

# ПРОЄКТУВАННЯ ТА РОЗРОБКА БЕКЕНДУ СУЧАСНИХ ВЕБСАЙТІВ ІЗ ВИКОРИСТАННЯМ МОВИ ПРОГРАМУВАННЯ PYTHON

Вінницький національний технічний університет

## Анотація

У статті досліджується процес проектування та розробки серверної частини (бекенду) сучасних вебсайтів із використанням мови програмування Python. Здійснено порівняльний аналіз найпопулярніших інструментів веброзробки — фреймворків Django та Flask. Визначено їхні архітектурні особливості, рівень гнучкості та вплив на швидкість створення готового продукту. Обґрунтовано відповідність екосистеми Python сучасним індустріальним стандартам для побудови стабільних та масштабованих вебресурсів.

**Ключові слова:** Python, бекенд, веброзробка, проектування, вебсайти, Django, Flask, серверна архітектура.

## Abstract

This paper investigates the process of designing and developing the server-side (backend) of modern websites using the Python programming language. A comparative analysis of the most popular web development tools, the Django and Flask frameworks, is conducted. Their architectural features, level of flexibility, and impact on the speed of creating a finished product are determined. The compliance of the Python ecosystem with modern industrial standards for building stable and scalable web resources is substantiated.

**Keywords:** Python, backend, web development, design, websites, Django, Flask, server architecture.

## Вступ

В умовах стрімкої цифровізації та постійного ускладнення інтернет-сервісів, проектування та розробка надійної серверної логіки (бекенду) є фундаментальним етапом створення будь-якого сучасного вебсайту. Серед розмаїття доступних технологій мова програмування Python впевнено утримує позиції лідера індустрії. Це зумовлено її високою читабельністю, лаконічністю синтаксису та надзвичайно потужною екосистемою. Сьогодні Python успішно застосовується як для швидкого прототипування локальних сервісів, так і для розгортання високонавантажених корпоративних вебплатформ.

У цій роботі проаналізовано ключові етапи проектування серверної архітектури на базі Python, розглянуто специфіку розробки з використанням провідних фреймворків та виокремлено їхні головні переваги для інженерів.

## Основна частина

Проектування сучасного бекенду вимагає від розробників не лише написання коду, але й побудови ефективної взаємодії між сервером, базою даних та клієнтською частиною. Python, як високорівнева мова програмування загального призначення, дозволяє архітекторам фокусуватися на реалізації складної бізнес-логіки, суттєво скорочуючи час на написання шаблонного коду. Для ефективної веброзробки найчастіше застосовуються спеціалізовані фреймворки, що автоматизують такі процеси як маршрутизація (роутинг), безпека та ORM-комунікація з базами даних [1].

У сучасній практиці проектування домінують два концептуально різні підходи: використання комплексних фреймворків та гнучких мікрофреймворків. Найяскравішими представниками цих напрямків у середовищі Python є Django та Flask відповідно.

**Django** — це потужний фреймворк, що базується на архітектурному шаблоні MVT (Model-View-Template) та філософії «все включено» (batteries-included). Він пропонує розробникам готовий набір інструментів: від вбудованої системи автентифікації до зручної панелі адміністратора. Такий підхід робить Django оптимальним вибором для проектування масштабних вебсайтів зі складною та розгалуженою інфраструктурою, де важлива стандартизація розробки [2].

**Flask**, на противагу, є мінімалістичним мікрофреймворком. Його базова конфігурація включає лише необхідні компоненти для обробки HTTP-запитів та запуску вебсервера. Це надає інженерам максимальну архітектурну свободу у виборі сторонніх компонентів. Flask є ідеальним рішенням для

розробки мікросервісів, RESTful API та нестандартних вебсайтів, де критично важливим є гнучке проектування кожного вузла системи [3].

Демонстрацією простоти використання Python під час розробки є процес створення базового маршруту. На рисунку 1 наведено фрагмент програмного коду у середовищі Flask, який приймає клієнтський запит та повертає згенеровану вебсторінку, що вимагає написання лише кількох зрозумілих рядків.

```
main.py X
C: > Users > admin > Desktop > main.py > ...
1  from flask import Flask, render_template
2
3  app = Flask(__name__)
4
5  @app.route('/')
6  def index():
7      # Повертає відрендерену головну сторінку
8      return render_template('index.html', title='Головна')
9
10 @app.route('/about')
11 def about():
12     # Приклад іншого маршруту
13     return "<h1>Про нас</h1>"
14
15 if __name__ == '__main__':
16     app.run(debug=True)
```

Рис. 1. Реалізація базової маршрутизації за допомогою фреймворку Flask

Розробка серверної частини також передбачає детальне проектування логіки взаємодії систем. Для кращого розуміння цих процесів, на рисунку 2 зображено UML-діаграму, яка ілюструє типовий життєвий цикл обробки запиту у сучасному вебсайті на базі Python.

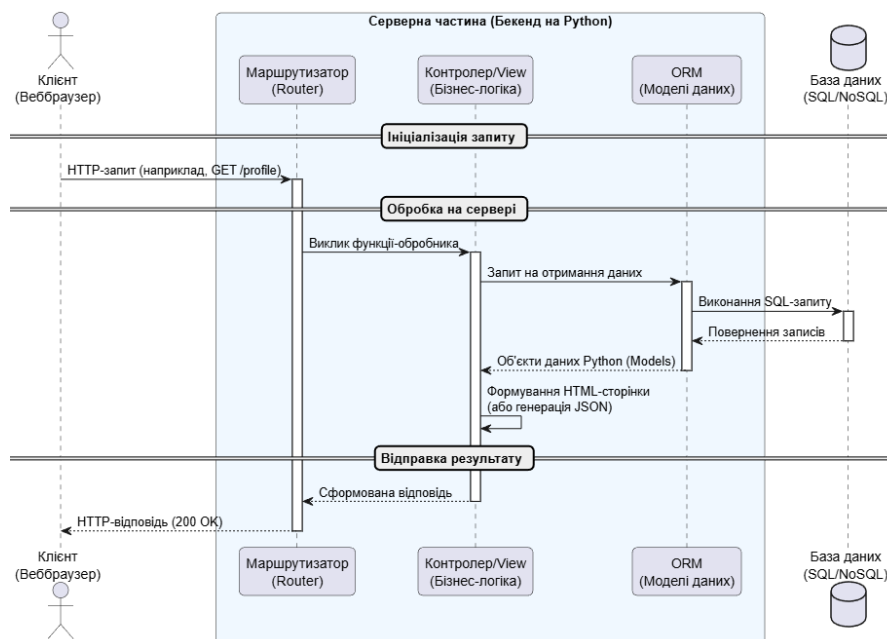


Рис. 2. UML-діаграма логіки взаємодії клієнтської та серверної частин вебсайту

Загалом, використання мови Python для проектування та розробки бекенду має низку беззаперечних переваг:

1. Прискорений цикл розробки (Time-to-Market): Прозорий синтаксис та доступність готових архітектурних патернів дозволяють мінімізувати час від ідеї до релізу готового вебсайту.

2. Масштабна екосистема: Пакетний менеджер `pip` відкриває доступ до тисяч бібліотек, що значно спрощує інтеграцію платіжних систем, хмарних сервісів та інструментів штучного інтелекту безпосередньо у бекенд.

3. Легкість підтримки: Чистота коду спрощує подальше масштабування проекту та онбординг нових фахівців, що є критичним фактором у командній розробці великих порталів.

4. Гнучкість інтеграції: Python дозволяє легко поєднувати вебсайти з аналітичними скриптами (Data Science) або системами автоматизації в межах єдиного інженерного середовища.

Незважаючи на те, що Python може поступатися у чистій швидкості виконання компільованим мовам (наприклад, Go), продуктивність сучасних вебсайтів частіше залежить від оптимізації бази даних та архітектури кешування. Ці аспекти успішно вирішуються на етапі грамотного проєктування системи.

### Висновки

Мова програмування Python є надзвичайно ефективним та адаптивним інструментом для проєктування та розробки бекенду сучасних вебсайтів. Вибір конкретного фреймворку залежить від технічного завдання та масштабу продукту: Django забезпечує надійність і швидкість розробки для комплексних порталів, тоді як Flask пропонує необхідну гнучкість для створення легких кастомних мікросервісів. Завдяки розлогій екосистемі та зручності проєктування архітектури, Python продовжує бути технологічним стандартом у сфері сучасної веброзробки.

### СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Python Software Foundation. Python 3 Documentation. [Електронний ресурс] – Режим доступу: <https://docs.python.org/3/> (дата звернення: 21.05.2026).
2. Django Software Foundation. Django Documentation. [Електронний ресурс] – Режим доступу: <https://docs.djangoproject.com/> (дата звернення: 21.05.2026).
3. Pallets. Flask Documentation. [Електронний ресурс] – Режим доступу: <https://flask.palletsprojects.com/> (дата звернення: 21.05.2026).

**Вальчишен Олександр Ігорович** – студент групи ІІСТ-226, кафедра автоматизації та інтелектуальних інформаційних технологій, факультет інтелектуальних інформаційних технологій та автоматизації, Вінницький національний технічний університет, м. Вінниця, e-mail: [valchishensasha12@gmail.com](mailto:valchishensasha12@gmail.com)

**Богач Ілона Віталіївна** – к.т.н., доцент кафедри автоматизації та інтелектуальних інформаційних технологій, факультет інтелектуальних інформаційних технологій та автоматизації, Вінницький національний технічний університет, м. Вінниця, e-mail: [ilona.bogach@gmail.com](mailto:ilona.bogach@gmail.com)

**Ремовський Юрій Андрійович** - студент групи 2ПКТ-24б, факультет інтелектуальних інформаційних технологій та автоматизації, Вінницький національний технічний університет, м. Вінниця, email:

**Valchyshen Oleksandr Ihorovych** – student of ІІСТ-22b group, Department of Automation and Intelligent Information Technologies, Faculty of Intelligent Information Technology and Automation, Vinnytsia National Technical University, Vinnytsia, e-mail: [valchishensasha12@gmail.com](mailto:valchishensasha12@gmail.com)

**Bogach Ilona Vitaliivna** – Associate Professor of Automation and Intelligent Information Technologies Department, Faculty of Intelligent Information Technology and Automation, Vinnytsia National Technical University, Vinnytsia, e-mail: [ilona.bogach@gmail.com](mailto:ilona.bogach@gmail.com)

**Removskiy Yurii Andriyovych** - student of group 2PKT-24b, Faculty of Intellectual Information Technologies and Automation, Vinnytsia National Technical University, Vinnytsia, email: [removskiy.edu@gmail.com](mailto:removskiy.edu@gmail.com)