NIL SHAH

Project Proposal

Prepared for: Dr. Sabah Mohammed

Prepared by: Nil Shah

March 27, 2015

NIL SHAH

Description

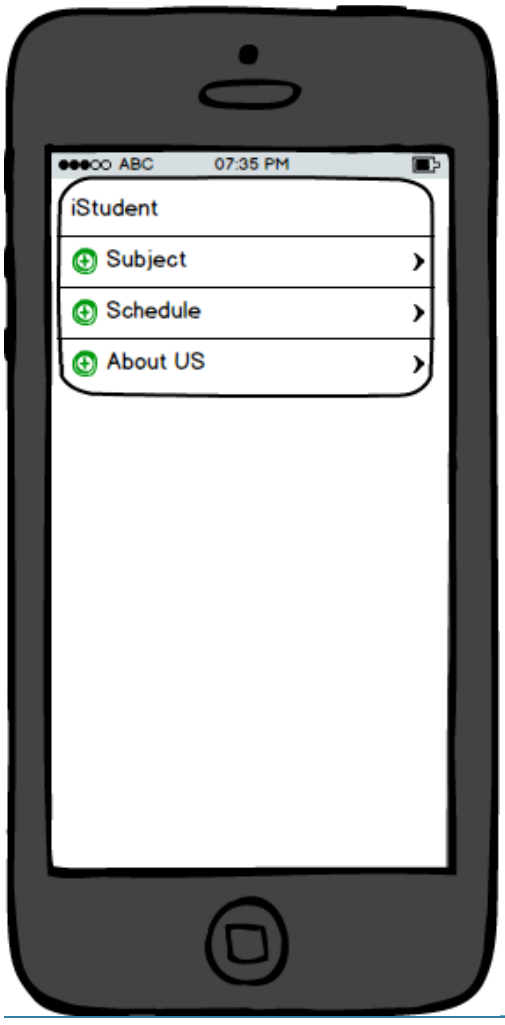
iStudent app is pure android application. iStudent app is new modern application for the students. this app basically record the lecture in audio format. student can create their day with the lecture and they can start record the lecture. also app is provide the facility to play to recorded audio to students so the students can easily study and refer the presentation of all lecture.

Target Platform

iStudent app is pure android application. you can run this application on android version 4.0 and Higher. you can run this application in any of the android emulator or any of the android device (Mobile and Tablets)

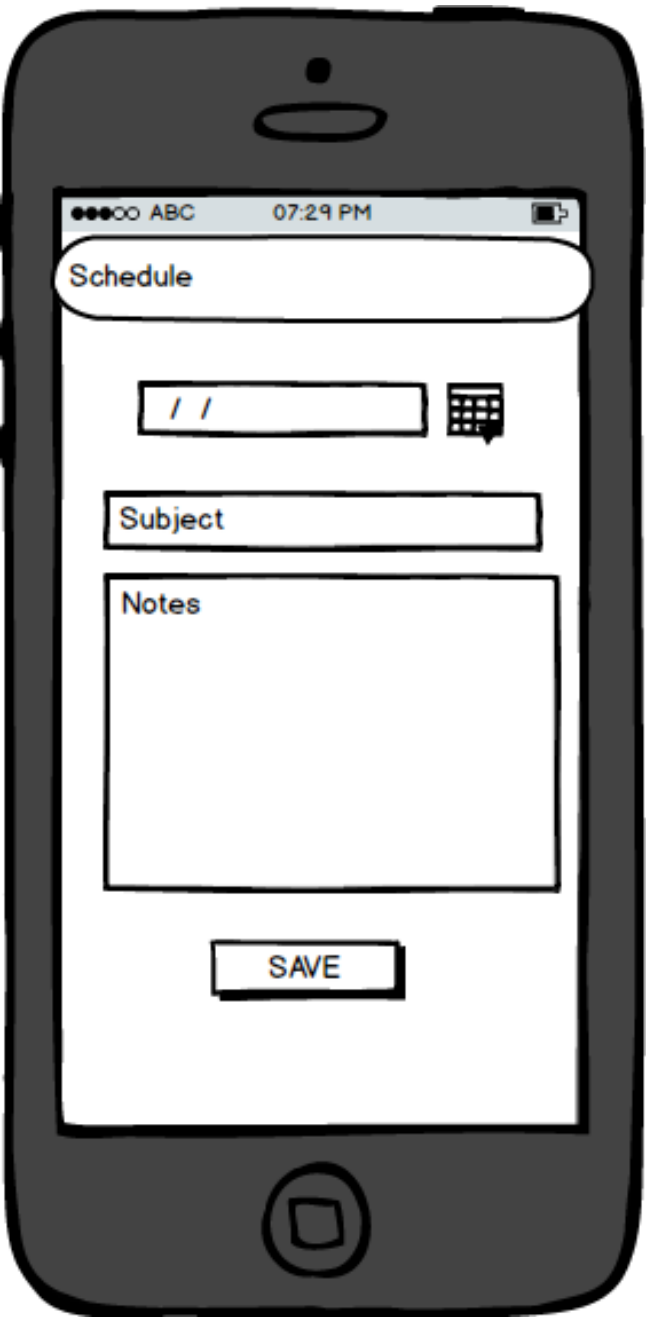
Development Platform

For Developing this app, i am using Eclipse IDE. ADT plugin will be require if you want to use this for android development.



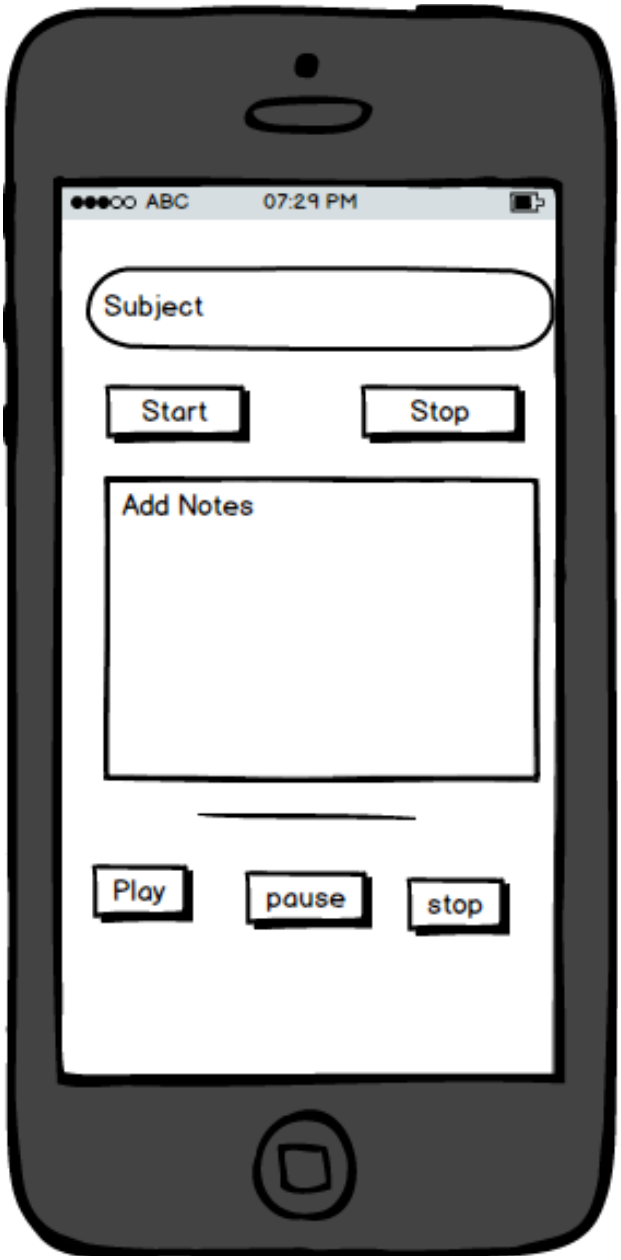
Detailed Description

The App is look like this screen shot. this is a starts screen of the application. here students can choose what they want to do.they can recored that lecture by choking subject option or also they can make a schedule of the subjects.



this is for the make the schedule. here students can make their schedule of their subject. application will notify him before the lecture time.

the data will be stored in the SQLite database. and on time data will pop to mobile screen to know about what the user have to do.



Here stundets can recored their lecture by adding the subject name and the notes
by clicking on start button user can start recording. at the end of class student can stop it.
Students can also add notes to the particular records.
below buttons play, pause and stop for the listen the recording whenever students need it.

Here the recored audio will store in the mobile storage in the one particular folder. so if user need directly that audio clip they can get that using some file manager application,

The application screen will fit to the all mobile devices. the application will automatically adjust the screen size.so we can run this app in any mobile or tablet.

You need some extra free storage for this app in your external Storage. the size of storage depends on how much lecture you record.

Risks:

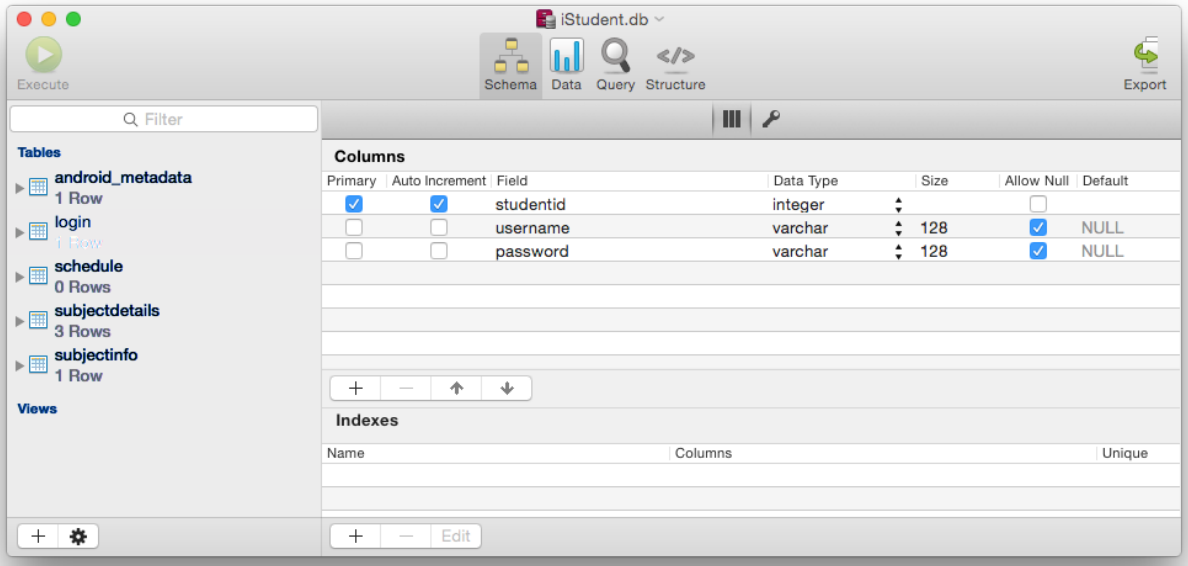
1) May be there a risk of loosing data or may be connection problem with the database.
2) Problem may be occur due to low storage space in device. because audio file required more space to store.all depends on how lone time you recored.
3) Its a only native android app so not available for another platform
4) Risk for run in emulator (External Storage)

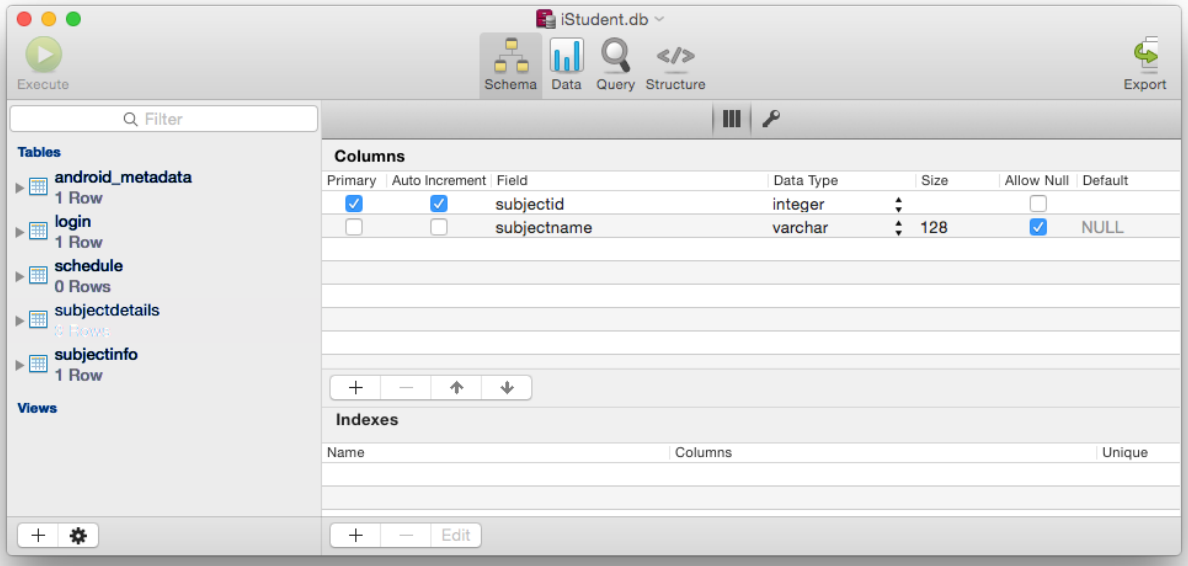
Demonstration Method:

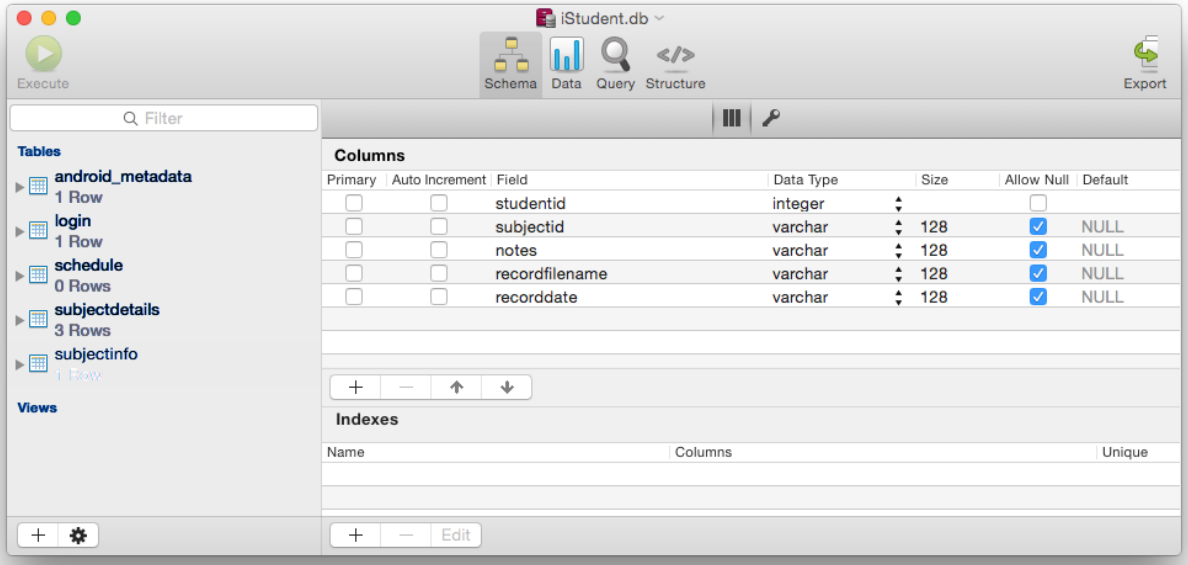
i'll going to show my app in my tablet(Nexus 7). and i'll also try to show in android emulator if all things goes perfect.

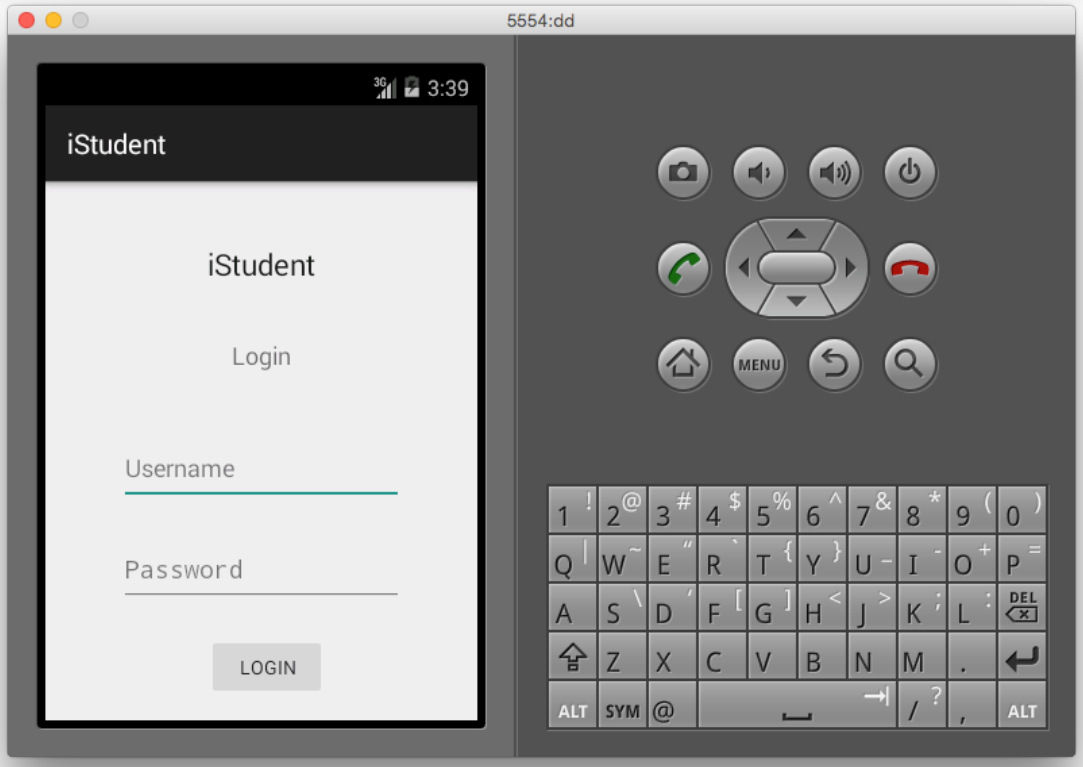
ReadME

My Database:

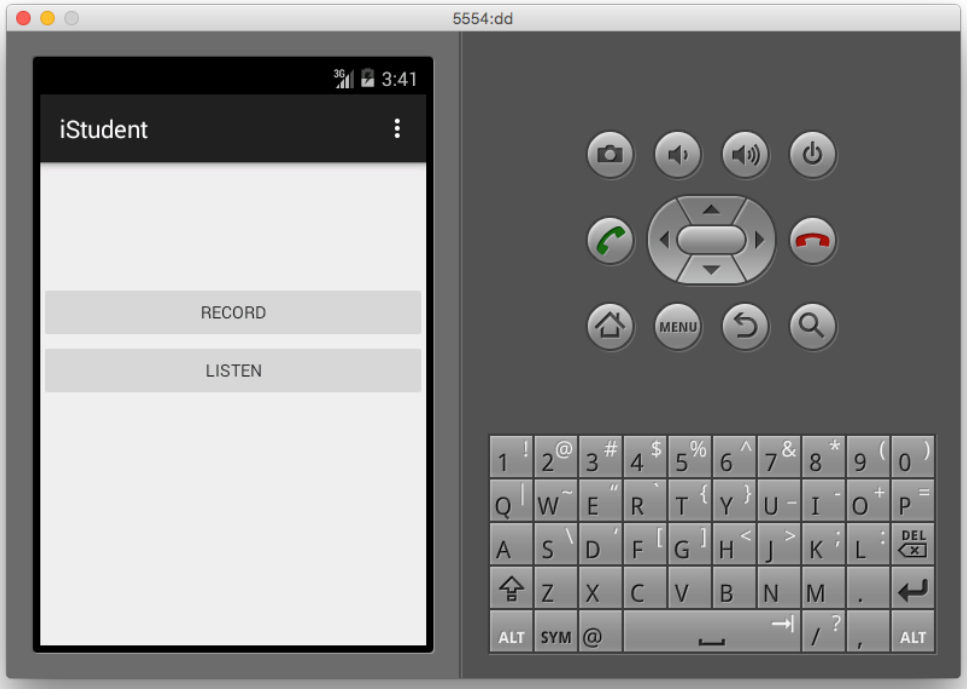


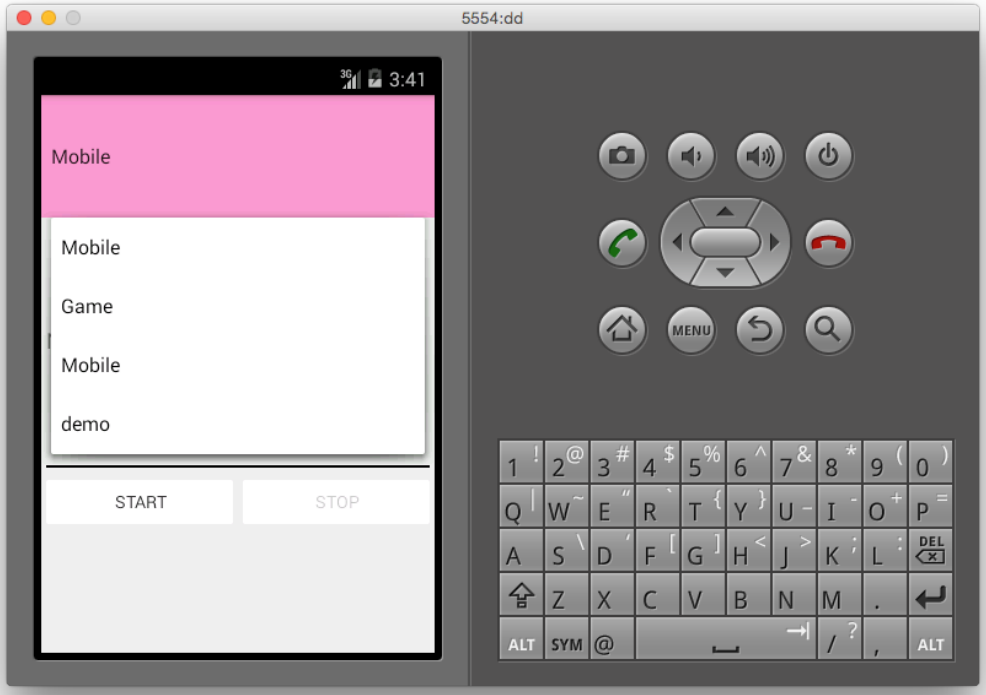


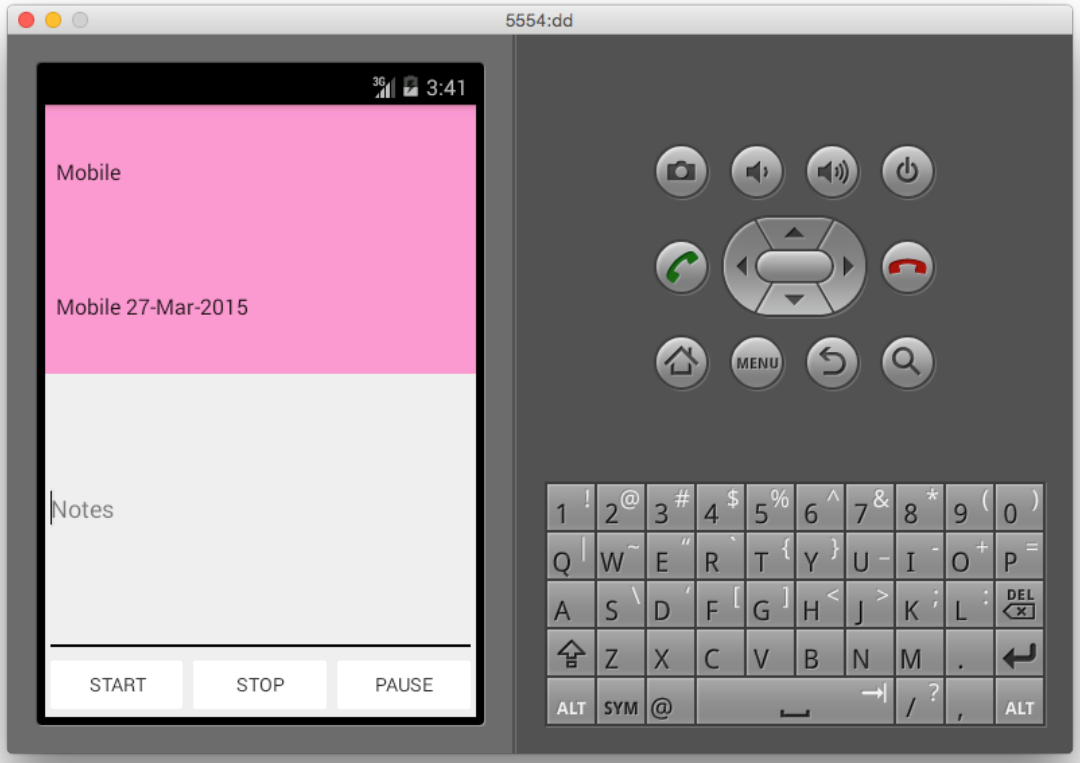

iStudent.db
Execute
Schema
Data
Query
Structure
Export
Filter
Tables
android_metadata
1 Row
login
1 Row
schedule
0 Rows
subjectdetails
3 Rows
subjectinfo
Views
Columns
Primary
Auto Increment
Field
Data Type
Size
Allow Null
Default
studentid
integer
subjectid
varchar
128
NULL
notes
varchar
128
NULL
recordfilename
varchar
128
NULL
recorddate
varchar
128
NULL
Indexes
Name
Columns
Unique
Edit

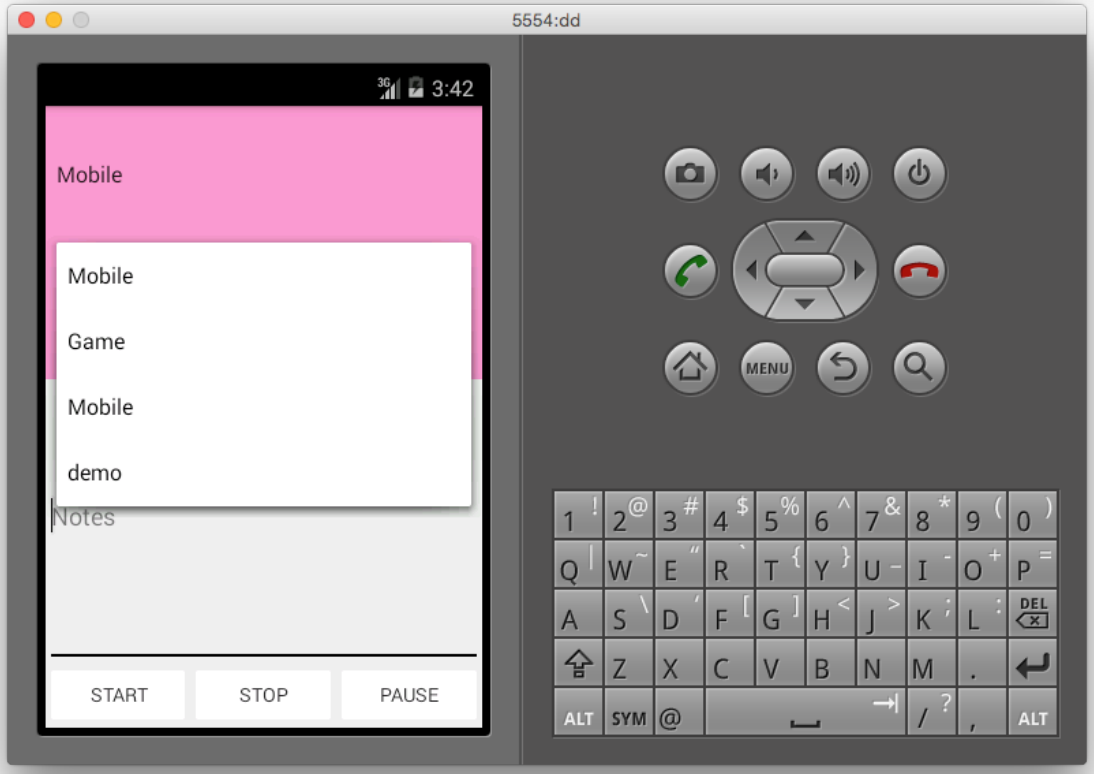

5554:dd
3:39
iStudent
iStudent
Login
Username
Password
LOGIN
MENU

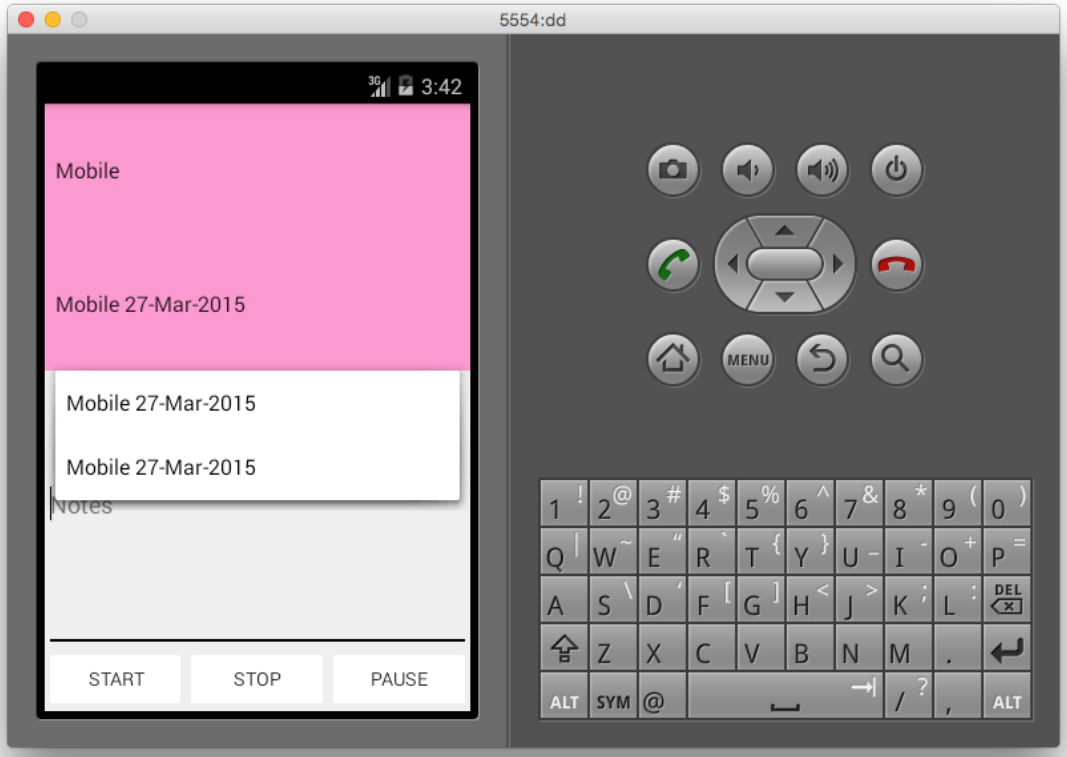
jhghh

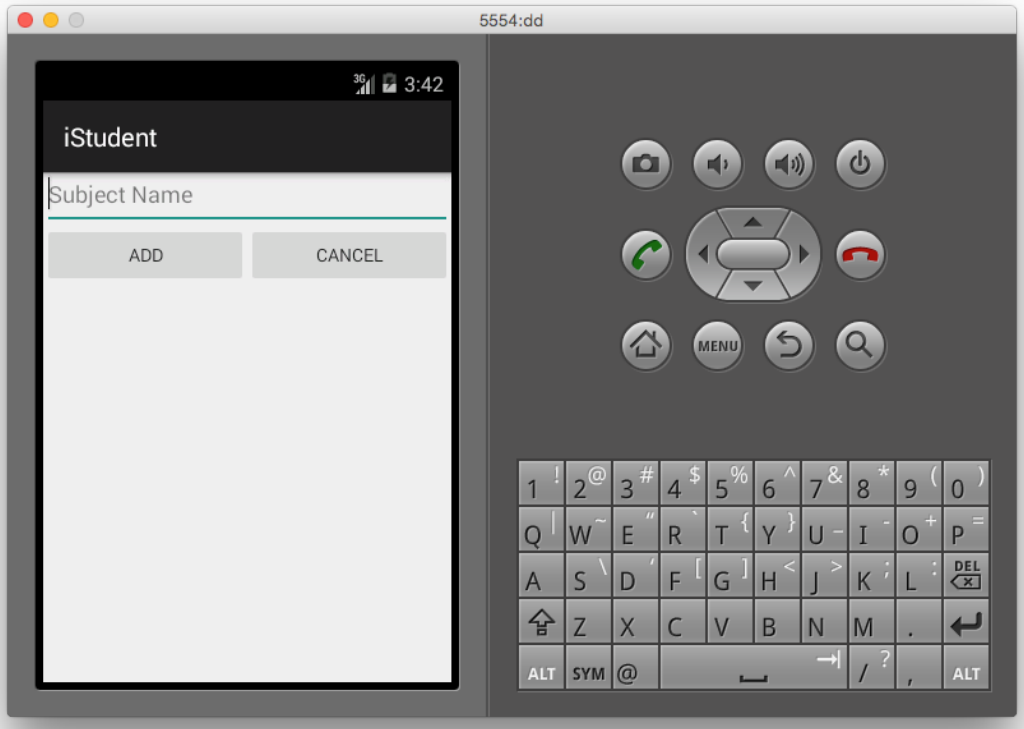

5554:dd
3:41
iStudent
RECORD
LISTEN
MENU
ALT
SYM
@
ALT


5554:dd
3:41
Mobile
Mobile
Game
Mobile
demo
START
STOP
MENU
ALT
SYM
@
ALT


5554:dd
3:41
Mobile
Mobile 27-Mar-2015
Notes
START
STOP
PAUSE
MENU


5554:dd
3:42
Mobile
Mobile
Game
Mobile
demo
Notes
START
STOP
PAUSE
MENU


5554:dd
3:42
Mobile
Mobile 27-Mar-2015
Mobile 27-Mar-2015
Mobile 27-Mar-2015
Notes
START
STOP
PAUSE
MENU
ALT
SYM


5554:dd
3:42
iStudent
Subject Name
ADD
CANCEL
MENU
ALT
SYM

Some Coding Functionalities:

android device permissione

```
<uses-sdk
    android:minSdkVersion="10"
    android:targetSdkVersion="21" />

<uses-permission android:name="android.permission.RECORD_AUDIO" />
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />

<supports-screens
    android:anyDensity="true"
    android:largeScreens="true"
    android:normalScreens="true"
    android:smallScreens="true"
    android:xlargeScreens="true" />
```

Database Create/Open/Close

```
public void createDataBase() throws IOException
{
    //If database not exists copy it from the assets

    boolean mDataBaseExist = checkDataBase();
    if(!mDataBaseExist)
    {
        this.getReadableDatabase();
        this.close();
        try
        {
            //Copy the database from assests
            copyDataBase();
            Log.e(TAG, "createDatabase database created");
        }
        catch (IOException mIOException)
        {
            throw new Error("ErrorCopyingDataBase");
        }
    }
}
```

```
    //Open the database, so we can query it
    public boolean openDataBase() throws SQLException
    {
        String mPath = DB_PATH + DB_NAME;
        //Log.v("mPath", mPath);
        mDataBase = SQLiteDatabase.openDatabase(mPath, null, SQLiteDatabase.CREATE_IF_NECESSARY);
        //mDataBase = SQLiteDatabase.openDatabase(mPath, null, SQLiteDatabase.NO_LOCALIZED_COLLATORS);
        return mDataBase != null;
    }
```

Audio Recording:

```
private void startRecording() {
    recorder = new MediaRecorder();

    recorder.setAudioSource(MediaRecorder.AudioSource.MIC);
    recorder.setOutputFormat(output_formats[currentFormat]);
    recorder.setAudioEncoder(MediaRecorder.AudioEncoder.AMR_NB);
    recorder.setOutputFile(getFilename());

    recorder.setOnErrorListener(errorListener);
    recorder.setOnInfoListener(infoListener);

    try {
        recorder.prepare();
        recorder.start();
    } catch (IllegalStateException e) {
        e.printStackTrace();
    } catch (IOException e) {
        e.printStackTrace();
    }
}
```

```
  private String getFilename() {
      String filepath = Environment.getExternalStorageDirectory().getPath();
      //String filepath="/data/data/" + this.getPackageName();
      File file = new File(filepath, AUDIO_RECORDER_FOLDER);

      if (!file.exists()) {
          file.mkdirs();
      }

      return (file.getAbsolutePath() + "/" + spnSubjectname.getSelectedItem().toString()+" "+TodayDate + fil
  }
```

```
try
{
    rec=spnRecordfile.getItemAtPosition(pos).toString();
    Log.i("Record",Environment.getExternalStorageDirectory().getPath()+"/AudioRecorder/"+rec+".mp4");
    mp.setDataSource(Environment.getExternalStorageDirectory().getPath()+"/AudioRecorder/"+rec+".mp4");
    //mp.setDataSource("/data/data/" + this.getPackageName()+"/AudioRecorder/"+spnRecordfile.getSelectedItem().toString()+".mp4");
    mp.prepare();
}
catch(Exception e){e.printStackTrace();}
```

```
    start.setOnClickListener(new OnClickListener() {
        @Override
        public void onClick(View v)
        {
            mp.start();
        }
    });
    pause.setOnClickListener(new OnClickListener() {
        @Override
        public void onClick(View v) {
            mp.pause();
        }
    });
    stop.setOnClickListener(new OnClickListener() {
        @Override
        public void onClick(View v) {
            mp.stop();
        }
    });
}
```