**Student Database Overview**

Michael Maye

Colorado State University Global

CSC475: Platform-Based Development

Dr. Muhammad A Khan

07/16/2023

**Student Database Overview**

The submitted application is a simple database that holds information on students. It offers essential functions such as listing all students in the database, searching for and updating current entries, as well as adding and removing them. The following overview will explore each function and how they work together to create the required user experience.

**Overview**

Upon opening, the application displays a RecycleView containing a series of CardViews’ that each hold a different student’s first name, last name, grade, and id. The main view contains a bar at the bottom that holds 4 ImageButtons. each button calls an AlertDialog that perform different functions. the first button searches for a student by first and last names or id and returns the students CardView in the alert dialog, the second searches for a student like the first returns the CardView along with a button that when clicked allows the user to enter new information to update a stored student with, the third allows the user to enter a new student to add them, and the fourth searches for a student and allows the user to delete that student.

**Main Activity**

The MainActiviy class holds the bulk of functionality of the application. This class is responsible for loading the applications initial UI which contains a RecycleView that will hold Cardview representing each student in the application and a pseudo-toolbar at the bottom of the screen that holds options for the user. Once loaded, the class calls on methods from the applications database to return an array of Student objects, then builds the CardViews for each entry, and displays them to the user in vertical list form.

The toolbar at the bottom of the screen holds four ImageButtons, each acts as an option to the user. Each button loads an AlertDialog page that offers different ways to input information. The first is the Search option, which allows the user to input First and Last name, or Student ID, then returns a CardView directly into the AlertDialog displaying the searched students information. The second Allows a user to search for a student, then displays that students information along with a button that brings the user back to the original dialog where the students information is automatically loaded into EditText views. Here the user can edit the students information and update the student in whatever way they need to, however they cannot have two students with the same ID. The third button allows a user to add a new student by adding their information into the AlertDialog then add the student to the database as long as the student ID is unique. The fourth option allows a user to search for a student, then remove that student from the database.

**Student Class**

The student class enables the system to build ‘Student’ objects that contain a First and Last name, a grade, and a Student ID. This class also has getters and setters for these fields. The Student object is used to build and store data on students in the database, as well as retrieve and display information to the user.

**StudentDataContract Class**

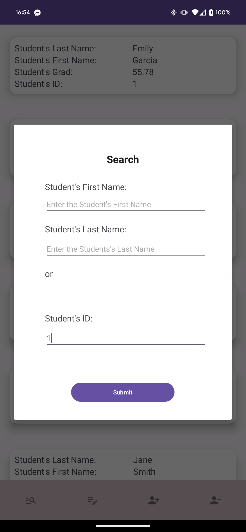
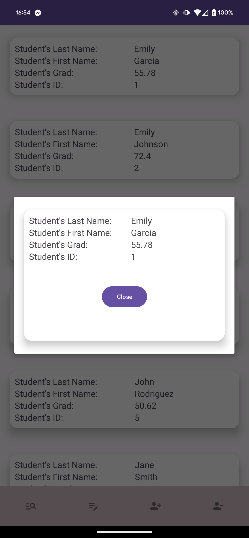
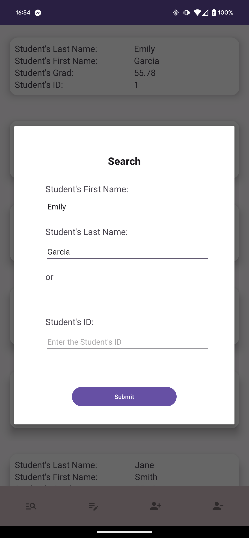
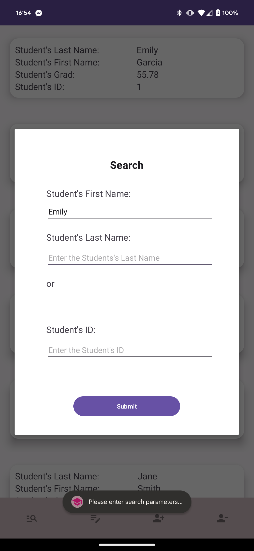
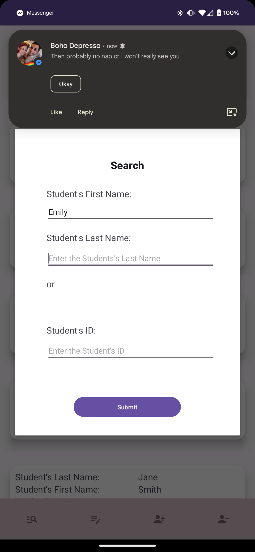
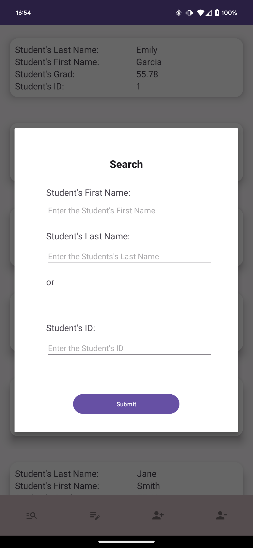
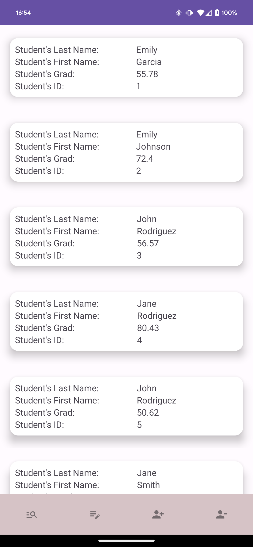
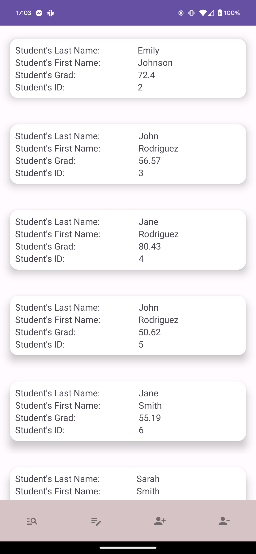
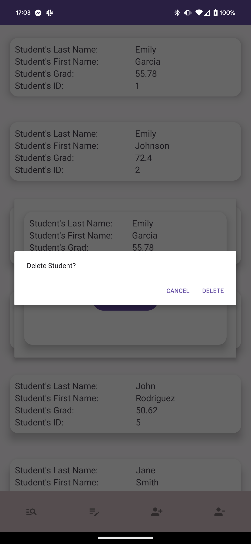
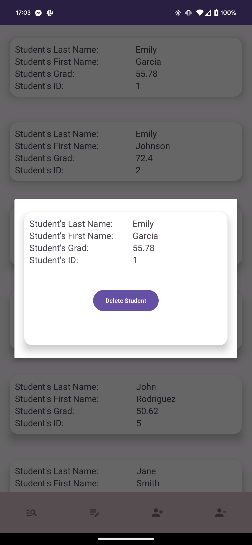
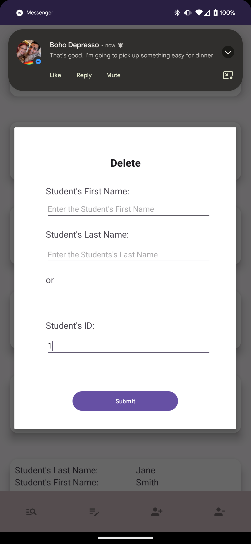
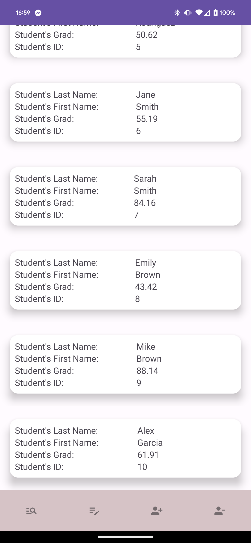
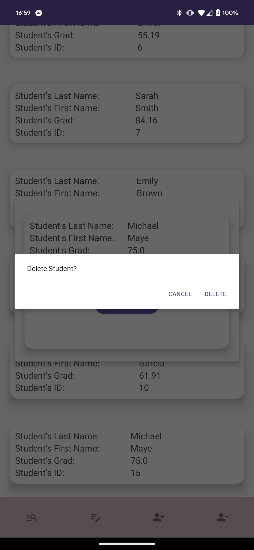
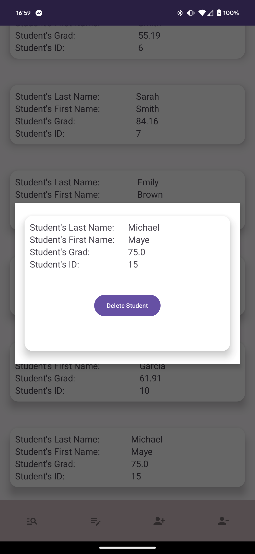
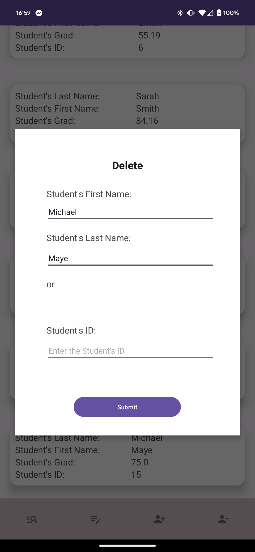
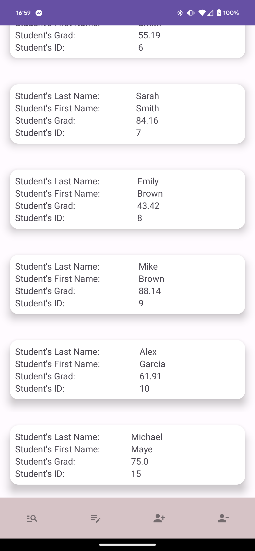
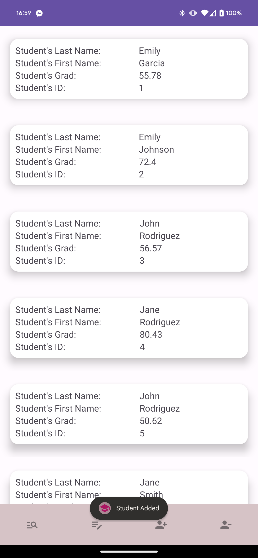
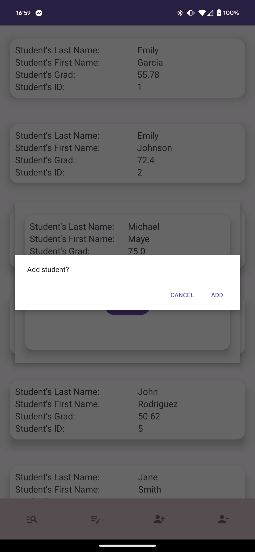
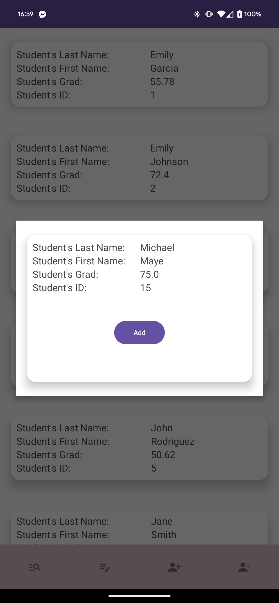
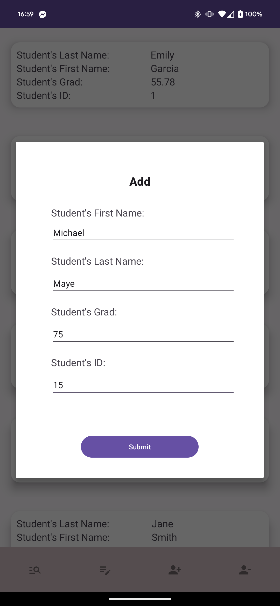
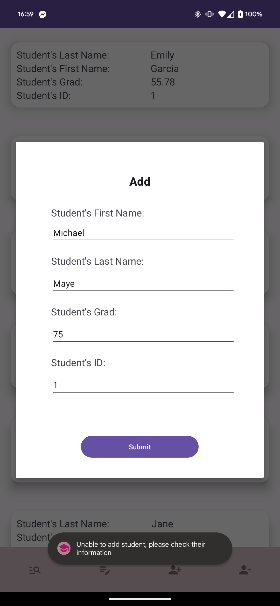
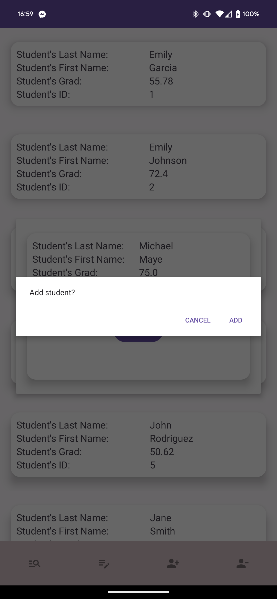
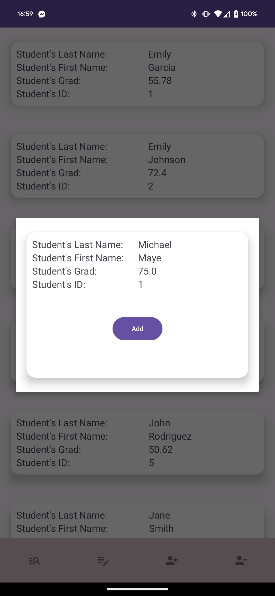
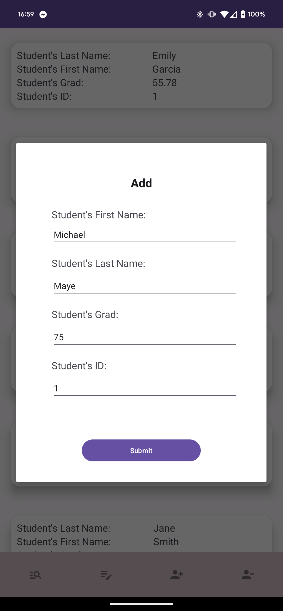
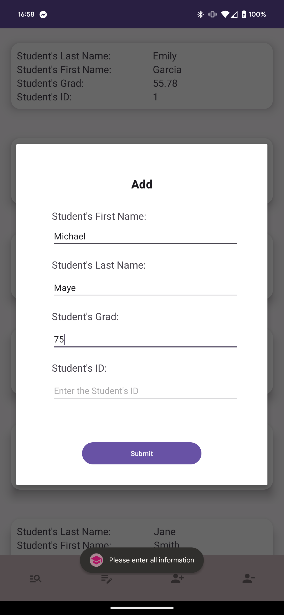
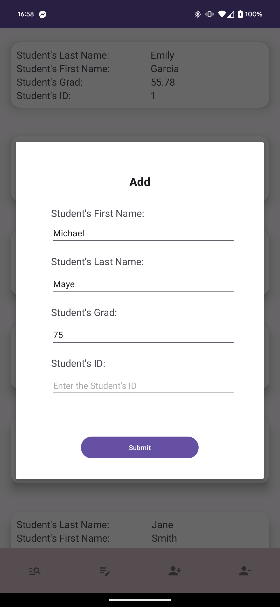
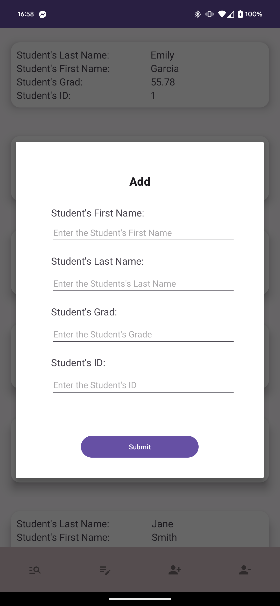
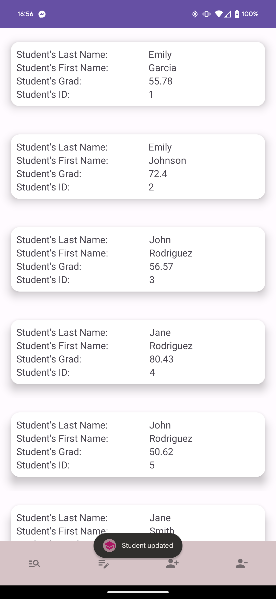
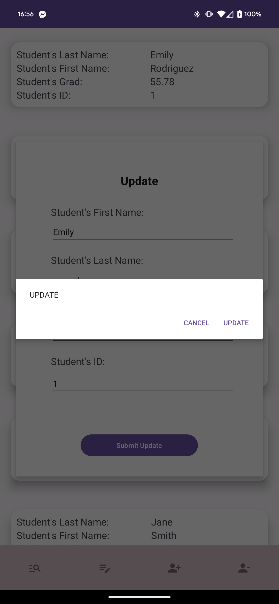
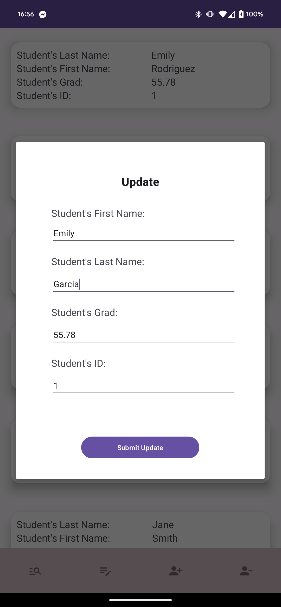
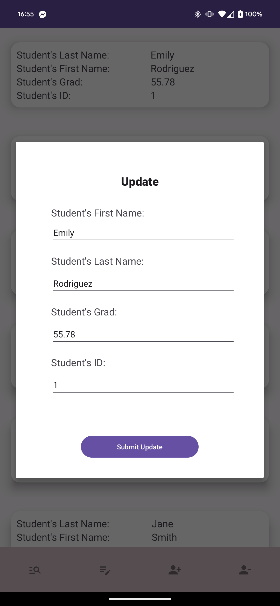
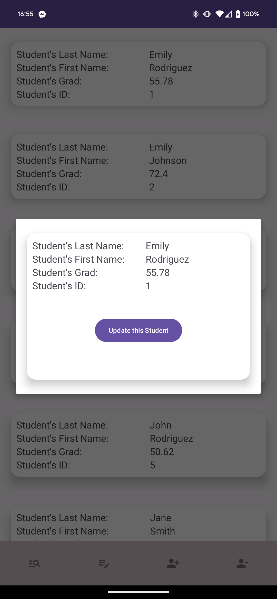
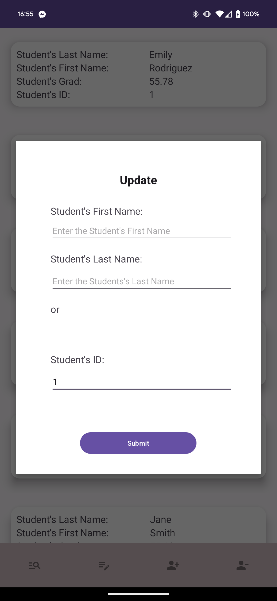
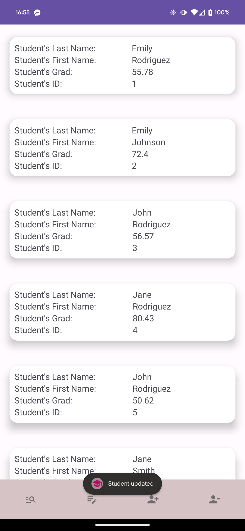
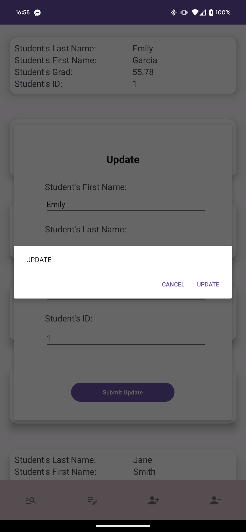
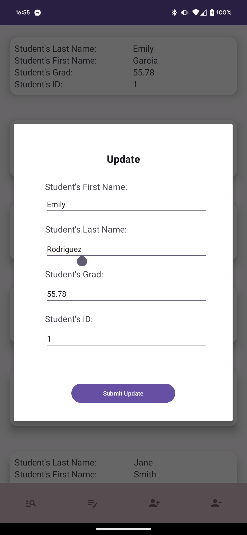
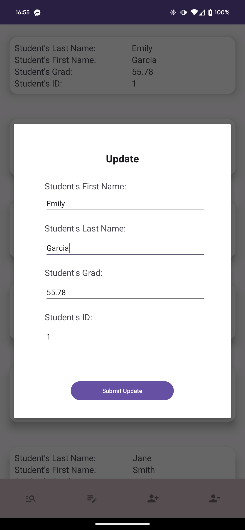
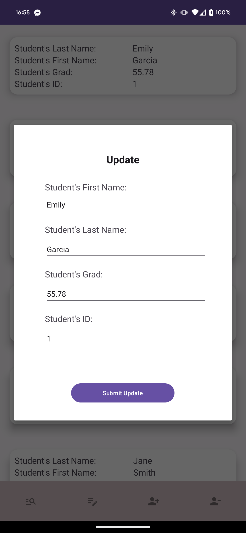
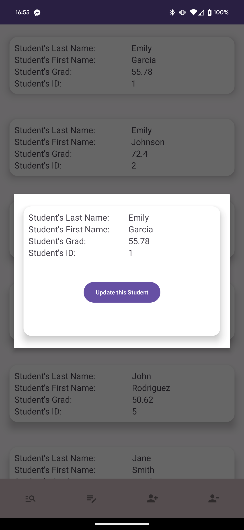
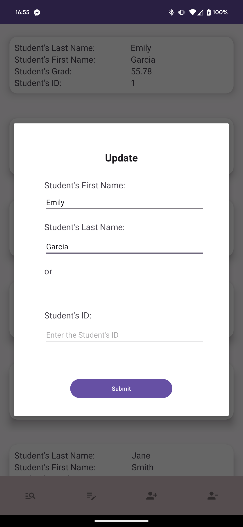
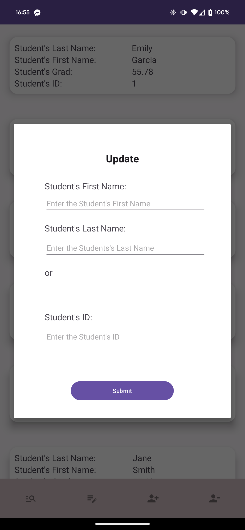
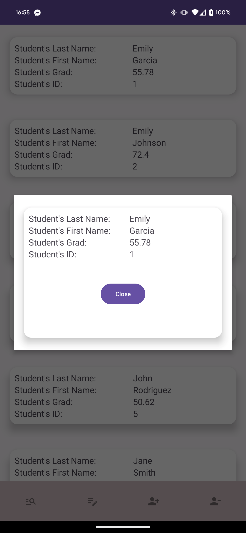
The StudentContractClass offers functionality to the applications SQLite database. This class defines the schema and offers a nested class called StudentDataEntry which defines the tables and column names along with methods to create and delete the tables. In addition, it contains another nested class called StudentDataHelper that extends SQLiteOpenHelper which allows public calls to add, delete, and update entries in the database.

**RandomStudent class**

This class generates an array of Student objects that the system uses to create an initial database for testing purposes only.

**Figure 1**

*Screenshots of application:*

**



**Conclusion**

Throughout this overview, the various components that make up the application have been explored, including the test component, question component, user component, and test layout component. Each component is critical in creating a seamless and interactive user experience. The application successfully educates users on traffic signs by leveraging the Test, Question, and User classes along with the well-structured test layout..

**References**

*GeeksforGeeks. (2022, November 16). CardView using Recyclerview in Android with example. GeeksforGeeks. https://www.geeksforgeeks.org/cardview-using-recyclerview-in-android-with-example/*

*Google. (2023a, March 13). Save data using sqlite&nbsp; :&nbsp; android developers. Android Developers. https://developer.android.com/training/data-storage/sqlite*

*Google. (2023b, April 12). SQLiteDatabase&nbsp; :&nbsp; android developers. Android Developers. https://developer.android.com/reference/android/database/sqlite/SQLiteDatabase*

*SQLite android bindings. RSS. (n.d.). https://www.sqlite.org/android/doc/trunk/www/index.wiki*