



Content Based Messaging Model for Library Information System

Surbhi Agarwal, Chandrika Chanda, Senthil Murugan B.

School of Information Technology and Engineering,
Vellore Institute of Technology
Vellore, Tamil Nadu, India.

ABSTRACT

This paper presents an application that introduces publish-subscribe model based service in the existing library information system. It applies content based messaging model in order to provide the due-date reminder service through SMS in the library system. The users of the library could subscribe to receive SMS notifications about the due date of their books as per their convenience and the library system will publish the information to its users. The users themselves select when they want to receive their notifications. Hence this model would be based on content based filtering. This paper proposes the addition of such a module to the existing library information systems in an attempt to enhance it.

Keywords

Content-based messaging, library management system, publish-subscribe, messaging model, web services.

1. INTRODUCTION

In today's cyber-world, it is becoming increasingly essential for all day-to-day facilities to be available in the form of web services. The majority of the student population finds it more convenient for all services and notifications to be delivered to their mobile phones as SMS instead of e-mails. Many libraries that have a web interface provide services like searching for available books, reserving books that are loaned out and requesting for new books through their websites. Services like re-issuing of books or reminders about books that are due are also provided, by sending a notification e-mail. The Short Messaging Service (SMS) has almost made e-mails obsolete in many walks of life.

The advancement of technology, especially in the areas of computer and networking technologies indicates an information era. This change needs to be implemented in all walks of life in order to keep up with technological advancements. There is a need to improve the traditional library management systems with the inclusion of emerging technological trends that are more user-friendly and make the library experience traditional yet cutting edge. The most common medium for providing any service today is the internet, and so it is essential for the library to start offering web services to maintain its clientele.



Existing enterprise library management systems provide solutions for basic needs for the management of the library from an administrative perspective. The only interaction that the client of the library has with the system is while accessing the OPAC and that again is not usually available on the internet. Newer functionalities like auto-generated reminders, subscribing to interested books and requesting for renewals could be implemented in the existing systems. These features can be provided by the use of concepts like publish-subscribe messaging model.

This paper presents a messaging model enhancement for an existing library system to implement the features of auto-generated reminders using Short Messaging Service (SMS) and notification services to interested subscribers using a web based platform. This system requires the Java 2 Enterprise Edition, Apache Tomcat Server, JSP, and Oracle 10g Database on a Windows 7 operating system for its implementation.

This paper proceeds as follows. In the next section, the related works in this field are described. Section 3 describes the proposed work. Section 4 describes a simple architecture for the proposed system along with a short algorithm. Section 5 discusses the results after implementation of the system and section 6 gives a conclusion.

2. RELATED WORKS

The existing library management systems are based on two tiered client-server architecture [1]. This system has several disadvantages since the development cycle is quite long and the amount of resources required from the client side is significant. Also problems related to installation, maintenance and scalability grow due to the usage of this model [1].

In India, very few colleges provide a web based library service for the students and the staff. These libraries implement an Integrated Library System which is an Enterprise Resource Planning system for libraries. It comprises of relational databases, software which serves as a middleware to interact with the database and a visually aesthetic graphical user interface for students and staff. It has various modules for acquisitions, cataloguing, classifying and indexing all kinds of materials, serials and OPAC [2]. Most libraries have their own system to which the user has limited access. Library users generally fill the form and data entry operators are employed to update the form contents into the database. When the due date of an item passes away, often the user forgets and is unable to return the item on time. Thus he/she has to pay huge amounts of fine and other users who may have requested for the book have to keep waiting.

During our research we found quite a few popular library management systems. One such system is provided by *Navayuga Infotech*. The system does have a '*generate overdue notification*' module [3], but this module does not send the notification generated to the borrowers (subscribers) of the



library service. Similar open source library management software is Evergreen, which also provides management services to the library and only searching facilities to the borrowers or patrons of the library [4]. Another such software that we came across was, *PhpMyBibli*, which is an open source integrated library system which provides cataloging and circulating feature with no mobile based notification services [5].

The existing systems provide features for improving the management of the library alone. They still do not provide any function for enhanced user experience. In the existing systems all service notifications are sent to the library users through emails. It is also common knowledge that students in general are lax about checking their emails regularly. The above stated problems have led to our proposal which will be able to solve most of these issues and make the library experience more modern and more suited to the younger generations of today.

3. PROPOSED WORK

As discussed in the above section the stated problems provide a new research scope for improving the library information system. We intend to convert the system of sending notifications through emails into a system that sends notifications through SMS to the mobile phones of the subscribers. For this we use publish-subscribe messaging model with content based filtering.

This system will be able to solve the problems faced till now. Since there is no need to check emails, so the problems of slow internet access, and intranet-based service will be eliminated. Also students who instantly check their SMS inbox will be reminded on time about the due date of their library items and hence will save on the large amount of fine they have to pay on overdue items.

The system will have a backend database which will be accessed using JSP at the front end. There will a login module for students, staff and administrators. The entire system will be web based. We will also provide a new book arrival notification service to all interested subscribers. Subscribers will also be notified on the arrival of the book that they have requested for but was not available earlier.

Since the system is being developed as a web application, this system will be reusable and can be implemented by any library vendor to improve his/her business. As print media is almost dying [6], we think that many such libraries will be interested to look for ways to attract customers and keep their business running by using easy web applications to provide quality services to their customers.



4. METHODOLOGY

The three main components of publish-subscribe model are the subscriber, the publisher and the message. The subscriber is the one who subscribes to receiving notifications of some information. The publisher is the one who supplies or sends the information to all its subscribers. The message is the way in which information is communicated between publishers and subscribers. Event-driven or notification based interaction pattern is most commonly used for inter-object communication [7]. The above described notification service serves as a middle layer between publishers and subscribers to avoid each publisher's requirement to know all possible subscribers.

The publisher and subscriber both communicate with a single entity – the Notification Service. All the subscriptions associated with respective subscribers are stored by the notification-service. This service also dispatches the published notifications to the correct subscribers [8]. The filtering process is based on the content, topic or type of the message. Content based filtering is most suitable for our application since the subscribers receive the messages which match the constraint defined by the users themselves. In this case the user (student, staff) himself/herself decides when he/she wishes to receive the reminder notification, a number of days in advance. He/she also decides which updates regarding new books he/she wishes to receive.

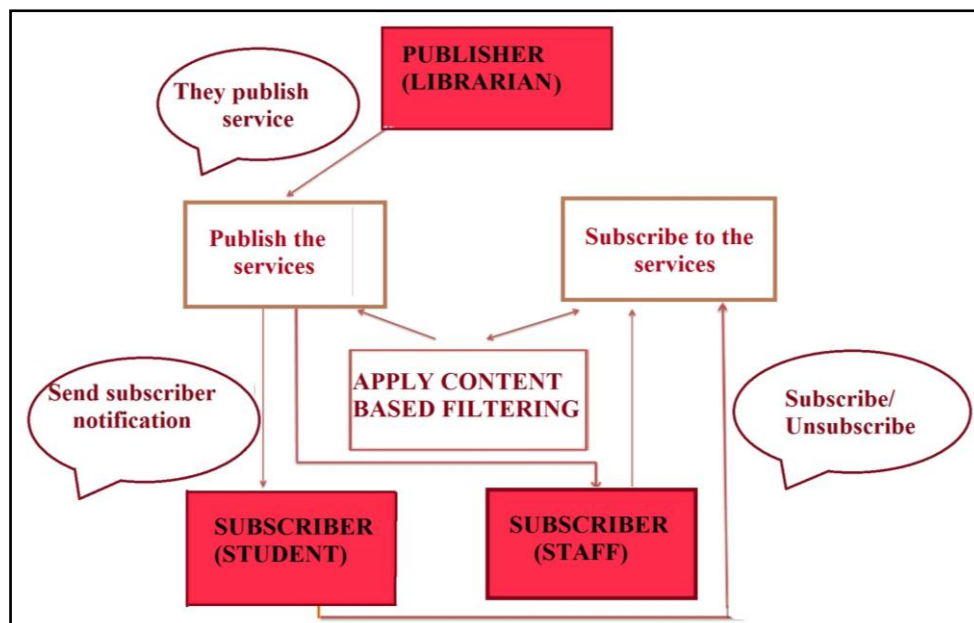


Figure 1. Publish-Subscribe Architecture Model for Library Reminder Notification Service



IJCSBI.ORG

Figure 1 explains about the basic architecture of publish-subscribe model using content-based filtering technique. As shown in the diagram, the role of the librarian is that of the publisher and the students and staff, that is the borrower, plays the role of the subscriber. The publisher publishes the services and the subscriber subscribes to these services according to the filtering constraints. The subscriber service also allows the subscriber to subscribe or unsubscribe to any registered service.

```

Send (msg, subscriber id){
    If (id is student id){
        Do :no=retrieve mobile no from student table;
        delvr_msg(no, msg);
    }
    Else if(id is staff id){
        Do :no=retrieve mobile no from staff table;
        delvr_msg(no, msg); }
    }
Notify (publisher id, subscriber id){
    date_diff:value;
    Do subscribe_filter=date_diff
    Switch (subscribe_filter){
        Case 1: do send (msg, id) break;
        Case 2: do send (msg, id) break;
        Case 3: do send (msg, id) break;
    }
}
notifynewbook(publisher id, subscriber id, category){
    do subscribe_filter=category
    switch (subscribe_filter){
        case category1:do send(msg, id) break;
        case category2:do send(msg, id) break;
        case category3:do send(msg, id) break;
        .
        .
        case category N:do send(msg, id) break;
    }
}
}

pub_sub_msg() {
    main()
    subscriber
    publisher
    do : subscriber id=retrieve each subscriber id according to
    date_diff
    do :notify(publisher id, subscriber id)
    do: notifynewbook(publisher id, subscriber id, category)
}

```

Figure 2. The proposed algorithm



IJCSBI.ORG

Figure 2 describes the algorithm used to implement the system. This algorithm has two primary modules each of which implement publish-subscribe model. The `pub_sub_msg()` acts as the entry point. It identifies the publisher that is the Librarian and the Subscriber that is the staff and student who are likely to avail this facility of the library. In the next step content based filtering is applied for both the modules. The content pertains to the number of days prior to which the notification must be sent as well as the list of book categories subscribed to. The subscribers are identified according to their subscription requests for reminder time of due date of books and for availability of interested books. In the first module the difference between due date of book and present date is calculated and accordingly reminder message is sent, based on the subscriber's preference. In the second module, the subscribers are requested to identify the categories they are interested in and according to their choice whenever a new book is added to that category, a notification is sent to the subscriber. To implement our library notification system, we used Java 2 Enterprise Edition to implement a web service using NetBeans 7.2 IDE. This service required the use of JSP pages to create the front end and handle business logic using standard Java classes and Servlets. The notifications will be sent using URL modification. This service will be hosted using Apache Tomcat 6.0 Server. For the backend we used Oracle 10g relational database on Windows 7 operating system.

5. RESULTS

The proposed system was implemented as a web application. It was deployed on the local server of a machine and temporary data was inserted into the database to test the working of the system.

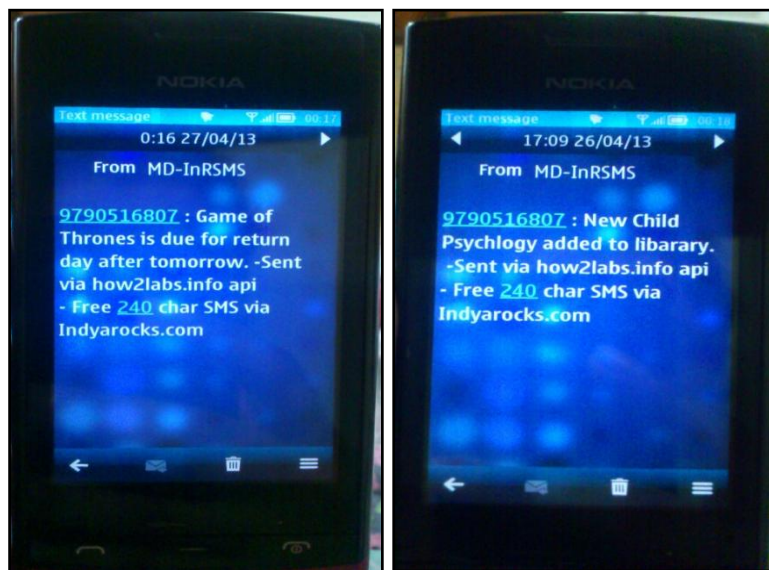


Figure 3. Notification SMS



IJCSBI.ORG

Figure 3 shows the messages received for books due and for new additions to the library. The message is sent to the subscribers, but not to all subscribers. The selection of the subscribers to which the message has to be sent is based on the constraints specified by the subscriber as per the content-based filtering technique. Content-based filtering is applied by categorizing the contents as subscriptions options and allowing the subscriber to select to which content he wishes to subscribe. The selection of the number of days prior to which reminder notification has to be sent, as well as the selection of the interested book categories provides constraints necessary in content-based filtering of publish-subscribe model.

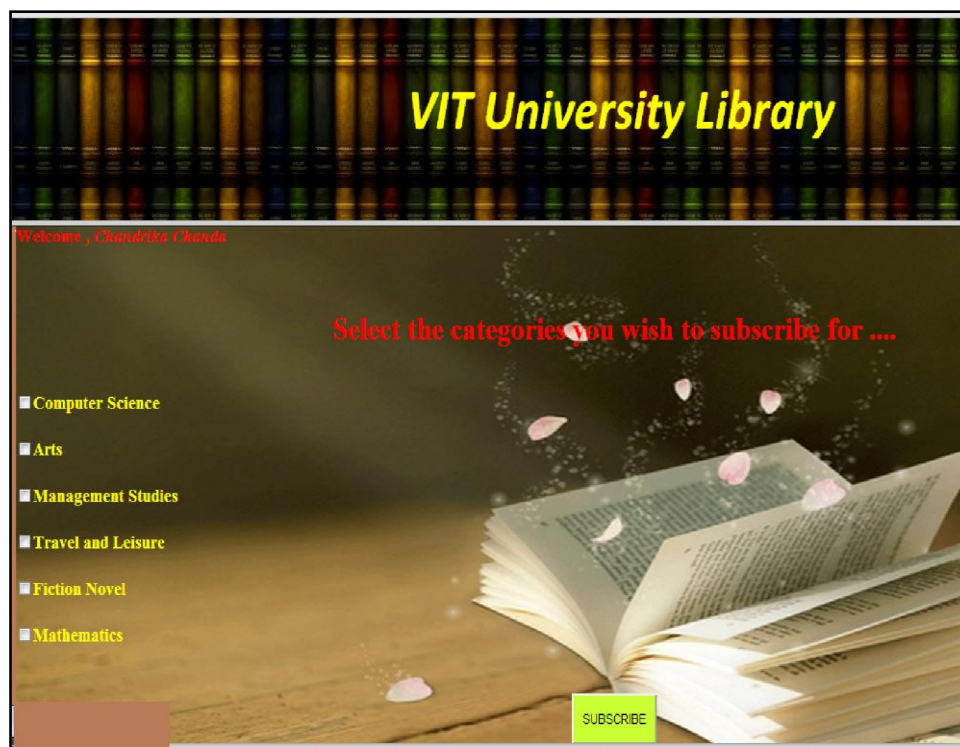


Figure 4. Screenshot of the subscription page

Figure 4 shows a screenshot of the page displayed when a subscriber wishes to subscribe to some category of books that he/she wishes to be notified about. Based on the options selected by the subscriber on this page, appropriate notifications will be sent to him/her whenever the content of the category attribute of the new book matches to the constraints specified by any of the subscribers.

The data relating to user's interested subscriptions was retrieved in the form of number of days of reminder from the databases and appropriate message was sent. The successful delivery of message was displayed on screen and



failure was displayed too. The messages were on an average successfully delivered to the subscribers and it helped them to avoid paying fine for late return of library item.

6. CONCLUSION

The student population will benefit highly from this new system. The library services provided to the students will be considerably enhanced in various areas. Since students have financial issues and paying huge amounts of fine to the library for delayed returns often leads to irritation and frustration. This system will be able to help the student remember at the right time and avoid paying fine. It will also inform the students about the new arrivals in which they have shown an interest and hence they will not need to go to the library everyday to find out if their books have arrived or not. Since it is a web service it can be deployed everywhere with little maintenance. Any library vendor can make use of this service. So this will improve the library facilities and will keep it at par with the advancement of technology that is so rampant these days.

7. REFERENCES

- [1] Yujun Li, Hao Zheng, Tengfei Yang, Zhiqiang Liu Design and Implementation of a Library Management System Based on the Web Service. *Multimedia Information Networking and Security (MINES)*, 2012 Fourth International Conference on , vol., no., pp.433,436, 2-4 Nov. 2012 doi: 10.1109/MINES.2012.94
- [2] Integrated Library Systems : http://en.wikipedia.org/wiki/Integrated_library_system [Accessed on 10 April 2013]
- [3] Navayuga Infotech Library Management System Brochure : <http://www.navayugainfotech.com/multimedia/lms.pdf> [Accessed on 25 April 2013]
- [4] Evergreen Library Management : <http://evergreen-ils.org/> [Accessed on 25 April 2013]
- [5] PhpMyBibli : <http://en.wikipedia.org/wiki/PhpMyBibli> [Accessed on 25 April 2013]
- [6] Death of print media : <http://www.davidakka.com/enterprise-mobility/deathofprintmedia/> [Accessed on 25 April 2013]
- [7] Oasis Web Services Base Notification : http://docs.oasis-open.org/wsn/wsn-ws_base_notification-1.3-spec-os.htm [Accessed on 7 April 2013]
- [8] Shehnaaz Yusuf. Survey of Publish Subscribe Communication System. *Advanced Internet Application and System Design*, 24 December 2004