

УДК 004.05

Мазурук О.В., студент 4 курсу
спеціальності «Комп'ютерні науки»
Антонов Ю.С., к.ф.-м.н., доцент, доцент кафедри
комп'ютерних наук та інформаційних технологій

ПРОЄКТУВАННЯ МІКРОСЕРВІСНОЇ АРХІТЕКТУРИ ДЛЯ ПІДСИСТЕМИ ВИБОРУ НАУКОВОГО КЕРІВНИКА

Донецький національний університет імені Василя Стуса, м. Вінниця

В кожному університеті студенти щорічно обирають наукових керівників для роботи над курсовими або кваліфікаційними роботами. Під час такого розподілу мають враховуватись інтереси як викладачів так і студентів та раціонально використовуватись особистий час. Станом на сьогодні існують роботи присвячені автоматизації діяльності університетів за такими напрямками як тестування знань, інформаційна підтримка навчального процесу, автоматизація діяльності різноманітних структурних підрозділів. Однак проблема обрання наукового керівника залишається не дослідженою та не автоматизованою [1-4]. Тому в даний час для університетів є актуальною - розробка підсистеми, одним із головних завдань якої є розподіл студентів і наукових керівників із урахуванням їх пріоритетів.

В якості базової організації системи, яка описує зв'язки між компонентами, було обрано мікросервісну архітектуру, оскільки: додатки можуть продовжувати працювати, навіть при несправності модуля чи компонента; кожний модуль або компонент не прив'язаний до конкретного технологічного стеку; функціональні сервіси містять мінімум коду, що спрощує входження для нових співробітників; оновлення додатку по частинам; відсутність спільного стану між модулями [5].

В першу чергу необхідно визначитись із переліком бізнес-задач, які необхідно реалізувати. До бізнес-задач можна віднести: авторизацію, аутентифікацію, CRUD-операції над опитуваннями, а також формування пар. Далі необхідно здійснити декомпозицію даних задач на декілька основних мікросервісів. Кожний мікросервіс буде відповідати за виконання певної бізнес-задачі.

Було виділено три основних мікросервіса (рис. 1):

- а) Auth-service – для авторизації користувачів та для захищеного спілкування між сервісами на основі OAuth2. З боку клієнта це нічим не відрізняється від традиційної авторизації за допомогою сесій.
- б) «Poll-services» – виконання CRUD операцій над опитуваннями покладено (для доступу необхіден токен від «Auth-service»).
- в) «Couting-Poll-Services» – формуванню пар науковий керівник - студент. По завершенні опитування цей сервіс буде формувати результат.

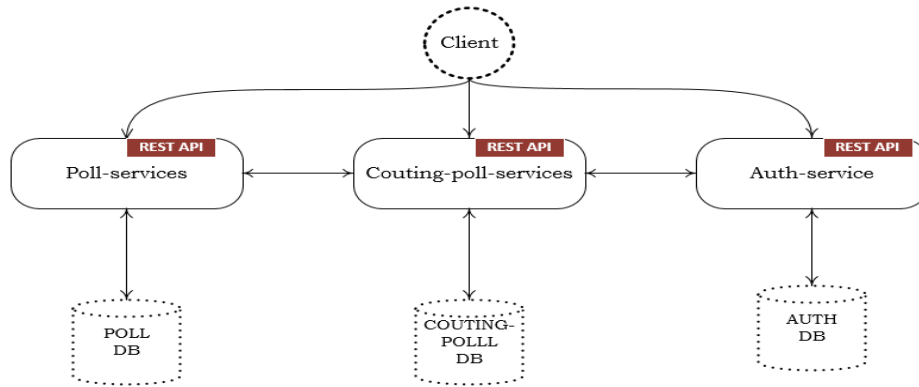


Рисунок4 – Функціональне розподілення мікросервісів

Для забезпечення спільної роботи описаних вище сервісів, було використано набір основних патернів і практик мікросервісної архітектури [6] та «SpringCloud»[7], та виділено наступні компоненти «SpringCloud» (Рис. 2):

- «ConfigServer» – горизонтально масштабоване сховище конфігурацій.
- «APIGateway» – надання зовнішнього Web-API.
- «Servicediscovery» – автоматичне визначення мережових адрес для доступних додатків.
- «Ribbon» – балансування навантаження на стороні клієнта.
- «Hystrix» – контроль над затримками і помилками при викликах по мережі.
- «Feign» – простий і гнучкий http-клієнт, нативно інтегрований з «Ribbon» і «Hystrix».
- RabbitMQ – брокер повідомлень що відповідає за доставку запитів і відповідей між сервісами та забезпечення збору метрик, для їх подальшого оброблення і візуалізації за допомогою «Hystrix dashboard».
- «Spring-BootAdmin» – інтерфейс користувача для моніторингу та управління додатком (сервісом) [8].

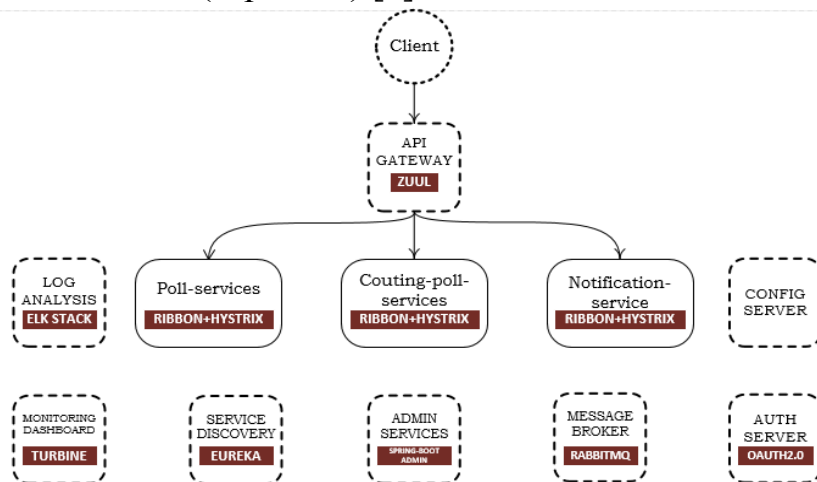


Рисунок 5 - Інфраструктурне розподілення мікросервісів

Отже, мікросервісна архітектура забезпечує покращення ізоляції при збоях компонентів, усуває залежність додатку від одного технологічного стеку,

зменшує поріг входження у проєкт для нових співробітників. У свою чергу фреймворк «Spring» реалізовує основні паттерни та практики мікросервісної архітектури.

Список використаної літератури

1. Астахова Н. А., Парамонова О. С., Парамонов А. И. Автоматизированная информационная система «Научный рейтинг». *Електротехнічні та комп'ютерні системи*. 2018. No 29 (105) С. 87-94.
2. Антонов Ю.С. Комп'ютерні системи тестування на основі технології трирівневих баз даних. *Інформаційні технології і засоби навчання*. 2008. Т.6, №2 URL: <https://journal.iitta.gov.ua/index.php/itlt/article/view/133> (дата звертання: 21.04.2021) <https://doi.org/10.33407/itlt.v6i2.133>
3. Мелєнтьєва М. В., Лебедєва О. Ю. Розробка чат-бота розкладу занять для месенджера TELEGRAM. *Комп'ютерні науки та інформаційні технології: матеріали семінару CSIT'2018 (м. Тернопіль, 2 червня 2018 р.)*. – Тернопіль: ТНЕУ, 2018. – С. 58-59. URL: <http://dspace.tneu.edu.ua/handle/316497/31976> (дата звертання: 21.04.2021)
4. Антонов Ю. С., Мулярчук О. П. Особливості розробки підсистем обліку академічної успішності студентів. *Матеріали наукової конференції професорсько-викладацького складу, наукових працівників і здобувачів наукового ступеня за підсумками науково-дослідної роботи за період 2017–2018 рр. (16–17 травня 2019 р.): у 2-х томах. Том 2 Вінниця: Донецький національний університет імені Василя Стуса*. С 106-107. URL: <http://jpv.s.donnu.edu.ua/article/view/7241> (дата звертання: 21.04.2021)
5. 7 reasons to switch to microservices — and 5 reasons you might not succeed. URL: <https://www.cio.com/article/3201193/7-reasons-to-switch-to-microservices-and-5-reasons-you-might-not-succeed.html> (дата звертання: 21.04.2021)
6. Микросервисы со SpringBoot, 2021. URL: <https://habr.com/ru/post/484130/> (дата звертання: 21.04.2021)
7. Introduction to Spring Cloud Netflix – Eureka, 2021. URL: <https://www.baeldung.com/spring-cloud-netflix-eureka> (дата звертання: 21.04.2021)
8. Spring Cloud Netflix: Ribbon, 2019. URL: <https://medium.com/@kirill.sereda/spring-cloud-netflix-ribbon-%D0%BF%D0%BE-%D1%80%D1%83%D1%81%D1%81%D0%BA%D0%B8-2ded121e1377> (дата звертання: 22.04.2021)

УДК 004.72

Наральник Б.Ю., студент 2 курсу
спеціальності 122 «Комп'ютерні науки»
Ніколюк П. К., д.ф.-м.н., професор, професор
кафедри комп'ютерних наук
та інформаційних технологій

БЛОКЧЕЙН ЯК ДЕЦЕНТРАЛІЗОВАНА СИСТЕМА ЦИФРОВОЇ ЕКОНОМІКИ

Донецький національний університет імені Василя Стуса, м. Вінниця