**Министерство образования и науки Российской федерации  
Федеральное государственное бюджетное образовательное учреждение высшего образования «Московский государственный университет технологий и управления имени К.Г. Разумовского (Первый казачий университет)». Университетский колледж информационных технологий**

Специальность 09.02.03 «Программирование в компьютерных системах»

КУРСОВОЙ ПРОЕКТ

МДК 03.01. Технология разработки программного обеспечения

ПМ.03 Участие в интеграции программных модулей

на тему разработка модуля Яндекс.Карты API для расчета основных показателей расчета

**Пояснительная записка**

**УКИТ 09.02.03.2016.304.06ПЗ**

|  |  |  |
| --- | --- | --- |
| Группа | П-404 |  |
| Студент | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (личная подпись) | Головешкин Р. И. |
| Руководитель проекта | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (личная подпись) | Кузьменко С. Ю. |

**СОДЕРЖАНИЕ**

[ВВЕДЕНИЕ](https://docs.google.com/document/d/1s3GVdNljtSfAVjBuzt2c-z2rgfwpco26AeggsEBnKbE/edit#heading=h.gjdgxs) 3

[ОСНОВНАЯ ЧАСТЬ 5](https://docs.google.com/document/d/1s3GVdNljtSfAVjBuzt2c-z2rgfwpco26AeggsEBnKbE/edit#heading=h.30j0zll)

[1 Введение 5](https://docs.google.com/document/d/1s3GVdNljtSfAVjBuzt2c-z2rgfwpco26AeggsEBnKbE/edit#heading=h.1fob9te)

[2 Основание для разработки](https://docs.google.com/document/d/1s3GVdNljtSfAVjBuzt2c-z2rgfwpco26AeggsEBnKbE/edit#heading=h.tyjcwt) 6

[3 Назначение разработки](https://docs.google.com/document/d/1s3GVdNljtSfAVjBuzt2c-z2rgfwpco26AeggsEBnKbE/edit#heading=h.4d34og8) 6

[4 Требования к программе или программному изделию](https://docs.google.com/document/d/1s3GVdNljtSfAVjBuzt2c-z2rgfwpco26AeggsEBnKbE/edit#heading=h.3rdcrjn) 6

5 [Цель испытаний 7](https://docs.google.com/document/d/1s3GVdNljtSfAVjBuzt2c-z2rgfwpco26AeggsEBnKbE/edit#heading=h.49x2ik5)

6 [Требования к программе 8](https://docs.google.com/document/d/1s3GVdNljtSfAVjBuzt2c-z2rgfwpco26AeggsEBnKbE/edit#heading=h.2p2csry)

7 [Требования программной документации 8](https://docs.google.com/document/d/1s3GVdNljtSfAVjBuzt2c-z2rgfwpco26AeggsEBnKbE/edit#heading=h.1hmsyys)

8 [Методы испытаний 10](https://docs.google.com/document/d/1s3GVdNljtSfAVjBuzt2c-z2rgfwpco26AeggsEBnKbE/edit#heading=h.37m2jsg)

9 [Технический проект 1](https://docs.google.com/document/d/1s3GVdNljtSfAVjBuzt2c-z2rgfwpco26AeggsEBnKbE/edit#heading=h.4k668n3)5

10 [Реализация программного изделия на языке программирования 1](https://docs.google.com/document/d/1s3GVdNljtSfAVjBuzt2c-z2rgfwpco26AeggsEBnKbE/edit#heading=h.34g0dwd)6

11 [Тестирование программного продукта](https://docs.google.com/document/d/1s3GVdNljtSfAVjBuzt2c-z2rgfwpco26AeggsEBnKbE/edit#heading=h.xvir7l) 17

[ЗАКЛЮЧЕНИЕ](https://docs.google.com/document/d/1s3GVdNljtSfAVjBuzt2c-z2rgfwpco26AeggsEBnKbE/edit#heading=h.2w5ecyt) 19

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ](https://docs.google.com/document/d/1s3GVdNljtSfAVjBuzt2c-z2rgfwpco26AeggsEBnKbE/edit#heading=h.1baon6m) 21

[ПРИЛОЖЕНИЕ А РУКОВОДСТВО ОПЕРАТОРА](https://docs.google.com/document/d/1s3GVdNljtSfAVjBuzt2c-z2rgfwpco26AeggsEBnKbE/edit#heading=h.3vac5uf) 22

[ПРИЛОЖЕНИЕ Б ИСХОДНЫЙ КОД](https://docs.google.com/document/d/1s3GVdNljtSfAVjBuzt2c-z2rgfwpco26AeggsEBnKbE/edit#heading=h.3mzq4wv) 24

[ПРИЛОЖЕНИЕ В ПРОТОКОЛ СИСТЕМЫ КОНТРОЛЯ ВЕРСИЙ](https://docs.google.com/document/d/1s3GVdNljtSfAVjBuzt2c-z2rgfwpco26AeggsEBnKbE/edit#heading=h.2250f4o) 28

[ПРИЛОЖЕНИЕ Г РУКОВОДСТВО ПРОГРАММИСТА](https://docs.google.com/document/d/1s3GVdNljtSfAVjBuzt2c-z2rgfwpco26AeggsEBnKbE/edit#heading=h.haapch) 28

[ПРИЛОЖЕНИЕ Д ДОКЛАД](https://docs.google.com/document/d/1s3GVdNljtSfAVjBuzt2c-z2rgfwpco26AeggsEBnKbE/edit#heading=h.40ew0vw) 30

ПРИЛОЖЕНИЕ Ж ПРЕЗЕНТАЦИЯ 31

# ВВЕДЕНИЕ

Благодаря компьютерам стало автоматизировать рутинную работу человека(менеджера). И сделать его работу более комфортной и быстрой, что очень важно для потребителя услуг.

В наше время есть много программ с различными решениями задач, но некоторые из них долго работают или выполняют не те функции и т.п.

И так как для предприятия ООО”АртМеталл” нужно средство определения расстояния кратчайшего расстояния от цели до ближайшей точки на МКАД и так же некоторые параметры(расстояние, место доставки и т.п), то было принято решение создать данное веб-приложение для компании, с целью использование на предприятии.

Основные задачи, которые были поставлены при разработке игры Яндекс.Карты API:

* Анализ предметной области;
* Выбор средств реализации;
* Изучение API Яндекс.Карты;
* Изучение vue.js, global scope;
* Создание программной документации к веб-приложению Яндекс.Карты API.

Используемые методы при разработке Яндекс.Карты API:

* Анализ;
* Изучение дополнительной литературы;
* Изучение необходимых фреймворков и API Яндекс.Карты.

Структура курсового проекта:

Пояснительная записка составлена таким образом, чтобы была возможность отследить программный продукт на всем жизненном цикле. А точнее от появления идеи о создании и до предоставления результатов тестирования и выводах о готовности программного обеспечения к выпуску.

1. Основная часть:

Раздел описывает предметную область, для которой разрабатывается Яндекс.Карты API.

1. Спецификация:

Раздел спецификации описывает курсовой проект веб-приложения Яндекс.Карты API, определяет его предметную область, основной функционал программы и условия эксплуатации, в которых данная программа будет стабильно работать. Еще в данном разделе описаны требования к программной документации, которая определяет другие характеристики программы курсового проекта.

1. Программа и методика испытаний:

Раздел описывает применяемые методы тестирования и тестовые примеры к разрабатываемой Яндекс.Карты API.

1. Технический проект:

Раздел описывает архитектурные решения, который применяются при разработке.

1. Реализация программного продукта:

Раздел описывает процесс разработки Яндекс.Карты API.

1. Тестирование программного продукта:

Раздел описывает результаты тестирования Яндекс.Карты API по программе и методике испытаний.

1. Заключение:

Раздел подводит итог по проделанной работе, о навыках, которые были получены при разработке веб-приложения Яндекс.Карты API.

1. Список используемых источников:

Раздел приводиться список используемых источников, которые использовались при разработке Яндекс.Карты API.

1. Приложения:

Раздел приводятся дополнительная информация, которая включает в себя руководство оператора, исходный код приложения, протокол системы контроля версий, руководство программиста, доклад.

Анализ предметной области

Элементы данного курсового:

* Менеджер(пользователь программы, решившие получить интересующие данные предъявляемым веб-приложением);
* Администратор(поддерживает программу в актуальном и работающем состоянии).

Основная функция данной веб-программы - рассчитать данные для менеджера и визуализировать это. Данные для менеджера заключаются в расстоянии(км) между точкой заказчика и ближайшей точкой МКАД(рассчитывается автоматически).

Итог: Нужно создать веб-приложения для менеджера, которая будет автоматически определять ближайшую точку на МКАД, растояния(км) между точками от заказчика до ближайшей точки МКАД.

# ОСНОВНАЯ ЧАСТЬ

## Введение

### Наименование программы

Разработка модуля Яндекс.Карты API для расчета основных показателей маршрута поездки.

### Область применения

Сфера логистики доставки товара\груза.

### Объект, в котором используется программа

Компания ООО”АртМеталл”, так же может быть использована на ПК пользователя с характеристиками, описанными в пункте 4.2 Основной.

## Основание для разработки

Целью создания данного курсового проекта послужило необходимость данной веб-программы для предприятия ООО”АртМеталл”.

* 1. **Обоснование разработки**

Веб-программа нужна для бизнеса. Логистика доставки товара.

### Документ, на основании которого ведется разработка

Техническое задание на курсовой проект.

### Наименование и (или) условное обозначение темы для разработки

Веб-приложение ”DYM API” (Delivery Yandex.Map API).

1. **Срок представление работы к защите**

Срок сдачи курсового проекта до 14 Апреля 2018 года.

1. **Формирование и анализ требование к ПО**

### Функциональное назначение

Расчет кратчайшего маршрута для грузовой машины от заказчика до МКАД, и получение параметров(расстояние, визуализация и т.д).

### Эксплуатационное назначение

DYM API предназначена для сотрудников ООО”АртМеталл”, которым нужно быстро и точно определить расстояние между покупателем и ближайшей точки МКАД.

### Проектирование ПО. Выбор технических и программных средств для решения задач

### Требования к составу и параметрам технических средств

В состав технических средств должен входить компьютер клиента, включающий:

* Процессор Pentium 4 или выше;
* Клавиатура и мышь;
* Видеокарта;
* Монитор;
* HDD объёмом свободного пространства не менее 1 ГБ;
* Windows XP и выше;
* Доступ на корпоративный портал ООО“АртМеталл”;
* Доступ к сети интернет.

### Требования к информационным структурам на входе и выходе не предъявляются

### Требования к методам решения

Веб-приложение ”DYM API” должно быть разработано с использованием следующий технологий:

* HTML(единственный и неповторимый);
* JavaScript(для динамическую интерактивности в проекте);
* CSS(для кастомизации проекта в браузере);
* Vue.js(для асинхронного обновление данных в приложении);
* Yandex.map API(имеет документацию по разработке);
* Sublime Text 3(удобный текстовый редактор с кучей полезных плагинов для быстрой разработки веб-приложений).

### Требования к программной документации

* 1. **Состав программной документации**

Состав программной документации должен включать:

* Техническое задание;
* Пояснительная записка;
* Программный код;
* Программа и методика испытания;
* Руководство пользователя;
* Руководство программиста.

### Специальные требования к пояснительной записке

Специальные требования не предъявляются.

### Стадии и этапы разработки

### Стадии разработки

Разработка осуществляется в три стадии:

* Техническое задание;
* Технический проект;
* Рабочий проект.

### Этапы разработки

На стадии техническое задание осуществляется разработка, согласование и утверждение технического задания.

На стадии технический проект осуществляется разработка, согласование и утверждение пояснительной записки.

На стадии рабочий проект осуществляется разработка текста программы.

### Порядок контроля и приёмки

### Приёмосдаточные испытания должно проводиться в соответствии с программой и методикой испытаний, разработанной, согласованной и утвержденной не позднее 30 Марта 2018 года

### Программа и методика испытания

### Введение

Данный раздел описывает основные методы тестирования для проверки соответствия программного продукта функциональным требованиям.

### Наименование

Веб-приложение ”DYM API”

1. **Проверка соответствия программного продукта требованиям технического задания.**

## Требования к программе

### Требования к функциональным характеристикам

### Требования к составу выполняемых функций

Программа должна обеспечивать выполнение следующих функций – пункт 7.1 Технического задания.

### Требования к информационной и программной совместимости

Пункт 7.1 Технического задания

## Требования программной документации

### Состав программной документации

Состав программной документации должен включать:

* Техническое задание;
* Пояснительная записка;
* Программный код;
* Программа и методика испытания;
* Руководство пользователя;
* Руководство программиста.

### Специальные требования к пояснительной записке

Специальные требования не предъявляются пункт 5.2 Технического задания

### Требования к исходным кодам изложены в документе:

А. И. Глускер «Сборник задач по курсу “Основы алгоритмизации и программирования” [Электронный ресурс] – 2011 /раздел 3.1/ (п. 4.5.3 технического задания)

### Технические средства, используемые при проведении испытаний

### В состав технических средств должен входить IBM-совместимый компьютер, включающий:

* Процессор;
* Видеокарту;
* Клавиатуру и мышь;
* Монитор;
* Жёсткий диск;
* Доступ на корпоративный портал ООО“АртМеталл”
* Подключение к сети интернет.

### Программные средства, используемые при проведении испытаний

### В состав программных средств должно входить:

* Операционная система семейства Windows (Начиная с Windows XP);

### Порядок проведения испытаний

### Подготовка к проведению испытаний заключается в обеспечении наличия компьютера в п. 5.1, и программных средств, указанных в п. 5.2, установленных на этом компьютере

### Ход проведения испытаний документируется в протоколе, где указывается перечень проводимых испытаний, результат каждого испытания и возможно замечания.

### Состав испытания

### Проверка состава программной документации в соответствии с методом, описанном в п. 6.2

### Проверка требований к программе

### Проверка обеспечений требования к программе (п. 3) в соответствии с методом, описанным в п. 6.1

## Методы испытаний

### Метод проверки требований к программе

Проверка осуществляется путём запуска программы и сравнения ожидаемых результатов приведенных в (Таблица 1).

### Метод проверки требований к составу программной документации

Проверка состава программной документации осуществляется визуально путем сравнения набора предъявленных документов (в форме распечатки или в рукописной форме) списку, приведенному в п. 4.1. При этом исходные тексты программ должны быть предоставлены также и в электронной форме.

В случае если набор предъявленных документов соответствует списку, а исходные тексты предоставлены также в электронной форме, то в протокол заносится запись: «Состав программной документации» – соответствует; в противном случае: «Состав программной документации» – не соответствует.

### Метод проверки требований к пояснительной записке

Проверка состоит из следующих этапов:

* проверка наличия блок-схемы (блок-схем) в пояснительной записке;
* проверка соблюдения требований ГОСТ 19.701-90 для каждой блок-схемы;
* проверка соблюдения локальных стандартов для блок-схем;
* проверка соответствия каждой блок-схемы алгоритму, закодированному в программе.

Проверка соблюдения требований ГОСТ 19.701-90 состоит из следующих работ:

* проверка использования только тех символов, которые указаны как применимые к схемам программ в п. 5 ГОСТ 19.701-90;
* проверка соответствия символов их назначению (экспертная оценка лица, проводящего испытания);
* проверка правильности выполнения соединения линий (п. 4.2.3 ГОСТ 19.701-90);
* проверка того, что линии потока управления, выходящие из символа «решение» подписана (п. 4.3.1.2 ГОСТ 19.701-90);

Проверка соблюдения локальных стандартов для блок-схем состоит из следующих работ:

* проверка того, что все символы (кроме терминаторов, соединителей, линий и комментариев) имеют одинаковые размеры;
* проверка того, что терминаторы имеют ту же ширину, что и другие символы;
* проверка того, что отношение ширины к высоте составляет 2 к 1 для каждого символа, кроме терминаторов, комментариев и линий;
* проверка того, что отношение ширины к высоте составляет 4 к 1 для терминаторов;
* проверка того, что высота соединителей совпадает с высотой терминаторов;
* проверка того, что линии потока управления входят в символ слева или сверху, а выходят снизу или справа;
* проверка того, что подписи к линиям не находятся на самих линиях.

Проверка соответствия каждой блок-схемы алгоритму, закодированному в программе, осуществляется путем экспертной оценки лицом, осуществляющим проведение испытаний. В случае, если все вышеприведенные проверки прошли успешно, в протокол заносится запись: «Специальные требования к пояснительной записке» – соответствует; в противном случае «Специальные требования к пояснительной записке» – не соответствует.

### Метод проверки к исходным кодам

Изложенный ниже метод применяется ко всем файлам, содержащим исходный текст, и входящим в состав программной документации по отдельности. Для каждого файла вносится в протокол запись: «Требования к исходным кодам для файла ##### – соответствует/не соответствует (где вместо #### указывается название файла).

Проверка состоит из следующих этапов:

* Проверка исходных кодов, написанных на HTML/JAVASCRIPT/CSS;
* Наличие комментария в начале файла, содержащего автора работы, номера задания и варианта, краткой формулировки задания (или его части);
* Наличие комментариев к неочевидным действиям (проверяется методом экспертной оценки лицом, осуществляющим испытания);
* Для каждой подпрограммы наличие комментария, содержащего полное описание ее работы, описание всех аргументов и результатов. Достаточность этого комментария для возможности использовать подпрограмму в других программах (без изучения собственно текста подпрограммы);
* Для каждой глобальной переменной указание ее назначения;
* Для всех переменных, кроме переменных цикла, использование «говорящих» названий;
* Для всех подпрограмм использование говорящих названий;
* Использование одного оператора на одной строке программы;
* Количество табулирования перед строкой программы должно соответствовать уровню вложенности;
* Операторы { и } , соответствующие друг другу, располагаются строго с одной и той же позиции по вертикали;
* Использование модулей для трех и более сходных по назначению подпрограмм;
* Отсутствие в подпрограммах использования глобальных переменных (напрямую);
* Разделение подпрограмм на предназначенные для вычислений (в них не должно быть ввода-вывода) и на предназначенные для ввода-вывода (в них вычисления должны быть только такие, что нужны для ввода-вывода);

Таблица 1 – Тестовые примеры

|  |  |  |  |
| --- | --- | --- | --- |
| **№** | **Предусловия** | **Действие** | **Ожидаемые результаты** |
| 1 | Зайти в веб приложение | При нажатии на карту | Должен построится маршрут от точки пользователя до ближайшей точки МКАД |
| 2 | После нажатия на карту | Ожидание в течении <1-2 секунд пользователя | В другом div блоке (снизу) должна появиться информация о расстояния между А(точкой пользователя) до Б (ближайшей точки МКАД) |
| 3 | Точка на карте уже поставлены (маршрут построен) | Пользователь ставит точку в другом месте | Предыдущий построенный маршрут перестраивается на новые данные полученные от пользователя и данный маршрут построен заново |
| 4 | Маршрут построен, но пользователь нажал на другое место на карте. | Ожидание пользователя <1-2 сек | Данные о маршруте меняются(расстояние и т.д) |

### Вывод

В этом разделе были изложены методы, которые будут применяться к веб-приложение ”DYM API” для проверки соответствия требованиям технического задания.

## Технический проект

### Введение

В данном разделе изложены и алгоритмы, используемые в веб-приложение ”DYM API”

### Наименование программы

Веб-приложение ”DYM API”

### Условное обозначение темы разработки

### Документ, на основании которого ведётся разработка

Техническое задание на курсовой проект.

### Назначение и область применения

### Назначение программы

Расчет ближайшей точки от пользователя до МКАД, и получение параметров(расстояние, координаты и т.п).

### Краткая характеристика области применения

Менеджерам компании ООО”АртМеталл ”.

### Технические характеристики

### Постановка задачи на разработку

### Создание веб-приложения ”DYM API” позволит пользователям:

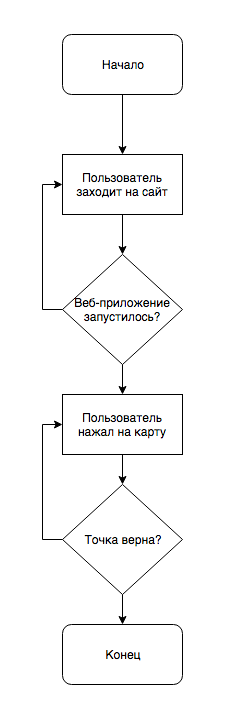
* Ускорить работу вычислений маршрута;
* Упростить восприятия(за счет визуализации);
* Узнать необходимые параметры, для дальнейшей работы с ними.

### Функциональная схема

### 

### Описание алгоритма

Алгоритм (блок-схема) с работой одной из функций курсового проекта.



### Аппаратные средства

ПК пользователя должен обладать следующими техническими характеристиками:

* Процессор: Pentium и выше;
* ОЗУ: 2GB и выше;
* HDD: 4GB и выше.
* Интернет 2-4 mb/s

### Программные средства

Для создания приложение использовалась Sublime Text 3, это обусловлено тем, что с этим текстовым редактором удобно работать и так же он имеет большинство вспомогательных для разработки плагинов.

### Описание используемых технологий

Для разработки приложения использовался Yandex.map API:

1. Программируемая карты.

Для разработки приложения использовался фреймворк Vue.js:

1. Асинхронное обновление ;
2. Более удобный фреймворк для разработки такого типа приложений;

### Вывод

Раздел описывает проектировочную часть разрабатываемого веб-приложения ”DYM API”.

## Реализация программного изделия на языке программирования

### Введение

В этом разделе описывается разработка веб-приложения ”DYM API”.

### Процесс реализации

Разработка веб-приложения началась в январе 2018 года и был реализован с помощью Yandex.map API на языках HTML/CSS/JAVASCRIPT.

Срок сдачи курсового проекта был 14.04.2018. Разработка веб-приложения ”DYM API” была начата со сбора информации о проекте, о методах реализации некоторых функций и т.д. После сбора информации было принято решение создание простой веб-страницы для выполнения данного курсового проекта. Дальше были созданы формы,<div class="col-md-8"><div id="map" style="height:500px;width: 100%;"></div></div>......, после этого добавили API Яндекс.Карты<script src="https://api-maps.yandex.ru/2.1/?lang=ru\_RU" type="text/javascript"></script>. После данного этапа были завершены расчеты данных, визуальное построение и т.д(данный текст кода не может быть предоставлен в связи закрытой разработки).

### Вывод

В предыдущем этапе было описана реализация курсового проекта программно.

## Тестирование программного продукта

### Введение

Тестирование исходного кода путём написания юнит-тестов не производилось.

### Протоколы прохождения тестов

Все тестовые примеры были пройдены успешно. Ниже на Таблице 2 приведены результаты некоторых тестов.

Таблица 2 – Тестовые примеры

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **№** | **Предусловия** | **Действие** | **Ожидаемые результаты** | **Результат** |
| 1 | Зайти в веб приложение | При нажатии на карту | Должен построится маршрут от точки пользователя до ближайшей точки МКАД | Положительный |
| 2 | После нажатия на карту | Ожидание в течении <1-2 секунд пользователя | В другом div блоке (снизу) должна появиться информация о расстояния между А(точкой пользователя) до Б (ближайшей точки МКАД) | Положительный |
| 3 | Точка на карте уже поставлены (маршрут построен) | Пользователь ставит точку в другом месте | Предыдущий построенный маршрут перестраивается на новые данные полученные от пользователя и данный маршрут построен заново | Положительный |
| 4 | Маршрут построен, но пользователь нажал на другое место на карте. | Ожидание пользователя <1-2 сек | Данные о маршруте меняются(расстояние и т.д) | Положительный |

### Вывод

В данном разделе были описаны методы тестирования, которые использовались для проверки соответствия требованиям технического задания.

* 1. **Оценка эффективности созданного продукта**

Данная веб-программа полезна компании ООО”АртМеталл” и используется менеджерами компании. В данный момент она все еще имеет статус бета, поэтому она тестируется менеджерами и в случае нахождение бага, данный ошибки программы исправляются с учетом требований пользователей программы(менеджерами).

В итоге на выходе всей разработки программы, я получил работающий продукт, который помогает менеджерам избавляться от рутинной работы и ускорить их работу.

# ЗАКЛЮЧЕНИЕ

Результатом работы оказался разработанное веб-приложение ”DYM API”, которое полностью реализовано и выполняет все заявленные функции.

Мои личные выводы по данному курсовому проекту таковы, не ленись и все получится и перед выполнении курсового проекта надо распланировать свое время и учесть в нем не только написание программной части, но и изучение дополнительного материала и иметь свой собственный дедлайн, который будет раньше, чем у заказчика\преподавателя для сдачи курсового в срок.

В плане профессионального развития мне понравился данный курсовой проект, за счет которого я немного приобретаю навыки программиста, который умеет общаться с профессиональной документацией, обсуждать свои проблемы на профильных форумах, а не сидеть сломя голову и также общаться с другими программистами по поводу программы, а именно выслушивать критику и т.д.

# СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Yandex.Map API. [Электронный ресурс]. – Режим доступа: https://tech.yandex.ru/maps/... (16.02.2018г.)
2. Сайт общение между программистами. [Электронный ресурс]. Режим доступа: https://ru.stackoverflow.com/(25.02.2018г.)

# ПРИЛОЖЕНИЕ А

**Руководство оператора**

1. **Назначение программы**

### Наименование

Веб-приложение ”DYM API”

### Назначение

Сфера доставки расчета пути от продавца до покупателя.

### Характеристика области применения

Менеджер компании ООО”АртМеталл”.

1. **Условия выполнения программы**

### Аппаратные средства

В состав аппаратных средств пользователя должен входит компьютер, который включает в себя:

* Процессор;
* Видеокарта;
* Жёсткий диск;
* Сетевая карта;
* Интернет соединения

### Программные средства

В состав программных средств пользователя должен входить интернет браузер.

1. **Выполнение программы**

### Запуск программы

Для запуска программы необходимо открыть crm систему ООО”АртМеталл”, и перейти в отдел калькуляции которой расположено веб-приложение ”DYM API”.

### Руководство пользователя

Интерфейс интуитивно понятный так же, как и управление персонажем**.**

# ПРИЛОЖЕНИЕ Б

**Исходный код**

Исходный код веб-приложение ”DYM API” в файле index.html.

Представлено только часть кода, так как данная веб программа разрабатывалась для использование в компании ООО”АртМеталл” и все ее права защищены .

var vivod\_array\_dist = sr\_km(coords,close\_point);

function compareNumeric(a, b) {

if (a.distanse > b.distanse) return 1;

if (a.distanse < b.distanse) return -1;

}

// Если метка уже создана – просто передвигаем ее.

if (myPlacemark) {

myPlacemark.geometry.setCoordinates(coords);

} // Если нет – создаем.

else {

myPlacemark = createPlacemark(coords);

myMap.geoObjects.add(myPlacemark);

// Слушаем событие окончания перетаскивания на метке.

myPlacemark.events.add(function () {

getAddress(myPlacemark.geometry.getCoordinates());

}); }

getAddress(coords);

var \_min\_distanse = 10000000\*10000000000000000000000000000000000000000000000;

var mkad\_km = [[55.611144,37.491672],[55.581012,37.572181],[55.576052,37.596728],[55.575274,37.688567],[55.59122,37.729079],[55.640671,37.820918],[55.657075,37.839286],[55.687343,37.831046],[55.707702,37.835509],[55.712572,37.838084],[55.731077,37.841002],[55.743862,37.841861],[55.777158,37.843234],[55.814099,37.839114],[55.882762,37.726504],[55.89212,37.706077],[55.895303,37.673633],[55.910155,37.588489],[55.908323,37.544028],[55.907262,37.537505],[55.887297,37.483603],[55.881604,37.445838],[55.87649,37.427642],[55.870989,37.412021],[55.83275,37.395884],[55.789927,37.371852],[55.76574,37.36859],[55.713638,37.38507],[55.701522,37.398631],[55.662048,37.432792],[55.638558,37.458884]];

var myMap;

function directgeocord(adress){

ymaps.geocode(adress, {

results: 1

}).then(function (res) {

// Выбираем первый результат геокодирования.

var firstGeoObject = res.geoObjects.get(0),

// Координаты геообъекта.

coords = firstGeoObject.geometry.getCoordinates(),

// Область видимости геообъекта.

bounds = firstGeoObject.properties.get('boundedBy');

firstGeoObject.options.set('preset', 'islands#darkBlueDotIconWithCaption');

// Получаем строку с адресом и выводим в иконке геообъекта.

firstGeoObject.properties.set('iconCaption', firstGeoObject.getAddressLine());

// Добавляем первый найденный геообъект на карту.

myMap.geoObjects.add(firstGeoObject);

// Масштабируем карту на область видимости геообъекта.

myMap.setBounds(bounds, {

// Проверяем наличие тайлов на данном масштабе.

checkZoomRange: true

});

})

<link rel="stylesheet" type="text/css" href="css/main.css">

<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css" integrity="sha384-Gn5384xqQ1aoWXA+058RXPxPg6fy4IWvTNh0E263XmFcJlSAwiGgFAW/dAiS6JXm" crossorigin="anonymous">

<!--Для прелоадер -->

<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/normalize/5.0.0/normalize.min.css">

<!--/Для прелоадер -->

<style type="text/css">

body {

background: #ffffff;

margin: 3%;

}

</style>

<script src="https://api-maps.yandex.ru/2.1/?lang=ru\_RU" type="text/javascript"></script>

</head>

<body>

<div id="delivery">

<div class="container-fluid">

<div class="row">

<div class="col-md-8">

<div id="map" style="height:500px;width: 100%;"></div>

</div>

<div class="col-md-4">

<h1>Параметры для расчета:</h1>

<p>Место доставки:</p>

<div>

<input class="form-control form-control-lg" type="text" placeholder="Адрес доставки" v-on:change="geo" v-model="address\_to\_customer" id="suggest1">

</div>

</div>

</div>

<div class="row">

<div class="col-md-6">

<h1>Общая информация о Маршруте:</h1>

<p>Расстояние до цели от МКАД: {{ dlinna\_km }} км.</p>

<p>Место доставки: {{address\_to\_customer}}</p>

<!-- Изменение значения себестоимости <input v-model = 'amount\_trub\_fct' type="text" /> -->

</div>

</div>

</div>

</div>

select {

min-width: 300px;

}

h3,h1{

padding: 1% 1% 2% 1%;

}

.row{

padding: 0.1%;

margin: 60px auto 30px;

padding: 15px;

position: relative;

/\* Styles \*/

background: #fffaf6;

border-radius: 4px;

color: black;

box-shadow:

0 2px 2px rgba(0,0,0,0.2),

0 1px 5px rgba(0,0,0,0.2),

0 0 0 12px rgba(226, 204, 204,0.4);

}

.negative\_cars{

box-shadow:10px 15px 40px rgba(0,0,0,0.3);

margin: 8px;

}

.positiv\_cars{

position: relative;

margin: 8px;

background-color: #fff;

transition:0.3s;

}

.positiv\_cars:hover {

transform: scale(1.1);

transition:1.7s;

z-index: 1000;

}

.col-md-4{

border-left: 1px double black;

}

.col-md-8{

}

.col-md-6{

}

.cursor{

cursor: pointer;

}

#sch\_image{

margin-top:2%;

transition:1s;

}

# ПРИЛОЖЕНИЕ В

**Протокол системы контроля версий**

Таблица B.1 – Протокол системы контроля версий

|  |  |
| --- | --- |
| **Summary** | **Description** |
| **Commits on Jan 15, 2018** | |
| Создание папок |  |
| Структурирование данных |  |
| [Анализ](https://github.com/pmswga/EDUKIT/commit/d4eb141e29f2f4f9acdb5ef78a527707f099c111) предметной области |  |
| Docum |  |
| **Commits on Jan 17, 2018** | |
| Programm |  |
| **Commits on Jan 20, 2018** | |
| Fix program |  |
| Smth things |  |
| [image](https://github.com/pmswga/EDUKIT/commit/62326a771caa52359d3d0979b5bd04f877e64ca9) complete |  |
| **Commits on Feb 15, 2018** | |
| small fix |  |
| Fix |  |
| [write documents](https://github.com/pmswga/EDUKIT/commit/e2bff22526e13604c0c918749b7de5d95a1be452) |  |
| **Commits on Feb 18, 2018** | |
| Something |  |
| [smth](https://github.com/pmswga/EDUKIT/commit/916e3e2dd8477ebc5f4d03294e0af111b6f96c2e) |  |
| **Commits on April 7, 2018** | |
| documentation |  |
| **Commits on April 10, 2018** | |
| Doc fix |  |

# ПРИЛОЖЕНИЕ Г

**Руководство программиста**

Руководство содержит следующие основные разделы:

* Окно веб-приложения (Рисунок Г.1);
* Файлы (Рисунок Г.2).

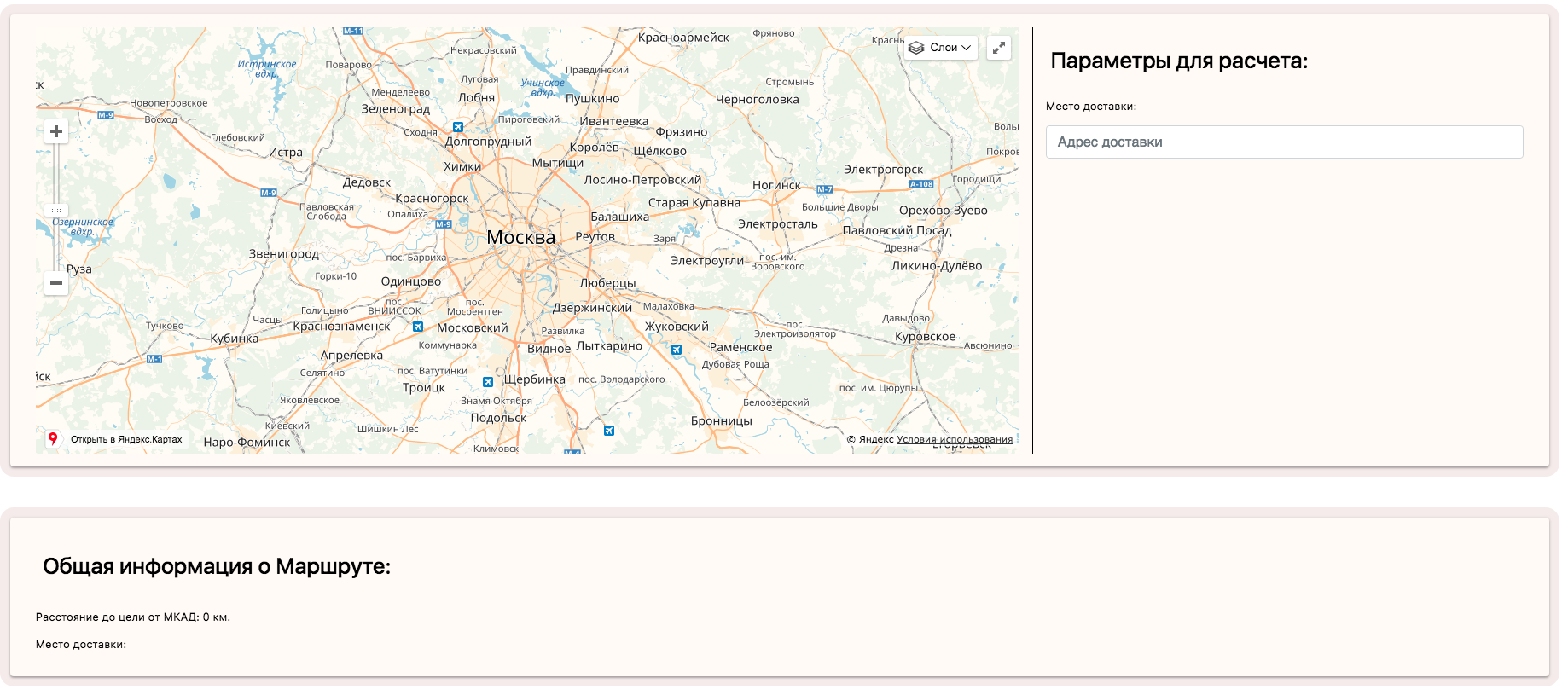


Рисунок Г.1 – Окно веб-приложения

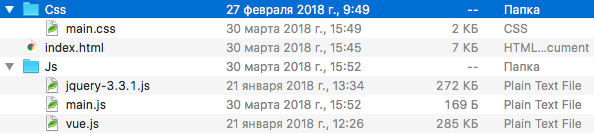


Рисунок Г.2 – Файлы

# 

# ПРИЛОЖЕНИЕ Д

**Доклад**

1. **Доклад**

### Введение

Веб-приложение ”DYM API” для менеджеров компании ООО”АртМеталл”.

### Основная часть

Основные задачи, решаемые веб-приложением ”DYM API”:

* Ускорение поиска кратчайшего маршрута от точки А(пользователя),до точки Б(ближайшая точка МКАД).

Основные достоинства:

1. Бесплатная;
2. Не ресурсозатратный;
3. Кроссплатформенность

Основные недостатки:

1. Любительская разработка.

### Заключение

Результаты моей работы заключается в изучении разработки веб-приложение ”DYM API”, предназначенных для менеджеров компании ООО”АртМеталл”, для избавления рутинной работы.

Разработанная система предназначена только для внутреннего использования внутри компании.

# ПРИЛОЖЕНИЕ Ж

**1 Презентация**



Рисунок Ж.2 – Cлайд 1

 Рисунок Ж.2 – Cлайд 2 

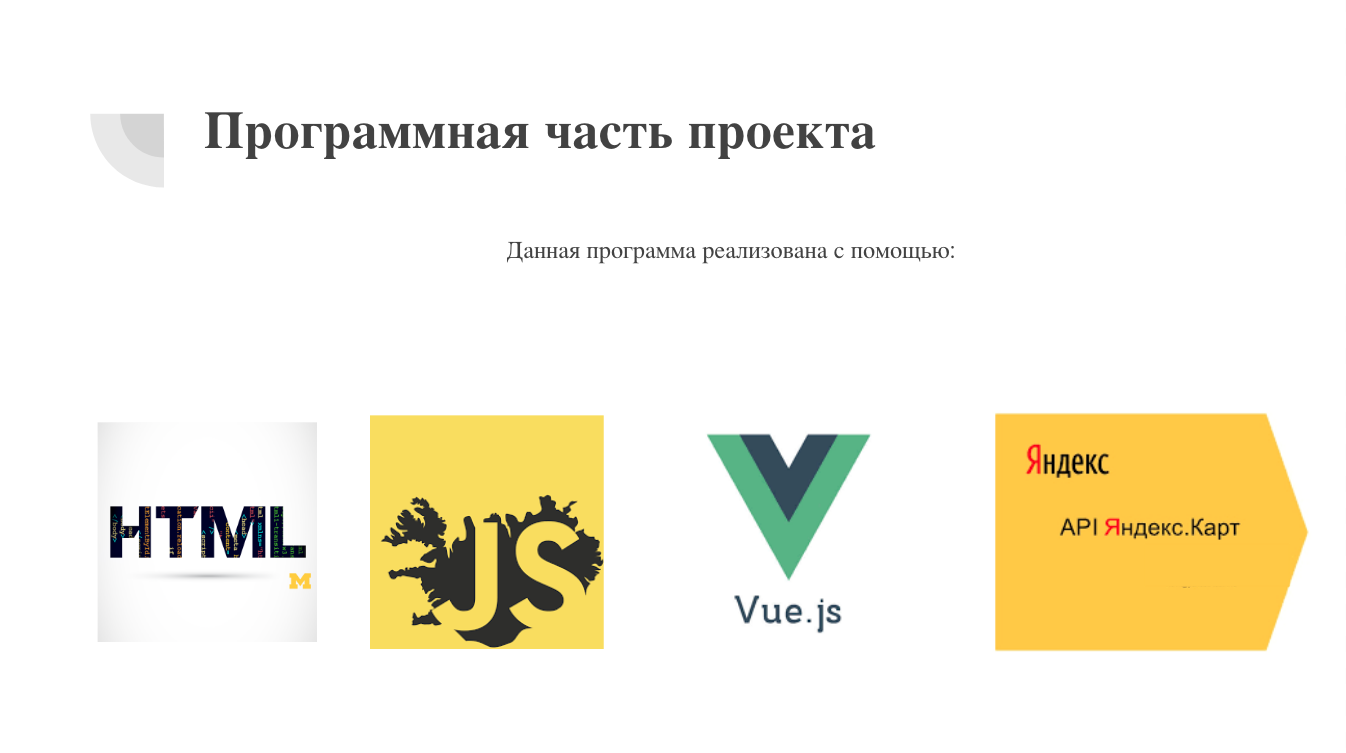
Рисунок Ж.2 – Cлайд 3 

Рисунок Ж.2 – Cлайд 4

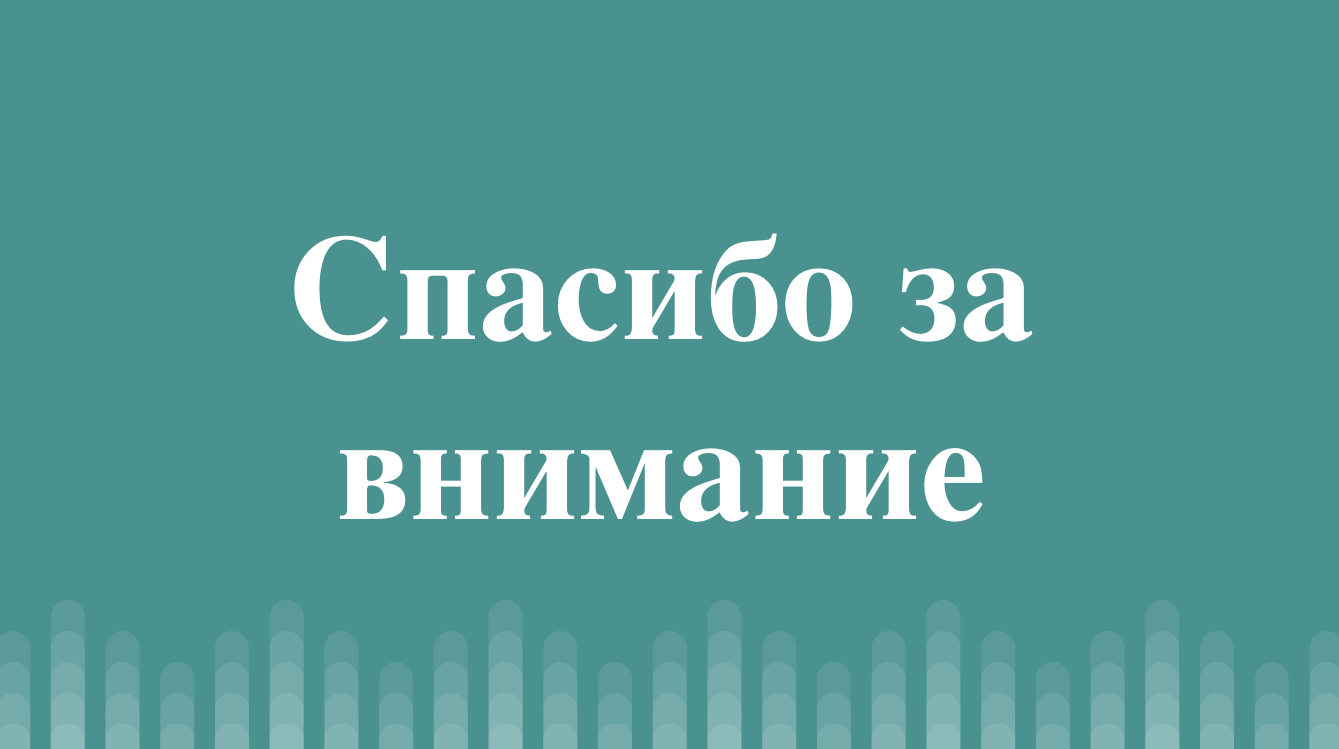


Рисунок Ж.2 – Cлайд 5