

Community Services Portal Based on the Campus of Technical University of Lodz. Using Ajax Technology

Michał Cisz, Wojciech Zabierowski

Abstract – The theme of the work is to design a web portal that integrates community services and the practical implementation that uses Ajax technology to build intuitive, user-friendly interface. To solve the problem contained in the topic was not purely theoretical, it will be presented on the example of site built to the needs of the VII Student House of Technical University of Lodz. The application will be the central point of access to services offered by the computer network in the dorm, will enable the use of the services available in the campus of TUL, and will serve as a source of current information. In addition, it is designed to support communication between the administration and residents as well as communication between users.

Keywords – Ajax, Web Service, SSL

I. INTRODUCTION

The portal is a unique solution in the campus of the Technical University of Lodz, and perhaps also on the country scale. In itself is an interesting design challenge. It is easy to build something from scratch, completely controlling the process of creating a service. More difficulty to integrate, where is need for a combination of several of the user accounts, each with potentially different login and password, within the same account throughout the portal. This forces the need to define (design) protocols and methods of communication between the portal and services, taking into account many factors.

II. ENVIRONMENT

Portal was designed to work in certain environment on the dormitory server. Below is the list of components which are required to run this web application.

- FreeBSD or other unix/linux operating system
- Apache HTTP Server with SSL support
- OpenSSL tools to build own certificate authority and generate all needed (server and user) certificates.
- MySQL Database – recommended, but it is possible to use other database management system after small changes in database connection configuration file.
- PHP technology

III. FRAMEWORK AND LIBRARIES

Using frameworks and libraries lets us to write application code faster, with less effort. Using well documented frameworks is good idea. We do not need to repeat comments – most of them is in manual. Our code is also readable for another programmers.

CodeIgniter

It is simple, flexible framework for PHP with less overhead and many helpers. It implements Model-View-Controller pattern and supports friendly URLs and anti-XSS filters by

default. CodeIgniter offers among other things abstract database access layer (with ActiveRecord to build SQL statements) and something like DAO objects.

jQuery

This is a concise toolkit supports implementing client side application in JavaScript. jQuery library simplifies parsing DOM tree, event handling, creating animation and using asynchronous XMLHttpRequests. Additional jQuery UI supplies nice looking basic user interface controls.

ExtJS

It is browser independent JavaScript library to built rich internet application with such elements like: windows, panels, layout managers, toolbars, progress bars, menus, trees. ExtJS has dual license, for open source usage is under GPL license.

IV. SERVICES

Every web service has identical, repeatable communication process. Each of them needs service proxy object that provides and maintains data interchange protocol and transforms sending and receiving information. Usually service proxy has cache memory to accelerate getting response and to decrease connections usage.

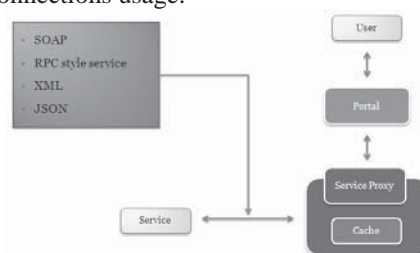


Fig. 1 Abstract communication process with service

The communication process steps:

1. User sends a request of using service
2. Application communicates with service proxy
3. Service proxy check data from cache memory and if it is up to date – go to step 6
4. Proxy communicates with services using SOAP, XML-RPC protocol
5. Service processes the request and return response
6. Proxy send the receiving data to application
7. Application presents results to end user

There are three common used kind of web services:

- Web services describing with WSDL and using SOAP protocol in communication process
- REST services
- RPC style services

The services get response in following formats:

- Flexible but heavy XML document
- JSON notation, known from JavaScript language
- Plain text

Following services are currently available for users:

- Imap client service – access to e-mail account, e-mails management

- Discussion board – last topics from dormitory discussion board
- Weather in Lodz
- Public transport timetable
- Dormitory inhabitants database
- RSS reader – universal feed reader

V. APPLICATION

Application consists of two main parts – server and client side.

Server side (Core)

To implement this part of application PHP technology was used. This language is a good choice in case of small and medium community solutions. PHP is distinguished by stability, speed, low requirements and low cost of implementation and has a lot of open source libraries we can use.

Server side of application is responsible for business logic, generating XML/JSON responses for user interface controls and certificates management.

Client side (UI)

User interface was implemented in JavaScript, XHTML and CSS. Windows, grids, panels, toolbars, menus was built using ExtJS framework and jQuery.

Single Page Application Pattern (SPA)

Client side of portal was designed according to the *Single Page Application* pattern. It means that all application is like a single html page without content reloading. All needed changes results from interaction with user are made as asynchronous request and DOM manipulation. This pattern reduces size of information exchanging between server and browser (only required data is loading), makes user-application interaction easier and fluent. Using SPA needs to implement some elements of business logic into user interface controls. This situation allows us to make efficiency profit, because it decreases usage of server.

VI. USER INTERFACE

In modern internet application user interface and overall look and feel is important such a functionality. Ajax and XHR change traditional websites into rightful application known from PC.

Using ExtJS library lets to build portal as SPA. Browser window is split on header, footer and portlets area. Layout manager helps to automatic fit panels and content to the window.

Main area is assigned for the portlets – separate boxes with contents, which can be freely moved, collapsed or turned off. Application has status area where visual notification (red or green boxes depend on type of operation) are presented.

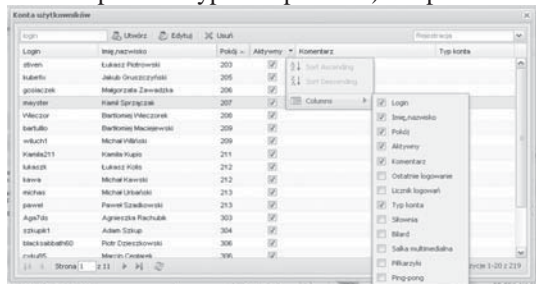


Fig. 2 Power of ExtJS library – users grid

The powerful control is users grid. It supports columns resizing and dynamic hiding, double click event, auto storing selected fields of table row and own defined callbacks rendering table cells.

VII. SECURITY

Security is very important issue. Application collects lots of personal data such usernames and passwords to e-mail or discussion board accounts. To protect this information system offers secure connection via HTTPS. There are also visual indicators that mark you are in secure connection zone, so it has an influence on users consciousness.

An additional advantage of the portal is to use certificates for users. They have opened up new opportunities. Provide secure, encrypted connection, but what is extremely important for the average user, eliminates the need to login every time, which in this case is done automatically on the basis of the data contained in the certificate previously installed in the browser. Certificates are more convenient than the logins and passwords.

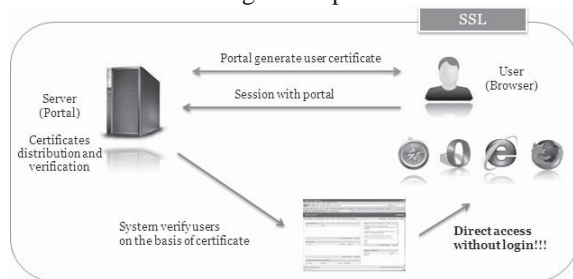


Fig. 3 User certificates in action

Every user can request for the own personal certificate. This digital document verifies user, guarantees high level security and allows to establish user's identity. User has direct access to the system if he has one. Additional login process is unnecessary, all information are inside user's certificate!

VIII. WHAT NEXT?

Author hopes that a successful implementation of this application may encourage the other dormitories to build similar solutions that, over time, combined, can create a network of portals for a service and information to which access will also have the cultural events organizers, the owners of catering outlets, the employers.

IX. REFERENCES

- [1] T. O'Reilly, What Is Web 2.0, <http://oreilly.com/web2/archive/what-is-web-20.html>, 2005
- [2] <http://grono.net>, 2009
- [3] <http://www.php.net/manual>, 2009
- [4] E. Lecky-Thompson, H. Eide-Goodman, S. D. Nowicki, A. Cole, PHP5 Zaawansowane programowanie, Wydawnictwo Helion, Gliwice 2005
- [5] M. Mahemoff, Ajax. Wzorce projektowe, Wydawnictwo Helion, Gliwice 2007
- [6] CodeIgniter User Guide Version 1.7.1, http://codeigniter.com/user_guide, 2009
- [7] jQuery Documentation, <http://docs.jquery.com>, 2009
- [8] jQuery Demos&Documentation, <http://jqueryui.com/demos>, 2009
- [9] Ext 3.0 – API Documentation, <http://extjs.com/depoly/dev/docs>, 2009
- [10] XMLHttpRequest - W3C Working Draft, <http://www.w3.org/TR/2009/WD-XMLHttpRequest-20090820>, 2009.