소프트웨어공학

프로젝트 SRS

셰어택시-택시 동승 어플리케이션

NULL 사로잡겠어 리턴즈

팀장 컴퓨터학부 이민석 2016117244

팀원 컴퓨터학부 곽준창 2012105001

컴퓨터학부 우정우 2016111829

컴퓨터학부 차수진 2016111813

일어일문학과 황원규 2011000934

|  |
| --- |
| **1. Introduction** |
| 1. Purpose  🡺본 문서는 소프트웨어 공학 과목의 ShareTaxi 프로젝트에 대한 요구사항을 명시하고 있다. 소프트웨어 개발과 개발자-사용자 간 이해의 용이함을 위해 구체적으로 본 소프트웨어의 기능/비기능적 요구사항, 인터페이스와 여러 모델 등에 대한 상세한 설명을 한다.  2. Product scope  🡺ShareTaxi는 택시 동승자를 찾아주는 소프트웨어이다. 본 소프트웨어에서는 탑승 정보 설정, 동승자 검색, 방 만들기, 방 입장, 채팅방 등의 여러 가지 기능을 제공한다. Android 기반 스마트폰에서만 동작하며 iOS 기반 스마트폰, PC 등에서는 동작하지 않는다. 본 소프트웨어를 사용할 때의 장점은 버스를 타기 불편한 출퇴근이나 등, 하교 시간대에 사람들은 택시를 많이 이용하지만 부족한 택시 수와 금전적, 시간적인 문제로 인해 불편함을 호소하는 사람들을 위해 편의를 제공해 줄 수 있다는 것이다.  3. User  🡺본 시스템은 대중교통을 이용하는 모든 사람들을 대상으로 한다.  4.Glossary of terms  - 없음  5.Reference  i) 참조문헌:  없음  ii) 참조사이트:  없음 |
| **2. Overall description** |
| 1. Context model    2. Process model (activity diagram)    3. Overall architecture  4. Use cases    **1)**   |  |  | | --- | --- | | name: ID | Setting the information: UC01 | | actors | User, Map | | description | 약속시간을 입력하고 지도에서 출발지와 도착지를 선택한다. [검색] 버튼을 누르면 입력된 정보를 서버에 전송한다. | | data | 출발지, 도착지, 출발시간, 사용자의 닉네임 | | stimulus | [검색] 버튼 | | response | Lobby UI를 보여줌. | | comments | 사용자는 고의로 비상식적인 정보를 입력하지 않는다. |   **2)**   |  |  | | --- | --- | | name: ID | Joining room: UC02 | | actors | User | | description | Lobby UI에서 방을 클릭하면 해당 방으로 입장한다. | | data | 방 목록 | | stimulus | 방 클릭 | | response | 방 UI, #UC04 | | comments | 방 인원이 전부 차있으면 입장이 불가능하게 한다. |   **3)**   |  |  | | --- | --- | | name: ID | Creating room: UC03 | | actors | User | | description | [방 만들기] 버튼을 누르면 이전에 입력 받은 정보(출발지, 도착지, 약속시간, 닉네임)를 가져와서 방을 만들고 서버에 있는 방 리스트를 갱신한다. | | data | 출발지, 도착지, 약속시간, 방장의 닉네임 | | stimulus | [방 만들기] 버튼 | | response | DB에 있는 방 리스트에 방을 추가 | | comments |  |       4)   |  |  | | --- | --- | | name: ID | Chatting: UC04 | | actors | User | | description | 유저는 다른 유저와 채팅하고, 모두가 [출발하기] 버튼을 누른 상태이면 다음(?)으로 넘어간다. | | data | 유저가 입력한 메시지 | | stimulus | [전송], [출발하기] 버튼 | | response | 유저가 입력한 메시지, Rating UI | | comments | 사용자는 자신이 소속된 방을 제외한 다른 방의 채팅 메시지들은 볼 수 없어야 한다.  입장 전 대화는 볼 수 없다.  메시지는 손실 없이 전달되어야 한다.  같은 방에 있는 모든 유저가 [출발하기] 버튼을 누른 상태여야만 Rating UI로 넘어간다.  유저는 다른 유저들의 인기도를 볼 수 있다. |     **5)**   |  |  | | --- | --- | | name: ID | Removing room: UC05 | | actors | User | | description | 방에 마지막으로 남은 유저가 [나가기] 버튼을 누를 경우 방이 삭제된다. | | data | 방에 남아 있는 유저 수 | | stimulus | [나가기] 버튼 | | response | Lobby UI를 보여줌. | | comments |  |   **6)**   |  |  | | --- | --- | | name: ID | Rating users: UC06 | | actors | User | | description | 유저끼리 상호 평가한다. | | data | 같은 방에 존재하는 유저들의 정보 | | stimulus | [좋아요], [싫어요], [그저 그래요], [평가 완료] 버튼 | | response | Setting UI를 보여줌, 유저 정보 갱신 | | comments | 평가 3가지는 복수 선택 불가능하다. | |
| **3. Environment** |
| 1. Operating environment  - Android(OS), Android 기반 스마트폰  2. Development environment  - HW: PC 및 Laptop 5대, Android 기반 스마트폰 2~3대  - SW: Android(OS), Android Studio, Git, MS Office, Atom, MySQL |
| **4. External interface requirements** |
| 1. User interface  -  2. Hardware interface  - Wi-Fi, 데이터 네트워크  3. Software interface  - Google Map API  4. Communication interface  - 채팅은 소켓통신 기반이다. |
| **5. Functional system requirements** |
| |  |  | | --- | --- | | name: ID | Selecting position: FR01 | | description | Map을 이용해 출발지와 도착지를 선택한다. | | inputs | Map | | outputs | 출발지/도착지의 정확한 주소 정보 | | rationale | 없음 | | action | 주소를 검색하거나 지도에서 출발지/도착지를 지정한다.  만약 출발지/도착지가 Map에 존재하는 위치라면, 출발지/도착지의 주소 정보를 반환한다. | | references | **- #UC01** |  |  |  | | --- | --- | | name: ID | Transferring information: FR02 | | description | 출발지/도착지 정보를 서버에 전송해서 처리해야 한다. 데이터는 신뢰성을 가져야 한다. | | inputs | 출발지/도착지 정보, 시간 정보 | | outputs | 성공 시 lobby UI, 실패 시 팝업 창 | | rationale | 없음 | | action | 출발지/도착지 정보, 시간 정보를 객체화한다.  객체를 서버에 전송한다. | | references | **- #UC01** |  |  |  | | --- | --- | | name: ID | Searching a suitable room: FR03 | | description | 출발지/도착지 반경 300m 이내에 조건에 맞는 방들을 보여줘야 한다. | | inputs | #FR02의 정보 객체 | | outputs | 방 목록 | | rationale | 없음 | | action | 선택한 출발지에서 반경 300m 이내에 존재하는 방들을 찾는다.  출발지에서 찾은 방들 중 도착지 반경 300m 이내 존재하는 방들을 찾는다.  조건에 맞는 방들을 보여준다.  새로고침 버튼을 누르면 방 정보가 갱신된다. | | references | **- #FR02** |  |  |  | | --- | --- | | name: ID | Creating room: FR04 | | description | 방의 생성에 관한 요구사항 | | inputs | 출발지/도착지/출발시간 정보, 유저 정보 | | outputs | 만들어진 방의 정보 | | rationale | 없음 | | action | 방을 만들 때 방 정보는 방을 새로 만드는 유저가 가지고 있는 출발지/도착지/출발시간으로 이루어진다.  방 구성원은 최대 4명까지 참여할 수 있다.  방은 채팅 형식으로 구성된다. | | references | **- #UC03, #UC04** |  |  |  | | --- | --- | | name: ID | Maintaining room: FR05 | | description | 방의 유지에 관한 요구사항 | | inputs | 방 정보 | | outputs | 갱신된 방 정보 | | rationale | 없음 | | action | 같은 방에 있는 모든 유저가 [출발하기] 버튼을 누른 상태가 되면 #UC06으로 진입한다.  유저가 입장하면 해당 방의 유저 목록에 입장한 유저 정보를 추가한다.  유저가 퇴장하면 해당 방의 유저 목록에서 퇴장한 유저 정보를 삭제한다. 단, 퇴장한 유저가 마지막 유저일 경우 #FR06을 호출한다. | | references | **- #UC03, #UC04, #FR06** |  |  |  | | --- | --- | | name: ID | Removing room: FR06 | | description | 방의 삭제에 관한 요구사항 | | inputs | 방 정보 | | outputs | 갱신된 방 리스트 | | rationale | 없음 | | action | 출발 시간이 지났을 경우, 방에서 [출발하기] 버튼을 누를 경우, 방에 남아 있는 유저가 존재하지 않을 경우 방 정보를 리스트에서 삭제한다. | | references | **- #UC04, #UC05** |  |  |  | | --- | --- | | name: ID | Chatting: FR07 | | description | 채팅에 관련된 요구사항이다. | | inputs | 유저가 입력한 메시지 | | outputs | 유저가 입력한 메시지, 갱신된 UI | | rationale | 없음 | | action | 채팅방 화면은 방에 있는 모든 유저가 공유한다.  유저가 텍스트 입력 구간에 텍스트를 입력한 후 전송 버튼을 누르면 해당 텍스트가 채팅방 화면에 출력된다.  채팅 로그는 방이 존재할 때까지만 존재하고 방이 종료되면 로그는 남지 않는다.  텍스트 전송 가능한 길이는 500 byte 이내로 한다 | | references | **- #UC04, #FR05, #NFR02** |  |  |  | | --- | --- | | name: ID | Membership: FR08 | | description | 포맷에 맞춰 회원 가입을 수행하는 요구사항이다. | | inputs | 유저의 ID/PW/닉네임 입력 | | outputs | 회원 가입 성공/실패 여부 | | rationale | 없음 | | action | 유저의 닉네임은 최소 4byte에서 최대 16byte로 제한한다.  유저가 ID/PW/PW확인/닉네임을 입력한다.  PW는 영문과 숫자 조합만 가능하며 길이는 8~16자로 한다.  만약 DB에 중복되는 정보가 있으면 다시 입력하라고 알려주고  없으면 DB에 저장하고 로그인 UI를 띄운다.  유저가 모든 정보를 올바르게 입력했을 때만 [완료] 버튼을 활성화한다. | | References | **- #FR09** |  |  |  | | --- | --- | | name: ID | Signing in: FR09 | | description | 로그인을 수행하는 요구사항이다. | | inputs | 로그인하고자 하는 ID/PW | | outputs | 실패 시 팝업 창, 성공 시 Setting the information UI | | rationale | 없음 | | action | 유저가 ID/PW를 입력하고 [로그인] 버튼을 누르면 시스템은 DB에 있는 ID/PW들과 비교한다.  일치하는 계정이 있으면 시스템은 유저에게 로그인된 상태를 부여한다.  일치하는 계정이 없으면 실패와 재입력을 알리는 팝업 창을 띄운다. | | references | **- #FR08** |  |  |  | | --- | --- | | name: ID | Rating users: FR10 | | description | 유저끼리 상호 평가를 수행하게 하는 요구사항이다. | | inputs | 해당 방에 있는 유저들의 정보 | | outputs | 갱신된 유저들의 정보, Setting the information UI | | rationale | 없음 | | action | [좋아요] 버튼을 눌렀을 경우 해당 유저의 인기도를 임시적으로 1 증가시킨다.  [싫어요] 버튼을 눌렀을 경우 해당 유저의 인기도를 임시적으로 1 감소시킨다.  [그저 그래요] 버튼을 눌렀을 경우 인기도는 변하지 않는다.  단, 옵션은 중복 선택 불가능하다.  [평가 완료] 버튼은 모든 유저에 대해 옵션 중 한 가지를 눌렀을 경우에만 활성화된다.  [평가 완료] 버튼을 눌렀을 경우 임시적인 변화를 서버에 제출해 유저 정보 갱신을 요청한다.  완료 시 Setting the information UI로 이동한다. | | References | **- #UC06** | |
| **6. Nonfunctional system requirements** |
| **1)**   |  |  | | --- | --- | | name: ID | Operational Requirements: NFR01 | | description | 이 요구사항은 시스템의 구동 환경을 정의하기 위한 것이다. | | rationale | 없음 | | action | Android 기반 스마트폰에서만 구동 가능하다.  인터넷 환경이 원활해야 한다.  유저는 GUI를 통해 시스템과 상호 작용할 수 있다.  서버는 Always-on 상태이어야 한다.  앱 설치 전 여유공간은 최소 50MB이다. | | References | **- 없음** |   **2)**   |  |  | | --- | --- | | name: ID | Performance Requirements: NFR02 | | description | 이 요구사항은 시스템의 성능을 측정하기 위한 것이다. | | rationale | 없음 | | action | 서버의 응답시간은 2초 이내이어야 한다. (유저 환경은 고려하지 않는다)  서버가 종료되었다 다시 구동되어도 기존 연결들은 유지되어야 한다. | | References | **- #UC04, #FR07** |   **3)**   |  |  | | --- | --- | | name: ID | Security Requirements: NFR03 | | description | 이 요구사항은 시스템의 보안과 접근성에 관한 것이다. | | rationale | 없음 | | action | DB에 저장된 유저 정보는 관리자만 열람할 수 있다.  모든 텍스트는 데이터 무결성을 가진다. | | references | **- 없음** |   **4)**   |  |  | | --- | --- | | name: ID | Cultural and Political Requirements: NFR04 | | description | 이 요구사항은 시스템의 구동 환경을 정의하기 위한 것이다. | | rationale |  | | action | 현재(2018.05) 이 어플리케이션은 대한민국에서만 서비스된다.  현재는 한국어만 지원한다.  이 어플리케이션을 사용함으로써 야기되는 모든 사회적, 법적 문제는 책임지지 않는다.  이 어플리케이션은 여객운수사업법, 개인정보 보호법을 준수한다.  유저 정보는 최근 접속 후 1년 동안 유지하고 이후 파기한다. | | references | **- 없음** | |
| **7. System models** |
| 1. Structural model    2. behavioral model |

