Design and Development of Industrial Failure Management System based on Mobile Terminal

Yiyang Yin[1,a], Haodong Wang[1,b]

[1]School of Information Technology Engineering, Tianjin University of Technology and Education, Tianjin, P. R. China.

[a]877716967@qq.com, [b]whdwhd123@outlook.com

Abstract

At present, computer technology and network technology in the rapid development, through the analysis of the traditional industrial failure declaration processing and feedback process, found that there is a slow response speed, communication is not smooth and other drawbacks, so this paper developed a mobile terminal based industrial failure management system. When the industrial equipment is abnormal, the management personnel can timely declare the equipment abnormality through the system. When the corresponding department maintenance personnel receive the declaration information, they immediately analyze and deal with the fault, so as to achieve timely maintenance. The system can realize portable industrial office, in the case of equipment failure in a timely manner to declare and deal with the fault, at the same time to facilitate the communication between departments, can effectively improve the work efficiency of industrial equipment maintenance, greatly ensure the reliable operation of industrial equipment, so as to ensure industrial economic benefits and social benefits.

Keywords

Industrial Failure Management; Mobile Terminal; Failure Declaration; Failure Handling.

1. Introduction

At present, our country belongs to the stage of industrial modernization promotion. The scale and control objects of various types of industrial automatic transportation and production lines are also diversified, and they use different core mechanisms, among which the constant link is the fault declaration and processing link. This is also an indispensable part of the whole production process. Therefore, if we turn our attention to the reliability of the industrial system and take precautions, we can check the industrial failure information in time and repair the failure problems in time, so as to reduce the industrial loss to the greatest extent [1-2].

In order to understand the fault information in real time and reduce the cost of labor, a set of industrial failure management system based on mobile terminal is designed and developed. The system uses Android system architecture and Java language under the Eclipse development environment, which aim to save manpower and increase the response speed of fault maintenance and facilitate the work of maintenance department members through mobile office mode. Finally achieve the goal of improving the work efficiency of the enterprise [3-4].

The system adds modules of user management, industrial fault declaration, industrial fault handling feedback, historical industrial fault statistics and message. The members of each department declare the industrial faults of their own department through the mobile phone front-end app. After receiving the fault information through the database, the background administrator classifies and notifies the corresponding maintenance personnel, so that the maintenance personnel can put into the maintenance work as soon as possible, so as to realize the information management of different nature and types of industrial faults [5]. The development and operation of this system will replace the traditional industrial fault reporting mode, such as telephone, table form, etc., avoiding the time lag and delay of layer upon layer reporting, especially suitable for urgent fault reporting and processing [6]. The system is convenient for the field staff to grasp the relevant industrial fault processing

dynamics through the mobile phone client in time, carry out fault scheduling and distribution, and give relevant feedback. In addition, this system has the advantages of simple and beautiful operation interface, simple software operation and easy to be accepted by the market [7]. It can effectively utilize modern information technology to standardize and optimize the enterprise's existing industrial fault processing workflow, which greatly improve the working efficiency of industrial failure management and maintenance.

2. Software framework

This system adopts the popular HTML5 responsive layout, JAVA language, Eclipse and other writing tools, combined with the Android open source system technology for design. The Eclipse development language is simple, and the interface design can be done directly elsewhere, saved as an XML file, which can be opened directly in Eclipse. Therefore, you can directly build the visual view you need in the corresponding XML layout of your Activity or Fragment, and it is also easy to modify [8-9].

The system adopts the design idea of object-oriented programming mode. The design mode adopts the classic MVC mode, which can realize the separation of data model and view implementation code, and enhance the flexibility of interconnection between data and view. The presence of the controller ensure the timeliness between the data and the view, so that when the data changes, the view can be updated synchronously. Therefore, MVC has many advantages such as good user experience, strong operability, high degree of freedom of operation, and more convenient to meet the needs [10].

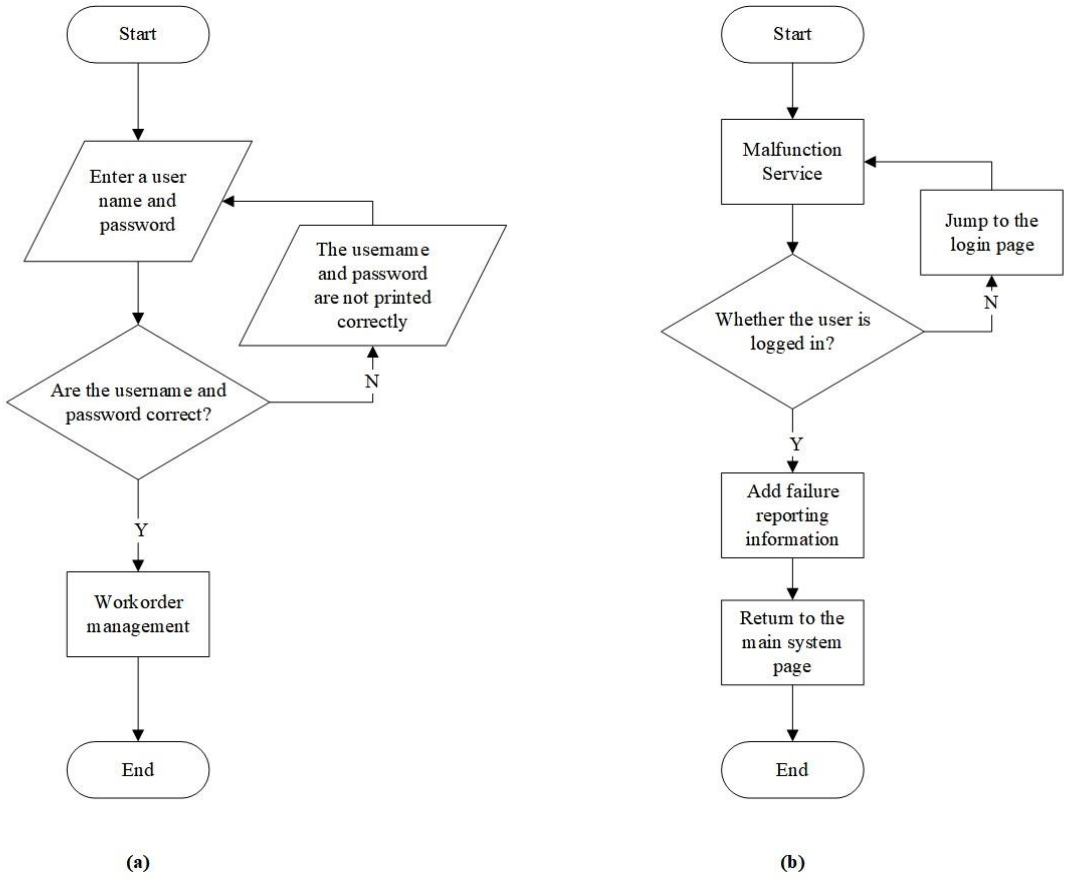

Fig. 1 (a) Login flow chart. (b) Failure declaration flow chart

The system uses SQLite database, which is a lightweight, cross-platform relational database. SQLite is different from the database software in C/S mode. SQLite database is an in-process database engine, so there is no client and server of the database [11]. Using SQLite, you only need to bring one of its

dynamic libraries to enjoy all its functions, and the size of the dynamic library is relatively small [12]. In addition to the mainstream operating systems, SQLite also supports a number of lesser operating systems [13]. Another feature of SQLite is green. Its core engine itself does not rely on third-party software, and it does not need "installation" to use it, so it can be deployed without a lot of hassle [14].

Flow chart of software design: The login flow chart is shown in Fig. 1(a). The flow chart of failure declaration is shown in Fig. 1(b).

3. Results

The results of this system are divided into the results of the front-end page of Android software and the results of the back-end database of the software. Users are logged in by different departments, and the declaration and feedback between different departments cannot be viewed each other. The most advanced users can see all user information.

3.1 Web Front End Page

As for user information, in the whole process of input, it should be considered whether the information can be saved immediately. If it is saved, whether the accurate conclusion can be fed back to the user when searching for information from the database. In the case of failure, the system should be required to return to the error mark. That is to say to judge whether the information retrieved from the user is within the entire system, if it is, it can be entered, and if it is not, an error will be displayed. The user login function interface is shown in Fig .2(a).

Users will enter the system after entering the correct account and password. Users can view the detailed information of their own work order, which can be divided into four states: my work order, work order to be reviewed, work order to be executed, and work order in execution. The detailed implementation interface of the work order is shown in Fig. 2(b).

(a) (b)

Fig. 2 (a) User login function interface. (b) Function realization interface of work order management

According to the current situation, the user can make a detailed plan for the equipment and do regular maintenance and repair. The user plan interface is shown in Fig. 3(a).

Users can make a detailed record of their current equipment, including: equipment list, failure repair, add equipment, repair list four functions. The interface for device management is shown in Fig. 3(b).

In the "My" page, the detailed information of the user is recorded, where the user can change the password, upload the avatar, set permissions and other operations. The user information interface is shown in Fig. 3(c).

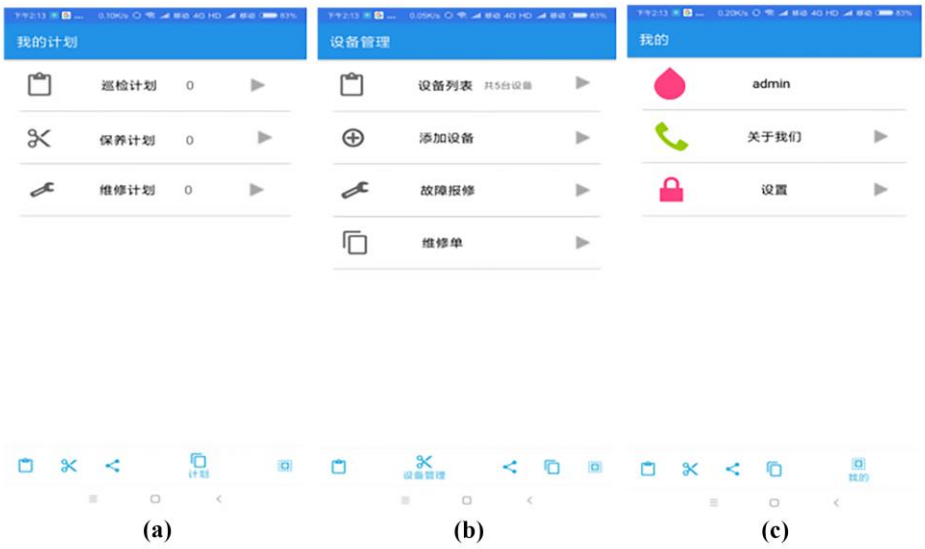

Fig. 3 (a) User Information Interface. (b) Device management interface. (c) The user plan interface.

3.2 Background Database

When the user enters the user name and password, the system will go to the database for comparison. If it succeeds, the user will enter the system; if it fails, the user will be reminded to log in again. When the user logs in, he/she will choose the department he/she is in. Users from different departments enter different pages. The interaction between departments can only be operated by the administrator who has this authority. The administrator login interface is shown in Fig. 4.

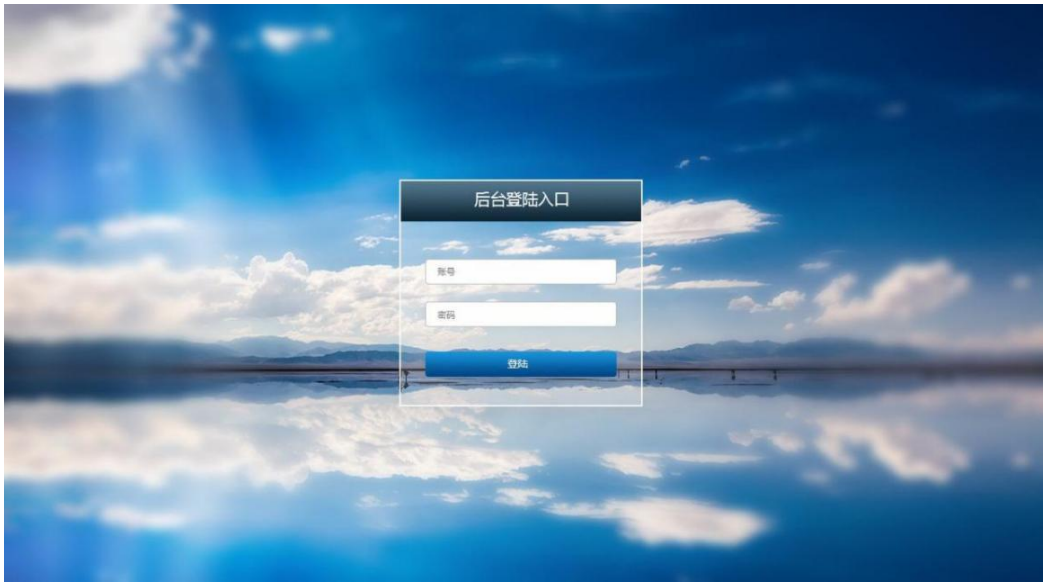

Fig. 4 Administrator login interface

The administrator can check and manage the user's declaration, arrange maintenance personnel to deal with it, and at the same time, he can also declare the failure by himself. The interface of the background management system is shown in Fig. 5.

Fig. 5 Interface of background management system

The administrator has the authority to make a detailed planning of the equipment of his own department, including what equipment is specific, which person is responsible for it, how long the time cycle is, and how long the current state is set in detail. The interface for the administrator to plan the device is shown in Fig. 6.

Fig. 6 Administrator planning device interface

The administrator mainly manages the status of the equipment, including the following functions: equipment problem tracking, equipment maintenance, equipment overhaul, equipment lubrication, spare parts, and warehouse out registration. According to different permissions, users at all levels can see different pages. The administrator management device interface is shown in Fig. 7.

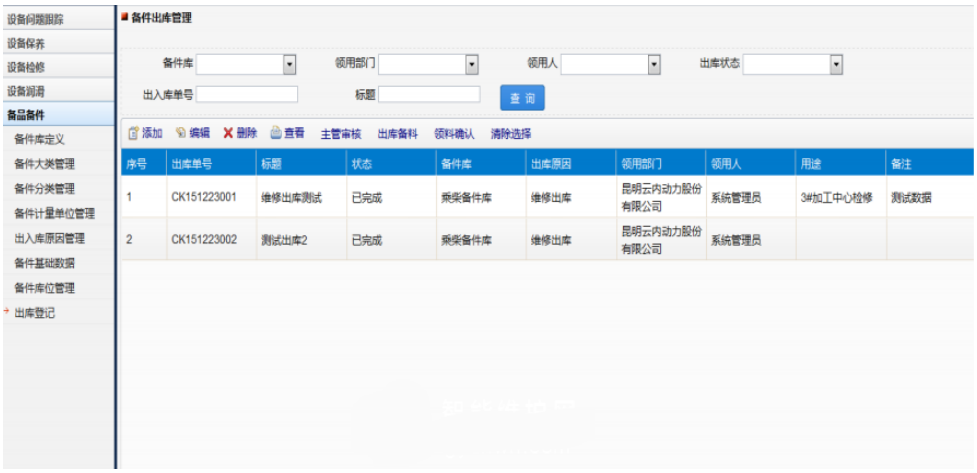

Fig.7 The administrator manages the device interface.

4. Conclusion

Different from traditional industrial failure report model, this work designed and developed industrial fault management system based on mobile terminal APP and background database. The system is convenient for on-site maintenance staff to grasp the relevant industrial failure processing dynamic through the mobile phone client in time, carry out fault scheduling and distribution, and give relevant feedback, so as to carry out timely scheduling and maintenance, effectively avoid the time lag and delay of layer by layer reporting caused by traditional telephone and form forms, especially suitable for the declaration and processing of urgent faults. In addition, it also has the function of industrial failure classification and statistics, which is convenient to view the historical faults and processing methods, and has reference value for the solutions of similar faults. The Android mobile app replaces the traditional fault declaration, processing, feedback and management, which is conducive to portable management and mobile office. It is an important application of industrial informatization and has the value of industrial promotion.

References

[1] Jia H, Yang B, Song C, et al. 41.4: Real time environment monitor with Augmented Reality device based on android system[C]// SID Symposium Digest of Technical Papers. 2018:454-456.

[2] Antonio J. Guillén et al. A framework for effective management of condition based maintenance programs in the context of industrial development of E-Maintenance strategies[J]. Computers in Industry, 2016, 82: 170-185.

[3] Jp A, Nan Z B, Quan X C, et al. A new intelligent and data-driven product quality control system of industrial valve manufacturing process in CPS[J]. Computer Communications, 2021.

[4] Wang, Jianan, Zhang, et al. An Improved Android Based Industrial Monitoring and Locating System [C].

[5] Fan L, Zhang H, Chen Q, et al. The Fault Record Analysis Display Technology Based on Mobile Terminal [J]. 2020.

[6] Yuan Y, Liu P, Liu Y. The application analysis of fault diagnosis with artificial Intelligent for industrial equipment [C]// 2020 IEEE 2nd International Conference on Civil Aviation Safety and Information Technology (ICCASIT). IEEE, 2020.

[7] J.F.DiMarzio. Android a programmer's Guide[M].McGraw.Hill,2012

[8] Rogers R, Lombardo J, Mednieks Z, et al. Android Application Development -Programming with the Google SDK. [J]. Wiley & Sons, 2009.

[9] Darwin I F. Android Cookbook: Problems and Solutions for Android Developers. 2017.

[10] Sherwin John Calleja Tragura. Spring MVC Blueprints [M]. Packt Publishing, 2016.

[11] E Hellman. ANDROID PROGRAMMING: PUSHING THE LIMITS[J]. 2013.

[12] Griffiths D, Griffiths D. Head first Android development: a brain-friendly guide. 2017.

[13] Oliveira W, Oliveira R, Castor F. A Study on the Energy Consumption of Android App Development Approaches [C]// 2017 IEEE/ ACM 14th International Conference on Mining Software Repositories (MSR). ACM, 2017.

[14] Lin J W, Salehnamadi N, Malek S. Test automation in open-source Android apps: a large-scale empirical study [C]// ASE' 20: 35th IEEE/ ACM International Conference on Automated Software Engineering. ACM, 2020.