



# 전자정부 표준프레임워크 개발가이드

## 요약

전자정부 표준프레임워크를 적용하여 개발하는 방법을 설명한다.

표준프레임워크센터  
egovframesupport@gmail.com

## 1. 개요

본 문서는 개발자들이 전자정부 표준프레임워크 기반으로 개발하기 위해 필요한 환경구성, 각종 표준 및 가이드를 제시하는 데 목적이 있다.

개발하기에 앞서 전자정부 표준프레임워크 구성에 대한 이해를 위해 구성 및 구성요소의 기능 및 역할에 대해 알아보도록 한다.

전자정부 표준프레임워크는 아래와 같이 4대환경으로 구성되어있다.



### [표준프레임워크 구성]

표준프레임워크로 개발을 하기 위해 위의 4대 환경 중 필수요소는 “실행환경” 이며, 선택요소로는

“개발환경”, “운영환경”을 사용할 수 있다. “관리환경”은 표준프레임워크센터에서 표준프레임워크를 관리/기술지원을 하기 위해 사용하는 환경으로 프로젝트 활용에는 제외하도록 한다.

“공통컴포넌트”는 표준프레임워크로 개발 된 모듈로써, 공통적으로 재사용이 가능하도록 개발된 모듈이다. 공통컴포넌트는 선택활용이 가능하고, 모듈 내 수정도 가능하다.



#### [공통컴포넌트 구성]

또한 모바일 관련해서는 모바일 웹(모바일 화면처리)과 오바일 디바이스API를 개발할 수 있는 기반을 아래와 같이 제공하고 있다.



전자정부 표준프레임워크는 시스템 구축 시 필요로 하는 기본적인 어플리케이션 아키텍처 및 구성과 함께 재사용 가능한 모듈인 공통컴포넌트를 함께 제공하는 표준프레임워크로서 각 구성요소의 기능 및 역할은 아래와 같다.

구분	기능 및 역할
실행환경	전자정부 사업에서 개발하는 업무 프로그램의 실행에 필요한 공통모듈 등 업무 프로그램 개발 시 화면, 서버 프로그램, 데이터 개발, 배치처리기능 개발을 표준화가 용이하도록 지원하는 응용프로그램환경
개발환경	전자정부 업무 프로그램의 개발에 필요한 환경 제공 데이터개발도구, 테스트자동화도구, 코드검사도구, 템플릿 프로젝트 생성도구, 공통컴포넌트 조립도구, 맞춤형 개발환경 구성도구, 서버환경관리도구, 모바일용 표준 소스코드 생성도구, 모바일용 템플릿 프로젝트 생성도구, 모바일용 공통컴포넌트 조립도구, 모바일용 맞춤형 개발환경 구성도구, 서버용 개발환경(Windows, Unix 계열)설치, 배치템플릿프로젝트 생성도구, 배치작업파일 생성도구, 배치실행파일 생성도구 등
운영환경	실행환경에서 운영되는 서비스를 운영하기 위한 환경제공(모니터링, 배포, 관리시스템 등) 배치환경을 운영하기 위한 환경제공(배치실행, 스케줄링, 결과모니터링 등)
관리환경	개발프레임워크 및 공통서비스를 각 개발 프로젝트에 배포 및 관리하기 위한 모듈
공통컴포넌트	전자정부 공통컴포넌트는 전자정부 사업에서 응용SW 개발 시 공통적으로 활용하기 위하여, 재사용이 가능하도록 개발한 어플리케이션의 집합
모바일 디바이스 API	CoreAPI : 모바일 하이브리드 어플리케이션에서 모바일 디바이스 자원에 대한 직접적인 접근과 활용이 가능한 다양한 API 제공 가이드 어플리케이션(앱) : 디바이스 API를 손쉽게 접근할 수 있도록 제공하는 디바이스 API의 활용 예제
모바일 디바이스 API 실행환경	디바이스 어플리케이션이 웹 리소스 기반으로 구현 및 실행될 수 있도록 지원하는 응용 프로그램 환경 디바이스API, 자바스크립트 프레임워크, 하이브리드 프레임워크 등
모바일 디바이스 API 개발환경	Android 기반환경에서의 디바이스 어플리케이션 개발을 위한 Eclipse 플러그인과 iOS 환경에서의 개발을 위한 Xcode내의 프레임워크 프로젝트로 구성되어 있다.

### [구성요소의 기능 및 역할]

## 2. 준비

### 2.1 사용자 환경

전자정부 표준프레임워크 개발을 위해서는 설치환경 및 시스템의 조건을 먼저 확인하도록 한다.

① 자바 기반의 웹 응용 시스템(WAS가 존재하는 경우)

② (2.7 기준) JavaEE(J2EE) 5 혹은 JDK1.5 이상의 환경

(단, 개발환경 2.7 이상에서는 JDK 1.6 이상 필요)

(3.0 기준) JavaEE(J2EE) 6 혹은 JDK1.6 이상의 환경

③ 신규 시스템으로써, 기존 시스템과 물리적 혹은 논리적으로 구분되는 경우

실행환경 내 모바일 표준프레임워크의 사용자 경험(UX) 지원 기능은 프레임워크와 개발 언어 종류에 상관없이 활용가능 (javascript 기반)

### 2.2 개발환경 설치

개발환경은 직접 설치/설정 하여 사용하는 방법과 제공되는 클라우드 환경의 활용이 가능하다. 전자정부 표준프레임워크 개발환경의 사용은 필수사항이 아니므로 개발편의에 따라 선택적으로 사용하도록 한다.

본 가이드는 전자정부 표준프레임워크 개발환경에 기반하여 설명한다.

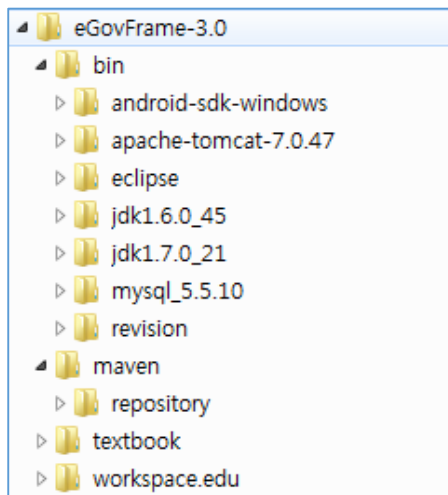
#### ① 로컬개발환경 설치 및 설정

로컬개발환경 설치하는 아래 가이드의 내용을 참고하여 설치하도록 한다.

<http://www.egovframe.go.kr/wiki/doku.php?id=egovframework:dev2:clntinstall>

#### ② 로컬개발환경 디렉터리 구조 (예시)

로컬개발환경은 아래와 같은 디렉토리로 구성할 수 있다.

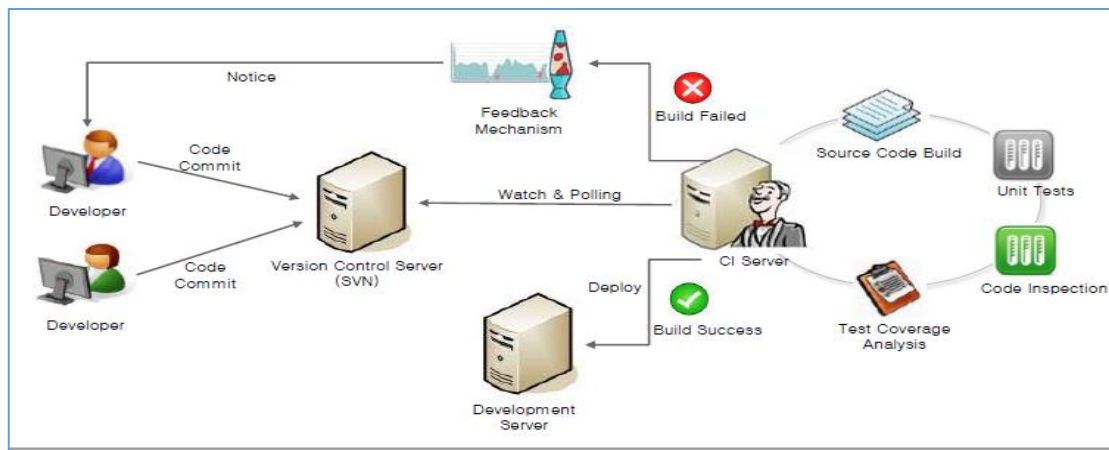


디렉토리	설명
bin	실행 파일
apache-tomcat-6.0.37	Apache Tomcat
apache-tomcat-7.0.47	
jdk1.6.0/jdk1.7.0	JDK_HOME
Eclipse	개발환경이 포함된 Eclipse JEE Kepler SR2 (4.3.2)
maven/repository	Local Maven Repository
mysql_5.5.10	실습용 DB
textbook	교재 파일
workspace.edu	Eclipse Workspace

### ③ 서버개발환경 구성 (예시)

서버개발환경은 CI(Continuous Integration)을 위한 환경으로, 표준프레임워크에서 제공하는 환경을 사용해도 되고, 별도로 구성하는 것도 가능하다. CI 개념 및 활용관련 추가적인 설명은 다음의 Wiki 가이드 내용을 참고하도록 한다.

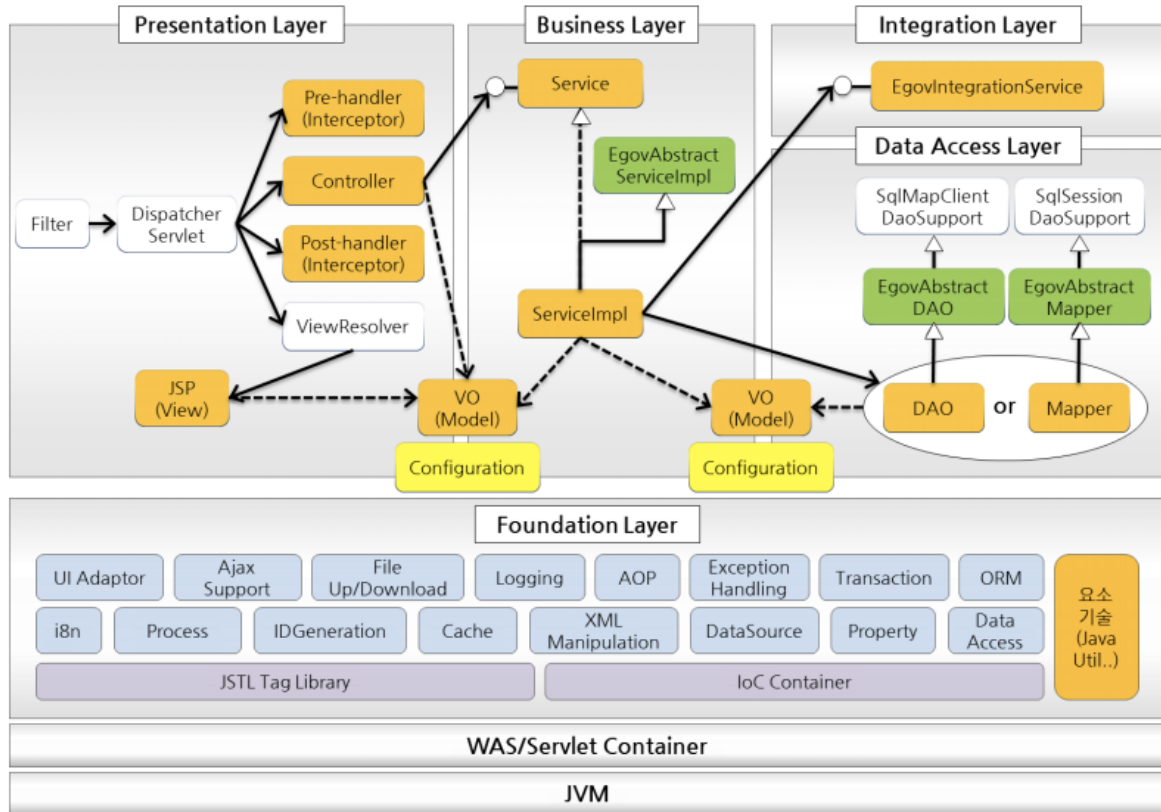
<http://www.egovframe.go.kr/wiki/doku.php?id=egovframework:dev2:server>



서비스	소프트웨어	비고
형상관리	Subversion	프로젝트에서 사용 파일들을 관리하기 위한 시스템으로써, 소스파일의 일관성 있는 관리 지원.
CI Server	Hudson	CI서버로써 자동 빌드 및 빌드 결과를 개발자에게 피드백 하는 메커니즘 제공
Build	Maven	POM기반 Build부터 의존성관리 및 Deploy관리 지원
Repository 관리	Nexus	Maven Repository manager로 프로젝트에서 필요로 하는 내부 Repository 구성

### 3. 표준프레임워크 기반 개발방법

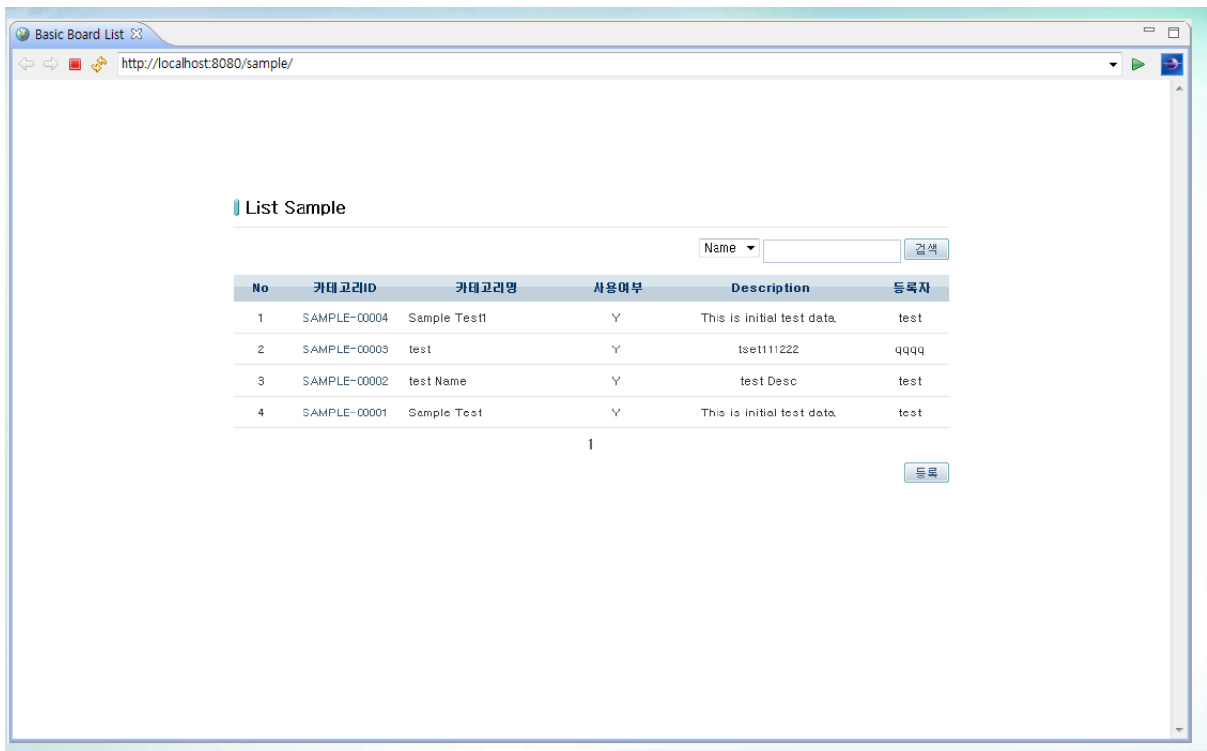
#### 3.1 표준프레임워크 아키텍처



#### 3.1 표준프레임워크 프로그램 개발방법

표준프레임워크로 Board(게시판) 프로그램을 만드는 예를 통해 각 구성항목을 설명하도록 한다.

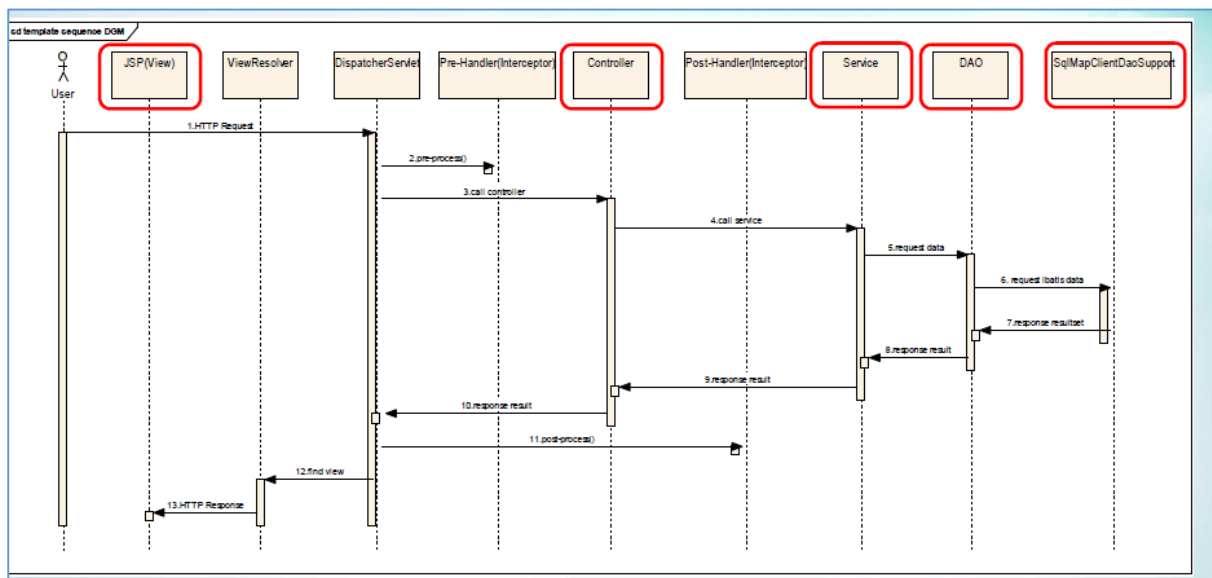
게시판 프로그램은 CRUD 기능이 모두 포함된 모듈로 화면을 구성하는 JSP와 Java (Controller, Service interface, Service, Dao), DB 처리를 위한 SQL 그리고 설정파일들로 구성된다.



### [게시판 실행화면]

게시판 프로그램은 아래 sequence diagram과 같은 순서로 처리된다.

(Sequence Diagram의 경우, 표준프레임워크 개발환경의 UML 모델링 기능을 활용하여 작성 가능하다. – Use Case Diagram, Class Diagram, Sequence Diagram)



### [Sequence Diagram]



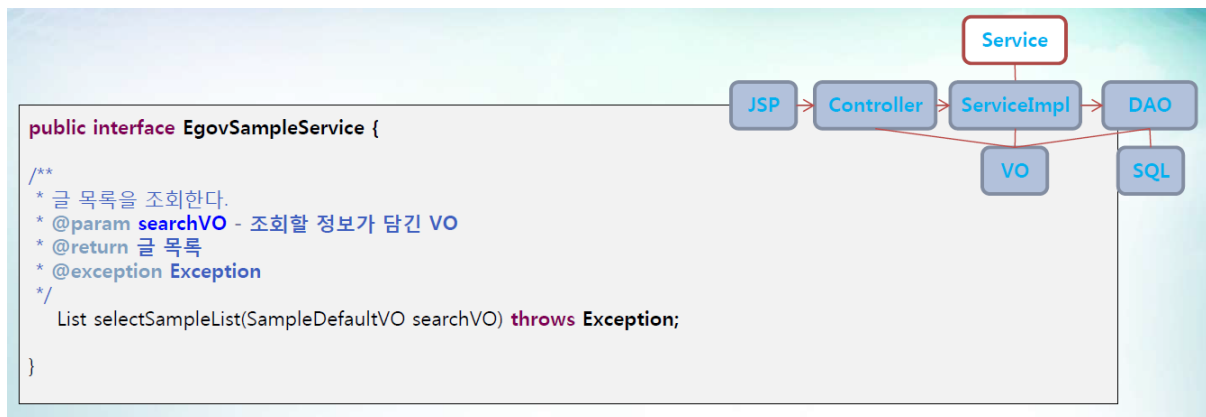
표준프레임워크 개발환경의 프로젝트 생성 메뉴를 이용하면 표준프레임워크 기반 프로젝트를 구조만 생성하거나, 소스를 포함한 프로젝트를 생성할 수 있다.

**Controller** - 사용자 Request(요청)을 처리하고, 데이터 바인딩, 폼 처리 또는 멀티 액션등의 기능 구현



[Controller]

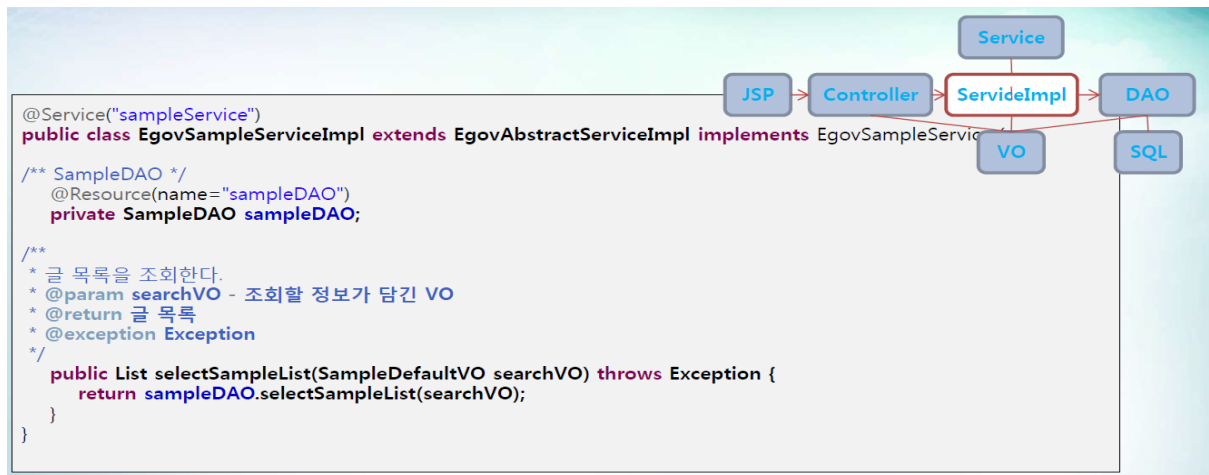
**Service** - 기능적 operation 에 대한 서비스를 인터페이스 정의



[Service]

**ServiceImpl** : 서비스 인터페이스에서 정의한 operation 의 상세 구현 클래스

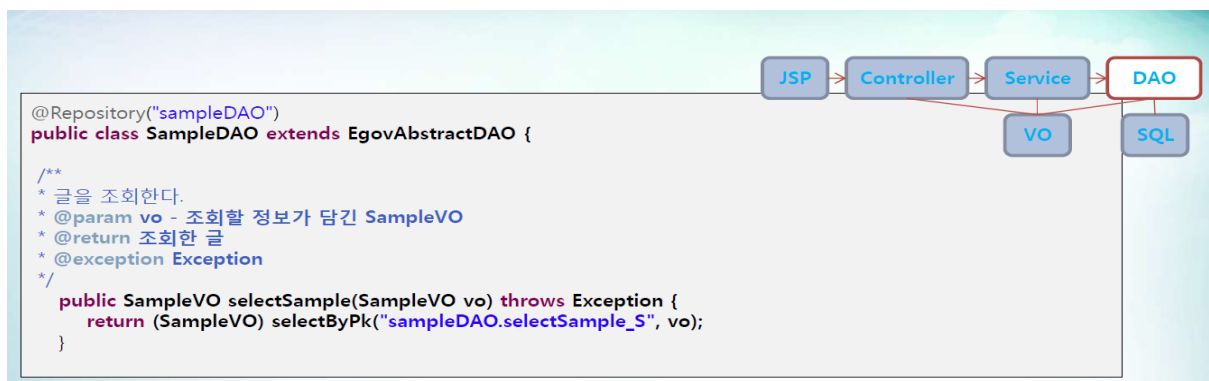
ServiceImpl은 EgovAbstractServiceImpl (또는 AbstractServiceImpl)을 반드시 extends하여 작성



### [ServiceImpl]

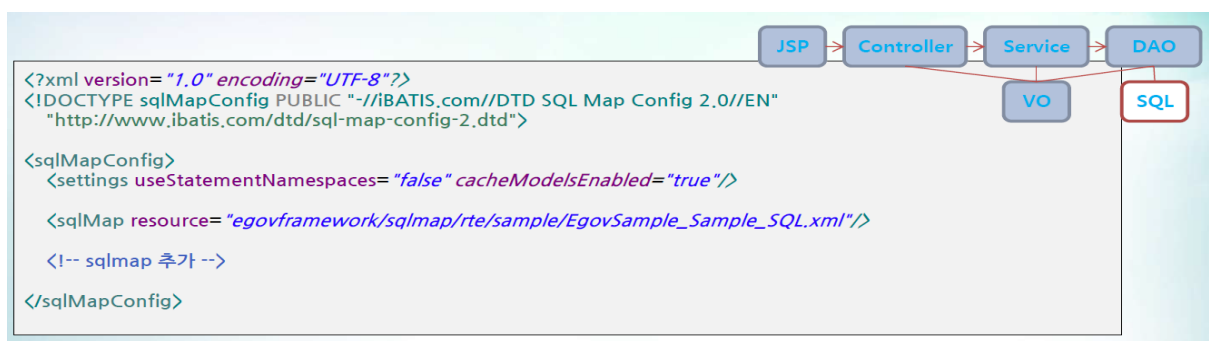
**DAO** : 데이터 처리 (iBatis/MyBatis/Spring Data 연동)

DAO는 EgovAbstractDAO를 반드시 extends 하여 작성한다.

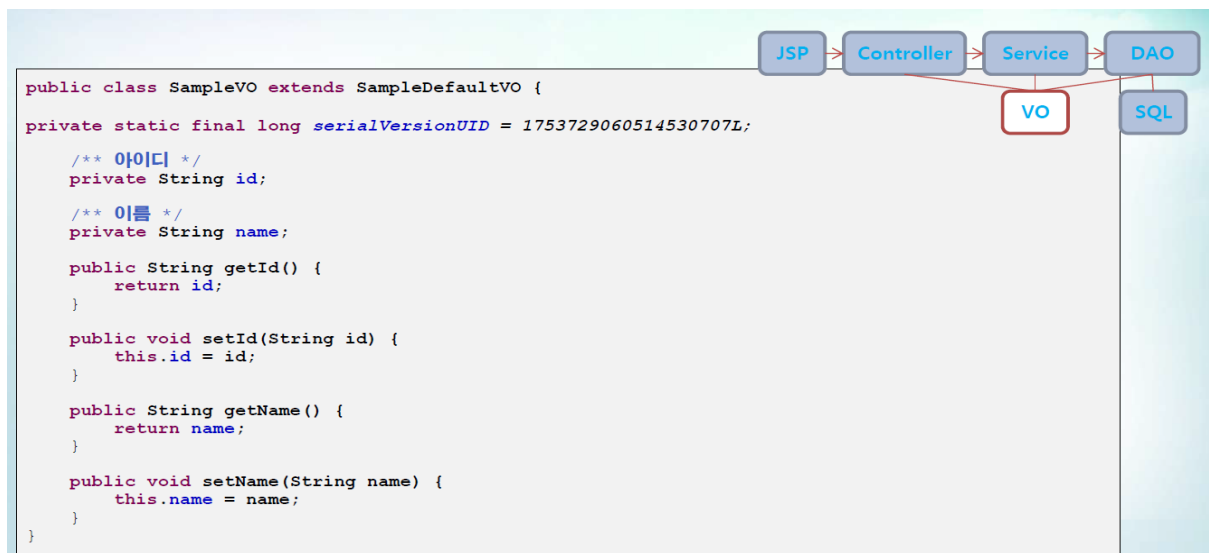


### [DAO]

**iBatis SQL Map Config** : iBatis 연동을 위한 기본 메일 설정 파일로서 transaction 관리, sql mapping xml 파일 경로 설정들을 포함



### [iBatis SQL Map Config]

**iBatis SQL Map** : 실행된 SQL문과 결과 mapping 정보 정의**[iBatis SQL Map]****VO(Value Object)** : 클래스 간 데이터 전달**[Value Object]**

## 4. 코딩규칙 - 패키지 구성

구분	Level1	Level 2	Level 3	Level4	설명
패키지 명	egovframework				표준프레임워크 공통 패키지
		rte			실행환경
			fdl		공통기반레이어
				comn	공통기반레이어의 공통 서비스
				logging	공통기반레이어의 로깅 서비스
			psl		데이터처리레이어
				dataaccess	데이터처리레이어 Dataaccess 서비스
			ext		확장 레이어
				XXX (신규기능명)	신규기능서비스명
		dev			개발환경
			ext		확장 레이어
				XXX(신규기능명)	신규기능서비스명
		com			공통컴포넌트
			ext		확장레이어
				XXX(신규기능명)	신규기능서비스명

- ✓ Level 1 : 표준프레임워크 공통 패키지
- ✓ Level 2 : 표준프레임워크 4대환경( 실제 코드상으로는 실행환경(rte), 개발환경 (dev), 운영 환경(XXX), 관리환경으로 구분할 수 있음.
- ✓ Level 3 : 각 환경의 레이어 별 패키지  
fdl : 공통기반 레이어, psl : 데이터처리 레이어 등으로 구분한다. 신규 확장의 경우 본 클래스를 ext로 한다.
- ✓ Level4 : 레이어를 구성하고 있는 서비스 별 패키지  
(ex: comn : 공통기반 레이어의 공통서비스, logging : 공통기반 레이어의 로깅서비스)

따라서 표준프레임워크 신규기능의 경우 확장의 영역으로 아래와 같은 규칙에 따라 패키지를 구성하도록 한다.

#### 실행환경의 경우

Level 1 + level 2 + level 3 + level 4 = **egovframe.rte.ext.XXX** (기능별 패키지).

#### 공통컴포넌트의 경우

Level 1 + level 2 + level 3 + level 4 = **egovframe.com.ext.XXX** (기능별 패키지).

## 5. 코딩 규칙 - Comment

표준프레임워크 확장기능 개발 시 향후, 표준프레임워크 실행환경에 포함(merge)작업을 위해 아래와 같은 Comment 규칙을 준수하여 작성하도록 한다. .

### 5.1 파일 주석

/* * eGovFrame XXXModule * *	모듈명
* Copyright 2014 MOSPA(Ministry of Security and Public Administration). * * Licensed under the <u>Apache</u> License, Version 2.0 (the "License"); * you may not use this file except in compliance with the License. * You may obtain a copy of the License at * * <u><a href="http://www.apache.org/licenses/LICENSE-2.0">http://www.apache.org/licenses/LICENSE-2.0</a></u> * * Unless required by applicable law or agreed to in writing, software * distributed under the license is distributed on an "AS IS" BASIS, * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. * See the License for the specific language governing permissions and * limitations under the License.	라이선스 기재
* @author 홍길동 /	작성자 정보

### 5.2 클래스 주석

/** * 클래스 * @author 홍길동 * @since 2010.10.17 * @version 1.0 * @see * * <pre> * << 개정이력(Modification Information) >> * *   수정일           수정자           수정내용 *   -----        -----        ----- *   2010.10.17    홍길남   최초 생성 *   2013.12.19   표준프레임워크   공통컴포넌트 추가 적용 (패키지 변경 및 속성 처리) * </pre> */		
---	--	--

### 5.3 메소드 주석

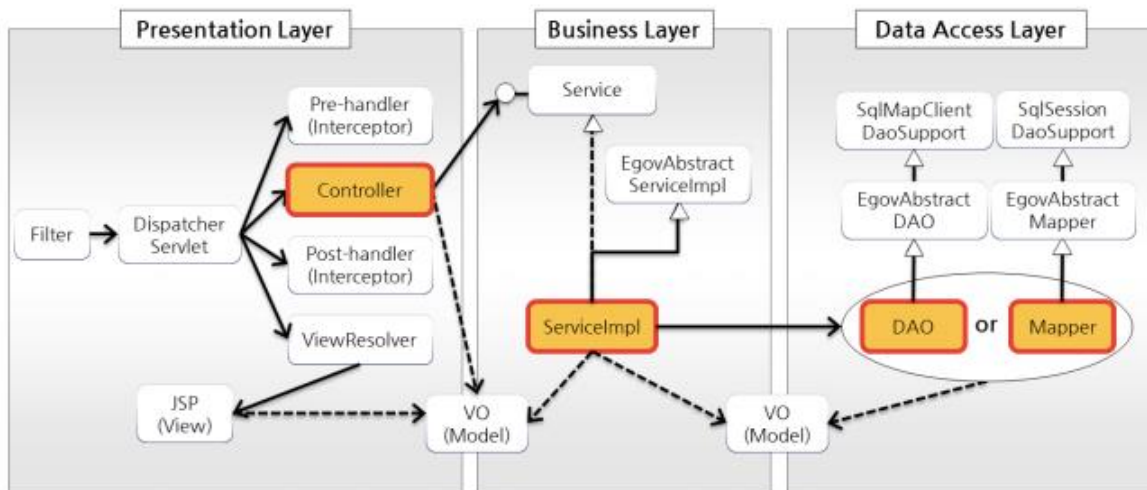
```
/**  
 * 파일 업로드 실행  
 * @param fileVo  
 * @param request  
 * @return  
 */
```

## 6. 코딩규칙 - 아키텍처

표준프레임워크 기반 개발 시 세부 적용 기준에 대해서는 “표준프레임워크 적용가이드 \_v3.0” 및 “표준프레임워크 세부적용기준 3.0” 문서를 참고하는 것을 권장한다.

### 6.1 Annotation 기반 Spring MVC 준수 및 Layered architecture 준수

- Annotation 기반 Spring MVC 준수 : @Controller 및 @RequestMapping을 통한 URL mapping 활용 (View 부분과 model(business logic 및 data) 부분을 controller를 통해 분리)
- Annotation 기반 layered architecture 준수 : 화면처리, 업무처리, 데이터처리에 부분에 대하여 각각 @Controller, @Service, @Repository 활용 (인접 layer간 호출만 가능)



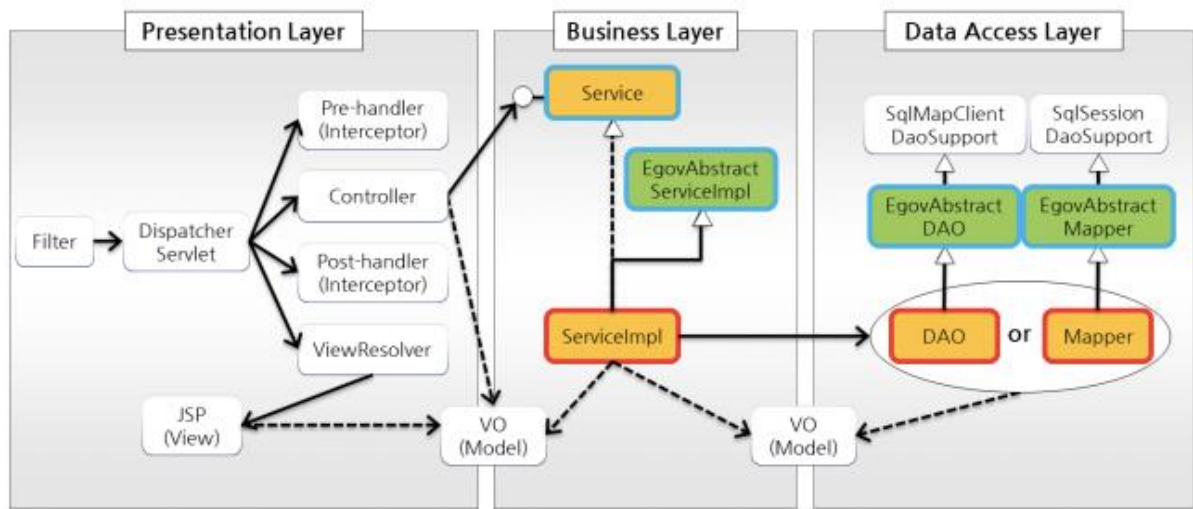
### 6.2 클래스 상속 및 interface 구현 규칙

- 업무처리를 담당하는 서비스 클래스(@Service)는 EgovAbstractServiceImpl(또는 AbstractServiceImpl)을 확장하고 업무에 대한 특정 인터페이스를 구현하여야 함
- 데이터처리를 담당하는 DAO 클래스(@Repository)는 EgovAbstractDAO(iBatis) 또는 EgovMaperDAO(MyBatis)를 상속하여야 함 (Hibernate/JPA를 적용한 경우는 예외이며 자세한 사항은 하단 ‘데이터처리 규칙’ 참조)

\* 표준프레임워크 3.0에서 명명규칙 표준화에 따라 기존 AbstractServiceImpl을

EgovAbstractServiceImpl로 변경함 (2.7의 경우 기존 AbstractServiceImpl 사용)





### 6.3. 데이터 처리 규칙

- Data Acces 서비스 준수 : 데이터처리 부분은 iBatis 활용 (SqlMapClientDaoSupport를 상속한 EgovAbstractDAO 활용) 또는 MyBatis 활용 (SqlSessionDaoSupport를 상속한 EgovAbstractMapper 활용)

※ MyBatis의 경우 Mapper interface 방식으로 사용가능(권장)하며, 이 경우는 interface 상에 @Mapper를 지정하여 사용되어야 함

- ORM 서비스 준수 : 데이터처리 부분은 Hibernate/JPA 적용 (DAO에서 SessionFactory 또는 EntityManagerFactory 설정을 통해 HibernateTemplate/JpaTemplate를 활용하거나, HibernateDaoSupport/JpaDaoSupport를 상속하여 활용)

- Data 서비스 준수 : 데이터 처리 부분은 다양한 persistence store(Big Data, NoSQL 등)를 지원하기 위한 Spring Data 적용 (DAO에서 CrudRepository를 상속하는 interface 방식의 Repository를 활용)

### 6.4. 활용 및 확장 규칙

- 표준프레임워크 실행환경은 적극적으로 활용 되어야 함
- 실행환경의 임의 변경을 금지함
- 표준프레임워크의 프레임워크 기능 확장의 경우는 **egovframe.rte.ext** 를 활용하고, 이외의 일반 시스템의 업무 클래스는 **egovframe.rte** 패키지 내에 정의 될 수 없음.

- 이외에 개발환경, 운영환경 및 공통컴포넌트 부분은 선택적으로 적용가능하며, 임의 변경 및 확장이 가능함
- UI 부분에 RIA(Rich Internet Application)가 적용되는 경우는 UI Adaptor 또는 RESTful 방식을 적용 활용해야 함
- 공통컴포넌트 적용을 검토한 후, 필요기능과의 차이를 분석하고 해당 업무에 최적화하여 활용