|  |  |
| --- | --- |
| **Gerb-BMSTU_01** | **Министерство науки и высшего образования Российской Федерации**  Калужский филиал  федерального государственного бюджетного  образовательного учреждения высшего образования  ***«Московский государственный технический университет имени Н.Э. Баумана (национальный исследовательский университет)»***  ***(КФ МГТУ им. Н.Э. Баумана)*** |

**ФАКУЛЬТЕТ \_ *ИУ-КФ «Информатика и управление»\_\_\_\_\_\_\_\_\_***

**КАФЕДРА** \_\_ ***ИУ5-КФ «Системы обработки информации»***

**НАУЧНО-ИССЛЕДОВАТЕЛЬСКАЯ РАБОТА**

**«Исследование JavaScript библиотек для создания интерактивных карт»**

Студент гр. САПР.Б-71 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (Рахманкулов А.Р.)

(подпись) (Ф.И.О.)

Руководитель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (Вершинин Е.В.)

(подпись) (Ф.И.О.)

Оценка руководителя \_\_\_\_\_ баллов \_\_\_\_\_\_\_\_\_\_\_

30-50 (дата)

Оценка защиты \_\_\_\_\_ баллов \_\_\_\_\_\_\_\_\_\_\_

30-50 (дата)

Оценка проекта \_\_\_\_\_ баллов \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(оценка по пятибалльной шкале)

Комиссия: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_)

(подпись) (Ф.И.О.)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_)

(подпись) (Ф.И.О.)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_)

(подпись) (Ф.И.О.)

Калуга, 2020 г.

Содержание

[Введение 3](#_Toc62649979)

[1. JavaScript-библиотеки для интерактивных карт-визуализаций 4](#_Toc62649980)

[1.1 amMap 4](#_Toc62649981)

[1.2 AnyMap (от AnyChart) 5](#_Toc62649982)

[1.3 D3.js (от Mike Bostock) 6](#_Toc62649983)

[1.4 Google GeoCharts (от Google) 7](#_Toc62649984)

[1.5 Highmaps (от Highsoft) 8](#_Toc62649985)

[1.6 jQuery Mapael (от Vincent Broute) 9](#_Toc62649986)

[1.7 jVectorMap (от Kirill Lebedev) 10](#_Toc62649987)

[1.8 Kartograph (от Gregor Aisch) 11](#_Toc62649988)

[1.9 React-yandex-maps (от Yandex) 12](#_Toc62649989)

[2. Пример разработки клиентского приложения 13](#_Toc62649990)

[Заключение 16](#_Toc62649991)

[Список литературы 16](#_Toc62649992)

# Введение

Цель научно-исследовательской работы – анализ библиотек, написанных на языке программирования JavaScript для отображения данных и создания интерактивных карт.

Для достижения поставленной цели необходимо решить следующие задачи:

* описать актуальности работы;
* проанализировать существующие библиотеки, выделить их достоинства и недостатки;
* продемонстрировать пример разработки клиентского приложения с помощью одной из рассмотренных библиотек.

Визуализация данных стала неотъемлемой частью жизни практически каждого веб-разработчика.

JavaScript-библиотеки для создания карт можно разделить на два типа. Одни просто позволяют отображать географическое положение одного или нескольких объектов. Для подобной задачи можно использовать карты типа Google Maps или OpenStreetMaps в качестве источника. В принципе, таких решений достаточно, и результат их работы примерно таков:

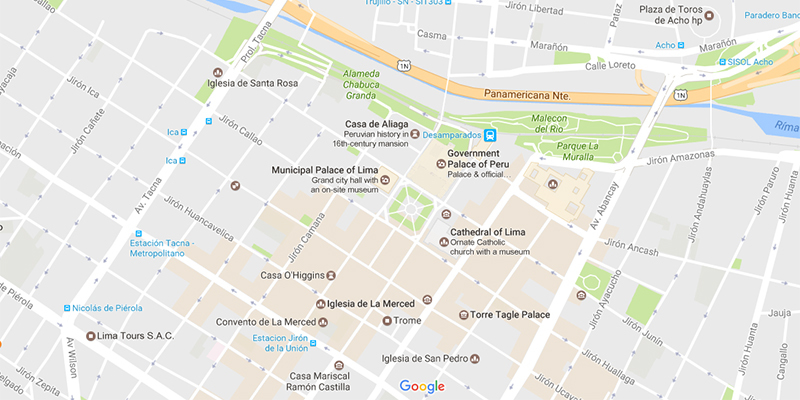


Рисунок 1 – google maps

Другой тип – JavaScript-библиотеки, с помощью которых можно делать именно визуализацию данных посредством создания красивых интерактивных карт. Они показывают либо связи между величинами в разных географических регионах, либо перемещение каких-либо объектов относительно их местоположения, и т.п. – всё то, что крайне важно в плане анализа данных и business intelligence.

## JavaScript-библиотеки для интерактивных карт-визуализаций

### amMap

[amMap](http://www.ammap.com/) – специальная JavaScript (HTML5) библиотека для построения карт, разработанная командой [amCharts](http://amcharts.com/). Она не нуждается в каких-либо внешних зависимостях и позволяет довольно просто делать красивые choropleth, bubble, dot (point), connector и flow карты, поддерживая много полезных интерактивных функций.



Рисунок 2 –пример anMap

В частности, amMap дает возможность легко «погружаться» в тот или иной выбранный участок карты с помощью drill down или, например, использовать любые картинки в качестве маркеров – довольно любопытная вещь.

Для начала работы с amMap нужно скачать ZIP-файл с бинарниками. Кстати, в нем уже лежит по 455 карт в каждом из поддерживаемых форматов (JavaScript/SVG). [Документация](https://www.amcharts.com/kbc/javascript-maps/) у amMap не очень большая и сводится по большей части к набору вопросов и ответов.

**Описание API:** есть.

**Зависимости:** нет.

**Лицензия:** бесплатно (брендированные карты) или платно (от $140).

### AnyMap (от AnyChart)

AnyMap – одна из популярных JavaScript (HTML5) библиотек для визуализации данных, созданных компанией AnyChart.

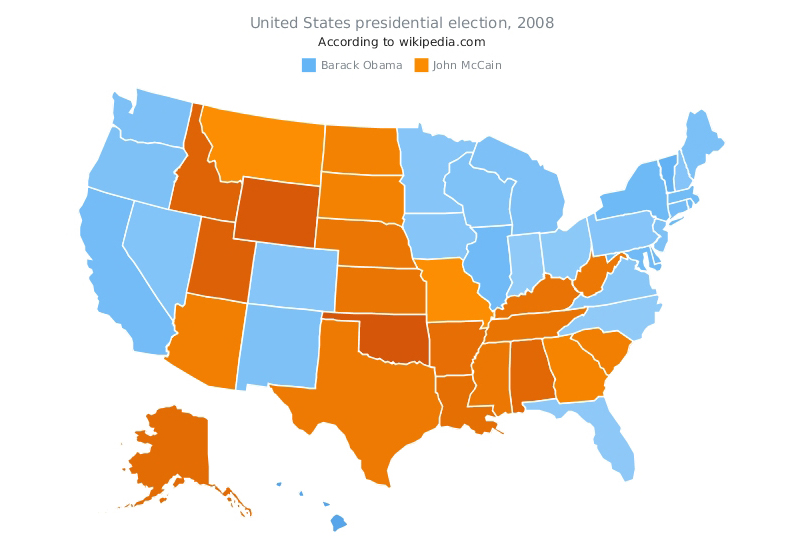


Рисунок 3 – пример визуализации данных от anyMap

. Как и в случае с amMap, для создания карты с ее помощью не требуется больших усилий, равно как и подключения каких-либо внешних ресурсов или сторонних библиотек (jQuery и т.д.). AnyMap также поддерживает все базовые типы серий, который только могут понадобиться: choropleth, bubble, dot, connector, flow.

В число основных интерактивных преимуществ AnyMap входят drill down для более плотного изучения отдельных районов, событийная модель для обработки событий (пользовательской интерактивности), цветовая шкала для автоматического раскрашивания карты по заданным настройкам.

AnyMap как бы сочетает в себе возможности «коробочных» решений и в то же время является хорошо расширяемой и контролируемой со стороны разработчика, позволяет полностью кастомизировать визаулизации путем добавления или изменения различных элементов и легко создать собственные карты.

Бинарники AnyMap находятся в ZIP-файле и содержат множество примеров, а на CDN у AnyChart доступны сотни карт в самых разных форматах: GeoJSON, TopoJSON, SHP, SVG. Также стоит заметить, что у этой библиотеки богатая документация и обширное описание API, кстати, довольно легкое для понимания и – опять же – с большим количеством примеров.

**Описание API:** есть.

**Зависимости:** нет.

**Лицензия:** бесплатно (брендированные карты) или платно (от $79).

### D3.js (от Mike Bostock)

D3.js – мощная в плане возможных результатов библиотека для визуализации данных с открытым исходным кодом. В отличие от других упомянутых JavaScript-библиотек, D3 – это, скорее, фреймворк, который дает максимальную свободу творчества. Хотя создание интерактивных карт здесь не так очевидно, как, скажем, в решениях от amCharts, AnyChart или Highcharts.



Рисунок 4 – пример отображения данных с помощью D3.js

В частности, чтобы сделать такую замечательную интерактивную карту, как на рисунке 4, потребуется изучить достаточно большое количество материала. Хотя, надо сказать, это стоит того, потому что в итоге можно сделать реально потрясающие, оригинальные визуализации.

Фактически D3 имеет много достоинств, среди которых полная кастомизация поведения, событийная модель и т.д. В целом это идеальное решение для веб-приложений.

Хотя, к сожалению, D3 не предоставляет собственных карт. Это значит – их придется самостоятельно искать в открытых источниках, но вряд ли сейчас это большая проблема.

Документации у D3 нет, однако есть хорошо расписанное API и множество различных туториалов, примеров и прочих полезных материалов буквально по всему Интернету.

**Описание API:** есть.

**Зависимости:** нет.

**Лицензия:** бесплатно.

### Google GeoCharts (от Google)

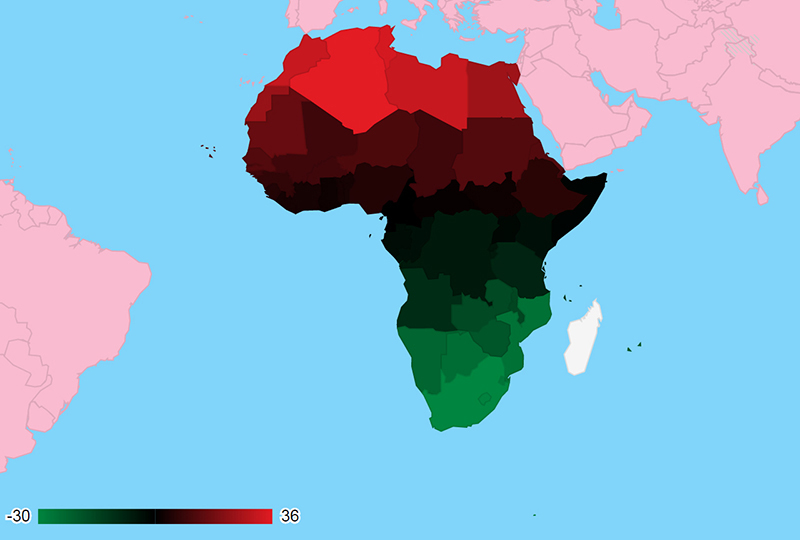


Рисунок 5 – пример визуализации данных с помощью библиотеки Google Charts

У гиганта Google есть своя JavaScript-библиотека для визуализации данных – [Google Charts](https://developers.google.com/chart/interactive/docs/gallery/geochart). Она содержит возможности и для построения интерактивных карт-визуализаций, называющихся здесь geocharts, видимо, чтобы их не путали с обычными картами, которые нужны только для отображения местоположения объектов.

Карта в Google Geocharts только одна, и это – карта мира. Однако можно выбрать регион, на который при рисовании будет сделан фокус с помощью зума.

К сожалению, API не поддерживает прокрутку (scrolling), изменение масштаба (zoom), перетаскивание (drag). Также Google не предлагает иных типов серий для карт, кроме choropleth и bubble.

Так или иначе, GeoCharts – отличное решение для тех людей и проектов, которым не нужен большой функционал, но кто ищет прежде всего скорость и стабильность вместо какой-то особенной красоты и кастомизации.

**Описание API:** есть.

**Зависимости:** нет.

**Лицензия:** бесплатно.

### Highmaps (от Highsoft)

Highmaps является разработкой компании Highsoft. Ее ключевые преимущества – открытый код со всеми его плюсами, небольшой размер, высокая скорость отрисовки и широкий набор интерактивных возможностей, таких как drill down и т.п.

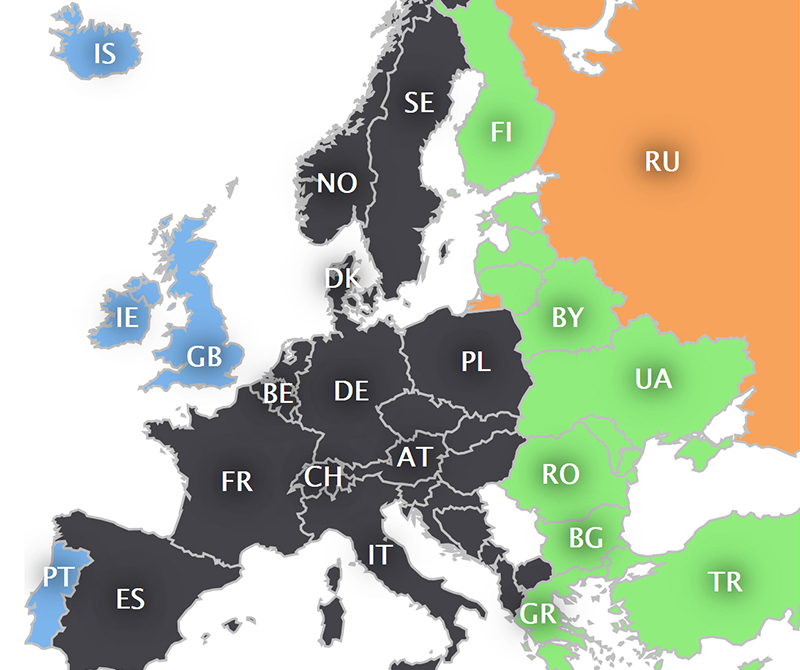


Рисунок 6 – пример отображения данных с помощью библиотеки Highmaps

Среди поддерживаемых в Highmaps серий – choropleth и bubble. Кроме того, с помощью этой библиотеки можно рисовать линии для обозначения дорог, рек и коннекторов. Хотя сделать, скажем, flow-карту здесь не так просто, как в тех же amMap и AnyMap.

Также стоит отметить, что библиотека Highmaps не работает без jQuery[[1]](#footnote-1), так что для их использования эта зависимость должна быть подключена.

Положительный момент – Highmaps имеет большое количество карт в форматах SVG[[2]](#footnote-2) и GeoJSON[[3]](#footnote-3). Также у нее обильная документация, которая позволяет довольно быстро и успешно разобраться во всевозможных настройках.

**Описание API:** есть.

**Зависимости:** jQuery.

**Лицензия:** бесплатно (брендированные карты) и платно (от $390).

### jQuery Mapael (от Vincent Broute)

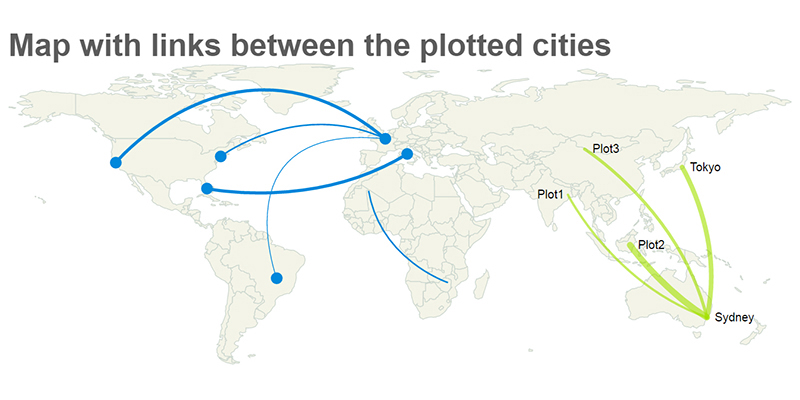


Рисунок 7 – пример визуализации данных с помощью библиотеки jQuery Mapael

jQuery Mapael – еще одна очень достойная, хорошо сделанная библиотека с открытым исходным кодом, предназначенная для создания динамических векторных карт. Она основана на jQuery и Raphael JS, так что эти зависимости нужно иметь в виду при построении поддерживаемых ею choropleth, bubble или connector карт.

Набор своих, готовых для использования карт лежит в репозитории Mapael. Документация и описание API соединены в одну статью.

**Описание API:** есть.

**Зависимости:** jQuery и Raphael.

**Лицензия:** бесплатно.

### jVectorMap (от Kirill Lebedev)

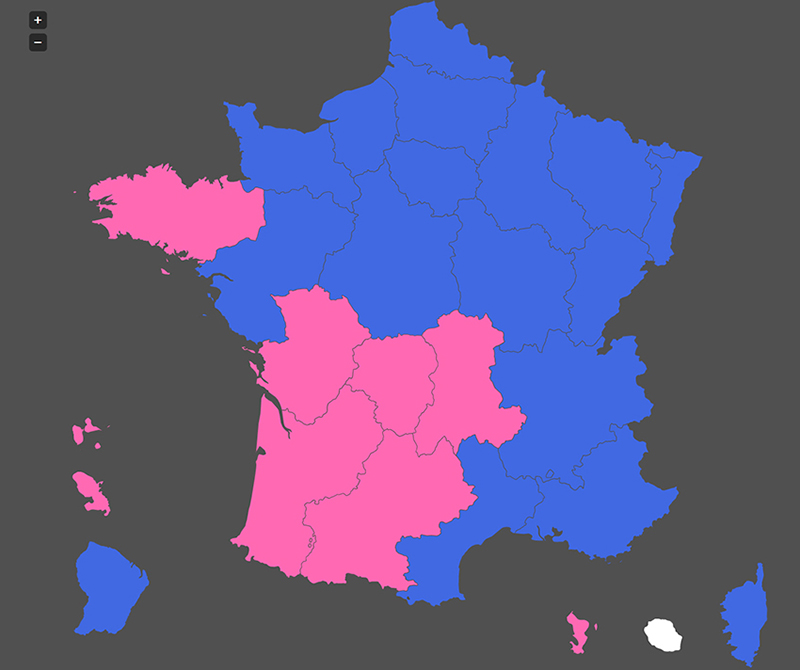


Рисунок 8 – пример отображения визуальных данных при помощи библиотеки jVectorMap

Библиотека для интерактивных карт jVectorMap имеет открытый исходный код.Она не будет работать без jQuery. Хотя кому-то может показаться, что такая внешняя зависимость делает данную (или любую другую) библиотеку не слишком легко-универсальной, это не должно стать большой проблемой в наше время, ведь jQuery с большим отрывом занимает первое место среди JavaScript-библиотек для сайтов и по популярности, и по доле рынка.

К сожалению, выбор доступных типов серий здесь не слишком велик: в jVectorMap предлагаются только choropleth и marker карты, но в то же они сделаны очень хорошо. Кроме того, тут можно использовать в качестве маркеров любые картинки, а также имеются drill down и ряд других интерактивных преимуществ.

У jVectorMap довольно небольшое описание API и нет документации в привычном по крайней мере для меня виде. Однако имеется маленький, но все равно очень полезный туториал Getting Started. Свои карты используются, но только в формате GeoJSON.

**Описание API:** есть.

**Зависимости:** jQuery.

**Лицензия:** бесплатно и платно (от $39).

### Kartograph (от Gregor Aisch)

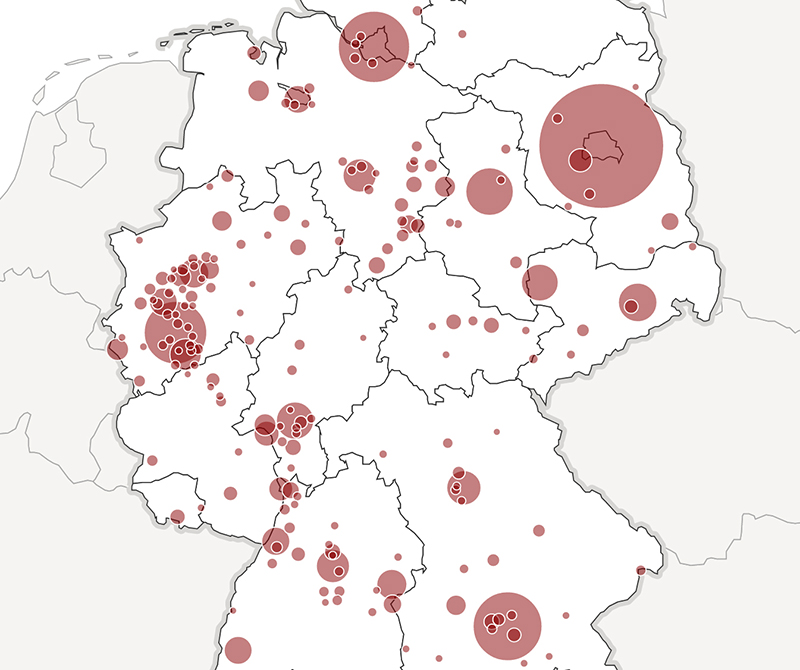


Рисунок 9 – пример отображения визуальных данных при помощи библиотеки Kartograph

Kartograph – еще одно открытое решение с массой преимуществ. Данную библиотеку характеризует обилие визуальных эффектов, таких как тени, размытие и т.д.

Интересно также, что вдобавок к традиционным choropleth и bubble типам серий здесь еще есть 3D columns (или, как их иногда называют, вертикальные 3D bars).

Готовой коллекции карт в Kartograph нет. Но библиотека работает с любыми картами в формате SVG. Также она предлагает утилиту Kartograph.py для создания карт.

Хотя Kartograph имеет зависимости и не будет работать без Raphael JS (используется как движок для рисования) и jQuery, эта библиотека делает процесс создания карт действительно довольно понятным и беспроблемным, тогда как документация и описание API данной библиотеки, соединенные в одну статью, позволяют вполне легко и удобно заполучить нужные интерактивные карты для ваших проектов.

**Описание API:** есть.

**Зависимости:** jQuery и Raphael.

**Лицензия:** бесплатно.

### React-yandex-maps (от Yandex)

react-yandex-maps - это оболочка вокруг API Яндекс.Карт, которая позволяет отображать Яндекс Карты в вашем приложении React с минимальными трудностями.

Эта библиотека представляет собой тонкую оболочку API Яндекс.Карт, многие объекты и функции API Яндекс.Карт поддерживаются компонентами React, но не все из них и не все их функции.

Поддерживаемые функции:

* изменение запроса API Яндекс.Карт через компонент провайдера YMaps;
* получение доступа к API Яндекс.Карт в компонентах через withYMaps HOC;
* гранулярная загрузка необходимых модулей API с помощью API модулей Яндекс.Карт;
* контролируемые и неконтролируемые компоненты для каждого поддерживаемого объекта.

Поддерживаемые объекты Yandex.Maps : Map, Panorama.

Поддерживаемые Гео Объекты : GeoObject, Placemark, Polyline, Rectangle, Polygon, Circle, Clusterer, ObjectManager.

Поддерживаемые контролы: Button, FullscreenControl, GeolocationControl, ListBox, ListBoxItem, RouteButton, RouteEditor, RoutePanel, RulerControl, SearchControl, TrafficControl, TypeSelector, ZoomControl

К сожалению, рендеринг узлов React DOM в фабрики шаблонов Yandex.Maps не поддерживается (т. е. нельзя отображать содержимое baloon с помощью React с этой библиотекой из коробки).

**Описание API:** есть.

**Зависимости:** API Яндекс карт[[4]](#footnote-4).

**Лицензия:** бесплатно и платно (от 1600$ в год в зависимости от количества запросов).

## Пример разработки клиентского приложения

Для начала необходимо создать react приложение. Проще всего это сделать с помощью ввода команды npx create-react-app [название\_проекта] в интерпретаторе командной строки cmd.exe в любом удобном месте на компьютере:

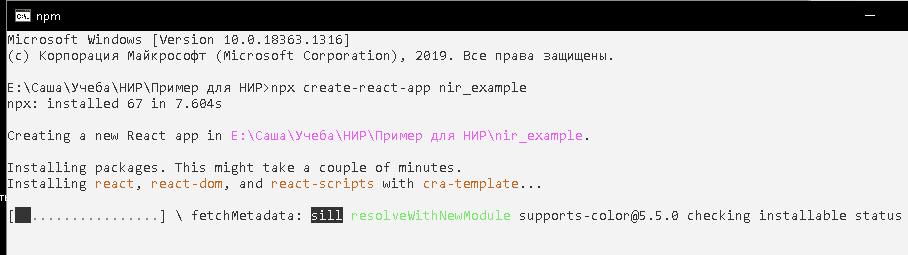


Рисунок 10 – создание проекта React

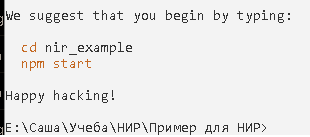


Рисунок 11 – успешное создание проекта

Далее необходимо добавить в созданный по шаблону проект саму библиотеку react-yandex-maps с помощью команды в терминале: npm install react-yandex-maps.

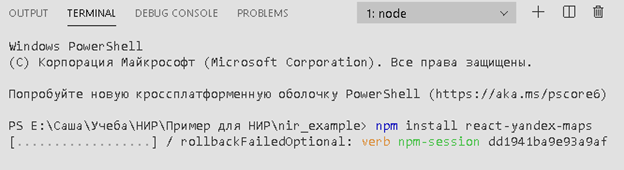


Рисунок 12 – добавление библиотеки react-yandex-maps в проект

Добавление карт в приложение можно разделить на два этапа.

Во-первых, нужно добавить компонент-провайдер, который будет обрабатывать загрузку API Яндекс.Карт и предоставлять его любому интересующему компоненту на странице.

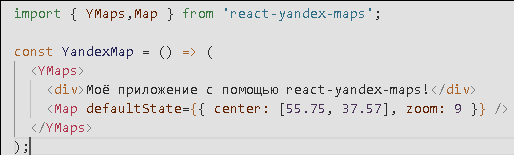


Рисунок 13 – создание компонента

Затем необходимо добавить компонент Map с желаемым состоянием:



Рисунок 14 – добавление компонента

Затем с помощью команды npm start в терминале нужно запустить клиентское приложение.

В результате выполнения приложения в браузере при отправке запроса на получение ресурса с localhost:3000 можно будет увидеть следующую картину:

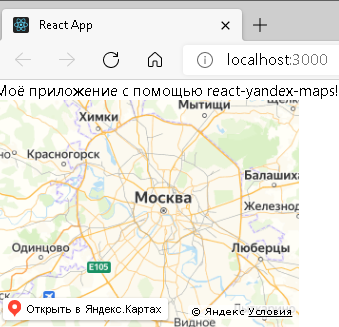


Рисунок 15 – результат работы программы

Теперь, как только API Яндекс.Карт загружается компонентом провайдера YMaps, компонент Map будет отображать объект ymaps.Map на странице.

# Заключение

Все перечисленные JavaScript-библиотеки для визуализации данных с помощью интерактивных карт – кросс-браузерные. Некоторые из них даже поддерживают старые браузеры, такие как Internet Explorer 6.

Все бесплатные библиотеки довольно-таки неплохо справляются с созданием интерактивных карт. Но они не обеспечивают такую широкую поддержку, как коммерческие.

# Список литературы

1. 8 JavaScript Librarites for Interactive Map Visualizations [Электронный ресурс] Режим доступа : <https://onextrapixel.com/author/ruslan-borovikov/> (Дата обращения 22.01.2021).
2. Официальный сайт разработчиков библиотеки react-yandex-maps [Электронный ресурс] Режим доступа: https://react-yandex-maps.now.sh/(Дата обращения 22.01.2021).
3. Официальный сайт разработчиков библиотеки Kartograph [Электронный ресурс] Режим доступа: <https://kartograph.org/> (Дата обращения 22.01.2021).

1. jQuery — [набор функций JavaScript](https://ru.wikipedia.org/wiki/%D0%91%D0%B8%D0%B1%D0%BB%D0%B8%D0%BE%D1%82%D0%B5%D0%BA%D0%B0_JavaScript), фокусирующийся на взаимодействии [JavaScript](https://ru.wikipedia.org/wiki/JavaScript) и [HTML](https://ru.wikipedia.org/wiki/HTML). [↑](#footnote-ref-1)
2. SVG (от [англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) Scalable Vector Graphics — [масштабируемая](https://ru.wikipedia.org/wiki/%D0%9C%D0%B0%D1%81%D1%88%D1%82%D0%B0%D0%B1%D0%B8%D1%80%D1%83%D0%B5%D0%BC%D0%BE%D1%81%D1%82%D1%8C) [векторная графика](https://ru.wikipedia.org/wiki/%D0%92%D0%B5%D0%BA%D1%82%D0%BE%D1%80%D0%BD%D0%B0%D1%8F_%D0%B3%D1%80%D0%B0%D1%84%D0%B8%D0%BA%D0%B0)) — [язык разметки](https://ru.wikipedia.org/wiki/%D0%AF%D0%B7%D1%8B%D0%BA_%D1%80%D0%B0%D0%B7%D0%BC%D0%B5%D1%82%D0%BA%D0%B8) масштабируемой векторной графики, созданный [Консорциумом Всемирной паутины (W3C)](https://ru.wikipedia.org/wiki/W3C) и входящий в подмножество расширяемого языка разметки [XML](https://ru.wikipedia.org/wiki/XML), предназначен для описания двумерной векторной и смешанной векторно/[растровой](https://ru.wikipedia.org/wiki/%D0%A0%D0%B0%D1%81%D1%82%D1%80%D0%BE%D0%B2%D0%B0%D1%8F_%D0%B3%D1%80%D0%B0%D1%84%D0%B8%D0%BA%D0%B0) графики в формате XML. [↑](#footnote-ref-2)
3. GeoJSON — открытый формат, предназначенный для хранения географических структур данных, основан на [JSON](https://ru.wikipedia.org/wiki/JSON)(JavaScript Object Notation). [↑](#footnote-ref-3)
4. API Яндекс.Карт — это набор сервисов, которые позволяют использовать картографические данные и технологии Яндекса в собственных проектах. [↑](#footnote-ref-4)