

# **Application “Easy Learn”**

## **Development Summary Report**

**For Lecture Practical Course on Mobile Application  
Development with Android (Lab)**

Developers

Mengru.Ji 21737249 [mengru.ji@stud.uni-goettingen.de](mailto:mengru.ji@stud.uni-goettingen.de)

Junqi.Sun 21737212 [junqi.sun@stud.uni-goettingen.de](mailto:junqi.sun@stud.uni-goettingen.de)

# 1 Introduction

## 1.1 Motivation

The reason that why we come up with this app idea is because of the lack of passion to learn German. So the purpose of developing this application is to make German learning software more suitable for our own needs. For example, we want to use this application to repeatedly practice the German grammar exercises or vocabulary exercises that we have organized by ourselves. As long as the exercises are stored in the file in the prescribed format, this application can get the exercises organized in the file.

## 2 Development Process

### 2.1 Function Specification

The app consists of multiple activities. All the questions, answers and corresponding data will be stored in a SQLite database. The user's account information will be stored in the Room Database. Achievements and Bonus will be unlocked after you reach the goals.

### 2.2 Main Classes of Application

- Signup Activity
- Login Activity
- Main Activity(AlarmReceiver)
- Daily Attendance Activity(SpecialCalendar, SquareGridView, SquareRelativeLayout, CheckInAdapter)
- Mission Activity(BackDialog,TimerDialog)
- Mission Result Activity(MyProgressBar)
- Wrong Mode Activity
- Competition Start Activity
- Competition Activity
- Question Book Activity(TabPagerAdapter, PagerFragment)
- Timer Activity
- Achievement Activity
- DatabaseCreator(User, UserDao)
- MySQLite(Questions)

### 2.3 Plan and Division

Mengru.Ji	Junqi.Sun
Daily Attendance Activity	Login Activity
Mission Activity	Signup Activity
Mission Result Activity	Main Activity
Wrong Mode Activity	Competition Start Activity
Question Book Activity	Competition Activity
Timer Activity	Achievement Activity
MySQLite	DatabaseCreator

## 2.4 Relationship between activities

After running the application, the first thing that runs is the login activity. If the user already has an account, you can log in directly, and the app will jump directly to the Main Activity. Otherwise, the app will jump to the Signup Activity. After successful registration, user can log in to the account and jump to the Main Activity.

Click the button on the main activity interface, the application will jump to the corresponding activities(Check in Activity, Mission Activity, Competition Start Activity, Question Book Activity, Timer Activity, Achievement Activity).

After completing an answer task (in the Mission Activity), the application will jump to the Mission Result Activity. On this interface, user can click the corresponding button to enter the Wrong Mode Activity interface to check the wrong questions and all questions.

After reading the competition rules(in the Competition Start Activity), the user will enter the Competition Activity. Every time the game ends, the user can choose to quit the game(back to Main Activity), try again(Start Competition Activity again), or check the German questions(will jump to Wrong Mode Activity) that appear in this game.

Click the logout button on the action bar of the main activity, it will jump to the login activity, and the user can re-select the account to login.

## 2.5 Function introduction of Activities

### 2.5.1 Login Activity

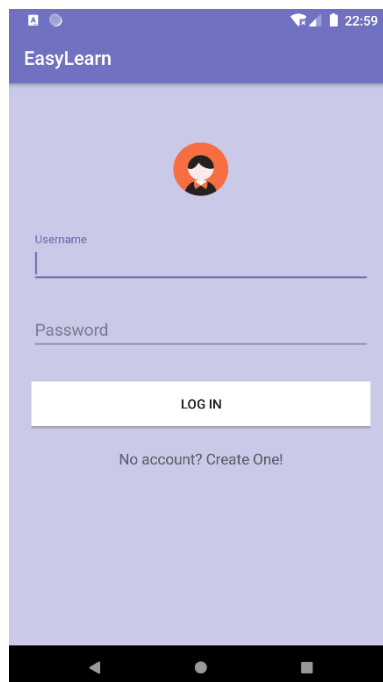


Figure 1

If you already have an account, you can choose to log in directly. After clicking the login button, the application will search in the database based on the user name you entered. If the user name is not registered, an error message will appear, otherwise the application will compare the found password with the password you submitted. If correct, the login is successful, otherwise an error message will appear.

Without an account, you should click the gray font below "No Account? Create One!" and jump to the Signup activity to register.

### 2.5.2 Signup Activity

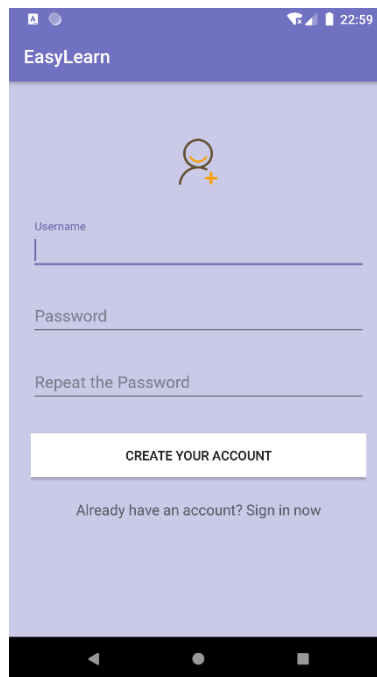


Figure II

Please complete the required fields and then click the register button. If the user name you fill in is not registered, a new data will be created in the corresponding table in the database, otherwise an error message will appear.

## 2.5.3 Main Activity

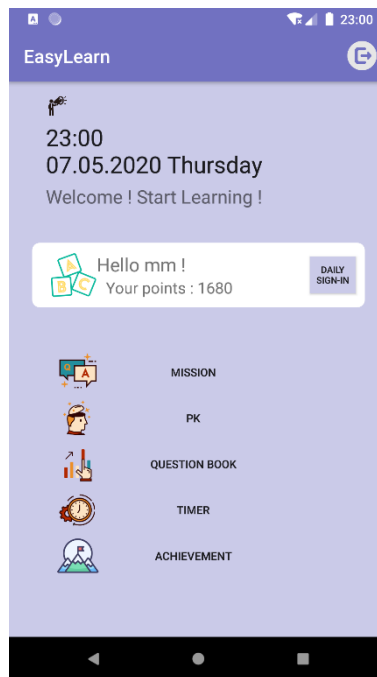


Figure III

On the Main Activity interface, the time will be displayed in real time, and whether you have set a reminder(Black villain with a horn). You can also check your points. Jump to the different activities can be achieved by clicking on corresponding buttons.

## 2.5.4 Daily Attendance Activity

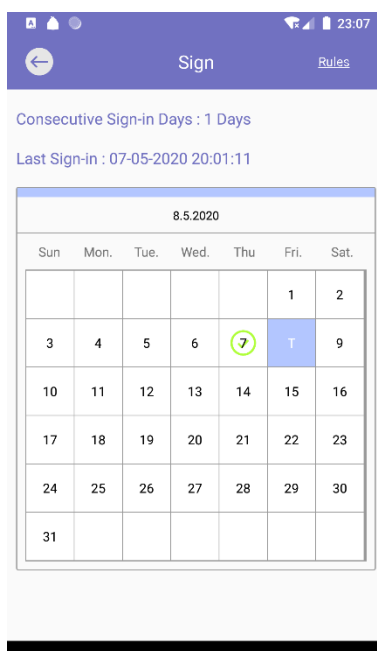


Figure V

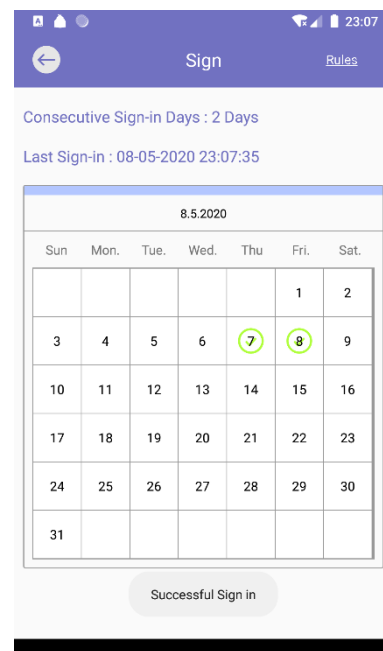


Figure IV

In this activity, you can view the current number of consecutive daily attendance and your last attendance time. In the calendar below, if you have not clocked today, today's date will be marked with a T. Dates that have been clocked will be marked with a green check mark. This calendar will only mark the date of continuous daily attendance. If you interrupt that, the number of days will be cleared. At the beginning of each month, the number of days will also be cleared. The number of consecutive daily attendance days for each user will be recorded in the database.

Class Special Calendar, Square Relative Layout(extends Relative Layout), Square Grid View(extends Grid View) and Check In Adapter(extends Base Adapter) are also implemented to calculate and draw the calendar of the current month.

### 2.5.5 Mission Activity

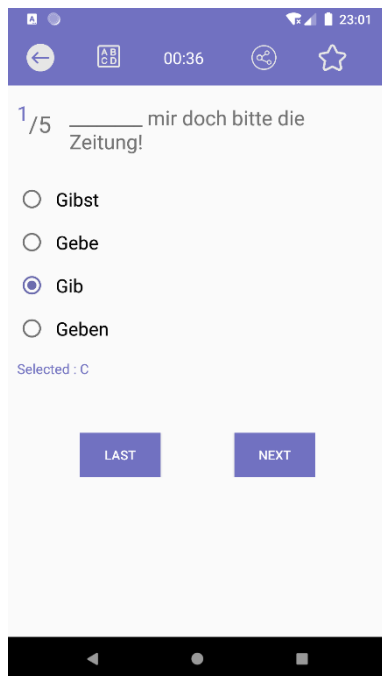


Figure VIII

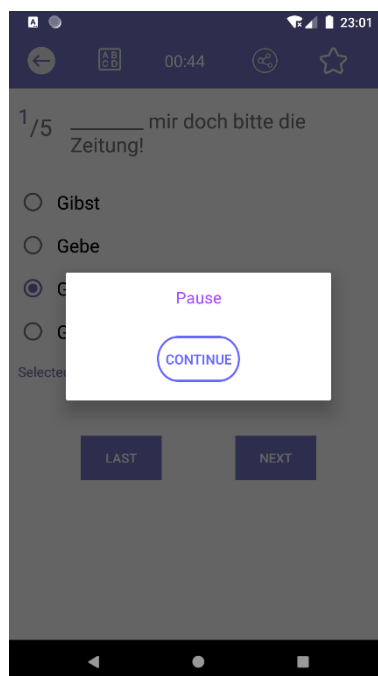


Figure VII

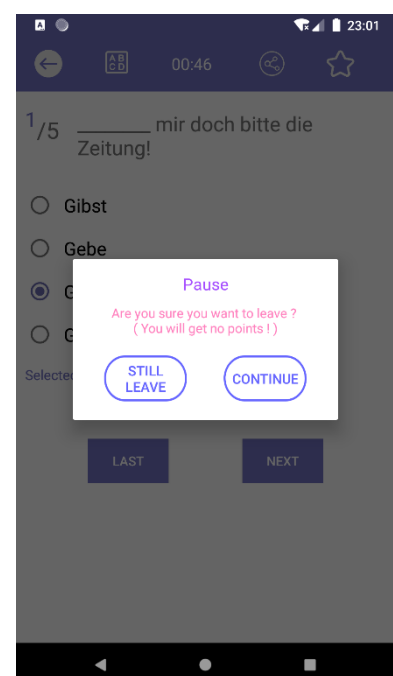


Figure VI

Each time the Mission Activity is run, five questions will be randomly selected from the database. The options are composed of four radio buttons in the radio group, and the selected options will be displayed on the interface. Click the previous or next button to jump to the previous or next question(Figure VIII). If it is already the first question or the last question, a prompt dialog box will appear.

Five buttons implemented by Text View are placed on the Tool Bar. From left to right: back button (return to the main activity), answer card button (unimplemented), stopwatch (real-time display of time spent answering questions), and share button (to share the current question in text form) and the favorite button (add or delete the current question from the favorites).

Clicking the stopwatch and the back button will display the corresponding prompt dialog box(Figure VII and Figure VI), which are implemented by Class "TimerDialog" and Class "BackDialog", and the stopwatch will also pause the timing.

## 2.5.6 Mission Result Activity

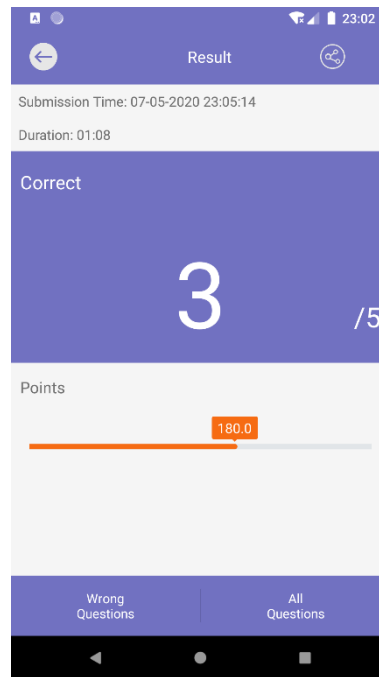


Figure IX

After the quiz is successfully submitted, the app will automatically jump to the Mission Result Activity.

In the current interface, the duration and submission time will be displayed. The correct number of questions will also be displayed in the middle container. Below that, there will be an animated progress bar(implemented by Class MyProgressBar) to show the score of this quiz (300 is the full score, and the score is calculated by multiplying the correct rate) Based on this score, the user information stored in the database will be updated. At the bottom of the interface are two buttons implemented with Text View. Clicking on these two parts will jump to Wrong Mode Activity to view the wrong questions and all the questions and answers of this quiz. A boolean value will be transferred to the next activity to mark whether the user wants to view wrong questions or all questions. If there are no wrong questions, after clicking "Wrong Questions", a dialog box(Alert Dialog) will prompt that all the answers are correct. Click the share button in the toolbar to share the test results in text form, and click the back arrow to return to the main activity.



### 2.5.7 Wrong Mode Activity

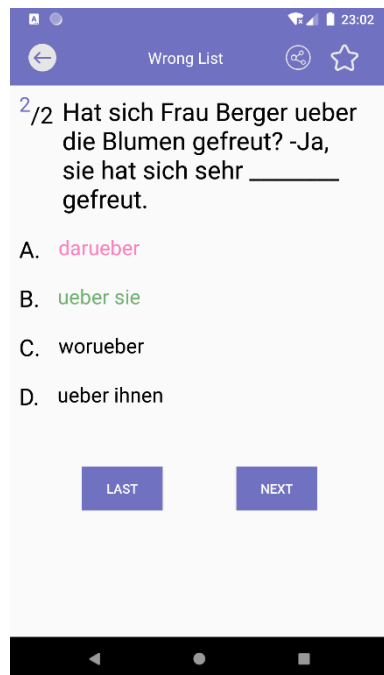


Figure X

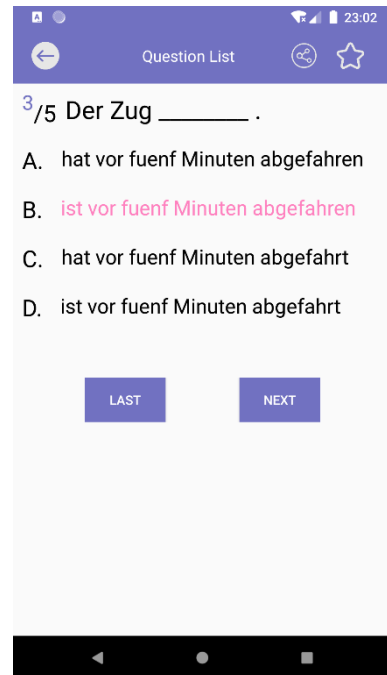


Figure XI

According to the obtained boolean value, it should be judged whether the wrong question layout or all the exercise layout should be displayed. In the wrong question mode(Figure XI), the correct option will be marked in pink and the option selected by the user will be marked in green. In all problem modes(Figure X), only the correct option is marked in pink. In this activity, the current problem can also be shared as a text or bookmarked in the user's favorites. The wrong question will also be automatically added to the user's wrong question book.

### 2.5.8 Competition Start Activity

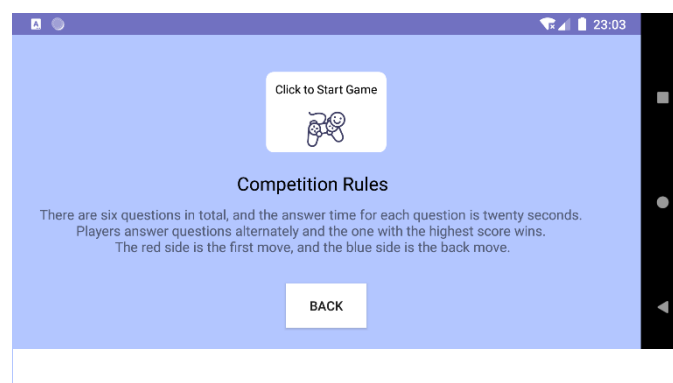


Figure XII

View the rules of the game, click on the picture to start the game(Jump to Competition Activity).

## Application "Easy Learn" Development Summary Report

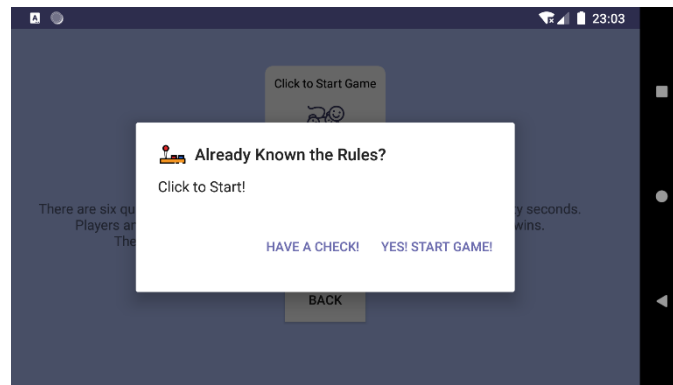


Figure XIII

### 2.5.9 Competition Activity

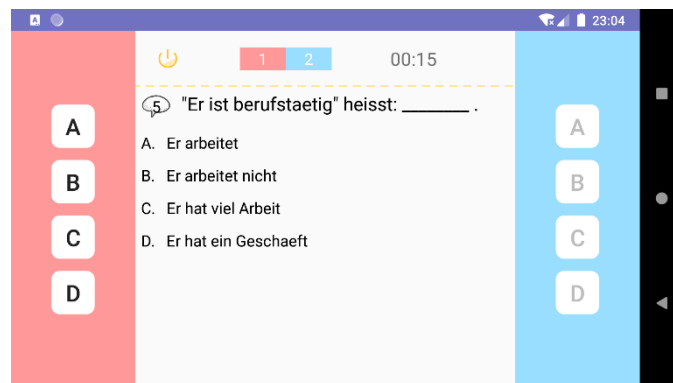


Figure XIV

In this activity, the interface will be divided into three containers. The left and right sides are the answer areas for red and blue. There are four option buttons placed in the container. The middle part is divided into upper and lower sub-parts, the upper part is used to display the current score and the countdown of each round, and the lower part is used to display the details of the current question.

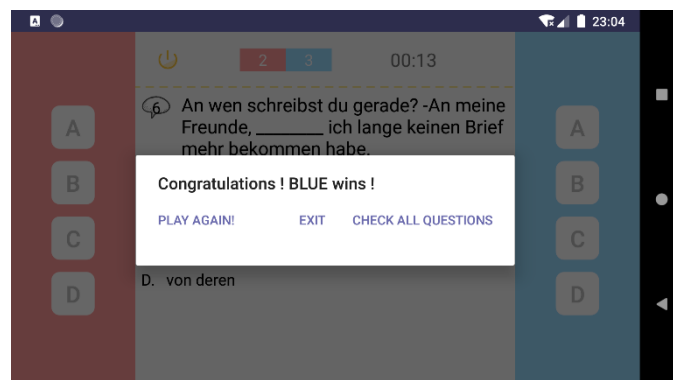


Figure XV

Each game six questions will be randomly selected from the database, and the red and blue sides will take turns to answer the questions. Each time a participant answers a question, the application will determine whether the question has been answered correctly and update the scores of both parties in real time. When all questions have been answered, the score will be used to determine which side wins(Prompt via AlertDialog).

After the game is over, the user can choose to view the questions answered (jump to Wrong Mode Activity), play again (wait for a second and re-run), or exit (return to the main activity).

### 2.5.10 Question Book Activity

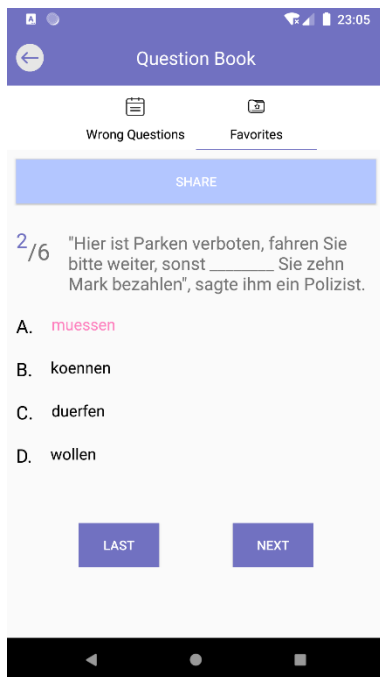


Figure XVI

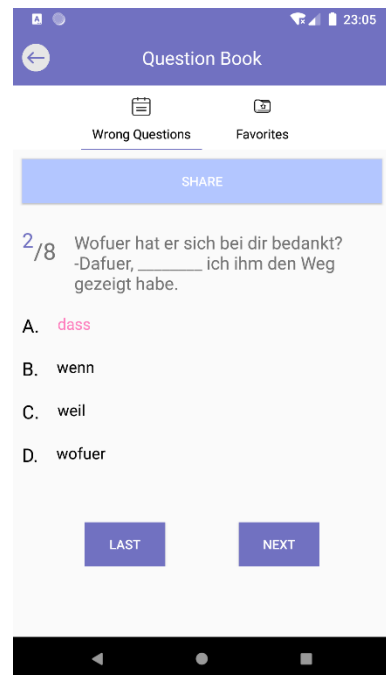


Figure XVII

In this activity, through the combination of TabLayout, TabItem, ViewPager and Fragment, the sliding switch between the wrong book and the favorites was implemented. Other functions are similar with Wrong Mode Activity.

### 2.5.11 Timer Activity

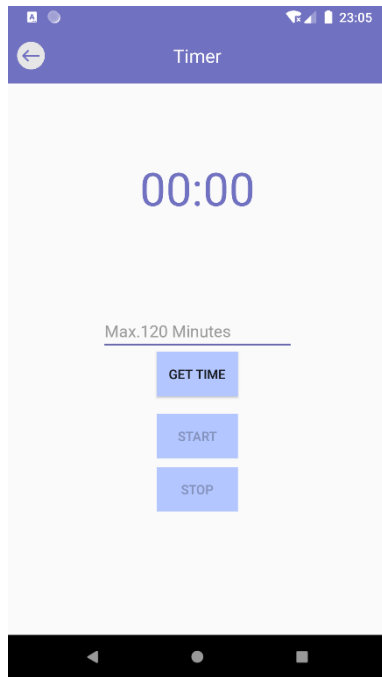


Figure XIX

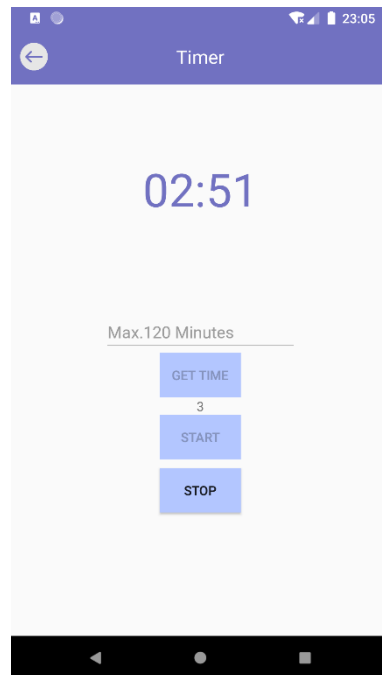


Figure XVIII

This Activity uses a countdown component to implement a simple Pomodoro.

The user can enter the number of minutes in EditText (more than 120 or less than or equal to 0, an error message "Invalid Input" will appear), click the "Get Time" button to get the time, and the "Start" button to start the countdown. The "Stop" button is only available after the countdown starts. Click the button to obtain the time without typing any number, then an error message will appear.

2.5.12 Achievement Activity

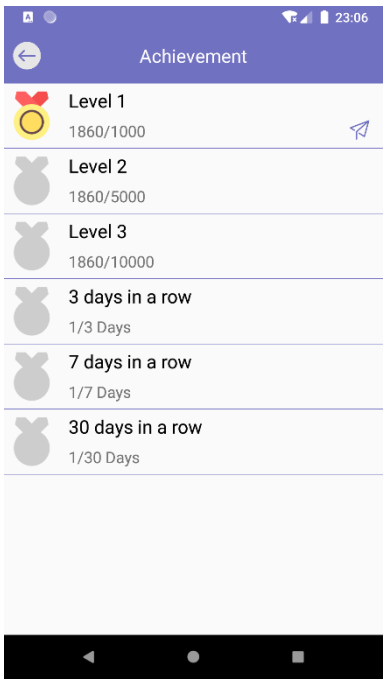


Figure XX

After meeting certain conditions, achievements will be unlocked. Achievements can be shared in text form. More achievements and bonus have not been added.

### 3 Summary

One of the biggest problems in developing this application is that before the project started, there was only a general idea of the function of the entire app in mind, without systematic planning. This caused the entire process to be very messy, and wasted a large portion of time at the beginning of the project to rewrite.

The second question is about notification. In fact, so far, this problem has not been solved. In the process of debugging on our own device, due to the API version of the device (API 23), channel ID (minSDK 26) is not available. If you do not add a Channel for notification, the notification will not be triggered.

The last problem is about our original motivation to develop this application. We want to increase our interest in learning German by implementing some special achievements. However, due to the demand for art designing, we were unable to complete this part for the time being, so in this project, only a German learning app that could meet our own needs was first developed.

The gain in this project is to try many components that have not been used in the exercise sheet, such as TabLayout, ViewPager and so on. We also try to customize UI components, use codes to draw graphics, calendars, etc.

We also want to try more components and classes in the future, such as maps and some animations.