Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Вятский государственный университет»

Колледж ВятГУ

**ТЕХНИЧЕСКИЙ ПРОЕКТ**  
Информационная система для клининговой компании

Киров,

2024

**Содержание**

[1. Концепция предлагаемого решения 3](#_Toc180356763)

[1.1. Пользовательский сценарий и интерфейс 3](#_Toc180356764)

[1.2. Организация хранения 10](#_Toc180356765)

[2. Бизнес-процессы системы 12](#_Toc180356766)

# **Концепция предлагаемого решения**

## **Пользовательский сценарий и интерфейс**

. Пользовательские истории представлены в таблице 1.

Таблица 1 – Пользовательские истории

|  |  |  |  |
| --- | --- | --- | --- |
| **№** | **Роль** | **Действие** | **Цель** |
| 1 | Администратор | Создание новых записей | Чтобы заносить актуальные данные в ИС. |
| 1.1 | Добавить нового клинера | Чтобы затем пользователь сайта смогу увидеть всех клинеров компании. |
| 1.2 | Добавить новую заявку | Чтобы затем направить на выполнение заказа клинеров. |
| 1.3 | Добавить новую услугу | Чтобы актуализировать услуги и их описания в ИС. |
| 1.4 | Добавить нового пользователя | Чтобы в дальнейшем добавленный пользователь мог авторизоваться в ИС. |
| 1.4.1 | Добавить нового администратора | Чтобы затем в ИС было несколько администраторов. |
| 2 | Редактирование записей | Чтобы внести изменения в существующие записи. |
| 2.1 | Изменить информацию о клинере | Чтобы внести изменения в информации о клинере. |
| 2.2 | Изменить информацию о заявке | Чтобы внести изменения в информации о заявке. |
| 2.2.1 | Изменить статус заявки | Чтобы помечать как продвигается заказ. |
| 2.3 | Изменить информацию об услуге | Чтобы внести изменения в информации об услуге. |
| 2.4 | Изменить администратора | Чтобы внести изменения в информации об администраторе. |
| 3 | Администратор | Удаление записей | Чтобы удалить устаревшую или ошибочную информацию. |
| 3.1 | Удалить клинера | Чтобы удалить устаревшую или ошибочную информацию о клинере. |
| 3.2 | Удалить заявку | Чтобы удалить устаревшую или ошибочную информацию о заявке. |
| 3.3 | Удалить услугу | Чтобы удалить устаревшую или ошибочную информацию об услуге. |
| 3.4 | Удалить пользователя | Чтобы удалить устаревшую или ошибочную информацию о пользователе. |
| 4 | Администратор | Распределить клинеров по заявкам | Чтобы назначать клинера (бригаду клинеров) для выполнения заявки. |
| 5 | Клиент | Отправить заявку | Чтобы в дальнейшем получить оказание выбранного перечня услуг. |
| 5.1 | Выбрать услуги | Чтобы в дальнейшем получить оказание выбранного перечня услуг. |
| 6 | Клиент | Просмотреть информацию о сайте | Чтобы узнать больше о клининговой компании. |
| 6.1 | Просмотреть услуги компании | Чтобы ознакомиться с доступными услугами, предоставляемыми компанией. |
| 6.2 | Просмотреть клинеров | Чтобы ознакомиться с информацией о доступных клинерах. |
| 7 | Пользователь | Авторизоваться в Системе | Чтобы войти в личный кабинет/панель администратора. |
| 7.1 | Зарегистрироваться в Системе | Чтобы в дальнейшем была возможность входа в личный кабинет/панель администратора. |

Все пользовательские сценарии представлены в таблице 2.

Таблица 2 – Пользовательские сценарии

|  |  |  |
| --- | --- | --- |
| **№** | **Пользователь** | **Система** |
| 1 | Администратор заполнил поля для ввода нового клинера корректными данными и нажал на кнопку «Сохранить» | * 1. Система получает введённые данные.   2. Сохраняет данные в БД.   3. Переходит на страницу «Клинеры», на которой отображены все раннее занесённые клинеры. |
| 2 | Администратор заполнил поля для ввода новой заявки корректными данными и нажал на кнопку «Сохранить» | * 1. Система получает введённые данные.   2. Сохраняет данные в БД.   3. Переходит на страницу «Заявки», на которой отображены все раннее занесённые заявки. |
| 3 | Администратор заполнил поля для ввода новой услуги корректными данными и нажал на кнопку «Сохранить» | * 1. Система получает введённые данные.   2. Сохраняет данные в БД.   3. Переходит на страницу «Услуги», на которой отображены все раннее занесённые услуги. |
| 4 | Администратор заполнил поля для ввода нового пользователя корректными данными и нажал на кнопку «Сохранить» | * 1. Система получает введённые данные.   2. Сохраняет данные в БД.   3. Переходит на страницу «Пользователи», на которой отображены все раннее занесённые пользователи. |
| 5 | Администратор заполнил поля для редактирования клинера и нажал на кнопку «Сохранить» | * 1. Система получает введённые данные.   2. Изменяет данные в БД.   3. Переходит на страницу «Клинеры», на которой отображены все раннее занесённые клинеры. |
| 6 | Администратор заполнил поля для редактирования заявки и нажал на кнопку «Сохранить» | 1. Система получает введённые данные.  2. Изменяет данные в БД.  3. Переходит на страницу «Заявки», на которой отображены все раннее занесённые заявки. |
| 7 | Администратор заполнил поля для редактирования услуги и нажал на кнопку «Сохранить» | 1. Система получает введённые данные.  2. Изменяет данные в БД.  3. Переходит на страницу «Услуги», на которой отображены все раннее занесённые услуги. |
| 8 | Администратор заполнил поля для редактирования пользователя и нажал на кнопку «Сохранить» | 1. Система получает введённые данные.  2. Изменяет данные в БД.  3. Переходит на страницу «Пользователи», на которой отображены все раннее занесённые пользователи. |
| 9 | Администратор нажимает на кнопку «Удалить клинера» | 1. Система отправляет подтверждающее сообщение.  1.1 Если ответ «Да»:  А) Получает ID удаляемого клинера.  Б) Находит по полученному ID запись в БД.  В) Удаляет найденную запись.  1.2 Если ответ «Нет»:  А) См. пункт 2.  2. Переходит на страницу «Клинеры». |
| 10 | Администратор нажимает на кнопку «Удалить заявку» | 1. Система отправляет подтверждающее сообщение.  1.1 Если ответ «Да»:  А) Получает ID удаляемой заявки.  Б) Находит по полученному ID запись в БД.  В) Удаляет найденную запись.  1.2 Если ответ «Нет»:  А) См. пункт 2.  2. Переходит на страницу «Заявки». |
| 11 | Администратор нажимает на кнопку «Удалить услугу» | 1. Система отправляет подтверждающее сообщение.  1.1 Если ответ «Да»:  А) Получает ID удаляемой услуги.  Б) Находит по полученному ID запись в БД.  В) Удаляет найденную запись.  1.2 Если ответ «Нет»:  А) См. пункт 2.  2. Переходит на страницу «Услуги». |
| 12 | Администратор нажимает на кнопку «Удалить пользователя» | 1. Система отправляет подтверждающее сообщение.  1.1 Если ответ «Да»:  А) Получает ID удаляемого пользователя.  Б) Находит по полученному ID запись в БД.  В) Удаляет найденную запись.  1.2 Если ответ «Нет»:  А) См. пункт 2.  2. Переходит на страницу «Пользователи». |
| 13 | Администратор распределяет клинеров по заявкам | * + 1. Получает ID выбранного клинера.     2. Получает ID выбранной заявки     3. Заносит эти идентификаторы в связующую таблицу «Бригада» в БД. |
| 14 | Клиент заполнил все поля для ввода данных в заявку и нажал на кнопку «Отправить» | 1. Автоматически генерирует ID для новой записи в БД.  2. Заносит в поля таблицы все введённые данные.  3. Показывает всплывающее окно о том, что заявка успешно отправлена.  4. Переходит на главную страницу. |
| 15 | Пользователь авторизуется в Системе | 1. Получает логин и пароль.  2. Находит запись в БД с введёнными логином и паролем.  3. Получает ID найденной записи.  4. Проверяет роль пользователя в Системе.  5. Если роль – администратора, то осуществляется переход в панель администратора, если роль – не администратор, то осуществляется переход в личный кабинет. |

При создании диаграммы прецедентов был выделен набор акторов, который представлен в таблице 3.

Таблица 3 – Акторы и их описание

|  |  |
| --- | --- |
| **Акторы** | **Краткое описание** |
| Клиент | Пользователь услугами компании |
| Администратор | Специалист, ответственный за координацию и управление определенными аспектами деятельности компании. |
| Пользователь | Человек, использующий ИС |

Рассмотрим, какие возможности должна предоставлять система:

* Актор «Клиент» пользуется системой для отправки заявок на оказание набора предложенных услуг.
* Актор «Администратор» пользуется системой для обработки заявок, добавления новых клинеров и услуг, а также для формирования клининговых бригад для оказания услуг.

Диаграмма прецедентов представлена на рисунке 1.

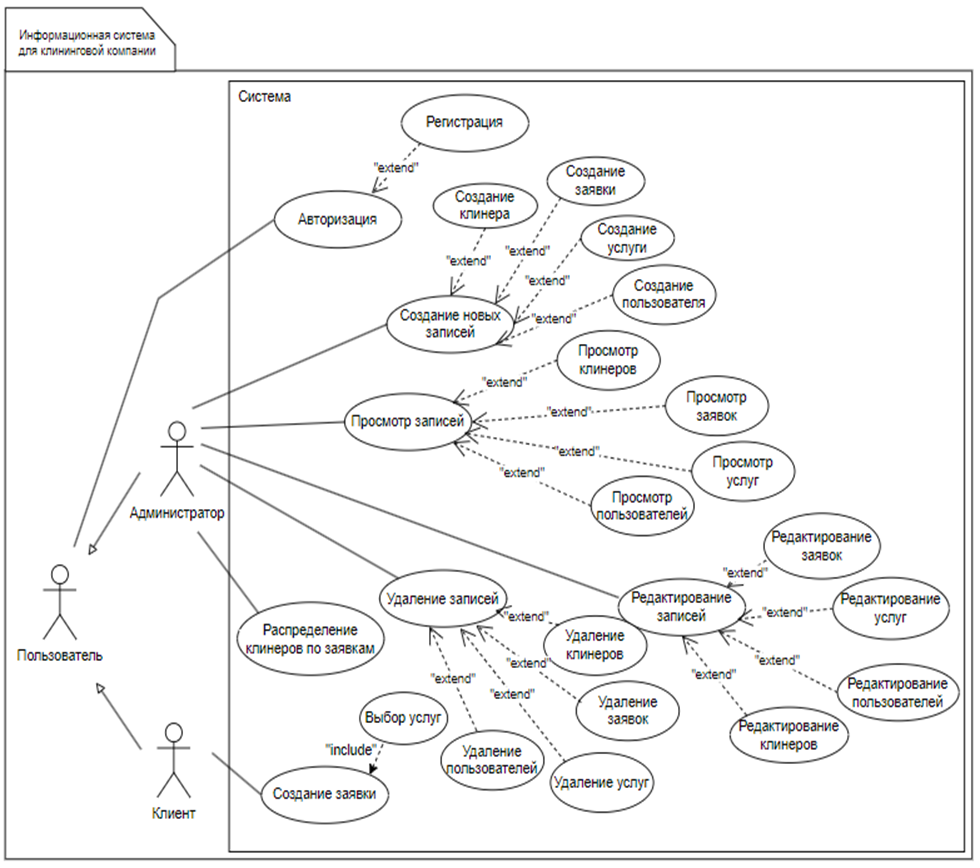


Рисунок 1 – Диаграмма прецедентов

## **Организация хранения**

Для организации хранения данных используется СУБД MySQL.

На рисунке 2 представлена физическая ER-диаграмма базы данных.

Изображение выглядит как текст, снимок экрана, диаграмма, Параллельный

Автоматически созданное описание

Рисунок 2 – Физическая ER-диаграмма

Исходя из диаграммы можно выделить 6 сущностей:

* Клиент.
* Уборщик.
* Заказ.
* Услуга.
* Список услуг.
* Бригада.

В таблице 5 выделен набор сущностей с их атрибутами.

Таблица 5 – Набор сущностей и их атрибутов

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Сущность** | **Свойство** | **Тип** | **Уникальность** | **Обязательность заполнения** | **Ограничения** |
| Client | ClientId | Int | Да | Да | PK |
| Surname | String |  | Да |  |
| Name | String |  | Да |  |
| MiddleName | String |  |  |  |
| PhoneNumber | String |  | Да |  |
| BirthDate | Date |  | Да |  |
| Cleaner | CleanerId | Int | Да | Да | PK |
| Surname | String |  | Да |  |
| Name | String |  | Да |  |
| MiddleName | String |  |  |  |
| PhoneNumber | String |  | Да |  |
| BirthDate | Date |  | Да |  |
| Order | OrderId | Int | Да | Да | PK |
| DateRegistration | Date |  | Да |  |
| Status | String |  | Да |  |
| Address | String |  | Да |  |
| Date | Date |  | Да |  |
| Time | String |  | Да |  |
| Workload | Int |  | Да |  |
| ClientId | Int |  | Да | FK |
| Service | ServiceId | Int |  | Да | PK |
| Title | String |  | Да |  |
| Description | String |  | Да |  |
| Price | Ushort |  | Да | >0 |
| ServiceList | ServiceId | Int | Да | Да | FK |
| OrderId | Int | Да | Да | FK |
| CleaningCrew | CleanerId | Int | Да | Да | FK |
| OrderId | Int | Да | Да | FK |

Связи между сущностями:

«Клиент» – «Заказ»: один ко многим, у каждого заказа определен конкретный клиент, но один клиент может оформлять несколько заказов.

«Заказ» – «Уборщик»: многие ко многим, конкретный заказ в данный момент времени может быть задействован в нескольких бригадах, но в этих бригадах может быть несколько уборщиков.

«Заказ» – «Услуга»: многие ко многим, конкретный заказ в данный момент времени может содержать несколько услуг.

# **Бизнес-процессы системы**

На рисунке 3 представлена контекстная диаграмма IDF0 для ИС.

Изображение выглядит как текст, диаграмма, снимок экрана, линия

Автоматически созданное описание

Рисунок 3 – Контекстная диаграмма IDF0

Исходя из контекстной диаграммы можно сделать вывод, что входными данными являются информации о заявке, клинерах и услугах, ресурсами является администратор и клиент, методами контроля являются устав компании и законодательство РФ, а выходными данными является обработанная заявка.

На рисунке 4 представлена декомпозиция контекстной диаграммы.

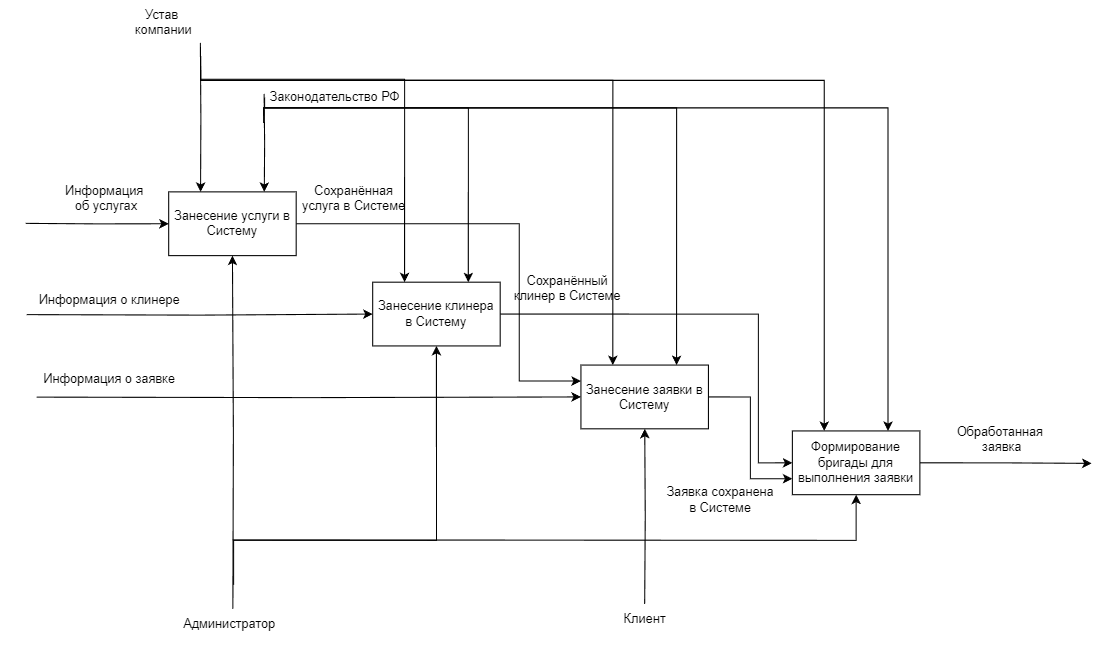


Рисунок 4 – Декомпозиция контекстной диаграммы

Исходя из декомпозиции можно сделать вывод, что для получения обработанной заявке нужно пройти через 4 процесса: Занесение услуги, клинера, заявки в систему и формирование бригады для выполнения заявки.

На рисунке 5 представлена декомпозиция процесса занесения заявки в Систему.

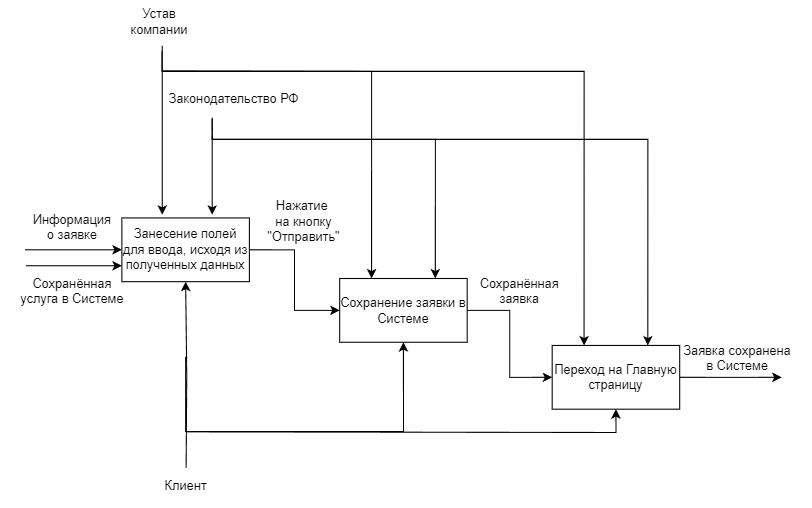


Рисунок 5 – Декомпозиция процесса занесения заявки

На рисунке 6 представлена декомпозиция процесса занесения клинера в систему.

Изображение выглядит как текст, диаграмма, Технический чертеж, План

Автоматически созданное описание

Рисунок 6 – Декомпозиция процесса занесения клинера в систему

На рисунке 7 представлен процесс декомпозиции процесса сохранения услуги в системе.

Изображение выглядит как текст, диаграмма, План, Технический чертеж

Автоматически созданное описание

Рисунок 7 – Декомпозиция процесса сохранения услуги в Системе

На рисунке 8 представлен процесс формирования бригад для выполнения заявки.

Изображение выглядит как диаграмма, текст, План, Технический чертеж

Автоматически созданное описание

Рисунок 8 – Процесс формирования бригад

# **Структура программы и алгоритмы**

Выбранный язык программирования – C# с использованием фреймворка ASP.NET Core MVC. ASP.NET (Active Server Pages для .NET) — платформа разработки веб-приложений, в состав которой входят веб-сервисы, программная инфраструктура и модель программирования. Технология построения пользовательского интерфейса и обработки ввода пользователя – MVC. ASP.NET Core MVC представляет в общем виде построения приложения вокруг трех основных компонентов - Model (модели), View (представления) и Controller (контроллеры), где модели отвечают за работу с данными, контроллеры представляют логику обработки запросов, а представления определяют визуальную составляющую.

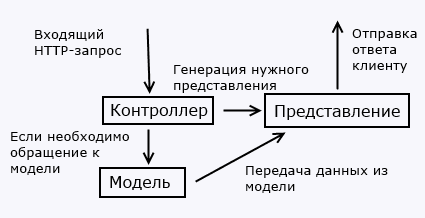


Рисунок 9 – MVC

# **Выбранная архитектура системы**

Для созданного решения была выбрала двухзвенная архитектура «Клиент-сервер» с базой данных. Архитектура MySQL включает в себя две основные компоненты: клиентскую и серверную. Серверная часть представляет собой основу, которая управляет базой данных и обработкой запросов. Клиентская часть выполняет запросы и получает ответы от серверной части.

Архитектура приложения представлена на рисунке 10.

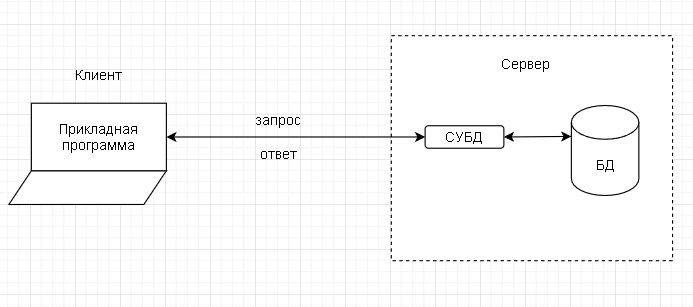


Рисунок 10 – Архитектура приложения