**Правительство Российской Федерации**

**Федеральное государственное автономное образовательное**

**учреждение высшего профессионального образования**

**"Национальный исследовательский университет**

**"Высшая школа экономики"**

**Отделение программной инженерии**

**Кафедра Управления разработкой программного обеспечения**

УТВЕРЖДАЮ

Зав. кафедрой УРПО

\_\_\_\_\_\_\_\_\_\_\_ С.М. Авдошин

«\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2014 г.

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА**

**по направлению 231000.62 Программная инженерия**

**подготовки бакалавра**

На тему: «Клиент-серверное веб-приложение для визуализации и оценки характеристик социального графа пользователя ВКонтакте»

|  |
| --- |
| Руководитель работы  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ /Буров В.В./  «\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2014 г. |
| Исполнитель: студент группы 471 ПИ  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ /Рогов П. В./  «\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2014 г. |

Москва, 2014 г.

# Реферат

|  |  |
| --- | --- |
| **Количество страниц:** | 31 страница |
| **Количество глав:** | 3 главы |
| **Количество иллюстраций:** | 23 иллюстрации |
| **Количество использованных источников:** | 16 источников |

**Ключевые слова:** теория графов, социальные сети, визуализация графов, force-directed алгоритмы, анализ социальных сетей.

Объектом разработки является клиент-серверное веб-приложение для визуализации и расчета характеристик социального графа пользователя сети ВКонтакте.

Целью работы является разработка веб-приложения, которое а) обеспечит автоматический сбор социально-демографических данных пользователей сети ВКонтакте, б) построит на веб-странице двухмерное представление социального графа друзей указанного пользователя, пригодное для дальнейшего анализа, и в) вычислит значения характеристик графа, применяемых в ходе анализа социальных сетей.

В ходе данной работе были рассмотрены элементы теории графов, алгоритмы визуализации графов и элементы анализа социальных сетей, применимые для обработки данных пользователей социальной сети ВКонтакте.

В результате выполнения работы было создано веб-приложение, обеспечивающее автоматическое получение персональных данных и списков друзей пользователей сети ВКонтакте и строящее на их основе социальный граф, а также производящее его визуализацию в качестве векторного изображения в формате SVG и расчет социальных характеристик.

Содержание

[Введение 4](#_Toc389131191)

[1. Анализ предметной области 6](#_Toc389131192)

[1.1 Основные понятия 6](#_Toc389131193)

[1.2 Анализ социальных сетей 6](#_Toc389131194)

[1.3 Обзор существующих решений 8](#_Toc389131195)

[1.4 Выявление требований к системе 10](#_Toc389131196)

[2. Алгоритмы визуализации графов 12](#_Toc389131197)

[2.1 Общие сведения 12](#_Toc389131198)

[2.2 Способы и критерии визуализации 12](#_Toc389131199)

[2.3 Виды алгоритмов визуализации неориентированных графов 15](#_Toc389131200)

[2.4 Force-directed алгоритмы 16](#_Toc389131201)

[3. Особенности реализации программы 20](#_Toc389131202)

[3.1 Клиент 20](#_Toc389131203)

[3.2 Сервер 23](#_Toc389131204)

[3.3 Визуализация 24](#_Toc389131205)

[3.4 Расчет характеристик 27](#_Toc389131206)

[3.5 Технические требования 28](#_Toc389131207)

[Заключение 29](#_Toc389131208)

[Список использованных источников 30](#_Toc389131209)

# Введение

Социальные сети как группы связанных и коммуницирующих друг с другом людей существовали всегда. В социологии этот феномен выделился и стал изучаться задолго до появления интернета. Однако именно специализированные интернет-ресурсы, появившиеся сравнительно недавно, позволили в одночасье собрать огромную базу социально-демографических данных, ценных для изучения. Результаты исследований данных, полученных из виртуальных социальных сетей, могут быть мгновенно использованы в маркетинговых компаниях, подборе персонала, криминалистике и других практических задачах. MySpace и LinkedIn были запущены в 2003 году, Facebook в 2004, Twitter и ВКонтакте в 2006 (Гобл, 2012). В соответствии с данными портала Alexa.com Facebook и YouTube в находятся в тройке самых посещаемых сайтов в мире. Таким образом, социальные сети стали весьма авторитетным и стремительно развивающимся медиа каналом. Пользователи общаются в социальных сетях, делятся персональным контентом и публикуют интересные новости, компании продвигают свои продукты, а звезды привлекают внимание свежими фотографиями и слухами. Все эти данные представляют собой колоссальный объем социально-демографической информации, которая очень ценна для аналитиков и маркетинговых экспертов.

В основе социальных сетей лежит понятие социального графа. Это термин на стыке дискретной математики и социологии. Социальный граф представляет собой множество вершин, представляющих участников сети и различные виды социальных связей между ними. Возможны два вида связей: «явные», такие как друзья или коллеги, и «неявные», например, такие как фанаты одного футбольного клуба (Шерман, 2011). Все пользователи с их историей переписки, фотографиями, видеофайлами и персональными данными хранятся в базах данных распределенных серверных кластеров. Например, работу Facebook в 2012 году обеспечивали около 180 тысяч серверов (Миллер, 2012). Для использования терабайт социально-демографических данных в научных и коммерческих целях граф должен быть корректно построен, и его характеристики должны быть оценены.

Целью выпускной квалификационной работы является разработка клиент-серверного веб-приложения. У него есть три главных задачи:

1. Получение социально-демографической информации пользователей из сети ВКонтакте.
2. Визуализация данных на плоскости в виде, пригодном для дальнейшего анализа.
3. Расчет характеристик социального графа.

В работе освящены темы касательно математических основ анализа социальных сетей, алгоритмов визуализации графов, а также технические аспекты работы с API ВКонтакте и программной реализации системы.

# Анализ предметной области

## Основные понятия

Теория графов – это раздел дискретной математики, изучающий графы и их свойства. Граф G – это пара (V, E), где V = {v1, v2, … vn} – множество вершин, а C:\Users\764\Desktop\E.PNG- множество ребер графа [4, c. 5]. Однако применительно к компьютерным дисциплинам вместо графа чаще используется термин «сеть».

Говоря о графах необходимо ввести понятия «ориентированный» и «неориентированный» граф. Если ребра не ориентированы и пара (u,v) = (v, u), то граф называется неориентированным. В случае, когда пары упорядочены и (u, v) != (v, u), граф называется ориентированным или орграфом.

Графы бывают взвешенными. Во взвешенных графах в соответствии каждому ребро ставится число, обозначающее некую величину, например, пропускную способность канала в сети.

Теория графов позволяет моделировать объекты во многих областях науки. В числе основных дисциплин, активно использующих математических аппарат теории графов: химия и молекулярная биология, компьютерные дисциплины и сетевые технологии, транспортные системы и логистика, экономика, проектирование микросхем и социология.

## Анализ социальных сетей

Анализ социальных сетей (АСС или “social network analysis / SNA” в англоязычной литературе) представляет собой набор методов, позволяющих получать ценную информацию об особенностях социальных сетей по их характеристикам. В аспекте АСС социальные связи, такие как «друзья», «коллеги», «родственники», представляют собой ребра, связывающие отдельных людей, служащих узлами в сети (Бриггс, 2011)

АСС имеет две главные задачи: 1) определение количественных и качественных метрик, позволяющих провести всеобъемлющий анализ социальной сети, и 2) визуализация ее в удобном для исследователя виде. Существует три типа значимых метрик социальной сети (Абрахам, 2009). Первый включает в себя пять видов «характеристик связи»: *гомогенность* – склонность участников организовывать связи с другими участниками, имеющие схожие свойства, *множественность* – число различных видов связей между участниками, *обоюдность/взаимность* – степень, в которой двое участников отвечают друг другу взаимностью в социальных связях, *закрытость* – вероятность участников иметь прямую связь друг с другом внутри сети, *соседство* – склонность участников образовывать связи с географически близкими себе.

Второй тип состоит из шести «характеристик распределения»: *мост* – индивид, служащий единственной связью между отдельными кластерами сети, *центральность* – набор параметров, описывающих «уровень влиятельности» конкретного участника, *плотность* – отношение между реальным числом связей и максимально возможным при данном количестве вершин, *расстояние* – минимальное количество связей между двумя участниками сети, *структурные пробелы* – отсутствие пути между двумя кластерами сети, *сила связи* – количественная характеристика связи.

Наконец третий набор – это три «характеристики сегментации»: *количество клик* (социальных групп), *коэффициент кластеризации* – вероятность, что два узла, связанные с третьим, будут связаны друг с другом, *сплоченность* – степень, в которой узлы образуют прямые связи друг с другом.

Перечень характеристик, которые могут быть рассчитаны для конкретной сети, полностью зависит от количества известной о сети информации (Расселл, 2011). Второй компонент анализа социальных сетей – визуализация графов, – будет подробно рассмотрен во второй главе.

## Обзор существующих решений

На данный момент существует несколько программных продуктов для анализа социальных сетей.

Таблица 1. Сводная таблица программных систем и библиотек для проведения анализа социальных сетей

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Название** | **Функционал** | **Входной формат** | **Выходной формат** | **Платформа** | **Условия распространения** |
| AllegroGraph | База графов. Визуализация RDF | RDF | EDF | Linux, Mac, Windows | Free и Commercial |
| EgoNet | Анализ эгоцентричных сетей | XML | CSV | Любая система с Java | Open Source |
| Gephi | Исследование и изменение графов | DOT, GML. GDF, GRAPHML, NET, GEXF, CSV, Database | GDF, GEXF, SVG, PNG | Любая система с Java 1.6 и OpenGL | Open Source (GPL3) |
| GraphStream | Библиотека работы со статическими и динамическими графами | DGS, DOT, GML, Edge list | DGS, DOT, GML, Images | Любая система с Java | Open Source |
| Graph-tool | Python-модуль для анализа и визуализации графов | DOT, GraphML | DOT, GraphML, BML, CANON, CMAP, EPS, FIG, GD, GD2, GIF, GTK, ICO, IMAP, CMAPX, ISMAP, JPEG, PDF, PLAIN. PNG, PS, PS2, SVG, SVGZ, TIF, VML, VMLZ, VRML, WBMP XLIB | GNU/Linux, MAC | Free (GPL3) |
| Graphviz | Визуализация графов | DOT | BMP, CANON, CMAP, EPS,FIG, GD, GD2, GIF, GTK, ICO, IMAP, CMAPX, ISMAP, JPEG, PDF, PLAIN, PNG, PS, PS2, SVG, SVGZ, TIF, VML, VMLZ, VRML, WBMP, XLIB | Linux, Mac, Windows | Open Source (CPL) |
| sigma.js | Библиотека для визуализация графов | GEXF, JSON, XML | GEXF, XML | Поддержка JavaScript, HTML5 и WebGL | Open Source (MITL) |
| Mathematica | Анализ графов, расчет статистических данных визуализация данных, оптимизация, распознавание изображение | Более 50 форматов | Более 50 форматов | Windows, Mac, Linux | Commerical |
| Wolfram Alpha | Анализ графов и временных выборок | Facebook API | Более 20 форматов | Современные браузеры | Free и Commercial |

Сравнительный анализ проводился по следующим критериям: функциональность программы или библиотеки, число поддерживаемых входных и выходных форматов, поддерживаемые платформы и условия распространения.

В результате сравнительного анализа была выявлена следующие закономерности:

* Подавляющее большинство ПО предназначены для выполнения на персональном компьютере
* Они требуют заранее подготовленные данные для исследования (за исключением Wolfram Alpha, который использует Facebook API)
* Большинство систем унифицировано и не имеет специальных опций для анализа именно социальных сетей
* Менее половины систем сочетают в себе возможности как для визуализации, так и для анализа графов

## Выявление требований к системе

В результате анализа существующих на рынке программных систем был выявлен ряд требований к разрабатываемому клиент-серверному приложению.

Перечень функциональных требований:

* Сбор данных пользователя социальной сети ВКонтакте, используя VK API (до 1000 друзей)
* Визуализация списка его друзей в качестве социального графа
* Расчет социальных характеристик, описанных в параграфе 3.4, для построенной сети

Перечень технических требований:

* Реализация в виде клиент-серверного приложения
* Стабильная работа на компьютерах с операционными системами Windows 7/8, Mac OS X, Linux при использовании последних версий браузеров Google Chrome, Mozilla Firefox, Яндекс.Браузер, Opera (число друзей до 1000 человек)
* Хранение данных в базе данных MySQL
* Реализация клиента на HTML/CSS с использованием JavaScript с библиотеками с использованием SVG
* Реализация сервера на PHP, выполняемом на Apache-сервере

Программа, удовлетворяющая всем вышеперечисленным требованиям, может представлять практическую ценность для исследователей и специалистов по работе с социальными сетями, так как позволит значительно снизить время на сбор, обработку и подготовку данных к анализу.

# Алгоритмы визуализации графов

## Общие сведения

Визуализацией графа называется построение двухмерного (реже для специфических исходных данных - трехмерного) представления его ребер и вершин. Для визуализации графов существуют различные алгоритмы. Они отличаются друг от друга требованиями к графу (взвешенные/невзвешенные ребра, ограничение на максимальное число вершин), скоростью получения конечного результата, видом представления графа, которое будет построено по окончанию работы, а главное - целью, с которой они применяются.

Визуализация графов находит применение в различных практических отраслях: программная инженерия (диаграммы сущностей, потоков данных, маршрутов), управление проектами, представление структурированных знаний и данных, проектирование микросхем, социологические исследования и др [8, c. 3].

## Способы и критерии визуализации

Поскольку существует большое количество видов графов, для их отображения используются различные способы [8, c. 4]:

* Произвольное:

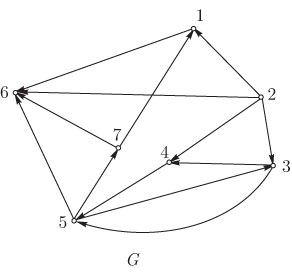


Рисунок 1. Граф, изображенный произвольным способом

* Полигональное – в нем для отображения ребер используются ломаные:

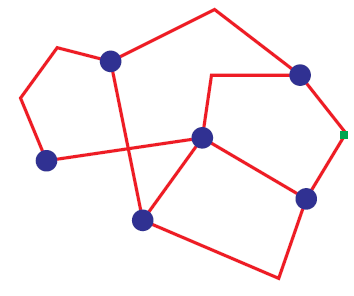


Рисунок 2. Граф, изображенный полигональным способом

* Прямолинейное – ребра изображаются отрезками:

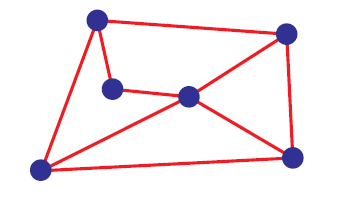


Рисунок 3. Граф, изображенный прямолинейным способом

* Ортогональное – ребра – ломаные, состоящие из вертикальных и горизонтальных отрезков:

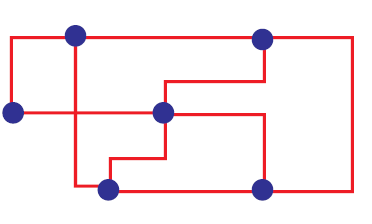


Рисунок 4. Граф, изображенный ортогональным способом

* Сеточное – вершины расположены в узлах сетки:

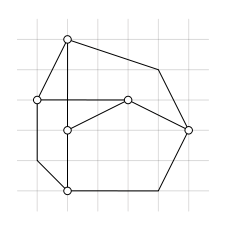


Рисунок 5. Граф, изображенный сеточным образом

* Восходящее и нисходящее – применяется для изображения ориентированных графов. Показывает «поток» - направлением движения по графу (например, сверху вниз). Также этот способ позволяет наглядно визуализировать иерархические отношения (например, генеалогическое древо) вершин:

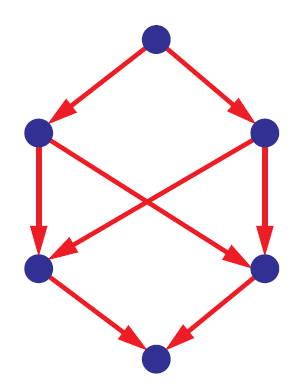
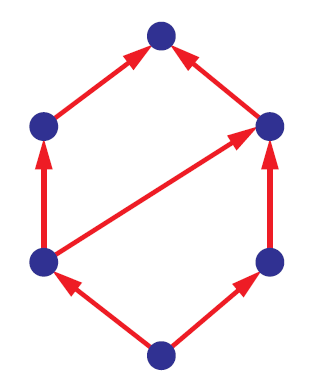


Рисунок 6. Граф, изображенный восходящим / нисходящим способом

Помимо выбора способа визуализации графа необходимо учесть ряд эстетических критериев [8, с. 11]:

* Минимизация числа пересечений ребер
* Минимизация пространства занимаемого изображением графа (размеров областей графа)
* Максимизация минимального угла между ребрами в графе
* Универсальность длин ребер (минимизация различий между ними)
* Минимизация числа изгибов ортогональных и наклонов полигональных отображений
* Максимизация симметрии

Чем большее число этих критериев соблюдено в ходе построения представления графа на плоскости или в трехмерном пространстве, тем доступнее будет его дальнейший анализ.

## Виды алгоритмов визуализации неориентированных графов

Поскольку социальный граф друзей пользователя ВКонтакте является неориентированным, то в этом разделе будет дано кратное описание алгоритмов, работающих именно с неориентированными графами. Существуют три базовых подхода к визуализации неориентированных графов: метод «планаризации», метод «ориентации» и метод «направленных сил» [8, c. 63].

Суть метода «планаризации» в том, что если граф не является планарным (то есть имеет пересечения ребер в своей двухмерной проекции), он искусственно делается им. Для этого в точках пересечения ребер расставляются искусственные «псевдо»-вершины, а затем к графу может быть применен один из методов построения планарного графа.

Метод «ориентации» основан по тому же принципу: производится искусственное преобразование неориентированного графа в ориентированный, после чего к графу может быть применен один из методов визуализации ориентированных графов.

В этой работе мы не будет подробно останавливаться ни на тех, ни на других вследствие того, что для визуализации социальных графов, представляющих фрагменты реальных социальных сетей, повсеместно применяется третий вид алгоритмов – метод «направленных сил», или «Force-directed method/approach» в англоязычной литературе. В его основе лежит следующий принцип. Во-первых, задается случайное начальное состояние физической системы, состоящей из пружин (ребер) и металлических колец (вершин). Деформированные пружины приводят систему в движение. Происходят колебания, приводящие к изменению деформации пружин и положения колец-вершин. Когда система достигнет минимального энергетического состояния, работа алгоритма будет завершена (Кобуров, 2012).

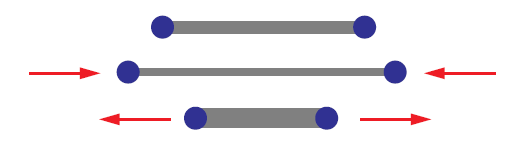


Рисунок 7. Сжатия и растягивания пружин

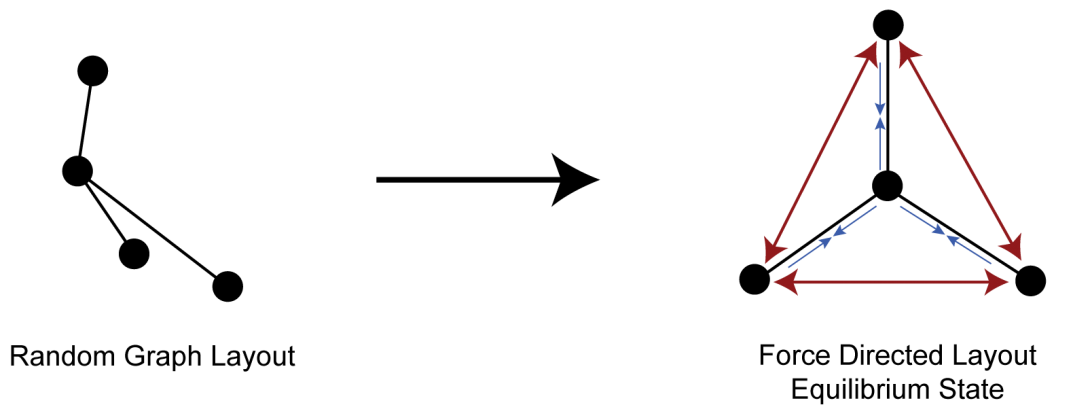


Рисунок 8. Демонстрация работы метода «направленных сил»

В следующем пункте будет приведен перечень различных алгоритмов визуализации графа, использующих метод «направленных сил».

## Force-directed алгоритмы

Алгоритм «Frushterman-Reingold» был разработан в 1991 году Томасом Фруштерманом и Эдвардом Рейнгольдом (Фруштерман и Рейгольд, 1991). Сложность алгоритма O (N^2). Он позволяет эффективно строить двумерные представления графов, в которых содержится не более 1000 вершин. Для построения не используется вес ребер.

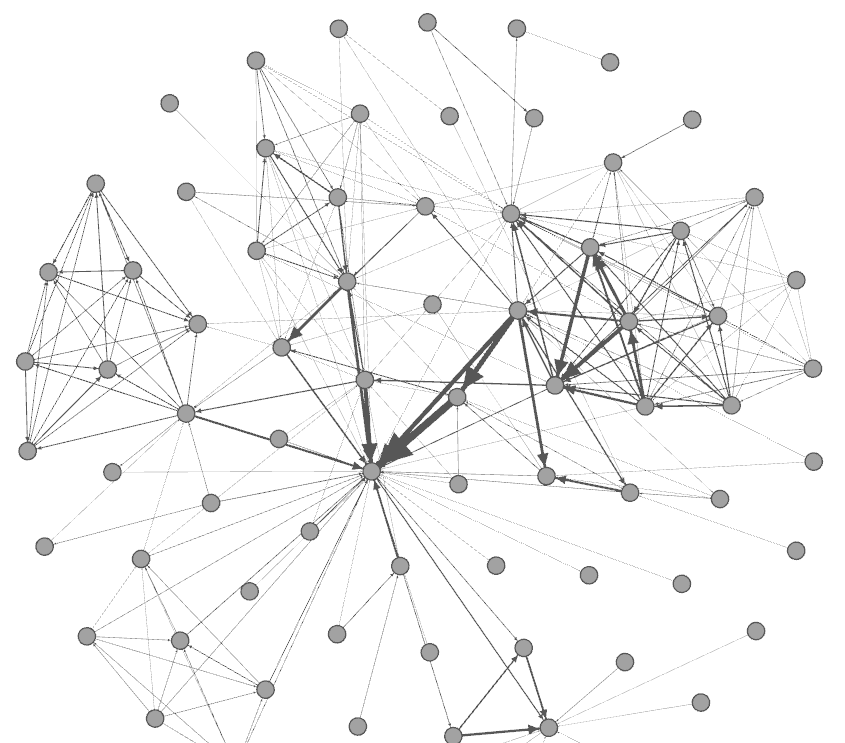


Рисунок 9. Результат работы алгоритма «Frushterman-Reingold»

Алгоритм «Yifan Hu Multilevel» был создан Йфаном Ху в 2005 году (Йфан, 2005). Его сложность O (N \* log(N)). Ограничение на размер графа: 100 – 100 000 вершин. Также, как и в предыдущем алгоритме, вес ребер не задействован. Алгоритм работает значительно быстрее предшественника, так как для расчета отталкивающих сил вершин используются силы отталкивания кластеров от вершин. В силу роста производительности алгоритм теряет в точности визуализации. Еще одной его особенностью является автоматическая остановка по достижении колебаний минимального порогового значения.

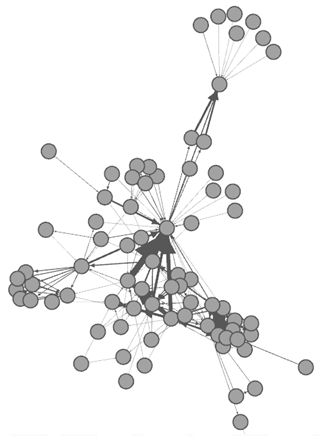


Рисунок 10. Результат работы алгоритма «Yifan Hu Multilevel»

Алгоритм «Force Atlas» разработан создателями Gephi в 2007 для визуализации безмасштабных сетей (графы, в которых степени вершин распределены по степенному закону) (Джакоми, Хэйман, Вентурини, Бастиан, 2007). Сложность составляет O (N^2), что позволяет обработать графы с числом вершин от 1 до 10 000 (именно такие ограничения имеет число друзей пользователя ВКонтакте). Если ребра графа имеют вес, он будет учтен при построении. При проектировании «Force Atlas» был сделан акцент на качестве визуализации, что делают раскладку графа, получающуюся на выходе максимально наглядной.



Рисунок 11. Результат работы алгоритма «Force Atlas»

Эмпирически было установлено, что многие естественно возникающие сети (такие как социальные, коммуникационные и биологические графы) хорошо моделируются безмасштабными сетями, то есть сетями, в которых степени вершин распределены по степенному закону (Бонатоа, Хадиб, Хорнк, Прахалад, Ванж, 2009). Вышеперечисленные характеристики стали причиной, по которой для визуализации социального графа пользователя ВКонтакте была выбрана библиотека, реализующая алгоритм направленных сил «Force Atlas».

# Особенности реализации программы

## Клиент

Клиентская часть веб-приложения выполняет большую часть его функционала, а именно:

* Предоставляет пользователю интерфейс по управлению программой (ввод исходных данных, проверка их корректности, ведение консольного журнала выполнения программы);

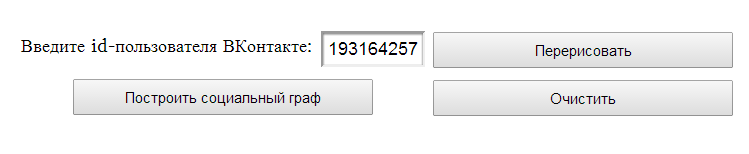


Рисунок 12. Интерфейс управления построением графа

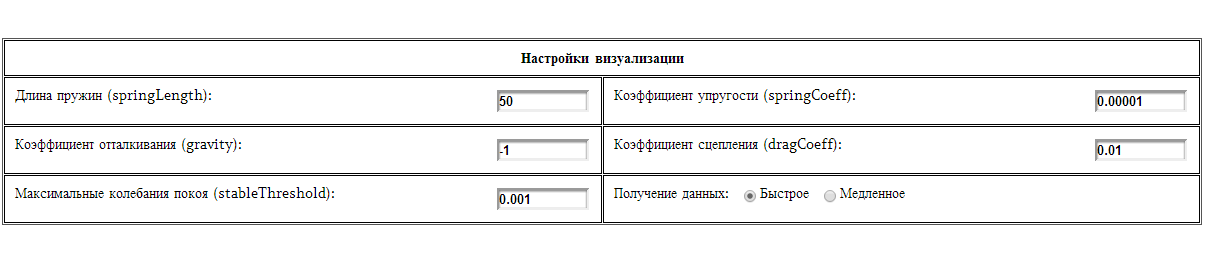


Рисунок 13. Настройки визуализации графа

Таблица 2. Описание настроек визуализации

|  |  |  |
| --- | --- | --- |
| **Настройка** | **Эффект на визуализацию** | **Диапазон допустимых значений** |
| Длина пружин | Чем больше длина, тем на большем расстояния вершины графа будут расположены друг от друга | Действительное число в диапазоне [45; 500] |
| Коэффициент отталкивания | Чем выше, тем дальше друг от друга будут расположены вершины графа | Действительное число в диапазоне [-10; -0.1] |
| Максимальные колебания покоя | Чем выше значение, тем быстрее алгоритм прекратит работу(продолжительность колебаний графа) | Действительное число в диапазоне [0.005; 0.5] |
| Коэффициент упругости | Чем выше, тем меньше расстояние между вершинами и выше частота колебаний | Действительное число в диапазоне [0.000001; 0.00001] |
| Коэффициент сцепления | Чем выше, тем ниже амплитуда и частота колебаний (граф рисуется более плавно) | Действительное число в диапазоне [0.005; 0.09] |
| Режим сбора данных | Скорость получения информации о списках друзей пользователей и добавления новых ребер в граф: 25 списков друзей (список ребер инцидентных 25 вершинам) в секунду -быстрый режим, 1 список друзей – медленный | Быстрый / медленный |

* Формирует и посылает на сервер запрос в зависимости от стадии выполнения (сбор персональных данных главного пользователя, сбор данных его друзей, получение списков друзей);

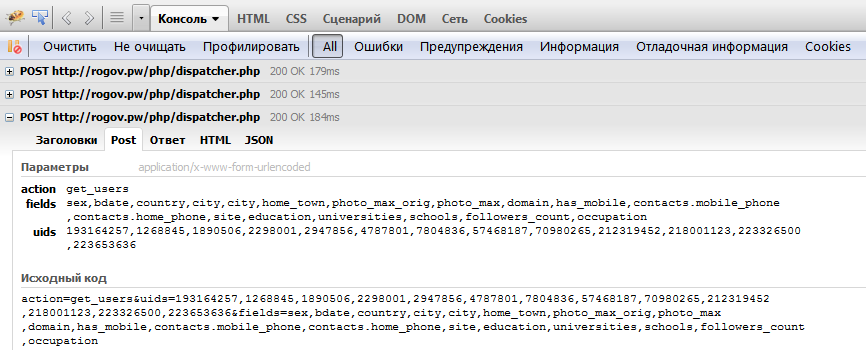


Рисунок 14. Пример сформированного HTTP-POST запроса для получения персональных данных

* Обрабатывает полученные с сервера ответы, динамически обновляя информацию о графе;

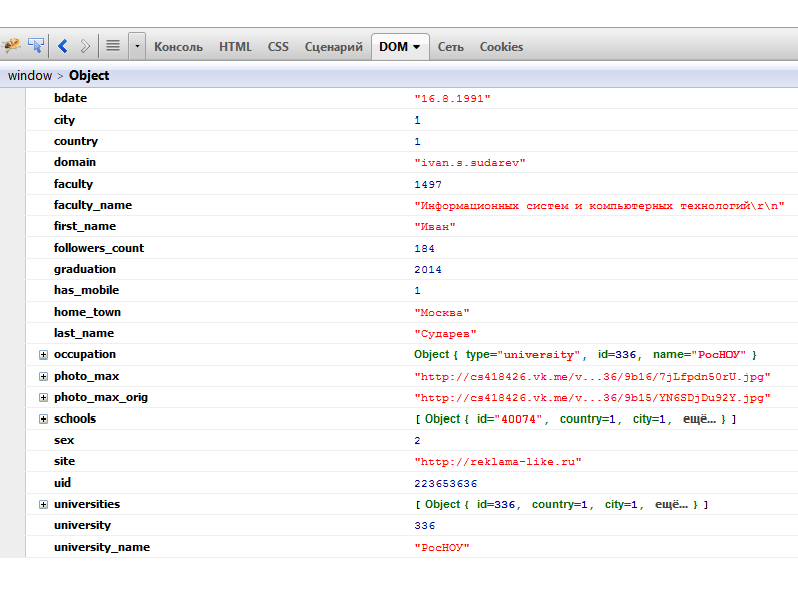


Рисунок 15. Полученные данные пользователя

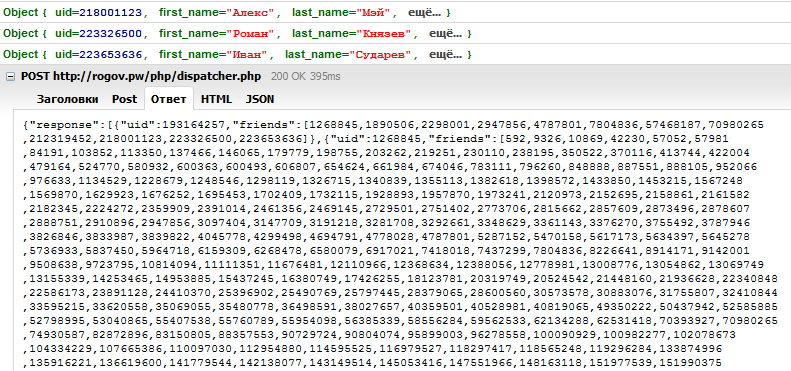


Рисунок 16. Полученные списки друзей пользователей

* Строит двухмерное представление социального графа;

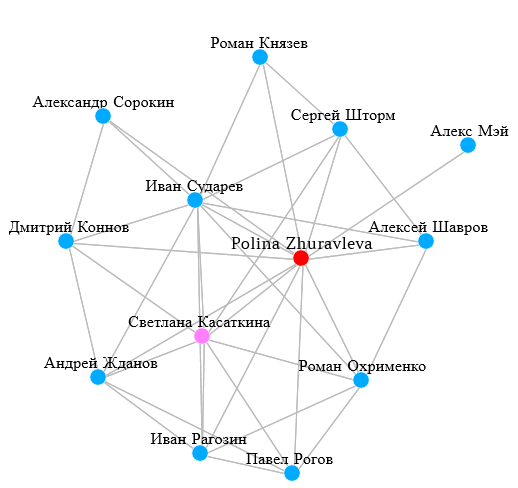


Рисунок 17. Построенное представление социального графа

* Рассчитывает социальные характеристики из параграфа [3.4]

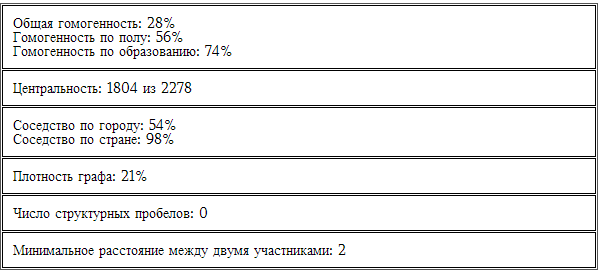


Рисунок 18. Рассчитанные характеристики социального графа

Изначально был выбран подход, реализующий все функции по работе с графом (за исключением его конечной визуализации) на сервере. Однако он не оправдал себя в ходе экспериментов: приложение ожидало ответа значительное время (в среднем не менее 40 секунд). Вынесение задач по формированию графа на клиентскую часть позволило значительно сократить время ожидания до отклика программы. В дополнение к этому процесс работы алгоритма «направленных сил» над графом стал более наглядным.

## Сервер

Серверная часть веб-приложения формирует запросы к методам API ВКонтакте, записывает собранные данные в базу данных и возвращает исходные данные о ребрах и вершинах социального графа на клиентскую часть. Для этого реализован PHP-скрипт «dispatcher.php», выполняющий роль диспетчера запросов. Получив и обработав входной запрос, он определяет, какому скрипту будет передано управление.

## Визуализация

Визуализация графа происходит на его клиентской части в несколько этапов. Сначала рисуется пользователь, его друзья и связывающие их связи.

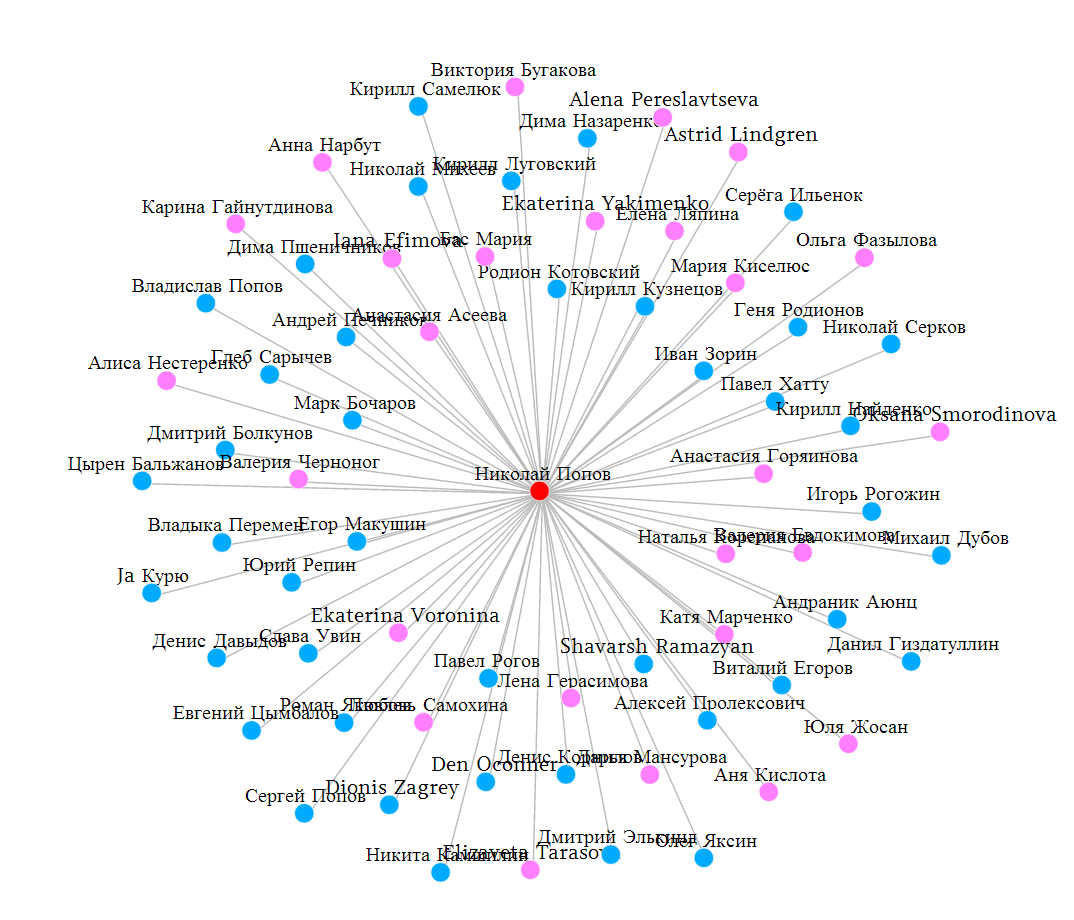


Рисунок 19. Строящийся социальный граф в первый момент времени

Голубым цветом обозначаются участники сети мужского пола, розовым – женского, красным – пользователь, для которого строиться граф. Их связи обозначены серыми отрезками. Затем клиент посылает запрос серверу на получение списков друзей участников сети, сервер в свою очередь обращается к API ВКонтакте. Ответ он возвращается на клиентскую часть, которая, обработав его, ищет тех пользователей в полученных списках, которые представлены в построенном социальном графе. Они добавляются в граф.

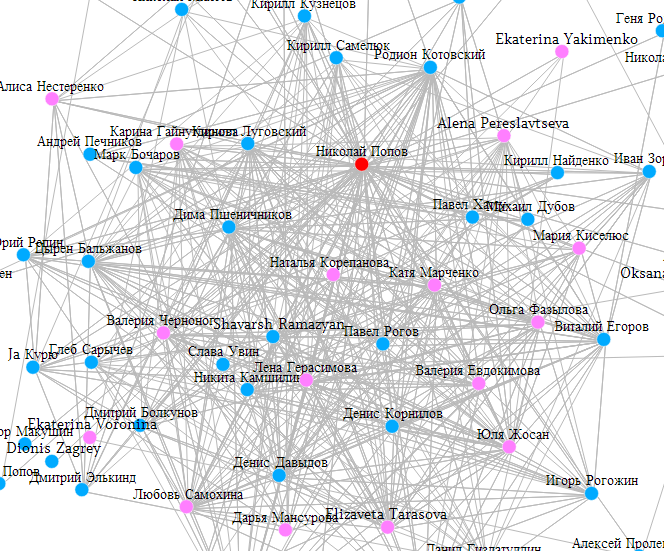


Рисунок 20. Масштабированный фрагмент социального графа

Когда все связи добавлены, граф постепенно стабилизируется, достигая минимального энергетического состояния. Когда колебания достигнут порогового значения, они прекратятся, и алгоритм закончит выполнение. После этого можно подробно изучить сформированные кластеры графа, используя режим полноэкранного просмотра и масштабирование отдельных частей графа.

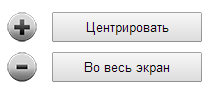


Рисунок 21. Интерфейс для управления изображением графа

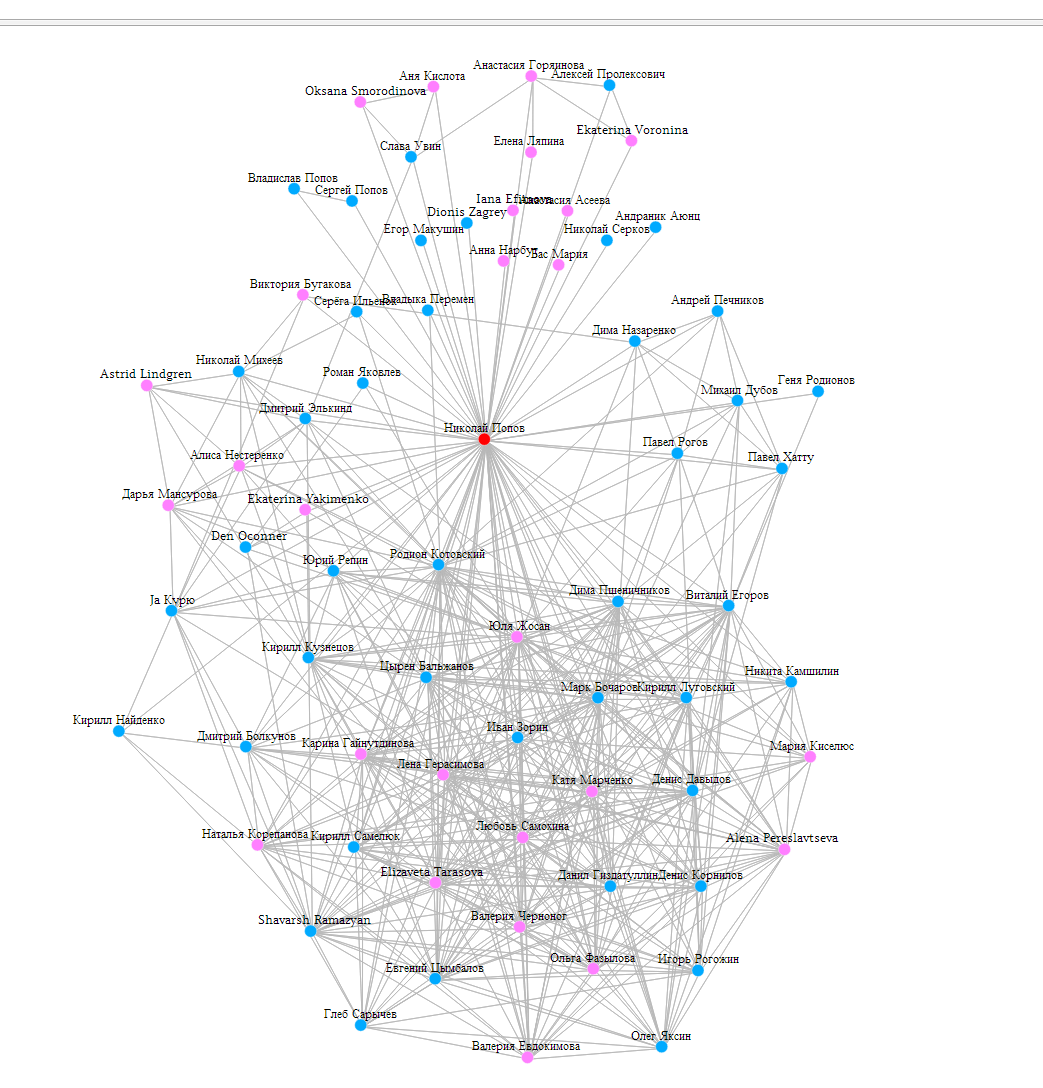


Рисунок 22. Построенный социальный граф пользователя

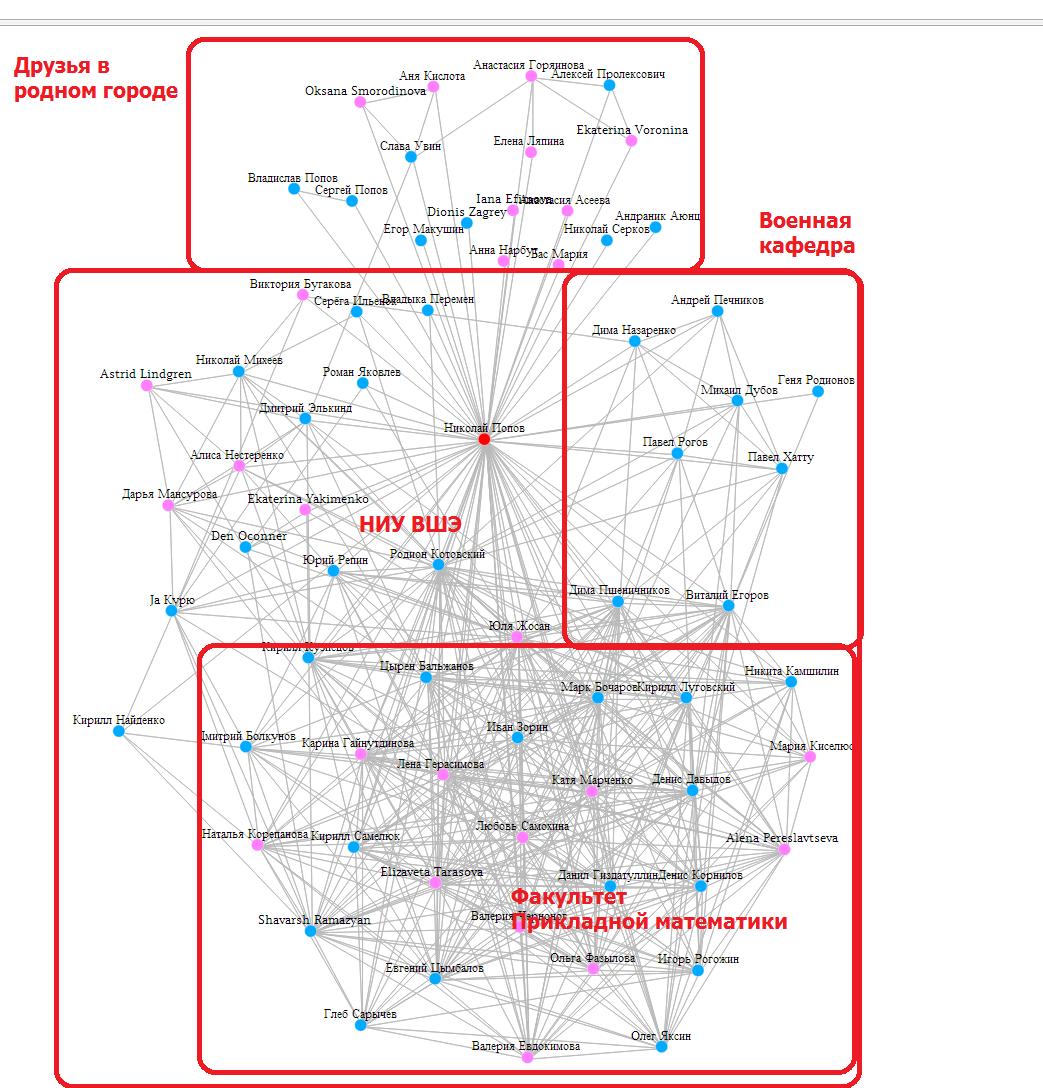
вцфв

Рисунок 23. Социальный граф пользователя с выделенными кластерами

## Расчет характеристик

Веб-приложении обеспечивает вычисление значений следующих социальных характеристик: гомогенность, центральность, плотность графа, расстояние.

Гомогенность по признаку = число участников сети обладающим этим признаком/общее число участников сети. Соседство = процент людей, проживающим в том же городе или стране, что и пользователь, для которого был построен социальный граф. Общая гомогенность = среднее арифметическое всех рассчитанных показателей гомогенности по признаку и соседству.

Центральность пользователя = число минимальных кратчайших путей между любыми двумя его друзьями, проходящими через него.

Плотность графа = отношение реального числа связей в графе к максимально возможному в неориентированном графе с тем же числом вершин (оно вычисляется по формуле N \* (N – 1) / 2, где N – число вершин графа).

Число структурных пробелов = отсутствие связей между частями социального графа. Во всех построенных графах эти пробелы отсутствуют, так как пользователь является связующим звеном («мостом») между всеми его друзьями. Таким образом, не проводя расчетов, можно приравнять этот показатель к нулю.

Расстояние = минимальное расстояние в графе между любыми двумя участниками. Так как во всех построенных графах каждый знает каждого через пользователя, то расстояние всегда будет не больше двух (друг №1 – пользователь – друг №2). Однако нужно проверить граф на полноту, так как в полном графе (все вершины имеют прямые связи друг с другом) расстояние равно единице.

## Технические требования

Для корректной работы веб-приложения может быть использован компьютер, имеющий мышь и характеристики не ниже следующих: однопроцессорный двухъядерный компьютер с 2 ГБ оперативной памяти и 4 ГБ дисковой памяти с выходом в интернет на скорости не менее 2 Mbit/сек.

На клиентских рабочих местах должно быть установлено следующее ПО:

* Операционная система Microsoft Windows Vista/7/8;
* Все подсистемы рассчитаны на использование посредством современных ПК-версий веб-браузеров Google Chrome, Яндекс.Браузер, Mozilla Firefox и Opera, актуальных на 1 апреля 2014 года;
* Для корректной работы необходима поддержка в браузерах SVG-графики, AJAX-запросов и скриптов языка Javascript.

Само веб-приложение должно быть размещено на хостинге, имеющем стандартный набор параметров для обеспечения работы веб-сайтов с PHP-скриптами и СУБД MySQL.

# Заключение

В ходе выполнения данной выпускной квалификационной работы было разработано клиент-серверное веб-приложение, решающие все поставленные задачи:

1. Сбор социально-демографических данных пользователей сети ВКонтакте;
2. Построение на их основе социального графа;
3. Визуализация социального графа на плоскости в виде, пригодном для дальнейшего анализа;
4. Расчет характеристик социального графа.

Программный продукт был снабжен документацией в составе:

* Техническое задание (ГОСТ 19.201-79)
* Программа и методика испытаний (ГОСТ 19.301-79)
* Руководство программиста (ГОСТ 19.505-79)
* Текст программы (ГОСТ 19.401-78)

Дальнейшие направления разработки включают в себя:

* Реализации возможности построения социальных графов пользователей других онлайн-социальных сетей;
* Оптимизация работы программы при пятизначном числе вершин;
* Разработка функционала для выявления и анализа кластеров социального графа;
* Внедрение опции добавления в социальный граф новых пользователей.

# Список использованных источников

[1] Goble, G., The History of Social Networking, DigitalTrends, 2012, URL: <http://www.digitaltrends.com/features/> (retrieved date January 12, 2014)

[2] Sherman M., Implicit vs. explicit social graphs, Matt Sherman's blog of web, 2011, URL: <http://clipperhouse.com/2011/06/19/implicit-vs-explicit-social-graphs/> (retrieved date February 3, 2014)

[3] Miller, R., Facebook Servers, Data Center Knowledge, 2012

[4] Носов В. И., Элементы теории графов, Новосибирск: СГУТИ, 2008

[5] Briggs, G., [Building The Implicit Social Graph](http://moz.com/blog/building-the-implicit-social-graph), *The Moz Blog,* 2011

[6] Abraham A., Hassanien A. E., Snášel V., Computational Social Network Analysis: Trends, Tools and Research Advances, London: Springer, 2009

[7] Russell, M. A., Mining the Social Web, O’Reilly Media, 2011

[8] I. F. Cruz, R. Tamassia, «How to Visualize a Graph: Specification and Algorithms», Tuffs University & Brown University, 1994

[9] Kobourov S. G., Spring Embedders and Force Directed Graph Drawing Algorithms, Cornell University Library of ARXIV, 2012

[10] Fruchterman, T. M. J., & Reingold, E. M., Graph Drawing by Force-Directed Placement. Software: Practice and Experience, 21(11), 1991

[11] Yifan Hu, «Efficient and High Quality Force-Directed Graph Drawing», Wolfram Research Inc, 2005

[12] Jacomy, M., Heymann, M., Venturini, T., Bastian, M., ForceAtlas2, A Graph Layout Algorithm for Handy Network Visualization, Sciencese Po medialab, Universite et Marie Curie, Gephi Consortium, 2007

[13] Bonatoa, A. & Hadib, N. & Hornc, P. & Prałatd, P. & Wange, C., Models of Online Social Networks. Internet Mathematics, 6(3), 285-313, 2009

[14] Krebs, V., Social Network Analysis, A Brief Introduction, Orgnetm, 2012, URL: <http://www.orgnet.com/sna.html> (retrieved date January 31, 2014)

[15] Ugander, J. & Karrer, B. & Backstrom, L. & Marlow, C. *The Anatomy of the Facebook Social Graph*. Cornell University Library of ARXIV, 2011

[16] Mishra, N. & Schreiber, R. & Stanton I. & Tarjan, R. E., Finding Strongly-Knit Clusters in Social Networks. Internet Mathematics, 5(1-2), 155-174, 2008