

# Smart Street Light Monitoring and Visualization Platform for Campus Management

Somrudee Deepaisarn  
*Sirindhorn International  
Institute of Technology,  
Thammasat University*  
Pathum Thani, Thailand  
somrudee@siit.tu.ac.th  
Corresponding Author

Paphana Yiwsiw  
*Sirindhorn International  
Institute of Technology,  
Thammasat University*  
Pathum Thani, Thailand  
6222780379@g.siiit.tu.ac.th

Chanon Tantiwattanapaibul  
*Sirindhorn International  
Institute of Technology,  
Thammasat University*  
Pathum Thani, Thailand  
chanon.tanti@gmail.com

Suphachok Buaruk  
*Sirindhorn International Institute of Technology,  
Thammasat University*  
Pathum Thani, Thailand  
d6522300067@g.siiit.tu.ac.th

Virach Sornlertlamvanich  
*Faculty of Engineering, Thammasat University*  
Pathum Thani, Thailand  
*Faculty of Data Science, Musashino University*  
Tokyo, Japan  
virach@gmail.com

**Abstract**—As a recent trend in urbanization and intelligent technologies, smart lighting systems have been implemented in many major cities to support smart urban environments. This research developed a web application platform for data visualization and lighting device monitoring at Thammasat University, Rangsit Campus, Thailand. This implementation provides administrative and operative staff with an all-in-one platform through a convenient interface for monitoring, controlling, and collecting data from area devices and sensors. Platform development was divided into two sections: back-end application, providing application programming interface (API) endpoints, and front-end application, offering an interface for interacting with on-campus staff. Finally, the web application was deployed on a cloud platform so that responsible persons may access it on any device and acquire data in real time. Given the platform's capabilities, further data analytics may be proposed for building a smarter lighting system.

**Index Terms**—Smart City, Light, Data Visualization, Internet of Things (IoT), Web Application

## I. INTRODUCTION

Nowadays, people tend to migrate to highly dense population areas of the world. The number of people moving to such areas is predicted to account for above 60 percent of the world's population by 2050 [1]. The smart city concept is developed following the idea of the quality of life improvement and socio-economic growth of the society. The concept is based on the implementation of various smart devices such as sensors and the Internet of Things (IoT) [2] and [3]. Giffinger et al. [4] categorize smart cities into six aspects: smart environment, smart mobility, smart people, living, smart economy, and smart governance. In this work, the smart environment aspect is focused on the implementation of the smart street lighting system. There are some previous attempts for smart street light projects in a number of major cities. For example,

the Amsterdam Smart City (ASC) project contains smart lighting devices which are capable of controlling remotely using sensors to adjust for the surrounding environments presented in the area, including weather conditions and pedestrian flow control. Their implementation of the Smart Lighting system is mainly focused on the improvement of energy saving to reduce carbon emissions [5]. Another example can be seen in Barcelona's solution to the Smart Street Lighting System. Their implementation includes an LED lighting system to promote energy efficiency and cost savings [6]. The other advantages presented by the smart street lighting system are the reduction in crime rate, the promotion of life security, and quality of life improvement [7].

The management and data visualization platform is essential to the success of the smart street light system. The system establishment is required to maintain and monitor devices installed and connected in the area. Barcelona, for instance, developed the Application Programming Interface (API) to communicate with the smart lighting management system and other applications such as traffic management to exchange data across the platform [8]. It also integrated the data collection from sensors, such as meteorological data, to increase the knowledge of the city, creating an intelligent system [9].

According to the successful adoption of the smart city concept in many major cities, one of the main strategies is gaining collaboration between the public and the private sectors [10]. Therefore, we applied the strategic plan to implement the smart street light system under the smart city project at Thammasat University, Rangsit Campus, Thailand. The 167 smart light poles are installed throughout the campus with a pole-to-pole separation of approximately 20 meters in order to meet the general regulation of street lights in Thailand. The smart light devices are equipped with light-emitting diode



Fig. 1. Map of Thammasat University, Rangsit Campus, Thailand; latitude/longitude location: 14°04'27"N 100°36'08"E.

(LED) lamps with adjustable dimming levels, of which the maximum power is 120 Watts. We aim to figure out the optimum usage of the smart light system so that they shine at appropriate levels when natural light is insufficient, i.e., in the evening, throughout the night, and early in the morning. In our campus scenario, the system benefits both driving vehicles and pedestrians. For that, ensuring proper illumination also depends on a reliable management and maintenance system that can enhance the system's usability. Hence, we propose a monitoring system with data visualization and device control capabilities such that the campus staffs are able to monitor and maintain the smart street lighting. Moreover, the platform offers a simple data collection solution for environmental and power usage data, which can be utilized in further analytic tasks and research related to the smart street light system for our campus management. The development of this platform, which includes system monitoring and data collection, extends the current infrastructure and moves toward optimal energy consumption with a smart campus concept, in accordance with Thammasat University's sustainable development agenda.

This paper is outlined as follows. Section II elaborates on the infrastructure and system requirements, including physical devices, its external management platform, and requirements for developing the web application. Section III presents the operation of data from the external management platform. The front-end application, including the user interface (UI) and data visualization, is demonstrated in Section IV. Section V discusses the contribution of the work, especially the web application platform design, and its usability. Lastly, the conclusion and future work are summarized in Section VI.

## II. INFRASTRUCTURE AND SYSTEM REQUIREMENTS

Smart street lights and environmental sensors (MinebeaMitsumi, Inc., Japan) were installed at Thammasat University, Rangsit Campus, Thailand. The 167 controllable lighting devices, along with an illuminance sensor attached to each lighting device. There are three gateways to connect devices to the external control system. An environmental station composed of weather and light sensors was also installed

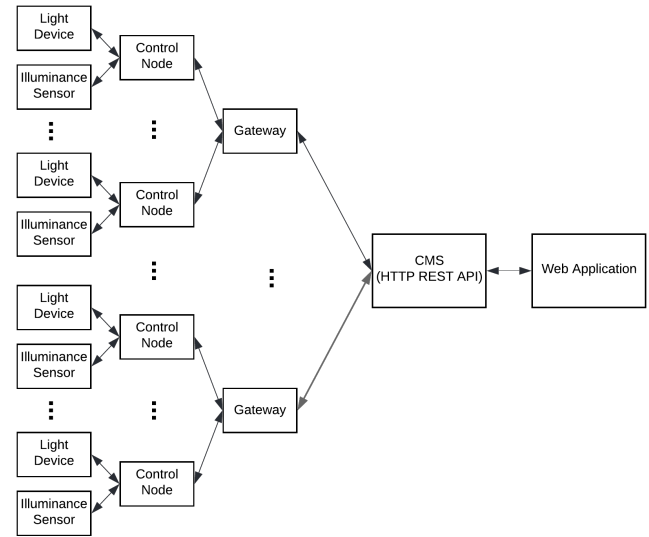


Fig. 2. The diagram of the connection between lighting devices, control node, CMS API, and the web application: Each control node is associated with a lighting device and an illuminance sensor. A control node is connected to CMS API via a gateway, and the web application is connected to the lighting system using CMS API.

in the region. To maximize the efficiency of controlling and managing these devices, They were grouped into six zones based mainly on the road within the campus, namely Prachasanti, Sanya-Thammasak, Talad-Wicha, Yung Thong, and Pithaktham roads.

Smart LED light devices are connected to the external platform, the CMS Neptune - SC-v6.0.3 (Paradox Engineering, Switzerland), operated by the lighting product manufacturer. Provided that the API can be called via the Representational State Transfer (REST) on Hypertext Transfer Protocol (HTTP), we are able to collect data and manage devices locally. The devices and the CMS API are connected as illustrated in Fig. 2. Therefore, the back-end part of our system must be capable of sending HTTP requests and receiving responses from the REST API that CMS provided. The system must also be capable of sending data for visualization on the front end to represent the acquired data and monitor the status of devices. The front-end part is presented as a web application that can be accessed with any web browser and is compatible with various computer/mobile operating systems. The web application has shown the benefit of being accessible from anywhere at any time, given that the devices are connected to the internet, as shown in Fig. 2.

## III. BACK-END SYSTEM DEVELOPMENT

In order to develop the web application, it is necessary to establish the back-end or the server-side and the front-end or the client side of the application. For web application creation, node.js, a JavaScript runtime environment, is used. This method is commonly used in web application development where JavaScript runs the program on both ends [11]. On top

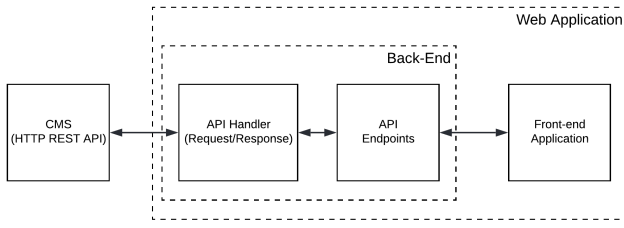


Fig. 3. The connection between CMS API and the back-end processing of the web application. On the back-end of the web application, an API endpoint will receive an HTTP request from the front-end application. Then, the API handler will process that request and send it to CMS API. After the data from CMS API is processed, a response will send back to the front end.

of node.js, the Express.js web framework module is chosen due to its highly configurable nature, thus allowing for more customization on the web application development [12]. Both node.js and Express.js are built using event-driven approaches. One of the benefits that Express.js offers is the reduction of code redundancy because the same code is used and written repeatedly to do the same task, such as cookie and URL parsing.

API endpoints on the back-end application are required to connect with the CMS API. These endpoints are built using the Express.js routing function. However, the Axios module [13] is responsible for sending HTTP requests and receiving responses to allow the back-end side application connected to the CMS API to obtain data and control devices. Authentication is also involved in connecting to the CMS API and complying with their security requirements. Bearer token must be appended to the HTTP request headers when sending requests to the CMS. Thus, the back-end side of the application must be authenticated with the CMS server first, and then the requests are sent afterwards. Cookies are used to store user tokens obtained from CMS to improve the authentication performance of the application.

After sending HTTP requests to the CMS in the back-end application, the HTTP response is received, which contains the requested data, whether from lighting devices or environmental sensors. Then, the data are repackaged and rearranged in an easy way to display and implement on the front-end side. For example, the data from the environmental sensors are rearranged by timestamp, and the measured values are converted to standard units. Finally, the data are sent as an HTTP response to the request in JSON format. Other data that support the monitoring system, such as the device information and their status, are also provided as an API endpoint and later shown on the front-end part. The process in the back-end application is illustrated in Fig. 3.

#### IV. FRONT-END USER APPLICATION DEVELOPMENT

A web application is designed to suit the front-end user. The web application has the advantage of accessibility using the internet connection or cellular network on HTTP protocol; see details in [14]. However, the web application should be

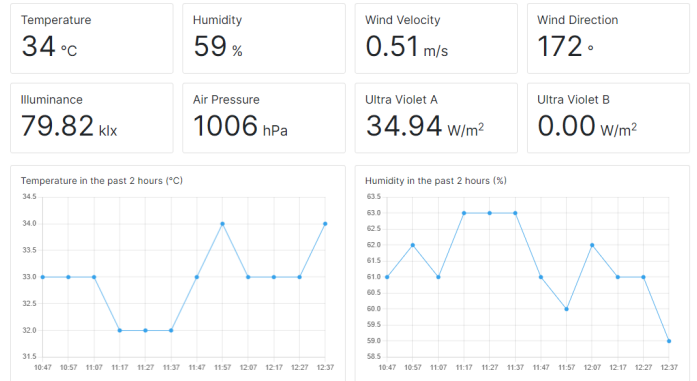


Fig. 4. The dashboard interface used in the web application: The displayed data obtained from environmental sensors include temperature, humidity, illuminance, etc. The data are presented in both numerical and graphical formats.

able to adapt and behave the same across all devices, ranging from mobile to desktop devices. Therefore, responsive web application design is implemented to allow for a fast and optimized method of rendering web pages on devices with different screen sizes, thus establishing a good user experience on mobile devices, see details in Almeida et al. [15]. The tool used in the responsive web design is Bootstrap, oriented around mobile devices first and later scales up to larger screen sizes as described in [16].

To represent data in the dashboard, Chart.js, an open source JavaScript library under MIT license, is used in data visualization, which provides a wide range of customization [17]. The web application used Chart.js to show data from environmental sensors as line graphs. Examples of data shown in the visualization are temperature and humidity, which are essential in monitoring the smart lighting system, as shown in Fig. 4. In Fig. 4, the numerical data shows only the last entry from the environmental sensor, updated every 10 minutes, and line graphs show the data from the recent 2 hours. Other sensors are illuminance, Ultra Violet A and B indices, wind velocity, wind direction, and air pressure. The data from the environmental sensors are obtained via a provided API endpoint that had been set up earlier in Section III.

Device location on an interactive map is plotted using Leaflet.js [18]. In addition, the device location map also allows the users to see the device information in real time. The precise locations of all devices can also be obtained via API endpoint and later rendered as markers on a device location map, as indicated in Fig. 5 to enhance the representation of those devices on the platform. All front-end and back-end development are combined into a platform, and the web application is tested and deployed on Microsoft Azure App Service. This cloud deployment suit the current implementation because it supports Node.js applications with a simple deployment process [19]. Once deployed, the public URL is created and used to access the web application to monitor the smart lighting system, look up, and acquired the data from the environmental sensors at any time.

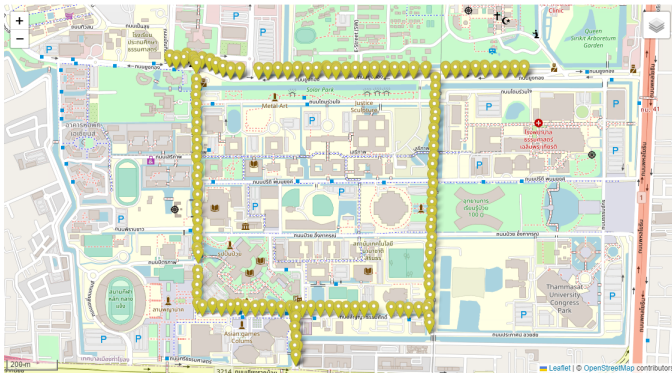


Fig. 5. The device map showing the interactive location of the 167 smart street lights on the campus.

## V. DISCUSSION

From our real use cases elaborated in the previous sections, the web application is developed in response to data visualization and device monitoring needs at Thammasat University, Rangsit Campus, Thailand. This web application assists the system supervisors with an easy-to-use interface. It also allows the real-time acquisition of related data from the environmental sensor and monitoring devices. In addition to the lighting parameters, which are illuminance and Ultra Violet A and B indices, other environmental parameters are observed in our location-specific area within the campus. These parameters, temperature, and humidity, correlate well with the light-related parameters, according to our preliminary observations. Hence, we paid attention to creating a system that would allow us to acquire these environmental values for further analytics of data. On the contrary, the process of collecting the environmental and power usage data would have been complicated without the proposed system. Our system was designed for those data to be collected with minimized human efforts. In addition, the easy-to-use monitoring and controlling interface makes street light management convenient for operative staff and other relevant stakeholders. In the current system implementation, system instability exists related to the electrical grid connected to the campus, resulting in data loss recorded by the system. Electrical blackouts and power drops are the main concerns that frequently happen, especially during the rainy season, which affects devices on the campus. Therefore, the web application can be improved by integrating the detection system when one or more devices are disconnected from the grid and later notify staff for maintenance. Responding to such problems properly as soon as they become known will promote the reliability of the installed lighting devices on the campus and efficiency in monitoring devices. Hence, the improvement in the acquired data can then be achieved as a consequence. Some gaps in data are the remaining challenges to fulfill. These require understanding those environmental quantities and their associations, support theory, and proper application of statistics to figure out the features of interest for an attempt at quantitatively predicting natural light. Furthermore, our next step is utilizing this smart street light monitoring and visu-

alization platform for campus management toward reducing energy consumption and promoting sustainability. Designing a model for automatically adjusting the lighting system on campus would create an intelligent solution for the outdoor environment which is beneficial to those hanging around on campus, those responsible for system administration, and those who are policymakers.

## VI. CONCLUSION AND FUTURE WORK

In conclusion, the web application is created for monitoring the smart lighting devices installed at Thammasat University, Rangsit Campus, Thailand. The back-end application provides API endpoints to send data from an external platform to the front-end application. The front-end application is designed using a responsive web design approach, implemented using the Bootstrap CSS framework. The device location map using Leaflet.js and data visualization from the environmental sensor using Chart.js are also available in the front-end application. This web application has finally deployed to the cloud using Microsoft Azure App Service and can access the application using its generated public URL. In the future, this web application can also implement the data analytics feature and create a prediction model for further reduction in energy consumption and promote sustainability.

## ACKNOWLEDGMENT

The authors gratefully acknowledge the financial support provided by the Thammasat University Research fund under the TSRI, Contract No. TUFF19/2564 and TUFF24/2565, for the project of “AI Ready City Networking in RUN”, based on the RUN Digital Cluster collaboration scheme.

## REFERENCES

- [1] G. C. Delgado-Ramos, “Water and the political ecology of urban metabolism: the case of Mexico city,” *Journal of Political Ecology*, vol. 22, no. 1, pp. 98–114, 2015.
- [2] T. M. Chen, “Smart grids, smart cities need better networks [editor’s note],” *IEEE Network*, vol. 24, no. 2, pp. 2–3, 2010.
- [3] L. G. Cretu, “Smart Cities Design using Event-driven Paradigm and Semantic Web,” *Informatica Economica*, vol. 16, no. 4, pp. 57–67, 2012.
- [4] R. Giffinger, C. Fertner, H. Kramar, and E. Meijers, “City-ranking of European medium-sized cities,” *Cent. Reg. Sci.*, pp. 1–12, Jan 2007.
- [5] S. Madakam and R. Ramachandran, “Amsterdam smart city (ASC): fishing village to sustainable city,” in *WIT Transactions on Ecology and the Environment*, WIT Press, Jul 2016.
- [6] Zigurat Global Institute of Technology, “Barcelona smart city: Most remarkable example of implementation,” <https://www.e-zigurat.com/blog/en/smart-city-barcelona-experience/>, Jul 2021.
- [7] D. P. Farrington and B. C. Welsh, “Improved street lighting and crime prevention,” *Justice Quarterly*, vol. 19, no. 2, pp. 313–342, 2002.
- [8] E. Pejstrup and J. Egenolf, “Integrated infrastructures factsheets,” [https://grow-smarter.eu/fileadmin/editor-upload/Smart/Action\\_Area\\_2-compressed.pdf](https://grow-smarter.eu/fileadmin/editor-upload/Smart/Action_Area_2-compressed.pdf), Nov 2019.
- [9] G. M. Lloide, “A concise review of fifteen intelligent street lighting systems which can pave the way for the emergence of smart cities in the Caribbean,” *i-manager’s Journal on Future Engineering and Technology*, vol. 15, no. 4, p. 28, 2020.
- [10] D. Das, “Smart city,” *The Wiley Blackwell Encyclopedia of Urban and Regional Studies*, pp. 1–7, 2019.
- [11] N. Chhetri, *A Comparative Analysis of Node.js (Server-Side JavaScript)*. Master’s thesis, St. Cloud State University, 2016.
- [12] A. Mardan, “Using express.js to create node.js web apps,” *Practical Node.js*, pp. 51–87, 2018.
- [13] “Axios,” <https://axios-http.com/>, 2022.



- [14] N. Dissanayake and K. Dias, "Web-based applications: Extending the general perspective of the service of web," in *10th International Research Conference of KDU (KDU-IRC 2017) on Changing Dynamics in the Global Environment: Challenges and Opportunities*, 2017.
- [15] F. Almeida and J. Monteiro, "The role of responsive design in web development," *Webology*, vol. 14, no. 2, pp. 48–65, 2017.
- [16] "Get started with bootstrap." <https://getbootstrap.com/docs/5.2/getting-started/introduction/>, 2022.
- [17] H. M. Millqvist and N. Bolin, *A comparison of performance and scalability of chart generation for Javascript data visualisation libraries : A comparative experiment on Chart.js, ApexCharts, Billboard, and ToastUI*. Bachelor's thesis, University of Skövde, 2022.
- [18] V. Agafonkin, "Leaflet - an open-source Javascript library for interactive maps." <https://leafletjs.com/>, 2022.
- [19] V. P. Desai, K. S. Oza, P. P. Shinde, and P. G. Naik, "Microsoft azure: Cloud platform for application service deployment," *International Journal of Scientific Research in Multidisciplinary Studies*, vol. 7, no. 10, pp. 20–23, 2021.