0. 서론

0.1 개발 배경

최근 코로나로 인해 많은 사람들이 외출을 꺼려 하며 집에서 많은 시간을 보내고 있다. 이에 따라 소비자의 욕구를 충족시킬 수 있는 다양한 배달 서비스들이 생겨났고 그에따라 배달 업계는 급격하게 성장하게 되었다. 그러나 성장과 더불어 배달에 관련된 다양한 문제점들 또한 야기되면서 많은 사람들이 불편함을 느끼고 있다.

배달 서비스를 이용하기 위해서는 음식을 주문한 고객뿐만 아니라 음식점 측에서도 배달 라이더에게 배달 비용을 지불해야 한다. 따라서 음식점 배달 서비스를 이용하기 위해서는 자영업자가 손해를 보지 않기 위해 배달 최소금액이 존재하게 된다. 하지만 대부분의 음식점이 1인분 가격이 배달 최소금액에 미치지 못하기 때문에 1인 가구가 주문할 경우 어쩔 수 없이 사이드 메뉴를 시키거나 2인분을 주문하는 상황이 발생하게 된다. 결국 소비자는 배달 최소금액을 위한 가격 부담이 점점 늘어나게 된다. 이는 단순히 금전적인 문제뿐만 아니라 음식물, 일회용 쓰레기 증가 등 다양한 환경에도 악영향을 미치고 있다.

설문조사 결과 약 80%의 소비자들이 배달 최소금액으로 인한 불편을 경험하고 있다. 이러한 문제점의 해결방안을 찾기 위해서 최소 주문 금액을 맞출 수 있도록 도와주는 앱이 있으면 좋겠다 생각했고 “배달 셰어” SNS 애플리케이션을 고안하게 되었다.

0.2 어플리케이션 컨셉 구상

1. 회원정보를 기입하여 회원가입 후 로그인한다.

2. 사용자는 음식 메뉴, 주소, 상세 정보들을 기입한 뒤 게시글을 올린다.

3. 게시판에 자신의 주소 근처의 게시물들이 가까운 순으로 정렬되어 보여진다.

4. 게시글 주소 근처의 같이 주문하고픈 사람들이 해당 게시글에서 글쓴이와 상세정보들을 확인 가능하다.

5. 조건에 맞는다면 해당 게시글의 메신저에서 함께 소통하여 배달 최소금액을 맞추며 한번에 배달을 주문할 수 있다.

6. 배달 음식을 받고 난 뒤, 사용자들끼리 서로를 평가하여 레이팅을 매긴다.

1. 과제 제안서

1.0 팀 소개

코로나 펜대믹으로 인해 배달 업계는 호황을 맞이했고 배달 음식 주문은 증가하고 있습니다. 하지만 역설적이게도 배달 음식의 증가는 사용자와 자영업자, 모두에게 악영향을 미치고 있습니다. 바로 배달시 필요한 최소 주문 금액 때문입니다. 저희는 이러한 최소 주문 금액 문제를 해결할 수 있는 SNS형 애플리케이션 “**배달 쉐어**”를 구축하고자 합니다.

**조장: 201736009 컴퓨터공학과 이정재**

애플리케이션 구현에 필요한 문제 정의서, 요구사항 명세서, UseCase 등을 작성할 예정입니다. 구현 측에서는 회원가입, 로그인, 사용자 프로필 설정, 애플리케이션 권한 설정, 데이터 베이스, 를 맞고 있습니다.

**조원: 201735982 컴퓨터공학과 석민제**

애플리케이션의 목적을 파악하기 위해 문제 정의서, 요구사항 명세서, UseCase 등을 조장과 함께 작성할 예정입니다. 구현 측면에서는 게시판, 게시글 권한, 게시글에 필요한 각종 상세정보, 채팅 메신저, 모바일 GPS를 이용한 위치파악 기능을 담당하고 있습니다.

1.1 문제정의서 (기술서)

**1.** 배달 최소 금액을 맞추기 위해, 함께 음식을 주문할 사용자를 위치 기반으로 구하는 SNS 애플리케이션을 구현한다.

**2.** 사용자는 회원가입 시 닉네임, 주소, 성별 등 필수 프로필 데이터를 기입해야 한다.

**3.** 프로필에는 닉네임, 성별, 평가정보, 등이 기술되며 다른 사용자들이 열람 가능하다.

**4.** 사용자는 로그인을 통해 SNS 애플리케이션에 대한 접근 권한을 얻는다.

**5.** 회원가입시 입력했던 주소 정보를 기본값(default value)으로 현재 위치가 설정된다. 또한 사용자의 필요에 따라 현재 위치를 변경할 수 있다.

**6.** 사용자의 현재 위치를 기준으로 근방에 다른 사용자들이 작성한 SNS 게시글을 메인 화면에 출력한다.

**7.** 사용자는 현재 주소를 기반으로 게시판에 게시글을 작성할 수 있다.

**8.** 사용자는 게시글 작성시 음식 메뉴, 게시글이 유지되는 최대 시간, 같이 주문할 최대 인원수를 필수적으로 입력하고 필요에 따라 추가 상제정보를 작성할 수 있다.

**9.** 게시판에는 필수정보인 음식 메뉴, 게시글이 유지되는 최대 시간, 같이 주문할 최대 인원수가 미리보기 형태로 표기된다.

**10.** 게시글이 유지되는 최대 시간이 지나거나 최대 인원이 넘을 경우, 해당 게시글은 비활성화 된다. 만약 중간에 인원이 빠져나간 경우 다시 활성화 된다.

**11.** 다른 사용자가 해당 게시글을 선택하여 주문 필수 정보, 추가 상세정보를 확인 가능하고, 추가적으로 게시글 작성자의 프로필 정보 또한 열람 가능하다.

**12.** 해당 게시글에서 신청을 누른 사용자들 간 SNS 메신저 기능을 사용할 수 있다.

**13.** 사용자들이 SNS 메신저 기능을 통해 대화를 주고 받고, 음식 배달에 대한 약속을 잡을 수 있다.

**14.** 주문 이후 사용자 간 평점을 매기고, 프로필 정보에 반영한다.

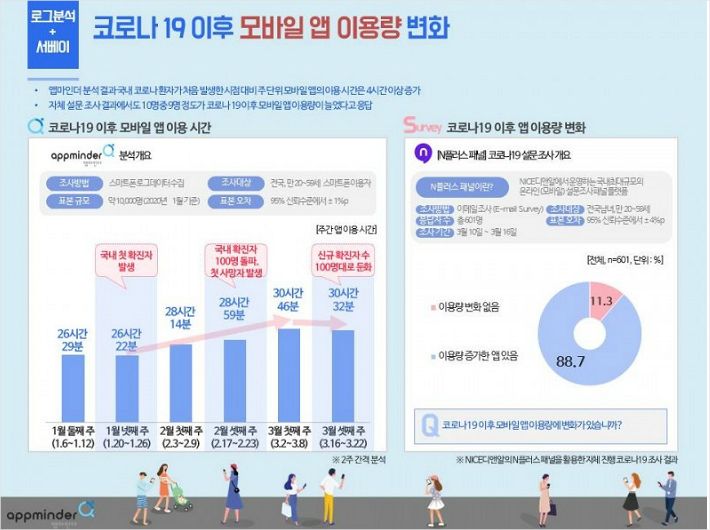
1.2 기술 동향

ㆍ배달앱의 이용 증가

O2O(online to offline) 비즈니스 중 갈수록 주목받는 분야 중 하나가 배달O2O서비스이다. 배달 O2O서비스는 배달앱을 통해 현재 제품 및 서비스를 받을 수 있는 업체를 확인하고, 주문, 결제, 도착에 이르는 전 과정을 담당하고 있다. 한국프랜차이즈 산업협회에 따르면 국내 배달앱 이용자 수는 2013년 87만 명에서 2020년 2,500만 명으로 급증했고, 거래규모도 3조원 규모로(중기이코노미, 2018. 10.1) 나타나고 있는데 이는 더욱 증가할 것으로 예상된다.

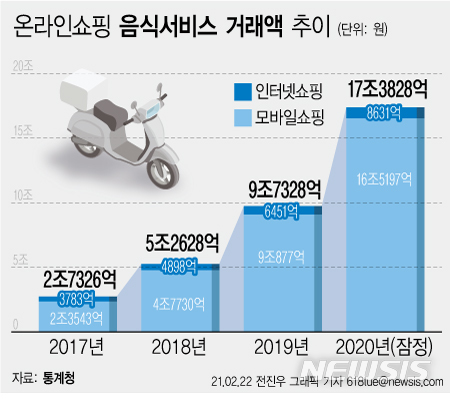
배달앱 시장의 성장으로 배달 대행서비스시장도 함께 확대되고 있다. 국내에서는 배달의 민족, 배달통, 요기요, 푸드플라이 등이 시장을 형성하고 있다.

ㆍ코로나 19 이후 모바일 앱 사용량 증가

 텍스트이(가) 표시된 사진

자동 생성된 설명

코로나19 장기화로 실내 활동이 늘어나면서 국내 스마트폰 이용자들의 모바일 앱 이용 시간이 전보다 크게 늘어난 것으로 나타났다. 분석 결과에 따르면 국내에서 코로나19 확진자가 처음 발생한 1월 넷째 주 스마트폰 이용자들의 주당 평균 앱 이용 시간은 26시간 22분으로 집계됐다. 하지만 코로나19가 한창 확산 중이던 3월 셋째 주에는 30시간 32분으로 4시간가량 늘어났다. 평균 앱 이용시간이 증가하면서 특정 앱 이용량 또한 늘었는데, 응답자 중 절반 이상이 "식음료 배달(51.8%) 앱 이용량이 증가했다"고 답했다.



또한 신종 코로나바이러스 감염증(코로나19) 팬데믹의 영향으로 지난해 배달 음식 주문이 크게 늘어난 것으로 조사됐다. 코로나19로 사회적 거리두기가 이어지면서 외출을 자제하고 집에 머무는 시간이 길어지면서 배달 음식을 주문하는 사람이 늘어났기 때문으로 분석된다.

22일 통계청에 따르면 지난해 치킨·피자 배달 같은 음식 서비스 거래액은 17조3828억원으로 전년(9조7328억원)보다 78.6%나 증가했다. 이는 관련 통계가 작성된 이후 최대치로 3년 새 536%나 증가했다. 이 가운데 모바일 주문이 16조5197억원으로 전체의 96.4%를 차지하는 등 모바일로 음식을 주문하는 비중이 높았다.

음식서비스 거래액은 배달음식만 별도로 통계를 작성하기 시작한 2017년 2조7326억원을 기록한 뒤 2018년 5조2628억원, 2019년 9조7328억원, 2020년 17조3828억원으로 연평균 85.4% 성장했다.

특히 코로나19가 재유행했던 지난해 8월에는 1조7050억원으로 전월보다 23.8% 늘었고 사회적 거리두기 강화로 식당 영업시간이 단축됐던 12월에는 2조1988억원으로 전월대비 34.2%나 증가했다. 코로나19가 배달음식 이용에도 영향을 미쳤다는 뜻이다.

ㆍ**소비자 편익과 피해**

배달앱 이용자들은 다양한 업체의 정보를 볼 수 있고, 한 번에 검색과 주문, 결제를 할 수 있는 점, 사용자의 리뷰를 확인할 수 있는 점 때문에 배달앱을 이용하는 경향이 있다. 이중 가장 중요한 요인은 한 번에 검색과 주문, 결제를 할 수 있다는 점이 꼽히고 있다. 이러한 점은 편의성(회원가입, 앱설치 및 접속의 편리, 다양한 결제수단의 제공, 검색·주문·결제 과정의 간단함, 메뉴와 업체 검색 용이)에 대해 소비자들이 대체로 우호적인 평가를 하는 것과 일맥상통한다(DMC MEDIA, 2019 배달 앱 이용 형태 조사). 반면에 배달비와 최소주문금액, 수수료, 등에 대해서는 소비자는 불만사항으로 지적하고 있다. 이처럼 배달앱은 소비자의 편의를 높이는데 기여한 측면이 있지만 피해도 늘고 있어 소비자 피해 예방 및 구제조치가 필요한 상황이다.

ㆍ**최소 주문 금액 관련 문제**

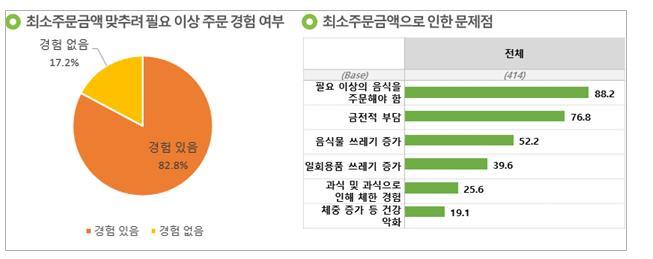
배달 애플리케이션(앱) 이용자 10명 중 8명은 최소 주문금액을 맞추기 위해 필요 이상으로 주문을 한 경험이 있다는 조사 결과가 나왔다. 배달비를 포함해 배달앱 이용 때 1회 평균 지출액은 2만∼3만원이 51.8%로 가장 많았다. 이어 2만원 미만이 32.8%, 3만원 이상은 15.4%였다. 조사대상 중 82.8%는 최소 주문금액을 맞추기 위해 필요 이상으로 주문한 경험이 있다고 답했다.

ㆍ**환경 오염 관련 문제**

1인 가구 증가와 함께 배달 서비스 애플리케이션(앱) 이용자가 급증하면서 일명 ‘악성쓰레기’도 늘었다. 악성쓰레기는 먹다 남은 음식물과 플라스틱 용기를 그대로 종량제 봉투에 넣어 버리는 혼합배출로 발생하며 재활용할 수 없어 환경오염의 ‘주범’으로 꼽힌다.

10일 공정거래위원회와 업계에 따르면 배달 앱 시장 이용자는 2013년 87만 명에서 지난해 2500만 명으로 급증했다. 같은 기간 거래규모는 3347억 원에서 지난해에는 약 3조원으로 훌쩍 뛰었다. 5년 새 약 10배가량 덩치가 커졌다.

배달 앱 시장의 성장 추이는 공교롭게도 악성쓰레기 배출량이 크게 늘던 시기와 맞물린다. 환경부에 따르면 가정에서 발생한 전국 음식물쓰레기 배출량은 2013년 140t에서 배달 앱이 활성화한 2014년 425t으로 세 배 이상 늘었다. 이후 △2015년 972t △2016년 1127t △2017년 1333t으로 매년 점증했다. 같은 시기 플라스틱 배출량도 △2509t △2656t △2858t △3265t △3546t으로 증가했다.



소비자연맹 관계자는 "최소 주문금액으로 원래 시키려던 양보다 많은 음식을 주문해 금전적 부담과 음식물·일회용품 쓰레기 증가 등의 불편함을 느끼는 소비자가 많다"면서 "최소 주문금액 수준과 배달비의 이중부담 문제도 개선되어야 할 것으로 보인다"고 말했다.

ㆍ**배달비 관련 소비자 부담**

코로나19(COVID-19)로 음식 배달시장이 급성장한 가운데 배달비 관련 소비자들의 불만도 커지고 있다. 배달 업계에서는 라이더(배달 기사) 부족과 '단건 배달' 열풍으로 배달비 인상이 불가피 할 것으로 진단한다.

배달비는 주문자가 내는 배달팁과 업주가 부담하는 배달료로 구분된다. 각 지역의 라이더 지사에서 거리나 상권에 따라 배달비를 정하면 업주는 자신이 낼 배달료와 고객이 부담할 배달팁을 분배한다. 라이더가 받는 배달비가 5000원이라면 고객이 2000원, 업주가 3000원 내는 식이다.

'단건 배달' 확산 분위기에서 배달비 폭증을 염려하는 목소리도 있다. 3년 전 2000원에 불과했던 배달비는 이미 평균 5000원까지 올랐다. 단건 배달은 한 번에 3~4건을 배달하는 기존 '묶음 배달'보다 라이더 수급에 더 큰 어려움을 겪는다. 이 때문에 배달앱은 적자를 감수하고 보조금으로 라이더를 유인하는 상황이다. 결국 장기적으로는 배달비의 지속적인 상승이 불가피해 보인다.

ㆍ**자영업자 배달비 부담 증가**

최근 비대면 수요 증가에 따른 배달시장이 확대되면서 배달대행 비용 부담과 업체별 배달비 경쟁으로 인한 자영업자들의 마이너스 매출이 지속되고 있다. 한식업종을 운영 중인 윤모씨는 “배달비는 업주가 설정하는데 1만2000원짜리 음식을 시키면 기본 배달비 6000원 정도가 나온다. 소비자는 배달비를 업주와 반반 부담하기 때문에 상대적으로 저렴한 업체를 선호한다”며 “배달비조차 타 업체와 경쟁해 낮은 금액으로 설정하면 그 또한 업주가 부담해야 하기 때문에 재료값도 남지 않는 상황으로 배달을 계속해야 하나 고민인 상황”이라고 말했다. 이 같은 문제는 코로나19 확산으로 인한 배달요금 상승에 따른 것으로 풀이된다.

실제 불과 3년 전 2000원이었던 배달비는 배달 수요 급증과 라이더 부족 현상으로 평균 5000원대까지 상승했으며, 최근 거리별 할증 금액까지 요금에 추가되면서 많게는 2만5000원까지 배달비가 책정되고 있는 상황이다.

**1.3 예상 사용 기술**

ㆍ**Firesbase의 Firestore database**

Cloud Firestore는 Firebase 및 Google Cloud의 모바일, 웹, 서버 개발에 사용되는 확장 가능한 데이터베이스이다. Firebase 실시간 데이터베이스와 마찬가지로 실시간 리스너를 통해 클라이언트 애플리케이션 간에 데이터의 동기화를 유지하고 모바일 및 웹에 대한 오프라인 지원을 제공해 네트워크 지연 시간이나 인터넷 연결에 상관없이 원활하게 반응하는 앱을 개발할 수 있다.

ㆍ**Firebase의 Realtime database**

Firebase 실시간 데이터베이스는 클라우드 호스팅 데이터베이스이다. 데이터는 JSON으로 저장되며 연결된 모든 클라이언트에 실시간으로 동기화 된다. iOS, Android, 자바스크립트 SDK로 크로스 플랫폼 앱을 빌드하면 모든 클라이언트가 하나의 실시간 데이터베이스 인스턴스를 공유하고 자동 업데이트로 최신 데이터를 수신한다.

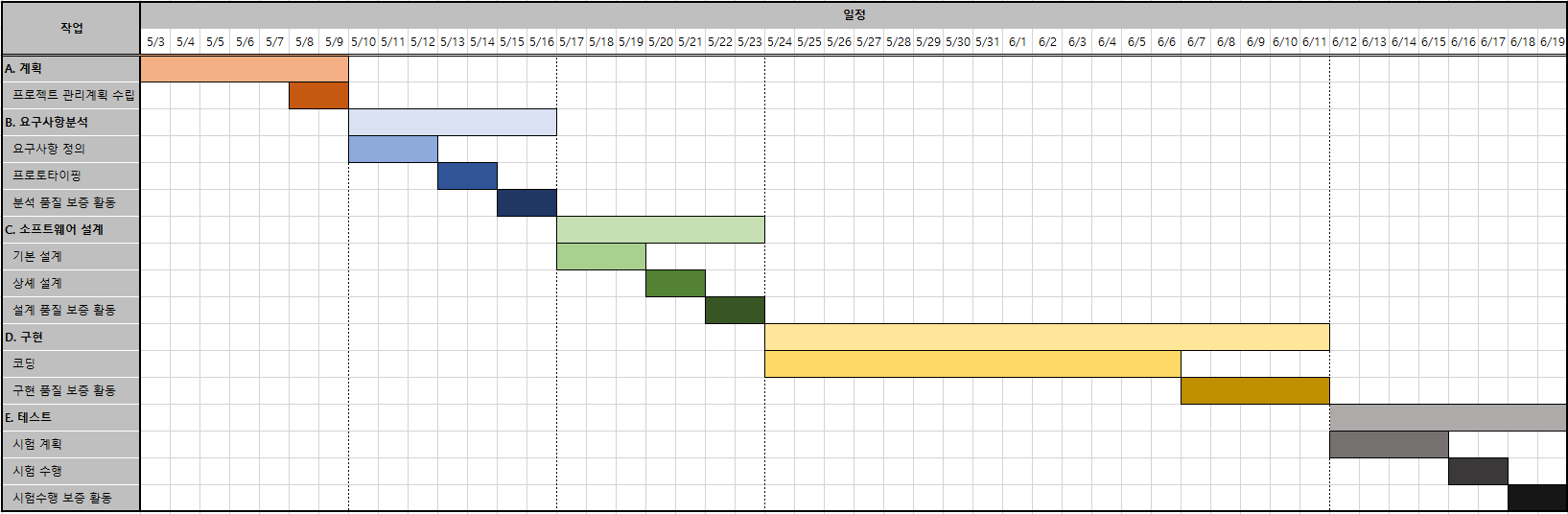
ㆍ**Google Maps API**

google에서 제공하는 위치 정보를 기반으로 한 API이다. GIS 규격을 기준으로 데이터를 처리한다.

ㆍ**Android Studio**

구글이 안드로이드 앱 개발을 위해 JetBrains 사의 IntelliJ IDEA를 기반으로 만든 통합 개발 환경이다. 2014년 10월부터 이클립스를 대체해 안드로이드의 공식 IDE가 되었다. 무료로 제공되고 있으며 Windows, macOS, Linux를 지원한다. 2017년 5월부터 안드로이드의 공식 프로그래밍 언어로 채택된 Kotlin도 사용 가능하다.

1.3 프로젝트 일정



1.4 기대 효과

**1. 일회용 쓰레기 감소**

사용자들이 배달의 민족, 쿠팡 잇츠, 요기요 등등 기존 배달 애플리케이션만 사용했을 경우 배달 최소 주문 금액을 맞추기 위해 불가피하게 추가적인 주문이 필요했다. 이에 따라 음식 포장에 필요한 추가적인 일회용품이 발생하고 환경에 악영향을 미친다. 하지만 “배달 셰어" 애플리케이

션을 사용할 경우 최소 주문 금액을 맞추기 위한 불필요한 주문이 필요 없게 되고 일회용품 사용을 감소시킬 수 있다.

**2, 음식물 쓰레기 감소**

배달 최소 금액을 맞추기 위해 사용자들의 추가 주문은 강요된다. 따라서 사용자 본인이 먹을 수 있는 음식의 양보다 추가 주문하게 되고 결국 음식물 쓰레기가 발생하게 된다. 이렇게 추가적으로 발생한 음식물 쓰레기는 사용자 뿐만 아니라 환경 관점에서도 악영향을 미친다. 하지만 “배달 쉐어” 애플리케이션을 사용하면 사용자가 강제적으로 추가 구매를 할 필요가 없어지며 이러한 문제를 해결할 수 있다.

**3.사용자 부담 감소**

사용자가 배달 서비스를 이용하기 위해서는 배달 최소 주문 금액을 맞추어야 한다. 예를 들어 8000원짜리 돈가스만 시키고 싶지만 최소 주문 금액이 10000인 경우 2000원 이상의 음식을 강제적으로 추가 주문해야 한다. 이는 사용자에게 금전적인 부담을 준다. 하지만 “배달 셰어"를 애플리케이션을 사용할 경우 사용자가 최소 주문 금액을 맞추는 데 도움을 주며 사용자의 부담을 덜어 줄 수 있다.

**4. 자영업자의 배달비 부담 감소**

배달 시스템 구조상 배달 수수료는 사용자 뿐만 아니라 자영업자까지 부담한다. 따라서 배달 주문이 많아질 경우 역설적으로 자영업자에게 부담이 커진다. 따라서 자영업자가 최대 수익을 내기 위해서는 배달 음식 주문이 적으면서 음식 주문이 많아야 한다. 하지만 기존에 배달 애플리케이션만을 사용할 경우 이는 불가능하다. 따라서 “배달 셰어"를 통해 다수의 사용자들이 많은 음식을 주문한다면 자영업자는 음식 주문 양을 증가시키면서 배달 수수료에 대한 부담을 줄일 수 있다.

1.5 참고 문헌

안수형.(2020).국내규제동향.배달을 둘러싼 소비자보호 입법 동향과 전망 <https://blog.naver.com>

/koreareg/221950125179

김수영.(2020).노컷뉴스.코로나 장기화에 배달앱, 쇼핑앱 이용량껑충<https://www.nocutnews.c>

o.kr/news/5327314

미상.(2020).매일경제.배달앱 이용자 최소주문금액 맞추려 필요 이상주문 .https://m.mk.co.kr/ne

ws/business/view/2020/11/1158854/

미상.(2021).NEWSIS20.코로나에 배달음식시장 호황.https://newsis.com/view/?id=NISX202102

22\_0001347095

이동우.(2021).머니투데이.8천원 돈가스 배달비가 7천원.https://news.mt.co.kr/mtview.php?no

=2021042316173412211

강신우.(2019).이데일리.배달앱 뜨니 ‘악성쓰레기’ 늘었다.https://www.edaily.co.kr/news/read?n

ewsId=01193926622553536&mediaCodeNo=257

2. 요구사항 분석 결과물

2.1 문제 정의서

**1.** 배달 최소 금액을 맞추기 위해, 함께 음식을 주문할 사용자를 위치 기반으로 구하는 SNS 애플리케이션을 구현한다.

**2.** 사용자는 회원가입 시 닉네임, 주소, 성별 등 필수 프로필 데이터를 기입해야 한다.

**3.** 프로필에는 닉네임, 성별, 평가정보, 등이 기술되며 다른 사용자들이 열람 가능하다.

**4.** 사용자는 로그인을 통해 SNS 애플리케이션에 대한 접근 권한을 얻는다.

**5.** 회원가입시 입력했던 주소 정보를 기본값(default value)으로 현재 위치가 설정된다. 또한 사용자의 필요에 따라 현재 위치를 변경할 수 있다.

**6.** 사용자의 현재 위치를 기준으로 근방에 다른 사용자들이 작성한 SNS 게시글을 메인 화면에 출력한다.

**7.** 사용자는 현재 주소를 기반으로 게시판에 게시글을 작성할 수 있다.

**8.** 사용자는 게시글 작성시 음식 메뉴, 게시글이 유지되는 최대 시간, 같이 주문할 최대 인원수를 필수적으로 입력하고 필요에 따라 추가 상세정보를 작성할 수 있다.

**9.** 게시판에는 필수정보인 음식 메뉴, 게시글이 유지되는 최대 시간, 같이 주문할 최대 인원수가 미리보기 형태로 표기된다.

**10.** 게시글 작성자가 주문을 완료했을 경우 해당 게시글은 비활성화 된다.

**11.** 게시글이 유지되는 최대 시간이 지나거나 최대 인원이 넘을 경우, 해당 게시글은 비활성화 된다. 만약 중간에 인원이 빠져나간 경우 다시 활성화 된다.

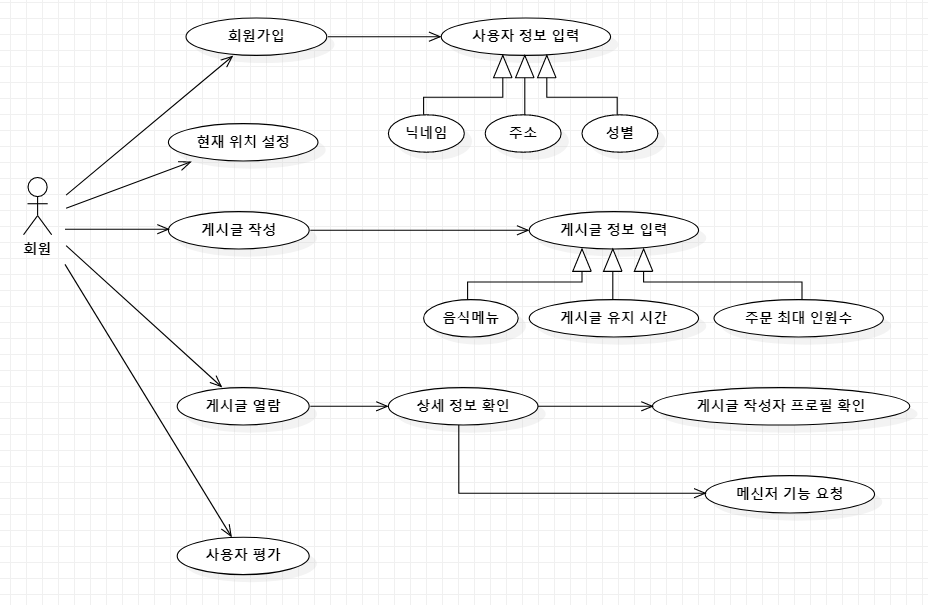
**12.** 다른 사용자가 해당 게시글을 선택하여 주문 필수 정보, 추가 상세정보를 확인 가능하고, 추가적으로 게시글 작성자의 프로필 정보 또한 열람 가능하다.

**13.** 해당 게시글에서 신청을 누른 사용자들 간 SNS 메신저 기능을 사용할 수 있다.

**14.** 사용자들이 SNS 메신저 기능을 통해 대화를 주고 받고, 음식 배달에 대한 약속을 잡을 수 있다.

**15.** 주문 이후 사용자 간 평점을 매기고, 프로필 정보에 반영한다.

2.2 유스케이스 다이어그램



2.3 유스케이스 기술서

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Use-Case Description | | | | | | |
| Use-Case명 | | 회원가입 | 작성일 | 2021.05.11 | 페이지 | 1/5 |
| Actor명 | 회원 | 작성자 | 석민제 |
| ■ **목적**  어플리케이션을 사용하기 위해 회원에 등록한다.  ■ **설명**  사용자는 회원가입 시 닉네임, 주소, 성별의 필수 프로필 데이터를 기입해야 한다. 추가로 사진과 같은 정보는 선택적으로 기입이 가능하다.  ■ **선행조건**  사용자가 현재 등록된 계정이 없다.  ■ **완료조건**  사용자가 필수 프로필 데이터를 기입하고 회원가입을 완료한다.  ■ **유형**  필수  ■ **Flow** | | | | | | |
| **번호** | **Actor** | | **System** | | | |
| **M01** | 회원 가입 요청 | |  | | | |
| **M02** |  | | 회원가입 작성 화면 출력 | | | |
| **M03** | ID, PW 입력 | |  | | | |
| **M04** |  | | [아이디가 중복되는 경우] (E01) | | | |
| **M05** | 닉네임, 주소, 성별 등 필수 정보 입력 | |  | | | |
| **M06** | 가입 신청 버튼 클릭 | |  | | | |
| **M07** |  | | [필수 데이터가 기입되지 않은 경우] (E02) | | | |
| **M08** |  | | 회원 가입 완료 | | | |
|  |  | |  | | | |
|  |  | |  | | | |
|  |  | |  | | | |
|  |  | |  | | | |
|  |  | |  | | | |
|  |  | |  | | | |
| **E01** |  | | 아이디가 중복되었다는 메시지 출력 | | | |
|  |  | | 회원 데이터 기입 화면 출력(M02) | | | |
| **E02** |  | | 필수 데이터 기입하라는 알림창 출력 | | | |
|  |  | | 회원가입 작성 화면 출력(M02) | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Use-Case Description | | | | | | |
| Use-Case명 | | 현재위치 설정 | 작성일 | 2021.05.11 | 페이지 | 2/5 |
| Actor명 | 회원 | 작성자 | 이정재 |
| ■ **목적**  회원의 현재 위치를 설정 한다.  ■ **설명**  회원의 현재 위치를 재설정 하고 현재 위치에 맞는 게시글을 사용자에게 출력할 수 있도록 한다.  ■ **선행조건**  회원으로 로그인 되어 있어야 한다.  ■ **완료조건**  회원의 현재 위치가 재설정 된다.  ■ **유형**  필 수  ■ **Flow** | | | | | | |
| **번호** | **Actor** | | **System** | | | |
| **M01** | 현재 위치 변경 기능을 요청한다. | |  | | | |
| **M02** |  | | 사용자 직접 입력 또는 현재 위치 추적 선택 화면을 출력한다. | | | |
| **M03** | [직접 입력일 경우] (A01) | |  | | | |
| **M04** | [위치 추적일 경우] (A02) | |  | | | |
| **M06** |  | | 현재 위치를 재설정 한다. | | | |
| **M07** |  | | 현재 위치를 기반으로 게시글을 업데이트한다. | | | |
| **A01** |  | | 현재 위치 입력 화면을 출력한다. | | | |
| **A101** | 주소를 입력한다. | |  | | | |
| **A02** |  | | GPS를 통해 사용자의 현재 위치를 추정한다. (E01) | | | |
| **A201** |  | | 사용자에게 현재 위치가 올바른지 물어본다. | | | |
| **A202** | [현재 위치가 올바를 경우] | | [현재 위치가 틀린 경우] (E02) | | | |
| **A203** |  | | 현재 위치를 재설정 한다. | | | |
|  |  | |  | | | |
| **E01** | [GPS를 사용할 수 없는 경우] | |  | | | |
|  |  | | GPS를 사용할 수 없음을 출력한다. | | | |
| **E02** | 현재 위치가 틀림을 알린다. | |  | | | |
|  |  | | 현재 위치 입력 화면을 출력한다. (M03) | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Use-Case Description | | | | | | |
| Use-Case명 | | 게시글 작성 | 작성일 | 2021.05.11 | 페이지 | 3/5 |
| Actor명 | 회원 | 작성자 | 석민제 |
| ■ **목적**  배달 최소 금액을 맞추기 위해 함께 음식을 주문할 회원을 구하기 위한 게시글을 작성한다.  ■ **설명**  회원은 게시글 작성시 음식 메뉴, 게시글이 유지되는 최대 시간, 같이 주문할 최대 인원수를 필수적으로 입력하고 필요에 따라 추가 상세정보를 작성할 수 있다.  ■ **선행조건**  회원으로 로그인 되어있어야 한다.  ■ **완료조건**  사용자가 필수 게시글 데이터를 기입하고 게시글 작성을 누른다.  ■ **유형**  필수  ■ **Flow** | | | | | | |
| **번호** | **Actor** | | **System** | | | |
| **M01** | 게시글 작성을 선택 | |  | | | |
| **M02** |  | | 게시글 작성 화면을 출력 | | | |
| **M03** | 게시글 작성 데이터 기입 | |  | | | |
| **M04** | 게시물 작성 버튼 클릭 | | [필수 데이터가 기입되지 않은 경우] (E01) | | | |
| **M05** |  | | 게시물 작성 완료 | | | |
|  |  | |  | | | |
|  |  | |  | | | |
|  |  | |  | | | |
|  |  | |  | | | |
|  |  | |  | | | |
|  |  | |  | | | |
| **E01** |  | | 필수 데이터 기입하라는 알림창 출력 | | | |
|  |  | | 게시글 작성 화면 출력 (M02) | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Use-Case Description | | | | | | |
| Use-Case명 | | 게시글 열람 | 작성일 | 2021.05.11 | 페이지 | 4/5 |
| Actor명 | 회원 | 작성자 | 이정재 |
| ■ **목적**  다른 회원이 작성한 게시글을 열람한다.  ■ **설명**  회원 위치를 기반으로 주변 회원이 작성한 게시글을 출력한다. 게시글을 열람하고 상세정보를 확인할 수 있고 게시글 작성자와 메신저를 주고 받을 수 있다.  ■ **선행조건**  회원으로 로그인 되어 있어야한다.  ■ **완료조건**  사용자에게 게시글, 게시글 상세정보 또는 메신저 내용을 출력한다.  ■ **유형**  필 수  ■ **Flow** | | | | | | |
| **번호** | **Actor** | | **System** | | | |
| **M01** |  | | 게시글 목록을 출력한다. | | | |
| **M02** | 게시글 상세 정보를 요청한다. | |  | | | |
| **M03** |  | | 게시글 상세정보 및 프로필을 출력한다. (E01) | | | |
| **M04** | 메신저 기능을 요청 한다. | |  | | | |
| **M05** |  | | 게시글 작성자와 메신저를 연결한다. (E02) | | | |
| **M06** |  | | 메신저 화면을 출력한다. | | | |
| **M07** | 메시지를 보낸다. | |  | | | |
| **M08** |  | | 상대방이 보낸 메시지를 출력한다. | | | |
| **M09** | {M08-M09를 반복한다.} | |  | | | |
|  |  | |  | | | |
| **E01** | [해당 게시글이 비활성화 되었을 경우] | |  | | | |
|  |  | | 게시글을 읽을 수 없음을 알린다. | | | |
| **E02** | [상대방과 연결할 수 없을경우] | |  | | | |
|  |  | | 상대방과 연결할 수 없음을 출력한다. | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Use-Case Description | | | | | | |
| Use-Case명 | | 사용자 평가 | 작성일 | 2021.05.11 | 페이지 | 5/5 |
| Actor명 | 회원 | 작성자 | 석민제 |
| ■ **목적**  주문이 완료되고 회원끼리 서로 평가한다.  ■ **설명**  공동 주문이 완료된 후, 서로의 매너를 평가하고 평점을 사용자 프로필에 반영한다.  ■ **선행조건**  회원으로 로그인 되어 있어야 하고, 같이 주문한 사용자들에 대해서만 평가할 수 있다.  ■ **완료조건**  사용자가 평가를 완료한다.  ■ **유형**  선택  ■ **Flow** | | | | | | |
| **번호** | **Actor** | | **System** | | | |
| **M01** | 사용자 평가 기능 요청 | |  | | | |
| **M02** |  | | 사용자 평가에 대한 레이팅 바 출력 | | | |
| **M03** | 사용자 평가 버튼 클릭 | |  | | | |
| **M04** |  | | 평가 완료 | | | |

2.4 클래스 다이어그램

**2.4.1클래스 후보 찾기**

배달 최소 금액, 음식 메뉴, 사용자, 위치, SNS 어플리케이션, 닉네임, 주소, 성별, 프로필 정보, 평가정보, 다른 사용자, 접근권한, 현재위치, 게시글, 메인화면, 게시판, 최대 시간, 최대 인원, 상세정보, 필수정보, 미리보기, 작성자, 주문, 인원, 주문 필수 정보, SNS 메신저, 대화, 음식배달, 약속, 평점

**2.4.2클래스 후보 정제**

**ㆍ중복 클래스**: 필수 정보, 주문 필수 정보, 게시판, 작성자, 다른 사용자

**ㆍ모호한 클래스**: 미리보기, 배달 최소 금액, 대화, 음식배달, 약속, SNS 어플리케이션, 주문

**ㆍ속성**: 최대시간, 최대 인원수, 음식 메뉴, 닉네임, 주소, 성별, 프로필 정보, 상세정보, 평점

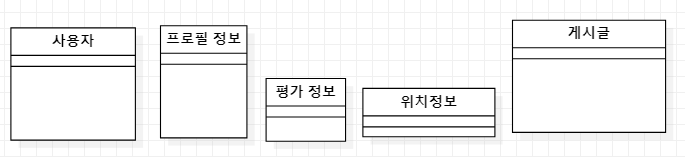
**ㆍ기능**: 메신저

**ㆍ구현**: 메인화면, 접근권한

**ㆍ구현 방법**:

**2.4.3 정제된 클래스**

**ㆍ**사용자, 프로필 정보, 평가 정보, 위치정보, 게시글



**ㆍ**정제된 클래스 외 속성은 attribute로 할당

텍스트, 시계, 스크린샷이(가) 표시된 사진

자동 생성된 설명

**2.4.4 연관관계 찾기**

배달 최소 금액을 맞춘다, 어플리케이션을 구현한다

사용자가 회원가입 한다, 필수 프로필 데이터를 기입한다.

프로필에 정보가 기술된다. 프로필 정보를 사용자가 열람가능하다

사용자가 로그인한다. 사용자는 어플리케이션의 접근권한을 얻는다.

현재 위치가 설정된다. 사용자가 현재 위치를 변경할 수 있다.

사용자 위치를 기준으로 게시글을 출력한다.

사용자는 게시글을 작성한다.

작성자는 게시글 필수 데이터를 입력한다. 사용자는 게시글 추가 상세정보를 작성한다.

게시판에 필수정보가 미리보기로 표기된다.

작성자는 주문을 완료한다, 게시글이 비활성화 된다.

게시글이 비활성화 된다. 인원이 빠져나간다, 게시글이 활성화 된다.

사용자가 게시글을 선택한다, 사용자가 상세정보를 확인한다, 사용자가 프로필 정보를 열람한다.

사용자가 게시글에서 신청을 누른다, 사용자가 메신저를 사용할 수 있다.

사용자들이 메신저를 통해 대화한다. 사용자들이 배달에 대한 약속을 한다. .

사용자들이 사용자간 평점을 매긴다, 평점을 프로필 정보에 반영한다.

**2.4.5 연관관계 정제**

**ㆍ구현**

어플리케이션을 구현한다

프로필에 정보가 기술된다. 사용자는 어플리케이션의 접근권한을 얻는다.

현재 위치가 설정된다, 사용자 위치를 기준으로 게시글을 출력한다.

게시판에 필수정보가 미리보기로 표기된다.

게시글이 비활성화 된다.

게시글이 활성화 된다.

평점을 프로필 정보에 반영한다.

**ㆍ행위 또는 기능**

배달 최소 금액을 맞춘다,

사용자가 회원가입 한다, 사용자가 로그인한다. , 필수 프로필 데이터를 기입한다.

작성자는 게시글 필수 데이터를 입력한다. 사용자는 게시글 추가 상세정보를 작성한다.

작성자는 주문을 완료한다. 인원이 빠져나간다, 사용자가 상세정보를 확인한다. 사용자가 프로필 정보를 열람한다.

사용자가 게시글에서 신청을 누른다,

사용자들이 메신저를 통해 대화한다 사용자들이 배달에 대한 약속을 한다. .

사용자가 메신저를 사용할 수 있다.

**2.4.6정제된 연관관계**

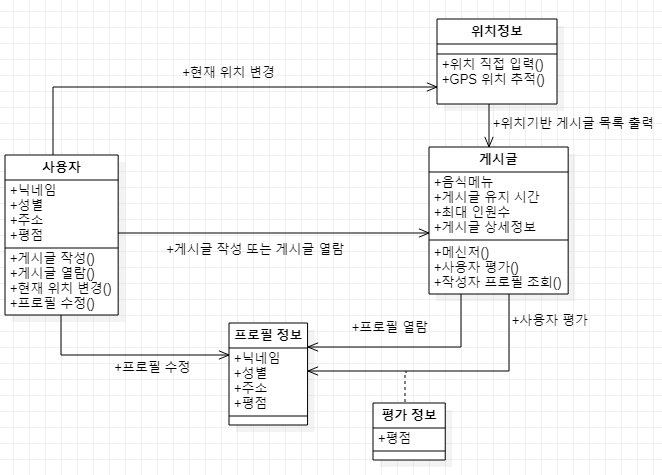
프로필 정보를 사용자가 열람 가능하다.

사용자가 현재 위치를 변경할 수 있다.

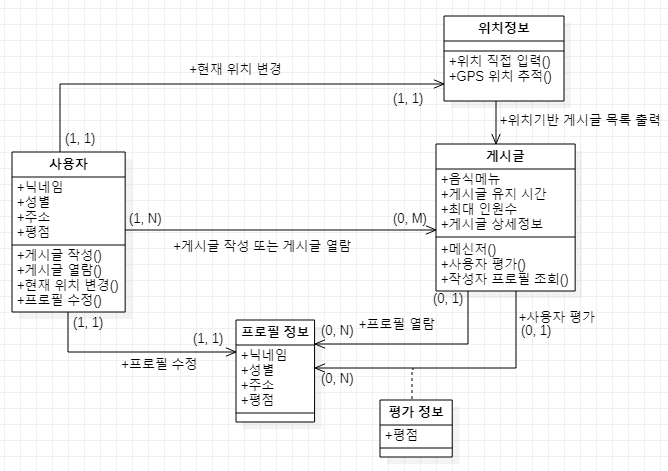
사용자는 게시글을 작성한다.

사용자가 게시글을 선택한다.

사용자들이 사용자간 평점을 매긴다.

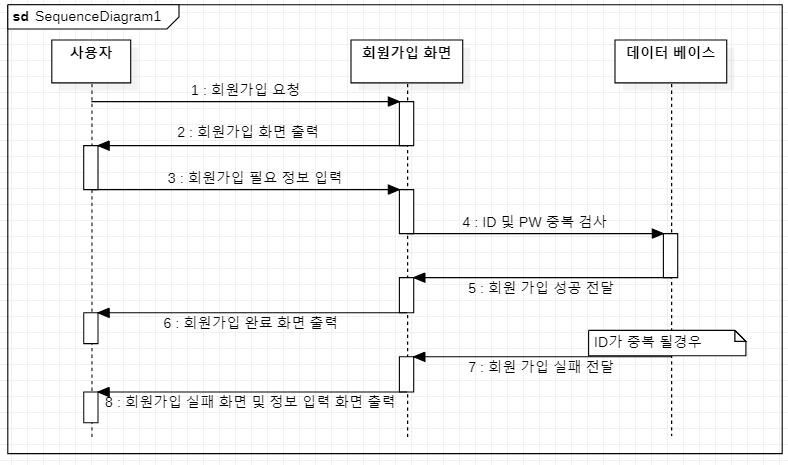


**2.4.7 도출된 클래스 다이어그램**

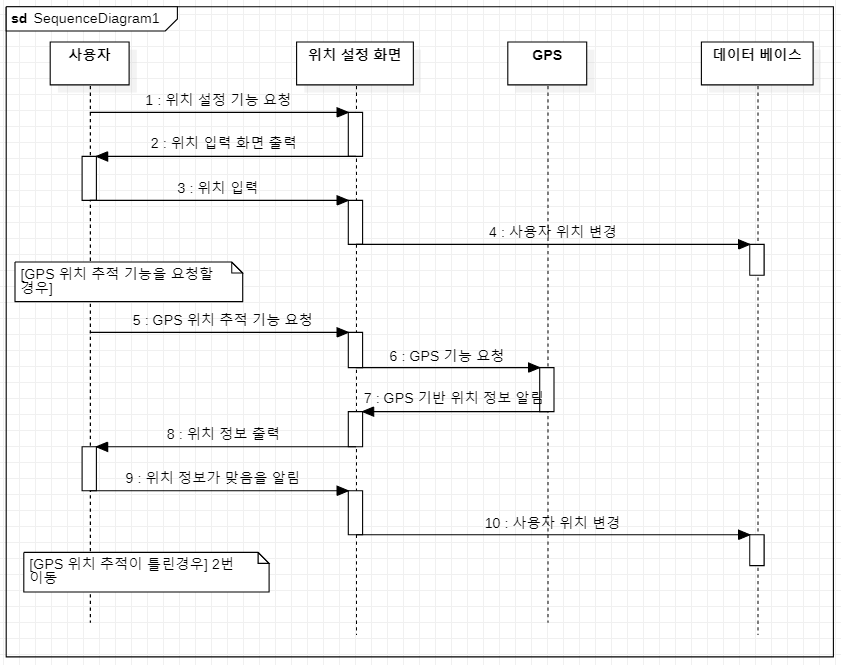


2.5 시퀸스 다이어그램

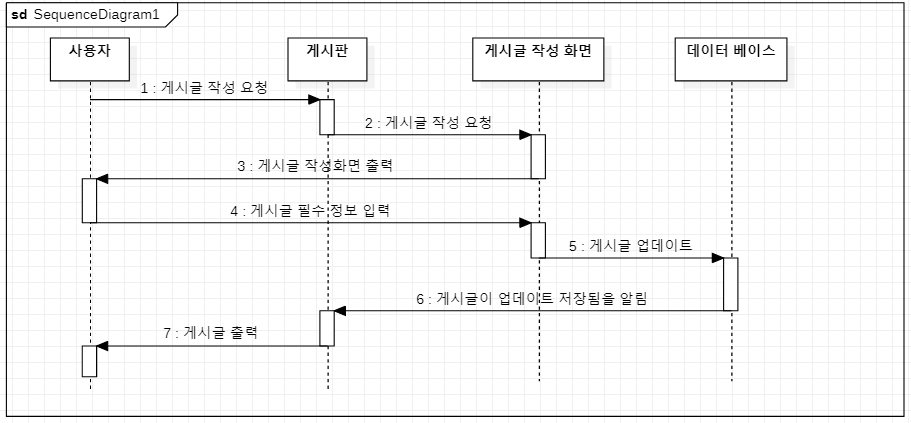
**ㆍ회원가입**



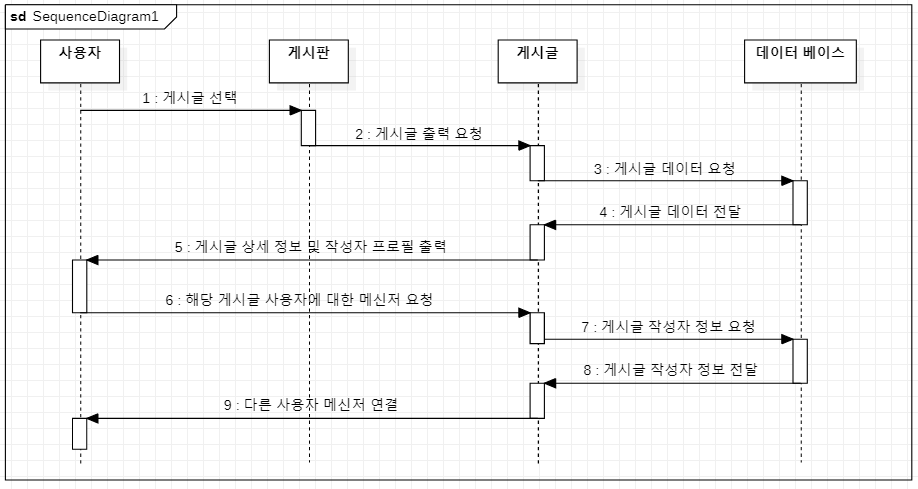
**ㆍ현재 위치 설정**



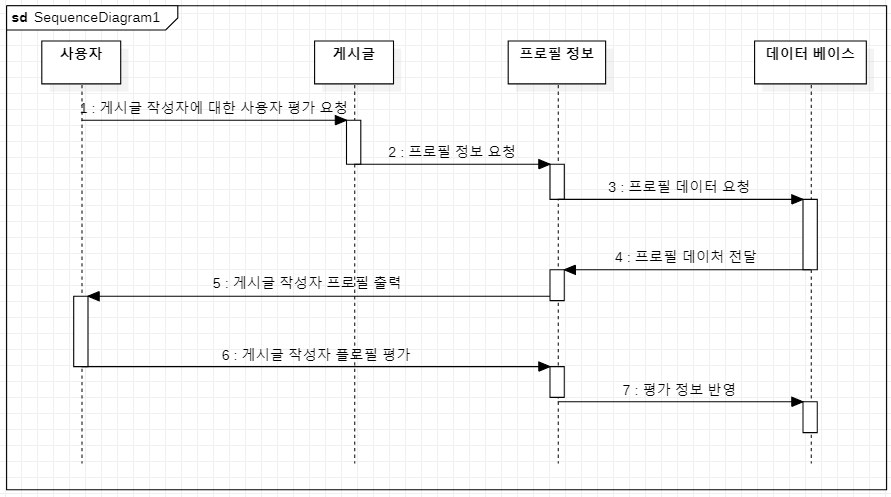
**ㆍ게시글 작성**



**ㆍ게시글 열람**

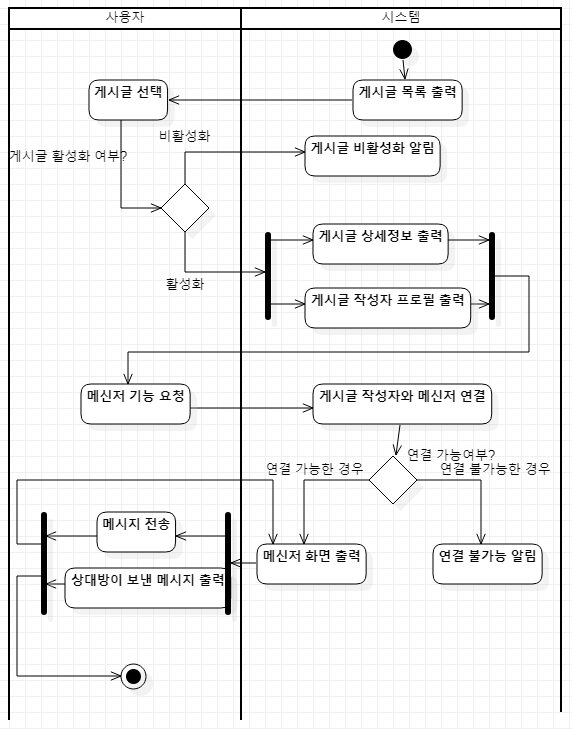


**ㆍ사용자 평가**



2.6 액티비티 다이어그램

**ㆍ게시판 열람**



2.7 요구사항 명세서

**2.7.1서론**

**ㆍ목적**

배달 최소 금액을 맞추기 위해 함께 음식을 주문할 사람을 구하는 SNS 애플리케이션을 구현하는 것이 목표이다. 따라서 기본적으로 사용자가 SNS 애플리케이션을 이용할 수 있도록 회원가입 및 로그인을 통해 SNS 애플리케이션에 접근하고 사용자가 함께 배달할 사람을 구할 수 있도록 SNS게시판을 구현한다.

**ㆍ범위**

필수기능: 회원가입, 로그인, 게시글 작성, 게시글 열람, 사용자 평가, 메신저

**ㆍ참고 문헌**

김정길,and 정지문. "안드로이드 기반 OpenAPI를 이용한 SNS 연동 지역정보 서비스를 위한 모바일 시스템 설계 및 구현." 디지털융복합연구 9.2 (2011): 131-140.

하울. (Udemy.com 베스트셀러 강사 하울의) Firebase로 안드로이드 SNS 앱 만들기. 서울: 비제이퍼블릭, 2018.

과학기술정보통신부 ·정보통신산업진흥원. “소프트웨어사업 요구사항 분석·적용 가이드.” (2021) 105~133

**2.7.2인터페이스 요구사항**

**ㆍ 시스템 인터페이스**

|  |  |  |
| --- | --- | --- |
| **인터페이스** | **정의** | **논리적 특성** |
| 회원가입 | 사용자의 프로필 정보를 입력하는 화면 | 앱 시작 시 메인 화면. 프로필 설정 화면으로 이동 |
| 로그인 | 회원 가입한 사용자의 ID, PW를 입력하는 화면 | 회원가입한 올바른 ID, PW 입력 시 게시판으로 이동 |
| 게시판 | 사용자의 위치에 기반하여 게시글 목록을 출력하며, 이때 게시글의 내용의 일부가 미리보기 형태로 표기되는 화면 | 디폴트 주소로 게시글 목록을 출력 |
| 게시글 열람 | 게시글에 입력한 상세 정보와 작성자의 간략한 프로필 화면을 출력하는 화면 | 열람될 때, 특정 조건을 충족하면 게시판에 해당 게시글이 비활성화 |
| 게시글 작성 | 게시글에 해당되는 정보를 기입하는 화면 | 필수 데이터들을 입력했을 시, 게시글로 등록되어 게시판에 활성화 |
| 위치 설정 | 사용자의 현재 위치를 설정하기 위한 화면 | 현재 위치가 바뀌었을 시, 게시판이 해당 위치 근처의 게시글 목록으로 업데이트 |
| 프로필 정보 | 사용자의 프로필 정보를 출력 및 수정이 가능한 화면 | 회원 가입 시, 입력받고 로그인 창으로 이동한다. 본인의 정보 변경 시, 해당 프로필 정보로 업데이트 |
| 사용자 평가 | 같은 게시물로 주문한 사용자끼리 서로 평가하는 화면 | 사용자를 평가하거나 창을 닫을 시, 게시판으로 이동 |
| 메신저 | 같은 게시물에 신청한 사용자끼리 연락이 가능한 대화 화면 | 메신저 종료 시, 사용자 평가 화면으로 이동 |

**ㆍ 인터페이스 세부 요구사항**

**(1) 회원가입 인터페이스**

1. ID를 입력받아 사용자가 구분될 수 있는 고유값을 생성한다.

2. ID에 상응하는 PW1를 입력받아 무분별한 접근을 막고, 해당 사용자의 비밀번호를 생성한다.

3. PW2를 입력하여 PW1과 비교하고, 비밀번호를 잘못 입력했을 경우를 대비한다.

**(2) 로그인 인터페이스**

1. 회원가입에서 생성한 ID와 PW를 입력받는다.

2. ID에 해당하는 PW가 일치하면 로그인에 성공한다.

3. ID에 해당하는 PW가 일치하지 않으면 실패 메시지를 출력한다.

**(3) 게시글 작성 인터페이스**

1. 주문할 음식 메뉴를 필수값으로 입력받는다.

2. 같이 주문할 최대 인원수 정보를 필수값으로 입력받는다.

3. 게시글이 유지될 시간을 필수값으로 입력받는다.

4. 추가 상세정보를 입력받아 작성자가 원하는 정보를 입력받는다.

5. 글 작성 중 “돌아가기 버튼”을 통해 게시판으로 돌아갈 수 있다.

6. 위 필수값을 모두 입력받고 “작성 완료 버튼”을 통해 게시글을 작성할 수 있다.

**(4) 위치 설정 인터페이스**

1. 사용자의 위치를 특정 API를 통해 직접 입력받는다.

2. GPS 기능 요청 버튼을 통해 GPS 기능을 요청한다.

3. GPS가 응답하지 않을 시, 위치 직접 입력 화면을 출력한다.

4. 위치 직접 입력 혹은 GPS 기능을 통한 위치 입력 둘 중 하나는 필수값이다.

5. 어떤 방식이던 위치를 입력받은 뒤, 주소 확인 버튼을 통해 재확인을 진행해야 한다.

6. 주소 확인 버튼이 눌린 후, 설정완료 버튼을 통해 현재 위치가 재설정 된다.

**(5) 프로필 설정 인터페이스**

1. 회원 가입 시, 평점을 제외한 모든 칸이 필수로 입력받아야한다.

2. 확인 버튼을 통해 회원가입을 완료할 수 있다.

3. 프로필 수정 시, 닉네임, 기본주소를 수정하고 “프로필 수정 버튼”을 누르면 완료된다.

**(6) 메신저 인터페이스**

1. 메시지 내용을 입력받아 상대방에게 전송한다.

2. 상대방이 전송한 메시지 내용을 출력한다.

**(7) 사용자 평가 인터페이스**

1. 레이팅 바를 입력받아 “평가 완료 버튼”으로 평가가 완료된다.

2. “돌아가기 버튼”을 통해 평가하지 않고 게시판으로 이동된다.

**(8) 게시판 인터페이스**

1. 작성된 게시글을 리스트 형식으로 출력한다.

2. 게시글 필수 정보를 간략하게 제목으로 출력한다.

**(9) 게시글 인터페이스**

1. 게시글에 작성된 음식 메뉴 데이터, 최대 인원수 정보, 유지 시간, 상세 정보를 출력한다.

2. 게시글 작성자의 프로필 정보를 출력한다.

3. “돌아가기 버튼”을 통해 게시판으로 돌아간다.

4. “메신저 신청하기 버튼”을 통해 사용자들간 메신저에 참여할 수 있다 .

5. 최대 인원 수가 모두 차거나, 게시글 유지 시간이 지나면 해당 게시글이 비활성화된다.

**ㆍ 외부 인터페이스 정의 (입력)**

**회원가입 인터페이스**

|  |  |  |
| --- | --- | --- |
| **변수 명** | **타입** | **내용** |
| Identification value | Byte32 | 애플리케이션 사용자에게 권한을 부여하고 사용자를 관리하기 위해 필요하다. |
| Password value1 | Byte32 | 해당 사용자가 맞는지 확인하기 위해 필요하다. |
| Password value2 | Byte32 | 비밀번호를 잘못 입력했을 경우를 대비하여 작성한다. |

**로그인 인터페이스**

|  |  |  |
| --- | --- | --- |
| Identification\_login\_value | Byte32 | 애플리케이션 사용자에게 권한을 부여하고 사용자를 관리하기 위해 필요하다. |
| Password\_login\_value | Byte32 | 해당 사용자가 맞는지 확인하기 위해 필요하다. |

**게시글 작성 인터페이스**

|  |  |  |
| --- | --- | --- |
| food\_type | String | 게시글 작성자가 주문할 음식 메뉴 데이터를 입력받는다. |
| maximum\_users | Int | 사용자가 지정한 최대 인원수 정보를 입력 받는다. |
| keeping\_time | Int | 게시글이 유지되는 시간을 입력 받는다. |
| extra\_data | String, Byte256 | 위 3개의 필수 입력 사항 외에 사용자가 추가하고 싶은 문자, 사진 등의 데이터를 입력 받는다. |
| make\_posts() | 필수적으로 food\_type, maximum\_users, keeping\_time 변수를 받고, 선택적으로 extra\_data를 input으로 받는다. 입력된 정보를 바탕으로 게시판에 올라갈 게시글이 작성된다. | |

**위치설정 인터페이스**

|  |  |  |
| --- | --- | --- |
| user\_location | String | 사용자의 현재 위치를 설정하기 위해 데이터를 입력 받는다.. |
| set\_location() | 기본값으로 사용자의 주소가default\_location으로 설정되고, 현재 위치로 재설정 시 직접입력 혹은 GPS 기능을 사용하여 현재 위치로 지정한다. | |

**프로필 설정 인터페이스**

|  |  |  |
| --- | --- | --- |
| user\_name\_value | String | 사용자의 닉네임을 재설정하기 위해 사용 |
| modify\_value() | 새롭게 입력된 프로필 데이터 변수들로 값을 재설정한다. | |

**메신저 인터페이스**

|  |  |  |
| --- | --- | --- |
| message\_send | String | 사용자의 메시지 내용을 입력 받는다. |
| send\_message() | 게시글로 만들어진 메신저방에서 메시지를 발신한다. | |

**사용자 평가 인터페이스**

|  |  |  |
| --- | --- | --- |
| score\_value | String | 안드로이드의 레이팅 바 인터페이스를 사용하기 때문에 string으로 값을 받는다. 추후 데이터를 처리 하기위해 string으로 받아온 값을 parseInt()를 통해 형변환하여 사용한다. |
| comments\_value | String | 사용자가 추가로 입력한 코멘트 데이터를 받아온다. |
| user\_grade() | 같이 서비스를 이용한 사용자들이 게시글 작성자를 평가한다. | |

**ㆍ 외부 인터페이스 정의 (출력)**

**게시판 인터페이스**

|  |  |  |
| --- | --- | --- |
| **변수 명** | **타입** | **내용** |
| post\_list | list[] | 사용자들이 작성한 게시글들을 목록의 형태로 출력하기 위해 사용한다. 웹 데이터 베이스에 저장된 게시글의 내용을 가져와 목록의 형태로 출력하기 위해 list를 사용한다. |
| post\_preview | String | 사용자가 게시판을 보고 한눈에 어떤글인지 대략적으로 파악할 수 있도록 게시글 내용의 일부를 출력한다. |
| printPosts() | 사용자의 위치인 location 변수를 input으로 하여 게시글들을 게시판에 출력한다. | |

**게시글 인터페이스**

|  |  |  |
| --- | --- | --- |
| food\_type | String | 게시글 작성자가 주문할 음식 메뉴 데이터를 출력한다. |
| maximum\_users | Int | 사용자가 지정한 최대 인원수 정보를 출력한다. |
| keeping\_time | Int | 게시글이 유지되는 시간을 출력한다. |
| exta\_data | String, Byte256 | 위 3개의 필수 입력 사항 외에 사용자가 추가하고 싶은 문자, 사진 등의 데이터를 출력한다. |
| writer\_score | String | 게시글 작성자의 평점을 나타낸다. 해당 내용은 다른 사용자에 의해 평가된 내용이다. 안드로이드 레이팅 바 인터페이스를 사용하기 때문에 string 타입으로 반환된다. |
| activation\_value | Boolean | 해당 개시글이 특정 조건을 만족하면 비활성화되고 그렇지 않을경우 활성화되었음을 나타내기 위해 사용한다. |

**프로필 데이터 인터페이스**

|  |  |  |
| --- | --- | --- |
| user\_name\_value | String | 사용자의 닉네임을 출력하기 위해 사용 |
| user\_gender\_value | Int | 사용자의 성별을 출력하기 위해 사용한다. |
| user\_location\_value | String | 사용자의 위치 정보를 출력하기 위해 사용한다. |
| user\_score\_value | String | 다른 사용자가 평가한 사용자의 정보를 출력하기 위해 사용한다. |
| print\_profile() | 게시글 작성자의 프로필을 조회한다. | |

**메신저 인터페이스**

|  |  |  |
| --- | --- | --- |
| message\_reveive | String | 상대방이 보낸 메신저 내용을 출력하기 위해 사용한다. |
| rcv\_message() | 게시글로 만들어진 메신저방에서 메시지를 수신한다. | |

**ㆍ사용자 특성**

코로나 여파로, 외식보다 배달음식을 선호하게된 1인가구부터 다인가구까지의 집단, 스마트폰을 어플을 사용하는 다양한 연령층의 사용자.

**ㆍ제약 사항**

작성자 프로필 확인 시의 주소는 사생활 노출의 우려가 있기 때문에, 해당 지역이지만 넓은 범위로 다른 사용자에게 표기된다.

본 애플리케이션의 목적에 맞지 않는 게시글 작성, 메신저 기능 이용과 같은 행위 적발 시 게시글이 삭제되고 강제 탈퇴 조치한다.

GPS는 실시간으로 사용자의 위치를 파악하는 것이 아니며, 게시글 탐색 및 작성 시 지정하는 위치 탐색 과정에서만 사용된다.

**ㆍ요구사항의 유보**

Google maps 마커를 활용한 게시판 표시

: 게시판의 병렬식 구성으로 사용자는 한눈에 자기 위치와 더 가까운 지점의 게시글을 찾거나 메뉴를 찾기 어렵다. 따라서 Google map 마커로 게시글을 표현하여 지도를 보며 자신의 위치와 게시글의 작성 지점을 알 수 있다. 현재의 모드와 Google map 버전을 동시에 제공하여 사용자의 편리성에 따라 선택하여 게시판 타입을 선택할 수 있다.

**2.7.3 기능/비기능 요구사항**

|  |  |
| --- | --- |
| **UseCase** | 회원가입 |
| **설명** | 사용자가 회원가입하여 SNS 애플리케이션에 대한 접근 권한을 얻을 수 있다. |
| **행위자** | 비회원 |
| **선행조건** | 사용자가 현재 등록된 계정이 없어야 한다. |
| **완료조건** | 사용자 회원가입 완료 |
| **기능 요구사항** | ㆍ사용자로부터 ID, PW1, PW2 값을 입력 받는다.  ㆍ ID, PW1, PW2 값을 웹 데이터 베이스에 등록한다.  ㆍ웹데이터 베이스에 사용자가 ID, PW 값을 저장하고 고유한 해쉬 값을 부여한다.  ㆍ등록된 해쉬값을 통해 사용자를 구분하고 관리할 수 있어야 한다.  ㆍ만약 ID값이 이미 데이터 베이스에 있을경우 사용자에게 이를 알린다. |
| **비기능 요구사항** | ㆍ신뢰성 요구사항  사용자가 입력한 ID, PW값이 데이터 베이스에 정확히 저장되어야 하며, 만약 사용자가 PW를 잘못 입력할 경우 이를 사용자가 알 수 있게 해야 한다.  ㆍ안전성 요구사항  사용자의 개인 정보 ID, PW, 주민등록 번호 등등 개인 신상정보가 안전하게 관리되어야 한다.  ㆍ구현 요구사항  FireStore를 웹 데이터 베이스로 사용한다. |
| **추가동작** | 없음 |

|  |  |
| --- | --- |
| **UseCase** | 현재위치 설정 |
| **설명** | 사용자의 현재 위치를 재설정 한다. |
| **행위자** | 회원 |
| **선행조건** | 사용자가 로그인 되어 있어야 한다. |
| **완료조건** | 사용자의 현재 위치가 재설정되어야 한다. |
| **기능 요구사항** | ㆍ사용자로부터 현재 위치 정보 값을 받는다.  ㆍ사용자로부터 받은 위치 정보를 특정API를 통해 시스템이 이해할 수 있는 데이터 형태로 변환한다.  ㆍ변환된 위치 정보를 데이터 베이스에 반영한다. 이때 사용자가 프로필 설정한 기본 위치는 변경되지 않아야 한다.  ㆍ임시로 변경된 사용자 위치 정보를 기반으로 주변에 다른 사용자들이 작성한 게시글 목록을 게시판에 출력한다.  ㆍ사용자가 GPS 추적 기능을 사용해 위치정보를 재설정할 경우 특정API를 사용하여 사용자 위치를 추적한다.  ㆍGPS를 통해 추적한 위치를 현재 위치 정보에 반영한다.  ㆍ만약 GPS를 통해 추적한 위치가 틀린경우, 사용자가 GPS를 통해 추정한 위치가 틀림을 시스템에 알릴 수 있어야 한다. |
| **비기능 요구사항** | ㆍ성능 요구사항  GPS를 통해 사용자 위치를 추적할 경우 오차 범위는 +-3m 이어야 한다.  ㆍ사생활 보호 요구사항  GPS 추적 기능을 통해 항상 사용자의 위치정보를 수집하면 안되며, 앱 사용중 또는 사용자가 원할때만 GPS 추적기능이 사용되어야 한다.  ㆍ표준요구사항  위치정보는 GIS 표준 데이터 규격 정보를 사용해야한다.  ㆍ구현 요구사항  Google Maps API, go location API를 사용하여 구현되어야 한다. |
| **추가동작** | 변경된 사용자의 위치를 기반으로 게시판의 게시글 목록이 업데이트 된다. |

|  |  |
| --- | --- |
| **UseCase** | 게시글 작성 |
| **설명** | 사용자가 게시글을 작성하고 이를 다른 사용자가 열람할 수 있다. |
| **행위자** | 회원 |
| **선행조건** | 사용자가 로그인 되어 있어야 한다. |
| **완료조건** | 사용자가 작성한 게시글이 데이터 베이스에 저장되어야 한다. |
| **기능 요구사항** | ㆍ사용자로부터 음식메뉴, 게시글 최대 유지시간, 최대 인원수 정보를 반드시 입력 받고 추가로 사용자가 작성한 정보를 입력받는다.  ㆍ게시글을 작성한 사용자의 현재 위치 정보와 사용자가 작성한 게시글을 데이터 베이스에 업데이트 한다.  ㆍ필수 데이터가 입력되지 않았을 경우 이를 사용자에게 알리고 다시 정보를 입력 받아야 한다. |
| **비기능 요구사항** | ㆍ구현 요구사항  사용자가 작성한 게시글은 데이터 베이스에 작성자의 위치정보와 함께 저장되어야 한다.  만약 사용자가 사진도 같이 업로드했을 경우 데이터 베이스에 URL 형태로 저장/관리 되어야 한다.  ㆍ법적 요구사항  사용자가 작성한 게시글이 법적으로 위배되지 않아야한다. 예를 들어 음란물 유포 등이 불가능해야 한다. |
| **추가동작** | 없음 |

|  |  |
| --- | --- |
| **UseCase** | 사용자 평가 |
| **설명** | 주문완료 후 게시글 작성자에 대한 평가를 할 수 있다. |
| **행위자** | 회원 |
| **선행조건** | 사용자가 회원으로 로그인 되어있어야 하며, 게시글 작성자와 같이 주문 완료해야 한다. |
| **완료조건** | 사용자 평가 완료 |
| **기능 요구사항** | ㆍ게시글 작성자의 매너 점수 등을 평가할 수 있다.  ㆍ게시글 작성자를 평가하기 위해서는 주문 완료한 기록이 있어야한다.  ㆍ사용자 평가 정보가 해당 사용자의 프로필에 반영되어야 한다. |
| **비기능 요구사항** | ㆍ구현 요구사항  특정 사용자가 악의적으로 레이팅 점수를 줄 수 있기 때문에 여러 사용자들의 정보를 평균 형태로 반영한다.  안드로이드 레이팅 바를 사용하혀 출력되고 사용자로부터 String 형태로 데이터를 받으며 시스템에서는 Int 타입으로 데이터를 관리한다. |
| **추가동작** | 없음 |

|  |  |
| --- | --- |
| **UseCase** | 게시글 열람 |
| **설명** | 사용자의 위치를 기반으로 주변 사용자들이 작성한 게시판에 게시글을 출력하고 사용자가 해당 게시글을 열람할 수 있다. |
| **행위자** | 회원 |
| **선행조건** | 사용자가 회원으로 로그인 되어 있어야하며 주변 사용자들이 작성한 글이 있어야 한다. |
| **완료조건** | 사용자에게 게시글 정보 출력 |
| **기능 요구사항** | ㆍ사용자의 현재 위치를 기반으로 주변 사용자가 작성한 게시글을 게시판에 출력한다.  ㆍ사용자가 특정 게시글을 선택했을 경우 선택된 게시글의 상세 정보를 출력한다.  ㆍ선택된 게시글은 작성자가 입력한 3개의 필수 정보 및 추가 정보, 작성자 프로필 정보를 출력해야한다.  ㆍ게시글에는 추가로 “메신저로 신청하기” 버튼을 클릭하여 게시글 작성자와 메신저를 주고받을 수 있어야한다.  ㆍ메신저는 이전에 해당 사용자와 주고받은 데이터를 일정 기간동안 유지할 수 있어야 한다.  ㆍ만약 주문 최대 인원수를 초과할 경우 “메신저로 신청하기” 버튼은 비활성화 되어야 하며, 특정 상황으로 인해 다시 최대 인원수를 충족시키지 못할 경우 “메신저로 신청하기” 버튼이 다시 활성화되어야 한다.  ㆍ게시글 작성자와 특정 사용자가 주문을 했을경우 게시글 작성자를 제외한 사용자들은 같이 주문한 사용자들 서로에 대해 평가할 수 있다. |
| **비기능 요구사항** | ㆍ구현 요구사항  사용자가 작성한 게시글을 저장하기 위해 FireBase의 FireStore 데이터 베이스를 사용한다.  메시저 기능을 구현하기 위해 RealTimeDataBase를 사용한다.  평가를 위해 안드로이드 레이팅바를 사용하여 해당 함수는 텍스형 데이터를 input, output 값으로 받는다. 따라서 parseInt()등의 함수를 사용하여 데이터를 Int형으로 변환하여 관리해야 한다.  ㆍ효율성 요구사항  모든 사용자의 게시글을 게시판에 출력할 경우 사용자는 같이 주문할 대상을 찾기 어려워진다. 따라서 현재 애플리케이션 사용자의 위치를 기반으로 주변 사용자가 작성한 게시글만 출력해야 한다.  ㆍ안전성  직접 만남을 유도하는 애플리케이션 이기 때문에 사용자간의 안전성 확보가 필수 적이다. 따라서 사용자 성별, 평점 등을 통해 게시글 작성자에 대한 정보를 미리 파악할 수 있어야 한다.  ㆍ법적 요구사항  음란물이 포함된 게시글 등 법적으로 위법되는 게시글은 삭제 및 관리 되어야한다. |
| **추가동작** | 없음 |

**2.7.4 시험 계획**

ㆍ**단위 테스트**

**회원가입**

ID와 PW가 데이터 베이스에 올바르게 저장되는지 테스트한다.

ID가 중복될경우 오류 메시지를 사용자에게 출력하는지 테스트한다.

PW1, PW2가 동일하지 않을경우 오류 메시지를 사용자에게 출력하는지 테스트한다.

**사용자 위치 설정**

사용자가 입력한 위치가 위치 정보를 데이터 베이스에 올바르게 저장되는지 테스트한다.

사용자가 입력 정보가 실제 위치정보와 일치하지 않을경우 사용자에게 오류 메시지를 출력하는지 테스트한다.

GPS를 사용하여 올바른 위치를 가져오는지 테스트 한다.

GPS기능이 작동하지 않을 때 사용자가 직접 위치를 입력할 수 있는 화면을 출력하는지 테스트한다.

**게시글 작성**

사용자가 필수 데이터를 입력하지 않았을 경우 오류 메시지를 출력하는지 테스트한다.

사용자가 작성한 게시글이 데이터 베이스에 저장되는지 테스트한다.

데이터 베이스에 저장된 게시글 데이터가 사용자별로 관리가 되는지 테스트한다.

**게시글 열람**

데이터 베이스에서 해당 게시글의 정보를 정확하게 가져오는지 테스트한다.

사용자의 위치를 기반으로 주변 사용자들이 작성한 게시글만 출력하는지 테스트한다.

게시글을 통해 작성자 프로필 정보를 올바르게 출력하는지 테스트한다.

“메신저로 신청하기” 버튼을 눌렀을 때 메신저 기능이 작동하는지 테스트 한다.

특정 조건을 만족했을 때 “메신저로 신청하기” 버튼이 활성화 또는 비활성화 되는지 테스트 한다.

**사용자 평가**

같이 주문한 사용자만 작성자를 평가할 수 있는지 테스트 한다.

사용자의 평점 정보가 데이터 베이스에 올바르게 업데이트 되는지 테스트한다.

사용자가 레이팅바를 설정하지 않고 “평가 완료” 버튼을 눌렀을 때 오류 메시지를 출력하는지 테스트한다.

**ㆍ통합 테스트**

로그인 후 사용자 정보가 없으면 프로필 설정 화면을 출력하는지 테스트한다.

로그인 후 사용자 프로필이 이미 설정되어 있다면 게시판을 출력하는지 테스트한다.

현재 위치설정을 하지 않았을 경우 프로필 설정시 입력한 주소로 설정되는지 테스트한다.

사용자가 평가를 원하지 않는 경우 “뒤로가기” 버튼을 눌렀을 때 게시판 화면을 출력하는지 테스트한다.

게시글 출력화면에서 “메신저로 신청하기” 버튼을 눌렀을 때 메신저 화면을 출력하는지 테스트한다.

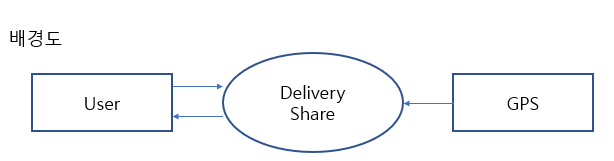
사용자가 메신저 화면에서 “뒤로가기” 버튼을 클릭했을 때 게시글 화면을 출력하는지 테스트한다.

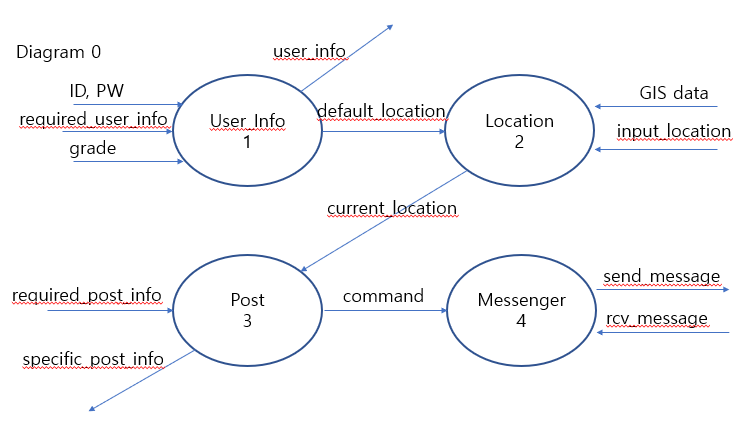
사용자가 게시글 화면에서 “뒤로가기”버튼을 클릭했을 때 게시판 화면을 출력하는지 테스트 한다.

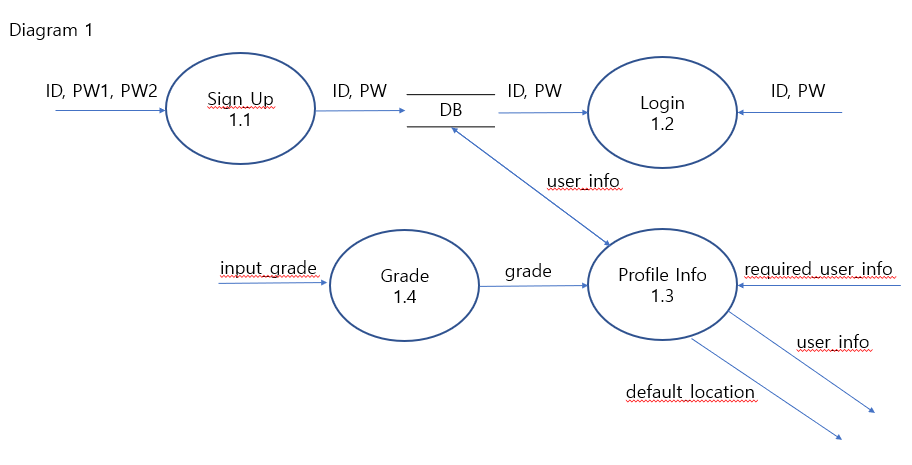
**3. 소프트웨어 설계 문서**

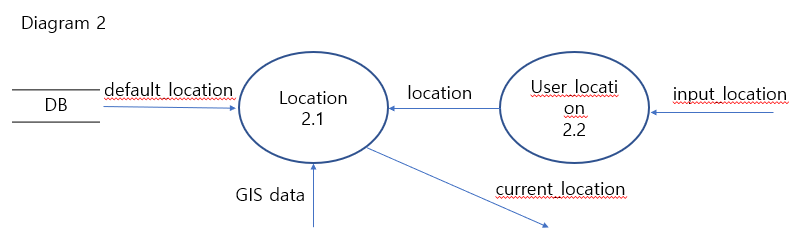
3.1 아키텍쳐 설계

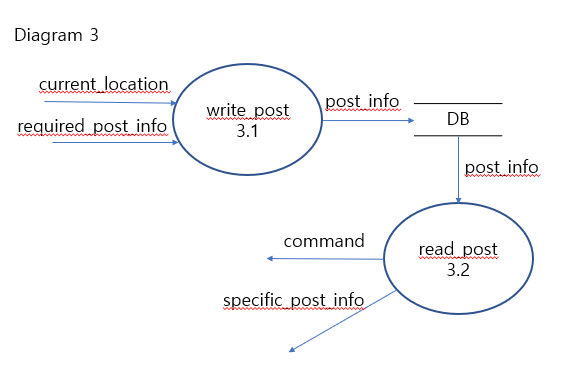
**3.1.1 요구사항 명세서로부터 하나의 자료흐름도 생성**

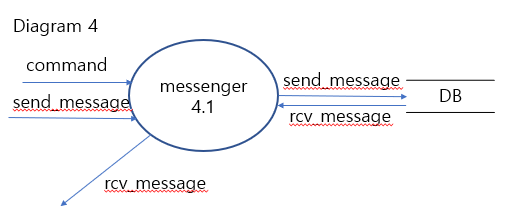








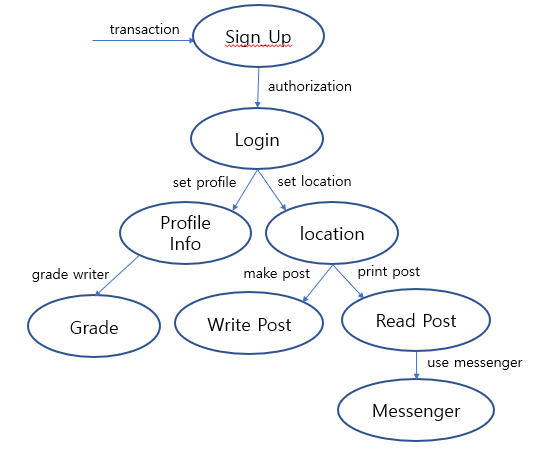




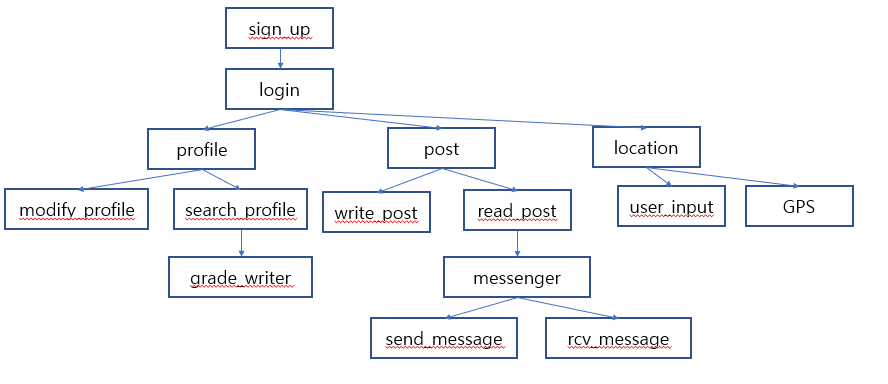
**3.1.2 자료흐름도가 트랜잭션흐름을 갖는지 결정**

입력(input)을 받아들여 가공처리한 후 새로운 결과물(output)을 만들어 내는 프로세스보다, 입력을 받아들여 입력값에 의해 결정을 내린 후 입력과 관계없는 결과물을 출력하는 경우가 많아 변환흐름보다는 트랜잭션 흐름을 이용한 설계가 적합하다고 판단했다.

**3.1.3 트랜잭션 중심과 동작 경로(action path)를 파악**

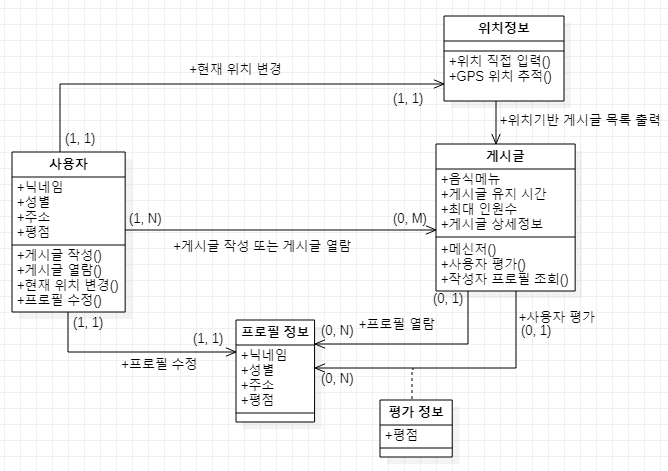


**3.1.4 트랜잭션에 기초한 프로그램 구조**



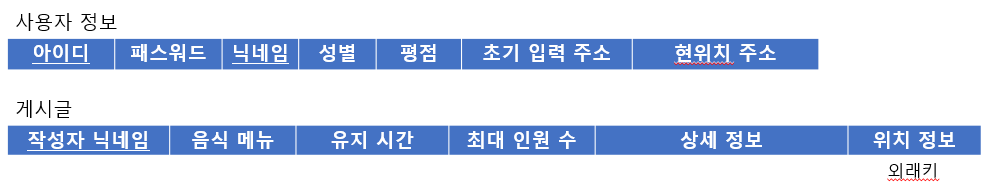
3.2 데이터 설계

**3.2.1 요구사항 분석 단계에서 도출한 클래스 다이어그램 작성**

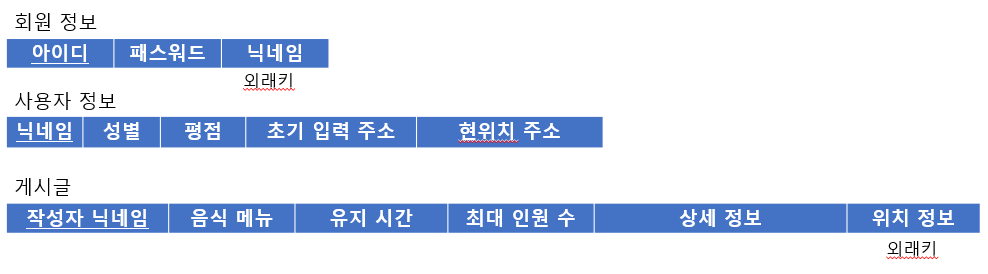


**3.2.2 엔티티와 테이블의 매핑**

초기 엔티티와 속성의 테이블 매핑



**3.2.3 정규화**



**3.2.4 데이터 베이스 설계**

테이블이(가) 표시된 사진

자동 생성된 설명

게시글의 위치정보는 현위치 주소가 존재하지 않을 경우 초기 입력 주소를 참조한다.

3.3 프로시저 설계

**3.3.1 현재위치 설정 후 게시글 출력 프로세스(Pseudo-code)**

텍스트이(가) 표시된 사진

자동 생성된 설명

텍스트이(가) 표시된 사진

자동 생성된 설명

**3.3.2 회원가입 및 로그인 프로세스(Pseudo-code)**

텍스트이(가) 표시된 사진

자동 생성된 설명

텍스트이(가) 표시된 사진

자동 생성된 설명

3.4 사용자 인터페이스 설계

|  |  |  |  |
| --- | --- | --- | --- |
|  | | | |
| 회원가입 | 로그인 | 프로필 설정 | 게시판 |
| Id, pw, pw2값을 받아 사용자 계정을 등록한다 | DB에 등록된 정보와 받아온 Id, pw값을 비교하여 사용자에게 권한을 부여한다. | 사용자 테이블을 업데이트한다. | 게시글들을 현재주소에  기반하여목록으로 출력  해준다. |

|  |  |  |  |
| --- | --- | --- | --- |
|  | | | |
| 게시글 작성 | 게시글 열람 | 현재 위치 재설정 | 사용자 평가 |
| 필수정보들을 입력하여 게시글을 작성한다. | 상세정보와프로필을 확인할  수 있다. | 사용자의 현재 위치  정보를 얻고, DB를 업데이트한다. | 주문 이용 이후에 작성자  에 대한 평가를 진행한다. |

|  |
| --- |
|  |
| 메신저  게시글에 접근한 인원들과 메신저가 가능하다. |

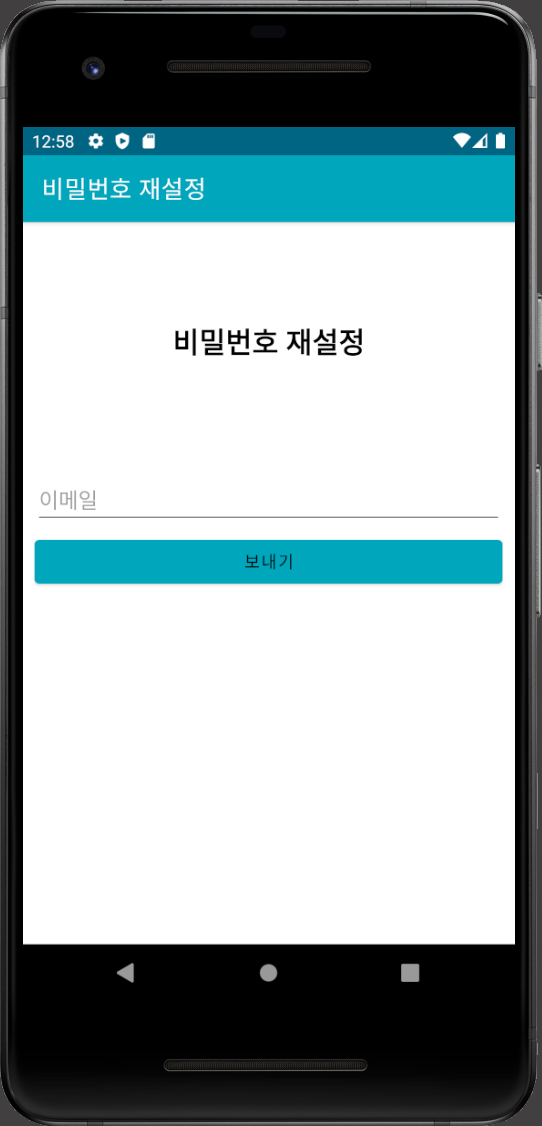
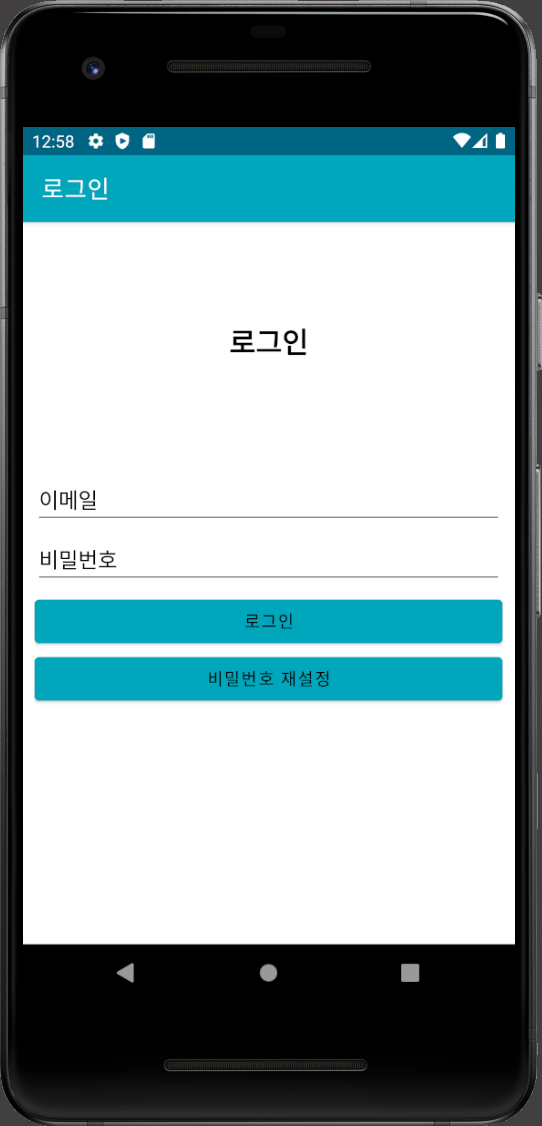
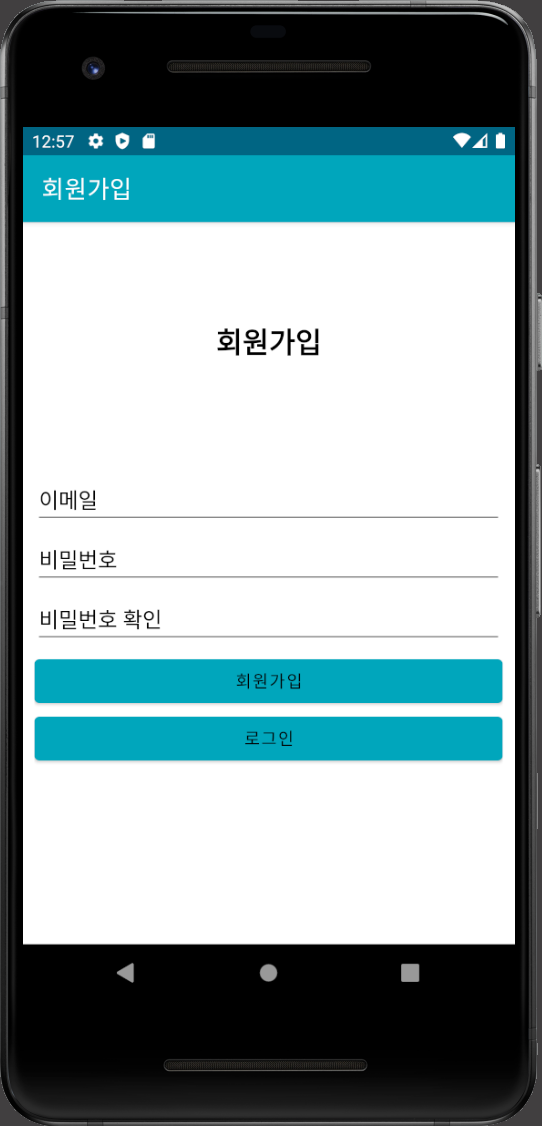
4. 구현 결과물

해당 과제 제출란에 제출된 zip 파일에 문서와 함께 소스코드를 첨부하였다. 프로그램은 안드로이드 스튜디오로 구현하였고, 실제 android 폰 또는 AVD emulator를 통해 실행가능하다.

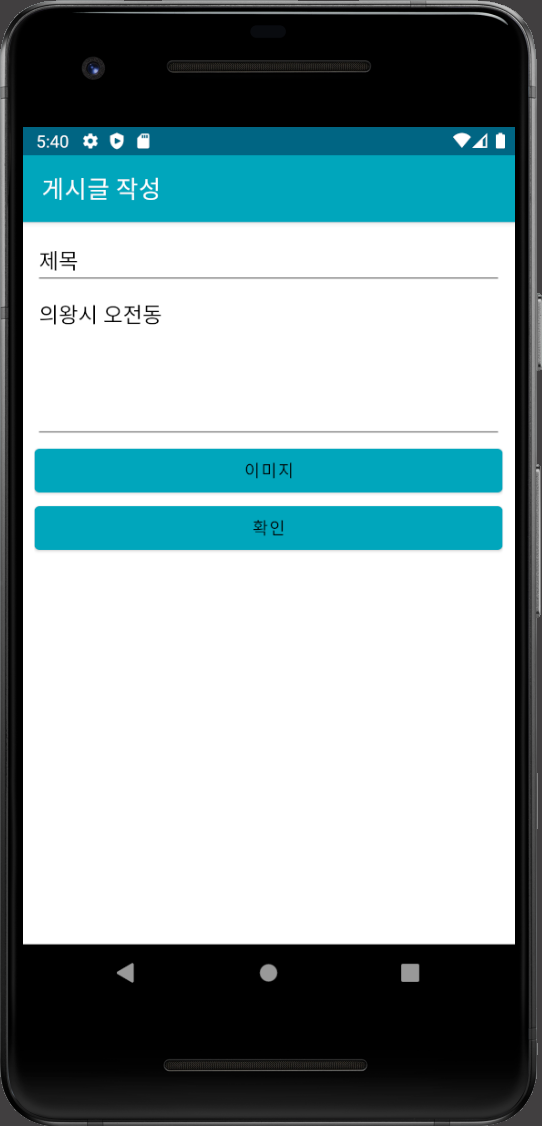
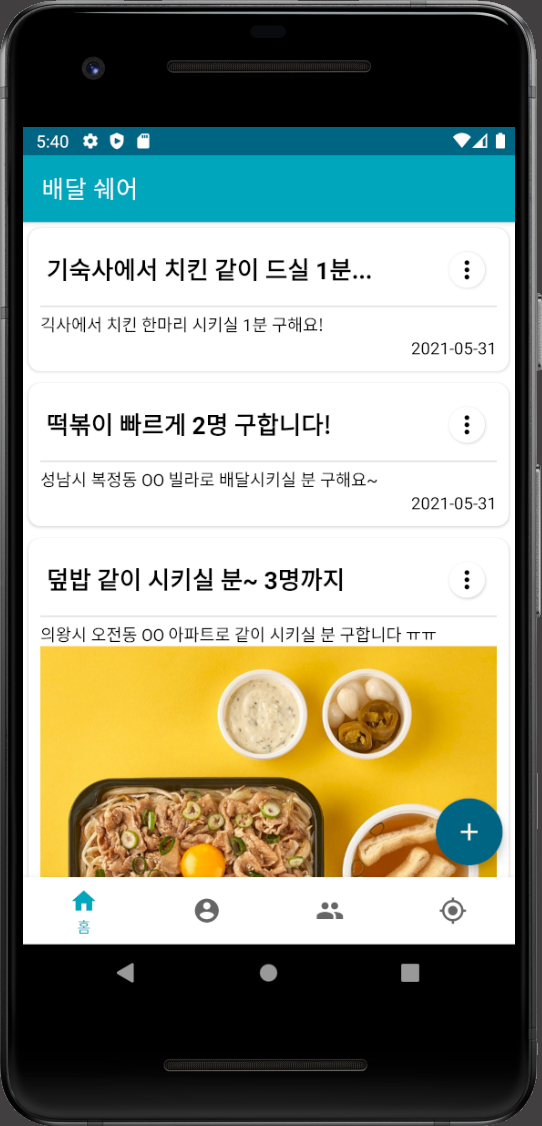
4.1 어플리케이션 기능 화면캡쳐

다음은 구현된 어플리케이션의 기능의 일부를 캡쳐한 모습이다.

|  |  |  |
| --- | --- | --- |
| 회원 가입 | 로그인 | 비밀번호 재설정 |



|  |  |  |
| --- | --- | --- |
| 게시판 열람(홈) | 게시글 작성 | 게시글 상세 정보 열람 |

텍스트, 전자기기, 스크린샷이(가) 표시된 사진

자동 생성된 설명

|  |  |  |
| --- | --- | --- |
| 채팅 | 현재 사용자 목록 | 주소 설정 |

텍스트이(가) 표시된 사진

자동 생성된 설명텍스트이(가) 표시된 사진

자동 생성된 설명텍스트, 스크린샷, 모니터이(가) 표시된 사진

자동 생성된 설명

|  |  |
| --- | --- |
| 지도에서 직접 선택 | GPS 자동 선택 |

지도이(가) 표시된 사진

자동 생성된 설명지도이(가) 표시된 사진

자동 생성된 설명

4.2 DB 사용 화면캡쳐

firebase의 Firestore DB, realtime DB에 데이터들을 저장하는식으로 구현하였다.

테이블이(가) 표시된 사진

자동 생성된 설명

텍스트이(가) 표시된 사진

자동 생성된 설명

텍스트이(가) 표시된 사진

자동 생성된 설명

5. 테스트 결과물

5.1 테스트 계획과 비교

**5.1.1 단위 테스트**

ㆍ회원가입

|  |  |  |
| --- | --- | --- |
| **테스트 항목** | **통과** | **비고** |
| ID와 PW가 데이터 베이스에 올바르게 저장되는지 | O |  |
| ID가 중복될경우 오류 메시지를 사용자에게 출력하는지 | O |  |
| PW1, PW2가 동일하지 않을경우 오류 메시지를 사용자에게 출력하는지 | O |  |

ㆍ사용자 위치 설정

|  |  |  |
| --- | --- | --- |
| 사용자가 입력한 위치가 위치 정보를 데이터 베이스에 올바르게 저장되는지 | O |  |
| 사용자가 입력 정보가 실제 위치정보와 일치하지 않을경우 사용자에게 오류 메시지를 출력하는지 | O |  |
| GPS를 사용하여 올바른 위치를 가져오는지 | O |  |
| GPS기능이 작동하지 않을 때 사용자가 직접 위치를 입력할 수 있는 화면을 출력하는지 | O |  |

ㆍ게시글 작성

|  |  |  |
| --- | --- | --- |
| 사용자가 필수 데이터를 입력하지 않았을 경우 오류 메시지를 출력하는지 | O |  |
| 사용자가 작성한 게시글이 데이터 베이스에 저장되는지 | O |  |
| 데이터 베이스에 저장된 게시글 데이터가 사용자별로 관리가 되는지 | O |  |

ㆍ게시글 열람

|  |  |  |
| --- | --- | --- |
| 데이터 베이스에서 해당 게시글의 정보를 정확하게 가져오는지 | O |  |
| 사용자의 위치를 기반으로 주변 사용자들이 작성한 게시글만 출력하는지 | X | 게시글에 대한 위치 정보는 저장 가능했지만, 사용자의 위치에 따라 게시판을 정렬 하는 기준과 필요한 값 호출을 구현하지 못하였다. |
| 게시글을 통해 작성자 프로필 정보를 올바르게 출력하는지 | X | 사용자 평가에 대한 기능을 구현하지 못하여 사용자 평가에 필요한 프로필 정보 접근 또한 구현하지 못하였다. |
| “메신저로 신청하기” 버튼을 눌렀을 때 메신저 기능이 작동하는지 | O |  |
| 특정 조건을 만족했을 때 “메신저로 신청하기” 버튼이 활성화 또는 비활성화 되는지 | X | 시간 관계상 게시글에 대한 정렬과 함께 접근 제한 조건까지는 구현하지 못하였다. |

ㆍ사용자 평가

|  |  |  |
| --- | --- | --- |
| 같이 주문한 사용자만 작성자를 평가할 수 있는지 | X | 시간 관계상 사용자 평가에 대한 기능은 구현하지 못하였다. |
| 사용자의 평점 정보가 데이터 베이스에 올바르게 업데이트 되는지 | X | “위와 동일” |
| 사용자가 레이팅바를 설정하지 않고 “평가 완료” 버튼을 눌렀을 때 오류 메시지를 출력하는지 | X | “위와 동일” |

**5.1.2 통합 테스트**

ㆍ액티비티 혹은 모듈

|  |  |  |
| --- | --- | --- |
| 로그인 후 사용자 정보가 없으면 프로필 설정 화면을 출력하는지 | O |  |
| 로그인 후 사용자 프로필이 이미 설정되어 있다면 게시판을 출력하는지 | O |  |
| 현재 위치설정을 하지 않았을 경우 프로필 설정시 입력한 주소로 설정되는지 | O |  |
| 사용자가 평가를 원하지 않는 경우 “뒤로가기” 버튼을 눌렀을 때 게시판 화면을 출력하는지 | X | 시간 관계상 사용자 평가에 대한 기능은 구현하지 못하였다. |
| 게시글 출력화면에서 “메신저로 신청하기” 버튼을 눌렀을 때 메신저 화면을 출력하는지 | O |  |
| 사용자가 메신저 화면에서 “뒤로가기” 버튼을 클릭했을 때 게시글 화면을 출력하는지 | O |  |
| 사용자가 게시글 화면에서 “뒤로가기”버튼을 클릭했을 때 게시판 화면을 출력하는지 | O |  |

**5.1.3 시스템 테스트, 인수 테스트**

짧은 기간 동안 진행된 프로젝트이고, 프로젝트 진행 단계에서 각 과정의 산출물 문서들에 초점을 두었기에 완벽하게 통합테스트까지 통과하는 프로그램 구현까지는 좀 버거워 타 시스템과 결합하는 시스템 테스트 혹은 고객에게 인수시키는 인수 테스트까지는 불가능했다. 따라 테스트 계획에서 이후의 과정들은 세우지 않았었고, 생략하였다.

5.2 미구현된 기능

**5.2.1 위치기반 게시글 정렬**

사용자의 위치에 따라 게시판을 정렬하는 기준 선정과 필요한 값 호출 실패

**5.2.2 게시글 별 채팅방 구분**

게시글에 대한 키값 호출 실패

**5.2.3 특정 조건 충족 시 게시글 잠금**

시간 관계상 미구현

**5.2.4 사용자 평가**

시간 관계상 미구현

**5.2.5 타 사용자 정보 열람**

시간 관계상 미구현

5.3 오류사항

**5.3.1 사용자 목록**

현재 사용자만큼 출력되는 것이 정상이지만, 처음 로딩 시에 현재 사용자들이 반복되어 스크롤뷰에 나타나는 경우가 존재한다.

6. 결론

6.1 결론

같이 배달할 사람을 구인하는 게시판의 게시글을 통해서 함께 주문이 가능하고 이를 통해 배달 최소 금액은 자연스럽게 맞춰지게 된다. 따라 배달 최소 금액에 대한 금전적인 불편함이 자영업자, 소비자 모두에게 감소하게 된다. 기존에 배달 최소 금액을 맞추기 위해 추가 주문 음식으로 인한 음식물 쓰레기 및 일회용품 사용 증가와 같은 문제 또한 자연스럽게 줄어들게 된다.

6.2 개발 각 단계별 보완점

**6.2.1 계획**

|  |
| --- |
| 문제점 |
| 최종적으로 사용자 평점 평가, 게시글 작성자 프로필 상세정보 확인, 위치 기반 게시물 접근 제어 기능을 구현하지 못했다. 짝 프로그래밍을 이용하여 생산성 증가와 검증까지 이득을 얻으며 19일 기간 안에 구현이 가능하다 판단했다. 하지만 3개의 기능을 구현하지 못했다. 즉 일정 계획 경험이 부족하여 계획 설정에 실패하였다. |
| 개선 방법 |
| WBS의 최저 수준에 해당하는 WP를 정확하게 작성한다. PERT/CPM을 사용하여, 각 WP를 기반으로 주 경로를 설정하고 현실적인 구현 계획을 완성한다. |

**6.2.2 요구사항 분석**

|  |
| --- |
| 문제점 |
| 초기 유스케이스 식별시 사용자 정보에 전화번호 목록이 누락되어 있다. 전화번호는 사용자의 개인정보이기 때문에 앱 사용의 편리성을 높이더라도 개인 정보 보호를 위해 제거되어야 한다 판단했었다. |
| 개선 방법 |
| 배달 앱의 특성상 사용자간 배달 합의시에 메신저를 이용할 뿐만 아니라 전화등의 매체를 이용할 수도 있다. 따라서 이를 개선하기 위해 사용자 필수 정보로 전화번호를 추가하였다. 또한 개인 정보 관리에 민감한 사용자를 위해 사용자가 원할 경우 전화번호를 비공개로 처리할 수 있도록 하는 방안을 제시할 수 있다. |

**6.2.3 설계**

|  |
| --- |
| 문제점 |
| 각 모듈의 응집도 및 결합도가 고려되지 않고 설계 되었다. 모듈의 기능적 관점에만 초점을 두고 설계를 하였고 구혈 결과 몇몇 모듈에서 우연적 응집도를 보이거나 결합도가 높은 모듈이 존재하게 되었다. |
| 개선 방법 |
| 기본 설계시 모듈을 블랙박스로 보고 모듈의 인터페이스를 명확하게 정의함으로서 응집도는 강하게, 결합도는 낮게 설계한다. |

**6.2.4 구현**

|  |
| --- |
| 문제점 |
| 모든 객체에 대한 주석이 일괄적인 형태로 작성되지 못했다. 또한 개발역사, 인터페이스와 같은 세부 내용들이 명확히 기술되지 못했다. |
| 개선 방법 |
| 개발역사, 인터페이스 주석이 명확하게 기술되지 못했다. 즉 모듈간의 종속성, 호출 관계가 명확하게 기술되지 못했다. 몇몇 객체는 하드코딩 되어 응집도가 낮거나 객체간의 결합도가 높아 호출관계가 명확하게 기술되기 어려웠다. 따라서 설계과정에서 모듈에 대한 명확한 설계를 진행한다. |

**6.2.5 테스트**

|  |
| --- |
| 문제점 |
| 프로젝트 진행 단계에서 각 과정의 산출물 문서들에 초점을 두었기에 완벽하게 통합테스트까지 통과하는 프로그램 구현까지는 좀 버거워 타 시스템과 결합하는 시스템 테스트 혹은 고객에게 인수시키는 인수 테스트까지는 불가능했다. |
| 개선 방법 |
| 테스트 단계 이전 프로젝트 라이프 사이클에 해당하는 각 단계들의 요구사항을 철저히 만족시키며 구현을 진행한 뒤 통합 테스트 이상의 테스트를 진행한다. |

6.3 느낀점

지금까지 진행했던 토이 프로젝트에서는 수업에서 배운 여러 단계를 거치지 않고, 바로 구현에 뛰어들었기에 코딩 과정이 가장 어렵다고 생각했었다. 하지만 이번에 소프트웨어 품질을 향상 시키기 위한 각 단계들을 거치며 프로젝트를 진행하면서, 코딩은 단순히 기계적인 작업일 뿐이며, 앞 단계인 계획, 요구사항 분석, 설계 과정이 훨씬 더 중요하다는 것을 알게 되었다. 또한 각 단계에서 제대로 된 태스크 진행이 되지 못할 경우 다음 단계에 큰 영향을 주어, 각 단계들이 뒤로 갈수록 문제를 수정하기 어려워 진다는 것을 느꼈다. 따라서 공식적인 검토, 코드 검사, 시험과 같은 V&V 기법들을 사용하여 소프트웨어 품질을 만족시키기 위한 과정의 필요성을 느낄 수 있는 계기가 되었다.

7. 참고문헌

7.1 기술동향 참고문헌

안수형.(2020).국내규제동향.배달을 둘러싼 소비자보호 입법 동향과 전망 https://blog.naver.com

/koreareg/221950125179

김수영.(2020).노컷뉴스.코로나 장기화에 배달앱, 쇼핑앱 이용량껑충https://www.nocutnews.c

o.kr/news/5327314

미상.(2020).매일경제.배달앱 이용자 최소주문금액 맞추려 필요 이상주문 .https://m.mk.co.kr/ne

ws/business/view/2020/11/1158854/

미상.(2021).NEWSIS20.코로나에 배달음식시장 호황.https://newsis.com/view/?id=NISX202102

22\_0001347095

이동우.(2021).머니투데이.8천원 돈가스 배달비가 7천원.https://news.mt.co.kr/mtview.php?no

=2021042316173412211

강신우.(2019).이데일리.배달앱 뜨니 ‘악성쓰레기’ 늘었다.https://www.edaily.co.kr/news/read?n

ewsId=01193926622553536&mediaCodeNo=257

7.2 개발 과정 참고문헌

과학기술정보통신부 ·정보통신산업진흥원. “소프트웨어사업 요구사항 분석·적용 가이드.” (2021) 105~133

김정길,and 정지문. "안드로이드 기반 OpenAPI를 이용한 SNS 연동 지역정보 서비스를 위한 모바일 시스템 설계 및 구현." 디지털융복합연구 9.2 (2011): 131-140.

하울. (Udemy.com 베스트셀러 강사 하울의) Firebase로 안드로이드 SNS 앱 만들기. 서울: 비제이퍼블릭, 2018.

FireBase, 작가미상, 문서, https://firebase.google.com/docs

Android Studio, 작가미상, 문서, https://developer.android.com/studio/intro?hl=ko

Google Maps Platform, 작가미상, 문서, https://developers.google.com/maps/documentation/android-sdk/overview?hl=ko