

Agora 포딩 메뉴얼

FrontEnd

Version

- Vue 3.1.5
- Vue Router 4.0.0-beta 1
- Vuex 4.0.0-beta 4
- Babel 7.10.4
- Webpack 4.43.0

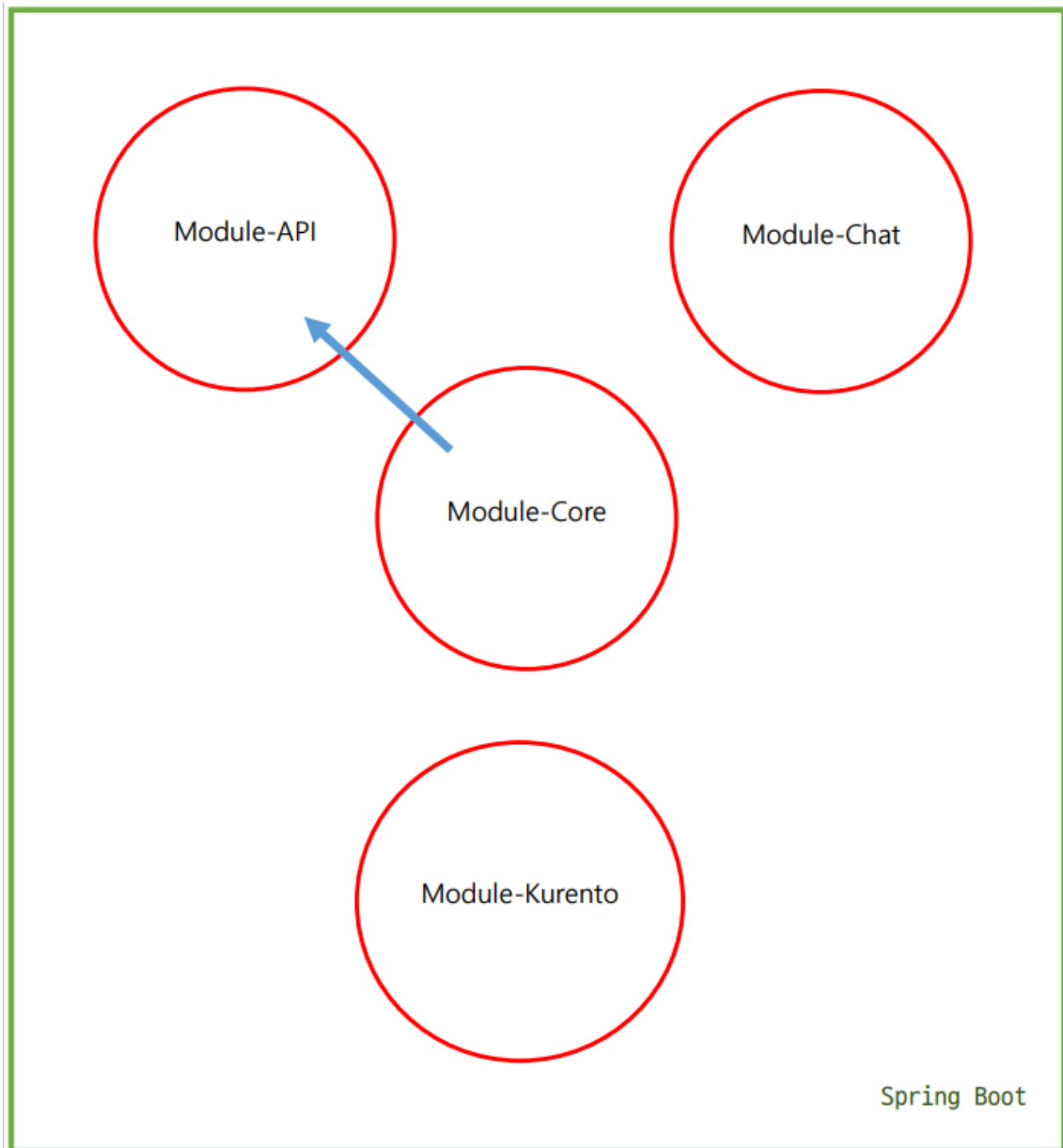
```
## Project setup
``
npm install --legacy-peer-deps
``

### Compiles and hot-reloads for development
``
npm run dev
``

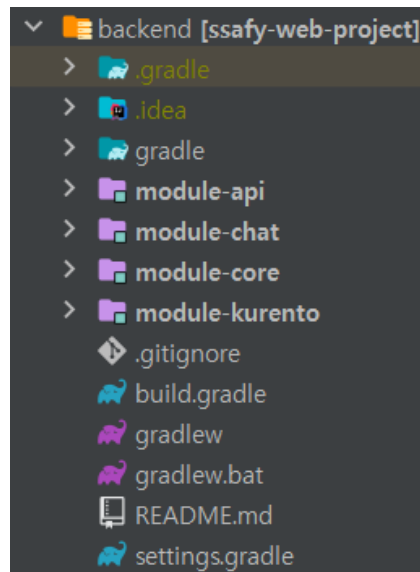
### Compiles and minifies for production
``
npm run build
``
```

Backend Module Structure

멀티 모듈 프로젝트 구조로 이루어져 있으며, Module-API, Module-Chat, Module-Core, Module-Kurento 모듈이 존재합니다.



여기서 Module Core는 엔티티나 중복되는 공통 코드들이 모여져 있는 모듈로써 Module-API가 Module-Core를 의존하고 있습니다.



루트 프로젝트

settings.gradle

```
rootProject.name = 'ssafy-web-project'

include 'module-api'
include 'module-core'
include 'module-kurento'
include 'module-chat'
```

Build.gradle

Module-API

```
dependencies {

    implementation project(':module-core')

    implementation 'org.springframework.boot:spring-boot-starter-security'
    implementation 'org.springframework.boot:spring-boot-starter-web'
    testImplementation 'org.springframework.boot:spring-boot-starter-test'
    testImplementation 'org.springframework.security:spring-security-test'

    // Swagger
    implementation("io.springfox:springfox-swagger2:3.0.0")
    implementation("io.springfox:springfox-swagger-ui:3.0.0")
    implementation("io.springfox:springfox-data-rest:3.0.0")
    implementation("io.springfox:springfox-bean-validators:3.0.0")
    implementation("io.springfox:springfox-boot-starter:3.0.0")

    // JWT
    implementation('com.auth0:java-jwt:4.2.1')
```

```

implementation group: 'io.jsonwebtoken', name: 'jjwt-api', version: '0.11.5'
runtimeOnly group: 'io.jsonwebtoken', name: 'jjwt-jackson', version: '0.11.5'
runtimeOnly group: 'io.jsonwebtoken', name: 'jjwt-impl', version: '0.11.5'

//email 인증
implementation group: 'org.springframework.boot', name: 'spring-boot-starter-mail', version: '2.5.4'

// 증명서 발급 관련
implementation("org.springframework.boot:spring-boot-starter-thymeleaf")
implementation group: 'org.xhtmlrenderer', name: 'flying-saucer-pdf-openpdf', version: '9.1.22'
implementation group: 'commons-io', name: 'commons-io', version: '2.11.0'
implementation 'com.itextpdf:html2pdf:3.0.0'
}

```

Module-core

```

buildscript {
    ext {
        querydslVer = '5.0.0'
    }
}
jar {
    enabled = true
}
bootJar {
    enabled = false
}
version 'unspecified'

repositories {
    mavenCentral()
}

dependencies {

    api 'org.springframework.boot:spring-boot-starter-data-jpa'
    api ('mysql:mysql-connector-java:8.0.31')

    //reids
    implementation group: 'org.springframework.boot', name: 'spring-boot-starter-data-redis', version: '2.7.5'

    //queryDsl
    implementation("com.querydsl:querydsl-jpa:${querydslVer}")
    implementation("com.querydsl:querydsl-apt:${querydslVer}")

    annotationProcessor("com.querydsl:querydsl-apt:${querydslVer}:jpa")
    testImplementation 'org.springframework.boot:spring-boot-starter-test'

    implementation ("javax.annotation:javax.annotation-api:1.2")

    testImplementation 'org.junit.jupiter:junit-jupiter-api:5.7.0'
    testRuntimeOnly 'org.junit.jupiter:junit-jupiter-engine:5.7.0'

    annotationProcessor "jakarta.persistence:jakarta.persistence-api"
    annotationProcessor "jakarta.annotation:jakarta.annotation-api"
}

```

Module-kurento

```

buildscript{
    ext {
        springBootVer = '2.4.5'
        springDependencyMgmtVer = '1.0.11'
        springLoadedVer = '1.2.8'
        nodePluginVer = '1.3.1'
    }
}

plugins {
    id 'java'
    id 'idea'
    id 'org.springframework.boot' version "${springBootVer}"
}

apply plugin: 'io.spring.dependency-management'
apply plugin: 'eclipse'

subprojects {
    group = 'com.ssafy'
    version = '0.0.1-SNAPSHOT'

    apply plugin: 'java-library'
    apply plugin: 'org.springframework.boot'
    apply plugin: 'io.spring.dependency-management'

    sourceCompatibility = '11'

    configurations {
        compileOnly {
            extendsFrom annotationProcessor
        }
    }
    repositories {
        mavenCentral()
    }

    dependencies {
        implementation("org.springframework.boot:spring-boot-starter-websocket")
        implementation("org.springframework.boot:spring-boot-starter-actuator")
        implementation("org.springframework.plugin:spring-plugin-core:2.0.0.RELEASE")

        developmentOnly("org.springframework.boot:spring-boot-devtools")
        annotationProcessor("org.springframework.boot:spring-boot-configuration-processor")

        implementation('commons-io:commons-io:2.6')
        implementation("org.apache.commons:commons-collections4:4.4")
        implementation("org.apache.commons:commons-lang3:3.9")

        implementation("com.squareup.retrofit2:retrofit:2.7.1")
        implementation("com.squareup.retrofit2:converter-jackson:2.7.1")
        implementation("com.squareup.okhttp3:logging-interceptor:3.9.0")

        implementation("com.google.guava:guava:29.0-jre")
        annotationProcessor("com.google.guava:guava:29.0-jre")

        testImplementation("com.jayway.jsonpath:json-path:2.4.0")

        annotationProcessor("javax.annotation:javax.annotation-api:1.3.2")

        implementation("org.projectlombok:lombok:1.18.20")
        annotationProcessor("org.projectlombok:lombok:1.18.20")
    }

    tasks.named('test') {
        useJUnitPlatform()
    }
}

```

Module-Chat

```
dependencies {
    implementation("org.springframework.boot:spring-boot-starter-web")
    implementation("org.springframework.boot:spring-boot-starter-websocket")
    implementation("org.springframework.kafka:spring-kafka")
    compileOnly("org.projectlombok:lombok")
    annotationProcessor("org.projectlombok:lombok")
    testImplementation("org.springframework.boot:spring-boot-starter-test")
    testImplementation("org.springframework.kafka:spring-kafka-test")
}

tasks.named('test') {
    useJUnitPlatform()
}
```

버전

IDE

IntelliJ IDEA 2022.3.1 (Ultimate Edition)

JVM

Zulu openjdk 11.0.1

Gradle

Gradle 6.8

Backend(DockerCompose, Jenkinipeline)

Module-API

```
version: '2'
services:
  back-module-api:
    build:
      context: .
    container_name: agora-back-module-api
    volumes:
```

```
- /home/docker/agora/volume-data/files:/usr/local/share
ports:
- "8082:8082"
```

Module-chat

```
version: '2'
services:
  back-module-chat:
    build:
      context: .
    container_name: agora-back-module-chat
    ports:
      - "8084:8084"
```

Module-Kurento

```
version: '2'
services:
  back-module-kurento:
    build:
      context: .
    container_name: agora-back-module-kurento
    ports:
      - "8083:8083"
    environment:
      JAVA_OPTS: -Dkms.url=ws://i8c205.p.ssafy.io:8888/kurento
```

Jenkins pipeline

```
pipeline
{
  agent any
  tools {
    gradle 'gradle 6.8'
  }
  environment {
    PROJECT = 'agora'
    APP_API = 'module-api'
    APP_CHAT = 'module-chat'
    APP_KURENTO = 'module-kurento'
  }
  stages {
    stage('environment') {
      when {
        changeset "env-config/**/*"
      }
    }
  }
}
```

```

    steps {
        echo 'Environment Settings Start'
        sh 'docker-compose -f env-config/docker-compose-env.yml down'
        sh 'docker-compose -f env-config/docker-compose-env.yml up -d'
        echo 'Environment Settings End'
    }
}
// stage('enviroment-nginx'){
// when {
//     changeset "env-config/nginx/**/*"
// }
// steps {
//     echo 'Nginx Settings Start'
//     sh 'docker-compose -f env-config/nginx/docker-compose.yml up -d'
//     echo 'Nginx Settings End'
// }
// }
stage('build-module-api') {
    when {
        anyOf {
            changeset "backend/module-core/**/*"
            changeset "backend/module-api/**/*"
        }
    }
    steps {
        echo 'Build Start "${APP_API}"'
        sh 'chmod +x backend/gradlew'
        sh '''
            backend/gradlew -p backend/${APP_API} build -x test
            '''
        echo 'Build End "${APP_API}"'
    }
}
stage('build-module-chat') {
    when {
        changeset "backend/module-chat/**/*"
    }
    steps {
        echo 'Build Start "${APP_CHAT}"'
        sh 'chmod +x backend/gradlew'
        sh '''
            backend/gradlew -p backend/${APP_CHAT} build -x test
            '''
        echo 'Build End "${APP_CHAT}"'
    }
}
stage('build-module-kurento') {
    when {
        changeset "backend/module-kurento/**/*"
    }
    steps {
        echo 'Build Start "${APP_KURENTO}"'
        sh 'chmod +x backend/gradlew'
        sh '''
            backend/gradlew -p backend/${APP_KURENTO} build -x test
            '''

        echo 'Build End "${APP_KURENTO}"'
    }
}
stage('build-front') {
    when {
        changeset "frontend_tem/**/*"
    }
    steps {
        echo 'Build Start Front App'
        sh 'docker build -t app-vue frontend_tem/. --no-cache'
        echo 'Build End Front App'
    }
}
stage('deploy-module-api') {

```



```

    when {
      anyOf {
        changeset "backend/module-core/**/*"
        changeset "backend/module-api/**/*"
      }
    }
  }
  steps {
    echo 'Deploy Start "${APP_API}"'
    sh 'docker-compose -f backend/${APP_API}/docker-compose.yml build --no-cache'
    sh 'docker-compose -f backend/${APP_API}/docker-compose.yml up -d'
    echo 'Deploy End "${APP_API}"'
  }
}
stage('deploy-module-chat') {
  when {
    changeset "backend/module-chat/**/*"
  }
  steps {
    echo 'Deploy Start "${APP_CHAT}"'
    sh 'docker-compose -f backend/${APP_CHAT}/docker-compose.yml build --no-cache'
    sh 'docker-compose -f backend/${APP_CHAT}/docker-compose.yml up -d'
    echo 'Deploy End "${APP_CHAT}"'
  }
}
stage('deploy-module-kurento') {
  when {
    changeset "backend/module-kurento/**/*"
  }
  steps {
    echo 'Deploy Start "${APP_KURENTO}"'
    sh 'docker-compose -f backend/${APP_KURENTO}/docker-compose.yml down'
    sh 'docker-compose -f backend/${APP_KURENTO}/docker-compose.yml up -d'
    echo 'Deploy End "${APP_KURENTO}"'
  }
}
stage('build-deploy-front') {
  when {
    changeset "frontend_tem/**/*"
  }
  steps {
    echo 'Deploy Start Front App'
    sh '''
      docker stop front-app
      docker rm front-app
      docker run -d -v /usr/local/share/ssl:/usr/local/share -p 80:8083 --name front-app app-vue
    '''
    echo 'Deploy End Front App'
  }
}
}
}
}
}

```