

1차 발표 보고서

과 목 명	소프트웨어공학개론_02	
담 당 교 수	이 호 정	
학 과	컴퓨터공학과	
팀 명	트래블	
팀 원	조광호	2018112072
	이희준	2018112070
	이대엽	2019113577
	조영승	2019112074

목 차

0. 팀원.....	3
1. 문제 개요.....	3
1.1 문제 배경.....	3
1.2 기존 문제.....	3
1.3 해결방안(개선방안).....	3
2. 문제 정의.....	4
3. 구현 범위 정의.....	5
4. 유스케이스 시나리오.....	6
5. 유스케이스 다이어그램.....	7
6. 유스케이스 명세.....	8
7. 회의 내용(backlog).....	12

0. 팀원

본 팀(트래블)의 인원 구성은 다음과 같다.

이름	학번	역할
조광호	2018112072	팀장, 백 엔드
이희준	2018112070	백 엔드, 프론트 엔드
이대엽	2019113577	백 엔드
조영승	2019112074	프론트 엔드

1. 문제 개요

1.1 문제 배경

최근 코로나 바이러스 관련 여행 제약들이 해제되면서 여행에 대한 관심이 증가하고 있다. 버스를 타고 관광지를 구경하는 버스 관광도 이중 하나로, 다른 관광 방식에 비해 가격이 저렴하고, 비교적 짧은 시간 내에 유명한 관광지를 구경할 수 있다는 점에 있어서 외국인이나 중, 노년층의 사람들에게 인기가 많다. 따라서 버스 관광을 예약하는 시스템을 구현하고자 한다.

1.2 기존 문제

투어 버스로 2층 버스를 이용하는 경우 한번 운행에 탑승하는 사람이 대략 40~60명으로 추정되는데, 이들에 대한 탑승 수속은 버스 기사에 의한 인력으로 수행되므로 승객들의 대기 시간이 늘어질 수도 있는 아쉬움이 있다. 또한 기존에는 온라인에서 예약 후 현장에서 티켓(종이표)를 발급하는 방식을 사용하는데, 이 경우 현장 티켓 박스에서 예약자를 인증 후 종이표를 수령하는 프로세스가 필요하며 여행 도중 해당 티켓을 분실할 수 있는 위험이 있다.

1.3 해결방안(개선방안)

이를 해결하기 위해 온라인으로 표를 구입하여 사용자에게 QR 이나 바코드 형태의 표(티켓)를 제공하고, 이를 이용하여 전자적으로 탑승 수속을 할 수 있도록 개선하고자 한다. 이를 통해 현장에서 종이표를 수령하는 프로세스를 생략할 수 있고, 표를 분실할 위험성이 없어진다.

2. 문제 정의

현재 프로젝트는 버스 관광 예약을 위한 사용자-웹사이트 간 상호작용 구현을 목표로 하고 있다. 이때 이전 토의 과정까지는 전체 시스템을 바라보는 각 팀원들의 관점이나 생각에 대해 구체적으로 논의하지 않아 의견 교환 중 일부 충돌이 발생했다. 이러한 충돌을 방지함과 동시에 차후 구현 단계에서 팀원 간 혼란을 빚지 않기 위해 문제 상황의 일부 사항들을 명확하게 정의하기로 했으며, 정해진 사항은 다음과 같다.

표 주문

- 각 버스는 탑승할 수 있는 최대 인원수가 존재하며, 좌석은 선착순으로 운영한다.
- 사용자는 최대 인원수 이상의 표를 주문할 수 없다.
- 승객의 타입은 성인과 어린이로 나뉜다.
- 사용자는 버스 관광의 종류, 사용자 타입 및 인원수를 선택하여 주문을 진행할 수 있다.
 - 주문에는 사용자 식별을 위한 정보(주문자 명, 전화번호)가 요구된다.
 - 주문의 결과로 주문 코드 및 QR코드가 발행된다.

구매 확인

- 사용자는 주문자 명 / 전화번호를 이용하여 사용자를 인증할 수 있다.
 - 사용자 인증은 구매 확인 및 환불 절차에 선행된다.
- 구매 확인 단계에서 사용자는 자신이 구매한 표 목록을 열람할 수 있다.

환불

- 환불은 구매 확인 단계에서 구매한 표를 선택하는 방식으로 수행할 수 있다.
- 환불 정책은 다음과 같다.
 - ① 탑승일 10일 전부터 전일까지는 100%을 환불한다.
 - ② 탑승일 전일부터 당일 출발 시점 이전까지는 50%을 환불한다.
 - ③ 출발 시점 이후에는 환불이 불가능하다.

표

- 표에는 주문코드 및 QR 코드가 존재한다.
- 주문 코드는 사용자의 주문 식별 및 환불 절차에 사용된다.
- QR 코드는 사용자의 탑승 수속에 사용된다.
- 하나의 표는 다수 인원의 탑승 수속을 동시에 처리할 수 있다.

3. 구현 범위 정의

이전 토의 과정에서는 예약, 구매 확인, 환불에 대한 시나리오를 구성하면서 해당 범위를 임시적인 구현 범위로 지정했으며, 이러한 시나리오들은 사용자 시점에 국한되었다. 이때 추가적인 토의 과정에서 사용자뿐만 아니라 관리자 입장의 기능 구현 필요성이 제기되었고, 이에 대해 논의했다.

사용자와 관리자를 각각 액터로 두었을 때 가능한 시나리오 목록은 다음과 같으며, 논의 내용의 깊이에 따라 구체적으로 심화될 수 있다.

사용자

- 관광 예약
 - ① 관광 상품 및 시간 선택
 - ② 사용자 타입(성인/어린이) 선택
- 구매 확인
- 예약 환불
 - ① 전체 상품 환불
 - ② 일부 상품 환불

관리자

- 관리자 인증: 로그인을 통해 관리자를 인증한다.
- 관광 상품 관리
 - ① 관광 상품 추가: 관광 상품을 추가한다.
 - ② 관광 정보 수정: 현재 존재하는 관광 상품의 정보를 수정한다.
 - ③ 관광 상품 삭제: 현재 존재하는 관광 상품을 삭제한다.
- 구매자 및 예약 관리
 - ① 상품 별 정보 열람
: 각 상품 별 예약 상태, 차량 상태, 기사 정보 등의 정보를 열람한다.
 - ② 버스 관광 상품 취소
: 자연재해 등 사정이 있어 관광을 운영할 수 없는 경우 해당 상품을 취소한다.
 - ③ 웹사이트 관리
 - 1. 배너 관리
 - 2. 게시판 관리(공지)
 - I. 공지 작성
 - II. 공지 수정
 - III. 공지 삭제

위 나열된 시나리오에는 버스기사가 액터인 경우나 특수한 상황에 대한 내용이 생략되어 있다. 이처럼 버스 관광 예약 시스템은 구현하고자 하는 범위에 따라 난이도가 크게 달라질 수 있다. 따라서 본 팀은 관리자 기능 구현에 대한 논의를 진행한 후, 관리자 기능 추가로 인해 발생할 수 있는 프로젝트의 복잡성 및 난이도 증가를 피하기 위하여 언급한 영역 전체를 구현하는 대신 초기 목표를 따라 액터가 사용자인 시나리오로 한정하여 구현범위를 구성하기로 결정했다.

4. 유스케이스 시나리오

기존 시나리오에서 변경된 사항은 적색 강조로 나타냈다.

4.1 예약 시나리오

사용자	시스템
1. 사용자가 버스관광 예매 페이지에 접속 2. 버스관광 종류 선택 4. 관광 코스, 인원, 날짜 선택 6. 사용자 정보(핸드폰 번호, 이메일 등)입력 7. 결제	3. 선택한 버스관광에 대한 정보(관광 코스, 신청가능 인원, 가격, 날짜 등)를 출력 5. 선택한 내용에 따라 가격 계산하여 출력 8. 결제 확인 9. 버스 좌석 개수 반영 10. 해당 구매에 대한 고유한 구매코드 생성 11. 생성한 코드 사용자에게 전송(이메일 등)

4.2 구매내역확인 시나리오

사용자	시스템
1. 사용자가 버스관광 예매 페이지에 접속 2. 구매 확인 페이지로 이동 3. 예약자 정보 입력	4. 예약자 정보 와 대응되는 구매 정보 찾아서 출력

4.3 환불(취소) 시나리오

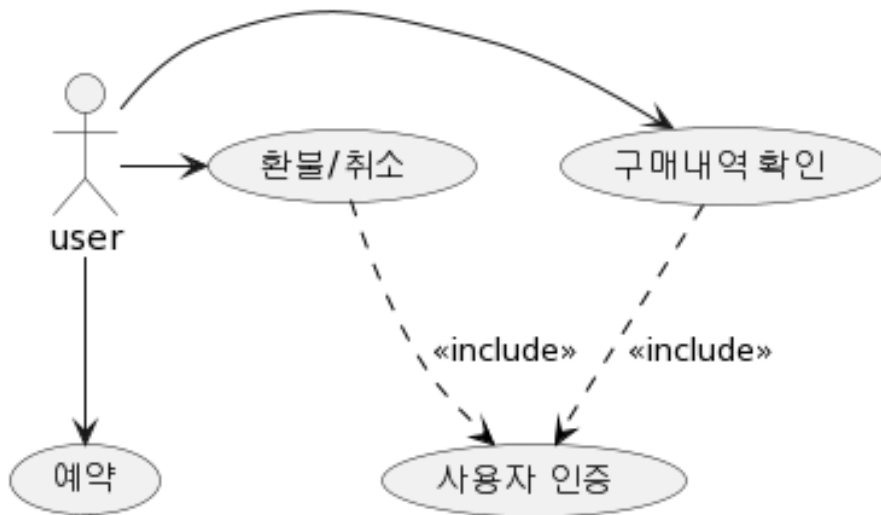
사용자	시스템
1. 사용자가 버스관광 예매 페이지에 접속 2. 구매 확인 페이지로 이동 3. 예약자 정보 입력 5. 취소하고 싶은 상품 선택 7. 전체 환불 / 부분 환불 선택 9. 환불(취소)확인 클릭	4. 예약자 정보 와 대응되는 구매 정보 찾아서 출력 6. 선택된 상품에 대한 취소 페이지 디스플레이 8. 사용자가 선택한 환불 인원에 따라 환불(취소)되는 가격 출력(적절한 환불 정책에 기반) 10. 환불(취소) -> 버스 좌석에 반영

5. 유스케이스 다이어그램

현재 프로젝트에서의 구현 범위는 다음과 같이 정한다.

- 버스 관광 예약
- 구매 확인
- 환불(취소)
- 구매자 인증

이에 따른 유스케이스 다이어그램은 다음과 같다.



사용자 인증의 경우 구매내역확인과 환불 절차에 선행되므로 기존 시나리오에서 인증 부분을 분리하여 개별적인 유스케이스로 나타내고, 이를 include 관계로 나타냈다. 나머지 유스케이스들은 기존 시나리오에서 언급된 유저측 상호작용과 대응된다.

6. 유스케이스 명세

위에서 작성한 유스케이스 사나리오와 다이어그램을 바탕으로 유스케이스 명세를 작성하였다. 존재하는 명세는 다음과 같다.

- 버스 상품 예약
- 구매 내역 조회
- 주문 환불
- 사용자 인증

시스템 제목	버스 관광 예약	
유스케이스 이름	버스 상품 예약	
액터	관광객	
시작 조건	상시 가능	
기본 흐름	사용자	시스템
	1. 사용자가 원하는 상품과 좌석 수를 선택 3. 자신의 휴대폰 번호, 이름 등의 사용자 정보를 입력 4. 결제 수단을 선택 6. 결제 화면에서 결제 정보를 입력하고 결제를 완료	2. 남은 좌석 수를 확인하여 예약 진행 5. 사용자가 선택한 결제 수단의 결제 화면을 디스플레이 7. 결제 완료 여부를 확인하고 결제 완료 화면을 디스플레이
대안 흐름	1A, 3A, 4A, 6A	
	1. 사용자가 뒤로 버튼을 클릭	2. 바로 전단계로 복귀
	2A	
		1. 사용자가 입력한 좌석 수가 남은 좌석 수보다 많을 시 안내 메시지를 띄우고 좌석 수를 가능한 최대치 상태로 되돌림
	7A	
		1. 잔액 부족/통신 오류 등 결제 미완료 시 안내 메시지를 띄우고 결제 정보 입력 단계로 복귀
종료 조건	결제 완료 후, 예약 완료 내역을 디스플레이	

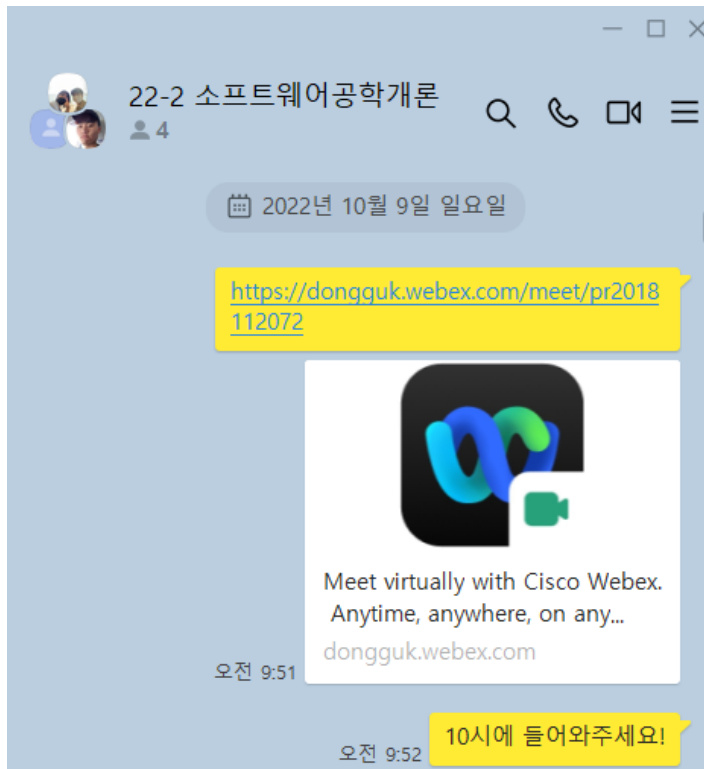
시스템 제목	버스 관광 예약	
유스케이스 이름	구매 내역 조회	
액터	관광객	
시작 조건	관광객이 예약하고 구매 내역을 조회하거나 환불 및 취소를 하려고 함	
기본 흐름	사용자	시스템
	1. 구매 내역 조회를 선택 2. 조회하고자 하는 구매 내역의 예약자의 개인 정보를 입력	3. 사용자가 입력한 예약자의 개인정보와 일치하는 구매 내역이 있는지 찾음 4. 찾은 구매내역을 화면에 디스플레이
대안 흐름	3A	
	3. 오류 메시지를 확인하고 다시 시도하거나 고객센터로 전화	1. 사용자가 입력한 정보와 일치하는 구매내역이 없음 2. 다시 시도해보고 안되면 오류 메시지를 디스플레이
종료 조건	구매 내역이 화면이 나온다.	

시스템 제목	버스 관광 예약	
유스케이스 이름	주문 환불	
액터	관광객	
시작 조건	사용자가 구매 내역을 조회한다.	
기본 흐름	사용자	시스템
	1. 원하는 상품의 '환불' 을 선택 3. 원하는 환불 방식을 선택 3.1 주문 취소를 선택 3.2 일부 환불을 선택 3.2.1 환불하고 싶은 표의 타입과 개수 선택	2. 대응되는 상품의 정보를 디스플레이 3.1.1 사용자가 요청한 환불이 유효한지 검사 3.1.2 전체 상품을 환불 3.2.2 사용자가 요청한 환불이 유효한지 검사 3.2.3 일부 표를 환불 4. 환불 결과를 디스플레이
대안 흐름	3.1.1A, 3.2.2A	
	2. 에러 메시지를 보고 웹사이트로 돌아가거나 고객센터에 연락	1. 요청한 환불이 유효하지 않다면 에러 메시지를 출력
	3.2A	
		1. 사용자가 요청한 환불 표가 전체 상품에 해당하는 경우 전체 상품을 환불
종료 조건	사용자가 요청한 환불이 완료된다.	

시스템 제목	버스 관광 예약	
유스케이스 이름	사용자 인증	
액터	-	
시작 조건	관광객이 사용자 인증이 필요한 서비스를 진행한다.	
기본 흐름	사용자	시스템
	3. 사용자 인증 화면에서 요구하는 예약자의 정보를 입력, 제출	1. 사용자 인증 요구를 받음 2. 사용자에게 사용자 인증에 필요한 정보를 요구하는 화면을 출력 4. 사용자가 입력한 정보로 해당 사용자의 구매내역을 찾음 5. 찾은 구매내역을 사용자 인증 요구를 요청한 서비스로 보냄
대안 흐름		4A 1. 찾지 못했다면 해당 사용자의 구매기록은 없다는 것을 사용자 인증 요구를 요청한 서비스로 보냄
종료 조건	사용자 인증 요구를 요청한 서비스로 사용자 인증 결과를 전달	

7. 회의 내용(backlog)

회의는 Webex를 이용해 비대면으로 진행했으며 채팅 내역은 별도의 text파일로 저장해 두었다. 1차 발표를 위한 회의는 10월 9일에 진행했고 전체 회의 내역은 text파일로 첨부하였다. 아래는 회의의 일부 채팅 내역이다.



회의 방식(Webex에서 비대면으로 진행)

이대엽(*****3577) 님이 모두에게: 오전 10:17
흠 어제 그거 하면서 생각났는데 저희도 관광회사직원 쪽 액터도 생각해야될까요?
이대엽(*****3577) 님이 모두에게: 오전 10:17
뭔가 그쪽에서의 유스케이스가 있을까요?
조광호(*****2072) 님이 모두에게: 오전 10:17
음 새로운 관광상품 추가가 있을 수 있겠네요
조광호(*****2072) 님이 모두에게: 오전 10:17
새로운 버스관광 추가한다던지
조영승(*****2074) 님이 모두에게: 오전 10:17
관광 정도 수정?
이희준(*****2070) 님이 모두에게: 오전 10:17
관광 웹사이트를 만들고 싶은게 아니면 굳이 고려할 요소는 아닌거 같습니다
조영승(*****2074) 님이 모두에게: 오전 10:17
정보
이희준(*****2070) 님이 모두에게: 오전 10:18
회사 직원쪽 액터의 존재는 유저의 관광 예약만 구현할건지, 아니면 관광 웹사이트를 구현할건지에 따른 문제인 것 같습니다.
이대엽(*****3577) 님이 모두에게: 오전 10:18
음.. 근데 관광 예약이라는게 유저만 하는게 아니라
이대엽(*****3577) 님이 모두에게: 오전 10:19
예약을 하면 그걸 받는 쪽도 당연히 존재해야된다고 생각합니다 저는
조영승(*****2074) 님이 모두에게: 오전 10:19
근데 저희가 예약 정보를 확인?
이희준(*****2070) 님이 모두에게: 오전 10:19
그러면 대충 어떤 내용을 생각해보셨나요?

시스템의 문제 범위 설정에 대한 회의 내용의 일부

조광호(*****2072)님이 모두에게: 오전 10:09
저희가 문제기술서를 작성하면서 이야기했을 때 나온 유스케이스가
조광호(*****2072)님이 모두에게: 오전 10:09
예약, 구매확인, 환불/취소 었나요?
이대엽(*****3577)님이 모두에게: 오전 10:10
음 메뉴 추가 수정 삭제 까지 인것 같습니다
이대엽(*****3577)님이 모두에게: 오전 10:10
아 이걸 다른 거였지
이대엽(*****3577)님이 모두에게: 오전 10:10
ㅋㅋㅋㅋㅋㅋ
조광호(*****2072)님이 모두에게: 오전 10:10
ㅋㅋ
조광호(*****2072)님이 모두에게: 오전 10:11
그럼 이 세 유스케이스에 대해 다이어그램을 그리면 될거 같은데
이대엽(*****3577)님이 모두에게: 오전 10:11
환불/취소나 구매 확인은 인증절차가 앞단에 있어야되지 않을까요??
이대엽(*****3577)님이 모두에게: 오전 10:12
예약부터 로그인을 하게 한다던지
조광호(*****2072)님이 모두에게: 오전 10:12
지금 여기서 그리기는 좀 그럴거 같고 액터랑 각 유스케이스별 관계정도 이야기할까요?
조광호(*****2072)님이 모두에게: 오전 10:12
아 네 인증(로그인)부분은 따로 분리해서 include관계로 연결하느게 좋을까요?
조광호(*****2072)님이 모두에게: 오전 10:12
아 근데 저희
조광호(*****2072)님이 모두에게: 오전 10:13
로그인 기능이 있었나요

유스케이스 추출과 구체화에 대한 회의 내용의 일부

이희준(*****2070)님이 모두에게: 오전 10:57
근데 버스 예약에 사람 제한이 있나요 저희
조광호(*****2072)님이 모두에게: 오전 10:57
버스가 수용할 인원을 정해놓는게 좋을것 같습니다
이대엽(*****3577)님이 모두에게: 오전 10:57
네넵 그럴것 같습니다
조광호(*****2072)님이 모두에게: 오전 10:57
인원을 초과하면 예약이 안되도록
이희준(*****2070)님이 모두에게: 오전 10:57
시스템에 대해 어느 정도 개념을 통일 하고 가야 할 것 같아요
조영승(*****2074)님이 모두에게: 오전 10:58
2층 버스로?
조광호(*****2072)님이 모두에게: 오전 10:59
시스템의 개념과 동작 방식을 2~3문장 정도로
조광호(*****2072)님이 모두에게: 오전 10:59
정리하고 갈까요?
이대엽(*****3577)님이 모두에게: 오전 11:00
넵 그럴까요?
조광호(*****2072)님이 모두에게: 오전 11:00
저번에 quiz2의 문제 설명에도 비슷한게 있었던거 같은데
조광호(*****2072)님이 모두에게: 오전 11:00
<문제 정의>
"우리 학교 학생식당(상록원)에 비치된 식권 판매 시스템과 같은 것을 개발하려 한다. 구매자는 메뉴와 개수를
조광호(*****2072)님이 모두에게: 오전 11:00
이런 식으로 하면 될것 같아요
이대엽(*****3577)님이 모두에게: 오전 11:00
네넵

문제 정의에 대한 회의 내용의 일부

	Yet	Doing	Done	Test
주제(범위) 결정			○	
사용 언어 결정			○	
사용 툴 결정 (프론트, 백)		○		
프론트 백 연동	○			
Database 연동	○			
예약 기능 구현	○			
사용자인증 기능 구현	○			
구매확인 기능 구현	○			
환불/취소 기능 구현	○			