

# Implementation of REST API in Android-based Student Services Application

Muhammad Hafidz  
University of Technology Yogyakarta  
Yogyakarta, Indonesia

Agus Suhendar  
University of Technology Yogyakarta  
Yogyakarta, Indonesia

## ABSTRACT

This research is focused on developing an Android-based student service system. This application was created to automate student complaints to school service providers to improve student and school relationships. However, to create a good system requires some testing to be able to produce a satisfactory application. REST API testing is done using Postman, which, from the test, found no problems. Thus, the backend system created has passed the test, which will be continued by testing using Whitebox. It is hoped that this system can be implemented in schools to improve interactions between students and school service providers and provide better improvements according to student needs.

## General Terms

Android, REST API, Kotlin, Postman

## Keywords

MobileApplication, REST API, StudentServices

## 1. INTRODUCTION

The Industrial Revolution 4.0 is growing rapidly in various fields of life, including education, which creates educational innovation 4.0, focusing on developing skills that are personal, smart, portable, global, and virtual [1]. Education is important in human life, meaning every individual has the right to receive it and is expected always to develop it. Education is a process that helps individuals develop themselves to live and contribute to society [2]. Therefore, being an educated person is very important. In schools, the role of student affairs services is an important aspect of supporting services to help students. Student affairs can help improve learning outcomes, cognitive skills, and personal and organizational skills [3]. In the school environment, of course, many violations occur. The efforts made are by having a student teacher who provides learning about good social behavior and can help students resolve their complaints. However, there is a problem where most students are afraid to report incidents that occur at school after the students go home.

According to a study funded by UNICEF and conducted by the Ministry of Communication and Information of the Republic of Indonesia (Kemkominfo) in 2014, 98 percent of children and adolescents know about the Internet, and 79.5 percent are Internet users [4]. This supports creating a system that can help make complaints quickly and precisely using their cell phones. Mobile applications are referred to as a type of software that runs on mobile devices, namely smartphones or tablets [5], [6]. Mobile applications have become very useful for individuals and various fields, such as education.

The research designs and implements the REST API to create a user-centered student service application where users are student teachers and students. REST API is used to connect the Android application to the cloud server. Android was chosen as

the platform for this system due to its widespread use, with at least 3.79 billion people, or 51 percent of the global population, reportedly using mobile devices [7]. REST API is an architectural tool that builds in web services and focuses on system resources such as data transmission and requests using http [8]. The community has widely accepted REST because it is easy to use in the mobile cloud. REST web services identify URIs and then modify them with GET, POST, PUT, or DELETE commands that can be represented in several programming languages [9]. After the design is complete, the next step is to test the REST API to ensure that the API provides the expected response using the Postman application.

This research aims to build an application that supports student teachers in recording, managing, and reporting student complaints to the school.

## 2. RESEARCH METHOD

This research method includes several steps, including planning needs and understanding the process involving users, such as student teachers and students. Next, an application system consisting of an Android application, database, and REST API was developed. The Android application and REST API were created with the Kotlin programming language, using the Visual Studio Code text editor. In contrast, MySQL and XAMPP were used as part of the database system. After the process is completed, test the REST API using Postman. Postman features are considered quite simple, which makes API testing can be done well and quickly [10]. Postman categorizes requests based on the request method, URL, and request parameters. In addition, monitoring API performance and response time provides an integrated working environment for real-time API creation and usage with shared context [11].

### 2.1 System Development

The system has been divided into three components: an Android application as a frontend, a database system for data storage and automation, and a REST API that acts as a backend. To build this application, a unified modeling language (UML) is needed to describe the interaction relationship between the system and the actor as the first step in creating a system design.

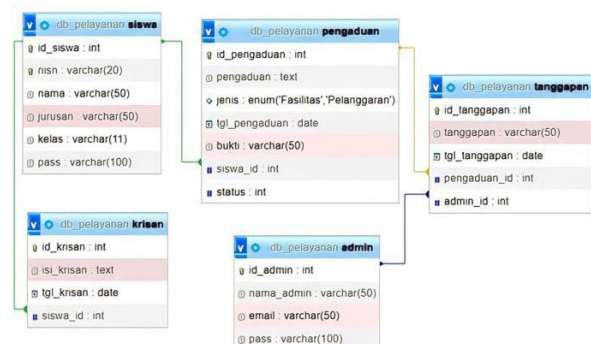


Fig. 1 Student Services Entity Relationship Diagram

The database system was built using MySQL, based on an Entity Relationship Diagram (ERD), which was then developed into a series of Relationship Tables. This relationship table includes several tables, such as the admin table, which is used to store data about student teachers; the student table, which contains information about students; the complaint table, which is used to record complaint reports; the response table, which is used to store information about responses to complaints, and the chrysanthemum table which is used to record information about criticism and suggestions.

The REST API acts as the system's backend and was developed using the Visual Studio Code text editor.

## 2.2 REST API Testing

The use of Postman in testing has a wide range of applications in software development, especially API and Cloud testing. This includes testing various functions within the API, such as request and response testing. These tests relate to the RESTful aspects and various response validations of the application. In addition, Postman can also be used to perform API performance testing by measuring response time, speed, and resource usage [12]. With Postman, users can write and run tests for any query using the Hypertext Preprocessor Language (PHP) [13].

## 3. RESULT AND DISCUSSION

### 3.1 System Results

The MySQL database consists of four tables representing the stored data: complaint data from students, complaint response data, criticism, suggestion (chrysanthemum) data, and login data. The trigger function automates changes to the complaint and response data and then adds them to the criticism and suggestion data, login data, and complaint data. The RESTful API design in the student application uses the HTTP method, which consists of GET and POST. In designing the GET and POST methods, use the primary key of each table as a parameter. This design is done on a local server using PhpMyAdmin. In the response and complaint classes, the program code contains a query to connect these classes with the database used. Query made tailored to the needs of each method. The list of endpoints and methods used in the application data module is shown in Table 1.

Table. 1 List of student service endpoints

Method	Endpoint
getPengaduanByIdSiswa	/api/pengaduan
CreateDataPengaduan	/api/Pengaduan
getTanggapanByIdPengaduan	/api/Tanggapan
CreateDataKrisan	/api/Krisan
CreateDataLogin	/api/Login

This application was developed using Kotlin programming language and Visual Studio Code. Users can add complaints, provide criticism and suggestions, and view detailed complaint information and profile pages. This application performs inputting complaint data connected to the database via REST API and the Internet, allowing students to use this application for their complaints and grievances.

### 3.1.1 Login View

The Login page only displays the login form to enter the application, users can enter the NISN and password that has been registered by the admin. After that, by clicking the login button, the user will enter the application.

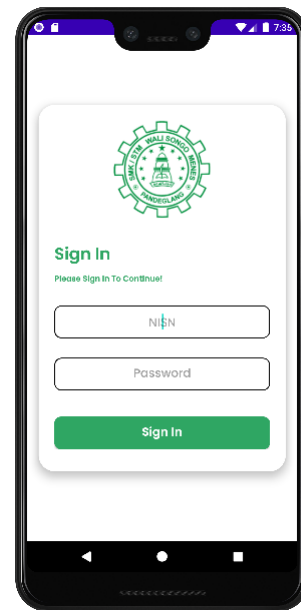


Fig. 2 Login View

### 3.1.2 Dashboard View

On the dashboard page, there is a history of complaints that have been inputted by students. The history only displays information about the type of complaint and the date of the complaint.

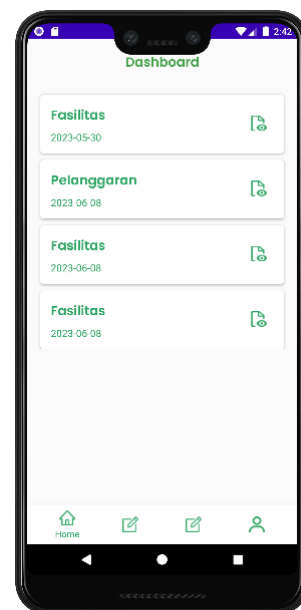


Fig. 3 Dashboard View

### 3.1.3 Complaint View

On the complaint page, there is a form that allows students to make complaints. The form contains inputs

for the type of complaint, the content of the complaint, and allows student to take photos as proof of the complaint directly through the student's smartphone camera.

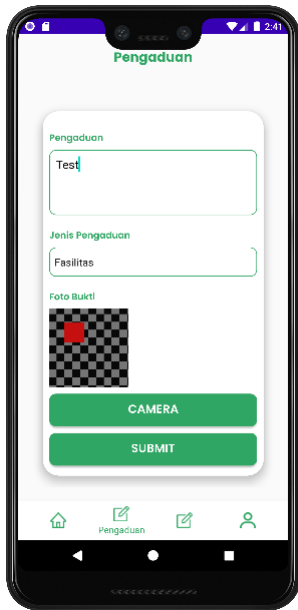


Fig. 4 Complaint View

### 3.1.4 Criticism and suggestion View

On this page, there is a form that allows students to provide criticism and suggestions. The form contains input for the content of criticism and suggestions that student want to submit.

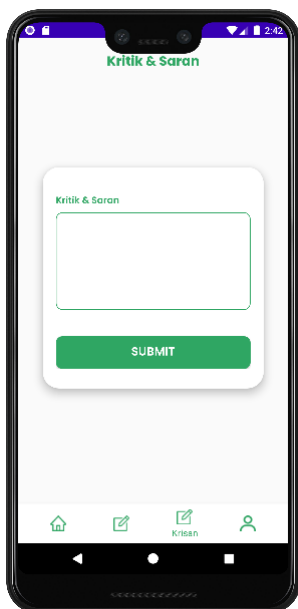


Fig. 5 Criticism and suggestion View

### 3.1.5 Complaint Detail View

On the complaint details page, there is a table that displays complaints that have been made by students and have been responded to by the admin. The table contains information such as the type of complaint, complaint content, complaint date, response content, and response date. The complaint details

page can only be accessed if the complaint made by the student has been responded to by the admin.

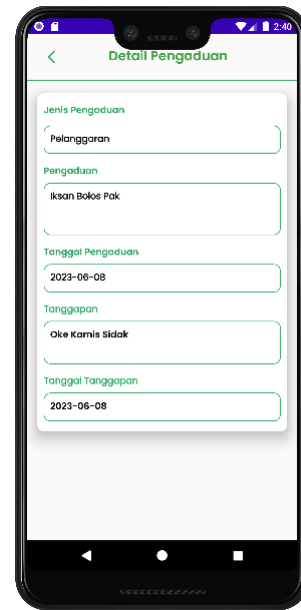


Fig. 6 Complaint Detail View

### 3.1.6 Profile View

On the profile page, student data such as NISN, name, major class, student profile photo, and logout button are displayed.

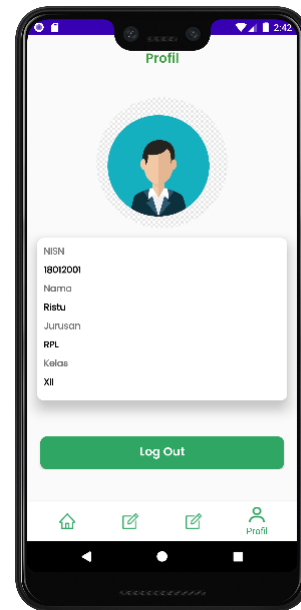


Fig. 7 Profile View

## 3.2 Rest API Testing Result

Testing is carried out in the form of functional testing of the results produced using the white box testing method to ensure that the features that have been designed can run as expected. Testing of the RESTful API is done on a local device using Postman.

1. `getPengaduanByIdSiswa`

The results of the `getPengaduanByIdSiswa` trial on Postman display a response code 200 with OK status, which means that

the request to display all student data was successful. Figure 3. It is a display of the results of the getPengaduanByIdSiswa trial.



Fig. 8 Trial Result getPengaduanByIdSiswa

## 2. CreateDataPengaduan

The results of the CreateDataPengaduan trial on Postman display response code 201 with Created status, which means that the request to add complaint data was successful. Figure 4 is a display of the results of the CreateDataPengaduan trial.

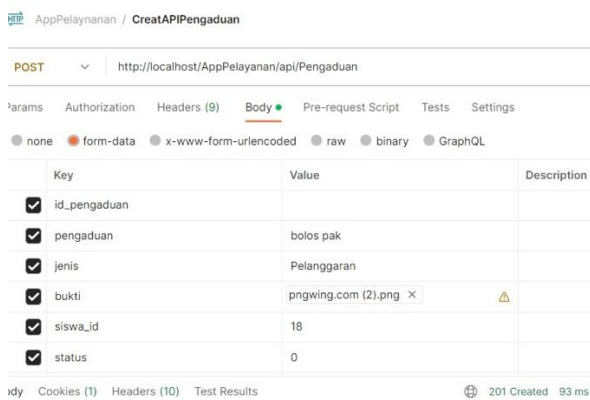


Fig. 9 Trial Result CreateDataPengaduan

## 3. getTanggapanByIdPengaduan

The results of the getTanggapanByIdPengaduan trial on Postman display a response code 200 with OK status, which means that the request to display all responses from the complaint was successful. Figure 5 is a display of the results of the getTanggapanByIdPengaduan trial.



Fig. 10 Trial Result getTanggapanByIdPengaduan

## 4. CreateDataKrisan

The results of the CreateDataKrisan trial on Postman display response code 201 with Created status, which means that the request to add criticism and suggestion data was successful. Figure 6 is a display of the results of the CreateDataKrisan trial.

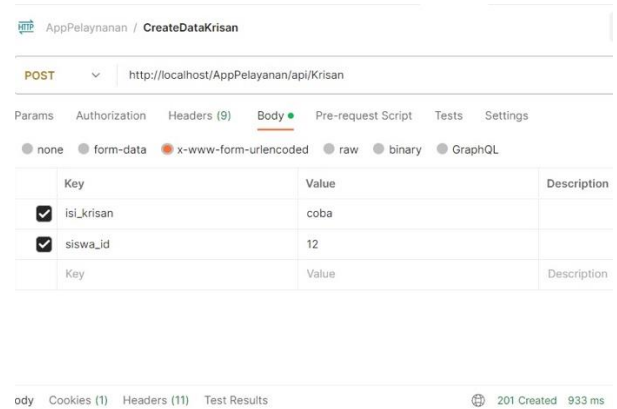


Fig. 11 Trial Result CreateDataKrisan

## 5. CreateDataLogin

The results of the CreateDataLogin trial on Postman display a response code 200 with OK status, which means that the request to enter the application was successful. Figure 7 is a display of the CreateDataLogin trial results.

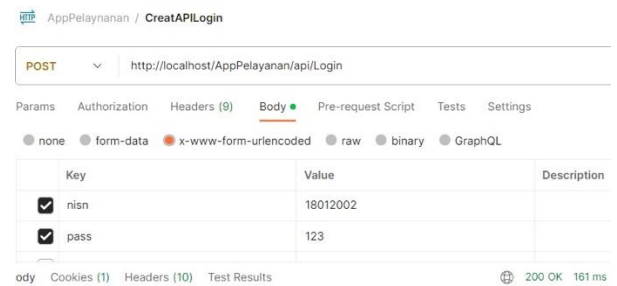


Fig. 12 Trial Result CreateDataLogin

## 3.3 Discussion

Developing software applications is a complex process that requires careful design and attention to detail. As explained earlier, choosing the right tools and technologies ensures the software runs efficiently and reliably. For example, using Kotlin, REST APIs, and MySQL to build Android applications and databases is crucial in ensuring software quality.

The generation is one of the fundamental aspects of software testing. Whitebox and Postman are tools that can automate the process of testing applications and REST APIs. From there, it becomes clear how important it is to conduct thorough testing to ensure the software is properly tested.

In conclusion, software development and testing require careful planning and attention to detail. Using the right tools and technologies goes a long way to ensuring that the software operates efficiently and reliably. This article highlights the importance of applying thorough testing to software.

## 4. CONCLUSION

This research aims to develop an Industry 4.0-based student service application that supports student affairs teachers in recording, managing, and reporting student complaints to the school. This application uses REST API and Android technology to create a platform that is easily accessible by student teachers and students. Using Postman as a testing tool ensures that the application and REST API function properly.

The system has three main components: an Android application at the front end, a MySQL database system for data storage,

and REST API at the back end. Through testing with Postman, various features and endpoints in the REST API were tested and performed as expected.

Software development requires careful planning and selecting the right technology to ensure quality and performance. Software testing, especially API testing, is crucial in ensuring software reliability and efficiency. Thus, using appropriate tools and technologies in software development and testing is essential to achieve good results.

It is hoped that this system can be implemented in schools to improve interactions between students and school service providers and provide better improvements according to student needs.

## 5. REFERENCES

- [1] A. A. Shahroom and N. Hussin, "Industrial Revolution 4.0 and Education," *International Journal of Academic Research in Business and Social Sciences*, vol. 8, no. 9, Oct. 2018, doi: 10.6007/ijarbss/v8-i9/4593.
- [2] Y. Alpian, W. S. Anggraeni, and U. S. M. N. Wiharti, "Pentingnya Pendidikan Bagi Manusia," *Jurna Buana Pengabdian*, vol. 1, Feb. 2019.
- [3] P. A. Bloland, L. C. Stamatakos, and R. R. Rogers, "Redirecting the Role of Student Affairs to Focus on Student Learning," *J Coll Stud Dev*, vol. Vol 37 No 2, pp. 217–226, 1996.
- [4] G. S and D. Broto, "98 Persen Anak dan Remaja Tahu Internet," Jakarta, Feb. 2014. Accessed: Sep. 12, 2023. [Online]. Available: [https://www.kominfo.go.id/index.php/content/detail/3836/98+Persen+Anak+dan+Remaja+Tahu+Internet/0/berita\\_satker#:~:text=Jakarta%2C%20Kominfo%20%2D%20Studi%20yang%20didanai,persen%20diantaranya%20adalah%20pengguna%20internet.](https://www.kominfo.go.id/index.php/content/detail/3836/98+Persen+Anak+dan+Remaja+Tahu+Internet/0/berita_satker#:~:text=Jakarta%2C%20Kominfo%20%2D%20Studi%20yang%20didanai,persen%20diantaranya%20adalah%20pengguna%20internet.)
- [5] V. N Inukollu, D. D. Keshamon, T. Kang, and M. Inukollu, "Factors Influencing Quality of Mobile Apps: Role of Mobile App Development Life Cycle," *International Journal of Software Engineering & Applications*, vol. 5, no. 5, pp. 15–34, Sep. 2014, doi: 10.5121/ijsea.2014.5502.
- [6] A. Phongtraychack and D. Dolgaya, "Evolution of Mobile Applications," *MATEC Web of Conferences*, vol. 155, Feb. 2018, doi: 10.1051/mateconf/201815501027.
- [7] S. Mujahidin, M. Reinaldy Hermawan, and C. Cahyo Utomo, "Implementation of Automated Test Case Generation in REST API on Android-Based Koperasi Application," *Journal of Information Systems and Informatics*, vol. 5, no. 1, pp. 123–133, Feb. 2023, doi: 10.51519/journalisi.v5i1.431.
- [8] S. P. Ma, C. Y. Huang, Y. Y. Fanjiang, and J. Y. Kuo, "Configurable RESTful service mashup: A Process-Data-Widget approach," *Applied Mathematics and Information Sciences*, vol. 9, no. 2, pp. 637–644, 2015, doi: 10.12785/amis/092L39.
- [9] K. E. Mohamed and D. Wijesekera, "Performance analysis of web services on mobile devices," *Procedia Comput Sci*, vol. 10, pp. 744–751, 2012, doi: 10.1016/j.procs.2012.06.095.
- [10] A. S. Sari and R. Hidayat, "Designing website vaccine booking system using golang programming language and framework react JS," *Journal of Information System, Informatics and Computing Issue Period*, vol. 6, no. 1, pp. 22–39, 2022, doi: 10.52362/jisicom.v6i1.760.
- [11] W. Galindra Wardhana, I. Arwani, and B. Rahayudi, "Implementasi Teknologi Restful Web Service Dalam Pengembangan Sistem Informasi Perekaman Prestasi Mahasiswa Berbasis Website (Studi Kasus: Fakultas Teknologi Pertanian Universitas Brawijaya)," *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. 4, no. 2, pp. 680–689, 2020, [Online]. Available: <http://j-ptiik.ub.ac.id>
- [12] A. Arcuri, "Automated Black- And White-Box Testing of RESTful APIs with EvoMaster," *IEEE Softw*, vol. 38, no. 3, pp. 72–78, May 2021, doi: 10.1109/MS.2020.3013820.
- [13] Dheeraj and K. Sharma, "Security Testing of API using Postman and Swagger tools and its use in Internet of Things (IoT)," *J Emerg Technol Innov Res*, vol. 6, no. 2, 2019, [Online]. Available: [www.jetir.org](http://www.jetir.org)