**[외부 서비스 정보]**

구미캠퍼스 1반 5팀 Fitmate

**[gitlab 소스 클론 이후 빌드 및 배포할 수 있도록 정리한 문서]**

* 기본 운영 체제 – Windows 10
* 기본 IDE: IntelliJ IDEA, VS Code
* 프론트엔드
* Vue2.x
* Vuex, Vue router, Axios
* CSS, SCSS
* openvidu
* 백엔드
* Spring Boot
* JPA
* Jwt
* AWS S3
* KMS(Kurento Media Server)
* Dockers
* Jenkins

**[외부 서비스 정보]**

**1. Kakao – 소셜 인증 및 카카오 페이**

-> Kakao developers 을 통해 카카오의 다양한 서비스들을 사용할 수 있습니다. 저희 Fitmate는 이 중 카카오 로그인과 카카오 페이를 사용하였습니다.

[가입 방법]

자신의 카카오로 로그인 후 ‘내 애플리케이션’ -> ‘애플리케이션 추가하기’를 통해 자신의 애플리케이션 만들기

서비스를 사용하기 위해서는 플랫폼 등록이 필수이다 -> 플랫폼 중 Web에 등록

**1-1. 카카오 로그인(소셜 인증)**

[카카오 간편 로그인 | Kakao Developers 문서](https://developers.kakao.com/docs/latest/ko/kakaologin/common)

[활용 방법]

- 카카오에서 제공하는 앱 키 중 JavaScript 키를 활용한다.

- 카카오 로그인을 통해 활용이 필요한 정보 동의 상태를 설정한다.

- 카카오 로그인 문서를 따라 진행한다(카카오 로그인은 OAuth 2.0 기반 공개 API(Open API)로 제공되는 표준 인증 방법, Frontend에서 진행).

로그인: Kakao.Auth.authorize(PARAMETER);

토큰 할당하기: Kakao.Auth.setAccessToken(ACCESS\_TOKEN);

**1-2. 카카오 페이(단건 결제)**

[단건 결제 시스템 | Kakao Developers 문서](https://developers.kakao.com/docs/latest/ko/kakaopay/single-payment)

[활용 방법]

- 카카오에서 제공하는 앱 키 중 Admin 키를 활용한다.

- 카카오 페이는 Backend에서 진행한다(REST API)

- 실제 결제는 사업자 정보가 필요하므로 테스트 버전으로 진행한다(TC0ONETIME)



결제 준비: POST /v1/payment/ready

결제 요청: POST /v1/payment/approve

**2. 구글 메일**

-> 예약을 한 고객에게 상담 인증키를 전달하기 위해 사용합니다.

[가입 방법]

구글에서 fitmate 계정을 만들어줍니다. Google의 Gmail SMTP 서버는 Gmail 계정이 있는 사람이라면 누구나 이메일을 보내는 데 사용할 수 있는 무료 SMTP 서비스이다.

[활용 방법]

백엔드에서 Gmail SMTP 서버를 사용하려면 보내는 이메일에 다음 설정이 필요합니다.

- build.gradle

dependencies에서 implementation 'org.springframework.boot:spring-boot-starter-mail' 설정 필요

보내는 메일(SMTP) 서버: smtp.gmail.com

인증 사용: 예

보안 연결 사용: 예 (메일 클라이언트/웹사이트 SMTP 플러그인에 따라 TLS 또는 SSL)

사용자 이름: Gmail 계정 (예: user@gmail.com)

비밀번호: Gmail 비밀번호

포트: 465 (SSL 필요) 또는 587 (TLS 필요)

- application.yml

mail: host: smtp.gmail.com port: 587 username: fitmate2022@gmail.com password: 비밀번호 properties: mail: smtp: auth: true starttls: enable: true

**3. KMS (Kurento Media Server)**

[Kurento Tutorials](https://doc-kurento.readthedocs.io/en/stable/tutorials/java/tutorial-helloworld.html)

화상 상담을 위한 WebRTC를 사용하였습니다.

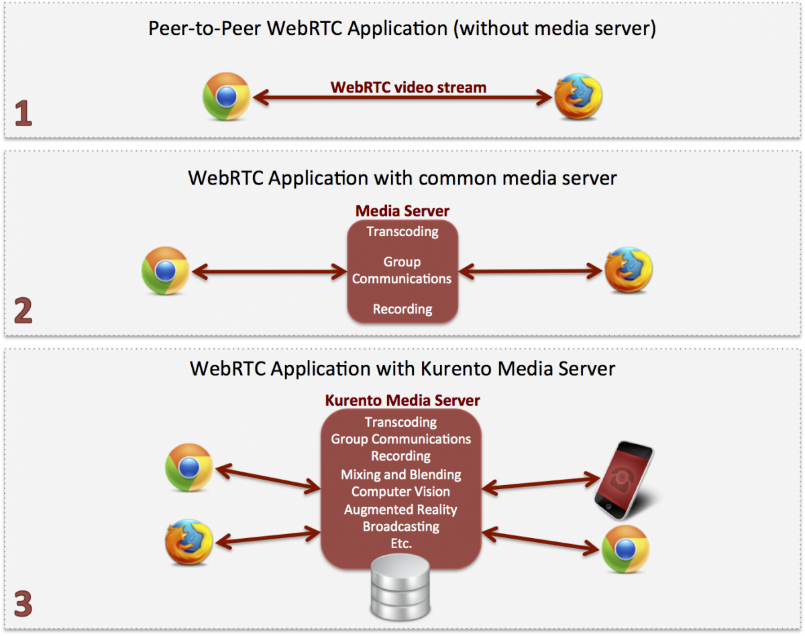
도커 또는 Ubuntu 서버가 설치되어 있는 상태에서 다음과 같은 명령어를 사용한다.

docker pull kurento/kurento-media-server:latest

docker run -d --name kms --network host \ kurento/kurento-media-server:latest

openvidu-server을 클론 받아 테스트할 수 있다.

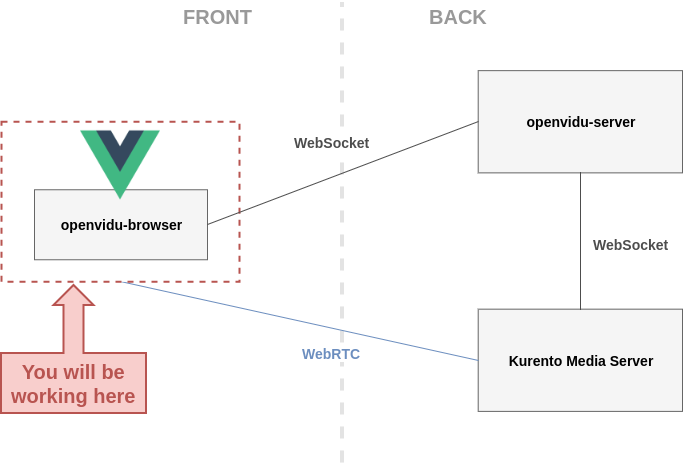
- 클라이언트 – 서버 – 쿠렌토 서버 연동 과정



**4. Openvidu**

[Openvidu Tutorials](https://docs.openvidu.io/en/stable/tutorials/openvidu-insecure-vue/)

Openvidu 역시 화상 상담을 위해 사용되었습니다.



- openvidu-browser : Vue 앱용 NPM 패키지. 고객과 직접 화상 통화를 관리할 수 있습니다.

- openvidu-server : Kurento Media Server를 제어하는 ​​Java 애플리케이션

- Kurento Media Server : 미디어 흐름 전송의 저수준 작업을 처리하는 서버

openvidu-server를 클론한다.

명령어: git clone https://github.com/OpenVidu/openvidu.git

해당 튜토리얼 폴더에서 openvidu-tutorials\openvidu-insecure-vue 에서 npm install / npm run serve를 차례로 입력

**5. docker**

[Doker Tutorial](https://docker-curriculum.com/)

**6. Jenkins**

[Jenkins Tutorial](https://www.tutorialspoint.com/jenkins/index.htm)

**7. Amazon S3**

[Amazon S3 Tutorial](https://aws.amazon.com/ko/s3/getting-started/?trkCampaign=acq_paid_search_brand&sc_channel=PS&sc_campaign=acquisition_KR&sc_publisher=Google&sc_category=Storage&sc_country=KR&sc_geo=APAC&sc_outcome=acq&sc_detail=amazon%20s3&sc_content=%7bad%20group%7d&sc_matchtype=e&sc_segment=489215169067&sc_medium=ACQ-P|PS-GO|Brand|Desktop|SU|Storage|S3|KR|EN|Sitelink&s_kwcid=AL!4422!3!489215169067!e!!g!!amazon%20s3&ef_id=CjwKCAiAgbiQBhAHEiwAuQ6Bkh3oWfePsCjxtSIrmHmuf972i4kFbItJKM6oD0UG_s_nWVc2lIFQrxoCSwQQAvD_BwE:G:s&s_kwcid=AL!4422!3!489215169067!e!!g!!amazon%20s3)