|  |  |
| --- | --- |
| **ООО "СЕУСЛАБ"**  **ИНН 5903097803, КПП 590501001**  **ОГРН 1125903002675**  **614066, Пермский край, г. Пермь**  **ул. Шоссе Космонавтов 111 к3 оф.302**  **тел.: 8-800-333-09-17 / 8-(342)-207-11-70**  [**info@seuslab.ru**](mailto:info@seuslab.ru)**,** [**www.seuslab.ru**](http://www.seuslab.ru) |  |
| Исх. №642/18 от 17.09.2018 г.  [Предложение о сотрудничестве] |  |

**Уважаемый коллега!**

ООО «СЕУСЛАБ», российский разработчик технических решений, предназначенных для подразделений российских компаний, занимающихся мониторингом информационного пространства. Один из продуктов компании – средство для мониторинга социальной сети «ВКонтакте», «Поисковая система «СЕУС» (далее Программа «СЕУС»), которая представляет собой облачный сервис с доступом через сеть Интернет.

Методические материалы ООО «СЕУСЛАБ» использовались при подготовке учебных материалов по мониторингу социально опасных субкультур в России, Кыргыстане, Никарагуа.

**Ваше резюме на hh.ru заинтересовало нашу компанию, а лучше сказать - команду. Предлагаем выполнить тестовое задание, потому что нам важно понять уровень ваших навыков. Выполнить задание желательно в течение 2-3 дней.**

**Будем признательны, если Вы в ответном письме сообщите о своем решении или зададите вопросы** на почту [a.loshchilin@seuslab.ru](mailto:a.loshchilin@seuslab.ru)

**Тестовое задание на вакансию Java middle**

Реализовать систему состоящую из двух микросервисов. Микросервисы могут располагаться на разных серверах и должны взаимодействовать между собой по сети.

Микросервис1 работает с входящими данными(файл), отправляется запросы на обработку и формирует ответ.

Микросервис2 обрабатывает запросы взаимодействуя с Redis и БД

**Формирование данных**

**Данные формируются самостоятельно.**

**Реляционная база данных(postgresql/mysql):**

Есть минимум две таблицы

1. detail(id,redis\_id, name)
2. sub\_detail(id, detail\_id, sub\_name)

Между таблицами есть связь один ко многим detail.id <-> sub\_detail.detail\_id, связь можно организовать, как с промежуточной таблицей, так и без нее.

**Redis:**

В redis есть соотношение project\_name <-> id0, id1, .., idX

id0, id1, .., idX связаны с detail.redis\_id

**Входные/Выходные данные**

На вход подается csv формата(от 0 до 5 строк)

project\_name;time\_out\_in\_ms;

project\_name – имя проекта

time\_out\_in\_ms – время ожидания до обработки следующей строки

На выходе создается несколько файлов с описанием деталей

Формат:

project\_name\_detail\_name\_detail\_id – имя файла(таблица detail)

detail\_id, sub\_name – содержимое(таблица sub\_detail)

**Реализация:**

**Сервис1:**

Используемый язык: java

**Функционал:**

1. Чтение данных из входного файла
2. Отправка данные на обработку сервису два
3. Получение данные от сервиса два
4. Создать файл ответа
5. Реализация задержки во время отправки на обработку сервису два

**Сервис2:**

Используемый язык: java

**Функционал**:

1. Получение данные от сервиса одни
2. Работа с redis(Получение id деталей по project\_name)
3. Работа с БД(Получение состава деталей по id детали)
4. Ответ сервису один
5. Эмуляция задержки во время обработки запроса

**Дополнительно:**

Предусмотреть вариант восстановление после обрыва связи.

Отказ в обслуживании сервиса 2

**Будем плюсом:**

Реализовать отправку/получение и обработку данных в отдельных потоках.

Использовать ORM Hibernate.