

11.1. Бекенд фреймворки

Завдяки широкій популярності веб-додатків, їх програмування та розробка спрощуються. Десятиліття тому розробка та програмування були доступні лише небагатьом, хто мав спеціальну і тривалу освіту в цьому питанні, то сьогодні все більше людей з відсутністю спеціальних знань здатні створювати програмне забезпечення.

Розробка програмного забезпечення стає простішою. Сьогодні у розпорядженні розробників є багато інструментів, які полегшують життя як новачкам, так і експертам. Бекенд веб-фреймворки є одним із найважливіших інструментів програмування, на які сьогодні можуть спиратися розробники.

Відомості про бекенд веб-фреймворки

Бекенд-фреймворк — це програмний інструмент, який може автоматизувати деякі аспекти процесу веб-розробки, спрощуючи та прискорюючи його. Бекенд -фреймворк — це бібліотека моделей та інструментів, які можуть допомогти розробнику створити архітектуру свого веб-сайту, веб-додатку, мобільного додатка або програмного забезпечення.

Можна навести приклад і асоціювати бекенд-фреймворк як шаблон для проекту графічного дизайну. Без фреймворка є біле полотно, на якому потрібно побудувати кожну частину, вручну спроектувати кожен елемент, вибрати будь-який колір та положення та переконатися, що все може працювати разом. З шаблоном (або фреймворком) вже половину роботи зроблено: є задана структура, яку можна налаштувати на свій розсуд, а також є заздалегідь створені елементи, які можна використовувати та інтегрувати разом.

З бекенд-фреймворком робота розробника спрощується аналогічно: замість того, щоб писати кожний рядок коду різними мовами програмування, розробник може використовувати задану структуру та задані інструменти та модулі для створення своєї програми.

Переваги використання бекенд веб-фреймворку

Переваги використання бекенду веб-фреймворку не тільки у простоті та швидкості процесу програмування. Так, використання фреймворку економить час і полегшує життя як досвідченим, так і розробникам-початківцям, але бекенд веб-фреймворк також забезпечує:

- **Безпека.** Фреймворк забезпечує безпеку з багатьох точок зору. Захист від зовнішніх атак і проблем, збоїв, повільності та інших негативних аспектів може знизити якість програми, що створюється.
- **Масштабованість.** Фреймворки дуже гнучкі. Після того, як обрано правильний бекенд фреймворк для проекту - можна використовувати ту ж структуру як при створенні простого веб-додатку, так і при розробці складного.
- **Інтеграція.** фреймворки спрощують інтеграцію, оскільки надають інструменти попереднього збирання.

Типи бекенд веб-фреймворків

Існує два основних типи бекенд веб-фреймворків:

- **Веб-фреймворк на стороні клієнта.** Це фреймворки, які твердо орієнтовані на інтерфейс користувача. Клієнтська частина містить всі аспекти, пов'язані з тим, що користувач бачить, візуалізує і як він може взаємодіяти з додатком.
- **Веб-фреймворк на стороні сервера.** Серверна структура використовується для бекенд функцій: маніпулювання базою даних, керування даними, керування веб-сервером тощо. Існують й інші способи класифікації веб-фреймворків: наприклад, на підставі мов програмування, на яких вони засновані: Python, Javascript, PHP та .NET є найбільш популярними, але є багато інших. Альтернативою є no-code підхід. Такі платформи, як AppMaster, можуть надати no-code фреймворк для мобільного або веб-додатку.

Фреймворк no-code AppMaster

Платформи no-code, такі як AppMaster, забезпечують надійну та гнучку структуру. Надійну, оскільки гарантується безпека та можна використовувати та інтегрувати готові елементи, які можуть бути дуже складними. Гнучку, оскільки AppMaster забезпечує найвищий рівень свободи та налаштування.

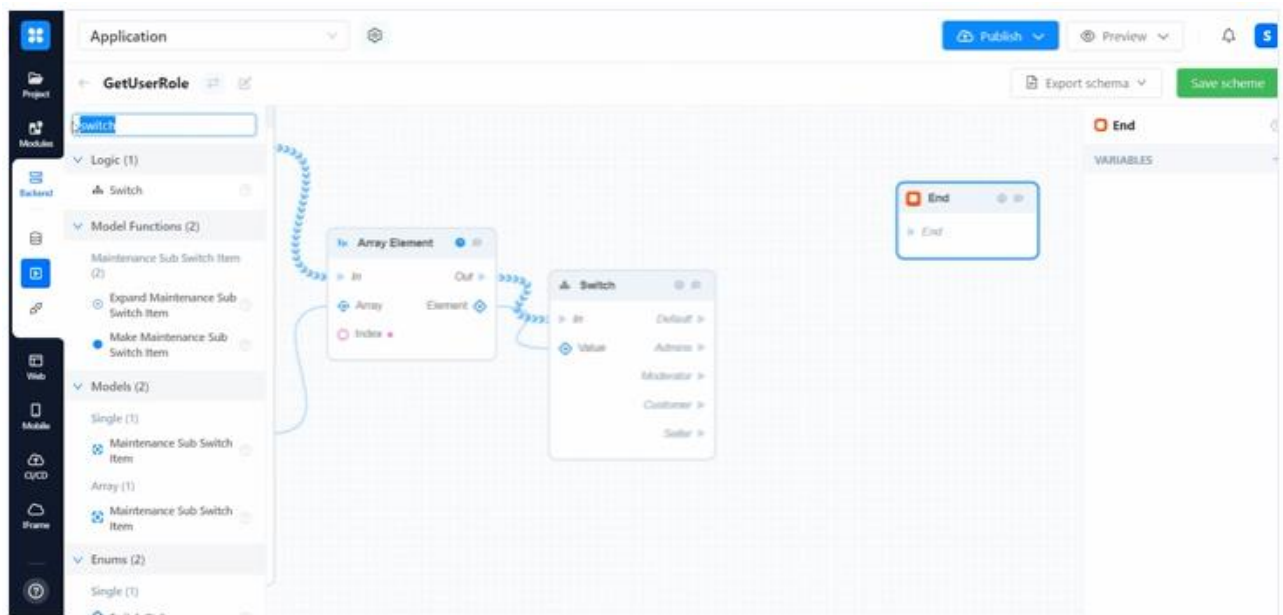


Рис.1. Робота з бекенд-фреймворком AppMaster

AppMaster надає фреймворк архітектуру для проектів, яка дозволяє взагалі уникати використання коду та мов програмування. З візуальним інтерфейсом можна створити веб- або мобільний додаток, додати функціональні можливості, опублікувати та зробити багато іншого.

AppMaster автоматично генерує бекенд код. Він також надає доступ до вихідного коду, щоб можна виконувати операції експортування, зберігання, редагування або інше. Впливаючи на код, можна повністю контролювати створене програмне забезпечення.



Рис.3. Інтерфейс бекенд-фреймворка AppMaster

Переваги використання AppMaster

- AppMaster ідеально підходить для початківців, оскільки не вимагає, щоб розробник був експертом з використання мов програмування.
- AppMaster корисний для досвідчених розробників: він допомагає їм полегшити та прискорити роботу, оскільки можна автоматизувати деякі процеси (наприклад, генерувати код автоматично, використовуючи візуальний інтерфейс, що завжди швидше ніж вручну писати кожен рядок коду).
- AppMaster автоматично генерує вихідний код та дозволяє розробнику його редагувати.
- AppMaster надає функції, які допомагають у процесі запуску, коли потрібно остаточно опублікувати мобільний або веб-додаток та подальші оновлення.

AppMaster - це платформа no-code, призначена для створення веб-додатків та мобільних додатків. AppMaster буде вашим ідеальним вибором для розробника будь якого рівня фаховості.

Бекенд веб-фреймворки

Django

Це веб-фреймворк на основі Python, орієнтований на створення складних веб-сайтів, що взаємодіють з базами даних.



```
models.py
1 from django.db import models
2 from django.conf import settings
3 from django.utils.text import slugify
4
5
6 class Image(models.Model):
7     user = models.ForeignKey(settings.AUTH_USER_MODEL,
8                             related_name='images_created')
9     title = models.CharField(max_length=200)
10    slug = models.SlugField(max_length=200, blank=True)
11    url = models.URLField()
12    image = models.ImageField(upload_to='images/%Y/%m/%d')
13    description = models.TextField(blank=True)
14    created = models.DateTimeField(auto_now_add=True,
15                                  db_index=True)
16
17    def __str__(self):
18        return self.title
19
20    def save(self, *args, **kwargs):
```

Рис.2. Інтерфейс бекенд-фреймворка Django

Переваги використання Django

- Масштабованість. Зокрема, його функція повторного використання коду полегшує розробникам адаптацію збільшення трафіку на веб-сайті.
- Django орієнтований на SEO, а це означає, що веб-сайти на його основі легше оптимізувати та зробити SEO-дружніми. Одна з основних причин цього полягає в тому, що можна підтримувати програми на основі Django на серверах за допомогою URL-адрес, а не IP-адрес.
- Django має велику спільноту, готову надати підтримку всім, хто використовує цей інструмент.
- Орієнтація на відсутність коду: Django НЕ Є фреймворком по-code, але має деякі функції по-code. Існують пакети, які розробник може використовувати без використання коду.
- Універсальність: Django особливо підходить для веб-сайтів, керованих базами даних, але він дуже універсальний. Його можна використовувати для створення будь-якого типу веб-сайту.

Django є одним з найкращих веб-фреймворків для створення веб-додатку, що керований базою даних.

Express.js

Express.js – це фреймворк для веб-додатків Node.js. Як і AppMaster, він націлений на створення веб- та мобільних додатків, але це не платформа по-code. Однак серед фахових розробників Express.js є доволі популярним бекенд-фреймворком. Його використано для створення відомих веб-додатків, таких як MySpace та Uber.

Express JS

```
js index.js x
1  const express = require('express');
2  const app = express();
3
4  const targetBaseUrl = 'http://www.letsboot.com/';
5
6  function handleRedirect(req, res) {
7    const targetUrl = targetBaseUrl + req.originalUrl;
8    res.redirect(targetUrl);
9  }
10
11 app.get('*', handleRedirect);
12
13 const port = process.env.port || 3000;
14
15 app.listen(port);
16 |
```

Рис.4. Інтерфейс бекенд-фреймворка

Переваги використання Express.js

- Містить багато стандартних функцій Node.js, які полегшують роботу розробника.
- Express пропонує кілька шаблонів, які допоможуть створити веб-сторінки.
- Перевірка в Express.js спрощується, оскільки легше визначити точні частини коду, які містять помилки.
- Express.js надає широкі можливості щодо масштабування.
- Велика спільнота Express.js, яка підтримує в роботі і за необхідності надає допомогу.
- З Express.js використовується JavaScript як для зовнішнього, так і для внутрішнього інтерфейсу веб-розробки.

Express.js може стати ідеальним бекенд-фреймворком у разі хорошого володіння JavaScript, але недостатньо досвіду роботи з іншими мовами програмування.

Flask

Flask це бекенд-фреймворк на основі Python. Flask це фреймворк WSGI. Web Server Gateway Interface – це стандарт взаємодії між Python-програмою, яка виконується на стороні сервера, і самим веб-сервером, наприклад, Apache. Flask запускає веб-сервер, а не браузер користувача. Багато сучасних веб-додатків працюють на Flask, оскільки це дозволяє створювати персоналізовані проекти, які працюють без збоїв.



```
server.py x login.html x upload.html x
1 from flask import Flask, redirect, url_for, request, render_template
2 from werkzeug.utils import secure_filename
3 app = Flask(__name__)
4
5 @app.route('/upload')
6 def upload_view():
7     return render_template('upload.html')
8
9 @app.route('/uploader', methods = ['GET', 'POST'])
10 def uploading():
11     if request.method == 'POST':
12         file = request.files['file']
13         file.save(secure_filename(file.filename))
14         return 'File Uploaded Successfully'
15
16 if __name__ == '__main__':
17     app.run(debug = True)
```

Рис.5. Інтерфейс бекенд-фреймворка

Переваги використання Flask

- З Flask надається вбудований сервер веб-розробки: він спрощує перевірку та дозволяє розробникам реалізовувати код Python у браузері при виникненні помилок у запиті.
- Flask, що базується на Python дозволяє швидко розробляти програми. Python відомий тим, що вимагає меншого коду, ніж інші мови програмування для розробки.
- З Flask можна легко налаштувати веб-сервери.
- Швидке навчання. Flask не простий у освоєнні, але спільнота навколо Flask настільки велика, що можна знайти багато документації та посібників. Більшість документації від самих розробників Flask. Навіть новачки можуть швидко навчитися.

Будь-який розробник, знайомий із мовою програмування Python, може легко використовувати Flask. Flask - ідеальний бекенд-фреймворк, якщо потрібно зосередитися на веб-розробці на стороні сервера,.

Laravel

Laravel це фреймворк на основі PHP з відкритим вихідним кодом і спеціально розроблений для створення веб-додатку на основі Symfony, який слідує за архітектурою MVC. Фреймворк ідеально підходить для бекенд розробки.



```
1 <?php
2
3 namespace App\Services;
4
5 use App\Transaction;
6 use Carbon\Carbon;
7 use Illuminate\Http\Request;
8
9 class ReportService {
10
11     public function transactionReport(Request $request)
12     {
13         $q = Transaction::with(['name', 'project'])
14             ->with(['name', 'transaction_type'])
15             ->with(['name', 'income_source'])
16             ->with(['name', 'currency'])
17             ->orderBy('name', 'transaction_date', 'desc');
18
19         if ($request->has('project')) {
20             $q->where('project_id', $request->project);
21         }
22
23         $transactions = $q->get();
24         $entries = [];
25
26         foreach ($transactions as $row) {
27             if ($row->transaction_date != null) {
28                 $date = Carbon::createFromFormat(config('app.panel.date_format'), $row->transaction_date)->format('Y-m');
29                 if (!isset($entries[$date])) {
30                     $entries[$date] = [];
31                 }
32
33                 $currency = $row->currency->code;
34
35                 if (!isset($entries[$date][$currency])) {
36                     $entries[$date][$currency] = [
37                         'income' => 0,
38                         'expenses' => 0,
39                     ];
40                 }
41             }
42         }
43     }
44 }
```

Рис.6. Інтерфейс бекенд-фреймворка Laravel

Переваги використання Laravel

- Laravel пропонує багато функцій безпеки. Він використовує алгоритм bcrypt хешування для створення паролів, який робить їх дуже безпечними.
- З Laravel можна використовувати багато вбудованих шаблонів для створення макетів і контенту.
- Крім PHP, Laravel надає віджети з кодом CSS та JS.
- Ідеально підходить для створення складних проектів.
- Важливою особливістю Laravel є просте API, що працює з бібліотекою SwiftMailer.

Laravel взагалі не підходить для новачків. Він є надто складним, якщо потрібно керувати простим або невеликим проектом. Laravel підходить лише для досвідчених розробників, знайомих з багатьма мовами програмування та їх інтеграції.

Ruby on Rails

Ruby on Rails — це фреймворк для веб-розробки, що базується на мові програмування Ruby, з відкритим вихідним кодом. Ruby – не найпопулярніша мова програмування, проте Ruby on Rails – дуже популярний фреймворк, оскільки він допомагає розробникам у процесі програмування, забезпечуючи простоту, автоматичне тестування та бібліотеки.



```
1 book = Book.create(  
2   title: "How the Grinch Stole Christmas",  
3   isbn: 394800796,  
4   description: "...",  
5   published_date: DateTime.new(1957, 11, 4),  
6 )  
7 book.authors << Author.where(first_name: "Dr.", last_name: "Seuss").first_or_create  
8  
9 book = Book.create(  
10  title: "The Talisman",  
11  isbn: 9780375507779,  
12  description: "...",  
13  published_date: DateTime.new(2001, 9, 15),  
14 )  
15 book.authors << Author.where(first_name: "Stephen", last_name: "King").first_or_create  
16 book.authors << Author.where(first_name: "Peter", last_name: "Straub").first_or_create  
17  
18 book = Book.create(  
19  title: "The Stand",  
20  isbn: 9780385199575,  
21  description: "...",  
22  published_date: DateTime.new(1990, 5, 1),  
23 )  
24 book.authors << Author.where(first_name: "Stephen", last_name: "King").first_or_create
```

Рис.7. Інтерфейс бекенд-фреймворка

Переваги використання Ruby on Rails

- Ruby on Rails запускає автоматичні тести коду, написаного розробниками. Це допомагає заощадити багато часу.
- Ruby on Rails є універсальним і гнучким. Це дозволяє розробникам повністю контролювати свої проекти: вони можуть вирішувати, як має працювати база даних додатку. Далі фреймворк автоматично створить код відповідно до інструкцій розробників.
- Ruby on Rails надає багато бібліотек, в яких розробники можуть знайти будь-який інструмент, необхідний для створення баз даних.

Початківці можуть легко вивчити Ruby on Rails, тому він рекомендується їм до використання.

Spring

Spring – це веб-фреймворк, що базується на мові програмування Java. Його використано для створення багатьох популярних веб-сайтів, наприклад Wix.



```
1 package com.vl.mnsp;
2
3 import org.apache.logging.log4j.LogManager;
4 import org.apache.logging.log4j.Logger;
5 import org.springframework.http.MediaType;
6 import org.springframework.web.bind.annotation.PathVariable;
7 import org.springframework.web.bind.annotation.RequestMapping;
8 import org.springframework.web.bind.annotation.RequestMethod;
9 import org.springframework.web.bind.annotation.RestController;
10
11 @RestController
12 @RequestMapping("/greeting")
13 public class GreetingServiceImpl {
14
15
16
17
18
19     private static final Logger LOGGER = LogManager.getLogger(GreetingServiceImpl.class);
20
21     @RequestMapping(path = "/{name}", method = RequestMethod.GET,
22         produces = MediaType.APPLICATION_JSON_VALUE)
23     public String greeting(@PathVariable(name = "name") String name) {
24
25
26         {
27             LOGGER.info("##### greeting #####");
28             System.out.println("duc nguyen");
29             return "Hello " + name + "\n";
30         }
31     }
32 }
```

Рис.8. Інтерфейс бекенд-фреймворка Spring

Переваги використання Spring

- Оскільки Java є популярною мовою програмування, Spring є також популярним.
- Spring просувається з іншими проектами, які можуть підвищити продуктивність.
- Spring має здатність до масштабування, особливо завдяки супутнім проектам.

Spring підходить для розробників, знайомих з мовою програмування Java.

ASP.NET

ASP.NET — це фреймворк із відкритим вихідним кодом, що використовується для створення програм .NET. Він використовує код C# і є ідеальним веб-фреймворком, якщо пріоритетом є забезпечення позитивного досвіду користувача.



```
Startup.cs
1 using Microsoft.AspNetCore.Builder;
2 using Microsoft.AspNetCore.Hosting;
3 using Microsoft.AspNetCore.Http;
4 using Microsoft.Extensions.DependencyInjection;
5
6 namespace MyWebApp
7 {
8     1 reference
9     public class Startup
10     {
11         // This method gets called by the runtime. Use this method to add services to
12         // the container.
13         // For more information on how to configure your application, visit
14         // http://go.microsoft.com/fwlink/?LinkID=398940
15         0 references
16         public void ConfigureServices(IServiceCollection services)
17         {
18         }
19
20         // This method gets called by the runtime. Use this method to configure the
21         // HTTP request pipeline.
22         0 references
23         public void Configure(IApplicationBuilder app)
24         {
25             app.Run(async (context) =>
26             {
27                 await context.Response.WriteAsync("Hello World!");
28             });
29         }
30     }
31 }
```

Рис.9. Інтерфейс бекенд-фреймворка

Переваги використання ASP.NET

- Фреймворк є вузькоспеціалізованим.
- Використовує JavaScript, який є однією з найпопулярніших мов програмування.

ASP.NET Code — ідеальна архітектура для створення мобільних або веб-додатків та забезпечення безперебійної роботи користувачів Window.

Fiber

Fiber – це серверна архітектура розробки для розробників Go. Go - це мова програмування з відкритим вихідним кодом, яку можна використовувати для загальних цілей.



```
1 // > Fiber is an Express-inspired web framework written in Go with >
2 // > Github Repository: https://github.com/gofiber/fiber
3 // > API Documentation: https://docs.gofiber.io
4
5 package fiber
6
7 // go test -v ./... -run=^$ -bench=Benchmark_Router -benchmem -count=2
8
9 import (
10     "encoding/json"
11     "errors"
12     "fmt"
13     "io/ioutil"
14     "net/http/httptest"
15     "strings"
16     "testing"
17
18     "github.com/gofiber/fiber/v2/utils"
19     "github.com/valyala/fasthttp"
20 )
21
22 var routesFixture = routeJSON{}
23
24 func init() {
25     dat, err := ioutil.ReadFile("../github/testdata/testRoutes.json")
26     if err != nil {
27         panic(err)
28     }
29     if err := json.Unmarshal(dat, &routesFixture); err != nil {
30         panic(err)
31     }
32 }
```

Рис.10. Інтерфейс бекенд-фреймворка Fiber

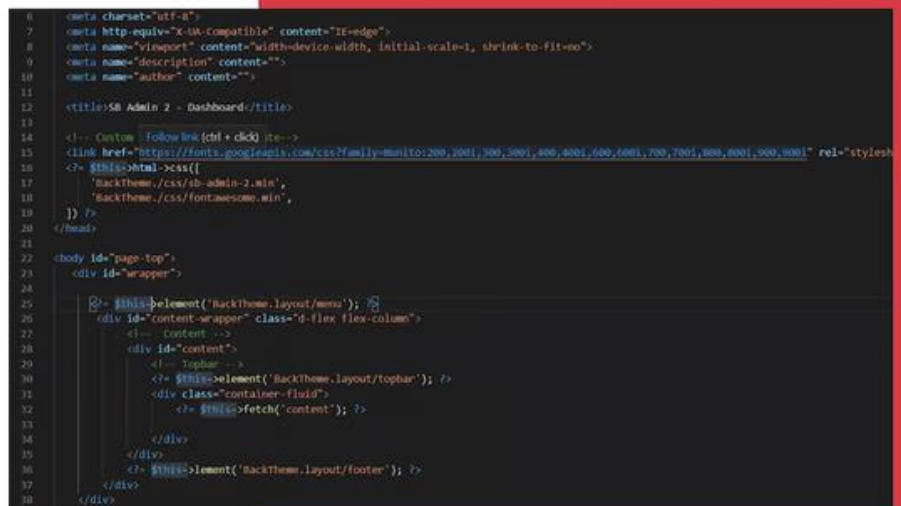
Переваги використання Fiber

- Проста, але потужна структура фреймворку.
- Fiber має здатність до масштабування
- Фреймворк Fiber можна легко інтегрувати зі сторонніми бібліотеками.

Fiber буде хорошим надбанням у разі створення звичайного веб-додатку.

CakePHP

CakePHP можна розглядати як альтернативу Ruby on Rails для PHP-розробників.



Переваги використання CakePHP

- Для розробників, знайомих з мовою програмування PHP, CakePHP може надати ідеальну архітектуру для створення веб-застосунку будь-якого рівня.

Більшість цих фреймворків написані популярними мовами програмування та пропонують відмінний функціонал та можливості для реалізації вашого проекту. Вибір правильного серверного середовища залежить від унікальних вимог проекту. Зрештою, ціль фреймворку веб-розробки — зменшити обсяг коду, який вам потрібно написати. Найефективніший спосіб скоротити обсяг написання коду – звернути увагу на AppMaster платформу no-code. Кількість рядків коду гарантовано буде зменшено до нуля.

<https://appmaster.io/ru/blog/10-luchshikh-veb-freimvorkov>