Progetto Prova Finale

Nome: Stefano Di Nucci   
Codice Persona: 10763612   
Repository: <https://github.com/Stepness/polimi-project>

**Requisiti**

Il progetto consiste in un servizio web che cerca di risolvere il problema di condivisione di file all’interno di una azienda.  
Quindi invece di dover setuppare dischi all’interno della rete aziendale tramite NFS/SMB si può ricorrere a questo servizio.

È costituito da una applicazione web che permette di caricare file da una interfaccia web e poterli scaricare da qualunque utente che abbia un account.

Il servizio funziona come un file storage, in cui ogni utente, dipendentemente dal proprio livello di autorizzazione, può leggere, caricare file o modificarli.

Agli utenti può essere assegnato uno tra I seguenti ruoli:

**Admin**:

* Gestisce il ruolo degli altri utenti. Il servizio richiede solo un primo controllo del ruolo, di conseguenza l’unica gestione dei ruoli attualmente disponibile è solo quella di promozione degli utenti da **Reader** a **Writer**
* In aggiunta ha anche i permessi che ha un **Writer**

**Writer**:

* Upload dei file
* Rinominare file
* Cancellare file
* Scaricare file

**Reader**:

* Scaricare file

Alla prima registrazione viene assegnato di default il ruolo di Reader.  
Per essere promosso a Writer cè bisogno di un Admin che se ne occupi.  
Un Admin può essere solo settato da database.

**Tecnologie**

**Frontend**

Il client è costituito da HTML, CSS e Javascript, con l’ausilio di Jquery per le richieste HTTP al backend.

**Backend**

Il servizio è implementato in C# attraverso .NET8.

Il servizio gestisce l’autenticazione e I livelli di autorizzazione dell’API.

L’autenticazione è una semplice funzione di hash che compara la password passata dal client a quella che si trova nel database.

L’autorizzazione si basa sul protocollo OAuth2, questo significa che a seguito dell’autenticazione, il server genera un token in formato json (chiamato Json Web Token o anche JWT), che viene rimandato al client.

Il client in ogni Request dovrà inviare al server questo token per potersi autorizzare.

Questo token contiene il nome dell’utente e il suo ruolo, e ha una scadenza di 1 giorno.

**Dati**

Come database viene utilizzato un database NoSQL, in particolare Cosmosdb, un servizio PaaS hostato sul cloud Azure.

Il servizio all’avvio si aspetta una stringa di connessione a questo servizio, se non è presente il servizio si avvia comunque e salverà i dati InMemory.

**Test**

Come framework di test è stato usato xUnit.

In aggiunta sono stati utilizzati:

Autofixture: libreria per generare oggetti con valori randomici

Fluent Assertion: liberia che migliora la lettura e la scrittura delle assertion fatte dai test

Nsubstitute: libreria per creare i mock nei test

**Class Diagram**

**A diagram of a software company

Description automatically generated**

**UseCase Diagram**

**A diagram of a software system

Description automatically generated**