

Rest API Implementation on Presence System using QR-Code and Web-Based Haversine Formula Method

Yiyi Supendi^{1*}, Rasidin², Amelia Hani²

Teknik Informatika, Universitas Langlangbuana, Bandung, Indonesia¹
Teknik Informatika, Universitas Halim Sanusi, Bandung, Indonesia²

*Email : yiyi.supendi@gmail.com

Abstract

Student presence is one of the most important things in lecturing. However, the student attendance system at Langlangbuana University currently still uses a manual system in which students sign the attendance sheet provided. The attendance list system has several weaknesses, namely the vulnerability to the safekeeping of student attendance lists, loss or damage to the attendance sheet, and the use of a relatively long time so that the manual attendance process as a whole becomes inefficient. In order to minimize these weaknesses, we need a system that can process student attendance lists. There are technologies that can be used to reduce these cases, one of which is QR-Code, which uses the REST API for the student attendance list system so that later students can carry out the attendance list process by scanning the QR code provided in the application. With this application, it will be easier for lecturers to manage attendance lists and for students to make attendance.

Keywords

presence, QR-Code, REST API, Harversine Algorithm

Introduction

Technological developments and the rapid dissemination of information enter various aspects of life, be they social, political, cultural, educational, or employment. This is evidenced by the existence of a computerized system for every activity in various agencies. The attendance list of students is one of the most important things in lectures. However, the student attendance system at Langlangbuana University currently still uses a manual system in which students sign the attendance sheet provided. The attendance list system has several weaknesses, namely the vulnerability to the safekeeping of student attendance lists, loss or damage to the attendance sheet, and the use of a relatively long time so that the manual attendance process as a whole becomes inefficient.

In order to minimize these weaknesses, we need a system that can process student attendance lists. One of the technologies that can be used to reduce this case is the QR code. QR-Code has a high capacity for data storage. For this reason, it is necessary to create a web application

Submission: 29 August 2023; **Acceptance:** 13 November 2023



Copyright: © 2023. All the authors listed in this paper. The distribution, reproduction, and any other usage of the content of this paper is permitted, with credit given to all the author(s) and copyright owner(s) in accordance to common academic practice. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license, as stated in the website: <https://creativecommons.org/licenses/by/4.0/>

and Rest API for the student attendance system, where students can process the attendance list by scanning the QR code generated from the web application. With this web application, it will be easier for lecturers to process student attendance lists and student absence permits.

Based on the background description above, the formulation of the problem is how to make it easier for students to take attendance, how to make it easier for lecturers to check attendance, increase security in the student attendance process using the haversine method, and implement distance in the presence system.

REST is an application programming interface (API) standard (Ehsan, A., et.al (2022), Liu Y., et.al (2022), Indrawan, G., et.al (2022)). REST is described as "An architectural style of designing a web service where the REST design has resources that can be accessed via a specific HTTP URL address," by Perkasa & Setiawan (2018). REST enables clients to send requests quickly over the HTTP protocol by using URLs. Perkasa & Setiawan (2018) offered a different viewpoint. Web services that use REST architecture are referred to as RESTful. Due to its employment as a link between various systems (clients and servers), An API, according to Perkasa & Setiawan (2018), is an interface that serves as a bridge between various application systems. Web services are just one way that APIs can be used; they can also do the following.

1. Client-Server: The most fundamental component of creating a REST API is client-server. Whether a client request is successful or unsuccessful, the server must be able to react to it. The HTTP protocol is used to communicate between clients and servers.
2. Stateless: No client-related state may be stored by REST APIs. The client must continue to store all state. In other words, the REST API does not support sessions. The client's requests must include precise information. You shouldn't anticipate the RESTful API to save data from earlier calls for use in new ones.
3. Cacheable: It is preferable if the REST API uses the cache principle to swiftly respond to requests. in order to prevent the database from being used for every request.
4. Layered: The client should not be required to understand the REST API server's sophisticated architecture in order to use it.

The QR-Code method, according to Rahmawati, A., and Rahman, A. (2011), transforms written data into 2-dimensional codes that are printed in a more condensed format. The Japanese company Denso-Wave first invented the QR-Code in 1994. It can store any kind of data, including numeric, alphanumeric, binary, and kanji/kana data. Even though it is small, it has a lot of storage capacity. Due to its ability to handle data both horizontally and vertically, the QR-Code has a smaller display than a barcode, which implies that its image can only be one-tenth the size of a barcode. According to the version, QR-Code may repair faults by up to 30%, making it resistant to damage. The purpose of the three-square signs at the corners of the symbol is to allow for consistent reading of the symbol from all directions.

Haversine Algorithm, according to N. Chopde and M. Nichat (2013), is a method used to calculate the distance between points on the earth's surface using latitude (longitude) and longitude (latitude) as input variables. Haversine is an important equation in navigation that can give the great circle distance between two points on the surface of the earth or a spherical object based on longitude and latitude. Utilization of the Haversine formula is used in deciding the shortest distance

between two points on Earth (Rahardian, Vinaya, & Diatmika, 2021). Based on the explanation above, the following is the formula for the Haversine Algorithm:

$$\begin{aligned}\Delta \text{ latitude} &= \text{latitude2} - \text{latitude1} \\ \Delta \text{ longitude} &= \text{longitude2} - \text{longitude1} \\ a &= \sin^2\left(\frac{\Delta \text{latitude}}{2}\right) + \cos(\text{latitude1}) \cdot \cos(\text{latitude2}) \cdot \sin^2\left(\frac{\Delta \text{longitude}}{2}\right) \\ b &= 2 \tan^{-1}\left(\sqrt{a} \cdot \sqrt{1-a}\right) \\ a &= R \cdot C\end{aligned}$$

With conditions where:

R = Earth radius (6371 Km)

Δlat = The difference between latitude A dan latitude B Δlong = The difference between longitude A dan longitude B C = Axes intersection calculation'

d = distance/range (Km)

$1^\circ = 0.0174532925$ Radian

The Haversine is an algorithm used to calculate the distance to where students are in a certain area based on coordinates for lecture attendance. So latitude and longitude can provide information on the distance from these two points (Prihantoro & Wahyuddin, 2022). Here's the haversine formula:

$$\Delta \text{lat} = \text{lat1} - \text{lat2} \quad (1)$$

$$\Delta \text{lon} = \text{long1} - \text{long2} \quad (2)$$

$$a = \sin^2(\Delta \text{lat}/2) + \cos(\text{lat1}) \cdot \cos(\text{lat2}) \cdot \sin^2(\Delta \text{lon}/2) \quad (3)$$

$$c = 2a \cdot \sin^{-1}\left(\sqrt{a} \cdot \sqrt{1-a}\right) \quad (4)$$

$$d = R \cdot c \quad (5)$$

Where: Δlat is *latitude* Δlong is, *longitude* R is the radius of the earth. 6371e3(m)

c = axes point

d = is distance (m)

$1 = 0.0174532925$ radian

Calculation and Simulation of the Haversine Method

```
Function haversine_distance($lat1, $lon1, $lat2, $lon2) {
  $R = 6371000; // Earth's radius in meters
  $lat1 = deg2rad($lat1);
  $lon1 = deg2rad($lon1);
  $lat2 = deg2rad($lat2);
  $lon2 = deg2rad($lon2);
  $delta_lat = $lat2 - $lat1;
  $delta_lon = $lon2 - $lon1;
  $a = sin($delta_lat / 2) ** 2 + cos($lat1) * cos($lat2) * sin($delta_lon / 2) ** 2;
  $c = 2 * atan2(sqrt($a), sqrt(1 - $a));
  $distance = $R * $c;
  return $distance;
}
```

```

}

// First location coordinates
$lat1 = -6.21462;
$lon1 = 106.84513;
// Second location coordinates
$lat2 = -6.22496;
$lon2 = 106.84572;
// Calculation of the distance between the two points in meters
$distance = haversine_distance($lat1, $lon1, $lat2, $lon2);
echo " The distance between the two points is {$distance}
meter.";

```

Methodology

Research phases involve engineering research methods and systems development methods. In the implementation of this study, the following stages were carried out. The research is explained as follows:

No	Stages	Information
1.	<i>Plainning</i>	This is the initial planning stage for carrying out the process. Study
2.	<i>Analysis</i>	The stages in which researchers conduct research and conduct collection of data needed for the software.
3.	<i>Design</i>	The stages in which researchers carry out design and modeling include data structure modeling, software architecture, interface display, and program algorithms. Aim to understand description of the system to be created
4.	<i>Construct</i>	The stage where we carry out the process of translating the design form that has been designed into a form of coding
5.	<i>Testing</i>	The stage where testing the system and the code that has been made
6.	Laporan	Stages of making a report of the entire research process have already been made.

The system development method used in the system development process in this final project uses the waterfall method. The choice of the waterfall method in this study is because the initial process of gathering user needs is very clear, so it can minimize errors that occur in the application development process. The stages of the waterfall method used are as follows:

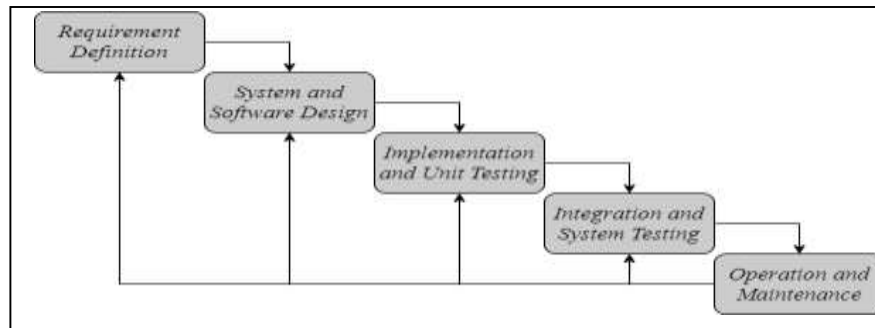


Figure 1: Steps of the Waterfall Method (Source: Somerville, 2011)

The steps taken in the waterfall method include:

1. **Requirements Analysis and Definition** The system's services, constraints, and objectives are determined by consultation with system users. It is then specified in detail and serves as a system specification.
2. **System and Software Design** The system design process allocates system hardware or software requirements by forming the overall system architecture. Software design involves identifying and describing software system abstractions.
3. **Implementation and Unit Testing** At this stage, the software design is a set of programs or program units. Unit testing involves verifying that each unit meets its specifications.
4. **Integration and System Testing** Individual program units or programs are integrated and tested as a complete system to ensure that software requirements have been met. After testing, the software system is delivered to the customer.
5. **Operation and Maintenance** usually (although not always) It is the longest phase; the system is installed and put into practical use.

Results and Discussion

The analysis in this study is an elaboration of the system and its various components. The goal is to be able to identify and evaluate various problems or obstacles that arise in the system so that countermeasures, improvements, and development can be carried out later.

The stages are carried out to find out the order of implementation in an activity that aims to benefit by using various resources. To minimize the weaknesses that have been described based on the previous business process, a system is needed that can facilitate managing student attendance and student absenteeism permits. The technology used to reduce these cases, one of which is the QR-Code, adds the haversine method by utilizing GPS, which is used to calculate the tolerance distance in carrying out student attendance. The proposed business process is shown in Figure 2.

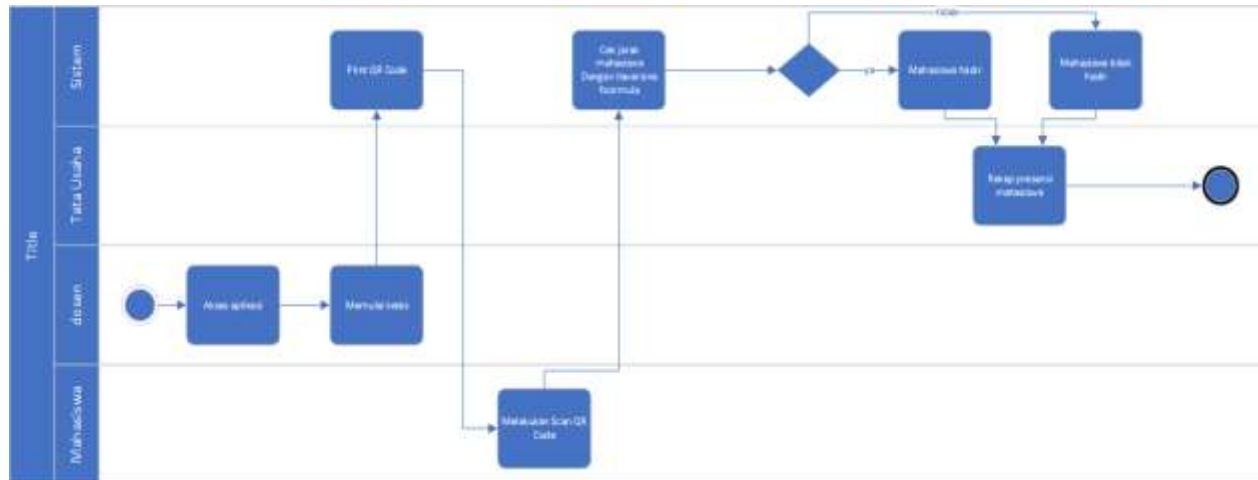


Figure 2. Business Process

Software Requirements Analysis

Software requirements analysis is a process carried out to define the needs of users and functional requirements related to the system to be built.

Identification of User Needs

Identification of the needs of software users is carried out to collect data and information about the required needs. The identification of software users is described in Table 1.

Table 1. Identification of user needs

Human Resources	
User	Access rights
Administration	<ul style="list-style-type: none"> • <i>Login</i> • Manage Students • Manage Schedules • Manage Lecturers • Manage Users
Dosen	<ul style="list-style-type: none"> • Akses <i>login</i> • Manage Referrals
Mahasiswa	<ul style="list-style-type: none"> • Akses <i>Login</i> • Do Presence

Identification of Functional Requirements

1. Identification of functional requirements is a requirement that includes the processes that will be provided by the software.

Identification of Administrative Functional Needs

The identification of administrative functional requirements is described in Table 2.

Table 2. Identification of administrative functional requirements

No. Requirement	Name Requirement	Description
REQ-TU-01.	<i>Login</i>	Administration can login through the login page on the application.
REQ-TU-02.	Manage Students	Administration can manage all student data.
REQ-TU-03.	Manage Schedule	Administration can manage schedules.
REQ-TU-04.	Manage Lecturers	Administration can manage lecturer data.
REQ-TU-05.	Manage User	Administration can manage user data.

2. Identification of Lecturer Functional Needs

The identification of the functional needs of lecturers is explained in Table 3.

Table 3. Identification of the Functional Needs of Lecturers

No. Requirement	Name Requirement	Description
REQ-DSN-01.	<i>Login</i>	Lecturers can log in via the login page on the application.
REQ-DSN-02.	Manage Presensi	Lecturers can manage attendance

3. Identification of Student Functional Needs

The identification of student functional needs is described in Table 4.

Table 4. Identification of student functional needs

No. Requirement	Name Requirement	Description
REQ-MHS-01.	<i>Login</i>	Students can log in via the login page on the application.
REQ-MHS-02.	Manage Presence	Students can do Presence

System design aims to design the system so that it can do the job effectively and efficiently. The system design process is in the form of Unified Modeling Language (UML) modeling. Designing Use Case Diagram

The use case describes an interaction between one or more actors and the information system to be created, as described in Figure 3.



Figure 3. Use Case Diagram Design

Class Diagram Design

The relationship between classes and a detailed explanation of each class is in the design model of a system described in Figure 4.



Figure 4. Class Diagrams

Conclusion

Based on the results of the research described in the previous chapter, it can be concluded that the application was built specifically to make it easier for students to take attendance. The application was built to facilitate lecturers in checking student attendance. The application was built specifically by utilizing the haversine formula technology to increase security in the student attendance process. The system utilizes the tolerance distance to register attendance on the attendance system.

Acknowledgement

This study is supported by Universitas Langlangbuana, Bandung, Indonesia.

Reference

- Ehsan, A., Abuhaliqa, M. A. M., Catal, C., & Mishra, D. (2022). RESTful API testing methodologies: Rationale, challenges, and solution directions. *Applied Sciences*, 12(9), 4369. <https://doi.org/10.3390/app12094369>
- Chopde, N., & Nichat, M. (2013). Landmark-based shortest path detection by using A* Haversine. *International Journal of Innovative Research in Computer and Communication Engineering*, 1(2), 298–302.
- Indrawan, G., Gunadi, I. G. A., & Sandhiyasa, I. M. S. (2022). REST API and real-time notification of SISKA-NG mobile for the academic progress information system. In A. Kumar, R. Singh, & A. K. Sangaiah (Eds.), *Information and communication technology for competitive strategies (ICTCS 2020): ICT—Applications and social interfaces* (pp. 193–201). Springer Singapore. https://doi.org/10.1007/978-981-16-8494-4_18
- Istiyanto, J. E., & Widodo, A. P. (2009). Karakteristik metodologi penelitian bidang ilmu komputer (IK) berlandaskan pendekatan positivistik. *Jurnal Sains & Matematik (JSM)*, 17(2), 115–120.
- Liu, Y., Li, Y., Deng, G., Liu, Y., Wan, R., Wu, R., ... & Bao, M. (2022, May). Morest: Model-based RESTful API testing with execution feedback. In *Proceedings of the 44th International Conference on Software Engineering* (pp. 1406–1417). <https://doi.org/10.1145/3510003.3510087>
- Perkasa, M. I., & Setiawan, E. B. (2018). Pembangunan web service data masyarakat menggunakan REST API dengan access token. *Ultima Computing: Jurnal Sistem Komputer*, 10(1), 19–26. <https://doi.org/10.31937/sk.v10i1.662>
- Prihantoro, D., & Wahyuddin, M. (2022). Implementasi algoritma Haversine formula dan location-based service pada aplikasi pencarian lokasi bird contest berbasis Android. In *Prosiding Seminar Nasional Sistem Informasi dan Teknologi Informasi (SENSASI)* (pp. 663–671).
- Rahardian, R., Vinaya, I., & Diatmika, I. (2021). Aplikasi pencarian surfing spot terdekat di Bali menggunakan formula Haversine berbasis web. *Jurnal Inovasi Teknologi Informasi dan Aplikasi (JITeA)*, 1(1), 36–42.
- Rahmawati, A., & Rahman, A. (2011). Sistem pengamanan keaslian ijazah menggunakan QR code dan algoritma. *Jurnal Sistem Informasi Indonesia*, 1(2), 105–112.
- Sommerville, I. (2011). *Software engineering (rekayasa perangkat lunak)* (9th ed.). Jakarta: Erlangga.