# Министерство образования и науки Российской Федерации

# Федеральное государственное бюджетное образовательное учреждение

# высшего профессионального образования

# «Ижевский государственный технический университет имени М. Т. Калашникова»

# Кафедра «Программное обеспечение»

Отчет по лабораторным работам

по дисциплине «Конструирование ПО»

тема «Личный помощник»

Выполнил

студент гр. Б08-191-2 К.Э.Солодянкина

Принял В.Г. Власов

Ижевск 2016

СОДЕРЖАНИЕ

1.ВВЕДЕНИЕ

1.1 Назначение системы

1.2 Область применения системы

1.3 Определения, акронимы, аббревиатуры

1.4 Ссылки

1.5 Обзор системы

2. ОБЩЕЕ ОПИСАНИЕ СИСТЕМЫ

2.1 Системный контекст

2.2 Режимы и состояния системы

2.3 Основные функциональные возможности системы

2.4 Основные условия системы

2.5 Основные ограничения системы

2.6 Характеристики пользователя

2.7 Допущения и зависимости

2.8 Оперативные сценарии

3. ФУНКЦИОНАЛЬНЫЕ ВОЗМОЖНОСТИ, УСЛОВИЯ И ОГРАНИЧЕНИЯ СИСТЕМЫ

3.1 Физические

3.1.1 Конструкция

3.1.2 Износостойкость

3.1.3 Адаптируемость

3.1.4 Условия окружающей среды

3.2 Рабочие характеристики системы

3.3 Безопасность системы

3.4 Информационный менеджмент

3.5 Работа системы

3.5.1 Эргономика системы

3.5.2 Ремонтопригодность системы

3.5.3 Надежность системы

3.6 Стратегия и регулирование

3.7 Устойчивость жизненного цикла системы

4. КЛАССЫ ПРОЕКТИРОВАНИЯ

5. КЛАССЫ КОДИРОВАНИЯ

6. ТЕСТОВЫЕ ДАННЫЕ

1. ВВЕДЕНИЕ

1.1 Назначение системы

Система предназначена для помощи пользователю с рядом повседневных нужд, таких как: планирование своего дня, выбор правильного питания и спортивных нагрузок, поиск нужных мест на карте.

1.2 Область применения системы

Данная система используется каждым пользователем для личных нужд и, в идеале, является универсальным помощником в течение всего дня. На данном этапе Личный помощник имеет не полный перечень возможностей, но в будущем это может быть исправлено.

Данная система будет применятся для решения следующих задач:

- поиск мест на карте с использованием данных Google Maps по заданному идентификатору (пример: ресторан, на выходе список: Планета Суши 13 и место на карте и т.п.);

- планирование расписания на день по типу Бизнес-планирование или Рациональное планирование;

- подбор правильной диеты;

- подбор спортивных нагрузок;

Преимущество системы в том, что она многофункциональна и не требует подключения к интернету.

1.3 Определения, акронимы, аббревиатуры

Пользователь - конечный пользователь системы, человек, использующий систему Личный помощник.

1.4 Ссылки

1.5 Обзор системы

Система делится на четыре главных модуля.

Первый модуль содержит в себе все рабочие подмодули системы (Планировщик, Правильное питание, Спортивные нагрузки и Поиск мест). Данные модулей отправляются пользовательскому интерфейсу для корректного отображения.

Второй модуль - это Интерактивный помощник. Содержит в себе интерактивные уроки по темам.

Третий модуль – это облачное хранилище данных. Осуществляет хранение данных на компьютере пользователя, обработку и предоставление информации.

Четвертый модуль – это пользовательский интерфейс. Отвечает за корректное отображение данных, управление графической частью, меню приложения, авторизацию.

2. ОБЩЕЕ ОПИСАНИЕ СИСТЕМЫ

2.1 Системный контекст

2.2 Режимы и состояния системы

Данная система может находиться в трех группах состояний:

1. Авторизация:

- авторизован;

- не авторизован.

2. Состояние облачного хранилища данных:

- синхронизировано (наличие интернета);

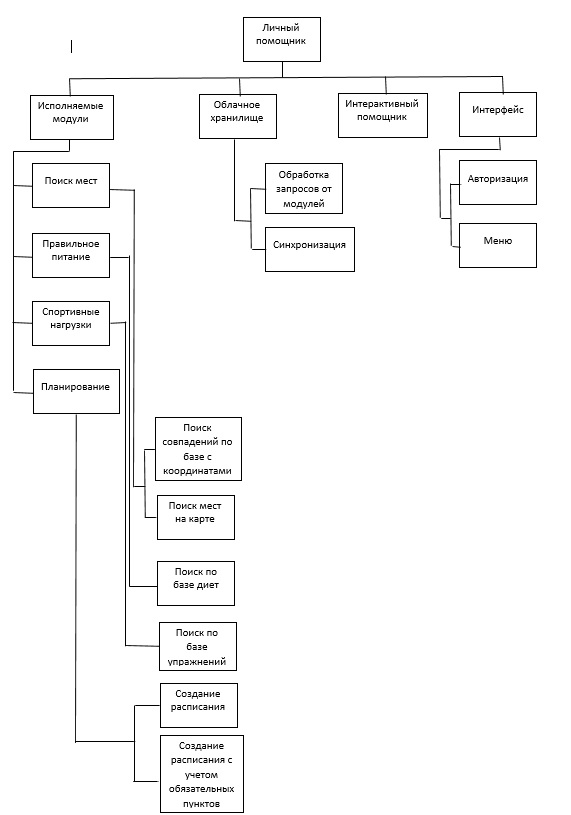
- не синхронизировано (без использования интернета).

3. Состояние программы:

- Ожидание;

- Исполнение рабочего модуля.

2.3 Основные функциональные возможности системы



2.4 Основные условия системы

2.5 Основные ограничения системы

Начальная версия системы будет работать с ограниченным функционалом возможностей.

2.6 Характеристики пользователя

Пользователь системы всегда только один. Данные для входа даются при покупке программы.

2.7 Допущения и зависимости

2.8 Оперативные сценарии

Пользователь выполняет авторизацию в системе. После этого попадает в главное меню программы. Из меню можно выбрать один из пунктов. Происходит его выполнение с вводом/выводом данных. Пользователь просматривает результат.

3. ФУНКЦИОНАЛЬНЫЕ ВОЗМОЖНОСТИ, УСЛОВИЯ И ОГРАНИЧЕНИЯ СИСТЕМЫ

3.1 Физические

3.1.1 Конструкция

3.1.2 Износостойкость

3.1.3 Адаптируемость

В случае полного заполнения облачного хранилища, его объем может быть увеличен.

3.1.4 Условия окружающей среды

3.2 Рабочие характеристики системы

3.3 Защита системы

Каждому пользователя разработчиком выдаются уникальные логин и пароль для входа в систему. Если данные были украдены, требуется связаться с разработчиком и заблокировать свое приложение.

3.4 Информационный менеджмент

3.5 Работа системы

3.5.1 Эргономика системы

3.5.2 Эксплуатационная технологичность

3.5.3 Надежность системы

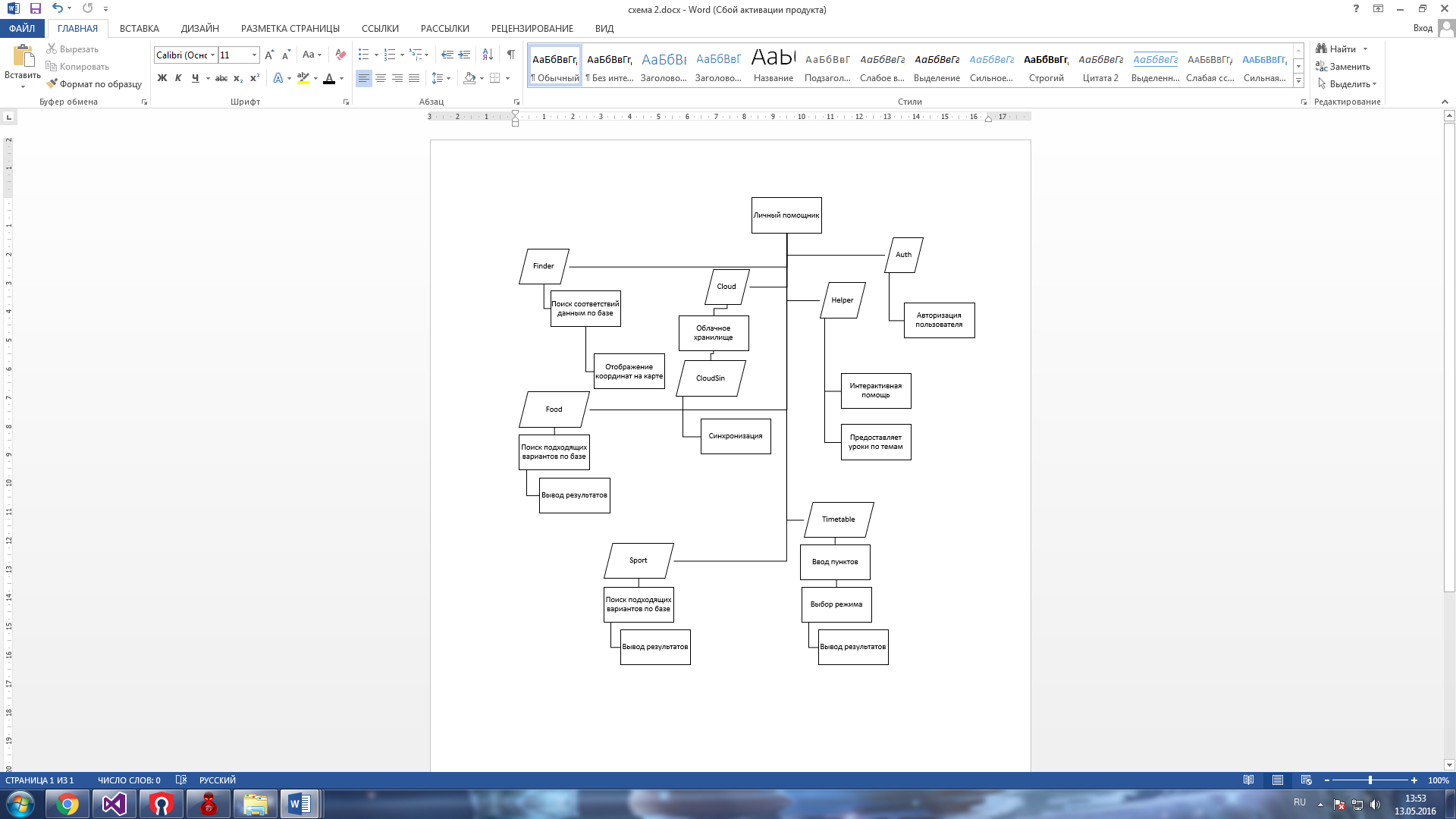
Система защищена от изменения исходных данных, т.к. облачное хранилище находится на компьютере самого пользователя и синхронизируется с хранилищем разработчика.

3.6 Стратегия и регулирование

3.7 Устойчивость жизненного цикла системы

Для устойчивости жизненного цикла системы используется обратная связь с пользователем.

1. КЛАССЫ ПРОЕКТИРОВАНИЯ



1. UML ДИАГРАММА КЛАССОВ КОДИРОВАНИЯ

Так как программа не имеет интерфейсов, то в ней нет и классов, их реализующих.

Список классов программы:

1. Класс Авторизации.

class Auth

{

string Login;

int Pass;

}

1. Класс Облачное хранилище.

class Cloud

{

void SinWithYanAPI();

}

1. Класс Поиск мест.

class Finder

{

string place;

void FindedPlaces(string place);

int x;

int y;

}

1. Класс Правильное питание.

class Food

{

int weight;

int height;

int wishWeight;

void GetFood(int weight, int height, int wishWeight);

}

1. Класс Помощник.

class Helper

{

string Lesson1;

string Lesson2;

string Lesson3;

string Lesson4;

string Lesson5;

}

1. Класс Спортивные нагрузки.

class Sport

{

int weight;

int height;

int wishWeight;

void GetActivity(int weight, int height, int wishWeight);

}

1. Класс Планировщик.

class Timetable

{

int mode;

string act;

void BusinessMode();

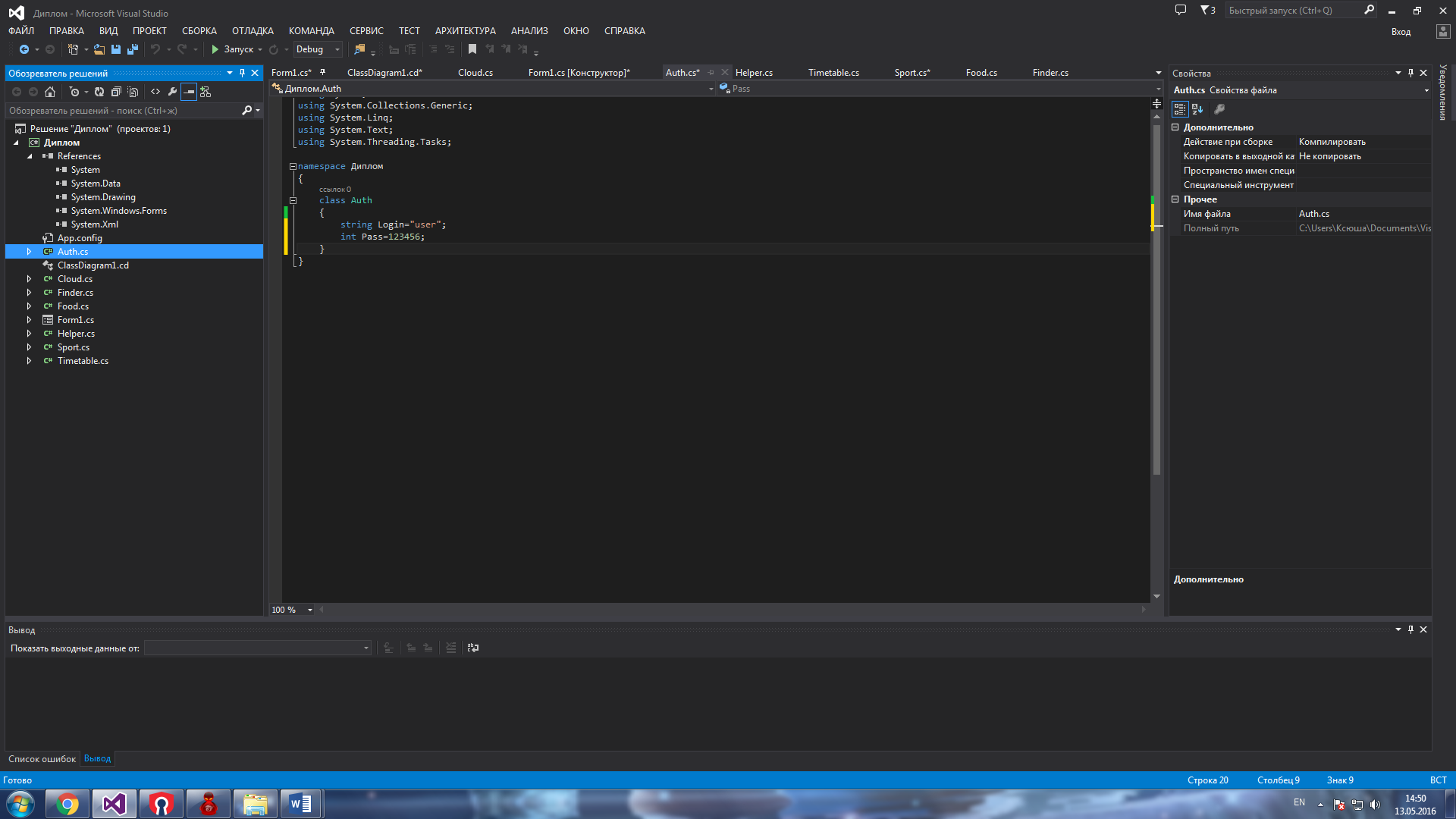
void HealthMode();

}

1. ТЕСТОВЫЕ ДАННЫЕ
2. Тестовые данные для модуля авторизации.

Ввод логина: user.

Ввод пароля: 123456.



Выходные данные: вход в программу.

1. Тестовые данные для модуля Поиск мест.

* Получение слова для поиска:

Входные данные:

Ресторан

Результат:

Планета Суши, 123, 674

Хоттей, 184, 603

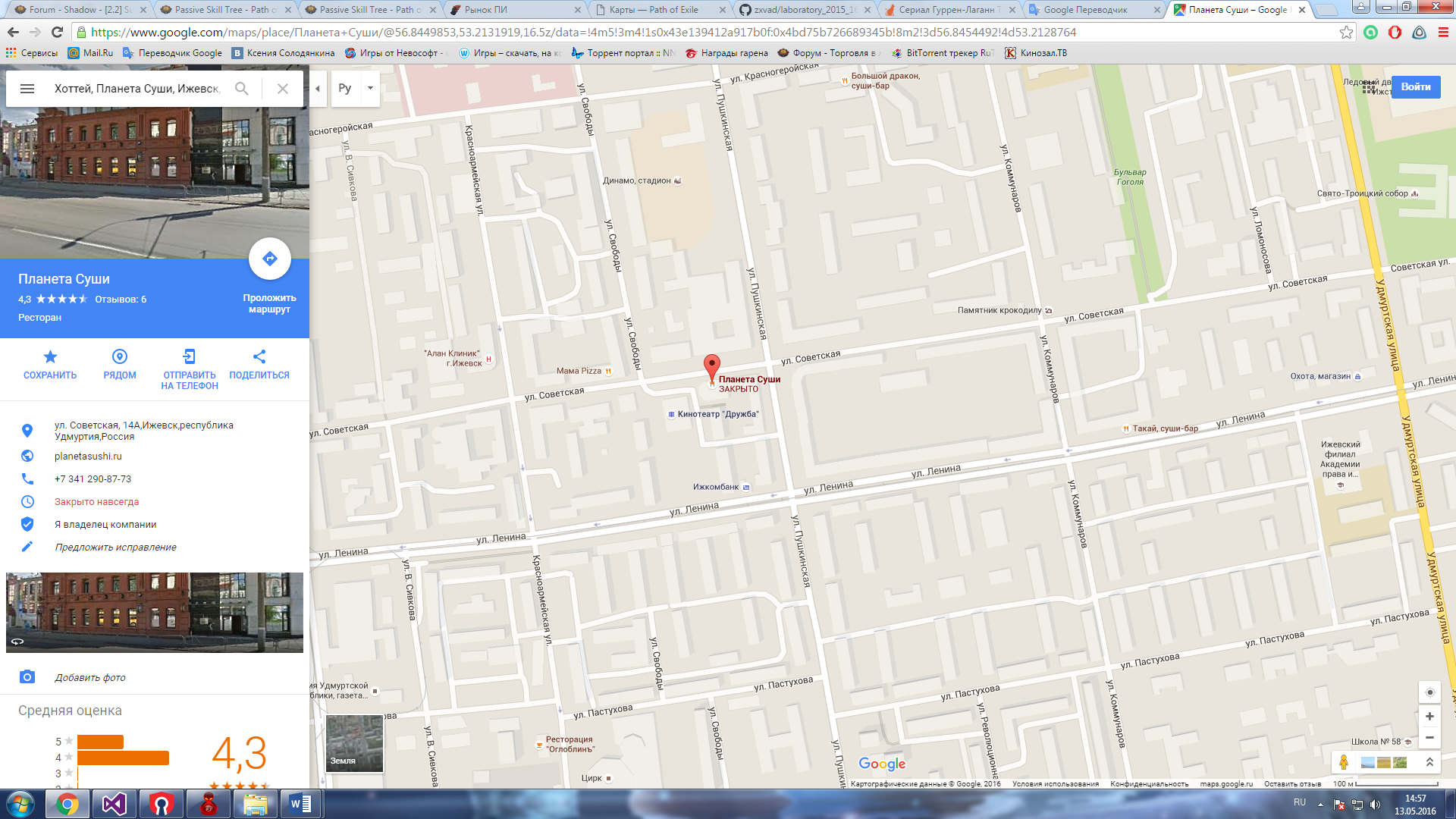
* Отображение:

Входные данные:

Планета Суши, 123, 674

Хоттей, 184, 603

Результат:



1. Тестовые данные для модуля Правильное питание.

Входные данные: рост 170, вес 60, желаемый вес 58

Результат: подходящая по параметрам диета.

1. Тестовые данные для модуля Спортивные нагрузки.

Входные данные: рост 170, вес 60, желаемый вес 58

Результат: подходящее расписание нагрузок.

1. Тестовые данные для модуля Планировщик.

Входные данные: режим 1, список задач

Результат: составленное расписание.