

ОНЛАЙН-СЕРВІС ДЛЯ СКОРОЧЕННЯ ПОСИЛАНЬ НА СТОРОННІ РЕСУРСИ З РОЗМІЩЕННЯМ ІНФОРМАЦІЙНИХ МАТЕРІАЛІВ

С. П. Пашко, студент;

*В. К. Ободяк, канд. техн. наук, доцент,
Сумський державний університет, м. Суми*

Проаналізовано існуючі сервіси скорочення довгих посилань та розміщення рекламних матеріалів. Показано, що відомі рішення мають певні недоліки. Розроблено онлайн-сервіс для скорочення довгих посилань з можливістю розміщення оплачених рекламних матеріалів на сторінках скорочених посилань. Програмна реалізація виконана на мові програмування PHP з використанням СУБД MySQL, засобів безпечного перегляду сторінок від Google, платіжної системи WebMoney.

***Ключові слова:** безпека, посилання, реклама, сервіс, фрейм, css, google, html, mysql, php, безпечний перегляд, url, веб 2.0, вебмани.*

Проанализированы существующие сервисы сокращения длинных ссылок и размещения рекламных материалов. Показано, что известные решения имеют определенные недостатки. Разработан онлайн-сервис для сокращения длинных ссылок с возможностью размещения оплаченных рекламных материалов на страницах сокращенных ссылок. Программная реализация выполнена на языке программирования PHP с использованием СУБД MySQL, средств безопасного просмотра от Google, платежной системы WebMoney.

***Ключевые слова:** безопасность, ссылки, реклама, сервис, фрейм, css, google, html, mysql, php, безопасный просмотр, url, веб 2.0, вебмани.*

Майже усі web 2.0-сервіси є соціальними мережами, адже, за означенням, їх користувачі самостійно наповнюють ці сайти контентом, що і є найвагомішою відмінністю web 2.0 від web 1.0 [1]. Та майже усі «соціальні інтернет-гіганти» вводять свої певні обмеження на розміщення інформації. Наприклад, сервіс мікроблогінгу Твіттер (Twitter Inc.) дозволяє розміщувати публікації у блогах його користувачів не більше 140 символів, що дорівнює довжині одного повідомлення, яке передається латиницею по SMS [2].

Часто посилання на запити у пошуковій системі Google сягають довжини значно більшої за 140 символів, що вже не вміститься у статус на більшості соціальних мереж чи окремою публікацією у Твіттері. Також варто урахувати те, що бажано, окрім посилання, передати деяку текстову інформацію – короткий опис сторінки, на яку користувач потрапить після переходу за посиланням.

Таким чином, необхідно вирішити проблему публікації довгих посилань у друкованих виданнях, соціальних мережах та передачі їх телекомунікаційними мережами [3]. Також потрібна мотивація користувачів для використання сервісу публікації довгих посилань і кошти для утримання серверу з розробленим проектом.

Робота пов'язана з розробленням саме сервісу web 2.0 з можливістю реєстрації користувачів та самостійною взаємодією між ними з мінімальним втручанням адміністративної сторони [1].

Ідея скорочення посилань вже не нова [4]. Останнім часом набув популярності термін «скорочувалка» (рос. «сокращалка»). Це сталося саме через різку популяризацію Твіттера.

У 2002 році компанія TinyURL, LLC вивела на простори Інтернету сайт, що дозволяє скорочувати посилання. Початкова унікальність проекту у тому, що можна власноруч обирати псевдонім лінку (слово

після слешу у записі «tinyurl.com/kichrum»). Але база даних за роки існування сайту настільки збільшилася, що найкоротші дозволені псевдоніми вже сягають семи символів, і це часто не скорочує, а, навпаки, подовжує посилання. Це пов'язано з тим, що використовуються латинські літери без урахування регістру.

Популярність TinyURL спричинила створення великої кількості подібних веб-сервісів [5].

Корпорація Bit.ly Inc. створила більш професійно виконаний скорочувач посилань. Його характеризує достатньо повна статистика по переходах за посиланням. Певний час Твіттер використовував bit.ly як веб-сервіс скорочування URL за замовчуванням. Але цей сайт не може бути використаний для вирішення поставлених цілей, оскільки не надає винагороди своїм користувачам.

Сайт посилань на закриті сторінки reeер.us дозволяє використати браузерну закладку, яка завантажує на сервер копію будь-якої сторінки у тому вигляді, в якому її бачить користувач. Можна показувати приватні повідомлення чи закриті теми на форумах. Використано XMLHttpRequest для завантаження копій, а також розміщення на Google AppEngine, для захисту від можливих DDOS-атак на сервер [6].

Крім звичайних скорочувачів, є такі, що дозволяють розміщувати інформацію «на сторінках», схованих за короткими посиланнями [7].

XXS оплачує усі переходи за короткими посиланнями. Щоб почати заробляти, досить опублікувати короткий лінк. Відвідувачі потрапляють на спеціально створену сторінку всередині домену xxs.ru. Зверху та знизу вікна вони бачать рекламні банери, а в центральному фреймі – власне цільовий сайт. Але розміщувати та оплачувати власну рекламу через цей сервіс не можна.

За результатами аналітичного огляду можна зробити висновок, що проблема найвигіднішого скорочування посилань та розміщення на сторонніх сайтах власної інформації потребує подальшого вирішення.

ПОСТАНОВКА ЗАДАЧІ

Необхідно спроектувати та розробити веб-сервіс обміну довгими посиланнями з такими можливостями:

- а) реєстрація користувачів;
- б) заміна посилань на скорочені з переадресацією на потрібну сторінку;
- в) попередження при створенні та переході за короткими посиланнями на небезпечні ресурси [8];
- г) розміщення додаткових рекламних матеріалів на сторінках за скороченими посиланнями:
 - власне авторами посилань (безкоштовно);
 - рекламодавцями (з попередньою оплатою за покази);
- д) поповнення та виведення коштів з рахунку.

РОЗРОБЛЕННЯ WEB-СЕРВІСУ

1. Розроблення структури програми

Реалізуємо програму на об'єктно-орієнтованій мові програмування PHP з використанням архітектури програмного забезпечення «Model-View-Controller». Згідно з умовами об'єктно-орієнтованого програмування програма повинна складатися з класів, полів та методів у класах [9].

Згідно з архітектурою «Model-View-Controller» (MVC) структура програми повинна розділятися на три окремі компоненти [10]:

- а) модель (Model) – отримує дані від бази даних та надає їх контролеру за відповідними запитами. До моделі віднесемо PHP-класи з відповідними об'єктами «посилання», «клієнт» і т. ін.;

б) представлення (View) – відповідає за відображення інформації. У цьому компоненті буде розміщено шаблон з формами (forms) для взаємодії з користувачем;

в) поведінка (Controller) – інтерпретує дані, що надає користувач, а також інформує модель та представлення про відповідну реакцію.

2. Програмна реалізація

У першу чергу реалізуємо клас MySQL для роботи за базами даних. У ньому передбачено функції підключення, відключення, запитів з різними типами відповідей, а також метод MySQL::quote_smart() для захисту від MySQL Injection, що може зашкодити базі даних та нанести непоправної шкоди [11-12]. Усі методи цього класу оголошено як статичні, оскільки використовується постійний тип з'єднання з базою даних.

Аналогічно реалізуємо клас Client. Використаємо конструктор, адже даний клас відповідає неабстрактному об'єкту Клієнт. Передбачимо поля: ідентифікаційний номер, логін, пароль, реферер, поштова скринька, розрахунковий номер та баланс. Також додаються «методи-гетери» (повертають значення приватних полів): отримання логіну, ідентифікаційного номера і т. ін. Замість більшості сетерів (методів, що мали б встановлювати значення полів) створимо метод Client::update_data(), що отримує всі потрібні дані з бази даних одним запитом та зберігає їх у полях поточного об'єкта.

За відсутньої можливості отримати дані метод повертає рядок «Login failed».

Перший зареєстрований у системі користувач вважається адміністратором ресурсу за замовчуванням, тому має деякі привілеї. Так, адміністратор ресурсу має доступ до кабінету адміністратора, який реалізовано за допомогою методу Client::get_admin_content() та методів, що викликаються через нього.

У класі статичних методів Link між іншими використовується функція Link::convert() [13] (модифікована). Вона здатна конвертувати звичайні числа у комбінацію цифр та літер верхнього і нижнього регістру англійського алфавіту у порядку, визначеному ANSI з пропуском інших символів.

Це означає, що псевдонім посилання, який складається із 7 символів, програма почне створювати лише тоді, коли у базі знаходиться більше ніж 56,8 мільярдів посилань. А порядковий номер посилання, що матиме псевдонім «zzzzzz» (сім символів), дорівнюватиме 3521614606207, тобто кількість відносно коротких посилань вимірюється трильйонами, що є ідеальною умовою для формування дійсно коротких посилань.

Метод Link::is_url() повертає true, якщо рядок, переданий у параметрі, відповідає звичайному посиланню (починається з «http://» та містить у собі крапки відділення зон від доменів), та false, якщо рядок не можна інтерпретувати як URL, що могло б спанталити користувачів.

Також у класі Link розміщено важливий для безпеки користувачів метод Link::is_blacklisted_url(). Цей метод перевіряє наявність отриманого параметром посилання у глобальній базі даних посилань, не рекомендованих для відвідування. Базу даних таких посилань надають лабораторії Google, а саме проект Google Safe Browsing API. Для розроблюваного у даній роботі сайту створено веб-сервіс Simple Safe Browsing API [14], який покликаний спростити надто складну та ресурсоємну роботу з БД від Google. Ідентифікатор доступу до даного веб-сервісу також заноситься у конфігураційний файл config.php.

При реєстрації клієнти повинні надавати свої e-mail-адреси. Для того щоб надалі їх можна було використати, необхідно проводити додаткову перевірку значення поля, що заповнив користувач при реєстрації. Для

цього необхідно використовувати стандарт RFC 822 [15]. За цим стандартом було реалізовано клас Rfc822 створюваної моделі.

Об'єктний клас Advt має, аналогічно класу Client, метод Advt::update_data(), що заповнює усі поля одним запитом до бази даних. Також у ньому містяться типові гетери. Метод Advt::get_paid_advt() генерує кілька запитів до БД. Перший отримує інформацію про випадкову рекламну кампанію із усіх активних, рекламодавці яких мають додатний стан рахунка. Другий перевіряє, чи бачив в останні 10 хвилин відвідувач з даної адреси рекламу, за яку автор посилання отримав кошти. Якщо ні, він відмічається як такий, що бачив, у цей же момент перераховуються кошти з рахунка рекламодавця на рахунок автора. Впродовж наступних 10 хвилин даний відвідувач може переглядати рекламний банер скільки завгодно разів без запису у БД і безкоштовно для рекламодавців. Таким чином зорганізується захист від «накруток» показів (а отже, і заробітку) авторами.

Клас View працює над коректним відображенням інформації у веб-браузерах відвідувачів. Він містить у собі шаблонізовану структуру для «спілкування» з клієнтами. Шаблон знаходиться у вигляді HTML-документів з використанням PHP-вставок, у яких дозволено використовувати глобальні константи та змінні, що їх генерує клас View.

Клас Money відповідає за операції з реальними грошима та зв'язує розроблений онлайн-сервіс із платіжною системою WebMoney. Через цей клас клієнтам надається можливість поповнювати рахунок, виводити кошти та переглядати поточний стан власного внутрішнього рахунку в системі. Реалізовано захист від помилок з'єднання веб-серверу, на якому розташовано даний веб-сервіс, та веб-серверу платіжної системи. Для виведення стану рахунку враховуються комісії платіжної системи. При запиті коштів на виведення відбувається перевірка на заповнення поля з розрахунковим номером та пропонується його ввести, перш ніж замовити виплату.

Крім того, створено глобальну функцію _e(): вона отримує на вхід текстовий рядок англійською мовою та перекладає його на мову створеного веб-сервісу. Мова системи обирається у конфігураційному файлі config.php, на даний момент доступні: українська, російська та англійська мови.

3. Інтерфейс програми

Для встановлення розробленого проекту на власний сервер необхідно і достатньо виконати три кроки:

1) скопіювати діючі файли на сервер.

Сервер повинен працювати з увімкненими модулями PHP 5 та вище, MySQL 5 та вище. Бажано Apache версії 2 або вище на будь-якій операційній системі. Підтримка файла користувацьких налаштувань серверу .htaccess або відповідний доступ до прямих налаштувань серверу обов'язкові.

Копіювати необхідно файли index.php, .htaccess та повністю директорію kich_reduce/ з усіма вкладеними файлами та піддиректоріями. У результаті на сервері вони повинні бути розміщені в одній директорії.

2) відредагувати файл config.php згідно з підказками у коментарях до кожного рядка у ньому.

Якщо потрібен захист від «шкідливих» посилань, також доведеться зареєструватися на веб-сервісі Simple Safe Browsing API (посилання на який зазначено у коментарях до конфігураційного файла).

3) проінсталювати веб-сервіс.

Для цього достатньо зайти через веб-браузер на сторінку /kich_reduce/install.php. Це створить у заданій базі даних усі таблиці,

необхідні для роботи веб-сервісу. Якщо відобразиться повідомлення про успішну установку, бажано видалити `install.php` з веб-серверу та зареєструватися у системі першим користувачем (він і буде адміністратором).

Тепер можна відвідати новостворений сайт.

Оскільки тестова версія запущена на домені SHAP.TK, то відповідно на головній сторінці великими літерами відображається саме цей домен. Таке ім'я генерується динамічно відповідно до домену, на якому запущено сервер, тому для відображення актуальної інформації у цьому місці жодних додаткових дій не потрібно.

Сайт вже працює та може скорочувати посилання. Але для отримання користувачам додаткових можливостей бажано зареєструватися. Для цього слід натиснути у правому верхньому куті на посилання «Реєстрація», відкриється сторінка реєстрації.

Номер гаманця WebMoney зазначає не обов'язково. У разі, якщо необхідно буде зробити виведення коштів, система сама попросить про це і не дозволить замовити виведення без записування коректного номера гаманця.

Після реєстрації можна заходити у систему за посиланням «Вхід» з ційно створеними логіном та паролем.

Увійшовши у систему один раз, деякий час користувач залишається у ній без потреби повторення процедури входу. Це досягається завдяки зберіганню зашифрованої інформації в «cookies» (від англ. печиво, слово не перекладається) веб-браузера клієнта.

Авторизовані (або такі, що увійшли у систему) клієнти мають можливість переглядати власні посилання, рекламні кампанії та створювати нові на вкладці «Моя сторінка».

Після переходу за скороченим посиланням відвідувач потрапляє не на сам сайт, за яким веде довге посилання, а лише його відображення у фреймі проекту. У правому верхньому куті вікна відображається банер з інформацією, отриманою псевдовипадково серед активних рекламних кампаній. Цей банер можна закрити.

Сторінка, що відкривається за скороченим посиланням, генерується оптимізованою для пошукових систем - посилання на банері містить у собі як назву, так і опис (title) і є індексованим [16]. Цей аспект позитивно впливає на позиції сайту рекламодавця в пошукових системах.

При розміщенні посилань на потенційно небезпечні ресурси згідно з Simple Safe Browsing API таке посилання помічається у локальній базі даних як «блоковане». При переході відвідувачів за таким скороченим посиланням вони отримають попередження про можливі наслідки. Єдиний спосіб потрапити за блокованим посиланням – скопіювати його з поля повідомлень та вставити у рядок URL веб-броузера.

Якщо коротке посилання створюється для посилання на графічне зображення, то воно відображається вбудованим у сторінку веб-сайту розробленого проекту.

При натисненні на суму балансу рахунку відкривається сторінка керування балансом «Мій рахунок». При поповненні рахунку рекламодавець обирає суму та спосіб оплати. Після натискання на кнопку «Сплатити» він потрапляє на сайт мерчанту WebMoney, де здійснює авторизацію та передачу коштів на рахунок системи.

Перший зареєстрований користувач системи – адміністратор за замовчуванням. Він має додаткові привілеї та доступ до панелі керування системою.

Через панель керування адміністратор може переглянути усі існуючі скорочені посилання, рекламні кампанії, створені рекламодавцями, заявки на виведення коштів, може виконати такі заявки, перевірити загальну суму виведених коштів за весь час існування системи та

поточний фонд системи. До фонду належить сума коштів усіх користувачів, що може бути запитана до виведення в будь-який момент. Тобто фонд – це та кількість грошей, яка бажана для зберігання на рахунок власника системи – на екстрений випадок, якщо усі користувачі одночасно замовлять виплату.

Таблиця замовлень виплат та поле фонду системи виводить вже підраховані суми з урахуванням власне комісії створеного веб-сервісу та комісії платіжної системи.

ВИСНОВКИ

Робота присвячена проблемі розміщення інформаційних матеріалів з використанням сторонніх веб-ресурсів. Завдяки такому веб-сервісу з'явилася можливість отримувати грошові винагороди за розповсюдження посилань на сайти, до яких немає безпосереднього адміністраторського доступу.

Розроблений web 2.0 сервіс характеризується об'єднанням позитивних якостей існуючих сервісів скорочення посилань і позбавленням від більшості негативних. Використано алгоритм, здатний значно зменшити довжину будь-яких посилань в Інтернеті та допомогти в більш легкому передаванні таких посилань по телефону, друкованими виданнями та соціальними мережами.

Використано базу даних «безпечного перегляду» від Google для забезпечення додаткової безпеки користування веб-сервісом.

SUMMARY

ONLINE SERVICE FOR SHORTENING THE LINKS TO EXTERNAL RESOURCES AND PLACEMENT OF INFORMATIONAL MATERIALS BY THEM

*S. P. Pashko, V. K. Obodyak,
Sumy State University, Sumy*

Existent services of reduction of long references and placing of selling aids are analyzed. It is shown that the known decisions have certain defects. Online service was designed for shortening the links to external resources and placement of informational materials by them. Software implementation is executed on the PHP programming language using the RDBMS MySQL, Google Safe Browsing, WebMoney Transfer.

Key words: *safety, links, advertising, service, frame, css, google, html, mysql, php, safe browsing, url, web 2.0, webmoney.*

СПИСОК ЛІТЕРАТУРИ

1. Эксперт социальных сетей, лаборатория Сарафанное Радио. Веб 1.0 против Веб 2.0 [электронный ресурс]/ Режим доступа: <http://www.sarafannoeradio.org/analitika/428-web-1-0-vs-web-2-0.html>
2. Twitter, Inc. Twitter is a real-time information network powered by people all around the world that lets you share and discover what's happening now. – <http://twitter.com/about>
3. Кирилл Николаев. Зачем нужны сервисы сокращения адресов? [электронный ресурс]/ Режим доступа: <http://habrahabr.ru/blogs/twitter/63301/>
4. Пашко С. П. Маркетинговое решение проблемы длинных URL-адресов в WEB 2.0 // Материалы научно-практической конференции преподавателей, аспирантов и студентов Сумского НАУ (20-29 апреля 2011 г.): 3 т. – Сумы, 2011. – С. 61.
5. URL shortening services [электронный ресурс]/ Режим доступа: <http://code.google.com/p/shortenurl/wiki/URLShorteningServices>
6. Кирилл Николаев. реер.us - ссылки на закрытые страницы [электронный ресурс]/ Режим доступа: – http://habrahabr.ru/blogs/i_am_advertising/68079/
7. Пашко С. П. Онлайн-сервіс розміщення інформаційних матеріалів з використанням сторонніх ресурсів [Текст] / С. П. Пашко, В. К. Ободяк // Інформатика, математика, механіка : матеріали та програма науково-технічної конференції. Суми, 18-22 квітня 2011 р. / відп. за вип. С.І. Проценко. — Суми : СумДУ, 2011. — С. 36.
8. Пашко С. П. Захист відвідувачів веб-сайту від небезпечних ресурсів / С. П. Пашко // Матеріали науково-практичної конференції викладачів, аспірантів та студентів Сумського НАУ (20-29 квітня 2011 р.): в 3 т. – Суми, 2011. – Т.1. – С. 60

9. Википедия – свободная энциклопедия. Объектно-ориентированное программирование [электронный ресурс] – Режим доступа: http://ru.wikipedia.org/wiki/Объектно-ориентированное_программирование
10. Dennis Pallett. Simple MVC on PHP 5. (Переклад / Vyrus. Создание простой MVC-системы на PHP 5) – <http://habrahabr.ru/blogs/php/31270/>
11. Руководство по PHP. mysql_real_escape_string – <http://www.php.ru/manual/function.mysql-real-escape-string.html>
12. Воа. SQL Injection для "совсем новичков" [электронный ресурс]/ Режим доступа: - <https://forum.antichat.ru/thread217895-MySQL+Injection.html>
13. PHP manual: base_convert / Mr.Fips. custombase_convert_big [электронный ресурс]/ Режим доступа: - <http://www.php.net/manual/en/function.base-convert.php#55204>
14. Пашко С. П. Спрощення інтерфейсу роботи з «Google Safe Browsing API» [Текст] / С. П. Пашко, В. К. Ободяк // Інформатика, математика, механіка : матеріали та програма науково-технічної конференції. Суми, 18-22 квітня 2011 р. / відп. за вип. С. І. Проценко. — Суми : СумДУ, 2011. – С. 46.
15. RFC 822 – Standard for the Format of ARPA Internet Text Messages [електронний ресурс]/ Режим доступу: - <http://www.faqs.org/rfcs/rfc822.html>
16. Пашко С. П. Методи пошукової оптимізації / С. П. Пашко // Матеріали науково-практичної конференції викладачів, аспірантів та студентів Сумського НАУ (20-29 квітня 2010 р.): в 3 т. – Суми, 2010. – Т. 1. – С. 38

Надійшла до редакції 5 січня 2012 р.