МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение высшего образования
«Тольяттинский государственный университет»

Кафедра «Прикладная математика и информатика»
(наименование)

09.03.03 Прикладная информатика
(код и наименование направления подготовки / специальности)

Цифровая трансформация бизнеса
(направленность (профиль)/специализация)

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА (БАКАЛАВРСКАЯ РАБОТА)

на тему «Разработка и внедрение автоматизированного решения для генерации тестовых открытых сертификатов электронной подписи»

Обучающийся В.В. Малыгин
(Инициалы Фамилия) (личная подпись)

Руководитель М.Г. Лисовская
(ученая степень (при наличии), ученое звание (при наличии), Инициалы Фамилия)

Тольятти 2025

Аннотация

Настоящая работа посвящена разработке отечественного автоматизированного решения для генерации тестовых сертификатов в условиях цифровизации экономики и повышения требований к скорости и безопасности обработки электронных данных. Актуальность исследования обусловлена необходимостью устранения недостатков существующих зарубежных решений, среди которых отмечаются сложность использования, ограниченная функциональность, слабая интеграция с российскими системами и зависимость от иностранных технологий.

В процессе исследования проведён всесторонний анализ существующих инструментов, сформулированы требования к новому решению, разработана архитектура системы, включающая модули для автоматической проверки данных, генерации сертификатов и их последующего анализа. Разработанная система отличается удобным пользовательским интерфейсом, полной интеграцией с внутренними корпоративными системами, а также полным соответствием требованиям российского законодательства в области информационной безопасности.

Экономическая оценка проекта показала высокую эффективность решения: при затратах на разработку в 1 100 000 рублей годовая экономия составляет 549 700 рублей, что обеспечивает срок окупаемости менее двух лет. Перспективы развития системы включают расширение поддерживаемых форматов сертификатов, интеграцию с системами электронного документооборота и адаптацию под отечественные криптографические алгоритмы.

Реализация данного проекта позволяет организациям существенно повысить производительность работы с сертификатами, минимизировать операционные затраты и укрепить технологическую независимость России в области информационных технологий и цифровой безопасности.

Содержание

Введение

Современные процессы цифровизации экономики и государственного управления предъявляют новые требования к обращению с цифровыми сертификатами. Несмотря на это, во многих организациях по-прежнему применяются ручные методы генерации и управления, что приводит к значительным затратам времени, повышенной вероятности ошибок и затрудняет масштабирование. В условиях роста электронного документооборота и необходимости быстрой обработки запросов такие подходы становятся все менее жизнеспособными.

Особенно остро проблема стоит перед структурами с высокой интенсивностью операций с сертификатами. Имеющиеся зарубежные решения не обеспечивают должного уровня автоматизации, требуют участия квалифицированных специалистов даже для рутинных задач, а также слабо интегрируются с отечественными системами и не учитывают специфику российского законодательства.

В этой связи создание эффективного автоматизированного инструмента для управления сертификатами приобретает стратегическое значение для цифровой инфраструктуры страны.

Актуальность темы обусловлена необходимостью преодоления ограничений существующих практик в сфере генерации и администрирования цифровых сертификатов. В условиях цифровой трансформации ручные методы становятся узким местом, препятствующим повышению эффективности и масштабируемости процессов. Возрастает потребность в разработке отечественного решения, обеспечивающего высокий уровень автоматизации. Исследование ориентировано на устранение разрыва между реальными потребностями организаций в оперативной работе с сертификатами и возможностями доступных технологий. Это делает проект особенно актуальным в контексте импортозамещения и развития национальной информационной безопасности [7].

Объектом исследования выступает комплексный процесс выпуска тестовых сертификатов – от поступления запроса до окончания срока действия. Рассматриваются ключевые этапы: обработка входящих обращений, генерация и интеграция с внутренними системами организации. Предметом исследования является инновационное автоматизированное решение, позволяющее оптимизировать процесс генерации тестовых сертификатов с точки зрения архитектуры, функциональности, модели данных и экономической эффективности.

Цель работы заключается в разработке и обосновании эффективного автоматизированного инструмента, который позволит сократить трудозатраты, повысить надежность операций и обеспечить соответствие нормативным требованиям.

Для реализации этой цели решается ряд взаимосвязанных задач: проводится анализ существующих решений с выявлением их слабых сторон, формируются требования к новой системе, проектируется архитектура с выделением модуля генерации, разрабатывается модель данных и интерфейс. Итогом становится комплексная оценка экономических параметров предлагаемого решения.

Методологическая основа исследования включает современные научные подходы: сравнительный анализ, моделирование бизнес-процессов с использованием нотации IDEF0, проектирование архитектуры, а также экономический анализ с расчетами затрат и эффективности. Источниками информации выступают научные публикации в области криптографии и автоматизации, техническая документация программных средств, нормативно-правовые акты в сфере информационной безопасности и официальная статистика, в частности данные Росстата.

Структура работы соответствует традиционной схеме, включающей введение, основное содержание четырех глав, заключение и список источников.

В первой главе анализируются существующие инструменты генерации сертификатов и определяются их ограничения.

Во второй части излагаются требования к системе и описываются ее ключевые функции.

Третья глава посвящена этапу разработки, включая архитектуру, модель данных и интерфейс.

Четвертая часть знакомит с экономической оценкой проекта: анализом затрат и расчетом эффективности.

Наконец, подводятся итоги исследований и намечаются перспективы дальнейших исследований.

Таким образом, предлагаемое исследование направлено на решение важной практической задачи: создать современный автоматизированный инструмент для формирования тестовых сертификатов, который сможет преодолеть существующие препятствия и соответствовать высоким стандартам.

1 Мир инструментов для создания сертификатов: возможности и проблемы

1.1 Инструменты и библиотеки: кто есть кто на рынке генерации сертификатов

Современный рынок предоставляет большое количество инструментов и библиотек для создания цифровых сертификатов. Каждое из этих решений имеет свой собственный набор характеристик, преимуществ и ограничений, что делает их пригодными для различных целей и условий применения. Ниже мы представим наиболее часто используемые инструменты для создания сертификатов и управления ими.

OpenSSL – одна из самых известных библиотек шифрования с широким спектром функций: генерация пар ключей, выдача сертификатов, подписание данных и поддержка различных стандартов шифрования [19]. Благодаря своей универсальности OpenSSL используется в самых разных проектах. Однако из-за командного интерфейса и необходимости глубоких технических знаний он подходит не всем пользователям, особенно в условиях ограниченной подготовки.

Certbot получил широкое распространение благодаря своей способности автоматически выдавать и обновлять сертификаты на веб-серверах [17]. Он тесно интегрирован с популярными веб-серверами (Apache, Nginx) для облегчения работы системных администраторов. Однако функциональность Certbot ограничена веб-средой, что снижает его применимость в других областях.

Для разработчиков Java в комплект Java Development Kit (JDK) входит инструмент Keytool. Он позволяет создавать ключи и сертификаты в формате Java KeyStore (JKS) и управлять ими [18]. Этот инструмент очень удобен в рамках платформы Java, но его сложно использовать вне ее.

Библиотека PyCryptodome предназначена для разработчиков, использующих Python. Она предоставляет множество возможностей для операций шифрования, включая создание сертификатов [20]. PyCryptodome легко интегрируется с проектами на Python, но, как и в случае с OpenSSL, вам нужно писать свои собственные скрипты и разбираться в криптографии.

Каждое решение имеет свои преимущества и подходит для конкретных условий. При построении нашей собственной системы мы используем комбинированный подход и полагаемся на возможности перечисленных инструментов для достижения максимальной гибкости и эффективности.

1.2 Сравнительная война: преимущества и недостатки существующих решений

Несмотря на разнообразие решений – OpenSSL, Certbot, Keytool, PyCryptodome, – каждое из них имеет ограничения, снижающие их эффективность в корпоративной среде. Эти инструменты ориентированы на конкретные сценарии и не всегда соответствуют конкретным условиям внутренних процессов нашей организации. Это увеличивает потребность в разработке нашего собственного решения для автоматизации, учитывающего специфику нашей организации.

OpenSSL предлагает обширную поддержку стандартов и гибкость, однако требует высокого уровня технической подготовки и ручной настройки, что замедляет внедрение и усложняет повседневную работу.

Certbot, удобный и простой в использовании, применим почти исключительно в контексте веб-серверов.

Keytool, полезный в Java-среде, не предназначен для кроссплатформенных решений.

PyCryptodome, хотя и гибкий, требует самостоятельной реализации логики автоматизации и не предоставляет готового интерфейса.

В таблице 1 приведено сравнение различных решений.

Таблица 1 – Сравнение существующих решений

Критерий	OpenSSL	Certbot	Keytool	PyCryptodome
Простота	Требует опыта	Удобен для веб-администраторов	Подходит Java-разработчикам	Требует знаний Python
Поддержка стандартов	Широкая	Только веб-сертификаты	JKS, X.509	Криптография, не для сертификатов
Интеграция	Гибкая, но сложная настройка	Интеграция с Apache, Nginx	Только для Java-среды	Интеграция в Python-приложения
Сфера применения	Общая криптография	Автоматическое обновление SSL	Управление сертификатами Java	Криптографические операции
Страна-разработчик	США	США	США	Франция

Существующие решения имеют ряд ограничений, которые делают их недостаточно подходящими для нашей организации. Нам необходимо решение, которое будет сочетать в себе простоту использования, универсальность и возможность интеграции с нашими внутренними системами.

Создание собственного инструмента позволит сформировать решение, оптимально соответствующее специфике нашей деятельности, обеспечить интуитивный интерфейс, автоматизировать рутинные операции и исключить критические зависимости от зарубежных технологий. Такой подход соответствует курсу на импортозамещение, определённому Минцифры России, и усиливает технологическую независимость на объектах критической информационной инфраструктуры [7].

Итак, сравнительный анализ подтверждает, что разработка собственного программного продукта – наиболее рациональный путь, позволяющий объединить лучшие стороны существующих решений, устранить их недостатки и адаптировать систему под конкретные нужды.

1.3 Почему существующие инструменты не подходят для тестовых задач?

Несмотря на функциональность OpenSSL, Certbot, Keytool и PyCryptodome, эти инструменты плохо приспособлены для задач, связанных с тестированием. Основная причина – ориентация на узкие прикладные случаи, без учёта требований к гибкости, автоматизации и пользовательской доступности, характерных для тестовых сред. Кроме того, в условиях современной реальности, когда приоритетом становится использование отечественного программного обеспечения и движение в сторону импортозамещения, зависимость от зарубежных решений становится не только неэффективной, но и рискованной [7].

Во-первых, большинство решений предполагают ручное управление или разработку дополнительных скриптов. Например, OpenSSL требует последовательного ввода команд в терминале и высокой технической компетенции. Certbot – ориентирован на автоматическое обновление SSL-сертификатов, но исключительно в контексте веб-приложений. Keytool ограничен экосистемой Java, а PyCryptodome не предоставляет инструментов «из коробки» для работы с сертификатами и требует самостоятельной настройки.

Во-вторых, все перечисленные инструменты не имеют дружественного графического интерфейса. Это затрудняет использование в команде, особенно при участии сотрудников без технической подготовки. Отсутствие визуальных средств управления сертификатами – от генерации до истечения срока действия – снижает оперативность и прозрачность процессов.

В-третьих, ни одно из решений не обеспечивает необходимой степени интеграции с внутренними системами организации. Автоматическая передача данных, уведомления, журналирование и контроль сроков требуют доработок, а зачастую – невозможны в рамках готовых решений. В условиях, когда

тестовая инфраструктура требует высокой адаптивности, такие ограничения становятся критичными.

Наконец, зависимость от зарубежного программного обеспечения увеличивает технологические риски. Санкции, ограничения доступа, отсутствие гарантий поддержки и обновлений – всё это делает использование иностранных инструментов нежелательным с точки зрения информационной безопасности и устойчивости. С учётом курса на импортозамещение, использование отечественных решений становится не только целесообразным, но и стратегически важным [15].

Таким образом, существующие инструменты не справляются с задачами тестовой генерации сертификатов из-за ограниченной автоматизации, отсутствия удобного интерфейса, слабой интеграции и зависимости от зарубежных разработок. Это подчёркивает необходимость создания собственной системы, учитывающей все эти факторы и способной обеспечить эффективное, безопасное и независимое управление тестовыми сертификатами.

2 Каким должно быть идеальное решение? Формулируем требования

2.1 Объект автоматизации

Объектом автоматизации в рамках данного проекта выступает процесс выпуска тестовых сертификатов, который на текущий момент осуществляется вручную и сопряжён с высокими трудо- и временными затратами. В его состав входит обработка поступающих заявок, обмен информацией между подразделениями и формирование самих сертификатов. Из-за участия человека на каждом этапе нередко возникают задержки и ошибки, снижающие эффективность и надёжность работы. Переход к автоматизированному подходу позволит ускорить выполнение операций и существенно повысить их точность, что особенно важно при соблюдении требований законодательства Российской Федерации, в том числе ФЗ № 63-ФЗ и Постановлению Правительства № 768 [8, 10].

Автоматизация данного процесса критически важна для организаций, работающих с данными, ограниченными в использовании или подлежащими защите по нормативно-правовым актам. Особенно это актуально для предприятий, участвующих в государственном оборонном заказе, где приоритет отдается строгому соблюдению сроков и безошибочной работе с конфиденциальной информацией. Ручной режим выпуска сертификатов не только замедляет выполнение задач, но и увеличивает риски нарушения регламентов. Поэтому автоматизация процесса выдачи тестовых сертификатов является центральной задачей проекта.

2.1.1 Модель процесса: Текущее состояние (AS-IS)

Чтобы понять текущее состояние процесса выдачи сертификата о тестировании и выявить узкие места, которые необходимо автоматизировать, необходимо подробно описать существующие бизнес-процессы. Это позволяет нам визуализировать, как выполняется работа в текущей среде,

какие этапы занимают больше всего времени и где возникают ошибки. Для моделирования процессов используется нотация IDEF0, которая помогает структурировать информацию и выделить ключевые элементы процесса.

Модель IDEF0 позволяет описать процесс выдачи сертификатов на высоком уровне, выделив основные функции, входные и выходные данные, а также механизмы и управление. Это помогает понять, как взаимодействуют различные отделы и какие ресурсы задействованы на каждом этапе.

Используя эту модель мы можем полностью понять текущее состояние (AS-IS) процесса и заложить основу для проектирования будущей системы автоматизации. Это важный шаг, он дает представление о том, какие изменения необходимы для повышения эффективности и надежности процесса выдачи сертификата о прохождении тестирования.

Элементы классификатора функциональной модели «Выпуск тестового сертификата»:

- пользователь – получение запроса на выпуск тестового сертификата,
- готовый сертификат в формате *.cer – итог работы модуля,
- локальные нормативно-правовые акты – внутренние нормы и правила АСТ ГОЗ, в соответствии с которыми ведется работа и оформляются документы,
- законодательство РФ – требования законодательства Российской Федерации по выпуску и оформлению открытых сертификатов электронной подписи,
- ответственный сотрудник за выпуск сертификата – сотрудник, осуществляющий коммуникацию с Пользователем и оформление документов для выпуска тестового сертификата,
- сотрудник отдела информационной безопасности – сотрудник, отвечающий за проверку данных о Пользователе и выпуск сертификата.

Иерархическое представление модели:

[A0] Выпуск тестового сертификата

- [A1] Создание запроса на выпуск тестового сертификата
- [A2] Проверка данных
- [A3] Создание сертификата
- [A4] Передача сертификата Пользователю

Для наглядного представления структуры процесса выпуска тестовых сертификатов разработаны графические модели: на Рисунке 1 показан первый уровень функциональной модели «Выпуск тестового сертификата» в нотации IDEF0, отображающий контекстную диаграмму процесса с основными входами, выходами и управляющими воздействиями, а на Рисунке 2 представлена декомпозиция блока A0 этой модели, детализирующая четыре ключевых подпроцесса - создание запроса (A1), проверку данных (A2), создание сертификата (A3) и его передачу пользователю (A4), что позволяет четко определить границы автоматизации, выявить узкие места, оптимизировать взаимодействие участников и сформировать требования к автоматизированной системе, обеспечивая полное понимание текущего состояния процесса (AS-IS) и направлений его совершенствования.

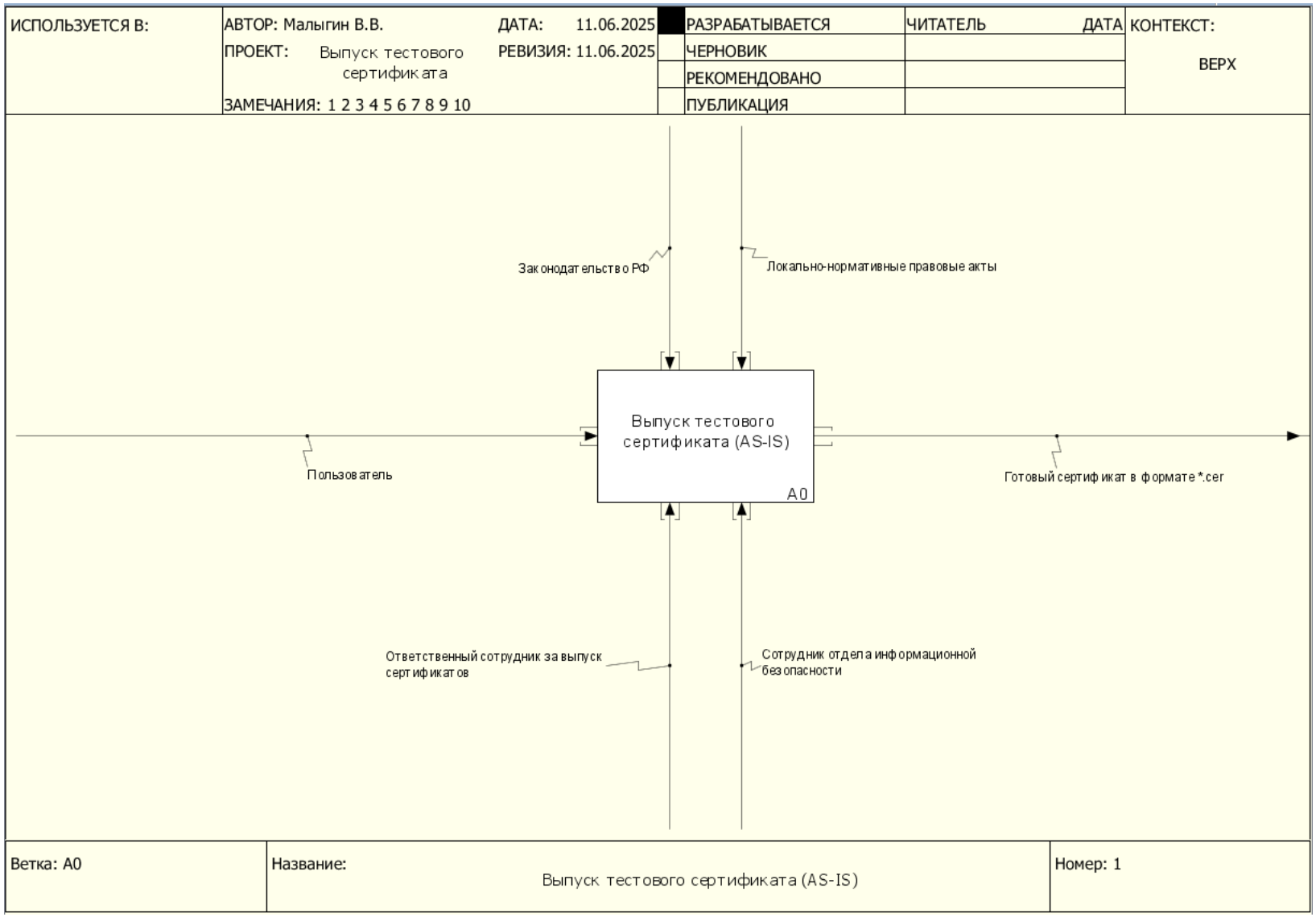

Рисунок 1 – Первый уровень функциональной модели «Выпуск тестового сертификата (AS-IS)»

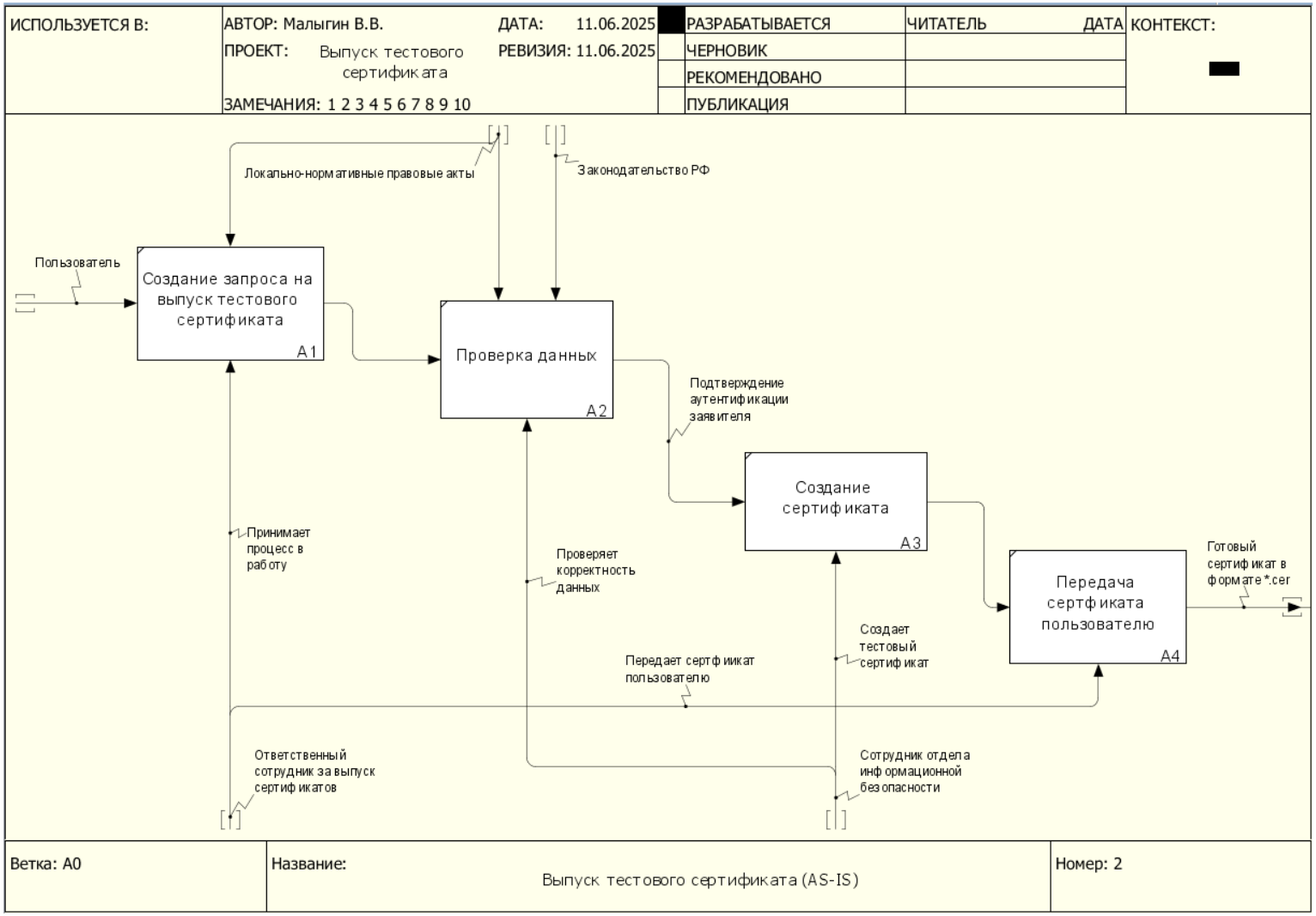

Рисунок 2 – Декомпозиция блока A0 функциональной модели «Выпуск тестового сертификата (AS-IS)»

2.1.2 Модель процесса: Будущее состояние (TO BE)

Для повышения эффективности процесса выдачи тестового сертификата, устранения задержек и ошибок, связанных с человеческим фактором, разработана модель будущего состояния (TO-BE), в которой ключевые этапы автоматизированы. Это обеспечивает сокращение времени выполнения операций, повышение точности и надёжности обработки, а также соблюдение нормативных требований законодательства Российской Федерации.

В автоматизированной модели пользователь формирует запрос на выпуск сертификата, заполняя контактные данные через предоставленный интерфейс. Полученные данные автоматически проверяются системой, и в случае их соответствия требованиям законодательства формируется тестовый сертификат. После успешной генерации сертификат передаётся в систему, а инициатор процесса уведомляется о его готовности.

Элементы классификатора функциональной модели «Выпуск тестового сертификата (TO-BE)»:

- запрос на выпуск сертификата – инициирующее событие, запускающее процесс обработки,
- контактные данные пользователя – набор сведений, необходимых для формирования сертификата,
- интерфейс системы (HTML-форма) – инструмент ввода и отправки данных,
- программный модуль обработки и генерации сертификатов (JavaScript + плагин КриптоПро) – обеспечивает проверку введённых данных, формирование и подписание сертификата,
- серверная инфраструктура / внутреннее ПО – обеспечивает выполнение фоновых процедур и взаимодействие компонентов,
- уведомление о готовности сертификата – информационное сообщение, направляемое пользователю после завершения процесса.

Иерархическое представление модели TO-BE:

[A0] Выпуск тестового сертификата

[A1] Обработка запроса

[A2] Автоматическая проверка данных

[A3] Генерация тестового сертификата

[A4] Уведомление и установка сертификата

Для наглядного представления автоматизированного процесса выпуска тестовых сертификатов были разработаны графические модели. На Рисунке 3 представлена контекстная диаграмма «Выпуск тестового сертификата (TO-BE)», содержащая входные и выходные данные, управляющие воздействия и механизмы реализации процесса. На Рисунке 4 показана декомпозиция блока A0, включающая четыре логически связанных подпроцесса: обработку запроса (A1), автоматическую проверку данных (A2), генерацию сертификата (A3) и установку с уведомлением пользователя (A4). Эта модель формирует целостное представление целевого состояния системы и служит основой для последующей реализации.

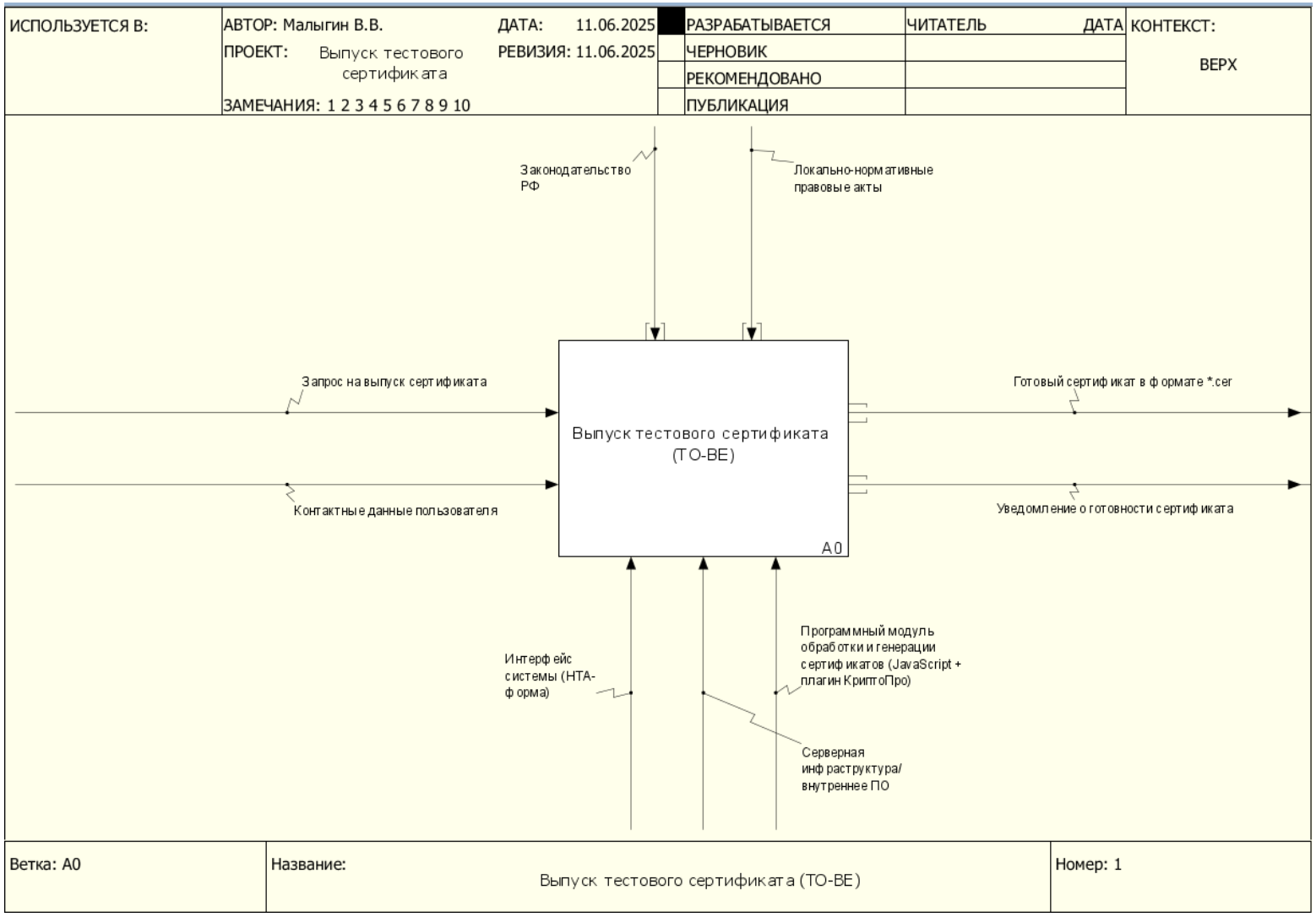

Рисунок 3 – Первый уровень функциональной модели «Выпуск тестового сертификата (TO-BE)»

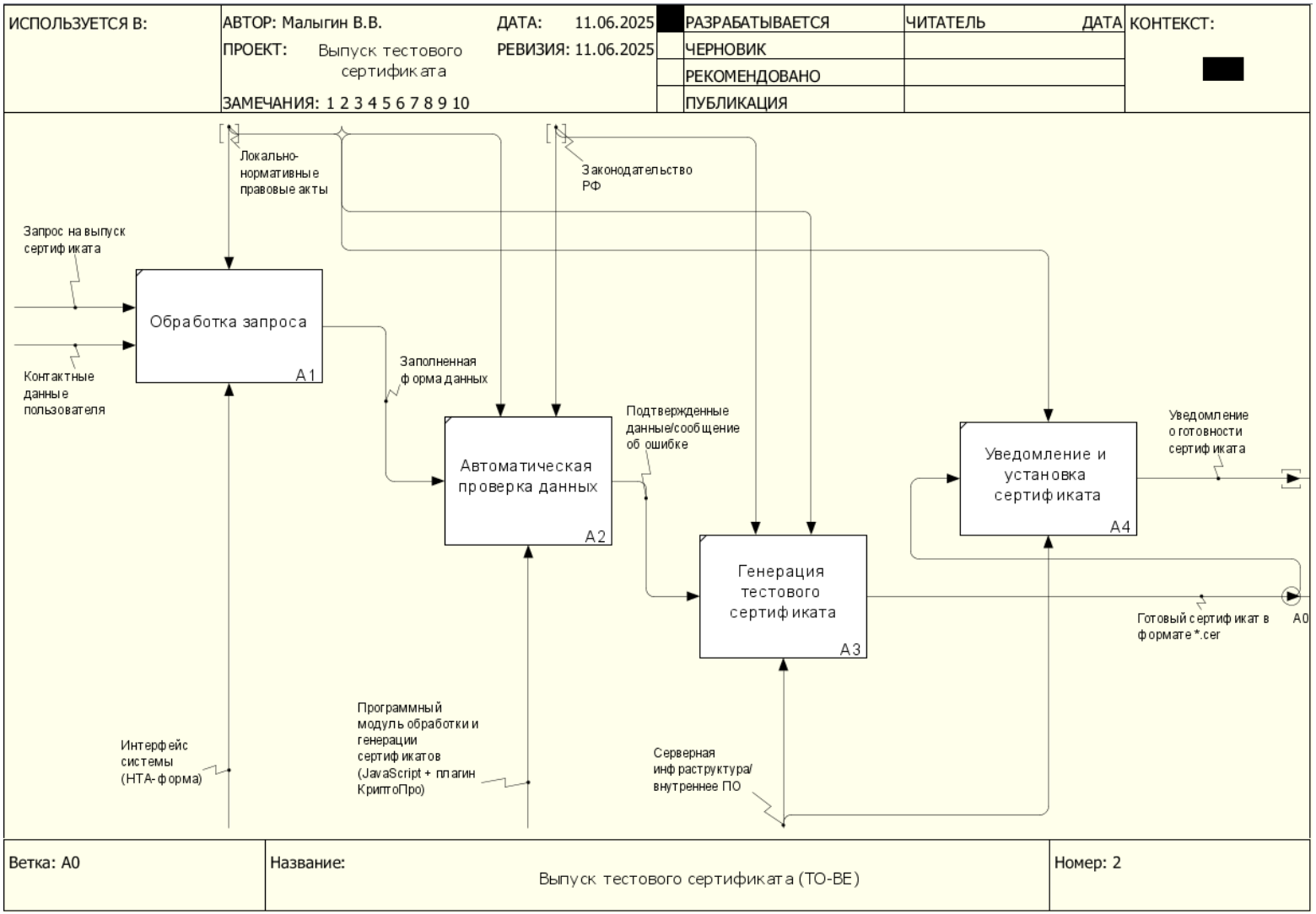

Рисунок 4 – Декомпозиция блока A0 функциональной модели «Выпуск тестового сертификата (TO-BE)»

2.1.3 Модель процессов AS–IS vs TO–BE.

Анализ текущего состояния процесса и проектирование его будущего состояния (TO-BE) является ключевым шагом во внедрении автоматизированного решения. Модель (AS-IS) позволяет нам охватить существующие бизнес-процессы, выявить их слабые стороны и понять, какие изменения необходимы для повышения эффективности. Модель TO-BE описывает, как эти процессы будут обрабатываться после внедрения автоматизации, какие этапы будут оптимизированы и какие новые функции появятся. [2]

Сравнение моделей AS-IS и TO-BE в таблице 2 наглядно показывает, как изменится процесс выдачи тестовых сертификатов после внедрения автоматизированного решения.

Таблица 2 – Сравнение AS-IS и TO-BE.

	AS–IS	TO–BE
Подготовка данных	Запрос на выпуск тестового сертификата поступает к ответственному сотруднику. Сотрудник заполняет форму запроса на выпуск сертификата, указывая необходимые данные о заявителе и целях использования сертификата. Сотрудник передает форму запроса на выпуск сертификата сотруднику отдела информационной безопасности.	Пользователь заполняет контактные данные для выпуска тестового сертификата.
Выпуск тестового сертификата	Сотрудник отдела информационной безопасности проверяет данные, указанные в форме запроса, и проводит аутентификацию заявителя. После подтверждения аутентификации заявителя, сотрудник отдела информационной безопасности создает тестовый сертификат в формате *.cer.	Программа проверяет введенные данные и, в случае положительного результата, заполняет сертификат.
Обработка результатов	Сотрудник отдела информационной безопасности передает тестовый сертификат ответственному сотруднику.	Сертификат в формате *.cer устанавливается в систему

2.2 Функционал мечты: что может сделать наше автоматизированное решение

Наше автоматизированное решение – это мощный инструмент, который не только упрощает, но и полностью меняет методику выдачи тестовых сертификатов. Основная цель - предоставить общую платформу, способную одновременно обрабатывать единичные запросы и масштабную генерацию в рамках сложных тестовых сценариев.

Система включает в себя интуитивно понятный интерфейс для подачи заявок, в котором пользователь задает параметры сертификата: алгоритм шифрования, назначение и другие настройки. Проверка введенных данных выполняется автоматически, что снижает риск возникновения ранних ошибок. Это особенно важно, когда нужно быстро создать несколько сертификатов с разными характеристиками. После проверки заявка передается во внутреннюю систему организации для обеспечения автоматического обмена данными между подразделениями. Все участники процесса будут получать уведомления в режиме реального времени – по электронной почте, через мессенджер или внутренние службы, – что может повысить прозрачность и ускорить координацию.

На этапе генерации система автоматически создает пару ключей и выдает сертификат на основе заданных параметров. Затем они подписываются тестовым удостоверяющим центром, что позволяет получить реальные данные для тестирования. При необходимости решение обеспечивает автоматическое продление сертификатов, что особенно удобно при длительных циклах тестирования, когда очень важно поддерживать актуальность данных.

Система также собирает и анализирует статистические данные: количество выданных сертификатов, тип и частоту их использования. Эта информация обобщается в визуальные отчеты, представленные в виде графиков и диаграмм, что облегчает оценку производительности и выявление

узких мест. Обнаружение аномалий (таких как неиспользуемые сертификаты) позволяет оптимизировать работу и снизить нагрузку на инфраструктуру организации. Таким образом, решение превратилось из простого инструмента в комплексную платформу для управления всем рабочим циклом сертификатов.

Автоматизация рутинных операций значительно сокращает трудозатраты, минимизирует человеческий фактор и повышает качество тестирования [3]. Решение становится незаменимым помощником для разработчиков, тестировщиков и специалистов по информационной безопасности, обеспечивая высокую гибкость и надёжность при работе с тестовыми данными. Это делает его не просто практичным, а стратегически важным компонентом для современных IT-команд.

Подводя итог, функциональные требования проекта должны охватывать автоматизацию следующих этапов:

- приём и обработка запросов,
- передача данных между подразделениями,
- выпуск и продление сертификатов,
- мониторинг и анализ использования.

2.3 Бизнес-видение: как решение изменит процессы тестирования

Автоматизированная система выпуска тестовых сертификатов – это решение, ориентированное на повышение эффективности внутренней ИТ-инфраструктуры, задействованной в разработке и тестировании. Её бизнес-ценность заключается в снижении эксплуатационных затрат, упрощении процессов и соблюдении требований информационной безопасности и нормативного соответствия [14].

Система обеспечивает централизованное управление всем циклом генерации сертификатов – от подачи заявки до выпуска готового тестового сертификата. Автоматизация операций повышает точность, минимизирует

влияние человеческого фактора и ускоряет подготовку тестовой среды для разработчиков и специалистов по контролю качества [5].

На этапе создания запроса пользователю предоставляется интерфейс с необходимыми параметрами – формат ключа, назначение сертификата и прочее. После валидации данные регистрируются и передаются на обработку. Генерация сертификата и его сохранение в формате *.cer происходят автоматически. Для подписания используется внутренний тестовый удостоверяющий центр, исключающий риск утечки или компрометации продуктивных данных [6].

Система включает модуль мониторинга, который фиксирует ключевые события рабочего цикла сертификатов: выпуск, срок действия, продление. Эти данные используются для формирования отчётов, анализа активности и выявления неактуальных сертификатов, способствуя эффективному использованию ресурсов.

Ключевые бизнес-цели автоматизации процесса включают:

- снижение трудозатрат и ускорение операций. Сокращается время на выполнение задач, уменьшается нагрузка на персонал и минимизируются ошибки, что приводит к росту производительности и снижению издержек;
- повышение стабильности и соответствия процессу. Строгое соблюдение процедур и правильность контрольных действий могут снизить вероятность нарушений и отклонений, особенно в областях, где предъявляются высокие требования к точности;
- эффективное управление истечением срока действия сертификата. Автоматические напоминания и отслеживание последних событий предотвращают нарушение сроков, снижают юридические риски и поддерживают доверие клиентов;
- анализ и принятие обоснованных решений. Собирая статистику, мы можем отслеживать загрузки, выявлять неиспользуемые сертификаты и

оптимизировать рабочие процессы для повышения прозрачности и эффективности управления;

– соблюдение регламентов и требований безопасности. Решение соответствует нормам, установленным для организаций, обрабатывающих конфиденциальные данные, что важно для участников гособоронзаказа. Это не только уменьшает уязвимости, но и улучшает репутацию компании.

Таким образом, данное решение позволяет не только автоматизировать технические задачи, но и способствует стратегическому развитию цифровых процессов внутри организации.

Автоматизация рутинных задач позволяет сократить потери времени, повысить уровень контроля и обеспечить надежную работу с тестовыми сертификатами. Решение сочетает в себе практическую выгоду и стратегическую важность, становясь важным элементом инфраструктуры ИТ-команд, работающих в условиях высоких требований к безопасности, скорости и качеству.

3 Создание автоматизированного решения: от идеи до реализации

3.1 Архитектура проекта: как мы создаём систему

Архитектура системы спроектирована с учётом ключевых требований к функциональности, масштабируемости и информационной безопасности. Она представляет собой набор взаимосвязанных компонентов, каждый из которых выполняет определённую роль в процессе выдачи тестовых сертификатов.

Основная цель архитектуры — автоматизировать все этапы от ввода данных и формирования запроса до анализа использования сертификатов, обеспечивая при этом гибкость, надёжность и совместимость с корпоративными стандартами безопасности.

Разработанное решение реализовано на базе технологии HTML Application (HTA) — специализированного формата от Microsoft для создания настольных приложений с использованием веб-технологий. Использование HTA обеспечивает глубокую интеграцию с операционной системой Windows и предоставляет доступ к ActiveX и COM-объектам. Это особенно важно при работе с криптографической инфраструктурой.

Несмотря на то, что браузер Internet Explorer официально выведен из эксплуатации в июне 2022 года [1], его поддержка ActiveX остаётся уникальной. В контексте проекта HTA и IE используются как временное решение для строго контролируемой среды исполнения, не требующее установки стороннего ПО. Такое ограничение критично в условиях повышенных требований к безопасности.

Выбор HTA и Internet Explorer обусловлен рядом объективных причин. Во-первых, в условиях, когда обеспечение защищённости информации возлагается на абонента, требуется использование решений, позволяющих строго контролировать среду исполнения. Информационная система специализированной электронной площадки АСТ ГОЗ, зарегистрированная в реестре российского ПО [12], соответствует требованиям, предъявляемым

государственным информационным системам 1 класса защищенности в соответствии с приказом ФСТЭК России от 11.02.2017 № 17 [9]. При этом аттестат соответствия распространяется исключительно на серверную часть системы и не охватывает средства, применяемые на стороне клиента. Таким образом, при подключении к АСТ ГОЗ пользователь самостоятельно выполняет требования по защите информации и обеспечивает безопасность своего рабочего места.

Использование HTA и Internet Explorer, несмотря на их устаревший статус, позволяет реализовать контролируемую и изолированную среду выполнения, которая легко настраивается с использованием групповых политик и внутренних инструментов Windows. Кроме того, эти технологии предоставляют уникальные возможности для взаимодействия с системными ресурсами без необходимости установки стороннего программного обеспечения – что критически важно в средах с ограниченным доступом и жёсткими регламентами безопасности.

Важно отметить, что в данной работе представлены только те технические детали, которые были официально согласованы с руководством организации. Это продиктовано внутренними нормативами в области информационной безопасности и режимом конфиденциальности.

Клиентская часть системы построена на языке JavaScript, что позволяет реализовать обработку пользовательского ввода, выполнять асинхронные операции (включая поддержку Promise с помощью полифиллов), а также взаимодействовать с криптографическими объектами через API КриптоПро. Такое решение обеспечивает гибкость разработки, масштабируемость и, что немаловажно, поддержку устаревших, но всё ещё активно используемых версий браузеров Internet Explorer 8 и 9. Информационное моделирование подобного рода решений активно описывается в профильной литературе, в частности, при использовании процессных нотаций, таких как IDEF0, что позволяет визуализировать взаимодействие модулей и компонентов системы [5].

Ключевым компонентом архитектуры является КриптоПро [4] ЭЦП Browser plug-in, предоставляющий доступ к интерфейсам CAdESCOM и (ограниченно) CAPICOM для выполнения операций электронной подписи. Поддерживается как синхронное создание объектов (CreateObject), так и асинхронное (CreateObjectAsync), что позволяет эффективно интегрировать криптографические функции в пользовательский интерфейс.

Автоматизация процесса обеспечивается за счёт встроенного модуля генерации сертификатов в формате *.cer с применением встроенных криптографических библиотек. Это позволяет сократить участие пользователя, минимизировать количество ошибок и повысить скорость обработки заявок.

Для обеспечения стабильной и надежной работы системы предусмотрены меры технической поддержки, включая регулярное резервное копирование данных, проведение нагрузочного тестирования, а также разработку и поддержание технической документации. Это особенно важно в условиях высоких требований к надежности, предъявляемых современными организациями. Отметим, что необходимость подобных решений обусловлена также ростом цифровизации процессов в государственных и частных структурах, на фоне чего существенно повышаются требования к качеству удостоверяющей инфраструктуры.

На Рисунке 5 представлена подробная архитектура системы, визуализирующая ключевые модули и взаимосвязи между ними в процессе генерации и выдачи тестовых сертификатов. Подобное представление соответствует методам системного анализа, применяемым в практике цифрового управления и стандартизации.

Рисунок 5 – Системная архитектура проекта

3.2 Модель данных: что храним и как обрабатываем

Информационная модель разработанной системы сформирована на основе ключевых сущностей, необходимых для автоматизации процесса выпуска тестовых сертификатов. Такая модель обеспечивает структурированное хранение информации, прозрачность логики обработки и возможность масштабируемой интеграции с криптографическими сервисами.

Основу модели составляют пять основных сущностей: Владелец**,** Запрос на сертификат**,** Сертификат**,** Информация о клиенте и Журнал событий**.**

Владелец — это физическое или юридическое лицо, на которого оформляется сертификат. Для каждого владельца сохраняются идентификационные сведения: фамилия, имя, отчество, наименование

организации (если применимо), электронная почта, должность, город, регион, страна, а также реквизиты: ИНН, ОГРН, ОГРНИП, СНИЛС и дополнительные неструктурированные поля. Также учитывается способ идентификации, применённый при создании запроса. Эти данные необходимы для однозначного связывания запроса с субъектом.

Запрос на выпуск сертификата содержит уникальный идентификатор, ссылку на владельца, дату подачи, статус обработки (например, В обработке» или «Выполнен»), а также сам запрос в формате CSR (Certificate Signing Request). В случае успешной валидации данных инициируется процедура генерации сертификата.

Сертификат создаётся на основе запроса и содержит информацию о дате выпуска, сроке действия, издателе, серийном номере и отпечатке (SHA-1/256). Также указывается путь к сохранённому файлу сертификата в формате .cer. Сертификат связан с конкретным запросом, что обеспечивает прослеживаемость и контроль рабочего цикла.

Уведомления информируют пользователей о событиях, таких как окончание срока действия сертификата или ошибки при обработке запроса. Каждое уведомление связано с конкретным сертификатом и пользователем, содержит дату отправки, тип сообщения и статус.

Информация о клиенте фиксирует параметры среды, в которой был сформирован запрос: тип браузера, его версию, операционную систему, использованный криптографический плагин и провайдер. Эти сведения позволяют выполнять технический аудит, а также обеспечивают совместимость с криптографическими модулями. Журналирование критически важно для аудита, диагностики и обеспечения информационной безопасности [16].

Журнал событий регистрирует ключевые действия в системе: создание запросов, генерацию сертификатов, ошибки и служебные операции. Каждая запись содержит дату и время события, уровень важности (INFO, WARNING, ERROR), описание действия и связанный идентификатор запроса. Такой

механизм критически важен для аудита, анализа инцидентов и обеспечения нормативного соответствия.

Связи между сущностями описывают логику работы системы: пользователь может отправить несколько запросов, каждый запрос – породить один сертификат, а сертификат может быть связан с множеством уведомлений. Логи, в свою очередь, отражают действия, совершаемые над всеми другими сущностями.

Работа системы начинается с ввода пользователем контактных данных и формирования запроса. После проверки данных на сервере создаётся запись, и при успешной валидации выполняется генерация сертификата с использованием криптографических библиотек.

Контроль сроков действия осуществляется через автоматизированные процедуры: система регулярно проверяет срок действия сертификатов и при необходимости формирует уведомления. Это позволяет минимизировать риски, связанные с просроченными или недействительными сертификатами.

Данные журнала событий служат также источником для аналитики и мониторинга: на их основе формируются отчёты о количестве выданных сертификатов, времени обработки запросов, типичных ошибках и активности пользователей.

На Рисунке 6 представлена модель данных, визуализирующая структуру хранения информации и взаимосвязь сущностей между собой. Она служит основой для проектирования базы данных и обеспечивает целостное представление бизнес-процесса.

Таким образом, предложенная модель данных обеспечивает достоверность информации, автоматизацию операций, надёжный аудит и гибкость при адаптации к новым требованиям, что делает её пригодной для применения как в закрытых корпоративных средах, так и в государственных информационных системах.

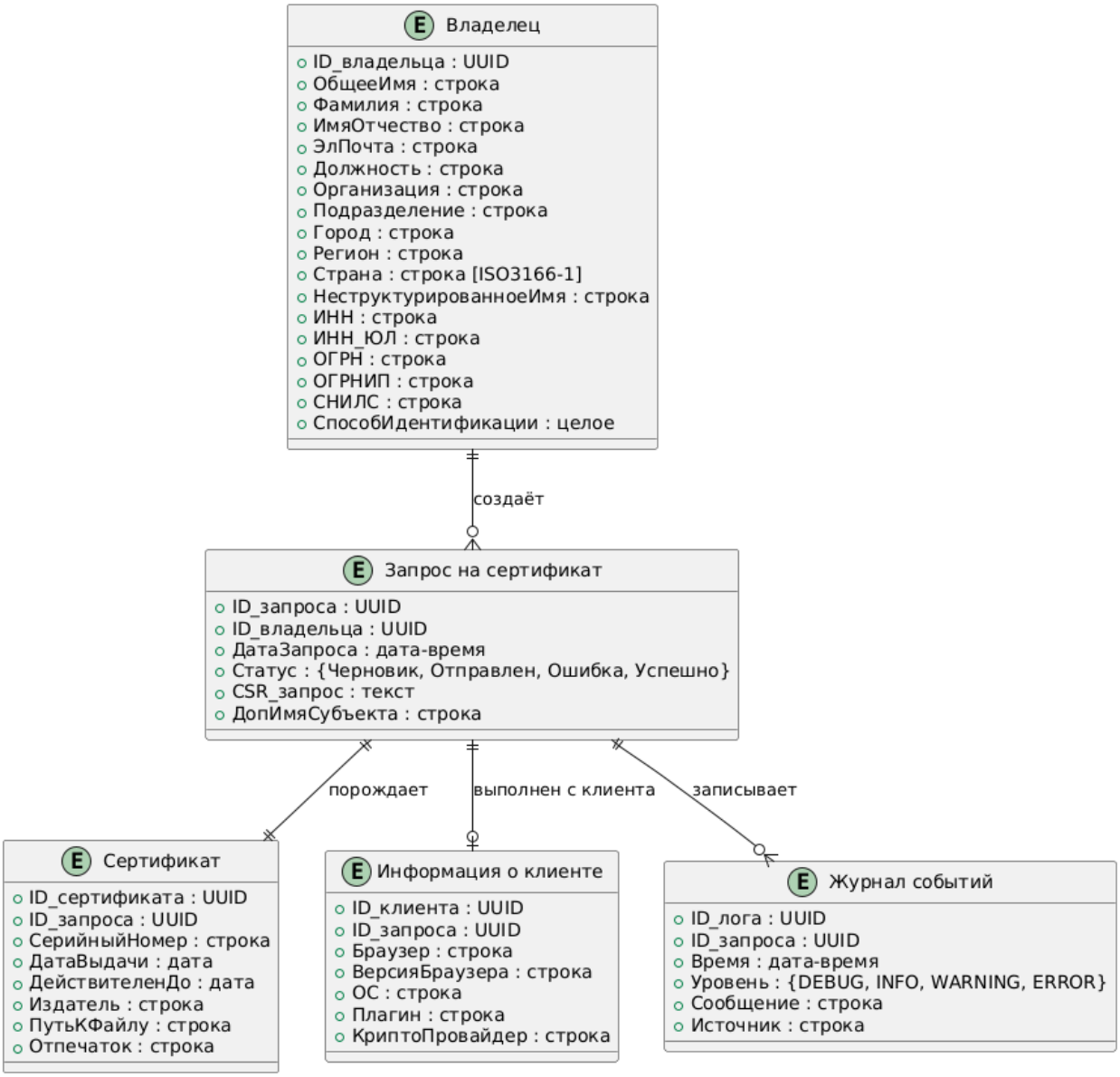

Рисунок 6 – Модель данных

3.3 Сердце системы: схема работы модуля генерации сертификатов

Модуль генерации сертификатов является ключевым элементом системы, отвечающим за создание тестовых сертификатов в формате *.cer. Его работа начинается с момента поступления запроса на создание нового сертификата. Пользователь, заполнив необходимые данные через веб-интерфейс, инициирует процесс, который проходит несколько этапов для обеспечения корректности и безопасности выпуска сертификата.

Первым шагом система проверяет наличие и корректность всех необходимых данных, предоставленных пользователем. Это включает проверку контактной информации, таких как электронная почта и телефон, а также других параметров, которые могут быть указаны в запросе. Если данные соответствуют требованиям, система генерирует уникальный идентификатор для нового сертификата. Этот идентификатор позволяет однозначно связать сертификат с конкретным запросом и пользователем, что важно для дальнейшего управления и работы.

После успешной проверки данных и создания уникального идентификатора система приступает к генерации сертификата в формате *.cer. Этот процесс включает формирование ключевой пары, подписание сертификата с использованием криптографических алгоритмов и сохранение файла сертификата в системе. Сертификат создаётся с учётом всех указанных параметров, что делает его пригодным для использования в тестовых сценариях.

Как только сертификат успешно создан, система уведомляет пользователя об этом. Уведомление может быть отправлено по электронной почте или через другие каналы связи, указанные пользователем. Это позволяет пользователю оперативно получить доступ к сертификату и использовать его для своих задач. В случае, если данные, предоставленные пользователем, оказались неполными или некорректными, система также уведомляет его об ошибке. В таком уведомлении указывается, какие именно данные требуют исправления, чтобы пользователь мог внести необходимые изменения и повторно отправить запрос.

Таким образом, модуль генерации сертификатов обеспечивает автоматизированный и надёжный процесс создания тестовых сертификатов. Он минимизирует участие человека в рутинных операциях, снижает вероятность ошибок и позволяет пользователям быстро получать необходимые сертификаты. Это делает систему удобной и эффективной для

всех участников процесса, будь то разработчики, тестировщики или специалисты по информационной безопасности.

На Рисунке 7 представлена детальная схема разрабатываемого модуля генерации сертификатов, которая наглядно демонстрирует его внутреннюю архитектуру и ключевые компоненты. Данная визуализация позволяет четко понять логику работы модуля, требования к его интеграции в общую систему и служит важным ориентиром при технической реализации решения.

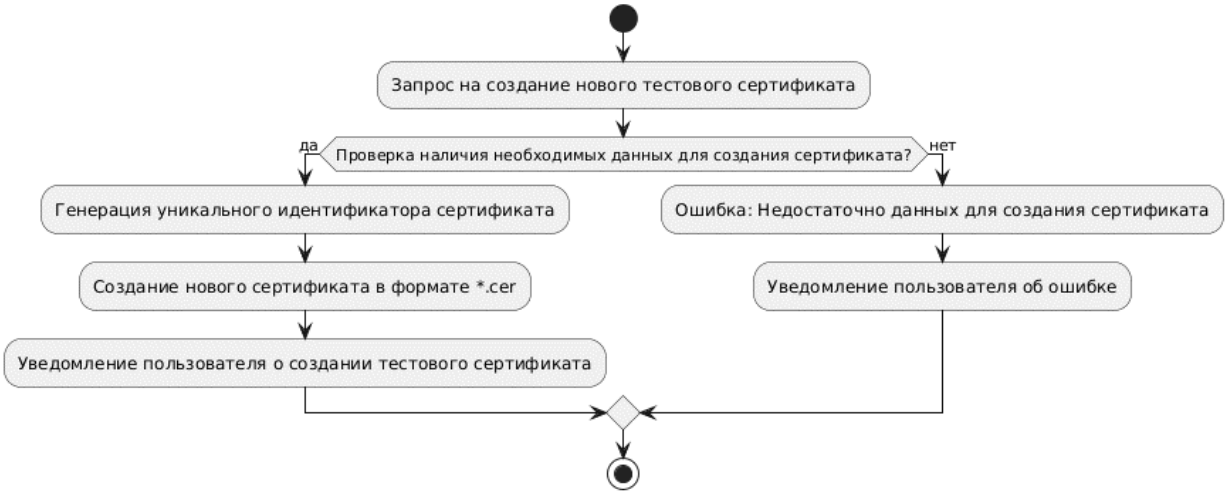

Рисунок 7 – Схема разрабатываемого модуля

3.4 Календарный план: планируем этапы реализации

Реализация проекта по разработке и внедрению автоматизированного решения для генерации тестовых сертификатов требует тщательного планирования и последовательного выполнения этапов. Каждый этап направлен на достижение конкретных целей, которые в совокупности обеспечивают создание функциональной, надёжной и удобной системы. Чёткое соблюдение сроков и контроль выполнения задач позволяют минимизировать риски, избежать задержек и обеспечить высокое качество конечного продукта.

Первый этап – анализ требований – занимает одну неделю и включает сбор и детальное изучение всех требований к системе. На этом этапе

определяются ключевые функции, которые должна выполнять система, а также потребности пользователей. Результатом этапа становится утверждённый список требований и функционала, который служит основой для дальнейшей работы.

Следующий этап – проектирование архитектуры – также рассчитан на одну неделю. Здесь разрабатывается общая архитектура системы, проектируются база данных и модули, такие как клиентская часть, серверная часть и модуль генерации сертификатов. На этом этапе создаются схемы архитектуры и модели данных, которые станут основой для разработки.

После завершения проектирования начинается этап разработки модулей, который занимает три недели. В это время создаются рабочие прототипы всех компонентов системы: веб-интерфейса для пользователей, серверной части для обработки запросов и модуля генерации сертификатов. Особое внимание уделяется интеграции модулей между собой, чтобы обеспечить их слаженную работу.

Затем следует этап тестирования и отладки, который длится еще одну неделю. На этом этапе проводятся функциональное, интеграционное и нагрузочное тестирование, чтобы выявить и устранить возможные ошибки. Тестирование позволяет убедиться, что система работает корректно и готова к внедрению.Финальный этап – внедрение и обучение –рассчитан на две недели. В это время система устанавливается в рабочую среду, настраивается и тестируется в реальных условиях. Параллельно проводится обучение пользователей, чтобы они могли эффективно работать с системой. Результатом этапа становится полностью работающая система, готовая к использованию, и подготовленные пользователи. Таким образом, календарный план реализации проекта охватывает все ключевые этапы – от анализа требований до внедрения и обучения. Каждый этап логически связан с предыдущим, что обеспечивает плавное продвижение к конечной цели.

В таблице 3 представлен календарный план, который отражает ключевые этапы, их продолжительность и ожидаемые результаты.

Таблица 3 – Календарный план

Этап	Результат	1 неделя	2 неделя	3 неделя	4 неделя	5 неделя	6 неделя	7 неделя	8 неделя
Определение целей и задач проекта	Чёткое понимание целей и задач проекта, а также их соответствие потребностям бизнеса								
Анализ текущего состояния и определение проблем	Выявление существующих проблем и возможностей для улучшения, определение приоритетов								
Разработка стратегии и плана реализации	Определение этапов выполнения проекта, установление сроков и ресурсов для каждого этапа								
Создание команды и распределение ролей	Формирование команды специалистов с учётом их навыков и опыта, назначение ответственных за каждый этап проекта								
Разработка технического задания и проектирование системы	Создание документа, описывающего требования к системе, и разработка архитектуры системы								
Реализация проекта	Выполнение всех работ согласно плану, контроль качества и соблюдение сроков								
Тестирование и отладка системы	Проверка работоспособности системы, обнаружение и устранение ошибок								
Внедрение и обучение пользователей	Подготовка пользователей к работе с новой системой, проведение обучающих мероприятий и технической поддержки								
Мониторинг и поддержка после внедрения	Отслеживание работы системы, решение возникающих проблем и внесение изменений при необходимости								
Завершение проекта и оценка результатов	Анализ результатов проекта, определение успехов и достижений, а также возможных улучшений и корректировок								

3.5 Интерфейс: простота и удобство в каждой детали

Интерфейс нашей системы разработан с учётом потребностей пользователей, чтобы сделать процесс работы с ней максимально простым, интуитивно понятным и удобным. Основной акцент сделан на минимизации сложных действий и обеспечении быстрого доступа к ключевым функциям. Пользовательский интерфейс позволяет легко заполнять формы, отслеживать статус запросов и скачивать готовые сертификаты, что делает систему доступной даже для тех, кто не обладает глубокими техническими знаниями.
Визуальное оформление интерфейса сочетает в себе минимализм и функциональность. Все элементы расположены логично, а навигация продумана так, чтобы пользователь мог быстро найти нужную информацию или выполнить необходимое действие. Например, форма для создания запроса на выпуск сертификата содержит только необходимые поля, а статус запроса отображается в реальном времени, что позволяет пользователю всегда быть в курсе текущего состояния процесса.

Одним из ключевых элементов интерфейса являются поля для ввода данных, которые позволяют пользователю указать всю необходимую информацию для создания запроса на выпуск сертификата. Данные элементы представлены в таблицах 4-7.

Таблица 4 – Основные данные

	Описание	Тип поля	Обязательность
Имя субъекта	Название сертификата, которое будет отображаться в системе	Текстовое поле	Обязательное
Фамилия	Фамилия пользователя, для которого выпускается сертификат	Текстовое поле	Обязательное
Имя	Имя пользователя, для которого выпускается сертификат	Текстовое поле	Обязательное
Отчество	Отчество пользователя, для которого выпускается сертификат	Текстовое поле	Обязательное

Таблица 5 – Данные организации

	Описание	Тип поля	Обязательность
Организация	Название организации, для которой выдается сертификат	Текстовое поле	Обязательное
Подразделение	Название подразделения организации	Текстовое поле	Обязательное
Должность	Должность субъекта, для которого выдается сертификат	Текстовое поле	Обязательное

Таблица 6 – Идентификационные данные

	Описание	Тип поля	Обязательность
ИНН	ИНН физического лица	Текстовое поле с валидацией (12 цифр для физ. лиц)	Обязательное
ИНН юр.лица	ИНН только для юридических лиц	Текстовое поле с валидацией (10 цифр)	Необязательное (заполняется только для юр. лиц)
ОГРН	ОГРН юридического лица	Текстовое поле с валидацией (13 цифр)	Необязательное (заполняется только для юр. лиц)
СНИЛС	СНИЛС физического лица	Текстовое поле с валидацией (11 цифр)	Обязательное для физ. лиц

Таблица 7 – Параметры ключа

Описание	Тип поля	Варианты выбора	Обязательность
Создание ключа	Радиокнопка	- Создать новый набор ключей - Использовать существующий набор ключей	Обязательное
CSP (Cryptographic Service Provider)	Выпадающий список	- Crypto-Pro GOST R 34.10-2012 Cryptographic Service Provider - Crypto-Pro GOST R 34.10-2012 Strong Cryptographic Service Provider	Обязательное
Использование ключей	Радиокнопка	- Exchange - Подпись - Оба	Обязательное
Размер ключа	Текстовое поле	- 512 (стандартное минимальное и максимальное значение)	Обязательное
Имя контейнера ключа	Радиокнопка	- Автоматическое имя контейнера ключа - Заданное пользователем имя контейнера ключа	Обязательное
Экспортируемость ключа	Чекбокс	Пометить ключ как экспортируемый	Необязательное
Хранилище сертификата	Чекбокс	Использовать локальное хранилище компьютера для сертификата (с пояснением)	Необязательное

Центральным элементом интерфейса является кнопка действия, которая выполняет ключевую функцию – отправку запроса на выпуск сертификата. Описание кнопки приведено в таблице 8.

Таблица 8 – Описание кнопки действия.

Описание	Действие	Состояния кнопки	
		Активна	Неактивна
Кнопка для подтверждения ввода данных и запуска процесса генерации сертификата	При нажатии система проверяет заполнение всех обязательных полей и валидность данных. Если данные корректны, сертификат генерируется и предоставляется для скачивания в формате *.cer	Все обязательные поля заполнены корректно.	Обязательные поля не заполнены или содержат ошибки

Валидация данных – это важный этап работы системы, который обеспечивает корректность введённой пользователем информации перед отправкой запроса на выпуск сертификата. Этот процесс включает проверку обязательных полей, контроль формата данных и своевременное уведомление пользователя об ошибках, что позволяет избежать некорректных запросов и повышает надёжность системы.

Все обязательные поля должны быть заполнены перед нажатием кнопки "Выдать". Если какое-то из обязательных полей остаётся пустым, система отображает предупреждение, указывающее на необходимость его заполнения. Это гарантирует, что пользователь не пропустит важные данные, необходимые для обработки запроса.

Формат данных строго контролируется, при этом валидация подразумевает исключительно проверку соответствия формату (структуре) данных без обращения к сторонним источникам или базам данных. Для ИНН предусмотрена проверка на количество цифр: 12 цифр для физических лиц и 10 цифр для юридических лиц. ОГРН должен содержать 13 цифр, СНИЛС – 11 цифр. Такой подход обеспечивает оперативную обработку запросов при соблюдении базовых требований к формату данных, сохраняя высокую производительность системы.

Ниже представлены скриншоты интерфейса (Рисунок 8 – Рисунок 16), которые наглядно демонстрируют его основные элементы и функциональные возможности.

Рисунок 8 – Скриншот интерфейса разработанного модуля «Выпуск тестового сертификата»

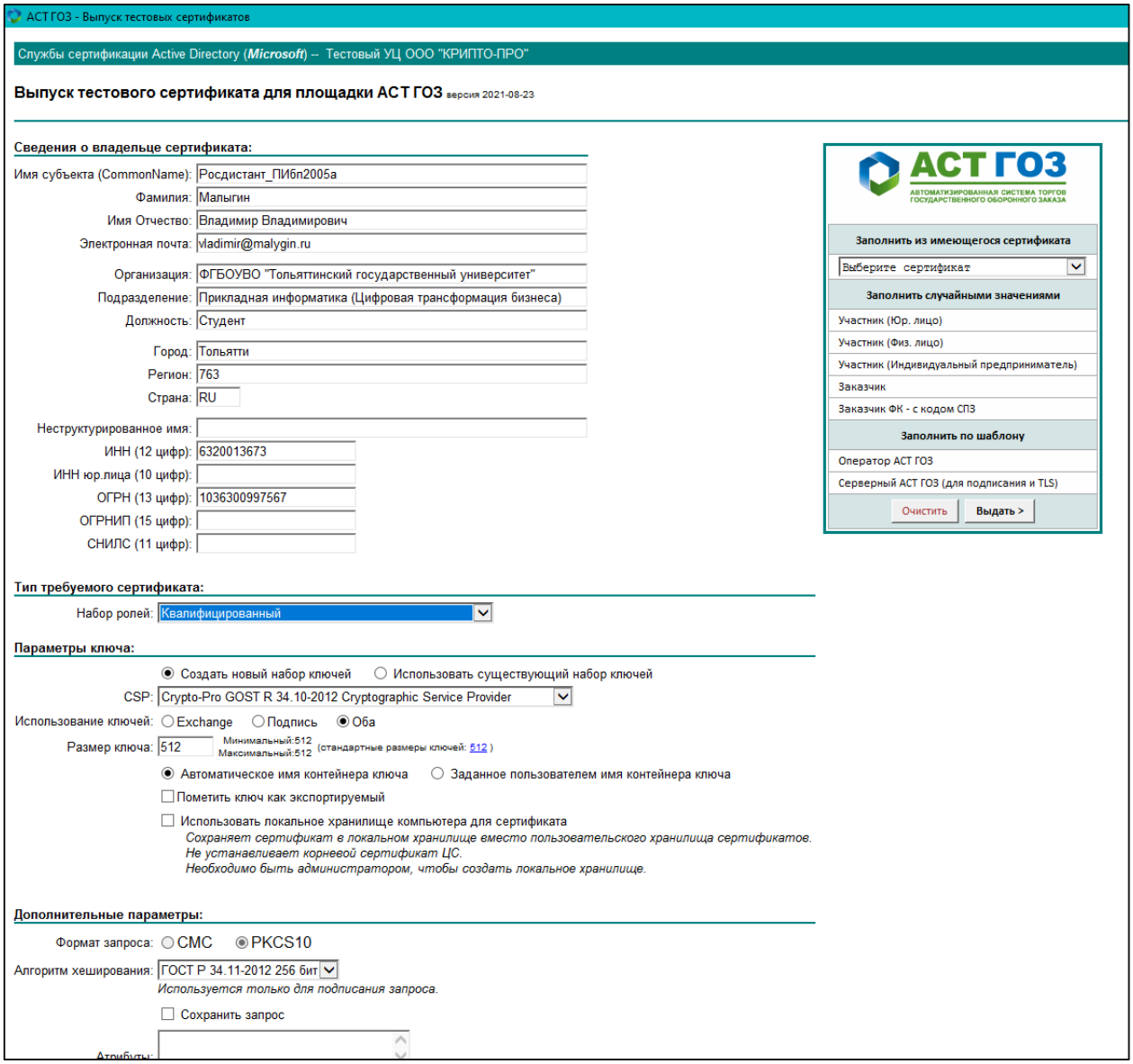

Рисунок 9 – Скриншот заполненных полей разработанного модуля «Выпуск тестового сертификата»

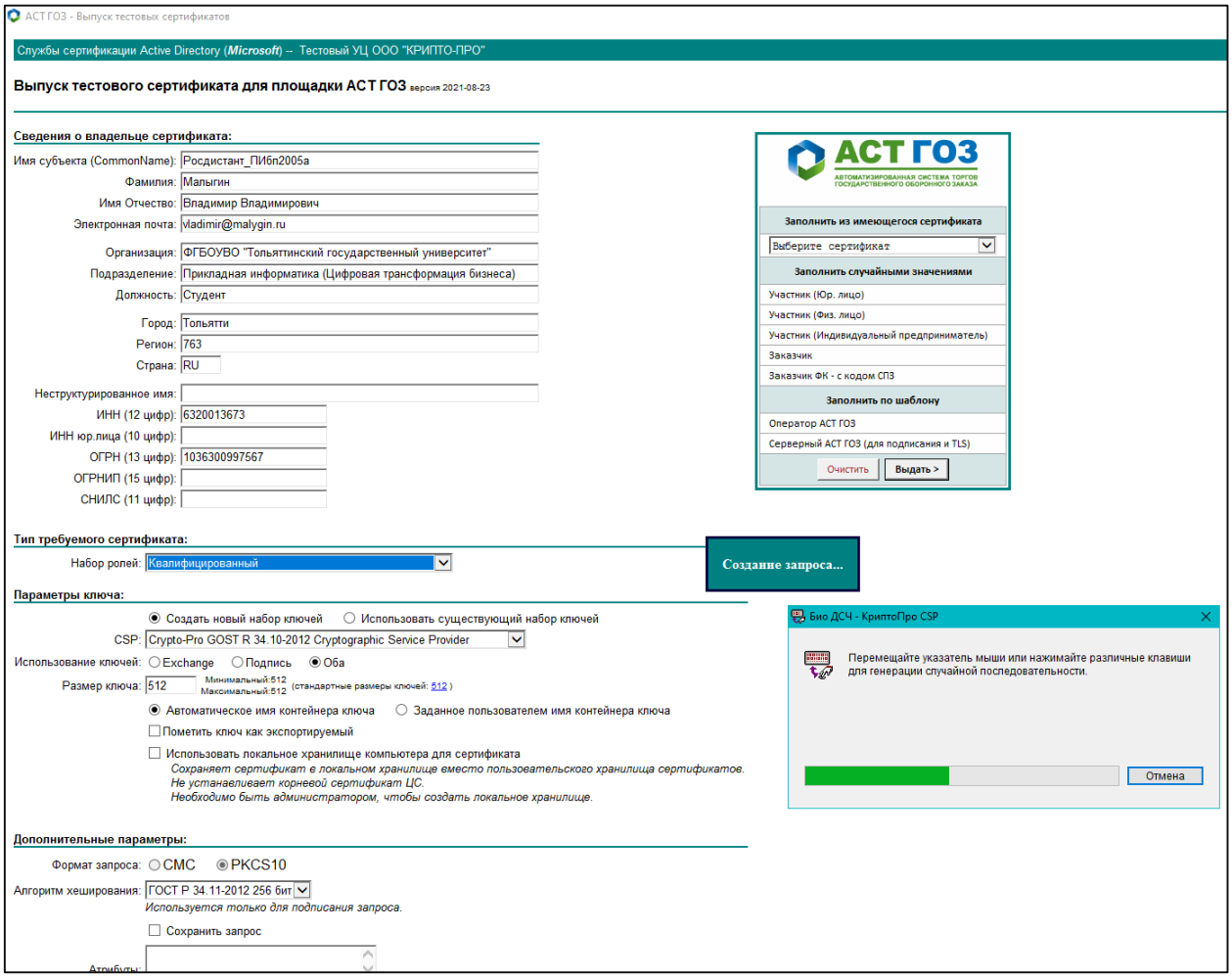

Рисунок 10 – Скриншот генерации случайной последовательности при создании тестового открытого сертификата

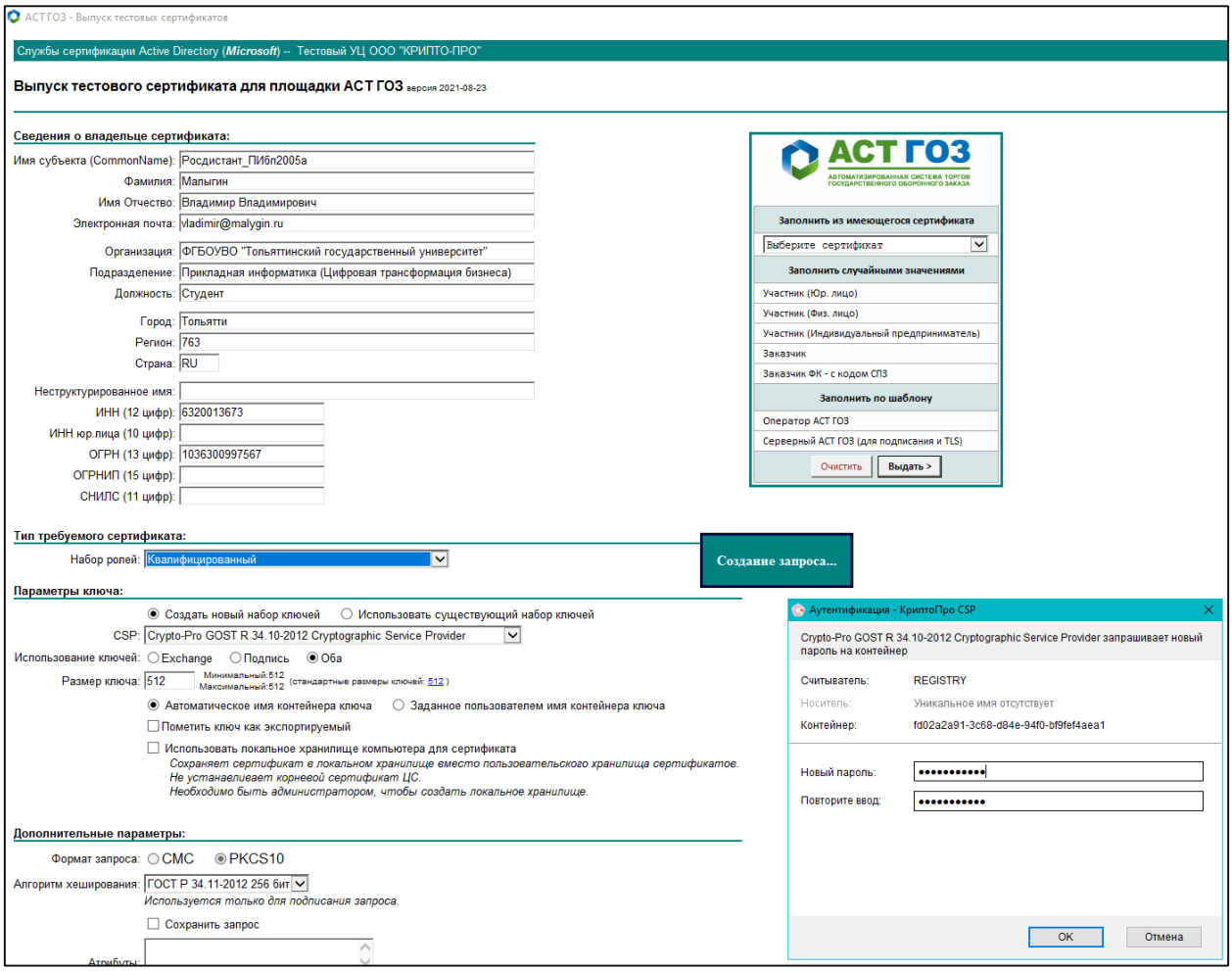

Рисунок 11 – Запрос создания ПИН-кода для защиты созданного контейнера закрытого ключа

Рисунок 12 – Скриншот выдачи тестового открытого сертификата

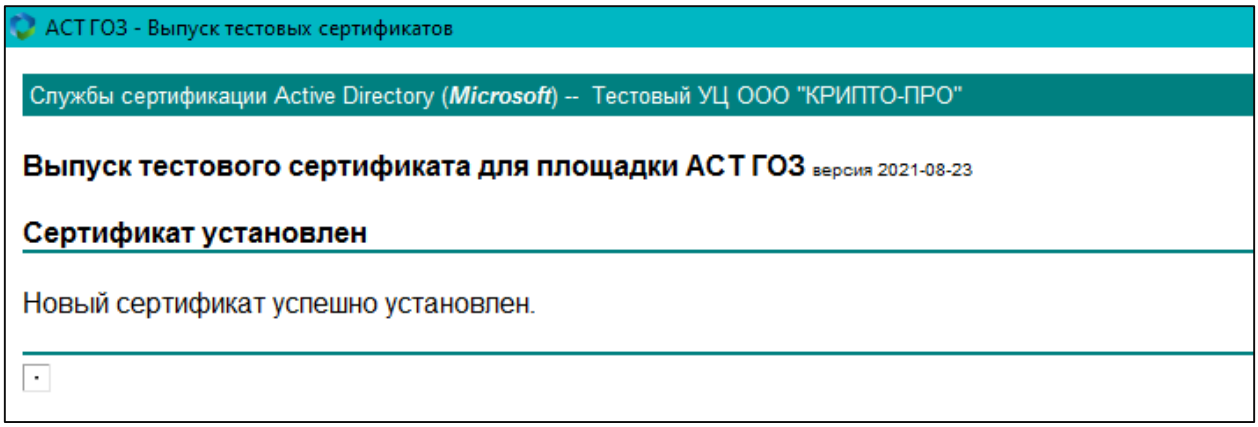

Рисунок 13 – Скриншот сообщения о успешной установки сертификата

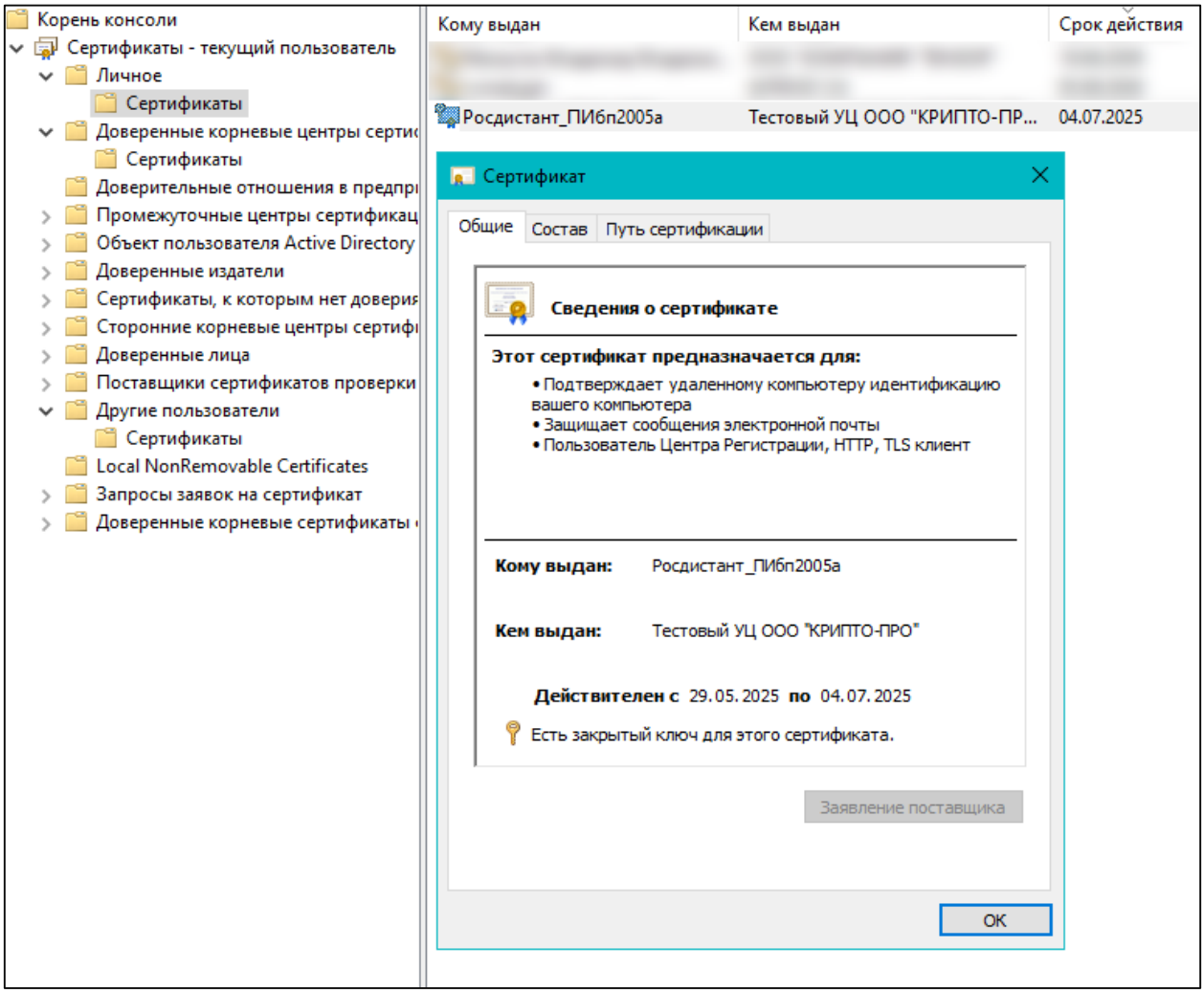

Рисунок 14 – Скриншот окна выданного сертификата – Вкладка “Общие”

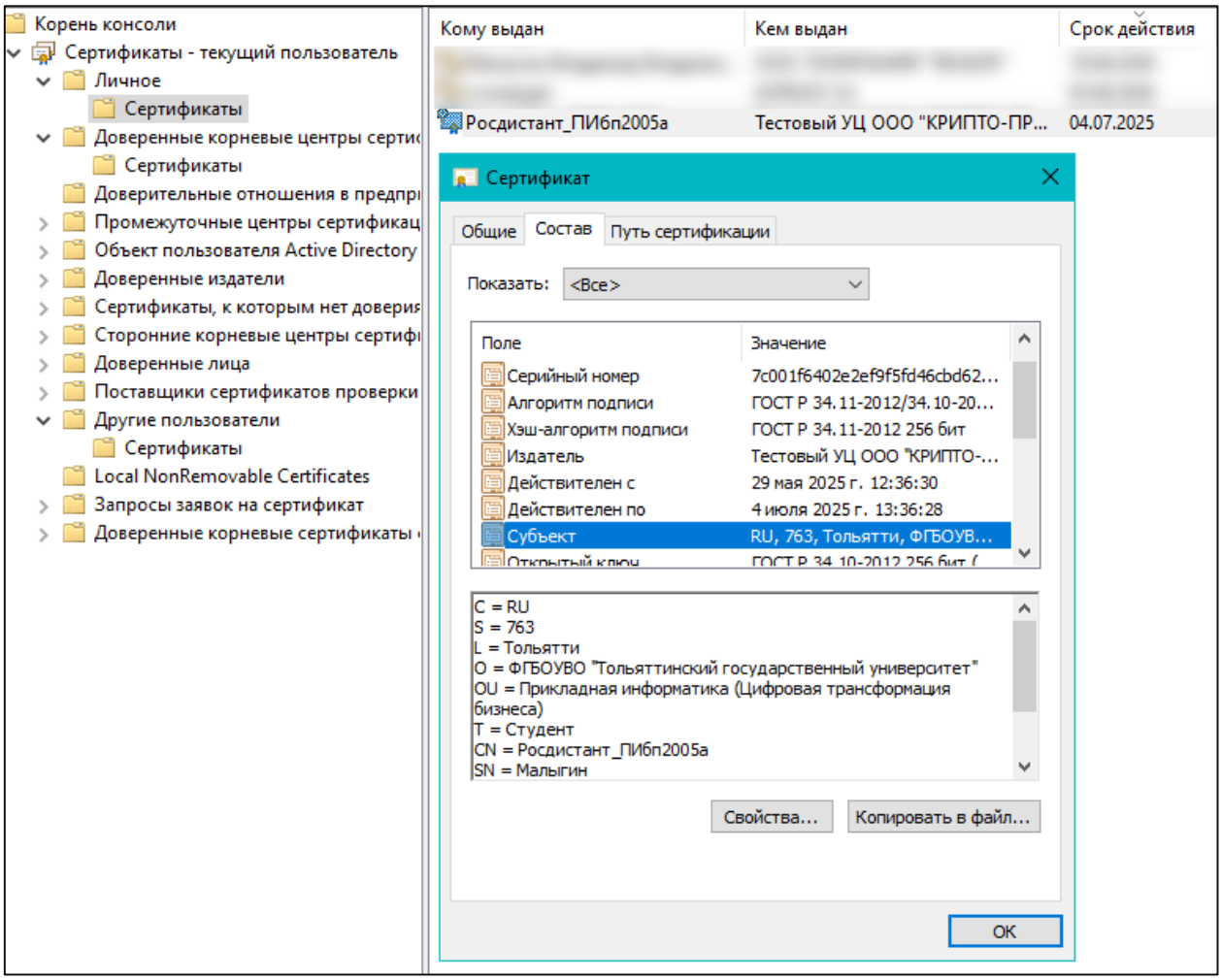

Рисунок 15 – Скриншот окна выданного сертификата – Вкладка “Состав”

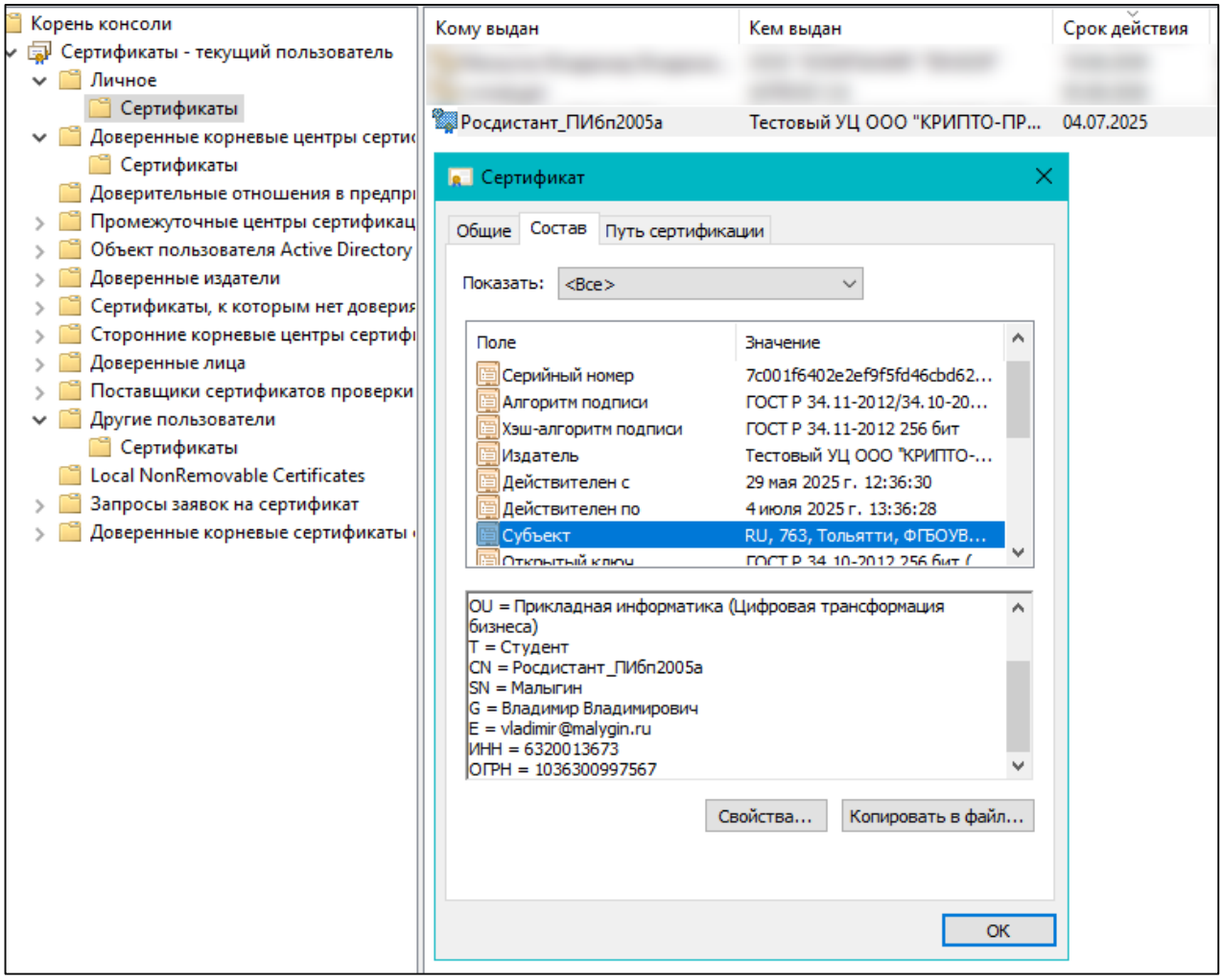

Рисунок 16 – Скриншот окна выданного сертификата – Вкладка “Состав”

4 Экономика проекта: сколько стоит автоматизация?

4.1 Затраты на разработку: во что нам обойдётся создание решения

Создание автоматизированного решения для генерации тестовых сертификатов требует не только временных, но и финансовых ресурсов. Чтобы оценить общую стоимость проекта, необходимо учесть все этапы разработки, начиная от проектирования и заканчивая внедрением и поддержкой системы. Это позволит не только спланировать бюджет, но и избежать непредвиденных расходов в процессе реализации.

Основные затраты связаны с разработкой программного обеспечения, которая включает создание клиентской и серверной частей, модуля генерации сертификатов, а также интеграцию всех компонентов в единую систему. Кроме того, необходимо учитывать расходы на облачные услуги или серверное оборудование, которые обеспечат хранение данных и стабильную работу системы.

Тестирование и внедрение также являются важными этапами, требующими финансовых вложений. Тестирование позволяет выявить и устранить ошибки, а внедрение включает установку системы, настройку и обучение пользователей. Наконец, нельзя забывать о ежегодных затратах на поддержку, которые включают обновление системы, устранение возможных сбоев и техническую помощь пользователям.

Ниже представлен расчёт затрат, который охватывает все ключевые статьи расходов, включая разработку программного обеспечения, облачные услуги, серверное оборудование, тестирование и внедрение, а также годовые затраты на поддержку.

Разработка программного обеспечения:

- заработная плата разработчиков: 500 000 руб.,
- лицензии и инструменты: 100 000 руб.

$$500\ 000 + 100\ 000 = 600\ 000\ руб. \quad (1)$$

Инфраструктура:

- серверное оборудование: 200 000 руб.,
- облачные услуги: 50 000 руб.

$$200\ 000 + 50\ 000 = 250\ 000\ руб. \quad (2)$$

Тестирование и внедрение:

- заработная плата тестировщиков: 100 000 руб.,
- обучение сотрудников: 30 000 руб.

$$100\ 000 + 30\ 000 = 130\ 000\ руб. \quad (3)$$

Эксплуатация и поддержка:

- годовые затраты на поддержку: 120 000 руб.

Итого затраты на проект:

$$\text{Разработка ПО} + \text{Инфраструктура} + \text{Тестирование и внедрение} + \text{Эксплуатация и поддержка} \quad (4)$$

$$600\ 000 + 250\ 000 + 130\ 000 + 120\ 000 = 1\ 100\ 000\ руб. \quad (5)$$

4.2 Эффективность и выгоды: как проект окупит себя

Внедрение автоматизированного решения для генерации тестовых сертификатов не только упрощает процессы, но и приносит значительную экономическую выгоду. Чтобы оценить эффективность проекта, необходимо рассчитать, как автоматизация повлияет на снижение затрат и повышение

производительности. Основной источник экономии – это сокращение времени, которое сотрудники тратят на обработку запросов вручную, что позволяет высвободить ресурсы для выполнения других задач.

Для расчёта экономии мы используем данные о средней заработной плате по России за 2024 год, предоставленные Росстатом. Это связано с тем, что конкретные данные о заработной плате в организации являются коммерческой тайной и не подлежат разглашению. Согласно данным Росстата, средняя заработная плата в 2024 году составляет 87 952 рубля в месяц [13]. На основе этой информации можно рассчитать стоимость одного часа работы сотрудника, что позволит оценить, сколько средств компания тратит на ручную обработку запросов и какую экономию можно получить за счёт автоматизации.

Для расчёта принято округлённое значение - 2000 запросов в год, что соответствует примерно 8 запросам в рабочий день (при 250 рабочих днях в году) [11]. Следует отметить, что эти данные являются оценочными, так как точные показатели организации составляют коммерческую тайну и не подлежат разглашению.

В текущих условиях сотрудник тратит в среднем 30 минут (0,5 часа) на обработку одного запроса вручную. Это включает проверку данных, создание сертификата и отправку его пользователю. При автоматизации процесса время на обработку одного запроса сокращается до минимума, что позволяет значительно снизить затраты на оплату труда.

Расчёт экономии включает несколько этапов. Сначала оценивается стоимость одного часа работы сотрудника, затем рассчитывается экономия на одном запросе за счёт сокращения времени обработки. После этого можно определить годовую экономию, учитывая общее количество запросов, которые обрабатываются в течение года.

Таким образом, автоматизация не только повышает скорость и точность обработки запросов, но и приносит ощутимую финансовую выгоду. Это делает

проект не только технологически эффективным, но и экономически обоснованным.

Ниже представлен подробный расчёт, который включает оценку стоимости одного часа работы сотрудника, экономии на одном запросе и годовой экономии.

Оценка стоимости одного часа работы сотрудника:

- средняя заработная плата в месяц: 87 952 руб.,
- количество рабочих часов в месяц (при 40-часовой рабочей неделе):

$$\text{Часов в месяц} = 40 \text{ часов/неделя} \times 4 \text{ недели} = 160 \text{ часов} \quad (6)$$

- стоимость одного часа работы:

$$87\,952 \text{ руб.} / 160 \text{ часов} = 549{,}7 \text{ руб./час} \quad (7)$$

Оценка экономии на одном запросе:

- время на обработку одного запроса: 0.5 часа.,
- экономия на одном запросе:

$$0.5 \text{ часа} \times 549{,}7 \text{ руб./час} = 274{,}85 \text{ руб.} \quad (8)$$

Годовая экономия:

- в год обрабатывается 2 000 запросов,
- годовая экономия:

$$2\,000 \text{ запросов} \times 274{,}85 \text{ руб./запрос} = 549\,700 \text{ руб.} \quad (9)$$

После оценки затрат на разработку и внедрение автоматизированного решения, а также расчёта потенциальной экономии, важно сравнить эти показатели, чтобы определить, насколько быстро проект окупится. Сравнение экономии с затратами позволяет понять, какую финансовую выгоду принесёт автоматизация, и оценить её целесообразность для организации.

Сравнение экономии с затратами:

- затраты на проект: 1 100 000 руб.,
- годовая экономия: 549 700 руб.

Для завершения анализа необходимо рассчитать срок окупаемости проекта. Этот показатель отражает, за какой период времени экономия от внедрения системы покроет все затраты на её создание. Чем короче срок окупаемости, тем быстрее проект начнёт приносить чистую прибыль, что делает его более привлекательным с точки зрения инвестиций.

Срок окупаемости:

$$\text{Затраты на проект} / \text{Годовая экономия} \quad (10)$$

$$1\,100\,000 / 549\,700 = 2{,}001 \text{ года} \quad (11)$$

Результаты расчётов показывают, что проект окупится за чуть менее чем 2 года. Это достигается за счёт значительного сокращения трудозатрат сотрудников, которые ранее тратили время на ручную обработку запросов.

Таким образом, проект окупается за чуть более года благодаря экономии на оплате труда. Это делает его экономически эффективным несмотря на то, что он не приносит прямого дохода. Автоматизация не только снижает операционные издержки, но и повышает производительность, что подтверждает целесообразность внедрения данного решения.

Заключение

Проведённое исследование убедительно подтвердило необходимость разработки отечественного автоматизированного решения, предназначенного для генерации тестовых сертификатов. В рамках работы был осуществлён всесторонний анализ существующих на рынке инструментов, позволяющий детально оценить их функциональные возможности, удобство использования и степень соответствия современным требованиям безопасности. В результате анализа были выявлены ключевые недостатки большинства доступных решений, среди которых особенно выделяются сложность внедрения и эксплуатации, ограниченность функционала, недостаточная адаптация к российским нормативам, а также высокая зависимость от зарубежных технологий и разработок, что в современных условиях несёт дополнительные риски.

На основании полученных данных были сформулированы исчерпывающие требования к новому программному продукту. В их числе – обеспечение полной работы с тестовыми сертификатами: от первичной подачи запроса и автоматической проверки введённых данных до генерации сертификатов и последующего анализа их применения. Особое внимание было уделено вопросам удобства взаимодействия пользователя с системой, надёжности работы, масштабируемости.

Разработанное в рамках проекта решение обладает рядом важных и значимых преимуществ, выгодно отличающих его от аналогов. Во-первых, особый акцент был сделан на создании интуитивно понятного и эргономичного пользовательского интерфейса, что делает систему доступной для специалистов с различным уровнем подготовки – от начинающих сотрудников до опытных экспертов. Во-вторых, внедрение механизмов автоматической валидации данных и генерации сертификатов позволило существенно повысить производительность труда: среднее время обработки одного запроса сократилось с 30 минут до всего 3–5 минут. В-третьих,

реализована полноценная интеграция с внутренними корпоративными системами, что обеспечивает непрерывность бизнес-процессов и поддержание актуальности данных в режиме реального времени.

Проведённый экономический анализ продемонстрировал высокую эффективность предложенного решения. При общих затратах на разработку в размере 1 100 000 рублей, годовая экономия благодаря внедрению системы составила 549 700 рублей, что обеспечивает срок окупаемости проекта менее двух лет. Ключевыми факторами экономической эффективности стали значительное снижение трудозатрат сотрудников, уменьшение количества ошибок при обработке заявок, а также оптимизация процессов взаимодействия между подразделениями.

Внедрение разработанного решения позволит организации значительно повысить эффективность процессов, связанных с обработкой и использованием тестовых сертификатов, обеспечить полное соответствие требованиям по защите информации и значительно снизить операционные расходы. Особенно высокую актуальность система приобретает для государственных учреждений, финансовых организаций, предприятий оборонно-промышленного комплекса и компаний, участвующих в выполнении государственного оборонного заказа.

Таким образом, проведённая работа наглядно демонстрирует, что создание специализированных автоматизированных решений для работы с цифровыми сертификатами является важным и перспективным направлением развития отечественных информационных технологий. Реализация подобных проектов способствует укреплению технологической независимости страны, повышению уровня национальной информационной безопасности, а также созданию конкурентных преимуществ российских организаций в условиях стремительно развивающейся цифровой экономики и глобальной трансформации бизнес-процессов.

Список используемой литературы и используемых источников

1. Будущее Internet Explorer на Windows 10 в Microsoft Edge [Электронный ресурс]. URL: https://blogs.windows.com/windowsexperience /2021/05/19/the-future-of-internet-explorer-on-windows-10-is-in-microsoft-edge/ (дата обращения: 15.05.2025).
2. Долганова, О. И. Моделирование бизнес-процессов: учебник и практикум для вузов / О. И. Долганова, Е. В. Виноградова, А. М. Лобанова; под редакцией О. И. Долгановой. – 2-е изд., перераб. и доп. – Москва: Издательство Юрайт, 2025.
3. Каменнова, М. С. Моделирование бизнес-процессов: учебник и практикум для вузов / М. С. Каменнова, В. В. Крохин, И. В. Машков. – Москва : Издательство Юрайт, 2025.
4. КриптоПРО [Электронный ресурс]. URL: https://www.cryptopro.ru/ (дата обращения: 15.05.2025).
5. Лёвина А.И. Моделирование бизнес-процессов: Учебное пособие/ А.И. Лёвина. – СПб., 2024.
6. О персональных данных: Федеральный закон от 27.07.2006 № 152-ФЗ (ред. от 29.12.2023) [Электронный ресурс]. URL: http://www.consultant.ru/document/cons_doc_LAW_61798/ (дата обращения: 27.04.2025).
7. Об утверждении Методических рекомендаций по переходу на использование российского программного обеспечения: Приказ Министерства цифрового развития, связи и массовых коммуникаций РФ от 18.01.2023 № 21. [Электронный ресурс]. URL: https://www.consultant.ru/document/cons_doc_LAW_439904/.
8. Об утверждении требований к защите информации в государственных информационных системах: Постановление Правительства РФ от 30.06.2018 № 768 (ред. от 26.10.2023) [Электронный ресурс].

URL: http://www.consultant.ru/document/cons_doc_LAW_302280/ (дата обращения: 27.04.2025).

9. Об утверждении Требований о защите информации, не составляющей государственную тайну, содержащейся в государственных информационных системах: Приказ ФСТЭК России от 11.02.2013 № 17 (ред. от 28.08.2024) [Электронный ресурс]. URL: https://www.consultant.ru/document/cons_doc_LAW_147084/ (дата обращения: 15.05.2025).

10. Об электронной подписи: Федеральный закон от 06.04.2011 № 63-ФЗ (ред. от 29.12.2023) [Электронный ресурс]. URL: http://www.consultant.ru/document/cons_doc_LAW_112701/ (дата обращения: 27.04.2025).

11. Производственный календарь [Электронный ресурс]. URL: https://www.consultant.ru/law/ref/calendar/proizvodstvennye/ (дата обращения: 15.05.2025).

12. Реестр российского ПО / Специализированная программная платформа АСТ СН [Электронный ресурс]. URL: https://reestr.digital.gov.ru/reestr/1234185/ (дата обращения: 25.05.2025).

13. Средняя заработная плата по России в 2024 году [Электронный ресурс]. URL: https://www.interfax.ru/russia/1012173 (дата обращения: 15.05.2025).

14. Суворова Г.М. Информационная безопасность : учебник для вузов – Москва : Издательство Юрайт, 2025.

15. Шадаев допустил ограничение зарубежных облачных сервисов со зрелыми аналогами в РФ [Электронный ресурс]. URL: https://www.interfax.ru/russia/1028459.

16. Шаньгин В.Ф. Информационная безопасность компьютерных систем и сетей: учеб. пособие. – М. : ИД «ФОРУМ» : ИНФРА-М, 2024.

17. Certbot [Электронный ресурс]. URL: https://certbot.eff.org/ (дата обращения: 15.05.2025).

18. Keytool [Электронный ресурс]. URL: https://docs.oracle.com/javase/8/docs/technotes/tools/unix/keytool.html (дата обращения: 19.05.2025)

19. OpenSSL [Электронный ресурс]. URL: https://www.openssl.org/ (дата обращения: 15.05.2025)

20. PyCryptodome [Электронный ресурс]. URL: https://pycryptodome.readthedocs.io/en/latest/ (дата обращения: 15.05.2025)