# Esercitazione di Fine Settimana – Week 4

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | Nome | Elena |
|  |  | Cognome | Zazzetti |
|  |  | Data | 16/04/2021 |

Leggete attentamente ogni domanda e argomentare quanto più possibile fornendo anche degli esempi.

1. Quali sono i componenti che caratterizzano un Endpoint?

L’Endpoint ha le tre informazioni note come ABC. A che sta per Address, dove il servizio è pubblicato, B che sta per Binding, come il servizio è pubblicato, con quale protocollo, la tipologia di collegamento e C che sta per Contract, cosa pubblica il servizio, cosa rende disponibile. Un servizio WCF si basa sugli EndPoint, che sono le porte attraverso le quali un client WCF si connette a un servizio WCF.

1. Cos’è un Web Service?

Un servizio web è un sistema software che serve per supportare l’interoperabilità tra elaboratori nella stessa rete o in un contesto distribuito. Per esempio supporta la comunicazione tra applicazioni, come tra chi detiene dei dati e chi li richiede, anche se sono in luoghi diversi. W3C, che sta per World Wide Web Consortium, descrive delle linee guida su come scrivere una pagina web, su come gestirla, in modo tale da avere un insieme di regole uguali per tutti. Il servizio web è un’interfaccia tramite cui due dispositivi possono comunicare tra loro. Per interfaccia si intende un canale tramite cui due dispositivi o due applicazioni possono comunicare. La prima sua caratteristica principale è che è quindi multipiattaforma. Per esempio chi detiene i dati e il client possono appartenere a piattaforme diverse ma per comunicare devono avere uno stesso protocollo di comunicazione per l’invio e la ricezione messaggi. La seconda caratteristica è quella della condivisione. Il servizio messo a disposizione dal server è messo a disposizione a più client.

1. Quali parti costituiscono una HTTP Response?

Il messaggio di risposta da parte del server contiene nella prima riga dell’header la versione http che utilizza il server e il codice dello stato della risposta (200 se è andato tutto bene, 400 se si sono verificati errori sul client, 500 se si sono verificati errori sul server). Sempre nell’header, dopo la prima riga con la versione http e lo stato, troviamo di che tipo sarà il risultato (es. json), la data che caratterizza quella risorsa, e informazioni aggiuntive come le caratteristiche sulle informazioni contenute nel body.

1. Descrivere l’utilizzo degli attributi [DataContract] e [OperationContract]

Il ServiceContract è un’interfaccia con i metodi che il servizio espone all’esterno. Il metodo che l’interfaccia dovrà contenere è caratterizzato da [OperationContract]. Poi si definisce una classe servizio che implementa l’interfaccia.

Il Data Contract descrive in modo astratto i dati da scambiare, in che modo vedrò all’esterno quell’oggetto utilizzando quel servizio. Posso definirlo esplicitamente con gli attributi DataContract e Datamember. Tutte le classi che rappresentano degli oggetti da mostrare all’esterno sono caratterizzati da DataContract e le singole proprietà che verranno esposte all’esterno sono caratterizzate da DataMember.

1. Come vengono utilizzati gli HTTP Methods in un servizio REST?

L’accesso ai dati avviene tramite un URL (https:// nome del server/ api / nome della risorsa) e gli HTTP Verbs, che sono GET per la lettura, POST per creare un’entità, PUT per aggiornarla o DELETE per eliminarla. Il GET può restituire una singola risorsa o un array di risorse, in entrambi i casi in formato json. Il GET per la singola risorsa può richiedere per esempio un ID che viene passato nell’url dopo il nome della risorsa. Lo stesso per il DELETE e il PUT. Per le modifiche o la creazione, quindi con PUT o POST, passerò nel body del messaggio la risorsa in formato json.

1. Come viene configurato un servizio realizzato con ASP.NET Core WebAPI?

Il file config è sostituito dal file Startup.cs. Il file Startup.cs è l’entry point di una web app, da dove parte l’applicazione. Può accettare una dependency injection nel costruttore e attraverso un ConfigurationBuilder è possibile gestire la configurazione. Poi contiene i metodi ConfigureServices e Configure.

Nel metodo ConfigureServices viene configurato ASP.NET, gestendo i servizi usati nell’app (swagger, dbcontext, controllers) e sfrutta la Dependency Injection. Configure gestisce come ASP.NET risponderà alle richieste. Con Use aggiunge le funzionalità, come i middleware per http, middleware per l’autorizzazione, per l’endpoint.

**Esercitazione Pratica**

* Realizzare un database per la Gestione degli Ordini e dei Clienti.
  + ***Cliente***
    - *ID* (int, PK), CodiceCliente (*string*), Nome (*string*), Cognome (*string*)
  + ***Ordine***
    - *ID* (int, PK), *DataOrdine* (date), *CodiceOrdine* (string), *CodiceProdotto* (string), *Importo* (decimal)
  + La realizzazione dello strato di accesso al dato deve essere realizzata con EF Code-first.
* Realizzare un servizio WCF per la gestione di una Anagrafica Clienti (CRUD)
* Realizzare un servizio REST per la gestione di una Anagrafica Ordini (CRUD)