

26 maggio 2021

1. Un *servizio web* è un'interfaccia attraverso la quale due dispositivi possono comunicare tra loro e scambiarsi delle informazioni.
2. Una *http response* è formata da diversi "pezzi di informazione". Innanzitutto, si può distinguere da una parte di *headers*, ovvero un'intestazione contenente i metadati della risposta (es. la versione del protocollo http, il codice dello stato, la lunghezza della risposta contenuta nel body...). La seconda parte di una *response* è, invece, costituita da un *body*, di solito una struttura chiave-valore contenente la risorsa richiesta dal client, oppure, nel caso in cui la comunicazione non sia andata a buon fine, può contenere un messaggio di errore. Il body può, eventualmente, contenere informazioni aggiuntive.
3. L'attributo *Data contract* consente di definire l'insieme delle risorse/informazioni che un server può fornire a un client e che costituiscono, insieme alle funzionalità che il server offre (identificate per mezzo dell'attributo *Operation contract*), l'*interfaccia pubblica* del server stesso.
4. Gli *http methods* sono una di quelle poche informazioni di cui ha bisogno un servizio Rest per funzionare. Un servizio Rest è una delle tante modalità attraverso cui è possibile realizzare dei servizi web (ovvero quei servizi che usano il protocollo di rete http). A differenza del WCF, un servizio Rest può sfruttare le informazioni contenute nel protocollo http stesso, tra cui, appunto, gli *http methods* (o *verbs*) che performano delle azioni. Ad esempio, GET serve per recuperare dei dati, POST per creare nuove risorse, PUT per aggiornarle e DELETE per cancellarle.
5. Innanzitutto, ASP.NET è una libreria dedicata allo sviluppo web. Nel caso in cui si decida di realizzare una Web API usando ASP.NET Core, si nota che la configurazione avviene in parti distinte del progetto stesso. Innanzitutto nei file *appsettings.json* e *appsettings.Development.cs*. Quest'ultimo, ad esempio, definisce le impostazioni del programma nell'eventualità in cui ci si trovi in un ambiente di development. Un terzo file di configurazione è lo *Startup.cs*: una classe all'interno della quale sono definiti due metodi. Il primo (*Configureservices*) che definisce i servizi che si andranno a usare nell'applicazione; il secondo (*Configure*) definisce, invece, una pipeline (ovvero una sequenza operazioni che devono essere eseguite per manipolare la richiesta). Andando nello specifico, in quest'ultimo metodo si definiscono gli step da compiere nel caso in cui ci si trovi in fase di sviluppo, ma anche le operazioni da compiere in qualsiasi tipo di ambiente. Tra queste ultime vi sono: la trasformazione di una richiesta http in https, la chiamata al modulo per soddisfare la richiesta, l'autenticazione e l'accesso agli endpoint che soddisfano la richiesta stessa.