



Учебно-методическое объединение федеральных казенных профессиональных образовательных учреждений, подведомственных Министерству труда и социальной защиты Российской Федерации

Секция по вопросам информационного обеспечения, реализации ДОТ и ЭО



## **СБОРНИК МАТЕРИАЛОВ**

**II студенческой научно-практической конференции**

**ПРИОРИТЕТНЫЕ НАПРАВЛЕНИЯ В  
ОБЛАСТИ ЦИФРОВОЙ  
ТРАНСФОРМАЦИИ**



15 декабря 2022 года

## ПРЕДИСЛОВИЕ

Информационные технологии играют важнейшую роль в современном мире. Они занимают уникальное положение в нашем обществе, не просто оказывают влияние на его экономические и социальные институты, но и являются основным двигателем глобального экономического роста, проникая во все сферы производственной деятельности.

Вашему вниманию предлагается сборник материалов II студенческой научно-практической конференции «ПРИОРИТЕТНЫЕ НАПРАВЛЕНИЯ В ОБЛАСТИ ЦИФРОВОЙ ТРАНСФОРМАЦИИ». Сборник включает результаты исследовательской деятельности обучающихся федеральных казенных профессиональных образовательных учреждений, подведомственных Министерству труда и социальной защиты Российской Федерации.

В конференции приняли участие 20 обучающихся, представляющих 6 профессиональных образовательных организаций, подведомственных Минтруда России.

В сборнике представлены исследовательские работы по 4-м важнейшим направлениям цифровизации и развития современных IT-технологий:

- информационные технологии в образовании и науке, автоматизация научно-исследовательской деятельности;
- информационные технологии в области искусственного интеллекта, машинное обучение, робототехника и киберфизические системы;
- информационные технологии в различных отраслях производственной и общественной деятельности: прикладные, офисные и мультимедийные технологии, интернет вещей, виртуальная реальность и т.д.
- информационные технологии в области хранения и обработки данных, Data Science: big data, грид-технологии, облачные вычисления, блокчейн и т.д.

*Материалы студенческой научно-практической конференции «Приоритетные направления в области цифровой трансформации», 2022 г.*

*©УМО ФКПОУ Минтруда России*

## СОДЕРЖАНИЕ:

### **СЕКЦИЯ 1. ДОСТИЖЕНИЯ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ В РАЗЛИЧНЫХ ОТРАСЛЯХ ПРОИЗВОДСТВЕННОЙ И ОБЩЕСТВЕННОЙ ДЕЯТЕЛЬНОСТИ: ПРИКЛАДНЫЕ, ОФИСНЫЕ И МУЛЬТИМЕДИЙНЫЕ ТЕХНОЛОГИИ**

Андреев Александр Михайлович. Использование 3D принтера в ремонте радиоэлектронной техники.....	4
Бекмурзина Алина Аликовна. Сервисы для организаций видеоконференций .....	8
Кокарева Александра Олеговна. QR коды вокруг нас.....	15
Котельников Сергей Сергеевич. разработка голосового помощника с применением языка программирования Python .....	19
Пручковский Николай Николаевич, Харламов Илья Константинович, Федорченко Дарья Андреевна. Разработка базы данных и web-приложения для организации online конкурсов .....	21
Сагайдак Валерий Александрович. Будущее веб – разработки.....	29

### **СЕКЦИЯ 2. ДОСТИЖЕНИЯ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ В ОБЛАСТИ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА, МАШИННОЕ ОБУЧЕНИЕ, РОБОТОТЕХНИКА И КИБЕРФИЗИЧЕСКИЕ СИСТЕМЫ**

Козлов Вадим Алексеевич. Искусственное искусство: как нейросети научились творить?.....	37
Мурашев Владимир Константинович. Что такое умный дом.....	40
Чистов Фёдор Алексеевич. Система «Умный дом». Комплекс датчиков протечек воды .....	44

### **СЕКЦИЯ 3. ДОСТИЖЕНИЯ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ В ОБРАЗОВАНИИ И НАУКЕ, АВТОМАТИЗАЦИЯ НАУЧНО-ИССЛЕДОВАТЕЛЬСКОЙ ДЕЯТЕЛЬНОСТИ**

Багина Екатерина Сергеевна, Казаков Никита Михайлович. Разработка электронного дневника .....	49
Зеленская Ангелина Владимировна, Уралова Ангелина Анатольевна. Интегрированный программный комплекс для психологической диагностики уровня эмоционального интеллекта обучающихся .....	55
Шевченко Данила Александрович. Языки программирования на русском языке .....	65
Ночевой Сергей Сергеевич. На русском языке программировать нужно!.....	77

### **СЕКЦИЯ 4. ДОСТИЖЕНИЯ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ В ОБЛАСТИ ХРАНЕНИЯ И ОБРАБОТКИ ДАННЫХ, DATA SCINCE: BIG DATA, ГРИД-ТЕХНОЛОГИИ, ОБЛАЧНЫЕ ВЫЧИСЛЕНИЯ, БЛОКЧЕЙН.**

Меркулова Елизавета Сергеевна. Практические аспекты использования облачных технологий.....	86
Сумская Мила Евгеньевна, Андросов Григорий Игоревич. Современный подход к разработке программного обеспечения. Практика DEVOPS, контейнеризация.....	91

# СЕКЦИЯ 1. ДОСТИЖЕНИЯ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ В РАЗЛИЧНЫХ ОТРАСЛЯХ ПРОИЗВОДСТВЕННОЙ И ОБЩЕСТВЕННОЙ ДЕЯТЕЛЬНОСТИ: ПРИКЛАДНЫЕ, ОФИСНЫЕ И МУЛЬТИМЕДИЙНЫЕ ТЕХНОЛОГИИ

## ИСПОЛЬЗОВАНИЕ 3D ПРИНТЕРА В РЕМОНТЕ РАДИОЭЛЕКТРОННОЙ ТЕХНИКИ

*Андреев Александр Михайлович  
ФКПОУ «ИвРТТИ» Минтруда России  
Гордеев Александр Викторович*

**Аннотация.** Ремонт радиоэлектронных приборов и электроники — комплекс мероприятий, направленных на сохранение и поддержание в работоспособном состоянии прибора. Ремонт можно подразделить на мелкий, средний и капитальный. При мелком ремонте устраняются мелкие неисправности, регулируются несложные узлы. При среднем ремонте заменяются вышедшие из строя отдельные детали, производится регулировка. При капитальном ремонте заменяются или восстанавливаются отдельные узлы, выполняются сложные наладочные и регулировочные работы.

Цена услуг по ремонту электроники зависит от:

- категории сложности оборудования;
- типа и модели оборудования;
- характера поломки.

Работы по восстановлению работоспособности включают в себя:

- Сборка/разборка блока;
- Анализ видимых дефектов;
- Проверка параметров электронных компонентов;
- Очистка плат;
- Проверка, протяжка электрических соединений силовых элементов;
- Проверка состояния изоляции отдельных узлов;
- Анализ кодов неисправностей;
- Замена неисправных деталей, в том числе с истекающим сроком годности;
- Устранение обнаруженных неисправностей;
- Тестирование на стенде;
- Гарантийные обязательства.

Спрос на услуги по ремонту и обслуживанию бытовой техники за последние три месяца вырос на 43% по сравнению с тем же периодом годом ранее. Причина такой заботы о бывших в употреблении девайсах — товарный дефицит и подорожание новой техники, говорят эксперты

В связи с изменением геополитической ситуации и сложностями с поставками новых товаров потребители пытаются поддержать работоспособность имеющейся у них бытовой техники и продлить срок ее службы. К такому выводу пришли аналитики «Авито Услуги» (анализировались запросы 23 млн человек).

Спрос на ремонт и обслуживание радиоэлектронных приборов (техники и электроники) в 2022 году увеличился на 13%, а по сравнению с аналогичным периодом прошлого года — на 43%. Предложение услуг по ремонту выросло на 12% по сравнению с прошлым годом.

«Если в прошлом потребители предпочитали регулярно обновлять бытовую технику в доме, то сейчас отношение к приборам становится более осознанным», — рассуждает директор «Авито Услуги» Георгий Акопян. По его словам, это касается как крупной бытовой техники, как, например, стиральных машин или холодильников, так и небольших приборов (микроволновок, пылесосов или чайников). Ранее их охотно меняли на новые, а теперь чаще отдают в ремонт. «Текущий рост спроса можно рассматривать как ренессанс индустрии бытовых услуг в онлайн-формате», — говорит Акопян.

Самыми активно растущими категориями по спросу стали ремонт мелких бытовых приборов (+ 20% в мае к марту 2022 года) и ремонт крупной бытовой техники (+32% в мае к марту 2022 года).

Цена услуг по ремонту обычно составляет от 30 до 50 % стоимости нового устройства. Экономия достигается путем скрупулезного расчета цен и уменьшения времени ремонта. Основную ценовую составляющую обычно составляют стоимость запчастей и стоимость работ.

В технике очень часто выходят из строя механические элементы: шестерни, корпуса, зажимающая часть, крепеж, держащий определённую часть прибора. При этом может потребоваться заменить запчасти.

Варианты доставки запчастей:

- Заказ в интернет магазинах России. В практике ремонта часто бывает , что нет нужных деталей на торговых площадках.
- Заказ с иностранных интернет магазинов(джум, алиэкспрес и т.д). Некоторые запчасти могут быть не представлены на данных торговых площадках ввиду санкций. Запчасти зачастую приходится ждать более 2х месяцев и нет гарантий их качества и хорошей доставки (может прийти сломанная или с браком). При этом ремонт затягивается и в большинстве случаев заказчик будет отказываться от ремонта – ведь рабочая техника ему нужна в минимально короткие сроки.
- Заказ с помощью авторизированных сервисных центров. Несмотря на свои обязанности не все авторизированные сервисные центры будут доставлять запчасти на технику, мотивируя отсутствием их на складе, снятием с производства и т.д. Некоторые фирмы (например, Indesit) ушли с рынка и запчасти не предоставляют.
- Печать на 3д принтере, создание корпусов и требуемых элементов. Вариант набирает популярность, среди ремонтников, он быстрее, зачастую выгоднее.

В работе будет рассмотрен ремонт мясорубки Braun, внутреннюю шестерню которой заказать в магазинах России невозможно по причине её простого отсутствия. Можно было заказать на зарубежных торговых площадках, но ждать пришлось бы около 3х месяцев. Со стоимостью доставки это составило 1210р.



Рисунок 1 – Мясорубка Braun

Было принято решение печати шестерни на 3D принтере. На разработку шестерни ушло 4,5ч и 12ч печати. По стоимости вышло на 500р. Но рассмотрим теперь плюсы: деталь мы получили в течении 5 рабочих дней по заказу, выдали сразу на руки, деталь полностью идентично как с завода. В ходе работы мы затратили денег на запчасти меньше, получили быстро более качественную деталь за счет применения качественного пластика и уложились в срок.

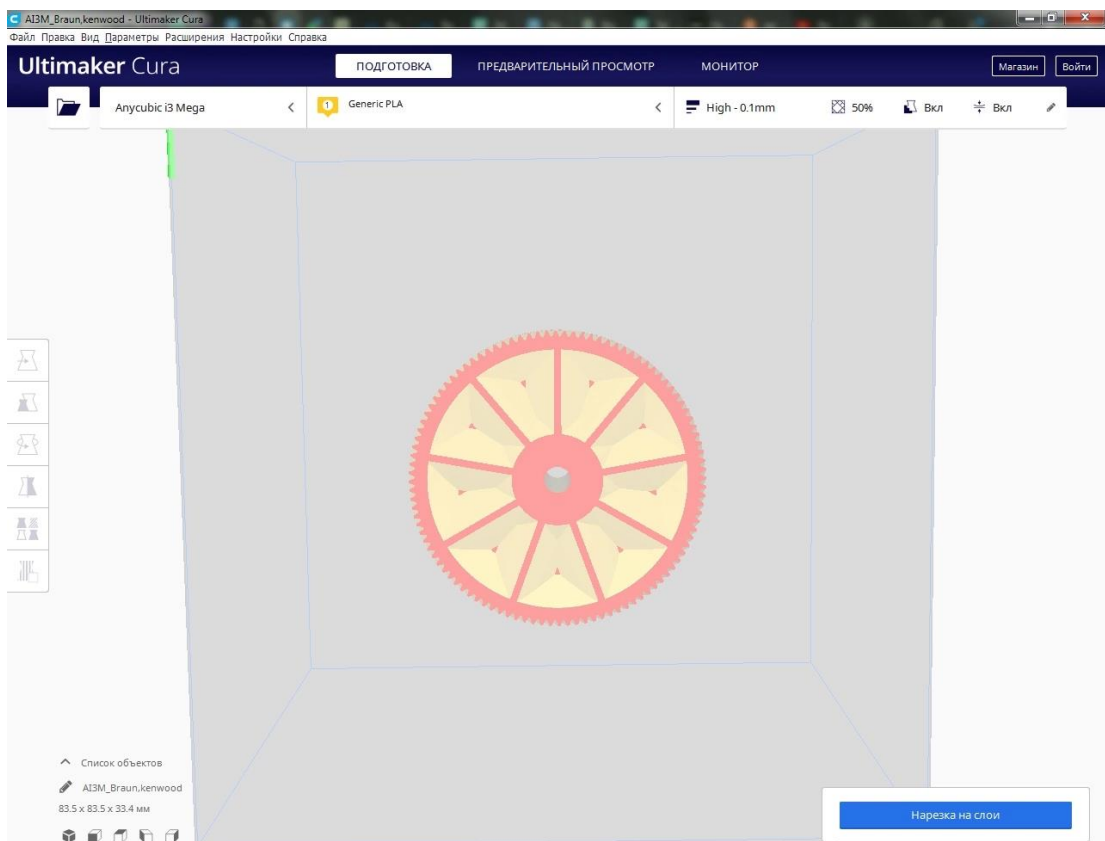


Рисунок 2 – Спроектированная шестерня для мясорубки (вид снизу)

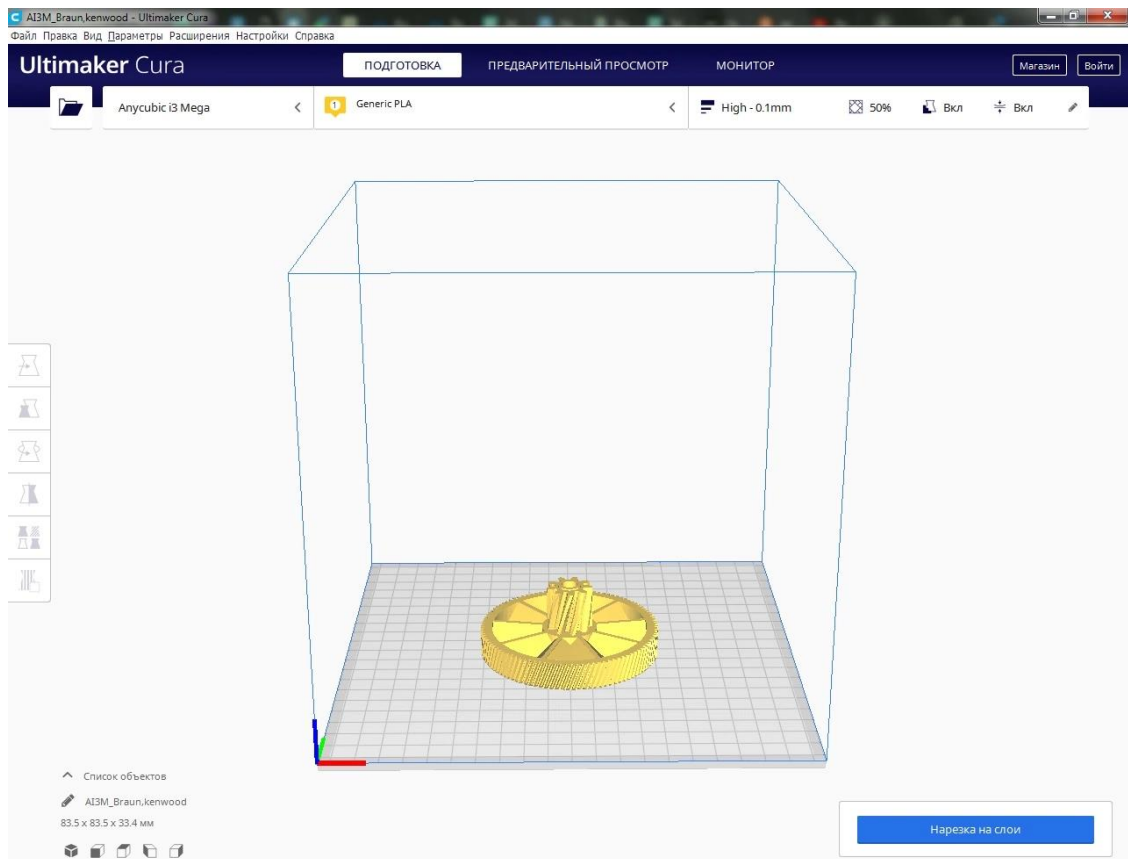


Рисунок 3 – Спроектированная шестерня для мясорубки (вид слева)

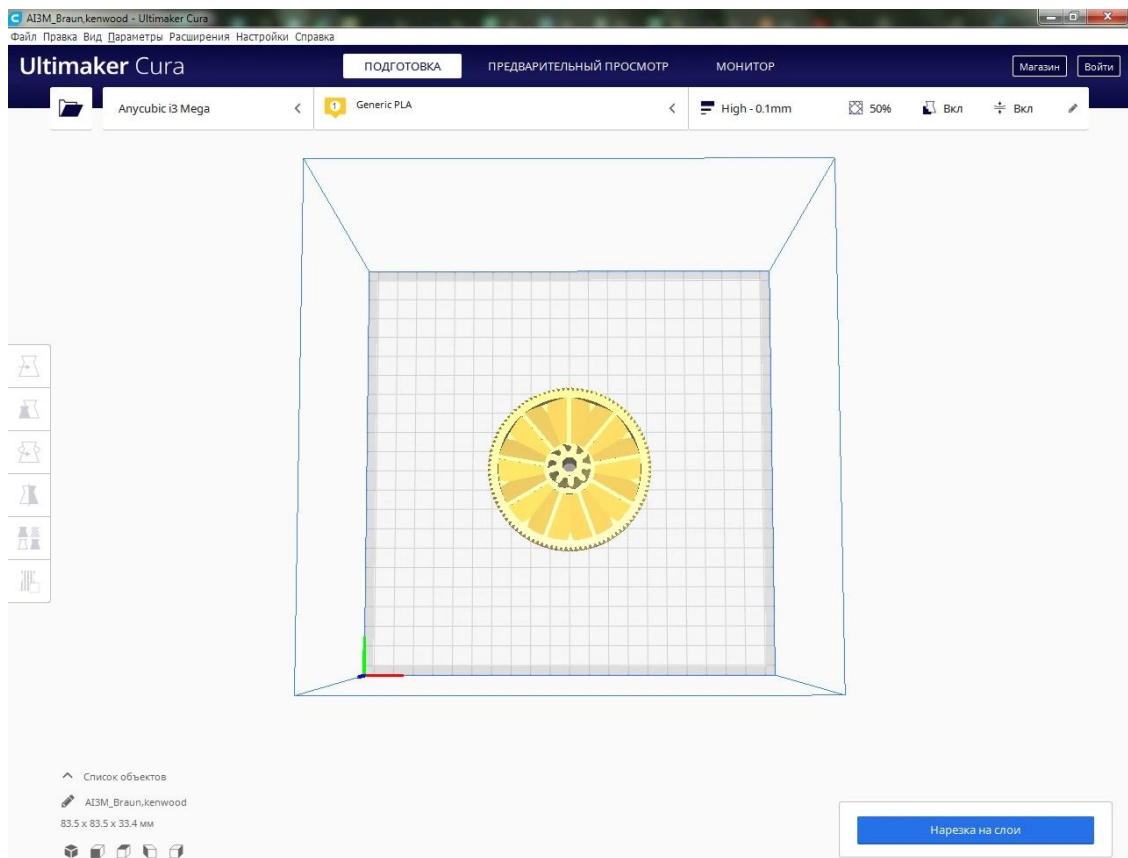


Рисунок 4 – Спроектированная шестерня для мясорубки (вид сверху)

В итоге получена деталь с относительно низкой себестоимостью, которую не производят в российских компаниях по причине нерентабельности. Заказ на иностранных сайтах не оправдан, т.к слишком большой срок доставки и цена.

#### **Список использованных источников:**

1. 3D печать. Коротко и максимально ясно (LittleTinyH Books), 2019 год
2. 3D-печать в малом бизнесе (Дмитрий Горьков), 2018 год
3. 3D-печать с нуля (Дмитрий Горьков), 2020 год
4. Leapfrog Creatr Single or Dual Extruder (Руководство)

## **СЕРВИСЫ ДЛЯ ОРГАНИЗАЦИЙ ВИДЕОКОНФЕРЕНЦИЙ**

*Бекмурзина Алина Аликовна  
ФКПОУ «ОГЭКИ» Минтруда России  
Белова Екатерина Сергеевна, преподаватель*

**Аннотация:** Современный компьютерный мир не знает пространственных границ. Друзья, коллеги, работники предприятий могут находиться на расстоянии сотен или даже тысяч километров друг от друга. Поэтому возникает вопрос, как лучше всего организовать полноценный контакт с ними. Личная встреча не всегда возможна. Часто электронная почта не способна решить все проблемы. Телефонные переговоры значительно ограничены в возможностях. Нужны инструменты, способные оперативно и качественно решать подобного рода задачи. Современные решения видеоконференцсвязи (ВКС), обладающие функциональностью систем высокого класса и доступностью простого телефона, существенно расширяют возможности телекоммуникаций.

### **Системы видеоконференцсвязи**

Видеоконференция — область информационной технологии, обеспечивающая одновременно двустороннюю передачу, обработку, преобразование и представление интерактивной информации на расстоянии в режиме реального времени с помощью аппаратно-программных средств вычислительной техники.

Взаимодействие в режиме видеоконференций также называют сеансом видеоконференцсвязи, [1].

Видеоконференцсвязь (сокращенное название ВКС) — это телекоммуникационная технология интерактивного взаимодействия двух и более удаленных абонентов, при которой между ними возможен обмен аудио- и видеоинформацией в реальном масштабе времени с учетом передачи управляющих данных.

Видеоконференция – это интерактивный инструмент, который включает в себя аудио, видео, компьютерные и коммуникационные технологии для осуществления связи удаленных территориально собеседников «лицом к лицу» в реальном времени, а также разделения всех типов информации, включая данные, звук, изображение, документы и т.п. В сущности, видеоконференция позволяет преодолеть барьер расстояния, который нас разделяет.

#### **Виды:**

– Видеоконференции 1-на-1

Здесь всё просто: участвуют два абонента, оба видят и слышат друг друга одновременно.

– Симметричные видеоконференции



Так называют сеанс видеоконференции, в котором участвуют более 2 человек и все участники видят и слышат друг друга одновременно.

– Видеоконференции с активацией по голосу

Эта видеоконференция предполагает следующий формат общения: все участники сеанса слышат и видят на своих экранах только выступающего докладчика, в то время, как он сам видит себя, либо предыдущего оратора.

Для общения в режиме видеоконференции абонент должен иметь терминальное устройство (кодек) видеоконференцсвязи, видеотелефон или иное средство вычислительной техники. Как правило, в комплекс устройств для видеоконференцсвязи входит:

- центральное устройство — кодек с видеокамерой и микрофоном, обеспечивающего кодирование/декодирование аудио- и видео- информации, захват и отображение контента;
- устройство отображения информации и воспроизведения звука.

В качестве кодека может использоваться персональный компьютер с программным обеспечением для видеоконференций.

Большую роль в видеоконференции играют каналы связи, то есть транспортная сеть передачи данных. Для подключения к каналам связи используются сетевые протоколы IP или ISDN, [2].

Существует два режима работы ВКС, которые позволяют проводить двусторонние (режим «точка-точка») и многосторонние (режим «многоточка») видеоконференции.

Для внедрения видеоконференцсвязи руководителю (лицу, принимающему решения) организации необходимо определить главную цель применения: проведение совещаний, подбор персонала, оперативность при принятии решений, осуществление контроля, дистанционное обучение, консультация врачей, проведение судебных заседаний, допрос свидетелей и так далее. При этом необходимо учитывать основные правила видеоконференцсвязи:

- гарантированная высокоскоростная услуга связи или выделенные каналы связи только для сеансов видеоконференций;
- стабильное и надёжное электропитание телекоммуникационного оборудования и видеоконференцсвязи;
- оптимальные шумо- и эхо- поглощающие особенности помещения в котором будет установлено оборудование видеоконференцсвязи;
- правильное расположение оборудования видеоконференцсвязи по отношению к световому фону помещения;
- корректная настройка телекоммуникационного оборудования и видеоконференцсвязи по обслуживанию качества услуги связи с приоритетизацией передачи данных;
- компетентный обслуживающий технический персонал;
- техническое сопровождение и подписка на обновление оборудования через сертифицированного производителем поставщика.

### **Преимущество видеоконференции**

Видеоконференция применяется как средство оперативного принятия решения в той или иной ситуации; при чрезвычайных ситуациях; для сокращения командировочных расходов в территориально распределенных организациях; повышения эффективности; проведения судебных процессов с дистанционным участием осужденных, а также как один из элементов технологий телемедицины и дистанционного обучения.

Во многих государственных и коммерческих организациях видеоконференция приносит большие результаты и максимальную эффективность, а именно:

1. снижает время на поездки и связанные с ними расходы;

2. ускоряет процессы принятия решений в чрезвычайных ситуациях;
3. сокращает время рассмотрения дел в судах общей юрисдикции;
4. увеличивает производительность труда;
5. решает кадровые вопросы и социально-экономические ситуации;
6. дает возможность принимать более обоснованные решения за счёт привлечения при необходимости дополнительных экспертов;
7. быстро и эффективно распределяет ресурсы, и так далее, [9].

#### **Преимущества:**

1. Простота в использовании.
2. Масштаб взаимодействия.
3. «Живая» картинка.
4. Экономия времени и простота организации
5. Реалистичность
6. Безопасность

### **Сравнение программ**

#### **1)Zoom**

Пожалуй, лучшее приложение для видеозвонков, и таким популярным оно стало после начала пандемии. Предусмотрен встроенный функционал для организации презентаций, вебинаров. В Зуме есть кому баг, кому отличная фишка – установка заикленного видео в чате. Некоторые участники конференций стали устанавливать его, чтобы не принимать участие в очень «увлекательных» встречах. Обойти баг можно, поддерживая постоянное живое общение со всеми участниками, [4, 6, 9].

#### **Плюсы:**

- самая популярная видеосвязь кроме скайпа;
- трендовая платформа 2021;
- огромные возможности даже в бесплатной версии.

#### **Минусы:**

- конфиденциальность данных не 100%.

**Личный опыт:** Когда звонишь по Zoom, ожидаешь, что на том конце слышат звонок. Когда присылаешь ссылку на конференцию в zoom — человек может ее не увидеть. Добавление в календарь в этом случае не панацея. Если участников много или есть те, кто на «созвоне» должен быть обязательно — приходится напоминать. Мне приходилось звонить человеку по сотовой связи, чтобы он открыл Zoom по ссылке. Приятно, когда в бесплатной версии тебе дарят дополнительные минуты.

**Вывод:** Zoom простой, быстрый, а еще популярный (особенно в США), поэтому рано или поздно он окажется у вас на устройстве, если вам нужно созваниваться с людьми вне вашей компании. Если же вы ищете решение «все в одном», обратите внимание на другие платформы.

#### **2)Skype**

Скайп – старая добрая программа для видеосвязи через интернет бесплатно, платные опции доступны, но обычно хватает и базовых возможностей. Версия для бизнеса относится к расширенным. Если планируете организовывать конференции на 99 человек, есть смысл воспользоваться именно ей. Скайп позволяет совершать звонки, показывать экран, [4, 6, 9].

#### **Плюсы:**

- известность, распространенность;
- углубленная интеграция с операционной системой Windows;
- платные и бесплатные версии.

#### **Минусы:**

- стабильность работы становится все хуже.

**Личный опыт:** Skype долго загружается, ненадежен в плане сохранения истории общения, передачи файлов. Узнать свой собственный логин — нетривиальная задача, другого человека найти тоже сложно. Если вам продиктовали имя, то обычно людей с таким именем находится несколько.

Большую часть этих проблем успешно разрешает Microsoft Teams.

**Вывод:** Если вы активно используете с коллегами Office 365, попробуйте Microsoft Teams, а про Skype забудьте.

### 3)Discord

Еще один способ, как сделать конференцию на компьютере – это установить Discord. У этого приложения много общего со Скайпом, но первично оно создавалось для общения геймеров друг с другом. Интегрирован широкий перечень функций. С началом карантина программа стала пользоваться спросом в бизнес-среде. Она бесплатная, но платные пакеты тоже есть – стоят они 9,99 долларов ежемесячно. Работа стабильная, особых требований к системным ресурсам нет. Качество звука отличное. Приглашения для участия приходят по ссылкам, [4, 6, 9].

#### Плюсы:

- стабильная работа;
- до 50 участников;
- быстрый поиск, возможность создания частных групп;
- быстрое действие на любых устройствах;
- многофункциональность;
- автоусиление голоса, подавление помех.

#### Минусы:

- геймерский функционал;
- частые слеты соединения.

**Личный опыт:** В App Store у приложения беда с отзывами, пользователи жалуются на постоянные сбои и зависания (в Google Play рейтинг сильно лучше). Я ни с чем подобным столкнуться не успел. Интересно, что вместо демонстрации экрана можно показывать фото и файлы с телефона или вебсайт, заходя на него прямо из приложения.

**Вывод:** Если у вас Android, возьмите бесплатную версию как альтернативу Zoom при условии, что вам не нужно много функций для демонстрации и модерирования встреч. Если вас интересует аналитика, и у вас большая компания, это хорошая замена.

### 4)Google Hangouts Meet

Не самый популярный, но все еще актуальный сервис Гугл, который позволяет делать видеозвонки с компьютера на телефон, и наоборот. Максимальное число участников – 10, картинка во время разговора может задерживаться на несколько секунд. Функционал базовый, зато программа бесплатная, [4, 6, 9].

#### Плюсы:

- есть в коробке гаджетов на ОС Андроид;
- понятный простой интерфейс;
- бесплатное применение.

#### Минусы:

- невысокая стабильность связи;
- максимум 10 участников.

**Личный опыт:** Я не испытывал проблем с качеством связи в Meet, кроме того, он имеет минималистичный и, на мой вкус, более симпатичный интерфейс, чем Zoom.

**Вывод:** Если вы уже платите за G Suite, попробуйте Meet. Если же вам нужны продвинутые функции модерирования, и вы будете использовать видеосвязь для демонстрации чего-либо, обратите внимание на другие сервисы.

#### И, в заключении:

Видеоконференция - это не просто видеотелефон. Видеоконференция - это компьютерная технология, которая позволяет людям видеть и слышать друг друга, обмениваться данными и совместно их обрабатывать в реальном режиме времени.

Все это осуществимо благодаря специализированным системам видеоконференцсвязи (ВКС).

Если учесть, что личная встреча позволяет достичь практически 100% понимания между собеседниками, то использование всевозможных мессенджеров в общении эффективно во взаимопонимании лишь на 25-30%, та же эффективность телефонного разговора – не более 43%, а вот видеозвонки и видеоконференции на 90% дают нужный эффект, [8].

Исследователи доказали, что в сравнении с обычным телефонным разговором, видеоконференция на 80% исключает возможность возникновения путаницы в диалогах.

Транспортные компании являются основными конкурентами тех, кто предлагает в качестве своей продукции IP-телефонию. Ведь именно личные встречи призывает заменить видеоконференция в тех случаях, когда такая замена возможна.

Для проведения сеансов видеоконференцсвязи необходимо выполнение двух важнейших условий:

- Вы должны иметь соответствующее оборудование видеоконференцсвязи;
- Вы должны иметь возможность соединиться с коллегой через любые каналы связи (в том числе и спутниковые), отвечающие требованиям видеоконференцсвязи

По разным источникам 80-85 % информации человек воспринимает зрительно, поэтому видеоконференцсвязь оказывает неоценимую помощь человеку в жизни. В связи с этим применение видеоконференций в управлении, медицине, дистанционном обучении, системах безопасности и многих других областях приносит огромную пользу.

Конечно, даже видеоконференции никогда не заменят личного общения, но они позволяют добиться принципиально нового уровня общения людей, подчас разделенных многими тысячами километров. Ведь согласно многочисленным исследованиям, на слух человек воспринимает всего лишь десятую часть информации (как, например, при телефонном разговоре). А в случае, когда есть возможность следить за жестикомикой и мимикой собеседника, КПД восприятия информации достигает 80-85 %.

Менеджеры компаний, использующие видеоконференции в повседневной жизни, утверждают, что системы видеоконференций резко сокращают временные и финансовые затраты фирмы на совещания, семинары, командировки их сотрудников и консультации.

#### **Список использованных источников**

1. Лукацкий А.В. Системы видеоконференций. Статья на сайте <http://www.infosec.ru>.
2. Фролов Б.А., Фролов Р.Л. Видеоконференции в IP-сетях. Статья на сайте <http://www.citforum.ru>.
3. Шукшина Е. Е. Система дистанционного образования с использованием Интернет–технологий: статья, Красноярский государственный университет, Красноярск, Россия 2018г.
4. <https://vc.ru/services/119494-13-servisov-dlya-videokonferenci>
5. <https://www.webmeetings.ru/tools/videoconferencing/>
6. <https://www.officemag.ru/info/guide/index.php?ID=13415222>
7. <https://mentamore.com/biznes/servisy-dlya-videokonferencii.html>
8. <https://infourok.ru/videokonferencii-v-obrazovatelnom-processe-5746992.html>
9. <https://www.sites.google.com/site/videoconfsys/home/celizadachi>

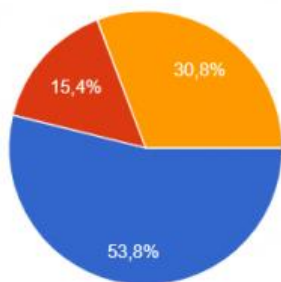
Анализ опроса студентов ФКПОУ «ОГЭКИ» Минтруда России 1-4 курсов

**Вопрос 1**

Выберите основные компоненты эффективности системы дистанционного обучения.



39 ответов



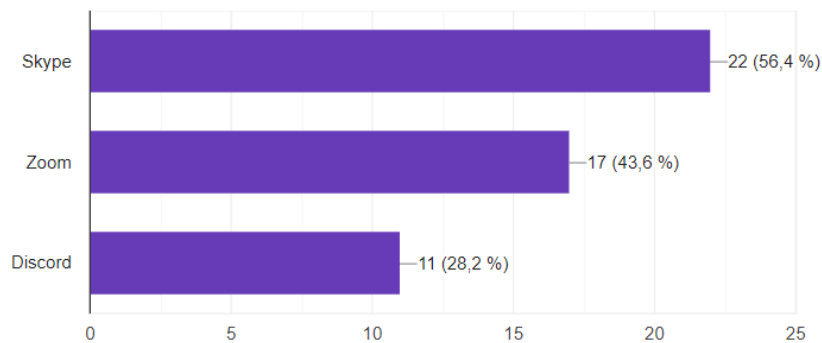
- техническое, программное, информационное, учебно-методическое, организационное, финансовое обеспечение;
- учебный центр, информационные ресурсы, средства методического и технического обеспечения, обучающиеся, тьюторы, консультанты;
- техническое обеспечение, преподаватели, обучающиеся;

**Вопрос 2**

Какая площадка видеоконференц связи вам нравится больше всего ?



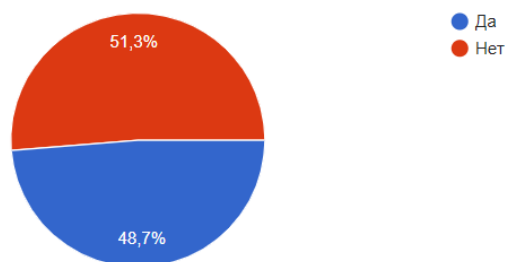
39 ответов



**Вопрос 3**

Вам нравится дистанционное обучение?

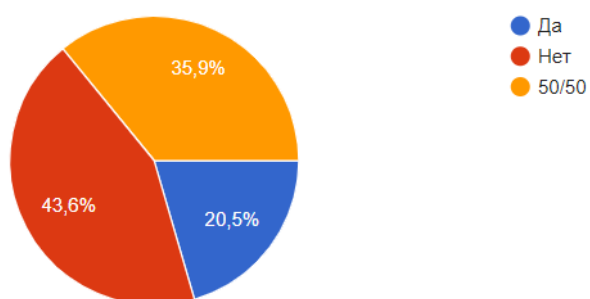
39 ответов



#### ***Вопрос 4***

Хотели бы вы остаться на дистанционном обучении?

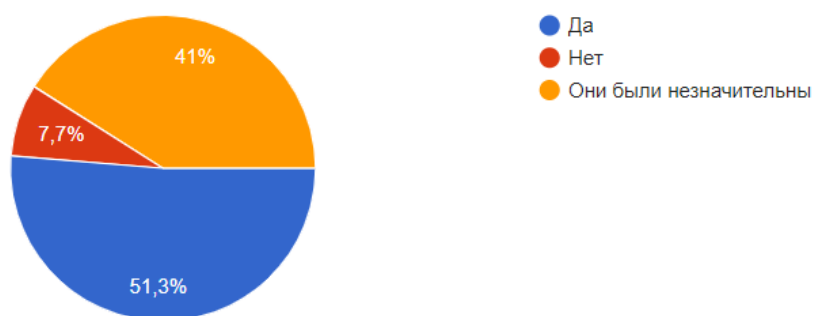
39 ответов



#### ***Вопрос 5***

Вы сталкивались с трудностями на дистанционном обучении?

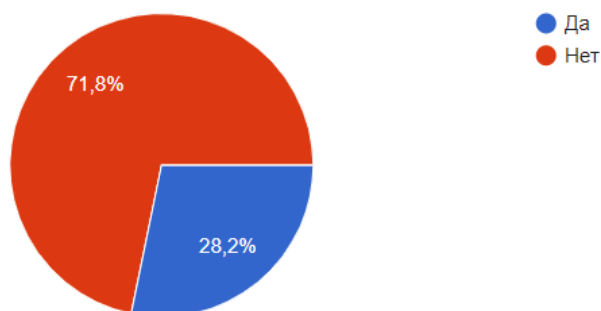
39 ответов



#### ***Вопрос 6***

Считаете ли вы, что дистанционное обучение может заменить очное?

39 ответов



По полученным данным можно сделать вывод, что многие из опрошенных студентов пользуются платформами для видеоконференцсвязи и имеют представление о том, что нужно для успешного дистанционного обучения. Но студенты пока не готовы дальше учиться и работать по такой схеме.

## QR КОДЫ ВОКРУГ НАС

*Кокарева Александра Олеговна  
ФКПОУ «СТИБ» Минтруда России  
Матвеева Надежда Георгиевна, преподаватель информатики*

Они встречаются в нашей жизни очень часто, при оплате квитанций. для получения информации, даже новейшие телефоны выпускаются уже со сканером QR - кода. Каждый человек встречая его в своей жизни знает, как им пользоваться, но никогда не задумывается что это и как оно устроено.

QR (от английского «быстрый отклик») — это двумерный тип штрих-кода (рис.1), который легко считывается цифровым устройством и хранит информацию в виде серии пикселей в квадратной сетке, которая внешне выглядит как черно-белый узор. QR-код, в отличие от штрих-кода, читается в двух направлениях — по горизонтали и по вертикали. Это позволяет хранить в нем больше данных. При сканировании QR-кода пользователь получает доступ к этим данным мгновенно. Код состоит из чёрных квадратов, расположенных в квадратной сетке и вмещает в себя три килобайта двоичного кода. В него можно поместить ссылку на сайт, номер телефона, географические координаты или текстовую информацию [1].



Рисунок 1. QR-код

Предшественником QR-кода был штрихкод. Штриховой код (штрихкод) — графическая информация, наносимая на поверхность, упаковки изделий, предоставляющая возможность считывания её техническими средствами — последовательность чёрных и белых полос. Штрихи наносились на пленку, а для считывания использовалась технология оптического считывания. Штрихи имели вид широких и узких полосок. Это стало прообразом для современных линейных штрих кодов, наносимых сейчас на упаковку почти всех товаров и продуктов. Первым товаром со штрих кодом стала жевательная резинка; она поступила в продажу в июне

1974 года. Штрих код позволяет закодировать 13...14 цифр. Этими цифрами обозначаются код страны изготовителя, код производителя и код товара. По этому коду можно искать товар в Интернете через поисковые системы. Они получили широкое распространение в мире, но их возможности сильно ограничены, и для кодирования большего объема информации придумали двухмерные штрих код. [7]

Первый QR-код появился в начале 90-х годов в Японии, его разработали в одном из подразделений Toyota. Сотрудники автоконцерна наклеивали QR-коды на запчасти автомобилей при транспортировке. Чтобы получить информацию о партии и принадлежности деталей, достаточно было отсканировать код специальным сканером. В отличие от обычного штрихкода, QR-код — двухмерный, он содержит больше информации, объемом до 7 089 цифр или 4 296 букв — это около четырех страниц текста [2].

В 2000 году Международная организация по стандартизации внесла QR-код в список одобренных стандартов кодирования информации, где он стал экспериментальным способом оплаты, но поначалу операция занимала слишком много времени — сканер считывал информацию целых 20 секунд. В 2003 году китайская компания, изобрела быстрый механизм считывания QR-кодов. Позже она же выпустила первый портативный сканер, а технология стала по-настоящему популярной

В Китае технология стала популярной вместе с появлением мессенджера WeChat. И очень сильно прижился QR-код, для обмена информацией таких как номер телефона, страница в социальных сетях, газеты, журналы, рекламы, плакаты. Внутренняя платежная система соцсети WeChat Pay — один из самых популярных финансовых инструментов в Китае, где все транзакции происходят с помощью QR. Через систему пользователи совершают платежи и переводы на сумму около 2 млрд долларов ежегодно. Это примерно треть всех мобильных платежей в стране. Не только в Китае это стало популярным, но и в других странах, как раз в этот период начался массовый рост производства планшетов, телефонов, и другой техники.

В США и Европе Starbucks одной из первых заметила перспективную технологию и разработала на её основе программу лояльности. Компания разместила QR-коды в журналах, буклетах и на рекламных щитах. Пользователи могли отсканировать код и получить ссылку. Она вела на сайт компании, где можно было узнать о ближайших кофейнях, оценить кофе, получить скидку или посмотреть видео о типах обжарки кофе.

В России QR-коды переживают вторую волну популярности. Сначала они пришлись по душе только маркетологам и тем, кто увлекается технологиями. Их начали применять в рекламных кампаниях или в партизанском маркетинге, чтобы оставлять скрытые сообщения. Изначально не поняв и разобравшись с QR кодом в России было сделано ошибочное мнение, что это не удобно, но после поняв ,что QR-код может хранить в себе очень много информации, стали использовать его в целях оплаты квитанций (коммунальных услуг ,штрафов и многое другое ).Но больше всего QR-код полюбился в музеях России ,т.к. можно в нем хранить всю информацию ,об экспонатах .Посетителям не нужно больше покупать бумажную версию гида по музею и искать информацию про картины или экспонаты , по простором интернета .Все это заменил маленький квадратик QR-код .И в дальнейшем все больше и больше становится востребованным этот метод оплаты (в основном в банках, магазинах) [6].

Узор QR-кода хранит зашифрованную последовательность данных в двоичном формате (1 и 0) в виде матрицы (рис.2). Каждой отдельной ячейке сетки присваивается значение в зависимости от цвета (черный или белый). Затем ячейки группируются в



более крупные узоры. Ключи закодированных данных содержат дубликаты, поэтому при повреждении поверхности QR-кода до определенных масштабов его можно считать.

QR-код не сканируется, что делать? Именно такой вопрос следовало бы задать тем, кто экспериментирует с изображением, например, увеличивает его, делая размытым. Это и в самом деле вариант, как сделать, чтобы QR-код не читался, поскольку считывать его можно лишь специальным устройством, а оно может и не распознать QR-код размытый. Черно-белое - именно такое сочетание цветов нужно для QR кода, но можно, конечно, цвет менять, только вот его совсем уже интересные оттенки могут не дать правильно QR-код считать. Ну, и размер тоже важен, слишком маленький может и не считаться так, чтобы можно было сделать расшифровку [4].



Рисунок 2- Описание полей QR –кода

В смартфонах последних лет выпуска присутствует встроенный считыватель QR-кода, который расположен в камере. Для считывания достаточно навести камеру на код под любым углом. Даже если на телефоне нет встроенного считывателя QR-кода, его можно скачать в магазинах приложений. В этом случае для считывания потребуется открыть приложение.

QR-коды можно считать и на компьютере с помощью одной из специализированных программ. Например, bcTester умеет сканировать разные типы QR-кодов, а bcWebCam работает напрямую с веб-камерой. Еще одной из таких программ, которая создает, сканирует и расшифровывает QR-коды является QuickMark for PC. Стандартная версия этой программы бесплатная и позволяет декодировать помощью веб-камеры или захвата с экрана. Доступно создание и сохранение своих, собственных штрих-кодов - Quick Code и QR-кодов в виде ссылки на сайт, текстовой информации, визитки, телефона, смс и e-mail.

#### ПОЛОЖИТЕЛЬНЫЕ СТОРОНЫ QR-КОДА:

– Данный код относится к открытым технологиям, то есть технологиям, доступным каждому, потому он и получил быстрое распространение, особенно в среде маркетинга;

– По сравнению с обычным штрих кодом, QR-код вмещает в себя намного больше информации и более устойчив к повреждениям графического рисунка (например, часть графического рисунка можно закрасить или вовсе удалить, код останется читаемым);

#### НЕДОСТАТКИ QR-КОДА:

– При использовании QR-кода необходимо быть уверенным, что адресат сможет его прочесть. Для чтения такого кода необходимы гаджеты, умеющие распознавать изображение QR-кода и расшифровывать его;

– Вмещает в себя относительно мало информации, например, закодировать целую книгу в один стандартный QR-код не представляется возможным. [3]

–Удобное чтение URL-адреса может играть на руку преступникам, которые хотят направить свою жертву на поддельный веб-сайт или другое место в сети, используемое для перехвата личных данных. Бывали даже ситуации, когда мошенники размещали наклейки с QR-кодами на банкоматах, то есть в общественных местах.

#### ЧАЩЕ ВСЕГО В QR-КОДЕ ШИФРУЮТ:

- Адрес в интернете. Считав такой код, вы сразу можете на своем смартфоне перейти на сайт, не набирая его руками.
- Визитка. В такой код зашифрованы контактные данные о человеке в специальном формате. Смартфон, считывая такой штрих код, предложит создать новый контакт. Пользователю при этом не нужно будет ничего вводить своими руками.
- СМС сообщение. В такой код зашифрован номер телефона, на который нужно отправить СМС, и сам текст сообщения. Считав такой код на смартфоне, можно одной кнопкой отправить нужное СМС, не набивая его руками.
- Email сообщение. В такой код зашифрован адрес электронной почты и сам текст сообщения.
- Любой текст, стихи, поздравительные открытки.
- Адресный кулон (одевается на ошейник животному, которое может потеряться).
- Меню в ресторане.

Чтобы минимизировать риски, стоит запомнить **несколько простых правил:**

- Прежде чем сканировать QR-код, убедитесь, что он не перекрыт другим кодом.
- Когда вы открываете магазин приложений или веб-сайт в браузере, убедитесь, что QR-код привёл вас туда, где вы хотите быть.
- Если вы собираетесь установить приложение, убедитесь, что оно разработано компанией, о которой вы читаете, и проверьте её рейтинги.
- Если код ведёт на веб-сайт, проверьте полный URL, иначе вы можете стать жертвой фишинга.
- Будьте особенно осторожны, если веб-сайт или приложение просит вас ввести ваши личные данные, логины и пароли.
- Установите приложение для защиты смартфона. [5]

Напоследок, **пять популярных вопросов о двухмерных кодах:**

1. Сколько информации можно поместить в код? Это зависит от возможностей онлайн-генератора, которым вы пользуетесь. Одни предлагают стандартный набор из адреса сайта и контактов, тогда как другие позволяют разместить множество ссылок.
2. Как изменить данные? Сгенерировать не только статический, но и динамический вариант QR-кода. Это позволит в любой момент обновить ссылки.
3. Создание кодов бесплатно? Не всегда. Обычно свободно можно получить только статическую версию, а за динамическую нужно платить — обращайтесь внимание на условия онлайн-сервисов.
4. Может ли истечь срок действия? Статические коды не имеют срока действия, а динамические могут перестать работать, если онлайн-генератор прекратит оказывать услуги.
5. Как кастомизировать картинку? Чтобы привлечь внимание пользователей, брендуйте QR-код: добавьте логотип, выберите цвета и форму, соответствующие фирменному стилю. Если у вас еще нет айдентики, разработайте ее в несколько кликов при помощи бесплатного онлайн-сервиса Logaster.

### Список литературы:

1. Визитки, реклама, платежи: как возникли и для чего используются QR-код. // Сетевой журнал РБК. 10.11.2021 URL: <https://trends.rbc.ru/trends/industry/6189517c9a79475deb5dbf9a> (дата обращения: 30.11.2022).
2. История появления QR-кодов. 12.09.2019. URL: <https://etiketki24.ru/interesnye-stati/proiskhozhdenie-razvitie-i-ispolzovanie-qr-koda> (дата обращения: 30.11.2022).
3. Ковалёв, А. И. QR-коды, их свойства и применение / А. И. Ковалёв. — Текст: непосредственный // Молодой ученый. — 2016. — № 10 (114). — С. 56-59. — URL: <https://moluch.ru/archive/114/29398/> (дата обращения: 30.11.2022).
4. Сайт LamerKomp.ru /QR-код и штрих-код. 15.08.2020. URL: [https://lamerkomp.ru/publ/kompjuter/qr\\_code/1-1-0-39](https://lamerkomp.ru/publ/kompjuter/qr_code/1-1-0-39) (дата обращения: 30.11.2022).
5. Что нужно знать о QR-коде – как они используются и для чего нужны. // Блог /Технологии. 21.10.2021. URL: [https://webznam.ru/blog/chto\\_nuzhno\\_znat\\_o\\_qr\\_kode/2021-10-25-1857](https://webznam.ru/blog/chto_nuzhno_znat_o_qr_kode/2021-10-25-1857) (дата обращения: 30.11.2022).
6. Эволюция QR-код. Что хранится в чёрных квадратах и как их используют бренды 20.12.2019. URL: <https://vc.ru/marketing/98263-evolyuciya-qr-koda.yjdktyj> (дата обращения: 30.11.2022).
7. Электронная Библиотека/Журнал Юный техник, 2013 № 04 URL: <https://www.litmir.me/br/?b=583386&p=16> (дата обращения 30.11.2022)

## РАЗРАБОТКА ГОЛОСОВОГО ПОМОЩНИКА С ПРИМЕНЕНИЕМ ЯЗЫКА ПРОГРАММИРОВАНИЯ PYTHON

*Котельников Сергей Сергеевич,  
ФКПОУ «НГГТКИ» Минтруда России, 3 курс, гр. ИСП-20  
Грисман Светлана Сергеевна, преподаватель*

**Аннотация.** В данной статье рассмотрен вопрос развития искусственного интеллекта. Проанализированы возможности практического применения искусственного интеллекта в прикладных задачах. Произведен анализ возможностей языка Python для реализации систем с применением функций искусственного интеллекта. Произведена реализация голосового помощника на языке программирования python.

Развитие ИИ началось в середине XX века. Сам термин ввел американский информатик Джон Маккарти в 1955 году. Летом 1956-го он создал в Дартмутском колледже научную конференцию, которая сегодня считается отправной точкой развития ИИ как раздела науки. Интересу к ИИ со стороны научного сообщества помогло стремительное развитие вычислительной техники, алгоритмов и языков программирования, а также желание выяснить, насколько машины способны приблизиться к возможностям человеческого разума и могут ли его превзойти.

В массовой культуре распространен миф о том, что развитие ИИ может лишить человечество работы и даже поставить под угрозу существование. Однако из за ряда неразрешимых фундаментальных проблем реальность такого будущего сомнительна.

Первая проблема отсутствие в действиях любого ИИ человеческой осмысленности. Например в онлайн переводчик заложены результаты статического анализа перевода слов и построения предложений. Какой бы огромной ни была выборка данных и насколько бы совершенными ни были алгоритмы, любая

статистическая система всегда будет допускать, ошибки по сколько она не осознает точный смысл фразы

Во вторых, ИИ недостает самостоятельности. Любая система статистического машинного обучения требует огромных объемов специальным образом подготовленных данных.

Внедрение искусственного интеллекта в интернет началось еще в 2011 году. В то время была запущена работа над проектом Google Brain. Результатом стало внедрение в поисковую систему Google новых опций, Контролируемых ИИ:

Переводчик Google с прямым переводом достаточно нажать на кнопку с микрофоном, чтобы активировалась функция голосового ввода. Система распознает речь и быстро переводит слова или фразы на нужный язык

Голосовой помощник Google Assistant можно запускать поиск информации, давать поручения, планировать дела.

Алгоритм Google Photo научился распознавать тексты и объекты на изображениях. Благодаря этой функции можно легко находить фотографии с помощью описания предметов или людей, которые присутствуют на изображении.

Международная платежная система MasterCard внедрила дополнительный сервис Decision Intelligence. Компания отмечает, что убытки из за ошибок системы безопасности превышают потери от мошенничества. Внедрение Decision intelligence повысило уровень своей прибыли компании. Главные функции этой технологии повышение точности подтверждения финансовых операций и снижение вероятности ложных отклонений при переводе средств.

За счет внедрения ИИ в будущем система образования будет развиваться в двух направлениях. Первое из них адаптивное. Его главная задача состоит в том, чтобы решить проблему разной успеваемости у учеников. ИИ будет анализировать результаты обучающихся и на их основе адаптировать порядок курсов, дополнительно информируя преподавателей о степени усвоения материала.

Первичная концепция языка Python была придумана в конце 1980-х Гвидо Ван Россумом, сотрудником голландского Национального исследовательского института математики и информатики (иначе — CWI, Центр математики и информатики).

Изначально язык был разработан как ответ языку программирования ABC, родиной которого также являются Нидерланды. Среди главных особенностей Python в сравнении с языком ABC можно выделить то, что Python мог обрабатывать исключения и был в основном ориентирован на операционную систему Amoeba.

Python не имеет никакого отношения к змеям. Язык получил свое название в честь британского телешоу — «Летающий цирк Монти Пайтона».

На начальном этапе разработки голосового помощника на основе языка Python был составлен план по разработке функционирования разрабатываемого программного продукта.

В процессе реализации понадобилось применение стронных библиотек: “Silero”, ”torch”, ”Playsound”.

С помощью этих основных библиотек можно воссоздать своего голосового помощника. Для начала для создания, нужно изучить основы языка, затем отобрать готовые библиотеки для голосового помощника которые выделены сверху, то есть основой библиотек будет являться Silero и Torch в основном.

В результате проделанной работы был реализован программный продукт, имеющий в своем функционале способности голосового помощника и возможности искусственного интеллекта. Для реализации создания голосового помощника было изучено большое количество документации по языку и библиотекам Python. Было выявлено множество проблем с подключением библиотек для использования дополнительных функций в разработке голосового помощника на языке Python, в программе PyCharm.

По итогу выполнения работы голосового помощника на языке Python, данный помощник имеет большой функционал такой как рассмешить пользователя или же взаимодействия с пользователем, то есть использования его как помощника в учебе или лучшего друга на момент одиночества.

В дальнейшем планируется доработка и расширение функционала голосового помощника.

#### **Список использованных источников:**

1. Чернышев, С. А. Основы программирования на Python : учебное пособие для вузов / С. А. Чернышев. — Москва : Издательство Юрайт, 2022. — 286 с. — (Высшее образование). — ISBN 978-5-534-14350-8. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/496893>

2. Гниденко, И. Г. Технологии и методы программирования : учебное пособие для вузов / И. Г. Гниденко, Ф. Ф. Павлов, Д. Ю. Федоров. — Москва : Издательство Юрайт, 2022. — 235 с. — (Высшее образование). — ISBN 978-5-534-02816-4. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/489920>

### **РАЗРАБОТКА БАЗЫ ДАННЫХ И WEB-ПРИЛОЖЕНИЯ ДЛЯ ОРГАНИЗАЦИИ ONLINE КОНКУРСОВ**

*Пручковский Николай Николаевич, Харламов Илья Константинович,  
Федорченко Дарья Андреевна  
ФКПОУ «НТТИ» Минтруда России  
Полякова Ольга Владимировна, преподаватель*

**Аннотация.** В статье рассматриваются теоретические и практические аспекты технологий разработки современных веб-приложений, в том числе новые возможности клиентской разработки на основе HTML5, CSS3, JavaScript API, СУБД MySQL, PHP, Bootstrap 5, OpenServer, AJAX и jQuery, а также инструментарий СпринтХост для разработки серверных приложений.

Реализация данного проекта состоялась в рамках дипломного проектирования по созданию системы для проведения заочного этапа конкурса ИНВА-ПРОФИ, проводимого ежегодно Минтруда России, Проект выполнялся студентами НТТИ Рябышенковым Виктором и Власовым Дмитрием. Мы продолжаем эту работу для расширения функциональности данного проекта с целью создания программного продукта универсального характера для организации практически любого онлайн конкурса.

Учредителем конкурса является Министерство труда и социальной защиты Российской Федерации. Конкурс направлен на выявление и поощрение талантливых и инициативных педагогических работников образовательно-реабилитационных организаций для инвалидов и лиц с ОВЗ.

К участию в конкурсе допускаются работники в двух номинациях:

а) “Педагог-предметник”;

б) “Социально-психологическое сопровождение”;

К участию в номинации «Педагог-предметник» приглашаются преподаватели и мастера производственного обучения образовательно-реабилитационных организаций для инвалидов и лиц с ОВЗ (профессиональных образовательных организаций (колледжей, техникумов, училищ-интернатов), учреждений социального обслуживания (реабилитационных центров, комплексов).

К участию в номинации «Социально-педагогическое сопровождение» приглашаются социальные педагоги, педагоги-психологи, педагоги-дефектологи, педагоги-логопеды, педагоги-организаторы, воспитатели, тьюторы образовательно-реабилитационных организаций для инвалидов и лиц с ОВЗ (профессиональных образовательных организаций (колледжей, техникумов, училищ-интернатов), учреждений социального обслуживания (реабилитационных центров, комплексов). При наличии большого количества поданных заявок номинация делится на 2 под номинации: «Социально-психологическое сопровождение образовательного процесса» (для организаций, имеющих лицензию на осуществление образовательной деятельности) и «Социально-психологическая реабилитация» (для специалистов реабилитационной сферы)[1].

Конкурс ИНВА ПРОФИ проводится в два этапа. В конкурсе участвуют представители субъектов Российской Федерации и образовательные учреждения Минтруда России.

Региональный этап проводится органами исполнительной власти субъектов Российской Федерации. Федеральный этап состоит из двух частей: заочной и очной.

Данный проект затрагивает только заочный этап.

На заочном этапе оценивается портфолио участника и видеоролик.

Пользователями системы являются Организатор, Участники и Эксперты.

Все данные, обеспечивающие работу данного сайта, хранятся в базе данных, концептуальная модель которой представлена в соответствии с рисунком 1.

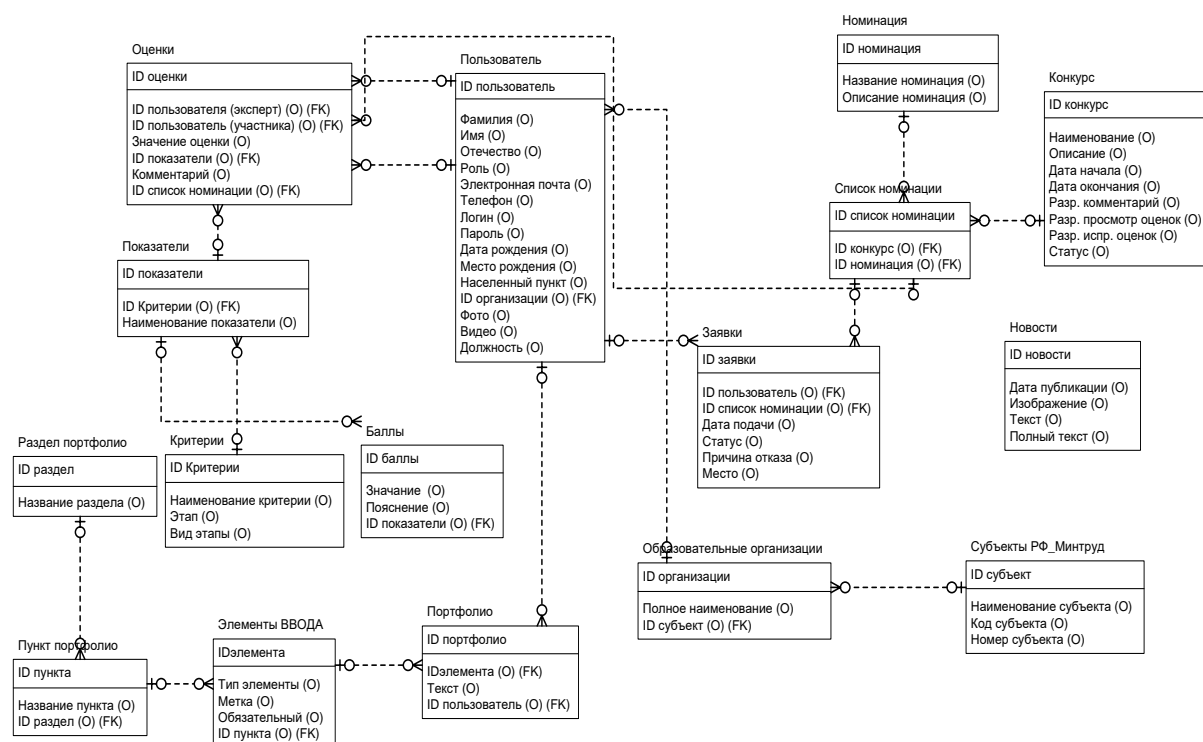


Рисунок 1 – Концептуальная модель

Участник регистрируется на сайте, вводит данные своего профиля, выбирает номинацию для участия, выбирает субъект РФ и образовательную организацию, которую он представляет, а затем отправляет заявку на сайт.

Организатор просматривает поданные заявки, может принять заявку или отказать в приеме по каким-либо причинам. Результат приема заявки отправляется Участнику на указанный им адрес электронной почты.

На карте, расположенной на главной странице сайта, при этом отображаются регионы, подавшие заявки (выделяются цветом).

Участник, заявка которого принята Организатором заполняет данные своего профессионального портфолио, загружает видео, и может затем просматривать результаты оценки его материалов экспертами.

ФИО	Роль	Номинация	Конкурс	Субъект	Статус	Принять	Отказать	Подробно
Арапова Елизавета Александровна	Участник	Педагог-предметник	ИнваПрофи-2022	Ростовская область	принято		✘	📄
Назар Ирина Юрьевна	Участник	Педагог-предметник	ИнваПрофи-2022	Ленинградская область	принято		✘	📄
Ключникова Юлия Александровна	Участник	Педагог-предметник	ИнваПрофи-2022	Краснодарский край	принято		✘	📄
Ашитко Елена Леонидовна	Участник	Педагог-предметник	ИнваПрофи-2022	Калужская область	принято		✘	📄
Чехонадских Анастасия Николаевна	Участник	Педагог-предметник	ИнваПрофи-2022	Белгородская область	принято		✘	📄
Воронина Ирина Ивановна	Участник	Педагог-предметник	ИнваПрофи-2022	Белгородская область	принято		✘	📄
Шульженко Наталья Владимировна	Участник	Педагог-предметник	ИнваПрофи-2022	Удмуртская Республика	принято		✘	📄
Ельцова Ольга Евгеньевна	Участник	Педагог-предметник	ИнваПрофи-2022	Мурманская область	принято		✘	📄
Алексеева Саргылана Егоровна	Участник	Педагог-предметник	ИнваПрофи-2022	Республика Саха	принято		✘	📄

Рисунок 2 – Страница Заявки

При заполнении данных портфолио Участник может загрузить определенное количество файлов, необходимых для иллюстрации его достижений.

Разработанное веб-приложение дает возможность участнику после заполнения портфолио на сайте и распечатать его при необходимости.

Эксперты также регистрируются на сайте, выбирая определённую номинацию, и после подтверждения регистрации Организатором получают доступ к материалам Участников. Они оценивают участников (портфолио и видео) по заданным в системе критериям и могут просматривать результаты оценки.

Для удобства работы Эксперта предусмотрена возможность просмотра портфолио участника и его видеоролика во время выставления оценки.

По окончании оценивания система формирует рейтинг участников по каждой номинации и позволяет распечатать соответствующий протокол.

Для того, чтобы сделать данное программное решение более гибким и масштабируемым была поставлена задача обеспечить возможность изменения пунктов и разделов портфолио. Портфолио состоит из разделов, а в каждом разделе несколько пунктов.

В каждом пункте вводятся различные данные, например, краткий текст, многострочный текст, загрузка файлов, дата.

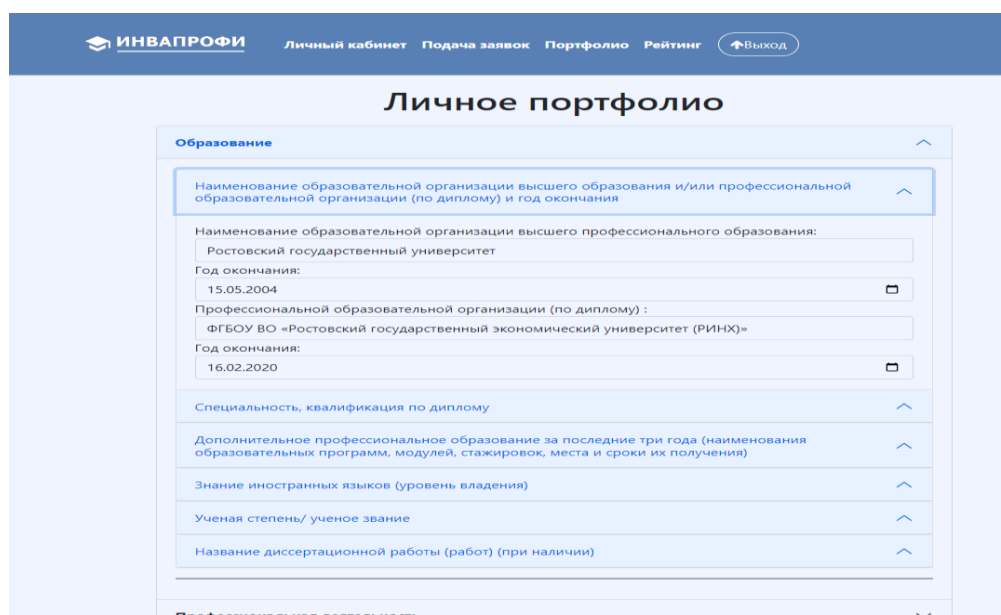


Рисунок 3 – Заполнение портфолио

При изменении пунктов и разделов портфолио должен измениться интерфейс ввода данных участником и выбора элементов для ввода данных в портфолио участника.

В соответствии с настройками, которые сделал Организатор нужно на форме показать различные элементы ввода, то есть нужно обеспечить гибкий интерфейс ввода данных пользователем в зависимости от того, какие данные и какого типа нужно вводить для заполнения данного пункта портфолио. Эта задача была успешно решена.

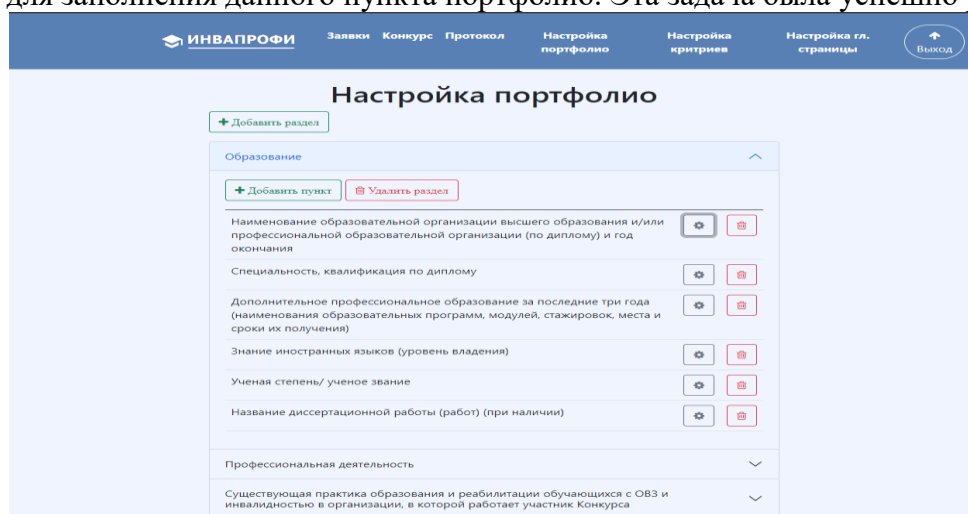


Рисунок 4 – Страница Настройка портфолио

Кроме того, обеспечена возможность для Организатора изменять критерии оценки показателей конкурса, а также добавлять новые.

При удалении какого-либо критерия он объявляется недействующим и не появляется в списке критериев оценки у Эксперта.

Таким образом, Организатор **ИМЕЕТ** возможность определять структуру портфолио участника – его разделы и отдельные пункты с указанием вводимых при этом данных, определять критерии оценки.



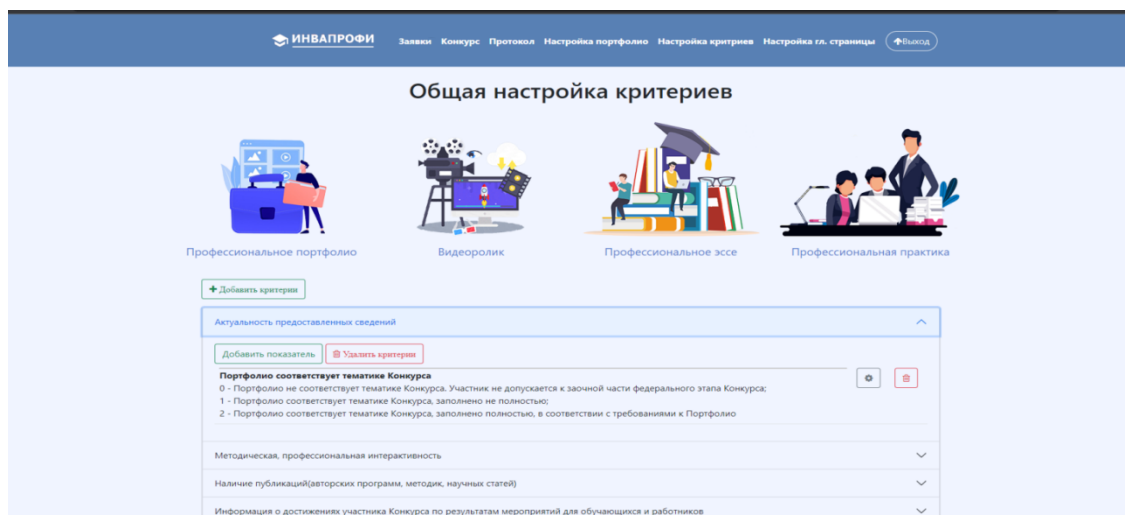


Рисунок 5 – Страница Общая настройка критериев

Эксперты оценивают «Профессиональное портфолио» и «Видеоролик», для каждого вида оценки существует множество критериев. В каждом критерии есть несколько показателей. В каждом показателе есть несколько вариантов выставления баллов с пояснениями. Например 0 – не соответствует тематике Конкурса. Все эти критерии, показатели, баллы могут, при необходимости, изменяться организатором.

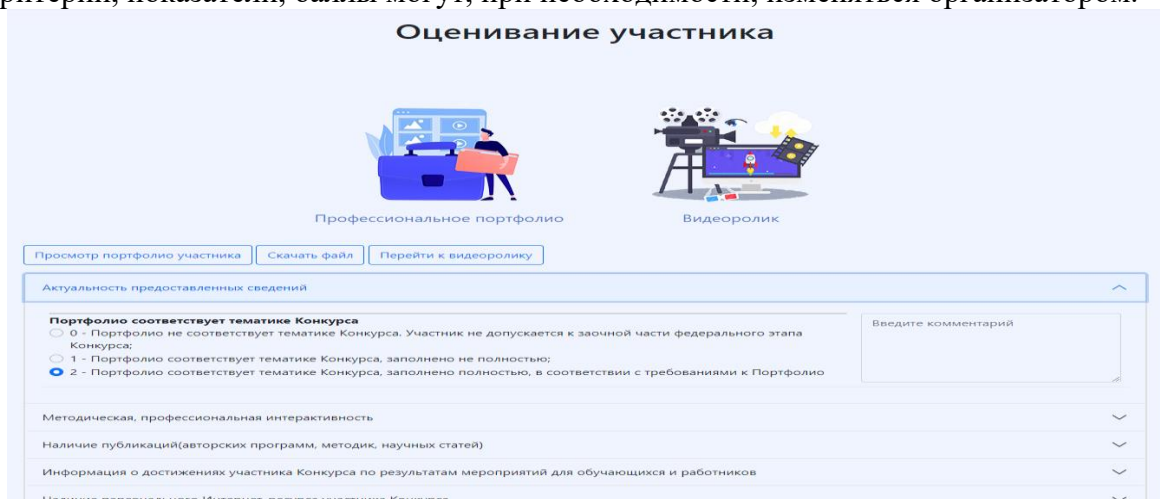


Рисунок 6 – Страница оценивание участника

В настоящее время в конкурсе две номинации, но могут добавиться новые, поэтому такая возможность также предусмотрена.

На главной странице Организатор может размещать новости, результаты проведенных конкурсов.

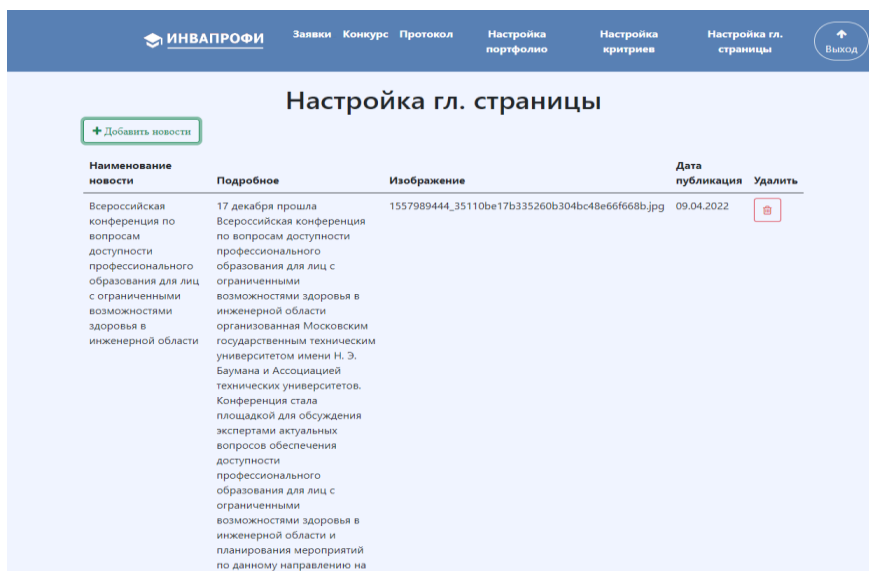


Рисунок 7 – Страница Настройка гл. страницы

Как Участник, так и Эксперт имеет возможность просмотра рейтинга всех участников после выставления оценок экспертами, а также изменения данных в своем личном кабинете.

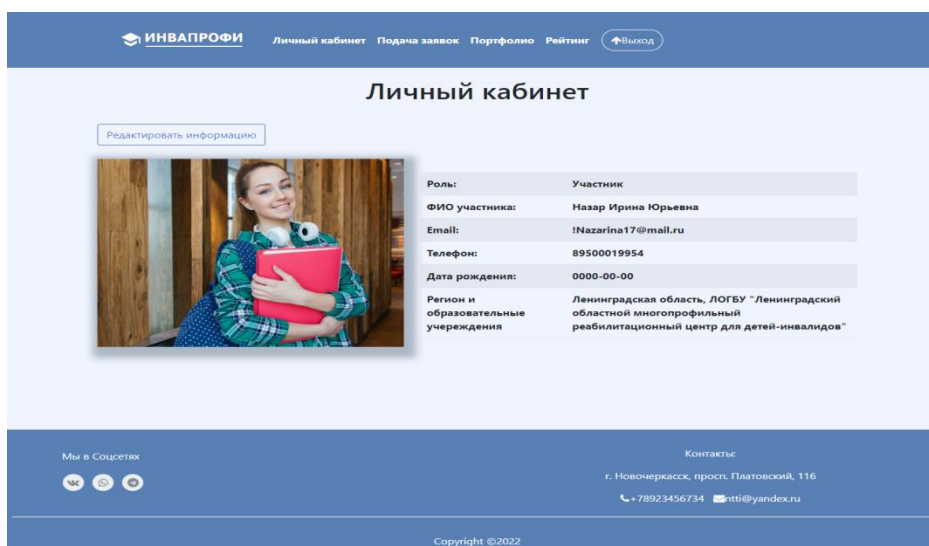


Рисунок 8 – Личный кабинет эксперта

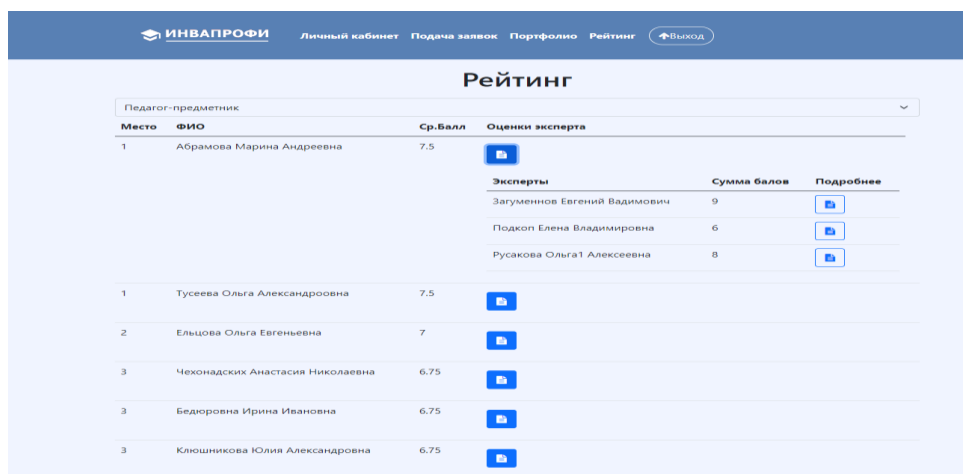


Рисунок 9 - Страница рейтинга

Общие критерии	Критерия	Показатель	Пояснение	Балл	Комментарий
Профессиональное портфолио	Актуальность предоставленных сведений	Портфолио соответствует тематике Конкурса	Портфолио не соответствует тематике Конкурса. Участник не допускается к заочной части федерального этапа Конкурса;	2	
	Методическая, профессиональная интерактивность	Взаимодействие участника Конкурса с другими субъектами образовательных отношений внутри организации.	Участник взаимодействует с другими субъектами образовательно-реабилитационных отношений только по вопросам реализации собственных профессиональных обя	2	
		Участие конкурсанта во внешнем межведомственном и сетевом взаимодействии	Участник не включен во внешнее взаимодействие	5	

Рисунок 10 - Детальный рейтинг

Структура сайта представлена в соответствии с рисунком 11.

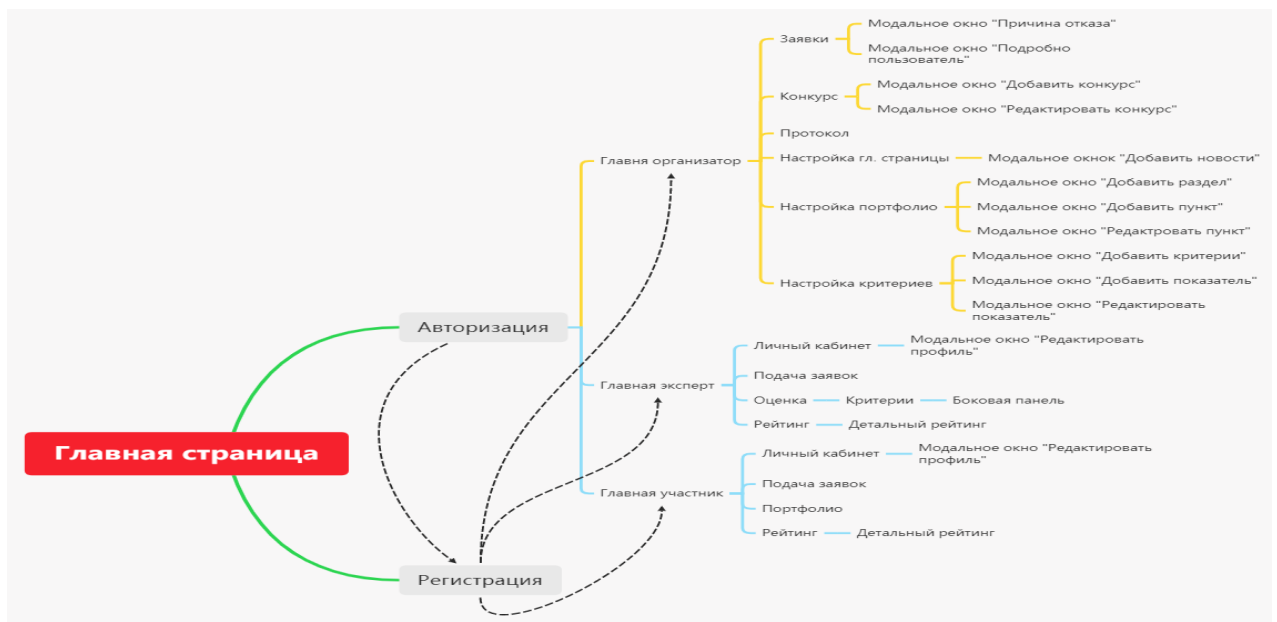


Рисунок 11 - Структура сайта

При разработке использованы следующие средства и языки программирования.

OpenServer — это портативная программная среда, созданная специально для веб-разработчиков с учётом их рекомендаций и пожеланий. Для отладки скриптов в различном окружении Open Server предлагает на выбор сразу два вида HTTP серверов, различные версии PHP и СУБД модулей, а так же возможность быстрого переключения между ним. В данном проекте использовался Apache HTTP-сервер — свободный веб-сервер. Основными достоинствами Apache считаются надёжность и гибкость конфигурации.

Тип используемой СУБД: MySQL "клиент-серверная".

PHP - скриптовый язык общего назначения, интенсивно применяемый для разработки веб-приложений. В настоящее время поддерживается подавляющим большинством хостинг-провайдеров и является одним из лидеров среди языков, применяющихся для создания динамических веб-сайтов.

При создании веб-приложения использовался CodeIgniter – это PHP фреймворк, предназначенный для создания сайтов и интернет-приложений. CodeIgniter использует MVC архитектуру- «Модель-Вид-Контроллер» - схема разделения данных приложения

и управляющей логики на три отдельных компонента: модель, представление и контроллер — таким образом, что модификация каждого компонента может осуществляться независимо[2].

Bootstrap 5 - свободный набор инструментов для создания сайтов и веб-приложений. Включает в себя HTML- и CSS-шаблоны оформления для типографики, веб-форм, кнопок, меток, блоков навигации и прочих компонентов веб-интерфейса, включая JavaScript-расширения[3].

CSS(Каскадные таблицы стилей) — формальный язык описания внешнего вида документа, написанного с использованием языка разметки.

Visual Studio Code - редактор исходного кода, разработанный Microsoft для Windows, Linux и macOS. Позиционируется как «лёгкий» редактор кода для кроссплатформенной разработки веб- и облачных приложений. Включает в себя отладчик, инструменты для работы с Git, подсветку синтаксиса, IntelliSense и средства для рефакторинга.

AJAX - подход к построению интерактивных пользовательских интерфейсов веб-приложений, заключающийся в «фоновом» обмене данными браузера с веб-сервером.

jQuery - набор функций JavaScript, фокусирующийся на взаимодействии JavaScript и HTML. Библиотека jQuery помогает легко получать доступ к любому элементу DOM, обращаться к атрибутам и содержимому элементов DOM, манипулировать ими.

JavaScript — язык программирования. Поддерживает объектно-ориентированный, императивный и функциональный стили.

Для вывода карты России использован jVectorMap, в который состоит из двух модулей: конвертера карт, написанного на Python, и плагина jQuery, который рендерит карты и предоставляет JavaScript API для управления ими. jVectorMap разработан Кириллом Лебедевым[4]. Для отображение поданных заявок на карте был разработан код на языках PHP и JavaScript.



Рисунок 12 – Отображение заявок на карте

Программный продукт размещен в сети Интернет по адресу <https://ovpolyk.ru/invaprofi/>

В дальнейшем предполагается доработка данного проекта с целью сделать его универсальным, обеспечить возможность использования его для проведения любого онлайн конкурса.

#### Список использованных источников

1. Порядок и условия проведения Всероссийского конкурса профессиональных достижений «ИнваПрофи» среди работников образовательно-

реабилитационных организаций для инвалидов и лиц с ограниченными возможностями здоровья

2. <https://codeigniter3.ru/> - это фреймворк разработки приложений - инструментарий - для тех, кто создает веб-сайты с помощью PHP.
3. <https://bootstrap-5.ru/> - материалы по популярному фреймворку для разработки адаптивных и мобильных web-проектов
4. <https://jvectormap.com/> - конвертер карт

## БУДУЩЕЕ ВЕБ – РАЗРАБОТКИ

*Сагайдак Валерий Александрович  
ФКПОУ «НТТИ» Минтруда России  
Бутова Жанна Владимировна, преподаватель*

Сайтов с каждым днем становится все больше. Одни презентуют, другие информируют, третьи продвигают, четвертые продают, пятые развлекают... Мир уверенно уходит в онлайн, что открывает огромные возможности для разработчиков и в то же время становится большим вызовом. Веб активно развивается, при этом стандарты порой меняются быстрее, чем могут быть реализованы и проанализированы. Поэтому сегодня недостаточно хорошо разбираться в действующих технологиях и успешно применять их. Необходимо держать руку на пульсе, мыслить на опережение — быть в курсе трендов веб-разработки, читать будущее индустрии.

### Прогрессивные веб-приложения (PWA)

PWA — Progressive Web Application (Прогрессивные Веб-Приложения) — технология, которая позволяет и визуально, и функционально трансформировать сайт в мобильное приложение, рисунок 1.



Рисунок 1 - Progressive Web Application

Обеспечивает быструю загрузку страниц, делает возможным корректную работу приложения без подключения к интернету, задействует меньше памяти на устройстве, чем классическое приложение. Но главное, что используют сегодня все больше компаний, это отправка Push-уведомлений без загрузки дополнительных ресурсов [1].

Push-уведомления – это маленькие окна с сообщениями, всплывающие на экране устройства. Они содержат совершенно разную информацию: от новостей и акций до обновлений и персональных предложений.

В целом, Progressive Web Application делает приложение:

- Более надежным. Обеспечивает быструю, корректную загрузку, независимо от наличия и качества интернет-соединения.
- Удобным. Исключает “подвисание”, длительную загрузку “тяжелых” страниц, существенно улучшает плавность, отзывчивость UI.
- Привлекательным. Реализуемы все типы веб-дизайна, от простого текстового до 3D. Это возможность сделать для пользователя работу с приложением максимально комфортной и приятной, чтобы в дальнейшем хотелось пользоваться им снова.

Также популярность технологии во многом обусловлена экономической выгодой: разработка и поддержка приложений на базе PWA на 70% дешевле аналогичных работ над обычными приложениями.

Технологию PWA уже используют такие бренды, как Twitter, Tinder, Uber, Telegram, Starbucks, Forbes, AliExpress.

Какие задачи можно решать при помощи приложения на базе PWA.

- Стимулировать повторные продажи.
- Упрощать поиск компании.
- Повышать узнаваемость бренда. Иконка (логотип) компании всегда на виду у пользователя
- Повышать средний чек сделки, допродавая продукты при помощи Push-уведомлений.
- Оставаться на связи с клиентом. Отправляйте клиенту уведомления о том, что товар прибыл в пункт самовывоза. Или что товар будет доставлен сегодня.
- Экономить на рекламе. Теперь не нужно настраивать догоняющую рекламу с новой акцией. Достаточно отправить уведомления, которое точно заметят ваши клиенты.

Избавиться от конкуренции в поисковой выдаче. Ваши клиенты обращаются через приложение, в котором нет конкурентов, в отличие от поисковой выдачи.

## **Искусственный интеллект (ИИ)**

Разработка функциональных сайтов сегодня не обходится без искусственного интеллекта.

Сегодня наличие веб-сайта является обязательным для всех бизнес-ниш на рынке. ИИ играет жизненно важную роль в изменении веб-дизайна и разработки, рисунок 2.



Рисунок 2 - Искусственный интеллект

## Чат-боты - информирующие, ведущие коммуникацию с пользователями от лица компании или бренда

Основное применение ИИ в чат-ботах - увеличить продажи бизнеса за счет лучшего прогнозирования ответов. От имени компании с клиентом говорит робот либо с подключенным искусственным интеллектом, либо без него, но тогда по заданному сценарию, который ведет клиента к достижению определенной цели.

Умные роботы помогают клиентам общаться на разные темы с любимыми брендами и выполнять некоторые действия: узнавать новости, решать проблемы, задавать вопросы, участвовать в акциях, получать моментальные бонусы, зарабатывать скидки, оставлять отзывы, следить за покупками, получать поздравления и полезную информацию, специальные цены, обучаться, пользоваться финансовыми услугами и многое другое. Помимо того, что компании с помощью виртуальных помощников уменьшают на 70% количество обращений по почте, телефону или в онлайн-чатах с сотрудниками и т.д., рисунок 3. [3].

Разработчики создают, а компании приобретают и внедряют чат-ботов как с пользой для себя, так и для клиентов, которым они помогают экономить время при обращении к ним и получать массу других преимуществ. Да и, как оказалось, у клиентов больше доверия и желания общаться в дистанционных чатах, нежели пользоваться мобильными приложениями.

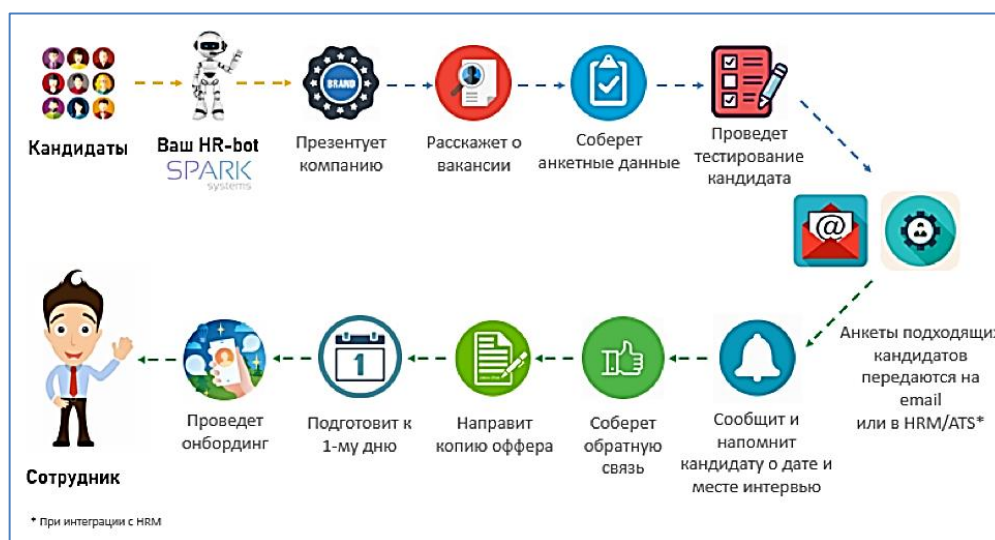


Рисунок 3 – Чат -бот

Методы искусственного интеллекта, такие как шифрование данных, анализ поведения и многофакторная аутентификация, обеспечивают максимальную безопасность общения чат-ботов.

Аналогичным образом, технология может быть использована для разработки продвинутых веб-сайтов или устройств с поддержкой Интернета, чтобы эффективно связать поведение человека с технологией.

## Более удобный пользовательский интерфейс

В эпоху цифровых технологий отрасли придерживаются подхода, ориентированного на клиента, для получения прибыли и получения конкурентных преимуществ. Клиентский опыт стал эпицентром современного делового мира.

В то время как технические достижения стимулируют поведение потребителей, искусственный интеллект привнес уникальный импульс и стал чемпионом трансформации.

Технологии машинного обучения и искусственного интеллекта настолько глубоко встроены в ежедневно используемые дизайнерами программы, что часто они перестают замечать их присутствие. Есть разные сценарии использования искусственного интеллекта (ИИ) в дизайне интерфейсов. К стандартным задачам относят, например, изменение размера изображения и его автоматическую коррекцию. В некоторых приложениях (Uizard и др.) технология ИИ превращает нарисованные от руки вайрфреймы (макеты интерфейса невысокой точности) для мобильного или веб-приложения в готовый прототип: ИИ анализирует элементы, расположение фигур, изображений и на основе анализа создает готовый файл. [4].

Кроме того, при помощи ИИ можно быстро повысить уникальность текста посредством его автоматического переписывания, преобразовать аудио в текст и подготовить описание фотографий, что полезно в частности для онлайн-магазинов и других сайтов, рисунок 4.



Рисунок 4 – ИИ в пользовательском интерфейсе.

ИИ-решения также используются в работе с визуальным контентом. Например, нейросети применяются для отрисовки сопровождающей графики для онлайн-статей и позволяют создавать гораздо больше идей в единицу времени. А технология дипфейков (deepfake) делает реалистичную подмену объекта на видео с возможностью автоматического наложения масок и спецэффектов (правда, стоит отметить, что эта технология сегодня нашла наибольшее применение для дискредитации отдельных людей).

ИИ применим для поиска референсов. Дизайнеры интерфейсов часто обращаются к референсам (вспомогательным изображениям) для вдохновения (чтобы открыть для себя новые стили, визуальные приемы в оформлении и т. д.), а также чтобы приблизительно визуализировать результат своей работы и донести свое видение до коллег. В этом смысле полезны сервисы, например Pinterest, которые выстраивают ленту пользователя с учетом его поискового запроса, причем с каждым новым запросом система обновляет контент.

Часто проектирование любых пользовательских интерфейсов (UX/UI ) используют возможности ИИ для генерации контента по запросу. Представим, что на сайте предполагается наличие множества фотографий людей, например, демонстрация аватарок пользователей какого-либо сервиса. Велик риск нарушить этические нормы, так как использовать фотографии реальных людей без их разрешения нельзя, и при этом поиск и проверка каждой фото на законность использования требует больших временных ресурсов. Альтернатива — генератор фальшивых лиц несуществующих людей на базе AI, который создает новые изображения из частей. Причем 90% фальшивых изображений не распознаются обычным человеком, а 50% — опытным



фотографом. Иногда ИИ допускает ошибки, и на изображении могут появляться баги: неправильно изогнутый контур, сомнительный цвет волос или глаз. Однако программы быстро обучаются, и уже сейчас вероятность таких неточностей близится к нулю.

### Голосовой поиск и навигация

Смартфоны, планшеты, ноутбуки, смарт-часы — для современных устройств становится нормой голосовое управление. Еще бы, это очень удобно, — нужно меньше действий, чтобы устройство выполнило какую-либо задачу, стоит только приказать.

К управлению голосом мы быстро привыкли, поэтому нередко на подсознательном уровне ожидаем, что любой сервис предоставит такую возможность. Веб-разработчики просто обязаны это учитывать. [2].

Тем более реализовать голосовой поиск и навигацию на сайте — не проблема. Специально для этого предусмотрена технология Web Speech API — добавление функционала распознавания и синтеза речи в веб-страницу.

Распознавание реализуется благодаря интерфейсу SpeechRecognition, за синтез отвечает SpeechSynthesis.

Поиск и навигация голосом — это удобно, быстро, просто. То, что может сделать сайт (приложение) более конкурентоспособным, популярным и существенно помочь его продвижению, рисунок 5.

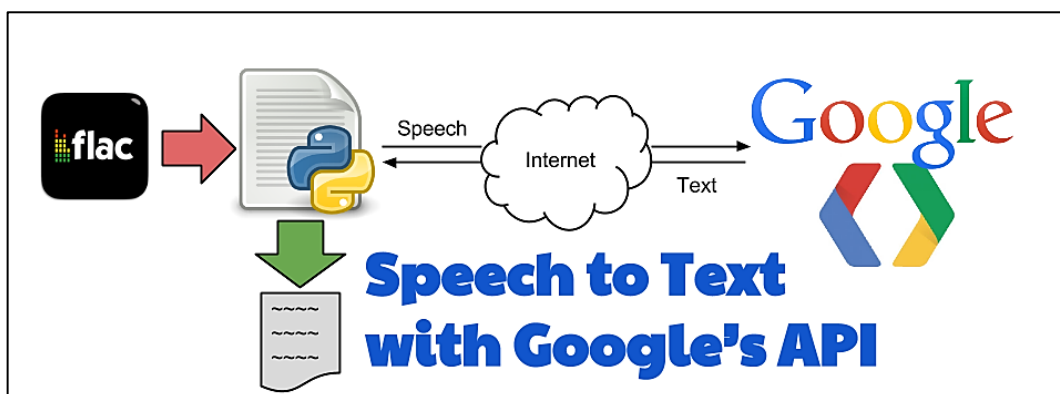


Рисунок 5 – Web Speech API

### Mobile First

Количество пользователей, использующих для выхода в Сеть мобильные устройства, становится все больше. Разработчики должны считаться с этой тенденцией, переориентируясь на создание сайтов и приложений, оптимизированных под смартфоны или планшеты. То есть к разработке должен быть применен подход Mobile First — создание быстрых, легких, корректно отображающихся на всех мобильных устройствах сайтов и приложений.

Mobile first — это стратегия разработки сайта, согласно которой изначально сайт создается под мобильные устройства, с последующей адаптацией под планшеты и ПК.

При таком подходе не возникает ощущение, что мобильная версия обрезана или недоработана — элементы не наезжают друг на друга, а все меню и ссылки корректно работают, рисунок 6.



Рисунок 6 - Mobile First

Главные тезисы Mobile First:

- Быстрая загрузка сайта, несмотря на качество подключения к Сети.
- Корректное отображение контента на всех мобильных устройствах.
- В первую очередь показ самого важного для пользователя содержания.
- Сайт не должен загружать больше ресурсов, чем требуется пользователю для получения информации. Дополнительные данные должны подгружаться только по требованию пользователя.

Для реализации подхода используют PWA(прогрессивные веб-приложения), о которых рассказывалось ранее, а также AMP — Accelerated Mobile Pages — технологии ускорения загрузки мобильных страниц. Сегодня применяются практически во всех популярных приложениях.

### **Motion Design**

Motion Design — Моушн-дизайн — анимированная графика в приложениях и сайтах, один из главных трендов веб-дизайна. [2].

В первую очередь делает веб-проект более привлекательным, запоминающимся. Кроме привлечения внимания пользователей, положительно влияет на юзабилити, рисунок 7.



Рисунок 7 - Motion Design

Позволяет реализовать красивые:

- Переходы заголовка;
- Диаграммы;
- Выпадающие списки;
- Всплывающие окна;

- Прокрутку страницы;
- Фон.

С помощью Motion Design можно сделать веб-проект уникальным, ярким, упростить его восприятие. Совместив с технологией AMP, внедрить анимированную графику, при этом сохранить высокую скорость загрузки страницы.

На лендингах и многостраничных сайтах часто можно встретить анимированные элементы. Промосайты могут быть полностью основаны на моушн-графике.

## Бессерверная архитектура

Как минимизировать риск перегрузки системы, потери данных веб-проекта и дорогостоящей разработки? Постепенно переходить со стандартных серверов на облачные технологии! [2].

Бессерверная архитектура — AWS, Google Cloud, Microsoft Azure и т.п. — еще один тренд веб-разработки. Создание, запуск приложений без необходимости обслуживания инфраструктуры. Все задачи, связанные с поддержанием работоспособности, управления серверами, система полностью берет на себя. Исправление ошибок, масштабирование, выделение дополнительных ресурсов — с бессерверными технологиями об этом можно не беспокоиться, сконцентрироваться только на разрабатываемом продукте, рисунок 8.

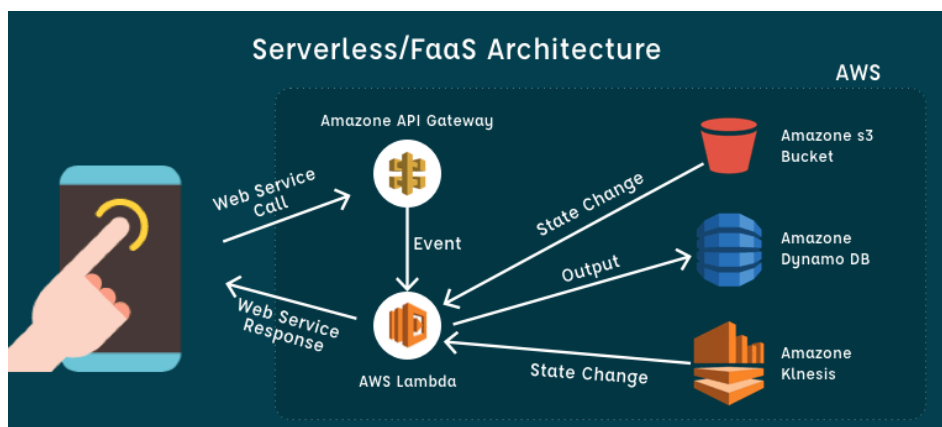


Рисунок 8 - Бессерверная архитектура

Бессерверная архитектура позволяет:

- Сэкономить силы и время сотрудников;
- Уменьшить расходы на обслуживание инфраструктуры;
- Сократить бюджет на разработку и поддержку;
- Быстро адаптироваться к непрерывно меняющимся требованиям к разработке;
- Повысить надежность хранения данных.

Технологии бессерверной архитектуры ускоряют загрузку резервных копий, экспорт объектов, доставку уведомлений. В целом повышают гибкость, надежность и качество разработки.

## Модель RAIL

RAIL — аббревиатура, образованная от четырех английских слов: Response (Отклик), Animation (Анимация), Idle (ожидание), Load (Загрузка). Это модель улучшения производительности сайтов и приложений, рисунок 9 [3].

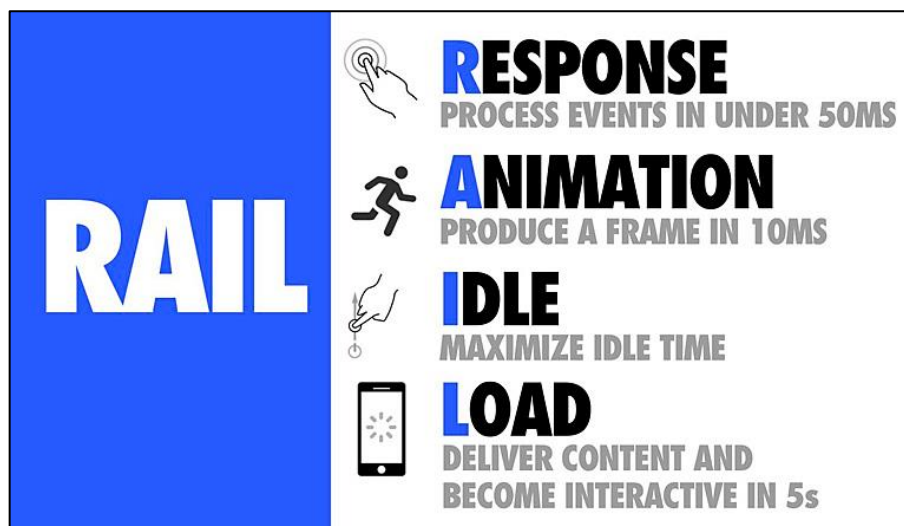


Рисунок 9 – RAIL

Позволяет реализовать:

- Мгновенный отклик — за менее чем 100 миллисекунд после действия пользователя.
- Анимацию — как традиционную (индикаторы загрузки, изменения состояния, динамические заставки и т.д.), так и при скроллинге, но обязательно непрерывную, с появлением каждого кадра на экране за менее чем 16 миллисекунд, то есть со скоростью 60 FPS.
- Ожидание — сначала загрузку важных компонентов страницы, к которым пользователь хочет моментально получить доступ, а затем уже в фоновом режиме подгрузку остальных составляющих. Т.е. использование время ожидания, когда приложение не обрабатывает взаимодействие с пользователем, для выполнения различных процессов, от загрузки страниц до сбора и отправки данных в сервис аналитики.
- Загрузку — всегда корректную, правильную с точки зрения логики и пользовательского опыта. Например, если загружается страница, то в первую секунду появляется ее главная часть, то есть максимально полезная для пользователя информация, а не, скажем, малоинформативная шапка, а затем уже основной контент — так не должно быть.

Модель RAIL помогает веб-проект сделать быстрым. Решает проблему с плавностью анимации. Позволяет использовать время ожидания для выполнения второстепенных процессов. Обеспечивает логику загрузки компонентов страницы и мгновенный отклик на действия пользователя.

#### Список использованных источников:

1. Будущее разработки или что ждет веб-индустрию в ближайшие 5 лет [Электронный ресурс]. – Режим доступа: <https://proglib.io/p/budushchee-razrabotki-ili-chto-zhdet-veb-industriyu-v-blizhayshie-5-let-2020-12-14>
2. Тренды веб-разработки 2022. Обзор главных тенденций. [Электронный ресурс]. - Режим доступа: <https://blog.sibirix.ru/web-development-trends-2022/>
3. Будущее веб-разработки [Электронный ресурс]. – Режим доступа: <https://alma-com.ru/articles/trendy-i-budushchee-veb-razrabotki/>
4. 10 самых популярных технологий, которые произведут фурор в веб-разработке в 2022 году [Электронный ресурс]. - Режим доступа: <https://media/simpletech/10-samyh-populiarnyh-tehnologii-kotorye-proizvedut-furor-v-vebrazrabotke-v-2022>

## СЕКЦИЯ 2. ДОСТИЖЕНИЯ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ В ОБЛАСТИ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА, МАШИННОЕ ОБУЧЕНИЕ, РОБОТОТЕХНИКА И КИБЕР-ФИЗИЧЕСКИЕ СИСТЕМЫ

### ИСКУССТВЕННОЕ ИСКУССТВО: КАК НЕЙРОСЕТИ НАУЧИЛИСЬ ТВОРИТЬ?

*Козлов Вадим Алексеевич*

*ФКПОУ «Сиверский техникум-интернат бухгалтеров» Минтруда России  
Матвеева Надежда Георгиевна, преподаватель информатики*

**Аннотация.** В статье рассматривается одно из самых перспективных направлений искусственного интеллекта - нейронные сети и их возможности в изобразительном искусстве. Основной вопрос данной статьи: сможет ли нейросеть написать лучше картину художника? Приведены примеры трех картин выдающихся художников: «Бурлаки на Волге» Репин И.Е., «Лунная дорожка» Айвазовский И.К., «Звёздная ночь» Винсент Ван Гог и их аналоги в исполнении нейросетевых технологий сайта Ostagram.

На сегодняшний момент искусственный интеллект прочно вошел в нашу жизнь и помогает в решении большого числа задач. Одно из самых перспективных направлений искусственного интеллекта, приближающего будущее из фантастических фильмов, являются нейронные сети. Уже сейчас они активно используются в бизнесе, особенно в маркетинговой работе, применяются в сфере безопасности, развлечения и других областях. Исследованиями в этой области занимаются все самые передовые компании, например, такие как Microsoft и Google, что способствует появлению все новых открытий в этой области чуть ли не каждый день [3].

Для начала, давайте разберемся, что собой представляют нейронные сети и как они работают. Искусственные нейронные сети построены по принципу биологических, конечно, с рядом допущений, в них действует огромное количество простых процессов с множеством связей. Подобно человеческому мозгу эти сети способны обучаться. Для искусственных нейронных сетей под обучением понимается процесс настройки архитектуры сети (структуры связей между нейронами) и весов синоптических связей (влияющих на сигналы коэффициентов) для эффективного решения поставленной задачи. Обычно обучение нейронной сети осуществляется на некоторой выборке. По ходу обучения сеть начинает все лучше выполнять поставленные задачи, реагировать на поставленные команды [3]. Искусственный интеллект может вдохновить художников работать дальше над усовершенствованием материала, может дать идеи и подсказать стиль написания новой картины [2].

Нейронные сети могут обрабатывать фото, как по заданным параметрам, например, превращая обычный снимок в изображение по стилю похожее на указанную автором репродукцию, или превратить эскиз в проработанный рисунок, дорисовав все элементы, так же сеть может творить по собственному усмотрению, самостоятельно выбирая стиль итогового изображения. Нейросети пишут музыку, некоторые сервисы придумывают и воспроизводят простые мелодии, а есть такие, что пишут целые альбомы, придумывая слова к музыке, саму музыку создают люди, как и потом накладывают на нее слова, на выходе совместный результат оказывается неотличимым от того же, но полностью сделанного людьми. Создание первого трейлера к фильму, написание сценария, по которому отсняли артхаусное кино – сфера искусства уже не является сугубо человеческой сферой [3].

Для создания новых оригинальных эффектов в современной живописи художники нередко совмещают различные живописные техники и стили. Нейросетевые технологии сайтов DeepArt [4] и Ostagram [5] позволяют создавать подобные картины, объединяя два изображения в одно. В этом случае художник и нейросеть становятся соавторами нового произведения искусства [2].

Чтобы создать свою неповторимую картину с помощью нейронных сетей необходимо выполнить ее в несколько этапов. Нужно выбирать исходный образ, например, фото портрета и фото пейзажа или натюрморта.

Рассмотрим этапы выполнения картины на основе нейросетевых технологий сайта Ostagram (<https://www.ostagram.me/>):

**1 этап.** Выбор нейросети сайта Ostagram, регистрация на сайте;

**2 этап.** Выбор исходного файла из Интернета, загрузка файла (Рис. 1. А, Г, Ж);

**3 этап.** Выбор фильтра из библиотеки сайта Ostagram (Рис. 1. Б, Д, З);

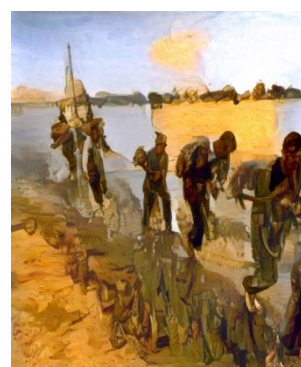
**4 этап.** Преобразование выбранных образов в новую картину с помощью нейросетевых технологий сайта Ostagram (рис. 1, В, Е, И).



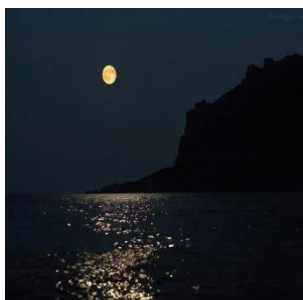
А) Исходное изображение



Б) «Бурлаки на Волге»  
Репин И.Е.



В) Работа нейросети



Г) Исходное изображение



Д) «Лунная дорожка»  
Айвазовский И.К.



Е) Работа нейросети



Ж) Исходное изображение



З) «Звёздная ночь»  
Винсент Ван Гог



И) Работа нейросети

Рисунок 1 - Этапы выполнения картины с помощью нейросетевых технологий

Результаты таких интегрированных работ могут быть проданы за огромные деньги. Например, на нью-йоркском аукционе Christie's произошло знаменательное событие: там впервые была продана работа (рис.2), созданная искусственным интеллектом. При первоначальной оценке в \$7000-\$10 000, ее продажная цена составила \$432 000. Порядок цен соответствует мастерам «первой величины»: на том же аукционе серия принтов Энди Уорхола была продана за \$780 000 [1].

Работа (рис.2) была создана трио 25-летних французских студентов. Они сделали серию портретов, изображающих членов вымышленной семьи Белами. Можно было бы сказать, что портрет Эдмона Белами стал вершиной их творчества, однако творили не они, а программный код. Автор кода — 19-летний Робби Баррат, студент Стэнфордского университета [1].



Рисунок 2 - Портрет Эдмона Беллами написанный алгоритмом.

Во всей истории с портретом Белами важно даже не кто и как сделал эту работу, а то, что покупатель был готов заплатить за нее почти \$0,5 млн. Это и стало причиной всех дискуссий: принт, созданный нейронной сетью и проданный за \$7000, вряд ли послужил бы предметом жарких споров [1].

Нейронные сети могут творить, но оригинальнее и красивее ли людей (рис.1)? Нейросеть практически полностью повторила картины деятелей культуры, смогла перенести стиль на другое изображение, но сделала это смазано.

Итог. Алгоритм способен создать изображение, повторить стиль художника, сделать что-то уникальное, но искусство ли это? Или же искусство, это только творение рук человека? Ныне нейронные сети далеки от человека, они не способны передать эмоции и переживания - это только новый инструмент в руках человека, искусственный интеллект пока не может превзойти талант.

#### Список использованных источников:

1. Кинякина Е. Искусственное искусство: как нейросети научились творить// Сетевой журнал Forbes. Рубрика «Технологии». 29.10.2018.

URL: <https://www.forbes.ru/tehnologii/368515-iskusstvennoe-iskusstvo-kak-neyroseti-nauchilis-tvorit>(дата обращения: 07.12.2022).

2. Пирязева Т.В., Меркушина Ю.В., Герасименко И.И., Павельева И.Н. Организация и проведение Международного конкурса проектов «Искусственный интеллект в создании картин» / Современные информационные технологии в образовании, науке и промышленности: XVIII Международная конференция, XVI Международный конкурс научных и научно-методических работ. Международный конкурс «Искусственный интеллект в создании картин»: Сборник трудов / Отв. ред. и сост. Т.В. Пирязева. – М.: Издательство «Экон-Информ», 2021. – С. 211-218. URL: <https://www.elibrary.ru/item.asp?id=30781773>(дата обращения: 07.12.2022).

3. Фаустова К.И. Нейронные сети: применение сегодня и перспективы развития// Территория науки. 2017. №4. URL: <https://cyberleninka.ru/article/n/neyronnye-seti-primeneniye-segodnya-i-perspektivy-razvitiya> (дата обращения: 07.12.2022).

4. DeepArtEffects. Приложение для обработки изображений с помощью искусственного интеллекта. URL: <https://www.deeparteffects.com/> (Дата обращения 07.12.2022г).

5. Ostagram. Сайт для обработки изображений с помощью искусственного интеллекта URL: <https://www.ostagram.me/> (Дата обращения 07.12.2022г).

## ЧТО ТАКОЕ УМНЫЙ ДОМ

*Мурашев Владимир Константинович  
ФКПОУ «МЭКИ» Минтруда России  
Тарасова Ксения Игоревна, преподаватель*

Кто из нас не мечтает о том, чтобы его дом был уютным, комфортным и функциональным? Но в полной мере ощутить на себе все эти качества смогли только те люди, которые установили в своем жилье систему «умный дом». Современный «умный дом» как никогда приблизил к реальности мечту фантастов о том, что все функции, связанные с комфортной жизнедеятельностью человека, лягут на плечи компьютеров.

### **И так, что же такое «умный дом»?**

«Умный дом» или **smart home** — это система автоматизации домашних устройств, которые выполняют свои действия и могут решать определенные задачи без участия человека [4].

История «умного дома» началась в 1961 году, когда Джоэль и Рут Спира изобрели и запатентовали специальное устройство для плавной регулировки света – диммер. Именно это изобретение стало поводом для создания всемирно известной сегодня компании Lutron Electronics Company, Inc. Данная фирма продолжала работать над «умными» технологиями, параллельно внедряя в обиход такие понятия, как световая зона и сцена.

Крупным событием в дальнейшем развитии технологии «умного дома» было создание шведской компанией Pico Electronics бытовой автоматики в 1975 году, которую впервые начали использовать для управления музыкальными проигрывателями. Усовершенствовали домашнюю автоматику американцы Скотт и Рослин Миллер [2].

Первым же полноценным проектом «умного дома» стал небольшой жилой дом на южном берегу Англии. В основу его автоматики легло использование широкополосной KNX-системы, отвечающей за управление освещением, сигнализацией, жалюзи, отоплением и дверями гаража. Также в данном доме был создан бассейн, который впоследствии дополнили LED-системой с оригинальными цветовыми эффектами.





Рисунок 1 – Smart home

### Какие функции у «умного дома»?

В доме могут стоять датчики, которые реагируют на повышение или понижение температуры. Если на улице и в доме становится холоднее, датчики это считывают, информация поступает в специальный центр управления, который дает команду повысить уровень обогрева. Как только температура в доме становится приемлемой, центр отменяет свою команду. Так уровень тепла поддерживается на одном, заданном хозяином, уровне.



Рисунок 2 – Поддержка уровня температуры дома с помощью специальных датчиков

**Для безопасности** в доме устанавливается камера, которая «видит», кто подходит к дому в отсутствии хозяина и записывает это изображение. Если нужно, домофон включает свет на крыльце, чтобы лучше разглядеть посетителя. А еще владелец может посмотреть изображения с камер прямо во время рабочего дня.

Для того, чтобы «умный дом» начал функционировать, нужно дать ему **некоторые параметры**. Например, указать точное время открытия и закрытия жалюзи, температуру отопления, воды из крана. Если из дома все ушли и внезапно датчики обнаруживают активность, должен поступить сигнал на пульт охраны. Более высокий уровень — когда «умный дом» способен сам принимать решения. Например, если отключат отопление, дом может активировать котел для обогрева помещения. Или на основании часто повторяющихся действий хозяина дом может предложить другое решение. Например, если хозяин вечером смотрит телевизор в темноте, то дом может автоматически выключать свет при включении телевизора в темное время суток.



Рисунок 3 – Функции «умного дома»

Если делить «умный дом» на области, то можно выделить три вида систем, таких как система управления освещением, система кондиционирования, вентиляции и отопления, система безопасности и наблюдения [5]. Помимо этого, в любом «умном доме» должно быть три элемента:

- Сами приборы: кофеварки, колонки, холодильники, видеокамеры, домофоны и так далее. Кофеваркой можно управлять со смартфона или задавать ей конкретное время, к которому нужно приготовить кофе.
- Датчики, которые принимают информацию из окружающей среды. Например, датчик протечки воды в связке с «умной» розеткой обесточит прибор, а в связке с «умным» краном — перекроет воду и обязательно оповестит владельца о случившемся.
- Центральный контроллер, который обрабатывает информацию. Панель управления позволяет контролировать все подключенные к ней устройства.

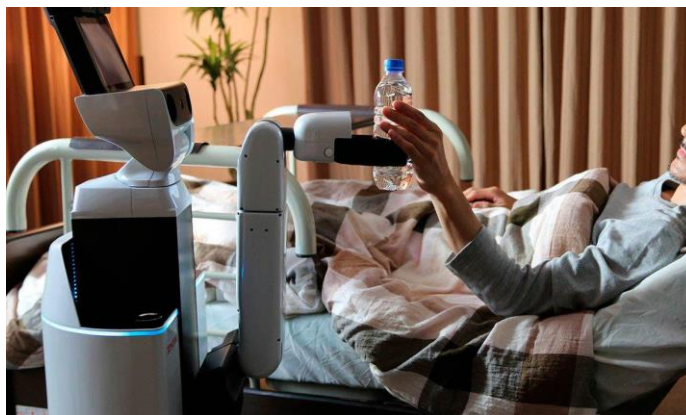


Рисунок 4 – «Умный дом» для людей с ограниченными возможностями здоровья

**Людам с ограниченными возможностями здоровья** зачастую сложно оперировать даже самыми обычными вещами, например, дотянуться до выключателя или вовремя закрыть кран с водой. Избежать этого можно, используя правильную настройку инженерных систем «умного дома»: освещение может активироваться по пути следования, подача воды и газа перекрываются, когда вы покидаете кухню. То же относится к замкам – «умные» модели, оснащенные огромным количеством датчиков, могут отпираться сканером изображения или любым другим способом [3].

Люди с нарушением слуха физически не могут получить звуковое сообщение о том, что, к примеру, кто-то звонит в дверь, обнаружена протечка воды или закипел чайник. Подобная проблема возникает и у тех, кто имеет проблемы со зрением, получая визуальную информацию.

Систему «умный дом» можно настроить на отправку сообщений на любое устройство, включая мобильный телефон, smart-часы. Тип оповещений легко настраивается – от мигающих пиктограмм до вибрации и голосового озвучивания.

На помощь часто приходят также разнообразные голосовые помощники – Apple HomePod, Alexa от Amazon, Ivese, многие другие. Используя их, можно вызвать такси, заказать еду на дом, позвонить в муниципальные службы, управлять домашним кинотеатром, аудиосистемой, искать информацию в интернете, контролировать температуру, освещение в помещении.



Рисунок 5 – Голосовые помощники «Умного дома»

«Умный» домофон может значительно упростить жизнь людям с нарушениями опорно-двигательного аппарата. Он дает возможность узнать, кто именно пришел, впустить его, используя при этом только планшет или смартфон. Более того, его можно настроить автоматически впускать конкретных посетителей, вести журнал визитов.



Рисунок 6 – «Умный дом» - будущее пришло [1]

Появление «умных домов» значительно облегчило жизнь всем категориям людей, особенно – лицам с ограниченными возможностями здоровья. Возможно, в далеком будущем человеку больше не придется вставать с дивана и вручную заниматься домашними делами – ведь дом сможет контролировать состояние самого себя без его участия.

#### **Список использованных источников:**

1. Изображения по теме «Умный дом» [Электронный ресурс] URL: <https://yandex.ru/images/search?from=tabbar&text=умный%20дом&family=yes> (последнее обращение 19.11.22).

2. История развития умных домов [Электронный ресурс] URL: <http://www.aptech.ru/istoriya-razvitiya-sistemy-umnyj-dom> (последнее обращение 18.11.22).
3. Умный дом для инвалидов [Электронный ресурс] URL: <https://mimismart.ru/articles/umnyij-dom-dlya-invalidov/> (последнее обращение 19.11.22).
4. Умный и безопасный дом [Электронный ресурс] URL: <https://www.smahem.com> (Дата обращения: 17.11.2022).
5. Что такое умный дом: функции, виды, составляющие и экосистемы [Электронный ресурс] URL: <https://www.e-katalog.ru/post/1990/618-chto-takoe-umnyu-dom-funkcii-vidy-sostavlyayuschie-i-ekosistemy/> (последнее обращение 17.11.22).

## **СИСТЕМА «УМНЫЙ ДОМ» КОМПЛЕКС ДАТЧИКОВ ПРОТЕЧЕК ВОДЫ**

*Чистов Фёдор Алексеевич  
ФКПОУ «ИвРТТИ» Минтруда России  
Ревьева Ирина Борисовна, заместитель директора по УПР*

**Аннотация.** Система «Умный дом» является серьёзным шагом вперёд в области инженерных, научных и информационных технологий. Система «Умного дома» призвана контролировать очень большое количество настроек, требуемых современному человеку для комфортного проживания. Данная статья раскрывает принцип действия одной из наиболее важных ветвей системы «Умного дома» - комплекса датчиков протечек воды. В статье описаны принцип действия и приоритеты интеграции комплекса датчиков протечек воды с системой «Умного дома» с точки зрения информационной технологии для пользователя.

В настоящее время, значительно увеличилась роль информационных технологий в жизни людей. Современное общество включилось в общеисторический процесс, называемый информатизацией. Этот процесс включает в себя проникновение информационных технологий в научные, производственные, общественные сферы, высокий уровень информационного обслуживания. Процессы, происходящие в связи с информатизацией общества, способствуют не только ускорению научно-технического прогресса, интеллектуализации всех видов человеческой деятельности, но и созданию качественно новой информационной среды социума, обеспечивающей развитие творческого потенциала человека.

Радиоэлектроника — быстроразвивающаяся отрасль в мировом производстве, от состояния которой зависят многие другие отрасли мировой экономики. Это производство и конструирование радиоэлектронных устройств, мобильная связь, системы видеонаблюдения, системы ограничения доступа, кабельное ТВ, обслуживание видео- и аудиотехники. Настройка и регулировка радиоэлектронных устройств на современном уровне – это владение профессиональными компьютерными программами и информационными технологиями.

Подготовка по специальности «Техническое обслуживание и ремонт радиоэлектронной техники» имеет свои особенности. Это связано с быстрой сменяемостью элементной базы и конструктивно-технологических характеристик радиоэлектронных систем, освоением новых технологических процессов и

оборудования, постоянным техническим перевооружением предприятий, а также применением информационных технологий в быту.

Эпоха автоматизации и информатизации привнесла много в различные сферы жизни человека, а именно множество сложных технических устройств и систем. Самое любопытное нововведение в применении информационных технологий в быту – «Умный дом», который в первую очередь избавляет от монотонных часто повторяющихся действий. [1]



Рисунок 1 – Умный дом

Исследуя какую-либо новую технологию, систему важно понимать то, как она устроена, какие компоненты в нее входят, для чего она предназначена, на какой платформе строится управление, как происходит объединение компонентов в единое целое.

В системе «Умного дома» можно выделить два типа устройств: датчики и акторы.[1] Следует понимать, что «умным» можно считать лишь то устройство, которое способно взаимодействовать с другими, используя беспроводную связь.

Система «Умного дома» предполагает большую функциональность:

1. Беспроводная связь между устройствами.
2. Возможность объединения устройств от различных производителей в единую устойчивую систему.
3. Возможность управления всеми исполнительными устройствами через интернет.
4. Возможность создания различных сценариев автоматического исполнения команд.

Как же устроена сама система? Датчики получают информацию о состоянии квартиры и передают ее специальному базовому модулю или хабу. Хаб в свою очередь выполняет роль администратора системы. Он собирает всю поступившую информацию и передает ее на смартфон через интернет, получает команды со смартфона и передает сигналы исполнительным устройствам. Базовый модуль распределяет каналы так, чтобы радиоволны, на которых общаются все электронные устройства в доме, не мешали друг другу, поэтому никаких неполадок при большом количестве устройств не возникает. [1]

Сама система «Умного дома» призвана контролировать очень большое количество настроек, требуемых современному человеку для комфортного

проживания. В этот список попадают устройства температурного контроля, датчики движения, влажности, протечки воды, дыма, схемы автоматического управления освещением и множество других полезных опций.

В данной статье мы подробнее рассмотрим и изучим новую, но действительно актуальную и нужную ветвь, представляющую собой комплекс датчиков протечки воды.

В комплексе система представляет собой:

- 1) Датчик протечки, который передаёт сигнал на контроллер. Принцип действия таких устройств основан на способности воды проводить электрический ток. Когда жидкость попадает на закрепленные в датчиках электроды, электрическая цепь замыкается, и импульс поступает на контроллер.
- 2) Контроллер, который принимает сигнал от датчика протечки и посылает сигнал на исполнительное устройство для соответствующего действия в рамках устранения аварийной ситуации.
- 3) Исполнительное устройство, представляющее собой шаровой электропривод, перекрывающий доступ воды или электронное реле, отключающее насосное оборудование.

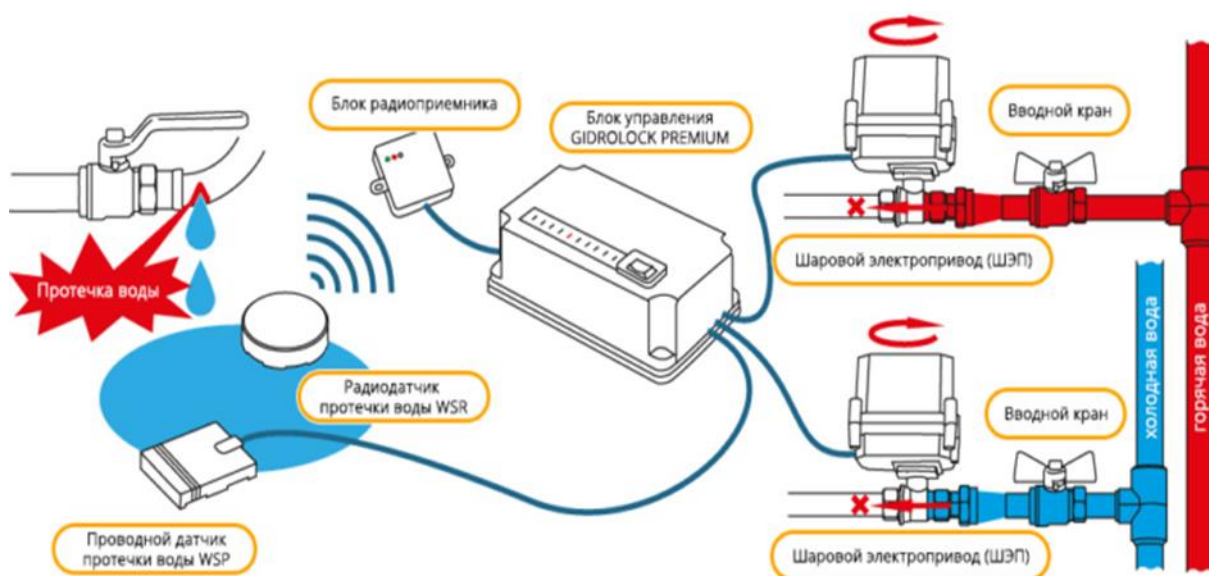


Рис.2 – Структура комбинированной системы защиты от протечки воды

Системы данного типа группируют по ряду признаков:

- 1) По числу датчиков.  
Современные системы «Умного дома» включают в себя до нескольких десятков подобных датчиков для отслеживания аварийных ситуаций.
- 2) По наличию или отсутствию проводного соединения датчика с контроллером.  
Многие датчики в современных домах соединяются с контроллерами посредством беспроводной связи. Такие системы «общаются» между собой путём передачи и принятия радиоволн. В таком случае сам датчик протечки берёт на себя роль передатчика, а контроллер выступает в качестве приёмника. Такая особенность удобна при размещении опции в уже распланированном и застроенном пространстве, в котором нет возможности прокладывать проводные соединения.
- 3) По методу оповещения о протечке.  
Существует множество способов уведомления пользователей о протечке: звуковые оповещения, визуальные уведомления на экранах управляющих

устройств, световые сигналы, а также уведомления в виде сообщений. Для последнего контроллер должен быть оснащен передатчиком GSM. Информация поступает в виде СМС-сообщения на номер абонента, введенный в память прибора. Если система датчиков протечки подключена к интернету, появляется возможность рассылать сообщения с помощью технологии GPRS. [2]

Правила проведения монтажа системы против протечки воды.

Защитный контур представляет собой конструктор, элементы которого соединяются между собой специальными разъемами, или «общаются» между собой на заданной частоте. Перед установкой составляют схему расположения отдельных частей и проверяют соответствие длины проводов тому расстоянию, которое потребуется для подключения измерителей и кранов к контроллеру, или, в случае беспроводного соединения, рассчитывают, будут ли доходить сигналы до компонентов системы. [2]

Порядок проведения работ включает в себя:

- 1) разметку точек монтажа;
- 2) прокладку проводов (в случае беспроводного соединения этот пункт отсутствует);
- 3) врезку кранов;
- 4) установку определителей протечки;
- 5) монтаж управляющего модуля;
- 6) подключение и проверку системы.

Врезка шарового крана.

Самым трудоемким этапом считается крепление шарового крана, что объясняется необходимостью его использования на разных типах труб. Водопровод перерезают в непосредственной близости от предварительно перекрытого вентиля воды. Затем снимают счетчик и закрепляют отсекающий клапан на кране, после чего возвращают в первоначальное положение счетчик воды и участки трубопровода.

Металлопластиковые элементы прижимают контргайкой, полипропиленовые конструкции соединяют пайкой или с помощью разъемных муфт. Для подключения шаровых кранов к распределителю блока питания используют выделенную силовую линию.

Установка датчиков протечки воды

Датчики располагают в местах возможной протечки, при этом особое внимание необходимо уделить переходу между коробом, где помещаются трубы. Это необходимо для того, чтобы при аварии вода попала на датчик, а не продолжала течь мимо него.

Если поверхность пола идет под уклон, устройства располагают в самой нижней точке помещения.

Схема их подключения может быть как напольной, так и внутренней, при которой элементы врезают в материал покрытия. В первом случае пластину размещают контактами вниз и фиксируют двусторонним скотчем или строительным клеем. Такой вариант применяют в тех случаях, когда монтаж системы датчиков выполняют после установки сантехнического оборудования.

При внутреннем расположении устройства его контакты размещают на 3-4 мм выше уровня покрытия, что позволяет исключить срабатывание при случайном разбрызгивании воды или уборке. Соединяющий провод укладывают в гофрированную трубу, непроницаемую для воды.

### Правила монтажа контроллера.

Аппарат размещают в нише или на стене рядом с электропроводкой и запорными вентилями. Блоком питания контроллера служит силовой шкаф, поэтому к устройству подводят фазу и ноль. Провода соединяют с помощью специальных клеммных разъемов, которые для удобства монтажа нумеруют и подписывают. Затем подключают определители протечки воды и приступают к диагностике.

### Проверка работы системы

При включении модуля управления на его панели загорается зеленый индикатор, свидетельствующий о готовности к работе. Если в этот момент намочить пластину датчика водой, свет лампочки изменится на красный, включится подача звукового импульса и отсекающие клапаны заблокируют пуск воды.

### Плюсы интеграции комплекса с системой «Умный дом».

- 1) Главный блок управления системы «Умный дом» будет самостоятельно отслеживать состояние всех систем и выводить их в единый орган оповещения. Такая опция будет крайне удобна для пользователя.
- 2) Осуществление более детальной настройки аварийных действий, перезагрузочных алгоритмов и возможность задания временных перерывов в функционировании аварийной системы.
- 3) Возможность использования встроенного в «Умный дом» алгоритма самостоятельной диагностики неисправностей аварийных систем и систем контроля прочих параметров.

В период развития информационных, научных и инженерных технологий технические достижения стали применяться и в повседневной жизни. Примером этому является представленный комплекс датчиков протечки воды с системой «Умный дом». На современном этапе жизни человека, когда приоритет упрощения и комфорта ведения домашнего быта за счёт внедрения передовых разработок встал в один строй с промышленными и более глобальными технологическими разработками, в наших домах появилось множество приборов и комплексов, о которых тридцать лет назад люди только мечтали. Наше постоянно растущее количество домашних забот отходит на второй план, ведь ими занимаются электронные автоматизированные устройства и «умные» алгоритмы, впоследствии названные системами «Умного дома».

### Список использованных источников:.

1. Иванько А.Ф., Иванько М.А., Котова К.Д. Информационные технологии в жизни обычного человека // Научное обозрение. Фундаментальные и прикладные исследования. – 2019. – № 1. URL: <https://scientificreview.ru/ru/article/view?id=51> (дата обращения: 07.12.2022).
2. Статья в интернете Кванта.ру. Датчик протечки воды: виды и принцип действия, как выбрать и установить систему «анти-потоп» своими руками/ [Электронный ресурс]. – Режим доступа: URL: <https://kvanta.ru/zashhita-ot-protechek/datchik-protechki-vody-vidy-i-printsip-dejstviya-kak-vybrat-i-ustanovit-sistemu-anti-potop-svoimi-rukami#>



## **СЕКЦИЯ 3. ДОСТИЖЕНИЯ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ В ОБРАЗОВАНИИ И НАУКЕ, АВТОМАТИЗАЦИЯ НАУЧНО-ИССЛЕДОВАТЕЛЬСКОЙ ДЕЯТЕЛЬНОСТИ**

### **РАЗРАБОТКА ЭЛЕКТРОННОГО ДНЕВНИКА**

*Багина Екатерина Сергеевна, Казаков Никита Михайлович  
ФКПОУ «ИвРТТИ» Минтруда России  
Завьялова Анастасия Игоревна, преподаватель*

**Аннотация:** статья посвящена информационным технологиям в образовании и науке. На сегодняшний день уровень информационных технологий достаточно высок и электронные дневники медленно, но верно входят в повседневную жизнь средних и высших учебных заведений. Образование не отстает от других сфер жизни, так как учебные заведения воспитывают людей, которым предстоит жить в обществе будущего. В ходе работы были проанализированы готовые информационные решения, применяемые в образовании. Для создания собственного решения была использована информационная система «1С».

Лидирующие позиции любой организации определяются в первую очередь ее возможностями по грамотному использованию передовых достижений в сфере информационных технологий. Особую роль при этом играет сфера образования, где формируется интеллектуальный потенциал страны, потребляющий и создающий ее информационные ресурсы. [1]

Актуальность внедрения цифровых технологий в образовательные учреждения обуславливается повышением качества и уровня образования, так как в данный момент существует тенденция к внедрению информационных технологий во все сферы общества, а образование является одной из наиболее важных сфер общества.

Целью работы является разработка электронного дневника школьника на базе платформы «1С: Предприятие 8.3».

Из данной цели вытекают следующие задачи:

- рассмотрение требований к функционированию системы;
- проанализировать рынок готовых решений;
- создать электронный дневник на базе платформы «1С: Предприятие 8.3».

#### **Требования к функционированию системы**

**Система «Электронный дневник» должна обеспечивать выполнение следующих функций:**

- предоставление информации о расписании занятий, домашних заданиях;
- отображение комментариев преподавателей, расписание звонков и другой информации.
- Внедрение специализированных автоматизированных информационных систем в повседневную деятельность образовательного учреждения позволит:
  - уменьшить количество бумажных данных и перейти на электронный вид управления;
  - стандартизировать процессы управления;

–повысить эффективность управленческих решений на основе использования оперативной, достоверной и актуальной информации.

Разрабатываемая система должна удовлетворять следующим требованиям:

- обеспечение надежности (основным критерием сохранности информации в неизменном виде является ее надежность хранения);
- быстродействие (достигается использованием новых технологий. А также правильной структурой и написанием ее на некотором языке программирования);
- простота управления (удобство интерфейса необходимо для быстрой навигации и является важной частью при проектировании системы);
- кроссплатформенность (способность работать на различных устройствах, операционных системах, серверах);
- адаптируемость (то есть возможность простой настройки под нужды заказчика).

На данный момент рынок программных продуктов предлагает некоторое количество информационных систем, цель которых является автоматизация бизнес-процессов образовательных учреждений. Одними из наиболее востребованных являются: Дневник.ru, 1С:Образование, «СОВА». Данные системы, как правило, обладают следующими функциями:

- управление электронным журналом;
- управление электронным дневником;
- возможность удаленного общения учителей, учеников и родителей.

Проведём анализ каждой из вышеперечисленных систем и выявим различия, а также их положительные и отрицательные стороны.

«Дневник.ру» (<http://dnevnik.ru/>)— школьная социальная сеть, объединяющая все школы России. Портал полезен школам, учителям, родителям. Основные услуги — ведение электронного журнала, личный кабинет, общение родителей с педагогами, управление учебным процессом, полезные ресурсы на сайте. Каждому ученику в Дневнике доступны все выставленные ему оценки по всем предметам. Можно просматривать оценки по предметам и за определённый период (неделя, четверть). [4]

Достоинством является информированность родителей об оценках, посещаемости и домашних заданиях.

Недостатки:

- имеются платные услуги;
- много лишней информации;
- реклама на сайте;
- проект не предполагает sms-рассылки и рассылки в почту.

*Информационная система «1С: Образование».*

Информационная система «1С: Образование» предназначена для организации и поддержки образовательного процесса. Данная система:

- позволяет с большой эффективностью применять в образовательном процессе большой объем цифровых образовательных ресурсов;
  - гарантирует помощь учителю в организации учебного процесса в группах;
  - использует стандарты отображения цифровых ресурсов образования;
  - обеспечивает синхронизацию данных с программным средством 1С: Хронограф.
- [6]

Функциональный потенциал системы:

- создание локального цифрового объёма ресурсов по выбору и организация работы с ним;
- назначение обучающимся как индивидуальных, так и групповых заданий;
- проверка учебной деятельности учащихся;
- формирование работ для конкретного пользователя или группы пользователей;
- коррекция учебных материалов;
- импорт и экспорт цифровых образовательных ресурсов;
- управление списком пользователей;
- ведение статистики успеваемости.

В результате анализа данной информационной системы можно выделить следующие достоинства:

- расширенные возможности по созданию ресурсов;
- невысокие системные требования;
- богатый функционал;
- предполагает совместимость с другими программами “1С”;
- система подходит для больших и малых образовательных учреждений;
- имеет функции документооборота; учёта кадров и анализ кадрового состава;
- начисление и выплата заработной платы[5].

Однако, уместно сказать и о недостатках данной системы:

- внедрение системы требует немалых затрат;
- для качественной и эффективной работы системы требуются другие пакеты программ 1С, которые совместимы с данной системой (как правило, платные).

### ***Информационная система «СОВА».***

Автоматизированная система «Сова» предлагает набор услуг, которые позволяют автоматизировать процедуры учебной деятельности в образовательных учреждениях. Главная идея – формирование контакта между участниками образовательного процесса (ученик, учитель, родители). Систему можно интерпретировать как электронную школу, а именно: расписание, дневник, журнал, и так далее. В системе располагаются такие функции как: отчёты, библиотека, учебные пособия, объявления. Система допускает формировать и просматривать расписание уроков, выставление оценок и фиксировать замечания в журнал и в дневник, опубликовывать как общие, так и индивидуальные задания и контролировать их. [4]

Достоинства:

- относительно невысокая стоимость продукта;
- легкость в использовании;
- упрощённый интерфейс;
- высокая скорость работы системы;
- возможность создавать неограниченное количество различных отчетов, в том числе доступных в Интернете;
- широкий набор функций.

Недостатки:

- не охватывает все бизнес-процессы образовательного учреждения;
- неудобное расположение функций;
- маленькие размеры кнопок.

– Также появился новый формат электронного дневника, одобренный правительством сервис «Московский регистр качества образования» (МРКО). Доступ к нему осуществляется на портале государственных услуг.

Было отмечено много минусов:

- родители на самом первом этапе не понимали, как зарегистрироваться на портале Госуслуг и отыскать там дневник;
- родители отмечают, что система часто зависает, особенно под конец четверти;
- задания в журнале появляются несвоевременно.

Таким образом, в ходе анализа были выявлены аналогичные программные продукты. В ходе тестирования найденного программного обеспечения были найдены следующие проблемы:

- Оценки зачастую исчезают в ходе программного сбоя;
- На определенных типах программного обеспечения некоторые программные продукты не работают;
- Функционал готовых программных продуктов весьма ограничен;
- Введение необходимых функций может быть времязатратным и экономически невыгодным (в некоторых случаях невозможно).

Таблица 1 – Сравнительная характеристика существующих решений

	<b>Дневник.ru</b>	<b>Информационная система 1С: Образование</b>	<b>Информационная система «СОВА»</b>
Ведение статистики успеваемости	+	+	+
Совместимость с другими программами «1С»	-	+	-
Подходит для больших и малых образовательных учреждений	+	+	+
Удобство выставления оценок	-	-	+
Понятный интерфейс	+	-	+
Стабильная работа системы	-	+	-

Из проведённых опытов над исследуемыми информационными системами, было решено создать бюджетную конфигурацию «с нуля» с минимально необходимым и достаточным функционалом. Также данное решение возможно интегрировать в готовую систему 1С.

В программе были созданы справочники, в которых можно посмотреть по каждому учителю информацию: почту, номер телефона, стаж работы, уровень образования и квалификацию (рис.1).

ФИО	Должность	Почта	Номер телефона	Стаж работы
Гирин Михаил Антонович	Учитель физической культуры	as343343as@mail.ru	89 203 457 869	15 лет
Енотиков Иван Вадимович	Учитель информатики	enya123@mail.ru	89 203 676 688	15 лет
Знанин Иван Иванович	Учитель природоведения	fghf123@mail.ru	89 204 658 593	20 лет
Иванов Евгений Петрович	Учитель ИЗО	zxzxx1234@mail.ru	89 205 674 589	5 лет
Котиков Александр Витальевич	Учитель математики	cv34cv@mail.ru	89 156 786 432	20 лет
Мишкина Евгения Владимировна	Учитель английского языка	sdsd@mail.ru	89 205 675 656	15 лет
Ноткина Антонина Витальевна	Учитель музыки	ghgj@mail.ru	89 206 532 489	2 года
Снегирёва Ольга Максимовна	Учитель физики	bn444bn@mail.ru	89 206 986 544	5 лет
Снеженина Анастасия Вадимовна	Учитель литературы	bnbnb@mail.ru	89 205 656 432	14 лет
Снежжова Мария Петровна	Учитель русского языка	xxxxxxx@mail.ru	89 205 778 976	5 лет
Советов Павел Олегович	Учитель истории	istori@mail.ru	89 156 784 578	2 года

Рисунок 3– Хранение информации об учителях

Была также создана карточка для заполнения данных об одноклассниках: дата рождения, номер телефона, почта и дополнительные сведения – хобби и любимые книги. Также можно вставить фото одноклассника (рис. 2).

Обухов Илья Вадимович (Одноклассники) (1С:Предприятие)

Обухов Илья Вадимович (Одноклассники)

Записать и закрыть    Записать    Добавить картинку    Еще ▾

Ссылка на картинку:

Дата рождения:   
 Номер телефона:   
 Почта:

Добавить          Еще ▾

N	Хобби	Любимые книги
1	Рисование	Ремарк "Три товарища"

Рисунок 4 – Заполнение данных об одноклассниках

Создан документ «Учебный день» предназначен для фиксации уроков, что по ним задано и какие оценки получены. Структура документа представлена на рисунке 3.

Провести и закрыть										
Записать										
Провести										
Печать										
Еще -										
Дата: 03.09.2020 12:00:00										
Добавить										
Еще -										
№	Предмет	Ка...	Учитель	Домашнее задание	Выполнено	Оценка	Комментарий учителя	Присутствовал		Другая причина
								Отсутствовал	Болеет	
1	Литература	103	Снежина... Анастасия Вадимо...	рассказ до стр.34	<input checked="" type="checkbox"/>			<input checked="" type="checkbox"/>	<input type="checkbox"/>	
2	Информатика	205	Енотиков Иван Вадимо...	упр.5	<input type="checkbox"/>		опоздал на урок!	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
3	Музыка	104	Ноткина Антонина Виталье...	прослушать композицию, описать в тетради	<input type="checkbox"/>	5		<input checked="" type="checkbox"/>	<input type="checkbox"/>	
4	Природоведение	105	Эквин Иван Иванович		<input type="checkbox"/>			<input checked="" type="checkbox"/>	<input type="checkbox"/>	

Рисунок 3 – Заполнение расписания

В документе завуч заполняет предмет, автоматически подставляются данные: кабинет и учитель. Учитель заполняет домашнее задание, ставит оценку, комментарий и отмечает присутствие ученика на предмете. Ученик ставит галочку, если выполнил задание.

Данную конфигурацию планируется развивать и наполнять всё более новыми функциональными возможностями. В ближайшее время планируется:

- добавление системы взаимодействия, с помощью которой можно будет реализовать живое общение пользователей между собой (чат);
- создать индивидуальное расписание занятий каждого учителя.

Таким образом, был разработан электронный дневник на базе платформы 1С. При переходе к информационному обществу возникает новая индустрия переработки информации на базе компьютерных и телекоммуникационных информационных технологий. Поэтому, одной из основных проблем функционирования современных образовательных учреждений является нехватка или полное отсутствие автоматизированных систем, что затрудняет процесс обработки различной информации в ходе своей деятельности[3].

#### Список использованных источников:

1. Конфигурирование и моделирование в системе «1С:Предприятие»: учебник / Э.Г. Дадян. — Москва: Вузовский учебник: ИНФРА-М, 2021. — 417с.
2. Основы языка программирования 1С 8.3: учебное пособие / Э.Г. Дадян. — Москва: Вузовский учебник: ИНФРА, 2020. — 133 с.
3. Избачков, Ю. Информационные системы : учеб. для вузов / Ю. Избачков. — 3-е изд. — СПб. : Питер, 2018. — 544 с.
4. Исакова А. И. Информационные технологии [Электронный ресурс]: учебное пособие / А. И. Исакова, М. Н. Исаков. - Томск: Эль Контент, 2012. — 174 с.
5. Заботина, Н. Н. Проектирование информационных систем - М.: ДРОФА, 2013. - 336 с.
6. Радченко М.Г. 1С:Предприятие 8.3. Практическое пособие разработчика. Примеры и типовые приемы / М.Г. Радченко, Е.Ю. Хрусталева. — М.: ООО «1С-Публишинг», 2013. — 956 с.: ил.
7. Программирование в 1С: Предприятие 8.3. — СПб.: Питер, 2014. — 304 с.: ил. — (Серия «1С-специалист»)

# ИНТЕГРИРОВАННЫЙ ПРОГРАММНЫЙ КОМПЛЕКС ДЛЯ ПСИХОЛОГИЧЕСКОЙ ДИАГНОСТИКИ УРОВНЯ ЭМОЦИОНАЛЬНОГО ИНТЕЛЛЕКТА ОБУЧАЮЩИХСЯ

*Зеленская Ангелина Владимировна, Уралова Ангелина Анатольевна  
ФКПОУ «НТТИ» Минтруда России  
Арапова Елизавета Александровна, преподаватель*

**Аннотация.** В работе представлена программная реализация методик психологической диагностики уровня эмоционального интеллекта. Программный комплекс направлен на проведение индивидуального и группового тестирования обучающихся, автоматизацию трудоемкого процесса обработки результатов и анализа данных. Выполненный программой статистический анализ может использоваться для психокоррекции и дальнейшей психологической работы с обучающимися. Анализ динамики результатов тестирования позволяет оценить эффективность и результативность коррекционной работы, оперативно принимать решения по ее совершенствованию.

## ВВЕДЕНИЕ

Результативность и обоснованность применения методик, направленных на психодиагностику и коррекцию психо-физиологического состояния обучающегося, определяется во многом качеством и полнотой аналитической информации, полученной в ходе психологического тестирования. Психологическое тестирование играет важнейшую роль в процессе психодиагностики, определения психо-физиологического статуса обучающихся. Особую роль в решении этой задачи играют диагностические инструменты, которые позволяют определить уровень эмоционального интеллекта (ЭИ) – способность человека распознавать и понимать эмоции, намерения и желания других людей, управлять ими [1].

Как правило, методы изучения ЭИ построены как психологические опросники, предполагающие самоотчет человека, и в большей степени измеряют его личностные качества.

Более точными измерительными инструментами, которые определяют уровень ЭИ, являются тесты задачного типа (например, MSCEIT [2]), в которых необходимо выполнять задания, связанные с распознаванием и пониманием эмоций, управлением эмоциональными состояниями и т. п. Трудность массового применения таких методик в образовательных организациях обусловлена сложностью проведения фронтального опроса, колоссальной трудоемкостью обработки полученных результатов.

*Целью данного исследования* является проектирование и разработка интегрированного программного комплекса для автоматизации психологической диагностики уровня эмоционального интеллекта обучающихся.

Для достижения этой цели требуется решить ряд задач:

1. Анализ предметной области, построение архитектуры и алгоритмов программного решения на основе существующих методик психодиагностики ЭИ.
2. Разработка и интеграция программных модулей, модульное тестирование
3. Системное тестирование, верификация и первичная апробация программного решения.

**Этап 1. Анализ предметной области и проектирование программного продукта**

**Методики тестирования эмоционального интеллекта.** В настоящее время психологи рекомендуют использование нескольких методик для измерения ЭИ, самыми популярными из которых являются *Метод Холла, опросник ЭМИн Люсина, тест «руки» Вагнера, тесты с набором зрительных стимулов.*

Наиболее широко используется методика М. Холла, которая предложена для выявления способности понимать отношения личности, репрезентируемые в эмоциях и управлять эмоциональной сферой на основе принятия решений [3].

Тест руки Вагнера [4] предназначен для диагностики агрессивности и широко применяется как для взрослых, так и для детей. Методический прием, положенный в основу теста руки, заключается в том, что испытуемого просят проинтерпретировать содержание действия, представленного в виде «стоп-кадра» изображения кисти руки, при показе которой просят представить кисть руки и описать ее воображаемые действия

При обработке полученных результатов каждый ответ испытуемого относят к одной из 11 категорий: агрессия, Указание, Страх, Эмоциональность, Коммуникация, Зависимость, Демонстративность, Увечность, Активная безличность, Пассивная безличность, Описание. После категоризации подсчитывается число высказываний в каждой категории. На основе этих данных вычисляется суммарный балл агрессивности по формуле:

$$S_A = (A + У) - (C + Э + K + З) \quad (1)$$

где  $S_A$  – суммарный балл агрессивности, А – Агрессия, У – Указание, С – Страх, Э – Эмоциональность, К – Коммуникация, З – Зависимость.

Таким образом, первый член формулы характеризует агрессивные тенденции, второй – тенденции, сдерживающие агрессивное поведение.

Интерпретация результатов теста основывается на сравнении суммарного балла с тестовыми нормами и принятии решения об уровне агрессивности и актуальном состоянии испытуемого. Анализ удельного веса и места агрессивных тенденций в общей системе диспозиций, который определяется как доля ответов в категории «Агрессия» по отношению к суммарному количеству ответов в остальных категориях, дает более точный прогноз агрессивного поведения обучающегося.

Одними из самых релевантных в психологической практике считаются тесты с набором зрительных стимулов в виде фотографий. Перед испытуемым появляются фотографии с лицами человека и из представленных фото необходимо выбрать заданную эмоцию максимально быстро. На каждую эмоцию должна быть серия предъявлений (в серии должны быть в равной степени представлены лица женщин и мужчин, если будут расовые фото, то они тоже должны быть в равной пропорции представлены).

**Проектирование архитектуры и структуры и данных.** Высокоуровневое проектирование архитектуры рассматриваемой системы выполняется на основе унифицированного языка структурного анализа и проектирования, предложенного в [5]. Концептуальная модель базируется на DFD – технологии, обеспечивая тем самым наглядность, простоту в понимании и использовании модели, а также преимущество в рамках цикла моделирования: от бизнес-моделей до моделей требований и проектных решений.

В контексте данной задачи целесообразно создание комплексной структурной модели на основе двух групп средств моделирования:

- 1) диаграммы потоков данных DFD (data flow diagrams), которые иллюстрируют бизнес-процессы системы и их интерфейсы. Главная цель таких диаграмм - показать, как циркулируют внутри системы информационные потоки, с момента их поступления от внешних источников и до получения внешними приемниками;



2) диаграммы «сущность-связь» ERD (Entity Relationship diagrams), которые моделируют реляционную структуру данных и их взаимосвязи.

Важнейшей связующей частью комплексной концептуальной модели [5] является DFD-диаграмма, которая показывает внешние по отношению к системе адресаты данных, выделяет внутренние логические функции (бизнес-процессы), связывающие их информационные потоки, а также накопители (хранилища) данных (рис.1).

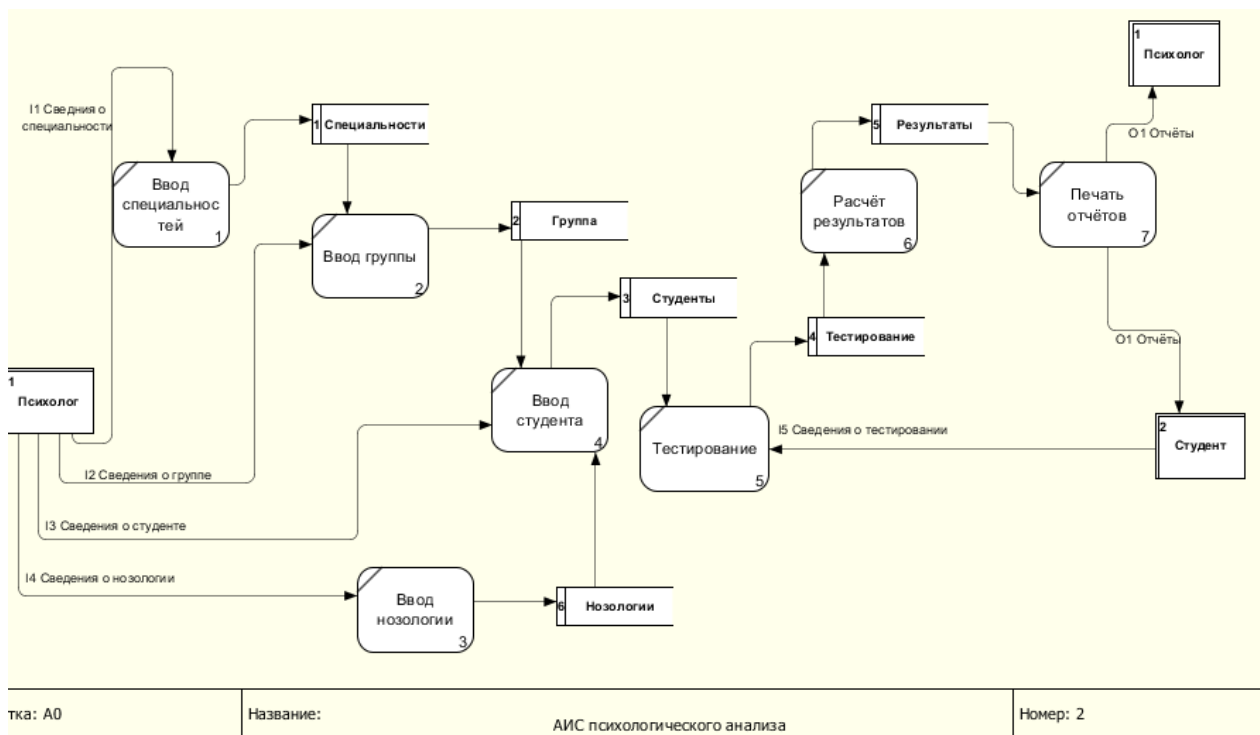


Рис. 1. DFD модель первого уровня

Здесь  $\sum_{i=1}^7 A_i$  – процессы системы, выполняющие преобразование и перераспределение входных информационных потоков, соответствующие базовым функциям: создание справочников специальностей, групп и нозологий ( $A_{1-3}$ ), загрузка данных об испытуемых ( $A_4$ ), тестирование ( $A_5$ ), обработка результатов тестирования ( $A_6$ ), анализ данных и печать отчетов по итогам тестирования ( $A_7$ ).

$\sum_{i=1}^6 D_i$  – система связанных хранилищ данных (data store), реализуемых в виде отдельных сущностей реляционной базы данных ( $D_1$ -  $D_6$ )

Второй компонент комплексной концептуальной модели информационной системы, включает ERD – модель, построение которой предполагает выделение сущностей и отношений (связей между ними) [6]. Структура реляционной базы данных представлена на рис.2.

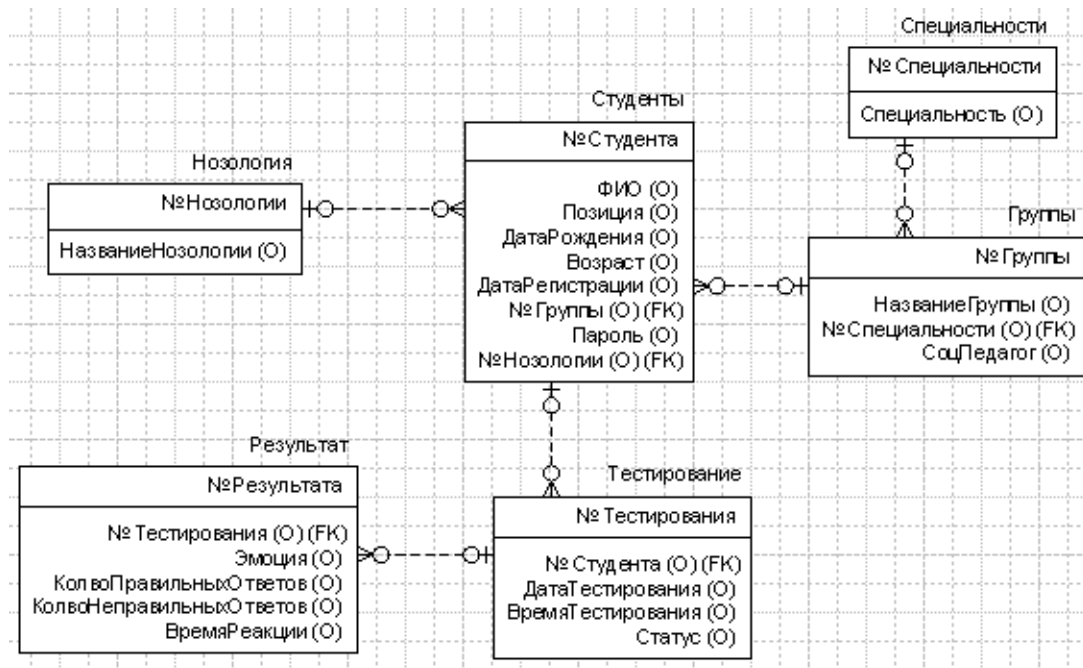


Рис. 2. Концептуальная модель базы данных (нотация Crows' feet)

**Выбор архитектуры и средств разработки.** Архитектура программного решения должна обеспечивать многопользовательский параллельный режим в пределах локальной вычислительной сети. Это достигается применением технологии клиент-сервер, модели удаленного доступа.

В модели удаленного доступа архитектура организована так, что база данных (БД) хранится на сервере, там же находится ядро системы управления базами данных (СУБД). Презентационная и бизнес – логика реализуется на клиентской стороне. Реляционный характер применяемой базы данных предполагает взаимодействие между серверной и клиентской частью приложения посредством структурированного языка запросов SQL. Клиентское приложение обеспечивает человеко-машинное взаимодействие посредством графического пользовательского интерфейса (GUI), а также реализацию модулей загрузки, обработки и анализа данных.

Требования к кроссплатформенности приложения реализуются за счет использования платформы .Net Core. Это открытая полностью автономная платформа, которая не требует зависимости от ядра систем на базе Windows, что позволяет запускать .Net - приложения на Unix – подобных системах без использования сторонних технологий (в частности, Mono) [7].

Интеграцию с базой данных обеспечивает технология ADO.Net (ActiveX Data Objects) – основная технология доступа к реляционным базам данных для платформы .Net, поддерживающая архитектуру «клиент - сервер». Объектную модель ADO.Net составляют 2 основных класса: Data Provider и Data Set [8].

Логика обработки данных реализуется на основе языка объектно-ориентированного программирования C#.Net 9.0, который являясь мощным языком, обладает рядом дополнительных преимуществ [9]: гибкость, ориентированность на объекты, типовая защищенность, «сборка мусора», «синтаксический сахар» и многое другое.

## Этап 2. Разработка и интеграция программного решения

Автоматизированная система строится как программный комплекс для пользователей двух типов: испытуемые (из числа обучающихся) и психологи, каждый

из которых получает соответствующие права после авторизации и входа в систему (рис.3).

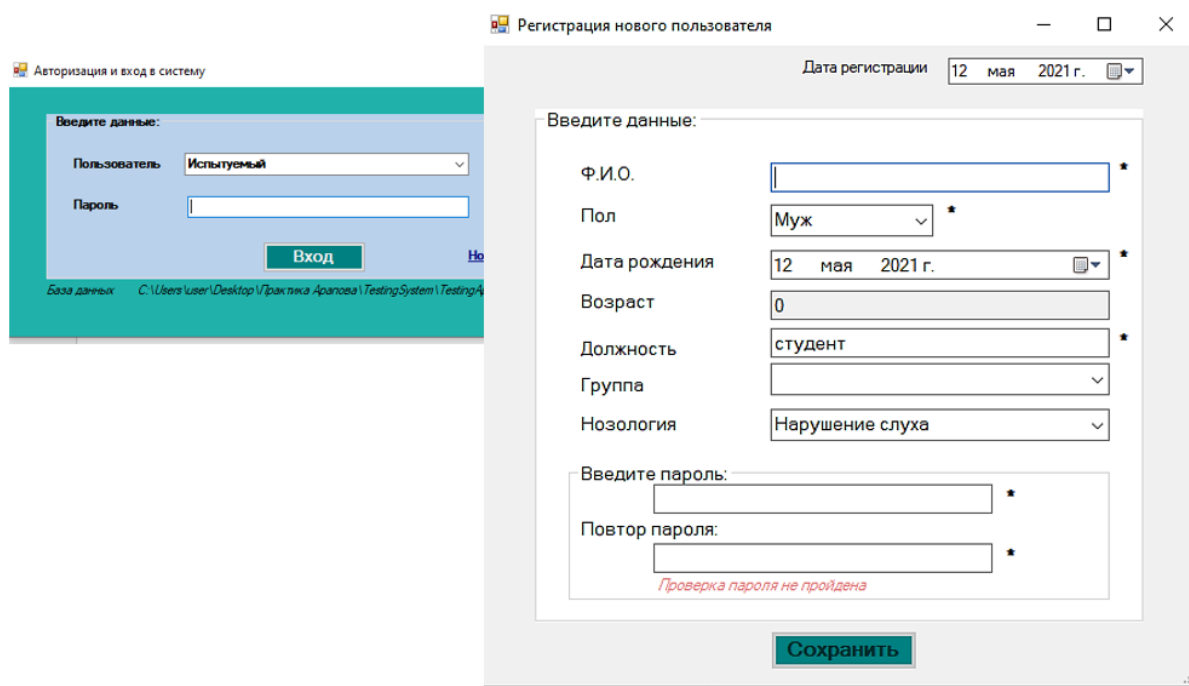


Рисунок 3 – Регистрация и авторизация пользователей

Доступ к данным выполняется на основе методов `DataGridBind(string ConnectionString, string sql, DataGridView dgv)`, `ComboBoxBind(ConnectionString, string sql, ComboBox cb)`, `execStProcedur(string procName)` созданного класса `ClassAdo`, которые через провайдера `System.Data.SqlClient` выполняют подключение к базе данных `SqlServer`, выборку и привязку данных, а также вызов и выполнение серверных хранимых процедур(рис.4). Графический интерфейс программы формируется на основе шаблонов `WindowsForms`.

```
Ссылка 4
public DataSet GetDataSet(string sqlQuery) //получить набор данных по sql-запросу
{
    SqlDataAdapter adapter = new SqlDataAdapter(sqlQuery, connectionString);
    DataSet ds = new DataSet();
    adapter.Fill(ds, "Table");
    return ds;
}

Ссылка 6
public void DataGridBind(string sqlQuery, DataGridView dataGridView) //"привязать" набор данных к DataGridView
{
    DataSet ds = GetDataSet(sqlQuery);
    dataGridView.DataSource = ds.Tables[0].DefaultView;
}
```

Рисунок 4 – Фрагмент листинга программного кода `ClassAdo`

Программное решение интегрирует несколько модулей для автоматизированной психодиагностики по четырем базовым методикам тестирования ЭИ.

**Модуль диагностики агрессии по методу Вагнера.** Алгоритм тестирования по методу Вагнера предлагает испытуемому произвольные изображения руки. Интерфейс соответствующего модуля предлагает несколько полей для ввода действий, которые, по мнению обучающегося, выражает эта рука (рис. 5).

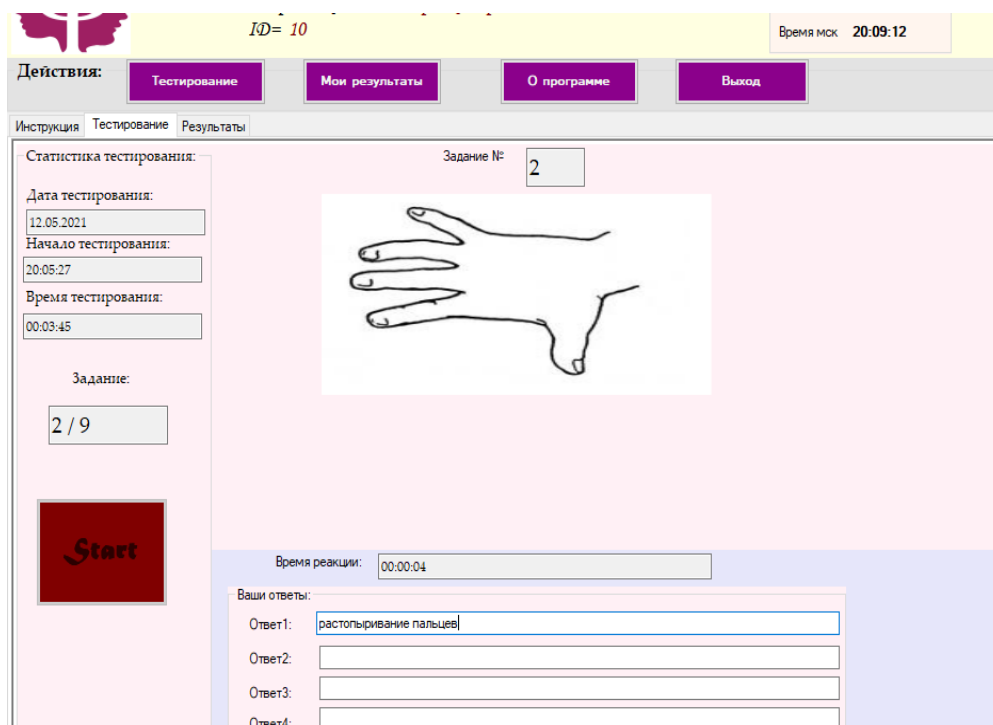


Рисунок 5 – Интерфейс испытуемого по методу «руки» Вагнера

Модуль психолога позволяет выбрать для обработки данные текущего тестирования и указать эмоциональные характеристики ответа (рис. 6).

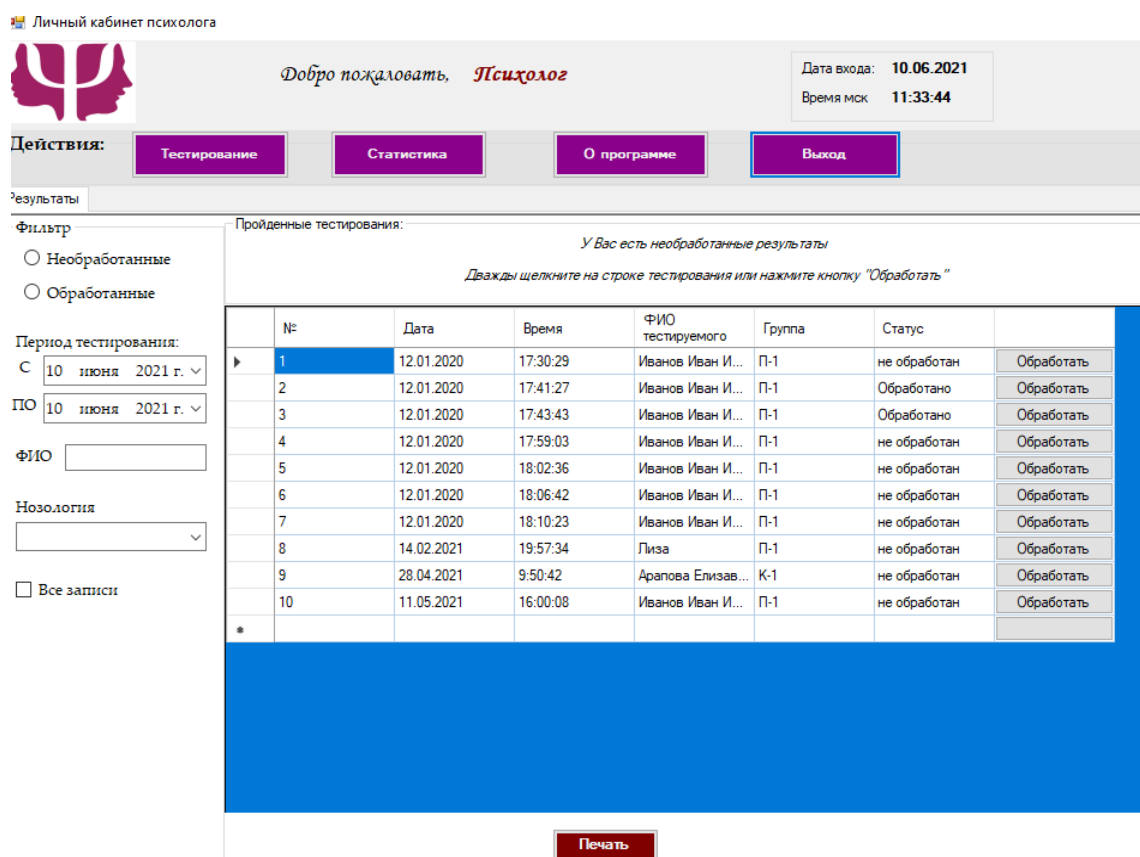


Рисунок 6 – Интерфейс Психолога для обработки данных тестирования

Процесс «ручной» обработки ответов испытуемого показан на рис. 7.

Обработка результатов тестирования

Данные тестирования: №-тестирования  Дата  Время  Статус

Данные испытуемого: ФИО  Дата рождения:  Возраст  Пол  Должность  Группа  Нозология

Обработка результатов тестирования

Соотнесите ответ испытуемого с одной из категорий в выдающемся списке и нажмите кнопку

ID	№вопроса	Время реакции	№ ответа	Ответы	Результат
2	1	00:00:08	1	приветствует	Агрессия
			2	просит	Указание
			3	протягивает	Страх
3	2	00:00:09	1	опирается	Эмоциональность
			2	пожимает	Коммуникация
			3	показывает	Зависимость
4	3	00:00:12	1	выпрямляет	Демонстративность
			2		Увечность
			3		Активная безличность
5	4	00:00:03	1	указывает	Пассивная безличность
			2	угрожает	Описание
			3	показывает	
			4	направляет	
6	5	00:00:05	1	берет	
			2		
			3		
7	6	00:00:25	1	пожимает	
			2		

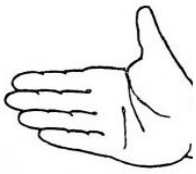


Рисунок 7 – Обработка ответов испытуемого

Итоговый результат содержит подсчет количество эмоций, позволяющий определять уровень агрессивности обучающегося (рис.8)

Результат	Итоговый результат
Агрессия	Агрессия-5
Коммуникация	Указание2
Демонстративность	Страх-0
Коммуникация	Эмоциональность 1
	Коммуникация4
	Зависимость-0
	Демонстративность4
	Активная безличность-1
	Пассивная безличность-0
	Описание-0
Зависимость	
Указание	
Указание	
Указание	
Указание	
Пассивная безличность	
Коммуникация	

Рисунок 8 – Результаты теста Вагнера

**Модуль оценки ЭИ по распознаванию лиц.** Испытуемому предлагается для рассмотрения серия фотографий лиц (мужских и женских), выражающих одну из 7-ми эмоций: гнев (злость), нейтральное, отвращение (брезгливость), печаль (грусть), радость (восторг), страх (ужас), удивление (шок). В каждом ряду испытуемый должен отметить то лицо, которое, по его мнению, выражает заданную эмоцию. В частности,

на рис. 9 показан пример, выбора изображения, которое соответствует нейтральному выражению лица. Все изображения загружаются случайным образом так, чтобы выполнялись следующие условия:

- в каждом из 4-х рядов выводится семь случайно отобранных фотографий, каждая из которых представляет одну из 7-ми базовых эмоций;
- испытуемой может выбрать только одну из фотографий в каждом ряду (множественный выбор запрещен) одним щелчком мыши или снять выделение повторным щелчком;
- для каждой эмоции в базе данных хранится 60 (30 мужских и 30 женских) фотографий, которые «рандомным» образом выбираются системой;
- при загрузке фотографии принадлежность к определенному полу (мужской или женский) определяется программой случайно, но таким образом, чтобы за весь процесс тестирования количество предлагаемых мужских и женских лиц было одинаково;
- эмоция, которая проверяется в каждом задании, определяется также «рандомно», но так, чтобы на тестирование каждой из семи эмоций приходилось по 6 заданий (общее количество задач теста, таким образом, равно 42.) (рис.9).

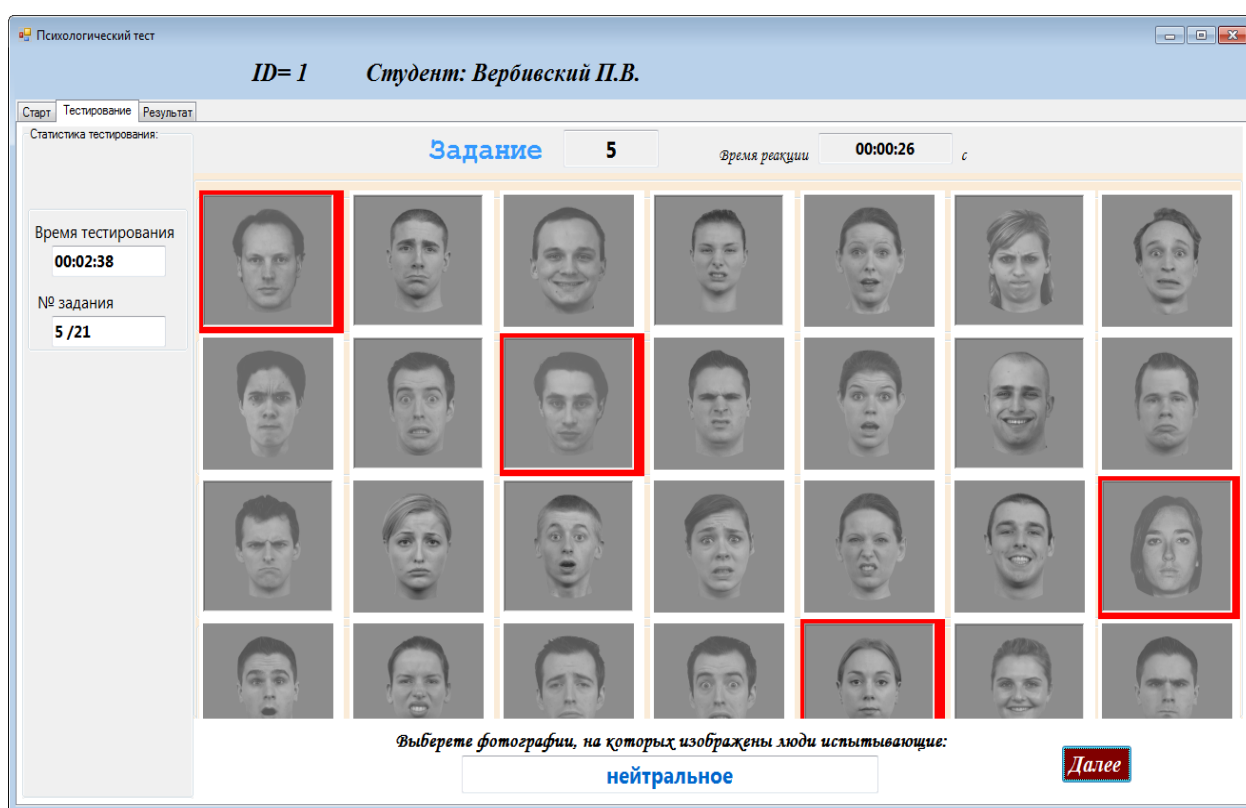


Рисунок 9 – Тестовое задание на распознавание эмоции

Один из вариантов реализации методики предлагает для испытуемого выбор эмоции из двух фотографий. В этом случае система определяет скорость реакции, которая является важным диагностическим фактором (рис. 10).

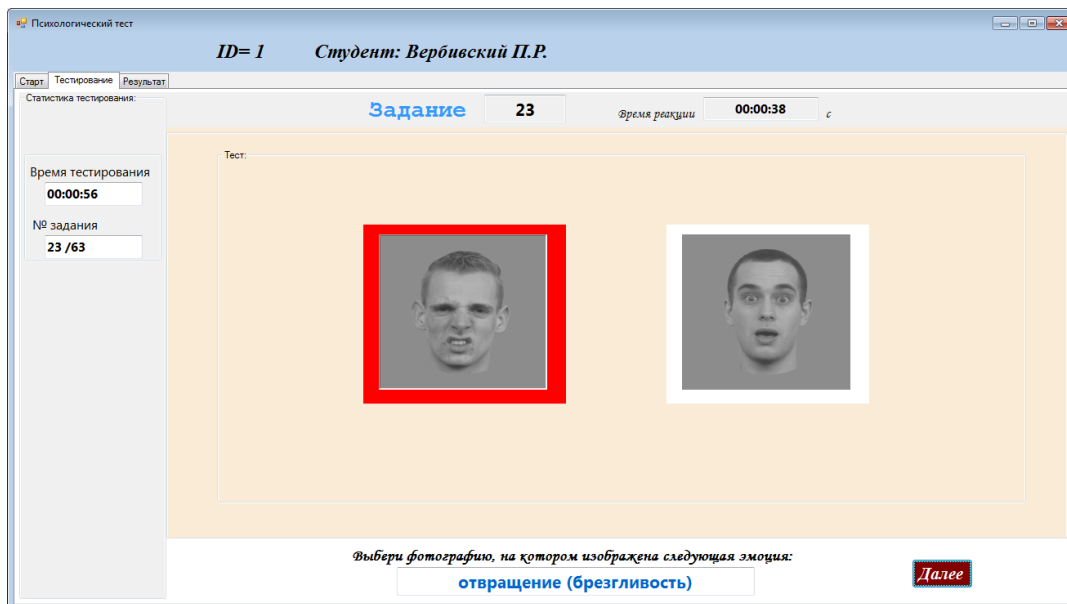


Рисунок 10 – Выбор эмоции из двух фотографий

По итогам тестирования система формирует отчет, включающий анализ верных и неверных ответов для каждой эмоции, а также среднее время ее поиска (в сек.). Для психолога такой анализ расширен сводными данными в разрезе специальностей, групп и нозологий (рис. 11).

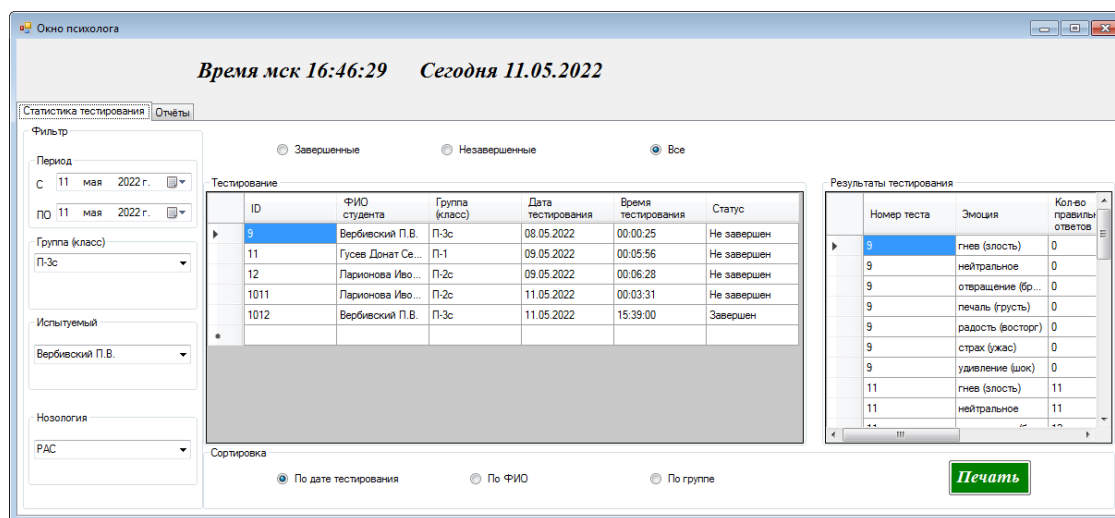


Рисунок 11 – Анализ результатов психодиагностик для психолога

**Опросник Холла.** Тест по опроснику Холла включает 30 утверждений и содержит 5 шкал: 1) эмоциональная осведомленность, 2) управление эмоциями 3) самомотивация 4) эмпатия 5) распознавание эмоций других людей.

Ответ на каждое утверждение предполагает 6-балльную шкалу: от -3 – «полностью не согласен» до +3 – «полностью согласен». По каждой шкале программа рассчитывает сумму баллов с учетом знака ответа (+ или -) и интерпретирует полученные результаты (чем больше плюсовая сумма баллов, тем больше выражено данное эмоциональное проявление) (рис. 12).

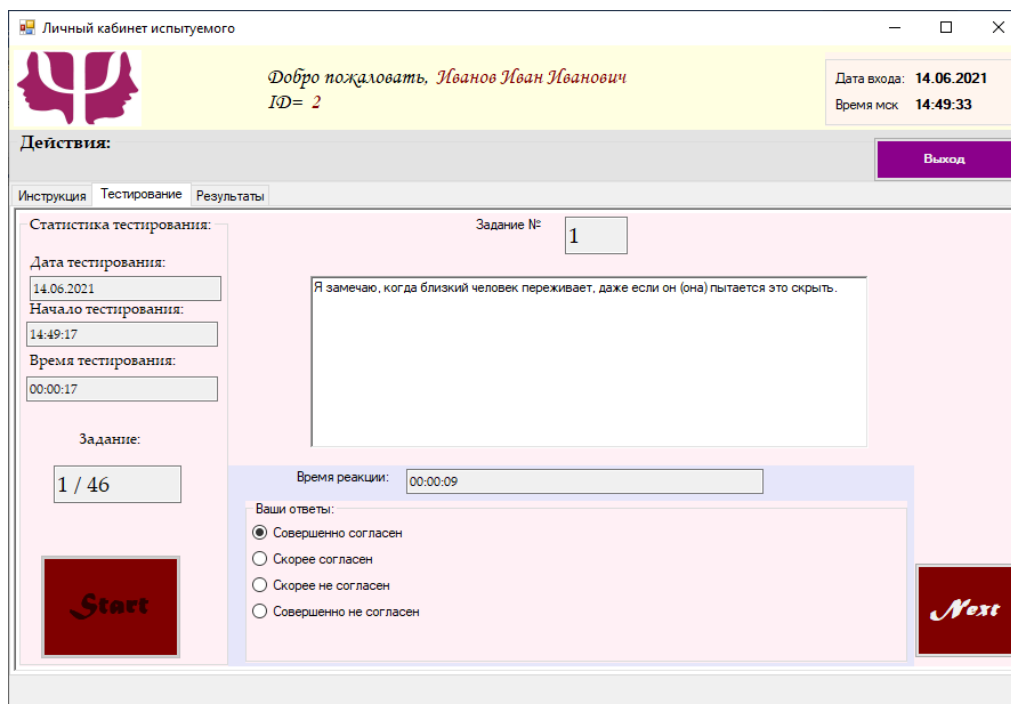


Рисунок 12- Тестирование по опроснику Холлла

Анализ результатов по итогам тестирования включает суммарные баллы по каждому обучающемуся, а также сводные итоги по учебным и нозологическим группам (рис. 13)

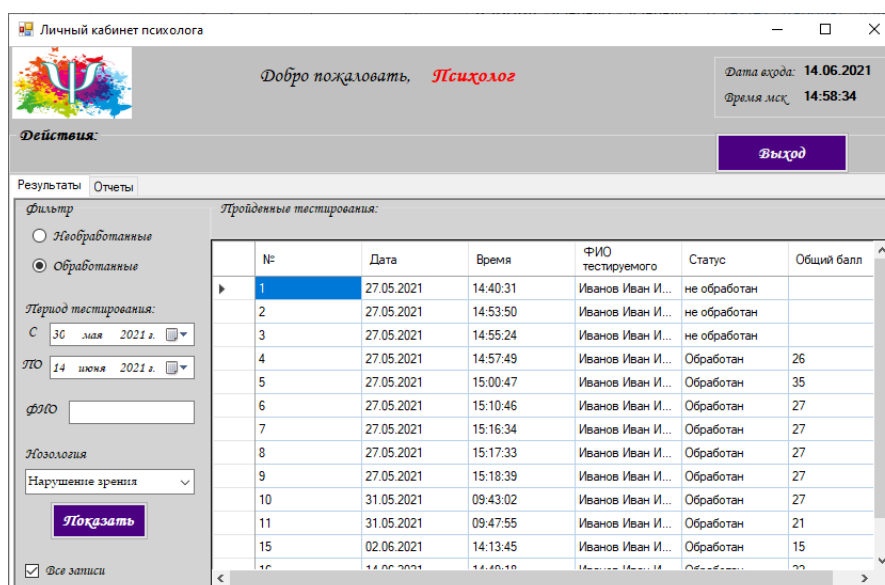


Рисунок 13 – Результаты теста Холлла

## ЗАКЛЮЧЕНИЕ

В работе представлен процесс анализа, проектирования и разработки интегрированного программного продукта, который включает комплексное решение по автоматизации психологического тестирования уровня эмоционального интеллекта.

Программный продукт содержит графический пользовательский интерфейс, интуитивно понятный и доступный пользователю любой ИТ-квалификации. Клиент-серверная архитектура позволяет обеспечить массовую диагностику с одновременным опросом группы обучающихся. По итогам тестирования система выполняет



статистический анализ полученных результатов, который может использоваться для психокоррекции и дальнейшей психологической работы с обучающимися. Анализ динамики результатов тестирования позволяет оценить эффективность и результативность коррекционной работы, оперативно принимать решения по ее совершенствованию.

#### Список использованных источников:

1. Ларина Альмира Тимербаевна Эмоциональный интеллект // АНИ: педагогика и психология. 2016. №3 (16).
2. Князева Т. С. Измерение эмоционального интеллекта у музыкантов и художников с помощью методики MSCEIT V. 2. 0 // Психология. Журнал ВШЭ. 2010. №3.
3. Методика Н. Холла оценки «эмоционального интеллекта» (опросник EQ). URL: <https://andreeva.by/metodika-ocenki-emocionalnogo-intellekta-oprosnik-eq-n-xolla.html> (дата обращения 30.11.2022)
4. Тест «Руки» Вагнера. URL: <https://testograd.com/test/ruki-vagnera/> (дата обращения 30.11.2022)
5. Калянов Г. Н. Концептуальная модель DFD-технологии // Открытое образование. 2017. №4.
6. Романов С.С. Об инфологическом моделировании баз данных с помощью нормализации ER-диаграмм // Таврический научный обозреватель. 2017. №1 (18).
7. Макаров Олег Сергеевич, Щенникова Елена Владимировна СРАВНИТЕЛЬНЫЙ АНАЛИЗ ТЕХНОЛОГИИ ASP.NET CORE // E-Scio. 2020. №7 (46).
8. ADO.Net. URL: <https://docs.microsoft.com/ru-ru/dotnet/framework/data/adonet/> (дата обращения: 20.07.2022).
9. Сарсимбаева Сауле Мусаевна, Саймагамбетова Азиза Жанболатовна Применение языка программирования С# и платформы. Net 4. 0 для объектно-ориентированного моделирования // Объектные системы. 2012. №1 (6).

## ЯЗЫКИ ПРОГРАММИРОВАНИЯ НА РУССКОМ ЯЗЫКЕ

*Шевченко Данила Александрович  
ФКПОУ «НТТИ» Минтруда России  
Шепелева Ирина Владимировна, преподаватель*

**Аннотация.** В статье представлен обзор, краткая история разработки, особенности синтаксиса некоторых языков программирования на русском языке. Целью работы является знакомство с разнообразием отечественных (советских и российских) языков программирования, имеющих русский синтаксис. Для подготовки статьи использованы фото и видеоматериалы, статьи, размещённые в свободном доступе в интернете.

В 1960–1970-е годы было довольно много языков программирования на русском языке в профессиональной среде. Это были собственно советские разработки, например, Рапира, Дракон, Диполь, , и локализованные версии: например, русский Кобол.

Русский синтаксис имел встроенный интерпретатор «ИСКРА-1256» [1]. Например, команда безусловного перехода на языке ИСКРА-1256 выглядела так:

**1 ПЕРЕХ 180**

что означает «перейти к строке 180».

Команда вызова подпрограммы по значению заданного значения индекса выглядела так:

**1 ПП 4**

что означает – перейти к подпрограмме номер 4 с последующим возвратом к следующей строке, в данном случае – 2. Функциональная клавиша с номером 4 также вызывает выполнение этой подпрограммы.

*1.1.2. Текст программы для расчета кривой титрования основания (сильного или слабого) кислотой*

```
1 M12
2 СЕЛЕКТ(0, 1, 5)
3 ПЕЧАТЬ(HEX(03010A0A0A), HEX(0E), " КРИВАЯ ТИТРО-
  ВАНИЯ", HEX(0A), " ОСНОВАНИЯ КИСЛОТОЙ"),)
4 СЕЛЕКТ(1, 67, 9)
5 СЕЛЕКТ(2, 8, 100)
6 СЕЛЕКТ(0, 0, 0)
7 СЕЛЕКТ(3, 5, 64)
8 ПЕЧАТЬ(HEX(0 F 0301), "ВВЕДИТЕ КОНЦЕНТРАЦИЮ
  ОСНОВАНИЯ",)
9 H= > A01
10 ПЕЧАТЬ("КАКОВ ОБЪЕМ ОСНОВАНИЯ?"),)
11 H= > A02
12 ПЕЧАТЬ("КАКОВА КОНЦЕНТРАЦИЯ КИСЛОТЫ?"),)
13 H= > A03
14 I= > A08
15 I.5= > A10
16 ПЕЧАТЬ("ЕСЛИ ТИТРУЕМОЕ ОСНОВАНИЕ –
  СИЛЬНЫЙ ЭЛ-Т, НАЖМИТЕ 1, ЕСЛИ ТИТРУЕМ.
  ОСНОВАНИЕ – СЛАБОЕ, НАЖМИТЕ 2"),)
17 H= > A09
18 A09 < A10 ПЕРЕХ M01
19 ПЕЧАТЬ("ВВЕДИТЕ ЗНАЧЕНИЕ К ДИС."),)
20 H= > A11
21 M01
22 I= > I01
23 ЦИКЛ(I01, 1, 63)
24 "= " > STR (CA00, I01, 1)
25 ЦИКЛ*
26 M06
27 ПЕЧАТЬ("КУДА ВЫВЕСТИ РЕЗУЛЬТАТ (1 – НА ЭК-
  РАН ЭВМ, 2 – НА АЦПУ"),)
```

Рисунок 1 – Текст программы на языке Искра-1256 [2]

### **Куда пропали советские языки программирования Рефал, Эль и Рапира?**

О советских языках программирования сегодня говорят редко. Машины, которые их понимают, стали экспонатами технических музеев или решают секретные задачи для обороной промышленности в закрытых НИИ.

Необходимо отметить, что некоторые из этих ЭВМ не уступали по мощности более популярным зарубежным аналогам и содержали очень смелые идеи. Сделаем обзор самых знаменитых языков программирования, разработанных в СССР [3].

#### **Рефал**

В шестидесятые годы в промышленности и науке уже начали активно применять высокоуровневые языки. Учёные в США и Канаде писали программы на Fortran, в Европе и СССР — на Algol. Даже у бизнеса был свой язык программирования — COBOL. Все они поддерживали императивную парадигму: программист подробно описывал каждый этап вычислений с помощью переменных и операторов. Императивная парадигма считалась перспективной и единственно верной.

На этом фоне декларативный Рефал казался чем-то особенным. Его создал в 1966 году Валентин Фёдорович Турчин — математик, кибернетик, философ, правозащитник и поэт.

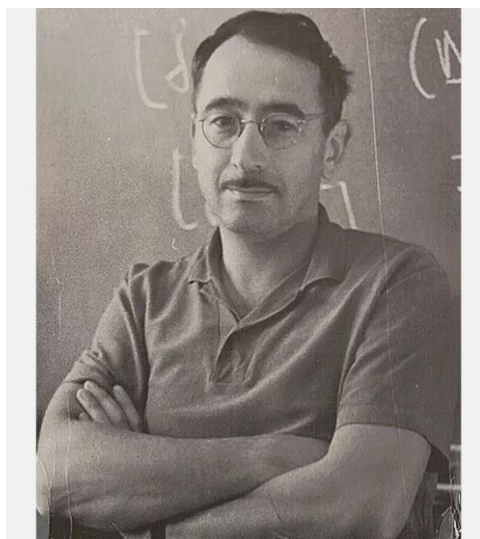


Рисунок 2 - Валентин Фёдорович Турчин [4]

Идеи В. Ф. Турчина были как будто из далёкого будущего — он опередил своё время на десятилетия и даже предсказал цифровое бессмертие в своём «Кибернетическом манифесте». Он любил мечтать и размышлять о высоком — даже язык программирования писал для потомков.

Валентин Турчин считал, что человек — это кибернетическая система, а сознание — управляющая программа. Наши кожа, кости и другие ткани постоянно обновляются. А значит, когда-нибудь люди заменят этот материал на кремний или металл, а сознание будет жить неограниченно долго.

Чтобы понять концепцию языка программирования Рефал, необходимо понять философские воззрения его создателя. Для него эволюция — это череда переходов от мелких систем к более сложным, высокоуровневым. Они воспроизводят системы предыдущего уровня, а потом, в свою очередь, объединяются в системы более высокого порядка. Валентин Турчин назвал эти превращения «метасистемными переходами» [3].

Пример из промышленности: раньше у нас были строительные инструменты, потом появились станки для производства инструментов, а потом сделали ПО для проектирования таких станков.

Пример из ИТ: раньше сайты писали на чистом HTML и CSS, потом появились конструкторы сайтов, а затем — no-code-платформы вроде Bubble, на которых можно создавать сами конструкторы сайтов.

В программировании В. Ф. Турчин видел ту же тенденцию. Он *создавал Рефал как язык для описания языков — надстройку над всеми языками* вроде Fortran, Algol и BASIC. На Рефале можно было создавать новые языки для решения узких задач. Например, В. Турчин написал транслятор с Algol 60 на язык ассемблера.

Валентин Турчин заложил в Рефал прорывные идеи: функциональное программирование, рекурсия и перехват исключений — позже они разошлись по другим языкам. Но советские инженеры не особо задумывались о миссии языка и метасистемных переходах, поэтому писали на Рефале преобразования алгебраических выражений и переводчики для естественных языков [3].

Рефал до сих пор жив и развивается. Он стал тем самым метаязыком, на котором учёные создают новые языки программирования. Но не удивляйтесь, если не знаете ни одного из них: эти языки создают специально для военных компьютеров. Информацию о Рефале можно найти в русскоязычном «комьюнити».

## Язык программирования Эль-76

В семидесятые-восемидесятые годы группа советских инженеров разработала серию суперкомпьютеров «Эльбрус». Они параллельно обрабатывали несколько инструкций и даже обзавелись встроенными типами данных. В то время у операционных систем и языков программирования не было типов — и из-за этого постоянно возникали ошибки вроде деления символа на число. Но главной фишкой «Эльбруса» стала работа с высокоуровневыми языками.

Специально для «Эльбруса» учёный Владимир Пентковский с сотрудниками Института точной механики и вычислительной техники разработал язык программирования Эль-76. Он был высокоуровневым, но его часто называли *автокодом*, потому что языка более низкого уровня в «Эльбрусе» не было. Программы на всех остальных языках транслировались в Эль-76.



Рисунок 3 - Владимир Мстиславович Пентковский. Фото: «Википедия»

**Эль-76** был полноценным языком программирования с «джентльменским набором» возможностей и операторами на русском языке (рисунок 4).

Что было в этом языке программирования:

- структурное программирование;
- циклы;
- условные операторы;
- оператор case;
- обработка исключений.

```

выбор день из
1 : , 3 : , 7 : печать (стр8"дневная смена"),
2 : , 4 : печать (стр8"вечерняя смена"),
5 : , 6 : печать (стр8"выходной день")
иначе печать (стр8"неверно задан день")
всевыб

процедура последень = функция (год , месяц , день)
день = выбор месяц из
    1 : , 3 : , 5 : , 7 : , 8 : , 10 : , 12 : 31 ,
    4 : , 6 : , 9 : , 11 : 30 ,
    2 : если год остат 4 = 0 и
        год остат 100 <> 0 или
        год остат 400 = 0 то
        .29 % високосный год
        иначе 28 все
иначе 0 всевыб)

```

Рисунок 4 - Пример кода на Эль-76. Фото: Public Domain

Инженерам нравился Эль-76 за встроенные типы данных. Современным программистам трудно понять, насколько это было новаторским: сейчас типы есть уже во всех языках. А раньше программист заботился о том, чтобы машина чётко поняла, с чем имеет дело: целым числом, символом или логическим выражением. Если вдруг машина начинала перемешивать типы данных, как ей вздумается, и без предупреждений выдавать непредсказуемый результат, найти источник таких ошибок было очень трудно.

Идея русскоязычного программирования была заманчивой и идеально встраивалась в политический курс СССР, но история и экономика распорядились иначе.

Сегодня не осталось ни одной машины, которая «разговаривает» на «Эль-76»: «Эльбрусы» производились для внутреннего рынка, а IBM, Microsoft и Apple продавали компьютеры по всему миру. Под их «железо» писали программы на Fortran, BASIC и C. Поэтому Эль-76, как и многие другие кириллические языки программирования, прекратил существование [3].

### Учебный язык программирования Робик.

Этот язык со смешным названием придумали в 1975 году основоположник советской школьной информатики Геннадий Анатольевич Звенигородский и академик Андрей Петрович Ершов для обучения основам программирования детей в возрасте 8—11 лет. «Робик» использовался в Харьковской «Школе юных программистов», занятия в которой проводил сам Г.А. Звенигородский. В 1977 году разработчик языка переехал в Новосибирск, где устроился на работу в Вычислительный центр Сибирского отделения АН СССР, после чего уроки для юных программистов продолжились уже в этом городе в формате воскресной школы и летних образовательных лагерей [5].

В основе синтаксиса «Робика» лежит русский язык, хотя авторы утверждали, что относительно несложно переработать «Робик» для использования других языков народов СССР. Набор базовых конструкций «Робика» был весьма ограничен, однако его возможности значительно расширились за счёт так называемых «исполнителей» — объектов, выполняющих задаваемые программистом команды.

Каждый исполнитель действовал в определённой для него среде, состоящей, в свою очередь, из элементов, которые могут находиться в разных состояниях.

Для каждого исполнителя существует свой набор предписаний (команд). Прежде чем школьник начнёт отдавать команды исполнителю, необходимо запустить хотя бы один экземпляр этого исполнителя и присвоить ему уникальное имя.

Делалось это с помощью команды ВКЛЮЧИТЬ, например, так:

```
ВКЛЮЧИТЬ ОБХОДЧИК:МИША;
```

Эта команда запускала экземпляр исполнителя «ОБХОДЧИК», которому присваивалось уникальное имя «МИША». Соответственно, завершалась программа командой ВЫКЛЮЧИТЬ. «ОБХОДЧИК» — это стандартный исполнитель из комплекта программ «Школьница», предназначенный для решения простых математических задач. Среда «ОБХОДЧИКА» представляет собой железнодорожную ветку определённой длины, состоящую из некоторого количества стыков. «ОБХОДЧИК» может двигаться по этой железной дороге вперёд или назад, если путь перед следующим шагом свободен (там есть рельсы), и проверять стыки.

Как работает этот исполнитель, можно понять на примере простой математической задачи. Имеется прямая железнодорожная ветка неизвестной длины, состоящая из некоторого количества стыков. Расстояние между стыками составляет 24 метра. «ОБХОДЧИК» стоит в начале этой железнодорожной ветки. Требуется проверить все стыки, найти неисправные, а затем вернуть «ОБХОДЧИКА» в начало ветки. Программа на языке «Робик» для решения этой задачи выглядит следующим образом (рисунок 5):

```
ВКЛЮЧИТЬ ОБХОДЧИК;  
ЗАПОМНИТЬ ПРОЦЕДУРУ ПРОВЕРКА_СТЫКА;  
    ЕСЛИ ОБХОДЧИК.СТЫК = "НЕИСПРАВЕН" ТО ОТМЕНИТЬ СТЫК;  
    ВПЕРЕД 24М;  
ЗАКОНЧИТЬ;  
ПОКА ОБХОДЧИК.ВПЕРЕДИ = "СВОБОДНО" ПОВТОРЯТЬ ПРОВЕРКА_СТЫКА;  
ПОКА ОБХОДЧИК.СЗАДИ = "СВОБОДНО" ПОВТОРЯТЬ НАЗАД 24М;  
ВЫКЛЮЧИТЬ ОБХОДЧИК;
```

Рисунок 5 - Программа на языке «Робик» для решения задачи проверки неисправных стыков

Результат выполнения этой программы будет наглядно продемонстрирован на дисплее «Агата»: пользователь увидит и железнодорожную ветку (пунктирную линию), и движущегося по ней «обходчика», и счётчик неисправных стыков.

Именно поэтому результат работы этой программы, например, количество выявленных дефектов и пройденное расстояние, не сохраняются в переменные и нигде не выводятся. Эти значения записываются в память внутри самого исполнителя, инициализируются автоматически и обнуляются при выключении экземпляра исполнителя — что значительно упрощает процесс программирования для детей. Им уже не нужно думать об объявлении переменных, об их типах, а вместо этого можно сосредоточиться на решении самой практической задачи.

Позже «Робик» был переработан и включён в состав пакета образовательных программ «Школьница» для ПК «Агат», которыми тогда оснащались многие компьютерные классы в советских учебных заведениях.

Предполагалось, что после изучения «Робика» дети смогут перейти на более сложный учебно-производственный язык «Рапира».

### **Учебный язык программирования Рапира.**

В 1985 году предмет «Основы информатики и вычислительной техники» вошёл в основную школьную программу, а в советских школах и институтах начали обучаться программированию на языке программирования Рапира: расширенный адаптированный поплан-интерпретатор, редактор, архив. Он был предназначен для

перехода от простого «Робика» к более сложным языкам высокого уровня, базировался на синтаксисе Поплан и Сетл, и тоже использовал для записи команд кириллицу [5].

До 1985 года о языке знали только читатели научного журнала «Квант» и его «Заочной школы программирования», а создали язык программирования Рапира в начале восьмидесятых выпускники Новосибирского государственного университета под руководством Геннадия Звенигородского (рисунок 6).

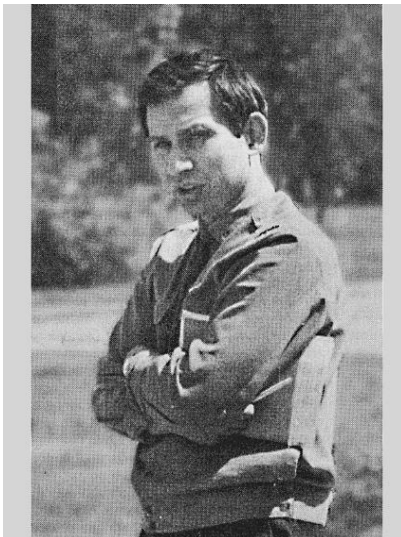


Рисунок 6 - Геннадий Звенигородский, один из создателей языка программирования «Рапира». Фото: архив академика А. П. Ершова

Как и в Эль-76, операторы в Рапире — слова русского языка (рисунок 7). При этом у «Рапиры» были и «локализации» — версии на английском, эстонском и грузинском. Школьники и студенты первых курсов писали на Рапире прикладные программы:

- калькуляторы;
- алгоритмы поиска и сортировки файлов;
- простые игры вроде костей;
- алгоритмы решения математических задач с помощью рекурсии.

```
247 ПРОЦ ВПЕРЕД();
248 ЕСЛИ ПЕРЕДСОСТАВОН(<)= "ТУПИК" ТО
249   ВЫВОД: "??НЕ МОГУ, НЕКУДА?";
250   ВЫХОД
251 ИНАЧЕ
252   <*>->ВОЗВСТР;
253   ОКНОДАИ(КАРТ);
254   ШАГСОСТАВА(<);
255   ПОКА ПЕРЕДСОСТАВОН(<)= "СВОБОДНО"::
256     ШАГСОСТАВА(<)
257   ВСЕ;
258   ДЛЯ И ИЗ ВОЗВСТР::
259     ПУТИСИ, 9J->ПУТИСИ, 8J;
260   ВСЕ;
261   ДЛЯ И ОТ 2 ДО #ПУТИ::
262     ЕСЛИ ПУТИСИ, 1J= "СТРЕЛКА" ТО
263       КРАСЬСТРЕЛК(И)
264     ИНАЧЕ
265       ПОЗ(0, ПУТИСИ, 5J*4);
266       ВЫВОД БПС: "?", ФТЕКСТ(3*ПУТИСИ, 4J-1, "=", "<"
267     ВСЕ
268   ВСЕ;
269   ДЛЯ И ОТ 1 ДО #ВАГОНЫ::
270     РИСВАГ(И, 0)
271   ВСЕ;
272   ВСЕ;
273   ОКНОДАИ(ДИАЛ);
274 КНЦ;
275
276 ФУНК ПЕРЕДСОСТАВОН(<);
277 ИМЕНА: ПУТЬ, МЕСТО, Р;
278 ВАГОНЫ(ССОСТАВ(1, 4J->ПУТЬ;
```

Рисунок 7 - Пример кода на языке программирования «Рапира». Фото: Public Domain

К преподаванию «Рапиры» подошли серьёзно: детей учили не только составлять инструкции, но и искать ошибки. У «Рапиры» даже была трассировка вызова процедур — разработчик мог посмотреть состояние переменных на каждом шаге программы.

Но и «Рапира» безвременно был забыт. Московская школа программирования продвигала европейский Algol, а пару лет спустя его заменил BASIC, который в итоге уступил место Pascal. При таком подходе, несмотря на современность и мощь, у «Рапиры» просто не было шансов остаться даже на школьных компьютерах.

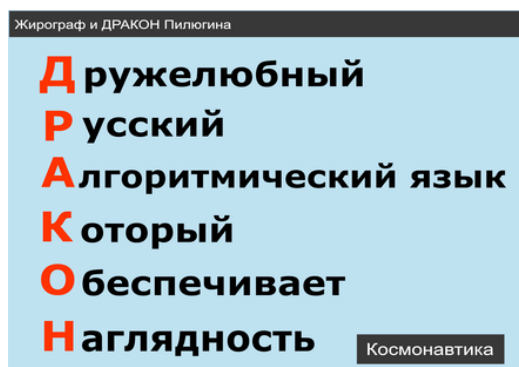
### **Почему советские языки программирования проиграли зарубежным.**

В СССР не экспортировали электронные вычислительные машины (ЭВМ) за границу, а ведь именно через них распространялись языки программирования. Поэтому советские языки проиграли зарубежным.

Программисты до сих пор пишут коды на языках Lisp и Erlang, а о более мощном Рефале вспоминают только наши отечественные учёные.

В СССР были и другие языки программирования: Сигма, Алмо или Бета. Но для решения *каких* задач их использовали, к сожалению, помнят только очень немногие в стенах исследовательских институтов или в архивах. *Заложенные в этих языках программирования идеи нередко опережали своё время.*

### **Визуальный алгоритмический язык программирования ДРАКОН**



Дружелюбный русский алгоритмический язык, который обеспечивает наглядность (сокр. ДРАКОН) — визуальный алгоритмический язык программирования и моделирования [8].

Язык построен за счёт формализации и эргономизации блок-схем алгоритмов, описанных в ГОСТ 19.701-90 и ISO 5807-85. ДРАКОН может быть использован *для разработки программ реального времени.*

Правила языка ДРАКОН по созданию диаграмм разрабатывались с учётом требований эргономики, то есть изначально оптимизированы для восприятия алгоритмов человеком с использованием технологий компьютерной графики. При этом, данный язык рассчитан на создание программ, которые можно было бы просматривать как модели, содержащие код на текстовом языке.

Возможности языка ДРАКОН могут расширяться в зависимости от нужд пользователя: на языке ДРАКОН можно писать программы для ЭВМ за счёт включения в себя функционала и синтаксиса поддерживаемого информационной системой (ИС) ДРАКОН или DRAKON Editor текстового языка программирования. Программа для ЭВМ, написанная таким образом, считается написанной на гибридном языке ДРАКОН-[название языка].

Вышеописанный подход повышает эргономику, особенно в ситуации, когда программа пишется продолжительное время и/или коллективно, создавая обстановку, позволяющую писать программу, имея в процессе её написания её же модель, которая облегчает возможность ориентироваться внутри исходного кода.

*Программа, считающаяся написанной на «чистом» языке ДРАКОН, является моделью поведения.*

Графический язык ДРАКОН похож на блок-схемы, но, строго говоря, это не блок-схемы. Разработка языка ДРАКОН и системы программирования началась в 1986 году.

Через 11 лет на базе ДРАКОНа была построена автоматизированная Технология разработки алгоритмов и программ (CASE-технология) под названием «ГРАФИТ-ФЛОКС».



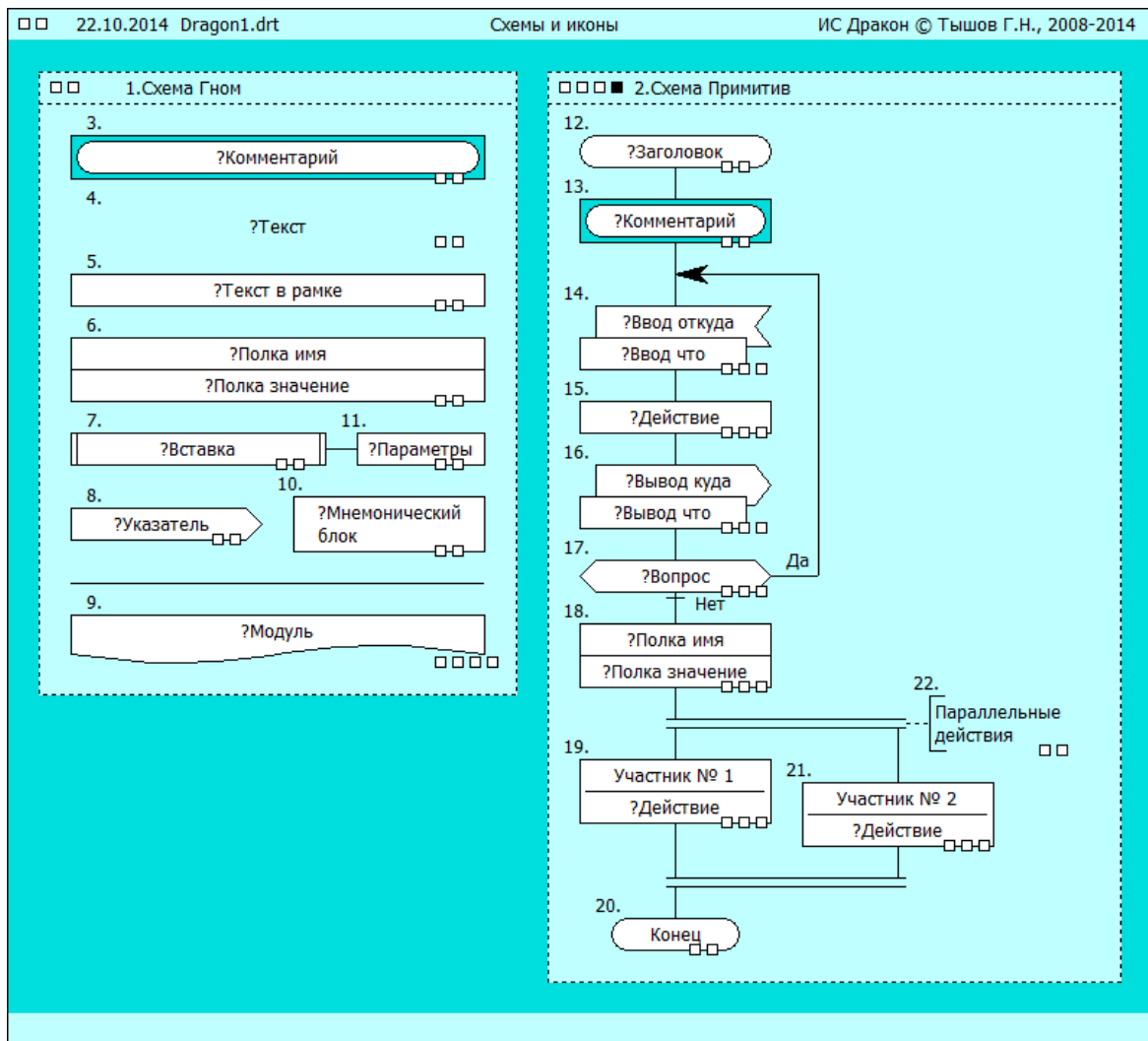


Рисунок 8 - Дракон-схемы: графический язык программирования

Язык ДРАКОН и система ГРАФИТ-ФЛОКС поступили в эксплуатацию. С их помощью были разработаны многие алгоритмы и программы разгонного блока космических аппаратов ДМ-SL Международного проекта «Морской старт». В общей сложности на разработку и отработку программного обеспечения и других элементов системы управления ушло три года. К 1999 году все работы были закончены. Система была готова к старту.

Первый пуск ракетного комплекса «Морской старт» состоялся 28 марта 1999 года. Этот пуск был ответственным испытанием языка ДРАКОН и технологии «ГРАФИТ-ФЛОКС». Он продемонстрировал их эффективность и надежность. С тех пор по программе «Морской старт» проведено свыше 30 ракетных пусков.

Язык ДРАКОН используется и в других космических программах. Например: разгонный блок космических аппаратов «Фрегат»; модернизированная ракета-носитель тяжелого класса «Протон-М»; разгонный блок космических аппаратов ДМ-SLB (проект «Наземный старт»); разгонный блок космических аппаратов ДМ-03; ракета-носитель легкого класса Ангара 1.2; ракета-носитель тяжелого класса Ангара-A5 и др.

Результаты использования ДРАКОНа были стабильно высокими, поэтому руководство «Научно-производственного центра автоматики и приборостроения имени академика Н. А. Пилюгина» (АО «НПЦАП») приняло решение об использовании ДРАКОН-технологии в последующих проектах.

DRAKON - это свободный и открытый язык алгоритмического визуального программирования и моделирования с открытым исходным кодом, разработанный в рамках космического проекта «Буран» [7].

Русские и другие национальные локализации имеет современный язык программирования **Лого**, предназначенный для обучения детей в младших классах школы.

Алгоритмический язык **КуМир** применяется исключительно в образовательных целях: в целях обучения программированию. Обычно изучается в школах. В среде КуМир используется Русский алгоритмический язык (РАЯ). Ниже представлено, как реализуется на нём команда «если - то – иначе – все» (полное ветвление) (рисунок 9).

### Общий вид команды:

*если условие*  
*то действия 1*  
*иначе действия 2*  
*все*

```

1  алг ветвление
2  нач вещь x, y
3  ▪ вывод "Введите значение x= "
4  ▪ ввод x
5  ▪ вывод "Введите значение y= "
6  ▪ ввод y
7  ▪ если x>0
8  ▪   то y:=y+5
9  ▪   иначе y:=y-2
10 ▪ все
11 ▪ вывод "Новое значение y= ", y
12 кон
13

```

Рисунок 9 - Пример кода на алгоритмическом языке **КуМир**

В среде **ЛогоМиры** используются команды для исполнителя Черепашка на русском языке [8]. Интерфейс программной среды **ЛогоМиры** состоит из трёх окон: Рабочее поле с панелью Меню сверху, Панель инструментов слева Поле команд внизу (рисунок 10).

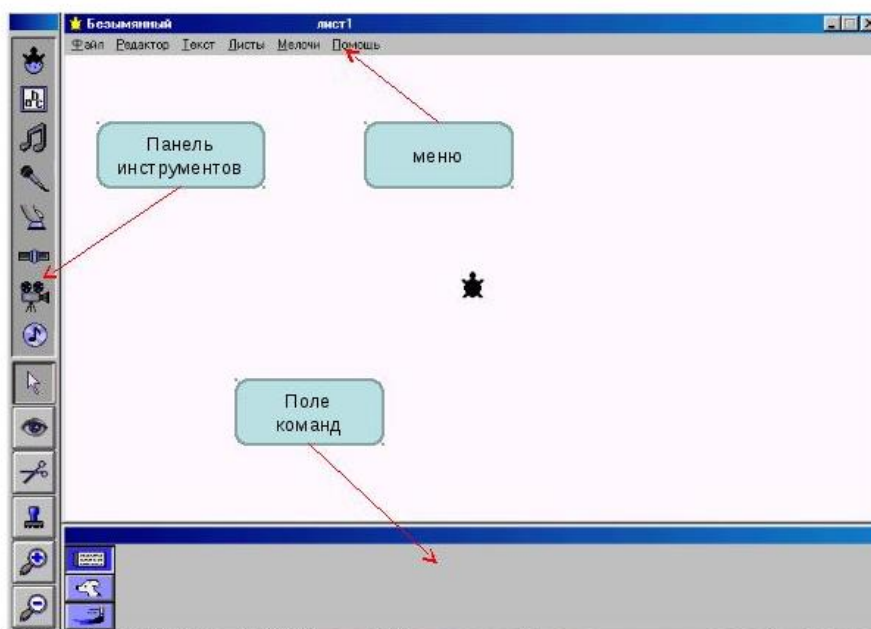


Рисунок 10 - Окно среды ЛогоМиры

На рабочем поле находится исполнитель Черепашка.

Пример последовательности команд в **ЛогоМирах**:

по повтори 3 [вп 100 пр 120]

Черепашка начертит правильный треугольник (рисунок 11).

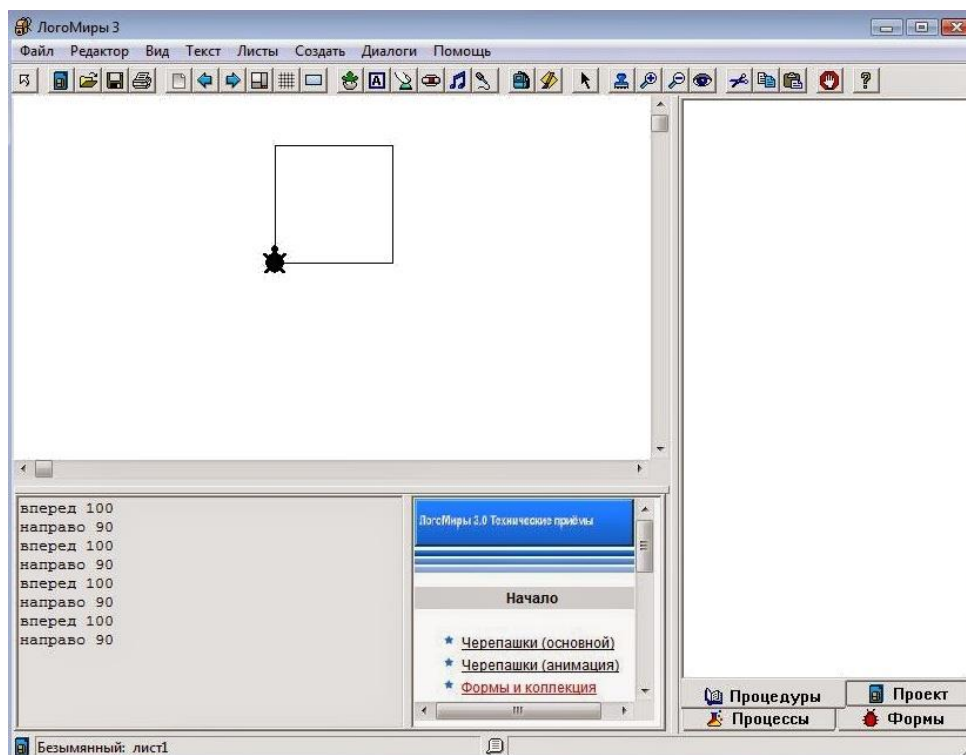


Рисунок 11 - Рабочая область среды ЛогоМиры 3

Существует и активно развивается русский учебный язык программирования **RUbasic**. Он имеет современный читаемый синтаксис и заявляет широкий функционал.

Собственно, для задач обучения - это единственный способ что-то сделать в текстовом программировании.

Программы на **RUbasic** вполне читаемы и понимаемы. На рисунке 12 представлен код на RUbasic.

```
Если Время.Часы < 12 То
    Консоль.ПечатьСтроки ("Доброе утро")
АЕсли Время.Часы < 16 То
    Консоль.ПечатьСтроки ("Добрый день")
АЕсли Время.Часы < 20 То
    Консоль.ПечатьСтроки ("Добрый вечер")
Иначе
    Консоль.ПечатьСтроки ("Доброй ночи")
Дальше
```

Рисунок 12 – Пример кода на RUbasic

Советские технологии образования и учебные языки программирования давно ушли в прошлое: их сменили Бейсик, Pascal, которые в последние годы потеснили Scratch и Python — их изучением занимаются современные школьники на уроках информатики. Но в начале 90-х Научно-исследовательский институт системных исследований РАН представил собственную разработку: язык и систему программирования для начальных курсов информатики «КуМир» (Комплект Учебных Миров). Примечательно, что «КуМир» основывается на учебном алгоритмическом языке А. П. Ершова (РАЯ) и включает ряд идей, воплощенных ранее в проекте «Е-практикум». В 2018 году разработчики выпустили вторую версию «КуМир». Версии редактора этого языка, распространяемого под лицензией GNU GPL 2.0, есть для Windows и для Linux [3].

«КуМир» был рекомендован Министерством образования и науки РФ в качестве основного учебного материала по курсу «Основы информатики и вычислительной техники» и в настоящее время активно развивается [8]. Идеи, заложенные академиком А. П. Ершовым и его коллегами еще в начале 80-х годов XX века, находят свое применение уже в XXI веке.

Новый скриптовый язык программирования *с синтаксисом на русском языке*. **RuSL** (Russian Scripting Language). Это учебный (полупрофессиональный), интерпретируемый, структурный, функциональный язык программирования с императивной структурой кода, последовательной типизацией и синтаксисом на русском языке. RuSL ориентирован на русскоговорящих пользователей с любым уровнем знаний в программировании [9]. **RuSL** облегчает общее понимание устройства любых ЯП и переход к профессиональным языкам.

```
1  изм к: число, сумма: число;
2  цикл к >> 100 {
3
4      изм х = число(ввести());
5      если х > 10 {
6          сумма += 1;
7      }
8
9  } вывести(сумма);
```

Рисунок 13 – Пример кода на языке RuSL

В среде *профессиональных языков программирования, использующих русский язык*, в мире сейчас в основном известен язык 1С: разработчик «Фирма 1С» под руководством Бориса Георгиевича Нуралиева.

#### Заключение.

Во время работы над статьёй утвердилась уверенность в том, что язык неразрывно связан с мышлением. **Русский язык** – это то, что всех нас и **программистов**, и **не программистов** объединяет. Надеемся, что современные языки программирования с синтаксисом **на русском языке** будут развиваться и совершенствоваться. Языки программирования с русским синтаксисом станут активнее применять и обучающиеся программированию, и мастера программирования.

#### Список литературы

1. [Искра-1256](https://ru.wikipedia.org/wiki/Искра-1256) — [Википедия \(wikipedia.org\)](https://ru.wikipedia.org/wiki/Искра-1256) Статья. [Электронный ресурс] Режим доступа свободный (дата обращения 9.12.2022 г.)
2. [Язык "Искры-1256"](https://livejournal.com/entry/1256) ([livejournal.com](https://livejournal.com/entry/1256)) Статья. [Электронный ресурс] Режим доступа свободный (дата обращения 11.12.2022 г.)
3. <https://skillbox.ru/media/code/kuda-propali-sovetskie-yazyki-programmirovaniya/> Статья. [Электронный ресурс] Режим доступа свободный (дата обращения 11.12.2022г.)
4. [Валентин Турчин | СЕМЬ ИСКУССТВ \(7iskusstv.com\)](https://7iskusstv.com/) Статья. [Электронный ресурс] Режим доступа свободный (дата обращения 11.12.2022г.)
5. [Язык сломаешь. Советские алгоритмические языки для обучения программированию/Хабр \(habr.com\)](https://habr.com/ru/articles/711111/) Статья. [Электронный ресурс] Режим доступа свободный (дата обращения 11.12.2022 г.)
6. [Языки программирования, разработанные российскими и советскими программистами \(livejournal.com\)](https://livejournal.com/entry/1256) Статья [Электронный ресурс] Режим доступа свободный (дата обращения 11.12.2022 г.)
7. [ДРАКОН Википедия \(turbopages.org\)](https://turbopages.org/) Статья [Электронный ресурс] Режим доступа свободный (дата обращения 11.12.2022 г.)
8. [http://www.pi8plus.ru/uchebnyy\\_centr/novosti\\_i\\_akcii/letnie\\_podgotovitelnye\\_kursy\\_po\\_russkomu\\_yazyku\\_i\\_informatike/](http://www.pi8plus.ru/uchebnyy_centr/novosti_i_akcii/letnie_podgotovitelnye_kursy_po_russkomu_yazyku_i_informatike/) Статья [Электронный ресурс] Режим доступа свободный (дата обращения 12.12.2022 г.)
9. [RuSL русский язык программирования \(ruslang.ru\)](https://ruslang.ru/) Статья [Электронный ресурс] Режим доступа свободный (дата обращения 12.12.2022 г.)

## НА РУССКОМ ЯЗЫКЕ ПРОГРАММИРОВАТЬ НУЖНО!

*Ночевной Сергей Сергеевич  
ФКПОУ «НТТИ» Минтруда России  
Нефедова Людмила Петровна, преподаватель*

Русский язык – это то, что всех нас и программистов и не программистов объединяет. Использование родного языка в программировании является и признаком независимого развития этой отрасли и одновременно объективной базой такого развития[1].

А можно просто рассказать компьютеру по-русски, что нужно сделать? Можно, но компьютер ничего не сделает. Более того — даже если вы напишете это на английском языке, то всё равно ничего не получится. Сейчас есть нейросети, которые умеют разбирать естественный текст и предлагать какой-то фрагмент кода, но задачи сложнее «Попроси у пользователя два числа и выведи их сумму» нейронная сеть сделать пока не может.

Гвидо Ван Россум — создатель Питона считает, что код пишется 1 раз, а читается многократно. Поэтому удобочитаемость важна!

### **Преимущества русскоязычных кодов.**

Достаточно много программистов используют кириллические названия переменных и функций — это позволяет:

- избежать пересечения пространства имён и сразу разделять свой код/переменные/функции от чужих (это самое важное по-моему!!)
- самодокументирование кода — комментарии меньше или они описывают более общие вещи
- обфусикация кода для иностранных разработчиков/хакеров/аналитиков

### **Недостатки**

- необходимо следить чтобы исходники были в кодировке UTF-8

Компьютеру же неважно, на каком языке написан код. Главное — чтобы существовал интерпретатор, который может перевести программный код, написанный человеком, в команды, понятные компьютеру. Всё дело в интерпретаторе.

В теории, можно взять любой язык программирования, перевести все его команды на русский язык и потом написать интерпретатор для этого языка. Работать вся конструкция будет так:

1. Берём код на русском языке.
2. Переводим каждую команду обратно в исходный вид в оригинальном языке.
3. Так делаем с каждой командой.
4. Получившийся код запускаем с помощью оригинального компилятора или интерпретатора.

Видно, что с таким подходом больше мороки, чем пользы:

- нужно написать отдельный новый интерпретатор;
- научить его проверять ошибки;
- научить его переводить код в оригинальные команды другого языка;
- по возможности, чтобы это не падало и не тормозило;
- написать документацию и обучающие материалы.

Гораздо проще сразу писать на нормальном языке программирования, а не изобретать свой собственный[2].

**Другой вариант** — использовать директивы вроде `#define` в языке С и С++. Смысл `#define` в том, что перед компиляцией все слова, которые мы таким образом определили, заменяются на изначальные. И тогда можно написать что-то такое:

```
#define ЕСЛИ if
#define ДА true
#define НЕТ false
#define АНЕТО else
#define ВЫВОД printf
#define ОТДАЛ return
И получится что-то такое (не вдаваясь в подробности):
ЕСЛИ (...) {
ВЫВОД (...);
}
АНЕТО ЕСЛИ (...)
ВЫВОД (...);
АНЕТО ВЫВОД (...);
ОТДАЛ...;
}
```

Долго эксплуатировать такой код будет неудобно, даже если вы сами его написали. Сначала нужно запомнить все свои дефайны, потом не перепутать их при употреблении. А потом другой разработчик придёт в ваш код и проклянёт вас.

Когда-то был придуман специальный язык для обучения школьников и студентов, которые до этого не изучали программирование. КуМир — это сокращение от Комплект Учебных МИРов. Его разработал советский математик Анатолий Кушниренко, но придумал его академик Андрей Ершов.

Например, ЭТО Бегемот задаёт переменную «Бегемот», ПОВТОРИ — это цикл, а «ЭТО...КОНЕЦ» позволяет организовать функцию (рис 1).

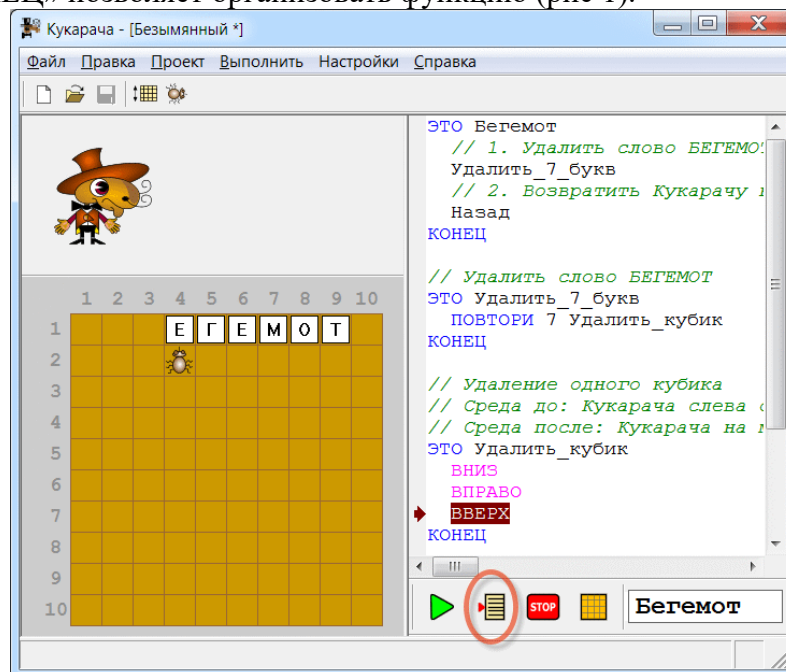


Рисунок 1 - Код для Кукарачи

Ещё есть обычные команды языка, не привязанные к исполнителю — они похожи на команды JavaScript, которые мы использовали для простых консольных проектов. Например, можно организовывать циклы, работать с массивами, запрашивать и вводить числа и писать почти полноценный алгоритмический код (рис 2).

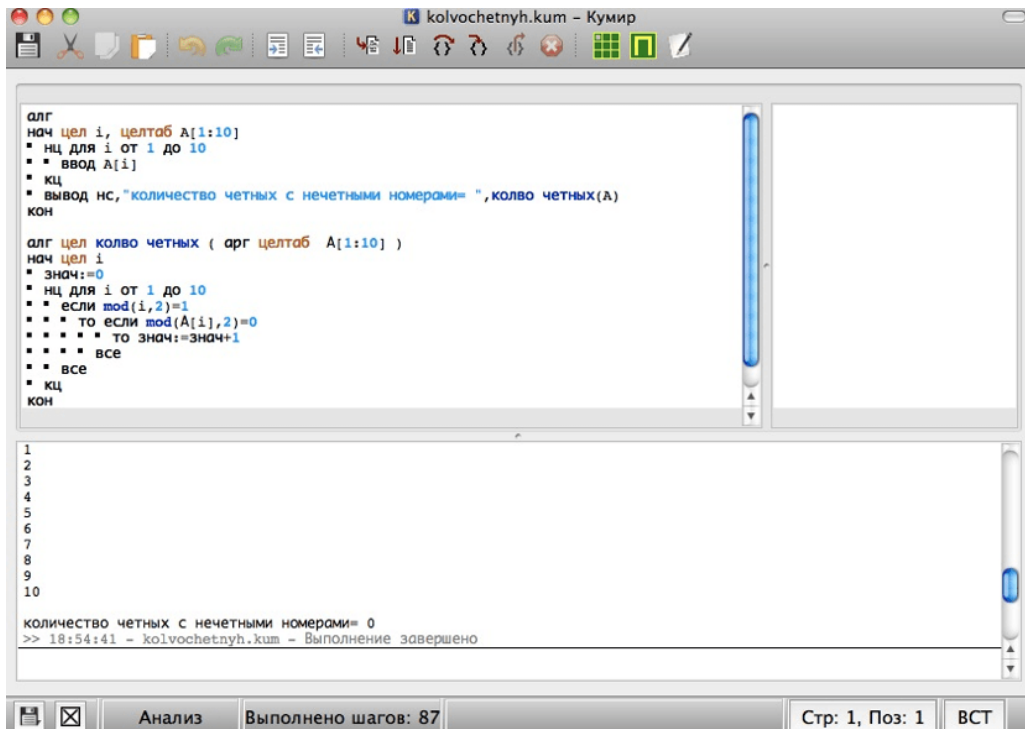


Рисунок 2 - Алгоритм подсчёта чётных чисел с нечётными порядковыми номерами  
Современные русскоязычные разработки.

Яр - это русскоязычный универсальный язык программирования. Яр строится на платформе языка "искусственного интеллекта" Common Lisp, подобно тому, как Kotlin - на платформе JVM или F# - на платформе .Net. Программы на Яре быстры и быстро запускаются (прекомпиляция в двоичный код) и надёжны (сборка мусора, статическая типизация). "Горячая замена кода" позволяет менять программу без её перезапуска. Например, можно будет добавлять в классы новые поля, а в функции - новые параметры. Это ключевое преимущество Яра перед другими языками, которое присуще используемой платформе Common Lisp. Возможная область применения такая же, как у C# или Java. Язык и кроссплатформенная среда разработки выпущены под лицензией MIT - можно без ограничений встраивать в коммерческие приложения.

Что касается Яра[9], то в нём кроме русификации, есть ряд инноваций, но большинство из них засекречены.

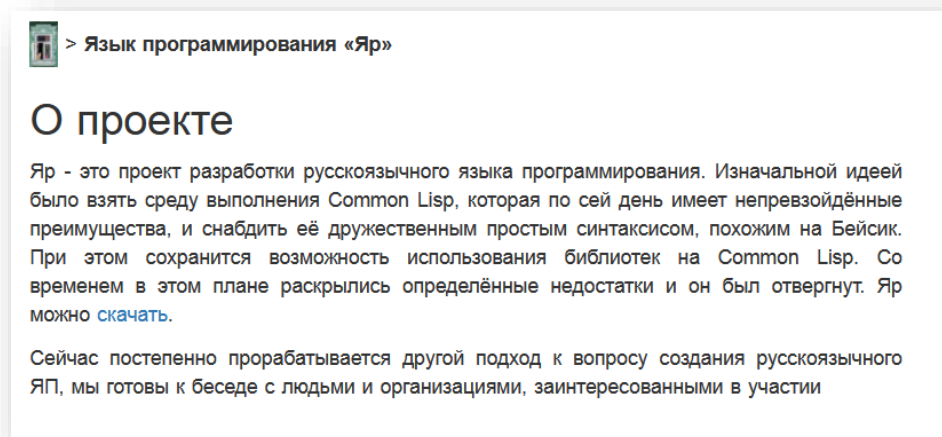


Рисунок 3 - Портал поддержки проекта разработчиков языка Яр

## Язык программирования бизнес-приложений – русский язык 1С

Главный инструмент разработки конфигураций информационных систем на платформе 1С - встроенный язык программирования 1С. У него двуязычный синтаксис: русский и английский. Программисты из России обычно выбирают русский. К нему быстро привыкают даже те, кто раньше программировал на языках с английским синтаксисом, — так понятнее.

Программы для 1С не разрабатывают на общепринятых языках программирования, вроде Python или C++. Дело в том, что встроенный язык ускоряет работу программиста — он специально оптимизирован для разработки программ учета. В языке 1С даже предусмотрены особые классы объектов: документы, справочники, отчеты, регистры бухгалтерии и другие

Программирование под 1С – это отдельный мир, в котором 1С-программисты ценятся высоко, потому что никто кроме них не может разобраться в их коде, хотя он и написан по-русски.

С простыми программами всё понятно:

```
Процедура ПриветМир()
    Сообщить("Привет, мир! Это журнал Код!");
КонецПроцедуры
```

Но если нам нужно что-то более сложное, начинается программистский ад: команды — это существительные, прилагательные и глаголы на русском языке, которые пишутся слитно и с больших букв в середине. При этом язык 1С – полноценный язык программирования высокого уровня, с ООП, классами и динамической типизацией.

Например, вот фрагмент программы, которая формирует отчет для бухгалтера:

```
&НаСервере
Процедура СформироватьНаСервере()
    ТабДок = Новый ТабличныйДокумент;
    ДатаС = ПериодОтчета.ДатаНачала;
    ДатаПо = ПериодОтчета.ДатаОкончания;
    Выборка = Документы.СчетФактураВыданный.Выбрать(ДатаС,
ДатаПо, , "Дата");

    ЭтотОтчет = РеквизитФормыВЗначение("Отчет");
    Макет = ЭтотОтчет.ПолучитьМакет("Реестр");
    ОбластьШапка = Макет.ПолучитьОбласть("Шапка");
    ТабДок.Вывести(ОбластьШапка);
    ОбластьСтрока = Макет.ПолучитьОбласть("Строка");

    Пока Выборка.Следующий() Цикл
        Если ЗначениеЗаполнено(Организация)
            И Выборка.Организация <> Организация Тогда
                Продолжить;
        КонецЕсли;

        ОбластьСтрока.Параметры.Дата = Выборка.Дата;
        ОбластьСтрока.Параметры.Номер = Выборка.Номер;
        ОбластьСтрока.Параметры.Контрагент = Выборка.Контрагент;
        ОбластьСтрока.Параметры.Сумма = Выборка.Сумма;

    ТабДок.Вывести(ОбластьСтрока);
```



КонецЦикла;

ПолеОтчета = ТабДок;

КонецПроцедуры

### Что такое 1С.Предприятие и где используются ее программы.

Все, кто интересовался работой бизнеса, точно слышали про 1С. Это целый комплекс программ для управления предприятием. Они автоматизируют вообще всё: от бухгалтерии в микроскопической компании до системы управления поставками в холдингах горнодобывающей отрасли.

Программы 1С популярны в России и ближайшем зарубежье. По данным [официального сайта 1С](#), ими пользуются **полтора миллиона организаций**, от микрокомпаний и ИП до огромных корпораций.

Причина в том, что программы 1С [строго соответствуют](#) законам России и стран СНГ: Налоговому кодексу, федеральным законам, постановлениям Правительства, приказам Минфина. Все изменения оперативно отражаются в программах. Разработка комплекса идет [с 1991 года](#) — за это время решения «обкатали» в сотнях тысяч самых разных компаний.

1С состоит из технологической платформы и прикладного решения (конфигурации).

**Платформа** — это основа программы и среда разработки со встроенным языком 1С, инструментами для администрирования базы данных и обновления конфигураций.

**Конфигурация** — программа на языке 1С для пользователей с открытым кодом. Она не может существовать отдельно от платформы. У программы удобный интерфейс и большие возможности для настройки. На одной платформе может быть установлена одна или сразу несколько конфигураций.

1С работает в двух режимах:

1. «1С:Предприятие» — это рабочий режим. Его применяют пользователи: бухгалтеры, менеджеры, экономисты.
2. Конфигуратор предназначен для настройки и администрирования программы. Здесь создаются новые объекты конфигурации и редактируются существующие, тестируется и исправляется информационная база, устанавливаются обновления и производятся другие действия по настройке и администрированию программы. Специально для этих специалистов встроили так называемую среду разработки (иными словами — конфигуратор), это опция удобная и ей легко обучиться. В процессе программирования отображается много подсказок, функция Справка очень многоплановая и подробная, есть опция подсветки конструкций языка и др.

**Конфигурации бывают типовыми и нетиповыми.**

**Типовые** — разработанные самой компанией 1С. Она их поддерживает и выпускает новые версии. Типовых конфигураций множество — например, те, о которых рассказано выше.

**Нетиповые** — конфигурации, созданные другими компаниями и отдельными программистами. Но программисты редко пишут их с нуля, обычно дорабатывают существующие. Это быстрее и дешевле.

**Интерфейс 1С** - это набор способов взаимодействия пользователя с системой. Многие годы программисты 1С работали и над интерфейсом, чтобы всем было удобно работать с ПП, чтобы он был интуитивно понятным каждому пользователю. [5]

На сегодняшний день взаимодействовать с системой можно следующими способами:

1. Толстый клиент. Это стандартная установка ПП на компьютер.

2. Тонкий клиент. Устанавливают на рабочее место ограниченный, быстрый дистрибутив, а для передачи данных допустимо иметь и слабое соединение (к примеру, GPRS). Тонкий клиент визуально не отличим от толстого. Версия платформы 8.3 приблизила систему к привычным web-интерфейсам.

3. Web клиент,

4. Мобильный клиент.

**База данных 1С.** Все сведения программный продукт 1С (8.3) размещает в базе данных. При этом ПП 1С работает как со своей системой управления, так и со сторонней СУБД. В ПП прописали способность интегрироваться с практически всеми передовыми и известными системами, что дает возможность гибко управлять масштабированием.

Поддерживаемые СУБД:

- встроенная файловая;
- MS SQL Server;
- PostgreSQL;
- Oracle DB;
- IBM DB2.

Программные продукты 1С ежедневно применяются для повышения эффективности управления и учета более чем на пяти миллионах рабочих мест в организациях различного размера и форм собственности, в России и других странах[6].

Технологическая платформа "1С:Предприятие 8" является уникальным сочетанием комплекса инструментов и технологий (full-stack platform), позволяющего создавать высоконагруженные кроссплатформенные бизнес-решения любой сложности, и концепции low-code, позволяющей минимизировать объем написанного кода, значительную часть разработки вести в режиме визуального конструирования, и в ряде случаев модифицировать бизнес-логику приложения силами конечных пользователей. В мире low-code платформы становятся все более востребованными, поскольку ускоряют цикл цифровой трансформации от бизнес-потребности до работающего бизнес-процесса.

В 2021 году аналитическая компания G2 включила платформу 1С:Enterprise (1С:Предприятие) в свой отчет как одного из лидеров в категории Low-code development platforms.

Согласно отчету Software Advice Inc, подразделения глобальной исследовательской и консалтинговой компании в области информационных технологий Gartner Inc., платформа "1С:Enterprise" ("1С:Предприятие") по состоянию на февраль 2021 года входит в число Frontrunners (лидеров) в категории программного обеспечения для управления бизнес-процессами.

В 2021 г. продолжилось интенсивное развитие платформы "1С:Предприятие". В ней реализуются такие инновации, как инструменты и сервисы машинного обучения, средства создания чат-ботов и виртуальных ассистентов.

В мобильных приложениях 1С стало доступно автоматизированное тестирование средствами платформы 1С:Предприятие, что позволяет построить замкнутый цикл разработки и тестирования мобильных приложений с использованием исключительно инструментов 1С.

Начиная с версии 8.3.20, стал доступен (пока в бета-версии) облачный сервис сборки и публикации мобильных приложений, созданных на платформе 1С. Новый облачный сервис существенно снижает порог входа в мобильную разработку 1С. Он значительно упрощает сборку мобильных приложений и их публикацию на маркетах App Store и Google Play, а также установку приложений на мобильные устройства. Сборка мобильных приложений производится в облаке 1С; в частности, теперь для сборки приложения под iOS разработчикам больше не нужен компьютер под управлением macOS.

В составе платформы 1С:Предприятие появился ряд новых продуктов, таких как:

- 1С:Аналитика – продукт класса BI (Business Intelligence), предназначенный для быстрого построения аналитических отчетов и интерактивного анализа данных "на лету". Особенность продукта в том, что он ориентирован не на разработчиков, а на бизнес-пользователей. Финальная версия 1С:Аналитики выпущена в сентябре 2021 года.
- 1С:Шина – продукт класса ESB (Enterprise Service Bus, сервисная шина предприятия), предназначенный для обмена данными между различными информационными системами, созданными как на платформе 1С:Предприятие, так и на других платформах. Финальная версия 1С:Шины выпущена в декабре 2021 года, у продукта уже есть несколько успешных внедрений, в частности, в нефтесервисном холдинге "ТНГ-Групп", в Тамбовском государственном университете им. Г.Р. Державина, в финтех-компании "Альфа партнер".

Расширить границы применения "1С:Предприятия" позволяет новая технология "1С:Предприятие.Элемент". Она позволяет создавать приложения, ориентированные не только на бизнес, но и на конечных пользователей ("консьюмерские"), в т.ч. "витрины", "личные кабинеты", фронт-офисы, "headless-сервисы". Эта технология уже используется в ряде новых продуктов - "1С:Кабинет сотрудника", "1С:Шина", "1С:Аналитика", получивших хорошие оценки пользователей.

Система "1С:Предприятие" стала современным трендом цифровизации. В числе крупнейших пользователей "1С:Предприятия" – Почта России (более 150 000 пользователей), "Трансмашхолдинг" (более 35 000 АРМ), Ростех (более 27 000 АРМ), Правительство Москвы (единая облачная система, более 18 000 пользователей), Вымпел-Коммуникации (Билайн, более 13 000 АРМ) "Башкирэнерго" (более 9200 АРМ), "КамАЗ" (более 7800 АРМ), "Деловые Линии" (более 5000 АРМ), Российский Университет Дружбы Народов (ФГАОУ ВО "РУДН", 4700 АРМ). Стратегические соглашения и меморандумы о сотрудничестве с "1С" заключили АК "Россия", АК "Победа", группа "Синара", "Магнит", QIWI, "Роскосмос", "Уралкалий", "Роснефть", "Ростех", РЖД, "Газпром Нефть", "Россети", "АвтоВАЗ", "ЛокоТех", "Объединенные машиностроительные заводы", "Мечел", "Деловые линии", "Уралхим", "Зарубежнефть" и ряд других флагманов экономики.

В России и ближайшем зарубежье очень много компаний, которые используют 1С. Большинству из них нужны специалисты для обслуживания программ. В августе 2021 года только на HeadHunter было почти 11 000 вакансий для программистов 1С.

#### **Проекты создания языка программирования с русским синтаксисом.**

На каком языке писать – спорный вопрос, но можно оставить свой голос на сайте голосования [8].

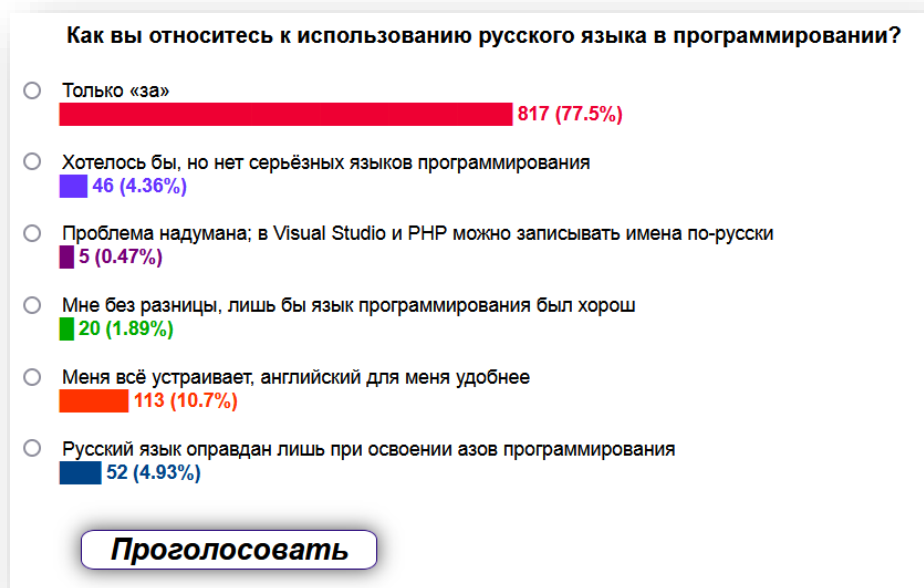


Рисунок 4 - Голосование программистов на сайте <http://compiler.su/russkij-yazyk-i-programmirovanie.php>

Существует общая бизнес идея:

- использовать существующие популярные ЯП с открытым исходным кодом,
- дополнить их русскими операторами
- перевести сообщения об ошибках и сообщения отладчика
- перевести документацию и учебники
- сделать конвертор английские операторы<->русские операторы
- дополнить плагины для IDE для подсветки синтаксиса и автодополнения
- объединить всё в 1 пакет установки (чтобы облегчить первый запуск)

#### **Преимущества использования ЯП с русскими операторами.**

- Снижение входного порога для программирования
- Улучшение понимаемости кода (особенно чужого) и снижение количества ошибок.

- Разделение на программистов и кодировщиков (как на западе)

#### **Недостатки.**

- От разработки российского открытого ПО отсекаются не русскоязычные пользователи
- Компенсируется конверторами рус<->лат операторы, реально больше мешают не английские комментарии и не английская документация

#### **Общие направление создания ЯП с русскими операторами.**

- Все ЯП с русским синтаксисом для одинаковых конструкций должны по возможности использовать одни и те же русские слова
- Использование существующих англоязычных ЯП и дополнение англоязычных операторов русскоязычным переводом, англоязычные операторы должны не исключаться, а дополняться русскими, чтобы интерпретатор/компилятор мог свободно обрабатывать код содержащий и русские и английские операторы
- Корректировка существующих учебников по русифицируемому языку
- Перевод на русский сообщений об ошибках и отладочной информации
- Корректировка плагинов к IDE для добавления подсветки синтаксиса и автодополнения
- Написание плагина для импорта экспорта кода из рус<->лат операторов

### Чего необходимо избегать

- Устаревшие ЯП - типа паскаля - есть более современные конкуренты;
- любительские проекты - вряд ли будут продажи;
- проекты для исследования какой либо "концепции".

Одной **русификации** мало. Хотя если узаконить её, то будет достаточно. Например, можно потребовать, чтобы все идентификаторы в программах, закупаемых для гос. нужд, были русскоязычными, и выпустить стандарт для того, чтобы не было зоопарка названий. Я не вижу здесь вообще никаких проблем - просто нужно принять такой закон.

**Другой вариант** - есть закон о гос. языке. Там написано (статью сейчас не вспомню), что при "взаимоотношениях" организации РФ должны использовать русский язык. Если считать передачу программы "взаимоотношениями", то можно сказать, что всё ИТ нарушает сегодня закон.

**Третий вариант** - включить русификацию уже существующий закон об импортозамещении.

Широко распространено мнение, что английский является международным языком программистов и программировать можно только на нём. На самом деле, опыт 1С показывает, что вполне можно программировать (и хорошо зарабатывать) с использованием русского языка.

### Источники

1. Статья. Создание языка программирования с русским синтаксисом [https://тхаб.рф/wiki/Создание языка программирования с русским синтаксисом](https://тхаб.рф/wiki/Создание_языка_программирования_с_русским_синтаксисом)
2. Статья. Языки программирования с русским синтаксисом [https://тхаб.рф/wiki/wiki/Языки программирования с русским синтаксисом](https://тхаб.рф/wiki/wiki/Языки_программирования_с_русским_синтаксисом)
3. Статья. Архитектура платформы 1С Предприятие. <https://v8.1c.ru/platforma/obekty-konfiguracii/>
4. Статья. Как стать 1С-программистом с нуля. <https://blog.skillfactory.ru/programmist-1s-obyazannosti-zarplata-kak-im-stat-s-nulya/>
5. Статья. Из чего состоит программа 1С. <https://mastersoft.ru/blog/iz-chego-sostoit-programma-1s/>
6. Материалы конференции 2022г. Диго С.М., Нуралиев Б.Г. Фирма "1С", г. Москва. Взаимодействие экосистемы 1С с системой образования в эпоху цифровой экономики.
7. [Официальный сайт 1С](#)
8. <http://compiler.su/russkij-yazyk-i-programmirovanie.php>
9. Статья. Яр (язык программирования) [https://тхаб.рф/wiki/%D0%AF%D1%80\\_\(%D1%8F%D0%B7%D1%8B%D0%BA\\_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F\)](https://тхаб.рф/wiki/%D0%AF%D1%80_(%D1%8F%D0%B7%D1%8B%D0%BA_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F))
10. <https://infostart.ru/1c/articles/724250/>

## СЕКЦИЯ 4. ДОСТИЖЕНИЯ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ В ОБЛАСТИ ХРАНЕНИЯ И ОБРАБОТКИ ДАННЫХ, DATA SCIENCE: BIG DATA, ГРИД-ТЕХНОЛОГИИ, ОБЛАЧНЫЕ ВЫЧИСЛЕНИЯ, БЛОКЧЕЙН И Т.Д.

### ПРАКТИЧЕСКИЕ АСПЕКТЫ ИСПОЛЬЗОВАНИЯ ОБЛАЧНЫХ ТЕХНОЛОГИЙ

*Меркулова Елизавета Сергеевна  
ФКПОУ «ОГЭКИ» Минтруда России  
Максимова Александра Витальевна преподаватель*

**Аннотация.** В данной статье рассматриваются отличительные особенности, принцип работы и применение облачных хранилищ данных, поскольку их использование в наше время становится все более и более распространенным из-за развития множества интернет-сервисов, позволяющих хранить и обрабатывать данные в «облаке».

Ключевые слова: облачное хранилище, данные, информация, модель, сервис.

Облачное хранилище данных (англ. cloud storage) — это модель онлайн-хранилища, данные в котором хранятся на множественных серверах, распределённых в сети. Сервера предоставляются клиентам, как правило, третьей стороной. Информация о количестве или каким-либо элементам внутренней структуры серверов клиенту обычно не видна, в отличие от метода хранения данных на собственных выделенных серверах, которые приобретаются или арендуются специально для подобных целей.[1]

В модели облачного хранилища данные хранятся и обрабатываются в «облаке», которое с точки зрения клиента представляет собой один большой виртуальный сервер. Однако, физически эти сервера могут географически располагаться друг от друга удалённо. Облачными хранилищами являются такие интернет-сервисы, как: Dropbox, OneDrive, Google Drive, iCloud, Яндекс.Диск, Облако Mail.Ru.

Заявленная тема является актуальной, так как в последние несколько лет облачные хранилища данных превратились из малоприменимого для персонального использования и дорогого способа хранения информации в вполне доступный и эффективный инструмент для решения таких задач.

Цель данной работы: изучение практического применения облачных технологий, как современного способа хранения информации.

Для достижения данной цели были поставлены следующие задачи:

- проанализировать данные из различных информационных источников по теме;
- изучить теоретические и практические аспекты процесса применения облачных технологий;
- выделить преимущества и недостатки облачных технологий, и их отличие от традиционных;
- обобщить варианты применения облачных технологий в образовательном процессе.

Объект исследования: облачные технологии.

Предмет исследования: перспективы развития, достоинства и недостатки облачных технологий.

Методы исследования:

- теоретические: анализ научных данных по проблеме исследования;
- эмпирические: наблюдение, анализ, опрос, беседа;
- статистические: обработка результатов.

Теоретическая и практическая значимость исследования заключается в систематизации знаний по проблеме использования облачных технологий в образовательном процессе.

### **Преимущества и недостатки облачных технологий**

Первое достоинство данной модели – это возможность синхронизации файлов между несколькими устройствами. Например, если вам нужно постоянно иметь доступ к вашим документам и деловым бумагам на разных устройствах (ноутбук, планшет, смартфон), то вы просто можете просто загрузить эти файлы в «облако» и просматривать их оттуда, вместо того, чтобы загружать документы на каждое устройство отдельно.

Также облачное хранилище данных может выполнять функцию резервного копирования ваших данных. Если вы загрузили в «облако» файлы с рабочего компьютера, то при его внезапном выходе из строя вы не потеряете свои данные. Это очень полезное свойство, так как в современном мире информация очень ценна и может понадобиться в любое время, поэтому хранить документы только на вашем рабочем компьютере небезопасно.

Еще одним достоинством данной технологии является возможность совместного использования файлов. С помощью облачного хранилища данных вы можете обмениваться файлами с друзьями и коллегами. Как правило, при обмене файлами по электронной почте вы ограничены несколькими десятками Мб. При использовании облачного хранилища вы сможете делиться терабайтами информации. Из этого вытекает еще одно полезное свойство данной технологии – расширение вашего локального хранилища. Любые устройства обладают ограниченным объемом памяти, и вам может ее не хватить для хранения всей необходимой вам информации. Вы, конечно же, можете установить дополнительно еще один жесткий диск SSD или SD карту, но этот путь связан с большими финансовыми затратами. Кроме того, ваше устройство может и вовсе не иметь дополнительного слота под жесткий диск. В этом случае вам на помощь приходит «облако», на которое вы можете загрузить данные, которые не помещаются на вашем рабочем устройстве.

Преимуществом облачного хранилища данных перед арендой сервера заключается в том, что клиент платит только за то место в хранилище, которое он фактически использует, а не за весь сервер, ресурс которого он может и не использовать в полном объеме. Также клиент избавлен от необходимости заниматься такими вещами как приобретение, поддержка и обслуживание собственной инфраструктуры по хранению данных. Это позволяет в конечном счете уменьшить общие издержки производства.

Еще хотелось бы отметить тот факт, что все процедуры по сохранению целостности данных и их резервному копированию предоставляются провайдером «облачного» центра, что позволяет не вовлекать в этот процесс клиента. В большинстве своем поставщики облачных хранилищ устойчивы к потере данных. При загрузке файлов на «облако» поставщик сохраняет копии ваших данных до шести раз на шести разных жестких дисках в отдельных изолированных центрах обработки данных. Вы будете думать, что существует только один файл, но его копии существуют. Такая облачная стратегия избыточных копий называется высокой доступностью и гарантирует возможность ваших данных в случае различных виртуальных или физических катастроф.

Первым недостатком является необходимость подключения к интернету. В наше время невозможно представить себе жизнь без интернета, на всех современных устройствах есть возможность пользоваться мобильным интернетом или Wi-Fi, но все-таки вы можете оказаться в месте, где нет сети Интернет или Wi-Fi, а мобильной связи нет или сигнал слишком слабый. В таком случае вы не сможете получить доступ к

своим файлам в «облаке», и если у вас отсутствуют эти данные на жестком диске, вы уже никак не сможете воспользоваться своими документами. Также от качества работы интернет провайдера клиента очень сильно зависит надёжность, своевременность получения и доступность данных в «облаке». А если сама компания, которая предоставляет онлайн хранилище будет ликвидирована, то клиент может потерять абсолютно все свои данные.

Вторым и самым главным недостатком хранения данных в «облаке» является низкая безопасность при хранении и пересылке данных. Провайдер имеет возможность просматривать данные клиента, которые не защищены паролем, поэтому хакеры, взломавшие системы защиты провайдера, могут заполучить ваши данные.

Аналитическая фирма IDC провела исследование, которое показало, что многие компании связывают с облачными хранилищами большие проблемы в области безопасности. Позднее исследовательская организация Portio Research подтвердила эти результаты и указала конкретные цифры: 68% руководителей европейских ИТ-компаний, участвовавших в опросе, отказываются пользоваться «облачными» технологиями из-за проблем с безопасностью, они опасаются, что их данные могут стать достоянием общественности.[1] Также к недостаткам облачного хранилища данных можно отнести абонентскую плату за дополнительные возможности, например, увеличенный объем хранения данных и передачу больших файлов.



Рис.1 Модели обслуживания облачных сервисов

Практически ни одна компания не обходится без ИТ-инфраструктуры (модели обслуживания и примеры сервисов, которые работают по этим моделям, вы можете увидеть на рисунке 1— даже небольшой фирме нужны серверы для хранения баз данных или инструменты, объединяющие компьютеры сотрудников в общую сеть. Компания может закупить серверы и настроить эту инфраструктуру у себя, но это достаточно долго и дорого, при этом можно ничего не покупать, если взять в аренду облачные сервисы IaaS.[2]

**1. IaaS — это Infrastructure as a Service.** В переводе на русский язык - инфраструктура как услуга. К инфраструктуре относят такие вычислительные ресурсы как виртуальные серверы, хранилища, сети. Это похоже на виртуальные «компьютеры», на которые можно установить операционную систему, программное обеспечение, приложения.

IaaS включает в себя:

Виртуальные серверы (VPS/VDS), на которые можно устанавливать различные программы.



Сетевые настройки, которые обеспечивают связь виртуальных серверов между собой, внешними серверами, которые принадлежат компании-клиенту, и интернетом. К ним относят:

- доступность серверов друг для друга и для внешней сети, маршрутизацию сетевых соединений серверов;
- балансировщики нагрузки;
- VPN — технологию шифрования данных, передаваемых компанией между облаком и ее физическим дата-центром;
- Управление доступом пользователей;
- Облачные хранилища для хранения файлов, данных или бэкапов;
- Сервисы резервного копирования;

Примеры IaaS — как можно использовать облачную инфраструктуру:

- Перенос IT-систем в облако.
- Экономия на инфраструктуре.
- Быстрый запуск бизнеса.
- Расширение инфраструктуры.

## **2. PaaS: облачная платформа как услуга:**

Провайдеры облачных услуг могут предоставлять уже настроенные инструменты (платформы) под разные задачи. Такие инструменты называют PaaS — это Platform as a Service, в переводе - платформа как услуга.

Ключевое отличие PaaS от IaaS в том, что здесь у вас есть определенные инструменты, к примеру: система управления базами данных, среда машинного обучения или обработки big data, промышленный IoT. Их необходимо настроить под потребности компании, но не требуется строить с нуля.

Вот несколько примеров PaaS:

- Базы данных. В облака можно перенести все или часть баз данных компании.
- Разработка приложений в контейнерах.
- Аналитика больших данных.
- Машинное обучение.

**3. SaaS: программное обеспечение как услуга:** SaaS — это полностью настроенная и готовая к работе программа, выполняющая определенные функции. Единственное отличие технологии SaaS от программы на смартфоне или компьютере в том, что сам софт находится в облаке.

SaaS — это Software as a Service, программное обеспечение как сервис. IaaS и PaaS для SaaS-сервисов могут выступать как инфраструктура и среды разработки и развертывания (вы можете это увидеть на рис. 1): разработчики программного обеспечения используют облачные мощности, чтобы разрабатывать, запускать и хранить SaaS-приложения, обеспечивать к ним доступ пользователям.

Примеры SaaS — это большинство сервисов в интернете: электронная почта, CRM-системы, планировщики задач, веб-конструкторы для создания сайтов, платформы для ведения блогов.[2]

### **Практическое применение облачных технологий**

Большинство компаний среднего и крупного бизнеса понимают, в чем польза облаков, и в той или иной степени используют облачные технологии.

Сейчас многие компании переходят к такому принципу как cloud only («только облака») — то есть рассматривают облачный вариант построения инфраструктуры как единственный сценарий развития бизнеса.

### **Устойчивость интернет-магазинов в пики продаж**

Нагрузка на интернет-магазины часто непредсказуема. В течение года могут быть как сезонные изменения, так и непредсказуемые пики и падения продаж.

Всплески покупательской активности обычно возникают в период распродаж — иногда они превосходят все ожидания и какая-нибудь «черная пятница» легко может обрушить сайт.

В «облаках» есть инструмент решения проблемы — это автоматическое масштабирование. Облачные сервисы могут распределить запросы пользователей по разным дата-центрам. Во время распродажи мощность может увеличиться многократно, чтобы магазин без проблем обработал все запросы. А в другое время мощность снижается, и компания не переплачивает за ресурсы, которые не использует.

### **Использование облаков в финансовом секторе**

Финансы, банки, страхование всегда считались консервативными отраслями, но сейчас они вырвались в IT-авангард и внедряют инновационные технологии одними из первых.

С помощью «облаков» компании финансового сектора могут значительно сэкономить на автоматизации рутинных операций, которую обеспечивает переход к облачной инфраструктуре.

### **Облака помогают хранить персональные данные в сфере здравоохранения**

Обращение с цифровыми медицинскими данными строго регулируется законодательством, так как они относятся к чувствительным данным. Есть возможность передать их в облако: сейчас облачные провайдеры уже умеют настраивать свои платформы для работы с конфиденциальной информацией и персональными данными в режиме, который необходим конкретной компании или установлен требованиями какого-либо ведомства.

### **Сервисы мультимедиа раздают контент с помощью облаков**

Медиа-сервисы сталкиваются с пиками посещаемости так же часто, как и интернет-магазины. На облачных платформах есть специальные объектные хранилища, которые предназначены для хранения востребованных фото, аудио и видео. В таком хранилище доступ к данным могут получить одновременно миллионы пользователей, и им не придется столкнуться с какими-либо проблемами.

### **Облачные технологии в сфере образования**

Сегодня образование в России стоит перед очевидной необходимостью пересмотра своих целевых установок. А именно, в ходе образовательного процесса современный человек должен не столько накапливать багаж знаний и умений, сколько приобретать способность самостоятельно и совместно с другими людьми ставить осмысленные цели, выстраивать ситуации самообразования, искать и продуцировать средства и способы разрешения проблем.

В этом контексте, прекрасно вписываются облачные технологии, от простых online инструментов, где обучающиеся могут совместно рисовать и делать записи, до сложных технологий совместной работы над проектами. Преподаватели и студенты, являются активными участниками этого процесса. Больше всего здесь подходят технологии SaaS (аренда IT-приложений и облачные веб-сервисы, ведь среди них есть много абсолютно бесплатных).

Как пример использования облачных технологий в образовании, можно назвать:

- электронные дневники, журналы;
- личные кабинеты для студентов и преподавателей;
- тематические форумы, где обучающиеся могут осуществлять обмен информацией;
- осуществление процесса дистанционного обучения;
- поиск информации, где обучающиеся могут решать определенные учебные задачи даже в отсутствие преподавателя или под его руководством;
- облачные хранилища данных.

Наиболее распространенной системой сервисов на основе технологии облачных вычислений, применяемой в образовательном процессе, является Google Apps. Это

web-приложения, предоставляющие участникам образовательного процесса инструменты, использование которых призвано повысить эффективность общения и совместной работы. В пакет входят популярные веб-приложения Google, в том числе Gmail, Google Диск, Google Календарь и Google Документы.

В данный момент облачные хранилища данных являются популярными и необходимыми. Это вызвано многими причинами: нехваткой места на жестком диске, недолговечность операционной системы, «беготней с флеш-картой» и множеством других причин.

Облачные хранилища данных имеют множество преимуществ, но также обладают весьма весомыми недостатками, из-за которых предприятия пока что не могут их использовать. Основными недостатками на данный момент являются уязвимости в безопасности при облачном хранении и опасность хакерских атак на сервер.

Чтобы еще больше расширить использование облачных хранилищ, данных необходимо работать над устранением этих недостатков, но так или иначе облачные хранилища уже глубоко вошли в нашу жизнь, и с каждым днём все больше людей пользуются ими.

Также современные облачные технологии ведут к расширению возможностей бизнеса, ведь основа бизнеса – его рентабельность, а с применением «облаков» она возрастает. В наше время скорость реакции на изменения рынка играют особую роль, поэтому инновационные технологии всегда будут востребованы.

#### **Список использованных источников:**

1. Облачное хранилище данных URL: [https:// ru.wikipedia.org](https://ru.wikipedia.org)
2. Разница между IaaS, PaaS и SaaS: самая понятная статья об облаках в интернете URL: [https://mcs.mail.ru/blog/raznica-mezhdu-iaas-paas-saas- statya-ob-oblakah-v-internete](https://mcs.mail.ru/blog/raznica-mezhdu-iaas-paas-saas-statya-ob-oblakah-v-internete)
3. Облачный рынок в 2020 году URL: [https://mcs.mail.ru/blog/oblachnyj- gynok-v-2020-godu](https://mcs.mail.ru/blog/oblachnyj-gynok-v-2020-godu)
4. Облачные хранилища с бесплатными гигабайтами URL: [https://zen.yandex.ru/media/id/5a58409afd96b13b170a026d/oblachnye- hranilisca-s-esplatnymi-gigabaitami-5a5f37563dceb7f26059a32e](https://zen.yandex.ru/media/id/5a58409afd96b13b170a026d/oblachnye-esplatnymi-gigabaitami-5a5f37563dceb7f26059a32e)
5. Бесплатные облачные хранилища для пользователей и разработчиков URL: [https://zen.yandex.ru/media/merion\\_networks/besplatnye-oblachnye- hranilisca-dlia-polzovatelei-i-razrabotchikov- 5e2dd5393639e600b7f88b7d?utm\\_source=serp](https://zen.yandex.ru/media/merion_networks/besplatnye-oblachnye-hranilisca-dlia-polzovatelei-i-razrabotchikov-5e2dd5393639e600b7f88b7d?utm_source=serp)
6. Computer Bild: журнал. — 2010. — № 4. — С. 62—67 7. Chip: журнал. — 2011. — № 8 (149). — С. 20—21
8. Маркелов А. А. OpenStac. Практическое знакомство с облачной операционной системой. М. : ДМК-Пресс, 2018 – с.306.

## **СОВРЕМЕННЫЙ ПОДХОД К РАЗРАБОТКЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ.**

### **ПРАКТИКА DEVOPS, КОНТЕЙНЕРИЗАЦИЯ**

*Сумская Мила Евгеньевна, Андросов Григорий Игоревич  
ФКПОУ «НТТИ» Минтруда России  
Полякова Ольга Владимировна*

**Аннотация.** Приведено описание DevOps–набора практик для повышения эффективности процессов разработки (Development) и эксплуатации (Operation) программного обеспечения (ПО), который реализует гибкий подход для устранения организационных и временных барьеров между командами разработчиков и других участников жизненного цикла ПО. Рассмотрены основные методы DevOps, приведены примеры эффективного их применения. Приведено описание основных особенностей одного из самых популярных инструментов Docker, помогающего решить задачу автоматизации и централизации развертывания программного обеспечения.

Организационные преобразования являются частью жизни любой компании или предприятия. Сегодня программное обеспечение стало ключевым фактором инноваций в бизнесе. Это возможность для вывода новых товаров или услуг, создания новых каналов для работы с клиентами, портал для выхода на новые рынки.

Для удовлетворения постоянно расширяющихся бизнес-потребностей необходимо обеспечить возможность оперативно вносить изменения в программное обеспечение.

Сначала появился Agile(гибкий) - принцип эффективной и оперативной разработки программного обеспечения в процессе постоянного взаимодействия с заказчиком, который предполагает, что:

1. Взаимодействие людей важнее процессов и инструментов,
2. Работающий продукт важнее исчерпывающей документации.
3. Сотрудничество с заказчиком важнее согласования условий контракта
4. Готовность к изменениям важнее следования первоначальному плану

Однако скорость развертывания функционала у клиента оставалась на прежнем уровне. Возник конфликт на стыке работы разработчиков и подразделений эксплуатации программного продукта.

Развитие микросервисной архитектуры, контейнеризации, облачных сервисов и инженерных практик DevOps позволило развертывать каждый компонент приложения независимо друг от друга и разделять их релизные циклы.

DevOps (DEvelopment OPerationS) – это набор практик для повышения эффективности процессов разработки (Development) и эксплуатации (Operation) программного обеспечения (ПО) за счет их непрерывной интеграции и активного взаимодействия профильных специалистов с помощью инструментов автоматизации.



Рисунок 1 – Состав DevOps

DevOps позиционируется как Agile-подход для устранения организационных и временных барьеров между командами разработчиков и других участников жизненного цикла ПО (тестировщиками, администраторами, техподдержкой), чтобы они могли быстрее и надежнее собирать, тестировать и выпускать релизы программных продуктов[1].

В модели DevOps группы разработки и эксплуатации иногда объединяются и инженеры совместно работают над всем жизненным циклом приложения – от разработки и тестирования до развертывания и эксплуатации, не ограничиваясь узкой специализацией[2].

Непрерывная интеграция – это практика разработки ПО, при которой разработчики регулярно объединяют изменения программного кода в центральном репозитории, после чего автоматически выполняется сборка и тестирование. Главная задача непрерывной интеграции – быстрее находить и исправлять ошибки, улучшать качество ПО и сокращать временные затраты на проверку и выпуск новых обновлений(релизов) ПО.

Непрерывная доставка – это практика разработки ПО, когда при любых изменениях в программном коде выполняется автоматическая сборка, тестирование и подготовка к окончательному выпуску. Все изменения кода после стадии сборки развертываются в тестовой и (или) в рабочей среде. При правильном внедрении непрерывной доставки у разработчиков всегда есть готовый к развертыванию собранный экземпляр ПО, прошедший стандартизированную процедуру тестирования.

Архитектура микросервисов – это подход к разработке приложения в виде набора небольших сервисов. Каждый сервис работает в своем собственном процессе и обменивается данными с другими сервисами через детально продуманный интерфейс (API). Каждый микросервис имеет одну задачу. Для написания микросервисов можно использовать различные среды и языки программирования, а развертывать их можно независимо, по одному, или группой.

Инфраструктура как код – это практика, в которой выделение инфраструктуры и управление ею происходит с помощью кода и методов разработки ПО, таких как управление версиями и непрерывная интеграция. Облачная модель на основе API позволяет разработчикам и системным администраторам взаимодействовать с инфраструктурой на программном уровне при любом масштабе, вместо того чтобы устанавливать и настраивать ресурсы вручную. Поскольку инфраструктура и серверы определяются кодом, их можно быстро развернуть с использованием стандартных моделей, установить на них последние исправления и версии или дублировать одинаковым способом.

Управление конфигурацией. С помощью кода разработчики и системные администраторы автоматизируют настройку операционной системы и хоста, эксплуатационные задачи, а также многое другое. Благодаря использованию кода изменение конфигурации становится более воспроизводимым и стандартизованным. Это освобождает разработчиков и системных администраторов от ручной настройки операционных систем, системных приложений или серверного программного обеспечения.

DevOps позволяет работать так: утром идея – вечером вывод нового функционала в рабочую среду.

При частом выпуске релизов команда получает оперативную обратную связь о продукте и может реагировать немедленно, создавая условия для постоянного обучения и совершенствования продукта. Кроме того, автоматизированный контроль качества предотвращает утечку проблем в тестовую и производственную среды и, следовательно, рационализирует потребление ценных ресурсов команды.

Например, крупный российский ритейлер, имеющий несколько брендов сетей магазинов разных форматов, в рамках создания нового бизнес-направления запросил разработку платформы для интеграции с партнерами и управления системой доставки. Внедрить платформу требовалось в кратчайшие сроки при обеспечении показателя T2M (Time to Market - время от начала разработки до передачи клиенту, т.е. «выхода на рынок») не более одной недели[2].

Проектирование, разработка и поддержка платформы с применением микросервисной архитектуры приложений, гибких методологий Agile и практик DevOps позволили сократить время доставки изменений с недели до 1 дня. Количество развертываний в продуктивную среду увеличилось до 5 раз в день вместо 1 раза в неделю до внедрения DevOps. Сокращение трудозатрат – в 4 раза: развёртывание 50+

микросервисов в продуктивную среду сейчас занимает 2 часа, вместо одного рабочего дня ранее[3].



Рисунок 2 – Путь DevOps

Поскольку DevOps — это командная работа сотрудников, занимающихся разработкой, операциями и тестированием, нет единого инструмента «DevOps»: это скорее набор, состоящий из нескольких инструментов. Как правило, инструменты DevOps вписываются в одну или несколько из этих категорий, что отражает ключевые аспекты разработки и доставки программного обеспечения

Несмотря на то, что доступно множество инструментов, некоторые категории из них имеют особо важное значение в настройке инструментальных средств DevOps для использования в организации. Одним из самых популярных инструментов, помогающих в решении этой задачи, является Docker.

Сегодня любой проект по созданию или изменению продукта, особенно предназначенного для работы с конечным потребителем, будь то B2B(business-to-business) или B2C(business-to-customer) модель, так или иначе требует использования DevOps практик.

DevOps выполняет роль оптимальной стратегии для оперативного перевода кода, созданного программистами компании, в продуктивную среду в рамках корпоративного ИТ-решения – веб-сайта, приложения, системы бэкэнда компании и т.д.

Оптимальные результаты сегодня достигаются в связке Agile – оптимального ускорения процесса разработки - и DevOps конвейера – для тестирования и развертывания функционала существующих и новых продуктов и сервисов.

Максимальный эффект от использования методологий Agile и практик DevOps достигается в случае использования облачных сервисов, микросервисной архитектуры, а также технологий контейнеризации для оперативного развертывания в разных средах.

Основной сценарий применений DevOps – для разработки и развития внешних ИТ-инструментов и решений. Здесь T2M требуется обеспечивать на максимальном уровне.

Например, глобальная нефтегазовая компания. Охват – более 70 стран, выпускает около 3,7 миллиона баррелей нефтяного эквивалента в день и имеет 44 000 станций обслуживания по всему миру. Компании требовалось сократить показатель T2M (time-to-market), чтобы иметь возможность быстрее отвечать на отзывы клиентов,

сократить трудозатраты на применение изменений для работающих приложений, быстрее доставлять новый функционал и бизнес-ценность для клиентов.

По итогам проекта внедрения DevOps для мобильной и веб-версии приложения для клиентов компании скорость внедрения новых форматов бизнес-ценности для клиентов компании выросла в 23 раза, а число активных пользователей увеличилось с 800 тыс. до 1,5 млн. за 1 год[2].

Таким образом, DevOps практики стали стандартом эффективной разработки ПО и ни один Agile-проект без них не обходится. Спрос на их внедрение продолжает расти по мере развития потребностей бизнеса в новых продуктах и новой функциональности. DevOps становится главной методологией для всех компаний, которые развивают цифровизацию и инновационные стратегии роста.

### Docker

Docker является одним из самых популярных инструментов, помогающих решить задачу автоматизации и централизации развертывания программного обеспечения - одну из основных задач DevOps.

Docker упаковывает ПО в стандартизованные блоки, которые называются контейнерами. Каждый контейнер включает все необходимое для работы приложения: библиотеки, системные инструменты, код и среду исполнения. Благодаря Docker можно быстро развертывать и масштабировать приложения в любой среде и сохранять уверенность в том, что код будет работать.

Docker - инструмент контейнеризации приложений с открытым исходным кодом, который делает универсальным процесс развертывания и управления программным обеспечением в изолированных контейнерах. Эти контейнеры могут в дальнейшем запускаться в различных средах - локальной инфраструктуре, облаке и пр.

Разрабатывается и поддерживается одноимённой компанией-стартапом, распространяется в двух редакциях — общественной и для организаций по проприетарной лицензии. Написан на языке Go.

Докер — универсальный способ доставки приложений на машины (локальный компьютер или удаленные сервера) и их запуска в изолированном окружении.

Docker работает по принципу, похожему на игровую приставку GameBoy. Чтобы поиграть, ты просто вставляешь картридж в приставку, и игра сразу же работает. Ты можешь поделиться своей игрой с друзьями, передав всего лишь картридж, который они вставят в приставку, и сразу же смогут играть

Docker позволяет запускать своё ПО также просто - кто угодно, у кого установлен Docker может запустить ваше приложение, выполнив для этого всего несколько команд.

Раньше, вы, создавая приложения, к примеру на PHP, устанавливали локально PHP, MySQL, возможно, NodeJS, при этом устанавливая зависимости в виде нужных расширений и библиотек. И, в случае передачи вашего скрипта другому разработчику, ему требовалось настраивать аналогичное окружение аналогичных версий, иметь аналогичные расширения и конфигурацию, чтобы успешно запустить ваше приложение.

Сейчас же, при использовании Докера, такой проблемы не возникнет в принципе. Теперь вам достаточно иметь установленную программу Docker, которая по одной вашей команде установит окружение, описанное в конфиге для запуска вашего приложения.

Docker чем-то похож на виртуальную машину, он запускает виртуальную операционную систему, имеющую чрезвычайно маленький вес, в течение нескольких минут. Докер, имеет особые образы ПО, запускающиеся в виртуальной среде, не создавая полную копию ОС.

Когда пользователи будут запускать приложение через Docker - конфигурация будет собрана автоматически, и код будет выполняться ВСЕГДА.

Как разработчик, теперь вы не должны волноваться о том, на какой системе будет запущено ваше приложение.

Как пользователь, вам не нужно волноваться о том, что вы скачаете неподходящую версию ПО. В Докере эта программа будет запущена в аналогичных условиях, при которых это приложение было разработано, потому, исключается вероятность получить какую-то новую, непредвиденную ошибку.

Для пользователя все действия сводятся к принципу подклочи и играй.

#### Docker Image

Docker образ (он же Docker Image), похож на Game Boy картридж. Docker имеет различные образы ПО: ubuntu, php, nodejs, и т.д.

Dockerhub.com - основной репозиторий Docker-образов, на котором можно посмотреть весь их список и подобрать нужный образ для вашей программы.

Это как поездка за новым картриджем в магазин, только намного быстрее :).

Образ занимает небольшое количество ресурсов по сравнению с реальной установкой необходимого для работы ПО. Например, чистый образ Ubuntu при установке из Docker-образа, весит всего 74 МБ.

#### Docker контейнер

Теперь представьте, что мы обновили нашу приставку с Game Boy на GameCube. Игры хранятся на диске, который предназначен только для чтения самого образа игры. А прочие файлы (сохранения, кеш и т.д.) сохраняются внутри самой приставки, локально.

Так же, как и игра на диске, исходный Docker Image (образ) - неизменяемый.

Docker контейнер - это экземпляр запущенного образа. Аналогично тому, что вы вставляете диск в приставку, после чего игра начинается.

А сам образ игры никак не модифицируется, все файлы, содержащие изменения хранятся где-то локально на приставке.

Запуск Docker контейнера соответствует тому, что вы играете в свою Gamecube игру. Docker запускает ваш образ в своей среде, аналогично тому, как Gamecube запускает игру с диска, не модифицируя оригинальный образ, а лишь сохраняя изменения и весь прогресс в какой-то песочнице.

После выполнения нужных операций Docker-контейнер завершает работу. Это похоже на режим сохранения энергии в новых игровых консолях - если вы не совершаете действий какое-то время, то система выключается автоматически.

Каждый раз, когда вы будете выполнять команду `docker run`, будет создаваться новый контейнер. Теперь, внутри этого контейнера можно выполнять любые команды, применимые к Ubuntu. Вы же можете представлять это как мини виртуальную машину, условно, к консоли которой мы подключились по SSH.

Docker контейнер является полностью независимым от системы хоста, из которой он запускался. Как изолированная виртуальная машина. И в ней вы можете производить любые изменения, которые никак не повлияют на основную операционную систему.

Это аналогично тому, как, если бы вы играли в Mario Kart на приставке Gamecube, и неважно, что вы делаете в игре, вы никак не сможете изменить само ядро игры, или изменить информацию, записанную на диске.

Контейнер является полностью независимым и изолированным от основной операционной системы, аналогично виртуальной операционной системе. Вы можете вносить любые изменения внутри виртуалки, и никакие из этих изменений не повлияют на основную операционную систему.

#### DockerFile

Docker позволяет делиться с другими средой, в которой ваш код запускался и помогает в её простом воссоздании на других машинах.



Dockerfile - это обычный конфигурационный файл, описывающий пошаговое создание среды вашего приложения. В этом файле подробно описывается, какие команды будут выполнены, какие образы задействованы, и какие настройки будут применены.

С *Dockerfile* вы сможете подробно описать инструкцию, по которой будет воссоздано конкретное состояние. И делается это довольно-таки просто и интуитивно понятно.

#### Монтирование локальной директории в Docker-контейнер

Монтирование директории в Docker контейнер - это предоставление доступа контейнеру на чтение содержимого вашей папки из основной операционной системы. Помимо чтения из этой папки, так же, контейнер может её изменять, и такая связь является двусторонней: при изменении файлов в основной ОС изменения будут видны в контейнере, и наоборот.

Монтирование директории в контейнер позволяет ему читать и писать данные в эту директорию, изменяя её состояние.

Монтирование папки позволяет вам изменять файлы вашей основной системы прямо во время работы внутри Docker контейнера.

Это удобная особенность, которая позволяет нам редактировать код в редакторе на основной ОС, а изменения будут сразу же применяться внутри контейнера.

#### Docker Volumes

Docker Volumes - что-то похоже на карты памяти, которые содержат данные для игры. Точно так же и Docker Volume может быть прикреплён к любому из контейнеров. С Docker Volume вы имеете контейнер, который хранит постоянные данные где-то на вашем компьютере (это актуально, потому что после завершения работы контейнер удаляет все пользовательские данные, не входящие в образ). Вы можете прикрепить Volume-данные к любому из запущенных контейнеров.

Вместо того, чтобы каждый раз, при запуске контейнера, писать, какие из папок вы хотите смонтировать, вы просто можете создать один контейнер с общими данными, который потом будете прикреплять.

Каждый раз, когда вы собираете образ, он кешируется в отдельный слой. Ввиду того, что образы являются неизменяемыми, их ядро никогда не модифицируется, потому что применяется система кеширования, которая нужна для увеличения скорости билдинга.

Слой Образа Докера это как точка сохранения в игре Super Mario. Если вы хотите изменить какие-то вещи, произошедшие до этой точки сохранения, то вам придётся перезапустить этот уровень полностью. Если вы хотите продолжить прогресс прохождения, вы можете начать с того места, где остановились.

Docker начинает кешировать с "того места, где остановился" во время билдинга Dockerfile. Если в Dockerfile не было никаких изменений с момента последнего билдинга, то образ будет взят полностью из кеша. Если же вы измените какую-то строку в Dockerfile - кеш будет взят только тех слоёв команд, которые находятся выше изменённой команды.

#### Docker-Compose

Docker-compose это как дирижёр оркестра, где ваш оркестр - это набор контейнеров, которыми нужно управлять. Каждый контейнер имеет отдельную задачу, как и музыкальные инструменты в разных частях песни. Docker Compose управляет контейнерами, запускает их вместе, в нужной последовательности, необходимой для вашего приложения.

Docker-compose организывает совместный запуск контейнеров, каждый из которых имеет свою конкретную задачу.

Docker-compose написан в формате YAML который по своей сути похож на JSON или XML, но имеет более удобный формат. В формате YAML имеют значения

пробелы и табуляции, именно пробелами отделяются названия параметров от их значений.

### Микросервисы

Docker часто используется как инструмент для написания микросервисов.

Микросервисы - это архитектурный шаблон проектирования который следует философии "разделения ответственности".

Каждый микросервис выполняет одну конкретную задачу в изолированной среде, он должен управлять своей собственной локальной информацией, и быть максимально независимым от общей системы. В основном, каждый из микросервисов имеет собственную, отдельную базу данных (если она нужна). Он никак не связан с другими существующими микросервисами. Вместе они образуют приложение.

Это позволяет создавать максимально гибкие и легко масштабируемые приложения.

Идея разделения ответственности предоставляет преимущество в том, что команда разработчиков может работать параллельно, фокусируясь над разными компонентами.

В основном, микросервисы имеют канал коммуникации между собой, в виде REST API, который возвращает данные в формате JSON, или другом формате.

Использование микросервисов позволяет быстро масштабироваться под большой нагрузкой, без масштабирования остальных частей вашего приложения. То есть, такая архитектура позволяет вам масштабировать систему компонентно, там, где это требуется. В случае монолитной, единой системы, вам придётся масштабировать всё приложение.

### Роль Docker в DevOps

Docker- это платформа, которая идеально подходит для DevOps. Преимущества, которые Docker предлагает среде DevOps, сделали его незаменимым инструментом в цепочке инструментов.

Причина, по которой Docker так хорош для DevOps, заключается в его преимуществах и вариантах использования контейнеризации приложений, поддерживающих разработку и быстрый выпуск. DevOps в основном используется для решения проблем "Dev" и "Ops", и Docker, решает большинство из них, главным из которых является то, что он может работать на любой машине. Таким образом, это позволяет всем командам сотрудничать и работать эффективно и результативно.

Docker позволяет создавать общие среды разработки, производства и промежуточной обработки, тем самым обеспечивая плавный контроль общих изменений. Если вы хотите вернуться к предыдущей версии, вы можете сделать это в любое время, поскольку все среды становятся более похожими. Docker гарантирует, что, если функция функционирует в среде разработки, она также будет работать в рабочей и промежуточной среде.

### Список используемых источников:

- 1 BD School Школа больших данных  
<https://www.bigdataschool.ru/wiki/devops>
- 2 DevOps и AWS.Облако AWS. Инструментальные средства и инфраструктурные ресурсы для практиков DevOps  
<https://aws.amazon.com/ru/devops/what-is-devops/>
- 3 Ростелеком Бизнес Редакция GlobalSIO Эра DevOps принять как данность, адаптироваться и применять <https://globalcio.ru/discussion/9145/>
- 4 Фоменко Александр. Docker самый простой и понятный туториал <https://badtry.net/docker-tutorial-dlia-novichkov-rassmatrivaiem-docker-tak-iesli-by-on-bylighrovoi-pristavkoi/>