



UNIVERSITÀ
DEGLI STUDI DI TRIESTE



Corso di Laurea in Ingegneria Clinica e Biomedica Informatica Medica I

WEB SERVICES, REST API, JSON

Prof. Sara Renata Francesca Marceglio

WEB SERVICES

“A Web service is a **software system** designed to support **interoperable machine-to-machine interaction over a network**. It has an interface described in a machine-processable format (specifically WSDL). Other systems interact with the Web service in a manner prescribed by its description using SOAP-messages, typically conveyed using HTTP with an XML serialization in conjunction with other Web-related standards.”

W3C – World Wide Web Consortium

*

WSDL = Web Service Description Language

SOAP = Service Oriented Architecture Programming

XML = eXtensible Markup Language

**I web services sono indipendenti
dalla piattaforma su cui operano,
quindi supportano
l'interoperabilità**

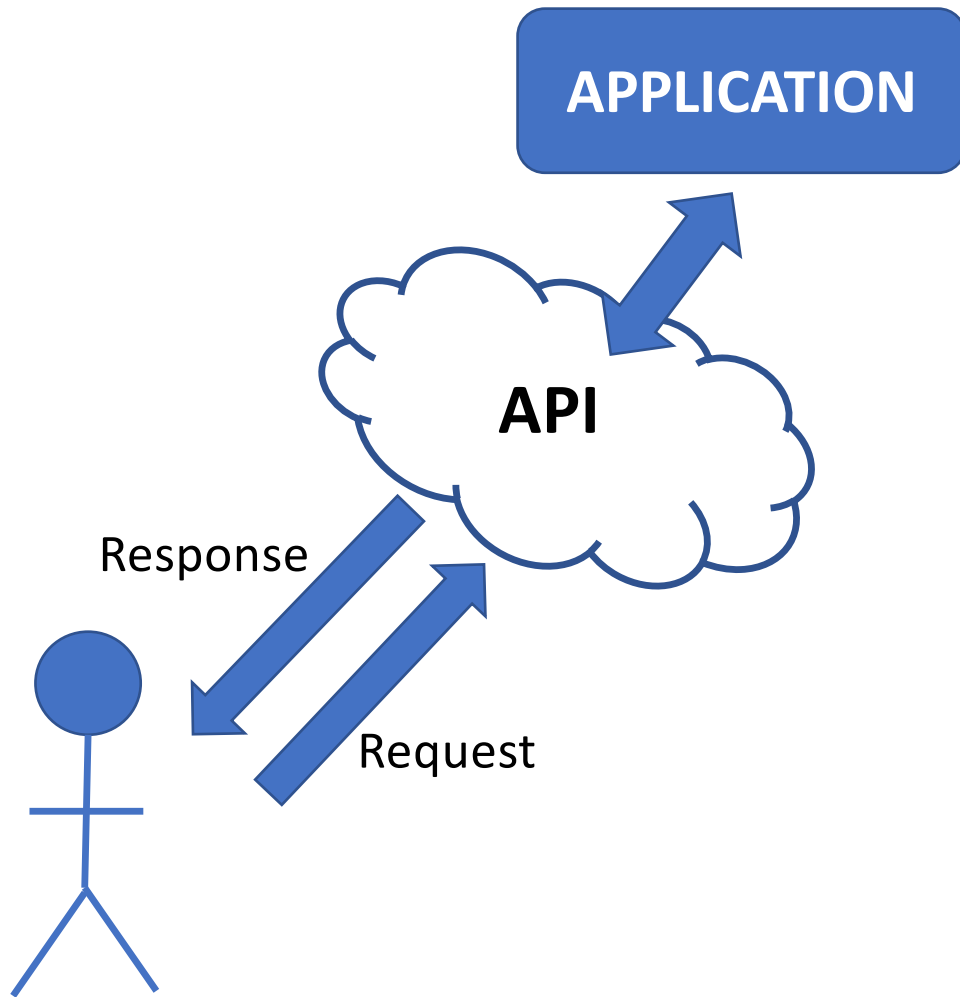
XML

- XML is Extensible Markup Language (www.w3c.org)
- In XML, structure and format are conveyed by markup which is embedded into the information

<markup>text</markup>

```
<section>  
  <title>Hospital Course</title>  
  <text> The patient was admitted and started on Lovenox and  
    nitroglycerin paste. The patient had serial cardiac  
    enzymes and was ruled out for myocardial infarction.  
    The patient underwent a dual isotope stress test.  
    There was no evidence of reversible ischemia on the  
    Cardiolute scan. The patient has been ambulated.  
  </text>  
</section>
```

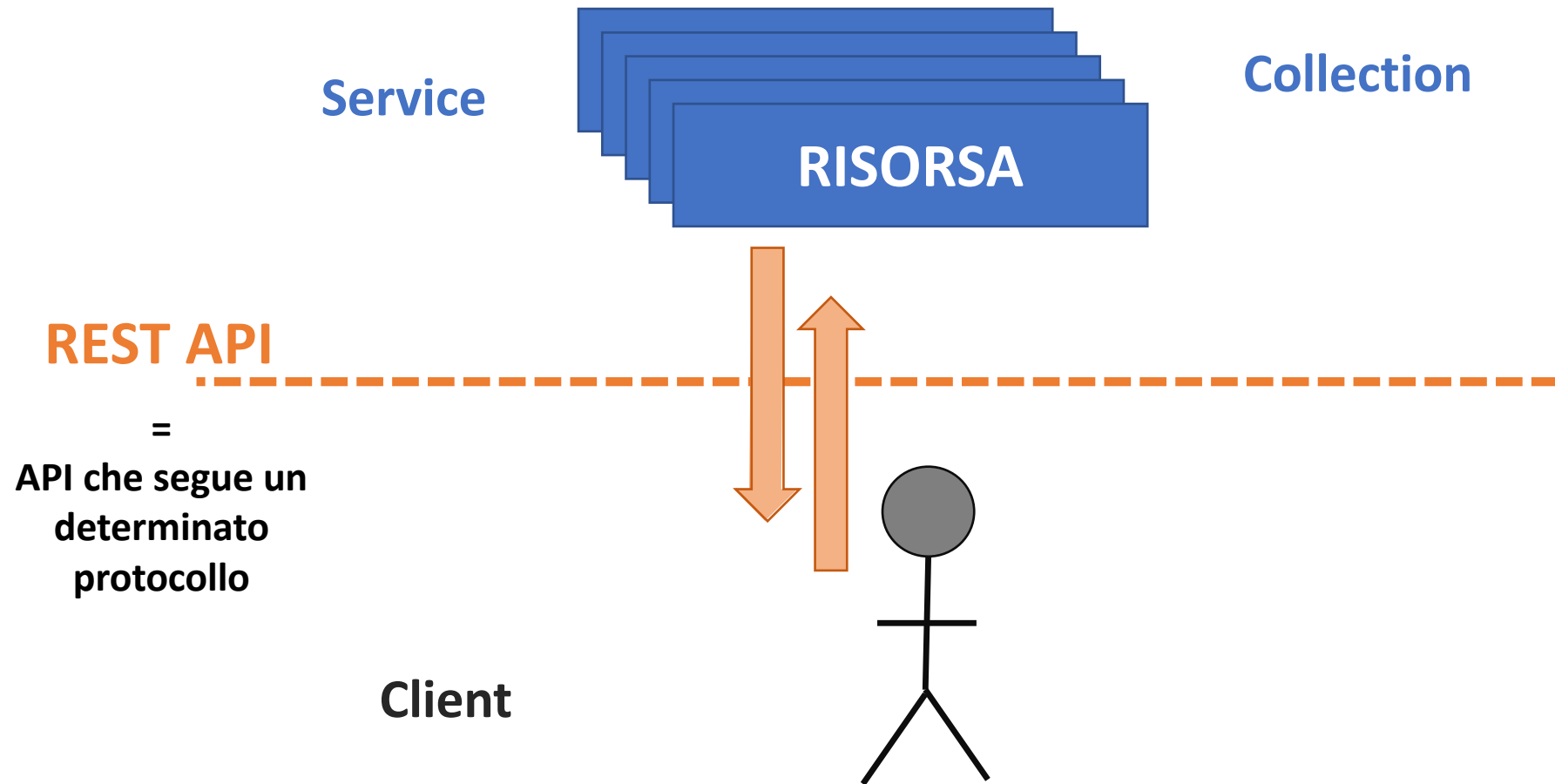
API - application programming interfaces



API =

- interface that can be used to program software that interacts with an existing application.
- a set of functions and procedures that allow you to access and build upon the data and functionality of an existing application.

REST API



REST API

- REST = REpresentational State Transfer
- It is an architectural style used to build Web services that are lightweight, maintainable, and scalable in nature.
- A service which is built on the REST architecture is called a RESTful service.
- The underlying protocol for REST is usually HTTP, which is the basic web protocol. However, other protocols (SMTP etc) can be used.
- REST makes resources available through an URI

REST KEY COMPONENTS

Resources – Element that contains the information.

Request Verbs - Description of what you want to do with the resource.

- The basic request is GET (= retrieve data)
- POST (=create a new element)
- PUT (= update an existing element)
- DELETE (= delete an element)

Request Headers – Additional instructions sent with the request (type of response required, authorization details)

Request Body - Data is sent with the request (usually in a POST call)

Response Body – This is the main body of the response (XML document, JSON)

Response Status codes –General codes which are returned along with the response from the web server. (200 = OK, 404 = NOT FOUND)

RESOURCES

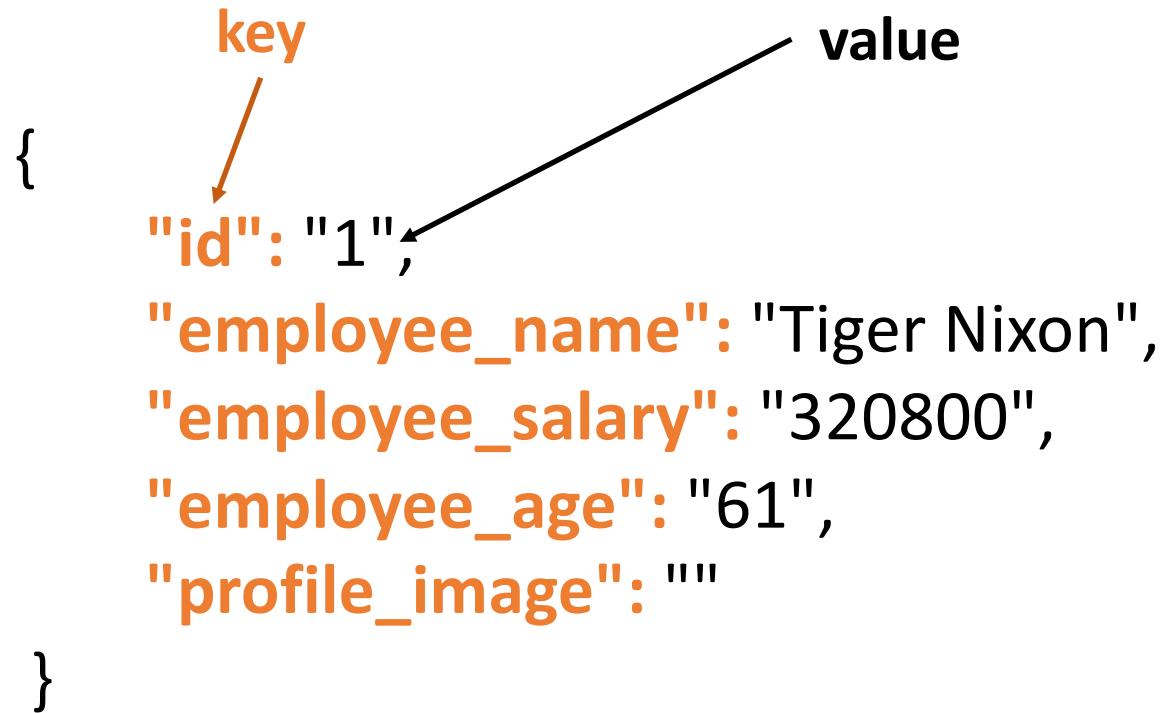
```
{
  "id": "1",
  "employee_name": "Tiger Nixon",
  "employee_salary": "320800",
  "employee_age": "61",
  "profile_image": ""
}, {
  "id": "2",
  "employee_name": "Garrett Winters",
  "employee_salary": "170750",
  "employee_age": "63",
  "profile_image": ""
}, {
  "id": "3",
  "employee_name": "Ashton Cox",
  "employee_salary": "86000",
  "employee_age": "66",
  "profile_image": ""
},
```

```
{
  "id": "4",
  "employee_name": "Cedric Kelly",
  "employee_salary": "433060",
  "employee_age": "22",
  "profile_image": ""
}, {
  "id": "5",
  "employee_name": "Airi Satou",
  "employee_salary": "162700",
  "employee_age": "33",
  "profile_image": ""
}, ...
```

<http://dummy.restapiexample.com>

JSON – JAVASCRIPT OBJECT NOTATION

Format to represent data exchanged in the Internet based on the concept of **key = value**



```
{  
  "id": "1",  
  "employee_name": "Tiger Nixon",  
  "employee_salary": "320800",  
  "employee_age": "61",  
  "profile_image": ""  
}
```

The diagram shows a JSON object with five key-value pairs. An orange arrow labeled "key" points to the string "id" in the first pair. A black arrow labeled "value" points to the string "1" in the same pair. The other pairs are "employee_name": "Tiger Nixon", "employee_salary": "320800", "employee_age": "61", and "profile_image": "".

REQUEST VERBS - GET

- The HTTP GET request method is used to get a resource from the server.
- The HTTP GET requests cannot have a message body but you still can send data to the server using the URL parameters.
- The GET requests should only receive data.
- The HTTP GET method is defined as idempotent, which means that multiple identical GET requests should have the same effect as a single request.

GET - EXAMPLE

- The request is simply an URL (you can copy and paste in your browser)
- You have to know the methods accepted by the service you are calling

`http://dummy.restapiexample.com/api/v1/employees`

REQUEST VERBS - POST

- The POST method allows to write a new resource
- The POST methods requires a body
- To post JSON to the server, you must set the appropriate content type for the request body.

Content-Type: application/json

- If your client is expecting a JSON string from the server, it should also send the Accept: application/json request header.

Accept: application/json

- The server informs the client that it has returned JSON using the Content-Type: application/json response header.

POST - EXAMPLE

- The request needs a body that has to be passed along with the request message

```
POST /api/v1/create HTTP/1.1
Host: dummy.restapiexample.com
Accept: application/json
Content-Type: application/json
Content-Length: 94
{
  "name": "test",
  "salary": "123",
  "age": "23",
  "id": 25
}
```