|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 유형 | **Server** | 요구사항 ID | | FR-I-001 | |
| NO | **1** | 요구사항 명 | | Log-in 기능 제공 | |
| 요구사항 설명 | O2O Editor 어플리케이션을 이용하기 위해 로그인 서비스를 제공한다. | | | | |
| 상세설계 | 사용자는 Application을 키게 되면 Main-Activity를 보게 된다.  화면 구성은 다음과 같다.    (어플리케이션 실행 시 화면)  사용자는 기존 등록된 사용자라면, 사용자의 ID와 Password 칸을 채운 뒤 아래의 Login 버튼을 누르면 Socket을 통해 Server에 CHECK LOGIN <ID><PW>  의 내용이 보내진다.  Server의 Handler에서 CHECK와 LOGIN을 받았을 경우, check\_login() 함수를 실행시켜, Client로부터 받은 <ID>와 <PW>을 기반으로 사용자의 정보가 담긴 데이터베이스 테이블에서 Select한다.  이때 프로토콜은 다음과 같다.   |  |  |  | | --- | --- | --- | |  | Android send Request | Server response | | 정보 확인 | CHECK LOGIN <ID><PW> | 200 OK | | 405 Failed information | | | | | |
| 중요도 | 중 | | 난이도 | | 하 |
| 담당자 | 이경민 | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 유형 | **Server** | 요구사항 ID | | FR-I-002 | |
| NO | **2** | 요구사항 명 | | 회원가입 기능 제공 | |
| 요구사항 설명 | 신규 사용자에게 로그인을 위한 회원가입 서비스를 제공한다. | | | | |
| 상세설계 | O2O Editor 신규 사용자라면 Service를 받기 위해 회원가입 절차를 반드시 거쳐야 한다. 사용자는 어플리케이션 실행 시 FR-I-001에서 제공하는 로그인 화면에서 하단의 오른쪽 Register 버튼을 이용해 회원가입을 한다.  회원가입의 화면 구성은 다음과 같다.    ( 어플리케이션 실행 시 화면 ) ( 회원 가입 화면 )    다음 화면은 Android Register Activity 라고 불리는데, 이 액티비티에는 Name, UserID, Password의 Entry를 제공한다. 사용자는 3개의 Entry에 대해 입력을 하고 Register 버튼을 통해 회원가입을 한다. 사용자가 Entry에 입력을 하고 Register버튼을 누르면 그 즉시 App에서 Server로 TCP/IP Socket을 통해 CHECK REGISTER <ID><PW><NAME>가 전송된다.  이때 프로토콜은 다음과 같다.   |  |  |  | | --- | --- | --- | |  | Android send Request | Server response | | 정보 확인 | CHECK REGISTER <ID><PW><NAME> | 200 OK | | 400 Already existed in |   사용자 정보를 담고 있는 데이터베이스를 이용하여 해당 ID가 존재하지 않을 경우 회원가입을 진행한다. 이때 해당 ID가 이미 존재하는 경우 사용자는 다시 회원가입 절차를 시도해야 한다. | | | | |
| 중요도 | 하 | | 난이도 | | 하 |
| 담당자 | 이경민 | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 유형 | **Server** | 요구사항 ID | | FR-I-003 | |
| NO | **3** | 요구사항 명 | | 회원가입 등록 성공 화면 제공 | |
| 요구사항 설명 | 사용자가 회원가입 절차를 거친 뒤 회원 가입 성공 여부에 대해 판별할 수 있도록 제공한다. | | | | |
| 상세설계 | 1. 사용자가 FR-I-002에서 해당 정보를 기입한 뒤, Register 버튼을 눌렀을 때, FR-I-002에서 Server로부터 '200 OK' 프로토콜을 받을 경우 회원가입 절차를 정상적으로 마침을 말한다. 회원가입이 성공했다는 Activity 또는 Toast 메시지를 보여준 후 로그인 화면을 보여준다.  2. FR-I-002에서 Server로부터 '400 Already existed in' 프로토콜을 받을 경우 해당 ID 사용자가 존재하고 있음을 말한다. 이때 Client에게 회원가입 실패에 대한 메시지와 다시 회원가입을 할 수 있도록 화면을 보여준다.   |  |  |  | | --- | --- | --- | |  | Server response | 어플리케이션 화면 | | 회원가입 성공여부 | 200 OK | ( 로그인 화면 ) | | 400 Already existed in | ( 회원 가입 화면 ) |   이때 사용자의 등록 여부는 FR-I-002에서 사용자 정보가 담긴 데이터베이스에서 사용자가 사용하고자 하는 ID을 기반으로 판별된다. | | | | |
| 중요도 | 중 | | 난이도 | | 하 |
| 담당자 | 이경민 | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 유형 | **Server** | 요구사항 ID | | FR-I-004 | |
| NO | **4** | 요구사항 명 | | 로그인 불일치 화면 제공 | |
| 요구사항 설명 | 사용자가 로그인 시도하여 실패했을 경우 사용자에게 로그인 실패 화면을 제공한다. | | | | |
| 상세설계 | 사용자가 어플리케이션을 실행하여 로그인을 시도하거나, 회원가입 후 로그인을 시도하였을 경우 결과는 로그인 성공과 실패 두 가지로 나뉜다.  이때 사용자가 입력한 ID와 Password가 일치하지 않는 경우, 또는 ID가 존재하지 않는 경우 로그인 실패로 Server에서 Client에게 '405 Failed information' 을 전송하고 로그인 실패 시 Activity 또는 Toast 메시지를 띄운 후 다시 로그인 화면을 보여준다.   |  |  |  | | --- | --- | --- | |  | Server response | 어플리케이션 화면 | | 로그인 성공여부 | 405 Failed information | Fail  ( 실패 메시지와 로그인 화면 ) | | | | | |
| 중요도 | 하 | | 난이도 | | 하 |
| 담당자 | 이경민 | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 유형 | **Server** | 요구사항 ID | | FR-I-005 | |
| NO | **5** | 요구사항 명 | | Main-menu 기능 제공 | |
| 요구사항 설명 | 사용자가 로그인 성공 시 O2O Editor의 Main-menu 기능을 제공 한다. | | | | |
| 상세설계 | 사용자가 어플리케이션을 실행하여 로그인을 시도하거나, 회원가입 후 로그인을 시도하였을 경우 결과는 로그인 성공과 실패 두 가지로 나뉜다.  이때 사용자가 입력한 ID와 Password가 일치하는 경우, 로그인 성공으로 Server에서 Client에게 '200 OK'을 전송하고 로그인 성공 시 Activity 또는 Toast 메시지를 띄운 후 Main-menu 화면을 보여준다.   |  |  |  | | --- | --- | --- | |  | Server response | 어플리케이션 화면 | | 로그인 성공여부 | 200 OK | ( Main-menu 화면 ) |   Main-menu이 왼쪽 버튼의 경우 Service Button, 오른쪽 버튼의 경우 Train Button이다.  사용자가 Server에서 제공하는 임의의 String에 대해 Train을 시키고 싶다면 Train Button을 누르고 자신이 손으로 작성한 문서를 Editable한 Computer 문서로 바꾸어주는 O2O Editor Service를 받아 보고 싶다면 Service 버튼을 누르면 된다. 각 Button에 대해서는 이벤트 처리를 하기 위한 Method가 구현되어 있는데, 각 버튼에 따라 Method가 실행 되고 그에 따른 Activity가 띄어지게 된다. | | | | |
| 중요도 | 중 | | 난이도 | | 하 |
| 담당자 | 이경민 | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 유형 | **Server** | 요구사항 ID | | FR-I-006 | |
| NO | **6** | 요구사항 명 | | Train 기능 제공 | |
| 요구사항 설명 | 사용자 개인의 글자체에 대한 정보를 학습시켜 인식률을 높이기 위한 기능을 제공한다. | | | | |
| 상세설계 | 사용자가 FR-I-005에서 Train Button을 누르게 되면 다음과 같은 화면이 나타난다.   |  |  | | --- | --- | |  |  | | Train Button 눌렀을 때 화면 | Upload 버튼 눌렀을 때 화면 | |  |  | | Camera 버튼 눌렀을 때 화면 | Confirm 버튼 눌렀을 때 화면 |   Train 버튼을 누르면 다음과 같은 화면으로 넘어간다. 맨 위 상단에는 Server에서 임의로 알파벳이 모두 들어간 영어 문장을 대문자와 소문자로 주어주고 그 밖에 숫자와 markdown 문법에 사용될 기호들을 제시한다. 그러면 사용자는 해당 제시 글에 맞춰 손으로 글을 작성한 뒤 Camera 버튼을 눌러 촬영하고 upload 버튼을 누른 뒤 아래 네모 박스에 위치한 사진 미리 보기 서비스를 이용하여 올바른 사진을 업로드 했는지 확인 한 뒤, Confirm 버튼을 누르면 Train 과정이 끝난다.  사용자가 Confirm 버튼을 누르면 Server에서는 TRAIN IMAGE 에 해당하는 protocol을 받으면 사용자가 보낸 사진을 받기 위한 get\_image() 함수가 실행되고 이렇게 받은 함수를 Tesseract Train과정으로 넘겨 해당 사용자의 Traineddata를 생성한다. 이때 Traineddata는 저장될 때 기존의 사용자가 몇 번 Train시켰는지를 알기 위해 사용자 정보를 담고 있는 데이터베이스에서 TRAIN\_NUM을 검사하여 해당 값을 부여받는다.  다음과 같이 Train이 되는 동안 Train Activity를 통해 Progress bar를 이용해 Train이 되고 있다는 것을 표시해준다. 만약 사용자가 Train을 시킬 이미지가 클 경우는 Trained Data가 생기기 전까지 계속 Progress bar 는 돌게 된다.  Server로부터 train이 완료 되었다는 message를 받게 되면 progressbar는 종료 되고 성공하게 된다.  Server와 Client가 주고받는 Protocol은 다음과 같다.   |  |  |  | | --- | --- | --- | |  | Android send Request | Server response | | O2O Train | TRAIN IMAGE | 200 OK | | 400 Already existed in | | 425 Failed train | | | | | |
| 중요도 | 중 | | 난이도 | | 하 |
| 담당자 | 이경민 | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 유형 | **Server** | 요구사항 ID | | FR-I-007 | |
| NO | **7** | 요구사항 명 | | Service 기능 제공 | |
| 요구사항 설명 | 사용자에게 O2O Editor System을 통해 Handwriting을 Editable한 Computer 문서로 바꾸어 주는 Service을 제공한다. | | | | |
| 상세설계 | 사용자가 FR-I-005에서 Service Button을 누르게 되면 다음과 같은 화면을 보게 된다.    ( Service 버튼을 눌렀을 때 화면)  Service 버튼을 눌렀을 때 사용자에게 제공하는 기능들은 다음과 같다. 먼저 사용자가 Service를 받아보고 싶은 문서의 사진을 Upload버튼을 이용하여 사진을 선택한 뒤 Recognition 버튼을 이용하여 사진을 Server에 전송한다. | | | | |
| 중요도 | 중 | | 난이도 | | 하 |
| 담당자 | 이경민 | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 유형 | **Server** | 요구사항 ID | | FR-I-008 | |
| NO | **8** | 요구사항 명 | | 카메라 기능 제공 | |
| 요구사항 설명 | 사용자가 어플리케이션 내에서 사진을 찍어 Service을 이용 받을 수 있게 카메라 기능을 제공한다. | | | | |
| 상세설계 | 사용자가 FR-I-007에서 카메라를 사용하고 싶을 경우 다음과 같은 Button을 제공한다.  C:\Users\이경민\Desktop\제목 없음.png  ( Camera 버튼)  Button을 누르게 되면 카메라와의 연동으로 바로 사진촬영을 할 수 있다.  Serivce Activity에서 카메라와 같은 버튼의 이미지는 ImageView를 통해 보여주었다. 카메라 연동은 Intent 객체를 통해 OnClickCamera 메소드에서 구현하였다. | | | | |
| 중요도 | 중 | | 난이도 | | 하 |
| 담당자 | 이경민 | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 유형 | **Server** | 요구사항 ID | | FR-I-009 | |
| NO | **9** | 요구사항 명 | | 업로드 기능 제공 | |
| 요구사항 설명 | 사용자가 문서 변환을 원하는 사진을 Server에 전송하기 위해 사진을 업로드 하는 기능을 제공한다. | | | | |
| 상세설계 | 사용자가 이미지 업로드를 위해 Application 에서는 다음과 같은 버튼을 제공한다.  C:\Users\이경민\Desktop\asdasd.png  ( Upload 버튼 )  FR-I-005에서 업로드를 원할 경우 업로드 버튼을 누르면 된다. Android에서 Image View라는 객체를 통해 다음의 이미지를 제공하고 id를 부여함으로써 해당 이미지에 대한 값을 객체에 저장할 수 있다. 그리고 Intent와 startActivity를 통해 갤러리로 이동하고 사진을 선택 할 수 있는 기능을 제공한다. | | | | |
| 중요도 | 중 | | 난이도 | | 하 |
| 담당자 | 이경민 | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 유형 | **Server** | 요구사항 ID | | FR-I-010 | |
| NO | **10** | 요구사항 명 | | Recognition 기능 제공 | |
| 요구사항 설명 |  | | | | |
| 상세설계 | 사용자가 자신이 인식시키고 싶은 이미지에 대한 버튼에 대해서는 다음과 같이 제공한다.  C:\Users\이경민\Desktop\zxcc.png  ( Recognition 버튼 )  FR-I-005에서 character image에 대해 인식을 시키고 싶을 경우 다음과 같은 버튼을 누르면 된다. 버튼을 누를 시 이미지는 Server로 넘어가게 되고 인식이 된 결과를 서버로부터 e-mail로 받아볼 수 있고, 스마트 폰에서 직접 확인할 수 있다.  Client에서 Recognition 버튼을 누르면 Server로 SERVICE IMAGE에 대한 프로토콜이 전송되고 이때 Server내에서 이미지를 받아 처리하는 get\_image() 함수를 실행시킨다. Server에서 이미지를 받을 준비가 되었다는 '210 Ready to Receive'를 보내면 그제서야 Client에서 이미지를 Server로 전송한다. 이미지를 제대로 전송 받았을 때 '200 OK'를 전송하고 load중 오류가 발생할 경우 '410 Failed load image'에 대한 프로토콜을 받게 된다.  Server-Client 프로토콜은 다음과 같다.   |  |  |  | | --- | --- | --- | |  | Android send Request | Server response | | O2O Service | SERVICE IMAGE | 210 Ready to Receive | | 200 OK | | 410 Failed load image | | | | | |
| 중요도 | 중 | | 난이도 | | 하 |
| 담당자 | 이경민 | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 유형 | **Server** | 요구사항 ID | | FR-S-001 | |
| NO | **11** | 요구사항 명 | | Get user ID and Password | |
| 요구사항 설명 | 사용자에게 Customizing된 문자인식을 제공하기 위해 개인의 로그인 정보를 전달 받을 수 있어야 한다. | | | | |
| 상세설계 | O2O Editor System은 사용자가 어플리케이션을 사용하는데 있어 기본 전제가 로그인을 통해 사용자 개인의 정보를 따로 저장하고 보관하는데 있다. 따라서 사용자는 Server에게 FR-I-001에서 제공하는 로그인 기능을 이용하여 ID와 Password을 제공해야 한다.    ( 로그인 화면 )  Server에서는 Client가 보낸 <ID>와 <PW>를 받아 FR-D-001을 사용하여 로그인 성공 여부를 처리한다. | | | | |
| 중요도 | 상 | | 난이도 | | 중 |
| 담당자 | 권영은 | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 유형 | **Server** | 요구사항 ID | | FR-S-002 | |
| NO | **12** | 요구사항 명 | | Get Image file | |
| 요구사항 설명 | 사용자가 문자인식 하고자 하는 사용자의 문자가 작성된 종이의 이미지 file을 전달 받을 수 있어야 한다. | | | | |
| 상세설계 | 사용자가 Service 부분을 받기 위해서는 문자 인식을 받고자 하는 사용자가 손으로 쓴 글씨의 문서를 찍어 Server으로 전송하는 과정이 필요하다. 사용자는 어플리케이션에서 FR-I-009를 사용하여 사진 이미지를 선택하고 FR-I-010을 이용하여 Service을 받기 위해 사진을 Server으로 전송하게 된다.  C:\Users\이경민\Desktop\asdasd.png C:\Users\이경민\Desktop\zxcc.png  ( Upload 버튼) ( Recognition 버튼)  Server에서 사진을 받을 준비가 되었을 때 Client에게 '210 Ready to Receive' 프로토콜을 보내주고 사진을 올바르게 전송 받았으면 Client에 '200 OK'를, download 중에 유실 및 에러가 발생 할 경우 '410 Failed load image’을 보내준다. | | | | |
| 중요도 | 상 | | 난이도 | | 중 |
| 담당자 | 권영은 | | | | |