

Тестовое задание

Необходимо написать сервис для “банковских” операций. В нашей системе есть пользователи (клиенты), у каждого клиента есть строго один “банковский аккаунт”, в котором изначально лежит какая-то сумма. Деньги можно переводить между клиентами. На средства также начисляются проценты.

Функциональные требования:

1. В системе есть пользователи, у каждого пользователя есть строго один “банковский аккаунт”. У пользователя также есть телефон и email. Телефон и или email должен быть минимум один. На “банковском счету” должна быть какая-то изначальная сумма. Также у пользователя должна быть указана дата рождения и ФИО.
2. Для простоты будем считать что в системе нет ролей, только обычные клиенты. Пусть будет служебный api (с открытым доступом), через который можно заводить новых пользователей в систему, указав логин, пароль, изначальную сумму, телефон и email (логин, телефон и email не должны быть заняты).
3. Баланс счета клиента не может уходить в минус ни при каких обстоятельствах.
4. Пользователь может добавить/сменить свои номер телефона и/или email, если они еще не заняты другими пользователями.
5. Пользователь может удалить свои телефон и/или email. При этом нельзя удалить последний.
6. Остальные данные пользователь не может менять.
7. Сделать АПИ поиска. Искать можно любого клиента. Должна быть фильтрация и пагинация/сортировка. Фильтры:
 - a. Если передана дата рождения, то фильтр записей, где дата рождения больше чем переданный в запросе.
 - b. Если передан телефон, то фильтр по 100% сходству.
 - c. Если передано ФИО, то фильтр по like форматом '{text-from-request-param}%'
 - d. Если передан email, то фильтр по 100% сходству.
8. Доступ к АПИ должен быть аутентифицирован (кроме служебного api для создания новых клиентов).
9. Раз в минуту баланс каждого клиента увеличиваются на 5% но не более 207% от начального депозита.
Например:
Было: 100, стало: 105.
Было: 105, стало: 110.25.
10. Реализовать функционал перевода денег с одного счета на другой. Со счета аутентифицированного пользователя, на счёт другого пользователя. **Сделать все необходимые валидации и потокобезопасной.**

Нефункциональные требования:

1. Добавить OpenAPI/Swagger

2. Добавить логирование
3. Аутентификация через JWT.
4. Нужно сделать тесты на покрытие функционала трансфера денег.

Стек:

1. Java 17
2. Spring Boot 3
3. База данных PostgreSQL
4. Maven
5. REST API
6. Дополнительные технологии (Redis, Elasticsearch и т.д.) на ваше усмотрение.
7. Фронтенд не нужен

Результат предоставить в виде публичного репозитория на github.