

## УСТРОЙСТВО СКАНИРОВАНИЯ ЦВЕТА И МОБИЛЬНОЕ ПРОГРАММНОЕ СРЕДСТВО АНАЛИЗА ЦВЕТОВОЙ ПАЛИТРЫ И ПОДБОРА ЦВЕТА

*Бондарь Е.А., студент гр.751006, Деменковец Д.В., аспирант*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Деменковец Д.В. – ст. преподаватель, м.т.н.*

**Аннотация.** В результате работы было сконструировано и запрограммировано устройство сканирования цвета, а также разработано мобильное программное средство анализа цветовой палитры и подбора цвета. Устройство и мобильное приложение взаимодействуют с помощью технологии Bluetooth. Данная система предназначена для того, чтобы пользователь имел возможность сканировать цвет любой ровной поверхности и в результате получить палитру цветов, которые необходимо смешать для того, чтобы получить отсканированный цвет.

**Ключевые слова.** Мобильное приложение, устройство, сканирование, цвет, субтрактивный синтез, мобильное программное средство, палитра, генерация.

В производстве строительных красок, а также порой в быту, возникает необходимость в подборе красок определенного цвета. Встречается ситуация, когда покупатель приходит в магазин и просит подобрать для него краску, которая была бы точно такой же, как и какой-либо другой элемент интерьера. Но, к сожалению, покупатель не знает номер и название цвета краски того самого элемента интерьера. С этой задачей может справиться устройство сканирования цвета и мобильное программное средство анализа цветовой палитры и подбора цвета. Кроме того, если в магазине не будет в наличии нужного цвета, мобильное приложение подберет палитру красок, которые необходимо смешать для достижения заданного цвета.

К немногочисленным аналогам данного устройства можно отнести: RM-Basf (Colortronic 1, 2), SpiesHecker (ColorDialog), DuPont (ChromaVision) и другие. Все представленные приборы специализируются на компьютерном подборе краски для автомобилей. Кроме представленных аналогов существует колориметр RAL COLORCATCH NANO, который также используется для определения цвета поверхности, но его существенным недостатком является то, что устройство сканирования цвета должно быть всегда соединено кабелем с мобильным телефоном.

Разработанное устройство сканирования цвета представляет собой конусообразную воронку диаметром 6 см. со встроенными светодиодами, датчиком и программируемой платой. Форма воронки даёт устройству возможность сканировать исключительно цвет поверхности, без воздействия внешнего света, чтобы исключить искажение сигнала выходного цвета. Светодиоды необходимы для того, чтобы максимально точно определить цвет поверхности. Степень освещенности светодиодами определяется методом подбора, учитывая отражающие свойства поверхности. Функция устройства заключается в том, чтобы принять на вход значение трех составляющих цвета поверхности (красный, зеленый, синий) и по интерфейсу Bluetooth передать эти значения цвета мобильному программному средству для последующей обработки. Схема передачи RGB-значений цвета с устройства сканирования цвета мобильному программному средству представлена на рисунке 1.

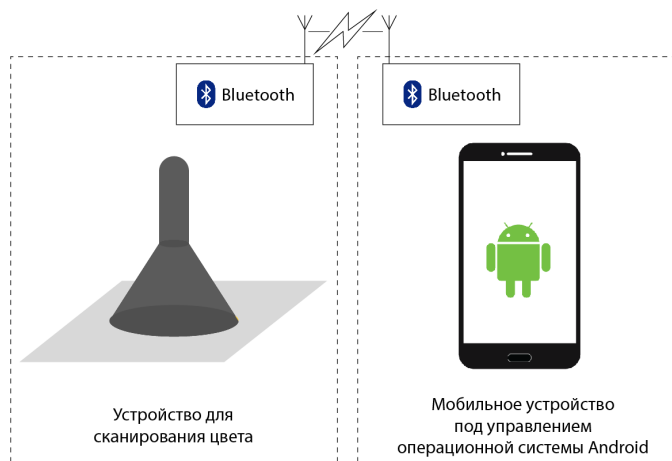


Рисунок 1 – Схема передачи RGB-значений цвета с устройства сканирования цвета мобильному программному средству

Мобильное программное средство представляет собой Android-приложение. На стартовом экране имеются три кнопки: «Сканировать» - кнопка, для сканирования цвета поверхности; «Библиотека» - кнопка, ведущая пользователя к списку сохраненных ранее цветов; «Калибровка» - кнопка, для калибровки устройства сканирования цвета. При нажатии пользователем кнопки «Сканировать», на экране появляется подсказка с изображением и описанием того, как необходимо расположить устройство сканирования цвета, чтобы цвет был определен максимально точно. После сканирования цвета поверхности устройством, на экране мобильного приложения отображается следующая информация: визуализация отсканированного цвета поверхности; номер отсканированного цвета в таблицах CMYK, RGB и RAL; возможность добавить цвет в библиотеку; визуализация, номера в таблице RAL и процентное соотношение цветов, которые необходимо смешать, для получения отсканированного цвета; визуализация и номера цветов в таблице RAL, похожих на отсканированный цвет; возможность поделиться отсканированным цветом в социальных сетях Facebook, Twitter и ВКонтакте.

Для определения цветов, которые необходимо смешать, чтобы получить отсканированный цвет, используется субтрактивная схема формирования цвета (CMYK) - это получение цвета за счёт вычитания из белого света его спектральных составляющих. То есть, поскольку устройство передает мобильному приложению RGB-значение цвета, то для дальнейших манипуляций с цветом, его необходимо перевести в систему CMY следующим образом:  $RGB(a, b, c) = CMY(255-a, 255-b, 255-c)$ . После чего, полученный результат следует нормализовать. В результате применения алгоритма смешения цветов, пользователь получает палитру цветов в системе RAL с процентным соотношением каждого цвета. Субтрактивная схема формирования цвета представлена на рисунке 2.

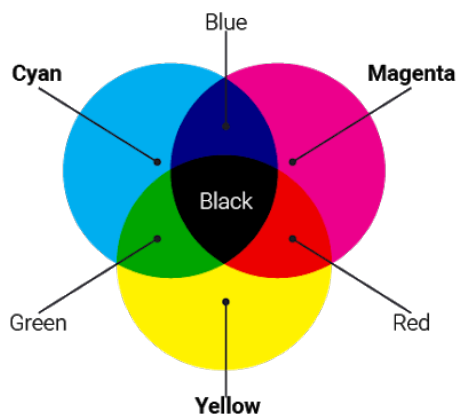


Рисунок 2 – Субтрактивная схема формирования цвета

К преимуществам устройства сканирования цвета и мобильного программного средства можно отнести:

1. Удобная форма и компактность (из-за небольшого размера устройства сканирования цвета, его можно переносить даже в кармане);
2. Портативность (при наличии портативного блока питания, устройство сканирования цвета можно использовать в любом, подходящем по требованиям, месте);
3. Низкая стоимость.

Недостатками устройства сканирования цвета и мобильного программного средства являются:

1. Особые требования к использованию (для сканирования цвета устройству необходима ровная поверхность, размером минимум 6\*6 см);
2. Для работы устройства сканирования цвета, ему необходимо быть подключенным к портативному блоку питания, что утяжеляет конструкцию устройства;
3. Погрешность в сканировании цвета поверхности.

Устройство сканирования цвета и мобильное программное средство анализа цветовой палитры и подбора цвета удобно использовать для генерации палитры цветов, которые нужно смешать, чтобы получить необходимый цвет, но пока человеческий глаз по-прежнему самый точный прибор для подбора красок.

**Список использованных источников:**

1. Компьютерный подбор краски для авто [Электронный ресурс]. – Режим доступа: <http://avtomalaria.by/podbor/computerniy-podbor-kraski>. – Дата доступа: 20.03.2021.
2. Колориметр RAL COLORCATCH NANO [Электронный ресурс]. – Режим доступа: <https://ral.ru/colorcatch>. – Дата доступа: 20.03.2021.
3. Субтрактивное смешение цветов [Электронный ресурс]. – Режим доступа: <http://knowledge.systema-stage.ru/encyclopedia/subtraktivnoe-smeshenie-cvetov>. – Дата доступа: 20.03.2021.

# МЕТОДЫ ЗАЩИТЫ СЕРВЕРОВ ОТ DDoS АТАК

Чудук А.В.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Серебряная Л.В. – к. т. н., доцент каф. ПОИТ

Одним из самых простых в исполнении и одновременно эффективным типов атак на серверы является так называемая DDoS-атака. Это целый спектр атак, направленных на перегрузку сервера. Данная статья содержит классификацию DDoS-атак по уровням OSI, их последствия и способы противодействия им.

Для рассмотрения защиты от DDoS атак необходимо дать определение данному типу угроз. Для этого необходимо разобраться с её историческим предшественником – DoS атакой. DoS (denial of service) – это атака на приложение, суть которой заключается в отправлении чрезмерного количества запросов на сервер из одного адреса. Атаки данного типа довольно просто локализовать и заблокировать, так как они исходят от одного адреса. DDoS (distributed denial of service) отличается от DoS атаки тем, что запросы в ней исходят не от одного адреса, а от многих. Соответственно, DDoS атаку невозможно предотвратить добавлением одного источника запросов в список игнорируемых адресов, так как для локализации необходимо составить полный список адресов, от которых исходит чрезмерное количество запросов.

Широкая распространённость данного типа атак обусловлена относительно простой реализацией и низким требованием к аппаратуре. Защититься от DDoS атаки полностью сложно, так как пройдёт некоторое время, пока она не будет локализована и адреса не будут добавлены в черный список. Существуют способы полной защиты от DDoS атак, однако они накладывают большие ограничения на пропускную способность сети.

В основном DDoS атаки проводятся с помощью сторонних компьютеров, заражённых ранее вредоносным программным обеспечением. По этой причине аппаратура злоумышленника может не быть высокого качества, так как она отвечает за синхронизацию заражённых устройств в момент запуска атаки.

Атаки данного типа опасны тем, что они блокируют физические сервера. Из-за превышения времени отклика сервера, компания теряет потенциальных и уже существующих клиентов и, соответственно, прибыль. Злоумышленники также применяют DDoS атаки для маскировки атак иного типа [1].

По данным отчета «2016 Verizon data breach incident report» (DBIR), в 2016 году количество DDoS атак заметно выросло. Так же была замечена тенденция на расширение списка целей DDoS атак – в 2016 году веб-приложения развлекательной направленности были наиболее подвержены данному типу атак. Так же было замечено увеличение числа DDoS атак смешанного типа – одновременного влияния на несколько уязвимостей. По данным «Лаборатории Касперского», с апреля по июнь 2020 года количество DDoS атак во всём мире увеличилось в три раза по сравнению с аналогичным периодом прошлого года [2].

Существует множество классификаций DDoS атак

DDoS атаки классифицируются по уровням OSI:

1. Физический. Уровень специализируется на передаче данных в двоичном коде. Результат успешных атак этого уровня – поломка оборудования. Вид защиты – постоянный мониторинг работы физического оборудования.

2. Канальный. Отвечает за взаимодействие сети на физическом уровне, оперируя кадрами по протоколам 802.3 и 802.5 с помощью контроллеров, точек доступа и мостов. Пример DDoS атаки: MAC-флуд – переполнение пакетами данных сетевых коммутаторов. Последствия: блокировка портов сервера. Защита: проверка аутентификации MAC адресов и фильтрация ненадёжных или «флудящих» адресов.

3. Сетевой. На этом уровне происходит маршрутизация между сетями посредством передачи пакетов с информацией. Пример DDoS атаки: ICMP-флуд – постоянное посылание ping запроса серверу нарушителями. Последствия: из-за постоянного ответа на ping запросы у сервера не хватает времени на обработку полезных пакетов и, соответственно, время отклика сервера повышается на порядки. Защита: отключение или ограничение ICMP запросов.

4. Транспортный. Обеспечивает стабильную передачу данных между узлами сети. Пример DDoS атаки: SYN-флуд – отправка большого количества SYN-запросов. Последствия: превышение количества доступных подключений, сбой в работе сетевого оборудования. Защита: blackholing – метод фильтрации трафика на уровне провайдера, заключающийся в игнорировании пакетов злоумышленников.

5. Сеансовый. Отвечает за управление установкой и завершением соединения, синхронизацией сеансов связи в рамках операционной системы через сеть. Пример DDoS атаки: использование

уязвимостей программного обеспечения Telnet-сервера. Последствия: блокировка управления свитчем для администратора. Защита: обновление прошивок аппаратного обеспечения.

6. Представительский. На этом уровне происходит передача данных от источника к получателю. Пример DDoS атаки: подложные SSL-запросы. Последствия: остановка приёма SSL соединений или автоматическая перегрузка системы. Защита: фильтрация трафика, вынесение функционала SSL на отдельные серверы.

7. Прикладной. Обеспечивает создание пакетов данных, соединение устройств и доступ к данным. Пример DDoS атаки: HTTP POST флуд. Последствия: нехватка ресурсов для обработки запросов. Защита: мониторинг системы в целом и ПО в частности для быстрого реагирования, идентификации и блокировки злоумышленника.

Существуют также и иные меры защиты от DDoS атаки, которые могут работать вместе с вышеперечисленными:

- Уклонение. Расположение ресурсов на нескольких серверах и/или в различных дата-центрах позволяет при атаке одного из них сохранить работоспособность приложения. Выгодно для большого объёма данных и высокого риска DDoS атаки. Стоимость данного решения довольно высока, однако в совокупности с распределёнными базами данных данный тип защит позволяет минимизировать потери данных и производительности.

- Аренда физического сервера с защитой от DDoS атаки. Существуют компании, предоставляющие уже готовые сервера с защитой от данной проблемы. Их стоимость высока, но это специализированные аппаратно-программные решения, проверенные другими заказчиками и сертифицированные для выполнения соответствующих задач. Использование данного типа защиты выгодно для ситуаций с высокой вероятностью DDoS атаки при отсутствии или нецелесообразности разработки своего решения.

- Captcha. Одним из способов защиты серверов от DDoS атаки на прикладном уровне для серверов является установление дополнительных действий, необходимых для выполнения пользователем, которые выполняют вычисление прикладного ПО, рассчитанного на выполнение DDoS атаки. Данный способ позволяет снизить нагрузку на наиболее ресурсоёмкие операции, такие как добавление новых данных в базу с помощью HTTP метода POST.

- Ограничение запросов. Увеличение ответа на запрос позволяет снизить нагрузку на сеть и аппаратное обеспечение сервера. Так же в данный тип защит включается и ограничение количества запросов от одного или нескольких источников. Данный способ уменьшает удобство пользования сайтом для обычных пользователей, но очень эффективен для защиты от DDoS атаки.

- Ответная реакция. Существуют способы развернуть трафик обратно злоумышленнику. В таком случае все последствия получит сам преступник. Однако для выполнения данного способа защиты необходимо высококачественно аппаратное оборудование.

- Коллективная защита. Существуют способы распространения списка подозрительных адресов. В этом случае банк данных пополняется из различных компаний, подвергшихся атаке [3].

Для качественной защиты от DDoS атаки необходим комплекс защитных мер, так как они могут быть направлены на различные уровни взаимодействия сети.

- Изоляция. Самый простой и надёжный способ защитить сервер от DDoS атаки – это изолировать сервер от сети Интернет. Однако, данный метод применим только к локальным серверам и не может использоваться для веб-серверов.

Так же руководству компании необходимо правильно оценивать потери репутации, финансов от успешного прохождения атаки на серверы. Чем меньше капитализация предприятия, тем меньше шанс быть подвергнутым атаке со стороны злоумышленников и, соответственно, снижается потребность в сложной и ресурсоёмкой защите. В таких случаях может быть достаточно правильно настроенных фаерволов и маршрутизаторов.

DDoS атаки – это обширная проблема в сфере защиты данных. Атаки данного рода получили широкое распространение в последнее десятилетие по следующим причинам: они нетребовательны к аппаратному обеспечению, информация о них содержится в открытом доступе и, несмотря на простоту, данный тип угроз имеет высокую эффективность. Однако, несмотря на эффективность данного рода атак, существует широкий спектр методов защиты от них, каждая из которых имеет своё применение и свою целевую аудиторию.

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

1. Kaspersky DDoS Protection [Электронный ресурс] – Режим доступа: <https://www.kaspersky.ru/enterprise-security/ddos-protection>.

2. Статья ««Лаборатория Касперского» отмечает резкий скачок DDoS-атак во втором квартале 2020 года» [Электронный ресурс] – Режим доступа: [https://www.kaspersky.ru/about/press-releases/2020\\_laboratoriya-kasperskogo-otmechaet-rezkii-skachok-ddos-atak-vo-vтором-kvartale-2020-goda](https://www.kaspersky.ru/about/press-releases/2020_laboratoriya-kasperskogo-otmechaet-rezkii-skachok-ddos-atak-vo-vтором-kvartale-2020-goda).

3. Статья «Что такое DDoS атака. Настройка эффективной защиты от DDoS атак на сервер» [Электронный ресурс] – Режим доступа: <https://rigweb.ru/support/khosting-i-domeny/typy-ddos-atak-i-sposoby-zashchity-ot-nikh>.

# МЕТОДЫ ЗАЩИТЫ СЕРВЕРОВ ОТ DDoS АТАК

Чудук А.В.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Серебряная Л.В. – к. т. н., доцент каф. ПОИТ

Одним из самых простых в исполнении и одновременно эффективным типов атак на серверы является так называемая DDoS-атака. Это целый спектр атак, направленных на перегрузку сервера. Данная статья содержит классификацию DDoS-атак по уровням OSI, их последствия и способы противодействия им.

Для рассмотрения защиты от DDoS атак необходимо дать определение данному типу угроз. Для этого необходимо разобраться с её историческим предшественником – DoS атакой. DoS (denial of service) – это атака на приложение, суть которой заключается в отправлении чрезмерного количества запросов на сервер из одного адреса. Атаки данного типа довольно просто локализовать и заблокировать, так как они исходят от одного адреса. DDoS (distributed denial of service) отличается от DoS атаки тем, что запросы в ней исходят не от одного адреса, а от многих. Соответственно, DDoS атаку невозможно предотвратить добавлением одного источника запросов в список игнорируемых адресов, так как для локализации необходимо составить полный список адресов, от которых исходит чрезмерное количество запросов.

Широкая распространённость данного типа атак обусловлена относительно простой реализацией и низким требованием к аппаратуре. Защититься от DDoS атаки полностью сложно, так как пройдёт некоторое время, пока она не будет локализована и адреса не будут добавлены в черный список. Существуют способы полной защиты от DDoS атак, однако они накладывают большие ограничения на пропускную способность сети.

В основном DDoS атаки проводятся с помощью сторонних компьютеров, заражённых ранее вредоносным программным обеспечением. По этой причине аппаратура злоумышленника может не быть высокого качества, так как она отвечает за синхронизацию заражённых устройств в момент запуска атаки.

Атаки данного типа опасны тем, что они блокируют физические сервера. Из-за превышения времени отклика сервера, компания теряет потенциальных и уже существующих клиентов и, соответственно, прибыль. Злоумышленники также применяют DDoS атаки для маскировки атак иного типа [1].

По данным отчета «2016 Verizon data breach incident report» (DBIR), в 2016 году количество DDoS атак заметно выросло. Так же была замечена тенденция на расширение списка целей DDoS атак – в 2016 году веб-приложения развлекательной направленности были наиболее подвержены данному типу атак. Так же было замечено увеличение числа DDoS атак смешанного типа – одновременного влияния на несколько уязвимостей. По данным «Лаборатории Касперского», с апреля по июнь 2020 года количество DDoS атак во всём мире увеличилось в три раза по сравнению с аналогичным периодом прошлого года [2].

Существует множество классификаций DDoS атак

DDoS атаки классифицируются по уровням OSI:

8. Физический. Уровень специализируется на передаче данных в двоичном коде. Результат успешных атак этого уровня – поломка оборудования. Вид защиты – постоянный мониторинг работы физического оборудования.

9. Канальный. Отвечает за взаимодействие сети на физическом уровне, оперируя кадрами по протоколам 802.3 и 802.5 с помощью контроллеров, точек доступа и мостов. Пример DDoS атаки: MAC-флуд – переполнение пакетами данных сетевых коммутаторов. Последствия: блокировка портов сервера. Защита: проверка аутентификации MAC адресов и фильтрация ненадёжных или «флудящих» адресов.

10. Сетевой. На этом уровне происходит маршрутизация между сетями посредством передачи пакетов с информацией. Пример DDoS атаки: ICMP-флуд – постоянное посылание ping запроса серверу нарушителями. Последствия: из-за постоянного ответа на ping запросы у сервера не хватает времени на обработку полезных пакетов и, соответственно, время отклика сервера повышается на порядки. Защита: отключение или ограничение ICMP запросов.

11. Транспортный. Обеспечивает стабильную передачу данных между узлами сети. Пример DDoS атаки: SYN-флуд – отправка большого количества SYN-запросов. Последствия: превышение количества доступных подключений, сбой в работе сетевого оборудования. Защита: blackholing – метод фильтрации трафика на уровне провайдера, заключающийся в игнорировании пакетов злоумышленников.

12. Сеансовый. Отвечает за управление установкой и завершением соединения, синхронизацией сеансов связи в рамках операционной системы через сеть. Пример DDoS атаки: использование

уязвимостей программного обеспечения Telnet-сервера. Последствия: блокировка управления свитчем для администратора. Защита: обновление прошивок аппаратного обеспечения.

13. Представительский. На этом уровне происходит передача данных от источника к получателю. Пример DDoS атаки: подложные SSL-запросы. Последствия: остановка приёма SSL соединений или автоматическая перегрузка системы. Защита: фильтрация трафика, вынесение функционала SSL на отдельные серверы.

14. Прикладной. Обеспечивает создание пакетов данных, соединение устройств и доступ к данным. Пример DDoS атаки: HTTP POST флуд. Последствия: нехватка ресурсов для обработки запросов. Защита: мониторинг системы в целом и ПО в частности для быстрого реагирования, идентификации и блокировки злоумышленника.

Существуют также и иные меры защиты от DDoS атаки, которые могут работать вместе с вышеперечисленными:

- Уклонение. Расположение ресурсов на нескольких серверах и/или в различных дата-центрах позволяет при атаке одного из них сохранить работоспособность приложения. Выгодно для большого объёма данных и высокого риска DDoS атаки. Стоимость данного решения довольно высока, однако в совокупности с распределёнными базами данных данный тип защит позволяет минимизировать потери данных и производительности.

- Аренда физического сервера с защитой от DDoS атаки. Существуют компании, предоставляющие уже готовые сервера с защитой от данной проблемы. Их стоимость высока, но это специализированные аппаратно-программные решения, проверенные другими заказчиками и сертифицированные для выполнения соответствующих задач. Использование данного типа защиты выгодно для ситуаций с высокой вероятностью DDoS атаки при отсутствии или нецелесообразности разработки своего решения.

- Captcha. Одним из способов защиты серверов от DDoS атаки на прикладном уровне для серверов является установление дополнительных действий, необходимых для выполнения пользователем, которые выполняют вычисление прикладного ПО, рассчитанного на выполнение DDoS атаки. Данный способ позволяет снизить нагрузку на наиболее ресурсоёмкие операции, такие как добавление новых данных в базу с помощью HTTP метода POST.

- Ограничение запросов. Увеличение ответа на запрос позволяет снизить нагрузку на сеть и аппаратное обеспечение сервера. Так же в данный тип защит включается и ограничение количества запросов от одного или нескольких источников. Данный способ уменьшает удобство пользования сайтом для обычных пользователей, но очень эффективен для защиты от DDoS атаки.

- Ответная реакция. Существуют способы развернуть трафик обратно злоумышленнику. В таком случае все последствия получит сам преступник. Однако для выполнения данного способа защиты необходимо высококачественное аппаратное оборудование.

- Коллективная защита. Существуют способы распространения списка подозрительных адресов. В этом случае банк данных пополняется из различных компаний, подвергшихся атаке [3].

Для качественной защиты от DDoS атаки необходим комплекс защитных мер, так как они могут быть направлены на различные уровни взаимодействия сети.

- Изоляция. Самый простой и надёжный способ защитить сервер от DDoS атаки – это изолировать сервер от сети Интернет. Однако, данный метод применим только к локальным серверам и не может использоваться для веб-серверов.

Так же руководству компании необходимо правильно оценивать потери репутации, финансов от успешного прохождения атаки на серверы. Чем меньше капитализация предприятия, тем меньше шанс быть подвергнутым атаке со стороны злоумышленников и, соответственно, снижается потребность в сложной и ресурсоёмкой защите. В таких случаях может быть достаточно правильно настроенных фаерволов и маршрутизаторов.

При высоких рисках целесообразно организовать специализированный отдел, который будет заниматься защитой данных. В таких случаях может быть полезно не использование существующих коммерческих решений, а разработка своих собственных.

DDoS атаки – это обширная проблема в сфере защиты данных. Атаки данного рода получили широкое распространение в последнее десятилетие по следующим причинам: они нетребовательны к аппаратному обеспечению, информация о них содержится в открытом доступе и, несмотря на простоту, данный тип угроз имеет высокую эффективность. Однако, несмотря на эффективность данного рода атак, существует широкий спектр методов защиты от них, каждая из которых имеет своё применение и свою целевую аудиторию.

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

1. Kaspersky DDoS Protection [Электронный ресурс] – Режим доступа: <https://www.kaspersky.ru/enterprise-security/ddos-protection>.

2. Статья ««Лаборатория Касперского» отмечает резкий скачок DDoS-атак во втором квартале 2020 года» [Электронный ресурс] – Режим доступа: [https://www.kaspersky.ru/about/press-releases/2020\\_laboratoriya-kasperskogo-otmechaet-rezkii-skachok-ddos-atak-vo-vтором-kvartale-2020-goda](https://www.kaspersky.ru/about/press-releases/2020_laboratoriya-kasperskogo-otmechaet-rezkii-skachok-ddos-atak-vo-vтором-kvartale-2020-goda).

3. Статья «Что такое DDoS атака. Настройка эффективной защиты от DDoS атак на сервер» [Электронный ресурс] – Режим доступа: <https://rigweb.ru/support/khosting-i-domenu/typy-ddos-atak-i-sposoby-zashchity-ot-nikh>.

# ГРАФИЧЕСКИЙ ПОЛЬЗОВАТЕЛЬСКИЙ ИНТЕРФЕЙС НА ОСНОВЕ REDUX

Щербаков М.А.

Белорусский государственный университет информатики и радиоэлектроник  
г. Минск, Республика Беларусь

Серебряная Л.В. – канд. тех. наук

В данном докладе будет рассмотрено использование перспектив использования шаблона дизайна Redux в настольных графических приложениях. Также будут освещены преимущества использования данного подхода к проектированию прикладных приложений в условиях современной вычислительной многопоточной техники.

Одним из подходов к решению многих проблем, в частности распараллеливанию задач, вынесению отрисовки графических элементов в отдельный поток, является Redux. Redux – это шаблон дизайна, который предполагает единый источник правды (SSoT) для бизнес-логики и графических компонентов, называемый «Store», доступ на чтение, к которому имеют как графические компоненты, так и бизнес-логика, а доступ на запись отсутствует вовсе. Вместо записи в Store напрямую используется система событий и их обработчиков. События называются «Actions», а обработчики называются «Reducers». Actions представляют собой обычные объекты с полями, одним из обязательных полей которых является поле «type», т.е. поле типа события. Reducers – это обычные чистые функции, которые принимают Store (оно же глобальное состояние) в качестве первого аргумента, а в качестве второго принимают Actions. Единственное предназначение Reducers заключается в изменении состояния Store в соответствии с Actions.

При проектировании графических компонент, нужно предусмотреть некоторый общий стандартный интерфейс, который будет использоваться для уведомления компонент об изменениях состояний Store. При изменении состояния Store вызывается специальный метод у графических компонент, который сообщает, что состояние было изменено. Компоненты же в свою очередь имеют свободный доступ к этому состоянию, могут считать его и выполнить отрисовку нового содержимого при необходимости. Подход чтения состояния и отсутствие возможности записи напрямую позволяет избежать синхронизации между потоками, которые отрисовывают пользовательский интерфейс, и потоками, в которых выполняются Reducers, бизнес-логика и другая полезная логика программы, что позволяет добиться минимальных подтормаживаний при отрисовке.

Существуют такие важные события, как нажатия кнопок, ввод текста, движение курсора мышки и т.д. Для обработки таких событий существуют Actions. Для каждого типа заводится свой собственный Action, который содержит помимо обязательного поля type еще поля с полезной нагрузкой, например, координатами мышки, зажатыми клавишами и т.д. После этого такое событие ставится в очередь. Этот процесс постановки в очередь выполняется максимально быстро и требует минимальной синхронизации, что позволяет добиться больших скоростей обработки.

Поскольку Reducers используются лишь с одной целью – выполнить изменение состояния в соответствии с подготовленными данными, выполнять в них бизнес-логику приложения нельзя. Для выполнения бизнес-логики приложения используется прослойка между очередью поступивших Actions и соответствующими для Actions обработчиками – «Middleware». Такая прослойка позволяет подписываться на происходящие события и выполнять тяжелые вычисления асинхронно. Достигается это за счет поглощения исходного Actions, постановки тяжеловесной задачи в очередь на выполнение, например, в пул потоков. Обработка продолжается дальше до тех пор, пока не будет выполнено асинхронное или тяжелое вычисление. Как только вычисление будет выполнено, оно заново породит Action со всеми необходимым для Reducer данными.

Рассмотренный шаблон дизайна может быть применен при проектировании приложений с графическим интерфейсом с использованием широкого спектра технологий. Одной из уже реализовавших данный шаблон дизайна является ReactJS.

Redux – мощный шаблон дизайна, позволяющий расширять и пополнять библиотеку графических компонентов быстро и надежно, поскольку ни один из компонентов не изменяет состояние программы напрямую, а лишь имеет доступ на чтение к нему. Также, использование данного шаблона позволяет разделять потоки выполнения графических компонент, изменения глобального состояния, выполнения бизнес-логики.

## Список использованных источников:

[1] Гращенко Л. А. Обобщенная модель угроз информационной безопасности визуальных интерфейсов пользователя // Известия Орловского государственного технического университета. Серия: Информационные системы и технологии. — 2016. — №. 1. — С. 41-45.

[2] Kunle Olukotun. Chip Multiprocessor Architecture - Techniques to Improve Throughput and Latency. — Morgan and Claypool Publishers, 2017. — 154 p. — ISBN 159829122X.

[3] Mario Nemirovsky, Dean M. Tullsen. Multithreading Architecture. — Morgan and Claypool Publishers, 2013. — 1608458555 p. — ISBN 1608458555

# ПРОГРАММА «РОДИТЕЛЬСКИЙ КОНТРОЛЬ» В СТРИМИНГОВЫХ СЕРВИСАХ

*Норейко А.А.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Серебряная Л.В. – кандидат технических наук, доцент*

О возможности создания детских аккаунтов у родителей с их учётных записей. С помощью этого родители смогут фильтровать видеоконтент у детей и не волноваться, что ребёнок посмотрит «не то» кино.

В наши дни поиск нужного кино или видеозаписи для просмотра стало тривиальной задачей. Зачастую дети, заходя на просторы интернета без присмотра со стороны родителей, натываются на видео-контент, содержащий в себе насилие, употребление алкоголя, табачной продукции, наркотиков и т.д. Последствия от такого просмотра без надзора от родителей, которые должны объяснить ребёнку, что хорошо, а что плохо, могут быть удручающими, так как ребёнок без жизненного опыта будет подражать «крутым» на его взгляд персонажам. Для избегания проблем просмотра взрослого контента ребёнком, был изобретен детский фильтр, или, так называемый, родительский контроль.

Родительский контроль — программа в Интернете для предотвращения его предполагаемого негативного воздействия на ребёнка. Ограничение при таком контроле может осуществляться как по времени (определённые часы в течение суток, либо определённые дни), так и по определённым интернет-ресурсам нежелательной направленности. Также, родительский контроль можно разделить на активный и пассивный.

К пассивным видам родительского контроля относятся такие методы как ограничение на время использования компьютера, ограничение на запуск программных продуктов, функции для защиты глаз и слуха, запрет на совершение встроенных интернет-покупок.

К активным видам родительского контроля относятся такие методы как отслеживания контента сайтов, посещаемых ребёнком; отслеживания видео-контента просматриваемым ребёнком; отслеживание местоположения; просмотр контактов, истории звонков, SMS, мессенджеров, загружаемых фото.

К сожалению, большинство стриминговых платформ не имеют функции родительского контроля. Вместо этого они тотально цензурируют доступный видеоконтент для всех пользователей, фактически, решая за зрителей, что они будут смотреть.

Для решения данной проблемы, разработчик диплома предоставит возможность аккаунту родителя создать профиль ребёнка, в который он сможет входить сам без проблем (аккаунт ребёнка уже не сможет создавать дочерний аккаунтов). Доступ к аккаунту ребёнка можно будет получить, если при входе с аккаунта родителя, указать соответствующий пункт (чекпоинт «Детский аккаунт») и нажать на кнопку «Войти». При этом пароль для ребёнка и для родителя отличаются, поэтому ребёнок не сможет войти на чужой профиль, в то время, когда родитель знает оба пароля. Теперь перейдём к основным функциям родительского контроля.

Из активных функций родительского контроля – это возможность просмотра истории просмотров ребёнка, а также истории поиска. Таким образом родитель будет знать, чем интересуется ребёнок и будет уверен в том, что он не смотрит лишнего. Чтобы посмотреть данную информацию, достаточно просто зайти на свой профиль, нажать на кнопку «История» и выбрать пункт «История детского профиля» (при условии, что детский профиль имеется).

Из пассивных функций родительского контроля – это возможность установлении фильтров на потребляемый ребёнком контент. То есть родитель сможет ограничить спектр просматриваемых ребёнком фильмов, и оградить его от бесконтрольного просмотра взрослого контента в одиночестве.

Выбор для фильтрации контента предоставляется на этапе создания аккаунта. Если же родитель решает, что ребёнок готов к просмотру того или иного контента, он сможет изменить настройки фильтра.

Таким образом установка данной программы в приложение- сервис решит вышеуказанные проблемы. Родители не будут волноваться, что ребёнок увидит лишнее, пока они будут заняты, а сам ребёнок сможет наслаждаться имеющимся доступным контентом. И при этом сам контент не будет так жёстко цензурироваться как в аналогичных стриминговых сервисах.

## **Список использованных источников:**

1. [https://ru.wikipedia.org/wiki/Родительский\\_контроль](https://ru.wikipedia.org/wiki/Родительский_контроль) .
2. <https://www.youtube.com/watch?v=5uhC09TCEDM> .
3. <https://help.keenetic.com/hc/ru/articles/360002532180-Функции-родительского-контроля>



# АППАРАТНО-ПРОГРАММНЫЙ КОМПЛЕКС ПО МОНИТОРИНГУ И АНАЛИЗУ СОСТОЯНИЯ ОКРУЖАЮЩЕЙ СРЕДЫ

Гулида М. А.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Видничук В.Н. – м.т.н.

Экология стала важным фактором, влияющим на качество жизнедеятельности человека. В данной работе предлагается аппаратно-программный комплекс на основе Arduino Nano для мониторинга и анализа состояния окружающей среды.

Согласно данным Всемирной Организации Здравоохранения (ВОЗ) 91% населения Земли живет в местах, где качество воздуха не соответствует нормам, рекомендованным ВОЗ. Более четырёх миллионов смертей наступает ежегодно из-за заболеваний, вызванных воздействием тонкодисперсных частиц в загрязненном воздухе [1].

Мониторинг состояния окружающей среды с помощью датчиков позволяет получать наиболее точную информацию о различных показателях загрязненности территории, таких как уровень оксида углерода, оксида азота, диоксида серы, формальдегида, а также уровень радиации. Дальнейший анализ полученных данных позволяет более точно предложить пользователю экологические активности различного рода, выполнение которых позволило бы улучшить данные показатели и как следствие снизить уровень загрязнения воздуха и улучшить качество жизни.

Для решения данной проблемы предлагается аппаратно-программный комплекс по мониторингу и анализу состояния окружающей среды. Данный комплекс предназначен как для индивидуального использования, так и для использования экологическими организациями для контроля уровня загрязнения окружающей среды и своевременной реакции на складывающиеся изменения.

В качестве аппаратной основы данного аппаратно-программного комплекса используется платформа Arduino Nano, построенная на микроконтроллере ATmega328P [2]. Комплекс включает в себя датчики контроля уровня оксида углерода, оксидов тяжелых металлов, формальдегидов, температуры и влажности воздуха, уровня и дозы радиации.

Данные с датчиков аккумулируются на сервере, где происходит их дальнейший анализ. На основании проведенного анализа пользователям предлагаются экологические активности или даются рекомендации, которые следует соблюдать для уменьшения влияния тех или иных загрязнений. Например, согласно руководству ВОЗ в помещениях в целях сведения к минимуму воздействия на здоровье, уровень оксида углерода должен находиться ниже 8-часового значения  $10\text{мг/м}^3$  [3]. Если уровень оксида углерода превышает данное значение в здании рекомендуется сделать влажную уборку, в случае наличия увлажнителя воздуха, воспользоваться им. В случае нахождения на открытой местности, рекомендуется уменьшить поток транспорта, избегать сжигания листьев.

Ключевой особенностью предлагаемого комплекса является то, что он представляет собой платформу для коллективного контроля за экологической обстановкой в определенном регионе. Помимо сбора данных, предлагаемая система реализует их аккумулирование в централизованном хранилище с последующим глубоким анализом характеристик состояния воздуха на покрытой комплексом территории равно, как и динамики изменения этих характеристик во времени. На основе результатов анализа автоматически формируются рекомендации относительно действий, как личных, так и коллективных, которые стоит предпринять пользователям для улучшения экологической обстановки окружающей их среды. Платформа включает в себя элементы социальных сетей как инструмент для эффективной коммуникации и самоорганизации пользователей в принятии коллективных действий в соответствии с рекомендациями данной системой на основе анализа собранных данных. Существующие аналоги реализуют только сбор и отображение данных о состоянии воздуха и лишены как функций анализа и вынесения рекомендаций, так и инструментов для самоорганизации пользователей для принятия коллективных действий, направленных на поддержание и улучшение экологической обстановки окружающей их среды.

## Список использованных источников:

1. World Health Organization. Ambient air pollution: a global assessment of exposure and burden of disease / World Health Organization. – 2016.
2. Arduino Nano (V2.3): User Manual [Electronic Resource] / Arduino. – Mode of access: <https://www.arduino.cc/en/uploads/Main/ArduinoNanoManual23.pdf>. Date of access: 30.03.2021.
3. Всемирная Организация Здравоохранения. Руководство ВОЗ по качеству воздуха в помещениях: избранные загрязняющие вещества / Всемирная Организация Здравоохранения. – 2011.

# ПРОГРАММНОЕ СРЕДСТВО ОБРАБОТКИ ЗВУКОВЫХ СИГНАЛОВ

Биткин Н.С.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Данилова Г.В. – ассистент каф. ПОИТ

Современный рынок программных и аппаратных средств обработки музыки огромен: различные платформы и операционные системы, приложения со множеством различных расширений и дополнений. Поэтому начинающему музыканту необходим простой и понятный пример для ознакомления с огромным числом возможностей по обработке звука.

Звук всегда присутствовал в жизни человека – от первых песен до сложных проектов, основанных на специальной обработке. На сегодняшний день существует огромное многообразие форматов, способов воспроизведения, наложения различных эффектов на звук.

Целью данного проекта является создание средства обработки медиа-файлов, которое будет поддерживать работу с файлами различных форматов, в том числе узкоспециализированными (например, использующиеся игровой индустрией) и предоставлять возможность обработки различными способами (частотные фильтры, наложение эффектов).

Фильтр изменяет сигнал, «убирая» в нем выбранные частоты. Существующие фильтры не идеальны. Полоса пропускания – полоса частот, которую фильтр не затрагивает. Полоса подавления – полоса нежелательных частот. В полосе перехода происходит спад частот. Фильтр ближе к идеальному тем, насколько меньше он искажает полосу пропускания, насколько сильно он подавляет частоты в полосе подавления и насколько узка полоса перехода [1].

В качестве примера фильтра был выбран фильтр Баттеворта первого порядка.

АЧХ (Амплитудно-частотная характеристика) фильтра Баттерворта максимально гладкая на частотах полосы пропускания и снижается практически до нуля на частотах полосы подавления [1]. При отображении частотного отклика фильтра Баттерворта на логарифмической АФЧХ, амплитуда снижается к минус бесконечности на частотах полосы подавления. В случае фильтра первого порядка АЧХ затухает с крутизной -6 децибел на октаву (-20 децибел на декаду) (на самом деле, все фильтры первого порядка независимо от типа идентичны и имеют одинаковый частотный отклик). Для фильтра Баттерворта второго порядка АЧХ затухает на -12 дБ на октаву, для фильтра третьего порядка – на -18 дБ и так далее. АЧХ фильтра Баттерворта – монотонно убывающая функция частоты.

В математическом приближении алгоритм работы фильтра сводится к свертке, которая делается последовательно для каждого семпла:

$$y(n) = b_0 * x(n) + b_1 * x(n - 1) + b_2 * x(n - 2) - a_1 * y(n - 1) - a_2 * y(n - 2) \quad (1),$$

$y(n)$  – это новое значение семпла, которое нужно рассчитать.  $x(n)$  – текущее значение семпла, соответственно  $y(n-1)$  и  $y(n-2)$  – предыдущие 2 рассчитанных семпла, а  $x(n-1)$  и  $x(n-2)$  – предыдущие входные значения семплов, а и  $b$  – некоторые коэффициенты.

Одним из самых простых эффектов является эхо, основанное на задержке. Эхо – это многократное повторение сигнала с экспоненциальным затуханием. То есть, текущее значение сигнала складывается как текущее новое значение плюс значение сигнала  $t$  времени назад,  $t$  – время задержки. Далее приведен общий вид формулы расчета:

$$y(n) = x(n) - y(n - T) \quad (2),$$

где  $x$  – входная последовательность семплов,  $y$  – результирующая,  $T$  – задержка в семплах.

Для регулирования уровня затухания, в формулу необходимо добавить долевые коэффициенты обработанного ( $P$ ) и необработанного сигнала ( $U$ ). Полученная формула представлена ниже:

$$y(n) = P * x(n) - U * y(n - T) \quad (3).$$

Далее необходимо добавить коэффициент снижения громкости по времени ( $F$ ):

$$y(n - T) = x(n) - F * y(n - T - 1) \quad (4).$$

Программная реализация будет представлять собой простейший циклический буфер.

**Список использованных источников:**

[1] Айфичер Э., Джервис Б. Цифровая обработка сигналов. Практический подход. М: Вильямс, 2016.

# КЛАССИФИКАЦИЯ ЭЛЕКТРОННОЙ ПОЧТЫ

Будник А.С.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Герман Ю.О. – канд. тех. наук

Рассматривается метод классификации электронной почты на основе выделения ключевых слов в письме и алгоритма кластеризации К-средних.

С недавнего времени электронная почта сменила устаревшую бумажную переписку по причине своей практичности. Актуальность создания методов классификации объясняется следующим: большое количество писем приходит на почтовые ящики ежедневно, сортировка этих писем поможет сэкономить время пользователя.

Для решения данной задачи можно использовать алгоритм нахождения ключевых слов в письме, а затем классифицировать письма на основе одного из методов кластеризации.

Любой текст можно охарактеризовать набором ключевых слов, которые этот текст представляют. Первоначально рассматривают только существительные. Подсчитываются частоты слов (сколько раз каждое слово вошло в текст). Поскольку слова входят в текст с разными окончаниями, то гласные выбрасываются и два слова считаются совпадающими, если они достаточно похожи друг на друга. При этом короткие слова должны быть похожими друг на друга в большей степени, чем длинные. Затем слова размещаются по убыванию частот, и берется примерно корень из общего числа слов с наибольшими значениями частот. При этом чем длиннее слова, тем меньше требуется допустимый процент совпадения [1]. Этот процент можно установить исходя из следующей эмпирически составленной таблицы 1.

Таблица 1 – Допустимое число неверных символов

Размер слова	Допустимое число неверных символов
До 5	0
5-6	1
7-8	2
9-10	3
Больше 10	4

На следующем этапе будет использоваться алгоритм К-средних.

Кластерный анализ – многомерная статистическая процедура, выполняющая сбор данных, содержащих информацию о выборке объектов, и затем упорядочивающая объекты в сравнительно однородные группы [2].

Метод К-средних – это метод кластерного анализа, цель которого – разделение  $m$  наблюдений на  $k$  кластеров, при этом каждое наблюдение относится к тому кластеру, к центру которого оно ближе всего. Результат разбиения на кластеры представлен на рисунке 1.

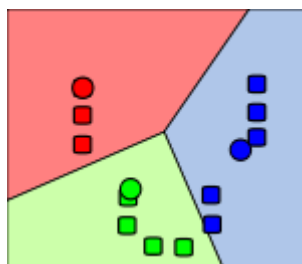


Рисунок 1 – Результат работы алгоритма К-средних

Таким образом, на основе тестовой выборки будут сформированы кластеры почтовых писем (например, деловые письма, поздравления, спам). Затем при получении нового письма будет определяться набор его ключевых слов, и письмо будет отнесено к тому или иному кластеру, тем самым будет выполнена процедура классификации писем.

#### Список использованных источников:

1. Герман, О.В. Искусственный интеллект: метод. пособие / О. В. Герман, Ю.О. Герман. – Минск : БНТУ, 2013. – 127с.
2. Мандель, И.Д. Кластерный анализ / И. Д. Мандель. – М.: Финансы и статистика, 1988. – 176 с.

# УДАЛЕННАЯ СИСТЕМА УПРАВЛЕНИЯ ДАННЫМИ

Герасимов Я.А.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Герман Ю.О. – канд. тех. наук

Рассматривается использование удаленной системы управления данными в качестве средства оптимизации процессов совместной работы по созданию и сопровождению программного обеспечения.

В современном мире трудно представить себе такую ситуацию, когда программное обеспечение будет разрабатывать один человек. В подавляющем большинстве случаев для разработки программных продуктов используется целая команда разноплановых специалистов: менеджеров, разработчиков, архитекторов решения, бизнес-аналитиков.

Большой объем информации и знаний, с которыми приходится работать командам, разрабатывающим программные продукты, требует целой стратегии по управлению этими знаниями. Большинство методологий, такие как Agile, Waterfall, Spiral Model, Iterative Model, требуют обязательного использования баз знаний в проекте, поскольку они существенно упрощают систематизацию знаний команды, необходимых для эффективной разработки программного обеспечения, для ввода новых участников и вовлечения всех членов команды в обсуждение и усовершенствование процесса разработки программного обеспечения.

Удаленная система управления данными (далее – система) предполагает постоянно функционирующее приложение с возможностью доступа через браузер, что позволяет оперативно вносить новые данные и получать изменения всем участникам разработки. Она является как базой знаний, так и мощным инструментом разработки программного обеспечения. Такая система будет полезна на всех стадиях разработки программ: техническом задании, эскизном проекте, техническом проекте, рабочем проекте и внедрении [1].

На стадии разработки технического задания система позволяет оценить целесообразность использования предыдущих решений, оперативно вносить изменения в техническое задание, дать представление всем участникам разработки о целях и средствах их достижения.

На стадии эскизного проекта система предоставляет возможность совместного доступа к диаграммам и моделям, совместную разработку структуры данных и общего алгоритма решения задачи.

На стадии технического проекта система позволяет совместную разработку структуры программы, разработку плана совместных мероприятий, определение формы представления входных и выходных данных.

На стадии рабочего проекта система становится наиболее полезной. В системе предоставляется возможность хранения и изменения плана тестирования, отдельных тестов, требований для разработчиков, дизайнов страниц, архитектуры приложения. Также хранятся и изменяются при необходимости со стороны заказчика бизнес-требования, предъявляемые к программному обеспечению. Сохраняются всевозможные записи встреч с заказчиком, записи обсуждений функциональных требований, записи презентаций.

На стадии внедрения необходимые документы, оформленные на предыдущих этапах в системе, передаются заказчику. Происходит сопровождение в соответствии с бизнес-процессами. При разногласиях данные из системы могут стать доказательством того, что все работает, как и было затребовано.

Таким образом, основные преимущества системы – в оперативном доступе к актуальным данным и их структурированность, а также безопасность и доступность. Различные типы и форматы данных могут быть иерархические, представлены в одном месте, что также немаловажное преимущество. Удаленная система управления данными на сегодняшний день востребована многими компаниями на всех стадиях разработки. А с текущей тенденцией подходов разработки программного обеспечения по методологии Agile подобные системы становятся жизненной необходимостью компаний по всему миру.

## **Список использованных источников:**

1. ГОСТ 19.102-77. Единая система программной документации. Стадии разработки – Введ. 01.01.1980, переиздан 04.03.2010. – М : Стандартинформ, 2010. – 2 с.

# ПРОГРАММНОЕ СРЕДСТВО ПРЕОБРАЗОВАНИЯ ДВУМЕРНЫХ ИЗОБРАЖЕНИЙ В 3D-МОДЕЛЬ

*Бобко А.В., Бахтизин В.В.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Бахтизин В.В. – к.т.н., профессор*

**Аннотация.** В работе рассматривается способ получения трехмерных моделей методом Structure from motion (структура на основе движения). Приводится пример соответствия между ключевыми точками на двух различных изображениях. Описывается последовательность действий преобразования двумерных изображений в 3D-модель.

**Ключевые слова.** Трехмерная реконструкция, 3D-модель, компьютерное зрение, метод Structure from motion, Python.

Трехмерная реконструкция – процесс получения формы и облика (3D-модели) реальных объектов. Существует несколько методик построения 3D-моделей объектов: с помощью различных датчиков, радиометрические методов, сенсоров и т.д.

Одним из способов получения трехмерных моделей является метод Structure from motion (структура на основе движения). Данный метод изучается в области компьютерного зрения и позволяет строить трехмерные структуры на основе существующего набора двумерных изображений (фотографий) объекта [1]. Метод состоит из нескольких последовательных этапов. На первом этапе на фотографиях осуществляется поиск так называемых ключевых точек – точек изображения, которые мы определяем, как отличительные точки во входном изображении, инвариантные к вращению, масштабу и искажению. Следующим шагом мы ищем соответствия между ключевыми точками различных изображений. Затем осуществляется фильтрация ложных соответствий между этими точками. Пример соответствия ключевых точек показан на рисунке 1.

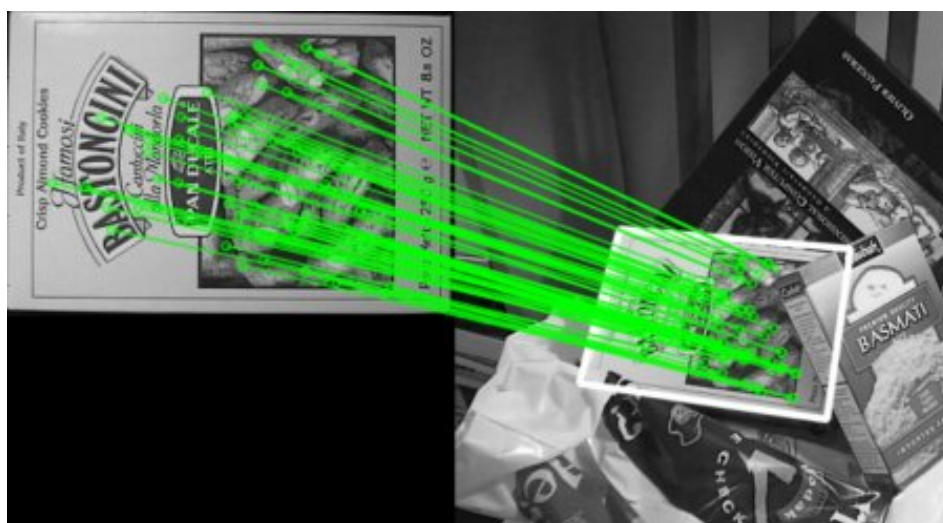


Рисунок 1 – Пример соответствия между ключевыми точками на двух различных изображениях

После этого решается система уравнений и находится трехмерная структура вместе с положениями камер, после этого мы получаем облако точек. На последнем этапе происходит преобразование из облака точек в полноценную 3D-модель объекта. Существует множество алгоритмов компьютерного зрения, с помощью которых можно реализовать каждый шаг данного метода [2].

Целью данной работы явилась разработка программного средства для преобразования двумерных изображений в 3D-модель с использованием языка программирования Python. В рамках разработки программного средства реализован метод генерации трехмерных моделей объектов с помощью метода Structure from motion.

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

1. *Evaluating the Performance of Structure from Motion Pipelines / S. Bianco, G. Ciocca, D. Marelli // Journal of Imaging, 2018. – P. 1-18.*
2. *Computer Vision: Algorithms and Applications / R. Szeliski // Springer, 2010. – P. 205-266, 343-380.*

# ПРОГРАММНОЕ СРЕДСТВО ИМИТАЦИИ МУЗЫКАЛЬНЫХ ИНСТРУМЕНТОВ НА БАЗЕ ПЛАТФОРМЫ .NET FRAMEWORK НА ЯЗЫКЕ C#

Южик М. Ю.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Медведев С.А. – канд. физ.-мат. наук

Программное средство, с помощью которого можно, используя компьютерную клавиатуру, имитировать игру на синтезаторе.

Программное средство имитации музыкальных инструментов позволяет при помощи компьютерной клавиатуры воспроизводить звуки, схожие со звучанием музыкальных инструментов.

Диапазон большинства музыкальных инструментов превышает 3 октавы [1]. Например, диапазон звучания синтезатора включает 7 октав. Однако размеры монитора компьютера или ноутбука не позволяют графически отобразить такое количество октав. Поэтому важно, чтобы программное средство имело возможность переключения между октавами.

При создании программы, с помощью которой можно имитировать игру на синтезаторе, необходимо учитывать расположение клавиш на музыкальном инструменте и правильно соотносить его с раскладкой клавиатуры компьютера. Так как указанная программа может быть использована как в развлекательно-игровых целях, так и для обучения игре на музыкальном инструменте, должно быть предусмотрено правильное расположение диэзно-бемольных (черных) клавиш фортепиано относительно белых. Схематическое представление соотношения расположения клавиш фортепиано [2] и раскладки клавиатуры компьютера приведено на рисунке 1.

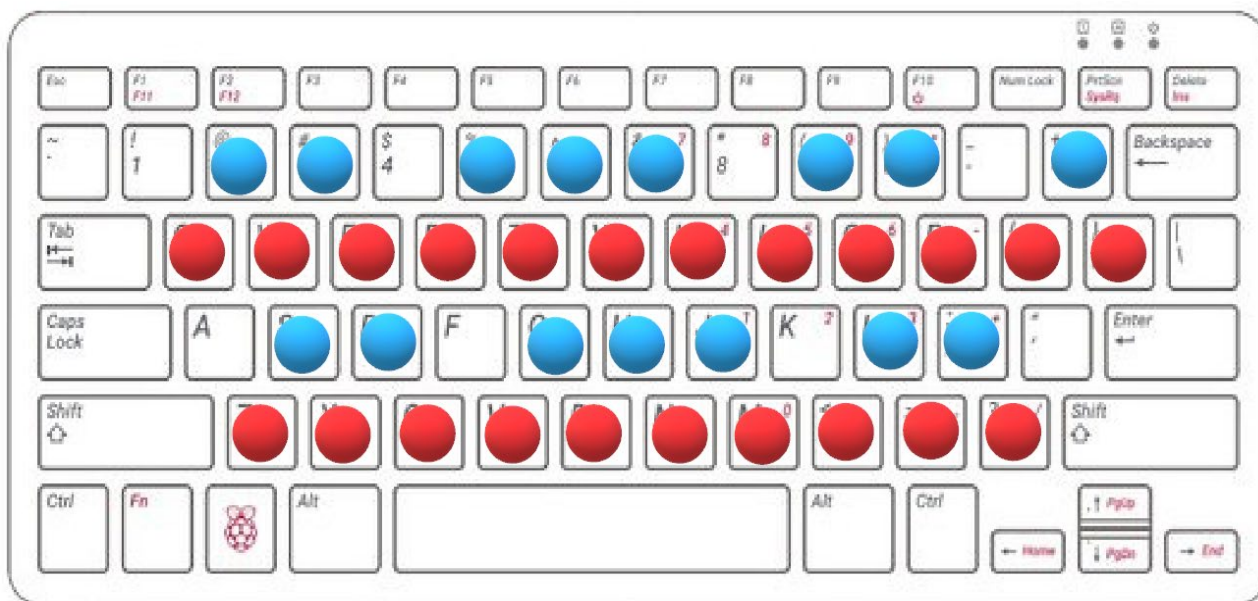


Рисунок 1 – Схема соответствия клавиш фортепиано и клавиш компьютера

При игре на музыкальных инструментах громкость звука варьируется с помощью механических воздействий на инструмент: силы нажатия клавиши, мощности потока воздуха, степени резкости удара и т.д. [3]. При этом для программного средства нет такой возможности изменения громкости звучания с помощью регулирования силы механических воздействий на клавиатуру. Поэтому в программном средстве должна быть предусмотрена опция изменения громкости при помощи клавиатуры

#### Список использованных источников:

1. Звуковой спектр, голос и музыкальные инструменты – Мюзик хоум [Электронный ресурс]. – Электронные данные. – Режим доступа: [http://music-home.ru/help/article/zvukovoy\\_spektr\\_golos\\_i\\_muzykalnye\\_instrumenty](http://music-home.ru/help/article/zvukovoy_spektr_golos_i_muzykalnye_instrumenty) – 2013.
2. Расположение нот на клавиатуре фортепиано – Музыкальный теоретик [Электронный ресурс]. – Электронные данные. – Режим доступа: <https://muz-teoretik.ru/raspolozhenie-not-na-klaviature-piano/> – 2005.
3. Мастеринг и громкость звука – Музыкальный блог [Электронный ресурс]. – Электронные данные. – Режим доступа: <http://danalex.ru/mastering/> – 20016.

# ПРИМЕНЕНИЕ ГЕНЕТИЧЕСКИХ АЛГОРИТМОВ ДЛЯ КРИПТОАНАЛИЗА ПОДСТАНОВОЧНЫХ ШИФРОВ

Фролов А.О.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Прохорчик Р.В. – ст. преп., м. т. н.

В данной работе рассматривается иной подход к криптоанализу подстановочных шифров, основанный на применении генетических алгоритмов. Показывается, что такие алгоритмы могут быть использованы для обнаружения ключа простых шифров подстановок. В качестве существенного фактора целевой функции генетического алгоритма используется частотный анализ.

В настоящее время безопасность является одной из важнейших проблем в информатике, криптография широко используется для ее реализации. Задачу, противоположную криптографии, решает криптоанализ – процесс, в котором пытаются нарушить безопасность. В свою очередь, сложность этого процесса рассматривается как мера безопасности. Поскольку криптографические алгоритмы открыты для всех, вся мощность алгоритма заключается в сложности ключа, сила которого в общем случае определяется его длиной.

Данная работа посвящена криптоанализу подстановочных шифров, которые являются одними из древнейших. Шифры такого рода явным или неявным образом используют таблицы подстановок. Рассмотрим в качестве шифра подстановки достаточно популярный шифр Виженера и покажем, насколько применим генетический алгоритм для его криптоанализа.

Генетические алгоритмы – это класс эвристических алгоритмов, которые пытаются решить задачу путем моделирования упрощенной версии генетических процессов. Такого рода алгоритмы основаны на концепциях генетики и эволюции. Генетический алгоритм функционирует на основе популяции выбранных организмов, в геномном материале которых (генотип) закодирован ответ на поставленный вопрос (фенотип). Для простых задач пространства фенотипов и генотипов одинаковы, однако в большинстве случаев эти пространства различны. Генетический алгоритм начинается с формирования изначального популяционного материала на основании произвольной выборки. В дальнейшем на основании аналитики дается оценка популяции по уровню приспособленности к среде обитания или заданным факторам. Затем самые приспособленные особи скрещиваются. Таким образом, популяционные особи дают потомство, которое имеет высокую вероятность сохранить необходимый генотип в своем и следующих поколениях. Потомство изначальной популяции создает следующую по хронологии популяцию, причем у большинства особей должен сохраняться на протяжении всей жизни выбранный генотип, а минимальное число особей подвержены генной мутации. Отметим, что каждое множество особей популяции в определенный промежуток времени – это поколение. Смена поколений образует популяционную эволюцию. При этом каждый этап эволюционного процесса может иметь разную длительность, что обусловлено различной скоростью поиска правильного решения, ограничением на количество возможных поколений, а также другими факторами.

Стоит отметить, что генетические алгоритмы не всегда дают точный ответ, а скорее дают решение, близкое к правильному. Несмотря на неточность, генетические алгоритмы показывают хорошие результаты при решении оптимизационных задач и поиске ответов на такого рода вопросы. По той причине, что целевая установка взлома шифра – это выявление единственно верного секретного ключа в ряду потенциально возможных ключей, то задача криптографического анализа может быть представлена в виде оптимизационной задачи. Таким образом, в поиске решения такой задачи может быть применен генетический алгоритм.

В первую очередь, при создании генетического алгоритма важно выбрать оптимальный, эффективный способ кодирования и описать основные операторы алгоритма – селекция, кроссинговер и мутация. На качественную составляющую алгоритма ключевое значение оказывает популяционная численность, количество поколений, значения уровня вероятности для базовых операторов.

В случае расшифровки зашифрованного текста каждое решение хранит ключ  $k$  длины  $n$ , а операторы мутации и кроссинговера изменяют  $k$ . Для реализации расшифровки текста была выбрана малая вероятность мутации (0.25), а уровень вероятности наследования генотипа от одного поколения к другому была, в свою очередь, выбрана достаточно большой (0.85). Оператор кроссинговера, используемый в данной работе, представляет собой двухточечный кроссинговер. Для выбора наиболее приспособленных особей для скрещивания была выбрана турнирная селекция.

Одной из главных задач при использовании генетических алгоритмов для расшифрования является выбор целевой функции. Реализовать хорошую целевую функцию довольно нетривиально. Языковая статистика и атаки по словарю должны противостоять этой проблеме. Весьма интересный подход к вычислению целевой функции описан в работе Э. Кларка [1]. Суть этого подхода заключается в сравнении  $n$ -граммной статистики расшифрованного сообщения с данными языка, которые считаются

известными. Уравнение, являющееся общей формулой, используемой для определения пригодности предполагаемого ключа  $k$ , имеет следующий вид:

$$f(k) = \alpha \cdot \sum_{i \in A} |D_i^u(k) - S_i^u| + \beta \cdot \sum_{i,j \in A} |D_{i,j}^b(k) - S_{i,j}^b| + \gamma \cdot \sum_{i,j,l \in A} |D_{i,j,l}^t(k) - S_{i,j,l}^t|, \quad (1)$$

где  $A$  – алфавит языка,  $D$  – частота появления  $n$ -граммы в анализируемом текстовом фрагменте, который получен из начального зашифрованного текста путем его расшифровки с применением предполагаемого ключа, обозначенного буквой  $k$ ,  $S$  – заданный средний уровень частоты повторения  $n$ -граммы в различных текстовых источниках, а индексы  $u$ ,  $b$  и  $t$  обозначают статистику униграмм, биграмм и триграмм соответственно. Значения  $\alpha$ ,  $\beta$  и  $\gamma$  позволяют присваивать различные веса каждому из трех типов  $n$ -грамм. Целевая функция опирается на статистическую характеристику языка, чтобы определить пригодность ключа. Таким образом, выбор меры пригодности полностью зависит от языковых характеристик и, следовательно, эти характеристики должны быть известны, как и сам используемый язык.

Разработанный на основе вышеизложенного алгоритм, позволяющий проводить криптографический анализ шифра Виженера, работает на заранее заданной длине ключа шифрования и находит с определенной точностью сам секретный ключ. Выбранная функция пригодности является основным фактором алгоритма. Текст, получаемый в результате работы алгоритма, не сильно отличается от исходного текста. В большинстве случаев отдельные символы легко восстанавливаются с помощью контекста. Результаты проведенных опытов дают возможность сказать, что в случае малой численности популяции результаты получаются не оптимальными, что продемонстрировано на рисунке 1. Это связано с тем, что генетический алгоритм достаточно быстро определяет место локального экстремума функции. Очевидно, что число поколений сильно зависит от длины ключа и, следовательно, с увеличением длины ключа нужно увеличивать число последовательно идущих друг за другом поколений. Соблюдение обозначенного условия дает возможность получить наиболее точное решение. Изменения основных генетических операторов алгоритма в допустимых пределах не дали улучшения результатов.

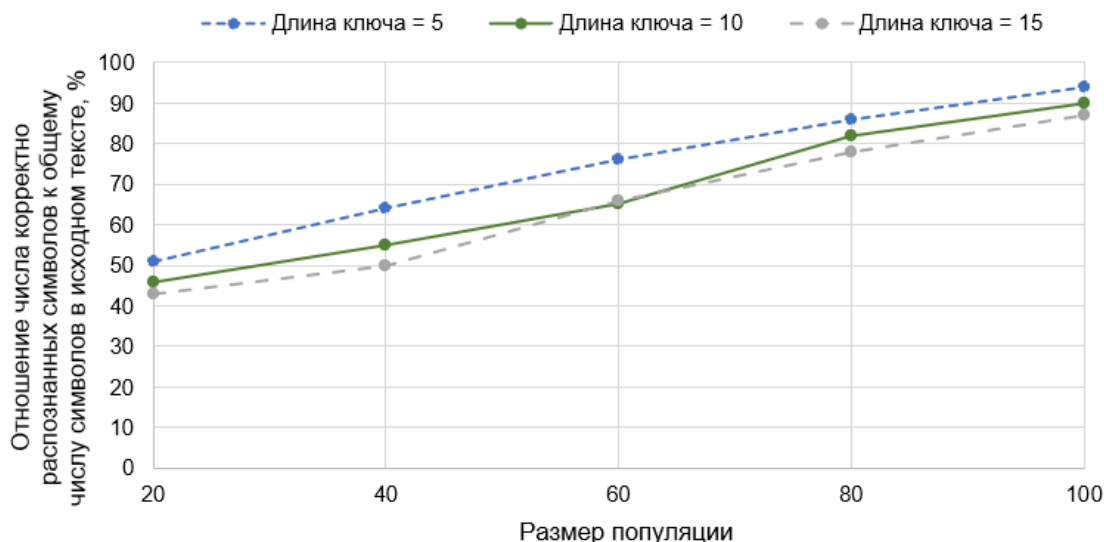


Рисунок 1 – Процентное соотношение корректных символов при различных размерах популяции

В данной работе была успешно реализована атака генетического алгоритма на простой криптографический шифр подстановки. Несмотря на то, что не всегда удается полностью автоматизировать процесс решения задачи криптоанализа, алгоритм в значительной степени помогает расшифровать зашифрованный текст. В ходе работы были протестированы различные параметры, такие как численность популяции, число поколений, а также вероятностные характеристики основных операторов. Использование генетического алгоритма для криптоанализа простого шифра подстановки оказалось эффективным методом криптоанализа, основанным на аспекте сравнения частот встречаемости букв. Так как подход, основанный на применении генетического алгоритма, оказался успешным для простого шифра подстановки, то имеет смысл применить тот же подход к более современным и сложным шифрам.

**Список использованных источников:**

1. Clark, A. *Modern optimisation algorithms for cryptanalysis*. In *Proceedings of the 1994 Second Australian and New Zealand Conference on Intelligent Information Systems*, November 29 – December 2 1994, p. 258-262.



# ОНЛАЙН ПЛАТФОРМА ПОКУПКИ И ПРОДАЖИ ТЕХНИКИ

Евдаков А.А.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Медведев С.А. – канд. тех. наук

Платформа позволяющая покупателям приобретать технику по оптовой цене, разработанная с использованием технологий ASP.Net.Core, Docker, Postgresql .

Возможность покупки электроники по оптовой цене, дает возможность обновлять её чаще. В текущих реалиях, новый софт становится все тяжеловеснее, а старый часто перестает поддерживаться, что заставляет многих пользователей переходить на более новые модели. Такую платформу смогут использовать как для обновления всего офиса, так и для обновления своего рабочего места.

Для обеспечения данной модели, был выбран подход при котором на платформе встречаются поставщики и клиенты. А сама процедура оплаты проходит в виде массовых сборов. Таким образом поставщик может объявить о продаже некоторого количества единиц определенной техники с указанием сроков и цены, если данная сумма собирается в течении указанного срока, то сбор считается успешным, после чего организуется доставка товара. В случае, когда срок истекает, а сумма не набрана, всем пользователям, которые участвовали в сборе, возвращаются деньги. Подход который использует данная платформа включает себя так называемый принцип краудфандинга, который заключается в коллективном сборе средств на достижение общей цели.

Существует множество вер ресурсов по продажам. Среди аналогов можно выделить такие платформы как Onliner и AliExpress, так как они больше всего соответствуют предложенной модели.

На платформе Onliner представлена агрегирующая модель, т.е. вы можете найти товар и купить его у одного из представленных поставщиков, но данный товар будет приобретен по розничной цене. В свою очередь AliExpress представляет модель при которой себестоимость товара довольно низкая, но при этом поставщиком данной продукции являются Китай. Если сравнивать данные платформы с разработанной то можно выделить следующее:

1. Возможность выбора продукции определенного поставщика
2. Себестоимость снижена из-за оптовой поставки
3. Процесс оплаты не гарантирует приобретение продукции, но в данном случае ваши средства будут возвращены
4. Сегмент рынка на который направлен данная платформа шире, чем у аналогов, так как данный подход выгоден лицам, заинтересованным в оптовой покупке техники

Таким образом данная платформа совмещает в себе плюсы перечисленных ранее платформ. Что делает ее конкурентоспособной на рынке. Кроме того, для разработанной платформы важной частью является безопасность, которая включает в себя защиту данных пользователей и безопасность перевода средств. Для достижения данной цели были применены такие подходы как авторизация на основе JWT[1] технологии, а именно JwtBearer[2] от Microsoft. Данная технология генерирует уникальный токен авторизации клиента исходя из его данных, который истекает через определенный срок, а для безопасности оплаты была использована такая технология как Stripe[3].

## Список использованных источников:

1. JSON Web Token (JWT) is an open industry standard RFC 7519 [Electronic Resource] / Auth0 - Model of access: <https://jwt.io/>. Date of access 02.04.2021
2. Microsoft.AspNetCore.Authentication.JwtBearer [Electronic Resource] / Microsoft - Model to access: <https://docs.microsoft.com/en-us/dotnet/api/microsoft.aspnetcore.authentication.jwtbearer?view=aspnetcore-5.0>. Date of access 02.04.2021
3. Stripe. Payments infrastructure for the internet [Electronic Resource] - Model of Access: <https://stripe.com/>. Date of access: 02.04.2021

## ПРОГРАММНОЕ СРЕДСТВО, РЕАЛИЗУЮЩЕЕ ОБРАБОТКУ ЗАКАЗА БАНКОВСКИХ КАРТ С ИСПОЛЬЗОВАНИЕМ ЯЗЫКА DELUGE SCRIPT НА БАЗЕ ПЛАТФОРМЫ ZOHO CREATOR

Шабалтас А.Ю.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Медведев С.А. – канд. тех. наук

**Аннотация.** Программное средство предназначено для автоматизации процесса обработки заказов банковских подарочных карт и обеспечивает повышение эффективности работы всех отделов предприятия, связанных с этим процессом. В качестве платформы разработки была выбрана платформа разработки с низким кодом Zoho Creator, ввиду комплексного потока задач процесса обработки заказов и наличия у платформы возможности интеграций с множеством внутренних и внешних сервисов. Разработка на базе платформы Zoho Creator выполнена на языке сценариев.

Банковские подарочные карточки — это неименные карты мгновенного выпуска, по которым гарантией безопасности платежей выступает пин-код [1]. Для корректного и безопасного обеспечения выполнения управления процессом заказа банковских карт и их обработки было разработано несколько инструментов, но их использование и поддержка стали затратны, т.к. популярность банковских подарочных карт и их спрос среди корпоративных клиентов с целью выдачи премий и других денежных вознаграждений в виде этих карточек значительно выросли и продолжают увеличиваться. Необходимость соблюдения PCI принципов [2] еще больше затрудняет процесс обработки заказа.

Предлагаемое программное средство разрабатывается с целью улучшения процесса управления заказами банковских подарочных карточек, его частичной автоматизации и в дальнейшем полного избавления от ручных операций со стороны менеджеров отдела компании. В настоящее время рабочий процесс заказа банковских подарочных карточек выполняется вручную с использованием трех различных инструментов: Tomcat (инструмент для отслеживания задач), Invoice Tool (инструмент создания заказов) и Ordering Tool (внешний инструмент по обработке заказов). Первые два инструмента уже устарели и трудоемки в обслуживании. Текущий процесс представлен на рисунке 1.

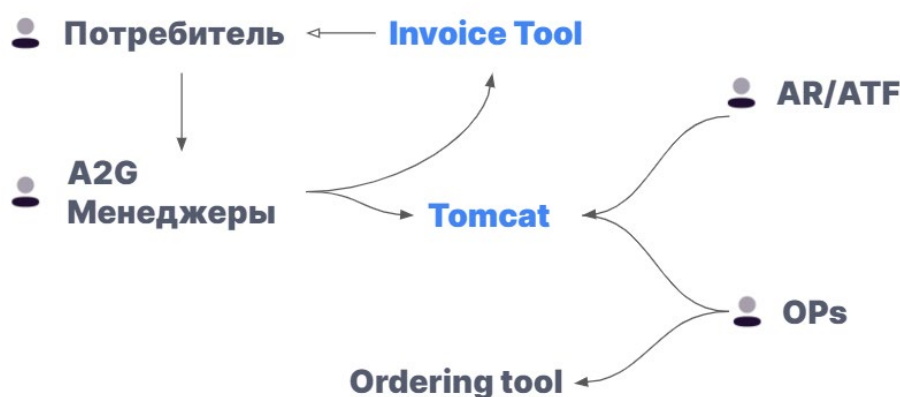


Рисунок 1 – Текущий процесс управления заказами банковских подарочных карточек

В данном процессе принимают участие менеджеры из 7 отделов, которым необходимо взаимодействовать с потребителями напрямую и обращаться к нескольким инструментам. Разрабатываемое программное средство призвано заменить два устаревших инструмента, что позволит сделать процесс безупречным от начала до конца и избежать многих проблем по причине человеческого фактора.

Главное преимущество данного приложения - это создание единой системы, которая позволит:

- создавать заказы;
- отправлять заказ клиенту файлом;
- согласовать и отслеживать заказы;
- отслеживать задачи по заказам;
- обеспечить безопасную доставку.

Кроме того, приложение позволит расширить процесс и сделать его полностью автоматическим за счет интеграции с Zoho Creator API [3] и исключения ручного ввода данных заказов при его создании.

Разработанная система управления бизнесом представлена на рисунке 2.

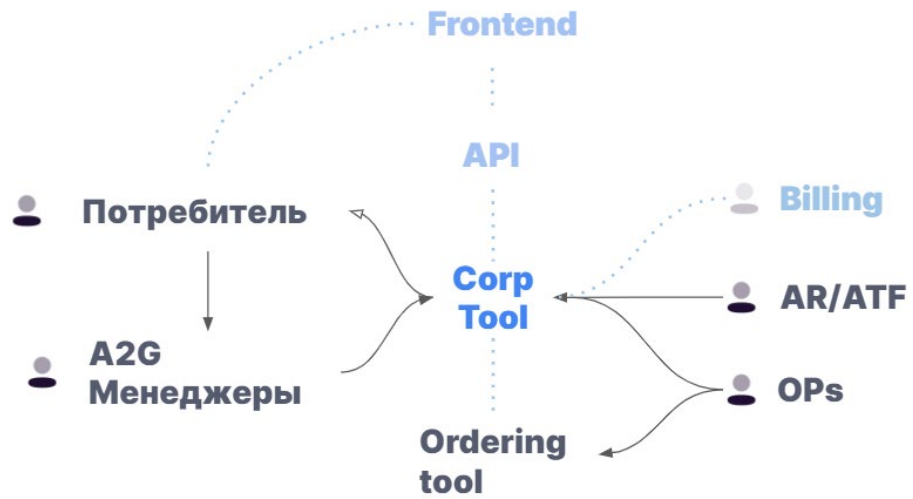


Рисунок 2 – Обновлённый процесс управления заказами банковских подарочных карточек

**Список использованных источников:**

1. Банковские подарочные карточки - Инфобанк[Электронный ресурс]. - Электронные данные. - Режим доступа: <https://infobank.by/> - 2021
2. Принципы PCI - Федерал Трейд Коммишион[Электронный ресурс]. - Электронные данные. - Режим доступа: <https://www.ftc.gov/tips-advice/business-center/guidance/protecting-personal-information-guide-business> - 2021
3. Zoho Creator API - Зохо[Электронный ресурс]. - Электронные данные. - Режим доступа: <https://www.zoho.com/creator/help/api/v2/> - 2021

# СИСТЕМА АВТОМАТИЧЕСКОГО РЕЗЕРВНОГО КОПИРОВАНИЯ И ОБРАБОТКИ ДАННЫХ НА ЯЗЫКАХ ПРОГРАММИРОВАНИЯ JAVA, JS

*Аль-Саррих З.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Видничук В.Н. – м.т.н. ассистент кафедры ПОИТ*

Облачные технологии фундаментально меняют ожидания в отношении того, как должны распределяться, управляться и потребляться ресурсы вычислений, хранения и сети в глобальном масштабе. Мощные вычислительные системы и системы хранения данных, высокий уровень безопасности, легкая доступность и настраиваемость, надежная масштабируемость и функциональная совместимость, экономичные и гибкие «облачные» вычисления являются самыми востребованными в современном быстрорастущем деловом мире. Клиент, организация или предприятие, внедряющее новую облачную среду, может выбрать подходящую инфраструктуру, платформу, программное обеспечение и сетевые ресурсы для каждого бизнеса, каждый из которых обладает эксклюзивными возможностями.

Целью проекта является повысить эффективность автоматического резервного копирования и обработки данных путём разработки онлайн-системы облачного хранения на языках программирования JavaScript и Java.

Объектом исследования являются процессы резервного копирования и обработки данных, реализуемые в системе облачного хранения данных.

Для разработки программного средства нам необходимо обратить внимание на архитектуру проектируемого программного обеспечения, которую мы проиллюстрируем с помощью диаграммы развертывания. Данная диаграмма используется для представления общей конфигурации и топологии распределенной программной системы и показывает распределение компонентов по отдельным узлам системы. Он также показывает наличие физических путей передачи информации между аппаратными устройствами, участвующими в реализации системы. Схема развертывания используется для визуализации элементов и компонентов программы, которые существуют только во время ее выполнения (выполнения). Он представляет только те компоненты, которые являются исполняемыми файлами или динамическими библиотеками. Компоненты, которые не используются во время выполнения, не отображаются на диаграмме развертывания.

При разработке архитектуры системы использовались услуги Amazon Web Services (AWS) – коммерческого общедоступного облака, поддерживаемого и разрабатываемого компанией Amazon с 2006 года. AWS предоставляет абонентам как услуги по моделированию инфраструктуры (виртуальные серверы, ресурсы хранения), так и услуги на уровне платформы (облачные базы данных, программное обеспечение для подключения к облаку, бессерверные облачные вычисления, средства разработки).

На рисунке 1 предоставлена диаграмма развертывания, показывающая узлы и связи между компонентами архитектуры.

Ключевая инфраструктурная услуга AWS – служба аренды виртуальных серверов EC2. Для реализации программно-аппаратной части сервиса используется два сервера: «DataBase Server» и «Backend Server». На виртуальные машины данных серверов была поставлена ОС CentOS Linux 7 x86\_64. Для реализации пользовательской части ПК использованы 2 идентичных по наполнению сервера «Frontend USA Server» и «Frontend Russia Server». Данное решение используется для балансировки нагрузки между сетевыми устройствами (серверами) с целью сокращения времени обслуживания запросов пользователей. За балансировку нагрузки отвечает средство «Balance Loader».

Запросы от устройств пользователя, гостя или менеджера поступают в первую очередь на «Balance Loader» по протоколу https (протокол безопасной передачи данных, поддерживает технологию шифрования TLS/SSL). Балансировщик нагрузки направляет пользователя на один из двух серверов «Frontend USA Server» или «Frontend Russia Server» в зависимости от адреса, с которого поступил запрос. Это позволит улучшить работу пользователя с пользовательской частью системы.

Балансировщик нагрузки работает с серверами, отвечающими за отображение пользовательской части ПК, и роутером по протоколу TCP (один из основных протоколов передачи данных интернета). Соответственно таблице маршрутизации роутера, которая описывает соответствие между адресами назначения и интерфейсами, пакеты данных поступают на «Backend Server».

«Backend Server» работает с «DataBase Server» по PostgreSQL client-server протоколу и с помощью стандарта JDBC, который позволяет приложению взаимодействовать с СУБД. Стандарт JDBC основан на «концепции так называемых драйверов, позволяющих получать соединение с базой данных по специально описанному URL. Драйверы могут загружаться динамически (во время работы программы). Загрузившись, драйвер сам регистрирует себя и вызывается автоматически, когда программа требует URL, содержащий протокол, за который драйвер отвечает» [1].

Частью архитектуры является NAT, механизм в сетях TCP/IP, позволяющий преобразовывать IP-адреса транзитных пакетов. Этот механизм предотвращает внешний доступ к внутренним хостам, одновременно разрешая внутренний доступ к внешним. Трансляция создается при инициации соединения изнутри сети. Ответные пакеты, поступающие извне, соответствуют созданной трансляции и поэтому разрешены к прохождению. Если для пакетов, поступающих извне, соответствующей трансляции не существует (а она может быть созданной при инициации соединения или статической), они не пропускаются.

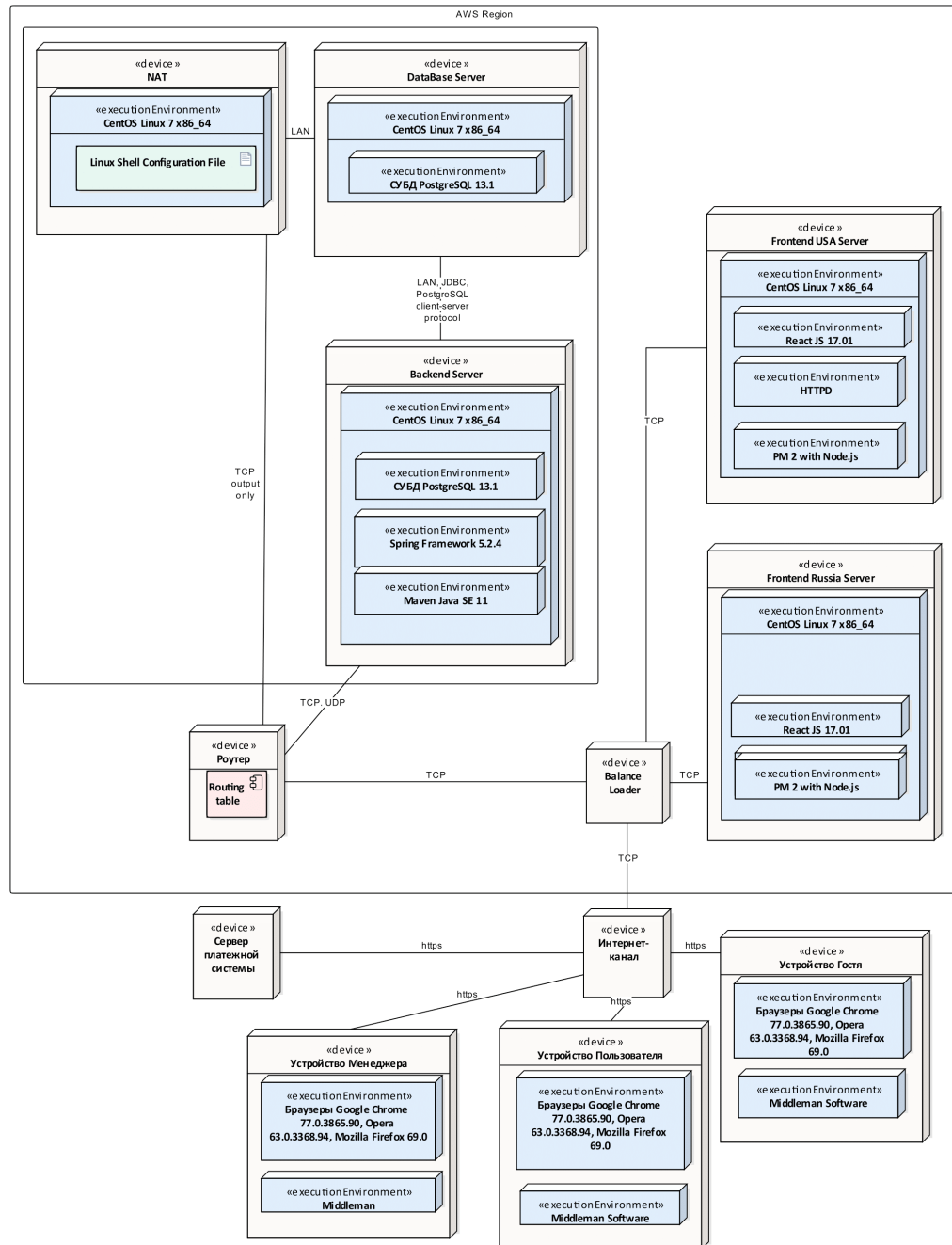


Рисунок 1 – Диаграмма развертывания

В итоге использование в системе двухзвенной клиент-серверной архитектуры с кластером с балансировкой нагрузки позволяет равномерно распределять запросы клиентов между доступными серверами и обеспечивает стабильную работу системы резервного копирования и обработки данных.

**Список использованных источников:**

1. Java Database Connectivity [Электронный ресурс]. – Режим доступа: [https://ru.wikipedia.org/wiki/Java\\_Database\\_Connectivity](https://ru.wikipedia.org/wiki/Java_Database_Connectivity). – Дата доступа: 03.04.2021.

# КЛИЕНТ-СЕРВЕРНОЕ ВЕБ-ПРИЛОЖЕНИЕ ПО АРЕНДЕ ПЛОЩАДЕЙ ЛЮБОГО ХАРАКТЕРА “PLOT RENT” НА БАЗЕ ТЕХНОЛОГИИ REACT

Кимаев А.А.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Мельникова Е.В. – ассистент

В докладе рассматриваются проблемы, связанные с арендой участков, и пути их решения.

В настоящее время для аренды участков можно воспользоваться некоторыми сайтами, такими как «Регистр стоимости земель, земельных участков государственного земельного кадастра» [1], «GoHome» [2] и другими. Но среди них достаточно сложно найти ресурс, на котором можно было бы арендовать любые виды участков от земельных и до офисных, с интегрированными системами оплаты и документооборота, связанными с арендой, на котором можно легко сдать в аренду или арендовать участок.

В разработанное программное средство были добавлены вышеупомянутые возможности, что делает его наиболее удобным для аренды участков.

Процесс работы с программным средством заключается в следующем: регистрация аккаунта с указанием платёжного аккаунта PayPal, авторизация, и далее доступны функции аренды участков, сдачи участка в аренду, чата с пользователями, оставление отзывов об участках.

Для аренды участка необходимо лишь нажать на участок на карте, затем кнопку «арендовать» и далее следовать подсказкам по аренде. В результате получится договор подтверждающий аренду, подписанный электронной цифровой подписью с двух сторон, и участок будет считаться арендованным.

В течение всей аренды, средства будут автоматически списываться со счета арендатора, и перечисляться на счёт арендодателя. В случае если оплата по аренде не прошла успешно, об том будут уведомлены обе стороны контракта и тогда арендатор должен будет выполнить платёж вручную. Любой из участников контракта может в любой момент посмотреть историю транзакций.

Также после заключения контракта об аренде доступна функция комментирования арендованного участка. Помимо этого в приложении есть функция чата, которую можно использовать, например, для урегулирования вопросов по контракту.

Главная страница приложения с участками на карте приведена на рисунке 1.

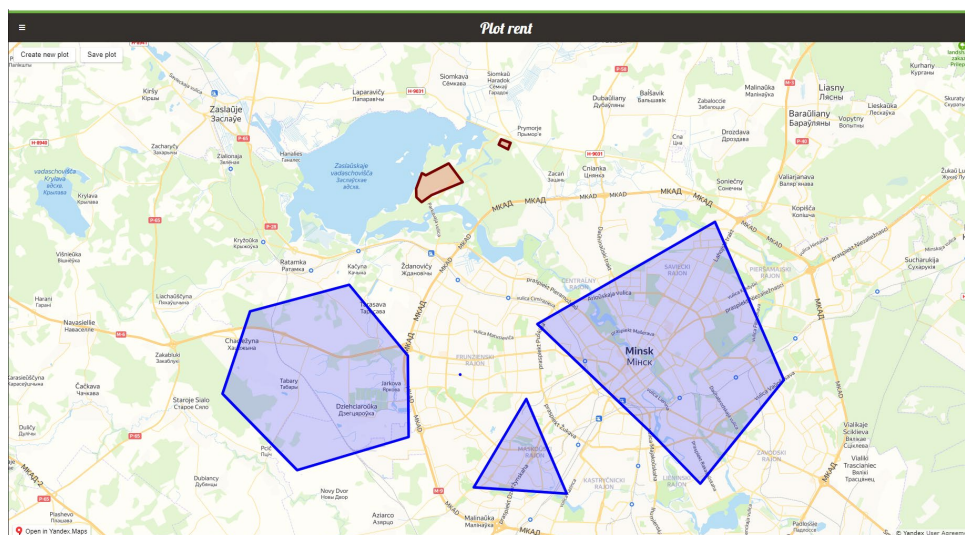


Рисунок 1 – Главный экран приложения с картой и участками

## Список использованных источников:

1. Регистр стоимости земель, земельных участков государственного земельного кадастра [Электронный ресурс]. – Электронные данные. – Режим доступа: <http://vi.nca.by/>
2. GoHome [Электронный ресурс]. – Электронные данные. – Режим доступа: <https://gohome.by/>

# МОБИЛЬНОЕ ПРИЛОЖЕНИЕ ДЛЯ ИЗУЧЕНИЯ ПРАВИЛ ДОРОЖНОГО ДВИЖЕНИЯ

*Борисенко А.Д.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Мельникова Е.В.*

В результате возрастающей популярности автомобилей как вида транспорта, потребность в получении водительского удостоверения становится все более и более актуальной. По данным ГАИ города Минска, большая часть курсантов автошкол проваливаются на первом этапе экзамена (сдача теории). В данной работе мною была предпринята попытка создания мобильного приложения, которое будет качественно отличаться от своих аналогов и помогать в освоении правил дорожного движения и подготовке к теоретическому экзамену на получения водительского удостоверения.

Хотелось бы начать с обоснования выбора темы, ее актуальности и перспектив разработанного продукта. Немного статистики:

- автомобильный транспорт является самым популярным видом средств передвижения;
- в соседней России автомобилисты составляют 47% населения и эти проценты продолжают расти (на данный момент на 2-3% в год) (по данным ВЦИОМ);
- по данным Международной ассоциации автопроизводителей, уже в 2015 году в мире эксплуатировалось более 1.2 миллиарда машин, при этом 95% из них - легковые;
- по данным ГАИ Минска, более чем 3/4 курсантов автошкол проваливаются на первом этапе экзамена (а это сдача теории);
- по мнению все тех же инспекторов ГАИ причиной тому является недобросовестное отношение многих автошкол к процессу обучения;
- ГАИ считают, что с приходом новых технологий, в частности – мобильного приложения, ситуация может улучшиться.

Основываясь даже на этих немногочисленных фактах (популярность автомобилей, непрекращающийся рост этой популярности, проблемы со сдачей экзамена на права, мнение ГАИ), можно сделать вывод, что при должном подходе приложение направленное на теоретическую подготовку к экзамену по правилам дорожного движения может значительно облегчить подготовку как для курсантов, так, возможно, и для самих автошкол.

Были выдвинуты следующие требования к функционалу нашего программного средства:

1. тренировка по главам ПДД;
2. тренировка по тематическим билетам;
3. тренировка по случайному билету;
4. контроль по тематическим билетам;
5. контроль по случайному билету x10;
6. контроль по случайному билету;
7. персональная тренировка (для авторизованных пользователей);
8. рейтинг пользователя в системе (для авторизованных пользователей);
9. история контролей (для авторизованных пользователей).

Все перечисленные функции были успешно реализованы в процессе разработки программного средства. Кроме того, важной отличительной особенностью нашего продукта является то, что технически продукт реализован таким образом, что отсутствует жесткая привязка к географическому положению, это значит, что тесты могут быть составлены для любой страны. Немаловажно, что в контексте приложений-аналогов, программное средство является уникальным ввиду того, что статистика пользователя, история его контролей и трудные для него вопросы хранятся в удаленной базе данных. Это говорит об отсутствии привязки к конкретному устройству.

Результатом работы является полностью готовое к использованию и внедрению программное средство «Правила дорожного движения». Приложение разработано на платформе iOS и обладает всеми необходимыми функциями для изучения правил дорожного движения, которые могут удовлетворить потребности как курсантов, так, при должной организации и подходе, автошколы и ГАИ.

## Программное средство для изучения правил дорожного движения на языке программирования Swift: архитектура приложения, слои доступа к данным и сервисный

*Митрофанов А. Б., студент гр.751003*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Мельникова Е. В.*

**Аннотация.** В наше время вождение автомобиля становится неотъемлемой частью человеческой жизнедеятельности, что делает вопрос получения водительских прав как никогда актуальным. На сегодняшний день как в нашей стране, так и в большинстве стран мира, для тренировок и контроля знаний Правил Дорожного Движения используются программные средства, созданные с помощью информационных технологий. Постоянное совершенствование настолько часто используемых инструментов, их обновление с применением современных подходов и технологий являются необходимыми процессами на пути к улучшению качества получения и контроля знаний. Цель работы - разработать часть мобильного приложения для тренировки и сдачи теоретического экзамена для получения водительских прав.

После постановки задачи были сформулированы требования к программному средству:

Платформы:

- iOS;
- Web.

Функционал:

- Тренировка по главам ПДД;
- Тренировка по тематическим билетам;
- Тренировка по случайному билету;
- Контроль по тематическим билетам;
- Контроль по случайному билету в закрытом режиме;
- Контроль по случайному билету в открытом режиме;
- Персональная тренировка (для авторизованных пользователей);
- Персональная статистика прохождения контролей (для авторизованных пользователей);
- История контролей (для авторизованных пользователей).

В процессе разработки мобильного приложения на платформе iOS было решено использовать современные инструменты и подходы для выбора и реализации архитектуры ПС, слоев доступа к данным и сервисного. В частности был применен популярный на сегодняшний день подход реактивного программирования с использованием современного фреймворка Combine от компании Apple в связке с часто применяемой, особенно в мобильных приложениях среднего объема, архитектурой Model View View Model. При реализации слоев доступа к данным и сервисного были применены подходящие шаблоны объектно-ориентированного проектирования и соблюдены принципы SOLID, что при дальнейшем развитии проекта позволит не только легко добавлять функциональность в уже работающее приложение, но и использовать готовые компоненты, написанные в прошлом. Модель реализованного в ПС клиент-серверного взаимодействия на основе REST API делает его уникальным в контексте приложений по изучению правил дорожного движения, так как данные пользователя о статистике, истории и трудных для него вопросах хранятся в удаленной базе данных, что помогает избежать привязки к конкретному цифровому устройству, а наличие личного кабинета позволяет реализовать полезную функцию в виде персонального контроля, в котором для пользователя формируются тесты на основе того, в каких вопросах чаще всего им были допущены ошибки. Также, даже те немногочисленные сведения о технологической составляющей, приведенные ранее, позволяют сделать вывод о том, что функциональность приложения легко расширяема, модули реализованы с помощью современных подходов и технологий, что при грамотной поддержке и продвижении программного продукта, делает его потенциальным лидером на рынке программных средств для изучения правил дорожного движения. Стоит отметить, что реализация программного средства не завязана на теоретические материалы правил дорожного движения какой-либо страны - этот факт делает ПС гибким и позволяет использовать его в любой стране с любимыми правилами дорожного движения, стоит только поменять наполнение базы данных с вопросами и теоретическими сведениями. Это еще раз доказывает то, что ПС потенциально имеет колоссальный экономический успех, при условии создания своей базы данных с вопросами и теоретическими сведениями для каждой страны. В результате выполнения дипломной работы все перечисленные функции были успешно реализованы. Было разработано мобильное приложение на платформе iOS. Приложение обладает всеми классическими функциями ПС для изучения правил дорожного движения: тут есть несколько режимов тренировок с возможностью совершения неограниченного количества ошибок, а также несколько режимов контроля, где в каких-то из них пользователь сразу видит свои ошибки, в каких-то - узнает результат только при ответе на все вопросы. Отдельным пунктом стоит режим персональной тренировки описанный ранее. Авторизованные пользователи могут просматривать персональную историю прохождения контролей, изучать тенденции своего обучения при помощи статистики правильных ответов при прохождении тестов. Программное средство удовлетворяет все потребности как курсантов, так и органов, принимающих экзамен по правилам дорожного движения, что делает его незаменимым инструментом для получения и контроля знаний в этой области.

**Ключевые слова.** Apple, Swift, мобильное приложение, правила дорожного движения, тренировка, контроль, персональная тренировка, персональная статистика, история контролей.



# КЛИЕНТСКАЯ ЧАСТЬ ПРОГРАММНОГО СРЕДСТВА ДЛЯ ИЗУЧЕНИЯ ПРАВИЛ ДОРОЖНОГО ДВИЖЕНИЯ С ИСПОЛЬЗОВАНИЕМ БИБЛИОТЕКИ REACTJS

Маркович Ф.Ю.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Мельникова Е. В. – ассистент

Проанализировав статистику количества машин на тысячу человек в Республике Беларусь, можно заметить, что количество автомобилей с каждым годом постоянно растет, например, в 2010 году на тысячу человек приходилось 264 транспортных средства, а в 2019 уже 329, следовательно количество желающих получить водительские права тоже постоянно растет, а для успешного прохождения экзамена в ГАИ необходимо справиться с теоретической частью экзамена, которая вызывает трудности у большинства будущих автомобилистов, согласно данным ГАИ. Исходя из вышеперечисленных фактов, для темы дипломной работы была выбрана разработка программного средства, которое позволяет значительно повысить качество и уровень знаний по правилам дорожного движения.

Было разработано приложение, с которым пользователь может взаимодействовать посредством браузера, также скачать его на мобильное устройство, которое работает на операционной системе "IOS", но также приложением удобно пользоваться при помощи мобильного устройства и через браузер, так как веб-приложение обладает адаптивным интерфейсом.

Также хочу обратить внимание на то, что пользователь зарегистрировавшись в приложении, например на мобильном устройстве, успешно может войти в свой аккаунт в WEB приложении и продолжить обучение, не потеряв свои результаты, которые он получил на мобильном устройстве. Вышеприведенная возможность была достигнута при помощи использования для обмена данными между клиентскими устройствами и сервером REST API.

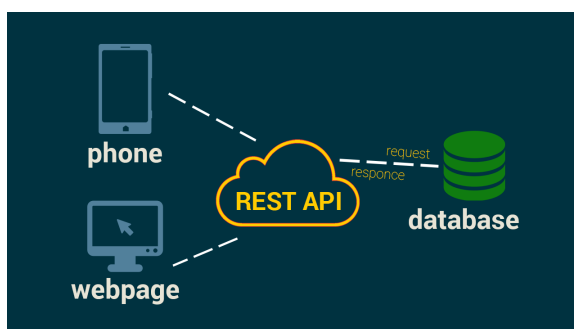


Рисунок 1 – Схема взаимодействия WEB, IOS приложения с сервером посредством REST API

В приложении реализовано два режима тестирования: тестирование, контроль. При выборе режима тестирования пользователь может отвечать неправильно на вопросы любое количество раз, но в режиме контроль после выбора двух неверных ответов тест будет завершен.

При помощи программного средства пользователь может улучшать свои знания по отдельным главам правил дорожного движения, проходя заранее подготовленные тесты, после выбора ответа на вопрос происходит отображение верного варианта ответа и есть возможность просмотреть главу правил, на которую ссылается вопрос. Также есть режим максимально приближенный к сдаче экзамена в ГАИ, то есть пользователь проходит тест, который состоит из 10 вопросов, выбранных случайным образом из каждой главы правил дорожного движения. Чтобы ликвидировать пробелы в своих знаниях пользователь приложения может проходить персонализированные тесты, которые состоят из вопросов, которые вызывают наибольшие трудности у клиента.

В программном средстве для пользователя реализована возможность просмотра истории пройденных тестов, то есть пользователь может просмотреть уже пройденные билеты и также повторить какие-то моменты в правилах. Также в приложении есть рейтинговая система, которая основывается на количестве правильно решенных вопросов, что предоставляет пользователю сравнить свои успехи с другими клиентами нашего программного средства. В приложении были исправлены выявленные недостатки конкурентов, что, как мне кажется, должно сделать программное средство достаточно успешным на рынке программного обеспечения в сфере образования по правилам дорожного движения.

УДК

# СЕРВЕРНАЯ ЧАСТЬ ПРИЛОЖЕНИЯ ДЛЯ ИЗУЧЕНИЯ ПРАВИЛ ДОРОЖНОГО ДВИЖЕНИЯ НА ЯЗЫКЕ ПРОГРАММИРОВАНИЯ JAVA

*Лукьянчик И.А., студент гр.751004*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Мельникова Е.В.*

**Аннотация.** В наше время вождение автомобиля становится неотъемлемой частью человеческой жизнедеятельности, что делает вопрос получения водительских прав как никогда актуальным. На сегодняшний день как в нашей стране, так и в большинстве стран мира, для тренировок и контроля знаний Правил Дорожного Движения используются программные средства, созданные с помощью информационных технологий. Постоянное совершенствование настолько часто используемых инструментов, их обновление с применением современных подходов и технологий являются необходимыми процессами на пути к улучшению качества получения и контроля знаний. Цель работы - разработать часть мобильного приложения для тренировки и сдачи теоретического экзамена для получения водительских прав.

После постановки задачи были сформированы требования к программному средству:

Платформы:

- iOS;
- Web.

Функционал:

- Тренировка по главам ПДД;
- Тренировка по тематическим билетам;
- Тренировка по случайному билету;
- Контроль по тематическим билетам;
- Контроль по случайному билету в закрытом режиме;
- Контроль по случайному билету в открытом режиме;
- Персональная тренировка (для авторизованных пользователей);
- Персональная статистика прохождения контролей (для авторизованных пользователей);
- История контролей (для авторизованных пользователей).

В результате выполнения дипломной работы все перечисленные функции были успешно реализованы. Было разработано REST API приложения, с которым пользователь может взаимодействовать посредством браузера, также скачать его на мобильное устройство, которое работает на операционной системе "IOS", но также приложением удобно пользоваться при помощи мобильного устройства и через браузер, так как веб-приложение обладает адаптивным интерфейсом. Также хочу обратить внимание на то, что пользователь, зарегистрировавшись в приложении, например, на мобильном устройстве, успешно может войти в свой аккаунт в WEB приложении и продолжить обучение, не потеряв свои результаты, которые он получил на мобильном устройстве. В приложении реализовано два режима тестирования: тестирование, контроль. При выборе режима тестирования пользователь может отвечать неправильно на вопросы любое количество раз, но в режиме контроль после выбора двух неверных ответов тест будет завершен. При помощи программного средства пользователь может улучшать свои знания по отдельным главам правил дорожного движения, проходя заранее подготовленные тесты, после выбора ответа на вопрос происходит отображение верного варианта ответа и есть возможность просмотреть главу правил, на которую ссылается вопрос. Также есть режим максимально приближенный к сдаче экзамена в ГАИ, то есть пользователь проходит тест, который состоит из 10 вопросов, выбранных случайным образом из каждой главы правил дорожного движения. Чтобы ликвидировать пробелы в своих знаниях пользователь приложения может проходить персонализированные тесты, которые состоят из вопросов, которые вызывают наибольшие трудности у клиента. В программном средстве для пользователя реализована возможность просмотра истории пройденных тестов, то есть пользователь может просмотреть уже пройденные билеты и также повторить какие-то моменты в правилах. Стоит отметить, что реализация программного средства не завязана на теоретические материалы правил дорожного движения какой-либо страны - этот факт делает ПС гибким и позволяет использовать его в любой стране с любимыми правилами дорожного движения, стоит только поменять наполнение базы данных с вопросами и теоретическими сведениями. Это еще раз доказывает то, что ПС потенциально имеет колоссальный экономический успех, при условии создания своей базы данных с вопросами и теоретическими сведениями для каждой страны.

**Ключевые слова.** Java, Spring, серверная часть, REST API, правила дорожного движения, тренировка, контроль, персональная тренировка, персональная статистика, история контролей.

# ВЕБ-ПРИЛОЖЕНИЕ ДЛЯ СИНХРОННОГО ВОСПРОИЗВЕДЕНИЯ МУЗЫКИ

Пашковская П. А.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Мельникова Е. В. – ассистент каф.ПОИТ

Миллионы людей по всему миру слушают музыку ежедневно. Она может успокаивать, воодушевлять, поднимать настроение, способна выражать радость и печаль, передавать мысли и чувства. Музыка — это неотъемлемая часть практически любой культуры, как современной, так и ушедшей в прошлое.

Совместное прослушивание музыки — это общественная деятельность. Однако в условиях пандемии люди зачастую лишены возможности слушать музыку вместе с друзьями, тут же делясь мыслями и эмоциями, которые она у них вызывает.

Целью данного проекта является создание средства синхронного воспроизведения музыки на разных устройствах, которое позволило бы людям, находящимся на расстоянии, слушать музыку вместе.

Работа приложения организована на основе сервера потокового мультимедиа с открытым исходным кодом Iccast, обычно используемого для создания интернет-радиостанций и других частных медиа-серверов.

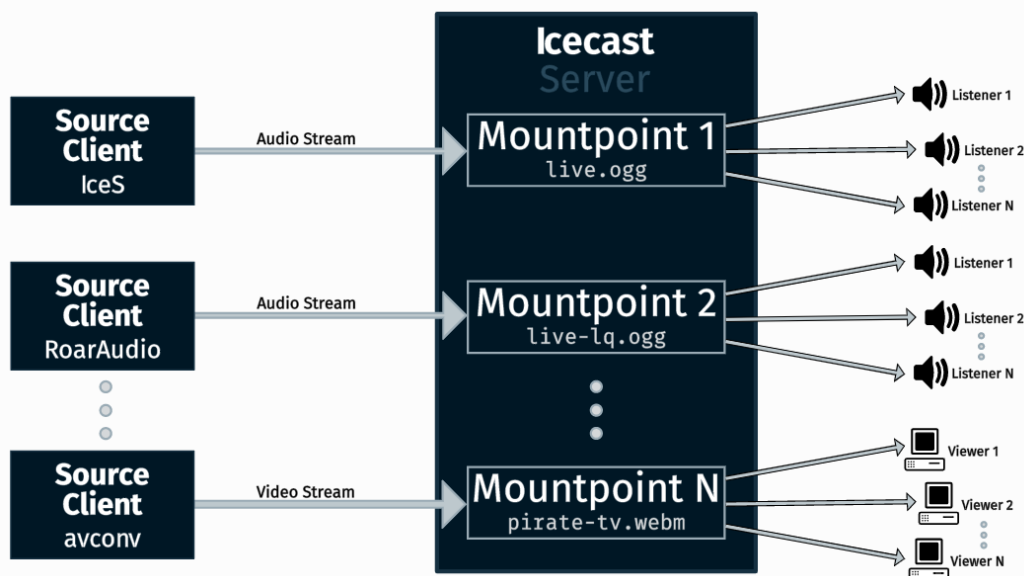


Рисунок 1 – Схема работы сервера Iccast

Исходный клиент подключается к точке монтирования на сервере Iccast и отправляет на него аудио- или видеоданные. Слушатели подключаются к точке монтирования, и Iccast отправляет поток им. К серверу Iccast будут подключаться все слушатели потока. Исходный клиент, как правило, работает не на Iccast, а на другом компьютере, но в этом нет необходимости.

Каждый сервер Iccast может содержать несколько потоков, которые называются точками монтирования. Точка монтирования — это уникальное имя на сервере, идентифицирующее конкретный поток. Оно выглядит как имя файла, например /stream.ogg. Слушатель может одновременно прослушивать только одну точку монтирования. Это означает, что один сервер Iccast может содержать несколько трансляций с разным контентом или, возможно, одну и ту же трансляцию, но с потоками с разным битрейтом или качеством. В этом случае каждая трансляция или поток представляет собой отдельную точку монтирования.

#### Список использованных источников:

1. Documentation and Information – Iccast [Электронный ресурс]. – Электронные данные. – Режим доступа: <https://iccast.org/docs>.

# МИКРОСЕРВИСНЫЙ СПОСОБ РАЗРАБОТКИ КЛИЕНТСКОЙ ЧАСТИ ВЕБ-ПРИЛОЖЕНИЙ

Голубцов М.В

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Серебряная Л.В. – канд. тех. наук

Реализация микрофронтенда для внедрения в веб-приложения, построенное с применением архитектурного подхода микрофронтенды. В данной работе будут описаны монолитные и микросервисные архитектуры, принципы использования, их преимущества и недостатки. Описаны основные проблемы использования микросервисной архитектуры в frontend части веб-приложения и приведены возможные способы их решения.

По статистике с каждым днем объем кода в веб-приложениях стремительно возрастает. В большинстве случаев из-за этого возникает увеличение параметров приложения, таких как:

- время разработки;
- время тестирования;
- временной интервал между релизами.

Большие приложения, которые содержат эти проблемы, чаще всего разработаны по принципу монолитной архитектуры.

Монолитная архитектура – схема построения приложения, из которой крайне сложно выделить автономные блоки из-за сильной связности кода. Главными достоинствами такого архитектурного подхода являются быстрота и легкость реализации, простота развертывания приложения.

Чаще всего все frontend-приложения монолитны, поэтому они в ходе своего развития превращаются в большой объем сложно поддерживаемого кода.

Для решения вышеперечисленных проблем, используется микросервисная архитектура.

Микросервисная архитектура – схема построения приложения, направленная на взаимодействие небольших, слабосвязанных и легко изменяемых модулей – микросервисов.

Из схемы микросервисной архитектуры веб-приложения видно, что каждый сервис в системе является отдельным приложением, кроме frontend части веб-приложения.

В настоящее время frontend часть веб-приложения – это не просто HTML-файлы, а сложные компоненты, которые могут содержать различный пользовательский интерфейс и бизнес логику. Поэтому микросервисный архитектурный подход можно использовать и в этой части веб-приложения.

Как известно, актуальная frontend-экосистема очень разнообразна и может быть достаточно сложной. И в ходе использования микросервисного подхода построения frontend части веб-приложения могут возникнуть определенные проблемы.

Исследование данной темы позволило выявить основные возможные проблемы. Ниже приведен их перечень:

– не согласованное поведение пользовательского интерфейса, так как у нас несколько абсолютно автономных веб-приложений. Возможное решение этой проблемы: реализация общей библиотеки для стилизации пользовательского интерфейса, но тогда все microfrontends перестают быть полностью независимыми.

– переписывание одной командой кода для стилизации веб-страницы другой командой. Можно решить данную проблему путем ограничения области видимости кода для стилизации веб-страницы с помощью селекторов.

– сделать глобальную информацию общей для разных microfrontend. Вариантов решений данной проблемы множество, но самый простой способ – использование кастомных событий в HTML5.

– маршрутизация на стороне клиента. In-memory реализация маршрутизации может решить данную проблему.

В настоящее время архитектурный подход «Microfrontends» стремительно развивается и привлекает внимание большего числа компаний, так как этот архитектурный подход имеет много достоинств. Анализ разработок показал, что в скором времени данная стратегия станет по умолчанию методом разработки в больших командах.

## Список использованных источников:

1. Интернет-ресурс [micro-frontends.org](http://micro-frontends.org) / Mechael Geers.
2. Онлайн конференция JSConf.Asia 2018 лекция Wilson Mendes: Micro Frontend.

# НАСТРАИВАЕМЫЕ ФИЛЬТРЫ ДЛЯ МОБИЛЬНЫХ ПРИЛОЖЕНИЙ

Усевич А.М.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Серебряная Л.В. – кандидат технических наук, доцент

О возможности расширения стандартных фильтров путём добавления настраиваемых пользователями групп фильтров, а также возможности называть данные группы пользователями по усмотрению.

В настоящее время во многих мобильных приложениях используются различные фильтры для выборки определённой информации. Благодаря разнообразному оформлению и широкому виду фильтров можно сделать описание товаров, представленных в приложении, не только функциональным, но и придать ему эстетичный и лаконичный вид, а пользователи легко смогут сузить поиск и выстроить его результаты в любом удобном для них порядке.

С целью изучения возможностей различных фильтров в мобильных приложениях автором были проанализированы несколько мобильных приложений такие как «Е-доставка», «Menu.by», «Fander», «iv», «emovi», «VK», «YouTube». В приложении «Е-доставка» есть различные фильтры по категориям продуктов, такие как «Овощи и фрукты», «Хлеб и кондитерские изделия», «Бакалея» и другие. «Menu.by» имеет большое разнообразие фильтров и кроме доставки еды известных брендов, предлагает доставку цветов. «Fander» и «emovi» являются приложениями по рекомендации фильмов и соответственно в них есть различные фильтры по жанрам предлагаемых к просмотру кинофильмов. «ivi» — онлайн-кинотеатр, коллекция которого насчитывает свыше 80 000 сериалов и фильмов: от голливудских студий до фестивальных премьер, от эксклюзивных новинок до классики кинематографа. В нём есть огромное количество фильтров по жанрам, фильмам и актёрам, а также в приложение реализован динамический поиск, что делает его удобным в использовании. «VK» является одной из самых популярных социальных сетей в Беларуси и имеет разнообразие фильтров как для пользовательских сообщений и чатов, так и для ленты новостей. «YouTube» является популярнейшим видеохостингом в мире с огромным количеством фильтров по различным категориям видео.

Таким образом, можно отметить, что в рассмотренных приложениях есть достаточное количество фильтров, чтобы пользователь мог быстро найти необходимую ему информацию. В каждом приложении фильтры стараются отобразить определённую группу товаров, которые пользователи наиболее часто ищут.

Однако практически во всех приложениях отсутствует возможность для пользователя создать свои группы фильтров, дать им названия и сохранить их для повторного использования. Такая функция пригодилась бы в приложениях, где пользователи часто применяют одинаковые группы фильтров, например, для повторного заказа еды или выбора фильмов с определённой группой жанров. Также можно предложить пользователю поделиться своими фильтрами с другими пользователями. Ещё можно отслеживать настраиваемые личные фильтры и если большое число пользователей создаёт одинаковые фильтры, то возможно добавить таковые как стандартные. Также можно добавить возможность чтобы при использовании поисковой строки отображались созданные пользователем фильтры. Для этого стоит реализовать алгоритм сохранения пользовательских фильтров на стороне сервера в БД, а не только на устройстве пользователя. После этого стоит подумать над обеспечением безопасности от возможной атаки MITM во время сохранения этих фильтров на сервере приложений. Можно также реализовать возможность пользователями создавать фильтры по неким ключевым словам или словосочетаниям для быстрого поиска определённого наименования. Это также позволит оценить какие конкретные товары являются приоритетными для пользователей. Кроме того, можно попробовать разработать нейронную сеть, которая на основе данных пользователей будет определять новые группы фильтров для их повсеместного внедрения.

Таким образом, разработка и внедрение настраиваемых фильтров является актуальной и востребованной. Такой функционал приложения сможет существенно упростить поиск нужной информации и сделает пользовательский опыт намного более приятным благодаря тому, что фильтры создаются на основе поведения пользователей и закрывают их потребности. Это также позволит вести более точную статистику популярности определённых наименований среди пользователей.

## Список использованных источников:

1. Дарвин Ян Ф. «Android. Сборник рецептов» Вильямс, 2018 год, 768 стр., 2-е изд., ISBN: 978-5-9909446-0-2.
2. Bill Phillips, Brian Hardy «Big Nerd Ranch», 2013 год, 602 стр., ISBN 0321804333, 9780321804334.

# ПРОГРАММНОЕ СРЕДСТВО ВИЗУАЛИЗАЦИИ ОСНОВНЫХ ОПЕРАЦИЙ ЦИФРОВОЙ ОБРАБОТКИ СИГНАЛОВ

Войтешонок А.Л., Деменковец Д.В.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Леванцевич В.А. – ст. преподаватель, м.т.н.

В работе рассматривается программное средство визуализации основных операций цифровой обработки сигналов. Выполняется обзор аналогов, существующих подобных программных средств. Рассматривается структура аудиофайла WAV файла и последующее применение его для визуализации операций ЦОС, дискретное преобразование Фурье, корреляции и свертки. Предложен пример использования разработанного программного средства.

В настоящее время активно используются методы и операции цифровой обработки сигналов. Однако, к сожалению, многие не понимают принципов работы этих методов. И для того, чтобы прояснить данный недочет было спроектировано программное средство визуализации основных операций цифровой обработки сигналов, которое демонстрирует визуально результаты работ этих методов, что помогает лучше разобраться в принципе их действия.

К немногочисленным аналогам данного программного средства можно отнести такие программы, как MathCAD, MatLab и LabView, однако у данных аналогов есть некоторые недостатки. К примеру, у основного модуля MathCAD в принципе отсутствует функционал для работы с цифровыми сигналами. У MatLab и LabView имеется недостаток в том, что эти программные средства не работают с аудиофайлами. В случае с MatLab, требуется написать некий код и задавать сигнал вручную. В LabView, требуется построить блок-схему, которая реализует функционал операции цифровой обработки, но при этом сигнал также задается вручную.

В разработанном программном средстве входным сигналом служит аудиофайл формата WAV. Структура аудиофайла формата WAV представлена на рисунке 1.

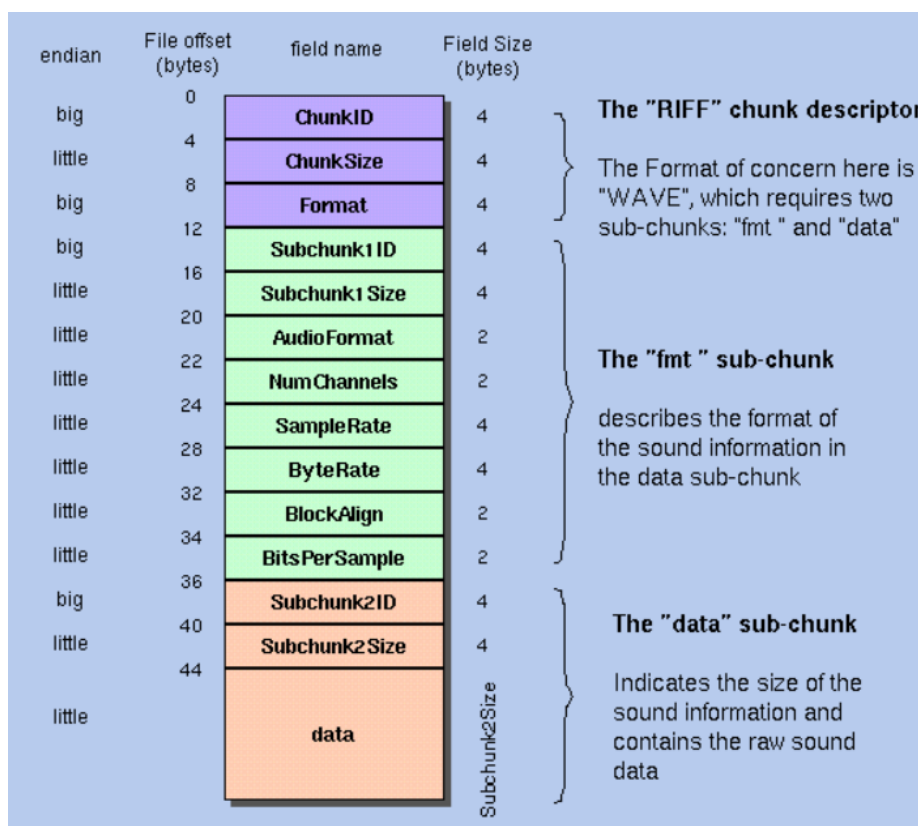


Рисунок 1 – Структура WAV-файла

На рисунке видно, что структура аудиофайла содержит подробное описание. В данном случае рассматривается обычный WAV файл (Windows PCM). Он представляет собой две, четко делящиеся области. Одна из них — заголовок файла, другая — область данных. В заголовке файла хранится следующая информация:

- размер файла;
- количество каналов;
- частота дискретизации;

- количестве бит в наборе (глубина звучания).

Звук состоит из колебаний, которые при оцифровке приобретают ступенчатый вид. Этот вид обусловлен тем, что компьютер может воспроизводить в любой короткий промежуток времени звук определенной амплитуды (громкости) и этот короткий момент не бесконечно короткий. Продолжительность этого промежутка и определяет частота дискретизации. К примеру, имеется файл с частотой дискретизации 44.1 kHz, это значит, что тот короткий промежуток времени равен 1/44100 секунды (следует из размерности величины Гц = 1/с). Современные звуковые карты поддерживают частоту дискретизации до 192 kHz. Однако в нашем случае для нас представляет интерес только та информация, расположенная за 44 байтом, который отвечает за входные данные, значения которого используются для построения входного дискретного сигнала.

Программное средство визуализации представляет собой приложение для ОС Windows. На стартовом экране имеются три блока с кнопками: «Play» - кнопка, предназначенные для воспроизведения аудиофайла; «Record» - кнопки для записи файла; «DSP» - кнопки, отвечающие за соответствующие функции преобразования ЦОС. При нажатии кнопки «Start» блока «Record», происходит запись исходного аудиофайла, и после окончания записи при нажатии на кнопку «Stop» этого же блока на экране появляется визуальное отображение аналогового вида нашего аудиофайла.

Для дальнейшей работы с нашим аудиофайлом мы преобразуем наш сигнал в дискретный вид, что открывает спектр операций над ним, такие как корреляция:

$$r(n) = \sum_{k=0}^n f(n+k) * g(k), n = 0 \dots N-1 \quad (1),$$

где  $f(n+k)$ ,  $g(k)$  – исходные сигналы;  $k$  – текущий элемент последовательности;  $n$  – длина последовательности.

Свертка:

$$s(t) = x(t) * h(t) \quad (2),$$

где  $h(t)$  – импульсная характеристика линейного фильтра,  $x(t)$  – входной сигнал.

Дискретное преобразование Фурье:

$$X_k = \sum_{n=0}^{N-1} x_n e^{-\frac{2\pi i}{N}kn} \quad (3),$$

Обратное преобразование Фурье:

$$x_n = \frac{1}{N} \sum_{k=0}^{N-1} X_k e^{\frac{2\pi i}{N}kn} \quad (4),$$

где  $N$  — количество значений сигнала, измеренных за период, а также количество компонент разложения;  $x_n$ ,  $n = 0 \dots N-1$ , — измеренные значения сигнала (в дискретных временных точках с номерами  $n = 0 \dots N-1$ ), которые являются входными данными для прямого преобразования и выходными для обратного;  $X_k$ ,  $k = 0 \dots N-1$  —  $N$  комплексных амплитуд синусоидальных сигналов, слагающих исходный сигнал; являются выходными данными для прямого преобразования и входными для обратного; поскольку амплитуды комплексные, то по ним можно вычислить одновременно и амплитуду, и фазу.

К преимуществам программного средства можно отнести:

- визуальное представление основных операций цифровой обработки сигналов;
- работа с WAV аудиофайлами;
- возможность работать не со всем файлом, а только с его частью.

Недостатками программного средства являются:

- невозможность работы программного средства с другими форматами аудиофайлов;
- для файлов большого размера будет использовано много времени на обработку и получение результатов операций цифровой обработки сигналов;
- погрешность при восстановлении сигнала в аналоговый вид, что приводит к изменению первоначального аудиофайла.

Программное средство визуализации основных операций цифровой обработки сигналов удобно использовать для демонстрации результатов работы основных операций цифровой обработки сигналов и последующего объяснения принципов их работы в учебных целях.

**Список использованных источников:**

1. Структура WAV файла [Электронный ресурс]. – Режим доступа: <https://audiocoding.ru/articles/2008-05-22-wav-file-structure/>. – Дата доступа: 04.04.2021.
2. Лайонс Р. Цифровая обработка сигналов: Второе издание / Р. Лайонс – М.: ООО «Бином-Пресс», 2006 г. – 656 с.:ил.

# ПРОГРАММНОЕ СРЕДСТВО МОНИТОРИНГА ФИЗИЧЕСКОГО РАЗВИТИЯ ДЛЯ МОБИЛЬНОЙ ПЛАТФОРМЫ ANDROID

Калтович В.А.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Парамонов А.И. – канд. техн. наук

Описывается программное средство, которое помогает пользователям в отслеживании показателей их физического развития. Приложение может использоваться в повседневной жизни и предоставляет пользователю актуальную информацию за счет применения анализа данных и автоматического расчёта различных нормативных показателей. Программное средство позволяет отслеживать и анализировать изменения показателей в динамике и отображает пользователю информацию в доступной и лёгкой для восприятия форме.

Мониторинг физического развития человека предполагает отслеживание его физических показателей и их последующий анализ с целью выявления отклонений. Разрабатывается программное средство, которое предоставляет такие возможности и отображает их в наглядной графической форме. Данное программное средство помогает пользователям в подсчёте значений, необходимых для поддержания здорового образа жизни и физической формы. Составленная индивидуальная программа определяет начальный вектор движения и даёт понимание того, что нужно делать для достижения цели. Добавление и отслеживание целей, создание и просмотр тренировок позволяют пользователю держать всю необходимую информацию в одном приложении, что позволяет сэкономить время на переключении между приложениями и способствует сохранению мотивации пользователя.

Целевой аудиторией программы являются начинающие спортсмены, люди, которые следят за своим здоровьем и фигурой, а также люди, которые хотят достичь необходимой физической формы.

На данный момент рынок заполнен огромным количеством фитнес-приложений, однако каждое из них создано для отдельной цели, например: для ведения дневника питания или для создания и отслеживания тренировок, для подсчёта количества шагов или для отслеживания изменения параметров.

Данное программное средство является мобильным приложением для платформы Android [1], которое разработано на языке программирования Kotlin [2].

Интерфейс программного средства содержит 8 основных экранов: экран для входа в приложение; 2 экрана регистрации - экран для создания данных для входа и экран для внесения данных о физических параметрах; стартовый экран – содержит информацию о количестве шагов, пройденных пользователем за день, количество необходимых калорий и питательных элементов; экран прогресса – содержит графики с отображением динамики прогресса пользователя; экран целей – содержит список целей пользователя и информацию об их статусе; экран тренировок – содержит информацию о тренировках пользователя и позволяет создавать новые на основе упражнений, хранящихся в базе данных; экран профиль - содержит данные пользователя и позволяет изменять их.

После внесения пользователем данных о его физическом состоянии, программа по этим параметрам производит расчёт таких показателей, как: Индекс массы тела (далее ИМТ), базовый расход калорий, потребность в питательных веществах, суточная норма воды [3].

ИМТ рассчитывается по формуле (1).

$$I = \frac{m}{h^2} \quad (1)$$

где  $m$  - масса человека в килограммах,  $h$  - рост человека в метрах.

Уравнение Миффлина для расчёта базового расхода калорий представлено формулой (2).

$$K = (10 * m) + (6.25 * h) - (5 * l) + Q1 \quad (2)$$

где  $m$  – масса человека в килограммах,  $h$  - рост человека в метрах,  $l$  - возраст человека в годах,  $Q1$  – гендерный коэффициент ( равен 5 для мужчин и -161 для женщин).

Для получения количества калорий, необходимых пользователю в зависимости от образа жизни, результаты, полученные с использованием формул 2, 3, следует умножить на коэффициент активности, выбранный пользователем по таблице 1.

Для расчёта необходимых питательных элементов (белки, жиры и углеводы) используется формула (3).

Таблица 1 – Коэффициенты активности

Коэффициент активности	Описание
------------------------	----------



1,2	Минимальные нагрузки (сидячая работа)
1,375	Немного дневной активности и легкие тренировки 1-3 раза в неделю
1,4625	Тренировки 4-5 раз в неделю или работа средней тяжести
1,550	Интенсивные тренировки 4-5 раз в неделю
1,6375	Ежедневные тренировки
1,725	Ежедневные интенсивные тренировки или тренировки 2 раза в день
1.9	Тяжелая физическая работа или сверхинтенсивные тренировки 2 раза в день

$$P = \frac{n * k}{Q2} \quad (3)$$

где n - суточная норма калорий человека, k - процент белка от суточной потребности в питательных элементах, Q2 – коэффициент элементов (равен 4 для белков и углеводов, 9 - для жиров).

Для расчёта суточной нормы воды используется формула (4).

$$W = m * Q3 \quad (4)$$

где m - масса человека в килограммах, Q3 – коэффициент (равен 31 для женщин и 35 для мужчин).

Для оценки ИМТ используется таблица 2, содержащая соответствие значений ИМТ и его интерпретации.

Таблица 2 – Значение ИМТ

Индекс массы тела	Интерпретация
от 15,99 и менее	Выраженный дефицит массы тела
от 16,0 до 18,49	Недостаточная масса тела
от 18,5 до 24,99	Нормальная масса тела
от 25,0 до 29,99	Избыточная масса тела
от 30,0 до 34,99	Ожирение первой степени
от 35,0 до 39,99	Ожирение второй степени
от 40 и более	Ожирение третьей степени

Пользователю доступен функционал для создания тренировок на основании упражнений, хранящихся в базе данных. Каждому упражнению соответствует количество затрачиваемых килокалорий на 50 кг за 1 час.

Для получения количества калорий, затраченных на выполнение упражнений, используется формула (5).

$$K = a * \frac{m}{50} * \frac{t}{60} \quad (5)$$

где a - количество затрачиваемых килокалорий на 50 кг за 1 час, m - масса человека в килограммах, t – время, затраченное на выполнение упражнения.

Разработанное программное средство автоматически рассчитывает приведённые показатели, анализирует их и выдаёт пользователю на основании результатов советы, необходимые для начала движения в сторону намеченной цели, что дает возможность в процессе занятий отследить правильность действий. Благодаря данной программе пользователю не придётся разбираться самостоятельно в большом количестве информации, необходимой для того, чтобы начать работу над своей физической формой.

**Список использованных источников:**

1. Что такое Android [Электронный ресурс]. – Режим доступа: [https://www.android.com/intl/ru\\_ru/what-is-android/](https://www.android.com/intl/ru_ru/what-is-android/). – Дата доступа: 02.04.2021.

2. Open Source Community. Руководство по языку Kotlin. [Электронный ресурс]. – Режим доступа: <https://kotlinlang.ru/>. – Дата доступа: 02.04.2021.

3. Юлианна Плискина: Счетчик калорий. Основы правильного питания / Издательство: Эксмо-Пресс, 2017. – 192 с.

# ПРОГРАММНОЕ СРЕДСТВО АВТОМАТИЗАЦИИ НАЙМА ПЕРСОНАЛА ИТ-КОМПАНИИ

Дубовская Е.В.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Данилова Г.В. – ассистент каф.ПОИТ

Постоянный рост рынка труда в белорусском ИТ-секторе, одновременно с дефицитом квалифицированных специалистов создаёт особые требования кандидатов к компаниям, а соответственно, и к процессу найма сотрудников. Для автоматизации поиска сотрудников разработано программное средство, представляющее из себя веб-приложение. Приложение учитывает структуру компании, как по отделам, так и по проектам, и ориентировано на организацию найма персонала в соответствии с требованиями компаний.

В современном мире технологии становятся настолько сложными, а процессы настолько запутанными, что за их динамикой становится крайне трудно уследить. Данное правило всецело относится в том числе и к подбору персонала. Квалифицированные и правильно подобранные команды являются неотъемлемой частью компании любого размера. Поэтому автоматизация и структуризация данного процесса является залогом успешной работы предприятия. Найм персонала (ИТ-рекрутинг) – быстро развивающееся направление, поэтому приложения, ориентированные на данный рынок, пользуются большой популярностью. Приложения данного направления решают такие задачи, как снижение нагрузки на менеджеров и собеседующих, хранение истории коммуникации с кандидатом, а также общая формализация процесса найма [1].

Анализ аналогов, а именно программных средств в сфере найма персонала, показал, что большинство решений являются универсальными конфигурируемыми системами по работе с персоналом для фирм разного направления. Типичными проблемами таких систем являются трудоемкая конфигурация, излишняя интеграция со сторонними сервисами (сайты, площадки), неиспользуемыми при найме в ИТ-секторе, множество излишних модулей, сложная поддержка системы, малая расширяемость системы.

Основная идея данной работы заключается в создании приложения, которое должно повысить эффективность найма в сфере информационных технологий. Приложение позволит добавить кандидатов в систему, редактировать информацию в процессе работы с ними, изменять статус кандидата. Также приложение будет содержать информацию обо всех проектах и вакансиях в компании. Внутри системы кандидаты будут соотноситься с вакансиями, вакансии – с проектами. Дизайн пользовательского интерфейса приложения представлен на рисунке 1.

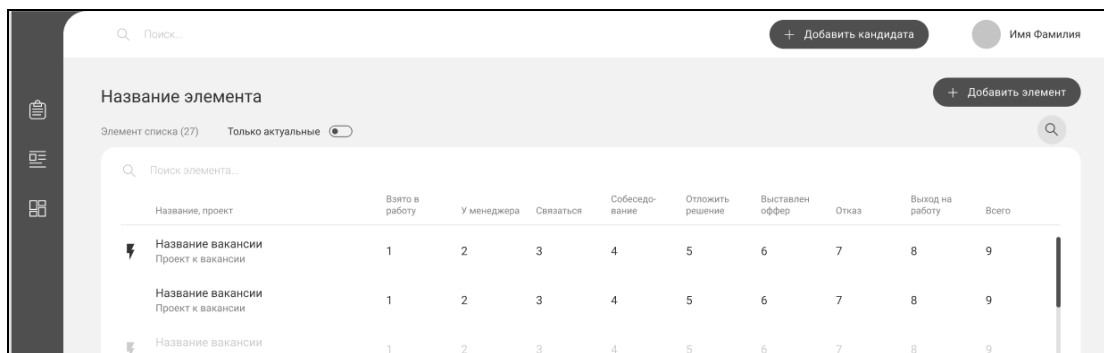


Рисунок 1 – UX-страницы «Вакансии»

Таким образом, приложение будет представлять собой систему для внутреннего пользования компании с разграничением прав (администратор, менеджер, рекрутер), которая централизованно хранит информацию о структуре компании, потенциальных сотрудниках, вакансиях, проектах. Программа имеет прикладное значение. Её внедрение поможет специалистам по найму персонала ИТ-компаний синхронизировать совместную работу, упростить подбор квалифицированных кадров.

#### Список использованных источников:

1. ИТ-рекрутинг: особенности работы и как подбирают специалистов в Google [Электронный ресурс]. – Режим доступа: <https://hurma.work/ru/blog/it-rekruting-osobennosti-raboty-i-kak-eto-delayut-v-google/>. – Дата доступа: 29.03.2021.

# АВТОМАТИЗИРОВАННАЯ СИСТЕМА УПРАВЛЕНИЯ РЕСТОРАНОМ

*Крукович П.Н.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Видничук В.Н. – м. т. н., старший пр. каф. ПОИТ*

В докладе рассматривается тема автоматизации бизнес-процессов на предприятиях общественного питания.

С ростом численности ресторанов стремительно усиливается и конкуренция, что неизбежно приводит к необходимости эффективно и рационально использовать имеющиеся ресурсы. В этих условиях для успешного ведения бизнеса необходимо инвестировать в средства и инструменты его поддержания и развития. Один из основных инструментов развития ресторанного бизнеса – это современная система автоматизации ресторанов.

Современная система автоматизации ресторана – это профессиональная система управления рестораном, многофункциональная и легко модернизируемая. Целью автоматизации является повышение эффективности управления рестораном, ускорение обслуживания и минимизация возможных злоупотреблений, особенно воровства. Значительная доля успеха складывается из отличного сервиса и оперативной работы персонала. Именно возможности автоматизации ресторана позволяют оптимально сочетать скорость и качество.

Очевидны неоспоримые преимущества автоматизированного ресторана перед другими подобными заведениями: высокое качество сервиса и скорость обслуживания клиентов, отсутствие ошибок при оформлении заказа, обработка и передача заказа в автоматическом режиме, абсолютный контроль всех процессов от момента приема заказа до его исполнения, возможность непрерывно отслеживать финансовые результаты работы заведения.

Данная работа подразумевает создание системы, в которой будут восполнены недостатки существующих систем. Основная цель проекта заключается в том, чтобы включить в систему необходимые функции для обеспечения эффективного управления рестораном.

Основные функции автоматизированной системы управления рестораном:

- складской учет и бухгалтерский учет – автоматизация складских операций: приход и расход продуктов, полуфабрикатов, инвентаризация, списание ингредиентов при приготовлении блюд для кухни и бара;
- управление персоналом – учет рабочего времени, создание расписаний, контроль количества выписанных чеков;
- контроль деятельности заведения – бронирование столиков, создание планов помещений;
- создание программ лояльности и скидок – создание групп клиентов и специальных предложений;
- формирование меню, контроль ценообразования – формирование технических карт блюд, расчет себестоимости и прибыли от продажи блюд;
- планирование и прогнозирование – генерация отчетов за выбранный период, анализ полученных данных.

Применение автоматизированной системы позволит решать корпоративные задачи, такие как развитие предприятия и обеспечение непрерывности функционирования, а также экономические задачи, такие как максимизация товарооборота и конечной прибыли, обеспечение ликвидности и рентабельности предприятия, минимизация текущих издержек за счет роста эффективности работы. Ожидаемые результаты использования автоматизированной системы:

- Повышение прибыльности и снижение издержек предприятия;
- Контроль и оптимизация деятельности предприятия;
- Улучшение качества обслуживания посетителей;
- Предотвращение хищений и прочих злоупотреблений;
- Увеличение производительности труда персонала;
- Поддержка маркетинговых мероприятий.

Автоматизированная система состоит из нескольких модулей, включающих функционал, предназначенный для отдельных типов сотрудников. Для доступа к каждому из модулей сотрудник должен обладать определенным набором прав. Каждый модуль представлен отдельным веб-приложением. На веб-сервере происходит обработка данных, выполняются аналитические операции и формируются отчеты, выполняются операции по работе с базой данных. Автоматизированная система реализована с использованием облачных технологий, что делает ее еще более мобильной и эффективной в использовании. Схематическое представление архитектуры системы приведено на рисунке 1.

Модуль управления рестораном содержит функционал, необходимый для менеджеров и руководителей. Складской учет позволяет контролировать остатки, своевременно списывать товары, производить инвентаризацию, работать с технологическими картами, вести расчет калорийности блюд и продуктов. Подмодуль управления персоналом позволяет отслеживать и корректировать работу сотрудников. Также данный модуль предоставляет возможность создания различных программ лояльности и скидок, объединения клиентов в группы и формирования иерархии перехода клиентов из одной группы в другую.

Модуль кухни и бара используется официантами и поварами. Данный модуль помогает уменьшить количество времени, затрачиваемое на принятие заказа и обслуживание столика, а также увеличить скорость работы персонала и оборачиваемость столов. Для кухни разработан подмодуль визуализации заказов. Как только заказ был подтвержден в модуле точки продаж, уведомление автоматически появляется на веб-сайте данного модуля.

Модуль точки продаж позволяет официантам, барменам и кассирам создавать, редактировать, отменять и оплачивать заказы, разделять чеки, отслеживать степень готовности заказанных блюд, добавлять в заказ блюда отдельно для каждого гостя за столиком. Также в нем есть возможность производить бронирование столиков, просматривать расписание бронирования и получать уведомления о приближении времени брони.

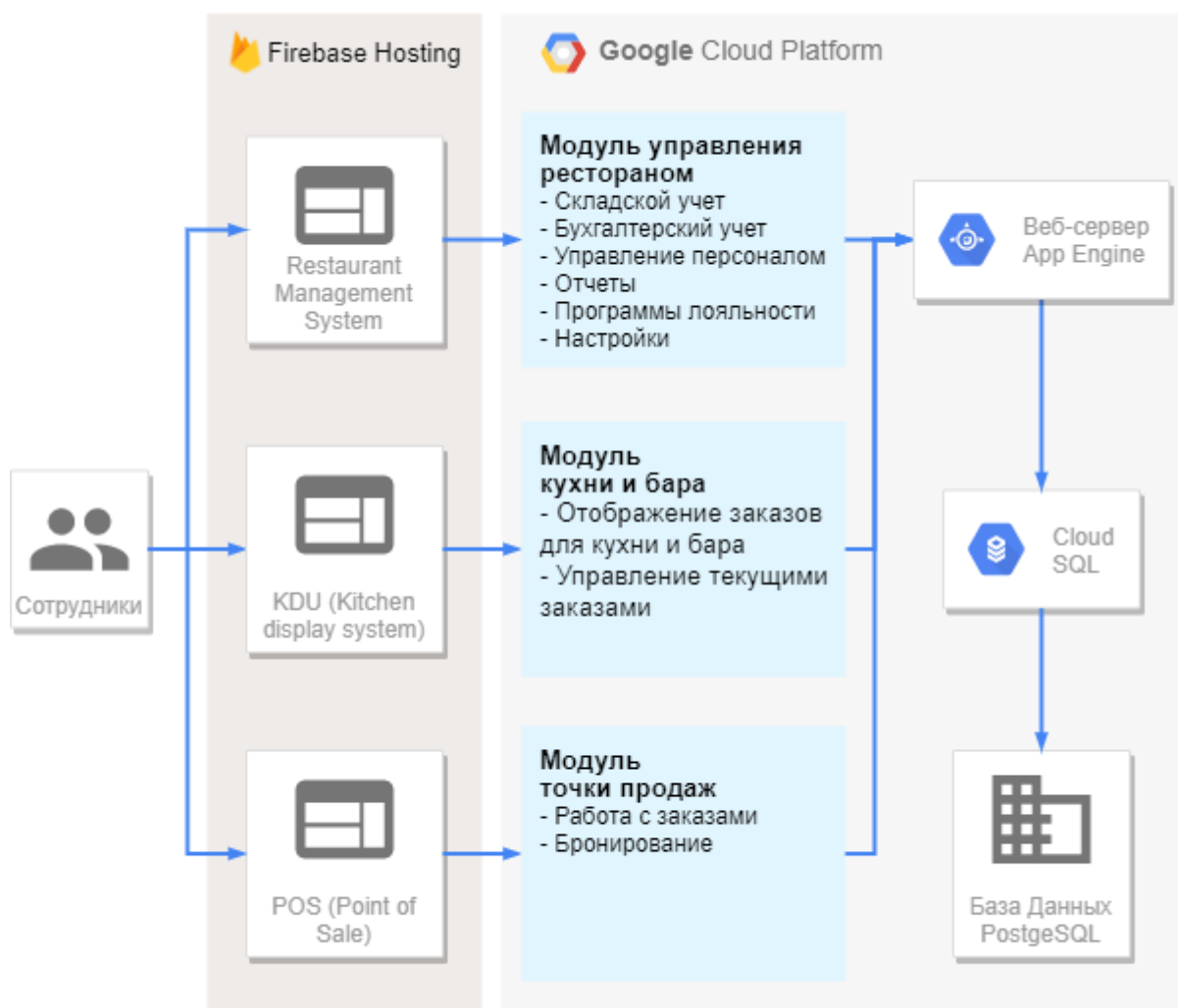


Рисунок 1 – Архитектура системы управления рестораном

Данная автоматизированная система управления рестораном позволит пользователям решать корпоративные задачи, такие как развитие предприятия и обеспечение непрерывности функционирования, а также экономические задачи, такие как максимизация товарооборота и конечной прибыли, обеспечение ликвидности и рентабельности предприятия, минимизация текущих издержек за счет роста эффективности работы эффективным решением проблем.

**Список использованных источников:**

1. Автоматизация складского учета [Электронный ресурс]. – Режим доступа: <https://www.datakrat.ru/software/r-keeper/avtomatizatsiya-ucheta>.
2. Бухгалтерский учет в общепите [Электронный ресурс]. – Режим доступа: <https://agentura-soft.ru/articles/AccountFood/>.
3. Норенков, И. Автоматизированные информационные системы: учеб. пособие / И. Норенков. – Москва: СПб.: Питер, 2011. – 526 с.

# ПРОГРАММНОЕ СРЕДСТВО МОДЕЛИРОВАНИЯ КАРТЫ НЕИСПРАВНОСТЕЙ ЗАПОМИНАЮЩИХ УСТРОЙСТВ С ЗАДАНЫМИ ПАРАМЕТРАМИ

*Наркевич И.Г., Деменковец Д.В.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Бахтизин В.В. – кандидат технических наук*

В работе рассматривается программное средство моделирования карты неисправностей запоминающих устройств с заданными параметрами. Рассматриваются существующие модели неисправностей запоминающих устройств. Предложен пример использования разработанного программного средства.

Вычислительная система может состоять из большого множества различных элементов, большинство которых не являются обязательными. Но одним из важнейших элементов, без которого вычислительная система не может функционировать, является запоминающее устройство. Это устройство служит для записи и хранения данных.

Причиной неисправного состояния запоминающего устройства является присутствие какого-то физического или механического дефекта либо их множества, при этом их количество и многообразие практически неограниченно. В зависимости от технологических особенностей при производстве памяти и внешних факторов при её эксплуатации могут появляться новые типы и разновидности дефектов. Установление факта появления дефекта и его классификация является весьма трудоёмкой и часто неразрешимой задачей. Это объясняется в первую очередь тем, что появление дефекта чаще всего можно определить лишь по косвенным признакам, как правило, по факту неправильной работы запоминающего устройства.

Современный этап развития методов и средств тестирования ОЗУ сопряжён со следующими основными проблемами. Во-первых, это неуклонный рост ёмкости ОЗУ, которая превышает  $10^9$  бит. Во-вторых, существенное отличие физической структуры ОЗУ от их логической организации, когда физически соседние запоминающие элементы имеют логически удалённые адреса. И, в-третьих, многие разновидности физических дефектов и отказов современных ОЗУ могут быть описаны только с помощью сложных моделей неисправностей ОЗУ, которые трудно обнаруживаются классическими тестами и их модификациями.

Для аналитического описания неисправных состояний памяти используются математические модели неисправностей, которые тем или иным образом отражают реальные физические дефекты запоминающих устройств. Основные физические дефекты этих устройств найдены и построены математические модели неисправностей, которые эти дефекты отражают. На основе имеющихся математических моделей можно построить компьютерные модели, которые будут иметь поведение, похожее на поведение реальных физических запоминающих устройств.

Первые модели неисправностей запоминающих устройств были предложены в начале 1980-х годов, которые включали константные неисправности, неисправности дешифратора адреса, неисправности взаимного влияния и кодочувствительные неисправности. Функциональные неисправности запоминающих устройств подразделяются на два подмножества: неисправности матрицы запоминающих элементов и неисправности электронного обрамления. Второе подмножество включает неисправности дешифраторов адреса и неисправности логики чтения-записи. Доминирующее значение имеют неисправности матрицы запоминающих элементов памяти, которые часто называют ячейками памяти.

К неисправностям матрицы ячеек запоминающих устройств в первую очередь относят неисправности, в которых участвуют: одна ячейка; две ячейки; несколько ячеек, в общем случае более чем две, без ограничений на их количество.

К неисправностям, затрагивающим одну ячейку запоминающего устройства, относят:

1. Константные неисправности (stuck-at faults – SAF). Неисправный запоминающий элемент памяти постоянно находится в состоянии логического нуля ( $s-a-0$ ) или логической единицы ( $s-a-1$ ), независимо от операций, выполняемых с неисправным элементом и другими элементами запоминающего устройства.

2. Переходные неисправности (transition faults - TF). Подобные неисправности характеризуются невозможностью перехода состояния неисправного запоминающего элемента из 0 в 1 ( $TF\uparrow$ ) или из 1 в 0 ( $TF\downarrow$ ) при выполнении соответствующих операций записи.

Среди неисправностей, в которых участвуют две ячейки запоминающего устройства, выделяют следующие неисправности:

1. Неисправности взаимного влияния (coupling fault - CF). При описании данной неисправности выделяют влияющую ячейку, определяемую её адресом  $i$ , изменение логического состояния которой воздействует на состояние зависимой ячейки с адресом  $j$ . Различают три типа неисправностей взаимного влияния:

а) инверсные неисправности взаимного влияния (inverse coupling faults - CFIn). При наличии данной неисправности изменение значения  $b_i$  влияющей ячейки вызывает инвертирование значения  $b_j$  зависимой ячейки. Возможны следующие виды CFIn:  $\wedge(\uparrow, \overline{b_j})$ ,  $\wedge(\downarrow, \overline{b_j})$ ,  $\vee(\uparrow, \overline{b_j})$ ,  $\vee(\downarrow, \overline{b_j})$ . Символ  $\wedge$  и  $\vee$  задают взаимное расположение влияющего и зависимого запоминающих элементов памяти. Первый символ  $\wedge$  означает, что запоминающий элемент с меньшим адресом влияет на запоминающий элемент с большим адресом ( $i < j$ ), а символ  $\vee$  используется в случае, когда адрес влияющего запоминающего элемента больше адреса зависимого запоминающего элемента ( $i > j$ );

б) неисправности взаимного влияния прямого действия (idempotent coupling faults - CFid). При изменении значения  $b_i$  влияющего запоминающего элемента происходит принудительная установка определённого логического значения 0 или 1 в зависимом запоминающем элементе. Различают восемь неисправностей прямого действия:  $\wedge(\uparrow, 0)$ ,  $\wedge(\uparrow, 1)$ ,  $\wedge(\downarrow, 0)$ ,  $\wedge(\downarrow, 1)$ ,  $\vee(\uparrow, 0)$ ,  $\vee(\uparrow, 1)$ ,  $\vee(\downarrow, 0)$ ,  $\vee(\downarrow, 1)$ ;

в) статические неисправности взаимного влияния (state coupling faults CFst). Переход зависимой ячейки в какое-либо состояние  $b_j$  возможен при определённом значении  $b_i$  влияющей ячейки. Возможно восемь неисправностей CFst:  $\wedge(0, 0)$ ,  $\wedge(0, 1)$ ,  $\wedge(1, 0)$ ,  $\wedge(1, 1)$ ,  $\vee(0, 0)$ ,  $\vee(0, 1)$ ,  $\vee(1, 0)$ ,  $\vee(1, 1)$ .

2. Кодочувствительные неисправности (pattern sensitive faults - PSF) рассматриваются как обобщение моделей неисправностей взаимного влияния. Для подобных неисправностей логическое состояние или изменение логического состояния одного запоминающего элемента запоминающего устройства может зависеть от содержимого (0 или 1) или от логических переходов из 1 в 0 или из 0 в 1 влияющих запоминающих элементов запоминающих устройств. В случае кодочувствительной неисправности PSFk, в которой участвует k запоминающих элементов запоминающего устройства, подразумевается, что влияющими запоминающими элементами в предельном случае могут быть любые из k – 1 из N запоминающих элементов запоминающего устройства, а зависимым один из оставшихся N - k + 1 запоминающих элементов. На практике при рассмотрении кодочувствительных неисправностей вводятся ограничения на количество запоминающих элементов k, так и на их местоположение.

В зависимости от эффекта влияния на базовый запоминающий элемент различают несколько классических типов кодочувствительных неисправностей NPSFk:

1. Пассивными кодочувствительными неисправностями (passive NPSF – PNPSF) являются неисправности, при которых состояние базового запоминающего элемента не может быть изменено для определённого кода в k – 1 соседних запоминающих элементах запоминающего устройства.

2. Под активными кодочувствительными неисправностями (active NPSF – ANPSF) понимают неисправности, в которых базовый запоминающий элемент изменяет своё состояние из-за изменения кода в соседних запоминающих элементах. Изменение кода для подобных неисправностей происходит в результате изменения состояния на противоположное только в одном соседнем элементе, в то время как остальные ячейки сохраняют предыдущее состояние.

3. Статические кодочувствительные неисправности (static NPSF – SNPSF) характеризуются тем, что для определённой комбинации значений в соседних запоминающих элементах состояние базового запоминающего элемента принудительно устанавливается в состояние 0 или состояние 1. Главным отличием статических неисправностей от активных кодочувствительных неисправностей является длительность процесса установления неверного значения в базовой ячейке. Для статических неисправностей это время существенно больше.

В структуре классических моделей неисправностей запоминающих устройств одно из самых видных мест в силу сложности их обнаружения тестами памяти занимают связные неисправности. Под связными неисправностями понимают неисправности различных типов, которые включают в себя общие ячейки. В случае двух неисправностей взаимного влияния, например  $(\uparrow, 1)$  и  $(\uparrow, 0)$ , с адресами ячеек i, j, l и г, участвующих в этих неисправностях и расположенных по возрастанию  $i < j < l < г$ , при j = г две неисправности взаимного влияния, имеющую общую ячейку-жертву, трансформируются в связную неисправность. Возможны любые сочетания типов и подтипов неисправностей, участвующих в конкретной связной неисправности. Их многообразие и количество всегда являлись препятствием для анализа их обнаружения конкретным тестом.

В результате было создано программное средство моделирования карты неисправностей запоминающих устройств с заданными параметрами. Данное приложение позволяет создавать карту неисправностей, изменяя состояние конкретной выбранной ячейки памяти или распределяя случайным образом определённое количество неисправностей с заданными параметрами, сохранять созданную карту в файл и загружать сохранённую карту из файла.

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

1. Ярмолик, В. Н. Контроль и диагностика вычислительных систем : учеб. пособие / В. Н. Ярмолик. – Минск : Изд-во : Бестпринт, 2019. – 387 с.

2. Ярмолик, С. В. Маршевые тесты для самотестирования ОЗУ : монография / С. В. Ярмолик, А. П. Занкович, А. А. Иванюк. – Минск : Изд. центр БГУ, 2009. – 271 с.

# ПРОГРАММНОЕ СРЕДСТВО ОТРИСОВКИ ТРЕХМЕРНЫХ СЦЕН С ИСПОЛЬЗОВАНИЕМ ТЕХНОЛОГИИ ТРАССИРОВКИ ЛУЧЕЙ

Яскевич Д.А., Бахтизин В.В.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Бахтизин В.В. – к.т.н., профессор

Рендеринг является основополагающим компонентом компьютерной графики, которая применяется в видеоиграх, инженерном моделировании, кинопроизводстве и многих других областях. С развитием платформы Nvidia RTX и графических API, позволяющих выполнять трассировку лучей в реальном времени, стало возможным реализовывать физически корректный рендеринг, который учитывает законы сохранения энергии, преломления и отражения света, а также реалистично моделировать всевозможные материалы.

Трассировка лучей – метод геометрической оптики, исследующий оптические системы путем отслеживания взаимодействия отдельных лучей с поверхностями. Идея использовать данную технологию для построения изображений трехмерных моделей появилась еще в 1960-х годах, однако аппаратные возможности того времени позволяли применять её только в оффлайн режиме, терпя значительные временные и ресурсные затраты, долгое время растеризация являлась основным методом визуализации трехмерных сцен. Трассировка лучей позволяет получать изображения с физически корректными тенями, отражениями, учитывать такие эффекты как преломление света на границе двух сред с различными показателями преломления.

Начиная с августа 2018 года, когда компания Nvidia анонсировала первый графический ускоритель серии RTX [1], аппаратные решения для применения трассировки лучей в рендеринге позволили использовать эту технологию в реальном времени. В ноябре 2020 года консорциум Khronos Group выпустил спецификацию расширений графического API Vulkan, предназначенных для трассировки лучей с использованием аппаратных возможностей графического ускорителя, включающих в себя возможность построения и использования акселерационных структур для геометрии сцены [2].

Целью данной работы является разработка программного средства отрисовки трехмерной сцены с использованием технологии трассировки лучей. В основе программного средства используется графический API Vulkan. Для хранения и загрузки трехмерных сцен был выбран один из самых широко используемых форматов – gltf. В качестве физически корректного метода моделирования материалов был выбран Metallic-Roughness подход, который описывает произвольный материал основываясь на теории микрограней [3], согласно которой любая поверхность при сильном увеличении может быть представлена как набор микроскопических зеркал, называемых микрогранями. Металличность и шероховатость, полученные в результате физических экспериментов над материалами, в виде коэффициентов используются в процессе рендеринга. Ввиду шероховатости поверхности эти микрозеркала могут быть ориентированы в разных направлениях, что влияет на её отражающие свойства, пример приведен на рисунке 1.

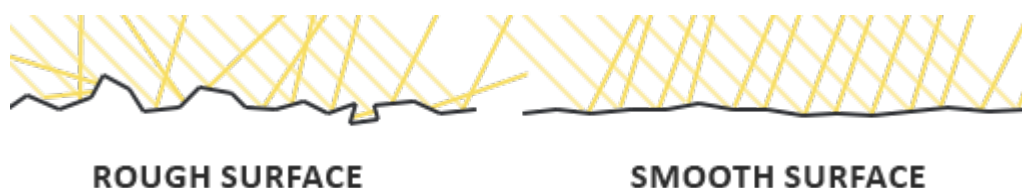


Рисунок 1 – Отражение от шероховатой и гладкой поверхностей

Для оценки освещенности поверхности в произвольной точке был выбран метод интегрирования Монте-Карло, который позволяет добиться высокой точности оценки, однако, ввиду своей стохастичности, на готовых изображениях формирует шум. Данный недостаток решается при помощи агрегации последовательности генерируемых кадров с течением времени.

Выбранный набор инструментов обеспечивает фотореалистичную, физически корректную отрисовку трехмерных сцен, описанных в одном из самых широко используемых форматов.

## Список использованных источников:

1. *GeForce RTX Founders Edition Graphics Cards: Cool and Quiet, and Factory Overclocked* [Электронный ресурс]. – Режим доступа: <https://www.nvidia.com/en-us/geforce/news/geforce-rtx-founders-graphics-card-breakdown/>
3. *Vulkan Ray Tracing Final Specification Release* [Электронный ресурс]. – Режим доступа: <https://www.khronos.org/blog/vulkan-ray-tracing-final-specification-release>
3. *Learn OpenGL, extensive tutorial resource for learning Modern OpenGL* [Электронный ресурс]. – Режим доступа: <https://learnopengl.com/PBR/Theory>

# ИСПОЛЬЗОВАНИЕ АРХИТЕКТУРНЫХ ОСОБЕННОСТЕЙ ЯДЕР ПРОЦЕССОРОВ В РАЗРАБОТКЕ МИКРОЯДЕРНОЙ ОПЕРАЦИОННОЙ СИСТЕМЫ РЕАЛЬНОГО ВРЕМЕНИ С МАКСИМАЛЬНОЙ ПРОИЗВОДИТЕЛЬНОСТЬЮ И МИНИМАЛЬНЫМ ВРЕМЕНЕМ ОТКЛИКА

*Морозов И.Ю.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Видничук В.Н. – м.т.н*

В докладе рассматривается тема оптимизации компонентов ядра операционной системы реального времени (ОСРВ) под различные архитектурные особенности современных процессоров для встраиваемых систем.

На протяжении как минимум последних 10 лет, закон Гордона Мура остается лишь частью истории компьютерных наук. И с каждым днем рост числа транзисторов на кристалле в относительном эквиваленте неумолимо падает, а с их количеством падает тактовая скорость процессоров – что в конечном итоге заставляет производителей процессоров улучшать архитектуру, а разработчиков программного обеспечения более тщательно подходить к особенностям используемой платформы и учитывать ее особенности.

Большинство современных ОСРВ разрабатываются в следующих направлениях: уменьшение размера ядра, максимальная переносимость, низкий порог входа в разработку, увеличение абстрагирования от железа. Однако, на сегодняшний день, учитывая огромное число кросс-компиляторов, и то, что память даже дешевых встраиваемых систем исчисляется сотнями мегабайт – данные направления развития операционных систем утрачивают свой смысл. Тем не менее, сложность и требования для современных устройств и систем растут и, несмотря на сотни мегабайт оперативной и flash памяти, сотни МГц тактовой частоты, общая производительность современных систем остается на месте, что позволяет нам наблюдать такие картины, как использование Cortex-A8 процессоров в мультимедиа, само собой безосновательно увеличивая их стоимость.

Перечислим основные технические параметры, которые должна обеспечивать разрабатываемая ОС реального времени:

- Время реакции системы на прерывание;
- Время реакции задачи (task response);
- Время переключения контекста (context switch);
- Задержка диспетчеризации (scheduling latency);
- Максимальное время исполнения каждого системного вызова;

Данная работа подразумевает несколько частей – исследование и систематизация влияния архитектурных особенностей процессора на ядра ОСРВ и непосредственно разработка ОСРВ на базе исследовательской части.

Исследование начинается с четкого определения метрик оценки производительности и скорости отклика. К показателям производительности можно отнести следующее:

- Сколько операций может выполнить система за определенный период времени;
- Сколько операций могут быть выполнены одновременно;
- Как долго система выполняет операцию;
- Сколько ошибок(пропусков) система генерирует под нагрузкой;

Неустойчивость таймера (timer jitter) - разница во времени срабатывания таймера с фиксированной длительностью. Однако не менее важным являются внешние параметры, такие как: температура, уровень ЭМП, качество и уровень питания. Данные показатели для разных систем могут сильно колебаться. Оптимальным уровнем температуры является диапазон 10-50 градусов по Цельсию. ПДУ ЭМП - 100 мкВт/см<sup>2</sup>. В качестве источника питания должен использоваться линейный источник, с пиковым током при 5В не менее 5А и уровнем пульсаций более 20мВ при 20 Ваттах потребляемой мощности.

С целью регулирования температуры нагрева многие современные процессорные системы имеют возможность использования дросселирования тактов. Во всех системах, где это возможно, данная функция отключается. Однако при тестах производительности используется активное охлаждение с отрицательной обратной связью по термистору. На рисунке 1 представлен пример оценки производительности чипа Broadcom BCM2711 в зависимости от температуры. Количество операций исчисляется тысячами. Программа оценки производительности написана на языках C и



ассемблер. Программа в дальнейшем будет использоваться для оценки производительности, так как полностью учитывает все метрики. Измерения производились лишь по контрольным точкам температур. На графике видно резкое падение (~20%) количество тактов при 60 градусах. Также наблюдается дальнейшее снижение производительности по мере увеличения температуры.

С использованием данного подхода был протестирован ряд популярных ОСРВ на разных популярных процессорах используемых во встраиваемых системах. Также были протестированы отдельные модули ядра операционных систем, с целью максимально точной оценки их производительности на конкретных аппаратных решениях.

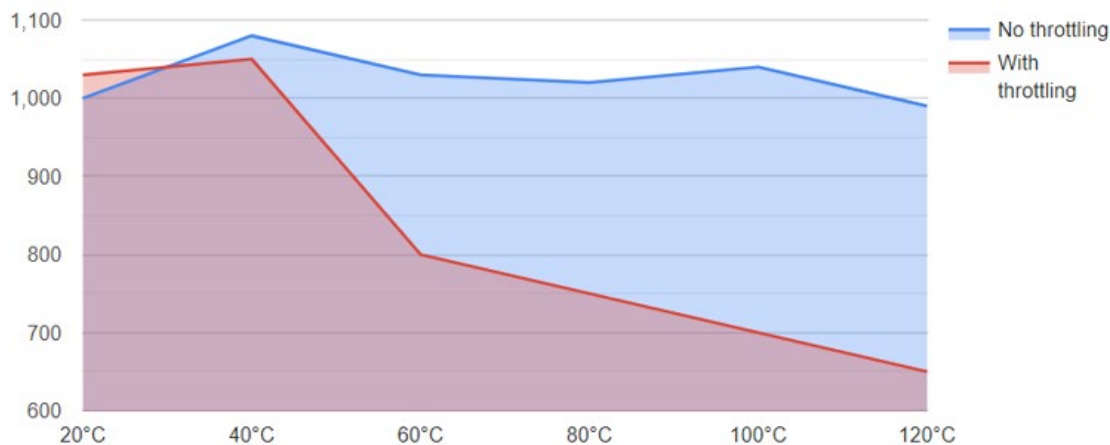


Рисунок 1 — Пример работы чипа Broadcom BCM2711 с тротлингем и без него

После проведения исследования, на основании результатов написана ОСРВ позволяющая гибко использовать ее отдельные компоненты под разные платформы. Само ядро, состоит из 4 основных компонентов:

- ITC(inter thread communication) – блок общения между потоками, в том числе и в режиме SMP;
- MMM(memory management module) – модуль управления оперативной и постоянной памятью;
- IOM(input-output module) – модуль ввода-вывода;
- TM(thread module) – модуль отвечающий за потоки системы;

Также следует отметить, что отдельно всеми компонентами управляет EM(execution module) – он отвечает за всю работу основных модулей. Также операционная система поддерживает обновление отдельных модулей ядра независимо от других за счет собственного бинарного формата. Также он позволяет динамически подгружать из загрузчика новые компоненты ядра. С целью верификации оригинальности сборки используется цифровая подпись, которая не позволит загрузчику прошить не оригинальную сборку.

В ОСРВ реализованы как динамическое, так и статическое планирование. Тип планирования выбирается при сборке ядра. Статические алгоритмы планирования используют приоритетное вытесняющее планирование. Приоритет присваивается каждой задаче до того, как она начала выполняться. Преимущество отдается задачам с самыми короткими периодами выполнения. В динамических алгоритмах планирования приоритет задачам присваивается динамически, причем предпочтение отдается задачам с наиболее ранним предельным временем начала выполнения. В системах реального времени на алгоритм планировки возлагается задача определения последовательности выполнения заданий в соответствии с их требованиями к ресурсам и ко времени исполнения.

Данная ОСРВ решает поставленные задачи при том оставаясь архитектурно гибкой и легко переносимой от платформы к платформе, позволяя разработчикам легко добавлять поддержку нового аппаратного средства. При компиляции ядра, разработчик может выбрать соответствующий набор модулей и компонентов который ему необходим, или выбрать рекомендуемую конфигурацию сформированную на основании предоставленной информации об аппаратной платформе в конфигурационном файле.

**Список использованных источников:**

1. Петин, В. Микрокомпьютеры Raspberry Pi. Практическое руководство/ В.А. Петин – БХВ-Петербург, 2017.-240с
2. Электронный справочник TECH Help! 6 by Flambeaux Software
3. Microsoft Extensible Firmware Initiative FAT32 File System Specification
4. DiskPart Documentation on docs.microsoft.com/en-us/windows-server/administration/windows-commands/diskpart

# ВЕБ-ПРИЛОЖЕНИЕ ДЛЯ ИНТЕРНЕТ-МАГАЗИНА ПО ПРОДАЖЕ ДЕКОРАТИВНЫХ ИЗДЕЛИЙ

Мархоцкий А.О.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Хмелева А.В. – доцент кафедры ПОИТ

В современном мире жизнь приобретает все большие темпы. Каждый хочет максимально выгодно распоряжаться своим драгоценным временем и не тратить его впустую. В век потребления человеку повсюду навязывают идею покупать все больше и больше товаров. На эти процессы бы уходило слишком много времени, которым каждый из нас дорожит, если бы не развитие бизнес-индустрии в интернете.

С каждым днем современные технологии становятся доступнее для каждого жителя планеты. С каждой минутой растет количество людей, которые используют интернет в своих целях. Еще 20 лет назад это число не превышало и 5%, а сегодня более 60% населения всей Земли имеют доступ к интернету. И было бы очень глупо не использовать этот огромный рынок для развития бизнеса, в частности для развития торговли.

Торговля в интернете предоставляет массу преимуществ как для покупателя, так и для продавца товаров. Ключевыми преимуществами для покупателя являются:

- а) низкие цены в интернет-магазинах;
- б) возможность доставки товара до дома.

Ключевые преимущества для продавца:

- а) снижение расходов на содержание торговых точек;
- б) возможность начать бизнес без большого стартового капитала;

Ввиду всех вышеперечисленных преимуществ торговли в интернете, подавляющее большинство предпринимателей не упускают возможность ведения своего бизнеса в интернете. Для этих целей создаются интернет-магазины на соответствующее тематики, в которых перечислен весь перечень имеющихся для продажи товаров.

Программное средство представляет из себя клиент-серверное приложение. Клиентская часть написана с помощью такой библиотеки как React.js [1]. Серверная часть приложения написана с помощью фреймворка express.js [2]. Приложение позволяет просматривать весь ассортимент магазина и оформлять заказ понравившихся товаров онлайн. Внешний вид клиентского приложения представлен на рисунке 1.

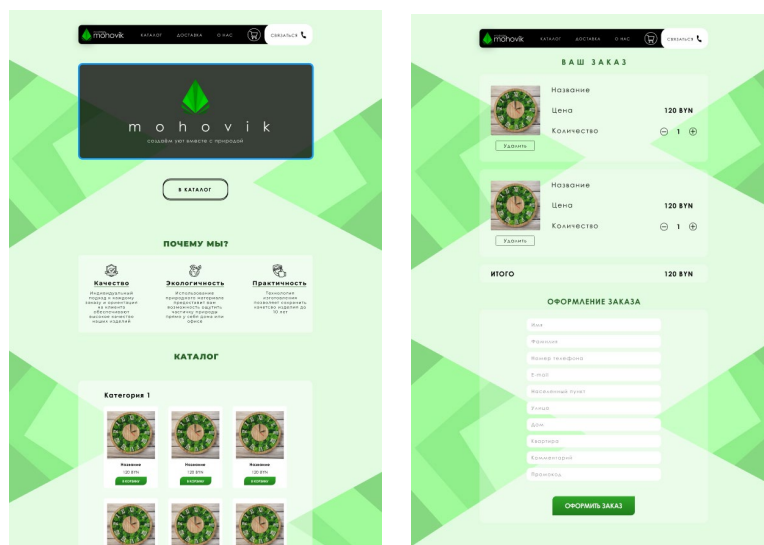


Рисунок 1 – Внешний вид клиентского приложения

Основными преимуществами веб-приложения являются простота в использовании, что достигается благодаря интуитивно-понятному пользовательскому интерфейсу, удобство и доступность, так как приложением можно воспользоваться из любой точки планеты, имея лишь смартфон.

#### Список использованных источников:

- [1] React [Электронный ресурс]. – Электронные данные. –Режим доступа: <https://reactjs.org/>  
[2] Express [Электронный ресурс]. – Электронные данные. –Режим доступа: <https://expressjs.com/>

# ВЕБ-ПРИЛОЖЕНИЕ ДЛЯ СИСТЕМЫ ЭЛЕКТРОННОГО УЧЕТА КНИГ НА БАЗЕ ПЛАТФОРМЫ .NET

Галай В.Э.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Хмелева А.В., к.т.н., доцент кафедры ПОИТ

Книги являются источником знаний для всего человечества уже на протяжении не одной сотни лет. Первые шумерские и египетские тексты датируются второй половиной IV тысячелетия до н. э. С увеличением числа книг возникла надобность в создании некоторого “хранилища” этих книг. Так появились первые библиотеки.

В первых библиотеках не существовало никакой системы учета книг ввиду того, что доступ к знаниям того времени имело лишь ограниченное число привилегированных лиц. Однако с течением времени все больше людей желало получить доступ к знаниям, что было обусловлено различными историческими процессами, в том числе и развитием общества как культурной единицы.

Система электронного учета книг представляет собой ни что иное, как некоторое программное средство, позволяющее регистрировать поступления новых книг в компанию (организацию), редактировать информацию об этих книгах и, при необходимости, удалять записи о тех книгах, которые больше не находятся в наличии у данной компании (организации). Также данное программное средство предназначено для ведения учета пользователей, которые зарегистрировались в системе, отслеживания их деятельности (какие книги взял пользователь, какие вернул и т.п.).

Целью данного исследования является разработка программного средства, которое будет выполнять все вышеперечисленные функции, что впоследствии приведет к оптимизации рабочего процесса компании (организации) и, как следствие, повышению имиджа этой компании (организации).

Системы электронного учета крайне разнообразны, поэтому их тяжело как-то классифицировать. Однако можно утверждать, что все системы электронного учета должны обладать следующими чертами:

- возможность добавления нового элемента в систему;
- возможность редактирования элемента системы;
- возможность удаления элемента из системы;
- возможность осуществления поиска элемента в системе.

Так, систему электронного учета электротехники с системой электронного учета книг объединяет возможность добавления, редактирования, удаления и поиска записей в этих системах.

Назначением дипломного проектирования является разработка системы электронного учета книг. На основании произведенного обзора существующих аналогов, выявленных преимуществ и недостатков данных средств, сделан вывод, что для решения поставленной цели необходимо выполнить следующие задачи:

- проектирование архитектуры программного средства;
- разработка алгоритмов поиска, добавления, удаления и редактирования книг;
- разработка пользовательского интерфейса;
- тестирование программного средства.

Программное средство “Система электронного учета книг” должно выполнять следующие основные функции:

- вывод информации о книге (название, имя автора, год издания, описание и т.д.);
- поиск книги по заданным критериям.

Результатом данного исследования будет являться программное средство, выполняющее вышеперечисленные задачи и отличительными чертами которого будут являться оптимизация и упрощенный, относительно аналогов, интерфейс.

## Список использованных источников:

[1] Дж. Рихтер. CLR via C#. Программирование на платформе Microsoft .NET Framework 4.5 на языке C#.

# ПРОГРАММНОЕ СРЕДСТВО НА ПЛАТФОРМЕ IOS, РЕАЛИЗУЮЩЕЕ ПОДБОР ОБЩЕСТВЕННЫХ МЕСТ НА ОСНОВЕ ГЕОЛОКАЦИИ

*Латфулин Р.Р.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Хмелева А.В. – доцент кафедры ПОИТ*

Мобильные телефоны играют важную роль в современном мире. Смартфон есть практически у каждого человека. Используя мобильный телефон, мы можем в любое время связаться с семьей, друзьями или коллегами, чтобы быстро узнать или передать информацию. Помимо контактов, в телефоне хранится много другой информации: заметки, номера кредитных карт, заметки календаря, фотографии, звукозаписи и другая личная информация. Множество полезных приложений, функций и опций, которыми оснащены современные мобильные телефоны, делают это устройство многофункциональным.

Рынок мобильных приложений растет как на дрожжах. Эта огромная отрасль расширяется с каждым днем. Мобильные приложения в свою очередь играют не менее важную роль в развитии современных информационных технологий посредством автоматизации многих процессов, связанных с упрощением рутинной деятельности человека.

Сфера туризма не стала исключением. Мобильная революция также задела и ее. Около 75% путешественников используют во время своей поездки как минимум одно приложение. Туристы особо ценят приложения, которые помогают ориентироваться на незнакомой местности. Подобные мобильные приложения предлагают туристам карты, навигацию по городу, гид по различным местам.

Сегодня у каждого есть картографический сервис, который можно интерпретировать как помощника туриста. Помимо карт, с датчиками GPS и ГЛОНАСС, определяющими ваше местоположение, существуют более продвинутые приложения, которые могут не только построить маршрут к определенному месту, но также предоставить краткое описание, поделиться фотографиями и просмотреть оценки других пользователей.

Целью исследования стала разработка мобильного приложения для подбора общественных мест на основе геолокации. Собираясь в очередное путешествие туристу проще скачать мобильное приложение нежели приобрести карту определенной местности. Однако при желании посетить кафе, или музей, или бар сперва придется узнать адрес. Здорово иметь в своем наборе приложений помощника, который подскажет ближайшие места, разбитые по категориям, поможет проложить маршрут к выбранному месту, укажет рабочее время, а также подскажет контакты администрации того или иного заведения.

Для реализации мобильного приложения используются следующие библиотеки:

– UIKit – библиотека, разработанная сотрудниками Apple, для реализации пользовательских интерфейсов, используя Storyboard, размещая на нем необходимые компоненты. Также UIKit содержит в себе всю необходимую базовую логику по обработке взаимодействия пользователя с мобильным приложением.

– SwiftLocation – сторонняя библиотека, представляющая собой обертку поверх представленной сотрудниками Apple библиотеки CoreLocation. SwiftLocation позволяет наиболее эффективно обрабатывать и получать текущее местоположение пользователя.

– Google Places – библиотека, разработанная компанией Google для работы с местами независимо от местоположения пользователя, т.к. Google places содержит в себе большую базу данных по тем или иным городам, позволяя получить необходимую информацию по месту.

Основными преимуществами мобильного приложения является простота в использовании, что достигается благодаря интуитивному интерфейсу. путешественник может ознакомиться с достопримечательностями города, используя мобильное приложение, которое предоставляет список тех или иных мест.

# ПРОГРАММНОЕ СРЕДСТВО УПРАВЛЕНИЯ КОРПОРАТИВНЫМ ТАРИФОМ СОТОВОЙ СВЯЗИ

Петрушко Е.А., студент гр.751001

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Куликов С.С. – доцент кафедры ПОИТ.

В данной работе описывается разработка программного средства управления корпоративным тарифом сотовой связи, а также рассматриваются его отличия от существующих программных средств.

Мобильная связь стала неотъемлемой частью жизни не только в повседневной жизни человека, но и в корпоративной среде. Сегодня на рынке сотовой связи существует большое количество различных тарифных планов, в том числе корпоративных. Основным преимуществом корпоративных предложений является большое количество полезных опций, которые индивидуально настраиваются под клиента: большие пакеты минут, безлимитный Интернет-трафик, SMS-сообщения или действие домашней тарификации во время путешествий по стране.

Существует множество приложений, позволяющих учитывать расходы, но все они являются настольными приложениями: привязаны к устройству, на которое установлено программное средство.

Клиент представляет собой интерфейс пользовательского взаимодействия, позволяющий производить учёт расходов. В дополнение к этому, пользователь может экспортировать расходы в форматы файлов xls и pdf, а также визуализировать расчёты с помощью графиков и диаграмм. Серверная часть обрабатывает запросы, поступающие со стороны клиента, а также взаимодействует с базой данных.

При разработке приложения используется следующий стек технологий, который предоставляет платформа SAP Business Technology Platform: SAP Cloud Platform, ABAP, SAPUI5, HANA[1]. SAP Cloud Platform – это облачный сервис, позволяющий размещать в себе базы данных, серверы и клиентов (рисунок 1). Язык четвёртого поколения ABAP является основным языком разработки приложений для платформы SAP. Для написания клиентской части использован фреймворк SapUI5, позволяющий делать запросы к серверной части и динамически отображать информацию в браузере. HANA - реляционная система управления базами данных, которая производит вычисления in-memory, и использует принцип поколочатого хранения данных.

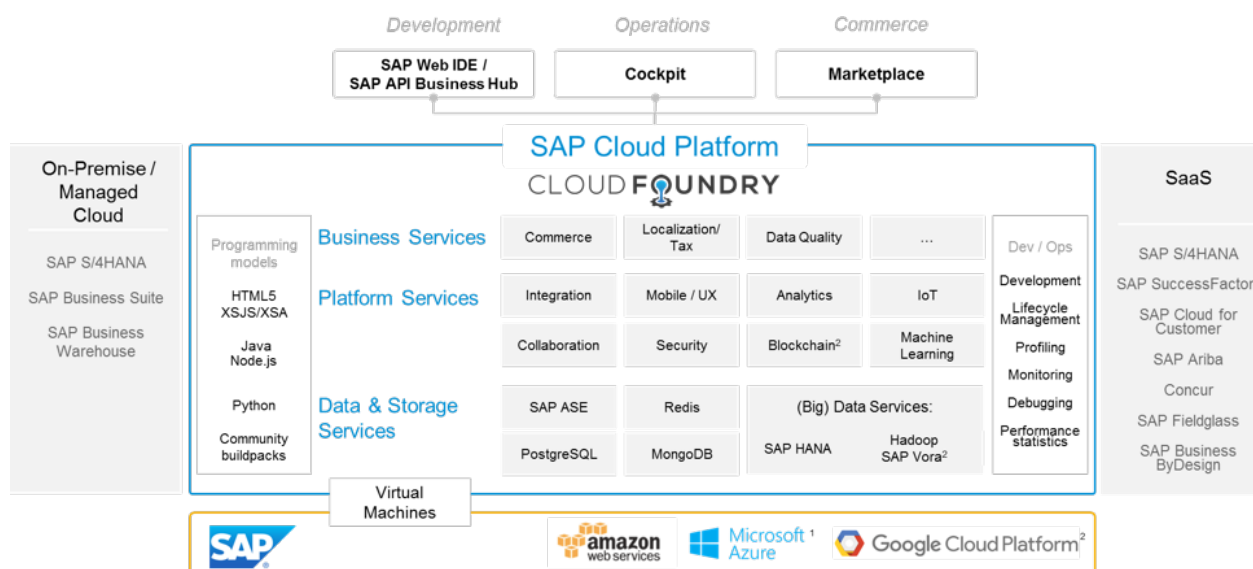


Рисунок 1 – модель архитектуры SAP Cloud Platform

Данный стек технологий позволяет достаточно быстро разработать рабочий прототип приложения. Разрабатываемое программное средство является веб-приложением, что выделяет его на фоне других программных средств учёта расходов, а также поможет ему набрать популярность среди компаний, которые используют корпоративные тарифы сотовой связи.

## Список использованных источников:

1. Портфель решений SAP Business Technology Platform [Электронный ресурс]. – Режим доступа: <https://www.sap.com/cis/products/business-technology-platform/products.html> — Дата доступа: 02.04.21.

# УПРАВЛЕНИЕ ПРОЕКТАМИ

*Пивоварчик Д.А.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Нестеренков С.Н. – к. т. н., доцент каф. ПОИТ*

Все проекты разные. Не существует идеальной системы управления проектами, подходящей для каждого из видов проектов. Также не существует системы, которая бы подходила каждому руководителю и была удобна для всех членов команды. Однако за время существования проектного управления было создано немало эффективных подходов, методик, стандартов и программных средств.

Для рассмотрения способов управления проектами необходимо дать определение проекту. Проект – это определенный процесс для достижения определённых целей и решения конкретной бизнес-задачи.

Следовательно, управление проектами — это деятельность, направленная на достижение поставленных задач, реализацию определённых планов, используя имеющиеся ресурсы - время, капитал, людей.

В основе управления проектами лежит планирование – краткосрочное или на более длительный период. В бизнес-процессах планирование основывается на определённых методиках планирования: в зависимости от приоритета задач и сроков их выполнения.

Управление проектами – это и есть решение ряда небольших отдельных задач на разных этапах проекта. Путем решения более мелких действий можно приближаться к поставленной цели.

То есть, управление проектами – это постоянный переход от простого к сложному, и трансформация одной большой задачи в более простые мероприятия, состоящие из шаблонных процедур. Главное – это закрепить отдельного исполнителя для решения каждой небольшой задачи, который должен выполнить это отдельное действие за конкретный промежуток времени.

Итого, можно выделить ряд определённых признаков проекта, которые отличают его от других видов деятельности:

1. Любой проект направлен на достижение конкретных целей.
2. Проект включает в себя координированное выполнение взаимосвязанных действий.
3. Проект имеет ограниченную протяженность во времени, с определенным началом и концом.
4. Каждый проект в определенной степени неповторим и уникален.

Основной особенностью проектов является то, что проекты – это мероприятия однократные и все задачи в них значительно отличаются друг от друга. Тем самым каждый проект требует своего уникального подхода и, как правило, различных методик, стандартов и программных средств управления.

Для каждого проекта предусмотрено более или менее чётко обозначенные начало и конец.

Проект начинается, когда сформируется команда в необходимом составе и решится вопрос с финансированием, и заканчивается, когда достигнуты его основные цели.

Одним из основных факторов в управлении проектами является время, поэтому для обеспечения выполнения всех работ в срок принято составлять графики, показывающие время начала и окончания заданий. Для этого используются различные программные средства для управления проектами.

Во время разработки проекта любая задача в нем выполняется однократно, а не циклически. Что усложняет планирование и контроль за выполнением задач. Для упрощения контроля и систематизации выполнения используются различные программные средства для управления проектами:

1. Jira представляющая собой веб приложение для отслеживания ошибок, но и также подходящее для управления проектами.
2. Битрикс 24 представляющий собой сервис для управления проектами, включающий в себя менеджер задач и корпоративную социальную сеть.
3. Asana мобильное и веб приложение для управления проектами компаний в больших и средних командах.

Таким образом управление проектами это сложный и длительный процесс требующий разнообразных и постоянно изменяющихся ресурсов.

## **Список использованных источников:**

1. Проектные сервисы [Электронный ресурс] – Режим доступа: <https://www.pmservices.ru/project-management-news/top-7-metodov-upravleniya-proektami-agile-scrum-kanban-prince2-i-drugie>.
2. Управление проектами [Электронный ресурс] – Режим доступа: [https://www.e-executive.ru/wiki/index.php/%D0%A3%D0%BF%D1%80%D0%B0%D0%B2%D0%BB%D0%B5%D0%BD%D0%B8%D0%B5\\_%D0%BF%D1%80%D0%BE%D0%B5%D0%BA%D1%82%D0%B0%D0%BC%D0%B8](https://www.e-executive.ru/wiki/index.php/%D0%A3%D0%BF%D1%80%D0%B0%D0%B2%D0%BB%D0%B5%D0%BD%D0%B8%D0%B5_%D0%BF%D1%80%D0%BE%D0%B5%D0%BA%D1%82%D0%B0%D0%BC%D0%B8).
3. Статья «Управление проектами 2.0» [Электронный ресурс] – Режим доступа: <https://www.ipm.by/programs/management/project>.

# ПРОГРАММНОЕ СРЕДСТВО АНАЛИЗА ПОВЕДЕНИЯ ПОЛЬЗОВАТЕЛЯ ПРИ РАБОТЕ С КЛАВИАТУРОЙ НА ОСНОВЕ ТЕХНОЛОГИИ DART

Тютюрин Е.Е.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Деменковец Д.В. – доцент кафедры ПОИТ

Меньше чем за месяц миллионы людей по всему миру перешли на удаленную работу из-за пандемии COVID-19. Перевод персонала на работу из дома – это только первый шаг. Второй – повысить эффективность в новой рабочей среде.

Для лидера, менеджера первостепенное значение имеет знание того, сколько времени тратится. Благодаря этому вы можете увидеть, зарабатываете ли вы деньги или сжигаете.

Контроль времени помогает не только компаниям, но и фрилансерам. С помощью приложений учета рабочего времени обнаруживается время, потерянное во время работы. Например, какие сайты и приложения используются чаще всего и сколько требуется, с общей оценкой производительности. Таким образом, вы можете сравнивать рабочие дни по неделям и оптимизировать свой график.

Еще поднимается вопрос про push-уведомления. Они стали настоящим трендом за последние годы - они доступны, заметны и быстры. Согласно исследованиям, более 44% пользователей соглашаются на получение уведомлений. Но во время работы они могут отвлекать, и часто можно забыть отключить уведомления вручную. Поэтому в программном средстве есть нейронная сеть, которая отслеживает, когда человек работает, а когда нет, чтобы блокировать push-уведомления.

Программное средство представляет из себя клиент-серверное приложение. Клиентская часть написана с помощью SDK Flutter [1]. Серверная часть приложения написана с помощью .NET [2]. Приложение позволяет просматривать весь ассортимент магазина и оформлять заказ понравившихся товаров онлайн. Внешний вид клиентского приложения представлен на рисунке 1.

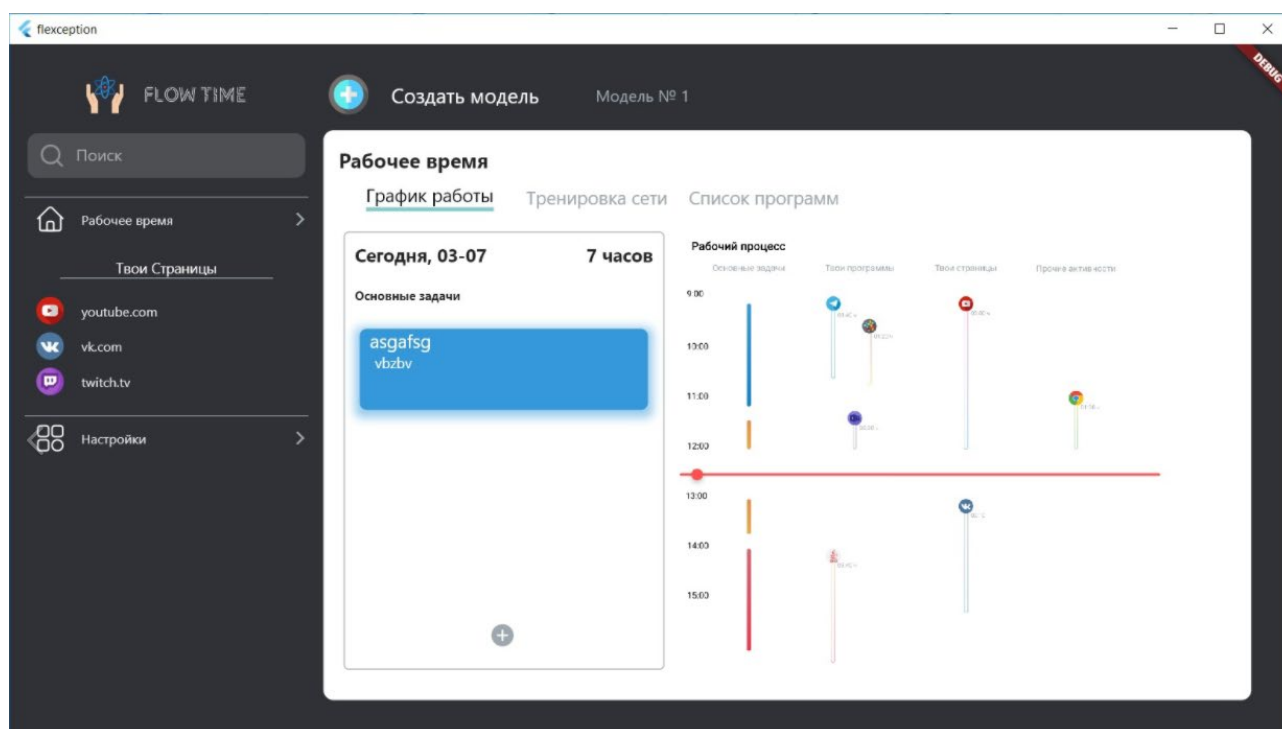


Рисунок 1 – Внешний вид клиентского приложения

Основными преимуществами приложения являются простота в использовании, что достигается благодаря интуитивно-понятному пользовательскому интерфейсу.

#### Список использованных источников:

[1] Flutter [Электронный ресурс]. – Электронные данные. –Режим доступа: <https://flutter.dev>

[2] .NET [Электронный ресурс]. – Электронные данные. –Режим доступа: <https://dotnet.microsoft.com>

# ПРОГРАММНОЕ СРЕДСТВО КОНТРОЛЯ КАЧЕСТВА ВОЗДУХА НА ОСНОВЕ ASP.NET CORE И МИКРОКОНТРОЛЛЕРА STM32

*Мухлядо Д.Ю., Деменковец Д.В.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Деменковец Д.В. – ст. преподаватель каф. ПОИТ*

Данный тезис описывает проблему загрязнения воздуха и его влияние на здоровье, самочувствие и работоспособность человека, также он демонстрирует устройство и программное средство, разработанное для мониторинга качества воздуха.

В связи с развитием высоких технологий в различных отраслях человеческой деятельности все более возрастают требования к стабильно обеспеченному качеству воздушной среды помещения, в котором осуществляется эта деятельность.

Качество воздуха в зданиях порой хуже, чем качество наружного воздуха. Проблемы с качеством воздуха в помещении, которые, как считается, связаны с загрязняющими веществами, сконцентрированными внутри зданий, могут вызывать недомогания и болезни среди людей, которые проводят время там. Если учесть количество работников в офисах, банках, школьных зданиях, ресторанах, гостиницах, больницах, неудивительно, что качество воздуха в помещении влияет на здоровье и производительность большинства работающих или проживающих в помещении.

Загрязнение воздуха может привести к неблагоприятным последствиям для здоровья. В развивающихся странах основным источником загрязнения воздуха внутри помещений является дым биомассы, содержащий взвешенные твердые частицы, двуокись азота (NO<sub>2</sub>), двуокись серы (SO<sub>2</sub>), окись углерода (CO), формальдегид и полициклические ароматические углеводороды.

Регулярные измерения качества воздуха в помещении и уровня комфорта на рабочем месте позволяют правильно настроить микроклимат и снизить риск заболеваний. Кроме того, эти измерения способствуют повышению продуктивности работы персонала.

Чтобы замерить качество воздуха в помещении, используются специальные приборы, которые называются – дата-логгеры.

Дата-логгер – это электронное устройство, которое записывает во внутреннюю память, на внешнее хранилище или передает в облачный сервис данные. Данные сохраняются с течением времени или по отношению к местоположению. Данные могут поступать от встроенного в прибор сенсора или датчика или от внешних приборов и датчиков.

Все больше и больше, но не полностью, регистраторы данных создаются на базе цифровых процессоров (или компьютеров). Они, как правило, небольшие, на батарейках, портативные, и снабжены микропроцессором, внутренней памятью для хранения данных и различными встроенными датчиками. Некоторые регистраторы данных обладают специальным интерфейсом для подключения к персональному компьютеру, а также используют программное обеспечение, чтобы активировать регистратор данных, просматривать и анализировать собранные данные. В то же время некоторые устройства могут иметь локальный интерфейс (клавиатура, дисплей) и могут быть использованы в качестве автономного устройства.

Регистраторы данных варьируются от изделий общего назначения для измерительных применений до очень специфических приборов для измерения в одной среде, или только одного параметра. Возможность программирования является общей особенностью для изделий общего назначения; в то время как многие другие дата-логгеры остаются как статическими машинами с ограниченным количеством или полным отсутствием сменных параметров. Электронные регистраторы данных заменили Диаграммные самописцы во многих областях.

Одним из основных преимуществ использования дата-логгеров является возможность автоматически собирать данные на 24-часовой основе. После активации регистраторы данных обычно продолжают сбор данных, оставленные без присмотра, для измерения и записи информации в течение периода мониторинга. Это позволяет обеспечить всестороннее, точное представление об условиях окружающей среды или наблюдаемых параметрах технологического процесса, таких как температура воздуха, относительная влажность или расход, давление, вибрация, ударная нагрузка и пр.

Термины регистрация данных и сбор данных часто используются как синонимы. Однако, в историческом контексте они совершенно разные. Регистратор данных – это система сбора данных, но система сбора данных не обязательно является регистратором.

1. Регистраторы данных, как правило, имеют более медленные частоты дискретизации. Максимальной частотой дискретизации 1 Гц можно считать очень быстро для регистратора, но очень медленно для типичной системы сбора данных.

2. Регистраторы данных как правило – это автономные устройства, в то время как типичная система сбора данных должна оставаться привязанным к компьютеру для получения данных. Автономность регистратора данных подразумевает наличие встроенной памяти, которая используется для хранения полученных данных. Иногда объем этой памяти очень большой, чтобы вместить много



дней, или даже месяцев, автоматической записи. Эта память может быть с резервированием от аккумуляторной батареи, статической оперативной памятью, флэш-памятью или ЭСППЗУ.

3. С учетом длительного времени записи данных, дата-логгеры, как правило, оснащены механизмом для записи даты и времени в метку времени, чтобы гарантировать, что каждая запись значения данных, связана с датой и временем получения для того, чтобы восстановить последовательность событий. Таким образом, данные регистраторы обычно используют встроенные часы реального времени, заявленный дрейф которых может быть важным фактором при выборе между регистраторами данных.

4. Портативный Регистратор может достигать до 20 каналов с максимальной 10мс (100Гц) частотой дискретизации.

5. Автоматический и дистанционный характер многих регистраторов данных подразумевает необходимость в некоторых применениях для работы от постоянного источника питания, например от батареи. Солнечная энергия может быть использована для пополнения этих источников энергии. Все эти факторы в целом привели к тому, что устройства, которые появляются на рынке являются крайне энергоэффективным по отношению к компьютеру.

6. Автономность регистраторов диктует условия их предельной надежности. Поскольку они могут работать длительное время без остановок практически без участия человека, и могут быть установлены в сложных или отдаленных местностях, важно, что пока они имеют питание, они не прекратят сбор данных по какой-либо причине. Такие регистраторы практически полностью застрахованы от проблем (таких как зависание программы и нестабильность некоторых операционных систем), которые могут повлиять на ЭВМ общего назначения.

Программное средство для контроля качества воздуха должно состоять из трех частей: микроконтроллерной, серверной и клиентской.

#### 1. Микроконтроллерная часть

Программное обеспечение для микроконтроллера должно выполнять следующие функции:

- считывание информации с датчика;
- взаимодействие с внешним носителем;
- считывание настроек;
- запись показаний датчика;
- взаимодействие с сервером по сети WiFi:
  - а) передача показаний датчика;
  - б) синхронизация времени;
  - в) опрос сервера на внеочередное считывание показаний датчика.

#### 2. Серверная часть

Программное обеспечение для сервера должно выполнять следующие функции:

- взаимодействие с микроконтроллерами:
  - а) авторизация микроконтроллера;
  - б) запись полученных данных с сенсоров в базу данных;
  - в) синхронизация времени;
- авторизация пользователей;
- регистрация пользователей;
- регистрация микроконтроллеров;
- обновление данных микроконтроллеров и пользователей;
- генерация файлов конфигурации для микроконтроллеров;
- загрузка ПО для микроконтроллеров;
- обработка файлов показаний датчика, записанных на внешний носитель;
- обработка данных датчиков.

#### 3. Клиентская часть

Программное обеспечение для клиента должно выполнять следующие функции:

- визуализировать функции, предоставляемые сервером;
- строить тренды по показаниям датчиков;
- отображать интерактивную карту, показывающую положение зарегистрированных микроконтроллеров;
- перенаправлять на страницы регистрации и авторизации, поддерживаемых сторонних сервисов.

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

1. Wikipedia [Электронный ресурс] – Режим доступа: <https://www.wikipedia.ru>.
2. Статья «Простая и эффективная система контроля окружающей среды: создаем с нуля» [Электронный ресурс] – Режим доступа: <https://www.compel.ru/lib/85493>.
3. testo [Электронный ресурс] – Режим доступа: <https://www.testo.ru/ru-RU/testo-160-iaq/p/0572-2014>.
4. ВОЗ [Электронный ресурс] – Режим доступа: <https://www.who.int/ru>.

# ПРЕДСТАВЛЕНИЕ СИГНАЛОВ В МОДОВОЙ ДЕКОМПОЗИЦИИ

Бондарева Т.О., Деменковец Д.В., Леванцевич В.А.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Бранцевич П.Ю. – канд. тех. наук

Описан метод представления сигналов в виде их модовой декомпозиции. В качестве основных информативных признаков метода выбираются амплитуда, мгновенная частота и фаза отдельных мод. Разработан алгоритм и реализовано программное средство модовой декомпозиции. Показана применимость метода для анализа нестационарных вибрационных сигналов, полученных с реальных технических объектов.

Одним из важнейших показателей оценки технического состояния машин и агрегатов являются параметры вибрации [1]. При этом для исследования структуры вибросигнала, как правило, используются методы, основанные на преобразовании Фурье. Необходимым условием для применения Фурье-анализа является требование стационарности исследуемых сигналов, что на практике не всегда реализуемо, так как большинство вибросигналов, получаемых с реальных объектов, являются нестационарными [2].

В настоящее время широкое распространение получил метод исследования нестационарных сигналов, основанный на эмпирической модовой декомпозиции, в результате которой исходный сигнал разлагается на ряд компонент, которые называются эмпирическими модами [3].

Эмпирическая мода - это такая функция, в которой количество экстремумов функции (максимумов и минимумов) и количество пересечений нуля не должны отличаться более чем на единицу.

Схема-алгоритма представления сигналов в модовой декомпозиции приведена на рис.1.

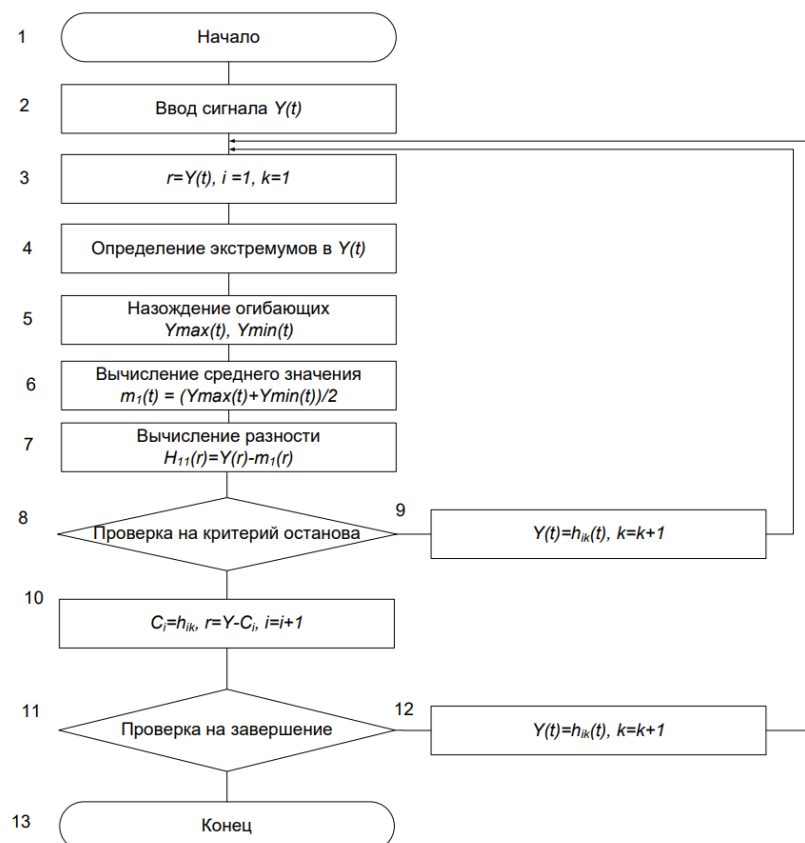


Рисунок 1 – Схема-алгоритма представления сигналов в модовой декомпозиции

Декомпозиция на модовые функции представляет собой итерационную процедуру. После открытия файла дискретных значений оцифрованного вибросигнала  $Y(t)$  в блоке 4 определяются все его максимальные и минимальные значения. В блоке 5 по полученным минимальным и максимальным значениям строится верхняя и нижняя огибающие  $Y_{max}(t)$  и  $Y_{min}(t)$ . Для построения огибающей, на

интервале между двумя экстремумами как правило, применяется интерполяция кубическими сплайнами.

В блоке 6 вычисляется среднее значение огибающих. Первое приближение  $h_{11}$  к первой функции эмпирической моды сигнала  $Y(t)$  определяется в блоке 7.

Принимая вместо функции  $Y(t)$  функцию  $h_{11}$  (блок 9) и повторяя шаги с первого по четвертый, находим второе приближение к первой функции моды  $h_{12}(t)$ . Аналогичным образом находим и следующие приближения  $h_{1k}(t)$  функции моды  $C_1(t)$ . Итерации продолжаются до тех пор, пока нормализованная квадратичная разность между двумя приближениями  $h_{1k}(t)$  и  $h_{1k-1}(t)$  Вычисляемая в блоке 8, не станет меньше некоторого предельного значения:

$$\alpha \geq \sum_{i=0}^T \left( \frac{(h_{ik-1}(t) - h_{ik}(t))^2}{h_{i-1}^2(t)} \right)$$

Последнее значение  $h_{ik}(t)$  итераций принимается за первую, наиболее высокочастотную моду  $C_1(k) = h_{ik}(k)$  семейства модовых функций, которая непосредственно входит в состав исходного сигнала  $Y(t)$ . Это позволяет вычистить  $C_1(k)$  из состава сигнала и оставить в нем более низкочастотные составляющие (блок 10).

Для получения второй эмпирической моды  $C_2(t)$  в блоке 12 функции  $Y(t)$  присваивается значение остатка  $r_1(t)$  и далее повторяются те же преобразования, что и для получения первой моды  $C_1(k)$ .

Просуммировав полученные значения мод  $C_i(t)$  и последний полученный остаток  $r_n$ , который может быть трендом или постоянной составляющей сигнала, получим разложение сигнала согласно формулы .

$$y(t) = \sum_{i=1}^n c_i(t) + r_n$$

Процесс декомпозиции может быть остановлен, когда остаток  $r_n(t)$  не содержит точек экстремумов, т.е. становится либо константой, либо монотонной функцией, из которой больше не могут быть извлечены модовые функции.

На рисунке 2 приведен пример декомпозиции сигнала, содержащего кратковременные возмущения, на две моды. Можно заметить, что первая мода выделяет сигнал возмущений из исходного сигнала. Определив амплитуду и частотный спектр первой моды, можно оценить характеристики сигнала возмущений.

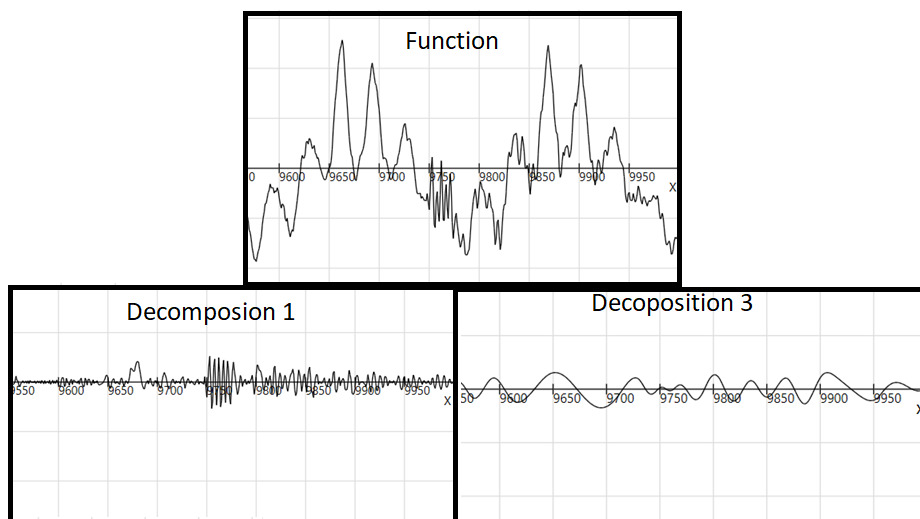


Рисунок 2 – Представление кратковременного возмущения сигнала в модовой декомпозиции

Выводы: Модовая декомпозиция сигналов может быть использована для анализа нестационарных вибрационных сигналов при оценке технического состояния машин и агрегатов.

1. Балицкий, Ф.Я. *Не разрушающий контроль. Справочник. Том 7. Книга Вибродиагностика / Ф.Я. Балицкий, А.В. Барков, Н.А. Баркова // М.: Машиностроение, 2005. – 829с.*

2. Кан, Ш.Ч. *Анализ нестационарных сигналов на основе преобразования Гильберта-Хуанга / Ш.Ч. Кан, А.В. Микулович, В.И. Микулович // Информатика – 2010. – №2 – С. 36-47.*

3. *The empirical mode decomposition method and the Hilbert spectrum for non-stationary timeseries analysis / N.E. Huang [et al.] // Proc. R. Soc. Lond. – 1998. – A454. – P. 903–995*

# МОБИЛЬНОЕ ПРОГРАММНОЕ СРЕДСТВО УПРАВЛЕНИЯ ПРЕДМЕТАМИ КОЛЛЕКЦИИ НА ОСНОВЕ РАДИОЧАСТОТНЫХ МЕТОК

*Голубев К.А., Фадеева Е.Е.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Деменковец Д.В. – ст. преподаватель, м.т.н.*

В результате работы было разработано мобильное программное средство управления предметами коллекции на основе радиочастотных меток. Мобильное приложение взаимодействует с радиочастотными метками с помощью технологии NFC. Данная система предназначена для того, чтобы пользователь имел возможность сканировать и записывать информацию на метки, поддерживающие стандарт NTAG 215. В качестве примера рассматривается учёт коллекционных моделей Amiibo, поставляющихся со встроенным RFID чипом.

Коллекционирование является феноменом социокультурного плана. Это процесс самовыражения человеческой личности, способ самореализации индивидуальности в свободное время. Коллекционеры образуют своеобразные социальные сети общения и формируют самобытные субкультуры по интересам. Такая деятельность наполнена социальным социокультурным смыслом [1]. В условиях массового производства и избытка информации, препятствующих концентрации, современный человек испытывает всё большую «потребность углубиться во что-то», что способствует бурному развитию коллекционирования как коммерческой индустрии в быстроразвивающемся современном мире [2].

В настоящее время среди коллекционеров существует потребность в новых способах управления предметами непосредственного коллекционирования: старые технологии ведения ручного каталога не отвечают необходимым критериям безопасности, удобства и доверия независимо от того, ведутся они на бумаге или с помощью компьютера. Наличие предмета в таком каталоге не гарантирует его наличие у коллекционера и потому во многих ситуациях может возникнуть необходимость перепроверок. Коллекционер ценит возможность презентовать свою коллекцию обществу. Это возможность оценить свой труд, узнать общественное мнение и расширить свои познания в поиске новых экземпляров, что усложняется устаревшими технологиями учёта коллекции [3].

С этой задачей может справиться технология внедрения в предметы радиочастотных меток и мобильное программное средство, которое будет сканировать и обрабатывать данные с них. К существующим аналогам данного метода перевода коллекционирования в цифровое поле можно отнести QR-коды и генерацию цифровой электронной подписи. Первый способ не решает проблемы с безопасностью и гарантией, а второй – удобства.

На данный момент на рынке существует серия коллекционных моделей Amiibo от компании Nintendo со встроенными радиочастотными метками стандарта NFC NTAG 215 [4]. В эти модели они встроены для взаимодействия с видеоиграми компании, однако на примере этой серии было принято решение продемонстрировать возможности рассматриваемой технологии управления коллекцией [5].

Мобильное программное средство представляет собой кроссплатформенное приложение. На стартовом экране имеются четыре вкладки: «Моя коллекция» - отсортированный список имеющихся у пользователя моделей; «Библиотека» - структурированный список всех существующих моделей по сериям выпуска, году, тематике и т.д. с индикатором полноты собранной коллекции; «Поиск» - меню поиска информации про интересующие модели; «Настройки» - меню настроек и информации о приложении. При нажатии пользователем кнопки «Сканировать» в разделе «Моя коллекция» на экране появляется подсказка с изображением и описанием того, как необходимо расположить мобильное устройство и коллекционную модель для сканирования радиочастотной метки, чтобы произошла синхронизация и чтение информации. После сканирования информации и модели, на экране мобильного приложения отображается следующая информация: визуализация отсканированной модели, серия и год выпуска, дополнительная информация о конкретном, возможность добавить модель в свою коллекцию. В случае, если эта модель уже была добавлена в коллекцию, пользователю предлагается на выбор либо обновить информацию о ней в соответствии с возможными изменениями, либо прервать процесс добавления модели. Приложение позволяет не только считывать, но и записывать информацию в пустые метки, тем самым упрощает возможность использования технологии для коллекционеров менее технологическими предметами. Для этого необходимо выбрать интересующую модель в библиотеке и проследовать инструкции по записи. Схема взаимодействия коллекционной модели и разработанного приложения представлена на рисунке 1.

Данное программное средство разработано на основе технологии Flutter и ориентировано для персонального использования на личном мобильном устройстве. Выбор технологии обусловлен задачей разработать приложение в минимальные сроки и на максимально широкий список устройств, а Flutter позволяет создавать современные мобильные приложения сразу на несколько платформ и независимо от версии операционной системы.



Рисунок 1 – Схема взаимодействия модели Amiibo и мобильного программного средства

Flutter – бесплатный и открытый набор средств разработки мобильного пользовательского интерфейса, созданный компанией Google и выпущенный в мае 2017 года. Проще говоря, с помощью Flutter возможно создать собственное мобильное приложение с одним массивом кода. Это означает, что для создания двух приложений (iOS и Android) можно использовать единый язык программирования и одну базу кода. Логотип платформы Flutter приведён на рисунке 2.



Рисунок 2 – Логотип платформы Flutter

К преимуществам технологии внедрения радиочастотных меток и использования мобильного программного средства можно отнести:

1. Доступность (пользователь может использовать своё личное мобильное устройство, которое всегда под рукой);
2. Безопасность (использование технологии NFC гарантирует подлинность предмета и требует непосредственного контакта для синхронизации);
3. Низкая стоимость (внедрение радиочастотной метки не создаёт высоких издержек за счёт низкой цены устройства);
4. Компактность (внедрение радиочастотной метки не создаёт высоких издержек за счёт низкой цены устройства).

Недостатками технологии и мобильного программного средства являются:

1. Особые требования к предметам коллекционирования (должны содержать в себе радиочастотные метки или быть дополнительно оснащены ими);
2. Удобство в использовании (технология NFC требует очень близкого контакта между устройством и меткой);
3. Распространённость (для повсеместного использования выбранной технологии от производителей коллекционных предметов требуется внедрять в свои продукты специальные радиочастотные устройства).

Разработанное мобильное программное средство удобно использовать для управления коллекцией моделей Amiibo, но пока производители других коллекционных предметов не разработают общий стандарт использования данной технологии в своих продуктах, широкое использования рассматриваемой технологии остаётся нецелесообразным.

Таким образом, главная задача – создание программного средства, упрощающего учёт коллекционных предметов и позволяющего пользователю иметь доступ к виртуальному журналу своей коллекции – была выполнена.

**Список использованных источников:**

1. Шахова И. А. Коллекционирование как социокультурный феномен современного общества / Вестник Амурского государственного университета – 2017. – Серия: Гуманитарные науки. – №78.
2. Хёйзинга Й. Тени завтрашнего дня / СПб – 2017 – С. 454.
3. Digitizing Collections [Электронный ресурс]. – Режим доступа: <https://siarchives.si.edu/what-we-do/digital-curation/digitizing-collections>. – Дата доступа: 03.04.2021.
4. Что такое Amiibo? [Электронный ресурс]. – Режим доступа: <https://www.nintendo.ru/amiibo/-amiibo-amiibo-932316.html>. – Дата доступа: 03.04.2021.
5. ISO/IEC 18092:2004 Information technology -- Telecommunications and information exchange between systems – Near Field Communication [Электронный ресурс]. – Режим доступа: <https://www.iso.org/standard/38578.html>. – Дата доступа: 03.04.2021.

# РАЗРАБОТКА ПРИЛОЖЕНИЯ УЧЕТА ФИНАНСОВ

*Пивовар С.Л.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Неборский С.Н. – канд. тех. наук, доцент*

Целью проекта является создание программного продукта для мобильной платформы Android, предназначенного для автоматизации учета финансов.

Актуальность темы обусловлена тем, что подавляющее большинство граждан большинства стран мира небрежно обращаются со своими доходами и не ведут учет своим финансовым средствам. Данная категория приложений способна решить некоторые проблемы пользователей, такие как:

- 1) Упрощение учета расходов
- 2) Просмотр истории финансовых операций
- 3) Просмотр статистики и отчетов за определенный период
- 4) Прогнозы бюджета пользователя через определенный период
- 5) Расчет депозитов и кредитов
- 6) Подсчет долгов

Некоторые мобильные приложения учета финансов способны по фотографии чека считать информацию о покупке и занести ее в свою базу. Некоторые приложения интегрированы с банковскими системами или банковскими картами, что позволяет напрямую получать информацию о тратах и доходах пользователя без его непосредственного участия. Автоматизация в данной категории приложениях направлена на упрощение ввода входных данных о пользовательских финансовых операциях, по которым сам пользователь впоследствии сможет получить отчет или наглядную статистику его расходов по разным категориям трат и доходов. Отдельные приложения имеют функцию планировщика или «умного помощника», который планирует финансовые траты пользователя так, чтобы по окончании определенного периода пользователь получил ожидаемый финансовый результат.

Данная категория приложений появилась для воспитания в их пользователях финансовой грамотности. Ведь учет долгов требует дисциплинированности личности, и учит распоряжаться собственными средствами так, чтобы получать от них максимальное благо. Ведь любая сфера жизни человека прямо или косвенно связана с финансами, которые подлежат тщательному счету и которыми следует распоряжаться с умом. Именно для этих целей будет разработано данное приложение.

Для эффективного управления процессом учета финансовых средств конечных пользователей предварительно проанализированы аналоги данного приложения, размещенные в сети Интернет и в крупных магазинах мобильных приложений, таких как Google Play Market [1]. Так же изучены предложения авторитетных компаний и организаций, которые своими акциями, мероприятиями и продуктами с интеграцией данной категории приложений продвигают рынок средств учета финансов. Была изучена предметная область, спроектирована архитектура приложения и структура базы данных. Были выделены основные моменты автоматизации действий пользователей по учету за финансовыми средствами.

Для разработки проекта использованы язык программирования Python для создания приложения. Для хранения данных мобильного приложения использована система управления базами данных SQLite3 [2] от создателя Ричарда Хиппа [3].

Данные технологии позволили оперативно разработать мобильное приложение, отвечающее требованиям стабильности и расширяемости. Мобильное приложение позволяет пользователям быстро и легко вносить новую информацию о своих финансовых операциях, просматривать историю произведенных финансовых операций, а также получать статистику по доходам и расходам в определенном временном периоде.

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

1. Google Play Market [Электронные данные] - Режим доступа: <https://play.google.com/> - Дата доступа: 19.04.2021.
2. SQLite3 Official [Электронные данные] - Режим доступа: <https://www.sqlite.org/> - Дата доступа: 19.04.2021.
3. D. Richard Hipp about, Antonz.ru [Электронные данные] - Режим доступа: <https://antonz.ru/richard-hipp/> - Дата доступа: 19.04.2021.

# ОНЛАЙН-СЕРВИС ДЛЯ СОВМЕСТНОЙ АРЕНДЫ ЖИЛЬЯ

*Карчмит В.С.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Болтак С.В. – ассистент кафедры ПОИТ*

Применение современных информационных технологий в бизнесе формирует новые направления в глобальной системе координат мировой экономики. Это направление сегодня принято называть интернет-экономикой. Интернет дает возможность отечественным компаниям выйти на мировой рынок, расширяет каналы сбыта, объединяет поставщиков и покупателей в единую систему.

За последний год темпы внемагазинной торговли, в особенности, электронной коммерции, возросли в несколько раз. Торговые площади, рынки, в данный момент сменяются интернет-пространством.

Главное назначение сайта заключается в том, чтобы рассказать пользователям о данной компании. В Интернете существуют разные способы информирования, но, как правило, все они, сводятся к одному – к веб-сайту.

Веб-сайт является совокупностью электронных документов, объединенных одним адресом. С ростом технологий веб-сайты становятся не просто совокупностью документов, но и представлением ряда технологий, которые используются как для привлечения внимания, так и для добавления различных функций, позволяющих делать пребывание пользователя на сайте не только увлекательным, но и удобным.

Уникальный контент сайта, грамотно разбавленный ключевыми словами, а также привлекательный дизайн страниц способны принести любой компании ощутимую экономическую прибыль. В этом и заключается актуальность темы дипломного проекта «Веб-сервис для аренды жилья»

Выбор программных средств для построения веб-сервисов достаточно обширен. И нельзя сказать, что одно решение дешево и непрактично, а другое дорого и универсально, требуется разработать решение, которое будет оптимально в данном случае. Необходимо отметить, что открытые решения должны быть перспективными, чтобы при желании в них можно добавить то, что нужно именно в данном магазине для данного товара, т.е. учесть многие нюансы, которые, несомненно, привлекут своего клиента.

Объект исследования – аренда жилья.

Предмет исследования – веб-сервис для аренды жилья.

Цель дипломного проекта – разработка веб-сервиса для аренды жилья.

Для достижения цели необходимо выполнить следующие задачи:

- выполнить аналитический обзор веб-сервисов для аренды жилья;
- спроектировать информационную систему для аренды жилья;
- разработать веб-сервис для аренды жилья;
- провести технико-экономическое обоснование дипломного проекта.

Следствием внедрения веб-сервиса для аренды жилья будут являться:

- значительное повышение производительности труда и снижение трудозатрат;
- повышение оперативности выполнения задач, связанных с учетом аренды;
- снижение количества ошибок в документации, отчетах и т. д.;
- повышение эффективности учета, а зачастую увеличение рентабельности предприятия.

## **Список использованных источников:**

- [1] Фаулер, М. Архитектура корпоративных программных приложений.: Пер. с англ. – М.: Издательский дом "Вильямс", 2006. – 544с.: ил. – Парал. тит. англ.
- [2] Фримен, А. ASP.NET MVC 4 с примерами на C# 5.0 для профессионалов. 4-е изд. – Россия : Вильямс, 2013. – 688 с.
- [3] Фримен, А. JQuery для профессионалов– Москва-Питер-Киев: Вильямс, 2013. – 953 с.
- [4] Хейлсберг, А. Язык программирования C#. Классика Computers Science / А. Хейлсберг, М. Торгерсен, С. Вилтамут, П. Голд. – СПб.: БХВ – Санкт-Петербург, 2012. – 784 с.
- [5] Яреер, Р.Дж.; Риз, Дж.; Кинг, Т. MySQL и mSQL: Базы данных для небольших предприятий и Интернета; СПб: Символ-Плюс, 2013. – 560 с.

# Поддержка принятия решений по стабилизации производительности сельскохозяйственных культур

Подлесский Г.В., студент гр.751004

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Болтак С.В. – ассистент кафедры ПОИТ

В статье рассматривается общее описание сельскохозяйственной деятельности.

Сельское хозяйство - отрасль экономики, направленная на обеспечение населения продовольствием и получение сырья для ряда отраслей промышленности. Сельское хозяйство является одной из важнейших, представленной практически во всех странах мира.

Основной целью сельскохозяйственной деятельности, связанной с выращиванием растений, является получение наилучшего урожая.

Высокая урожайность зависит от множества факторов: характеристик почвы, оптимального выбора сортов растений, ухода растениями, правильного применения технологий при возделывании культур и другие, следовательно, для того чтобы повысить урожайность, необходимо провести следующий комплекс мероприятий:

1. Повышение плодородия почвы.
2. Соблюдение сроков посева культур.
3. Соблюдения сроков удобрения культур.
4. Использование семян высокого качества.
5. Правильный уход.
6. Соблюдение севооборота.
7. Прогнозирование погодных факторов.
8. Использование качественной сельскохозяйственной техники.

Увеличение урожайности – это процесс, контролируемый самостоятельно или специалистами, придерживаясь главных рекомендаций.

В настоящее время агрономическая наука содержит много знаний для получения высоких урожаев сельскохозяйственных культур и повышения плодородия почвы.

Агрономия превращается в науку точных расчетов, моделей и нормативов с использованием компьютерных вычислений. Создаются новые жидкие, сухие и листовые удобрения, которые улучшают почву и увеличивают урожай.

Агроном - специалист, который зна

ет как выращивать растения, ухаживать за ними, повышать уровень плодородности земли. Главная его цель – увеличение количества и качества урожая. Он управляет всем производственным процессом: принимает решения о засеивании той или иной культуры, следит за тем, чтобы ей оказывали должный уход, принимает решение о своевременном сборе урожая, и так далее. Под его ответственностью могут находиться сотни гектаров посевов, поэтому эта работа отличается повышенным уровнем ответственности и внимательности. Также для агронома важно уметь организовать подчиненных, быстро принимать решения в экстренных ситуациях – например, при пожаре или ухудшении погодных условий.

Агроному приходится постоянно считать. Современные технологии не позволяют хорошему агроному быть плохим математиком. Правильно рассчитать норму расхода удобрений, расход семян или топлива – все это можно сделать на листике, на калькуляторе и в приложении «Excel», но когда нужно определить потребность растений в минеральном питании на запланированный урожай, с учетом свойств почвы, с климатическими особенностями региона и др. Все выше изложенные варианты не подойдут, так как потребует много времени для выполнения данных работ. Для поддержки принятия решений по стабилизации производительности сельскохозяйственных культур стоит использовать приложение.

**Увеличение урожайности** – это контролируемый процесс, которым можно самостоятельно руководить и при необходимости корректировать.

Список используемых источников:

1. Глухин М.А., Агрометеорология – Лань, 2015 – 210с.
2. Голованов А. ., Зимин Ф. ., Сметанин В. ., Рекультивация нарушенных земель – Лань, 2015 – 336с.



# ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ОБРАБОТКИ ЗАКАЗОВ ДЛЯ ПРЕДПРИЯТИЯ ОБЩЕСТВЕННОГО ПИТАНИЯ НА БАЗЕ ПЛАТФОРМЫ .NET FRAMEWORK

*Сергей П.Д.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Неборский С.Н. – доцент кафедры ПОИТ,  
кандидат технических наук*

В данной работе описывается разработка программного обеспечения обработки заказов для предприятий общественного питания, таких как рестораны, кафе, бары и т.д.

Стремительный рост индустрии общественного питания обуславливает необходимость постоянного совершенствования ведения бизнеса не только в управленческом, но и в технологическом плане и требует более современных инструментов профессионального управления бизнес-процессами на производстве. В связи с этим в последнее время активно наметилась тенденция к автоматизации бизнес-процессов на предприятиях питания различных форм и масштабов деятельности. Современному предприятию общественного питания сложно функционировать без автоматизированных систем управления ресторанным бизнесом. Поэтому при разработке бизнес-плана любого проекта, сопряженного с ресторанным бизнесом, закладываются финансовые затраты на его автоматизацию.

Большинство из существующих решений для автоматизации ресторанный бизнеса является весьма дорогостоящими и требуют покупки дополнительного технического оснащения для работы. Это приводит к тому, что многие заведения не могут себе позволить такие решения, особенно после вспышки пандемии, которая стала причиной кризиса в сфере общественного питания [1]. Разрабатываемое программное обеспечение рассчитано как раз на такие заведения, которые не готовы тратить крупные средства на программное обеспечение, однако нуждаются в модернизации.

Основной функцией данного приложения является оформление заказа и отправка его на кухню. Таким образом обслуживающему персоналу не нужно самостоятельно идти на кухню, чтобы передать заказ поварам. Как следствие, можно выявить очевидные преимущества:

1. сотрудники меньше контактируют во время рабочего процесса, что улучшает санитарно-гигиенические условия, это особенно актуально в борьбе с COVID-19;
2. обслуживающий персонал может находиться в зале и оперативно удовлетворять потребности клиентов;
3. сокращается время обслуживания, следовательно, удовлетворенность клиентов возрастает. Более того уменьшение времени обработки заказов дает возможность принимать большее количество посетителей в день: клиент которого обслужили, освобождает место новому.

Еще одной важной функцией программного обеспечения является учет продуктов на складе. После каждого заказа ведется пересчет оставшихся ингредиентов, что позволяет менеджерам видеть актуальную ситуацию на складе и планировать дальнейшие закупки. Также важным достоинством является предотвращение такого воровства, как, например, непробитие чека [2].

Внедрение в производство инновационных технологий в области менеджмента позволяет руководству предприятия решать такие проблемы, как контроль работы трудового коллектива, возможность управления взаимоотношениями с потребителями, планирование маркетинговых мероприятия, ведение финансового и управленческого учета, ведение складского учета, учета движения и расхода материально-сырьевых ресурсов, калькуляция производства и другие.

Автоматизированные системы управления позволяют выявить сильные и слабые стороны предприятия, суметь вовремя подстроиться под изменяющиеся тенденции ресторанный рынка и предупредить возникновение проблем, не дав им возможность нанести материальный ущерб.

Представленное программное обеспечение обработки заказов для предприятия общественного питания представляет собой доступное и многофункциональное решение для ресторанный бизнеса.

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

2. Как пандемия отразилась на белорусском бизнесе [Электронный ресурс]. – Режим доступа: <https://officelife.media/article/money/24341-biznes-v-2020-godu-prakticheski-sovershil-chudo/> — Дата доступа: 17.04.21.

3. Как и сколько персонал ворует в ресторанах [Электронный ресурс]. – Режим доступа: <https://vc.ru/food/223903-kak-i-skolko-personal-voruet-v-restoranaх> - Дата доступа: 17.04.21.

# ПРОГРАММНОЕ СРЕДСТВО УПРАВЛЕНИЯ ОЧЕРЕДЯМИ НА ПЛАТФОРМЕ JAVA

Панкратьев А.С.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Нестеренков С.Н. – канд. тех. наук, доцент

В результате работы было разработано приложения для управления очередями. Пользователи имеют возможность создавать очереди на события, регистрироваться на события, тем самым пользователи системы (как авторы очереди, так и участники) могут планировать свое время более продуктивно.

За всю жизнь человек проводит огромное количество времени в очереди: от сдачи лабораторных работ до очереди в поликлинике или на выступление на конференции.

В большинстве случаев очереди оставляют негативные впечатления как для организаторов, так и для участников. Чаще всего основная проблема состоит в том, что очередь получается неорганизованной. Так же людям сложно просчитать, когда лучше прийти, если они даже не знают количества людей, предшествующих им. Понимание того, сколько людей планирует прийти на то или иное событие так же может помочь компании и сотрудникам компании планировать свое рабочее время более продуктивно[1].

Разработанное решение представляет собой клиент-серверное приложение. Серверная часть приложения построена на платформе Java при использовании фреймворка «Spring». Клиентская часть является браузерным приложением, написанным с использованием библиотеки React. Таким образом пользователи смогут использовать клиент приложения с любого устройства, где установлен браузер: компьютер, смартфон, планшет, ноутбук и т.д.

При переходе на главную страницу, пользователь должен аутентифицироваться или зарегистрироваться. После того, как пользователь аутентифицирован, у него появляется возможность создавать очереди и регистрироваться в очереди. Скриншот страницы регистрации на очередь приведен на рисунке 1.

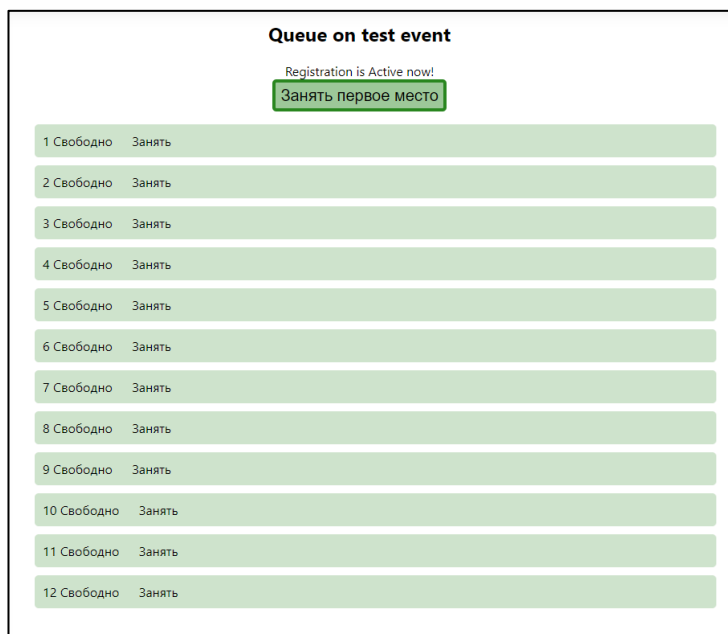


Рисунок 1 – Страница регистрации в очереди

Автор очереди имеет возможность редактировать очередь, а так же помечать задачи как выполненные, давая возможность людям понимать состояние очереди на данный момент.

В качестве корпоративных пользователей системы могут выступать организации, занятые в сферах услуг, образования, медицины. Применение программного средства управления очередями позволит этим компаниям решать задачи организации очередей на события и собирать необходимую информацию от участников мероприятия.

#### Список использованных источников:

1. *Master Your Time, Master Your Life: The Breakthrough System to Get More Results, Faster, in Every Area of Your Life* / B. Tracy // TarcherPerigee, 2016 – 240 p..
2. Шилдт, Г. Полное руководство; 9-е изд.: Пер. с англ. / Г. Шилдт. - М. : ООО "И.Д. Вильямс 2015.

# ВЕБ-ПРИЛОЖЕНИЯ ДЛЯ РАЗМЕЩЕНИЯ ОБЪЯВЛЕНИЙ НА ПРОДАЖУ АВТОМОБИЛЕЙ С ИНТЕЛЛЕКТУАЛЬНЫМ ПОМОЩНИКОМ ОПРЕДЕЛЕНИЯ ЦЕНЫ

Лизунова И.Д.,

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Неборский С.Н. – доцент кафедры ПОИТ,

кандидат тех. наук

Интернет все сильнее захватывает различные сферы человеческой жизни. С его появлением даже такое занятие, как покупка или продажа автомобиля претерпела серьезные изменения. Сегодня эти процедуры можно осуществить, не выходя из дома – достаточно воспользоваться одним из сайтов автопродаж, а чтобы не прогадать со стоимостью автомобиля при размещении объявления, можно воспользоваться интеллектуальным помощником, который соберет данные, проанализирует их и выдаст оптимальную стоимость на рынке.

Работа в сети интернет позволяет создать канал обмена информацией с аудиторией и помогает в решении целого ряда сложных бытовых, технических, маркетинговых и другого рода задач. Например, можно увеличить узнаваемость бренда, повысить лояльность аудитории, провести исследования или организовать продажи.

Также современную эпоху определяет не только сеть интернет, а еще такое изобретение человечества, как автомобиль. Он уже является не роскошью, а обыденным средством передвижения, значение автомобиля ощущается не только в транспортной отрасли, но и во всех сферах человеческой жизни. Он стал осязаемым воплощением технического прогресса, преобразил облик планеты.

Целью данного проекта является создание веб-приложения для размещения объявлений на продажу автомобилей с интеллектуальным помощником определения цены, которое позволяет пользователю размещать объявление о продаже, указывая характеристики автомобиля, фото, а также контактные данные владельца, при этом встроенный интеллектуальный помощник может рассчитать конкурентоспособную цену по рынку.

В веб-приложении помимо стандартных функций: добавление пользователя, удаления аккаунта пользователя, размещение объявления о продаже автомобиля, удаления объявления – также реализованы функции просмотра новостей и сервис интеллектуального помощника генерации цены для пользователя, продающего автомобиль.

Функция определения цены работает на основе обработанных многотысячных данных. Для построения модели необходимо провести целый ряд действий, они приведены на рисунке 1[1].

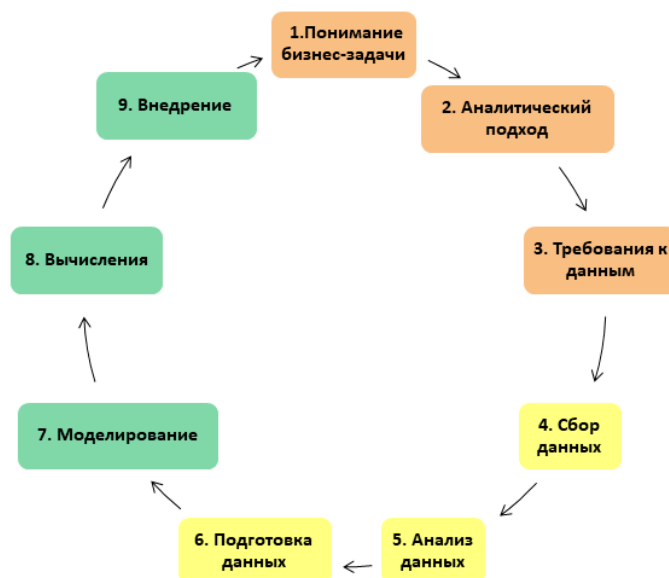


Рис.1 – Этапы построения модели

На первом этапе мы должны понять, какой результат требуется от проекта(модели), провести анализ участников и заинтересованных лиц, а также сформировать постановку задачи.

Далее необходимо определить тип бизнес-задачи, выбрать подход для её решения.

Следующим этапом является работа с данными, а именно поиск доступных источников, а также подготовка таких требований, как формат, контент и т.д. Потом происходит сбор данных, после которого проводится анализ на наличие необходимых данных.

Следующий шаг – это формирование выборки, с которой предстоит работа в дальнейшем. Необходимо провести ряд исследований, чтобы иметь представление, как устроена выборка: исследуется вариабельность, выявляется корреляция между признаками, ищутся выбросы, исследуется центральное положение, а также строится график распределения.

Далее идет подготовка данных, которая занимает большую часть всего процесса. На этой стадии данные обрабатываются в удобный для работы формат: удаляются дубликаты, обрабатываются отсутствующие или неверные данные, проверяются и при необходимости исправляются ошибки форматирования.

Следующим этапом является программирование модели, далее её обучение на обучающейся выборке, проводится валидация с помощью тестовой выборки, если результат удовлетворительный, модель внедряется в основную систему, приходят отчеты о работе модели и цикл начинается сначала.

Как устроены модели в общем представлении, можно посмотреть на примере Random Forest. Общая идея метода "расфасовки" заключается в том, что комбинация моделей обучения увеличивает общий результат, т.е. случайный лес строит несколько деревьев решений и объединяет их вместе для получения более точного и стабильного прогноза [2]. Подмножество данных производится случайным выбором  $x$  числа признаков (столбцов) и  $y$  числа примеров (строк) из исходного набора данных  $n$  признаков и  $m$  примеров, т.е. деревья решений имеют разный набор гиперпараметров(существует множество факторов, которые могут быть использованы для определения случайной модели леса: максимальное количество признаков, используемых для разделения узла или количество деревьев в лесу - можно вручную установить и настроить эти значения или изменить и обновить их) и разное подмножество обучающих данных, следовательно, решение или предсказание, даваемое этими деревьями, может сильно варьироваться – в таком случае алгоритм учитывает предсказания деревьев простым голосованием: какое решение выбрало большинство, то и будет результатом тестового примера.

Одним из преимуществ случайного леса является то, что его можно использовать как для задач классификации, так и для задач регрессии, которые формируют большинство современных систем машинного обучения [3]. Случайный лес в классификации с двумя деревьями изображен на рис.2.

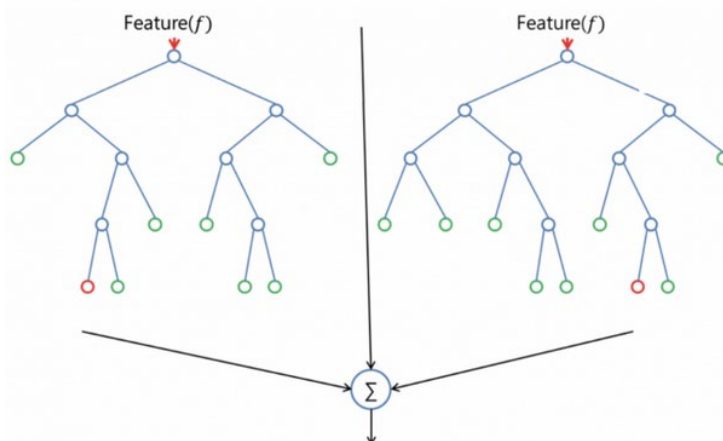


Рис.2 – Случайный лес с двумя деревьями

Программная реализация интеллектуального помощника представляет собой аналогичное структурное решение в веб-приложении.

**Список использованных источников:**

1. 7 steps to implement a data science project [Электронный ресурс]. – Режим доступа: <https://www.xpand-it.com/blog/implement-data-science-project/> — Дата доступа: 17.04.21.
2. Bayesian Reasoning and Machine Learning / David Barber – Cambridge University Press, 2010 – С. 610
3. The definitive guide to Random Forests and Decision Trees. [Электронный ресурс]. – Режим доступа: <https://victorzhou.com/blog/intro-to-random-forests/> — Дата доступа: 17.04.21.
4. Let's open the black box of random forests [Электронный ресурс]. – Режим доступа: <https://www.analyticsvidhya.com/blog/2020/12/lets-open-the-black-box-of-random-forests/> — Дата доступа: 15.04.21.
5. Data Science. Наука о данных с нуля / Грас Джоэл – СПб: БХВ-Петербург, 2019 – С. 336

# СИСТЕМА УПРАВЛЕНИЯ КАЛИБРОВОЧНЫМИ ГАЗАМИ НА ПЛАТФОРМЕ .NET

Дрожжа Е.А.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Болтак С.В. – ассистент каф.ПОИТ

Оборудование, применяющееся для высокоточных измерений концентраций различных газов, нуждается в частой калибровке. Для проведения калибровки необходимо помещать датчики в среду газа с заранее определенной концентрацией. Разрабатываемая система мониторинга предназначена для контроля состояния газового оборудования и баллонов, обнаружения утечек и обеспечения обслуживания системы.

В множестве различных сфер используется оборудование, измеряющее концентрации газов. Одной из таких сфер является измерение выбросов загрязняющих веществ двигателями внутреннего сгорания, для их последующей сертификации [1]. Данные испытания требуют высокой точности измерений, оборудование, определяющее концентраций отдельных газов в общей смеси, нуждается в периодической калибровке [2]. Для этого на измерительную станцию поставляются баллоны с эталонными газами, концентрации компонентов которых точно определены заводом изготовителем. Основная задача разрабатываемой системы, организовать учет газовых баллонов и обеспечить подачу газов из них на измерительное оборудование. При этом необходимо максимально автоматизировать процесс, в целях сокращения количества обслуживающего персонала и потери газов.

Возможности системы:

- контроль давления в газовых баллонах и уровня их заполнения;
- отслеживание истечения срока эксплуатации и прогнозирование оставшегося времени использования баллонов;
- оповещение операторов о необходимости замены баллонов;
- автоматическая отправка уведомлений при отклонении отслеживаемых показателей от установленных пределов;
- автоматическая продувка газопроводов;
- запираение и отпираение газовых баллонов;
- регистрация новых газовых баллонов в системе путем сканирования QR-кодов или обработки - PDF-сертификатов, полученных от производителей;
- создание отчетов о состоянии системы в выбранном промежутке времени.

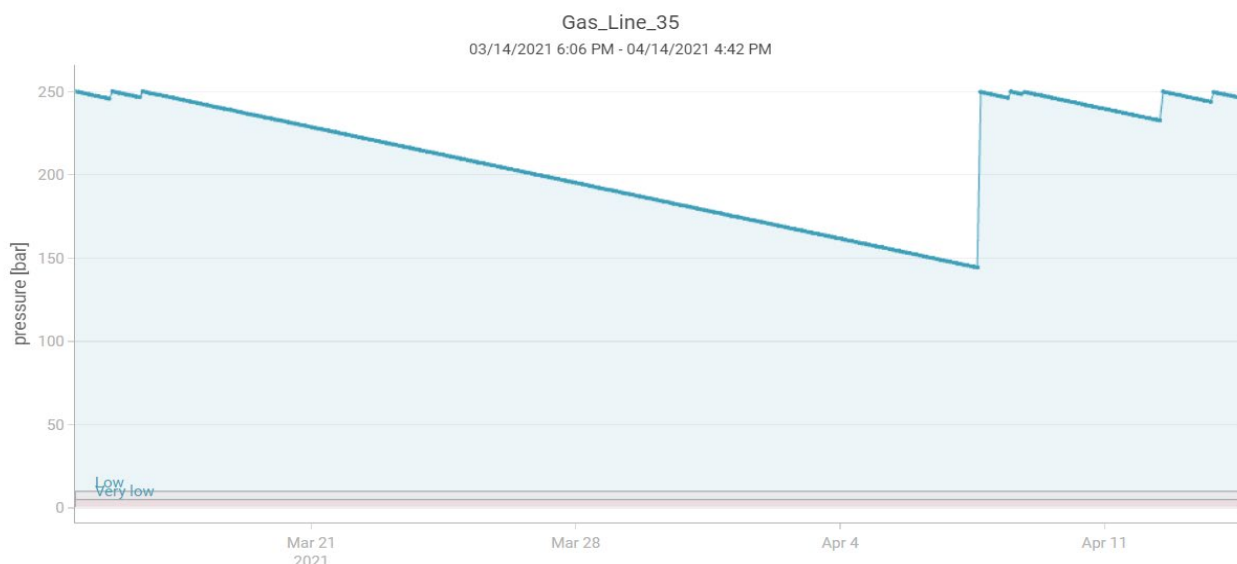


Рисунок 1 – График изменения давления в газопроводе

## Список использованных источников:

2015. [1] Kuranc A.R. Exhaust emission test performance with the use of the signal from air flow meter. University of Life Sciences in Lublin, 2015.
- [2] Davison R.C., Nevill A.M. Performance Characteristics of Gas Analysis Systems. Georg Thieme Verlag KG, 2004.

# ВЕБ-ПРИЛОЖЕНИЕ ДЛЯ СОЗДАНИЯ ИНТЕРАКТИВНЫХ ТРЕНИНГОВ С ИСПОЛЬЗОВАНИЕМ SPRING И ANGULAR

Шуляк Р.В., студент гр.751003

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Можей Н.П. – доцент кафедры ПОИТ,  
кандидат физ.-мат. наук

В данной работе описывается разработка веб-приложения для создания интерактивных тренингов, а также рассматриваются его отличия от существующих программных средств.

В настоящее время всё больше людей выбирает формат онлайн-обучения. Это может быть связано с тем, что многим такой формат удобнее обычных аудиторных занятий, так как он позволяет получить доступ к образовательным материалам из любой точки мира. Особенно актуален этот вид обучения стал во время пандемии COVID-19. За последний год компании, которые занимаются разработкой тренинговых платформ, показали небывалый рост числа пользователей. Так, например, на одной из самых популярных платформ для онлайн обучения “Coursera” количество пользователей выросло на 640%, а на “Udemy” – рост на 400% [1]. Также за последние годы количество доступных онлайн курсов выросло в несколько раз. На рисунке 1 приведена статистика роста количества онлайн курсов.

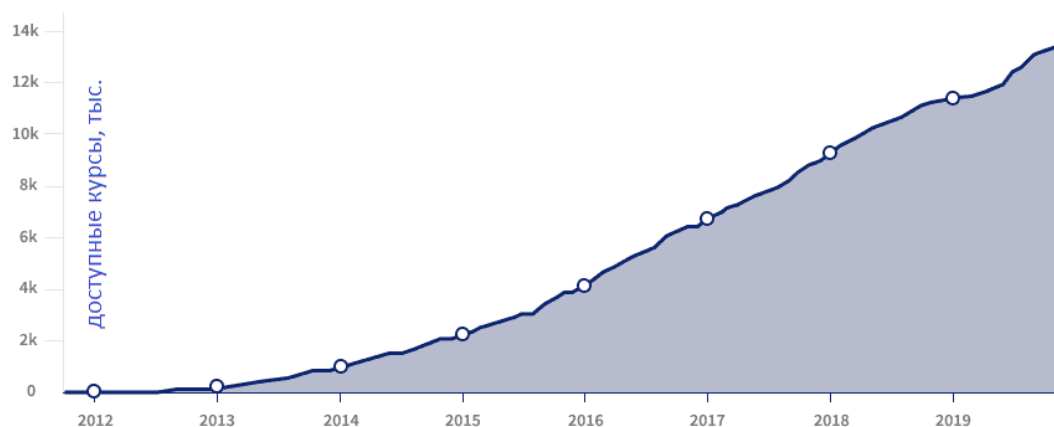


Рисунок 1 – статистика роста количества онлайн курсов с 2012 по 2019 годы [2]

Однако, при всей своей популярности, существующие платформы для тренингов достаточно однообразны. Зачастую в них просто присутствует видео длиной в несколько часов или огромная презентация. Данный формат подходит далеко не всем пользователям, так как он является очень монотонным и через какое-то время человек перестает эффективно усваивать информацию. Одна из ключевых особенностей разрабатываемого приложения заключается в том, что каждый курс разбивается на отдельные фрагменты. Каждый из них, в свою очередь, может содержать разнообразный контент, например:

1. Видеофайлы;
2. Аудиофайлы;
3. Картинки;
4. Текстовую информацию;
5. Возможность добавления мини-тестов при переходе между фрагментами.

Данный список может быть расширен и дополнен новыми форматами. Это позволит добавить элемент интерактивности в тренинги, так как пользователь постоянно будет взаимодействовать с приложением и получать информацию в разных формах. После прохождения теоретической части тренинга, пользователю будет доступен тест, необходимый для завершения тренинга. В свою очередь, автор сможет просматривать подробную статистику по своим тренингам.

При разработке приложения используется следующий стек технологий: Spring, Angular, MySQL. Spring Framework — универсальный фреймворк с открытым исходным кодом для Java-платформы [3]. Он является простым, надёжным и наиболее популярным для написания серверной части Java-приложений. Для написания клиентской части использован Angular. Он позволяет делать запросы к серверной части и динамически отображать информацию в браузере. Обмен информацией между клиентской и серверной частью осуществляется по протоколу HTTP в формате JSON. MySQL - свободная реляционная система управления базами данных [4].

Данный стек технологий позволяет достаточно быстро разработать рабочий прототип приложения, а описанные выше отличия выделяют веб-приложение на фоне других тренинговых платформ, что поможет ему набрать популярность среди пользователей. Так же важной особенностью является то, что авторы не должны будут платить за размещение своих курсов на платформе, а доход от платформы будет формироваться за счёт комиссии при продаже курсов.

Поскольку сейчас тренинговые платформы популярны как никогда и велика вероятность, что с каждым годом они будут становиться еще популярнее, вышеописанное веб-приложение для создания интерактивных тренингов является актуальным и может оказать существенную помощь как разработчикам онлайн-курсов, так и их слушателям.

**Список использованных источников:**

4. *Massive online open courses see exponential growth during COVID-19 pandemic* [Электронный ресурс]. – Режим доступа: <https://theconversation.com/massive-online-open-courses-see-exponential-growth-during-covid-19-pandemic-141859> — Дата доступа: 02.04.21.
5. *By The Numbers: MOOCs in 2019* [Электронный ресурс]. – Режим доступа: <https://www.classcentral.com/report/mooc-stats-2019> — Дата доступа: 04.04.21.
6. *Spring Framework* [Электронный ресурс]. – Режим доступа: [https://ru.wikipedia.org/wiki/Spring\\_Framework](https://ru.wikipedia.org/wiki/Spring_Framework) — Дата доступа: 03.04.21.
7. *MySQL* [Электронный ресурс]. – Режим доступа: <https://ru.wikipedia.org/wiki/MySQL> — Дата доступа: 05.04.21.

# ПРОГРАММНЫЙ МОДУЛЬ САМООБСЛУЖИВАНИЯ НА БАЗЕ MICROSOFT DYNAMICS 365 COMMERCE POS

*Олешко И. В., студент гр. 7510003*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Можей Н.П. – кандидат физико-математических наук, доцент*

В данной работе описывается разработка программного модуля для расширения функциональности Microsoft Dynamics 365 Commerce Pos до системы кассы самообслуживания.

Главной причиной недовольства покупателей и потери прибыли компаний часто являются огромные очереди в магазинах [1]. Собрав необходимые товары, люди хотят побыстрее расплатиться и уйти, не раздражаясь лишней раз из-за медленных кассиров и других покупателей. Исследования указывают на то, что наиболее чувствительная к очередям доля покупателей – это так называемые экспресс-покупатели с небольшой корзиной товаров. Они совсем не желают стоять в очереди с покупателями, разгружающими огромные тележки. К тому же экспресс-покупатели составляют большую часть всех посетителей супер- и гипермаркетов.

Самым очевидным решением сложившейся ситуации был бы способ увеличения производительности и пропускной способности супермаркетов через повышения гибкости системы расчетно-кассовых узлов. Это позволило бы сократить очереди и время ожидания в них. Решением данной проблемы могут стать «кассы самообслуживания», позволяющие покупателям самостоятельно сканировать, упаковывать и оплачивать товары.

Внедрение в свой бизнес касс-самообслуживания сопровождается большими затратами, связанными с приобретением аппаратного и программного обеспечения. Зачастую на этом проблемы не заканчиваются. Все крупные компании используют ERP-системы для управления своими ресурсами и внутренними процессами. И почти всегда главным вопросом становится то, возможна ли интеграция приобретаемой системы с уже имеющейся ERP-системой. Ведь большинство систем самообслуживания, предоставленных на рынке, являются обособленными решениями, не имеющими такой возможности.

Целью работы является разработка программного расширения, предназначенного для одной из самых популярных ERP-систем Microsoft Dynamics 365, позволяющее использовать Microsoft Dynamics 365 Commerce POS в качестве кассы самообслуживания. Данное приложение может быть использовано в магазинах розничной торговли. Оно позволит автоматизировать процесс продажи товаров и увеличить пропускную способность точки продажи.

Главной особенностью, выделяющей разрабатываемое приложение среди конкурентов, является то, что оно изначально разработано для интеграции с ERP-системой Dynamics 365, что позволит пользователям данной системы в минимальные сроки внедрить данное решение в свой бизнес. Так же одним из преимуществ приложения является возможность конфигурации некоторых бизнес-процессов и пользовательского интерфейса POS под требования клиентов;

При разработке приложения используется следующий стек технологий: ASP.NET Web API, Knockout.js, MS SQL, X++. ASP.NET Web API — универсальный фреймворк для создания веб-интерфейсов для .NET-платформы. Он используется для расширения серверной части приложения. Для расширения клиентской части использован Knockout.js. Он позволяет делать запросы к серверной части и создавать гибкий пользовательский интерфейс. X++ - язык, используемый для расширения логики системы Microsoft Dynamics 365 [2]. MS SQL - реляционная система управления базами данных, созданная компанией Microsoft и используемая в большинстве продукции компании [3].

Данный стек технологий предопределен архитектурой системы Microsoft Dynamics 365 Commerce. Приложение включает в себя расширение 3 частей Microsoft Dynamics 365 Commerce: Microsoft Dynamics 365, Retail Server и Microsoft Dynamics 365 Commerce POS.

Поскольку сейчас наблюдается тенденция увеличения количества внедряемых касс самообслуживания и роста доли Microsoft Dynamics 365 среди остальных ERP-систем, вышеописанное приложение является актуальным и сможет оказать существенную пользу как компаниям, приобретающим данное решение, так и покупателям.

## **Список использованных источников:**

1. *The Cost of Queues: How Improper Queue Management Affects Your Bottom Line [Электронный ресурс]. – Режим доступа: <https://www.qminder.com/cost-customer-service-wait-times/> — Дата доступа: 02.04.21.*
2. *X++ [Электронный ресурс]. – Режим доступа: <https://ru.wikipedia.org/wiki/X++> — Дата доступа: 03.04.21.*
3. *Microsoft Sql Server [Электронный ресурс]. – Режим доступа: [https://ru.wikipedia.org/wiki/Microsoft\\_SQL\\_Server](https://ru.wikipedia.org/wiki/Microsoft_SQL_Server) — Дата доступа: 05.04.21.*



# МОДУЛЬ РАСЧЕТА ДИРЕКТИВ ДЛЯ МЕСТ ХРАНЕНИЯ НА СКЛАДЕ В ERP-СИСТЕМЕ D365FO

Киселев Г.А., студент гр.751006

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Можей Н.П. – доцент кафедры ПОИТ,  
кандидат физ.-мат. наук

В работе для разрабатываемого модуля предложена математическая модель формирования директив для мест хранения в пределах склада с учетом топологии складских помещений и стратегии хранения и отбора продукции.

На сегодняшний день любое предприятие, которое хочет успешно и стабильно развиваться нуждается во внедрении WMS-системы (Warehouse Management System — Система управления складом) в свое производство. Основной целью такой системы является управление технологическими процессами склада, а именно: увеличение скорости набора товара, получение точной информации о местонахождении товара, получение инструмента для повышения эффективности и развития процессов по обработке товара, оптимизация использования складских площадей и т.д. [1].

В успешном выполнении складом своих функций важную роль играет его топология. Планировка складских помещений может существенно варьироваться в зависимости от вида и предназначения склада, хранимой на нем продукции.

Для успешной автоматизации процессов на складе требуется решить ряд задач. Одной из таких задач является задача транспортная задача. Суть задачи заключается в построении оптимальных маршрутов на складе для роботов-кладовщиков. Построение таких маршрутов невозможно без разбиения мест хранения на складе на директивы. В работе представлен такой алгоритм, классифицирующий места хранения на основе их приоритетности. Алгоритм учитывает топологию складских помещений и стратегии хранения и отборки продукции. Также в работе рассмотрена существующая реализация модуля управления складом в ERP-системе Microsoft Dynamics 365 для последующей ее оптимизации посредством внедрения разработанных алгоритмов [2].

В Microsoft Dynamics 365 алгоритм, решающий задачу определения ячеек, из которых будет отбираться продукция, можно разделить на два этапа: определение зоны или зон, в рамках которых будет производиться отборка и определение конкретных ячеек из подходящих зон. Так реализован учет топологии склада посредством разделения на зоны, каждая из которых имеет фиксированный приоритет. Однако у существующего подхода имеются места, требующие усовершенствования для более эффективного использования ресурсов робота.

В частности, разрабатывается механизм определения приоритетов зон с помощью алгоритмов кластеризации. Он предоставит возможность выбора количества приоритетов зон, а также будет учитывать топологию склада и расположения зон отгрузки и загрузки [3]. Ячейки хранения, находящиеся ближе к оптимальному пути от зоны отгрузки до зоны загрузки и обратно будут иметь более высокий приоритет по сравнению с другими. Для поиска оптимального пути складское помещение необходимо представить в виде матрицы, в которой будет храниться информация о проходах и местах хранения на складе. Применив волновой алгоритм поиска кратчайшего пути в лабиринте к этой матрице, такой путь будет рассчитан. Полученная после расчета информация послужит входными данными для алгоритма разбиения мест хранения на директивы.

По сравнению с традиционными методами расчета директив хранения, разработанный модуль обеспечит более высокую эффективность и точность [4], в результате чего такой подход к автоматизации складских систем найдет применение в различных отраслях промышленности и сферы услуг.

## Список использованных источников:

8. Система управления складом [Электронный ресурс]. – [https://ru.wikipedia.org/wiki/Система\\_управления\\_складом](https://ru.wikipedia.org/wiki/Система_управления_складом) — Дата доступа: 01.04.21.

9. Finance and Operations application documentation [Электронный ресурс]. – Режим доступа: Finance and Operations application documentation - Finance & Operations | Dynamics 365 | Microsoft Docs — Дата доступа: 02.04.21.

10. Методы и алгоритмы принятия решений: учебно-методическое пособие — [Электронный ресурс]. – Режим доступа: <https://libeldoc.bsuir.by/handle/123456789/10713> — Дата доступа: 03.04.21.

11. Concept logic [Электронный ресурс]. – Оценка повышения эффективности розничного комиссионирования на складах e-commerce (clogic.ru) — Дата доступа: 05.04.21.

# ВЕБ-ПРИЛОЖЕНИЕ ДЛЯ СИНТЕЗА, ХРАНЕНИЯ И РАСПРОСТРАНЕНИЯ АУДИОКНИГ НА БАЗЕ SPRING FRAMEWORK

Гринчик В.В., студент гр.751003  
Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Можей Н.П. – доцент кафедры ПОИТ,  
кандидат физ.-мат. наук

В данной работе описывается разработка веб-приложения для хранения и распространения аудиокниг, а также их синтеза на основе печатного текста, рассматриваются его отличия от существующих программных средств.

С каждым годом доля электронных изданий на международном рынке неуклонно растет. Например, по данным группы компаний «ЛитРес» – крупнейшего представителя рынка электронных и аудиокниг в России и странах СНГ, по итогам первого полугодия 2020 года объем продаж аудио вырос более чем на 45,5%. А общая доля аудиокниг на рынке цифровых изданий достигла 23% [1]. Столь стремительный рост рынка аудиокниг вызвал потребность в приложениях, предназначенных для синтеза, хранения и распространения аудиокниг. Существующие на данный момент приложения могут удовлетворить эту потребность только частично.

На текущий момент существует множество приложений для хранения и распространения аудиокниг, и только некоторые из них позволяют пользователям заказать добавление новой аудиокниги. Однако, как правило, такой заказ является платной услугой, а на его исполнение требуется значительный промежуток времени. И если для большинства людей это приемлемо, то для инвалидов и людей с ограниченными возможностями по зрению это становится серьезным препятствием на пути получения информации из литературных произведений.

Одной из ключевых особенностей разрабатываемого приложения является предоставление пользователю возможности самостоятельно синтезировать аудиокниги на основе печатного текста. Внедрение технологии синтеза речи в приложение для хранения и распространения аудиокниг серьезно упростит процессы создания и распространения аудиокниг, а также значительно облегчит доступ к информации, получаемой из литературы, людям с ограниченными возможностями.

Технология синтеза речи, используемая в приложении, основана на компилятивной модели синтеза. Модель компилятивного синтеза предполагает синтез речи путем конкатенации (составления) записанных образцов отдельных звуков, произнесенных диктором. При использовании этой модели составляется база данных звуковых фрагментов, из которых в дальнейшем будет синтезироваться речь [2].

При разработке приложения используется следующий стек технологий: Spring, JavaScript V8, MySQL. Spring Framework — универсальный фреймворк с открытым исходным кодом для Java-платформы [3]. Он является простым, надёжным и наиболее популярным для написания серверной части Java-приложений.

Для написания клиентской части использован язык JavaScript с версией ядра V8. Он позволяет делать запросы к серверной части и динамически отображать информацию в браузере. Обмен информацией между клиентской и серверной частью осуществляется по протоколу HTTP в формате JSON. MySQL - свободная реляционная система управления базами данных [4].

Данный стек технологий позволяет достаточно быстро разработать рабочий прототип, а описанные выше отличия выделяют веб-приложение на фоне других приложений для синтеза, хранения и распространения аудиокниг, что поможет ему быстро набрать популярность среди пользователей. Так же важной особенностью является то, что весь функционал приложения предоставляется абсолютно бесплатно, а доход от проекта будет формироваться из выплат за размещения рекламных объявлений.

Приложение для синтеза, хранения и распространения аудиокниг окажет положительное влияние на социально-психологическую адаптацию инвалидов и людей с ограниченными возможностями в обществе, а также повысит их качество жизни.

## Список использованных источников:

1. Литература растёт в гаджете [Электронные данные] - Режим доступа: <https://www.rspectr.com/articles/678/literatura-rastet-v-gadzhete> - Дата доступа: 05.04.2021.
2. An Introduction to Text-to-Speech Synthesis [Электронные данные] - Режим доступа: [https://www.academia.edu/416871/A\\_Short\\_Introduction\\_to\\_Text\\_to\\_Speech\\_Synthesis](https://www.academia.edu/416871/A_Short_Introduction_to_Text_to_Speech_Synthesis) - Дата доступа: 01.04.2021.
3. Spring Framework [Электронный ресурс]. – Режим доступа: [https://ru.wikipedia.org/wiki/Spring\\_Framework](https://ru.wikipedia.org/wiki/Spring_Framework) — Дата доступа: 03.04.2021.
4. MySQL [Электронный ресурс]. – Режим доступа: <https://ru.wikipedia.org/wiki/MySQL> — Дата доступа: 05.04.2021.

# МЕССЕНДЖЕР НА БАЗЕ ПЛАТФОРМЫ .NET

Калиниченко Д.С

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Можей Н.П. – канд. физ.-мат. наук

Работа посвящена разработке средства коммуникации внутри IT-предприятия. Присутствует поддержка шифрования сообщений. Реализована отправка текстовых и графических сообщений.

Мессенджер - это программа, мобильное приложение или веб-сервис для мгновенного обмена сообщениями. Чаще всего под мессенджером понимают программу, в которую вы пишете сообщения и где вы их читаете. Однако, за каждой такой программой стоит сеть обмена сообщениями, которая тоже входит в понятие "мессенджер". Это может быть сеть внутри вашей компании, а может быть глобальная сеть [1].

На сегодняшний момент язык программирования C# один из самых мощных, быстро развивающихся и востребованных языков в IT-отрасли. На нем пишутся самые различные приложения: от небольших настольных программ до крупных веб-порталов и веб-сервисов. Технология WPF (Windows Presentation Foundation) представляет собой подсистему для построения графических интерфейсов. Главным ее преимуществом является то, что за отрисовку элементов управления отвечает DirectX, набор API которого способны работать с видеоускорителями, тем самым обеспечивая приложению возможность аппаратного ускорения графики [2]

Программное средство реализовано в виде настольного приложения. Разработка проводилась на базе платформы .NET с использованием языка программирования C# и технологии WPF, также в ходе разработки использовалась библиотека программного обеспечения SignalR, которая позволяет серверу отправлять асинхронные уведомления клиенту [3].

Программное средство имеет интуитивно-понятный интерфейс и высокую производительность без использования большого объема оперативной памяти, при этом не требует от пользователя наличия производительного компьютера, поддерживает все необходимые функции. Мессенджер предоставляет возможность авторизации, отправки текстовых и графических сообщений, возможность редактировать данные учетной записи, поддерживает шифрование. Программное средство использует легковесную SQL базу данных, которая обеспечивает надежную защиту и хранение данных при этом не снижая производительность приложения. Для эффективного внедрения базы в программное средство использовалась технология Entity Framework от компании Microsoft для доступа к данным. Entity Framework представляет собой более высокий уровень абстракции в работе с сущностями базы данных, что в свою очередь упростило разработку приложения и его возможного дальнейшего расширения [4]. Программное средство способно хорошо масштабироваться под экраны разных размеров с разным разрешением, что позволяет при необходимости с легкостью перенести разработку на мобильные устройства. Благодаря выбранной технологии программное средство имеет большие возможности для дальнейшего расширения: поддержки большого количества файлов мультимедиа, использование двумерной и трехмерной графики, возможность создания шаблонов и тем оформления предоставляет новое направление для еще более тонкой настройки пользовательского интерфейса. Простота в развертывании приложения на компьютере позволяет обеспечить легкое сопровождение программного средства администраторами для добавления, редактирования или обновления информации о пользователях или для управления процессами программного средства.

Все используемые в данной работе технологии обеспечивают хорошую производительность приложения и его комфортное использование пользователями. А все это в свою очередь поможет повысить производительность труда и упростить общение друг с другом, отходя от не очень удачного способа ввести быстрые переговоры через электронную почту. Почти любой человек привык быть на связи всегда и везде, а мессенджер позволяет этого добиться. Программное средство реализовано в виде настольного приложения и рекомендуется для использования на предприятиях с ограниченной вычислительной мощностью рабочих компьютеров, на которые будет установлена клиентская часть приложения. Поддерживается аппаратное ускорение графики, что, в свою очередь, подразумевает поддержку качественной графики и возможность дальнейшего развития пользовательского интерфейса. Оно достаточно простое для дальнейшего расширения.

#### Список использованных источников:

1. Что такое мессенджер [Электронный ресурс]. – Режим доступа: <https://www.voipoffice.ru/tags/messendzhery/>
2. Особенности платформы WPF [Электронный ресурс]. – Режим доступа: <https://metanit.com/sharp/wpf/1.php>
3. Эффективное управление подключениями SignalR [Электронный ресурс]. – Режим доступа: <https://habr.com/ru/post/470299>
4. Что такое Entity Framework [Электронный ресурс]. – Режим доступа: <https://metanit.com/sharp/entityframework>

# ПРОГРАММНОЕ СРЕДСТВО «ОНЛАЙН-КИНОТЕАТР С ИНТЕЛЛЕКТУАЛЬНЫМ ПОМОЩНИКОМ» НА БАЗЕ ПЛАТФОРМЫ ASP.NET CORE

Абраменко А.В., студент гр.751003  
Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Можей Н.П. – доцент кафедры ПОИТ,  
кандидат физ.-мат. наук

В данной работе описывается разработка программного средства для просмотра фильмов онлайн, а также для их подбора на основе пожеланий пользователя при помощи интеллектуального помощника, дополнительно рассматриваются его отличия от существующих программных средств.

На сегодняшний день просмотр фильмов является одним из самых частых видов времяпрепровождения [1]. Всё из-за того, что фильмы способны подарить невероятные эмоции наравне с чтением любой книги. В результате в интернете одними из самых популярных сайтов являются онлайн-кинотеатры, которые предоставляют огромное количество фильмов, мультфильмов и сериалов.

Однако бывают случаи, когда пользователь онлайн-кинотеатра теряется в выборе просматриваемого фильма из-за его большого количества в наличии. В этом случае пользователю было бы удобно, если бы онлайн-кинотеатр выполнил какой-нибудь подбор фильма на основе критериев, выбранных пользователем, и на основе просмотренных и оценённых фильмов того же пользователя. Однако такая функциональность зачастую отсутствует у онлайн-кинотеатров, и для подобного вида подбора фильма пользователям требуется использовать другие программные средства, например, «Кинонавигатор» [2], которые зачастую именуются интеллектуальным помощником для подбора фильмов.

Главной идеей разрабатываемого программного средства является совмещение функциональности онлайн-кинотеатра и интеллектуального помощника для подбора фильмов. Данное совмещение позволит пользователю программного средства без затруднений подобрать фильм на основе своих пожеланий и сразу начать его просмотр, и всё это в рамках одного программного средства. Однако, для осуществления интеллектуальным помощником подбора наиболее подходящего фильма потребуются предоставить пользователю обширный набор критериев для фильтрации фильмов, а для этого потребуются хранить достаточно обширную информацию о каждом из них. Также следует сообщить пользователю о том, что оценивание просмотренного им фильма приведёт к улучшению точности подбора фильмов.

Платформа с открытым исходным кодом ASP.NET Core представляет технологию от компании Microsoft, предназначенную для создания различного рода веб-приложений: от небольших веб-сайтов до крупных веб-порталов и веб-сервисов. Данная платформа является кроссплатформенной и характеризуется расширяемостью [3]. Вместе с данной платформой используется Microsoft SQL Server - свободная реляционная система управления базами данных, разработанная компанией Microsoft [4].

Данный стек технологий позволяет достаточно быстро разработать рабочий прототип программного средства, а описанные выше отличия выделяют его на фоне других онлайн-кинотеатров, что поможет ему набрать популярность среди пользователей.

В качестве возможных направлений улучшения разработанного программного средства являются: качество и количество информации, хранимой о каждом фильме, что позволит пользователю побольше узнать о том, что ему интересно, и позволит увеличить возможности фильтрации, и качество подбора интеллектуальным помощником.

#### Список использованных источников:

1. Популярные способы времяпрепровождения [Электронный ресурс] – Режим доступа: <https://adukar.by/news/abiturientu/kakie-hobbi-populyarny-u-belorusskikh-podrostkov>. – Дата доступа: 05.04.2021.
2. «Кинонавигатор» [Электронный ресурс] – Режим доступа: <https://kinonavigator.ru/>. – Дата доступа: 05.04.2021
3. ASP.NET Core [Электронный ресурс] – Режим доступа: <https://metanit.com/sharp/aspnet5/1.1.php>. – Дата доступа: 05.04.2021.
4. Microsoft SQL Server [Электронный ресурс]. – Режим доступа: [https://ru.wikipedia.org/wiki/Microsoft\\_SQL\\_Server](https://ru.wikipedia.org/wiki/Microsoft_SQL_Server). – Дата доступа: 05.04.2021.

# МОБИЛЬНОЕ ПРИЛОЖЕНИЯ ДЛЯ ОКАЗАНИЯ И ПОЛУЧЕНИЯ ПОМОЩИ С ИСПОЛЬЗОВАНИЕМ ТЕХНОЛОГИИ FLUTTER И ЯЗЫКА DART

Козлова Ю.И.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Петюкевич Н. С. – старший преподаватель

Для оказания и получения помощи существует огромное количество организаций и фондов. Мобильное приложение – это вспомогательный инструмент, посредник между пользователем, заинтересованным в оказании помощи, и специализированными организациями. Приложения могут выступать как платформа для безопасного сбора материальных средств или в качестве структурированного места с подробной информацией о фондах и вариантах оказания и получения помощи.

Человек проводит достаточно большое количество времени в мобильном телефоне. Одно устройство является и универсальным помощником, и средством связи, и местом для отдыха и игр. Он компактный, поэтому он находится рядом с человеком почти в любой ситуации. Разработка мобильного приложения – однозначно верное решение для привлечения современного потребителя.

Цель работы – создание мобильного приложения для помощи людям в тяжёлой жизненной ситуации, животным в приютах, начинающим и талантливым специалистам, интересным стартапам. Также приложение предоставляет возможность получения этой помощи, если пользователь нуждается в ней. В нём предусмотрена функция сортировки по темам, предоставление подробной информации по каждому разделу, выбор вида помощи (материальная, покупка вещей/еды/предметов, волонтерство, помощь в качестве специалиста какой-либо отрасли). Зарегистрированный пользователь может отслеживать развитие фондов, которым он оказывает помощь, может получать рассылку с актуальной информацией о подписанных темах и срочных взносах. Функционал приложения отображает рисунок 1.



Рисунок 1 – Вариант использования мобильного приложения

Данная идея реализована с использованием фреймворка Flutter, который является кроссплатформенным и подходит как для разработки мобильных приложений как для Android, так и для iOS. В качестве вспомогательных инструментов были выбраны интегрированная среда разработки Android Studio, язык Dart, облачная база данных Firebase и BLoC паттерн для отделения бизнес-логики от интерфейса [1-2].

#### Список использованных источников:

1. Flutter. Плюсы и минусы [Электронный ресурс]. – Режим доступа: <https://habr.com/ru/company/simbirsoft/blog/441766/>
2. Архитектура Flutter-приложения [Электронный ресурс]. – Режим доступа: <https://medium.com/@inostudio/flutter-bloc-21f2274203d6>

# ВЕБ-ПРИЛОЖЕНИЕ УПРАВЛЕНИЯ ИНТЕРНЕТ-ЗАКАЗАМИ ДЛЯ ЗАВОДА ЖБИ С ИСПОЛЬЗОВАНИЕМ REACTJS

Шмеерова А.Л.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Петюкевич Н. С. – старший преподаватель

Наличие веб-сайта решает ряд таких задач, как представление компании клиентам, расширение круга потенциальных клиентов, привлечение новых партнеров по бизнесу, что в последствии ведет к увеличению объема продаж и способствует развитию компании. Разработанная система, являющаяся комбинацией веб-сайта и веб-приложения по управлению заказами, призвана облегчить работу сотрудника, занимающегося ведением веб-сайта и обработкой поступающих с данного сайта заказов.

За последнее время интернет превратился в мощный инструмент общения с конечными пользователями. В связи с чем веб-приложения получили большую популярность. Основное преимущество веб-приложений заключается в установке конечному клиенту.

Целью данной работы является создание веб-приложения, включающего в себя управление контентом и поступающими заказами веб-сайта завода ЖБИ. Веб-приложение позволит клиентам данного предприятия ознакомиться с каталогом продукции, оформить заказ выбранного наименования, позволит сформировать у клиента или будущего партнера представление о компании и пути ее развития. Для администратора веб-сайта предусмотрены возможности управления каталогом продукции (создание новой единицы, редактирование или удаление существующей), ведение новостной страницы, управление заказами, поступающими с веб-сайта, управление профилями сотрудников. Функционал веб-приложения представлен на рисунке 1.



Рисунок 1 – Вариант использования веб-приложения

Проект реализован в виде веб-приложения, подходящего для современных браузеров. Разработка производилась на языке программирования JavaScript с использованием технологии React и СУБД MongoDB [1]. React является фреймворком с большим количеством оптимизаций рендеринга, за счет чего пользователю придется меньше ждать загрузки страницы. СУБД MongoDB была выбрана в силу того, что веб-приложение оперирует данными разных форматов, а данная СУБД как раз обеспечивает сохранение данных в формате документов вместо форматов реляционного типа [2].

#### Список использованных источников:

1. Плюсы и минусы React [Электронный ресурс]. – Режим доступа: <https://habr.com/ru/post/476312/>
2. Особенности MongoDB [Электронный ресурс]. – Режим доступа: <https://mcsjournal.ru/blog/osobennosti-mongodb-kogda-baza-dannyyh-vam-podhodit>

# КОНТРОЛЬ ЗА БАНКОВСКИМИ ОПЕРАЦИЯМИ

Матвеев И.Ю.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Грибович А.А. – ассистент каф. ПОИТ

**Аннотация.** Контроль имеет широкое определение как процесс, осуществляемый органом управления организации или другими сотрудниками, с целью получения информации относительно выполнения некоторых задач. [1]

В Республике Беларусь сложилась система банковского контроля, в целом соответствующая мировым стандартам. Она включает регистрацию и лицензирование банков и небанковских кредитно-финансовых организаций (банки), осуществление дистанционного надзора за банками и ОАО "Банк развития Республики Беларусь" (Банк развития) на основании отчетности, надзора на местах в виде инспекционных проверок, применение соответствующих мер надзорного реагирования к банкам и Банку развития в случае нарушения ими банковского законодательства, ухудшения их финансового состояния, реорганизацию и ликвидацию банков, системный анализ рисков банковского сектора, регулирование и контроль в сфере предотвращения легализации доходов, полученных преступным путем, финансирования террористической деятельности и финансирования распространения оружия массового поражения, осуществление валютного контроля.[2]

В Банках организована эффективно действующая система внутреннего контроля, призванная обеспечивать надлежащий уровень финансовой надежности и информационной безопасности, соответствующий характеру и объемам осуществляемых банковских операций и иной деятельности.

Система внутреннего контроля - совокупность внутреннего контроля, внутреннего аудита, организационной структуры, локальных правовых актов, определяющих стратегию, политику, методики и процедуры внутреннего контроля, а также полномочий и ответственности органов управления Банка и должностных лиц.

Основными целями системы внутреннего контроля являются:

- Контролирование обеспечения эффективности результативности и хозяйственной деятельности Банка при совершении банковских операций и других сделок, эффективности управления активами и пассивами, включая обеспечение сохранности активов и вложений Банка
- Контролирование достижения стратегических целей;
- Контролирование распределения полномочий и ответственности, а также исключения конфликта интересов в деятельности Банка;
- Внутренний контроль за организацией работы по предотвращению легализации доходов, полученных преступным путем, финансирования террористической деятельности и т.п.

К банковским операциям относятся:

- привлечение денежных средств физических и (или) юридических лиц на счета и (или) во вклады (депозиты);
- размещение указанных в абзаце втором настоящей части привлеченных денежных средств от своего имени и за свой счет на условиях возвратности, платности и срочности;
- открытие и ведение банковских счетов физических и (или) юридических лиц;
- открытие и ведение счетов в драгоценных металлах;
- осуществление расчетного и (или) кассового обслуживания физических и (или) юридических лиц, в том числе банков-корреспондентов;
- валютно-обменные операции;
- купля-продажа драгоценных металлов и (или) драгоценных камней в случаях, предусмотренных Национальным банком;
- привлечение и размещение драгоценных металлов и (или) драгоценных камней во вклады (депозиты) физических и (или) юридических лиц;
- выдача банковских гарантий; [3]

Департамент аудита является частью постоянного мониторинга системы внутреннего контроля, поскольку обеспечивает независимую оценку адекватности установленных правил и процедур, их соблюдения и подготовку предложений по повышению эффективности системы внутреннего контроля.

**Список использованных источников:**

1. Internal Control — Integrated Framework [Электронный ресурс] – Режим доступа: <https://www.coso.org/IC-IntegratedFramework-summary.htm>
2. Статья «Система внутреннего контроля» [Электронный ресурс] – Режим доступа <https://www.priorbank.by/priorbank-main/priorbank-today/bank-management/about-sistema-vnutrennego-kontrola>
3. Статья «Банковские операции. Иная деятельность, осуществляемая банками и небанковскими кредитно-финансовыми организациями» [Электронный ресурс] – Режим доступа: [https://kodeksy-by.com/bankovskij\\_kodeks\\_rb/14.htm](https://kodeksy-by.com/bankovskij_kodeks_rb/14.htm)

# ИСКУСТВЕННЫЙ ИНТЕЛЛЕКТ ДЛЯ ПРОГНОЗИРОВАНИЯ ЦЕНЫ НА ВАЛЮТУ

Шурукин Ю.А.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Грибович А.А. – ассистент кафедры ПОИТ

В данной работе рассмотрены возможности прогнозирования цены на валюту

Данная работа предназначена для прогнозирования цены на валютные пары, используя различные метрики и способы анализа графиков и тех же метрик. Основными способами анализа являются технический и фундаментальный анализ, которые в совокупности дают большую вероятность верного прогноза. Также для большей продуктивности программы в анализе валюты в течение времени участвуют нейронные сети.

Взаимодействие с пользователем происходит за счёт пользовательского десктопного приложения, написанного на технологии WPF с использованием языков C# и XAML. Также хранение, получение результатов работы программы производятся посредством взаимодействия с базой данных MSSQL с использованием процедурного расширения языка SQL - T-SQL. В случае недоступности базы данных, есть возможность взаимодействия с файловой системой. Получение данных о валюте происходит за счёт API торговых бирж.

Основные факторы, влияющие на валютный курс – это решения центральных банков государств и международных организаций в области кредитно-денежной политики относительно выпускаемой ими валюты. Несмотря на то, что управление курсом обычно не является задачей центробанков, любое изменение количества денег в обращении – это постоянно используемый инструмент воздействия со стороны центральных банков – приводит к изменениям курса валюты.

Особенность торговли валютой заключается в том, что она происходит круглосуточно. Начало – в раннее утро понедельника, конец – ночь пятницы. Торговля ведётся без остановок за счет того, что торги открываются в Новой Зеландии, далее подключаются биржи Австралии, Японии, потом Европы и США. Непрерывность торговли достигается за счет того, что в любое время суток хоть какая-то крупная биржа работает.

Основное, от чего зависит на международном рынке курс двух самых популярных валют – доллара и евро - это, конечно, политика центробанков и экономические показатели США и Евросоюза. При этом на курс доллара США, а также евро как на ведущие резервные валюты мира ориентируется огромное количество стран, держащих в них валютах часть своих резервов, – поэтому сам статус этих валют не предусматривает резких колебаний или стремительных взлётов или падений.

С другими валютами возможны и качественно другие ситуации, когда на валюту влияют решения и заявления внешних стран. Это так называемые сырьевые валюты, которые зависят от экспорта природных ресурсов своими странами. Классическим примером является нефть: один из факторов, от которых зависит курс рубля, – это мировая цена на этот ресурс. Существенную часть мирового предложения нефти (около 50%) контролирует международный картель стран-экспортеров нефти – ОПЕК, и его решения об увеличении или снижении добычи колеблет цену барреля, укрепляя или ослабляя тем самым российский рубль, иранский риал, венесуэльский боливар и другие «нефтяные» валюты.

Наконец, курс валют может колебаться под влиянием валютных спекуляций. При этом такой эффект обычно кратковременный и быстро нивелируется, а также требует огромного объема операций для того, чтобы как-либо значимо «подвинуть» валютных курс.

На текущий день трейдеры в странах СНГ и всего остального мира используют свои варианты прогнозирования цены на валюту:

- Приложения с различными сложностями метрик;
- Приложения, разработанные под конкретную методику прогнозирования;
- Боты с различными простыми математическими уравнениями.

#### Список использованных источников:

- 1) *Binance Academy* [электронный ресурс]. – Электронные данные. – Режим доступа: <https://academy.binance.com/ru/>.
- 2) *Wiki* [электронный ресурс]. – Электронные данные. – Режим доступа: <https://ru.wikipedia.org/>.
- 3) *Министерство по налогам и сборам РБ* [электронный ресурс]. – Электронные данные. – Режим доступа:

<http://www.nalog.gov.by/ru/>



# ВЕБ-ПРИЛОЖЕНИЕ ПО РАБОТЕ С ОБРАЩЕНИЯМИ ГРАЖДАН НА ПЛАТФОРМЕ JAVA

*Дубровинский А.В.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Можей Н.П. – канд. физ.-мат. наук, доцент кафедры ПОИТ*

В данной работе описывается веб-приложение, позволяющее гражданам авторизоваться в системе, создать, удалить либо редактировать обращение, просмотреть существующие категории обращений, а также подписать их.

Социальные сети, мессенджеры и группы позволяют людям взаимодействовать между собой, однако таким способом они не могут непосредственно обратиться к министрам, исполнительным органам власти и чиновникам. Для этого существуют платформы в сети Интернет для коллективных обращений. Если человек знает, что такое коллективное обращение, образец этого документа, причины подачи, то ему гораздо проще отстаивать свои права. Обращение, сформированное гражданами, подаётся по абсолютно разным причинам в зависимости от рода деятельности [1].

В последнее время граждане все чаще выбирают электронный формат подачи обращения. Это связано с развитием удобных онлайн инструментов, простым поиском единомышленников и возможностью быстро и просто создавать обращение из дома [2]. По этим же причинам было разработано веб-приложение по работе с обращениями граждан.

При разработке программного продукта использовались технологии Java, Spring и MySQL. Язык Java превратился из просто универсального языка в целую платформу и экосистему, которая объединяет различные технологии, используемые для целого ряда задач: от создания десктопных приложений до написания крупных веб-порталов и сервисов. Поэтому из всех языков программирования был выбран именно он.

В основе реализации приложения лежит технология Spring Framework. Spring обеспечивает решения многих задач, с которыми сталкиваются Java-разработчики и организации, которые хотят создать информационную систему, основанную на платформе Java. Spring известен как источник расширений, нужных для эффективной разработки сложных бизнес-приложений. Этот фреймворк предлагает последовательную модель и делает её применимой к большинству типов приложений, которые уже созданы на основе платформы Java [3]. Эта технология нашла свое применение и в реализованном программном средстве.

Основными сущностями, которые формируют информационную модель приложения, являются пользователь, обращение, категория, подпись и фотография. Данные сущности хранятся в базе данных MySQL, которая давно конкурирует с другими известными системами управления базами данных (СУБД). Чаще всего ее используют при разработке веб-решений, что объясняется тесной интеграцией с популярными языками программирования и высокими показателями скорости. MySQL легко устанавливается, имеет понятный интерфейс, включает в себя практически весь необходимый набор инструментов, который может пригодиться при разработке любого проекта [4].

К основным функциям, которые выполняет разработанное приложение, относятся регистрация и авторизация пользователей, создание, удаление и редактирование обращений, просмотр списка обращений выбранной категории, подписание обращений и прикрепление к ним фотографий. Помимо основных функций разработанное веб-приложение включает дополнительные функции, позволяющие обладать преимуществами перед аналогами программного продукта. К ним относятся наличие руководства для написания обращения, содержание шаблонов заявлений для различных ситуаций, а также возможность делиться ссылками на обращения в социальные сети.

В настоящее время все больше граждан хотят своим голосом и общественной деятельностью менять ситуацию в стране, а также улучшить положение животных, состояния окружающей среды и т.д. Веб-ресурсы с каждым годом становятся еще популярнее для таких целей, поэтому вышеописанное веб-приложение для работы с коллективными обращениями является актуальным и может помочь гражданам в воплощении их идей и намерений.

## **Список использованных источников:**

1. Коллективные обращения граждан в администрацию [Электронный ресурс]. – Электронные данные. – Режим доступа: <https://fabrikaklientov.ru/info/kollektivnye-obrashhenija-grazhdan-v/>.
2. Электронные коллективные обращения граждан в контексте прав человека [Электронный ресурс]. – Электронные данные. – Режим доступа: [http://sympa-by.eu/sites/default/files/library/sympa\\_etitions\\_full.pdf](http://sympa-by.eu/sites/default/files/library/sympa_etitions_full.pdf).
3. Spring Framework [Электронный ресурс]. – Электронные данные. – Режим доступа: [https://ru.wikipedia.org/wiki/Spring\\_Framework](https://ru.wikipedia.org/wiki/Spring_Framework).
4. Что такое MySQL [Электронный ресурс]. – Электронные данные. – Режим доступа: <https://mchost.ru/articles/chto-takoe-mysql/>.

*57-я Научная Конференция Аспирантов, Магистрантов и Студентов БГУИР,  
Минск, 2021*

# ПРОГРАММНО-АППАРАТНЫЙ КОМПЛЕКС ДЛЯ ДИСТАНЦИОННОГО УПРАВЛЕНИЯ СОПРЯЖЕННЫМИ ФИЗИЧЕСКИМИ УСТРОЙСТВАМИ

*Николаев Д. С.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

**Видничук В.Н. – м.т.н**

В данной работе предлагается программно-аппаратный комплекс на основе микроконтроллера ESP8266 и мессенджера Telegram для управления сопряженными электромеханическими устройствами через интернет. В качестве практического применения рассматривается управление микроклиматом теплицы для выращивания урожая, а также автоматизация этого процесса.

Подключенные друг к другу устройства существовали со времен появления первых компьютерных сетей и бытовой электроники. Однако пока не появился Интернет, никому не приходило в голову, что связь может быть глобальной. В 1990х исследователи выстраивали теоретические построения о том, как сочетание человека и машины даст совершенно новую форму коммуникации и взаимодействия [1]. Теперь же это стало повседневной реальностью.

В качестве практического применения дистанционного управления периферией рассматривается решение проблемы по реализации микроклиматического контроля теплицы для выращивания урожая. Например, закрытая на весь день поликарбонатная теплица в солнцепек может “сварить” весь урожай. Если хозяина не будет на участке, то он понесет убытки. Решением проблемы будет являться возможность хозяина контролировать климатические параметры внутри теплицы, управлять её проветриванием и поливом. Логические выводы микроконтроллера могут быть использованы как для управления электромеханическими реле, которые, в свою очередь, будут активировать моторы для открытия дверей и активации полива, так и для считывания показателей атмосферного давления, температуры и влажности воздуха с датчиков. Разработанное устройство позволяет устанавливать верхнюю и нижнюю границу температуры воздуха. По превышении верхней границы температуры активируются реле для открытия дверей проветривания. Если же температура воздуха опустится ниже нижней границы – проветривание будет остановлено путем закрытия дверей. Данные установки будут работать при активации автоматического режима работы. Также пользователь может перейти в ручной режим и управлять периферией самостоятельно.

Ключевой особенностью данного проекта является многопользовательское взаимодействие с микроконтроллером через Интернет посредством мессенджера Telegram, используя технологию Telegram Bot API [2]. Устройство выступает в качестве собеседника, которому можно отдавать команды и получать в ответ отчет о статусе производимых операций. Пример такого взаимодействия проиллюстрирован на рисунке 1:

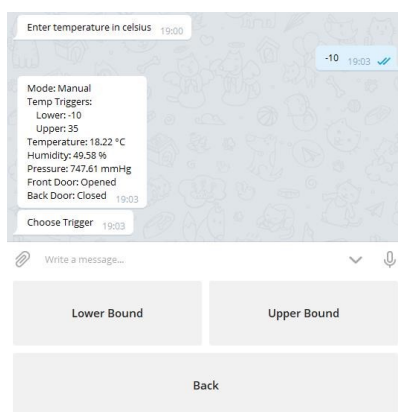


Рисунок 1 – Пользовательский интерфейс для взаимодействия с устройством, установка нижнего порога температуры

Одно физическое устройство может сочетать в себе несколько логических устройств. Гибкая архитектура ПО позволяет использовать незадействованные выводы микроконтроллера для иных логических устройств для максимальной утилизации потенциала микроконтроллера, тем самым экономя финансовые ресурсы. Каждое логическое устройство имеет собственную иерархию меню для конфигурирования его параметров. Данный подход позволяет эффективно расширять функционал устройства.

**Список использованных источников:**

1. С. Грингард, *Интернет вещей: Будущее уже здесь*, Изд-во: Альпина Паблишер, 2016. – 188 с.
2. *Telegram Bot API [Электронный ресурс]*. – Электронные данные. – Режим доступа: <https://core.telegram.org/bots/api>

# ПРОГРАММНОЕ СРЕДСТВО ОРГАНИЗАЦИИ ПАРКОВОЧНОЙ ДЕЯТЕЛЬНОСТИ

*Терлецкий А.А.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Хмелева А.В. – доцент каф. ПОИТ*

В настоящее время существует множество компаний, работающих в транспортной сфере, а именно открывающие автомобильные парковки.

Компании, занимающиеся оказанием услуг в сфере автомобильной парковки, сталкиваются с проблемой ведения и регистрации автомобильных владельцев, так как это осуществляется на бумажных носителях.

Уровень безопасности данных от несанкционированного доступа очень низок и в случае потери информации восстановление практически невозможно. В случае если автомобильная парковка имеет большую площадь, то на ее обслуживание требуется большое количество денежных затрат.

Целью проекта является разработка системы автоматического управления парковкой для облегчения внесения и учета данных, а также упрощения ведения документооборота и сокращения затрат.

Одним из важнейших элементов данного программного средства является система распознавания номеров автомобиля.

В качестве примера была выбрана система оптического распознавания символов (OCR).

Задача распознавания текстовой информации при переводе печатного и рукописного текста в электронную форму является одной из важнейших составляющих любого проекта, имеющего целью автоматизацию документооборота или внедрение безбумажных технологий. Вместе с тем эта задача является одной из наиболее сложных и наукоемких задач полностью автоматического анализа изображений.

В последние десятилетия, благодаря использованию современных достижений компьютерных технологий, были развиты новые методы обработки изображений и распознавания образов, благодаря чему стало возможным создание таких промышленных систем распознавания печатного текста, которые удовлетворяют основным требованиям систем автоматизации документооборота. Тем не менее, создание каждого нового приложения в данной области по-прежнему остается творческой задачей и требует дополнительных исследований в связи со специфическими требованиями по разрешению, быстродействию, надежности распознавания и объему памяти, которыми характеризуется каждая конкретная задача.

Системы OCR состоят из следующих основных блоков, предполагающих аппаратную или программную реализацию:

- блок сегментации (локализации и выделения) элементов текста;
- блок предобработки изображения;

- блок выделения признаков;
- блок распознавания символов;
- блок постобработки результатов распознавания.

Эти алгоритмические блоки соответствуют последовательным шагам обработки и анализа изображений, выполняемым последовательно.

Сначала осуществляется выделение текстовых областей, строк и разбиение связных текстовых строк на отдельные знакоместа, каждое из которых соответствует одному текстовому символу.

После разбиения (а иногда до или в процессе разбиения) символы, представленные в виде двумерных матриц пикселей, подвергаются сглаживанию, фильтрации с целью устранения шумов, нормализации размера, а также другим преобразованиям с целью выделения образующих элементов или численных признаков, используемых впоследствии для их распознавания.[1]

Распознавание символов происходит в процессе сравнения выделенных характерных признаков с эталонными наборами и структурами признаков, формируемыми и запоминаемыми в процессе обучения системы на эталонных и/или реальных примерах текстовых символов.

На завершающем этапе смысловая или контекстная информация может быть использована как для разрешения неопределенностей, возникающих при распознавании отдельных символов, обладающих идентичными размерами, так и для корректировки ошибочно считанных слов и даже фраз в целом.

*Список использованных источников:*

1. Ю.В. Визильтер, С.Ю. Желтов, А.В. Бондаренко, М.В. Ососков, А.В. Моржин. Обработка и анализ изображений в задачах машинного зрения: Курс лекций и практических занятий. – М.: Физматкнига, 2010. – 672 с.