

# CodeLink 최종보고서

유학생 오픈채팅 앱, Yumi



201320453 정상현

201320191 이윤구

201421150 홍동기

201521036 김정환

# 목 차

<b>1</b>	<b>초록</b>	5
<b>2</b>	<b>개요</b>	5
2.1	컨셉 기획	5
2.1.1	문제 정의	5
2.1.2	Target 및 Stakeholders	6
2.1.3	가치	7
2.1.4	목표	7
<b>3</b>	<b>배경</b>	9
3.1	기존 사례	9
3.1.1	카카오톡 오픈채팅	9
3.1.2	아자르	10
3.2	기술적 배경	11
3.2.1	기초지식	11
3.3	제한 요소 분석	12
<b>4</b>	<b>시스템 구성 및 기능</b>	13
4.1	활용 기술 스택	13
4.2	핵심 알고리즘/기술에 대한 상세 설명	14
4.2.1	Socket.io	14
4.3	시스템 설명	15
4.3.1	전체 시스템 구조도	15
4.4	최종 구현 환경	17
<b>5</b>	<b>구현 결과</b>	18
5.1	기능 설명	18
5.2	동작 과정 설명	19
5.2.1	회원가입	19
5.2.2	로그인	19
5.2.3	채팅방 검색	19
5.2.4	채팅방 생성	19
5.2.5	채팅방 추천	19
5.2.6	채팅	20
5.2.7	자동번역	20
5.2.8	사진 업로드	20
5.2.9	팝 퀴즈	20
5.2.10	언어 교환	21
5.2.11	프로필 변경	21
5.2.12	로그아웃	21
5.2.13	국제협력처 정보 확인	21

5.2.14	국제협력처 정보 등록.....	21
5.2.15	국제협력처 정보 승인.....	21
5.3	테스트 시나리오.....	22
5.4	매뉴얼/예제 .....	25
<b>6</b>	<b>과제 관리</b> .....	<b>26</b>
6.1	BD 차트 기반 진행과정 설명.....	26
6.2	협업 방식/도구 설명 .....	26
6.3	구성원간 업무 분장.....	27
6.4	비용 .....	27
6.4.1	SW 구축 .....	27
6.4.2	유지/보수 .....	27
6.4.3	증가율.....	27
<b>7</b>	<b>결론 – Post Mortem</b> .....	<b>28</b>
7.1	프로젝트 진행(실행) 과정에서 잘 된 점 그 이유.....	28
7.2	프로젝트 진행 과정에서 아쉬운 점, 개선 아이디어.....	28
7.3	본 과제 결과물의 발전 방향.....	28
7.4	기타 소감 .....	29
7.4.1	정상현.....	29
7.4.2	이윤구.....	29
7.4.3	홍동기.....	29
7.4.4	김정환.....	29
<b>8</b>	<b>참고 문헌</b> .....	<b>30</b>
<b>9</b>	<b>Appendix</b> .....	<b>30</b>
9.1	Glossary.....	30
<b>10</b>	<b>해당 프로젝트 github 링크</b> .....	<b>30</b>

## 그림 목차

그림 1 Yumi Stakeholders.....	6
그림 2 카카오 오픈채팅 Stakeholders .....	9
그림 3 아자르 Stakeholders.....	10
그림 4 전체 시스템 구조도 .....	15
그림 5 각 구성 요소 별 SW 구조 정의.....	16

## 표 목차

표 1 기초지식.....	151
표 2 프로젝트 관련 학과목 .....	151
표 3 활용 기술 스택.....	15
표 4 각 구성 요소별 역할 .....	155
표 5 각 컴포넌트 역할 .....	176
표 6 최종 구현 환경.....	177
표 7 기능 설명 .....	188
표 8 테스트 시나리오.....	24
표 9 Iteration 상세 정보 .....	266
표 10 협업 방안/도구.....	266
표 11 업무 분장.....	277
표 12 Glossary.....	3030

## 1 초록

현재 한국에는 점점 더 많은 외국인 유학생들이 유학을 오고 있다. 하지만, 한국에 온 외국인 유학생들이 맨땅에서 정보를 얻기까지 많은 무리가 따른다.

우리는 이러한 문제점을 해결하고, 외국인 유학생들의 국내 유학 생활의 질을 높이기 위하여 하나의 환경에서 유용한 정보들을 접할 수 있도록 socket.io 를 통해 오픈 채팅, 언어 교환, 채팅방 추천 등의 기능을 구현하였고, 또 하나의 탭에서는 유학생에게 필요한 공지를 한 눈에 모아 볼 수 있는 기능을 접목하여 애플리케이션을 만들었다.

또한, 유학생과 관련된 단체들은 Yumi 에게 자신의 단체가 진행하는 행사에 대한 홍보나 인원 모집을 신청하는 것으로, 빠르고 효율적으로 행사를 진행할 수 있다.

Express 와 mongodb 를 이용한 서버는 위의 기능들이 지연이나 누락, 트래픽 낭비를 줄이도록 구성되었으며, Sqlite 를 이용해 만든 로컬 DB 는 서버 DB 와 일관성을 유지하도록 논리적으로 구성되었으며, React Native 환경에서 구성된 UI 는 사용자의 입장에서 편리한 사용을 중심으로 구성하였다. 빌드와 배포는 expo 를 이용하였고, expo 의 hot-reload 기능을 통해 코드 수정 후 빠르게 결과물을 확인할 수 있었다. 또한, 관리자 페이지는 vue.js 를 통해 한번에 넓은 데이터를 볼 수 있는 웹 환경에서 서버와 통신을 진행한다.

Yumi 는 위의 기능들을 바탕으로, 결과적으로 유학생들이 풍성한 유학 생활을 영위할 수 있도록 돕는 무료 애플리케이션이다.

## 2 개요

### 2.1 컨셉 기획

#### 2.1.1 문제 정의

##### 2.1.1.1 증가하고 있는 외국인 유학생

- 교육부 통계에 따르면 현재 한국에 거주하는 외국인 유학생은 약 14 만 명으로, 작년 대비 20% 제 작년 대비 40% 증가하는 추세에 있다.
- 정부에선 학사 졸업 시 한국 영주권 취득 및 귀화할 수 있는 제도를 구축하였고 장학금, 생활비 지원 등의 지원사업 규모를 확대하며 적극적으로 외국인 유학생을 유치하고 있다. 이는 한국의 저 출산으로 인한 인구감소를 위한 정책 중 하나이며 때문에 유학생 관련 지원사업은 지속해서 이루어질 것으로 보인다.
- 이에 맞추어 외국인 유학생 수 또한 꾸준히 증가할 것으로 보인다.

##### 2.1.1.2 한국어로 된 공지사항 외 정보습득 수단 미비, 동아리 외 친목 수단 부족

- 2019 년 고등교육기관 외국인 유학생 현황 자료에 따르면 아주대학교(대학원 포함)에 재학중인 외국인 유학생은 약 1000 명 정도이며 이들은 한국어로 되어있는 아주대학교 국제협력처 공지사항, 교내 건물 현수막 이외의 정보 습득 수단이 전무하며 교내 동아리에서 개최하는 행사 이외의 친목 수단 또한

없는 실정이다. 따라서 교내 외국인 유학생들은 정보 부족, 제한된 친목 수단에 불편을 호소하고 있다.

#### 2.1.1.3 외국인 유학생 관리 수단 미비

- 국제협력처의 입장에서 홈페이지 공지사항, 개개인에게 연락 이외의 외국인 유학생들을 관리, 컨트롤 할 수 있는 수단이 전무하다.

#### 2.1.1.4 일반 한국인 학생 – 외국인 유학생 간 교류 수단 전무

- 관련 동아리에 소속 되어있지 않은 한국인 학생의 경우 외국인 유학생과 교류할 수 있는 수단이 없다.
- 제한된 친목 수단에 시달리는 외국인 유학생들을 위해 한국인 학생과 언어장벽을 해소해주며 교류할 수 있는 수단이 필요하다.

### 2.1.2 Target 및 Stakeholders

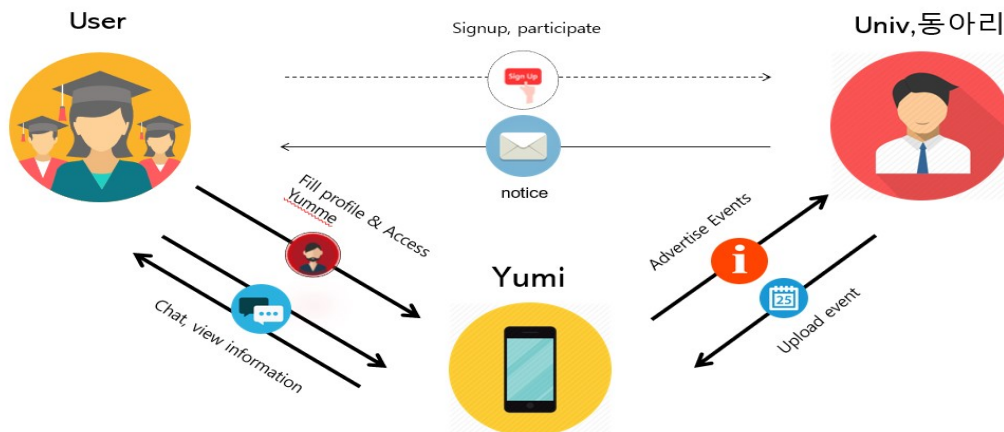


그림 1 Yumi Stakeholders

#### 2.1.2.1 User

- 외국인 유학생, 한국인 학생이 주 타겟이며 Yumi 를 통해 언어 장벽에서 벗어나 다양한 주제로 자유롭게 소통하고 싶어하며, 서로의 언어 또한 배우고 싶어한다.

#### 2.1.2.2 Yumi

- 외국인 유학생, 한국인 학생들에게 언어의 장벽 없이 소통, 교류할 수 있는 공간을 제공하고 언어 교환으로 서로의 언어를 배울 수 있게 해준다.
- 국제협력처, 동아리의 공지사항을 번역하여 보여준다.

#### 2.1.2.3 Univ, 동아리

- 외국인 유학생 협력처이며, 외국인 유학생들에게 효과적으로 정보 전달을 해줄 수단이 필요하다.

### 2.1.3 가치

#### 2.1.3.1 Yumi -> User

- 사용자는 Yumi 의 기능들을 무료로 사용할 수 있으며 인맥 형성 수단이 부족한 유학생들에게 교내 다양한 유학생 및 한국인 학생들과 소통, 교류할 수 있는 공간을 제공하고 팝 퀴즈로 대학 생활에 유용한 정보를 얻는다.

#### 2.1.3.2 Yumi -> 국제협력처

- Yumi 는 채팅방 추천 탭에서 등록한 유학생 서포터즈의 소개, 이벤트 홍보에 대한 정보를 사용자에게 보여준다.
- User 에게 다양한 유학생 서포터즈의 존재를 알리고 가입 유도를 함으로써 수많은 서포터즈의 존재를 모르는 유학생들이 자신에게 맞는 서포터즈를 찾고 가입을 할 수 있도록 도와준다.

#### 2.1.3.3 User -> Yumi

- 사용자는 회원가입 시 Yumi 에게 이름, 이메일, 성별, 사는 지역, 소속 대학교 등의 정보를 입력하고 이메일 인증을 통해 자신이 학교 소속 학생임을 인증한다.
- 사용자는 Yumi 에게 이메일 인증, 개인정보를 제공함으로써 현재 자신이 외국인 유학생 또는 한국인 학생임을 알리고 Yumi 의 채팅방 추천 시 사용자의 입력된 개인정보를 활용하여 더욱 정확한 추천이 가능하다.

#### 2.1.3.4 국제협력처 -> Yumi

- 국제협력처는 유미 관리자 페이지를 통해 홍보 신청을 한다. Yumi 의 서포터즈 홍보 양식에 서포터즈 이름, 이메일, 전화번호, 소개 글을 작성하여 Yumi 에 전송하면 Yumi 에서 검토 후 해당 이벤트 정보를 국제협력처 추천 탭에 표시해준다.
- Yumi 는 국제협력처에게 자신들의 이벤트를 알릴 수 있는 홍보 플랫폼을 제공해주고 국제 협력처는 콘텐츠 생산에만 집중하는 국제협력처-Yumi 간의 B2B(business to Business) 플랫폼을 구축할 수 있다.

### 2.1.4 목표

- 외국인 유학생: 언어의 장벽에서 벗어나 다양한 국가의 사람들과 함께 소통하고 외국인 유학생 관련 최신 정보를 습득함으로써 성공적인 한국 유학생생활이 가능하도록 만들어주는 것이 목표다.

- 한국인 학생: 멀게만 느껴졌던 외국인 유학생들과 자유롭게 소통, 1 대 1 언어 교환을 통해 새로운 인간관계 형성, 문화체험을 가능하게 함으로써 교내에서도 외국 대학생활을 간접 체험할 수 있다는 것을 알려주는 것이 목표다.
- 국제협력처: 기존의 현수막, 공지사항 등의 제한된 정보전달 경로에서 벗어나, Yumi 가 정보 전달을 전담함으로써 대학교 - Yumi 간의 업무 분장이 목표이다. 대학교는 유학생들을 위한 콘텐츠 생산에만 집중함으로써 유학생의 복지를 높일 수 있다.



### 3 배경

#### 3.1 기존 사례

##### 3.1.1 카카오톡 오픈채팅

- 카카오톡에서 개발한 카카오톡 오픈채팅은 카카오톡의 주요 기능 중 하나로 소셜 네트워크 서비스이다. 사용자가 직접 원하는 주제의 채팅방에 접속하여 익명으로 채팅, 투표, 미디어 파일 업로드, 사용자 밴, 방장 설정 등의 기능을 사용할 수 있다. 앱은 iOS, Android, RestAPI, Kotlin 으로 개발되었으며 PC 버전은 C++로 개발되었다.
- 카카오톡 오픈채팅은 익명성을 바탕으로 규정된 규칙을 지키며 사회, 인간관계에 얽매어 하지 못했던 자신의 솔직한 의견을 말할 수 있고 반대로 자신과 다른 성격, 가치관을 가진 아바타를 창조해내어 감정이입 함으로써 자신과 다른 정체성을 가진 가상의 인물을 간접 경험해볼 수 있다.

##### 3.1.1.1 Stakeholders



그림 2 카카오 오픈채팅 Stakeholders

##### 3.1.1.2 규모, 수익구조, 지속 성장 가능성

- 카카오톡 오픈채팅은 앱 내 광고 노출, 채팅목록에 광고를 삽입하는 B2B 모델, 이모티콘 판매, 카카오 커머스, 게임, 음악 등의 콘텐츠 제공을 통해 수익을 내고 있다.
- 카카오는 2019 년 1 분기 상반기 매출 7063 억 원, 영업이익 277 억 원을 달성하여 역대 최고 매출을 달성하였다. 또한, 연내 성장 목표인 매출 23% 성장을 1 분기 매출 증가 26%로 미리 달성하는 등 폭발적이고 안정적인 규모와 성장세를 이어 나가고 있다.

### 3.1.2 아자르

- 아자르는 하이퍼커넥트에서 개발한 비디오 기반의 소셜 엔터테인먼트 플랫폼이다.
- 사용자가 아자르에 접속하면 랜덤하게 뽑힌 다른 사용자와 1 대 1 영상통화를 진행하게 되며 나라, 사용 언어의 제한 없이 매칭된다.
- 1 대 1 영상통화 중 비디오, 얼굴인식 필터를 적용할 수 있고 채팅을 보내면 해당 사용자의 모국어로 자동 번역함으로써 사용하는 언어와 관계없이 자유롭게 화상 통화가 가능하다는 점으로 다른 영상통화 앱과 차별점을 두고 있다.
- 아자르는 언어의 장벽에 얽매이지 않고 다양한 국가의 사람들과 직접 얼굴을 보며 소통함으로써 해당 나라의 문화, 분위기를 간접 체험할 수 있고 랜덤 채팅보다 훨씬 더 사용자 간의 직접적인 관계를 맺을 수 있다.

#### 3.1.2.1 Stakeholders

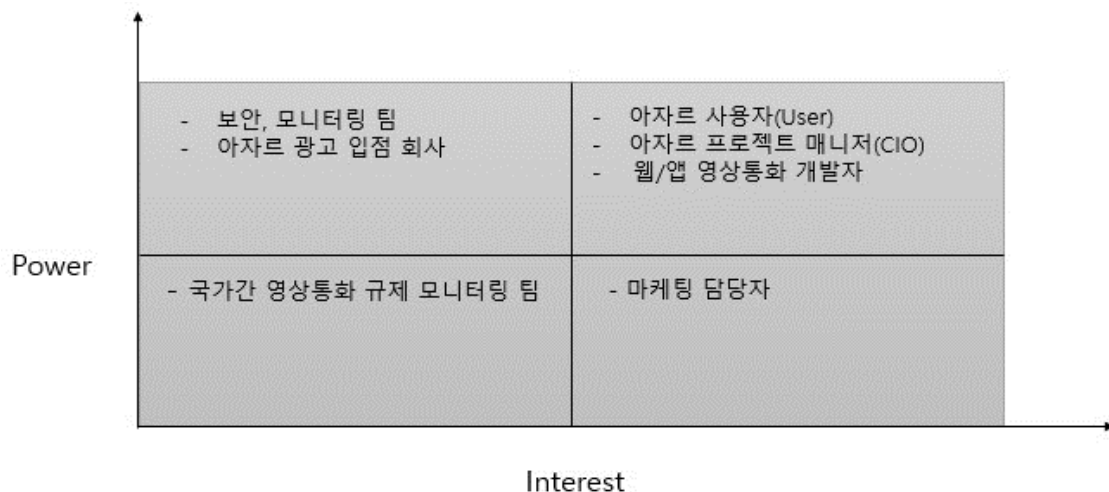


그림 3 아자르 Stakeholders

#### 3.1.2.2 수익구조, 규모, 지속 성장 가능성

- 아자르는 통화할 상대의 성별, 지역 선택 시 일정 비용을 내야 한다. 영상통화를 진행할 때마다 비용을 내야 함으로 아자르의 사용빈도가 높을수록 지출금액이 증가하는 수익구조로 되어 있다.
- 아자르는 지난해 연 매출 1 천억 원, 2019 년 1 분기 구글플레이 비 게임 앱 매출 순위 5 위 누적 다운로드 수 2 억 건을 달성하였다. 같은 집계에서 2017 년 9 위, 2018 년 7 위를 달성하며 꾸준히 성장세에 있다.

## 3.2 기술적 배경

### 3.2.1 기초지식

유형	내용	비고
프론트 엔드	Html, css, Vue, React-native, Sqlite	프로젝트의 웹/앱 프로그래밍을 위해 해 각 요소별 자세한 설명은 아래에 후술
백 엔드	MongoDB, Socket.io, Node.js	프로젝트의 데이터 저장, 채팅 서비스 구현에 필수적이며 각 요소별 자세한 설명은 아래에 후술
공통	RestfulAPI, Javascript, Java, API	프로젝트의 기초 프로그래밍, 알고리즘 적용, 프론트-백엔드 통신에, 동시번역 구현의 필수적이며 각 요소별 자세한 설명은 아래에 후술

표 1 기초지식

### 3.2.2 프로젝트 관련 학과목

과목 이름	비고
자료구조	배열, List, queue 등의 기초적인 데이터 사용법을 학습한다.
객체지향 및 프로그래밍	객체 프로그래밍의 원리 및 생성자, 상속, 함수 선언 등 Yumi 프로그래밍에 필수적인 개념들을 학습한다.
컴퓨터 네트워크	데이터 통신에 대한 기초원리 및 프로토콜에 대하여 학습한다.
컴퓨터 통신	Socket.io 로 웹 클라이언트 구축에 대하여 학습한다.
오픈소스 입문	다양한 오픈소스 및 API 를 사용하는 방법을 학습한다.
데이터 베이스	기본적인 데이터베이스 사용법과 쿼리문을 배운다.
웹 시스템 설계	프론트-백엔드 정의, 자바스크립트, Express MongoDB, Vue, webpack 에 대하여 학습한다.
도메인 분석 및 설계	소프트웨어 시스템 설계 방법, 리팩토링에 대하여 학습한다.
실전 코딩	Git, 가상머신(AVD manager), React, mongoose, postman 등의 사용법을 학습한다.
알고리즘	데이터의 Sorting, 프로젝트 내에 사용될 알고리즘을 개발하기위해 학습한다.

표 2 프로젝트 관련 학과목

### 3.3 제한 요소 분석

- 기술적 제약
  - 1) 프론트 엔드 관련: React-native 를 사용하여 개발하였기 때문에 유저정보 변경 등의 이벤트 발생시, 앱이 자주 렌더링(새로 고침) 될 수 있다. 또한, 개인 정보 이슈로 인해 사용자 관련 정보는 기기의 내부 DB 에 저장되도록 설계되어 있다. 이로 인해 사용자가 기기를 변경할 경우, 사용자 관련 정보를 다시 설정해야 한다.
  - 2) 백 엔드 관련: 채팅 서비스 제공에 Socket.io 를 사용하였기 때문에 다양한 network 사용으로 인한 이벤트 유실(채팅 보내지지 않음 등)이 발생할 수 있다.
- 데이터 수집
  - 1) 외국인 유학생 관련: 앱 내의 interaction 을 통해 외국인 유학생의 이메일, 취미, 사용 언어 정보를 받는다. 수동으로 사용자 데이터를 받기 때문에 사용자가 고의로 정보를 누락할 경우 원활한 서비스 제공(동시 번역, 채팅방 추천)을 받지 못할 수 있다.
  - 2) 한국인 학생 관련: 외국인 유학생과 동일
  - 3) 국제협력처, 동아리 관련: Yumi 관리자 페이지에서 홍보할 공지사항, 이벤트 정보를 수동으로 입력하게 된다.
- 사회적 영향
  - 1) 아주대학교에 미치는 영향: 기존 공지사항, 현수막을 통한 정보전달에서 벗어나 새로운 홍보 플랫폼인 Yumi 를 통해 좀 더 효율적인 정보전달이 가능하다. 이를 통해 기존의 외국인 유학생 지원 시스템에서 벗어나 비용, 콘텐츠 생산 면에서 효율성 개선에 영향을 줄 수 있을 것이라고 예상된다.
- 비용/경제성
  - 1) 비용: Yumi 는 개발하는 데에 서버 유지비를 제외한 추가적인 비용이 들지 않으며, 연간 약 90 만원(자세한 분석은 6.4 에 후술 되어있음)의 유지비만 사용될 것으로 예상된다.
  - 2) 경제성: 국제협력처와 B2B(Business to Business) 협약을 체결할 경우, 공지사항을 운영할 인건비, 현수막 설치비용(5M 기준 1 장당 5 만 7000 원) 의 비용을 절약 할 수 있다. 따라서 국제협력처에서 현수막, 공지사항을 설치하여 운용하는 것 보다, 재학생들에게 Yumi 를 설치하도록 유도하는 것이 유학생 관리와 비용 면에서 더욱 경제성이 높다고 판단된다.

## 4 시스템 구성 및 기능

### 4.1 활용 기술 스택

목적	기술 스택	선택 이유 및 활용 방안
개발 및 데이터 저장/분석	JavaScript	Node.js 와 React Native 모두 JavaScript 기반으로 되어 있는 언어이므로 같은 언어를 사용하여 개발할 수 있다는 편의성을 고려하여 선택하였다.
	HTML5	정보를 제공할 때 html은 단말기, OS, 브라우저에 상관없이 사용될 수 있으므로 선택하였다.
	CSS	문서의 모양을 제어할 때, 문서의 구조와 스타일을 분리하여 관리할 수 있어서 선택하였다. CSS를 사용하여 앱을 디자인하였다.
	Expo Image Picker	Expo를 활용한 환경에서 이미지를 선택할 수 있는 기능을 제공한다. 프로필사진 변경, 채팅방 내 사진 전송 등에서 사용한다.
	Expo Sqlite	유저 정보, 채팅 내용, 채팅방 목록, 개인 설정 등을 사용자의 로컬 DB에 저장하기 위해서 사용한다. 해당 내용을 서버 DB에만 저장할 시 사생활 침해 문제가 생길 수 있으며, 매 실행마다 서버의 응답을 기다려야 한다. 따라서 로컬 DB인 SQLite를 신뢰성 있게 사용하여 사생활 침해를 방지하며 앱의 속도도 향상시킨다.
	React Native	Vue.js와 비교하여 더 단순화하여 컴포넌트를 정의할 수 있으며, JavaScript에 익숙하여 빠른 개발이 가능하다고 생각하였고 하이브리드 앱을 목표로 하고 있어 선택하였다.
	React Native Dialog Input	팝업창에서 값을 입력 받을 수 있게 해준다. 팝퀴즈 정답 입력을 위해 사용하였다.
	React Native Elements	React Native UI Toolkit 중 하나로, 스크롤 뷰를 통한 리스트 구현에 사용되며, 디자인적으로 뛰어난 컴포넌트를 제공하기에 선택하였다.
	React Native Gesture Handler	제스처 동작을 구현하는데 도움을 주는 패키지로, Yumi 내에서는 채팅방 내에서 Drawer(좌측 메뉴)를 구현하는데 사용하였다.
	React Navigation	앱 내의 원활한 화면 전환 및 탐색 기록 관리가 수월하여 선택하였다.
	React Navigation Stack	React Navigation에서 Stack를 이용해 탭을 구성하여 이동 경로를 저장하고, 이전 경로로 돌아가는 기능을 수행한다.
	React Navigation Tabs	React Navigation에서 하단 바 기능이 분리되어 기능을 사용하기 위해 선택하였다.
	Native Base	React Native UI 컴포넌트 중 하나로, 많고 깔끔한 벡터 아이콘을 제공하기에 선택하였다.
	Socket.io Client	FrontEnd에서 Backend의 socket.io와 통신하기 위하여 사용하는 패키지다.
	Node.js	이벤트 기반의 비동기 방식으로 서버 측에 지속적인 메시지가 왔는지 확인하는 클라이언트의 작업이 축소되어 서버에 무리가 적어 선택하게 되었다. 또한, 한 가지 언어를 통해 서버-클라이언트를 모두 개발할 수 있어 선택하였다.

	MongoDB	채팅 앱의 경우 복잡한 스키마가 필요 없어 RDB 가 필요 없다고 판단하였다. 따라서 RDB 에 비해 성능이 빠른 MongoDB 를 사용하였다.
	Express	API 를 설계하기 쉽고, 내장된 라우터를 이용하여 코드를 쉽게 재사용할 수 있어 선택하였다. 또한 Node.js 기반의 프레임워크로 Node.js 로 서버, 클라이언트 모두 개발할 수 있어서 편리하다.
	NAVER CLOUD PLATFORM	서버를 클라우드에 올리기 위하여 사용한다. 간단한 서버 구축하는데 사용하기 용이하여 선택하였다.
	Socket.io	실시간 웹 어플리케이션을 위한 JavaScript 라이브러리이다. 우리의 서비스에서 채팅을 통해 발생하는 모든 이벤트는 이 Socket.io 를 기반으로 이루어진다.
	mongoose	MongoDB 를 이용해서는 low level 의 드라이버 역할을 하였고, mongoose 는 ODB(Object Document Modeling)으로 관계 모델링 툴로써 사용하였다. 이를 통해서, 스키마 작성을 용이하게 하였다.
	Vue.js	관리자 웹 제작을 위한 Node.js 템플릿 엔진이다. 관리자 웹에 사용되는 기능이 많지 않기 때문에 다른 프론트엔드 프레임워크를 사용하지 않고 vue 를 사용하여 구현할 계획이다.
데브옵스	GitHub	코드 변경 관리 및 코드 리뷰를 위한 협업 툴로써 사용한다.
	Postman	API 작성 및 테스트를 위해 사용한다. 툴을 활용하여 API 작성을 하면 보다 편리하게 작성할 수 있어서 선택하였다.
	Expo	React Native 의 빠른 배포와 간단한 프로젝트 설치 기능의 장점이 필요하다고 생각하여 선택하였다.
	Expo CLI	Expo 를 설치하면 자동으로 설치되는 CLI 툴이다. 빠른 빌드와 직관적인 UI 가 특히 장점이다.
	Robo 3 T	mongoDB 조작 및 관리 툴로써, 보다 쉽게 데이터를 관리하기 위한 데브옵스 툴이다.
비즈니스	Slack	GitHub 과 Jenkins 와 연동하여 프로젝트 진행 사항을 한눈에 알아볼 수 있고, 팀원들간의 협업에서도 유용하게 사용할 수 있어서 선택하였다.

표 3 활용 기술 스택

## 4.2 핵심 알고리즘/기술에 대한 상세 설명

### 4.2.1 Socket.io

- Socket.io 는 실시간 웹 어플리케이션을 위한 JavaScript 라이브러리이다. 클라이언트와 서버에서 사용되는 API 의 구조가 동일하여 개발하는데 편리하고 웹 클라이언트와 서버 간에 실시간 양방향 통신을 가능하게 해준다. Node.js 와 같이 이벤트 중심으로 이루어져 있다. 여러 소켓에게 브로드 캐스트, 각 클라이언트와 연관된 데이터 저장 등의 기능을 통하여 단체 채팅과 관련된 기능에 사용되기 좋다고 판단하여 사용하였다.

## 4.3 시스템 설명

### 4.3.1 전체 시스템 구조도

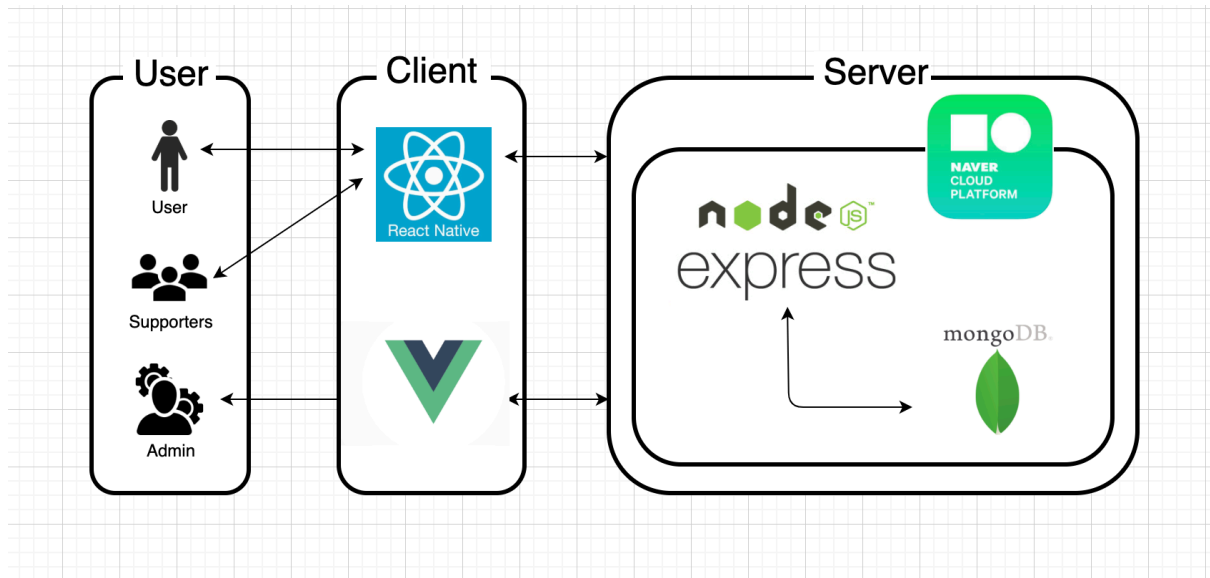


그림 4 전체 시스템 구조도

구성 요소	역할
User	시스템을 사용하는 사용자
Supporters	시스템을 이용하여 사용자(고객)에게 정보를 알리고 싶은 사용자.
Admin	시스템을 관리하는 관리자
React Native	시스템을 사용할 수 있는 Frontend 를 구성하는 Framework.
Vue.js	관리자가 사용자(고객)의 요구에 응답 및 시스템 관리를 할 수 있는 Frontend
Naver Cloud Platform	서버를 구축하기 위한 네이버의 클라우드 플랫폼
Node.js	서버를 구축하기 위한 플랫폼
Express	Node.js 기반의 서버를 구축하기 위한 프레임워크
MongoDB	NoSQL 기반의 데이터베이스로 사용자에게 대한 정보를 저장 및 관리

표 4 각 구성 요소별 역할

#### 4.3.2 시스템 구조도

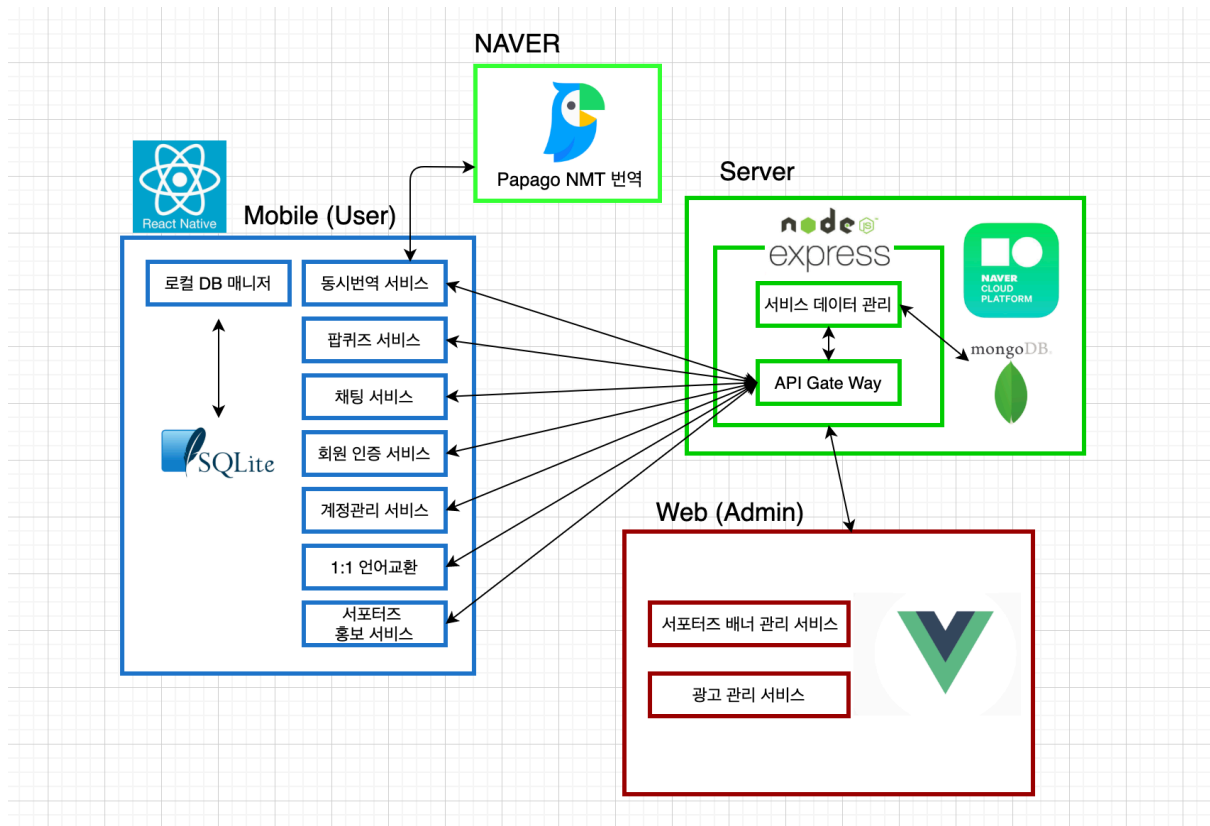


그림 5 각 구성 요소 별 SW 구조 정의

컴포넌트	역할
로컬 DB 매니저	로컬 DB 에 저장되어질 데이터의 저장 기간 및 저장 필요 유무에 대한 판단을 통해 데이터를 관리한다.
채팅 서비스	채팅을 할 수 있는 기능을 가진다.
Pop Quiz 서비스	채팅방 내의 Pop Quiz 서비스에 대한 기능을 가진다.
1:1 언어교환	외국어를 배우고 싶은 사용자에게 배우고 싶은 언어의 사용자와 1:1 채팅방을 만들 수 있게 하여 서로 언어를 연습할 수 있는 환경을 동시번역 서비스를 이용하여 만들어준다.
회원인증 서비스	우리의 서비스를 사용할 수 있는 회원인지 인증해주는 서비스이다.
계정관리 서비스	자신의 계정을 관리할 수 있는 기능이다.
서포터즈 홍보 서비스	서포터즈 배너 관리 서비스를 통해 승인을 받은 홍보 배너를 볼 수 있는 서비스이다.
서포터즈 배너 관리 서비스	홍보를 원하는 서포터즈가 배너를 사용하기 위해 관리자에게 승인을 받을 수 있게끔 해주는 서비스이다.
광고 관리 서비스	관리자가 광고를 관리하는 서비스이다.
서비스 데이터 관리	서비스에 필요한 다양한 데이터를 관리할 수 있는 서비스이다.



API Gate Way	각각의 컴포넌트를 서버에서 필요에 따라 연결하고 사용할 수 있도록 하는 기능을 한다.
--------------	---

**표 5 각 컴포넌트 역할**

#### 4.4 최종 구현 환경

최종 구현 환경	분류
디바이스	갤럭시 탭
	Android Virtual device android 10.0
	Expo version : 3.3.0
서버/클라우드	OS image type : ubuntu 16.04-64-server
	Disk : 50GB
	Server type : Standard
	Node js version : v12.10.0

**표 6 최종 구현 환경**

## 5 구현 결과

### 5.1 기능 설명

기능	설명
회원가입	사용자는 학교 이메일로 인증을 한 뒤, 아이디, 비밀번호, 언어, 관심분야를 입력하여 회원가입을 한다.
로그인	사용자는 아이디와 비밀번호를 입력하여 로그인을 한다.
채팅방 검색	사용자는 검색어를 입력하여 원하는 채팅방을 검색한다.
채팅방 생성	사용자는 원하는 관심분야를 선택하여 해당 관심분야에 대한 채팅방을 생성한다.
채팅방 추천	사용자가 회원가입시에 입력한 관심분야를 바탕으로 알맞은 채팅방을 추천받는다.
채팅	사용자는 다른 사용자와 함께 채팅을 한다.
자동번역	사용자는 다른 외국인과 채팅을 할 때 자동으로 본인의 언어로 번역되어 확인한다. 단, 채팅 메시지를 클릭할 경우 기존 채팅 메시지를 확인할 수 있다.
사진 업로드	사용자는 채팅방에서 사진을 업로드 할 수 있으며, 다른 사용자가 보낸 사진을 확인할 수 있다.
팝 퀴즈	시스템은 일정 주기마다 학교 정보 및 해당 관심분야 관련 퀴즈를 출제하며, 사용자는 다른 사용자와 경쟁하며 답을 맞힌다.
언어 교환	사용자가 본인의 언어를 제외한 외국어를 공부하고 싶을 경우 해당 언어를 검색하면, 해당 언어를 사용하는 사용자와 1:1로 채팅을 할 수 있다.
프로필 변경	사용자는 프로필 사진, 관심분야, 사용 언어를 변경한다.
로그아웃	사용자는 본인의 계정을 로그아웃 한다.
국제협력처 정보 확인	사용자는 국제협력처에서 제공하는 정보들을 한 곳에서 확인한다.
국제협력처 정보 등록	국제협력처는 사용자들에게 노출을 원하는 정보를 관리자에게 등록 요청을 한다.
국제협력처 정보 승인	관리자는 국제협력처에서 등록 요청한 정보들을 검토한 후 승인한다. 승인 시 사용자들이 확인할 수 있다.

표 7 기능 설명

## 5.2 동작 과정 설명

### 5.2.1 회원가입

- 1) 회원가입 버튼을 누른다.
- 2) 학교 이메일 인증을 위해 이메일을 입력한 후, 확인 버튼을 누른다. 이메일 정보는 서버에 보낸다.
- 3) 서버에서 nodemailer 를 사용하여 인증번호가 담긴 이메일을 보낸다.
- 4) 인증번호를 입력하여 인증한 뒤, 개인정보(비밀번호, 닉네임, 사용 언어, 관심분야)를 입력한다. 개인정보는 서버에 보낸다.
- 5) 서버에서 개인정보를 받아 비밀번호는 SHA1 암호화 기법을 사용하여 암호화한 뒤, DB 에 개인정보를 저장한다.
- 6) 회원가입이 완료되며, 로그인 화면으로 전환된다.

### 5.2.2 로그인

- 1) 이메일과 비밀번호를 입력하고 로그인 버튼을 눌러 서버에 보낸다.
- 2) 서버에서 비밀번호를 DB 에 저장되어 있는 비밀번호와 비교하고, 일치하지 않으면 실패 메시지를, 일치한다면 성공 메시지와 함께 사용자의 정보와 JWT 를 함께 클라이언트에 보낸다.
- 3) 로그인에 성공했다면 사용자의 정보와 Access Token 과 Refresh Token 을 로컬 DB 에 저장하고, 메인 화면으로 전환된다.
- 4) 어플에 재접속 시 로컬 DB 에 저장되어 있는 Access Token 의 유효성 검사를 한 뒤 유효하다면 메인 화면으로, 유효하지 않다면 Refresh Token 으로 Access Token 을 받아온 뒤 메인 화면으로 전환한다.

### 5.2.3 채팅방 검색

- 1) 원하는 키워드를 입력하여 서버에 보낸다.
- 2) DB 에서 채팅방의 이름, 관심분야를 키워드로 조회한다.
- 3) 조회한 데이터들을 클라이언트로 보내 화면에 표시한다.

### 5.2.4 채팅방 생성

- 1) 만들고 싶은 채팅방의 대분류와 소분류, 이름을 입력하여 서버에 보낸다.
- 2) 받은 데이터들로 채팅방 데이터를 만들어서 DB 에 저장한다.
- 3) DB 에 성공적으로 저장이 되었다면 채팅방 목록에 추가한다.

### 5.2.5 채팅방 추천

- 1) 채팅방 추천 버튼을 누른다.
- 2) 해당 사용자의 이메일 정보를 서버에 보내 사용자의 관심분야들을 DB 에서 조회한다.

- 3) 관심분야가 존재하지 않다면 전체 채팅방 목록에서 랜덤으로 채팅방 하나를 반환한다. 관심분야가 존재한다면 관심분야를 키워드로 하여 채팅방을 조회한 뒤 그 중 랜덤으로 하나를 반환한다.
- 4) 클라이언트에서 받은 채팅방을 화면 하단에 표시한다.

#### 5.2.6 채팅

- 1) 채팅방에 입장하면 소켓이 연결되어 소켓 ID 가 부여된다. 또한 로컬 DB 에서 저장되어 있는 채팅 기록을 가져와 화면에 표시한다.
- 2) 사용자가 채팅을 입력하면 서버와 소켓통신으로 데이터가 전송되며 로컬 DB 에 채팅 기록을 저장한 후, 화면에 표시한다.
- 3) 서버에서 받은 데이터를 같은 방 내의 다른 사용자들에게 전송한다.
- 4) 다른 사용자가 받은 데이터는 로컬 DB 에 채팅 기록을 저장한 후, 화면에 표시한다.

#### 5.2.7 자동번역

- 1) 소켓통신으로 받은 채팅메세지를 파파고 언어감지 API 를 활용하여 해당 언어가 무엇인지 판단한다.
- 2) 해당 언어가 사용자의 언어와 다를 경우 파파고 NMT 번역 API 를 활용하여 사용자의 언어로 번역해준다.

#### 5.2.8 사진 업로드

- 1) 사진을 선택하여 FormData 에 저장한 뒤, 서버에 전송한다.
- 2) 서버에서 multer 모듈을 활용하여 사진을 받고, 폴더에 저장한다.
- 3) 저장한 사진 이름을 소켓 통신으로 같은 방 내의 모든 사용자들에게 전송한다.
- 4) 클라이언트에서 받은 사진 이름을 로컬 DB 에 저장한 뒤, 사진을 화면에 표시해준다

#### 5.2.9 팝 퀴즈

- 1) 일정 주기가 되면 DB 에서 모든 채팅방의 채팅방 ID 와 관심분야를 조회한다.
- 2) 각 채팅방의 관심분야에 맞는 팝 퀴즈를 조회한다.
- 3) 조회된 팝 퀴즈 중 랜덤으로 하나를 각 채팅방에 소켓 통신으로 전송한다.
- 4) 클라이언트에서 받은 팝 퀴즈(질문, 답) 데이터를 화면에 표시한다.
- 5) 사용자가 답을 입력하면 정답이 맞는지 검사를 한다.
- 6) 정답이 맞다면 정답 결과를 서버에 전송한다.
- 7) 서버에서는 각 채팅방에서의 정답자 수를 카운트하고 있고, 최초 정답자가 나오면 소켓 통신으로 방 내의 전체 사용자에게 정답이라는 메세지를 보낸다.

- 8) 최초 정답자가 나오면 다른 사용자가 정답을 맞혀도 정답 메시지를 보내지 않는다.

#### 5.2.10 언어 교환

- 1) 사용자가 배우고 싶은 언어와 방 이름을 입력하여 언어 교환 방 생성 버튼을 누르고 서버에 해당 데이터를 전송한다.
- 2) 서버에서 받은 데이터를 DB 에 저장한다.
- 3) 사용자가 배우고 싶은 언어를 검색한다.
- 4) 사용자의 언어와 배우고 싶은 언어를 변수로 가진 방이 존재한다면 해당 방 목록을 화면에 표시한다.
- 5) 사용자가 원하는 방을 클릭하여 입장하면 1:1 로 상대방의 언어를 사용하며 대화를 할 수 있다.

#### 5.2.11 프로필 변경

- 1) 프로필 사진, 관심분야, 언어 중 변경하고자 하는 프로필을 선택한다.
- 2) 변경하려는 정보를 입력하여 확인 버튼을 누르면 해당 데이터가 서버에 전송된다.
- 3) 서버에서는 해당 데이터를 DB 에 저장한 뒤, 성공 여부를 반환한다.
- 4) 성공적으로 변경되었다면, 로컬 DB 에 변경된 정보를 저장한다.

#### 5.2.12 로그아웃

- 1) 로그아웃 버튼을 누른다.
- 2) 로컬 DB 에 토큰 정보를 삭제하며 로그인 화면으로 돌아간다.

#### 5.2.13 국제협력처 정보 확인

- 1) 사용자는 국제협력처 탭에서 다양한 동아리, 공지사항 등의 정보를 확인할 수 있다.
- 2) More info 버튼을 누르면 해당 정보가 공지되어 있는 링크로 연결된다.
- 3) Apply 버튼을 누르면 해당 공지의 관계자에게 메일을 보낼 수 있다.

#### 5.2.14 국제협력처 정보 등록

- 1) 국제협력처 관계자는 정보 등록 홈페이지에 접속한다.
- 2) 동아리 홍보나 다양한 공지사항의 정보를 등록 요청한다.
- 3) 요청한 정보는 서버에 보내지게 되며, DB 에 저장된다. DB 에 저장할 때 isAccept 라는 boolean type 의 변수를 false 로 하여 저장된다.

#### 5.2.15 국제협력처 정보 승인

- 1) 관리자는 관리자 홈페이지에 접속한다.

- 2) 등록을 요청한 정보의 유효성을 체크한다.
- 3) 등록할 정보의 승인 버튼을 누른다.
- 4) 등록된 정보는 서버로 보내지게 되며, DB 의 isAccept 변수를 true 로 수정한다.

### 5.3 테스트 시나리오

기능	테스트	결과
회원가입	이메일은 정상적으로 전송되는가	20 번의 테스트 결과 평균 전송 속도 2500ms
	인증번호 입력 시 체크는 잘 되는가	이상 없음
	이미 가입된 이메일을 입력했을 경우 중복을 체크하는가	이상 없음
	유효하지 않은 이메일을 입력했을 경우 오류 체크를 하는가	이상 없음
	이메일이 체크되지 않은 경우 다음 페이지로 넘어갈 수 없는가	이상 없음
	이메일이 체크된 경우 다음 페이지로 정상적으로 넘어가는가	이상 없음
	비밀번호 자릿수가 유효하지 않는 경우 오류 체크를 하는가	이상 없음
	재입력한 비밀번호가 맞지 않는 경우 오류 체크를 하는가	이상 없음
	닉네임의 자릿수는 유효하지 않는 경우 오류 체크를 하는가	이상 없음
	관심분야 선택은 잘 되는가	이상 없음
로그인	이메일과 비밀번호가 유효한 경우 정상적으로 로그인 되는가	이상 없음
	이메일이 유효하지 않는 경우 오류 체크를 하는가	이상 없음
	비밀번호가 유효하지 않는 경우 오류 체크를 하는가	이상 없음
	정상적으로 로그인이 되면 메인 화면으로 넘어가지는가	20 번의 테스트 결과 평균 전송 속도 700ms
	재 접속 시 자동 로그인이 되는가	이상 없음
채팅방 검색	검색 버튼을 누르면 검색 창이 나오는가	이상 없음
	검색 창에 입력은 정상적으로 되는가	이상 없음
	검색어 입력 후 확인 버튼을 누르면 팝업 창이 뜨는가	이상 없음
	팝업 창에는 검색어와 관련된 결과가 나오는가	이상 없음
	검색된 채팅방을 누르면 채팅방 목록에 추가되는가	이상 없음

채팅방 생성	생성 버튼을 누르면 생성 창이 나오는가	이상 없음
	대분류, 소분류 선택은 잘 되는가	이상 없음
	채팅방 이름 입력은 잘 되는가	이상 없음
	채팅방 생성 시 채팅방 목록에 추가되는가	이상 없음
채팅방 추천	추천 버튼을 누르면 추천된 채팅방이 화면에 표시되는가	이상 없음
	관심분야를 선택하지 않으면 랜덤으로 추천되는가	이상 없음
	추천된 채팅방을 선택하면 채팅방 목록에 추가되는가	이상 없음
	관심분야에 해당하는 채팅방이 없으면 랜덤으로 추천되는가	이상 없음
채팅	채팅 입력 시 화면에 잘 표시되는가	이상 없음
	다른 사용자의 채팅은 잘 수신되는가	이상 없음
	채팅에 지연 시간은 없는가	20 번의 테스트 결과 평균 지연 속도 200ms
	채팅 메시지가 없으면 전송 버튼이 안 눌리는가	이상 없음
	채팅방 입장 시 기존 채팅 내역이 호출되는가	이상 없음
자동번역	채팅 수신 시 사용자의 언어로 번역되는가	이상 없음
	사용자의 언어와 같은 채팅 수신 시 번역되지 않는가	이상 없음
	채팅 메시지를 클릭 시 원본 메시지를 보여주는가	이상 없음
사진 업로드	사진이 정상적으로 업로드가 되는가	이상 없음
	다른 사용자가 전송한 사진을 확인할 수 있는가	20 번의 테스트 결과 평균 전송 속도 1500ms
	사진 메시지를 클릭하면 전체 화면으로 볼 수 있는가	이상 없음
팝 퀴즈	팝 퀴즈는 정상적으로 수신 되는가	이상 없음
	채팅방의 관심분야에 맞는 팝 퀴즈가 수신 되는가	이상 없음
	팝 퀴즈 메시지 클릭 시 답 입력란이 보여지는가	이상 없음
	정답에 틀린 경우 안내 메시지가 뜨는가	이상 없음
	최초로 정답을 맞출 경우 안내 메시지가 뜨는가	이상 없음
	최초 정답자 이후에 정답을 맞출 경우 안내 메시지가 뜨는가	이상 없음
언어 교환	언어 교환 방 생성 시 본인의 언어는 선택할 수 없는가	이상 없음
	방 생성은 정상적으로 동작하는가	이상 없음
	언어 교환 방 검색 시 본인의 언어는 선택할 수 없는가	이상 없음
	방 검색 시 방 목록은 정상적으로 보여지는가	이상 없음
	채팅 시 자동 번역 기능은 비활성화 되어 있는가	이상 없음

	채팅 메시지 클릭 시 본인의 언어로 번역이 되는가	이상 없음
프로필 변경	프로필 사진 변경 시 사진 선택은 잘 되는가	이상 없음
	프로필 사진 변경 시 변경된 사진으로 적용이 되는가	이상 없음
	프로필 사진 변경 후 재 접속 시 변경된 사진으로 적용이 되는가	이상 없음
	관심분야 선택은 잘 되는가	이상 없음
	관심분야 변경 시 반영이 잘 되는가	이상 없음
	사용 언어 선택은 잘 되는가	이상 없음
	사용 언어 변경 시 반영이 잘 되는가	이상 없음
로그아웃	로그아웃 시 로그인 화면으로 넘어가지는가	이상 없음
국제협력처 정보 확인	국제협력처 정보는 확인할 수 있는가	이상 없음
	국제협력처 사진 클릭 시 전체 화면으로 보여지는가	이상 없음
	More info 버튼 클릭 시 해당 사이트로 이동되는가	이상 없음
	Apply 버튼 클릭 시 관계자에게 이메일을 보낼 수 있는가	이상 없음
국제협력처 정보 등록	새로운 정보를 등록할 수 있는가	이상 없음
	새로운 정보 등록 시 안내 메시지가 팝업으로 보여지는가	이상 없음
국제협력처 정보 승인	관리자 로그인 시 신청한 정보들의 목록이 보여지는가	이상 없음
	신청된 정보의 승인/거절 버튼을 누르면 안내 메시지가 보여지는가	이상 없음
	신청된 정보의 승인/거절 버튼을 누르면 목록에서 삭제되는가	이상 없음
	신청된 정보의 승인 버튼을 누르면 국제협력처 정보가 업데이트 되는가	이상 없음
	업데이트 된 정보를 사용자들이 확인할 수 있는가	이상 없음

표 8 테스트 시나리오



#### 5.4 매뉴얼/예제

- 사용자는 회원가입 버튼을 누른다.
- 학교 이메일로 인증을 한다.
- 개인 정보 및 관심분야를 입력해 회원가입을 한다.
- 이메일과 비밀번호를 입력해 로그인을 한다.
- 프로필 사진을 변경한다.
- 원하는 관심분야를 선택하여 채팅방을 생성한다.
- 원하는 검색어로 채팅방을 검색한다.
- 채팅방을 추천 받는다.
- 채팅방에 입장한다.
- 다른 사용자들과 채팅을 한다.
- 원하는 사진을 선택하여 다른 사용자들과 공유한다.
- 팝 퀴즈가 올라오면 정답을 맞힌다.
- 언어 교환 방을 생성 및 검색한다.
- 언어 교환 방에 입장하여 다른 사용자와 채팅을 하며 상대방의 언어를 배운다.
- 국제협력처 정보를 확인한다.
- 원하는 정보를 선택하여 상세 정보를 확인한다.
- 참여를 원하는 정보를 선택하여 관계자에게 이메일을 보낸다.
- 로그아웃 한다.

## 6 과제 관리

### 6.1 BD 차트 기반 진행과정 설명

Iteration	Term	Theme	Description
Iteration 1	2019.09.26 ~ 10.09	관련 지식 습득	개발 시작 전 관련 지식에 대한 습득 Mockup 제작하기
Iteration 2	2019.10.10 ~ 10.21	기초 필수 기능 구현	Frontend 와 Backend 연동 로그인, 회원가입, 시작, 메인 화면 구현
Iteration 3	2019.10.22 ~ 11.03	채팅 관련 필수 기능 구현	채팅방 검색, 추천, 입장, 생성 구현 채팅방 구현
Iteration 4	2019.11.04 ~ 11.22	채팅 관련 부가 기능 구현	팝 퀴즈, 언어 교환, 사진 업로드, 개인 프로필 수정 기능 구현 iteration 3 의 지연 과제 구현
Iteration 5	2019.11.23 ~ 12.11	부가 기능 구현 미구현 과제 구현	로그아웃, 탈퇴, 서포터즈 UI, 광고 기능 구현 그 외 지연 과제 구현

표 9 Iteration 상세 정보

- 최초로 BD 차트를 구성할 때, 중간고사, 각자의 개인 일정을 충분히 고려하지 못하였기에, iteration3 부터 지연이 발생하였다. 이전 iteration 에서 해결하지 못한 구현 과제들은, 다음 iteration 으로 연기하여 진행하였다.
- 이러한 과제 지연 발생으로 인해, iteration 3~4 의 Burndown Chart 의 그래프가 이상적이지 못한 방향으로 그려졌다.

### 6.2 협업 방식/도구 설명

분류	협업 도구	협업 방안
소스 변경 관리	Github	프로젝트를 진행하며 변경되는 소스는 Github 을 사용하여 관리된다. 현재 팀 레파지토리를 개설해 놓은 상태이며, 코드 수정 시 commit 후 레파지토리에 push 하는 방식으로 관리한다.
코드 리뷰	Github	Github 에 새로운 코드를 반영하기 전, Pull requests 를 활용하여 서로의 코드를 확인 후 리뷰를 해준다. 서로의 코드를 확인해주며 Clean Code 작성을 목표로 한다.
이슈 관리	Google Drive	프로젝트를 진행하면서 생기는 이슈에 대해선 구글 공유드라이브에 기록 후 팀장 주도하에 팀원과 의논하여 담당자를 선출, 이슈를 처리한다.
빌드 배포 방식	Expo	Expo 의 빌드 툴을 이용하여 빌드하고, 수동으로 배포한다.

표 10 협업 방안/도구

### 6.3 구성원간 업무 분장

이름	개발분야	비개발분야
정상현	API 정의 및 구현 보조 Backend-Frontend 연동 Frontend, Backend 전체 이해	Burn down chart 작성 및 보완 보고서 작성 모든 문서 최종 검토 팀원 작업 분배
이윤구	Frontend 전반적인 기능 구현 Web UI 구현	보고서 작성 및 지속적인 수정 정보 수집
홍동기	App 내 UI 구현 및 개선 로컬 DB 관리 Frontend-Backend 연동	보고서 검토 및 취합 PPT 작성 및 발표
김정환	API 정의 및 구현 외부 API 연동 서버 DB 관리	보고서 작성 및 지속적인 수정 정보수집

표 11 업무 분장

### 6.4 비용

#### 6.4.1 SW 구축

- Yumi 는 React Native, MongoDB, SQLite, Express, Node js, Naver Cloud Platform, GitHub, Slack, Expo, Postman, Robo 3T, Naver Developers, Socket IO 를 사용하여 개발할 예정이며 모두 오픈소스이기 때문에 별도의 SW 나 HW 구매는 필요하지 않을 것으로 예상된다.

#### 6.4.2 유지/보수

- Yumi 는 1 대의 Naver Cloud Standard(2vCPU, 4GB Mem, 50GB Disk) 서버에서 실행되고 있으며 AB(Apache Bench)를 통해 부하 테스트를 시행한 결과 동시 요청 수 800 건에 대한 평균 처리속도 0.55 초를 기록하였다. Yumi 의 목표인 채팅 전송 지연시간 0.55ms 미만, 2020 년 내 동시 접속자 500 명 달성 시 현재의 서버로도 충분히 트래픽을 감당할 수 있다.

#### 6.4.3 증가율

- 2020 년 내 동시 접속자 500 명 달성 후에도 꾸준히 Yumi 사용자는 증가할 것으로 예상하며 동시 접속자 800 명 돌파 시 목표 채팅 전송 지연시간인 0.55ms 이상이 걸릴 것으로 예상하므로 이후에 추가적인 서버 증설이 요구될 수 있을 거라 예측됨.

따라서 Yumi 는 개발 시 서버 유지보수 외의 추가적인 비용이 들지 않으며 초기 규모는 연평균 91 만 원에서 점점 증가할 것으로 예측된다.

## 7 결론 – Post Mortem

### 7.1 프로젝트 진행(실행) 과정에서 잘 된 점 그 이유

- 프로젝트 진행 과정에서 채팅에 대한 기본적인 구현이 잘 되었다고 생각한다. Socket.io 를 사용한다는 점을 이용하여 쉽게 양방향 통신을 구현할 수 있었고 그 과정에서 API 에 대한 이해도가 증가했다.
- Frontend 2 명과 Backend 2 명으로 나누어서 개발이 진행된 부분도 프로젝트가 잘 진행될 수 있었던 부분 중 하나라고 생각한다. 혼자 한 부분을 맡아 진행하면 조언을 구할 사람이 없지만, 파트너가 있기에 서로 도움을 주고받을 수 있었고, 넷이 혼재되어 프로젝트를 진행했을 경우엔 상대적으로 전문성이 떨어져 구현 속도가 더디어졌을 것이다. 이런 부분을 감안하고 진행하였기에 긍정적으로 프로젝트를 진행할 수 있었다.
- 유학생들에게 큰 영향을 주지 못할 것 같다는 평가를 많이 받았었다. "서비스가 구체적이지 않다.", "이 서비스가 유학생에게 특화된 서비스인가?"라는 평가를 들었고, 해결방안을 모색하여 1:1 언어 교환 서비스, 서포터즈 정보를 쉽게 획득할 수 있는 탭을 추가로 구현하여 외국인 유학생들에게 실질적인 도움을 줄 수 있는 서비스를 만들 수 있었다.

### 7.2 프로젝트 진행 과정에서 아쉬운 점, 개선 아이디어

- 외국인들이 정보를 얻는 것에 대해 힘들다는 점을 알게 되어 외국인에게 교내 공지사항을 영어로 보여주는 서비스 또한 생각하였다. 크롤링을 통하여 공지사항을 가져오고 자동 번역 기능을 연동하여 번역된 공지사항을 자동으로 보여줄 수 있는 서비스를 만든다면 좀 더 효율적이고 관리가 쉬울 것이다.
- 자동 번역을 NAVER API 를 사용하여 구현하였는데 조금 더 공부를 하여 자동 번역에 대한 시스템을 이해하고 구현한다면 의존도를 낮추고, 완성도는 높은 프로젝트로 거듭날 수 있을 것이다.

### 7.3 본 과제 결과물의 발전 방향

- 점점 많은 외국인들이 우리나라 대학교로 유학을 온다는 통계가 있다. 앞으로 지속적으로 많아질 외국인 유학생들에게 편리한 서비스들을 제공한다는 점에서 우리의 서비스는 확실한 고객을 보유하고 있다고 생각한다.
- 서포터즈와 각 학교의 국제협력처를 통하여 확실한 홍보 기회를 잡을 수 있다고 생각한다.
- Yumi 의 서비스를 통하여 서포터즈와 국제협력처 또한 편리하게 정보 제공이 가능하다고 생각한다.

## 7.4 기타 소감

### 7.4.1 정상현

- 저는 본 전공이 수학과로써 17년 2학기부터 복수전공을 하기 시작했습니다. 졸업 전 프로젝트 경험을 쌓기 위하여 SW 캡스톤디자인 과목에 신청하게 되었습니다. 한 학기 동안 하나의 프로젝트를 하며 새로운 개념들을 많이 공부하게 되었고, 팀장을 맡아 책임감 있게 프로젝트 진행을 이끌어 나갈 수 있었습니다. 팀원들과 함께 협력하며 의견을 나누고 개발을 진행하는 등 팀 프로젝트를 진행하며 협동심을 배울 수 있었습니다. 또한 수업시간에 리팩토링, 클린 코드 작성, TDD 등 개발자가 갖춰야 할 자질에 대해 학습하며 예비 개발자로서 한 단계 발전한 듯 합니다. 전공 선택 과목인 SW 캡스톤디자인을 수강신청 할 때 많은 걱정을 했었지만, 좋은 경험을 할 수 있어서 탁월한 선택이었다고 생각합니다. 감사합니다.

### 7.4.2 이윤구

- SW 캡스톤디자인 과목을 수강하여 한 학기 동안 프로젝트를 진행하며 많은 것을 느끼고 경험할 수 있었습니다. 기획, 설계, 개발, 배포까지의 과정을 팀원들과 협업하여 진행하면서, 의견 제시, 비판 수용, 합의를 하며 팀워크를 키워 나간 것 같습니다. 또한 Agile 방법론을 적용하여 작은 기능들을 만들고 그것들을 하나로 합쳐 완성되어가는 결과물을 보며 큰 만족감을 느꼈습니다. 이 과목을 통해 예상했던 것 보다 더 많은 것을 배울 수 있었고 특히 제가 담당했던 프론트엔드 분야와 협업 툴 사용에서 큰 발전을 이룬 것 같습니다. 향후 개발자 인생의 큰 밑거름이 될 경험을 쌓은 것 같습니다. 감사합니다.

### 7.4.3 홍동기

- 저는 16학번 커리큘럼을 따라가는 복수전공자이기에 SW 캡스톤디자인이 졸업 요건이 아니라 듣지 않아도 문제 없었습니다. 처음에는 “그래도 졸업하기 전에 뭐 하나 해보자.” 라는 무겁지 않은 생각으로 수강을 신청했지만, 해보고 정말 많은 점을 느꼈습니다. 실전과 이론은 다른 점이 너무나도 많다는 점, 짧은 기간에는 느낄 수 없는 팀 별 대형 프로젝트에서 배울 수 있는 팀워크, 내 시야 밖의 부분을 고쳐 내기 위해서는 타인의 비판을 수용해야 한다는 점, 그리고 팀원과의 관계가 딱딱하지 않아야 장난 속에서 발굴되는 창의성 등 이번 4개월간 이전 3년 6개월의 대학에서 배운 과정들을 다듬을 수 있었습니다. 감사합니다.

### 7.4.4 김정환

- 소프트웨어학과에 다니면서 많은 전공 지식을 쌓았다고 생각했습니다. 하지만 실제로 큰 프로젝트를 하는 것은 처음이었고 경험을 해 보니 제가 아는 지식이 부족하다는 것을 다시 느낄 수 있었습니다. 그리고 새롭게 웹과 앱에 대해서 공부를 해보고 결과물을 만들어 내 보니 좀 더 전공에 관심을 가지게 된 것 같습니다. 한 학기 동안 공부로 얻는 것 보다 많은 학습을 하고 적용해보는 중요한 경험을 얻는 값진 시간이었습니다. 감사합니다.

## 8 참고 문헌

1. 셸리 파워즈 저/안재우 역, 제대로 배우는 Node.js 프로그래밍, 비제이퍼블릭
2. 바니 아이젠먼 저/이종은 역, 빠른 모바일 앱 개발을 위한 React Native 2/E:자바스크립트로 만드는 네이티브 모바일 앱 개발 가이드, 인사이트
3. <https://socket.io/docs/server-api/>
4. <https://docs.mongodb.com/manual/reference/database-references/>

## 9 Appendix

### 9.1 Glossary

Term	Definition and Information
서포터즈	서포터즈란 해외 유학생과 각 대학교, 외국인 동아리, 유학생 서포터즈와 같이 다양한 외국인 유학생을 지원하는 집단을 가리키는 용도로써 사용된다.
User	User 는 외국인 한국에 온 외국인 유학생이며 유미를 통해 채팅, 팝 퀴즈, 장소 추천 기능을 사용하는 자를 지칭한다.
팝 퀴즈	입장한 채팅 방 내에서 채팅봇을 통해 채팅 방 주제와 관련된 문제를 채팅방에 올리고 제일 먼저 정답을 맞힌 사용자에게 정답을 맞혔다는 메시지를 출력하는 기능을 지칭한다.
채팅방 추천	사용자의 취미, 사는 지역, 다니는 학교의 정보를 종합하여 사용자에게 적절한 주제로 대화하는 채팅 방을 생성 또는 생성 되어있는 방에 초대 메시지를 보내는 기능을 지칭한다.

표 12 Glossary

## 10 해당 프로젝트 github 링크

<https://github.com/AjouCodelink/Yumi>