|  |  |
| --- | --- |
| **개발 프레임워크**  **샘플 가이드(참고用)** | |
| **차세대 OOO시스템 구축사업 (공공 분야)**  업무명  **Version 0.1**  발주사명 | |
|  |  |
| SK주식회사 | |

**제.개정 이력**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **버전** | **제.개정 일자** | **제.개정 내용** | **작성자** | **승인자** |
| 0.1 | YYYY-MM-DD | 최초 등록 | OOO | OOO |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Index / 목차

[1. 개요 3](#_Toc104901401)

[1.1. 개발 프레임워크 샘플 가이드 필요성 3](#_Toc104901402)

[1.2. 샘플 가이드 참고시 유의 사항 3](#_Toc104901403)

[2. 개발 프레임워크 패키지 4](#_Toc104901404)

[2.1. Core 패키지 구조 4](#_Toc104901405)

[2.2. Core 패키지 설명 5](#_Toc104901406)

[2.3. 개발 환경 (AS-IS vs TO-BE) 9](#_Toc104901407)

[2.4. 실행 환경 (AS-IS vs TO-BE) 10](#_Toc104901408)

[3. 소스 관리 13](#_Toc104901409)

[3.1. 소스 폴더 구조 13](#_Toc104901410)

[3.2. 폴더 구조 설명 16](#_Toc104901411)

# 개요

본 문서는 공공 분야 차세대 시스템 구축 개발 시 개발 프레임워크의 패키지 구조에 대한 샘플을 가이드 한다.

## 개발 프레임워크 샘플 가이드 필요성

개발 프레임워크에 대한 패키지 구조 샘플을 가이드 함으로써

* 개발자들의 생산성을 향상 시킨다..
* 유지보수 시점의 생산성을 증가시킨다.
* 프로그램 개발 소스에 대한 관리 표준을 마련 한다.

## 샘플 가이드 참고시 유의 사항

본 문서는 참고용으로 실제 공공 분야 차세대 시스템 구축시 필수적으로 지켜야할 사항은 아니며, **개발 프레임워크의 기본적인 패키지 구조 및 가이드**에 대한 설명을 하는 문서 이다.

프로젝트 수행시 개발 프레임워크 구축을 위한 참고 용도로만 사용하고, 실제로는 고객의 요구사항 및 아키텍처 구조에 맞게 Framework을 구축하여야 한다.

페이지의 마지막 줄

# 개발 프레임워크 패키지

## Core 패키지 구조

***Package Structure***

고객사 PJT명

**com(Core)**

config

adapter

annotation

aspect

datasource

filter

mapper

mvc

resolver

security

RequestMappingHandlerAdapter 클래스

세션 및 권한 관리, Argument 처리 어노테이션 클래스

데이터소스 생성을 위한 유틸리티 클래스

CookieLocaleResolver 클래스

AnnotationParamArgumentResolver, ModelAndViewArgumentResolver 클래스

익명 사용자 접근 제어를 위한 인증/인가 관련 클래스

constants

exception

공통 예외처리(Exception) 클래스가 위치한 패키지

log

service

support

별도 지원 기능 서비스 관련 클래스가 위치한 패키지

(Json 데이터 파싱 등)

구성 정보 설정 패키지

프로젝트 패키지명

공통 모듈 패키지명

컨트롤러, 서비스, 레파지토리 구성요소 로깅 및 사용자 정의 메시지 처리를 위한 Aspect 클래스

RequestBody 로깅 필터 및 http request를 여러 번 다시 읽을 수 있는 HttpServletRequest 객체 클래스

스프링 MyBatis 연동 지원 공통 parent DAO 클래스 (EgovAbstractMapper 클래스)

공통 상수 클래스가

위치한 패키지

로깅 이벤트 정보 관리 및 필터 관련 클래스가 위치한 패키지

공통 서비스 제공 클래스가 위치한 패키지

(Redis 캐시, Rest 통신, Object Convert 유틸 등)

유용한 유틸 클래스가

위치한 패키지

util

BigDecimal

BigExcelUtil

CamelUtil

FileUtil

JSONUtil

MessageUtil

StringUtil

SessionUtil

…

자료형에 따라 수 연산 기능을 지원하는 클래스

입력 문자열에 대해 Camel표기로 변환 지원하는 클래스

파일 업/다운로드, 압축/해제를 지원하는 클래스

JSON 데이터 변환을 지원하는 클래스

.properties 등록된 메시지 처리를 지원하는 클래스

세션 정보 관리를 지원하는 클래스

PropertyUtil, ObjectUtil 등 프로젝트 개발시 유용한 Util 관련 클래스 들을 얼마든지 추가할 수 있음

validation

vo

xss

크로스 사이트 스크립트 방지 처리 클래스가 위치한 패키지

annotation

validator

유효성 검증을 지원하는 객체 클래스

…

그 외 공통 모듈(Core 프레임워크)에 필요한 패키지 추가 가능

대량 엑셀 파일을 OOM없이 처리를 지원하는 클래스

(\*OOM: Out Of Memory)

문자열 형변환, 치환, 특수문자 처리 등을 지원하는 클래스

유효성 검증 클래스가

위치한 패키지

객체 값에 대한 유효성 검증을 지원하는 어노테이션 클래스

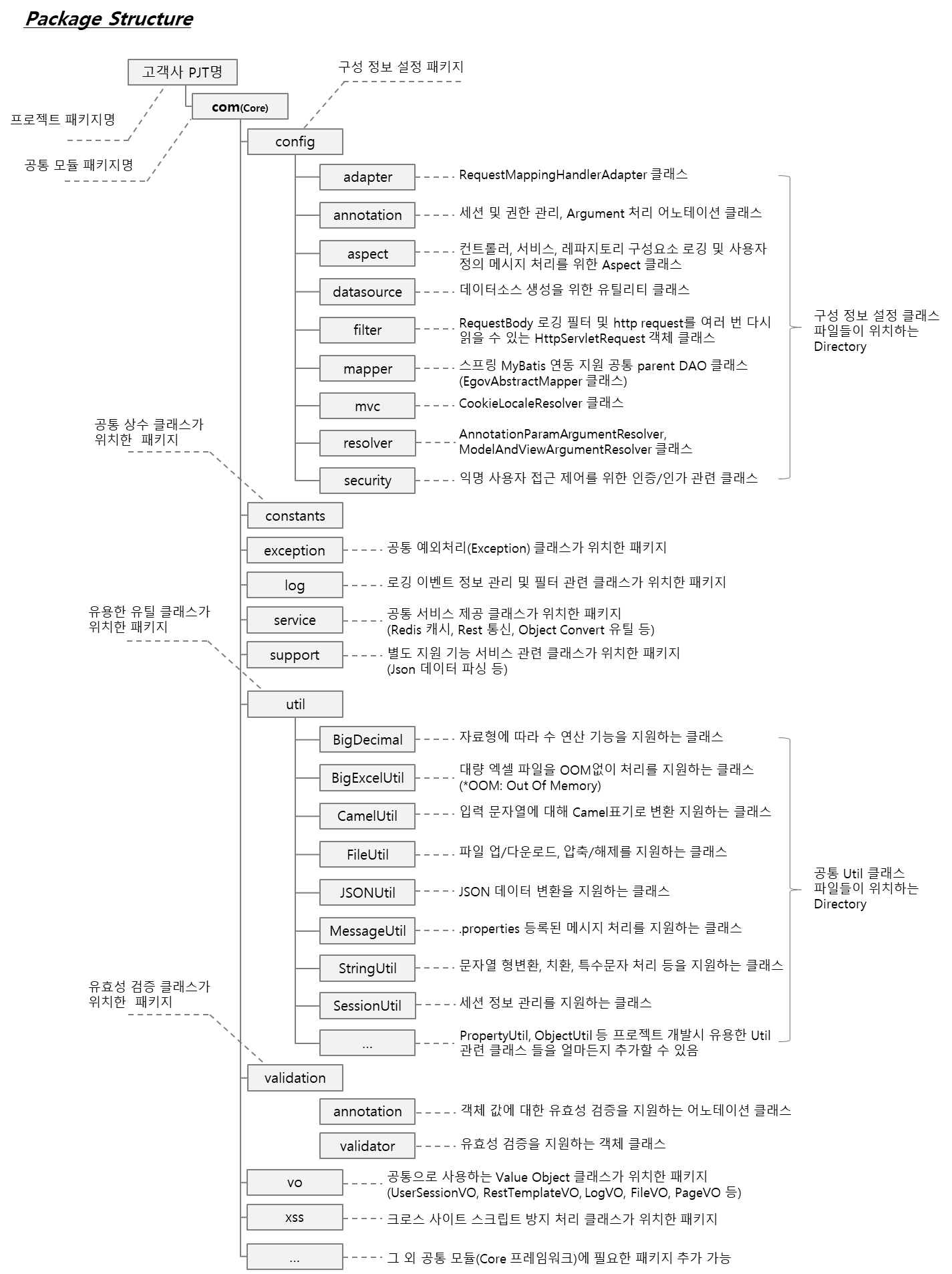
공통으로 사용하는 Value Object 클래스가 위치한 패키지

(UserSessionVO, RestTemplateVO, LogVO, FileVO, PageVO 등)

## Core 패키지 설명

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Package Structure | | | | 설명 |
| Depth 1 | Depth 2 | Depth 3 | Depth 4 |
| 고객사 PJT명  (프로젝트명) | com(Core)  (공통 모듈명) | Config  (구성 정보 설정) | adapter | * Handler Adpapter 클래스 |
| annotation | * 세션 및 권한 어노테이션 클래스 |
| aspect | * Aspect 클래스 |
| datasource | * 데이터소스 생성 클래스 |
| filter | * HttpServletRequest 객체 클래스 |
| mapper | * EgovAbstractMapper 클래스 |
| mvc | * CookieLocaleResolver 클래스 |
| resolver | * AnnotationParamArgumentResolver * ModelAndViewArgumentResolver |
| security | * 인증/인가 관련 클래스 |
| constants | - | * 공통 상수 관련 클래스 |
| exception | - | * 공통 예외처리(Exception) 클래스 |
| log | - | * 로깅 이벤트 정보 관리 및 필터 관련 클래스 |
| service | - | * 공통 서비스 제공 클래스 * Redis 캐시, Rest 통신, Object Convert 유틸 등 |
| support | - | * 별도 지원 기능 서비스 관련 클래스 * Json 데이터 파싱 등 |
| Util  (유용한 유틸) | BigDecimal | * 자료형에 따라 수 연산 기능을 지원하는 클래스 |
| BigExcelUtil | * 대량 엑셀 파일을 OOM없이 처리를 지원하는 클래스   (\*OOM: Out Of Memory) |
| CamelUtil | * 입력 문자열에 대해 Camel표기로 변환 지원하는 클래스 |
| FileUtil | * 파일 업/다운로드, 압축/해제를 지원하는 클래스 |
| JSONUtil | * JSON 데이터 변환을 지원하는 클래스 |
| MessageUtil | * .properties 등록된 메시지 처리를 지원하는 클래스 |
| StringUtil | * 문자열 형변환, 치환, 특수문자 처리 등을 지원하는 클래스 |
| SessionUtil | * 세션 정보 관리를 지원하는 클래스 |
| … | * PropertyUtil, ObjectUtil 등 프로젝트 개발시 유용한 Util 관련 클래스 들을 얼마든지 추가할 수 있음 |
| validation  (유효성 검증) | annotation | * 객체 값에 대한 유효성 검증을 지원하는 어노테이션 클래스 |
| validator | * 유효성 검증을 지원하는 객체 클래스 |
| vo | - | * 공통으로 사용하는 Value Object 클래스가 위치한 패키지   (UserSessionVO, RestTemplateVO, LogVO, FileVO, PageVO 등) |
| xss | - | * 크로스 사이트 스크립트 방지 처리 클래스가 위치한 패키지 |
| … | … | * 그 외 공통 모듈(Core 프레임워크)에 필요한 패키지 추가 가능 |

* Package Structure 캡쳐 화면 [참고用]



## 개발 환경 (AS-IS vs TO-BE)

차세대 구축 전 개발 환경과 차세대 구축 시 개발 환경에 대한 비교 현황은 다음과 같다.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 구분 | 서비스 명 | 서비스 상세 | AS-IS | TO-BE |
| 기본 환경 | JAVA | JDK | 1.6 | 11 |
| WEB | Nginx | - | 1.21.3 |
| WAS | Apache Tomcat | 6 | 9.0.56 |
| Undertow | - | 2.2.12.Final |
| Tmax JEUS | 7 | - |
| Oracle WebLogic | 11g | - |
| DBMS | Oracle | 11g | 19c |
| Redis | - | 6.2.6 |
| Message Queue | Kafka  **Sample** | - | 2.13 |
| 구현 도구 | Editor | Eclipse | 4.5.2 | 4.21.0 |
| ERD Editor | DA# Modeler | - | 5 |
| Code Inspection | PMD | 4.0.8 | 4.0.17 |
| 테스트 도구 | Test Case | Junit | 4.10 | 5.8.2 |
| Test Automation | Maven | - | 3.8.1 |
| 빌드 도구 | Build | Maven | - | 3.8.1 |
| CI Server | Jenkins | 1.653 | - |
| Tekton | - | 0.28.1 |
| 형상 관리 | Configuration Mgt. | Subversion | 1.8.13 | - |
| GitLab | - | 14.3.1 |

## 실행 환경 (AS-IS vs TO-BE)

차세대 구축 전 개발 Framework 실행 환경과 차세대 구축 시 개발 Framework 실행 환경에 대한 비교 현황은 다음과 같다.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 구분 | 서비스 명 | Artifact ID | AS-IS | TO-BE |
| Front-End | HTML5 Framework | WebSquare | - | 5 |
| 화면처리 레이어 | WEB | spring-web | 3.0.5.RELEASE | 5.3.14 |
| WEB, MVC | spring-webmvc | 3.0.5.RELEASE | 5.3.14 |
| Validation | commons-validator | 1.3.0 | - |
| Spring-boot-starter-validation | - | 2.6.2 |
| JSON parser | jackson-core | 1.9.5 | 2.13.1 |
| JSON binding | jackson-databind | 1.9.5 | 2.13.1 |
| 연계통합 레이어 | Integration Service | egovframework.rte.itl.integration | 2.0.0 | 4.0.0 |
| Web Service | egovframework.rte.itl.webservice | 2.0.0 | 4.0.0 |
| 데이터처리 레이어 | Data Access | egovframework.rte.psl.dataaccess | 2.0.0 | 4.0.0 |
| mybatis | 2.3.5 | 3.5.6 |
| Data Source | spring-jdbc | 3.0.5.RELEASE | 5.3.14 |
| Object/XML Mapping | spring-oxm | 1.5.6 | 5.3.14 |
| Transaction | spring-tx | 3.0.5.RELEASE | 5.3.14 |
| 공통기반 레이어 | Compress/Decompress | commons-compress | 1.2 | 1.19 |
| File Upload/Download | commons-fileupload | 1.2.1 | 1.4 |
| Object Pooling | commons-pool | 1.3 | - |
| commons-pool2 | - | 2.11.1 |
| XML Manipulation | dom4j | 1.6.1 | 1.6.1 |
| Common Util | egovframework.rte.fdl.cmmn | 2.0.0 | 4.0.0 |
| Encryption/Decryption | egovframework.rte.fdl.crypto | 2.0.0 | 4.0.0 |
| Excel | egovframework.rte.fdl.excel  **Sample** | 2.0.0 | 4.0.0 |
| File Handling | egovframework.rte.fdl.filehandling | 2.0.0 | 4.0.0 |
| ID Generation | egovframework.rte.fdl.idgnr | 2.0.0 | 4.0.0 |
| Logging | egovframework.rte.fdl.logging | 2.0.0 | 4.0.0 |
| Property | egovframework.rte.fdl.property | 2.0.0 | 4.0.0 |
| Server Security | egovframework.rte.fdl.security | 2.0.0 | 4.0.0 |
| String Util | egovframework.rte.fdl.string | 2.0.0 | 4.0.0 |
| XML Manipulation | egovframework.rte.fdl.xml | 2.0.0 | 4.0.0 |
| Logging | Slf4j | 1.6.4 | 1.7.32 |
| log4j | - | 1.2.17 |
| Excel | poi | 3.8-20120326 | 4.1.2 |
| poi-ooxml | 3.8-20120326 | 4.1.2 |
| poi-ooxml-schemas | 3.8-20120326 | 4.1.2 |
| AOP | spring-aop | 3.0.5.RELEASE | 5.3.14 |
| Core | spring-beans | 3.0.5.RELEASE | 5.3.14 |
| spring-context | 3.0.5.RELEASE | 5.3.14 |
| spring-context-support | 3.0.5.RELEASE | 5.3.14 |
| spring-core | 3.0.5.RELEASE | 5.3.14 |
| spring-expression | 3.0.5.RELEASE | 5.3.14 |
| Server Security | spring-security-core | 2.0.4 | 5.6.1 |
| Unmarshalling | xmlbeans | 2.3.0 | 3.0.1 |

페이지의 마지막 줄

# 소스 관리

본 문서는 공공 분야 차세대 시스템 구축 개발 시 개발 프로젝트 소스의 폴더 구조에 대한 샘플을 가이드 한다.

## 소스 폴더 구조

본 문서는 참고용으로 실제 공공 분야 차세대 시스템 구축시 필수적으로 지켜야할 사항은 아니며, **개발 프로젝트의 기본적인 소스 폴더 구조 및 가이드**에 대한 설명을 하는 문서 이다.

프로젝트 수행시 개발 프로젝트 구축을 위한 참고 용도로만 사용하고, 실제로는 고객의 요구사항 및 아키텍처 구조에 맞게 프로젝트 소스 폴더 구조를 구축하여야 한다.

***프로젝트 소스 폴더 구조***

프로젝트 명

프로젝트 패키지명

src/main/java

최상위 비즈니스 모듈 패키지명

비즈니스 모듈명

controller

service

VO

Biz 모듈명1

Biz 모듈명n

controller

service

VO

공통 모듈명

공통코드 모듈

controller

service

VO

Controller 클래스 (웹처리를 위한 데이터 송수신 및 담당 서비스 호출)

Service 클래스 (비즈니스 로직 서비스)

VO 클래스 (Value Object 객체)

상동(上同)

비즈니스 모듈에 사용되는 공통코드 그룹 Controller, Service, VO 클래스

프로젝트 자바 소스 패키지

하위 비즈니스 모듈 패키지명

(다수의 모듈 존재)

공통코드 및 공통설정 패키지명

공통코드 그룹 모듈명

config

api

datasource

interceptor

mapper

message

mvc

security

tx

mapper

…

그 외 공통 모듈에 필요한 클래스 추가 가능

API 테스트를 위한 Swagger Config 설정 클래스

jdbc properties, Configuration properties 설정 클래스

로그인 Session 처리를 위한 인터셉터 설정 클래스

(URL 체크 등 포함)

SqlSessionConfig 설정 클래스

Spring Message 설정, DB MessageSource 객체 클래스

등이 위치

WebMvcConfig 설정, error 페이지 및 Redis 구동 설정 등이 위치한 클래스

API 접근에 대한 인가 정보 확인 객체, Controller에 접근 권한을 설정하기 위한 객체, Spring Security 등이 위치

트랜잭선 처리 설정 클래스

xxxWebApplication.java

어플리케이션을 시작하는 Main()메소드가 있는 클래스

공통 설정 모듈 패키지명

EgovAbstractMapper 클래스

src/main/resources

config

log

mapper

message

mapper

비즈니스 모듈명

Biz 모듈명1

Biz 모듈명n

공통 모듈명

auth

code

msg

Log4jdbc.log4j.properties

프로젝트 리소스 패키지

Logback설정, mybatis설정 정보 파일(mybatis-config.xml), 메시지 설정, JDBC

연결 설정 및 서버 설정 등 구성정보 파일 (application.yml) 등이 위치

Logback 설정 xml 파일 위치

mybatis-config.xml 설정 파일 위치

메시지 Properties 설정 파일 위치

application.yml

자바에서 일반적인 Spring설정 파일

(jdbc, 세션, context-path 설정 등)

SQL Mapping XML 파일 (SQL.xml) 위치

비즈니스 모듈 패키지명

(다수의 모듈 존재)

mybatis Mapper 인터페이스를 활용한 업무 쿼리문 작성 파일(.xml)

권한 관련 쿼리문 작성 파일(.xml)

공통 코드 쿼리문 작성 파일(.xml)

메시지 쿼리문 작성 파일(.xml)

공통코드 및 공통설정 패키지명

lucy-xss-servlet-filter-rule.xml

URL별 필터링 룰 작성 파일(.xml)

Log4j 설정 파일

src/test/java

JRE System Library

Maven Dependencies

target

자바 라이브러리 파일 위치

메이븐 의존성 파일 위치

메이븐 빌드 결과 경로

deployment.yml

Dockerfile

mvnw

mvnw.cmd

pom.xml

Docker 컨테이너를 실행시키기 위한 파일 위치

컨테이너 배포를 위한 설정 파일 위치

메이븐 시작 배치 스크립트

메이븐 의존성 설정 파일

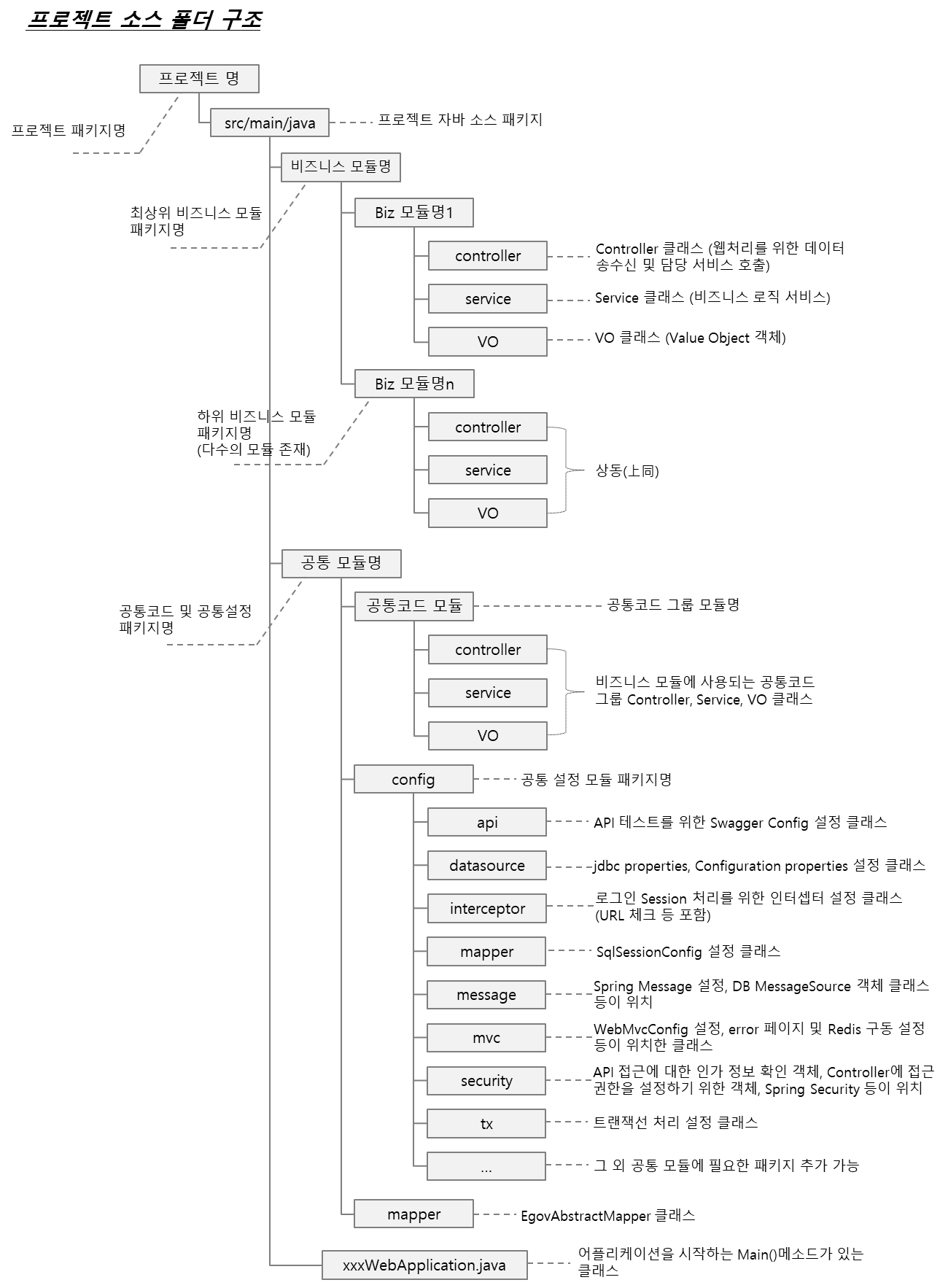
메이븐 래퍼, 메이븐 미설치시 mvn 명령 대신 사용

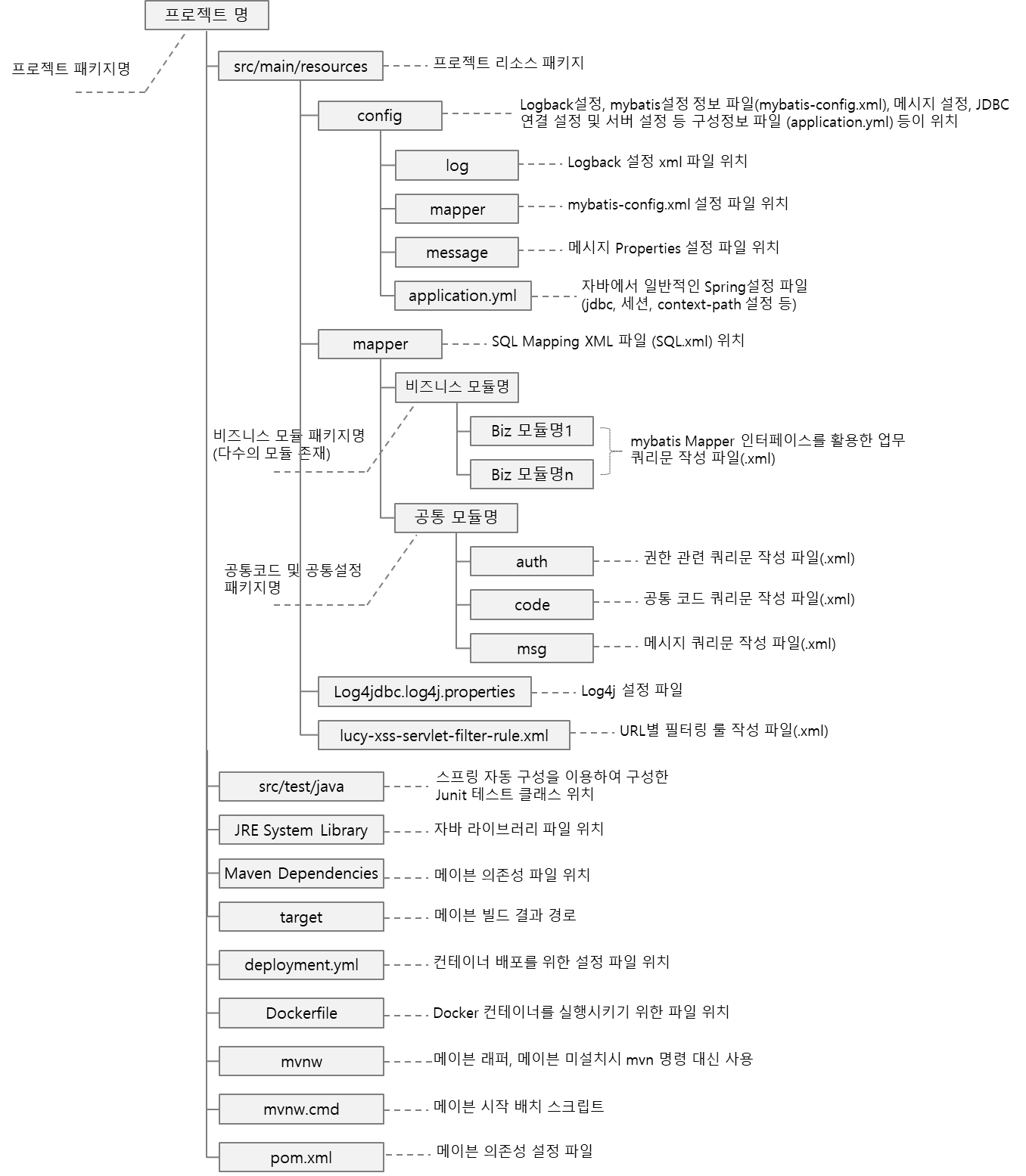
스프링 자동 구성을 이용하여 구성한 Junit 테스트 클래스 위치

## 폴더 구조 설명

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 소스 폴더 구조 (Hierarchical) | | | | | 설명 |
| Depth 1 | Depth 2 | Depth 3 | Depth 4 | Depth 5 |
| 프로젝트명 | **src/main/java**  **(자바 소스 패키지)** | 비즈니스 모듈명 (최상위) | Biz모듈명1 | Controller | * Controller 클래스 (웹처리를 위한 데이터 송수신 및 담당 서비스 호출) |
| Service | * Service 클래스 (비즈니스 로직 서비스) |
| VO | * VO 클래스 (Value Object 객체) |
| Biz모듈명n | Controller | * 상동(上同) |
| Service |
| VO |
| 공통 모듈명  (common) | 공통코드 모듈  (code) | Controller | * 비즈니스 모듈에 사용되는 공통코드 그룹 Controller, Service, VO 클래스 |
| Service |
| VO |
| 공통설정 모듈  (config) | api | * API 테스트를 위한 Swagger Config 설정 클래스 |
| datasource | * jdbc properties, Configuration properties 설정 클래스 |
| interceptor | * 로그인 Session 처리를 위한 인터셉터 설정 클래스 (URL 체크 등 포함) |
| mapper | * SqlSessionConfig 설정 클래스 |
| message | * Spring Message 설정, DB MessageSource 객체 클래스 등이 위치 |
| mvc | * WebMvcConfig 설정, error 페이지 및 Redis 구동 설정 등이 위치한 클래스 |
| security | * API 접근에 대한 인가 정보 확인 객체, Controller에 접근 권한을 설정하기 위한 객체, Spring Security 등이 위치 |
| tx | * 트랜잭선 처리 설정 클래스 |
| … | * 그 외 공통 모듈에 필요한 패키지 추가 가능 |
| mapper | xxxMapper.java | * EgovAbstractMapper 클래스 |
| xxxxxxxWebApplication.java | | | * 어플리케이션을 시작하는 Main()메소드가 있는 클래스 |
| **src/main/resources**  **(리소스 패키지)** | config | - | - | * Logback설정, mybatis설정 정보 파일(mybatis-config.xml), 메시지 설정, JDBC 연결 설정 및 서버 설정 등 구성정보 파일 (application.yml) 등이 위치 |
| log | Logback.xml | * Logback 설정 xml 파일 위치 |
| mapper | Mybatis-config.xml | * mybatis-config.xml 설정 파일 위치 |
| message | Message.  properties | * 메시지 Properties 설정 파일 위치 |
| application.yml | local  dev  파일 포함 | * 자바에서 일반적인 Spring설정 파일 (jdbc, 세션, context-path 설정 등) |
| mapper | - | - | * SQL Mapping XML 파일 (SQL.xml) 위치 |
| 비즈니스 모듈명 | Biz 모듈명1 | * mybatis Mapper 인터페이스를 활용한 업무 쿼리문 작성 파일(.xml) |
| Biz 모듈명n |
| 공통 모듈명 | auth | * 권한 관련 쿼리문 작성 파일(.xml) |
| code | * 공통 코드 쿼리문 작성 파일(.xml) |
| msg | * 메시지 쿼리문 작성 파일(.xml) |
| Log4jdbc.log4j.  properties | slf4j 등 | | * Log4j 설정 파일 |
| lucy-xss-servlet-filter-rule.xml | lucy-xss-servlet  config설정 등 | | * URL별 필터링 룰 작성 파일(.xml) |
| **src/test/java** | xxxxxxxWebApplicationTests.java | | | * 스프링 자동 구성을 이용하여 구성한 Junit 테스트 클래스 위치 |
| JRE System Library | Java, jdk관련 라이브러리 등 | | | * 자바 라이브러리 파일 위치 |
| Maven Dependencies | spring-boot, logback, spring-web, Jakarta, Jackson, spring, spring-data, Junit 등 | | | * 메이븐 의존성 파일 위치 |
| target | xxxxxxx-0.0.1-SNAPSHOT.jar 등 | | | * 메이븐 빌드 결과 경로 |
| deployment.yml | Containers spec, name, image, ports, replicas, apiVersion, namespace 설정 등 | | | * 컨테이너 배포를 위한 설정 파일 위치 |
| Dockerfile | - | | | * Docker 컨테이너를 실행시키기 위한 파일 위치 |
| mvnw | - | | | * 메이븐 래퍼, 메이븐 미설치시 mvn 명령 대신 사용 |
| mvnw.cmd | - | | | * 메이븐 시작 배치 스크립트 |
| pom.xml | dependencies 설정 등 | | | * 메이븐 의존성 설정 파일 |

* 프로젝트 소스 폴더 구조 캡쳐 화면 [참고用]





페이지의 마지막 줄