**CSE4086 Mobile Development For Android Term Project**

****

* **Kadir Hızarcı**

**150116004**

* **Deniz Arda Gürzihin**

**150116065**

* **Ömer Cem Ağaoğlu**

**150115002**

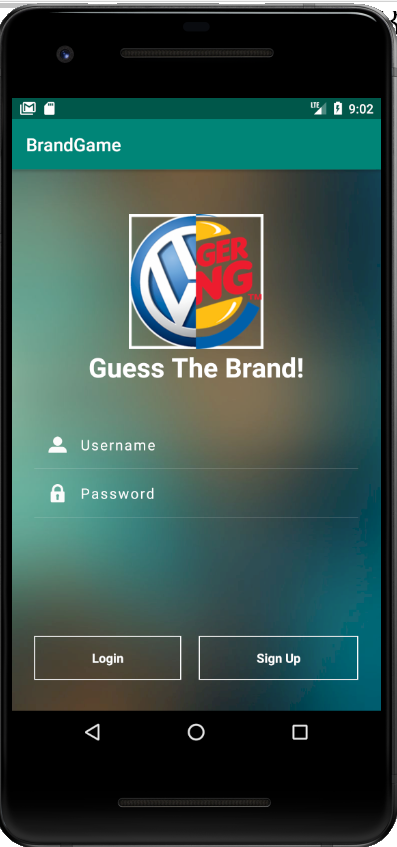
1. **TABLE OF CONTENTS**

**1. Brief Project Design**

**2. Technologies**

**3. How To Play**

1. **Brief Project Design**

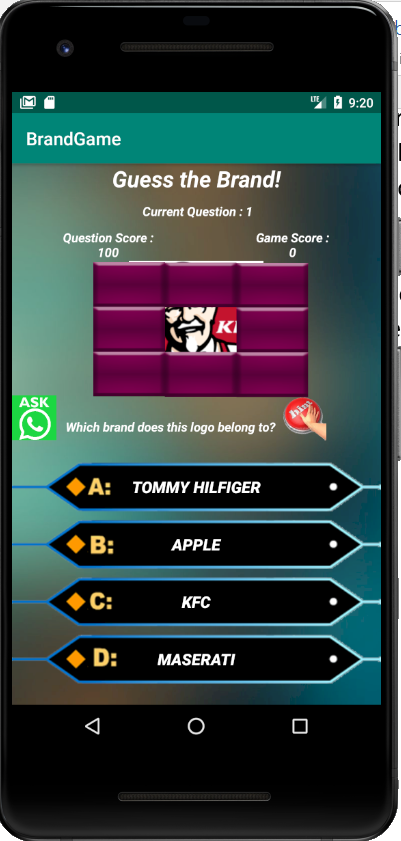
Our project consists of 3 main phases. The log in & sign up phase, database modification phase and the game phase. Upon running the app, user will find themselves at a login screen. User can sign up and login with their credentials later on, saving his data and scores.

Main screen has complex designs with various layouts and a customized password field. After a log in, user will be introduced with a screen in which they are greeted, their high score is shown and through the “plus” menu on bottom right corner, they are given the option to make additions and changes to the game database. Users can select gallery pictures and create their own brand to pop in the game.

“**LOGIN SCREEN AT START”**

When the user clicks “Start Game” button, a new activity will be opened and game will start.

There are 10 questions in the game, each one rewarding a maximum of 100 points. “Question Score” shows the instant score that can be gained from the current question. Clicking a wrong choice will reduce the point gained by 33, whilst pressing the “Hint” button gives a hint to user at the cost of 20 points. Each question has a hint defined to it in the database. Also, clicking on the blocks in front of the picture will remove it at the cost of 5 points. Another option the user has is that they can press the “Ask” button to send the question to one of their friends on WhatsApp. Upon completing the game database is cross-checked and the user’s score is recorded to high score table, if it’s high enough.



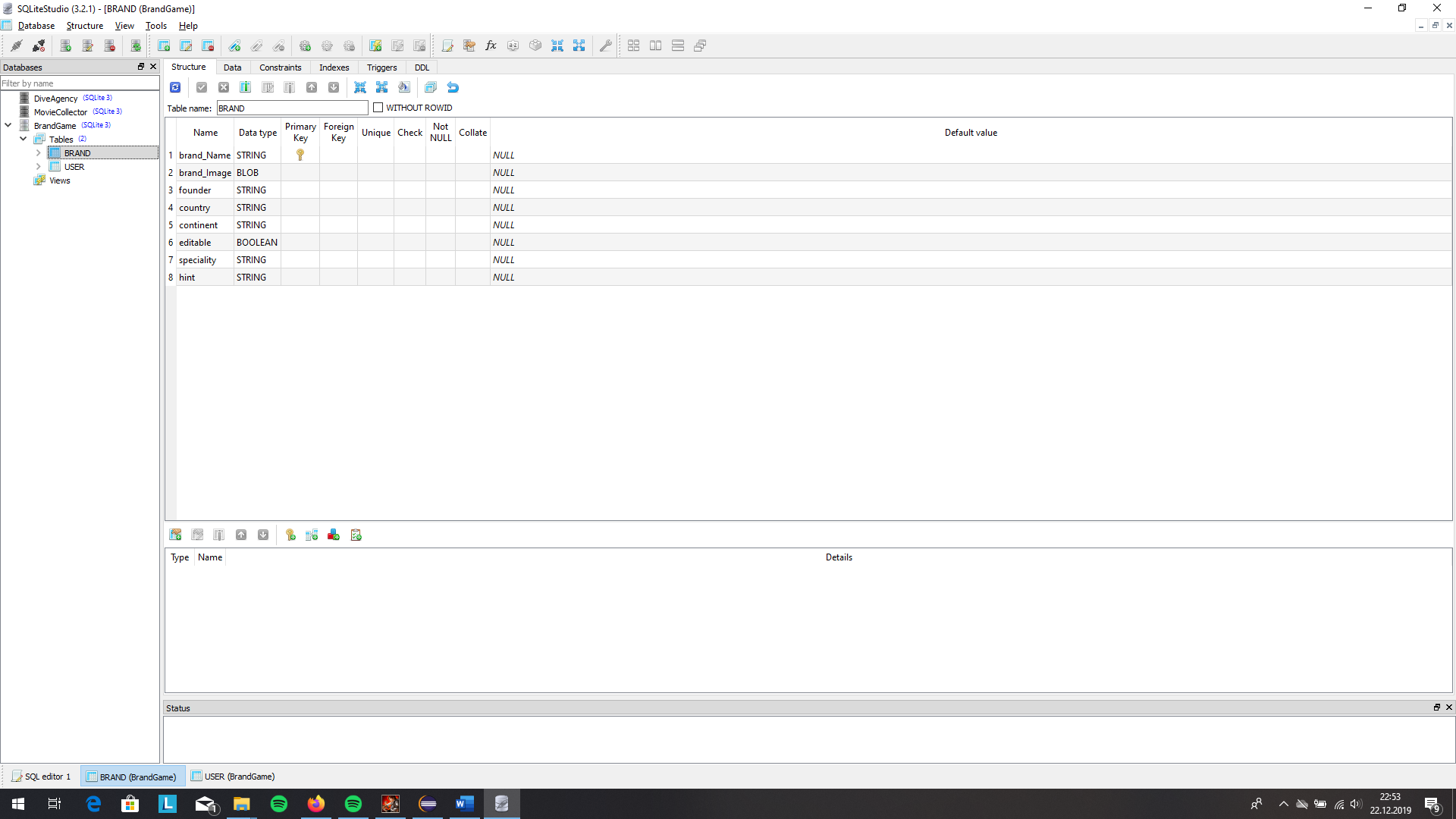
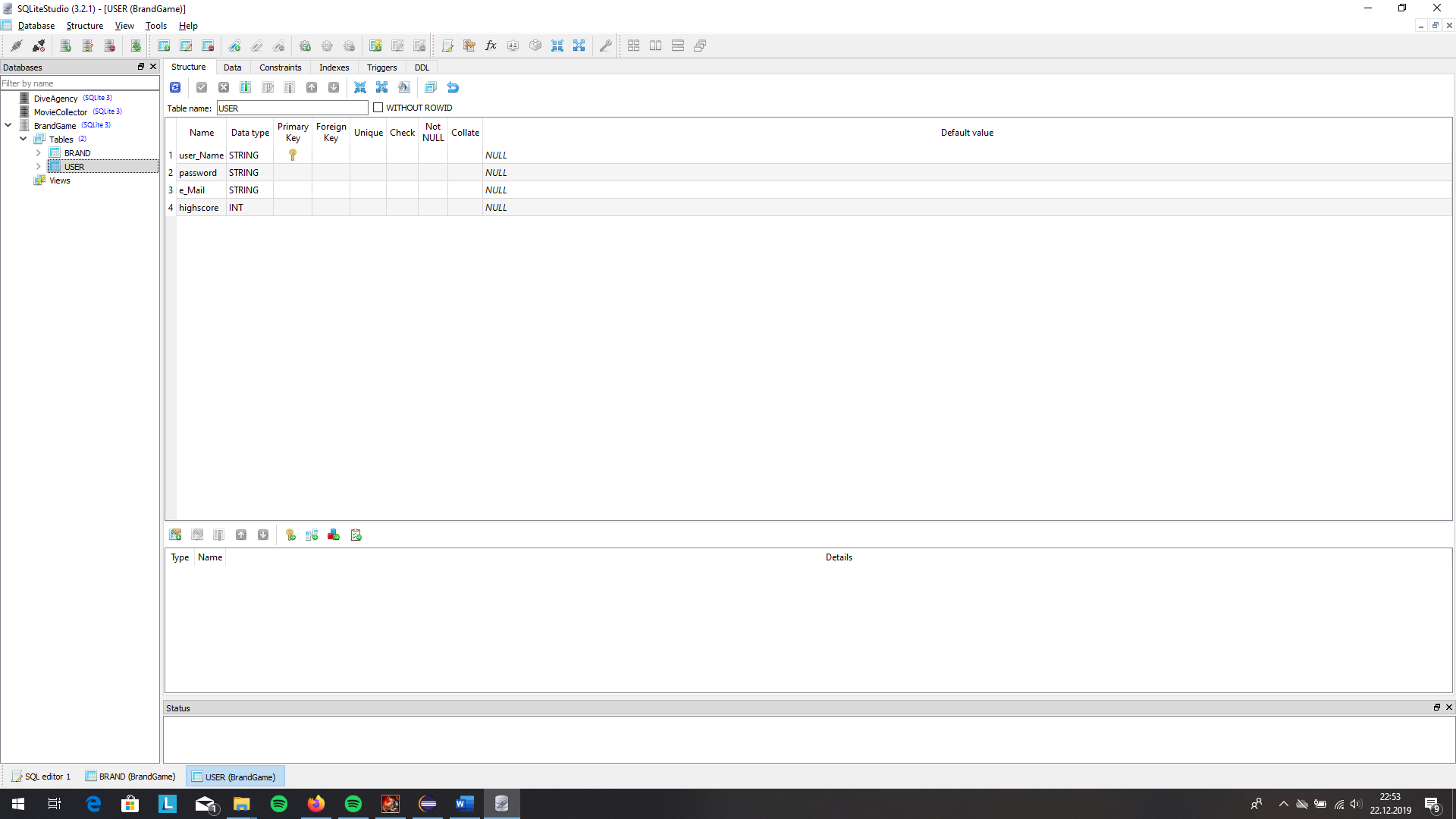
1. **Technologies**

We indicated in our proposal that we would add certain technologies to our game. Important ones are as follows:

* Accessible and real-time responsive database
* Credential saving and login system
* Inter-application multimedia sharing (WhatsApp)
* Multiple factored game design, high score saving

**Accessible and real-time responsive database & Login system:**

On our app, we need to store information. So, we have decided on the use a Database. After some researches, we have discovered that most suitable database system is SQLite Databases for Android. So, we have created a database on SQLite Studio to practice and build the system. On our, system we have two tables. One table to hold user information, one table for hold the brand information. They have the name USER and BRAND in order.

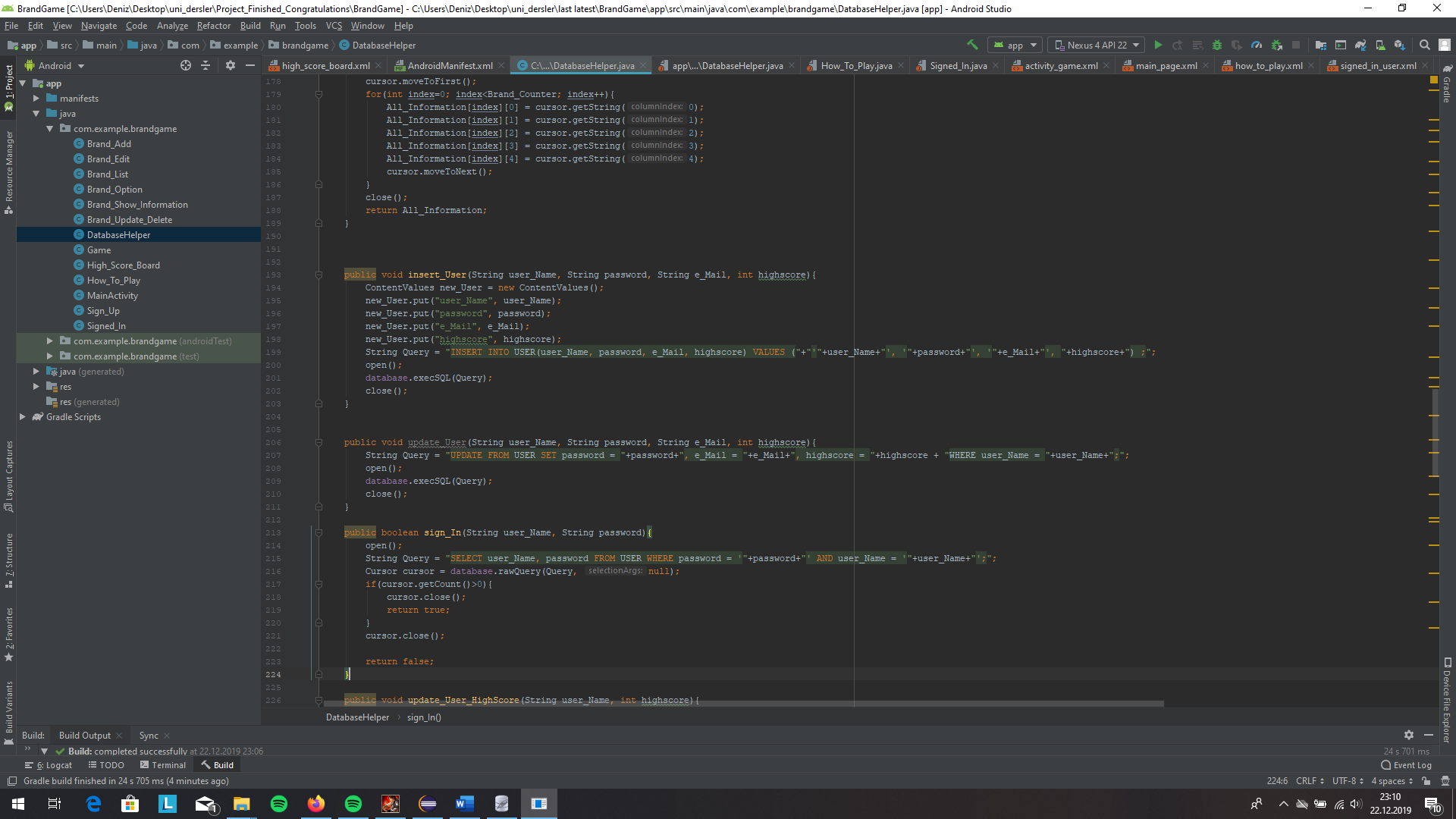
 

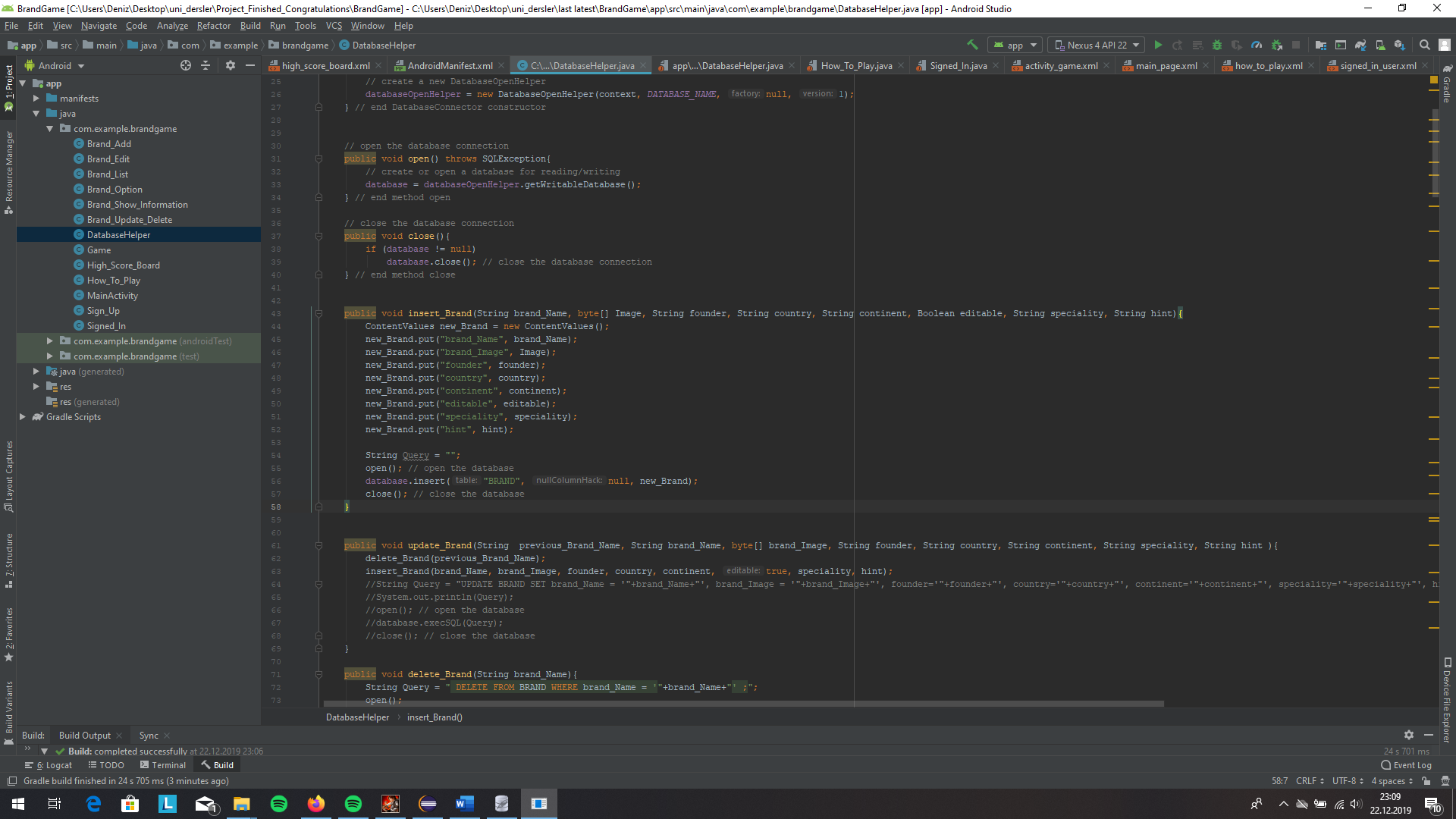
On the USER table, we are holding the User name, password, e-Mail and high score of the User. In here, the unique key is user\_Name, because on every game, one user name can be picked by exactly one user.

On the BRAND table, we are holding the brand name, brand logo, founder , the established country and continent, the speciality, hint for user to answer question easily and finally editable option. We have add editable field, because we want some of the brand as our default brands and not editable from user. In this table, unique key is brand name because there is exactly one brand with that name on the world, none of the brand can be use others brand name.

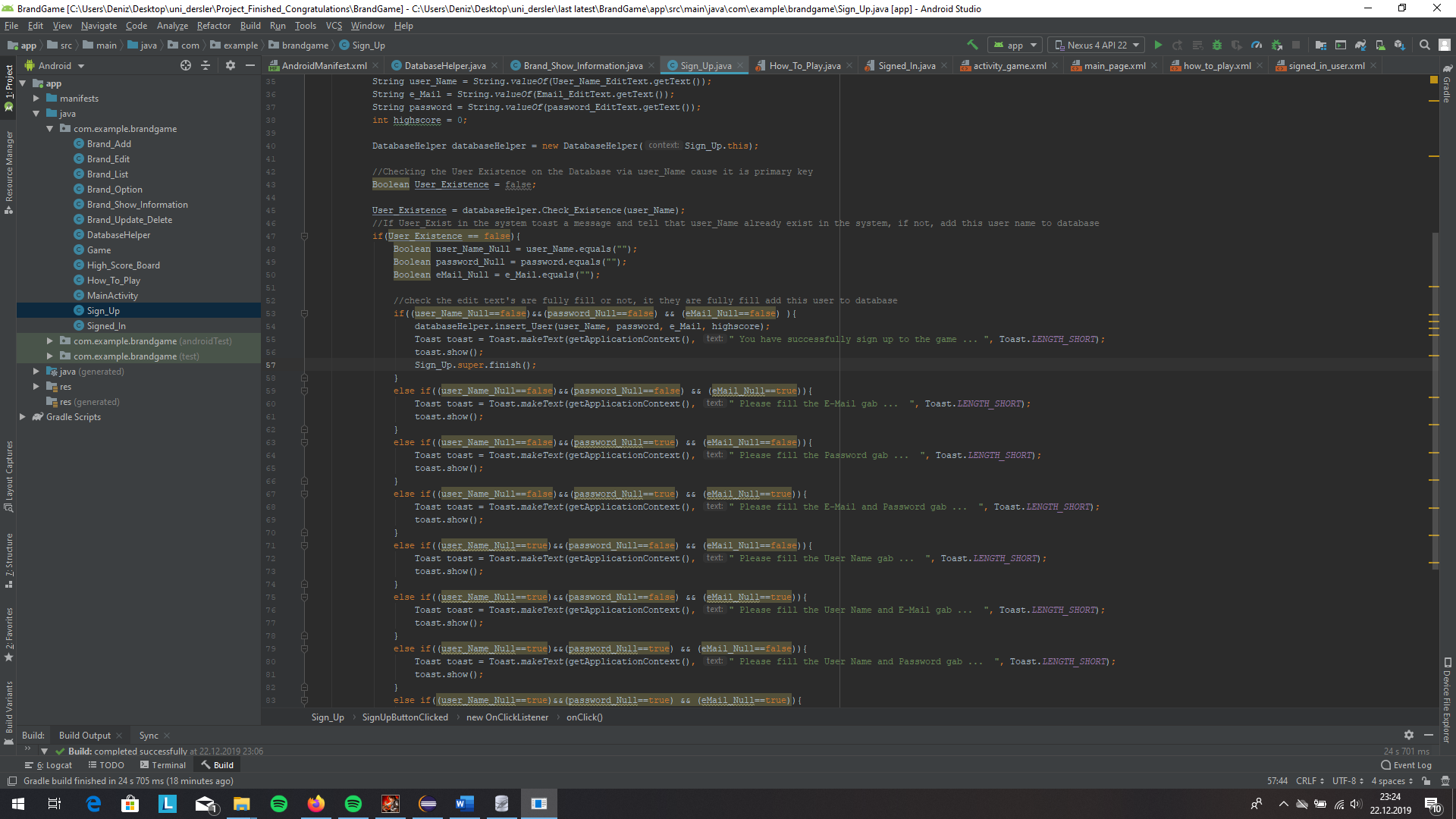
Then, we have implemented this databases on Android Studio environment with a class which name is DatabaseHelper and we have used this class to interact with database and make operations on it. Now, I will talk about most of the functionalities we have added our database system to use on the game.

Insert and Update operations for tables are the basics ones. For insertion and update to database, we have simply use Queries and execute them. We have take the necessary information and organize them as Query and with database.execl(“Query”) function execute them. Here is how we implement them.



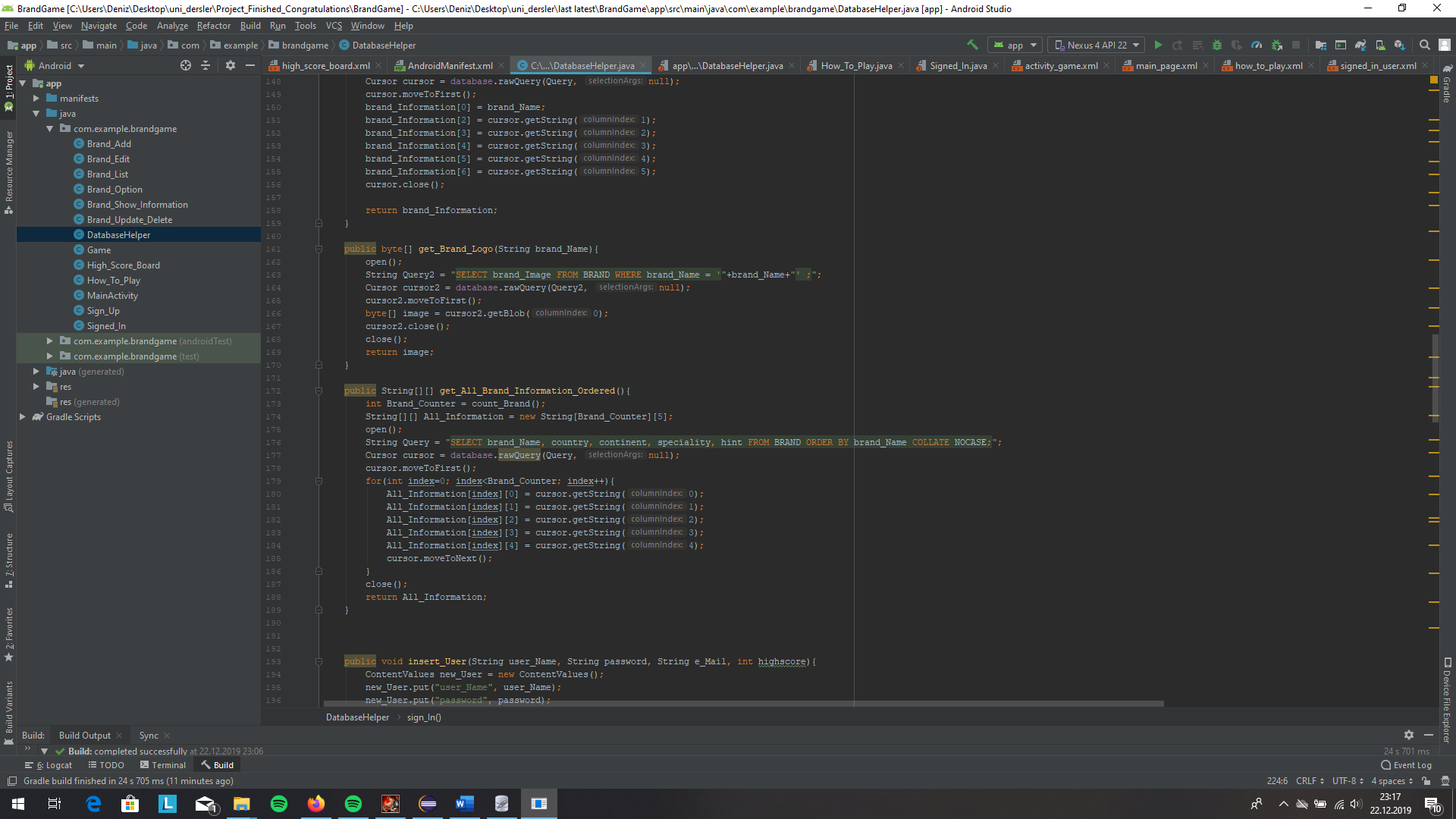


Such as, if a user want to sign up, he/she will provide the user name, password and e-Mail. This is, how we implemented. We take the information from the edit texts, then database object to done operation and the function insert. We, have also used another database function on here as User\_Existence, this functions checking the user existence with user\_Name and if it exist returns true and if the database has not got a user name like this return false.



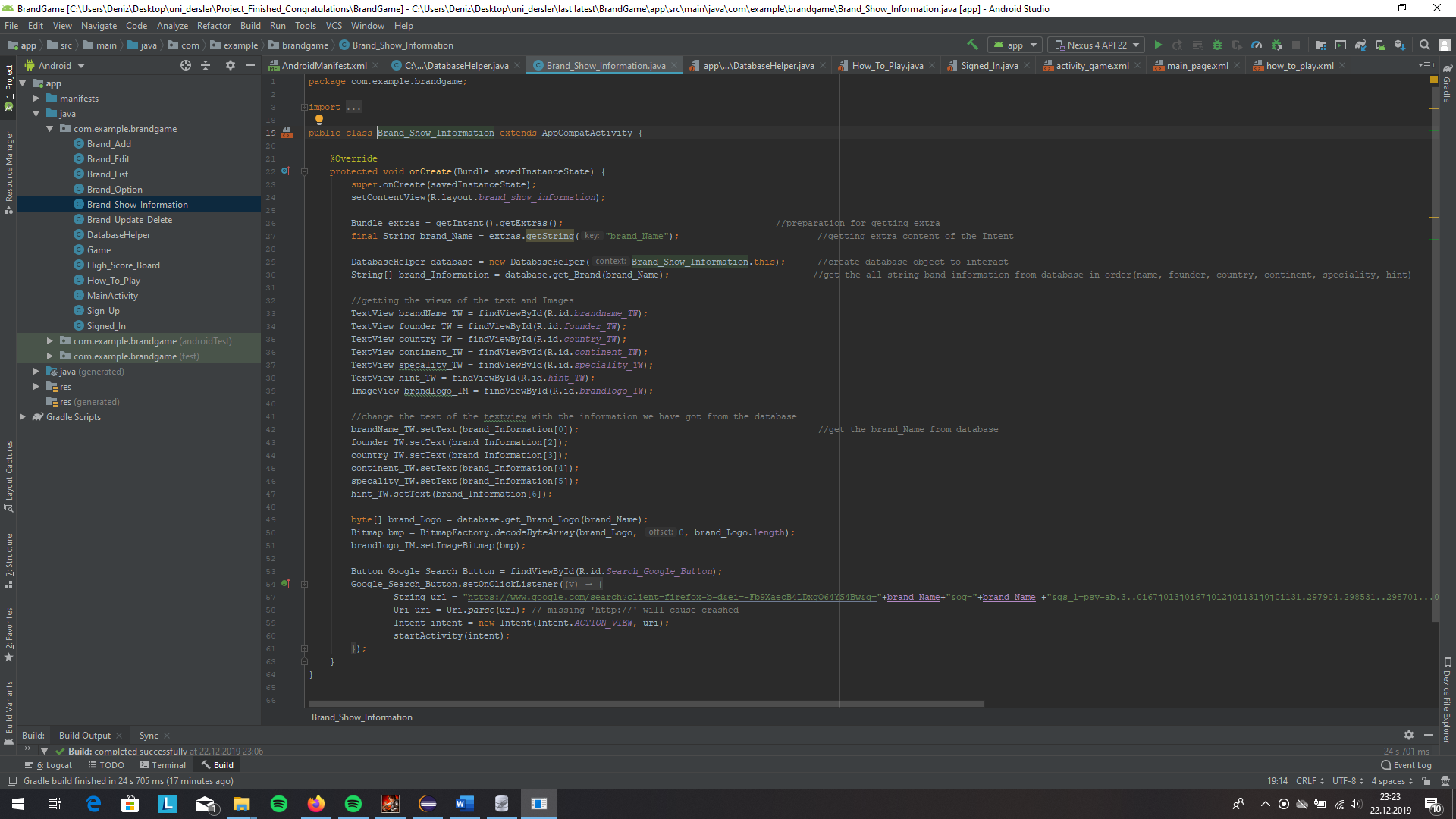
Later, we have focused on more complex database operation such as retrieving every information for Brand except Image, cause Images can be stored on the database as BLOB where the other can be stored as String.

Here is the one of the most complex, functionality we have used on Android.

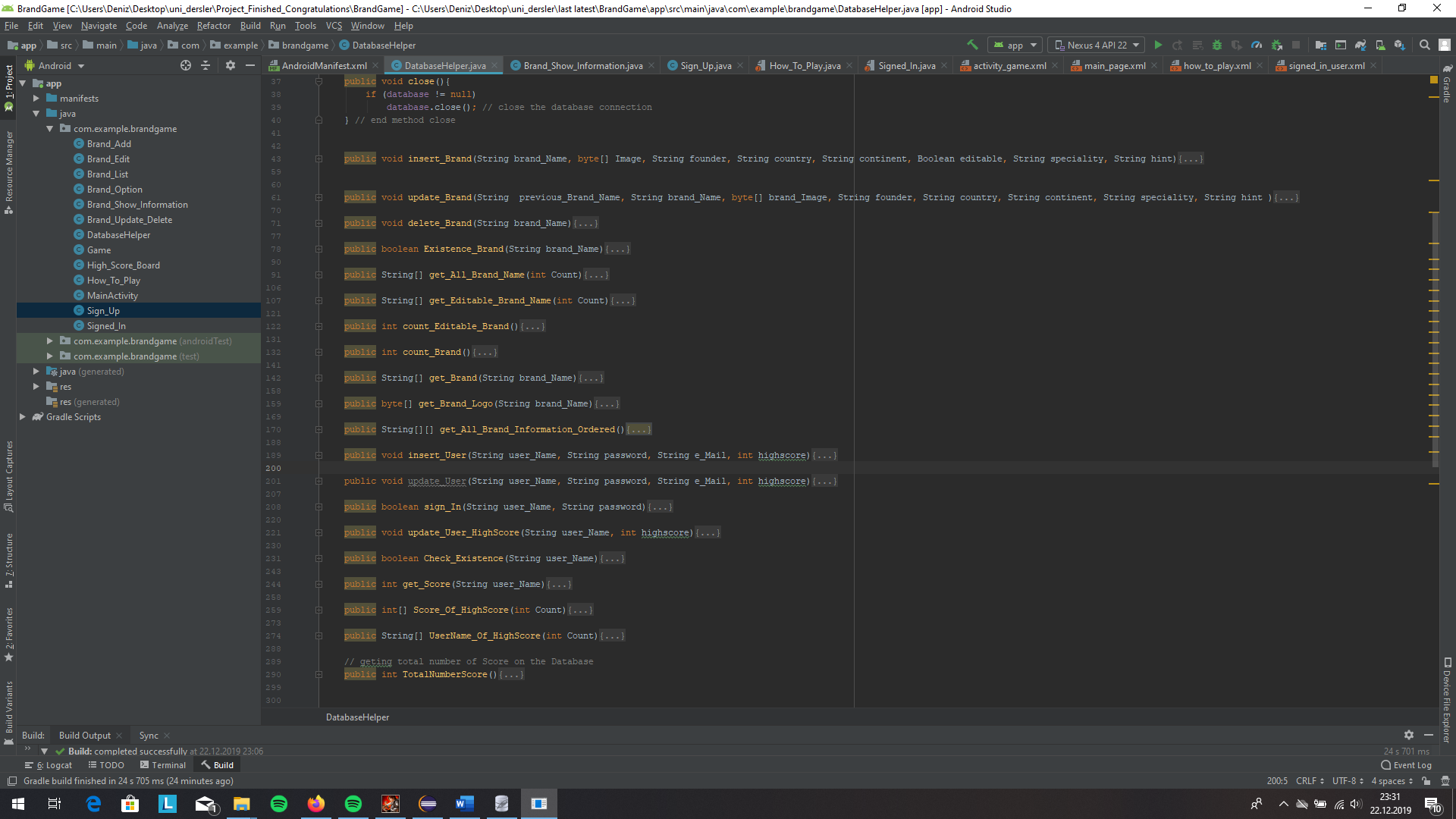


With this function, we are retrieving all of the information of the brand on one 2dimensional array. First dimension is for holding the number of the brand, the seconds for holding the information’s. On this Array, we are holding the brand\_name, country, continent, specality and hint in the order. One of the usage of this functions is showing of all the Brands information in the databases.

Which is one of the main function for our game. Here is the implementation of the showing the all information and assigning them one by one text view.

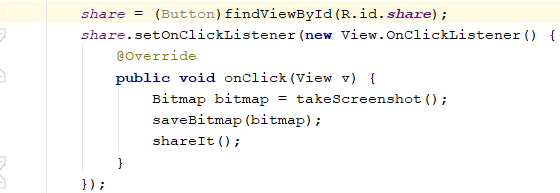


Finally, I want to show all the functions we have used with their name, their functionalities are corelated with their names.

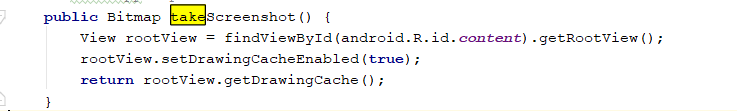


**Inter-application multimedia sharing:**

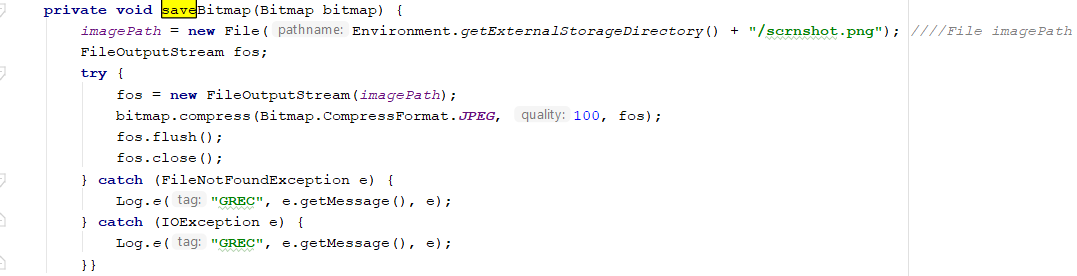
When the user clicks on “ask” button during the game, first, Android system is prompted to take a screenshot of the current screen and save it to a bitmap variable.



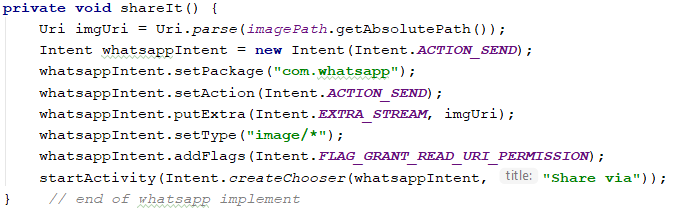
takeScreenshot method sets view of the current content, sets drawing cache to true, and returns it as a Bitmap object



saveBitmap function is tasked to save the Bitmap variable to external storage in the name “scrnshot.png”. FileOutputStream is used to set image specs and some error catchings are also added.



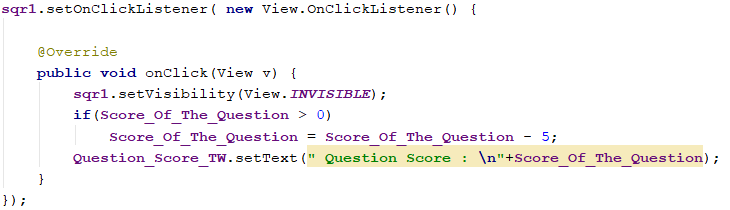
Finally, shareIt function gets the Uri of the imagePath variable (where the screenshot is saved) and creates a send intent. Image is then put into intent with putExtra method, type specified so WhatsApp will accept it. Then the intent is executed by startActivity method.



**Multiple factored game design:**

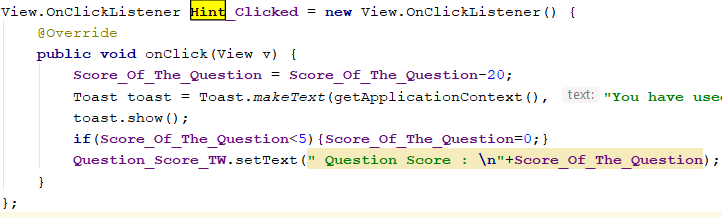
Game score is influenced by various factors. Removed blocks, asked hints or simply a wrong answer reduces the gained points. Here is a look at how they are implemented:



Blocks are originally buttons with a customized drawable field. Upon clicking on each button, onClick function is initiated, and code below is run.

Above image, the clicked button’s visibility is changed and question score is reduced by 5. Then the updated question score is set as the new text.

Looking at the implementation of the hint button, we used a Toast message to appear on the screen.

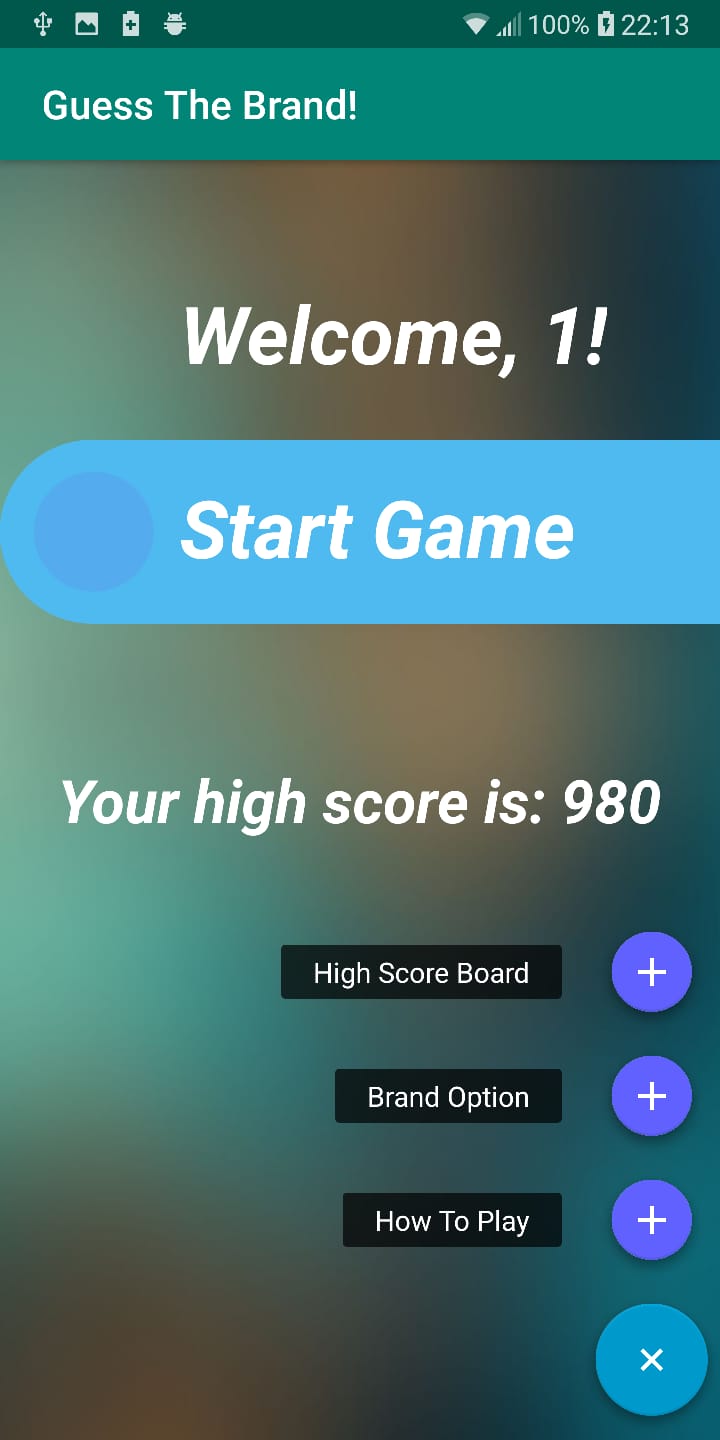


**3. How to Play?**

To play our game, Guess the Brand, first of all you need to sign up to the system on the opening screen. Once your account is created, you can log in. A successful login screen is shown on fig-1. Clicking on “Start Game” button will initiate the start. Main game screen will be like fig-2.

User will start each question with a maximum of 100 points. As seen in fig-2, game interface consists of a hint button, blocks, ask a friend button and choices. If your choice is the right one you will gain the “question score” and will be prompted to another question. You will lose 33 points for each time your answer is wrong. In case you don’t know the brand, worry not! You can use the hint button to receive a hint, at the cost of 20 points.

We also have a cost free way for you, it is to use the ask a friend button to ask someone of your choosing on WhatsApp! Finally, you can destroy the blocks on the screen by clicking on them at the cost of 5 points. Each session of game consists of 10 questions and you can gain 0 up to 100 points from each question.

 ****

**Fig-1** **Fig-2**