

DTO

User

+username: String
+password: String
+authorities: List<String>

BlogPost

+postId: int
+postDate: LocalDate
+expDate: LocalDate

DAO

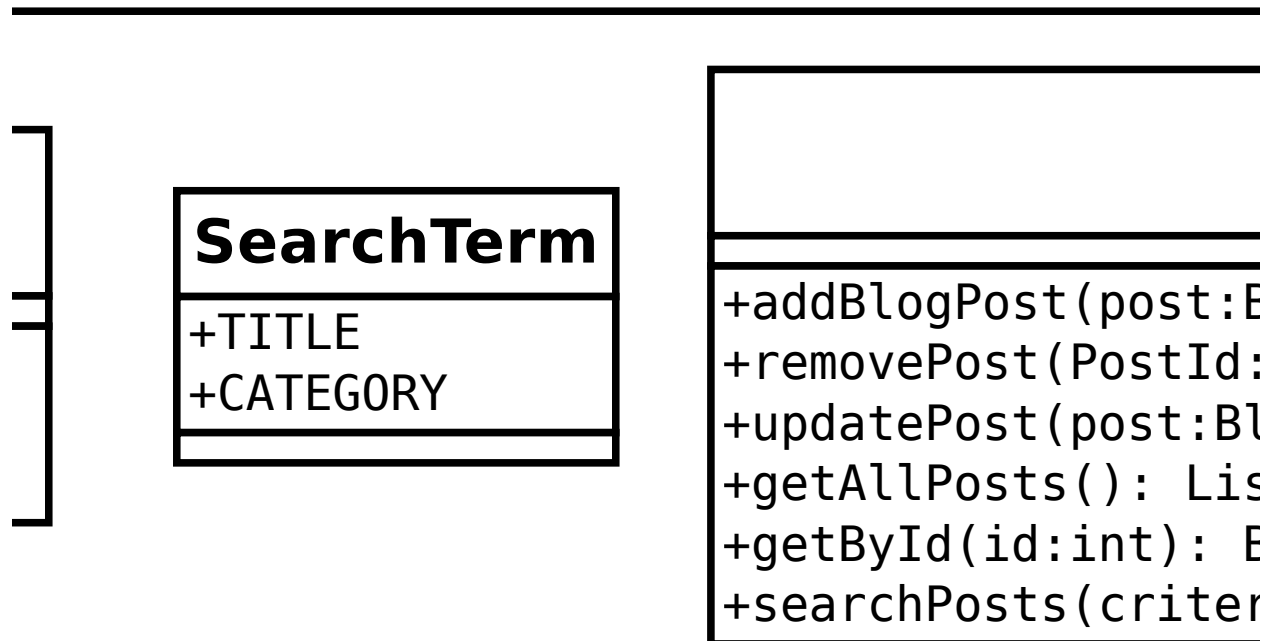
<<interface>>

UserDao

```
+addUser(newUser:User): User  
+deleteUser(Username:String): void  
+listAllUsers(): List<User>
```

```
+SQL_INSERT_USER: String = "insert  
+SQL_INSERT_AUTHORITY : String = "  
+SQL_DELETE_USER      : private static  
+SQL_DELETE_AUTHORITIES : String  
+SQL_GET_ALL_USERS : String = "sel  
+SQL_GET_AUTHORITIES : String  
+jdbcTemplate: JdbcTemplate
```

```
+addUser(newUser:User): User  
+ getAllUsers(): List<User>  
+deleteUser(userName:String): void  
+UserMapper(rs:ResultSet,i:int): Us  
+AuthorityMapper(rs:ResultSet,i:int
```



UserDaoDbImpl

```

    insert into users (username, password, enabled) values (?, ?, ?)
    insert into authorities (username, authority) values (?, ?)
    private final String deleteUsers = "delete from users where username = ?"
    private final String deleteAuthorities = "delete from authorities where username = ?"
    private final String selectUsers = "select username, password from users"
    private final String selectAuthorities = "select authority from authorities where username = ?"
  
```

```

    private User user
    private String password
  
```

<<Interface>>

BlogDao

BlogPost): BlogPost

int): void

BlogPost): void

List<BlogPost>

BlogPost

via:Map<SearchTerm, String>): List<BlogPost>

values (?, ?, 1)"
y) values (?, ?)
username = ?"
= ?"
username = ?"

BlogDaoDbImpl

```
+SQL_INSERT_POST: String
+SQL_INSERT_TAGS: String
+SQL_DELETE_POST: String
+SQL_UPDATE_POST: String
+SQL_SELECT_POST: String
+SQL_SELECT_TAG: String
+SQL_SELECT_ALL_POSTS: String
+SQL_SEARCH_POSTS: String
+
+setJdbcTemplate(): void
+addBlogPost(post:post): post
+removeBlogPost(id:int): void
+updateBlogPost(post:post): void
+getAllBlogPosts(): List<post>
+getBlogPostById(id:int): post
+searchBlogPosts(List<post>): Map<Se
    String>
+PostMapper()
+TagsMapper()
```

archTerm,

Controller

```
+dao: UserDao  
+encoder: PasswordEncoder  
+displayUserList()  
+displayUserForm()  
+addUser(req:HttpServletRequest)  
+deleteUser(model:Model)  
    ("user/{id}")
```

HomeController

```
+dao: DatabaseDao  
+displayHomePage()  
+createPost(): BlogPost  
+deletePost(id:post)  
+putPost(id:post, content: String)  
+getPost(id:post)  
+getAllPosts(): List<BlogPost>
```

UserController

rdEncoder

(model:Map<String, Object>): String

(model:Map<String, Object>): String

oServletRequest): String

l:Map<String, Object>,userName:String,
ername"):@RequestParam): String

ntroller

o

(): String

logPost

ostId): void

Id): void

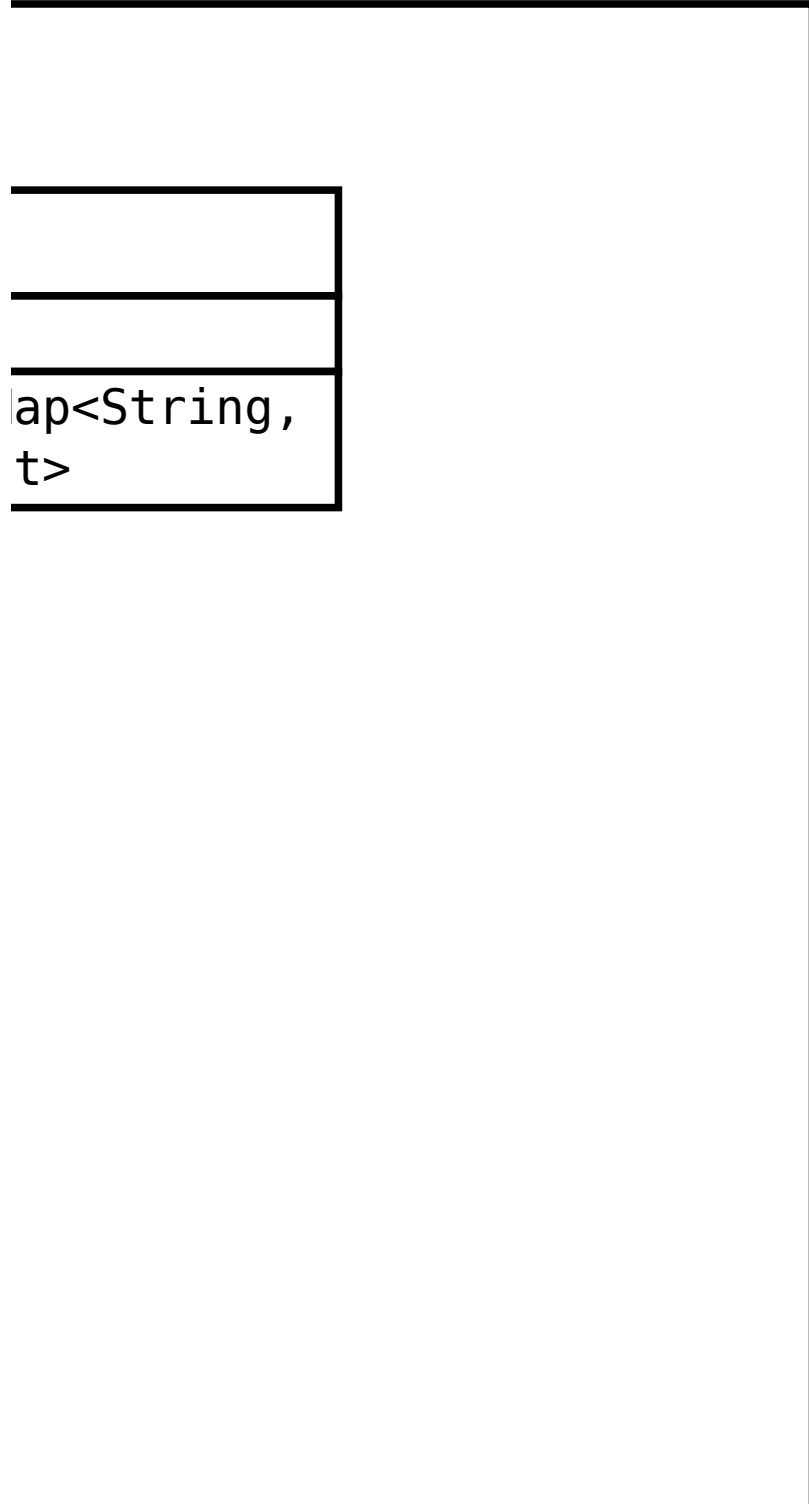
Id): BlogPost

List<Post>

SearchController

+dao: DatabaseDao

+searchPosts(searchMap:@RequestBody M
String>): List<BlogPos



ap<String,
t>





