

Personal Project Report

- Mobile Application Lab 2022 -

Student ID : 2018313063

Student Name : 송정현

어플리케이션 이름
Today's Routine
어플리케이션 설명
<p>오늘 할 운동루틴을 짜고 운동수행을 도와주는 앱입니다.</p> <p>자체 회원가입기능이 있고 회원가입할 때 입력한 id, password, nickname 정보가 관리됩니다.</p> <p>로그인을 하면 오늘 할 운동을 선택 할 수 있는데 기능 앱 자체에서 제공하는 운동 목록 말고 본인의 운동을 따로 추가할 수 있습니다. 최대 11개의 운동을 추가할 수 있습니다.</p> <p>이후 선택한 운동별로 세트수와 휴식시간을 설정하게 됩니다.</p> <p>이후 운동수행기능에서는 앞서 입력받은 운동이름, 휴식시간, 세트수를 통해서 자체 타이머가 세팅 되고 휴식버튼을 누르면 타이머가 작동하고 미리 설정해놓은 휴식시간이 다 가게 되면 beep 소리가 울립니다. 모든 세트가 끝났을 때는 세트가 남아있을 때와 다르게 beep소리가 두번 울립니다.</p> <p>운동 세트가 끝나면 알람 앱을 키고 휴식시간을 설정하는 귀찮은 작업이 사라졌습니다.</p> <p>운동을 SNS에 공유하는 사람들이 많은데 일일이 타이핑 할 필요가 없이 운동 종료 버튼을 누르면 오늘 수행한 운동들의 이름과 운동별 세트수가 클립보드에 저장됩니다.</p>
액티비티 설명
<p>Splash Activity</p> <p>앱을 키면 앱의 이름과 함께 MainActivity전 띄워지는 액티비티</p> <p>MainActivity</p> <p>로그인을 위해 id나 password를 입력하고, 로그인 버튼, 회원가입버튼을 통해 루틴선택 액티비티 혹은 회원가입 액티비티로 넘어갈 수 있음. 백엔드와 통신하여 id와 password 판별.</p> <p>SignUpActivity</p> <p>회원가입을 위한 액티비티. 백엔드와 통신하여 회원정보 추가. 중복된 id 방지.</p> <p>RoutineSelectActivity,</p> <p>오늘할 운동 루틴을 선택하는 액티비티. 앱 내에 미리 저장되어있는 운동루틴들을 checkbox를 체크해서 선택할 수 있고 목록에 없는 운동들은 사용자가 직접 추가할 수 있다. 최대 11개의 운동까지 선택이 가능하다. RoutineSelect Adapter로 리스트뷰를 Control한다.</p>

TrainingActivity, TrainingAdapter

선택한 운동들의 목록을 띄워주고 세트수와 휴식시간을 입력할 수 있는 액티비티. 운동을 수행했는지 안했는지 O와 X으로 옆에 표시해준다. Go버튼을 클릭하면 운동수행화면으로 넘어간다.

운동종료 버튼을 누르면 오늘 수행한 운동명과 각 운동들의 세트수가 String으로 변환되어 클립보드에 복사된다. 프로그램종료 버튼을 누르면 프로그램이 종료된다.

GoActivity

TrainingActivity에서 입력한 운동이름, 세트수, 휴식시간이 넘어와 이를 토대로 운동을 수행하는 액티비티. 한세트가 끝나고 set finish 버튼을 누르면 남아있는 세트의 수가 줄어들고 타이머가 작동된다. 타이머가 끝나면 beep소리가 울린다. 만약 남아있는 세트수가 없는경우 beep 소리가 두번 울린다. 종료버튼을 누르면 TrainingActivity로 돌아간다.

Network API 디자인 (AWS lambda – 안드로이드 간 통신 프로토콜 설명)

로그인과 회원가입에서 Network API 디자인 (MainActivity, SignupActivity)

회원정보의 형태는 id,password,nickname 이고 모두 String. Id는 unique한 값이다.

1. 로그인시 안드로이드로부터 입력받은 id와 password가 DB에 저장되어있는지 확인한다(POST). 입력받은 id와 password와 일치하는 값이 DB에 존재하면 True 를 리턴하고 없다면 False를 리턴한다. 안드로이드에서 리턴값을 확인해서 True라면 운동선택 액티비티로 넘어가고 False라면 오류 토스트 메시지를 출력한다.
2. 로그인의 리턴값이 True라면 입력받은 id를 GET method의 parameter로 넘겨 해당하는 id의 nickname을 리턴하고 이를 저장 및 전달한다.
3. 회원가입시, DB의 컬럼중 안드로이드에서 입력한 id와 같은 id를 가지는 컬럼이 있는지 확인한다(POST). 만약 있다면 회원가입에 입력한 id와 password, nickname을 회원정보로 해서 유에 저장하고 success를 리턴하고 아니라면 false를 리턴한다. 안드로이드에서 리턴값을 확인해서 True라면 다시 로그인화면으로 돌아가고 False라면 오류 토스트 메시지를 출력한다.

도전 요소 (Challenge)

1. Splash Activity 사용

기존에 있는 앱들처럼 앱을 실행하면 잠깐 띄워주는 화면을 구성하기 위해 `SplashActivity`를 사용

2. 운동선택화면에서 리스트뷰에 있는 체크박스를 체크하고 스크롤을 하면 체크박스의 체크상태가 초기화되는 현상 발생. 효율적인 자원관리를 위해 리스트뷰가 초기 상태를 재사용하는 특징 때문으로 판단 됨. 체크박스가 터치될 때마다 체크박스의 상태를 기존상태의 반대로 바꿔주는 작업을 `ListView Adapter`에 구현하여 해결함.

3. 다음화면으로 넘어가고 뒤로가기 하면 선택한 체크박스들은 모두 초기화 되지만 이전에 추가한 운동들은 그대로 남아있게 구현.

4. 앱 내장 Timer 구현, 소리 구현

- 분과 초를 받아서 시간이 가게하는 Timer구현.

-타이머가 끝나면 beep소리가 나는데, 마지막 루틴의 경우를 구분하기 위해 beep소리가 두번 나게 구현.

5. TrainingActivity의 리스트뷰안에 EditText 와 버튼 구현

-어댑터에서 버튼을 눌러 새로운 액티비티로 넘어갈 때 editText의 값이 저장되지 않고 넘어가는 문제가 발생. TextView의 값이 바뀔때마다 한글자글자를 저장하는 listener를 만들어 처리

6. ListView의 Item들을 스트링 값으로 변환하고 Clipboard에 접근해 복사하는 기능 구현

7. 앱 종료 기능 구현

`finishAffinity()`사용

어플리케이션 디자인 패턴

1. MainActivity, MainPresenter, MainModel

MainActivity :

- 새로운 액티비티 시작
- Toast오류 메시지 출력
- 화면에 있는 값들 return.

MainModel :

- 로그인시 입력한 ID와 Password가 DB에 있는지, valid한 값인지 판단하는 작업 수행
- 로그인시 입력한 ID에 해당하는 NICKNAME을 가져오는 작업 수행
- 리턴한 결과들을 가져오는 작업 수행
- onValueChangeListener안에 onChanged 함수 2가지 구현
: model의 작업이 끝났다는 것을 알리기 위함.

MainPresenter:

- 버튼이 touch되었을 때 Model과 Activity 연결

2. SignupActivity, SignupPresenter, SignupModel

SignupActivity :

- MainActivity로 돌아가는 기능
- 에러메세지 출력
- 화면에 있는 값들 return

SignUpModel

- 회원가입시 입력한 ID가 이미 DB에 있는지 판단하는 작업후 성공여부 리턴
- onValueChangeListener안에 onChanged 함수 구현
: model의 작업이 끝났다는 것을 알리기 위함.

SignupPresenter :

- 회원가입 버튼이 터치되었을 때 Model과 Activity 연결

3. RoutineSelectActivity, RoutineSelectPresenter, RoutineSelectModel

RoutineSelectActivity:

- listview의 체크박스 초기화
- listview 화면 업데이트
- 화면에 있는 값들 return
- 에러메세지 출력

-TrainingActivity로 전환

-EditText 초기화

RoutineSelectPresenter

버튼 두개를 터치 했을 때 Model과 Activity 연결

RoutineSelectModel

-리스트뷰 관리

-리스트뷰를 구성하는 ArrayList에 원소 추가

-운동추가기능 관리

4. GoActivity, GoModel, GoPresenter

GoActivity :

-타이머 값 update

-남아있는 세트 update

-에러메세지 출력

-TrainingActivity로 돌아가는 기능

GoModel:

-남아있는 세트수 return

-쉬는시간 return

-타이머 기능 구현

-타이머의 분과 초가 바뀔때마다 호출하는 onValuechangedListener 구현

-세트수 줄이는 줄이는 작업

Presenter:

GoActivity와 GoModel을 연결

- 추가할 내용이 있으면 표의 행을 아래에 추가하여도 괜찮습니다