Оглавление

[**Введение** 2](#_Toc103205497)

[**Описание предприятия** 3](#_Toc103205498)

[**1.** **Предметная область** 8](#_Toc103205499)

[**2.** **Анализ требований к приложению с использованием CASE средств** 9](#_Toc103205500)

[**3.** **Требования к приложению** 15](#_Toc103205501)

[**4.** **Архитектура приложения** 16](#_Toc103205502)

[**5.** **Средства разработки** 17](#_Toc103205503)

[**6.** **Модель данных** 19](#_Toc103205504)

[**7.** **Прототип приложения** 21](#_Toc103205505)

[**8.** **Источники данных** 26](#_Toc103205506)

**Введение**

Преддипломная практика направлена на углубление первоначального практического опыта обучающегося, развитие общих и профессиональных компетенций, проверку его готовности к самостоятельной трудовой деятельности, а также на подготовку к выполнению выпускной квалификационной работы в организациях различных организационно-правовых форм.

Задачами производственной (преддипломной) практики являются:

* подготовка выпускника к выполнению основных профессиональных функций в соответствии с квалификационными требованиями;
* ознакомление студентов непосредственно на предприятиях, в учреждениях и организациях с передовой техникой и технологией, с организацией труда и экономикой производственной деятельности;
* изучение принципов проектирования программного обеспечения вычислительной техники и автоматизированных систем с использованием современных средств и методов автоматизации основных этапов проектирования;
* приобретение практических навыков по разработке и проектированию функциональных задач и подсистем в соответствии с темой дипломного проекта;
* сбор необходимого материала для выполнения дипломного проекта в соответствии с полученными индивидуальными заданиями;
* изучение эффективности функционирования информационных систем предприятия, анализа качества работы и исследование проблем информационных систем на предприятии;
* освоение опыта экономического анализа действующих информационных систем;
* закрепление и совершенствование знаний и практических навыков, полученных студентами в процессе обучения.

**Описание предприятия**

Высшим звеном структуры колледжа является Администрация колледжа. Возглавляет администрацию директор колледжа Кольцов Александр Германович.

В администрацию колледжа также входят:

* совет колледжа
* педагогический совет
* методический совет
* бухгалтерия
* юрисконсульт
* отдел кадров
* заместители директора.
* Далее идут отделы колледжа, такие как:
* учебный отдел
* отдел учебно-воспитательной работы
* отдел учебно-производственной работы
* учебно-вспомогательный отдел
* отдел вспомогательных и учебных служб.

В свою очередь отделы состоят из отделений, служб.

Учебный отдел включает в себя учебную часть, отделение технологии аэрокосмического производства, отделение электромеханики, радиотехники и технологии сварки, отделение информационных технологий и экономики, отделение очно-заочного (вечернего) обучения, отделение инновационных образовательных программ, содействия трудоустройству и профессиональной ориентации.

Отдел учебно-воспитательной работы состоит из социально-психологической службы, службы организации внеучебной и досуговой деятельности обучающегося.

Лаборатории, мастерские и служба технического обеспечения образуют отдел учебно-производственной работы.

В учебно-вспомогательный отдел включены научно-методический центр, библиотека, служба информатизации и приёмная комиссия.

Служба информатизации состоит из заведующего лабораторией, инженеров и лаборантов.

Служба информатизации поддерживает функциональность персональных компьютеров и информационного оборудования:

* персональные компьютеры: 347 из которых 230 компьютеров используются в учебных целях

Для обучения студентов есть отдельные кабинеты для каждой специальности, 3 кабинета используются для обучения студентов на технические специальности, 1 кабинет используется для преподавания экономических дисциплин, 8 кабинетов используются для ИТ специальностей.

Для ИТ специальностей кабинет 316 оснащён персональными компьютерами со следующими характеристиками:

* процессор: Intel Core i5 6400
* оперативная память: DDR4 - 16гб
* материнская плата: h110m-s2v-cf
* жёсткий диск: 500гб
* видеокарта: Nvidea Geforce gt710
* монитор: Dell e231 16h

установленное ПО кабинета 312:

* ОС WINDOWS 10 Enterprises
* антивирус Kaspersky Endpoint Security
* SQL Server 2016
* Visual studio 2017
* Cisco Packet Tracer 7.2
* Putty
* Oracle VM VirtualBox 6.0.0
* Microsoft Office 2016
* Microsoft Visio 2016
* SQL Server management studio 17.8.1
* Oracle VM VirtualBox 6.0.0
* Visual Paradigma 15.0

Сотрудники отдела бухгалтерии работают в системе 1С-Бухгалтерия. Для снижения нагрузки на персональные компьютеры в колледже имеется сервер 1С, к которому подключаются с помощью протокола «Удалённого рабочего стола» (RDP) на базе ОС Windows Server 2012.

Характеристики данного сервера:

* процессор: Xeon E5630
* оперативная память: 16 Гб
* жёсткий диск: 150 ГБ

Последний отдел, отдел вспомогательных и обеспечивающих служб, состоит из канцелярии, службы охраны, службы снабжения, службы хозяйственного обеспечения и здравпункта.

Одним из важных звеньев структуры колледжа является ИВЦ.

На сегодняшний день основной задачей ИВЦ колледжа является поддержание и усовершенствование корпоративной управленческой системы, куда входят:

* создание и обслуживание базы данных о студентах
* база данных о преподавателях
* база данных научной библиотеки
* база данных расписания занятий, практик и многое другое
* поддержка сайта колледжа
* создание и поддержка информационной базы по специальностям

Благодаря вычислительному центру абитуриенты могут узнать интересующую их информацию о колледже, кафедрах, специальностях.

На базе вычислительного центра существует настоящая летопись колледжа в виде фото и видеоматериалов.

Должностные обязанности лаборанта:

* выполняет лабораторные испытания, измерения и другие виды работ при проведении исследований и разработок.
* принимает участие в сборе и обработке материалов в процессе исследований в соответствии с утвержденной программой работы.
* следит за исправным состоянием лабораторного оборудования, осуществляет его наладку.
* подготавливает оборудование (приборы, аппаратуру) к проведению экспериментов, осуществляет его проверку и простую регулировку согласно разработанным инструкциям и другой технической документации.
* участвует в выполнении экспериментов, осуществляет необходимые подготовительные и вспомогательные операции, проводит наблюдения, ведет рабочие журналы.
* обеспечивает сотрудников подразделения необходимыми для работы оборудованием, материалами и др.
* обрабатывает, систематизирует и оформляет в соответствии с методическими документами результаты испытаний, измерений, ведет их учет.
* производит выборку данных из литературных источников, реферативных и информационных изданий, нормативно - технической документации в соответствии с установленным заданием.
* принимает участие в составлении и оформлении технической документации по выполненным работам.
* обеспечивает соблюдение правил эксплуатации оборудования и требований по охране труда, технике безопасности, производственной санитарии, гигиене труда, гражданской обороны и противопожарной безопасности в соответствии с действующими правилами и инструкциями.

Должностные обязанности инженера:

* выполняет работы в области научно-технической деятельности, информационного обслуживания, организации труда и управления, метрологического обеспечения, технического контроля.
* разрабатывает методические и нормативные документы, техническую документацию, а также предложения и мероприятия по осуществлению разработанных проектов и программ.
* проводит технико-экономический анализ, обосновывает принимаемые и реализуемые решения, изыскивает возможности сокращения цикла выполнения работ, содействует подготовке процесса их выполнения, обеспечению отдела информатизации необходимыми техническими данными, документами, материалами, оборудованием.
* участвует в работах по исследованию, разработке проектов и программ отдела информатизации в проведении мероприятий, связанных с испытанием оборудования и внедрением его в эксплуатацию, а также в выполнении работ по стандартизации технических средств, процессов, оборудования и материалов, в рассмотрении технической документации.
* подготавливает обзоры, отзывы, заключения по вопросам выполняемой работы.
* изучает и анализирует информацию, технические данные, показатели и результаты работы, обобщает и систематизирует их и проводит расчёты.
* составляет графики работ, заказы, заявки, инструкции, пояснительные записки, карты, схемы и другую техническую документацию, а также установленную отчётность по утверждённым формам и в установленные сроки.
* оказывает методическую и практическую помощь при реализации проектов и программ, планов и договоров.
* осуществляет экспертизу технической документации, надзор и контроль состояния и эксплуатации оборудования.
* следит за соблюдением установленных требований, действующих норм, правил и стандартов.
* организует работу по повышению научно-технических знаний работников отдела информатизации.

1. **Предметная область**

Управление договорными отношениями, подразумевает собой приложение для взаимодействия менеджера с договорами. В данном приложении должна быть возможность создать договор на оказание какой-либо услуги, которую заказчик заказал у компании.

Менеджер имеет свой личный профиль, в котором указана основная информация о менеджере: имя, фамилия, номер телефона, электронная почта, город, дата рождения. Менеджер может просматривать оформленные им договора и информацию о них, например, ФИО клиента, с которым оформлен договор, номер телефона клиента, номер договора, содержание договора и т.д.

Менеджер может создать договор на оказание услуги, об оказании которой он договорился с клиентом. Для создания договора нужно выбрать тип договора в выпадающем списке. После ее выбора мы сможем увидеть первоначальный шаблон договора, примерные временные рамки на выполнение, назначить полную стоимость по договору, указать сумму аванса, разбить платежи на несколько частей и указать сроки оплаты для клиента. Менеджер обсуждает подробные условия с клиентом и вносит корректировки в договор, позже менеджер отправляет предварительный договор юристу компании, который просматривает его и после утверждает его или вносит контрольные корректировки.

Если клиента не устроят условия договора, он сможет связаться с менеджером для обсуждения внесения корректировок. После итогового утверждения менеджер и передает всю необходимую информацию исполнителям. Когда наступает платежный период по договору, клиенту приходит письмо на почту о необходимости погасить задолженность.

1. **Анализ требований к приложению с использованием CASE средств**

На рисунке 1 представлена диаграмма вариантов использования приложения, отражающая отношения между актерами (пользователями) и прецедентами системы (варианты использования разрабатываемой системы). Она поможет понять, как необходимо построить логику приложения.

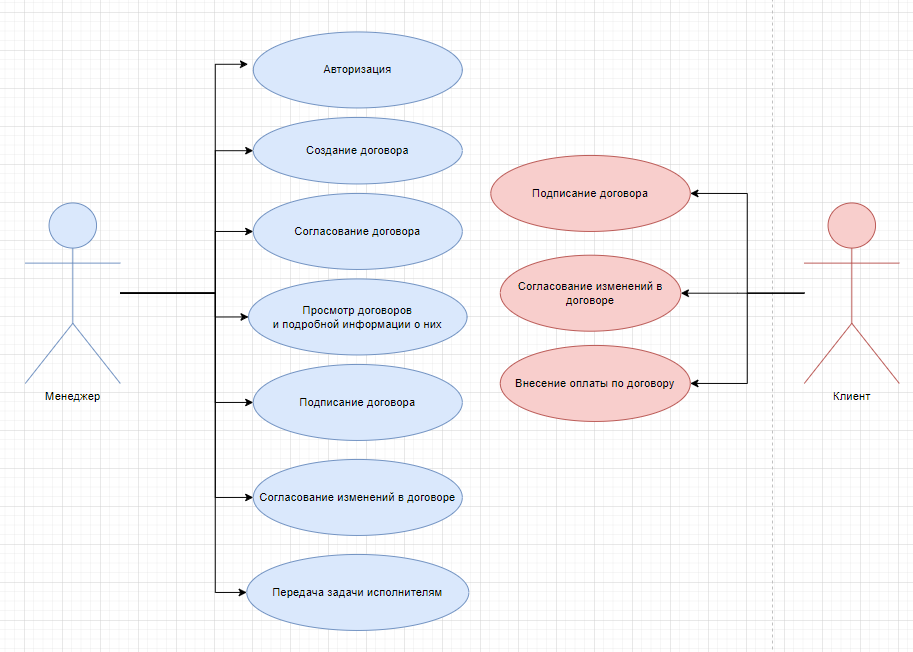


Рисунок 1. Диаграмма вариантов использования

На данной диаграмме представлены следующие категории пользователей:

-Клиент – это человек, который решил подписать договор с компанией на оказание услуг. Такой пользователь имеет возможность подписания договора, а также возможное внесение корректировок в договор и внесение оплаты по договору.

-Менеджер – сотрудник компании, который имеет права на просмотр своих договоров, а также, их подписание, согласование и внесение изменений. Менеджер может просматривать всю информацию о договоре.

Выполним описание основных вариантов использования приложения. Например, опишем функцию «Подписание договора». Результат описания представлен в таблице 1.

Таблица 1 Вариант использования "Создание договора"

|  |  |
| --- | --- |
| *Название прецедента* | Создание договора |
| *Цель* | Создать договор с клиентом компании |
| *Актеры* | Менеджер |
| *Тип* | Основной |
| *Стиль* | Базовый |
| *Ссылки (предусловия)* | Реализован прецеденты авторизация |
| *Результаты (постусловия)* | В случае успешного основного сценария передача данных о договоре в базу данных и появление договора в приложении. |

Основной успешный сценарий для данного варианта использования представлен в таблице 2.

Таблица 2 Основной успешный сценарий "Создание договора"

| *Действие актера* | *Отклик системы* |
| --- | --- |
| 1.Менеджер переходит на вкладку «Создание договора».  3. Менеджер выбирает тип договора, временные сроки, заполнить сам договор и выставить цену для клиента. | 2.Приложение переводит пользователя на страницу создания договора.  4.Сервер создает запись нового договора в базе данных и отображает информацию о ней в приложении. |

Далее опишем функцию «Согласование договора»

Таблица 3. Вариант использования «Согласования договора»

|  |  |
| --- | --- |
| *Название прецедента* | Согласование договора |
| *Цель* | Согласовать договор с клиентом компании |
| *Актеры* | Менеджер, клиент |
| *Тип* | Основной |
| *Стиль* | Базовый |
| *Ссылки (предусловия)* | Реализован прецеденты создание договора |
| *Результаты (постусловия)* | В случае успешного основного сценария статус договора измениться на согласован |

Основной успешный сценарий для данного варианта использования представлен в таблице 4.

Таблица 4. Основной успешный сценарий "Согласование договора"

| *Действие актера* | *Отклик системы* |
| --- | --- |
| 1.Менеджер отправляет договор на согласование юристом.  3. Менеджер выбирает нужный договор и изменяет его статус на согласованный. | 2.Юрист согласовывает или вносит корректировки в договор и подписывает договор.  4.Сервер изменяет статус договора на согласованный. |

Далее опишем функцию «Просмотр договоров и подробной информации о них».

Таблица 5. Вариант использования «Просмотр договоров и информации о них»

|  |  |
| --- | --- |
| *Название прецедента* | Просмотр договоров и подробной информации о них |
| *Цель* | Просмотреть информацию о договоре |
| *Актеры* | Менеджер |
| *Тип* | Основной |
| *Стиль* | Базовый |
| *Ссылки (предусловия)* | Реализован прецеденты авторизация |
| *Результаты (постусловия)* | В случае успешного основного сценария в приложении отобразиться вся информация о просматриваемом договоре. |

Основной успешный сценарий для данного варианта использования представлен в таблице 6.

Таблица 6. Основной успешный сценарий "Просмотр договоров и информации о них"

| *Действие актера* | *Отклик системы* |
| --- | --- |
| 1.Менеджер переходит на страницу «Договоры».  3. Менеджер выбирает нужный договор и нажимает на кнопку «подробнее». | 2. Приложение отображает все договоры менеджера.  4. Приложение отображает информацию о договоре. |

Далее опишем функцию «Подписание договора».

Таблица 7. Вариант использования «Подписание договора»

|  |  |
| --- | --- |
| *Название прецедента* | Подписание договора |
| *Цель* | Подписать договор с клиентом компании |
| *Актеры* | Менеджер, клиент |
| *Тип* | Основной |
| *Стиль* | Базовый |
| *Ссылки (предусловия)* | Реализован прецеденты согласование договора |
| *Результаты (постусловия)* | В случае успешного основного сценария статус договора измениться на подписан. |

Основной успешный сценарий для данного варианта использования представлен в таблице 8.

Таблица 8. Основной успешный сценарий «Подписание договора»

| *Действие актера* | *Отклик системы* |
| --- | --- |
| 1.Менеджер подписывает письменный договор с клиентом.  2. Менеджер изменяет статус договора в приложении на «подписан». | 4. Сервер изменяет статус договора на подписан. |

1. **Требования к приложению**

Минимальные системные требования:

* процессор – Intel Core 2 Duo2000 MHz;
* память - 2048 Mb;
* разрешение экрана 800 x 600 или больше;
* операционная система: MSWindows 7 и выше;
* MS .Net Framework

1. **Архитектура приложения**

Приложение будет построено на клиент-сервер архитектуре. Под клиент-серверным приложением мы будем понимать информационную систему, основанную на использовании серверов баз данных.

На стороне клиента выполняется код приложения, в который обязательно входят компоненты, поддерживающие интерфейс с конечным пользователем, производящие отчеты, выполняющие другие специфичные для приложения функции

Клиентская часть приложения взаимодействует с клиентской частью программного обеспечения управления базами данных, которая, фактически, является индивидуальным представителем СУБД для приложения.

К преимуществам архитектуры «клиент-сервер» можно отнести:

* централизованность, поскольку все данные и управление сосредоточены в центральном сервере;
* информационная безопасность, поскольку ресурсы общего пользования администрируются централизованно;
* производительность, использование выделенного сервера повышает скорость работы ресурсов общего пользования;
* масштабируемость, количество клиентов и серверов можно увеличивать независимо друг от друга;

1. **Средства разработки**

Для разработки приложения можно использовать язык программирования C# и платформу пользовательского интерфейса для создания клиентских приложений для настольных систем WPF в среде разработки Visual Studio.

Рассмотрим преимущества и недостатки использования данных средств разработки:

Преимущества C#:

* с# – это объектно-ориентированный, простой и в то же время мощный язык программирования, который позволяет разработчикам создавать многофункциональные приложения;
* с# относиться к языкам компилируемого типа, поэтому он обладает всеми преимуществами таких языков;
* мощная библиотека классов. .NET представляет единую для всех поддерживаемых языков библиотеку классов. Какое бы приложение мы ни собирались писать на C# — текстовый редактор, чат или сложный веб-сайт — так или иначе мы задействуем библиотеку классов .NET;
* с# объединяет лучшие идеи современных языков программирования Java, C++, Visual Basic и т.д;
* из-за большого разнообразия синтаксических конструкций и возможности работать с платформой .Net, C# позволяет быстрее, чем любой другой язык, разрабатывать программные решения;
* разнообразие технологий. Общеязыковая среда исполнения CLR и базовая библиотека классов — это основа для целого стека технологий, которые разработчики могут задействовать при создании разных приложений. Например, для баз данных в этом стеке имеется технология ADO.NET и Entity Framework Core. Для графических приложений с насыщенным интерфейсом — технологии WPF и UWP. Для более простых графических приложений — Windows Forms. Для разработки мобильных приложений — Xamarin. Для создания веб-сайтов - ASP.NET и т.д;

Но есть у C# и некоторые недостатки:

* приоритетная ориентированность на платформу Windows;

Преимущества использования WPF:

* **независимость графики от разрешения и устройства.** Основной единицей измерения в графической системе WPF является аппаратно-независимый пиксель, размер которого составляет 1/96 дюйма. Это создает основу для независимой от разрешения и аппаратной платформы отрисовки. Каждый аппаратно-независимый пиксель автоматически масштабируется в соответствии с заданным в системе количеством точек на дюйм (DPI);
* **повышение точности.** Система координат WPF основана на числах двойной точности с плавающей запятой, а не числах одинарной точности. Значения преобразования и прозрачности также выражаются числами двойной точности. Платформа WPF также поддерживает широкую цветовую палитру (scRGB) и имеет встроенную поддержку управления входными данными из разных цветовых схем;
* **расширенная поддержка графики и анимации.** Платформа WPF упрощает программирование графики, автоматически управляя анимированными сценами. Вам не нужно беспокоиться об обработке сцен, циклах отрисовки и билинейной интерполяции. Кроме того, WPF обеспечивает поддержку проверки попадания и полную поддержку альфа-версии компоновки;
* **аппаратное ускорение.** Система графики WPF использует возможности графического оборудования, чтобы снизить нагрузку на ЦП;

Недостатки использования WPF:

* плохая документированность;
* чувствительность к драйверам видеокарты;

Исходя из преимуществ и недостатков было принято решение использовать данные средства разработки для реализации требований приложения. С помощью них мы можем полностью реализовать функционал и визуальную составляющую приложения, на языке C# и XAML, с которыми, как раз, мы уже знакомы, так как изучали его в нашем колледже.

1. **Модель данных**

На этапе построения концептуальной модели и базы данных было выявлено 4 таблицы: Клиент, Договор, Типы договоров, Менеджер.

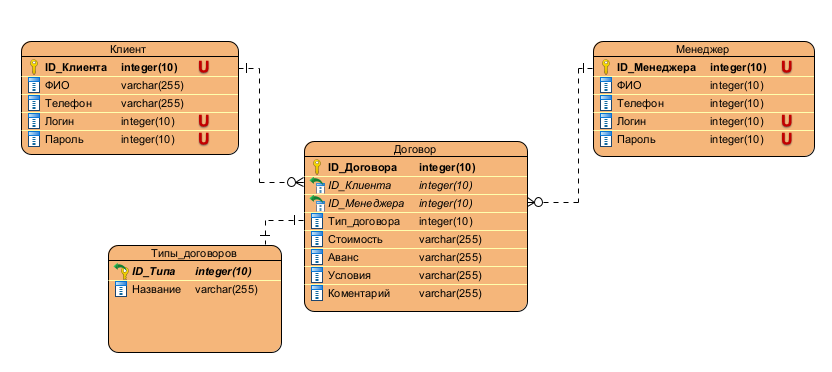


Рисунок 2 - ER-диаграмма

На основе ЕР диаграммы будут построены таблицы: Клиент, Договор, Типы\_договоров, Менеджер.

Таблица «Клиент» - в данной таблице хранятся данные о клиенте. Здесь имеются поля: ID\_Клиента, ФИО, Телефон, Логин, Пароль. Имеет связи с сущностями «Договор».

Таблица 9. Описание полей таблицы «Клиент».

|  |  |
| --- | --- |
| ID\_Клиента (PK) | Integer |
| ФИО | nvarchar |
| Телефон | nvarchar |
| Логин | nvarchar |
| Пароль | nvarchar |

Таблица «Менеджер» - в данной таблице хранятся данные о менеджере. Здесь имеются поля: ID\_Менеджера, ФИО, Телефон, Логин, пароль.

Таблица 10. Описание полей таблицы "Менеджер"

|  |  |
| --- | --- |
| ID\_Менеджера | Integer |
| ФИО | nvarchar |
| Телефон | nvarchar |
| Логин | nvarchar |
| Пароль | nvarchar |

Таблица «Договор» - в данной таблице храниться информация о договорах. Здесь имеются поля: ID\_Договора, ID\_Клиента, ID\_Менеджера, Тип\_Договора, Стоимость, Аванс, Условия, комментарий. Имеет связи с сущностями «Клиент», «Типы\_договоров», «Менеджер».

Таблица 11.Описание полей «Менеджер».

|  |  |
| --- | --- |
| ID\_Договора (РК) | Integer |
| ID\_Клиента (FK) | Integer |
| ID\_Менеджера (FK) | Integer |
| Тип\_Договора | Integer |
| Стоимость | money |
| Аванс | money |
| Условия | nvarchar |
| комментарий | nvarchar |

Таблица «Тип\_договоров» - здесь имеются поля: ID\_Типа, Название.

Таблица 12. Описание таблицы «Тип\_договоров»

|  |  |
| --- | --- |
| ID\_Типа (PK) | Integer |
| Название | nvarchar |

1. **Прототип приложения**

Открыв приложение, пользователь будет видеть страницу авторизации, на которой он может вести логин и пароль, после авторизации пользователь попадает на главную страницу, страницу «Договора» на которой отображаются договора менеджера и основная информация о них, с этой страницы менеджер может перейти на страницу выбранного договора, в ней будет отображаться подробная информация о договоре, так же можно будет скачать сам договор в формате word. Так же с главной страницы пользователь может перейти на страницу «Профиль» и «Создание договора», на странице профиль отображается основная информация о менеджере, а на странице создания договоров пользователь может заполнить необходимые поля и создать договор.

С главной страницы можно будет перейти в следующие разделы:

* Страница авторизации;
* Профиль;
* Договора;
* Подробная информация о договоре;
* Создание договора;

Более подробное описание всех переходов по приложению отображено на концептуальной модели приложения, представленной на рисунке 3.

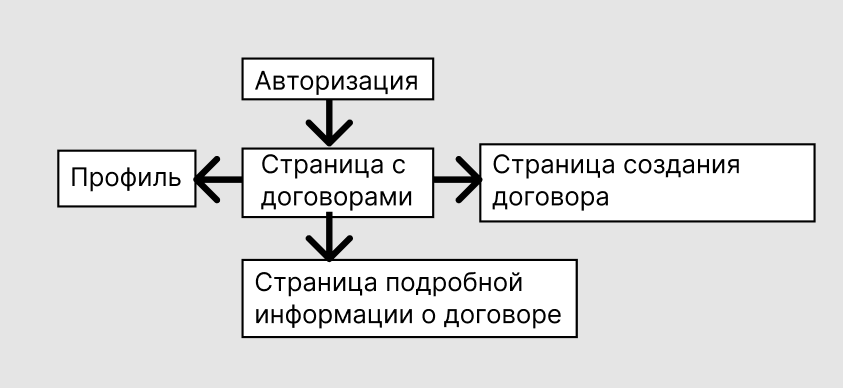


Рисунок 3 - Карта приложения

Макеты данных страниц представлены на рисунках 4 - 8.



Рисунок 4 – Окно авторизации

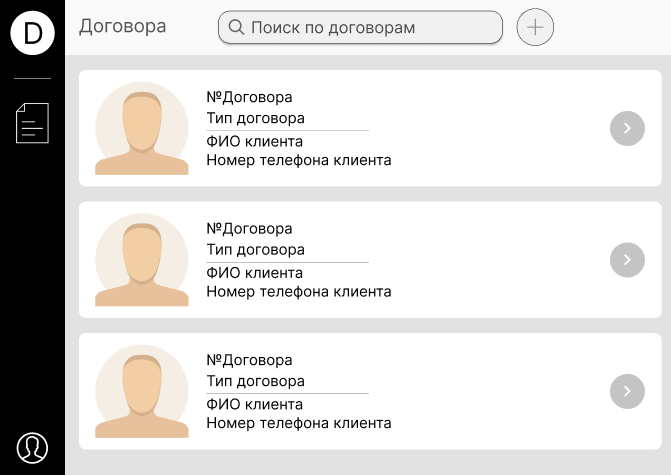


Рисунок 5 – Окно договоров

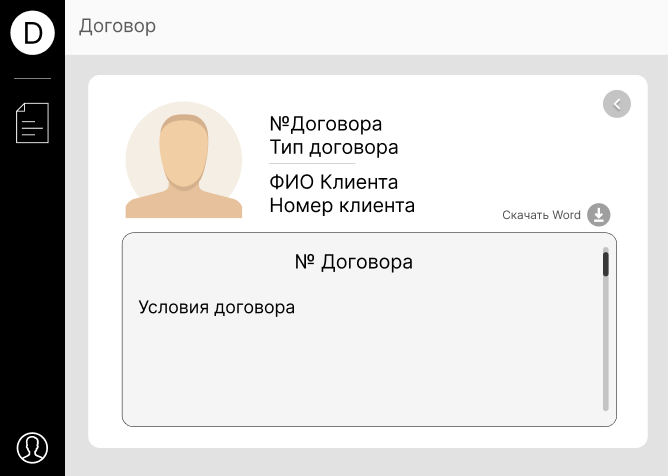


Рисунок 6 – Окно подробной информации о договоре

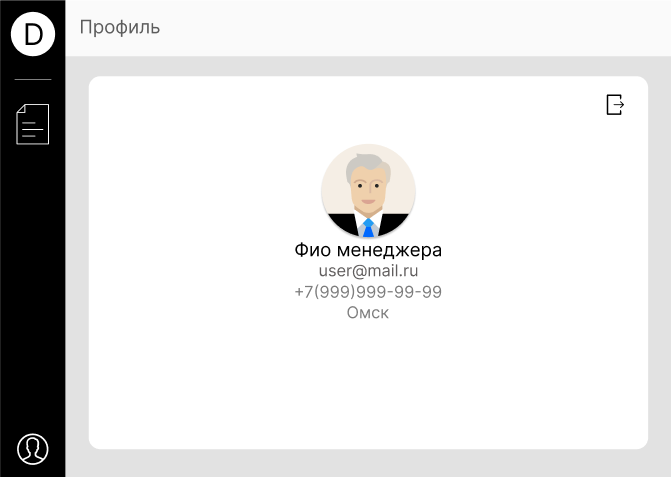


Рисунок 7 – Окно профиля

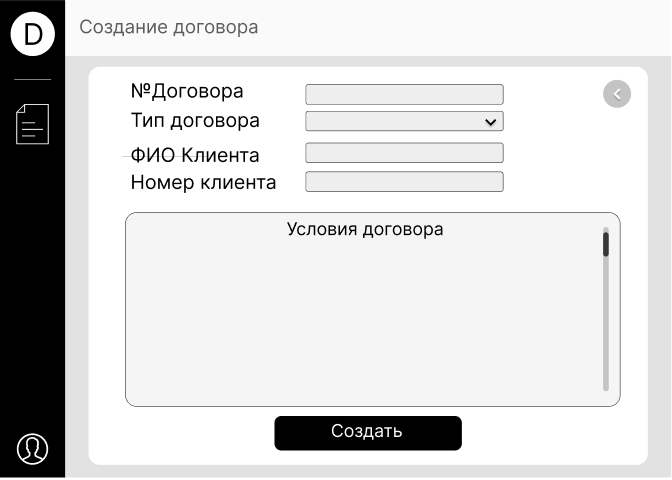


Рисунок 8 – Окно создания договора

Прототип приложения:

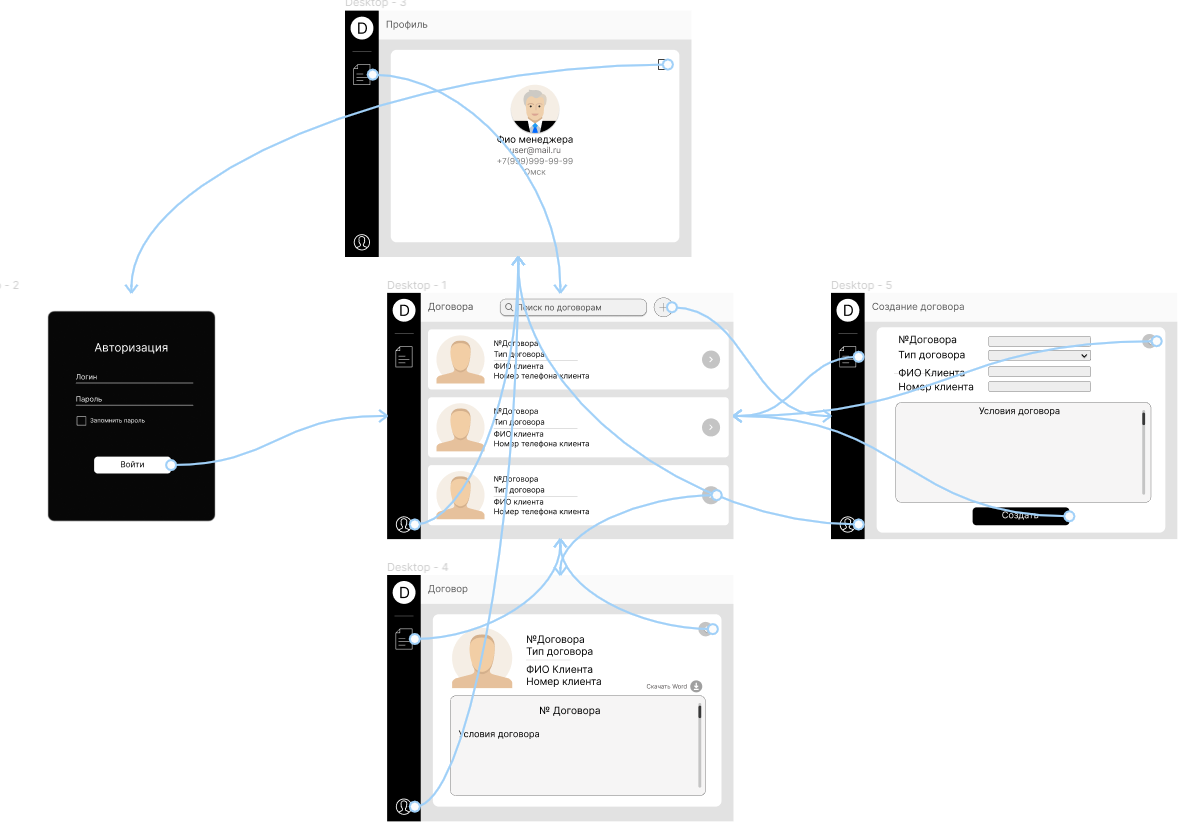


Рисунок 8. Прототип приложения

1. **Источники данных**
2. ГОСТ 28195-89 Оценка программных средств. Общие положения. – 01.09.2013
3. ГОСТ Р 59194-2020 Управление требованиями. Основные положения – 01.01.2021
4. ГОСТ Р ИСО ТО 12182-2002 Класификация программных средств – 01.07.2003
5. ГОСТ Р 56920-2016 Системная и программная инженерия. Тестирование ПО – 01.06.2017
6. ГОСТ Р 57100-2016 Системная и программная инженерия. Описание архитектуры – 01.09.2017
7. Фленов Михаил Библия C#; БХВ-Петербург - М., 2011.Гарнаев А. Самоучитель Visual Studio .NET 2003; БХВ-Петербург - М., 2003
8. Лотка Р. C# и CSLA .NET Framework. Разработка бизнес-объектов; Диалектика / Вильямс - М. Алекс Макки Введение в .NET 4.0 и Visual Studio 2010 для профессионалов; Диалектика / Вильямс - М., 2010.
9. Подбельский, В. В. Язык С#. Базовый курс / В.В. Подбельский. - М.: Финансы и статистика, Инфра-М, 2011. - 384 c.
10. Рихтер, Джеффри CLR via C#. Программирование на платформе Microsoft .NET Framework 4.0 на языке C# / Джеффри Рихтер. - М.: Питер, 2013. - 928 c.
11. Албахари, Джозеф C# 3.0. Справочник / Джозеф Албахари , Бен Албахари. - М.: БХВ-Петербург, 2012. - 944 c.
12. Альфред, В. Ахо Компиляторы. Принципы, технологии и инструментарий / Альфред В. Ахо и др. - М.: Вильямс, 2015. - **266** c.
13. https://metanit.com/sharp/entityframeworkcore/
14. <https://habr.com/ru/c#/>
15. <https://ru.wikipedia.org/wiki/C_Sharp>
16. <https://www.microsoft.com/ru-ru/windows-server>