수업시간 발표 내용

- 지난 번 수정사항 반영한 결과 발표

불필요하거나 중복되는 클래스를 제거 혹은 하나로 통합하였음

PersonalSettingStudentActivity, PersonalSettingProfessorActivity를 PersonalSettingActivity 하나로 통합

ProfessorSettingActivity, StudentSettingActivity를 SettingActivity 하나로 통합

- 사용된 자료구조를 자세하게 어느 부분에서 사용되었는 가에 대해 설명(실제 예를 들어 설명할 것)

뭘 설명해야할지 모르겠는데 굳이하자면…..

ClassSettingStudentActivity에서 자신의 수강과목을 보여주는 부분에서 ArrayList를 사용하여 정보를 보여줌

ClassSettingProfessorActivity에서 자신의 교과목을 보여주는 부분에서 ArrayList를 사용하여 정보를 보여줌

ProfessorMainActivity에서 화면에 과제, 과제이름, 과제기한, 과제 내용을 각각 ArrayList를 사용하여 정보를 보여줌

StudentMainActivity에서 화면에 과제, 과제이름, 과제기한, 과제 내용을 각가 ArrayList를 사용하여 정보를 보여줌

특히 StudentMainActivity에서는 자바에서 제공하는 Collator 오브젝트를 사용하여 리스트뷰를 보여줄때 과제제출 기한이 가까운 순서대로 리스트를 정렬하여 보여준다.

StudentRegistClass에서 과목을 검색할때 ArrayList를 사용하여 정보를 보여준다.

ProfessorMainActivity, StudentMainActivity, StudentRegistClass 3군데 모두 리스트뷰의 데이터가 많아질 경우 디바이스의 메모리가 부족해질 수 있기때문에 adapter를 만들어 주어 관리를 해준다.

adapter를 사용할때도 리스트 항목 row 마다 XML을 inflating하는 작업을 반복하게 된다면 100개, 1000개의 리스트 아이템을 보여줄 경우 그만큼의 inflating작업과 View 생성 작업을 해야 한다.(하나의 view는 Ram에서 1kilobytes,2kilobytes를 차지한다.) 아이템이 많아질 수록 앱의 동작은 느려지거나 끊기게 된다. 이렇게 하지않고 효율적으로 하기위해 convertView(재활용 view)를 사용하였다. convertView를 할때도 inflating하는 작업은 줄이지만 textview를 가져오기위해서는 매번 findViewById작업을 반복하기 때문에 ViewHolder Pattern을 사용하였다.

ViewHolder pattern은 작은 data저장소를 만들어 각각 리스트 아이템들을 static으로 연결하는 것이다.

재활용 View가 null일때 View를 inflate한 후 findViewById를 통해 찾은 View들을 ViewHolder에 저장 후 convertView의 Tag에 저장해놓는다. 그 후 재활용 View를 사용할때는 convertView의 Tag에 있는 ViewHolder를 가져와서 사용하는 구조이다. 이런 구조를 가짐으로써 중복적인 findViewById()를 호출하지 않아 성능적으로 우수한 구조를 가질 수 있다. (google I/O - The world of listview 세미나를 보시면 자세한 정보들을 얻을 수 있으므로 더 궁금하신분들은 요걸 참조하십쇼 라고 말해서 질문 차단ㅎ)

어플자체에는 사용자 이름, 사용자 id, 사용자의 프로필 사진 url, 사용자 신분(학생인지 교수인지) 이렇게 4가지만 string형태로 저장한다.

저장방식은 SharedPreferences를 사용하여 앱내에 저장하는 방식이다. 많은 양을 저장하지 않고 기본정보만 저장할때 유용하다. 상속받지 않고 어디서든 사용할 수 있다는 장점이 있다.

나머지 모든 정보는 파이어베이스에서 제공하는 Realtime Database를 사용하여 어플에 저장되는 정보를 최소한으로 하였다.

어플 내에서 값을 추가하거나(강좌 추가, 과제 만들기 등) 삭제되는 경우(과목삭제, 과제 삭제) 실시간으로 데이터베이스에 값이 변경되고 이에 관련된 모든 사용자가 즉시 변경사항이 반영된다.

제가 생각나는건 요거 뿐인데 이 항목에 넣기 애매한 부분 같으면 빼주시고 코드 보시고 혹시 추가할 내용 있으면 추가합시다.

- 프로젝트 ver1.0 시연

제출 내용

- 프로그램 코드를 압축하여 제출

- 보고서 / 발표자료 제출