**3 ОПИСАНИЕ ПРОГРАММНОЙ РЕАЛИЗАЦИИ**

**3.1 Описание структуры базы данных**

В соответствии с физической моделью БД, была создана структура таблиц СУБД MySQL. Далее представлен текст из консоли MySQL-сервера, показывающий, в каком виде создавались и хранятся таблицы данной БД:

CREATE DATABASE IF NOT EXISTS `pizzeria` /\*!40100 DEFAULT CHARACTER SET utf8 \*/;

USE `pizzeria`;

-- MySQL dump 10.13 Distrib 8.0.13, for Win64 (x86\_64)

--

-- Host: 127.0.0.1 Database: pizzeria

-- ------------------------------------------------------

-- Server version 8.0.13

/\*!40101 SET @OLD\_CHARACTER\_SET\_CLIENT=@@CHARACTER\_SET\_CLIENT \*/;

/\*!40101 SET @OLD\_CHARACTER\_SET\_RESULTS=@@CHARACTER\_SET\_RESULTS \*/;

/\*!40101 SET @OLD\_COLLATION\_CONNECTION=@@COLLATION\_CONNECTION \*/;

SET NAMES utf8 ;

/\*!40103 SET @OLD\_TIME\_ZONE=@@TIME\_ZONE \*/;

/\*!40103 SET TIME\_ZONE='+00:00' \*/;

/\*!40014 SET @OLD\_UNIQUE\_CHECKS=@@UNIQUE\_CHECKS, UNIQUE\_CHECKS=0 \*/;

/\*!40014 SET @OLD\_FOREIGN\_KEY\_CHECKS=@@FOREIGN\_KEY\_CHECKS, FOREIGN\_KEY\_CHECKS=0 \*/;

/\*!40101 SET @OLD\_SQL\_MODE=@@SQL\_MODE, SQL\_MODE='NO\_AUTO\_VALUE\_ON\_ZERO' \*/;

/\*!40111 SET @OLD\_SQL\_NOTES=@@SQL\_NOTES, SQL\_NOTES=0 \*/;

--

-- Table structure for table `categorydrink`

--

DROP TABLE IF EXISTS `categorydrink`;

/\*!40101 SET @saved\_cs\_client = @@character\_set\_client \*/;

SET character\_set\_client = utf8mb4 ;

CREATE TABLE `categorydrink` (

`Id` int(11) NOT NULL AUTO\_INCREMENT,

`Names` varchar(20) NOT NULL,

`Enable` int(11) NOT NULL DEFAULT '1',

PRIMARY KEY (`Id`)

) ENGINE=InnoDB AUTO\_INCREMENT=17 DEFAULT CHARSET=utf8;

/\*!40101 SET character\_set\_client = @saved\_cs\_client \*/;

--

-- Table structure for table `categoryeat`

--

DROP TABLE IF EXISTS `categoryeat`;

/\*!40101 SET @saved\_cs\_client = @@character\_set\_client \*/;

SET character\_set\_client = utf8mb4 ;

CREATE TABLE `categoryeat` (

`Id` int(11) NOT NULL AUTO\_INCREMENT,

`Names` varchar(20) NOT NULL,

`Enable` int(11) NOT NULL DEFAULT '1',

PRIMARY KEY (`Id`)

) ENGINE=InnoDB AUTO\_INCREMENT=7 DEFAULT CHARSET=utf8;

/\*!40101 SET character\_set\_client = @saved\_cs\_client \*/;

--

-- Table structure for table `checks`

--

DROP TABLE IF EXISTS `checks`;

/\*!40101 SET @saved\_cs\_client = @@character\_set\_client \*/;

SET character\_set\_client = utf8mb4 ;

CREATE TABLE `checks` (

`Id` int(11) NOT NULL AUTO\_INCREMENT,

`Customer` varchar(45) CHARACTER SET utf8 COLLATE utf8\_general\_mysql500\_ci NOT NULL,

`Table` int(11) DEFAULT NULL,

`Employeer` int(11) NOT NULL,

`DateOrder` datetime NOT NULL,

`DateGive` datetime NOT NULL,

`Adres` varchar(150) CHARACTER SET utf8 COLLATE utf8\_general\_mysql500\_ci DEFAULT NULL,

`Status` enum('Принят','Готовится','Доставляется','Выдан','Оплачен','Отменен') CHARACTER SET utf8 COLLATE utf8\_general\_mysql500\_ci NOT NULL,

`CardKey` int(11) DEFAULT NULL,

`Enable` int(11) NOT NULL DEFAULT '1',

PRIMARY KEY (`Id`),

KEY `Table\_idx` (`Table`),

KEY `Employeer\_idx` (`Employeer`),

KEY `CardKey\_idx` (`CardKey`),

CONSTRAINT `CardKey` FOREIGN KEY (`CardKey`) REFERENCES `customers` (`id`),

CONSTRAINT `Employeer` FOREIGN KEY (`Employeer`) REFERENCES `employeers` (`id`),

CONSTRAINT `Table` FOREIGN KEY (`Table`) REFERENCES `tables` (`id`)

) ENGINE=InnoDB AUTO\_INCREMENT=58 DEFAULT CHARSET=utf8 COLLATE=utf8\_general\_mysql500\_ci;

/\*!40101 SET character\_set\_client = @saved\_cs\_client \*/;

--

-- Table structure for table `customers`

--

DROP TABLE IF EXISTS `customers`;

/\*!40101 SET @saved\_cs\_client = @@character\_set\_client \*/;

SET character\_set\_client = utf8mb4 ;

CREATE TABLE `customers` (

`Id` int(11) NOT NULL AUTO\_INCREMENT,

`Lname` varchar(20) NOT NULL,

`Fname` varchar(20) NOT NULL,

`Adres` varchar(150) NOT NULL,

`Procent` int(11) NOT NULL,

`Enable` int(11) NOT NULL DEFAULT '1',

PRIMARY KEY (`Id`)

) ENGINE=InnoDB AUTO\_INCREMENT=9 DEFAULT CHARSET=utf8;

/\*!40101 SET character\_set\_client = @saved\_cs\_client \*/;

--

-- Table structure for table `drink`

--

DROP TABLE IF EXISTS `drink`;

/\*!40101 SET @saved\_cs\_client = @@character\_set\_client \*/;

SET character\_set\_client = utf8mb4 ;

CREATE TABLE `drink` (

`Id` int(11) NOT NULL AUTO\_INCREMENT,

`Names` varchar(30) NOT NULL,

`Volume` int(10) unsigned NOT NULL,

`Cost` int(10) unsigned NOT NULL,

`Category` int(11) NOT NULL,

`CanShop` int(11) NOT NULL,

`Enable` int(11) NOT NULL DEFAULT '1',

PRIMARY KEY (`Id`),

KEY `Category\_idx` (`Category`),

CONSTRAINT `CategoryDrink` FOREIGN KEY (`Category`) REFERENCES `categorydrink` (`id`)

) ENGINE=InnoDB AUTO\_INCREMENT=16 DEFAULT CHARSET=utf8;

/\*!40101 SET character\_set\_client = @saved\_cs\_client \*/;

--

-- Table structure for table `eat`

--

DROP TABLE IF EXISTS `eat`;

/\*!40101 SET @saved\_cs\_client = @@character\_set\_client \*/;

SET character\_set\_client = utf8mb4 ;

CREATE TABLE `eat` (

`Id` int(11) NOT NULL AUTO\_INCREMENT,

`Names` varchar(30) NOT NULL,

`Portion` enum('Маленькая','Стандарт','Большая') NOT NULL,

`Cost` int(10) unsigned NOT NULL,

`Mass` int(10) unsigned NOT NULL,

`Kaloryes` int(10) unsigned NOT NULL,

`Category` int(11) NOT NULL,

`CanShop` int(11) NOT NULL,

`Enable` int(11) NOT NULL DEFAULT '1',

PRIMARY KEY (`Id`),

KEY `Category\_idx` (`Category`),

CONSTRAINT `CategoryEat` FOREIGN KEY (`Category`) REFERENCES `categoryeat` (`id`)

) ENGINE=InnoDB AUTO\_INCREMENT=17 DEFAULT CHARSET=utf8;

/\*!40101 SET character\_set\_client = @saved\_cs\_client \*/;

--

-- Table structure for table `employeers`

--

DROP TABLE IF EXISTS `employeers`;

/\*!40101 SET @saved\_cs\_client = @@character\_set\_client \*/;

SET character\_set\_client = utf8mb4 ;

CREATE TABLE `employeers` (

`Id` int(11) NOT NULL AUTO\_INCREMENT,

`Lname` varchar(20) NOT NULL,

`Fname` varchar(20) NOT NULL,

`Sname` varchar(20) NOT NULL,

`Born` date NOT NULL,

`Number` varchar(15) NOT NULL,

`Adres` varchar(150) NOT NULL,

`Job` int(11) NOT NULL,

`Days` enum('Четные','Нечетные') NOT NULL,

`Password` varchar(10) DEFAULT NULL,

`Enable` int(11) NOT NULL DEFAULT '1',

PRIMARY KEY (`Id`),

UNIQUE KEY `Password\_UNIQUE` (`Password`),

KEY `Job\_idx` (`Job`),

CONSTRAINT `Job` FOREIGN KEY (`Job`) REFERENCES `job` (`id`)

) ENGINE=InnoDB AUTO\_INCREMENT=28 DEFAULT CHARSET=utf8;

/\*!40101 SET character\_set\_client = @saved\_cs\_client \*/;

--

-- Table structure for table `ingredients`

--

DROP TABLE IF EXISTS `ingredients`;

/\*!40101 SET @saved\_cs\_client = @@character\_set\_client \*/;

SET character\_set\_client = utf8mb4 ;

CREATE TABLE `ingredients` (

`Id` int(11) NOT NULL AUTO\_INCREMENT,

`Dish` int(11) DEFAULT NULL,

`Brew` int(11) DEFAULT NULL,

`Product` int(11) NOT NULL,

`Count` int(10) unsigned NOT NULL,

PRIMARY KEY (`Id`),

KEY `Brew\_idx` (`Brew`),

KEY `Dish\_idx` (`Dish`),

KEY `Product\_idx` (`Product`),

CONSTRAINT `Brews` FOREIGN KEY (`Brew`) REFERENCES `drink` (`id`),

CONSTRAINT `Dish` FOREIGN KEY (`Dish`) REFERENCES `eat` (`id`),

CONSTRAINT `Product` FOREIGN KEY (`Product`) REFERENCES `products` (`id`)

) ENGINE=InnoDB AUTO\_INCREMENT=77 DEFAULT CHARSET=utf8;

/\*!40101 SET character\_set\_client = @saved\_cs\_client \*/;

--

-- Table structure for table `job`

--

DROP TABLE IF EXISTS `job`;

/\*!40101 SET @saved\_cs\_client = @@character\_set\_client \*/;

SET character\_set\_client = utf8mb4 ;

CREATE TABLE `job` (

`Id` int(11) NOT NULL AUTO\_INCREMENT,

`Names` varchar(30) NOT NULL,

`Enable` int(11) NOT NULL DEFAULT '1',

PRIMARY KEY (`Id`)

) ENGINE=InnoDB AUTO\_INCREMENT=18 DEFAULT CHARSET=utf8;

/\*!40101 SET character\_set\_client = @saved\_cs\_client \*/;

--

-- Table structure for table `orders`

--

DROP TABLE IF EXISTS `orders`;

/\*!40101 SET @saved\_cs\_client = @@character\_set\_client \*/;

SET character\_set\_client = utf8mb4 ;

CREATE TABLE `orders` (

`Id` int(11) NOT NULL AUTO\_INCREMENT,

`Dish` int(11) DEFAULT NULL,

`Brew` int(11) DEFAULT NULL,

`Count` int(10) unsigned NOT NULL,

`Check` int(11) NOT NULL,

PRIMARY KEY (`Id`),

KEY `DishOrder\_idx` (`Dish`),

KEY `DrinkOrder\_idx` (`Brew`),

KEY `Check\_idx` (`Check`),

CONSTRAINT `Check` FOREIGN KEY (`Check`) REFERENCES `checks` (`id`),

CONSTRAINT `DishOrder` FOREIGN KEY (`Dish`) REFERENCES `eat` (`id`),

CONSTRAINT `DrinkOrder` FOREIGN KEY (`Brew`) REFERENCES `drink` (`id`)

) ENGINE=InnoDB AUTO\_INCREMENT=67 DEFAULT CHARSET=utf8;

/\*!40101 SET character\_set\_client = @saved\_cs\_client \*/;

--

-- Table structure for table `products`

--

DROP TABLE IF EXISTS `products`;

/\*!40101 SET @saved\_cs\_client = @@character\_set\_client \*/;

SET character\_set\_client = utf8mb4 ;

CREATE TABLE `products` (

`Id` int(11) NOT NULL AUTO\_INCREMENT,

`Names` varchar(45) NOT NULL,

`Count` int(10) unsigned NOT NULL,

`Ones` enum('шт','кг','л') NOT NULL,

`MinCount` int(10) unsigned NOT NULL,

`Enable` int(11) NOT NULL DEFAULT '1',

PRIMARY KEY (`Id`),

UNIQUE KEY `Names\_UNIQUE` (`Names`)

) ENGINE=InnoDB AUTO\_INCREMENT=27 DEFAULT CHARSET=utf8;

/\*!40101 SET character\_set\_client = @saved\_cs\_client \*/;

--

-- Table structure for table `tables`

--

DROP TABLE IF EXISTS `tables`;

/\*!40101 SET @saved\_cs\_client = @@character\_set\_client \*/;

SET character\_set\_client = utf8mb4 ;

CREATE TABLE `tables` (

`Id` int(11) NOT NULL AUTO\_INCREMENT,

`Names` varchar(45) NOT NULL,

`CanTake` int(11) NOT NULL DEFAULT '1',

`Enable` int(11) NOT NULL DEFAULT '1',

PRIMARY KEY (`Id`)

) ENGINE=InnoDB AUTO\_INCREMENT=15 DEFAULT CHARSET=utf8;

/\*!40101 SET character\_set\_client = @saved\_cs\_client \*/;

--

-- Dumping events for database 'pizzeria'

--

--

-- Dumping routines for database 'pizzeria'

--

/\*!40103 SET TIME\_ZONE=@OLD\_TIME\_ZONE \*/;

/\*!40101 SET SQL\_MODE=@OLD\_SQL\_MODE \*/;

/\*!40014 SET FOREIGN\_KEY\_CHECKS=@OLD\_FOREIGN\_KEY\_CHECKS \*/;

/\*!40014 SET UNIQUE\_CHECKS=@OLD\_UNIQUE\_CHECKS \*/;

/\*!40101 SET CHARACTER\_SET\_CLIENT=@OLD\_CHARACTER\_SET\_CLIENT \*/;

/\*!40101 SET CHARACTER\_SET\_RESULTS=@OLD\_CHARACTER\_SET\_RESULTS \*/;

/\*!40101 SET COLLATION\_CONNECTION=@OLD\_COLLATION\_CONNECTION \*/;

/\*!40111 SET SQL\_NOTES=@OLD\_SQL\_NOTES \*/;

-- Dump completed on 2019-03-10 17:12:15

**3.1 Описание задач автоматизации интерфейса**

Чтобы приступить к работе с приложением, для начала нужно запустить файл CourseWork. При открытии данного файла, появится окно авторизации.

Окно авторизации имеет следующий вид:

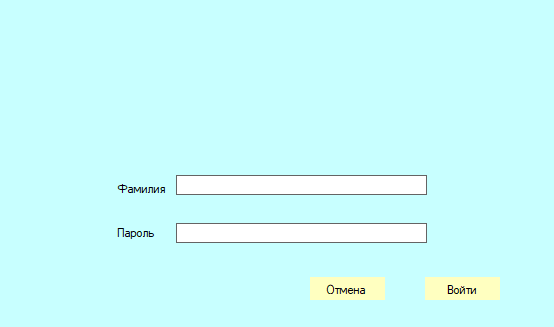


Рисунок 3.1 – Окно «Авторизация»

Данное окно содержит: поле для ввода пароля и логина, кнопки выхода и входа в систему.

После заполнения пользователем полей, ему необходимо нажать на кнопку «Войти» и система проверит правильность вводимых данных и, если все верно, перенаправит его на окно «Главное меню», в случае неправильности вводимых данных система известит пользователя об этом и потребует повторного заполнения полей.

Окно «Главное меню» загружается после авторизации и, в зависимости от должности пользователя может иметь разный функционал.

Главное меню со всеми возможными функциями имеет следующий вид: 

Рисунок 3.2- Окно «Главное меню»

Данное окно предназначено для организации простого и удобного доступа ко всем необходимым для работы действиям. Каждая кнопка отвечает за переход к конкретному окну для работы, кнопка «Закрыть» завершает работу с системой.

После нажатия на кнопку «Сотрудники» пользователь увидит следующую форму:

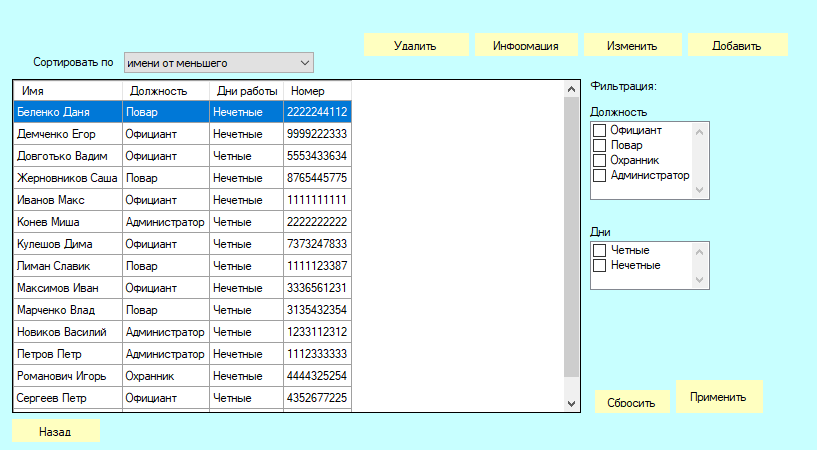


Рисунок 3.3 – Окно «Сотрудники»

Данное окно имеет следующий функционал:

1. Просмотр всех сотрудников заведения с их должностями и днями работы
2. Сортировка по имени, должности и дням работы.
3. Фильтрация по должности и дням работы.
4. После нажатия на кнопку «Информация», пользователь увидит детальные сведенья о выбранном сотруднике.
5. Кнопка «Удалить» отвечает за удаление всей информации о уволенном сотруднике.
6. Кнопки «Добавить» и «Изменить» перенаправляют администратора на новое окно в котором он может добавить нового сотрудника или изменить информацию о уже имеющемся

Окно для добавления пользователя имеет следующий вид:

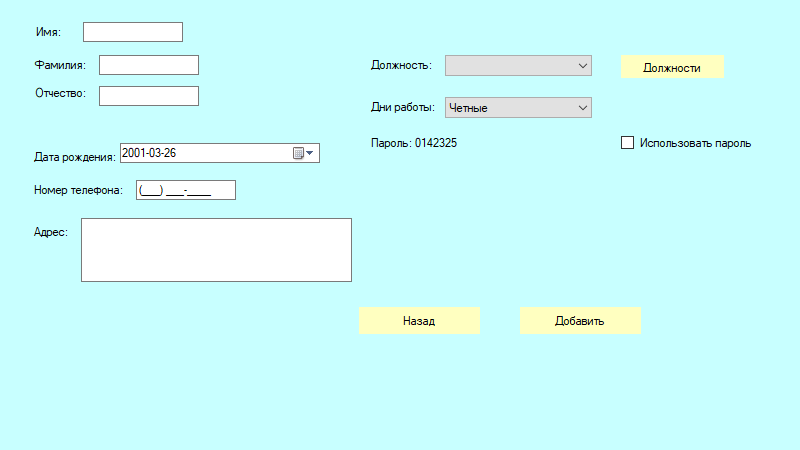


Рисунок 3.4 – Окно «Добавить пользователя»

В данном окне пользователю нужно заполнить все поля и нажать кнопку «Добавить» для сохранения информации о сотруднике, после чего, система вернет его на окно «Сотрудники» в котором будет уже обновленная информация.

После нажатия на кнопку «Блюда» в главном меню сотрудник увидит следующее окно:



Рисунок 3.5 – Окно «Блюда»

В данном окне отображается все блюда которые готовятся в пиццерии, а также предоставляется возможность фильтрации блюд по типу, весу, цене, порции и инградиентам входящих в их состав. Поле сортировки поможет выбрать критерий по которому следует отсортировать блюда.

Окно также иммет следующие основные кнопки:

1. Кнопка удаления блюда – после нажатия на нее система запросит подтверждение удаления выбраного блюда и, после согласия пользователя, удалит всю информацию о нем.
2. Кнопки «Добавить» и «Изменить» необходимы для добавления нового блюда и изменения информпции об уже имеющемся.

Окно добавления после нажатия на кнопку «Добавить» имеет следующий вид:

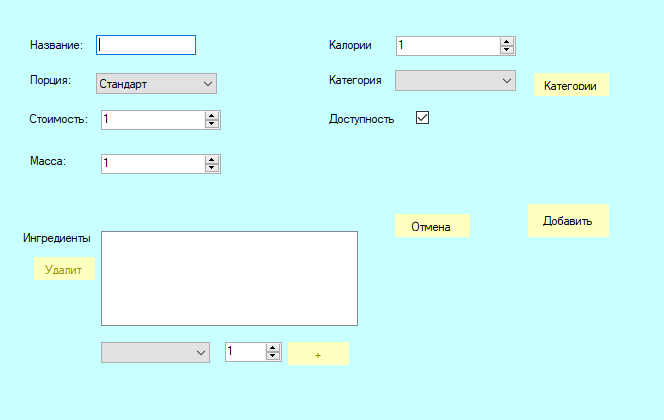


Рисунок 3.6 – Окно «Добавить болюдо»

В случае нажатия «Изменить» форма будет данными выделеного блюда:

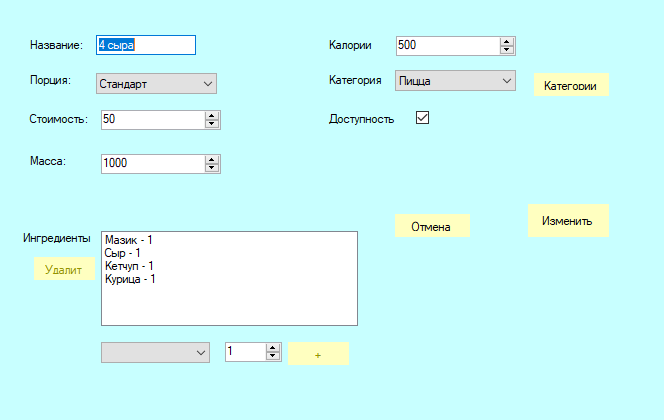


Рисунок 3.7 – Окно «Изменить блюдо»

При работе с данными окнами сотруднику нужно только заполнить поля и нажать кнопку «Добавить» или «Измненить» и система занесет новые данные в базу данны и обновить данные в окне.

Окно «Напитки» имеет функционал аналогичный форме с блюдами, с учетом некоторой корректировки под специфику напитков, оно имеет следующий вид:

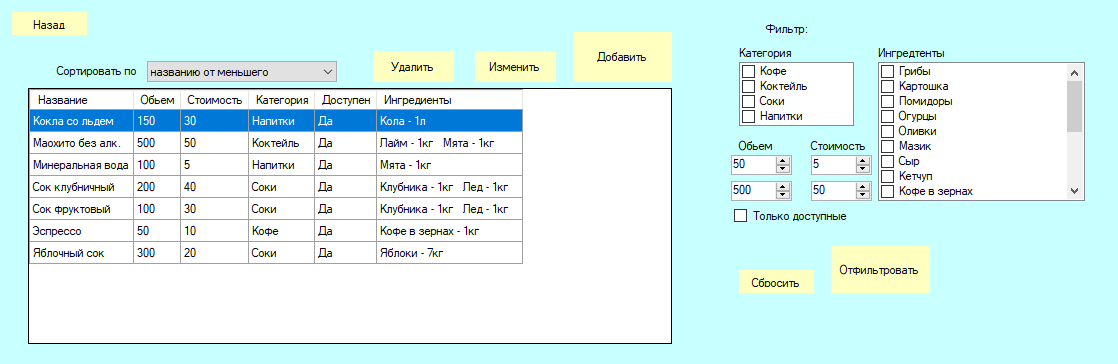


Рисунок 3.8 –Окно «Напитки»

Система также позволяет сотруднику увидеть статитсику, финансовые отчеты и рейтинг продаваемых товаров и блюд.

Окно статистики содержит следуюющую информацию:

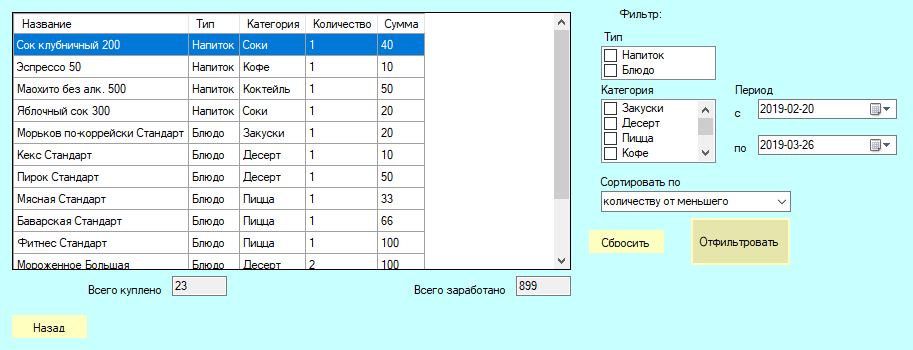


Рисунок 3.9 – Окно «Статистика»

На данной странице пользователем доступна информация о всех проданных блюдах и полученной прибыли за конкретыне периоды, также можно отфильтровать данные для удобства.

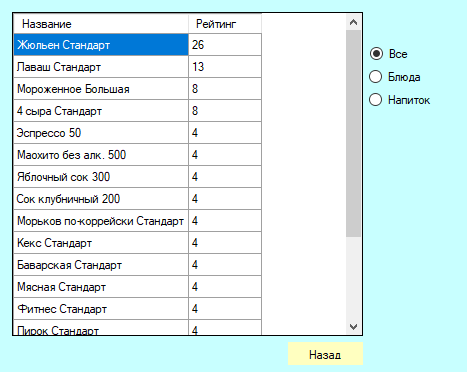


Рисунок 3.10 – Окно «Рейтинг»

Окно рейтинга показывает рейтинг блюд и напитков, который высчитывается по формуле: количствое продаж данного блюда разделить на количнство продаж всех блюд и умножить на 100. Данная информация полезна для оценки рентабельности и популярности продаваемых блюд.

Для оформления заказа официанту необходимо нажать на кнопку «Оформить заказ», после этого он будет перенапрвлен на специальное окно:

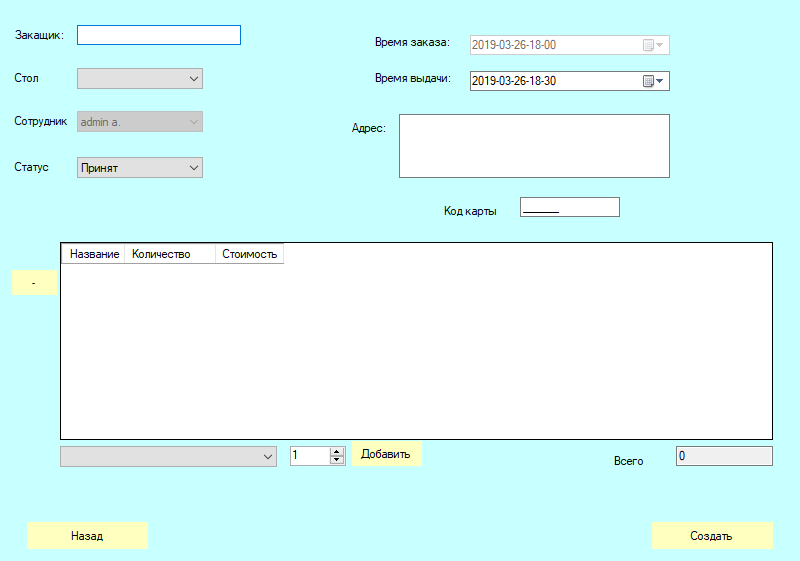


Рисунок 3.11 – окно «Принять заказ»

Вданном окне официант заполняет все необходимые поля и выбирает товары, которые заказал клиент, после чего отправляет данные на оформление и создается новый заказ.

Управлять заказми можно через окно со списком заказов, которое имеет следующий вид:

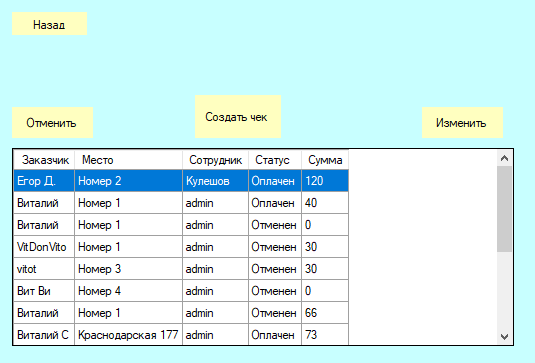


Рисунок 3.12 – Окно «Заказы»

Данное окно имеет следующий функционал:

1. Отмена выбраного заказа.
2. Создание чека для печати определенного заказа.
3. Изменения заказа, если позволяет его статус.

Также система предоставляет данные по складу с продуктами:

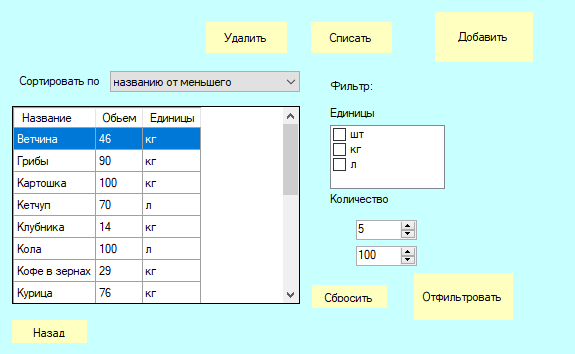


Рисунок 3.13 – Окно «Продукты»

В данном окне сотрудник может посмотреть список дотупных товаров на складе и их количество и закупить их при необходимости.

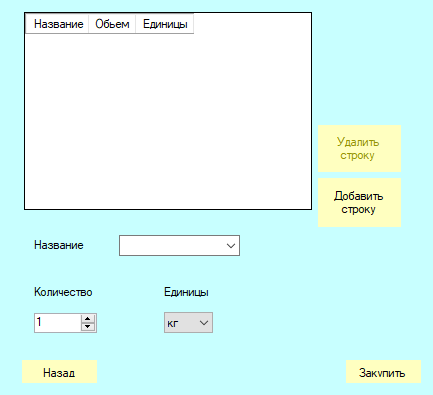


Рисунок 3.14 – Окно «Закупить продукты»

В этом окне сотрудник создает список покупок выбирая необходимые товары и их количество, после чего нажав на кнопку «Закупить», произойдет закупка и обновится список продуктов на складе.