



Java Task | Final

Duration = **7 hours**

Deadline: **20.04.2019**

Поштовик "Роги та копита" після запровадження хеш-системи розширились у велику корпорацію. Перед ними постала проблема оновлення солі та алгоритму хешування для плавної міграції систем на нову конфігурацію.

1. Завдання:

Реалізувати можливість міграції старих значень хешей в нові значення.
Можливість автоматичної заміни алгоритма і солі хешування по запланованій даті міграції.
Забезпечити підтримку періода міграції. Коли система знаходиться в режимі міграції за головним посиланням буде генеруватися новий хеш, а по додатковій - можливість конвертації старого хеша в новий. Після міграції посилання для конвертації буде недоступною.

Вхідні дані:

- дата і час міграції БД хешей
- дата і час закінчення міграції БД хешей
- новий алгоритм хешування
- нова сіль хешування
- підтримка basic authentication
- старий хеш для отримання нового хеш-значення номера телефону.
Формат хеш-значення залежить від алгоритму хеширування

Вимоги:

1. RESTful API
2. Тип алгоритму хешування SHA-1, SHA-2, SHA-3. Тип алгоритму повинен конфігуруватися
3. Сіль для хеширування повинна конфігуруватися
4. Кількість номерів абонентів в БД не менше 50 000 000
5. Час першого запуску додатку не більше 4 хвилин
6. Час виконання одиночного запиту не повинен перевищувати 300 ms
7. Обробка хеш-коллізій

Очікуваний результат:

1. Серверний API з методом "Отримати нове хеш-значення номеру абонента по старому хеш-значенню номеру абонента"
2. Метод доступний до заданої дати закінчення міграції
3. Основні методи починаючи з дати міграції працюють вже з новими заданими параметрами (алгоритм хешування, сіль)

2. Формат представлення результатів

1. Рішення оформлюється за вказаними критеріями
 - 1.1. Рішення зберігається в окремій папці
 - 1.2. Назва папки це назва номінації
 - 1.3. Рішення архівується в архів з назвою у форматі **Ім'я_Прізвище.zip**
 - 1.4. Архів завантажується на сайті в особистому кабінеті

Зверніть увагу, що назва архіву - єдине місце, де ви вказуєте свої персональні дані.
Назви файлів всередині архіву не мають містити вашого імені чи прізвища.
Розмір архіву з рішенням не має перевищувати 10 MB.



DEV CHALLENGE BACKEND

2. README файл, в якому обов'язково вкажіть:

- 3.1. кроки для запуску сервісу для запуску сервісу
- 3.2. кроки для запуску тестів
- 3.3. опис API
- 3.4. речі, на які ви хотіли б звернути увагу або наступні кроки по вдосконаленню вашого рішення

3. scaleme file

4. Після запуску API повинно бути запущено за адресою - <http://localhost:8080/>.

Обов'язково перевіряйте запуск системи на іншому ПК перед надсиланням.

Проект повинен запускатись однією командою «docker-compose up» (Docker & Docker Compose).

5. Організатори та судді залишають за собою право дискваліфікувати роботу учасника, якщо робота:

- 5.1. Містить будь-яку вказівку на ім'я, прізвище, електронну пошту, компанію, адресу чи інші персональні дані учасника
- 5.2. Виконана у іншому форматі, ніж вказано у завданні
- 5.3. Виконана за допомогою сторонніх осіб, а не учасником особисто

3. Обмеження та критерії оцінювання

Категорія	Критерії	Бали
	Працездатність додатку	250
	Відповідність кодстайлу, структурованість, простота коду	75
	Система не потребує перезавантаження при застосуванні нових параметрів для міграції БД хешей	102
	Час виконання 99% запитів повинно не бути більше за 10 ms при 100 одночасних з'єднаннях	70
	multi-node серверний API	15
BONUS	UNIT тести	40
BONUS	Наявність SIT тестів	16
BONUS	Підтримка https	20
BONUS	Розподілена БД	8
BONUS	Наявність логування запитів	20
BONUS	Наявність API для отримання метрик системи	20

4. Переглянути завдання Онлайн Раунду можете [за посиланням](#)