

**АБДУЛАЕВ А., БОРУБАЕВ М.Ч.**<sup>1</sup>КГУСТА им. Н. Исанова, Бишкек, Кыргызская Республика**ABDULAEV A., BORUBAEV M.CH.**<sup>1</sup>KSUCTA n.a. N. Isanov, Bishkek, Kyrgyz Republic  
absamat\_abdulaev@mail.ru mir\_bo@mail.ru**РАЗРАБОТКА ИНФОРМАЦИОННОЙ СИСТЕМЫ СРЕДСТВАМИ ASP.NET****DEVELOPMENT OF INFORMATION SYSTEM BY MEANS OF ASP.NET**

*Көпчүлүк мамлекеттик мекемелер жана ишканалар үчүн республика боюнча филиалдар ортосунда бөлүштүрүлгөн бизнес-процессти автоматташтыруу көйгөйлөрү бар. Бул макалада аймактык бөлүштүрүлгөн филиалдарды башкаруу үчүн тутумду долбоорлоо маселеси каралган. Филиалдардын тарамын башкаруу маалыматтык тутуму үчүн бизнес-процессти башкаруу жана автоматташтыруу тутумун иштеп чыгуу жана тутумду иштетүүнүн жалпы наркын минималдаштыруу процесси баяндалган. «Кыргыз почтасы» мамлекеттик ишкананын филиалдарында жол кыймылынын эрежелерин административдик укук бузуулары үчүн төлөмдөрдү жүргүзүү үчүн маалыматтык тутумун долбоорлоо мисалы келтирилген.*

**Өзөк сөздөр:** маалыматтык тутум, программалык камсыздоо, веб-тиркеме, кирүү технологиясы, берилмелер базасы, контроллер, тутумду жайгаштыруу, административдик укук бузуу, жол кыймылынын эрежеси.

*Для большинства государственных учреждений и предприятий существует проблема автоматизация бизнес-процесса, которая распределена между филиалами по всей республике. В статье рассмотрена задача проектирования системы для управления территориально распределенными филиалами. Описан процесс разработки системы управления и автоматизации бизнес-процесса для информационной системы управления сетью филиалов и минимизации суммарной стоимости эксплуатации системы. Приведен пример проектирования информационной системы для управления приемами платежей за административные правонарушения ПДД в филиалах государственного предприятия «Кыргыз почтасы».*

**Ключевые слова:** информационная система, программное обеспечение, веб-приложение, технология доступа, база данных, контроллер, развертывание системы, административное правонарушение, правила дорожного движения.

*For most state institutions and enterprises, there is a problem of automation of the business process, which is distributed among branches throughout the republic. The article discusses the problem of designing a system for managing geographically distributed branches. The process of development of the business process management and automation system for the branch network management information system and minimization of the total operating cost of the system is described. An example of designing an information system for managing payments for administrative violations of traffic rules in branches of the Kyrgyz Post State Enterprise is given.*



**Key words:** *information system, software, web application, access technology, database, controller, system deployment, administrative offence, traffic rules.*

**Введение.** Для оптимизации управления и минимизации расходов большинство государственных учреждений и предприятий применяют процесс автоматизации. Существуют множество проблем связанных с территориальной распределенностью филиалов предприятия – это отсутствие каналов связи, ограниченность количества персонала и нехватка квалифицированных специалистов на местах. Для решения этой проблемы, администрация использует автоматизацию бизнес-процесса на основе архитектуры «клиент-сервер». Актуальные решения подобных проблем для государственных организации, которые существуют на рынке Кыргызской республики как принято не учитывают структурную специфику организации и обычно направлены на решения конкретной задачи бизнес-процесса предприятий. Внедренное решение позволяет руководству создавать, просматривать и управлять событиями бизнес-процесса внутри корпоративной сети. Корпоративная информационная система призвана повысить потенциал предприятия. В качестве примера работы была выбрана разработка программы «Прием платежей за административные нарушения ПДД КР» в государственном предприятии «Кыргыз почтасы» [4]. Основное назначений разработанного приложения – это ведение оперативного учёта и понижение расходов по введению конкретного бизнес-процесса.

Услуга “Прием платежей за административные нарушения ПДД КР» для предприятия была новой и существующие решения не удовлетворяли требованиям создаваемого бизнес-процесса.

**Цель исследования.** Применение одной из современных систем веб-разработки для оптимизации управления бизнес-процессом предприятия в территориально распределенной среде.

**Постановка задачи.** Разработка автоматизированной информационной системы для оптимизации управления приемами платежей за административные штрафы ПДД по всей республике филиалами ГП «Кыргыз Почтасы». Важно, чтобы данное решение производилось с учетом территориальной распределенности предприятия, архитектуры существующего информационного отдела и кадровой недостаточностью персонала.

**Новизна исследований.** Разработано веб-приложение в информационной системе предприятий с применением средств ASP.NET в целях развития управления услугами принятия платежей подразделениями ГП «Кыргыз Почтасы».

**Выбор программного продукта.** При решении задачи построений корпоративной информационной системы особую роль играет выбор основных программных продуктов, их модульность, что позволяет гибко реагировать при условии изменения бизнес-процессов. Данная многопараметрическая задача и ее решения должна удовлетворять всем потребностям предприятия и должна как принято привести к оптимизации любых производственных задач, которая должна включать материальные расходы включая приобретения аппаратного и программного обеспечения. Несомненным составляющим таких затрат также является сопровождения информационной системы охватывающим процесс обучения персонала задействованного в эксплуатации данной системы.

Первым следует учитывать существующую архитектуру сети организации, чтобы разрабатываемая система не повлияло на работу других информационных систем и объединяло все филиалы. Корпоративная сеть между филиалами предприятия приведена на Рис. 1.

Основным вторым критерием выбора является учет опыта эксплуатации используемого программного обеспечения и возможность интеграции с этими решениями, а может быть и их замена, которая должна происходить в полном соответствии с существующей бизнес-правилом.



По результатам проведенного исследования на рынке республики не было найдено решений, способных предоставить необходимый функционал для ведения учёта приема платежей за административные штрафы, поскольку бизнес процесс был новым. Также необходимо учитывать тот факт, что окружающий мир стремительно изменяется – вместе с ним изменяются и бизнес-процессы, которых становится всё больше, а их сложность увеличивается, в связи с этим требуется автоматизация и поддержка данных бизнес-процессов.

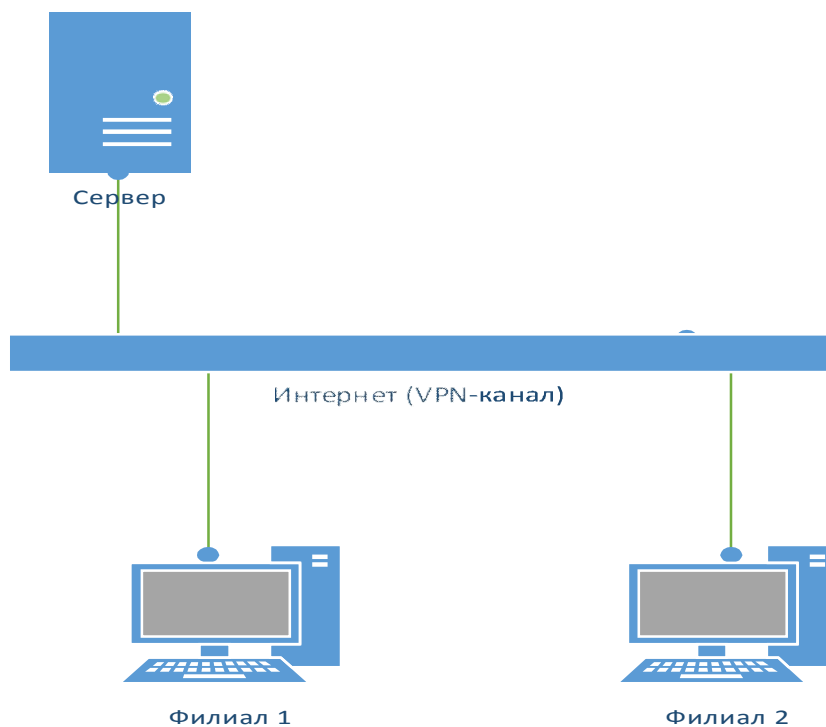


Рис. 1. Сеть государственного предприятия

**Требования к структуре программного обеспечения.** Как было сказано для решения проблемы системной работы географически отдаленных филиалов подходящей архитектурой является клиент-серверная. Серверная часть архитектуры должна работать под управлением серверных операционных систем и предпочтительным в данном случае является операционная система на баз семейства Windows. Windows является предпочтительным поскольку технология, использованная для базы данных, наиболее подходит для СУБД MS SQL Server 2012.

В качестве клиентской компоненты должен использоваться браузер, которая не требует дополнительного персонала и каких-либо затрат.

#### **Выбора архитектуры решения**

**Выбор настольного приложения** не рассматривался, поскольку недостатками такого варианта являются потребность, моральное устаревание настольных приложений, необходимость установки и конфигурирования определенного программного обеспечения на местах, что требует дополнительных сотрудников на местах.

**Разработка веб-приложения.** С учетом вышеперечисленных проблем на наш взгляд наиболее подходящим решением, которое позволит уменьшить финансовые затраты получается архитектура веб-приложения. Преимущества очевидны, поскольку взаимодействия между клиентом и сервером происходит на основе протокола HTTP, что в данном случае для всех межсетевых экранов прозрачно и работа клиента, которая представляет собой веб-браузер, происходит минимальными правами. А это безопасность, которая является основным требованием системных администраторов предприятия. Веб-



приложение отвечает за фронт и за формирование пользовательского интерфейса с помощью которой происходит непосредственное взаимодействие с базой данных. В данном случае веб-приложение представляет собой слой, которая занимается обработкой данных с учетом всех требований бизнеса, а сами результаты расчетов и сходные данные для расчета находятся на слое модели, т.е. в СУБД. СУБД отвечает за сохранность и контроль доступа к информации, хранящейся в ней.

Все это позволяет перечислить преимущества, которые заключаются в следующем:

- архитектура информационной системы, основанная на Веб-приложении, является кроссплатформенной, что означает возможность работы на любых устройствах и операционных системах для конечного пользователя;

- несомненным преимуществом веб-ориентированной системы – это работа без всяких установок, настроек со стороны системного администратора, а также наличие или установка дополнительного программного обеспечения и отсутствие требований к аппаратному обеспечению на стороне клиента;

- Веб-технология, позволяет получать доступ и работать 24 часа в сутки с приложением фактически из любой точки, где есть доступ к сети Интернет;

- клиентами веб-приложения могут быть не только настольные компьютеры, но и любые телефоны и планшеты, что немаловажно для предприятия, филиалы которых могут располагаться в труднодоступных областях республики, где каналом связи служат мобильные сети.

Одним еще важным преимуществом является обновление и расширения системы, которая делается незаметно и быстро для клиентов.

По результатам системного анализа предметной области и организационной структуры предприятия была сформирована инфологическая модель о системе и ее предназначении. Разработано техническое задание, включающее весь процесс разработки системы.

**Среда разработки.** Microsoft Visual Studio 2013 Express использовалась как среда разработки, поскольку она обладает отличными возможностями и поддержкой широких технологии для создания веб-приложений. Visual Studio предлагает для разработчиков общедоступные, бесплатные инструменты для создания и развертывания современных веб-приложений. К этим средствам можно отнести: язык представления HTML5, каскадные стили CSS3, множество языков программирования, платформа ASP.NET, OPM-EF, инструмент NuGet и многое другое.

**Выбор СУБД.** Основой корпоративной информационной системы является система управление базой данных. Важно выбрать такую СУБД, которая не только в полной мере удовлетворяет текущим потребностям, но и имеет необходимый «запас прочности» для дальнейшего расширения и интеграции. Выбор среды разработки Microsoft Visual Studio 2013 Express, регламентирует использовать максимально все возможности СУБД Microsoft SQL Server 2012 Express, которая также входит в семейство технологий от Microsoft и лучше других гарантирует безошибочную работу и поддержку технологий для работы с базами данных.

Отличительным свойством данного СУБД является удобство пользовательского интерфейса, а также наличие утилит администрирования. Все это дополняется содержанием высокой производительностью и относительно невысокой стоимостью эксплуатации.

**Реализация.** Разработка информационной системы как было отмечено выше было в интегрированной среде разработки Visual Studio 2013. Тип приложения был выбран как ASP.NET MVC 4. Для работы с базой данных использовался технология ORM (object relation mapping)- объектное отображения данных - ADO.NET Entity Framework. База данных было создано в СУБД MS SQL SERVER 2012. Рассмотрим эти технологии подробнее, отметив их преимущества.



C# - это объектно-ориентированный язык программирования, который был и создан для применения на платформе Microsoft .NET Framework, что является преимуществом. На сегодняшний день, он является одним из наиболее популярных языков и входит в первую десятку. Для разработки больших корпоративных информационных систем с поддержкой масштабируемости и возможностью работы со сложными базами данных язык C# подходит полностью. Visual Studio позволяет использовать два основных подхода к созданию современных веб-приложений:

- Классическое приложение на основе Web-Form с использованием шаблона ASPX. Каждая страница представляет собой страницу формата ASPX. И этот файл имеет класс, который программируется языком C# и в конечном итоге управляет генерацией конечной HTML страницы.

- MVC. Паттерн, который представляет собой подход разделения бизнес-логики и интерфейса пользователя. Этот подход позволяет разработчикам легко и одновременно изменять отдельные части приложения, не затрагивая другие части.

Для разработки корпоративной системы в качестве шаблона была выбрана технология MVC, учитывая, что она более современна и проста для реализации. MVC позволяет при необходимости модифицировать приложение, делать оптимизации одних компонент не затрагивая функциональность других. Кроме того, MVC является легковесной платформой отображения с широкими возможностями UNIT тестирования и интегрирована с существующими функциями ASP.NET. Паттерн архитектуры Model-View-Controller (MVC) разделяет логику приложения на три основных компонента: модель, представление и контроллер.

- **Модель.** Модели являются виртуальными представления реально существующих объектов бизнес процесса, которые охватывают бизнес логику. Состояние объектов моделей приложения получают, а в конце работы и сохраняют в базе данных. Например, объект Raion может получать информацию из базы данных, работать с ней, а затем записывать обновленные данные в таблицу Raions базы данных SQL Server.

- **Представления.** Представления служат для отображения пользовательского интерфейса приложения, а именно состояние моделей. Пользовательский интерфейс обычно создается на основе данных модели. Примером может служить представление для редактирования таблицы Raions, которое содержит текстовые поля и раскрывающиеся списки, значения которых основаны на текущем состоянии объекта Raion.

- **Контроллеры.** Контроллеры осуществляют работу с моделью в базе данных, а также выбор представления, для отображения пользовательского интерфейса. В приложении MVC представления только отображают данные модели, а контроллер обрабатывает вводимые данные и отвечает на действия пользователя и при необходимости обращается к базе данных.

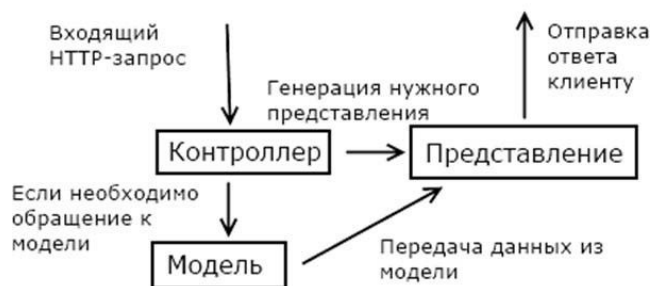


Рис. 2. Шаблон разработки MVC

Данная схема указывает расположение каждого вида слоя логики в приложении. Интерфейс располагается в представлении. Логика взаимодействия располагается в контроллере. Бизнес-логика находится в модели. Это разделение позволяет работать со



структурами любой сложности, поскольку обеспечивает одновременную реализацию только одного аспекта. Например, разработчик может сконцентрироваться на создании представления отдельно от бизнес-логики.

**Проектирование системы управления доступом.** Управление доступом к данным будет осуществляться на основе ролей. В системе должны быть доступны две роли: «Оператор» и «Администратор». Оператор обладает доступом к ведению платежа и получения отчетов. Администратору системы доступны максимальные возможности. Ниже перечислены функциональные требования, выявленные в процессе анализа требований к разрабатываемой системе:

1. Система должна предоставлять возможность работы со справочниками.
2. Система должна позволять принимать штрафы по административным нарушениям ПДД.

3. Система должна обладать возможностью генерации отчетов.

**Технология доступа к данным.** В процессе обработки и хранения данных, и корректного их отображения диаграмма при генерации виртуального мира реального бизнес-процесса используются такие технологии доступа к данным базы данных как ADO.NET Entity Framework и LinqToSql. При использовании Entity Framework (EF) у разработчика нет необходимости в написании большей части кода для запросов и доступа к данным, а в LinqToSql это требуется. Поэтому мы использовали ADO.NET Entity Framework с паттерном разработки Code First. В ходе генерации моделей в таблицу были созданные таблицы, которые приведены на рисунке 4.

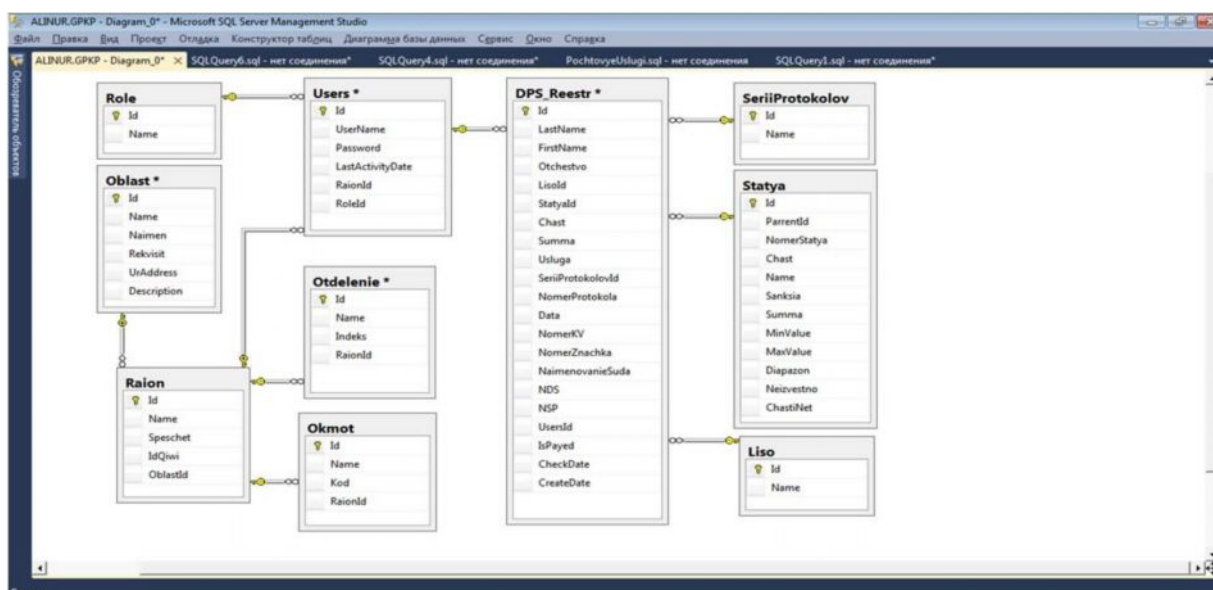


Рис. 4. Схема базы данных

Основу структуры системы составляют девять контроллеров:

- «Home» – для управления данными, доступными неавторизованному пользователю;
- «Account» – для реализации логики авторизации и аутентификации;
- «Seria» – для реализации справочника по сериям протоколов;
- «Statya» - для реализации справочника статей по административному кодексу;
- «Oblast», «Raion», «Otdelenie», «Okmot» - для реализации справочников по областям, районам, отделениям и айыл окмоту.
- «ProtokolService» – отвечающий за добавление, удаление и редактирование данных об принятых платежах за административные правонарушения. Контроллер также позволяет получать различные виды отчетов за различные периоды времени.



Контроллер *Home* взаимодействует с тремя «полными» представлениями: «Index», «About», «Contact». Эти представления представляют такие базовые данные о приложении как: информация с главной страницы («Index»); краткую информацию о возможностях программы («About»); информацию для обратной связи («Contact»). Именно эти представления доступны пользователю системы с ролью «Гость».

Контроллер-справочники взаимодействует с моделями в базе данных и позволяют совершать стандартные операции «Добавление», «Редактирование» и «Удаление». Кроме того каждый из них имеет представления «Index», который служит для просмотра справочников.

«ProtocolService» служит главным контроллером, в котором происходит процесс оплаты и получения квитанции. Здесь генерируются также реестра за промежутки времени, по которым производятся различные отчисления по налогам и в местные бюджеты.

Для ускорения работы многие представления используют так называемые «частичное представление», т.е. для выполнения асинхронных запросов.

Кроме того, разработаны представления, хранящиеся в папке Shared:

- «LayerControl» - визуализация классической мастер страницы.
- «LoginPartial» – визуализация области-шапки, для отображения логотипа организации и кнопок для перехода к представлению авторизации или выхода из текущего сеанса.

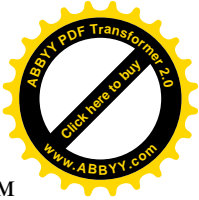
**Развертывание системы.** В результате реализации функциональных возможностей системы и ее интерфейса получено готовое MVC приложение. Данное приложение доступно лишь на локальном компьютере, а в производственных проектах оно представлено широкой публике.

Следующий шаг - развертывание созданного веб-приложения на сервере. Это развертывание необходимо для того, чтобы оно было доступно для клиентов из браузера. При этом один из серверов предприятия использован в качестве сервера. Сервер приложений IIS (Internet Information Services), предоставляющий множество функциональных средств используется для развертывания приложения, реализованных на платформе ASP.NET. Среди них наиболее важным является поддержка хостинга приложений ASP.NET. На сервере используются операционная система Windows Server 2008 R2 и программная платформа .NET Framework 4.5. Компонент IIS был включен как часть установки Windows. В дальнейшем необходимо было лишь активизировать его и верно сконфигурировать. На сервере была сконфигурирована база данных для приложения. Для ее реализации на сервер была установлена СУБД Microsoft SQL Server 2012 Express. При этом на рабочем компьютере разработана и восстановлена резервная копия базы. Затем были настроены учетные записи входа на сервер базы данных и настроены необходимые права доступа.

После запуска IIS можно приступать к развертыванию. В данной работе рассмотрено развертывание с помощью пакета приложения в качестве пути развертывания. Был создан и сконфигурирован пул приложений. Далее были настроены ограничения ISAPI и CGI для разрешения выполнения сборок платформы .NET Framework 4.

Следующий шаг - установка пакета развертывания «Web Deploy 3.5», приобретенный с официального сайта компании Microsoft. Пакет интегрируется с IIS и предоставляет необходимые для развертывания инструменты, обеспечивает перенос пакетов приложения. Затем на рабочей станции, на которой велась разработка системы, с помощью встроенных средств Visual Studio был создан пакет приложения «Web Deploy». Созданный пакет был скопирован на сервер и импортирован в IIS. После импорта приложения в файле «web-config» было настроено подключение к базе данных.

**Заключение.** Услуга приема платежей за административные штрафы предприятия находится в периоде изменений. Государственным предприятиям следует принимать дополнительные условия, которые обеспечат конкурентоспособность и развитие предприятий. Развивая процесс управления приема платежей, необходимо учитывать, что



информационные технологии развиваются каждый день, и использование систем автоматизации может принести предприятия много преимуществ, и веб-разработка- является одной из важных составляющих процесса управления бизнес-процессом в территориально распределенных предприятий.

### Список литературы

1. Таненбаум Э. Распределенные системы. Принципы и парадигмы [Текст] / Таненбаум Э., Ван Стеен М. - СПб: Питер, 2003. - 845 с.
2. Josuttis N.M. SOA in Practice: The Art of Distributed System Design (Theory in Practice). O'Reilly Media, Inc., 2007. – 344 p.
3. WCF 4: Windows Communication Foundation и .NET 4 для профессионалов / Сибраро П., Клайс К., Косолино Ф. и др. - М.: Диалектика, 2011. - 464 с.
4. А.с. № 544. Автоматизированная информационная система «Регистрация административных протоколов главного управления организации безопасности дорожного движения Кыргызской Республики» / Борубаев М.Ч., Абдулаев А., 2019. – 2 с.
5. Шаршенбаева А.К. Развитие облачных технологий и особенности их применения при разработке автоматизированных систем [Электронный ресурс] / А.К. Шаршенбаева / Вестник КГУСТА, - 2019. - №1(63). – Режим доступа: <https://www.elibrary.ru/item.asp?id=41132098>
6. Искендерова М.Ж. Автоматизация управления учета основных фондов высших учебных заведений [Электронный ресурс] / М.Ж. Искендерова Вестник КГУСТА. – 2019. - №”(64). – Режим доступа: <https://www.elibrary.ru/item.asp?id=41415327>