

**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 «Надёжность вычислительной техники»

1. Условия

Современный мир зависит от вычислительной техники, в каждой семье есть подобные устройства: смартфоны, микроволновка, духовка, холодильник и т.д. Все они стали незаменимы для человека в современном обществе, но вне зависимости от условий эксплуатации, работа их не вечна.

Все современные устройства — это комплекс микросхем. Чаще всего ломается не само устройство, а его конкретная часть: одна или несколько микросхем. В промышленном масштабе заменять каждый раз целиком сложное устройство (например: промышленный компьютер) экономически нецелесообразно, поэтому актуален вопрос выявления вышедших из строя компонентов для их последующей замены.

На данный момент не существует унифицированных программ для расчёта срока службы электронных компонентов.

Участникам Олимпиады предлагается создать программу для расчёта срока службы электронного компонента устройства. Тип вычислительной техники выбирается участником самостоятельно. В программе необходимо реализовать подсчёт срока службы минимум трёх типов электронных компонентов (например: резисторы, транзисторы, кабели и т.д.).

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:

Функционал программы:

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ
Профиль «Информационные технологии»
Командный кейс № 1 «Надёжность вычислительной техники»**

- выбор типа электронных компонентов;
- подключение базы данных;
- расчёт срока службы устройства;
- вывод графиков зависимости срока годности от конкретного критерия (например, вибростойкость);
 - программа может иметь дополнительный функционал по желанию участника;
 - 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;
- обоснование выбора языка программирования и используемых программных средств;
- структурная и функциональная схемы программного продукта;
- блок-схема работы основного алгоритма;
- схема базы данных;
- описание проведённых испытаний в соответствии с регламентом кейса (снимки экрана и/или запись экрана с работой);
- программный код (ссылка на репозиторий).

4. Регламент испытаний

- установка из файла;

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ
Профиль «Информационные технологии»
Командный кейс № 1 «Надёжность вычислительной техники»**

- выбор типа электронного компонента;
- подключение базы данных;
- расчёт срока годности;
- вывод графиков;
- масштабирование интерфейса;
- проверка работоспособности функционала программы, описанного в документации.

5. Примерный перечень средств и инструментов для выполнения задания

- <https://pythonworld.ru/>
- <https://ies.unitech-mo.ru/files/upload/publications/25559/fffa0bd8da0313be21ad5d99ab17c22c.pdf>
- <https://ohranatruda.ru/upload/iblock/131/4293850197.pdf>

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
**Командный кейс № 2 «Проектирование мультимедийного приложения с
элементами дополненной реальности»**

1. Условия

Участникам Олимпиады предлагается создать мультимедийное мобильное приложение с элементами дополненной реальности. При разработке приложения участники могут самостоятельно выбрать тематику. К примеру, некоторые из возможных тем перечисленных ниже:

- история;
- физика;
- астрономия.

Целевая аудитория и функциональные особенности приложения выбираются участником самостоятельно.

2. Техническое задание

Приложение должно быть спроектировано для запуска на одной из актуальных версий операционной системы мобильного устройства.

Описание функциональных особенностей приложения

Функциональные особенности приложения определяется участником олимпиады, исходя из выбранной участниками тематики. Приложение должно быть не только развлекательным, но и образовательным.

Интерфейс приложения должен быть понятным, при первом запуске необходимо провести обучение пользователя основным возможностям приложения.

Приложение может использовать сторонние сервисы для обработки информации, также в качестве способа представления информации можно использовать аудиоряд, но он не должен заменять текстовую информацию, а лишь дополнять её.

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
**Командный кейс № 2 «Проектирование мультимедийного приложения с
элементами дополненной реальности»**

3D-модели, используемые в приложении, могут быть сделаны самостоятельно или заимствованы с ресурсов без авторского права. В приложении необходимо использовать меточную или безметочную технологии дополненной реальности. В случае использования безметочной технологии приложение должно быть дополнено метками для возможности демонстрации разработанного продукта. Это может быть небольшой справочник, иллюстрирующий некоторую область выбранной темы.

3. Требования к документации разрабатываемого продукта

- титульный лист (с указанием названия кейса и перечислением членов команды);
- анализ технических требований;
- тематика приложения, для какой целевой аудитории и решения каких задач оно создано;
- обоснование выбора используемых программных средств;
- описание реализации, структура и особенности приложения, перечисление использованных сторонних библиотек и сервисов;
- структурная и функциональная схемы программного продукта;
- ссылка на видео, демонстрирующее работу приложения продолжительностью не более 2 минут;
- выводы по работе, возможности для доработки и развития;
- программный код (ссылка на репозиторий).

Пояснительная записка и демонстрация работы приложения призваны дать членам жюри понятное и краткое описание приложения и его функциональных

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
**Командный кейс № 2 «Проектирование мультимедийного приложения с
элементами дополненной реальности»**

возможностей.

4. Регламент испытаний

Проверка работоспособности приложения, описанного в пояснительной записке.

5. Примерный перечень средств и инструментов для выполнения задания

- среда разработки «Unity» <https://Unity.com>
- среда разработки Android Studio <https://developer.android.com/studio>
- документация для разработчиков Android приложений <https://developer.android.com/>
- документация для разработчиков IOS <https://developer.apple.com/documentation/>
- фреймворк Vuforia <https://developer.vuforia.com>
- справочник ARCore <https://developers.google.com/ar>
- справочник ARKit <https://developer.apple.com/augmented-reality/>

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
Командный кейс № 3 «Контроль повседневных расходов»

1. Условие

Современное мобильное устройство – удобный инструмент для каждодневного мониторинга расходов. Приложение для финансового мониторинга будет выполнять информационную функцию, автоматически осуществлять расчёты, предоставляя пользователю отчёты за определённый промежуток времени.

2. Техническое задание

Предлагается разработать веб-приложение или мобильное приложение для операционной системы Android и/или iOS.

Приложение должно предоставлять пользователю следующие возможности:

- регистрация, доступ к личному кабинету;
- добавление доходов и расходов с привязкой ко времени, назначение категорий трат из доступных (например, транспорт, аптеки, дом, одежда, супермаркеты, такси, прочее);
- фильтрация по категориям и периоду времени (например, месяц, квартал, полгода, год);
- построение диаграмм или графиков расходов за определённый период времени по категориям;
- прогнозирование общих расходов к концу месяца по текущим данным с помощью линейной регрессии и других моделей;
- экспорт данных за определённый промежуток времени в файл *.csv или *.xlsx;
- измерение общего остатка или стоимости каждой покупки в ценах базового товара, выбранного пользователем (при выборе конкретной операции как базовой стоимостью всех отображаемых на экране операций пересчитывается

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
Командный кейс № 3 «Контроль повседневных расходов»

в условных единицах, где одна условная единица равна стоимости выбранной операции);

- отображение текущего общего остатка с учётом инфляции на срок 3, 6 месяцев и 1 год (дисконтированная/приведённая стоимость с использованием индекса потребительских цен, актуальные данные об ИПЦ загружаются из сети Интернет).

Целевая аудитория включает людей с 14 лет, заинтересованных в эффективном управлении личным бюджетом. Интерфейс приложения должен быть интуитивно понятным, все графические элементы должны иметь единый стиль, рекомендуется использовать анимированные переходы при навигации. Приложение должно работать стабильно, адаптироваться под различные конфигурации устройств, не должно требовать избыточных вычислительных мощностей или большого объёма памяти. Архитектура приложения должна обеспечивать возможность внесения небольших изменений с минимальным переписыванием программного кода: например, добавление нового типа фильтра должно приводить к редактированию только нескольких строк в исходном коде. Разработанная программная система должна обеспечивать целостность и конфиденциальность данных пользователей.

3. Требования к документации

При сдаче материалов по данному кейсу предоставляются:

- ссылка на удалённый репозиторий в системе контроля версий с полным документированным исходным кодом проекта, историей изменений и исполняемым файлом приложения;
- ссылка на видео с демонстрацией основных функциональных возможностей приложения длительностью не более 1 минуты;

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
Командный кейс № 3 «Контроль повседневных расходов»

- техническая документация, включающая титульный лист со списком команды, обоснование выбора архитектуры приложения, структур данных и СУБД, краткое описание используемых средств разработки, информацию о функциональных возможностях системы, UML диаграммы, описание реализованных алгоритмов и скриншоты приложения.

При построении клиент-серверной архитектуры необходимо обеспечить работоспособность серверной части на время проведения испытаний.

4. Регламент испытаний

- участники устанавливают приложение, запускают его на физическом устройстве, демонстрируют некоторые функциональные возможности приложения согласно техническому заданию;

- участники запускают приложение на другом физическом устройстве или на эмуляторе, проводят авторизацию с использованием существующего аккаунта, где заранее добавлены покупки, демонстрируют оставшиеся возможности приложения.

5. Примерный перечень средств и инструментов для выполнения задания

- Android. Программирование для профессионалов. Б. Филлипс, К. Стюарт, К. Марсикано

- The Busy Coder's Guide to Android Development. Mark L. Murphy

- Elements of Android Room. Mark L. Murphy

- Flutter Cookbook: Over 100 proven techniques and solutions for app development with Flutter 2.2 and Dart. Simone Alessandria, Brian Kayfitz

- документация для разработчиков Android приложений
<https://developer.android.com/>

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
Командный кейс № 3 «Контроль повседневных расходов»

- документация для разработчиков iOS приложений – Технологии <https://developer.apple.com/documentation/technologies>
- список иконок (Material icons) <https://fonts.google.com/icons>
- сервис для разработки интерфейсов Figma <https://www.figma.com/>
- документация по React Native <https://reactnative.dev/docs/getting-started>

МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ ОЛИМПИАДА ШКОЛЬНИКОВ

Профиль «Информационные технологии» Командный кейс № 4 «Туристский помощник»

1. Условия

Мотивация обучающихся – это важная задача на всех уровнях обучения. Достаточно легко повысить мотивацию обучающихся на среднем уровне образования. Геймификация – это процесс внесения игровых элементов в образовательный процесс с целью его разнообразия, повышения мотивации обучающихся, наглядной демонстрации сложных процессов более простыми объяснениями. При этом современные технические средства могут значительно облегчить внедрение этого подхода в образовательный процесс. Так, например, технологии распознавания образов и поиска информации по найденным изображениям – это весьма популярная и очень нужная технология сегодня.

Участникам Олимпиады предлагается создать мобильное приложение для получения информации о памятнике, скульптуре, здании и т.п., который пользователь видит перед собой.

2. Техническое задание

Приложение должно быть разработано для носимого устройства (мобильные телефоны, планшеты, устройства разработанные на платформах Raspberry Pi, NVIDIA® Jetson™, Arduino и т.д.) любой архитектуры и производителя с использованием любых языков программирования. Допустимо использовать сторонние библиотеки, облачные сервисы. Недопустимо использование стационарного компьютера или ноутбука для финального использования приложения, в процессе разработки они могут быть использованы как средства разработки ПО.

Описание функциональных требований приложения.

Пользователь имеет возможность навести камеру мобильного устройства с запущенным приложением на находящийся перед ним памятник (скульптуру, здание, стелу и т.д.) и предложить приложению определить этот памятник и найти информацию по нему в локальной базе данных или в интернете.

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
Командный кейс № 4 «Туристский помощник»

Приложение имеет внутреннюю базу данных памятников и нейронную сеть, с помощью которой оно может распознать памятник (т.е. найти подходящие объекты в базе данных). Найденные объекты предлагаются пользователю в порядке уменьшения схожести с рассматриваемым. Если пользователь отверг все предложения, то приложение должно попытаться найти фотографию в интернете (например, используя API существующих поисковиков). Если в интернете изображение найдено (т.е. пользователь признал памятник) – он добавляется в базу данных и нейронная сеть дообучается. Если и этот вариант не приносит результата – приложение предлагает пользователю добавить памятник в свою БД с указанием его параметров (и дообучает нейронную сеть). Процесс дообучения и переобучения нейросети должен происходить с использованием нескольких фотографий определяемого памятника с разных ракурсов – для повышения эффективности работы приложения. Функция дообучения нейронной сети, используемой для распознавания объектов, может происходить не на самом устройстве пользователя, а в облачных сервисах или на высокопроизводительной рабочей станции отдельно от мобильного устройства.

3. Требования к документации

- титульный лист (с указанием названия кейса и перечислением членов команды);
- анализ технических требований;
- обоснование выбора программно-технических средств, используемых при реализации кейса;
- структурная и функциональная и схемы программного продукта;
- блок-схема работы основного алгоритма;
- схема базы данных;

МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ ОЛИМПИАДА ШКОЛЬНИКОВ

Профиль «Информационные технологии» Командный кейс № 4 «Туристский помощник»

- описание проведённых испытаний в соответствии с регламентом кейса (снимки экрана и/или запись экрана с работой);
- программный код (ссылка на репозиторий).

4. Регламент испытаний

- участник должен продемонстрировать видео с работой приложения;
- при работе приложения должны быть идентифицированы разные объекты, представляющие интерес для туристов;
 - необходимо подтвердить работу приложения на известном приложению объекте;
 - необходимо подтвердить работу приложения при поиске информации об объекте в сети интернет;
 - необходимо подтвердить возможность добавления объекта в БД и нейронную сеть (с учётом последующего повторного поиска по его фотографии);
 - по найденному объекту приложение должно выдавать статью из любого источника (например, Wikipedia), если она добавлена в БД.

5. Примерный перечень средств и инструментов для выполнения задания

- среда разработки Android Studio <https://developer.android.com/studio>
- документация для разработчиков Android приложений <https://developer.android.com/>
- документация для разработки под iOS <https://developer.apple.com/documentation/>
- документация по инструментам Firebase <https://firebase.google.com/docs>
- документация по инструментам OpenCV <https://opencv.org/>

МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ ОЛИМПИАДА ШКОЛЬНИКОВ

Профиль «Информационные технологии» Командный кейс № 4 «Туристский помощник»

- Keras — открытая библиотека, написанная на языке Python и обеспечивающая взаимодействие с искусственными нейронными сетями.

<https://keras.io/>

- PyTorch — фреймворк машинного обучения для языка Python с открытым исходным кодом. <https://pytorch.org/>

- TensorFlow — открытая программная библиотека для машинного обучения. <https://tensorflow.org/>

- Yolo v4, v3 and v2 for Windows and Linux <https://github.com/AlexeyAB/darknet>

- платформа Raspberry Pi <https://www.raspberrypi.org/>

- платформа NVIDIA® Jetson™

<https://developer.nvidia.com/embedded/jetson-modules>

- платформа Arduino <http://www.arduino.cc/>

- возможно использование других библиотек, сервисов, платформ.

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
**Командный кейс № 5 «Мобильное приложение –
Upcycling»**

1. Условия

Мобильные устройства стали неотъемлемой частью жизни и помощником человека в большом городе. Большинство из нас раз в год-два приобретают новейшую модель смартфона, умных часов или планшета, пополняя свою коллекцию неиспользуемых устройств. Наверняка у каждого из вас дома, где-нибудь в тумбочке, лежат несколько уже неиспользуемых смартфонов, которые, тем не менее, жалко выкидывать.

Именно для таких устройств, вполне рабочих, но уже несколько устаревших, была придумана идея Upcycling. Наверняка, вы все знаете об идее Recycling, т.е. переработки отходов с целью их дальнейшего использования. Идея Upcycling также направлена на повторное использование вещей, однако без их переработки. Более подробно об Upcycling и примерах воплощения этой идеи вы можете посмотреть в нашем видеокурсе на канале IT Школы Samsung.

Наш кейс призван генерировать идеи для социально-ориентированных мобильных приложений, которые реализовывали бы идею Upcycling, и сделать первый шаг в их реализации.

Вам необходимо разработать мобильное приложение для платформы Android, которое решало бы какую-либо социально значимую проблему в вашем городе, и использовало устаревшие мобильные устройства, реализуя идею Upcycling.

2. Техническое задание

Приложение должно быть реализовано на языках программирования Java или Kotlin для платформы **Android** вплоть до версии **5.0**.

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
**Командный кейс № 5 «Мобильное приложение –
Upcycling»**

Описание функционала приложения

Функционал приложения определяется участником олимпиады, исходя из тех задач, которые решает приложение.

Приложение должно использовать в своей работе камеру смартфона или gps и другие датчики смартфона.

Приложение может использовать сторонние сервисы для обработки информации, например, облачные сервисы компьютерного зрения или распознавания голоса.

3. Требования к документации

- титульный лист (с указанием названия кейса и перечислением членов команды);
- ссылка на репозиторий в системе контроля версий, где размещаются исходный код и исполняемый файл (apk) приложения;
- пояснительная записка с описанием назначения и функционала приложения. В пояснительной записке должны быть описаны все возможные варианты использования приложения. Именно в соответствии с описанным в пояснительной записке функционалом приложения будут происходить испытания приложения при очном оценивании. В случае наличия в приложении механизма аутентификации, в пояснительной записке должны присутствовать тестовые учётные данные;
- схема базы данных;
- ссылка на видео, демонстрирующее работу приложения продолжительностью не более 2 минут;
- презентация в формате ppt со следующей структурой:

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
**Командный кейс № 5 «Мобильное приложение –
Upcycling»**

- заголовок: наименование кейса, ФИО участников;
 - введение: идея приложения, для какой целевой аудитории и решения каких задач оно создано;
 - описание реализации: структура и особенности приложения, схема классов, перечисление использованных сторонних библиотек и сервисов;
 - заключение: выводы по работе, возможности для доработки и развития.
- Видео и презентация призваны дать членам жюри понятное и краткое описание приложения и его преимуществ.

4. Регламент испытаний

- установка из apk файла минимум на нескольких мобильных устройствах с разными версиями Android, для проверки совместимости со старыми версиями Android;
- проверка работоспособности функционала приложения, описанного в пояснительной записке. В случае наличия в реализации серверной части необходимо обеспечить её работоспособность в режиме 24×7.

5. Примерный перечень средств и инструментов для выполнения задания

- открытый учебник «IT Школа Samsung» <https://myitschool.ru/edu/edu>
 - курс «Galaxy Upcycling – Новая жизнь старого смартфона»
<https://www.youtube.com/playlist?list=PLa2T1zmZ6w5JiGjWTyaYamyHN151HWV2>
- Q**
- материалы Samsung Android Bootcamp 2020
<https://www.youtube.com/playlist?list=PLa2T1zmZ6w5IVEYXseMrWOGfShsswVl2A>

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
**Командный кейс № 5 «Мобильное приложение –
Upcycling»**

- материалы Samsung Android Bootcamp 2021
<https://www.youtube.com/playlist?list=PLa2T1zmZ6w5KzKoh9M91vk1LBqpc-WtoS>
- среда разработки Android Studio <https://developer.android.com/studio>
- документация для разработчиков Android-приложений
<https://developer.android.com/>
- документация по языку Kotlin <https://kotlinlang.org/docs/reference/>
- документация по языку Java <https://docs.oracle.com/>
- документация по инструментам Firebase <https://firebase.google.com/docs>

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
**Командный кейс № 6 «Система обнаружения поддельных сайтов кредитно-
финансовых организаций»**

1. Условия

Требуется разработать веб-приложение решающее задачу автоматизированного поиска и выявления мошеннических ресурсов в сети Интернет. Полученное решение должно выполнять поиск и выявление в автоматизированном режиме потенциальных мошеннических ресурсов, к которым следует относить:

- сайты, фактически являющиеся клонами официальных сайтов действующих организаций кредитно-финансовой сферы;
- сайты, наименования доменных имен которых схожи по написанию (вхождения слов, комбинаций, аббревиатур) с названиями действующих организаций кредитно-финансовой сферы независимо от их содержимого (которое может быть временным, для последующей замены на клон, или вообще отсутствовать);
- сайты, нацеленные на сбор персональных данных, сбор данных банковских карт и других платежных средств, сбор аутентификационных данных для входа в личные кабинеты клиентов организаций кредитно-финансовой сферы, сбор денежных средств под предлогом вознаграждения, а также имитирующие деятельность организаций кредитно-финансовой сферы.

К примеру, решением может служить запуск рейтингового веб-ресурса с пополняемой базой данных фишинговых сайтов. К ресурсу может обращаться любой желающий для его наполнения и просмотра. При этом аналитическая подсистема такого веб-ресурса может быть основана как на комментариях (рейтингах) пользователей, так и на применении каких-либо интеллектуальных алгоритмов анализа.

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
**Командный кейс № 6 «Система обнаружения поддельных сайтов кредитно-
финансовых организаций»**

2. Техническое задание

Никаких жёстких требований и ограничений на наличие внешних источников, применение готовых модулей программного обеспечения с открытым исходным кодом и т.п. разрабатываемого решения не накладывается.

В рамках реализации продукта рекомендуется воспользоваться следующими приёмами:

для упаковки веб-приложения лучше всего подходит технология контейнеризации Docker;

- документирование кода позволяет избавиться от необходимости его изучения и лучше понять принцип его работы;
- для хранения данных лучше всего использовать реляционные и нереляционные базы данных;
- размещение веб-приложения в облаке избавляет от необходимости писать подробную инструкцию по его развёртыванию на локальной машине.

3. Требования к документации

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

- титульный лист (с указанием названия кейса и перечислением членов команды);
- анализ технических требований;
- обоснование выбора языка программирования и используемых программных средств;
- структурная и функциональная и схемы разрабатываемых модулей;
- блок-схема работы разрабатываемых модулей;

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
**Командный кейс № 6 «Система обнаружения поддельных сайтов кредитно-
финансовых организаций»**

- общая схема базы данных;
- скриншоты и ссылка на видео, демонстрирующее работу приложения продолжительностью не более 5 минут;
- программный код (ссылка на репозиторий в git).

4. Регламент испытаний

1. Установка сервиса в соответствии с инструкцией из технической документации. В случае функционирования продукта в облаке необходимо обеспечить ее работоспособность в режиме 24x7;
2. Проверка работоспособности функционала сервиса осуществляется по каждому из следующих пунктов в соответствии с технической документацией:
 - приложение функционирует в облаке;
 - наличие обновляемого списка функционирующих фишинговых сайтов;
 - наличие умного поиска среди найденных фишинговых сайтов;
 - наличие возможности добавления пользователем информации о мошенническом сайте;
 - наличие регистрации и аутентификации пользователя на сайте;
 - наличие функции автоматического выявления мошеннических сайтов на основе ключевых признаков;
 - наличие рейтинговой системы пользователей;
 - наличие функции модерирования;
 - пользовательский интерфейс удобен и интуитивно понятен.

3. Примерный перечень средств и инструментов для выполнения задания

- система контейнеризации сервисов <https://docs.docker.com/get-started/>

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
**Командный кейс № 6 «Система обнаружения поддельных сайтов кредитно-
финансовых организаций»**

- среда разработки Visual Code <https://code.visualstudio.com/>
- серверное программирование веб-сайтов:
<https://developer.mozilla.org/ru/docs/Learn/Server-side>
- особенности фишинговых сайтов: <https://iclubspb.ru/fishing/>

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
Командный кейс № 7 «Умный дом»

1. Условия

Современный и комфортный дом уже сложно представить без какой-либо автоматизации и учёта. Но ни то, ни другое не может работать без информации. Как мы можем управлять батареей, если не понимаем, какая температура в комнате. Или, как, без понимания уровня углекислого газа мы сможем понять, что необходимо включить систему вентиляции. Соответственно, умный дом начинается с «органов осязания»: всевозможных датчиков, которые могут сообщать, что происходит и, опираясь на данные с которых, мы можем управлять теми или другими управляющими механизмами.

Дано: набор датчиков (к кейсу приложена программа-генератор данных с набора датчиков (температуры, влажности, давления и т.п.), которые шлют информацию через MQTT-брокера. Датчики разные, и часть из них совместима с системой умный дом HomeAssistant, а часть – нет. Данные датчиков, несовместимые с Home Assistant необходимо на лету преобразовывать в совместимые.

Данные длительного хранения от датчиков хранятся в базе данных InfluxDB, куда их необходимо записать, получая их с брокера.

Текущее состояние датчиков можно посмотреть на панелях HomeAssistant, но гораздо интереснее, если их можно будет посмотреть в системе Grafana.

2. Техническое задание

Разработать систему, состоящую из:

- сервера умного дома (HomeAssistant);
- MQTT-брокера (Mosquitto, или любого иного MQTT-брокера);
- сервера БД (InfluxDB, Prometheus, OpenTSDB или иной time-series базы данных – TSDB);
- сервера работы с графиками (Grafana);

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
Командный кейс № 7 «Умный дом»

- компонента конвертирующего несовместимые данные с HomeAssistant в совместимые (разрабатывается в рамках кейса);
- компонента, сохраняющего данные датчиков на стороне сервера БД InfluxDB (разрабатывается в рамках кейса).

Компоненты системы работают следующим образом:

- входящая информация от датчиков, поступающая от брокера MQTT обрабатывается компонентом [5] и в случае нахождения несовместимых данных данные конвертируются в нужный формат и отправляются обратно брокеру;
- входящая информация от MQTT-брокера сохраняется компонентом [6] в БД InfluxDb;
- корректная информация, поступающая от датчиков, визуализируется на стороне сервера умного дома HomeAssistant (компонент [1]);
- информация, хранящаяся в базе данных InfluxDB, визуализируется компонентом построения графиков Grafana (компонент [4]).

Компонента 5 должна иметь архитектуру, которая позволила бы, не изменяя программной части, добавлять новые несовместимые датчики, т.е., в случае появления нового датчика, мы добавляем его описание и правила конвертации данных в конфигурацию, и компонента продолжает работать без перекомпиляции.

Компонента 6 должна настраиваться через конфигурацию и должна уметь сохранять в базу любую информацию, поступающую от сенсоров через MQTT-брокера. Формат данных от датчиков подробно описан в документации на HomeAssistant. И компонента может настраиваться самостоятельно используя информацию, поступающую от MQTT-брокера.

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
Командный кейс № 7 «Умный дом»

Таким образом, требуется разработать два модуля: 5 и 6 и развернуть и настроить готовые сервера: компоненты 1–4. После этого произвести интеграцию модулей друг с другом и показать итоговый результат.

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

Действия компонент 5 и 6 должны логироваться в файл, или стандартными средствами систем Unix.

3. Требования к документации

Итоговый документ, прилагаемый к кейсу, должен включать в себя следующие разделы:

- титульный лист (с указанием названия кейса и перечислением членов команды);
- анализ технических требований;
- обоснование выбора языка программирования и используемых программных средств;
- структурная и функциональная и схемы разрабатываемых модулей;
- блок-схема работы разрабатываемых модулей;
- общая схема базы данных, которую создаёт компонент 6;
- описание проведённых испытаний в соответствии с регламентом кейса (снимки экрана и/или запись экрана с работой);
- программный код (ссылка на репозиторий в git).

4. Регламент испытаний

Испытания разработанного комплекса заключаются в последовательной демонстрации следующих шагов:

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ
Профиль «Информационные технологии»
Командный кейс № 7 «Умный дом»**

- производится запуск всех компонентов системы;
- производится показ, какие данные проходят через компоненту 2;
- производится накопление получаемых данных;
- демонстрируется, как данные отображаются в компоненте 1;
- демонстрируется, как конвертируются несовместимые данные, показывается их отображение в компоненте 1;
- демонстрируется, как производится сохранение данных в компоненте 3;
- демонстрируется наполнение данных компонентой 3 путём выполнения запросов к данным в консоли;
- демонстрируется отображение полученных данных компонентой 4;
- демонстрируются расширенные возможности по отображению данных (различные графики и дашборды) в компонентах 1 и 4.

5. Примерный перечень средств и инструментов для выполнения задания

<https://www.home-assistant.io/> – сервер HomeAssistant

<https://www.home-assistant.io/docs/> – документация к серверу HomeAssistant

<https://mosquitto.org/> – брокер MQTT

<https://www.influxdata.com/> – БД InfluxDB

<https://prometheus.io/> – БД Prometheus

<http://opentsdb.net/> – БД OpenTSDB

<https://grafana.com/> – Grafana

<https://www.virtualbox.org/> – диспетчер виртуальных машин

<https://ubuntu.com/> – серверная ОС Ubuntu

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
**Командный кейс № 8 «Мониторинг комплекса
энергосбережения»**

1. Условия

Адаптивный комплекс энергосбережения (АКЭС) относится к стационарным комплексам улучшения качества электроэнергии на трансформаторных подстанциях и в электросетях предприятий. Улучшения достигаются за счет повышения коэффициента мощности, ограничения максимальных пусковых токов и компенсации перекоса фаз.

АКЭС подключается параллельно потребителям электроэнергии и производит измерения основных параметров электрической сети в соответствии с режимом её работы.

При осуществлении компенсации реактивной мощности в трёхфазной электросети, основным показателем работы АКЭС является эффективность, которая определяется следующей формулой:

$$\eta = \frac{P_{off} - P_{on}}{P_{off}} \cdot 100\%$$

где P_{off} – сумма активной мощности по каждой фазе при выключенной системе, а P_{on} – сумма активной мощности при включённой системе.

Каждый АКЭС оснащён встроенным 3G-модемом и может осуществлять передачу данных телеметрии на удалённый сервер.

Описание параметров телеметрии комплекса представлено на [специализированном информационном ресурсе](#).

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
**Командный кейс № 8 «Мониторинг комплекса
энергосбережения»**

2. Техническое задание

Функциональные требования

Разработать информационную систему, обеспечивающую работу с данными телеметрии от АКЭС.

Информационная система должна реализовывать следующую функциональность:

- отображение графика эффективности работы комплекса, за различные периоды времени:

- неделя;

- день;

- выгрузка данных в виде таблицы с полями:

- время начала теста;

- время окончания теста;

- косинус угла по каждой фазе;

- накопительная активная мощность;

- эффективность;

- авторизация пользователей двух типов:

- администратор;

- пользователь;

- осуществление регистрации новых пользователей администратором системы;

- обеспечение контроля доступа к ресурсам информационной системы:

- только авторизованные пользователи могут просматривать данные телеметрии;

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
**Командный кейс № 8 «Мониторинг комплекса
энергосбережения»**

- администратор выдаёт права доступа к АКЭС пользователям системы;
- процедура выдачи прав осуществляется по серийному номеру АКЭС;
- пользователь имеет доступ к только к тем комплексам, которые были закреплены за ним администратором системы.

Дополнительные требования

Коммуникация с комплексами должна осуществляться по протоколу HTTPS, при этом необходимо реализовать серверную часть, принимающую данные телеметрии от АКЭС.

Необходимо реализовать клиентскую часть системы в виде графического интерфейса пользователя.

Рекомендации

Пользовательский интерфейс рекомендуется делать кроссплатформенным, удобным и понятным пользователю.

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

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

При разработке системы рекомендуется использовать unit-тестирование.

Вспомогательные ресурсы

Пример отправляемых комплексом данных можно скачать по [ССЫЛКЕ](#).

Расшифровку полей в отправляемых файлах можно посмотреть по [ССЫЛКЕ](#).

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
**Командный кейс № 8 «Мониторинг комплекса
энергосбережения»**

Для облегчения разработки системы можно воспользоваться [программной моделью АКЭС](#).

3. Требования к документации

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

- 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 проведённых испытаний в соответствии с регламентом кейса (снимки экрана и/или запись экрана с работой);
- программный код (ссылка на репозиторий).

4. Регламент испытаний

Демонстрация испытаний осуществляется с использованием программной модели комплекса:

- производится запуск серверной части;
- от имени администратора производится регистрация не менее двух пользователей;
- от имени администратора производится привязка не менее трех серийных номеров комплексов к пользователям;

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
**Командный кейс № 8 «Мониторинг комплекса
энергосбережения»**

- производится запуск программной модели с указанием серийных номеров комплексов и адреса сервера, не менее четырёх раз производится передача телеметрии (даты измерений находятся в периоде 2–3 недель);
- от имени пользователя осуществляется просмотр графиков за произвольные неделю и за день;
- от имени пользователя выгружается таблица за произвольные неделю и день;
- жюри осуществляет проверку того, что от имени одного пользователя нет возможности просматривать данные другого.

5. 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 № 9 «Компьютерное зрение в сельском хозяйстве»

1. Условия

Ключевая потребность современного сельского хозяйства – эффективное производство. Тренд № 1 – автоматизация процессов посева и уборки урожая. В настоящем кейсе учащимся предлагается создать проект сервиса, который автоматически рассчитывает число колосков пшеницы на поле. Исходя из этого производство может точно рассчитать количество техники, необходимое для уборки урожая.

Квадрокоптер летает над пшеничным полем. Он зависает в нескольких точках над полем и делает фотографии пшеницы под собой. Каждая фотография снабжается метками высоты и координат, на которых фото выполнено. Данные с квадрокоптера обрабатываются специальной программой, предсказывающей урожай с поля. Форма поля заранее известна.

2. Техническое задание

Разработать приложение рассчитывающие размер урожая с пшеничного поля, опираясь на входные данные.

Приложение должно включать в себя:

- алгоритм машинного обучения, детектирующий на изображении колосья пшеницы;
- алгоритм расчёта плотности пшеницы на поле (количество колосков на квадратный метр) и общего количества пшеницы на поле;
- алгоритм визуализации результатов расчётов;
- интерфейс пользователя, связывающий воедино вышестоящие составляющие и позволяющий работать с входными и выходными данными.

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
Командный кейс № 9 «Компьютерное зрение в сельском хозяйстве»

Входные данные:

а) форма и размеры поля.

Координаты двумерных точек, которые при последовательном соединении образуют контур поля;

б) набор изображений с метками.

Метки представляют собой двумерные координаты точки, в которой выполнено фото, и расстояние от объектива камеры до поверхности земли.

При разработке приложения, следует принять следующие допущения:

- поверхность поля – горизонтальная плоскость;
- высота пшеницы на всём поле одинакова и составляет 105 см;
- камера всегда направлена вертикально вниз;
- высота полёта квадрокоптера от 1,5 до 3 метров;
- вертикальный угол обзора камеры 70 градусов;
- горизонтальный угол обзора камеры 120 градусов;
- искажения объектива пренебрежительно малы;
- все возможные поля помещаются в квадрат 1000 на 1000 метров.

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

Алгоритм расчёта плотности пшеницы на поле, может быть не один. Наличие дополнительных, существенно отличающихся друг от друга алгоритмов расчёта, является плюсом для приложения.

Наличие визуализации, не только конечных результатов расчёта, но и промежуточных результатов расчёта, является плюсом. Кроме того, визуализация должна помогать понять суть алгоритма расчёта и отличие одного алгоритма расчёта от другого, при наличии нескольких.

Формат входных, с которыми будет работать приложение, необходимо разработать! Формат должен быть разработан с точки зрения удобства

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
Командный кейс № 9 «Компьютерное зрение в сельском хозяйстве»

хранения и работы с данными. Исходите из того, что после внедрения программы в отрасль, квадрокоптер, будет передавать входные данные в разработанном вами формате.

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.

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

- титульный лист (с указанием названия кейса и перечислением членов команды);
- описание и обоснование используемой модели машинного обучения, подготовки данных для обучения и параметров обучения, включая оценку точности на платформе Sim;
- описание и обоснование используемых алгоритмов расчёта плотности пшеницы и количества пшеницы на поле;
- программный код, например, ссылка на репозиторий;
- описание и обоснование разработанной структуры входных данных;
- описание интерфейса приложения;
- описание проведенных испытаний в соответствии с регламентом кейса (снимки экрана и/или запись экрана с работой);

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
Командный кейс № 9 «Компьютерное зрение в сельском хозяйстве»

Формат описания и используемые виды схем на усмотрение участников. Выбор должен способствовать понятности и наглядности документации. Например, ссылки на файлы исходного кода, помогут разобраться в реализации алгоритмов.

4. Регламент испытаний

- запуск приложения;
- ввод в качестве входных данных заранее подготовленных данных;
- демонстрация визуализации результатов расчёта;
- демонстрация возможностей работы с входными и выходными данными;
- 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.