

# 2021 한이음 공모전 개 발 보 고 서

2021. 10. 17

프로젝트명	국문	빅데이터&챗봇 기반 도서 추천
	영문	Big Data & Chatbot-based book recommendation
작 품 명	레코북(Recommend Books)	

# 요 약 본

작품 정보		
프로젝트명	국문	빅데이터&챗봇 기반 도서 추천
	영문	Big Data & Chatbot-based book recommendation
작품명	레고북(Recommend Books)	
작품 소개	'레고북'은 쉬운 접근이 어려운 기존 서비스의 불편함을 해소하고 맞춤 추천을 통해 사용자가 만족할 만한 서비스를 제공하기 위해 개발된 챗봇 시스템이다. '레고북'은 사용자와의 대화를 통해 데이터를 수집하며 이를 통해 도서를 추천한다.	
작품 구성도		
작품의 개발배경 및 필요성	챗봇 시장이 계속해서 성장하고 꾸준히 수요가 증가하고 있음에도 불구하고, 사용자와의 대화가 활성화되어 있거나 도서 정보를 쉽게 확인할 수 있는 챗봇은 드물다. 이러한 상황을 고려하여 사용자가 원하는 정보를 전달하는 챗봇을 개발하게 되었다.	
작품의 특징점	<ol style="list-style-type: none"> <li>1. 접근성이 좋은 카카오톡 등 챗봇 애플리케이션을 통해 쉽게 대화를 시작할 수 있다.</li> <li>2. 입력받은 정보와 이전 평가한 작품들을 기준으로, 시중에 존재하는 수많은 책들을 대상으로 추천을 진행하기 때문에 사용자 맞춤형 추천이 가능하다.</li> <li>3. 위시리스트, 나의 책장, 알림 등으로 독서 습관을 생성할 수 있다.</li> </ol>	
작품 기능	<ol style="list-style-type: none"> <li>1. 채팅 기능 : 사용자에게 질문을 받고 답변을 반환</li> <li>2. 사용자 정보 조회 : 선호하는 작가, 장르, 읽은 책, 위시리스트를 입력받고 조회</li> <li>3. 도서 정보 조회 : 원하는 도서의 정보를 조회</li> <li>4. 도서 추천: 제목, 작가, 장르 기반 추천을 진행</li> <li>5. 사용자 맞춤 도서 추천 : 사용자 정보를 기반으로 사용자 맞춤 추천을 진행</li> <li>6.. 통계 조회 : 문답을 통한 읽은 책/읽고 싶은 책/좋아하는 작가/장르 조회</li> <li>7. 오늘의 도서 조회 : 매일 사용자 맞춤 혹은 랜덤 책을 조회할 수 있다.</li> </ol>	
작품의 기대효과 및 활용분야	<ol style="list-style-type: none"> <li>1. 시간 단축: 도서 검색에 필요한 시간 단축</li> <li>2. 정보 획득: 베스트셀러에 추천된 도서 외 본인의 취향에 맞는 도서 추천</li> <li>3. 훌륭한 접근성 : 접근성이 높은 채팅 어플리케이션과 챗봇을 연동</li> </ol>	

# 본 문

## I. 작품 개요

※ 평가항목 : 기획력 (필요성, 차별성)

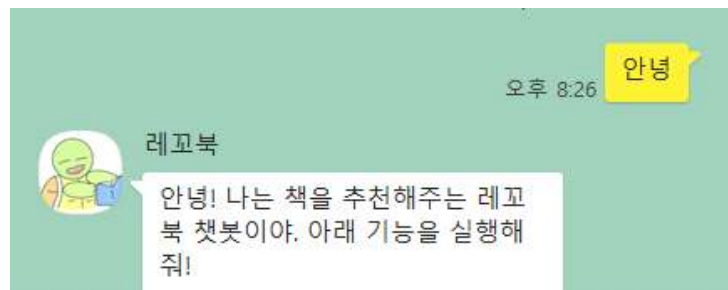
### 1. 작품 소개

#### 1) 기획 의도

독서를 시작하거나 새로운 도서를 원하는 사람들을 위한 맞춤형 도서 추천 서비스

#### 2) 작품 내용

- 카카오톡, 네이버톡톡과 같은 채팅 애플리케이션으로 챗봇과의 대화가 가능
- 사용자는 자유롭게 문답을 나눌 수 있으며 사용자의 의도를 분석해서 답변을 출력



#### 3) 시스템 구축

- Naver Cloud Platform의 서버, 데이터베이스, 챗봇 서비스를 이용
- open API와 서점 웹사이트 크롤링을 통해 정보를 가져오고 DB를 구성
- 이용자가 입력한 정보(선호 작가, 주제 등)와 대화 기록을 이용해 추천 알고리즘을 구성하고 적합한 도서를 추천

## 2. 작품의 개발 배경 및 필요성

#### 1) 도서 정보 접근성의 부족

한 해에도 수 만권의 책이 출간되고, 현재 전자책 시장도 매우 활성화가 되어있다. 그럼에도 불구하고 도서에 대한 정보가 부족해서 새롭게 독서에 관심을 가지기 어려워하는 사람이 많다. 우리는 접근성이 좋은 서비스를 개발해서 도서 정보에 대한 접근성을 높이는 것을 목표로 한다.

#### 2) 개인 맞춤형 추천 시스템의 부족

- 책은 주로 베스트 셀러, 방송 등의 추천을 통해 접하는 경우가 많지만 이러한

책은 항상 본인의 취향에 맞는다는 보장이 없다.

- 넷플릭스에서 사용자가 시청하는 영상의 3분의 2가 고객에게 추천된 영화이며, 아마존의 경우 매출의 35%가 추천을 통해 발생하고 있다. 하지만 도서 추천은 아직 소비층에서 활성화 되어 있지 않다. 좋은 성능의 도서 추천 시스템이 만들어지고 대중화된다면 이용자의 만족도가 높아지고 도서 소비가 촉진될 것이다.

### 3) 비대면 소통의 필요성 및 챗봇 서비스의 성장

- 코로나19로 인해 대면 소통이 어려워지면서 비대면 소통의 필요성이 대두되고 있다. 이러한 점에서 비대면으로도 쉽고 빠르게 접할 수 있는 챗봇 수요가 급증하고 있다.
- 글로벌 시장조사업체 마켓앤드마켓에 따르면 챗봇 시장의 연평균 성장률은 29.7%에 달한다. 국내 챗봇 시장 또한 연평균 51%씩 성장하고 있다는 분석이 나오고 있다. 실제로 국내외 IT, 금융, 통신, 소매, 제조 등 다양한 분야의 기업들은 챗봇을 도입하고 있다.
- 현재 웹이나 앱 기능 내 도서 추천 시스템을 구현한 곳은 있으나 챗봇을 통한 도서 추천 시스템을 활용하는 곳은 찾아보기 힘들다.

## 3. 작품의 특징 및 장점

### 1) 챗봇 서비스

- 챗봇을 통해 자유로운 문답이 가능하다.
- 기존 서비스는 도서 추천 서비스를 자체 앱 혹은 웹으로만 제공하는 경우가 아서 서비스의 존재도 모르거나 다운로드가 귀찮아서 사용하지 못하는 경우가 있었다. 우리의 서비스는 접근성이 쉬운 여러 채팅 플랫폼에 챗봇 연동을 시도해서, 서비스의 접근성을 높일 것이다.

### 2) 사용자 맞춤형 도서 추천

사용자의 답변 내용을 기반으로 맞춤 추천 기능을 제공한다.

### 3) 사용자 도서 책장 관리

사용자는 읽은 책, 읽고 싶은 책, 선호 장르, 선호 작가를 등록하여 자신만의 도서 책장 관리를 할 수 있다.

### 4) 다양하고 색다른 데이터 활용

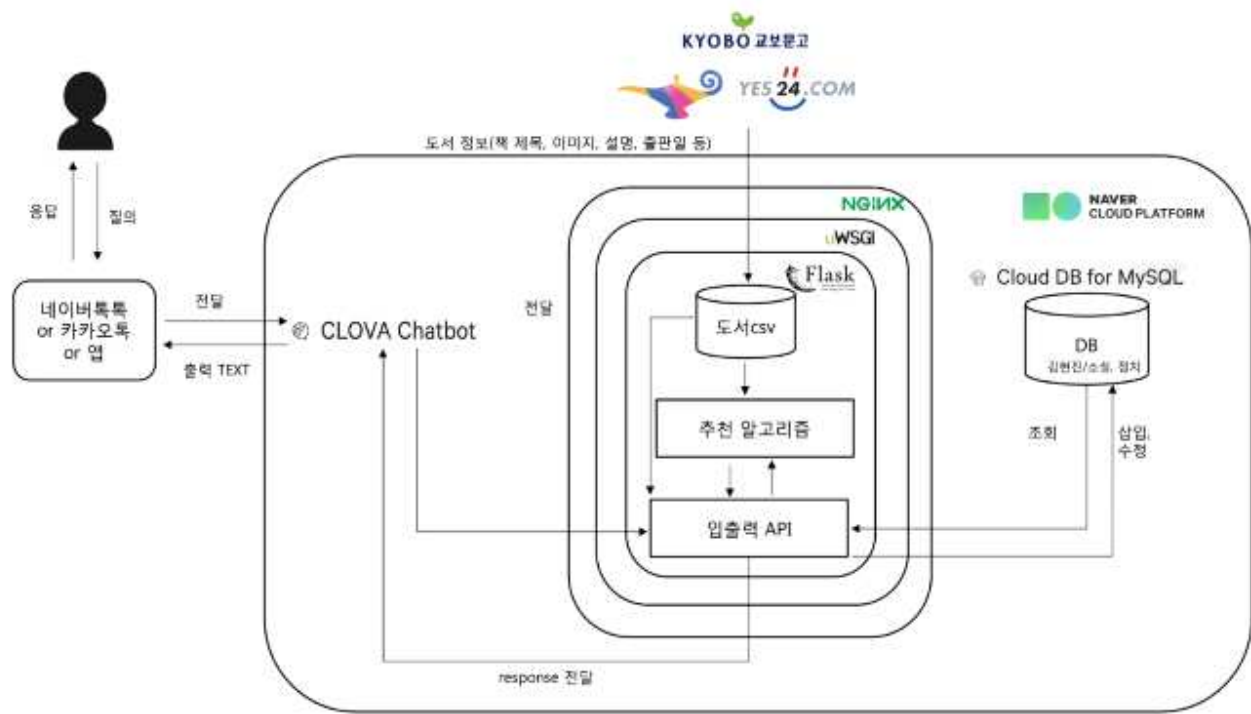
- 기존 서비스는 기업 내의 데이터만을 사용하는 경우가 많다.
- 우리의 서비스는 여러 사이트에서 데이터를 가져와서 활용한다. 다양하고 색다른 데이터를 활용해서 추천하는 것을 목표로 한다.

# II. 작품 내용

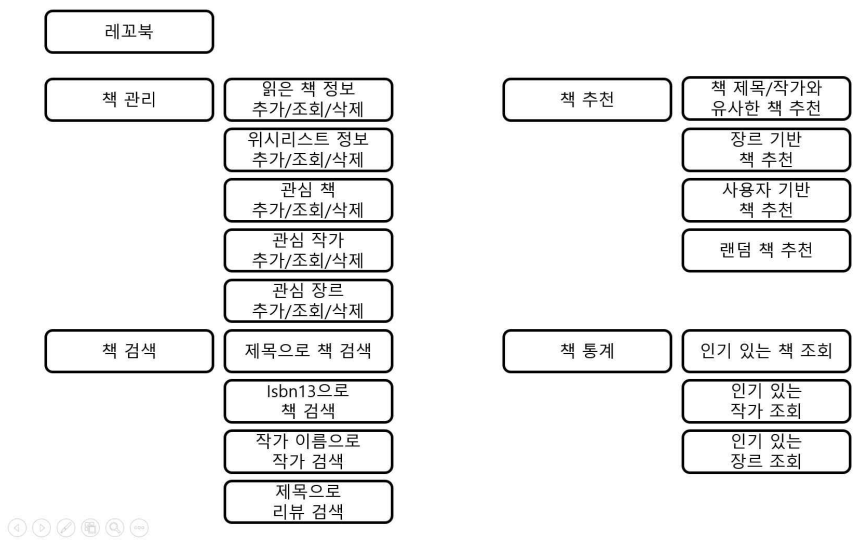
※ 평가항목 : 기술력 (기능구체성, 난이도, 완성도)

## 1. 작품 구성도

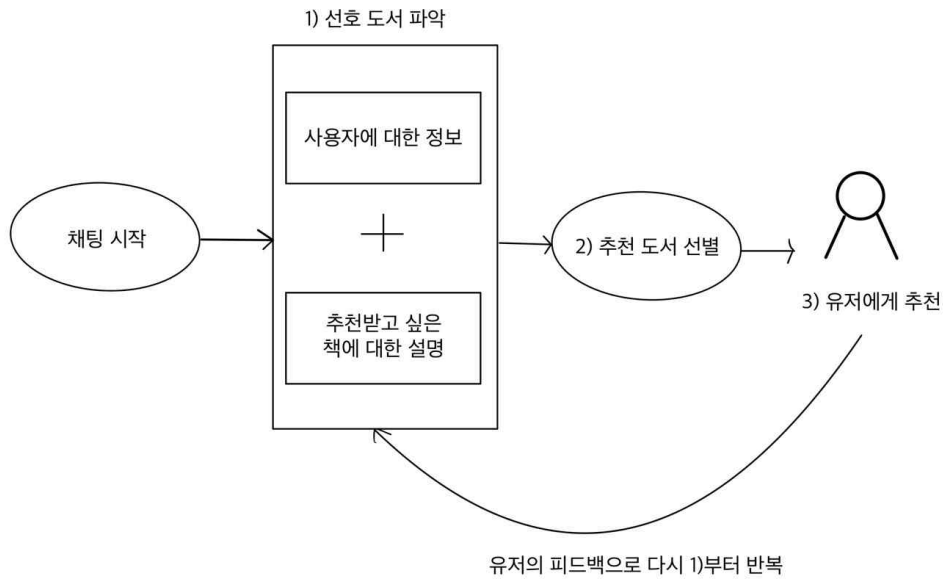
### 1) 서비스 구성도



### 2) 메뉴구성도

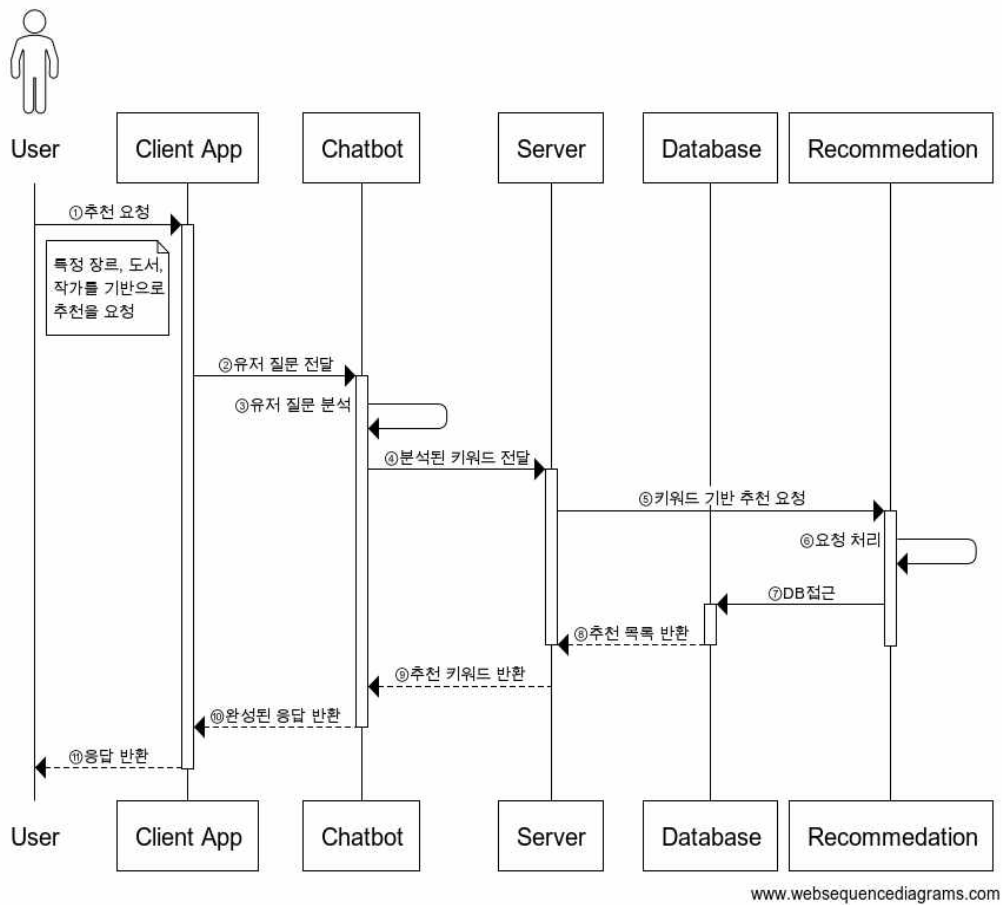


### 3) 서비스 흐름도



### 4) 기능 처리도

기능 처리도(흐름도)



## 2. 작품 기능

### 1) 전체 기능 목록

구분	기능	설명	현재 진척도(%)
S/W	채팅 서비스	궁금한 것을 물어볼 수 있고, 질문에 대한 답변을 받는다. 먼저 채팅을 받을 수도 있다.	100
	알림	채팅이 올 때마다 알림을 받을 수 있다.	100
	도서 등록	관리자는 open API와 크롤링을 통해 도서 정보 및 리뷰 정보를 등록할 수 있다. 자동화 코드를 이용한 자동 도서 등록 또한 가능하다.	100
	유저 정보 조회	채팅을 보낸 사용자의 정보를 조회할 수 있다	100
	도서 정보 조회	도서 정보를 조회할 수 있다.	100
	읽은 책 등록/조회	문답을 통해 사용자가 읽은 책을 등록하고 조회할 수 있다.	100
	위시리스트 등록/조회	문답을 통해 사용자가 읽고 싶은 책의 위시리스트를 만들고 조회할 수 있다.	100
	선호 작가 등록/조회	문답을 통해 선호 작가를 등록하고 조회할 수 있다.	100
	통계조회	문답을 통해 사람들이 많이 읽은 책/읽고 싶은 책/좋아하는 작가/좋아하는 장르를 조회할 수 있다.	100
	사용자 기반 추천	사용자 정보를 기반으로 사용자 맞춤 추천이 가능하다.	100
	오늘의 도서 추천	매일 사용자 맞춤 혹은 랜덤 책을 조회할 수 있다.	100
H/W	스마트폰	스마트폰으로 사용자가 서비스를 사용할 수 있다.	100
	데스크탑	데스크탑으로 사용자가 서비스를 사용할 수 있다.	100
	클라우드 서버	클라우드 서버에서 Server, Database, Chatbot을 제공받아 SW 기능 개발에 활용한다.	100

## 2) S/W 주요 기능

기능	설명	작품실물사진
채팅	채팅을 보내거나, 질문에 대한 답변을 받거나 또는 먼저 채팅을 받을 수 있는 기능	
알림	채팅이 올 때 알림을 받는 기능	
도서 등록	관리자가 open API, 크롤링, 자동화 코드를 통해 도서 정보를 등록할 수 있는 기능	<pre> #-*- coding:utf-8 -*- import requests import json from db_model.mysql import conn_mysqlldb from decouple import config # from flask import IntegrityError # from pymysql import PYMYSQL_DUPLICATE_ERROR  class Aladin:     """     openAPI 및 크롤링 -&gt; 도서 수집 자동화     """ </pre>



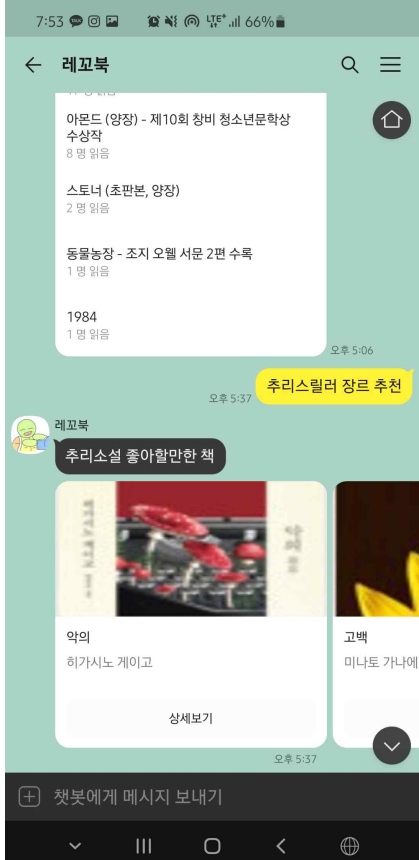

<p>유저 정보 조회</p>	<p>채팅을 보낸 사용자의 정보를 조회하는 기능</p>	
<p>도서 정보 조회</p>	<p>사용자가 요청한 책의 정보를 조회하는 기능</p>	
<p>읽은 책 등록/조회</p>	<p>사용자가 읽은 책을 등록하거나 조회하는 기능</p>	

<p>위시리스트 등록/조회</p>	<p>사용자가 읽고 싶은 책의 위시리스트를 등록하거나 조회하는 기능</p>	
<p>선호 작가 등록/조회</p>	<p>선호 작가를 등록하거나 조회하는 기능</p>	
<p>선호 장르 등록/조회</p>	<p>선호 장르를 등록하거나 조회하는 기능</p>	
<p>통계 조회</p>	<p>사람들이 많이 읽은 책, 많이 읽고 싶은 책, 선호하는 작가, 선호하는 장르를 조회하는 기능</p>	

<p>랜덤 도서 추천</p>	<p>사용자가 랜덤으로 추천을 요청하면, DB에 등록된 책에 저장되어 있는 책중 랜덤으로 사용자에게 반환하는 기능</p>	
<p>베스트셀러 추천</p>	<p>베스트셀러 추천을 요청시 DB에 등록된 책 중 BestRank가 0 이상인 책을 필터링하여 책을 추천하는 기능</p>	
<p>책 제목 기반 추천</p>	<p>유저가 특정 책과 유사한 책에 대한 추천을 요청할 시, DB에 등록된 상위 유사도를 가진 5개의 책을 반환하는 기능</p>	

<p>장르 기반 추천</p>	<p>유저가 특정 장르(키워드)에 대해 추천을 요청할 경우, 해당 장르 임베딩 벡터와 가장 유사도가 큰 5개의 책을 반환하는 기능</p>	 <p>오후 9:02</p> <p>추리 스릴러 장르 추천해줘</p> <p>레꼬북</p> <p>추리소설 좋아할만한 책</p> <p>악의 히가시노 게이고</p> <p>상세보기</p> <p>오후 9:02</p>
<p>사용자 맞춤 추천</p>	<p>사용자가 자신이 좋아할만한 책 추천을 요청하는 경우, 사용자가 등록한 선호책, 읽은책을 조회하여 그와 가장 유사한 5개의 책을 반환하는 기능</p>	 <p>오후 8:23</p> <p>내가 좋아할만한 책 알려줘</p> <p>레꼬북</p> <p>좋아할만한 책</p> <p>개밥바라기벌 황석영</p> <p>상세보기</p> <p>오후 8:23</p>
<p>오늘의 도서 조회</p>	<p>오늘의 도서를 조회할 수 있는 기능</p>	 <p>오후 7:50</p> <p>오늘의 도서는 뭐야</p> <p>레꼬북</p> <p>면역에 관하여</p> <p>면역에 관하여 미국의 논픽션 작가 올라 비스의 세 번...</p> <p>작가 올라 비스, 김명남</p> <p>출판사 열린책들</p>

### 3) H/W 주요 기능

기능/부품	설명	작품 실물 사진
스마트폰	스마트폰으로 사용자가 챗봇 기능을 사용하고, 도서 웹사이트를 확인 수 있다.	
데스크탑	데스크탑으로 사용자가 챗봇 기능을 사용하고, 도서 웹사이트를 확인 수 있다.	

클라우드 서버	클라우드 서버에서 Server, Database, Chatbot을 제공받아 SW 기능 개발에 활용한다.	
---------	---	--

### 3. 주요 적용 기술

구분	기술	주요 기능	주요 적용 기술
S/W	Python 및 SQL 프로그래밍, Flask 서버 위에서 실행	크롤링	Selenium을 이용해 도서 웹사이트를 크롤링해서 도서 데이터를 수집한다. 수집한 데이터는 DB에 저장한다. 크롤링 자동화 코드를 작성한다.
		Open API의 데이터 저장	도서 웹사이트 open API를 활용해 원하는 도서 정보를 불러오고 DB에 저장해서 활용한다. 이 때 원하는 정보만 가져오고, 중복된 값은 가져오지 않는 자동화 코드를 작성한다.
		API	도서 정보와 사용자 정보를 삽입, 수정, 조회, 삭제할 수 있는 API를 제작한다. 이 때, pymysql, sqlalchemy 라이브러리를 활용해서 SQL문으로 DB에 접근한다.
			konlpy로 토큰화된 리뷰의 단어들을 Word2Vec, Sentence-BERT, Doc2Vec, FastText을 통해 벡터값으로 변환하고, 이들의 평균으로 문서 벡터를 구하여 입력값으로 받은 도서와 유사한 도서를 찾아주는 도서 추천 시스템을 만든다. Numpy, Pandas, Scikit-Learn, Gensim 등의 라이브러리를 활용한다.
			사용자 대화 의도를 분석하기 위해 Clova 챗봇, 카카오 i오픈빌더 서비스와 연동하는 코드를 작성한다.
			챗봇 어플리케이션과 연동하는 코드를 작성한다.

	Cloud DB (Cloud DB for MySQL)	데이터 보관	사용자가 입력한 정보와 크롤링이나 open API로 얻은 도서 관련 정보를 저장한다.
	Cloud Server (Ubuntu, Nginx, uWSGI)	서버 환경 구성	NaverCloudPlatform에서 Ubuntu 18.04 서버를 사용한다. 이때 Flask 웹 프레임워크, Nginx 웹서버, 둘 사이 통신을 돕는 uWSGI로 서버 환경을 구성한다. Nginx를 사용함으로써 클라이언트의 요청을 분산시켜서 처리할 수 있고, 서버 환경 설정을 더욱 수월하게 할 수 있다.
	Cloud Chatbot	대화 의도 파악	Clova 챗봇 서비스, 카카오 i오픈빌더 서비스를 이용해서 이용자의 의도를 파악하고 엔터티를 서버에 제공한다.
	Object Storage	Object image url 저장	리사이즈된 이미지를 Body값에 제공하는 URL을 생성하기 위해, NaverCloudPlatform의 Object Storage 서비스를 이용하여 객체 별 url을 만들고 이 url을 body값에 담음으로써 사용자에게 이미지를 제공할 수 있다.
	Chatbot Application	챗봇 애플리케이션과 연동	카카오톡의 연동을 위한 kakao i오픈빌더 Builder, 네이버 앱 연동을 위한 네이버톡톡 파트너센터 등도 함께 활용한다.

#### 4. 작품 개발 환경

구분		상세내용
S/W 개발환경	OS	Window, Mac, Ubuntu
	개발환경(IDE)	Visual Studio Code, Jupyter Notebook, Pycharm
	개발도구	Git, Postman
	개발언어	Python
	기타사항	-
H/W 구성장비	디바이스	스마트폰, 데스크탑, 클라우드
	센서	-
	통신	-
	언어	-
	기타사항	-
프로젝트 관리환경	형상관리	Gitlab, Notion
	의사소통관리	Notion
	기타사항	-

## 5. 기타 사항 [본문에서 표현되지 못한 작품의 가치(Value)] 및 제작 노력

### 1) 차별화된 성능

추천시스템에는 크게 협업 필터링과 콘텐츠 기반 추천 두 가지가 존재한다. 넷플릭스나 당근 마켓 등 개인화된 추천시스템을 제공하는 기업의 경우 두 가지를 조합하여 사용하기 때문에 성능이 높은 것으로 유명하다. 그러나 현재 시장에 존재하는 도서 추천 챗봇 서비스는 대부분 통계적 수치에 기반한 Rule-based 모델이나 랜덤 추천, 그리고 베스트셀러 추천 기능밖에 없기 때문에 챗봇을 통한 도서 추천의 만족도는 떨어질 수밖에 없다. 그러나 우리 작품은 책 정보와 리뷰 정보를 이루고 있는 자연어 처리를 위해 임베딩에 특화된 여러 모델을 사용함으로써 일반화된 추천 성능을 향상시켰다. 또한 추천을 위해 유저정보도 활용함으로써 모두에게 일관적인 추천이 아닌 다양한 추천을 해줌으로서 개인화된 추천을 실현하였다.

### 2) 응답 속도

대화가 실시간으로 이루어지는 챗봇 시스템의 특성 상, 아무리 성능이 좋아도 추천 처리가 길어져 응답 속도가 느려진다면 이용자가 충분히 만족하지 못할 것이다. 우리는 이러한 문제를 인지하고 응답 속도를 높이기 위해 두 가지 방식을 채택하였다. 첫 번째로는 모델 측면에서의 성능 향상, 두 번째로는 배포 측면에서의 성능 향상이다. 모델 측면에서의 성능 향상을 이루기 위해, 문장 임베딩에 특화되면서도, 기존의 BERT보다 약 46,000배 빠르다고 알려진 Sentence-BERT 모델을 임베딩에 사용하였다. 또한 한국어를 사용하기 때문에 한국어로 Pre-train 된 Sentence-BERT 모델을 사용하여 임베딩의 성능을 향상했다. 배포 측면에서의 성능을 향상시키기 위해서 우리는 추천을 배치 단위로 학습하고 배치 단위로 배포하는 방식을 선택하였다. 배치 단위로 학습하는 방법은 먼저 DB에 있는 모든 정보를 모델로 학습 시켜 임베딩 벡터로 변환 시킨 뒤, 모든 정보에 대해 코사인 유사도를 구하는 것이다. 배치 단위로 배포하는 것은 해당 유사도를 책 별로 유사한 상위 5개의 책 ISBN 리스트를 DB에 업데이트 함으로써, 유저가 특정 책에 대한 추천을 요청할 시, 단순히 DB에 저장된 리스트만 조회하여 반환하게 된다. 따라서 매우 빠른 챗봇의 응답 속도를 받을 수 있었다.

### 3) 보안

- 프로젝트의 중요한 정보인 open API key값, db 정보 등을 python decouple 모듈로 처리해서 외부에 노출되지 않도록 했다.
- 챗봇 애플리케이션에서 가져온 암호화된 사용자 정보를 저장하되 노출되지 않도록 했다.



#### 4) 알고리즘의 창작성

- 기존의 추천 알고리즘은 유저에 대한 데이터가 있어야만 추천이 가능하다. 하지만 Word2Vec, Sentence-BERT, Doc2Vec, FastText을 이용한 레코북의 추천 알고리즘은 유저에 대한 데이터가 전혀 없어도 요구사항에 따라 책 추천이 가능하다. 이는 추천 알고리즘이 유저 데이터 기반이 아니라 DB에 저장된 리뷰 데이터 기반이기 때문이다.
- 그렇지만 유저에 대한 데이터를 사용하지 않는 것은 아니다. 유저가 서비스를 이용하면서 쌓인 정보를 기반으로 추천 시스템을 동작시켜 개인화된 추천을 가능하게 한다.

#### 5) 데이터와 프로그램의 가치

베스트셀러에서부터 시작해, 스테디셀러와 고전까지의 순서로 데이터를 수집하며 사용자의 필요에 더 부합하는 도서추천이 가능하도록 하였다. 방대한 양의 데이터로 인해, 자신에게 가장 필요하고 원하는 데이터를 얻기란 쉽지 않다. 우리 프로그램은 개인화 맞춤 추천을 통해 불필요한 정보는 최대한 제외하고 핵심적인 내용만을 바로 얻어볼 수 있게 설계되었다.

#### 6) 작품을 통한 가치창출

종이책보다는 e-book을 많이 활용하는 시대의 흐름에 비해, 도서 관리를 할 수 있는 시스템은 찾아보기 힘들다. 우리의 서비스는 도서 관리를 보다 쉽게 할 수 있는 편의를 제공해준다.

### III. 프로젝트 수행 내용

※ 평가항목 : 수행능력 (문제해결능력, 수행충실성)

#### 1. 프로젝트 수행일정

프로젝트 기간 (ICT멘토링 사이트 기준)		2021.03.30. ~ 2021.11.30.										
구분	추진내용	프로젝트 기간										
		3월	4월	5월	6월	7월	8월	9월	10월	11월	12월	
계획	주기적인 미팅을 통한 아이디어 구체화 및 관련 스터디 계획											
분석	챗봇, 추천시스템, 한국어 임베딩, 크롤링 기술 조사 및 학습											
설계	사용자 시나리오 설정, 시스템구조 정의											
	DB schema 설계											
개발	도서 정보 크롤링											
	사용자에게 입력받은 정보를 처리할 서버 구축											
	사용자/도서 기반 추천시스템 구축											
테스트	기능 테스트											
종료	한이음 공모전 출품											

#### 2. 프로젝트 추진 과정에서의 문제점 및 해결방안

##### 1) 프로젝트 관리 측면

###### - 작업 환경

코로나로 인해 대면 회의가 어려운 상황이었다. 하지만 개발 성과를 내기 위해서는 효율적이 소통이 필요했기 때문에, 매주 시간을 정해서 회의를 진행했고 Google Meets 등 온라인 회의 플랫폼을 활용했다.

###### - 코드 리뷰

기능이 추가될 수록 코드가 복잡해져 가독성이 떨어지는 문제가 발생했다. 이를 해결하고자, 매주 회의시간 각 팀원이 gitlab의 merge request를 공유하며 모든 팀원들이 함께 코드 리뷰를 진행하였다. 함수가 길어지면 중복 되는 부분을 다른 함수로 재정의하는 등, 리팩토링을 통해 코드의 경량화 및 직관성을 높였다.

###### - 프로젝트 관리

프로젝트 진행 초반에는 회의록만을 작성했지만, 점점 프로젝트 관리의 필요성을 느끼게 되었다.

그래서 각자 프로젝트를 어떻게 진행하고 있는지를 확인할 수 있는 로드맵 페이지, 문제가 생길 때마다 작성하는 이슈 페이지, 코드 변경사항을 공유하는 코드 형상 관리 페이지를 만들어서 작성하도록 했다. 또한, 회의에서의 중요한 사항은 잊지 않도록 따로 페이지를 만들어서 공유할 수 있도록 했다.

## 2) 작품 개발 측면

### - 챗봇 builder 선정

프로젝트 초기에는 Google Cloud Platform에서 제공하는 dialogflow 챗봇 서비스를 이용하려 했다. 하지만 한국어 임베딩의 정확도가 비교적 낮으며, 한이음에서 제공하는 Cloud 서비스가 아님을 고려해서 Naver Cloud Platform의 Clova 챗봇 서비스를 이용하기로 결정했다. 그리고 대중적으로 사용하는 채팅 서비스는 카카오톡이기 때문에, 우선적으로 카카오 I오픈빌더를 사용하여 카카오톡 챗봇 레코북을 구현하였다.

### - open API 데이터 수집

open API에서 도서 정보를 가져올 때 문제집과 같은 사용자에게 추천하기 좋지 않고 추천 시스템의 성능을 떨어뜨리는 데이터를 가져오는 문제가 있었다. 또한, 같은 데이터가 중복해서 들어오는 문제가 있었다.

open API에서 가져오는 책의 카테고리들 시/소설만 가져오도록 했고, 도서마다 할당된 고유한 값인 isbn13을 DB에서 primary key로 설정해서 데이터의 중복을 피했다.

### - 크롤링 IP 차단

알라딘 사이트를 크롤링하기 위해서는 동적으로 크롤링하는 Selenium의 사용이 필수적인데, 이를 자동화하게 될 경우 서버에서 크롤링하려는 PC의 IP가 막힌다. 또한, 알라딘은 웹 구조가 복잡하여 모든 페이지에 통용되는 크롤러를 만들기 어려웠다. 그래서 알라딘 openAPI를 통해 도서 정보를 수집하였고, yes24 사이트에서 리뷰 데이터를 수집하였다.

### - 추천 시스템의 시간적 비용

추천 모델을 동적으로 돌릴 경우 시간적 비용이 많이 소모되는 문제가 있었지만 추천을 진행할 때마다 모델을 만드는 것이 아니라 하나의 모델을 만들고 그 결과로 나온 임베딩 값으로 한 도서의 문서 벡터값을 구해 이를 DB에 저장하여 불러오는 식으로 수정하여 문제를 해결했다. 이 방법을 사용할 경우 사용자에게 입력값을 받고 동적으로 모델을 만들어 추천 시스템을 돌리는 것보다 월등히 빠르게 동작하는 것이 가능하다.

- 한국어 자연어 처리의 정확도

한국어에 대한 자연어 처리의 정확도가 낮은 문제가 있었는데 konlpy에 사용자 사전에 DB에 저장된 작가, 책 제목 그리고 기타 자주 쓰는 용어를 등록해둔 뒤에 모델을 만들어 토큰화하는 과정에서 중요한 의미를 지닌 단어들을 훼손하는 일을 줄이는 것으로 해결하였다.

### 3. 프로젝트를 통해 배우거나 느낀 점

#### 1) 프로젝트 관리 측면

- 프로젝트를 진행하면서 성과를 내기 위한 프로젝트 진행 방법을 알 수 있었고, 실무적인 개발 프로세스를 경험해볼 수 있었다.
- 여러 협업 툴을 사용함으로써 다른 사람들과 프로젝트를 진행할 때 능률적으로 진행하는 방법에 대해 배울 수 있었다.

#### 2) 작품 개발 측면

- 백엔드 시스템과 맞물려 유기적으로 프로그램이 작동하기 위해서는 어떤 프로세스들이 필요한지 알 수 있었다.
- 챗봇 시스템이 어떠한 방식으로 작동하는지 전체적인 흐름을 이해하게 되었다.
- 모델을 구축할 때 다양한 접근방식을 시도해보면서 끊임없이 도전하는 자세를 배우게 되었고, 다양한 방면으로 생각해볼 수 있는 사고력을 기르게 되었다.
- 원하는 대로 작동하면서 최대한 깔끔하고 정리된 코드를 작성하는 법을 배우게 되었다.
- 단어 임베딩이라는 새로운 기술을 배우고 직접 적용해보면서 새로운 정보 표현 방식을 익히게 되었다.
- 같은 결과를 내는 코드라도 더 효율적이고 신속하게 돌아갈 수 있도록 하는 방법을 탐구할 수 있게 되었다.

## IV. 작품의 기대효과 및 활용분야

### ※ 평가항목 : 기획력 (활용가능성)

#### 1. 작품의 기대효과

기존의 도서 추천 챗봇 서비스는 많지 않으며 그마저도 교보문고, 인터파크도서 등 대기업에서 제공하는 서비스가 대부분이다. 대기업에서 부가 서비스로 제공하는 챗봇 서비스는 사용하기 불편한 점이 많다. 자체 어플에서만 제공하는 경우가 많아 접근성이 좋지 않고, 도서 추천뿐만 아니라 관련 홍보 메시지, 공지사항 등의 정보를 함께 제공해서 알림이 과도 사용자에게 불편함을 느낀다. 그리고, 베스트셀러 혹은 프로모션 위주의 추천을 해준다. 프로젝트의 도서추천 챗봇 서비스는 기존 챗봇 애플리케이션과의 연동을 통해 사용자들의 접근성을 높이고, 알림톡 서비스를 통해 먼저 책 추천 기능을 제공하되 필요 없는 정보는 제공하지 않을 것이며, 빅데이터 분석을 통해 베스트셀러에 추천된 도서 외 본인의 취향에 맞는 도서를 추천해줄 것이다. 이러한 기능은 사용자가 더욱 편리하게 서비스를 이용할 수 있도록 도움을 준다.

최근 안양시에서 만든 카카오톡 도서정보 챗봇 서비스는 서비스를 개시한 이후 이용자 수가 급증하는 등 놀라운 효과를 보였다. 도서 추천 챗봇 서비스를 출시한다면 평소에 책을 읽지 않는 사람부터 책 매니아까지 새로운 책을 읽어볼 기회를 제공할 수 있을 것이다.

#### 2. 작품의 활용분야

- 사용자가 직접 입력한 정보를 기반으로 추천이 진행되므로 도서에 흥미를 갖게 할 수 있다.
- 코로나로 인해 집에서 즐길 새로운 취미 생활을 찾는 사용자들에게 책을 추천 해주며 사람들이 독서에 입문하는 것을 돕는다.
- 사용자가 도서 검색에 필요한 시간을 단축할 수 있다. 이는 독서에 투자할 수 있는 시간을 확보해준다.
- 베스트셀러에 추천된 도서 외 본인의 취향에 맞는 도서를 추천해주며 새로운 책을 접할 기회를 늘린다.
- 숨어있는 명작들을 발견해서 추천해주며 기성 작가들의 새로운 독자층을 만들어줌으로써 독자와 작가 모두 만족할 수 있도록 한다.