

Платежи

Видение

Требуется создать рабочий прототип web-приложения для работы со счетом и платежами.

Приложение должно реализовывать следующий функционал:

В режиме клиента:

- Пополнение и блокировка счета;
- Просмотр списка платежей и оплаты их при помощи счета;

В режиме администратора

- Снятие блокировки со счета;
- Просмотр списка клиентов;
- Просмотр списка и редактирование платежей;

1. Клиенты

1.1. Просмотр списка клиентов

Данный режим предназначен для просмотра списка клиентов Администратором.

Основной сценарий:

- Администратор выбирает пункт меню "Clients";
- Отображается форма просмотра всех клиентов;

В списке отображаются следующие колонки:

- Id – идентификационный номер клиента;
- Name - Фамилия, имя клиента;
- Date - Дата регистрации;
- Status - Статус счета (активный/заблокирован);

1.2. Фильтрация

Для просмотра списка клиентов, которые зарегистрировались за определенный период, предусмотрена возможность выбора фильтра периода отбора данных с до и по.

Начальная дата фильтра не должна быть больше конечной.

Если данные не введены, фильтрация не производится.

Если не введено значение начальной даты, то фильтрация производится только по конечной дате.

Если не введена конечная дата, то фильтрация производится только по начальной дате.

Фильтр по умолчанию выставляется с 1-го числа текущего месяца и по текущую дату.

Обновление данных после выбора даты фильтрации осуществляется нажатием кнопки "Update".

Также предусмотрена фильтрация по статусу счета и фио клиента.

The screenshot shows the 'Payments 1.0' application interface. At the top, there are three window control buttons and the title 'Payments 1.0'. Below this is a navigation bar with a user profile icon and the name 'Admin', and two tabs: 'Clients' (selected) and 'Payments'. The main area contains a filter section with 'Period from' and 'to' date pickers (showing 03.03.2019 and 05.03.2019), a 'sort by' dropdown menu (set to 'Select'), and an 'Update' button. Below the filters is a table with columns: ID, NAME, REG. DATE, and STATUS. The table contains three rows of client data. Each row has 'Lock' and 'Detail' buttons. The third row also has an 'Unlock' button.

ID	NAME	REG. DATE	STATUS		
021	Gorichev Mikhail	04.03.2019	Active	Lock	Detail
022	Vasiliy Kovalev	04.03.2019	Active	Lock	Detail
023	Andrey Andreev	04.03.2019	Blocked	Unlock	Detail

1.3. Снятие блокировки со счета

Основной сценарий:

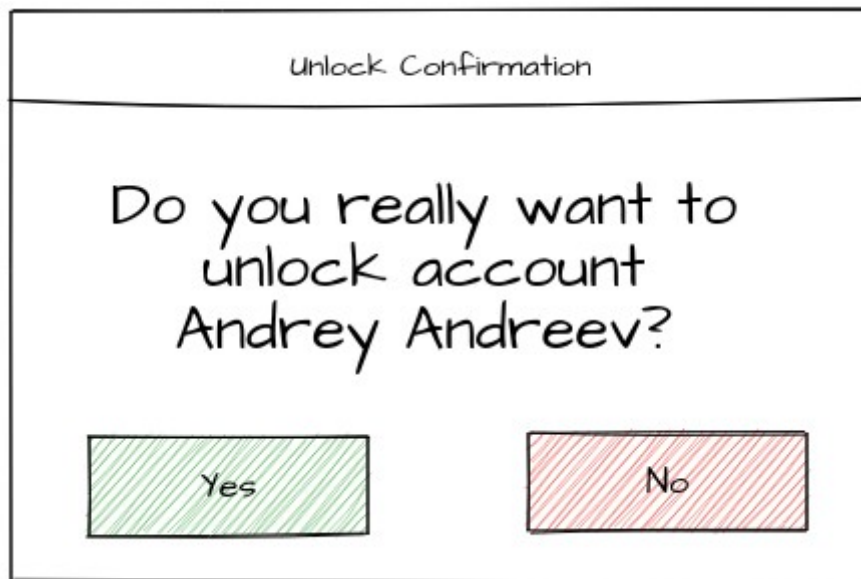
- Администратор находясь в режиме просмотра списка клиентов жмет по кнопке "Unlock" в строке клиента, счет которого заблокирован;
- Если счет клиента можно разблокировать, то отображается диалог подтверждения снятия блокировки:
"Do you really want to unlock account<Name>?";
- Администратор нажимает кнопку "Yes";
- Происходит снятие блокировки со счета в базе данных;
- Если произошла ошибка снятия блокировки, то выводится сообщение об ошибке "Data deleting error!";
- Если блокировка успешно снята, то открывается обновленный список клиентов.

Сценарий отмены изменений:

- Администратор находясь в режиме просмотра списка клиентов жмет по кнопке “Unlock” в строке клиента, счет которого заблокирован;
- Если счет клиента можно разблокировать, то отображается диалог подтверждения снятия блокировки:

“Remove lock from the account<Name>?”;

- Администратор нажимает кнопку “No”;
- Открывается список клиентов без изменений



2. Платежи

2.1. Просмотр списка платежей администратором

Данный режим предназначен для просмотра списка платежей Администратором.

Основной сценарий:

- Администратор выбирает пункт меню “Payments”;
- Отображается форма просмотра всех платежей;

В списке отображаются следующие колонки:

- Id – идентификационный номер платежа;
- Type – тип платежа;
- Date - Дата добавления платежа;

The screenshot shows a web application titled "Payments IO". At the top, there is a navigation bar with a user profile icon and the name "Admin", and two tabs: "Clients" and "Payments" (which is active). Below the navigation bar, there is a filter section with "Period from" and "to" date pickers (showing 03.03.2019 and 05.03.2019), a "sort by" dropdown menu (set to "Select"), and an "Update" button. Below the filter section is a table of payments. The table has columns for "ID", "Type", "REG. DATE", and an action column. The action column contains an "Add Payment" button for the first row and "Remove" buttons for the subsequent rows. The table contains three rows of data.

ID	Type	REG. DATE	
001	Education	04.03.2019	Add Payment
002	Medical Services	04.03.2019	Remove
003	Public Services	04.03.2019	Remove

2.2. Фильтрация

Как и в случае с клиентами, есть возможность фильтрации по дате и типу платежа.

2.3. Добавление платежа

Основной сценарий:

- Администратор находясь в режиме просмотра списка платежей жмет по кнопке "Add payment";
- Открывается форма добавления платежа;
- Администратор вводит данные и нажимает "Add";
- Если данные введены не корректно, то выводится соответствующее сообщение;
- Если данные корректны, то платеж добавляется в базу данных
- Если произошла ошибка сохранения данных в базе данных, то выводится соответствующее сообщение;
- Если данные успешно сохранились, открывается список обновленных платежей.

Сценарий отмены изменений:

- Администратор находясь в режиме просмотра списка платежей жмет по кнопке "Add payment";
- Открывается форма добавления платежа;
- Администратор нажимает кнопку "Cancel";
- Открывается список платежей без изменений;

Hand-drawn sketch of the "Add Payment" form. The form has a title bar "Add Payment". It contains two input fields: "Type:" with the value "insurance" and "Cost:" with the value "500\$". At the bottom, there are two buttons: "Add" (green with diagonal lines) and "Cancel" (red with diagonal lines).

2.4. Ограничения для вводимых данных

Для поля "Type":

- Длина не более 30-ти символов;
- Состоит только из букв;
- Не может начинаться с цифры;
- Значение должно быть уникальным в базе данных

Для поля "Cost":

- Состоит только из цифр;
- Не может быть больше 10000;

2.5. Удаление платежа

Основной сценарий:

- Администратор находясь в режиме просмотра списка платежей жмет по кнопке "Remove" в строке платежа, который он хочет удалить;
- Если платеж можно удалить, то отображается форма подтверждения удаления;
- Администратор нажимает кнопку "Yes";
- Происходит удаление платежа в базе данных;
- Если произошла ошибка удаления, то выводится сообщение об ошибке "Data deleting error!";
- Если платеж успешно удален, то открывается список платежей после удаления;

Сценарий отмены изменений:

- Администратор находясь в режиме просмотра списка платежей жмет по кнопке "Remove" в строке платежа, который он хочет удалить;
- Если платеж можно удалить, то отображается форма подтверждения удаления;
- Администратор нажимает кнопку "No";
- Открывается список платежей без изменений.

3. Счёт

3.1. Просмотр списка платежей клиентом

Данный режим предназначен для проведения платежей клиентом

Основной сценарий:

- Пользователь выбирает пункт меню "Payments";
- Отображается форма просмотра всех платежей;

В списке отображаются следующие колонки:

- Id – идентификационный номер платежа;
- Type – тип платежа;
- Date - Дата добавления платежа;
- Cost – Стоимость платежа;

The screenshot shows a web application interface titled "Payments I.O.". At the top, there's a navigation bar with a user profile icon, the name "Gorichev Mikhail", a balance of "1500\$", and buttons for "Deposit", "Payments" (highlighted), and "Lock Account". Below the navigation bar, there's a filter section with "Period from" (03.03.2019), "to" (05.03.2019), "sort by" (Select), and an "Update" button. The main content area displays a table of payments with columns: ID, Type, REG. DATE, Cost, and a "Pay" button for each row.

ID	Type	REG. DATE	Cost	
001	Education	04.03.2019	200\$	Pay
002	Medical Services	04.03.2019	300\$	Pay
003	Public Services	04.03.2019	400\$	Pay

3.2. Проведение платежа

Основной сценарий:

- Клиент находясь в режиме просмотра списка платежей жмет по кнопке "Pay" в строке платежа, который он хочет оплатить;
- Открывается форма проведения платежа;
- Если платеж можно провести, то отображается форма подтверждения оплаты;
- Пользователь нажимает кнопку "Yes";
- Происходит оплата платежа в базе данных;
- Если произошла ошибка оплаты, то выводится сообщение об ошибке;

- Если платеж успешно оплачен, то со счёта снимается указанная сумма денег и открывается список платежей;

Сценарий отмены оплаты:

- Клиент находясь в режиме просмотра списка платежей жмет по кнопке "Pay" в строке платежа, который он хочет оплатить;
- Открывается форма проведения платежа;
- Если платеж можно провести, то отображается форма подтверждения оплаты;
- Пользователь нажимает кнопку "No";
- Открывается список платежей;

Сценарий при недостаточном количестве средств на счету:

- Клиент находясь в режиме просмотра списка платежей жмет по кнопке "Pay" в строке платежа, который он хочет оплатить;
- Открывается форма проведения платежа;
- Если платеж нельзя провести (недостаточно средств на счету), выводится соответствующая ошибка;
- Открывается список платежей;

A hand-drawn dialog box with a title bar labeled "Payment". The main text inside the box asks: "Do you really want to make a payment 'Education' on the cost of 200\$". At the bottom of the box, there are two buttons: a green button labeled "Yes" and a red button labeled "No".

3.3. Пополнение счёта

Основной сценарий:

- Клиент находясь в режиме просмотра списка платежей жмет по кнопке "Deposit";
- Открывается форма пополнения счета;
- Пользователь вводит соответствующие данные карты, сумму пополнения и нажимает кнопку "Deposit";
- Происходит проверка введенных данных;
- Если данные корректны, то производится пополнения счета;
- Если произошла ошибка пополнения, выводится соответствующее сообщение;
- Если пополнение произошло успешно, то выводится соответствующее сообщение и открывается список платежей;

Сценарий отмены пополнения:

- Клиент находясь в режиме просмотра списка платежей жмет по кнопке "Deposit";
- Открывается форма пополнения счета;
- Клиент нажимает кнопку "Cancel"
- Если данные корректны, то производится пополнения счета;
- Открывается список платежей;

Deposit

Card No.

4650 5888 6622 8936

Sum:

500\$

Deposit

Cancel

3.4. Блокировка счета

Основной сценарий:

- Клиент находясь в режиме просмотра списка платежей жмет по кнопке "Lock account";
- Открывается форма подтверждения блокировки платежа;
- Клиент жмёт по кнопке "Yes";
- Производится соответствующая запись в базу данных;
- Если произошла ошибка записи в базу данных, выводится соответствующее сообщение;
- Если запись произошла успешно, тип счета клиента меняется на "Locked" и ему не доступен список платежей для оплаты;

Сценарий отмены блокировки:

- Клиент находясь в режиме просмотра списка платежей жмет по кнопке "Lock account";
- Открывается форма подтверждения блокировки платежа;
- Клиент жмёт по кнопке "No";
- Открывается список платежей;

The diagram shows a rectangular window titled "Block account". Inside the window, the text "Do you really want to block your account?" is centered. Below the text are two buttons: a green button labeled "Yes" and a red button labeled "No".