



Development of a Website-Based Running Text Application for Kaputama STMIK

Arini Sabila^{1*}, Husnul Khair², Milli Alfhi Syari³

^{1,2}Informatics engineering, STMIK Kaputama

³Information System, STMIK Kaputama

Arnis5218@gmail.com^{1*}, Husnul.khair@gmail.com², milli.alfhisyari@yahoo.co.id³

Abstract

In today's digital era, fast and efficient information delivery is a must. One popular medium is running text, especially in educational institutions. This research aims to develop a website-based running text application for STMIK Kaputama to deliver information in real-time. The system was built using Internet of Things (IoT) technology, React.js on the frontend, and Hapi.js and MySQL on the backend. The implementation results show that this application is capable of dynamically updating information, is easy to use, and can be accessed from various devices. With JWT authentication and thorough system testing, this application is proven to be stable and secure. This research is expected to be a solution for efficient information digitization in the campus environment.

Keywords: *Hapi.js, IoT, Running Text, React.js, Website, STMIK Kaputama*

1. Introduction

The rapid development of information and communication technology in recent decades has driven various sectors to digitize work systems and information delivery. In education, particularly in higher education, fast, accurate, and efficient information delivery is a crucial element in supporting academic and administrative activities. One increasingly popular medium for this purpose is running text, a digitally displayed moving text, usually on an LED board. This medium allows information such as class schedules, activity announcements, or other important notifications to be conveyed in an engaging and easily updated manner.

However, despite its many advantages, the current running text system still has several limitations. Many institutions still rely on manual or semi-manual systems, where content configuration and updates can only be made by directly accessing the device. This is certainly difficult, especially in situations where information updates must be made quickly from different locations. This limitation hinders the dissemination of information that should be delivered instantly.

Given these challenges, a solution is needed in the form of a running text system that can be controlled remotely, is flexible, and is efficient. By leveraging Internet of Things (IoT) technology and web-based development, the running text system can be controlled through a responsive web interface that is easily accessible anytime and anywhere. This integration allows information to be displayed in real time without the need for manual reconfiguration.[1]

This research proposes the development of a web-based running text application at STMIK Kaputama, with the primary goal of simplifying information management by the campus. By utilizing technologies such as React.js, Node.js (Hapi.js), and an ESP8266 microcontroller connected to a P10 LED module, this system enables automated and secure information control and updates. This application not only prioritizes technical efficiency but also supports the campus's goal of providing modern, fast, cost-effective, and easy-to-operate information services.

Through this research, it is hoped that the developed system will become a practical digital solution that can be adopted by other educational institutions in the future.

2. Previous research

The purpose of this research is to determine the extent to which delay affects system performance and to provide recommendations for improving communication system performance. This research uses a development or Research and Development (R&D) method with a

focus on developing a more efficient P4 running text system on LRT, including efforts to minimize components in the communication system.[2]

2.1 Internet of Things (IoT)

The Internet of Things, also known as IoT, is a sophisticated technology that aims to expand and develop the benefits of continuous internet connectivity. Connecting surrounding objects to make daily activities easier and more efficient, significantly aiding all human endeavors. The importance of the Internet of Things can be seen in its increasing application in various aspects of life today. According to the RFID (Radio Frequency Identification) identification method, the term IoT is classified as a communication method, although it can also encompass other sensor technologies, wireless technology, or QR (Quick Response) codes. The term "Internet of Things" consists of two main parts: "Internet," which connects and regulates connectivity, and "Things," which refers to an object or device. Simply put, you have "Things" that can connect to each other to collect data and send it to the Internet. This data can also be accessed by other "Things." Certain "Things" have the ability to transmit data over a network wherever they are, without human-to-human or human-to-computer interaction.

At the beginning of the 21st century, the term "internet of things" began to be widely used by companies such as The Guardian, Forbes, and the Boston Globe. Interest in IoT technology continued to grow, leading to the First International Conference on the Internet of Things held in Switzerland in 2008, where participants from 23 countries discussed RFID, short-range wireless communications, and sensor networks. With the development of IoT that year, several examples of IoT emerged, including an internet-connected refrigerator introduced by LG Electronics in 2000, allowing users to shop online and make video calls. Another notable development was a small rabbit-shaped robot named Nabaztag, created in 2005, capable of delivering breaking news, weather forecasts, and stock market fluctuations.[3]

2.2 Running Text

Running Text is an electronic technique that displays moving or running text consisting of an arrangement of LEDs (Light Emitting Diodes), then connected in a matrix with a combination of LEDs between rows and columns. Running Text is a form of conveying information to the public with the help of LEDs. Running Text is an effective information and promotional media because it is able to attract attention, can create.[4]

2.3 Wi-Fi

Wi-Fi is short for Wireless Fidelity, which means a set of standards used for Wireless Local Area Networks (WLAN) based on the IEEE 802.11 specifications. WI-FI is the term given to wireless LAN systems that use the current 802.11 standard. The term WI-FI was created by an organization called the WI-FI alliance that works to test and certify WLAN devices. Wireless devices are tested based on their interoperability with other wireless devices that use the same standard. After being tested and passed, a device will be certified, meaning that this device can work well with other wireless devices that are also certified. Initially, WI-FI certification was only given to wireless devices that work on the 802.11b standard.[5]

2.4 Website

The definition of a website is a collection of pages in a domain that contains various information so that it can be read and viewed by users or internet users through a search engine. Information that can be loaded on a website usually contains image content, illustrations, videos, and text for various purposes. Generally, the initial display of a website can be accessed through the main page or also called the homepage by using a browser by entering and writing the URL address completely and correctly. On a main page or homepage, the website also contains several subsidiary website pages that are connected to one another.[6]

3. Research Methodology

A research method is a systematic series of steps used to collect, analyze, and interpret data. Research methods are used in various disciplines and can vary depending on the research objectives.

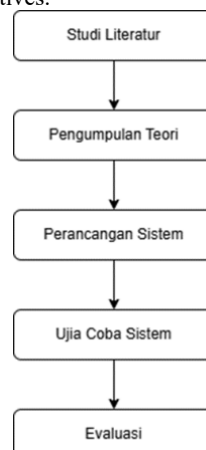


Fig. 1: Research Methodology Structure

1. Literature Study

Identifying and understanding problems based on references from relevant journals, books, and scientific articles.

2. Theory Collection

Collecting supporting theories related to the technologies used, such as React.js, Node.js (Hapi.js), MySQL, and IoT technologies like ESP8266 and LED Matrix.

3. System Role

Designing the overall system, from flowcharts, activity diagrams, class diagrams, to use case diagrams. This includes designing the interface and system structure.

4. System Testing

Conducting tests on the developed system to determine whether all features function as expected. This includes testing login, data input, and IoT integration.

5. System Evaluation

Evaluating the system ensures that the system functions as expected, based on the test results, and detecting deficiencies or bugs before the system is released.

2.1. Text font of entire document

Broadly speaking, this project is organized into two main folders: frontend and backend. The frontend folder contains various user interface (UI) components developed using React, while the backend folder contains the main program logic such as API creation, server configuration, routing settings (route handlers), and database connections. This structure aims to clearly separate the system's appearance and logic, thus facilitating the development, maintenance, and further development processes.

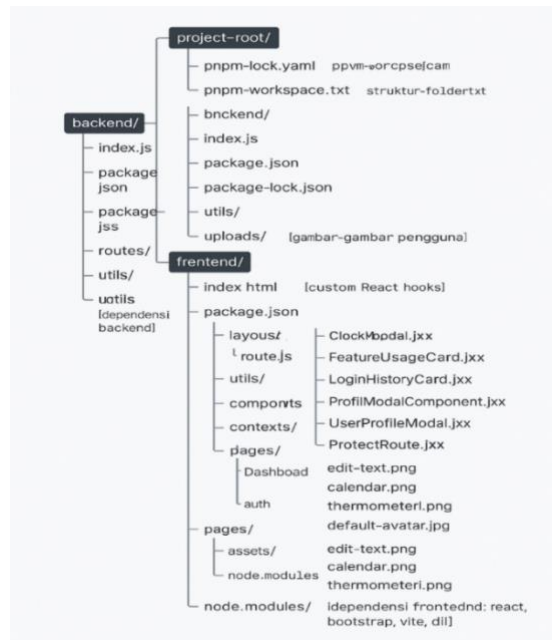
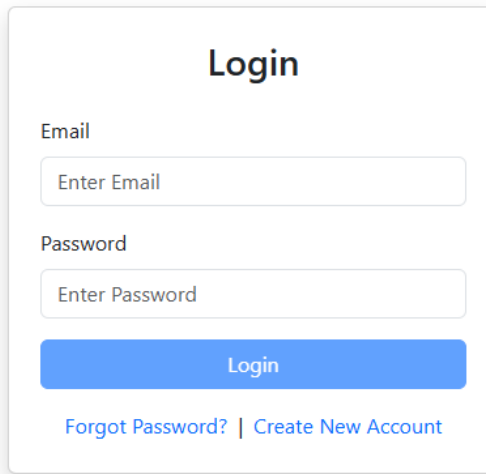


Fig. 2: Project Structure

3. Page style

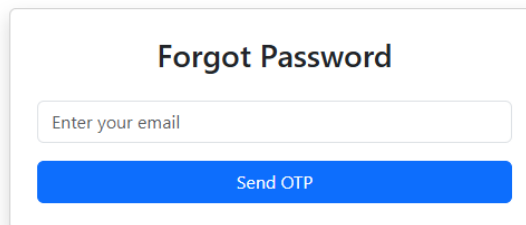
Users wishing to access the system must log in using the email address and password stored in the database. Upon successful login, a JWT token is provided for session authentication. Users can access the running text editing feature, edit the date and time, and update the temperature. Every change is sent to the backend via the API and recorded in the usage history.

If the user does not have an account, they must register by filling out several identity forms. Alternatively, if the user already has an account but has forgotten their username or password, they must reset their password by filling out the email form to receive an OTP code sent from the registered Google IoT admin..



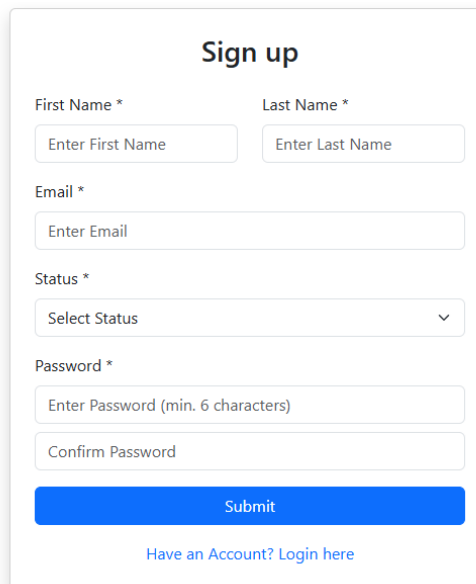
The Login page features a white background with a light gray border. At the top center is the title "Login" in bold black font. Below it are two input fields: "Email" with the placeholder "Enter Email" and "Password" with the placeholder "Enter Password". A blue "Login" button is positioned below the password field. At the bottom, there are two links: "Forgot Password?" and "Create New Account", separated by a vertical bar.

Fig. 3: Login Page



The Forgot Password page has a white background with a light gray border. The title "Forgot Password" is centered at the top in bold black font. Below the title is a single input field with the placeholder "Enter your email". A blue "Send OTP" button is located below the input field.

Fig. 4: Forgot Password Page



The Sign up page has a white background with a light gray border. The title "Sign up" is centered at the top in bold black font. The form consists of several fields: "First Name *" and "Last Name *" (each with a placeholder "Enter First Name" and "Enter Last Name"), "Email *" (placeholder "Enter Email"), "Status *" (a dropdown menu with "Select Status" and a downward arrow), "Password *" (placeholder "Enter Password (min. 6 characters)"), and "Confirm Password" (placeholder "Confirm Password"). A blue "Submit" button is at the bottom. Below the button is a link: "Have an Account? Login here".

Fig. 5: Account Registration Page

6. Mobile Dashboard

The dashboard display on mobile devices has a different style and layout compared to the desktop version. This mobile view automatically displays when the screen size is less than or equal to 789 pixels. The component structure in the mobile version is specifically designed to be more compact and efficient for use on small screens, while still providing an optimal user experience.

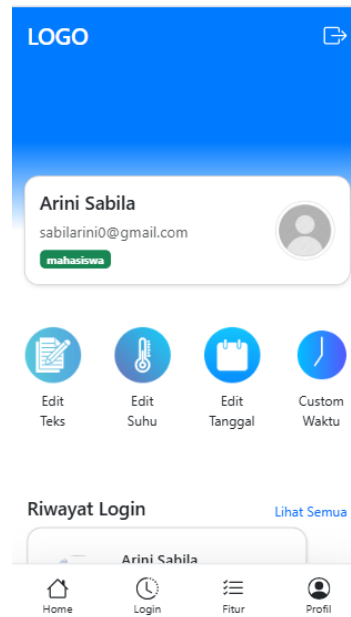


Fig. 6: Mobile Dashboard

On the mobile view, all components and features are displayed on a single page, and features have icons to highlight them to the user. The profile component is placed at the top center of the page and is designed as a card that displays the user's identity.

7. Desktop Dashboard

The dashboard display on desktop devices has a quite different style than the mobile dashboard. The desktop dashboard will automatically display when the screen size is larger than 789 pixels. The component structure on the desktop dashboard is also adjusted to the screen size, so the display remains fast and efficient.

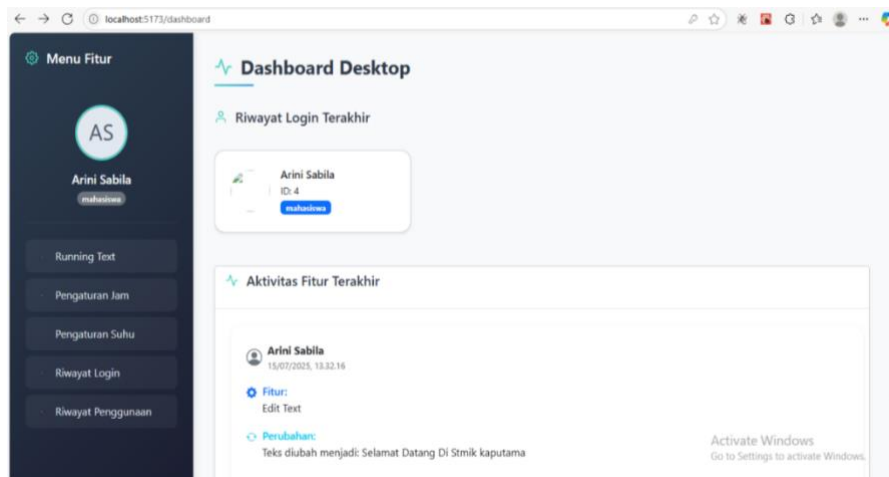


Fig. 7: Desktop Dashboard

Unlike the mobile view, on the desktop view the main components are separated into two sidebar components, the left sidebar to accommodate the profile and main features of the system, and the right side displays content, such as login history and usage history.

8. Features and Components

The Web IoT Control system is equipped with several key features designed to make it easier for users to manage and monitor IoT devices directly through a web interface.

1. Running Text

The first function is editing running text. This function is used to change the text displayed on the running text display on the IoT. It includes several forms, such as text input, saving text as a template, and using a template.

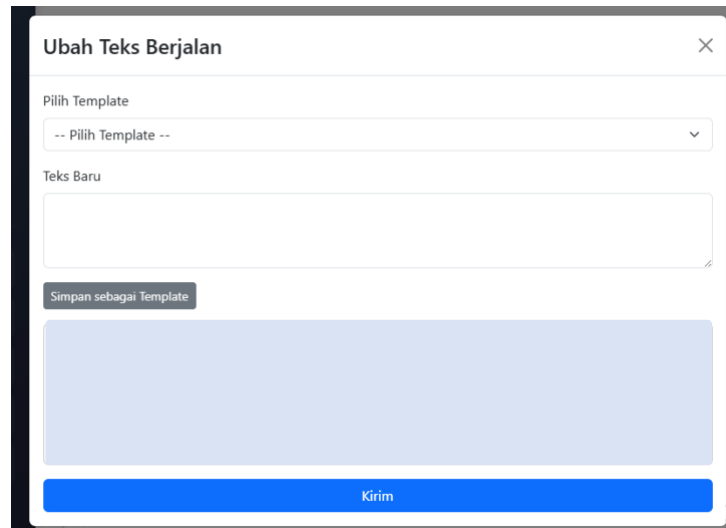


Fig. 8: Modal Running Text Edit Feature

2. Clock Settings

The next feature is the clock editing feature, which is used to set the time displayed in the running text. It has two control functions: automatic and manual. Automatic adjustments will adjust the clock to the current time, while manual adjustments will adjust the clock to the input time.

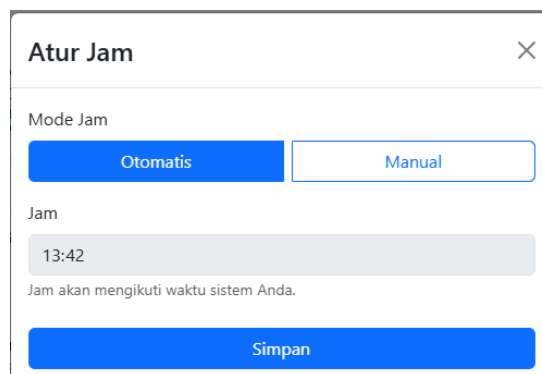


Fig. 9: Edit Automatic Clock



Fig. 11: Manual Clock Edit

3. Date Settings

Date settings are generally the same as clock settings. There are two input options: automatic and manual. If you select automatic, the date will be adjusted to the current date. If you select manual, the date will be displayed according to the input in the date input form.

Fig. 12: Automatic Date Edit

Fig. 13: Manual Date Editing

The main components that make up this system consist of several frontend and backend components. On the frontend side, there are components such as UserProfileModal, ClockModal, FeatureUsageCard, and LoginHistoryCard, each of which is responsible for managing user data, setting time, recording feature usage, and login history. Meanwhile, on the backend side, the main components include handlers, routes, and utils, which handle program logic such as processing API requests, JWT authorization, and connecting to databases and IoT devices. Each feature in the system is seamlessly integrated through a REST API, ensuring that any changes made by users are recorded in the database and can be accessed or forwarded to IoT devices for real-time display. This combination of features and components makes the Web IoT Control system not only functional but also flexible for further development.

4. Conclusion

The conclusions drawn from this research are as follows:

1. Successful development of a web-based running text application system has been successfully developed as a modern platform that enables STMIK Kaputama to manage information in real time, efficiently, and without relying on physical media or paid applications.
2. Based on the integration of IoT and Web technology, the system can utilize ESP8266 devices and LED Matrix devices that are optimally integrated with a web interface (React.js and Hapi.js), enabling remote control over a Wi-Fi network and supporting a variety of devices.
3. In terms of security, stability, and ease of use, features such as JWT authentication, data management with MySQL, and testing of text editing and time management functions demonstrate that this system is stable, secure, and easy to use as a solution for digitizing information on campus.

References

- [1] Syahputra, A. T., Fauzi, A., & Syari, M. A. (2024). Design and Build Running Text Controls Using IoT. *Journal of Artificial Intelligence and Engineering Applications (JAIEA)*, 4(1), 295–301.
- [2] Priandhika, V. (2024). Performance Analysis of Running Text Data Communication Network on LRT Trains in the Greater Jakarta Area. *Jakarta State Polytechnic*.
- [3] Febrina Saragih, G. (n.d.). Internet of Things (IoT)-Based Running Text Design
- [4] Widya, H., Alam, H., Wiguna, J., & Syafrawali, S. (2020). Design and Construction of an Arduino Uno-Based LED Running Text Prayer Time Schedule Display as an Information Media. *JET (Journal of Electrical Technology)*, 5(2), 61–67.
- [5] FIKRI, A. K. (2020). Potential Dangers of Using Public Wireless Fidelity (Wi-Fi) for Privacy Security. *Mercu Buana University, Jakarta*.
- [6] Ghiya Zahira. (2024). *Software Engineering Book*.
- [7] Atisina, S. (2024). Running Text Design Algorithm Using a Wifi-Based Huidu W02 V4.1 Controller. *Journal of Informatics and Applied Electrical Engineering*, 12(2).