

Національний технічний університет України  
«Київський політехнічний інститут ім. І. Сікорського»  
Факультет інформатики та обчислювальної техніки  
Кафедра обчислювальної техніки

## **Розрахунково-графічна робота**

з курсу «Інтеграція програмних систем»

на тему: “Шкільний вебсайт”

Виконали:  
студенти 4-го курсу  
ФІОТ групи ІО-31  
Крук Ярослав  
Шарпань Антон  
Головенько Вячеслав  
Горох Олександр

## 1. Опис проекту

Розроблена програма призначена для учнів та вчителів Корделівської загальноосвітньої школи. Панель адміністратора дозволяє йому динамічно генерувати контент сайті. На даному етапі реалізована лише генерація контенту для головної сторінки, а саме відділ новин. Адміністратор може зайти у свій кабінет /dashboard й заповнивши форму, додати новину разом із зображенням. Також реалізована примітивна реєстрація, логін та генерація контенту панелі керування в залежності від ролі (розраховано на те, що буде 3 ролі — адмін, учень та вчитель). Поки що такого розділення немає. Генерація html файлу сторінки відбувається за допомогою Jade template engine.

## 2. Система атоматичної збірки. Maven

Maven - це засіб автоматизації роботи з програмними проектами, який спочатку використовувався для Java проектів. Використовується для управління (management) та складання (build) програм. Створений Джейсоном ван Зилом (Jason van Zyl) у 2002 році. За принципами роботи кардинально відрізняється від Apache Ant, та має простіший вигляд щодо build-налаштувань, яке надається в форматі XML. XML-файл описує проект, його зв'язки з зовнішніми модулями і компонентами, порядок будування (build), папки та необхідні плагіни. Сервер із додатковими модулями та додатковими бібліотеками розміщується на серверах. Раніше Maven, де він був частиною Jakarta Project.

Для опису програмного проекту який потрібно побудувати (build), Maven використовує конструкцію відому як Project Object Model (POM), залежності від зовнішніх модулів, компонентів та порядку побудови. Виконання певних, чітко визначених задач - таких, як компіляція коду та пакетування відбувається шляхом досягнення заздалегідь визначених цілей (targets).

Ключовою особливістю Maven є його мережева готовність (network-ready).

Двигун ядра може динамічно завантажувати плагіни з репозиторію, того самого репозиторію, що забезпечує доступ до багатьох версій різних Java-проектів з відкритим кодом, від Apache та інших організацій та окремих розробників. Цей репозиторій та його реорганізований наступник, - Maven 2 репозиторій, - намагається бути де-факто механізмом для дистрибуції Java програм, але прийняття його в такій якості йде повільно.

Maven забезпечує підтримку побудови не просто перебираючи файли з цього репозиторію, але й завантажуючи назад артефакти у кінці побудови. Локальний кеш

звантажених артефактів діє як первісний засіб синхронізації виходу проектів на локальній системі.

### 3. Сервер безперервної інтеграції. Travis-ci

Як виявилось, термін «continuous integration» досить старий. Він був введений Мартіном Фаулером (Martin Fowler) у 2000-му році і викладений у статті «Continuous Integration» і по-російськи звучить як «безперервна інтеграція». Це частина процесу розробки, в якій розробляється проект збирається / тестується в різних середовищах виконання автоматично і безперервно. Задумувалася дана методика для найбільш швидкого виявлення помилок / протиріч інтеграції проекту, а отже зниження витрат на наступні простоти.

Принцип досить простий: на окремій машині працює якась служба, в обов'язки якої входить отримання вихідного коду проекту, його збірка, тестування, логування, а також можливість надати для аналізу дані виконання перерахованих операцій.

Виділити окремий сервер і підтримувати його в робочому стані, забезпечити наявність необхідних програмних комплексів, налаштувати середовища виконання, робити резервні копії даних і т.д... Все це вимагає чимало часу і ресурсів. І цілком логічним здається можливість делегувати цю відповідальність на сторонні сервіси. От якраз таким і є travis-ci - «хостинг безперервної інтеграції для open source співтовариства». Настав час подивитися на нього ближче.

Travis-ci підтримує безліч мов програмування. Почати користуватися сервісом дуже просто. Потрібно всього лише настроїти спеціальний файл конфігурації .travis.yml.

Задачі, які вирішуються на сервері безперервної інтеграції:

1. Запуск тестів:

- a) DAO тести
- b) REST тести

2. Побудова проекту в docker.

### 4. Експоненціальна витримка

Експоненціальна витримка реалізована і використовується у ситуації, коли адміністратор намагається додати зображення, але виникає помилка читання-запису у файл (блокування іншими процесами, тощо). В даному проекті, за експоненціальну витримку відповідальний Spring framework. За допомогою анотації

```
@Retryable(maxAttempts = 10, value= IOException.class, backoff = @Backoff(multiplier = 2, delay = 200))
```

метод викликається 10 разів, якщо у ньому виникає `IOException`. «Експотенціальність» досягається за рахунок вкзавування множника витримки (в даному випадку початкова - 200 мс, і щоразу подвоюється). Результати тестування наступні (вважаємо, що було 6 невдалих викликів):

```
-- Try to write open an input stream  
Called method for backoff - 2440  
Called method for backoff - 2641  
Called method for backoff - 3042  
Called method for backoff - 3843  
Called method for backoff - 5445  
Called method for backoff - 8647
```

Спостерігається експотенціальне збільшення інтервалу спроби.