Mobile Application for Keeping Records of Class Attendance

Petar Nenadić(✉), Damir Blažević, and Tomislav Keser

Faculty of Electrical Engineering, Computer Science and Information Technology, Osijek, Croatia
petar.nenadic@student.ferit.hr

Abstract. Recording attendance in the class meeting session is often time-consuming, prone to human error(s) and frequent recording of incorrect data. Also, tracking attendance per student can be a tedious process since it involves some sort of manual tracking/counting by pen and paper. This method consumes additional time and data may be prone to error from time to time. To avoid these problems, this paper describes a mobile application for student attendance. This application is mainly designed for students and professors who maintain attendance. Using this application, students can mark their attendance with mobile devices, furthermore, application allows the professors to keep track of student's attendance and store data for further retrieval. With advent of smartphones and tablets, which are very convenient to use, this process can be fully automated using the right technology. The tools used for application development are Kotlin and XML.

Keywords: Android · Mobile · Application · Records · Attendance · Student · Professor

1 Introduction

In current scenario, tracking attendance in the class session is the basic task of the subject handlers, since attendance marking can regulate class attendance and check the number of students in the class. Management and maintenance of student information is a key task of every institution. The task of marking attendance has traditionally been carried out manually with a diary. Later, this task is performed by the desktop applications. A desktop application is a standalone application installed on specific desktop or laptop computer and tasks can only be performed with that specific desktop system.

The main disadvantage of this system is that computer systems are not portable, so they cannot be kept anywhere to perform the task such as attendance recording. The entered attendance can be seen on an individual system if the desktop is connected to the network. Another method for reporting presence is the web application. In this method, attendance data is uploaded to the server over the Internet and users, such as students and professors, can view attendance through a web browser using any device, desktop, laptop or handheld mobile device. This system is only active when the internet connection is on because the data is not updated by the local database.

D. Blažević et al. (Eds.): OTO 2022, LNNS 592, pp. 37–44, 2023.
https://doi.org/10.1007/978-3-031-21429-5_4

These limitations of traditional systems are overcome by mobile applications. The mobile application allows users to install this application on their mobile devices. The professor can update the attendance data and update the data on the server. In order to reduce physical work and achieve greater efficiency in managing student information, this paper presents a mobile application based on Kotlin for easier and more efficient management of class attendance. The proposed application can store student information in a server database and can be retrieved by mobile devices. Through this system, professors can easily record attendance.

2 Mobile Application for Keeping Records of Class Attendance

This section explains the mobile application environment and the attendance management system mobile application. Figure 1 shows the architecture of the mobile application environment. An application for the attendance management system was developed and implemented in a cloud server such as Firebase. This application is also installed on mobile devices of users such as students and professors to access class attendance data. Users are divided into two groups: students and professors. Students can see the attendance that has been uploaded to Firebase. Professors can update student's attendance via mobile device.

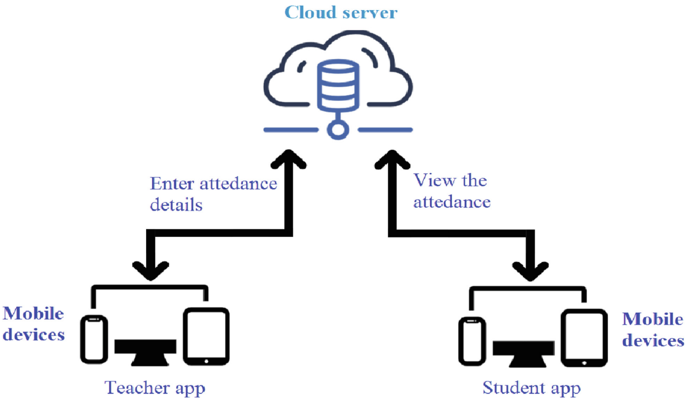


Fig. 1. Mobile application architecture for an attendance management environment

Figure 2 shows a schematic diagram of a mobile application for an attendance management system. This mobile application consists of two login modules namely staff login and student login. The student login enables students to view the data on class attendance that students receive. The professor login allows professors to update and review student attendance. Attendance data is stored in the database.

Figure 3 shows use case diagram for students and professors. Student can update his/her information and password, view attendance for each course and scan QR code from professor's mobile device. The professors can generate QR code for the students to take the attendance, check attendance data and see underrated students.

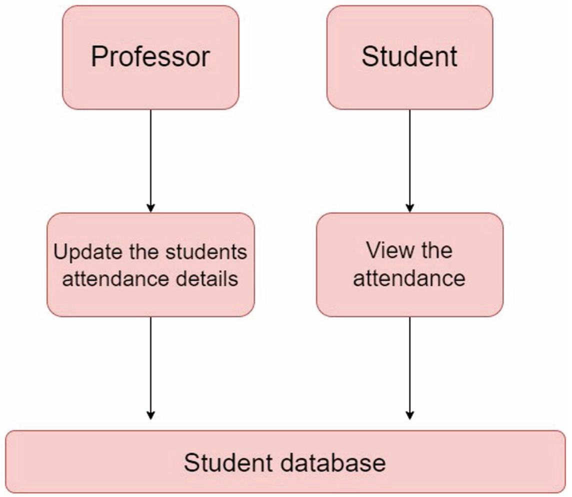


Fig. 2. Schematic diagram for attendance management system

3 Used Technologies

This mobile application is developed using Android Studio and Kotlin programming language. The application is implemented with the specification of the computer system with Windows 10 operating system, 8 GB of RAM memory and 250 GB SSD disk with Intel Pentium B960 CPU @ 2.20 GHz. The following procedure carried out for Kotlin based mobile application development.

3.1 Kotlin

Kotlin [6] is currently on of the most modern programming languages even though it started its development in 2011. It is a language that is constantly evolving and is constantly being improved with new features and functionalities. Since it enjoys a lot of support from Google, further support for Kotlin is expected, which is why more and more developers are using it instead of Java [7].

Kotlin can also be used to develop Web and server applications [8]. Although it is currently impossible to use it to develop applications for the iOS operating system, there is also a version of Kotlin Native that should allow developers to create applications for Android and iOS at the same time due to the support for the development of cross-platform applications (Fig. 4).

3.2 XML

XML [9] is a language used to mark-up data and documents. The idea behind this language is to create a language that can be read and understood even by the non-programmers. It is designed to store data in text form and frame it with tags so that they know what kind of data it is.

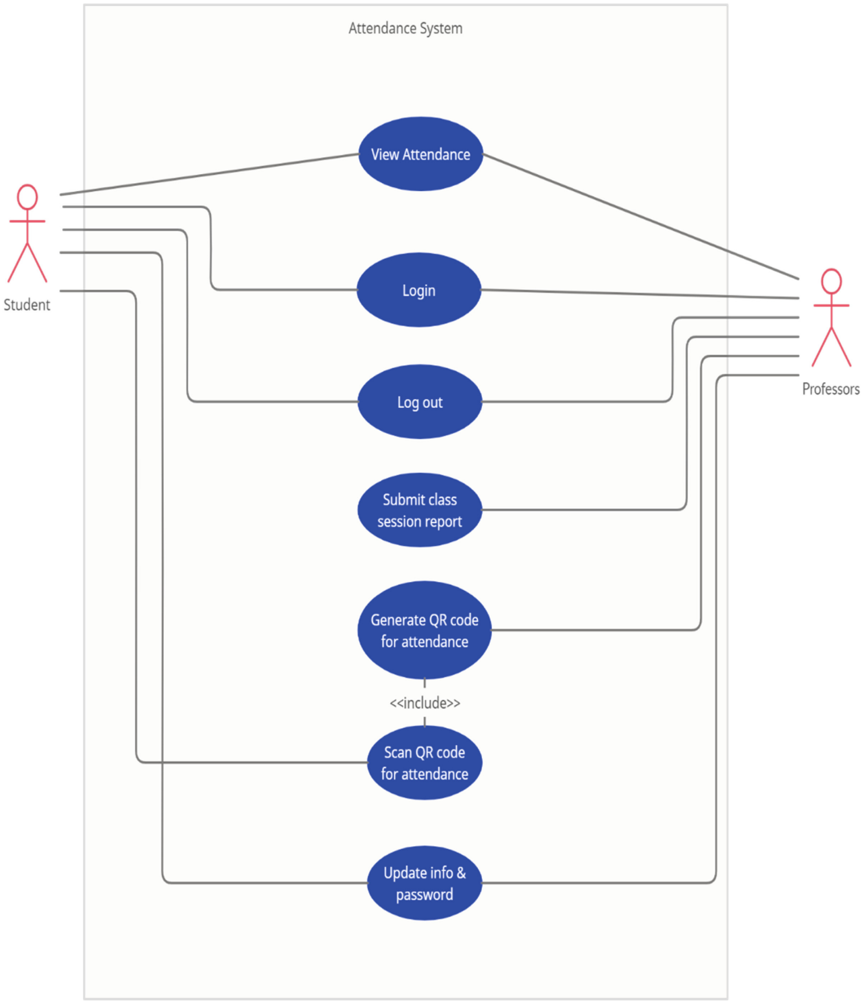
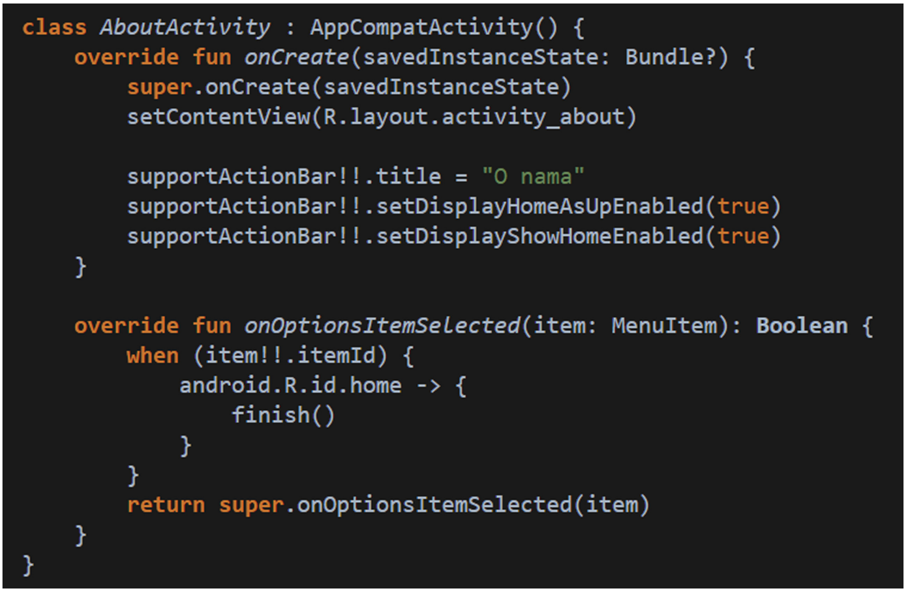


Fig. 3. Use case diagram for attendance

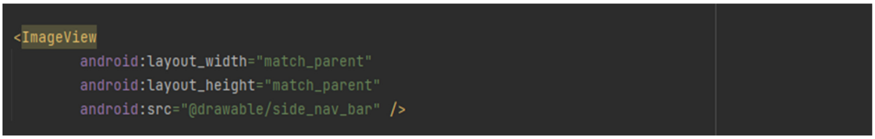
XML emphasizes usability over the Internet, simplicity and generality, it has strong support through Unicode for different languages. Although XML is intended primarily for use in documents, it can also be used for other purposes, such as data exchange, separation of data from presentation, data storage and increased data availability. XML is standardized language, which was taken care of by the World Wide Web Consortium [10].

For this project, XML is used to create layouts for application, input forms, buttons, images, creating backgrounds, etc. Example code is shown on Fig. 5.



```
class AboutActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_about)

        supportActionBar!!.title = "O nama"
        supportActionBar!!.setDisplayHomeAsUpEnabled(true)
        supportActionBar!!.setDisplayShowHomeEnabled(true)
    }

    override fun onOptionsItemSelected(item: MenuItem): Boolean {
        when (item!!.itemId) {
            android.R.id.home -> {
                finish()
            }
        }
        return super.onOptionsItemSelected(item)
    }
}
```

Fig. 4. Example of code written in Kotlin programming language



```
<ImageView
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:src="@drawable/side_nav_bar" />
```

Fig. 5. Example for inserting image into layout of application

3.3 Firebase

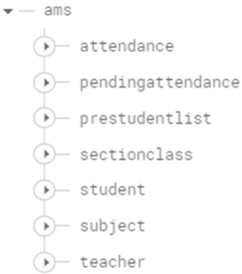
Firebase [5] is platform of the company of the same name, which is used for the development of mobile and web applications, which can be used to quickly create application and increase the base of engaged users. The platform includes several well-integrated features that can be combined, and has analytics tools, mobile backends and tools to develop and maximize application success.

Firebase is used for this application because it enables real-time data storage and that data is available anywhere in the world. Figure 6 shows example table from Firebase.

4 Results and Discussion

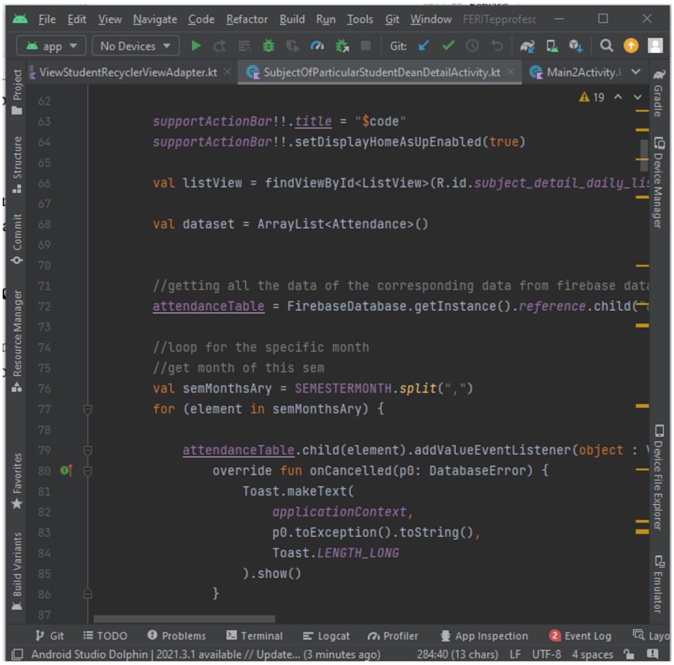
This section presents sample code and results of the developed application. Figure 7 shows an example of the mobile application code for the student attendance monitoring system. Figure 8 shows the professor's login screen (a) and main screen (b).

Figure 8 shows screenshots from professor's application. Professors use their ID and password to login and gain access to information about students' attendance.



```
ams
    attendance
    pendingattendance
    prestudentlist
    sectionclass
    student
    subject
    teacher
```

Fig. 6. Table from Firebase real-time database

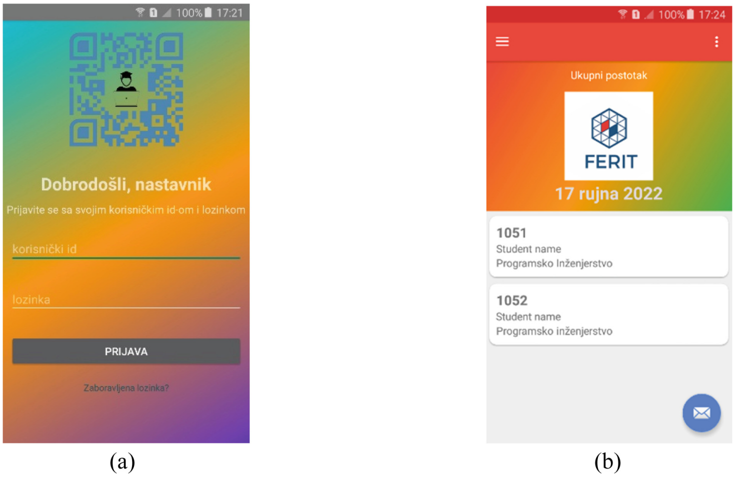


```

        supportActionBar!!.title = "$code"
        supportActionBar!!.setDisplayHomeAsUpEnabled(true)

        val listView = findViewById<ListView>(R.id.subject_detail_daily_lis

        val dataset = ArrayList<Attendance>()


        //getting all the data of the corresponding data from firebase data
        attendanceTable = FirebaseDatabase.getInstance().reference.child("a

        //loop for the specific month
        //get month of this sem
        val semMonthsAry = SEMESTERMONTH.split(",")
        for (element in semMonthsAry) {

            attendanceTable.child(element).addValueEventListener(object : V
                override fun onCancelled(p0: DatabaseError) {
                    Toast.makeText(
                        applicationContext,
                        p0.toException().toString(),
                        Toast.LENGTH_LONG
                    ).show()
                }

```

Fig. 7. Mobile application code example for attendance management system [16]

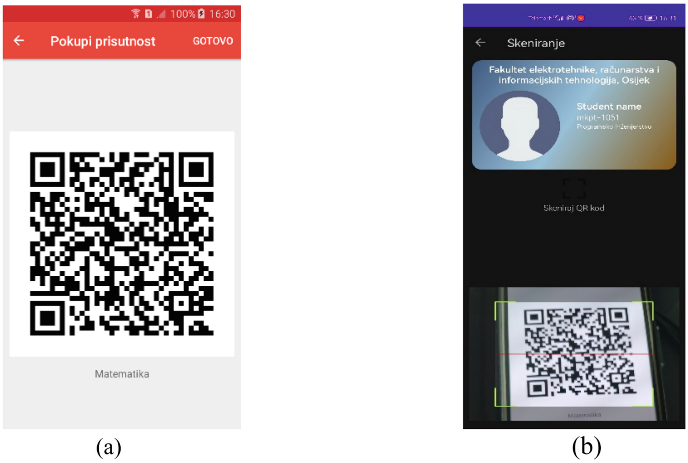
After professor is logged in, he/she can generate QR code which allows students to scan it with their mobile devices and confirm their attendance. Figure 9(a) shows application for professors with generated QR code and application for students for scanning QR code (b).

After student scans QR code from professor's applications, his attendance is recorded in database (Fig. 10). Those records are available to both professors and students.



(a) (b)

Fig. 8. Professor login screen (a) and main screen (b)

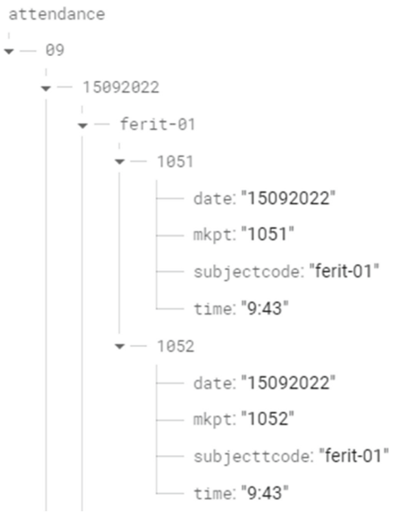


(a) (b)

Fig. 9. (a) QR code from professor's application for attendance submission, (b) scanning QR code with camera from student's application

5 Conclusion

This paper presents a mobile application for the attendance management system. This system is enabled with two applications, one for students and one for professors. Registering attendance can be very time-consuming process, which request manual steps and is error prone. This application solves all the downfalls of registering attendance manually. It allows student to scan QR code from professor's mobile phone and register presence, which not only saves time but is also error free. All data is stored in Firebase database and is available anytime and everywhere. Also, professor can look at attendance of students whenever he/she wants and see which student has minimum attendance.



```
attendance
  09
    15092022
      ferit-01
        1051
          date: "15092022"
          mkpt: "1051"
          subjectcode: "ferit-01"
          time: "9:43"
        1052
          date: "15092022"
          mkpt: "1052"
          subjecttcode: "ferit-01"
          time: "9:43"
```

Fig. 10. Example of data stored in real-time database [5]

References

1. Android Developers: developer.android.com
2. Android Developers Blog: http://android-developers.blogspot.com/
3. Darwin, I.F.: Android Cookbook. O'Reilly Media, Inc. (2012)
4. Stack Overflow: http://stackoverflow.com
5. Firebase: firebase.google.com
6. Kotlin: kotlinlang.org
7. Lackner, P.: Kotlin Newbie to pro.https://www.youtube.com/playlist?list=PLQkwcJG4YTCRSQikwhtoApYs9ij_Hc5Z9
8. Griffiths, D., Griffith, D.: Head First Kotlin, 1st edn. O'Reilly Media (2019)
9. Horton, J.: Android Programming with Kotlin for Beginners. Packt Publishing (2019)
10. Isakova, S., Jemerov, D.: Kotlin in Action, 1st edn. Manning (2017)
11. Kotlin Programming: The Big Nerd Ranch Guide, Big Nerd Ranch Guides, 1st edn (2018)
12. Cafer, F.: Attendance Taker (2015). https://play.google.com/store/apps/details?id=com.ferid.app.classroom
13. Nirvan, R.: Attendance Register (2014). https://play.google.com/store/apps/details?id=com.rudra.attendanceRegister
14. Jibble: Free Time Clock App. Jibble Ltd (2020). https://play.google.com/store/apps/details?id=io.jibble.androidclient
15. Brar, A.: Virtual Attendance (2014). https://play.google.com/store/apps/details?id=deftsoft.virtual.attendance
16. Wifi Attendance, Redbytes Software Pvt Ltd (2016). https://play.google.com/store/apps/details?id=com.edsys.wifiattendance