



RELAZIONE PROGETTO DI TELEMATICA

Andrea Bertogalli - Nicolò Tombini

2021-2022

DNSOLVER

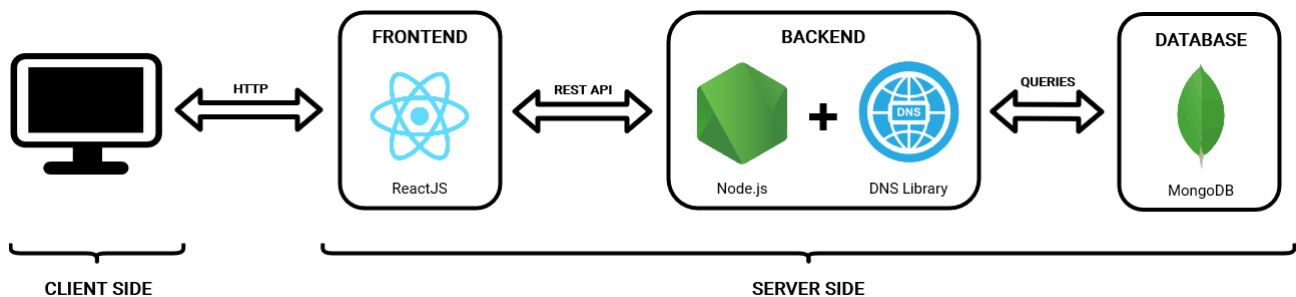
SOMMARIO

Introduzione	3
struttura del progetto.....	3
Frontend	3
Backend.....	4
Database.....	4
Steps di sviluppo del progetto	4
Guida utente.....	5
Utilizzo del programma	5
Registrazione.....	5
Login	6
Dashboard	6
Conclusioni	8

INTRODUZIONE

Per quanto riguarda l'attività di progetto per il corso di Telematica si è deciso di sviluppare un Resolver DNS utilizzabile mediante delle REST API quindi mediante HTTP, inoltre per rendere più user-friendly il progetto si è voluta sviluppare una GUI accessibile mediante il browser.

STRUTTURA DEL PROGETTO



Lo schema riportato mostra in maniera generale la struttura del progetto, tutto il progetto è stato sviluppato lato server, il client (browser) si limita ad accedere alla web-application e interagire con il backend attraverso di essa. Lato server distinguiamo tre macro-blocchi:

- Frontend
- Backend
- Database

È bene specificare che frontend e backend sono due processi server, due server NodeJS, distinti.

FRONTEND

Per frontend si intende tutto quello che è visualizzabile sottoforma di GUI da parte dell'utente, in particolare per lo sviluppo del frontend si è preferito utilizzare [ReactJS](https://reactjs.org/).

React è una libreria open-source, front-end, JavaScript per la creazione di interfacce utente. È mantenuto da Meta e da una comunità di singoli sviluppatori e aziende.

React permette appunto di sviluppare interfacce grafiche che siano molto efficienti, questo deriva dal fatto che React utilizza un sofisticato meccanismo di sincronizzazione tra un DOM virtuale e quello reale e permette quindi di aggiornare la pagina senza doverla ricaricare, ma solamente aggiornando i vari pezzi. Il frontend, inoltre, girando su NodeJS fa sì che si possano integrare una grande varietà di moduli, i principali moduli che sono stati utilizzati, oltre a React ovviamente, sono:

- **Axios:** modulo estremamente utile che permette di effettuare in maniera estremamente efficace qualsiasi richiesta http e largamente utilizzato con React.
- **Material-ui:** questa libreria sviluppata da Google nel 2014 è attualmente la libreria appunto usata nei Frontend Google, mette a disposizione una vasta gamma di componenti grafici integrabili con React.

- **React-router-dom:** questo modulo è stato di estrema importanza per quanto riguarda la gestione degli accessi alle pagine e ai componenti, integrandosi con il metodo di autenticazione sviluppato.

Sono stati usati anche altri moduli di minor importanza quali: js-sha256 utile per generare hash sha256, vari set di icone ecc...



BACKEND

Il backend ha varie funzionalità, tra queste distinguiamo ovviamente quella di Resolver DNS, quella di server HTTP che mette a disposizione delle API REST per l'utilizzo del Resolver DNS in maniera estremamente semplice e controllata, infine quella di accesso e gestione della base di dati No-SQL.

Per lo sviluppo sono state utilizzate le seguenti tecnologie:

- **Express:** Express.js è un framework per applicazioni web per Node.js serve per creare web applications e API ed ormai definito il server framework standard de facto per Node.js.
- **Nodemon:** Un tool molto utile durante lo sviluppo del server in quanto permette di mantenere aggiornato il server senza doverlo riavviare, funziona in maniera simile al rendering di react, ma per applicazioni node.
- **Mongoose:** Dato l'utilizzo di MongoDB come DBMS è stato necessario installare su node il modulo che ne permettesse l'interazione.
- **DNS:** Questo modulo di node ci ha permesso di effettuare risoluzioni dns, quindi di far sì che il backend agisse da resolver dns.



DATABASE

Per il database è stato utilizzato MongoDB che è una delle principali tecnologie per basi di dati NoSQL, MongoDB è basato su documenti JSON organizzati in collezioni. Sulla base di dati vengono salvate le informazioni dell'utente username, password (sha-256), nome, cognome e inoltre vengono salvate la cronologia delle ricerche e le preferenze di configurazione per quanto riguarda i DNS.

La scelta di utilizzare MongoDB è stata dettata dal fatto che esso si integra perfettamente con Javascript e quindi con NodeJS e inoltre è estremamente prestazionale.

STEPS DI SVILUPPO DEL PROGETTO

Per lo sviluppo del progetto si è partiti dallo sviluppo del backend che utilizza la libreria DNS e successivamente sono state scritte le API che forniscono i dati mediante JSON ad un eventuale client. Come secondo step si è progettata la base di dati, e infine l'interfaccia grafica con ReactJS. Come ultimo passaggio si è migliorata la stabilità del progetto e si sono eseguiti vari test.

GUIDA UTENTE

L'installazione del sistema è abbastanza semplice, quello che occorre è NodeJS installato sul sistema, dopo di che basterà recarsi nella cartella backend e lanciare il comando ***npm install*** e, in maniera analoga, nella cartella frontend. per lanciare il frontend servirà il comando ***npm start*** mentre per il backend ***node index.js***. Per configurare la base di dati basta importare su MongoDB il file *user.json* per caricare la base di dati.

Per semplicità di utilizzo forniamo una demo pubblica al seguente indirizzo:

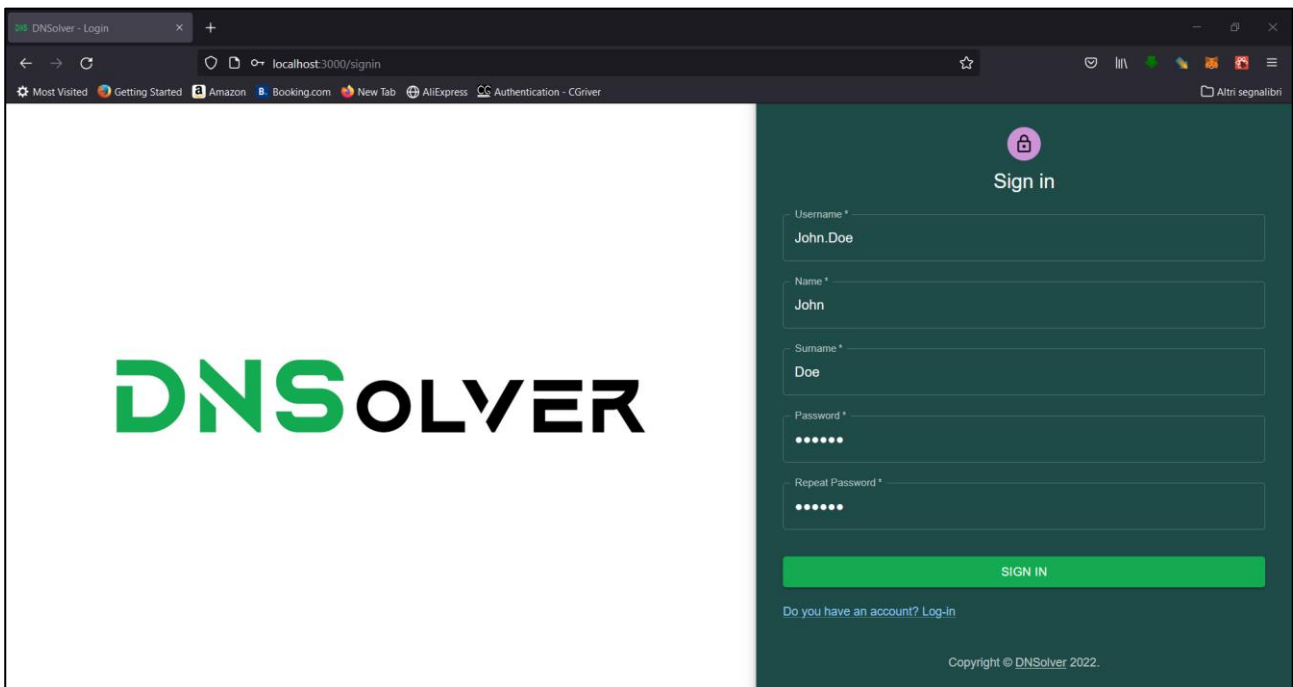
<https://dnsolver.herokuapp.com/>

UTILIZZO DEL PROGRAMMA

L'applicazione è costituita da tre schermate:

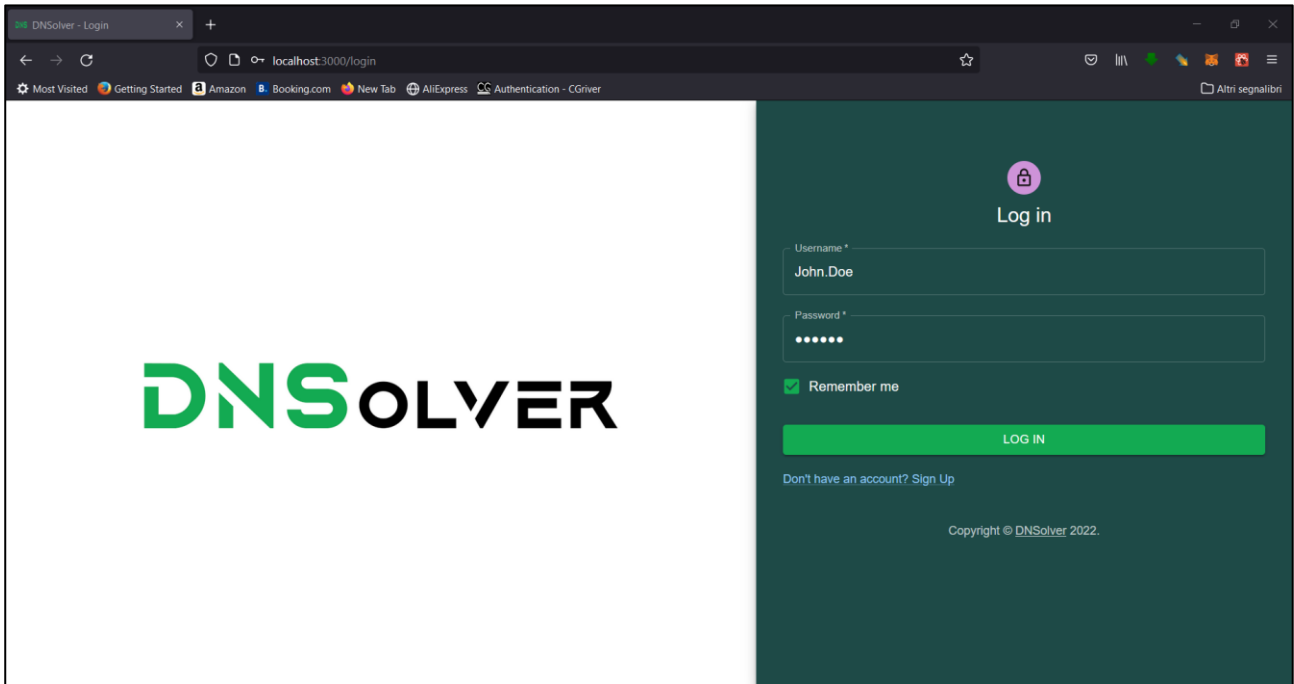
- Login
- Registrazione
- Dashboard

REGISTRAZIONE



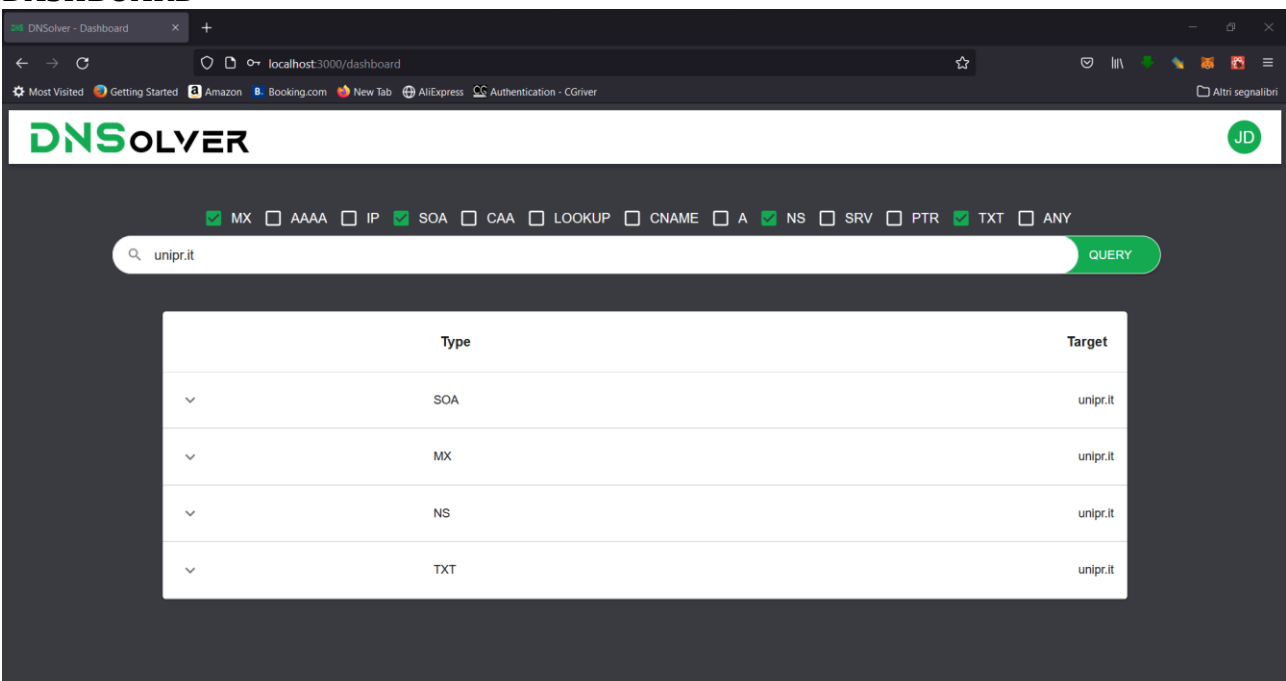
Questa pagina permette ad utenti non registrati di registrarsi alla piattaforma, in questa sezione è presente un form che permette di inserire le credenziali di un nuovo utente solo se non è già presente un utente con lo stesso username, l'invio verso il server è reso possibile tramite un apposito bottone di Signup. Una volta confermata la registrazione dal server l'utente procederà presso la pagina di login altrimenti verrà mostrato un messaggio di errore.

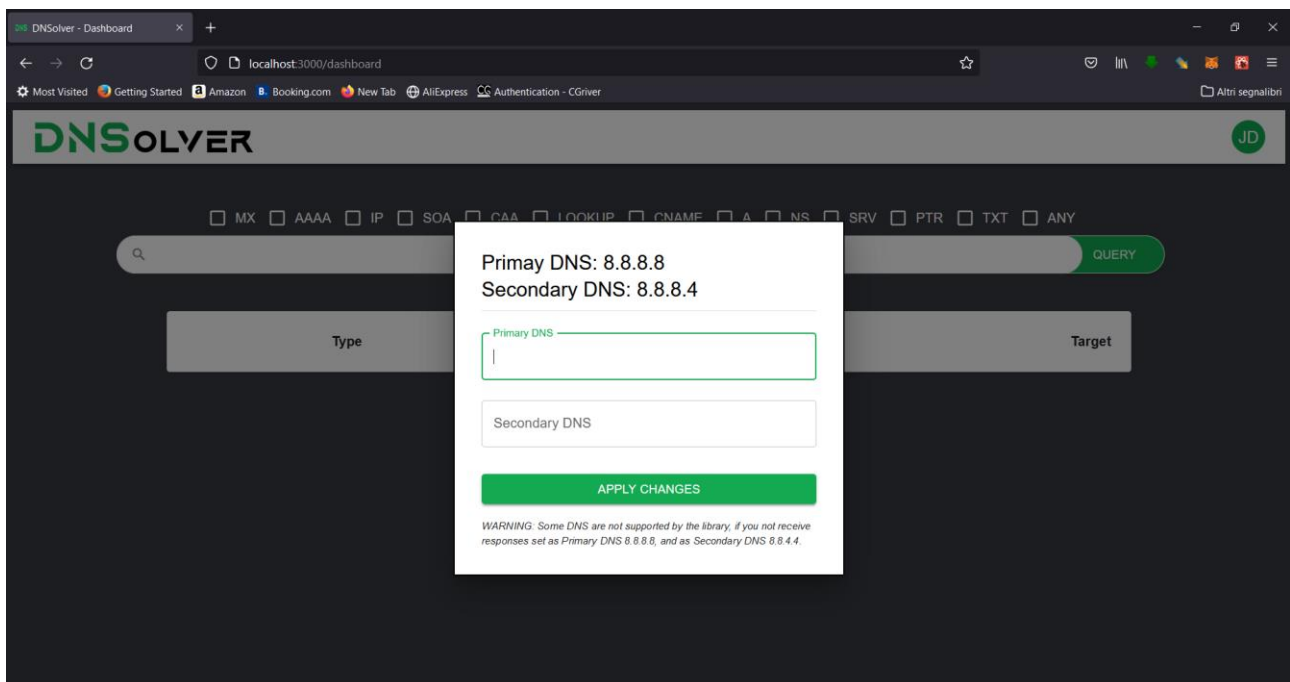
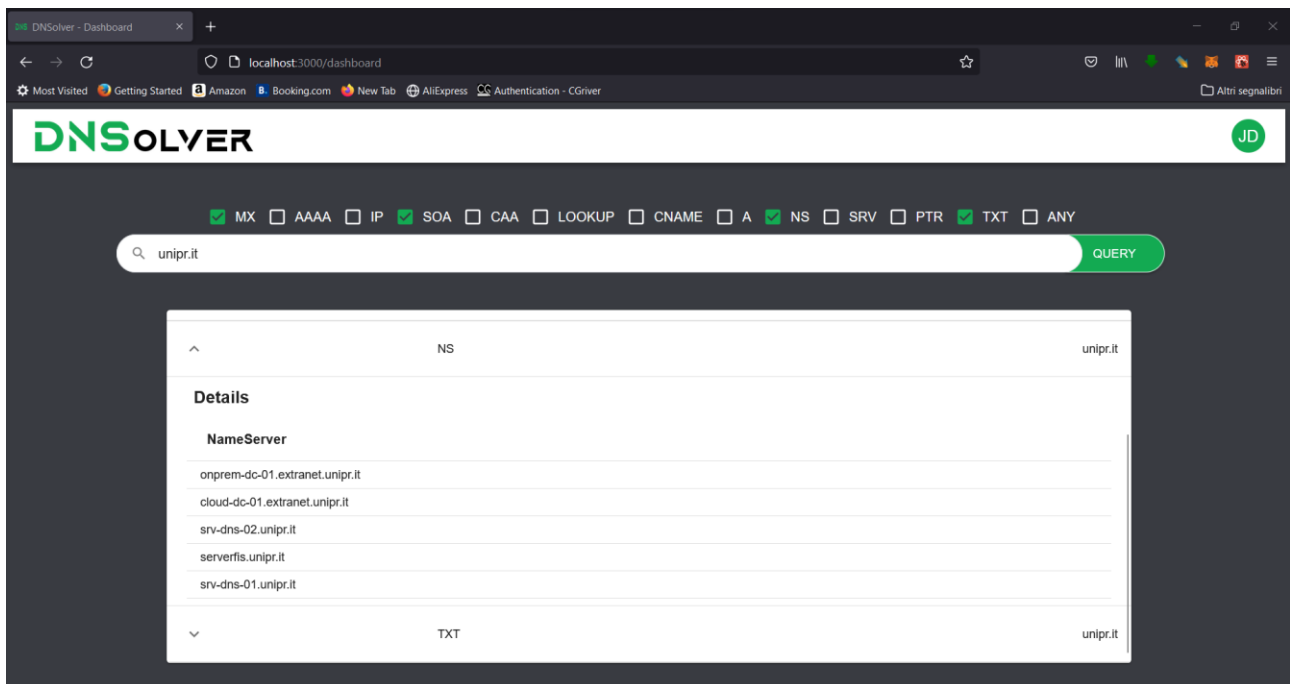
LOGIN



Nella parte dedicata al login un utente iscritto alla piattaforma può fornire, attraverso un apposito form le sue credenziali che sono composte da username e password, l'apposito bottone di login permetterà di inviare una richiesta di autenticazione al server che confermerà o meno la possibilità di accesso al sistema. In caso le credenziali siano esatte l'utente verrà reindirizzato presso la dashboard mentre se sono errate verrà comunicato l'esito negativo tramite un apposito messaggio. Il form di login prevede inoltre la possibilità di rimanere connessi (checkbox remember me), in questo caso, fino a che l'utente non sceglierà il contrario effettuando il logout, la piattaforma autenticcherà il dispositivo e quindi l'utente in automatico. Dalla pagina di login è inoltre possibile spostarsi alla pagina di registrazione tramite l'apposito link.

DASHBOARD





Nella dashboard possiamo accedere al menu utente in alto a destra, e possiamo effettuare il logout, cancellare la cronologia e settare i nostri DNS preferiti.

Centralmente troviamo una barra dove è possibile specificare l'host o il dominio al quale si vuole effettuare la query DNS e tramite le checkbox si può selezionare una o più tipologie di query effettuabili, premendo il bottone query si eseguono le query.

Mentre si scrive nella barra di ricerca è possibile selezionare una delle opzioni proposte dalla cronologia.

I risultati appariranno nella tabella sotto la barra, se i risultati sono multipli è possibile visualizzarne i dettagli espandendo la tabella.

Le tipologie di query effettuabili sono:

- **MX:** Risolve record MX
- **AAAA:** Da hostname a IPv6
- **IP:** Da ip ad hostname
- **SOA:** Risolve i record SOA
- **CAA:** Risolve i record CAA
- **LOOKUP:** Fa il lookup su un servizio ad uno specifico indirizzo
- **CNAME:** Risolve i record CNAME
- **A:** Da hostname IPv4
- **NS:** Risolve i record NS
- **SRV:** Risolve i record SRV
- **PTR:** Risolve i record PTR
- **TXT:** Ritorna i record TXT
- **ANY:** Ritorna tutti i record DNS per la ricerca effettuata

CONCLUSIONI

Questo progetto se pur semplice fornisce varie possibilità utili per analizzare record DNS, potrebbe essere utile in caso si voglia testare il DNS server della propria organizzazione.