Research on Library Facilities Monitoring and Warning System Based on IoT Technology

Dongping Ma[1], Jianping Du[1*], and Qin Yang[2]

[1] Library of Beijing Union University, Beijing 100101, China.
[2] The Library and Information Center of Chengdu College of Arts and Sciences, Chengdu 610401, China.

* Corresponding author. Email: hgtdongping@buu.edu.cn
Manuscript submitted March 5, 2025; revised April 26, 2025; accepted April 28, 2025; published June 11, 2025.
doi: 10.17706/jsw.20.1.37-50

Abstract: To enhance the utilization efficiency of library self-service equipment and minimize manual intervention, this study proposes a systematic framework for operational status monitoring and automated alarm issuance. Leveraging technical methodologies, the system continuously tracks the operational status of network-connected self-service facilities, including online connectivity status, printer consumable levels, and operational application software functionality. The monitoring center serves as the central control hub, conducting scheduled network device diagnostics to ensure seamless normal operation. A Quick Response (QR)-code-based repair reporting mechanism is established, enabling readers to efficiently report equipment issues via code scanning, with repair requests directly transmitted to librarian workstations. Collected monitoring data and user-submitted repair information undergo systematic screening and processing to generate actionable alerts, which are delivered to designated terminals as pop-up notifications for immediate attention. By integrating proactive equipment monitoring, user repair reporting, and automated alerting, the proposed system swiftly addresses operational disruptions, thereby improving equipment availability, reducing staff workload, and elevating the intelligent service capabilities of libraries. This approach demonstrates a practical solution to optimize self-service facility management, enhancing user experience and advancing institutional service efficiency through technology-driven interventions.

Keywords: status monitoring, reader-reported repairs, information processing

1. Introduction

With the rapid development of information technology, libraries are undergoing a profound transformation toward digitalization and intelligent transformation. The integration of Internet of Things (IoT) technology has notably elevated library service capabilities [1–15]. First, Radio Frequency Identification (RFID) streamlines operations like self-service book borrowing/return and automatic sorting, drastically improving circulation efficiency. Libraries equipped with RFID systems enable significantly faster book borrowing and returning than traditional manual methods, reducing human error and accelerating inventory management [2, 6, 11]. Second, IoT technology enables intelligent environmental control through real-time occupancy monitoring and adaptive lighting system integration. By dynamically adjusting illumination levels based on detected human presence and activity patterns, this integrated approach optimizes energy usage while maintaining operational efficiency [3, 4]. De Sarkar's analysis underscores IoT's distinctive advantages in extending internet connectivity boundaries and powering machine learning-driven workflows. Using Manchester's smart city initiative as a case in point, the study elucidates the operational mechanisms through which IoT achieves real-time responsiveness via sensor networks and big data analytics [5]. Zhou proposes an IoT-enabled innovative framework incorporating the Clustered Centered Routing Cryptography Scheme (CCRCS) to optimize digital library services through real-time data collection and efficient secure routing, enabling user behavior analysis, improved resource accessibility, and enhanced personalized experiences [1]. Hakim has developed an IoT-based smart library attendance system utilizing barcode technology, which achieves efficient attendance tracking, real-time data management and optimized resource allocation through Raspberry Pi hardware and customized software, significantly enhancing library operational efficiency and user experience [9]. At a systemic level, IoT constructs a seamless sensor network that enables real-time monitoring of books, equipment, and patron interactions across the library space. This interconnected infrastructure fosters deep integration between "people, books, and environment," shrinking the service response times, improving resource utilization rate, and labor costs are reduced significantly through automated workflows. Now, self-service equipment has emerged as a critical component of modern library services, delivering more convenient and efficient experiences for readers. These systems enable functionalities such as Self-service book borrowing and returning, Self-service document copying/printing, and seat reservation—all designed to streamline user interactions. However, the reliable operation of these technologies is pivotal to maintaining service quality, unaddressed malfunctions or downtime can cause significant inconvenience for patrons and potentially disrupt the library's daily operations.

Thus, establishing a monitoring and warning system for library facilities is critical. Such a system not only improves the utilization rates of self-service equipment by ensuring the reliable operation, but also reduces staff workload through intelligent monitoring and automated information processing, and ultimately elevating the library's overall service standards. This paper presents a study on developing a library facilities monitoring and warning system based on Internet of Things technology, aiming to devise a more scientific and efficient approach to facility management.

To ensure the stable and reliable operation of these unattended electronic devices, it is essential to analyze their characteristics and design targeted monitoring strategies. As the majority of equipment failures result from network outages, regularly monitoring the network status of these devices and triggering an alarm as soon as a device goes offline can be an effective solution. When the operational application software in the device malfunctions or freezes during operation, installing a monitoring tool within the device can be a practical solution. This tool will continuously track the software's status and automatically initiate a restart upon detecting any irregularities, enabling the software to quickly resume normal operation. This enables devices to either automatically recover from minor malfunctions or trigger immediate alerts when issues arise. For those devices lacking autonomous recovery or alerting capabilities, readers can report anomalies directly via their mobile phones. This integrated approach not only strengthens the devices' self-diagnostic and self-healing capacities but also establishes a robust mechanism for timely fault notification. As a result, librarians can quickly identify and resole problems, significantly improving the effective utilization rates of the devices and enhancing reader satisfaction with library infrastructure.

Constructing a monitoring system to oversee the status of unattended library equipment enables the prompt detection of malfunctions, allowing for either automated or manual intervention and repairs. This not only boosts the equipment's effective utilization rate but also elevates the reading experience for patrons and enhances the library's intelligent management capabilities.

2. System Composition and Working Mechanism

All electronic service facilities in the library are included in the monitoring scope, such as book retrieval terminals, self-check-in/check-out stations, self-service document printers/copiers seat reservation systems, digital newspaper kiosks, storage lockers, access control units, and digital signage. The system architecture, illustrated in Fig. 1, centers on a management platform that proactively performs status diagnostics for unattended devices and receives repair requests sent by readers.

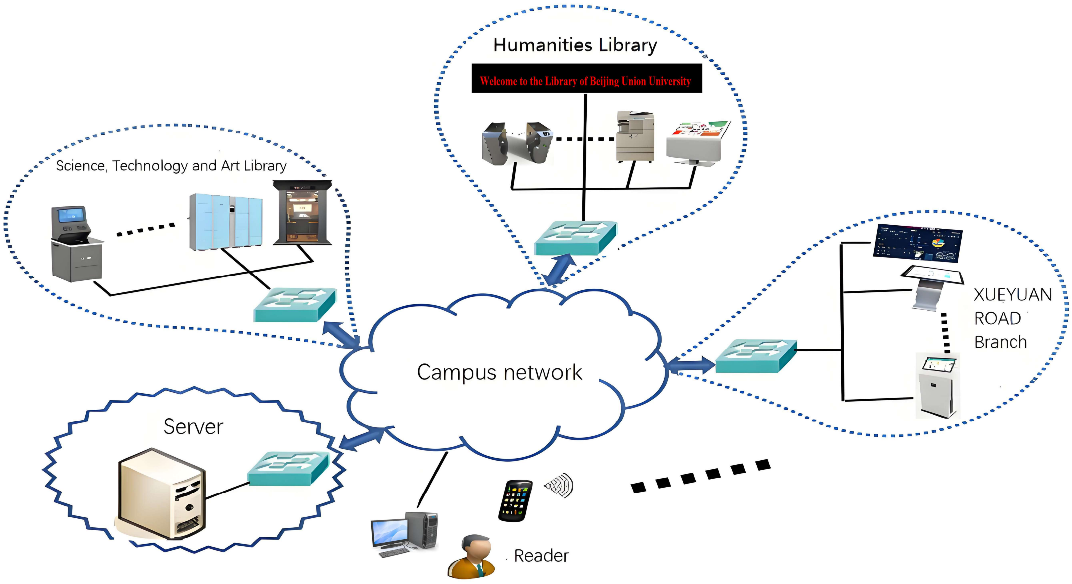


Fig. 1. Composition of the Monitoring System.

Each monitored device is assigned a unique device number (ID). The system regularly monitors device status, including network connectivity and consumable levels (e.g., paper/ink levels) for document printers and copiers. When an abnormal status is detected—such as a network outage or low consumables—the monitoring center immediately flags the issue, and a pop-up warning is displayed on the workstations at the library service desktop or relevant librarian terminals.

For devices controlled by a computer or industrial computer (such as retrieval machines, self-service book borrowing and returning devices, etc.), monitoring software can be installed into their systems to monitor the status of the operational application software, facilitating timely fault recovery or alarm. Various device controllers run specialized operational software or interface programs that facilitate device functionality and Human-Machine Interaction (HMI).

In some cases, device malfunctions that disrupt service may originate not from the controller or operational application software, but from server-side failures or network communication anomalies. Readers will be prompted with a failure during operation. To obtain warnings for such problems as soon as possible, a repair QR code for this device can be posted in a prominent position of the device. By scanning the code with their mobile phones, readers can submit a repair request directly. The alert will be immediately sent to the monitoring center and then distributed to the workstations of responsible staff or service desks. The alarms include detailed information such as device type, location, fault description, etc., which facilitates librarians to remotely diagnose or address issues on-site in a timely manner. The monitoring application software installed on the server performs status surveillance and alert management for self-service devices according to predefined operational workflows. As shown in Fig. 2, the system's core operational modules consist of: Online Status Monitoring, Device-Side Fault Detection, Alert Generation & Processing, Fault Response Protocol, Comprehensive Logging. Detailed explanations of each functional module will be provided in subsequent sections.

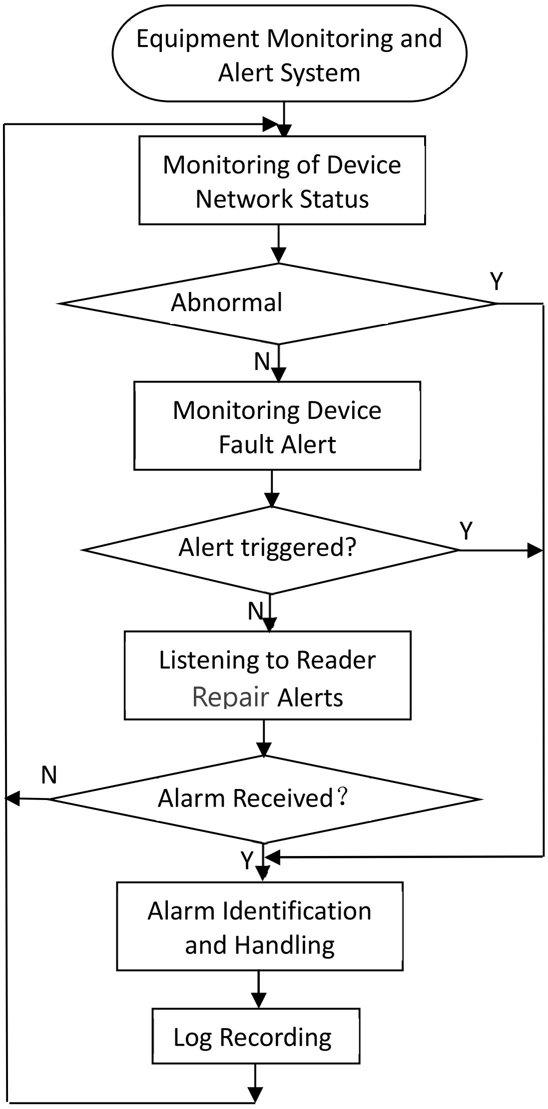


Fig. 2. Library equipment monitoring & repair request workflow.

3. Monitoring of Device Network Status

Most of the self-service devices in libraries require network connectivity to function properly. Once the network connection is broken, they cannot provide services. To ensure the stable and reliable operation of these devices, monitoring their network connectivity status is essential.

In a library's local area network, devices are assigned static IP addresses. Usually, to determine whether a device is online, the PING command can be executed via the command line. This command is used to check whether the local host can successfully exchange (send and receive) data packets with another device. The returned information indicates network connectivity. If the device is online, the PING command will be successful, while a failed PING signals an offline status.

Based on this principle, the monitoring center can automate the process of sending PING commands via software, receive response data, and analyze it to determine the online status of the target device. The primary C# code implementation is as follows:

```
Process ping = new Process();   //Create a process
ping.StartInfo.FileName = "cmd.exe";//Set the command name
ping.StartInfo.CreateNoWindow = true;//Set the command window not to be displayed
ping.Start(); //Start the Command Prompt (CMD)
```

```
ping.StandardInput.WriteLine("ping " + destIP); //Execute the ping command, where destIP is the IP address of the target device
ping.StandardInput.WriteLine("exit"); //Exit the Command Prompt (CMD)
string strrst = p.StandardOutput.ReadToEnd(); //Get the returned information
```

The returned information mainly falls into the following situations:

1. "Destination host unreachable";
2. "Request timed out";
3. "Unknown host";
4. "bytes=** time=** TTL=**";

If the feedback results match the first three cases, the ping is considered failed. The fourth case indicates that the network connectivity of the target device is unobstructed.

To avoid misjudgments from unstable networks, the system retries the PING command for devices that fail to respond on the first attempt. If three consecutive PING failures occur, the device is considered as offline, and an alarm needs to be issued to notify the librarian for maintenance.

This approach applies equally to the monitoring of devices in a power-off or frozen state. In both scenarios, a failed PING indicates the device cannot provide normal services, and the librarians should be prompted for resolution.

4. Consumable Monitoring for Document Printers

Document printers in the library are among the most frequently used self - service devices. To ensure their continuous and stable operation, it is essential to monitor the status of their consumables. The main consumables in a document printer typically include toner cartridges, ink cartridges, and paper.

The consumables for the general - purpose printers and copying copiers include A4 paper, A3 paper, black toner, red toner, yellow toner, blue toner, etc. (Color printers may have toner in other colors). There are generally three statuses of the paper tray: "ready", "low" and "empty". Toner levels are displayed as a percentage. When the paper tray is detected in the "low" or "empty" status, or when any toner's remaining quantity is less than 5%, an alert will trigger to prompt librarians to replenish supplies promptly, preventing disruptions to printer operations.

The monitoring center initiates the command to retrieve this web page's source code. It can be achieved using either the WebClient or WebRequest class.

Below is a method for reading the web page source code via the WebClient class:

```
string pageHtml = "";// Define a variable for the character sequence of the page source code
WebClient MyWebClient = new WebClient();//Create a WebClient
MyWebClient.Credentials = CredentialCache.DefaultCredentials;// Get or set the network credentials used to authenticate requests to Internet resources.
pageHtml = MyWebClient.DownloadString(url); // Download data from the website that queries consumables. The URL is the address of the webpage where the consumables data is displayed.
```

`pageHtml` is the full source code of the downloaded webpage, from which the consumable data needs to be extracted. The process involves analyzing source code patterns and leveraging keyword markers surrounding target data.

Extraction Workflow (Example: A4 Paper Tray Status):

Step 1: Locate Data Block:

Use the header marker <TD>Paper Tray 1</TD></TR> to identify the A4 tray's status section:

```
int a4HeaderPos = pageHtml.IndexOf("<TD>Paper Tray 1</TD></TR>"); // Locate section start
```

Step 2: Extract Status Value:

Find the subsequent
</TD></TR> closing marker and parse the substring between markers. The value following "Paper Status:" within this block directly indicates A4 paper remaining.

For A3 paper and toner levels (black/red/yellow/blue), repeat the above workflow: First, locate the data packet of this consumable description, and then extract the status string between header and closing markers. Skip this parsing step entirely for printers that fail ICMP Ping checks (network-unreachable devices), avoiding unnecessary processing.

5. Monitoring of Operational Application Software

For devices such as self-service loan/return systems and seat reservation systems that use industrial control computers as controllers, manufacturers usually deploy dedicated software to support relevant operations on these devices. During prolonged operation, the software may encounter runtime errors or even terminate unexpectedly, causing service disruptions. By installing a monitoring module on the controller to track the status of such operational application software, anomalies can be detected in real time and appropriate interventions can be made. This can minimize downtime and significantly extends the effective uptime of the devices.

The monitoring software does not alter the operational workflows or configurations of the manufacturers' business applications, but it can control the software to terminate and restart. Generally, when a local operational application software fails and terminates unexpectedly, triggering an automatic software restart or rebooting the device itself can restore normal service. Therefore, the processes of the operational application software need to be continuously monitored. When the "abnormal exit" phenomenon is detected, the monitoring software should initiate a restart procedure.

```
MyProcesses = Process.GetProcessesByName("Software Name"); //Define the names of the operational application software to be monitored.
MyProcesses.Exited += new EventHandler(myprocess_Exited); //When it is detected that the program is closed, trigger the method `myprocess_Exited`.
In the service program of myprocess_Exited, execute:
Process.Start("Path\\Software Name"); //Launch the operational application software
```

If the operational application software is restarted three times within a 10-minutes monitoring window, the device is considered as malfunctioning. An alarm will be actively sent to the monitoring center to notify librarians for inspection, and the device will be automatically restarted. In typical scenarios, normal functionality is restored after the reboot. This self-diagnostic approach enables the device to identify recurring issues autonomously, significantly improving operational uptime. The monitoring software runs hidden in the controller's background and will not affect the normal operation of the operational application software.

6. Readers Can Report Issues by Scanning the QR Code

When readers encounter device malfunctions—that the monitoring center fails to detect or cannot actively be informed of, they can use their mobile phones to scan the repair-reporting QR code posted or displayed on the device. Scanning the code will direct them to the repair-reporting system, where they can submit a repair request. To implement this function, a unique repair-reporting QR code needs to be generated for each device. Additionally, the IIS (Internet Information Services) on the server and deploy a website that supports online repair requests. IIS is a flexible, scalable, and high-performance web server software developed by Microsoft for Windows Server. It facilitates the hosting and management of websites, web applications, and services over HTTP/HTTPS protocols.

6.1. Device Repair-Reporting QR Code

Each device's repair-reporting QR code is uniquely encoded with a URL containing its device ID, structured as:
http://SERVERIP?ID=****. The device ID implicitly contains the venue code of the device, the device's number within the venue, and the device type.
Device ID Composition (Format: [VenueCode][DeviceSequence][TypeCode]):
Venue Code: 1 digit identifier for the location (e.g., "1" = Humanities Hall)
Device Sequence: 2–3 digit number for the device's position within the venue (e.g., "08" = 8th device).
Type Code:
1 = Document printers (require distinction between malfunctions and consumable shortages).
0 = All other devices (uniformly categorized as "malfunction").

Example: Humanities Hall Retrieval Machine
Venue Code: 1
Device Sequence: 08 (8th device in the hall)
Type Code: 0 (non-printer device)
Full Device ID: 1080 → QR Code URL: http://SERVERIP?ID=1080

Then use a QR-code generator to encode the URL http://SERVERIP?ID=**** and print and affix the QR code to the device's visible surface.

For example:
Server address: 192.168.1.10
The QR code for the Humanities Pavilion retrieval machine is shown in Fig.3(a)

When readers scan this QR code using their mobile app's scanning function, their mobile browser will be directed to the URL: http://192.168.1.10?ID=1080. The report-receiving software is deployed on the server with IP address 192.168.1.10.

This makes it convenient for readers to scan the code and report problems when they encounter difficulties during use.

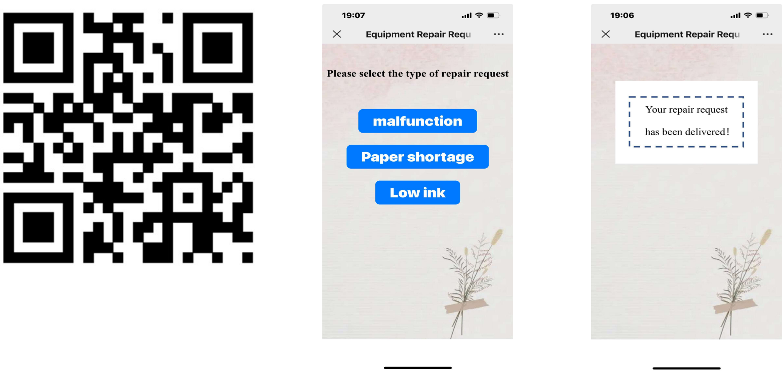
6.2. Website Construction

Create an ASP.NET Web application project, which includes the design of the front-end web pages and the back-end application programs.

6.2.1. Web Page Design

The web page contains two primary `<div>` elements. One `<div>` (with Type Code:0, the ID `else`) contains a `<p>` element with the prompt message "Your repair request has been sent!". As shown in Fig. 3(c). This message appears when the reported device is not a document printer. It serves as the system's post-alarm confirmation for non-printer devices, triggered by backend logic that checks the ID parameter in the URL (e.g., ID=1080 with type code 0).

The other `<div>` (with Type Code:1, the ID `printer`) evenly arranges three equally spaced `<button>` elements for document printers. These buttons are labeled as 'Malfunction', 'Out of Paper', and 'Out of Ink' respectively. After the reader scans the QR code, the `<div>` layer page with the ID `printer` is first displayed on the mobile phone, as shown in Fig. 3(b). After the reader selects a reporting type (e.g., "Out of Paper") and submits the alert to the monitoring system, this layer is hidden, and at the same time, the `<div>` layer page with the ID `else` as shown in Fig. 3(b) is displayed to inform the reader that the alert has been successfully sent.



(a) (b) (c)

Fig. 3. Repair request page for readers.

6.2.2. Back-end Program Design

The back-end program controls page visibility attributes and handles the processing and transmission of repair-reporting information. The network service will utilize UDP Sockets, with the recipient's IP address set to that of the monitoring center server and its listening port enabled.

During page initialization, the following logic executes:

1. Device ID Retrieval: Extract the DeviceId value from the query string using Request.QueryString["ID"].

2. Device Type Determination: Based on design specifications, parse the last character of DeviceId to identify the device category (e.g., 1 = printer, 0 = else).

3. UI Visibility Control:

Enforce mutually exclusive visibility for the <div> elements:

printer div: Visible (display: block) when the last character is 1

else div: Visible (display: block) when the last character is 0

4. Interactive Binding: For every <button> nested within the printer div, attach click event handlers referencing the pre-defined service program.

```
Data = System.Text.Encoding.Default.GetBytes( "ID" + "3");// Concatenate the device ID with the type code, and then convert the information to be sent into a binary stream.
IPEndPoint server = new IPEndPoint(severip, Port);// Define the Network Service Access Point. Here, "serverIP" represents the IP address of the monitoring center, and "Port" refers to the port that the other party is listening on.
UDP_Server.Send(Data, Data.Length, server);// Send statement
printer.Visible = false;// Hide UI(Fig.3(b))
else.Visible = true;// Unhide UI(Fig.3(c))
```

When readers click the Button, the device number and type code (where 3 indicates a malfunction, 4 represents an out-of-paper situation, and 5 denotes an out-of-ink condition) will be encapsulated and transmitted to the monitoring center as a binary stream. Concurrently, the <div> layer page with the ID "printer" will be hidden, and the <div> layer page with the ID "else" will be revealed.

For non-printer devices, when readers scan the QR code to report a malfunction, the web page immediately presents only the <div> layer page with the ID "else". At the same time, the device number and the type code "3" will be packaged and transmitted to the monitoring center as a binary stream. In this way, the acquisition and transmission of readers' repair-reporting information are achieved, ensuring that the monitoring center receives accurate and timely alerts of malfunction.

7. Centralized Monitoring and Information Processing

The monitoring center is a suite of application software installed on the server. Its key functions include continuously monitoring of the network status of all devices, checking printer consumables levels, receiving alerts from operational application software and reader repair requests. It then processes the received information further and forwards it to the designated workstations. Additionally, it visualizes the status of the monitored devices in the form. It is the central control hub of the entire system.

7.1. Equipment Relationship Mapping Table

In the library, there is a large number of devices, and their locations are constantly variable. To effectively monitor these devices better, it is necessary to establish an equipment relationship mapping table in the database. This table should include necessary attributes (fields) for each device, such as the device number, local IP address, device type, target IP address for sending alerts, and any required remarks, as illustrated in Table 1. Each device is assigned a unique device number (DeviceID). Once the monitoring center receives the alert, it queries this table to obtain the information that needs to be presented to the IP address of the destination workstation, so as to send the warning message to the specified terminal.

Table 1. Field Design for Equipment Table

Field Name	Description
DeviceID	The unique identification number of the device
Devicetype	Device type (1: Retrieval Machine; 2: Self-service Book Borrowing/Return Machine; 3: Self - service printer; ……)
DeviceIP	The IP address of the Self-service device
DestIP	The IP address(es) of the warning information receiving device(s), which can be multiple
note	Text description related to the Self-service device, such as “Retrieval machine No. 1 on the southwest side of the first floor”, used to provide clear warning text

7.2. Circular Monitoring and Information Processing

The monitoring center will continuously monitor and display the status of all electronic devices during the entire opening hours of the library, and enter a dormant state during the closing hours.

7.2.1. Circular Monitoring of Equipment Status

Implement a timer control with a setting of 30 minutes in its overflow service routine, check the online status of all electronic devices. For printing devices, the remaining amount of consumables also needs to be queried.

First, read the equipment table to obtain the IP addresses of all devices. Sequentially call the network status monitoring class functions described in Section 3. If the return result is “success”, it signifies that the device’s network connection is normal. In this case, set the indicator light of the corresponding device on the monitoring interface to green; otherwise, Conversely, if the result is not “success”, set it to red, display the word “Offline” beneath the light to denote that the device is either powered off or malfunctioning, and record the event.

Timed patrol monitoring is constrained by its fixed time intervals which means it cannot ensure immediate detection of equipment issues. Therefore, if readers encounter problems while using the equipment, they can trigger alerts by scanning the equipment repair QR code, and the monitoring center must be able to receive this information.

7.2.2. Receiving Alerts

Spawn a new thread to receive abnormal alerts from operational application software and repair requests triggered by readers scanning the QR code. This thread will listen on specific ports and log all received events.

7.2.3. Processing and Storage of Warning Information

Whether it is the monitored abnormal information or the received warning information, it needs to be processed, stored, parsed into intuitive text warnings, and distributed to the designated work machines.

Based on event types, equipment abnormalities are categorized into five primary classes: (1): Offline, (2) Software Failure: Operational application software abnormal, (3) User-Reported Fault: Reader submits a repair request via QR code scanning. (4) Out of paper, (5) Toner Depletion. The device's status indicator light color adjusts dynamically based on receiving information of equipment. A descriptive message is displayed in the monitoring interface. Each incident is recorded in the FaultRecords table with structured fields: the time of the event, device number and problem type (1–5). This enables querying by DeviceID, trending analysis of ProblemType frequencies, and generating maintenance reports.

When an abnormal state is detected in a device, the system queries the equipment table using the device's unique identifier (DeviceID). It then retrieves the corresponding textual description (notes) of the device and generates a detailed warning message. Based on the pre - defined the mapping relationship, this warning information is distributed to the designated target machine (the computer with the IP address of Destip). For example, the equipment fault information of the Humanities Hall is only sent to the computer terminal of the service desk of the Humanities Hall or the work machine of relevant technicians, and will not be sent to other locations such as the Science and Engineering Hall. This targeted distribution ensures that the relevant personnel are promptly informed about the equipment issues within their specific areas of responsibility.

7.2.4. Design of the Status Indicator Form

Design an equipment status indicator form. This enables technical librarians to effortlessly view the current working conditions of the facilities in the library, grasp the overall operational status of the library equipment, and quickly determine whether an issue is isolated to a single unit or affects a specific area. Based on the types of equipment deployed inside the library, set status indicator lights for each piece of equipment. Below each indicator light, clearly label its corresponding equipment location, as illustrated in Fig. 4. The color of the indicator lights corresponds to the status of the associated equipment. 0: Green light, signifies normal operation; 1–5: Red light, indicates malfunctions. When the red light illuminates, the specific fault type will be displayed in text directly beneath the indicator light for easy identification.

Fig. 4. Equipment status monitoring interface.

Due to the system's patrol inspection cycle and the lack of an active mechanism to determine whether the faults of Equipment No. 2 and No. 3 have been resolved, the monitoring platform cannot update the normal status indication of the equipment promptly. Therefore, the indicator light needs to be designed to double as a reset button. Once librarians have resolved the issues, they can click this button to clear the fault indication. The status indication form can assist librarians in grasping the recent operational status of the equipment, but it has limited real-time capabilities. Therefore, the exact time of a fault occurrence should be determined by the pop-up warning message received by the librarian.

For libraries with multiple buildings, the tabControl control can be employed to present the operational status of equipment in each building on separate pages. In addition, to enable librarians to query historical fault records in a targeted manner, a DataGridView control is incorporated into the form. Its data source of this control is bound to the fault record table, and relevant information can be retrieved through customized settings.

8. Warning Pop-up Module

To ensure that librarians can receive repair warnings promptly, it is necessary to install the alarm-receiving client software on their work machines. Once a message is received, a pop-up notification will be displayed immediately.

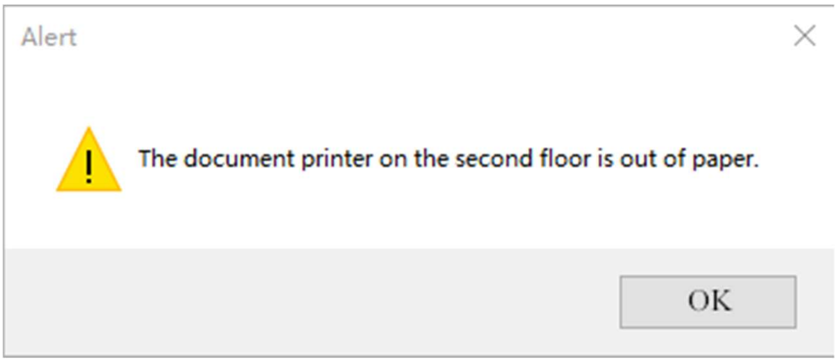


Fig. 5. Pop-up Alert.

The client software is configured to start automatically upon computer boot-up. During the initial loading phase, it hides the form and runs in the system background, immediately initiating listening on the pre-defined network port. When it detects incoming data, the software activates the service program, receives the binary stream data and converts it into a string format. Then, it employs the "MessageBox.Show(message, 'Alarm')" command to trigger a pop-up a window on the work machine to present the repair request information, as illustrated in Fig. 5. The alarm content is transmitted from the monitoring center. Upon viewing the prompt, librarians can promptly and effectively address the specific issues at hand.

9. Conclusion

The system leverages the combination of asp.net and C# programming languages to implement the aforementioned functional design uses the MySQL database to store device and alarm information. This system has been tested and operated in the library of Beijing Union University.

The library of Beijing Union University consist of 8 library buildings, which are distributed across 7 campuses. There are over a hundred pieces of equipment that operate without on-site attendants. In some library buildings, there are only 1 or 2 on-duty librarians, making it difficult to promptly detect and troubleshoot malfunctions. Therefore, this system was developed. The main control application software, website, and database are deployed on the server in the library server room, which are used for patrol monitoring, receiving alarms, and handling reader reports of faults via QR code scanning.

Currently, the equipment being actively monitored includes retrieval machines, digital displays, self-service book borrowing and returning machines, self-service printers, seat reservation machines, and locker cabinets, totaling 115 pieces of equipment. However, 122 facilities have been affixed with fault reporting QR codes, since some other devices do not require an internet connection, such as book disinfection cabinets. Fig. 4 shows the monitoring interface of the equipment in the Humanities Library of Beijing Union University.

The equipment that requires the installation of the operational application software's status monitoring function includes self-service book borrowing and returning machines, self-service printer, and seat reservation machines, totaling 30 pieces. Warning monitoring and receiving software is installed on the librarian workstations at the service desks of each library building. Most of the software installations are carried out through remote devices.

With the monitoring center server serving as the management hub, it integrates patrol monitoring, information receiving/parsing/sending services, a repair-request website, and status display functionalities. The application in the library of Beijing Union University has demonstrated that the system enables continuous monitoring of the operational status of unattended library service facilities during opening hours. It can promptly detect or diagnose issues, and send alarms to the designated work machines based on a predefined mapping relationship, prompting librarians to check and resolve malfunctions as soon as possible. However, when librarians are away their work-machine workstations for various reasons, they may fail to receive the alerts in a timely manner, leading to information oversight and potentially disrupting readers' normal use of library services. To address this limitation, in the subsequent development phase, an external SMS modem will be deployed at the monitoring center to implement an SMS reminder feature. This enhancement will ensure that even if librarians are away from their workstations, the alarm notifications can be directly sent to their mobile phones, promptly prompting them to solve issues for readers.

Analysis of fault incidents reveals that certain equipment experiences recurring issues in a regular pattern. For instance, a specific self-service book borrowing and returning machine fails to start automatically during opening hours, while other devices reboot or shut down spontaneously at specific time periods. Manual interventions by librarians only provide temporary fixes. In the future, plans are underway to integrate this system with an intelligent diagnostic system (AI System). Leveraging the AI system's capabilities, it will analyze the collected fault characteristics, diagnose the root causes accurately, and propose effective solutions. This proactive approach aims to prevent cyclical malfunctions, ensuring sustained and reliable operation of the equipment. This initiative is designed to optimize the usage experience for library patrons while elevating the intelligent service standards of the library to a new level.

Khan, A.U. comprehensively reviewed the research progress of emerging technologies in the field of smart libraries from 2013 to 2022. He emphasized the crucial significance of technological innovation and its application in enhancing the quality of reader services [4]. Gao, Yuan introduced the utilization of Internet of Things (IoT) data technology to achieve comprehensive interconnection among librarians, readers, space, document resources, and reading promotion activities. Moreover, this technology enables real-time monitoring and evaluation of the development process [13]. The content of this study is carried out from the perspective of improving readers' usage experience of library facilities. Yang, Chun-Jie introduced the monitoring of library energy consumption with the aim of achieving energy conservation [8]. In contrast, this study focuses on monitoring the status of self-service library equipment to reduce the failure rate of readers using library facilities and improve the effective utilization rate of the equipment. Both of these efforts, based on IoT technology, aim to optimize library management and enhance services. Khan, A.U.'s research indicates that the IoT-enhanced services in libraries need to be supported by robust management practices and effective utilization of technical resources [7]. Although this system can promptly detect equipment failures and issue warnings, an effective management mechanism is also required to ensure that librarians can quickly troubleshoot and eliminate the faults.

Conflict of Interest

The author declares no conflict of interest.

Author Contributions

Dongping Ma conducted the research with Jianping Du. Dongping Ma and Jianping Du carried out the solution design, while Dongping Ma and Qin Yang completed the program design and system implementation. Dongping Ma wrote the paper. All authors had approved the final version.

Acknowledgment

We would like to express our sincere gratitude to the Library of Beijing Union University for providing the research facilities. Special thanks also go to the library staff for their assistance in data collection and system testing. Finally, we appreciate the constructive feedback from the reviewers, which helped refine the manuscript.

References

[1] Zhou, W. (2024). IOT-enabled innovative environment with efficient routing for the digital library services to examine the behavior of users. *International Journal of Maritime Engineering, 1(1),* 663–674.

[2] Saha, S., & Roknuzzaman, M. (2024). Library practitioners' perceptions on the applications of IoT in university libraries of Bangladesh. *Library Management, 45(1/2)*, 141–156.

[3] Shahzad, K., Khan, S. A., & Iqbal, A. (2024). Factors influencing the adoption of Internet of Things (IoT) in university libraries: A systematic literature review (SLR). *The Electronic Library, 42(2),* 255–287.

[4] Khan, A. U., Ma, Z., Li, M., Zhi, L., Hu, W., & Yang, X. (2023). From traditional to emerging technologies in supporting smart libraries. A bibliometric and thematic approach from 2013 to 2022. *Library Hi Tech*.

[5] De Sarkar, T. (2022). Internet of Things (IOT) and library services. *Library Hi Tech News*, *39(9),* 18–22.

[6] Igbinovia, M. O., & Okuonghae, O. (2021). Internet of Things in contemporary academic libraries: Application and challenges. *Library Hi Tech News*, *38(5),* 1–4.

[7] Khan, A. U., Zhang, Z., Chohan, S. R., & Rafique, W. (2022). Factors fostering the success of IoT services in academic libraries: A study built to enhance the library performance. *Library Hi Tech, 40(6)*, 1976–1995.

[8] Yang, C. J., Kang, H. B., Zhang, L., & Zhang, R. Y. (2019). A design of smart library energy consumption monitoring and management system based on IoT. *Advances in Intelligent Systems and Computing, 891,* 217–224.

[9] Hakim, L., Kristanto, S. P., & Yusuf, D. (2023). Empowering Library attendance system with IoT-enabled student ID cards, *Proceedings of 2023 International Conference on Advanced Mechatronics, Intelligent Manufacture and Industrial Automation, ICAMIMIA 2023* (pp. 673–678).

[10] Nagowah, S. D., Sta, H. B., & Gobin-Rahimbux, B. A. (2021). An ontology for an IoT-enabled smart library in a university campus, *Proceedings of 2021 IEEE 23rd Int Conf on High Performance Computing & Communications* (pp. 1952–1957).

[11] Achary, R. (2021). Performance analysis of RFID and IoT based smart library management system, *Proceedings of 2021 2nd International Conference on Smart Electronics and Communication, ICOSEC* (pp. 68–72).

[12] Devi, P. D., Mirudhula, S., & Devi, A. (2021). Advanced library management system using IoT, *Proceedings of the 5th International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud), I-SMAC 2021* (pp. 150–154).

[13] Gao, Y. (2022). Construction of smart library service system based on internet of things data technology, *Proceedings of IEEE International Conference on Knowledge Engineering and Communication Systems,*

ICKES 2022 (pp. 1–6).

[14] Zhuang, Y. (2021). Optimization of the personalized service system of university library based on internet of things technology. *Wireless Communications and Mobile Computing*, *2021(1),* 1–10.

[15] Li, C. (2021). Construction of the collection retrieval system of intelligent library based on cloud computing, *Proceedings of 2021 5th International Conference on Intelligent Computing and Control Systems, ICICCS 2021* (pp. 471–474).