

HUST

ĐẠI HỌC BÁCH KHOA HÀ NỘI
HANOI UNIVERSITY OF SCIENCE AND TECHNOLOGY

ONE LOVE. ONE FUTURE.



ĐẠI HỌC
BÁCH KHOA HÀ NỘI
HANOI UNIVERSITY
OF SCIENCE AND TECHNOLOGY

BÁO CÁO BÀI TẬP LỚN

Học phần: Lập trình hướng đối tượng

Mã môn học: IT3103

Giảng viên: Lê Đức Hậu

ONE LOVE. ONE FUTURE.

Giới thiệu thành viên nhóm 1

1. Nguyễn Quỳnh Anh	20225785
2. Vũ Minh Hiếu	20225841
3. Hồ Tuấn Huy	20225856
4. Nguyễn Bùi Tuấn Linh	20225732
5. Nguyễn Bùi Việt Linh	20225733
6. Nguyễn Khánh Toàn	20225936
7. Phạm Quốc Cường	20225604
8. Phạm Duy Hưng	20225850
9. Hà Thành Long	20225875
10. Hà Ngọc Huy	20225855

ĐỀ TÀI

Quản lý Bất động sản



Quản lý Dự án

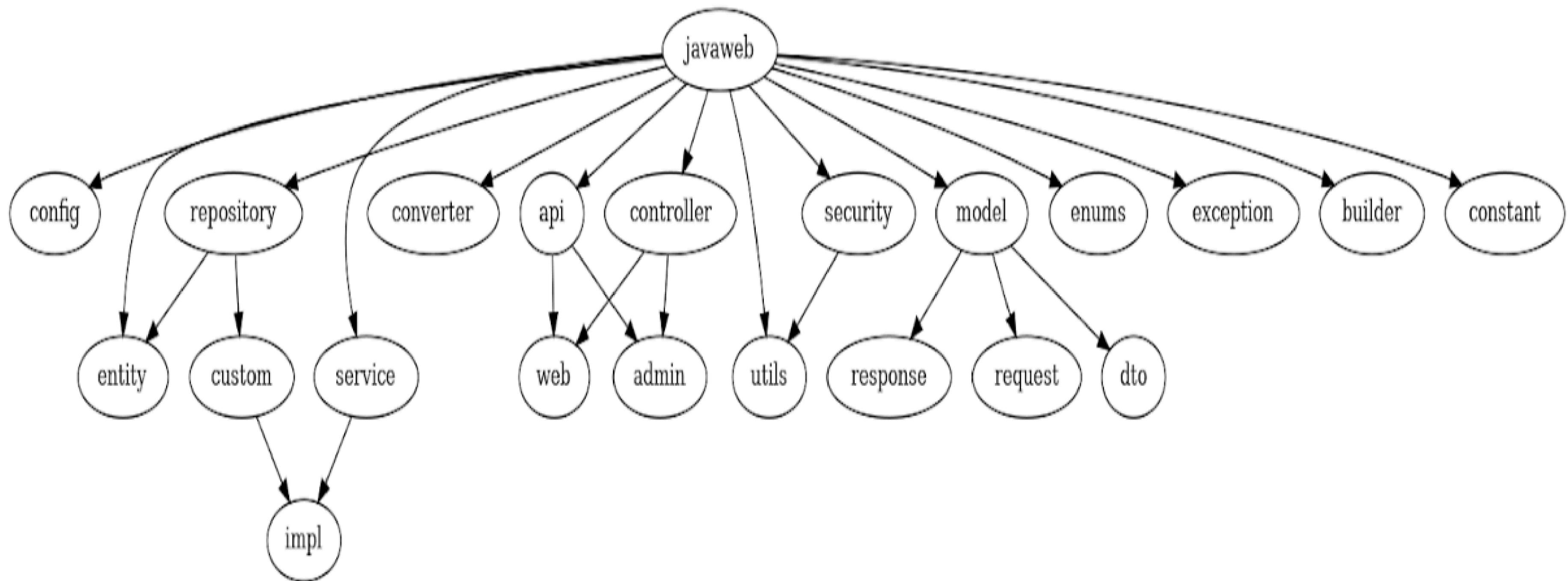


1. Biểu đồ UML
2. Chi tiết roles và thực thể
3. Các kỹ thuật lập trình hướng đối tượng đã sử dụng
4. Các công nghệ đã sử dụng
5. Demo sản phẩm

1. Biểu đồ UML

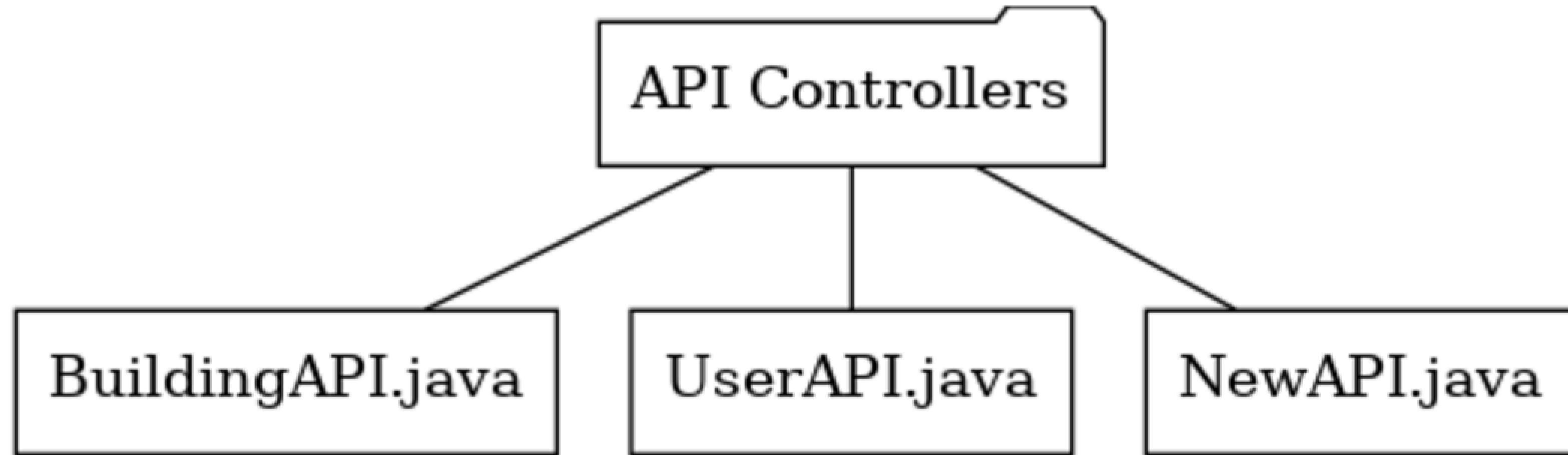
- Biểu đồ phụ thuộc gói
- Biểu đồ lớp

1. Biểu đồ UML - Biểu đồ phụ thuộc gói



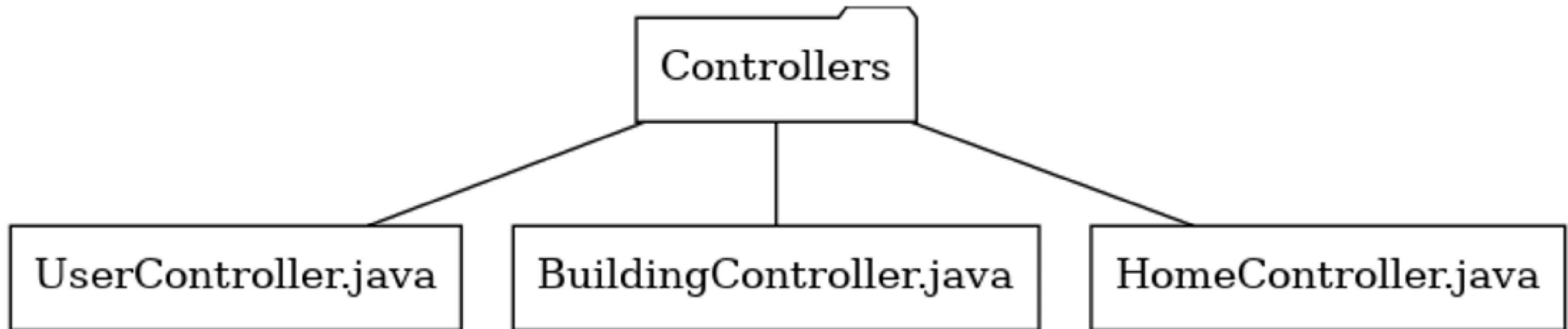
1. Biểu đồ UML - Biểu đồ lớp

- **API Controllers:** BuildingAPI.java, UserAPI.java, NewAPI.java: Xử lý yêu cầu HTTP từ client và điều phối dữ liệu giữa các tầng khác nhau



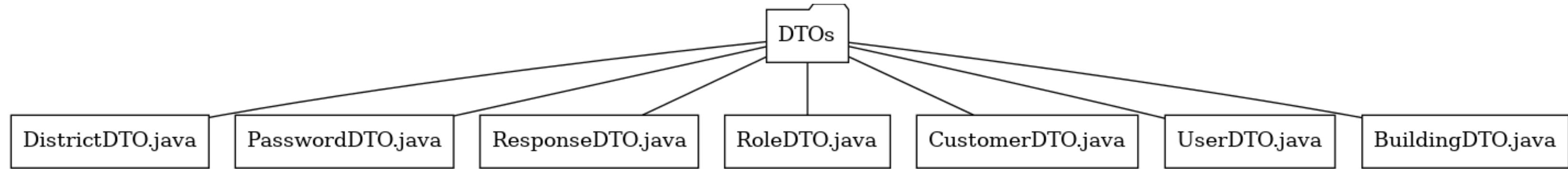
1. Biểu đồ UML - Biểu đồ lớp

- **Controllers:** `BuildingController.java`, `HomeController.java`, `UserController.java`: Quản lý điều hướng giao diện và xử lý các tác vụ liên quan đến người dùng



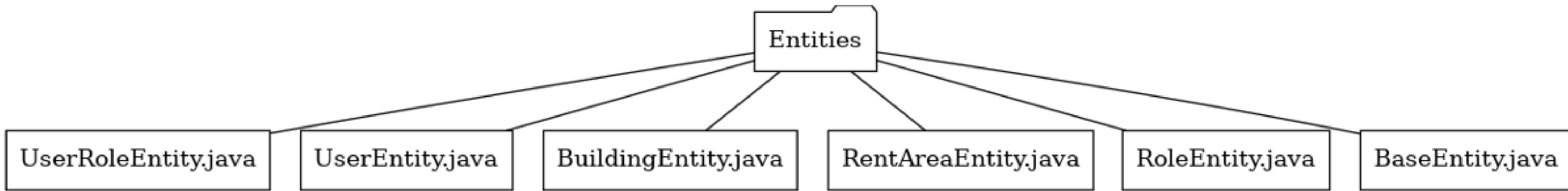
1. Biểu đồ UML - Biểu đồ lớp

- **DTOs**: `BuildingDTO.java`, `CustomerDTO.java`, `RoleDTO.java`, `UserDTO.java`: Đại diện dữ liệu gửi/nhận giữa client và server



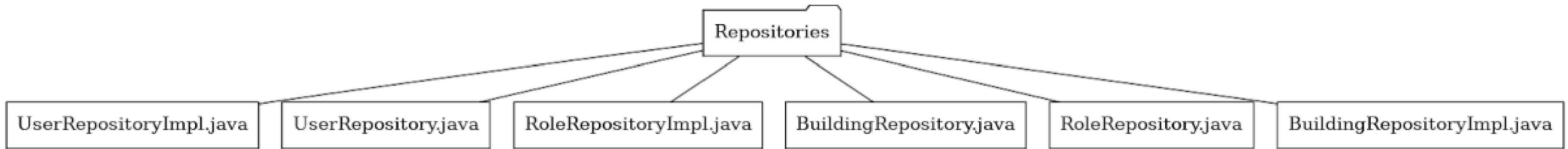
1. Biểu đồ UML - Biểu đồ lớp

- **Entities:** BaseEntity.java, UserEntity.java, RoleEntity.java, BuildingEntity.java: Các lớp mô tả cấu trúc bảng trong cơ sở dữ liệu



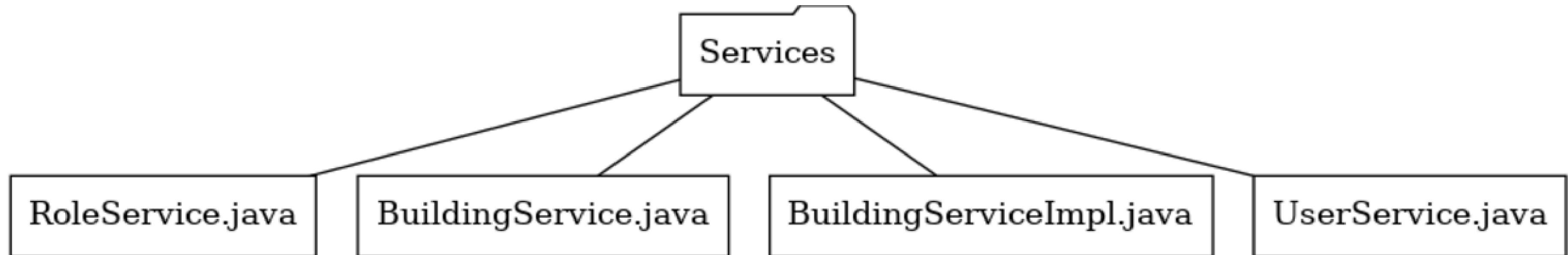
1. Biểu đồ UML - Biểu đồ lớp

- **Repositories:**
 - **UserRepository.java, BuildingRepository.java, RoleRepository.java:** Giao diện truy vấn dữ liệu từ database
 - Custom Implementations: **BuildingRepositoryImpl.java, RoleRepositoryImpl.java**



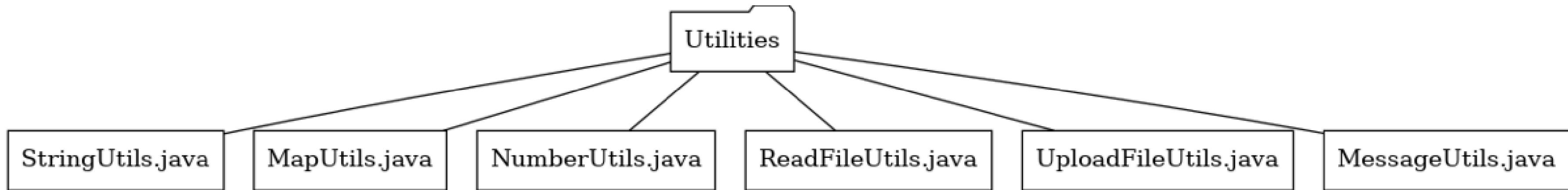
1. Biểu đồ UML - Biểu đồ lớp

- **Services**: BuildingService.java, UserService.java, RoleService.java: Xử lý logic nghiệp vụ



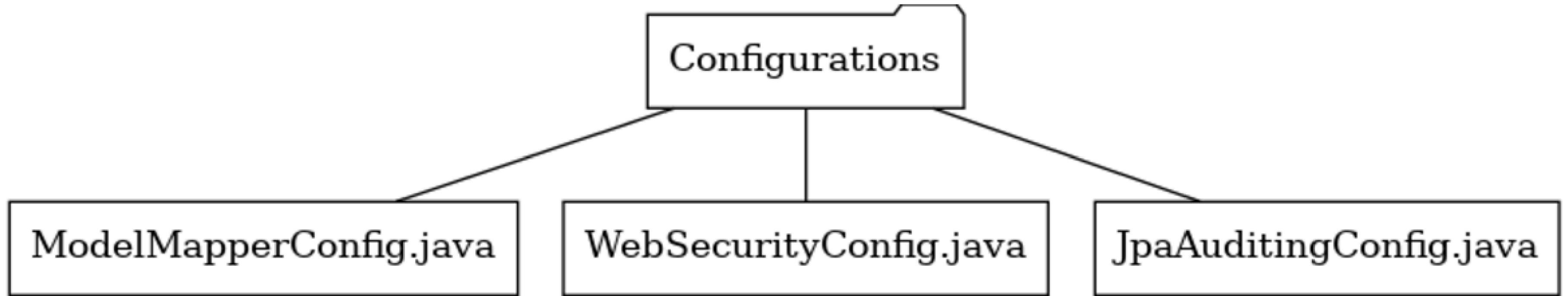
1. Biểu đồ UML - Biểu đồ lớp

- **Utilities:** MapUtils.java, StringUtils.java, UploadFileUtils.java, MessageUtils.java: Các tiện ích hỗ trợ trong quá trình xử lý dữ liệu



1. Biểu đồ UML - Biểu đồ lớp

- **Configurations:** `WebSecurityConfig.java`, `JpaAuditingConfig.java`, `ModelMapperConfig.java`: Cấu hình bảo mật và các thành phần Spring Boot



2. Chi tiết roles và thực thể

- Roles: Admin, Quản lý, Khách hàng
- Thực thể

2. Chi tiết roles và thực thể - Roles

Admin

Chức năng	Mô tả
Thêm/Sửa/Xóa tòa nhà	Cho phép thêm, chỉnh sửa hoặc xóa thông tin các tòa nhà
Quản lý người dùng	Tạo, sửa, xóa người dùng và phân quyền
Quản lý Roles	Tạo các vai trò mới và cấp quyền cho từng vai trò cụ thể
Thống kê báo cáo	Thống kê thông tin tòa nhà và người dùng dưới dạng biểu đồ và bảng
Quản lý diện tích thuê	Gán các diện tích thuê cho từng tòa nhà cụ thể

2. Chi tiết roles và thực thể - Roles

Quản lý

Chức năng	Mô tả
Cập nhật thông tin tòa nhà	Chỉnh sửa các thông tin như địa chỉ, diện tích...
Quản lý diện tích thuê	Cập nhật các diện tích còn trống hoặc đã cho thuê
Xử lý yêu cầu khách hàng	Tiếp nhận và phản hồi các yêu cầu từ phía khách hàng (Customer)



2. Chi tiết roles và thực thể - Roles

Khách hàng

Chức năng	Mô tả
Tìm kiếm tòa nhà	Lọc và tìm kiếm tòa nhà dựa trên diện tích, giá thuê, địa chỉ
Gửi yêu cầu thuê	Gửi thông tin yêu cầu thuê diện tích hoặc tòa nhà cụ thể
Xem lịch sử yêu cầu	Xem lại các yêu cầu thuê đã gửi và trạng thái phản hồi



2. Chi tiết role và thực thể - Thực thể

Tất cả các thực thể đều kế thừa từ lớp **AbstractDTO**, lớp này cung cấp các thuộc tính dùng chung như:

- **id**: Long
- **createdDate**: Date
- **createdBy**: String
- **modifiedDate**: Date
- **modifiedBy**: String

2. Chi tiết role và thực thể - Thực thể

1. **BuildingDTO**: Đại diện cho thông tin tòa nhà trong hệ thống
2. **UserDTO**: Quản lý thông tin người dùng và vai trò
3. **RoleDTO**: Đại diện cho vai trò của người dùng
4. **CustomerDTO**: Thông tin khách hàng trong hệ thống
5. **AssignmentBuildingDTO**: Gán tòa nhà cho nhân viên
6. **DistrictDTO**: Thông tin quận/huyện liên quan đến tòa nhà
7. **PasswordDTO**: Xử lý yêu cầu đổi mật khẩu của người dùng

3. Các kỹ thuật lập trình hướng đối tượng đã sử dụng

- Đóng gói
- Trừu tượng
- Kế thừa
- Đa hình



HUST

4. Các công nghệ đã sử dụng

Công nghệ sử dụng

- Java
- SpringBoot
- Gson
- LomBok
- MySQL





HUST

5. Demo sản phẩm