Министерство образования Московской области

Государственное образовательное учреждение высшего образования Московской области

«Государственный гуманитарно-технологический университет»

**Ликино-Дулевский политехнический колледж – филиал ГГТУ**

**КУРСОВОЙ ПРОЕКТ**

Разработка приложения по систематизации работы менеджеров в организации.

МДК 02.01 «Технология разработки ПО»

**Выполнил:**

Кузнецов Александр Алексеевич\_\_\_\_

студент группы ИСП.18.1А\_\_\_\_\_\_\_\_\_\_\_\_\_

09.02.07 Информационные системы и программирование)\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

очной формы обучения\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Руководитель:**

Селиверстова Ольга Михайловна\_\_\_\_\_\_\_\_\_\_

Оценка

\_\_\_\_\_\_\_\_\_/\_\_\_­­­­­­\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/

Подпись руководителя

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Ликино-Дулево

2021 год

**Оглавление**

[**Введение** 3](#_Toc91543820)

[1. **Разработка системного проекта** 5](#_Toc91543821)

[1.1 Назначение разработки 5](#_Toc91543822)

[1.2 Требования к функциональным характеристикам 5](#_Toc91543823)

[1.3 Требования к надёжности и безопасности 5](#_Toc91543824)

[1.4 Требования к составу и параметрам технического средства 6](#_Toc91543825)

[**2. Разработка технического проекта 7**](#_Toc91543826)

[2.1 Обоснование выбора CASE-средств 7](#_Toc91543827)

[2.1.1 Диаграмма прецедентов (бизнес-процессов) 7](#_Toc91543828)

[2.1.2 Определение бизнес-процессов 8](#_Toc91543829)

[2.1.3 Словесный алгоритм бизнес-процесса 3-Прд\_фр\_зк 8](#_Toc91543830)

[2.1.4 Диаграмма действий бизнес-процесса 9](#_Toc91543831)

[2.1.5 Таблица операций 9](#_Toc91543832)

[2.1.6 Таблица описаний документов 10](#_Toc91543833)

[**3. Реализация 11**](#_Toc91543834)

[3.1 Обоснование выбора средств разработки 11](#_Toc91543835)

[3.2 Руководство программиста 12](#_Toc91543836)

[3.3 Руководство пользователя 16](#_Toc91543837)

[**4. Тестирование и отладка 19**](#_Toc91543838)

[4.1 Таблица класса эквивалентности 20](#_Toc91543839)

[**5. Разработка механизмов защиты данных 21**](#_Toc91543840)

[Заключение 23](#_Toc91543841)

[Список литературы 24](#_Toc91543842)

# Введение

Система управления взаимоотношениями с клиентами — [прикладное программное обеспечение](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%B8%D0%BA%D0%BB%D0%B0%D0%B4%D0%BD%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%BD%D0%BE%D0%B5_%D0%BE%D0%B1%D0%B5%D1%81%D0%BF%D0%B5%D1%87%D0%B5%D0%BD%D0%B8%D0%B5) для организаций, предназначенное для [автоматизации](https://ru.wikipedia.org/wiki/%D0%90%D0%B2%D1%82%D0%BE%D0%BC%D0%B0%D1%82%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D1%8F) стратегий взаимодействия с [заказчиками](https://ru.wikipedia.org/wiki/%D0%97%D0%B0%D0%BA%D0%B0%D0%B7%D1%87%D0%B8%D0%BA) ([клиентами](https://ru.wikipedia.org/wiki/%D0%97%D0%B0%D0%BA%D0%B0%D0%B7%D1%87%D0%B8%D0%BA)), в частности для повышения уровня продаж, оптимизации [маркетинга](https://ru.wikipedia.org/wiki/%D0%9C%D0%B0%D1%80%D0%BA%D0%B5%D1%82%D0%B8%D0%BD%D0%B3) и улучшения обслуживания клиентов путём сохранения информации о клиентах и истории взаимоотношений с ними, установления и улучшения [бизнес-процессов](https://ru.wikipedia.org/wiki/%D0%91%D0%B8%D0%B7%D0%BD%D0%B5%D1%81-%D0%BF%D1%80%D0%BE%D1%86%D0%B5%D1%81%D1%81) и последующего анализа результатов.

CRM — модель взаимодействия, основанная на теории, что центром всей философии бизнеса является клиент, а главными направлениями деятельности компании являются меры по обеспечению эффективного маркетинга, [продаж](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D1%8F%D0%BC%D1%8B%D0%B5_%D0%BF%D1%80%D0%BE%D0%B4%D0%B0%D0%B6%D0%B8) и обслуживания клиентов. Поддержка этих бизнес-целей включает сбор, хранение и анализ информации о потребителях, поставщиках, партнёрах, а также о внутренних процессах компании. Функции для поддержки этих бизнес-целей включают продажи, маркетинг, [поддержку](https://ru.wikipedia.org/wiki/%D0%A2%D0%B5%D1%85%D0%BD%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B0%D1%8F_%D0%BF%D0%BE%D0%B4%D0%B4%D0%B5%D1%80%D0%B6%D0%BA%D0%B0) потребителей.

CRM-система может включать:

* фронтальную часть, обеспечивающую обслуживание клиентов на точках продаж с автономной, распределенной или централизованной обработкой информации;
* операционную часть, обеспечивающую авторизацию операций и оперативную отчётность;
* [хранилище данных](https://ru.wikipedia.org/wiki/%D0%A5%D1%80%D0%B0%D0%BD%D0%B8%D0%BB%D0%B8%D1%89%D0%B5_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85);
* аналитическую подсистему;
* распределенную систему поддержки продаж: [реплики](https://ru.wikipedia.org/wiki/%D0%A0%D0%B5%D0%BF%D0%BB%D0%B8%D0%BA%D0%B0%D1%86%D0%B8%D1%8F_(%D0%B2%D1%8B%D1%87%D0%B8%D1%81%D0%BB%D0%B8%D1%82%D0%B5%D0%BB%D1%8C%D0%BD%D0%B0%D1%8F_%D1%82%D0%B5%D1%85%D0%BD%D0%B8%D0%BA%D0%B0)) данных на точках продаж или [смарт-карты](https://ru.wikipedia.org/wiki/%D0%A1%D0%BC%D0%B0%D1%80%D1%82-%D0%BA%D0%B0%D1%80%D1%82%D0%B0).

Процесс создания:

* создание информационного наполнения;
* создание системы пользователей, их регистрации и авторизации;
* система постов и комментариев к ним;
* система репутации;
* система личных сообщений.

Изучить предметную область Целью курсовой работы является изучение разработка CRM приложения в соответствии с этапами ЖЦ ПП.

При создании проекта предположительно будет использоваться следующее ПО:

* фронт-энд технологии: HTML, CSS, JavaScript – стандарты в сфере фронт-энд разработки веб-сайтов;
* MongoDB – это ориентированная на документы база данных NoSQL с открытым исходным кодом, которая использует для хранения структуру JSON. Модель данных MongoDB позволяет представлять иерархические отношения, проще хранить массивы и другие более сложные структуры..
* Javascript – язык программирования, который используют для написания frontend - и backend-частей сайтов, а также мобильных приложений. Часто в текстах и обучающих материалах название языка сокращают до JS. Это язык программирования высокого уровня, то есть код на нем понятный и хорошо читается.
* Visual Studio Code (VS Code) — редактор исходного кода, разработанный Microsoft для Windows, Linux и macOS;
* GitHub Desktop для контроля версий. GitHub — крупнейший веб-сервис для хостинга IT-проектов и их совместной разработки;
* любой веб-браузер для просмотра и отладки сайта

# Разработка системного проекта

# Назначение разработки

Менеджер должен заносить сотрудников и клиентов в crm приложение, собирая следующие данные: ФИО, дата рождения, контактные данные, если это сотрудник, то еще менеджер вносит в обязательном порядке email и придумывает пароль, который потом сотрудник может поменять, после чего если новый пользователь является сотрудником, ему выдаются пароль и логин для входа в crm приложение. Данные вносятся в разные таблицы клиенты и сотрудники. После заключения договора с клиентом менеджер в crm приложение создает проект в проекте создает список этапов и на каждый этап назначает сотрудника, а сотрудник в свою очередь может создать в своем этапе неограниченное число задач.

# Требования к функциональным характеристикам

* Регистрация сотрудников в CRM приложении внося следующие данные: ФИО, дата рождения, контактные данные и email
* Внесение в базу клиентов внося следующие данные о клиенте: ФИО, дата рождения, контактные данные.
* Иметь возможность добавлять проекты и список задач в соответствующий проект.
* Вывод данных из БД.
* Осуществлять поиск данных: пользователя, и проектов
* Иметь функцию авторизации

# Требования к надёжности и безопасности

Приложение должно иметь функцию хеширования пароля при регистрации нового сотрудника. В приложении должно использоваться разграничение прав доступа, во избежание редактирования и просмотра данных, лицами, не уполномоченными на это. Хостинг в обязательном порядке должен иметь SSL-сертификат.

# Требования к составу и параметрам технического средства

Минимальные технические требования к ЭВМ, необходимой для нормального функционирования программного обеспечения:

* Процессор Многопоточный процессор с тактовой частотой не менее 2.5гГц
* Минимум ОЗУ 4Гб DDR4
* Накопители 1гб

Требования к информационной и технической совместимости:

* Операционная система: Windows XP или выше.
* ПО Microsoft Office Word 2003 или выше.

Рекомендуемые требования:

* Процессор Многопоточный процессор с тактовой частотой не менее 3.1гГц
* Минимум ОЗУ 6Гб DDR4
* Накопители 1гб

Требования к информационной и технической совместимости:

* Операционная система: Windows XP или выше.
* ПО Microsoft Office Word 2003 или выше.

# Разработка технического проекта

# Обоснование выбора CASE-средств

Анализ и проектирование информационной системы являются наиболее трудоемкими этапами разработки, в процессе которых CASE-средства помогают принимать наиболее грамотные технические решения и подготавливать проектную документацию. В процессе разработки большую роль играет визуализация представления информации. В этих системах построение диаграмм происходит в масштабе реального времени, происходит сквозная проверка синтаксических правил.

# 2.1.1 Диаграмма прецедентов (бизнес-процессов)

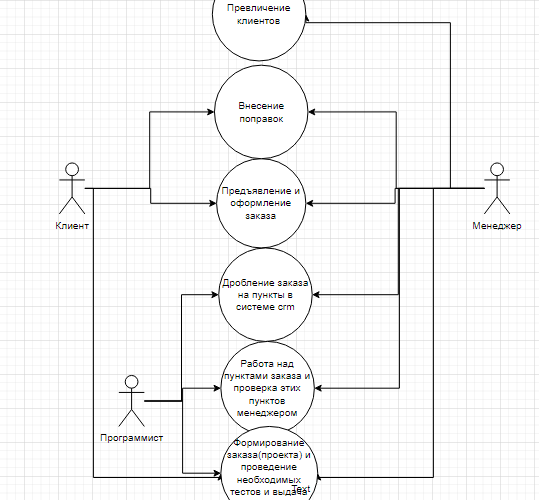


Рис.1 «Прецедентов»

# 2.1.2 Определение бизнес-процессов

Таблица №1. “Бизнес процессы”

|  |  |
| --- | --- |
| **Номер бизнес-процесса** | **Название бизнес-процесса** |
| 1-Првлч\_кл | Привлечение клиентов |
| 2-Вн\_Пр | Внесение правок |
| 3-Прд\_фр\_зк | Предъявление и оформление заказа |
| 4-Др\_зк\_фрм\_зк | Дробление заказа на пункты |
| 5-Рбт\_пн\_зк\_прв\_пнк | Работа над пунктами заказа и проверка этих пунктов менеджером |
| 6-Фрмр\_зк\_пр\_вд | Формирование заказа и проведение необходимых тестов, и выдача |

# 

# 2.1.3 Словесный алгоритм бизнес-процесса 3-Прд\_фр\_зк

Клиенты, заинтересованные в создании личного или для своей фирмы сайта, могут связаться с менеджером или приехать в офис, чтобы обсудить все детали, дизайн, функционал, желаемый трафик, способы оплаты и цену в обсуждении кроме клиента и менеджера могут участвовать программисты и дизайнеры для более детального понимания всей картины.

Если клиенту нравится предложенные дизайн, и прочее включая способ оплаты, поэтапно или за весь проект, оформляется договор на создание сайта, для заключения договора требуется комплект документов, предоставляемых клиентом. Оплата идет поэтапно на реквизиты компании.

При выполнение всех этапов готовый проект предоставляется клиенту, если все устраивает, проект передается клиенту или клиенту помогают перенести сайт на хостинг, иначе вносятся необходимые правки.

# 2.1.4 Диаграмма действий бизнес-процесса

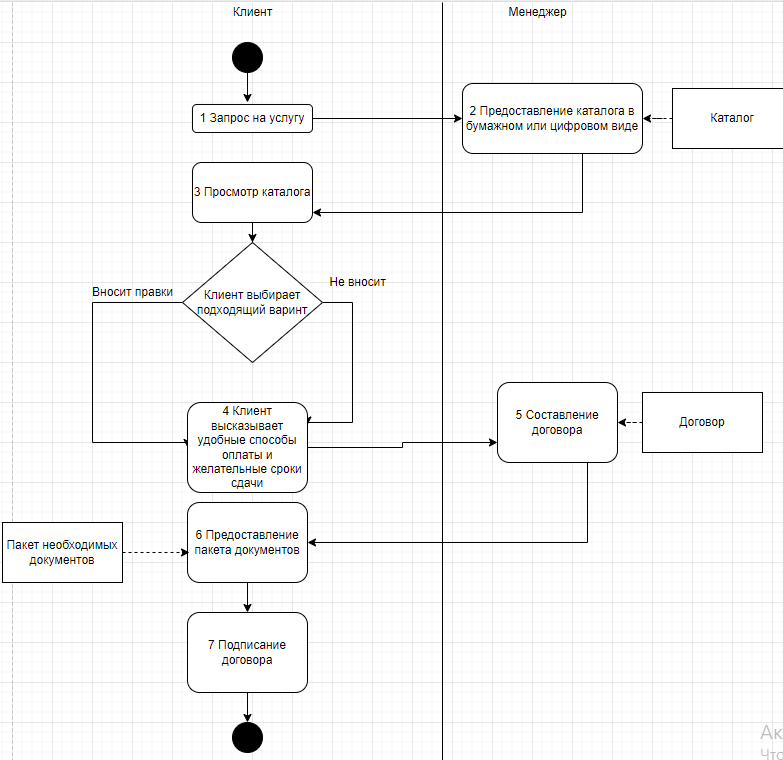


Рис.2 «Диаграмма действий»

# 2.1.5 Таблица операций

Таблица №2. “Таблица операций”

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Диаграмма и номер операции на диаграмме | Операция | Исполнитель | Как часто | Входящие документы (документыоснования) | Исходящий документ (составляемый документ) |
| 1 | 2 | 3 | 4 | 5 | 6 |
| 3-Прд\_фр\_зк | Запрос на услугу | Клиент |  | Каталог | Нет |
| 3-Прд\_фр\_зк | Предоставление каталога в бумажном или цифровом виде | Менеджер |  | Каталог | Нет |
| 3-Прд\_фр\_зк | Просмотр каталога | Клиент |  | Каталог | Нет |
| 3-Прд\_фр\_зк | Клиент высказывает удобные способы оплаты и желательные сроки сдачи | Клиент |  |  | Нет |
| 3-Прд\_фр\_зк | Составление договора | Менеджер |  | Каталог | Договор |
| 3-Прд\_фр\_зк | Предоставление документов | Клиент |  | Пакет необходимых документов | Нет |
| 3-Прд\_фр\_зк | Подписание договора | Клиент |  | Пакет необходимых документов, каталога | Договор нет |

# 2.1.6 Таблица описаний документов

Таблица №3. “Таблица документов”

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Диаграмма и номер операции на диаграмме | Составляемый документ (Исходящий документ) | Операция | Исполнитель | Как часто | Документы-основания (входящие документы) |
| 3-Прд\_фр\_зк\_2 | Каталог | Предоставление каталога в бумажном или цифровом виде | Менеджер | 1 раз | нет |
| 3-Прд\_фр\_зк\_5 | Договор | Составление договора | Менеджер | 1 раз | нет |
| 3-Прд\_фр\_зк\_6 | Пакет документов | Предоставление пакета документов | Клиент | 1 раз | Пакет документов |

# Реализация

# Обоснование выбора средств разработки

HTML – это язык разметки, который указывает браузерам (Google Chrome, Яндекс.Браузер и другим подобным программам), где и какие элементы выводить на странице сайта. Например, где находится заголовок, основной текст, ссылки на другие страницы, меню, списки, таблицы и так далее. CSS – это каскадные таблицы стилей. С их помощью задают оформление различных элементов.

JavaScript - [мультипарадигменный](https://ru.wikipedia.org/wiki/%D0%9C%D1%83%D0%BB%D1%8C%D1%82%D0%B8%D0%BF%D0%B0%D1%80%D0%B0%D0%B4%D0%B8%D0%B3%D0%BC%D0%B5%D0%BD%D0%BD%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5" \o "Мультипарадигменное программирование) [язык программирования](https://ru.wikipedia.org/wiki/%D0%AF%D0%B7%D1%8B%D0%BA_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F). Поддерживает [объектно-ориентированный](https://ru.wikipedia.org/wiki/%D0%9E%D0%B1%D1%8A%D0%B5%D0%BA%D1%82%D0%BD%D0%BE-%D0%BE%D1%80%D0%B8%D0%B5%D0%BD%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%BD%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5), [императивный](https://ru.wikipedia.org/wiki/%D0%98%D0%BC%D0%BF%D0%B5%D1%80%D0%B0%D1%82%D0%B8%D0%B2%D0%BD%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5) и [функциональный](https://ru.wikipedia.org/wiki/%D0%A4%D1%83%D0%BD%D0%BA%D1%86%D0%B8%D0%BE%D0%BD%D0%B0%D0%BB%D1%8C%D0%BD%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5) стили. Является реализацией спецификации [ECMAScript](https://ru.wikipedia.org/wiki/ECMAScript" \o "ECMAScript).

JavaScript обычно используется как встраиваемый язык для программного доступа к объектам [приложений](https://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BC%D0%BF%D1%8C%D1%8E%D1%82%D0%B5%D1%80%D0%BD%D0%B0%D1%8F_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B0). Наиболее широкое применение находит в [браузерах](https://ru.wikipedia.org/wiki/%D0%91%D1%80%D0%B0%D1%83%D0%B7%D0%B5%D1%80) как язык сценариев для придания [интерактивности](https://ru.wikipedia.org/wiki/%D0%98%D0%BD%D1%82%D0%B5%D1%80%D0%B0%D0%BA%D1%82%D0%B8%D0%B2%D0%BD%D0%BE%D1%81%D1%82%D1%8C) [веб-страницам](https://ru.wikipedia.org/wiki/%D0%92%D0%B5%D0%B1-%D1%81%D1%82%D1%80%D0%B0%D0%BD%D0%B8%D1%86%D0%B0)[[8]](https://ru.wikipedia.org/wiki/JavaScript#cite_note-10).

Основные архитектурные черты: [динамическая типизация](https://ru.wikipedia.org/wiki/%D0%94%D0%B8%D0%BD%D0%B0%D0%BC%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B0%D1%8F_%D1%82%D0%B8%D0%BF%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D1%8F), [слабая типизация](https://ru.wikipedia.org/wiki/%D0%A1%D0%BB%D0%B0%D0%B1%D0%B0%D1%8F_%D1%82%D0%B8%D0%BF%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D1%8F), [автоматическое управление памятью](https://ru.wikipedia.org/wiki/%D0%A1%D0%B1%D0%BE%D1%80%D0%BA%D0%B0_%D0%BC%D1%83%D1%81%D0%BE%D1%80%D0%B0_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)), [прототипное программирование](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D1%82%D0%BE%D1%82%D0%B8%D0%BF%D0%BD%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5" \o "Прототипное программирование), функции как [объекты первого класса](https://ru.wikipedia.org/wiki/%D0%9E%D0%B1%D1%8A%D0%B5%D0%BA%D1%82_%D0%BF%D0%B5%D1%80%D0%B2%D0%BE%D0%B3%D0%BE_%D0%BA%D0%BB%D0%B0%D1%81%D1%81%D0%B0).

JavaScript является [объектно-ориентированным](https://ru.wikipedia.org/wiki/%D0%9E%D0%B1%D1%8A%D0%B5%D0%BA%D1%82%D0%BD%D0%BE-%D0%BE%D1%80%D0%B8%D0%B5%D0%BD%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%BD%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5) языком, но используемое в языке [прототипирование](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D1%82%D0%BE%D1%82%D0%B8%D0%BF%D0%BD%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)[[36]](https://ru.wikipedia.org/wiki/JavaScript#cite_note-39)[[37]](https://ru.wikipedia.org/wiki/JavaScript#cite_note-40) обуславливает отличия в работе с объектами по сравнению с традиционными класс-ориентированными языками. Кроме того, JavaScript имеет ряд свойств, присущих [функциональным языкам](https://ru.wikipedia.org/wiki/%D0%A4%D1%83%D0%BD%D0%BA%D1%86%D0%B8%D0%BE%D0%BD%D0%B0%D0%BB%D1%8C%D0%BD%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5) — функции как [объекты первого класса](https://ru.wikipedia.org/wiki/%D0%9E%D0%B1%D1%8A%D0%B5%D0%BA%D1%82_%D0%BF%D0%B5%D1%80%D0%B2%D0%BE%D0%B3%D0%BE_%D0%BA%D0%BB%D0%B0%D1%81%D1%81%D0%B0), объекты как списки, [карринг](https://ru.wikipedia.org/wiki/%D0%9A%D0%B0%D1%80%D1%80%D0%B8%D0%BD%D0%B3" \o "Карринг), [анонимные функции](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%BE%D0%BD%D0%B8%D0%BC%D0%BD%D1%8B%D0%B5_%D1%84%D1%83%D0%BD%D0%BA%D1%86%D0%B8%D0%B8), [замыкания](https://ru.wikipedia.org/wiki/%D0%97%D0%B0%D0%BC%D1%8B%D0%BA%D0%B0%D0%BD%D0%B8%D0%B5_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5))[[38]](https://ru.wikipedia.org/wiki/JavaScript#cite_note-41) — что придаёт языку дополнительную гибкость.

Несмотря на схожий с Си синтаксис, JavaScript по сравнению с языком [Си](https://ru.wikipedia.org/wiki/%D0%A1%D0%B8_(%D1%8F%D0%B7%D1%8B%D0%BA_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F)) имеет коренные отличия:

* [объекты](https://ru.wikipedia.org/wiki/%D0%9E%D0%B1%D1%8A%D0%B5%D0%BA%D1%82_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)) с возможностью [интроспекции](https://ru.wikipedia.org/wiki/%D0%98%D0%BD%D1%82%D1%80%D0%BE%D1%81%D0%BF%D0%B5%D0%BA%D1%86%D0%B8%D1%8F_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5));
* функции как [объекты первого класса](https://ru.wikipedia.org/wiki/%D0%9E%D0%B1%D1%8A%D0%B5%D0%BA%D1%82_%D0%BF%D0%B5%D1%80%D0%B2%D0%BE%D0%B3%D0%BE_%D0%BA%D0%BB%D0%B0%D1%81%D1%81%D0%B0);
* автоматическое [приведение типов](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%B8%D0%B2%D0%B5%D0%B4%D0%B5%D0%BD%D0%B8%D0%B5_%D1%82%D0%B8%D0%BF%D0%BE%D0%B2);
* автоматическая [сборка мусора](https://ru.wikipedia.org/wiki/%D0%A1%D0%B1%D0%BE%D1%80%D0%BA%D0%B0_%D0%BC%D1%83%D1%81%D0%BE%D1%80%D0%B0_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5));
* [анонимные функции](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%BE%D0%BD%D0%B8%D0%BC%D0%BD%D1%8B%D0%B5_%D1%84%D1%83%D0%BD%D0%BA%D1%86%D0%B8%D0%B8).

MongoDB – это ориентированная на документы база данных NoSQL с открытым исходным кодом, которая использует для хранения структуру JSON. Модель данных MongoDB позволяет представлять иерархические отношения, проще хранить массивы и другие более сложные структуры.

Visual Studio Code (VS Code) — редактор исходного кода, разработанный Microsoft для Windows, Linux и macOS;

GitHub Desktop для контроля версий. GitHub — крупнейший веб-сервис для хостинга IT-проектов и их совместной разработки;

любой веб-браузер для просмотра и отладки сайта

# Руководство программиста

На всех окнах есть кнопка добавить, которая в зависимости от окна ведет либо на страницу регистрации, либо на страницу добавления проекта.

Так же на всех страницах, которые имеют списки, элементы содержат три точки, которые позволяют удалить или изменить элемент.

Окно контактов, при открытии выводит список контактов включая сотрудников. При необходимости нужного сотрудника или клиента можно найти с помощью поиска. А для того что бы отделить сотрудников и клиентов можно воспользоваться группировкой.

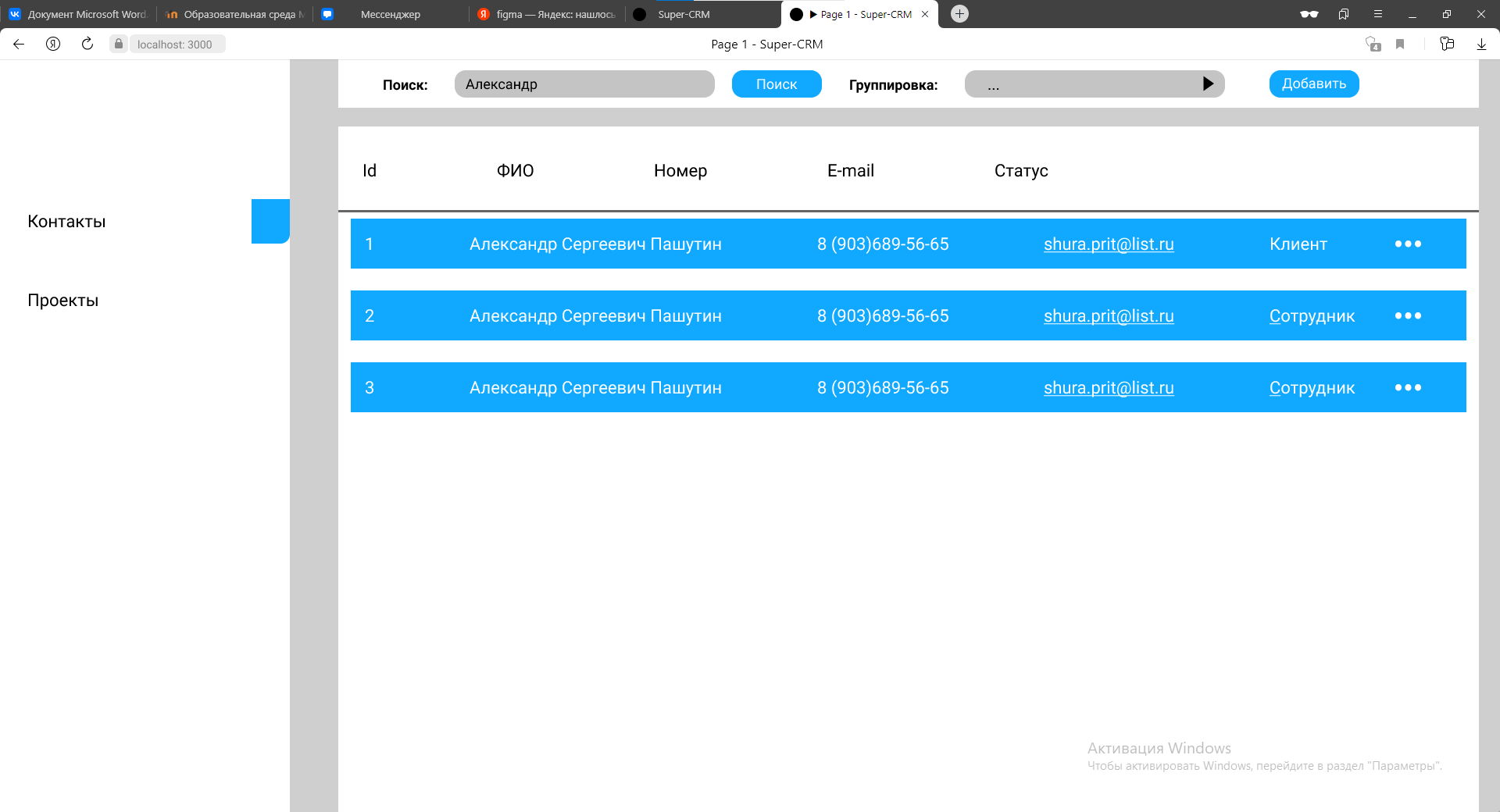


Рис.3 «Окно Контактов»

Окно проектов выводит список проектов, количество этапов внутри и статус готовности.

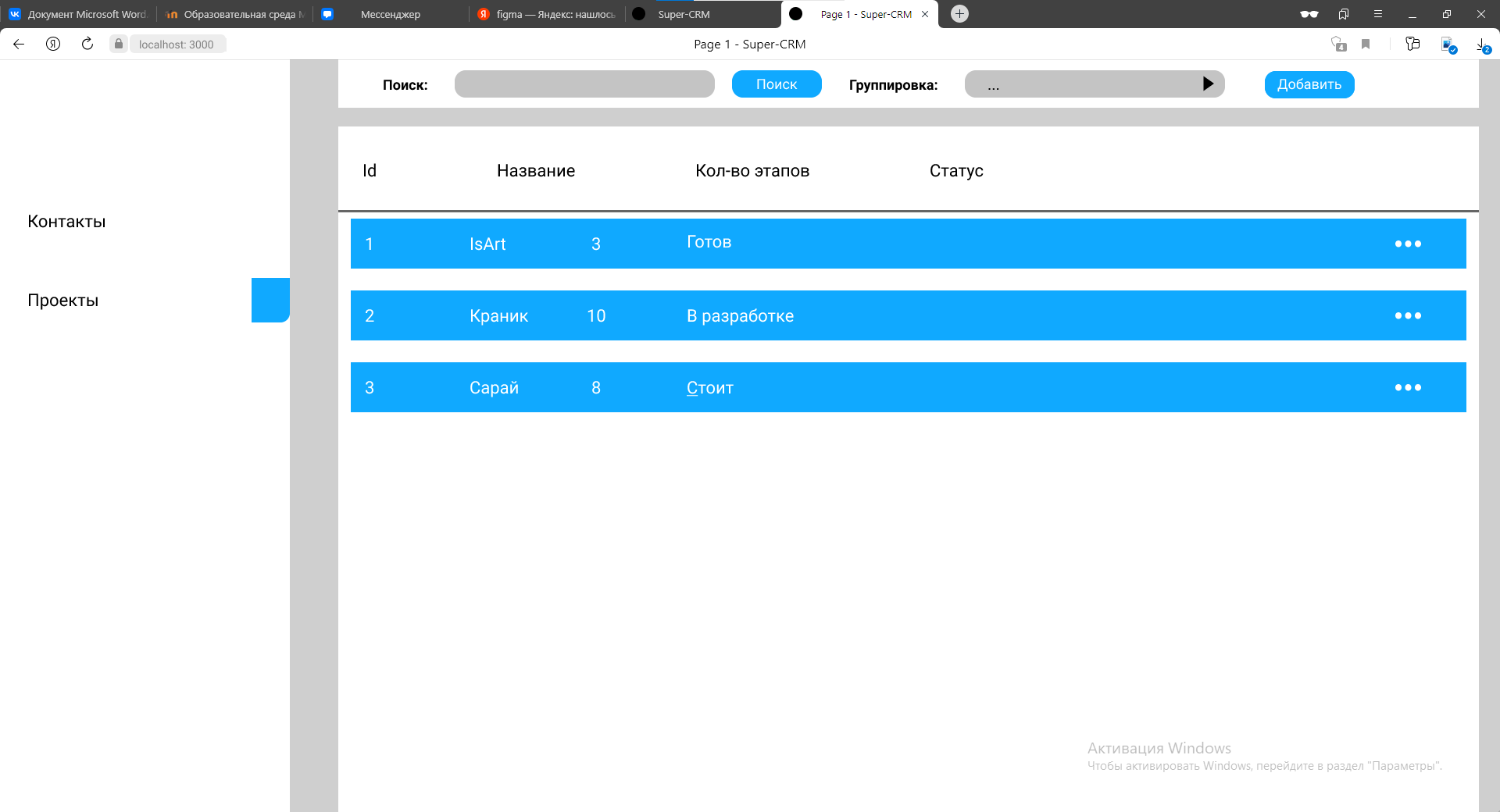


Рис.4 «Окно Проектов»

Окно этапов выводит список этапов и количество задач внутри, а также сотрудника, работающего над данным этапом.

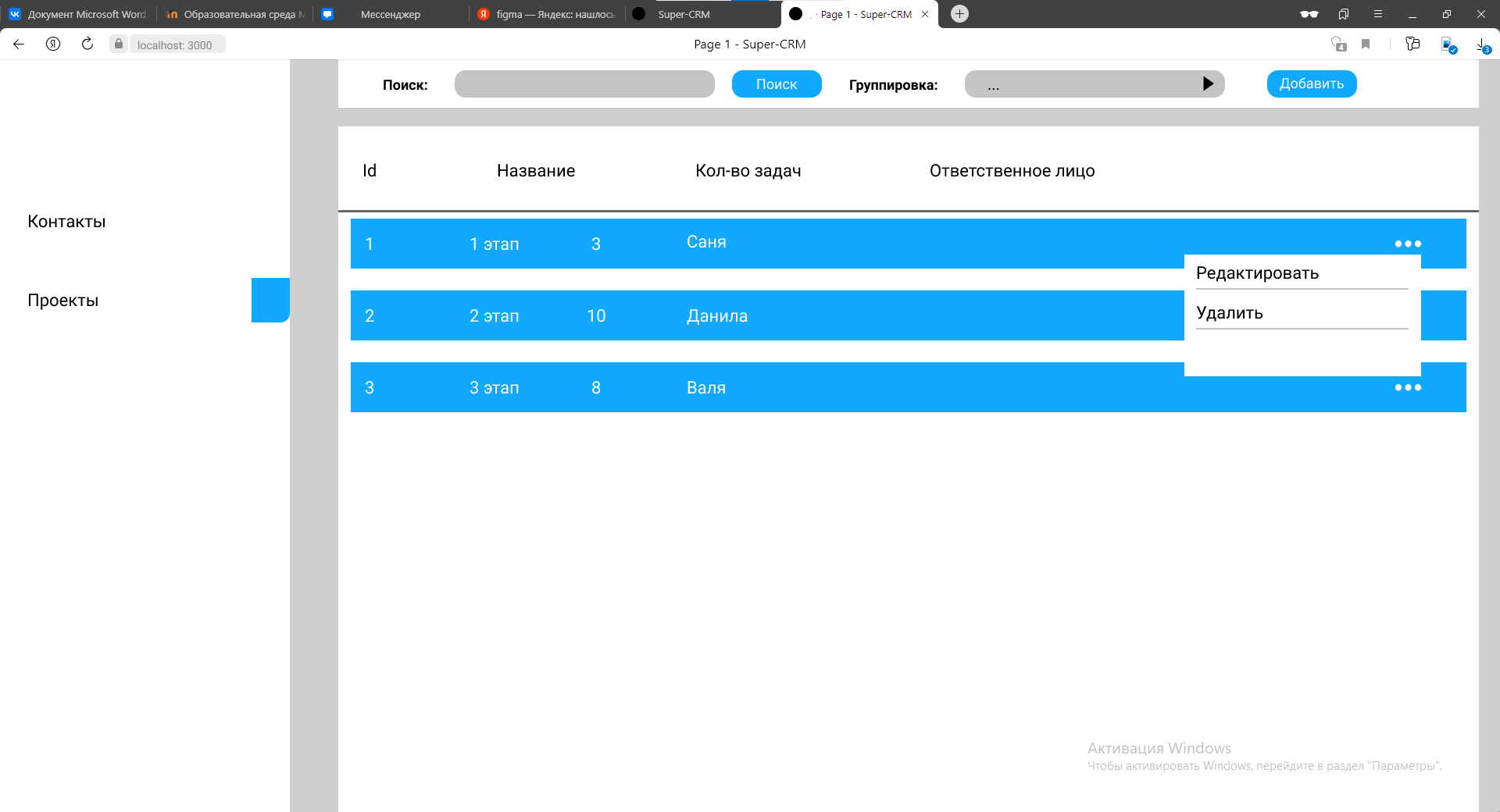


Рис.5 «Окно Этапов»

Окно зада, выводит список задач.

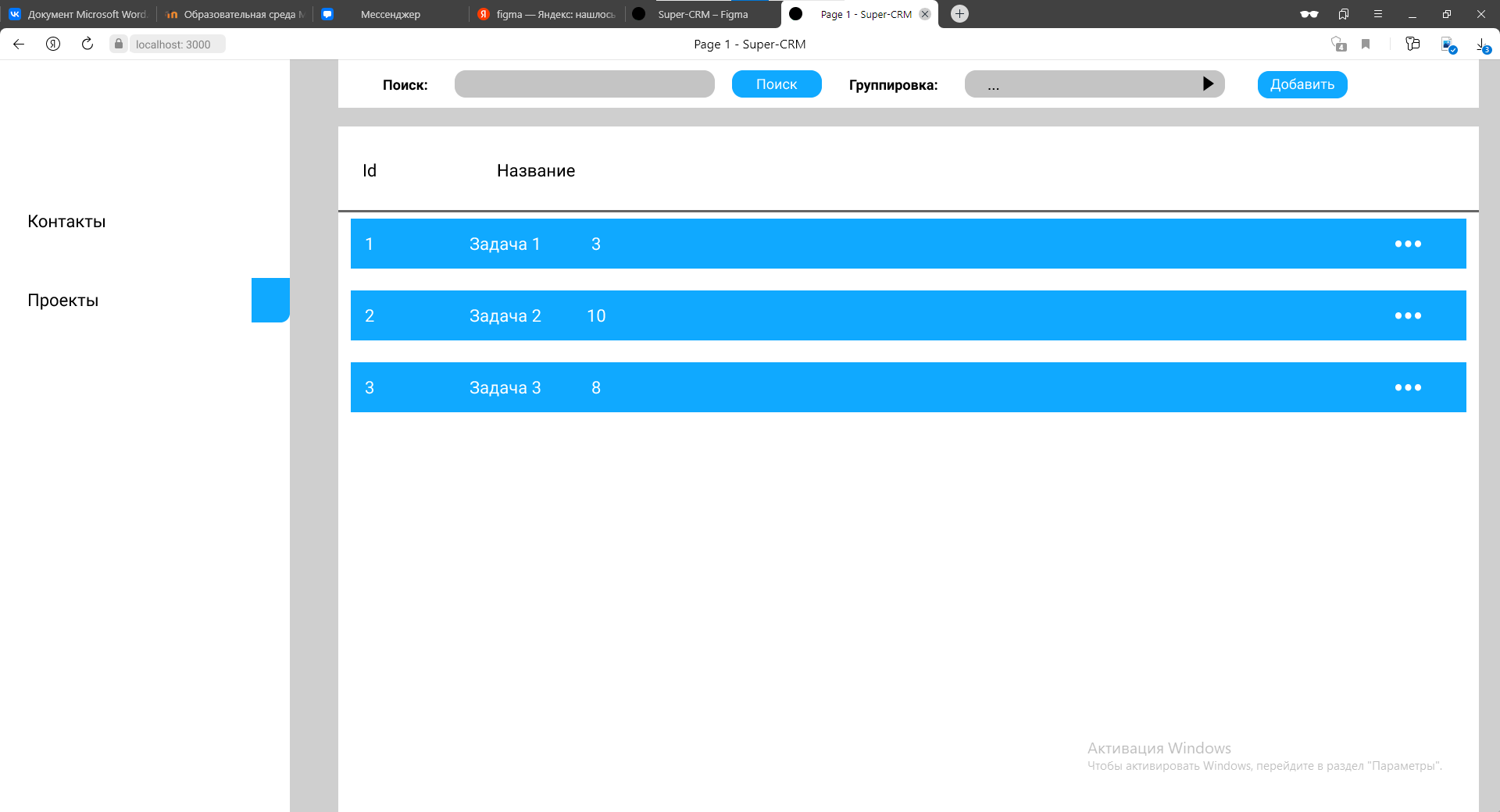


Рис.6 «Окно Задач»

Окно добавления(редактирования). Позволяет добавить новый проект, его список этапов и задач. Так же позволяет редактировать уже существующий проект.

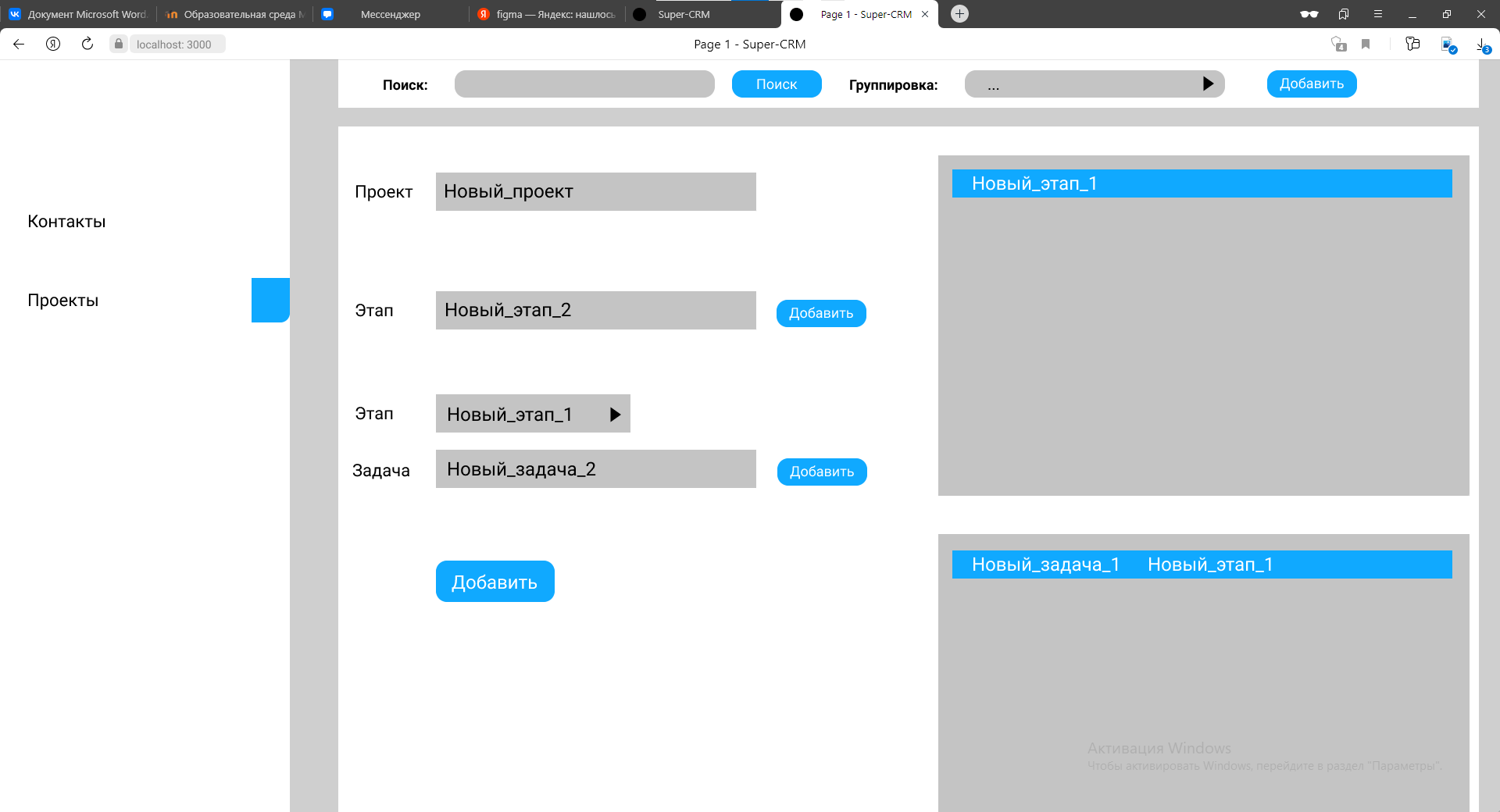


Рис.7 «Окно Добавления проекта»

Окно регистрации, позволяет создавать нового сотрудника или клиента. Для упрощения задачи, и клиенту и сотруднику вводится пароль, но только сотрудник получает его.

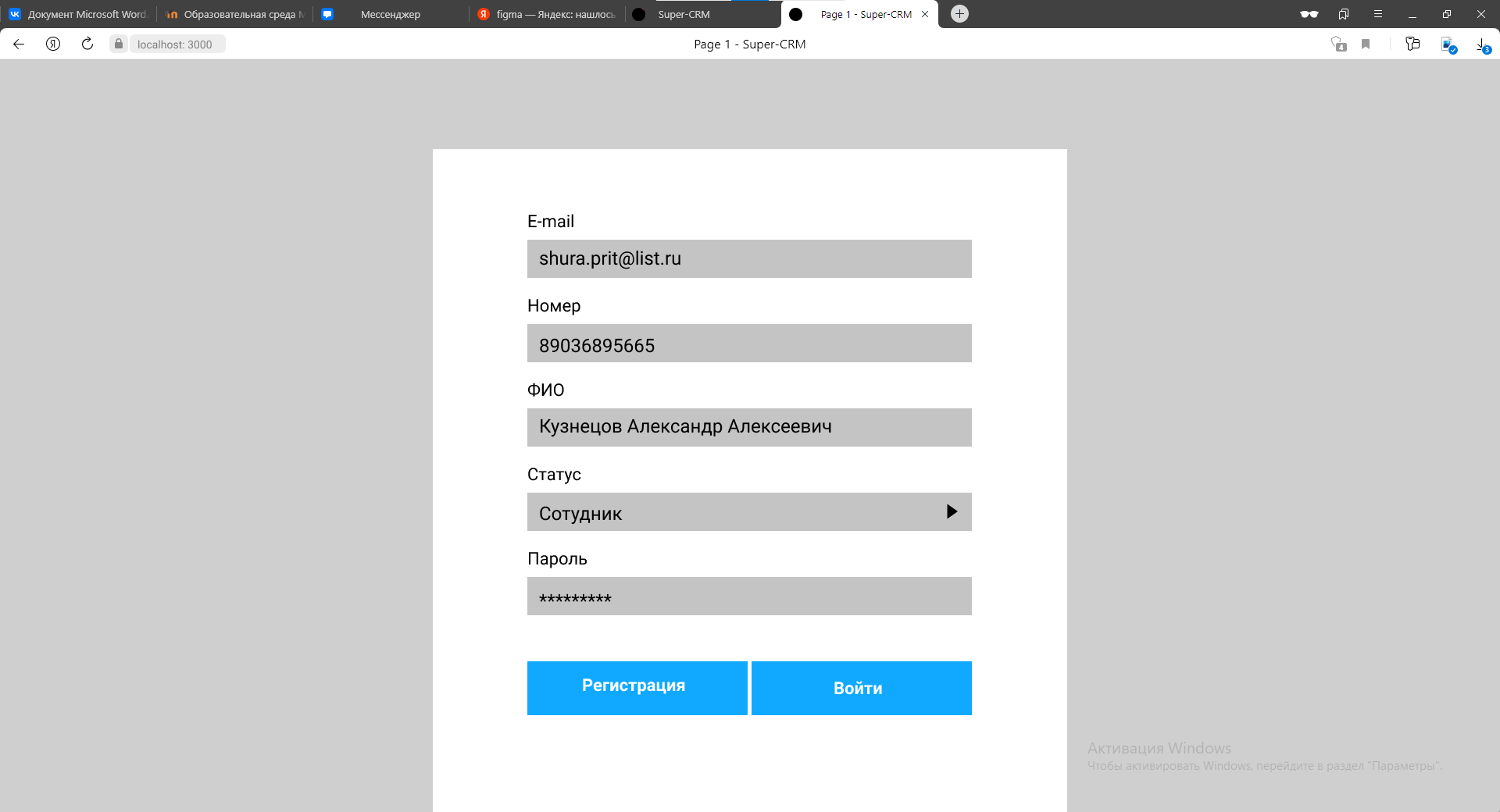


Рис.8 «Окно Регистрации»

Окно входа.

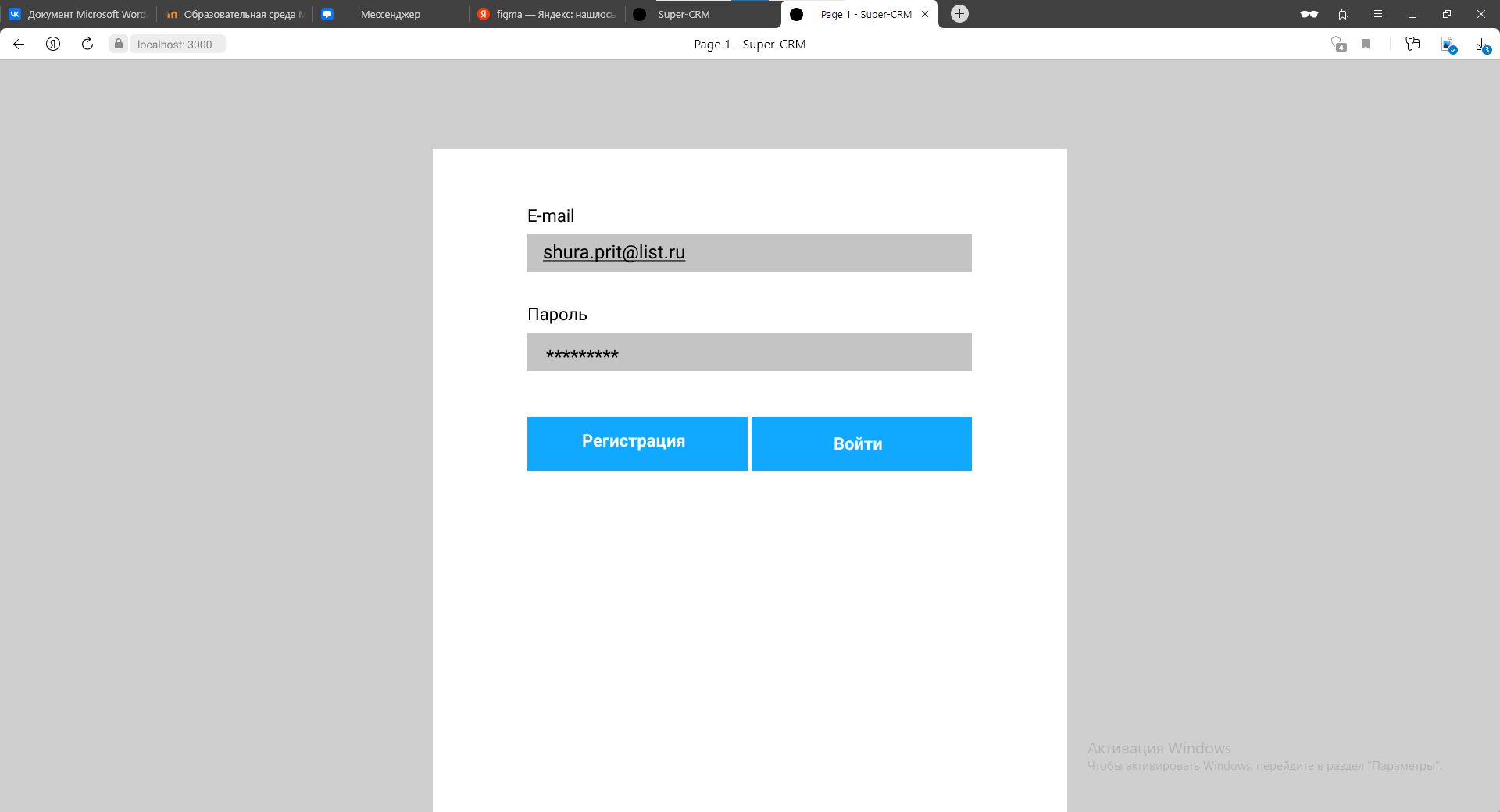


Рис.9 «Окно Входа»



Рис.10 «Таблица Контактов»

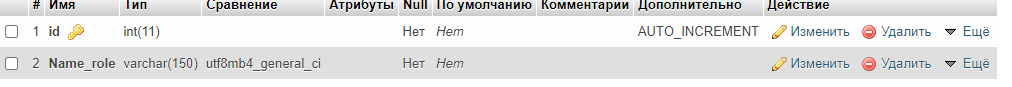


Рис.11 «Таблица Ролей»

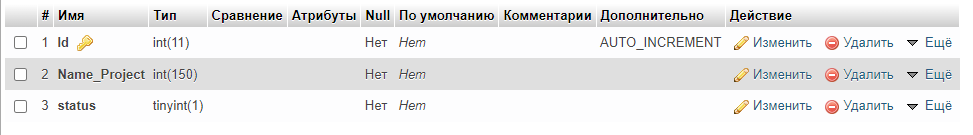


Рис.12 «Таблица Проектов»

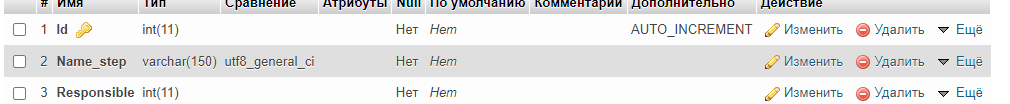


Рис.13 «Таблица Этапы»

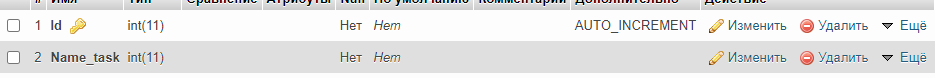


Рис.14 «Таблица Задачи»

# Руководство пользователя

Для входа в приложение нужно перейти по ссылке «ссылка» после чего откроется окно входа:

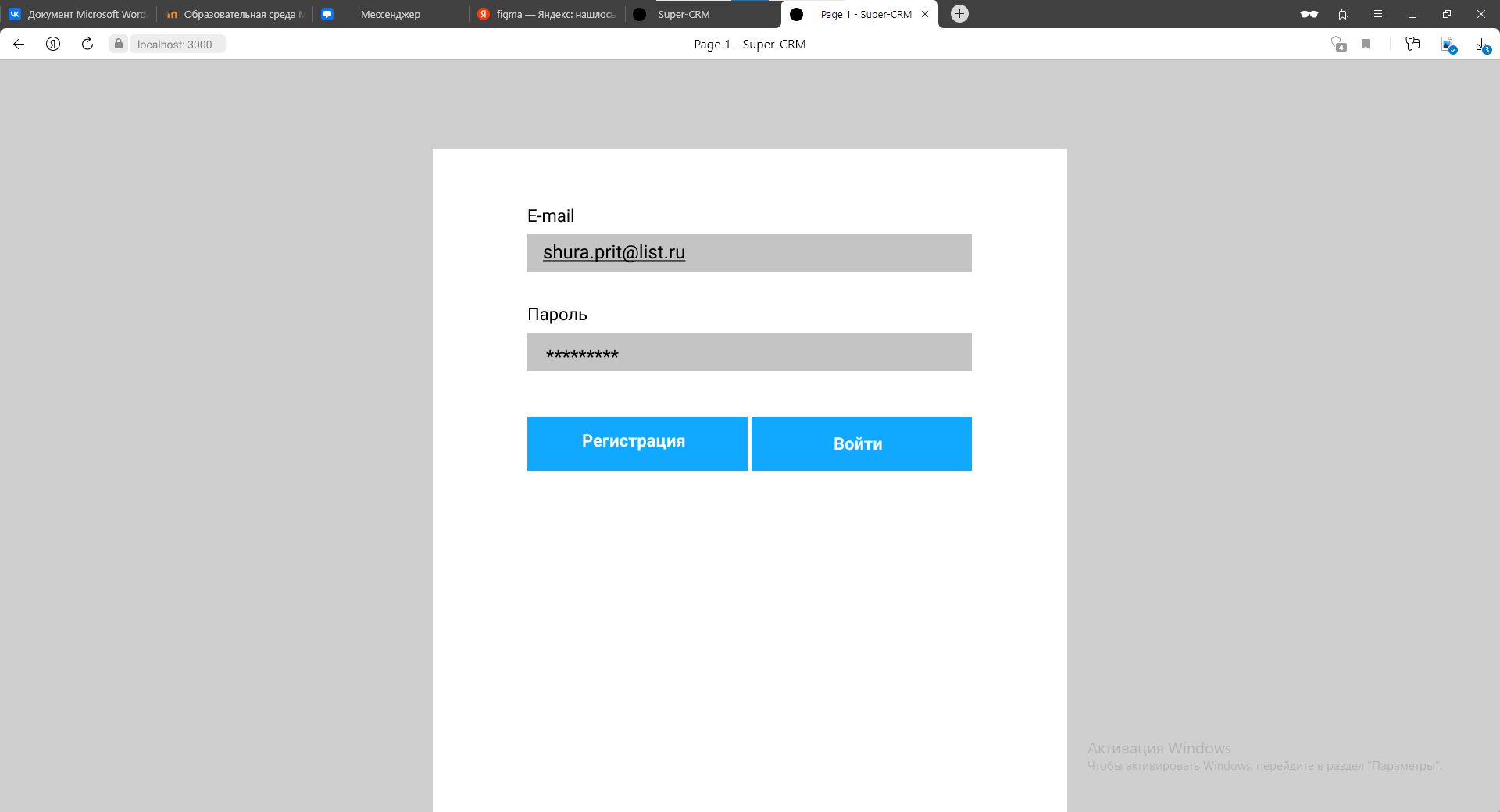


Рис.15 «Окно Входа»

В нем нужно ввести E-mail и пароль если пользователь является сотрудником. Если же сотрудника нет в приложении, нужно кликнуть на «Регистрацию». Откроется окно регистрации.

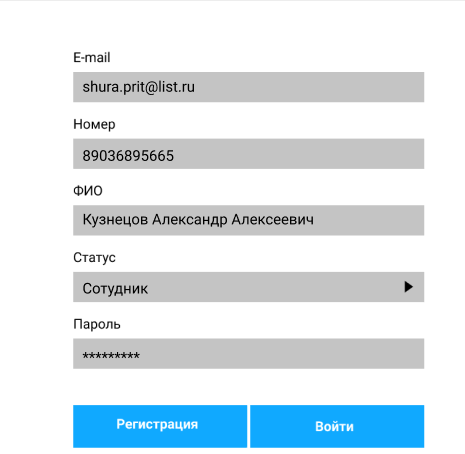


Рис.16 «Окно Регистрации»

Нужно заполнить все поля, для упрощения приложения и для сотрудников, и для клиентов вводится пароль, но пароль выдается только сотруднику вместе с E-mail-лом.

Если Регистрация прошла успешно, откроется окно входа и где нужно войти в приложение. При входе в приложении открывается окно «Контактов».

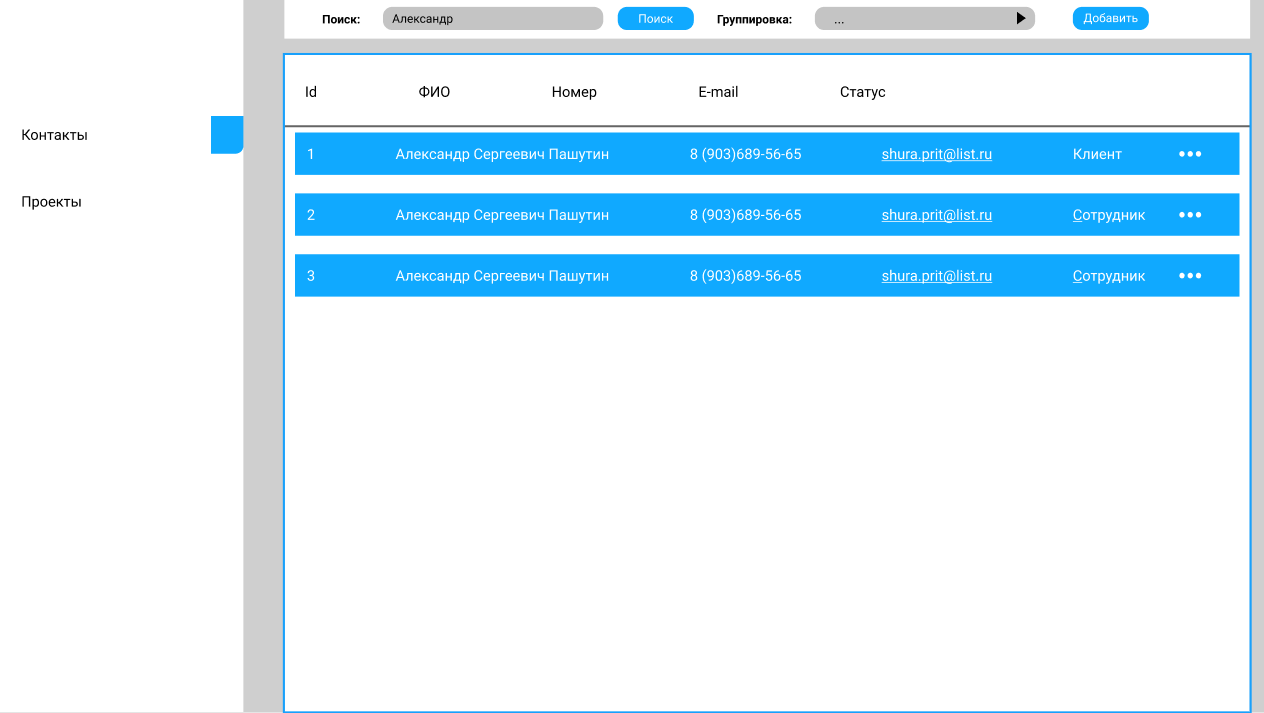


Рис.17 «Окно Контактов»



Рис.18 «Окно Добавления проекта»

Общее:

Для всех дальнейших окон принцип одинаков. Левый блок, вкладки окон меду которыми можно переключатся по клику мышки. В верхнем блоке есть поиск, который в зависимости от окна ищет контакты, проекты и т.д. Группировка, если это окно контакты выводит либо сотрудников, либо клиентов, либо всех сразу. Группировка для остальных окон группировка не работает. Кнопка добавить открывает в зависимости от окна окно регистрации, если это окно контактов или добавление проекта, если это окно проектов, этапов или задач. При клике на три точки на элементе списка открывается контекстное меню, в котором два пункта редактировать и удалить. Пункт редактировать открывает окно добавления, в котором отображаются все связанные данные с проектом, этапом или задачей (см. Рис.18 «Окно Добавления проекта»).

# 4. Тестирование и отладка

Разработка интернет-магазинов и больших информационных сайтов включает в себя создание большого количества сервисов, которые в обязательном порядке требуют тестирования и отладки после завершения работы программистов. Этап тестирования является одним их ключевых и может занимать от 5 до 7 рабочих дней. Как правило, выявить все недоработки и ошибки с первого раза не всегда удается, особенно когда речь идёт о нестандартных сайтах или больших интернет-магазинах. Идеальный вариант — когда тестированием продукта занимаются обе стороны — исполнителю не всегда могут быть понятны очевидные со стороны заказчика или клиента ошибки.  
С стороны программиста в тестировании сайта участвует несколько человек. Как правило, это руководитель проекта, арт-директор и непосредственно программисты. Команда программистов последовательно производит все типичные действия: просмотр всех страниц каталога, добавление контента на сайт, редактирование контента и элементов каталога, добавление товаров в корзину, оформление заказа и так далее.  
При этом обязательными критериями для сдачи проекта являются следующие пункты:

* Добавление, редактирование элементов в административной части с выводом на сайт (новости, статьи, фотогалерея и так далее);
* Добавление и редактирование элементов каталога (свойства элемента, стоимость, фотографии);
* Процедура регистрации и авторизации пользователей;
* Процедура покупки товаров и/или услуг (корзина), оформление заказа (добавление заказа в административную часть и дублирование информации по составу и статусу заказа на почту администратора сайта и клиента);
* Работа дополнительных сервисов (веб-формы, заказ обратного звонка, калькуляторы, фильтры, поиск по сайту и так далее).

Некоторые пункты актуальны исключительно при разработке интернет-магазинов. Тестирование и отладка сайта происходит в закрытом режиме (для сторонних пользователей сайт недоступен) или на тестовом доменном имени, которое также скрыто от сторонних пользователей и поисковых машин.  
Таким образом, в тестировании и отладке сайта принимают участие несколько высококвалифицированных в своей области специалистов, что гарантирует работоспособность проекта. Также не стоит забывать, что в течение месяца после сдачи проекта мы оказываем месяц бесплатной поддержки сайта, в этот период мы устраняем все возможные оставшиеся недоработки и упущенные моменты.

# 4.1 Таблица класса эквивалентности

Таблица №4. “Таблица классов”

|  |  |  |
| --- | --- | --- |
| Входное или выходное событие | Допустимые классы эквивалентности | Недопустимые классы эквивалентности |
| Ввод номера | Числовой символы | Буквенный символы |
| Email | при регистрации должен после имени иметь «@» и доменное имя. | Пропуск доменного имени и «@» |

# 5. Разработка механизмов защиты данных

Для обеспечения защиты сайта нужно соблюсти как можно больше пунктов из списка ниже, тогда сайт будет максимально защищён. В любом случаи все методы защиты не дают 100% гарантии на безопасность.

Можно выделить несколько основных способов защитить свой сайт:

* обеспечить защиту от DDoS-атак;
* подключить SSL-сертификат;
* использовать [надёжный хостинг](https://vdsina.ru/?partner=habr15);
* использовать безопасные плагины/библиотеки/фреймворки/CMS (далее – «сторонние модули»);
* применять существующие техники защиты от SQL-инъекций и XSS-атак;
* обеспечить ведение журнала веб-сайта и мониторинг событий безопасности;
* производить регулярное резервное копирование веб-сайта и всех важных данных;
* использовать надёжные и сложные пароли, а также защиту от перебора паролей;
* в случае наличия административной панели, с помощью которой происходит управление содержимым веб-сайта, необходимо изменить стандартный адрес входа и обеспечить контроль доступа.

Естественно, у каждого пункта есть своё «но» и ряд подпунктов, на которых следует заострить внимание. Также их можно разделить на подгруппы исходя из следующих соображений: одни действия требуют одноразового подключения, настройки и редких проверок работоспособности (настройка хостинга и SSL-сертификата), а другие подразумевают под собой постоянные проверки, обновления и требуют пристального внимание (всё остальное).

**Надёжный хостинг и SSL**

Итак, начнём с того, что в нашем арсенале имеется надёжный сервер или виртуальный хостинг. Что касается подключения SSL-сертификата, то это является обязательной мерой.

**SQL injection and XSS**

Уязвимости SQL-инъекций позволяют злоумышленникам выполнять произвольный код SQL в базе данных, позволяя получать, изменять или удалять данные независимо от разрешений пользователя. Успешная инъекционная атака может подделать удостоверения, создать новые удостоверения с правами администратора, получить доступ ко всем данным на сервере или уничтожить / изменить данные, чтобы сделать их непригодными для использования. Типы внедрения SQL включают внедрение SQL на основе ошибок, внедрение SQL на основе логических ошибок и внедрение SQL на основе времени.

# Заключение

В ходе разработки приложения были определены требования к функциональности, безопасности и составу, и параметрам технических средств.

Далее были выбраны CASE-средства для разработки диаграммы прецедентов и диаграммы действий. На основе диаграммы прецедентов был составлен словесный алгоритм бизнес-процесса. На основе словесного описания бизнес-процесса была построена диаграмма действий, благодаря которой построились две таблицы: таблица операций и таблица документов.

После чего началась реализация проекта, а если быть точнее разработка самого приложения. Интерфейс приложения был разработан с учетом пожеланий заказчика.

При завершении процесса разработки начался процесс тестирования и отладки, при котором были исправлены недочеты и переработаны некоторые моменты.

Как только проект был готов для него были написаны 2 руководства, руководство программиста и пользователя. В первом были описаны окна приложения и таблицы БД. Во втором было ознакомление пользователя с приложением.

Приложение имеет надежную защитную систему. Все пароли хешируются, встроена защита от DDoS-атак и SQL-Injections. Хост имеет SSL-сертификат.

# Список литературы

1. <https://blog.logrocket.com/building-structuring-node-js-mvc-application> - автор Precious Luke
2. <https://dev.to/eaetukudo/understanding-mvc-pattern-in-nodejs-2bdn> - автор E[mmanuel Etukudo](https://dev.to/eaetukudo)
3. Оппель, Эндрю Дж. SQL. Полное руководство / Оппель Эндрю Дж.. - М.: Диалектика / Вильямс, 2016. - 902 c.
4. А. В. Рудаков - Технология разработки программных продуктов 2006г, 2-е издание.
5. Грекул\_Проектирование\_ИС. – Курс лекций. Учебное пособие 2005г
6. Технология разработки программных продуктов. Практикум. Рудаков А.В и Федорова Г.Н. 2014г. Издательский центр «Академия»