

A PROOF OF CONCEPT APPLICATION FRAMEWORK FOR MOBILE ERP SYSTEMS

D. Seneviratne, I. Sugathsiri, S. Udeshinie, R. Ragel and S. Radhakrishnan

Department of Computer Engineering, University of Peradeniya.

Introduction

Enterprise Resource Planning (ERP) is a way to integrate the data and processes of an organization into one single system and for software to be an ERP system; it must provide an organization with functionality for more than one system (ERP, 2009). Thousands of lines of code are used in real world ERP applications and developing such applications are time consuming and complex activity. A *framework* is an abstraction in which the common code providing generic functionality can be selectively overridden or specialized by user code providing specific functionality (SF, 2009).

The concept of framework can be used to reuse the common patterns of ERP business operations. Major advantages of developing frameworks are enabling the reusability, simplifying the software design, reducing the development and maintenance cost and reducing the workload of application developers. This paper presents the process of developing a framework for ERP mobile applications. The developed framework is divided into three: (1) the mobile framework (abstraction of common functionalities for developing the mobile application), (2) the server framework (abstraction of common functionalities for developing windows server application that communicate with the mobile application), and (3) the common framework (abstraction of common functionalities for developing both server and mobile applications).

Related Work

RFgen's mobile and wireless framework (RFgen, 2009) and *iNotify* mobile application framework (iNotify, 2009) are

two of existing frameworks. *RFgen* framework is proven to shorten development time up to 80%. *iNotify* framework has a unique feature of working on all devices on such as *Nokia*, *Motorola* mobile phones, *Windows Mobile* smart phones, PDA etc. It also works on both *GSM* and *CDMA* networks. Although both *RFgen* and *iNotify* have some good features and they are aimed at developers, they are not targeted toward the ERP domain. We argue that a framework developed targeting a specific domain such as ERP would render lot more benefits (such as additional amount of common code and therefore even lesser development time) compared to common framework. Therefore, this work is to show such benefits in application specific framework development.

Architecture of the Framework

Figure 1 shows the basic architecture of the mobile application framework. The functionalities which should be implemented in the mobile applications such as database persistence, security, etc are taken in to the framework in a reusable manner and the mobile applications can be developed top of the framework.

To use the mobile devices in an ERP system, it is necessary to connect the device with the enterprise network. The mobile device is connected to the network so that data exchange with the backend server can take place. Among the several techniques that can be used for this, web services were selected by us considering the simplicity of programming and the flexibility. Therefore, as mentioned in the introduction, two more set of abstractions were established in this process: one with

common code of the backend and the other with the common code of the mobile application and the backend. We called these two the server and the common frameworks respectively. More details on these frameworks are discussed under implementation.

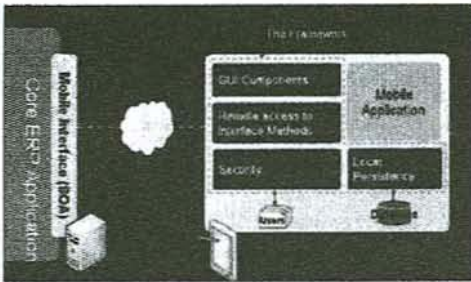


Figure 1. Architecture of the Framework

Implementation

The first need of developing a framework was identifying the common functionalities of ERP mobile applications. For this, we have to develop two prototype ERP applications and they are: “work order” and “customer order” - simplified versions of real ERP applications.

Both these applications had database operations (create, retrieve, update and delete) and it was recognized that these operations can be reused. Therefore database operations were taken into the mobile framework. Here, each operation was implemented to cater different scenarios considering the necessity of the application developer.

For instance, if we consider a *save command*, four scenarios have been implemented in the mobile application framework letting the developer to select the scenario according to the requirement as illustrated in Table 1. Parameter ‘dirty’ indicates whether this is inserting or updating. Apart from the database operations, some extended data types from the data types which are provided by .Net were also included in the framework.

These data types have some additional properties than normal data types.

Table 1. Scenarios of “Save” method

Method name	Usage
SaveData(bool dirty)	Can be used to save or update all the fields
SaveData(List<string> fieldNames, List<SqlDbType> typeList, bool dirty)	Can be used to insert or update data to selected datafields or to entire table. This returns ‘sqlceCommand’ and developer can add values to data fields.
SaveData(List<string> fieldNames, List<object> values, bool dirty)	Used by giving all the names of data fields that should be updated to the database and their values.
saveData(DataSet dataset, bool dirty)	Can be used to insert or update data to selected data fields of the data table and data is given as a data table.

To synchronize the data between mobile device and server, web services were used and data was sent as *Data Transfer Objects* (DTO). Database methods in the mobile framework are used to implement this data synchronization as well.

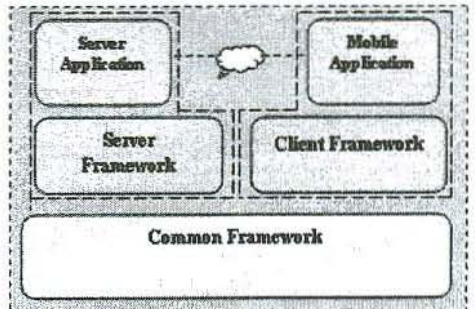


Figure 2. Design and Usage of Frameworks

As illustrated in Figure 2 *common framework* can be used to develop both mobile and server applications. For example, when a developer needs to encrypt or decrypt a text when developing mobile or server application, he can use the common framework method directly without writing his own methods. Common framework was developed by taking all the common parts of the server

and the mobile framework together. Extended data types and the data encryption and decryption that are used for authenticating mobile and windows users were included to this framework. Other common functionalities for developing mobile and server applications can be accessed from the server and the mobile framework respectively. Therefore, any mobile or server application can be developed on top of the relevant framework and the common framework achieving a good level of code reduction and reducing the complexity of application development.

Evaluation of the Framework

The code reuse (and therefore reduction) that is achieved due to the framework is illustrated with an example in Figure 3.

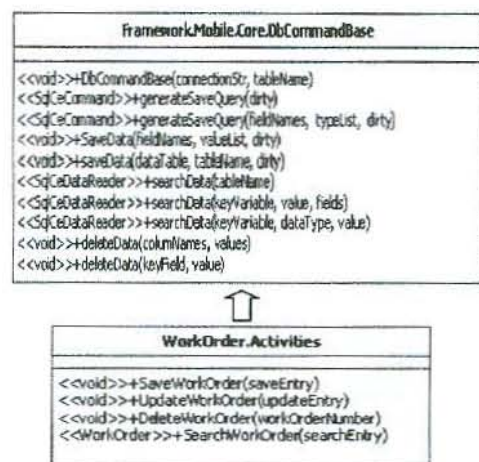


Figure 3. Mobile Framework Methods

In Figure 3, WorkOrder.Activities class belongs to a mobile application - work order - developed on top of the mobile framework. All the methods as illustrated for this class perform database activities. Since the application is developed on top of the framework, developers can use the relevant database functionalities which have already been written and tested without worrying about writing SQL queries for the large database. In this class, we could achieve code reduction of more than 60% when developing it using the framework. Similarly in non database

methods (such as data encryption and decryption, and synchronization) the framework helped to achieved about 40% code reduction.

Future Work and Conclusion

The three frameworks (Common, Server, and Client/Mobile) which were developed during this project are helpful in the development process of ERP applications. Although there are considerable amount of functionalities implemented in the three frameworks, some more features (such as implementing GUI components) can be added.

Frameworks take the tedium out of writing all the program code for an application from scratch and they are structured as a class library. Each class library has its way of doing things, and although the purpose of a framework is to eliminate a certain amount of programming drudgery, programmers must first learn the structure and peculiarities of the framework in order to use it. It can be concluded that a framework may take a lot of time and effort to develop, but it saves time during the application development.

References

- ERP (2009). What is ERP?, techFAQ. Available at <http://www.techfaq.com/erp.shtml>, accessed on 10-08-2009.
- Wigley (2007), Andy Wigley, Microsoft Mobile Development Handbook, Microsoft Press, 2007.
- iNotify (2009). Mobile Application framework, Available at <http://www.mobientech.com/pressreleases/Press-release-13%20JUNE.pdf>, accessed on 10-08-2009.
- RFgen (2009). Mobile and Wireless Framework. Available at http://www.rfgen.com/app_overview.asp, accessed on 10-08-2009.
- SF (2009), Software framework. Available at http://en.wikipedia.org/wiki/Software_framework, accessed on 10-08-2009.