**2 ПРОЕКТИРОВАНИЕ БАЗЫ ДАННЫХ**

**2.1 Построение ER-диаграммы.**

На основе выполненного анализа предметной области была построена ER-диаграмма (рисунок 2.1).

Столик

Статус

Карта

Код

Время выд.

Заказчик

Адрес

Сотрудник

Время прин.

Код

ФИО

Дата рожд.

Номер тел.

Адрес

Чек

Работники

Товары

Состав

Заказ

Должность

Оклад

Дни работы

Пароль

Код

Калорийн.

Название

Порция.

Номер тел.

Вес/объем

Категория

Наличие

Код

Блюдо

Продукт

Количество

Код

Блюдо

Колич.

Категор.

Клиент

Код карты

ФИО

Адрес

Скидка

Рисунок 2.1- ER-диаграмма

**2.2 Построение схемы реляционной модели базы данных.**

На основании проведенного анализа предметной области и построенной ER- диаграммы создана реляционная модель базы данных (рисунок 2.2), которая отвечает всем требованиям нормализации отношений.

Id

Ингредиенты

Продукт

Напиток

Id

Категория

Блюдо

Категория

Id

Id

Работники

Id

Должность

Id

Чек

Заказ

Id

Заказчик

Id

Стол

Id

Id

Id

Id

Id

Id

Id

Рисунок 2.2 –Реляционная модель БД

Таблица2.1- Отношение КАТЕГОРИИ НАПИТКОВ

|  |  |
| --- | --- |
| Код категории | Категория напитка |

Таблица2.2.- Отношение КАТЕГОРИИ БЛЮД

|  |  |
| --- | --- |
| Код категории | Категория блюда |

Таблица2.3- Отношение ЧЕК

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Код чека | Имя закащика | Номер стола | Код сотрудника | Время принятия | Время выдачи | Адрес | Статус | Код карточки на скидку |

Таблица2.4- Отношение КАРТА НА СКИДКУ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Код карты | Фамилия | Имя | Адрес | Процент скидки |

Таблица2.5- Отношение БЛЮДА

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Код напитка | Название | Стоимость | Объем | Код категории | Можно ли купить |

Таблица2.6- Отношение НАПИТКИ

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Код блюда | Название | Порция | Стоимость | Масса | Калории | Код категории | Можно ли купить |

Таблица2.7- Отношение СОТРУДНИКИ

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Код сотрудника | Фамилия | Имя | Отчество | Дата рождения | Номер | Адрес | Код должности | Оклад | Дни работы | Пароль |

Таблица2.8- Отношение СОСТАВ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Код записи | Код блюда | Код напитка | Код продукта | Количество |

Таблица2.9- Отношение ДОЛЖНОСТИ

|  |  |
| --- | --- |
| Код должности | Название |

Таблица2.10- Отношение ЗАКАЗ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Код записи | Код блюда | Код напитка | Код чека | Количество |

Таблица2.11- Отношение ПРОДУКТЫ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Код продукта | Название | Количество | Единицы | Мин. количество |

Таблица2.12- Отношение СТОЛЫ

|  |  |
| --- | --- |
| Код стола | Название |

2**.3 Описание физической модели базы данных.**

В качестве СУБД была выбрана MySQL и визуальное приложение для работы с ней Workbench. Первое, что стоит отметить – Workbench дает возможность визуально проектировать базу данных, т.е. составлять схему БД. Визуальное представление БД всегда дает куда большую информацию, чем сухой список таблиц. В таком варианте сразу видно, каким образом связаны между собой таблицы, можно группировать таблицы по каким либо параметрами отражать это на схеме. Во-вторых, программа имеет встроенный редактор SQL-кода, с помощью которого можно быстро внести любые правки в SQL-запросы. При этом возможно строить запросы любой сложности, получать различные выборки из таблиц, связывать их, создавать новые таблицы и редактировать существующие, работать с ключами, полями, записями. В качестве языка был выбран C#.

Выделяются такие достоинства данного языка:

1.С# является полностью объектно-ориентированным языком, где даже типы, встроенные в язык, представлены классами.

2. Реализация, сочетающая построение надежного и эффективного кода, является немаловажным фактором, способствующему успеху C#.

3. Мощная библиотека каркаса поддерживает удобство построения различных типов приложения на C#, позволяя легко строить Web-службы, другие виды компонентов, достаточно просто сохранять и получать информацию из базы данных и других хранилищ данных.

С учетом особенностей выбранной СУБД спроектирована физическая модель для создания ранее реляционной модели (таблица 2.1). Физические модели БД:

Таблица 2.13-Физическая модель таблицы «Сотрудники (Employeers)»

|  |  |  |  |
| --- | --- | --- | --- |
| Имя столбца | Тип | Ограничения | Имя |
| Id | Int | AI, NN, PK | Код |
| Lname | Varchar(20) | NN | Фамилия |
| Fname | Varchar(20) | NN | Имя |
| Sname | Varchar(20) | NN | Отчество |
| Born | Data | NN | Дата рождения |
| Number | Varchar(15) | NN, UN | Номер телефона |
| Adress | Varchar(150) | NN | Адрес |
| Money | Int | NN,US | Зарплата |
| Job | Int | FK,NN | Код должности |
| Days | Enum(‘Четные’, ‘Нечетные’) | NN | Дни работы |
| Password | Varchar(10) | NN,UN | Пароль |

Таблица 2.14-Физическая модель таблицы «Должности (Jobs)

|  |  |  |  |
| --- | --- | --- | --- |
| Имя столбца | Тип | Ограничения | Имя |
| Id | int | PK, NN, AI | Код |
| Job | Varchar(20) | NN,UN | Должность |

Таблица 2.15-Физическая модель таблицы «Меню напитков (Drink)»

|  |  |  |  |
| --- | --- | --- | --- |
| Имя столбца | Тип | Ограничения | Имя |
| Id | Int | PK,NN,AI | Код |
| Name | Varchar(20) | NN | Название |
| Cost | Int | US,NN | Стоимость |
| Litrs | Int | US,NN | Объем |
| Category | Int | FK,NN | Код категории |
| CanShop | Int | NN | Наличие |

Таблица 2.16-Физическая модель таблицы «Меню блюд (Eat)

|  |  |  |  |
| --- | --- | --- | --- |
| Имя столбца | Тип | Ограничения | Имя |
| Id | Int | PK,NN,AI | Код |
| Name | Varchar(20) | NN | Название |
| Portion | Enum(«Маленькая», «Стандарт», «Большая) | -//- | Размер |
| Cost | Int | US,NN | Стоимость |
| Mass | Int | US,NN | Масса |
| Kaloryes | Int | US,NN | Калории |
| Category | Int | FK,NN | Код категории |
| CanShop | Int | NN | Наличие |

Таблица 2.17-Физическая модель таблицы «Категории напитков (KategoryDrink)»

|  |  |  |  |
| --- | --- | --- | --- |
| Имя столбца | Тип | Ограничения | Имя |
| Id | int | PK,AI,NN | Код |
| Category | Varchar(20) | NN | Категория напитков |

Таблица 2.18-Физическая модель таблицы «Категории блюд (KategoryEat)»

|  |  |  |  |
| --- | --- | --- | --- |
| Имя столбца | Тип | Ограничения | Имя |
| Id | int | PK,AI,NN | Код |
| Category | Varchar(20) | NN | Категория блюд |

Таблица 2.19-Физическая модель таблицы «Данные для чека (Checks)»

|  |  |  |  |
| --- | --- | --- | --- |
| Имя столбца | Тип | Ограничения | Имя |
| Id | int | PK,AI,NN | Код |
| Customer | Varchar(30) | NN | Заказчик |
| Table | int | NN,FK | Код стола |
| Продолжение таблицы 2.19 | | | |
| Имя столбца | Тип | Ограничения | Имя |
| Employeer | Int | FK, NN | Код сотрудника |
| DateOrder | DateTime | NN | Дата и время принятия |
| DateGive | DateTime | NN | Дата и время доставки |
| Adress | Varchar (100) | -//- | Адрес доставки  (Если нужно) |
| Status | Enum () | NN | Статус заказа |
| Custome | int | FK | Карта клиента |

Таблица 2.20-Физическая модель таблицы «Заказ (Orderss)»

|  |  |  |  |
| --- | --- | --- | --- |
| Имя столбца | Тип | Ограничения | Имя |
| Id | int | PK,AI,NN | Код |
| Dish | int | FK | Код блюда |
| Brew | Int | FK | Код напитка |
| Count | int | NN,US, default(1) | Количество |
| Check | int | NN,FK | Код чека |

Таблица 2.21-Физическая модель таблицы «Продукты (Products)»

|  |  |  |  |
| --- | --- | --- | --- |
| Имя столбца | Тип | Ограничения | Имя |
| Id | int | PK,AI,NN | Код |
| Name | Varchar(20) | NN | Название |
| Count | Int | US,NN | Количество единиц |
| Ones | Enum(«л»,«кг»,«шт») | NN | Единицы измерения |

Таблица 2.22-Физическая модель таблицы «Состав (Ingredients)»

|  |  |  |  |
| --- | --- | --- | --- |
| Имя столбца | Тип | Ограничения | Имя |
| Id | int | PK,AI,NN | Код |
| Dish | int | FK | Код блюда |
| Brew | Int | FK | Код напитка |
| Count | int | NN,US, default(1) | Количество |
| Product | int | NN,FK | Код продукта |

Таблица 2.23-Физическая модель таблицы «Tables)»

|  |  |  |  |
| --- | --- | --- | --- |
| Имя столбца | Тип | Ограничения | Имя |
| Id | int | PK,AI,NN | Код |
| Name | Varchar(20) | NN | Название |
| CanTake | Bool | NN | Доступен ли |

Таблица 2.24-Физическая модель таблицы «Клиенты (Customers)»

|  |  |  |  |
| --- | --- | --- | --- |
| Имя столбца | Тип | Ограничения | Имя |
| Id | int | PK,AI,NN | Код |
| Lname | Varchar(20) | NN | Фамилия |
| Fname | Varchar(20) | NN | Имя |
| Adress | Varchar(150) | NN | Адрес |
| Procent | int | NN, US | Процент скидки |

Сокращения:

NN ­­­­-- Not null;

US -- Unsigned;

UQ -- Unique;

AI -- Auto increment;

PK -- Primary key;

FK -- Foreign key.

**2.4 Эскиз интерфейса**

Эскиз интерфейса - это визуальный набросок будущего приложения. Он дает возможность сгруппировать объекты и правильно распределить функции. От удобства интерфейса зависит то, насколько клиент «привяжется» к системе, насколько ему удобно будет работать. Интерфейс приложения дляработы с БДбудет разработан в виде форм с кнопками, полями и переключателями. Далее нарисунках представлены эскизы окон, с которыми будет работать пользователь во время эксплуатации приложения.

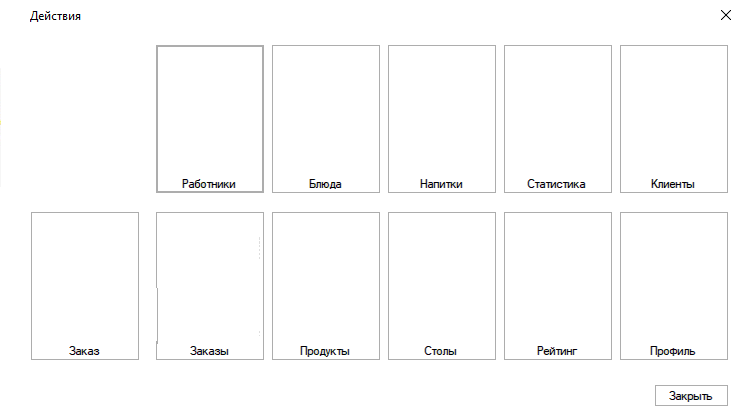


Рисунок 2.3 – Эскиз интерфейса главного меню

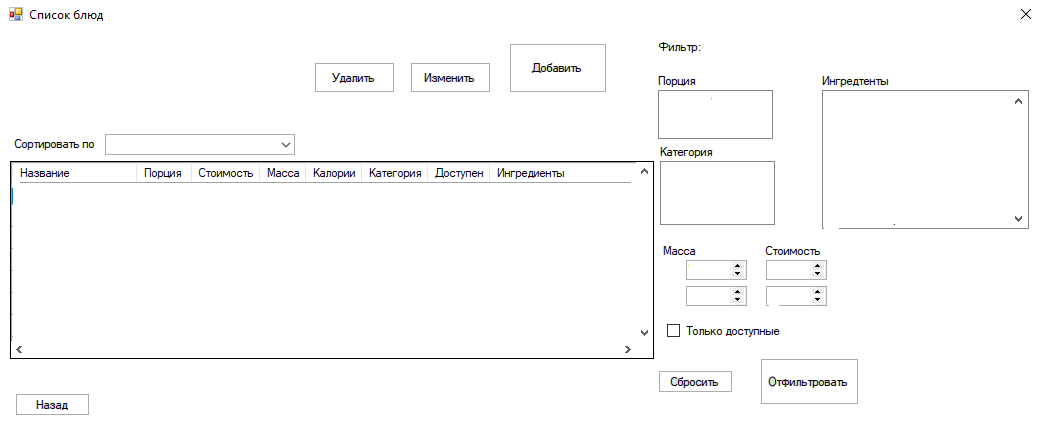


Рисунок 2.4 – Эскиз интерфейса списка блюд

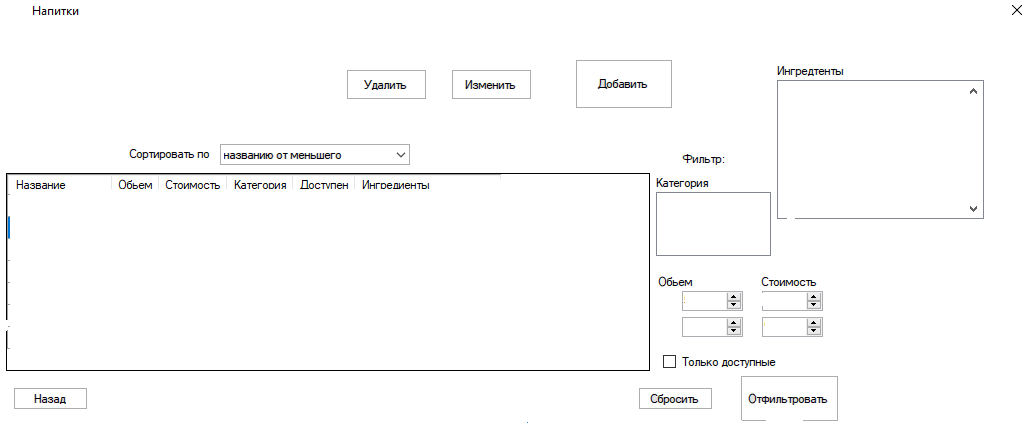


Рисунок 2.5 – Эскиз интерфейса списка напитков

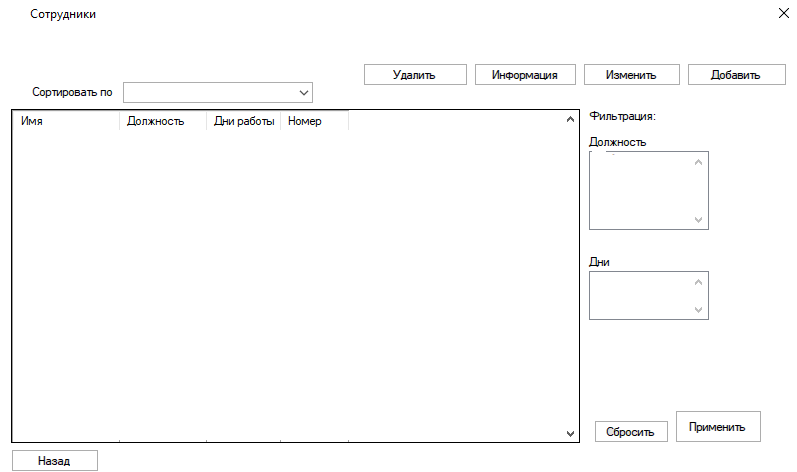


Рисунок 2.6 – Эскиз интерфейса списка сотрудников

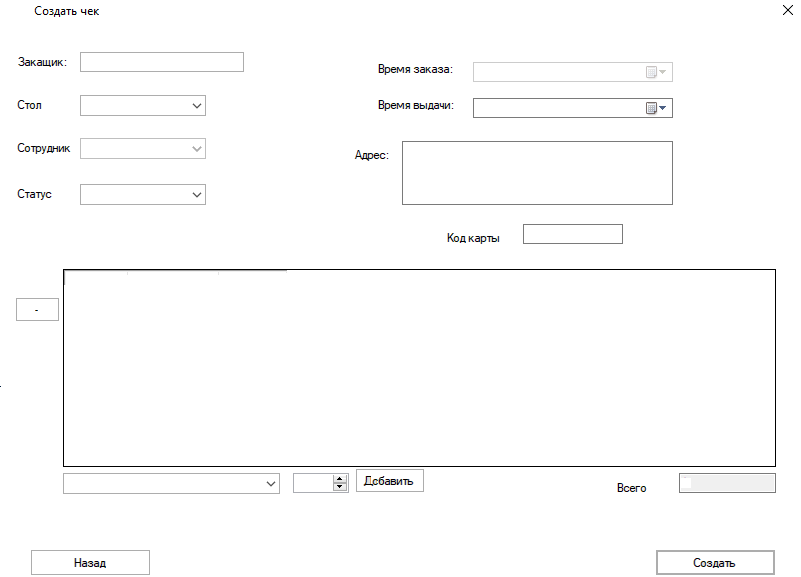


Рисунок 2.7 – Эскиз интерфейса создания нового заказа

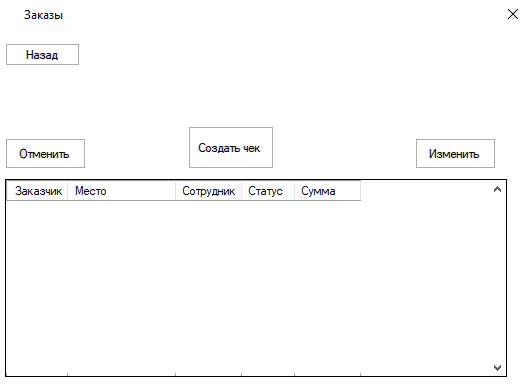


Рисунок 2.8 – Эскиз интерфейса списка заказов

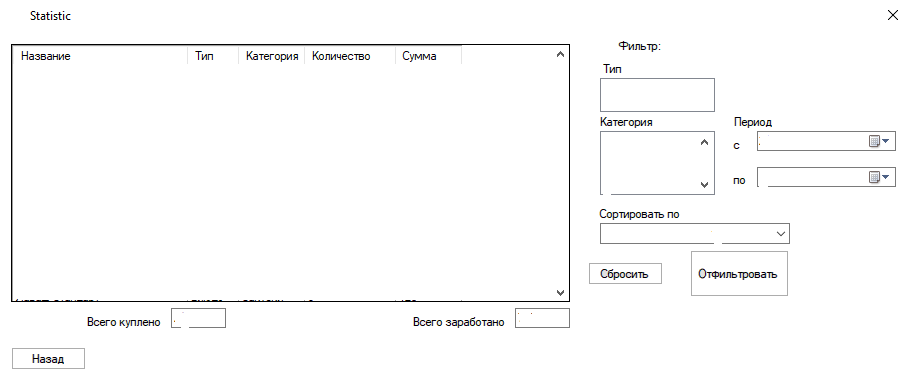


Рисунок 2.9 – Эскиз интерфейса для просмотра статистики

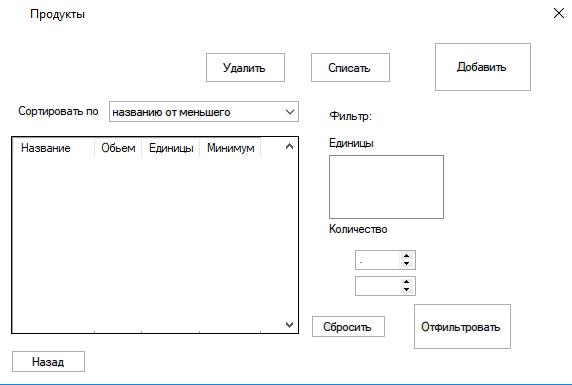


Рисунок 2.10 – Эскиз интерфейса списка продуктов