УДК 004.942

Анализ методов отображения индексов пожарной опасности на картах Google Maps

Ганьков Сергей Юрьевич
Приамурский государственный университет имени Шолом-Алейхема
магистрант

Глаголев Владимир Александрович
Приамурский государственный университет имени Шолом-Алейхема
к.г.н., доцент кафедры информационных систем, математики и правовой информатики

Аннотация
В работе описаны способы визуализации индексов пожарной опасности на картах Google Maps.
Ключевые слова: Google Maps, пожарная опасность, JavaScript, Google API Maps, карты.

Analysis of methods for displaying fire hazard indices on Google Maps

Gankov Sergei Yurievich
Sholom-Aleichem Priamursky State University
undergraduate

Bazhenov Ruslan Ivanovich
Sholom-Aleichem Priamursky State University
candidate of geographical sciences, associate professor, Department of Information Systems, Mathematics and Legal Informatics

Abstract
The article describes ways to visualize fire hazard indices on Google Maps.
Keywords: Google Maps, fire hazard, JavaScript, Google API Maps, maps.

Google JavaScript API Maps позволяет настраивать карты с собственным контентом и изображениями для отображения на веб-страницах и мобильных устройствах. Google PI Maps содержит четыре основных типа карт (дорожная карта, спутниковая, гибридная и рельефная), которые можно изменять с помощью слоев и стилей, элементов управления и событий, а также различных служб и библиотек.

Для использования возможностей API необходимо объявить директиву DOCTYPE в нашем веб-приложении. Объявим наше приложение как HTML5, используя код ниже.

```
<!DOCTYPE html>
```

Рис. 1 Код приложения

Для корректного отображения карты все размеры блоков CSS процентах должны наследоваться от элементов родительского блока, и если какой-либо из этих предков не содержит размер, предполагается, что они имеют размер 0 x 0 пикселей. По этой причине мы используем следующую <style> декларацию.

```
<style>
  #map {
    height: 100%;
  }
  html, body {
    height: 100%;
    margin: 0;
    padding: 0;
  }
</style>
```

Рис. 2 Декларация <style>

Эта декларация CSS указывает, что контейнер карты (с идентификатором map) должен занимать 100% высоты тела HTML документа. Мы обязательно должны объявить эти размеры для элементов <body> и <html>.

Подключение интерфейса Google API Maps к нашему веб приложению осуществляется с помощью скрипт – тега, который можно динамически добавить в наш HTML-файл или использовать его в отдельном скрипт-файле. Чтобы динамически загрузить модуль JavaScript Google API Maps, необходимо выполнить следующий код (Рис.19).

```
// Create the script tag, set the appropriate attributes
var script = document.createElement('script');
script.src = 'https://maps.googleapis.com/maps/api/js?key=YOUR_API_KEY&callback=initMap';
script.defer = true;
script.async = true;

// Attach your callback function to the `window` object
window.initMap = function() {
  // JS API is loaded and available
};

// Append the 'script' element to 'head'
document.head.appendChild(script);
```

Рис. 3 Код загрузки JavaScript Google API Maps

Целиком код создания объекта карты выглядит следующим образом (Рис.20).

```html
<!DOCTYPE html>
<html>
  <head>
    <title>Simple Map</title>
    <meta name="viewport" content="initial-scale=1.0">
    <meta charset="utf-8">
    <style>
      /* Always set the map height explicitly to define the size of the div
       * element that contains the map. */
      #map {
        height: 100%;
      }
      /* Optional: Makes the sample page fill the window. */
      html, body {
        height: 100%;
        margin: 0;
        padding: 0;
      }
    </style>
  </head>
  <body>
    <div id="map"></div>
    <script>
      var map;
      function initMap() {
        map = new google.maps.Map(document.getElementById('map'), {
          center: {lat: -34.397, lng: 150.644},
          zoom: 8
        });
      }
    </script>
    <script src="https://maps.googleapis.com/maps/api/js?key=YOUR_API_KEY&callback=initMap"
    async defer></script>
  </body>
</html>
```

Рис. 4. Создание карты с центром в Сиднее, Новый Южный Уэльс, Австралия

Рис. 5. Результат выполнения кода

Для отрисовки полигональной сетки с возможностью раскраски на картах Google Maps существуют следующие методы API: Polygon и UserOverlay.

Метод UserOverlay сводится к отрисовки специфическим образом html – таблицы поверх слоя карты. Ниже представлен пример реализации данного метода на Java script.

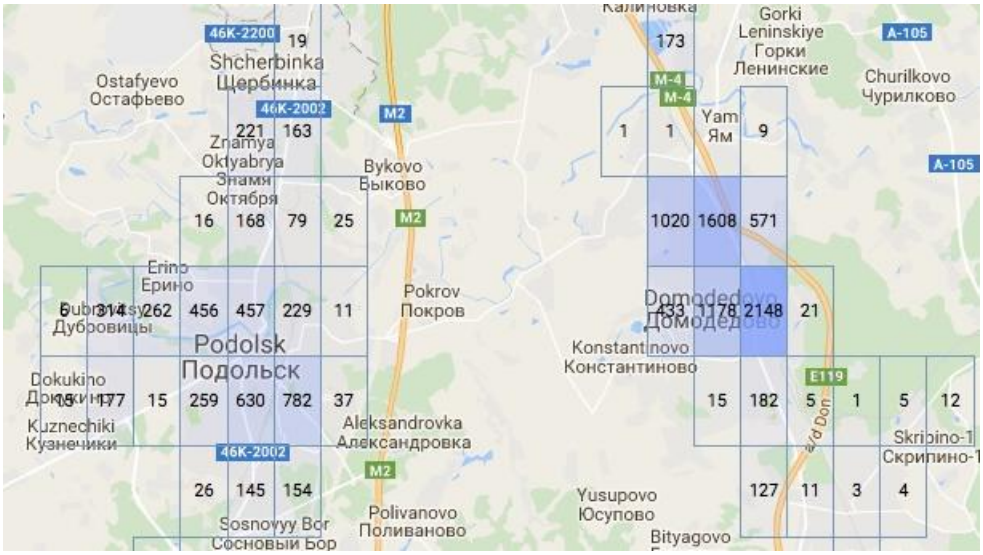


Рис. 6. Результат работы Метод UserOverlay

Данный метод имеет ряд недостатков: отсутствие точной геопривязки каждой ячейки сетки, большая ресурсоемкость в сравнении с API: Polygon, слабые возможности интерактивности ячеек, а также более сложная реализация с в коде. В своей работе мы использовали метод API: Polygon

Метод API: Polygon сводится к отрисовки каждой ячейки полигональной сетки с помощью отдельного объекта типа Polygons. Это основной метод отображения объектов произвольной формы на картах Google.

В дополнение к универсальному классу Polygon, Google Maps JavaScript включает в себя специальный класс для объектов прямоугольной формы, именуемых Rectangle.

Мы можем определить пользовательские цвета, веса и непрозрачности для края прямоугольника (обводка), а также пользовательские цвета и непрозрачности для области внутри прямоугольника (заливка). Цвета должны быть указаны в шестнадцатеричном числовом стиле HTML.

У прямоугольника есть свойство bounds, которое определяет его форму путем указания координат нижнего левого и правого верхнего угла прямоугольника.

Свойство прямоугольника editable указывает, могут ли пользователи редактировать форму. Возможно также установить свойство draggable позволяющее пользователям перетаскивать прямоугольник.

Этот пример добавляет красный прямоугольник на карту (Рис.7).

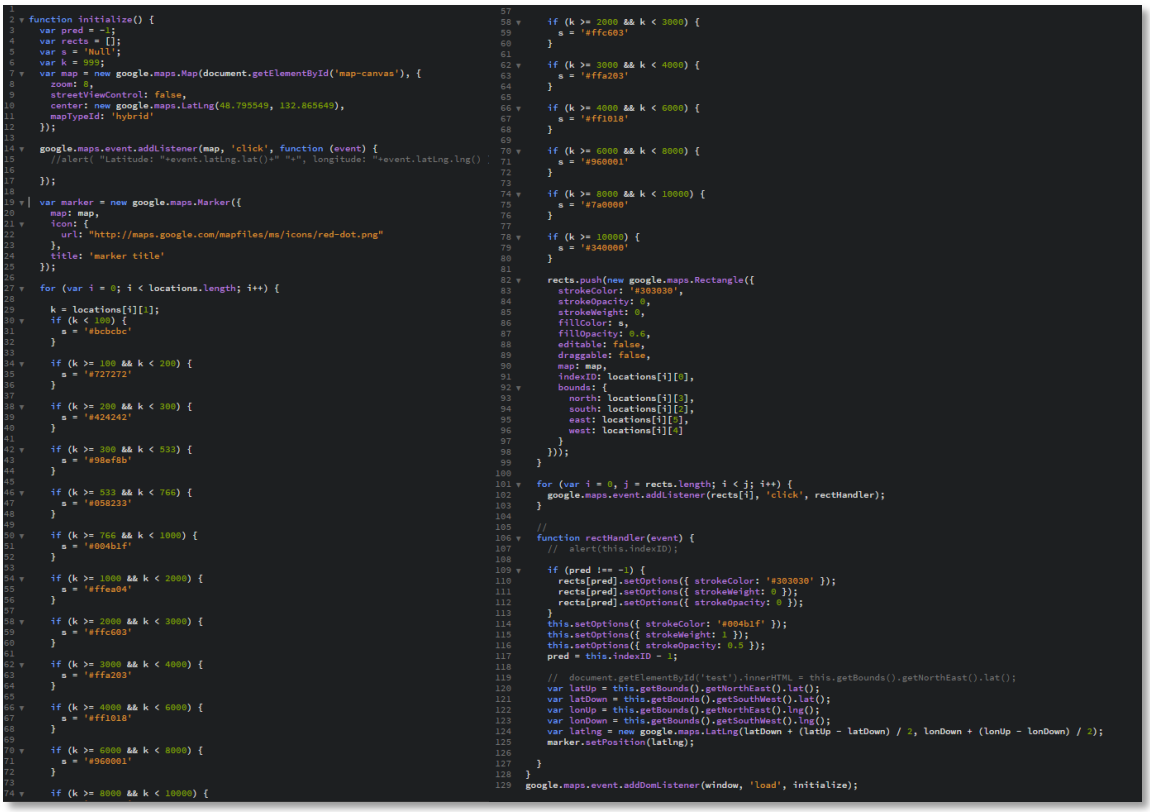
```
function initMap() {
  var map = new google.maps.Map(document.getElementById('map'), {
    zoom: 11,
    center: {lat: 33.678, lng: -116.243},
    mapTypeId: 'terrain'
  });

  var rectangle = new google.maps.Rectangle({
    strokeColor: '#FF0000',
    strokeOpacity: 0.8,
    strokeWeight: 2,
    fillColor: '#FF0000',
    fillOpacity: 0.35,
    map: map,
    bounds: {
      north: 33.685,
      south: 33.671,
      east: -116.234,
      west: -116.251
    }
  });
}
```

Рис. 7. Создание карты с центром в Сиднее, Новый Южный Уэльс, Австралия

Рис. 8. Результат выполнения кода

Основной компонент пользовательского интерфейса – это карта Google Maps. Мы реализовали его как фрейм на JavaSript, его листинг представлен ниже (Рис.9).



```
function initialize() {
    var pred = -1;
    var rects = [];
    var s = 'Null';
    var k = 999;
    var map = new google.maps.Map(document.getElementById('map-canvas'), {
      zoom: 8,
      streetViewControl: false,
      center: new google.maps.LatLng(48.795549, 132.865648),
      mapTypeId: 'hybrid'
    });

    google.maps.event.addListener(map, 'click', function (event) {
      //alert( "Latitude: "+event.latLng.lat()+" "+", longitude: "+event.latLng.lng() );

    });

    var marker = new google.maps.Marker({
      map: map,
      icon: {
        url: "http://maps.google.com/mapfiles/ms/icons/red-dot.png"
      },
      title: 'marker title'
    });

    for (var i = 0; i < locations.length; i++) {

      k = locations[i][1];
      if (k < 100) {
        s = '#bcbcbc'
      }

      if (k >= 100 && k < 200) {
        s = '#727272'
      }

      if (k >= 200 && k < 300) {
        s = '#424242'
      }

      if (k >= 300 && k < 533) {
        s = '#98ef6b'
      }

      if (k >= 533 && k < 766) {
        s = '#058233'
      }

      if (k >= 766 && k < 1000) {
        s = '#004b1f'
      }

      if (k >= 1000 && k < 2000) {
        s = '#ffea04'
      }

      if (k >= 2000 && k < 3000) {
        s = '#ffc603'
      }

      if (k >= 3000 && k < 4000) {
        s = '#ffa203'
      }

      if (k >= 4000 && k < 6000) {
        s = '#ff1018'
      }

      if (k >= 6000 && k < 8000) {
        s = '#960001'
      }

      if (k >= 8000 && k < 10000) {
        s = '#7a0000'
      }

      if (k >= 10000) {
        s = '#340000'
      }

      rects.push(new google.maps.Rectangle({
        strokeColor: '#303030',
        strokeOpacity: 0,
        strokeWeight: 0,
        fillColor: s,
        fillOpacity: 0.6,
        editable: false,
        draggable: false,
        map: map,
        indexID: locations[i][0],
        bounds: {
          north: locations[i][3],
          south: locations[i][2],
          east: locations[i][5],
          west: locations[i][4]
        }
      }));
    }

    for (var i = 0, j = rects.length; i < j; i++) {
      google.maps.event.addListener(rects[i], 'click', rectHandler);
    }

    //
    function rectHandler(event) {
      //  alert(this.indexID);

      if (pred !== -1) {
        rects[pred].setOptions({ strokeColor: '#303030' });
        rects[pred].setOptions({ strokeWeight: 0 });
        rects[pred].setOptions({ strokeOpacity: 0 });
      }
      this.setOptions({ strokeColor: '#004b1f' });
      this.setOptions({ strokeWeight: 1 });
      this.setOptions({ strokeOpacity: 0.5 });
      pred = this.indexID - 1;

      //  document.getElementById('test').innerHTML = this.getBounds().getNorthEast().lat();
      var latUp = this.getBounds().getNorthEast().lat();
      var latDown = this.getBounds().getSouthWest().lat();
      var lonUp = this.getBounds().getNorthEast().lng();
      var lonDown = this.getBounds().getSouthWest().lng();
      var latlng = new google.maps.LatLng(latDown + (latUp - latDown) / 2, lonDown + (lonUp - lonDown) / 2);
      marker.setPosition(latlng);

    }
}
google.maps.event.addDomListener(window, 'load', initialize);
```

Рис. 9. Работа с картой Google Maps на JavaScript

В данном коде мы использовали вызов процедур Google Maps JavaScript API. Для использования данного набора API необходимо зарегистрироваться на платформе Google Cloud и получить доступ к специальному ключу и указать его в качестве параметра при вызове процедуры создание объекта Google Maps (Рис.10).

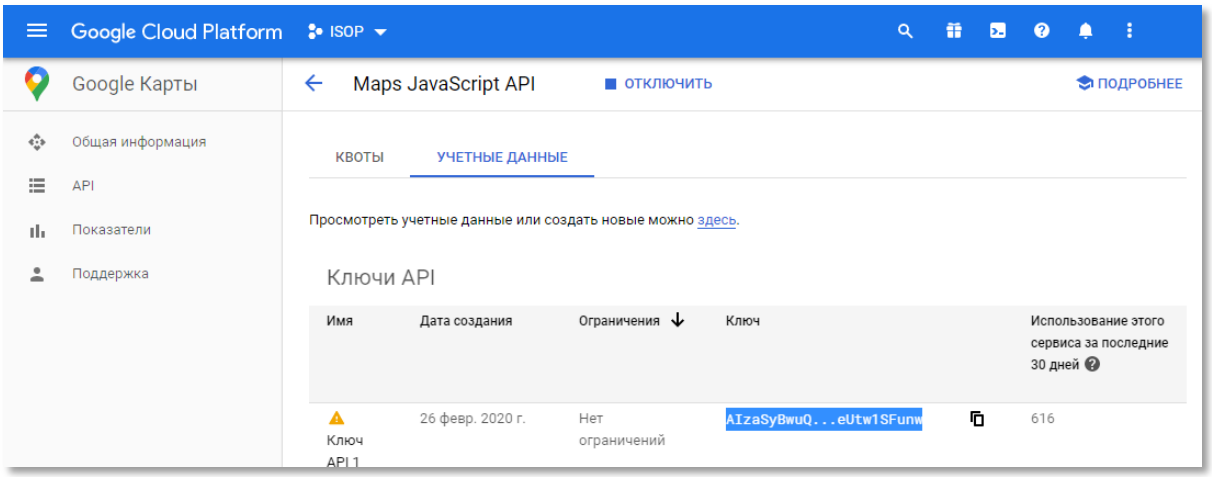


Рис. 10. Работа с платформой Google Cloud

Для отрисовки полигонов и раскраски их в соответствие с индексом пожарной опасности с помощью SQL запросов создается специальный подключаемый модуль JS, Data.js. Модуль содержит координаты полигонов и их индекс пожарной опасности (Рис.11).

```
var locations = [
  [1,827.842051,55.875759,55.938262,120.499997,120.5625],
  [2,827.842051,55.938262,56.000765,120.499997,120.5625],
  [3,827.842051,56.000765,56.063268,120.499997,120.5625],
  [4,827.842051,56.063268,56.125771,120.499997,120.5625],
  [5,600.975551,56.125771,56.188273,120.499997,120.5625],
  [6,600.975551,56.188273,56.250776,120.499997,120.5625],
  [7,600.975551,56.250776,56.313279,120.499997,120.5625],
  [8,600.975551,56.313279,56.375782,120.499997,120.5625],
  [9,294.390851,56.375782,56.438285,120.499997,120.5625],
  [10,294.390851,56.438285,56.500788,120.499997,120.5625],
  [11,294.390851,56.500788,56.563291,120.499997,120.5625],
  [12,294.390851,56.563291,56.625794,120.499997,120.5625],
  [13,780.418951,56.625794,56.688296,120.499997,120.5625],
  [14,780.418951,56.688296,56.750799,120.499997,120.5625],
  [15,780.418951,56.750799,56.813302,120.499997,120.5625],
  [16,780.418951,56.813302,56.875805,120.499997,120.5625],
  [17,657.463251,56.875805,56.938308,120.499997,120.5625],
  [18,657.463251,56.938308,57.000811,120.499997,120.5625],
  [19,657.463251,57.000811,57.063314,120.499997,120.5625],
```

Рис. 11. Модуль Data.js на Javascript

Основной интерфейс клиентской части информационной системы представлен следующим скриншоте (Рис.12).

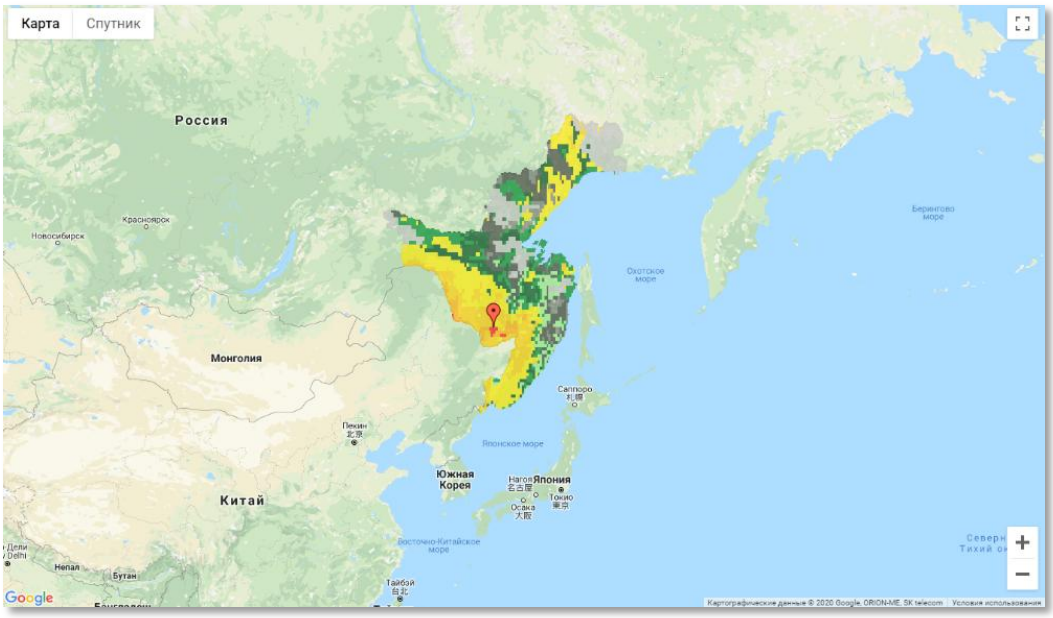


Рис. 12. Интерфейс клиентской части ИС

Таким образом, работе описаны способы визуализации индексов пожарной опасности на картах Google Maps.

Библиографический список

1. Andy, Harris HTML, XHTML and CSS All–In–One For Dummies® / Andy Harris. M.: Наука, 2014. 173 с.
2. David Sawyer McFarland, "JavaScript and jQuery - The missing manual", 3rd ed, 2014 503 с.
3. Google Maps Platform // Docs. URL:

https://developers.google.com/maps/documentation/javascript/tutorial?hl=ru (дата обращения 19.4.2020).
4. Google Maps Platform Documentation // Google Maps API Documentation URL: https://developers.google.com/maps/documentation (дата обращения 10.10.2019).
5. habr // Google Map API (custom labels and tooltips) URL: https://habr.com/ru/post/131654/ (дата обращения 19.10.2019).
6. Chaffer J. and Swedbery K. Learning jQuery. Packt Publishing, 2013. 428 с.
7. Cantù M. Delphi XE Handbook: A Guide to New Features in Delphi XE. CreateSpace Independent Publishing Platform, 2011. 132 с.
8. Дакетт Дж. Основы веб-программирования с использованием HTML, XHTML и CSS. М.: Эксмо, 2013. 768 с.
9. Дронов В. JavaScript в Web-дизайне. М.: БХВ-Петербург, 2017. - 880 с
10. Закас Н. JavaScript для профессиональных веб-разработчиков. М.: Питер, 2015. 831 с.
11. Чекко Р. Графика на JavaScript. М.: Питер, 2017. 733 с