МИНИСТЕРСТВО ОБРАЗОВАНИЯ ОРЕНБУРГСКОЙ ОБЛАСТИ **ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ПРОФЕССИОНАЛЬНОЕ ОБРАЗОВАТЕЛЬНОЕ**

«**ОРЕНБУРГСКИЙ КОЛЛЕДЖ ЭКОНОМИКИ И ИНФОРМАТИКИ**»

**(ГАПОУ СПО ОКЭИ)**

**ОТЧЁТ**

*ОКЭИ 09.02.07.70 23. 11*

МДК 02.02 Инструментальные средства разработки программного обеспечения

Разработал: *Панов Е.М.*

Руководитель: *Вареников Л.А.*

**Лабораторная работа №4**

Тема: построение архитектуры программного средства.

Цель: реализация начальных этапов процесса разработки программного средства в соответствии с ГОСТ Р ИСО/МЭК 12207.

Ход работы

1. Введение

Основание для разработки:

* договор с заказчиком №12345 от 15.01.2025;
* утверждённый бизнес-план фотостудии;
* результаты анализа потребностей рынка услуг автоматизации фотостудий.

3. Назначение разработки

ИС должна обеспечивать выполнение следующих функций:

а) Для клиентов:

1. онлайн-запись на фотосессии через веб-интерфейс;
2. просмотр свободных временных слотов и выбор фотографа;
3. получение уведомлений о записи (email, SMS);
4. доступ к личному кабинету с историей заказов.

б) Для сотрудников:

1. управление расписанием работы фотографов;
2. ведение базы данных клиентов и заказов;
3. контроль выполнения заказов (статусы: «в работе», «готово»);
4. возможность загрузки готовых фотографий в систему для передачи клиенту.

в) Для администрации:

1. формирование отчётов по доходам, загруженности сотрудников и популярности услуг;
2. управление ценами на услуги и акциями;
3. мониторинг работы системы (логи действий пользователей).

4.2. Требования к надёжности

1. Общая концепция архитектуры

Архитектура программного средства для информационной системы (ИС) «Фотостудия» строится на основе трёхуровневой архитектуры. Она включает в себя:

* уровень представления (Frontend) отвечает за взаимодействие с пользователем;
* уровень бизнес-логики (Backend) обрабатывает запросы и реализует основную функциональность;
* уровень данных (Database) обеспечивает хранение и управление данными.

2. Уровни архитектуры

2.1. Уровень представления

Обеспечивает взаимодействие пользователя с системой через графический интерфейс.

Технологии: «HTML», «CSS», «JavaScript» («React» или «Vue.js»).

Основные компоненты:

* личный кабинет клиента с функцией записи на фотосессии;
* панель администратора для управления расписанием, заказами и сотрудниками;
* мобильная адаптация интерфейса для доступа с устройств «Android» и «iOS».

Функции:

* регистрация и авторизация пользователей;
* просмотр доступных временных слотов и фотографов;
* отображение статусов заказов и загрузка готовых фотографий.

2.2. Уровень бизнес-логики (Backend)

Реализует обработку данных, управление бизнес-логикой и взаимодействие между клиентом и базой данных.

Технологии: «Python» («Django» или «Flask»), «Node.js» (опционально для обработки асинхронных задач).

Основные компоненты:

* модуль управления заказами: запись, изменение, удаление;
* модуль работы с фотографиями: загрузка, хранение, передача клиенту;
* интеграция с внешними API: платёжные системы («Stripe», «PayPal»), SMS/email-уведомления.

Функции:

* обработка запросов от Frontend;
* управление расписанием сотрудников;
* обработка платежей и генерация отчётов.

2.3. Уровень данных (Database)

Хранит данные о клиентах, заказах, расписании сотрудников и обработанных фотографиях.

Технологии: «PostgreSQL» или «MySQL» для реляционных данных, «AWS S3» или аналогичные сервисы для хранения изображений.

Основные таблицы базы данных представлены в таблице 1.

Таблица 1 － Таблицы БД

|  |  |
| --- | --- |
| Таблица | Описание |
| clients | Данные о клиентах (ID, имя, контакты). |
| orders | Информация о заказах (ID заказа, клиент, дата, статус). |
| employees | Данные о сотрудниках (ID, имя, роль). |
| schedule | Расписание работы сотрудников. |
| photos | Ссылки на загруженные фотографии. |

1. Взаимодействие уровней

Пользовательский запрос: клиент через интерфейс Frontend отправляет запрос на запись на фотосессию.

Обработка запроса: Backend принимает запрос через «REST API» или «GraphQL», проверяет доступность времени в базе данных.

Ответ пользователю: Backend возвращает подтверждение через Frontend с деталями записи.

Хранение данных: информация о заказе сохраняется в таблице «orders», а данные о клиенте — в таблице «clients».

1. Технические требования

Требования представлены:

* масштабируемость, поддержка до 100 одновременных пользователей;
* надёжность, резервное копирование базы данных ежедневно;
* безопасность, использование HTTPS для передачи данных; шифрование паролей клиентов.

5. Особенности реализации

Модуль уведомлений: отправка SMS/email клиентам о статусе заказа.

Модуль аналитики: формирование отчётов по загруженности сотрудников и популярности услуг.

Интеграция с облачными сервисами: хранение фотографий на «AWS S3» или «Google Cloud Storage».

1. Преимущества архитектуры

Преимуществами данной архитектуры являются:

* разделение уровней упрощает поддержку и масштабирование системы;
* использование современных технологий обеспечивает высокую производительность;
* возможность интеграции с внешними сервисами расширяет функционал системы.

Эта архитектура позволяет эффективно автоматизировать процессы фотостудии и обеспечить удобство работы как для клиентов, так и для сотрудников.