

## Введение

Цель учебного проекта заключается в разработке программного обеспечения, предназначенного для автоматизации совместной работы с роботом-официантом и документации к нему. Главной задачей этого проекта является мониторинг заказов и их обработка.

Основная реализация проекта заключается в разработке личного профиля администратора, предназначенного для обработки заказов и данных сотрудников, и сотрудников, предназначенного для обработки заказов, предприятия.

Созданная программа рассчитана на сотрудников предприятия, оказывающие услуги области общественного питания.

Далее приведем краткое описание разделов пояснительной записки.

Первый раздел носит название «Анализ задачи». В нем вы сможете ознакомиться с постановкой задачи, которая включает в себя: исследование предметной области поставленной задачи, определение ее организационно-экономической сущности. Также в этом разделе вы сможете узнать о том, как данная задача решается в настоящее время. Все входные и выходные данные тоже будут описаны в первом разделе. В подразделе «Выбор стратегии разработки и модели жизненного цикла» будет определение стратегии разработки и модели жизненного цикла. В подразделе «Инструменты разработки» будет рассмотрена среда, в которой создается данный проект.

В разделе «Проектирование задачи» будут рассмотрены основные аспекты разработки программного продукта. Здесь можно будет узнать об организации данных в контексте среды разработки. В данном разделе будет четко описан пользовательский интерфейс, составлены алгоритмы процесса обработки информации.

«Реализация задачи» – это третий раздел пояснительной записки, в котором описываются все элементы и объекты, которые будут использованы при реализации данного приложения. В этом разделе будут четко описаны функции пользователя и их структура. Здесь можно будет найти таблицу, в которой будет представлена полная аннотация файлов, используемых в данном проекте.

Четвертый раздел – «Тестирование». В нем будет описано полное и функциональное тестирование данной программы, т.е. будет оттестирован каждый пункт меню, каждая операция, которая выполняется приложением. Будут смоделированы все возможные действия пользователя при работе с программой, начиная от запуска до выхода.

					ТРПО 2-40 01 01.33.33.06.23 ПЗ	Лист
Изм.	Лист	№докум.	Подпись	Дата		3

Пятый раздел – «Руководство пользователя», в котором описана подробная инструкция по использованию этого ПП для будущих пользователей интернет-ресурсом.

«Заключение» будет содержать краткую формулировку задачи, результаты проделанной работы, описание использованных методов и средств, описание степени автоматизации процессов на различных этапах разработки.

В «Литературе» будет приведен список используемых при разработке источников.

В приложениях к пояснительной записке будет приведен листинг программы и необходимые чертежи: диаграмма вариантов использования, модель данных, функциональная модель, диаграмма последовательности.

					ТРПО 2-40 01 01.33.33.06.23 ПЗ	Лист
Изм.	Лист	№ док-м.	Подпись	Дата		4

## 1 Анализ задачи

### 1.1 Постановка задачи

**Наименование задачи:** Разработка программного обеспечения, автоматизирующего совместную работу с роботом-официантом.

**Цель разработки:** создание приложения, которое рассчитано на людей, работающих в предприятиях общественного питания, облегчая им работу.

**Назначение:** для обработки документно-ориентированной информации, предоставляет современный способ хранения и упорядочения данных.

**Предметная область:** это программный продукт, реализующий оформление заказа и занесение его базу данных (далее БД).

**Периодичность использования:** зависит от нужд потребителя, может использоваться ежедневно.

**Источники и способы получения данных:** проект будет содержать личный кабинет, представляющий собой личную информацию, базу данных, в которой занесены все заказы, информация о клиенте, заказе и сотрудников.

Разрабатываемый программный продукт позволит выполнить следующие действия:

- регистрация;
- авторизация;
- редактирование профиля сотрудника;
- загрузка изображения;
- выход из аккаунта;
- просмотр заказа;
- удаление заказа;
- редактирование заказа.

Сотрудник имеет возможность загрузки изображения, просмотр всех оформленных не готовых заказов.

Условно-постоянной информацией в проекте является текстовая информация на главной странице.

Входной информацией является логин, пароль при авторизации; фамилия, имя, отчество, должность, E-mail, город, улица, номер дома, квартира, мобильный телефон, логин, пароли в регистрации; фамилия, имя, отчество, номер стола, перечень заказа, замечание к заказу и его статус в бронировании.

Выходной информацией в проекте является данные, подгружаемые БД, пользователя выводимые в профиле личного кабинета и в таблице, в которой выводится вся необходимая информация о заказе и сотрудниках.

					ТРПО 2-40 01 01.33.33.06.23 ПЗ	Лист
Изм.	Лист	№докум.	Подпись	Дата		5

## 1.2 Выбор стратегии разработки и модели жизненного цикла

Для разработки веб-ресурса следует выбрать стратегию разработки и модель жизненного цикла. Осуществляем выбор посредством составления таблиц.

Выбор модели, метода и подхода разработки программы представлен в таблице 1.

Таблица 1 – Выбор модели жизненного цикла на основе характеристик требований

Критерии категории требований	Каскадная	V-образная	RAD	Инкрементная	Быстрого прототипирования	Эволюционная
1	2	3	4	5	6	7
Являются ли требования к проекту легко определяемыми и реализуемыми?	Да	Да	Да	Нет	Нет	Нет
Могут ли требования быть сформулированы в начале ЖЦ?	Да	Да	Да	Да	Нет	Нет
Часто ли будут изменяться требования на протяжении ЖЦ?	Нет	Нет	Нет	Нет	Да	Да
Нужно ли демонстрировать требования с целью их определения?	Нет	Нет	Да	Нет	Да	Да
Требуется ли проверка концепции программного средства или системы? Можно ли реализовать?	Нет	Нет	Да	Нет	Да	Да
Будут ли требования изменяться или уточняться с ростом сложности системы (программного средства) в ЖЦ?	Нет	Нет	Нет	Да	Да	Да
Нужно ли реализовать основные требования на ранних этапах разработки?	Нет	Нет	Да	Да	Да	Да
Итого:	3	3	4	2	4	4

На основе результатов заполнения таблицы 1 наиболее подходящей являются RAD, быстрого прототипирования и эволюционная модели.

Выбор модели жизненного цикла на основе характеристик команды разработчиков представлен в таблице 2.

Таблица 2 – Выбор модели жизненного цикла на основе характеристик команды разработчиков

Критерии категории команды разработчиков проекта	Каскадная	V-образная	RAD	Инкрементная	Быстрого прототипирования	Эволюционная
Являются ли проблемы предметной области проекта новыми для большинства разработчиков?	Нет	Нет	Нет	Нет	Да	Да
Являются ли инструментальные средства, используемые в проекте, новыми для большинства разработчиков?	Да	Да	Нет	Нет	Нет	Да
Изменяются ли роли участников проекта на протяжении ЖЦ?	Нет	Нет	Нет	Да	Да	Да
Является ли структура процесса разработки более значимой для разработчиков, чем гибкость?	Да	Да	Нет	Да	Нет	Нет
Важна ли легкость распределения человеческих ресурсов проекта? Хватит ли членов команды и согласны ли они?	Да	Да	Да	Да	Нет	Нет
Приемлет ли команда разработчиков оценки, проверки, стадии разработки?	Да	Да	Нет	Да	Да	Да
Итого:	4	4	5	4	3	2

На основе результатов заполнения таблицы 2 наиболее подходящей является RAD модель.

Выбор модели жизненного цикла на основе характеристик коллектива пользователей представлен в таблице 3.

Таблица 3 – Выбор модели жизненного цикла на основе характеристик коллектива пользователей

Критерии категории коллектива пользователей	Каскадная	V-образная	RAD	Инкрементная	Быстрого прототипирования	Эволюционная
Будет ли присутствие пользователей ограничено в ЖЦ разработки?	Да	Да	Нет	Да	Нет	Да
Будут ли пользователи оценивать текущее состояние программного продукта (системы) в процессе разработки?	Нет	Нет	Нет	Да	Да	Да
Будут ли пользователи вовлечены во все фазы ЖЦ разработки? Совет.	Нет	Нет	Да	Нет	Да	Нет
Будет ли заказчик отслеживать ход выполнения проекта?	Нет	Нет	Нет	Нет	Да	Да
Итого:	0	0	2	1	4	2

На основе результатов заполнения таблицы 3 наиболее подходящей является модель быстрого прототипирования.

Выбор модели жизненного цикла на основе характеристик типа проектов и рисков представлен в таблице 4.

Таблица 4 – Выбор модели жизненного цикла на основе характеристик типа проектов и рисков

Критерии категории типов проекта и рисков	Каскадная	V-образная	RAD	Инкрементная	Быстрого прототипирования	Эволюционная
Разрабатывается ли в проекте продукт нового для организации (команды) направления?	Нет	Нет	Нет	Да	Да	Да
Будет ли проект являться расширением существующей системы?	Да	Да	Да	Да	Нет	Нет
Будет ли проект крупно- или среднемасштабным?	Нет	Нет	Нет	Да	Да	Да
Ожидается ли длительная эксплуатация продукта?	Да	Да	Нет	Да	Нет	Да
Необходим ли высокий уровень надежности продукта проекта?	Нет	Да	Нет	Да	Нет	Да
Предполагается ли эволюция продукта проекта в течение ЖЦ?	Нет	Нет	Нет	Да	Да	Да
Велика ли вероятность изменения системы (продукта) на этапе сопровождения?	Нет	Нет	Нет	Да	Да	Да
Является ли график сжатым?	Нет	Нет	Да	Да	Да	Да
Предполагается ли повторное использование компонентов?	Нет	Нет	Да	Да	Да	Да
Являются ли достаточными ресурсы (время, деньги, инструменты, персонал)?	Нет	Нет	Нет	Нет	Да	Да
Итого:	4	5	3	6	4	7

На основе результатов заполнения таблицы 4 наиболее подходящей является эволюционная модель.

Исходя из результатов заполнения таблиц 1-4 (каскадная – 11, V-образная – 12, RAD – 14, инкрементная – 13, быстрого прототипирования – 15, эволюционная – 15) наиболее подходящей моделью, для разработки данного ПП, является быстрого прототипирования и эволюционная модели.

### 1.3 Инструменты разработки

Для разработки данного проекта выбрана среда VS Code, которая является наиболее актуальной средой для создания приложений данного типа. Языки программирования, используемые для разработки: HTML, PHP, MySQL, CSS, JavaScript.

Также для разработки программы необходимы:

- Open Server Panel – нужен для используется веб-разработчиками со всего мира с целью разработки, отладки и тестирования веб-проектов, а также для предоставления веб-сервисов в локальных сетях;

- Google-браузер – нужен для нахождения информации и картинок для теоритической части приложения, фоновых изображений;

- Word 2010 – нужен для написания пояснительной записки, создания html-страниц;

- Figma – нужна для создания UX/UI макетов интерфейса;

- Draw.io – нужен для создания чертижей.

При разработке данного программного продукта был использован компьютер со следующими характеристиками:

- Процессор: 12th Gen Intel(R) Core(TM) i5-12450H 2.00 GHz;

- ОЗУ: 16Gb;

- Память: SDD 512Gb;

- ОС – Windows 10.

					ТРПО 2-40 01 01.33.33.06.23 ПЗ	Лист
Изм.	Лист	№докум.	Подпись	Дата		9

## 2 Проектирование задачи

### 2.1 Организация данных

Проектирование задачи – это очень важный и ответственный этап в разработке любого приложения.

Важным является он вследствие того, что методы, по средствам которых пользователь управляет формами, построены на высокой степени специализации каждого из компонентов.

Необходимым условием при разработке данного приложения является описание организации данных, т.е. логическая и физическая структура данных в контексте среды разработки. В разрабатываемой программе будут использоваться три вида данных.

Первым видом являются данные, которые будут введены разработчиком на этапе реализации задачи. Сюда можно отнести изображения (иконки, кнопки), текст.

Вторым видом данных, используемых в программе, является вводимая пользователем информация. К входной информации можно отнести вводимые пользователем значения, например, ввод текста для выполнения регистрации, авторизации.

Третьим видом данных является результат программы – например, статус заказа, вывод текстовой и графической информации.

Таким образом, организация данных является важной задачей при разработке данной и любой программы.

БД имеет название «datamon». В ней расположено 6 таблиц: users, booking, work, status, personal, images.

Таблица «users» состоит из 3 полей: id, login, pass, acces. Поле «id» отвечает за номер пользователя в БД. Поля «login», «pass», «acces» отвечают за хранение логина, пароля и уровня доступа пользователя.

Таблица «personal» состоит из 11 полей: id, surname, name, patron, post, city, tel, login, pass, acces. Поле «id» отвечает за номер пользователя в БД. Поля «surname», «name», «patron», «city», «tel» отвечают за хранение фамилии, имени, отчества, города, улицы, номера дома и квартиры, а также номера телефона пользователя соответственно.

Таблица «booking» состоит из 10 полей: id, id\_pers, surname, name, patron, numTable, book, book2, cost, status. Поле «id» и «id\_pers» отвечает за номер заказа и сотрудника, обслуживавшего стол, в БД. Поля «numTable», «book», «book2», «status», «cost» отвечают за хранение номера стола, перечня заказа, замечания к заказу, статуса и стоимости заказа. Поле «id\_pers» и «status» представляют собой

					ТРПО 2-40 01 01.33.33.06.23 ПЗ	Лист
Изм.	Лист	№докум.	Подпись	Дата		10



ключи «personal» и «status», по которым будет связана таблица «booking» с другими таблицами.

Таблица «status» состоит из 2 полей: status, name. Поле «status» номер статуса, по которому можно обратиться к этой строке. Поле «name» отвечает за хранение наименования статуса.

Таблица «work» состоит из 15 полей: id, MonW, MonL, TueW, TueL, WedW, WedL, ThuW, ThuL, FriW, FriL, SatW, SatL, SunW, SunL. Поле «id» номер пользователя, по которому можно обратиться к этой строке. Поле «MonW», «MonL», «TueW», «TueL», «WedW», «WedL», «ThuW», «ThuL», «FriW», «FriL», «SatW», «SatL», «SunW», «SunL» отвечает за хранение времени работы и перерыва.

Таблица состоит из 2 полей: id, image. Поле «id» номер сотрудника, по которому можно обратиться к этой строке. Поле «image» отвечает за хранение изображения.

Разработанная модель данных представлена в приложении Б.

## 2.2 Процессы

Согласно всем перечисленным требованиям и указаниям, которые были рассмотрены в разделе «Анализ задачи», было определено, чем конкретно должна заниматься разрабатываемая программа. Главной ее задачей будет являться просмотр заказов и сотрудников и информации о них.

Для реализации задач будут использоваться БД. С помощью БД, например, будет осуществляться хранение данных пользователя и заказов.

Было бы важным на этапе проектирования рассмотреть еще состав и внешний вид используемой формы. Хотя проектирование формы заранее не так принципиально.

Разработанная функциональная модель представлена в приложении В.

## 2.3 Описание внешнего пользовательского интерфейса

Важным при выполнении проекта является организация диалога между пользователем и сайтом. Во многом это зависит от того, как программист разработает данный сайт, какие компоненты будут использованы и какие методы будут автоматизированы.

Особое внимание следует уделить интерфейсу. Разработчик должен так организовать внешний вид своей программы, что бы пользователь понял, что от него требуется. При решении задачи будут использованы компоненты для ввода информации и её вывода, для вставки текста и др.

					ТРПО 2-40 01 01.33.33.06.23 ПЗ	Лист
Изм.	Лист	№докум.	Подпись	Дата		11

Для организации эффективной работы пользователя нужно создать сайт данной предметной области, в которой все компоненты сайта будут сгруппированы по функциональному назначению. При этом необходимо обеспечить удобный графический интерфейс пользователя.

Таким образом, для успешной работы всего проекта в целом следует обеспечить интуитивно понятный интерфейс с приятными цветами и шрифтами.

Разработанная структура навигации по проекту в представлена в приложении А.

Ссылка на Figma (макеты интерфейсов):  
<https://www.figma.com/file/6j12Re9ZN0xvdMCukV6qc3/UX%2FUI?type=design&node-id=0%3A1&mode=design&t=12tI0Sjp2o7d8SfE-1>.

Разработанные UX/UI представлены в приложении Л.

## 2.4 Разработка UML-диаграмм

UML-диграммы нужны для создания «чертежей» программы, схем, которые показывают, как будет устроено программное обеспечение изнутри, — то есть для проектирования. В данном проекте будет представленно 4 UML-диграммы: диаграмма вариантов использования и диаграмма последовательности.

На диаграммах вариантов использования отображается взаимодействие между вариантами использования, представляющими функции системы, и действующими лицами, представляющими людей или системы, получающие или передающие информацию в данную систему.

Разработанная диаграмма вариантов использования представлена в приложении Г.

Диаграммы последовательностей используются для уточнения диаграмм прецедентов, более детального описания логики сценариев использования. На диаграмме последовательности более детально рассмотренно оформление заказа.

Разработанная диаграмма последовательности представлена в приложении Д.

Разработанная диаграмма деятельности позволяет более детально визуализировать функционал в профиле сотрудника.

Разработанная диаграмма деятельности представлена в приложении Е.

Диаграмма классов демонстрирует общую структуру иерархии классов системы в ПП, их коопераций, атрибутов (полей), методов, интерфейсов и взаимосвязей (отношений) между ними.

Разработанная диаграмма классов представлена в приложении Ж.

Диаграмма объектов описывает оформление заказа продукции шеф-повара.

Разработанная диаграмма объектов представлена в приложении К.

					ТРПО 2-40 01 01.33.33.06.23 ПЗ	Лист
Изм.	Лист	№докум.	Подпись	Дата		12

### 3 Реализация

#### 3.1 Руководство программиста

Данный, проект содержит 6 страниц. Далее рассмотрим назначение каждой страницы:

1) Index – главная страница сайта (смотреть приложение М страница 47), для стилизации был использован файл (смотреть приложение М страница 62);

2) Cabinet – содержит личный кабинет пользователя (смотреть приложение М страница 48), для стилизации был использован файл (смотреть приложение М страница 60);

3) Staff – отвечает за страницу «Сотрудники»(смотреть приложение М страница 55), для стилизации был использован файл (смотреть приложение М страница 64);

4) updateStaff — страница формы редактирования сотрудника (смотреть приложение М страница 59), для стилизации был использован файл (смотреть приложение М страница 65);

5) booking — отвечает за страницу «Заказы»(смотреть приложение М страница 51), для стилизации был использован файл (смотреть приложение М страница 60);

6) updateBook — страница формы редактирования заказа (смотреть приложение М страница 58), для стилизации был использован файл (смотреть приложение М страница 65);

7) modal\_staff — модальное окно для добавления(регистрации) сотрудника (смотреть приложение М страница 54), для стилизации был использован файл (смотреть приложение М страница 63), использовался скрипт на языке JavaScript(смотреть приложение М страница 69).

Для связи запросов MySQL и форм HTML был использован язык программирования PHP, который проверял соединение БД, а также примал, записывал и удалял значения запросов БД и полей страниц. Язык программирования JavaScript использовался для сортировки данны в таблице заказов и сотрудников.

#### 3.2 Спецификация веб-приложения

Таблица 5 – Спецификация программы

Имя файла	Назначение
1	2
index.php	Файл главной страницы сайта
cabinet.php	Файл личного кабинета
staff.php	Файл страницы сотрудников
booking.php	Файл страницы заказов

					ТРПО 2-40 01 01.33.33.06.23 ПЗ	Лист
Изм.	Лист	№докум.	Подпись	Дата		13

Продолжение таблицы 5

1	2
modal_staff.php	Файл модального окна регистрации сотрудников
updateBook.php	Файл страницы формы редактирования заказа
updateStaff.php	Файл страницы формы редактирования сотрудника
booking.css	Файл со стилями страницы заказов
cabinet.css	Файл со стилями страницы личного кабинета
header.css	Файл со стилями шапки страниц сайта
index.css	Файл со стилями главной страницы
modal_index.css	Файл со стилями модального окна для авторизации
modal_staff.css	Файл со стилями модального окна для регистрации сотрудника
staff.css	Файл со стилями страницы сотрудников
update.css	Файл со стилями страницы редактирования заказов
updateStaff.css	Файл со стилями страницы редактирования сотрудника
db_connect.php	Файл подключения БД
auth.php	Файл авторизации
delete.php	Файл удаления сотрудника из БД
deleteTable.php	Файл удаления заказа из БД
logout.php	Файл выхода из авторизованного аккаунта
reg.php	Файл регистрации
updateStaff.php	Файл обновления информации сотрудника в БД
updateTable.php	Файл обновления информации заказа в БД
uploadImg.php	Файл загрузки изображения в БД
sorttable.js	Файл сортировки таблицы
ind_mod_auth.js	Файл модального окна
users.ibd	Таблица БД «register»
personal.ibd	Таблица БД «register»
booking.ibd	Таблица БД «register»
status.ibd	Таблица БД «register»
work.ibd	Таблица БД «register»
images.ibd	Таблица БД «register»

## 4 Тестирование

### 4.1 Отчет о результатах тестирования

Проектирование задачи – это очень важный и ответственный этап в разработке любого приложения.

При разработке данной программы многие возникающие ошибки и недоработки были исправлены на этапе реализации проекта. После завершения этапа написания программы было проведено тщательное функциональное тестирование. Функциональное тестирование должно гарантировать работу всех элементов программы в автономном режиме.

В таблице 6 представлены тест-кейсы для проведения функционального тестирования разработанного прототипа.

Таблица 6 — Тесты на использование функций

Идентификатор	Приоритет	Заглавие	Исходная информация	Ожидаемый результат
1	2	3	4	5
P1	C	Переход на страницы сайта. Нажать на ссылку станицы.		Происходит переход на выбранную страницу.
S_Reg	A	Регистрация на сайте 1. Заполнить обязательные поля. 2. Нажать на кнопку регистрации.	Фамилия: Ложкина Имя: Наталья Отчество: Сергеевна Город: Брест Улица: Морозово Дом: 17 Квартира: 4 Моб. тел.: 1236578925 Логин: Nata_23 Пароль: Nata123 Подтв. пароля: Nata123	После заполнения полей и нажатия кнопки, происходит регистрация с переходом на станицу сотрудников.
S_Reg_2	A	Регистрация на сайте 1. Заполнить обязательные поля. 2. Нажать на кнопку регистрации	Фамилия: Иванов Имя: Алексей Отчество: Антонович Страна: Брест Город: Беларусь Улица: Морозово Дом: 17 Квартира: 4 Моб. тел.: +7999123456 Логин: lozhka Пароль: user22 Подтв. пароля: user222	После заполнения полей и нажатия кнопки, переходит на страницу с сообщением об ошибке.
S_Log	C	Авторизация Нажать на ссылку «Войти».		Происходит появление модального окна с формой авторизации.
S_Log_1	A	Авторизация на сайте 1. Ввод логина и пароля. 2. Нажать на кнопку «Войти».	Логин: admin Пароль: admin	1. Происходит вход в аккаунт с переходом на главную станицу сайта.
S_Log_2	A	Авторизация на сайте 1. Ввод логина и пароля. 2. Нажать на кнопку «Войти».	Логин: admin Пароль: Admin	Вход в аккаунт не происходит, т.к. введенный пароль неверный.

Продолжение таблицы 6

1	2	3	4	5
S_Log_3	A	Авторизация на сайте 1. Ввод логина и пароля. 2. Нажать на кнопку «Войти».	Логин: Admin Пароль: admin	1. Происходит вход в аккаунт с переходом на главную страницу сайта.
S_Out	A	Выход из аккаунта 1. Нажать на ссылку «Выйти». Примечание: Cookie-файлы удаляются после выхода		Происходит выход из аккаунта с переходом на главную страницу сайта.
P_Book		Переход на страницу «Заказы» 1. Нажать на кнопку «Заказы» в личном кабинете.		Происходит переход на страницу «Заказы».
S_delBook	A	Удаление заказа 1. Нажать на кнопку «Удалить».		В таблице исчезает выбранный заказ со всей информацией.
P_updateBook	C	Переход на страницу для редактирования 1. Нажать на кнопку «Изменить».		Происходит переход на страницу для редактирования.
S_updateBook		Редактирование заказа 1. Нажать на кнопку «Изменить».	Измененная информация Номер стола: 14 Стоимость: 140	Происходит переход на страницу «Заказы», с изменением данных о заказе.
P_Staff		Переход на страницу «Сотрудники» 1. Нажать на кнопку «Сотрудники» в личном кабинете.		Происходит переход на страницу «Сотрудники».
P_updateStaff	C	Переход на страницу для редактирования 1. Нажать на кнопку «Изменить».		Происходит переход на страницу для редактирования.
S_updateStaff		Редактирование сотрудника 1. Нажать на кнопку «Изменить».	Измененная информация Улица: Шпакова Дом: 3	Происходит переход на страницу «Заказы», с изменением данных о сотруднике.
S_delStaff	A	Удаление сотрудника 1. Нажать на кнопку «Удалить».		В таблице исчезает выбранный сотрудник со всей информацией.

В таблице 7 представлен отчет о результатах проведения функционального тестирования тестирования

Таблица 7 — Отчет о результатах тестирования

Идентификатор	Результат тестирования
1	2
P1	Выполнено успешно
S_Reg	Выполнено успешно

Продолжение таблицы 7

1	2
S_Reg_2	Выполнено успешно

					ТРПО 2-40 01 01.33.33.06.23 ПЗ	Лист
Изм.	Лист	№докум.	Подпись	Дата		16

S_Log	Выполнено успешно
S_Log_1	Выполнено успешно
S_Log_2	Выполнено успешно
S_Log_3	Выполнено некорректно
S_Out	Выполнено успешно
P_Book	Выполнено успешно
S_delBook	Выполнено успешно
P_updateBook	Выполнено успешно
S_updateBook	Выполнено успешно
P_Staff	Выполнено успешно
P_updateStaff	Выполнено успешно
S_updateStaff	Выполнено успешно
S_delStaff	Выполнено успешно

В результате проведения тестирования сайта выяснилось, что все ранее оговоренные функции, были разработаны, а также протестированны. Тесты показали, что некоторые функции работают некорректно, а именно логин не ригистрозависимый, что делает ПП незаконченным, но в работоспособном состоянии. Функции с некорректным результатом тестирования нужно доработать. Также тестирование показало, что приоритетно важные проверки работают удовлетворительно и недопускают сбой при неполной/некорректной регистрации или входа в чужой аккаунт не имея правильного логина и пароля.

## 5 Руководство пользователя

### 5.1 Общие сведения о программном продукте

Цель учебного проекта заключается в разработке программного обеспечения, предназначенного для автоматизации совместной работы с роботом-официантом и документации к нему. Главной задачей этого проекта является мониторинг заказов и их обработка.

Основная реализация проекта заключается в разработке личного профиля администратора, предназначенного для обработки заказов и данных данных сотрудников, и сотрудников, предназначенного для обработки заказов, предприятия,

Созданная программа рассчитана на сотрудников предприятия, оказывающие услуги области общественного питания.

Сотрудники предприятия могут авторизироваться, мониторить заказы и обрабатывать их. Администратор может зарегистрировать новых сотрудников простматривать таблицу заказов и сотрудников, редактировать и удалять по необходимости.

Тестирование проводилось на разных классах ЭВМ и работать с данной программой было комфортно. Программа разработана на ПК со следующими характеристиками:

- процессор AMD Ryzen 7 4700U with Radeon Graphics, 2000 МГц, ядер: 8, логических процессоров: 8
- объем ОЗУ 16Гб;
- графический адаптер AMD Radeon Graphics Processor (0x1636), Advanced Micro Devices, Inc.-совместимый;
- операционная система Windows 10.

### 5.2 Выполнения программы

#### 5.2.2 Инструкции по работе с программой

После входа на сайт сотрудник может произвести авторизацию (рисунок 1).



Рисунок 1 – Главная страница сайта

					ТРПО 2-40 01 01.33.33.06.23 ПЗ	Лист
Изм.	Лист	№докум.	Подпись	Дата		18



Сотрудник может зайти в аккаунт. Для входа в аккаунт нужно ввести логин, пароль и нажать на кнопку «Войти», если все введено правильно, произойдет вход (рисунок 2).

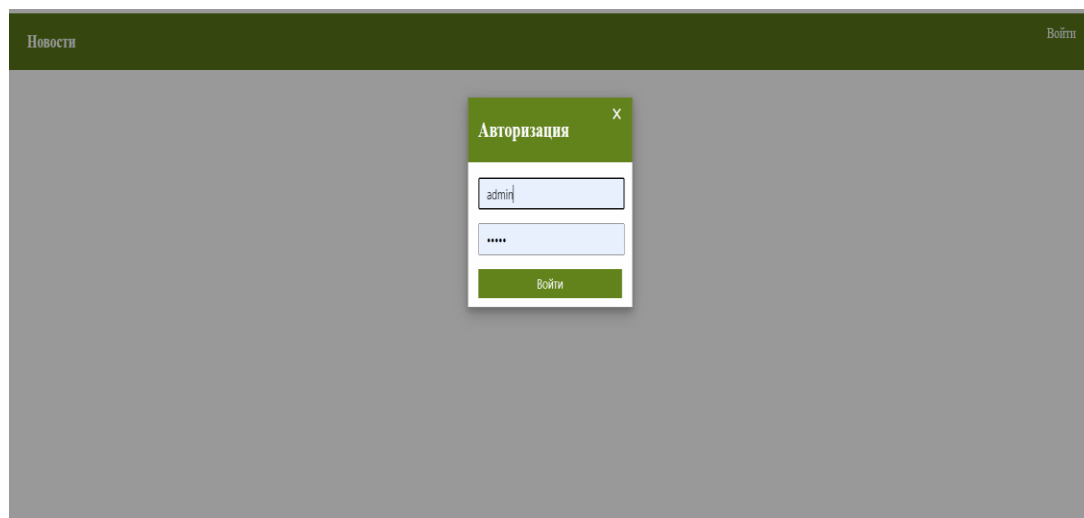


Рисунок 2 – Вход в аккаунт

На странице «Сотрудники» сотрудник может посмотреть список сотрудников, добавить/зарегистрировать сотрудника (рисунок 3), произвести редактирование/изменение сотрудника (рисунок 8) или удалить его (рисунок 7).

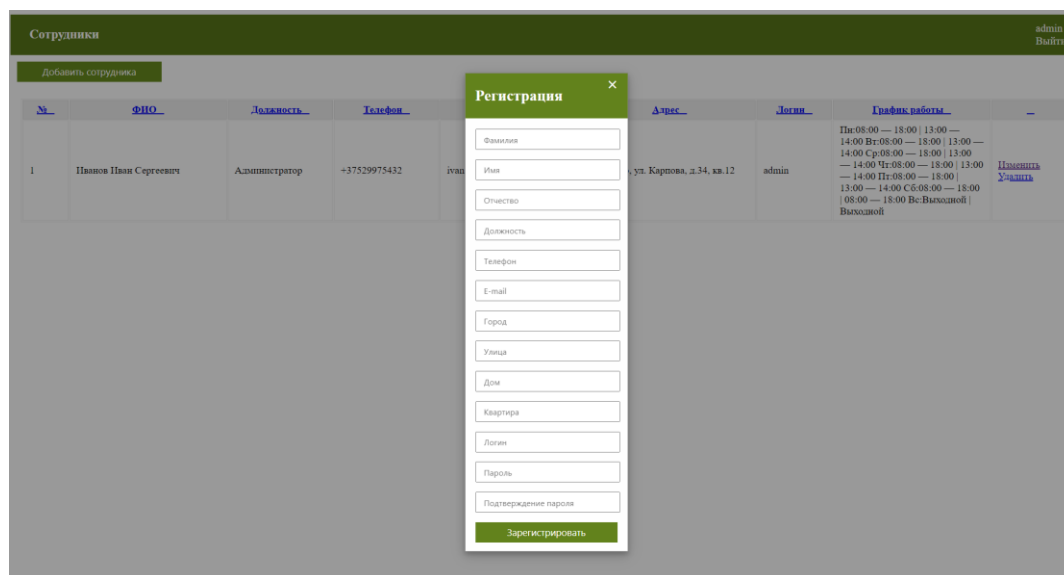



Рисунок 3 – Регистрация на сайте

В личном кабинете сотрудник может посмотреть заказы, нажав на кнопку «Заказы», если уровень доступа соответствуют требованиям, то появляется кнопка

«Сотрудники». Также возможен выход из аккаунта по нажатию на «Выйти» (рисунок 4).

Личный кабинет admin Выйти



Выберите файл | Файл не выбран

День	Время работы	Перерыв
Понедельник	08.00 — 18.00	13.00 — 14.00
Вторник	08.00 — 18.00	13.00 — 14.00
Среда	08.00 — 18.00	13.00 — 14.00
Четверг	08.00 — 18.00	13.00 — 14.00
Пятница	08.00 — 18.00	13.00 — 14.00
Суббота	08.00 — 18.00	09.00 — 18.00
Воскресенье	Выходной	Выходной

Фамилия: Иванов  
Имя: Иван  
Отчество: Сергеевич  
Телефон: +37529975432  
E-mail: ivan123@gmail.com  
Адрес: г. Гродно, ул. Карпова, д.34, кв.12

Рисунок 4 – Личный кабинет

Для удаления заказа необходимо нажать на «Удалить», соответствующего заказа (рисунок 5).

Заказы								admin Выйти
№	ФНО	Номер стола	Перестановка заказа	Примечание	Стоимость	Статус	Обслуживал	
1	Павлов Михаил Маркович	68	Заказ	Примечание к заказу	150 руб.	не готов		<a href="#">Изменить</a> <a href="#">Удалить</a>

Рисунок 5 – Страница «Заказы»

Для редактирования информации заказа необходимо нажать на «Изменить», после его происходить переход на страницу для редактирования заказа (рисунок 6).

Редактирование заказа

Рисунок 6 – Редактирование сотрудника

Для удаления сотрудника необходимо нажать на «Удалить», соответствующего сотрудника (рисунок 7).

Сотрудники								admin Выйти
Добавить сотрудника								
№	ФИО	Должность	Телефон	E-mail	Адрес	Логин	График работы	
1	Иванов Иван Сергеевич	Администратор	+37529975432	ivan123@gmail.com	г. Гродно, ул. Карпова, д.34, кв.12	admin	Пн: 08:00 — 18:00   13:00 — 14:00 Вт: 08:00 — 18:00   13:00 — 14:00 Ср: 08:00 — 18:00   13:00 — 14:00 Чт: 08:00 — 18:00   13:00 — 14:00 Пт: 08:00 — 18:00   13:00 — 14:00 Сб: 08:00 — 18:00   08:00 — 18:00 Вс: Выходной   Выходной	<a href="#">Изменить</a> <a href="#">Удалить</a>

Рисунок 7 – Страница «Сотрудники»

Для редактирования информации сотрудника необходимо нажать на «Изменить», после его происходит переход на страницу для редактирования сотрудника (рисунок 8).

Редактирование сотрудника

Рисунок 8 – Редактирование сотрудника

## Заключение

Цель учебного проекта заключается в разработке программного обеспечения, предназначенного для автоматизации совместной работы с роботом-официантом и документации к нему. Главной задачей этого проекта является мониторинг заказов и их обработка.

Основная реализация проекта заключается в разработке личного профиля администратора, предназначенного для обработки заказов и данных сотрудников, и сотрудников, предназначенного для обработки заказов, предприятия,

Созданная программа рассчитана на сотрудников предприятия, оказывающие услуги области общественного питания. Оно поможет познакомиться с деятельностью и предоставляемыми услугами предприятия, произвести регистрацию, авторизацию и бронирование мест на парковке.

Благодаря этому проекту сотрудникам легче выполнять свою работу и сформировать БД заказов и сотрудников.

В ходе выполнения данной работы было разработан и протестирован сайт «[https:// http://datamon/index.php](https://http://datamon/index.php)».

В качестве дальнейшего развития возможна доработка сайта с целью добавления функционала и улучшения качества работы и оформления сайта.

При разработке программного продукта использовалась декларативная парадигма, как наиболее подходящая. Использование среды разработки VS Code при программной реализации, позволило создать качественное современное программное обеспечение, обладающее высокой скоростью работы, удовлетворяющее всем уровням требований, предъявляемых к интерфейсу пользователя, обеспечивающее эффективность и гибкость работы.

В заключении можно сказать, что данный программный продукт является законченной, полнофункциональной программой.

## Список использованных источников

- 1 PHP Manual [Электронный ресурс]. – Электронные данные. – Режим доступа: <https://www.php.net/manual/en/index.php>
- 2 htmlbook [Электронный ресурс]. – Электронные данные. – Режим доступа: <https://htmlbook.ru/html/div>
- 3 Нормоконтроль
- 4 Изучаем PHP 7. Руководство по созданию интерактивных веб-сайтов (2017) [Электронный ресурс]. – Электронные данные. – Режим доступа: <https://library-it.com/web/php-web/izuchaem-rnr-7-rukovodstvo-po-sozdaniyu-interaktivnyh-veb-sajtov-2017/>
- 5 Создаем динамические веб-сайты с помощью PHP, MySQL, JavaScript, CSS и HTML5. 2е издание (2016) [Электронный ресурс]. – Электронные данные. – Режим доступа: <https://library-it.com/web/php-web/sozdaem-dinamicheskie-veb-sajty-s-pomoshyu-php-mysql-javascript-css-i-html5-2e-izdanie-2016/>
- 6 Тестирование программного обеспечения. Базовый курс [Электронный ресурс]. – Электронные данные. – Режим доступа: [https://skillbox.ru/media/code/15\\_knig\\_po\\_testirovaniyu\\_programmnogo\\_obespecheniya/?authuser=0](https://skillbox.ru/media/code/15_knig_po_testirovaniyu_programmnogo_obespecheniya/?authuser=0)

					ТРПО 2-40 01 01.33.33.06.23 ПЗ	Лист 23
Изм.	Лист	№ док-м.	Подпись	Дата		

**Приложение А**  
Структурное проектирование ПО

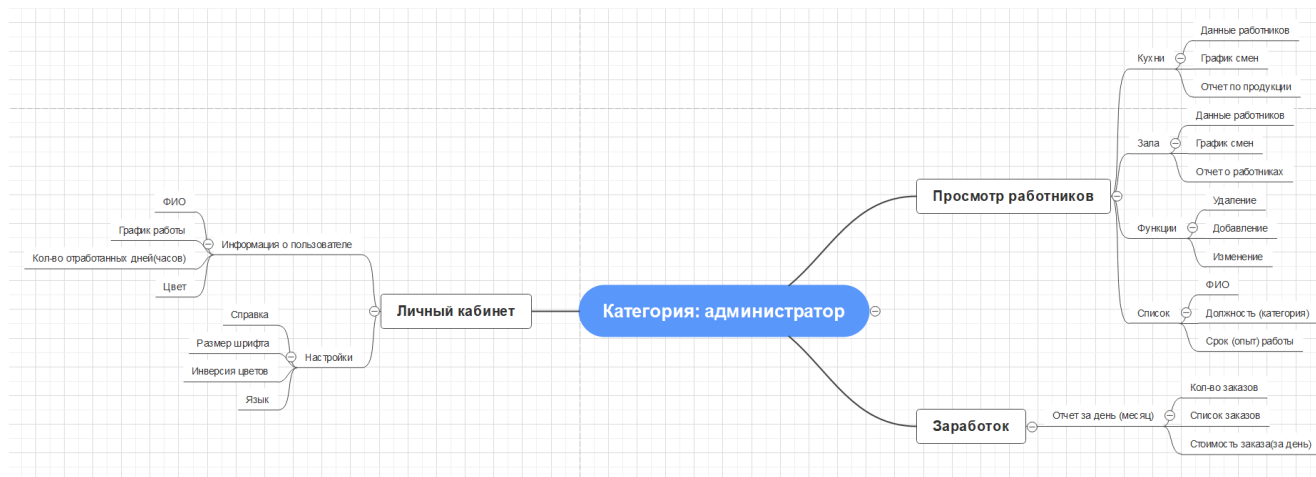


Рисунок А.1 – Структурное проектирование ПО

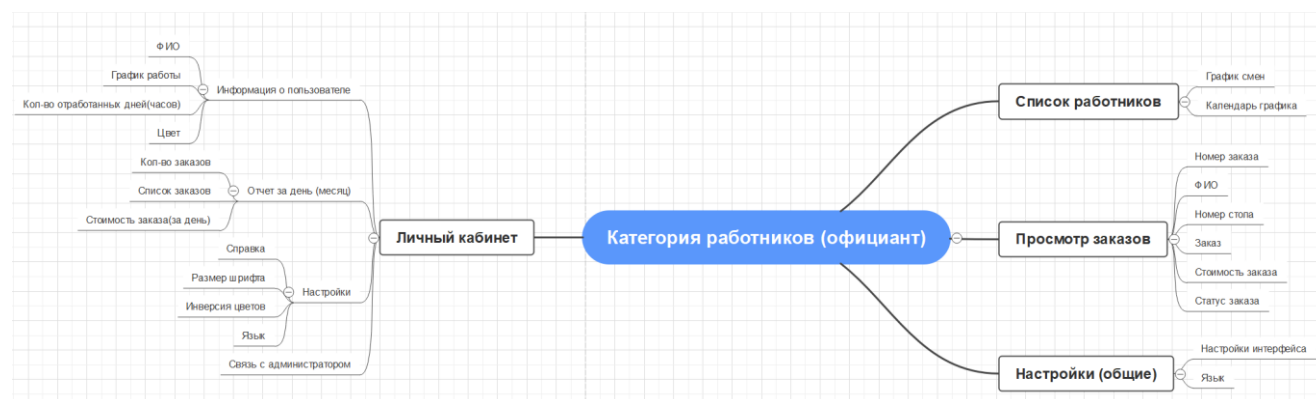


Рисунок А.2 – Структурное проектирование ПО

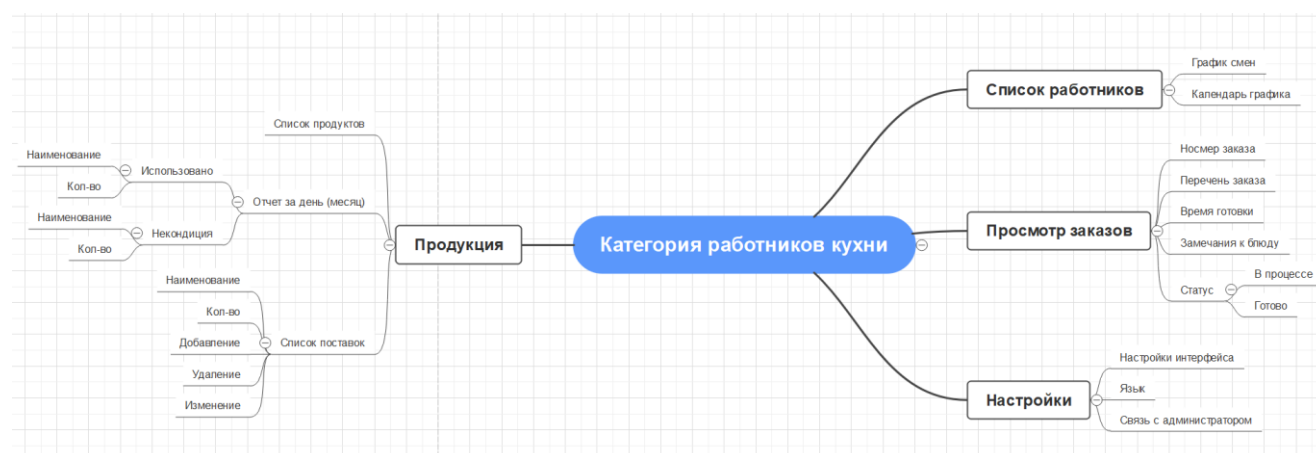


Рисунок А.3 – Структурное проектирование ПО

**Приложение Б**  
Модель данных



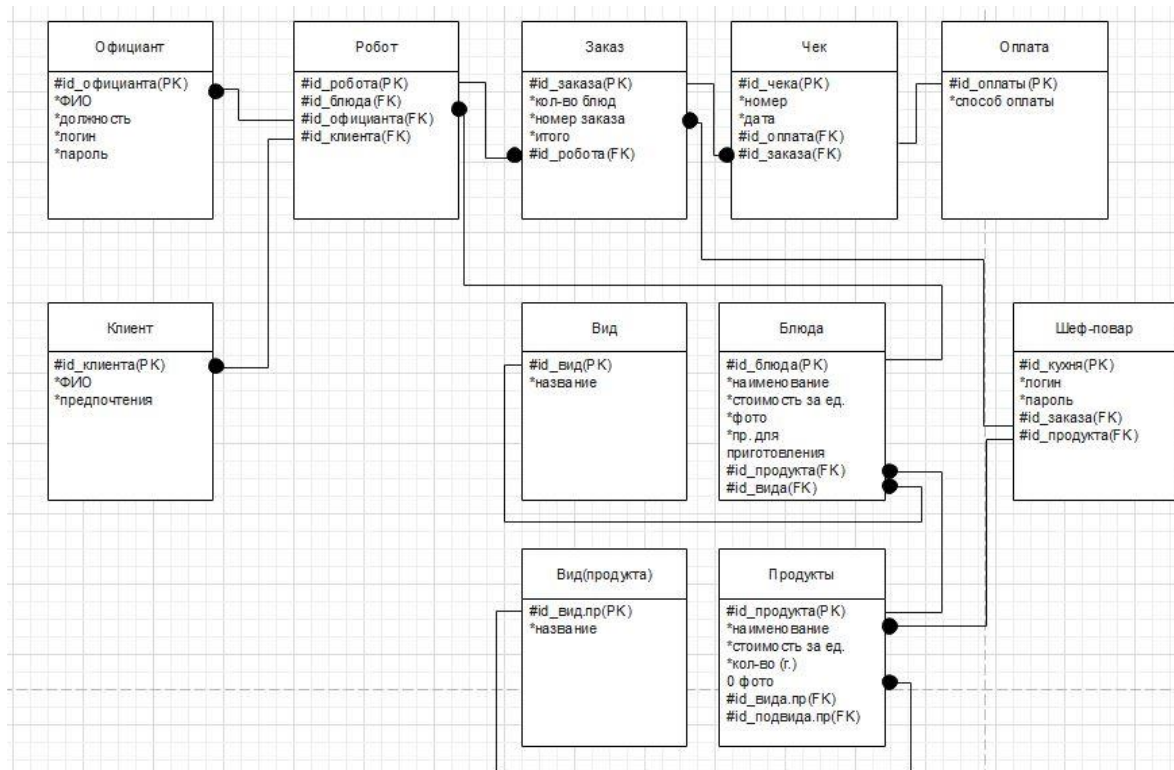


Рисунок Б.1 – Модель данных

**Приложение В**  
Функциональная модель

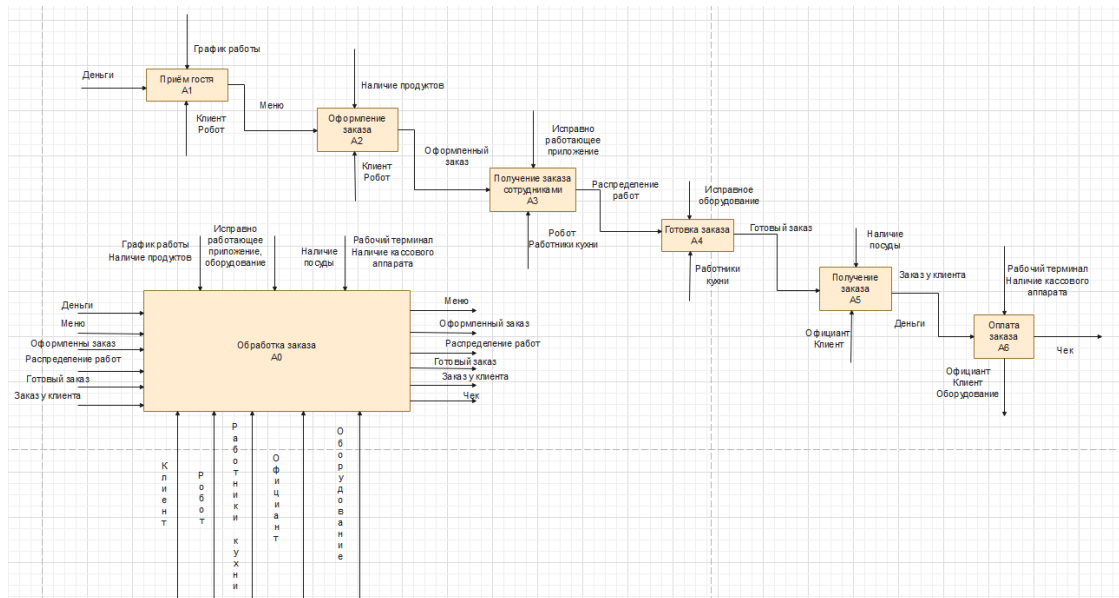


Рисунок В.1 – Функциональная модель

**Приложение Г**  
Диаграмма вариантов использования

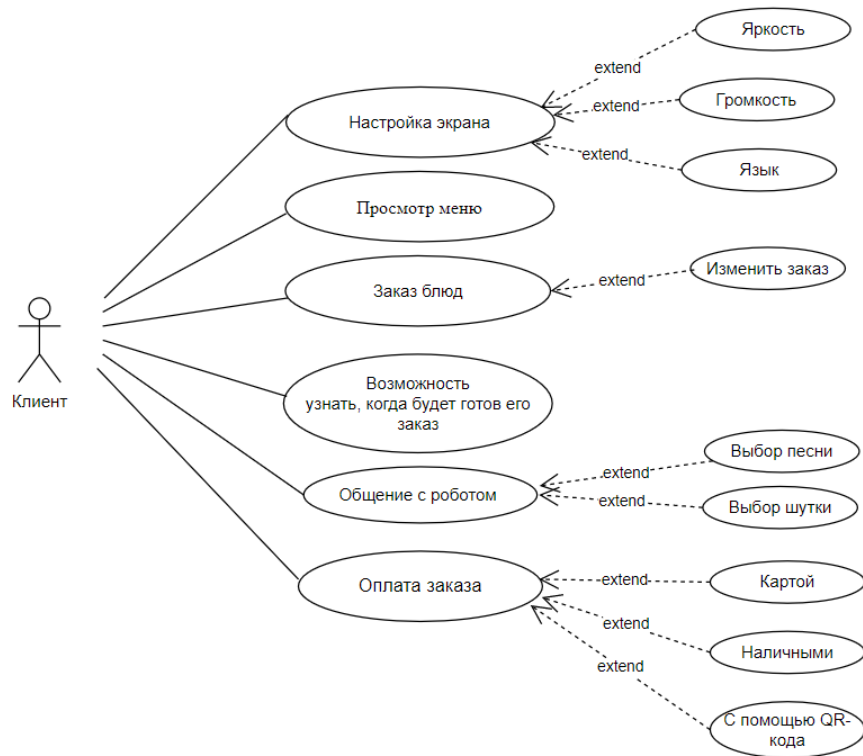


Рисунок Г.1 – Диаграмма вариантов использования (Клиент)

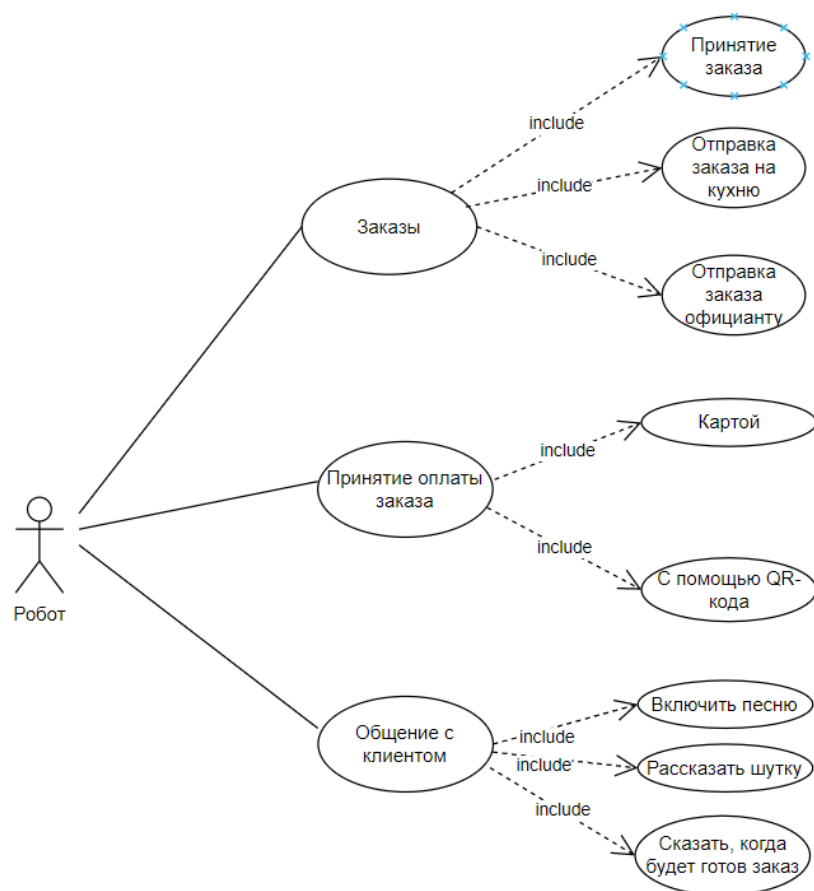


Рисунок Г.2 – Диаграмма вариантов использования (Робот)

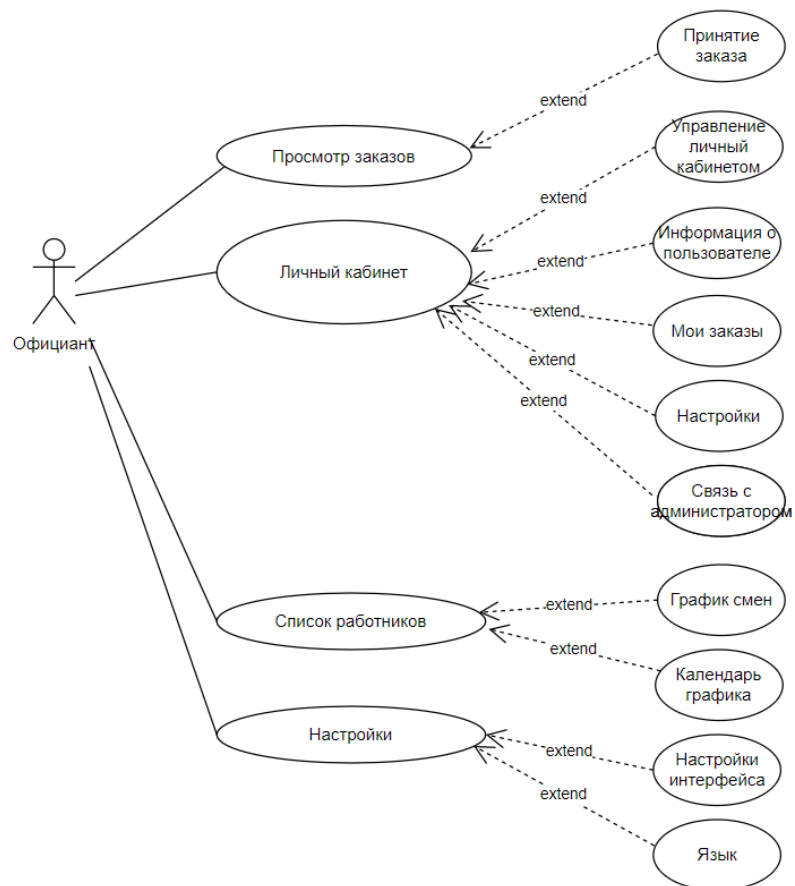


Рисунок Г.3 – Диаграмма вариантов использования (Официант)

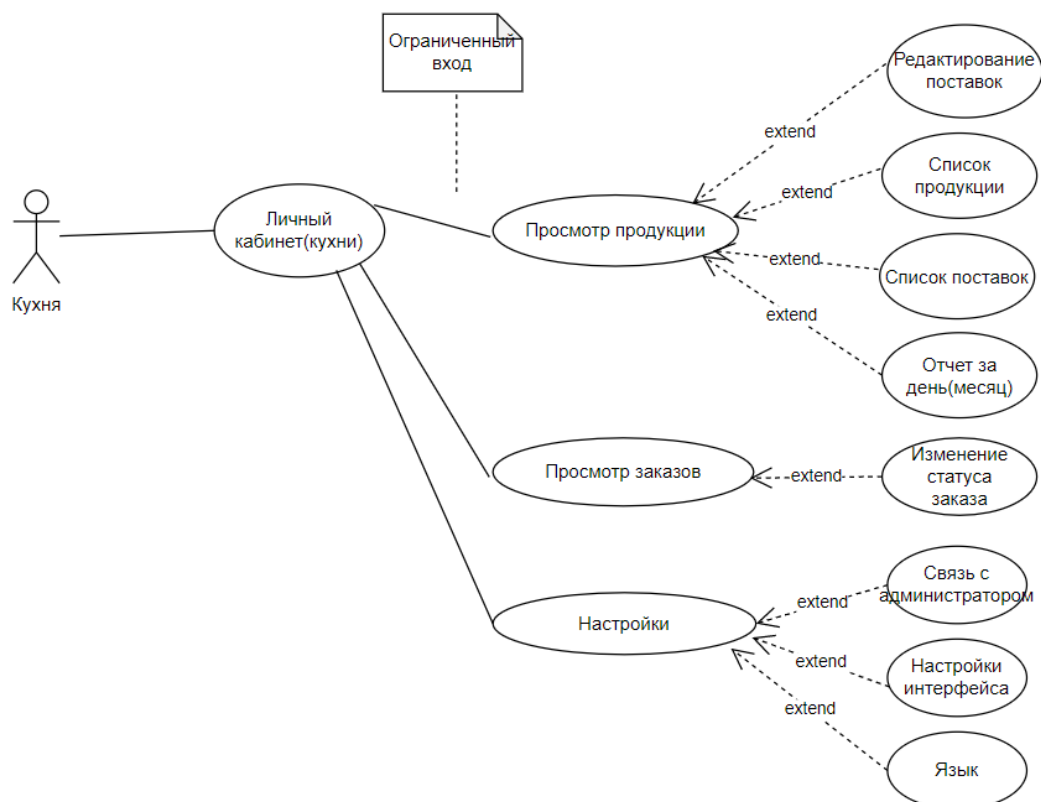


Рисунок Г.4 – Диаграмма вариантов использования (Кухня)

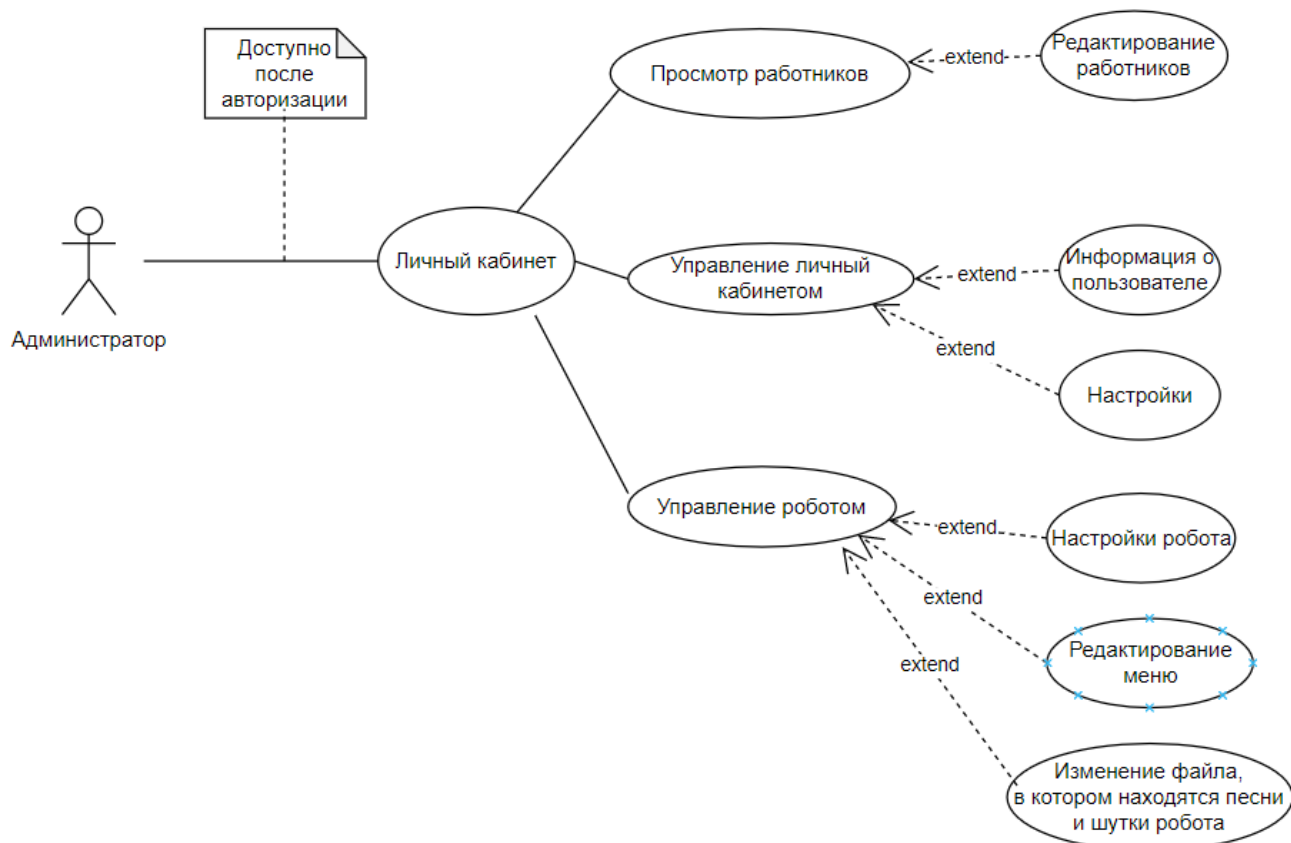


Рисунок Г.5 – Диаграмма вариантов использования (Администратор)

**Приложение Д**  
Диаграмма последовательности



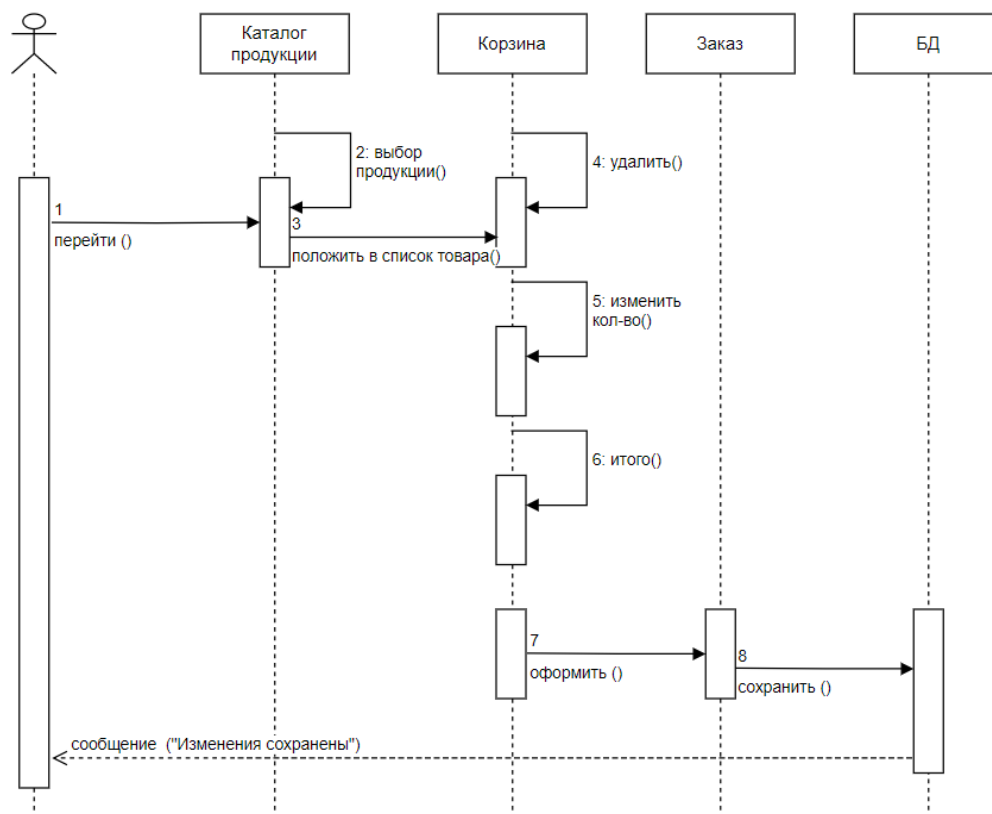


Рисунок Д.1 – Диаграмма последовательности

**Приложение Е**  
Диаграмма деятельности

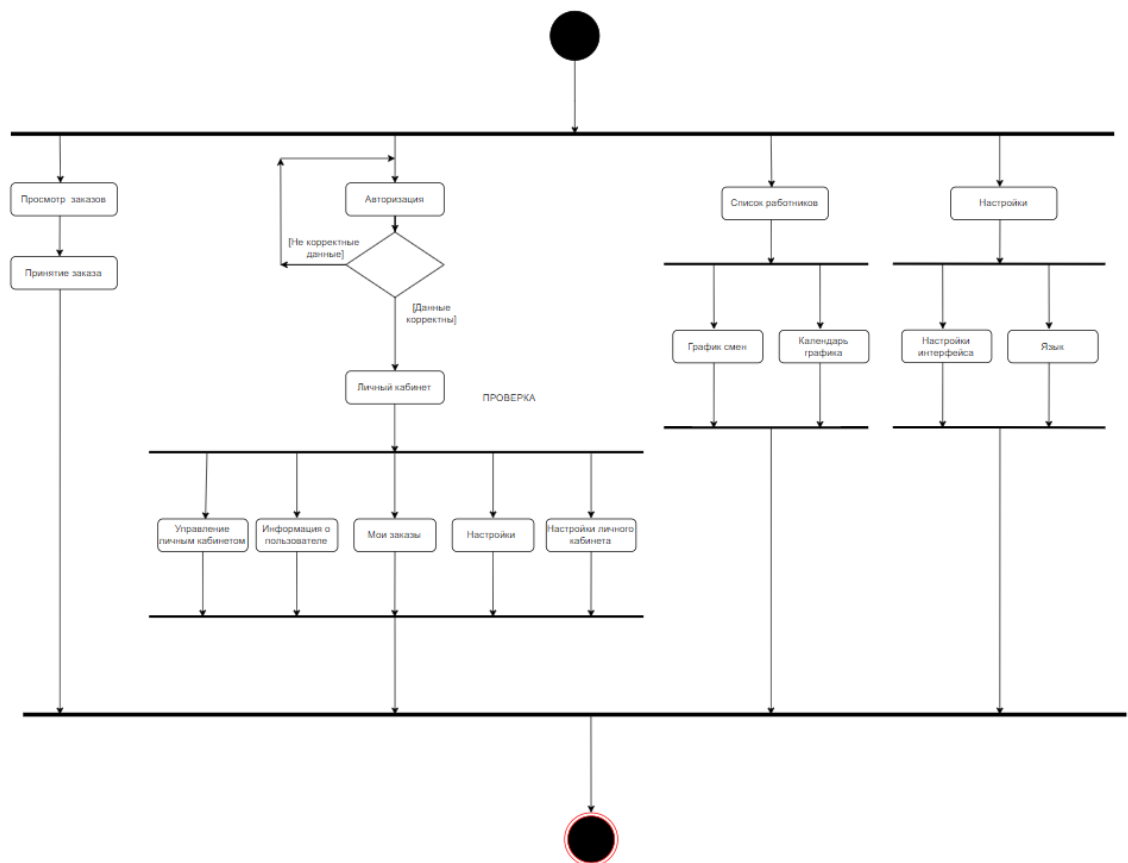


Рисунок Е,1 – Диаграмма деятельности

**Приложение Ж**  
Диаграмма классов

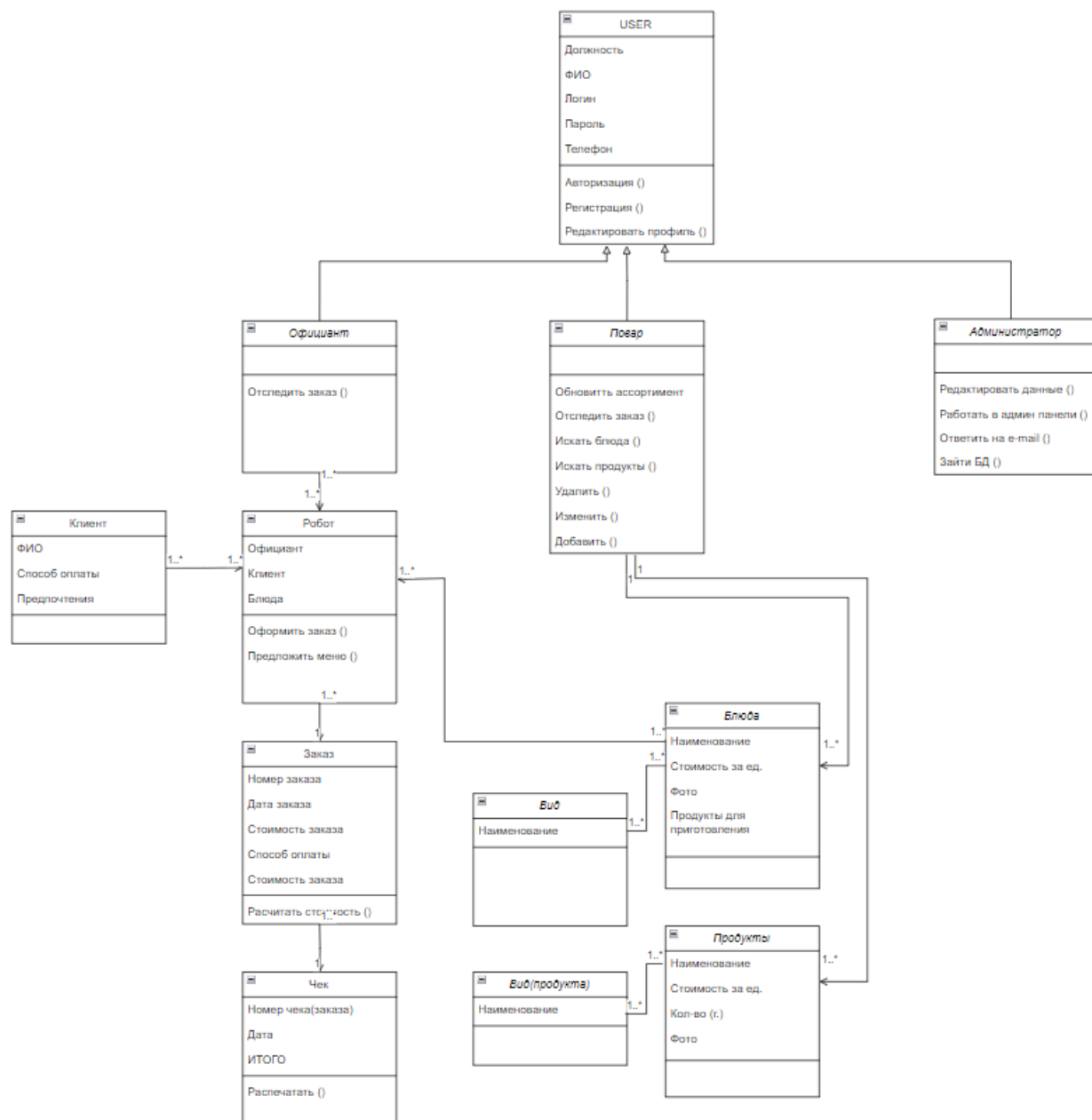


Рисунок Ж.1 – Диаграмма классов

**Приложение К**  
Диаграмма объектов

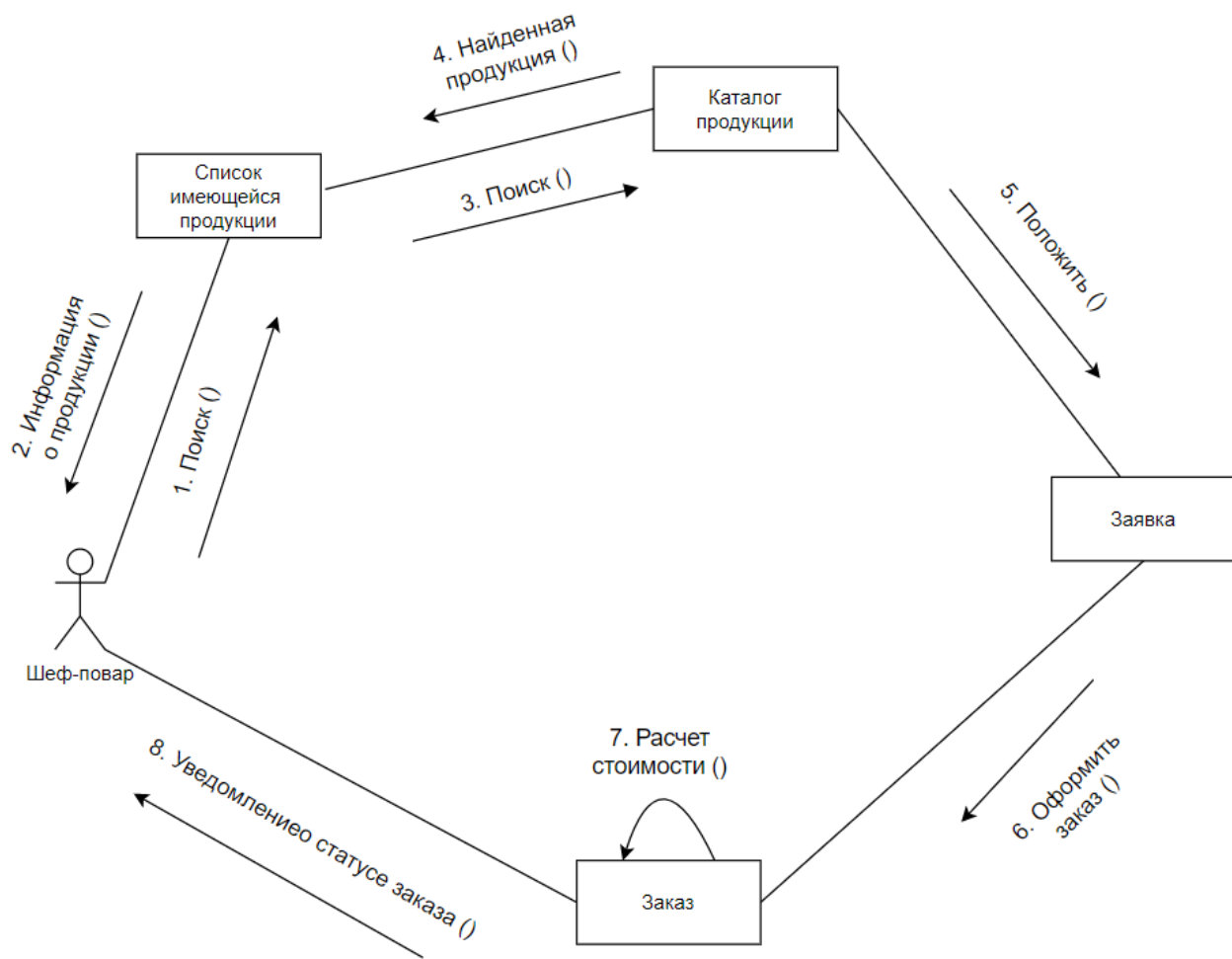


Рисунок К.1 – Диаграмма объектов

Приложение Л  
UX/UI макеты

					ТРПО 2-40 01 01.33.33.06.23 ПЗ	Лист
Изм.	Лист	№ док.м.	Подпись	Дата		42



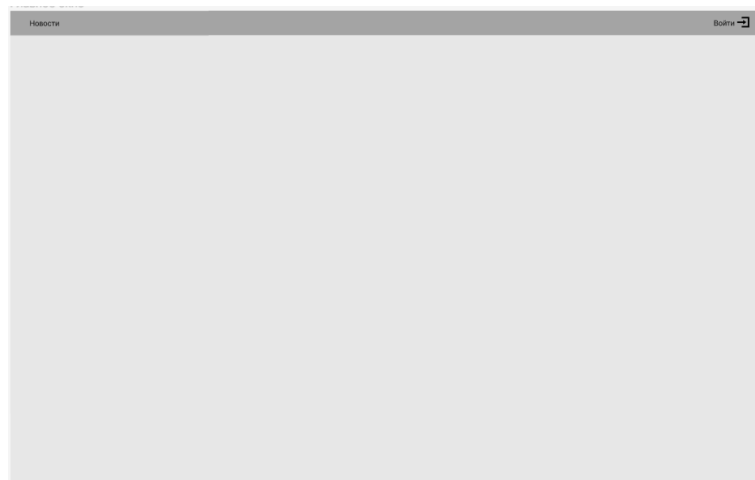


Рисунок Л.3 – UX макет главной страницы



Рисунок Л.4 – UI макет главной страницы

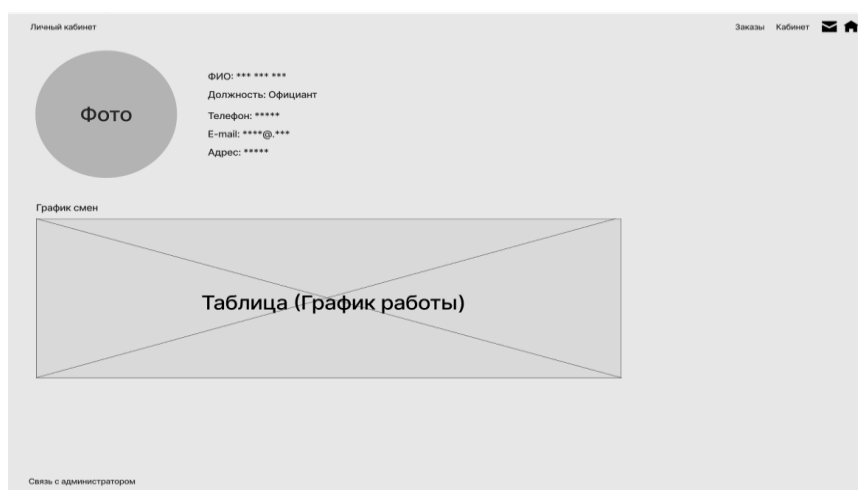


Рисунок Л.3 – UX макет личного кабинета

					ТРПО 2-40 01 01.33.33.06.23 ПЗ	Лист
Изм.	Лист	№докум.	Подпись	Дата		43



Рисунок Л.4 – UI макет личного кабинета

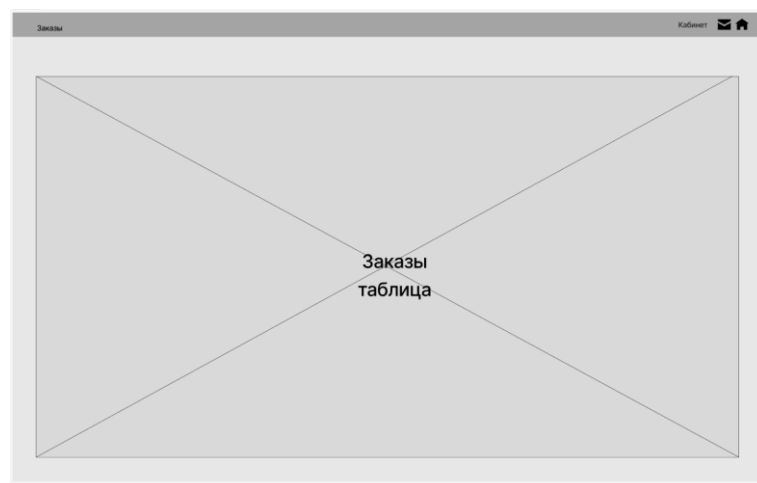


Рисунок Л.5 – UX макет страницы заказов

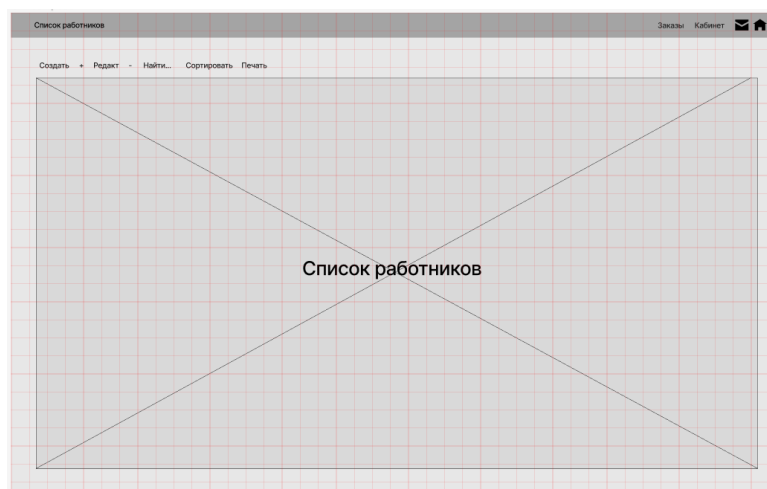


Рисунок Л.6 – UX макет страницы сотрудников



**Приложение М**  
Листинг программы

## Файлы гипертекстовой разметки страницы

### index.php // Файл главной страницы

```
<?php
include "php/db_connect.php";?>
<!DOCTYPE html>
<html lang="ru">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Главная</title>
  <link rel="stylesheet" href="/css/index.css">
  <link rel="stylesheet" href="/css/modal_index.css">
</head>
<body>
  <div class="header">
    <div class="header_client_section">
      <div class="header_section">
        Новости
      </div>
      <div class="clientData headerButton">
        <div class="headerButton">
          <?php
            session_start();
          ?>
          <?php if ($_SESSION['users'] == "") : ?>
            <a class="a1" id="myBtn">Войти</a>
          <?php else : ?>
            <a class="a1" href="/cabinet.php">
              <?php
                if ($_SESSION['users']) {
                  echo ($_SESSION['users']['login']);
                }
              ?>
            </a>
          <?php endif ?>
        </div>
        <div class="headerButton">
          <?php if (!$_SESSION['users'] == "") : ?>
            <a class="a1" href="php/logout.php">Выйти</a>
          <?php endif ?>
        </div>
      </div>
    </div>
    <!-- Модальное окно -->
    <div id="myModal" class="modal">
      <!-- Модальное содержание -->
      <div class="modal-content">
        <div class="modal-header">
          <span class="close">&times;</span>
          <h2>Авторизация</h2>
        </div>
        <div class="modal-body">
          <form action="php/auth.php" method="post" class="form" id="add-form">
            <div class="input-box">
              <input type="text" name="login" class="log_input" placeholder="Логин" required>
            </div><!-- required    onsubmit="return false;" -->
          </form>
        </div>
      </div>
    </div>
  </div>
</body>
</html>
```

					ТРПО 2-40 01 01.33.33.06.23 ПЗ	Лист
Изм.	Лист	№ док.м.	Подпись	Дата		47

```

        <div class="input-box">
            <input type="password" name="pass" class="log_input" placeholder="Пароль" required>
        </div>
        <div class="log_but" >
            <button type="submit" class="log_but_1 log">Войти</button>
        </div>
    </form>
</div>
<script defer src="/js/ind_mod_auth.js"></script>
</div>
</div>
</body>
</html>

```

### **Cabinet.php // Файл страницы личного кабинета пользователя**

```

<?php
include "php/db_connect.php";
session_start();
$login = $_SESSION['users']['login'];
$query = mysqli_query($mysqli, "SELECT * FROM `users` WHERE `login` = '{ $login }'");
$id = mysqli_fetch_assoc($query);
// print_r($id);

$Pquery = mysqli_query($mysqli, "SELECT * FROM `personal` WHERE `id` = '{ $id['id'] }'");
$row = mysqli_fetch_assoc($Pquery);
//print_r($row);?>
<!DOCTYPE html>
<html lang="ru">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Личный кабинет</title>
    <link rel="stylesheet" href="/css/cabinet.css">
    <link rel="stylesheet" href="/css/modal_cabinet.css">
    <link rel="stylesheet" href="/css/header.css">
</head>
<body>
    <div class="header">
        <div class="header_client_section">
            <div class="header_section">
                Личный кабинет
            </div>
            <div class="clientData">
                <div class="headerButton">
                    <?php
                    session_start();
                    ?>
                    <?php if ($_SESSION['users'] == "") : ?>
                        <a class="a1" href="/login.php">Войти</a>
                    <?php else : ?>
                        <!--
                        <a class="a1" href="/index.php">Главная</a> -->
                        <a class="a1" href="/index.php">
                            <?php
                            if ($_SESSION['users']) {
                                echo ($_SESSION['users']['login']);

```

					ТРПО 2-40 01 01.33.33.06.23 ПЗ	Лист
Изм.	Лист	№докум.	Подпись	Дата		48

```

    }
    ?>
  </a>
  <?php endif ?>
</div>
<div class="headerButton">
  <?php if ($_SESSION['users'] == "") : ?>
    <!--<a class="a1" href="/register.php">Регистрация</a -->
  <?php else : ?>
    <a class="a1" href="php/logout.php">Выйти</a>
  <?php endif ?>
</div>
</div>
</div>
</div>
<div class="information">
  <div class="image_pers">
    <?php
      $query_img = mysqli_query($mysqli, "SELECT * FROM `images` WHERE `id`= '{$_id['id']}'");
      $img = mysqli_fetch_assoc($query_img);
      // print_r($img);
      $show_img = base64_encode($img['image']);
    ?>
    
    <div class="function">
      <form action="php/uploadImg.php" method="post" enctype="multipart/form-data">
        <input type="file" name="img_upload"><input type="submit" name="upload"
value="Загрузить">
      </form>
    </div>
  </div>
</div>
<div class="m2">
  <div>
    <div class="row">
      <div class="lef"><label>Фамилия</label></div>
      <div class="rig"><?php echo ($row['surname']); ?></div>
    </div>
    <div class="row">
      <div class="lef"><label>Имя</label></div>
      <div class="rig"><?php echo ($row['name']); ?></div>
    </div>
    <div class="row">
      <div class="lef"><label>Отчество</label></div>
      <div class="rig"><?php echo ($row['patron']); ?></div>
    </div>
    <div class="row">
      <div class="lef"><label>Телефон</label></div>
      <div class="rig"><?php echo ("+" . $row['tel']); ?></div>
    </div>
    <div class="row">
      <div class="lef"><label>E-mail</label></div>
      <div class="rig"><?php echo ($row['mail']); ?></div>
    </div>
    <div class="row">
      <div class="lef"><label>Адрес</label></div>

```

					ТРПО 2-40 01 01.33.33.06.23 ПЗ	Лист
Изм.	Лист	№докум.	Подпись	Дата		49

```

        <div class="rig"><?php echo ("г. " . $row['city'] . ", ул. " . $row['street'] . ", д." . $row['house'] . ",
кв." . $row['apart']); ?></div>
    </div>
</div>
</div>
<div class="button">
    <?php
    if ($_SESSION['users']) : ?>
        <button class="button_item" onclick="document.location='/booking.php'">Заказы</button>
        <?php if ($id['access'] == 1) : ?>
            <button class="button_item" onclick="document.location='/staff.php'">Сотрудники</button>
        <?php endif ?>
    <?php endif ?>
</div>
<div class="tabl">
    <table class="table">
        <thead>
            <tr>
                <th>День</th>
                <th>Время работы</th>
                <th>Перерыв</th>
            </tr>
        </thead>
        <tbody>
            <?php
            $work = mysqli_query($mysqli, "SELECT * FROM `work` WHERE `id` = '{ $id['id'] }'");
            //получаем данные БД work
            $work = mysqli_fetch_all($work);
            // print_r($personal);
            foreach ($work as $wor) {
            }
            ?>
            <tr>
                <td>Понедельник</td>
                <td><?= $wor[1] /*Время работы понедельник*/ ?></td>
                <td><?= $wor[2] ?></td>
            </tr>
            <tr>
                <td>Вторник</td>
                <td><?= $wor[3] /*Время работы понедельник*/ ?></td>
                <td><?= $wor[4] ?></td>
            </tr>
            <tr>
                <td>Среда</td>
                <td><?= $wor[5] /*Время работы понедельник*/ ?></td>
                <td><?= $wor[6] ?></td>
            </tr>
            <tr>
                <td>Четверг</td>
                <td><?= $wor[7] /*Время работы понедельник*/ ?></td>
                <td><?= $wor[8] ?></td>
            </tr>
            <tr>
                <td>Пятница</td>
                <td><?= $wor[9] /*Время работы понедельник*/ ?></td>
            </tr>
        </tbody>
    </table>
</div>

```



```

        <td><?= $wor[10] ?></td>
    </tr>
    <tr>
        <td>Суббота</td>
        <td><?= $wor[11] /*Время работы понедельник*/ ?></td>
        <td><?= $wor[12] ?></td>
    </tr>
    <tr>
        <td>Воскресенье</td>
        <td><?= $wor[13] /*Время работы понедельник*/ ?></td>
        <td><?= $wor[14] ?></td>
    </tr>
</tbody>
</table>
</div>
</body>
</html>

```

### booking.php // Файл страницы заказов

```

<?php
include "php/db_connect.php";
session_start();
$login = $_SESSION['users']['login'];
$query = mysqli_query($mysqli, "SELECT * FROM `users` WHERE `login` = '{ $login }'");
$row = mysqli_fetch_assoc($query);
//print_r($row);?>
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Заказы</title>
    <link rel="stylesheet" href="/css/booking.css">
    <link rel="stylesheet" href="/css/header.css">
</head>
<body>
    <div class="header">
        <div class="header_client_section">
            <div class="header_section">
                Заказы
            </div>
            <div class="clientData">
                <div class="headerButton">
                    <?php if ($_SESSION['users'] == "") : ?>
                        <a class="a1" href="/login.php">Войти</a>
                    <?php else : ?>
                        <a class="a1" href="/cabinet.php">
                            <?php
                                if ($_SESSION['users']) {
                                    echo ($_SESSION['users']['login']);
                                }
                            ?>
                        </a>
                    <?php endif ?>
                </div>
                <div class="headerButton">

```

```

<?php if ($_SESSION['users'] == '') : ?>
    <!--<a class="a1" href="/register.php">Регистрация</a> -->
<?php else : ?>
    <a class="a1" href="php/logout.php">Выйти</a>
<?php endif ?>
</div>
</div>
</div>
</div>
<script src="js/sorttable.js"></script>
<table class="table" id="unique_id">
    <thead>
        <tr>
            <th width="50">№</th>
            <th width="240">ФИО</th>
            <th width="50">Номер стола</th>
            <th>Перечень заказа</th>
            <th>Примечание</th>
            <th width="120">Стоимость</th>
            <th width="94">Статус</th>
            <?php
            if ($row['access'] == 1) { ?>
                <th width="240">Обслуживал</th>
            <?php
            } ?>
            <th width="94"></th>
        </tr>
    </thead>
    <tbody>
        <?php
        $query_b = mysqli_query($mysqli, "SELECT * FROM `booking`");
        //получаем данные БД category users <th width="94">ф-ции</th>
        $book = mysqli_fetch_all($query_b);
        // print_r($book);
        $status = mysqli_query($mysqli, "SELECT * FROM `status`");
        //получаем данные БД category
        $status = mysqli_fetch_all($status);
        // print_r($status);
        $personal = mysqli_query($mysqli, "SELECT * FROM `personal`");
        //получаем данные БД personal
        $personal = mysqli_fetch_all($personal);
        // print_r($personal);
        $query = mysqli_query($mysqli, "SELECT * FROM `booking`");
        //получаем данные БД booking
        $query = mysqli_fetch_all($query);
        // print_r($query[0]);
        if ($row['access'] == 1) {
            foreach ($query as $quer) {?>
                <tr>
                    <td><?= $quer[0] /*Номер*/ ?></td>
                    <td><?= $quer[1] . " " . $quer[2] . " " . $quer[3] /*ФИО*/ ?></td>
                    <td><?= $quer[4] /*Номер стола*/ ?></td>
                    <td><?= $quer[5] /*Перечень заказа*/ ?></td>
                    <td><?= $quer[6] /*Примечание*/ ?></td>
                    <td><?= $quer[7] . " руб." /*Стоимость*/ ?></td>
                    <?php

```

```

        foreach ($status as $stat) {
            if ($stat[0] == $quer[8]) {
                echo "<td>" . $stat[1] . "</td>";
            } /*Статус заказа */
        }
    ?>
    <td>
        <?php
        foreach ($personal as $pers) {
            if ($pers[0] == $quer[9]) {
                echo $pers[1] . " " . $pers[2] . " " . $pers[3];
                break;
            }
        }
        /*Обслуживал*/?>
    </td>
    <td>
        <div class="function">
            <a href="updateBook.php?id=<?= $quer[0] ?>">Изменить</a>
            <a href="php/deleteTable.php?id=<?= $quer[0] ?>">Удалить</a>
        </div>
    </td>
</tr>
<?php }
} else {
    $stat = 1;
    $query = mysqli_query($mysqli, "SELECT * FROM `booking` WHERE `status` != {$stat}");
    //получаем данные БД booking
    $query = mysqli_fetch_all($query);
    // print_r($query);
    foreach ($query as $quer) {
    } ?>
    <tr>
        <td><?= $quer[0] /*Номер*/ ?></td>
        <td><?= $quer[1] . " " . $quer[2] . " " . $quer[3] /*ФИО*/ ?></td>
        <td><?= $quer[4] /*Номер стола*/ ?></td>
        <td><?= $quer[5] /*Перечень заказа*/ ?></td>
        <td><?= $quer[6] /*Примечание*/ ?></td>
        <td><?= $quer[7] /*Стоимость*/ ?></td>
        <?php
        foreach ($status as $stat) {
            if ($stat[0] == $quer[8]) {
                echo "<td>" . $stat[1] . "</td>";
            } /*Статус заказа*/?>
        <td>
            <div class="function">
                <a href="updateBook.php?id=<?= $quer[0] ?>">Изменить</a>
                <a href="php/deleteTable.php?id=<?= $quer[0] ?>">Удалить</a>
            </div>
        </td>
        <?php } ?>
    </tr>
</tbody>
</table>
</body>
</html>

```

## addstaff.php // Файл добавления сотрудника

```
<?php
include "php/db_connect.php";
// print_r($_GET);
$book_id = $_GET['id'];
$booking = mysqli_query($mysqli, "SELECT * FROM `booking` WHERE `id` = '{$book_id}'");
$booking = mysqli_fetch_assoc($booking);
$personal = mysqli_query($mysqli, "SELECT * FROM `personal` WHERE `id` = '{$booking['id_personal']}'");
$personal = mysqli_fetch_assoc($personal);
print_r($personal);?>
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Регистрация сотрудника</title>
  <link rel="stylesheet" href="/css/update.css">
</head>
<body>
  <div class="text">
    <h1>Регистрация сотрудника</h1>
  </div>
  <div class="rows">
    <form action="php/updateTable.php" method="post">
      <div class="input-box">
        <input type="text" name="surname" class="log_input" placeholder="Фамилия" required>
        <input type="text" name="name" class="log_input" placeholder="Имя" required>
        <input type="text" name="patron" class="log_input" placeholder="Отчество" required>
        <input type="text" name="post" class="log_input" placeholder="Должность" required>
        <input type="text" name="tel" class="log_input" placeholder="Телефон" required>
        <input type="text" name="mail" class="log_input" placeholder="Е-mail" required>
        <input type="text" name="city" class="log_input" placeholder="Город" required>
        <input type="text" name="street" class="log_input" placeholder="Улица" required>
        <input type="text" name="house" class="log_input" placeholder="Дом" required>
        <input type="text" name="apart" class="log_input" placeholder="Квартира" required>
        <input type="text" name="login" class="log_input" placeholder="Логин" required>
        <input type="password" name="pass" class="log_input" placeholder="Пароль" required>
        <input type="password" name="P_pas" class="log_input" placeholder="Подтверждение пароля"
required>
        <div class="log_but">
          <button type="submit" class="log_but_1 log">Изменить</button>
        </div>
      </div>
    </form>
  </div>
</body>
</html>
```

## modal\_staff.php //Файл html-разметки модального окна

```
<!-- Модальное окно -->
<div id="myModal" class="modal">
  <!-- Модальное содержание -->
  <div class="modal-content">
    <div class="modal-header">
      <span class="close">&times;</span>
      <h2>Регистрация</h2>
```

					ТРПО 2-40 01 01.33.33.06.23 ПЗ	Лист
Изм.	Лист	№докум.	Подпись	Дата		54

```

</div>
<div class="modal-body">
  <form action="php/reg.php" method="post" class="form" id="add-form">
    <div class="input-box">
      <input type="text" name="surname" class="log_input" placeholder="Фамилия" required>
      <input type="text" name="name" class="log_input" placeholder="Имя" required>
      <input type="text" name="patron" class="log_input" placeholder="Отчество" required>
      <input type="text" name="post" class="log_input" placeholder="Должность" required>
      <input type="text" name="tel" class="log_input" placeholder="Телефон" required>
      <input type="text" name="mail" class="log_input" placeholder="Е-mail" required>
      <input type="text" name="city" class="log_input" placeholder="Город" required>
      <input type="text" name="street" class="log_input" placeholder="Улица" required>
      <input type="text" name="house" class="log_input" placeholder="Дом" required>
      <input type="text" name="apart" class="log_input" placeholder="Квартира" required>
      <input type="text" name="login" class="log_input" placeholder="Логин" required>
      <input type="password" name="pass" class="log_input" placeholder="Пароль" required>
      <input type="password" name="P_pas" class="log_input" placeholder="Подтверждение пароля"
required>
    </div>
    <div class="log_but" ><!-- required    onsubmit="return false;-->
      <button type="submit" class="log_but_1 log">Зарегистрировать</button>
    </div>
  </form>
</div>
<script defer src="/js/ind_mod_auth.js"></script>
</div>
</div>

```

#### staff.php // Файл страницы сотрудников

```

<?php
include "php/db_connect.php";
session_start();
$login = $_SESSION['users']['login'];
$query = mysqli_query($mysqli, "SELECT * FROM `users` WHERE `login` = '{ $login }'");
$row = mysqli_fetch_assoc($query);
// $id = $id['id'];
// print_r($id);
?>
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Сотрудники</title>
  <link rel="stylesheet" href="/css/header.css">
  <link rel="stylesheet" href="/css/staff.css">
  <link rel="stylesheet" href="/css/modal_staff.css">
</head>
<body>
  <div class="header">
    <div class="header_client_section">
      <div class="header_section">
        Сотрудники
      </div>
      <div class="clientData">
        <div class="headerButton">

```

					ТРПО 2-40 01 01.33.33.06.23 ПЗ	Лист
Изм.	Лист	№ док.м.	Подпись	Дата		55

```

<?php
session_start();
?>
<?php if ($_SESSION['users'] == "") : ?>
    <a class="a1" href="/login.php">Войти</a>
<?php else : ?>
    <!--
        <a class="a1" href="/index.php">Главная</a> -->
    <a class="a1" href="/cabinet.php">
        <?php
            if ($_SESSION['users']) {
                echo ($_SESSION['users']['login']);
            }
        ?>
    </a>
<?php endif ?>
</div>
<div class="headerButton">
    <?php if ($_SESSION['users'] == "") : ?>
    <?php else : ?>
        <a class="a1" href="php/logout.php">Выйти</a>
    <?php endif ?>
</div>
</div>
</div>
<div class="button">
    <button class="button_item" id="myBtn">Добавить сотрудника</button>
    <?php
        include "modal_staff.php";
    ?>
</div>
<script src="js/sorttable.js"></script>
<table class="table" id="unique_id">
    <thead>
        <tr>
            <th width="50">№</th>
            <th width="200">ФИО</th>
            <th width="130">Должность</th>
            <th width="130">Телефон</th>
            <th width="180">E-mail</th>
            <th width="220">Адрес</th>
            <th width="90">Логин</th>
            <th width="200">График работы</th>
            <!--<th width="120">Пароль</th> -->
            <th width="94"></th>
        </tr>
    </thead>
    <tbody>
        <?php
            $personal = mysqli_query($mysqli, "SELECT * FROM `personal`");
            //получаем данные БД personal
            $personal = mysqli_fetch_all($personal);
            // print_r($personal);
            $users = mysqli_query($mysqli, "SELECT * FROM `users`");
            //получаем данные БД booking

```

					ТРПО 2-40 01 01.33.33.06.23 ПЗ	Лист
Изм.	Лист	№докум.	Подпись	Дата		56

```

$users = mysqli_fetch_all($users);
// print_r($users);
$work = mysqli_query($mysqli, "SELECT * FROM `work`");
//получаем данные БД booking
$work = mysqli_fetch_all($work);
// print_r($users);
if ($row['access'] == 1) {
    foreach ($personal as $pers) { ?>
        <tr>
            <td><?= $pers[0] /*Номер*/ ?></td>
            <td><?= $pers[1] . " " . $pers[2] . " " . $pers[3] /*ФИО*/ ?></td>
            <td><?= $pers[4] /*Должность*/ ?></td>
            <td><?= "+" . $pers[5] /*Телефон*/ ?></td>
            <td><?= $pers[6] /*E-mail*/ ?></td>
            <td><?= "г. " . $pers[7] . ", ул. " . $pers[8] . ", д. " . $pers[9] . ", кв. " . $pers[10]; /*Адрес*/
?></td>

            <td>
                <?php
                foreach ($users as $user) {
                    if ($pers[0] == $user[0]) {
                        echo $user[1];
                        //echo "<td>" . $user[2] . "</td>";
                        break;
                    }
                } /*Логин*/?>
            </td>
            <td>
                <div class="work">
                    <?php
                    foreach ($work as $wor) {
                        if ($pers[0] == $wor[0]) {
                            echo "Пн:" . $wor[1] . " | " . $wor[2];
                            echo " Вт:" . $wor[3] . " | " . $wor[4];
                            echo " Ср:" . $wor[5] . " | " . $wor[6];
                            echo " Чт:" . $wor[7] . " | " . $wor[8];
                            echo " Пт:" . $wor[9] . " | " . $wor[10];
                            echo " Сб:" . $wor[11] . " | " . $wor[12];
                            echo " Вс:" . $wor[13] . " | " . $wor[14];
                            break;
                        }
                    } /*График работы*/?>
                </div>
            </td>
            <td>
                <div class="function">
                    <a href="updateStaff.php?id=<?= $pers[0] ?>">Изменить</a>
                    <a href="php/delete.php?id=<?= $pers[0] ?>">Удалить</a>
                </div>
            </td>
        <?php
        }}?>
    </tr>
</tbody>
</table>
</body>
</html>

```

					ТРПО 2-40 01 01.33.33.06.23 ПЗ	Лист
Изм.	Лист	№ док-м.	Подпись	Дата		57

## updateBook.php // Файл страницы редактирования заказа

```
<?php
    include "php/db_connect.php";
    // print_r($_GET);
    $book_id = $_GET['id'];
    $booking = mysqli_query($mysqli, "SELECT * FROM `booking` WHERE `id` = '{$book_id}'");
    $booking = mysqli_fetch_assoc($booking);
    $personal = mysqli_query($mysqli, "SELECT * FROM `personal` WHERE `id` =
'{$booking['id_personal']}'");
    $personal = mysqli_fetch_assoc($personal);
    print_r($personal);?>
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Редактирование заказа</title>
    <link rel="stylesheet" href="/css/update.css">
</head>
<body>
    <div class="text">
        <h1>Редактирование заказа</h1>
    </div>
    <div class="rows">
        <form action="php/updateTable.php" method="post">
            <div class="input-box">
                <input type="text" name="id" class="log_input" placeholder="id" value="<?=$booking['id'] ?>"
readonly>
                <input type="text" name="surname" class="log_input" placeholder="Фамилия" value="<?=$
$booking['surname'] ?>" required>
                <input type="text" name="name" class="log_input" placeholder="Имя" value="<?=$
$booking['name'] ?>" required>
                <input type="text" name="patron" class="log_input" placeholder="Отчество" value="<?=$
$booking['patron'] ?>" required>
                <input type="text" name="num" class="log_input" placeholder="Номер стола" value="<?=$
$booking['numTable'] ?>" required>
                <input type="text" name="book" class="log_input" placeholder="Перечень заказа" value="<?=$
$booking['book'] ?>" required>
                <input type="text" name="book2" class="log_input" placeholder="Примечание" value="<?=$
$booking['book2'] ?>">
                <input type="text" name="cost" class="log_input" placeholder="Стоимость" value="<?=$
$booking['cost'] ?>" required>
                <input type="text" name="stat" class="log_input" placeholder="Статус" value="<?=$
$booking['status'] ?>" required>
                <input type="text" name="pers" class="log_input" placeholder="Официант" value="<?=$
$personal['surname']. " ". $personal['name']. " ". $personal['patron'] ?>">
                <input type="hidden" name="id_pers" value="<?=$personal['id'] ?>">
                <div class="log_but" >
                    <button type="submit" class="log_but_1 log">ИЗМЕНИТЬ</button>
                </div>
            </div>
        </form>
    </div>
</body>
</html>
```

					ТРПО 2-40 01 01.33.33.06.23 ПЗ	Лист
Изм.	Лист	№докум.	Подпись	Дата		58



**updateStaff.php // Файл страницы редактирования данных сотрудника**

```
<?php
include "php/db_connect.php";
// print_r($_GET);
$id_us = $_GET['id'];
$personal = mysqli_query($mysqli, "SELECT * FROM `personal` WHERE `id` = '{ $id_us }'");
$personal = mysqli_fetch_assoc($personal);
$user = mysqli_query($mysqli, "SELECT * FROM `users` WHERE `id` = '{ $id_us }'");
$user = mysqli_fetch_assoc($user);
// print_r($personal);?>
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Редактирование сотрудника</title>
    <link rel="stylesheet" href="/css/update.css">
    <link rel="stylesheet" href="/css/updateStaff.css">
</head>
<body>
    <div class="text">
        <h1>Редактирование сотрудника</h1>
    </div>
    <div class="rows">
        <form action="php/updateStaff.php" method="post">
            <div class="input-box">
                <input type="text" name="id" class="log_input" placeholder="id" value="<?=$personal['id'] ?>"
readonly>
                <input type="text" name="surname" class="log_input" placeholder="Фамилия" value="<?=$personal['surname'] ?>" required>
                <input type="text" name="name" class="log_input" placeholder="Имя" value="<?=$personal['name'] ?>" required>
                <input type="text" name="patron" class="log_input" placeholder="Отчество" value="<?=$personal['patron'] ?>" required>
                <input type="text" name="post" class="log_input" placeholder="Должность" value="<?=$personal['post'] ?>" required>
                <input type="text" name="acc" class="log_input" placeholder="Уровень доступа" value="<?=$user['access'] ?>"
                <input type="text" name="tel" class="log_input" placeholder="Телефон" value="<?=$personal['tel']
?>" required>
                <input type="text" name="mail" class="log_input" placeholder="Е-mail" value="<?=$personal['mail'] ?>" required>
                <input type="text" name="city" class="log_input" placeholder="Город" value="<?=$personal['city']
?>" >
                <input type="text" name="street" class="log_input" placeholder="Улица" value="<?=$personal['street'] ?>" >
                <input type="text" name="house" class="log_input" placeholder="Дом" value="<?=$personal['house'] ?>" >
                <input type="text" name="apart" class="log_input" placeholder="Квартира" value="<?=$personal['apart'] ?>" >
                <input type="hidden" name="id_pers" value="<?=$personal['id'] ?>">
                <div class="log_but" ><!-- required    onsubmit="return false;"-->
                    <button type="submit" class="log_but_1 log">Изменить</button>
                </div>
            </div>
        </form>
    </div>
```

					ТРПО 2-40 01 01.33.33.06.23 ПЗ	Лист
Изм.	Лист	№докум.	Подпись	Дата		59

```

</div>
</form>
</body>
</html>

```

## Файлы стилизации страниц CSS

### booking.css

```

.work{
    display: grid;
    grid-template-rows: 14% 14% 14% 14%
14% 14% 14%;
}
.function{
    display: grid;
    grid-template-rows: 50% 50%;
}
.table{
    border: 1px solid #eee;
    table-layout: fixed;
    width: 98%;
    margin: 1%;
}
.a.sorthead {
    color:#666666;
}
.table th {
    font-weight: bold;
    padding: 5px;
    background: #efefef;
    border: 1px solid #dddddd;
}
.table td{
    padding: 5px 10px;
    border: 1px solid #eee;
    text-align: left;
}
.table tbody tr:nth-child(odd){
    background: #fff;
}
.table tbody tr:nth-child(even){
    background: #F7F7F7;
}

```

### cabinet.css

```

.m2{
    display: flex;
    flex-direction: column;
    justify-self: flex-start;

    min-width: 500px;
    height: 100%;
    padding: 8px 8px;
}
.row{
    display: grid;
    grid-template-columns: 30% 70%;

    padding: 5px 8px;
    font-family: -apple-
system,BlinkMacSystemFont,"Segoe
UI",Roboto,"Helvetica Neue",Arial,"Noto
Sans",sans-serif,"Apple Color Emoji","Segoe UI
Emoji","Segoe UI Symbol","Noto Color Emoji";
    font-size: 16px;
}
.button{
    display: grid;
    grid-template-columns: 10% 10%;
}
.button_item{
    background-color: #678a1e;
    border-color: #678a1e;
    color: #fff;

    font-family: -apple-system,
BlinkMacSystemFont, "Segoe UI", Roboto,
"Helvetica Neue", Arial, "Noto Sans", sans-serif,
"Apple Color Emoji", "Segoe UI Emoji", "Segoe UI
Symbol", "Noto Color Emoji";
    border: none;
    width: 90%;
    margin: 10px;
    padding: 6px 12px;
    font-size: 15px;
}
.lef{
    font-weight:500;
}
.information{
    display: grid;
    /*
    grid-template-rows: 30% 70%;
    */
}

```

					ТРПО 2-40 01 01.33.33.06.23 ПЗ	Лист
Изм.	Лист	№ док-м.	Подпись	Дата		60

```

    grid-template-columns: 25% 30%;
    padding: 2px 15px;
    font-size: 17px;
    margin: 3%;
}
.container_inf{
    display: flex;
    align-content: center;
    padding: 10px 10px;
    font-size: 10px;
}
.container_pers{
    display: grid;
}
.image_pers{
    width: 250px;
    height: 250px;
}
.function{
    margin-top: 7px;
    width: 350px;
}
.avatar {
    border-radius: 50%; /* Радиус скругления
padding: 2px 15px; */
    vertical-align: middle;
    width: 250px;
    height: 250px;
    border: 3px solid rgb(0, 0, 0); /* Параметры
рамки */
    /* box-shadow: 0 0 7px #666; /* Параметры
тени */
}

.button{
    font-family: -apple-system,
BlinkMacSystemFont, "Segoe UI", Roboto,
"Helvetica Neue", Arial, "Noto Sans", sans-serif,
"Apple Color Emoji", "Segoe UI Emoji", "Segoe UI
Symbol", "Noto Color Emoji";
    font-size: 12px;
    margin-left: 2%;

    cursor: pointer;
}
.edit{
    font-family: -apple-system,
BlinkMacSystemFont, "Segoe UI", Roboto,
"Helvetica Neue", Arial, "Noto Sans", sans-serif,
"Apple Color Emoji", "Segoe UI Emoji", "Segoe UI
Symbol", "Noto Color Emoji";
    font-size: 12px;
    width: 50px;
}

```

```

.tabl{
    display: flex;
    justify-content: flex-start;

    padding: 2px 15px;
    margin: 3%;
}
.table {
    width: 20%;

    margin-bottom: 20px;
    border: 1px solid #dddddd;
    border-collapse: collapse;
}
.table th {
    font-weight: bold;
    padding: 5px;
    background: #678a1e;
    border: 1px solid #678a1e;
    color: #fff2f2;
}
.table td {
    border: 1px solid #dddddd;
    padding: 5px;
}

```

## header.css, index.css

```
.a1 {
    color: #ffffff;
    text-decoration: none;
}

.header {
    padding: 10px 10px;
    background: #597719;
    color: #ffffff;
    font-family: Uni Sans;
}

.header_item {
    padding: 10px 15px;
    font-size: 20px;
    margin-left: 5px
}

.header_section {

    font-weight: bold;
    padding: 10px 15px;
    font-size: 20px;
    margin-left: 5px
}

.headerButton {
    cursor: pointer;
}

.header_client_section {
    display: flex;
    justify-content: space-between;
}

.clientData {
    display: grid;
    grid-template-rows: 50% 50%;
    padding: 2px 15px;
    font-size: 17px;
    margin-left: 5px
}
```

## modal\_index.css

```
.log_input {
    width: 90%;
    padding: 6px 12px;
    font-weight: 450;
    font-size: 15px;
    margin-top: 15px;
    font-family: "Segoe UI", "Helvetica Neue";
}

.log_but_1 {
    background-color: #62821c;
    border-color: #62821c;
    color: #fff;
    font-family: -apple-system,
BlinkMacSystemFont, "Segoe UI", Roboto,
"Helvetica Neue", Arial, "Noto Sans", sans-serif,
"Apple Color Emoji", "Segoe UI Emoji", "Segoe UI
Symbol", "Noto Color Emoji";
    border: none;
    width: 100%;
    margin-top: 15px;
    padding: 6px 12px;
    font-size: 15px;
}

.log_but_1:hover {
    background-color: #425813;
    border-color: rgb(73, 148, 73);
    color: #ffffff;
}

/* Модальный (фон) */
.modal {
    display: none;
    position: fixed;
    /* Оставаться на месте */
    z-index: 1;
    padding-top: 100px;
    /* Расположение коробки */
    left: 0;
    top: 0;
    width: 100%;
    height: 100%;
    overflow: auto;
    /* Включите прокрутку, если это необходимо
    */
    background-color: rgb(0, 0, 0);
    /* Цвет запасной вариант */
    background-color: rgba(0, 0, 0, 0.4);
    /*Черный с непрозрачностью */
}

/* Модальное содержание */
.modal-content {
    position: relative;
    background-color: #fefefe;
    margin: auto;
    padding: 0;
    border: 1px solid #888;
```

Изм.	Лист	№докум.	Подпись	Дата

ТРПО 2-40 01 01.33.33.06.23 ПЗ

Лист

62

```

width: 15%;
min-width: 230px;
height: 18%;
min-height: 230px;
box-shadow: 0 4px 8px 0 rgba(0, 0, 0, 0.2), 0 6px
20px 0 rgba(0, 0, 0, 0.19);
-webkit-animation-name: animatetop;
-webkit-animation-duration: 0.4s;
animation-name: animatetop;
animation-duration: 0.4s
}
/* Добавить анимацию */
@-webkit-keyframes animatetop {
  from {
    top: -300px;
    opacity: 0
  }

  to {
    top: 0;
    opacity: 1
  }
}
@keyframes animatetop {
  from {
    top: -300px;
    opacity: 0
  }
  to {
    top: 0;
    opacity: 1
  }
}
/* Кнопка закрытия */
.close {
  color: white;
  float: right;
  font-size: 28px;
  font-weight: bold;
}
.close:hover,
.close:focus {
  color: #000;
  text-decoration: none;
  cursor: pointer;
}
.modal-header {
  padding: 2px 16px;
  background-color: #62821c;
  color: white;
}
.modal-body {
  padding: 2px 16px;
}

```

## modal\_staff.css

```

.log_input {
  width: 90%;
  padding: 6px 12px;
  font-weight: 450;
  font-size: 13px;
  margin-top: 15px;
  font-family: "Segoe UI", "Helvetica Neue";
}
.log_but_1 {
  background-color: #62821c;
  border-color: #62821c;
  color: #fff;
  font-family: -apple-system,
BlinkMacSystemFont, "Segoe UI", Roboto,
"Helvetica Neue", Arial, "Noto Sans", sans-serif,
"Apple Color Emoji", "Segoe UI Emoji", "Segoe UI
Symbol", "Noto Color Emoji";
  border: none;
  width: 100%;
  margin-top: 15px;
  padding: 6px 12px;
  font-size: 15px;
}
.log_but_1:hover {
  background-color: #425813;
  border-color: rgb(73, 148, 73);
  color: #ffffff;
}
/* Модальный (фон) */
.modal {
  display: none;
  position: fixed;
  /* Оставаться на месте */
  z-index: 1;
  padding-top: 100px;
  /* Расположение коробки */
  left: 0;
  top: 0;
  width: 100%;
  height: 100%;
  overflow: auto;
  /* Включите прокрутку, если это необходимо */
  background-color: rgb(0, 0, 0);
  /* Цвет запасной вариант */
  background-color: rgba(0, 0, 0, 0.4);
  /*Черный с непрозрачностью */
}
/* Модальное содержание min-height: 600px;*/
.modal-content {
  position: relative;
  background-color: #fefefe;
  margin: auto;
  padding: 0;
  border: 1px solid #888;
}

```

Изм.	Лист	№докум.	Подпись	Дата

ТРПО 2-40 01 01.33.33.06.23 ПЗ

Лист

63

```

width: 15%;
min-width: 230px;
height: 66%;

box-shadow: 0 4px 8px 0 rgba(0, 0, 0, 0.2), 0 6px
20px 0 rgba(0, 0, 0, 0.19);
-webkit-animation-name: animatetop;
-webkit-animation-duration: 0.4s;
animation-name: animatetop;
animation-duration: 0.4s
}
/* Добавить анимацию */
@-webkit-keyframes animatetop {
  from {
    top: -300px;
    opacity: 0
  }
  to {
    top: 0;
    opacity: 1
  }
}
@keyframes animatetop {
  from {
    top: -300px;
    opacity: 0
  }
  to {
    top: 0;
    opacity: 1
  }
}
/* Кнопка закрытия */
.close {
  color: white;
  float: right;
  font-size: 28px;
  font-weight: bold;
}
.close:hover,
.close:focus {
  color: #000;
  text-decoration: none;
  cursor: pointer;
}
.modal-header {
  padding: 2px 16px;
  background-color: #62821c;
  color: white;
}
.modal-body {
  padding: 2px 16px;
}

```

## staff.css

```

.button{
  display: grid;
  grid-template-columns: 15% 10%;
}
.button_item{
  background-color: #678a1e;
  border-color: #678a1e;
  color: #fff;
  font-family: -apple-system,
BlinkMacSystemFont, "Segoe UI", Roboto,
"Helvetica Neue", Arial, "Noto Sans", sans-serif,
"Apple Color Emoji", "Segoe UI Emoji", "Segoe UI
Symbol", "Noto Color Emoji";
  border: none;
  width: 90%;
  margin: 10px;
  padding: 6px 12px;
  font-size: 15px;
}
.function{
  display: grid;
  grid-template-rows: 50% 50%;
}
.table{
  border: 1px solid #eee;
  table-layout: fixed;
  width: 98%;
  margin: 1%;
}
.a.sorthead {
  color: #666666;
}
.table th {
  font-weight: bold;
  padding: 5px;
  background: #efefef;
  border: 1px solid #dddddd;
}
.table td{
  padding: 5px 10px;
  border: 1px solid #eee;
  text-align: left;
}
.table tbody tr:nth-child(odd){
  background: #fff;
}
.table tbody tr:nth-child(even){
  background: #F7F7F7;
}

```

Изм.	Лист	№ док-м.	Подпись	Дата

ТРПО 2-40 01 01.33.33.06.23 ПЗ

Лист

64

## update.css

```
.text{
    display: flex;
    justify-content: center;
}
h1{
    margin: 20px 0px;
    font-weight: 500;
    padding: 10px;
    background: #f8f9fa;
    font-family: Uni Sans;
    padding: 8px 25%;
}
.rows{
    display: flex;
    justify-content: center;
}
.input-box{
    display: grid;
    grid-template-rows: 8% 8% 8% 8% 8% 8% 8%
8% 8% 8% 8% 8%;
}
.log_input {
    width: 100%;
    padding: 6px 12px;
    font-weight: 450;
    font-size: 13px;
    margin-top: 15px;
    margin-right: 5px;
    height: 12px;
    font-family: "Segoe UI", "Helvetica Neue";
}
.log_but_1 {
    background-color: #5cb85c;
    border-color: #50a050;
    color: #fff;
    font-family: -apple-system,
BlinkMacSystemFont, "Segoe UI", Roboto,
"Helvetica Neue", Arial, "Noto Sans", sans-serif,
"Apple Color Emoji", "Segoe UI Emoji", "Segoe UI
Symbol", "Noto Color Emoji";
    border: none;
    width: 100%;
    margin-top: 15px;
    padding: 6px 12px;
    font-size: 15px;
}
.log_but_1:hover {
    background-color: #50a050;
    border-color: rgb(73, 148, 73);
    color: #ffffff;
}
```

## Файлы php

**auth.php** // файл авторизации

<?php

## updateStaff.css

```
.work{
    display: grid;
    grid-template-rows: 14% 14% 14% 14%
14% 14% 14%;
}
.row{
    display: grid;
    grid-template-columns: 30% 40% 50%;

    width: 20%;
    padding: 6px 12px;
    font-weight: 450;
    font-size: 13px;
    margin-top: 15px;
    height: 12px;
}
.lab{
    font-weight: 450;
    font-size: 15px;
    margin-top: 15px;
}
```

|      |      |         |         |      |                                |      |
|------|------|---------|---------|------|--------------------------------|------|
|      |      |         |         |      | ТРПО 2-40 01 01.33.33.06.23 ПЗ | Лист |
| Изм. | Лист | №докум. | Подпись | Дата |                                | 65   |

```

session_start();
$mysqli = new mysqli("datamon", "root", "", "datamon");
if ($mysqli->connect_error) {
    die('Ошибка : ('. $mysqli->connect_errno .') '. $mysqli->connect_error);
    exit();
}
$login = $_POST['login'];
$pass = $_POST['pass'];
$md5_pass = md5($pass);
$query = mysqli_query($mysqli, "SELECT * FROM `users` WHERE `login`='{ $login}' AND `pass` = '{ $md5_pass}'");
if (mysqli_num_rows($query) == 1){
    $_SESSION['users'] = ['login' => $login];
    header("Location: /index.php");
} else{
    header("Location: /index.php");
}?>

```

### **db\_connect.php // файл подключения БД**

```

<?php
// $db = mysqli_connect("omegatrak", "root", "", "register");
$mysqli = new mysqli("datamon", "root", "", "datamon");
if ($mysqli->connect_error) {
    die('Ошибка : ('. $mysqli->connect_errno .') '. $mysqli->connect_error);}
$charset = 'utf8';
if (!$mysqli->set_charset($charset)) {
    echo "Ошибка установки кодировки UTF8 ";
    exit();
}?>

```

### **delete.php // файл удаления сотрудника**

```

<?php
session_start();
$mysqli = new mysqli("datamon", "root", "", "datamon");
if ($mysqli->connect_error) {
    die('Ошибка : ('. $mysqli->connect_errno .') '. $mysqli->connect_error);
    exit();}
$login = $_SESSION['users']['login'];
$query = mysqli_query($mysqli, "SELECT * FROM `users` WHERE `login` = '{ $login}'");
$row = mysqli_fetch_assoc($query);
if ($row['access'] == 1){
    $Pquery = mysqli_query($mysqli, "SELECT * FROM `personal` WHERE `id` = '{ $row['id']}'");
    $Prow = mysqli_fetch_assoc($Pquery);
    $query = "";
    $id = $_GET['id'];
    //Отправляем запрос БД users с условием `id_users` = '$id_users' AND `id` = '$id'
    $query = mysqli_query($mysqli, "SELECT * FROM `users` WHERE `id` = '$id'");
    if (mysqli_num_rows($query) > 0) {
        mysqli_query($mysqli, "DELETE FROM `users` WHERE `users`.`id` = '{ $id}'");
        mysqli_query($mysqli, "DELETE FROM `personal` WHERE `personal`.`id` = '{ $id}'");
        mysqli_query($mysqli, "DELETE FROM `work` WHERE `work`.`id` = '{ $id}'");
        mysqli_query($mysqli, "DELETE FROM `images` WHERE `images`.`id` = '{ $id}'");
        header("Location: /staff.php");
        exit();
    } else{
        echo "Error";
    }
}

```

|      |      |          |         |      |                                |      |
|------|------|----------|---------|------|--------------------------------|------|
|      |      |          |         |      | ТРПО 2-40 01 01.33.33.06.23 ПЗ | Лист |
| Изм. | Лист | № док.м. | Подпись | Дата |                                | 66   |



```

        //print_r($id);
    }
}??>

```

#### **deleteTable.php // файл удаления заказа**

```

<?php
session_start();
$mysqli = new mysqli("datamon", "root", "", "datamon");
if ($mysqli->connect_error) {
    die('Ошибка : ('. $mysqli->connect_errno .') '. $mysqli->connect_error);
    exit();}
$login = $_SESSION['users']['login'];
$query = mysqli_query($mysqli, "SELECT * FROM `users` WHERE `login` = '{$login}'");
$row = mysqli_fetch_assoc($query);
$id_book = $_GET['id'];
//Отправляем запрос БД booking с условием `id_users` = '$id_users' AND `id` = '$id'
$query = mysqli_query($mysqli, "SELECT * FROM `users` WHERE `id` = '{$row['id']}' AND `access` = 1");
if (mysqli_num_rows($query) > 0) {
    mysqli_query($mysqli, "DELETE FROM `booking` WHERE `booking`.`id` = '{$id_book}'");
    header("Location: /booking.php");
    exit();
} else{
    echo "Error";
    //print_r($id);
}??>

```

#### **logout.php // файл выхода из аккаунта**

```

<?php
if (ini_get("session.use_cookies")){
    $params = session_get_cookie_params();
    setcookie(session_name(), "", time() - 42000,
        $params["path"], $params["domain"],
        $params["secure"], $params["httponly"]);}
session_destroy();
unset($_SESSION['users']);
header("Location: /index.php");
?>

```

#### **reg.php // файл регистрации(добавления) сотрудника**

```

<?php
session_start();
$mysqli = new mysqli("datamon", "root", "", "datamon");
if ($mysqli->connect_error) {
    die('Ошибка : ('. $mysqli->connect_errno .') '. $mysqli->connect_error);
    exit();
}
$surname = trim($_POST['surname']);
$name = trim($_POST['name']);
$patron = trim($_POST['patron']);
$post = trim($_POST['post']);
$land = trim($_POST['land']);
$tel = trim($_POST['tel']);
$mail = trim($_POST['mail']);
$city = trim($_POST['city']);
$street = trim($_POST['street']);
$house = trim($_POST['house']);

```

|      |      |         |         |      |                                |      |
|------|------|---------|---------|------|--------------------------------|------|
|      |      |         |         |      | ТРПО 2-40 01 01.33.33.06.23 ПЗ | Лист |
| Изм. | Лист | №докум. | Подпись | Дата |                                | 67   |

```

$apart = trim($_POST['apart']);
$login = trim($_POST['login']);
$pass = trim($_POST['pass']);
$P_pas = trim($_POST['P_pas']);
if ($pass != $P_pas){
    echo("Пароли не совпадают");}
$md5_pass = md5($pass);
//$pattern_phone = '/^(+7|7|8)?[\s-]?(\(?[489][0-9]{2}\)?)[\s-]?[0-9]{3}[\s-]?[0-9]{2}[\s-]?[0-9]{2}$/';
$time = '08:00 — 18:00';
$time2 = '13:00 — 14:00';
$query = mysqli_query($mysqli, "SELECT * FROM `users` WHERE `login`='{ $login}'");
if (mysqli_num_rows($query) == 0) {
    mysqli_query($mysqli, "INSERT INTO `users` (`login`, `pass`) VALUES('{ $login}', '{ $md5_pass}')");
    mysqli_query($mysqli, "INSERT INTO `personal` (`surname`, `name`, `patron`, `post`, `tel`, `mail`, `city`,
`street`, `house`, `apart`) VALUES('{ $surname}', '{ $name}', '{ $patron}', '{ $post}', '{ $tel}', '{ $mail}', '{ $city}',
'{ $street}', '{ $house}', '{ $apart}')");
    mysqli_query($mysqli, "INSERT INTO `work` (`MonW`, `MonL`, `TueW`, `TueL`, `WedW`, `WedL`,
`ThuW`, `ThuL`, `FriW`, `FriL`, `SatW`, `SatL`, `SunW`, `SunL`) VALUES('{ $time}', '{ $time2}', '{ $time}',
'{ $time2}', '{ $time}', '{ $time2}', '{ $time}', '{ $time2}', '{ $time}', '{ $time}', 'Выходной',
'Выходной')");
    header("Location: /staff.php");
    exit();
} else {
    echo("Ошибка: Данный логин занят другим пользователем");
    exit();}
$mysqli->close();
?>

```

#### **updateStaff.php // файл редактирования данных сотрудника**

```

<?php
$mysqli = new mysqli("datamon", "root", "", "datamon");
if ($mysqli->connect_error) {
    die('Ошибка : ('. $mysqli->connect_errno .') '. $mysqli->connect_error);
    exit();}
$id_us = trim($_POST['id']);
$surname = trim($_POST['surname']);
$name = trim($_POST['name']);
$patron = trim($_POST['patron']);
$post = trim($_POST['post']);
$acc = trim($_POST['acc']);
$tel = trim($_POST['tel']);
$mail = trim($_POST['mail']);
$city = trim($_POST['city']);
$street = trim($_POST['street']);
$house = trim($_POST['house']);
$apart = trim($_POST['apart']);
if ($acc == "")
{
    $acc = 0;
}
mysqli_query($mysqli, "UPDATE `personal` SET `surname` = '{ $surname}', `name` = '{ $name}', `patron` =
'{ $patron}', `post` = '{ $post}', `tel` = '{ $tel}', `mail` = '{ $mail}', `city` = '{ $city}', `street` = '{ $street}', `house`
= '{ $house}', `apart` = '{ $apart}' WHERE `personal`.`id` = '{ $id_us}'");
mysqli_query($mysqli, "UPDATE `users` SET `access` = '{ $acc}' WHERE `users`.`id` = '{ $id_us}'");
header("Location: /staff.php");
?>

```

|      |      |         |         |      |                                |      |
|------|------|---------|---------|------|--------------------------------|------|
|      |      |         |         |      | ТРПО 2-40 01 01.33.33.06.23 ПЗ | Лист |
| Изм. | Лист | №докум. | Подпись | Дата |                                | 68   |

### updateTable.php // файл редактирования заказа

```
<?php
$mysqli = new mysqli("datamon", "root", "", "datamon");
if ($mysqli->connect_error) {
    die('Ошибка : ('. $mysqli->connect_errno .') '. $mysqli->connect_error);
    exit();
}
$book_id = trim($_POST['id']);
$surname = trim($_POST['surname']);
$name = trim($_POST['name']);
$patron = trim($_POST['patron']);
$numTable = trim($_POST['num']);
$book = trim($_POST['book']);
$book2 = trim($_POST['book2']);
$cost = trim($_POST['cost']);
$status = trim($_POST['stat']);
$status = mysqli_query($mysqli, "SELECT * FROM `status` WHERE `name` = '{ $status }'");
if (mysqli_num_rows($status) == 1)
{
    $status = mysqli_fetch_assoc($status);
    $status = $status['status'];
} else {
    $status = 1;
}
$pers = $_POST['pers'];
$id_pers = $_POST['id_pers'];
mysqli_query($mysqli, "UPDATE `booking` SET `surname` = '{ $surname }', `name` = '{ $name }', `patron` = '{ $patron }', `numTable` = '{ $numTable }', `book` = '{ $book }', `book2` = '{ $book2 }', `cost` = '{ $cost }', `status` = '{ $status }' WHERE `booking`.`id` = '{ $book_id }'");
header("Location: /booking.php");
?>
```

### uploadImg.php // файл загрузка изображения

```
<?php
session_start();
$login = $_SESSION['users']['login'];
$mysqli = new mysqli("datamon", "root", "", "datamon");
if ($mysqli->connect_error) {
    die('Ошибка : ('. $mysqli->connect_errno .') '. $mysqli->connect_error);
    exit();
}
$query = mysqli_query($mysqli, "SELECT * FROM `users` WHERE `login` = '{ $login }'");
$id = mysqli_fetch_assoc($query);
if(!empty($_FILES['img_upload']['tmp_name'])) $img =
addslashes(file_get_contents($_FILES['img_upload']['tmp_name']));
$query = mysqli_query($mysqli, "SELECT * FROM `images` WHERE `id` = '{ $id[id] }'");
if (mysqli_num_rows($query) == 1){
    mysqli_query($mysqli, "UPDATE `images` SET `image` = '{ $img }' WHERE `images`.`id` = '{ $id[id] }'");
} else{
    mysqli_query($mysqli, "INSERT INTO `images` (`id`, `image`) VALUES('{ $id[id] }', '{ $img }')");
}
header("Location: /cabinet.php");
?>
```

### Файлы JavaScript

#### ind\_mod\_auth.js // файл модального окна

|      |      |         |         |      |                                |      |
|------|------|---------|---------|------|--------------------------------|------|
|      |      |         |         |      | ТРПО 2-40 01 01.33.33.06.23 ПЗ | Лист |
| Изм. | Лист | №докум. | Подпись | Дата |                                | 69   |





```
for (var ci=0;ci<allspans.length;ci++) {  
    if (allspans[ci].className == 'sortarrow') {  
        if (getParent(allspans[ci],"table") == getParent(lnk,"table")) { // Условие: в той же таблице, что и мы?  
            allspans[ci].innerHTML = '&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&';  
        }  
    }  
}  
  
span.innerHTML = ARROW;  
}  
  
function getParent(el, pTagName) {  
    if (el == null) return null;  
    else if (el.nodeType == 1 && el.tagName.toLowerCase() == pTagName.toLowerCase()) //  
Gecko bug, supposed to be uppercase  
        return el;  
    else  
        return getParent(el.parentNode, pTagName);  
}  
function ts_sort_date(a,b) {  
    // Примечания: двузначные годы меньше 50 рассматриваются как 20XX, больше 50 рассматриваются  
как 19XX  
    aa = ts_getInnerText(a.cells[SORT_COLUMN_INDEX]);  
    bb = ts_getInnerText(b.cells[SORT_COLUMN_INDEX]);  
    if (aa.length == 10) {  
        dt1 = aa.substr(6,4)+aa.substr(3,2)+aa.substr(0,2);  
    } else {  
        yr = aa.substr(6,2);  
        if (parseInt(yr) < 50) { yr = '20'+yr; } else { yr = '19'+yr; }  
        dt1 = yr+aa.substr(3,2)+aa.substr(0,2);  
    }  
    if (bb.length == 10) {  
        dt2 = bb.substr(6,4)+bb.substr(3,2)+bb.substr(0,2);  
    } else {  
        yr = bb.substr(6,2);  
        if (parseInt(yr) < 50) { yr = '20'+yr; } else { yr = '19'+yr; }  
        dt2 = yr+bb.substr(3,2)+bb.substr(0,2);  
    }  
    if (dt1==dt2) return 0;  
    if (dt1<dt2) return -1;  
    return 1;  
}  
function ts_sort_currency(a,b) {  
    aa = ts_getInnerText(a.cells[SORT_COLUMN_INDEX]).replace(/^[^0-9.]/g,"");  
    bb = ts_getInnerText(b.cells[SORT_COLUMN_INDEX]).replace(/^[^0-9.]/g,"");  
    return parseFloat(aa) - parseFloat(bb);  
}  
function ts_sort_numeric(a,b) {  
    aa = parseFloat(ts_getInnerText(a.cells[SORT_COLUMN_INDEX]));  
    if (isNaN(aa)) aa = 0;  
    bb = parseFloat(ts_getInnerText(b.cells[SORT_COLUMN_INDEX]));  
    if (isNaN(bb)) bb = 0;  
    return aa-bb;  
}  
function ts_sort_caseinsensitive(a,b) {  
    aa = ts_getInnerText(a.cells[SORT_COLUMN_INDEX]).toLowerCase();
```

|      |      |          |         |      |                                |      |
|------|------|----------|---------|------|--------------------------------|------|
|      |      |          |         |      | ТРПО 2-40 01 01.33.33.06.23 ПЗ | Лист |
| Изм. | Лист | № док-м. | Подпись | Дата |                                | 72   |

```

    bb = ts_getInnerText(b.cells[SORT_COLUMN_INDEX]).toLowerCase();
    if (aa==bb) return 0;
    if (aa<bb) return -1;
    return 1;
}
function ts_sort_default(a,b) {
    aa = ts_getInnerText(a.cells[SORT_COLUMN_INDEX]);
    bb = ts_getInnerText(b.cells[SORT_COLUMN_INDEX]);
    if (aa==bb) return 0;
    if (aa<bb) return -1;
    return 1;
}
function addEvent(elm, evType, fn, useCapture)
// Добавление события и удаление события
{
    if (elm.addEventListener){
        elm.addEventListener(evType, fn, useCapture);
        return true;
    } else if (elm.attachEvent){
        var r = elm.attachEvent("on"+evType, fn);
        return r;
    } else {
        alert("Handler could not be removed");
    }
}

```

|      |      |         |         |      |                                |      |
|------|------|---------|---------|------|--------------------------------|------|
|      |      |         |         |      | ТРПО 2-40 01 01.33.33.06.23 ПЗ | Лист |
| Изм. | Лист | №докум. | Подпись | Дата |                                | 73   |