

Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное  
учреждение высшего образования  
«Пермский государственный аграрно-технологический университет  
имени академика Д. Н. Прянишникова»

Беляков А.Ю.

## **ПРОГРАММНАЯ ИНЖЕНЕРИЯ**

Методические рекомендации по курсовому проектированию

*Пермь*  
*ФГБОУ ВО Пермский ГАТУ*  
*2022*

**УДК 004.43**  
**ББК 32.973-018.1**  
**Б 448**

*Рецензенты:*

Е.А. Муратова, к.э.н., доцент, заведующий кафедры информационных технологий и программной инженерии (ФГБОУ ВО Пермский ГАТУ)

А.А. Зорин, к.т.н., доцент кафедры информационных технологий и программной инженерии (ФГБОУ ВО Пермский ГАТУ)

**Б 448 Беляков А.Ю.**

Программная инженерия: методические рекомендации по курсовому проектированию / авт. А.Ю. Беляков; М-во науки и высшего образования РФ, федеральное гос. бюджетное образов. учреждение высшего образования «Пермский гос. аграрно-технолог. университет им. акад. Д.Н. Прянишникова», каф. информационных технологий и программной инженерии. – Пермь: ФГБОУ ВО Пермский ГАТУ, 2022. – 31 с.

В методических рекомендациях по выполнению курсового проекта представлена постановка задачи на курсовое проектирование и сформулированы требования к технической реализации проекта.

Методические рекомендации по курсовому проектированию предназначены для обучающихся очной и заочной форм обучения по направлениям подготовки 09.03.03 Прикладная информатика, 09.03.04 Программная инженерия.

**УДК 004.43**  
**ББК 32.937-018.1**

Рекомендованы к изданию методической комиссией факультета экономики и информационных технологий ФГБОУ ВО Пермский ГАТУ, протокол № \_ от \_\_\_\_ 202\_г.

© ФГБОУ ВО Пермский ГАТУ, 202\_  
© Беляков А.Ю., 202\_

## Содержание

Введение .....	4
1. Постановка задачи на курсовое проектирование.....	5
2. Анализ бизнес-процесса.....	6
3. Разработка информационной системы.....	8
4. Пример программной реализации .....	13
5. Подготовка и защита курсового проекта .....	17
Заключение .....	20
Перечень основной и дополнительной литературы .....	21
Базы данных, информационно-справочные и поисковые системы .....	22
Приложение А. Шаблон технического задания на разра- ботку информационной системы .....	24
Приложение Б. Шаблон титульного листа отчёта о кур- совом проектировании .....	29
Приложение В. Шаблон рецензии на проект .....	30

## Введение

Методические рекомендации предназначены для выполнения курсового проектирования по направлениям подготовки 09.03.03 Прикладная информатика, направленность (профиль) «Прикладная информатика в экономике», 09.03.04 Программная инженерия, направленность (профиль) «Разработка программно-информационных систем» по дисциплине «Программная инженерия». Существенную роль в освоении компетенций по дисциплине «Программная инженерия» играет отработка умений самостоятельного выполнения технических проектов автоматизации бизнес-процессов, что вызывает потребность в разработке методических рекомендаций.

Целью данных методических рекомендаций по курсовому проектированию является получение опыта подготовки технического задания на проектирование информационной системы, разработка и тестирование информационной системы обучающимися, усвоенных ими при контактной работе с преподавателем и при изучении учебной, методической, справочной литературы.

Основной задачей методических рекомендаций по курсовому проектированию является расширение навыков самостоятельной профессиональной деятельности обучающимися и проверка знаний по программированию сложных информационных систем.

В методических рекомендациях по курсовому проектированию представлены постановка задачи на курсовой проект, особенности разработки информационной системы, включая описание используемых технологий, архитектуру приложения, интерфейс пользователя. Методические рекомендации содержат описание результатов курсового проектирования,

перечень рекомендуемой основной и дополнительной литературы, базы данных и информационно-справочные и поисковые системы.

В рамках выполнения задач курсового проектирования будут закреплены следующие комплексные умения и практические навыки по выполнению программной реализации:

- проектировать и разрабатывать архитектуру web-приложения;
- передавать данные с формы на форму;
- возвращаться с дочерней формы на родительскую;
- проверять соединение с базой данных;
- делать запросы SELECT, INSERT, UPDATE с параметрами;
- хешировать пароли;
- запускать методы по обработке данных синхронно или асинхронно.

## **1. Постановка задачи на курсовое проектирование**

Курсовое проектирование посвящено рассмотрению методики автоматизации бизнес-процессов предприятия и выполнению этапов, связанных с разработкой программной реализации.

Последовательность автоматизации бизнес-процессов предприятия можно разбить на следующие самостоятельные этапы:

- исследование предприятия;
- выбор неавтоматизированного или слабо автоматизированного процесс;
- составление модели процесса «как-есть» и «как-будет»;

- подготовка Технического Задания на разработку информационной системы;
- разработка прототипа и проведение апробации;
- разработка программной реализации;
- осуществление тестирования;
- внедрение технического решения;
- сопровождение и доработки технического решения.

Из приведённой последовательности этапов в курсовом проектировании будут задействованы:

- составление Технического задания;
- разработка прототипа;
- апробация информационной системы.

Таким образом, в рамках курсового проектирования предстоит разработать и апробировать техническое решение в виде web-приложения для автоматизации бизнес-процесса предприятия.

В техническое решение будут интегрированы такие вопросы проектирования и программирования как: архитектура web-приложения, MVC-паттерн проектирования, организация обмена данными между программой и базой данных, регистрация и авторизация пользователя, передача параметров и данных между формами приложения, хеширование паролей пользователей.

## **2. Анализ бизнес-процесса**

В рамках курсового проектирования рассмотрим бизнес-процесс работы с клиентами интернет-магазина по продаже некоторой продукции. Важная составная часть такого процесса – это обратная связь от клиентов о продукции, об услугах, об оказанном сервисе.

Отзывы клиентов – это форма обратной связи от пользователей на продукты или услуги и неотъемлемый атрибут современного бизнеса. Отзывы формируют лояльность клиента, информируют пользователей о продукте с точки зрения потребителя, помогают выявить недостатки, улучшить продукт и принести компании увеличение продаж.

Подавляющее большинство покупателей читают отзывы перед покупкой. Продукт без отзывов вызывает опасения, а продукт с исключительно положительными отзывами – недоверие, поэтому игнорировать обратную связь от клиентов не стоит. Наличие раздела на сайте по продаже продукции с отзывами реальных клиентов способствует увеличению объёма продаж.

Попробуем формализовать и описать преимущества, которые получает компания, когда уделяет внимание автоматизации сбора и последующему анализу отзывов клиентов.

*Обратная связь поможет усовершенствовать ваш продукт или услугу.* Автоматизация ведения отзывов позволит собрать статистику о том, что нравится вашим клиентам, а что их отталкивает и сможете уделить внимание действительно актуальным вопросам, тем самым получив преимущество над остальными конкурирующими компаниями.

*Повышение лояльности покупателей.* Спрашивать отзывы и реагировать на них – это способ показать, что вы работаете для клиента и готовы ответить на его ожидания. Плохие отклики нужно отрабатывать, а не игнорировать. Другие пользователи увидят ответ и узнают, что ваша компания не безразлична к проблемам клиентов – это плюс к лояльности.

*Удержание клиентов и повышение дохода.* Если покупатель разочаровался в продукте или получил некачественную

услугу и сообщил об этом, то оперативный ответ и исправление ситуации не только уменьшат негатив, но и помогут вернуть расположение клиента.

*Формирование будущих стратегий компании* на основе обратной связи от клиентов и создание продуктов под потребности людей.

*Дополнительное продвижение.* Отзывы – это уникальный контент с упоминанием вашего продукта. Берите отзывы на свой сайт и используйте на страницах в соцсетях.

*База отзывов может стать частью CRM-системы* – системы управления взаимоотношениями с клиентами. Это не только дополнение клиентской базы, но и подход к увеличению продаж услуг и продуктов. С помощью обратной связи вы поймете, как работает ваш продукт или услуга, какие есть недостатки, чего действительно хотят пользователи. Тогда вы улучшите продукт или услугу и доработаете их до ожиданий клиентов.

Таким образом, актуальна задача разработки web-приложения для работы с отзывами клиентов по продаваемой продукции.

### **3. Разработка информационной системы**

В рамках курсового проектирования в качестве программной реализации (прототипа) требуется разработать web-приложение "*Отзывы на продукт*".

*Используемые технологии:*

- Nodejs – платформа для разработки серверных приложений на JavaScript;
- Express – фреймворк для разработки web-приложений на JavaScript;



- `ejs` – шаблонизатор `html`-страниц;
- `MVC` – паттерн проектирования `web`-приложения;
- `csv` – способ представления табличных данных;
- `json` – способ представления иерархически организованных данных;
- `re` (`regular expressions`) – формальный язык обработки текстовой информации, для поиска и проверки комбинаций символов в тексте, основанный на использовании метасимволов;
- `SQLite` – реляционная база данных и язык запросов к табличным данным;
- `html` – язык разметки для описания структуры `web`-страниц приложения;
- `css` – таблицы стилей для описания дизайна `web`-приложения;
- `lodash` – функциональная библиотека;
- `rambda` – библиотека для обработки данных.

*Архитектура приложения* основана на паттерне проектирования `MVC` (`Model-View-Controller`) для удобства разработки, тестирования, модификации и масштабирования.

Рассмотрим кратко составные части паттерна.

*Model* (модель) – это часть архитектуры приложения, которая:

- включает в себя бизнес-логику приложения (классы, методы, функции обработки данных);
- модель "не знает" о контроллерах и представлениях, не ориентируется на них;
- данные модели могут быть представлены таблицами базы данных или файлами `xml`, `json`, `csv`;

– это может быть просто набор объектов или функций для реализации логики приложения.

*View* (представление) – это часть архитектуры приложения (примеры представления: HTML-страница, WPF форма, Windows Form), которая:

– отвечает за визуальное отображение данных, полученных от модели;

– представление может читать данные, но не может записывать или изменять их;

– представление может иметь программный код, который реализует логику отображения данных;

*Controller* (контроллер) – это часть архитектуры приложения, которая:

– задаёт совокупность обработчиков событий представления;

– принимает события от представления;

– отдаёт событие на обработку в модель;

– возвращает пользователю обновлённое представление;

– или выбирает какое именно представление должно быть отображено в ответ.

Исходя из предназначения составных частей паттерна физически можно разбить приложение на отдельные функциональные модули и разнести по соответствующим директориям приложения (рис.1).

Директория *models* предназначена для хранения модулей по реализации бизнес-логики приложения. Именно в этих модулях будут осуществляться запросы к базе данных по получению, обработке и последующему сохранению данных.

Директория *views* предназначена для хранения шаблонов web-страниц приложения, которые будут динамически заполняться в контексте работы с пользователем.

Директория `routes` предназначена для хранения контроллеров, осуществляющих функционал маршрутизации приложения в контексте обработки запросов пользователя.

В директории `public` следует размещать те файлы, к которым следует предоставить доступ всем страницам приложения без авторизации.

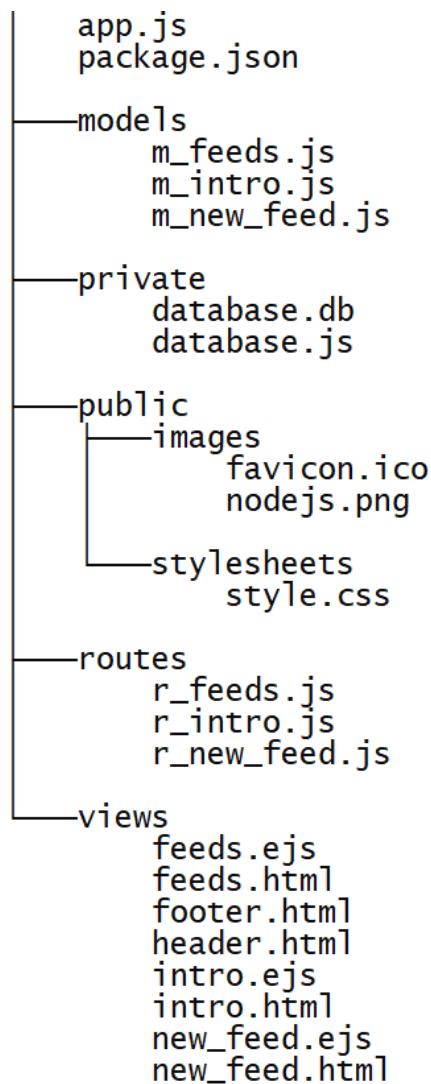


Рисунок 1. Дерево директорий приложения.

В директории `private` следует размещать те файлы. Которые могут содержать конфиденциальную информацию о пользователях, локально используемую базу данных, такую как SQLite, настроечные и инициализационные файлы приложения,

файлы для логирования действий пользователей и работы самого приложения.

*Организация интерфейса пользователя и функционал визуальных форм приложения.*

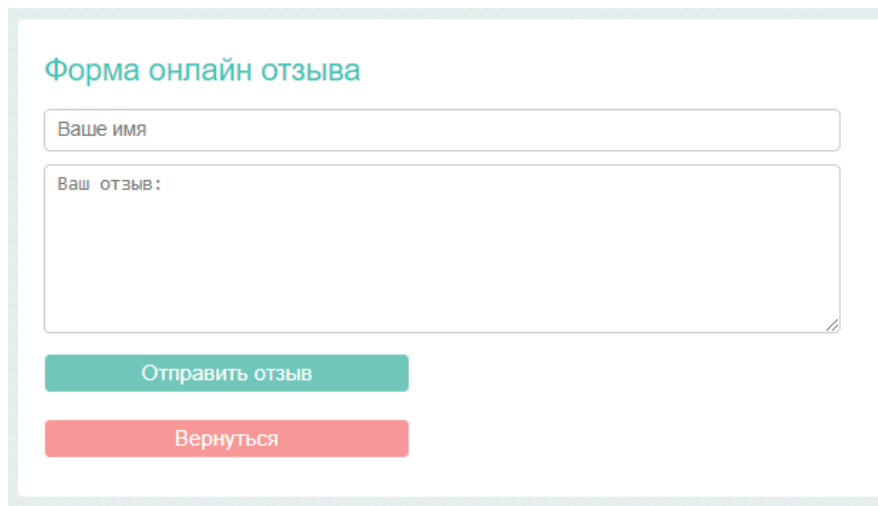
Больше всего востребован функционал по просмотру отзывов на продукт с возможностью фильтрации по выбранным параметрам и сортировки (по дате, рейтингу), поэтому удобнее всего подготовить представление данных в виде таблицы (рис.2).

В ряде случаев у покупателя может возникнуть потребность поделиться своим опытом приобретения или эксплуатации продукции. Для этого сценария использования web-приложения следует разработать специальную лаконичную форму с интуитивно понятным интерфейсом пользователя (рис.3).



Оставить новый отзыв		
Отзывы		
Дата/ Время	Имя	Отзыв
2022-5-22 12:37	Яшин Иван	Компактный, прочная ножка. Можно поворачивать вертикально. На столе занимает мало места. Кнопочки сенсорные.
2022-3-20 21:56	Акопян Амаяк	Дизайн норм, можно брать, использую как второй монитор...
2022-3-20 21:47	Елисеев Иван	Не отключается режим энергосбережения.
2022-3-20 21:46	Иванов Порфирий	Опыт использования менее месяца, полёт нормальный, всё устраивает
2022-2-20 21:44	Беляков Андрей	Попался с битым пикселем
2022-2-20 21:34	Сашко Олеся	Хороший безрамочный IPS моник за свои деньги
2022-2-19 18:20	Сидоров Олег	Дизайн устаревший
2022-2-19 17:23	Belyakov Andrey	Дороговато за 19" и есть засветы на чёрном фоне (((

Рисунок 2. Главная форма приложения.



Форма онлайн отзыва

Ваше имя

Ваш отзыв:

Отправить отзыв

Вернуться

Рисунок 3. Форма отзыва.

#### 4. Пример программной реализации

Пусть web-приложение стартует с формы intro, которая даёт представление о продаваемом продукте или услуге (рис.4).

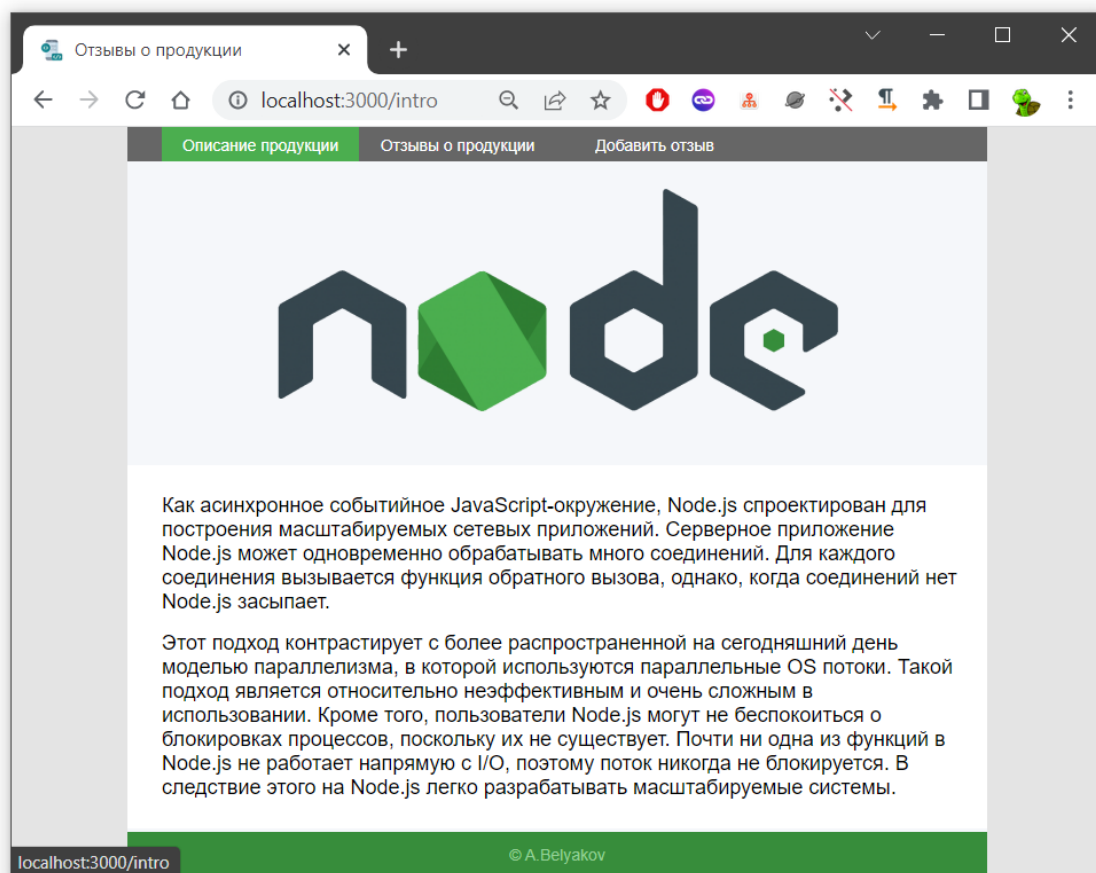


Рисунок 4. Стартовая форма приложения.

На рисунке 4 приведён пример стартовой формы приложения, но в рамках выполнения курсового проекта следует выбрать свой вид оказываемых услуг или продаваемого продукта и заменить логотип и текст описания продукта

Для пользователя в приложении предоставлены две возможности: просмотреть существующие отзывы (рис.5) и добавить свой отзыв (рис.6).

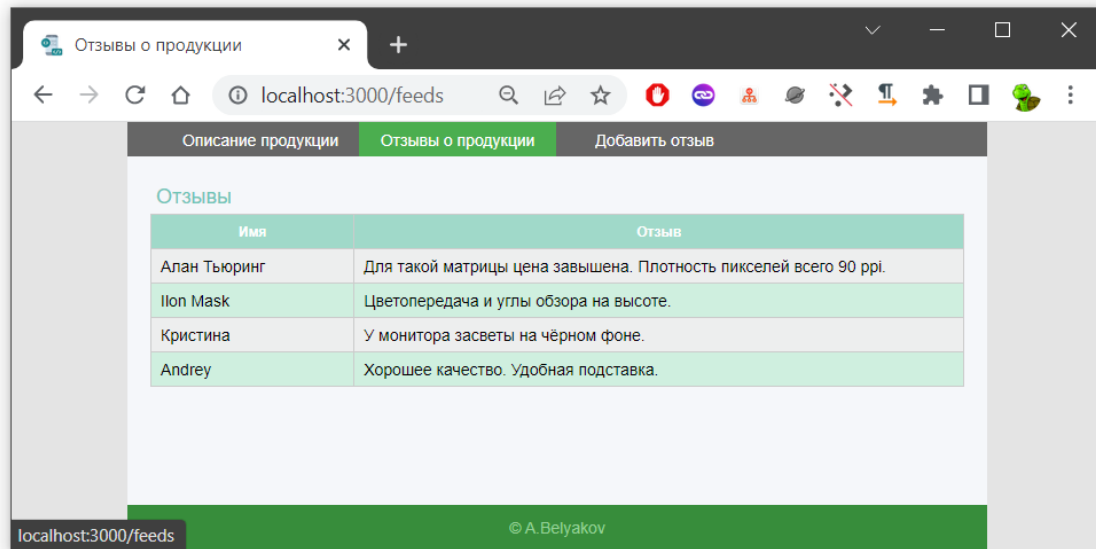


Рисунок 5. Форма просмотра отзывов.

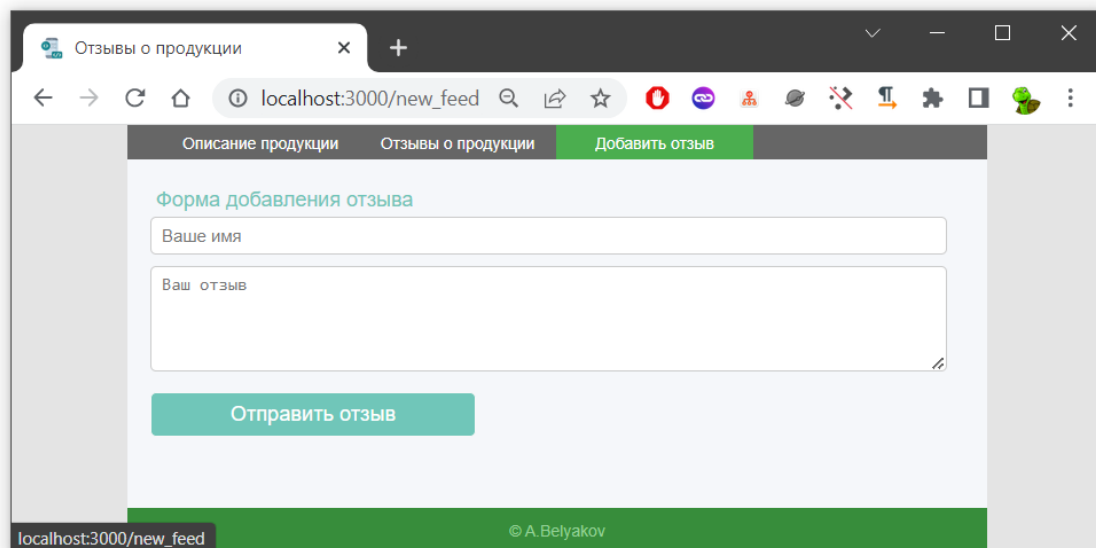


Рисунок 6. Форма добавления нового отзыва.

Так как архитектура приложения выполнена по паттерну MVC, то весь программный код разбит на отдельные модули,

которые связаны между собой стартовым модулем app.js (Листинг 1).

Листинг 1.

Файл приложения app.js.

```
const express = require('express');
const path = require('path');

const app = express(); // создаём объект приложения
app.use('/public', express.static(path.join(__dirname, 'public')));
app.set('view engine', 'ejs'); // подключаем шаблонизатор

const r_intro = require('./routes/r_intro.js');
const r_feeds = require('./routes/r_feeds.js');
const r_new_feed = require('./routes/r_new_feed.js');

app.use(['/intro', '/'], r_intro);
app.use('/feeds', r_feeds);
app.use('/new_feed', r_new_feed);

app.listen(3000, "localhost", () => console.log(`to stop: Ctrl+C`));
```

Для реализации возможности просмотра отзывов в приложении предусмотрена соответствующая маршрутизация – модуль r\_feeds.js (Листинг 2).

Листинг 2.

Модуль r\_feeds.js обработки события.

```
const express = require('express');
const router = express.Router();
const { model } = require("../models/m_feeds.js");
const dp = require("../private/database.js");

router.get('/', (req, res) => {
  model.feeds = dp.getFeeds();
  res.render('feeds.ejs', model); // render view
});

module.exports = router;
```

Обработчик события получает данные из базы данных с помощью соответствующего метода getFeeds() из класса DataProcessing (Листинг 3).

Листинг 3.  
Модуль database.js.

```
const pathDB = "./private/database.db";
const db = require('better-sqlite3')(pathDB);

class DataProcessing {
  getFeeds() {
    let query = `
      SELECT userName, userFeed
      FROM feeds
      ORDER BY idFeed DESC`;
    let rows = db.prepare(query).all();
    return rows;
  }
  insertFeed(name, feed) {
    let values = { name: name, feed: feed };
    let query = `
      INSERT INTO feeds (userName,userFeed)
      VALUES (@name, @feed)`;
    db.prepare(query).run(values);
  }
}

module.exports = new DataProcessing();
```

Полученный из базы данных **database.db** массив объектов требуется разместить на соответствующей форме приложения, для чего был подготовлен шаблон html-страницы (Листинг 4).

Листинг 4.  
Модуль feeds.ejs.

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8" />
  <title><%= title %></title>
  <link rel="stylesheet" type="text/css"
    href="../public/stylesheets/style.css" />
  <link rel="shortcut icon" href="../public/images/favicon.ico" />
</head>
<body>
  <div class="wrapper">
    <div class="content">
      <%- include("../header.html") %>
```



```

<div class="desktop">
<br>
<h4>Отзывы</h4>
<table>
  <tr>
    <th width="25%">Имя</th>
    <th>Отзыв</th>
  </tr>
  <% for (let item of feeds) { %>
    <tr>
      <td class="td"><%= item.userName %></td>
      <td class="td"><%= item.userFeed %></td>
    </tr>
  <% } %>
</table>
</div>
</div>
<%- include("./footer.html") %>
</div>
</body>
</html>

```

Из примера видно, что часть динамически формируемой html-страницы подгружается из двух отдельных статичных страниц – header.html и footer.html. Дело в том, что эти части страницы одинаковы на всех формах приложения, поэтому их можно без изменений подключать в динамически формируемые страницы приложения.

## 5. Подготовка и защита курсового проекта

*Этапы курсового проектирования:*

- получение задания на выполнение курсового проекта;
- обсуждение этапов проектирования и порядка разработки информационной системы;
- подготовка технического задания на проектирование информационной системы;
- разработка схемы базы данных для информационной системы;
- проектирование модулей информационной системы;
- тестирование и доработка;
- составление отчёта о курсовом проектировании;

- защита курсового проекта.

Рекомендуемый объем работы, содержание Отчёта о курсовом проекте и постраничное описание отражены в таблице 1.

Таблица 1.

Название пункта и рекомендуемое содержание	Рекомендуемый объем, кол-во страниц
Титульный лист	1
Содержание	1
1. Постановка задачи на проектирование информационной системы <i>Рекомендуется кратко описать бизнес процесс и обосновать необходимость проведения автоматизации.</i>	1
2. Анализ технологий проектирования – обзор технологий для хранения данных csv-файлы, json-файлы, базы данных (например, SQLite, MySQL, PostgreSQL); – обзор языков программирования (например, C#, Python, Node.js); – описание паттерна MVC для проектирования приложения; – следует уделить внимание обоснованию выбора технологий для проектирования приложения.	4-8
3. Реализация функционала информационной системы – логическая модель базы данных (схема и описание к ней) или описание структур для хранения данных (csv, json, xml); – описание SQL-запросов к базе данных или методов для обработки csv, json, xml; – интерфейс пользователя – скриншоты программы с описанием как пользоваться, последовательность переходов между окнами; – регистрация и авторизация (при наличии); – описание отдельных, наиболее значимых, классов и методов программной реализации с Листингами кода.	5-12

<i>– хеширование (при наличии) – описание программной реализации с Листингами кода – 2-3 стр.</i>	
Закключение <i>описание что сделано, выводы об эффективности автоматизации и перспективы дальнейшей разработки</i>	1
Список литературы	1
Приложение А. Техническое задание	5-6
Приложение Б. Программный код	5-10
Рецензия <i>рецензию после сдачи отчёта заполнит преподаватель</i>	1

## **Заключение**

Работая на технической реализации информационной системы в рамках курсового проектирования, используя представленные в методических рекомендациях материалы обучающийся дополняет и систематизирует знания, полученные им в рамках контактной работы с преподавателем, что позволит ему полностью освоить компетенции, предусмотренные при изучении дисциплины «Программная инженерия».

В рамках организации курсового проектирования обучающихся должны реализовываться следующие принципы:

- принцип интерактивности обучения (обеспечение интерактивного диалога и обратной связи, которая позволяет осуществлять контроль и коррекцию действий студента);

- принцип развития интеллектуального потенциала (формирование алгоритмического, наглядно-образного, теоретического стилей мышления, умений принимать оптимальные или вариативные решения в сложной ситуации, умений обрабатывать информацию);

- принцип обеспечения целостности и непрерывности дидактического цикла обучения (предоставление возможности выполнения всех звеньев дидактического цикла в пределах темы).

В процессе курсового проектирования обучающийся должен активно использовать электронные библиотечные системы, электронные поисковые системы и электронные периодические справочники. Кроме того, обучающийся должен регулярно использовать Интернет-ресурсы, находящиеся в свободном доступе.

## **Перечень основной и дополнительной литературы**

### **Основная:**

1. Прохоренок, Н. А. JavaScript и Node.js для веб-разработчиков / Н. А. Прохоренок, В. А. Дронов. – СПб: БХВ-Петербург, 2022. – 768 с.
2. Полуэктова, Н. Р. Разработка веб-приложений: учебное пособие для вузов / Н. Р. Полуэктова. – Москва: Издательство Юрайт, 2022. – 204 с. – (Высшее образование). – ISBN 978-5-534-13715-6. – Текст: электронный // Образовательная платформа Юрайт [сайт]. – URL: <https://urait.ru/bcode/496682> (дата обращения: 27.11.2022).
3. Молинаро Э. SQL/ Сборник рецептов. – 2-ое изд.: пер. с англ. / Э. Молинаро, Р. де Граф. – СПб: БХВ-Петербург, 2022. – 592 с.
4. Маркин, А.В. Программирование на SQL: учебное пособие для среднего профессионального образования / А.В. Маркин. – Москва: Издательство Юрайт, 2022. – 435 с. – (Профессиональное образование). – ISBN 978-5-534-11093-7. – Текст : электронный // Образовательная платформа Юрайт [сайт]. – URL: <https://urait.ru/bcode/495666> (дата обращения: 27.11.2022).

### **Дополнительная:**

1. Лаврищева, Е. М. Программная инженерия и технологии программирования сложных систем: учебник для вузов / Е. М. Лаврищева. – 2-е изд., испр. и доп. – Москва: Издательство Юрайт, 2022. – 432 с. – (Высшее образование). – ISBN 978-5-534-07604-2. – Текст : электронный // Образовательная платформа Юрайт [сайт]. – URL: <https://urait.ru/bcode/491029> (дата обращения: 27.11.2022).
2. Гниденко, И. Г. Технологии и методы программирования: учебное пособие для вузов / И. Г. Гниденко, Ф. Ф. Павлов, Д. Ю. Федоров. – Москва: Издательство Юрайт, 2022. – 235 с. – (Высшее образование). – ISBN 978-5-534-02816-4. – Текст:

электронный // Образовательная платформа Юрайт [сайт]. – URL: <https://urait.ru/bcode/489920> (дата обращения: 27.11.2022).

3. Лаврищева, Е. М. Программная инженерия. Парадигмы, технологии и CASE-средства: учебник для вузов / Е. М. Лаврищева. – 2-е изд., испр. – Москва: Издательство Юрайт, 2022. – 280 с. – (Высшее образование). – ISBN 978-5-534-01056-5. – Текст: электронный // Образовательная платформа Юрайт [сайт]. – URL: <https://urait.ru/bcode/491048> (дата обращения: 27.11.2022).

4. Малов, А. В. Концепции современного программирования: учебное пособие для вузов / А. В. Малов, С. В. Родионов. – Москва: Издательство Юрайт, 2022. – 96 с. – (Высшее образование). – ISBN 978-5-534-14911-1. – Текст : электронный // Образовательная платформа Юрайт [сайт]. – URL: <https://urait.ru/bcode/485436> (дата обращения: 27.11.2022).

### **Базы данных, информационно-справочные и поисковые системы**

1. Электронный каталог библиотеки Пермского ГАТУ: базы данных, содержащие сведения обо всех видах литературы, поступающей в фонд Научной библиотеки Пермского ГАТУ. – URL: <https://pgsha.ru/generalinfo/library/webirbis/>.
2. Электронная библиотека / Пермский государственный аграрнотехнологический университет имени академика Д. Н. Прянишникова. – URL: <https://pgsha.ru/generalinfo/library/elib/>.
3. ConsultantPlus (КонсультантПлюс) : компьютерная справочно-правовая система. – URL: <https://www.consultant.ru/>. – Режим доступа: для авторизированных пользователей. – Доступ из корпусов ПГАТУ.
4. eLIBRARY.RU : научная электронная библиотека. – URL: <https://elibrary.ru/defaultx.asp>. – Режим доступа: для зарегистрированных пользователей.
5. Polpred.com (Полпред.ком) : электронно-библиотечная система. – URL: <https://polpred.com/news>. – Режим доступа:

для зарегистрированных пользователей.

6. IPRSMART : электронно-библиотечная система. – URL: <https://www.iprbookshop.ru/>. – Режим доступа: для зарегистрированных пользователей.

7. Гребенникон : электронная библиотека. – URL: <https://grebennikon.ru/>. – Режим доступа: для зарегистрированных пользователей.

8. Руконт : национальный цифровой ресурс : межотраслевая электронная библиотека. – URL: <https://lib.rucont.ru/search>. – Режим доступа: для зарегистрированных пользователей.

9. Лань : электронно-библиотечная система. – URL: <https://e.lanbook.com/>. – Режим доступа: для зарегистрированных пользователей.

10. Юрайт : электронно-библиотечная система. – URL: <https://urait.ru/>. – Режим доступа: для зарегистрированных пользователей.

11. Сетевая электронная библиотека (СЭБ). – URL: <https://e.lanbook.com/>. – Режим доступа: для зарегистрированных пользователей.

12. Электронные информационные ресурсы ФГБНУ ЦНСХБ. – Режим доступа: для авторизированных пользователей. – Доступ из интернет-зала главного корпуса.

13. **Перечень открытых интернет-ресурсов:**

Интуит - Открытый университет:

– Курс «Введение в стандарты WEB»: [https://intuit.ru/studies/professional\\_skill\\_improvements/1432/info](https://intuit.ru/studies/professional_skill_improvements/1432/info)

– Курс «Web-технологии»:

[https://intuit.ru/studies/professional\\_skill\\_improvements/1252/info](https://intuit.ru/studies/professional_skill_improvements/1252/info)

\_\_\_\_\_  
наименование организации – разработчика ТЗ на АС

**УТВЕРЖДАЮ**

Руководитель \_\_\_\_\_  
(должность, наименование предприятия – заказчика АС)

\_\_\_\_\_  
Личная подпись

\_\_\_\_\_  
Расшифровка подписи

(печать)

Дата \_\_\_\_\_

**УТВЕРЖДАЮ**

Руководитель \_\_\_\_\_  
(должность, наименование предприятия – разработчик АС)

\_\_\_\_\_  
Личная подпись

\_\_\_\_\_  
Расшифровка подписи

(печать)

Дата \_\_\_\_\_

\_\_\_\_\_  
наименование вида АС

\_\_\_\_\_  
наименование объекта автоматизации

\_\_\_\_\_  
сокращённое наименование АС

**ТЕХНИЧЕСКОЕ ЗАДАНИЕ**

На \_\_\_\_\_ листах

Действует с \_\_\_\_\_

202\_\_



## 1. Общие сведения

### 1.1 Наименование системы

Полное наименование разрабатываемой системы – «**Название Вашего приложения**».

Краткое наименование – «**Краткое название Вашего приложения**».

### 1.2 Наименование заказчика и исполнителя

Организация: ФГБОУ ВО Пермский ГАТУ.

Адрес: ул. Петропавловская, 23.

Телефон: +7 (342) 217-90-66;

Исполнитель: **Фамилия Имя Отчество**.

### 1.3 Плановые сроки начала и окончания работ

Дата начала работ: **\_\_.\_\_.20\_\_**.

Дата окончания работ: **\_\_.\_\_.20\_\_**.

## 2. Назначение и цели создания системы

К целям создания **приложения «Отзывы на продукт»** можно отнести (**далее самостоятельно укажите пункты, которые характеризуют Ваше приложение**):

- улучшение оперативного взаимодействия и интеграция модулей;
- автоматизировать опрос сотрудников и студентов университета.

Достижение целей приведёт к следующим положительным результатам:

- руководитель сможет быстрее оценивать оперативную картину видя назначенные мероприятия и поручения в едином потоке информации;
- благодаря возможности комментировать информационную публикацию в модуле «Живая лента» отправители избавятся от необходимости получать обратную связь от исполнителей или участников мероприятий через модуль «Сообщения» или «Чат» каждый раз при проведении мероприятия или назначении поручения;
- кураторам или старостам будет удобнее информировать свои группы студентов при помощи информационных публикаций или проводить опросы;
- пользователи портала смогут видеть информационные публикации в отсортированном в порядке убывания даты.

### 3. Характеристика объекта автоматизации

Объектом автоматизации является подсистема организационной коммуникации интернет-сайта по продажам продукции.

### 4. Требования к системе

Общие требования к **приложению «Отзывы на продукт»** являются:

- надёжность и работоспособность;
- интуитивно понятный интерфейс;
- лицензионная чистота – применение средств в рамках общего лицензионного соглашения касательно корпоративного портала;
- соблюдение информационной безопасности и разграничение прав доступа к данным.

#### 4.1 Требования к способам и средствам связи для информационного обмена между компонентами

Для обеспечения информационного обмена компоненты подсистемы должны взаимодействовать с объединённой информационной базой данных. Благодаря хранению данных в различных схемах, таких как продукция или отзывы **приложение «Отзывы на продукт»** может объединить эти данные представив их как единый информационный поток.

##### 4.1.1 Перспективы развития, модернизация системы

Дальнейшим развитием **приложения «Отзывы на продукт» модуля «Живая лента»** может быть объединение иных подсистем интернет-сайта.

##### 4.1.2 Требования к квалификации персонала и режиму его работы

Для обеспечения максимальной работоспособности пользователей должны устанавливаться перерывы:

- через 2 часа после начала смены и через 1,5–2 часа продолжительностью 15 минут;
- через каждый час работы продолжительностью 10 минут

Для эксплуатации **приложения «Отзывы на продукт»** определены следующие роли:

- системный администратор – должен быть квалифицированным специалистом с практическим опытом выполнения работ по администрированию программных и технических средств. В обязанности входит: установка, модернизация, настройка программного обеспечения, ведение учётных записей портала;

– администратор баз данных – должен быть квалифицированным специалистом с практическим опытом выполнения работ по администрированию СУБД, проектированию БД, оптимизации производительности, разграничению прав и ролей, а также резервного копирования и обеспечение целостности БД;

– пользователь интернет-сайта – должен иметь опыт работы с персональным компьютером на уровне начинающего пользователя и свободно осуществлять базовые действия по добавлению или изменению информации посредством браузера с доступом в интернет.

#### **4.1.3 Требования к надёжности технических средств и программного обеспечения**

Надёжность по отношению к техническим средствам должна обеспечиваться использованием в системе средств повышенной отказоустойчивости и их резервированием, а также дублированием носителей информационных банков данных.

Надёжность программного комплекса обеспечивается использованием сертифицированных операционных систем, общесистемных программных средств и инструментальных программных систем, используемых при разработке программного обеспечения. Само программное обеспечение должно обеспечивать защиту от некорректных действий пользователей и ошибочных исходных данных.

#### **4.1.4 Требования к безопасности**

Разрабатываемый информационный **приложения «Отзывы на продукт»** должен обеспечивать безопасный доступ к данным, предотвращая несанкционированный доступ или модифицирование данных. Модуль аутентификации должен обеспечивать защищённый доступ ко всему программному интерфейсу приложения, за исключением статичной формы авторизации в системе предоставляя возможность пройти аутентификацию в корпоративном портале при помощи логина и пароля.

Также при разработке модуля необходимо соблюдать разграничение прав на публикацию информации отправителю должны быть доступны только те адресаты, которые относятся к его зоне ответственности.

#### **4.1.5 Требования по эргономике и технической эстетике**

Модуль должен иметь удобный и интуитивно понятный графический пользовательский интерфейс. Диалоговый интерфейс должен соблюдать контекст подсистемы организационной коммуникации университета и управления в целом, тем самым действия конечного пользователя должны быть ясны и знакомы.

Пользовательский интерфейс модуля также должен аккомпанировать цветовой гамме и общему стилю корпоративного портала.

#### **4.1.6 Требования к программному обеспечению**

При проектировании приложения «Отзывы на продукт» необходимо эффективно использовать веб-фреймворк Express.js, в качестве серверного окружения – программную платформу Node.js, а для хранения данных СУБД PostgreSQL (MySQL, SQLite – по выбору разработчика).

#### **4.1.7 Требования к техническому обеспечению**

Техническое обеспечение системы должно максимально и наиболее эффективно использовать существующее в отделе автоматизации оборудование:

- процессор – 2x Intel Xeon 3.7 ГГц;
- оперативная память – 32 ГБ;
- дисковая система – 1ТБ;
- сетевой адаптер – 1 Гб/с.

### **5. Порядок контроля и приёмки системы**

Приёмо-сдаточные испытания системы проводятся с привлечением сотрудников отдела автоматизации. По результатам опытной эксплуатации оформляется акт о приёме работ. Акт содержит заключение о соответствии системы техническому заданию.

#### **5.1 Требования к составу и содержанию работ подготовки объекта автоматизации к вводу системы в действие**

При подготовке к вводу в эксплуатацию приложения «Отзывы на продукт» отдел управления информатизации должен обеспечить выполнение следующих работ:

- определить подразделение и ответственных должностных лиц для внедрения информационного модуля;
- обеспечить пользователей кратким руководством, которое поможет быстрее освоить внедрённый модуль, или интуитивно понятным интерфейсом;
- провести опытную эксплуатацию приложения «Отзывы на продукт».

Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное  
учреждение высшего образования  
«Пермский государственный аграрно-технологический университет  
имени академика Д. Н. Прянишникова»

Кафедра Информационных технологий  
и программной инженерии

## ОТЧЁТ ПО КУРСОВОМУ ПРОЕКТУ

на тему: «Разработка web-приложения по работе с отзывами  
для сайта по продаже *ТУТ НАЗВАНИЕ НА ВАШ ВЫБОР*»

Выполнил:

студент группы ПИБ-xxxx  
направления подготовки  
09.03.03 Прикладная информатика  
Иванов Иван Иванович

Проверил:

доцент кафедры ИТиПИ, к.т.н., доцент  
Беляков Андрей Юрьевич

Пермь – 20\_\_

РЕЦЕНЗИЯ

на проект \_\_\_\_\_


Рецензент: \_\_\_\_\_  
 \_\_\_\_\_

Учебное издание  
**ПРОГРАММНАЯ ИНЖЕНЕРИЯ**

Методические рекомендации по курсовому проектированию

Автор  
**Беляков Андрей Юрьевич**

Подписано в печать \_\_. \_\_. 2022.

Формат 60х84/16.

Усл.печ.л. 3,7. Заказ № \_\_\_\_\_. Тираж 50 экз.

*ФГБОУ ВО Пермский ГАТУ*

Пермского государственного аграрно-технологического университета  
имени академика Д.Н. Прянишникова,  
614990, г. Пермь, ул. Петропавловская, 23 тел. (342) 210-35-34