



Faculty of Engineering and Technology

Computer Science Department

Second Semester 2024/2025

1. Course information:

A. Course Code: COMP4382

B. Course Name: SP.TOP: WEB SERVICES TECHNOLOGY

C. Prerequisite: COMP230-Computer and Programming, COMP333-Database Systems.

2. Instructor: Ahmad Hamo, **Email:** hahmad@birzeit.edu

Office hour information is available on Ritaj.

3. Course description

Web service and APIs are powerful methods used in modern enterprise systems to achieve seamless applications interaction and integration.

This course introduces the student to web services concepts and design patterns. It describes and compares various architectural styles used in service-oriented systems with emphasis on RESTful services as one of the most popular architectural styles used in APIs implementation.

4. Course goals

A. Build a solid understanding of the fundamental concepts of web services, and its usages and advantages.

B. Explain different implementations and architectural styles of web services, including the SOAP standard and the Representational state transfer (REST). C. Introduce the student to the current trends in software architectures like SOA and microservices and how to approach them using web services.

D. Train the student on web service planning, design, implementation, deployment and testing using the latest technologies.

5. Course Outcomes

At the end of this course, student will have developed skills and abilities in the

following areas:

A. Knowledge and concepts understanding

1. Understand the usage and benefits of web services.
2. Compare and differentiate between RESTful APIs and SOAP web services and describe the pros and cons of both architectural styles.
3. Describe the basics of SOA and Microservices software architectures.
4. Design software application with a Restful APIs approach.
5. Web services security.
6. Leverage and benefit from Cloud Computing and public cloud providers to build robust, scalable and highly available software applications APIs.

B. Communication, teamwork, and presentation Skills

1. Cooperate and work both individually and in groups.
2. Presentation and research skills and be able to explore new topics in

technology. **C. Practical and technical Skills**

1. Develop skills in implementing web service and how to consume and process data returned from public APIs vendors.
2. Develop skills in design, building, maintaining, and deploying services on private and public cloud infrastructure.
3. Plan, design, develop and document an end-to-end backend application with exposing all resources through RESTful APIs with adopting the microservice architecture.

6. Course content

Week	Topics
1-4	Introduction to web services and Services Architectural Styles
1	Introduction to API and web services: definition, history, background and evolution of web services and API.

2,3,4	<ol style="list-style-type: none"> 1. Introduction to REST: REST architectural style, concepts, usage and advantages. 2. JSON & XML 3. REST client: Consuming RESTful services. 4. SOAP protocol: history, usage and overview of SOA components. 5. REST vs SOAP.
5,6	<p>Web Services and APIs in Practice</p> <ol style="list-style-type: none"> 1. Web services and Cloud Computing: overview of Amazon Web services 2. Building RESTful Service using FASTAPI. Publishing Restful Service on the Cloud.
7,8	<p>Web Services Security</p> <ol style="list-style-type: none"> 1. Introduction to Web services security and authorization 2. OAuth 2.0 protocols for securing APIs and Web services. 3. Research Paper or Presentation on one of the following: <ul style="list-style-type: none"> • Web services Security Challenges. • SOAP vs REST, which one is more secure? • Security Assertion Markup Language (SAML) for Web service authentication. 4. The difference between web services authorization and authentication.
9-10	<p>Building Web Application and Restful APIs using cloud technologies</p>
11-12	<p>Modern Application Architectures and Web services</p> <ol style="list-style-type: none"> 1. Overview of Service Oriented Architecture. 2. Introduction to Microservices Architecture
13, 14,15	<p>Practical Project presentation</p>

6. Teaching and learning methods

- A. Lectures.
- B. Assignments/projects and quizzes.
- C. Presentations and Exams.

7. Assessment methods based on outcomes

1. Mid and final exams to assess the student's knowledge and concepts understanding.
2. Projects and assignments to assess the student's practical skills.

8. Weighting of assessments

Research, presentation, assignments and project	40%
Midterm	25%
Final Exam	35%

9. Student responsibilities:

3. Class participation and independent work. Students are expected to actively participate in all classes and perform independent work.
4. Attendance. Attendance is mandatory. University regulations regarding this matter will be strictly enforced.
5. Academic Honesty. Individual work must be each student's own work. Plagiarism or cheating will result in official University disciplinary review.
6. Missed Exams. There are no makeup exams.
7. Class Etiquette. Please keep all cell phones and other electronic devices turned off during class. If your activities during class are deemed disruptive, you will be asked to leave.

10. References

1. Pro RESTful APIs: Design, Build and Integrate with REST, JSON, XML and JAX-RS (1st Edition) by Sanjay Patni 2017.
2. Undisturbed REST: a Guide to Designing the Perfect API (1st Edition) by Michael Stowe 2015.
3. Building Python Web APIs with FastAPI by Abdulazeez Adeshina 2022
4. Manning - Microservice APIs - Using Python, Flask, FastAPI, OpenAPI and more by José Haro Peralte 2023