized knowledge in the design, development and management of digital interfaces and applications. Therefore, professionals will be prepared to understand the latest trends and technologies in the field of computer science and information technology, as well as to master design and programming tools and methodologies. In addition, critical thinking and problem solving will be encouraged, promoting innovation and creativity in the development of digital solutions that are efficient, intuitive and accessible to a wide variety of users.



“

*This Hybrid Master's Degree will turn you into a highly qualified professional with strategic vision to lead innovative digital projects in a constantly changing technological environment"*



### General Objective

---

- ♦ The objective of the Hybrid Master's Degree in Network Interfaces and Applications Creation is to specialize graduates scientifically and technologically to practice computer engineering, providing them with a solid knowledge base in key fields. In this way, professionals will acquire in-depth knowledge in the broad field of computing, ranging from theoretical aspects to practical applications in the design and development of interfaces and network applications. In addition, emphasis will be placed on the understanding of the structure of computers, allowing the understanding of the components and the inner workings of computer systems. Likewise, software engineering will be delved into, providing computer scientists with the necessary skills to plan, design, implement and maintain high quality software





## Specific Objectives

---

### Module 1. Human-Computer Interaction

- ♦ Acquire solid knowledge related to human-computer interaction and the creation of usable interfaces
- ♦ Understand the importance of application usability and why it is important to take it into account when designing our software
- ♦ Understand the different types of human diversity, the limitations they imply and how to adapt the interfaces according to the specific needs of each one of them
- ♦ Learn the process of interface design, from requirements analysis to evaluation, going through the different intermediate stages necessary to carry out an adequate interface
- ♦ Know the different accessibility guidelines, the standards that establish them and the tools that allow us to assess them
- ♦ Understand the different methods of interaction with the computer, by means of peripherals and devices

### Module 2. Databases

- ♦ Learn the different applications and purposes of database systems, as well as their operation and architecture
- ♦ Understand the relational model, from its structure and operations to extended relational algebra
- ♦ Learn in depth what SQL databases are, how they work, the definition of data and the creation of queries from the most basic to the most advanced and complex

- ♦ Learn how to design databases using the entity-relationship model, how to create diagrams and the characteristics of the extended E-R model
- ♦ Delve into the design of relational databases, analyzing the different normal forms and decomposition algorithms
- ♦ Laying the groundwork for understanding the operation of NoSQL databases, as well as introducing the Mongo DB database

### Module 3. Development of Network Applications

- ♦ Know the characteristics of the HTML markup language and its use in web creation together with CSS style sheets
- ♦ Learn how to use the browser-oriented programming language JavaScript, and some of its main features
- ♦ Understand the concepts of component-oriented programming and the component architecture
- ♦ Learn how to use the Bootstrap front-end framework for website design
- ♦ Understand the structure of the controller view model in the development of dynamic web sites
- ♦ Know the service-oriented architecture and the basics of the HTTP protocol

#### **Module 4. Free Software and Open Knowledge**

- ♦ Learn the concepts of Free Software and Open Knowledge, as well as the different types of associated licenses
- ♦ Know the main free tools available in different areas such as operating systems, business management, content management systems and multimedia content creation
- ♦ Understand the importance and benefits of free software in the business world, both for its features and costs
- ♦ Delve into the knowledge of the GNU/Linux operating system, as well as the different existing distributions
- ♦ Learn about the operation and development of WordPress, given that this CMS accounts for more than 35% of the active websites in the world, and more than 60% in the particular case of CMSs
- ♦ Understand how the operating system for Android mobile devices works, as well as the basics for the development of mobile applications: , both native development and with cross-platform frameworks

#### **Module 5. Advanced Databases**

- ♦ Introduce the different database systems currently available on the market.
- ♦ Learn the use of XML and databases for the web
- ♦ Understand the operation of advanced databases such as parallel and distributed databases
- ♦ Understand the importance of indexing and association in database systems
- ♦ Understand how transactional processing and retrieval systems work
- ♦ Acquire knowledge related to non-relational databases and data mining

#### **Module 6. Software Engineering**

- ♦ Know the software engineering framework and the ISO/IEC 12207 standard
- ♦ Learn the characteristics of the unified software development process and planning in the context of agile software development
- ♦ Learn the different styles of distributed software design and service-oriented software architectures
- ♦ Learn the essential concepts in graphical user interface design
- ♦ Understand the basics of web application development
- ♦ Delve into software testing strategies and techniques, software quality factors and different metrics used

#### **Module 7. Advanced Programming**

- ♦ Delve into the knowledge of programming, especially as it relates to object-oriented programming, and the different types of relationships between classes
- ♦ Know the different design patterns for object-oriented problems.
- ♦ Learn about event-driven programming and user interface development with Qt
- ♦ Acquire the essential knowledge of Concurrent Programming, processes and threads
- ♦ Learn how to manage the use of threads and synchronization, as well as the resolution of common problems within Concurrent Programming
- ♦ Understand the importance of documentation and testing in software development

**Module 8. Software Reuse**

- ♦ Know the big picture in software reuse strategy
- ♦ Learn the different patterns related to software reuse, both in terms of design, creation, structure and behavior
- ♦ Learn about the concept of framework, as well as to the main types such as those for graphical user interface design, web application development and object persistence management in databases
- ♦ Understand the current widely used Model View Controller (MVC) pattern

**Module 9. Artificial Intelligence and Knowledge Engineering**

- ♦ Lay the foundations of artificial intelligence and knowledge engineering, making a brief tour through the history up to the present day
- ♦ Understand the essential concepts of search in Artificial Intelligence, both informed and uninformed search
- ♦ Understand how Artificial Intelligence works in games
- ♦ Learn the fundamental concepts of neural networks and the use of genetic algorithms
- ♦ Acquire the appropriate mechanisms to represent knowledge, especially taking into account the semantic web
- ♦ Understand the functioning of expert systems and decision support systems

**Module 10. Advanced Software Engineering**

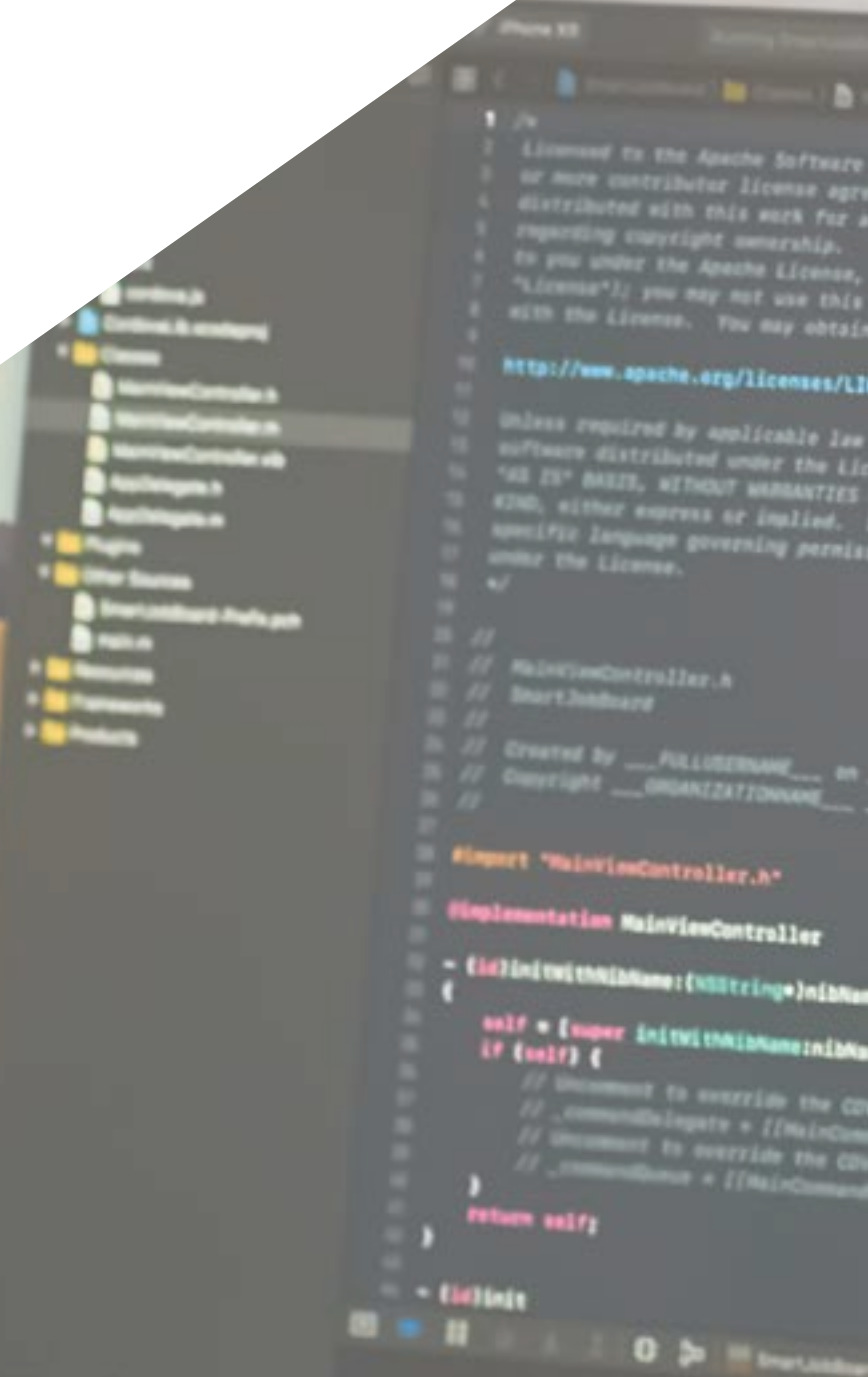
- ♦ Know in depth the different agile methodologies used in software engineering
- ♦ Learn to develop using Scrum, extreme programming and reuse-based software development techniques
- ♦ Understand the different patterns of system architectures and software design, as well as the architecture of cloud applications
- ♦ Learn how to test software, with methodologies such as Test-Driven Development, Acceptance Test-Driven Development, Behavior-Driven Development, BDD and Cucumber
- ♦ Delve into the improvement of the software development process and software quality using ISO/IEC standards
- ♦ Introduce the DevOps concept and its main practices



*You will be equipped with technical skills, theoretical knowledge and professional competencies to excel in roles related to network interface and application design"*

# 04 Skills

Among the competencies acquired in this university program will be the ability to analyze and understand user needs and design intuitive and functional interfaces that enhance the user experience. In addition, graduates will develop technical skills in areas such as web programming, database design and implementation of distributed systems. Teamwork and communication with other professionals in the sector will also be encouraged, as well as the ability to adapt to technological changes and the capacity for continuous learning, in order to keep updated in an increasingly evolving digital environment.





“

*You will be able to design attractive and functional interfaces and applications, ensuring an optimal user experience, thanks to the revolutionary Relearning methodology"*



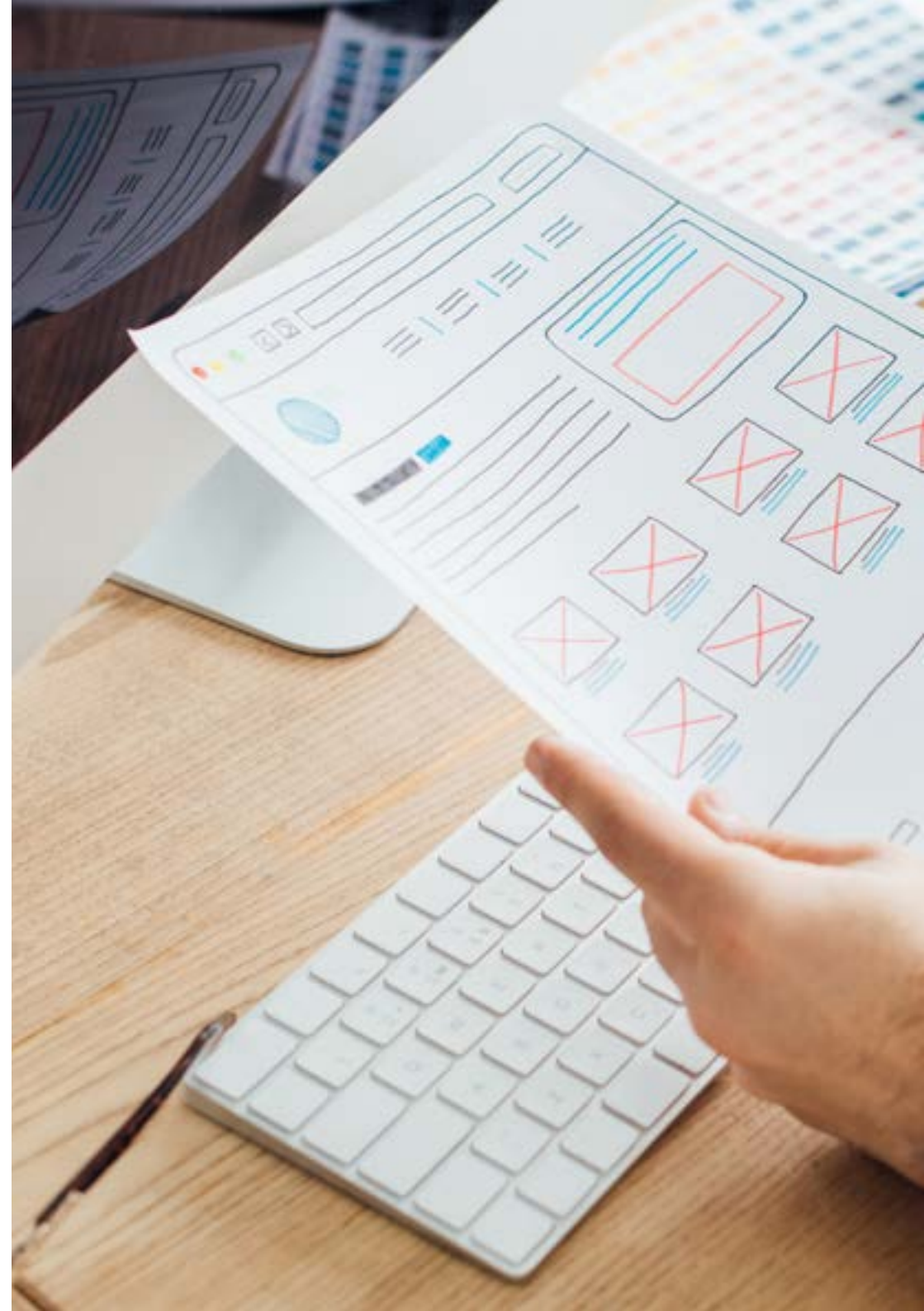
## General Skill

---

- ♦ Acquire the necessary skills for the professional practice of computer engineering with the knowledge of all the necessary factors to carry it out it with quality and solvency

“

*You will gain technical skills  
in software development,  
mastering various programming  
languages, frameworks and web  
and mobile development tools”*





## Specific Skills

---

- ♦ Develop in-depth knowledge of all facets of human-computer interaction and how they involve computer developments
- ♦ Be proficient in the use of databases
- ♦ Develop different types of network applications
- ♦ Describe and take advantage of free software and open knowledge on the web
- ♦ Work as a software engineer
- ♦ Control the use of advanced databases
- ♦ Perform advanced programming
- ♦ Know how to reuse software
- ♦ Create interfaces and network applications
- ♦ Have mastery of the different systems of work in advanced software engineering

# 05 Educational Plan

From the fundamentals of human-computer interaction to the most advanced technologies in the field of Artificial Intelligence and cloud computing, this program will cover essential aspects such as requirements analysis, usable interface design, web and mobile programming, networked systems architecture and computer security, among others. In addition, professionals will address topics related to databases, software engineering and open source software, preparing them to meet the challenges and seize the opportunities in the digital technology industry.



“

*You will be equipped with skills in the use of modern tools and technologies, as well as in the application of agile methodologies and good software development practices"*

## Module 1. Human-Computer Interaction

- 1.1. Introduction to Human-Computer Interaction
  - 1.1.1. What is Human-Computer Interaction
  - 1.1.2. Relationship of Human-Computer Interaction with Other Disciplines
  - 1.1.3. The User Interface
  - 1.1.4. Usability and Accessibility
  - 1.1.5. User Experience and User-Centered Design
- 1.2. The Computer and Interaction: User Interface and Interaction Paradigms
  - 1.2.1. Interaction
  - 1.2.2. Paradigms and Styles of Interaction
  - 1.2.3. Evolution of User Interfaces
  - 1.2.4. Classic User Interfaces: WIMP/GUI, Commands, Voice, Virtual Reality.
  - 1.2.5. Innovative User Interfaces: Mobiles, Laptops, Collaboratives, BCI
- 1.3. The Human Factor: Psychological and Cognitive Aspects
  - 1.3.1. The Importance of the Human Factor in Interaction
  - 1.3.2. Human Information Processing
  - 1.3.3. The Input and Output of Information: Visual, Auditory and Tactile
  - 1.3.4. Perception and Attention
  - 1.3.5. Knowledge and Mental Models: Representation, Organization, and Acquisition
- 1.4. The Human Factor: Sensory and Physical Limitations
  - 1.4.1. Functional Diversity, Disability and Impairment
  - 1.4.2. Visual Diversity
  - 1.4.3. Hearing Diversity
  - 1.4.4. Cognitive Diversity
  - 1.4.5. Motor Diversity
  - 1.4.6. The Case of Digital Immigrants
- 1.5. The Design Process (I): Requirements Analysis for User Interface Design
  - 1.5.1. User-Centered Design
  - 1.5.2. What is Requirements Analysis?
  - 1.5.3. Information Gathering
  - 1.5.4. Analysis and Interpretation of the Information
  - 1.5.5. Usability and Accessibility Analysis
- 1.6. The Design Process (II): Prototyping and Task Analysis
  - 1.6.1. Conceptual Design
  - 1.6.2. Prototyping
  - 1.6.3. Hierarchical Task Analysis
- 1.7. The Design Process (III): Evaluation
  - 1.7.1. Evaluation in the Design Process: Objectives and Methods
  - 1.7.2. Evaluation Methods Without Users
  - 1.7.3. Evaluation Methods with Users
  - 1.7.4. Evaluation Standards and Norms
- 1.8. Accessibility: Definition and Guidelines
  - 1.8.1. Accessibility and Universal Design
  - 1.8.2. The WAI Initiative and the WCAG Guidelines
  - 1.8.3. WCAG 2.0 and 2.1 Guidelines
- 1.9. Accessibility: Assessment and Functional Diversity
  - 1.9.1. Web Accessibility Evaluation Tools
  - 1.9.2. Accessibility and Functional Diversity
- 1.10. The Computer and Interaction: Peripherals and Devices
  - 1.10.1. Traditional Devices and Peripherals
  - 1.10.2. Alternative Devices and Peripherals
  - 1.10.3. Cell Phones and Tablets
  - 1.10.4. Functional Diversity, Interaction and Peripherals

**Module 2. Databases.**

- 2.1. Applications and Purposes of Database Systems
  - 2.1.1. Applications of the Different Database Systems
  - 2.1.2. Purpose of the Different Database Systems
  - 2.1.3. View of the Data
- 2.2. Database and Architecture
  - 2.2.1. Relational Database
  - 2.2.2. Database Design
  - 2.2.3. Object-Based and Semi-Structured Databases
  - 2.2.4. Data Storage and Queries
  - 2.2.5. Transaction Management
  - 2.2.6. Data Mining and Analysis
  - 2.2.7. Database Architecture
- 2.3. Relational Model: Structure, Operations and Extended Relational Algebra
  - 2.3.1. The Structure of Relational Databases
  - 2.3.2. Fundamental Operations in the Relational Algebra
  - 2.3.3. Other Relational Algebra Operations
  - 2.3.4. Extended Relational Algebra Operations
  - 2.3.5. Null Values
  - 2.3.6. Database Modification
- 2.4. SQL (I)
  - 2.4.1. What is SQL?
  - 2.4.2. The Definition of Data
  - 2.4.3. Basic Structure of SQL Queries
  - 2.4.4. Operations on Sets
  - 2.4.5. Aggregation Functions
  - 2.4.6. Null Values
- 2.5. SQL (II)
  - 2.5.1. Nested Subqueries
  - 2.5.2. Complex Queries
  - 2.5.3. Views
  - 2.5.4. Cursors
  - 2.5.5. Complex Queries
  - 2.5.6. Triggers
- 2.6. Database Design and the E-R Model
  - 2.6.1. Overview of the Design Process
  - 2.6.2. The Entity-Relationship Model
  - 2.6.3. Restrictions
- 2.7. Entity-Relationship Diagrams
  - 2.7.1. Entity-Relationship Diagrams
  - 2.7.2. Entity-Relationship Design Aspects
  - 2.7.3. Weak Entity Sets
- 2.8. The Extended Entity-Relationship Model
  - 2.8.1. Characteristics of the Extended E-R Model
  - 2.8.2. Design of a Database
  - 2.8.3. Reduction to Relational Schemas
- 2.9. Designing from Relational Databases
  - 2.9.1. Characteristics of Good Relational Designs
  - 2.9.2. Atomic Domains and the First Normal Form (1FN)
  - 2.9.3. Decomposition by Functional Dependencies
  - 2.9.4. Theory of Functional Dependencies
  - 2.9.5. Decomposition Algorithms
  - 2.9.6. Decomposition by Means of Multivalued Dependencies
  - 2.9.7. More Normal Forms
  - 2.9.8. Database Design Process
- 2.10. NoSQL Databases
  - 2.10.1. What are NoSQL Databases?
  - 2.10.2. Analysis of the Different NoSQL Options and their Characteristics
  - 2.10.3. Mongo DB

### Module 3. Development of Network Applications

- 3.1. HTML5 Markup Languages
  - 3.1.1. HTML Basics
  - 3.1.2. New HTML 5 Elements
  - 3.1.3. Forms: New Controls
- 3.2. Introduction to CSS Style Sheets
  - 3.2.1. First Steps with CSS
  - 3.2.2. Introduction to CSS3
- 3.3. Browser Scripting Language: JavaScript
  - 3.3.1. JavaScript Basics
  - 3.3.2. DOM
  - 3.3.3. Events
  - 3.3.4. JQuery
  - 3.3.5. Ajax
- 3.4. Concept of Component-Oriented Programming
  - 3.4.1. Context
  - 3.4.2. Components and Interfaces
  - 3.4.3. States of a Component
- 3.5. Component Architecture
  - 3.5.1. Current Architectures
  - 3.5.2. Component Integration and Deployment
- 3.6. Framework Front-End: Bootstrap
  - 3.6.1. Grid Design
  - 3.6.2. Forms
  - 3.6.3. Components
- 3.7. Model View Controller
  - 3.7.1. Web Development Methods
  - 3.7.2. Design Pattern: MVC
- 3.8. Information Grid Technologies
  - 3.8.1. Increased Computing Resources
  - 3.8.2. Concept of Grid Technology

- 3.9. Service-Oriented Architecture
  - 3.9.1. SOA and Web Services
  - 3.9.2. Topology of a Web Service
  - 3.9.3. Platforms for Web Services
- 3.10. HTTP Protocol
  - 3.10.1. Messages
  - 3.10.2. Persistent Sessions
  - 3.10.3. Cryptographic System
  - 3.10.4. HTTPS Protocol Operation

### Module 4. Free Software and Open Knowledge

- 4.1. Introduction to Free Software
  - 4.1.1. History of Free Software
  - 4.1.2. "Freedom" in Software
  - 4.1.3. Licenses for the Use of Software Tools
  - 4.1.4. Intellectual Property of Software
  - 4.1.5. What is the Motivation for Using Free Software?
  - 4.1.6. Free Software Myths
  - 4.1.7. Top500
- 4.2. Open Knowledge and CC Licenses
  - 4.2.1. Basic Concepts
  - 4.2.2. Creative Commons Licenses
  - 4.2.3. Other Content Licenses
  - 4.2.4. Wikipedia and Other Open Knowledge Projects
- 4.3. Main Free Software Tools
  - 4.3.1. Operating Systems
  - 4.3.2. Office Applications
  - 4.3.3. Business Management Applications
  - 4.3.4. Web Content Managers
  - 4.3.5. Multimedia Content Creation Tools
  - 4.3.6. Other Applications

- 4.4. The Company: Free Software and its Costs
  - 4.4.1. Free Software: Yes or No?
  - 4.4.2. Truths and Lies about Free Software
  - 4.4.3. Business Software Based on Free Software
  - 4.4.4. Software Costs
  - 4.4.5. Free Software Models
- 4.5. The GNU/Linux Operating System
  - 4.5.1. Architecture
  - 4.5.2. Basic Directory Structure
  - 4.5.3. File System Characteristics and Structure
  - 4.5.4. Internal Representation of the Files
- 4.6. The Android Mobile Operating System
  - 4.6.1. History
  - 4.6.2. Architecture
  - 4.6.3. Android Forks
  - 4.6.4. Introduction to Android Development
  - 4.6.5. Frameworks for Mobile Application Development
- 4.7. Website Creation with WordPress
  - 4.7.1. WordPress Features and Structure
  - 4.7.2. Creation of Sites on WordPress.com
  - 4.7.3. Installation and Configuration of WordPress on your own Server
  - 4.7.4. Installing Plugins and Extending WordPress
  - 4.7.5. Creation of WordPress Plugins
  - 4.7.6. WordPress Theme Creation
- 4.8. Free Software Trends
  - 4.8.1. Cloud Environments
  - 4.8.2. Monitoring Tools
  - 4.8.3. Operating Systems
  - 4.8.4. Big Data and Open Data 2.0
  - 4.8.5. Quantum Computing

- 4.9. Version Control
  - 4.9.1. Basic Concepts
  - 4.9.2. Git
  - 4.9.3. Cloud and Self-hosted Git Services
  - 4.9.4. Other Version Control Systems
- 4.10. Custom GNU/Linux Distributions
  - 4.10.1. Main Distributions
  - 4.10.2. Distributions Derived from Debian
  - 4.10.3. Deb Package Creation
  - 4.10.4. Modification of the Distribution
  - 4.10.5. ISO Image Generation

## Module 5. Advanced Databases

- 5.1. Introduction to the Different Database Systems
  - 5.1.1. Historical Recap
  - 5.1.2. Hierarchical Databases
  - 5.1.3. Network Databases
  - 5.1.4. Relational Databases
  - 5.1.5. Non-Relational Databases
- 5.2. XML and Databases for the Web
  - 5.2.1. Validation of XML Documents
  - 5.2.2. XML Document Transformations
  - 5.2.3. XML Data Storage
  - 5.2.4. XML Relational Databases
  - 5.2.5. SQL/XML
  - 5.2.6. Native XML Databases
- 5.3. Parallel Databases
  - 5.3.1. Parallel Systems
  - 5.3.2. Parallel Database Architectures
  - 5.3.3. Parallelism in Queries
  - 5.3.4. Query Parallelism
  - 5.3.5. Design of Parallel Systems
  - 5.3.6. Parallel Processing in SQL

- 5.4. Distributed Databases
  - 5.4.1. Distributed Systems
  - 5.4.2. Distributed Storage
  - 5.4.3. Availability
  - 5.4.4. Distributed Query Processing
  - 5.4.5. Distributed Database Providers
- 5.5. Indexing and Association
  - 5.5.1. Ordered Indexes
  - 5.5.2. Dense and Sparse Indexes
  - 5.5.3. Multilevel Indices
  - 5.5.4. Index Updating
  - 5.5.5. Static Association
  - 5.5.6. How to Use Indexes in Databases
- 5.6. Introduction to Transactional Processing
  - 5.6.1. States of a Transaction
  - 5.6.2. Implementation of Atomicity and Durability
  - 5.6.3. Sequentiality
  - 5.6.4. Recoverability
  - 5.6.5. Isolation Implementation
- 5.7. Recovery Systems
  - 5.7.1. Failure Classification
  - 5.7.2. Storage Structures
  - 5.7.3. Recovery and Atomicity
  - 5.7.4. Retrieval Based on Historical Record
  - 5.7.5. Concurrent Transactions and Retrieval
  - 5.7.6. High Availability in Databases
- 5.8. Execution and Processing of Queries
  - 5.8.1. Cost of a Query
  - 5.8.2. Selection Operation
  - 5.8.3. Sorting
  - 5.8.4. Introduction to Query Optimization
  - 5.8.5. Performance Monitoring



- 5.9. Non-Relational Databases
  - 5.9.1. Document-Oriented Databases
  - 5.9.2. Graph-Oriented Databases
  - 5.9.3. Key-Value Databases
- 5.10. Data Warehouse, OLAP and Data Mining
  - 5.10.1. Components of Data Warehouses
  - 5.10.2. Architecture of a Data Warehouse
  - 5.10.3. OLAP
  - 5.10.4. Data Mining Functionality
  - 5.10.5. Other Types of Mining

## Module 6. Software Engineering

- 6.1. Software Engineering Framework
  - 6.1.1. Software Features
  - 6.1.2. The Main Processes in Software Engineering
  - 6.1.3. Software Development Process Models
  - 6.1.4. Standard Reference Framework for the Software Development Process: ISO/IEC 12207 Regulations
- 6.2. Unified Software Development Process
  - 6.2.1. The Unified Process
  - 6.2.2. Dimensions of the Unified Process
  - 6.2.3. Case Studies Driven Development Process
  - 6.2.4. Fundamental Unified Process Workflows
- 6.3. Planning in the Context of Agile Software Development
  - 6.3.1. Characteristics of Agile Software Development
  - 6.3.2. Different Planning Time Horizons in Agile Development
  - 6.3.3. Scrum Agile Development Framework and Planning Time Horizons
  - 6.3.4. User Stories as a Planning and Estimating Unit
  - 6.3.5. Common Techniques for Deriving an Estimate
  - 6.3.6. Scales for Interpreting Estimates
  - 6.3.7. Planning Poker
  - 6.3.8. Common Types of Planning: Delivery and Iteration Planning
- 6.4. Distributed Software Design Styles and Service-Oriented Software Architectures
  - 6.4.1. Communication Models in Distributed Software Systems
  - 6.4.2. Middleware
  - 6.4.3. Architecture Patterns for Distributed Systems
  - 6.4.4. General Software Service Design Process
  - 6.4.5. Design Aspects of Software Services
  - 6.4.6. Composition of Services
  - 6.4.7. Web Services Architecture
  - 6.4.8. Infrastructure and SOA Components
- 6.5. Introduction to Model Driven Software Development
  - 6.5.1. The Model Concept
  - 6.5.2. Model-Driven Software Development
  - 6.5.3. MDA Model-Driven Development Framework
  - 6.5.4. Elements of a Transformation Model
- 6.6. Graphical User Interface Design
  - 6.6.1. Principles of User Interface Design
  - 6.6.2. Architectural Design Patterns for Interactive Systems: Model View Controller (MVC)
  - 6.6.3. UX (User Experience)
  - 6.6.4. User-Centered Design
  - 6.6.5. Graphical User Interface Analysis and Design Process
  - 6.6.6. Usability of User Interfaces
  - 6.6.7. Accessibility in User Interfaces
- 6.7. Web Application Design
  - 6.7.1. Characteristics of Web Applications
  - 6.7.2. Web Application User Interface
  - 6.7.3. Navigation Design
  - 6.7.4. Basic Interaction Protocol for Web Applications
  - 6.7.5. Architecture Styles for Web Applications
- 6.8. Software Testing Strategies and Techniques and Software Quality Factors
  - 6.8.1. Testing Strategies
  - 6.8.2. Test Case Designs
  - 6.8.3. Value for Money
  - 6.8.4. Quality Models
  - 6.8.5. ISO/IEC 25000 Family of Standards (SQuaRE)

- 6.8.6. Product Quality Model (ISO 2501n)
- 6.8.7. Data Quality Models (ISO 2501n)
- 6.8.8. Software Quality Management
- 6.9. Introduction to Software Engineering Metrics
  - 6.9.1. Basic Concepts: Measures, Metrics and Indicators
  - 6.9.2. Types of Metrics in Software Engineering
  - 6.9.3. The Measurement Process
  - 6.9.4. ISO 25024. External and Quality Metrics in Use
  - 6.9.5. Object-Oriented Metrics
- 6.10. Software Maintenance and Reengineering
  - 6.10.1. Maintenance Process
  - 6.10.2. Standard Maintenance Process Framework. ISO/EIEC 14764
  - 6.10.3. Software Reengineering Process Model
  - 6.10.4. Inverse Engineering

## Module 7. Advanced Programming

- 7.1. Introduction to Object-Oriented Programming
  - 7.1.1. Introduction to Object-Oriented Programming
  - 7.1.2. Class Design
  - 7.1.3. Introduction to UML for Problem Modeling
- 7.2. Relationships Between Classes
  - 7.2.1. Abstraction and Inheritance
  - 7.2.2. Advanced Inheritance Concepts
  - 7.2.3. Polymorphism
  - 7.2.4. Composition and Aggregation
- 7.3. Introduction to Design Patterns for Object-Oriented Problems
  - 7.3.1. What are Design Patterns?
  - 7.3.2. Factory Pattern
  - 7.3.4. Singleton Pattern
  - 7.3.5. Observer Pattern
  - 7.3.6. Composite Pattern

- 7.4. Exceptions
  - 7.4.1. What are Exceptions?
  - 7.4.2. Exception Catching and Handling
  - 7.4.3. Throwing Exceptions
  - 7.4.4. Exception Creation
- 7.5. User Interfaces
  - 7.5.1. Introduction to Qt
  - 7.5.2. Positioning
  - 7.5.3. What Are Events?
  - 7.5.4. Events: Definition and Capture
  - 7.5.5. User Interface Development
- 7.6. Introduction to Concurrent Programming
  - 7.6.1. Introduction to Concurrent Programming
  - 7.6.2. The Concept of Process and Thread
  - 7.6.3. Interaction Between Processes or Threads
  - 7.6.4. Threads in C++
  - 7.6.5. Advantages and Disadvantages of Concurrent Programming
- 7.7. Thread Management and Synchronization
  - 7.7.1. Life Cycle of a Thread
  - 7.7.2. Thread Class
  - 7.7.3. Thread Planning
  - 7.7.4. Thread Groups
  - 7.7.5. Daemon Threads
  - 7.7.6. Synchronization
  - 7.7.7. Locking Mechanisms
  - 7.7.8. Communication Mechanisms
  - 7.7.9. Monitors
- 7.8. Common Problems in Concurrent Programming
  - 7.8.1. The Problem of Consuming Producers
  - 7.8.2. The Problem of Readers and Writers
  - 7.8.3. The Problem of the Philosophers' Dinner Party

- 7.9. Software Documentation and Testing
  - 7.9.1. Why is it Important to Document Software?
  - 7.9.2. Design Documentation
  - 7.9.3. Documentation Tool Use
- 7.10. Software Testing
  - 7.10.1. Introduction to Software Testing
  - 7.10.2. Types of Tests
  - 7.10.3. Unit Test
  - 7.10.4. Integration Test
  - 7.10.5. Validation Test
  - 7.10.6. System Test

## Module 8. Software Reuse

- 8.1. General Overview of the Software Reuse
  - 8.1.1. What is Software Reuse?
  - 8.1.2. Advantages and Disadvantages of Software Reuse
  - 8.1.3. Main Techniques of Software Reuse
- 8.2. Introduction to Design Patterns
  - 8.2.1. What is a Design Patterns?
  - 8.2.2. Catalog of the Main Design Patterns
  - 8.2.3. How to Use Patterns to Solve Design Problems
  - 8.2.4. How to Select the Best Design Pattern
- 8.3. Creation Patterns (I)
  - 8.3.1. Creation Patterns
  - 8.3.2. Abstract Factory Pattern
  - 8.3.3. Example of Abstract Factory Pattern implementation
  - 8.3.4. Builder Pattern
  - 8.3.5. Builder Implementation Example
  - 8.3.6. Abstract Factory Pattern vs. Builder
- 8.4. Creation Patterns (II)
  - 8.4.1. Factory Method Pattern
  - 8.4.2. Factory Method vs. Abstract Factory
  - 8.4.3. Singleton Pattern
- 8.5. Structural Patterns (I)
  - 8.5.1. Structural Patterns
  - 8.5.2. Adapter Pattern
  - 8.5.3. Bridge Pattern
- 8.6. Structural Patterns (II)
  - 8.6.1. Composite Pattern
  - 8.6.2. Decorator Pattern
- 8.7. Structural Patterns (III)
  - 8.7.1. Facade Pattern
  - 8.7.2. Proxy Pattern
- 8.8. Behavioral Patterns (I)
  - 8.8.1. Concept of Behavioral Patterns
  - 8.8.2. Pattern of Behavior: Responsibility Chain
  - 8.8.3. Behavior Pattern Order
- 8.9. Behavioral Patterns (II)
  - 8.9.1. Interpreter Pattern
  - 8.9.2. Iterator Pattern
  - 8.9.3. Observer Pattern
  - 8.9.4. Strategy Pattern
- 8.10. Frameworks
  - 8.10.1. Concept of Framework
  - 8.10.2. Development Using Frameworks
  - 8.10.3. Model View Controller Pattern
  - 8.10.4. Framework for Graphical User Interface Design
  - 8.10.5. Frameworks for Web Application Development
  - 8.10.6. Frameworks for Managing Object Persistence in Databases

## Module 9. Artificial Intelligence and Knowledge Engineering

- 9.1. Introduction to Artificial Intelligence and Knowledge Engineering
  - 9.1.1. Brief History of Artificial Intelligence
  - 9.1.2. Artificial Intelligence Today
  - 9.1.3. Knowledge Engineering
- 9.2. Searching
  - 9.2.1. Common Search Concepts
  - 9.2.2. Uninformed Search
  - 9.2.3. Informed Search
- 9.3. Boolean Satisfiability, Constraint Satisfiability and Automatic Planning
  - 9.3.1. Boolean Satisfiability
  - 9.3.2. Constraint Satisfiability Problems
  - 9.3.3. Automatic Planning and PDDL
  - 9.3.4. Planning as Heuristic Search
  - 9.3.5. Planning with SAT
- 9.4. Artificial Intelligence in Games
  - 9.4.1. Game Theory
  - 9.4.2. Minimax and Alpha-Beta Pruning
  - 9.4.3. Simulation: Monte Carlo
- 9.5. Supervised and Unsupervised Learning
  - 9.5.1. Introduction to Machine Learning
  - 9.5.2. Classification
  - 9.5.3. Regression
  - 9.5.4. Validation of Results
  - 9.5.5. Clustering
- 9.6. Neural Networks
  - 9.6.1. Biological Fundamentals
  - 9.6.2. Computational Model
  - 9.6.3. Supervised and Unsupervised Neuron Networks
  - 9.6.4. Simple Perceptron
  - 9.6.5. Multilayer Perceptron

- 9.7. Genetic Algorithms
  - 9.7.1. History
  - 9.7.2. Biological Basis
  - 9.7.3. Problem Coding
  - 9.7.4. Generation of the Initial Population
  - 9.7.5. Main Algorithm and Genetic Operators
  - 9.7.6. Evaluation of Individuals: Fitness
- 9.8. Thesauri, Vocabularies, Taxonomies
  - 9.8.1. Vocabulary
  - 9.8.2. Taxonomy
  - 9.8.3. Thesauri
  - 9.8.4. Ontologies
- 9.9. Knowledge Representation Semantic Web
  - 9.9.1. Semantic Web
  - 9.9.2. Specifications RDF, RDFS and OWL
  - 9.9.3. Inference/ Reasoning
  - 9.9.4. Linked Data
- 9.10. Expert systems and DSS
  - 9.10.1. Expert Systems
  - 9.10.2. Decision Support Systems

## Module 10. Advanced Software Engineering

- 10.1. Introduction to Agile Methodologies
  - 10.1.1. Process Models and Methodologies
  - 10.1.2. Agility and Agile Processes
  - 10.1.3. Agile Manifesto
  - 10.1.4. Some Agile Methodologies
  - 10.1.5. Agile vs. Traditional
- 10.2. Scrum
  - 10.2.1. Origins and Philosophy of Scrum
  - 10.2.2. Scrum Values
  - 10.2.3. Scrum Process Flow
  - 10.2.4. Scrum Roles
  - 10.2.5. Scrum Artifacts

- 10.2.6. Scrum Events
- 10.2.7. User Stories
- 10.2.8. Scrum Extensions
- 10.2.9. Agile Estimates
- 10.2.10. Scrum Scaling
- 10.3. Extreme Programming
  - 10.3.1. Justification and Overview of XP
  - 10.3.2. The XP Life Cycle
  - 10.3.3. The Five Core Values
  - 10.3.4. The Twelve Basic Practices in XP
  - 10.3.5. Roles of Participants
  - 10.3.6. XP Industrial
  - 10.3.7. Critical Assessment of XP
- 10.4. Software Development Based on Reusability
  - 10.4.1. Software Reuse
  - 10.4.2. Code Reuse Levels
  - 10.4.3. Specific Reuse Techniques
  - 10.4.4. Component-Based Development
  - 10.4.5. Benefits and Problems of Reuse
  - 10.4.6. Reuse Planning
- 10.5. System Architecture and Software Design Patterns
  - 10.5.1. Architectural Design
  - 10.5.2. General Architectural Patterns
  - 10.5.3. Fault Tolerant Architectures
  - 10.5.4. Distributed Systems Architectures
  - 10.5.5. Design Patterns
  - 10.5.6. Gamma Patterns
  - 10.5.7. Interaction Design Patterns
- 10.6. Cloud Application Architecture
  - 10.6.1. Cloud Computing Fundamentals
  - 10.6.2. Cloud Application Quality
  - 10.6.3. Architectural Styles
  - 10.6.4. Design Patterns
- 10.7. Software Testing: TDD, ATDD and BDD
  - 10.7.1. Software Verification and Validation
  - 10.7.2. Software Testing
  - 10.7.3. Test Driven Development (TDD)
  - 10.7.4. Acceptance Test Driven Development (ATDD)
  - 10.7.5. Behavior Driven Development (BDD)
  - 10.7.6. BDD and Cucumber
- 10.8. Software Process Improvement
  - 10.8.1. Software Process Improvement
  - 10.8.2. The Process Improvement Approach
  - 10.8.3. Maturity Models
  - 10.8.4. The CMMI Model
  - 10.8.5. CMMI V2.0
  - 10.8.6. CMMI and Agile
- 10.9. The Quality of the Software Product: SQuaRE
  - 10.9.1. Software Quality
  - 10.9.2. Software Product Quality Models
  - 10.9.3. ISO/IEC 25000 Family
  - 10.9.4. ISO/IEC 25010: Model and Quality Characteristics
  - 10.9.5. ISO/IEC 25012: Data Quality
  - 10.9.6. ISO/IEC 25020: Software Quality Measurement
  - 10.9.7. ISO/IEC 25022, 25023 y 25024: Software and Data Quality Metrics
  - 10.9.8. ISO/IEC 25040: Evaluation of the Software
  - 10.9.9. Accreditation Process
- 10.10. Introduction to DevOps
  - 10.10.1. DevOps Concept
  - 10.10.2. Core Practices

06

# Clinical Internship

After passing the online theoretical period, the program includes a period of practical training in a leading IT company. Therefore, the graduate will have at his disposal the support of a personal tutor who will accompany him throughout the process, both in the preparation and in the development of the internship.



“

*You will participate in practical projects that require the application of the theoretical concepts learned, allowing you to develop innovative and functional solutions"*

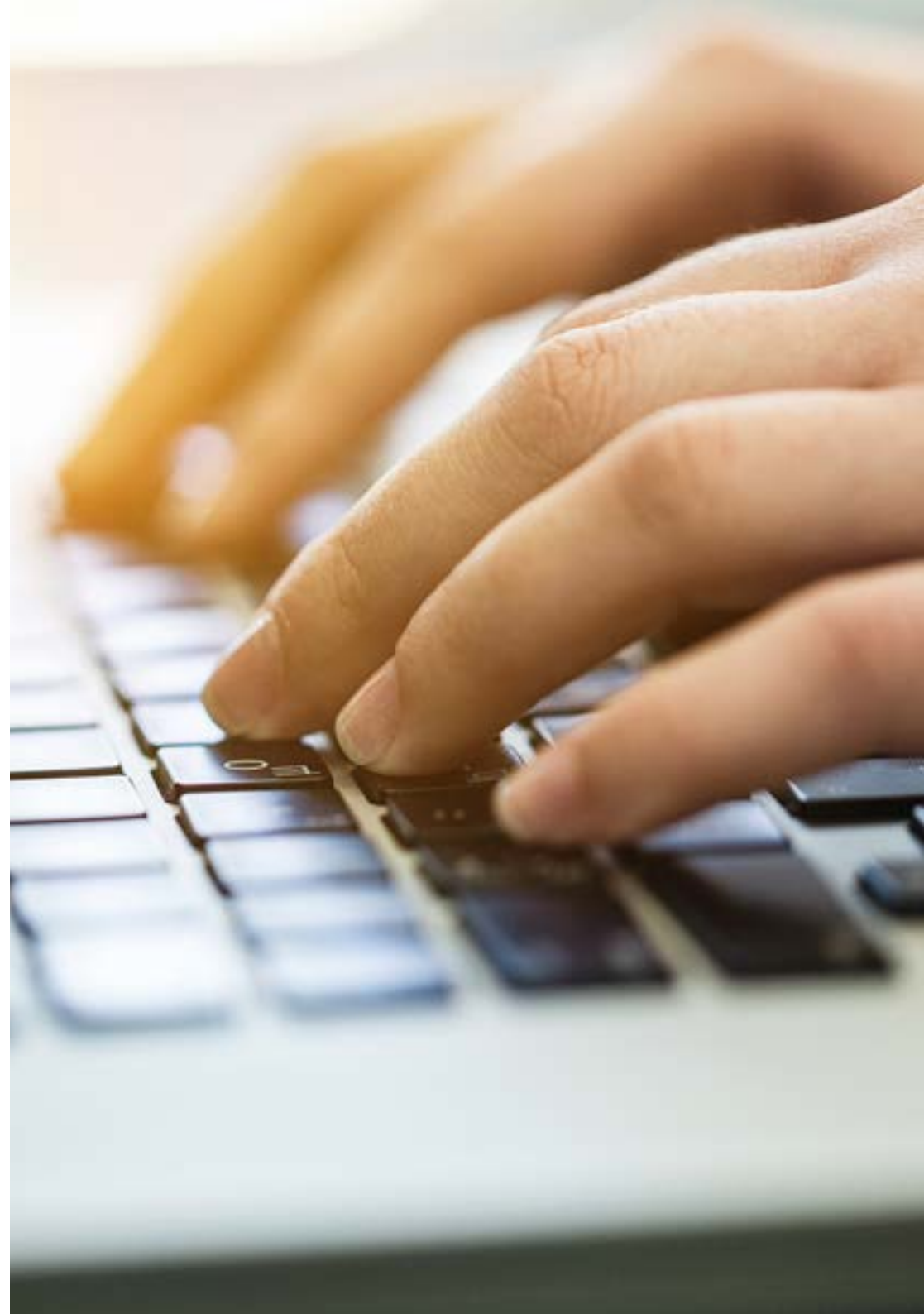
The period of the Internship Program of this program in Creation of Interfaces and Network Applications consists of a 3-week practical internship in a prestigious IT company, from Monday to Friday, with 8 consecutive hours of practical training with a specialist. During this stay, the graduate will be able to apply the theoretical knowledge acquired to real situations, consolidating their understanding and skills in the design and development of web interfaces and applications.

This program proposal, completely practical in nature, is aimed at involving students in real web application development projects, from the conceptualization phase to their implementation and deployment. This will allow them to face technical and creative challenges, as well as learn to work in a team, manage resources and meet demanding deadlines.

In addition, this is a great opportunity to learn by working, experimenting with various tools and cutting-edge technologies, preparing professionals to face the challenges of the ever-changing world of work. All this working in a collaborative environment with experts in the field, providing a broader and updated view of emerging trends and practices in the field of information technology and communication.

The practical part will be carried out with the active participation of the student performing the activities and procedures of each area of competence (learning to learn and learning to do), with the accompaniment and guidance of teachers and other training partners that facilitate teamwork and multidisciplinary integration as transversal competencies for the praxis of the creation of interfaces and web applications (learning to be and learning to relate).

The procedures described below will be the basis of the practical part of the program, and their implementation will be subject to the center's own availability and workload, the proposed activities being the following:



| Module           | Practical Activity                                                              |
|------------------|---------------------------------------------------------------------------------|
| Interface Design | Design intuitive and attractive user interfaces                                 |
|                  | Conduct usability testing to improve user experience                            |
|                  | Implement security measures to protect sensitive information                    |
|                  | Implement data analytics systems to improve decision making                     |
| Web Applications | Develop responsive and high-performance web applications                        |
|                  | Integrate Content Management Systems (CMS) to facilitate content administration |
|                  | Optimize the speed and efficiency of web applications                           |
|                  | Collaborate in digital marketing strategy to promote web applications           |
| Accessibility    | Adapt interfaces and applications for various devices and platforms             |
|                  | Incorporate accessibility techniques to ensure inclusion of all users           |
|                  | Prepare end-users in the use of developed applications                          |
|                  | Adapt interfaces and applications to different languages and cultures           |

| Module                        | Practical Activity                                                                                       |
|-------------------------------|----------------------------------------------------------------------------------------------------------|
| Documentation and Maintenance | Investigate and resolve technical issues related to the interface and the application                    |
|                               | Document the development process and keep technical documentation up to date                             |
|                               | Manage the complete application lifecycle, from development to maintenance and upgrades                  |
|                               | Keep up to date on privacy and data security related regulations and standards privacy and data security |
| Research and Collaboration    | Keep abreast of the latest trends and technologies in web design and application development             |
|                               | Collaborate with cross-functional teams to ensure consistency between design and functionality           |
|                               | Create prototypes and perform proof-of-concepts for new functionality                                    |
|                               | Analyze user data to identify areas for improvement in interface and functionality                       |



*You will collaborate closely with experienced professionals in the field, understanding the fundamental principles of interface design and web application development"*

## Civil Liability Insurance

This institution's main concern is to guarantee the safety of the trainees and other collaborating agents involved in the internship process at the company. Among the measures dedicated to achieve this is the response to any incident that may occur during the entire teaching-learning process.

To this end, this entity commits to purchasing a civil liability insurance policy to cover any eventuality that may arise during the course of the internship at the center.

This liability policy for interns will have broad coverage and will be taken out prior to the start of the practical training period. That way professionals will not have to worry in case of having to face an unexpected situation and will be covered until the end of the Internship Program at the center.



## General Conditions of the Internship Program

The general terms and conditions of the internship program agreement shall be as follows: follows:

**1. TUTOR:** During the Hybrid Master's Degree, students will be assigned with two tutors who will accompany them throughout the process, answering any doubts and questions that may arise. On the one hand, there will be a professional tutor belonging to the internship center who will have the purpose of guiding and supporting the student at all times. On the other hand, they will also be assigned with an academic tutor whose mission will be to coordinate and help the students during the whole process, solving doubts and facilitating everything they may need. In this way, the student will be accompanied and will be able to discuss any doubts that may arise, both clinical and academic.

**2. DURATION:** The internship program will have a duration of three continuous weeks, in 8-hour days, 5 days a week. The days of attendance and the schedule will be the responsibility of the center and the professional will be informed well in advance so that they can make the appropriate arrangements.

**3. ABSENCE:** If the students does not show up on the start date of the Hybrid Master's Degree, they will lose the right to it, without the possibility of reimbursement or change of dates. Absence for more than two days from the internship, without justification or a medical reason, will result in the professional's withdrawal from the internship, therefore, automatic termination of the internship. Any problems that may arise during the course of the internship must be urgently reported to the academic tutor.

**4. CERTIFICATION:** Professionals who pass the Hybrid Master's Degree will receive a certificate accrediting their stay at the center.

**5. EMPLOYMENT RELATIONSHIP:** the Hybrid Master's Degree shall not constitute an employment relationship of any kind.

**6. PRIOR EDUCATION:** Some centers may require a certificate of prior education for the Hybrid Master's Degree. In these cases, it will be necessary to submit it to the TECH internship department so that the assignment of the chosen center can be confirmed.

**7. DOES NOT INCLUDE:** The Hybrid Master's Degree will not include any element not described in the present conditions. Therefore, it does not include accommodation, transportation to the city where the internship takes place, visas or any other items not listed.

However, students may consult with their academic tutor for any questions or recommendations in this regard. The academic tutor will provide the student with all the necessary information to facilitate the procedures in any case.

07

# Where Can I Do the Internship?

This Hybrid Master's Degree program includes in its itinerary a practical stay in a prestigious IT company, where students will put into practice everything they have learned in the field of Network Interfaces and Applications Creation. In this sense, and to bring this degree to more professionals, TECH offers the opportunity to study it in different organizations around the country. In this way, this institution strengthens its commitment to quality and affordable education for all.



“

*Complete your theoretical education with the best practical stay in the market. You will achieve success in your career in an agile and simple way thanks to TECH!"*

## tech 42 | Where Can I Do the Internship?



The student will be able to complete the practical part of this Hybrid Master's Degree at the following centers:



IT specialist

### Ogilvy Barcelona

|         |           |
|---------|-----------|
| Country | City      |
| Spain   | Barcelona |

Address: Calle Bolivia 68-70, 08018, Barcelona

Ogilvy is a pioneer in Pervasive Advertising, Marketing and Corporate Communications.

---

**Related internship programs:**

- Artificial Intelligence in Design
- Personal Brand Construction



“

*Boost your career path with holistic teaching, allowing you to advance both theoretically and practically"*

# 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.



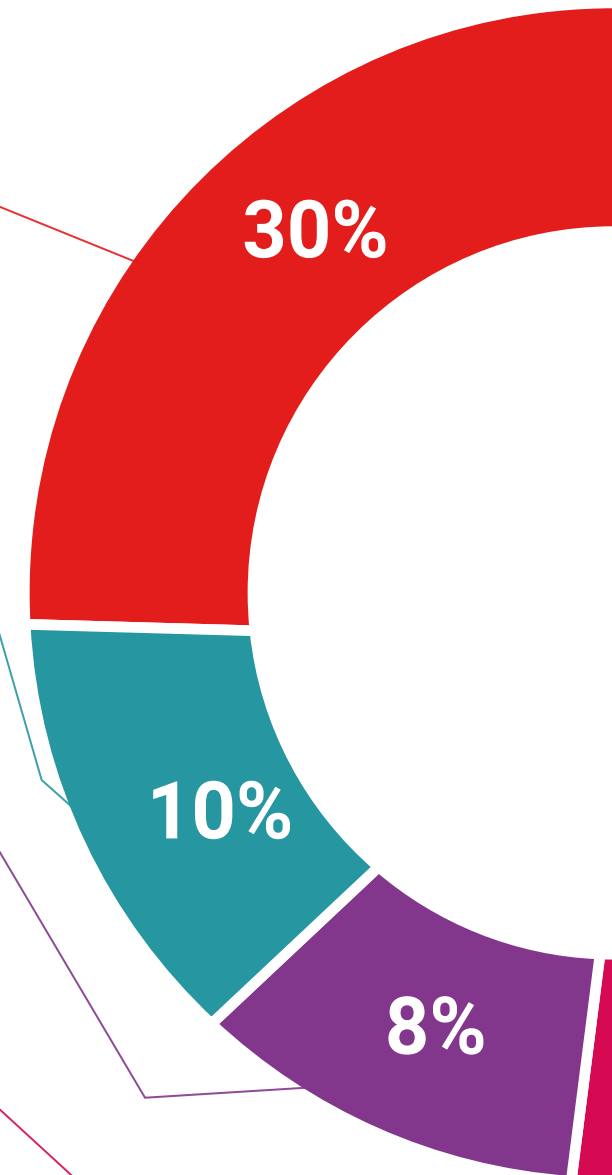
#### Practising Skills and Abilities

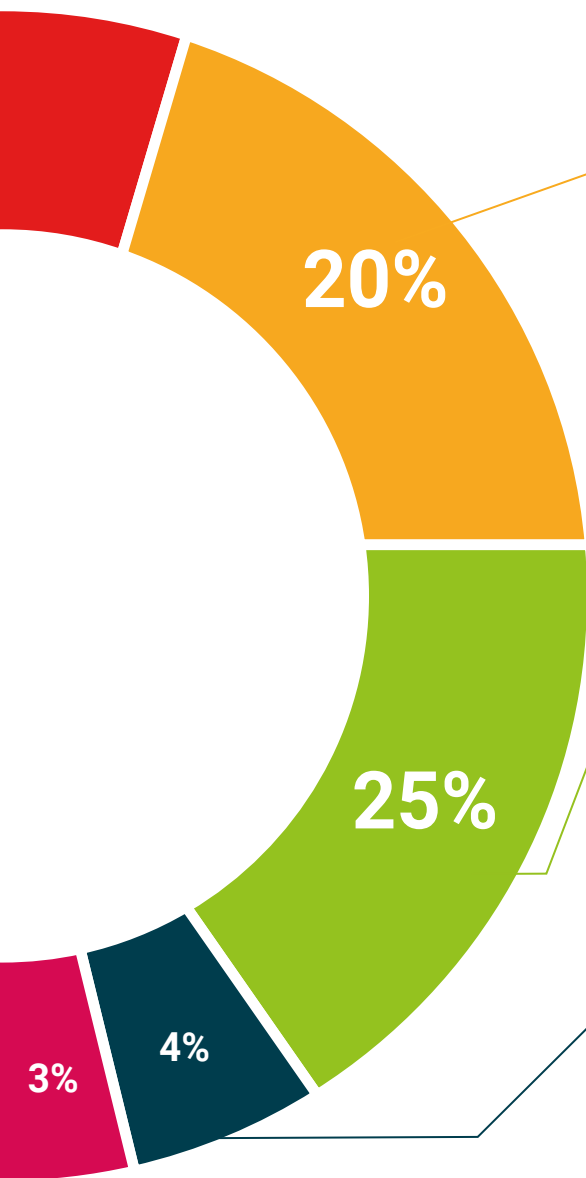
They will carry out activities to develop specific skills and abilities in each subject area. Exercises and activities to acquire and develop the skills and abilities that a specialist needs to develop in the context of the globalization that we are experiencing.



#### 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.



# 09 Certificate

The Hybrid Master's Degree in Creation of Network Interfaces and Applications guarantees students, in addition to the most rigorous and up-to-date education, access to a Hybrid Master's Degree diploma issued by TECH Global University.



“

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

This private qualification will allow you to obtain a **Hybrid Master's Degree in Creation of Network Interfaces and Applications** 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.

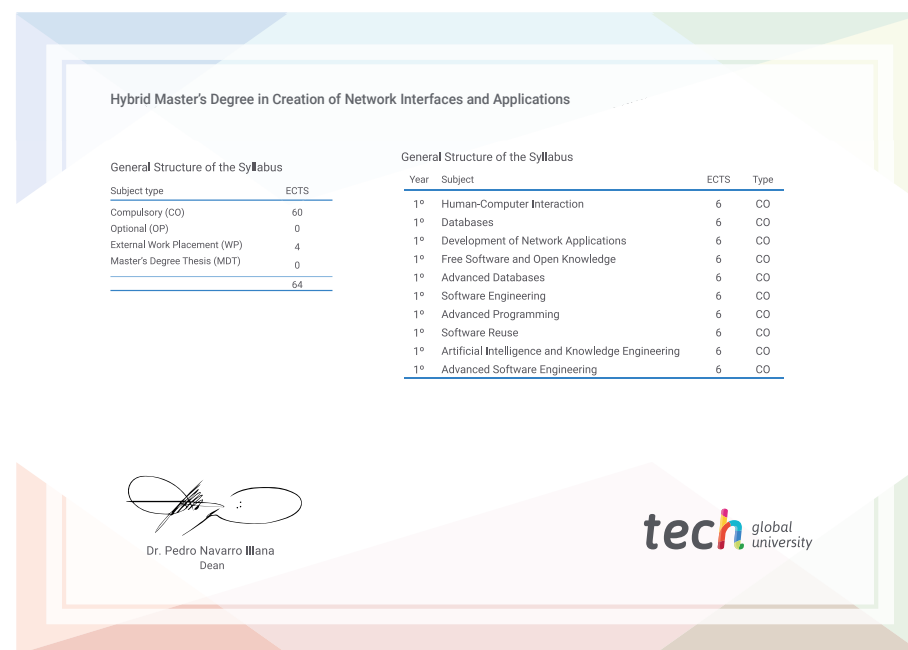
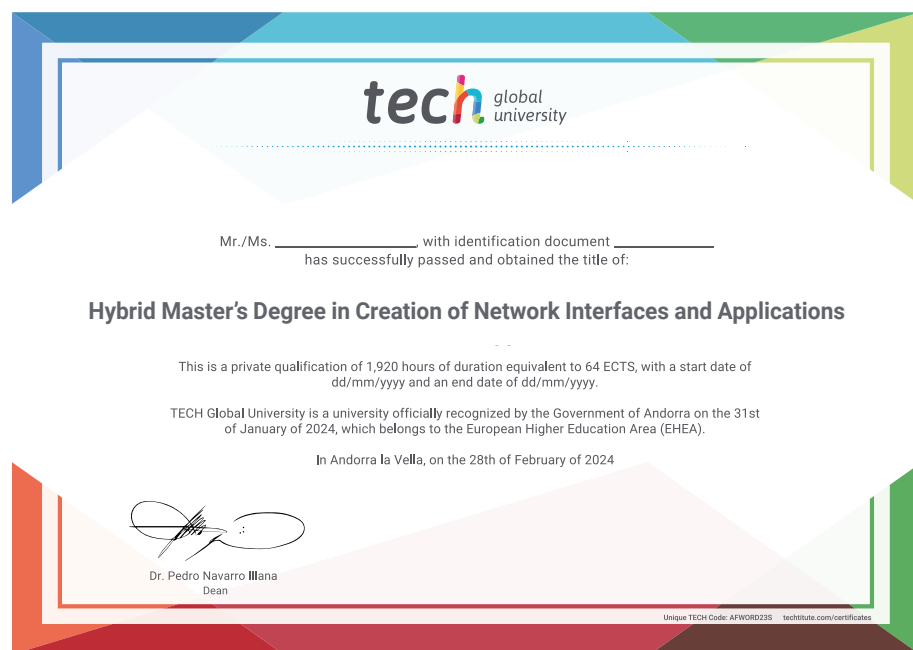
This **TECH Global University** private qualification is a European program of continuing education and professional updating that guarantees the acquisition of competencies in its area of knowledge, providing a high curricular value to the student who completes the program.

Title: **Hybrid Master's Degree in Creation of Network Interfaces and Applications**

Modality: **Hybrid (Online + Internship)**

Duration: **12 months.**

Accreditation: **64 ECTS Credits**





## Hybrid Master's Degree Creation of Network Interfaces and Applications

Modality: Hybrid (Online + Internship)

Duration: 12 months

Certificate: TECH Global University

Credits: 60 + 4 ECTS

# Hybrid Master's Degree

Creation of Network Interfaces  
and Applications