План   
  
1) закончить раздел заказы  
2) адаптировать раздел выполнить действие   
3) адаптировать сам меню сайдбара  
4) адаптировать раздел добавить выполненный заказ  
5) импортировать выполненные заказы из аккаунтов, диалога ТГ и рассчитать прибыль за последние 2-3 месяца  
6) сделать раздел рассчета прибыли, адаптировать к этому курс   
7) показать статистику карт, аккаунтов, заказанных товаров  
8) оптимизировать puppeteer скрипт (перезагрузка при зависаний а также пауза при долгой загрузке)  
9) вывести все настройки скрипта в интерфейс   
  
  
  
**1) закончить раздел заказы**  
**Отдельные таблицы для разной даты с обозначением дня недели**:  
Мы должны разделить таблицу по дням, по неделям и по месяцу.   
должна быть иконка (или несколько иконок), кликнув на который мы могли бы изменить отображения таблицы раздельно по дате.   
Пример: кликаем на иконку отображать по дням и у нас таблица разделится на заказы, где группируются только заказы, совершенные в один день.   
Сверху должна отображаться дата с обозначением дня недели, например 07 Апреля, понедельник и все заказы, которые сделаны в этот день.   
если кликнули на неделю, то сверху отображение порядка дня недели, например 2 -ая неделя, апрель и все заказы, заказанные за 2 ую неделю апреля.  
Соответственно за месяц тоже самое.   
И кнопка отобразить все заказы.   
 Также **раздел статистика,** который находится снизу должен показывать статистику так и отдельно, когда мы меняем вид, то есть за день, за неделю и за месяц и также общую.   
  
  
**1) Редактирование с кликом на нужное значение   
2) Отмена действий  
3) Подсветка статуса заказа и быстрое изменение статуса   
4) Статистика суммы по дням и общий**Нам нужно теперь решить 2 задачи.   
1) индивидуальная статистика под заказом

2) внедрить поле статус оплаты и также внедрить его в статистику.   
  
1) Теперь нам нужно добавить статистику под каждой таблицей   
например когда у нас разделяются таблицы по дням, неделям и месяцам у нас имеется одна общая статистика снизу, а теперь нам нужна индивидуальная статистика для каждой таблицы.   
Естественно, он должен корректно динамический работать с таблицами, когда они изменяются.   
Он также должен показывать новую статистику.  
Кол-во неоплаченных заказов.  
Сумма неоплаченных заказов.  
Сумма оплаченных заказов.

2)   
 **5) Расширенная сортировка по дате   
6) Существующие значения для дропдаунов   
7) добавить поля orderID, статус оплаты,**

**Статус оплаты:**будем модифицировать поле статус оплаты, а также сделаем интерфейс для него.   
Интерфейс должен быть максимально продуманным, быстрым и красивым.   
также должен использовать элементы react bootstrap где это необходимо.   
Должен иметь понятную подсветку поля или шрифта или что то еще что бы администратору было легче ориентировать в статусе оплаты заказа.   
А также должен быть возможность быстро его поменять если клиент оплатил и тд.  
еще будем взаимодействовать с файлом "D:\Didar1520\CRM\data\ClientData\clientData.json"  
  
  
У нас будет несколько статусов оплаты:  
1) оплачен (paid)  
стандартная оплата заказа. Надо подсветить зеленым или синим что бы было понятно, что заказ оплачен.  
также когда выбираем этот пункт (у неоплаченного заказа) мы естественно и логично должны в ordersData.json статус оплаченного заказа изменить на paid

2) не оплачен (not paid)  
надо показать его красным и с надписью не оплачен

3) передан агенту.  
это означает что клиент оплатил за заказ, но не на прямую а отправил деньги через другого человека (потому что сам не умеет пользоваться байбит)  
агентом у меня выступают другие клиенты, по этому когда выбираем этот пункт нам нужно показать инпут для выбора клиентов из того же ClientsData json. ("D:\Didar1520\CRM\data\ClientData\clientData.json")  
агентом могут выступать клиенты, которые имеют transferAgent": true  
вот тут должны сохранять историю и сумму переводов, а также общую сумму, которая накопилось у агента:  
"safekeepingMoney": {

        "totalCollected": 65000.00,

        "transfers": [

          {

            "sender": "Мария Николаева",

            "amount": 40000.00,

            "date": "11-04-2025"

          },

          {

            "sender": "Вероника Храмова",

            "amount": 25000.00,

            "date": "12-04-2025",

            "time": "14:30"

          }

        ]},

Позже сделаем поля с информациями о передачах агента, но это уже в другом разделе.  
Я думаю здесь тебе понятна логика.   
например если у клиента выбираем поле (дропдаунт или кнопка) передан агенту у нас должна появится опция выбора агента.   
они также берутся из ClientsData json, берутся клиенты с пометкой transferAgent": true.  
Затем администратор должен выбрать клиента – агента.   
затем в нужные автоматический записывается сумма, дата и имя клиента из таблицы который передал сумму агенту.  
5) рассчет комиссий и новая функция изменение полей.   
нажимать каждый раз редактировать, а затем искать нужный инпут в поп ап может быть неудобен, когда редактируется только одно поле, например частое поле статус заказа. (но эту функцию нужно оставить)  
только вместо надписей редактировать или удалить которые не помещаются нужно поставить иконки карандаша и корзины)  
По этому нам нужно внедрить что то вроде редактирование по клику.   
а для поле статус оплаты, наверное, нужно сделать по-особенному, наверное, нужно сделать красивые радиокнопки из реакт бутстрапа что бы по-быстрому можно было поменять статус оплаты.   
также нужно кнопка редактировать все, что бы пометить (выделить) нужные заказы и разом поменять статус оплаты или что то еще.   
А функцию изменения, наверное, нужно сделать по двойному клику либо по специальном мини иконке для редактирование поля.   
также у меня в сервере не настроена передача нужных данных для фунционирования того что я запросил, нужно будет их тоже добавить.   
  
  
  
…………..

 1) хорошо, также в ordersData есть поле paymentStatus, нужно также передать ее через сервер или передать весь файл и не морочится. Также нужно передать clientData потому что он тоже не передан в данный момент.  
2-5 хорошо, ок  
  
Уточняющие вопросы по функционалу статуса оплаты:  
1) у нас уже есть поле paymentStatus, значения сделаем на английском но отобразим в таблице на русском.   
да, можно добавить поле paid Amount, хотя у нас в логике частично оплаченные заказы тоже предусмотрены, но дополнительный маркер указывающий на это тоже не помешает   
2) да, это должно быть индивидуально для каждого значения, особенно важно для статуса оплаты.  
да, обязательно нужно сохранить редактирование через поп ап окно, также добавить в окно новоле поле статус оплаты где можно быстро изменить статус или ввести нужную сумму если заказ оплачен частично   
  
3) все это я вроде уже написал, но повторю на всякий случай  
3,1 долг клиента содержится здесь: "balance": {

        "positiveBalance": 0,

        "debt": 145421,

debt это долг клиента, а позитивный баланс это если клиент переплатил. (переплатой клиента займемся в другом модуле)  
3,2, не понял вопрос, но там вроде ничего сложного, если пометить заказ как не оплачен он должен перейти в неоплаченные заказы с указанием всех этих данных.   
Но, этим занимается другой модуль, этим занимается модуль puppeteer во время заказа он сразу помечает заказы как неоплаченные и сохраняет в профили клиента после заказа. Это на всяккий случай если я буду добавлять уже заказанные заказы в ручную.   
4,1 все правильно  
4,2 наверное нужна кнопка сохранить значение

5, наверное да, только там нужна возможность редактировать не только статус но и любое поле как клиент, сумма заказа, комиссия и тд  
5,1 нужно реализовать сейчас   
6)для этого есть поле orderComission в clientData. Для каждого клиента индивидуальное значение, в целом 4-6%/   
пример "ordersComission": 4,

Да, он должен вычислятся как сумма заказа из ордерс дата плюс ордерскомишн% из клиентдата   
например если сумма заказа 20000 руб а ordersComission равна 5 то долг будет   
20000 + 5%  
   
  
**online online online online online online online online online online online online online online online online online online online o**

мне нужно сделать кое какие изменения, напиши какие файлы тебе нужны для этого

1) Определить конкретно, когда заказ завершилось удачно, а когда произошла ошибка

1) Считается, когда заказ завершилось удачно это когда модуль finalReceipt.js успешно отработал и вернул результат выполненного заказа, а если заказ до него не дошел, значит заказ не был выполнен.

Это справедливо только для заказа (когда у нас есть cartLink), для синхронизаций, утилиты reviewManager результат успешной работы другой.

У нас есть два состояния выполненного заказа  
1) товары успешно куплены, это когда мы получаем апи api/checkout/order?OrderNumber со всеми вытекающими данными заказа, это означает что заказ куплен и выполнен.  
2) второй случай это когда на счету не было денег или какой то другой случай помешал нашему заказу.  
тогда мы оказываемся на странице <https://p-proc-srv.iherb.com/ui/transaction?transactionId=9c1e2b94-1b35-4e8e-8792-c57a94b5c897&returnUrl=https://checkout12.iherb.com/scd/order-receipt?on=535641190&sca=1&origin=SimpleCheckoutPayment>  
где 535641190 это номер заказа (ественно цифры могут быть любыми, я его взял для примера)

И получаем такой апи: <https://orders-order-api.iherb.biz/api/v1/Order?on=535641190> (опять же номер уникальный для примера, он может быть любым) /  
в этом случае заказ стоит в брони на 48 часов и мы должны записать из этого апи данные:  
  
нужно записать в те же поля что при обычном заказа,  
только в iherbStatus нужно поставить в этом случае booked, что означает что заказ забронирован

адрес: внутри объекта shipAddr поле firstName

Использовання карта: paymMethod": {

"billAddrID": null,

"cardNb": "5581",

Аккаунт и сумма заказа из этого апи (приходит одновременно с тем): https://p-proc-srv.iherb.com/api/Transactions/92a17f7f-9773-47e8-a610-14f8c8a20803/TransactionData

поля  
userEmail

totalAmount

пример: (    "referenceId": "535641573",

    "paymentAccountId": "100000000049227058",

    "totalAmount": 198.45,

    "currency": "USD",

    "userEmail": "astana111ikz@proton.me",

)

Вот эти два кейса — это пример успешного выполнения заказа. Я не софтверный архитектор и не знаю как правильно, но вообщем нам нужно сделать так что бы программа определяла успешность заказа этими двумя кейсами, если мы не получили ни то ни другое это означает что заказ завершился неудачно.