



국민대학교  
소프트웨어융합대학  
소프트웨어학부

# 캡스톤 디자인 I

## 종합설계 프로젝트

프로젝트 명	항만 소규모 물류 견적 중계 O2O 플랫폼 개발
팀 명	팀 배타버스 (BAETAVERSE) / 캡스톤 팀 43
문서 제목	팀 43 수행결과보고서

Version	1.2
Date	2022-05-23

팀원	한정택 (팀장)	20171718
	김현우	20181603
	서보성	20171629
	신준호	20191616

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>다학제간캡스톤디자인I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	항만 소규모 물류 견적 중계 O2O 플랫폼 개발	
	<b>팀 명</b>	팀 배타버스 (BAETAVERSE) / 캡스톤 팀 43	
	Confidential Restricted	Version 1.0	2022-05-20


#### CONFIDENTIALITY/SECURITY WARNING

이 문서에 포함되어 있는 정보는 국민대학교 소프트웨어융합대학 소프트웨어학부 및 소프트웨어학부 개설 교과목 캡스톤 디자인 수강 학생 중 프로젝트 "항만 소규모 물류 견적 중계 O2O 플랫폼 개발"을 수행하는 팀 "배타버스 (BAETAVERSE)" 팀원들의 자산입니다. 국민대학교 소프트웨어학부 및 팀 "배타버스 (BAETAVERSE)"의 팀원들의 서면 허락 없이 사용되거나, 재가공 될 수 없습니다.

## 문서 정보 / 수정 내역

<b>Filename</b>	팀43-수행결과보고서.docx
<b>원안작성자</b>	한정택, 김현우, 서보성, 신준호
<b>수정작성자</b>	한정택, 김현우, 서보성, 신준호

수정날짜	대표수정자	Revision	추가/수정 항목	내 용
2022-05-20	한정택	1.0	최초 작성	
2022-05-22	김현우	1.1	추가 작성	Back-End 개발 진행 사항 구체화
2022-05-23	김현우	1.2	수정	보고서 전체 오타자 검사

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>다학제간캡스톤디자인I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	항만 소규모 물류 건적 중계 O2O 플랫폼 개발	
	<b>팀 명</b>	팀 배타버스 (BAETAVERSE) / 캡스톤 팀 43	
	Confidential Restricted	Version 1.0	2022-05-20

## 목 차

1	개요 .....	5
1.1	프로젝트 개요 .....	5
1.1.1	프로젝트 소개 .....	5
1.1.2	프로젝트의 전체적 구조 .....	5
1.1.3	개발 내용 .....	6
1.1.3.1.	Back-End (HS Code Recommendation System) .....	6
1.1.3.2.	Back-End (REST-API Server) .....	6
1.1.3.3.	Back-End (Database) .....	6
1.1.3.4.	Front-End (Apple Platform Client Application) .....	6
1.2	추진 배경 및 필요성 .....	7
1.2.1	소규모 수출 화주 소비자 관점 .....	7
1.2.2	소규모 물류 운송 서비스 공급자 관점 .....	7
1.2.3	항만 소규모 물류 건적 중계 O2O 플랫폼 .....	7
2	개발 내용 및 결과물 .....	8
2.1	목표 .....	8
2.1.1	소규모 수출 화주 소비자 관점 .....	8
2.1.2	소규모 물류 운송 서비스 공급자 관점 .....	8
2.1.3	O2O 플랫폼 서비스 개발자 관점 .....	8
2.2	연구/개발 내용 및 결과물 .....	9
2.2.1	연구/개발 내용 .....	9
2.2.1.1.	개발 수행 일정 .....	9
2.2.1.2.	계획 및 분석 진행 세부 내용 .....	10
2.2.1.3.	설계 진행 세부 내용 .....	10
2.2.1.4.	개발 진행 세부 내용 .....	10
2.2.1.5.	테스트 진행 세부 내용 .....	10
2.2.2	시스템 기능 요구사항 .....	11
2.2.3	시스템 비기능(품질) 요구사항 .....	13
2.2.4	활용/개발된 기술 .....	19
2.2.5	현실적 제한 요소 및 그 해결 방안 .....	19
2.2.6	결과물 목록 .....	20
2.3	기대효과 및 활용방안 .....	21
2.3.1	소규모 수출 화주 소비자 관점 .....	21
2.3.2	소규모 물류 운송 서비스 공급자 관점 .....	21

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>다학제간캡스톤디자인I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	항만 소규모 물류 건적 중계 O2O 플랫폼 개발	
	<b>팀 명</b>	팀 배타버스 (BAETAVERSE) / 캡스톤 팀 43	
	Confidential Restricted	Version 1.0	2022-05-20

2.3.3	대한민국 국가적 성장 관점 .....	21
3	자기평가 .....	21
4	참고 문헌 .....	22
5	부록 .....	22
5.1	운영자 및 사용자 매뉴얼 .....	22
5.2	프로젝트 기술 문서 .....	22
5.3	테스트 케이스 .....	23

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>다학제간캡스톤디자인I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	항만 소규모 물류 건적 중계 O2O 플랫폼 개발	
	<b>팀 명</b>	팀 배타버스 (BAETAVERSE) / 캡스톤 팀 43	
	Confidential Restricted	Version 1.0	2022-05-20

# 1 개요

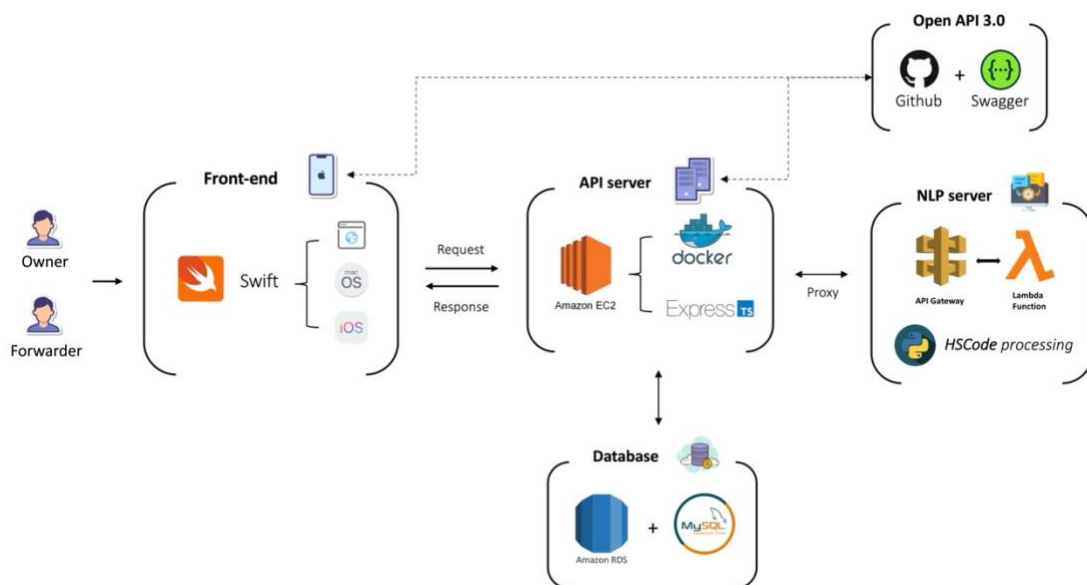
## 1.1 프로젝트 개요

### 1.1.1 프로젝트 소개

- 화주 고객과 물류 서비스 공급자 사이 소규모 화물 물류 운송 건적 중계 플랫폼

### 1.1.2 프로젝트의 전체적 구조

- Back-End (HS Code Recommendation System)
  - 입력한 화물에 적합한 HS Code 번호를 제안해주는 검색 시스템
  - AWS Lambda를 통한 마이크로 웹서비스로 운영
- Back-End (REST-API Server)
  - Front-End 애플리케이션에 REST-API를 통해 비즈니스 로직을 제공하는 서버 시스템
  - AWS EC2를 통한 웹서비스로 운영
- Back-End (Database)
  - 비즈니스 로직을 구성하기 위해 데이터를 관리하는 데이터베이스 시스템
  - AWS RDS를 통한 MySQL DBMS 서버 운영
- Front-End (Apple Platform Client Application)
  - 플랫폼 사용자에게 비즈니스 서비스에 대한 사용자 인터페이스를 제공하는 시스템
  - iOS 등의 Apple 플랫폼에서 사용될 수 있도록 운영



 <b>국민대학교</b> <b>소프트웨어학부</b> <b>다학제간캡스톤디자인I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	항만 소규모 물류 건적 중계 O2O 플랫폼 개발	
	<b>팀 명</b>	팀 배타버스 (BAETAVERSE) / 캡스톤 팀 43	
	Confidential Restricted	Version 1.0	2022-05-20

### 1.1.3 개발 내용

#### 1.1.3.1. Back-End (HS Code Recommendation System)

- 사용자가 입력한 화물 정보와 유사한 HS Code 번호를 제안해주는 검색 시스템을 개발
- Amazon Gateway 및 lambda 클라우드 서버리스 컴퓨팅을 통한 마이크로 서비스 운영
- 자연어 처리 기술 (NLP) 을 활용하여 Python 언어로 개발됨
- 플랫폼 이용자에게 HS Code 자동 완성 편의 기능을 제공함
- 서로 다른 유사한 한글 단어 입력에 적절한 HS Code 를 반환하도록 NLP 기술로 구현
- Word2Vec 단어 유사도 모델을 통해 한글 단어를 HS Code로 자동 완성되도록 동작

#### 1.1.3.2. Back-End (REST-API Server)

- Front-End 애플리케이션에 REST-API를 통해 비즈니스 로직을 제공하는 서버 시스템
- AWS EC2 클라우드 컴퓨팅 환경에서 docker 컨테이너를 통한 웹 서비스로 운영
- Server 프레임워크 Express.js 개발 환경에서 TypeScript를 통해 개발
- MVC 5계층 디자인 패턴 적용을 통해 서비스 전체 로직의 계층화 구조 개발
- TypeORM을 통해 서버 모델 객체와 데이터베이스 간의 데이터 타입을 규격화
- JWT 토큰을 활용한 사용자 인증, 건적 요청, 건적 작성, 리뷰 작성 비즈니스 로직 구현

#### 1.1.3.3. Back-End (Database)

- 비즈니스 로직을 구성하기 위해 사용자 데이터를 관리하는 데이터베이스 시스템 개발
- AWS Relational Database Service를 통해 MySQL DBMS로 시스템 설치 관리 및 운영
- 소비자와 공급자 사이 전반적인 서비스에 기반이 되는 데이터베이스 스키마 개발
- 데이터베이스 사용자 인증, 건적 요청서, 건적서 관리 CRUD SQL 쿼리 개발

#### 1.1.3.4. Front-End (Apple Platform Client Application)

- 플랫폼 사용자에게 비즈니스 서비스에 대한 사용자 인터페이스를 제공하는 시스템
- iOS, macOS 등의 Apple 플랫폼을 통해 사용자가 서비스를 사용할 수 있도록 구현
- 소비자와 공급자에게 알맞은 사용자 인터페이스가 제공될 수 있도록 어플을 분리 구현
- UIKit, SwiftUI 사용자 인터페이스 기술을 사용하여 Swift5 언어로 개발됨
- Mac Catalyst 기술 적용을 통해 macOS를 통해 동시 서비스되도록 구현
- 객체 지향적 설계를 도입하여 유지보수성을 고려한 네트워크 통신 코드 개발
- GitHub Action을 통한 테스트 및 통합 코드 관리 CI/CD 코드 개발

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>다학제간캡스톤디자인I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	항만 소규모 물류 견적 중계 O2O 플랫폼 개발	
	<b>팀 명</b>	팀 배타버스 (BAETAVERSE) / 캡스톤 팀 43	
	Confidential Restricted	Version 1.0	2022-05-20

## 1.2 추진 배경 및 필요성

### 1.2.1 소규모 수출 화주 소비자 관점

- 소비자는 화물의 주인(화주)으로 정의
- 소비자는 소규모 물류 서비스 사용자가 대상
- 소비자는 물류 수·출입 서비스 이용을 위해 개별 공급자에게 견적 요청을 보냄
- 소비자는 공급자의 선택지가 적어서 시간과 비용 측면에서 비효율적으로 서비스 이용
- 소비자는 화물 관세 물류 체계인 HS Code의 복잡성으로 인해 잦은 행정 오류를 경험

### 1.2.2 소규모 물류 운송 서비스 공급자 관점

- 공급자는 운송 주선 업자(포워더)로 정의
- 공급자는 소규모 물류의 포워딩 서비스 제공
- 공급자는 고객을 영입하기 위해 현장 영업에 많은 운영비를 지출
- 공급자는 물류 대란 사태로 인해 많은 컨테이너 운영 비용을 지출
- 공급자는 소비자에게 자사 서비스의 신뢰도를 증명하고 싶음
- 공급자는 물류 운송 행정 절차에서 발생할 수 있는 오류를 개선하고 싶음

### 1.2.3 항만 소규모 물류 견적 중계 O2O 플랫폼

- 소비자와 공급자 사이 물류비 견적 통합 비교 및 고객 관리 서비스를 제공하도록 기획
- 소비자(화물의 주인, 화주)에게 소규모 화물 물류비 견적 비교 서비스를 제공
- 공급자(포워더)에게 견적서 제공 및 고객 관리 서비스를 제공
- 소비자와 공급자에게 HS Code 추천 검색 서비스를 제공
- 소비자와 공급자 간의 견적 정보 비대칭성으로 인한 시간 및 비용 측면의 비효율성 개선
- 물류 운송 서비스의 행정 처리 도중 HS Code 기재 오류로 발생하는 행정 오류를 개선

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>다학제간캡스톤디자인I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	항만 소규모 물류 건적 중계 O2O 플랫폼 개발	
	<b>팀 명</b>	팀 배타버스 (BAETAVERSE) / 캡스톤 팀 43	
	Confidential Restricted	Version 1.0	2022-05-20

## 2 개발 내용 및 결과물

### 2.1 목표

#### 2.1.1 소규모 수출 화주 소비자 관점

- 소비자는 공급자에게 소규모 화물 물류 운송 건적 정보를 요청할 수 있음
- 소비자는 운송 화물에 대한 HS Code 추천 검색 기능을 사용할 수 있음
- 소비자는 공급자에게 육, 해, 공 상의 물류 운송 연계 정보를 요청할 수 있음
- 소비자는 공급자에게 인코텀즈 기준의 화물 인도 세부 조건을 규정할 수 있음
- 소비자는 여러 공급자가 제공한 건적 중 선호하는 건적을 선택할 수 있음
- 소비자는 선호하는 건적서를 제공한 공급자에게 세부 상담 요청을 할 수 있음
- 소비자는 공급자의 서비스 제공 사항에 리뷰를 남길 수 있음

#### 2.1.2 소규모 물류 운송 서비스 공급자 관점

- 공급자는 소비자를 영입 및 관리 할 수 있음
- 공급자는 소비자의 물류 건적 요청서 명단을 확인할 수 있음
- 공급자는 소비자의 건적 요청서를 선택할 수 있음
- 공급자는 선택한 건적 요청서에 응수할 수 있는 건적서 정보를 제시할 수 있음
- 공급자는 소비자의 상담 요청에 응답할 수 있음
- 공급자는 소비자가 동의한 건적서에 거래 확정 선언을 할 수 있음
- 공급자는 소비자가 제시한 리뷰를 확인하고 점수 통계를 관리할 수 있음

#### 2.1.3 O2O 플랫폼 서비스 개발자 관점

- 개발자는 프로젝트의 유지보수성을 고려하여 코드를 작성
- 개발자는 인터페이스 규격화를 통해 컴포넌트별 병렬적인 애자일 개발 진행
- 개발자는 사용자 이용 규모를 고려한 서비스 배치를 고려하여 프로젝트 설계
- 개발자는 클라우드 컴퓨팅 서비스를 이용하여 서비스 배포 및 운영할 수 있음
- 개발자는 기술 문서에 기반하여 구조 파악 및 커뮤니케이션 진행할 수 있음
- 개발자는 코드 형상 관리 규칙을 정해서 수정 사항을 빠르게 파악할 수 있음
- 개발자는 GitHub action을 통해 자동화 테스트 환경을 지원 받을 수 있음



 <b>국민대학교</b> <b>소프트웨어학부</b> <b>다학제간캡스톤디자인I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	항만 소규모 물류 건적 중계 O2O 플랫폼 개발	
	<b>팀 명</b>	팀 배타버스 (BAETAVERSE) / 캡스톤 팀 43	
	Confidential Restricted	Version 1.0	2022-05-20

## 2.2 연구/개발 내용 및 결과물

### 2.2.1 연구/개발 내용

#### 2.2.1.1. 개발 수행 일정

- 설계 및 개발 문서화 과정은 폭포수 모델 기반 수행
- Agile 방식의 프로젝트 개발 수행
- 주황색 셀은 현재 진행이 완료된 상태의 일정을 의미

구분	추진일정											
	3월				4월				5월			
	1주	2주	3주	4주	1주	2주	3주	4주	1주	2주	3주	4주
계획												
분석												
설계												
개발												
테스트												
종료												

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>다학제간캡스톤디자인I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	항만 소규모 물류 견적 중계 O2O 플랫폼 개발	
	<b>팀 명</b>	팀 배타버스 (BAETAVERSE) / 캡스톤 팀 43	
	Confidential Restricted	Version 1.0	2022-05-20

### 2.2.1.2. 계획 및 분석 진행 세부 내용

- 3월 1주 차 ~ 3월 3주 차까지 진행
- 화물 운송 도메인에 대한 지식 습득
- 화물 운송 도메인 분야의 소비자와 공급자에 대한 애로사항 파악
- 화물 운송 도메인 관련 종사자에 대한 인터뷰 수행
- 견적 중계 서비스에 대한 현실성 및 추진 가능성 분석 작업 진행

### 2.2.1.3. 설계 진행 세부 내용

- 3월 4주 차 ~ 4월 3주 차까지 진행
- 프로젝트 Use Case UML 스키마 작성
- 클라이언트 앱(소비자, 공급자)의 UI/UX 스토리보드 설계도 작성
- 데이터베이스 데이터 스키마 ERD 설계도 작성
- HS Code 자동 완성 시스템 데이터 정리
- 클라이언트, 서버 코드 설계도 작성

### 2.2.1.4. 개발 진행 세부 내용

- 4월 4주 차 ~ 5월 3주 차까지 진행
- 클라이언트 앱(소비자, 공급자)의 목업 프로그램 개발 진행
- 클라이언트 앱(소비자, 공급자)의 백엔드 API 연동 개발 진행
- HS Code 자동 완성 시스템 개발
- 데이터베이스 CRUD SQL 쿼리 개발
- REST-API 서버 비즈니스 로직 및 라우팅 기능 개발
- 서버 및 데이터베이스 환경 Amazon 서비스로 이관
- Docker 컨테이너 기반의 마이크로 서비스 아키텍처로 재구성

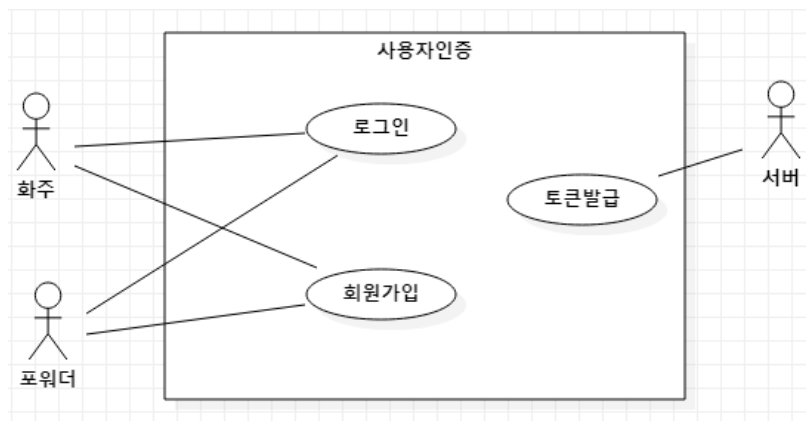
### 2.2.1.5. 테스트 진행 세부 내용

- 4월 4주 차 ~ 5월 3주 차까지 진행
- 클라이언트 앱(소비자, 공급자)의 백엔드 연동 테스트 코드 구현
- Swagger 활용 REST API 문서화 및 서버 테스트 기능 구현
- GitHub Action을 활용한 CI/CD 파이프라인 구축
- 클라이언트, 서버 코드 리팩토링

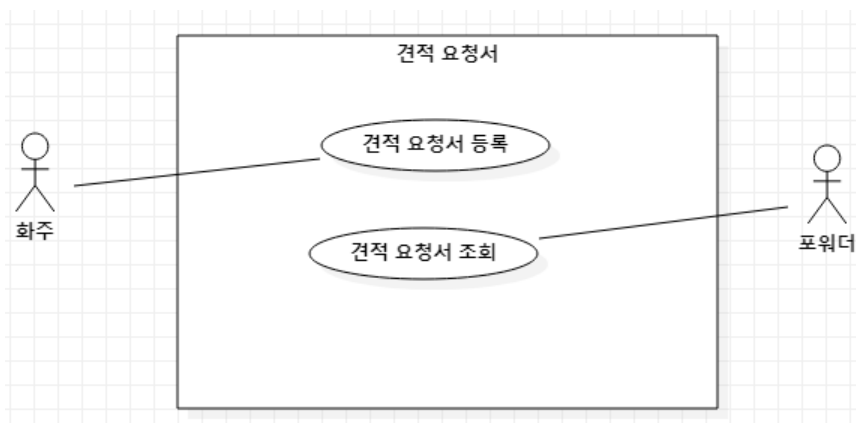
 <b>국민대학교</b> <b>소프트웨어학부</b> <b>다학제간캡스톤디자인I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	항만 소규모 물류 건적 중계 O2O 플랫폼 개발	
	<b>팀 명</b>	팀 배타버스 (BAETAVERSE) / 캡스톤 팀 43	
	Confidential Restricted	Version 1.0	2022-05-20

## 2.2.2 시스템 기능 요구사항

- Use Case 공통 전제 사항
  - 플랫폼의 소비자는 화주로 정의
  - 플랫폼의 공급자는 포워더로 정의
  - 클라이언트 앱은 소비자용, 공급자용으로 분리되어 제공
  - 서버는 클라이언트 앱이 전달하는 사용자 이벤트의 비즈니스 로직 처리
- 사용자 인증 기능
  - 소비자와 공급자는 클라이언트 앱을 통해 플랫폼에 회원으로 가입할 수 있음
  - 소비자와 공급자는 클라이언트 앱을 통해 플랫폼에 로그인할 수 있음
  - 웹 서버는 클라이언트 앱에 사용자를 식별할 수 있는 JWT 토큰을 발급함

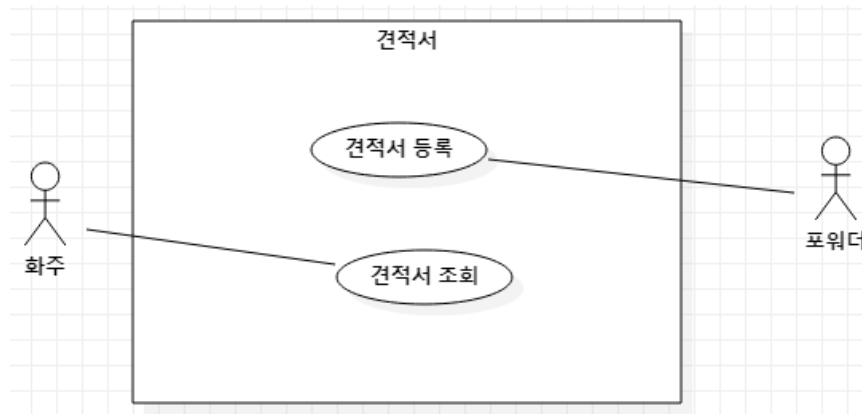


- 건적 요청서 기능
  - 소비자는 클라이언트 앱을 통해 공급자에게 건적 요청서를 전달할 수 있음
  - 공급자는 클라이언트 앱을 통해 소비자의 건적 요청서를 조회할 수 있음



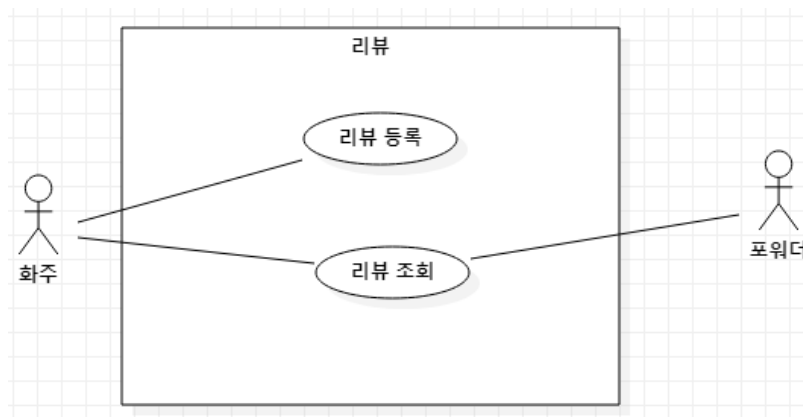
### ● 건적서 기능

- 공급자는 클라이언트 앱을 통해 소비자 건적요청서에 대한 건적서를 등록할 수 있음
- 소비자는 클라이언트 앱을 통해 공급자가 등록한 건적서 정보를 조회할 수 있음
- 소비자는 클라이언트 앱을 통해 선호하는 건적서 정보를 선택할 수 있음



### ● 리뷰 기능

- 소비자는 클라이언트 앱을 통해 공급자가 제공한 건적서에 리뷰를 작성할 수 있음
- 공급자는 클라이언트 앱을 통해 소비자가 작성한 리뷰 내용을 확인할 수 있음
- 공급자는 클라이언트 앱을 통해 소비자 리뷰 점수 통계를 확인할 수 있음



 <b>국민대학교</b> <b>소프트웨어학부</b> <b>다학제간캡스톤디자인I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	항만 소규모 물류 건적 중계 O2O 플랫폼 개발	
	<b>팀 명</b>	팀 배타버스 (BAETAVERSE) / 캡스톤 팀 43	
	Confidential Restricted	Version 1.0	2022-05-20

## 2.2.3 시스템 비기능(품질) 요구사항

- Software Product Quality Model 중 [ISO 25010 표준](#)을 기준으로 요구사항을 작성

대분류	소분류	달성 / 미달성	세부 사항
<b>Functional Suitability</b>	Functional Completeness	달성	소비자와 공급자에게 제공될 Use Case 를 모두 충족함
	Functional Correctness	달성	소비자와 공급자에게 제공되는 기능이 모든 테스트 케이스를 통과함
	Functional Appropriateness	달성	소비자와 공급자 클라이언트 분리를 통해 사용자 목표 달성을 촉진함
<b>Performance Efficiency</b>	Time Behavior	달성	HS Code 검색 시간이 약 0.01 초 미만 시간 복잡도 : O(n)
	Resource Utilization	달성	Concurrency 프로그래밍 사용 시 메모리, 스레드 등의 시스템 자원 관리를 Low Level API 에 위임
	Capacity	달성	시스템 Request, Response 처리에 문제가 없을 정도의 네트워크 버퍼 큐 설정
<b>Compatibility</b>	Co-existence	달성	HTTP 통신 프로토콜에 기반한 웹 마이크로 서비스 아키텍처 형태의 구현
	Interoperability	달성	서로 다른 시스템이 JSON 규격을 사용하여 데이터를 공유함



Usability	Appropriateness	달성	클라이언트 SW 가 사용자 목표를 적절하게 수행할 수 있도록 분리 구현
	Recognizability		
	Learnability	달성	Apple 사용자 인터페이스 표준 가이드라인을 따르기 때문에 관습적인 사용법으로 접근 가능
	Operability	달성	Use Case 에 정의된 기능에 대한 컴포넌트가 명확히 구분함
	User Error Protection	달성	네트워크 등의 오류가 발생했을 때 사용자에게 문제 사항을 인지시키고 대응할 수 있도록 예외 처리 구현
	User Interface Aesthetics	달성	HS Code 자동완성 기능을 통해 사용자 편의 기능 제공
	Accessibility	달성	장애가 있는 사람들이 TTS 기술 등을 활용하여 클라이언트 SW 를 사용할 수 있도록 대응
Reliability	Maturity	미달성	사용자가 입력한 자료의 유효성 검증 절차의 부족, 기능 개발에 집중하여 처리 우선순위가 밀림
	Availability	달성	웹서버는 24 시간 가용할 수 있도록 동작하며 HS Code 검색엔진은 사용자의 요청이 발생할 때마다 활성화됨
	Fault Tolerance	달성	사용자 Use Case 에 대한 모든 테스트 케이스를 통과함
	Recoverability	달성	데이터베이스 처리 중 오류가 발생했을 때, 갱신 내용이 반영되지 않음

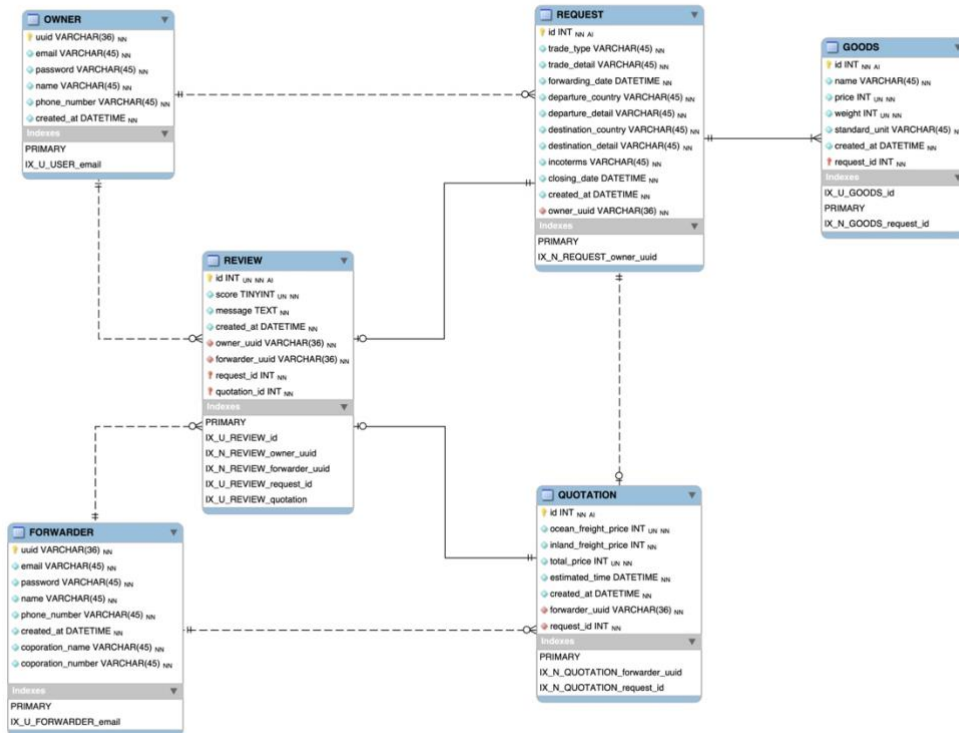


Security	Confidentiality	달성	소비자와 공급자의 권한을 나눠 관리하여 각 기능에 알맞은 사용자가 접근할 수 있도록 관리
	Integrity	달성	데이터베이스에 사용자 정보가 암호화 되어 저장되었음
	Non-repudiation	달성	모든 API 수행 전 JWT 토큰 기술을 통한 사용자 식별 진행
	Authenticity	달성	모든 API 수행 전 JWT 토큰 기술을 통한 사용자 식별 진행
	Accountability	미달성	HTTPS 등의 보안 통신 기술이 적용되지 않음, 기능 개발에 집중하여 처리 우선순위가 밀림
Maintainability	Modularity	달성	마이크로 웹 아키텍처 기반의 설계를 통한 컴포넌트 간의 느슨한 커플링
	Reusability	달성	객체지향 설계를 통한 컴포넌트 재사용성 고려한 설계
	Analyzability	달성	시스템 로그 기능 모듈 개발을 통한 실시간 모니터링 환경 구축
	Modifiability	달성	추상 클래스 활용, 서비스 확장에 따른 비즈니스 로직이 수정되었을 때 수정사항이 많아지지 않도록 설계
	Testability	달성	MVC, MVVM 디자인 패턴 도입을 통해 데이터 로직과 View 로직 분리 구현, 개별 모듈의 테스트가 가능한 구조
Portability	Adaptability	달성	클라이언트 SW 가 모바일, 태블릿, PC 에 알맞은 SW 로 손쉽게 변환할 수 있도록 설계
	Installability	달성	손쉬운 설치 가이드 제공
	Replaceability	달성	제품이 동일한 환경에서 동일한 목적으로 다른 지정된 소프트웨어 제품을 대체

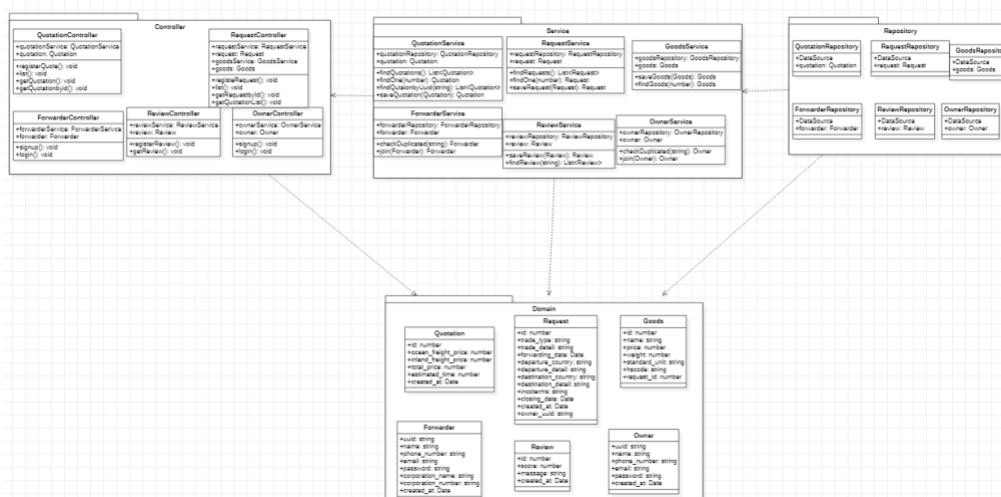


## 시스템 구조 및 설계도

- 데이터베이스 EER 다이어그램



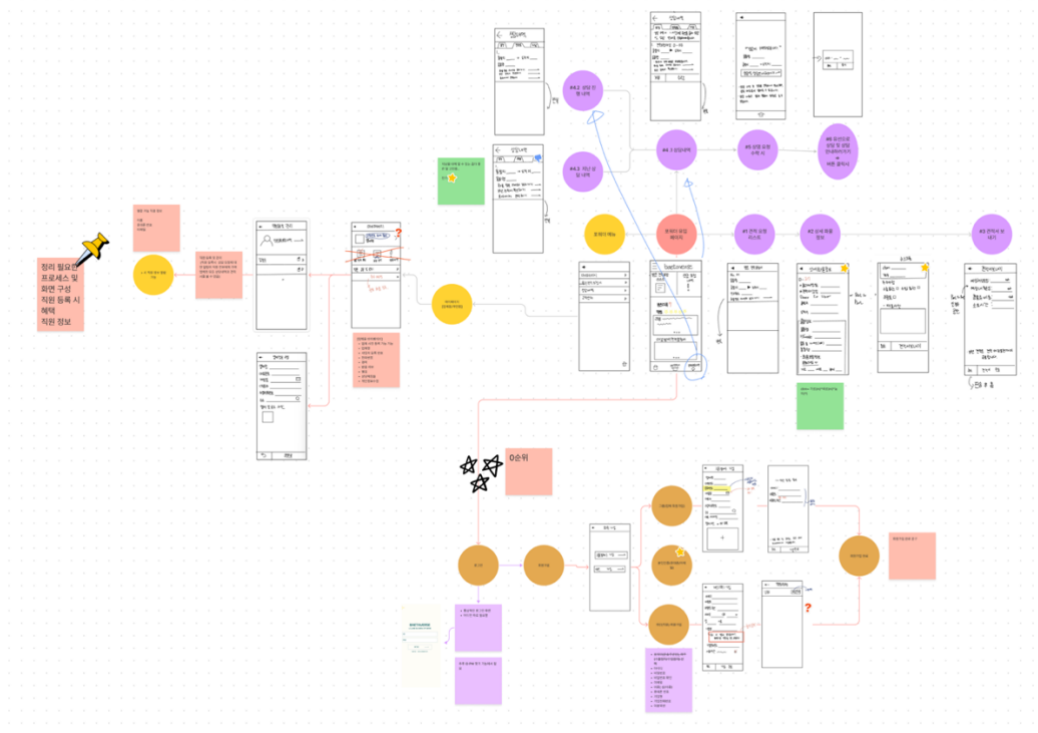
- REST-API Server UML 다이어그램



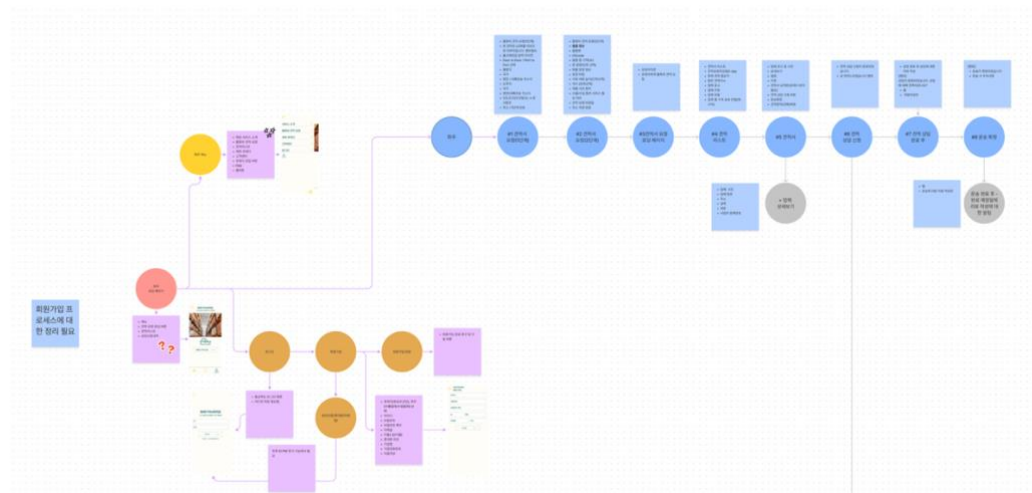




- 모바일 앱 스토리보드 설계도
  - 공급자용



- 소비자용



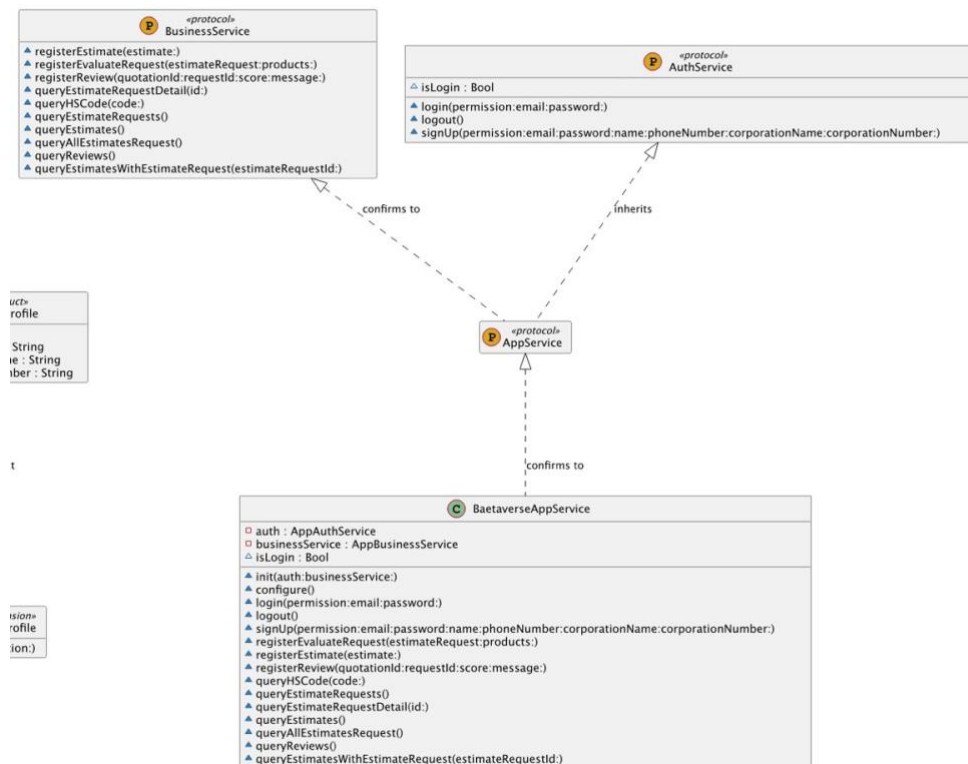
 <div> <b>국민대학교</b>  <b>소프트웨어학부</b>  <b>다학제간캡스톤디자인</b> </div>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	항만 소규모 물류 건적 중계 O2O 플랫폼 개발	
	<b>팀 명</b>	팀 배타버스 (BAETAVERSE) / 캡스톤 팀 43	
	Confidential Restricted	Version 1.0	2022-05-20

- 클라이언트 앱 설계 (MVVM 패턴, Layered Architecture)

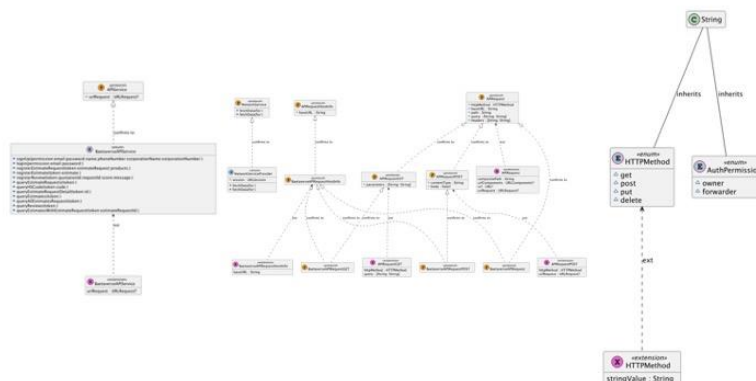
- Present Logic


UIKit, SwiftUI View, View Model Code Implement

- Business Logic



- Network Abstraction




 <b>국민대학교</b> <b>소프트웨어학부</b> <b>다학제간캡스톤디자인I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	항만 소규모 물류 건적 중계 O2O 플랫폼 개발	
	<b>팀 명</b>	팀 배타버스 (BAETAVERSE) / 캡스톤 팀 43	
	Confidential Restricted	Version 1.0	2022-05-20

## 2.2.4 활용/개발된 기술

- HS Code Recommendation System
  - 자연어 처리 기술 (NLP) 을 활용하여 Python 언어로 개발
  - 플랫폼 이용자에게 HS Code 자동 완성 편의 기능을 제공
  - 유사한 한글 단어 입력에 적절한 HS Code를 반환하도록 NLP 기술로 구현
  - Word2Vec 단어 유사도 모델을 통해 한글 단어를 HS Code로 자동 완성 기능 개발
  - AWS의 API Gateway와 Lambda 서비스를 활용한 서버리스 컴퓨팅 환경 구축
- REST-API 및 Database 서버
  - AWS EC2 : AWS가 운영하는 컴퓨팅 임대 서비스, 사용자 맞춤형 서비스 구성 가능
  - AWS RDS : AWS의 관계형 데이터베이스 서버로 안정된 운영 및 스케일링이 가능
  - Docker : 소프트웨어 구동 환경을 관리할 수 있는 VM 컨테이너
  - Express.js : Node.js 환경에서 동작하는 웹 서버 프레임워크
  - TypeScript : JavaScript의 슈퍼 셋으로 강력한 타입을 지원하는 정적 컴파일 언어
  - TypeORM : 서버 모델 객체와 데이터베이스 간의 데이터 타입을 규격화 하는 ORM
  - MySQL : 오라클에서 제공하는 오픈소스 관계형 데이터베이스 관리 시스템
  - MVC : 데이터 모델과 인터페이스 코드와의 커플링을 낮추기 위한 개발 패턴
  - JWT : JSON Web Token으로 서비스 사용자의 인증, 인가 과정에 사용
  - PM2: Cluster 모듈을 통한 멀티 프로세싱 기능 지원 및 컴퓨터 자원 모니터링
- 모바일
  - UIKit : Apple 플랫폼 SW의 사용자 인터페이스 정의용 명령형 객체지향 프레임워크
  - SwiftUI : Apple 플랫폼 SW의 사용자 인터페이스 정의용 선언형 함수형 프레임워크
  - mac Catalyst : 모바일 애플리케이션을 macOS 용으로도 실행할 수 있도록 변환
  - Swift : Apple 플랫폼의 소프트웨어를 선언할 수 있는 프로그래밍 언어
  - MVVM : 데이터 모델과 인터페이스 코드와의 커플링을 낮추기 위한 개발 패턴
  - Network Abstraction : http 통신을 위한 response, request 관리 코드 개발

## 2.2.5 현실적 제한 요소 및 그 해결 방안

- O2O 플랫폼의 활성화를 위해 초기 사용자를 만들어야 함
- 포워딩 협력 업체 구성을 통해 서비스 공급자를 구성하는 방식으로 해결
- 물류 운송 도메인 분야 고객들을 대상으로 적극적인 홍보 활동을 진행하여 해결

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>다학제간캡스톤디자인I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	항만 소규모 물류 건적 중계 O2O 플랫폼 개발	
	<b>팀 명</b>	팀 배타버스 (BAETAVERSE) / 캡스톤 팀 43	
	Confidential Restricted	Version 1.0	2022-05-20

## 2.2.6 결과물 목록

- HS Code Recommendation System
  - [GitHub PR #59](#)
  - [GitHub PR #46](#)
- Express REST-API Web application Server
  - [GitHub PR #84](#)
- MySQL Database Schema
  - [GitHub PR #66](#)
- Docker Container Micro Service Software Architecture
  - [GitHub PR #63](#)
  - [Github PR #66](#)
- AWS Service DevOps
  - [GitHub PR #66](#)
- Apple Platform Native Client Application
  - [GitHub PR #56](#)
  - [GitHub PR #61](#)

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>다학제간캡스톤디자인I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	항만 소규모 물류 건적 중계 O2O 플랫폼 개발	
	<b>팀 명</b>	팀 배타버스 (BAETVERSE) / 캡스톤 팀 43	
	Confidential Restricted	Version 1.0	2022-05-20

## 2.3 기대효과 및 활용방안

### 2.3.1 소규모 수출 화주 소비자 관점

- 소규모 화물의 수·출입 과정에서 소비자에게 효율적인 방식으로 건적 선택지 보장
- 소규모 화물 운송 과정 중 공급자와 발생할 수 있는 분쟁을 리뷰 시스템을 통해 개선
- 건적 정보를 플랫폼을 통해 통합 관리하여 사용자 물류비용 절감 및 업무 시간 감소

### 2.3.2 소규모 물류 운송 서비스 공급자 관점

- 컨테이너 운영 공급자는 더 많은 화물을 수주하여 물류 운영 비용을 절감
- 중소기업의 소규모 물류 행정 처리 진입 장벽 완화 및 비용 경감
- 소비자에게 공급자의 신뢰도를 증명하여 효율적인 고객 영입 가능
- 리뷰 시스템을 통한 공급자의 평점 유지 노력으로 소비자 운송 과정의 선순환 신뢰 구조

### 2.3.3 대한민국 국가적 성장 관점

- 국제 무역 시장에서 대한민국의 소규모 물류 운송 경쟁력을 확보
- 무역 과정의 비효율성을 개선하여 처리 시간을 줄이고 운영 비용을 절감
- 4차 산업혁명 O2O 물류 플랫폼 개발에 따른 국가적 일자리 창출

## 3 자기평가

- 프로젝트의 최종 결과물이 **충분한 사용 가능성을 갖는다고 판단**
- 개발된 모든 소프트웨어는 실행 및 검증이 가능한 상태
- 관리자가 플랫폼을 운영할 수 있도록 손쉬운 가이드를 제공하고 있음
- 소프트웨어를 사용할 도메인 사용자의 최종 SW 사용성 검증을 마침
- 소프트웨어 아키텍처 측면에서 확장성 및 유지보수성이 고려된 설계가 진행됨
- 모든 Use Case 기능 동작과 SW 품질 요구사항이 충족됨

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>다학제간캡스톤디자인</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	항만 소규모 물류 건적 중계 O2O 플랫폼 개발	
	<b>팀 명</b>	팀 배타버스 (BAETAVERSE) / 캡스톤 팀 43	
	Confidential Restricted	Version 1.0	2022-05-20

## 4 참고 문헌

번호	종류	제목	출처	발행 년도	저자	기타
1	웹문서	iOS Application Dev Guide	Apple	2022	Apple	<a href="#">링크</a>
2	웹문서	Swift Programming Language Guide	Apple	2022	Apple	<a href="#">링크</a>
3	웹문서	Express.js API Reference	Express	2022	StrongLoop	<a href="#">링크</a>
4	웹문서	TypeScript Docs	Microsoft	2022	MicroSoft	<a href="#">링크</a>
5	웹문서	Amazon Web Service	Amazon	2022	Amazon	<a href="#">링크</a>

## 5 부록

### 5.1 운영자 및 사용자 매뉴얼

- [BAETAVERSE] 프로젝트 총괄 운영 가이드 – [GitHub README.MD](#)
- [NLP] HS Code System 구동 가이드 – [GitHub Wiki](#)
- [Server] REST-API System 구동 가이드 – [GitHub Wiki](#)
- [Client] Mobile Application 구동 가이드 – [GitHub Wiki](#)

### 5.2 프로젝트 기술 문서

- [NLP] HS Code 기술 문서 – [GitHub Wiki](#)
- [Server] REST-API 기술 문서 – [GitHub Wiki](#), [Swagger Docs](#)
- [Server] 데이터베이스 기술 문서 – [Github Wiki](#)

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>다학제간캡스톤디자인I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	항만 소규모 물류 견적 중계 O2O 플랫폼 개발	
	<b>팀 명</b>	팀 배타버스 (BAETAVERSE) / 캡스톤 팀 43	
	Confidential Restricted	Version 1.0	2022-05-20

## 5.3 테스트 케이스

- CI/CD를 통해 PR 내용이 Merge 되면 자동으로 테스트 수행
- [GitHub Action](#)

대분류	소분류	기능	테스트 방법	기대 결과	테스트 결과
인증	사용자 인증	로그인	정상적인 사용자 ID 와 비밀번호를 입력 후 로그인 요청 전송	JWT 토큰 반환	성공
인증	사용자 인증	로그인	정상적인 사용자 ID 에 잘못된 비밀번호를 입력 후 로그인 요청 전송	http response error 반환	성공
인증	사용자 인증	로그인	존재하지 않는 사용자 계정으로 로그인 요청 전송	http response error 반환	성공
인증	사용자 추가	회원가입	새로운 사용자 정보를 입력한 후 회원가입 요청 전송	http response 200 반환	성공
인증	사용자 추가	회원가입	이미 등록한 사용자 정보로 중복 가입 요청을 전송	http response error 반환	성공
검색	HS Code	자동완성	사용자가 화물 정보를 입력 후 검색 요청을 전송	HS Code 자동완성 반환	성공
조회	견적 요청서	명단 조회	로그인 후 사용자의 견적 요청서 명단 조회 요청을 전송	사용자의 모든 견적 요청서 명단 반환	성공
등록	견적 요청서	요청서 등록	로그인 후 사용자의 견적 요청서 정보 입력 후 등록 요청을 전송	http response 200 반환	성공

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>다학제간캡스톤디자인I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	항만 소규모 물류 견적 중계 O2O 플랫폼 개발	
	<b>팀 명</b>	팀 배타버스 (BAETAVERSE) / 캡스톤 팀 43	
	Confidential Restricted	Version 1.0	2022-05-20

조회	견적서	견적서 조회	특정 id 를 가진 견적서를 조회하는 요청을 전송	견적서 세부 내용 JSON 객체 반환	성공
등록	견적서	견적서 등록	로그인한 사용자가 특정 견적 요청서에 대한 견적서 정보를 작성하고 등록 요청을 전송	http response 200 반환	성공
조회	견적서	견적서 조회	현재 로그인한 사용자가 자신이 작성한 모든 견적서 명단을 조회해달라는 요청 전송	견적서 명단 JSON 객체 반환	성공
조회	견적 요청서	견적 요청서 조회	데이터베이스에 등록된 모든 견적 요청서 정보를 요청 전송	견적 요청서 명단 JSON 객체 반환	성공
조회	리뷰	리뷰 조회	현재 로그인한 사용자를 대상으로 작성된 리뷰 명단 조회 요청 전송	리뷰 명단 JSON 객체 반환	성공
등록	리뷰	리뷰 등록	특정 견적서에 대한 리뷰 점수 및 메시지 작성 후 리뷰 등록 요청 전송	http response 200 반환	성공