



국민대학교  
소프트웨어융합대학  
소프트웨어학부


# 다학제 간 캡스톤 디자인 I

## 종합설계 프로젝트

프로젝트 명	모아요(Moayo)
팀 명	고리고리
문서 제목	결과보고서

Version	1.1
Date	2020-06-09

팀원	맹 산하 (조장)
	강 길웅
	김 사라
	이 정현
	정 준권

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	모아요(MoaYo)	
	<b>팀 명</b>	고리고리	
	Confidential Restricted	Version 1.1	2020-06-09


#### CONFIDENTIALITY/SECURITY WARNING

이 문서에 포함되어 있는 정보는 국민대학교 소프트웨어 융합대학 소프트웨어학부 및 소프트웨어 학부 개설 교과목 캡스톤 디자인I 수강 학생 중 프로젝트 “모아요(MoaYo)”를 수행하는 팀 “고리고리”의 팀원들의 자산입니다. 국민대학교 소프트웨어학부 및 팀 “고리고리”의 팀원들의 서면 허락없이 사용되거나, 재 가공 될 수 없습니다.

## 문서 정보 / 수정 내역


<b>Filename</b>	23조_최종결과보고서.doc
<b>원안작성자</b>	강길웅
<b>수정작업자</b>	맹산하,강길웅,김사라,이정현,정준권

수정날짜	대표수정자	Revision	추가/수정 항목	내 용
2020-06-09	강길웅	1.0	최초 작성	
2020-06-09	정준권	1.1	목록 2 수정	

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	모아요(MoaYo)	
	<b>팀 명</b>	고리고리	
	Confidential Restricted	Version 1.1	2020-06-09

## 목 차

1	개요 .....	4
1.1	프로젝트 개요 .....	4
1.2	추진 배경 및 필요성 .....	6
1.2.1	SNS 시장 현황 .....	6
1.2.2	기 개발된 시스템 현황 .....	7
1.2.3	기 개발된 시스템 문제점 .....	9
1.2.4	개발할 시스템의 필요성 .....	10
2	개발 내용 및 결과물 .....	13
2.1	목표 .....	13
2.2	연구/개발 내용 및 결과물 .....	13
2.2.1	연구/개발 내용 .....	14
2.2.2	시스템 기능 요구사항 .....	19
2.2.3	시스템 비기능(품질) 요구사항 .....	27
2.2.4	시스템 구조 및 설계도 .....	28
2.2.5	활용/개발된 기술 .....	29
2.2.6	현실적 제한 요소 및 그 해결 방안 .....	32
2.2.7	결과물 목록 .....	33
2.3	기대효과 및 활용방안 .....	33
3	자기평가 .....	17
4	참고 문헌 .....	35
5	부록 .....	36
5.1	사용자 매뉴얼 .....	36
5.2	운영자 매뉴얼 .....	36
5.3	배포 가이드 .....	36
5.4	XXX 매뉴얼 .....	17
5.5	테스트 케이스 .....	37
5.6	XXX에 대한 기술 문서 .....	18

 국민대학교 소프트웨어학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	모아요(MoaYo)	
	팀 명	고리고리	
	Confidential Restricted	Version 1.1	2020-06-09

# 1 개요


## 1.1 프로젝트 개요

관심 있는 걸 모아서 보고, 내 관심사를 모아서 표출하는 것은 우리에게 아주 익숙하고 중요한 일이다. 매일매일 SNS 에 접속해서 마음에 드는 게시물에 좋아요를 누르고, 재미있어 보이는 게시물을 보는 것은 우리에게 아주 익숙하다. 이런 SNS 중 요새 가장 대세인 걸 꼽으라면, 단연 **인스타그램**이다. 최근 SNS 의 사용 목적으로 떠오르는 정보 검색은 SNS 사용을 더 확장시키는 요인이다.

하지만 인스타그램은 단일 태그, 분류할 수 없는 게시물, 태그와 관련 없는 게시물 등 정보 검색에 있어서는 적절하지 않다. 그래서 우리는 태그를 1 개만 입력할 수 있는 인스타그램의 한정된 검색 기능을 개선하여 다중 복합 연산을 통한 메타 검색 서비스를 제공하려 한다. 계층적인 구조를 가지는 카테고리를 사용자가 입력하여 생성하고, 시스템은 사용자가 입력한 카테고리를 가지고 인스타그램에서 메타 검색을 실시한다.

일반적으로 여러 검색 엔진에 질의를 던져 얻은 결과물을 모아 한꺼번에 제공하는 검색 방식을 '메타 검색' (meta search) 이라 한다. 여기서는 인스타그램이라는 하나의 사이트에 동시에 여러 개의 질의를 던져 얻은 결과를 종합하여 사용자에게 제공하는데, 이를 '작은 의미의 메타 검색'이라 칭한다. 편의상 후술할 메타 검색이란 이 '작은 의미의 메타 검색'을 지칭하는 것이다.

사용자는 검색 결과에서 마음에 드는 게시물을 사용자가 만든 계층적 구조의 카테고리에 저장할 수 있는데, 이렇게 게시물을 저장할 수 있는 계층적인 카테고리를 '도감'이라고 한다. 한번 생성한 카테고리와 저장된 게시물은 도감의 형태로 저장되어 언제든지 확인 가능하며, 새로운 게시물을 보고싶을 때마다 기존

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	모아요(MoaYo)	
	<b>팀 명</b>	고리고리	
	Confidential Restricted	Version 1.1	2020-06-09

인스타그램의 번거로운 절차 없이 본 시스템에 저장된 도감의 카테고리를 클릭만 하면 바로 확인할 수 있다.

‘모아요’(‘MoaYo’) 는 사용자가 관심있는 카테고리에 대해 게시물을 모은다는 의미와 나의(‘My’) 라는 의미를 담고있다.

‘모아요’는 다음과 같은 서비스를 제공할 것을 목표로 한다.

### 1. 다중 태그 기반의 메타 검색


- A. 각각의 태그 및 그 유의어 추출
- B. 인스타그램의 게시물 중 1.A로부터 얻은 데이터의 교집합을 검색

### 2. 사용자가 직접 만드는 계층 구조의 도감

- A. 계층 구조로 된 입력창에 사용자가 원하는 태그를 입력하면 완성되는 도감
- B. 태그 입력시 키워드에 대한 연관 태그 제안
- C. 재검색을 위해 다시 검색창에 입력할 필요 없이 도감을 클릭하면 해당 태그들을 검색

### 3. 만든 도감을 다른 사람들과 공유

- A. 각각의 사용자들이 만든 도감을 다른 사람들과 공유

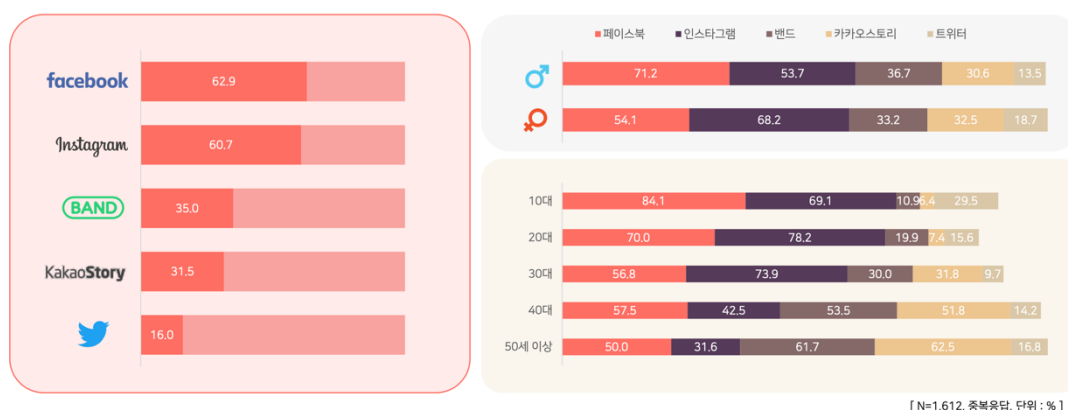
 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	모아요(MoaYo)	
	<b>팀 명</b>	고리고리	
	Confidential Restricted	Version 1.1	2020-06-09

## 1.2 추진 배경 및 필요성

### 1.2.1 SNS 시장 현황



[그림 1-1] SNS 이용 여부 현황(2018)




[그림 1-2] SNS 주 이용 서비스(2019)

나스미디어 제공

2018년 과학기술정보통신부 「정보보호실태조사」에 따르면, 국내 SNS 이용률은 74.7%에 이른다.

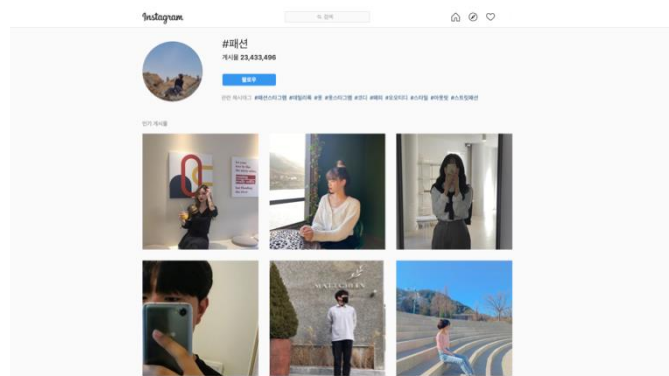
또한 [그림 1-2] SNS의 주 이용 서비스를 보면 인스타그램의 이용률이 2위이고, 이 수치는 점점 증가해 1위인 페이스북과 격차가 점점 줄어드는 추세를 보이고 있다.

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	모아요(MoaYo)	
	<b>팀 명</b>	고리고리	
	Confidential Restricted	Version 1.1	2020-06-09

## 1.2.2 기 개발된 시스템 현황

인스타그램에서 사용자의 편의를 위해 다음의 시스템을 제공하고 있다.


### 1) 태그 검색 기능

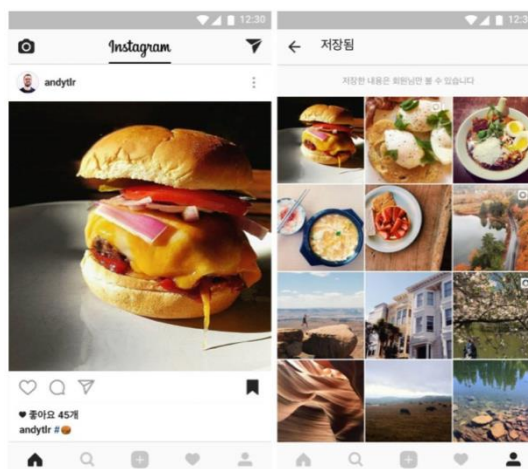


[그림 1-5] 인스타그램 태그 서칭

특정 태그를 검색하여 해당 태그가 포함된 게시글들을 검색해서 볼 수 있다.  
해당 태그가 달린 인기 게시글 들을 보여주며 최근에 올라온 게시글도 바로바로 보여준다.

### 2) 인스타그램 - 저장 기능

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	모아요(MoaYo)	
	<b>팀 명</b>	고리고리	
	Confidential Restricted	Version 1.1	2020-06-09

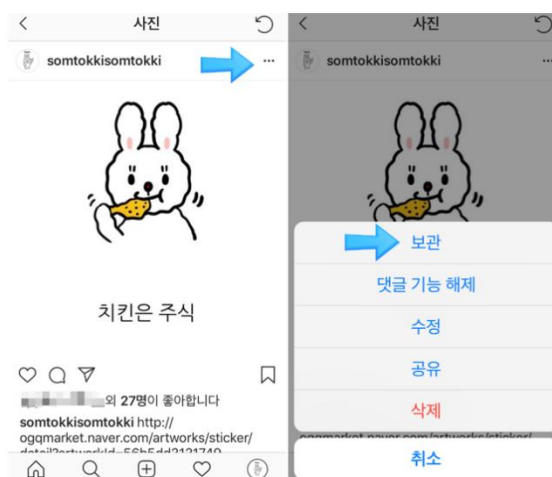


[그림 1-3] 인스타그램 저장 기능

인스타그램의 '저장' 기능은 원하는 콘텐츠들을 따로 모아서 보고 싶을 때 게시글에서 '더 보기-저장' 순으로 클릭하면 마이 페이지의 저장됨 탭에서 저장했던 콘텐츠들을 모아서 볼 수 있다.

이렇게 저장된 게시글 들은 특정 주제들끼리 모아서 볼 수도 있고, 게시글을 바로 클릭해 댓글들을 보거나 댓글을 다는 기능도 가능하다.


### 3) 인스타그램 - 보관 기능



[그림 1-4] 인스타그램 보관 기능

해당 기능은 타 사용자에게 나의 특정 게시글이나 스토리를 보이게 하고 싶지 않을 때, 나만 게시글을 보이도록 하고 싶을 때 이용하는 방법이다.



 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	모아요(MoaYo)	
	<b>팀 명</b>	고리고리	
	Confidential Restricted	Version 1.1	2020-06-09

언제든 보관을 해제하여 다른 사용자들이 볼 수 있도록 할 수 있다.

### 1.2.3 기 개발된 시스템 문제점

#### 1) 검색의 문제

현재 태그 검색은 해당 태그가 포함된 게시물들을 모두 보여준다.

이 때문에 태그 내용과는 관련이 없는 게시물도 단지 태그가 달려있다는 이유만으로 게시물 목록에 뜨기 때문에 실망스러운 검색 결과가 나오기 쉽다.

또한 태그 검색은 반드시 해당 태그가 포함된 게시글들만 검색되기 때문에 비슷한 의미를 가진 태그로 검색한 결과는 나오지 않는다.

예를 들어, '고양이'라는 태그를 검색할 때 '고양이' 태그가 없이 'Cat'이라는 태그가 달린 경우, 동일한 의미를 가진 태그가 달려있음에도 불구하고 검색되지 않는다.


또한 여러 태그를 가지고 동시에 검색을 하고 싶어도 인스타그램에서 다중 검색을 지원하지 않아 한 태그를 가지고 검색한 뒤 검색 결과를 가지고 사용자가 직접 필터링을 해야 하며, 이 작업은 매번 다시 수행해야 한다.

위와 같은 문제는 인스타그램의 API에도 동일하게 나타나는데, 인스타그램 API에서는 단일 태그 검색만을 지원하며, 태그에 대한 검색을 실시해도 한 번의 요청에 반환해주는 게시글의 갯수가 50개 미만이기 때문에 이용하기 어렵다.

#### 2) 저장의 세분화 문제

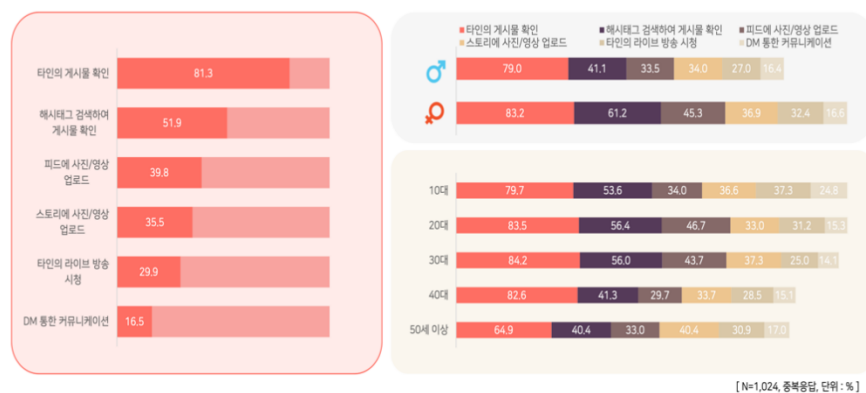
인스타그램에서 게시글에 대해 저장할 때 분류가 가능하지만 단순 1차적 분류만 가능하다는 문제를 가지고 있다. 즉, 저장되는 게시글들은 계층(hierarchy)을 가질 수 없다. 디렉토리 내부에 디렉토리를 가질 수 없으며 분류가 모두 독립적으로만 저장되는 형태를 가진다. 이러한 분류 형태는 컬렉션을 늘려 가다보면 다시 구분이 어려워지고 찾기 불편한 점이 존재한다.

#### 3) 공유의 문제

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	모아요(MoaYo)	
	<b>팀 명</b>	고리고리	
	Confidential Restricted	Version 1.1	2020-06-09

인스타그램의 게시물 공유는 하나씩 가능한 형태이다. 이미 저장되고 분류된 게시물일지라도 특정 태그나 콜렉션, 주제에 대해 다량의 게시글을 공유하려면 일일이 공유해야 하는 번거로움이 발생한다. 또한 공유 받은 사람도 공유받은 모든 게시물에 대해 다시 분류를 수행해야 하는 불편함이 존재한다.

## 1.2.4 개발할 시스템의 필요성



[그림 1-6] 인스타그램 사용 목적

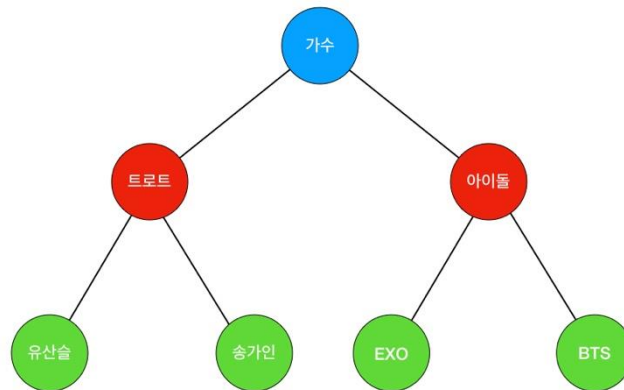
나스미디어 제공

최근 인스타그램의 사용 이유에 '태그 검색을 위함'의 수치가 높게 나타나고 있다. [그림 1-6]을 보면 인스타그램 사용 목적에서 '해시태그 검색하여 게시물 확인'이 2위로 나타난다. 인스타그램은 더 이상 지인들의 안부나 소식을 접하는 매체에서 벗어나 정보를 획득하고 필요한 정보를 저장하는 매체로 자리잡고 있는 것이다.

### 1) 검색의 측면

단일 태그로만 검색되는 인스타그램의 한계를 극복할 필요가 있다.

카테고리 생성 시 유사한 의미를 가진 단어를 함께 검색하여 검색 결과의 폭을 넓히고, 검색 결과를 카테고리의 계층 별로 분류하여 검색의 정확도를 향상시킨다.



**[그림 1-7] 카테고리 생성 예시**

카테고리 생성을 [그림 1-7]과 같은 형태로 제작했고 [가수] - [트로트] - [유산슬]을 선택했다고 한다면,

3계층인 [유산슬]에는 '가수', '트로트', '유산슬' 세가지 키워드를 조합하여 검색했을 시 나오는 모든 결과들이,

2계층인 [트로트]에는 '유산슬'에는 들어가지 않지만 '가수'와 '트로트'로 키워드를 조합하여 검색했을 시 나오는 모든 결과들이,

1계층인 [가수]에는 '트로트'와 '유산슬'에는 들어가지 않지만, '가수' 키워드로 검색했을 시 나오는 결과들이 보여지고, 이 때 각 검색어마다 동의어로 검색했을 때의 결과도 포함되므로 사용자는 이전보다 더 풍부하면서도 체계적이고 정교한 검색 결과를 얻게 된다.

따라서, 사용자는 관심있는 분야의 카테고리를 최초 생성한 후로, 새로 검색하고자 하는 키워드를 원 클릭으로 선택하기만 하면 조합된 키워드들의 새로 업데이트된 게시물을 받아볼 수 있다.


 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	모아요(MoaYo)	
	<b>팀 명</b>	고리고리	
	Confidential Restricted	Version 1.1	2020-06-09

## 2) 정보 관리의 측면

태그 검색을 통한 정보를 좀 더 체계적으로 분류하고 사용자가 관리할 수 있는 툴을 제공한다. 사용자가 원하는 태그를 조합하여 계층구조를 가지는 카테고리를 직접 만들고, 저장하고 싶은 게시글들을 사용자가 만든 카테고리에 저장하여 관리할 수 있도록 한다. 사용자는 분류된 카테고리를 통해 좀 더 관리하기 편하고 찾기 편하게 할 수 있다. 이 기능은 마치 컴퓨터의 폴더를 연상하도록 하는 기능을 제공하는 것이다.

## 3) 공유의 측면

다른 사용자가 이미 분류하고 저장해 놓은 정보에 대해서 타 사용자들도 쉽게 접근 가능하도록 한다. 인스타그램에서는 단일 게시물 공유만을 지원하기 때문에 특정 카테고리에 대해 다량의 공유가 불가능 한데, 정보의 획득이라는 목표가 늘어남에 따라 다량의 정보 공유는 필수 불가결한 사항이다.

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	모아요(MoaYo)	
	<b>팀 명</b>	고리고리	
	Confidential Restricted	Version 1.1	2020-06-09

## 2 개발 내용 및 결과물

### 2.1 목표

프로젝트 '모아요' ('MoaYo') 는 사용자에게 인스타그램 태그 검색에 대해 보다 정교한 검색 기능을 간편하게 제공하는 것을 목표로 한다.

이는 사용자가 직접 만든 카테고리, 즉 계층 (Hierarchy) 구조로 게시물을 검색하게 함으로써 이루어지는데, 계층 구조로 분류된 검색 결과는 단일 태그 검색에 비해 정확하고 체계적인 검색을 가능하게 한다.

계층구조 검색을 통해 최신 정보들을 실시간으로 가져오며, 이러한 카테고리 형태의 계층구조를 우리는 '도감'이라 정의한다.

즉, 사용자에게 의해 '도감'을 생성/관리하고, 검색하는 서비스를 하나의 플랫폼으로 제공하며, 타 사용자와 공유할 수 있는 퍼블리싱(Publishing) 기능까지 제공하는 것이 해당 프로젝트의 목표이다.

#### 1. 검색 목표

- 사용자가 작성한 카테고리에서 메타 검색을 위한 유의어 크롤
- 카테고리와 얻어낸 유의어를 가지고 인스타그램 내부 게시물 크롤
- 얻어낸 게시물을 사용자에게 제공


#### 2. 카테고리 목표

- 사용자가 입력한 태그를 통한 계층 구조의 도감 생성
- 생성된 도감의 수정 및 삭제
- 도감을 통해 검색된 게시물을 저장

#### 3. 공유 목표

- 사용자의 도감을 공유 서버 내부에 저장
- 공유 서버 내부 정보를 사용자에게 제공

### 2.2 연구/개발 내용 및 결과물

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	모아요(MoaYo)	
	<b>팀 명</b>	고리고리	
	Confidential Restricted	Version 1.1	2020-06-09

## 2.2.1 연구/개발 내용

크게 모바일 프론트엔드, 모바일 백엔드, 서버 세 가지로 구분한다.

모바일 프론트엔드는 앱의 UI/UX 및 이벤트 핸들링에 대한 처리와 사용자에게 제공할 인터페이스를 개발한다.

모바일 백엔드는 프론트엔드로 부터 받아온 데이터를 처리할 로직의 개발과 데이터의 저장 등 내부 로직에 대한 개발을 담당한다.

서버는 크게 검색 서버와 공유 서버 두가지로 분류한다.

### 1) 모바일 프론트엔드

모바일 프론트엔드는 위에 서술된 검색 목표, 카테고리 목표, 공유 목표에 해당하는 UI/UX에 관련된 작업을 실시한다. 또한, 사용자를 통해 발생하는 이벤트를 핸들링하고, 해당 이벤트를 통해 백엔드와 통신하는 것을 목표로 한다.

#### A. 검색 목표

- i. 사용자가 도감을 클릭하면 발생한다. 사용자가 원하는 계층에 해당하는 태그들을 JSON 형태로 변환해주고, 백엔드에 해당 내용을 전달하면서 게시물을 요청한다.
- ii. 백엔드 로직을 통해 얻어낸 게시물을 Recycler view를 이용한 Card View형태로 제공한다.

#### B. 카테고리 목표

- i. 사용자가 입력한 태그를 통한 계층 구조의 도감 생성
  1. 사용자가 태그들을 입력창을 통해 입력할 수 있고, 입력한 태그를 클릭하여 태그의 다음 계층으로 들어갈 수 있도록 함으로써 순차적으로 도감의 계층을 입력할 수 있도록 유도하는 것을 목표로 한다.
  2. 사용자가 생성을 완료하면, 해당 태그들을 카테고리 형태를 가진 도감으로 생성하고 해당 내용을 백엔드에 전달하는 것을 목표로 한다.
- ii. 생성된 도감의 수정 및 삭제
  1. 생성된 도감을 수정하고 삭제할 수 있는 기능을 제공하는 것을

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	모아요(MoaYo)	
	<b>팀 명</b>	고리고리	
	Confidential Restricted	Version 1.1	2020-06-09

목표로 한다. 생성된 도감의 구조를 확장 패널을 사용한 트리 뷰 형태로 제공하며, 수정 및 삭제 버튼을 이용하여 수정 및 삭제할 수 있도록 하는 것을 목표로 한다.

2. 수정은 초기 생성을 위한 화면으로 전환되고, 해당 입력창에는 도감의 내용들이 채워져 있는 것을 목표로 한다.

3. 수정 및 삭제된 도감의 정보를 백엔드에 전달하는 것을 목표로 한다.

### iii. 도감을 통해 검색된 게시물 저장

1. 도감의 각 계층에 관련된 게시물을 저장하는 것을 목표로 한다. 더블 클릭 등 여러 이벤트를 사용하여 저장할 수 있는 기능을 목표로 한다.

2. 저장된 게시물에 대한 정보를 백엔드에 전달하는 것을 목표로 한다.

## C. 공유 목표

### i. 사용자의 도감을 서버 내부에 저장

1. 사용자가 생성한 도감을 3X3의 Circle Image View를 통해 해당 도감의 대표 이미지와 설명을 보여주는 것을 목표로 한다.


2. 사용자는 도감을 클릭하여 공유를 할 경우, 해당 도감의 정보를 작성할 수 있는 화면을 보여주고, 작성된 내용을 백엔드에 전달하는 것을 목표로 한다.

### ii. 공유 서버 내부의 정보를 사용자에게 저장

1. 백엔드에서 전달받은 공유 서버의 도감들을 RecyclerView를 이용하여 보여주고, Card View를 통해 해당 이미지를 보여주는 것을 목표로 한다.

## 2) 모바일 백엔드

모바일 백엔드는 위에 서술된 검색 목표, 카테고리 목표, 공유 목표에 해당하는 비즈니스 로직과 해당 데이터를 데이터베이스 및 서버에 요청을 보내는 작업을 실시한다. 비즈니스 로직은 프론트엔드 및 서버에서 받아온 데이터에 대한 유효성을 검사하고, 알맞은 형태로 변환시키는 것을 목표로 한다.

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	모아요(MoaYo)	
	<b>팀 명</b>	고리고리	
	Confidential Restricted	Version 1.1	2020-06-09

## A. 검색 목표

- i. 사용자가 작성한 카테고리에서 메타 검색을 위한 유의어 크롤
  1. 작성한 카테고리에 해당하는 태그명을 태그와 관련된 유사어를 찾아주는 사이트를 크롤링하여 정보를 얻어내는 것을 목표로 한다.
- ii. 카테고리화 얻어낸 유의어를 가지고 인스타그램 내부 게시물 크롤
  1. 카테고리화 얻어낸 유의어를 JSON 형태로 변환하여 검색 서버에 정보를 보내는 것을 목표로 한다. 이 작업은 Retrofit을 이용한 HTTP통신하고, 쓰레드를 이용하여 비동기 처리하는 것을 목표로 한다.


## B. 카테고리 목표

- i. 사용자가 입력한 태그를 통한 계층 구조의 도감 생성
  1. 사용자가 입력한 각각의 카테고리를 하나의 도감으로 합치고, 이를 도감과 카테고리로 테이블을 나누어 저장하는 것을 목표로 한다.
- ii. 생성된 도감의 수정 및 삭제
  1. 사용자가 수정한 도감에 대해 데이터베이스의 데이터와 비교하여, 존재하는 경우 수정된 도감을 업데이트하는 것을 목표로 한다.
  2. 사용자가 삭제한 도감에 대해 데이터베이스의 데이터와 비교하여, 존재하는 경우 해당 도감을 삭제하는 것을 목표로 한다.
- iii. 도감을 통해 검색된 게시물 저장
  1. 사용자가 저장을 원하는 게시물에 대해 해당 게시물의 정보와 관련된 도감, 카테고리의 정보를 담은 테이블에 저장하는 것을 목표로 한다.

## C. 공유 목표

- i. 사용자의 도감을 공유 서버 내부에 저장
  1. 사용자가 공유를 요청한 도감에 대해 해당 도감을 JSON형태로 변환하여 공유 서버에 보내는 것을 목표로 한다. 이 작업은 Retrofit을 이용한 HTTP 통신을 수행하고, 쓰레드를 이용하여 비동기 처리하는 것을 목표로 한다.
- ii. 공유 서버 내부 정보를 사용자에게 제공



 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	모아요(MoaYo)	
	<b>팀 명</b>	고리고리	
	Confidential Restricted	Version 1.1	2020-06-09

1. 공유 서버에서 받아온 정보를 일치하는 형태로 변환시켜서 프론트엔드에 전달하는 것을 목표로 한다. 이 작업은 Retrofit을 이용한 HTTP 통신을 수행하고, 쓰레드를 이용하여 비동기 처리하는 것을 목표로 한다.

### 3) 서버

서버는 크게 두종류의 API를 제공하는 것을 목표로 한다.

인스타그램 크롤링 및 메타 검색을 수행하는 검색 서버와 사용자 간 도감을 공유하는 공유 서버를 두고 운용한다.

#### 1. 검색서버


인스타그램에서 제공하는 API는 한번에 얻을 수 있는 게시물의 갯수가 33개로 한정되어 있고, 검색어도 하나밖에 넣을 수 없기 때문에 인스타그램 크롤러가 필요하다. 단, 실시간으로 검색 결과를 보여주기 위해 따로 데이터베이스는 두지 않는다.

인스타그램 게시글들을 크롤링하고 메타 검색 로직을 적용하여 정확하고 체계적으로 분류된 검색 결과를 모바일 백엔드에 제공한다. 요청은 계층별 검색어와 해당 검색어의 동의어 목록이 담긴 http request이다. 데이터의 교환은 JSON형식을 기본으로 하며 필요한 경우 Header에 값을 포함시켜 보낸다. 모바일 백엔드에 API를 제공하는 것이 목적이므로 모바일 앱과 독립되도록 한다.

검색 서버는 모바일 백엔드에 검색의 기능을 제공하는 단일 목표를 가진다.

#### A. 검색 목표

- I. 카테고리화 얻어낸 유의어를 가지고 인스타그램 내부 게시물 크롤
  1. 모바일 백엔드가 검색 서버로 검색할 태그들과 태그들

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	모아요(MoaYo)	
	<b>팀 명</b>	고리고리	
	Confidential Restricted	Version 1.1	2020-06-09

의 연관 태그를 전달하면, 해당 데이터를 기반으로 인스타그램에서 게시물들을 크롤링하고, 태그와 연관 태그들의 교집합에 속하는 게시물들을 얻어내 모바일 백엔드에게 제공함을 목표로 한다.

## 2. 공유 서버

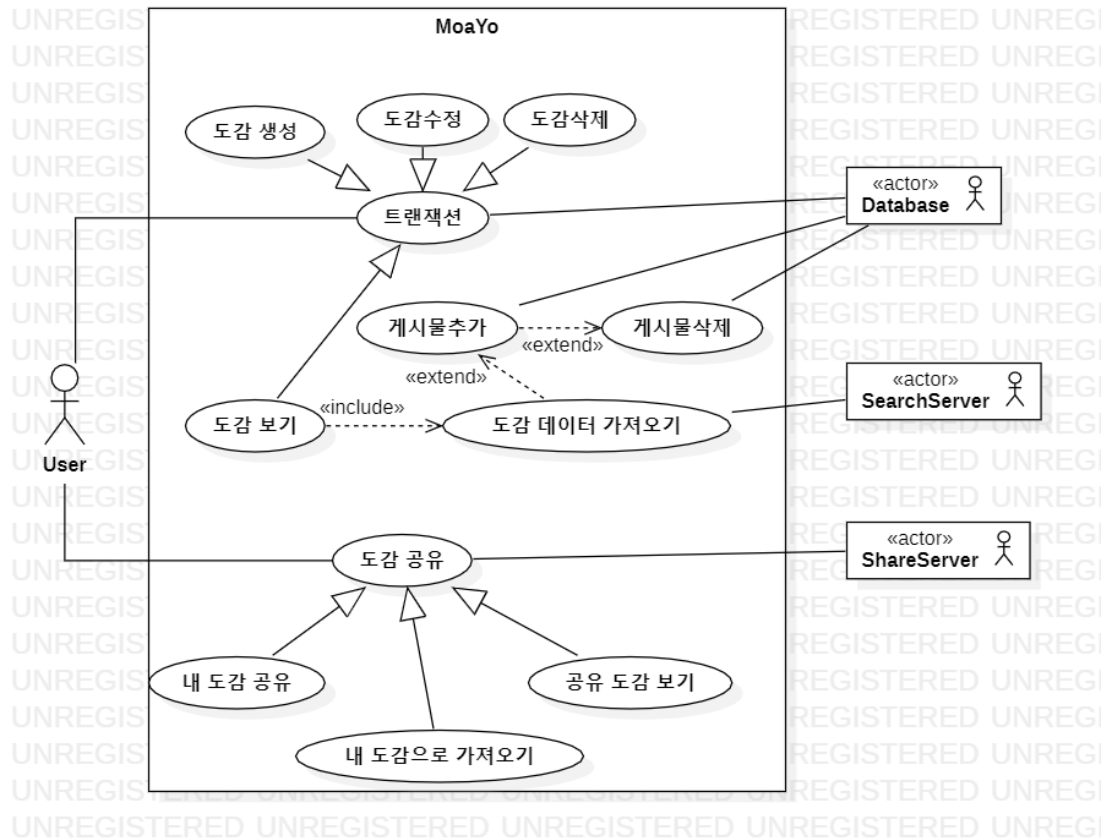
공유서버는 모바일 백엔드에서 보낸 공유 요청 및 공유된 도감 요청에 대해 서버 내부 DB를 이용하여 데이터를 저장 및 제공함을 목표로 한다. 모바일 백엔드로부터 받은 데이터를 가공하여 저장하고, 저장된 데이터를 JSON의 형태로 재 가공하여 백엔드에게 제공한다.

공유서버는 공유의 기능만을 가지고 있기 때문에 공유 목표에 대한 세부 목표만을 가진다.


### A. 공유 목표

- I. 사용자의 도감을 공유 서버 내부에 저장
  1. 백엔드가 공유서버 API를 통해 보낸 JSON 데이터를 파싱하여 엔티티 객체에 매핑해 이용 가능하도록 만드는 것을 목표로 한다.
  2. 매핑된 엔티티 객체 데이터를 공유서버와 연결된 DB에 저장함을 목표로 한다.
- II. 공유서버 내부 정보를 사용자에게 제공
  1. 백엔드가 공유서버 API로 공유된 도감 목록을 요청하면, 요청 바디에 연관 태그를 분석하고 이에 대한 데이터를 파싱하여 내부 로직을 처리 할수 있도록 함을 목표로 한다.
  2. 공유서버와 연결된 DB로부터 공유된 도감 정보들을 가져와 모바일 백엔드에게 전달함을 목표로 한다.
  3. 백엔드가 특정 도감에 대한 공유 요청을 보내면 해당 도감에 대한 정보를 DB에서 가져와 제공함을 목표로 한다.

## 2.2.2 시스템 기능 요구사항




Use Case	도감 보기 (완료)	
Triggering Event	사용자가 나의 도감 탭에서 도감을 누른다.	
Brief Description	사용자가 나의 도감 탭에서 도감을 눌러 도감에 대한 정보를 확인한다.	
Actors	Primary actor: User Secondary actor: Database, Crawler server	
Related Use Case	도감 생성, 도감 데이터 가져오기	
Preconditions	해당 도감이 존재해야 한다. 사용자가 해당 도감을 가지고 있어야 한다.	
Postconditions	사용자가 도감에 관련된 게시물을 확인한다.	
Flow of Activities	Actor	System
	1.1 사용자가 메인 화면에서	1.2 시스템은 데이터베이스에서

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	모아요(MoaYo)	
	<b>팀 명</b>	고리고리	
	Confidential Restricted	Version 1.1	2020-06-09


	도감을 클릭한다. 2.1 사용자가 탭을 눌러서 다른 카테고리 선택한다.	도감에 관련된 데이터를 찾는다. 2.2 시스템은 데이터를 카테고리 객체로 재조합하여 반환한다. 2.2 시스템은 해당 카테고리에 관련된 게시물을 띄어준다.
<b>Exception Conditions</b>	1.2a 시스템과 데이터베이스의 연결이 끊긴 경우 1.2a.1 시스템은 에러 메시지를 출력한다.	

<b>Use Case</b>	도감 데이터 가져오기 (완료)	
<b>Triggering Event</b>	사용자가 메인 화면에서 도감을 누른다.	
<b>Brief Description</b>	사용자가 메인 화면에서 도감을 누르면, 도감에 관련된 게시물을 인스타그램에서 가져온다.	
<b>Actors</b>	Primary actor: User Secondary actor: Crawler server	
<b>Related Use Case</b>	도감 보기	
<b>Preconditions</b>	해당 도감이 생성되어 있어야 한다.	
<b>Postconditions</b>	시스템은 도감에 관련된 게시물을 가져온다 사용자는 도감에 관련된 게시물을 볼 수 있다.	
<b>Flow of Activities</b>	<b>Actor</b>	<b>System</b>
	1.1 사용자는 메인 화면에서 도감을 누른다.	1.2 시스템은 선택된 도감을 데이터베이스에 서 찾는다. 1.3 시스템은 찾은 도감 객체를 크롤러 서버에 Json 형태로 요청한다. 1.4 시스템은 받은 데이터를 토대로 게시물 객체들을 생성한다. 1.5 시스템은 받은 게시물과 사용자가 저장한 게시물을 화면에 띄어준다.
<b>Exception</b>	1a.3 도감에 해당하는 게시물이 없을 때	

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	모아요(MoaYo)	
	<b>팀 명</b>	고리고리	
	Confidential Restricted	Version 1.1	2020-06-09

<b>Conditions</b>	1a.3.1 시스템은 해당 도감의 검색결과에 대한 에러 메시지를 출력한다.
-------------------	---


<b>Use Case</b>	도감 생성 (완료)	
<b>Triggering Event</b>	사용자가 도감 생성 버튼을 누른다.	
<b>Brief Description</b>	사용자가 카테고리 생성 버튼을 눌러서 새로운 계층을 가진 카테고리를 생성한다.	
<b>Actors</b>	Primary actor : User Secondary actor : Database	
<b>Related Use Case</b>	None	
<b>Preconditions</b>	None	
<b>Postconditions</b>	카테고리가 생성된다. 사용자가 만든 카테고리가 데이터베이스에 저장된다. 카테고리가 생성되지 않는다. 에러 메시지를 출력하고, 생성을 취소한다.	
<b>Flow of Activities</b>	<b>Actor</b>	<b>System</b>
	1.1 사용자가 카테고리 생성버튼을 누른다. 2.1 사용자가 카테고리 이름을 입력한다. 3.1 사용자가 카테고리 생성버튼을 누른다. 4.1 사용자가 2차 카테고리들을 입력한다. 5.1 사용자가 2차 카테고리를 클릭한다. 6.1 사용자가 3차 카테고리를 입력한다. 7.1 사용자가 Create 버튼을 누른다.	1.2 시스템은 카테고리 생성 화면을 띄어준다. 2.2 시스템은 입력된 이름을 가진 카테고리 객체를 생성한다. 3.2 시스템은 2차 카테고리 입력 화면을 띄어준다. 4.2 시스템은 생성된 카테고리 객체에 입력된 값들을 추가한다. 5.2 시스템은 3차 카테고리 입력 화면을 띄어준다. 6.2 시스템은 생성된 카테고리 객체에 입력된 값들을 추가한다. 7.2 시스템은 생성된 카테고리를 데이터베이스에 저장한다.
<b>Exception</b>	4a.1 사용자가 카테고리를 삭제하는 경우	

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	모아요(MoaYo)	
	<b>팀 명</b>	고리고리	
	Confidential Restricted	Version 1.1	2020-06-09

<b>Conditions</b>	4a.1.1 시스템은 해당 카테고리를 삭제한다. 7a.2 시스템과 데이터베이스의 연결이 끊길 경우 7a.2.1 시스템은 에러 메시지를 출력하고, 해당 카테고리를 임시 저장한다.
-------------------	--


<b>Use Case</b>	도감 수정 (완료)	
<b>Triggering Event</b>	사용자가 도감 수정 버튼을 누른다.	
<b>Brief Description</b>	사용자가 도감 수정 버튼을 눌러서 해당 도감의 카테고리를 수정한다	
<b>Actors</b>	Primary actor : User Secondary actor : Database	
<b>Related Use Case</b>	도감 생성, 도감 보기	
<b>Preconditions</b>	1. 생성된 도감이 존재해야 한다. 2. 사용자가 해당 도감을 가지고 있어야 한다.	
<b>Postconditions</b>	1. 도감의 내용이 수정된다. 수정된 도감이 데이터베이스에 갱신된다.	
<b>Flow of Activities</b>	<b>Actor</b>	<b>System</b>
	1.1 사용자가 나의 도감 탭을 누른다. 2.1 사용자가 도감을 선택한다. 3.1 사용자가 수정하고 싶은 카테고리를 수정한다. 4.1 사용자가 Update 버튼을 누른다.	1.2 시스템은 나의 도감 화면을 띄어준다. 2.2 시스템은 해당 객체를 데이터베이스에서 찾는다. 2.3 시스템은 해당 도감에 관련된 화면을 띄어준다. 4.2 시스템은 해당 객체의 내용을 수정한다. 4.3 시스템은 수정된 해당 객체를 데이터베이스에 갱신한다.
<b>Exception Conditions</b>	4.3a 시스템과 데이터베이스의 연결이 끊긴 경우 4.3a.1 시스템은 에러 메시지를 출력하고, 해당 객체를 임시저장한다.	

<b>Use Case</b>	도감 삭제 (완료)
<b>Triggering</b>	사용자가 도감 삭제 버튼을 누른다.

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	모아요(MoaYo)	
	<b>팀 명</b>	고리고리	
	Confidential Restricted	Version 1.1	2020-06-09

<b>Event</b>		
<b>Brief Description</b>	사용자가 도감 삭제 버튼을 눌러서 해당 도감을 삭제한다.	
<b>Actors</b>	Primary actor : User Secondary actor : Database	
<b>Related Use Case</b>	도감 생성, 도감 보기	
<b>Preconditions</b>	해당 도감이 존재해야 한다. 사용자가 해당 도감을 가지고 있어야 한다.	
<b>Postconditions</b>	해당 도감이 삭제된다. 삭제된 도감은 데이터베이스에서 삭제된다.	
<b>Flow of Activities</b>	<b>Actor</b>	<b>System</b>
	1.1 사용자가 나의 도감 탭을 누른다. 2.1 사용자가 도감을 선택한다. 3.1 사용자가 Delete 버튼을 누른다.	1.2 시스템은 나의 도감 화면을 띄어준다. 2.2 시스템은 해당 객체를 데이터베이스에서 찾는다. 2.3 시스템은 해당 도감에 관련된 화면을 띄어준다. 3.2 시스템은 해당 객체를 데이터베이스에서 삭제한다.
<b>Exception Conditions</b>	3.3a 시스템과 데이터베이스의 연결이 끊긴 경우 3.3a.1 시스템은 에러 메시지를 출력하고, 해당 객체를 임시 저장한다.	


<b>Use Case</b>	게시물 추가하기 (추가-완료)
<b>Triggering Event</b>	사용자가 나의 도감에 있는 게시물을 더블 클릭한다.
<b>Brief Description</b>	사용자가 나의 도감에서 게시물을 클릭하여 게시물을 저장한다.
<b>Actors</b>	Primary actor : User Secondary actor : Database
<b>Related Use Case</b>	도감 보기
<b>Preconditions</b>	해당 도감이 존재해야 한다. 게시물이 존재해야 한다.

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	모아요(MoaYo)	
	<b>팀 명</b>	고리고리	
	Confidential Restricted	Version 1.1	2020-06-09

<b>Postconditions</b>	나의 도감에 해당 게시물이 정상적으로 저장된다.	
<b>Flow of Activities</b>	<b>Actor</b>	<b>System</b>
	1.1 사용자는 나의 도감 탭을 누른다. 2.1 사용자는 검색된 게시물 뷰에서 원하는 게시물을 더블 클릭한다.	1.2 시스템은 해당 도감에 관련된 화면을 띄어준다. 2.2 시스템은 해당 게시물을 데이터베이스에 저장한다. 2.3 시스템은 해당 게시물을 저장된 게시물 뷰로 옮긴다.
<b>Exception Conditions</b>	2.2a 시스템과 데이터베이스의 연결이 끊긴 경우 2.2a.1 시스템은 에러메시지를 출력하고 저장을 취소한다.	


<b>Use Case</b>	게시물 삭제 (추가-완료)	
<b>Triggering Event</b>	사용자가 나의 도감에 있는 게시물을 길게 터치하여 삭제버튼을 누른다.	
<b>Brief Description</b>	사용자가 나의 도감에 저장되어 있는 게시물을 길게 터치하여 도감에서 해당 게시물을 삭제한다.	
<b>Actors</b>	Primary actor : User Secondary actor : Database	
<b>Related Use Case</b>	도감 보기	
<b>Preconditions</b>	나의 도감 및 카테고리 내부에 게시글이 존재한다.	
<b>Postconditions</b>	정상적으로 삭제하려는 게시글이 삭제된다.	
<b>Flow of Activities</b>	<b>Actor</b>	<b>System</b>
	1. 나의 도감 - 카테고리 - 게시글 순으로 접근한다. 2. 게시글 삭제를 클릭한다.	2. 해당 게시글을 나의 도감에서 삭제시킨다.
<b>Exception Conditions</b>	삭제 불가에 대한 오류를 표기한다.	



 국민대학교 소프트웨어학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	모아요(MoaYo)	
	팀 명	고리고리	
	Confidential Restricted	Version 1.1	2020-06-09

<b>Use Case</b>	내 도감 공유 (완료)	
<b>Triggering Event</b>	사용자는 도감 공유 버튼을 누른다.	
<b>Brief Description</b>	사용자는 도감 공유 버튼을 눌러 해당 도감을 공유한다.	
<b>Actors</b>	Primary actor: User Secondary actor: Share server	
<b>Related Use Case</b>	공유 도감 보기	
<b>Preconditions</b>	공유하려는 도감이 존재해야 한다.	
<b>Postconditions</b>	내 도감이 공유 서버에 정상적으로 업로드된다.	
<b>Flow of Activities</b>	<b>Actor</b>	<b>System</b>
	1.1 나의 도감에서 공유를 클릭한다.	1.2 현재 도감에 대한 데이터를 서버에 전송한다. 1.3 전송받은 데이터를 서버 내부에 저장하고 공유 시킨다.
<b>Exception Conditions</b>	서버와의 통신 문제 등을 팝업으로 보여준다.	


<b>Use Case</b>	공유 도감 수정 (삭제)	
<b>Triggering Event</b>	사용자는 공유 도감 수정 버튼을 누른다.	
<b>Brief Description</b>	공유한 도감에 대해 내용을 수정한다.	
<b>Actors</b>	Primary actor: User Secondary actor: Share server	
<b>Related Use Case</b>	공유 도감 보기	
<b>Preconditions</b>	공유한 도감이 있고, 해당 도감에 대해 비밀번호를 가지고 있음.	

 국민대학교 소프트웨어학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	모아요(MoaYo)	
	팀 명	고리고리	
	Confidential Restricted	Version 1.1	2020-06-09

<b>Postconditions</b>	수정이 정상적으로 반영됨.	
<b>Flow of Activities</b>	<b>Actor</b>	<b>System</b>
	1. 공유한 도감에서 수정 클릭 2. 정상 수정 후 저장	2. 요청받은 도감에 대해 정상 반영
<b>Exception Conditions</b>	1. 비밀번호가 다를 경우 팝업. 2. 현재 수정이 불가하다는 팝업.	

<b>Use Case</b>	내 도감으로 가져오기 (완료)	
<b>Triggering Event</b>	사용자 도감 가져오기 버튼을 누른다.	
<b>Brief Description</b>	타 사용자가 공유한 도감을 나의 도감에 저장한다.	
<b>Actors</b>	Primary actor: User Secondary actor: Share server	
<b>Related Use Case</b>	공유 도감 보기	
<b>Preconditions</b>	사용자의 디바이스가 공유서버와 정상적으로 통신해야 한다.	
<b>Postconditions</b>	사용자는 해당 도감을 자신의 도감에 추가할 수 있다.	
<b>Flow of Activities</b>	<b>Actor</b>	<b>System</b>
	1.1 사용자는 해당 도감을 클릭한다. 2.1 사용자는 도감 가져오기 버튼을 누른다.	1.2 시스템은 해당 도감에 저장된 데이터를 보여준다. 2.2 시스템은 해당 도감을 사용자의 데이터베이스에 저장한다.
<b>Exception Conditions</b>	서버와의 통신 문제 등을 팝업으로 보여준다.	

<b>Use Case</b>	공유 도감 보기 (완료)
<b>Triggering</b>	사용자는 모두의 도감을 클릭한다.

 국민대학교 소프트웨어학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	모아요(MoaYo)	
	팀 명	고리고리	
	Confidential Restricted	Version 1.1	2020-06-09

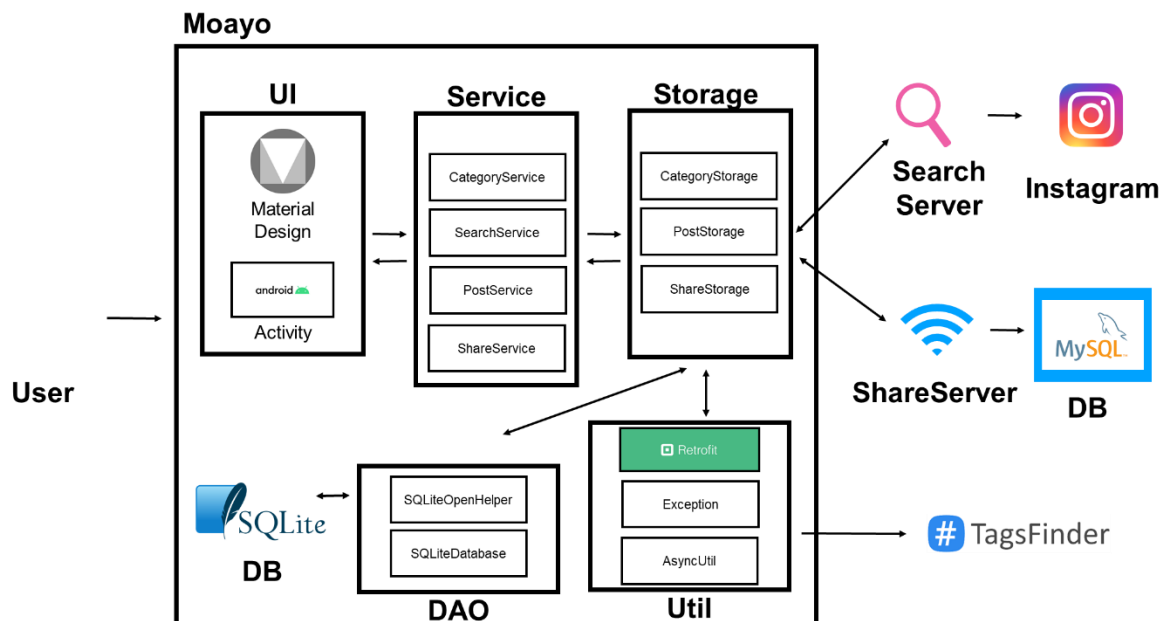
<b>Event</b>		
<b>Brief Description</b>	타 사용자가 공유한 도감을 확인한다.	
<b>Actors</b>	Primary actor: User Secondary actor: Share server	
<b>Related Use Case</b>	None	
<b>Preconditions</b>	사용자의 디바이스가 공유 서버와 정상적으로 통신해야 한다.	
<b>Postconditions</b>	사용자는 타 사용자가 공유한 도감을 확인한다.	
<b>Flow of Activities</b>	<b>Actor</b>	<b>System</b>
	1. 도감 공유를 통해 접근하여 타 유저의 도감을 클릭. 2. 전송받은 데이터를 보여줌	1. 해당 도감에 대한 정보를 사용자에게 전송
<b>Exception Conditions</b>	올바르지 않은 데이터등에 대해 다시 시도 해 달라는 팝업.	

### 2.2.3 시스템 비기능(품질) 요구사항

비기능 요구사항	우선순위	내용	비고
NFR 1 / 성능	높음(1)	시스템 성능은 동시 사용자 100 명 기준 사용자 모두 10 초 이내에 정보를 받아 볼수 있도록 한다.	달성
NFR 2 / 보안	높음(2)	타 사용자의 도감 무단 수정 및 삭제는 허가하지 않는다.	개인정보 없음.(달성)
NFR 3 / 편의	중간(3)	사용자가 접근하고 보기에 편한 UI/UX 를 제공한다.	달성
NFR 4 / 정확성	중간(4)	검색해서 나온 데이터는 정확도가 50%이상 이 되도록 한다.	미달성 - 태그에 따라 정확성의 차이가 존재한다.

NFR 5 / 상호운용	낮음(6)	서버-앱간 통신 딜레이는 수 초 미만으로 한다.	달성
NFR 6 / 법적 요구사항	낮음(7)	정보 통신법의 위배되는 도감 형성 및 공유가 이루어지지 않도록 주의한다. 사생활 침해에 위배되는 도감 형성이 되지 않도록 한다.	달성
NFR 7 / 배포	중간(5)	APK 하나로 배포 가능한 형태를 취한다.	달성

## 2.2.4 시스템 구조 및 설계도



### 1. 변경된 구조

#### A. AsyncUtil

- i. 동기적으로 처리하던 서비스 로직에서는 사용자가 호출한 메소드가 끝나기 전까지 기다려야 했던 문제가 존재한다. 이에 비즈니스 로직을 비동기로 처리하기 위하여 AsyncUtil 패키지를 생성하여 비동기 처리를 위한 객체 구조를 지정하였다. Util 패키지는 AsyncCallback<T>,

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	모아요(MoaYo)	
	<b>팀 명</b>	고리고리	
	Confidential Restricted	Version 1.1	2020-06-09

AsyncExecutor<T> 클래스로 구성되어있고, AsyncExecutor<T> 클래스는 AsyncTask를 상속받는다. 해당 클래스는 Callable<T>, AsyncCallback<T>를 변수로 가지고 있으며, doInBackground에서는 Callable<T>에 지정된 비즈니스 로직을 처리하고, onPostExecute에서는 AsyncCallback<T>의 onResult 메소드를 호출한다. 해당 유틸은 제네릭 클래스로 모든 데이터 타입에 대해서 활용할 수 있고, AsyncTask의 메소드를 활용할 수 있기 때문에 UI스레드가 처리해야 하는 작업도 동시에 설정할 수 있다.

## B. 공유서버 통신을 위한 객체 생성

- i. ModelForm 클래스를 생성하여 도감에 대한 정보를 파싱하고, 이를 공유 서버와 통신하는 객체로 활용한다.

## 2.2.5 활용/개발된 기술

### 1. 모바일 프론트엔드


#### A. Material Design

- i. 구글에서 제공되는 안드로이드 최적화 디자인 프레임워크.
- ii. 플랫폼디자인을 기반으로 입체감이 느껴지는 표면질감과 선명한 그래픽 디자인을 제공.
- iii. 어플리케이션 상 제공되는 모든 View는 Material Design을 적용하였다.

#### B. Bumptech Glide

- i. Bumptech에 포함된 안드로이드이미지 로딩 라이브러리.  
타 이미지 로딩 라이브러리에 비해 성능이 빠르고 안정적이며, 썸네일, GIF 로딩, 동영상 또한 지원한다.
- ii. 검색결과 게시물과 도감 썸네일 이미지를 해당 Glide 라이브러리를 사용하여, 실제 인스타그램 포스트의 이미지 URL을 로딩하여 표시하였다.

#### C. Circle ImageView

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	모아요(MoaYo)	
	<b>팀 명</b>	고리고리	
	Confidential Restricted	Version 1.1	2020-06-09

- i. Vince Mi의 RoundedImageView를 기반으로 개발된 원형 이미지뷰 라이브러리.
- ii. 사용자의 도감을 표시하는 화면에서 각 도감은 circle imageView를 통해 원형의 바둑판 형태로 표시된다.

## 2. 모바일 백엔드

### A. SQLite 데이터베이스

- i. 안드로이드에서 제공하는 구조화된 내부 SQL Database.

### B. Retrofit(REST API)

- i. Retrofit은 Square사에서 제공하는 오픈소스 라이브러리로 REST API를 안드로이드에서 쉽게 이용할 수 있게 해주는 도구
- ii. REST는 HTTP를 효율적으로 사용하기 위한 아키텍처
- iii. Retrofit을 활용하여 검색서버, 공유서버에 보낼 요청을 쿼리문으로 보내거나, 바디에 Object를 넣어서 실행할 수 있다.

### C. GsonConverter

- i. GsonConverter는 객체를 JSON 형태로 자동 변환해주는 라이브러리다. Gson을 활용하기 위하여 변환하고 싶은 객체의 변수에 @SerializedName을 설정한다. SerializedName은 서버에서 받고, 서버에서 반환하는 JSON의 변수명에 맞춰야 한다. 또한, 모든 변수의 Getter, Setter를 지정해줘야 한다.

### D. TagsFinder

- i. 도감 생성시 관련 있는 해시태그를 찾기 위하여 TagsFinder 사이트에서 크롤링하는 방식을 활용하였다. 각 도감의 카테고리들의 이름에 관련된

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	모아요(MoaYo)	
	<b>팀 명</b>	고리고리	
	Confidential Restricted	Version 1.1	2020-06-09

해시태그들을 가져온다.

### 3. 공유 서버

#### A. Spring Boot

- 별도의 서버를 개설할 필요 없이 톰캣 서버를 내장하여 서버 구성에서의 편의성이 제공된다.
- 배포 시 독립적인 서버보다 더 단축된 빌드 속도를 제공하며 .JAR 파일로 독립적이고 간단하게 서버를 실행할 수 있다.
- 라이브러리 및 프로퍼티 설정, 관리가 유리하다.

#### B. MySQL

- 오픈 소스 라이선스를 따르기 때문에 무료이다.
- 다양한 운영체제 및 프로그래밍 언어를 지원한다.
- 표준 SQL을 따르며 크기가 큰 데이터 집합도 빠르고 효과적으로 처리가 가능하다.


#### C. MyBatis

- SQL구문을 독립적으로 관리 할 수 있다.
- 자바 오브젝트와 SQL문 사이 자동 Mapping을 지원한다.
- 다소 복잡한 JDBC의 코드를 걷어내어 소스코드의 총량을 줄일 수 있다.

#### D. AWS – EC2

- 사용자의 필요 및 사용량에 따라 성능을 고를 수 있다.
- 언어 및 운영체제에 구매 받지 않으며 몇 분만에 전세계로 쉽게 배포 가능하다.

#### E. AWS – RDS

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	모아요(MoaYo)	
	<b>팀 명</b>	고리고리	
	Confidential Restricted	Version 1.1	2020-06-09

- i. 따로 DB서버를 구축할 필요 없이 간편하게 DB를 구축하도록 하여 시간을 절약할 수 있다.
- ii. 사용자가 원하는 DB엔진을 설정 가능하다.
- iii. Scale Up / Scale Out이 간편하다.

#### 4. 검색 서버

##### A. Golang

- i. 높은 성능
- ii. 복잡한 동시성 문제를 효율적으로 처리할 수 있음
- iii. 코드가 짧고 배포가 단순하여 유지/보수에 유리

##### B. AWS – EC2

- i. 사용자의 필요 및 사용량에 따라 성능을 고를 수 있다.
- ii. 언어 및 운영체제에 구애 받지 않으며 몇 분만에 전세계로 쉽게 배포 가능하다.

### 2.2.6 현실적 제한 요소 및 그 해결 방안

#### 1. 인스타그램 데이터 문제

- A. 기본적으로 인스타그램 게시물은 여러 가지의 해시태그를 가지고 있다. 사용자가 적절한 해시태그를 입력해도 사용자가 원하지 않은 게시물이 해당 해시태그를 가지고 있을 경우 해당 게시물이 검색될 수 있다.

#### 2. 사용자 태그 입력 문제

- A. MoaYo의 도감은 카테고리의 계층에 따라 각 카테고리의 해시태그를 동시에 포함하고 있는 게시물을 가져온다. 사용자가 카테고리의 해시태그를 관련이



 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	모아요(MoaYo)	
	<b>팀 명</b>	고리고리	
	Confidential Restricted	Version 1.1	2020-06-09

없게 적용하거나, 검색하는 카테고리의 해시태그의 양이 적을 경우 적절하지 않은 게시물이 검색될 수 있다. 이 경우, 사용자는 유의미한 해시태그를 적용해야 하고, 적절한 양의 해시태그를 넣어야 사용자가 의도한 게시물이 검색될 확률이 증가한다.

## 2.2.7 결과물 목록

대분류	소분류	내용	기술문서 유/무	부록
모바일 어플리케이션	APK	개발된 안드로이드 어플리케이션 APK 추출 후, 구글 플레이 스토어 출시.	유	Play Store link
서버	공유 서버	사용자가 제작한 도감을 서버에 업로드하여 타 사용자와 도감 정보를 공유	유	<a href="#">공유 서버 기술 문서</a>
	검색 서버	사용자에 의해 요청된 검색 키워드들을 인스타그램 게시물들을 크롤, 다중복합 연산을 수행.	유	<a href="#">검색 서버 기술 문서</a>


## 2.3 기대효과 및 활용방안

### 1. 해시 태그 검색의 정확성 향상


인스타그램의 낮은 검색 정확도 문제를 개선하고 메타 검색 기능으로 양과 질 양면으로 이전보다 더욱 많고, 정교하며 체계적인 검색 결과를 제공받을 수 있다.

### 2. 게시물을 더욱 구조적으로 관리·공유

자신만의 카테고리를 만들어 인스타그램 게시글을 저장 할 수 있다. 또한 만들어진 카테고리를 공유하거나 타 사용자의 카테고리를 공유 받아 이용할 수 있다.

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	모아요(MoaYo)	
	<b>팀 명</b>	고리고리	
	Confidential Restricted	Version 1.1	2020-06-09

나아가, 각 분야의 전문가들은 인스타그램의 게시물을 분류·분석할 수 있고, 일반인들은 이렇게 분류된 전문가의 도감을 열람할 수 있는 데이터 공유의 장이자 플랫폼으로 발전할 수 있다.

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	모아요(MoaYo)	
	<b>팀 명</b>	고리고리	
	Confidential Restricted	Version 1.1	2020-06-09

### 3 자기평가


프로젝트 구상을 위해 브레인스토밍에 많은 시간을 투자했고, 적절한 기술을 찾아내어 잘 적용하였다. 설계를 위해 소프트웨어 아키텍처 지식을 적절히 활용했고, 핵심 기능들(인스타그램 크롤링, 게시물 메타 검색, 도감 생성, 도감 저장, 도감 공유, 도감 검색 등)에 대한 구현 및 디버깅을 완료했고 최적화를 위해 많은 노력을 기울였다.

협업을 위해 GitHub, Slack, Notion 등의 협업 툴을 적절히 사용했고, 원활한 팀워크를 위해 의사소통을 많이 하려 했고, 이를 위해 매주 3회 줌을 통한 회의 시간을 가졌다.

그 결과 상용화가 가능할 만큼 프로젝트를 완성할 수 있었고, 앱 스토어에 MoaYo 1.0 을 출시했다.

### 4 참고 문헌

번호	종류	제목	출처	발행년도	저자	설명
1	통계	과학기술정보통신부, 「정보보호실태조사」	통계청	2018	통계청	SNS 이용률에 대한 통계.
2	통계	나스미디어, 인터넷 이용자 조사 NPR	나스미디어	2019	나스미디어	SNS 이용률에 대한 통계.
3	기사	국내 SNS 이용률 첫 감소...인스타그램만 상승	BLOTTER	2020	-	SNS 이용률에 대한 통계.
4	웹페이지	인스타그램	-	-	-	인스타그램 크롤링을 위함.
5	포스트	[AWS] 가장쉽게	#해리의유목	2019	Harry The	AWS 구축을 위해 AWS 구성 및

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	모아요(MoaYo)	
	<b>팀 명</b>	고리고리	
	Confidential Restricted	Version 1.1	2020-06-09

		VPC 개념잡기	코딩		Great	내부 요소 설명
6	웹페이지	Google Developer JetPack	Google	-	-	안드로이드 백엔드 개발 툴 개발 편의 제공.
7	웹페이지	Material Design	Material.io	-	-	UI/UX 디자인 툴. 보편화된 레이아웃 제공 및 물리적 효과와 같은 디자인 요소 제공.
8	오픈소스	Retrofit2	Square	-	-	모듈간 통신을 위한 오픈소스. HttpURLConnection 보다 성능 우수.
9	웹서비스	TagsFinder	TagsFinder	-	-	입력된 태그에 대해 관련 태그를 반환함.
10	웹사이트	MavenRepository	MavenRepository	-	-	Gradle 의존성 사이트.
11	웹사이트	AWS 설명서	AWS	-	-	AWS 설명서. EC2, RDS 등 설명.
12	웹사이트	Gin GitHub	GitHub	-	-	Gin Framework 설명서

## 5 부록

### 5.1 사용자 매뉴얼


사용자 매뉴얼은 [여기](#)에서 보실 수 있습니다.

### 5.2 운영자 매뉴얼

운영자 매뉴얼은 [여기](#)에서 보실 수 있습니다.

### 5.3 배포 가이드

A. 검색 서버 배포 가이드

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	모아요(MoaYo)	
	<b>팀 명</b>	고리고리	
	Confidential Restricted	Version 1.1	2020-06-09


- i. 검색 서버 배포 가이드는 [여기](#)에서 보실 수 있습니다.

#### B. 공유 서버 배포 가이드


- i. Git repository master 브랜치에 수정 사항을 푸쉬한 뒤,  
App/moayo/deploy.sh 를 실행시키면 됩니다.

## 5.4 테스트 케이스


대분류	소분류	기능	테스트 방법	기대 결과	테스트 결과
도감	도감 보기	도감의 정보를 확인한다.	1) 앱을 실행한다.	사용자가 저장한 도감과 공유서버의 도감이 보인다.	성공
	나의 도감	사용자가 저장한 도감을 확인한다.	1) 실행화면의 우측 상단의 아이콘을 클릭한다. 2) 실행화면의 나의 도감 우측 상단의 + 버튼을 클릭한다.	사용자가 저장한 도감이 보인다.	성공
	도감 생성	사용자가 도감을 생성할 수 있다.	1) 실행화면의 우측상단의 아이콘을 클릭하여 나의 도감 페이지로 이동한다. 2) 새도감 생성, 우측 상단의 +버튼을 클릭한다. 3) 도감명을 입력하고, 다음 버튼을 클릭한다. 4) 키워드를 입력하고, 추가 버튼을 클릭한다. 5) 관련 태그 목록들을 선택하고, 선택완료 버튼을 누른다. 6) 생성한 키워드의 우측 화살표를 클릭하여, 하위	새로운 도감이 생성된다.	성공

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	모아요(MoaYo)	
	<b>팀 명</b>	고리고리	
	Confidential Restricted	Version 1.1	2020-06-09

			계층을 생성한다. 7) 위와 같은 방법으로 도감을 만든다. 8) 우측 상단의 저장 버튼을 클릭한다.		
	도감 수정	사용자가 생성한 도감을 수정할 수 있다.	1) 실행화면의 저장된 도감을 클릭한다. 2) 우측 상단의 메뉴 버튼을 클릭한다. 3) 도감 수정 버튼을 클릭한다. 4) 도감 생성의 흐름과 같이 도감을 만든다. 5) 우측 상단의 저장버튼을 클릭한다.	도감이 수정된다.	성공
	도감 삭제	사용자가 생성한 도감, 공유한 도감을 삭제할 수 있다.	1) 실행화면의 저장된 도감을 클릭한다. 2) 우측 상단의 메뉴 버튼을 클릭한다. 3) 도감 삭제 버튼을 클릭한다.	도감이 삭제된다.	성공
도감 게시물	게시물 가져오기	도감에 관련된 게시물을 가져온다.	1) 실행화면의 저장된 도감, 공유 게시판의 도감을 클릭한다. 2) 2 계층의 경우 클릭 후, 돋보기 아이콘을 클릭한다. 3) 3 계층의 경우, 2 계층 클릭 후, 해당 3 계층을 클릭한다.	해당 도감에 관련된 게시물을 가져온다.	성공
	게시물 이벤트	게시물 url 로 이동한다.	1) 검색된 게시물, 저장된 게시물을 클릭한다.	해당 게시물의 url 로 이동한다.	성공
	게시물 추가	가져온 게시물을 도감에	2) 검색된 게시물에서 해당 게시물을 더블 클릭한다.	해당 게시물이 도감에 저장된다.	성공

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	모아요(MoaYo)	
	<b>팀 명</b>	고리고리	
	Confidential Restricted	Version 1.1	2020-06-09

		추가한다.			
	게시물 삭제	저장된 게시물을 도감에서 삭제한다.	1) 저장된 게시물을 더블 클릭한다.	해당 게시물이 도감에서 삭제된다.	성공
도감	도감 공유	생성한 도감을 공유한다.	1) 실행화면에서 우측 상단의 아이콘을 클릭한다. 2) 저장된 도감 중 자신이 생성한 도감을 클릭한다. 3) 우측 상단의 메뉴바를 클릭한다. 4) 도감 공유 버튼을 클릭한다. 5) 수정 허용,금지 버튼을 클릭한다. 6) 금지의 경우, 비밀번호를 입력한다. 7) 도감의 설명을 입력한다. 8) 공유하기 버튼을 클릭한다.	해당 도감이 공유 서버에 생성된다.	성공
	공유 도감 보기	공유 서버에 저장된 도감을 확인한다.	1) 앱의 실행화면에서 확인한다. 2) 좌측 상단의 메뉴 버튼을 클릭한다. 3) 도감 공유 게시판을 클릭한다.	공유된 도감을 확인한다.	성공
	공유 도감 저장	공유된 도감을 저장한다.	1) 앱의 실행화면에서 좌측 상단의 메뉴 버튼을 클릭한다. 2) 도감 공유 게시판의 도감을 클릭한다. 3) 우측 상단의 메뉴바를 클릭한다. 4) 도감 저장 버튼을 클릭한다.	공유된 도감을 내 도감으로 저장한다.	성공
	공유 취소	공유한 도감을 취소한다.	1) 앱의 실행화면에서 우측 상단의 아이콘을 클릭한다. 2) 공유한 도감을 클릭한다.	공유 게시판에 해당 도감이 삭제된다.	성공

 <b>국민대학교</b> <b>소프트웨어학부</b> <b>캡스톤 디자인 I</b>	<b>결과보고서</b>		
	<b>프로젝트 명</b>	모아요(MoaYo)	
	<b>팀 명</b>	고리고리	
	Confidential Restricted	Version 1.1	2020-06-09

			3) 우측 상단의 메뉴바를 클릭한다. 4) 공유 취소를 클릭한다.		
	도감 정렬	공유 게시판의 도감을 정렬한다.	1) 앱의 실행화면에서 좌측 상단의 메뉴바를 클릭한다. 2) 공유 게시판 버튼을 클릭한다. 3) 우측 하단의 플로팅 버튼을 클릭한다. 4) 상단의 버튼을 클릭한다.	최신순일 경우, 좋아요 순으로 정렬되고, 좋아요순일 경우, 최신순으로 정렬된다.	성공