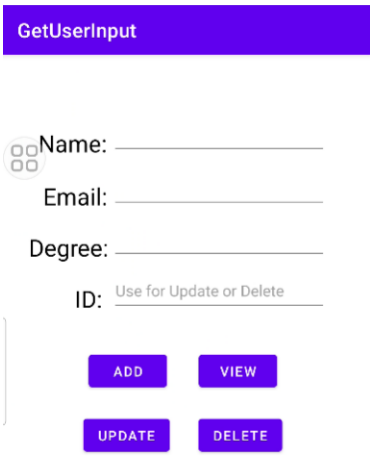
CT60A4800 Fundamentals of smart systems – Assignment 2

Name: Trieu Huynh Ba Nguyen Student number: 000405980

The application was developed in Java, using Android API (level 32) and takes in user-entered data. The data will be saved using SQLite database. The users can view, update, and delete the data later.



The user enters their name, email, and degree program for insertion. After the first insertion, the data can be viewed. To update the data, the user needs to enter the new details and the ID of the data that needs to be changed. To delete the data, the user only needs to enter the ID of the data that needs to be deleted. These functions are performed using SQLite queries.

DatabaseHelper.java: This class primarily contains functions that creates the SQLite database and table, and executes relevant queries.

```
package com.example.getuserinput;
import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;

public class DatabaseHelper extends SQLiteOpenHelper{
    public static final String DATABASE_NAME = "data.db";
    public static final String TABLE_NAME = "student";
    public static final String COL1 = "id";
    public static final String COL2 = "name";
    public static final String COL3 = "email";
    public static final String COL4 = "degree";
    /* Create a new database*/
    public DatabaseHelper(Context context) {
        super(context, DATABASE_NAME, null, 1);
    }
    /* Create a table */
    @Override
    public void onCreate(SQLiteDatabase db) {
        String createTable = "CREATE TABLE " + TABLE_NAME + " (ID INTEGER PRIMARY
KEY AUTOINCREMENT, NAME TEXT, EMAIL TEXT, DEGREE TEXT)";
        db.execSQL(createTable);
    }
    /* Drop the table if it exists */
    @Override
    public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
        db.execSQL("DROP TABLE IF EXISTS " + TABLE_NAME);
        onCreate(db);
    }
    /* Insert data into the table */
    public boolean addData(String name, String email, String degree) {
        SQLiteDatabase db = this.getWritableDatabase();
        ContentValues contentValues = new ContentValues();
        contentValues.put(COL2, name);
        contentValues.put(COL3, email);
        contentValues.put(COL4, degree);
```

```
        long result = db.insert(TABLE_NAME, null, contentValues);
        return result != -1;
    }
    /* Get all data from the table */
    public Cursor showData() {
        SQLiteDatabase db = this.getWritableDatabase();
        Cursor data = db.rawQuery("SELECT * FROM " + TABLE_NAME, null);
        return data;
    }
    /* Update data in the table */
     public boolean updateData(String id, String name, String email, String
degree) {
        SQLiteDatabase db = this.getWritableDatabase();
        ContentValues contentValues = new ContentValues();
        contentValues.put(COL1, id);
        contentValues.put(COL2, name);
        contentValues.put(COL3, email);
        contentValues.put(COL4, degree);
        db.update(TABLE_NAME, contentValues, "ID = ?", new String[] {id});
        return true;
    }
    /* Delete data from the table */
    public Integer deleteData(String id) {
        SQLiteDatabase db = this.getWritableDatabase();
        return db.delete(TABLE_NAME, "ID = ?", new String[] {id});
    }
}
```

MainActivity.java: This class collects the data user entered on the interface and passes them as parameters to relevant functions. The functions will continue to call the corresponding functions in the DatabaseHelper class to insert, view, update, or delete the data.

```
package com.example.getuserinput;
import androidx.appcompat.app.AlertDialog;
import androidx.appcompat.app.AppCompatActivity;
import android.database.Cursor;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {
    DatabaseHelper database;
    Button addData, viewData, updateData, deleteData;
    EditText eName, eEmail, eDegree, eID;
    /* Create variables connected to the interface */
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        database = new DatabaseHelper(this);
        eID = (EditText) findViewById(R.id.inputID);
        eName = (EditText) findViewById(R.id.inputName);
        eEmail = (EditText) findViewById(R.id.inputEmail);
        eDegree = (EditText) findViewById(R.id.inputDegree);
        addData = (Button) findViewById(R.id.addBtn);
        viewData = (Button) findViewById(R.id.viewBtn);
        updateData = (Button) findViewById(R.id.updateBtn);
```

```
        deleteData = (Button) findViewById(R.id.deleteBtn);
        addData();
        viewData();
        updateData();
        deleteData();
    }
    /* Add data to the database when user clicks on the button */
    public void addData() {
        addData.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                String name = eName.getText().toString();
                String email = eEmail.getText().toString();
                String degree = eDegree.getText().toString();

                boolean insertData = database.addData(name, email, degree);

                if (insertData) {
                    Toast.makeText(MainActivity.this, "Successfully inserted",
Toast.LENGTH_LONG).show();
                } else {
                        Toast.makeText(MainActivity.this, "Failed to insert",
Toast.LENGTH_LONG).show();
                }
            }
        });
    }
    /* Display data from the database when user clicks on the button */
    public void viewData() {
        viewData.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Cursor data = database.showData();

                if (data.getCount() == 0) {
                    display("Error", "No data to display");
                    return;
                }
                StringBuffer buffer = new StringBuffer();
                while (data.moveToNext()) {
                    buffer.append("ID " + data.getString(0) + "\n");
                    buffer.append("Name: " + data.getString(1) + "\n");
                    buffer.append("Email: " + data.getString(2) + "\n");
                    buffer.append("Degree: " + data.getString(3) + "\n");
                    display("All stored data:",buffer.toString());
                }
            }
        });
    }
    /* Throw a message when action is completed */
    public void display(String title, String message) {
        AlertDialog.Builder builder = new AlertDialog.Builder(this);
        builder.setCancelable(true);
        builder.setTitle(title);
        builder.setMessage(message);
        builder.show();
    }
    /* Update data in the database when user clicks on the button */
    public void updateData() {
```

```
        updateData.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                int temp = eID.getText().toString().length();
                if (temp > 0) {
                                                        boolean     update     =
database.updateData(eID.getText().toString(),       eName.getText().toString(),
eEmail.getText().toString(), eDegree.getText().toString());
                    if (update) {
                            Toast.makeText(MainActivity.this, "Data updated",
Toast.LENGTH_LONG).show();
                    } else {
                        Toast.makeText(MainActivity.this, "Failed to update",
Toast.LENGTH_LONG).show();
                    }
                } else {
                        Toast.makeText(MainActivity.this, "Please enter ID",
Toast.LENGTH_LONG).show();
                }
            }
        });
    }
    /* Delete data from the database when user clicks on the button */
    public void deleteData() {
        deleteData.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                int temp = eID.getText().toString().length();
                if (temp > 0) {
                                                        Integer     delete     =
database.deleteData(eID.getText().toString());
                    if (delete > 0) {
                            Toast.makeText(MainActivity.this, "Data deleted",
Toast.LENGTH_LONG).show();
                    } else {
                        Toast.makeText(MainActivity.this, "Failed to delete",
Toast.LENGTH_LONG).show();
                    }
                } else {
                        Toast.makeText(MainActivity.this, "Please enter ID",
Toast.LENGTH_LONG).show();
                }
            }
        });
    }
}
```

Video demonstration is available at *https://github.com/AndrewTrieu/apitest.*

The XML file *activity_main.xml* is used in the construction of the app's user interface as seen on the first page. I decided not to put the file here because there was not enough space, and it was created using Android Studio Layer Editor, not through coding. The file is available at the GitHub repository mentioned above.