**Progettino Spring**

**Entities**

CatalogEntity:

La tabella nel db si chiama “catalog”.

Rappresenta un oggetto del catalogo, ha un nome una descrizione e un immagine (rappresentata da un percorso alla risorsa statica). La chiave primaria è un intero auto incrementale.

Role:

La tabella nel db si chiama “role”.

Rappresenta il ruolo di un utente, (“ADMIN” o “USER”) ha un nome. La chiave primaria è un intero auto incrementale.

User:

La tabella nel db si chiama “user”

Un utente, ha username, password, una booleana che rappresenta se l’utente è bannato o no, uno o più Role e un UserData che rappresenta informazioni aggiuntive sull’utente. L’UserData è collegata con una relazione OneToOne unidirezionale e viene aggiornata o cancellata quando vengono effettuate le stesse operazioni sull’utente. I ruoli sono collegati con una relazione ManyToMany attraverso la tabella “users\_roles”.

UserData:

La tabella nel db si chiama “user\_data”, contiene nome, cognome e email.

CrmUser:

É la classe usata per inserire i dati dell’utente nella pagina della registrazione. Non viene usato direttamente un User nel modello per avere un campo per la machingPassword.

UserService:

L’UserService prende in ingresso un CrmUser nelle operazioni save e update, crea un utente con username e dati corrispondenti o lo carica da db e infine salva su db. Durante la creazione viene aggiunto il ruolo di utente (ROLE\_USER), la password viene criptata con un BCryptPasswordEncoder. Le operazioni su db sono delegate alle Dao. Contiene anche un operazione per aggiungere o rimuovere il ruolo di admin.

CatalogEntityService:

Service semplice con operazioni findAll, findById, save e deleteById.

CatalogEntityRepository, RoleRepository e UserRepository:

Estendono JpaRepository e usano metodi query, quindi non richiedono un’ implementazione concreta.

Controller:

I controller sono implementati come semplici @Controller. L’account controller e il Registration controller creano un CrmUser e lo aggiungono al modello per poter inserire i dati di login.

MyAuthenticationSuccessHandler:

onAuthenticationSuccess viene chiamata quando l’autenticazione ha successo, assegna l’user alla session.