

СОДЕРЖАНИЕ

1 ВВЕДЕНИЕ.....	4
СПИСОК ИСПОЛЬЗОВАННЫХ СОКРАЩЕНИЙ	5
1 АНАЛИТИЧЕСКАЯ ЧАСТЬ	6
1.1. АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ МИНИ-ОТЕЛЯ	6
1.2. ПОСТАНОВКА ЗАДАЧИ.....	6
2 РАЗРАБОТКА ЛОГИЧЕСКОЙ МОДЕЛИ БАЗЫ ДАННЫХ	7
2.1. ПОСТРОЕНИЕ ДИАГРАММЫ «СУЩНОСТЬ-СВЯЗЬ» В НОТАЦИИ П.ЧЕНА	7
2.2. ПОСТРОЕНИЕ МОДЕЛИ, ОСНОВАННОЙ НА КЛЮЧАХ	9
2.3. ПОСТРОЕНИЕ ПОЛНОЙ АТТРИБУТИВНОЙ МОДЕЛИ	9
3 РАЗРАБОТКА ФИЗИЧЕСКОЙ МОДЕЛИ БАЗЫ ДАННЫХ	11
3.1. ВЫБОР АППАРАТНОЙ И ПРОГРАММНОЙ ПЛАТФОРМЫ ДЛЯ РЕАЛИЗАЦИИ БД	11
3.2. РЕАЛИЗАЦИЯ БАЗЫ ДАННЫХ.....	11
3.3. ТЕСТИРОВАНИЕ БАЗЫ ДАННЫХ	12
3.4. РАЗГРАНИЧЕНИЕ ПРАВ ДОСТУПА.....	13
3.5. РАСЧЕТ ИНФОРМАЦИОННЫХ ПАРАМЕТРОВ БД.....	13
4 РАЗРАБОТКА КЛИЕНТСКОГО ПРИЛОЖЕНИЯ.....	15
4.1. ОБОСНОВАНИЕ ВЫБОРА ЯЗЫКА ПРОГРАММИРОВАНИЯ	15
4.2. РАЗРАБОТКА ИНТЕРФЕЙСА ПОЛЬЗОВАТЕЛЯ.....	15
4.3. АЛГОРИТМ РАБОТЫ КАЖДОГО ИЗ МОДУЛЕЙ	17
4.4. ТЕСТИРОВАНИЕ РАБОТЫ ПРИЛОЖЕНИЯ	18
ЗАКЛЮЧЕНИЕ	20
СПИСОК ЛИТЕРАТУРЫ И ИНФОРМАЦИОННЫХ РЕСУРСОВ	21
ПРИЛОЖЕНИЕ А. “SQL СКРИПТ СОЗДАНИЯ БАЗЫ”.....	22
ПРИЛОЖЕНИЕ Б “PHP СТРАНИЦЫ ПРИЛОЖЕНИЯ”	24

1 ВВЕДЕНИЕ

В современном мире вся информация, хранимая и использующаяся в процессе работы, должна быть жестко структурирована и упорядочена. Это необходимо для того, чтобы упростить поиск, работу и сохранение данных.

В ходе выполнения курсового проекта разрабатывается физическая модель базы данных для учета и хранения постояльцев мини-отеля.

Целями курсового проектирования являются закрепление теоретических знаний, полученных в результате изучения курса «Управление данными», формирование практических навыков в области информационного моделирования и проектирования баз данных, разработка физической модели базы данных.

Для достижения цели на разных этапах курсового проектирования должны быть решены следующие задачи:

- выбор варианта задания, детализация постановки задачи и анализ предметной области;
- разработка логической модели базы данных;
- разработка и тестирование физической модели БД;
- разработка клиентского приложения.

Данная пояснительная записка состоит из: содержания, введения, а также других разделов. Раздел «Аналитическая часть», в котором описывается предметная область. Раздел «Разработка логической модели базы данных», в котором отражены логические связи между сущностями. Раздел «Разработка физической модели базы данных» описывает физические свойства базы данных (типы данных, размер полей, индексы). Раздел «Разработка клиентского приложения» в котором находятся сведения о выбранной платформе реализации, описание взаимодействия модулей, а также описание работы пользователя с ней.

СПИСОК ИСПОЛЬЗОВАННЫХ СОКРАЩЕНИЙ

БД – база данных

ERD - Entity-Relationship Diagrams (диаграмма «сущность-связь»)

1 АНАЛИТИЧЕСКАЯ ЧАСТЬ

1.1. Анализ предметной области мини-отеля

В базе данных для выбранной предметной области «Мини-отель» необходимо как можно более точно производить учет занятых и свободных номеров, находящихся постояльцев в отеле, а также дополнительных услуг для каждого клиента.

Типичный пример работы отеля: постоялец заселяется в номер, заказывает дополнительные услуги, выселяется из номера.

1.2. Постановка задачи

Основные объекты предметной области:

- номер;
- постоялец;
- услуга.

Связи между объектами:

- постоялец заселяется в один из заранее созданных номеров на некоторое время;
- услуга предоставляется клиенту, заселившемуся на некоторое время.

Пользователем данной базы данных является управляющей мини-отелем. Он может добавлять, удалять, изменять информацию о номерах и услугах, предоставляемых постояльцам, добавлять, изменять информацию о постояльцах, изменять даты заезда и менять информацию о том, заселены ли они в отель.

2 РАЗРАБОТКА ЛОГИЧЕСКОЙ МОДЕЛИ БАЗЫ ДАННЫХ

2.1. Построение диаграммы «сущность-связь» в нотации П.Чена

Разработка «сущность-связь» основывается на переходе от сложной сетевой структуры к древовидной и построения диаграммы.

Для начала была сформирована сложная сетевая структура, основанная на основных объектах предметной области:



Рисунок 1 – Сложная сетевая структура

На основе сложной сетевой структуры была синтезирована простая сетевая структура путем добавления промежуточных таблиц для того, чтобы связь многие ко многим трансформировать в связь один ко многим:

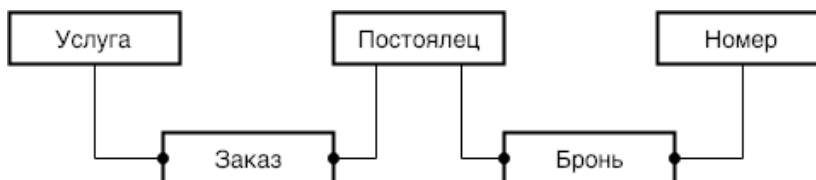


Рисунок 2 – Простая сетевая структура

На основе простой сетевой структуры была построена древовидная структура, отображающая взаимосвязь ближайших объектов базы, а также показывающая будущие внешние ключи:

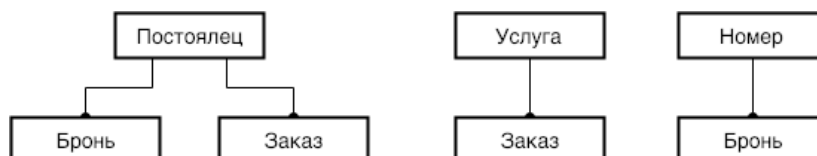


Рисунок 3 – Древовидная структура

Далее на основе прошлых таблиц была синтезирована ER диаграмма, отображающая взаимосвязь объектов, а также основные элементы взаимодействующих объектов:

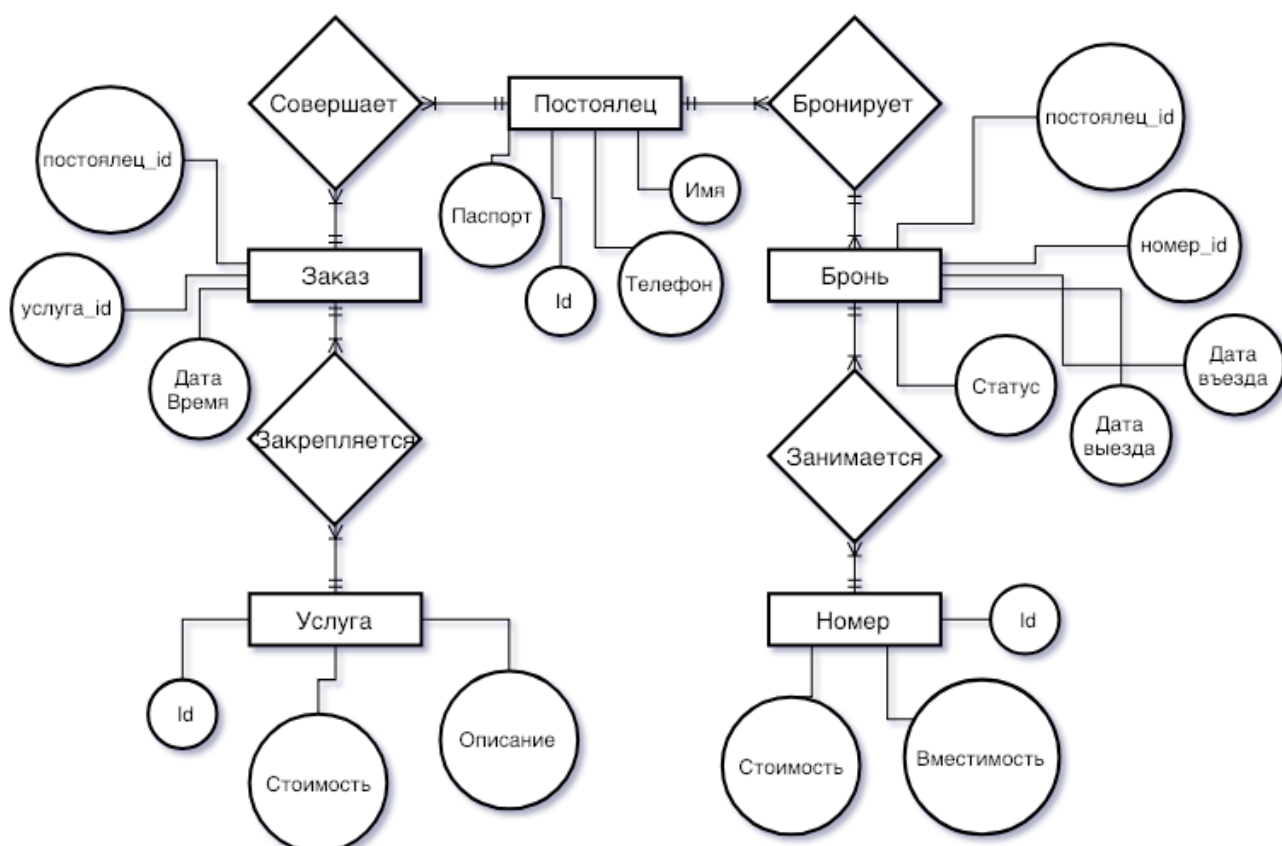


Рисунок 4 – ER диаграмма в нотации П.Чена

Объект «Постоялец» (*id* – уникальный номер, паспорт – информация о паспортных данных гостя, имя – ФИО постояльца, телефон – мобильный номер телефона постояльца) связывается с объектом «Номер» (*id* – номер в отеле, стоимость данного номера и его вместимость) через объект «Бронь» (уникальный номер постояльца, номер в отеле, дата въезда и выезда, статус брони(постоялец в отеле или нет)), объект «Услуга» (уникальный номер, стоимость услуги, описание услуги) соединяется с постояльцем через объект «Заказ» (уникальный номер постояльца, уникальный номер услуги, дата предоставления).

2.2. Построение модели, основанной на ключах

Данная модель описывает основные структуры данных и сущности

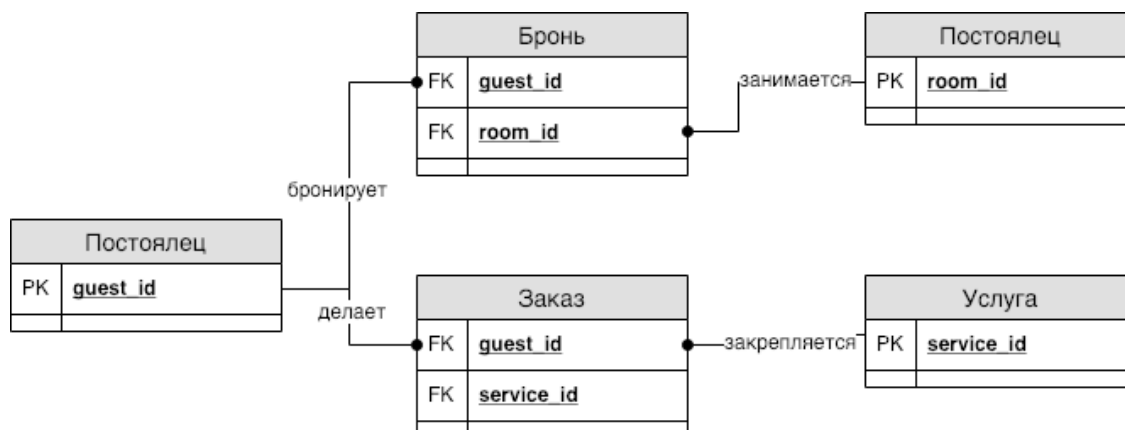


Рисунок 5 – Модель основанная на ключах

Модель, основанная на ключах, строится на основе ERD, но отображает больше деталей: ключевые поля, внешние ключи.

2.3. Построение полной атрибутивной модели

Нормализация – это разбиение таблицы на две или более, обладающих лучшими свойствами при включении, изменении и удалении данных. Окончательная цель нормализации сводится к получению такого проекта базы данных, в котором каждый факт появляется лишь в одном месте, т.е. исключена избыточность информации. Это делается не столько с целью экономии памяти, сколько для исключения возможной противоречивости хранимых данных. За основу возьмем выделенные ранее сущности с соответствующими им атрибутами.

Рассмотрим все выделенные нами сущности, и проверим их на атомарность. В каждой из сущности соблюдается атомарность, следовательно, делаем вывод о том, что наша база данных находится в первой нормальной форме.

Так как наша база данных, находится в первой нормальной форме, рассмотрим её на удовлетворение второй нормальной форме (Отношение R находится во второй нормальной форме в том и только в том случае, когда находится в 1НФ, и каждый не ключевой атрибут полностью зависит от первичного ключа). Каждая сущность содержит уникальный, искусственно введенный, первичный ключ, не обладающий смысловой нагрузкой, от которого полностью зависят все не ключевые атрибуты, это удовлетворяет условиям второй нормальной форме, следовательно, делаем вывод о том, что база данных находится во второй нормальной форме

Так как наша база данных находится во второй нормальной форме рассмотрим на её удовлетворение третьей нормальной форме (Отношение R находится в третьей нормальной форме в том и только в том случае, если находится в 2НФ и каждый не ключевой атрибут не транзитивно зависит от первичного ключа). Во всех созданных сущностях транзитивные зависимости отсутствуют, что говорит о том, что наша база данных находится в третьей нормальной форме.

В ходе нормализации была построена полная атрибутивная модель в нотации IDEF1X, которая представлена на рисунке 6.

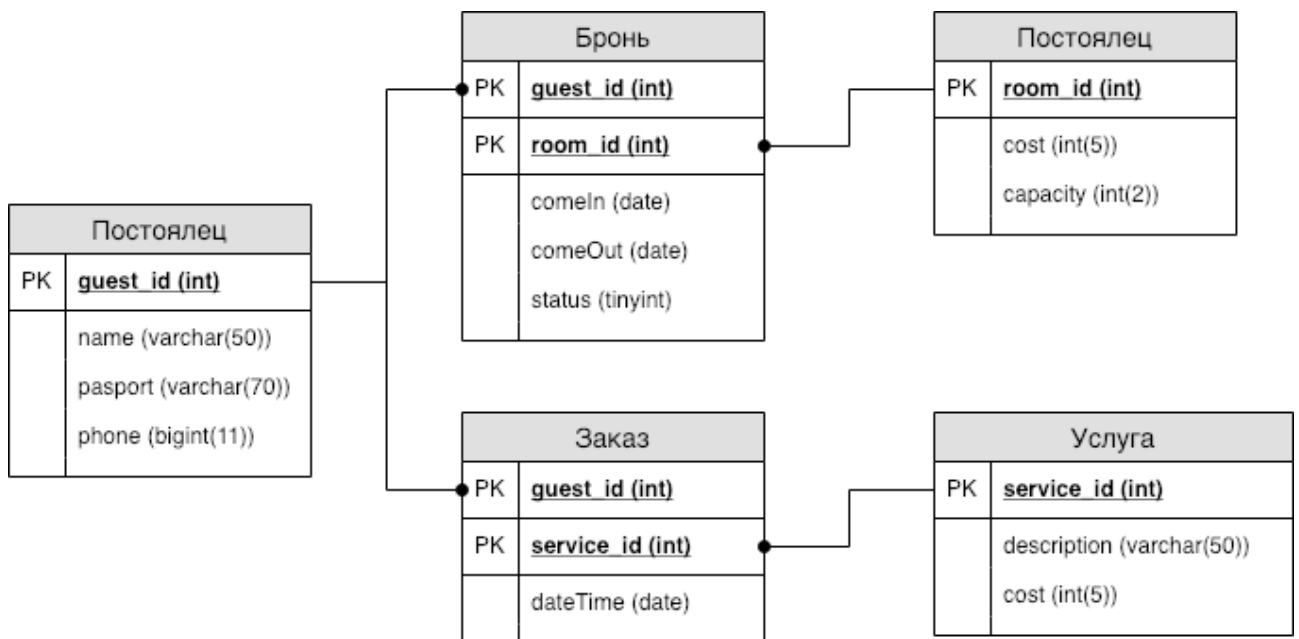


Рисунок 6 – полная атрибутивная модель

3 РАЗРАБОТКА ФИЗИЧЕСКОЙ МОДЕЛИ БАЗЫ ДАННЫХ

3.1. Выбор аппаратной и программной платформы для реализации БД

В качестве СУБД служит MySQL, которая была выбрана из-за того, что к этой базе относительно просто подключаться, есть встроенный графический интерфейс, который позволяет ускорить работу с вводом и выводом данных, бесплатная лицензия для некоммерческих проектов, для нее существует утилита (MySQL Workbench), позволяющая ускорить разработку физической модели БД и упростить создание базы.

Альтернативой MySQL могут служить: MariaDB, PostgreSQL.

3.2. Реализация базы данных

Полный листинг SQL скрипта представлен в разделе «Приложение А»

Физическая модель была разработана в приложении MySQL Workbench.

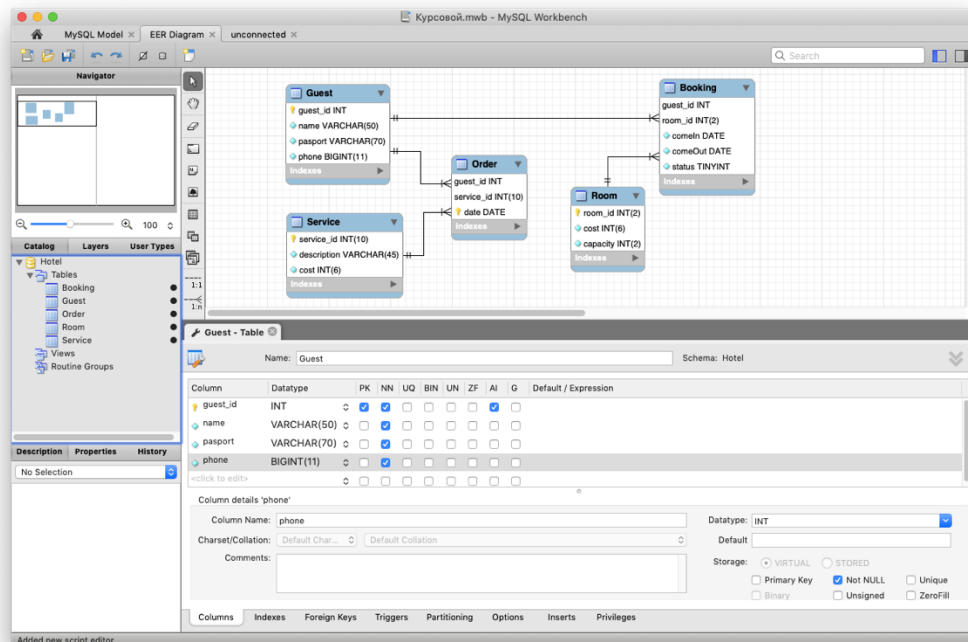


Рисунок 7 – скриншот представления разработанной базы данных

3.3. Тестирование базы данных

После создания таблиц и заполнения основными данными были произведены изменения в основных таблицах, удаление некоторых полей и наблюдение за тем, изменились ли данные в дочерних таблицах.

Showing rows 0 - 1 (2 total, Query took 0.0005 seconds.)

`SELECT * FROM `Guest``

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

+ Options

	guest_id	name	pasport	phone
<input type="checkbox"/> Edit Copy Delete	3	Анна	935r309453549	9787257183
<input type="checkbox"/> Edit Copy Delete	4	Дима	935r309453549	9787257183

Showing rows 0 - 1 (2 total, Query took 0.0004 seconds.)

`SELECT * FROM `Booking``

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

+ Options

	guest_id	room_id	comeIn	comeOut	status
<input type="checkbox"/> Edit Copy Delete	3	1	2018-12-23	2018-12-31	1
<input type="checkbox"/> Edit Copy Delete	4	2	2018-12-20	2018-12-31	0

Рисунок 8 – таблицы Guest и Booking до редактирования

Showing rows 0 - 1 (2 total, Query took 0.0004 seconds.)

`SELECT * FROM `Guest``

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

+ Options

	guest_id	name	pasport	phone
<input type="checkbox"/> Edit Copy Delete	1	Дима	935r309453549	9787257183
<input type="checkbox"/> Edit Copy Delete	6	Анна	935r309453549	9787257183

Showing rows 0 - 1 (2 total, Query took 0.0004 seconds.)

`SELECT * FROM `Booking``

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

+ Options

	guest_id	room_id	comeIn	comeOut	status
<input type="checkbox"/> Edit Copy Delete	1	2	2018-12-20	2018-12-31	0
<input type="checkbox"/> Edit Copy Delete	6	1	2018-12-23	2018-12-31	1

Рисунок 9 – Таблицы Guest и Booking после редактирования

Showing rows 0 - 0 (1 total, Query took 0.0005 seconds.)

```
SELECT * FROM `Order`
```

Number of rows: 25 Filter rows: Search this table

guest_id	service_id	date
6	4	2018-12-25

Showing rows 0 - 0 (1 total, Query took 0.0004 seconds.)

```
SELECT * FROM `Service`
```

Number of rows: 25 Filter rows: Search this table

service_id	description	cost
4	Трансфер	550

Рисунок 10 – Таблицы Order и Service до удаления из таблицы Service

MySQL returned an empty result set (i.e. zero rows). (Query took 0.0007 seconds.)

```
SELECT * FROM `Order`
```

Number of rows: 25 Filter rows: Search this table

guest_id	service_id	date
----------	------------	------

Рисунок 11 – таблица Order после удаления поля из таблицы Service

3.4. Разграничение прав доступа

Данная БД создана для администратора гостиницы, следовательно в ее права входит: создание, удаление, редактирование полей всех таблиц и запрет на удаление таблиц и базы данных.

3.5. Расчет информационных параметров БД

Для переменных длин полей возьмем максимальную длину поля, следовательно, размер типов данных для хранения в памяти будет составлять:

1. integer – 2 байта
2. Varchar(x) – x байт
3. Date – 3 байта

Были рассчитаны длины логических записей:

1. Таблица «Guest»: $L1 = 2 + 50 + 70 + 3 = 152$ байта

2. Таблица «Booking» $L2 = 2 + 2 + 2 = 6$ байт
3. Таблица «Service»: $L3 = 2 + 45 + 2 = 49$ байт
4. Таблица «Order»: $L4 = 2 + 2 + 3 = 5$ байт
5. Таблица «Room»: $L5 = 2 + 2 + 2 = 6$ байт

Предположим, что в таблице «Room» 10 записей ($L5 = 6$), в таблице «Service» 10 записей ($L3 = 49$), в таблице «Guest» 10 постояльцев ($L1 = 152$), в таблице «Booking» 10 записей ($L2 = 6$), в таблице «Order» 20 записей ($L4 = 5$).

Рассчитаем объем памяти, необходимой для размещения информационного фонда

$$I = 10 \cdot 152 + 6 \cdot 10 + 10 \cdot 49 + 5 \cdot 20 + 6 \cdot 10 = 2230 \text{ (байт)}$$

$$\Delta I = 152 \cdot 0,5 + 6 \cdot 0,2 + 49 \cdot 0,5 + 5 \cdot 0,1 + 6 \cdot 0,2 = 103,4 \text{ (1/байт)}$$

Время резервного копирования

$$T_{рк} = 0,2 \cdot 2230 / 103,4 = 4,31 \text{ (дня)}.$$

4 РАЗРАБОТКА КЛИЕНТСКОГО ПРИЛОЖЕНИЯ

4.1. Обоснование выбора языка программирования

В качестве языка программирования был выбран PHP: исполняемый на стороне сервера скрипт. Его плюсами является простота внедрения в html страницы, простота кода, легкость подключения к БД и оперирования данными с помощью SQL запросов.

4.2. Разработка интерфейса пользователя

Интерфейс пользователя был разработан на HTML, полный листинг представлен в разделе «Приложение Б».

На странице «Гости» (рисунок 12) отображается список гостей находящихся в отеле, выселившихся за прошлый месяц и список всех гостей. Так же сверху панель с добавлением информации о новом постояльце: Имя, паспортные данные, телефон, дата въезда, выезда и номер заселения.

Hotel

[Гости](#)
[Номера](#)
[Услуги](#)

Добавить

Имя

Паспорт

Телефон

Заязд

Выезд

2 1500P 1

Добавить

В отеле

№	Имя	С	До	Услуги	Телефон
1	Анна	2018-12-23	2018-12-31		9787257183

Выселившиеся

№	Имя	Выселился	Телефон
2	Дима	2018-12-31	9787257183

Все гости

Имя	Паспорт	Телефон
Дима	935г309453549	9787257183
Анна	935г309453549	9787257183

Рисунок 12 – Страница «Гости»

Если нажать на имя, то откроется страница с редактированием данных гостя.

Рисунок 13 – Страница редактирования информации о постояльце

На странице «Номера» отображаются свободные и занятые номера (с именем постояльца)

№	Постоялец	Стоимость в сутки	Вместимость
1	Анна	1200 Р	2

№	Стоимость в сутки	Вместимость
2	1500 Р	1

Рисунок 14 – Страница «Номера»

На странице «Услуги» отображается список всех услуг, а также возможность применить их к одному из постояльцев. Если нажать на название услуги, то откроется страница с редактированием по аналогии с гостем.

Hotel
[Гости](#)
[Номера](#)
[Услуги](#)

Добавить услугу

Добавить

Применить услугу

1 - Анна

Дополнительная уборка - 300Р

Применить

Список услуг

Описание	Стоимость
Дополнительная уборка	300
Трансфер	550

Рисунок 15 – Страница «Услуги»

4.3. Алгоритм работы каждого из модулей

Страница «Гости»

Во время загрузки страницы выполняется запрос к базе данных и в форму добавления клиента загружаются свободные номера. После в таблицы «В отеле», «Выселившиеся» загружаются данные из таблиц “Booking” и “Guests”.

После добавления информации о новом пользователе и нажатия на кнопку «Добавить», производится запрос на сервер и в базу добавляется информация о новом постояльце.

Страница «Номера»

На странице номер при загрузке страницы со стороны сервера выполняются запрос к базе, выводятся 2 таблицы: занятые и свободные номера. Если в нижней форме ввести данные о номере и нажать «Добавить», выполняется запрос на сервер и в таблицу “Rooms” добавляется запись о новом номере.

Страница «Услуги»

После выбора в форме «Применить услугу» значений номера и вида услуги, введения даты и нажатия на кнопку «применить», выполняется запрос на

сервер и в таблицу “Order” добавляется новая запись. По аналогии с гостями выполняется и редактирование услуг.

4.4. Тестирование работы приложения

Тестирование проводилось путем добавления, удаления, редактирования данных в формах и таблицах.

Hotel
[Гости](#)
[Номера](#)
[Услуги](#)

Добавить

Имя

Паспорт

Телефон

dd / mm / yyyy

dd / mm / yyyy

3 2000Р 1

Добавить

В отеле

№	Имя	С	До	Услуги	Телефон
1	Анна	2018-12-24	2019-01-01	Дополнительная уборка - 2018-12-25 - 300Р	9787257183
2	Дима	2018-12-26	2018-12-28	Трансфер - 2018-12-27 - 550Р	9787257183

Выселившиеся

№	Имя	Выселился	Телефон
4	Артур	2018-12-24	9787257182

Все гости

Имя	Паспорт	Телефон
Анна	345	9787257183
Дима	ывавыва	9787257183
Артур	ывавывавыва	9787257182

Рисунок 16 – Страница «Гости» после добавления записей

Hotel
[Гости](#)
[Номера](#)
[Услуги](#)

Изменить

Анна

345

9787257183

24 / 12 / 2018

01 / 01 / 2019

1

Изменить

Выселить

Удалить

Рисунок 17 – Страница редактирования данных по госте

Hotel

[Гости](#)[Номера](#)[Услуги](#)

Занятые

№	Постоялец	Стоимость в сутки	Вместимость
1	Анна	1200 ₽	2
2	Дима	1500 ₽	1

Свободные

№	Стоимость в сутки	Вместимость
3	2000 ₽	1
4	2000 ₽	2

Добавить номер

Добавить

Рисунок 18 – Страница информации о номерах после добавления данных

В результате тестирования приложения не было выявлено дефектов, вызывающих нарушение структуры базы данных, приложение выполняет необходимые действия по добавлению, изменению, удалению данных в таблицах.

ЗАКЛЮЧЕНИЕ

В данном курсовом проекте была спроектирована реляционная база данных на тему «База данных мини-отеля».

Для выбранной предметной области был проведен анализ, построена логическая модель и проведена нормализация, которая позволяет решить проблемы рационального выбора вариантов схем отношений из возможного множества альтернативных решений. В полученной в результате нормализации логической схеме содержатся следующие таблицы: гости, номера, услуги, заказы, бронирование.

Так же разработано приложение для работы с базой на РНР. В нем можно выполнять введение, изменение, удаление и отображение данных.

Были закреплены навыки организации баз данных, разработки программного обеспечения для него и обеспечения целостности.

СПИСОК ЛИТЕРАТУРЫ И ИНФОРМАЦИОННЫХ РЕСУРСОВ

1. Шарипова Н.Н. Теория и практика проектирования баз данных: учебное пособие для студентов высших учебных заведений, обучающихся по группе математических и механических направлений и специальностей / Н.Н. Шарипова; науч. ред. А.П. Замятин; Федер. агентство по образованию, Урал. гос ун-т им. А.М. Горького – Екатеринбург: Изд-во Урал. ун-та, 2006. – 394 с.
2. Дейт, К. Дж. Введение в системы баз данных; К.: Диалектика; Издание 6-е – Москва, 2010. – 784 с.
3. Кузнецов С.Д. Основы баз данных; Бином. Лаборатория знаний, Интернет-университет информационных технологий – Москва, 2010. – 488 с.
4. Голицына О.Л., Партыка Т.Л., Попов И.И. Основы проектирования баз данных; Форум – Москва, 2012. – 416 с.
5. Кузнецов, С.Д. Основы баз данных; Бином – Москва, 2013. – 484 с.

ПРИЛОЖЕНИЕ А. “SQL СКРИПТ СОЗДАНИЯ БАЗЫ”

```

DROP SCHEMA IF EXISTS `Hotel` ;

-----
-- Schema Hotel
-----
CREATE SCHEMA IF NOT EXISTS `Hotel` DEFAULT CHARACTER SET utf8 ;
USE `Hotel` ;

-----
-- Table `Hotel`.`Service`
-----
DROP TABLE IF EXISTS `Hotel`.`Service` ;

CREATE TABLE IF NOT EXISTS `Hotel`.`Service` (
  `service_id` INT(10) NOT NULL AUTO_INCREMENT,
  `description` VARCHAR(45) NOT NULL,
  `cost` INT(6) NOT NULL,
  PRIMARY KEY (`service_id`))
ENGINE = InnoDB;

-----
-- Table `Hotel`.`Guest`
-----
DROP TABLE IF EXISTS `Hotel`.`Guest` ;

CREATE TABLE IF NOT EXISTS `Hotel`.`Guest` (
  `guest_id` INT NOT NULL AUTO_INCREMENT,
  `name` VARCHAR(50) NOT NULL,
  `passport` VARCHAR(70) NOT NULL,
  `phone` INT NOT NULL,
  PRIMARY KEY (`guest_id`))
ENGINE = InnoDB;

-----
-- Table `Hotel`.`Room`
-----
DROP TABLE IF EXISTS `Hotel`.`Room` ;

CREATE TABLE IF NOT EXISTS `Hotel`.`Room` (
  `room_id` INT(2) NOT NULL,
  `cost` INT(6) NOT NULL,
  `capacity` INT(2) NOT NULL,
  PRIMARY KEY (`room_id`))
ENGINE = InnoDB;

-----
-- Table `Hotel`.`Order`
-----
DROP TABLE IF EXISTS `Hotel`.`Order` ;

CREATE TABLE IF NOT EXISTS `Hotel`.`Order` (
  `guest_id` INT NOT NULL,
  `service_id` INT(10) NOT NULL,
  `date` DATE NOT NULL,
  PRIMARY KEY (`guest_id`, `service_id`, `date`),
  INDEX `service_id_idx` (`service_id` ASC),
  CONSTRAINT `guest-service`
    FOREIGN KEY (`guest_id`)
      REFERENCES `Hotel`.`Guest` (`guest_id`)
    ON DELETE CASCADE
    ON UPDATE CASCADE,
  CONSTRAINT `service-guest`
    FOREIGN KEY (`service_id`)

```

```

REFERENCES `Hotel`.`Service` (`service_id`)
ON DELETE CASCADE
ON UPDATE CASCADE)
ENGINE = InnoDB;

```

```

-----
-- Table `Hotel`.`Booking`
-----
DROP TABLE IF EXISTS `Hotel`.`Booking` ;

CREATE TABLE IF NOT EXISTS `Hotel`.`Booking` (
  `guest_id` INT NOT NULL,
  `room_id` INT(2) NOT NULL,
  `comeIn` DATETIME NOT NULL,
  `comeOut` DATETIME NOT NULL,
  `status` TINYINT NULL,
  INDEX `room_id_idx` (`room_id` ASC),
  PRIMARY KEY (`guest_id`, `room_id`),
  CONSTRAINT `guest-room`
    FOREIGN KEY (`guest_id`)
      REFERENCES `Hotel`.`Guest` (`guest_id`)
      ON DELETE CASCADE
      ON UPDATE CASCADE,
  CONSTRAINT `room-guest`
    FOREIGN KEY (`room_id`)
      REFERENCES `Hotel`.`Room` (`room_id`)
      ON DELETE CASCADE
      ON UPDATE CASCADE)
ENGINE = InnoDB;

```

ПРИЛОЖЕНИЕ Б “РНР СТРАНИЦЫ ПРИЛОЖЕНИЯ”

Admin-Guests.php

```

<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <title>Гости</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet" type="text/css" media="screen" href="main.css" />
    <script src="main.js"></script>
</head>

<?php
    $host = "localhost"; $user = "root"; $password = "root"; $database = "Hotel"; $port
= "8889";
    $socket = "/Applications/MAMP/tmp/mysql/mysql.sock";
    $connection = mysqli_connect($host, $user, $password, $database, $port, $socket);
?>

<body>
    <div class="top-panel">
        <a href="admin.html" class="logo">Hotel</a>
        <div>
            <a href="" id="" class="link-selected">Гости</a>
            <a href="Admin-Rooms.php" >Номера</a>
            <a href="Admin-Services.php" id="">Услуги</a>
        </div>
    </div>

    <div class="pageElement">
        <h1>Добавить</h1>
        <?php
            $result = mysqli_query($connection, "SELECT Room.room_id, Room.cost,
Room.capacity FROM `Room` WHERE Room.room_id NOT IN (SELECT Booking.room_id FROM Booking
WHERE Booking.status=1)");
            ?>
            <form action="" method="POST" target="_self">
                <table>
                    <tr>
                        <td><input type="text" name="guestName" placeholder="Имя"></td>
                        <td><input
placeholder="Паспорт"></td>
                        <td colspan="2"><input type="text" name="guestPhone"
placeholder="Телефон"></td>
                    </tr>
                    <tr>
                        <td><input type="date" name="comeIn" placeholder="Заезд"></td>
                        <td><input type="date" name="comeOut" placeholder="Выезд"></td>
                        <td>
                            <select name="roomSelect" id="roomSelect">
                                <option disabled>Выберите номер</option>
                                <?php
                                    while ($record = mysqli_fetch_assoc($result)) {
                                        echo
value=".$record["room_id"].">".$record["room_id"]."
". $record["cost"]."Р
". $record["capacity"]."□". "</option>";
                                    }
                                ?>
                            </select>
                        </td>
                    </tr>
                    <tr><td colspan="3"><input type="submit" name="addUser"
value="Добавить"></td></tr>
                </table>
            </form>
        </div>
    </div>

```

```

</form>
<?php
    if (isset($_POST['addUser'])) {
        $sql = mysqli_query($connection, "INSERT INTO `Guest` (`guest_id`,
`name`, `passport`, `phone`)
VALUES
                                (NULL,
                                '$_POST['guestName'].',
'$_POST['guestPassport'].', '$_POST['guestPhone'].')");

        //выбор максимального значени id в result
        $result = mysqli_query($connection, "SELECT max(guest_id) as id from
Guest");

        $record = mysqli_fetch_assoc($result);
        $result = $record['id'];

        $sql = mysqli_query($connection, "INSERT INTO `Booking` (`guest_id`,
`room_id`, `comeIn`, `comeOut`, `status`)
VALUES
                                ('$_result.',
                                '$_POST['roomSelect'].',
'$_POST['comeIn'].', '$_POST['comeOut'].', TRUE)");
    }

?>
</div>

<div class="pageElement outTable">
    <h1>В отеле</h1>
    <?php $result = mysqli_query($connection, "SELECT Booking.room_id,
Guest.name, Guest.phone,Booking.comeIn, Booking.comeOut,Guest.guest_id FROM Guest JOIN
Booking ON Booking.guest_id=Guest.guest_id AND Booking.status=1"); ?>
    <table >
        <tr>
            <th>№</th>
            <th>Имя</th>
            <th>С</th>
            <th>До</th>
            <th>Услуги</th>
            <th>Телефон</th>
        </tr>
        <?php while ($record = mysqli_fetch_assoc($result)) {
            $services = mysqli_query($connection, "SELECT Service.description,
Service.cost, Order.date FROM Service
JOIN `Order` ON Service.service_id = `Order`.service_id
INNER JOIN Guest ON `Order`.`guest_id` = Guest.guest_id AND
Guest.guest_id='$_record['guest_id'].
INNER JOIN Booking ON Booking.guest_id = Guest.guest_id AND
Booking.status=1 AND Order.date BETWEEN Booking.comeIn AND Booking.comeOut");
            $services_rows = "";
            while($record_services=mysqli_fetch_assoc($services)){
                $services_rows .= $record_services["description"]." -
'$_record_services['date']. - '$_record_services['cost'].<br>";
            }
            echo '<tr>'.
                '<td>' .$_record["room_id"].'</td>'.
                '<td>'.
                    <form action="Admin-Guests-Edit.php" method="POST"
target="_self">
                        <input type="text" style="display:none"
name="guestId" value="' .$_record["guest_id"].'">
                        <input type="submit" value="' .$_record["name"].'">
                    </form>
                </td>'.
                '<td>' .$_record["comeIn"].'</td>'.
                '<td>' .$_record["comeOut"].'</td>'.
                '<td>' .$_services_rows.'</td>'.
                '<td>' .$_record["phone"].'</td>'
                .</tr>';
        }?>
    </table>
</div>

```

```

<div class="pageElement outTable">
    <h1>Выселившиеся</h1>
    <?php $result = mysqli_query($connection, "SELECT Booking.room_id,
Guest.name,Guest.phone,Booking.comeOut,Guest.guest_id FROM Guest JOIN Booking ON
Booking.guest_id=Guest.guest_id AND Booking.status = 0"); ?>
    <table>
        <tr>
            <th>№</th>
            <th>Имя</th>
            <th>Выселился</th>
            <th>Телефон</th>
        </tr>
        <?php while ($record = mysqli_fetch_assoc($result)) {
            echo '<tr>'.
                '<td>'. $record["room_id"]. '</td>'.
                '<td><form action="Admin-Guests-Edit.php" method="POST"
target="_self"><input type="text" style="display:none" name="guestId"
value="'. $record["guest_id"]. '"><input type="submit"
value="'. $record["name"]. '"></form></td>'.
                '<td>'. $record["comeOut"]. '</td>'.
                '<td>'. $record["phone"]. '</td>'
                . '</tr>';
        } ?>
    </table>
</div>

<div class="pageElement outTable">
    <h1>Все гости</h1>
    <?php $result = mysqli_query($connection, "SELECT * FROM Guest "); ?>
    <table>
        <tr>
            <th>Имя</th>
            <th>Паспорт</th>
            <th>Телефон</th>
        </tr>
        <?php while ($record = mysqli_fetch_assoc($result)) {
            echo '<tr>'.
                '<td>'. $record["name"]. '</td>'.
                '<td>'. $record["passport"]. '</td>'.
                '<td>'. $record["phone"]. '</td>'
                . '</tr>';
        } ?>
    </table>
</div>
</body>

<footer>
    <?php
        if ($connection==false) {
            echo '<p style="color: darkred">Ошибка подключения к БД:
'.mysqli_connect_error(). '</p>';
            exit();
        } else echo '<p style="color: darkcyan">Успешное подключение к БД</
p>';
    ?>
</footer>

<?php mysqli_close($connection); ?>
</html>

```

Admin-Guests-Edit.php

```

<!DOCTYPE html>
<html>

```



```

<head>
  <meta charset="utf-8" />
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <title>Подробнее</title>
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link rel="stylesheet" type="text/css" media="screen" href="main.css" />
  <script src="main.js"></script>
</head>

<?php
  $host = "localhost"; $user = "root"; $password = "root"; $database = "Hotel"; $port
= "8889";
  $socket = "/Applications/MAMP/tmp/mysql/mysql.sock";
  $connection = mysqli_connect($host, $user, $password, $database, $port, $socket);
  $guestId = $_POST['guestId'];
  ?>

<body>
  <div class="top-panel">
    <a href="admin.html" class="logo">Hotel</a>
    <div>
      <a href="Admin-Guests.php" id="">Гости</a>
      <a href="Admin-Rooms.php" >Номера</a>
      <a href="Admin-Services.php" id="">Услуги</a>
    </div>
  </div>

  <div class="pageElement">
    <h1>Изменить</h1>
    <?php
      $result = mysqli_query($connection, "SELECT * FROM `Guest` JOIN Booking
ON Guest.guest_id = Booking.guest_id AND Guest.guest_id = ".$guestId." ");
      $row = mysqli_fetch_assoc($result);

      $temp;
      if($row['status']==1) $temp = '<td><input type="submit" name="checkOut"
value="Выселить"</td>';
      else $temp = '<td><input type="submit" name="checkIn"
value="Заселить"</td>';

      $rooms = mysqli_query($connection, "SELECT * FROM `Room` WHERE room_id
!= ".$row['room_id']."");

      $currentRoom = mysqli_fetch_assoc(
        mysqli_query($connection, "SELECT room_id FROM `Room` WHERE room_id
= ".$row['room_id']."")
      )["room_id"];
      $currentRoomString = "<option>".$currentRoom."</option>";

      $roomsOptions="";
      while ($room = mysqli_fetch_assoc($rooms)) {
        $roomsOptions .= "<option
value=".$room["room_id"].">".$room["room_id"]."
".$room["cost"]."P
".$room["capacity"]."</option>";
      }

      echo '<form action="" method="POST" target="_self">
<input type="text" value="'.$guestId.'" name="guestId"
style="display:none">
<table>
  <tr>
    <td><input type="text" name="guestName" placeholder="Имя"
value='.$row["name"].'></td>
    <td><input type="text" name="guestPasport"
placeholder="Паспорт" value='.$row["passport"].'></td>
    <td colspan="2"><input type="text" name="guestPhone"
placeholder="Телефон" value='.$row["phone"].'></td>
  </tr>
</table>

```

```

                                <td><input type="date" name="comeIn" placeholder="Заезд"
value='.$row["comeIn"].'></td>
                                <td><input type="date" name="comeOut" placeholder="Выезд"
value='.$row["comeOut"].'></td>
                                <td>
                                    <select name="roomSelect" id="roomSelect">
                                        <option disabled>Выберите номер</option>
                                        '.$currentRoomString.'
                                        '.$roomsOptions.'
                                    </select>
                                </td>
                            </tr>
                            <tr>
                                <td colspan="1">
                                    <input
value="Изменить"></td>' .$temp.'
                                <td><input
value="Удалить"></td>
                                <td colspan="1">
                                    <input
type="submit" name="editUser"
type="submit" name="deleteGuest"
                                </td>
                            </tr>
                        </table>
                    </form>' ;
                ?>
</div>
<?php
    if (isset($_POST['editUser'])) {
        echo $guestId;
        $sql = mysqli_query($connection, "UPDATE `Guest`
SET
    `name` = '". $_POST['guestName']."' ,
    `passport` = '". $_POST['guestPasport']."' ,
    `phone` = '". $_POST['guestPhone']."'
WHERE `Guest`.`guest_id` = '". $guestId."'");

        $sql = mysqli_query($connection, "UPDATE `Booking`
SET
    `room_id` = '". $_POST['roomSelect']."' ,
    `comeIn` = '". $_POST['comeIn']."' ,
    `comeOut` = '". $_POST['comeOut']."'
WHERE `Booking`.`guest_id` = '". $guestId."
AND `Booking`.`room_id` = '". $currentRoom['room_id']");

        header ('Location: Admin-Guests.php'); // перенаправление на страницу
        exit();
    }
    if(isset($_POST['checkIn'])){
        $sql = mysqli_query($connection, "UPDATE `Booking`
SET `status` = '1'
WHERE `Booking`.`guest_id` = '". $guestId."'");
        header ('Location: Admin-Guests.php'); // перенаправление на страницу
        exit();
    }
    if(isset($_POST['checkOut'])){
        $sql = mysqli_query($connection, "UPDATE `Booking`
SET `status` = '0'
WHERE `Booking`.`guest_id` = '". $guestId."'");
        header ('Location: Admin-Guests.php'); // перенаправление на страницу
        exit();
    }
    if(isset($_POST['deleteGuest'])){
        $sql = mysqli_query($connection, "DELETE FROM `Guest` WHERE
`Guest`.`guest_id` = '". $guestId.");
        header ('Location: Admin-Guests.php'); // перенаправление на страницу
        exit();
    }
    ?>
</body>

<footer>
<?php

```

```

        if ($connection==false) {
            echo '<p style="color: darkred">Ошибка подключения к БД:
'.mysqli_connect_error().'</p>';
            exit();
        } else echo '<p style="color: darkcyan">Успешное подключение к БД</
p>';

    ?>
</footer>

<?php mysqli_close($connection);?>
</html>

```

Admin-Rooms.php

```

<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <title>Гости</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet" type="text/css" media="screen" href="main.css" />
    <script src="main.js"></script>
</head>

<?php
    $host = "localhost";$user = "root";$password = "root";$database = "Hotel";$port
= "8889";
    $socket = "/Applications/MAMP/tmp/mysql/mysql.sock";
    $connection = mysqli_connect($host,$user,$password,$database,$port,$socket);
    ?>

<body>
    <div class="top-panel">
        <a href="admin.html" class="logo">Hotel</a>
        <div>
            <a href="Admin-Guests.php" id="">Гости</a>
            <a href="Admin-Rooms.php" class="link-selected">Номера</a>
            <a href="Admin-Services.php" id="">Услуги</a>
        </div>
    </div>

    <div class="pageElement outTable">
        <h1>Занятые</h1>
        <?php $result = mysqli_query($connection, "SELECT Guest.name,
Booking.room_id, Booking.comeIn, Booking.comeOut, Room.cost, Room.capacity
FROM Room JOIN Booking ON Booking.room_id = Room.room_id
INNER JOIN Guest ON Guest.guest_id=Booking.guest_id AND Booking.status=1
ORDER BY `Booking`.`room_id` ASC"); ?>
        <table>
            <tr>
                <th>№</th>
                <th>Постоялец</th>
                <th>Стоимость в сутки</th>
                <th>Вместимость</th>
            </tr>
            <?php while ($record = mysqli_fetch_assoc($result)) {
                echo '<tr>'.
                    '<td>'.$record["room_id"].'</td>'.
                    '<td>'.$record["name"].'</td>'.
                    '<td>'.$record["cost"].' P</td>'.
                    '<td>'.$record["capacity"].'</td>'.
                    '</tr>';
            }?>
        </table>
    </div>

```

```

<div class="pageElement outTable">
    <h1>Свободные</h1>
    <?php $result = mysqli_query($connection, "SELECT Room.room_id, Room.cost,
Room.capacity FROM `Room` WHERE Room.room_id NOT IN (SELECT Booking.room_id FROM Booking
WHERE Booking.status=1)"); ?>
    <table>
        <tr>
            <th>№</th>
            <th>Стоимость в сутки</th>
            <th>Вместимость</th>
        </tr>
        <?php while ($record = mysqli_fetch_assoc($result)) {
            echo '<tr>'.
                '<td>'. $record["room_id"]. '</td>'.
                '<td>'. $record["cost"]. ' P</td>'.
                '<td>'. $record["capacity"]. '</td>'.
                '</tr>';
        } ?>
    </table>
</div>

<div class="pageElement">
    <h1>Добавить номер</h1>
    <form action="" method="POST" target="_self">
        <table>
            <tr>
                <td><input type="text" name="roomId" placeholder="Номер"></td>
                <td><input type="text" name="roomCost"
placeholder="Стоимость"></td>
                <td><input type="text" name="roomCapacity"
placeholder="Вместимость" ></td>
            </tr>
            <tr><td colspan="3"><input type="submit" name="addRoom"
value="Добавить"></td></tr>
        </table>
    </form>
    <?php
        if (isset($_POST['addRoom'])) {
            $sql = mysqli_query($connection, "INSERT INTO `Room` (`room_id`,
`cost`, `capacity`)
VALUES
('".$_POST['roomId']. "','".$_POST['roomCost']. "','".$_POST['roomCapacity']. "')");
        }
    ?>
</div>

</body>

<footer>
    <?php
        if ($connection==false) {
            echo '<p style="color: darkred">Ошибка подключения к БД:
'.mysqli_connect_error(). '</p>';
            exit();
        } else echo '<p style="color: darkcyan">Успешное подключение к БД</
p>';
    ?>
</footer>

<?php mysqli_close($connection); ?>
</html>

```

Admin-Services.php

```

<!DOCTYPE html>
<html>

```

```

<head>
  <meta charset="utf-8" />
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <title>Гости</title>
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link rel="stylesheet" type="text/css" media="screen" href="main.css" />
  <script src="main.js"></script>
  <script src="plugins/jquery-3.3.1.js"></script>
</head>

<?php
  $host = "localhost"; $user = "root"; $password = "root"; $database = "Hotel"; $port
= "8889";
  $socket = "/Applications/MAMP/tmp/mysql/mysql.sock";
  $connection = mysqli_connect($host, $user, $password, $database, $port, $socket);
  ?>

<body>
  <div class="top-panel">
    <a href="admin.html" class="logo">Hotel</a>
    <div>
      <a href="Admin-Guests.php" id="">Гости</a>
      <a href="Admin-Rooms.php">Номера</a>
      <a href="" class="link-selected">Услуги</a>
    </div>
  </div>

  <div class="pageElement">
    <h1>Добавить услугу</h1>
    <form action="" method="POST" target="_self">
      <table>
        <tr>
          <td><input
placeholder="Описание"></td>
          <td><input
placeholder="Стоимость"></td>
        </tr>
        <tr><td colspan="3"><input
value="Добавить"></td></tr>
      </table>
    </form>
    <?php
      if (isset($_POST['addService'])) {
        $sql = mysqli_query($connection, "INSERT INTO `Service`
(`service_id`, `description`, `cost`)
VALUES
(NULL, '$_POST[serviceDescription].', '$_POST[serviceCost].')");
      }
    >
  </div>

  <div class="pageElement outTable">
    <form action="" method="POST" target="_self">
      <h1>Применить услугу</h1>
      <?php
        $services = mysqli_query($connection, "SELECT * FROM Service ORDER
BY description");
      >
      <div class="flexed">
        <select name="selectGuest" id="selectGuest">
          <option disabled>Выбор номера</option>
          <?php
            $rooms = mysqli_query($connection, "SELECT Booking.room_id,
Guest.name, Guest.guest_id FROM `Booking` JOIN Guest ON Booking.guest_id = Guest.guest_id
AND Booking.status = 1");

            while ($record1 = mysqli_fetch_assoc($rooms)){

```

```

                                echo                                ' <option
value="'. $record1["guest_id"].'">'. $record1["room_id"].' □
'. $record1["name"].' □ </option>';
                                }
                                ?>
                                </select>
                                <select name="selectService" id="selectService">
                                <?php
                                    while ($service = mysqli_fetch_assoc($services)) {
                                        echo
                                        ' <option
value="'. $service["service_id"].'">'. $service["description"].
                                        ' - '. $service["cost"].' P </option>';
                                    }
                                ?>
                                </select>
                                <input type="date" name="dateService" id="dateService">
                                <button                                type="submit"                                id="applyService"
name="applyService">Применить</button>
                                </div>
                                </form>
                                </div>

                                <div class="pageElement outTable">
                                <h1>Список услуг</h1>
                                <table>
                                <tr>
                                    <th>Описание</th>
                                    <th>Стоимость</th>
                                    <th></th>
                                </tr>
                                <?php
                                    $services = mysqli_query($connection, "SELECT * FROM Service
ORDER BY description");
                                    while ($record = mysqli_fetch_assoc($services)) {
                                        echo ' <tr>'.
                                        ' <td>
                                            <form                                action="Admin-Services-Edit.php"                                method="POST"
target="_self">
                                                <input                                type="text"                                style="display:none"
name="serviceId" value="'. $record["service_id"].'">
                                                <input                                type="submit"
value="'. $record["description"].'">
                                                </form>
                                            </td>'.
                                                // ' <td>'. $record["description"].' </td>'.
                                                ' <td>'. $record["cost"].' </td> </tr>';
                                    }
                                ?>
                                </table>
                                </div>

                                <?php
                                    if (isset($_POST['applyService'])) {
                                        $sql = mysqli_query($connection, "INSERT INTO `Order` (`guest_id`,
`service_id`, `date`)
                                        VALUES
                                        ('".$_POST['selectGuest']."', '".$_POST['selectService']."', '".$_POST['dateService']. "')");
                                    }
                                ?>
                                </body>

                                <footer>
                                <?php
                                    if ($connection==false) {
                                        echo ' <p style="color: darkred">Ошибка подключения к БД:
'. mysqli_connect_error().' </p>';
                                        exit();
                                    } else echo ' <p style="color: darkcyan">Успешное подключение к БД</

```

```

        p>';
    ?>
</footer>

<?php mysqli_close($connection);?>
</html>

```

Admin-Services-Edit.php

```

<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <title>Изменить услугу</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet" type="text/css" media="screen" href="main.css" />
    <script src="main.js"></script>
</head>

<?php
    $host = "localhost"; $user = "root"; $password = "root"; $database = "Hotel"; $port
= "8889";
    $socket = "/Applications/MAMP/tmp/mysql/mysql.sock";
    $connection = mysqli_connect($host, $user, $password, $database, $port, $socket);
    $serviceId = $_POST['serviceId'];
?>

<body>
    <div class="top-panel">
        <a href="admin.html" class="logo">Hotel</a>
        <div>
            <a href="Admin-Guests.php" id="">Гости</a>
            <a href="Admin-Rooms.php" >Номера</a>
            <a href="Admin-Services.php" id="">Услуги</a>
        </div>
    </div>

    <div class="pageElement">
        <h1>Изменить услугу</h1>
        <?php
            $result = mysqli_query($connection, "SELECT * FROM `Service`");
            $row = mysqli_fetch_assoc($result);

            echo '<form action="" method="POST" target="_self">
                <input type="text" value="'. $serviceId. '" name="serviceId"
style="display:none">
                <table>
                    <tr>
                        <td><input type="text" name="serviceDescription"
placeholder="Описание" value="'. $row["description"]. '></td>
                        <td><input type="text" name="serviceCost"
placeholder="Стоимость" value="'. $row["cost"]. '></td>
                    </tr>
                    <tr>
                        <td colspan="1">
                            <input type="submit" name="editService"
value="Изменить"></td>'. $temp. '
                        <td><input type="submit" name="deleteService"
value="Удалить"></td>
                    </tr>
                </table>
            </form>' ;
        ?>
    </div>

```

```

<?php
    if (isset($_POST['editService'])) {
        $sql = mysqli_query($connection, "UPDATE `Service`
        SET
            `description` = '". $_POST['serviceDescription']."' ,
            `cost` = '". $_POST['serviceCost']."'
        WHERE `Service`.`service_id` = '". $serviceId."'");

        header ('Location: Admin-Services.php'); // перенаправление на страницу
        exit();
    }
    if(isset($_POST['deleteService'])){
        $sql = mysqli_query($connection, "DELETE FROM `Service` WHERE
        `Service`.`service_id` = '". $serviceId);
        header ('Location: Admin-Services.php'); // перенаправление на страницу
        exit();
    }
?>
</body>

<footer>
    <?php
        if ($connection==false) {
            echo ' <p style="color: darkred">Ошибка подключения к БД:
            '.mysqli_connect_error().'</p>';
            exit();
        } else echo ' <p style="color: darkcyan">Успешное подключение к БД</p>';
    ?>
</footer>

<?php mysqli_close($connection);?>
</html>

```