

Інформаційний Портал для Обміну Контентом між Соціальними Мережами

О. Гладка

кафедра комп'ютерних наук
Національний університет водного господарства та природокористування
м. Рівне, Україна
o.m.hladka@nuwm.edu.ua

Я. Бабич

кафедра комп'ютерних наук
Національний університет водного господарства та природокористування
м. Рівне, Україна

Information Portal for Exchange Content between Social Networks

O. Hladka

Department of Computer Science
National University of Water and Environmental Engineering
Rivne, Ukraine
o.m.hladka@nuwm.edu.ua

Ya. Babych

Department of Computer Science
National University of Water and Environmental Engineering
Rivne, Ukraine

Анотація—Розроблено Web-сервіс з організації інформаційного порталу для автоматизованого обміну даними між різними соціальними мережами. Розроблений Web-ресурс забезпечує централізоване управління процесом розповсюдження контенту у соціальних мережах та максимально спрощує його для користувача.

Abstract—The Web-service for the organization of an information portal for automated data exchange between different social networks has been developed. The developed web-resource provides centralized management of the process of distribution of content in social networks and maximally simplifies it for the user.

We reviewed the main features of interaction with social networks, analyzed the existing software solutions that provide automated content management. The main advantages and disadvantages of these systems were determined and we selected the main tendencies in the development of the functional, which ensure the originality of its own software solution.

The actual technologies of the development are investigated and the most convenient for deployment of the project based on the client-server model are selected, in particular: Python, Django, SQLite, HTML 5, CSS 3, JS, jQuery, Bootstrap. In the development process, modern design tools, such as the integrated development environment of PyCharm Educational Edition and the Git version control system with a remote repository in the GitLab repository system, were used.

Ключові слова—Web-сервіс; соціальні мережі; обмін контентом; мова програмування Python

Keywords—Web-service; social networks; exchange content; programming language Python

I. ВСТУП

Очевидним є той факт, що Інтернет вже давно є не просто засобом обміну інформацією, а окремим самостійним ринком зі своєю інфраструктурою. Сьогодні кожен може за лічені секунди залишити свій слід у нескінченному цифровому потоці, і основна проблема будь-якого інформаційного ресурсу – не відкрити доступ до контенту, а донести його до максимальної кількості людей. На допомогу у цьому приходять соціальні мережі, які стали однією з ключових ланок цієї системи. Розповсюдження контенту у популярних соціальних мережах є стандартною маркетинговою стратегією, спрямованою на популяризацію продуктів [1].

Залучення трафіку або уваги до бренду чи продукту через соціальні платформи – Social Media Marketing (SMM) (англ. Маркетинг в соціальних мережах) або, якщо більш точно, просування в соціальних мережах (Social Media Promotion) – комплекс заходів щодо використання соціальних медіа в якості каналів для просування компаній і вирішення інших бізнес-завдань є на сьогодні чи не найбільш динамічним і перспективним напрямком розвитку сучасного бізнес-ринку [2]. При цьому акцент робиться на створенні контенту, який люди будуть поширювати через соціальні мережі самостійно, вже без участі організатора, оскільки вважається, що повідомлення, які передаються соціальними мережами, викликають більше довіри у потенційних споживачів товару чи послуги. Це узгоджується з рекомендаційною схемою поширення в соціальних медіа за рахунок соціальних зв'язків, які лежать в основі взаємодії [3].

Просування в соціальних мережах дозволяє точно впливати на цільову аудиторію, вибирати майданчики, де ця аудиторія представлена більшою мірою, і найбільш підходящі способи комунікації з нею, при цьому щонайменше зачіпаючи незацікавлених в цій рекламі людей. Важливо відзначити, що просування в соціальних мережах застосовується не тільки для товарів і послуг – дану технологію активно використовують також засоби масової інформації. Вони створюють свої облікові записи в соціальних мережах, розміщують свій контент і тим самим збирають передплатників – читачів свого продукту.

Зазвичай процес розповсюдження контенту у популярних соціальних мережах є рутинним і не раціональним. Враховуючи задачу, яка полягає у розповсюдженні однієї і тієї ж інформації на велику кількість інформаційних ресурсів виникає потреба в автоматизації та оптимізації даного процесу шляхом реалізації сервісу з автоматичного розповсюдження матеріалу, який забезпечить зручність роботи SMM-спеціаліста або користувача, що веде активну діяльність у соціальних мережах. Ця концепція має потенціал до розвитку у вигляді системи відслідковування статистики зворотнього зв'язку, системи збору та відправки повідомлень користувачам, публікування матеріалів у заданий час, підбору потенційних клієнтів тощо.

Описана вище проблема не є новою, а тому на ринку існує чимало професійних вирішень з досить потужним функціоналом. Проте, зазвичай, проаналізовані сервіси є комерційними з платним користуванням і не розраховані на українського користувача. У англійських (та й інших мовних) сервісах, як правило, відсутня підтримка української мови, а використання російських (російськомовних) сервісів (які, зазвичай, зрозумілі українськомовним користувачам), зокрема, таких як NovaPress, KUKU.io, SMMR тощо, є небажаним (чи навіть небезпечним) через можливість доступу до облікових даних зі сторони державних органів РФ.

II. ОПИС ПРОГРАМНОГО ПРОДУКТУ

Оскільки розроблюване програмне забезпечення не вимагає високої обчислювальної потужності для реалізації проекту було обрано web-платформу, що надає низку переваг, зокрема: можливість зберігати дані на сервері, що забезпечує їх захист від фізичної втрати девайсу користувачем; можливість користуватись сервісом з різних платформ (Windows, Ubuntu, MacOS, Android, IOS тощо); основні обчислення проводяться на стороні сервера, що забезпечує низькі апаратні вимоги до пристрою користувача; для використання сервісу не потрібна інсталяція програмного комплексу; можливість створення гнучкого інтерфейсу завдяки тенденції розвитку web-платформ в останні роки.

Принципова схема роботи web-сервісу наведена на рис. 1.



Рис. 1. Принципова схема роботи web-сервісу

Ця схема візуалізує сценарій, що реалізований програмним продуктом: користувач реєструється на сайті для того, щоб система змогла зберігати у базі даних матеріали з його соціальних мереж; користувачем надається доступ системі до аккаунтів соціальних мереж зі списку, що наданий системою; за допомогою відповідного інтерфейсу користувач створює запис, який хоче поширити; система відправляє запити до соціальних мереж з метою публікації та записує результати у базу даних.

Відповідно до поданої схеми розроблене програмне застосування задовольняє такі технічні вимоги: прив'язка аккаунтів соціальних мереж без використання логіна та пароля користувача web-сервісу; керуванням доступом (можливість відкликати дозвіл на використання особистих даних); створення записів; інтерфейс моніторингу відправлених записів; зручна навігація сайтом; крос-платформність; наявність засобів адміністрування; можливість одночасної роботи великої кількості (100 і більше) користувачів.

Надійне функціонування програмного комплексу забезпечене шляхом контролю коректності та повноти вхідних даних (всі дані, що вводяться користувачем, перевіряються на формальну коректність); надання можливості періодичного створення резервних копій інформаційної бази; ведення протоколів дій користувачів.

Захист інформації від несанкціонованого доступу забезпечується такими засобами як: формування різних рівнів доступу із переліком допустимих прав; авторизація користувачів; дані, необхідні для функціонування програмного комплексу, зберігаються на сервері програмного комплексу у зашифрованому вигляді.

III. АРХІТЕКТУРА СЕРВІСУ

Для реалізації проекту обрано класичну архітектуру «клієнт-сервер». Клієнтська програма (браузер) відправляє HTTP-запити до сервера, а він, в свою чергу, обробляє їх і у відповідності до параметрів запитів формує правильні запити до бази даних. Відповідь від бази даних також обробляється сервером і повертається повноцінний об'єкт, який трансформується у зручне подання програмним забезпеченням на стороні клієнта.

Модель даних (рис. 2) має таку структуру:

AbstractUser – абстрактна модель сутності «Користувач», що використовується для авторизації користувачів на сайті;

Credentials – відповідає за облікові дані користувачів у відповідних соціальних мережах;

Attachment – забезпечує зберігання зображень, містить одне поле формату “image”, в яке може бути додано зображення з файлової системи користувача;

PublishInfo – зберігає посилання та назву соціальної мережі, в який було опубліковано запис;

Record – модель запису, створеного користувачем.

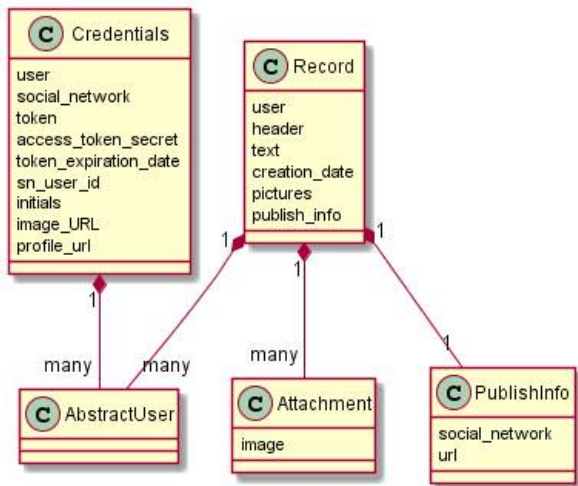


Рис. 2. Модель даних

IV. ТЕХНОЛОГІЇ РОЗРОБКИ СЕРВІСУ

Для реалізації програмного продукту використано інтегроване середовище розробки PyCharm Educational Edition та систему контролю версій Git[8] з віддаленим сховищем у системі репозиторіїв GitLab.

Серверна частина програмного застосування написана мовою Python[5], зокрема, з використанням веб-фреймворка Django[6] – безкоштовного, з відкритим вихідним кодом пакета веб-розробки, який забезпечує швидку і легку розробку безпечних та надійних сайтів. Django допомагає розробникам уникнути багатьох помилок у впровадженні загальної безпеки, створюючи основу для автоматичного захисту сайту. Зокрема, передбачається безпечний спосіб управління обліковими записами користувачів, що забезпечує уникнення поширених помилок, таких як зберігання інформації у файлах “cookies”, де вона є вразливою або зберігання паролів у “сирому” вигляді, замість використання хешу. Також, за замовчуванням, Django забезпечує захист від багатьох вразливостей, зокрема, SQL-ін'єкцій, міжсайтового скриптингу, підробки запитів та клікджекінгу (англ. Clickjacking – механізм обману користувачів Інтернету, при якому зловмисник може отримати доступ до конфіденційної інформації або навіть отримати доступ до комп'ютера користувача, заманивши

його на зовні нешкідливу сторінку чи запровадивши шкідливий код на безпечну сторінку)[7]. Схему обробки запитів системою Django наведено на рис. 3.

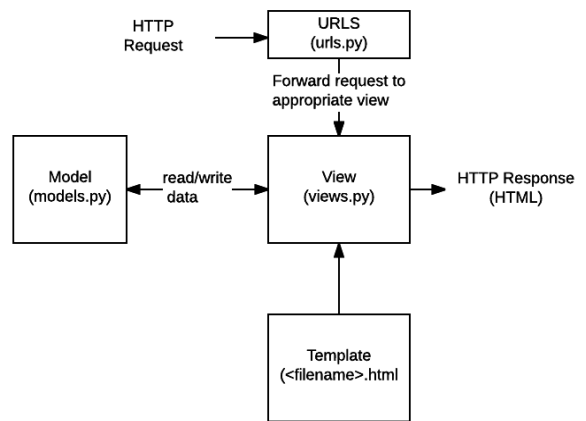


Рис. 3. Візуалізація обробки HTTP-запитів системою Django

Для проектування і реалізації бази даних використано бібліотеку SQLite, яка є вбудованою реляційною базою даних за замовчуванням при створенні проекту на базі Django. Сама бібліотека SQLite написана на C, але існує велика кількість прив'язок до інших мов програмування, в тому числі до Python. Оскільки ядро і інтерфейс реалізовані як єдине ціле, величезною перевагою SQLite є висока продуктивність – для більшості типових завдань невеликий і середній за обсягом додаток, побудований на SQLite, працює швидше, ніж при використанні MySQL, в 2-3 рази і ніж PostgreSQL – в 10-20 разів. Особливо актуальним є використання SQLite у випадках, коли в основному проводяться операції запису і зчитування даних. Однак при надзвичайно активному зверненні до даних SQLite працює повільніше від своїх конкурентів через вбудований механізм блокування файлів (тільки при модифікації даних) і необхідності перевірки типу полів для вибору способу сортування.

Для реалізації ПЗ на стороні сервера використано також NGINX – безкоштовний HTTP-сервер з відкритим вихідним кодом, а також проксі-сервер IMAP/POP3. NGINX відомий своєю високою продуктивністю, стабільністю, великим набором функцій, простою конфігурацією, а також низьким споживанням ресурсів. На відміну від традиційних серверів, NGINX не залежить від потоків для обробки запитів. Замість цього він використовує набагато більш масштабовану асинхронну архітектуру. Ця архітектура використовує невеликі, але що більш важливо, більш передбачувані обсяги пам'яті.

При створенні програмного забезпечення на стороні клієнта (браузерної частини сервісу) використано набір інструментів для створення сайтів і веб-додатків Bootstrap 3, що включає в себе HTML- і CSS-шаблони оформлення для типографіки, веб-форм, кнопок, міток, блоків навігації та інших компонентів веб-інтерфейсу, а також JavaScript-розширення.

Для забезпечення інтерактивності веб-сервісу використано jQuery – JavaScript-бібліотеку з відкритим

вихідним кодом. jQuery є вільним програмним забезпеченням під ліцензією MIT. Синтаксис jQuery розроблено таким чином, щоб зробити орієнтування у навігації зручнішим завдяки вибору елементів DOM, створенню анімації, обробки подій, і розробки AJAX-застосунків.

V. ОПИС РОБОТИ З ПРОГРАМНИМ ЗАБЕЗПЕЧЕННЯМ

Головна сторінка web-сервісу є максимально простою і не перенасиченою зайвою інформацією, а тому складається лише з трьох частин: головного меню, інформаційного слайдера та блоку з інструкцією.

При спробі доступу до будь-якого з розділів сайту, користувачу пропонується авторизуватися чи зареєструватися у випадку відсутності облікового запису.

Після успішної авторизації сайт перенаправляє користувача у розділ «Профіль», де можна змінити основну інформацію про себе та зв'язати аккаунти у соціальних мережах з обліковим записом сайту. Для тестування сервісу використовувалися три популярні соцмережі: Facebook, Google+ та Twitter. Також реалізовано можливість зміни пароля. Цей розділ є модульним і його відображення можливе після натискання відповідної кнопки «Змінити пароль».

За умови, що користувач надав доступ до керування своїми записами програмі, на сайті буде відображено інформацію про аккаунт. Дана інформація включає в себе ім'я користувача у соцмережі, id – унікальний ідентифікатор, який також є посиланням на профіль у відповідній соціальній мережі та час дії доступу, який вказує на дату закінчення доступу з наявним ключем. Після закінчення терміну дії ключа доступу сервіс відправить відповідний запит до прикладного інтерфейсу на оновлення ключа. Якщо доступ ще не було надано, користувач буде мати можливість прив'язати аккаунт за допомогою відповідного посилання «Активувати». Після переходу за посиланням, сервіс перенаправить користувача на вікно авторизації відповідної соціальної мережі, де буде вказано, які саме рівні доступу будуть доступними розробленому додатку. Зазвичай, для повноцінної роботи сервісу потрібні наступні дозволи: доступ до загальної інформації користувача; можливість публікації від імені користувача; можливість зчитувати наявні записи.

Після активації потрібних соцмереж користувач може приступити до створення запису у розділі «Створити запис». Далі, після збереження запису, управління перейде до сервісу і контент буде відправлено відповідно до активованих соціальних мереж. Після цього слідує перенаправлення користувача у розділ «Мої записи», де відображається інформація про запис та соціальні мережі, в яких його було опубліковано. Мережі, які не приймали участі у поширенні контенту, будуть мати не активний статус і позначені сірим кольором. Шляхом клацання на одній із активних іконок можна відкрити у новій вкладці вікно з записом у відповідній соцмережі.

Крім основного функціоналу на сайті є панель адміністрування, за допомогою якої можна керувати усіма записами сайту, а також створювати групи користувачів з різними рівнями доступу. Це дозволяє проводити гнучкі маніпуляції з існуючою базою, не змінюючи код проекту.

ВИСНОВКИ

У роботі розглянуто основні особливості взаємодії з соціальними мережами, проаналізовано існуючі програмні рішення, які забезпечують автоматизоване керування контентом. Визначено основні переваги та недоліки даних систем, обрано основні тенденційні напрями у розробці функціоналу, що забезпечують оригінальність власного програмного рішення.

Результатом роботи є web-сервіс для автоматичного розповсюдження контенту між соціальними мережами, що реалізує запланований функціонал і відповідає сучасним стандартам сайтобудування: має дружній до користувача інтерфейс, адаптивність та надійне зберігання даних.

Досліджено актуальні технології розробки та обрано найзручніші для розгортання проекту на базі моделі «клієнт-сервер», зокрема: Python, Django, SQLite, HTML 5, CSS 3, JS, jQuery, Bootstrap. В процесі розробки використано сучасні засоби проектування, зокрема інтегроване середовище розробки PyCharm Educational Edition та систему контролю версій Git з віддаленим сховищем у системі репозиторіїв GitLab.

Структура web-сервісу спроектована таким чином, щоб його можна було легко розширювати. Передбачено прикладний інтерфейс (API) для зручної роботи з даними сервера, що в свою чергу дає можливість для портування сервісу на мобільні платформи.

ЛІТЕРАТУРА REFERENCES

- [1] О.М. Степанова, Розробка комунікаційної стратегії із застосуванням аналізу даних соціальних медіа / О.М. Степанова, А.А. Волков // Інформаційні технології та комп'ютерне моделювання: матеріали статей Міжнародної науково-практичної конференції. – Івано-Франківськ, 2016. – С. 53-57.
- [2] G.Gregory, M. Karavdic and Sh. Zou, "The Effects of E-Commerce Drivers on Export Marketing Strategy", Journal of International Marketing, Vol. 15, No. 2, pp. 30-57, Jan. 2007.
- [3] C. Trattner, F. Kappe, "Social Stream Marketing on Facebook: A Case Study", International Journal of Social and Humanistic Computing (IJSHC), 2012.
- [4] К.В. Молодецька, Соціальні інтернет-сервіси як інструмент масової комунікації / К.В. Молодецька // Інформаційні технології та комп'ютерне моделювання: матеріали статей Міжнародної науково-практичної конференції. – Івано-Франківськ, 2016. – С. 60-61.
- [5] М. Лутц, Изучаем Python, 4-е издание. – Пер. з англ. – СПб.: Символ-Плюс, 2011. – 1280 с.
- [6] Дж. Форсьє, П. Біссек, У. Чан, Django. Разработка веб-приложений на Python – Пер. з англ. – СПб.: Символ-Плюс, 2010. – 456с.
- [7] «WebNotes», замітки веб-розробника. / [Електронний ресурс]. – Режим доступу: <http://webnotes.by/docs/sql/259>
- [8] Система контролю версій «Git». / [Електронний ресурс]. – Режим доступу: <https://git-scm.com/book/ru/v1/>