



Eyad Fallatah

June 6, 2011

Loyola University Chicago, Chicago IL 60611, USA,

efallatah@luc.edu

Abstract

Written in PHP and powered by MySQL, 11in1 is an Open-Source Content Management System that does not only manage personal blogs but also maintain postings at social networks. 11in1 establishes consistency among the data transmitted from the main source (the blog) to all targeted social networks and sustains continuous harmonization of the data over time. Distinctively from other Content Management Systems, such as WordPress, this system aims to reduce the amount of time needed to periodically publish entries of commentary.

Additionally, 11in1 promotes interactivity by allowing visitors to leave comments, send messages, acknowledge their visitation in the guestbook and join the built-in mailing list. Conveniently, 11in1 is equipped with a control panel that provides a comprehensive overview of contents and allows bloggers to easily publish, import and export their postings.

Principles of Human-Computer Interaction, such as design disciplines, were carefully considered during the process of developing 11in1 in order to assure simplicity and easiness. The system uses HTML templates to function, which make it easy to personalize and integrate into existing websites and systems. It is this combination of efficiency, interactivity, simplicity and flexibility that distinguishes 11in1 from other blogging systems.

1 Executive Summary

With the approach of a new era of technology, web-based applications, such as blogs, are becoming extremely vital. Blogs provide commentary and news on a wide variety of subjects and areas. In fact, blogs have gained increasing for their role in breaking, shaping, and spinning news stories. These blogs are typically made of a combination of text, pictures, and hyper links to Web pages and other media related resources. The ability of readers to leave comments in an interactive format is also an important part of blogs. This proposal contributes to the enhancement of the mainstream media by introducing a reliable open source Content Management System called 11in 1, which aims to properly managing blogs and pushing messages directly to the public. In order to reach as many individuals as possible, 11in1 makes use of certain technologies to reach the members of famous social networks such as Facebook and Twitter. 11in1 gets its name from the set of 11 applications that comes with it, which are:

1. Entries Manager: is a built-in application that allows the blog owner to add, edit, update and delete entries. The application uses a very sophisticated JavaScript rich-text editor, which modifies the textual data straight away using a technique called WYSIWYG (What You See Is What You Get).
2. Comments Manager: is a built-in application that allows visitors to comment on any entry. This application can be turned on and off by the blog owner and it uses a filtering system that singles out undesirable phrases and words.
3. Social Network Engine: is a built in engine that exports entries and comments to social networks such as Facebook. The application is also capable of importing entries and comments from these social networks into the blog in under 60 seconds.
4. Communication Manager: is a built-in application that allows visitors to communicate with the blog owner using messages. The application uses CAPTCHA, a type of challenge-response test, which attempts to ensure that messages are generated by live people.
5. Guestbook: is a built-in application that allows visitor to acknowledge their visitation to the blog and leave their name, postal or electronic address (if desired), and a comment or note, if desired.
6. Templates Engine: is a built-in engine that allows developers and blog owners to modify the appearance of the blog. It generates HTML 5, CSS and JavaScript files that can be easily modified and/or changed.
7. Activity Monitor: is a built-in application that keeps track of all users activities. The monitor the IP address and the requested pages of each and every user in order to give the blog owner an idea of how to personalize

the visitors' experiences and how to improve the blog. The monitor also enforces security by granting access to restricted pages and resources to authorized users only.

8. Notification Center: is a built-in application that scans the entire system periodically for security threats and possible attacks. It generates daily reports that can only be viewed by the authorized users only in order to bring vital notices to their attention.
9. Backup Manager: is a built-in application that performs hourly, daily, monthly and yearly backups of the files on server and the database.
10. Friendship Manager: is a built-in application that allows all users of 11in1 to keep in touch by linking their blogs to each other.
11. Control Panel: is a built-in application that allows most or all of the settings to be changed through a user interface. Additionally, the control panel gives statistics and high-level view of the performance and status of the entire system. This control panel is only accessed by the blog owner and authorized users (if any). In order to use any of the 10 other applications, the administrator (usually the blog owner), will have to access this control panel using a set of pre-defined username and password.

2 Objectives

To provide a reliable open source Content Management System that serves bloggers of all types particularly those who are engaged in participatory journalism.

3 Problem Statement

Bloggers and individuals with regular entries of commentary usually face troubles while attempting to maintain their personal blog and their social networks at the same time. While some can not afford the time needed to organize and sync their posts, other are in desperate need for systems that dynamically transmit data among the different sources.

Because there is no Content Management System that comes with a built-in synchronization engine, bloggers tend to either sync their posts manually or heir web developers to integrate these functionalities using undependable tools. Such integration does not only jeopardize the stability of the website, but also require deep understanding of web development and access to a great number of APIs.

Even if the integration process was successfully completed, bloggers and webmasters usually find the results of the process extremely unsatisfactory. Such impression is absolutely understandable since the majority of these tools are expensive in terms of memory usage and do not take into consideration major

issues such as code injunctions, security threats, future modifications, framework customization and memory efficiency.

4 Background

According to Harmanci, the modern blog evolved from the online diary, which people used to keep a running account of their personal lives. Justin Hall, for instance, who is known to be one of the earliest bloggers, used his blog to publish reviews from game conferences such as E3 as well as the Tokyo Game Show. Such freelance diarists, journalists, and writers have positively contributed to the rapid increase of use of blogs. Since 2002, blogs have gained increasing notice and coverage for their role in breaking news. The Iraq war, for example, saw bloggers taking measured and passionate points of view that go beyond the traditional left-right divide of the political spectrum (Harmanci, 2005). Blogs are more than tools to just communicate; they are tools to reflect on life, or works of art. Thus, the existence of reliable blogging tools and Content Management Systems is extremely crucial to this type of Internet users.

5 Related Work

There is a decent number of Content Management Systems such as WordPress and Drupal that are currently being used. These systems lack the ability to perfectly transmit data to and from outside resources. Major changes have to be done to the systems in order to get them to allow them to exchange data. 11in1 is specifically built to address this issue and permanently solve this problem without requiring any interference from the user.

There are also so many tools, plugins and widgets such as Tweet-Twoo, Twitterific and Facebook badge that can be integrated into blogs to exchange data with outside resources. Unfortunately, these tools are not reliable enough and lack the consistency. While most of the tools were designed without consideration of the basic security issues such as code injunction, the majority of the tools are really expensive in terms of bandwidth. On the other hand, 11in1 uses Ajax and JavaScript to send and receive data. Instead of refreshing the entire page, Ajax only reloads small portion of pages when new information is imported. 11in1 also saves bandwidth by caching the RSS feed, disabling file hotlink and grouping server request as much as possible.

In order for 11in1 to become significantly different than the already existing Content Management Systems, All the principles of human-computer interaction has been considered while developing the system. Because human-computer interaction concentrates on the human and the machine in conjunction, it draws from supporting knowledge on both the machine and the human side. According to Sears and Jacko, an ideal system has to combine techniques in computer graphics and development environments from the machine side with communication theory, graphic and industrial design disciplines, linguistics, social sciences

and cognitive psychology from the human side (Sears and Jacko, 2007). The two authors suggests that attention to human-machine interaction is important, because poorly designed human-machine interfaces can lead to many unexpected problems. Therefore, implementing the human-computer interaction principles makes 11in1 a unique system that is extremely different than the already existing systems.

6 Milestones

The time required to complete this project is 4 weeks. During this period the following will be accomplished:

- 06-June-2011** Draft Proposal Due
- 09-June-2011** Priority Setting & Situation Analysis
- 13-June-2011** Database Design and Coding Phase
- 16-June-2011** Coding Phase
- 20-June-2011** System Implementation
- 23-June-2011** Evaluation and Accountability
- 27-June-2011** Testing Phase and Collecting Feedback
- 28-June-2011** Finalizing the Project
- 29-June-2011** Done! Final Demo

7 System Requirements

1. A webserver or web hosting account (Linux/Apache recommended).
2. PHP version 4.0.5 or higher.
3. MySQL version 3.23 or higher
4. safe_mode should be disabled

8 Technical Approach

The technical approach for implementing 11in1 is divided into five key components: (1) Preliminary Design, (2) system architecture, (3) database structure, (4) user interface and (5) products (output). The system will be implemented using a phased, rapid-prototyping approach as a way to solicit user feedback in the development process. This will insure that user needs are met.

8.1 Preliminary Design Phase:

This phase will allow us to identify the direction to proceed regarding the proposed functionalities of the systems as well as the needed tools and APIs. The overall functions of the system will be:

- To establish a reliable channel for sending information from and to outside resources.
- To allow users to compose and establish entries.
- To allow visitors to express their thoughts about the content.

8.2 System Architecture:

11in1 will be a stand-alone web-based system that is powered by PHP and MySQL. The system makes use of HTML 5, Ajax, CSS and JavaScript.

8.2.1 PHP

PHP code is embedded into the HTML source document and interpreted by a web server with a PHP processor module, which generates the web page document.

8.2.2 MySQL

MySQL runs as a server providing multi-user access to a number of databases

8.2.3 HTML5

HTML5 is the predominant markup language for web pages. HTML is the basic building-blocks of webpages.

8.2.4 Ajax

Ajax is a group of interrelated web development methods used on the client-side to create interactive web applications. With Ajax, 11in1 can send data to, and retrieve data from the server asynchronously (in the background) without interfering with the display and behavior of the existing page.

8.2.5 Javascript

Javascript provides enhanced user interfaces and dynamic pages. It enables programmatic access to computational objects within a host environment.

8.2.6 CSS

CSS is used to style web pages written in HTML

8.3 Database Structure

The initial database structure is organized into 17 related tables containing information related to entries, comments and activities. The following 17 figures shows the tables that construct the database:

8.3.1 Alerts Table

| | Field | Type | Collation | Attributes | Null | Default |
|--------------------------|-----------------|---------|-----------|------------|------|-------------|
| <input type="checkbox"/> | id | int(30) | | UNSIGNED | No | <i>None</i> |
| <input type="checkbox"/> | comments | int(10) | | | No | <i>None</i> |
| <input type="checkbox"/> | gb | int(10) | | | No | <i>None</i> |
| <input type="checkbox"/> | sub | int(10) | | | No | <i>None</i> |
| <input type="checkbox"/> | mails | int(10) | | | No | <i>None</i> |

8.3.2 Blacklist Table

| | Field | Type | Collation | Attributes | Null |
|--------------------------|-------------|-------------|-------------------|------------|------|
| <input type="checkbox"/> | id | int(30) | | UNSIGNED | No |
| <input type="checkbox"/> | ip | varchar(50) | latin1_swedish_ci | | No |
| <input type="checkbox"/> | date | varchar(50) | latin1_swedish_ci | | No |

8.3.3 Comments Table

| | Field | Type | Collation | Attributes | Null |
|--------------------------|-----------------|--------------|-------------------|------------|------|
| <input type="checkbox"/> | id | int(30) | | UNSIGNED | No |
| <input type="checkbox"/> | author | varchar(150) | latin1_swedish_ci | | No |
| <input type="checkbox"/> | date | varchar(150) | latin1_swedish_ci | | No |
| <input type="checkbox"/> | contents | text | latin1_swedish_ci | | No |
| <input type="checkbox"/> | topicID | int(30) | | | No |
| <input type="checkbox"/> | statusID | int(30) | | | No |
| <input type="checkbox"/> | location | varchar(100) | latin1_swedish_ci | | No |

8.3.4 Contacts Table

| | Field | Type | Collation | Attributes | Null |
|--------------------------|--------------|--------------|-------------------|------------|------|
| <input type="checkbox"/> | id | int(30) | | UNSIGNED | No |
| <input type="checkbox"/> | name | varchar(100) | latin1_swedish_ci | | No |
| <input type="checkbox"/> | email | varchar(100) | latin1_swedish_ci | | No |

8.3.5 Countries Table

| | Field | Type | Collation | Attributes | Null |
|--------------------------|-------------|--------------|-------------------|------------|------|
| <input type="checkbox"/> | id | int(30) | | UNSIGNED | No |
| <input type="checkbox"/> | name | varchar(100) | latin1_swedish_ci | | No |

8.3.6 Frinds Table

| | Field | Type | Collation | Attributes | Null |
|--------------------------|-------------|--------------|-------------------|------------|------|
| <input type="checkbox"/> | id | int(30) | | UNSIGNED | No |
| <input type="checkbox"/> | name | varchar(100) | latin1_swedish_ci | | No |
| <input type="checkbox"/> | url | varchar(100) | latin1_swedish_ci | | No |
| <input type="checkbox"/> | icon | text | latin1_swedish_ci | | No |

8.3.7 Guestbook Table

| | Field | Type | Collation | Attributes | Null |
|--------------------------|-----------------|--------------|-------------------|------------|------|
| <input type="checkbox"/> | id | int(30) | | UNSIGNED | No |
| <input type="checkbox"/> | title | varchar(150) | latin1_swedish_ci | | No |
| <input type="checkbox"/> | author | varchar(150) | latin1_swedish_ci | | No |
| <input type="checkbox"/> | date | varchar(150) | latin1_swedish_ci | | No |
| <input type="checkbox"/> | location | varchar(150) | latin1_swedish_ci | | No |
| <input type="checkbox"/> | comment | text | latin1_swedish_ci | | No |
| <input type="checkbox"/> | statusID | int(30) | | | No |

8.3.8 Messages Table

| | Field | Type | Collation | Attributes | Null |
|--------------------------|-----------------|--------------|-------------------|------------|------|
| <input type="checkbox"/> | id | int(30) | | UNSIGNED | No |
| <input type="checkbox"/> | sender | varchar(100) | latin1_swedish_ci | | No |
| <input type="checkbox"/> | receiver | varchar(100) | latin1_swedish_ci | | No |
| <input type="checkbox"/> | title | varchar(300) | latin1_swedish_ci | | No |
| <input type="checkbox"/> | date | varchar(30) | latin1_swedish_ci | | No |
| <input type="checkbox"/> | content | text | latin1_swedish_ci | | No |
| <input type="checkbox"/> | statusID | int(30) | | | No |
| <input type="checkbox"/> | email | text | latin1_swedish_ci | | No |

8.3.9 Online Table

| | Field | Type | Collation | Attributes | Null |
|--------------------------|----------------|--------------|-------------------|------------|------|
| <input type="checkbox"/> | id | int(30) | | UNSIGNED | No |
| <input type="checkbox"/> | ip | varchar(150) | latin1_swedish_ci | | No |
| <input type="checkbox"/> | name | varchar(50) | latin1_swedish_ci | | No |
| <input type="checkbox"/> | page | text | latin1_swedish_ci | | No |
| <input type="checkbox"/> | timeout | varchar(300) | latin1_swedish_ci | | No |

8.3.10 Pages Table

| | Field | Type | Collation | Attributes | Null |
|--------------------------|----------------|--------------|-------------------|------------|------|
| <input type="checkbox"/> | id | int(30) | | UNSIGNED | No |
| <input type="checkbox"/> | title | varchar(150) | latin1_swedish_ci | | No |
| <input type="checkbox"/> | content | text | latin1_swedish_ci | | No |
| <input type="checkbox"/> | url | varchar(500) | latin1_swedish_ci | | No |
| <input type="checkbox"/> | status | varchar(5) | latin1_swedish_ci | | No |
| <input type="checkbox"/> | type | varchar(5) | latin1_swedish_ci | | No |
| <input type="checkbox"/> | window | varchar(5) | latin1_swedish_ci | | No |
| <input type="checkbox"/> | visible | varchar(5) | latin1_swedish_ci | | No |

8.3.11 Sections eligible for sync Table

| | Field | Type | Collation | Attributes | Null | Default |
|--------------------------|---------------|---------|-----------|------------|------|-------------|
| <input type="checkbox"/> | secID | int(30) | | UNSIGNED | No | <i>None</i> |
| <input type="checkbox"/> | status | int(30) | | UNSIGNED | No | <i>None</i> |

8.3.12 Sections Table

| | Field | Type | Collation | Attributes |
|--------------------------|-----------------------|--------------|-------------------|------------|
| <input type="checkbox"/> | id | int(30) | | UNSIGNED |
| <input type="checkbox"/> | title | varchar(300) | latin1_swedish_ci | |
| <input type="checkbox"/> | disc | text | latin1_swedish_ci | |
| <input type="checkbox"/> | allow_comments | int(1) | | |
| <input type="checkbox"/> | parent | varchar(30) | latin1_swedish_ci | |

8.3.13 Settings Table

| | Field | Type | Collation | Attributes |
|--------------------------|-----------------------|--------------|-------------------|------------|
| <input type="checkbox"/> | id | int(30) | | UNSIGNED |
| <input type="checkbox"/> | site_name | varchar(100) | latin1_swedish_ci | |
| <input type="checkbox"/> | site_url | varchar(100) | latin1_swedish_ci | |
| <input type="checkbox"/> | admin_username | varchar(100) | latin1_swedish_ci | |
| <input type="checkbox"/> | admin_password | varchar(100) | latin1_swedish_ci | |
| <input type="checkbox"/> | admin_email | varchar(100) | latin1_swedish_ci | |
| <input type="checkbox"/> | meta_tag | text | latin1_swedish_ci | |
| <input type="checkbox"/> | css | text | latin1_swedish_ci | |
| <input type="checkbox"/> | bad_words | text | latin1_swedish_ci | |
| <input type="checkbox"/> | replace_words | varchar(100) | latin1_swedish_ci | |
| <input type="checkbox"/> | allow_comments | int(1) | | |
| <input type="checkbox"/> | visitors | int(40) | | |
| <input type="checkbox"/> | login_attempts | int(3) | | |
| <input type="checkbox"/> | lang | varchar(100) | latin1_swedish_ci | |
| <input type="checkbox"/> | template | varchar(100) | latin1_swedish_ci | |
| <input type="checkbox"/> | popup | int(1) | | |

8.3.14 Social Table

| | Field | Type | Collation | Attributes | Null |
|--------------------------|--------------|--------------|-------------------|------------|------|
| <input type="checkbox"/> | id | int(30) | | UNSIGNED | No |
| <input type="checkbox"/> | name | varchar(100) | latin1_swedish_ci | | No |
| <input type="checkbox"/> | link | text | latin1_swedish_ci | | No |
| <input type="checkbox"/> | image | text | latin1_swedish_ci | | No |

8.3.15 Social Network Profile Table

| | Field | Type | Collation | Attributes | Null |
|--------------------------|-----------------|--------------|-------------------|------------|------|
| <input type="checkbox"/> | id | int(30) | | UNSIGNED | No |
| <input type="checkbox"/> | sn | varchar(100) | latin1_swedish_ci | | No |
| <input type="checkbox"/> | name | varchar(100) | latin1_swedish_ci | | No |
| <input type="checkbox"/> | icon | text | latin1_swedish_ci | | No |
| <input type="checkbox"/> | username | varchar(100) | latin1_swedish_ci | | No |
| <input type="checkbox"/> | password | varchar(100) | latin1_swedish_ci | | No |

8.3.16 Todo Table

| | Field | Type | Collation | Attributes | Null |
|--------------------------|----------------|--------------|-------------------|------------|------|
| <input type="checkbox"/> | id | int(30) | | UNSIGNED | No |
| <input type="checkbox"/> | level | varchar(100) | latin1_swedish_ci | | No |
| <input type="checkbox"/> | content | text | latin1_swedish_ci | | No |

8.3.17 Entry Table

| | Field | Type | Collation | Attributes | Null |
|--------------------------|-------------------|--------------|-------------------|------------|------|
| <input type="checkbox"/> | id | int(30) | | UNSIGNED | No |
| <input type="checkbox"/> | sec | int(30) | | | No |
| <input type="checkbox"/> | title | varchar(150) | latin1_swedish_ci | | No |
| <input type="checkbox"/> | date | varchar(150) | latin1_swedish_ci | | No |
| <input type="checkbox"/> | contents | text | latin1_swedish_ci | | No |
| <input type="checkbox"/> | thumb | text | latin1_swedish_ci | | No |
| <input type="checkbox"/> | main | text | latin1_swedish_ci | | No |
| <input type="checkbox"/> | allviewers | int(30) | | | No |

8.4 User Interface

The 11in1 user interface will be developed by using HTML and CSS to customize the default interface. Because 11in1 uses HTML and CSS, bloggers can easily customize the user interface. Alternatively, bloggers can download themes from the official website of 11in1 and download one of the available themes. By uploading the themes to their blogs, bloggers will be able to change the interface style in under 60 seconds.

9 Evaluation of Outcome

1. Poll: A poll will be posted so that all evaluators may rate the system. There will be some fields in which evaluators will have the chance to express what they liked and what they did not like about 11in1.
2. Statistical Tools: The official website of the project, from which the evaluators can download their copies of 11in1 is equipped with statistics tools that gather information about each visitation and the total number of downloads.
3. Professionals Evaluation: Professionals in the computer science fields will be asked to evaluate the system and give their inputs. Their constructive criticism greatly contributes to the enhancements of the system.

10 References

- [1] Harmanic, R. (2005, February 2). Time to get a life - pioneer bloggers
- [2] Sears, Andrew and Jacko, Julie. (2007, April 17). Human-Computer Interaction.