

A Project Course Redesign Using Open Social Utilities

Hsui-lin Winkler

Seidenberg School of Computer Science and Information Systems
Pace University
hwinkler@pace.edu

ABSTRACT

This paper describes a redesign of an Information Systems Implementation course using open-source social utilities. We show how an existing IS implementation course can be augmented by using the open platform recently released by Facebook. The key advantage of this approach is to achieve a balance between satisfying academic curriculum requirements and having a real-world development experience without external scheduling and organizational constraints.

Keywords: capstone course, course project, information systems implementation, open social utility, academic curriculum requirements, real-world development experience

1. INTRODUCTION

Most college IS courses require students to work on team-based projects. This is especially demanding in a course designed to have a project as its main component, such as in a capstone course. Often a course project is intended to balance certain academic requirements with a real-world development experience. Problems may arise on both sides including limited student implementation skills, rigid semester scheduling, external client's project timetable, post-installation maintenance, etc. Jones and McMaster (2004) describe how it is possible to achieve a successful outcome. However, in many situations the balance is never easy to achieve. Often, it is difficult to arrange for students to be involved in real-world information system development as part of course work due to various constraints, both in students' lack of specific skills and in external organization schedules. This is especially true for a large system with millions of current and frequent users.

Facebook has become a popular social networking site where many college students post their own personal information, connect with friends, form

interest groups, and plan gathering events. Each college or university is given a sandbox or protected online community such that any university student, faculty, or staff member can join the university community provided they have a valid university email address. Given the easy entry for college students, the number of college students who use Facebook has increased dramatically in the past few years. In recent months, Facebook has expanded its user base outside of the college community. The total number of users reached 24 million in July 2007. The most significant event in the context of system implementation is the recent release of the Facebook system development platform to the public. Now anyone with an interest in developing applications for the system can use the published API to do so (Facebook, 2007). The open social utilities generalize the idea of letting users not only publish personal information or organize group activities, but also build user-preferred Web applications.

The rapid growth of, and tremendous interest in, many social networking applications motivated the author to redesign an IS capstone course by using the open source platform recently released by Facebook. Utilizing such an open source platform not only satisfies the needs for a

course requiring a multi-tiered Web project implementation, it also provides an efficient real-world implementation environment with few constraints in scheduling. Facebook's social platform is a flexible project integration environment. Even for a small group of students, a project can be properly sized and distributed to friends for product testing and deployment.

This paper begins with an introduction to an existing capstone course for IS undergraduate students that implements a Web-based system. It then describes how to use the open social platform available from Facebook to redesign such a course.

2. A CAPSTONE COURSE IN INFORMATION SYSTEM IMPLEMENTATION

2.1 Description A system implementation course based on the IS2002.9 model curriculum as described by Gorgone, J.T. et al. (2002) has been offered as a required core course to undergraduate IS students. Table 1 contains the formal definition and scope of IS2002.9.

The primary goals of this course are for students to learn about the concepts of system implementation, to apply a set of technology skills to implement an information system, and to manage the implementation process. Students are required to have completed the IS core requirements of database management and its prerequisites. The course incorporates basic project management concepts in the beginning three weeks of the course, so a chosen development process can be followed throughout the course. This course covers the systems implementation process and management issues and provides simulated hands-on labs for students to learn and experience how to build an information system. The main component of this course is for each student to complete a course project which is described below.

2.2 Course Project Students can propose a course project from an internship or work-related experience, a specific interest in emerging technology, or from an external client contacted by the instructor. Most students are undergraduate juniors and seniors who have completed most of the IS

core requirements. Many have internship or job experience.

From 2003-2007, eight classes were taught with similar course design. The course was scheduled according to a rapid project development process so that it can be completed in 14 weeks within a semester. Table 2 lists all the projects with at least a B+ project grade completed by students in the past two years. There is an interesting shift of the project topics from more traditional e-commerce services (indicated as Web1.0) to more recent social participation and consumer-centered services (indicated as Web2.0) as advocated by T. O'Reilly (2005). Various project scopes were applied depending on the size of the team and the project idea.

2.3 Implementation Environment and Requirements

Students are required to set up a project development environment with an Apache/Tomcat Server and use either SQLServer or MySQL database for their project development. An environment used in system implementation is illustrated in Figure 1.

A completed project is a web-based application system that can be deployed to a Web server and allow users to access a pre-defined business process. The scope of the system varies but it is in general proportional to the number of the students in a team. Each student is required to contribute to at least one module of the system. For example, *user management* is a module that allows any user to register, to login, and to update member information. *Product management* is another module that allows a business administrator to update a product catalog and trigger an automatic inventory update during purchasing. A module can also include more sophisticated functionality like password encryption and user behavior monitoring or event notification.

Students can choose to use either JSP or PHP technology to implement the application. Various interactive development environments (IDE) are available in the classroom including Eclipse and J-Builder for JSP and DreamWeaver for either technology. Over the years, students' interest in the type of project tend to follow

the e-commerce trend happening in the real-world from a more traditional application such as a Web store for CD rental to a more consumer or user-centered exchange such as an online used-bookstore or a special interest social club. We also note that the emerging interest in Web2.0 that is loosely defined as emphasizing user-generated contents has shifted the technology interest from server to client features.

2.4 Development Process The basic time constraint for the project is that it has to be completed within the semester school calendar. For an external client project, we need to choose a soft schedule for which only prototyping is required. Along with the course syllabus, a project schedule is provided to ensure that the project implementation process is consistent with the course lectures that provide technology reviews, and with the labs that practice server installation, development environment setup, and coding.

3. FACEBOOK OPEN SOCIAL PLATFORM

Facebook is one of the most popular social networking sites, with reported members now reaching about 24 million or greater (see facebook.com). It allows users to publish personal profiles, form interest groups, and participate in events online and offline. Until recently, Facebook members were mostly college students and their associates. Stutzman, F. (2006) has an interesting research report about Facebook and claims that a large majority of his college students, more than 88%, are members of this social network. Similar claims were also made by H. Jones and J. H. Soltren (2005) and by N. Ellison et. al. (2006), although they tend to focus on the privacy and social aspects of the site.

In April 2007 Facebook released an open platform to the public for application development. In the following, a brief review of the Facebook architecture and Facebook application development steps are provided to illustrate how integration with the current course can take place.

3.1 Facebook Architecture If we re-draw the published Facebook Architecture using our notation, we come up with components

shown in Figure 2. consisting of a 3-tiered Web-based browser, server and database. It is a loosely-coupled Web system that is flexible to integrate with other application servers or databases. This indicates that any system application developed in the current course (see Figure 1. for comparison) can be integrated with the Facebook system in a straightforward 'push' by keeping the backend database and application server on the course application server and use the Facebook frontend Web server to connect to the distributed network users. This has a very significant implication for student projects, because the approach can not only fulfill the academic requirements as described previously, it can also provide a real-world project development experience and complete a deployable application to a very popular network system.

3.2 Facebook Application Development Steps

The basic structure of a Facebook application (2007) consists of a searchable directory, a presentation and management page, integration and links to other applications, privacy settings, and other utilities such as news feed, alerts, and message attachments. Since the release of the open social utilities, many non-Facebook applications have been published. Numerous development groups have been formed and many can be found right on the site's developer resources including its recent wiki.

4. A SAMPLE PROJECT USE FACEBOOK OPEN SOURCE PLATFORM

There are two main reasons why we think the Facebook platform is suitable for integrating into the IS System Implementation curriculum. First is that it is a familiar system for college students. Many of the interest groups formed on the site could be considered replicas of a digital college campus. This can encourage and help students not only to engage in the community but also to be creative in bringing new applications to the community. Second, the platform uses a familiar Web service oriented architecture. It is simple and compatible with the basic architecture used in the current course.

We use a simple example project here to describe how the current course project environment can be extended by using the Facebook utilities. We use an application called *My Job Skills* to illustrate how to use the integration point of Facebook server. The *My Job Skills* application is to allow any user to publish their job skills on their main Facebook page. One key concept in the integration is to differentiate the developer's own application server from Facebook's Web server. (We use *localhost* here as the developer's server to simplify the implementation.) The developer can save all the data of interest to his or her own database when a user enters the user data or the job skills as well as other useful user information such as the user ID from Facebook.

The following steps (from 'a' to 'd') show how this simple application *My Job Skills* can be built and integrated with a Facebook profile as an application. It can be viewed by any connected friends or groups once it is published.

a. Construct a basic page in PHP to specify the developer's server URL or the callback URL for the Facebook server. For this demonstration, we use *localhost* so that it can easily be tested by anyone without a real server on the Web. For server and database we use the XAMPP package, which is a bundle of Apache and MySQL and is available for download and installation from apacheFriends.org. The diagram in Figure 3. shows the PHP scripts for the *My Job Skills* application.

b. Have a PHP page as shown in figure 4. that can capture user data. This data can be stored to the database or displayed to the Facebook client page shown below.

c. Test the client page. This is shown in Figure 5. where the *My Job Skills* application appears on the Facebook site with other existing Facebook applications and the basic description of what a user can do. The advantage of using the Facebook provided API is that there are a lot of built-in functions that can be used to simplify the coding process.

d. Code the scripts needed to connect to a MySQL database to store user data to the database. For this demonstration, as shown in Figure 6. the user is set to *root*. Normally this would be set for a specific user with a password. We suggest that in the beginning stage the integrated application be distributed to a small group of friends to be tested before a broad release.

5. DISCUSSION AND SUMMARY

We present a project course redesign using Facebook's open social utilities. The current Facebook system serves many college students and associates, and is rapidly gaining enormous popularity. It is difficult if not impossible to locate such a large, readily accessible system for students to use for classroom implementation. The project course redesign described here takes a web-based information system implementation course and integrates it with the commercial system API. The procedure is straightforward and with little extra effort we can get a course project deployed to a commercial server. Since it is an add-on application to an existing system, the size of the project can be flexible and the application can be innovative.

This approach may be a concern for some because it is relatively new and untested. We recommend that the implementation be carried out through the original course design but to budget about three weeks to integrate it with the Facebook deployment. This can be done by having one week in the early semester for students to become familiar with the Facebook platform and two weeks for the final deployment stage.

As in any new course development, management and assessment are two major concerns besides the adjustments of course contents and schedule. This is different from traditional open source where a software or packages are distributed and it is up to users to figure out what to do. In this case, the system not only has a well-defined environment, but it is operational and has a massive network. The only constraint is a virtual agreement that certain rules be followed and a non-Facebook sticker be posted on the application once it is deployed to the Facebook site. The course management may have to include

monitoring of a specific project that students are working on and a constant observation of the larger eco-system to which the application is being attached. Because the open platform is open to all, many developers are posting their applications to the site at a rapid rate. How to separate the academic component from innovation in a commercial world will be an interesting area to assess for future projects resulting from this new course design.

6. REFERENCES

- Ellison N., C. Steinfield, C. Lampe, 2006, "Spatially Bounded Online Social Networks and Social Capital: The Role of Facebook". Annual Conference of the International Communication Association (ICA), June 19-23, 2006 in Dresden, Germany.
- Facebook Developers Resources, 2007, "Facebook Step-by-step Guide to Creating an Application" <http://developers.facebook.com/step-by-step.php>
- Gorgone, John T., Gordon B. Davis, Joseph S. Valacich, Heikki Topi, David L. Feinstein, Herbert E. Longenecker, Jr., 2002, "IS 2002 Model Curriculum and Guidelines for Undergraduate Degree Programs in Information Systems", ACM, AIS and AITP.
- Jones H., J. Hiram Soltren, 2005, "Facebook: Threats to Privacy", Dec 14, 2005.
- Jones, M. C. and T. McMaster, 2004, "Addressing Commercial Realism and Academic Issues In Group-Based IS Undergraduate Project Work", JISE, vol15(4), P. 375.

O'Reilly, T., 2005, "Design Patterns and Business Models for the Next Generation of Software"

<http://www.oreillynet.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html>

Stutzman F., 2006, "Student Life on the Facebook"

http://ibiblio.org/fred/facebook/stutzman_fbook.pdf Jan 20, 2006.

7. AUTHOR BIOGRAPHY



Dr. Hsui-lin L. Winkler is an Associate Professor in the Department of Information Systems at Seidenberg School of Pace University. She teaches college courses in database management, system analysis, design, and implementation, and project management. Prior to joining Pace University, she had many years research experience conducting both academic and industry projects using data recorded in natural environments. She also worked in recent years on multi-media database integration and visualization, and applied web-enabled technology to design user interfaces for information navigation and integration. She holds a Ph.D. in Geophysics from California Institute of Technology, Pasadena, CA, and a M.S. in Information System Management from Carnegie Mellon University, Pittsburgh, PA.

APPENDIX

IS 2002.9 – Physical Design and Implementation in Emerging Environments (Prerequisites: IS'02.2 and IS'02.8)	
SCOPE	This course covers physical design and implementation of information systems applications. Implementation in emerging distributed computing environments using traditional and contemporary development environments.
TOPICS	Selection of development environments and standards; software construction: structured, event driven and object oriented application design; testing; software quality assurance; system implementation; user training; system delivery; post implementation review; configuration management; maintenance. Multi-tier architectures and client independent design

Year	Project Subject	Project Description	Type
2005 Fall	CD Rental	A web-based CD rental service for members to select CDs online. A set of rental rules were developed and four modules were implemented including customer management, product management, order and payment system, and delivery service. (13 students)	Web1.0
2006 Spring	Cartoon Publisher	A Web service for users to share cartoons, a very innovative project for art students to exchange their creative ideas. (3 students)	Web2.0
2006 Spring	PC Service	A basic Web-based service system that can record all the service calls and answers and archive a searchable knowledge base for reference. (6 students)	Web1.0/2.0
2006 Spring	Emergency Service	A Web-based system for local firemen to respond during an emergency service which provides real-time logistic support and activity monitoring. (3 students)	Web1.0
2006 Fall	Digital Archive	A prototype system designed for a non-profit organization to scan and upload all paper documents for a future searchable archive. The main three modules are document archive, search and download service, user and usage monitoring. (6 students)	Web1.0
2007 Spring	Get Together	A social club site for users to engage in organizing group activities and upload personal profiles. (4 students)	Web2.0
2007 Spring	Ringtone	A web service for users to upload or download cell phone ring tones. (2 students)	Web2.0

Table 2. List of selected course projects with grade of B+ or better.

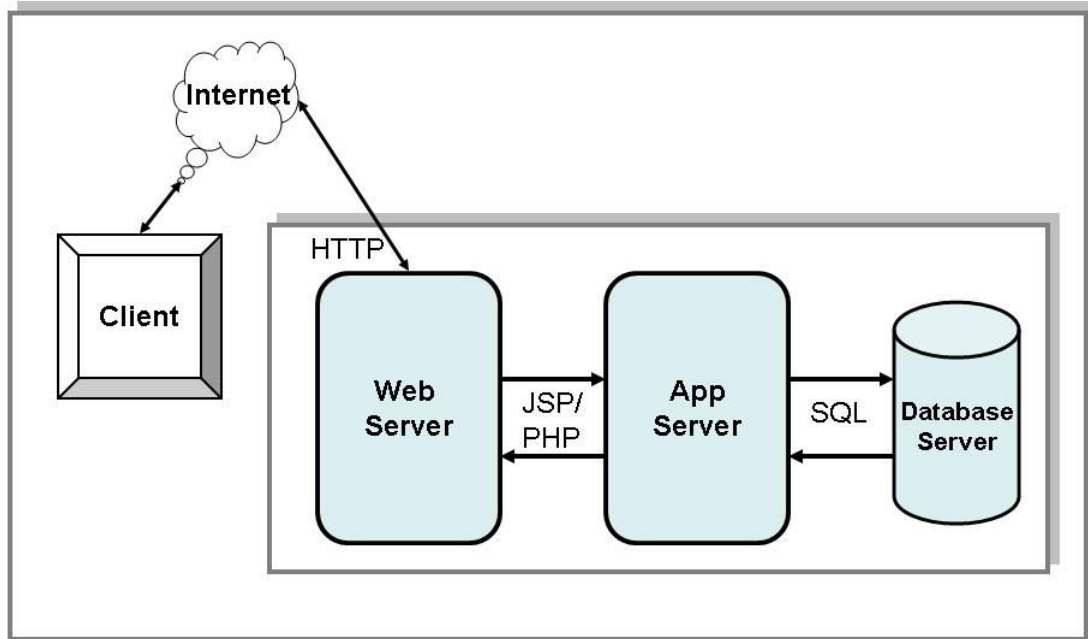


Figure 1. The current implementation architecture.

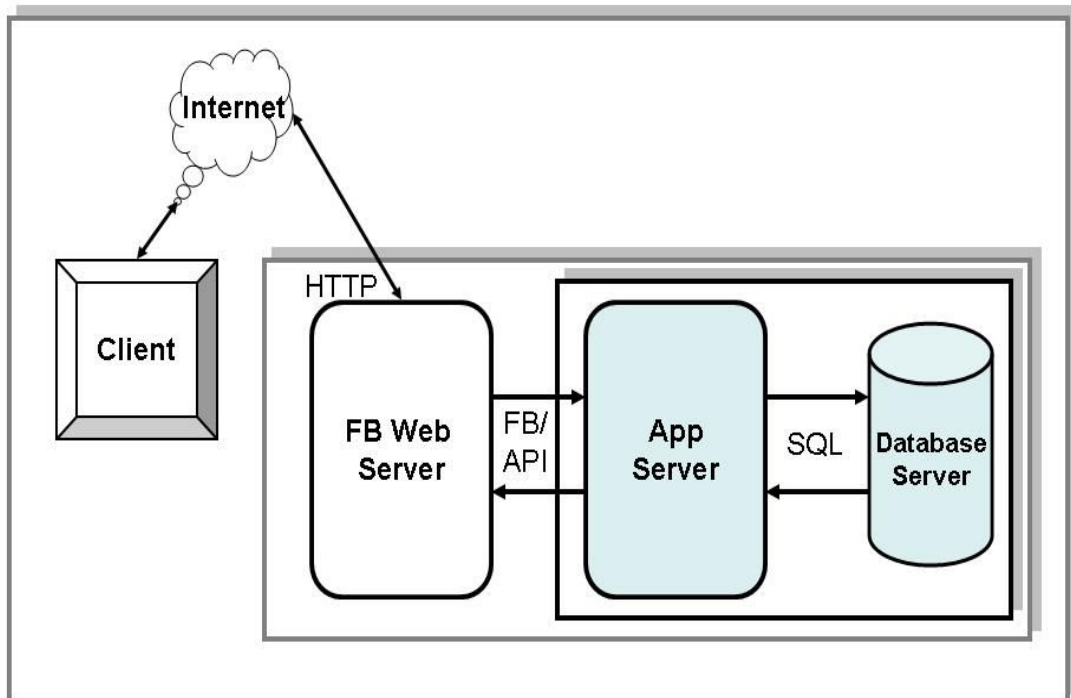


Figure 2. Illustration of the Facebook (FB) architecture and workflow.

```
<?php
require_once 'facebook.php';

$appapikey = '[your app_key]';
$appsecret = '[your secret]';

$facebook = new Facebook($apikey, $appsecret);
$user = $facebook->require_login();

//[todo: change the following url to your callback url]
//$appcallbackurl = 'http://tperry256.dreamhost.com/f8/tutorialapp/';
$appcallbackurl = 'http://localhost/myjobskills/';

//catch the exception that gets thrown if the cookie has an invalid
session_key in it
try {
    if (!$facebook->api_client->users_isAppAdded()) {
        $facebook->redirect($facebook->get_add_url());
    }
} catch (Exception $ex) {
    //this will clear cookies for your app and redirect them to a login prompt
    $facebook->set_user(null, null);
    $facebook->redirect($appcallbackurl);
}
```

Figure 3. PHP scripts describing the steps of defining the developer's URL and how Facebook web server can use it to make connection.

```
<?php

require_once 'dbappinclude.php';

echo "<p>hello $user</p>";









if (isset($_REQUEST['profiletext'])) {
    $facebook->api_client->profile_setFBML($_REQUEST['profiletext'], $user);
    $facebook->redirect($facebook->get_facebook_url() . '/profile.php');
}

echo '<form action="" method="get">';
echo '<input name="profiletext" type="text" size="30" value=""><br>';
echo '<input name="submit" type="submit" value="Display text on profile">';
echo '</form>';
```


Figure 4. A PHP page to connect to client.

Edit My Applications

Use this page to control which applications appear on your profile, application menu, or News Feed. You can change your preferences at any time. You can change your applications' privacy settings from the [Privacy page](#).

 Developer <small>(about)</small>	Edit Settings		Remove
 Events <small>(about)</small>	Edit Settings		Remove
 Groups <small>(about)</small>	Edit Settings		Remove
 job skills <small>(about)</small>	Edit Settings	Profile restricted; News Feed, Mini-Feed, profile links disabled	Remove
 Marketplace <small>(about)</small>	Edit Settings		Remove
 Notes <small>(about)</small>	Edit Settings		Remove
 Photos <small>(about)</small>	Edit Settings		Remove
 Posted Items <small>(about)</small>	Edit Settings		Remove

job skills

 [Share](#)

This is for a user to describe the employable skills, i.e., Java programming, C++, etc

Discussion Board

There are no discussions. [Start the first topic](#)

Reviews

[Write something](#)

Figure 5. A screen capture of author's Facebook application page once the *My Job Skills* application is deployed.

```
<?php
require_once 'appinclude.php';

//This is to setup MySQL database; any data submitted by a client can be
captured by student's own database

//$dbhost = '[your db host]';
//$dbuser = '[your db user]';
//$dbpass = '[your db pass]';
//$dbname = '[your db name]';
$dbhost = "localhost";
$dbuser = "root";
$dbpass = "";
$dbname = "skills";

$conn = mysql_connect($dbhost, $dbuser, $dbpass);
mysql_select_db($dbname, $conn);

function query($q) {
    global $conn;
    $result = mysql_query($q, $conn);
    if (!$result) {
        die("Invalid query -- $q -- " . mysql_error());
    }
    return $result;
}
```

Figure 6. PHP scripts for the backend database connection.