

Маковський Д. Ю.

студент фізико-математичного факультету

С. М. Горобець

к.п.н., доцент, доцент кафедри

прикладної математики та інформатики,

Житомирський державний університет імені Івана Франка

ОСОБЛИВОСТІ МОВИ ПРОГРАМУВАННЯ RUBY ТА ФРЕЙМВОРКУ RUBY ON RAILS

***Анотація.** В статті розглянуто історію, особливості архітектури та філософії мови програмування Ruby та фреймворку Ruby on Rails.*

Розглянуто принципи архітектури MVC. Окреслено перспективні напрями розвитку та приклади використання цих технологій.

***Ключові слова:** мова Ruby, фреймворк Ruby on Rails, архітектура MVC, gem-бібліотеки.*

***Аннотация.** В статье рассмотрено историю, особенности архитектуры и философии языка программирования Ruby и фреймворка Ruby on Rails. Рассмотрены принципы архитектуры MVC. Определены перспективные направления развития и примеры использования этих технологий.*

***Ключевые слова:** язык Ruby, фреймворк Ruby on Rails, архитектура MVC, gem-библиотеки.*

***Abstract.** In the article described history, features of the architecture and philosophy of the Ruby programming language and framework Ruby on Rails. Reviewed basic principles of architecture MVC. Also there are described perspective directions of progress with examples of the use of these technologies.*

***Key words:** language Ruby, framework Ruby on Rails, MVC architecture, gem-libraries.*

Світ зустрів Ruby у 1995 році. Автор мови програмування Ruby – Юкіхіро Матцумото, – японець (відомий у Ruby-спільнотах як Matz), цікавився основами програмування ще будучи студентом [1]. Мета створення нової мови програмування – бажання розробити справжню об'єктно-орієнтовану, просту у використанні, інтерпретовану мову програмування. Ю. Матцумото був великим прихильником Perl, що можна знайшло відображення у подібності семантики, схожості підходу до філософії побудови коду, та навіть у співзвучних назвах цих двох мов програмування (Perl – "перлина", а Ruby – "рубін"). Основне призначення Ruby – створення простих за синтаксисом і в той же час зрозумілих програм загального призначення, де важлива не швидкодія роботи програми, а малий час розробки.

Мова програмування Ruby є повністю об'єктно-орієнтованою в якій всі дані являються об'єктами, а кожна функція – методом. Цій мові притаманний цілий ряд цікавих можливостей, таких як [2]:

- багатопоточність (не залежить від операційної системи);
- строга динамічна типізація;
- вбудований збирач сміття;
- лаконічний синтаксис (код на Ruby можна читати вголос, і команди звучатимуть як прості англійські речення та вирази);
- підтримка блоків коду;
- реалізація шаблонів проектування безпосередньо у самій мові (спрощує архітектурне проектування);
- наявність групи особливостей, які пов'язані з "об'єктністю" у Ruby (перезавантаження операторів, які насправді являються методами тощо).

Фреймворк — платформа для програмних рішень, яку було створено для полегшення створення складних систем. Також можна сприймати як комплексну бібліотеку.

Випуск першої робочої версії найвідомішого фреймворку Ruby, - Ruby on Rails, здійснено командою програмістів Rails Core Team під

керівництвом Девіда Гайнемаєра Генссона у 2004 році. Ruby on Rails, як і сам Ruby, було створено для спрощення процесу веб-програмування, скорочення часу розробки веб-додатків та втілення можливості просто і дешево підтримувати вже написаний код на мові Ruby.

Ruby on Rails - багаторівневий фреймворк з відкритим кодом під ліцензією MIT (Massachusetts Institute of Technology), створений на мові програмування Ruby.

Не дивлячись на те, що спочатку Ruby та Ruby on Rails були створені і розвивались як технології для Linux-сумісних систем, сьогодні, набувши популярності, вони портовані на усі відомі операційні системи, включаючи Windows. Останні роки Ruby on Rails використовують в основному молоді компанії, некомерційні організації, і навіть крупний бізнес. Rails - це перед усім інфраструктура, тому це середовище вдало використовується для створення веб-додатків.

На сьогоднішній день Ruby on Rails є класичним представником архітектури MVC (Model-View-Controller, або Модель-Вид-Контролер) для створення веб-додатків, що забезпечує інтеграцію з веб-сервером та базами даних [3].

Архітектурі MVC притаманні наступні якості:

– "Модель" відповідає за об'єктно-орієнтоване представлення даних та роботу з реляційними базами даних (у Ruby On Rails за роботу з моделями відповідає *Active Record*);

– "Вид" відповідає за інтерфейс користувача, тобто використовує дані які отримує з "Контролера", зазвичай, до "Моделі" не звертається і описується за допомогою RHTML, які являють собою звичайні HTML сторінки зі вставками Ruby коду ERb, або Embedded Ruby (у Ruby On Rails за вигляд інтерфейсу користувача відповідає *Action View*).

– "Контролер" – це основний по вмісту коду фрагмент моделі MVC, що активно взаємодіє з "Моделлю" та "Видом" і відповідає за взаємодію з

користувачем (у Ruby On Rails за роботу "Контролера" відповідає *Active Controller*).

Ruby on Rails можна описати з ідеологічної точки зору, або просто розповісти про філософію розробки, яку образно називають *The Rails Way*.

Філософія Ruby on Rails включає наступні важливі принципи [4]:

– *Don't Repeat Yourself (DRY)* - принцип розробки програмного забезпечення, який свідчить про те, що кожна частина коду повинна бути точно виражена та не повинна повторюватись.

– *Convention Over Configuration* - принцип який передбачає попередню наявність певних важливих налаштувань веб-додатку за замовчуванням.

– *Principle of Least Astonishment (POLA)* - принцип передбаченості внутрішнього алгоритму роботи коду та прогнозованості результату.

Значну популярність Ruby та Ruby On Rails здобули не тільки через свою філософію та чітко визначений напрямок розвитку, але й завдяки культурі та соціальним традиціям, що передбачають взаємодопомогу між розробниками та вільний обмін знайденими рішеннями типових задач. Таким чином на даний момент зберігається тисячі готових рішень тих чи інших завдань, наприклад: системи аутентифікації, авторизації, коментування, інтеграції з платіжними системами, роботи поштової розсилки тощо.

Соціальне середовище програмістів Ruby та Ruby On Rails створило своєрідну культурну екосистему бібліотек та доповнень – джемів (*gems –дорогоцінні камінці*). Це явище доволі специфічне для комерційних мов програмування, тому що їм важко змагатися за рівень якості інструментів та повноту документації і специфікацій. Особливо це відчувається на таких дрібницях, як архітектурна незалежність або інтеграція з популярними платіжними системами. Існує всесвітній репозиторій джемів, – *Ruby Application Archive (RAA)*. Практично усі

розширення є вільними, розповсюджуються за ліцензією LGPL (Lesser General Public License) та можуть бути використані у будь-якому проекті без юридичних перешкод. Цікаво, але Ruby On Rails, в свою чергу, є джемом для Ruby. Ruby On Rails має власні бібліотеки та може використовувати велику кількість бібліотек написаних для Ruby.

RubyGems – система керування пакетами, яка включає в себе стандарт для програм та бібліотек у самодостатньому форматі джемів (gems). Основні особливості системи RubyGems [5]:

- простота інсталяції пакетів;
- можливість керування локальними пакетами;
- узгодження залежностей між встановленими пакетами;
- наявність веб-інтерфейсу для перегляду документації.

Сьогодні Ruby та Ruby on Rails використовуються у багатьох відомих та унікальних проектах, серед яких можна зазначити найбільш популярні [6]:

- Role Player Game Maker (RPG Maker), або Ruby Game Scripting System (RGSS), - відома серія програм для створення рольових відеоігор.
- Amarok відомий для користувачів *KDE* медіапрогравач.
- Metasploit найвідоміша програма для тестування програмного забезпечення на наявність експлоїтів.
- XChat - кросплатформовий IRC-клієнт.
- Kickstarter найвідоміший у світі сайт для залучення грошових коштів на реалізацію аматорських проектів,
- GitHub найрозповсюдженіший веб-сервіс для хостінгу ІТ-проектів,
- Twitter одна із передових соціальних мереж.
- SoundCloud онлайн-платформа для розповсюдження оцифрованої звукової інформації.

Не зважаючи на архітектурну простоту та зрозумілість синтаксису, рівень входження у світ Ruby доволі високий, що сприяє становленню висококваліфікованих спеціалістів-розробників на Ruby on Rails, які часто

самі займаються як Back-End, так і Front-End розробкою в межах одного проєкту.

“Народна ліцензія”, наявність значної кількості бібліотек та довідкових матеріалів для роботи з Ruby та Ruby on Rails свідчить про те, що популярність цих технологій у сфері розробки веб-додатків в майбутньому буде тільки зростати.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ ТА ЛІТЕРАТУРИ:

1. Smyth N. Ruby Essentials [Electronic resource] / N. Smyth – Mode of access: http://www.techotopia.com/index.php/What_is_Ruby%3F

2. Руби – лучший друг программиста: о Ruby [Электронный ресурс] – Режим доступа: <https://www.ruby-lang.org/ru/about/>

3. Керминський М. Ruby on Rails шаг за шагом. #1 Теория [Электронный ресурс] / М. Керминський – Режим доступа: <http://habrahabr.ru/post/49700/>

4. Ruby on Rails: Philosophy_and_design [Electronic resource] – Mode of access: https://en.wikipedia.org/wiki/Ruby_on_Rails

5. Guides: What is a gem [Electronic resource] – Mode of access: <http://guides.rubygems.org/what-is-a-gem/>

6. Ruby (programming language): Alternate implementations [Electronic resource] – Mode of access: https://en.wikipedia.org/wiki/Ruby_%28programming_language%29