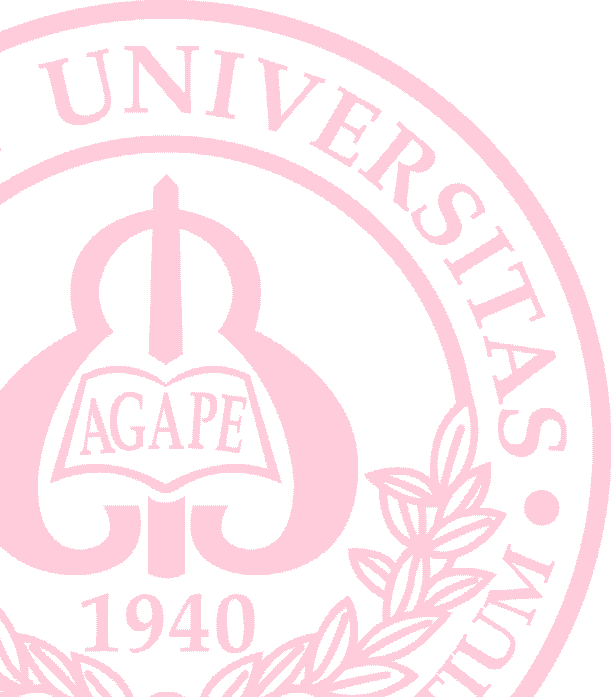
**머신러닝 기반 온라인 기사 분석 서비스 - PressCheck**



### Capstone Design 안용학, 양효식 교수님 컴퓨터공학과



**과목 담당교수 학과 팀원 제출일**

**김경남, 양재연, 최유철, 한기인 2020.11.26**

**목차**

1. 프로젝트 주제 및 주요기능 1
2. [시스템 개발 환경 5](#_TOC_250016)
3. [산출물 체계 6](#_TOC_250015)
4. [산출물 양식 7](#_TOC_250014)
   1. [분석 단계](#_TOC_250013)
      1. [요구사항 분석서 7](#_TOC_250012)
      2. 유스케이스 명세서 22
   2. [설계 단계](#_TOC_250011)
      1. [클래스 다이어그램 31](#_TOC_250010)
      2. [시퀀스 다이어그램 37](#_TOC_250009)
      3. [사용자 인터페이스 설계서 47](#_TOC_250008)
      4. 아키텍처 설계서 53
      5. 총괄시험 계획서 54
      6. [단위시험 설계 및 시나리오 55](#_TOC_250007)
      7. [데이터베이스 설계서 56](#_TOC_250006)
      8. [통합시험 시나리오 60](#_TOC_250005)
   3. [구현 단계](#_TOC_250004)
      1. [프로그램 모듈 63](#_TOC_250003)
      2. 단위 시험 결과서 77
   4. 시험 단계
      1. [프로젝트 시험 결과서 78](#_TOC_250002)
      2. [사용자 지침서 80](#_TOC_250001)
      3. 운영자 지침서 85
5. 개발 과정에서의 문제점 86
6. [향후 계획 86](#_TOC_250000)

네이버는 모바일 ‘언론사 홈’ 구독자 수가 꾸준히 증가해 2020년 7월 24일 기준 2000만을 달생했다고

# 시스템 개발 환경

|  |  |  |  |
| --- | --- | --- | --- |
| 소프트웨어 환경 | | 하드웨어 환경 | |
| Language | Python, JavaScript | CPU | Intel Core i5 7세대 이상 |
| DBMS | MongoDB | RAM | 8GB |
| WAS | MongoDB Atlas | O/S | Window, Chrome |
| IDE | Visual Studio | Network | 유선 : IEEE 802.3u  무선 : IEEE 802.11g |
| Modeling Tool | Robo 3T |
| Tech. Used | React |  |  |

# 산출물 체계

## 산출물 목록

|  |  |  |
| --- | --- | --- |
| 단계 | 코드 | 산출물 |
| 분석 | R1 | 사용자 요구사항 정의서 |
| R2 | 유스케이스 명세서 |
| 설계 | D1 | 클래스 설계서 |
| D2 | 시퀀스 다이어그램 |
| D3 | 사용자 인터페이스 설계서 |
| D4 | 아키텍처 설계서 |
| D5 | 총괄시험 계획서 |
| D6 | 데이터베이스 설계서 |
| D7 | 통합시험 시나리오 |
| D8 | 단위시험 케이스 |
| 구현 | I1 | 프로그램 코드 |
| I2 | 단위시험 결과서 |
| 시험 | T1 | 프로젝트 시험 결과서 |
| T2 | 사용자 지침서 |
| T3 | 운영자 지침서 |

# 산출물 양식

# 분석 단계

## 요구사항 분석서

### 기능적 요구사항

#### (1) 요구사항 목록 - 공통

|  |  |  |  |
| --- | --- | --- | --- |
| 공통 | | | |
| 번호 | 분류 | 요구사항 명 | 상세내용 |
| R001 | Market | Market | 메인 창 중 첫번째에 위치하여 이용자들이 게시한 자신이 나누고 싶은 물품과 설명이 리스트 형태로 표현된다. |
| R002 | Community | Community | 메인 창 중 두번째에 두 번째에 위치하여 이용자들이 게시한 공유하고 싶은 이야기와 정보를 리스트 형태로 표현된다. |
| R003 | Chat | Chat | 구매자는 원하는 마켓 게시물의 게시자와 1:1대화를 채팅 형식으로 대화 가능하게 한다. |
| R004 | Main | User Info | 메인 창의 가장 마지막에 위치되고 상대방의 프로필 이미지를 클릭 시 상대방이 거래했거나 게시한 내역과 그 이용자의 리뷰내역을 파악할 수 있다. |
| R005 | Market,  Community | Comment | 게시물들마다 가장 하단부에 위치되며 해당 게시물에 댓글을 달 수 있다. |
| R006 | Review | Review | 거래를 성사한 이용자들은 서로에게 거래의 내용에 대해 평가를 할 수 있으며 글 작성 또한 가능하다. 해당 내용들은 이용자의 정보에 기입이 된다. |
| R007 | Main | 앱 설정 | 추가 예정 |

|  |  |  |  |
| --- | --- | --- | --- |
| 유저 | | | |
| 번호 | 분류 | 요구사항 명 | 상세내용 |
| R008 | Login | 로그인 창 | 최초 실행 시 로그인 창이 나타나며 구글과 카카오 로그인 버튼이 있다. 해당 유저가 가입이 되어 있지 않다면 프로필 입력 창으로 넘어가게 된다.  회원가입이 되어 있고 아이디 정보가 저장되어 있다면 자동 로그인이 되어 해당 창은 넘어가게 된다. |
| R009 | Login | 구글 로그인 | 구글 로그인 시 해당 디바이스에 저장 되어있는 구글 아이디를 가져오고 없을 시 구글 사이트로 접속하여 구글 아이디 정보를 받게 된다. |
| R010 | Login | 카카오 로그인 | 카카오 로그인 시 해당 디바이스에 저장 되어있는 카카오 아이디를 가져오고 없을 시 구글 사이트로 접속하여 카카오 아이디를 받게 된다. |
| R011 | MemberInit | 프로필 입력 창 | 유저가 가입을 최초로 실행할 때 실행되며 유저의 사진, 닉네임, 생년월일, 학교의 정보를 입력 받을 수 있게 배치되어 있다. |
| R012 | MemberInit | 프로필 사진 선택 버튼 | 프로필 사진을 선택할 수 있는 프로필 앨범 창으로 넘어가게 해준다. |
| R013 | MemberInit | 프로필 사진 앨범 창 | 프로필 사진을 선택할 수 있는 창이다. 앨범 권한을 받으며 앨범의 사진들을 배치해서 보여준다. 상단에는 프로필 사진 초기화 버튼이 배치되어 있고 프로필로 원하는 사진을 선택 시 자동으로 프로필 입력창으로 넘어가게 된다. |
| R014 | MemberInit | 프로필 사진 앨범 초기화 버튼 | 프로필 사진을 다시 초기화 하는 버튼이다. 버튼 클릭 시 프로필 입력창으로 다시 돌아오게 되며 프로필 사진 선택 버튼은 기본 고양이 사진으로 바뀌게 된다. |
| R015 | MemberInit | 닉네임 입력 칸 | 닉네임을 입력 받을 수 있는 텍스트 창이 있다.  입력을 하지 않을 시 자동으로 로그인한 아이디를 입력 받는다. |
| R016 | MemberInit | 생년 월일 입력 칸 | 생년월일을 입력 받을 수 있는 스피너가 있다. |
| R017 | MemberInit | 학교 입력 칸 | 학교를 입력 받을 수 있는 메뉴 버튼이 있다. |
| R018 | MemberInit | 프로필 등록 확인 버튼 | 유저가 입력한 정보를 받고 메인 창으로 넘어가게 된다. |
| R019 | Main | 메인 창 | 메인 창에는 총 다섯가지의 창으로 구성되어 있다.  처음으로 나오는 창으로 마켓창으로 고정 되어있다. 하단에는 창을 움직일 수 있는 탭 버튼이 배치되어 있다. |
| R020 | Market | 마켓 창 | 메인 창의 첫번째로 위치되어 있고 상단부에는 학교분류 메뉴, 검색 버튼, 카테고리 메뉴가 위치되어 있으며 이용자가 등록한 마켓 게시물의 정보들이 리스트 형태로 간략하게 배치되어 있다. |
| R021 | Community | 커뮤니티 창 | 메인 창의 두번째로 위치되어 있고 상단부에는 학교분류 메뉴, 검색 버튼, 핫 게시물 버튼이 위치되어 있으며 이용자가 등록한 커뮤니티 게시물의 정보들이 리스트 형태로 간략하게 배치되어 있다. |
| R022 | Market  Community | 게시물 최신화 기능 | 게시물들을 최신화를 원할 시 리스트를 하단으로 드래그시 초기화되어 최신의 정보들을 보여준다. |
| R023 | Market  Community | 학교 분류 메뉴 | 이용자의 학교 정보에 따라 해당 학교의 게시물만 분류할 수 있는 메뉴버튼이다.  \*현재 미작동 |
| R024 | Market  Community | 카테고리 메뉴 | 게시물들을 해당 카테고리에 따라 분류할 수 있는 메뉴버튼이다.  마켓 : 핫 게시물, 음식교환, 물건교환, 대여하기, 퀘스트  커뮤니티 : 핫 게시물 |
| R025 | Market  Community | 검색 창 버튼 | 찾고 싶은 게시물의 제목을 검색할 수 있는 창으로 넘어가는 버튼이다. |
| R026 | Market  Community | 검색 창 | 찾고자 하는 게시물의 제목을 검색하는 창이다.  마켓 창 : 뒤로 가기 버튼, 핫 게시물 리스트, 카테고리 버튼  커뮤니티 창 : 뒤로 가기 버튼, 핫 게시물 리스트 |
| R027 | Market | 검색 창  카테고리 버튼 | 카테고리 버튼을 누르면 각 카테고리의 리스트로 구성된 창으로 넘어갈 수 있다. |
| R028 | Market | 검색 창 카테고리 리스트 창 | 검색 창의 카테고리 버튼에서 각각의 카테고리 게시물의 리스트로 구성된 창이다.  (음식교환, 물건교환, 대여하기, 퀘스트) |
| R029 | Market  Community | 핫 게시물 리스트 | 핫 게시물의 리스트로서 간단한 게시물의 정보를 담고, 클릭 시 해당 게시물 창으로 넘어간다. |
| R030 | Market  Community | 검색 창 뒤로 가기 버튼 | 검색하기 이전의 창으로 돌아가게 된다. |
| R031 | Market  Community | 검색 입력 칸 | 찾고 싶은 게시물의 제목을 기입할 수 있는 텍스트 칸이 있다. 입력 후 자판의 확인을 누르면 원하는 게시물을 리스트로 보여주는 창으로 넘어가게 된다. |
| R032 | Market  Community | 검색 결과 창 | 해당 원하는 게시물을 리스트로 보여주게 된다. 원하는 게시물을 클릭 시 해당 게시물 창으로 이동하게 된다. |
| R033 | Market | 마켓 게시물 창 | 해당 마켓 게시물의 자세한 정보를 볼 수 있다. |
| R034 | Community | 커뮤니티 게시물 창 | 해당 커뮤니티 게시물의 자세한 정보를 볼 수 있다. |
| R035 | Market  Community | 게시물 사진 (스와이프,확대버튼) | 해당 게시물의 게시자가 사진을 한 장 이상으로 올렸을 경우 왼쪽으로 스와이프 할 경우 사진이 넘어가게 된다. 사진 클릭 시 사진 전체 화면으로 넘어가게 된다. |
| R036 | Market  Community | 전체 사진 창 | 사진을 왼쪽으로 스와이핑 할 수 있다.  확대 기능 추가 예정 |
| R037 | Market  Community | 게시자 프로필 &게시자 정보 이동 기능 | 게시자의 프로필 사진, 닉네임, 게시일자가 표시되며 프로필을 클릭 시 게시자 유저의 정보가 담겨있는 창으로 넘어가게 된다. |
| R038 | Market  Community | 좋아요 버튼 | 하트 아이콘을 클릭 시 하트가 바뀌게 되며 ‘좋아요’의 개수가 증가한다. 한 번 더 클릭 시 좋아요 취소가 가능하다. |
| R039 | Market  Community | 게시물 창 메뉴 버튼 | 방문자일 경우에는 신고하기, 채팅하기가 있고, 게시자일 경우에는 수정하기, 삭제하기가 있다.  \*신고하기 기능 추가 예정 |
| R040 | Market  Community | 게시물 수정 창 | 게시물을 수정하는 창이다. 마켓과 커뮤니티의 차이는 카테고리 선택의 유무이다. |
| R041 | Market  Community | 제목 & 본문 | 게시자가 작성한 게시물의 제목과 본문의 내용이 요약되지 않고 표시된다. |
| R042 | Market  Community  [Comment] | 댓글 목록 리스트 | 해당 게시물에 이용자들은 댓글을 기입할 수 있으며 해당 댓글들은 리스트로 표현된다. |
| R043 | Market  Community  [Comment] | 댓글 메뉴 버튼 | 댓글마다의 신고하기 메뉴가 있다.  \*신고하기 기능 추가 예정 |
| R044 | Market  Community  [Comment] | 댓글 입력 칸 | 입력 칸은 하단부에 고정되어 있고 댓글의 내용을 기입이 가능하다. |
| R045 | Market  Community  [Comment] | 댓글 등록 버튼 | 입력한 내용을 댓글을 등록하고 싶다면 버튼을 클릭 시 댓글이 등록하게 된다. |
| R046 | Market | 채팅하기 버튼 | 해당 게시물의 게시자와 채팅을 하기 위한 버튼이다.  버튼 클릭 시 채팅방으로 이동하게 된다. |
| R047 | Market  Community | 글쓰기 선택 창 | 메인 화면 중 세번째에 위치하게 되며 클릭 시 마켓과 커뮤니티 중 선택이 가능하게 된다.  선택 시 해당 글쓰기 창으로 이동하게 된다. |
| R048 | Market  Community | 글쓰기 창 | 게시물의 글쓰기를 위한 창이다.  \*마켓 글쓰기만 카테고리 선택이 있다. |
| R049 | Market  Community | 글쓰기 창 뒤로 가기 버튼 | 클릭 시 글쓰기 선택 창으로 돌아가게 된다. |
| R050 | Market  Community | 사진 선택 버튼 | 클릭 시 다중선택 앨범창으로 이동하게 된다.  사진의 선택의 개수에 따라 표시된다.  최대로 다섯개까지 가능하다. |
| R051 | Market  Community | 사진 다중 선택 창 | 앨범의 권한을 받아 앨범 사진을 띄운다.  사진 찍기 또한 가능하다. |
| R052 | Market  Community | 사진 다중 선택 창 뒤로 가기 버튼 | 버튼 클릭 시 글쓰기 창으로 이동된다. |
| R053 | Market  Community | 사진 다중선택 기능 &사진확대기능 | 배열되어 있는 사진의 오른쪽 상단 부분의 체크표시를 선택 시 다중 선택이 가능하고 사진에 표시된다. 또한 사진을 클릭 시 확대화면으로 이동된다. |
| R054 | Market  Community | 사진 다중선택 창  확인 버튼 & 사진 선택 개수 확인 기능 | 사진의 선택의 개수가 기록되며 확인 버튼을 클릭 시 글쓰기에 해당 사진들을 정보를 넘겨준다. 그 후 글쓰기 화면으로 돌아가게 된다. |
| R055 | Market | 카테고리 버튼 | 마켓 게시물의 카테고리를 선택할 수 있다. |
| R056 | Market  Community | 제목 & 본문 입력 칸 | 게시물의 제목과 본문의 입력 가능하다. |
| R057 | Market  Community | 등록 버튼 | 게시자가 기입한 해당 게시물의 정보들을 마켓 혹은 커뮤니티에 기재된다. 그후 마켓 창으로 넘어가게 된다. |
| R058 | Chat | 채팅 목록 창 | 메인의 4번째 창이며 현재 접속한 유저와 채팅을 했던 모든 채팅 방의 정보들이 리스트 형태로 나타나게 된다. 해당 채팅 방 중 원하는 채팅방을 클릭 시 해당 채팅 창으로 넘어가게 된다.  \*채팅목록 방 오류 빈번히 발생 |
| R059 | Chat | 채팅 목록 창 삭제 스와이핑 & 삭제 버튼 | 채팅 방 리스트 중 삭제하고 싶은 채팅방이 있을 경우 왼쪽 스와이핑을 할 경우 삭제 버튼이 오른쪽에 생성된다. 그후 삭제를 클릭 시 해당방의 정보가 삭제된다. |
| R060 | Chat | 나가기 확인 기능 | 상대방 유저가 만약 현재 자신과 대화했던 방을 삭제했을 경우 나갔다는 표시가 된다.  \*둘 중 하나만 삭제 후 똑같은 마켓의 채팅을 한다면 정보는 사라지지 않고 남아있다. 그러나 둘 다 나갔을 경우 해당 채팅 정보는 완전히 사라진다. |
| R061 | Chat | 채팅 창 | 원하는 유저와 1대1 채팅을 할 수 있는 방이다. 채팅은 마켓 게시물에서만 가능하므로 마켓의 정보 또한 간략히 표시된다. |
| R062 | Chat | 채팅 방 뒤로 가기 버튼 | 뒤로 가기 버튼 클릭 시 채팅목록 창으로 이동된다. |
| R063 | Chat | 채팅 방 삭제하기 메뉴 | 현 채팅 방을 삭제 가능하게 해준다.  \*신고하기 추가예정 |
| R064 | Chat | 채팅 방 마켓 정보 | 상단부에 어떤 마켓에서 채팅을 했는지 마켓의 정보를 표현한다. |
| R065 | Chat | 채팅 방 마켓 정보 예약 버튼 | 만약 채팅방에 있는 현재 유저가 마켓의 게시자였다면 예약버튼이 생성되고 예약 버튼을 클릭 시 마켓 정보에 “예약중” 정보가 생성된다. 또한 거래완료 버튼이 활성화된다. 다시한번 클릭 시 “예약중” 정보가 없어지고 거래완료 버튼이 비활성화 된다. |
| R066 | Chat | 채팅 방 마켓 정보 거래완료 버튼 | 거래완료 버튼을 클릭 시 마켓의 정보에 거래완료가 생성되며 현재 유저에게 리뷰창을 띄워준다. |
| R067 | Chat | 채팅 방 현재 유저 대화 목록 & 사진 | 현재 유저의 대화 내용을 노란색 말꼬리 형식으로 압축하여 리스트 형태로 오른쪽에 표시된다. 또한 보냈던 사진도 작은 크기의 형태로 표현된다.  \*사진 확대 기능 추가예정 |
| R068 | Chat | 채팅 방 상대방 유저 대화 목록 &사진 | 상대 유저의 대화 내용을 흰색 말꼬리 형식으로 압축하여 리스트 형태로 오른쪽에 표시된다. 또한 보냈던 사진도 작은 크기의 형태로 표현된다.  \*사진 확대 기능 추가예정 |
| R069 | Chat | 앨범 사진 보내기버튼 및 앨범 창 이동 | 왼쪽 하단부에 고정되어 있으며 클릭 시 앨범의 사진을 불러온 창의 띄운다. 원하는 사진을 클릭 시 상대방에게 보여지는 대화목록에 추가되게 된다. |
| R070 | Chat | 대화 입력 칸 | 하단 중앙에 고정되어 있으며 상대방에게 원하는 대화 내용을 기입할 수 있다. |
| R071 | Chat | 대화 보내기 버튼 | 오른쪽 하단부에 고정되어 있으며 클릭 시 기입되어 있던 대화 내용을 상대방도 볼 수 있는 대화목록에 추가시켜준다. |
| R072 | User Info | 내 정보 창 | 메인에서 다섯번 째에 배치된 창이며 현재 유저의 입력되어 있었던 정보들이 기록되어 보여지는 창이다. |
| R073 | User Info | 상대 정보 창 | 상대 프로필을 눌렀을 때 보여지는 창이며 상대 유저의 정보들이 기록되어 보여지는 창이다. |
| R074 | User Info  [내 정보 창] | 회원정보 변경 버튼 | 클릭 시 현재 유저가 기입해 놓은 정보들을 변경할 수 있는 창으로 넘어가게 된다. |
| R075 | User Info  [내 정보 창] | 회원 정보 수정 창 | 현재 유저의 입력되어 있었던 정보들이 기본 세팅으로 입력되어 있으며 변경이 가능한 창이다. |
| R076 | User Info  [내 정보 창] | 회원 정보 수정 창 뒤로 가기 | 클릭 시 내 정보 창으로 돌아가게 된다. |
| R077 | User Info  [내 정보 창] | 프로필 이미지 변경 버튼, 앨범 초기화 버튼 | 회원이 선택해 놓은 사진이 프로필로 세팅 되어 있으며 클릭 시 앨범의 사진을 불러오는 창으로 넘어가게 된다. |
| R078 | User Info  [내 정보 창] | 닉네임 변경 칸 | 변경하고 싶은 닉네임을 기입할 수 있는 칸이다. |
| R079 | User Info  [내 정보 창] | 생년 월일 변경 칸 | 변경하고 싶은 생년월일을 기입할 수 있는 칸이다. |
| R080 | User Info  [내 정보 창] | 학교 변경 칸 | 변경하고 싶은 학교를 기입할 수 있는 칸이다. |
| R081 | User Info  [내 정보 창] | 수정 버튼 | 버튼 클릭 시 변경되었던 정보로 현재 유저의 정보가 수정된다. |
| R082 | User Info  [내 정보 창]  [상대 정보 창] | 진행중 버튼 | 버튼 클릭 시 현재 자신이 거래 중 이였던 마켓의 게시물을 보여주는 창으로 넘어가게 된다. |
| R083 | User Info  [내 정보 창]  [상대 정보 창] | 진행중 창 | 현재 자신이 거래 중 이였던 마켓의 게시물의 정보가 리스트로 나타나며 원하는 게시물을 클릭 시 해당 게시물로 이동된다. |
| R084 | User Info  [내 정보 창]  [상대 정보 창] | 거래 완료 버튼 | 버튼 클릭 시 현재 자신이 거래 완료였던 마켓의 게시물을 보여주는 창으로 넘어가게 된다. |
| R085 | User Info  [내 정보 창]  [상대 정보 창] | 거래 완료 창 | 현재 자신이 거래 완료였던 마켓의 게시물의 정보가 리스트로 나타나며 원하는 게시물을 클릭 시 해당 게시물로 이동된다. |
| R086 | User Info  [내 정보 창]  [상대 정보 창] | 작성한 게시물 버튼 | 버튼 클릭 시 자신이 작성되었던 게시물을 보여주는 창으로 이동하게 된다. |
| R087 | User Info  [내 정보 창]  [상대 정보 창] | 작성한 게시물 창 | 현재 자신이 작성하였던 게시물의 정보가 마켓과 커뮤니티로 분류되어 리스트로 나타나며 원하는 게시물을 클릭 시 해당 게시물로 이동된다. |
| R088 | User Info  [내 정보 창]  [상대 정보 창] | 리뷰 평가 현황 | 현재 마켓에서 거래 후 리뷰 선택사항을 얼마나 받았는가를 숫자로 표시된다.  \*추가로 총 ‘좋아요’개수까지 표현되었으면 좋겠음 |
| R089 | User Info  [내 정보 창]  [상대 정보 창] | 작성된 리뷰 버튼 | 버튼 클릭 시 글로 작성된 리뷰의 내용들을 볼 수 있는 창으로 넘어 가게 된다. |
| R090 | User Info  [내 정보 창]  [상대 정보 창] | 작성된 리뷰 창 | 리뷰 중 글을 남긴 리뷰의 내용들을 리스트로 표현되는 곳이다.  \*이거 클릭하면 게시물로 넘어감? |
| R091 | Review | 리뷰 창 | 마켓의 거래 완료 시 창이 자동적으로 나타나게 되며 거래되었던 마켓의 정보와 상대방의 평가지표가 나타난다. |
| R092 | Review | 리뷰 평가 지표 | 총 네 가지의 평가지표가 있으며 이중 하나만 선택이 가능하다. “불쾌함”을 선택 시 글로 작성해야 하는 후기를 입력해야만 한다. |
| R093 | Review | 후기 버튼 | 클릭 시 후기 작성 창이 나타난다. |
| R094 | Review | 후기 창 | 후기를 기입할 수 있는 창이다. 원하는 글을 작성 후 확인 버튼을 클릭 시 창은 사라지게 되며 리뷰 창 하단부에 자신이 작성되었던 내용이 기입되어 있다. |
| R095 | Review | 리뷰 확인 버튼 | 버튼 클릭 시 해당 내용들은 상대방 유저의 정보에 기입된다. 그후 리뷰창은 사라지게 된다. |
| R096 | Main | 로그아웃 하기 | 현재 유저의 정보를 디바이스에서 삭제하여 로그아웃을 할 수 있다. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 관리자 | | | | |
| 번호 | 분류 | 요구사항 명 | 상세내용 | 난이도 |
| R097 | Manage  [Login] | 구글 로그인 | Authentication : 구글회원으로 저장되며 자동로그인 가능  \*구글 정보를 더 많이 받을 수 있는지 조사 현재는 실행못함 | |
| R098 | Manage  [Login] | 카카오 로그인 | Authentication : 카카오에서 닉네임의 정보를 받아 이메일로 변환하여 저장한다. 자동로그인 가능  \*카카오를 통해서 로그인을 한 것이 아니기 때문에 보안에 문제가 있음 | |
| R099 | Manage  [MemberInit] | 프로필 입력 | 최초 로그인 시에만 발생하며 유저의 정보를 FireStore에 담는다. | |
| R100 | Manage  [Market]  [Community] | 게시물 리스트 | + 끌어서 최신화 업데이트를 통해 Firebase의 마켓의 정보와 커뮤니티 모든 정보를 불러온다. | |
| R101 | Manage  [Chat] | 채팅 리스트 | FireStore의 채팅의 모든 정보를 불러온 후 그 중 자신의 고유번호가 있는 정보들만 가져온다. | |
| R102 | Manage  [UserInfo] | 유저 정보 | FireStore의 유저정보 중 에서 원하는 유저의 고유번호와 일치하는 정보를 가져와 보여준다. | |
| R103 | Manage | 알림 기능 | 채팅, 게시물, 리뷰를 자신에게 남겼다면 알림을 보여준다. | |
| R104 | Manage | 신고하기 | 추가예정 | |
| R105 | Manage | 공지 사항 | 추가 예정 | |
| R106 | Manage | 자주 하는 질문 | 추가 예정 | |

* + 1. **요구사항 세부 명세**

1. R001 Market

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | Market | **요구사항구분** | Market |
| 메인 창 중 첫번째에 위치하여 이용자들이 게시한 자신이 나누고 싶은 물품과 설명이 리스트 형태로 구성된다. | | | |
| 1. 파이어스토어에서 “Market” 컬렉션의 정보를 해당 뷰에 맞추어 정보를 기입한다. 2. 정보를 게시일자를 기준으로 나열한다. 3. 리스트의 맨 위에서 드래그 시 게시물 최신화 기능을 작동시킨다. 4. 리스트의 게시물을 클릭하면 해당 게시물 화면으로 들어간다. 5. 검색창 화면으로 넘어가기 위해 검색창 버튼을 화면 상단 우측에 생성한다. 6. 학교 분류는 작동하지 않음 | | | |
| **위험요소** | * 1. 한 번에 모든 리스트를 받아오면 시간이 오래 걸리기 때문에 일정 개수의 (8개)게시물이 생성되면 연달아 생성 | | |
| **설계 시 고려사항** | 메인의 첫번째 탭에 고정된다. | | |
| **관련요구사항** | 마켓 창 | | |
| **시나리오** | 1. 첫번째 탭에서 마켓의 정보들이 나열된다. 2. 원하는 마켓의 정보를 클릭하여 자세한 해당 마켓 정보에 들어간다. 3. 아래로 드래그를 통해 최신화 업데이트가 가능하다. 4. 상단 카테고리를 통해 마켓의 정보를 원하는 정보로 분류한다. 5. 검색창 버튼을 클릭하여 검색창 화면으로 이동한다. | | |

1. R002 Community

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | Community | **요구사항구분** | Community |
| 메인 창 중 두번째에 위치하여 이용자들의 게시물의 정보가 리스트 형태로 구성된다. | | | |
| 1. 파이어스토어에서 “Community” 컬렉션의 정보를 해당 뷰에 맞추어 정보를 기입한다. 2. 정보를 게시일자를 기준으로 나열한다. 3. 리스트의 맨 위에서 드래그 시 게시물 최신화 기능을 작동시킨다. 4. 리스트의 게시물을 클릭하면 해당 게시물 화면으로 들어간다. 5. 검색창 화면으로 넘어가기 위해 검색창 버튼을 화면 상단 우측에 생성한다. 6. 학교 분류는 작동하지 않음 | | | |
| **위험요소** | 1. 한 번에 모든 리스트를 받아오면 시간이 오래 걸리기 때문에 일정 개수의 (8개)게시물이 생성되면 연달아 생성 | | |
| **설계 시 고려사항** | 메인의 첫번째 탭에 고정된다. | | |
| **관련요구사항** | 커뮤니티 창 | | |
| **시나리오** | 1. 두번째 탭에서 커뮤니티의 정보들이 나열된다. 2. 원하는 커뮤니티의 정보를 클릭하여 자세한 해당 커뮤니티 정보에 들어간다. 3. 아래로 드래그를 통해 최신화 업데이트가 가능하다. 4. 검색창 버튼을 클릭하여 검색창 화면으로 이동한다. | | |

1. R003 Chat

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | Chat | **요구사항구분** | Chat |
| 마켓 게시물 중 게시자와 1:1 채팅이 가능하다. | | | |
| 1. 유저는 게시자의 마켓 게시물에서 채팅버튼을 클릭 시 채팅방으로 넘어가게 된다. 2. 유저는 게시자와 텍스트, 이미지 등으로 1:1 대화가 가능하다. 3. 최초 채팅 시도 시 파이어스토어에 “ROOM” 컬렉션에 정보가 저장된다. 4. 읽지 않은 채팅을 알린다. | | | |
| **위험요소** | 실시간으로 대화를 주고 받아야한다. | | |
| **설계 시 고려사항** | 1. 마켓의 정보, 이전 대화 정보를 띄워 주어야 한다. 2. 구매자가 최초 채팅을 실행해야만, 채팅방이 생성된다. | | |
| **관련요구사항** |  | | |
| **시나리오** | 1. 유저는 원하는 마켓 게시물에 있는 채팅하기 버튼을 통해 채팅을 시작한다. 2. 유저는 글과 사진을 통해 서로의 정보를 채팅 형식으로 주고받을 수 있다. 3. 읽지 않은 채팅을 채팅방 목록에서 숫자로 알린다. 4. 이전의 채팅방을 다시 들어가면 채팅이 순서대로 나열된다. | | |

1. R004 User Information

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | User Information | **요구사항구분** | Main |
| 메인 창 중 다섯번째에 위치되어 자신의 정보를 확인 및 수정이 가능하며, 보고 싶은 유저의 프로필을 클릭하여 상대방 유저의 정보를 확인할 수 있다. | | | |
| 정보 : [닉네임, 프로필 사진, 학교, 생일, 거래완료/예약중인/작성한 게시물, 리뷰, 후기]   1. 자신의 정보 : 메인의 다섯번째 탭에 위치되어 확인 및 수정 2. 상대방 정보 : 게시물에서 상대방의 프로필 사진을 클릭하여 확인 3. 파이어베이스에서 “USERS” 컬렉션에서 해당 고유번호를 찾아 정보를 받아온다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** |  | | |
| **관련요구사항** |  | | |
| **시나리오** | 1. 자신의 정보를 보고 싶을 경우 메인 다섯번째 탭을 클릭하여 확인한다. 2. 상대방의 정보를 보고 싶을 경우 상대방의 프로필을 클릭하여 확인한다. | | |

1. R005 Comment

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | Comment | **요구사항구분** | Market, Community |
| 마켓, 커뮤니티에서 게시물 화면 하단에 댓글을 남길 수 있다. | | | |
| 1. 게시자 : 댓글의 메뉴에서 ‘삭제’를 선택할 수 있다. 2. 비 게시자 : 냇글의 메뉴에서 ‘신고하기’를 선택할 수 있다. 3. 댓글이 입력되면, “COMMENTS”컬렉션에 저장된다. 4. 댓글들이 순서대로 나열된다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** | 게시물 삭제 시 댓글도 함께 사라져야 한다. | | |
| **관련요구사항** |  | | |
| **시나리오** | 1. 유저가 게시물에서 댓글을 작성한다. 2. 본인 것만 삭제할 수 있다. 3. 게시물 삭제 시 함께 삭제된다. | | |

1. R006 Review

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | Review | **요구사항구분** | Review |
| 마켓의 거래성사 후 리뷰를 작성할 수 있다. | | | |
| 1. 게시자 : 채팅에서 거래완료 선택 시 리뷰창을 띄운다. 2. 비 게시자 : 게시자 가 리뷰를 작성하면 다음 앱 실행 시 바로 리뷰창을 띄운다. 3. 리뷰가 입력되면, “REVIEW” 컬렉션에 저장된다. 4. 비 게시자 가 작성하지 않은 리뷰의 개수 상관없이 앱 실행 1번당 1개의 리뷰창을 띄운다. | | | |
| **위험요소** | 리뷰를 쓰려는 타이밍에 게시글이 삭제가 될 수 있다. | | |
| **설계 시 고려사항** | 비 게시자 에게는 마켓의 정보, 게시자 의 정보가 주어져야 한다. | | |
| **관련요구사항** |  | | |
| **시나리오** | 1. 게시자 가 채팅창에서 거래완료를 선택한다. 2. 게시자 가 리뷰창을 작성한다. 3. 비 게시자 가 리뷰창을 작성한다. | | |

1. R007 앱 설정

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 앱 설정 | **요구사항구분** | Main |
| 앱의 설정 상태를 설정할 수 있다. | | | |
| 1. 알림의 수신 동의여부 (댓글, 채팅, 리뷰) 등을 체크한다. 2. 추가 예정 | | | |
| **위험요소** | 알림에 대한 수신을 상황에 맞게 삭제할 수 있어야한다. | | |
| **설계 시 고려사항** |  | | |
| **관련요구사항** |  | | |
| **시나리오** | 1. 유저가 앱 설정 창으로 들어간다. 2. 유저가 알림에 대한 수신 동의 여부를 설정한다. | | |

1. R008 로그인 창

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 로그인 창 | **요구사항구분** | Login |
| 유저가 앱을 실행하여 로그인을 할 수 있고, 구글, 카카오 중에 원하는 방식으로 로그인을 한다. | | | |
| 1. 만일 이전에 로그인이 되어 있었다면 해당 페이지는 넘어간 채 바로 메인 창으로 이동한다. | | | |
| **위험요소** | 이전의 로그인을 감지할 수 있어야한다. | | |
| **설계 시 고려사항** |  | | |
| **관련요구사항** |  | | |
| **시나리오** | 1. 유저가 앱을 실행한다. 2. 원하는 로그인 방식을 선택한다. 3. 이전에 로그인을 한 적이 있다면 넘어간다. | | |

1. R009 구글 로그인

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 구글 로그인 | **요구사항구분** | Login |
| 유저가 로그인 창에서 구글 로그인을 선택했을 때이다. | | | |
| 1. 해당 디바이스에 있는 구글 아이디를 가져온다. 2. 없을 시에는 구글 사이트로 접속하여 아이디 정보를 가져온다. | | | |
| **위험요소** | 기기에 있는 구글 로그인 정보를 알 수 있어야한다. | | |
| **설계 시 고려사항** |  | | |
| **관련요구사항** | 로그인 창 | | |
| **시나리오** | 1. 유저가 로그인 창에서 구글 로그인을 선택한다. 2. 디바이스에 있는 아이디가 있을 시 받아온 것을 선택하여 로그인한다. 3. 디바이스에 아이디가 없을 시 구글 사이트로 이동하여 로그인정보를 받아온다. | | |

1. R010 카카오 로그인

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 카카오 로그인 | **요구사항구분** | Login |
| 유저가 로그인 창에서 카카오 로그인을 선택했을 때이다. | | | |
| 1. 해당 디바이스에 있는 카카오 아이디를 가져온다. 2. 없을 시에는 카카오 사이트로 접속하여 아이디 정보를 가져온다. | | | |
| **위험요소** | 기기에 있는 카카오 로그인 정보를 알 수 있어야한다. | | |
| **설계 시 고려사항** |  | | |
| **관련요구사항** | 로그인 창 | | |
| **시나리오** | 1. 유저가 로그인 창에서 카카오 로그인을 선택한다. 2. 디바이스에 있는 아이디가 있을 시 받아온 것을 선택하여 로그인한다. 3. 디바이스에 아이디가 없을 시 구글 사이트로 이동하여 로그인정보를 받아온다. | | |

1. R011 프로필 입력 창

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 프로필 입력 창 | **요구사항구분** | MemberInit |
| 유저가 로그인을 완료하고 Main으로 넘어가고, 회원정보를 입력하지 않았으면 입력할 수 있다. | | | |
| 1. 로그인 후에 해당 아이디로 입력 받은 로그인 정보가 없다면 정보를 입력 받는다. 2. 입력 받은 정보는 파이어베이스의 “USERS” 컬렉션에 저장된다. | | | |
| **위험요소** | 로그인 후 회원정보를 입력하지 않았으면 계속 프로필 입력 창을 띄우고,  정보를 입력했다면, 이 창은 이후로는 띄우지 않는다. | | |
| **설계 시 고려사항** |  | | |
| **관련요구사항** |  | | |
| **시나리오** | 1. 유저가 로그인 후 넘어온다. 2. 입력 받은 유저의 정보가 없다면, 프로필 입력 창이 뜬다. 3. 입력 받은 유저의 정보가 있다면, 메인으로 넘어간다. | | |

1. R012 프로필 사진 선택 버튼

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 프로필 사진 선택 버튼 | **요구사항구분** | MemberInit |
| 유저가 프로필 입력 창에서 프로필 사진을 설정하기 위해 앨범으로 들어가기 위한 버튼이다. | | | |
| 1. 유저가 프로필 사진을 설정하기 위해 클릭하는 버튼이다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** |  | | |
| **관련요구사항** |  | | |
| **시나리오** | 1. 유저가 프로필 정보 입력창으로 들어온다. 2. 프로필 사진을 설정하기 원할 시에 클릭한다. | | |

1. R013 프로필 사진 앨범 창

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 프로필 사진 앨범 창 | **요구사항구분** | MemberInit |
| 유저가 프로필 입력 창에서 프로필 사진 선택 버튼을 클릭했을 때 앨범을 불러온 창이다. | | | |
| 1. 유저의 앨범을 불러온다. 2. 사진을 선택하면 해당 사진을 프로필 입력창으로 불러온다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** |  | | |
| **관련요구사항** | 프로필 입력 창, 프로필 사진 선택 버튼 | | |
| **시나리오** | 1. 유저가 프로필 정보 입력창에서 프로필 사진 앨범 버튼을 클릭한다. 2. 사진을 선택한다. | | |

1. R014 프로필 사진 앨범 초기화 버튼

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 프로필 사진 앨범 초기화 버튼 | **요구사항구분** | MemberInit |
| 유저가 프로필 사진으로 쓸 이미지를 선택하고 싶지 않을 때 초기화를 위한 버튼이다. | | | |
| 1. 유저가 사진을 선택한 것을 취소 | | | |
| **위험요소** | 초기화 버튼을 눌렀을 때 선택한 사진이 초기화 되야 하고, 프로필 입력창에 띄워진 사진도 초기화 되야 한다. | | |
| **설계 시 고려사항** |  | | |
| **관련요구사항** | 프로필 사진 앨범 창 | | |
| **시나리오** | 1. 유저가 사진을 선택한다. 2. 유저가 사진을 초기화하기 위해 프로필 사진 앨범으로 들어온다. 3. 프로필 사진 앨범 초기화 버튼을 클릭한다. 4. 선택한 이미지가 초기화된다. | | |

1. R015 닉네임 입력 칸

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 닉네임 입력 칸 | **요구사항구분** | MemberInit |
| 프로필 입력창에서 닉네임을 입력하기 위한 칸이다. | | | |
| 1. 유저의 닉네임을 입력 받는다. 2. 유저가 따로 닉네임을 선택하지 않을 시 로그인 한 ID로 한다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** |  | | |
| **관련요구사항** | 프로필 입력 창 | | |
| **시나리오** | 1. 유저가 원하는 닉네임이 있을 경우 입력한다. 2. 원하는 닉네임이 없을 경우 자동으로 설정한다. | | |

1. R016 생년 월일 입력 칸

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 생년 월일 입력 칸 | **요구사항구분** | MemberInit |
| 프로필 입력창에서 생년 월일을 입력하기 위한 칸이다. | | | |
| 1. 유저의 생년월일을 입력 받는다. 2. 유저가 따로 입력하지 않을 시 2021/01/01로 한다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** | 스피너로 설정한다. | | |
| **관련요구사항** | 프로필 입력 창 | | |
| **시나리오** | 1. 유저가 생년월일을 입력한다. 2. 설정하지 않을 경우 2021/01/01으로 설정한다. | | |

1. R017 학교 입력 칸

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 학교 입력 칸 | **요구사항구분** | MemberInit |
| 유저가 학교를 입력하기 위한 메뉴 버튼이다. | | | |
| 1. 유저의 학교를 선택 받는다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** | 메뉴 버튼으로 설정한다. | | |
| **관련요구사항** | 프로필 입력 창 | | |
| **시나리오** | 1. 유저가 학교를 설정한다. 2. 설정하지 않을 경우 ‘홍익대학교’로 설정한다. | | |

1. R018 프로필 등록 확인 버튼

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 프로필 등록 확인 버튼 | **요구사항구분** | MemberInit |
| 프로필 입력창에서 정보를 전부 입력 받고, 등록을 하기 위한 버튼이다. | | | |
| 1. 유저의 정보를 “USERS” 컬렉션에 저장한다. 2. 메인 창으로 넘어간다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** |  | | |
| **관련요구사항** | 프로필 입력 창 | | |
| **시나리오** | 1. 유저가 프로필 입력 창에서 정보를 입력한다. 2. 프로필 등록 확인 버튼을 클릭한다. 3. 메인 창으로 넘어간다. | | |

1. R019 메인 창

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 메인 창 | **요구사항구분** | Main |
| 메인 창으로 다섯가지의 창으로 구성되어 있다. | | | |
| 1. 마켓 창, 커뮤니티 창, 게시물 작성 창, 채팅 목록 창, 내 정보 창으로 구성되어 있다. 2. 첫 화면은 마켓 창이다. 3. 하단부에 위의 5개의 창을 움직일 수 있는 탭이 존재한다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** |  | | |
| **관련요구사항** |  | | |
| **시나리오** | 1. 유저가 로그인이나 회원 정보 입력 후 넘어온다. 2. 유저가 원하는 곳으로 하단 탭을 클릭하여 이동한다. | | |

1. R020 마켓 창

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 마켓 창 | **요구사항구분** | Market |
| 메인 창의 첫번째 위치 | | | |
| 1. 상단에는 학교분류, 검색 창 버튼, 카테고리 메뉴 버튼이 위치 2. 중단에는 유저들의 마켓 게시물이 리스트의 형태로 배치 | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** | 리스트의 정보 :  이미지, 제목, 내용, 좋아요 수, 댓글 수, 거래상태, 작성일자, 카테고리 | | |
| **관련요구사항** | 메인 창, Market | | |
| **시나리오** | 1. 앱의 첫 실행이라면, 로그인과 회원정보 입력한 후에 실행 2. 그 이후라면, 앱이 실행될 때 가장 첫 화면 | | |

1. R021 커뮤니티 창

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 커뮤니티 창 | **요구사항구분** | Community |
| 메인 창의 두번째 위치 | | | |
| 1. 상단에는 학교분류, 검색 창 버튼이 위치 2. 중단에는 유저들의 커뮤니티 게시물이 리스트의 형태로 배치 | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** | 리스트의 정보 :  이미지, 제목, 내용, 좋아요 수, 댓글 수, 작성일자 | | |
| **관련요구사항** | 메인 창, Community | | |
| **시나리오** | 1. 유저가 메인 창에서 두번째 탭을 누른다. | | |

1. R022 게시물 최신화 기능

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 게시물 최신화 기능 | **요구사항구분** | Market, Community |
| 게시물 리스트의 최신화를 한다. | | | |
| 1. 최상단에서 하단으로 드래그 하는 이벤트를 감지 2. 드래그 감지 성공 시 최신의 게시물 리스트를 생성한다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** |  | | |
| **관련요구사항** | Market, Community | | |
| **시나리오** | 1. 유저가 마켓 창이나 커뮤니티 창에서 최상단에서 아래로 드래그한다. 2. 게시물 리스트가 최신화 된다. | | |

1. R023 학교 분류 메뉴

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 학교 분류 메뉴 | **요구사항구분** | Market, Community |
| 유저가 선택한 학교 정보에 따라 해당 학교의 게시물만을 접할 수 있는 메뉴 버튼이다. | | | |
| 1. 마켓 창, 커뮤니티 창의 왼쪽 상단에 위치해 있다. 2. 클릭 시 학교들의 정보를 나열한다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** | 현재 미작동 | | |
| **관련요구사항** | Market, Community | | |
| **시나리오** | 1. 유저가 학교 분류 메뉴 버튼을 누른다. 2. 원하는 학교를 선택한다. 3. 마켓 창, 커뮤니티 창에서 선택한 학교의 게시물을 리스트로 나열한다. | | |

1. R024 카테고리 메뉴

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 카테고리 메뉴 | **요구사항구분** | Market, Community |
| 게시물 리스트를 선택한 카테고리에 따라 나열할 수 있다. | | | |
| 1. 마켓 : (전체) 핫 게시물, 음식교환, 물건교환, 대여하기, 퀘스트 2. 커뮤니티 : (전체) 핫 게시물 3. 처음에는 (전체) 게시물이 리스트로 나열 4. 카테고리를 클릭하고, 한 번 더 클릭하면 (전체) 로 나열 | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** |  | | |
| **관련요구사항** | Market, Community | | |
| **시나리오** | 1. 유저가 마켓 창이나 커뮤니티 창에서 원하는 카테고리를 클릭 2. 카테고리에 따른 게시물 리스트가 나열된다. 3. 유저가 눌렀던 카테고리를 한 번 더 누른다. 4. (전체) 게시물의 리스트가 나열된다. | | |

1. R025 검색 창 버튼

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 검색 창 버튼 | **요구사항구분** | Market, Community |
| 찾고 싶은 게시물의 제목을 검색할 수 있는 검색 창으로 넘어가는 버튼이다. | | | |
| 1. 버튼 클릭 시 검색 창으로 넘어간다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** |  | | |
| **관련요구사항** | Market, Community | | |
| **시나리오** | 1. 유저가 마켓 창이나 커뮤니티 창에서 검색 창 버튼을 누른다. 2. 검색 창으로 넘어간다. | | |

1. R026 검색 창

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 검색 창 | **요구사항구분** | Market, Community |
| 찾고자 하는 게시물의 제목을 검색하는 창이다. | | | |
| 1. Market : 뒤로 가기 버튼, 핫 게시물 리스트, 카테고리 버튼 2. Community : 뒤로 가기 버튼, 핫 게시물 리스트 | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** |  | | |
| **관련요구사항** | Market, Community | | |
| **시나리오** | 1. 유저가 마켓 창이나 커뮤니티 창에서 검색 창 버튼을 누른다. 2. 검색 창으로 넘어온다. | | |

1. R027 검색 창 카테고리 버튼

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 검색 창 카테고리 버튼 | **요구사항구분** | Market |
| 검색 창에 구성되어 있고, 클릭 시 각 카테고리의 리스트가 나열된 창으로 이동한다. | | | |
| 1. 마켓 창의 카테고리 메뉴와 같은 아이콘을 쓴다. 2. 클릭 시 해당 카테고리의 게시물 리스트로 나열된 창으로 이동한다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** | 마켓에만 있다. | | |
| **관련요구사항** | 검색 창 | | |
| **시나리오** | 1. 유저가 검색 창으로 들어온다. 2. 유저가 카테고리를 선택하여 클릭한다. 3. 해당 카테고리에 알맞은 게시물 리스트가 나열된 검색 창 카테고리 리스트 창으로 이동한다. | | |

1. R028 검색 창 카테고리 리스트 창

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 검색 창 카테고리 리스트 창 | **요구사항구분** | Market |
| 검색창의 카테고리 버튼에서 클릭된 카테고리의 게시물 리스트로 구성된 창이다. | | | |
| 1. 마켓 창에 있는 리스트와 같은 리스트이다. (카테고리만 선택된 카테고리일 뿐) 2. 작성일자 순서대로 나열한다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** | 모든 카테고리와 같은 창을 이용하지만 리스트에 따라 다르게 출력한다.  리스트의 정보 :  이미지, 제목, 내용, 좋아요 수, 댓글 수, 거래상태, 작성일자, 카테고리 | | |
| **관련요구사항** | Market, 검색 창 | | |
| **시나리오** | 1. 유저가 검색창에서 카테고리를 선택한다. 2. 해당 카테고리의 게시물 리스트로 구성된 창으로 넘어온다. | | |

1. R029 핫 게시물 리스트

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 핫 게시물 리스트 | **요구사항구분** | Market, Community |
| 검색 창에 있는 핫 게시물 리스트이다. | | | |
| 1. 마켓 창에 있는 리스트와 같은 리스트이다. (카테고리만 핫 게시물일 뿐) 2. 작성일자 순서대로 나열한다. 3. 게시물 클릭 시 해당 게시물 창으로 이동한다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** | 마켓 : 3개 까지만 나열한다. 커뮤니티 : 4개 까지만 나열한다.  리스트의 정보 :  이미지, 제목, 내용, 좋아요 수, 댓글 수, 작성일자 (마켓 일 경우 : 카테고리, 거래상태 추가) | | |
| **관련요구사항** | Market, Community, 검색 창 | | |
| **시나리오** | 1. 유저가 마켓 창이나 커뮤니티 창에서 검색 창 버튼을 클릭한다. 2. 검색 창에 있는 핫 게시물 리스트에서 원하는 게시물을 클릭한다. 3. 해당 게시물 창으로 넘어간다. | | |

1. R030 검색 창 뒤로 가기 버튼

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 검색 창 뒤로 가기 버튼 | **요구사항구분** | Market, Community |
| 검색 창 이전의 창으로 돌아간다. | | | |
| 1. 클릭 시 마켓 창이나 커뮤니티 창으로 돌아간다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** |  | | |
| **관련요구사항** | 검색 창 | | |
| **시나리오** | 1. 유저가 검색 창으로 들어온다. 2. 검색 창 뒤로 가기 버튼을 클릭한다. 3. 마켓 창이나 커뮤니티 창으로 돌아간다. | | |

1. R031 검색 입력 칸

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 검색 입력 칸 | **요구사항구분** | Market, Community |
| 찾고 싶은 게시물의 제목을 검색하는 칸이다. | | | |
| 1. 입력 후 자판에 있는 돋보기 모양의 확인 버튼을 클릭하면 검색 결과 창으로 이동한다. 2. 검색 입력 칸 옆에 있는 검색 버튼을 클릭하여도 검색 결과 창으로 이동한다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** |  | | |
| **관련요구사항** | 검색 창 | | |
| **시나리오** | 1. 유저가 검색 창에서 검색 입력 칸에 검색어를 입력한다. 2. 검색 버튼이나 자판의 확인 버튼을 클릭하여 검색 결과 창으로 넘어간다. | | |

1. R032 검색 결과 창

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 검색 결과 창 | **요구사항구분** | Market, Community |
| 검색된 결과의 게시물 리스트를 보여주는 창이다. | | | |
| 1. 마켓 창, 커뮤니티 창에 있는 리스트와 같은 리스트이다. 2. 작성일자 순서대로 나열한다. 3. 게시물 클릭 시 해당 게시물 창으로 이동한다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** | 리스트의 정보 :  이미지, 제목, 내용, 좋아요 수, 댓글 수, 작성일자 (마켓 일 경우 : 카테고리, 거래상태 추가) | | |
| **관련요구사항** | 검색 창 | | |
| **시나리오** | 1. 유저가 검색 창에서 검색어를 검색어 입력 칸에 입력한다. 2. 검색 버튼을 클릭하여 넘어온다. 3. 검색 결과 창에 게시물 리스트를 출력한다. 4. 원하는 게시물을 클릭할 경우 게시물 창으로 넘어간다. | | |

1. R033 마켓 게시물 창

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 마켓 게시물 창 | **요구사항구분** | Market |
| 해당 마켓 게시물의 자세한 정보를 볼 수 있는 창이다. | | | |
| 1. 마켓 게시물의 리스트에서 하나의 게시물을 클릭해서 들어온 창이다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** | 구성요소 : 사진, 게시자 프로필, 좋아요 버튼, 메뉴 버튼, 제목, 내용, 댓글 리스트, 채팅하기 버튼 | | |
| **관련요구사항** | Market, 마켓 창, 검색 창 카테고리 리스트 창, 검색 결과 창, 핫 게시물 리스트, 댓글 목록 리스트 | | |
| **시나리오** | 1. 유저가 관련 요구사항의 마켓 게시물 리스트에서 게시물을 클릭한다. | | |

1. R034 커뮤니티 게시물 창

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 커뮤니티 게시물 창 | **요구사항구분** | Community |
| 해당 커뮤니티 게시물의 자세한 정보를 볼 수 있는 창이다. | | | |
| 1. 커뮤니티 게시물의 리스트에서 하나의 게시물을 클릭해서 들어온 창이다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** | 구성요소 : 사진, 게시자 프로필, 좋아요 버튼, 메뉴 버튼, 제목, 내용, 댓글 리스트 | | |
| **관련요구사항** | Community, 커뮤니티 창, 검색 결과 창, 핫 게시물 리스트, 댓글 목록 리스트 | | |
| **시나리오** | 1. 유저가 관련 요구사항의 커뮤니티 게시물 리스트에서 게시물을 클릭한다. | | |

1. R035 게시물 사진 (스와이프&확대버튼)

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 게시물 사진(스와이프&확대버튼) | **요구사항구분** | Market, Community |
| 해당 게시물의 사진에 대한 기능이다. | | | |
| 1. 게시물의 사진이 1장 이상일 경우 왼쪽으로 스와이핑 할 경우 사진이 다음 사진으로 넘어간다. 2. 사진 자체를 클릭 시 사진을 전체 화면으로 볼 수 있는 창으로 넘어간다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** | 사진의 하단부 중단에 점모양의 페이징 기능을 추가한다. | | |
| **관련요구사항** | 마켓 게시물 창, 커뮤니티 게시물 창 | | |
| **시나리오** | 1. 유저가 게시물을 들어온다. 2. 사진을 스와이핑 하여 게시물의 모든 사진을 본다. 3. 사진 중 하나를 클릭하여 전체 화면으로 볼 수 있는 창으로 넘어간다. | | |

1. R036 전체 사진 창

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 전체 사진 창 | **요구사항구분** | Market, Community |
| 전체 화면으로 보는 사진을 스와이핑 할 수 있다. | | | |
| 1. 전체 화면의 사진을 왼쪽으로 스와이핑 하여 다음 사진을 보여준다. 2. 확대 기능 추가 예정 | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** | 사진의 하단부 중단에 점모양의 페이징 기능을 추가한다. | | |
| **관련요구사항** | 마켓 게시물 창, 커뮤니티 게시물 창 | | |
| **시나리오** | 1. 유저가 게시물 창에 있는 사진을 클릭한다. 2. 사진을 스와이핑 하여 게시물의 모든 사진을 본다. | | |

1. R037 게시자 프로필

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 게시자 프로필 | **요구사항구분** | Market, Community |
| 게시물에서는 게시자의 정보를 볼 수 있다. | | | |
| 1. 게시자의 정보를 나타낸다. 2. 게시자의 프로필을 클릭 시 게시자의 정보가 담겨 있는 창으로 넘어간다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** | 구성요소 : 게시자 프로필. 닉네임, 게시일자 | | |
| **관련요구사항** | 마켓 게시물 창, 커뮤니티 게시물 창, UserInfo | | |
| **시나리오** | 1. 유저가 게시자의 정보를 확인한다. 2. 게시자의 사진을 클릭한다. 3. 게시자의 정보가 있는 창으로 이동한다. | | |

1. R038 좋아요 버튼

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 좋아요 버튼 | **요구사항구분** | Market, Community |
| 게시물의 방문자들에 대한 ‘좋아요’ 이다. | | | |
| 1. 게시물의 좋아요 수가 하트 아이콘 옆에 뜬다. 2. 하트 아이콘 클릭 시 하트 아이콘이 바뀌며 좋아요 수도 증가한다. 3. 한 번 더 클릭 시 좋아요 취소가 된다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** | ‘좋아요’는 방문자의 Uid를 리스트로 저장 받는데, 크기가 10 이상이 되면 핫 게시물이 “O”로 바뀌게 된다. | | |
| **관련요구사항** | 마켓 게시물 창, 커뮤니티 게시물 창 | | |
| **시나리오** | 1. 유저가 게시물의 좋아요 버튼을 누른다. 2. 취소도 가능하다. 3. 핫 게시물에 대한 정보가 “O” 가 되었다면 핫 게시물 리스트에서도 볼 수 있게 된다. | | |

1. R039 게시물 창 메뉴 버튼

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 게시물 창 메뉴 버튼 | **요구사항구분** | Market, Community |
| 게시물에 있는 메뉴 버튼이다. | | | |
| 1. 방문자일 경우에는 신고하기, 채팅하기 버튼이 있다. (커뮤니티에는 채팅하기가 없다) 2. 게시자일 경우에는 수정하기, 삭제하기 버튼이 있다. 3. 삭제 할 경우 마켓은 “MARKETS”에서 커뮤니티는 “COMMUNITY”에서 완전히 삭제된다. 4. 신고하기 기능 추가 예정 | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** | 게시물을 삭제할 시 댓글과, 스토리지도 전부 함께 삭제되어야 한다. | | |
| **관련요구사항** | 마켓 게시물 창, 커뮤니티 게시물 창 | | |
| **시나리오** | 1. 방문자가 채팅하기를 누르면 게시자와의 채팅방으로 넘어간다. 2. 게시자가 수정하기를 누르면 게시물 수정 창으로 이동하고, 삭제하기 버튼을 누르면 게시물이 삭제된다. | | |

1. R040 게시물 수정 창

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 게시물 수정 창 | **요구사항구분** | Market, Community |
| 게시물의 수정을 위한 창이다. | | | |
| 1. 글쓰기 창과 구성이 같다. 2. 수정이 완료된다면 “MARKETS” 나 “COMMUNITY”에 알맞은 Uid에 저장되어야 한다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** | 게시물의 정보가 게시물 수정 창에 최초로 들어가면 기입되어 있어야한다. | | |
| **관련요구사항** | 마켓 게시물 창, 커뮤니티 게시물 창, 글쓰기 창 | | |
| **시나리오** | 1. 게시자가 게시물 창 메뉴 버튼에서 수정하기를 누른다. 2. 게시물 수정 창에서 수정을 한다. 3. 수정된 사항이 적용된다. | | |

1. R041 제목&본문

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 제목&본문 | **요구사항구분** | Market, Community |
| 게시물의 제목과 내용이다. | | | |
| 1. 게시물의 제목과 내용을 요약되지 않고 알맞게 받아온다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** | 너무 긴 제목은 글자 수 제한을 설정한다. | | |
| **관련요구사항** | 마켓 게시물 창, 커뮤니티 게시물 창 | | |
| **시나리오** | 1. 게시물에 사진이 있는 경우 유저가 사진의 하단에서 확인할 수 있다. | | |

1. R042 댓글 목록 리스트

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 댓글 목록 리스트 | **요구사항구분** | Comment |
| 유저들의 입력된 댓글들이 리스트로 표현된다. | | | |
| 1. 댓글의 정보를 “COMMENT”에서 받아온다. 2. 댓글을 작성일자 순서대로 나열한다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** | 댓글의 정보 : 작성자의 프로필 사진, 닉네임, 댓글 내용, 작성일자 | | |
| **관련요구사항** | 마켓 게시물 창, 커뮤니티 게시물 창 | | |
| **시나리오** | 1. 유저가 게시물 창의 하단부에서 확인할 수 있다. | | |

1. R043 댓글 메뉴 버튼

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 댓글 메뉴 버튼 | **요구사항구분** | Comment |
| 댓글 목록 리스트의 댓글마다 있는 메뉴 버튼이다. | | | |
| 1. 댓글의 작성자일 경우 : 삭제하기 버튼 2. 댓글의 작성자가 아닐 경우 : 신고하기 버튼 3. 삭제할 경우 “COMMENTS”에서 완전히 삭제된다. 4. 신고하기 기능 추가 예정 | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** |  | | |
| **관련요구사항** | 댓글 목록 리스트 | | |
| **시나리오** | 1. 댓글의 작성자이면 메뉴버튼을 클릭하고 삭제할 수 있다. 2. 댓글의 작성자가 아니라면 신고하기를 할 수 있다. | | |

1. R044 댓글 입력 칸

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 댓글 입력 칸 | **요구사항구분** | Comment |
| 댓글 입력 칸은 화면의 하단부에 고정되어 있다. | | | |
| 1. 댓글의 내용을 입력할 수 있다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** | 너무 긴 제목은 글자 수 제한을 설정한다. | | |
| **관련요구사항** | 마켓 게시물 창, 커뮤니티 게시물 창 | | |
| **시나리오** | 1. 유저가 댓글을 입력한다. | | |

1. R045 댓글 등록 버튼

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 댓글 등록 버튼 | **요구사항구분** | Comment |
| 입력한 댓글을 등록 할 수 있는 버튼이다. | | | |
| 1. 댓글이 “COMMENT” 컬렉션에 등록된다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** | 댓글이 등록되면 댓글 목록 리스트에 즉시 반영된다. | | |
| **관련요구사항** | 마켓 게시물 창, 커뮤니티 게시물 창 | | |
| **시나리오** | 1. 유저가 댓글 입력 칸에 댓글을 입력한 뒤 댓글 등록 버튼을 눌러서 등록한다. | | |

1. R046 채팅하기 버튼

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 채팅하기 버튼 | **요구사항구분** | Market |
| 게시자와 채팅을 하기 위한 버튼이다. | | | |
| 1. 댓글 입력 칸 왼쪽에 위치한다. 2. 마켓에서만 채팅이 가능하므로 커뮤니티 게시물 창의 댓글 입력 칸에는 없다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** | 채팅하기 버튼을 누르면 자판이 사라짐과 동시에 사라진다. | | |
| **관련요구사항** | 마켓 게시물 창 | | |
| **시나리오** | 1. 유저가 물물교환하고 싶은 게시자의 게시물 창에 들어왔다. 2. 교환을 위해 채팅하기 버튼을 누른다. 3. 채팅방으로 이동한다. | | |

1. R047 글쓰기 선택 창

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 글 쓰기 선택 창 | **요구사항구분** | Market, Community |
| 메인 창의 세번째에 위치한다. 메인 창에서 세번째 탭을 누르면 이동된다. | | | |
| 1. 마켓의 게시물을 쓸 것인지, 커뮤니티의 게시물을 쓸 것인지 고를 수 있다. 2. 선택 시 해당 글쓰기 창으로 넘어간다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** |  | | |
| **관련요구사항** | 메인 창 | | |
| **시나리오** | 1. 유저가 메인 창에서 세번째 탭을 누른다. 2. 마켓과 커뮤니티 중 하나를 선택한다. 3. 해당 글쓰기 창으로 넘어간다. | | |

1. R048 글쓰기 창

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 글쓰기 창 | **요구사항구분** | Market, Community |
| 게시물의 글쓰기를 위한 창이다. | | | |
| 1. 입력된 정보는 마켓은 “MARKETS”에 커뮤니티는 “COMMUNITY”에 저장된다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** | 구성요소 : 사진 선택 버튼, 제목/내용 입력 칸, 등록 버튼, 선택한 사진 리스트 및 개수 (마켓 글쓰기 창 일 경우 : 카테고리 추가) | | |
| **관련요구사항** | Market, Community, 글쓰기 선택 창 | | |
| **시나리오** | 1. 유저가 글쓰기 선택 창에서 원하는 글쓰기를 선택한다. 2. 글쓰기 창으로 넘어온다. | | |

1. R049 글쓰기 창 뒤로 가기 버튼

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 글쓰기 창 뒤로 가기 버튼 | **요구사항구분** | Market, Community |
| 글쓰기 창에 있으며, 이전의 창으로 돌아가는 버튼이다. | | | |
| 1. 클릭 시 글쓰기 선택 창으로 돌아간다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** |  | | |
| **관련요구사항** | 글쓰기 창, 글 쓰기 선택 창 | | |
| **시나리오** | 1. 유저가 글쓰기 창에 들어와서 글쓰기 창 뒤로 가기 버튼을 누른다. 2. 글쓰기 선택 창으로 이동한다. | | |

1. R050 사진 선택 버튼

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 사진 선택 버튼 | **요구사항구분** | Market, Community |
| 글쓰기 창에 있으며, 앨범으로 이동하는 버튼이다. | | | |
| 1. 사진 선택 버튼에 선택된 사진의 개수를 표시한다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** |  | | |
| **관련요구사항** | 글쓰기 창, 사진 다중 선택 창 | | |
| **시나리오** | 1. 유저가 글쓰기 창에 들어와서 사진을 고르기 위해 누른다. | | |

1. R051 사진 다중 선택 창

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 사진 다중 선택 창 | **요구사항구분** | Market, Community |
| 사진 선택 버튼을 클릭하여 넘어온 앨범의 사진들이 나열된 창이다. | | | |
| 1. 앨범의 접근 권한을 받아온다. 2. 사진을 나열한다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** | 디렉토리 별 사진을 불러오는 것을 구현한다. | | |
| **관련요구사항** | 사진 선택 버튼, 사진 다중선택 창 뒤로 가기 버튼, 사진 다중 선택 기능, 사진 선택 확인 버튼 | | |
| **시나리오** | 1. 유저가 글쓰기 창에서 사진 선택 버튼을 누른다. | | |

1. R052 사진 다중선택 창 뒤로 가기 버튼

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 사진 다중선택 창 뒤로 가기 버튼 | **요구사항구분** | Market, Community |
| 사진 다중 선택 창에서 뒤로 가기 위한 버튼이다. | | | |
| 1. 클릭 시 글쓰기 창으로 돌아간다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** |  | | |
| **관련요구사항** | 사진 다중 선택 창 | | |
| **시나리오** | 1. 유저가 사진 다중 선택 창에서 사진 다중 선택창 뒤로 가기 버튼을 누른다. 2. 글쓰기 창으로 돌아간다. | | |

1. R053 사진 다중 선택 기능

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 사진 다중 선택 기능 | **요구사항구분** | Market, Community |
| 사진 다중 선택 창에서 사진을 다중으로 선택할 수 있는 기능이다. | | | |
| 1. 사진마다의 오른쪽 상단의 체크 표시를 클릭하여 5장까지 다중 선택할 수 있다. 2. 사진을 클릭 시 확대 화면으로 볼 수 있다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** |  | | |
| **관련요구사항** | 사진 다중 선택 창 | | |
| **시나리오** | 1. 유저가 사진 다중선택 창에서 사진을 다중으로 선택할 수 있다. 2. 혹은 사진을 클릭하여 확대된 화면으로 볼 수 있다. | | |

1. R054 사진 다중 선택 확인 버튼

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 사진 다중 선택 확인 버튼 | **요구사항구분** | Market, Community |
| 사진 다중 선택 창에서 선택한 사진들을 확인하기 위한 버튼이다. | | | |
| 1. 클릭 시 글쓰기 창으로 돌아간다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** | 돌아간 글쓰기 창에서는 사진 선택 버튼의 숫자에 선택한 사진의 숫자가 표시된다. | | |
| **관련요구사항** | 사진 다중 선택 창 | | |
| **시나리오** | 1. 유저가 사진 다중 선택 창에서 사진을 선택하고 확인 버튼을 누른다. 2. 글쓰기 화면으로 돌아간다. | | |

1. R055 카테고리 버튼

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 카테고리 버튼 | **요구사항구분** | Market |
| 마켓 게시물의 카테고리를 선택할 수 있다. | | | |
| 1. 마켓의 게시물 글쓰기에서 카테고리를 선택할 수 있다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** |  | | |
| **관련요구사항** | Market, 글쓰기 창 | | |
| **시나리오** | 1. 유저가 마켓 글쓰기 창에서 카테고리를 선택한다. | | |

1. R056 제목 & 본문 입력 칸

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 제목 & 본문 입력 칸 | **요구사항구분** | Market, Community |
| 글쓰기 창에서 제목과 내용을 입력하기 위한 칸이다. | | | |
| 1. 제목과 내용을 입력 받는다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** | 내용을 입력 받을 때에 화면을 넘어가지 않게 설계한다. | | |
| **관련요구사항** | Market, Community, 글쓰기 선택 창 | | |
| **시나리오** | 1. 유저가 글쓰기 창에서 제목과 내용을 입력한다. | | |

1. R057 등록 버튼

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 등록 버튼 | **요구사항구분** | Market, Community |
| 글쓰기 창에 있으며, 입력 받은 정보를 등록하기 위한 버튼이다. | | | |
| 1. 입력된 정보는 마켓은 “MARKETS”에 커뮤니티는 “COMMUNITY”에 저장된다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** |  | | |
| **관련요구사항** | Market, Community, 글쓰기 창 | | |
| **시나리오** | 1. 유저가 글쓰기 창에서 게시물 정보를 입력한다. 2. 등록 버튼을 클릭한다. | | |

1. R058 채팅 목록 창

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 채팅 목록 창 | **요구사항구분** | Chat |
| 메인 창의 네번째에 있으며, 채팅방 목록이 있는 창이다. | | | |
| 1. 모든 채팅방의 정보들이 리스트의 형태로 나타난다. 2. 채팅방들 중 원하는 채팅방을 클릭하면 채팅 창으로 들어간다. | | | |
| **위험요소** | 오류가 빈번히 발생 | | |
| **설계 시 고려사항** | 채팅방 정보 : 상대방 프로필, 닉네임, 최근 채팅, 시간, 읽지 않은 메시지 개수 | | |
| **관련요구사항** | 메인 창, 채팅 목록 창 삭제, 나가기 확인 기능, 채팅 창 | | |
| **시나리오** | 1. 유저가 메인 창에서 네번째 탭을 누른다. 2. 원하는 채팅방이 있다면 클릭하여 해당 채팅 창으로 들어간다. | | |

1. R059 채팅 목록 창 스와이핑 삭제

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 채팅 목록 창 스와이핑 삭제 | **요구사항구분** | Chat |
| 채팅 목록 창에서 채팅 방을 삭제할 수 있다. | | | |
| 1. 채팅 목록 중 하나를 왼쪽으로 스와이핑 하면 삭제할 수 있다. 2. 해당 채팅방의 정보는 “ROOMS”에서 완전히 삭제된다. 3. 해당 채팅방의 채팅은 “MESSAGE”에서 완전히 삭제된다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** | 스와이핑 해서 나타난 삭제 버튼의 색은 빨간색으로 한다. | | |
| **관련요구사항** | Chat, 채팅 목록 창 | | |
| **시나리오** | 1. 유저가 채팅 목록 창에서 삭제할 채팅 방을 정한다. 2. 왼쪽으로 스와이핑 한다. 3. 삭제하기를 눌러 삭제한다.. | | |

1. R060 나가기 확인 기능

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 나가기 확인 기능 | **요구사항구분** | Chat |
| 채팅을 하던 상대방이 나갔음을 알려주는 표시이다. | | | |
| 1. 상대방이 나갔을 대 확인 메시지를 받을 수 있다. 2. 한 명만 나갔을 경우 상대방에게 채팅방 정보가 남아있다. 3. 모두 나갔을 경우 해당 채팅방의 정보는 “ROOMS”에서 완전히 삭제된다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** | 나가기 확인 기능 메시지 : (닉네임) 님이 나갔습니다. | | |
| **관련요구사항** | Chat, 채팅 목록 창 | | |
| **시나리오** | 1. 유저1이 채팅방을 나간다. 2. 유저2에게 유저1이 나갔음을 알리는 메시지를 보낸다. 3. 유저2도 나간다. 4. 채팅방의 정보가 완전히 삭제된다. | | |

1. R061 채팅 창

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 채팅 창 | **요구사항구분** | Chat |
| 물물교환을 원하는 유저와 1:1 채팅을 할 수 있는 방이다. | | | |
| 1. 두 사람은 실시간으로 채팅을 주고받을 수 있다. 2. 게시자의 채팅방에만 예약 버튼과 거래완료 버튼이 존재한다. 3. 두 유저가 나누는 대화는 “ROOMS”의 “MESSAGE” 컬렉션에 저장된다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** |  | | |
| **관련요구사항** | 채팅 목록 창, 채팅 방 뒤로 가기 버튼, 채팅 방 삭제하기 메뉴, 채팅 방 마켓 정보, 채팅 방 예약 버튼, 채팅 방 거래완료 버튼, 대화목록, 사진, 앨범 이동 버튼, 대화 입력 칸, 대화 보내기 버튼, (Market : 채팅하기 버튼, 메뉴 버튼) | | |
| **시나리오** | 1. 유저가 메인 창에서 네번째 탭을 누른다. 2. 원하는 채팅방이 있다면 클릭하여 해당 채팅 창으로 들어간다. 3. 물물교환의 상태를 조정하거나, 채팅을 나눈다. | | |

1. R062 채팅 방 뒤로 가기 버튼

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 채팅 방 뒤로 가기 버튼 | **요구사항구분** | Chat |
| 채팅 방에 있는 뒤로 가기 버튼이다. | | | |
| 1. 클릭 시 채팅 방에서 뒤로 가기가 되며, 채팅 목록 창으로 이동한다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** |  | | |
| **관련요구사항** | Chat, 채팅 목록 창, 채팅 창 | | |
| **시나리오** | 1. 유저가 채팅 방에 들어간다. 2. 채팅 방 뒤로 가기 버튼을 클릭한다. 3. 채팅 목록 창으로 돌아온다. | | |

1. R063 채팅 방 삭제하기 메뉴

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 채팅 방 삭제하기 메뉴 | **요구사항구분** | Chat |
| 현재 들어온 채팅방을 삭제하게 해준다. | | | |
| 1. 파이어스토어 “ROOMS”에서 현재 룸의 UID를 이용하여 찾는다. 2. “RoomModel\_USER\_OUT” 이란 필드에서 채팅방을 나간 유저의 UID를 1->0으로 만들어준다. 3. “RoomModel\_USER\_OUT”가 0이 된 유저는 채팅방을 자동으로 나가지게 되고 채팅 목록에서도 삭제된다. 4. 상대 유저에게는 “(닉네임)이 나갔습니다.” 라는 텍스트가 전송된다. 5. “RoomModel\_USER\_OUT”에 두 유저의 값이 전부 0일경우 현 채팅 방은 완전히 삭제된다. | | | |
| **위험요소** | 완전히 삭제되는 것이 아니므로 마켓에서 다시 채팅을 누르게 되면 원래 있던 채팅방으로 들어가진다. | | |
| **설계 시 고려사항** | 현재 유저의 UID, 상대방 UID, 채팅 방 UID, RoomModel\_USER\_OUT | | |
| **관련요구사항** | 채팅 창, 채팅 목록 창 | | |
| **시나리오** | 1. 유저는 자신이 삭제하고 싶은 채팅 방을 들어가 메뉴 버튼 중 삭제를 클릭한다. 2. 삭제한 유저는 자동으로 채팅 방을 나가게 되고 채팅목록에 없어진다. 3. 상대 유저에게 “(닉네임)이 나갔습니다.” 란 텍스트가 전해진다. 4. 상대 유저 또한 삭제를 했다면 완전히 채팅 방이 삭제된다. | | |

1. R064 채팅 방 마켓 정보

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 채팅 방 마켓 정보 | **요구사항구분** | Chat |
| 채팅 상단부에 어떤 마켓에서 채팅을 했는지 마켓의 정보를 표현한다. | | | |
| 1. 유저의 UID에 따라 생성되는 마켓 정보 창이 달라진다. 2. 마켓 필드 값 중 “MarketModel\_Host\_Uid”에 따라 게스트와 호스트를 구분한다. 3. 유저가 게스트였을 경우 마켓의 정보만을 표현한다. 4. 유저가 호스트였을 경우 마켓 정보와 “예약하기”, ”거래완료” 란 버튼이 생성된다. 5. 마켓의 정보가 사라졌을 경우 삭제된 마켓이란 공지를 띄워준다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** | 마켓 UID, 현재 유저의 UID, 상대방 UID, MarketModel\_Host\_Uid | | |
| **관련요구사항** | 채팅 창 | | |
| **시나리오** |  | | |

1. R065 채팅 방 마켓 정보 예약 버튼

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 채팅 방 마켓 정보 예약 버튼 | **요구사항구분** | Chat |
| 채팅의 접속자가 호스트 일 경우 예약버튼을 선택 가능하다. | | | |
| 1. 유저는 현재 채팅을 하는 유저와 마켓을 예약하고 싶을 경우 예약하기 버튼을 클릭 시 거래완료 버튼이 활성화된다. 2. 현재 마켓의 필드 값 중 “MarketModel\_reservation”를 상대 유저의 UID로 변경한다. 3. 만약 예약을 취소하고 싶다면 다시 클릭하면 거래완료 버튼은 비활성화 되며 “MarketModel\_reservation”의 값을 없애준다. 4. 예약하기 버튼을 클릭하면 마켓의 정보 또한 “예약중” 이라는 문구를 띄워준다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** | 상대방 UID, 마켓 UID, MarketModel\_reservation | | |
| **관련요구사항** | 채팅 창, 마켓 창, 채팅 방 마켓 정보, 마켓 게시물 창 | | |
| **시나리오** | 1. 호스트인 유저가 상대방 유저와의 마켓 거래를 예약하고 싶을 경우 예약하기 버튼을 클릭한다. 2. 마켓의 정보가 예약하기로 바뀐다. | | |

1. R066 채팅 방 거래 완료 버튼

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 채팅 방 거래 완료 버튼 | **요구사항구분** | Chat |
| 채팅의 접속자가 호스트 일 경우 거래완료버튼을 선택 가능하다. | | | |
| 1. 유저는 현재 채팅을 하는 유저와 현 마켓이 거래완료 되었을경우 거래완료 버튼을 클릭 시 거래완료 와 예약하기가 비활성화 된다. 2. 현재 마켓의 필드 값 중 “MarketModel\_deal”에 상대 유저의 UID로 변경한다. 3. 예약하기, 거래완료 버튼 전부 비활성화 시킨다. 4. 상대방의 리뷰창을 띄운다. | | | |
| **위험요소** | 거래 완료 버튼을 클릭 시 취소가 불가능하다. | | |
| **설계 시 고려사항** | 상대방 UID, 마켓 UID, MarketModel\_deal | | |
| **관련요구사항** | 채팅 창, 마켓 창, 채팅 창 마켓 정보, 마켓 게시물 창, 예약 버튼, 리뷰 창 | | |
| **시나리오** | 1. 호스트인 유저가 상대방 유저와의 마켓 거래를 거래 완료하고 싶을 때 거래완료 버튼을 클릭한다. 2. 마켓의 정보가 거래완료로 바뀐다. 3. 리뷰창이 발생한다. | | |

1. R067 채팅 방 현재 유저 대화 목록 & 사진

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 채팅 방 현재 유저 대화 목록 & 사진 | **요구사항구분** | Chat |
| 유저는 상대방과 대화를 텍스트와 사진을 준다. | | | |
| 1. 유저는 상대 유저에게 텍스트와 사진을 보여줄 수 있다. 2. “ROOMS” 컬렉션에서 해당 채팅 방의 컬렉션 “MESSAGE”의 정보 중 “MessageModel\_UserUid” 이 UID가 같은 메시지 정보만을 리스트형태로 시간순으로 보여준다. 3. 현재 유저의 채팅 기록들은 오른쪽 노란 말꼬리 이미지에 텍스트가 담겨 보여 진다. 4. 현재 유저의 사진은 오른쪽 작은 이미지 형식으로 보인다. | | | |
| **위험요소** | 사진 확대 기능 추가 예정 | | |
| **설계 시 고려사항** | 현재 유저의 UID, ROOM의 UID, MessageModel\_UserUid , MESSAGE 정보 | | |
| **관련요구사항** | 채팅 방 | | |
| **시나리오** |  | | |

1. R068 채팅 방 상대방 유저 대화 목록 & 사진

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 채팅 방 상대방 유저 대화 목록 & 사진 | **요구사항구분** | Chat |
| 유저는 상대방과 대화를 텍스트와 사진을 받는다. | | | |
| 1. 유저는 상대 유저의 텍스트와 사진을 볼 수 있다. 2. “ROOMS” 컬렉션에서 해당 채팅 방의 컬렉션 “MESSAGE”의 정보 중 “MessageModel\_UserUid” 이 UID가 같은 메시지 정보만을 리스트형태로 시간순으로 보여준다. 3. 상대방 유저의 채팅 기록들은 왼쪽 흰 말꼬리 이미지에 텍스트가 담겨 보여 진다. 4. 상대방 유저의 사진은 왼쪽 작은 이미지 형식으로 보인다. | | | |
| **위험요소** | 사진 확대 기능 추가 예정 | | |
| **설계 시 고려사항** | 현재 유저의 UID, 상대방 UID, ROOM의 UID, 해당 맞는 디렉토리의 모든 정보들 | | |
| **관련요구사항** | 채팅 방 | | |
| **시나리오** |  | | |

1. R069 앨범 사진 보내기버튼 및 앨범 창 이동

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 앨범 사진 보내기버튼 및 앨범 창 이동 | **요구사항구분** | Chat |
| 클릭 시 앨범의 사진을 불러온 창을 띄우며 원하는 사진을 클릭 시 상대방에게 사진을 보여줄 수 있다. | | | |
| 1. 항상 하단부 왼쪽에 고정된다. 2. 유저는 상대방에게 사진을 보여주고 싶을 경우 버튼을 클릭 시 앨범을 불러와 원하는 사진을 클릭한다. 3. 해당 컬렉션이 “RoomModel\_ImageCount”이 +1되고 스토리지에 저장된다. 4. 해당 사진을 다시 스토리지의 ROOM이란 디렉토리에서 같은 번호를 찾아 보여준다. 5. 사진을 저장하고 띄우는 동안에 로딩 화면을 띄운다. 6. 완료 시 로딩 화면을 없애 준다. | | | |
| **위험요소** | * 사진의 이름으로 스토리지의 경로를 찾기 때문에 오류 시 아예 띄울 수가 없다. * 사진 확대 기능 추가 예정 * 사진을 보내지 않을 시 메시지 중 Null이 생긴다. | | |
| **설계 시 고려사항** | 현재 유저의 UID, 상대방 UID, ROOM의 UID, 스토리지 사진 경로 | | |
| **관련요구사항** | 채팅 방, 채팅 방 현재 유저 대화 목록 & 사진 | | |
| **시나리오** | 1. 유저는 하단 왼쪽부에 있는 사진 아이콘을 클릭한다. 2. 앨범에서 보여줄 사진을 선택한다. 3. 오른쪽 채팅 목록 중 작은 사진의 형태로 사진을 보여줄 수 있다. | | |

1. R070 대화 입력 칸

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 대화 입력 칸 | **요구사항구분** | Chat |
| 상대방에게 텍스트를 보내고 싶을 때 원하는 텍스트를 입력창에 기입한다. | | | |
| 1. 항상 채팅 하단부 가운데에 위치된다. 2. 유저는 텍스트 창을 누를 시 자동으로 자판부가 하단에서 올라온다. 3. 50자까지 제한을 건다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** |  | | |
| **관련요구사항** | 채팅 방, 채팅 방 현재 유저 대화 목록 & 사진 | | |
| **시나리오** | 1. 유저는 하단부에 있는 채팅 창을 누른다. 2. 자판이 올라오고 자판을 이용하여 원하는 텍스트를 입력한다. | | |

1. R071 대화 보내기 버튼

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 대화 보내기 버튼 | **요구사항구분** | Chat |
| 클릭 시 입력해 놓았던 채팅 입력 칸의 텍스트를 상대방에게 보내준다. | | | |
| 1. 항상 오른쪽 하단부에 고정된다. 2. 클릭 시 입력해 놓았던 채팅 입력 칸의 텍스트를 MESSAGE 라는 모델로 객체화 한다. 3. “ROOMS”에서 해당 채팅방을 찾아 MASSAGE 컬렉션을 추가시켜준다. 4. 오른쪽 노란색 말꼬리 아이콘에 텍스트를 추가하며 상대방에게 보낸다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** | 현재 유저의 UID, 상대방 UID, ROOM의 UID | | |
| **관련요구사항** | 채팅 방, 채팅 방 현재 유저 대화 목록 & 사진, 대화 입력 칸 | | |
| **시나리오** | 1. 채팅 이력 창에 입력되어 있는 것을 확인한다. 2. 오른쪽 아이콘을 눌러 채팅을 보낸다. | | |

1. R072 내 정보 창

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 내 정보 창 | **요구사항구분** | User Info |
| 현재 유저의 정보를 알 수 있다. | | | |
| 1. 유저의 UID로 “USERS”란 컬렉션에 자신의 정보를 찾는다. 2. 해당 칸에 맞추어 정보가 입력된다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** | 현재 유저의 UID, USERS 컬렉션 | | |
| **관련요구사항** |  | | |
| **시나리오** | 1. 메인 중 하단 바의 마지막의 버튼을 클릭한다. 2. 내 정보를 볼 수 있는 창으로 이동된다. | | |

1. R073 상대 정보 창

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 상대 정보 창 | **요구사항구분** | User Info |
| 원하는 상대 유저 정보를 볼 수 있다. | | | |
| 1. 상대 프로필을 클릭 시 프로필 창으로 이동된다. 2. 프로필을 선택한 유저의 UID를 불러와 “USERS” 컬렉션에서 해당 정보를 불러와 해당 칸에 맞추어 정보가 입력된다. | | | |
| **위험요소** | * 마켓, 커뮤니티 정보 창에서만 정보 확인 가능 * 채팅방에서도 클릭가능 하게 만들어야함 * 프로필 이미지만이 아닌 닉네임을 눌러도 가능하게 만들어야 함 | | |
| **설계 시 고려사항** |  | | |
| **관련요구사항** | 마켓 정보 창, 커뮤니티 정보 창 | | |
| **시나리오** | 1. 원하는 유저의 프로필을 클릭한다. 2. 해당 유저의 정보를 볼 수 있는 창으로 들어간다. | | |

1. R074 회원 정보 변경 버튼

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 회원 정보 변경 버튼 | **요구사항구분** | User Info |
| 클릭 시 회원 정보 창으로 이동된다. | | | |
| 1. 버튼 클릭 시 회원정보를 변경할 수 있는 창으로 넘어간다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** | 현재 유저의 UID | | |
| **관련요구사항** | 내 정보 창, 회원 정보 수정 창 | | |
| **시나리오** | 1. 버튼을 클릭한다. 2. 회원 정보 수정 창으로 이동한다. | | |

1. R075 회원 정보 수정 창

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 회원정보 수정 창 | **요구사항구분** | User Info |
| 현재 유저의 입력되어 있었던 정보 | | | |
| 1. 현재 유저의 UID를 “USERS” 컬렉션에서 찾아서 정보들을 기본 세팅으로 맞춘다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** | 현재 유저 UID | | |
| **관련요구사항** | 회원 정보 변경 버튼 | | |
| **시나리오** |  | | |

1. R076 회원 정보 수정 창 뒤로 가기

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 회원 정보 수정 창 뒤로 가기 | **요구사항구분** | User Info |
| 클릭 시 내 회원정보 창으로 이동된다. | | | |
| 1. 클릭 시 회원정보 수정창에서 내 회원 정보 창으로 이동된다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** |  | | |
| **관련요구사항** | 회원정보 창, 회원정보 수정 창 | | |
| **시나리오** | 1. 회원정보 수정창에서 뒤로 가기 버튼을 클릭한다. 2. 내 회원정보 창으로 이동된다. | | |

1. R077 프로필 이미지 변경 & 앨범 초기화 버튼

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 프로필 이미지 변경 & 앨범 초기화 버튼 | **요구사항구분** | User Info |
| 프로필 사진을 클릭 시 앨범으로 이동된다. 사진을 선택 시 사진이 변경되고 초기화 버튼을 클릭 시 기본 이미지로 세팅 된다. | | | |
| 1. “USERS” 컬렉션의 필드 값 중 “UserModel\_ProfileImage”의 정보를 받아 기본 사진으로 세팅한다. 2. 앨범의 사진을 불러와 원하는 사진을 클릭 시 사진과 “UserModel\_ProfileImage”의 정보를 바꿔준다. 3. 초기화 버튼을 클릭 시 “UserModel\_ProfileImage”의 정보를 NULL로 바꾼다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** | 현재 유저의 UID UserModel\_ProfileImage | | |
| **관련요구사항** | 회원 정보 수정 창 | | |
| **시나리오** | 1. 저장해 놓았던 프로필 사진을 클릭한다. 2. 앨범 사진 중 변경하고 싶었던 사진을 클릭한다. 3. 초기화를 원할 경우 오른쪽 상단의 초기화 버튼을 클릭한다. 4. 회원 정보 수정 창의 프로필 사진이 변경되어 있다. | | |

1. R078 닉네임 변경 칸

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 닉네임 변경 칸 | **요구사항구분** | User Info |
| 닉네임 변경 칸을 클릭하여 자신의 닉네임을 바꾸준다. | | | |
| 1. “USERS” 컬렉션의 필드 값 중 “UserModel\_Nickname”의 정보를 받아 기본 닉네임으로 세팅한다. 2. 클릭 후 자신이 원하는 닉네임을 입력한다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** | 현재 유저의 UID UserModel\_Nickname | | |
| **관련요구사항** | 회원 정보 수정 창 | | |
| **시나리오** | 1. 닉네임 칸을 클릭한다. 2. 자판을 이용하여 변경하고 싶은 닉네임을 입력한다. | | |

1. R079 생년 월일 변경 칸

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 생년 월일 변경 칸 | **요구사항구분** | User Info |
| 클릭 시 앨범의 사진을 불러온 창을 띄우며 원하는 사진을 클릭 시 상대방에게 사진을 보여줄 수 있다. | | | |
| 1. “USERS” 컬렉션의 필드 값 중 “UserModel\_BirthDay”의 정보를 받아 세팅한다. 2. 원하는 날짜로 변경하여 입력한다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** | 현재 유저의 UID, UserModel\_BirthDay | | |
| **관련요구사항** | 회원정보 수정 창 | | |
| **시나리오** | 1. 생년 월일을 원하는 날짜로 변경한다. | | |

1. R080 학교 변경 칸

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 학교 변경 칸 | **요구사항구분** | User Info |
| 클릭 시 원하는 학교로 변경할 수 있다. (홍익대학교, 고려대학교) | | | |
| 1. “USERS” 컬렉션의 필드 값 중 “UserModel\_University”의 정보를 받아 세팅한다. 2. 만약 0일경우 홍익대학교 1일경우 고려대학교로 변경한다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** | 현재 유저의 UID, UserModel\_University | | |
| **관련요구사항** | 회원정보 수정 창 | | |
| **시나리오** | 1. 학교 메뉴를 클릭한다. 2. 학교를 원하는 학교로 변경한다. | | |

1. R081 수정 버튼

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 수정 버튼 | **요구사항구분** | User Info |
| 버튼 클릭 시 해당 변경 한 모든 정보들로 변경된다. | | | |
| 1. 버튼을 클릭한다. 2. 파이어스토어 “USERS” 컬렉션의 필드에서 해당 UID의 모든 필드 값을 유저가 원하는 정보로 변경한다. 3. 사진을 변경했다면 스토리지도 변경한다. 4. 로딩 창을 띄운다. 5. 완료되었다면 로딩창을 지우고 내 정보창으로 넘어간다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** | 현재 유저의 UID, USERS | | |
| **관련요구사항** | 채팅 방, 채팅 방 현재 유저 대화 목록 & 사진 | | |
| **시나리오** | 1. 버튼을 클릭한다. 2. 정보가 변경되고 내 정보 창으로 이동된다. | | |

1. R082 진행중 버튼

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 진행중 버튼 | **요구사항구분** | User Info |
| 클릭 시 현재 유저가 마켓에서 거래중인 게시물이 있는 창으로 넘어가게 된다. | | | |
| 1. 버튼을 클릭한다. 2. 거래중 창으로 넘어간다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** | 현재 유저의 UID | | |
| **관련요구사항** | 진행중 창, 내 정보 창 | | |
| **시나리오** | 1. 버튼을 클릭한다. 2. 진행중 창으로 이동된다. | | |

1. R083 진행중 창

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 진행중 창 | **요구사항구분** | User Info |
| 현재 유저의 거래중인 마켓의 정보들을 리스트 형태로 보여준다. 또한 원하는 마켓을 누를 시 마켓정보창으로 넘어가게 된다. | | | |
| 1. “MARKETS” 컬렉션에서 “MarketModel\_reservation”이 내 UID가 같은 모든 마켓의 정보를 가져온다 2. 리스트 형태로 마켓의 정보를 가져온다. 3. 마켓을 클릭 시 마켓정보 창으로 이동된다. | | | |
| **위험요소** | * 판매자는 볼 수 없고 구매자만 뜬다. | | |
| **설계 시 고려사항** | 현재 유저의 UID | | |
| **관련요구사항** | 거래 중 버튼 | | |
| **시나리오** |  | | |

1. R084 거래완료 버튼

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 거래완료 버튼 | **요구사항구분** | User Info |
| 클릭 시 현재 유저가 마켓에서 거래완료인 게시물이 있는 창으로 넘어가게 된다. | | | |
| 1. 버튼을 클릭한다. 2. 거래 완료 창으로 넘어간다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** | 현재 유저의 UID | | |
| **관련요구사항** | 거래완료 창, 내 정보 창 | | |
| **시나리오** | 1. 버튼을 클릭한다. 2. 거래 완료 창으로 넘어간다. | | |

1. R085 거래완료 창

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 거래완료 창 | **요구사항구분** | User Info |
| 현재 유저의 거래완료인 마켓의 정보들을 리스트 형태로 보여준다. 또한 원하는 마켓을 누를 시 마켓정보창으로 넘어가게 된다. | | | |
| 1. “MARKETS” 컬렉션에서 “MarketModel\_deal”이 내 UID가 같은 모든 마켓의 정보를 가져온다 2. 리스트 형태로 마켓의 정보를 가져온다. 3. 마켓을 클릭 시 마켓정보 창으로 이동된다. | | | |
| **위험요소** | * 판매자는 볼 수 없고 구매자만 뜬다. | | |
| **설계 시 고려사항** | 현재 유저의 UID | | |
| **관련요구사항** | 거래완료 버튼 | | |
| **시나리오** |  | | |

1. R086 작성한 게시물 버튼

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 작성한 게시물 버튼 | **요구사항구분** | User Info |
| 클릭 시 현재 유저가 마켓에서 작성한 게시물이 있는 창으로 넘어가게 된다. | | | |
| 1. 버튼을 클릭한다. 2. 거래 완료 창으로 넘어간다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** | 현재 유저의 UID | | |
| **관련요구사항** | 거래완료 창, 내 정보 창 | | |
| **시나리오** | 1. 버튼을 클릭한다. 2. 거래 완료 창으로 넘어간다. | | |

1. R087 작성한 게시물 창

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 작성한 게시물 창 | **요구사항구분** | User Info |
| 현재 유저의 작성한 게시물의 마켓의 정보들을 리스트 형태로 보여준다. 또한 원하는 마켓을 누를 시 마켓정보창으로 넘어가게 된다. | | | |
| 1. “MARKETS” 컬렉션에서 “MarketModel\_host”이 내 UID가 같은 모든 마켓의 정보를 가져온다 2. 리스트 형태로 마켓의 정보를 가져온다. 3. 마켓을 클릭 시 마켓정보 창으로 이동된다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** | 현재 유저의 UID | | |
| **관련요구사항** | 채팅 방, 채팅 방 현재 유저 대화 목록 & 사진 | | |
| **시나리오** |  | | |

1. R088 리뷰 평가 현황

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 리뷰 평가 현황 | **요구사항구분** | User Info |
| 현재 유저가 마켓에서 거래 후 받은 리뷰 현황을 표시한다. | | | |
| 1. “USERS” 컬렉션에서 “UserModel\_kindReviewList”, “UserModel\_correctReviewList”, “UserModel\_completeReviewList”, “UserModel\_badReviewList”의 개수를 가져온다. 2. 해당 목록에 맞는 곳에 개수를 표현한다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** | 현재 유저의 UID | | |
| **관련요구사항** | 내 정보 창 | | |
| **시나리오** |  | | |

1. R089 작성된 리뷰 버튼

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 작성된 리뷰 버튼 | **요구사항구분** | User Info |
| 클릭 시 작성된 리뷰 창이 나타난다. | | | |
| 1. 버튼을 클릭한다. 2. 작성된 리뷰 창으로 넘어간다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** | 현재 유저의 UID | | |
| **관련요구사항** | 내 정보 창 | | |
| **시나리오** |  | | |

1. R090 작성된 리뷰 창

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 작성된 리뷰 창 | **요구사항구분** | User Info |
| 현재 유저에게 다른 유저가 후기를 텍스트로 작성한 리스트형태로 표현된다. | | | |
| 1. “USERS” 컬렉션에서 “REVIEW” 컬렉션의 모든 정보를 가져온다. 2. 리스트 형태로 리뷰의 정보를 가져온다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** | 현재 유저의 UID, REVIEW | | |
| **관련요구사항** | 작성된 리뷰 버튼 | | |
| **시나리오** |  | | |

1. R091 리뷰 창

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 리뷰 창 | **요구사항구분** | User Info |
| 마켓에서 거래되었던 마켓의 정보가 보여 지며 상대방의 평가지표를 입력할 수 있는 창이다. | | | |
| 1. 마켓에서 거래되었을 경우 “MARKET” 컬렉션의 모든 정보를 가져와 정보가 제공되며 리뷰의 평가지표가 나타난다. 2. 평가지표는 4개중 하나만 고를 수 있으며 “불쾌함”을 클릭 했을 경우 자동으로 후기 창이 나타난다. 3. 나타나는 시점은 거래완료를 눌렀을 때 와 앱을 다시 재실행 했을 때 나타난다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** | 현재 유저의 UID, MARKET, REVIEW | | |
| **관련요구사항** | 작성된 리뷰 버튼 | | |
| **시나리오** | <판매자 시점>   1. 채팅방에서 거래 완료 버튼을 클릭한다. 2. 리뷰창이 나타난다. 3. 리뷰를 작성하지 않을 시 앱을 재실행 되면 나타난다.   <구매자 시점>   1. 앱을 재실행한다. 2. 리뷰가 생성된다. 3. 리뷰를 작성하지 않을 시 앱을 재실행 되면 나타난다. | | |

1. R092 리뷰 평가 지표

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 리뷰 평가 지표 | **요구사항구분** | User Info |
| 평가 지표 중 한가지만 고를 수 있다. | | | |
| 1. 친절함, 정확함, 완벽함, 불쾌감 중 하나만 고를 수 있다. 2. 불쾌감을 선택했을 경우 텍스트를 적을 수 있는 후기 창이 뜬다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** | 현재 유저의 UID, REVIEW | | |
| **관련요구사항** | 리뷰 창, 후기 창 | | |
| **시나리오** |  | | |

1. R093 후기 버튼

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 후기 버튼 | **요구사항구분** | User Info |
| 클릭 시 후기 작성 창이 나타난다. | | | |
| 1. 버튼 클릭 시 후기 창이 나타난다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** | 현재 유저의 UID, REVIEW | | |
| **관련요구사항** | 리뷰 창, 후기 창 | | |
| **시나리오** |  | | |

1. R094 후기 창

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 후기 창 | **요구사항구분** | User Info |
| 후기를 텍스트로 입력 가능하다. | | | |
| 1. 텍스트를 작성 후 확인 버튼을 클릭 시 자신이 작성되어 있던 리뷰창에 텍스트가 나타난다. 2. 불쾌감을 클릭했을 경우 자동으로 나타난다. 3. 취소를 클릭했을 경우 화면에서 후기 창이 사라진다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** | 현재 유저의 UID, REVIEW | | |
| **관련요구사항** | 리뷰 창, 후기 창 | | |
| **시나리오** | 1. 텍스트를 입력한다. 2. 확인을 누를 시 화면에서 사라지면서 입력한 텍스트가 리뷰창에 나타난다. 3. 취소 시 화면에서 사라진다. | | |

1. R095 리뷰 확인 버튼

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 리뷰 창 | **요구사항구분** | User Info |
| 클릭 시 리뷰 창에 입력되었던 정보들을 입력하려 했던 유저의 정보에 기입된다. | | | |
| 1. 버튼을 클릭 시 상대방 UID를 이용하여 “USERS” 컬렉션에서 해당 정보를 찾는다. 2. 입력된 정보가 평가 지표라면 “UserModel\_kindReviewList”, “UserModel\_correctReviewList”, “UserModel\_completeReviewList”, “UserModel\_badReviewList”에 작성자의 UID와 함께 입력된다. 3. 만약 후기까지 했다면 REVIEW에 추가된다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** | 현재 유저의 UID, REVIEW | | |
| **관련요구사항** | 리뷰 창 | | |
| **시나리오** |  | | |

1. R096 로그아웃 하기

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 로그아웃 하기 | **요구사항구분** | Main |
| 클릭 시 해당 유저의 정보를 디바이스에서 삭제하여 로그아웃 할 수 있다. | | | |
| 1. 버튼을 클릭 시 디바이스에서 데이터를 삭제한다. 2. 자동 로그인이 되지 않는다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** | 현재 유저의 UID | | |
| **관련요구사항** | 내 정보 창, 카카오 로그인, 구글 로그인 | | |
| **시나리오** |  | | |

1. R097 구글 로그인

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 구글 로그인 | **요구사항구분** | Manage |
| Authentication에 구글 화원을 등록하고 자동 로그인 가능하게 한다. | | | |
| 1. 구글 사이트를 통한 로그인 접속하게 한다. 2. Authentication에 등록이 되어 있지 않다면 회원가입을 한다. 3. 자동 로그인을 가능하게 한다. | | | |
| **위험요소** | * 정보를 좀더 받을 수 있는지 조사해야 한다. 할 수 있다면 회원정보에 자동 기입이 가능하다. | | |
| **설계 시 고려사항** | 현재 유저의 UID, 구글 유저 | | |
| **관련요구사항** | 로그인 창 | | |
| **시나리오** | 1. 구글 로그인 버튼을 누른다. 2. 회원가입 및 자동 로그인을 가능하게 해준다. | | |

1. R098 로그아웃 하기

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 카카오 로그인 | **요구사항구분** | Manage |
| 카카오에 회원임을 확인하고 이메일의 정보를 받아 Authentication에 이메일 회원가입 및 자동로그인을 가능하게 한다. | | | |
| 1. 카카오 사이트를 통한 로그인 접속하게 한다. 2. Authentication에 등록이 되어 있지 않다면 이메일을 받아와 등록시켜준다. 3. 자동 로그인을 가능하게 한다. | | | |
| **위험요소** | * 카카오 자체 로그인이 아니므로 보안이 취약함 * 비밀번호 세팅은 1234567890 로 세팅 되어 있다. | | |
| **설계 시 고려사항** | 현재 유저의 UID 카카오 유저 | | |
| **관련요구사항** | 내로그인 창 | | |
| **시나리오** | 1. 카카오 로그인 버튼을 누른다.   회원가입 및 자동 로그인을 가능하게 해준다. | | |

1. R099 프로필 입력

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 프로필 입력 | **요구사항구분** | Manage |
| 최초 로그인 시에만 발생하며 유저의 정보를 FireStore “USERS”에 담는다. | | | |
| 1. 최초 로그인 시 회원가입전 회원 정보를 받는다. 2. “USERS” 필드에 문서를 생성하게 되고 정보를 담는다. 3. 프로필 이미지를 받았다면 Storage에 경로를 생성해 담는다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** | 현재 유저의 UID USERS | | |
| **관련요구사항** | 회원정보 창 | | |
| **시나리오** |  | | |

1. R100 게시물 리스트

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 게시물 리스트 | **요구사항구분** | Manage |
| Market 창 과 Community 창에 정보를 FireStore “MARKET” 과 “COMMUNITY”의 정보를 불러와 담는다. | | | |
| 1. Market창에는 “MARKET”의 모든정보를 Community창에는 “COMMUNITY” 정보를 담는다. 2. 하단 바를 클릭 시 최신화 해준다. 3. 아래로 스와이핑 시 다시 최신 정보를 불러온다. | | | |
| **위험요소** | MARKET COMMUNITY | | |
| **설계 시 고려사항** |  | | |
| **관련요구사항** | 마켓 창, 커뮤니티 창 | | |
| **시나리오** |  | | |

1. R101 채팅 리스트

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 채팅 리스트 | **요구사항구분** | Manage |
| FireStore에 “ROOMS”의 모든 정보를 가져와 그중 자신의 고우 번호가 있는 정보들만 가져온다. | | | |
| 1. “USERS”에서 모든 정보를 가져온다. 2. “ROOMS”에 “”RoomModel\_USERS\_OUT”에 해당 정보 중 유저의 UID와 같은 정보만을 가져온다. 3. 상대방 UID와 맞는 USERS 정보를 가져온다. | | | |
| **위험요소** | * 설계가 조금 문제가 있음 다시 바꿀 필요 있음 | | |
| **설계 시 고려사항** | 현재 유저의 UID ROOMS, USERS | | |
| **관련요구사항** | 채팅 목록 창 | | |
| **시나리오** |  | | |

1. R102 유저 정보

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 유저 정보 | **요구사항구분** | Manage |
| FireStore에 “USERS”의 유저의 UID의 정보를 가져와 보여준다. | | | |
| 1. “USERS”에서 원하는 유저의 UID로 정보를 찾는다. 2. 해당 창에 맞게 정보를 보여준다. | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** | 현재 유저의 UID 상대 유저의 UID | | |
| **관련요구사항** | 내 정보 창 상대 정보 창 | | |
| **시나리오** |  | | |

1. R103 알림 기능

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 알림 기능 | **요구사항구분** | Manage |
| 채팅, 게시물 댓글, 리뷰가 등록이 되었다면 자동으로 디바이스에 알림을 띄워준다. | | | |
| 1. 해당 사건 발생시 파이어 베이스 메세징 서비스를 이용하여 해당 Token을 가진 디바이스에 메시지를 보내준다. | | | |
| **위험요소** | * 안되는 디바이스가 많다. 버전에 따른 것일 확률이 높다. | | |
| **설계 시 고려사항** | Token | | |
| **관련요구사항** | 채팅 창, 댓글 창, 리뷰 창 | | |
| **시나리오** |  | | |

1. R104 신고하기

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 신고하기 | **요구사항구분** | Manage |
| 추가 예정 | | | |
| 추가 예정 | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** |  | | |
| **관련요구사항** |  | | |
| **시나리오** |  | | |

1. R105 공지 사항

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 공지사항 | **요구사항구분** | Main |
| 추가 예정 | | | |
| 추가 예정 | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** |  | | |
| **관련요구사항** |  | | |
| **시나리오** |  | | |

1. R106 자주 하는 질문

|  |  |  |  |
| --- | --- | --- | --- |
| **요구사항 명** | 자주하는 질문 | **요구사항구분** | Main |
| 추가 예정 | | | |
| 추가 예정 | | | |
| **위험요소** |  | | |
| **설계 시 고려사항** |  | | |
| **관련요구사항** |  | | |
| **시나리오** |  | | |

1. 제품 요구사항

##### 사용성 요구사항

* 마켓 창, 커뮤니티 창, 글쓰기 창, 채팅 목록 창, 내 정보 창과 같이 탭을 사용해서 원하는 항목으로 쉽게 이동할 수 있게 한다.
* 마켓 창에서는 마켓 게시물의 리스트, 카테고리 선택, 검색 창 버튼을 제공한다.
* 커뮤니티 창에서는 커뮤니티 게시물의 리스트, 검색 창 버튼을 제공한다.
* 글 쓰기 창에서는 마켓이나 커뮤니티 게시물을 쓸 수 있는 창을 선택할 수 있는 창을 제공한다.
* 채팅 목록 창에서는 채팅 목록 리스트를 제공한다.
* 내 정보 탕에서는 유저의 회원정보를 보여주어야 한다.

##### 효율성 요구사항

* 게시물의 리스트를 연달아 잘 생성해야 한다.
* 채팅이 실시간으로 진행되어야 한다.

ㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡ

#### 조직 요구사항

##### 구현 요구사항

* Mern(MongoDB, Express.js, React.js, Node.js) stack 사용한다.
* 데이터베이스는 MongoDB를 사용한다.
* 웹 서버는
* 웹 구현은 React.js를 사용한다.

#### 외부 요구사항

##### 상호 운용성 요구사항

* + - 다른 소프트웨어와의 연동은 없다.

##### 윤리적 요구사항

* + - 모든 연령층이 사용 가능하다.

##### 법적 요구사항

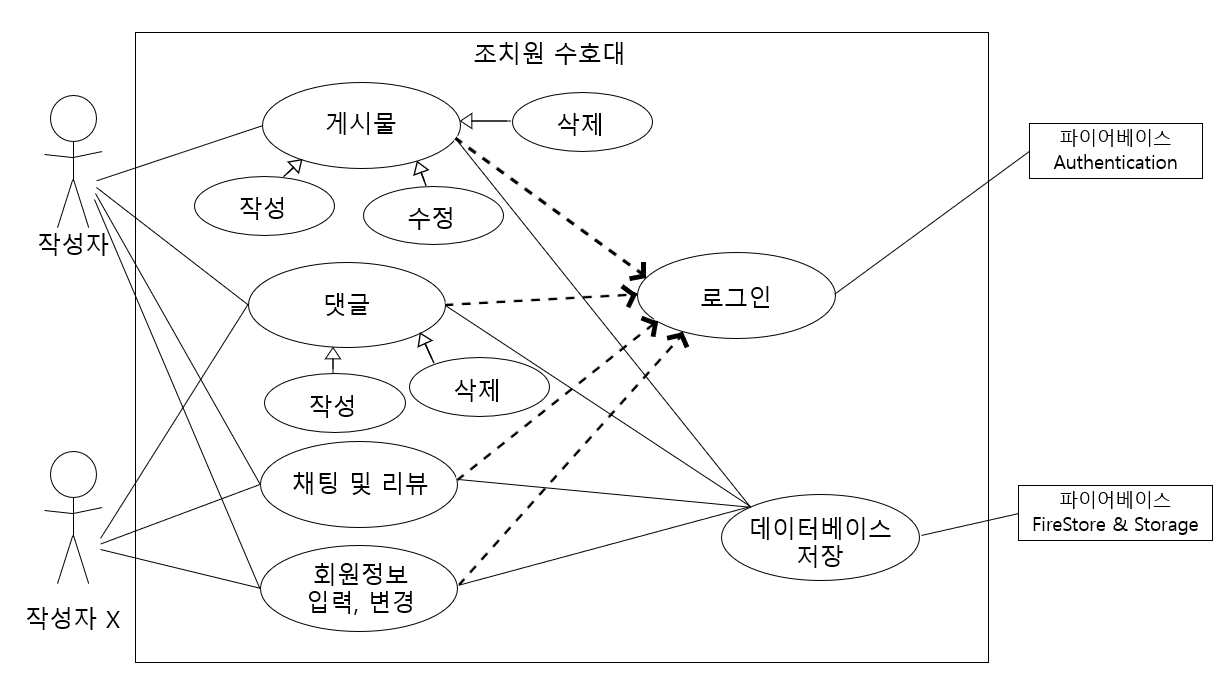
* + - DBMS는 파이어베이스를 사용한다. 이미지에 대한 저장은 Storage에 저장한다.

ㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡ

### 서브시스템 목록

|  |  |
| --- | --- |
| 서브시스템명 | 서브시스템 설명 |
| App | 어플리케이션 자체에서 운용되는 시스템 |
| 파이어베이스 | 어플리케이션 내에서의 데이터 관리 |

* + 1. **유스케이스 다이어그램(UCD)**



ㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡ

* + 1. **유스케이스 기술서**

#### 뉴스 순위 열람

|  |  |  |  |
| --- | --- | --- | --- |
| 유스케이스 명 | 뉴스 순위 열람 | 관련 엑터 | 사용자 |
| 유스케이스 개요 | 사용자가 뉴스 순위를 열람한다. | | |
| 시전 조건 | 해당 없음 | | |
| 사후 조건 | 많이 본 뉴스 순위 순서로 기사들이 웹 페이지에 나타난다. | | |
| 시작 화면 | 웹 페이지 Home UI | | |
| 흐름 | 사용자 | | Web |
| 기본 흐름 (시나리오) | 1. 많이 본 뉴스 순위에 맞게 기사를 Home화면에 보여준다. | |  |
| 대체 흐름 | 많이 본 뉴스 순위에 맞는 기사들이 없는 경우 관련 메시지를 출력한다. | | |

1. 원문 기사 열람

|  |  |  |  |
| --- | --- | --- | --- |
| 유스케이스 명 | 원문 기사 열람 | 관련 엑터 | 사용자 |
| 유스케이스 개요 | 사용자가 원문 기사를 열람하기 위해 기사를 선택한다. | | |
| 시전 조건 | 사용자가 선택한 조건에 맞는 기사를 RSS가 1개 이상 제공해야한다. | | |
| 사후 조건 | 사용자가 선택한 기사의 원문이 웹 페이지에 나타난다. | | |
| 시작 화면 | 웹 페이지 News UI | | |
| 흐름 | 사용자 | | Web |
| 기본 흐름 (시나리오) | 1. 열람할 기사 선택 | |  |
|  | | 2. 해당 기사의 원문을 News 화면에  보여준다. |
| 대체 흐름 | 사용자가 선택한 기사의 원문을 불러오지 못한 경우 관련 메시지를 출력한  다. | | |
|  |  | | |

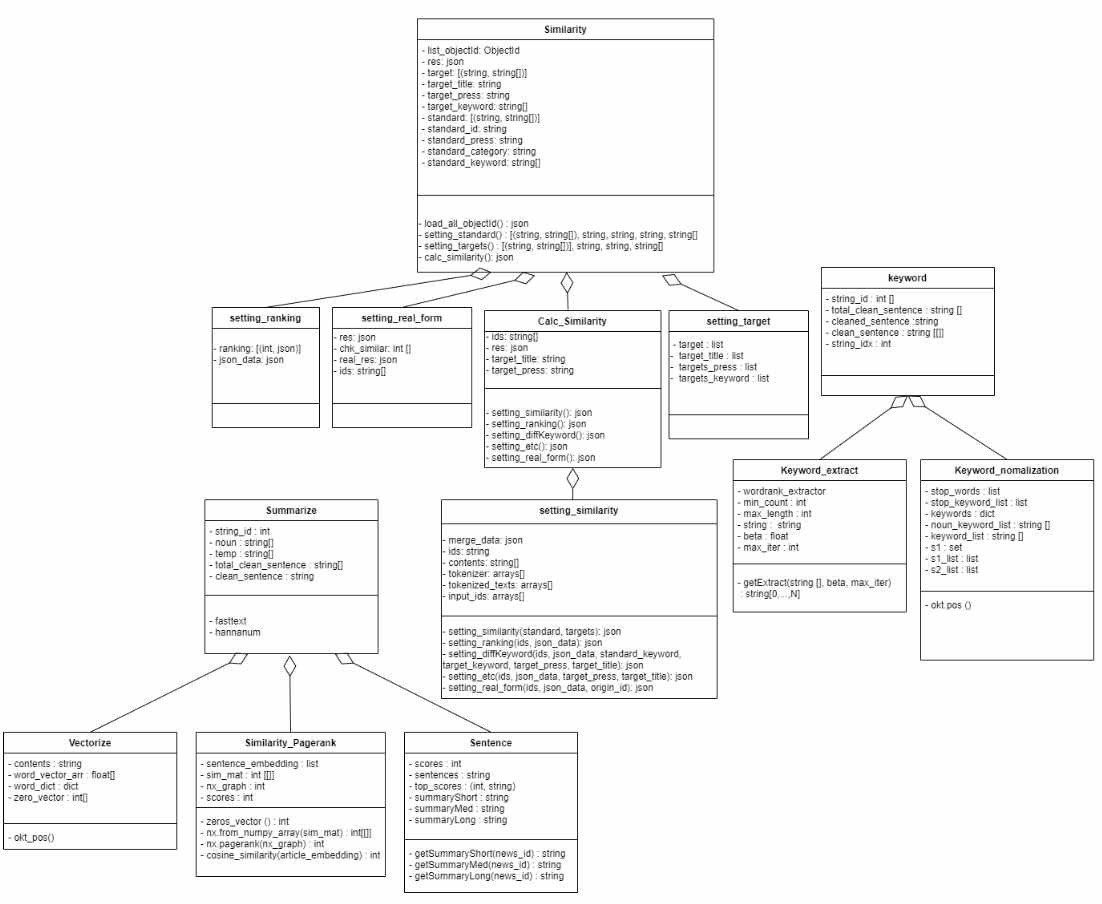
ㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡ

# 설계 단계

## 클래스 다이어그램

ㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡ

* + 1. **클래스 다이어그램**



* + 1. **클래스 기술서**

1. Similarity

: 웹 사이트에서 기사들을 가져와서 처리한 뒤 저장하는 클래스

|  |  |  |
| --- | --- | --- |
| 멤버 명 | 자료형 | 설명 |
| list\_objectId | ObjectId | 분석한 뉴스 데이터 id 리스트 |
| res | json | 분석한 뉴스와 비교군 뉴스들의 유사도, 랭킹, 키워드 차이점을 저장한 json |
| target | [(string, string[]) | 비교군 뉴스의 id와 문장 단위로 나눠진 내용 리스트 |
| target\_title | string | 비교군 뉴스의 제목 |
| target\_press | string | 비교군 뉴스의 언론사 |
| target\_keyword | string[] | 비교군 뉴스의 키워드 |
| standard | [(string, string[]) | 분석한 뉴스의 id와 문장 단위로 나눠진 내용 리스트 |
| standard\_id | string | 분석한 뉴스의 id |
| standard\_press | string | 분석할 뉴스의 언론사 |
| standard\_category | string | 분석할 뉴스의 카테고리 |
| standard\_keyword | string[] | 분석할 뉴스의 키워드 리스트 |

|  |  |
| --- | --- |
| 메소드 명 | 설명 |
| load\_all\_objectId() json: | collected collection에서 모든 기사의 id를 추출하여 저장한  다. |
| setting\_standard() : [(string, string[]), string, string, string, string[]) | 분석한 뉴스 기사의 id와 뉴스 본문 문장별 리스트를 튜플 형  태로 추출한다. 그리고 뉴스 기사의 id, 언론사, 카테고리, 키  워드를 추출한다. |
| setting\_targets() : [(string, string[]), string, string, string[])] | 비교군 뉴스 리스트들의 id와 뉴스 본문 문장별 리스트를 튜  플 형태로 추출한다. 그리고 뉴스 기사의 제목, 언론사, 키워  드를 추출한다. |
| calc\_similarity() : json | similarity collection에 매핑 되는 데이터를 추출 및 삽입한다. |

ㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡ

## 시퀀스 다이어그램

### 서브시스템 목록

|  |  |  |
| --- | --- | --- |
| 서브시스템 ID | 서브시스템명 | 서브시스템 설명 |
| SS\_SVR\_010 | 기사 수집 | 설정된 크롤링 주기에 따라 기사를 수집하고, 전처리 및  중복 기사를 검사하는 시스템 |
| SS\_SVR\_021 | 키워드 추출 | 기사에서 키워드를 추출하는 시스템 |
| SS\_SVR\_022 | 문서 요약 | 기사 내용을 요약하는 시스템 |
| SS\_SVR\_023 | 유사도 분석 | 기사들 사이의 유사도를 분석하는 시스템 |
| SS\_WEB\_030 | 뉴스 순위 | 카테고리/언론사 별 뉴스 순위를 제공하는 시스템 |
| SS\_WEB\_041 | 기사 제공 | 기사의 원문과 분석 리포트를 제공하는 시스템 |
| SS\_WEB\_042 | 분석 리포트 메일링 | 기사의 분석 리포트를 메일링 해주는 시스템 |
| SS\_WEB\_050 | Today’s 제공 | 카테고리 별 오늘의 키워드/주요 기사 요약, 시간별 키워  드 추이를 제공하는 시스템 |

* + 1. **시퀀스 다이어그램(Sequence Diagram)**

#### (1) SS\_SVR\_010

|  |  |  |  |
| --- | --- | --- | --- |
| SD ID | SS\_SVR\_010 | 시퀀스 명 | 기사 수집 |
| 관련 서브시스템 ID |  | 관련 서브시스템명 |  |
|  | | | |

* + 1. **시퀀스 목록**

|  |  |  |
| --- | --- | --- |
| 시퀀스 ID | 시퀀스 명 | 시퀀스 설명 |
| SS\_SVR\_010 | 기사 수집 | 1. 현재 시간의 정각 여부를 확인한다. 이 때 현재 시간이 55   분이면 다음의 순서를 진행하고, 아닐 시 1분 후 현재 시 간을 확인한다.   1. 선택한 언론사의 RSS를 이용하여 뉴스 데이터를 크롤링한 다. 2. 크롤링 결과 값을 전처리하여 데이터를 정리한다. 3. 크롤링 시간에 따라 다음 항목을 수행한다. 4. 오후 11시 55분일 시: json파일이 없을 시 생성하고, 있다 면 기존의 json파일에 데이터를 덮어씌운다. 5. 오후 11시 55분이 아닐 시: json파일에 새로운 데이터 항 목을 추가한다.   5. 최종 선발 된 데이터로 json을 업데이트 및 Server로 전송  한다. Server는 해당 값을 저장한다. |
| SS\_SVR\_021 | 키워드 추출 | 1. Server에 저장된 데이터 중 뉴스 아이디, 뉴스 내용을 요청   한다.   1. 뉴스 내용에서 뉴스 키워드를 추출한다. 2. 추출된 키워드를 Server 전송하고, Server는 해당 값을 저장   한다. |
| SS\_SVR\_022 | 문서 요약 | 1. Server에 저장된 데이터 중 뉴스 아이디, 뉴스 내용을 요청   한다.   1. 뉴스 내용을 30%로 요약한다. 2. 요약된 뉴스 내용을 Server에 전송하고, Server는 해당 값을 저장한다. 3. 뉴스 내용을 60%로 요약한다. 4. Flask API는 요약된 뉴스 내용을 Server에 전송하고, Server 는 해당 값을 저장한다. 5. 뉴스 내용을 90%로 요약한다. 6. 요약된 뉴스 내용을 Server에 전송하고, Server는 해당 값을   저장한다. |
| SS\_SVR\_023 | 유사도 분석 | 1. Server에 저장된 데이터 중 카테고리, 뉴스 아이디, 뉴스 내   용을 요청한다.   1. 카테고리 별 기사들의 유사도 분석을 진행한다. 2. 유사도 결과를 Server에 전송하고, Server는 해당 값을 저장   한다. |

|  |  |  |
| --- | --- | --- |
| 시퀀스 ID | 시퀀스 명 | 시퀀스 설명 |
| SS\_WEB\_030 | 뉴스 순위 | 1. 기본 값으로 카테고리는 전체, 언론사는 조선일보가 선택되   어있다.   1. Web은 Server에 카테고리와 언론사에 맞는 뉴스 순위를 요 청한다. 2. Server는 Web에게 뉴스 순위 요청에 대한 response를 보 낸다. 3. 사용자는 원할 때마다 카테고리와 언론사를 선택할 수 있 다. 4. 사용자가 카테고리와 언론사를 새로 선택할 때마다 2-3 과   정을 반복한다. |
| SS\_WEB\_041 | 기사 제공 | 1. Web은 Server에 기사 원문과 분석 리포트를 요청한다. 2. Server는 Web에게 기사 원문과 분석 리포트 요청에 대한   response를 보낸다. |
| SS\_WEB\_042 | 분석 리포트 메일링 | 1. 사용자가 원할 경우, 사용자는 Email 버튼을 클릭해 분석   리포트 메일 서비스를 요청할 수 있다.   1. Web은 사용자에게 메일 주소를 입력하라고 요청한다. 2. 사용자는 메일 주소를 입력한다. 3. Web은 사용자가 입력한 메일 주소로 메일 전송을 Server에 요청한다. 4. Server는 메일을 전송한다. 5. Server는 Web에게 메일 전송 여부 response를 보낸다. 6. Web은 사용자에게 메일 전송 여부 response를 보낸다. |
| SS\_WEB\_050 | Today’s 제공 | 1. 기본 값으로 카테고리는 전체가 선택되어 있다. 2. Web은 Server에 오늘의 키워드, 오늘의 주요 기사 요약, 시 간별 키워드 추이를 요청한다. 3. Server는 Web의 요청에 대한 response를 보낸다. 4. 사용자는 원할 때마다 카테고리를 선택할 수 있다. 5. 사용자가 카테고리를 선택할 때마다 2-3 과정을 반복한다. |

## 사용자 인터페이스 설계서

* + 1. **사용자 인터페이스 구조도**

|  |  |  |
| --- | --- | --- |
| Level1 | Level 2 | Level 3 |
| 로그인 화면 | 회원정보 입력 화면 | (앨범 화면) |
|  | 마켓 리스트 화면 |  |
| 마켓 리스트 화면 | 검색창 화면 (마켓) | 검색 결과 화면 (마켓) |
|  |  | 각 카테고리 리스트 화면 |
|  | 마켓 게시물 | 호스트 정보 화면 |
| 커뮤니티 리스트 화면 | 검색창 화면 (커뮤니티) | 검색 결과 화면 (커뮤니티) |
|  | 커뮤니티 게시물 | 호스트 정보 화면 |
| 게시물 작성 화면 | 마켓 작성 화면 | (앨범 화면) |
|  | 커뮤니티 작성 화면 | (앨범 화면) |
| 채팅방 목록 화면 | 채팅방 화면 | (앨범 화면) |
| 내 정보 화면 | 회원정보 수정 |  |
|  | 진행중 게시물 리스트 |  |
|  | 거래완료 게시물 리스트 |  |
|  | 내가 작성한 게시물 리스트 |  |
|  | 나에게 작성된 리뷰 |  |
|  | 로그아웃 |  |

* + 1. **사용자 인터페이스 목록**

|  |  |
| --- | --- |
| 화면 ID | 화면 명 |
| SS\_SC\_000 | 로그인 화면 |
| SS\_SC\_000-01 | 회원정보 입력 화면 |
|  | 마켓 리스트 화면 (= SS\_SC\_001) |
| SS\_SC\_001 | 마켓 리스트 화면 |
| SS\_SC\_001-01 | 검색창 화면 (마켓) |
| SS\_SC\_001-01-1 | 검색 결과 화면 (마켓] |
| SS\_SC\_001-01-2 | 각 카테고리 리스트 화면 |
| SS\_SC\_001-02 | 마켓 게시물 |
| SS\_SC\_001-02-1 | 호스트 정보 화면 |
| SS\_SC\_002 | 커뮤니티 리스트 화면 |
| SS\_SC\_002-01 | 검색창 화면 (커뮤니티) |
| SS\_SC\_002-01-1 | 검색 결과 화면 (커뮤니티) |
| SS\_SC\_002-02 | 커뮤니티 게시물 |
|  | 호스트 정보 화면 (= SS\_SC\_001-02-1) |
| SS\_SC\_003 | 게시물 작성 화면 |
| SS\_SC\_003-01 | 마켓 작성 화면 |
| SS\_SC\_003-02 | 커뮤니티 작성 화면 |
| SS\_SC\_004 | 채팅방 목록 화면 |
| SS\_SC\_004-01 | 채팅방 화면 |
| SS\_SC\_005 | 내 정보 화면 |
| SS\_SC\_005-01 | 회원정보 수정 |
| SS\_SC\_005-02 | 진행중 게시물 리스트 |
| SS\_SC\_005-03 | 거래완료 게시물 리스트 |
| SS\_SC\_005-04 | 내가 작성한 게시물 리스트 |
| SS\_SC\_005-05 | 나에게 작성된 리뷰 |
| SS\_SC\_005-06 | 로그아웃 |

* + 1. **화면 상세 설계**

1. SS\_SC\_000

|  |  |  |  |
| --- | --- | --- | --- |
| **화면ID** | SS\_SC\_000 | **화면 명** | 로그인 화면 |
| **화면개요** | * 사용자가 앱을 실행하면 가장 먼저 보게 되는 화면 | | |
|  | | | |
| **처리 내용** | | | |
| * 구글 로그인 시 계정 선택하게 함 / 카카오 로그인 시 연결되어 있는 계정으로 자동 로그인 | | | |
| **기술적 고려사항** | | | |
| - 로그인을 했으면 로그아웃을 하기 전까지 다시 이 로그인 화면이 실행될 일이 없도록 자동 로그인 구현  - 로그인 후 회원정보가 있으면 바로 마켓 리스트 화면으로, 없으면 회원정보 입력 화면으로 이동한다.  - 알림을 위해 기기의 토큰 정보를 함께 넘겨준다. 회원정보가 있어도 현재 토큰 정보가 저장되어 있는 것과 다르면 새로 저장 | | | |

1. SS\_SC\_000-01

|  |  |  |  |
| --- | --- | --- | --- |
| **화면ID** | SS\_SC\_000-01 | **화면 명** | 회원정보 입력 화면 |
| **화면개요** | * 회원정보의 입력을 받는 화면 | | |
|  | | | |
| **처리 내용** | | | |
| * 닉네임은 따로 입력하지 않으면, 로그인 이메일의 @아래 부분을 제외한 부분을 닉네임으로 함 * 사진 선택 좀…클릭 시 앨범으로 접근하여 프로필로 사용할 사진을 정하기 위해 앨범으로 접근 * 생일은 스피너로 지정할 수 있음, 학교는 정하지 않으면 홍익대학교로 함 * 앨범에서 초기화 버튼을 누르면 선택했던 사진을 초기화 할 수 있다. | | | |
| **기술적 고려사항** | | | |
| * 입력 받은 회원정보를 잘 저장해야 한다. * 저장소(앨범)의 접근 권한을 받아야한다. | | | |

1. SS\_SC\_001

|  |  |  |  |
| --- | --- | --- | --- |
| **화면ID** | SS\_SC\_001 | **화면 명** | 마켓 리스트 화면 |
| **화면개요** | * 사용자가 로그인 후 앱을 실행하면 가장 먼저 보게 되는 화면 * 마켓의 게시물들이 나열되어 있는 리스트 | | |
|  | | | |
| **처리 내용** | | | |
| * 검색창 버튼 클릭 시 검색창 화면으로 이동 * 게시물 클릭 시 마켓 게시물 화면으로 이동 * 각 카테고리 클릭 시 해당 카테고리의 게시물 리스트로 중단에 위치한 리스트 변경 * 게시물의 정보 : 사진, 제목, 내용, 좋아요 수, 댓글 수, 거래상태 표시, 카테고리, 작성시간 * 리스트를 아래로 드래그 시 새로 고침 기능 * 핫 게시물의 기준은 좋아요 수 10개 이상 * 상단 왼쪽에 있는 학교설정은 비활성화 되어있음 | | | |
| **기술적 고려사항** | | | |
| - 사용자가 선택한 알맞은 카테고리의 게시물을 띄울 수 있는 색인 설정 필요  - 새로 고침을 활용하여 한 번에 모든 게시물을 불러오는 것이 아닌 조금씩 호출 | | | |

1. SS\_SC\_001-01

|  |  |  |  |
| --- | --- | --- | --- |
| **화면ID** | SS\_SC\_001-01 | **화면 명** | 검색창 화면 (마켓) |
| **화면개요** | * 마켓 리스트 화면에서 검색창 버튼을 눌러서 넘어온 화면이다. * 마켓의 제목을 검색할 수 있다. | | |
|  | | | |
| **처리 내용** | | | |
| * ‘뒤로가기’ 버튼 : 클릭 시 이전화면으로 이동 * 검색어를 입력하여 키패드의 확인 버튼이나 돋보기 모양 검색버튼을 눌러 검색 가능 * 음식교환 / 물건교환 / 대여하기 / 퀘스트의 카테고리를 선택, 해당 카테고리의 게시물 리스트로 이동 가능 * 핫 게시판의 게시물을 3개까지 리스트로 미리보기 가능 | | | |
| **기술적 고려사항** | | | |
| - 각 카테고리의 키 값을 넘겨주어 한 화면을 교체하여 구현  - 핫 게시물의 최근 게시물 3개를 호출 | | | |

1. SS\_SC\_001-01-1

|  |  |  |  |
| --- | --- | --- | --- |
| **화면ID** | SS\_SC\_001-01-1 | **화면 명** | 검색 결과 화면 |
| **화면개요** | * 검색 화면에서 검색 버튼을 눌러서 검색할 수 있다. * 제목에 해당 검색어가 포함되어 있는 검색 결과 리스트의 화면으로 이동할 수 있다. | | |
|  | | | |
| **처리 내용** | | | |
| * 검색창 화면에서 검색어를 함께 넘겨받아, 검색어가 제목에 포함되어 있는 게시물 리스트를 구현 * 마켓 리스트 화면과 같이 게시물을 클릭하면 해당 마켓 화면으로 넘어간다. | | | |
| **기술적 고려사항** | | | |
| - 검색결과를 작성시간을 기준으로 나열  - 검색 결과에 해당하는 마켓의 데이터를 잘 받아와야 한다. | | | |

1. SS\_SC\_001-01-2

|  |  |  |  |
| --- | --- | --- | --- |
| **화면ID** | SS\_SC\_001-01-2 | **화면 명** | 검색 결과 화면 (마켓) |
| **화면개요** | * 검색 화면에서 카테고리 버튼을 눌러서 넘어온 화면이다. * 해당 카테고리의 모든 게시물이 나열된다. | | |
|  | | | |
| **처리 내용** | | | |
| * 각 카테고리와 같은 게시물들이 작성시간을 기준으로 내림차순으로 나열된다. | | | |
| **기술적 고려사항** | | | |
| - 검색결과를 작성시간을 기준으로 나열  - 카테고리에 해당하는 마켓의 데이터를 잘 받아와야 한다.  - 카테고리 선택 시 같은 화면으로 넘어가지만 게시물은 해당 카테고리에 맞게 받아온다. | | | |

1. SS\_SC\_001-02

|  |  |  |  |
| --- | --- | --- | --- |
| **화면ID** | SS\_SC\_001-02 | **화면 명** | 마켓 게시물 |
| **화면개요** | * 사용자가 마겟 리스트 화면에서 리스트 중 하나의 마켓을 클릭하여 넘어온 화면이다. * 게시자 와 방문자의 화면 구성이 차이가 있다. | | |
|  | | | |
| **처리 내용** | | | |
| * 마켓 게시물의 구성 : 사진, 게시자 의 프로필 사진, 닉네임, 작성일자, 좋아요 개수, 좋아요 버튼, 내용, 메뉴, 댓글, 댓글 작성 칸, 댓글 작성 버튼 * 게시자 와 방문자의 차이 : 메뉴 버튼 (게시자 : 수정하기, 삭제하기 / 방문자: 신고하기, 채팅하기 )   댓글 입력 칸 (방문자일 경우 채팅하기 버튼 존재 )  댓글의 메뉴 버튼 (게시자 : 삭제하기 / 방문자: 신고하기 )   * 게시물의 사진 클릭 시 큰 화면으로 볼 수 있음 * 게시자의 프로필 사진 클릭 시 게시자 의 프로필 정보 화면으로 이동 | | | |
| **기술적 고려사항** | | | |
| - 좋아요 눌렀을 때 좋아요 누른 사용자들의 숫자가 10명 이상일 경우 핫 게시물로 지정  - 게시물의 삭제 시 댓글도 함께 삭제 구현  - 댓글은 작성시간을 기준으로 내림차순으로 나열  - 댓글 작성 시 게시자 에게 알림 발신 (보완 필요 | | | |

1. SS\_SC\_001-02-1

|  |  |  |  |
| --- | --- | --- | --- |
| **화면ID** | SS\_SC\_001-02-1 | **화면 명** | 호스트 정보 화면 |
| **화면개요** | * 사용자가 마겟 리스트 화면에서 리스트 중 하나의 마켓을 클릭하여 넘어온 화면이다. * 게시자 와 방문자의 화면 구성이 차이가 있다. | | |
|  | | | |
| **처리 내용** | | | |
| * 마켓 게시물의 구성 : 사진, 게시자 의 프로필 사진, ID, 작성일자, 좋아요 개수, 좋아요 버튼, 내용, 메뉴, 댓글, 댓글 작성 칸, 댓글 작성 버튼 * 게시자와 방문자의 차이 : 메뉴 버튼 (게시자 : 수정하기, 삭제하기 / 방문자: 신고하기, 채팅하기 )   댓글 입력 칸 (방문자일 경우 채팅하기 버튼 존재 )  댓글의 메뉴 버튼 (게시자 : 삭제하기 / 방문자: 신고하기 )   * 게시물의 사진 클릭 시 큰 화면으로 볼 수 있음 * 게시자의 프로필 사진 클릭 시 게시자의 프로필 정보 화면으로 이동 | | | |
| **기술적 고려사항** | | | |
| - 좋아요 눌렀을 때 좋아요 누른 사용자들의 숫자가 10명 이상일 경우 핫 게시물로 지정  - 게시물의 삭제 시 댓글도 함께 삭제 구현  - 댓글은 작성시간을 기준으로 내림차순으로 나열 | | | |

1. SS\_SC\_002

|  |  |  |  |
| --- | --- | --- | --- |
| **화면ID** | SS\_SC\_002 | **화면 명** | 커뮤니티 리스트 화면 |
| **화면개요** | * 사용자가 앱을 실행하면 가장 먼저 보게 되는 화면 * 커뮤니티의 게시물들이 나열되어 있는 리스트 | | |
|  | | | |
| **처리 내용** | | | |
| * 검색창 버튼 클릭 시 검색창 화면으로 이동 * 게시물 클릭 시 커뮤니티 게시물 화면으로 이동 * 핫 게시물 카테고리 버튼 클릭 시 핫 게시물 리스트로 중단에 위치한 리스트 변경 * 게시물의 정보 : 사진, 제목, 내용, 좋아요 수, 댓글 수, 작성시간 * 리스트를 아래로 드래그 시 새로 고침 기능 * 핫 게시물의 기준은 좋아요 수 10개 이상 * 상단 왼쪽에 있는 학교설정은 비활성화 되어있음 | | | |
| **기술적 고려사항** | | | |
| - 새로 고침을 활용하여 한 번에 모든 게시물을 불러오는 것이 아닌 조금씩 호출 | | | |

(10) SS\_SC\_002-01

|  |  |  |  |
| --- | --- | --- | --- |
| **화면ID** | SS\_SC\_002-01 | **화면 명** | 검색창 화면 (커뮤니티) |
| **화면개요** | * 커뮤니티 리스트 화면에서 검색창 버튼을 눌러서 넘어온 화면이다. * 커뮤니티의 제목을 검색할 수 있다. | | |
|  | | | |
| **처리 내용** | | | |
| * ‘뒤로가기’ 버튼 : 클릭 시 이전화면으로 이동 * 검색어를 입력하여 키패드의 확인 버튼이나 돋보기 모양 검색버튼을 눌러 검색 가능 * 핫 게시판의 게시물을 4개까지 리스트로 미리보기 가능 | | | |
| **기술적 고려사항** | | | |
| - 핫 게시물의 최근 게시물 4개를 호출 | | | |

1. SS\_SC\_002-01-1

|  |  |  |  |
| --- | --- | --- | --- |
| **화면ID** | SS\_SC\_002-01-1 | **화면 명** | 검색 결과 화면 (커뮤니티) |
| **화면개요** | * 검색 화면에서 검색 버튼을 눌러서 검색할 수 있다. * 제목에 해당 검색어가 포함되어 있는 검색 결과 리스트의 화면으로 이동할 수 있다. | | |
|  | | | |
| **처리 내용** | | | |
| * 검색창 화면에서 검색어를 함께 넘겨받아, 검색어가 제목에 포함되어 있는 게시물 리스트를 구현 * 커뮤니티 리스트 화면과 같이 게시물을 클릭하면 해당 커뮤니티 화면으로 넘어간다. | | | |
| **기술적 고려사항** | | | |
| - 검색결과를 작성시간을 기준으로 나열  - 검색 결과에 해당하는 커뮤니티의 데이터를 잘 받아와야 한다. | | | |

(12) SS\_SC\_002-02

|  |  |  |  |
| --- | --- | --- | --- |
| **화면ID** | SS\_SC\_002-02 | **화면 명** | 커뮤니티 게시물 |
| **화면개요** | * 사용자가 커뮤니티 리스트 화면에서 리스트 중 하나의 커뮤니티를 클릭하여 넘어온 화면이다. | | |
|  | | | |
| **처리 내용** | | | |
| * 커뮤니티 게시물의 구성 : 사진, 게시자의 프로필 사진, 닉네임, 작성일자, 좋아요 개수, 좋아요 버튼, 내용, 메뉴, 댓글, 댓글 작성 칸, 댓글 작성 버튼 * 게시자와 방문자의 차이 : 메뉴 버튼 (게시자 : 수정하기, 삭제하기 / 방문자 : 신고하기)   댓글의 메뉴 버튼 (게시자 : 삭제하기 / 방문자 : 신고하기 )   * 게시물의 사진 클릭 시 큰 화면으로 볼 수 있음 * 게시자의 프로필 사진 클릭 시 게시자의 프로필 정보 화면으로 이동 | | | |
| **기술적 고려사항** | | | |
| - 좋아요 눌렀을 때 좋아요 누른 사용자들의 숫자가 10명 이상일 경우 핫 게시물로 지정  - 게시물의 삭제 시 댓글도 함께 삭제 구현  - 댓글은 작성시간을 기준으로 내림차순으로 나열  - 댓글 작성시 게시자에게 알림 발신 (보완 필요 | | | |

(13) SS\_SC\_003

|  |  |  |  |
| --- | --- | --- | --- |
| **화면ID** | SS\_SC\_003 | **화면 명** | 게시물 작성 화면 |
| **화면개요** | * 게시물을 작성할 때 화면 * 작성하려는 게시물이 마켓인지 커뮤니티인지 선택한다. | | |
|  | | | |
| **처리 내용** | | | |
| * 마켓을 클릭하면 마켓 작성화면으로 커뮤니티를 선택하면 커뮤니티 작성 화면으로 이동 | | | |
| **기술적 고려사항** | | | |
| - | | | |

(14) SS\_SC\_003-01

|  |  |  |  |
| --- | --- | --- | --- |
| **화면ID** | SS\_SC\_003-01 | **화면 명** | 마켓 작성 화면 |
| **화면개요** | * 마켓을 작성할 때 화면 * 게시물 작성 화면에서 마켓을 선택하여 넘어온 화면이다. | | |
|  | | | |
| **처리 내용** | | | |
| * 뒤로가기 버튼을 클릭 시 게시물 작성 화면으로 이동 * 카메라 모양 버튼 클릭 시 앨범으로 접근 (아래의 숫자는 선택한 이미지 수) * 내용 입력 : 카테고리 선택, 제목 작성, 내용 작성 * 등록 버튼 누를 시 등록과 동시에 게시물 작성 화면으로 이동 | | | |
| **기술적 고려사항** | | | |
| - 앨범 접근 권한을 요구해야 한다.  - 카테고리와 제목은 선택 및 작성해야한다.  - 등록 버튼 클릭 시 다른 위셋들의 클릭을 막는다.  - 등록 시 다이얼로그를 실행시켰다가 등록이 완료되면 다이얼로그를 사라지게 한다. | | | |

(15) SS\_SC\_003-02

|  |  |  |  |
| --- | --- | --- | --- |
| **화면ID** | SS\_SC\_003-02 | **화면 명** | 커뮤니티 작성 화면 |
| **화면개요** | * 커뮤니티를 작성할 때 화면 * 게시물 작성 화면에서 커뮤니티를 선택하여 넘어온 화면이다. | | |
|  | | | |
| **처리 내용** | | | |
| * 뒤로가기 버튼을 클릭 시 게시물 작성 화면으로 이동 * 카메라 모양 버튼 클릭 시 앨범으로 접근 (아래의 숫자는 선택한 이미지 수) * 내용 입력 : 제목 작성, 내용 작성 * 등록 버튼 누를 시 등록과 동시에 게시물 작성 화면으로 이동 | | | |
| **기술적 고려사항** | | | |
| - 앨범 접근 권한을 요구해야 한다.  - 제목은 작성해야 한다.  - 등록 버튼 클릭 시 다른 위셋들의 클릭을 막는다.  - 등록 시 다이얼로그를 실행시켰다가 등록이 완료되면 다이얼로그를 사라지게 한다. | | | |

1. SS\_SC\_004

|  |  |  |  |
| --- | --- | --- | --- |
| **화면ID** | SS\_SC\_004 | **화면 명** | 채팅방 목록 화면 |
| **화면개요** | * 채팅방 목록이 있는 화면 | | |
|  | | | |
| **처리 내용** | | | |
| * 채팅방 클릭 시 채팅방 화면으로 이동 * 채팅방 목록의 정보 : 상대방 프로필 사진, 닉네임, 작성일자, 최근채팅, 읽지 않은 채팅 개수 * 오른쪽에서 왼쪽 슬라이드 시 채팅방 삭제 가능 | | | |
| **기술적 고려사항** | | | |
| - 선택한 채팅방 데이터를 잘 받아와야 한다. (안 읽은 채팅 개수, 업데이트 된 최근 채팅)  - 같은 상대라도 다른 마켓 게시물을 통해 채팅을 할 시에는 새로운 방을 만든다.  - 같은 상대와 같은 게시물을 통해 채팅을 한다면 기존 채팅방을 이용한다. | | | |

1. SS\_SC\_004-01

|  |  |  |  |
| --- | --- | --- | --- |
| **화면ID** | SS\_SC\_004-01 | **화면 명** | 채팅방 화면 |
| **화면개요** | * 채팅방이 있는 화면 * 마켓 게시물의 게시자 / 방문자의 화면이 다르다. | | |
|  | | | |
| **처리 내용** | | | |
| * 채팅방 목록 화면에서 채팅방 클릭 시 넘어온 화면 * 뒤로가기 버튼 클릭 시 채팅방 목록 화면으로 이동 * 채팅방의 정보 : 상대방의 닉네임, 프로필 사진, 닉네임, 채팅, 채팅 일자, 게시물의 사진, 제목, 내용   (게시자 : 진행중, 거래 완료 버튼 / 방문자 : - ), 채팅방 메뉴(삭제)   * 오른쪽에서 왼쪽 슬라이드 시 채팅방 삭제 가능 | | | |
| **기술적 고려사항** | | | |
| - 앨범 접근이 가능해야 하고, 이미지가 전송 중이라면 화면의 클릭을 다이얼로그로 막는다  - 진행중 클릭 후 거래완료까지 클릭이 되면 리뷰창을 띄운다.  - 후기를 입력 받게 하며, 입력한 후기는 리뷰창에 띄울 수 있어야한다. (후기에서 불쾌함 선택 시 후기 입력창 자동 실행)  - 마켓 게시물의 내용을 잘 받아올 수 있어야한다.  - 새로 이루어지는 채팅과 나갔다 들어왔을 때의 이전의 채팅이 순서대로 띄워져야 한다.  - 리뷰 작성 완료 시 상대방이 앱을 시작했을 때 나에 대해 쓸 수 있는 리뷰창을 띄운다.  - 채팅 전송 시, 리뷰 작성 시 상대방에게 알림 발신 (보완 필요) | | | |

1. SS\_SC\_005

|  |  |  |  |
| --- | --- | --- | --- |
| **화면ID** | SS\_SC\_005 | **화면 명** | 내정보 화면 |
| **화면개요** | * 내 정보가 있는 화면 | | |
|  | | | |
| **처리 내용** | | | |
| * 자신의 정보가 있는 화면 : 프로필 사진, 닉네임, 학교, 진행중/거래완료/내가 쓴 게시물, 리뷰, 후기 * 회원정보 변경 버튼을 누르면 회원정보 변경 창으로 넘어간다. * 로그아웃 버튼이 있다. * 공지사항, 자주 묻는 질문, 앱 설정이 있다. (미사용) | | | |
| **기술적 고려사항** | | | |
| - 진행중, 거래완료, 내게 작성된 리뷰가 같은 화면으로 넘어가지만, 게시물 정보를 알맞게 받아와야 한다.  - 내 정보에 알맞은 정보를 받아와야 한다. : 프로필 사진, 닉네임, 학교, 리뷰  - 버튼들을 한 화면에 다 위치시킬 수 없을 때는 스크롤이 가능하게 한다. | | | |

1. SS\_SC\_005-01

|  |  |  |  |
| --- | --- | --- | --- |
| **화면ID** | SS\_SC\_005-01 | **화면 명** | 회원정보 수정 화면 |
| **화면개요** | * 내 정보를 수정할 수 있는 화면 | | |
|  | | | |
| **처리 내용** | | | |
| * 자신의 정보가 미리 기입되어 있는 화면 : 프로필 사진, 닉네임, 생년월일, 학교, 계정정보 * 프로필 사진 변경 시에는 앨범으로 들어간다. * 수정 버튼 클릭 시 내 정보 화면으로 이동, 변경 사항 반영 | | | |
| **기술적 고려사항** | | | |
| - 처음 화면에 들어 갔을 때에 정보를 미리 셋팅 해 놓는다.  - 수정 버튼 클릭 시 변경한 정보와 변경되지 않은 정보를 함께 새로 저장한다.  (닉네임만 변경 시 변경하지 않은 프로필과 학교 등의 데이터가 포기화 되지 않게 함께 저장) | | | |

1. SS\_SC\_005-02

|  |  |  |  |
| --- | --- | --- | --- |
| **화면ID** | SS\_SC\_005-02 | **화면 명** | 진행중 게시물 리스트 화면 |
| **화면개요** | * 진행 중인 마켓 게시물 리스트 화면 | | |
|  | | | |
| **처리 내용** | | | |
| * 마켓 리스트 화면과 같은 방법으로 구성된다. : 내가 관여하고 있는 진행중 상태의 마켓 게시물 리스트 * 게시물을 클릭 시 해당 마켓 게시물 화면으로 이동 | | | |
| **기술적 고려사항** | | | |
| - 마켓 게시물의 거래 상태에 대한 정보를 잘 받아와야 하고, 내림차순으로 리스트로 출력해 낸다.  - 진행중, 거래완료, 내게 작성된 리뷰가 같은 화면으로 넘어가지만, 게시물 정보를 알맞게 받아와야 한다. | | | |

1. SS\_SC\_005-03

|  |  |  |  |
| --- | --- | --- | --- |
| **화면ID** | SS\_SC\_005-03 | **화면 명** | 거래완료 게시물 리스트 화면 |
| **화면개요** | * 거래완료 된 마켓 게시물 리스트 화면 | | |
|  | | | |
| **처리 내용** | | | |
| * 마켓 리스트 화면과 같은 방법으로 구성된다. : 내가 관여하고 있는 거래완료 된 상태의 마켓 게시물 리스트 * 게시물을 클릭 시 해당 마켓 게시물 화면으로 이동 | | | |
| **기술적 고려사항** | | | |
| - 마켓 게시물의 거래 상태에 대한 정보를 잘 받아와야 하고, 내림차순으로 리스트로 출력해 낸다.  - 진행중, 거래완료, 내게 작성된 리뷰가 같은 화면으로 넘어가지만, 게시물 정보를 알맞게 받아와야 한다. | | | |

1. SS\_SC\_005-04

|  |  |  |  |
| --- | --- | --- | --- |
| **화면ID** | SS\_SC\_005-04 | **화면 명** | 내가 작성한 게시물 리스트 화면 |
| **화면개요** | * 내가 작성한 게시물 리스트 화면 | | |
|  | | | |
| **처리 내용** | | | |
| * 내가 작성한 게시물의 리스트를 보여주는 화면이다. (왼쪽 상단 클릭 시 : 마켓 / 오른쪽 상단 클릭 시 : 커뮤니티) * 게시물을 클릭 시 해당 게시물 화면으로 이동 | | | |
| **기술적 고려사항** | | | |
| - 게시물에 대한 정보를 잘 받아와야 하고, 내림차순으로 리스트로 출력해 낸다. | | | |

1. SS\_SC\_005-05

|  |  |  |  |
| --- | --- | --- | --- |
| **화면ID** | SS\_SC\_005-05 | **화면 명** | 나에게 작성된 리뷰 화면 |
| **화면개요** | * 나에게 작성된 리뷰 리스트 화면 | | |
|  | | | |
| **처리 내용** | | | |
| * 마켓 리스트 화면과 같은 방법으로 구성된다. : 나에게 작성된 리뷰의 리스트 * 게시물을 클릭 시 해당 마켓 게시물 화면으로 이동 * 마켓 리스트 화면과 같지만, 리스트에서 요구하는 정보가 다르다 : 리뷰, 닉네임, 프로필, 학교, 리뷰 작성일자, 후기 | | | |
| **기술적 고려사항** | | | |
| - 알맞은 리뷰의 정보를 가져와야 한다.  - 리뷰 클릭 시 해당 마켓 게시물 화면으로 잘 이동될 수 있게 해야 한다. | | | |

1. SS\_SC\_005-06

|  |  |  |  |
| --- | --- | --- | --- |
| **화면ID** | SS\_SC\_005-06 | **화면 명** | 로그아웃 화면 |
| **화면개요** | * 로그아웃의 다이얼로그 화면 | | |
|  | | | |
| **처리 내용** | | | |
| * 로그아웃을 누르면 어플리케이션의 로그인 화면으로 넘어가게 된다. | | | |
| **기술적 고려사항** | | | |
| - 로그인 정보가 잘 로그아웃 되어야한다.  - 로그아웃 이전의 화면들이 뒤로가기를 눌러도 없어져야 한다. | | | |

**2.4.1. 시스템 아키텍처 설계서**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | | | | |
| 그림 | 8. | 전체 | 시스템 | 구조도 |

ㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡ

* + 1. **시스템 개요**

PressCheck은 온라인 뉴스를 수집하여 분석하는 웹 서비스이다. PressCheck의 주요 기능은 각 기사 의 키워드, 본문 요약, 이 기사와 유사한 내용의 기사를 분석하여 제공하는 것이다. 7개의 언론사에 서 제공하는 RSS를 통해 7개의 카테고리에 해당하는 온라인 기사를 feedparser, beautifulsoup 라이브 러리를 이용하여 수집한 뒤, 본문에서 추출한다. 수집된 기사 정보는 데이터베이스에 저장되고, 수집 후에 wordrank, cosine유사도를 사용해 각 기사에 대해 키워드를 추출한다. 키워드 추출이 끝나면 fasttext를 사용하여 본문을 요약한 뒤, 기사 정보를 갱신한다. 수집된 기사의 본문은 kobert-sentence-tokenizer를 이용해 문장단위로 분리되어 다른 비교대상 기사들의 문장들과 코사인 유사도 비교를 통해 기준 기사 - 타깃 기사 - 유사도 쌍을 갱신한다.

그 외에, PressCheck은 Todays페이지에서 카테고리 별로 하루 동안 가장 화제가 되었던 키워드, 시 간별 키워드 추이, 오늘의 기사 요약 기능을 제공한다. Homepage의 많이 본 기사는 '다음'에서 제공 하는 랭킹 뉴스의 제목과 데이터베이스에 수집된 뉴스들의 제목을 벡터화한 뒤, 코사인 유사도를 통 해 유사도가 높은 후보들을 추려낸 뒤, 다시 순위를 부여하여 홈페이지에 게재된다.

### 응용 목표 시스템 구성도

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | | | | |
| 그림 | 8. | 서비스 | 측면에서의 | 구성도 |

ㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡ

* + 1. **시험 시나리오 및 절차**

- VV데모 시나리오 기반 내용

|  |  |
| --- | --- |
| **가정** | * 앨범에 대한 접근이 정상적으로 작동하는 상태를 전제로 한다. * 파이어베이스에 대한 어플리케이션의 연결이 잘 작동되는 것을 전제로 한다. |
| **전략** | 1. 파이어베이스의 정보들을 잘 수집하여 출력하는지 확인한다. 2. 파이어베이스에 입력한 데이터가 잘 저장되었는지 확인한다. 3. 어플리케이션 내의 삭제, 수정 기능들이 잘 반영되었는지 확인한다. |

## 단위 시험 설계 및 시나리오

|  |  |
| --- | --- |
| **입출력 시험** | [PC-CT-010] 마켓 창, 커뮤니티 창, 검색결과 창, 검색 창 카테고리 게시물 창에서 어댑터를 통해 게시물이 잘 출력되었는지 확인한다.  [PC-CT-020] 게시물 리스트에서 게시물 창으로 이동하면 해당 게시물의 자세한 정보를 잘 받아온다.  [PC-CT-030] 게시물 창에서 댓글의 리스트가 잘 출력되었는지 확인한다.  [PC-CT-040] 글쓰기, 회원정보 입력, 채팅 창, 회원정보 변경 창에서 선택한 이미지가 뷰에 잘 출력되었는지 확인한다.  [PC-CT-050] 채팅 창의 이전의 채팅 기록과, 마켓의 정보가 잘 출력되었는지 확인한다.  [PC-CT-060] 개인 정보 창에서 본인 혹은 타인의 회원정보가 잘 출력되었는지 확인한다.  [PC-CT-070] 리뷰창에서 마켓의 정보를 잘 출력했는지 확인한다.  [PC-CT-080] 작성된 리뷰 창에서 유저에게 작성된 리뷰와 후기의 정보를 잘 출력하였는지 확인한다. |
| **서버**  **연동**  **시험** | [PC-CT-090] 회원정보 입력 창에서 입력한 사용자의 정보가 파이어베이스에 잘 저장되었는지 확인한다.  [PC-CT-100] 글쓰기 창에서 입력한 게시물의 정보가 파이어베이스에 잘 저장되었는지 확인한다.  [PC-CT-110] 게시물 수정에서 유저가 변경한 게시물의 정보가 잘 저장되었는지 확인한다.  [PC-CT-120] 게시물 창에서 작성한 댓글의 정보가 파이어베이스에 잘 저장되었는지 확인한다.  [PC-CT-130] 채팅 창에서 주고받은 유저 간의 채팅 정보가 파이어베이스에 잘 저장되었는지 확인한다.  [PC-CT-140] 회원정보 변경 창에서 변경한 회원의 정보가 잘 수정되었는지 확인한다.  [PC-CT-150] 리뷰 작성 시 입력된 정보가 잘 저장되었는지 확인한다.  [PC-CT-160] 게시물 삭제, 댓글 삭제, 채팅방 삭제가 잘 작동되는지 확인한다.  [PC-CT-170] 게시물 삭제 시 게시물의 댓글들이 함께 잘 삭제되는지 확인한다.  [PC-CT-180] 채팅방 삭제 시 채팅방의 메시지들이 함께 잘 삭제되었는지 확인한다. |

## 데이터베이스 설계서

### 데이터베이스 정의

|  |  |  |  |
| --- | --- | --- | --- |
| DB ID | DB 명 | 컬렉션 ID | 컬렉션 명 |
| CP\_PR\_011 | Hommade Guardian Beta | CP\_PRC\_01 | COMMUNITY |
| CP\_PR\_011 | Hommade Guardian Beta | CP\_PRC\_01-1 | COMMENT |
| CP\_PR\_011 | Hommade Guardian Beta | CP\_PRC\_02 | MARKETS |
| CP\_PR\_011 | Hommade Guardian Beta | CP\_PRC\_02-1 | COMMENT |
| CP\_PR\_011 | Hommade Guardian Beta | CP\_PRC\_03 | ROOMS |
| CP\_PR\_011 | Hommade Guardian Beta | CP\_PRC\_03-1 | MESSAGE |
| CP\_PR\_011 | Hommade Guardian Beta | CP\_PRC\_04 | USERS |
| CP\_PR\_011 | Hommade Guardian Beta | CP\_PRC\_04-1 | REVIEW |

* + 1. **데이터베이스 명세**

#### (1) COMMUNITY

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 컬렉션명 | COMMUNITY | | | 컬렉션 ID | CP\_PRC\_01 |
| 컬렉션 설명 | 커뮤니티 게시물의 데이터를 저장한다. | | | | |
| Key 명 | Key ID | 타입 및 길이 | Not Null | 기본 값 | 제약조건 |
| 댓글 개수 | CommunityModel\_CommentCount | Int | Y | O | COMMENT의 size를 받아옴 |
| 게시물 Uid | CommunityModel\_Community\_Uid | String | Y | X |  |
| 작성일자 | CommunityModel\_DateOfManufacture | Date | Y | X |  |
| 게시자 Uid | CommunityModel\_Host\_Uid | String | Y | X |  |
| 핫게시물 여부 | CommunityModel\_HotCommunity | String | Y | X | 핫게시물 : O / 핫게시물X :X |
| 이미지 리스트 | CommunityModel\_ImageList | ArrayList<String> | N | O | 처음 Null |
| 좋아요 리스트 | CommunityModel\_LikeList | ArrayList<String> | N | O | 처음 Null |
| 게시물 내용 | CommunityModel\_Text | String | Y | X |  |
| 게시물 제목 | CommunityModel\_Title | String | Y | X |  |

#### (2) COMMENT

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 컬렉션명 | COMMENT | | | 컬렉션 ID | CP\_PRC\_01-1 |
| 컬렉션 설명 | 커뮤니티 게시물의 댓글 데이터를 저장한다. | | | | |
| Key 명 | Key ID | 타입 및 길이 | Not Null | 기본 값 | 제약조건 |
| 댓글 내용 | Community\_CommentModel\_Comment | String | Y | X |  |
| 댓글 Uid | Community\_CommentModel\_Comment\_Uid | String | Y | X |  |
| 작성일자 | Community\_CommentModel\_DateOfManufacture | Date | Y | X |  |
| 게시자 이미지 | Community\_CommentModel\_Host\_Image | String | N | X |  |
| 게시자 닉네임 | Community\_CommentModel\_Host\_Name | String | Y | X |  |
| 게시자 Uid | Community\_CommentModel\_Host\_Uid | String | Y | X |  |
| 마켓 Uid | Community\_CommentModel\_Market\_Uid | String | Y | X |  |

1. MARKETS

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 컬렉션명 | MARKETS | | | 컬렉션 ID | CP\_PRC\_02 |
| 컬렉션 설명 | 마켓 게시물의 데이터를 저장한다. | | | | |
| Key 명 | Key ID | 타입 및 길이 | Not Null | 기본 값 | 제약조건 |
| 카테고리 | MarketModel\_Category | String | Y | X |  |
| 댓글 개수 | MarketModel\_CommentCount | Int | Y | O | COMMENT의 size를 받아옴 |
| 작성일자 | MarketModel\_DateOfManufacture | Date | Y | X |  |
| 게시자 Uid | MarketModel\_Host\_Uid | String | Y | X |  |
| 핫게시물 여부 | MarketModel\_HotMarket | String | Y | X | 핫게시물 : O / 핫게시물X :X |
| 이미지 리스트 | MarketModel\_ImageList | ArrayList<String> | N | O | 처음 Null |
| 좋아요 리스트 | MarketModel\_LikeList | ArrayList<String> | N | O | 처음 Null |
| 게시물 Uid | MarketModel\_Market\_Uid | String | Y | X |  |
| 게시물 내용 | MarketModel\_Text | String | Y | X |  |
| 게시물 제목 | MarketModel\_Title | String | Y | X |  |
| 거래완료 상대 Uid | MarketModel\_deal | String | Y | O | 채팅에서 거래완료 누른 상대방 Uid |
| 진행중 산대 Uid | MarketModel\_reservation | String | Y | O | 채팅에서 진행중 누른 상대방 Uid |

#### COMMENT

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 컬렉션명 | COMMENT | | | 컬렉션 ID | CP\_PRC\_02-1 |
| 컬렉션 설명 | 마켓 게시물의 댓글 데이터를 저장한다. | | | | |
| Key 명 | Key ID | 타입 및 길이 | Not Null | 기본 값 | 제약조건 |
| 댓글 내용 | Market\_CommentModel\_Comment | String | Y | X |  |
| 댓글 Uid | Market\_CommentModel\_Comment\_Uid | String | Y | X |  |
| 작성일자 | Market\_CommentModel\_DateOfManufacture | Date | Y | X |  |
| 게시자 이미지 | Market\_CommentModel\_Host\_Image | String | N | X |  |
| 게시자 닉네임 | Market\_CommentModel\_Host\_Name | String | Y | X |  |
| 게시자 Uid | Market\_CommentModel\_Host\_Uid | String | Y | X |  |
| 마켓 Uid | Market\_CommentModel\_Market\_Uid | String | Y | X |  |

1. ROOMS

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 컬렉션명 | ROOMS | | | 컬렉션 ID | CP\_PRC\_03 |
| 컬렉션 설명 | 채팅방의 데이터를 저장한다. | | | | |
| Key 명 | Key ID | 타입 및 길이 | Not Null | 기본 값 | 제약조건 |
| 채팅 시간 | RoomModel\_DateOfManufacture | Date | Y | X |  |
| 파일 이름 | RoomModel\_FileName | String | N | X | 없으면 null |
| 파일 크기 | RoomModel\_FileSize | String | N | X | 없으면 null |
| 방문자 Uid | RoomModel\_GusetUser | String | Y | X | 방문자의 Uid |
| 채팅방의 이미지 개수 | RoomModel\_ImageCount | String | Y | X |  |
| 메시지 내용 | RoomModel\_Message | String | Y | X |  |
| 메시지 타입 | RoomModel\_MessageType | String | Y | X |  |
| 게시물의 Uid | RoomModel\_PostUid | String | Y | X |  |
| 채팅방 제목 | RoomModel\_Title | String | Y | X |  |
| 채팅 확인 여부 | RoomModel\_USERS | Map<String, Integer> | Y | O | 채팅 확인 여부  ( 읽은 것 : 0 /안 읽은 것 1) |
| 채팅방 이탈 여부 | RoomModel\_USER\_OUT | Map<String, Integer> | Y | O | 채팅방 이탈 여부  ( 나간 것 : 0 /있는 것 1) |
| 게시자 Uid | RoomModel\_UserUid | String | Y | X | 게시자의 Uid |

#### MESSAGE

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 컬렉션명 | MESSAGE | | | 컬렉션 ID | CP\_PRC\_03-1 |
| 컬렉션 설명 | 채팅 데이터를 저장한다. | | | | |
| Key 명 | Key ID | 타입 및 길이 | Not Null | 기본 값 | 제약조건 |
| 채팅 시간 | MessageModel\_DateOfManufacture | Date | Y | X |  |
| 파일 이름 | MessageModel\_FileName | String | N | X | 없으면 null |
| 파일 크기 | MessageModel\_FileSize | String | N | X | 없으면 null |
| 메시지 내용 | MessageModel\_Message | String | Y | X |  |
| 메시지 타입 | MessageModel\_MessageType | String | Y | X |  |
| 채팅 확인 여부 | MessageModel\_ReadUser | Map<String, Integer> | Y | O | 0인 사람 : 보낸 사람  1인 사람 : 받는 사람 |
| 게시자 Uid | MessageModel\_UserUid | String | Y | X |  |

1. USERS

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 컬렉션명 | USERS | | | 컬렉션 ID | CP\_PRC\_04 |
| 컬렉션 설명 | 유저의 데이터를 저장한다. | | | | |
| Key 명 | Key ID | 타입 및 길이 | Not Null | 기본 값 | 제약조건 |
| 유저 생일 | UserModel\_BirtDay | String | Y | O | 기본값은 2021/01/01 |
| 생성 일시 | UserModel\_DateOfManufacture | Date | Y | X |  |
| 유저 ID | UserModel\_ID | String | Y | X |  |
| 거래된 마켓 리스트 | UserModel\_Market\_dealList | ArrayList<String> | N | X | 없으면 null |
| 예약된 마켓 리스트 | UserModel\_Market\_reservationList | ArrayList<String> | N | X | 없으면 null |
| 유저 닉네임 | UserModel\_NickName | String | Y | O |  |
| 유저 프로필 사진 | UserModel\_ProfileImage | String | N | X | 없으면 null |
| 유저 토큰 | UserModel\_Token | String | Y | X | 알림을 위한 토큰 값 |
| 유저 Uid | UserModel\_Uid | String | Y | X |  |
| 유저 학교 | UserModel\_University | int | Y | O | 기본값은 홍익대학교 |
| 유저가 작성하지 않은 리뷰 리스트 | UserModel\_Unreview | ArrayList<HashMap<String, String>> | N | X | 리뷰를 작성하지 않은 게시물 Uid, 상대방의 Uid |
| 유저에게 작성된 후기 | UserModel\_WritenReviewList | ArrayList<String> | N | X | 게시자의 Uid |
| 불쾌함 리뷰 리스트 | UserModel\_badReviewList | ArrayList<String> | N | X | 없으면 null |
| 완벽함 리뷰 리스트 | UserModel\_completeReviewList | ArrayList<String> | N | X | 없으면 null |
| 정확함 리뷰 리스트 | UserModel\_correctReviewList | ArrayList<String> | N | X | 없으면 null |
| 친절함 리뷰 리스트 | UserModel\_kindReviewList | ArrayList<String> | N | X | 없으면 null |

#### REVIEW

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 컬렉션명 | REVIEW | | | 컬렉션 ID | CP\_PRC\_04-1 |
| 컬렉션 설명 | 리뷰 데이터를 저장한다. | | | | |
| Key 명 | Key ID | 타입 및 길이 | Not Null | 기본 값 | 제약조건 |
| 작성 시간 | ReviewModel\_DateOfManufacture | Date | Y | X |  |
| 게시물 Uid | ReviewModel\_PostUid | String | Y | X |  |
| 후기 내용 | ReviewModel\_Review | String | N | X | 없으면 null |
| 선택된 리뷰 | ReviewModel\_Selected\_Review | int | Y | X | 0 : 친절함 / 1 : 정확함  2 : 완벽함 / 3 : 불쾌함 |
| 상대방 닉네임 | ReviewModel\_To\_User\_NickName | String | Y | X |  |
| 상대방 프로필 이미지 | ReviewModel\_To\_User\_ProfileImage | String | N | X | 없으면 null |
| 상대방 Uid | ReviewModel\_To\_User\_Uid | String | Y | X |  |
| 리뷰 Uid | ReviewModel\_Uid | String | Y | X |  |

## 통합시험 시나리오

ㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡ

* + 1. **시험 시나리오**

1. PC\_IT\_IS\_010

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 시험 시나리오 ID | PC\_IT\_TS\_010 | | | |
| 시험 시나리오명 | 크롤링 데이터 삽입 | | | |
| 시험 시나리오설명 | 크롤링 데이터를 데이터베이스에 삽입한다. | | | |
| 관련 시퀀스ID | SS\_SVR\_010 | | | |
| 시험케이스 ID | 시험케이스 설명 | 시험 절차 | 시나리오 설명 | 비 고 |
| PC\_IT\_TC\_011 | 크롤링 데이터 삽입 | 1. 언론사와 로컬 간 데이터 통신 2. 로컬 엔진과 몽고DB의 데이터통신 | RSS링크를 통해 로컬에 수집된 크롤링 데이 터를 DB로 전송한다. |  |

1. PC\_IT\_IS\_020

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 시험 시나리오 ID | PC\_IT\_TS\_020 | | | |
| 시험 시나리오명 | 키워드\_본문요약 | | | |
| 시험 시나리오설명 | 삽입된 기사의 키워드를 추출하고 본문을 요약하여 데이터베이스에 갱신한다. | | | |
| 관련 시퀀스ID | SS\_SVR\_021, SS\_SVR\_022 | | | |
| 시험케이스 ID | 시험케이스 설명 | 시험 절차 | 시나리오 설명 | 비 고 |
| PC\_IT\_TC\_021 | 키워드 추출 | 1. 로컬에서 DB에 기사 정보 요청 2. DB에서 로컬엔진으로 데이터 전송 3. 로컬엔진에서 키워드 추출 4. 로컬엔진에서 DB로 키워드 업데이트 | 반복문을 돌면서 DB에 삽입된 기사의 본문 을 전송받아 키워드를 처리한 뒤 키워드 데 이터를 갱신한다. |  |
| PC\_IT\_TC\_022 | 본문 요약 | 1. 로컬에서 DB에 기사 정보 요청 2. DB에서 로컬엔진으로 데이터 전송 3. 로컬엔진에서 본문 요약 4. 로컬엔진에서 DB로 요약된 본문을 업데 이트 | 반복문을 돌면서 DB에 삽입된 기사의 본문 을 전송받아 본문을 3단계로 요약한 뒤, 본 문 요약 데이터를 갱신한다. |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 시험 시나리오 ID | PC\_IT\_TS\_030 | | | |
| 시험 시나리오명 | 유사도 추출 | | | |
| 시험 시나리오설명 | 삽입된 기사간의 유사도를 추출하여 유사도 컬렉션에 삽입한다. | | | |
| 관련 시퀀스ID | SS\_SVR\_023 | | | |
| 시험케이스 ID | 시험케이스 설명 | 시험 절차 | 시나리오 설명 | 비 고 |
| PC\_IT\_TC\_31 | 자신을 제외한 기사 간의 유사도 값 추출 | 1. 로컬에서 DB에 기사 정보 요청 2. DB에서 로컬엔진으로 데이터 전송 3. 로컬엔진에서 유사도 추출 4. 로컬엔진에서 DB로 유사도 쌍 삽입 | 반복문을 돌면서 DB에 삽입된 기사의 본문 을 전송받아 기준 기사와 타깃 기사의 유사 도 값을 추출한 뒤, 기준 기사, 타깃 기사, 유사도 쌍을 유사도 컬렉션에 삽입한다. |  |

(4) PC\_IT\_IS\_040

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 시험 시나리오 ID | PC\_IT\_TS\_040 | | | |
| 시험 시나리오명 | 많이 본 뉴스 | | | |
| 시험 시나리오설명 | 랭킹 뉴스와 데이터베이스의 뉴스를 비교한 뒤 후보 기사들을 데이터베이스에 저장한다. | | | |
| 관련 시퀀스ID | SS\_WEB\_030 | | | |
| 시험케이스 ID | 시험케이스 설명 | 시험 절차 | 시나리오 설명 | 비 고 |
| PC\_IT\_TC\_41 | 많이 본 뉴스를 기준에 따라 필터링하여 DB에 삽입 | 1. 랭킹 뉴스 크롤링 2. DB에서 로컬로 기사 데이터 전송 3. DB의 뉴스 중 랭킹 뉴스와 제목 유사도 가 높은 기사 추출 4. 순위 재할당 5. 로컬에서 DB로 많이 본 기사 전송 | 1. 로컬에서 랭킹 뉴스 주소의 페이지를 크 롤링하여 조회 수가 많은 30개의 기사 정보 를 가져온다. 2. DB에서 로컬엔진으로 모든 기사의 정보 를 가져온다. 3. 로컬엔진에서 랭킹 뉴스의 벡터화된 제목 과 DB의 뉴스 기사 제목을 비교한다. 4. 코사인 유사도가 높은 후보 기사들을 추 출한다. 5. 후보 기사들의 랭킹을 조정한다. 6. 로컬에서 DB로 후보 기사들을 전송한다. |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 시험 시나리오 ID | PC\_IT\_TS\_050 | | | |
| 시험시나리오 명 | Todays | | | |
| 시험시나리오 설명 | Todays탭에 표시될 오늘의 키워드, 키워드 추이, 오늘의 헤드라인을 데이터베이스에 저장한다. | | | |
| 관련 시퀀스ID | SS\_WEB\_050 | | | |
| 시험케이스 ID | 시험케이스 설명 | 시험 절차 | 시나리오 설명 | 비고 |
| PC\_IT\_TC\_51 | Todays탭 표시 정보 DB에 삽입 | 1. 로컬 엔진에서 DB로 키워드 데이터 요청 2. DB에서 로컬 엔진으로 키워드 데이터 발 송 3. 카테고리 별로 기사제목과 링크를 요청 받고 랜덤하게 3개 추출하여 link와 headline속성에 저장 4. 카테고리 별로 키워드를 요청받고 빈도수 상위 3개를 추출하여 keyword속성에 저장 5. 카테고리 별로 키워드를 요청받고 해당시 간 키워드 상위 5개를 빈도수와 함께 추출 하고 timeKeywords속성에 저장 | 카테고리 별로 반복문을 돌면서 DB 에 삽입된 기사의 키워드와 링크, 기 사제목을 전송받아서 랜덤함수와 Counter함수를 통해서 headline과 keyword, timeKeywords속성에 저장 한다. |  |

(6) PC\_IT\_IS\_060

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 시험 시나리오 ID | PC\_IT\_TS\_060 | | | |
| 시험시나리오 명 | 메일링 | | | |
| 시험시나리오 설명 | 메일을 입력하면 분석 리포트를 발송한다. | | | |
| 관련 시퀀스ID | SS\_WEB\_042 | | | |
| 시험케이스 ID | 시험케이스 설명 | 시험 절차 | 시나리오 설명 | 비고 |
| PC\_IT\_TC\_61 | 기입된 메일로 분석리포트 전송 | 1. 홈페이지에서 이메일 버튼을 누른 후, 이 메일 기입 2. 기입된 메일로 현재 기사의 분석 리포트 를 발송 | 홈페이지의 이메일 버튼을 누르면 사 용자의 메일을 기입할 수 있는 창이 뜬다.  사용자의 메일을 기입하면 현재 보고 있는 기사의 분석 리포트를 메일로 발송한다. |  |

1. **구현 단계**

(1) PC\_UT\_010

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 단위시험 ID | PC\_UT\_010 | | | | |
| 설명 | 프로젝트 전반적으로 입출력에 관한 시험을 한다. | | | | |
| 케이스ID | 케이스명 | 시험결과 | | | |
| 수행일 | 결과 | 결함  내용 | 결함  조치여부 |
| PC-CT-010 | 게시물 리스트 출력 |  | pass |  |  |
| PC-CT-020 | 게시물 출력 |  | pass |  |  |
| PC-CT-030 | 댓글 리스트 출력 |  | pass |  |  |
| PC-CT-040 | 선택된 이미지 출력 |  | pass |  |  |
| PC-CT-050 | 채팅 방 정보 출력 |  | pass |  |  |
| PC-CT-060 | 회원 정보 출력 |  | pass |  |  |
| PC-CT-070 | 리뷰창에서 마켓 정보 출력 |  | pass |  |  |
| PC-CT-080 | 리뷰, 후기 정보 출력 |  | pass |  |  |

(2) PC\_UT\_011

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 단위시험 ID | PC\_UT\_011 | | | | |
| 설명 | 프로젝트 전반적인 서버 연동에 대한 시험을 한다. | | | | |
| 케이스ID | 케이스명 | 시험결과 | | | |
| 수행일 | 결과 | 결함  내용 | 결함  조치여부 |
| PC-CT-090 | 회원 정보 저장 |  | pass |  |  |
| PC-CT-100 | 게시물 정보 저장 |  | pass |  |  |
| PC-CT-110 | 게시물 수정사항 저장 |  | pass |  |  |
| PC-CT-120 | 댓글 정보 저장 |  | pass |  |  |
| PC-CT-130 | 채팅 정보 저장 |  | pass |  |  |
| PC-CT-140 | 회원 정보 수정사항 저장 |  | pass |  |  |
| PC-CT-150 | 리뷰 정보 저장 |  | pass |  |  |
| PC-CT-160 | 삭제 기능 동작 |  | pass |  |  |
| PC-CT-170 | 게시물 삭제 시  댓글도 삭제 |  | pass |  |  |
| PC-CT-180 | 채팅방 삭제 시 메시지도 삭제 |  | pass |  |  |

ㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡㅡ

## 프로젝트 시험 결과서

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **프로젝트 시험 결과서** | | | | | |
| 시스템명 | PressCheck | 서브시스템명 | 온라인 기사 분석 시스템 | | |
| 단계명 | 시험 | 작성일자 | 2020/11/26 | 버전 |  |

* + 1. **프로젝트 시험**

(1) PC\_IT\_TS\_010 ~ PC\_IT\_TS\_040

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 시험시나리오 ID | | PC\_IT\_TS\_010~PC\_IT\_TS\_040 | | |
| 시험시나리오 명 | | 기사 분석리포트 확인 | | |
| 시험시나리오 설명 | | 가장 많이 본 뉴스기사의 유사한 뉴스 차이점 키워드 확인 | | |
| 시험시나리오 결과 | | 가장 많이 본 뉴스기사의 유사한 뉴스 차이점 키워드 확인할 수 있었다. | | |
| 시험 절차 | | 시험 항목 | 예상결과 | 시험 결과 |
| 순번 | 업무처리의 내용 |
| 1 | 탭 클릭 | Home 탭을 클릭 | Home 화면전환 | Home 화면전환 |
| 2 | 기사 클릭 | 가장 많이 본 뉴스 3번째를 클릭 | 기사 화면전환 | 기사 화면전환 |
| 3 | 차이점 키워드 확인 | 2번째 차이점 키워드를 확인 | 차이점 키워드 확인 | 차이점 키워드 확인 |

1. PC\_IT\_TS\_050

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 시험시나리오 ID | | PC\_IT\_TS\_050 | | |
| 시험시나리오 명 | | Todays 테스트 | | |
| 시험시나리오 설명 | | Todays화면에 문화 카테고리에서 하단의 오늘의 주요 기사 요약 부분에서 나온 기사를  클릭해서 해당 기사의 주요 키워드를 확인한다. | | |
| 시험시나리오 결과 | | 오늘의 주요 기사의 키워드를 확인할 수 있었다. | | |
| 시험 절차 | | 시험 항목 | 예상결과 | 시험 결과 |
| 순번 | 업무처리의 내용 |
| 1 | 탭 클릭 | Todays 탭을 클릭 | Todays 화면전환 | Todays 화면전환 |
| 2 | 카테고리 클릭 | 카테고리의 문화 탭을 클릭 | 카테고리 화면전환 | 카테고리 화면전환 |
| 3 | 오늘의 주요 기사 클릭 | 오늘의 주요 기사 요약에서 가운데 기사를 확인 | 기사 화면전환 | 기사 화면전환 |
| 4 | 분석리포트의 키워드를 확인 | 해당 기사의 분석리포트 키워드 확인 | 키워드 확인 | 키워드 확인 |

1. PC\_IT\_TS\_060

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 시험시나리오 ID | | PC\_IT\_TS\_060 | | |
| 시험시나리오 명 | | 분석 리포트 메일링 테스트 | | |
| 시험시나리오 설명 | | Todays화면에 문화 카테고리에서 하단의 오늘의 주요 기사 요약 부분에서 나온 기사를  클릭해서 해당 기사의 주요 키워드를 확인한다. | | |
| 시험시나리오 결과 | | 오늘의 주요 기사의 키워드를 확인할 수 있었다. | | |
| 시험 절차 | | 시험 항목 | 예상결과 | 시험 결과 |
| 순번 | 업무처리의 내용 |
| 1 | 탭 클릭 | Todays 탭을 클릭 | Todays 화면전환 | Todays 화면전환 |
| 2 | 카테고리 클릭 | 카테고리의 문화 탭을 클릭 | 카테고리 화면전환 | 카테고리 화면전환 |
| 3 | 오늘의 주요 기사 클릭 | 오늘의 주요 기사 요약에서 가운데 기사를 확인 | 기사 화면전환 | 기사 화면전환 |
| 4 | 분석리포트의 키워드를 확인 | 해당 기사의 분석리포트 키워드 확인 | 키워드 확인 | 키워드 확인 |

## 사용자 지침서

|  |  |  |  |
| --- | --- | --- | --- |
|  | | | |
| 그림 | 10. | Home | 화면 |

그림 8은 PressCheck 홈페이지에 처음 접속 했을 때 나오는 화면입니다. 1번 영역의 가장 많이 본 뉴스에서는 수집된 뉴스에서 가장 조회 수가 높은 기사를 제공합니다. 2번 영역을 통해 랭킹의 뉴스 분석 리포트 화면으로 이동할 수 있습니다. 3번 영역을 통해 수집된 뉴스 전체 목록을 볼 수 있는 페이지로 이동합니다.

|  |  |  |  |
| --- | --- | --- | --- |
|  | | | |
| 그림 | 11. | Newslist | 화면 |

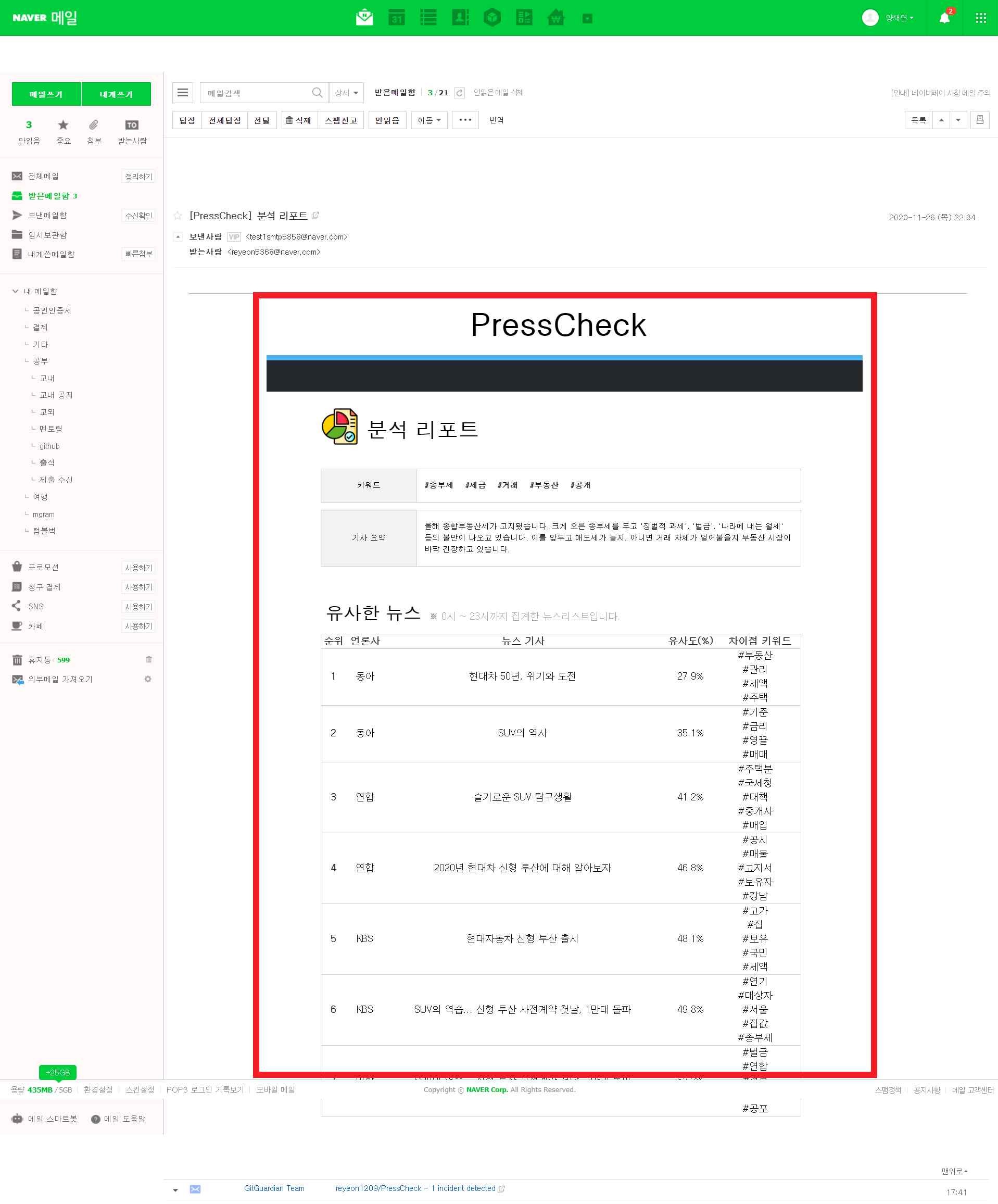
앞서 3번 영역의 더보기를 통해 이동하면 전체 뉴스 목록을 열람할 수 있습니다. 1번 영역을 통해 원하는 카테고리와 언론사를 선택하여 목록을 열람할 수 있습니다. 2번 영역을 통해 기사를 선택하면 선택된 기사의 분석 리포트 페이지로 이동합니다.

|  |  |  |  |
| --- | --- | --- | --- |
|  | | | |
| 그림 | 12. | Analyze | 화면 |

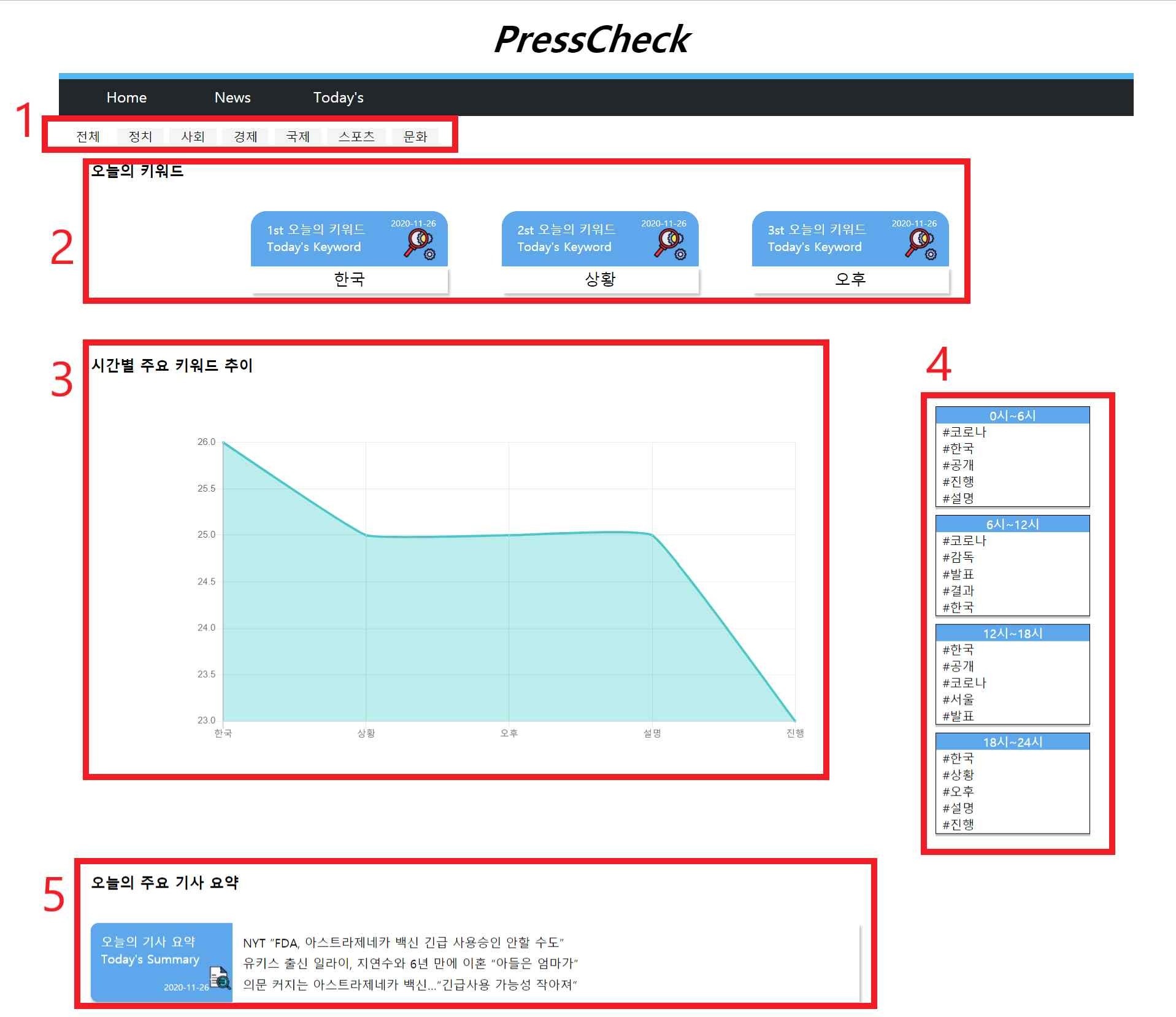
뉴스를 클릭하게 되면 해당 뉴스에 대한 본문 정보와 분석리포트를 보여주는 페이지로 이동합니

다. 1번의 탭 항목으로 자유롭게 페이지 전환이 가능합니다. 2번 영역에서는 해당 뉴스에 대한 본문 정보를 제공하고 있습니다. 3번 영역에서는 해당 뉴스에 대한 키워드와 본문 요약 기능을 제공합니다. 5번 영역에서는 이 기사와 가장 유사도가 낮은 순으로 타깃 기사들을 정보와 어떤 키워드에 차이가 있는지 목록을 제공합니다. 4번 영역을 통해 이메일로 분석 리포트를 받을 수 있게 합니다.

|  |  |  |  |
| --- | --- | --- | --- |
|  | | | |
| 그림 | 13. | Mailing | 화면 |

이메일 팝업창에 이메일을 입력하게 되면 해당 뉴스에 대한 분석 리포트가 이메일로 발송됩니다.

|  |  |  |  |
| --- | --- | --- | --- |
|  | | | |
| 그림 | 14. | Today’s | 화면 |

Today's 탭으로 이동하게 되면 카테고리 별 상위 키워드, 시간별 키워드 추이, 오늘의 기사 요약을 열람할 수 있습니다. 1번 영역을 통해 카테고리 전환이 가능합니다. 2번 영역에서는 하루 동안 현 시 간까지의 가장 많이 검출된 키워드를 보여줍니다. 3번 영역에서는 시간대별로 어떤 키워드가 많이 검 출되는지 그래프로 보여줍니다. 4번 영역에서는 시간대 별로 가장 많이 검출된 키워드를 볼 수 있습 니다. 5번 영역에서는 오늘의 주요 기사3개를 요약된 형태로 보여줍니다.

## 운영 상태 관리

#### 장애관리

1. 채팅 목록 창 오류

- 유저의 면수가 늘어남에 따라 튕김 현상이 발생할 수 있음

1. 카카오 로그인

- 로그인이 되지 않는 유저가 있음 (튕김 현상 발생)

1. 앨범 불러오기

- 앨범의 사진을 불러오지 못하는 경우가 있음

#### 추가

#### 앱 설정 – 푸시 알림 기능, 캐시 데이터 삭제

#### 학교 분류 메뉴

#### 신고하기 기능

#### 리뷰 평가 항목 중 게시물 좋아요 총 개수 추가

#### 공지사항

#### 자주 하는 질문

## 보안 관리

#### 신고 하기

1. 항목의 선택지를 준다(ex 판매금지 품목, 사기글, 기타사유 등)
2. 이용자의 신고 항목도 포함한다. (ex tnvnfdla12 님을 신고하고 싶으신가요??)
3. 해당 선택지는 운영자가 확인 후 제재를 행한다.

첫 번째로 파이어 베이스를 이용함에 따라 좀더 데이터베이스의 용도가 제한적 이였다. 추후 파이어 베이스에서 제공하는 라이브러리가 늘어남에 따라 추가할 예정이다.

두 번째로 인터넷에 올라온 오픈소스를 이용함에 따라 변형이 제한적 이였다. 코딩의 실력이 늘어남에 리뉴얼 할 예정인다.

세 번째로 UX/XI 디자인을 계속 변형해서 쓸데없는 시간을 소비하였다. 다음부터는 계획부터 잡고 갈 것이다.

# 향후 계획

1. UX/UI 디자인 리뉴얼
2. 관리자 인계
3. 이용자 증가 계획