

**МИНИСТЕРСТВО ВЫСШЕГО И СРЕДНЕ СПЕЦИАЛЬНОГО
ОБРАЗОВАНИЯ РЕСПУБЛИКИ УЗБЕКИСТАН
НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ УЗБЕКИСТАНА
ИМЕНИ МИРЗО УЛУГБЕКА**

**КАФЕДРА МАТЕМАТИЧЕСКОГО МОДЕЛИРОВАНИЯ И
КРИПТОАНАЛИЗА**

5330300 – ИНФОРМАЦИОННАЯ БЕЗОПАСНОСТЬ

Нуруллаев Зафар Фарход угли

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

На тему:

«Применения ЭЦП в системе Документообороте»

Выпускник: З.Нуруллаев

Руководитель:

Доц. М.У.Худойбергганов

Тошкент-2018

Содержание

Введение	3
I Глава Электронный документооборот.....	5
1.1 Понятие электронного документооборота.....	5
1.2 Электронный документ	10
1.4 Электронная цифровая подпись.....	12
II Глава. Qr-code в системе документооборота	16
2.1 Описания Qr-кода.....	16
2.2 Кодирование данных.....	17
2.2.1 Цифровое кодирование.....	17
2.2.2 Буквенно-цифровое кодирование.....	18
2.2.3 Байтовое кодирование	18
2.2.4 Кандзи	19
2.2.5 Добавление служебной информации	19
2.2.6 Разбиение на блоки.....	22
2.2.7 Создание байтов коррекции	23
2.2.8 Объединение информационных блоков.....	24
2.2.9 Этап размещения информации на поле кода.....	25
2.2.10 Занесение данных.....	28
III ГЛАВА СИСТЕМЫ ДОКУМЕНТООБОРОТА.....	30
3.1 Система Документооборота: Ijro.gov.uz.....	30
3.2 Описание интерфейса системы	32
Заключения	34
Литература.....	36
ПРИЛОЖЕНИЕ.....	37

Введение

С каждым годом, все больше и больше технологий автоматизируются. И в последнее время в нашей стране набирает популярность использование электронного документооборота взамен бумажного. Однако необходимо не забывать о защите передаваемых документов. Для этого обычно используется электронная цифровая подпись.

В первой главе, речь пойдет об электронном документообороте: появлении первых систем электронного документооборота. Будет рассказано об основных задачах, которые ставятся перед такими системами и их общей структуре. Также будет затронута тема правового регулирования данного понятия.

Во второй главе, будет рассказываться об электронной цифровой подписи: какими документами она регламентируется, какими свойствами обладает. Также будут показаны хронология появления различных стандартов ЭЦП и дана их характеристика.

Цель выпускной работы

Обеспечение безопасности передаваемых данных между организациями с помощью создание защищенных каналов связи обмена документов с шифрованием открытыми и закрытыми ключами.

Актуальность работы

Актуальностью этой задачи является защитой конфиденциальных данных в организациях.

Объект исследования.

Система защиты информации

Предмет исследования

Цифровое кодирование – защищенные каналы связи обмена данных между организациями. При решении поставленных задач использованы: статистические методы, методы нечеткой логики, что позволило выявить определенные закономерности.

I Глава Электронный документооборот

1.1 Понятие электронного документооборота

Бурное развитие информационных технологий, начавшееся в середине прошлого века, дало возможность перейти к электронному документообороту. Первые системы электронного документооборота (СЭД) появились в банковской сфере. В дальнейшем системы электронного документооборота стали широко применяться и для обмена другой коммерческой информацией.

В последние время подобные системы стали активно внедряться и в нашей стране: они уже используются в большинстве крупных банков и банковских объединений для межбанковского обмена и для работы с клиентами. Развитие компьютерных технологий наряду с постоянным увеличением объемов документов, которые необходимо обрабатывать, послужили толчком для развития СЭД. Среди преимуществ таких систем над традиционными (на бумажных носителях) можно выделить следующие:

- Увеличение скорости передачи и обработки информации. Использование СЭД позволяет не только увеличить скорость передачи информации между двумя сотрудниками, но и используя дополнительный инструментарий, систем тратить гораздо меньше времени на обработку входящих документов и поиск.

- Сокращение пространства необходимого для хранения информации. Помимо того, что войну и мир теперь можно хранить не в четырех томах на полке, а на жестком диске компьютера или вообще в облаке. Одновременно с этим существенно упрощаются такие операции, как поиск необходимой информации, копирование документов.

- Уменьшение издержек компании. По оценкам американских фирм-производителей автомобилей, внедрение ЭДО позволяет сэкономить около 200 долл. на каждом автомобиле. Суммарное годовое сокращение издержек

благодаря использованию систем ЭДО в Норвегии оценивалось в 1989 г. в 5 млрд норв. кр.4 (около 1 млрд долл.).

Очевидно, что системы электронного документооборота, пришедшие на смену традиционным, серьезно облегчили данный процесс и позволил крупным компаниям уменьшить затраты. Что же представляет собой эти новые системы? По мнению Пахчаняна А., в общем виде структура такой системы состоит из трех основных блоков: хранилище атрибутов документов, хранилище документов и компонентов, осуществляющих бизнес-логику системы[1].

Хранилище атрибутов документов предназначено для хранения набора полей (карточек), характеризующих документ. Обычно в системах электронного документооборота имеется понятие типа документов и для каждого типа имеется своя собственная карточка. Карточки разных типов имеют обязательные поля, общие для всех документов, и специальные поля, относящиеся к документам данного типа. Типы документов, в свою очередь, могут иметь подтипы, имеющие общий набор полей, который они наследуют от основного типа, и при этом дополнительные поля, уникальные для подтипа. Кроме понятия типа документов, возможно присваивание документам категорий, причем один документ может принадлежать одновременно к нескольким категориям. Для организации хранилища карточек возможны три варианта решения: использование собственного хранилища, стандартной базы данных или средств той среды, на основе которой построена база данных[2].

Хранилище атрибутов документов используется для хранения набора полей, с помощью которых описывается документ. В большинстве систем электронного документооборота вводится понятия «тип документа». Каждый тип документов имеет так называемую карточку, в которой перечислены все обязательные поля, которые присутствуют в документе каждого типа и специальные поля – у каждого типа свой набор таких полей. Типы

документов также могут делиться на подтипы. Подтипы могут иметь помимо набора полей основного типа свои собственные (примерно как принцип наследования в объектно-ориентированном программировании). Кроме типа, у документ также есть понятие «категории». Для организации хранилища карточек возможны три варианта решения: использование собственного хранилища, стандартной базы данных или средств той среды, на основе которой построена база данных [1].

В современных системах электронного документооборота хранилища документов может быть реализовано двумя способами. Первый предполагает хранение документов в файловой системе, второй же использует специальное хранилище. С точки зрения прагматичного пользователя между этими подходами, нет большой разницы. [1]

Если первые два элемента системы отвечают за хранение документов и их атрибутов, то последний компонент отвечает за все процессы, выполняемые в системе. Основные функции данного блока, представлены на схеме ниже (рис. 1).



Рисунок 1 Функции компонентов бизнес-логики

Управление документами в хранилище. Данная функция отвечает за жизненный цикл документа.

Поиск документов. Поиск может производиться по различным фильтрам – здесь все зависит от реализации конкретной системы.

Маршрутизация и контроль исполнения. Данная функция обеспечивает транспортировку внутри системы от одного сотрудника к другому. Маршруты документов можно разделить на 2 типа:

- Гибкий маршрут. Основной особенностью является то, что текущий пользователь, работающий с документом, сам определяет какому сотруднику или группе сотрудников он пойдет в дальнейшем.

- Жесткий маршрут. Этот вид подразумевает, что путь документ заранее определен в данной системе (например счет-фактура от бухгалтера – главному бухгалтеру и т.д.)

В большинстве современных систем используется совмещение данных двух типов.

Отчеты. Данная функция позволяет проследить кто? что? и когда? делал с документом. Параметрами, определяющими состояние документа, обычно служат:

- Время создания;
- От кого пришел;
- Версия;
- Время исполнения;
- И др.

Администрирование. Данная функция является регулирующей. Обычно в системах заводятся пользователи, являющиеся администраторами. В их функции входит определение ролей пользователей, регулирование

некорректных состояний документа в системе, поддержание работоспособности самой системы и т. д.

Главным назначением систем электронного документооборота является организация хранения и работы с электронными документами. В общем виде стадии жизненного цикла документа в такой системе можно представить в виде следующей схемы:



Рисунок 2 Жизненный цикл документа в СЭД

Рождение. При первом появлении документе в системе для него сразу же определяется тип и соответственно заполняется определенный для данной карточки набор полей. Разумеется, что все эти действия логируются, и при желании, сотрудник компании, с достаточным набором прав может отследить все действия, которые совершались над документом. Таким образом поддерживается версионность и в любой момент времени, можно посмотреть изменения документа, между двумя версиями.

Становление. На данной стадии, документ редактируется пользователями. Примечательно, что если один сотрудник «забрал документ на редактирование», то для всех остальных он будет заблокирован для редактирования, т.е. доступен только в режиме чтения. При этом для документа устанавливается новая версия, а всем остальным сотрудникам (с необходимым набором прав) доступна предыдущая версия документа. После того, как пользователь завершил редактирование, актуальная версия документа меняется, а изменения по сравнению с предыдущей версией, заносятся в журнал логирования.

Публикация. По сути это момент, когда редактирование документа завершается и он, становится неизменяемым. Данная функция всегда позволяет увидеть финальную версию документа. В случае если такой документ все же необходимо изменить, создается новый документ взамен текущего и опять проходит весь жизненный цикл сначала.

Архивирование. На данном этапе, документы отправляются в некое хранилище, в котором содержатся какое-то количество времени. Время нахождения документа в архиве обычно определяется его типом и настройками системы.

Теперь, когда мы уже познакомились с особенностями СЭД, их структурой и функциями пришло время разобраться с тем, что представляет собой такое понятие, как «электронный документ».

1.2 Электронный документ

Основными функции такого понятия как документооборот принято считать информационную – хранение и передачи какой-либо информации и доказательственную – возможность со стопроцентной вероятностью назвать автора документа. Основным аргумент в пользу использования бумажных

документов было то, что их достаточно сложно подделать и изменить. Таким образом сохранялась целостность документа и информация, которую он содержит, не изменялась. Как уже было сказано, сегодня системы электронного документооборота получают все большее распространение. Исходя из этого появляется необходимо закрепить правовой статус электронного документа[4].

Однако, несмотря на все достоинства электронного документа, у него имеется очень серьезный недостаток. При внесении в него правок, доказать такой факт становится очень сложной задачей. Конечно же, и бумажные документы нередко подделывают (да так, что потом даже автор не может понять где его документ, а где подделка), но тем не менее, а случае с электронными документами этот процесс намного проще.

Решить приемлемым образом данную задачу удалось только во второй половине 1970-х годов. Американские математики У. Диффи и М. Хеллман предложили использовать цифровую подпись для подтверждения подлинности электронных документов. Предложенная идея решила проблему идентификации электронных документов и легла в основу понятия электронной цифровой подписи, которая в наши дни получила достаточно широкое применение, в том числе в системах электронного документооборота.

1.3 Правовое регулирование электронного документооборота

Закон Республики Узбекистан 29.04.2004 г. №-611 об электронном документообороте

Целью настоящего Закона является регулирование отношений в области электронного документооборота. Электронный документооборот представляет собой совокупность процессов отправки и получения электронных документов через информационную систему. Электронный документооборот может использоваться для совершения сделок (в том числе

заключения договоров), осуществления расчетов, официальной и неофициальной переписки и передачи иной информации.

1.4 Электронная цифровая подпись

Сертификат ключа проверки электронной подписи – электронный документ или документ на бумажном носителе, выданные удостоверяющим центром либо доверенным лицом удостоверяющего центра и подтверждающие принадлежность ключа проверки электронной подписи владельцу сертификата ключа проверки электронной подписи. Владелец сертификата ключа проверки электронной подписи – лицо, которому в установленном настоящим Федеральным законом порядке выдан сертификат ключа проверки электронной подписи. Ключ электронной подписи – уникальная последовательность символов, предназначенная для создания электронной подписи. Ключ проверки электронной подписи – уникальная последовательность символов, однозначно связанная с ключом электронной подписи и предназначенная для проверки подлинности электронной подписи (далее - проверка электронной подписи).

В общем виде алгоритм подписания и проверки электронного документа можно представить в виде следующих шагов:

1. Генерация ключевой пары. При помощи алгоритма генерации ключа равновероятным образом из набора возможных закрытых ключей выбирается закрытый ключ, вычисляется соответствующий ему открытый ключ.

2. Формирование подписи. Для заданного электронного документа с помощью закрытого ключа вычисляется подпись.

3. Проверка (верификация) подписи. Для данных документа и подписи с помощью открытого ключа определяется действительность подписи.

Ниже представлена схема подписания и проверки электронного документа.

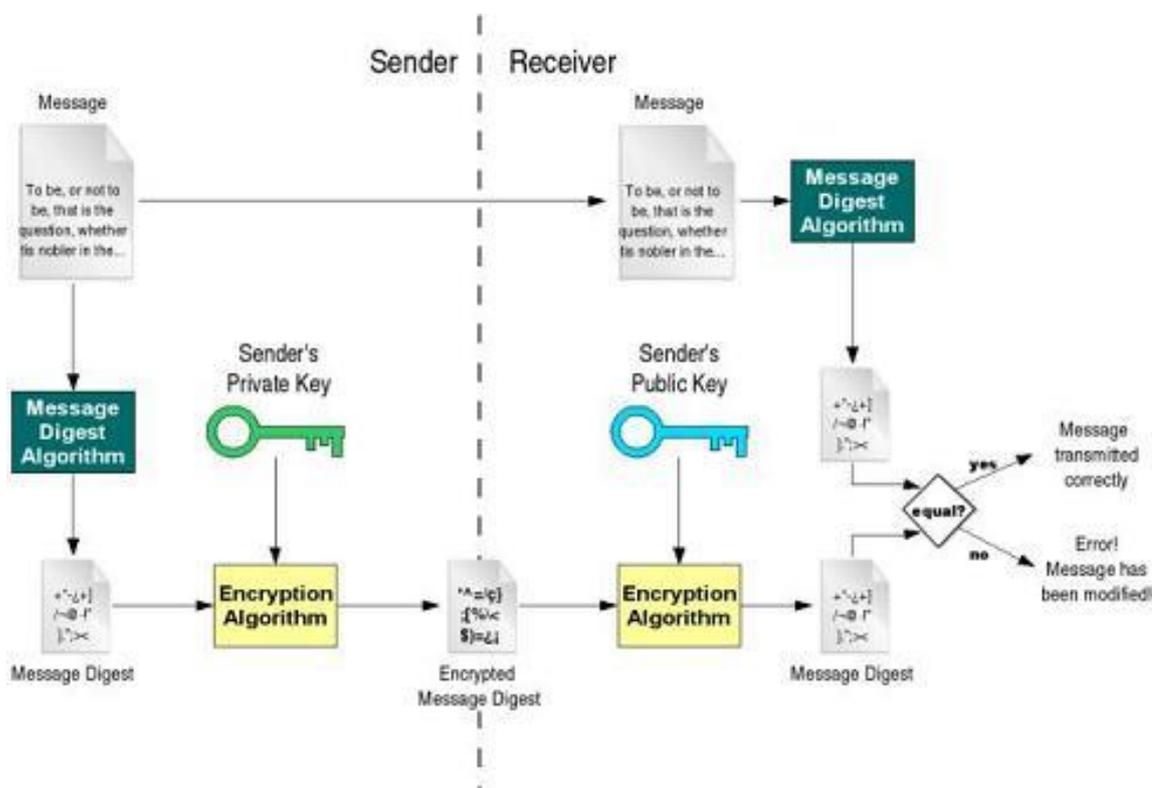


Рисунок 3 Схема подписание и проверки электронного документа

1.5 Назначение и сферы применения

Основной функцией электронной цифровой подписи является идентификации лица, подписавшего электронный документ. Кроме этого, использование электронной подписи позволяет решить следующие задачи:

- Обеспечение целостности документа. Данная функция является одной из ключевых для систем электронного документооборота. При каком-либо изменении документа стадия проверки подписи выдаст отрицательный результат.
- Обеспечение неизменяемости передаваемого документа. Опять же в случае подделки документа, стадия проверки выявит такой факт (если он был).
- Невозможность отказа и доказательное подтверждения авторства. Действительно с одной стороны, при наличии подписи под документом, автор не может от нее отказаться – так как создать такую подпись можно лишь используя закрытый ключ, владельцем которого он является. С другой

стороны, так как только этот человек является владельцем закрытого ключа, это полностью подтверждает его авторство документа. В данном пункте можно проследить прямую аналогию с собственноручной подписью.

1.6 Пример применения ЭЦП в Узбекистане

Для примера можно превести Единый портал интерактивных государственных услуг (ЕПИГУ), благодаря которому можно дистанционно (при помощи Интернета) обратиться в определенный государственный орган с жалобой или за разъяснением, а также для получения интерактивных услуг, как-то - онлайн-подача заявок в органы загса, госархив, райздравы для выдачи справок и иных документов.

Электронное заявление - почти такое же, как бумажное: вы указываете свои данные, описываете суть вопроса, ставите число... Но как быть с подписью? На мониторе компьютера ведь не распишешься, а подтвердить свое обращение нужно именно так. На этот случай предусмотрено - электронная подпись (ЭЦП).

Для получения интерактивных услуг через Единый портал или веб-сайты госорганов, а также для подачи электронных обращений в министерства и ведомства - во всех случаях, когда заявителю нужно подписать свое обращение электронной подписью.

Также ЭЦП необходима для подтверждения личности заявителя и придания онлайн-обращению юридической силы. Согласно Закону Республики Узбекистан «Об обращениях физических и юридических лиц» обращения, не подтвержденные электронной цифровой подписью, считаются анонимными и не рассматриваются.

Электронные цифровые подписи выдаются центрами регистрации ключей ЭЦП. В настоящее время в Узбекистане действуют 6 подобных учреждений, которыми выдано порядка 1,5 млн ключей ЭЦП.

Согласно постановлению Кабинета Министров Республики Узбекистан от 15.07.2015 г. за № 190 выдача электронных цифровых подписей, обеспечивающих доступ ко всем видам услуг органов государственного и хозяйственного управления, органов государственной власти на местах и коммерческими банками, осуществляется Научно-информационным центром новых технологий при Государственном налоговом комитете Республики Узбекистан. Такую подпись можно получить в ближайшей районной Государственной налоговой инспекции (ГНИ) на платной основе.

II Глава. Qr-code в системе документооборота

2.1 Описания Qr-кода

QR-код (англ. *Quick Response Code* — код быстрого реагирования; сокр. *QR code*) — товарный знак для типа матричных штрихкодов (или двумерных штрихкодов), изначально разработанных для автомобильной промышленности Японии. Штрихкод — считываемая машиной оптическая метка, содержащая информацию об объекте, к которому она привязана. QR-код использует четыре стандартизированных режима кодирования (числовой, буквенно-цифровой, двоичный и кандзи) для эффективного хранения данных; могут также использоваться расширения¹

QR-код разработан и представлен японской компанией Denso-Wave^[3] в 1994 году. Огромная популярность штрихкодов в Японии привела к тому, что объём информации, зашифрованной в них, вскоре перестал устраивать промышленность. Японцы начали экспериментировать с новыми современными способами кодирования небольших объёмов информации в графической картинке. В отличие от старого штрихкода, который сканируют тонким лучом, QR-код определяется датчиком или камерой как двумерное изображение. Три квадрата в углах изображения и меньшие синхронизирующие квадратики по всему коду позволяют нормализовать размер изображения и его ориентацию, а также угол, под которым датчик расположен к поверхности изображения. Точки переводятся в двоичные числа с проверкой по контрольной сумме.

Основное достоинство QR-кода — это лёгкое распознавание сканирующим оборудованием, что даёт возможность использования в торговле, производстве, логистике.

Максимальное количество символов, которые помещаются в один QR-код:

- цифры десятичной системы счисления — 7089;

- цифры десятичной системы счисления и буквы (латиница) — 4296;
- байты — 2953 (следовательно, около 2953 букв кириллицы в кодировке windows-1251 или около 1450 букв кириллицы в utf-8);
- иероглифы — 1817.

2.2 Кодирование данных

Закодировать информацию в QR-код можно несколькими способами, а выбор конкретного способа зависит от того, какие символы используются. Если используются только цифры от 0 до 9, то можно применить цифровое кодирование, если кроме цифр необходимо зашифровать буквы латинского алфавита, пробел и символы $\pm*/\$%*.:,$, используется алфавитно-цифровое кодирование. Ещё существует кодирование кандзи, которое применяется для шифрования китайских и японских иероглифов, и побайтовое кодирование. Перед каждым способом кодирования создаётся пустая последовательность бит, которая затем заполняется. [3]

2.2.1 Цифровое кодирование

Этот тип кодирования требует 10 бит на 3 символа. Вся последовательность символов разбивается на группы по 3 цифры, и каждая группа (трёхзначное число) переводится в 10-битное двоичное число и добавляется к последовательности бит. Если общее количество символов не кратно 3, то если в конце остаётся 2 символа, полученное двузначное число кодируется 7 битами, а если 1 символ, то 4 битами.

Например, есть строка «12345678», которую надо закодировать. Последовательность разбивается на числа: 123, 456 и 78, затем каждое число переводится в двоичный вид: 0001111011, 0111001000 и 1001110, и объединяется это в один битовый поток: 000111101101110010001001110. [3]

2.2.2 Буквенно-цифровое кодирование

В отличие от цифрового кодирования, для кодирования 2 символов требуется 11 бит информации. Последовательность символов разбивается на группы по 2, в группе каждый символ кодируется согласно таблице «Значения символов в буквенно-цифровом кодировании». Значение первого символа умножается на 45, затем к этому произведению прибавляется значение второго символа. Полученное число переводится в 11-битное двоичное число и добавляется к последовательности бит. Если в последней группе остаётся один символ, то его значение кодируется 6-битным числом. Рассмотрим на примере: «PROOF». Разбиваем последовательность символов на группы: PR, OO, F. Находим соответствующие значения символам к каждой группе (смотрим в таблицу): PR-(25,27), OO-(24,24), F-(15). Находим значения для каждой группы: $25*45+27=1152$, $24*45+24=1104$, $15=15$. Переводим каждое значение в двоичный вид: $1152=10010000000$, $1104=10001010000$, $15=001111$. Объединяем в одну последовательность: $1001000000010001010000001111$. [3]

2.2.3 Байтовое кодирование

Таким способом кодирования можно закодировать любые символы. Входной поток символов кодируется в любой кодировке (рекомендовано в UTF-8), затем переводится в двоичный вид, после чего объединяется в один битовый поток.

Например, слово «Мир» кодируем в Unicode (HEX) в UTF-8: М — D09C; и — D0B8; р — D180. Переводим каждое значение в двоичную систему счисления: D0=11010000, 9C =10011100, D0=11010000, B8=10111000, D1=11010001 и 80=10000000; объединяем в один поток бит: $11010000\ 10011100\ 11010000\ 10111000\ 11010001\ 10000000$ [3].

2.2.4 Кандзи

В основе кодирования иероглифов (как и прочих символов) лежит визуально воспринимаемая таблица или список изображений иероглифов с их кодами. Такая таблица называется «character set». Для японского языка основное значение имеют две таблицы символов: JIS 0208:1997 и JIS 0212:1990. Вторая из них является дополнением к первой. JIS 0208:1997 разбита на 94 страницы по 94 символа. К примеру, страница 4 — хирагана, 5 — катакана, 7 — кириллица, 16-43 — кандзи уровня 1, 48-83 — кандзи уровня 2. Кандзи уровня 1 («JIS дайити суйдзюн кандзи») упорядочены по онам. Кандзи уровня 2 (JIS дайни суйдзюн кандзи) упорядочены по ключам, и внутри них — по количеству черт. [3]

2.2.5 Добавление служебной информации

После определения версии кода и кодировки необходимо определиться с уровнем коррекции ошибок. В таблице представлены максимальные значения уровней коррекции для различных версий QR-кода. Для исправления ошибок применяется код Рида-Соломона с 8-битным кодовым словом.

Таблица. **Максимальное** **количество** **информации.**
Строка — уровень коррекции, столбец — номер версии.

	1	2	3	4	5	6	7	8	9	1	1	1	1	1	1	1	1	1	1	2
										0	1	2	3	4	5	6	7	8	9	0
L	1	2	4	6	8	1	1	1	1	2	2	2	3	3	4	4	5	5	6	6
	5	7	4	4	6	0	2	5	8	1	5	9	4	6	1	7	1	7	3	8
	2	2	0	0	4	8	4	5	5	9	9	6	2	8	8	1	7	6	6	8
					8	8	2	6	2	2	0	4	8	4	2	6	8	0	8	
M	1	2	3	5	6	8	9	1	1	1	2	2	2	2	3	3	4	4	5	5
	2	2	5	1	8	6	9	2	4	7	0	3	6	9	3	6	0	5	0	3

	8	4	2	2	8	4	2	3	5	2	3	2	7	2	2	2	5	0	1	5
								2	6	8	2	0	2	0	0	4	6	4	6	2
Q	1	1	2	3	4	6	7	8	1	1	1	1	1	2	2	2	2	3	3	3
	0	7	7	8	9	0	0	8	0	2	4	6	9	0	3	6	9	1	5	8
	4	6	2	4	6	8	4	0	5	3	4	4	5	8	6	0	3	7	6	8
									6	2	0	8	2	8	0	0	6	6	0	0
H	7	1	2	2	3	4	5	6	8	9	1	1	1	1	1	2	2	2	2	3
	2	2	0	8	6	8	2	8	0	7	1	2	4	5	7	0	2	5	7	0
		8	8	8	8	0	8	8	0	6	2	6	4	7	8	2	6	0	2	8
											0	4	0	6	4	4	4	4	8	0

После определения уровня коррекции ошибок необходимо добавить

21 22 23 24 25 26 27 28 29 30 31 32

L	745	804	875	939	1020	1096	1174	1224	1304	1388	1474	1564
	6	8	2	2	8	0	4	8	8	0	4	0
	571	625	688	731	8000	8496	9024	9544	1013	1098	1164	1232
	2	6	0	2					6	4	0	8
Q	409	454	491	531	5744	6032	6464	6968	7288	7880	8264	8920
	6	4	2	2								
H	324	353	371	411	4304	4768	5024	5288	5608	5960	6344	6760
	8	6	2	2								

служебные поля, они записываются перед последовательностью бит, полученной после этапа кодирования. В них указывается способ кодирования и количество данных. Значение поля способа кодирования состоит из 4 бит, оно не изменяется, а служит знаком, который показывает, какой способ кодирования используется. Оно имеет следующие значения:

- 0001 для цифрового кодирования,
- 0010 для буквенно-цифрового и
- 0100 для побайтового

Пример:

Ранее в примере байтового кодирования кодировалось слово «Мир», при этом получилась следующая последовательность двоичного кода:

11010000 10011100 11010000 10111000 11010001 10000000, содержащая 48 бит информации.

Пусть необходим уровень коррекции ошибок H , позволяющий восстанавливать 30 % утраченной информации. По таблице максимальное количество информации выбирается наиболее оптимальная версия QR-кода (в данном случае 1 версия, которая позволяет закодировать 72 символа полезной информации при уровне коррекции ошибок H).

Информация о способе кодирования: побайтовому кодированию соответствует поле 0100.

Указание количества данных (для цифрового и буквенно-цифрового кодирования — количество символов, для побайтового — количество байт): данная последовательность содержит 6 байт данных (в двоичной системе счисления: 110).

По таблице определяется необходимая длина двоичного числа — 8 бит. Дописываются недостающие нули: 00000110.

Версия 1-9 Версия 10-26 Версия 27-40

Цифровое	10 бит	12 бит	14 бит
Буквенно-цифровое	9 бит	11 бит	13 бит
Побайтовое	8 бит	16 бит	16 бит

Вся информация записывается в порядке <способ кодирования> <количество данных> <данные>, получается последовательность бит:

0100 00000110 11010000 10011100 11010000 10111000 11010001 10000000.

2.2.6 Разбиение на блоки

Последовательность байт разделяется на определённое для версии и уровня коррекции количество блоков, которое приведено в таблице «Количество блоков». Если количество блоков равно одному, то этот этап можно пропустить. А при повышении версии — добавляются специальные блоки.

Сначала определяется количество байт (данных) в каждом из блоков. Для этого надо разделить всё количество байт на количество блоков данных. Если это число не целое, то надо определить остаток от деления. Этот остаток определяет, сколько блоков из всех дополнены (такие блоки, количество байт в которых больше на один, чем в остальных). Вопреки ожиданию, дополненными блоками должны быть не первые блоки, а последние. Затем идёт последовательное заполнение блоков. [3]

Пример: для версии 9 и уровня коррекции M количество данных — 182 байта, количество блоков — 5. Поделив количество байт данных на количество блоков, получаем 36 байт и 2 байта в остатке. Это значит, что

блоки данных будут иметь следующие размеры: 36, 36, 36, 37, 37 (байт). Если бы остатка не было, то все 5 блоков имели бы размер по 36 байт.

Блок заполняется байтами из данных полностью. Когда текущий блок полностью заполняется, очередь переходит к следующему. Байтов данных должно хватить ровно на все блоки, не больше и не меньше.

2.2.7 Создание байтов коррекции

Процесс основан на алгоритме Рида-Соломона. Он должен быть применён к каждому блоку информации QR-кода. Сначала определяется количество байт коррекции, которые необходимо создать, а затем, с ориентиром на эти данные, создаётся многочлен генерации. Количество байтов коррекции на один блок определяются по выбранной версии кода и уровню коррекции ошибок (приведено в таблице).

Таблица. Количество байтов коррекции на один блок
Строка — уровень коррекции, столбец — номер версии. По количеству байтов коррекции определяется генерирующий многочлен.

Таблица. Генерирующие многочлены.

Количество байт коррекции	Генерирующий многочлен
7	87, 229, 146, 149, 238, 102, 21
10	251, 67, 46, 61, 118, 70, 64, 94, 32, 45
13	74, 152, 176, 100, 86, 100, 106, 104, 130, 218, 206, 140, 78
15	8, 183, 61, 91, 202, 37, 51, 58, 58, 237, 140, 124, 5, 99, 105
16	120, 104, 107, 109, 102, 161, 76, 3, 91, 191, 147, 169, 182,

	194, 225, 120
17	43, 139, 206, 78, 43, 239, 123, 206, 214, 147, 24, 99, 150, 39, 243, 163, 136
18	215, 234, 158, 94, 184, 97, 118, 170, 79, 187, 152, 148, 252, 179, 5, 98, 96, 153
20	17, 60, 79, 50, 61, 163, 26, 187, 202, 180, 221, 225, 83, 239, 156, 164, 212, 212, 188, 190
22	210, 171, 247, 242, 93, 230, 14, 109, 221, 53, 200, 74, 8, 172, 98, 80, 219, 134, 160, 105, 165, 231
24	173, 125, 158, 2, 103, 182, 118, 17, 145, 201, 111, 28, 165, 53, 161, 21, 245, 142, 13, 102, 48, 227, 153, 145, 218, 70
26	
28	168, 223, 200, 104, 224, 234, 108, 180, 110, 190, 195, 147, 205, 27, 232, 201, 21, 43, 245, 87, 42, 195, 212, 119, 242, 37, 9, 123
30	41, 173, 145, 152, 216, 31, 179, 182, 50, 48, 110, 86, 239, 96, 222, 125, 42, 173, 226, 193, 224, 130, 156, 37, 251, 216, 238, 40, 192, 180

Расчёт производится исходя из значений исходного массива данных и значений генерирующего многочлена, причём для каждого шага цикла отдельно.

2.2.8 Объединение информационных блоков

На данном этапе имеется два готовых блока: исходных данных и блоков коррекции (из прошлого шага), их необходимо объединить в один поток байт. По очереди необходимо брать один байт информации из каждого блока данных, начиная от первого и заканчивая последним. Когда же очередь доходит до последнего блока, из него берётся байт и очередь переходит к первому блоку. Так продолжается до тех пор, пока в каждом блоке не

закончатся байты. Есть исключения, когда текущий блок пропускается, если в нём нет байт (ситуация, когда обычные блоки уже пусты, а в дополненных ещё есть по одному байту). Так же поступается и с блоками байтов коррекции. Они берутся в том же порядке, что и соответствующие блоки данных[3].

В итоге получается следующая последовательность данных: <1-й байт 1-го блока данных><1-й байт 2-го блока данных>...<1-й байт n-го блока данных><2-й байт 1-го блока данных>...<(m — 1)-й байт 1-го блока данных>...<(m — 1)-й байт n-го блока данных><m-й байт k-го блока данных>...<m-й байт n-го блока данных><1-й байт 1-го блока байтов коррекции><1-й байт 2-го блока байтов коррекции>...<1-й байт n-го блока байтов коррекции><2-й байт 1-го блока байтов коррекции>...<1-й байт 1-го блока байтов коррекции>...<1-й байт n-го блока байтов коррекции>.

Здесь n — количество блоков данных, m — количество байтов на блок данных у обычных блоков, l — количество байтов коррекции, k — количество блоков данных минус количество дополненных блоков данных (тех, у которых на 1 байт больше). [3]

2.2.9 Этап размещения информации на поле кода

На QR-коде есть обязательные поля, они не несут закодированной информации, а содержат информацию для декодирования. Это:

- Поисковые узоры
- Выравнивающие узоры
- Полосы синхронизации
- Код маски и уровня коррекции
- Код версии (с 7-й версии)

	3	4	5	6		9	0	1	2	3	4	5	6	7	8	9	0		
6,	6,	6,	6,	6,	6,	6,	6,	6,	6,	6,	6,	6,	6,	6,	6,	6,	6,		
28	26	3	2	3	3	34	26	3	2	3	3	3	3	2	2	3	2	3	
,	,	0,	8,	2,	0,	,	,	0,	6,	0,	4,	0,	4,	0,	4,	8,	2,	6,	0,
50	50	5	5	5	5	62	50	5	5	5	6	5	6	5	5	5	5	5	5
72	,7	4,	4,	8,	8,	,	,	4,	2,	6,	0,	8,	2,	4,	0,	4,	8,	4,	8,
94	4	7	8	8	8	90	74	7	7	8	8	8	9	7	7	8	8	8	8
	98	8,	0,	4,	6,	,	,	8,	8,	2,	6,	6,	0,	8,	6,	0,	4,	2,	6,
	1	1	1	1	11	98	1	1	1	1	1	1	1	1	1	1	1	1	1
	0	0	1	1	8	,	0	0	0	1	1	1	0	0	0	1	1	1	1
	2	6	0	4		12	2,	4,	8,	2,	4,	8,	2,	2,	6,	0,	0,	4,	
						2	1	1	1	1	1	1	1	1	1	1	1	1	1
							2	3	3	3	4	4	2	2	3	3	3	4	
							6	0	4	8	2	6	6,	8,	2,	6,	8,	2,	
														1	1	1	1	1	1
														5	5	5	6	6	7
														0	4	8	2	6	0

Полосы синхронизации — используются для определения размера модулей. Располагаются они уголком, начинается одна от левого нижнего поискового узора (от края чёрной рамки, но переступив через белую), идёт до левого верхнего, а оттуда начинается вторая, по тому же правилу, заканчивается она у правого верхнего. При наслоении на выравнивающий модуль он должен остаться без изменений. Выглядят полосы синхронизации как линии чередующихся между собой чёрных и белых модулей. [4]

Код маски и уровня коррекции — расположен рядом с поисковыми узорами: под правым верхним (8 модулей) и справа от левого нижнего (7 модулей), и дублируются по бокам левого верхнего, с пробелом на 7 ячейке — там, где проходят полосы синхронизации, причём горизонтальный код в вертикальную часть, а вертикальный — в горизонтальную. [4]

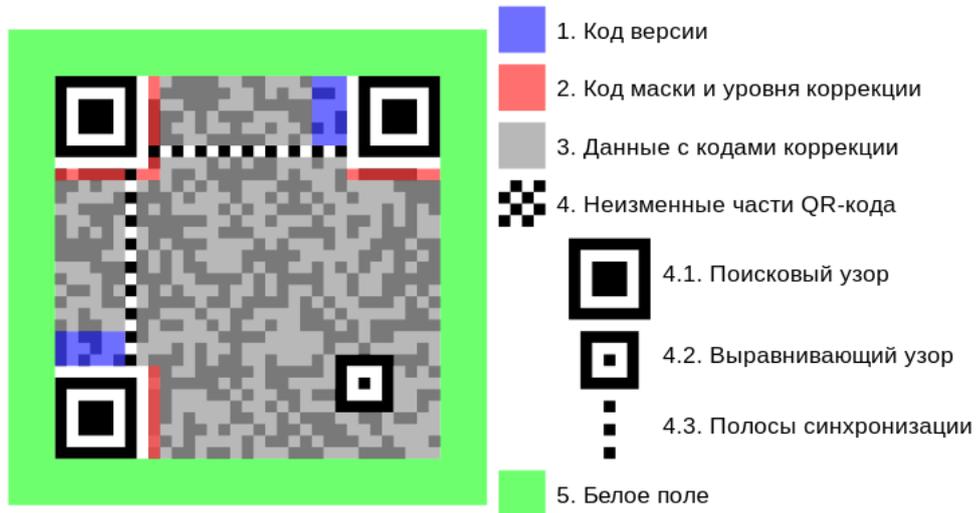
Код версии — нужен для определения версии кода. Находятся слева от верхнего правого и сверху от нижнего левого, причём дублируются. Дублируются они так — зеркальную копию верхнего кода поворачивают против часовой стрелки на 90 градусов. Ниже представлена таблица кодов, 1 — чёрный модуль, 0 — белый. [4]

Вер	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
сия																	
	000	010	110	101	001	001	101	110	010	011	111	100	000	000	100	111	011
Ко	010	001	111	001	111	101	011	101	011	100	010	100	010	000	110	000	110
д	011	011	011	111	111	100	100	000	000	010	010	110	110	101	101	001	001
вер	110	100	000	110	010	100	000	110	010	001	101	011	111	001	101	011	111
сии	100	111	000	000	111	011	100	100	011	011	100	100	011	111	000	000	111
	110	000	100	000	100	010	110	010	110	100	000	100	000	110	010	110	010
24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	
	001	101	110	010	010	110	101	001	101	001	010	110	110	010	001	101	111
	101	011	101	011	001	111	001	111	000	110	000	110	100	010	100	010	001
	001	001	101	101	110	110	010	010	011	011	111	111	100	100	000	000	000
	101	001	111	011	101	001	111	011	000	100	010	110	000	100	010	110	100
	100	011	011	100	000	111	111	000	101	010	010	101	001	110	110	001	010
	100	000	100	000	110	010	110	010	101	001	101	001	111	011	111	011	101

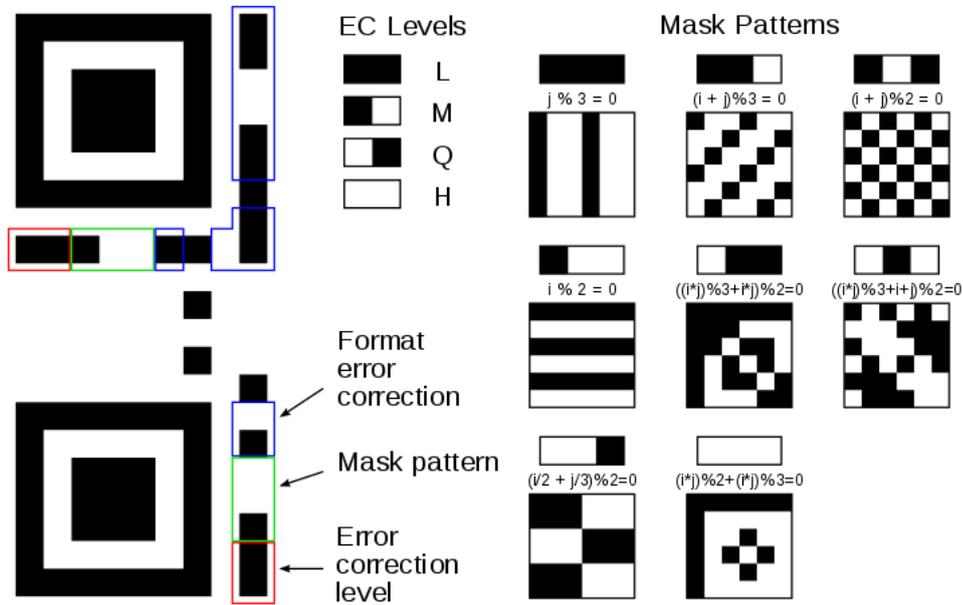
2.2.10 Занесение данных

Оставшееся свободным место делят на столбики шириной в 2 модуля и заносят туда информацию, причём делают это «змейкой». Сначала в правый нижний квадратик заносят первый бит информации, потом в его левого соседа, потом в тот, который был над первым и так далее. Заполнение столбцов ведётся снизу-вверх, а потом сверху вниз и т. д., причём по краям заполнение битов ведётся от крайнего бита одного столбца до крайнего бита

соседнего столбца, что задаёт «змейку» на столбцы с направлением вниз. Если информации окажется недостаточно, то поля просто оставляют пустыми (белые модули). При этом на каждый модуль накладывается маска. [3]



Описание полей QR-кода.



Код маски и уровня коррекции, возможные XOR-маски[3]

III ГЛАВА СИСТЕМЫ ДОКУМЕНТООБОРОТА

3.1 Система Документооборота: Ijro.gov.uz

Единая система контроля исполнительской дисциплины, Безопасность системы осуществлена на базе стандартов:

O`zDST 1105:2009

Алгоритм шифрования данных O`zDST 1092:2009

Процессы формирования и проверки ЭЦП Версия 3.0 © UNICON.UZ.

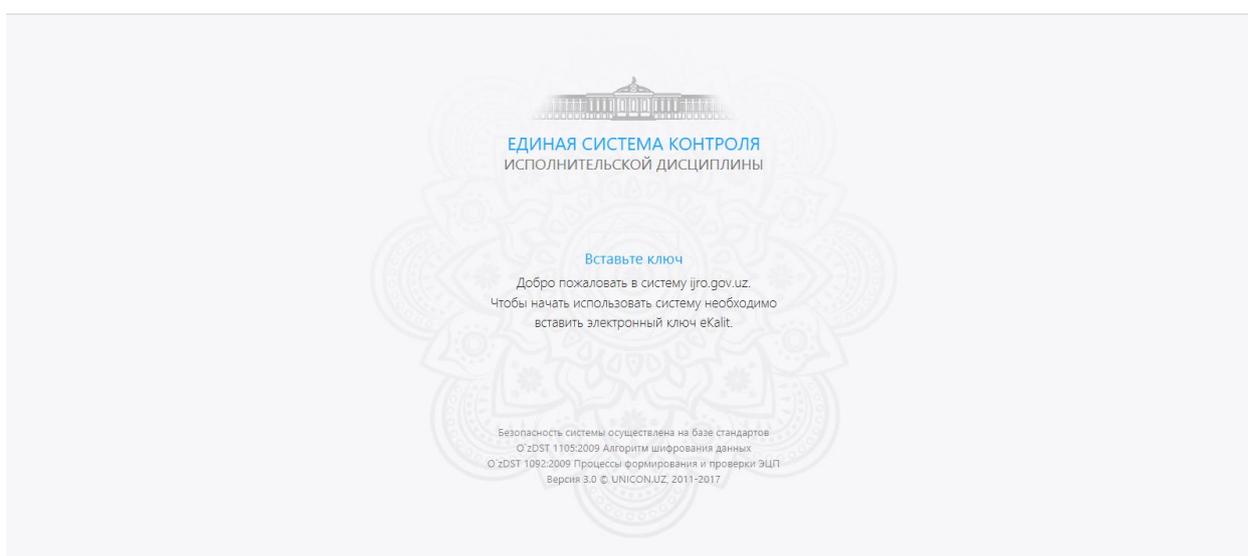


Рис 3.1 Ijro.gov.uz Вход в систему

Для входа в систему используется электронный ключ E-KALIT который содержит в себе пару ключей. E-KALIT специально разработано для использования в целях защищенного хранения секретных ключей шифрования и закрытых ключей электронной цифровой подписи.

E-KALIT выполнен в виде USB-носителя с небольшими габаритами и обеспечивает хранение личных криптографических данных пользователя в защищенной доверенной среде.

«E-KALIT» обеспечивает: - безопасное хранение личных криптографических данных пользователя в защищенной доверенной среде, включая закрытые

ключи ЭЦП, ключи хеширования, ключи шифрования и т.д.; - безопасную авторизацию пользователя для доступа к содержимому памяти устройства на основе идентификационных данных (логин и пароль);

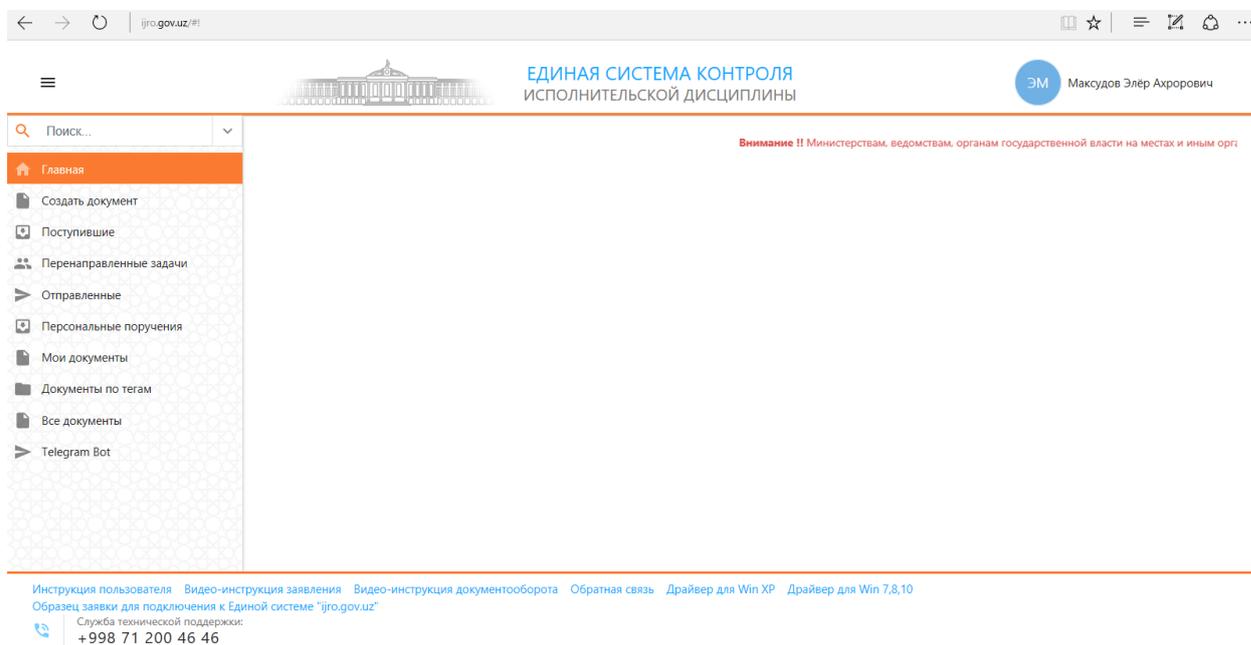


Рис 3.2 Главная страница Ijro.gov.uz

Система ijro предназначена для обеспечения межведомственного электронного взаимодействия посредством взаимной интеграции внутренних систем электронного документооборота государственных органов.

Система ijro позволяет:

- осуществлять оперативный мониторинг и контроль хода исполнения постановлений, приказов, распоряжений, обращений граждан и иных документов, сформированных на различных уровнях вертикали государственного управления (Кабинет Министров Республики Узбекистан, государственные органы, государственные организации);
- формировать оперативные отчеты о состоянии исполнения документа с разбивкой по исполнителям;

- сократить издержки (затраты времени и бумаги) информационного взаимодействия участников межведомственного электронного документооборота;
- повысить эффективность выполнения поручений и распоряжений правительства;

Как видно на рисунке 3.2 на главной странице ijro.gov.uz ведутся доработки системы в связи с чем в этой выпускной квалификационной работе было разработана приложение которая может использоваться в ijro.gov.uz.

3.2 Описание интерфейса системы

При первом запуске системы, открывается окно авторизации

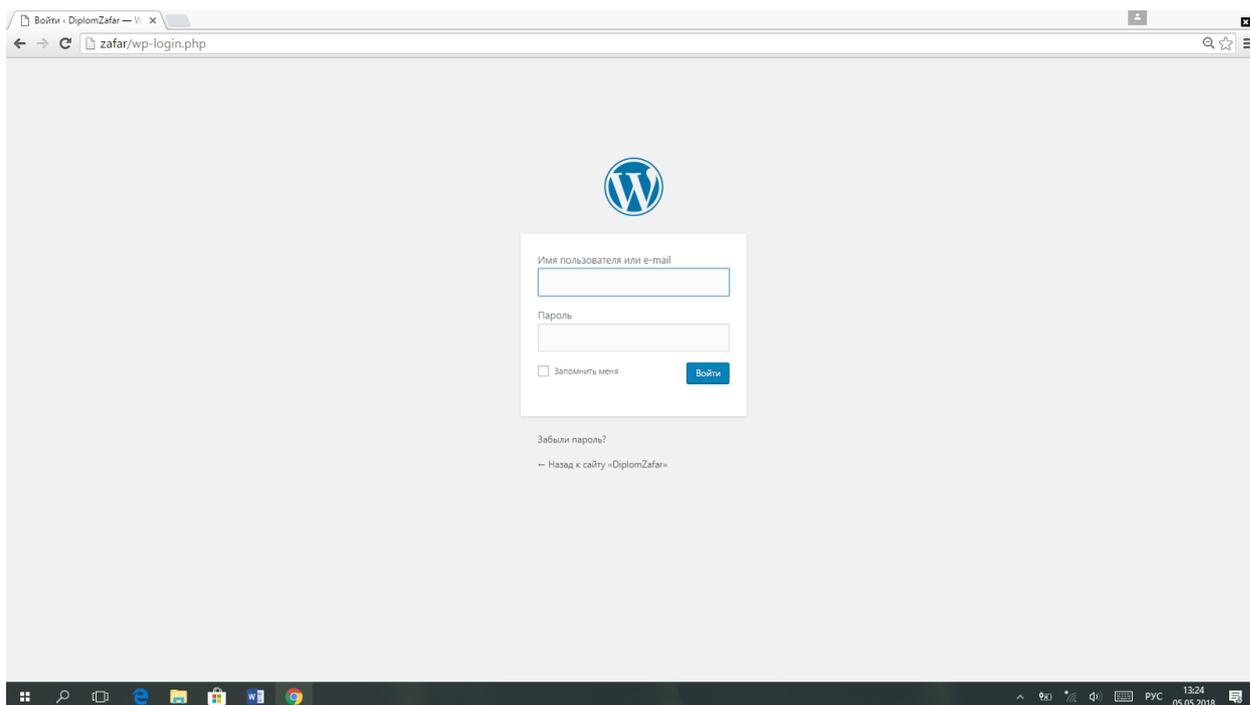


Рис 3.3 Авторизация

В созданную систему вход осуществляется с помощью логина и пароля. Логин и пароль сохраняется в базе данных при регистрации нового пользователя. После успешной авторизации пользователь будет иметь возможность создавать электронный документ также изменять и удалять.

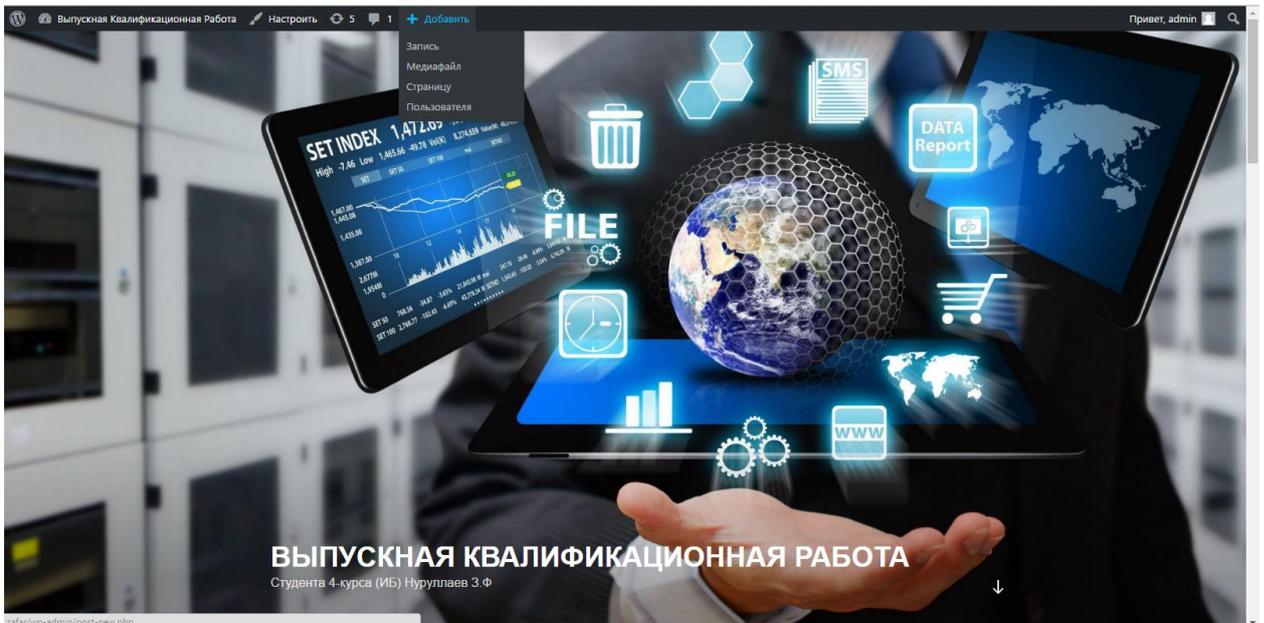


Рис 3.4 главная страница системы

Чтобы создать новый документ пользователь должен нажать на кнопку добавить, кнопка добавить находится на верхнем панели приложения после чего открывается окно:

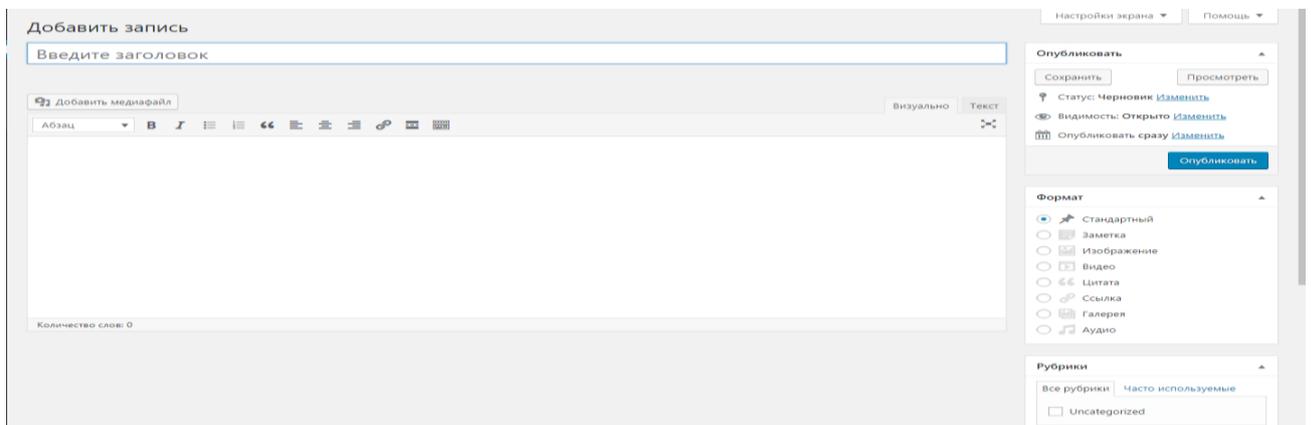


Рис 3.4 создания документа

Как видно на рис 3.4 где написано введите заголовок туда вводим название заголовки и далее если есть файл электронный то можем закрепить его вместе с создаваемым документом, на в тором панели вводится содержания документа, после заполнения документа на правом панели приложения (рис 3.4) видно кнопка опубликовать, нажав на эту кнопку

пользователь создает документ который будет находится на главной странице приложения (рис 3.5)

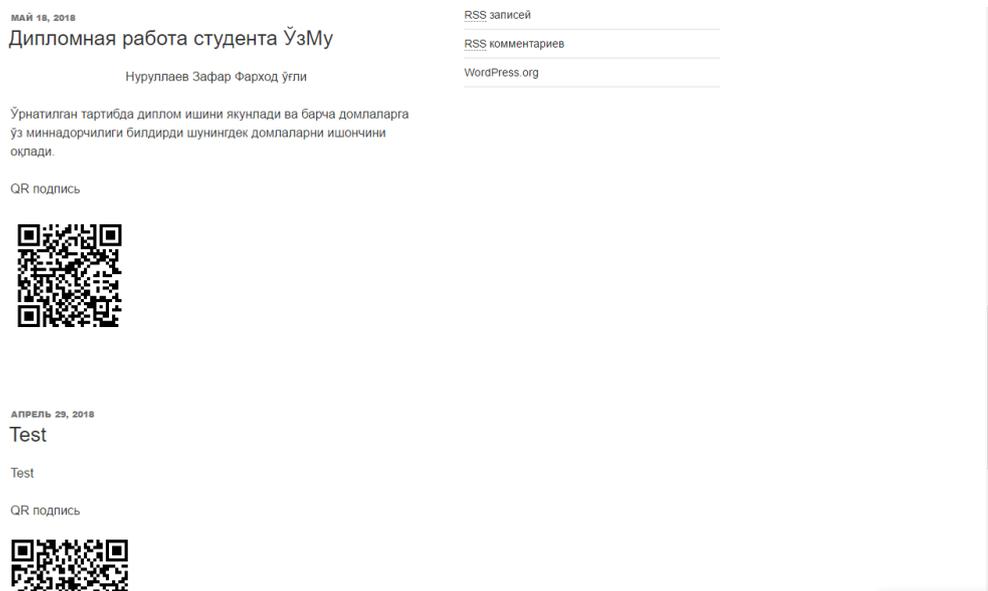


Рис 3.5 Главная страница где осуществляется подпись с помощью QR-кода.

QR-код – это двухмерный штрих-код, в котором может быть в закодированном виде представлена информация о пользователе или о владельце документа. QR (Quick Response) дословно означает «быстрый отклик» – информация с него действительно может быть считана практически мгновенно. Как и цифровой код, размещается непосредственно на самой бумаге вместо ручного подписи. При необходимости QR-код может быть дополнен электронной цифровой подписью (ЭЦП). ЭЦП является юридически значимым документом, подделка которого преследуется по закону.

Заключение

Использование методов безбумажной обработки и обмена документами позволяет значительно сократить время, затрачиваемое на оформление сделки и обмен документацией, усовершенствовать и удешевить процедуру подготовки, доставки, учета и хранения документов, построить

корпоративную систему обмена документами. Однако при переходе на электронный документооборот встает вопрос авторства документа, достоверности и защиты от искажений. В этой системе защита документов обеспечивается разграничением доступа также аутентификацией пользователя в системе документооборота.

С помощью Qr-code система документооборота будет более простим и понятным, электронная подпись в виде Qr-code подтверждает написанный текст и также дает информацию о владельце документа, чтобы утвердит юридическую силу документа можем применять печать организации ,это даст нам защиту от копии, но при этом визуально документ не будет меняться, если начнем применять QR-CODE как подпись, то получим электронный документ с электронным подписью это облегчит процесс документооборота и ускорит обмена, также с помощью Qr-code можно оставить примечания или комментарии для документа.

Литература

1. Бичевина Н. Применения ЭЦП (электронная цифровая подпись): возможности, проблемы и перспективы. [Электронный ресурс] <http://esm-journal.ru/docs/Primenenie-EhCP-ehlektronnaja-cifrovaja-podpis--vozmozhnosti-problemy-i-perspektivy.aspx>
2. Жданов А.Н., Чалкин Т.А. Применение эллиптических кривых в криптографии. <http://window.edu.ru>
3. Игоничкина Е.В. Анализ алгоритмов электронной цифровой подписи. [Электронный ресурс] <http://www.security.ase.md/publ/ru/pubru86/>
4. Ливак Е.Н. Электронная цифровая подпись. [Электронный ресурс] http://mf.grsu.by/Kafedry/kaf001/academic_process/umo/074/lec_07/
5. Пахчанян А. Технологии электронного документооборота. [Электронный ресурс] <http://citforum.ru/consulting/docflow/technologies/>
6. Рассолов М.М. Информационное право. – М.: Проспект, 2013. – 350 с.
7. Ротков Л.Ю., Зобнев А.В. Электронная цифровая подпись в электронном документообороте. – Нижний Новгород, 2006. – 42 с.
8. Янковая Ф.А. Электронный документ как объект документоведения. [Электронный ресурс] <http://cyberleninka.ru/article/n/elektronnyy-dokument-kak-obekt-dokumentovedeniya>
9. Википедия. [Электронный ресурс] <https://ru.wikipedia.org/wiki>
10. Справочник юриста. [Электронный ресурс] <http://www.sudru.ru/>
11. Электронный документооборот. [Электронный ресурс] <http://iesp.ru/ep/law-review/edo>

ПРИЛОЖЕНИЕ

Приложения написана на языке PHP с помощью WordPress.

WordPress — система управления содержимым сайта с открытым исходным кодом; написана на PHP; сервер базы данных — MySQL; выпущена под лицензией GNU GPL версии 2. Сфера применения — от блогов до достаточно сложных новостных ресурсов и интернет-магазинов. Встроенная система «тем» и «плагинов» вместе с удачной архитектурой позволяет конструировать проекты широкой функциональной сложности.

```
-- phpMyAdmin SQL Dump
-- version 4.6.5.2
-- https://www.phpmyadmin.net/
--
-- Хост: 127.0.0.1:3306
-- Время создания: Май 03 2018 г., 15:22
-- Версия сервера: 5.7.16
-- Версия PHP: 5.6.29

SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";
SET time_zone = "+00:00";

/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
/*!40101 SET NAMES utf8mb4 */;

--
-- База данных: `zafar`
--
-----

--
-- Структура таблицы `wp_commentmeta`
--

CREATE TABLE `wp_commentmeta` (
  `meta_id` bigint(20) UNSIGNED NOT NULL,
```

```

`comment_id` bigint(20) UNSIGNED NOT NULL DEFAULT '0',
`meta_key` varchar(255) COLLATE utf8mb4_unicode_520_ci DEFAULT NULL,
`meta_value` longtext COLLATE utf8mb4_unicode_520_ci
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_520_ci;

-- -----
--
-- Структура таблицы `wp_comments`
--

CREATE TABLE `wp_comments` (
  `comment_ID` bigint(20) UNSIGNED NOT NULL,
  `comment_post_ID` bigint(20) UNSIGNED NOT NULL DEFAULT '0',
  `comment_author` tinytext COLLATE utf8mb4_unicode_520_ci NOT NULL,
  `comment_author_email` varchar(100) COLLATE utf8mb4_unicode_520_ci NOT NULL
  DEFAULT '',
  `comment_author_url` varchar(200) COLLATE utf8mb4_unicode_520_ci NOT NULL
  DEFAULT '',
  `comment_author_IP` varchar(100) COLLATE utf8mb4_unicode_520_ci NOT NULL
  DEFAULT '',
  `comment_date` datetime NOT NULL DEFAULT '0000-00-00 00:00:00',
  `comment_date_gmt` datetime NOT NULL DEFAULT '0000-00-00 00:00:00',
  `comment_content` text COLLATE utf8mb4_unicode_520_ci NOT NULL,
  `comment_karma` int(11) NOT NULL DEFAULT '0',
  `comment_approved` varchar(20) COLLATE utf8mb4_unicode_520_ci NOT NULL
  DEFAULT '1',
  `comment_agent` varchar(255) COLLATE utf8mb4_unicode_520_ci NOT NULL
  DEFAULT '',
  `comment_type` varchar(20) COLLATE utf8mb4_unicode_520_ci NOT NULL DEFAULT
  '',
  `comment_parent` bigint(20) UNSIGNED NOT NULL DEFAULT '0',
  `user_id` bigint(20) UNSIGNED NOT NULL DEFAULT '0'
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_520_ci;

--
-- Дамп данных таблицы `wp_comments`
--

INSERT INTO `wp_comments` (`comment_ID`, `comment_post_ID`, `comment_author`,
`comment_author_email`, `comment_author_url`, `comment_author_IP`,
`comment_date`, `comment_date_gmt`, `comment_content`, `comment_karma`,

```

```

`comment_approved`, `comment_agent`, `comment_type`, `comment_parent`,
`user_id`) VALUES
(1, 1, 'A WordPress Commenter', 'wapuu@wordpress.example',
'https://wordpress.org/', '', '2018-04-20 04:32:19', '2018-04-20 04:32:19',
'Hi, this is a comment.\nTo get started with moderating, editing, and
deleting comments, please visit the Comments screen in the
dashboard.\nCommenter avatars come from <a
href=\"https://gravatar.com\">Gravatar</a>.', 0, '1', '', '', 0, 0);

```

```

-- -----

```

```

--

```

```

-- Структура таблицы `wp_links`

```

```

--

```

```

CREATE TABLE `wp_links` (
  `link_id` bigint(20) UNSIGNED NOT NULL,
  `link_url` varchar(255) COLLATE utf8mb4_unicode_520_ci NOT NULL DEFAULT '',
  `link_name` varchar(255) COLLATE utf8mb4_unicode_520_ci NOT NULL DEFAULT
'',
  `link_image` varchar(255) COLLATE utf8mb4_unicode_520_ci NOT NULL DEFAULT
'',
  `link_target` varchar(25) COLLATE utf8mb4_unicode_520_ci NOT NULL DEFAULT
'',
  `link_description` varchar(255) COLLATE utf8mb4_unicode_520_ci NOT NULL
DEFAULT '',
  `link_visible` varchar(20) COLLATE utf8mb4_unicode_520_ci NOT NULL DEFAULT
'Y',
  `link_owner` bigint(20) UNSIGNED NOT NULL DEFAULT '1',
  `link_rating` int(11) NOT NULL DEFAULT '0',
  `link_updated` datetime NOT NULL DEFAULT '0000-00-00 00:00:00',
  `link_rel` varchar(255) COLLATE utf8mb4_unicode_520_ci NOT NULL DEFAULT '',
  `link_notes` mediumtext COLLATE utf8mb4_unicode_520_ci NOT NULL,
  `link_rss` varchar(255) COLLATE utf8mb4_unicode_520_ci NOT NULL DEFAULT ''
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_520_ci;

```

```

-- -----

```

```

--

```

```

-- Структура таблицы `wp_options`

```

```

--

```

```

CREATE TABLE `wp_options` (
  `option_id` bigint(20) UNSIGNED NOT NULL,
  `option_name` varchar(191) COLLATE utf8mb4_unicode_520_ci NOT NULL DEFAULT
'',
  `option_value` longtext COLLATE utf8mb4_unicode_520_ci NOT NULL,
  `autoload` varchar(20) COLLATE utf8mb4_unicode_520_ci NOT NULL DEFAULT
'yes'
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_520_ci;

--
-- Дамп данных таблицы `wp_options`
--

INSERT INTO `wp_options` (`option_id`, `option_name`, `option_value`,
`autoload`) VALUES
(128x128.jpg?rev=1148084\");s:7:\"banners\";a:1:{s:2:\"1x\";s:75:\"https://ps.
w.org/wp-qr-code-generator/assets/banner-
772x250.jpg?rev=1145234\");s:11:\"banners_rtl\";a:0:{}}, 'no');

-----

--
-- Структура таблицы `wp_postmeta`
--

CREATE TABLE `wp_postmeta` (
  `meta_id` bigint(20) UNSIGNED NOT NULL,
  `post_id` bigint(20) UNSIGNED NOT NULL DEFAULT '0',
  `meta_key` varchar(255) COLLATE utf8mb4_unicode_520_ci DEFAULT NULL,
  `meta_value` longtext COLLATE utf8mb4_unicode_520_ci
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_520_ci;

--
-- Дамп данных таблицы `wp_postmeta`
--

INSERT INTO `wp_postmeta` (`meta_id`, `post_id`, `meta_key`, `meta_value`)
VALUES
(1, 2, '_wp_page_template', 'default'),
(2, 1, '_edit_lock', '1524992654:1'),
(3, 1, '_edit_last', '1'),
(4, 27, '_edit_last', '1'),

```

```
(5, 27, '_edit_lock', '1524993062:1'),
(8, 34, '_edit_lock', '1524993238:1'),
(9, 34, '_edit_last', '1'),
(12, 38, '_wp_trash_meta_status', 'publish'),
(13, 38, '_wp_trash_meta_time', '1525108976');
```

```
-----
```

```
--
```

```
-- Структура таблицы `wp_posts`
```

```
--
```

```
CREATE TABLE `wp_posts` (
  `ID` bigint(20) UNSIGNED NOT NULL,
  `post_author` bigint(20) UNSIGNED NOT NULL DEFAULT '0',
  `post_date` datetime NOT NULL DEFAULT '0000-00-00 00:00:00',
  `post_date_gmt` datetime NOT NULL DEFAULT '0000-00-00 00:00:00',
  `post_content` longtext COLLATE utf8mb4_unicode_520_ci NOT NULL,
  `post_title` text COLLATE utf8mb4_unicode_520_ci NOT NULL,
  `post_excerpt` text COLLATE utf8mb4_unicode_520_ci NOT NULL,
  `post_status` varchar(20) COLLATE utf8mb4_unicode_520_ci NOT NULL DEFAULT
'publish',
  `comment_status` varchar(20) COLLATE utf8mb4_unicode_520_ci NOT NULL
DEFAULT 'open',
  `ping_status` varchar(20) COLLATE utf8mb4_unicode_520_ci NOT NULL DEFAULT
'open',
  `post_password` varchar(255) COLLATE utf8mb4_unicode_520_ci NOT NULL
DEFAULT '',
  `post_name` varchar(200) COLLATE utf8mb4_unicode_520_ci NOT NULL DEFAULT
'',
  `to_ping` text COLLATE utf8mb4_unicode_520_ci NOT NULL,
  `pinged` text COLLATE utf8mb4_unicode_520_ci NOT NULL,
  `post_modified` datetime NOT NULL DEFAULT '0000-00-00 00:00:00',
  `post_modified_gmt` datetime NOT NULL DEFAULT '0000-00-00 00:00:00',
  `post_content_filtered` longtext COLLATE utf8mb4_unicode_520_ci NOT NULL,
  `post_parent` bigint(20) UNSIGNED NOT NULL DEFAULT '0',
  `guid` varchar(255) COLLATE utf8mb4_unicode_520_ci NOT NULL DEFAULT '',
  `menu_order` int(11) NOT NULL DEFAULT '0',
  `post_type` varchar(20) COLLATE utf8mb4_unicode_520_ci NOT NULL DEFAULT
'post',
  `post_mime_type` varchar(100) COLLATE utf8mb4_unicode_520_ci NOT NULL
DEFAULT '',
```

```
    `comment_count` bigint(20) NOT NULL DEFAULT '0'  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_520_ci;  
  
--  
-- Дамп данных таблицы `wp_posts`  
--  
/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;  
/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
```