

УДК 004.7

РОЗРОБКА ІНТЕРАКТИВНИХ МЕРЕЖЕВИХ ДОДАТКІВ НА ПЛАТФОРМАХ RUBY ТА PYTHON**Суворов А.М., Волковський О.С***Дніпропетровський національний університет ім. О.Гончара
кафедра автоматизованих систем обробки інформації
e-mail:artemsuv@gmail.com***Анотація.**

Суворов А.М., Волковський О.С. *Розробка інтерактивних мережеских додатків на платформах Ruby та Python. У статті розглядаються переваги та недоліки застосування платформ Ruby та Python при розробці інтерактивних мережеских додатків.*

Загальна постановка проблеми

До глобальної мережі Інтернет підключені системи засновані на різних апаратних і програмних платформах, виникає необхідність розробки для кожної платформи, яку передбачається підтримувати, спеціалізованої версії програмного забезпечення. Що призводить або до обмеження підтримуваних платформ, або до зростання витрат на розробку і підтримку. Тому проблема багатоплатформності для Web-додатків дуже гостра. Також при створенні Web-додатків значну роль грають зусилля які направлені на створення продукту, тому бажано щоб засоби розробки надавали як можна більше допомоги розробнику, тобто мали інтегровані інструменти розробки – фреймворки, та передбачали можливість легкої адаптації вже створених рішень під нові задачі.

Постановка задачі дослідження.

На цей час існує багато різноманітних середовищ, платформ, фреймворків для розробки Web-додатків, серед яких особливою популярністю користуються PHP, ASP.NET, Perl. Вони займають більшу частину ринку програмного забезпечення для розробки Web-додатків.

Розповсюдженість Web-додатків на мові PHP, обумовлена тим що вона поєднує в собі дві найбільші парадигми програмування – об'єктну та процедурну, вона створювалась саме з метою її використання при розробці Web-додатків. На мові PHP можна знайти багатий спектр готових рішень майже у будь-якій сфері застосування, які можна вільно використовувати згідно з ліцензією PHP. Для PHP були спроби реалізувати інтегровані пакети розробки, але вони не дуже обмежували розробників у можливостях, тому не мали загального застосування. А також те що PHP став багатоплатформною мовою, він підтримується на більшості платформах та операційних системах, а також на найрізноманітніших Web-серверах.

Технологія ASP.NET надає розробнику великий інструментарій для створення великих повноцінних Web-додатків, вона не обмежує розробника однією мовою, надаючи можливість розробки на будь якій мові що входить до пакету .NET. Але ця технологія має одну суттєву ваду – вона орієнтована для застосування лише на платформі MS Windows, що є значним обмеженням для Web-додатку.

Мова Perl є мовою що інтерпретується, це сприяє легкому переносу, додатків що розробляються між різними платформами. Інтерпретатори для мови Perl реалізовані на усіх існуючих платформах. Багатоплатформність є однією з основних вимог до сучасних додатків. Але мова Perl не надає розробнику повноцінного каркасу для розробки Web-додатків, надаючи для допомоги у розробці лише окремі бібліотеки функцій.

Разом з тим на ринку програмного забезпечення з'явилися дві нові платформи для розробки Web-додатків, це мова Ruby зі своїм фреймворком Rails та мова Python з фреймворком Django. Ці два рішення мають усі позитивні риси вище наведених інструментів,

та позбавлені їх недоліків. Вони пропонують повністю об'єктно-орієнтований каркас для створення універсальних платформнезалежних Web-додатків, взаємодіючих з різноманітними серверами баз даних та Web-серверами. Завдяки використанню при формуванні каркасу додатків патерну MCV (Model-Controller-View) – «модель-контролер-представлення» розроблювані додатки мають великий потенціал для масштабування та адаптації під нові завдання.

Метою дослідницької роботи є встановлення можливостей використання платформ Ruby та Python, для розробки Web-додатків, з урахуванням їх особливостей за наступними критеріями: складність системи, що розроблюється, адаптивність отриманої системи, її масштабованість, тиражованість, швидкість розробки додатку, розмір коду, розмір унікального коду, вимоги додатку до ресурсів обчислювальної системи на якій він виконується. Складність встановлення зв'язку між додатком та базою даних, що використовується.

Рішення задачі та результати досліджень

Для аналізу і дослідження задачі було спроектовано і розроблено Web-додаток, який моделює роботу інтернет-магазину. Для платформи Ruby розробка проводилась на базі бібліотек Ruby версії 1.8.6-26 та фреймворку Rails версії 2.3.4. Для платформи Python використовувались бібліотеки версії 2.6 та фреймворк Django версії 1.9. В якості серверу баз даних для збереження даних використовувався сервер MySQL версії 5.1.38.

Збільшений алгоритм функціонування розроблюваного додатку зображено на малюнку 1.

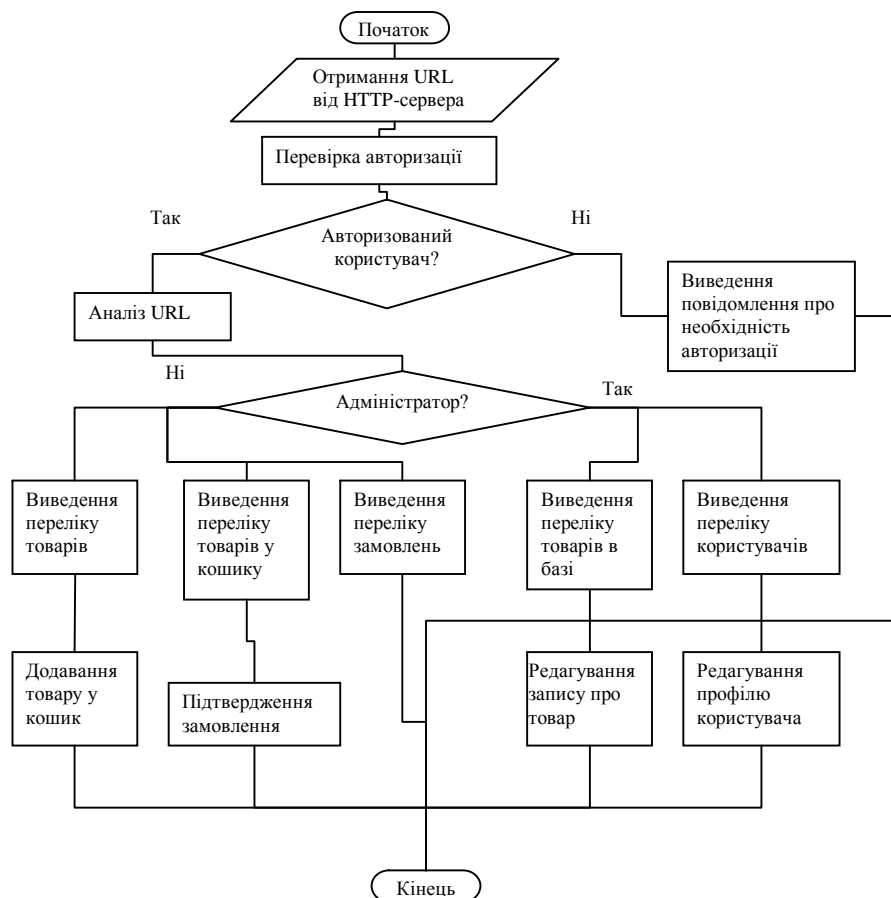


Рисунок 1. Збільшений алгоритм функціонування інтернет-магазину

Для збереження інформації про товари, користувачів магазину та замовлення використовувалась база даних, яка має схему зображену на малюнку 2.

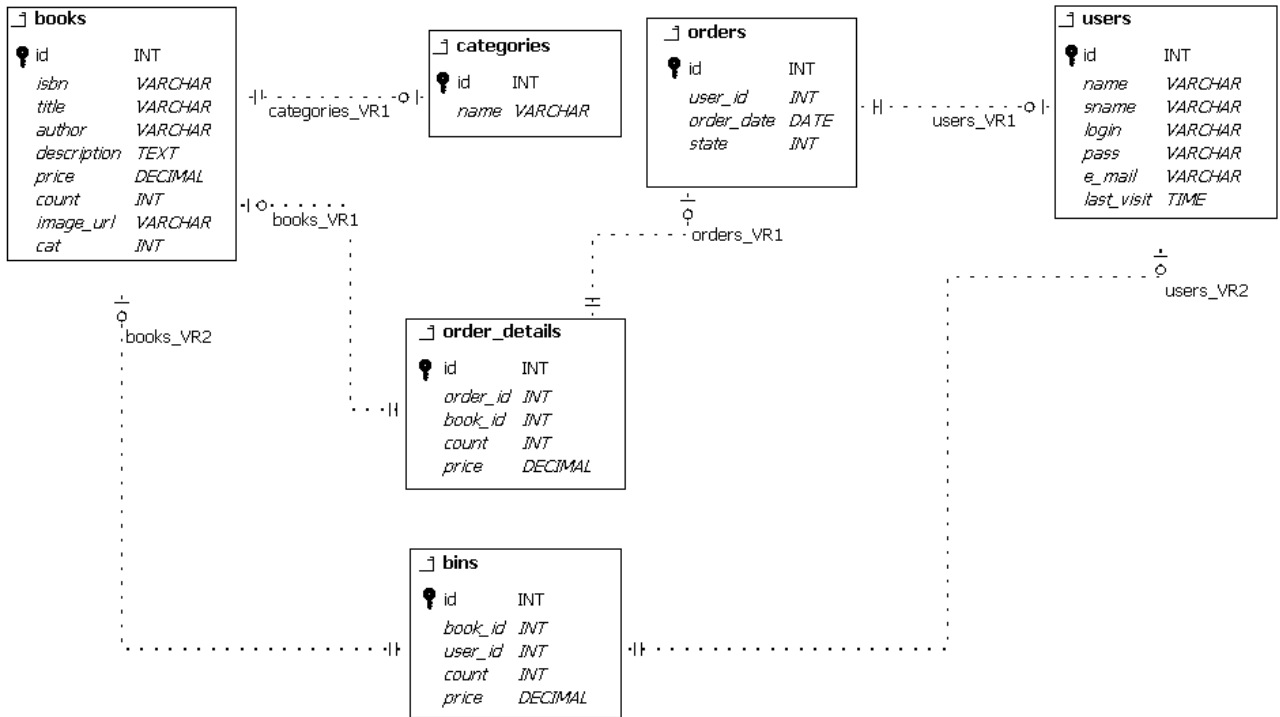


Рисунок 2. Схема розробленої бази даних

В результаті практичної реалізації, було отримано два додатки: один на мові Ruby, інший на мові Python. Виходячи з того що обидва фреймворки: Rails та Django, використовують патерн проектування MCV, отримана структура додатків збігається та має вигляд наведений на малюнку 3.

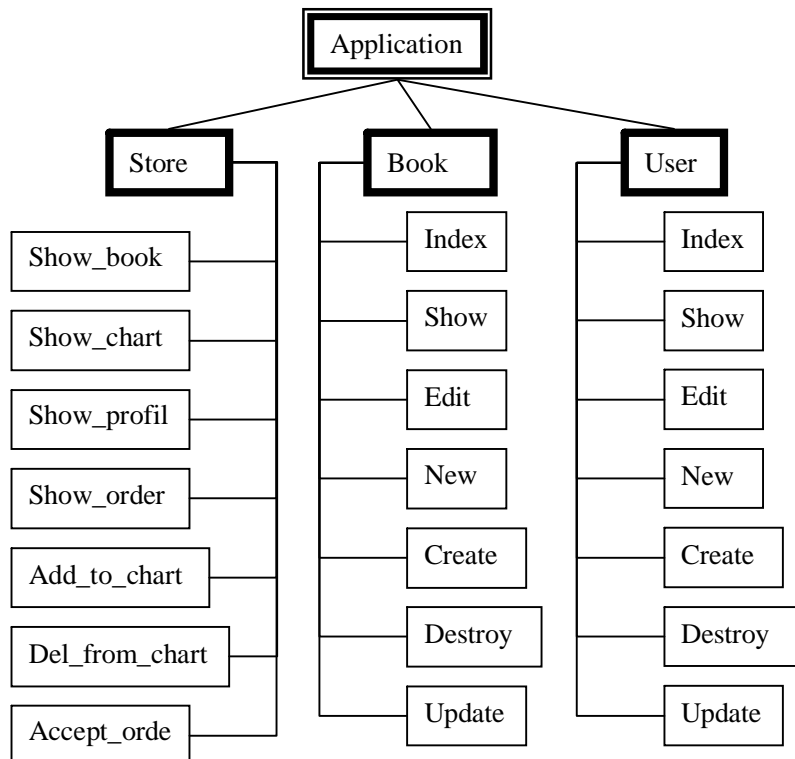


Рисунок 3. Модульна структура додатку

За допомогою вбудованих в середовище розробки засобів аналізу коду та тестування продуктивності програм, були проведені вимірювання витрат пам'яті та часу обробки

запитів. Для вимірювання було обрано фрагмент програми який виконує завантаження стартової сторінки, як той що найчастіше викликається, та фрагмент завантаження переліку замовлень, як той що виконує найбільшу кількість операцій. Оскільки мови Ruby Python є інтерпретованими, то окрім самих додатків ресурс пам'яті витрачається ще й на функціонування інтерпретаторів. Результати вимірювань витрат пам'яті наведено нижче, в таблиці 1.

Таблиця 1. Витрати пам'яті.

	Ruby	Python
Інтерпретатор	26 450кб	12 520кб
Додаток	13 520кб	4 590кб

В таблиці 2 наведено часові характеристики виконання запитів.

Таблиця 2. Часові характеристики.

	Ruby	Python
Завантаження стартової сторінки	63 мс	59мс
Завантаження переліку замовлень	313 мс	250мс

В таблиці 3 представлені результати обчислення метричних характеристик вихідного коду.

Таблиця 3. Метричні характеристики коду

	Ruby		Python	
Запит стартової сторінки	Строк коду	11	Строк коду	13
	Операторів	14	Операторів	16
Запит переліку замовлень	Строк коду	12	Строк коду	12
	Операторів	13	Операторів	13
Запит на додавання замовлення.	Строк коду	21	Строк коду	30
	Операторів	43	Операторів	51

Висновки

Розглянувши запропоновані платформи можна дійти висновку що вони надають потужні засоби для розробки Web-додатків, які відповідають сучасним вимогам – платформонезалежності, безпеки та наявності фреймворків і каркасів для швидкої розробки додатків. Виходячи із результатів вимірювання можна сказати що платформа Python є менш ресурсовитратною ніж платформа Ruby. Це є важливою перевагою, особливо якщо використовуються сервіси хост-провайдерів, адже від витрат ресурсів залежить їх вартість. Однак мова Ruby, формує менший за розмірами вихідний код, який до того ж ще й легший до сприйняття, ніж код на мові Python. Обидві платформи мають великі перспективи для розвитку та застосування при створенні Web-додатків. Їх впровадження на даний час обмежується лише новизною, та непристосованістю постачальників хост-сервісів для підтримки цих платформ.

Список літератури

1. Томас Д. Гибкая разработка веб-приложений в среде Rails. / Томас Д.,
2. Хенссон Д. Х. –СПб.:Питер,2008.
3. Холовати А. The Definitive Guide to Django: Web Development Done Right. / Холовати А., Калпан-Мосс.Я – Нью-Йорк:APRESS,2008