

ВИКОРИСТАННЯ МОВИ ПРОГРАМУВАННЯ PHP 5 ДЛЯ СТВОРЕННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ІНТЕРНЕТ-МАГАЗИНІВ

В статті наведені результати дослідження методів створення і розробки програмного забезпечення Інтернет-магазинів з використанням мови програмування php 5.

Постановка проблеми. Актуальність створення і розробки програмного забезпечення для мережі Інтернет викликаний глобальним розвитком ІТ-технологій за останні 30 років.

Активний розвиток торгівлі в глобальній мережі Інтернет став можливим завдяки створенню і розробці програмного забезпечення для Інтернет-магазинів. Інтернет-магазин (англ. *Internet shop*) — сайт в глобальній мережі Інтернет, де відбувається прямий продаж товарів споживачеві. При цьому розміщення споживацької інформації, замовлення і сама покупка відбуваються в глобальній мережі – на сайті Інтернет-магазину. Це передбачає грошові розрахунки. Для безпечного використання сайту передбачені надійні протоколи — https, та інші системи безпеки. Для того, щоб Інтернет-магазин видавав інформацію залежно від запитів, надавав можливості пошуку тощо — на сервер встановлюється підтримка скриптів (наприклад PHP, Perl) [1].

PHP (англ. Hypertext Preprocessor — PHP: гіпертекстовий препроцесор), попередня назва: Personal Home Page Tools — скриптова мова програмування, була створена для генерації HTML-сторінок на стороні веб-сервера. PHP є однією з найпоширеніших мов, що використовуються у сфері веб-розробок (разом із Java, .NET, Perl, Python, Ruby). PHP підтримується переважною більшістю хостинг-провайдерів. PHP — проект відкритого програмного забезпечення.

Історія PHP починається з 1995 року, коли Рasmus Лердорф (англ. Rasmus Lerdorf) створив мовою Perl простий аналізатор, що аналізував відвідування користувачами його резюме на веб-сайті. Потім Лердорф назвав своє творіння «Особисті інструменти домашньої сторінки» (англ. Personal Home Page Tools) версія 1 і виставив для вільного завантаження. З цієї миті почався небувалий зліт популярності PHP. Для реалізації нових доповнень Рasmus створює нову версію пакета, тепер уже написану на С. Отриманий таким чином інструмент набуває робочої назви PHP/FI Персональна Домашня сторінка / Інтерпретатор Форм (англ. Personal Home Page / Forms Interpreter), надалі він також буде відомий під назвою PHP 2. Ця версія вже більшою мірою схожа на сьогоденний PHP. Вона мала синтаксис і спосіб іменування змінних в стилі мови Perl, можливість вбудовування PHP операторів в html-код сторінки, автоматичну інтерпретацію форм, інтеграцію з базами даних. При цьому все працювало досить швидко, оскільки PHP компілювалось до веб-серверу Apache. [2] До 1997 року PHP використовувався вже на 50,000 доменах (не більше 1 % всіх веб-серверів). У 1997 році до проекту PHP підключилися Зев Сураскі (англ. Zeev Suraski) і Енді Гутманс (англ. Andi Gutmans).

В 1998 році виходить нова версія під назвою PHP 3, при цьому головною особливістю PHP 3 була можливість розширення ядра, що привернуло до роботи над PHP безліч сторонніх розробників, що створюють спеціалізовані модулі. Їх наявність дала PHP можливість працювати з величезною кількістю баз даних, протоколів, підтримувати велике число API. До кінця 1998 року кількість користувачів PHP перевищила за 100000, а PHP був уже встановлений на не менше ніж 10 % серверах Інтернету. У той ж час значному поширенню даної мови сприяли публікації в електронній пресі та вихід книжок для вивчення PHP. Відразу ж після виходу PHP 3, Енді Гутманс і Зев Сураскі почали переробку ядра PHP. В першу чергу належало вирішити проблему підвищення продуктивності. Новий продукт, названий Zend Engine (від імен творців: Zeev і Andi), успішно справлявся з поставленим завданням і був реалізований в 1999 році. Основними реалізованими ідеями є можливість компіляції сценарію у виконуваний модуль, за рахунок чого продуктивність можна було підняти на порядок [3].

PHP 4 вийшов у 2000 році. На додаток до збільшення продуктивності, PHP 4 мав нові можливості щодо підтримки сесій, буферизацію виводу, безпечні способи обробки інформації,

що вводиться користувачем, і нові мовні конструкції. З виходом 4 версії PHP став використовуватися вже на більш ніж 20 % доменів Інтернету. За час з 2000 по 2004 роки продовжувалися активні роботи з покращення 4 версії, але майже відразу PHP Group приступила до продумування можливостей нової версії. В першу чергу було вирішено підсилити об'єктні можливості мови, що дозволяло використовувати його для реалізації масштабних проєктів. Роботи зі створення версії 5 велися тривалий час, в них брало участь рекордна кількість фахівців, зокрема Стерлінг Хьюз (англ. Sterling Huse) і Маркус Бергера (англ. Marcus Bergera).

У липні 2004 року виходить офіційний реліз PHP 5. В першу чергу, як і планувалося, було перероблено весь механізм роботи з об'єктами. PHP 5 володіє прекрасним потенціалом реалізації об'єктного програмування. Окрім цього, PHP збагатився рядом цінних розширень для роботи з XML, різними джерелами даних, генерації графіки, що робить його незамінним для розробки програмного забезпечення для мережі Інтернет [4].

Аналіз публікацій і постановка задачі досліджень. Особливості програмування для Інтернет-проєктів характеризуються посиленням вимог до конкурентоспроможності та відкритістю інформаційного простору. Розвиток конкурентних відносин в глобальній мережі Інтернет характеризується створенням більш жорстких умов, що посилюють необхідність застосування нових форм просування товару і послуг із використанням сучасних інформаційних технологій. Використання найновіших мов програмування, таких як PHP 5, зумовлюють необхідність застосування нових форм і засобів у створенні та розробці програмних продуктів.

Проблемами програмного забезпечення для глобальної мережі Інтернет та управлінням цим процесом у своїх працях велику увагу приділяють такі зарубіжні вчені, як Р.Лердорф, З.Сураскі і Е.Гутманс, П.Друкер, Г.Стьюарт, Дж.Болт, А.Стиркланд, Е.Кофлан, Д.Ланкастер, Д.Джоббер, С.Хьюз і М.Бергера, Д.Бізос та ін. Вагомий внесок у дослідження проблем діяльності в мережі Інтернет зробили такі вітчизняні вчені: Л.Балабанова, А.Балабаниць, О.Кузьмін, Є.Ромат, О.Майборода, Н.Тарнавська, Р.Ларіна. В дослідженнях українських науковців достатньо глибоко досліджено структуру та основні засади використання маркетингових комунікацій для інформаційних технологій.

Метою статті є висвітлення науково-практичних аспектів використання нових напрямів розробки і створення програмного забезпечення на мові програмування PHP 5 для створення Інтернет-магазинів у глобальній мережі Інтернет та проведення порівняльного аналізу ефективності використання PHP 5 та PHP 4.

Викладення основного матеріалу. Створення і розробка програмного забезпечення для Інтернет-магазину. Існує декілька основних методів створення Інтернет-магазинів – шляхом створення спеціалізованого програмного забезпечення, або методом конструктора на безкоштовних спеціалізованих сайтах. Програмне забезпечення для Інтернет-магазину має певну функціональність, тому що вимагає наявності складної системи управління. До основних елементів програмного забезпечення в системі управління Інтернет-магазинами належить: каталог, кошик товарів, процедура оформлення замовлення, купівельний розділ, обробка замовлення, управління доставкою, інтеграція з різними платіжними системами, а також із системами управління товарами (наприклад, 1С).

Однією з найбільш поширеніших мов програмування для створення серверної частини сайтів (в тому числі Інтернет-магазинів) є PHP [5]. Основною перевагою PHP є відносна простота та безкоштовність. 50–60 % Інтернет магазинів на пост радянському просторі та 30–40 % європейських магазинів створені саме за допомогою PHP.

Невід'ємною частиною будь-якого Інтернет-магазину є база даних. Формат бази даних може бути будь-яким. Але як правило, при програмуванні на php використовується MySQL (є безкоштовним), а при використанні .NET технологій використовується MS SQL, який має доволі потужні засоби для полегшення створення і редагування даних – наприклад, SQLServerManagementStudio. Варто зазначити, що майже всі продукти Microsoft, які необхідно використовувати для створення Інтернет-магазину, мають безкоштовні Express версії, які мають певні обмеження – наприклад на максимальний розмір бази даних.

Крім основної клієнтської частини створюється так званий BackOffice – це окремий сайт, з обмеженим доступом, де адміністратор може змінювати налаштування магазину (додавати та видаляти продукти, користувачів, проглядати статистику замовлень тощо). Такими є основні кроки створення програмного забезпечення для Інтернет-магазинів.

Для створення Інтернет-магазину методом конструктора необхідно зареєструватись на певному умовно-безкоштовному порталі в глобальній мережі Інтернет, що надає такі послуги [5]. Після реєстрації та її підтвердження вибирається макет із наявних і виконується інформаційне наповнення макету необхідною інформацією. Наступним кроком є початок роботи вже готового Інтернет-магазину та прийом замовлень в режимі он-лайн. Такий метод створення окрім певної простоти, має свої недоліки – неможливість зміни певних опцій, використання лише певного наявного забезпечення та дизайну і повна залежність від умов роботи порталу, на якому створено даний Інтернет-магазин [6].

Розглянемо особливості розробки програмного забезпечення на мові програмування PHP та відмінності версій PHP 5 від попередньої PHP 4.

Основні відмінності версій PHP 4 і PHP 5. У версії PHP 4 основним недоліком було нераціональне використання пам'яті, оскільки об'єкти оброблялися як примітивні типи (змінні, списки), тобто через копіювання [7]. У новій об'єктній моделі, яка використовується в PHP 5 – Zend Engine 2.0 використовується технологія об'єктних дескрипторів. Тепер об'єкти доступні лише через дескриптори, що дозволяє економити пам'ять і отримувати повний контроль над створенням і дублюванням об'єктів. Zend Engine 2.0 використовує новий менеджер пам'яті. Дві головних його переваги: покращена підтримка багатопоточних середовищ (розподілені блоки не мають потреби у взаємних виняткових блокуваннях) і те, що після кожного запиту набагато ефективніше звільняються розподілені блоки пам'яті.

Зміни торкнулися продуктивності вбудованих функцій PHP. За різними оцінками, збільшення продуктивності досягає 40 %. У конфігураційному файлі PHP – php.ini тепер доступні нові директиви:

- session.hash_function – вибір функції хешування сесій (SHA1 або MD5);
- session.hash_bits_per_character – визначає, яке число біт в символі використовується при хешуванні сесій (від 4 до 6);
- register_long_arrays – вмикає/вимикає реєстрацію довгих масивів;
- mail.force_extra_parameters – додаткові поштові параметри;
- zend.zel_compatibility_mode – включення сумісності з Zend Engine 1 (PHP4).

Підтримка потоків зазнала змін. Ось деякі з них:

- більш висока продуктивність функцій readfile(), fpassthru() та деяких інших стандартних функцій для роботи з потоками;
- нова функція stream_socket_client() – більш потужний аналог мережевої функції fsocketopen();
- нова функція stream_socket_accept() – приймає з'єднання клієнта;
- нова функція stream_socket_get_name() – отримує локальне або віддалене ім'я сонета;
- нова функція stream_copy_to_stream() – дозволяє копіювати потік в потік;
- нова функція stream_get_line() – дозволяє читати інформацію з потоку до певного символу або рядка;
- підтримка динамічного завантаження OpenSSL;
- можливість роботи функцій opendir(), stat(), unlink() з FTP – з'єднанням.

PHP5 містить досить велику кількість нових функцій:

1. Функції для роботи з масивами (Array).
2. Нові функції для роботи з базою даних InterBase.
3. Функції розширення iconv.
4. Функції для роботи з потоками (Streams).
5. Функції, які використовуються для роботи з датами і часом:
 - idate() – форматує місцевий час/дату як ціле число;
 - date_sunset() – час заходу даного дня залежно від місця розташування;
 - date_sunrise() – час сходу даного дня залежно від місця розташування;
 - time_nanosleep() – затримка в секундах і мілі-(чи нано)секундах.
6. Функції для роботи з рядками (Strings):
 - str_split() – перетворює рядок у масив;
 - strpbrk() – виконує пошук рядка з будь-яким набором символів;
 - substr_compare() – виконує реєстро-незалежне порівняння двох рядків, починаючи зі зміщення і до довжини рядка.
7. Інші нові функції:
 - convert_uudecode() – декодує uuencoded – закодований рядок;
 - convert_uuencode() – кодує за типом uuencoded;
 - curl_copy_handle() – копіює дескриптор cURL;
 - dba_key_split() – розбиває ключ на рядкове представлення у вигляді масиву;
 - dbase_get_header_info() – отримує інформацію про заголовок бази даних DBASE;
 - dbx_fetch_row() – проводить вибірку рядків з query_result, який має прапор DBX_RESULT_UNBUFFERED;
 - fbsql_set_password() – змінює пароль поточного користувача бази даних fbsql;
 - file_put_contents() – записує рядок у файл;
 - ftp_alloc() – розподіляє простір для файлу, який буде переданий через FTP;
 - get_declared_interfaces() – повертає масив всіх оголошених інтерфейсів;
 - get_headers() – отримує всі заголовки, послані сервером у відповідь на запит HTTP;
 - headers_list() – повертає список відісланих заголовків;
 - http_build_query() – генерує URL-кодований рядок запиту;
 - image_type_to_extension() – повертає розширення файлу графічного типу, який був повернутий графічними функціями: getimagesize(), exif_read_data(), exif_thumbnail(), exif_imagetype();

- `imagefilter()` – функція, що дозволяє застосовувати до графічних зображень різні фільтри;
- `imap_getacl()` – повертає ACL для даної поштової скриньки;
- `ldap_sasl_bind()` – зв'язує з каталогом LDAP, використовуючи SASL;
- `mb_list_encodings()` – повертає масив всіх підтримуваних кодувань;
- `pcntl_getpriority()` – повертає пріоритет будь-якого процесу;
- `pcntl_wait()` – очікує повернення стану поточної дочірньої гілки, визначеної системним викликом `waitpid()`;
- `pg_version()` – повертає масив з версіями клієнта, протоколу і сервера (за наявності);
- `php_check_syntax()` – здійснює перевірку синтаксису зазначеного PHP скрипта;
- `php_strip_whitespace()` – повертає вихідний код з розділеними коментарями та міжстроковим простором;
- `proc_nice()` – змінює пріоритет поточного процесу;
- `pspell_config_data_dir()` – змінює місце розташування файлів лінгвістичної інформації;
- `pspell_config_dict_dir()` – змінює місце розташування основного списку слів;
- `setrawcookie()` – посилає cookies без кодування URL-а;
- `snmp_read_mib()` – читає і застосовує парсинг MIB файлу в активне MIB дерево;
- `sqlite_fetch_column_types()` – повертає масив типів стовпця конкретної таблиці.

В PHP 5 використовуються нові розширення:

- розширення Tidy: PHP 5 містить підтримку дуже корисної бібліотеки Tidy (<http://tidy.sf.net/>). Вона дозволяє розробникам PHP розбирати, діагностувати, чистити і відновлювати документи HTML. Розширення Tidy підтримує як функціональний, так і об'єктно-орієнтований інтерфейс, і її API використовує механізм винятків PHP 5;
- розширення Perl: хоча й не вміщене за умовчанням в PHP 5, розширення Perl дозволяє викликати Perl-скрипти, використовувати об'єкти Perl і використовувати інші функціональні можливості Perl прямо з коду PHP;
- розширення SQLite: підтримка SQLite. Спочатку була введена в PHP 4.3.x. Це вбудована бібліотека SQL, яка не вимагає SQL сервера і дуже підходить для додатків, які не вимагають масштабованих SQL-серверів, або якщо ви розгортаєте свій додаток у провайдера, який не надає доступ до SQL-сервера. Всупереч назві, SQLite дуже багата можливостями і підтримує транзакції, вкладені вибірки, представлення (view) і великі DB-файли. Тут це згадано, як можливість PHP 5, тому що підтримка SQLite була введена досить у пізніх релізах PHP 4, і оскільки PHP 5 надає новий об'єктно-орієнтований інтерфейс і підтримує ітератори;
- нове розширення MySQLi (вдосконалена MySQL): у PHP 5 для MySQL AB було написано нове розширення MySQL, яке дозволяє повністю використовувати переваги нових функціональних можливостей в MySQL 4.1 і більш нових версій. На противагу старому розширенню MySQL нове дає можливість використовувати обидва інтерфейси: функціональний та об'єктно-орієнтований. Отже тепер є вибір чому віддати перевагу. Нові можливості, підтримувані цим розширенням, містять контроль транзакцій, підтримку реплікацій, SSL тощо.

Після змін, що стосуються самої мови, зміни в роботі з XML в PHP 5 є, ймовірно, найсуттєвішими. Розширення функціональних можливостей XML в PHP 5 роблять мову повністю рівноправною з іншими, що використовуються в мережі. Підтримка XML в PHP 4 базувалася на різноманітних бібліотеках XML. Підтримка SAX здійснювалася старою бібліотекою Expat, для використання XSLT необхідна була бібліотека Sablotron, для DOM використовувалася більш потужна libxml2 – бібліотека проекту GNOME. Використання різноманітних сторонніх бібліотек не робило PHP 4 краще за інші мови, коли справа стосувалася підтримки XML. Супровід бібліотек розширень був слабким, нові стандарти XML не завжди підтримувалися, продуктивність була не настільки добра, як була б можлива, взаємодії між різними розширеннями XML не існувало.

У PHP 5 усі розширення XML були переписані, щоб використовувати відмінний набір інструментів, що надається libxml2 (<http://www.xmlsoft.org/>). Ця бібліотека має багато можливостей, відмінно супроводжує й ефективно реалізовує стандарти XML, що надає передові можливості технології XML в PHP. Усі згадані вище розширення (SAX, DOM і XSLT) тепер використовують libxml2, включаючи нові додаткові розширення – SimpleXML і SOAP.

Нова реалізація SAX переведена від використання Expat до libxml2. Хоча нове розширення повинно бути сумісно, можуть бути деякі тонкі відмінності. Розробники, які хочуть продовжувати працювати з бібліотекою Expat, можуть це зробити, конфігуруючи і збираючи PHP відповідним чином.

Підтримка DOM в PHP 4 була також заснована на бібліотеці libxml2, вона рясніла помилками, грішила витоками пам'яті, і API в багатьох випадках не відповідав рекомендаціям W3C. Розширення DOM було повністю переписано для PHP 5. Мало того, що розширення було докорінно переписано, тепер воно відповідає рекомендаціям W3C. Наприклад, назви функцій тепер використовують нотацію `studlyCaps` (присвоювання імен зі змішаним вживанням великих і малих літер) як наказано за стандартом

W3C, полегшуючи для вас застосування прочитаного в документації W3C у PHP. Крім того, розширення DOM тепер підтримує три види схем для валідації XML документів – DTD, XML Schema та RelaxNG. В результаті цих змін код, написаний для PHP 4 з використанням DOM, не завжди буде виконуватися в PHP 5. Однак просте коректування назв функцій до нового стандарту в більшості випадків вирішує проблему.

У PHP 4, було два розширення, які підтримували XSL-перетворення. Перше використовувало Sablotron, а друге – підтримку XSLT в розширенні DOM. У PHP 5 нове розширення XSL було засновано на використанні libxml2. Так, в PHP 5 XSL-перетворення не приймає таблицю стилів XSLT як параметр, але залежить від розширення DOM, щоб завантажити її. Таблиця стилів може кешувати в пам'яті, і може бути застосована до багатьох документів для істотної економії часу виконання.

Офіційно підтримка SOAP в PHP 4 була відсутня. Зазвичай, за необхідності використовувати SOAP, доводилося використовувати PEARS, але оскільки код був написаний повністю на PHP, програма не могла виконуватися так само, як вбудоване C-розширення. Інші доступні C-розширення так і не змогли досягти стабільної версії і широкого застосування, і тому не були включені в PHP 5. Підтримка SOAP в PHP 5 була повністю переписана як C-розширення і, хоча це розширення на поточний момент знаходиться в останній стадії бета-тестування, вирішили включити його до стандартного дистрибутива, оскільки воно практично повністю реалізує стандарт SOAP.

Висновок. Таким чином, використання PHP 5 для створення і розробки програмного забезпечення Інтернет-магазинів має ряд переваг:

- у новій об'єктній моделі Zend 2.0 використовується технологія об'єктних дескрипторів, що дозволяє економити пам'ять;
- отримання повного контролю над створенням і дублюванням об'єктів;
- використання нових функцій;
- використання нових розширень Tidy та Perl.

Все це дає збільшення продуктивності на 40 %, порівняно з версіями PHP 4, що значно скорочує час роботи та витрати на розробку програмного забезпечення для Інтернет-магазинів.

Отже, комплексна оцінка ефективності роботи програмного забезпечення на мові програмування PHP 5 для Інтернет-магазинів та Інтернет-реклами може бути основою розвитку та впровадження ефективної політики розвитку інформаційних технологій та програмного забезпечення в Україні.

Список використаної літератури:

1. Основи інформаційних систем : навч. Посібник / за ред. *В.Ф. Ситника*. – 2-ге вид., перероб. і доп. – 2011.
2. Andi Gutmans, Stig Bakken, Derick Rethans PHP5 Power Programming. — Prentice Hall, 2005. – 704 с.
3. PHP [Електронний ресурс]. – Режим доступу : <http://uk.wikipedia.org/wiki/PHP>.
4. PHP офіційний сайт розробників [Електронний ресурс]. – Режим доступу : <http://www.php.net>.
5. Інформаційні системи : підручник / *В.О. Новак, В.В. Матвеев, В.П. Бондар, М.О. Карпенко*. – 2-е вид. – К. : Каравелла, 2010. – 536 с.
6. *Хамел Г.* Конкурируя за будущее. Создание рынков завтрашнего дня : пер. с англ. / *Г.Хамел, К.К. Прахалад*. – М. : ЗАО «Олимп-Бизнес», 2012. – 288 с.
7. PHP 5 [Електронний ресурс]. – Режим доступу : http://site-konstruktor.com.ua/php/nove_v_php5.

ГРАБАР Ольга Іванівна – кандидат технічних наук, доцент кафедри програмного забезпечення Житомирського державного технологічного університету.

Наукові інтереси:

- інформаційні системи і технології;
- програмування;
- Інтернет-технології.

Стаття надійшла до редакції 07.06.2013

Грабар О.І. Використання мови програмування PHP 5 для створення програмного забезпечення Інтернет-магазинів

Грабар О.И. Использование языка программирования PHP 5 для создания программного обеспечения интернет-магазинов

Grabar O.I. Using a programming language PHP 5 for creation of software for internet-shops

УДК 004.738.5:338.46

Использование языка программирования PHP 5 для создания программного обеспечения интернет-магазинов / О.И. Грабар

В статье приведены результаты исследования методов создания и разработки программного обеспечения интернет-магазинов с использованием языка программирования PHP 5.

УДК 004.738.5:338.46

Using a programming language PHP 5 for creation of software for internet-shops / O.I. Grabar

The results of the searching of creation and development a software for internet-shops with using of PHP 5 are mentioned in the article.