Syrian Arab Republic

Higher Institute for Applied Sciences and Technology

Course Specification Document

Title	Fundamentals of Distributed Systems

Credits	5 ECTS

Aims	This course aims to introduce the student to the fundamentals of distributed systems, their challenges, construction techniques, and traditional algorithms that address some of these challenges. It serves as an extension of operating systems courses, providing solutions to problems encountered when connecting multiple computer platforms and operating systems within a network, whether on a small scale (cluster) or on a larger scale (Intranet, Internet). By the end of the course, the student will have grasped the basics of distributed systems and will be equipped to develop programs and distributed systems within distributed computing environments.

Intended learning outcomes

On successful completion of this course, the student will be able to:

- Explain the concepts of distributed systems.
- Understand the problems and challenges of distributed systems.
- Familiarize himself with fundamental techniques and algorithms for building and developing distributed systems.
- Design and implement distributed applications within the available computing environments.

Syllabus

- **Introduction to distributed systems:** Definition of distributed system, advantages and challenges of distributed systems, hardware alternatives for distributed systems, types of distributed systems and their functions and characteristics, middleware and some of its models, cluster computing, and mobile distributed systems.
- **Distributed system models:** Client/Server (C/S) architecture and its types, studying faults in C/S architecture, communication in distributed systems, studying Remote Procedure Call (RPC) and some of its design issues, studying Remote Method Invocation (RMI) and its issues, object server and object adapter, multi-threading in distributed systems, group communication, fundamentals of code migration, and mobile agent.
- **Web-based distributed systems:** Traditional web system, multi-tier web architecture, web services, web processes, web server clusters, communication protocols in web systems, Web caching and replication.
- **Event ordering:** Logical clocks technique, studying a distributed application using RPC, studying sequence number generator (sequencer) and its distributed implementations, election algorithms.
- **Synchronization in distributed systems:** Concept of synchronization in distributed systems, designing centralized and distributed synchronization algorithms in distributed systems.

- **Consistency and replication:** Fundamental concepts and types of consistencies, various consistency algorithms.
- **Naming:** Fundamental naming concepts, naming issues in distributed systems, namespace and its implementations, case studies (DNS, X.500), locating mobile entities, issues & solutions.
- **Distributed file systems:** Architecture and models of distributed file systems, communication in distributed file systems, naming in distributed file systems, synchronization, consistency, and replication, case studies.
- **Deadlock problem in distributed systems:** Defining the deadlock problem and its modeling, strategies for handling deadlock in distributed systems (avoidance, prevention, detection and recovery).
- **Distributed system design - Google case study:** architecture and design, core communication models, storage and data formatting services, distributed computation services.
- **Practical sessions:**
  - **Practical RPC:** Understanding standard RPC package in Windows and Linux environments and implementing applications using it.
  - **Practical RMI:** Building distributed applications in Java using RMI.
  - **Building a distributed application:** Designing and implementing a distributed software application using the studied techniques.