

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**

Заключительный этап

Профиль «Информационные технологии»

Индивидуальное тестирование по кейсам

1. Название команды

2. Какой кейс вы выполняли?

3. Какие языки программирования и библиотеки использовали?

Критерии оценивания:

1 балл - совпадении ответа участника с ответами членов команды и сданной документации по кейсу

0 баллов - не совпадение ответа участника с ответами членов команды и сданной документации по кейсу.

4. На какой платформе работает продукт?

Критерии оценивания:

1 балл - совпадении ответа участника с ответами членов команды и сданной документации по кейсу

0 баллов - не совпадение ответа участника с ответами членов команды и сданной документации по кейсу.

5. Как организовано хранение данных?

Критерии оценивания:

1 балл - совпадении ответа участника с ответами членов команды и сданной документации по кейсу

0 баллов - не совпадение ответа участника с ответами членов команды и сданной документации по кейсу.

6. Какой тип пользовательского интерфейса?

Критерии оценивания:

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**

Заключительный этап

Профиль «Информационные технологии»

Индивидуальное тестирование по кейсам

1 балл - совпадении ответа участника с ответами членов команды и сданной документации по кейсу

0 баллов - не совпадение ответа участника с ответами членов команды и сданной документации по кейсу.

7. Перечислите основные модули программного продукта.

Критерии оценивания:

1 балл - совпадении ответа участника с ответами членов команды и сданной документации по кейсу

0 баллов - не совпадение ответа участника с ответами членов команды и сданной документации по кейсу.

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
Командный кейс №1 «AR-приложение для изучения фразовых глаголов»

1. Условия

Современное образование все чаще прибегает к IT-технологиям. Например, широко применяются приложения для изучения иностранных языков. Известно, что учить языки легче, если делать это при помощи ассоциаций. Данный метод подходит для изучения слов как максимально приближенный к естественным процессам запоминания.

В то же время активно развиваются технологии дополненной реальности (augmented reality, далее — AR). Существует обширное множество мобильных программных продуктов, которые дополняют реальный мир изображениями. AR-технологии позволяют погрузить пользователя в новую, расширенную среду.

Участникам олимпиады предлагается разработать программное обеспечение, реализующее визуализацию фразовых глаголов английского языка с помощью технологии AR для устройств на операционных системах Android или iOS.

2. Техническое задание

Требуется разработать программное приложение, которое должно запускаться как минимум на одной из популярных операционных систем: Android, iOS.

Функциональное задание:

Обязательная функциональность программы:

1. Выбор фразового глагола из набора (минимум пять вариантов).
2. Наличие уникальной 3D-модели для каждого фразового глагола.
3. Отображение модели на плоской поверхности.

Дополнительная функциональность:

1. База данных для хранения 3D-моделей.
2. Анимация 3D-моделей.

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
Командный кейс №1 «AR-приложение для изучения фразовых глаголов»



Рисунок 1 — пример 3D-модели для фразового глагола look up.

3. Требования к документации

- Титульный лист (с указанием названия кейса и перечислением членов команды);
- Анализ технических требований;
- Обоснование используемых программных средств;
- Структурная и функциональная схемы программного продукта;
- Схема организации данных;
- Описание проведенных испытаний в соответствии с регламентом кейса (снимки экрана и/или запись экрана с работой);
- Программный код (ссылка на репозиторий);
- Подробное описание проекта в репозитории.

4. Регламент испытаний

1. Установка build-файла на мобильное устройство.
2. Проверка работоспособности возможностей приложения, описанного в техническом задании.
 - 2.1. Выбор фразового глагола из предложенного списка.

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
Командный кейс №1 «AR-приложение для изучения фразовых глаголов»

- 2.2. Проверка на соответствие модели фразовому глаголу.
- 2.3. Проверка на корректность отображения модели на ровной поверхности.

Проверку работоспособности приложения необходимо записать на видео.

5. Методические материалы

- [Что такое дополненная реальность \(AR\) | Microsoft Dynamics 365](#)
- Межплатформенная среда разработки [Unity](#)
- Официальная документация Unity [Unity Documentation](#)
- Официальная документация Unity по AR-разработке [Manual: Getting started with AR development in Unity](#)
- Официальные курсы по Unity [Unity Learn](#)
- Начало работы с AR Foundation [Getting started with AR Foundation | ARCore | Google Developers](#)
- Начало работы с Vuforia (SDK для дополненной реальности) [Getting Started with Vuforia Engine in Unity | VuforiaLibrary](#)
- Программное обеспечение для создание 3D моделей [Blender](#)
- Фреймворк для интеграции дополненной реальности на iOS [ARKit | Apple Developer Documentation](#)
- [Программирование на C# в Unity — бесплатные видеоуроки для детей и подростков](#)
- Среда разработки для работы с платформой Android [Download Android Studio & App Tools](#)
- Среда разработки для работы с платформой iOS [Xcode 14 Overview - Apple Developer](#)
- Документация по языку Java [Oracle Help Center](#)
- Документация по языку Kotlin [Kotlin Docs](#)
- Документация по языку Swift [Documentation](#)
- Документация по языку C

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
**Командный кейс №2 «Разработка приложения с использованием моделей
машинного обучения»**

1. Условия

Фотограмметрия — научно-техническая дисциплина, занимающаяся определением параметров формы, размеров и положения в пространстве объекта по его фотографическим изображениям. Данная дисциплина нашла широкое применение в различных отраслях и видах деятельности: создание карт и планов, охрана окружающей среды, проектирование и строительство зданий и сооружений, археологические раскопки, киноиндустрия и т.д. Такое широкое применение фотограмметрии обусловлено ее следующими достоинствами: высокая точность измерений, большая производительность, высокая степень автоматизации процесса измерений, безопасность выполнения работ (неконтактный метод).

Существует множество программ для работы с фотограмметрией, при этом каждая из них работает по общим для дисциплины алгоритмам и с некоторыми особенностями. Например, трудности в работе с глянцевыми и отражающими поверхностями. Так, в модели могут появляться провалы или выпуклости, что отличается от реальной формы объекта.

Процесс создания модели по снимкам состоит из большого числа шагов. Один из первых этапов — это восстановление позиций, откуда были сделаны снимки.

Участникам Олимпиады предлагается создать приложение с использованием моделей машинного обучения для обработки минимального набора изображений: определение пересечений между снимками (соответствие) и восстановление позиций съёмки для последующего построения 3D модели с помощью фотограмметрии. Тип объекта для построения модели выбирается участником самостоятельно, за исключением простых геометрических тел. В программе необходимо реализовать обработку минимум трёх изображений.

2. Техническое задание

Требуется разработать программное приложение, которое должно запускаться как минимум на одной из популярных операционных систем: Windows 10, дистрибутив Linux, MacOS.

Функциональное задание:

Функциональность программы:

1. выбор файлов (изображений) для обработки;

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
**Командный кейс №2 «Разработка приложения с использованием моделей
машинного обучения»**

2. определение пересечений между снимками (соответствие);
3. определение позиций съёмки;
4. вывод и демонстрация степени пересечения между снимками и соответствие файлов друг с другом;
5. вывод данных о позиции съёмки;
6. вывод состояния процессов программы (готово к работе, проводится анализ, ошибка, выполнено);
7. программа может иметь дополнительный функционал по желанию участника.
8. интерфейс программы должен быть простым и интуитивным, не требующим дополнительного обучения.

3. Требования к документации

- Титульный лист (с указанием названия кейса и перечислением членов команды);
- Обоснование выбора языка программирования и используемых программных средств и алгоритмов;
- Структурная и функциональная схемы программного продукта;
- Блок-схема работы основного алгоритма;
- Используемые для анализа изображения;
- Результаты обучения нейронной сети на исходных изображениях;
- Построенная 3D-модель по изображениям, взятым для анализа, с помощью любой доступной программы для фотограмметрии (снимки экрана и/или файл модели);
- Описание проведенных испытаний в соответствии с регламентом кейса (снимки экрана и/или запись экрана с работой);
- Программный код (ссылка на репозиторий).

4. Регламент испытаний

1. Установка из файла
2. Выбор изображений для обработки
3. Вывод степени пересечения между снимками и соответствие файлов друг с другом;
4. Вывод данных о позиции съёмки;
5. Масштабирование интерфейса;

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
**Командный кейс №2 «Разработка приложения с использованием моделей
машинного обучения»**

6. Проверка работоспособности функционала программы, описанного в документации.

5. Примерный перечень средств и инструментов для выполнения задания

<https://www.blender.org/>

<https://books.ifmo.ru/file/pdf/2455.pdf>

<http://www.nauteh-journal.ru/files/2ad8c604-f999-4224-a740-3c67dec0a417>

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
Командный кейс № 3 «Электронная система судейства»

1. Условия

Современные технологии проникают не только в быт, но и в спортивные состязания. Примером может служить VAR (Video assistant referee) в футболе. В лёгкой атлетике фотофиниш применяется еще с первой половины 20-го века, а начиная с 1992 года используются цифровые технологии для определения победителя в забеге. Автогонки не стали исключением и с появлением скоростных камер, ими оснащается каждая трасса, на которой проводятся высокоскоростные гонки.

В последние годы широкое распространение получил дрифт, как дисциплина в автоспорте. С учетом специфики дисциплины, фотофиниш не позволяет выявить победителя ни в квалификации, ни в парном заезде. Основными критериями оценки заезда являются:

- угол заноса;
- скорость движения автомобиля;
- траектория движения автомобиля.

Баллы за все три критерия выставляются судьями, но скорость и угол в определенных участках трассы имеется возможность измерить используя сенсоры, обработать с помощью программного обеспечения и предоставить судейской коллегии в виде инфографики.

Вам предлагается реализовать программно-аппаратный комплекс, способный дать максимально объективную оценку проездов пилотов по критерию «Угол».

2. Техническое задание

Для реализации комплекса вам необходимо разработать аппаратную и программную части.

Аппаратная часть должна состоять из сенсора (гироскоп и/или акселерометр), вычислительного устройства (Arduino, Raspberry Pi, ESP) и устройства, передающего по беспроводному каналу измеренную сенсором величину. В зависимости от выбранного вычислительного устройства, может варьироваться способ беспроводной передачи информации. Требования к конструктивным особенностям и питанию аппаратной части не предъявляются.

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ
Профиль «Информационные технологии»
Командный кейс № 3 «Электронная система судейства»**

Программная часть должна быть запущена на персональном компьютере или телефоне и иметь возможность соединения по беспроводному каналу с аппаратной частью.

Необходимо предусмотреть в программной части добавление соревнования, в рамках которого будет происходить последующие заезды. Информация о соревновании должна включать в себя:

- наименование соревнования;
- дату;
- наименование организатора;
- место проведения.

Перед началом заезда по инициативе программной части активируется аппаратная часть. При этом пользователем указываются порядковые номера пилота(ов) (число от 1 до 99) и один из типов заезда:

- qualifying (квалификация);
- top 32 (1/16)
- top 16 (1/8)
- top 8 (четвертьфинал)
- semifinal (полуфинал)
- battle for 3rd place (заезд за 3 место)
- final (финал)

Аппаратной частью осуществляется измерение необходимой телеметрии с сохранением её в памяти аппаратной части. Измерения должны производиться не менее 10 раз в секунду. Длительность заезда не должна превышать 60 секунд.

После окончания заезда по инициативе программной части необходимо осуществить передачу по беспроводному каналу измеренных данных с аппаратной части с последующей очисткой памяти аппаратной части и сохранением на программной части информации о заезде:

- тип заезда;
- номер(а) пилота(ов);
- время начала заезда;
- время окончания заезда;
- данные телеметрии.

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
Командный кейс № 3 «Электронная система судейства»

Непосредственно после передачи данных необходимо осуществить демонстрацию полученных данных программной частью в виде графика. График должен представлять зависимость угла от общего времени заезда. В случае квалификационного заезда отображается график телеметрии одного пилота, в случае парного заезда отображаются графики телеметрии обоих пилотов с наложением друг на друга.



Пример графика зависимости угла от времени заезда

После прохождения всех заездов необходимо предусмотреть возможность выбрать любой из заездов для просмотра данных телеметрии по нему в табличном виде и в виде графика.

Необходимо предусмотреть возможность архивации всех заездов соревнования.

3. Рекомендации к выполнению

- Рекомендуется предусмотреть модульность программной части, а именно серверную и клиентскую части.
- Необходимо предусмотреть сохранение данных, получаемых от аппаратной части в системе управления базами данных (СУБД). Выбор СУБД не регламентируется.

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
Командный кейс № 3 «Электронная система судейства»

- Разработку аппаратной и программной частей рекомендуется вести с помощью системы контроля версий git.
- Рекомендуется использовать unit-тестирование при разработке программной части.

4. Требования к документации

- Титульный лист (с указанием названия кейса и перечислением членов команды);
- Анализ технических требований;
- Обоснование выбора языка программирования и используемых программных средств;
- Структурная и функциональная схемы программного продукта;
- Блок-схема работы основного алгоритма;
- Схема базы данных;
- Описание проведенных испытаний в соответствии с регламентом кейса (снимки экрана и/или запись экрана с работой);
- Программный код (ссылка на репозиторий).

5. Регламент испытаний

Для демонстрации работы комплекса, в частности аппаратной части, допускается использование моделей автомобилей масштабе не более 1/8.

- Производится запуск комплекса
- Производится добавление соревнования
- Производится добавление и старт квалификационного заезда
- Производится окончание заезда и просмотр графика телеметрии
- Производится добавление и старт парного заезда
- Производится окончание заезда и просмотр графиков телеметрии
- Производится просмотр графика(ов) по уже прошедшему заезду
- Производится архивация соревнования и добавление нового

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ
Профиль «Информационные технологии»
Командный кейс № 3 «Электронная система судейства»**

6. Примерный перечень средств и инструментов для выполнения задания

- <https://www.python.org/>
- <https://sqlite.org/>
- <https://www.postgresql.org/>
- <https://flask.palletsprojects.com/en/2.0.x/>
- <https://vuejs.org/>
- <https://www.raspberrypi.com/>
- <https://www.espressif.com/en/products/socs/esp32>

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
**Командный кейс № 4 «Разработка Web-приложения по построению
маршрутов по интересным местам»**

1. Условия

Существует большое количество сервисов, связанных с картами и маршрутами, но нет сервиса, способного предложить в автоматическом или в полуавтоматическом режиме построение пешеходных маршрутов с посещением определённых точек. Подобное приложение было бы интересно для гостей города или экскурсоводов.

2. Техническое задание

Функциональные требования

Разработать web-приложение, обеспечивающую работу с точками интереса (POI) из рубрики «Достопримечательность» на территории ВДНХ и построение различных пешеходных маршрутов с начальной точки у павильона №1 и ограниченной длины.

Информационная система должна реализовывать следующую функциональность:

- Отображать точки интереса в рамках ВДНХ;
- Строить маршруты по точкам интереса с указанными особенностями:
 - С ограничением по времени;
 - С обязательным посещением определённой точки;
 - Статические маршруты;
- Визуализировать маршруты;
- Инструменты добавления статических маршрутов;
- Сохранение информации об ранее построенных маршрутах:
 - Маршрут;
 - Длина маршрута;
 - Затраченное время;
 - И так далее на усмотрение разработчика.

Дополнительные требования

Web-приложение должно состоять из WEB-сервиса и базы данных.

3. Рекомендации к выполнению

Допускается использовать локально запущенный сервис.

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
**Командный кейс № 4 «Разработка Web-приложения по построению
маршрутов по интересным местам»**

Рекомендуется использовать СУБД для хранения полученных данных, требуется также предоставить схему базы данных.

Рекомендуется строить автоматические маршруты с не более чем 5 пересечениями.

Проект рекомендуется вести с помощью системы контроля версий, например, [Git](#).

4. Требования к документации

- Титульный лист (с указанием названия кейса и перечислением членов команды);
- Анализ технических требований;
- Обоснование выбора языка программирования и используемых программных средств;
- Структурная и функциональная схемы программного продукта;
- Блок-схема работы основного алгоритма;
- Схема базы данных;
- Описание проведенных испытаний в соответствии с регламентом кейса (снимки экрана и/или запись экрана с работой);
- Программный код (ссылка на репозиторий).

5. Регламент испытаний

Демонстрация испытаний осуществляется с использованием программной модели комплекса.

- Производится запуск серверной части;
- Производится запуск веб-интерфейса;
- Производится демонстрация точек интереса;
- Демонстрация интерфейса добавление и редактирования статических маршрутов
- Демонстрация статических маршрутов, ранее записанных участниками;
- Демонстрация динамических маршрутов с ограничением по времени;

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
**Командный кейс № 4 «Разработка Web-приложения по построению
маршрутов по интересным местам»**

- Демонстрация динамических маршрутов с обязательным посещением места и по времени;
- Демонстрация истории ранее построенных маршрутов.

6. Примерный перечень средств и инструментов для выполнения задания

- <https://www.openstreetmap.org/#map=2/69.6/-74.9>
- <https://yandex.ru/dev/maps/mapsapi/?from=api-maps>
- <https://www.djangoproject.com/>

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
**Командный кейс № 5 «Разработка сервиса мониторинга доступности
сервисов российских высших учебных заведений»**

1. Условия

Требуется разработать веб-приложение, решающее задачу автоматизированного выявления недоступности ресурсов, сервисов и/или приложений высших учебных заведений (далее – ВУЗ). Полученное решение должно выполнять оперативное выявление в автоматизированном режиме: недоступности сервисов, сбоев на ресурсах и/или в приложениях ВУЗ (далее – сбой). Осуществлять мониторинг сети Интернет и поиск сообщений пользователей о сбоях. Проводить аккумулирование информации о выявленных сбоях по результатам обработки сообщений в сети Интернет. Необходимо организовать информирование по электронной почте.

2. Техническое задание

Функциональные требования

- адаптивная верстка сайта;
- наличие функции регистрации пользователей
- наличие возможности добавления ресурса для отслеживания его доступности
- наличие возможности сообщения о факте недоступности ресурса
- наличие функции оповещения о доступности ресурса посредством электронной почты
- наличие функции модерации заявок о доступности ресурсов
- наличие страницы мониторинга доступности популярных ресурсов
- наличие системы отзывов и рейтинга ресурса
- наличие функции оповещения о доступности ресурса посредством телеграмм бота
- наличие возможности автоматизированного развертывания веб-приложения на сервере
- наличие функции составления отчета о недоступности и сбоях ресурсах
- наличие автоматизированной системы отслеживания состояния ресурсов
- наличие функции автоматического определения массовой атаки на ресурсы

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
**Командный кейс № 5 «Разработка сервиса мониторинга доступности
сервисов российских высших учебных заведений»**

3. Требования к документации

Отчёт, прилагаемый к кейсу, должен включать в себя следующие разделы:

1. Титульный лист (с указанием названия кейса и перечислением членов команды)
2. Анализ технических требований
3. Обоснование выбора языка программирования и используемых программных средств
4. Структурная и функциональная и схемы разрабатываемых модулей
5. Блок-схема работы разрабатываемых модулей
6. Общая схема базы данных
7. Инструкция по развертыванию веб-приложения
8. Скриншоты и ссылка на видео, демонстрирующее работу приложения, продолжительностью не более 5 минут.
9. Программный код (ссылка на репозиторий в git)

4. Примерный перечень средств и инструментов для выполнения задания

- SIEM системы (сбор и анализ информации о ресурсах):
<https://cisoclub.ru/top-10-besplatnyh-open-source-siem-tools-v-2020-godu/>
- Веб-пауки (автоматизация сбора данных)
<https://www.octoparse.com/blog/10-best-open-source-web-scraper>
- Сервисы мониторинга доступности сайтов:
<https://habr.com/ru/company/hosting-cafe/blog/312958/>

5. Регламент испытаний:

1. Произвести запуск продукта, включая процесс развертывания
2. Продемонстрировать работу следующих функций:
 - 2.1 адаптивная верстка сайта
 - 2.2 регистрация пользователей
 - 2.3 добавление ресурса для отслеживания доступности
 - 2.4 вывод сообщения о факте недоступности ресурса
 - 2.5 оповещение о доступности ресурса посредством электронной почты

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
**Командный кейс № 5 «Разработка сервиса мониторинга доступности
сервисов российских высших учебных заведений»**

- 2.6 модерация заявок о доступности ресурсов
- 2.7 станция мониторинга популярных ресурсов
- 2.8 добавление отзыва и выставление рейтинга ресурса
- 2.9 оповещение о доступности ресурса посредством телеграмм бота
- 2.10 функция автоматизированного развертывания веб-приложения на сервере
- 2.11 составление отчета о недоступности и сбоях ресурса
- 2.12 автоматизированное отслеживание состояние доступности ресурсов
- 2.13 автоматическое обнаружение массовой атаки на ресурс
- 2.14 продемонстрировать пользовательский интерфейс

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
Командный кейс №6 «Чек-лист покупок»

1. Условие

Поход в магазин — еженедельная рутина. Перед походом в магазин многие покупатели составляют список нужных товаров.

В рамках данного кейса вы будете разрабатывать мобильное приложение для операционной системы Android — мобильного ассистента для похода в магазин.

Пользователь до похода в магазин сможет составлять список покупок, а находясь в магазине, с помощью камеры мобильного устройства фотографировать ценник; приложение будет распознавать информацию на нём и автоматически отмечать данный товар из списка покупок, как добавленный в корзину.

В рамках кейса вам дополнительно предстоит обучить модель машинного обучения, которая будет распознавать информацию о товаре с ценника магазина. Модель может работать локально на мобильном устройстве или на удалённом сервере. Для обучения модели вам будут предоставлены данные — датасет из изображений ценников из одной продуктовой сети.

2. Техническое задание

Необходимо разработать Android-приложение, которое будет распознавать наименование товара на ценнике, удалять этот товар из списка покупок (так как он куплен) и подсказывать пользователю, какие товары остались.

При запуске приложения пользователя встречает главный экран со списком покупок. Пользователь с этого экрана может:

1. Отсканировать товар
2. Добавить в список новый товар
3. Отредактировать существующий товар

Сканирование товара запускает встроенную камеру, затем фотография с камеры выводится на экран предварительного просмотра.

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ
Профиль «Информационные технологии»
Командный кейс №6 «Чек-лист покупок»**

Экран предварительного просмотра позволяет просмотреть полученную с камеры фотографию и выполнить распознавание, затем пользователь возвращается на главный экран (на нём распознанный товар будет отсутствовать).

Добавление и редактирование товара подразумевает ввод и изменение названия товара с возможностью сохранения, отмены и удаления товара (удаление доступно только в режиме редактирования).

Все действия должны происходить без подтверждающих диалоговых окон/экранов.

Для определения полного наименования товара и его цены по фотографии ценника потребуется обучить модель машинного обучения (модель ML). Выбор конкретных методов ML не регламентируется. Датасет можно скачать в учебной системе myitschool.ru/edu, в курсе «Московская предпрофессиональная олимпиада».

Ваше решение будет валидироваться на скрытых данных. Скрытые данные не содержат краткие наименования товаров, отличные от публичного датасета. Для обучения моделей можно использовать любые дополнительные открытые источники данных.

Базовое решение представлено в виде проекта Android Studio / IntelliJ IDEA: myitschool.ru/edu в курсе «Московская предпрофессиональная олимпиада» после подачи заявки. В нем продемонстрирована навигация по приложению.

При разработки программной системы рекомендуется использовать масштабируемые подходы.

Приложение должно корректно работать на устройстве с ОС Android 10.0 и выше. Все строковые ресурсы должны быть локализованы для русского языка.

Рекомендуется использовать СУБД (выбор СУБД не регламентируется).

Приложение должно содержать следующие экраны (активности/фрагменты/диалоговые окна):

- 1) Главный экран: Список покупок

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ
Профиль «Информационные технологии»
Командный кейс №6 «Чек-лист покупок»**

На этом экране у пользователя должен быть виден список RecyclerView с идентификатором `recycler`. Список товаров, составленный пользователем, не будет превышать 80 уникальных наименований товаров. В списке, согласно заданию, необходимо отображать все элементы, необходимые к покупке. Каждый элемент должен содержать элемент TextView с идентификатором `productName`. На этом экране доступно две кнопки: определения (распознавание) товара и добавления товара (идентификаторы `scan` и `add`).

2) Добавление продукта

Этот экран содержит View с идентификатором `editProductName` для ввода краткого названия продукта. При нажатии на View с идентификатором `save` (сохранить) происходит возврат на главный экран, в список добавляется новый продукт.

3) Редактирование продукта

Такой же экран, что и добавление продукта, но дополнительно есть возможность удаления (кнопка `remove`) ошибочно добавленного продукта. Нажатие на кнопку сохранения — изменяет текущий продукт в списке согласно изменённым данным.

4) Предварительный просмотр сделанного фото

Данный экран появляется после сделанного фото на встроенное приложение камеры (после сканирования) и содержит:

- Элемент изображения, который отображает сделанное пользователем фото (идентификатор `preview`).
- Действия отменить и распознать с соответствующими идентификаторами `cancel` и `recognize`.

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ
Профиль «Информационные технологии»
Командный кейс №6 «Чек-лист покупок»**

3. Требования к документации

При сдаче материалов по данному кейсу предоставляются:

1. Ссылка на удалённый репозиторий в системе контроля версий с полным документированным исходным кодом проекта (клиентского приложения и, при наличии, серверной части), историей изменений каждого члена команды, файлами ark и app-bundle в разделе Releases;
2. Ссылка на видео с краткой демонстрацией работы приложения длительностью не более 1 минуты;
3. Техническая документация, включающая титульный лист со списком команды, обоснование выбора архитектуры приложения, структур данных и СУБД, краткое описание используемых средств разработки, UML-диаграммы, распределение зон ответственности каждого участника команды, раскрытие оригинальных авторских подходов детектирования товаров, описание реализованных алгоритмов и скриншоты приложения. Допустимые форматы файла технической документации: .pdf, .docx, .tex

При построении клиент-серверной архитектуры приложения (размещения модели распознавания на удалённом компьютере) необходимо обеспечить полную работоспособность серверной части на время проведения валидации отборочного и заключительного этапов. Ожидание ответа от сервера не должно превышать 15 секунд.

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
Командный кейс №6 «Чек-лист покупок»

4. Регламент испытаний

Один или несколько членов команды должны зарегистрироваться в системе myitschool.ru/edu, подать заявку на курс «Московская предпрофессиональная олимпиада» и создать команду.

Загрузить zip-архив с полным проектом Android Studio / IntelliJ IDEA необходимо в систему myitschool.ru/edu

После загрузки файла будет произведена распаковка архива, сборка проекта, установка apk на устройстве и валидация на части публичного датасета. Предварительный рейтинг, учитывающий новое решение команды, будет обновлён в течение 4 часов с момента успешного запуска приложения.

Загружать архив в систему могут несколько членов команды. Будет учтена последняя попытка от команды.

После окончания срока сдачи решения кейса для отборочного этапа будет произведена валидация на тестовом датасете 1 в течение 120 ч. Ограничение по времени для тестового датасета 1: 8 минут. Оценивание на отборочном этапе учитывает качество модели при валидации на тестовом датасете 1, полноту технической документации, ведение git-репозитория, удобство использования приложения, понятность пользовательского интерфейса, наличие видеодемонстрации.

Минимум за 48 ч до начала заключительного этапа необходимо загрузить исходный код приложения в систему myitschool.ru/edu повторно. В течение 48 ч будет проводиться валидация на тестовом датасете 2. Ограничение по времени для тестового датасета 2: 8 минут.

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ
Профиль «Информационные технологии»
Командный кейс №6 «Чек-лист покупок»**

Метрика, используемая при валидации на тестовых датасетах 1 и 2

Для оценки качества модели используется метрика Mean Consequential Error (MCE):

$$MCE = \frac{1}{Q} \sum_{q=1}^Q rel(q),$$

где Q — число фотографий, $rel(q)$ — функция-индикатор, принимающая значение 1, если q -й продукт вычеркнут корректно, и 0 иначе.

На заключительном этапе участники в режиме реального времени демонстрируют работу приложения на двух изображениях ценников, предоставленных экспертной комиссией. Баллы заключительного этапа будут включать баллы за валидацию на тестовом датасете 2 и баллы за защиту проекта. Рекомендуется подготовить небольшую презентацию (не более 5 слайдов).

5. Примерный перечень средств и инструментов для выполнения задания

Открытый курс «Мобильная разработка на Java». Samsung Innovation Campus. URL: <https://myitschool.ru/edu/>

Материалы Samsung Android Bootcamp 2020. URL: <https://youtube.com/playlist?list=PLa2T1zmZ6w5IVEYXsemrWogFShsswVI2A>

Материалы Samsung Android Bootcamp 2021. URL: <https://youtube.com/playlist?list=PLa2T1zmZ6w5KzKoh9M91vk1LBqpc-WtoS>

Материалы Samsung ML Bootcamp 2022. URL: <https://www.youtube.com/playlist?list=PLa2T1zmZ6w5JPxnz1AoGAzlk8-OuY8EVn>

AI Лекториум. IT Академия. Samsung Innovation Campus. URL: <https://youtube.com/playlist?list=PLJEYfuHbcEIB-DdeoWaQ6Bzt0903kbnWK>

Курс Нейронные сети и компьютерное зрение. URL: <https://stepik.org/course/50352/>

Курс Нейронные сети и обработка текста. URL: <https://stepik.org/course/54098/>

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ
Профиль «Информационные технологии»
Командный кейс №6 «Чек-лист покупок»**

Mohri M., Rostamizadeh A., Talwalkar A. *Foundations of machine learning*. – MIT press, 2018.

Марсикано К, Стюарт К, Филлипс Б. *Android. Программирование для профессионалов*. СПб: Питер. 2017.

Murphy, M.L. *The busy coder's guide to Android development*. United States: CommonsWare, 2008.

Murphy, M.L. *Elements of Android Room*. United States: CommonsWare, 2021.

Документация для разработчиков Android приложений. URL:

<https://developer.android.com/>

One UI Design Guidelines. URL: <https://developer.samsung.com/one-ui>

Kotlin для Data Science. URL: <https://kotlinlang.ru/docs/data-science-overview.html>

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). ImageNet Classification with Deep Convolutional Neural Networks. *Communications of the ACM*, 60(6), 84–90.

<https://doi.org/10.1145/3065386>

Гудфеллоу Я., Бенджио И., Курвилль А. Глубокое обучение - Издательство "ДМК Пресс" - 2018 - ISBN: 978-5-97060-618-6 - Текст электронный // ЭБС ЛАНЬ

Курс “Feature Engineering” Kaggle. URL: <https://www.kaggle.com/learn/feature-engineering>

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
Командный кейс №7 «Компьютерное зрение в беспилотном транспорте»

1. Условия

Классификация автоматизации автомобилей разработана [Сообществом автомобильных инженеров \(SAE\)](#) и содержит 6 уровней:

Уровень 0. Никакой автоматизации, водитель выполняет всю работу.

Уровень 1, «hands on», «помощь водителю». Водитель и система вместе управляют автомобилем. Пример: водитель рулит, а система регулирует мощность двигателя, сохраняя заданную скорость (круиз-контроль) или регулирует мощность двигателя и управляет тормозом, сохраняя заданную скорость и при необходимости снижая её, чтобы соблюдать дистанцию (адаптивный круиз-контроль). Другим примером является автоматическая парковка, когда скорость определяется водителем, а руление автоматическое.

Уровень 2, «hands off», «частичная автоматизация». Система полностью управляет автомобилем, осуществляя ускорение, торможение и поворот рулевых колёс. Водитель следит за ситуацией и готов вмешаться в любой момент, если система не может правильно отреагировать. Несмотря на название «hands off», такие системы часто требуют от водителя держать руки на руле, как подтверждение готовности вмешаться.

Уровень 3, «eyes off», «условная автоматизация». От водителя не требуется немедленной реакции. Он может, например, писать сообщения или смотреть фильм. Система сама реагирует на ситуации, требующие немедленных действий, таких как экстренное торможение. От водителя требуется готовность вмешаться в течение какого-то ограниченного времени, определённого производителем.

Уровень 4, «mind off», «широкая автоматизация». Отличается от уровня 3 тем, что от водителя не требуется постоянного внимания. Например, он может лечь спать или покинуть место водителя. Полностью автоматическое вождение осуществляется лишь в некоторых пространственных областях (геозонах) или в некоторых ситуациях, например, в пробках. Вне таких мест или ситуаций система способна прекратить вождение и припарковать машину, если водитель не взял управление на себя.

Уровень 5, «steering wheel optional», «полная автоматизация». Никакого человеческого вмешательства не требуется.

Несколько лет назад автопромышленность смело предсказывала выход полностью беспилотных автомобилей 5 уровня в 2020 или 2021 году. Сейчас производители

МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ ОЛИМПИАДА ШКОЛЬНИКОВ

Профиль «Информационные технологии»

Командный кейс №7 «Компьютерное зрение в беспилотном транспорте»

беспилотных автомобилей признают, что даже 4-ый уровень появится только в определённых областях с идеальными погодными условиями. 5-ый уровень во всех автомобилях, обеспечивающий езду куда угодно и когда угодно, теперь признан более сложным, чем казалось изначально. Производители беспилотных автомобилей теперь не дают прогнозов о том, когда появится 5-ый уровень автономности.

Сложности при разработке вызывают разнообразные погодные условия. Снег, дождь, туман блокируют видимость, для большинства сенсоров беспилотного автомобиля это критично. Качество и видимость дорожной разметки вызывают не меньшие затруднения чем разнообразные погодные условия. Определение степени опасности, которую представляют объекты на пути следования, тоже весьма непростая задача. Например, научить беспилотник отличать безобидный листопад от «опасных» объектов: стаи небольших птиц, обломков ограждения трассы и т.п.

Не стоит отказываться от мечты о полностью автономном беспилотном транспорте из-за перечисленных сложностей. Сложности делают стоящие перед исследователями и инженерами задачи более интересными. Нерешённых задач в области беспилотного транспорта такое количество, что часть из них будет решать следующее поколение специалистов.

Участникам Олимпиады предлагается создать программу локального позиционирования беспилотного автомобиля для участков дороги с пешеходными переходами. По изображению с видеокамеры программа должна определять:

- положение центральной линии полосы движения



- наличие на пешеходном переходе объектов

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
Командный кейс №7 «Компьютерное зрение в беспилотном транспорте»

- дистанцию до пешеходного перехода.



2. Техническое задание

Разработать приложение, извлекающее данные для планирования траектории беспилотного автомобиля, из изображения участка дороги с пешеходным переходом. Приложение должно включать в себя:

- алгоритм компьютерного зрения, детектирующий на изображении центральную линию полосы движения;
- алгоритм компьютерного зрения, рассчитывающий дистанцию до пешеходного перехода;
- алгоритм компьютерного зрения, определяющий наличие на перекрёстке объектов, закрывающих разметку: пешеходов, других автомобилей, любых предметов;
- интерфейс пользователя, позволяющий работать с входными, выходными данными и визуализацией результатов анализа изображения.

Для анализа изображений с камеры необходимо знать параметры её оптической системы. Для получения параметров камеры проводят её калибровку. В приложении должна быть возможность рассчитать параметры камеры по нескольким фотографиям «шахматной доски» (см. [калибровка камеры](#)).

Результат работы приложения — численное значение расстояния до пешеходного перехода, графическое отображение центральной линии полосы движения и индикатор наличия объектов, перекрывающих разметку пешеходного перехода.

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
Командный кейс №7 «Компьютерное зрение в беспилотном транспорте»

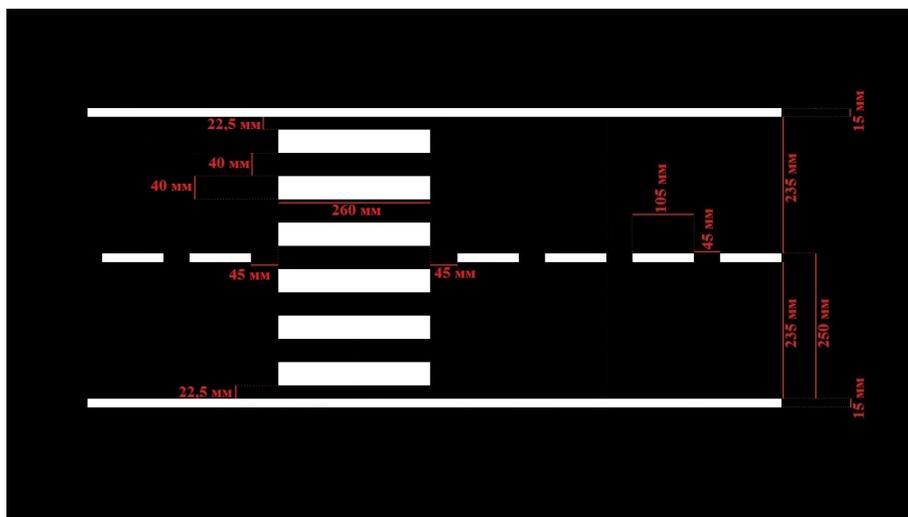
Входные данные:

- изображения участка дороги с пешеходным переходом
- набор данных для выявления параметров видеокамеры

Скачайте [архив с подготовленными входными данными](#).

При разработке приложения следует принять следующие допущения:

- На всех изображениях с камеры двусторонняя дорога с двумя полосами движения;
- Движение по дороге правостороннее;
- Изображение получено с камеры учебной модели беспилотного автомобиля в масштабе 1:18;
- Дорожная разметка выполнена по единым правилам;



- Дистанция от камеры до пешеходного перекрёстка не превышает 80 см;
- Центральная линия полосы движения — это линия равноудалённая от сплошной и пунктирной линий разметки, ограничивающих полосу движения;
- Дистанция до пешеходного перехода — это расстояние от точки проекции центра объектива на дорожное полотно до прямой, образованной ближайшими короткими сторонами прямоугольников пешеходного перехода.
- Размер квадрата на «шахматной доске» для калибровки камеры 25 мм.

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
Командный кейс №7 «Компьютерное зрение в беспилотном транспорте»



Программа должна быть ориентирована на рядового пользователя ПК, т.е. быть простой и понятной в освоении, желательно с графическим интерфейсом.

Наличие дополнительных, существенно отличающихся друг от друга методов расчёта одних и тех же данных, является плюсом для приложения.

Пользователь должен иметь возможность изменять параметры камеры, применяемые при анализе входных данных.

Формат выходных данных, с которыми будет работать приложение, необходимо разработать! Формат должен быть разработан с точки зрения удобства хранения и работы с данными. Возможность экспортировать или распечатать входные и выходные данные, будет плюсом для приложения.

Проект рекомендуется вести с помощью системы контроля версий git. При разработке рекомендуется использовать модульное тестирование и использовать комментарии в коде.

3. Требования к документации

Разработка программы должна сопровождаться ведением документации. Документация необходима для заочного этапа оценки, именно по ней будет проводиться оценка приложения.

Требуемая документация:

- Титульный лист (с указанием названия кейса и перечислением членов команды);
- Анализ технических требований;
- Структурная и функциональная схемы программного продукта;

МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ ОЛИМПИАДА ШКОЛЬНИКОВ

Профиль «Информационные технологии»

Командный кейс №7 «Компьютерное зрение в беспилотном транспорте»

- Блок-схема работы основного алгоритма;
- Схема базы данных, если используется;
- Описание проведенных испытаний в соответствии с регламентом кейса (снимки экрана и/или запись экрана с работой);
- Программный код (ссылка на репозиторий).

Формат описания и используемые виды схем остаются на усмотрение разработчиков. Выбор должен способствовать понятности и наглядности документации.

Представьте, что другой разработчик должен модернизировать ваше приложение. Ваша документация должна помочь ему быстро понять принципы работы программы и её структуру.

4. Регламент испытаний

1. Установка и запуск исполняемого файла приложения;
2. Ввод заранее подготовленных входных данных;
3. Демонстрация визуализации результатов расчётов приложения;
4. Демонстрация возможностей работы с входными и выходными данными;
5. Демонстрация прочих возможностей пользовательского интерфейса;
6. Корректное завершение работы приложения на любых этапах работы.

5. Примерный перечень средств и инструментов для выполнения задания

- язык программирования Python — <https://www.python.org>;
- библиотека OpenCV — <https://opencv.org> и другие библиотеки для работы с компьютерным зрением;
- библиотека машинного обучения TensorFlow — <https://www.tensorflow.org> и другие библиотеки машинного обучения;
- сервис для удалённого исполнения кода, ориентированный на специалистов по обработке данных и исследователей в области искусственного интеллекта — <https://colab.research.google.com>;
- детектор YOLO — <https://github.com/AlexeyAB/darknet> и другие детекторы;
- фреймворк QT для разработки графических интерфейсов — <https://github.com/pyqt/python-qt5> и другие фреймворки.

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
Командный кейс № 8
«Автоматизированная система удаленного управления теплицей»

1. Условия

Юные фермеры решили автоматизировать свою теплицу, установив в нее автоматизированную систему контроля и управления климатом.

Автоматизированная система состоит из:

1. Четыре равноудалённых друг от друга датчика температуры и влажности воздуха, что позволяет корректно определять среднюю температуру воздуха и влажности в теплице
2. Шесть датчиков влажности почвы, каждый из которых установлен в свою собственную тепличную бороздку
3. Автоматизированный привод удаленного открытия форточек теплицы для обеспечения проветривания
4. Шесть систем автоматизированного полива грядки, каждая из которых установлена в собственную тепличную бороздку по аналогии с датчиками влажности почвы
5. Единая система увлажнения воздуха в теплице.

Автоматизированная система представляет из себя комплекс специальных измерительных и управляющих приборов, взаимодействие с которыми осуществляется по средствам API:

Получение измерений с датчиков температуры и влажности осуществляется запросом GET на адрес https://dt.miet.ru/ppo_it/api/temp_hum/<number>, где <number> – порядковый номер устройства от 1 до 4.

Пример корректного ответа датчика температуры и влажности воздуха в формате JSON:

```
{  
  "id": device_id,  
  "temperature": temp_value,  
  "humidity": hum_value  
}
```

Где device_id – номер устройства, temp_value – температура воздуха в момент запроса, hum_value – влажность воздуха в момент запроса

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ
Профиль «Информационные технологии»
Командный кейс № 8
«Автоматизированная система удаленного управления теплицей»**

Получение измерений с датчиков влажности почвы осуществляется запросом GET на адрес https://dt.miet.ru/ppo_it/api/hum/<number>, где <number> – порядковый номер устройства от 1 до 6.

Пример корректного ответа датчика влажности почвы в формате JSON:

```
{  
  "id": device_id,  
  "humidity": hum_value  
}
```

Где device_id – номер устройства, hum_value – влажность воздуха в момент запроса

Управление автоматизированным приводом открытия форточек осуществляется запросом PATCH на адрес https://dt.miet.ru/ppo_it/api/fork_drive/ со следующими параметрами:

```
{  
  "state": state_value  
}
```

Где state_value принимает значение 0 – закрыть форточку, 1 – открыть форточку

Управление автоматизированным поливом грядки осуществляется запросом PATCH на адрес https://dt.miet.ru/ppo_it/api/watering со следующими параметрами:

```
{  
  "id": device_id  
  "state": state_value  
}
```

Где device_id – номер устройства от 1 до 6, state_value принимает значение 0 – закрыть полив, 1 – открыть полив

Управление единой системой увлажнения осуществляется запросом PATCH на адрес https://dt.miet.ru/ppo_it/api/total_hum со следующими параметрами:

```
{  
  "state": state_value  
}
```

Где state_value принимает значение 0 – закончить увлажнение, 1 – начать увлажнение

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**

Профиль «Информационные технологии»

Командный кейс № 8

«Автоматизированная система удаленного управления теплицей»

2. Техническое задание

Имеется программный интерфейс, с помощью которого можно взаимодействовать с аппаратной частью автоматизированной системы управления теплицей. Для взаимодействия необходимо выполнить GET и PATCH запросы к методам API, описанным выше. В заголовках запроса необходимо указать токен авторизации, выданный организаторами: {"X-Auth-Token": "<token>"}

Необходимо разработать программный продукт, который позволит:

1. Отображать динамику и последние данные с каждого датчика температуры и влажности, как в табличном виде, так и в виде графиков. Данные формировать в рамках текущей рабочей сессии. Период демонстрации определяется участником
2. Отображать динамику и последние данные средней температуры и средней влажности в теплице, как в табличном виде, так и в виде графиков
3. Открыть или закрыть форточки в теплице нажатием на специальные кнопки, а также просматривать текущий статус
4. Отображать динамику и последние данные с каждого датчика влажности почвы, как в табличном виде, так и в виде графиков
5. Открыть или закрыть полив конкретной бороздки в теплице нажатием на специальные кнопки, а также просматривать текущий статус
6. Открыть или закрыть общее увлажнение теплицы нажатием на специальные кнопки, а также просматривать текущий статус

Предусмотреть также выполнение следующих условий:

1. Если средняя температура в теплице больше, чем **Т градусов**, кнопка открытия форточек активна, в противном случае нажатие невозможно
2. Если средняя влажность в теплице меньше, чем **Н %**, кнопка открытия общей системы увлажнения активна, в противном случае нажатие невозможно

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**

Профиль «Информационные технологии»

Командный кейс № 8

«Автоматизированная система удаленного управления теплицей»

3. Если средняя влажность почвы в конкретной бороздке теплицы меньше, чем **Нв** %, кнопка открытия системы полива активна, в противном случае нажатие невозможно

Предусмотреть режим настройки системы и указания параметров: **Т, Н, Нв**.

Также предусмотреть переход в режим экстренного управления, при котором возможно нажатие всех описанных выше кнопок даже если условия не соблюдены.

3. Рекомендации к выполнению

- Программный продукт должен работать или на мобильных устройствах, или иметь веб-интерфейс.
- Полезной будет возможность внесения новых данных от датчиков вручную в интерфейсе пользователя.
- Необходимо предусмотреть сохранение данных, получаемых от датчиков в системе управления базами данных (СУБД). Выбор СУБД не регламентируется.
- Разработку рекомендуется вести с помощью системы контроля версий git.
- Рекомендуется использовать unit-тестирование при разработке продукта.

4. Требования к документации

- Титульный лист (с указанием названия кейса и перечислением членов команды);
- Анализ технических требований;
- Обоснование выбора языка программирования и используемых программных средств;
- Структурная и функциональная схемы программного продукта;
- Блок-схема работы основного алгоритма;
- Схема базы данных;
- Описание проведенных испытаний в соответствии с регламентом кейса (снимки экрана и/или запись экрана с работой);
- Программный код (ссылка на репозиторий).

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**

Профиль «Информационные технологии»

Командный кейс № 8

«Автоматизированная система удаленного управления теплицей»

5. Регламент испытаний

- Производится запуск программного продукта
- Отображение показателей всех датчиков в виде графиков и таблиц
- Отображение текущего состояния все устройств управления теплицей
- Настройка изменяемых параметров для работы системы
- Демонстрация работоспособности всех кнопок с учётом настроенных параметров
- Демонстрация режима экстренного управления
- Демонстрация unit-тестов

6. Примерный перечень средств и инструментов для выполнения задания

- <https://www.python.org/>
- <https://nodejs.org/>
- <https://sqlite.org/>
- <https://www.postgresql.org/>
- <https://pypi.org/project/requests/>
- <https://flask.palletsprojects.com/en/2.0.x/>
- <https://reactjs.org/>
- <https://vuejs.org/>
- <https://matplotlib.org/>

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
Командный кейс №1 «Классификатор по тематикам интернет ресурсов»

1. Условия

В наше время количество информации, получаемой человеком, растёт невероятно быстрыми темпами. Чтобы оптимизировать наше время и снизить нагрузку на мозг, людям проще классифицировать определенные данные, чтобы была возможность, например, не вчитываясь в новость, понять, с чем она связана.

Большинство объектов в информационном поле сейчас классифицировано, есть специальные датасеты в свободном доступе, начиная от квартир до категоризации эмоций людей на фотографиях. Такие данные при использовании машинного обучения дают огромный пул инструментов, позволяющих создавать невероятные приложения для оптимизации некоторых процессов жизни человека.

На данный момент классификаторы по тематикам ресурсов имеются, но у них очень много недостатков, например, низкая точность или же скудная функциональность.

Участникам Олимпиады предлагается разработать программу для классификации интернет-ресурсов по тематикам (например: образовательный, развлекательный, финансы и т. д.). Участникам нужно внести не менее 6 классов. Для обучения и тестирования программы можно использовать датасеты с открытым доступом. В программе необходимо реализовать классификацию ресурсов с применением машинного обучения и хранением обработанных и классифицированных данных в БД.

2. Техническое задание

Требуется разработать программное приложение, которое должно запускаться как минимум на одной из популярных операционных систем: Windows 10, дистрибутив Linux, MacOS.

Функциональность программы:

1. Внесение ссылки на сайт с целью его классификации;
2. Реализация базы данных для хранения классифицированных и обработанных данных интернет-ресурсов;
3. Возможность выгрузки данных сайта в формате CSV;
4. Возможность подгрузки файла с ссылками и данными для классификации и выгрузка файла с уже классифицированными данными офлайн.

МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ ОЛИМПИАДА ШКОЛЬНИКОВ

Профиль «Информационные технологии»

Командный кейс №1 «Классификатор по тематикам интернет ресурсов»

5. Возможность увидеть метрики при прогоне тестовой выборки, например: accuracy, precision, recall.

6. Программа может иметь дополнительную функциональность (например, реализован выбор алгоритма, по которому происходит классификация: линейная регрессия, логистическая регрессия и т.д.) по желанию участника.

7. Интерфейс программы должен быть простым и интуитивным, не требующим дополнительного обучения.

3. Требования к документации

- Титульный лист (с указанием названия кейса и перечислением членов команды);
- Анализ технических требований;
- Структурная и функциональная схемы программного продукта;
- Блок-схема работы основного алгоритма;
- Описание проведённых испытаний в соответствии с регламентом кейса (снимки экрана и/или запись экрана с работой);
- Программный код (ссылка на репозиторий).

4. Регламент испытаний

1. Установка из файла;
2. Выбор и ввод ссылки для дальнейшей классификации;
3. Классификация, просмотр результатов;
4. Выгрузка результатов в таблицу;
5. Проверка работоспособности функций программы, описанного в документации.
6. Импорт ссылок из CSV для дальнейшей классификации.

5. Примерный перечень средств и инструментов для выполнения задания

<https://www.python.org>

<https://scikit-learn.org/stable/>

<https://www.tensorflow.org>

<https://aws.amazon.com/ru/machine-learning/>