ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»

Факультет компьютерных наук
Департамент программной инженерии

Подп. и дата	
Инв. № дубл.	
Взам. инв. №	
Подп. и дата	
Инв. № подл	

СОГЛАСОВАНО

НИУ ВШЭ, Приглашенный Преподаватель

_____ А. А. Топтунов
«_____» _____ 2024 г.

УТВЕРЖДАЮ

Академический руководитель образовательной программы «Программная инженерия» профессор департамента программной инженерии, канд. техн. наук

_____ Н. А. Павлочев
«_____» _____ 2024 г.

ВИЗУАЛИЗАЦИЯ СОЦИАЛЬНОГО ГРАФА ПОЛЬЗОВАТЕЛЯ TELEGRAM

Пояснительная записка

Лист Утверждения

RU.17701729.05.04-01 ПЗ 01-1-ЛУ

Исполнитель: Студент группы БПИ 213
_____ Н. А. Бирюлин
«_____» _____ 2024 г.

Москва 2024

Утверждён
RU.17701729.05.04-01 ПЗ 01-1-ЛУ

ВИЗУАЛИЗАЦИЯ СОЦИАЛЬНОГО ГРАФА ПОЛЬЗОВАТЕЛЯ TELEGRAM

Пояснительная записка

RU.17701729.05.04-01 ПЗ 01-1

Листов 12

Инв. № подл	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

Москва 2024

Содержание

1 Введение **3**
1.1 Наименование программы 3

2 Основание для разработки **4**
2.1 Документы, на основании которых ведется разработка 4

3 Технические характеристики **5**
3.1 Назначение программы 5
3.2 Алгоритм работы программы 5
3.3 Входные и выходные данные 6

4 Технические и программные средства **8**
4.1 Необходимые технические средства 8
4.2 Обоснование необходимых технических средств 8
4.3 Необходимые программные средства 8
4.4 Обоснование необходимых программных средств 8

5 Ожидаемые технико-экономические показатели **9**
5.1 Экономическая эффективность 9
5.2 Предполагаемая потребность 9
5.3 Преимущества разработки по сравнению с отечественными и зарубежными аналогами 9

6 Приложение 1. Терминология **10**

1. Введение

1.1. Наименование программы

1.1.1. Наименование программы на русском языке

Визуализатор социального графа «Telegram».

1.1.2. Наименование программы на английском языке

«Telegram» social graph visualizer.

2. Основание для разработки

2.1. Документы, на основании которых ведется разработка

Учебный план подготовки бакалавров по направлению 09.03.04 «Программная инженерия» и утвержденная академическим руководителем программы тема курсового проекта.

3. Технические характеристики

3.1. Назначение программы

3.1.1. Функциональное назначение программы

Функциональным назначением программы является сводный и реляционный анализ данных, экспортируемых из аккаунтов Telegram [11].

3.1.2. Эксплуатационное назначение программы

Приложение позволяет находить связи между пользователями в выгрузке сообщений из Telegram. После импорта .json файла с данными, пользователь приложения может получить для заданного списка пользователей Telegram граф их связей. Под связями подразумеваются как прямые (A писал B), так и косвенные (A и B писали в один чат, A упоминал B в сообщении, A писал в чат в котором упоминался B).

3.2. Алгоритм работы программы

Перед проведением анализа пользователь выбирает экспортированный из Telegram .json файл на своем устройстве. Этот файл передается в веб-приложение. При этом файл не загружается на удаленный сервер - вся обработка производится на устройстве. Это предоставляет приложению следующие преимущества:

- Анализ и визуализация данных проходят максимально быстро - нет необходимости загружать наборы данных, достигающие гигабайт, на удаленный сервер, и загружать с него сгенерированные графы.
- Приложение максимально устойчиво к нагрузке. На серверную часть нагрузка сводится к обеспечению доступа к нескольким статическим файлам, что обеспечивает устойчивость работы даже на сравнительно маломощном оборудовании.
- Данные не покидают устройство пользователя. Это гарантирует анонимность, приватность и предотвращает компрометацию аккаунта.

После загрузки вычисляется граф зависимостей, имеющий следующую структуру:

```
class Graph {
    chatIdToName = {};
    chatNameToId = {};

    actorIdToName = {};
    actorNameToId = {};

    chatIdToMessages = {}; // Messages sent
    actorIdToMessages = {};

    actorToChatMapping = {}; // User writing something in chat. [user_id][chat_id] = [
        messages]
    actorToActorMapping = {}; // User mentioning another user. [user_a_id][user_b_id] = [
        messages]
```

```
    chatCategories = {};
}
```

Собираются отношения Telegram ID к видимым именам для всех объектов (пользователи, чаты, каналы). Они используются для упрощения взаимодействия с пользователем (автодополнение, имена в графах вместо ID).

Кроме того, сообщения группируются по чатам и по пользователям. На этом этапе строится базовая связь чатов с пользователями (пользователь писал в чат) и пользователей с пользователями (пользователь упоминал пользователя в одном из своих сообщений). Стоит заметить, что построение связей упоминаний пользователей в может найти интересные связи, но при этом приводит к созданию в графе несуществующих связей. Например, при тестировании обнаружилось, что очень много пользователей связаны с одним конкретным пользователем. Причина была в том, что имя пользователя - “Да”, что, конечно, много где упоминалось в переписке.

Данные вычисляются один раз, после чего построение графа использует заранее просчитанный набор связей. Это позволяет значительно уменьшить задержку при взаимодействии с пользователем и сделать скорость построения графа незначительной. Так, даже если время импорта файла будет порядка 20-30 секунд, непосредственно построение графа всегда будет измеряться в миллисекундах.

При непосредственно построении графа мы уже имеем заранее просчитанную информацию о чатах каждого из пользователей, так что задача сводится к поиску общих в массивах. В совокупности с использованием производительной библиотеки для визуализации данных (Apache ECharts [10]) это позволяет приложению почти мгновенно рисовать графы любого размера.

3.3. Входные и выходные данные

3.3.1. Организация входных данных

Входные данные - .json файл, экспортированный из приложения Telegram [12].

3.3.2. Обоснование метода организации входных данных

Для подобного анализа требуется, с одной стороны, проанализировать большой набор данных (речь может идти о миллионах сообщений), с другой стороны, сделать это без потери доверия пользователя (если пользователь не доверяет сервису, он не будет загружать туда свою личную переписку).

Анализ экспорта позволяет нам получить все сообщения не взаимодействуя с Telegram напрямую (а такое большое количество запросов к API могло бы привести к блокировке), что делает сервис надежнее и быстрее. Более того, разница по скорости - не просто вопрос нескольких секунд: при прямом взаимодействии с Telegram речь идет о замедлении как минимум на несколько порядков, т.к. Telegram ограничивает количество запросов, которые могут быть выполнены сторонними клиентами.

Кроме того, так пользователю не нужно давать доступ к своему Telegram-аккаунту (что, для большинства пользователей, было бы неприемлемо). В совокупности с обработкой данных исключительно на устройстве пользователя это обеспечивает максимальный уровень анонимности и приватности.

3.3.3. Организация выходных данных

Базовая статистика (количество пользователей в экспорте, контактов, базовая информация о аккаунте) выводится в текстовом виде. Анализ отношений пользователей выводится на экран

в виде интерактивного графа, отображающего пользователей и ассоциированные с ними чаты. В качестве дополнительной информации на графе по разному (тип и цвет объекта) отображаются разные элементы графа: пользователи, личные чаты, публичные и приватные группы, каналы. С каждым объектом указывается его уникальный Telegram ID.

3.3.4. Обоснование метода организации выходных данных

Базовая статистика - набор из нескольких чисел и имен, простое текстовое представление для нее ожидаемо, нет повода от него отказываться.

Представление связей между пользователями в виде интерактивного графа мотивированно доступностью пользователям: на маленьких примерах и статическая картинка, и интерактивный вариант, который можно подвигать или приблизить могут быть достаточно удобны, но на больших, запутанных схемах интерактивность значительно улучшает пользовательский опыт. Цветовая маркировка различных элементов улучшает читаемость графа, особенно в случаях конфликта имен (например, так проще отличить пользователя “Котенок” и канал с картинками с аналогичным названием). Указание Telegram ID решает случаи конфликтов, когда объекты одного типа имеют одинаковое имя (например, два разных пользователя, но оба с именем “Олег”).

4. Технические и программные средства

4.1. Необходимые технические средства

- Персональный компьютер с доступом в интернет
- От 8 GB оперативной памяти
- Соответствие прочим техническим требованиям для используемых программных средств (браузера)

4.2. Обоснование необходимых технических средств

Приложение представляет из себя веб-страницу, его использование без доступа в интернет невозможно. Требование использования приложения с персонального компьютера вытекает напрямую из характера выходных данных: работа с графами, особенно большими, значительно затруднена при работе с мобильного устройства без полноразмерного экрана. Требование к количеству оперативной памяти мотивированно размером входных данных: даже для сравнительно малоиспользуемого аккаунта размер анализируемых данных может достигать гигабайта. При этом при обработке приложение потребляет в среднем в 1.5-2 раза больше оперативной памяти, чем размер самого входного файла.

4.3. Необходимые программные средства

На устройстве пользователя должен быть установлен один из совместимых браузеров:

1) Firefox версии не менее 122.0
2) Chrome версии не менее 121.0
3) Яндекс.Браузер версии не менее 21.0
4) Microsoft Edge версии не менее 121.0

4.4. Обоснование необходимых программных средств

Несмотря на то, что непосредственно код программы не использует функционала, требующего самых новых версий браузеров, используемая библиотека для визуализации графов [10] работает быстрее на актуальных версиях браузеров. Кроме того, использование актуальных версий браузеров гарантирует отсутствие проблем совместимости с программным кодом и повышает скорость работы приложения.

5. Ожидаемые технико-экономические показатели

5.1. Экономическая эффективность

Расчёт экономической эффективности в рамках работы не предусмотрен.

5.2. Предполагаемая потребность

У активных пользователей Telegram могут быть тысячи контактов, которые пересекаются через разные чаты и каналы. Понять, откуда ты знаешь этого человека, или пересекались ли люди в такой ситуации - сложная, но регулярно необходимая задача. Такой инструмент позволяет строить потрет человека на основании его публичных сообщений (я вижу, что он активно обсуждал в чатах манулов), или оценивать его связи с другими людьми (я вижу, что и Алиса, и Боб регулярно пишут в чат по криптографии).

5.3. Преимущества разработки по сравнению с отечественными и зарубежными аналогами

В свободном доступе существуют только инструменты для вычисления агрегированной статистики по экспорту данных. Ни один общедоступный (включая коммерческие) инструмент не подразумевает анализ связей отдельных пользователей, на который нацелено разрабатываемое приложение. В такой ситуации приложение занимает нишу без прямых конкурентов.

6. Приложение 1. Терминология

1) JSON - “Java Script Object Notation”, текстовый формат хранения данных, используемый, в частности, в входных данных приложения.

2) (Apache) ECharts - открытая библиотека для визуализации данных, в том числе графов [10].

3) Telegram ID - уникальный идентификатор пользователей и чатов, используемый Telegram. Имеет вид “user0000000” или “0000000”.

4) Экспорт (Telegram) - набор данных, экспортированный из клиента Telegram [12]. Имеет вид папки с .json-файлом и медиафайлами. Для работы приложения необходим только .json-файл.

5) Видимое имя (Telegram) - комбинация имени и фамилии для пользователя, название для чатов и каналов.

Список литературы

[1] ГОСТ 19.101-77. ЕСПД. Виды программ и программных документов. — М.: ИПК Издательство стандартов, 2001.

[2] ГОСТ 19.201-78. ЕСПД. Техническое задание. Требования к содержанию и оформлению. — М.: ИПК Издательство стандартов, 2001.

[3] ГОСТ 19.301-79. ЕСПД. Программа и методика испытаний. Требования к содержанию и оформлению. — М.: ИПК Издательство стандартов, 2001.

[4] ГОСТ 19.401-78. ЕСПД. Текст программы. Требования к содержанию и оформлению. — М.: ИПК Издательство стандартов, 2001.

[5] ГОСТ 19.404-79. ЕСПД. Пояснительная записка. Требования к содержанию и оформлению. — М.: ИПК Издательство стандартов, 2001.

[6] ГОСТ 19.505-79. ЕСПД. Руководство оператора. Требования к содержанию и оформлению. — М.: ИПК Издательство стандартов, 2001.

[7] ГОСТ 19.504-79. ЕСПД. Руководство программиста. Требования к содержанию и оформлению. — М.: ИПК Издательство стандартов, 2001.

[8] ГОСТ 19.106-78 Требования к программным документам, выполненным печатным способом. — М.: ИПК Издательство стандартов, 2001.

[9] ГОСТ 15150-69 Машины, приборы и другие технические изделия. Исполнения для различных климатических районов. Категории, условия эксплуатации, хранения и транспортирования в части воздействия климатических факторов внешней среды. — М.: Изд-во стандартов, 1997.

[10] ECharts [электронный ресурс] Режим доступа: echarts.apache.org, свободный (дата обращения 22.03.24)

[11] Telegram [электронный ресурс] Режим доступа: telegram.org, свободный (дата обращения 22.03.24)

[12] Telegram Data Export [электронный ресурс] Режим доступа: telegram.org/blog/export-and-more, свободный (дата обращения 22.03.24)

Лист регистрации изменений

Изм.	Номера листов				Всего листов в документе	№ документа	Входящий № сопровод. докум. и дата	Подпись	Дата
	измененных	замененных	новых	аннулированных					