Беляков А.Ю.

<https://t.me/AndreyPerm>

Ссылка на репозиторий с материалами к курсовому проекту:

<https://github.com/permCoding/practice-nodejs/tree/main/2023/CourseProject>

В репозитории: Шаблон Отчёта, Пример Отчёта,

пример программы, ссылки на полезные материалы.

Методическое пособие

**по курсовому проектированию**

Тема: Разработка web-приложения

для автоматизации бизнес-процесса предприятия

Содержание

|  |  |
| --- | --- |
| Введение | 2 |
| Разработка информационной системы | 6 |
| Используемые технологии | 6 |
| Архитектура приложения | 6 |
| Дерево директорий приложения | 7 |
| Интерфейс пользователя | 7 |
| Результаты курсового проектирования | 8 |
| Постраничное описание и содержание Отчёта | 9 |
| Приложение А. Техническое задание | 10 |
| Приложение Б. Титульный лист | 15 |
|  |  |

**Введение**

Данное курсовое проектирование посвящено рассмотрению автоматизации бизнес-процессов и выполнению этапов, связанных с разработкой программной реализации.

Этапы автоматизации бизнес-процесса:

* исследуем предприятие;
* выбираем плохо автоматизированный процесс;
* составляем модели процесса «как-есть» и «как-будет»;
* **составляем Техническое Задание на разработку информационной системы;**
* **разрабатываем прототип и проводим апробацию;**
* разрабатываем программную реализацию;
* тестируем, внедряем.

Из приведённой последовательности этапов в рамках курсового проектирования будут задействованы:

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

В данную задачу будут интегрированы такие вопросы проектирования и программирования как: архитектура web-приложения, MVC-паттерн проектирования, организация обмена данными между программой и ресурсами, расположенными в сети Интернет, регистрация и авторизация пользователя, передача параметров и данных между формами приложения, хеширование паролей пользователей.

**Разработка информационной системы**

В рамках курсового проектирования будет разработано web-приложение "**Рейтинг абитуриентов**".

**Используемые технологии:**

|  |  |
| --- | --- |
| * Nodejs * Express * hbs * MVC * json | * request * SQLite * html * css * lodash |

**Архитектура приложения** основана на паттерне проектирования **MVC** (Model-View-Controller).

Составные части паттерна:

**model** - модель - это часть архитектуры приложения:

- 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 логика приложения.

**view** - представление - это часть архитектуры приложения (примеры представления: HTML-страница, WPF форма, Windows Forms):

- 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:

- совокупность обработчиков событий представления

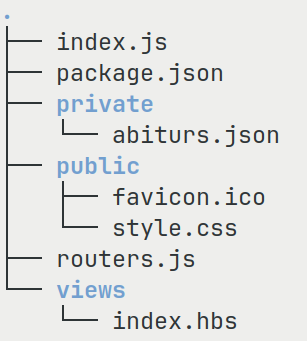
- принимает события от представления

- отдаёт событие на обработку в модель

- возвращает пользователю обновлённое представление

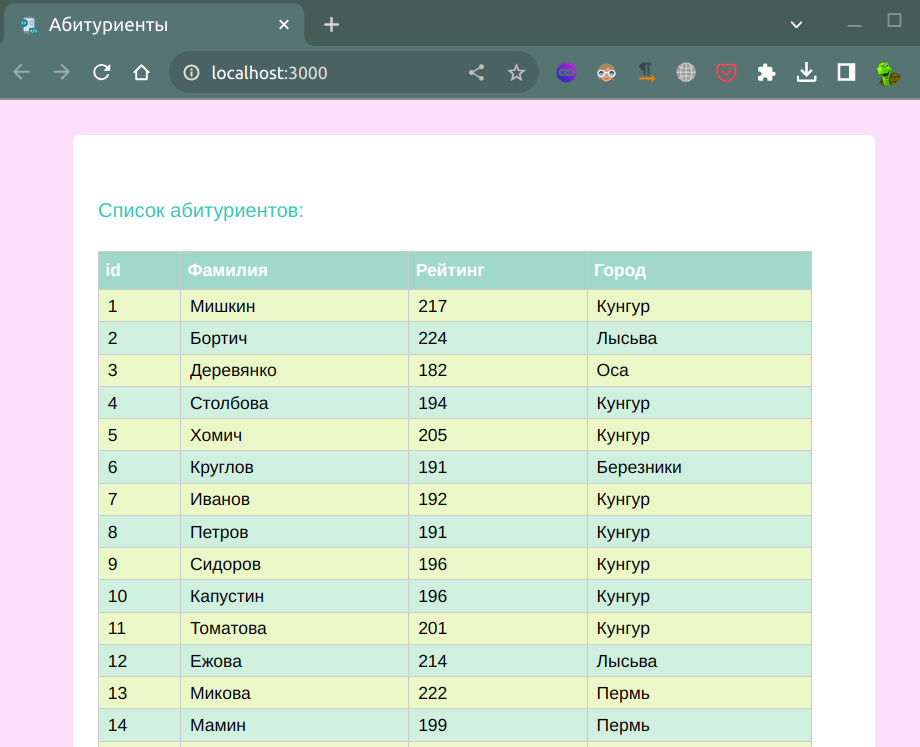
- или выбирает какое именно представление должно быть отображено в ответ.

**Дерево файлов и директорий** приложения:



**Организация интерфейса пользователя** и функционал визуальных форм приложения.

Главная форма приложения:



**Результаты курсового проектирования**

**Закрепление умений и навыков 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 SQL-запросы с параметрами;
* хешировать пароли;
* запускать методы по обработке данных синхронно или асинхронно.

**По результатам курсового проектирования** нужно сдать:

* ***файлы с данными*** (csv, json, xml) *или* ***базу данных*** с таблицами и связями (для SQLite это может быть файл \*.sql) – преподаватель во время постановки задачи на курсовое проектирование указывает как хранить данные;
* программную реализацию с функционалом (или ссылка на репозиторий студента - автора);
* отчёт об учебной практике (электронный и бумажный вариант).
* ***репозиторий*** *(не для всех групп обязательно, по указанию преподавателя) создаёт студент и туда размещает программный код, данные для работы приложения и Отчёт, ссылку присылает преподавателю* ***через чат в Портале Университета****;*
* ***отчёт электронный*** *(\*.docx, \*.pdf) поместить в репозиторий заблаговременно для предварительной проверки и указания замечаний;*
* ***отчёт бумажный*** *(с учтёнными замечаниями) принести на кафедру (преподавателю или в лаборантскую).*

**Постраничное описание и содержание Отчёта:**

* Титул – 1 стр.
* Содержание – 1 стр.
* Постановка задачи на проектирование информационной системы – 1 стр.

описание что именно требуется разработать, в каком формате хранить и обрабатывать данные, как сортировать и фильтровать, куда выводить результаты.

* Анализ технологий проектирования

обзор технологий для хранения данных csv-файлы, json-файлы, yaml-файлы, xml-файлы, базы данных (например, SQLite, MySQL, PostgreSQL) – 2-3 стр.

обзор языков программирования (например, C#, Python, Node.js) – 2-3 стр.

описание паттерна MVC – 1-2 стр.

* Реализация функционала информационной системы

логическая модель базы данных (схема и описание к ней) или описание структур для хранения данных (csv, json, xml, yaml) – 1 стр.

описание методов для обработки csv, json, xml, yaml или SQL-запросов к базе данных – 1-2 стр.

интерфейс пользователя – скриншоты программы с описанием как пользоваться, последовательность переходов между окнами – 1-3 стр.

регистрация и авторизация (*при наличии*) – описание программной реализации с Листингами кода – 2-3 стр.

хеширование (*при наличии*) – описание программной реализации с Листингами кода – 2-3 стр.

* Заключение – описание что сделано, выводы об эффективности автоматизации и перспективы дальнейшей разработки – 1 стр.
* Список источников (5-10 источников, можно интернет-ресурсы) – 1 стр.
* Приложение А. Техническое задание. – 5-7 стр.
* Приложение Б. Программный код. – 5-10 стр.

Приложение А

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
2. Наименование системы

Полное наименование разрабатываемой системы – «Название Вашего приложения».

Краткое наименование – «Краткое Название Вашего приложения».

1. Наименование заказчика и исполнителя

Организация: ФГБОУ ВО Пермский ГАТУ.

Адрес: ул. Петропавловская, 23.

Телефон: +7 (342) 217-90-66;

Исполнитель: \_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.

1. Плановые сроки начала и окончания работ

Дата начала работ: \_\_.\_\_.202\_.

Дата окончания работ: \_\_.\_\_.202\_.

1. Назначение и цели создания системы

К целям создания приложения «Рейтинг абитуриентов» можно отнести:

* улучшение оперативности получения данных о поступающих в вуз;
* автоматизация обработки данных по разным критериям;
* отказ от использования бумажных способов хранения и обработки информации.

1. Характеристика объекта автоматизации

Объектом автоматизации является подсистема хранения и обработки данных абитуриентов, поступающих в высшее учебное заведение.

1. Требования к системе

Общие требования к приложению «Рейтинг абитуриентов» 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.

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 приложение «Рейтинг абитуриентов» может объединить эти данные, представив их как единый информационный поток.

1. Перспективы развития, модернизация системы

Дальнейшим развитием приложения «Рейтинг абитуриентов» может быть объединение иных подсистем интернет-сайта.

1. 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 «Рейтинг абитуриентов» 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.

1. 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.

1. Требования к безопасности

Разрабатываемое приложение «Рейтинг абитуриентов» должно 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.

1. 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.

1. Требования к программному обеспечению

При проектировании приложения «Рейтинг абитуриентов» 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).

1. Требования к техническому обеспечению

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

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

1. 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.

1. Требования к составу и содержанию работ подготовки объекта автоматизации к вводу системы в действие

При подготовке к вводу в эксплуатацию приложения «Рейтинг абитуриентов» отдел управления информатизации должен обеспечить выполнение следующих работ:

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

Приложение Б

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 информационной системы

для автоматизации бизнес-процесса»

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 – 202\_\_