|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  | | --- | --- | --- | |  |  |  | | МИНОБРНАУКИ РОССИИ | | | | Федеральное государственное бюджетное образовательное учреждение  высшего образования  **«МИРЭА – Российский технологический университет»**  **РТУ МИРЭА** | | | | |
| Институт информационных технологий (ИТ) | |
| Кафедра инструментального и прикладного программного обеспечения (ИиППО) | |

|  |  |  |  |
| --- | --- | --- | --- |
| **ОТЧЁТ ПО ПРАКТИЧЕСКИМ РАБОТАМ** | | | |
| **по дисциплине «Проектирование информационных систем»**  на тему  **«Система контроля закупок и складского учёта магазина одежды»** | | | |
|  | | | |
| Выполнил студент группы ИКБО-20-21 | | Шаповалов К.А. | |
|  | |  | |
| Принял  *Ассистент* | | Литвинов В.В. | |
| Практические работы выполнены | «\_\_\_»\_\_\_\_\_\_\_2024 г. | | (подпись студента) | |
| «Зачтено» | «\_\_\_»\_\_\_\_\_\_\_2024 г. | | (подпись руководителя) | |
|  |  | |  | |

Москва 2024

СОДЕРЖАНИЕ

[ВВЕДЕНИЕ 6](#_Toc162543381)

[ПРАКТИЧЕСКАЯ РАБОТА № 1 7](#_Toc162543382)

[ПРАКТИЧЕСКАЯ РАБОТА № 2 22](#_Toc162543413)

[ПРАКТИЧЕСКАЯ РАБОТА №3 28](#_Toc162543415)

[ПРАКТИЧЕСКАЯ РАБОТА №4 54](#_Toc162543470)

[ПРАКТИЧЕСКАЯ РАБОТА №5 59](#_Toc162543471)

[ПРАКТИЧЕСКАЯ РАБОТА №6 65](#_Toc162543472)

[ПРАКТИЧЕСКАЯ РАБОТА №7 75](#_Toc162543473)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 83](#_Toc162543474)

ВВЕДЕНИЕ

В современном мире, где эффективность и точность играют ключевую роль в бизнесе, система контроля закупок и складского учёта магазина одежды становится неотъемлемым инструментом для успешного управления предприятием. Она представляет собой надежное решение для оптимизации процессов снабжения, управления запасами и контроля за движением товаров на складе.

Одной из основных причин востребованности такой системы является её способность предоставлять точную и своевременную информацию о наличии товаров, уровнях запасов и потребностях в новых закупках. Это позволяет избежать избыточного накопления товаров и одновременно предотвращает дефицит, что в конечном итоге способствует увеличению прибыли и удовлетворенности клиентов.

Современные системы контроля закупок и складского учёта интегрируют различные технологии, такие как автоматизация процессов, анализ больших данных и интернет вещей (IoT). Они позволяют магазинам одежды не только отслеживать текущие запасы, но и прогнозировать будущие потребности на основе исторических данных и трендов потребительского спроса. В результате, предприятия получают возможность более эффективно планировать свои закупки и оптимизировать работу склада, что в конечном итоге повышает конкурентоспособность на рынке.

# ПРАКТИЧЕСКАЯ РАБОТА № 1

ЗАДАНИЕ НА ПРАКТИЧЕСКУЮ РАБОТУ

Целью практической работы является анализ предметной области информационной системы «Система контроля закупок и склад. учёта магазина одежды». Задачей практической работы является описание объекта автоматизации, формулировка основных задач автоматизации объекта, описание основных параметров проектируемой информационной системы и описание путей достижения целей. Кроме того, необходимо сформулировать требования к информационной системе.

ВЫПОЛНЕНИЕ ПРАКТИЧЕСКОЙ РАБОТЫ

# Общие сведения

## Список терминов и определений

БД (База Данных) – представленная в объективной форме совокупность самостоятельных материалов, систематизированных таким образом, чтобы эти материалы могли быть найдены и обработаны с помощью электронной вычислительной машины (ЭВМ).

ИС (Информационная Система) – система, предназначенная для хранения, поиска и обработки информации, и соответствующие организационные ресурсы (человеческие, технические, финансовые и т. д.), которые обеспечивают и распространяют информацию.

СУБД (Система Управления Базами Данных) – совокупность программных и лингвистических средств общего или специального назначения, обеспечивающих управление созданием и использованием баз данных.

CSS (Cascading Style Sheets) – формальный язык описания внешнего вида документа, написанного с использованием языка разметки.

HTML (Hyper Text Markup Language) – стандартизированный язык разметки веб-страниц во Всемирной паутине.

Node.js — это кроссплатформенная среда выполнения JavaScript с открытым исходным кодом.

URL (Uniform Resource Locator) – система унифицированных адресов электронных ресурсов, или единоразный определитель местонахождения ресурса (файла).

## Описание бизнес-ролей

Неавторизованный пользователь – может зарегистрировать аккаунт.

Авторизованный пользователь – роль, носитель которой имеет возможность просматривать и искать информацию о товарах и запасах, оценивать актуальность и точность информации о запасах, делать запросы на добавление новых товаров и категорий, проходить авторизацию и аутентификацию, просматривать и изменять определенные данные на странице профиля, завершать текущую сессию.

Администратор – может настраивать и оптимизировать процессы закупок и складского учёта в системе, рассматривать и принимать решения по заявкам на добавление новых категорий товаров.

# Требования к системе

## Требования к системе в целом

### Требования к структуре и функционированию системы

Система имеет – модульную структуру, включающую в себя следующие модули:

* модуль раздела «Личный кабинет»;
* модуль раздела «Контроль закупок»;
* модуль раздела «Склад»;
* модуль раздела «Категории товаров»;
* модуль раздела «Контроль пользователей»

Система должна выполнять следующие функции:

* просмотр и поиск информации о товарах и запасах;
* контроль пользователей;
* управление категориями товаров и их характеристиками;
* хранение и защита информации о закупках и запасах;
* информирование о сбоях и изменениях в системе.

### Требования к численности и квалификации персонала системы и режиму его работы

Для обслуживания сервиса необходимо от 1 до 5 администраторов.

Для поддержания приложения и эксплуатации веб-интерфейса системы управления сайтом от сотрудников требуется наличие общих навыков работы с персональным компьютером и стандартным веб-браузером (например, MS Internet Explorer 7.0 или выше).

Режим работы сотрудников зависит от работы организации, использующей систему, за исключением работы по устранению ошибок ПО, которые были обнаружены в период экспериментальной эксплуатации в нерабочее время.

### Показатели назначения

Подсистемы, разработанные и доработанные в рамках данного раздела, обязательно должны отвечать следующим требованиям:

1. Время на полный запуск (или перезапуск) системы и компонентов системы должно составлять не более 15 минут.
2. REST API подсистемы администрирования: 50 запросов в минуту при времени отклика не более трех секунд.
3. В режиме отправки/приемки сообщений, подсистема должна поддерживать интенсивность минимум 100 запросов в секунду.
4. Система должна обеспечивать аудит операций: запись и хранение логов за последние 7 дней действий пользователей с возможностью просмотра и анализа, для повышения безопасности и отслеживания ошибок.
5. Кроссплатформенность: подсистема должна быть доступна для использования на 95% устройств конечных пользователей.
6. Система должна обеспечивать резервное копирование данных не реже одного раза в сутки, с возможностью восстановления данных за последние 7 дней.

Поддержка масштабирования: система должна быть способна масштабироваться горизонтально и вертикально для обеспечения увеличения производительности и обработки данных при увеличении нагрузки.

### Требoвания к надeжности

Программное обеспечение не должно выходить из строя более чем на 3 минуты.

Для устойчивости к потере данных необходимо регулярно производить резервное копирование хранимых данных, как описано в пункте 2.1.3.

Надежность требуемого уровня достигается путем комплексного применения организационных и организационно-технических мероприятий. При этом необходимо использовать соответствующие требованиям к программно-аппаратным средствам. В частности, можно использовать следующие базовые подходы:

* системное и базовое ПО и технические средства, соответствующие классу решаемой задачи; ­
* четкое соблюдение правил эксплуатации, а также регламентных сроков обслуживания используемых программно-аппаратных средств; ­
* допуск к обслуживанию информационной системы только пользователей, прошедших предварительное обучение.

### Требования к безопасности

Безопасность данных пользователей должна обеспечиваться шифрованием и использованием защищенных протоколов передачи данных, а также обеспечением устойчивости программно-технических средств к возможным кибератакам.

### Требования к эргономике и технической эстетике

Взаимодействие пользователей с прикладным программным обеспечением, входящим в состав системы должно осуществляться посредством визуального графического интерфейса (GUI). Интерфейс системы должен быть интуитивно понятным и удобным, не должен быть перегружен графическими элементами и должен обеспечивать быстрое отображение экранных форм.

### Требования к транспортабельности для подвижных АС

Требования к транспортабельности не предъявляются.

### Требования к эксплуатации, техническому обслуживанию, ремонту и хранению компонентов системы

Техническим обслуживанием, ремонтом и хранением сервера занимаются сетевые инженеры-техники, специалисты по серверным и сетевым технологиям, а также мастера по ремонту компьютерного и другого технического оборудования.

### Требования к защите информации от несанкционированного доступа

При работе с системой необходимо, чтобы данные могли быть восстановлены в случае их потери. Информация компании и пользователей должна была защищена от доступа или модификации несанкционированными лицами. В случае многократных попыток несанкционированного доступа, система должна производить блокировку взломщика.

### Требования по сохранности информации при авариях

Серверное программное обеспечение системы должно легко восстанавливать свое функционирование при перезапуске аппаратных средств. При аварийных ситуациях должно производиться автоматическое восстановление системы и данных. Для обеспечения сохранности информации требуется предусмотреть резервное копирование данных.

### Требования к защите от влияния внешних воздействий

Требование к защите от влияния внешних воздействий не предъявляются.

### Требования к патентной чистоте

Требования к патентной чистоте не предъявляются.

### Требования по стандартизации и унификации

Для реализации клиентской части приложения должна использоваться библиотека React. Код клиентской части приложения должен преобразовываться в набор HTML, CSS и JS файлов при сборке.

Для реализации серверной части приложения должен использоваться фреймворк в полноте обеспечивающий возможность удобной разработки приложения с учетом возможности дальнейшего масштабирования, а также библиотеки или модули, обеспечивающие удобное и быстрое взаимодействие с хранилищами данных.

Взаимодействие клиентской и серверной части должно происходить в соответствии с архитектурным стилем REST (REpresentational State Transfer).

### Дополнительные требования

Дополнительные требования не предъявляются.

## Требования к функциям (задачам), выполняемым системой

В таблице 1 представлены функциональные требования к информационной системе в виде user story.

Таблица 1 – User story

|  |  |  |  |
| --- | --- | --- | --- |
| **Кто?** | **Что хочет?** | **С какой целью?** | **Функциональное требование** |
| Неавторизованный пользователь  (Любой пользователь) | Зарегистрироваться/  Авторизоваться | Получить доступ к управлению складом и системе контроля закупок | Регистрация или авторизация на веб-сервисе |
| Авторизованный пользователь | Внести данные о прибытии товаров на склад | Обновление информации о товарах на складе | Добавление товаров |

Продолжение таблицы 1

|  |  |  |  |
| --- | --- | --- | --- |
| Авторизованный пользователь | Внести данные о продаже товара | Актуализировать информацию о товарах на складе | Удаление товаров |
| Авторизованный пользователь | Удалить собственных отзывы о различных товарах, услугах, заведениях и местах отдыха | Удаление информации оказавшейся недостоверной или устаревшей | Удаление собственных отзывов |
| Авторизованный пользователь | Оценить актуальность и точность информации о запасах | Составление рейтинга точности и актуальности данных | Оценка информации о запасах |
| Авторизованный пользователь | Просмотреть и изменить определенные данные на странице профиля | Контроль личных данных | Просмотр и изменение данных на странице профиля |
| Авторизованный пользователь | Завершить сессию на текущем устройстве в текущем браузере | Предотвращение дальнейшей возможности аутентификации на устройстве | Завершение авторизованной сессии |
| Администратор | Рассматривать и принимать решения по заявкам на добавление категорий товаров | Определение удобной структуры предоставляемых пользователю данных | Рассмотрение и принятие решений по заявкам на добавление категорий товаров |
| Администратор | Изменять значения стандартных метрик | Улучшение работы системы закупок для уникального склада | Изменение значений метрик |

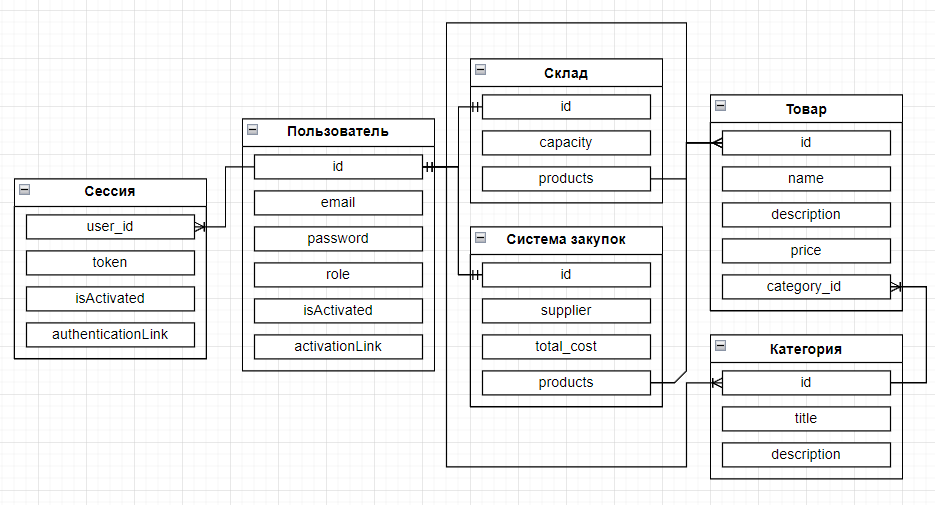
## Требования к видам обеспечения

### Требования к математическому обеспечению системы

Математическое обеспечение системы должно обеспечивать реализацию перечисленных в данном ТЗ функций, а также выполнение операций конфигурирования, программирования, управления базами данных и документирования. Алгоритмы должны быть разработаны с учетом возможности получения некорректной входной информации и предусматривать соответствующую реакцию на такие события.

### Требования к информационному обеспечению системы

Состав данных ИС должен состоять из пользователей, сессий, складов, систем закупок, товаров, категорий. На рисунке 1 представлена модель сущностей данных и связей между ними.



* 1. Модель данных

Данные, используемые системой, должны храниться в СУБД. Структура базы данных определяется с учетом особенностей внутренней модели системы принятия решений. Информационный обмен между серверной и клиентской частями системы должен осуществляться по протоколу HTTP.

### Требования к лингвистическому обеспечению системы

Интерфейс пользователя должен быть реализован на русском языке. Система ввода-вывода должна поддерживать английский и русский языки.

### Требования к программному обеспечению системы

Программное обеспечение клиентской части должно удовлетворять следующим требованиям: ­

веб-браузер: Internet Explorer 10.0 и выше, или Firefox 10.0 и выше, или Opera 12 и выше, или Safari 14 и выше, или Chrome 88 и выше;

включенная поддержка JavaScript и cookies.

### Требования к техническому обеспечению системы

Платформа, на которой будет развернута серверная часть системы, должна удовлетворять следующим минимальным требованиям: ­

* не менее 4 GB оперативной памяти; ­
* не менее 1TB дискового пространства; ­
* 4-х ядерный процессор с минимальной тактовой частотой 2ГГц.
* OC на базе Linux или ОС Windows; ­
* поддерживаемый протокол передачи данных HTTP / HTTPS, скорость передачи данных 100 Мбит/с; ­
* поддержка node.js версии 18.19.0;
* поддержка необходимых пакетов для серверной части приложения.

### Требования к метрологическому обеспечению системы

Требования к метрологическому обеспечению не предъявляются.

### Требования к организационному обеспечению системы

Требования к организационному обеспечению не предъявляются.

### Требования к методическому обеспечению системы

Необходимо разработать несколько типов руководств: ­

* руководство пользователя для администраторов сервиса;
* руководство пользователя для клиентов сервиса.

РЕЗУЛЬТАТЫ ВЫПОЛНЕНИЯ РАБОТЫ

В ходе выполнения практической работы был проведен анализ предметной области проектируемой информационной системы «Приложения книжного магазина для хранения запасов на складе». Также были сформированы требования к проектируемой информационной системе на основе ГОСТ 34.602-2020 [3].

КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Что такое требование к информационной системе?

Требование к информационной системе – это описание функциональных и нефункциональных характеристик, которые должны быть реализованы в данной системе. Они определяют то, каким должно быть поведение системы, какие данные она должна обрабатывать, какими должны быть ее характеристики производительности, надежности и безопасности.

1. Какие нормативные документы регламентируют сбор и управление требованиями к информационной системе?

ГОСТ Р 59194-2020. ГОСТ 34.602-2020.

1. Что является важным при формировании нефункциональных требований к информационной системе?

Нефункциональные требования – определяют внутренние и внешние условия или атрибуты функционирования системы. Важным для формирования нефункциональных требований к ИС являются:

* качества информационных систем: применимость, надежность, производительность, функциональная пригодность;
* ограничения: формулировка условий, выбор платформы реализации и/или развертывания, протоколы, серверы приложений, баз данных, и т. д.;
* внешние интерфейсы: UI, аппаратные интерфейсы (взаимодействие с внешними устройствами), программные интерфейсы и интерфейсы передачи информации (коммуникационные интерфейсы).

1. Какие средства существуют для формирования функциональных требований к информационной системе?

Для формирования функциональных требований к информационной системе обычно используют user-stories, текстовые предписания, диаграммы, например use case.

1. При помощи каких диаграмм можно описать функциональные требования к информационной системе?

UML(Use case, Sequence Diagrams), DFD, IDEF0

1. Для чего выполняется ранжирование требований к информационной системе?

Ранжирование требований к информационной системе выполняется для определения приоритетов в их реализации. Это позволяет оптимизировать процесс разработки системы, сосредотачивая усилия на реализации наиболее важных и критических функций.

1. Что значит связные требования к информационной системе?

Это значит, что требования связаны логически и технически, то есть их выполнение влияет на выполнение других требований. Например, изменение одного требования может потребовать изменений в других требованиях или привести к пересмотру архитектуры системы.

1. Приведите критерии, определяющие качество собранных требований к информационной системе.

Однозначность, полнота, последовательность, непротиворечивость, корректность.

# ПРАКТИЧЕСКАЯ РАБОТА № 2

ЗАДАНИЕ НА ПРАКТИЧЕСКУЮ РАБОТУ

Создать диаграмму прецедентов (use case) для одного из классов или прецедентов проектируемой информационной системы.

ВЫПОЛНЕНИЕ ПРАКТИЧЕСКОЙ РАБОТЫ

В ходе анализа предметной области были выявлены следующие действующие субъекты, взаимодействующие с проектируемой системой: неавторизованный пользователь, авторизованный пользователь администратор и модератор.

На основе представленной в прошлой работе user story построим диаграмму прецедентов, представленную на рисунке 2.1.

Для неавторизованного пользователя доступны функции:

* просмотр отзывов о различных товарах, услугах, заведениях и местах отдыха;
* возможность пройти авторизацию;
* возможность пройти регистрацию.

Для авторизованного пользователя доступны функции:

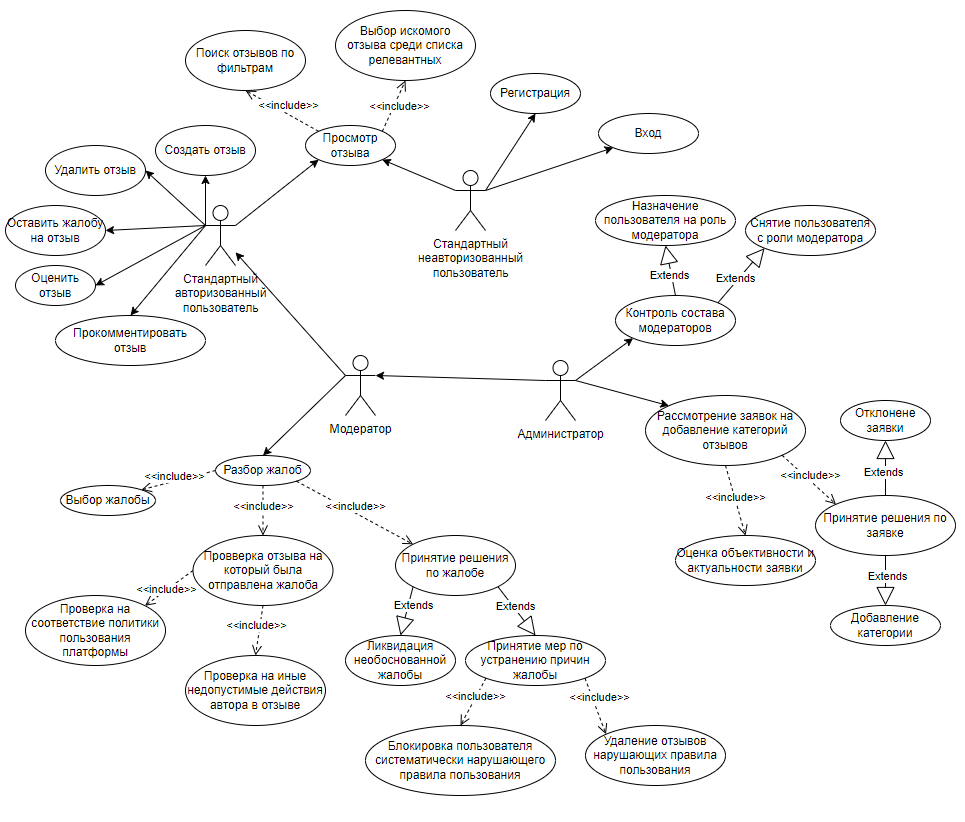
* создание и публикация отзывов о различных товарах, услугах, заведениях и местах отдыха;
* удаление собственных отзывов о различных товарах, услугах, заведениях и местах отдыха;
* возможность оставить жалобу на конкретный отзыв;
* возможность оценить отзывы о различных товарах, услугах, заведениях и местах отдыха;
* возможность прокомментировать отзывы о различных товарах, услугах, заведениях и местах отдыха;
* просмотр и изменение определенных данных на странице профиля;
* завершение сессии на текущем устройстве в текущем браузере.

Для модератора системы доступны следующие функции:

* просмотр списка жалоб и принятие решения по конкретным жалобам;
* контроль активности пользователей и принятие мер по наказанию пользователей, нарушающих правила пользования платформой.

Для администратора доступны функции:

* возможность назначения на роль / снимать с роли модератора пользователей;
* возможность рассматривать и принимать решения по заявкам на добавление категорий отзывов в фильтры.



* 1. Диаграмма прецедентов разрабатываемой информационной системы

РЕЗУЛЬТАТЫ ВЫПОЛНЕНИЯ РАБОТЫ

В ходе выполнения практической работы была построена диаграмма прецедентов проектируемой информационной системы «Агрегатор отзывов». Были выявлены действующие субъекты, созданы прецеденты и расставлены связи между субъектами и прецедентами.

КОНТРОЛЬНЫЕ ВОПРОСЫ

* + - 1. Для чего создается диаграмма прецедентов?

Диаграмма прецедентов (также use case или диаграмма вариантов использования) создается для описания общих функциональных требований к системе. Углубленное проектирование системы требует более детального описания, которое создается, в частности, с помощью диаграмм вариантов использования и в диаграммах, описывающих поток событий.

* + - 1. Что позволяет определить использование диаграммы прецедентов в процессе проектирования информационной системы?

Использование диаграммы вариантов использования в процессе проектирования информационной системы позволяет определить: пользователей и границы проектируемой информационной системы; интерфейс системы.

* + - 1. Назовите главный элемент диаграммы прецедентов.

Основными элементами диаграмм вариантов использования являются «активный субъект» или actor (участник, исполнитель, действующее лицо или действующий субъект)

* + - 1. Дайте определение, что такое «прецедент» или «вариант использования» в рамках диаграммы прецедентов UML.

Вариант использования (прецедент или use case) – графическое описание некоторого набора последовательных событий (включая варианты этих событий), выполнение системой которых приводит к тому результату, который наблюдает участник.

* + - 1. На какой вопрос должна отвечать созданная диаграмма прецедентов?

Описание потока событий должно определять, ЧТО должно быть осуществлено, а не то, КАК это должно быть осуществлено.

* + - 1. Для чего необходимы связи на диаграмме прецедентов UML, что они показывают?

Связь необходима на схеме для того, чтобы показывать взаимодействие субъекта с системой в рамках предлагаемого варианта использования. Направление связи указывает на инициатора взаимодействия.

* + - 1. Назовите два типа связей прецедентов UML.

Диаграммы UML предлагают два типа связей − включающие (inclusive) и расширяющие (extensive). Отношение включения показывает, что некоторый прецедент включает в себя сценарий другого прецедента, причем составные части обязательно входят в состав общего прецедента. Отношение расширения – выборочное отношение включения; составные элементы необязательно входят в общий вариант использования (тем самым расширяя его функционал).

Так же существует 3 тип связи – связь обобщения (generalization relationship), которая означает, что действующий субъект (вариант использования) может быть обобщен до другого субъекта (варианта использования).

# ПРАКТИЧЕСКАЯ РАБОТА №3

ЗАДАНИЕ НА ПРАКТИЧЕСКУЮ РАБОТУ

Уточнить требования, созданные в предыдущих практических работах, на основании макета информационной системы. Создать эскизный проект информационной системы в соответствии со стандартами ГОСТ 34 серии.

ВЫПОЛНЕНИЕ РАБОТЫ

ВВЕДЕНИЕ

В современном мире, где информация обладает превосходящей ценностью, сервис агрегации отзывов станет полезной обыденностью в жизни потребителей. Он выступит в качестве надежного источника информации о различных товарах, услугах, заведениях и местах отдыха.

Одной из ключевых причин востребованности подобного сервиса является его способность предоставлять объективную и непредвзятую информацию. Отзывы реальных пользователей, оставленные на подобных платформах, формируют ценный источник данных для принятия решений. Потребители все чаще обращаются к таким сервисам перед совершением покупок или выбором места для посещения, стремясь избежать негативного опыта.

Целью практической работы является анализ предметной области информационной системы «агрегатор озывов». Задачей практической работы является описание объекта автоматизации, формулировка основных задач автоматизации объекта, описание основных параметров проектируемой информационной системы и описание путей достижения целей. Кроме того, необходимо сформулировать требования к информационной системе.

# Общие сведения

## Полное наименование системы и ее условное обозначение

Наименование системы: FitFeedBack

Условное обозначение: FFB

## Номер договора

Шифр темы: АИС-FFB.

Номер контракта: №1/11-11-11-001 от 25.02.2024.

## Наименование организаций – Заказчика и Разработчика

Заказчиком системы является РТУ МИРЭА.

Адрес заказчика: Проспект Вернадского, д. 78

Разработчиком системы является ООО «NikiDan».

## Основания для разработки системы

Необходимость в автоматизации процессов бально-рейтингового оценивания различных товаров, услуг, заведений, мест отдыха и т.д. с возможностью написания развернутого комментария к оценке, а также получения данных об опыте использования, посещения и т.п. тех или иных товаров, услуг, заведений, мест отдыха и т.д.

## Плановые сроки начала и окончания работы по созданию системы

Плановый срок начала работ по созданию системы – 12 февраля 2024 года.

Плановый срок окончания работ по созданию системы – 4 июня 2024 года.

## Источники и порядок финансирования работ

Собственные средства разработчика.

## Порядок оформления и предъявления заказчику результатов работ по созданию системы

Результаты работ передаются Заказчику в порядке, определенном контрактом в соответствии с Календарным планом работ контракта на основании Актов сдачи-приемки выполненных работ (этапа работ).

Документация FFB передается на бумажных (два экземпляра, один экземпляр после подписания Заказчиком должен быть возвращен Исполнителю) и на машинных носителях (DVD) (в двух экземплярах). Текстовые документы, передаваемые на машинных носителях, должны быть представлены в форматах PDF.

Все материалы передаются с сопроводительными документами Исполнителя.

## Перечень нормативно-технических документов, методических материалов, использованных при разработке ТЗ

При разработке автоматизированной системы и создании проектно-эксплуатационной документации Исполнитель должен руководствоваться требованиями следующих нормативных документов:

* ГОСТ 19.106-78. Единая система программной документации. Требования к программным документам, выполненным печатным способом.
* ГОСТ 34.602 – 2020 Техническое задание на создание автоматизированной системы
* ГОСТ Р 59793-2021. Информационные технологии. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Стадии создания.
* ГОСТ 34.201–2020. Информационные технологии. Комплекс стандартов на автоматизированные системы. Виды, комплектность и обозначение документов при создании автоматизированных систем.
* ГОСТ Р 59795-2021. Информационные технологии. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Требования к содержанию документов.

## Определения, обозначения и сокращения

БД (База Данных) – представленная в объективной форме совокупность самостоятельных материалов, систематизированных таким образом, чтобы эти материалы могли быть найдены и обработаны с помощью электронной вычислительной машины (ЭВМ).

ИС (Информационная Система) – система, предназначенная для хранения, поиска и обработки информации, и соответствующие организационные ресурсы (человеческие, технические, финансовые и т. д.), которые обеспечивают и распространяют информацию.

СУБД (Система Управления Базами Данных) – совокупность программных и лингвистических средств общего или специального назначения, обеспечивающих управление созданием и использованием баз данных.

CSS (Cascading Style Sheets) – формальный язык описания внешнего вида документа, написанного с использованием языка разметки.

HTML (Hyper Text Markup Language) – стандартизированный язык разметки веб-страниц во Всемирной паутине.

Node.js — это кроссплатформенная среда выполнения JavaScript с открытым исходным кодом.

URL (Uniform Resource Locator) – система унифицированных адресов электронных ресурсов, или единоразный определитель местонахождения ресурса (файла).

## Описание бизнес-ролей

Неавторизованный пользователь – имеет доступ к просмотру и поиску существующих отзывов, регистрации.

Стандартный авторизованный пользователь – роль, носитель которой имеет возможность просматривать, искать, создавать и публиковать отзывы, а также удалять лично опубликованные, оценивать достоверность, полезность и прочие характеристики уже опубликованных отзывов, составлять жалобы, публиковать комментарии к отзывам, делать запрос на открытие категории на сервисе, проходить авторизацию и аутентификацию, завершать текущую сессию, просматривать и изменять определенные данные на странице профиля.

Модератор – имеет возможности стандартного пользователя и вдобавок к ним просмотр списка жалоб, принятие решений по жалобам, контроль активности пользователей и принятие мер по наказанию (блокировка временная/постоянная) пользователей, нарушающих правила пользования сервисом.

Администратор – обладает доступом к функционалу модератора, а также имеет возможность предоставления пользователям роли модератора, снятия пользователей с роли модератора, рассмотрения и принятия решений по заявкам на добавление категорий отзывов.

# Назначение и цели создания(развития) системы

## Назначение системы

Назначением системы является с одной стороны публикация отзывов о различных товарах, услугах, заведениях, местах отдыха и т.д., и с другой стороны получение информации о пользовательском опыте.

## Цели создания системы

Основными целями создания ИС являются:

* автоматизация процессов публикации и распространения информации о пользовательском опыте;
* оперативный доступ к опубликованным отзывам;
* определения рейтинга для объектов пользовательского опыта на основе усреднения всех пользовательских оценок объекта.

# Характеристика объекта автоматизации

## Краткие сведения об объекте автоматизации

Объектом автоматизации является публикация и просмотр отзывов.

## Сведения об условиях эксплуатации объекта автоматизации

Операционная система: Сервис должен быть совместим с различными операционными системами, такими как Windows, MacOS, Linux и другие.

Браузер: Сервис должен быть доступен и корректно отображаться в различных браузерах, таких как Google Chrome, Mozilla Firefox, Safari, Microsoft Edge и других.

Интернет-соединение: Для использования сервиса необходимо стабильное и быстрое интернет-соединение.

Разрешение экрана: Сервис должен корректно отображаться на различных разрешениях экрана, включая мобильные устройства.

Авторизация: Для использования сервиса может потребоваться авторизация пользователя, например, через учетную запись в социальных сетях или электронную почту.

Безопасность: Сервис должен обеспечивать безопасность данных пользователей, включая защиту от несанкционированного доступа, хакерских атак и утечек информации.

Обновления: Сервис должен регулярно обновляться и улучшаться, чтобы соответствовать современным требованиям и потребностям пользователей.

Техническая поддержка: Сервис должен предоставлять техническую поддержку пользователям, чтобы помочь им решить возникающие проблемы и ответить на вопросы.

# Требования к системе

## Требования к системе в целом

### Требования к структуре и функционированию системы

Система имеет – модульную структуру, включающую в себя следующие модули:

* модуль раздела «Отзывы»;
* модуль раздела «Личный кабинет»;
* модуль раздела «Заявки»;
* модуль раздела «Жалобы»;
* модуль раздела «Контроль пользователей»

Система должна выполнять следующие функции:

* просмотр поиск отзывов;
* публикация комментариев и оценка отзывов;
* контроль пользователей;
* управление отзывами и категориями;
* хранение информации;
* информирование о сбоях.

### Требования к численности и квалификации персонала системы и режиму его работы

Для обслуживания сервиса необходимо от 1 до 3 администраторов и от 1 до 10 модераторов. Количество модераторов должно зависеть от пользовательской нагрузки на сервис. Примерная градация: 1 администратор и 1 модератор – 100 пользователей, 2 администратора и 5 модераторов – 1000 пользователей, 3 администратора и 10 модераторов для более 1000 пользователей.

Для поддержания приложения и эксплуатации веб-интерфейса системы управления сайтом от сотрудников требуется наличие общих навыков работы с персональным компьютером и стандартным веб-браузером (например, MS Internet Explorer 7.0 или выше).

Режим работы сотрудников зависит от работы организации, использующей систему, за исключением работы по устранению ошибок ПО, которые были обнаружены в период экспериментальной эксплуатации в нерабочее время.

### Показатели назначения

Подсистемы, разработанные и доработанные в рамках данного раздела, обязательно должны отвечать следующим требованиям:

Время на полный запуск (или перезапуск) системы и компонентов системы должно составлять не более 15 минут.

REST API подсистемы администрирования: 50 запросов в минуту при времени отклика не более трех секунд.

В режиме отправки/приемки сообщений, подсистема должна поддерживать интенсивность минимум 100 запросов в секунду.

Система должна обеспечивать аудит операций: запись и хранение логов за последние 7 дней действий пользователей с возможностью просмотра и анализа для повышения безопасности и отслеживания ошибок.

Кроссплатформенность: подсистема должна быть доступна для использования на 95% устройств конечных пользователей.

Система должна обеспечивать резервное копирование данных не реже одного раза в сутки, с возможностью восстановления данных за последние 7 дней.

Поддержка масштабирования: система должна быть способна масштабироваться горизонтально и вертикально для обеспечения увеличения производительности и обработки данных при увеличении нагрузки.

### Требoвания к надeжности

Программное обеспечение не должно выходить из строя более чем на 3 минуты.

Для устойчивости к потере данных необходимо регулярно производить резервное копирование хранимых данных, как описано в пункте 4.1.3.

Надежность требуемого уровня достигается путем комплексного применения организационных и организационно-технических мероприятий. При этом необходимо использовать соответствующие требованиям к программно-аппаратным средствам. В частности, можно использовать следующие базовые подходы:

* системное и базовое ПО и технические средства, соответствующие классу решаемой задачи; ­
* четкое соблюдение правил эксплуатации, а также регламентных сроков обслуживания используемых программно-аппаратных средств; ­
* допуск к обслуживанию информационной системы только пользователей, прошедших предварительное обучение.

### Требования к безопасности

Безопасность данных пользователей должна обеспечиваться шифрованием с помощью хэш-функций (SHA-256) и использованием защищенных протоколов передачи данных (HTTPS), а также обеспечением устойчивости программно-технических средств к возможным кибератакам, таким как DoS и DDoS атаки и SQL-инъекции.

### Требования к эргономике и технической эстетике

Взаимодействие пользователей с прикладным программным обеспечением, входящим в состав системы должно осуществляться посредством визуального графического интерфейса (GUI). Интерфейс системы должен быть интуитивно понятным и удобным, не должен быть перегружен графическими элементами и должен обеспечивать быстрое отображение экранных форм.

### Требования к транспортабельности для подвижных АС

Требования к транспортабельности не предъявляются.

### Требования к эксплуатации, техническому обслуживанию, ремонту и хранению компонентов системы

Техническим обслуживанием, ремонтом и хранением сервера занимаются сетевые инженеры-техники, специалисты по серверным и сетевым технологиям, а также мастера по ремонту компьютерного и другого технического оборудования.

### Требования к защите информации от несанкционированного доступа

При работе с системой необходимо, чтобы данные могли быть восстановлены в случае их потери. Информация компании и пользователей должна была защищена от доступа или модификации несанкционированными лицами. В случае многократных попыток несанкционированного доступа, система должна производить блокировку взломщика.

### Требования по сохранности информации при авариях

Серверное программное обеспечение системы должно легко восстанавливать свое функционирование при перезапуске аппаратных средств. При аварийных ситуациях должно производиться автоматическое восстановление системы и данных. Для обеспечения сохранности информации требуется предусмотреть резервное копирование данных.

### Требования к защите от влияния внешних воздействий

Требование к защите от влияния внешних воздействий не предъявляются.

### Требования к патентной чистоте

Требования к патентной чистоте не предъявляются.

### Требования по стандартизации и унификации

Для реализации клиентской части приложения должна использоваться библиотека React. Код клиентской части приложения должен преобразовываться в набор HTML, CSS и JS файлов при сборке.

Для реализации серверной части приложения должен использоваться фреймворк в полноте обеспечивающий возможность удобной разработки приложения с учетом возможности дальнейшего масштабирования, а также библиотеки или модули, обеспечивающие удобное и быстрое взаимодействие с хранилищами данных.

Взаимодействие клиентской и серверной части должно происходить в соответствии с архитектурным стилем REST (REpresentational State Transfer).

### Дополнительные требования

Дополнительные требования не предъявляются.

## Требования к функциям (задачам), выполняемым системой

В таблице 1 представлены функциональные требования к информационной системе в виде user story.

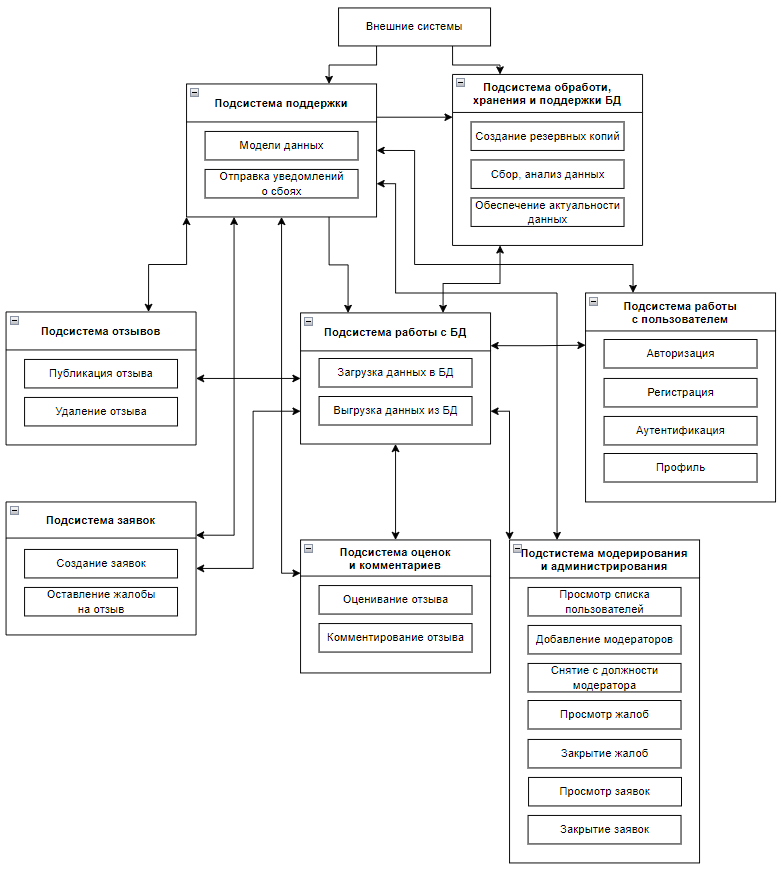
Таблица 1 – User story

|  |  |  |  |
| --- | --- | --- | --- |
| **Кто?** | **Что хочет?** | **С какой целью?** | **Функциональное требование** |
| Неавторизованный пользователь  (Любой пользователь) | Найти отзывы о различных товарах, услугах, заведениях и местах отдыха | Получить достоверную информацию об интересующем объекте для предотвращения негативного опыта | Просмотр и поиск отзывов |
| Неавторизованный пользователь  (Любой пользователь) | Зарегестироваться/  Авторизоваться | Получение доступа к большему функционалу сервиса | Регистрация или авторизация на веб-сервисе |
| Стандартный авторизованный пользователь | Создать и опубликовать отзывы о различных товарах, услугах, заведениях и местах отдыха | Публикация достоверной информации о полученном опыте | Публикация отзывов |
| Стандартный авторизованный пользователь | Удалить собственные отзывы о различных товарах, услугах, заведениях и местах отдыха | Удаление информации оказавшейся недостоверной или устаревшей | Удаление собственных отзывов |
| Стандартный авторизованный пользователь | Оставить жалобу на конкретный отзыв | Пресечение действий нарушающих политику пользования платформой | Оставление жалобы на конкретный отзыв |
| Стандартный авторизованный пользователь | Оценить отзывы о различных товарах, услугах, заведениях и местах отдыха | Составление рейтинга достоверности автора отзыва и самого отзыва | Оценка отзывов |

Продолжение таблицы 1

|  |  |  |  |
| --- | --- | --- | --- |
| Стандартный авторизованный пользователь | Прокомментировать отзывы о различных товарах, услугах, заведениях и местах отдыха | Развернутое текстовое дополнение оценки отзыва | Комментирование отзывов |
| Стандартный авторизованный пользователь | Просмотреть и изменить определенные данные на странице профиля | Контроль личных данных | Просмотр и измененение определенные данные на странице профиля |
| Стандартный авторизованный пользователь | Завершить сессию на текущем устройстве в текущем браузере | Предотвращение дальнейшей возможности аутентификации пользователя на устройстве в текущем браузере | Завершение авторизованной сессии |
| Модератор | Просмотреть список жалоб и принять решение по конкретным жалобам | Поддержание порядка и соблюдения правил пользования платформой | Просмотр списка жалоб и принятие решений по конкретным жалобам |
| Модератор | Контролировать активность пользователей и принимать меры по наказанию пользователей, нарушающих правила пользования платформой | Пресечение систематического нарушения прав пользования платформой | Контроль активности пользователей и принятие мер по наказанию пользователей, нарушающих правила пользования платформой |
| Администратор | Назначать на роль / снимать с роли модератора пользователей | Определение рабочего персонала сервиса | Назначение на роль / снятие с роли модератора |
| Администратор | Рассматривать и принимать решения по заявкам на добавление категорий отзывов в фильтры | Определение удобной структуры предоставляемых пользователю данных | Рассмотрение и принятие решений по заявкам на добавление категорий отзывов |

## Функциональная структура системы



* 1. Функциональная структура системы

Связи между подсистемами отзывов, заявок, оценок и комментариев, работы с пользователями и поддержки определяет процесс добавления и изменения данных в БД и вывода данных из БД.

Связи между подсистемой поддержи и другими подсистемами определяет возможность отправлять уведомления о сбоях и ошибках.

Связь «Подсистема работы с БД – Подсистема обработки, хранения и поддержки БД» определяет работу с данными в БД на уровне сервера.

Связь «Подсистема поддержки – Подсистема работы с БД» определяет процесс передачи уведомлений о сбоях в БД и дальнейшее устранение неполадок администратором.

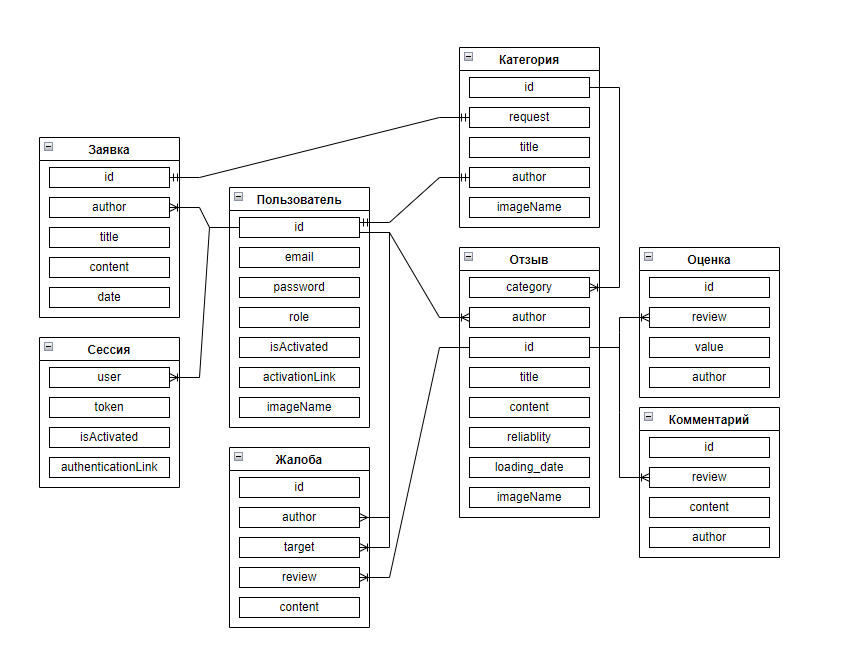
## Требования к видам обеспечения

### Требования к математическому обеспечению системы

Математическое обеспечение системы должно обеспечивать реализацию перечисленных в данном ТЗ функций, а также выполнение операций конфигурирования, программирования, управления базами данных и документирования. Алгоритмы должны быть разработаны с учетом возможности получения некорректной входной информации и предусматривать соответствующую реакцию на такие события.

### Требования к информационному обеспечению системы

Состав данных ИС должен состоять из пользователей, сессий, категорий, отзывов, оценок, комментариев, жалоб, заявок. На рисунке 3.2 представлена модель сущностей данных и связей между ними.



* 1. Модель данных

Данные, используемые системой, должны храниться в СУБД. Структура базы данных определяется с учетом особенностей внутренней модели системы принятия решений. Информационный обмен между серверной и клиентской частями системы должен осуществляться по протоколу HTTP.

### Требования к лингвистическому обеспечению системы

Интерфейс пользователя должен быть реализован на русском языке. Система ввода-вывода должна поддерживать английский и русский языки.

### Требования к программному обеспечению системы

Программное обеспечение клиентской части должно удовлетворять следующим требованиям: ­

веб-браузер: MS Edge 100 и выше, или Firefox 10.0 и выше, или Opera 12 и выше, или Safari 14 и выше, или Chrome 88 и выше;

включенная поддержка JavaScript и cookies.

### Требования к техническому обеспечению системы

Платформа, на которой будет развернута серверная часть системы, должна удовлетворять следующим минимальным требованиям: ­

* не менее 4 GB оперативной памяти; ­
* не менее 1TB дискового пространства; ­
* 4-х ядерный процессор с минимальной тактовой частотой 2ГГц.
* OC на базе Linux или ОС Windows; ­
* поддерживаемый протокол передачи данных HTTP / HTTPS, скорость передачи данных 100 Мбит/с; ­
* поддержка node.js версии 18.19.0;
* поддержка необходимых пакетов для серверной части приложения.

### Требования к метрологическому обеспечению системы

Требования к метрологическому обеспечению не предъявляются.

### Требования к организационному обеспечению системы

Требования к организационному обеспечению не предъявляются.

### Требования к методическому обеспечению системы

Необходимо разработать несколько типов руководств:

* руководство пользователя для администраторов сервиса;
* руководство пользователя для модераторов;
* руководство пользователя для клиентов сервиса.

# Состав и содержание работ по созданию (развитию) системы

Разработка системы предполагается по укрупненному календарному плану, приведенному в таблице 5.1.

Таблица 5.1 – Календарный план работ по созданию АС FFB

|  |  |  |
| --- | --- | --- |
| **Этапы работ** | **Содержание работ** | **Сроки** |
| 1. Исследование и обоснование АС | 1.1. Исследование (сбор и анализ данных)  автоматизированного объекта, включая сбор  сведений о зарубежных и отечественных  аналогах | 12.02.2024-19.02.2024 |
| 2.Составление  технического задания | 2.1. Разработка функциональных и нефункциональных требований к системе | 19.02.2024- 26.02.2024 |
| 3. Эскизное  проектирование | 3.1. Разработка предварительных решений по выбранному варианту АС и отдельным видам обеспечения | 26.02.2024- 04.03.2024 |
| 4. Техническое  проектирование | 4.1. Разработка диаграмм | 04.03.2024- 11.03.2024 |
| 4.2. Разработка макетов интерфейса | 11.03.2024- 18.03.2024 |
| 5. Разработка программной части | 5. Разработка серверной и клиентской части | 18.03.2024-01.04.2024 |
| 6. Предварительные комплексные испытания | 6.1. Проверка работоспособности системы в условиях, приближенных к реальным | 01.04.2024- 08.04.2024 |
| 7. Опытная эксплуатация | 7.1. Эксплуатация с привлечением небольшого количества участников | 08.04.2024- 15.04.2024 |
| 7.2. Устранение замечаний, выявленных при эксплуатации, АС | 15.04.2024- 22.04.2024 |
| 8. Ввод в промышленную эксплуатацию | 8.1. Приемка АС в промышленную эксплуатацию (внедрение АС) | 22.04.2024- 25.05.2024 |

# Порядок контроля и приемки системы

В соответствии с разделом 5 необходимо на каждой стадии создания системы установить контроль и приемку результатов работ.

На стадии 5 происходит прием готовой версии программного продукта (модели), а остальные результаты работ представляются в виде документов согласно таблице 5.1.

Приемка этапа включает в себя рассмотрение и оценку объема работ и предоставленной технической документации в соответствии с требованиями технического задания.

Организацию и проведение приемки системы должен осуществлять заказчик, а приемка системы должна производиться только после того, как будут выполнены все задачи системы.

Заказчик обязан предоставить материальную часть (технические средства), проектную документацию и специально выделенный персонал.

Последним этапом при приемке системы является составление акта приемки.

# Требования к составу и содержанию работ по подготовке объекта автоматизации к вводу системы в действие

Для обеспечения готовности объекта к вводу системы в действие провести комплекс мероприятий:

* приобрести компоненты программного обеспечения, заключить договора на их лицензионное использование;
* завершить работы по установке технических средств;
* провести диагностику устойчивости сети к нагрузкам;
* провести обучение сотрудников.

## Приведение поступающей в систему информации к виду, пригодному для обработки с помощью ЭВМ

Информация вводится пользователем в разработанные экранные формы компонентов системы.

## Изменения, которые необходимо осуществить в объекте автоматизации

Изменений не требуется.

## Создание условий функционирования объекта автоматизации, при которых гарантируется соответствие создаваемой системы требованиям, содержащимся в ТЗ

Для функционирования создаваемой системы требуется платформа, технические характеристики которой соответствуют предъявленным.

## Создание необходимых для функционирования системы подразделений и служб

Для функционирования системы не требуется дополнительных подразделений и служб.

## Сроки и порядок комплектования штатов и обучения персонала

Комплектование штатов подразделений и служб, необходимых для функционирования системы, а также подготовка их сотрудников должны быть завершены до начала опытной эксплуатации системы.

# Требования к документированию

Проектная документация должна быть разработана в соответствии с ГОСТ 34.201-2020 и ГОСТ 7.32-2017.

Отчетные материалы должны включать в себя текстовые материалы (представленные в виде бумажной копии и на цифровом носителе в формате MS Word) и графические материалы.

Предоставить документы:

1) схема функциональной структуры автоматизируемой деятельности;

2) описание технологического процесса обработки данных;

3) описание информационного обеспечения;

4) описание программного обеспечения АС;

5) схема логической структуры БД;

6) руководство пользователя;

7) описание контрольного примера (по ГОСТ 24.102);

8) протокол испытаний (по ГОСТ 24.102).

# Источники разработки

* ГОСТ 34.602-2020. Информационные технологии. Комплекс стандартов на автоматизированные системы. Техническое задание на создание автоматизированной системы.
* ГОСТ Р 59793-2021. Информационные технологии. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Стадии создания.
* ГОСТ 34.201-2020. Информационные технологии. Комплекс стандартов на автоматизированные системы. Виды, комплектность и обозначение документов при создании автоматизированных систем.
* ГОСТ Р 59795-2021. Информационные технологии. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Требования к содержанию документов.
* ГОСТ 19.106-78. Единая система программной документации. Требования к программным документам, выполненным печатным способом.
* ГОСТ 19.105-78. Единая система программной документации. Общие требования к программным документам.

РЕЗУЛЬТАТЫ ПРАКТИЧЕСКОЙ РАБОТЫ

В результате выполнения практической работы были уточнены требования и разработан эскизный макет информационной системы.

ОТВЕТЫ НА КОНТРОЛЬНЫЕ ВОПРОСЫ

1. В чем состоит цель эскизного проектирования информационной системы?

- Цель эскизного проектирования - создать высокоуровневое представление о структуре и функциональности будущей информационной системы, определить основные компоненты и взаимосвязи между ними.

2. Для чего создаётся макет информационной системы?

- Макет информационной системы создается для визуализации и оценки общей структуры и распределения компонентов системы. Он позволяет лучше понять, как будут взаимодействовать различные части системы.

3. Дайте определение термина «Макетирование информационной системы»

- Макетирование информационной системы - это процесс создания визуальных представлений (макетов) для иллюстрации общей структуры и компонентов системы

4. Назовите основные особенности, на которые необходимо обратить внимание при создании макета информационной системы.

- Основные особенности макета: четкость структуры, взаимодействие компонентов, удовлетворение требований заказчика, учет потребностей пользователей, простота восприятия.

5. Какие нотации можно использовать для создания макета информационной системы?

- Для создания макета информационной системы можно использовать нотации, такие как UML (Unified Modeling Language), блок-схемы, диаграммы потоков данных (DFD) и другие.

6. Какие формы создаются в процессе макетирования информационной системы?

- В процессе макетирования создаются графические формы, такие как диаграммы, блок-схемы, макеты интерфейсов пользователя, диаграммы потоков данных и другие.

7. Приведите структуру и дайте описание процесса макетирования информационной системы.

- Процесс макетирования включает в себя определение требований, создание общей структуры системы, разработку макетов компонентов, оценку взаимосвязей, а также создание визуальных представлений системы.

8. Приведите последовательность действий при макетировании программного обеспечения.

- а. Определение требований.

- б. Создание общей структуры системы.

- в. Разработка макетов компонентов.

- г. Оценка взаимосвязей.

- д. Создание визуальных представлений.

9. Приведите достоинства и недостатки итеративного макетирования.

- Достоинства: Гибкость, возможность коррекции ошибок на ранних этапах, лучшее взаимодействие с заказчиком.

Недостатки: может потребовать больше времени и ресурсов.

10. Опишите процесс создания и оформления пояснительной записки эскизного проекта информационной системы.

- Создание пояснительной записки включает в себя описание общей структуры системы, функциональности, взаимодействия компонентов, требований к системе и интерфейсам.

11. Какие нормативные документы регламентируют процесс создания и оформления эскизного проекта?

- Нормативные документы могут включать в себя стандарты UML, методологии разработки программного обеспечения, а также требования заказчика.

12. Какие действия должны быть выполнены, чтобы пояснительная записка эскизного проекта информационной системы стала программным документом?

- Пояснительная записка становится программным документом после ее утверждения и внесения соответствующих изменений в проект.

13. Какие нотации можно использовать для описания в графическом виде функциональной модели информационной системы?

- Для описания функциональной модели информационной системы можно использовать нотации, такие как UML

ПРАКТИЧЕСКАЯ РАБОТА №4

ЗАДАНИЕ НА ПРАКТИЧЕСКУЮ РАБОТУ

Целью данной практической работы является выбор и проектирование функциональной модели информационной системы в нотации IDEF0, составление краткого описания ИС, включая цель, способ и средства её создания. В рамках данной работы выполняется моделирование диаграммы контекстного уровня A-0. Работа должна быть выполнена с соблюдением требований руководящего документа РД IDEF0 – 2000 Методология функционального моделирования IDEF0.

ВЫПОЛНЕНИЕ ПРАКТИЧЕСКОЙ РАБОТЫ

Введение

Для проектирования была выбрана информационная система для сбора отзывов пользователей. Система создается для сбора пользовательского опыта в пользовании различными результатами деятельности общества.

Цель создания

Целью создания ИС является:

* агрегация отзывов на товары и услуги;
* создание и публикация отзывов о различных товарах;
* сбор статистики и оценок пользователей.

По определению ИС: «Информационная система – это сложный программный комплекс, который способен собирать, сохранять, обрабатывать и выдавать по запросу пользователя информацию». Проектируемая ИС полностью удовлетворяет всему перечню требований, указанных в определении, т.к.:

1. Сайт собирает оценки, отзывы и статистику, полученные от пользователей.

2. Хранит полученную информацию в базе данных.

3. Информация из подпунктов выше обрабатывается, на основе чего при помощи специальных алгоритмов пользователь при каждом последующем поиске получает более релевантный ответ.

4. Доступ пользователей к информации на сайте.

Краткое описание

ИС представлена в виде сайта. Сайт является удобным интернет-сервисом, позволяющим удобно пользоваться представленным функционалом работы с отзывами. Для комфортного и круглосуточного доступа сайт также адаптирован для мобильных устройств и представлен на русском и английском языках.

Одно из важных достоинств проектируемой ИС – функционал не только пользователей, но и для модераторов. Модераторы могут просматривать информацию об отзывах и проводить их проверку.

Способ создания ИС

В качестве способа определения требований была выбрана методология «последовательных приближений», которая основана на том, что все расчеты и графические построения, связанные с определением основных элементов, разбиваются на несколько более мелких элементов, в которых происходит их уточнение. Данный метод также хорошо сочетается с нотацией IDEF0, которая основана на декомпозиции каждого блока на более мелких с уточнением деталей.

Средства создания ИС

В качестве средств создания ИС были использованы языки программирования JavaScript, HTML, CSS, СУБД PostgreSQL, node.js для реализации серверной части приложения и сервис для развертывания сервера Apache HTTP Server. Для моделирования проектируемой ИС будет использоваться нотация IDEF0 в CASE-средстве Ramus Educational.

Проектирование контекстной диаграммы функциональной модели ИС

Была спроектирована контекстная диаграмма A-0 в нотации IDEF0. В качестве управления были выбраны следующие нормативные и правовые документы:

1. Законодательство;

2. Политика организации.

В качестве входящих информационных потоков, которые подлежат обработке и преобразованию в процессе работы ИС, были указаны:

1. Данные для входа;

2. Запрос.

В качестве механизмов (ресурсов, выполняющих работу) были выделены:

1. Пользователи;

2. Приложение.

В качестве выходов получены следующие информационные элементы:

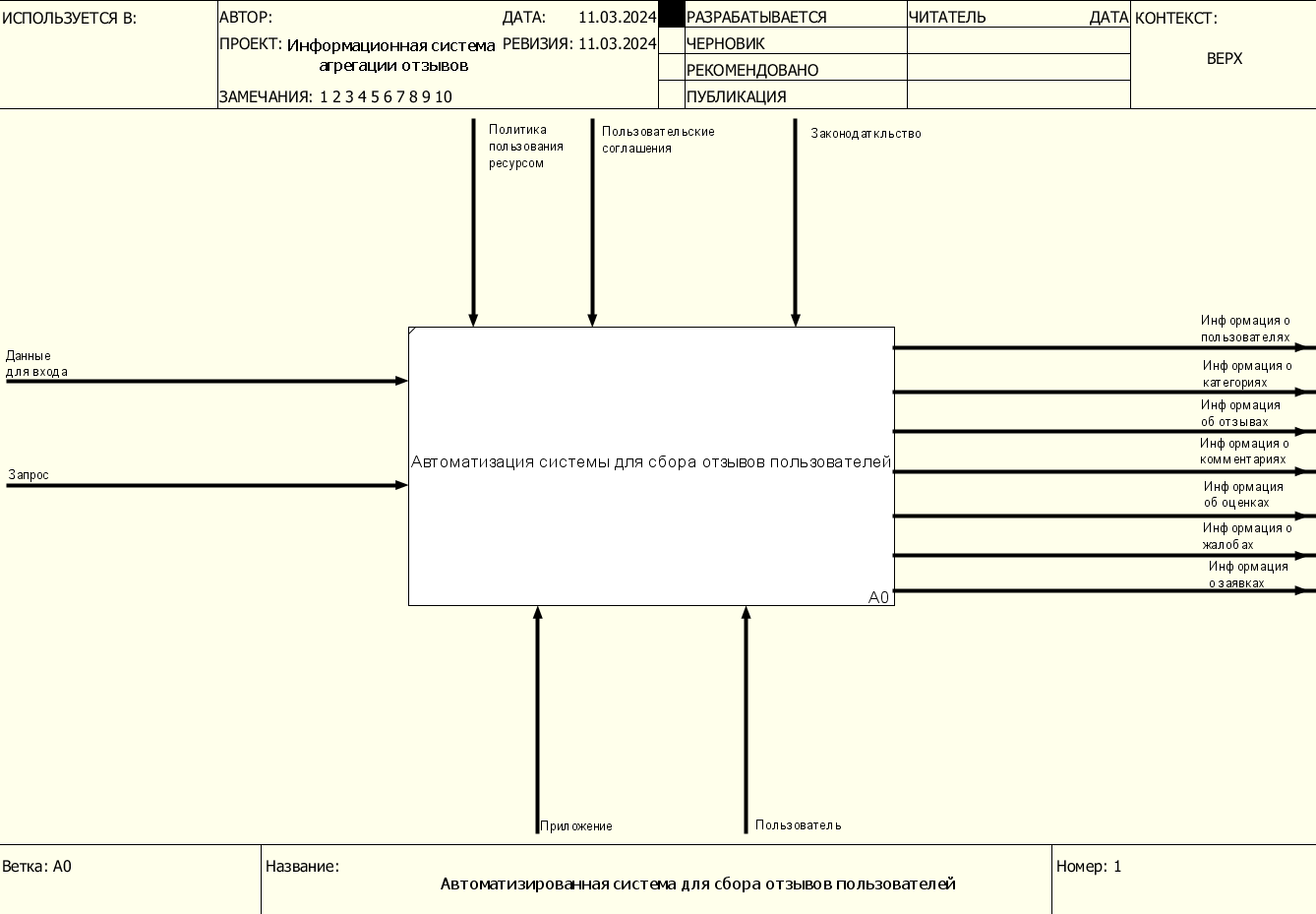
1. Информация о отзывах.

2. Информация о категорях;

3. Информация о оценках;

4. Информация о пользователях.

На рисунке 4.1 представлена контекстная диаграмма проектируемой информационной систем



* 1. Контекстная диаграмма

РЕЗУЛЬТАТЫ ПРАКТИЧЕСКОЙ РАБОТЫ

В результате выполнения данной практической работы определена цель, способ и средства создания ИС, составлено краткое описание, а также смоделирована контекстная диаграмма A-0 в нотации IDEF0.

ПРАКТИЧЕСКАЯ РАБОТА №5

ЗАДАНИЕ НА ПРАКТИЧЕСКУЮ РАБОТУ

Декомпозировать функциональную модель проектируемой системы в нотации IDEF0, создать декомпозиции как минимум двух уровней. Допустимо декомпозировать один, наиболее значимый для проектируемой системы, функциональный блок на каждом уровне. Добавить описание функциональных блоков и потоков данных, а также выводы к работе.

ВЫПОЛНЕНИЕ ПРАКТИЧЕСКОЙ РАБОТЫ

При декомпозиции контекстной диаграммы «Автоматизация системы агрегации отзывов» были спроектированы следующие функциональные блоки:

1. Вход в систему (А1);

2. Операции с категориями отзывов (А2);

3. Операции с отзывами (А3);

4. Операции с пользовательской обратной связью (А4);

5. Операции с заявками на модерирование (А5).

Все процессе проходят на основе законодательства, пользовательского соглашения и политики пользования ресурсом.

Функциональный блок «Вход в систему». В этом процессе происходит авторизация пользователя и получение данных о нём. На вход поступают данные для входа и запрос. Процесс выполняется с помощью приложения и пользователя. В результате отработки процесса получится информация о пользователе, которая потребуются во всех следующих процессах для подтверждения уровня доступа пользователя.

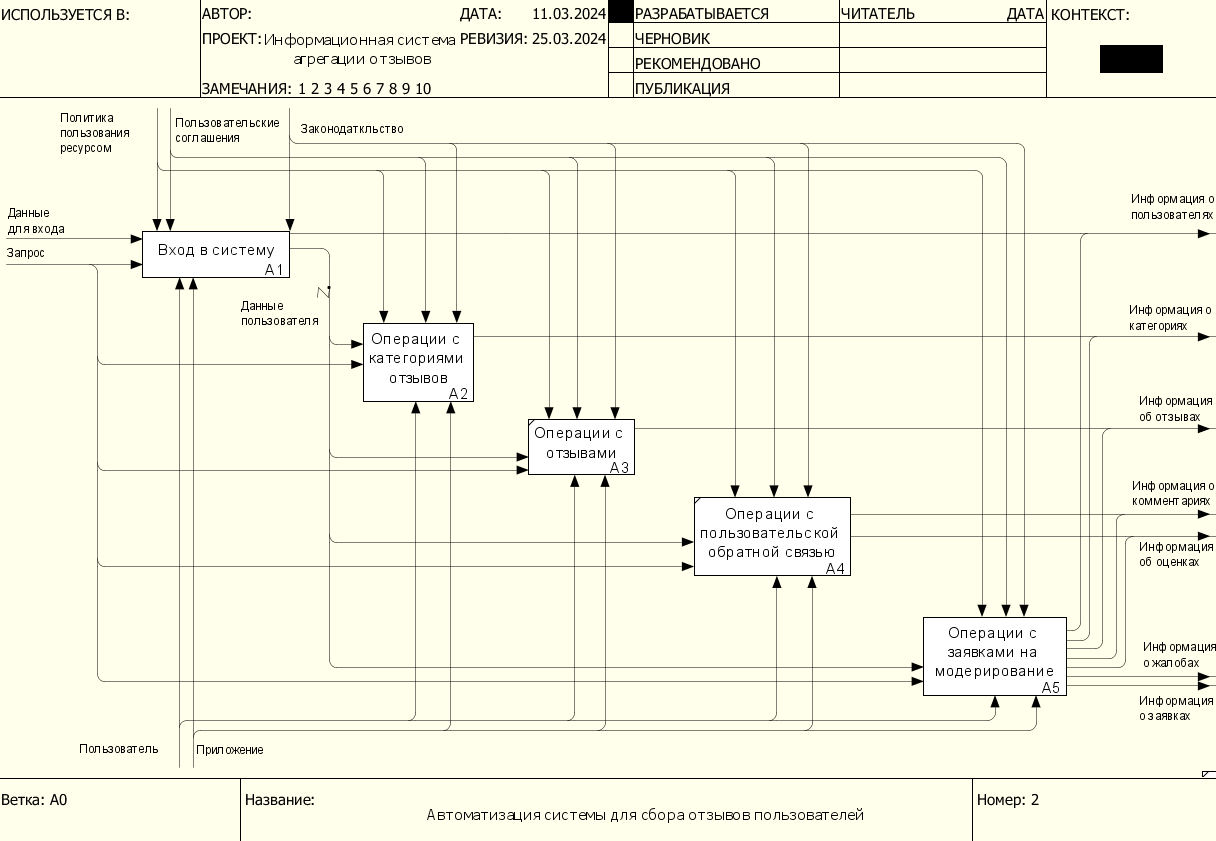
Функциональный блок «Операции с категориями отзывов». В этом процессе происходит работа с категориями в приложении: получение информации об категориях, ручное добавление категорий и т.п. На вход поступают информация о пользователе и запрос. Процесс выполняется с помощью приложения и пользователя. На выходе получается информация о категориях.

Функциональный блок «Работа с отзывами». В этом процессе происходит управление отзывами: получение информации об отзывах, добавление информации о новых отзывах. На вход поступает запрос и информация о пользователе. Процесс выполняется с помощью приложение и пользователя. На выходе получается информация об отзывах.

Функциональный блок «Операции с пользовательской обратной связью». В этом процессе происходит получение информации о пользовательских комментариях, создание, выполнение и подтверждение обратной связи, и сбор статистики. На вход поступают информация о пользователе, запрос и информация о объекте обратной связи. Процесс выполняется с помощью пользователя и приложения. На выходе получается информация о комментариях и информация об оценках.

Функциональный блок «Операции с заявками на модерирование». В этом процессе происходит получение информации о пользовательских комментариях и жалобах. На вход поступают информация о пользователе, запрос и информация о объекте обратной связи. Процесс выполняется с помощью пользователя и приложения. На выходе получается информация о жалобах, пользователях, оценках, отзывах, комментариях.

На рисунке 5.1 представлена описанная выше контекстная диаграмма процесса.



* 1. Декомпозиция контекстной диаграммы

Далее произведем декомпозицию функционального блока «Операции с заявками на модерирование». В результате получены следующие функциональные блоки:

1. Отправка заявки на добавление категории (А51);

2. Принятие решения по заявке на добавление категории (А52);

3. Отправка жалобы на действия иного пользователя (А53).

4. Устранение жалобы на действия пользователя (А54).

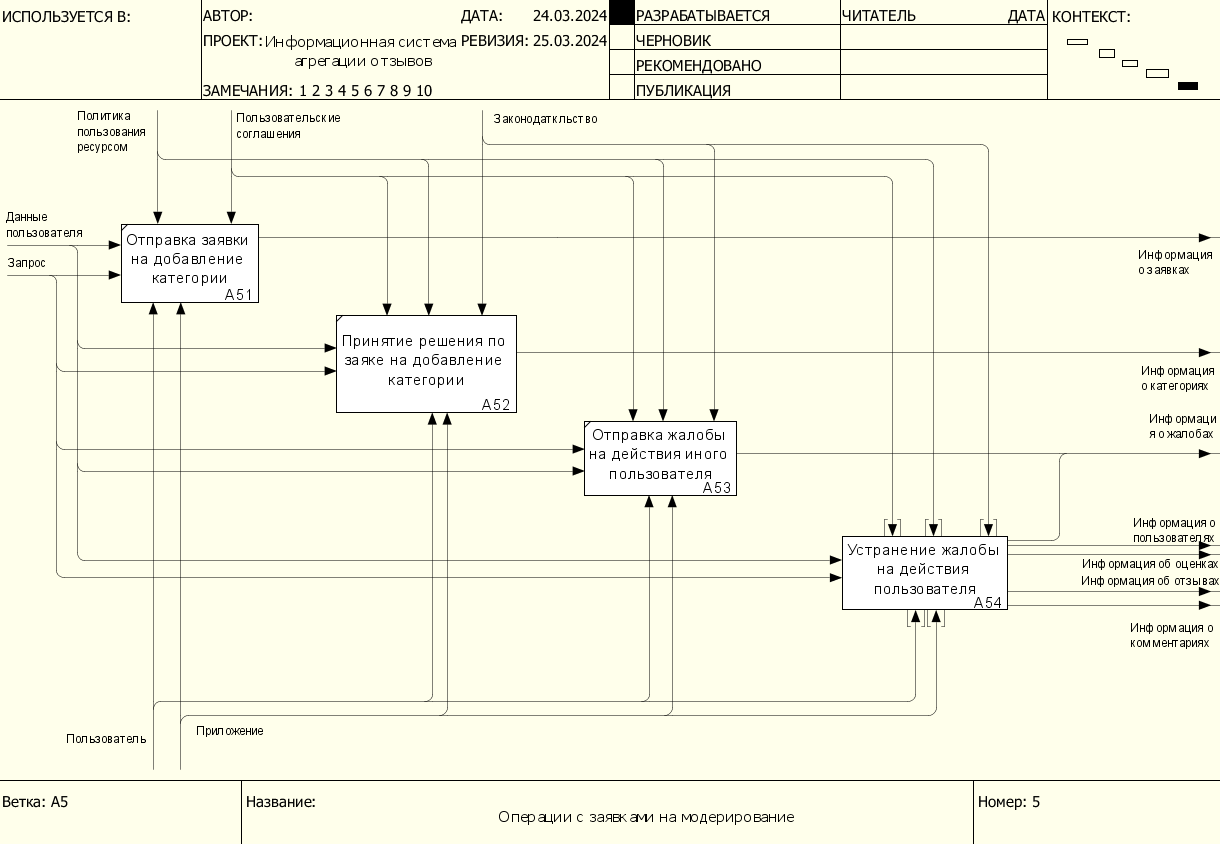
Функциональный блок «Отправка заявки на добавление категории». На вход поступает запрос и информация о пользователе. В этом процессе происходит получение заявки на добавление категории. Процесс выполняется с помощью приложение и пользователя. На выходе получается информация о заявках.

Функциональный блок «Принятие решения по заявке на добавление категории». В этом процессе происходит обработка заявки на добавление категории. На вход поступает запрос, данные пользователя. Процесс выполняется с помощью приложение и пользователя. На выходе получается информация о категориях.

Функциональный блок «Отправка жалобы на действия иного пользователя». На вход поступает запрос, информация о пользователе. В этом процессе происходит получение жалобы на те или иные действия пользователя, которые противоречат политике пользования ресурсом. Процесс выполняется с помощью приложения и пользователя. На выходе получается информация о жалобах.

Функциональный блок «Устранение жалобы на действия иного пользователя». На вход поступает запрос, информация о пользователе. В этом процессе происходит обработка жалобы на те или иные действия пользователя, которые противоречат политике пользования ресурсом. Процесс выполняется с помощью приложения и пользователя. На выходе получается информация о жалобах, пользователях, оценках, отзывах, комментариях.

На рисунке 5.2 представлена декомпозиция процесса «Операции с заявками на модерирование».



* 1. Декомпозиция процесса «Операции с заявками на модерирование»

РЕЗУЛЬТАТЫ ВЫПОЛНЕНИЯ ПРАКТИЧЕСКОЙ РАБОТЫ

В результате выполнения практической работы была произведена декомпозиция функциональной модель проектируемой системы в нотации IDEF0 и создана декомпозиция двух уровней, а также было составлено описание функциональных блоков и потоков данных.

ПРАКТИЧЕСКАЯ РАБОТА №6

ЗАДАНИЕ НА ПРАКТИЧЕСКУЮ РАБОТУ

Выбрать наиболее значимый функциональный блок нижнего уровня декомпозиции из предыдущей практической работы и выполнить его декомпозицию в нотации DFD. Декомпозиция выполняется как отдельный файл в нотации диаграммы потоков данных. Необходимо создать двухуровневое описание потоков данных. Добавить текстовое описание потоков и блоков, сделать выводы к работе.

ВЫПОЛНЕНИЕ ПРАКТИЧЕСКОЙ РАБОТЫ

При декомпозиции контекстной диаграммы «Устранение жалобы на действия пользователя» были спроектированы следующие функциональные блоки:

1. Выбор жалобы;

2. Ознакомление с жалобой;

3. Оценка объективности жалобы;

4. Отклонение жалобы;

5. Устранение причин жалобы.

Функциональный блок «Выбор жалобы». В этом процессе происходит оформление выбор жалобы для ее последующей обработки. На вход поступают данные пользователя, запрос. На выходе получаются данные выбранной жалобы, с которой будут происходить следующие действия.

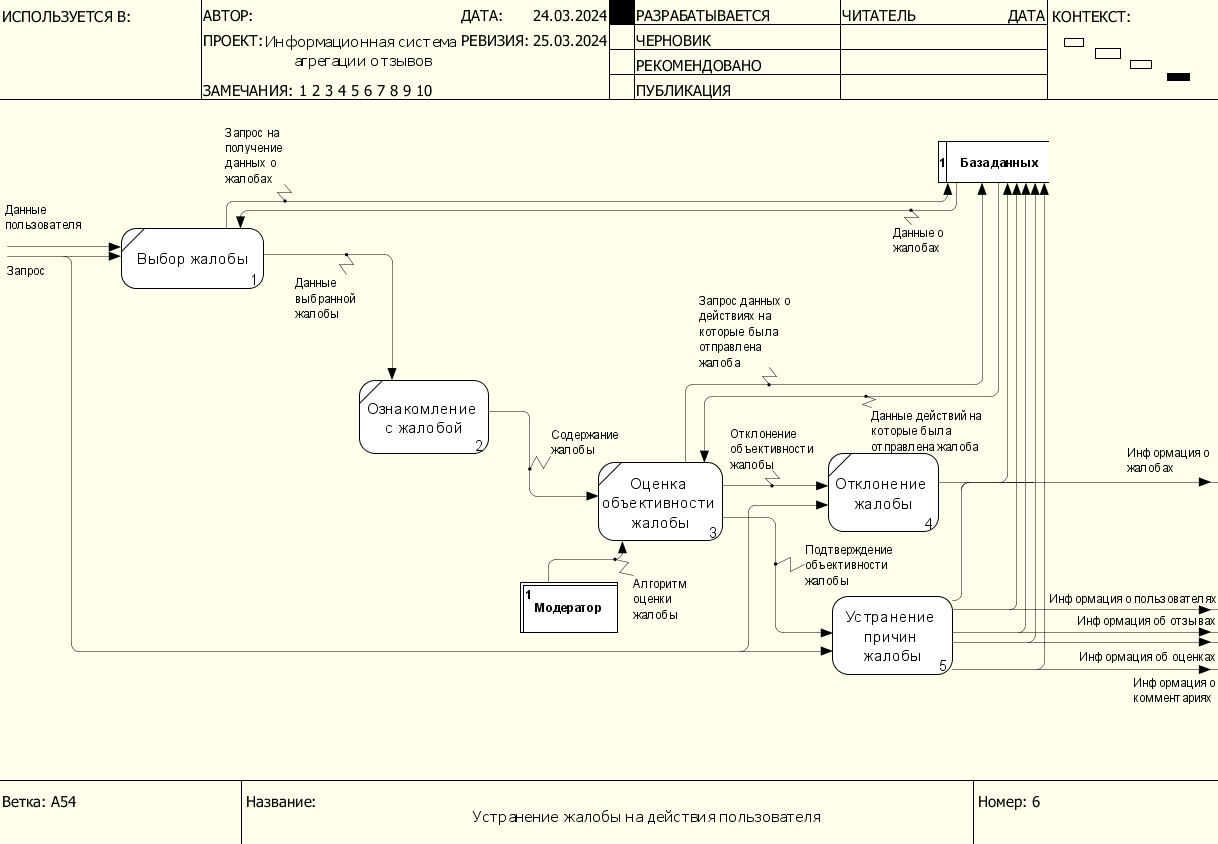
Функциональный блок «Ознакомление с жалобой». В этом процессе происходит анализ и изучение жалобы. На вход поступают данные выбранной жалобы. На выходе получается содержание жалобы, которое будет исследоваться далее.

Функциональный блок «Оценка объективности жалобы». В этом процессе сотрудник сервса проводит оценку жалобы и выносит решение о ее отклонении либо одобрении. На вход поступает содержание жалобы. На выходе получается решение сотрудника по жалобе.

Функциональный блок «Отклонение жалобы». В этом процессе отклоняется жалоба. На вход поступает информация об отклонении жалобы от сотрудника. На выходе получается информация о жалобах.

Функциональный блок «Устранение причин жалобы». В этом процессе заключение по жалобе приводится в действие. На вход подтверждение объективности жалобы от сотрудника. На выходе получается информация о жалобах, пользователях, оценках, отзывах, комментариях.

На рисунке 6.1 представлена декомпозиция процесса «Устранение жалобы на действия пользователя».



* 1. Декомпозиция процесса «Устранение жалобы на действия пользователя» в нотации DFD

При декомпозиции контекстной диаграммы «Устранение причин жалобы» были спроектированы следующие функциональные блоки:

1. Закрытие жалобы;

2. Удаление действий-причин жалобы;

3. Проверка аторитета пользоватля а которого была направлена жалоба;

4. Блокировка пользователя.

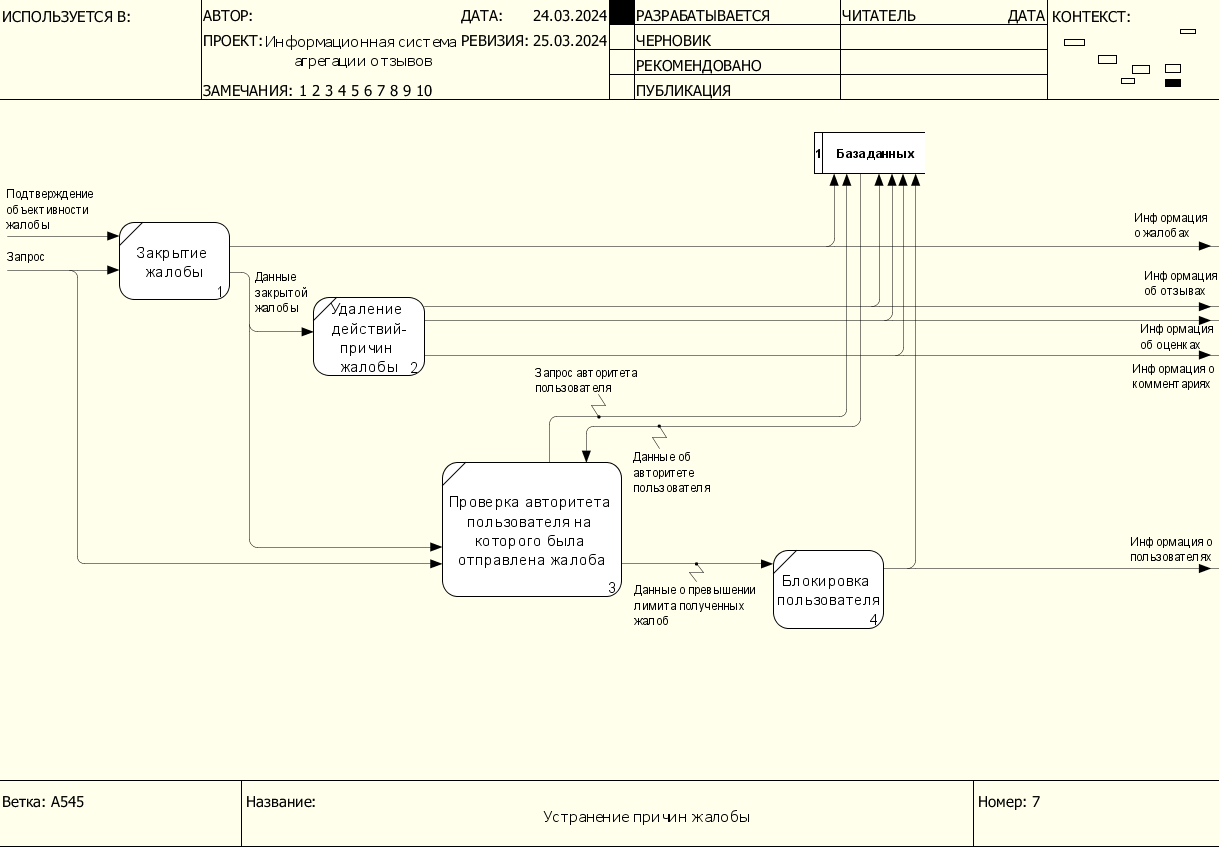
Функциональный блок «Закрытие жалобы». В этом процессе сотрудник выбирает выбирает еще не закрытую жалобу и проводит над ней операцию закрытия. На вход получаем подтверждение объективности жалобы и запрос. На выходе получаются данные закрытой жалобы.

Функциональный блок «Удаление действий-причин жалобы». В этом процессе сотрудник выполняет набор действий, для устранения причин и следствий, вызвавших жалобу.

На этапе «Проверка аторитета пользоватля а которого была направлена жалоба» сотрудник исследует недавние действия пользователя, на которого поступила жалоба, чтобы вручную модерировать его контент. На вход подаются данные закрытой жалобы и запрос, на выходе получаютися данные о превышении лимита полученных жалоб.

Функциональный блок «Блокировка пользователя». В этом процессе сотрудник выполняет набор действий, для блокировки учетной записи пользователя, получившего опредленное количество жалоб. На выходе получаем информацию о пользователях.

На рисунке 6.2 представлена декомпозиция процесса «Устранение причин жалобы».



* 1. Декомпозиция процесса «Устранение причин жалобы»

РЕЗУЛЬТАТЫ ПРАКТИЧЕСКОЙ РАБОТЫ

В ходе выполнения практической работы был выбран наиболее значимый функциональный блок нижнего уровня декомпозиции из предыдущей практической работы и была произведена его декомпозиция на 2 уровня в нотации DFD. Также было составлено текстовое описание полученных диаграмм.

ОТВЕТЫ НА КОНТРОЛЬНЫЕ ВОПРОСЫ К ПРАКТИЧЕСКИМ РАБОТАМ № 4–6

1. Что такое функциональная модель информационной системы?

Функциональная модель информационной системы – абстрактная графическая модель, которая отражает функции, выполняемые информационной системой, их связь.

2. Какие нотации можно использовать для создания функциональной модели информационной системы?

IDEF0, DFD,

3. Как строится функциональная модель информационной системы в нотации IDEF0?

Функциональная модель в нотации IDEF0 строится иерархически

4. Что является самым важным при создании диаграммы IDEF0 уровня А-0?

Самое важное на диаграмме А-0 – это название функционального блока. Название должно отражать цель функционирования всей системы.

5. Какие нормативные документы регламентируют создание диаграмм IDEF0?

РД IDEF0 – 2000

6. Приведите основные требования к названию функционального блока диаграммы IDEF0 уровня А-0.

ясность изложения цели для всех читающих диаграмму

7. Приведите основные требования к названиям функциональных блоков диаграммы IDEF0.

Основные требования к названиям функциональных блоков на диаграмме IDEF0:

Название должно отражать основную функцию блока.

Название должно быть кратким и информативным.

Название должно быть уникальным в рамках всей диаграммы.

В названии необходимо использовать глаголы, чтобы ясно указать действие или процесс, выполняемый блоком.

8. Какие атрибуты отличают диаграмму IDEF0, отвечающую требованиям нормативных документов, от простой картинки?

схему только тогда можно считать диаграммой IDEF0, когда присутствует рамка и штампы

9. Приведите требования нормативных документов по определению числа стрелок на диаграмме IDEF0. Являются ли эти требования выполнимыми? Как правильно интерпретировать эти требования?

Стрелок не должно быть много (одна из рекомендаций РД IDEF0 – 2000 – всего одна стрелка!!! Эту рекомендацию, конечно же, следует считать сильно идеализированной). Увлекаться сокращением числа стрелок также не стоит. Их должно быть столько, чтобы другие участники проекта информационной системы могли четко понять и разграничить потоки, связанные с главным функциональным блоком. Традиционно рекомендуют использовать не более 6 стрелок.

10. Что необходимо для однозначного и полного понимания диаграммы IDEF0?

требуется текстовое пояснение, а термины, использованные в процессе создания и описания контекстной диаграммы, должны быть отражены в ГЛОССАРИИ

11. Что подразумевается под декомпозицией диаграмм в нотации IDEF0? Как можно осуществлять поиск функций-потомков и функций-родителей в нотации IDEF0? Какие CASE-средства поддерживают правильную нумерацию диаграмм и функциональных блоков IDEF0?

Декомпозиция предназначена для детализации процессов, протекающих в системе.

каждый блок имеет одну родительскую и несколько дочерних диаграмм (кроме контекстной диаграммы). Иерархия производится с помощью номера, состоящий из префикса и числа.

Нумерация на определенном уровне осуществляется слева направо по расположению блоков. Контекстная диаграмма имеет номер А-0, а ее декомпозиция А0.

12. Назовите основные правила декомпозиции функциональных моделей согласно методологии SADT.

Рекомендуется использовать от трех до шести функциональных блоков на одном уровне декомпозиции.

13. Какие требования предъявляются к стрелкам, входящим в блок функциональной модели IDEF0?

используются для обозначения потоков данных между блоками функциональной модели

Направление: Стрелки всегда указывают направление передачи данных или управления. Они выходят из исходного блока и входят в целевой блок.

Надпись: На стрелках обязательно должна быть надпись, указывающая на то, какие данные или управление передаются между блоками.

Интерпретация: Стрелки должны быть однозначно интерпретируемыми для того, чтобы понять, что они представляют.

Согласованность: Стрелки должны быть согласованы с описанием потоков данных в модели и не должны противоречить другим элементам модели.

Цвет и стиль: Иногда в IDEF0 используют цвета или стили стрелок для указания особых типов данных или свойств передаваемых потоков.

Взаимосвязь: Стрелки должны четко показывать, какие данные передаются между блоками, и отражать логику потоков в системе.

14. Какие процессы отражает диаграмма в нотации DFD?

Диаграмма в нотации DFD (Data Flow Diagram) отражает процессы системы, которые могут быть представлены в виде функций или операций, взаимосвязи между этими процессами, а также потоки данных между ними. Главными элементами, которые могут быть отражены на диаграмме DFD, являются:

Процессы (Processes): Представляют выполнение определенных операций или действий над данными в системе. Эти процессы могут быть как высокоуровневыми, так и детализированными.

Потоки данных (Data Flows): Обозначают передачу данных между различными процессами или акторами системы. Они показывают, как информация движется через систему.

Хранилища данных (Data Stores): Репозитории данных, где информация хранится в системе для дальнейшего использования.

Внешние сущности (External Entities): Представляют внешние источники или назначения данных, которые взаимодействуют с системой.

15. На каком уровне декомпозиции можно использовать нотацию DFD?

любом

16. Сколько уровней описания в нотации DFD рекомендуется выполнить в работе? 2

17. Как рекомендуется создавать DFD-диаграмму с использованием CASE-средства Ramus Educational?

При построении диаграммы в нотации DFD не забывайте, что процессы должны активно взаимодействовать либо друг с другом, либо через хранилища данных, тем самым передавая информацию. При необходимости обозначайте, кто получает или передает информацию (внешние сущности) и через что это происходит (документы, хранилища данных).

18. Как расшифровывается аббревиатура DFD?

Data Flow Diagram

19. Как расшифровывается аббревиатура IDEF0?

Integrated Computer Aided Manufacturing Definition

20. Как расшифровывается аббревиатура SADT?

structured analysis and design technique

21. Назовите ограничения нотации методологии SADT.

SADT сосредотачивается на предметных функциях и логической структуре информации, исключая из рассмотрения организационную структуру и вещественные компоненты.

SADT-модели не позволяют моделировать изменение состояния систем во времени или в результате внешних воздействий и управленческих решений.

22. Назовите достоинства методологии SADT.

универсальность и простота, доступный графический язык, возможность использования на первых стадиях жизненного цикла информационных систем.

23. Поясните связь между методологией SADT и нотациями IDEF0 и DFD.

Методология SADT (Structured Analysis and Design Technique) использует нотации IDEF0 (Integration Definition for Function Modeling) и DFD (Data Flow Diagrams) для моделирования функциональных аспектов системы. К примеру, IDEF0 часто используется в рамках SADT для графического представления функций системы, в то время как DFD помогает в визуализации потоков данных в системе.

ПРАКТИЧЕСКАЯ РАБОТА №7

ЗАДАНИЕ НА ПРАКТИЧЕСКУЮ РАБОТУ

Создание модели «сущность – связь» в нотации ERD. В процессе выполнения работы необходимо решить задачу анализа модели потоков данных в нотации DFD и формирования ER-диаграммы. Последовательность выполнения работы представлена далее.

1. Создание плана разработки проекта БД ИС. В процессе необходимо рассматривать БД как составную часть проектируемой ИС.

2. Создание текстового описания информационных объектов, сущностей и связей Проектируемой БД. Для этой цели повторно выполнить анализ предметной области создания ИС с целью выявления сущностей и связей. В процессе полезно выполнить описание типовых запросов проектируемой БД.

3. Построение концептуальной модели данных, создание ER-диаграммы «сущность – связь».

4. Проверка полноты и корректности созданной ER-диаграммы с использованием языка SQL. Для проверки необходимо создать типовые запросы, например, поиск и анализ данных.

5. Оформление отчета о выполненной работе. Отчет должен содержать: план разработки модели БД, выдержки из анализа предметной области, касающиеся определения сущностей БД и связей между ними, ER-диаграмму, примеры тестовых запросов SQL.

ВЫПОЛНЕНИЕ ПРАКТИЧЕСКОЙ РАБОТЫ

Задача системы – выступить в качестве надежного источника информации о различных товарах, услугах, заведениях и местах отдыха.

Одной из ключевых причин востребованности подобного сервиса является его способность предоставлять объективную и непредвзятую информацию. Отзывы реальных пользователей, оставленные на подобных платформах, формируют ценный источник данных для принятия решений. Потребители все чаще обращаются к таким сервисам перед совершением покупок или выбором места для посещения, стремясь избежать негативного опыта.

Таким образом, проектируемая система должна выполнять следующие действия:

* + - 1. Хранить данные о категориях, отзывах, оценках, комментариях, жалобах;
      2. Предоставлять возможность отслеживания оставления отзывов;
      3. Предоставлять возможность выбора категорий;
      4. Предоставлять функционал для модераторов системы;
      5. Осуществление общего контроля контента пользователями.

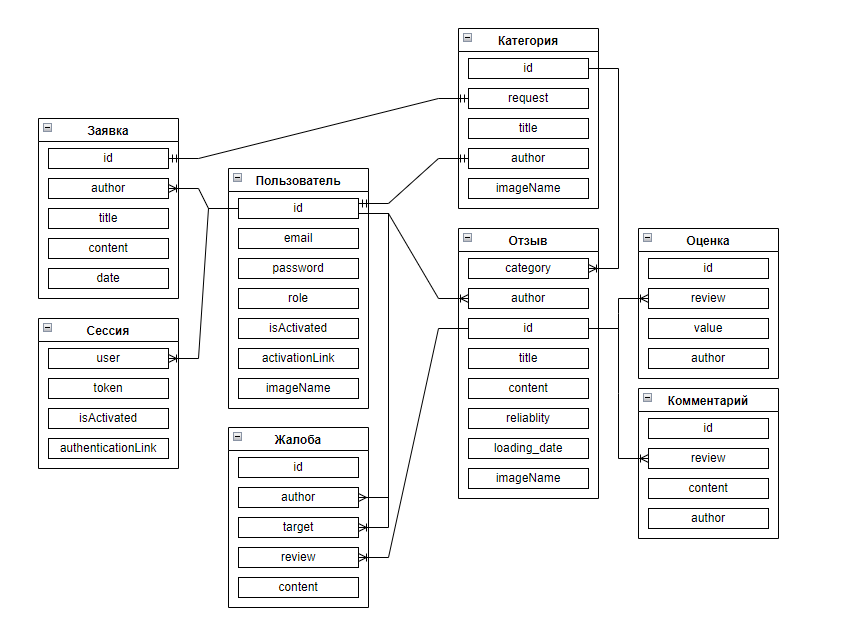
Выделим сущности из действий системы:

* **заявка** – явная сущность;
* **категория** – явная сущность;
* **пользователь** – сущность, но требуется уточнение роли для пользователя.
* **отзыв** – явная сущность;
* **сессия** – сущность;
* **оценка** – явная сущность;
* **комментарий** – явная сущность.

Выделим связи между сущностями и таблицами:

* **заявка-категория** – один-к-одному – у одной заявки может быть только одна категория;
* **заявка-пользователь** – многие-к-одному – у пользователя может быть несколько заявок на добавление категории;
* **сессия-пользователь** – многие-к-одному – у пользователя может быть несколько сессий;
* **пользователь-категория** – один-ко-одному – у категории может быть только один автор;
* **пользователь-отзыв** – один-ко-многим – у одного пользователя может быть много отзывов;
* **категория-отзыв** – один-ко-многим – в одной категории может быть много отзывов;
* **отзыв-оценка** – один-ко-многим – у одного отзыва может быть много оценок;
* **отзыв-комментарий** – один-ко-многим – у одного отзыва может быть много комментариев;
* **отзыв-жалоба** – один-ко-многим – у одного отзыва может быть много жалоб;
* **пользователь-жалоба** – один-ко-многим – у одного пользователя может быть много жалоб;

На рисунке 7 представлена логическая ER-диаграмма проектируемой системы системы.



1. Логическая ER-диаграмма системы

Примеры SQL запросов в случае использования SQL БД согласно представленной диаграмме:

INSERT INTO Заявка (название\_заявки, описание, id\_категории, id\_пользователя) VALUES ('Новая заявка', 'Описание новой заявки', 1, 1);

INSERT INTO Отзыв (текст\_отзыва, id\_пользователя) VALUES ('Отличный отзыв!', 1);

UPDATE Заявка SET id\_категории = 2 WHERE id\_заявки = 1;

UPDATE Отзыв SET текст\_отзыва = 'Измененный отзыв' WHERE id\_отзыва = 1;

DELETE FROM Заявка WHERE id\_заявки = 1;

DELETE FROM Комментарий WHERE id\_отзыва = 1;

РЕЗУЛЬТАТЫ ПРАКТИЧЕСКОЙ РАБОТЫ

В результате практической работы была создана модель «сущность – связь» в нотации ERD. В процессе выполнения работы была решена задача анализа модели потоков данных в нотации DFD и сформирована ER-диаграммы системы.

ОТВЕТЫ НА КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Что такое база данных информационной системы? Дайте определение.

База данных (БД) – важнейший компонент информационной системы. База данных – набор данных, структурированный особым образом в рамках выбранной модели.

2. Что такое модель данных? Дайте определение.

Модель данных – это концептуальное представление данных (в частности сущностей и связей между ними) в проектируемой БД, выраженное графически (в виде схемы) и/или описательно (текстовое представление). Модель данных создается для того, чтобы разработчик понимал концепцию организации хранения и обработки данных.

3. Что такое «сущность» в рамках ERD-моделирования?

Сущность, как объект диаграммы entity-relation, описывает или представляет некоторые объекты или наборы объектов, как реальных, так и абстрактных.

4. Что такое «связь» в рамках ERD-моделирования?

Связь − определенная ассоциация (или соответствие, взаимозависимость) между двумя сущностями. Связь может соединять не только две сущности, но сущность «саму с собой». Сущность необходимы для поиска взаимосвязей внутри БД

5. Какую роль играет ER-диаграмма в проекте информационной системы?

Модель данных создается для того, чтобы разработчик понимал концепцию организации хранения и обработки данных.

6. Какую задачу надо решить в процессе выполнения работы?

В процессе выполнения работы необходимо решить задачу анализа модели потоков данных в нотации DFD и формирования ER-диаграммы.

7. В каком аспекте рассматривается БД информационной системе в рамках данной работы?

В процессе необходимо рассматривать БД как составную часть проектируемой ИС.

8. Каким образом определяются сущности в процессе создания ER-диаграммы?

Для этой цели повторно выполнить анализ предметной области создания ИС с целью выявления сущностей и связей.

9. Кто из авторов предложил наиболее часто используемую нотацию ERD-моделирования?

Питер Чен.

10. Что такое «экземпляр сущности»? Дайте определение.

«Экземпляр» сущность – её конкретный представитель.

11. Каков принцип группировки объектов в общую сущность ERD?

Объекты описываются или группируются в виде сущности по характерному признаку или характеристике, которые позволяют обособить данный объект от других.

12. Какие элементы входят в модель ERD?

Сущности и связи.

13. Какие CASE-средства рекомендованы для разработки ER-диаграммы?

draw.io, ER Win.

14. Какие типы связей доступны при создании ER-диаграммы?

Один-к-одному, один-ко-многим, многие-ко-многим.

15. Назовите правила и приведите пример использования связей один-ко-многим ER-диаграммы?

При таком типе связей несколько строк из дочерний таблицы зависят от одной строки в родительской таблице. Например, в одной группе учатся несколько студентов. В этом случае таблица группы является родительской, а таблица студентов – дочерней.

16. Назовите правила и приведите пример использования связей один-к-одному ER-диаграммы?

При таком типе связи только одна запись первой сущности связана только с одной запись в другой сущности. Например, у одного преподавателя на кафедре есть один компьютер.

17. Назовите правила и приведите пример использования связей многие-ко-многим ER-диаграммы?

При использовании типа связи «многие-ко-многим» подразумевается, что множественным записям из одной таблицы соответствуют множественные записи из другой таблицы. Например, у студентов групп предметы ведут разные преподаватели.

ПРАКТИЧЕСКАЯ РАБОТА №8

ЗАДАНИЕ НА ПРАКТИЧЕСКУЮ РАБОТУ

Создать диаграмму состояний проектируемой информационной системы для одного из ранее разработанных классов или прецедентов.

ВЫПОЛНЕНИЕ ПРАКТИЧЕСКОЙ РАБОТЫ

Для создания диаграммы состояний проектируемой системы выберем прецендент «Рассмотрение заявок на добавление категорий отзывов».

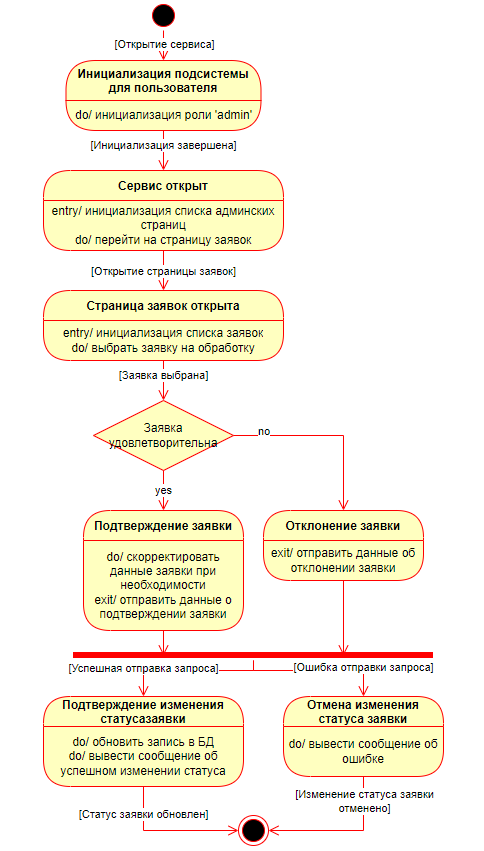
Первым состоянием жизненного цикла системы будет ее инициализация. В текущем состоянии выполняется инициализация подсистемы для пользователя с ролью «admin».

При инициализации роли процесс переходит к следующему состоянию «Сервис открыт». Происходит открытие главной страницы приложения с инициализацией списка дополнительных страниц к которым имеет доступ пользователь с ролью «admin», после чего пользователь переходит на страницу администрирования заявок. На данной странице происходит инициализация списка всех заявок. Пользователь выбирает заявку из списка на рассмотрение.

После выбора заявки процесс переходит в состояние «Подтверждение заявки» или «Отклонение заявки» в зависимости от соблюдения условия удовлетворительности данной заявки. В состоянии «Подтверждение заявки» пользователь при необходимости корректирует данные заявки и отправляет их на сервер. В случае состояния «Отклонение заявки» администратор сразу отправляет данные об изменении статуса заявки на «Отклонено» на сервер.

Далее следует развилка в виде успешной или неудачной отправки запроса на сервер. Если запрос успешно отправился, то происходит переход в состояние «Подтверждение изменения статуса заявки». В этом состоянии система автоматически обновляет статус заявки в БД, а у пользователя отображается сообщение об успешной отправке заявки. В случае ошибки система переходит в состояние «Отмена изменения статуса заявки», в котором система сразу отправляет пользователю сообщение об ошибке.

После всех действий процесс системы заканчивается в первом случае обновлением статуса заявки, в ином случае отменой изменения статуса заявки по причине ошибки отправки запроса. Полная диаграмма представлена на рисунке 8.1.



* 1. Диаграмма состояний прецедента «Рассмотрение заявок на добавление категорий отзывов»

РЕЗУЛЬТАТЫ ПРАКТИЧЕСКОЙ РАБОТЫ

В ходе выполнения практической работы была создана диаграмма состояний для одного из ранее разработанных прецедентов.

ОТВЕТЫ НА КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Для чего в язык UML введена диаграмма состояний?

Диаграммы введены в язык UML для моделирования динамических аспектов системы. По сути, диаграмма состояний соответствует таковой в теории автоматов.

2. Для каких целей используется диаграмма состояний UML?

Полезно использовать диаграмму состояний для моделирования жизненного цикла информационной системы или ее частей.

3. Что выступает в качестве объекта моделирования диаграммы состояний UML?

В качестве моделируемого объекта выступают, как правило, классы, прецеденты или система в целом.

4. Каким образом передается основная информация, заложенная в диаграмме состояний UML?

Кроме редких исключений основная часть несомой ею информации содержится не в размере или расположении еѐ элементов, а в еѐ топологической конфигурации.

5. Какие типы отношений чаще всего применяются в диаграмме состояний UML?

Чаще всего применяются три основных типа визуальных отношений между элементами: соединение (в виде линий от одной двумерной формы к другой), включение (в виде помещения одних форм внутрь других) и визуальное подкрепление (в виде размещения одних символов недалеко от других).

* + - 1. Как можно представить диаграмму состояний UML в аналитическом виде?

Возможно также представить данную нотацию и в аналитической форме как совокупность связей узлов графа.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Лобанов, А. А. Проектирование информационных систем [Электронный ресурс]: учебное пособие / А. А. Лобанов, Ю. С. Лобанова, Е. Н. Абраш, Н. В. Братусь; под ред. А. А. Лобанова. – Электрон. текст. дан. (3,2 Мб). – Киров: Изд-во МЦИТО, 2023. – Изд. 2-е, перераб. и доп. – 1 электрон. опт. диск (CD-R). – ISBN 978-5-907743-35-9 (дата обращения 12.02.2024)
2. ГОСТ 7.32−2017. Межгосударственный стандарт. Система стандартов по информации, библиотечному и издательскому делу. Отчет о научноисследовательской работе. Структура и правила оформления": введен в действие Приказом Росстандарта от 24.10.2017 N 1494-ст: дата введения 2018-01- 07. – URL: http://www.consultant.ru/document/cons\_doc\_LAW\_292293/ (дата обращения: 12.02.2024). – Текст: электронный.
3. ГОСТ 34.602−2020. Межгосударственный стандарт. Информационные технологии. Комплекс стандартов на автоматизированные системы. Техническое задание на создание автоматизированной системы: Приказом Федерального агентства по техническому регулированию и метрологии № 1522-ст от 19 ноября 2021 г.: дата введения 2022-01-01. – М.: Российский институт стандартизации, 2021. – 12 с. – URL: https://docs.cntd.ru/document/1200181804 (дата обращения 12.02.2024). – Текст: электронный.