# **Creating A UserDetailsService**

UserDetailsService is a core interface in Spring Security that loads user-specific data for authentication. When we want to authenticate users against a database instead of hardcoded values, we need to create our own implementation of this interface.

Create a class called MyUserDetailsService in the service layer that implements the UserDetailsService interface. This class will be responsible for fetching user information from our database when a user attempts to log in.

## **Example:**

```
package com.telusko.springsecdemo.service;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.security.core.userdetails.UserDetails;
import org.springframework.security.core.userdetails.UserDetailsService;
import org.springframework.security.core.userdetails.UsernameNotFoundException;
import org.springframework.stereotype.Service;

@Service
public class MyUserDetailsService implements UserDetailsService {
    @Autowired
    private UserRepo repo;

    @Override
    public UserDetails loadUserByUsername(String username) throws UsernameNotFoundException {
        return null; // Will be implemented to fetch user from database
    }
}
```

#### > The @Service Annotation

Add the @Service annotation to class so Spring can detect it as a bean during component scanning. This allows it to be automatically injected into our AuthenticationProvider in the SecurityConfig class.

### > UserRepo Dependency

To access database, autowire a UserRepo interface. This repository will handle all database operations needed to fetch user information.

### ➤ The loadUserByUsername Method

The most important part of this service is the loadUserByUsername() method:

- It takes a username as input
- It's called by Spring Security during authentication
- It should query the database for the user with the matching username
- It must return a UserDetails object containing the user's credentials and authorities
- If no user is found, it should throw a UsernameNotFoundException