

Тестовое задание (Kotlin / Ktor)

Техническое задание

Общее описание

Существенная часть практической работы по анализу геномных данных связана с поиском отличий между генетическим материалом некоего образца и последовательностью референсного генома. Такие отличия называются генетическими вариантами, их наличие может свидетельствовать о изменении структуры белков (или функциональных РНК) организма, в свою очередь приводящих к изменениям фенотипа.

Характеристика генетических вариантов с точки зрения возможности их влияния на фенотип называется аннотированием, а информация, которая появляется в результате такого процесса – аннотацией.

Существует масса баз данных, выступающих в качестве источников аннотации (например, dbSNP, Clinvar, gnomAD и проч.). На практике часто требуется оперативно получить аннотацию для некоторого генетического варианта (например, при анализе последовательности генома конкретного человека). Разные источники по-разному организуют доступ к данным аннотации. Например, некоторые предоставляют веб-интерфейсы для поиска генетических вариантов и выдачи данных аннотации по ним (здесь следует отметить, что в таком случае речи не идет о каком-либо публичном API, который позволял бы получать эти данные программно), а некоторые идут вместе со специальными утилитами, устанавливаемыми локально (то есть подразумевается, что файлы базы данных должны быть скачаны и доступны с клиентской машины). Так как в биоинформатике существует устойчивая тенденция к переходу к веб-сервисам, такой функционал было бы удобно предоставить в виде HTTP сервиса.

Веб-сервис

HTTP сервис, предоставляющий информацию об аннотации некоторого генетического варианта в рамках данного источника аннотации (одной из соответствующих баз данных).

Реализация

Сервис предлагается создать на языке Kotlin, используя Ktor (фреймворк для написания веб-сервисов).

Техническое описание

- Клиентское приложение должно быть написано с помощью Kotlin и Ktor.
- Приложение должно предоставлять аннотацию для переданного генетического варианта.
- Приложение должно обрабатывать и соответствующим образом информировать пользователя в случае, когда по каким-либо причинам запрос не удалось обработать успешно.
- Код приложения должен быть доступен для удаленного просмотра и комментирования.
- Для обеспечения удобного просмотра и сопровождения прогресса в ходе разработки мы предлагаем работать через pull request. Т.е. создать новую ветку разработки на момент пустого репозитория (либо его базового состояния), в которой будет происходить процесс разработки, а затем создать pull request на слияние созданной ветки в master, в рамках которого мы сможем следить за процессом разработки, вести диалог.

Задачи

Этап 1:

1. Разобраться с предметной областью.
2. Предложить вариант архитектуры сервиса.
3. Составить план работы, оценить количество времени, необходимого для его осуществления.

Этап 2:

1. Непосредственная разработка приложения (написание кода, его обсуждение в рамках pull request).