

Descrizione Del Progetto

- Principali tecnologie utilizzate
 - Database
 - MySQL
 - BackEnd
 - Java
 - Spring Boot - Spring Security - JPA
 - FrontEnd
 - Javascript
 - JQuery
 - HTML
 - CSS
- Organizzazione Del Progetto
 - controller
 - contiene lo UserController che espone le routes principali del progetto e gestisce un primo livello di business logic. Esiste un secondo controller specializzato per la gestione delle chiamate che coinvolgono un utente admin
 - DTO
 - Ho definito un package contenente i Data Transfer Object per ridurre i dati da trasportare al minimo necessario e per non esporre informazioni potenzialmente sensibili
 - Service
 - Gestiscono la business logic principale dell'applicativo
 - Nel tentativo di mantenere la coesione e di contenere la dimensioni delle classi esiste un service per gestire la business logic delle entità principali
 - Ogni Service implementa un interfaccia
 - La creazione dei Service è disaccoppiata per mezzo di DependencyInjection
 - DAO / Repository
 - Gestiscono la logica di persistenza del progetto
 - Ogni DAO è associato ad una specifica entità dello strato di modello
 - Ogni DAO estende l'interfaccia JpaRepository
 - La creazione dei DAO è disaccoppiata per mezzo di DependencyInjection
 - Model
 - Mappano le tabelle del database del progetto definendone alcune proprietà principali
 - Security

- Sezione specializzata per la sicurezza dell'applicativo, accorpa le configurazioni principali per la gestione dell'autenticazione, dei ruoli, della navigabilità del sistema e dell' encrypt at rest.
 - Contiene inoltre gli artefatti per alcune funzionalità che avevo iniziato ad implementare ma che per ragioni di tempo ho dovuto abbandonare (generazione dei token, aggiunta di una funzionalità "ricordami")
- Templates e Static
 - pagine html principali e una pagina custom di errore
 - fogli di stile
 - gestione lato front-end di alcune funzionalità, fra cui la definizione di una regex per il controllo del formato email
- application.properties
 - Definisce alcune proprietà globali e la creazione del database
- Tabelle
 - è stata definita una tabella per ogni entità principale considerata (User, Login, Colonna, Tile) più una tabella di configurazione in cui si prevedeva di salvare solo alcune righe (poi attualmente solo una) con valenza di informazione globale a cui accedere. In particolare si è deciso di salvare l'informazione relativa alla presenza o meno di un utente amministratore senza dover eseguire una scansione su tutta la tabella utenti. Questo per creare una politica di gestione automatica dei ruoli degli utenti.
- Note
 - Mi scuso in anticipo con chi testerà il progetto. Iniziativa che mi ha entusiasmato ma che si è poi mal conciliata con l'effettivo tempo a disposizione. Sfortunatamente, come studente lavoratore non ho avuto molto tempo per gestire opportunamente tutti gli aspetti del progetto. In particolare lo strato di front-end e gestione di HTML e CSS è praticamente solo abbozzata, sia in senso di user experience sia in termini di organizzazione e ottimizzazione del codice (come ad esempio astruendo la logica di costruzione di molti elementi o studiando un framework apposito, cosa che sono riuscito a fare invece con SpringBoot e Spring Security). Inoltre come indicato nel paragrafo sulla sicurezza non è stata completata l'implementazione di alcune funzionalità che intendevo aggiungere e di cui sono presenti degli evidenti artefatti.

Crispino Samuele 856339