



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

Московский техникум космического приборостроения

Отделение Информационные системы и программирование

ОТЧЕТ ПО УЧЕБНОЙ ПРАКТИКЕ
НА ТЕМУ: «УЧЁТ ОПЛАТЫ ЗА ГАЗ И ГАЗОВОЕ
ОБОРУДОВАНИЕ»

по профессиональному модулю ПМ.11 Разработка, администрирование и защита баз данных

Код, специальность 09.02.07 Информационные системы и программирование

Выполнил студент Владимиров Иван Сергеевич
(фамилия, имя, отчество)

Курс 2 Группа ТИП-42

Подпись студента

Оценка отлично

Дата присма зачета « 31 » июл 2023 г.

Руководитель практики ()
(подпись) (фамилия, имя, отчество)

2023 г.

Московский техникум космического приборостроения МГТУ имени Н.Э. Баумана

ЗАДАНИЕ на прохождение учебной практики

по: ПМ.11 Разработка, администрирование и защита баз данных в объеме 162 часа

Студент Владимиров Иван Сергеевич, 09.02.07, ТИП-42

(фамилия, имя, отчество; индекс специальности, группа)

Студент во время практики с «01» Мая 2023 г. по «31» Мая 2023 г.

должен:

Знать:

1. Основные положения теории баз данных, хранилищ данных, баз знаний;
2. Основные принципы структуризации и нормализации базы данных;
3. Основные принципы построения концептуальной, логической и физической модели данных;
4. Методы описания схем баз данных в современных системах управления базами данных;

Уметь:

1. Работать с современными case-средствами проектирования баз данных;
2. Проектировать логическую и физическую схемы базы данных;

Получить практический опыт:

В работе с объектами базы данных в конкретной системе управления базами данных; работе с документами отраслевой направленности.

По итогам учебной практики студент обязан представить:

1. Отчет по учебной практике (Титульный лист, Дневник практики, Письменный отчет, в который входит: текстовый документ, созданная БД, Аттестационный лист)

Дата выдачи задания «02» Мая 2023 г.

Руководитель практики от техникума

Осипова Н.М.

Студент

(подпись, дата)

Владимиров И.С.
(Фамилия И.О.)

Министерство науки и высшего образования Российской Федерации
Федерального государственного бюджетного образовательного учреждения высшего образования
Московский государственный технический университет имени Н.Э. Баумана
(национальный исследовательский университет)
Московский техникум космического приборостроения

ОТЧЕТ ПО ПРАКТИЧЕСКОЙ РАБОТЕ

по теме: **УЧЕТ ОПЛАТЫ ЗА ГАЗ И ГАЗООБОРУДОВАНИЯ**

Специальность: 09.02.07 Информационные системы и
программирование (квалификация «Программист»)

Группа ТИП-42

Проверил

Н. М. Осипова
С. В. Гончаренко
А. И. Медведев
О. Ю. Малыхина

Разработал

И. С. Владимиров

Москва 2023

СОДЕРЖАНИЕ

<u>ВВЕДЕНИЕ.....</u>	<u>5</u>
1 Анализ предметной области	8
2 Проектирование базы данных.....	13
2.1 Определение связей	14
2.2 Нотации Баркера	14
2.3 Сущности предметной области	16
2.4 Логическая модель	18
2.5 Создание таблиц с помощью SQL-кода.....	18
2.6 Физическая схема БД.....	21
3 Проектирование шаблона выходного документа.....	23
4 Проектирование интерфейса приложения.....	24
5 Тестирование разработанного приложения.....	29
<u>ЗАКЛЮЧЕНИЕ</u>	<u>32</u>
<u>СПИСОК ЛИТЕРАТУРЫ.....</u>	<u>34</u>
<u>ПРИЛОЖЕНИЕ А</u>	<u>35</u>
A.1 Листинг файла Entrance.php	35
A.2 Листинг файла login.php	37
A.3 Листинг файла top.php	39
A.4 Листинг файла map.php	42
A.5 Листинг файла profile_add.php.....	51
A.6 Листинг файла profile_add1.php.....	56
A.7 Листинг файла Profile.php	59
A.8 Листинг файла Profile1.php	64
A.9 Листинг файла Receipt.php	73
A.10 Листинг файла table_out.php	81
A.11 Листинг файла top1.php	90
A.12 Листинг файла buet.css.....	93
A.13 Листинг файла buet.js.....	116
A.14 Листинг файла enter.css	119
A.15 Листинг файла enter.js.....	130
A.16 Листинг файла index.css.....	131
A.17 Листинг файла index1.css.....	141
A.18 Листинг файла top.css	152
A.19 Листинг файла top.js.....	157

ВВЕДЕНИЕ

Современный мир невозможен без энергетики, которая является необходимым условием для функционирования всех отраслей экономики и комфортной жизни общества. Один из наиболее востребованных энергоносителей - газ - используется в быту, промышленности, транспорте и других сферах. От правильного использования газа и газооборудования зависит не только экономическая эффективность, но и благополучие окружающей среды.

В современной жизни учет оплаты за газ и газооборудование является одной из важнейших задач. Контроль затрат на газ и энергоэффективность применяемых газовых устройств позволяет не только снизить расходы, но и бережно относиться к природным ресурсам.

Однако, чтобы правильно решать эти задачи, необходимо обладать соответствующими знаниями и навыками. Учитывая значимость вопросов учета оплаты за газ и газооборудования, представляется актуальной разработка программного продукта, который позволит проводить автоматизированный учет и контроль этих ресурсов.

Программный продукт для учета оплаты за газ и газооборудование должен быть удобным и функциональным. Он должен позволять контролировать объем потребления газа, его стоимость, состояние газовых устройств и проводить мониторинг энергоэффективности. Кроме того, данное программное обеспечение должно обладать возможностью анализа полученных данных и формирования отчетности для оптимизации затрат на газ и газооборудование.

В наши дни все большую популярность набирают различные онлайн сервисы, позволяющие получить требуемую услугу, не выходя из дома, компьютеры используются почти повсеместно. Они с каждым годом все больше интегрируются в жизнь и рабочее пространство людей. С развитием компьютерной техники все больше рутинных работ было переадресовано автоматизированным информационным системам. Именно потому, самую

огромную ценность в мире сейчас имеет информация, и умение работать с информацией становится одним из ключевых факторов для бизнес-предпринимателей. Информация и управление её данными имеют огромную значимость для тех, кто хочет эффективно управлять своим предприятием. Для этого существуют базы данных, которые обеспечивают не просто хранение данных, а хранение их в структурированном виде. Автоматизированная система оплаты газа, позволяет обеспечить быструю оплату потребляемого газа и оптимизирует способ выдачи квитанций об оплате.

Информационные технологии очень сильно упрощают работу во всех сферах жизни, в том числе и в сфере услуг. Эта сфера нуждается в автоматизации из-за больших потоков данных, которые нужно хранить, обрабатывать и передавать. Создание программного обеспечения, автоматизирующего процесс оплаты газа, позволит значительно сократить время, требуемое на заполнение данных пользователя и выдачу квитанции. В данной работе представлен сайт 'Газкомплект', предназначенный для автоматизации процесса оплаты газа, а также формирования выходных квитанций для пользователя.

Рабочий процесс и БД неразделимые понятия. Базы данных способны содержать любую возможную информацию на буквально малом расстоянии, на расстоянии буквально одного клика. Коммерческие базы данных помогают руководящему составу систематизировать информацию о своих клиентах, задолженности и оплате. Продуктом моей учебной практики будет спроектированная база данных и приложение, которое будет реализовано в виде сайта для систематизирования и упрощения оплаты за газ.

Актуальность темы данной работы заключается в том, что автоматизация труда и создание базы данных позволяет повысить точность и скорость оплаты газа клиентам. Внедрение базы данных позволит вести учет задолженности, а также позволит быстро воспользоваться любой информацией о клиенте, тарифах и ценах.

Использование информационных систем в отслеживании данных о своих клиентов со стороны работы сайта позволит избежать большинство проблем,

которые были обусловлены, использованием бумажных носителей, например, таких как:

- повысится оперативность учета;
- уменьшится вероятность потери информации;
- на расчеты больше не будет влиять человечески фактор, потому что их будет выполнять машина по строго определенному алгоритму.
- информация будет выдаваться и храниться в структурированном виде;
- поиск информации будет занимать считанные секунды;

Предметом исследования является процесс автоматизации оплаты задолженностей за потребление газа для клиентов посредством создания базы данных MySQL, а также выполнение запросов с использованием СУБД MySQLWorkbench. Объектом исследования данной работы является компания 'ГазКомплект'.

Целью работы является создание базы данных, которая будет содержать информацию о предметной области.

Для достижения и решения поставленной цели необходимо решить следующие задачи:

1. Анализ предметной области.
2. Анализ существующих информационных систем, выполняющих схожие функции.
3. Создание логической и физической модели базы данных с описанием.
4. Спроектировать реляционную схему данных.
5. Создание диаграмм: контекстной диаграммы и диаграммы потоков данных с соответствующими пояснениями.
6. Разработка приложения
7. Создание приложения для управления базой данных.
8. Создание интерфейса.

1 Анализ предметной области

Предметная область предполагает онлайн – оплату газа, хранение данных об оплате, а также информацию о предоставляемых данных самим пользователем.

Основной задачей этой работы является разработка автоматизированного программного обеспечения деятельности газ учёта, которое поможет пользователю быстро оплатить газ.

Входными данными являются информация о пользователе и даты бронирования выбранного номера.

Выходными данными является забронированный номер и квитанция о брони, в которой отражены данные выбранного номера, даты бронирования, данные о пользователе.

Описание предметной области: Клиент решил оплатить счета за газ. Зайдя на сайт, он видит свои реквизиты, он должен заполнить информацию о используемом газе. В его распоряжении имеется тариф, сведения о котором находятся на сайте и обновляются в период определенного времени. Эти данные включают в себя цену и количество неоплаченных месяцев. Если клиент уверен, что хочет оплатить задолженность он оплачивает её с помощью сайта.

Сотрудники сайта обеспечивают обратную связь и полный контроль поступающих данных от клиента, затем формируют квитанцию об оплате. В результате работы у клиента появится документ, в котором будет прописаны все критерии оплаты газа, после чего они заносятся в базу данных.

В отчете прилагаются рисунки 1.1 и 1.2, на которых изображены контекстная диаграмма и её декомпозиция. На них можно наглядно увидеть принцип работы сайта.

На рисунке 1.1 представлена «Контекстная диаграмма».

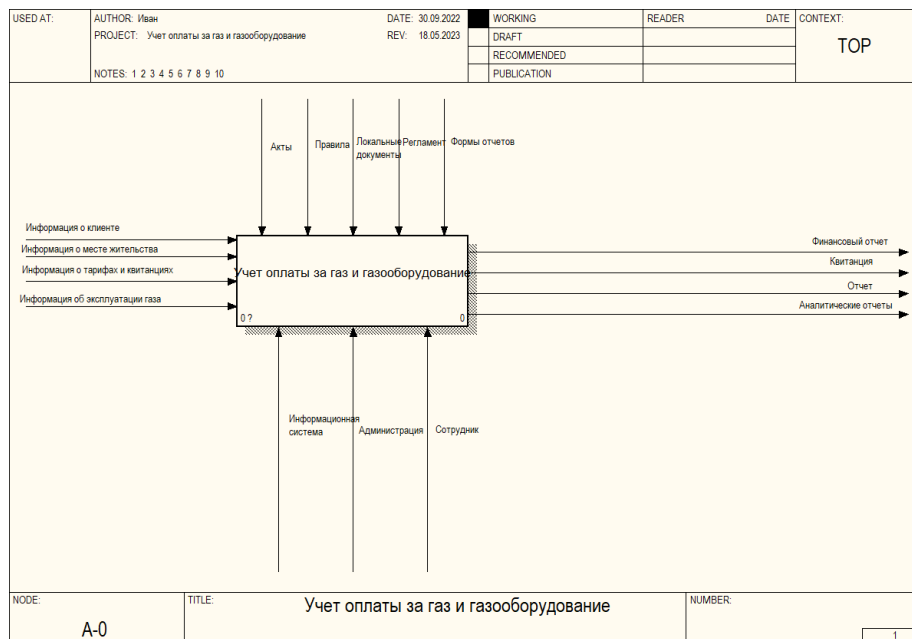


Рисунок 1.1 – Контекстная диаграмма IDEF0

На рисунке 1.2 представлена «Декомпозиция контекстной диаграммы».

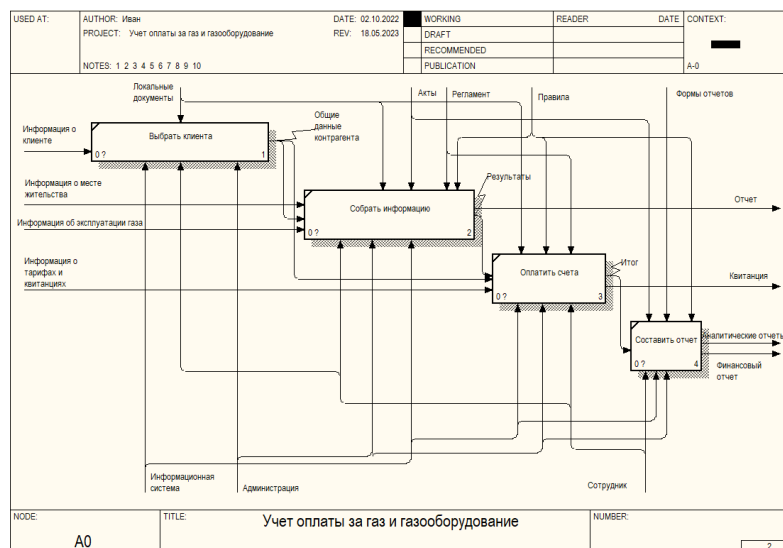


Рисунок 1.2 – Декомпозиция контекстной диаграммы

На сайте пользователь регистрируется, вводит данные, а затем производит оплату.

Ниже, на рисунках 1.3 и 1.4, представлена схема DFD, на ней отражен принцип работы сайта и процессы, связанные с потоками данных.

Далее, на рисунке 1.3 будет представлена диаграмма DFD, которая поможет лучше понять принцип работы приложения с точки зрения программы, связанных с потоками данных. Пользователи могут добавить информацию, а также при необходимости удалить или изменить её.

На рисунке 1.3 представлена «Контекстная диаграмма DFD».

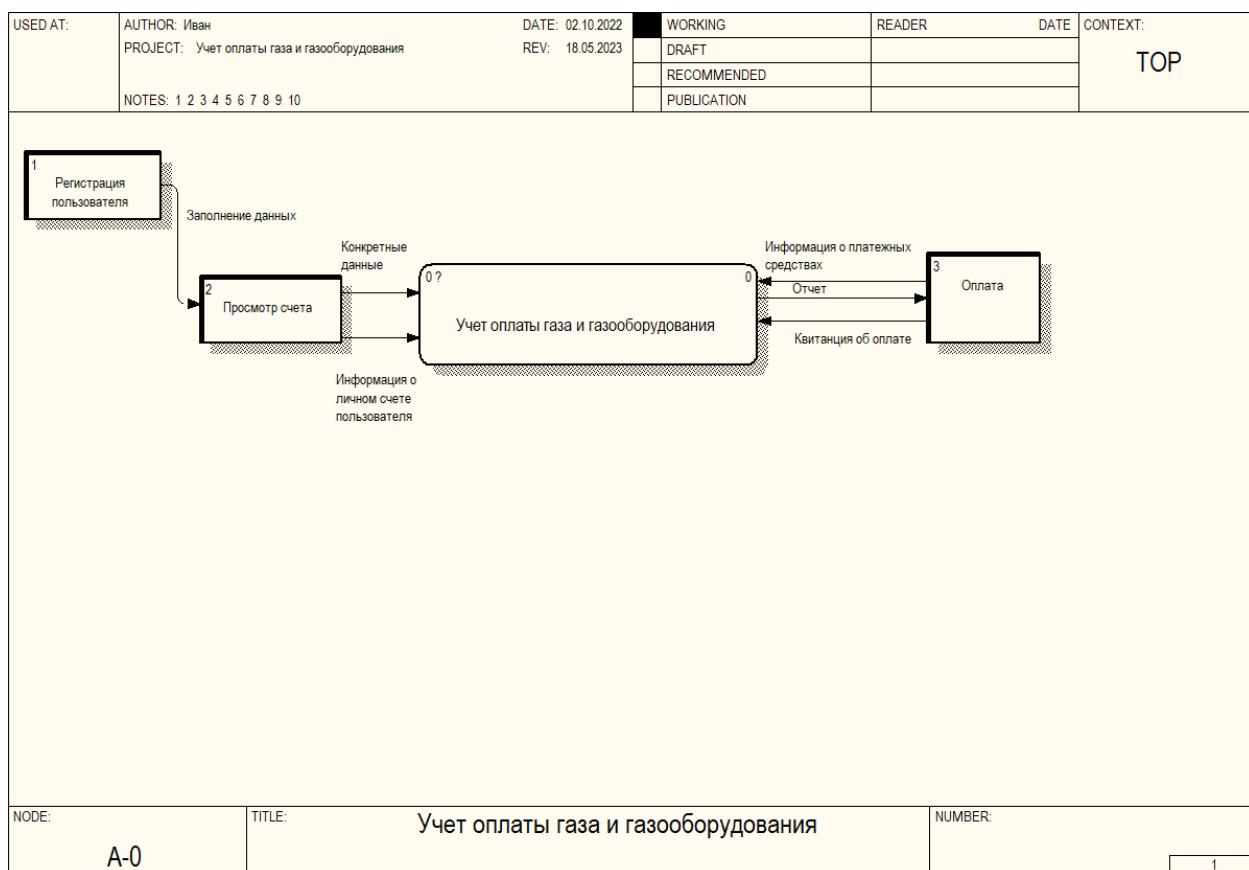


Рисунок 1.3 – Контекстная диаграмма DFD

На рисунке 1.4 представлена «Декомпозиция контекстной диаграммы DFD».

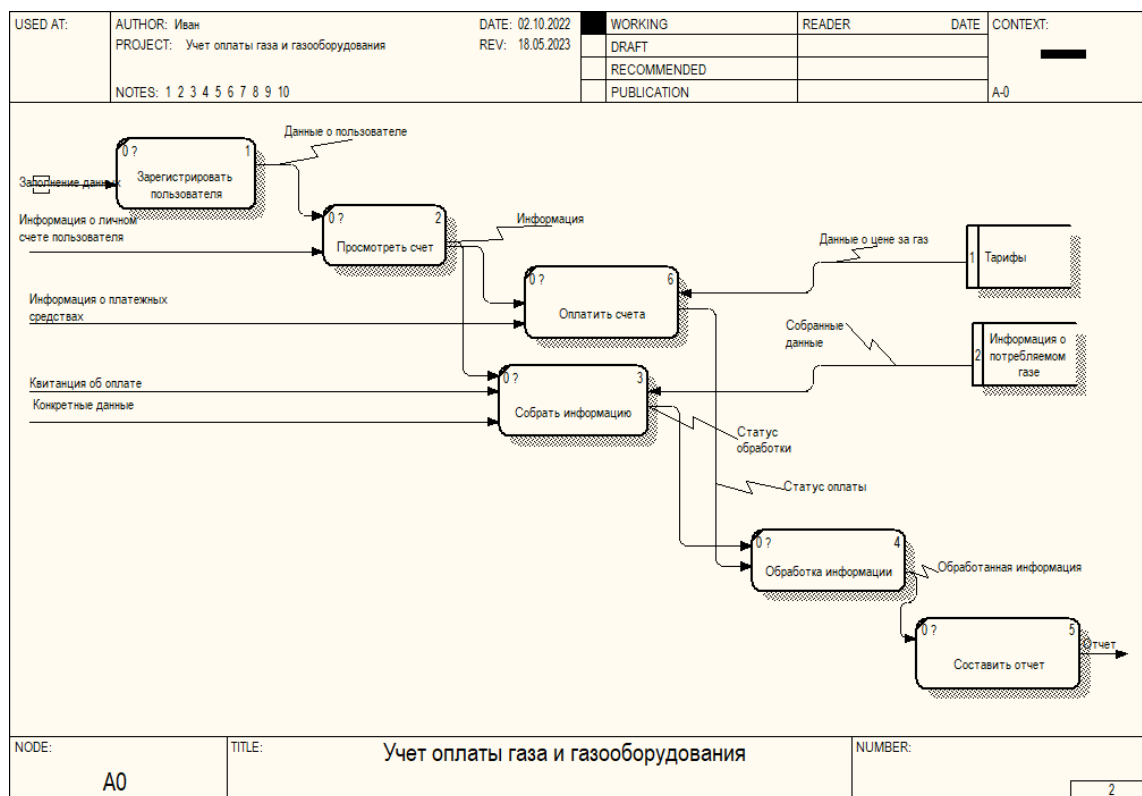


Рисунок 1.4 – Декомпозиция контекстной диаграммы DFD

После обращения пользователя происходит сбор и обработка данных. Информация о пользователе и потребляемом газе берется из базы данных. Выходными данными блока «Квитанция» является данные об оплате счета и личные данные пользователя.

Из этого мы можем выделить главную задачу сайта – это оплата газа, а также при необходимости автоматическое создание документов.

Подводя итог, можно выделить основные функциональные требования к разрабатываемому программному продукту:

1. Понятный для пользователя интерфейс и удобная навигация по сайту.
2. Обеспечение защиты информации от несанкционированного доступа и изменений.
3. Возможность заполнения личных данных, данных паспорта и данных об оплате.
4. Функция автоматического заполнения документа по шаблону.

5. Функция проверки вводимых данных.
6. Перехват и вывод ошибок в доступной форме в интерфейс при их возникновении.
7. Занесение всех вводимых данных в БД.
8. Возможность оплаты счета в любое время.

2 Проектирование базы данных

Основной задачей проектирования базы данных является обоснованный выбор такой структуры, которая обеспечит согласованное взаимодействие всех компонентов информационной системы согласно заданным требованиям и ограничениям по ряду интегрированных параметров.

Проектирование базы данных – процесс создания необходимых схем (концептуальной и логической). Для их создания необходимо выделить главные сущности:

1. Building – данные о здании, в котором проживает клиент, адрес вводит пользователь.
2. Apartament – данные о квартире клиента, вводимые при регистрации.
3. Client – информация о клиенте, его ФИО, паспорт, лицевой счет.
4. Profile – информация о профиле клиента: его почта и пароль.
5. Counter – счетчик, информация о том когда и где его изготовили, а также, что самое главное, результат показания счетчика.
6. Responsible – информация о проверяющем счетчик.
7. Checking – информация о проверки, то когда была проведена проверка, кем и результат проверки (1 – если все хорошо, 0 – если все плохо).
8. Taffiff_sheet – тарифный лист, информация о тарифе на газ.
9. Pay – оплата, информация об оплате.

Все связи являются бинарными и представляются линиями, соединяющими родительскую сущность с одной или несколькими сущностями потомками. Для связей должно быть определено имя (выражаемое грамматическим оборотом глагола), степень множественности (один или много объектов участвуют в связи) и степень обязательности (т.е. обязательная или необязательная связь между сущностями). Для множественной связи линия

присоединяется к прямоугольнику сущности в трех точках, а для одиночной связи – в одной точке. При обязательной связи рисуется непрерывная линия до середины связи, при необязательной – пунктирная линия.

Разработка ERD-диаграммы включает следующие основные этапы:

- Идентификация сущностей, их атрибутов, а также первичных и альтернативных ключей.
- Идентификация отношений между сущностями и указание типов отношений.
- Разрешение неспецифических отношений (отношений $n*m$).

Таким образом, для построения ERD-диаграммы необходимо иметь предварительно построенную DFD-диаграмму.

2.1 Определение связей

Определяем связи один ко многим:

«Building – Pay» – 1 ко М.

«Apartment– Pay» – 1 ко М.

«Taffiff_sheet – Pay» – 1 ко М.

«Client – Pay» – 1 ко М.

«Client – Profile» – 1 ко М.

«Counter – Pay » – 1 ко М.

«Counter – Checking » – 1 ко М.

« Checking – Responsible » – 1 ко М.

2.2 Нотации Баркера

В нотации Баркера используется только один тип диаграмм: собственно ERD-диаграммы. Сущность на ERD-диаграмме представляется прямоугольником любого размера, содержащим внутри себя имя сущности, список имен атрибутов (возможно, неполный) и указатели ключевых атрибутов

(знак «#» перед именем атрибута). Понятия категории и общей сущности Цена заменяются Баркером на эквивалентные понятия подтипа и супер-типа соответственно. Атрибуты сущностей могут быть обязательными и необязательными. Кроме того, атрибуты могут быть описательными (дескрипторами сущности) или входить в состав первичного ключа (уникального идентификатора сущности). Сущность, как правило, может быть полностью идентифицирована своим первичным ключом. Однако, могут быть случаи, когда в идентификации данной сущности участвуют также атрибуты другой сущности – родителя. Кроме первичных ключей могут использоваться альтернативные (возможные) ключи.

Следующим этапом разработки структуры базы данных является создание концептуальной модели данных. На рисунке 2.1 изображена концептуальная модель БД в нотации Баркера:

Предметная область, описанная ранее, представлена на рисунке 2.2 следующей нотацией Баркера.

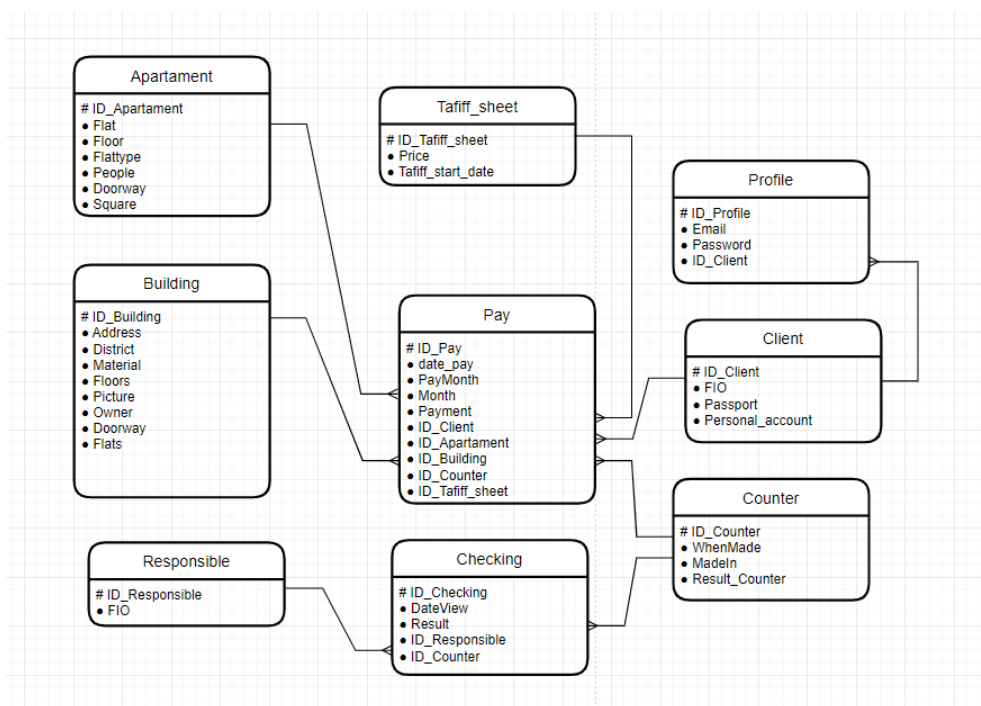


Рисунок 2.1 – Нотация Баркера

2.3 Сущности предметной области

Ниже представлено структурное описание таблиц базы данных в таблицах 2.1–2.10

Таблица 2.1 - Описание таблицы Building

Идентификатор	Описание	Тип данных	Размер
ID_Building	ID здания	Числовой	Целый
Address	Адрес	Текстовый	100
District	Район	Текстовый	100
Material	Материал, из которого сделано здание	Текстовый	100
Floors	Количество этажей	Числовой	Целый
Picture	Картинка	Бинарный	Переменный
Owner	Владелец здания	Текстовый	100
Doorway	Количество подъездов	Числовой	Целый
Flats	Количество квартир	Числовой	Целый

Таблица 2.2 - Описание таблицы Apartament

Идентификатор	Описание	Тип данных	Размер
ID_Apartament	ID квартиры	Числовой	Целый
Flat	Номер квартиры	Числовой	Целый
Floor	Номер этажа	Числовой	Целый
Flattype	Тип квартиры	Текстовый	100
People	Количество человек	Числовой	Целый
Doorway	Номер подъезда	Числовой	Целый
Square	Площадь квартиры	Числовой	Целый

Таблица 2.3 - Описание таблицы Client

Идентификатор	Описание	Тип данных	Размер
ID_Client	ID клиента	Числовой	Целый
FIO	ФИО клиента	Текстовый	100
Passport	Номер паспорта	Числовой	Целый
Personal_account	Лицевой счет	Числовой	Целый

Таблица 2.4 - Описание таблицы Profile

Идентификатор	Описание	Тип данных	Размер
ID_Profile	ID профиля	Числовой	10
Email	Почта	Текстовый	100
Password	Пароль	Текстовый	100
ID_Client	ID клиента	Числовой	Целый

Таблица 2.5 - Описание таблицы Counter

Идентификатор	Описание	Тип данных	Размер
ID_Counter	ID счетчика	Числовой	Целый
WhenMade	Когда был изготовлен	Дата	
MadeIn	Где был произведен	Текстовый	100
Result_Counter	Итоговое число счетчика	Числовой	Целый

Таблица 2.6 - Описание таблицы Responsible

Идентификатор	Описание	Тип данных	Размер
ID_Responsible	ID проверяющего	Числовой	Целый
FIO	ФИО проверяющего	Текстовый	100

Таблица 2.7 - Описание таблицы Checking

Идентификатор	Описание	Тип данных	Размер
ID_Checking	ID проверки	Числовой	Целый
DateView	Дата проверки	Дата	
Result	Результат проверки	Логический	
ID_Responsible	ID проверяющего	Числовой	Целый
ID_Counter	ID счетчика	Числовой	Целый

Таблица 2.8 - Описание таблицы Tafiff_sheet

Идентификатор	Описание	Тип данных	Размер
ID_Tafiff_sheet	ID тарифного листа	Числовой	Целый
Price	Цена по тарифу	Числовой	Целый
Tafiff_start_date	Дата введения тарифа	Дата	

Таблица 2.9 - Описание таблицы Pay

Идентификатор	Описание	Тип данных	Размер
ID_Pay	ID оплаты	Числовой	Целый
date_pay	Дата оплаты	Дата	
PayMonth	Итоговая цена оплаты	Числовой	Целый
Month	Количество не оплаченных месяцев	Числовой	Целый
Payment	Статус оплаты	Текстовый	11
ID_Building	ID здания	Числовой	Целый
ID_Apartment	ID квартиры	Числовой	Целый
ID_Client	ID клиента	Числовой	Целый
ID_Counter	ID счетчика	Числовой	Целый
ID_Tariff_sheet	ID тарифного листа	Числовой	Целый

2.4 Логическая модель

За концептуальной моделью следует разработка логической модели БД, представленной на рисунке 2.2

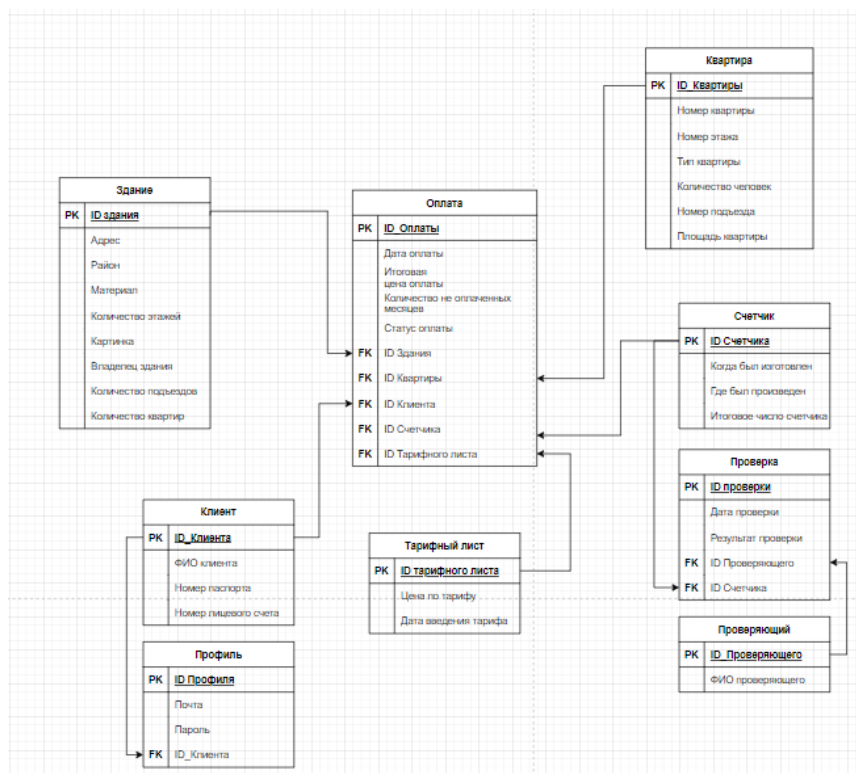


Рисунок 2.3 – Логическая схема базы данных

2.5 Создание таблиц с помощью SQL-кода

В работе была использована СУБД PHPMyAdmin. С помощью следующих запросов создаётся база данных «MYBD_Practice» и ее сущности.

```
drop database if exists MYBD_Practice;
create database MYBD_Practice;
use MYBD_Practice;
create table Building(
ID_Building INT PRIMARY KEY auto_increment,
Address varchar(100) NOT NULL,
District varchar(100) NOT NULL,
Material varchar(100) NOT NULL,
Floors int NOT NULL,
Picture Blob,
Owner varchar(100) NOT NULL,
Doorway INT NOT NULL,
Flats INT NOT NULL
);
CREATE TABLE Apartament (
ID_Apartament INT PRIMARY KEY auto_increment,
Flat INT NOT NULL,
Floor INT NOT NULL,
Flattype varchar(100) NOT NULL,
People INT NOT NULL,
Doorway INT NOT NULL,
Square INT NOT NULL
);
CREATE TABLE Client (
ID_Client INT PRIMARY KEY auto_increment,
```

```

FIO varchar(100) NOT NULL,
Passport INT NOT NULL,
-- Лицевой счет
Personal_account INT NOT NULL
);
CREATE TABLE Profile (
ID_Profile INT PRIMARY KEY auto_increment,
Email varchar(100) NOT NULL,
Password varchar(100) NOT NULL,
ID_Client INT NOT NULL,
foreign key (ID_Client) references Client (ID_Client)
);
CREATE TABLE Counter (
ID_Counter INT PRIMARY KEY auto_increment,
WhenMade DATE NOT NULL,
MadeIn varchar(100) NOT NULL,
Result_Counter INT NOT NULL
);

CREATE TABLE Responsible (
ID_Responsible INT PRIMARY KEY auto_increment,
FIO varchar(100) NOT NULL
);

CREATE TABLE Checking (
ID_Checking INT PRIMARY KEY auto_increment,
DateView DATE NOT NULL,
Result Boolean NOT NULL,
ID_Responsible INT NOT NULL,
foreign key (ID_Responsible) references Responsible (ID_Responsible),

```

```
ID_Counter INT NOT NULL,  
foreign key (ID_Counter) references Counter (ID_Counter)  
);
```

```
CREATE TABLE Tafiff_sheet (  
ID_Tafiff_sheet INT PRIMARY KEY auto_increment,  
Price DECIMAL NOT NULL,  
Tafiff_start_date Date  
);
```

```
CREATE TABLE Pay (  
ID_Pay INT PRIMARY KEY auto_increment,  
date_pay DATE,  
PayMonth INT NOT NULL,  
Month INT,  
Payment varchar(11),  
ID_Client INT NOT NULL,  
foreign key (ID_Client) references Client (ID_Client),  
ID_Apartament INT NOT NULL,  
foreign key (ID_Apartament) references Apartament (ID_Apartament),  
ID_Building INT NOT NULL,  
foreign key (ID_Building) references Building (ID_Building),  
ID_Counter INT NOT NULL,  
foreign key (ID_Counter) references Counter (ID_Counter),  
ID_Tafiff_sheet INT NOT NULL,  
foreign key (ID_Tafiff_sheet) references Tafiff_sheet (ID_Tafiff_sheet)  
);
```

2.6 Физическая схема БД

Физическая модель - логическая модель базы данных, выраженная в терминах языка описания данных конкретной СУБД.

Физическая модель базы данных содержит все детали, необходимые конкретной СУБД для создания базы: наименования таблиц и столбцов, типы данных, определения первичных и внешних ключей и т.п.

Физическая модель строится на основе логической с учетом ограничений, накладываемых возможностями выбранной СУБД.

На рисунке 2.3 представлена схема готовой базы данных “MYBD_Practice” в RHPMyAdmin.

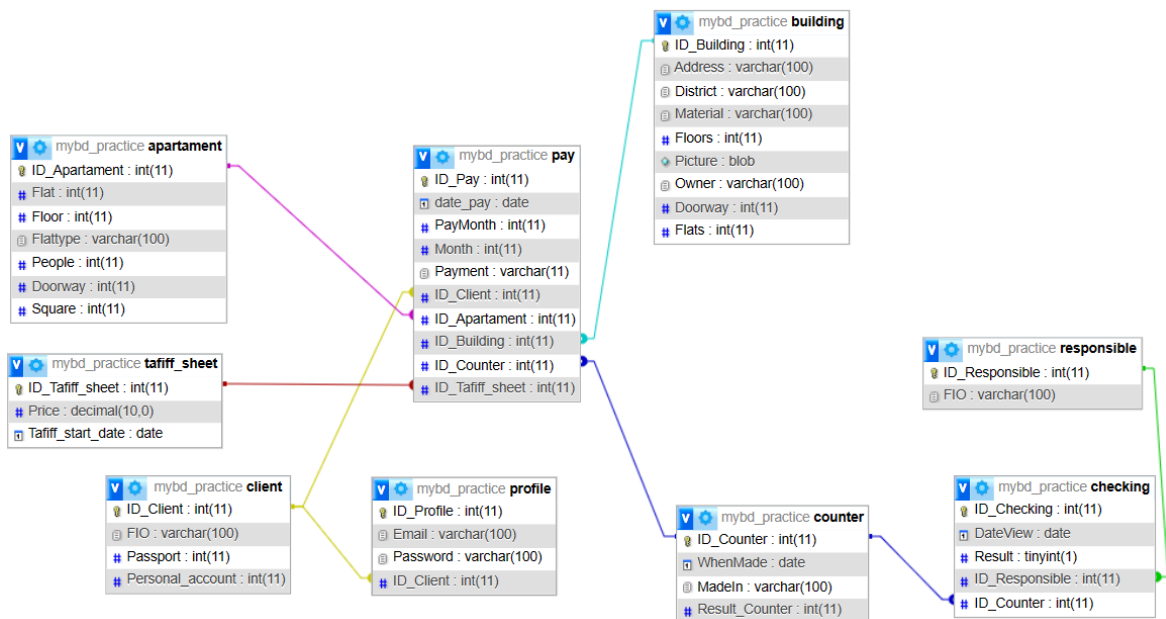


Рисунок 2.3 – Физическая схема в RHPMyAdmin

3 Проектирование шаблона выходного документа

Выходной документ – документ, являющийся носителем результатов обработки данных. Выходной документ, формируемый автоматизированной системой, и выдается системными средствами вывода.

Шаблон выходного документа представляет собой созданный заранее текстовый документ со специальными системными словами-переменными, которые будут заменяться данными, введенными пользователем.

Шаблон выходного документа составления путевки изображен на рисунке 3.1.


 Газкомплект	
Извещение	Форма ПД №4
May 18, 2023	
Ф.И.О. плательщика	Vladimirov Ivan Sergeevich
Номер карты	6721 4626 4795 6432
Сумма к оплате	5000руб.
Печать:	
	

Рисунок 3.1 – Шаблон выходного документа

4 Проектирование интерфейса приложения

Дизайн пользовательского интерфейса является фактором, оказывающим влияние на три основных показателя качества программного продукта: его функциональность, красивый внешний вид и производительность.

Функциональность является фактором, на который разработчики приложений зачастую обращают основное внимание. Они пытаются создавать программы так, чтобы пользователи могли выполнять свои задачи и им было удобно это делать. Функциональность важна, но, тем не менее, это не единственный показатель, который должен учитываться в ходе разработки.

Красивый внешний вид самого приложения и способа его представления позволяет сформировать у потребителя положительное мнение о программе. Однако эстетические характеристики весьма субъективны и описать их количественно гораздо труднее, чем функциональные требования или показатели производительности. Вся красота приложения зачастую сводится к простому выбору: соотносятся ли между собой используемые цвета, передают ли элементы интерфейса их назначение и смысл представляемых операций, что ощущает человек при использовании тех или иных элементов управления и насколько успешно он их использует.

Производительность и надежность, также влияют на перспективу применения программы. Если приложение хорошо выглядит, имеет простое и удобное управление, но, к примеру, медленно прорисовывает экраны, регулярно «подвисает» на десяток-другой секунд или, того хуже, падает с критической ошибкой при некорректных действиях пользователя, у него, вероятно, будет мало шансов на длительную эксплуатацию. В свою очередь, быстрая и стабильная работа приложения могут отчасти

компенсировать его не самый стильный дизайн или отсутствие каких-то вторичных функций.

Для обеспечения успешной работы пользователя от дизайнера интерфейса требуется соблюдать баланс между вышеперечисленными факторами на протяжении всего жизненного цикла разработки приложения. Это достигается последовательной и тщательной проработкой деталей интерактивного взаимодействия на каждом из этапов разработки пользовательского интерфейса, включающих:

1. Проектирование

Функциональные требования: определение цели разработки и исходных требований.

Анализ пользователей: определение потребностей пользователей, разработка сценариев, оценка соответствия сценариев ожиданиям пользователей.

Концептуальное проектирование: моделирование процесса, для которого разрабатывается приложение.

Логическое проектирование: определение информационных потоков в приложении.

Физическое проектирование: выбор платформы, на которой будет реализован проект и средств разработки.

2. Реализация

Прототипирование: разработка бумажных и/или интерактивных макетов экранных форм.

Конструирование: создание приложения с учетом возможности изменения его дизайна.

3. Тестирование

В соответствии с этими правилами был разработан интуитивно понятный интерфейс приложения. Интерфейс программы разработан с помощью CSS и HTML.

На рисунках с 4.1-4.5 представлено меню настроек пользователя на сайте, в котором можно заполнить данные о клиенте

На рисунке 4.1 изображена регистрация и вход на сайт.

The image shows a web interface for 'Компания 'ГазКомплект''. The main heading is 'Регистрация'. Below it, there are three input fields: 'ФИО', 'Email', and 'Пароль'. A dark button labeled 'Зарегистрироваться' is positioned below the password field. At the bottom of the form area, there is a link labeled 'Вход'. The background of the form area features a landscape with hot air balloons.

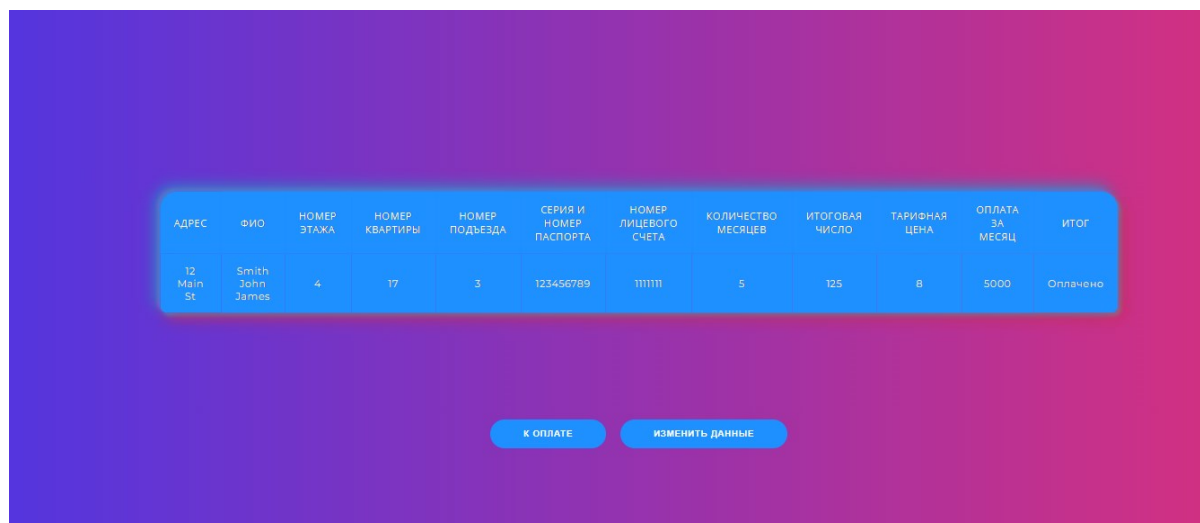
Рисунок 4.1 – Фотография входа

На рисунке 4.2 изображено регистрационное поле для ввода данных.

The image displays a registration form on the left side of the screen, with a background image of a gas stove burner on the right. The form is titled 'ДОБРО ПОЖАЛОВАТЬ' and includes a link 'Введите свои данные'. It is divided into several sections for user information: 'Информация о пользователе' (with a profile icon and name 'Игорь Елизар Константинович'), 'Адрес' (with 'Адрес проживания' and 'Номер квартиры'), 'Номер подъезда' and 'Номер этажа', 'Лицевой счет' (with 'Номер лицевого счета' and 'Номер паспорта'), 'Показатель счетчика' (with 'Число' and 'Дата'), 'Инициалы привязываемого' (with 'Инициалы привязываемого' and 'Дата проверки'), and 'Площадь квартиры' (with 'Площадь квартиры' and 'Число месяцев'). At the bottom, there is a section for payment: 'Оплата через SberPay' with a 'Сбербанк' button. The background image on the right shows a close-up of a gas stove burner with a blue flame.

Рисунок 4.2 – Фотография регистрации

На рисунке 4.3 изображена таблица, в которой отображаются данные пользователя.



АДРЕС	ФИО	НОМЕР ЭТАЖА	НОМЕР КВАРТИРЫ	НОМЕР ПОДЪЕЗДА	СЕРИЯ И НОМЕР ПАСПОРТА	НОМЕР ЛИЦЕВОГО СЧЕТА	КОЛИЧЕСТВО МЕСЯЦЕВ	ИТОГОВАЯ ЧИСЛО	ТАРИФНАЯ ЦЕНА	ОПЛАТА ЗА МЕСЯЦ	ИТОГ
12 Main St	Smith John James	4	17	3	123456789	111111	5	125	8	5000	Оплачено

Рисунок 4.3 – Фотография таблицы с информацией пользователя

На рисунке 4.4 изображено регистрационное поле для ввода данных уже зарегистрированного пользователя.



ДОБРО ПОЖАЛОВАТЬ
Введите свои данные

Информация о пользователе

ФИО
Smith John James

Адрес
12 Main St

Номер квартиры
17

Номер подъезда
3

Номер этажа
4

Лицевой счет
1111111

Номер паспорта
123456789

Показ. счетчика
Число

Д. И. счетчика
2022-04-29

ФИО проверяющего
Smith John James

Д. П. счетчика
2022-05-02

Площадь квартиры
50

Ч. И. месяцев
Число месяцев

Оплата через SberPay

Сохранить

Рисунок 4.4 – Фотография изменения данных

На рисунке 4.5 изображен блок ввода данных в поле оплаты.

Оплата через SberPay

Номер карты

Номер

Владелец карты

Введите ФИО

Срок действия

Месяц Год

CVV

Номер

Итоговый счет

Р 0 .00

Отмена Оплата

Рисунок 4.5 – Фотография карты

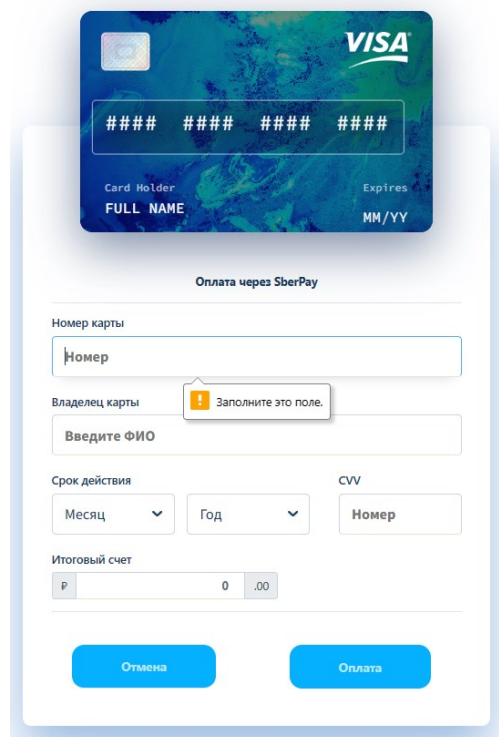
5 Тестирование разработанного приложения

Работоспособность программы – это состояние программы, при котором она выполняет необходимые требования, установленные при анализе предметной области. Тестирование разработанного программного средства производится с целью выявления каких-либо ошибок (синтаксических, алгоритмических и других). Иными словами, тестирование позволяет установить то, насколько корректно разработанное программное средство работает.

Кроме этого, производится функциональное тестирование программного средства, которое проверяет, насколько правильно работают алгоритмы.

Дальше будут представлены примеры ошибок, которые можно допустить в процессе эксплуатации программы.

Примеры возможных ошибок представлены на рисунках 6.1 – 6.4.



The image shows a payment interface for SberPay. At the top, there is a Visa card with a blue and green abstract pattern. Below the card, the text 'Оплата через SberPay' is displayed. The form contains several input fields: 'Номер карты' (Card number) with a placeholder 'Номер', 'Владелец карты' (Cardholder) with a placeholder 'Введите ФИО' and a tooltip 'Заполните это поле.' (Fill in this field.), 'Срок действия' (Expiration date) with dropdowns for 'Месяц' (Month) and 'Год' (Year), and 'CVV' with a placeholder 'Номер'. At the bottom, there is a field for 'Итоговый счет' (Total amount) with a placeholder 'Р 0 .00'. Two buttons, 'Отмена' (Cancel) and 'Оплата' (Pay), are located at the bottom of the form.

Рисунок 5.1 – Ошибка пустого ввода личных данных

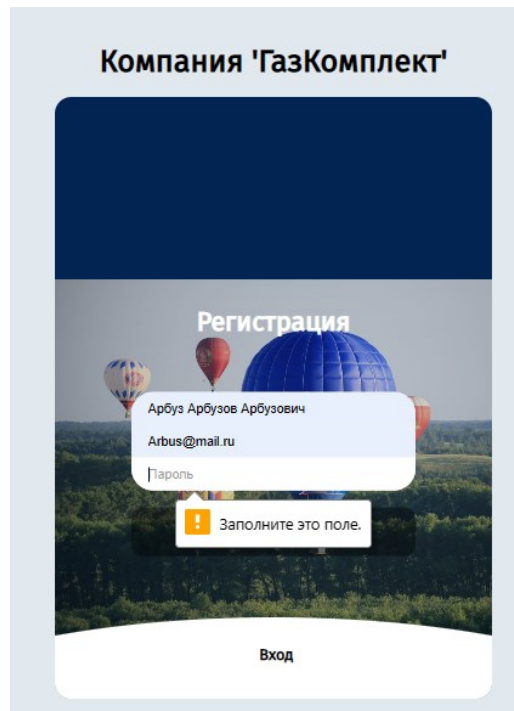


Рисунок 5.2 – Ошибка пустого ввода личных данных

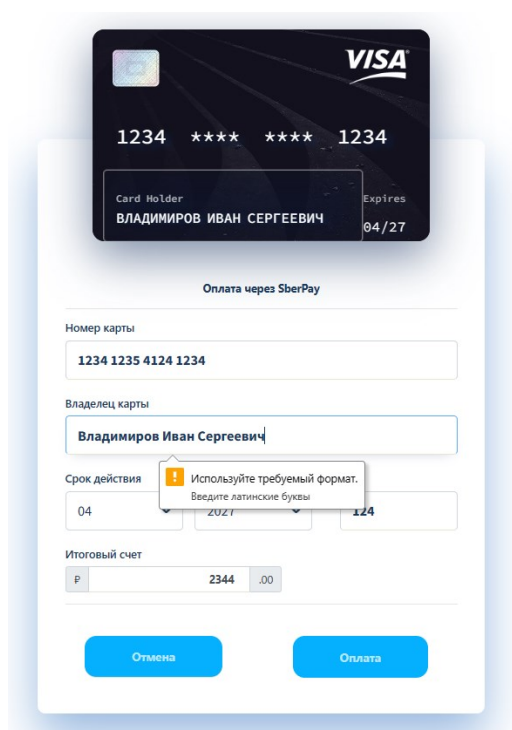


Рисунок 5.3 – Ошибка ввода в поле владельца карты

Компания 'ГазКомплект'

Регистрация

Вход

Email

Пароль

Неверно введен логин или пароль
Забыли пароль?

Войти

Рисунок 5.4 – Ошибка ввода неправильного пароля

ЗАКЛЮЧЕНИЕ

В ходе работы был создан сайт для автоматизации деятельности оплаты газа, заключенной в выборе добавление и изменение информации о пользователе, позволяет полноценно работать с базой данных по заданной предметной области и облегчает процесс создания и обработки данных пользователей, а также создания квитанции об оплате.

В заключении проанализируем проведенную во время практики выполненную работу. В данной работе была спроектирована и разработана база данных туров, которая содержит основную информацию, необходимую для автоматизации газовой компании. А также приложение, которое наглядное для пользователя. Для этого, были решены соответствующие задачи:

- проанализирована предметная область;
- определены бизнес-правила в организации;
- проведен анализ существующих информационных систем;
- спроектирована инфологическая схема данных;
- спроектирована реляционная схема данных;
- разработана физическая база данных для СУБД;
- физическая база данных была выгружена в СУБД MySQL;
- разработан интерфейс приложения;
- создана программа.

Полученная база данных соответствует правилам целостности и пригодна для ведения учета. В дальнейшей перспективе возможно ее усложнение, расширение функционала и возможностей.

Проектирование базы данных позволяет более подробно рассмотреть любой проектируемый продукт, также они позволяют свести любые ошибки системы, возникающие в ходе использования базы данных, к минимуму.

СПИСОК ЛИТЕРАТУРЫ

- 1) Гражданский кодекс Российской Федерации, Закон РФ от 07.02.1992 г. №2300-1 «О защите прав потребителей».
- 2) Кодекс Российской Федерации об административных правонарушениях.
- 3) Кодекс Российской Федерации об административных правонарушениях.
- 4) Федеральный закон Российской Федерации «О товарных биржах и биржевой торговле».
- 5) "Учет газа в промышленности и жилищном строительстве" автор Николай Иванович Петров, издательство "Энергия", 2019 год.
- 6) Мигель Гринберг. Flask Web Development. – М.: Юрайт, 2018.
- 7) Информационная система «Read the Docs». URL: <https://readthedocs.org>
- 8) Робсон Э. "Изучаем HTML, XHTML и CSS".
- 9) "Учет газа в промышленности и жилищном строительстве" автор Николай Иванович Петров, издательство "Энергия", 2019 год.
- 10) "Системы газоснабжения и газоиспользования" авторы Евгений Геннадьевич Габдулин, Виктор Дмитриевич Рыбакин, издательство "Лань", 2019 год.
- 11) "Основы газоснабжения" автор Семен Андреевич Марков, издательство "Новая экономика", 2019 год.
- 12) "Экономика газоиспользования" автор Андрей Игоревич Журавлев, издательство "Экзамен", 2020 год.
- 13) "Технологии и системы газоснабжения" автор Дмитрий Андреевич Петров, издательство "Техносфера", 2018 год.
- 14) Паленов, В. Н. Базы данных / В.Н. Паленов, Д.Э. Фуфаев. - М.:Академия, 2019 год - 189 с

- 15) Лутц, Марк. Л86 Изучаем Python, том 1, 5-е изд.: Пер. с англ. — СПб.: ООО “Диалектика”, 2019
- 16)"Mastering CSS: Advanced Techniques for Modern Web Design" 2019 года
- 17)Чистый Python. Тонкости программирования для профи, Бейдер Дэн
- 18)Статья «Организация учета газа и оплаты за его потребление. Услуги газоснабжения в России».
- 19)Статья «Оказание услуг по учету газа: особенности и требования. Межведомственный учет энергоресурсов».
- 20)Статья «Организация учета газа и оплаты за его потребление. Услуги газоснабжения в России»
- 21)Статья «Установка, настройка и эксплуатация газового оборудования: руководство для владельцев».
- 22)Инструкция по эксплуатации газового счетчика.
- 23)Инструкция по установке газового счетчика.
- 24)Руководство по эксплуатации газооборудования.
- 25)Правила эксплуатации газа и газового оборудования.
- 26)Технические нормы и правила для газово-электрических установок.
- 27)Информационная система «Erase.bg» <https://www.erase.bg/ru/upload>
- 28)Информационная система «Icons8.com»
<https://icons8.com/icon/set/user/windows>
- 29)Информационная система «Github» <https://github.com>

ПРИЛОЖЕНИЕ А

А.1 Листинг файла Entrance.php

```
<html lang="ru">
<link>
<meta charset="UTF-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<link rel="stylesheet" href="enter.css"> </link>
<title>Document</title>
</head>
<body>
<header id="header" class="header">
<div class="container">
<h1 >Компания 'ГазКомплект'</h1>
<div>
</header>
<section id="about" class = "about">
<div class="form-structor">
<form action="login.php" method="POST">
<div class="signup">
<h2 class="form-title" id="signup"><span></span>Регистрация</h2>
<div class="form-holder">
<input type="FIO" class="input" name="FIO" id="FIO" pattern="[А-Яа-яЁё
]+"title="Только русские буквы" maxlength="50" placeholder="ФИО"
required="required"/>
<input type="email" class="input" placeholder="Email" name="Email"
id="Email" pattern="[a-zA-Z0-9._%+-]+@[a-zA-Z]+\.[a-zA-Z]{0,4}"
```

```

required="required" required oninvalid="setCustomValidity('Ошибка:не введен @
или точка')" oninput="setCustomValidity('')"/>
        <input type="password" class="input" placeholder="Пароль"
name="Password" required/>
    </div>
    <button class="submit-btn">Зарегистрироваться</button>
</div>
</form>
<form action="top.php" method="POST">
    <div class="login slide-up">
        <div class="center">
            <h2 class="form-title" id="login"><span></span>Вход</h2>
            <div class="form-holder">
                <input type="email" class="input" style="text-transform:none" name="Email"
placeholder="Email" required="required" required
oninvalid="setCustomValidity('Ошибка:не введен @ или точка')"
oninput="setCustomValidity('')"/>
                <input type="password" class="input" style="text-transform:none"
name="Password" placeholder="Пароль" required="required" />
            </div>
        </div>
    </div>
</form>
<?php
session_start();
if (isset($_SESSION['message'])) {
    echo "<p class='parrol' > ". $_SESSION['message'] . "</p>";
}
else {
    unset($_SESSION['message']);
}
?>

```



```

        <a                                class='parrol'
href="mailto:ivan_vladimirov_888@mail.ru?subject=Здравствуйте, у меня
проблема со входом. Не могли бы вы мне помочь?">Забыли пароль?</a>

    </div>

    <button name='formSubmit'class="submit-btn">Войти</button>

</div>

</div>

</form>

</div>

</section>

<footer id="footer" class="footer">

<div class="container">

<p class="overflow">

    Возникли проблемы? Обратитесь в поддержку: <i class="fa fa-heart"></i>

    <a      target="_blank"      href="https://vk.com/ivan_vladimirov_888">Иван
    Владимиров</a>

    Можете поддержать финансово &#9829;

</p>

</div>

</footer>

<script type="text/javascript" src="enter.js"></script>

</body>

</html>

```

А.2 Листинг файла login.php

```

<?php
session_start();
$serverName = "localhost";
$username = "root";
$pass = "";

```

```

$dbName = "mybd_practice";
$conn = mysqli_connect($serverName, $username, $pass, $dbName);
if (!$conn) { die("Connection failed: " . mysqli_connect_error()); }
$email = $_POST['Email'];
$password = $_POST['Password'];
$FIO = $_POST['FIO'];
$_SESSION['Email'] = $email;
$_SESSION['Password'] = $password;
$sql = "SELECT (max(ID_Client)+1) as ID_Client from Client";
$result = mysqli_query($conn, $sql);
$row = mysqli_fetch_assoc($result);
$id_client = $row['ID_Client'];
$sql_check = "SELECT Email FROM Profile WHERE Email='$email'";
$result_check = mysqli_query($conn, $sql_check);
if(mysqli_num_rows($result_check)>0){
    $new_url = 'Entrance.php';
    header('Location: '.$new_url);
    $_SESSION['message1'] = 'Такой Email уже зарегистрирован!';
} else {
    $sql1 = "INSERT INTO Client ( FIO, Passport,Personal_account) VALUES
('$FIO', ',')";
    $sql2 = "INSERT INTO Profile (Email, Password, ID_Client) VALUES
('$email', '$password', '$id_client')";
    if (mysqli_query($conn, $sql1)) {
        if(mysqli_query($conn, $sql2)){
            unset($_SESSION['message1']);
            $new_url = 'Profile.php';
            header('Location: '.$new_url);
            mysqli_close($conn);
        } else {

```

```

echo "Error: " . $sql2 . " " . mysqli_error($conn);
$new_url = 'Entrance.php';
header('Location: '.$new_url);
$_SESSION['message1'] = 'Ошибка при регистрации!';
}
} else {
echo "Error: " . $sql1 . "<br>" . mysqli_error($conn);
}
}
?>

```

A.3 Листинг файла top.php

```

<!DOCTYPE html>
<html lang="ru" >
<head>
<script
src="https://cdnjs.cloudflare.com/ajax/libs/jquery/2.0.2/jquery.min.js"></script>
<meta charset="UTF-8">
<title>база данных</title>
<link rel="stylesheet" href="/top.css">
</head>
<body>
<!-- partial:index.partial.html -->
<html>
<head>
<script
src="https://cdnjs.cloudflare.com/ajax/libs/jquery/2.0.2/jquery.min.js"></script>
</head>
<body id="gradient">

```

```

</body>
</html>
<!-- partial -->
<script src="./top.js"></script>
<div class="form">
<div class="table">
<?php
session_start();
$serverName = "localhost";
$username = "root";
$pass = "";
$dbName = "mybd_practice";
// Установка соединения с базой данных
$conn = mysqli_connect($serverName, $username, $pass, $dbName);
// Retrieve data from the "Profile" table
$email = $_POST['Email'];
$password = $_POST['Password'];
$_SESSION['Email'] = $email;
$_SESSION['Password'] = $password;

$sql = "SELECT Building.Address AS Address, Client.FIO AS FIO,
Apartament.Floor AS Floor, Apartament.Flat AS Flat, Apartament.Doorway AS
Doorway, Client.Personal_account AS Personal_account, Client.Passport AS Passport,
Pay.Month AS Month, Counter.Result_Counter As Result_Counter, Tafiff_sheet.Price
AS Price, Pay.PayMonth AS PayMonth, Pay.Payment AS Payment
From Client Inner join Profile ON Client.ID_Client = Profile.ID_Client
Inner join Pay on Client.ID_Client = Pay.ID_Client
Inner join Apartament on Apartament.ID_Apartament = Pay.ID_Apartament
Inner join Building on Building.ID_Building = Pay.ID_Building
Inner join Tafiff_sheet on Tafiff_sheet.ID_Tafiff_sheet = Pay.ID_Tafiff_sheet
Inner Join Counter on Counter.ID_Counter = Pay.ID_Counter

```

```

Where Email = '$Email' and Password = '$Password';
$result = mysqli_query($conn, $sql);
echo "<table>";
echo "<tr><th>Адрес</th><th>ФИО</th><th>Номер этажа</th><th>Номер
квартиры</th><th>Номер подъезда</th><th>Серия и номер
паспорта</th><th>Номер лицевого счета</th><th>Количество
месяцев</th><th>Итоговая число</th><th>Тарифная цена</th><th>Оплата за
месяц</th><th>Итог</th></tr>";
// Display the data in a table
if (mysqli_num_rows($result) > 0) {
while($row = mysqli_fetch_assoc($result)) {
unset($_SESSION['message']);
echo "<tr>";
echo "<td>" . $row["Address"] . "</td>";
echo "<td>" . $row["FIO"] . "</td>";
echo "<td>" . $row["Floor"] . "</td>";
echo "<td>" . $row["Flat"] . "</td>";
echo "<td>" . $row["Doorway"] . "</td>";
echo "<td>" . $row["Passport"] . "</td>";
echo "<td>" . $row["Personal_account"] . "</td>";
echo "<td>" . $row["Month"] . "</td>";
echo "<td>" . $row["Result_Counter"] . "</td>";
echo "<td>" . $row["Price"] . "</td>";
echo "<td>" . $row["PayMonth"] . "</td>";
echo "<td>" . $row["Payment"] . "</td>";
echo "</tr>";
}
} else {
$new_url = 'Entrance.php';
header('Location: '.$new_url);

```

```

$_SESSION['message'] = 'Неверно введен логин или пароль';
}
mysqli_close($conn);
echo "</table>";
?>
</div>
<div class="Button">
<form action="Map.php" method="POST">
<button class="btn" type="submit" >К оплате</button>
</form>
<form action="Profile1.php" method="POST">
<button style="margin-left: 20px" class="btn" type="submit" >Изменить
данные</button>
</form>
</div>
</div>
</body>
</html>

```

А.4 Листинг файла map.php

```

<!DOCTYPE html>
<html lang="ru" >
<head>
<meta charset="UTF-8">
<title>CodePen - Credit Card Form - VueJs</title>
<link rel="stylesheet" href="/buet1.css">
<link href="//maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css"
rel="stylesheet" id="bootstrap-css">
<script src="//cdnjs.cloudflare.com/ajax/libs/jquery/3.2.1/jquery.min.js">

```

```

</script>
<?php
session_start();
?>
</head>
<body>
<!-- partial:index.partial.html -->
<form action="Receipt.php" method="POST" >
<div class="wrapper" id="app">
<div class="card-form">
<div class="card-list">
<div class="card-item" v-bind:class="{ '-active' : isCardFlipped }">
<div class="card-item__side -front">
<div class="card-item__focus" v-bind:class="{ '-active' : focusElementStyle }"
v-bind:style="focusElementStyle" ref="focusElement"></div>
<div class="card-item__cover">

</div>
<div class="card-item__wrapper">
<div class="card-item__top">

<div class="card-item__type">
<transition name="slide-fade-up">


```

```

</transition>
</div>
</div>
<label for="cardNumber" class="card-item__number" ref="cardNumber">
<template v-if="getCardType === 'amex'">
<span v-for="(n, $index) in amexCardMask" :key="$index">
<transition name="slide-fade-up">
<div
class="card-item__numberItem"
v-if="$index > 4 && $index < 14 && cardNumber.length > $index && n.trim()
!== ""
>*</div>
<div class="card-item__numberItem"
:class="{ '-active' : n.trim() === '' }"
:key="$index" v-else-if="cardNumber.length > $index">
{{cardNumber[$index]}}
</div>
<div
class="card-item__numberItem"
:class="{ '-active' : n.trim() === '' }"
v-else
:key="$index + 1"
>{{n}}</div>
</transition>
</span>
</template>
<template v-else>
<span v-for="(n, $index) in otherCardMask" :key="$index">
<transition name="slide-fade-up">
<div

```



```

class="card-item__numberItem"
v-if="$index > 4 && $index < 15 && cardNumber.length > $index && n.trim()
!=="
>*</div>
<div class="card-item__numberItem"
:class="{ '-active' : n.trim() === " }"
:key="$index" v-else-if="cardNumber.length > $index">
{{cardNumber[$index]}}
</div>
<div
class="card-item__numberItem"
:class="{ '-active' : n.trim() === " }"
v-else
:key="$index + 1"
>{{n}}</div>
</transition>
</span>
</template>
</label>
<div class="card-item__content">
<label for="cardName" class="card-item__info" ref="cardName">
<div class="card-item__holder">
Card Holder
</div>
<transition name="slide-fade-up">
<div class="card-item__name" v-if="cardName.length" key="1">
<transition-group name="slide-fade-right">
<span class="card-item__nameItem" v-for="(n, $index) in
cardName.replace(/\s\s+/g, ' ')" v-if="$index === $index" v-bind:key="$index +
1">{{n}}</span>

```

```

</transition-group>
</div>
<div class="card-item__name" v-else key="2">Full Name</div>
</transition>
</label>
<div class="card-item__date" ref="cardDate">
<label for="cardMonth" class="card-item__dateTitle">Expires</label>
<label for="cardMonth" class="card-item__dateItem">
<transition name="slide-fade-up">
<span v-if="cardMonth" v-bind:key="cardMonth">{{cardMonth}}</span>
<span v-else key="2">MM</span>
</transition>
</label>
/
<label for="cardYear" class="card-item__dateItem">
<transition name="slide-fade-up">
<span
                                v-if="cardYear"                                v-
bind:key="cardYear">{{String(cardYear).slice(2,4)}}</span>
<span v-else key="2">YY</span>
</transition>
</label>
</div>
</div>
</div>
</div>
<div class="card-item__side -back">
<div class="card-item__cover">

    </div>
    <div class="card-item__band">
    </div>
    <div class="card-item__cvv">
    <div class="card-item__cvvTitle">
    CVV
    </div>
    <div class="card-item__cvvBand">
    <span v-for="(n, $index) in cardCvv" :key="$index">
    *
    </span>
    </div>
    <div class="card-item__type">
    
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    <div class="card-form__inner">
    <div>
    <label class="Nazv">Оплата через SberPay</label>
    </div>
    <hr>
    <div class="card-input">

```

```

<label for="cardNumber" class="card-input__label">
Номер карты
</label>

<input type="text" id="cardNumber" name="cardNumber" class="card-
input__input" v-mask="generateCardNumberMask" v-model="cardNumber" v-
on:focus="focusInput" v-on:blur="blurInput" data-ref="cardNumber"
autocomplete="off" required="required" placeholder="Номер">
</div>

<div class="card-input">
<label for="cardName" class="card-input__label">Владелец карты</label>

<input type="text" id="cardName" name="cardName" class="card-
input__input" v-model="cardName" v-on:focus="focusInput" v-on:blur="blurInput"
data-ref="cardName" autocomplete="off" pattern="[a-zA-Z]{1,50}[' ']{1}[a-zA-
Z]{1,50}[' ']{1}[a-zA-Z]{1,50}"title="Введите латинские буквы"
placeholder="Введите ФИО" required="required">
</div>

<div class="card-form__row">
<div class="card-form__col">
<div class="card-form__group">
<label for="cardMonth" class="card-input__label">
Срок действия
</label>

<select class="card-input__input -select" id="cardMonth" v-
model="cardMonth" v-on:focus="focusInput" v-on:blur="blurInput" data-
ref="cardDate" required="required">

<option value="" disabled selected>Месяц</option>

<option v-bind:value="n < 10 ? '0' + n : n" v-for="n in 12" v-bind:disabled="n <
minCardMonth" v-bind:key="n">
{{n < 10 ? '0' + n : n}}
</option>

```

```

</select>
    <select class="card-input__input-select" id="cardYear" v-model="cardYear" v-
on:focus="focusInput"          v-on:blur="blurInput"          data-ref="cardDate"
required="required">
        <option value="" disabled selected>Год</option>
        <option v-bind:value="$index + minCardYear" v-for="(n, $index) in 12" v-
bind:key="n">
            {{ $index + minCardYear }}
        </option>
    </select>
</div>
</div>
<div class="card-form__col-cvv">
    <div class="card-input">
        <label for="cardCvv" class="card-input__label">CVV</label>
        <input type="text" class="card-input__input" id="cardCvv" v-mask="#####"
maxlength="3"          placeholder="Homep"          v-model="cardCvv"          v-
on:focus="flipCard(true)"          v-on:blur="flipCard(false)"          autocomplete="off"
required="required">
    </div>
</div>
</div>
<label for="cardCounter" class="card-input__label">Итоговый счет</label>
<div class="form-inline">
    <div class="input-group">
        <div class="input-group-prepend">
            <span class="input-group-text">Р</span>
        </div>
        <input type="number" name="sum" id="sum" class="form-control text-right"
placeholder="0" pattern="[0-9]" required="required">

```

```

<div class="input-group-append">
  <span class="input-group-text">.00</span>
</div>
</div>
</div>
<hr>
<div class="menu">
  <div class="row">
    <div class="col-md-6">
      <form action="Profile.html" method="get">
        <input  onclick="window.history.go(-1);  return  false;"  value="Отмена"
type="submit" class="submit-btn-left" ></input>
      </form>
    </div>
    </div>
    <div class="col-md-6">
      <button style="border: 0px; outline:none" type="submit" class=" submit-btn-
right">
        Оплати
      </button>
    </div>
  </div>
</div>
</div>
</div>
</div>
</div>
</div>
</div>
</div>
</div>
</div>
<!-- partial -->
<script src='https://cdnjs.cloudflare.com/ajax/libs/vue/2.6.10/vue.min.js'>
</script>
<script src='https://unpkg.com/vue-the-mask@0.11.1/dist/vue-the-mask.js'>

```

```
</script>
<script src="./buet1.js">
</script>
</body>
</html>
```

А.5 Листинг файла profile_add.php

```
<?php
$serverName = "localhost";
$username = "root";
$pass = "";
$dbName = "mybd_practice";
// Установка соединения с базой данных
$conn = mysqli_connect($serverName, $username, $pass, $dbName);
// Проверка успешности соединения
if (!$conn) {
die("Connection failed: " . mysqli_connect_error());
}
// Получение данных из формы
$Address = $_POST['Address'];
$Flat = $_POST['Flat'];
$Doorway = $_POST['Doorway'];
$Floor = $_POST['Floor'];
$Personal_account = $_POST['Personal_account'];
$Passport = $_POST['Passport'];
$Result_Counter = $_POST['Result_Counter'];
$WhenMade = $_POST['WhenMade'];
$FIO = $_POST['FIO'];
$DateView = $_POST['DateView'];
```

```

$Square = $_POST['Square'];
$Month = $_POST['Month'];
$date_pay= date("Y-m-d");
$sql10="SELECT Price from Tafiff_sheet where Tafiff_start_date = (SELECT
MAX(Tafiff_start_date) FROM Tafiff_sheet)";
$result10 = mysqli_query($conn, $sql10);
$row10 = mysqli_fetch_assoc($result10);
$Price = $row10['Price'];
$sql11="SELECT Tafiff_start_date from Tafiff_sheet where Tafiff_start_date =
(SELECT MAX(Tafiff_start_date) FROM Tafiff_sheet)";
$result11 = mysqli_query($conn, $sql11);
$row11 = mysqli_fetch_assoc($result11);
$Tafiff_start_date = $row11['Tafiff_start_date'];
$PayMonth = $Price*$Month*$Result_Counter;
// Получение следующего значения ID_Client
$sql1 = "SELECT (max(ID_Client)) as ID_Client from Client";
$result1 = mysqli_query($conn, $sql1);
$row1 = mysqli_fetch_assoc($result1);
$ID_Client = $row1['ID_Client'];
$sql2 = "SELECT (max(ID_Profile)) as ID_Profile from Profile";
$result2 = mysqli_query($conn, $sql2);
$row2 = mysqli_fetch_assoc($result2);
$ID_Profile = $row2['ID_Profile'];
$sql3 = "SELECT (max(ID_Building)+1) as ID_Building from Building";
$result3 = mysqli_query($conn, $sql3);
$row3 = mysqli_fetch_assoc($result3);
$ID_Building = $row3['ID_Building'];
$sql4 = "SELECT (max(ID_Apartament)+1) as ID_Apartament from
Apartament";
$result4 = mysqli_query($conn, $sql4);

```



```

$row4 = mysqli_fetch_assoc($result4);
$ID_Apartment = $row4['ID_Apartment'];
$sql5 = "SELECT (max(ID_Counter)+1) as ID_Counter from Counter";
$result5 = mysqli_query($conn, $sql5);
$row5 = mysqli_fetch_assoc($result5);
$ID_Counter = $row5['ID_Counter'];
$sql6 = "SELECT (max(ID_Responsible)+1) as ID_Responsible from
Responsible";
$result6 = mysqli_query($conn, $sql6);
$row6 = mysqli_fetch_assoc($result6);
$ID_Responsible = $row6['ID_Responsible'];
$sql7 = "SELECT (max(ID_Checking)+1) as ID_Checking from Checking";
$result7 = mysqli_query($conn, $sql7);
$row7 = mysqli_fetch_assoc($result7);
$ID_Checking = $row7['ID_Checking'];
$sql8 = "SELECT (max(ID_Tafiff_sheet)+1) as ID_Tafiff_sheet from
Tafiff_sheet";
$result8 = mysqli_query($conn, $sql8);
$row8 = mysqli_fetch_assoc($result8);
$ID_Tafiff_sheet = $row8['ID_Tafiff_sheet'];
$sql9 = "SELECT (max(ID_Pay)+1) as ID_Pay from Pay";
$result9 = mysqli_query($conn, $sql9);
$row9 = mysqli_fetch_assoc($result9);
$ID_Pay = $row9['ID_Pay'];
// Вставка данных в таблицу Profile
$sql1 = "INSERT INTO Building (Address, district, Material, Floors, Picture,
Owner, Doorway, Flats) VALUES ('$Address', '', '', '', '', '', '')";
$sql2 = "INSERT INTO Apartment (Flat, Floor, Flattype,
People,Doorway,Square) VALUES ('$Flat', '$Floor', '', '$Doorway', '$Square')";

```

```

    $sql3 = "UPDATE Client SET Personal_account = '$Personal_account' WHERE
ID_Client = '$ID_Client'";

    $sql4 = "UPDATE Client SET Passport = '$Passport' WHERE ID_Client =
'$ID_Client'";

    $sql5 = "INSERT INTO Counter (WhenMade, MadeIn, Result_Counter)
VALUES ('$WhenMade', 'Russia', '$Result_Counter')";

    $sql6 = "INSERT INTO Responsible (FIO) VALUES ('$FIO')";

    $sql7 = "INSERT INTO Checking (DateView, Result, ID_Responsible,
ID_Counter) VALUES ('$DateView', '1', '$ID_Responsible', '$ID_Counter')";

    $sql8 = "INSERT INTO Tafiff_sheet (Price, Tafiff_start_date) VALUES
('$Price', '$Tafiff_start_date')";

    $sql9 = "INSERT INTO Pay (date_pay, PayMonth, Month, Payment, ID_Client,
ID_Apartment, ID_Building, ID_Counter, ID_Tafiff_sheet) VALUES ('$date_pay',
'$PayMonth', '$Month', 'He оплачено', '$ID_Client', '$ID_Apartment', '$ID_Building',
'$ID_Counter', '$ID_Tafiff_sheet')";

    // Проверка успешности запроса
    if (mysqli_query($conn, $sql1)) {
        if (mysqli_query($conn, $sql2)) {
            if (mysqli_query($conn, $sql3)) {
                if (mysqli_query($conn, $sql4)) {
                    if (mysqli_query($conn, $sql5)) {
                        if (mysqli_query($conn, $sql6)) {
                            if (mysqli_query($conn, $sql7)) {
                                if (mysqli_query($conn, $sql8)) {
                                    if (mysqli_query($conn, $sql9)) {
                                        if (mysqli_query($conn, $sql10)) {
                                            if (mysqli_query($conn, $sql11)) {
                                                } else {
                                                    echo "Error: " . $sql11 . "<br>" . mysqli_error($conn);
                                                }
                                            }
                                        }
                                    }
                                }
                            }
                        }
                    }
                }
            }
        }
    }

```

```

}else {
echo "Error: " . $sql10 . "<br>" . mysqli_error($conn);
}
}else {
echo "Error: " . $sql9 . "<br>" . mysqli_error($conn);
}
}else {
echo "Error: " . $sql8 . "<br>" . mysqli_error($conn);
}
}else {
echo "Error: " . $sql7 . "<br>" . mysqli_error($conn);
}
}else {echo "Error: " . $sql6 . "<br>" . mysqli_error($conn);
}
}else {
echo "Error: " . $sql5 . "<br>" . mysqli_error($conn);
}
}else {echo "Error: " . $sql4 . "<br>" . mysqli_error($conn);
}
}else {
echo "Error: " . $sql3 . "<br>" . mysqli_error($conn);
}
}else {
echo "Error: " . $sql2 . "<br>" . mysqli_error($conn);
}
}else { echo "Error: " . $sql1 . "<br>" . mysqli_error($conn);
}
$new_url = 'table_out.php';
header('Location: '.$new_url);
// Заккрытие соединения

```

```
mysqli_close($conn);  
?>
```

А.6 Листинг файла profile_add1.php

```
<?php  
session_start();  
$serverName = "localhost";  
$username = "root";  
$pass = "";  
$dbName = "mybd_practice";  
// Установка соединения с базой данных  
$conn = mysqli_connect($serverName, $username, $pass, $dbName);  
// Проверка успешности соединения  
if (!$conn) {  
    die("Connection failed: " . mysqli_connect_error());  
}  
// Получение данных из формы  
$Result_Counter = $_POST['Result_Counter'];  
$Month = $_POST['Month'];  
$Email = $_SESSION['Email'];  
$Password = $_SESSION['Password'];  
$sql = "SELECT Client.ID_Client AS ID_Client  
From Client Inner join Profile ON Client.ID_Client = Profile.ID_Client  
Where Email = '$Email' and Password = '$Password'";  
$result = mysqli_query($conn, $sql);  
$row = mysqli_fetch_assoc($result);  
$ID_Client = $row['ID_Client'];  
$sql1 = "SELECT Counter.ID_Counter As ID_Counter  
From Counter Inner join Pay on Counter.ID_Counter = Pay.ID_Counter
```

```

Inner Join Client on Client.ID_Client = Pay.ID_Client
Where Client.ID_Client = '$ID_Client'';
$result1 = mysqli_query($conn, $sql1);
$row1 = mysqli_fetch_assoc($result1);
$ID_Counter = $row1['ID_Counter'];
$sql2 = "SELECT Pay.ID_Pay AS ID_Pay
From Pay Inner join Client on Pay.ID_Client = Client.ID_Client
Where Client.ID_Client = '$ID_Client'';
$result2 = mysqli_query($conn, $sql2);
$row2 = mysqli_fetch_assoc($result2);
$ID_Pay = $row2['ID_Pay'];
// Вставка данных в таблицу Profile
$sql3 = "UPDATE Counter SET Result_Counter = '$Result_Counter' WHERE
ID_Counter = '$ID_Counter'';
$sql4 = "UPDATE Pay SET Month = '$Month' WHERE ID_Pay = '$ID_Pay'';
$sql5 = "SELECT Tafiff_sheet.Price AS Price
From Tafiff_sheet Inner join Pay on Tafiff_sheet.ID_Tafiff_sheet =
Pay.ID_Tafiff_sheet
Inner join Client on Pay.ID_Client = Client.ID_Client
Where Client.ID_Client = '$ID_Client'';
$result5 = mysqli_query($conn, $sql5);
$row5 = mysqli_fetch_assoc($result5);
$Price = $row5['Price'];
$PayMonth = $Price*$Month*$Result_Counter;
$sql6 = "UPDATE Pay SET PayMonth = '$PayMonth' WHERE ID_Pay =
'$ID_Pay'';
// Проверка успешности запроса
if (mysqli_query($conn, $sql)) {
if (mysqli_query($conn, $sql1)) {
if (mysqli_query($conn, $sql2)) {

```

```

if (mysqli_query($conn, $sql3)) {
if (mysqli_query($conn, $sql4)) {
if (mysqli_query($conn, $sql5)) {
if (mysqli_query($conn, $sql6)) {
} else {
echo "Error: " . $sql6 . "<br>" . mysqli_error($conn);
}
} else {
echo "Error: " . $sql5 . "<br>" . mysqli_error($conn);
}
} else {
echo "Error: " . $sql4 . "<br>" . mysqli_error($conn);
}
} else {
echo "Error: " . $sql3 . "<br>" . mysqli_error($conn);
}
} else {
echo "Error: " . $sql2 . "<br>" . mysqli_error($conn);
}
} else {
echo "Error: " . $sql1 . "<br>" . mysqli_error($conn);
}
} else {
echo "Error: " . $sql . "<br>" . mysqli_error($conn);
}

$new_url = 'top1.php';
header('Location: '.$new_url);
// Заккрытие соединения
mysqli_close($conn);
?>

```

А.7 Листинг файла Profile.php

```
<!DOCTYPE html>
<html lang="ru">
<head>
<meta charset="UTF-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Document</title>
<link rel="stylesheet" href="index.css" />
<script type="text/javascript" src="index.js"></script>
</head>
<body>
<body>
<form action="profile_add.php" method="POST">
<div class="container">
<div class="left">
<div class="header">
<h2 class="animation a1">ДОБРО ПОЖАЛОВАТЬ</h2>
<h4 class="animation a2">Введите свои данные</h4>
</div>
<div class="form">
<div class="col-sm-8">
<div class="card-block">
<h6 class="animation a3">Информация о пользователе</h6>
<div class="группа animation a4">
<ul class="menu">
<li></li>
<div class="группа1">
```

```

<li><h10>ФИО</h10></li>
<?php
$serverName = "localhost";
$username = "root";
$password = "";
$dbName = "mybd_practice";
// Создание подключения
$conn = mysqli_connect($serverName, $username, $password, $dbName);
// Проверка подключения
if (!$conn) {
    die("Ошибка подключения: " . mysqli_connect_error());
}
$sql = "SELECT FIO FROM Client WHERE ID_Client = (SELECT
MAX(ID_Client) FROM Client)";
$result = mysqli_query($conn, $sql);
echo "<table>";
if (mysqli_num_rows($result) > 0) {
    while($row = mysqli_fetch_assoc($result)) {
        echo "<tr>";
        echo "<td>" . $row["FIO"] . "</td>";
        echo "</tr>";
    }
} else {
    echo "0 результатов";
}
echo "</table>";
?>
</div>
</ul>
</div>

```



```

<div class="row">
  <div class="col-sm-6">
    <div class="grypa2">
      <div class="grypa">
        <p class="text animation a5">Адрес <br>
        <input class="form-field animation a5" type="text" placeholder=" Адрес
проживания" name="Address" required="required"/></p>
      </div>
    <div class="grypa">
      <p class="text animation a5">Номер квартиры<br>
      <input class="form-field animation a5" type="text" name="Flat"
placeholder="Номер квартиры" required="required" /></p>
    </div>
  </div>
  <div class="grypa2">
    <div class="grypa">
      <p class="text animation a6">Номер подъезда<br>
      <input class="form-field animation a6" type="text" name="Doorway"
placeholder="Номер подъезда" required="required" /></p>
    </div>
    <div class="grypa">
      <p class="text animation a6">Номер этажа<br>
      <input class="form-field animation a6" type="number" name="Floor"
placeholder="Номер этажа" required="required"/></p>
    </div>
  </div>
  <div class="grypa2">
    <div class="grypa">
      <p class="text animation a7">Лицевой счет<br>

```

```

        <input class="form-field animation a7" name="Personal_account" type="text"
placeholder="Номер лицевого счета" required="required" title="Введите 20 цифр"
minlength=20 maxlength="20"/></p>
    </div>
    <div class="группа">
        <p class="text animation a7">Номер паспорта <br>
        <input
            class="form-field
            animation
            a7" type="text"
name="Passport" pattern="[0-9]+" title="Введите 6 цифр" minlength=6
maxlength="6" placeholder="Номер паспорта" required="required" /></p>
    </div>
</div>
<div class="группа2">
    <div class="группа3">
        <p class="text animation a8">Показатель счетчика<br>
        <input class="form-field animation a8" name="Result_Counter" type="text"
placeholder="Число" required="required" /></p>
    </div>
    <div class="группа">
        <p class="text animation a8">Дата изготовления счетчика<br>
        <input class="form-field animation a8" name="WhenMade" type="text"
placeholder="Дата" required="required" /></p>
    </div>
</div>
<div class="группа2">
    <div class="группа">
        <p class="text animation a9">Инициалы проверяющего<br>
        <input class="form-field animation a9" type="text" name="FIO" id="FIO"
placeholder="Инициалы проверяющего" pattern="[А-Яа-яЁё ]+" title="Только
русские буквы" maxlength="50" placeholder="Инициалы" required="required"/>
    </div>

```

```

<div class="grypa">
  <p class="text animation a9">Дата проверки счетчика<br>
  <input class="form-field animation a9" name="DateView" type="text"
placeholder="Дата" required="required" /></p>
</div>
</div>
<div class="grypa2">
  <div class="grypa3">
    <p class="text animation a10">Площадь квартиры<br>
    <input class="form-field animation a10" type="number" name="Square"
placeholder="Площадь квартиры" required="required"/></p>
  </div>
  <div class="grypa">
    <p class="text animation a10">Число неоплаченных месяцев<br>
    <input class="form-field animation a10" name="Month" type="text"
placeholder="Число месяцев" required="required" /></p>
  </div>
</div>
</div>
</div>
</div>
</div>
</div>
</div>
<h6 class="m-b-20 m-t-40 p-b-5 b-b-default f-w-900 animation a11">Оплата
через SberPay</h6>
<meta name="description" content="оплата по карте" />
<button class="btn animation a12 ">Сохранить</button>
</div>
</div>
<div class="right"></div>
</div>

```

```
</form>
</body>
</html>
```

А.8 Листинг файла Profile1.php

```
<!DOCTYPE html>
<html lang="ru">
<head>
<meta charset="UTF-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Document</title>
<link rel="stylesheet" href="index1.css" />
</head>
<?php
session_start();
$serverName = "localhost";
$username = "root";
$password = "";
$dbName = "mybd_practice";
// Создание подключения
$conn = mysqli_connect($serverName, $username, $password, $dbName);
// Проверка подключения
if (!$conn) {
die("Ошибка подключения: " . mysqli_connect_error());
}
?>
<body>
```

```

<form action="profile_add1.php" method="POST">
<meta name="description" content="оплата по карте" />
<div class="container">
<div class="left">
<div class="header">
<h2 class="animation a1">ДОБРО ПОЖАЛОВАТЬ</h2>
<h4 class="animation a2">Введите свои данные</h4>
</div>
<div class="form">
<div class="card-block">
<h6 class="animation a3">Информация о пользователе</h6>
<div class="grypa animation a4">
<ul class="menu">
<li>

</li>
<div class="grypa1">
<li>
<h10>ФИО</h10>
</li>
<?php
$Email = $_SESSION['Email'];
$Password = $_SESSION['Password'];
$sql = "SELECT Apartment.Square AS Square, Building.Address AS Address,
Client.FIO AS FIO_C, Apartment.Floor AS Floor, Apartment.Flat AS Flat,
Apartment.Doorway AS Doorway, Client.Personal_account AS Personal_account,
Client.Passport AS Passport, Counter.WhenMade AS WhenMade, Responsible.FIO
AS FIO_R, Checking.DateView AS DateView
From Client Inner join Profile ON Client.ID_Client = Profile.ID_Client

```

```

Inner join Pay on Client.ID_Client = Pay.ID_Client
Inner join Apartament on Apartament.ID_Apartament = Pay.ID_Apartament
Inner join Building on Building.ID_Building = Pay.ID_Building
Inner join Tafiff_sheet on Tafiff_sheet.ID_Tafiff_sheet = Pay.ID_Tafiff_sheet
Inner join Counter on Counter.ID_Counter = Pay.ID_Counter
Inner join Checking on Checking.ID_Counter = Counter.ID_Counter
Inner join Responsible on Checking.ID_Responsible =
Responsible.ID_Responsible

Where Email = '$Email' and Password = '$Password';
$result = mysqli_query($conn, $sql);
echo "<table>";
if (mysqli_num_rows($result) > 0) {
while($row = mysqli_fetch_assoc($result)) {
echo "<tr>";
echo "<td>" . $row["FIO_C"] . "</td>";
echo "</tr>";
}
} else {
echo "0 результатов";
}
echo "</table>";
?>
</div>
</ul>
</div>
<div class="row">
<div class="grypa2 animation a5">
<div class="grypa-1">
<p class="text ">Адрес <br></p>
<?php

```

```

$result = mysqli_query($conn, $sql);
echo "<table>";
if (mysqli_num_rows($result) > 0) {
while($row = mysqli_fetch_assoc($result)) {
echo "<tr>";
echo "<td>" . $row["Address"] . "</td>";
echo "</tr>";
}
} else {
echo "0 результатов";
}
echo "</table>";
?>
</div>
<div class="grypa-r">
<p class="text">Номер квартиры<br></p>
<?php
$result = mysqli_query($conn, $sql);
echo "<table>";
if (mysqli_num_rows($result) > 0) {
while($row = mysqli_fetch_assoc($result)) {
echo "<tr>";
echo "<td>" . $row["Flat"] . "</td>";
echo "</tr>";
}
} else {
echo "0 результатов";
}
echo "</table>";
?>

```

```

</div>
</div>
<div class="grypa2 animation a6">
<div class="grypa-l">
<p class="text">Номер подъезда<br></p>
<?php
$result = mysqli_query($conn, $sql);
echo "<table>";
if (mysqli_num_rows($result) > 0) {
while($row = mysqli_fetch_assoc($result)) {
echo "<tr>";
echo "<td>" . $row["Doorway"] . "</td>";
echo "</tr>";
}
} else {
echo "0 результатов";
}
echo "</table>";
?>
</div>
<div class="grypa-r">
<p class="text">Номер этажа<br></p>
<?php
$result = mysqli_query($conn, $sql);
echo "<table>";
if (mysqli_num_rows($result) > 0) {
while($row = mysqli_fetch_assoc($result)) {
echo "<tr>";
echo "<td>" . $row["Floor"] . "</td>";
echo "</tr>";
}
}
}

```



```

}
} else {
echo "0 результатов";
}
echo "</table>";
?>
</div>
</div>
<div class="группа2 animation a7">
<div class="группа-1">
<p class="text">Лицевой счет<br></p>
<?php
$result = mysqli_query($conn, $sql);
echo "<table>";
if (mysqli_num_rows($result) > 0) {
while($row = mysqli_fetch_assoc($result)) {
echo "<tr>";
echo "<td>" . $row["Personal_account"] . "</td>";
echo "</tr>";
}
} else {
echo "0 результатов";
}
echo "</table>";
?>
</div>
<div class="группа-r">
<p class="text">Номер паспорта <br></p>
<?php
$result = mysqli_query($conn, $sql);

```

```

echo "<table>";
if (mysqli_num_rows($result) > 0) {
while($row = mysqli_fetch_assoc($result)) {
echo "<tr>";
echo "<td>" . $row["Passport"] . "</td>";
echo "</tr>";
}
} else {
echo "0 результатов";
}
echo "</table>";
?>
</div>
</div>
<div class="grypa2 animation a8">
<div class="grypa-l">
<p class="text">Показ. счетчика<br></p>
<input style="margin: 0px 0px 0px 0px" class="form-field animation a12"
name="Result_Counter"type="text" placeholder="Число" required="required" />
</div>
<div class="grypa-r">
<p class="text">Д. И. счетчика<br></p>
<?php
$result = mysqli_query($conn, $sql);
echo "<table>";
if (mysqli_num_rows($result) > 0) {
while($row = mysqli_fetch_assoc($result)) {
echo "<tr>";
echo "<td>" . $row["WhenMade"] . "</td>";
echo "</tr>";

```

```

}
} else {
echo "0 результатов";
}
echo "</table>";
?>
</div>
</div>
<div class="grypa2 animation a9">
<div class="grypa-l">
<p class="text">ФИО проверяющего<br></p>
<?php
$result = mysqli_query($conn, $sql);
echo "<table>";
if (mysqli_num_rows($result) > 0) {
while($row = mysqli_fetch_assoc($result)) {
echo "<tr>";
echo "<td>" . $row["FIO_R"] . "</td>";
echo "</tr>";
}
} else {
echo "0 результатов";
}
echo "</table>";
?>
</div>
<div class="grypa-r">
<p class="text">Д. П. счетчика<br></p>
<?php
$result = mysqli_query($conn, $sql);

```

```

echo "<table>";
if (mysqli_num_rows($result) > 0) {
while($row = mysqli_fetch_assoc($result)) {
echo "<tr>";
echo "<td>" . $row["DateView"] . "</td>";
echo "</tr>";
}
} else {
echo "0 результатов";
}
echo "</table>";
?>
</div>
</div>
<div class="grypa2 animation a10">
<div class="grypa-1">
<p class="text">Площадь квартиры<br></p>
<?php
$result = mysqli_query($conn, $sql);
echo "<table>";
if (mysqli_num_rows($result) > 0) {
while($row = mysqli_fetch_assoc($result)) {
echo "<tr>";
echo "<td>" . $row["Square"] . "</td>";
echo "</tr>";
}
} else {
echo "0 результатов";
}
echo "</table>";

```

```

?>
</div>
<div class="grypa-r">
  <p class="text">Ч. Н. месяцев<br>
    <input class="form-field" name="Month" type="text" placeholder="Число
месяцев" required="required" /></p>
  </div>
</div>
</div>
</div>
</div>
<h6 class="m-b-20 m-t-40 p-b-5 b-b-default f-w-900 animation a11">Оплата
через SberPay</h6>
  <button class="btn animation a12">Сохранить</button>
</div>
</div>
<div class="right"></div>
</div>
</form>
</body>
</html>

```

А.9 Листинг файла Receipt.php

```

<?php
session_start();
header("Content-type: application/vnd.ms_word; format=attachment;");
header("Content-Disposition: attachment ; Filename = Receipt.doc");
header("Content-Transfer-Encoding: binary");
echo '<meta http-equiv=Content-Type content="text/html; charset=UTF-8">';
?>

```

```

<html>
<style>
    .imgblock {
        position: relative;
        display: inline-block;
    }

    .imgblock img {
        height: 160px;
        width: 250px;
    }

    .imgblock span {
        /*background: rgba(0,0,0,0.7); (полупрозрачность)*/
        background: #222;
        color: #fff;
        border-radius: 2px;
        position: absolute;
        right: 0;
        bottom: 10px;
        font-size: 18px;
        padding: 3px 5px;
    }
</style>
<body>
<h1>
<HTML>
<HEAD>
<TITLE>Создание формы</TITLE>
</HEAD>
<style>

```

```

table {
border: 1px dashed black;
border-spacing: 10px;
width: 100%;
height:30%;}
#user{
width:100%;
}
#use{
width:73%;
}
</style>
<BODY BGCOLOR="White" TEXT="black">
<FORM>
<TABLE>
<TR>
<TD ALIGN="CENTER">
    <div id="logo">
        
    </div>
    </TD>
</TR>

<hr align="Center" width="100%" size="1" color="black">
<TR></TR>
<TR></TR>
<TABLE>
<TR>
<TD ALIGN="left" width="30%"> Извещение </TD>

```

```

<TD>
</TD>
<TD ALIGN="right" COLSPAN="2"> <i> Форма ПД №4 </i></TD>
</TR>
<TD ALIGN="left" width="30%">
<?php
$today = date("F j, Y");
echo ("$today");
?>
</TD>

<TR>
<TD>
Ф.И.О. плательщика
</TD>
<TD>
</TD>
<TD ALIGN="right" COLSPAN="2" WIDTH="25%">
<?php
$cardName=$_POST['cardName'];
echo("$cardName");
?>

<HR ALIGN="right" WIDTH="25%" SIZE="1" COLOR="black">
</TD>
</TR>

<TR>
<TD>Номер карты</TD>

```



```

<TD>
</TD>
<TD ALIGN="right" COLSPAN="2" WIDTH="25%">
<?php
$cardNumber = $_POST['cardNumber'];
echo("$cardNumber");
?>
<HR ALIGN="right" WIDTH="25%" SIZE="1" COLOR="black">
</TD>
</TR>

```

```

<TR>
<TD>
Сумма к оплате
</TD>
<TD>
</TD>
<TD ALIGN="right" COLSPAN="2" WIDTH="25%">
<?php
$sum = $_POST['sum'];
echo("$sum");
?>
руб.
<HR ALIGN="right" WIDTH="25%" SIZE="1" COLOR="black">
</TD>
</TR>
</TABLE>
<TR></TR>
<TR></TR>

```

```
<hr align="Center" width="100%" size="1" color="black">
```

```
<TR>
```

```
<p>Печать:</p>
```

```
<TABLE>
```

```
<p></p>
```

```
</TABLE>
```

```
</TR>
```

```
<?php
```

```
$serverName = "localhost";
```

```
$username = "root";
```

```
$pass = "";
```

```
$dbName = "mybd_practice";
```

```
// Установка соединения с базой данных
```

```
$conn = mysqli_connect($serverName, $username, $pass, $dbName);
```

```
// Проверка успешности соединения
```

```
if (!$conn) {
```

```
    die("Connection failed: " . mysqli_connect_error());
```

```
}
```

```
$sum=$_POST['sum'];
```

```
$Email = $_SESSION['Email'];
```

```
$Password = $_SESSION['Password'];
```

```
$sql = "SELECT Client.ID_Client AS ID_Client
```

```
From Client Inner join Profile ON Client.ID_Client = Profile.ID_Client
```

```
Where Email = '$Email' and Password = '$Password'";
```

```
$result = mysqli_query($conn, $sql);  
$row = mysqli_fetch_assoc($result);  
$ID_Client = $row['ID_Client'];
```

```
$sql2 = "SELECT Pay.PayMonth AS PayMonth  
From Pay Inner join Client ON Pay.ID_Client = Client.ID_Client  
Where Client.ID_Client = '$ID_Client'";  
$result2 = mysqli_query($conn, $sql2);  
$row2 = mysqli_fetch_assoc($result2);  
$PayMonth = $row2['PayMonth'];
```

```
$sql3 = "SELECT Pay.ID_Pay AS ID_Pay  
From Pay Inner join Client ON Pay.ID_Client = Client.ID_Client  
Where Client.ID_Client = '$ID_Client'";  
$result3 = mysqli_query($conn, $sql3);  
$row3 = mysqli_fetch_assoc($result3);  
$ID_Pay = $row3['ID_Pay'];  
$Payment = 'Оплачено';  
$Result = ($PayMonth-$sum);  
if ($Result<= 0) {  
    $sql4 = "UPDATE Pay SET Payment = '$Payment' WHERE ID_Pay =  
'$ID_Pay'";  
    $sql5 = "UPDATE Pay SET PayMonth = '$Result' WHERE ID_Pay =  
'$ID_Pay'";  
} else{  
    $sql6 = "UPDATE Pay SET Payment = 'Не оплачено' WHERE ID_Pay =  
'$ID_Pay'";  
    $sql7 = "UPDATE Pay SET PayMonth = ('$PayMonth-$sum') WHERE  
ID_Pay = '$ID_Pay'";
```

```
}
```

```
if (mysqli_query($conn, $sql)) {  
    if (mysqli_query($conn, $sql2)) {  
        if (mysqli_query($conn, $sql3)) {  
            if (mysqli_query($conn, $sql4)) {  
                if (mysqli_query($conn, $sql5)) {  
                    }else {  
                        echo "Error: " . $sql5 . "<br>" . mysqli_error($conn);  
                    }  
                }else {  
                    echo "Error: " . $sql4 . "<br>" . mysqli_error($conn);  
                }  
            }else {  
                echo "Error: " . $sql3 . "<br>" . mysqli_error($conn);  
            }  
        }else {  
            echo "Error: " . $sql2 . "<br>" . mysqli_error($conn);  
        }  
    }else {  
        echo "Error: " . $sql . "<br>" . mysqli_error($conn);  
    }  
}
```

```
?>
```

```
<TR>
```

```
<TD>
```

```
</TD>
```

```
<TD>
```

</TD>

</TR>

</TABLE>

</FORM>

</BODY>

</HTML>

</div>

</h1>

</h1>

</body>

</html>

A.10 Листинг файла table_out.php

<!DOCTYPE html>

<html lang="ru" >

<head>

<script

src="https://cdnjs.cloudflare.com/ajax/libs/jquery/2.0.2/jquery.min.js"></script>

<meta charset="UTF-8">

<title>база данных</title>

<link rel="stylesheet" href="/top.css">

</head>

<body>

<!-- partial:index.partial.html -->

<html>

<head>

```

<script
src="https://cdnjs.cloudflare.com/ajax/libs/jquery/2.0.2/jquery.min.js"></script>
</head>
<body>
<div id="gradient"></div>
</body>
</html>
<!-- partial -->
<script src="./top.js"></script>
<div class="form">
<div class="table">
<?php
$serverName = "localhost";
$username = "root";
$pass = "";
$dbName = "mybd_practice";
// Создание подключения
$conn = mysqli_connect($serverName, $username, $pass, $dbName);
// Проверка подключения
if (!$conn) {
die("Ошибка подключения: " . mysqli_connect_error());
}
// Получение следующего значения ID
$sql1 = "SELECT (max(ID_Client)) as ID_Client from Client";
$result1 = mysqli_query($conn, $sql1);
$row1 = mysqli_fetch_assoc($result1);
$ID_Client = $row1['ID_Client'];
$sql2 = "SELECT (max(ID_Profile)) as ID_Profile from Profile";
$result2 = mysqli_query($conn, $sql2);
$row2 = mysqli_fetch_assoc($result2);

```

```

$ID_Profile = $row2['ID_Profile'];
$sql3 = "SELECT (max(ID_Building)) as ID_Building from Building";
$result3 = mysqli_query($conn, $sql3);
$row3 = mysqli_fetch_assoc($result3);
$ID_Building = $row3['ID_Building'];
$sql4 = "SELECT (max(ID_Apartament)) as ID_Apartament from Apartament";
$result4 = mysqli_query($conn, $sql4);
$row4 = mysqli_fetch_assoc($result4);
$ID_Apartament = $row4['ID_Apartament'];
$sql5 = "SELECT (max(ID_Counter)) as ID_Counter from Counter";
$result5 = mysqli_query($conn, $sql5);
$row5 = mysqli_fetch_assoc($result5);
$ID_Counter = $row5['ID_Counter'];
$sql6 = "SELECT (max(ID_Responsable)) as ID_Responsable from
Responsable";
$result6 = mysqli_query($conn, $sql6);
$row6 = mysqli_fetch_assoc($result6);
$ID_Responsable = $row6['ID_Responsable'];
$sql7 = "SELECT (max(ID_Checking)) as ID_Checking from Checking";
$result7 = mysqli_query($conn, $sql7);
$row7 = mysqli_fetch_assoc($result7);
$ID_Checking = $row7['ID_Checking'];
$sql8 = "SELECT (max(ID_Tafiff_sheet)) as ID_Tafiff_sheet from
Tafiff_sheet";
$result8 = mysqli_query($conn, $sql8);
$row8 = mysqli_fetch_assoc($result8);
$ID_Tafiff_sheet = $row8['ID_Tafiff_sheet'];
$sql9 = "SELECT (max(ID_Pay)) as ID_Pay from Pay";
$result9 = mysqli_query($conn, $sql9);
$row9 = mysqli_fetch_assoc($result9);

```

```

$ID_Pay = $row9['ID_Pay'];
// Получение данных из формы
$sql10 = "SELECT Address as Address from Building Where ID_Building =
'$ID_Building'";
$result10 = mysqli_query($conn, $sql10);
$row10 = mysqli_fetch_assoc($result10);
$Address = $row10['Address'];
$sql11 = "SELECT FIO as FIO from Client Where ID_Client = '$ID_Client'";
$result11 = mysqli_query($conn, $sql11);
$row11 = mysqli_fetch_assoc($result11);
$FIO = $row11['FIO'];
$sql12 = "SELECT Floor as Floor from Apartament Where ID_Apartament =
'$ID_Apartament'";
$result12 = mysqli_query($conn, $sql12);
$row12 = mysqli_fetch_assoc($result12);
$Floor = $row12['Floor'];
$sql13 = "SELECT Flat as Flat from Apartament Where ID_Apartament =
'$ID_Apartament'";
$result13 = mysqli_query($conn, $sql13);
$row13 = mysqli_fetch_assoc($result13);
$Flat = $row13['Flat'];
$sql14 = "SELECT Doorway as Doorway from Apartament Where
ID_Apartament = '$ID_Apartament'";
$result14 = mysqli_query($conn, $sql14);
$row14 = mysqli_fetch_assoc($result14);
$Doorway = $row14['Doorway'];
$sql15 = "SELECT Passport as Passport from Client Where ID_Client =
'$ID_Client'";
$result15 = mysqli_query($conn, $sql15);
$row15 = mysqli_fetch_assoc($result15);

```



```

$Passport = $row15['Passport'];

$sql16 = "SELECT Personal_account as Personal_account from Client Where
ID_Client = '$ID_Client'";

$result16 = mysqli_query($conn, $sql16);
$row16 = mysqli_fetch_assoc($result16);
$Personal_account = $row16['Personal_account'];

$sql17 = "SELECT Month as Month from Pay Where ID_Pay = '$ID_Pay'";
$result17 = mysqli_query($conn, $sql17);
$row17 = mysqli_fetch_assoc($result17);
$Month = $row17['Month'];

$sql18 = "SELECT Result_Counter as Result_Counter from Counter Where
ID_Counter = '$ID_Counter'";

$result18 = mysqli_query($conn, $sql18);
$row18 = mysqli_fetch_assoc($result18);
$Result_Counter = $row18['Result_Counter'];

$sql19 = "SELECT Price as Price from Tafiff_sheet Where ID_Tafiff_sheet =
'$ID_Tafiff_sheet'";

$result19 = mysqli_query($conn, $sql19);
$row19 = mysqli_fetch_assoc($result19);
$Price = $row19['Price'];

$PayMonth = $Price*$Month*$Result_Counter;

$sql = "SELECT Building.Address AS Address, Client.FIO AS FIO,
Apartament.Floor AS Floor, Apartament.Flat AS Flat, Apartament.Doorway AS
Doorway, Client.Personal_account AS Personal_account, Client.Passport AS Passport,
Pay.Month AS Month, Counter.Result_Counter As Result_Counter, Pay.PayMonth AS
PayMonth, Tafiff_sheet.Price AS Price, Pay.Payment AS Payment

From Client Inner join Profile ON Client.ID_Client = Profile.ID_Client
Inner join Pay on Client.ID_Client = Pay.ID_Client
Inner join Apartament on Apartament.ID_Apartament = Pay.ID_Apartament
Inner join Building on Building.ID_Building = Pay.ID_Building

```

```

Inner join Tafiff_sheet on Tafiff_sheet.ID_Tafiff_sheet = Pay.ID_Tafiff_sheet
Inner Join Counter on Counter.ID_Counter = Pay.ID_Counter
Where Client.ID_Client = ' $ID_Client'';
$result = mysqli_query($conn, $sql);
// Вывод таблицы
echo "<table>";
echo
    "<tr><th>Адрес</th><th>Инициалы</th><th>Номер
этажа</th><th>Номер  квартиры</th><th>Номер  подъезда</th><th>Серия  и
номер  паспорта</th><th>Номер  лицевого  счета</th><th>Количество
месяцев</th><th>Итоговая  число</th><th>Тарифная  цена</th><th>Оплата  за
месяц</th><th>Итог</th></tr>";
// Проверка успешности запроса
if (mysqli_query($conn, $sql1)) {
if (mysqli_query($conn, $sql2)) {
if (mysqli_query($conn, $sql3)) {
if (mysqli_query($conn, $sql4)) {
if (mysqli_query($conn, $sql5)) {
if (mysqli_query($conn, $sql6)) {
if (mysqli_query($conn, $sql7)) {
if (mysqli_query($conn, $sql8)) {
if (mysqli_query($conn, $sql9)) {
if (mysqli_query($conn, $sql10)) {
if (mysqli_query($conn, $sql11)) {
if (mysqli_query($conn, $sql12)) {
if (mysqli_query($conn, $sql13)) {
if (mysqli_query($conn, $sql14)) {
if (mysqli_query($conn, $sql15)) {
if (mysqli_query($conn, $sql16)) {
if (mysqli_query($conn, $sql17)) {
if (mysqli_query($conn, $sql18)) {

```

```

if (mysqli_query($conn, $sql19)) {
if (mysqli_query($conn, $sql)) {
} else {
echo "Error: " . $sql . "<br>" . mysqli_error($conn);
}
} else {
echo "Error: " . $sql19 . "<br>" . mysqli_error($conn);
}
} else {
echo "Error: " . $sql18 . "<br>" . mysqli_error($conn);
}
} else {
echo "Error: " . $sql17 . "<br>" . mysqli_error($conn);
}
} else {
echo "Error: " . $sql16 . "<br>" . mysqli_error($conn);
}
} else {
echo "Error: " . $sql15 . "<br>" . mysqli_error($conn);
}
} else {
echo "Error: " . $sql14 . "<br>" . mysqli_error($conn);
}
} else {
echo "Error: " . $sql13 . "<br>" . mysqli_error($conn);
}
} else {
echo "Error: " . $sql12 . "<br>" . mysqli_error($conn);
}
} else {

```

```

echo "Error: " . $sql11 . "<br>" . mysqli_error($conn);
}
}else {
echo "Error: " . $sql10 . "<br>" . mysqli_error($conn);
}
}else {
echo "Error: " . $sql9 . "<br>" . mysqli_error($conn);
}
}else {
echo "Error: " . $sql8 . "<br>" . mysqli_error($conn);
}
}else {
echo "Error: " . $sql7 . "<br>" . mysqli_error($conn);
}
}else {
echo "Error: " . $sql6 . "<br>" . mysqli_error($conn);
}
}else {
echo "Error: " . $sql5 . "<br>" . mysqli_error($conn);
}
}else {
echo "Error: " . $sql4 . "<br>" . mysqli_error($conn);
}
}else {
echo "Error: " . $sql3 . "<br>" . mysqli_error($conn);
}
}else {
echo "Error: " . $sql2 . "<br>" . mysqli_error($conn);
}
}else {

```

```

echo "Error: " . $sql1 . "<br>" . mysqli_error($conn);
}
if (mysqli_num_rows($result) > 0) {
while($row = mysqli_fetch_assoc($result)) {
echo "<tr>";
echo "<td>" . $row["Address"] . "</td>";
echo "<td>" . $row["FIO"] . "</td>";
echo "<td>" . $row["Floor"] . "</td>";
echo "<td>" . $row["Flat"] . "</td>";
echo "<td>" . $row["Doorway"] . "</td>";
echo "<td>" . $row["Passport"] . "</td>";
echo "<td>" . $row["Personal_account"] . "</td>";
echo "<td>" . $row["Month"] . "</td>";
echo "<td>" . $row["Result_Counter"] . "</td>";
echo "<td>" . $row["Price"] . "</td>";
echo "<td>" . $row["PayMonth"] . "</td>";
echo "<td>" . $row["Payment"] . "</td>";
echo "</tr>";
} else {
echo "0 результатов";
}
echo "</table>";
//$new_url = 'выводТабл2.php';
//header('Location: '.$new_url);
// Закрытие соединения
mysqli_close($conn);
?>
</div>
<div class="Button">
<form action="Map.php" method="POST">

```

```

<button class="btn" type="submit" >К оплате</button>
</form>
</div>
</div>
</body>
</html>

```

А.11 Листинг файла top1.php

```

<!DOCTYPE html>
<html lang="ru" >
<head>
<script
src="https://cdnjs.cloudflare.com/ajax/libs/jquery/2.0.2/jquery.min.js"></script>
<meta charset="UTF-8">
<title>база данных</title>
<link rel="stylesheet" href="/top.css">
</head>
<body>
<!-- partial:index.partial.html -->
<html>
<head>
<script
src="https://cdnjs.cloudflare.com/ajax/libs/jquery/2.0.2/jquery.min.js"></script>
</head>
<body id="gradient">
</body>
</html>
<!-- partial -->
<script src="/top.js"></script>

```

```

<div class="form">
    <div class="table">
<?php
session_start();

$serverName = "localhost";
$username = "root";
$pass = "";
$dbName = "mybd_practice";
// Установка соединения с базой данных
$conn = mysqli_connect($serverName, $username, $pass, $dbName);

// Retrieve data from the "Profile" table

$email = $_SESSION['Email'];
$password = $_SESSION['Password'];

$sql = "SELECT Building.Address AS Address, Client.FIO AS FIO,
Apartament.Floor AS Floor, Apartament.Flat AS Flat, Apartament.Doorway AS
Doorway, Client.Personal_account AS Personal_account, Client.Passport AS Passport,
Pay.Month AS Month, Counter.Result_Counter As Result_Counter, Pay.PayMonth AS
PayMonth, Tafiff_sheet.Price AS Price, Pay.Payment AS Payment
From Client Inner join Profile ON Client.ID_Client = Profile.ID_Client
Inner join Pay on Client.ID_Client = Pay.ID_Client
Inner join Apartament on Apartament.ID_Apartament = Pay.ID_Apartament
Inner join Building on Building.ID_Building = Pay.ID_Building
Inner join Tafiff_sheet on Tafiff_sheet.ID_Tafiff_sheet = Pay.ID_Tafiff_sheet
Inner Join Counter on Counter.ID_Counter = Pay.ID_Counter
Where Email = '$email' and Password = '$password'";
$result = mysqli_query($conn, $sql);

```

```

echo "<table>";

echo "<tr><th>Адрес</th><th>ФИО</th><th>Номер этажа</th><th>Номер
квартиры</th><th>Номер подъезда</th><th>Серия и номер
паспорта</th><th>Номер лицевого счета</th><th>Количество
месяцев</th><th>Итоговая число</th><th>Тарифная цена</th><th>Оплата за
месяц</th><th>Итог</th></tr>";

// Display the data in a table
if (mysqli_num_rows($result) > 0) {
    while($row = mysqli_fetch_assoc($result)) {
        unset($_SESSION['message']);
        echo "<tr>";
        echo "<td>" . $row["Address"] . "</td>";
        echo "<td>" . $row["FIO"] . "</td>";
        echo "<td>" . $row["Floor"] . "</td>";
        echo "<td>" . $row["Flat"] . "</td>";
        echo "<td>" . $row["Doorway"] . "</td>";
        echo "<td>" . $row["Passport"] . "</td>";
        echo "<td>" . $row["Personal_account"] . "</td>";
        echo "<td>" . $row["Month"] . "</td>";
        echo "<td>" . $row["Result_Counter"] . "</td>";
        echo "<td>" . $row["Price"] . "</td>";
        echo "<td>" . $row["PayMonth"] . "</td>";
        echo "<td>" . $row["Payment"] . "</td>";
        echo "</tr>";
    }
} else {}

mysqli_close($conn);

echo "</table>";

```



```

?>
</div>
<div class="Button">
<form action="Map.php" method="POST">
<button class="btn" type="submit" >К оплате</button>
</form>
<form action="Profile1.php" method="POST">
<button style="margin-left: 20px" class="btn" type="submit" >Изменить
данные</button>
</form>
</div>
</div>
</body>
</html>

```

А.12 Листинг файла buet.css

```

@import
url("https://fonts.googleapis.com/css?family=Source+Code+Pro:400,500,600,700|Source+Sans+Pro:400,600,700&display=swap");

body {
    background: #ddeefc;
    font-family: "Source Sans Pro", sans-serif;
    font-size: 16px;
}

* {
    box-sizing: border-box;
}

*:focus {
    outline: none;
}

```

```

}
.Nazv{
    vertical-align: middle;
    font-size: 18px;
    font-weight: bold;
    display: flex;
    align-items: center;
    justify-content: center;
    font-size: 14px;
    margin-bottom: 5px;
    color: #1a3b5d;
}
.form-group input:focus{
border: 0;
outline: 0;
    margin-bottom: 0%;
}
input{
border-bottom: 1.5px solid #E8E5D2!important;
font-weight: bold;
border-radius: 0;
border: 0;

}
.menu{
    display: flex;
    align-items: center;
    text-transform: uppercase;
    text-decoration: none;
    margin-right: auto;
    margin-top: 30px;
    letter-spacing: 0.05em;

```

```

        justify-content: space-between;
    }
    .form-group input:focus{
border: 0;
outline: 0;
        margin-bottom: 0%;
    }
    .submit-btn-right {
background-color: #05b0ff;
color: rgba(256,256,256,0.7);
border:0;
border-radius: 15px;
display: block;
margin: 10px 25px;
padding: 15px 60px;
font-size: 15px;
font-weight: bold;
cursor: pointer;
opacity: 1;
visibility: visible;
-webkit-transition: all .3s ease;
        outline: none;

    &:hover {
            transition: all .3s ease;
            background-color: rgba(0,0,0,0.8);
            outline: none;
        }
    }
    .submit-btn-left {

background-color: #05b0ff;

```

```
color: rgba(256,256,256,0.7);
border:0;
border-radius: 15px;
display: block;
margin: 10px 25px;
padding: 15px 60px;
font-size: 15px;
font-weight: bold;
cursor: pointer;
opacity: 1;
visibility: visible;
-webkit-transition: all .3s ease;

&:hover {
    transition: all .3s ease;
    background-color: rgba(0,0,0,0.8);
}
}
```

```
.wrapper {
    min-height: 100vh;
    display: flex;
    padding: 50px 15px;
}

@media screen and (max-width: 700px), (max-height: 500px) {
    .wrapper {
        flex-wrap: wrap;
        flex-direction: column;
    }
}
```

```

    }
}

.card-form {
    max-width: 570px;
    margin: auto;
    width: 100%;
}

@media screen and (max-width: 576px) {
    .card-form {
        margin: 0 auto;
    }
}

.card-form__inner {
    background: #fff;
    box-shadow: 0 30px 60px 0 rgba(0, 80, 177, 0.4);
    border-radius: 10px;
    padding: 35px;
    padding-top: 180px;
}

@media screen and (max-width: 480px) {
    .card-form__inner {
        padding: 25px;
        padding-top: 165px;
    }
}

@media screen and (max-width: 360px) {
    .card-form__inner {
        padding: 15px;
        padding-top: 165px;
    }
}

```

```

.card-form__row {
  display: flex;
  align-items: flex-start;
}

@media screen and (max-width: 480px) {
  .card-form__row {
    flex-wrap: wrap;
  }
}

.card-form__col {
  flex: auto;
  margin-right: 35px;
}

.card-form__col:last-child {
  margin-right: 0;
}

@media screen and (max-width: 480px) {
  .card-form__col {
    margin-right: 0;
    flex: unset;
    width: 100%;
    margin-bottom: 20px;
  }

  .card-form__col:last-child {
    margin-bottom: 0;
  }
}

.card-form__col.-cvv {
  max-width: 150px;
}

@media screen and (max-width: 480px) {
  .card-form__col.-cvv {

```

```

        max-width: initial;
    }
}

.card-form__group {
    display: flex;
    align-items: flex-start;
    flex-wrap: wrap;
}

.card-form__group .card-input__input {
    flex: 1;
    margin-right: 15px;
}

.card-form__group .card-input__input:last-child {
    margin-right: 0;
}

.card-form__button {
    width: 100%;
    height: 55px;
    background: #2364d2;
    border: none;
    border-radius: 5px;
    font-size: 22px;
    font-weight: 500;
    font-family: "Source Sans Pro", sans-serif;
    box-shadow: 3px 10px 20px 0px rgba(35, 100, 210, 0.3);
    color: #fff;
    margin-top: 20px;
    cursor: pointer;
}

@media screen and (max-width: 480px) {
    .card-form__button {
        margin-top: 10px;
    }
}

```

```
}  
}
```

```
.card-item {  
    max-width: 430px;  
    height: 270px;  
    margin-left: auto;  
    margin-right: auto;  
    position: relative;  
    z-index: 2;  
    width: 100%;  
}
```

```
@media screen and (max-width: 480px) {  
    .card-item {  
        max-width: 310px;  
        height: 220px;  
        width: 90%;  
    }  
}
```

```
@media screen and (max-width: 360px) {  
    .card-item {  
        height: 180px;  
    }  
}
```

```
.card-item.-active .card-item__side.-front {  
    transform: perspective(1000px) rotateY(180deg) rotateX(0deg) rotateZ(0deg);  
}
```

```
.card-item.-active .card-item__side.-back {  
    transform: perspective(1000px) rotateY(0) rotateX(0deg) rotateZ(0deg);  
}
```

```
.card-item__focus {  
    position: absolute;
```



```

z-index: 3;
border-radius: 5px;
left: 0;
top: 0;
width: 100%;
height: 100%;
transition: all 0.35s cubic-bezier(0.71, 0.03, 0.56, 0.85);
opacity: 0;
pointer-events: none;
overflow: hidden;
border: 2px solid rgba(255, 255, 255, 0.65);
}

.card-item__focus:after {
  content: "";
  position: absolute;
  top: 0;
  left: 0;
  width: 100%;
  background: #08142f;
  height: 100%;
  border-radius: 5px;
  filter: blur(25px);
  opacity: 0.5;
}

.card-item__focus.-active {
  opacity: 1;
}

.card-item__side {
  border-radius: 15px;
  overflow: hidden;
  box-shadow: 0 20px 60px 0 rgba(14, 42, 90, 0.55);
  transform: perspective(2000px) rotateY(0deg) rotateX(0deg) rotate(0deg);

```

```

transform-style: preserve-3d;
transition: all 0.8s cubic-bezier(0.71, 0.03, 0.56, 0.85);
backface-visibility: hidden;
height: 100%;
}
.card-item__side.-back {
position: absolute;
top: 0;
left: 0;
width: 100%;
transform: perspective(2000px) rotateY(-180deg) rotateX(0deg) rotate(0deg);
z-index: 2;
padding: 0;
height: 100%;
}
.card-item__side.-back .card-item__cover {
transform: rotateY(-180deg);
}
.card-item__bg {
max-width: 100%;
display: block;
max-height: 100%;
height: 100%;
width: 100%;
object-fit: cover;
}
.card-item__cover {
height: 100%;
background-color: #1c1d27;
position: absolute;
height: 100%;
background-color: #1c1d27;

```

```

left: 0;
top: 0;
width: 100%;
border-radius: 15px;
overflow: hidden;
}
.card-item__cover:after {
  content: "";
  position: absolute;
  left: 0;
  top: 0;
  width: 100%;
  height: 100%;
  background: rgba(6, 2, 29, 0.45);
}
.card-item__top {
  display: flex;
  align-items: flex-start;
  justify-content: space-between;
  margin-bottom: 40px;
  padding: 0 10px;
}
@media screen and (max-width: 480px) {
  .card-item__top {
    margin-bottom: 25px;
  }
}
@media screen and (max-width: 360px) {
  .card-item__top {
    margin-bottom: 15px;
  }
}

```

```

.card-item__chip {
  width: 60px;
}

@media screen and (max-width: 480px) {
  .card-item__chip {
    width: 50px;
  }
}

@media screen and (max-width: 360px) {
  .card-item__chip {
    width: 40px;
  }
}

.card-item__type {
  height: 45px;
  position: relative;
  display: flex;
  justify-content: flex-end;
  max-width: 100px;
  margin-left: auto;
  width: 100%;
}

@media screen and (max-width: 480px) {
  .card-item__type {
    height: 40px;
    max-width: 90px;
  }
}

@media screen and (max-width: 360px) {
  .card-item__type {
    height: 30px;
  }
}

```

```

}
.card-item__typeImg {
  max-width: 100%;
  object-fit: contain;
  max-height: 100%;
  object-position: top right;
}
.card-item__info {
  color: #fff;
  width: 100%;
  max-width: calc(100% - 85px);
  padding: 10px 15px;
  font-weight: 500;
  display: block;
  cursor: pointer;
}
@media screen and (max-width: 480px) {
  .card-item__info {
    padding: 10px;
  }
}
.card-item__holder {
  opacity: 0.7;
  font-size: 13px;
  margin-bottom: 6px;
}
@media screen and (max-width: 480px) {
  .card-item__holder {
    font-size: 12px;
    margin-bottom: 5px;
  }
}

```

```

.card-item__wrapper {
  font-family: "Source Code Pro", monospace;
  padding: 25px 15px;
  position: relative;
  z-index: 4;
  height: 100%;
  text-shadow: 7px 6px 10px rgba(14, 42, 90, 0.8);
  user-select: none;
}

@media screen and (max-width: 480px) {
  .card-item__wrapper {
    padding: 20px 10px;
  }
}

.card-item__name {
  font-size: 18px;
  line-height: 1;
  white-space: nowrap;
  max-width: 100%;
  overflow: hidden;
  text-overflow: ellipsis;
  text-transform: uppercase;
}

@media screen and (max-width: 480px) {
  .card-item__name {
    font-size: 16px;
  }
}

.card-item__nameItem {
  display: inline-block;
  min-width: 8px;
  position: relative;

```

```

}
.card-item__number {
  font-weight: 500;
  line-height: 1;
  color: #fff;
  font-size: 27px;
  margin-bottom: 35px;
  display: inline-block;
  padding: 10px 15px;
  cursor: pointer;
}
@media screen and (max-width: 480px) {
  .card-item__number {
    font-size: 21px;
    margin-bottom: 15px;
    padding: 10px 10px;
  }
}
@media screen and (max-width: 360px) {
  .card-item__number {
    font-size: 19px;
    margin-bottom: 10px;
    padding: 10px 10px;
  }
}
.card-item__numberItem {
  width: 16px;
  display: inline-block;
}
.card-item__numberItem.-active {
  width: 30px;
}

```

```

@media screen and (max-width: 480px) {
  .card-item__numberItem {
    width: 13px;
  }
  .card-item__numberItem.-active {
    width: 16px;
  }
}

@media screen and (max-width: 360px) {
  .card-item__numberItem {
    width: 12px;
  }
  .card-item__numberItem.-active {
    width: 8px;
  }
}

.card-item__content {
  color: #fff;
  display: flex;
  align-items: flex-start;
}

.card-item__date {
  flex-wrap: wrap;
  font-size: 18px;
  margin-left: auto;
  padding: 10px;
  display: inline-flex;
  width: 80px;
  white-space: nowrap;
  flex-shrink: 0;
  cursor: pointer;
}

```



```

@media screen and (max-width: 480px) {
  .card-item__date {
    font-size: 16px;
  }
}

.card-item__dateItem {
  position: relative;
}

.card-item__dateItem span {
  width: 22px;
  display: inline-block;
}

.card-item__dateTitle {
  opacity: 0.7;
  font-size: 13px;
  padding-bottom: 6px;
  width: 100%;
}

@media screen and (max-width: 480px) {
  .card-item__dateTitle {
    font-size: 12px;
    padding-bottom: 5px;
  }
}

.card-item__band {
  background: rgba(0, 0, 19, 0.8);
  width: 100%;
  height: 50px;
  margin-top: 30px;
  position: relative;
  z-index: 2;
}

```

```

@media screen and (max-width: 480px) {
  .card-item__band {
    margin-top: 20px;
  }
}

@media screen and (max-width: 360px) {
  .card-item__band {
    height: 40px;
    margin-top: 10px;
  }
}

.card-item__cvv {
  text-align: right;
  position: relative;
  z-index: 2;
  padding: 15px;
}

.card-item__cvv .card-item__type {
  opacity: 0.7;
}

@media screen and (max-width: 360px) {
  .card-item__cvv {
    padding: 10px 15px;
  }
}

.card-item__cvvTitle {
  padding-right: 10px;
  font-size: 15px;
  font-weight: 500;
  color: #fff;
  margin-bottom: 5px;
}

```

```

.card-item__cvvBand {
  height: 45px;
  background: #fff;
  margin-bottom: 30px;
  text-align: right;
  display: flex;
  align-items: center;
  justify-content: flex-end;
  padding-right: 10px;
  color: #1a3b5d;
  font-size: 18px;
  border-radius: 4px;
  box-shadow: 0px 10px 20px -7px rgba(32, 56, 117, 0.35);
}

@media screen and (max-width: 480px) {
  .card-item__cvvBand {
    height: 40px;
    margin-bottom: 20px;
  }
}

@media screen and (max-width: 360px) {
  .card-item__cvvBand {
    margin-bottom: 15px;
  }
}

.card-list {
  margin-bottom: -130px;
}

@media screen and (max-width: 480px) {
  .card-list {
    margin-bottom: -120px;
  }
}

```

```

    }
}

.card-input {
    margin-bottom: 20px;
}
.card-input__label {
    font-size: 14px;
    margin-bottom: 5px;
    font-weight: 500;
    color: #1a3b5d;
    width: 100%;
    display: block;
    user-select: none;
}
.card-input__input {
    width: 100%;
    height: 50px;
    border-radius: 5px;
    box-shadow: none;
    border: 1px solid #ced6e0;
    transition: all 0.3s ease-in-out;
    font-size: 18px;
    padding: 5px 15px;
    background: none;
    color: #1a3b5d;
    font-family: "Source Sans Pro", sans-serif;
}
.card-input__input:hover, .card-input__input:focus {
    border-color: #3d9cff;
}
.card-input__input:focus {

```

```

        box-shadow: 0px 10px 20px -13px rgba(32, 56, 117, 0.35);
    }
    .card-input__input.-select {
        -webkit-appearance: none;
        background-image:
url("data:image/png;base64,iVBORw0KGgoAAAANSUhEUgAAADIAAAAAeCAYAAABu
UU38AAAAGXRFWHRTb2Z0d2FyZQBBZG9iZSBjbWFnZVJIYWR5ccllPAAAAUxJRE
FUErM1sEJwkAQBDcsngXPHsQO9O5FS7AAMVYgdqAd2IGCDWgFnryLFQiCZ8EGnJ
UNimiyM/tnk4HNEAg/8y6ZmMRVqz9eUJvRaSbvutCZ347bXVJy/ZnvTmdJ862Me+hAbZ
CTs6GHpyUi1tTSvPnqTpoWZPUa7W7ncT3vK4h4zVeJy8QzM3WhVUO8ykI6jOxoGA4ig
3BLHcNFSCGqGAkig2yqgpEiMsjSfY9LxYQg7L6r0X6wS29YJiYQYecemY+wHrXD1+b
klGhpAhBDeu/JfIVGxaAQ9sb8CI+CQSJ+QmJg0Ii/EE2MBiIXooHRQhRCkBhNhBcEhLk
wf05ZCG8ICCOpk0MULmvDSY2M8UawIRExLIQIEgHDRoghihgRIgiigBEjgiFATBACA
gFgghEwSAAGgoBCBBgYAg5hYKAIFYgHBo6w9RRgAFfy160QuV8NAAAAAEIFTkSu
QmCC");
        background-size: 12px;
        background-position: 90% center;
        background-repeat: no-repeat;
        padding-right: 30px;
    }

.slide-fade-up-enter-active {
    transition: all 0.25s ease-in-out;
    transition-delay: 0.1s;
    position: relative;
}

.slide-fade-up-leave-active {
    transition: all 0.25s ease-in-out;
    position: absolute;
}

```

```
.slide-fade-up-enter {  
  opacity: 0;  
  transform: translateY(15px);  
  pointer-events: none;  
}
```

```
.slide-fade-up-leave-to {  
  opacity: 0;  
  transform: translateY(-15px);  
  pointer-events: none;  
}
```

```
.slide-fade-right-enter-active {  
  transition: all 0.25s ease-in-out;  
  transition-delay: 0.1s;  
  position: relative;  
}
```

```
.slide-fade-right-leave-active {  
  transition: all 0.25s ease-in-out;  
  position: absolute;  
}
```

```
.slide-fade-right-enter {  
  opacity: 0;  
  transform: translateX(10px) rotate(45deg);  
  pointer-events: none;  
}
```

```
.slide-fade-right-leave-to {  
  opacity: 0;  
  transform: translateX(-10px) rotate(45deg);
```

```

    pointer-events: none;
}

.github-btn {
    position: absolute;
    right: 40px;
    bottom: 50px;
    text-decoration: none;
    padding: 15px 25px;
    border-radius: 4px;
    box-shadow: 0px 4px 30px -6px rgba(36, 52, 70, 0.65);
    background: #24292e;
    color: #fff;
    font-weight: bold;
    letter-spacing: 1px;
    font-size: 16px;
    text-align: center;
    transition: all 0.3s ease-in-out;
}

@media screen and (min-width: 500px) {
    .github-btn:hover {
        transform: scale(1.1);
        box-shadow: 0px 17px 20px -6px rgba(36, 52, 70, 0.36);
    }
}

```

A.13 Листинг файла buet.js

```
/*
See on github: https://github.com/muhammederdem/credit-card-form
*/
new Vue({
  el: "#app",
  data() {
    return {
      currentCardBackground: Math.floor(Math.random()* 25 + 1), // just for
fun :D
      cardName: "",
      cardNumber: "",
      cardMonth: "",
      cardYear: "",
      cardCvv: "",
      cardCounter:"",
      minCardYear: new Date().getFullYear(),
      amexCardMask: "##### ####",
      otherCardMask: "##### ####",
      cardNumberTemp: "",
      isCardFlipped: false,
      focusElementStyle: null,
      isInputFocused: false
    };
  },
  mounted() {
    this.cardNumberTemp = this.otherCardMask;
    document.getElementById("cardNumber").focus();
  },
});
```



```

computed: {
  getCardType () {
    let number = this.cardNumber;

    let re = new RegExp("^4");
    if (number.match(re) !== null) return "visa";

    re = new RegExp("(34|37)");
    if (number.match(re) !== null) return "amex";

    re = new RegExp("^5[1-5]");
    if (number.match(re) !== null) return "mastercard";

    re = new RegExp("^6011");
    if (number.match(re) !== null) return "discover";

    re = new RegExp('^9792')
    if (number.match(re) !== null) return 'troy'

    return "visa"; // default type
  },
  generateCardNumberMask () {
    return this.getCardType === "amex" ? this.amexCardMask :
this.otherCardMask;
  },
  minCardMonth () {
    if (this.cardYear === this.minCardYear) return new Date().getMonth() +
1;

    return 1;
  }
},

```

```

watch: {
  cardYear () {
    if (this.cardMonth < this.minCardMonth) {
      this.cardMonth = "";
    }
  }
},
methods: {
  flipCard (status) {
    this.isCardFlipped = status;
  },
  focusInput (e) {
    this.isInputFocused = true;
    let targetRef = e.target.dataset.ref;
    let target = this.$refs[targetRef];
    this.focusElementStyle = {
      width: `${target.offsetWidth}px`,
      height: `${target.offsetHeight}px`,
      transform: `translateX(${target.offsetLeft}px)
translateY(${target.offsetTop}px)`
    }
  },
  blurInput() {
    let vm = this;
    setTimeout(() => {
      if (!vm.isInputFocused) {
        vm.focusElementStyle = null;
      }
    }, 300);
    vm.isInputFocused = false;
  }
}

```

```
    }  
  }  
});
```

A.14 Листинг файла enter.css

```
@import url("https://fonts.googleapis.com/css?family=Fira+Sans");
```

```
* {  
  box-sizing: border-box;  
}  
.section {  
  padding-top: 100px;  
}  
.header {  
  padding-top: 100px;  
}  
.container {  
  width: 500px;  
  margin: 0 auto;  
}  
  
h1 {  
  font-weight: bold;  
  margin: -50px 0px 0px 0px;  
  text-align: center;  
}  
  
h2 {  
  text-align: center;  
}
```

```

p {
  font-size: 14px;
  font-weight: 100;
  line-height: 20px;
  letter-spacing: 0.5px;
  margin: 100px 0px 0px 25px;

}

span {
  font-size: 12px;
}

.parrol{
  color: #333;
  font-size: 10px;
  text-decoration: none;
  margin: 0px 0px 0px 0px;
  display: flex;
  align-items: center;
  justify-content: center;
}

a {
  color: #333;
  font-size: 15px;
  text-decoration: none;
  margin: 15px 0px;
}

html,body {
  display: flex;

```

```
position: relative;
min-height: 100vh;
background-color: #E1E8EE;
display: flex;
align-items: center;
justify-content: center;
font-family: "Fira Sans", Helvetica, Arial, sans-serif;
-webkit-font-smoothing: antialiased;
-moz-osx-font-smoothing: grayscale;
flex-direction: column;
}
```

```
.form-structor {
background-color: #012452;
border-radius: 15px;
height: 550px;
width: 400px;
position: relative;
overflow: hidden;
```

```
&::after {
    content: "";
    opacity: .8;
    position: absolute;
    top: 0;right:0;bottom:50;left:0;
    background-repeat: no-repeat;
    background-position: left bottom;
    background-size: 500px;
    background-image: url('https://phonoteka.org/uploads/posts/2021-06/1624925880_30-phonoteka_org-p-gazprom-oboi-krasivo-30.jpg');
```

```
}
```

```
.signup {  
    position: absolute;  
    top: 55%;  
    left: 50%;  
    -webkit-transform: translate(-50%, -50%);  
    width: 65%;  
    z-index: 5;  
    -webkit-transition: all .3s ease;  
  
    &.slide-up {  
        top: 5%;  
        -webkit-transform: translate(-50%, 0%);  
        -webkit-transition: all .3s ease;  
    }  
  
    &.slide-up .form-holder,  
    &.slide-up .submit-btn {  
        opacity: 0;  
        visibility: hidden;  
    }  
  
    &.slide-up .form-title {  
        font-size: 1em;  
        cursor: pointer;  
    }  
  
    &.slide-up .form-title span {
```

```

margin-right: 5px;
opacity: 1;
visibility: visible;
-webkit-transition: all .3s ease;
}

.form-title {
color: #fff;
font-size: 1.7em;
text-align: center;

span {
color: rgba(0,0,0,0.4);
opacity: 0;
visibility: hidden;
-webkit-transition: all .3s ease;
}
}

.form-holder {
border-radius: 20px;
background-color: #fff;
overflow: hidden;
margin-top: 50px;
opacity: 1;
visibility: visible;
-webkit-transition: all .3s ease;

.input {
border: 0;

```

```

        outline: none;
        box-shadow: none;
        display: block;
        height: 30px;
        line-height: 30px;
        padding: 8px 15px;
        border-bottom: 1px solid #eee;
        width: 100%;
        font-size: 12px;

        &:last-child {
            border-bottom: 0;
        }
        &::-webkit-input-placeholder {
            color: rgba(0,0,0,0.4);
        }
    }
}

```

```

.submit-btn {
    background-color: rgba(0,0,0,0.4);
    color: rgba(256,256,256,0.7);
    border: 0;
    border-radius: 10px;
    display: block;
    margin: 15px auto;
    padding: 15px 45px;
    width: 100%;
    font-size: 13px;
    font-weight: bold;
}

```



```

        cursor: pointer;
        opacity: 1;
        visibility: visible;
        -webkit-transition: all .3s ease;

        &:hover {
            transition: all .3s ease;
            background-color: rgba(0,0,0,0.8);
        }
    }
}

.overflow{
    position: absolute;
    top: 20%;
    left: 20;
    right: 20;
    bottom: 0;
}

.form {
    position: absolute;
    top: 20%;
    left: 20;
    right: 20;
    bottom: 0;
    background-color: #fff;
    z-index: 5;
    -webkit-transition: all .3s ease;
}

```

```

.login {
    position: absolute;
    top: 20%;
    left: 0;
    right: 0;
    bottom: 0;
    background-color: #fff;
    z-index: 5;
    -webkit-transition: all .3s ease;

    &::before {
        content: "";
        position: absolute;
        left: 50%;
        top: -20px;
        -webkit-transform: translate(-50%, 0);
        background-color: #fff;
        width: 200%;
        height: 250px;
        border-radius: 50%;
        z-index: 4;
        -webkit-transition: all .3s ease;
    }

    .center {
        position: absolute;
        top: calc(50% - 10%);
        left: 50%;
        -webkit-transform: translate(-50%, -50%);
        width: 65%;
    }

```

```

z-index: 5;
-webkit-transition: all .3s ease;

.form-title {
    color: #000;
    font-size: 1.7em;
    text-align: center;

    span {
        color: rgba(0,0,0,0.4);
        opacity: 0;
        visibility: hidden;
        -webkit-transition: all .3s ease;
    }
}

.form-holder {
    border-radius: 15px;
    background-color: #fff;
    border: 1px solid #eee;
    overflow: hidden;
    margin-top: 50px;
    opacity: 1;
    visibility: visible;
    -webkit-transition: all .3s ease;

    .input {
        border: 0;
        outline: none;
        box-shadow: none;

```

```

        display: block;
        height: 30px;
        line-height: 30px;
        padding: 8px 15px;
        border-bottom: 1px solid #eee;
        width: 100%;
        font-size: 12px;

        &:last-child {
            border-bottom: 0;
        }

        &::-webkit-input-placeholder {
            color: rgba(0,0,0,0.4);
        }
    }
}

```

```

.submit-btn {
    background-color: #6B92A4;
    color: rgba(256,256,256,0.7);
    border: 0;
    border-radius: 15px;
    display: block;
    margin: 20px auto;
    padding: 15px 45px;
    width: 100%;
    font-size: 13px;
    font-weight: bold;
    cursor: pointer;
    opacity: 1;
}

```

```

        visibility: visible;
        -webkit-transition: all .3s ease;

        &:hover {
            transition: all .3s ease;
            background-color: rgba(0,0,0,0.8);
        }
    }
}

&.slide-up {
    top: 90%;
    -webkit-transition: all .3s ease;
}

&.slide-up .center {
    top: 10%;
    -webkit-transform: translate(-50%, 0%);
    -webkit-transition: all .3s ease;
}

&.slide-up .form-holder,
&.slide-up .submit-btn {
    opacity: 0;
    visibility: hidden;
    -webkit-transition: all .3s ease;
}

&.slide-up .form-title {
    font-size: 1em;

```

```

        margin: 0;
        padding: 0;
        cursor: pointer;
        -webkit-transition: all .3s ease;
    }

    &.slide-up .form-title span {
        margin-right: 5px;
        opacity: 1;
        visibility: visible;
        -webkit-transition: all .3s ease;
    }
}
}

```

A.15 Листинг файла enter.js

```

console.clear();

const loginBtn = document.getElementById('login');
const signupBtn = document.getElementById('signup');

loginBtn.addEventListener('click', (e) => {
    let parent = e.target.parentNode.parentNode;
    Array.from(e.target.parentNode.parentNode.classList).find((element) => {
        if(element !== "slide-up") {
            parent.classList.add('slide-up')
        } else {
            signupBtn.parentNode.classList.add('slide-up')
            parent.classList.remove('slide-up')
        }
    })
})

```

```

});
});

signupBtn.addEventListener('click', (e) => {
  let parent = e.target.parentNode;
  Array.from(e.target.parentNode.classList).find((element) => {
    if(element !== "slide-up") {
      parent.classList.add('slide-up')
    } else {
      loginBtn.parentNode.parentNode.classList.add('slide-up')
      parent.classList.remove('slide-up')
    }
  });
});
});

```

А.16 ЛИСТИНГ файла index.css

```

* { box-sizing: border-box; }

@import
url('https://fonts.googleapis.com/css?family=Rubik:400,500&display=swap');

body {
  font-family: 'Rubik', sans-serif;
}

.menu{
  display: flex;
  align-items: center;
  text-decoration: none;
  margin-right: auto;

```

```

        letter-spacing: 0.05em;

    }

    .form-field1 {
        height: 46px;
        padding: 0 16px;
        border: 2px solid #ddd;
        border-radius: 4px;
        font-family: 'Rubik', sans-serif;
        outline: 0;
        transition: .2s;
        margin-top: 10px;
    }

    ul,li {
        display: block;
        padding: 0;
        margin: auto;
    }

    .text{
        margin-bottom: 10px;
        font-weight: 600;
        margin-top: 20px;
    }

    h6{
        display: flex;
        font-size: 18px;
        vertical-align: middle;
        margin-bottom: 20px;
    }

```



```

    font-weight: 600;
    border-bottom: 1px solid #e0e0e0;
    padding-bottom: 5px !important;
}
h10 {
    text-transform: uppercase;
    display: flex;
    vertical-align: middle;
    align-items: center;
    font-size: 10px;
    font-weight: bold;
    margin: auto;
    margin-bottom: 10px;
}
.grypa{
    padding-top: 5px;
    display: flex;
    justify-content: space-between;
    margin-bottom: 0px;
    justify-content: center;
}
.grypa2{
    padding-top: 5px;
    display: flex;
    justify-content: space-between;
    align-items: center;
    margin-bottom: 0px;
    justify-content: center;
}
.grypa3{

```

```
margin-top: 6%;
padding-top: 5px;
display: flex;
justify-content: space-between;
align-items: center;
margin-bottom: 0px;
justify-content: center;
}
.img-radius{
padding: 35px;
margin-right: 10px;
}
```

```
.container {
display: flex;
height: 100vh;
}
```

```
.left {
overflow: hidden;
display: flex;
flex-wrap: wrap;
justify-content: center;
animation-name: left;
animation-duration: 1s;
animation-fill-mode: both;
animation-delay: 1s;
}
```

```
.right {
```

```
flex: 1;
background-color: black;
transition: 1s;
background-image: url(soblyudayte-ostorozhnost-pri-obrashchenii-s-
bytovym-gazom_16195969651044611869__2000x2000.png);
background-size: cover;
background-repeat: no-repeat;
background-position: center;
}
```

```
.header{
  display: flex;
  flex-direction: column;
  justify-content: end;
}
```

```
.header > h2 {
  margin: 0;
  color: #4f46a5;
}
```

```
.header > h4 {
  margin-top: 10px;
  font-weight: normal;
  font-size: 15px;
  color: rgba(0,0,0,.4);
}
```

```
.form {
```

```
    max-width: 80%;  
    display: flex;  
    flex-direction: column;  
}
```

```
.form > p {  
    text-align: right;  
}
```

```
.form > p > a {  
    color: #000;  
    font-size: 14px;  
}
```

```
.form-field {  
    height: 46px;  
    padding: 0 16px;  
    border: 2px solid #ddd;  
    border-radius: 4px;  
    font-family: 'Rubik', sans-serif;  
    outline: 0;  
    transition: .2s;  
    margin-top: 10px;  
}
```

```
.form-field:focus {  
    border-color: #0f7ef1;  
}
```

```
.btn {
```

```
background: linear-gradient(to right, #de48b5 0%,#0097ff 100%);
background-size: 100%;
color: #fff;
border:0;
border-radius: 3px;
display: block;
padding: 12px 10px;
width: 90%;
border-radius: 25px;
margin-top: 10px;
letter-spacing: 1px;
font-family: 'Rubik', sans-serif;
font-size: 13px;
font-weight: bold;
cursor: pointer;
opacity: 1;
visibility: visible;
-webkit-transition: all .3s ease;
    transition: 1.5s;

&:hover {
    background: rgba(251,75,2,1);
box-shadow:
    -7px -7px 20px 0px rgba(25, 149, 206, 0.9),
    -4px -4px 5px 0px rgba(185, 69, 201, 0.9),
    7px 7px 20px 0px rgba(212, 37, 75, 0.842),
    4px 4px 5px 0px rgba(24, 228, 211, 0.589);
    background: linear-gradient(to left, #de48b5 0%,#0097ff 100%);
    background-position: left;
    transition: all .3s ease;
```

```
        background-color: rgba(0,0,0,0.8);  
    }  
  
}
```

```
.animation {  
    animation-name: move;  
    animation-duration: .4s;  
    animation-fill-mode: both;  
    animation-delay: 2s;  
}
```

```
.animation {  
    animation-name: move;  
    animation-duration: .4s;  
    animation-fill-mode: both;  
    animation-delay: 2s;  
}
```

```
.animation {  
    animation-name: move;  
    animation-duration: .4s;  
    animation-fill-mode: both;  
    animation-delay: 2s;  
}
```

```
.animation {  
    animation-name: move;  
    animation-duration: .4s;  
    animation-fill-mode: both;
```

```
    animation-delay: 2s;  
}
```

```
.a1 {  
    animation-delay: 2s;  
}
```

```
.a2 {  
    animation-delay: 2.1s;  
}
```

```
.a3 {  
    animation-delay: 2.2s;  
}
```

```
.a4 {  
    animation-delay: 2.3s;  
}
```

```
.a5 {  
    animation-delay: 2.4s;  
}
```

```
.a6 {  
    animation-delay: 2.5s;  
}
```

```
.a7 {  
    animation-delay: 2.6s;  
}
```

```
.a8 {  
  animation-delay: 2.7s;  
}
```

```
.a9 {  
  animation-delay: 2.8s;  
}
```

```
.a10 {  
  animation-delay: 2.9s;  
}
```

```
.a11 {  
  animation-delay: 3.0s;  
}
```

```
.a12 {  
  animation-delay: 3.1s;  
}
```

```
.a13 {  
  animation-delay: 3.2s;  
}
```

```
.a14 {  
  animation-delay: 3.3s;  
}
```

```
@keyframes move {  
  0% {  
    opacity: 0;  
    visibility: hidden;  
    transform: translateY(-40px);
```



```

    }

    100% {
        opacity: 1;
        visibility: visible;
        transform: translateY(0);
    }
}

```

```

@keyframes left {
    0% {
        opacity: 0;
        width: 0;
    }

    100% {
        opacity: 1;
        padding: 20px 40px;
        width: 440px;
    }
}

```

A.17 Листинг файла index1.css

```

* { box-sizing: border-box; }

@import
url('https://fonts.googleapis.com/css?family=Rubik:400,500&display=swap');

body {
    font-family: 'Rubik', sans-serif;
}

```

```

}
table{
    margin: 0 0 0 0;

}
.menu{
    display: flex;
    align-items: center;
    text-decoration: none;
    margin-right: auto;
    letter-spacing: 0.05em;
}
td , th{
    text-align: center;
}
th{
    align-items: center;
    font-weight: 100;
    text-align-last:center ;
}
.form-field1 {
    height: 30px;
    padding: 0 16px;
    border: 2px solid #ddd;
    border-radius: 4px;
    font-family: 'Rubik', sans-serif;
    outline: 0;
    transition: .2s;
    margin-top: 10px;
}

```

```
ul,li {
    display: flex;
    padding: 0;
    margin: auto;
}

.text{
    margin-bottom: 10px;
    font-weight: 600;
    margin-top: 20px;
}

h6{
    display: flex;
    font-size: 18px;
    vertical-align: middle;
    margin-bottom: 20px;
    font-weight: 600;
    border-bottom: 1px solid #e0e0e0;
    padding-bottom: 5px !important;
}

h10 {
    text-transform: uppercase;
    display: flex;
    vertical-align: middle;
    align-items: center;
    font-size: 10px;
    font-weight: bold;
    margin: auto;
    margin-bottom: 10px;
}
```

```

}
.grypa{
    padding-top: 5px;
    display: flex;
    justify-content: space-between;
    margin-bottom: 0px;

}

.grypa-r{
    display: flex;
    flex-direction: column;
}

.grypa-l{
    display: flex;
    flex-direction: column;
}

```

```

.row{
    justify-content: space-between;
}

.grypa2{
    padding-top: 5px;
    /*align-items: center;*/
    display: flex;
    justify-content: space-between;
    margin-bottom: 0px;
}

```

```
.grypa3 {  
  padding-top: 5px;  
  display: flex;  
  justify-content: space-between;  
  /*align-items: center;*/  
}
```

```
.img-radius {  
  padding: 35px;  
  margin-right: 10px;  
}
```

```
.container {  
  display: flex;  
  height: 100vh;  
}
```

```
.left {  
  overflow: hidden;  
  display: flex;  
  flex-wrap: wrap;  
  justify-content: center;  
  animation-name: left;  
  animation-duration: 1s;  
  animation-fill-mode: both;  
  animation-delay: 1s;  
}
```

```
.right {  
  flex: 1;
```

```
background-color: black;
transition: 1s;
background-image: url(soblyudayte-ostorozhnost-pri-obrashchenii-s-
bytovym-gazom_16195969651044611869__2000x2000.png);
background-size: cover;
background-repeat: no-repeat;
background-position: center;
}
```

```
.header{
  display: flex;
  flex-direction:column;
  justify-content: end;
}
```

```
.header > h2 {
  margin: 0;
  color: #4f46a5;
}
```

```
.header > h4 {
  margin-top: 10px;
  font-weight: normal;
  font-size: 15px;
  color: rgba(0,0,0,.4);
}
```

```
.form {
  max-width: 80%;
```

```
display: flex;
flex-direction: column;
}
```

```
.form > p {
  text-align: right;
}
```

```
.form > p > a {
  color: #000;
  font-size: 14px;
}
```

```
.form-field {
  height: 30px;
  width: 100px;
  padding: 0 10px;
  border: 2px solid #ddd;
  border-radius: 4px;
  font-family: 'Rubik', sans-serif;
  outline: 0;
  transition: .2s;
  margin: 10px 0px 0px 0px;
}
```

```
.form-field:focus {
  border-color: #0f7ef1;
}
```

```
.btn {
```

```
background: linear-gradient(to right, #de48b5 0%,#0097ff 100%);
background-size: 100%;
color: #fff;
border:0;
border-radius: 3px;
display: block;
padding: 12px 10px;
width: 90%;
border-radius: 25px;
margin-top: 10px;
letter-spacing: 1px;
font-family: 'Rubik', sans-serif;
font-size: 13px;
font-weight: bold;
cursor: pointer;
opacity: 1;
visibility: visible;
-webkit-transition: all .3s ease;
    transition: 1.5s;

&:hover {
    background: rgba(251,75,2,1);
box-shadow:
    -7px -7px 20px 0px rgba(25, 149, 206, 0.9),
    -4px -4px 5px 0px rgba(185, 69, 201, 0.9),
    7px 7px 20px 0px rgba(212, 37, 75, 0.842),
    4px 4px 5px 0px rgba(24, 228, 211, 0.589);
    background: linear-gradient(to left, #de48b5 0%,#0097ff 100%);
    background-position: left;
    transition: all .3s ease;
```



```
        background-color: rgba(0,0,0,0.8);  
    }  
  
}
```

```
.animation {  
    animation-name: move;  
    animation-duration: .4s;  
    animation-fill-mode: both;  
    animation-delay: 2s;  
}
```

```
.animation {  
    animation-name: move;  
    animation-duration: .4s;  
    animation-fill-mode: both;  
    animation-delay: 2s;  
}
```

```
.animation {  
    animation-name: move;  
    animation-duration: .4s;  
    animation-fill-mode: both;  
    animation-delay: 2s;  
}
```

```
.animation {  
    animation-name: move;  
    animation-duration: .4s;  
    animation-fill-mode: both;
```

```
    animation-delay: 2s;  
}
```

```
.a1 {  
    animation-delay: 2s;  
}
```

```
.a2 {  
    animation-delay: 2.1s;  
}
```

```
.a3 {  
    animation-delay: 2.2s;  
}
```

```
.a4 {  
    animation-delay: 2.3s;  
}
```

```
.a5 {  
    animation-delay: 2.4s;  
}
```

```
.a6 {  
    animation-delay: 2.5s;  
}
```

```
.a7 {  
    animation-delay: 2.6s;  
}
```

```
.a8 {  
  animation-delay: 2.7s;  
}
```

```
.a9 {  
  animation-delay: 2.8s;  
}
```

```
.a10 {  
  animation-delay: 2.9s;  
}
```

```
.a11 {  
  animation-delay: 3.0s;  
}
```

```
.a12 {  
  animation-delay: 3.1s;  
}
```

```
.a13 {  
  animation-delay: 3.2s;  
}
```

```
.a14 {  
  animation-delay: 3.3s;  
}
```

```
@keyframes move {  
  0% {  
    opacity: 0;  
    visibility: hidden;  
    transform: translateY(-40px);
```

```
}

100% {
  opacity: 1;
  visibility: visible;
  transform: translateY(0);
}
}
```

```
@keyframes left {
  0% {
    opacity: 0;
    width: 0;
  }

  100% {
    opacity: 1;
    padding: 20px 40px;
    width: 440px;
  }
}
```

А.18 Листинг файла top.css

```
@import
url('https://fonts.googleapis.com/css?family=Montserrat|Open+Sans|Roboto');

*{
  margin:0;
  padding: 0;
  outline: 0;
}
```

```

body{
  background-color: #f70000;
  padding: 0px;
  margin: 0px;
  flex-direction: column;
}

#gradient
{
  width: 100%;
  height: 100vh;
  background-size: cover;
  padding: 0px;
  margin: 0px;
}

.table{
  display: flex;
  align-items: center;
  justify-content: center;
  text-transform: uppercase;
  position: absolute;
  z-index: 2;
  transform: translate(-50%,-50%);
  top: 10%;
}

.form{
  display: flex;
  align-items: center;
  justify-content:center ;
  flex-direction: column;

```

```

}
.Button{
  display: flex;
  align-items:center;
  justify-content:center ;
  position:absolute;
  transition: transform 80ms ease-in;
  top: 55%;
  z-index: 1;
}
.btn{
  border: 1px solid #1E90FF;
  background-color: #1E90FF;
  color: #FFFFFF;

  text-align: center;
  font-weight: bold;
  font-size: 12px;
  padding: 12px 45px;
  letter-spacing: 1px;
  border-radius: 20px;
  text-transform: uppercase;
  cursor: pointer;
  opacity: 1;
  visibility: visible;
  -webkit-transition: all .3s ease;

  &:hover {
    transition: all .3s ease;
  }
}

```

```

        background-color: rgba(0,0,0,0.8);
        border: 1px rgba(0,0,0,0.8);
    }
}
.filter{
    position: absolute;
    left: 0;
    top: 0;
    bottom: 0;
    right: 0;
    z-index: 1;
    opacity: .7;
}
table{
    display: flex;
    align-items: center;
    justify-content: center;
    text-transform: uppercase;
    position: absolute;
    z-index: 2;
    left: 50%;
    transform: translate(-50%,-50%);
    border-collapse: collapse;
    border-spacing: 0;
    box-shadow:
        -7px -7px 20px 0px rgba(25, 149, 206, 0.9),
        -4px -4px 5px 0px rgba(185, 69, 201, 0.9),
        7px 7px 20px 0px rgba(212, 37, 75, 0.842),
        4px 4px 5px 0px rgba(24, 228, 211, 0.589);
    border-radius: 25px 25px 10px 10px;

```

```

overflow: hidden;
top: 300px;
}
td , th{
padding: 15px 20px;
text-align: center;
color: white;
text-transform: none;
text-shadow: 0.5px 0.5px 0.5px black;
letter-spacing: 0.5px;
}
th{
background-color: #1E90FF;
color: white;
text-decoration: none;
font-family: 'Open Sans', 'Times New Roman', Times, serif;
font-weight: 100;
text-transform: uppercase;
text-align-last:center ;
text-shadow: 0.5px 0.5px 0.5px black;
letter-spacing: 0.5px;
}
tr{
background-color: #1E90FF;
background-color: linear-gradient(to left, #de48b5 0%,#0097ff 100%);
font-family: 'Montserrat', 'Times New Roman', Times, serif;
}
tr:nth-child(even){
background-color: #1E90FF;
}

```


А.19 Листинг файла top.js

```
var colors = new Array(
    [62,35,255],
    [60,255,60],
    [255,35,98],
    [45,175,230],
    [255,0,255],
    [255,128,0]);
var step = 0;
//color table indices for:
// current color left
// next color left
// current color right
// next color right
var colorIndices = [0,1,2,3];
//transition speed
var gradientSpeed = 0.002;
function updateGradient()
{
    if ( $===undefined ) return;
    var c0_0 = colors[colorIndices[0]];
    var c0_1 = colors[colorIndices[1]];
    var c1_0 = colors[colorIndices[2]];
    var c1_1 = colors[colorIndices[3]];
    var istep = 1 - step;
    var r1 = Math.round(istep * c0_0[0] + step * c0_1[0]);
    var g1 = Math.round(istep * c0_0[1] + step * c0_1[1]);
    var b1 = Math.round(istep * c0_0[2] + step * c0_1[2]);
    var color1 = "rgb("+r1+","+g1+","+b1+)";
```

```

var r2 = Math.round(istep * c1_0[0] + step * c1_1[0]);
var g2 = Math.round(istep * c1_0[1] + step * c1_1[1]);
var b2 = Math.round(istep * c1_0[2] + step * c1_1[2]);
var color2 = "rgb("+r2+","+g2+","+b2+)";

$('#gradient').css({
    background: "-webkit-gradient(linear, left top, right top, from("+color1+"),
to("+color2+"))").css({
    background: "-moz-linear-gradient(left, "+color1+" 0%, "+color2+"
100%)");

step += gradientSpeed;
if ( step >= 1 )
{
    step %= 1;
    colorIndices[0] = colorIndices[1];
    colorIndices[2] = colorIndices[3];




    //pick two new target color indices
    //do not pick the same as the current one
    colorIndices[1] = ( colorIndices[1] + Math.floor( 1 + Math.random() *
(colors.length - 1))) % colors.length;
    colorIndices[3] = ( colorIndices[3] + Math.floor( 1 + Math.random() *
(colors.length - 1))) % colors.length;

}
}
setInterval(updateGradient,10);

```

Дневник

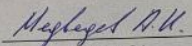
прохождения учебной практики _____ УП.11 _____
 по профессиональному модулю ПМ.11 Разработка, администрирование и защита баз данных
 Код, специальность, группа 09.02.07 Информационные системы и программирование. ТИП-42
 Студента _____ Владимирова Ивана Сергеевича _____
 (Ф.И.О.)

Дата	Тема занятия	Объем выполненной работы	Подпись руководителя практики от предприятия
01.05.23 - 13.05.23	Анализ предметной области. Составление схем предметной области	72	
15.05.23 - 24.05.23	Проектирование БД. Создание физической схемы. Заполнение БД	54	
25.05.23 - 31.05.23	Создание запросов на выборку к своей БД	36	

Итоговая оценка отлично

Руководитель практики


 (подпись)


 (Ф.И.О.)

Аттестационный лист
по учебной практике

студента Московского техникума космического приборостроения МГТУ им.Н.Э.Баумана
Владимирова Ивана Сергеевича
(Ф.И.О. студента)

Группа **ТИП-42**

Специальность **09.02.07 Информационные системы и программирование** Квалификация –
Программист

(код, наименование специальности)

прошел учебную практику **УП 11 ПМ.Разработка, администрирование и защита баз данных**
(наименование практики)

по профессиональному модулю **ПМ.11 Разработка, администрирование и защита баз данных**
в объеме 162 часа

(наименование профессионального модуля)

с «01» Мая 2023 года по «31» Мая 2023 года

Виды и качество работ в период учебной практики

Виды работ, выполненные студентом во время практики, согласно программы учебной практики	Результат (по 5-ти бальной шкале)
Разработка баз данных и соответствующими ему компетенциями и необходимыми умениями и опытом практической работы по профессии	5

В ходе учебной практики студентом освоены следующие профессиональные компетенции

Код и название профессиональной компетенции	Результат освоения (освоена/не освоена)
ПК 11.1.Осуществлять сбор, обработку и анализ информации для проектирования баз данных.	Освоена
ПК 11.2.Проектировать базу данных на основе анализа предметной области	Освоена
ПК 11.3.Разрабатывать объекты базы данных в соответствии с результатами анализа предметной области.	Освоена
ПК 11.4.Реализовывать базу данных в конкретной системе управления базами данных.	Освоена

Характеристика профессиональной деятельности обучающегося во время учебной практики:

Итоговая оценка по практике Отлично

Руководитель практики от образовательного учреждения

преподаватель
Должность

Подпись
Подпись

Осипова Н.М.
Ф.И.О. руководителя практики