Students' Portal Architecture based on SOA

Rami A. Y. Raba
Information System Department,
Limkokwing University
Cyberjaya, Malaysia
ramiraba@yahoo.com

Mohammed M. I. Awad
Software Engineering Department,
University of Palestine
ALzahra, Gaza, Palestine
m.awad@up.edu.ps

Abstract—The students' portal represents a major part among university information systems. Lack of components distribution is a gap that leads to many problems in students' portals in most of universities worldwide, which is due to tightly-coupled components in the current portal framework. This paper reviews the common features of the general portal structure. A comparison between the traditional students' portal framework and the new students' portal framework is presented. In addition, the paper discusses how to distribute the students' portal components. To achieve the components distribution of the portal, Service Oriented Architecture [1] is adopted to address the missing distribution feature. The research contributes towards the higher education field worldwide by providing a components distributable framework that could be followed as a base for building university portals with distributed components.

Keywords-portal; tightly-coupled; SOA; loosly-coupled; distribution.

I. INTRODUCTION

A portal is a website that provides a gateway to other websites. It gathers links and data in one place and then refers visitors to other places for more information. For example, Yahoo and America Online [2] are popular portal websites. Developing such a portal offers several benefits for businesses and organizations [3]. Therefore, a portal as a collection of resources, is not only used to save the content information and user customized information [4].

A students' portal is an online gateway where students can log into a university website to access important program information. Student portals contain information about courses offered, transcripts, e-mail programs, timetables, exam schedules and department contact numbers. They may also offer links to useful web resources such as research tools and online journals. Online students' portal makes the environment easier for staff and students to access important information from anywhere, at any time. Portals are commonly used in colleges and universities where prompt information and necessary updates must be readily available to a large number of students.

There is a gap in the current student portals framework. This gap is related to the absence of component distribution. Tightly-coupled software architecture results in low reusability and difficult maintainability of software [5]. In addition to software problems, a tightly-coupled system where multiple systems share a workload, the entire system usually would need to be powered down to fix a major hardware problem [6]. Among the independent systems, data sharing and integration has become a difficult task [7].

According to [8], current collaborative portals have been developed at one time for specific science domains. Therefore, these portals cannot be easily extended beyond their initial features or reused by other science domains. Therefore, tightly-coupled components cause problems in term of cost, maintenance, enhancement, and reusability of components [5]. On the other hand, an extensible students' portal framework with loosely-coupled components can resolve this complication [9]. This leads to the necessity that the architecture of the software framework needs to consider the component coupling factor and how much loosely the components should be coupled.

The overall objective of this research is to define a new framework for distributed students' portal components. This framework is an enhancement to the current students' portal architectures in term of components distribution.

II. UNIVERSITY SYSTEM ARCHITECTURE

A. General University E-System Design

Figure 1 shows the use-case diagram of university system which is e-university website that is managed by IT-unit available in a university. The user uses the university system to view or update his/her transactions using registration and admission, students' affairs, financial affairs, graduate studies, library and/or any extended components. All these components are represented by use-case. The use-case use university actor that is shown in figure 1. This actor provides services and information for student and staff according to their profile and specific needs.

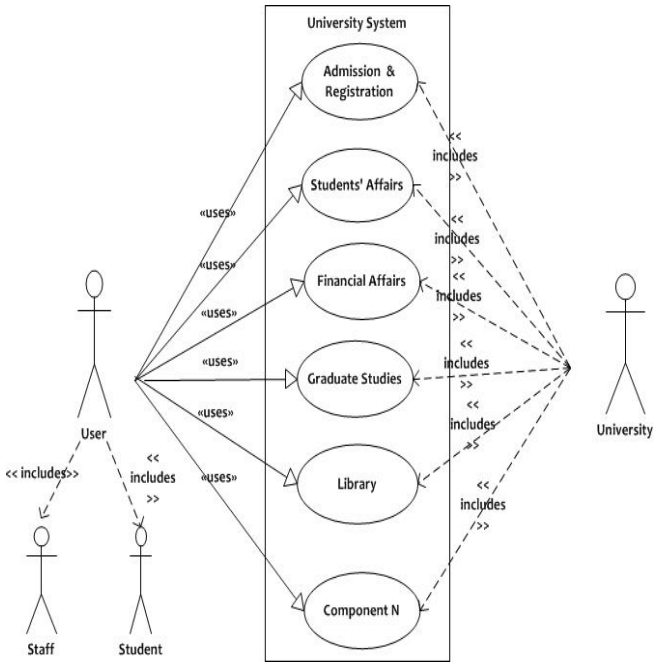

Fig.1. Use-case diagram of the General university e-system

B. General University E-System Flow of Actions

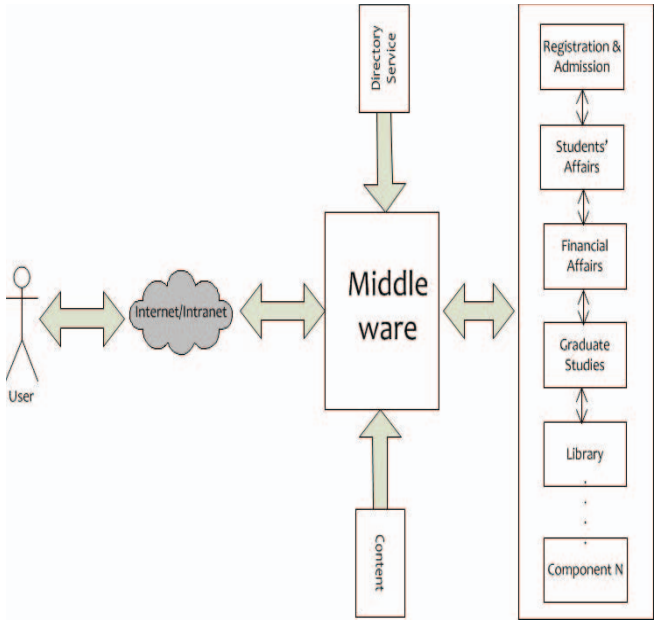

Fig.2. General University E-System Flow of Actions

Figure 2 shows the general university online system flow of actions. In addition to the portal functionalities in a general university system, the university system provides users with online browsing, in which a user can browse the university website content and follow its news and advertisements. The user signs in to gain access to different information resources and services, whereas; the user selects the information and services of his/her interests among the available categories. In addition, users can view and update their own profiles.

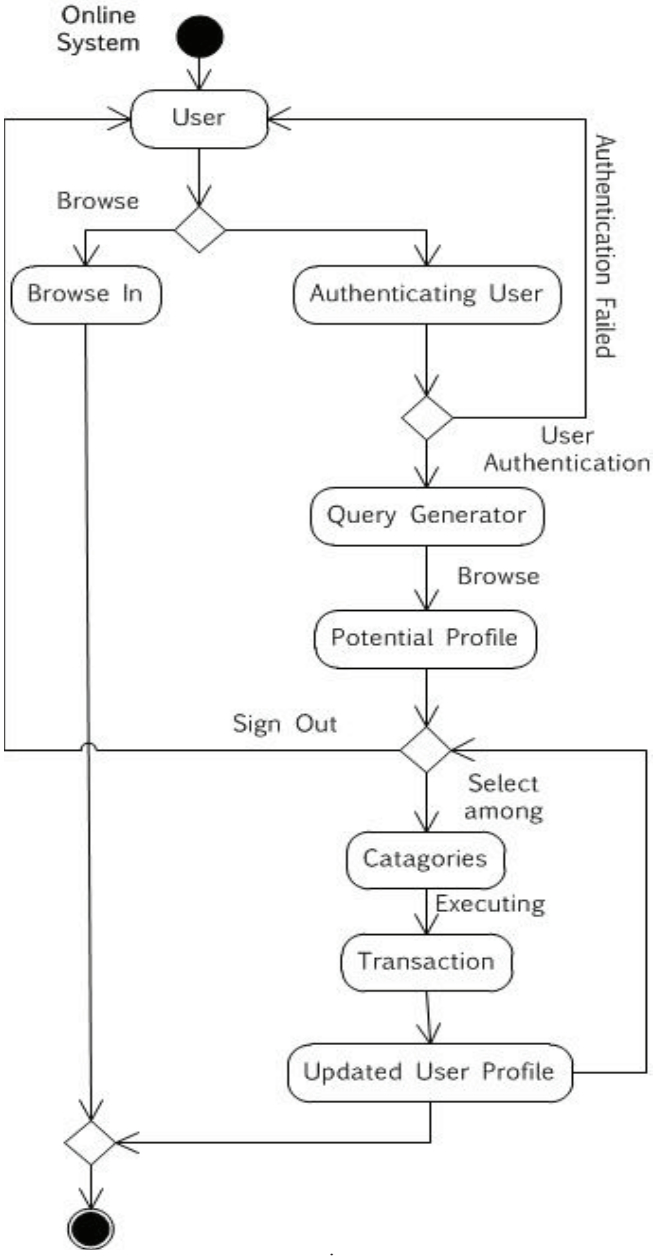

Fig.3. General university online system flow of actions.

III. TIGHTLY-COUPLED UNIVERSITY SYSTEM

Figure 3 shows the current common computerized system used at universities, which aggregates the common features

of a university portal and shows that academic and administrative components are related and dependent on each other. This framework adopts tightly-coupled architecture to compose student portal systems.

As shown in figure 3, the users (students or staff) that are connected to the internet/intranet can use the directory service and update portal content. In addition, the users can view and update their profile and specific needs via registration and admission, students' affairs, financial affairs, graduate studies, library and/or any other extended components. Furthermore, the middleware is a computer software that provides services to software applications that are not directly applicable to the operating system. Therefore, middleware is not certainly part of an operating system, and not a database management system. The middleware component shown in figure 3 is a general abstract middleware that can be customized to suit any university portal. For example, Boston College has defined "Integration Broker Portal" as a middleware for their colleges' portal.

IV. SERVICE ORIENTED ARCHITECTURE [1]

Service Oriented Architecture [1] based applications are highly dynamic, by defining loosely coupled architecture that is subjects to frequent changes. The framework based on SOA, which is an architectural approach that effectively eliminates redundancy, while simultaneously accelerating the delivery of projects through consolidation and the reuse of services [10]. Therefore, SOA is adopted in this research as the main architecture; to achieve loose coupling of software components (service provider, service consumer, and agents).

SOA is the most popular technical framework for web services realization [11]. It uses Web Services Description Language (WSDL) to describe the services, which does not include any technical details of service achievement.

V. UNIVERSITY SYSTEM BASED ON SOA

Figure 4 shows the new framework in which all academic and administrative departments of a university portal are distributed to components of a complete portal. These components are software packages or modules that encapsulate a set of related data. As shown in figure 4, the users (students or staff) that are connected to the internet/intranet can view and update their profiles and specific needs via registration and admission, students' affairs, financial affairs, graduate studies, library and/or other extended components. Furthermore; the Orchestration Point is the station to verify the validity of user and to identify his/her request. The new students' portal framework that is based on SOA has several advantages over the traditional framework. These advantages are summarized in table 1.

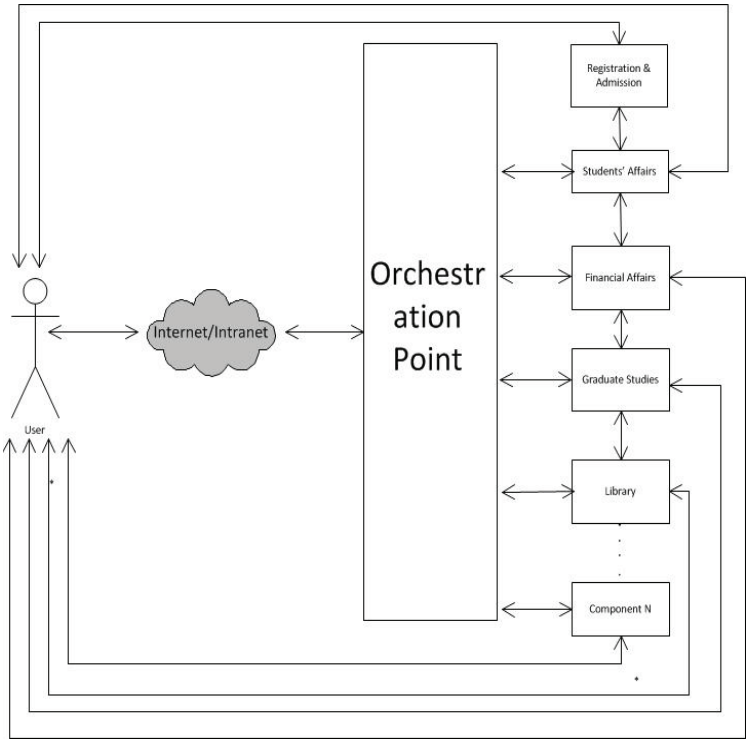

Fig.4. The new framework based on SOA

Table 1 shows a comparison between the traditional framework and the new framework regarding to coupling, distribution, interoperability, maintainability, cost, complexity and reusability.

As shown in table1, the traditional framework components are tightly-coupled. This traditional framework shows that all portal components are dependents on each other. On the other hand, the new framework is a loosely-coupled framework with independent components. In addition, the distribution of the traditional portal framework face difficulties to distribute the portal components in an orchestrated way that can easily be implemented in university systems. A change in one component in the traditional framework leads to other changes in other components. However, the new framework totally supports components distribution and interoperability. It shows that portal components of a university are distributed portal components, and the framework allows any component to be added or modified without any effect on other components. Furthermore, the traditional framework components have limited reusability. The traditional framework shows that any portal component might be more difficult to reuse because dependent components must be

included. On the other hand, the new framework components are reusable in which any portal component can be reused because dependent components are already included. Moreover, the traditional framework components have impediments to maintain, in which portal components might require more effort and/or time to maintain due to the increased inter-component dependency. However, the new framework components are easy to maintain due to the loose coupling features that isolate components and then isolate components maintenance.

The traditional portal framework is difficult to extend. Any change or extension on the traditional framework usually forces a ripple effect of changes to other components because of lack of distribution. On the other hand, the new portal framework components are easily extensible, which shows that any portal component of a university can be extended or modified without any effect on other components. In addition, the traditional framework components are more complex, in which portal components depend on each other's which increases time and effort of components coding, while the new framework components are less complex because portal components are isolated components, which does not require more effort to code the portal components.

The loose-coupling features of the new portal framework together with other features shown in table 1 reduce the cost of implementing and maintaining a university portal.

TABLE 1 COMPARISON BETWEEN TRADITIONAL FRAMEWORK AND NEW FRAMEWORK

	Traditional Framework	**New Framework**
Coupling	Components are Tightly-Coupled	Components are Loosely-coupled
Distribution	Difficult to Distribute	Supported
Interoperability	Not Supported	Supported
Maintainability	Has Impediments	Easy to Maintain
Extensibility	Difficult to Extend	Easy Extensible
Cost	Relatively High	Relatively Low
Complexity	More Complex	Less Complex
Reusability	Conditionally Supported	Supported

VI. CONCLUSION

The students' portal represents a major part of a university information system. Current students' portal framework adopts tightly-coupled architecture to compose students' portal systems. This tightly-coupled architecture causes problems due to the lake of components distribution. The design of a new students' portal framework, which is loosely-coupled has several advantages over the traditional framework such as distribution, interoperability, maintainability, extensibility, cost effectiveness, complexity and reusability.

REFERENCES

[1] D. Z. Alexander Davis, "A Comparative Study of DCOM and SOAP," *IEEE,* 2002.

[2] Y. B. Jin Guojie, Zhao Qiyang, "Enhance Reusability with Application-level Software Components," *IEEE,* 2010.

[3] X. Y. Xiao Dong Wang, Rob Allan, "Top Ten Questions To Design A Successful Grid Portal," *IEEE,* 2006.

[4] W. Z. Dancheng Li, Shuangshuang Zhou,Cheng Liu,Weipeng Jin, "Portal-based Design for SaaS System Presentation Layer Configurability," *IEEE,* pp. 1327-1330, August 3-5 2011.

[5] W. Zhang, "2-Tier Cloud Architecture with Maximized RIA and SimpleDB via Minimized REST," *IEEE,* 2010.

[6] J. X. Xiaohong Hu, Hongjie Liu, "Research of Architecture Pattern Based on .NET Distributed System," *IEEE,* 2011.

[7] Y. L. Mao Tan, "Design and Implementation of General Distributed Heterogeneous Data Exchange System," *IEEE,* 2011.

[8] Y. H. Ming-Qiang Guo, Xian-Gang Luo, Yong Liu, "Design and Implementation of a Distributed Web Service Directory in SOA-oriented Urban Spatial Information Sharing Platform," *IEEE,* pp. 127-130, 2009.

[9] T. U. Thomas Arndt, Karsten Einwich, Ingmar Neumann, "Using SystemCAMS for Heterogeneous Systems. Modelling at TIER-1 Level," *IEEE,* pp. 1-6, 2010.

[10] J. S. Fitri Susanti, "The Mapping of Interconnected SOA Governance and ITIL v3.0," *IEEE,* 2011.

[11] W. D. Mengxing Huang, "Research on Acooperative Model for Service Chain of Digital Library.," *ACIS,* November 2010 2010.