



# 안녕하세요! “베어 그릴스” 개발자, 조용현입니다 🙌

## 😊 About Me

### 👤 Introduction

- 안녕하세요! Back-end Developer를 꿈꾸며 도전을 멈추지 않는 조용현입니다 😊

### 🔍 Final Goal

- FE 생태계를 이해하고 있는 BE 개발자가 되고 싶습니다.
- “베어 그릴스”처럼, 다음과 같은 모습으로 성장하고 싶습니다.
  1. 빠르게 변화하는 IT 생태계에서 고객의 필요를 적극 반영하고, 적용하는 개발자
  2. 새로운 언어를 배울 때 빠르게 적응하며 더 나아가 나만의 것으로 만드는 개발자
  3. 개발하면서 어려움에 부딪치더라도 끈기와 열정을 가지고 포기하지 않는 개발자
- “팀원의 문제는 곧 내 문제다”라는 책임감을 가지고 프로젝트에 임하는 개발자가 되고 싶습니다.

니다.

## Contact & Channel

- Email : tofan@naver.com
- Github : <https://github.com/tofan0412>

## Skills

### FE

vue.js (2.x) vue.js (3.x) vue router vuex JavaScript(ES6+)

HTML CSS JSP JSTL Bootstrap element UI primeVue Vuetify

### BE

Django springBoot Oracle SQL Developer MySQL spring data JPA swagger

### 협업 관련

Figma.io Jira Redmine Gitlab Git Github Draw.io Notion

## Projects

저는 8개월 간의 국비 교육과 1년 간의 삼성 청년 소프트웨어 아카데미(SSAFY)를 통해 4번의 프로젝트를 경험했습니다. 매 프로젝트를 수행할 때마다 팀장 역할을 맡으며 프로젝트에 대한 강한 책임감과 목표의식을 가지고 임했으며, 덕분에 프로젝트를 완료할 때마다 한 단계 성장할 수 있었습니다.

### 삼성 청년 소프트웨어 아카데미(SSAFY) 자율 프로젝트 : WebRTC 기반 AR 소개팅 서비스(Mokkozi)

기간 : 2021.10.11 ~ 2021.11.19

인원 : 5명

첨부 : [Github](#), [발표자료](#), [포팅 메뉴얼](#), [구현 페이지](#)

활용 기술 : [openvidu](#) [nginx](#) [webRTC](#) [JWT](#) [Spring Security](#) [AWS S3](#) [vue.js \(2.x\)](#)  
[vuetify](#) [springboot](#) [swagger](#) [Gradle](#)

담당 역할 및 기여도 : 팀장 / 프론트엔드 30% / 백엔드 50%

### 구현 사항

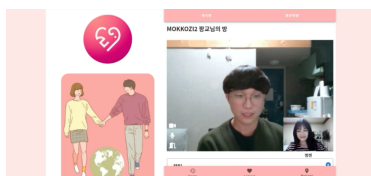
- AWS EC2 서버 내 OpenVidu build
- Spring Security & JWT를 활용한 인증 기능 구현
- AWS S3를 통한 multiple image upload 구현
- vue.js(2.x)와 vuetify(2.x)를 활용한 로그인 및 회원가입 페이지 구현 (FE)
- 로그인 및 회원가입 BE 구현
- 이미지 업로드 기능 BE 구현

### 프로젝트를 수행하며 성장한 점 & 보완해야 할 점

- **Jira를 통한 프로젝트 일정 관리의 중요성**을 깨달았습니다. 이전 프로젝트가 7주였던 반면, 자율 프로젝트의 경우 6주 동안 진행되었기 때문에 꾸준한 Jira 이슈 업데이트를 통해 지속적으로 팀원들과 의사소통하였습니다.
  - Jira를 하면서 팀원들이 작성한 이슈, 스토리 등을 분류해야 할 필요성을 느꼈으며, JQL에 대한 학습 필요성을 느꼈습니다.
- 이전 프로젝트(blanket)에서 경험했던 S3의 back-end, front-end 코드를 직접 작성하였으며, 제 AWS 계정을 통해 직접 관리했습니다.
- AWS S3 스토리지에 업로드하는 것은 가능했지만, S3 버킷 내 등록된 파일을 springboot back-end 코드를 통해 수정하거나 삭제하는 것을 구현할 수 없었습니다. 이에 대해 보완한다면 더 좋을 것 같습니다.
- SSAFY에서 진행한 첫 프로젝트 때는, openVidu를 다른 팀원이 배포해주었지만 이번에는 제가 직접 배포해야 했습니다. 이 과정에서 많은 문제가 있었으며 소요되는 시간도 길었습니다. 이 과정에서 WebRTC의 개념과 STUN, TURN, Kurento Media Server를 학습할 수 있었습니다.
- 작성한 게시글의 이미지 수정 기능 구현이 다소 어려웠습니다. 이전에 업로드한 이미지의 수정과 동시에 새로운 이미지의 업로드가 가능해야 했으며, DB와 S3에서도 마찬가지로 신규 이미지의 업로드와 삭제를 적용해야 했습니다. 이를 위해 front-end에서 array를 통해 업로드할 이미지와 삭제할 이미지의 PK를 저장하였으며, 최종 수정 버튼을 누르면 API 호출을 통해 back-end에서 신규 이

이미지 업로드(S3 업로드 후 DB에 URL 저장) → 이미지 삭제(S3에서 기존에 업로드한 파일 삭제 후 DB column 삭제)의 순서로 동작할 수 있도록 기능을 구현했습니다. (다만 S3 스토리지 내 파일 삭제는 기술적 문제로 구현할 수 없어 Error가 발생하였고, catch-throw를 통해 예외 처리하였습니다.)

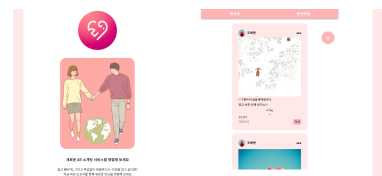
- FE & BE 모두를 개발한 경험을 얻었습니다. 로그인, 회원가입 기능을 맡아 BE, FE 코드를 작성하였으며 제가 원했던 full-stack 개발을 경험할 수 있었습니다. 이런 Full-Stack의 경험은 이후 수행할 다양한 프로젝트에서 Front-end 담당 엔지니어와의 원활한 의사 소통을 가능하게 할 것입니다.
- Back-end 코드를 작성하면서 팀원마다 Null 처리하는 방식이 모두 달랐습니다. (Optional 객체 이용, orElseThrow를 통한 예외처리) 관련된 예제를 살펴보고 Null 처리에 대한 다양한 방법을 학습할 수 있었습니다.
- 이전 프로젝트에서 다른 팀원이 적용했던 JWT & Spring Security를 직접 구현함으로써, 사용자 인증 과정을 공부할 수 있었습니다. 과정 중 Front-end에서 넘어오는 Request를 Back-end Server에서 어떤 순서로 처리하는지 공부할 수 있었습니다. (Filter, spring MVC)
- AWS EC2에서 실제로 서비스를 배포한 경험을 얻었습니다. nginx를 이용하여 FE, BE, Openvidu 서버에 대한 포트 포워딩을 설정했습니다.



openVidu를 활용, 1:1 화상 회의를 구현한 페이지. AWS EC2 환경에서 openVidu를 설치 및 배포하였다.



회원가입 페이지. 이메일과 닉네임을 입력할 때 clearTimeout, setTimeout을 이용하여 실시간으로 중복검사를 진행하며 다음 주소 API를 활용하여 주소지를 쉽게 입력할 수 있다. 또한 S3를 통해 사용자 프로필 이미지를 업로드할 수 있으며 미리보기 기능을 제공한다.



커뮤니티 게시판 구현. S3를 통해 업로드한 이미지를 확인할 수 있으며 수정을 통해 이미지를 추가로 업로드하거나 삭제할 수 있다. 유저 팔로우, 게시물 좋아요, 댓글 작성 등의 기능도 함께 구현하였다.

# 삼성 청년 소프트웨어 아카데미(SSAFY) 특화 프로젝트 : 블록체인 기반 자격증명 서비스(Blanket)

기간 : 2021.08.30 ~ 2021.10.08

인원 : 5명

첨부 : [Github](#), [시연 시나리오](#), [포팅 메뉴얼](#), [시스템 아키텍처](#)

활용 기술 : [ethereum Network](#) [web3.js](#) [infra](#) [MetaMask](#) [springboot](#) [swagger](#)  
[javascript](#) [vue.js\(3.x\)](#) [primevue](#) [aws S3](#)  
[공공데이터 API](#) [gradle](#)

담당 역할 및 기여도 : 팀장 / 프론트엔드 60% / 백엔드 30%

## 구현 사항

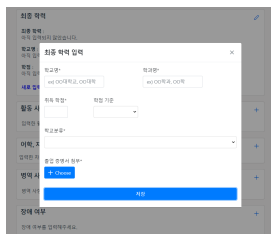
- web3.js, infra를 이용한 block data 저장 기능 구현
- Amazon S3 이미지 업로드 Front-end 코드 작성
- 공공데이터 API를 이용한 대학교 이름 검색 기능 구현
- async-await을 이용한 Javascript 비동기 처리 함수 구현
- 최종 학력, 활동 사항, 어학 및 자격증 Front-end 페이지 구현 및 axios를 이용한 Back-end Server와 통신 기능 구현

## 프로젝트를 수행하며 성장한 점 & 보완해야 할 점

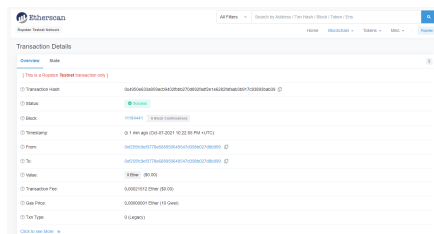
- 처음 블록체인 기술을 접했지만, 레퍼런스 문서를 기반으로 직접 번역해가며 학습했고, 이를 통해 블록체인 생태계를 이해하는데 필요한 기본기를 다질 수 있었습니다.
  - 구체적으로 WSL2를 이용한 이더리움 네트워크 노드의 생성과 채굴, 트랜잭션 전송을 학습했습니다.
  - Infra를 통해 온라인에 존재하는 이더리움 네트워크 노드를 빌려 복잡한 설정 단계 없이도 채굴과 트랜잭션을 전송할 수 있었습니다.
  - Infra에 등록된 Node를 Chrome Extension인 MetaMask를 통해 등록하여 계좌 잔고 확인, 송금 과정을 실습할 수 있었습니다.
  - 최종적으로 web3.js를 통해 infra node의 private key를 활용, 기존에 생성한 관리자 계정에 javascript 코드를 통해 트랜잭션을 전송하고 노드 내 계좌를 생성할 수 있었습니다.
- vue.js(3.x)를 front-end framework로 사용하면서, vue.js의 3.x 버전에 해당하는 다양한 CSS framework를 경험할 수 있었습니다. (element

UI, primeVue)

- `async - await`을 이용한 비동기 처리를 처음으로 학습했습니다.
  - `Vue.js`의 `script`를 작성하면서 `back-end API` 처리가 완료되기 이전에 데이터를 `vuex`에 저장하는 문제점이 있었고, 이 때문에 빈 데이터가 저장되었습니다. 이를 해결하기 위해 별도의 `.js` 파일을 작성하고 함수 형태로 모듈화하여 작성하였으며, 함수에 `async - await`을 적용하였습니다.
  - 이를 통해 `API` 호출이 종료된 후에 전송된 데이터를 온전하게 저장할 수 있었습니다.
  - 이와 더불어 또 다른 비동기 처리 방식인 `Promise` 객체에 대해 공부할 수 있었습니다.
- `Front-end`와 데이터를 주고 받으면서 필요한 개념인 `DTO`, `entity`, `VO`에 대한 개념과 차이를 명확하게 학습할 수 있었습니다.
- 이전 프로젝트에서 `Maven`을 통해 프로젝트를 관리하고 배포했다면, 이번 프로젝트부터 `Gradle`을 적용하여 `Maven`보다 좀 더 쉽게 `dependency` 관리, 프로젝트 배포를 수행할 수 있었습니다.



입사 지원서 작성시 필요한 기본 정보와 이를 증명할 수 있는 이미지 파일을 함께 저장한다. 관리자는 사용자가 작성한 인적 사항을 확인하며 필요한 경우 최종적으로 블록체인 형태로 해당 인적사항을 기록한다.



관리자는 사용자가 작성한 인적 사항을 검증하며, 검증이 완료되면 이더리움 네트워크에 해당 이미지 파일을 해시값으로 변경하여 올라가게 된다.



신상정보를 입력하는 페이지. 입사 지원서 공통적으로 필요한 요소들을 등록할 수 있다. (인턴 경력, 어학 및 자격증, 최종 학력 사항등)

# 삼성 청년 소프트웨어 아카데미(SSAFY) 공통 프로젝트 : WebRTC 기반 소개팅 서비스(Cow & Dog)

기간 : 2021.07.12 ~ 2021.08.20

인원 : 5명

첨부 : Github

활용 기술 : `openVidu` `springboot` `vue.js (3.x)` `vue-router` `vuex` `Maven`

담당 역할 및 기여도 : 팀원 / 프론트엔드 30% / 백엔드 20%

## 구현 사항

- Appeal 게시판 Front-end 페이지 작성
- Appeal 게시판 게시글 작성, 수정, 삭제 Back-end 코드 작성
- Pagination을 이용한 게시글 페이지징 처리
- 게시글 내 댓글 작성 및 수정, 삭제

## 프로젝트를 수행하며 성장한 점 & 보완해야 할 점

- 처음으로 SSAFY 내 컴퓨터 공학 전공자들과 함께 한 프로젝트였습니다.
- SSAFY 과정 중 1학기는 Django를 기반으로 웹 서비스를 개발했기 때문에, springboot를 통해 back-end 코드를 작성하는 능력이 필요했습니다. 이를 위해 inflearn 강의를 통해 기초 지식을 학습했으며 팀원이 작성한 예제 코드를 클론 코딩하여 필요한 back-end 코드를 작성할 수 있었습니다.
- 처음으로 vue.js (3.x)를 적용했습니다. vue.js (2.x)와의 차이점인 Composition API, Life Cycle hooks, watch와 computed의 변경점 등을 vue.js 공식 레퍼런스 문서를 통해 참고하여 학습했습니다.
- vue의 중앙 스토리지 역할을 하는 vuex의 데이터 접근 및 저장 방법 (action, mutation, getters)을 학습했으며, 각 단계별로 별도의 .js 파일을 작성하여 관리했습니다. (action.js, mutation.js, getters.js)
- 이전에 수행한 Spring 기반 프로젝트에서는 MyBatis를 이용하여 DB에 접근했으나, 이번 프로젝트부터 JPA를 활용했습니다. 이 덕분에 복잡한 SQL을 직접 작성하지 않더라도 DB에서 원하는 데이터를 가져올 수 있었습니다.
- JPA를 통해 게시글과 댓글 목록을 불러오는 기능을 구현하던 중, N+1 문제를 겪게 되었습니다. 이는 양방향 참조를 기본으로 채택하고 있는 JPA의 특성상 발생하는 문제로, 게시글과 댓글이 서로를 무한 참조하기 때문에 발생하는 오류였습니다. 이를 위해 @JsonManagedReference, @JsonBackReference 어노

테이션을 각각 게시글의 comments column과 댓글의 article column에 추가하여 해결할 수 있었습니다.

- 유저 Follow, 게시글 Like 기능을 구현하기 위해 필요한 DB 테이블의 관계 설정과 JPA를 통한 작성법을 학습할 수 있었습니다.
- 이전 프로젝트까지 ES5를 기반으로 작성했기 때문에 javascript 변수를 var로 선언하는 경우가 많았습니다. 하지만 var 타입의 경우 호이스팅 문제가 발생할 수 있습니다. 이번 프로젝트부터는 ES6를 기반으로 프로젝트를 수행했기 때문에 이를 해결할 수 있는 const, let 타입의 변수 선언을 통해 호이스팅 문제를 해결할 수 있었습니다.

## 전자정부프레임워크 기반 협업관리툴 개발(Ants)

기간 : 2020.10 ~ 2021.01

인원 : 5명

첨부 : [Github](#)

활용 기술 : [spring](#) [jQuery](#) [javascript](#) [전자정부 표준 프레임워크](#) [myBatis](#) [SQL](#) [Maven](#)  
[tomcat](#)

담당 역할 및 기여도 : 팀장 / 프론트엔드 20% / 백엔드 30%

### 구현 사항

- 일대일 및 일대다 채팅방 기능 구현
- 프로젝트 생성 시 팀원 초대 기능 구현
- 프로젝트 현재 상태(대기, 승인, 완료 등)에 따른 프로젝트 메인 페이지 작성

### 프로젝트를 수행하며 성장한 점 & 보완해야 할 점

- SockJS와 Javascript를 이용하여 채팅 기능을 구현했습니다. 이전에 배운 적이 없어 구현하기 어려웠지만, 관련 문서를 참조하여 최종적으로 구현할 수 있었습니다.
  - 현재 채팅방에 접속하지 않은 다른 유저에게도 메시지가 전달되는 문제가 있었습니다. 이를 해결하기 위해 Back-end 내 메시지 전송을 담당하는 Filter(echoHandler.java)에서 메시지가 출력되어야 할 채팅방과 발신인 아이디를 함께 보내, front-end에서 출력 여부를 결정하게끔 구현했습니다.



- 위 문제는 STOMP를 활용하면 간단하게 해결할 수 있으며, 이후의 토이 프로젝트를 통해 SockJS와 STOMP를 활용한 채팅방을 구현할 계획입니다.
- setTimeout과 clearTimeout을 통해 사용자의 입력이 끝나면 검색어에 해당하는 팀원 목록을 불러오는 javascript 코드를 작성했습니다.

## ☀ 프로젝트를 수행하면서 느낀 나만의 강점

- 프로젝트에 대한 강한 책임감
  - SSAFY 교육 과정에서 진행하는 3번의 프로젝트 중 특화, 자율 프로젝트에서 팀장 역할을 맡아 책임감을 가지고 프로젝트를 주도하였습니다.
- FE와 BE를 구분하지 않고 개발한 경험
  - 국비 교육 때 배운 Java를 바탕으로 SSAFY 교육과정동안 전공자들과 함께 BE를 작성할 수 있었습니다.
- 새로운 기술에 대한 적극적인 자세
  - 국비 교육 때 배운 MyBatis, JSP 기반 개발 프로세스에서 벗어나 별도의 FE framework, ORM 기반 DB 개발 환경을 배우고 이를 프로젝트에 적용하였습니다.
  - 또한 SSAFY 특화 프로젝트 주제인 블록체인에 대해 배경 지식이 전무했지만, 레퍼런스 문서를 바탕으로 꾸준히 학습했고 이를 바탕으로 블록체인 관련 코딩을 작성했습니다.

## 🎨 My Career

☐ 2019년 3월~2019년 12월

충남대학교 우리은하연구실 학부연구생 경험

- 처음으로 코딩을 입문했습니다! 😊 Python을 학습하고 `MathPlotLib` , `Numpy` , `Pandas` 라이브러리를 활용하여 데이터를 그래프화 하는 작업을 경험했습니다.

● 2020년 5월~2021년 1월

2020년 5월 ~ 2020년 7월 : 초급 프로젝트 수행

대덕인재개발원

전자정부 표준 프레임워크 기반

풀-스택 개발자 양성과정 수료

- 2달 동안 JAVA 언어를 교육받으면서 OOP, 상속, Overload, Override 등의 기초 개념을 배웠습니다.
- **Oracle SQL Developer** 를 통해 DB 기초 개념, 기본적인 SQL문과 정규화 개념을 습득했습니다.
- 2인 미니 프로젝트를 수행하였으며, **팀장 역할을 수행**했습니다. JDBC(Java DataBase Connectivity)를 이용하여 CLI 환경에서 **영화 예약 서비스**를 개발했습니다.

#### 2020년 8월 ~ 2020년 10월 : 중간 프로젝트 수행

- 기초적인 HTML, CSS, JS(Under ES6)를 학습했습니다.
- **jQuery**, **ajax** 를 배우며 백엔드 서버와 통신하는 방법을 공부했습니다.
- Spring Framework를 배우며 Bean, Spring MVC Pattern, 의존성 주입 등의 기본 개념을 공부했습니다.
- **iBatis** 를 통해 spring과 DB를 연결하였습니다.
- 형상관리툴인 SubVersion(SVN)과 협업툴인 RedMine을 통해 프로젝트를 관리했습니다.
- Project Build Tool로 **Maven** 을 사용하였습니다.
- **Servlet** 과 **JSP** 를 공부했으며 팀장 역할을 맡아 Spring을 통해 **구인구직 서비스를 주제로 프로젝트**를 진행했습니다.

#### 2020년 11월 ~ 2021년 1월 : 최종 프로젝트 수행

- 해당 교육에서 진행한 3번의 프로젝트 중, 최종 프로젝트로 팀장 역할을 맡아 **협업관리툴 서비스를 주제로 프로젝트**를 진행했습니다.
- **MyBatis** 를 통해 Spring과 **Oracle SQL Developer** 를 연결하였습니다.

- 프로젝트 형상관리툴로 **Git** 을 사용하였습니다. 이와 관련된 **commit** , **push** 등의 기초 개념을 학습했습니다.
- **Git** 을 GUI 환경에서 사용할 수 있는 **Source Tree** 를 사용하여 프로젝트를 관리했습니다.

<https://github.com/tofan0412/finalAnts>

🤔 내가 되고 싶은 개발자가 되기 위해선, 어떤 노력을 해야 할까?

8개월 동안의 국비교육을 마친 후 저는 실력있는 개발자가 되기 위해 반드시 필요한 CS, 알고리즘, 웹 개발 최신기술을 학습하고 싶었습니다. 이를 위해 관련 교육을 찾아보던 중, SSAFY를 알게 되었고 최종합격하여 1년 동안 교육을 받게 되었습니다☺

#### □ 2021년 1월 ~ 2021년 12월

삼성전자 삼성 청년 소프트웨어 아카데미

(Samsung Software Academy For Youth)

5기 교육 수료

#### 2021년 1월 : 스타트 캠프

- Jupyter Notebook을 통해 Python 기초 내용을 학습했습니다. 기초적인 문법부터 시작하여 간단한 알고리즘도 함께 공부했습니다.
- 아이디어톤 경진대회에서 **웹툰 & 웹소설 인물 검색 서비스** 를 주제로 우수상을 수상했습니다.

#### 2021년 2월 ~ 2021년 5월 : 알고리즘 집중 교육 & 웹 개발 기초 학습

- 알고리즘
  - 시간복잡도

- 정렬
- Greedy
- Array
- 완전탐색과 이진탐색
- String
- DFS와 BFS - Graph 탐색
- DP
- 재귀함수
- Backtracking
- Tree
- ES6를 기반으로 한 Vanilla Javascript
- 기초 HTML 문법 학습
- Vue.js를 통한 웹 페이지의 구현 및 router와 vuex 학습
- Bootstrap을 통한 CSS 디자인
- Django & ORM, SQL
- 1학기 관통 프로젝트 : **영화 추천 서비스 (SSAFYDB)**

<https://github.com/tofan0412/MovieService>

**2021년 7월 ~ 2021년 11월 : 공통 프로젝트 / 특화 프로젝트 / 자율 프로젝트 수행**

- 공통 프로젝트 : **WebRTC 기반 소개팅 서비스 - Cow&Dog**

<https://github.com/tofan0412/Cow-Dog>

- 특화 프로젝트 : [비트코인 기술 기반 자격증명 서비스 - Blanket](#)

<https://github.com/tofan0412/Blanket>

- 자율 프로젝트 : [WebRTC 기반 AR 소개팅 서비스 - Mokkozi](#)

<https://github.com/tofan0412/mokkozi>