Research and Implementation of Intelligent ERP Platform for SMEs Based on Cloud Computing

Yong Wen*

Department of finance, GuangDong University of Science & Technology, Dongguan, Guangdong, China

*534235485@qq.com

Abstract. According to the characteristics and business needs of small and medium-sized business management, the paper analyzes and designs the functions and performance of ERP platform, applies Internet of Things and artificial intelligence technology, and builds a dynamic mobile intelligent SME ERP platform through business database, rule database and knowledge database. The platform adopts three-tier B/S architecture, uses MySql as the back-end database, uses Java to encode, implements financial management, supply chain management, manufacturing, human resource management and other functional modules, and finally carries out platform testing. The implementation results show that the intelligent SME ERP greatly reduces the workload of business personnel and improves the efficiency of real-time dynamic management of enterprises.

1. Introduction
The ERP system based on the cloud computing platform is that the system and the database are deployed on the cloud server-side, and the user accesses the cloud server through the peripheral terminal devices such as a smart phone, a notebook computer, a tablet computer, and a desktop computer to access the cloud server to obtain the ERP online service[1].

It has the advantages of on-demand use, low cost, security and stability, easy upgrade and maintenance, etc. It has become the best choice for SMEs to achieve digital management. Emerging technologies such as big data, cloud computing, and artificial intelligence have given life to the traditional ERP system[2]. In March 2016, Deloitte, one of the world's four largest firms, took the lead in integrating artificial intelligence into the financial process, launched the Xiaoqin financial robot, and put it into operation at Sinochem International Financial Sharing Center. In May, PricewaterhouseCoopers also developed financial robots, which were developed by Xiaoqin people in the financial sector to other areas such as supply chain and human resources. In June, Ernst & Young and KPMG also launched their own financial robots. In the same year of June, Ernst & Young and KPMG also launched their own financial robots. The above financial robots are designed and developed for large enterprises and financial institutions. There is a lack of SMEs' intelligent ERP platform in the market. SMEs have simple business, clear processes, and flat management, making it easier to build an intelligent ERP platform[3].

2. Platform development process
Firstly, according to the user's needs, build the platform logic model, determine the functions that the platform, and secondly, carry out the overall design, as shown in Figure 1, the functional modules and
their interrelationships of the design platform, and then carry out detailed design, define the database model and structure. Again, write the program code, implement the interface and function of the platform, and finally, conduct the platform test and acceptance[4]. Due to the limited space, the following only describes the process of platform implementation and platform testing.

3. Platform implementation

platform implementation is the process of putting the new system into implementation based on the system design[5]. The ERP platform mainly includes six modules, such as system management subsystem, financial management subsystem, human resource management subsystem, supply chain management subsystem, production manufacturing subsystem, and daily office subsystem. Because of the limited space, the following is a brief introduction to the implementation process of several sub-modules under the financial management module. And the platform is tested from three aspects: platform function, performance, and user interface.

3.1. Platform development environment

3.1.1. platform development tool selection. Java with JDK1.6 version or higher is used as the foreground development tool, and the system code is written in accordance with the unified coding mode. UML modeling tools are available: Microsoft Office Visio and Rational Rose Enterprise Edition.

3.1.2. Database selection. At present, the electronic engineering of large and medium-sized enterprises adopts such large and medium-sized relational databases with powerful functions, stable operation and good after-sales service such as Oracle, IBM DB2, Microsoft SQL Server and Mysql. However, considering that MySql is more economical than Oracle, as well as engineering scale and management level, it is determined to use MySql as the back-end database.

3.2. Software architecture

The platform adopts Java development, adopts struts + spring + jstl + jquery + Sql framework and Server2005 database, adopts multi-layer distributed, the architecture is easy to maintain and secondary development, and the system module adopts the idea of plug-in component technology for service-oriented (SOA). Flexible and scalable.

3.3. System management module implementation

System management module is system data management center. After logging in to the system management, you can set up jobs, set up organizational models, user management, enter data dictionary, and download center maintenance.
3.4. Implementation of sub-module of engineering project
The project sub-module is a sub-module under the general ledger module. Its main function is to
complete the entry of cost and expense documents; receivables, payables input or automatic generation;
addition, modification and deletion of bank deposits and cash-storage business; shared document
maintenance, project cost budget, etc.

3.5. Implementation of report query sub-module
The report query sub-module is a sub-module under the general ledger module. Its main functions
include: request summary inquiry, project overview query, current summary inquiry, summary
analysis, customer receivable inquiry, business status inquiry, cost expense inquiry and Analysis,
detailed account statement inquiry, bank cash journal inquiry, balance sheet inquiry, profit statement
inquiry, cash flow statement inquiry, etc.

3.6. Part of the source code display
```java
package com.dreamlv.persiste.manager.dao;
import com.dreamlv.exp.RtExp;
import com.dreamlv.iface.sql.IBatch;
import com.dreamlv.persiste.manager.DBMgr;
import com.dreamlv.uti.StrUti;
public class Update {
    private static Update update = new Update();
    private Update() {
        super();
    }
    public static Update instance() {
        return update;
    }
    public boolean update4SureRows(Connection con, String sql, int updateRows, Object... params)
        throws Exception {
        if (StrUti.isNull(sql)) {
            RtExp.alert("Update error: updated related sql statement is empty");
        }
        PreparedStatement ptm = null;
        boolean ok = true;
        try {
            ptm = con.prepareStatement(sql);
            int param_num = params.length;
            for (int i = 0; i < param_num; i++) {
                ptm.setObject(i + 1, params[i]);
            }
            int rows = ptm.executeUpdate();
            ok = (rows == updateRows ? true : false);
        } catch (Exception e) {
            throw e;
        } finally {
            DBMgr.close(null, null, ptm, null);
        }
        return ok;
    }
    public boolean update4SureRows(String sql, String db, int updateRows, Object... params) {
        if (StrUti.isNull(sql)) {
            RtExp.alert("Update error: updated related sql statement is empty");
        }
        Connection con = null;
        boolean ok = true;
        try {
            con = DBMgr.getTranConnection(db);
            ok = update4SureRows(con, sql, updateRows, params);
            if (ok) {
                con.commit();
            } else {
                con.rollback();
            }
        }
    }
```
```java
} catch (Exception e) {
    try {
        con.rollback();
    } catch (SQLException e1) {
    }
    DBMgr.logAndAlert(this, "Update data error: ", e, "Update data error:<br/>");
} finally {
    DBMgr.close(con);
}
return ok;
}
public boolean update(String sql, String db, Object... params) {
    if (StrUti.isNull(sql)) {
        RtExp.alert("Update error: updated related sql statement is empty");
    }
    Connection con = null;
    try {
        con = DBMgr.getTranConnection(db);
        update(con, sql, params);
        con.commit();
    } catch (Exception e) {
        try {
            con.rollback();
        } catch (SQLException e1) {
        }
        DBMgr.logAndAlert(this, "Update data error: ", e, "Update data error:<br/>");
    } finally {
        DBMgr.close(con);
    }
    return true;
}
public boolean batch(Connection con, String sql, int batch_size, IBatch iBatch) throws Exception {
    if (StrUti.isNull(sql)) {
        RtExp.alert("Batch error: batch related sql statement is empty");
    }
    if (batch_size < 1) {
        RtExp.alert("Batch error: batches less than 1");
    }
    PreparedStatement ptm = null;
    try {
        ptm = con.prepareStatement(sql);
        for (int i = 0; i < batch_size; i++) {
            if (iBatch.oneBatch(ptm, i)) {
                ptm.addBatch();
            }
        }
        ptm.executeBatch();
    } catch (Exception e) {
        throw e;
    } finally {
        DBMgr.close(ptm);
    }
    return true;
}
public boolean batch4OneRow(Connection con, String sql, int batch_size, IBatch iBatch) throws Exception {
    if (StrUti.isNull(sql)) {
        RtExp.alert("Batch error: batch related sql statement is empty");
    }
    if (batch_size < 1) {
        RtExp.alert("Batch error: batches less than 1");
    }
    PreparedStatement ptm = null;
    boolean ok = true;
    int updateRows = 0;
    try {
        ptm = con.prepareStatement(sql);
        for (int i = 0; i < batch_size; i++) {
```
```java
if (iBatch.oneBatch(ptm, i)) {
    updateRows++;
    ptm.addBatch();
}
ok = DBMgr.isBatchSuccess(ptm, updateRows);
} catch (Exception e) {
    throw e;
} finally {
    DBMgr.close(ptm);
}
return ok;
```

```java
public <T> List<T> listAddParam(Connection con, String sql, List<T> listAddParam, IQuery<T> iQuery, Object... params)
throws Exception {
    if (StrUtF.isNull(sql)) {
        RtExp.alert("Error: query related sql statement is empty");
    }
    if (listAddParam == null || listAddParam.isEmpty()) {
        return null;
    }
    PreparedStatement ptm = null;
    ResultSet rs = null;
    try {
        ptm = con.prepareStatement(sql);
        int param_num = params.length;
        for (int i = 0; i < param_num; i++) {
            ptm.setObject(i + 1, params[i]);
        }
        rs = ptm.executeQuery();
        int i = 0;
        T row = null;
        while (rs.next()) {
            row = iQuery.row(rs, i);
            if (row != null) {
                listAddParam.add(row);
            }
            i++;
        }
    } catch (Exception e) {
        throw e;
    } finally {
        DBMgr.close(rs, null, ptm, null);
    }
    return listAddParam;
}
```

4. Platform test

4.1. Function test
Introduce the function module test with the general ledger and report module as an example.

The main content of the test: Whether the module can generate accounting vouchers from the original vouchers, register the books, correctly prepare financial statements, etc., whether the module can perform anti-audit and anti-bookkeeping on the voucher, whether each report, detailed account and accounting vouchers can implement an integrated query. Test results: The report module and the general ledger module are seamlessly connected, and each report and detailed account can be linked to the corresponding accounting documents. The functions of the system conform to the business specifications and requirements, and each function runs normally.

4.2. User interface test
The main content of the test: whether the user interface is beautiful, whether the image is clear,
whether the text description is appropriate, whether it meets the customer's needs, including: system main menu test, function module test, prompt information test, help window test. Test results: The user interface is user-friendly and can meet customer needs. The interface text and prompt information are correct and the semantics are clear.

4.3. Database running test
The main content of the test: increase the data table to the maximum load to see if it can run correctly. After deleting all the data tables, whether can add new data to the empty table or not. After the database data items are added, deleted, modified, etc., the system whether can run normally or not.

4.4. Safety and reliability test
Security reliability testing includes software fault tolerance testing and security confidentiality testing. Fault tolerance test results: The system can make a correct judgment on the wrong operation and prompt the user to return to the normal state. Security and confidentiality test results: User authentication function is rigorous, and authority allocation meets the requirements of internal control principles.

5. Summary and outlook
According to the current cloud computing technology and the needs of SME business management, the article designs and implements the intelligent SME ERP platform. Due to the limitation of time and business level, the platform still has many shortcomings. For example, the Internet of Things technology is not well integrated into the ERP platform; ERP platform, CRM and e-commerce are not well integrated; the level of intelligent analysis and decision-making of ERP platform is limited. In the future, we will absorb talents from CRM and e-commerce, enhance the research breadth and depth of the team, and improve the above three major deficiencies of intelligent ERP.

Acknowledgement
[Fund Project] Characteristic Innovation Project of “Innovation and Strong School Project” of Higher Education in 2017 in Guangdong Province (Project No.: 2017WTSCX125).

References
[1] Xiqiang M 2018 China. D. Research on risk evaluation of SaaS ERP service provider selection in small and medium sized enterprises (Xi’an:Library of Xi’an University of Technology)
[2] Jian W.2013 China. D. Intelligent three-dimensional dynamic Accounting Information Platform Research (Qingdao: Ocean University of China Libraries)
[3] Xu H.2013 China. D. Research and Design of Financial Management Function of ERP System Based on .NET(Harbin: Jilin University Library)
[4] Annamalai.C and Ramayah.T 2011 Enterprise resource planning(ERP) benefits survey of Indian manufacturing firms: An empirical analysis of SAP versus Oracle package.vol 17(3) (London: Business Process Mangemet Jounral)pp 495-509
[5] Huayang M, Mingyun F 2016 Research on SaaS Model under Management Accounting Informationization Finance and accounting monthly vol 4 (Wuhan: Hubei ) pp 102-104