

# Himp Manual

V2.0.1



Frank-Peter Poschmann - March 18, 2024

Date	Account	Title / Foreign A...	Details	Transfer Type	Category	User	Group	Amount
01/03/2024	XY Bank	Electricity		Direct Debit	Electricity	MeMyselfA...		-\$78.00
01/16/2024	XY Bank	Vegetables + Bo...		Debit Card	Household...	MeMyselfA...		-\$12.45
01/27/2024	Cash	Rolls		Cash Paym...	Groceries	MeMyselfA...		-\$3.40
01/27/2024	Cash	Soap		Cash Paym...	Drugstore	MeMyselfA...		-\$2.20
02/01/2024	XY Bank	Electricity		Direct Debit	Electricity	MeMyselfA...		-\$78.00
02/17/2024	Cash	ATM		Cash Paym...	Others	MeMyselfA...		\$200.00
02/19/2024	XY Bank	Credit Card xyz		Credit Card	Others	MeMyselfA...		-\$200.00
<b>Sum</b>								<b>-\$174.05</b>
Balance								

---

# Content

<b>Content</b>	<b>2</b>
<b>1. Introduction</b>	<b>3</b>
<b>2. Open Himp for the first time</b>	<b>3</b>
<b>3. The Maintenance View</b>	<b>4</b>
3.1. Accounts	4
3.2. Bookings	5
3.3. Users	7
3.4. Categories	7
3.5. Transfer Types	8
3.6. Groups	9
<b>4. The Input View</b>	<b>9</b>
<b>5. Automated Take-Over of Bookings</b>	<b>10</b>
<b>6. The Analysis View</b>	<b>14</b>
6.1. Identify Rebookings	15
6.2. Bookings in the Analysis View	16
6.3. Control of the Analysis View	16
6.3.1. The Distribution Graphs	17
6.3.2. The Trend Graphs	17
6.4. Split Up Transfer	19
<b>7. Synchronization on multiple Devices</b>	<b>19</b>

---

## 1. Introduction

Himp (Ringo Starr in Yellow Submarine: “I’ve got a **hole in my pocket.**”) is an app for iOS, iPadOS and macOS for recording and analyzing incoming and outgoing payments. It is intended to serve as a budget book and, in particular, to provide an up-to-date overview of income and expenses. To achieve this, it is important to record and classify as many payment transactions as possible. Himp provides the following contributions:

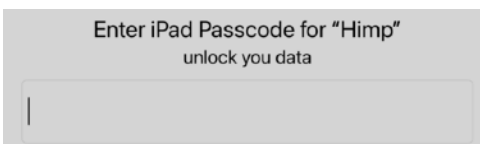
- Management of any number of accounts (including cash accounts)
- Interface for a (regular) import of account transactions from banking systems
- Synchronization of all account transactions across all devices with the same Apple ID via private iCloud (if desired)
- Classification of account transactions based on self-defined categories (including subcategories)
- Classification according to (self-defined) booking types
- Classification by users
- Assignment to self-defined groups of payments (e.g. vacation expenses)
- Convenient search for account transactions
- Graphical analysis of payments at monthly and annual levels
- Export of analyzes for further processing

This document is intended to serve as a manual to introduce the use of Himp. Particular emphasis is placed on practical examples that demonstrate the range of functions of Himp.

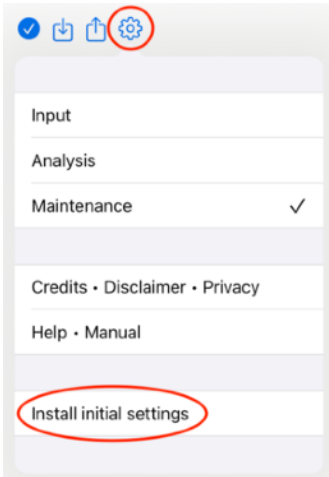
## 2. Open Himp for the first time

Then let’s get started right away and open Himp for the first time. Wow, there is a request for our password (Mac) or the unlock code (iPhone and iPad). This is used to unlock your data stored in Himp. Himp will display confidential data and you don't want it to be immediately visible to everyone who opens the app. Hence the password query, which occurs again after a certain period of inactivity or after returning from the background. Be safe: your data is protected and will not be shared with others.

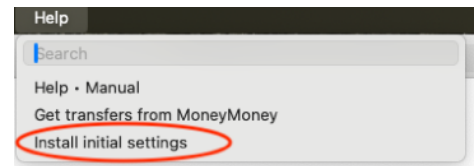
Once we’ve mastered that, our app still looks pretty orphaned. Since Himp is still empty, we ended up in the maintenance view and we can now make a few basic settings. The most im-



important thing is to create an account so that we can record bookings and account movements. But the possible classifications also have to be set up first.



That could be a lot of work, but Himp can help us here: On the iPad or iPhone, there is the “Install start settings” function in the gear menu in the menu bar, with which we can make a few basic settings. On the Mac we find this function in the Help menu.



If we have already made entries in Himp, this menu item will no longer be visible, so we cannot accidentally install the startup settings again.

Let's go. Let's install “Startup Settings” and see what's behind it.

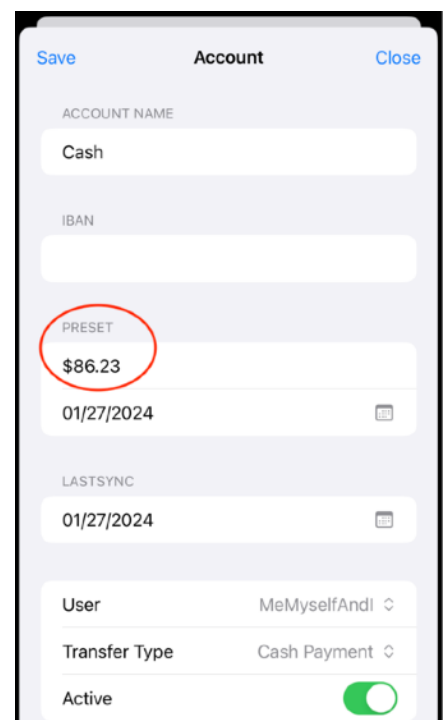
## 3. The Maintenance View

### 3.1. Accounts

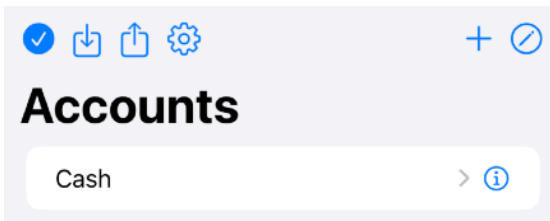
Wow, that was quick. A “Cash” account has already been created for us. We are now in the “Accounts” tab and see a list of all accounts that have been set up (currently 1). On the far right of the lines is a small ⓘ. If we tap on it (iOS) or click on it (macOS), we will see the settings of the corresponding account.

First of all, there is the name of our account, “Cash” in this case. This is followed by the IBAN or another account identifier. This is important if we want to do an automatic synchronization with a banking system to clearly identify the account. This doesn't matter for a cash account and can be omitted here.

Now comes the preset. Here we can enter what balance the account had on the morning of a specific day. In our example, we had \$86.23 in our wallet on January 27, 2024. The “last synchronization” is again only important for automatic synchronization with a banking system. We'll come back to that [later](#).



What follows are the default user and booking type for new bookings on this account. We'll get to the [users](#) and [transfer types](#) later. Then there is the indicator as to whether the account is still active. Non-active accounts are still taken into account in the [analysis](#), but they no longer appear in the [input view](#). We'll come back to that later too.



So when we go to “Save”, the changes are applied and we leave the detailed view of this account and see the list of accounts set up again.

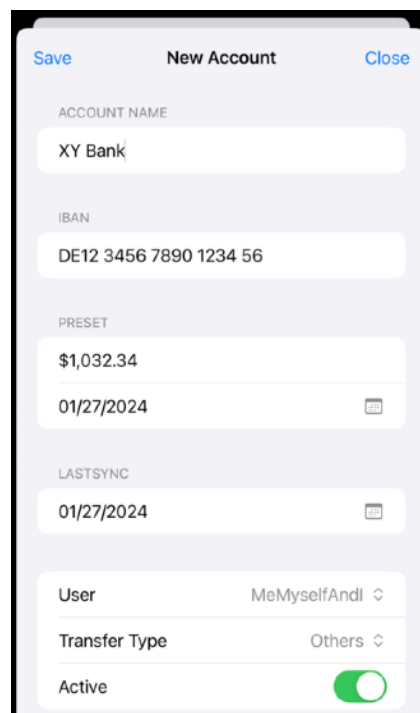
The symbols in the top line have the following meaning:

- ✔: This solid sign means we only see the active accounts in the list; the inactive ones are hidden. When we tap or click on this toggle, the icon is no longer filled (⊘) and we see all accounts including the inactive ones.
- ↓: This command allows us to set up accounts from a text file. This text file must be in a specific format for Himp to interpret it. The easiest way is to look at this format when exporting the account list (next function):
- ↑: Here we get the accounts that have been set up as a text file. For example, our cash account looks like this:

Name	IBAN	Preset	Preset Date	Last Sync	Default User	Default Transfer Type	Active
Cash		86.23	01/27/2024	01/27/2024	MeMyselfAndI	Cash Payment	true

The first line is the title line. It will be ignored during import. The individual fields are separated by tabs and correspond to the fields from the account detail view that we already know.

- ⚙: We already know the gear menu from setting up the initial settings. We'll come back to the other functions later.
- +: With the plus sign we can set up additional accounts. We want to do that right away and open a checking account at XY Bank. With “Save” we create the new account. And our account list shows two entries.
- ⊘: With this symbol we can edit our account list and delete accounts from it. We don't need that for now.



### 3.2. Bookings

When we tap (iOS) or click (macOS) on an account, we see the list of bookings of that account. We want to do that for the cash account. OK, that's still quite a bit. But there is also a + sign here. This allows us to create a new booking.

We can enter a few things here. Title and details help us describe the booking. In this example we bought rolls.

The amount we enter here is negative because we spent money. We'll come to a more convenient way to enter bookings [later](#). We're still in the maintenance view here, where we can see and change all the actual entries in Himp.

Our cash account was already selected as the account. The target account is only relevant when it comes to rebookings. We'll get to that later. We select "Groceries" as the category under "Household & Groceries". We'll come back to that soon. We will also come back to group classification later. We don't need them now.

The symbols in the title bar have the following meaning:

- ✔: We hereby save a new booking or changes to an existing booking. **Attention:** There is no "undo"!
- ✕: This allows us to delete an existing booking. **Here too:** There is no "undo"! This function is not available for a new booking.
- |□: This function is also only available for existing bookings. This allows us to split a transfer into two.
- ⊗: And hereby we discard all changes and do not create a new booking.

But we want to keep our new booking and select ✔. And now we have a booking in our cash account. We can change it again with the small ⓘ next to the booking.

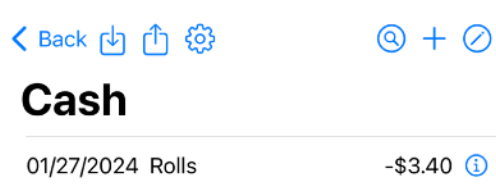
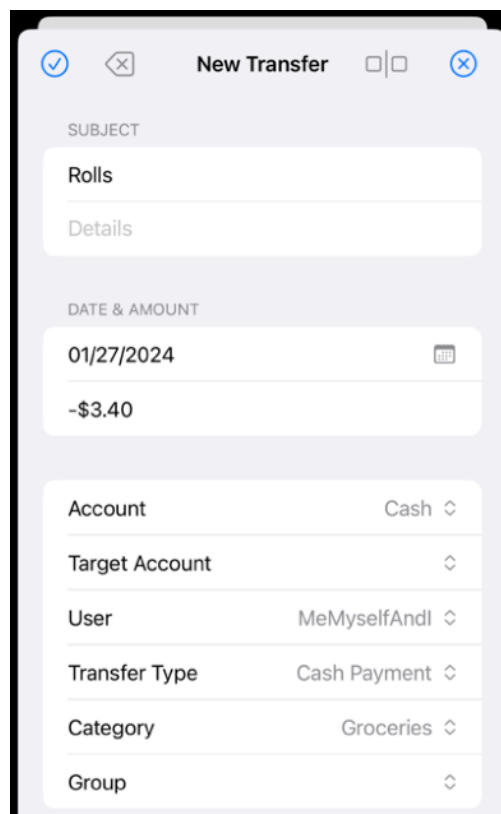
Here the symbols in the title bar have the following meaning:

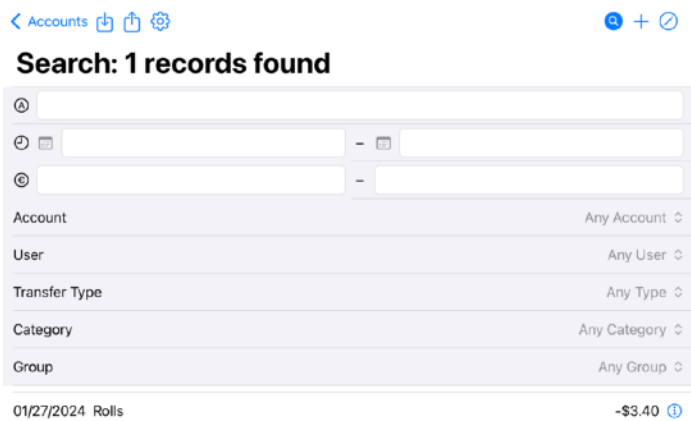
- ◀: As usual, we leave this view and return to the account list
- 📄: Here too we can import bookings from a text file. The format is the same as export:
- 📤: The bookings currently shown in the list are exported:

Title	Details	Date	Amount	Account	Secondary Account	Category
Rolls		01/27/2024	-3.4	Cash		Groceries
User	Transfer Type		Group			
MeMyselfAndI	Cash Payment					

Again the first line is the title line, which is ignored during import.

- ⚙️: We already know the gear menu.
- 🔍: Here we start the search for bookings. We can search for snippets of text in the titles or details, a booking date range, or a range of the amount. We can also limit the search according to criteria such as account, user, booking type, category or group.



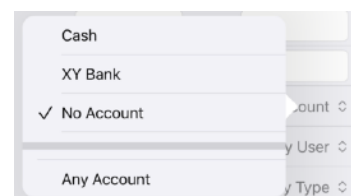


You can also search for bookings that are missing a criterion, e.g. all bookings without an account.

The bookings found appear below the search window. If we click or tap on the magnifying glass again, the search window disappears and all the account's bookings are shown again.

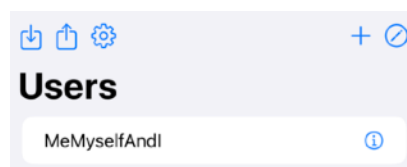
+: We have already created a new booking with this function.

⊘: And of course we can also delete the bookings.



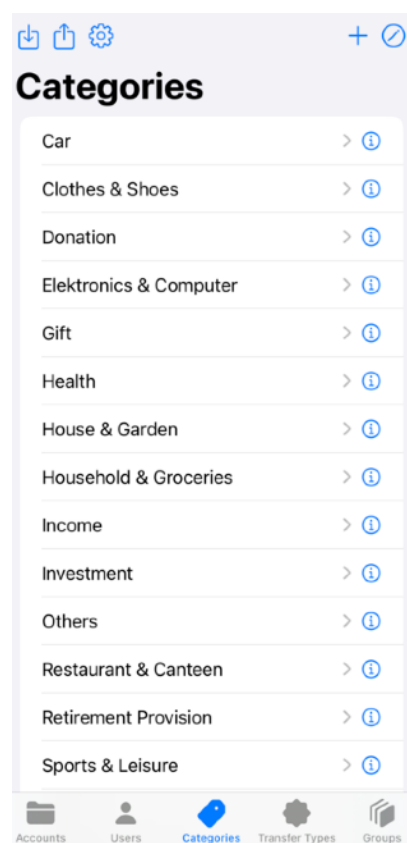
### 3.3. Users

So now we finally come to the classifications that we can give the bookings. At the bottom of the window we see the tabs for the individual classifications. There is the "User" indicator. We may use Himp in the family and want to analyze who actually generates the most income or expenses. We can assign the "User" flag for this.



The list of users looks quite familiar to us. In addition to importing and exporting, we can create or delete new users here. If we delete a user who we have already assigned to individual bookings, these bookings lose this indicator, but otherwise remain intact.

We can also change the corresponding names again using the small (i) next to the users. The assignments to the bookings are retained.

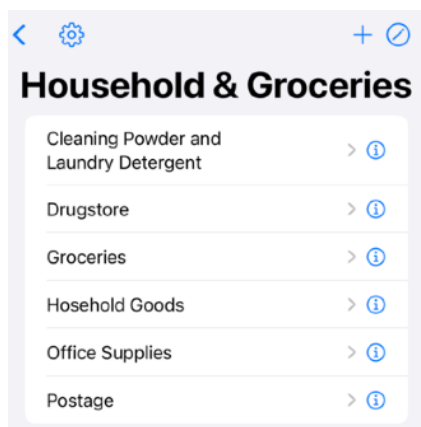


### 3.4. Categories

One of the most important characteristics of a booking is its category. Categories are organized hierarchically in Himp. A category can have any number of subcategories, which themselves may have subcategories. But we don't want to make it too complicated here and limit ourselves to two levels.

"Household & Groceries" only has one level of subcategories

like “Drugstore” or “Household Goods”.



We can assign categories to bookings at all levels. One booking can have the category “Drugstore” and another the category “Household & Groceries”. If we then search for the “Household & Groceries” category, we find both bookings because the first booking belongs to a subcategory of it. On the other hand, if we search for the “Drugstore” category, we only find the first booking. This also applies to the evaluations (see below).

When importing categories from a text file, the lines have a different number of fields depending on which hierarchy level the new category should belong to. The “parents” of a category must already exist or be at the top of the import file. So if we want to import the “Household & Groceries” categories, the text file should look something like this:

```
Household & Groceries
Household & Groceries      Office Supplies
Household & Groceries      Drugstore
Household & Groceries      Groceries
Household & Groceries      Postage
Household & Groceries      Cleaning Powder and Laundry Detergent
Household & Groceries      Household Goods
```

The sorting is irrelevant; Himp sorts the categories itself.

### 3.5. Transfer Types

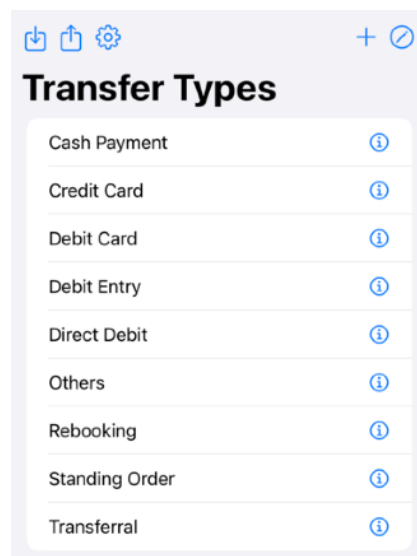
Transfer types are simple again. Here we just want to see which expenses we paid for, for example, by credit card and which by bank transfer.

“Rebooking” occupies a special position here. A rebooking always affects two accounts in Himp, an account from which the transfer took place and a target account. The transfer then appears in both accounts, in the outgoing account with a negative sign (as an expense) and in the destination account with a positive one (as income).

This makes a lot of sense, for example, when we withdraw cash from the ATM. This is debited from the current account (negative sign), and we suddenly have income in our cash account (positive sign).

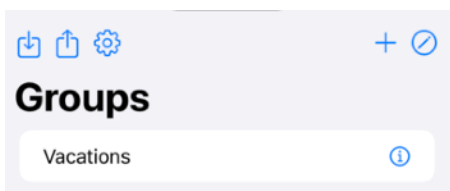
In the analysis ([see below](#)), the rebookings are ignored when we look at all accounts. They should not artificially increase income.

If we also have several checking accounts in Himp, rebookings can also occur there.





### 3.6. Groups

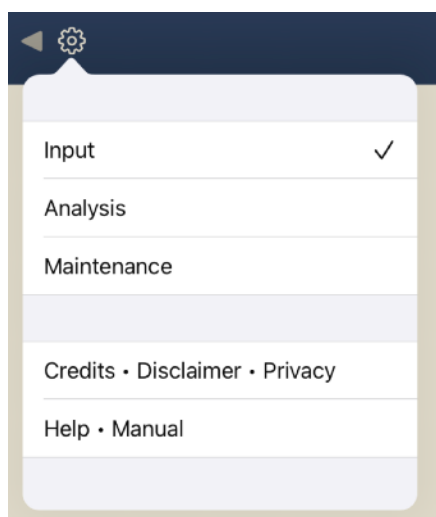


And then we have the groups. Categories are great for classifying bookings, but perhaps we want to combine bookings from different categories. A wide variety of expenses arise during a vacation, such as travel costs, but also entrance fees, meals or restaurant costs. If we want to know how much a vacation cost, we also need to be

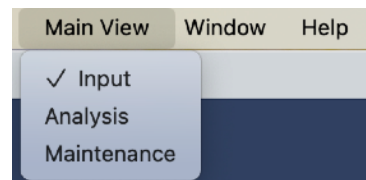
able to classify the bookings according to the “vacation” criterion. That's what the groups are for.

But we can also think of other groups, e.g. for special projects or hobbies.

## 4. The Input View



So, that was all just preliminaries to set Himp up. Now we come to the actual use of Himp. The input view is intended for entering bookings “by hand”, e.g. for cash accounts on the go. That’s why it’s the default view for mobile devices (iPhone and iPad).

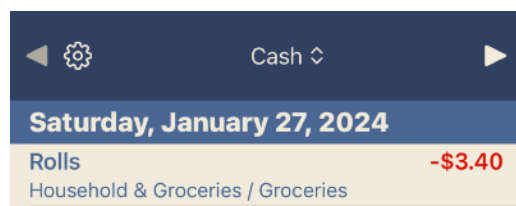


On iOS we change the view using the already familiar gear menu. On the Mac we find this under the “Main View” menu.

Then let’s switch to the input view. A month is always displayed here for a specific account. We have already created a booking in the maintenance view. It is now visible in

our cash account for January 2024.

In the input view, we change the account using the drop-down menu in the middle of the title bar. Himp remembers our last choice and automatically shows this account the next time we open it. We can scroll through the months using the triangles on the left and right in the title bar. If we have found January 2024, we will also see the booking that we created in the maintenance view.



In the lower part of the view we then see the balance for this month. The previous bookings and the preset of the account (see above) are taken into account. In our example, we should still have had €82.83 in our wallet at the end of January 2024. We



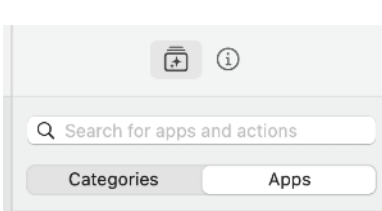
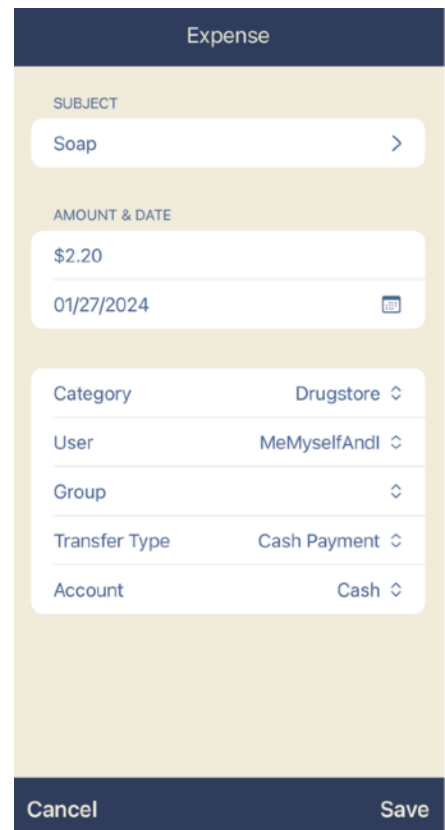
can then quickly switch between months using the drop-down menu in the middle of the balance line.

We can create new bookings using the two buttons below. We want to enter a new expense. Since Himp already knows that this is an expense, we don't enter a minus in the amount here.

We can also change a booking that has already been entered by simply tapping or clicking on the corresponding line. Or we delete a booking using the usual swipe gesture from right to left on the corresponding line.

## 5. Automated Take-Over of Bookings

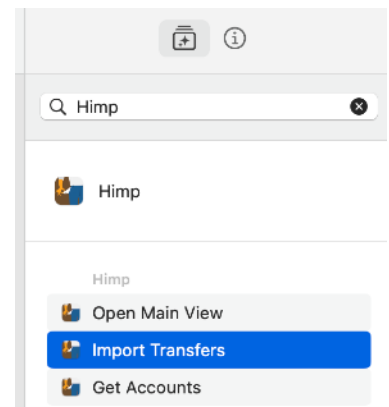
Now it is quite tedious to transfer all bookings from a current account into Himp by hand. Fortunately, there is an interface to banking systems to support this. This works best on the Mac and is called "Shortcuts". This is a default installed app for macOS and iOS.



If we open it, there is a column on the right where we click on the "Apps" tab. There we see the actions that the installed apps provide for automating processes. There can be a lot of them. Let's enter "Himp" in the search field at the top to see what Himp has to offer.

OK, there is something present.

- **Open Main Window:** With this action we can make Himp show a specific view, for example the input view. By default, Himp opens the analysis view on the Mac when it starts ([see below](#)).
- **Get Accounts:** This action is more interesting for our purpose. With this, Himp provides us with the accounts that have been set up, either all or just the active ones.
- **Import Transfers:** And with this action we can finally import bookings into Himp. The format is the same as we already know from the maintenance view for importing bookings.



Now, of course, it is unlikely that our banking system will deliver the bookings in the exact format we need for Himp. So we still have to reformat something. As an example, let's take a look at "[MoneyMoney](#)", an app that can be used to easily organize electronic banking on the Mac. Unfortunately, MoneyMoney doesn't yet offer actions for Shortcuts, but there are

commands for AppleScript. Don't worry, we'll go through step by step to create a shortcut to take over the bookings from MoneyMoney.

So we create a new shortcut and call it "Get transfers from MoneyMoney". The first action we take is to get the accounts that are set up in Himp and only the active ones. These accounts are then passed on to the next action. This is a loop that we want to run for all accounts. We can find the loop action under "Categories" / "Scripting" under "Control Flow" as "Repeat with Each". It takes a while to find our way around the many actions that Shortcuts provide.

But what does "Account" mean? If we click on the blue "Accounts" we can see what information is available to us for next actions.

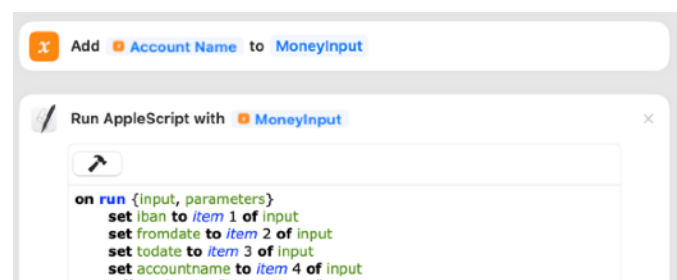
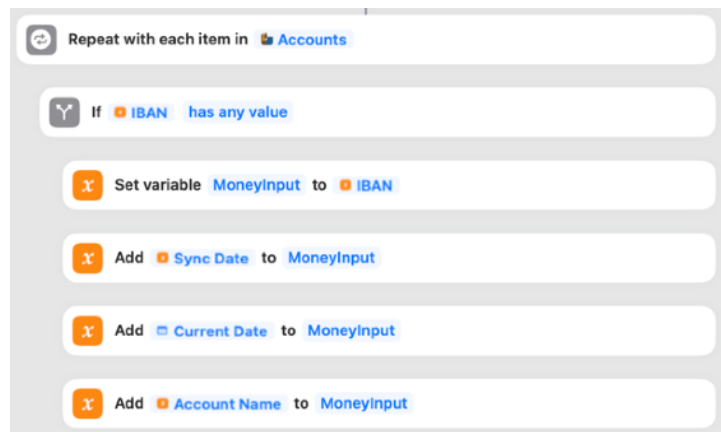
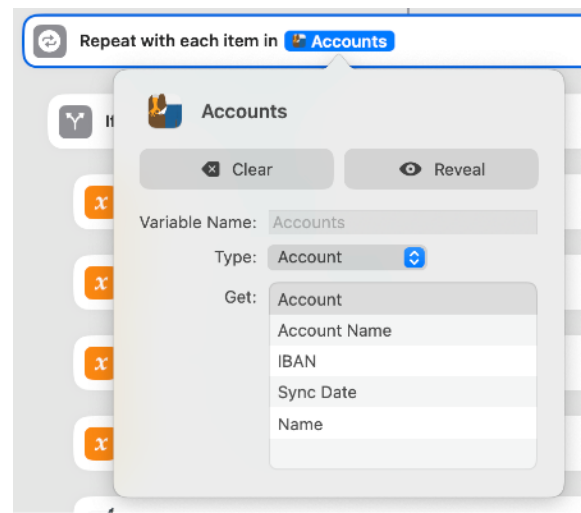
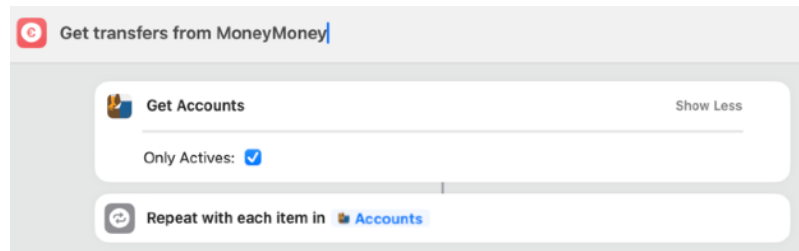
Aha, so the "Account" type from Himp delivers

- Account Name,
- IBAN and
- Sync Date.

This is all the information we entered when setting up the account in Himp. The value "Name" is assigned by the system itself and is not of interest to us here.

But we need the other information to get MoneyMoney to issue the correct bookings. To do this, we put together a variable "MoneyInput" that contains everything our AppleScript needs.

But first we check whether our account really has an IBAN; If not, we don't need to bother MoneyMoney about it. This can be done using the "If" action, which we can also find in "Control Flow". Then we fill a variable "MoneyInput" with the IBAN. We can also find this in "Scripting" under "Variables". Here we can also add more values to MoneyInput. These are the last sync date, the current



---

date, and the account name.

Well, now comes the AppleScript that does the actual work, and it's actually a bit more complicated:

```
on run {input, parameters}
    set iban to item 1 of input
    set fromdate to item 2 of input
    set todate to item 3 of input
    set accountname to item 4 of input
    tell application "MoneyMoney"
        try
            set result to export transactions from account iban from date fromdate to date todate as "plist"
        on error errMsg number errNr
            set result to errMsg
            if result = "Locked database." then
                display dialog "Please unlock MoneyMoney first and try again"
            end if
            return result
        end try
    end tell
    set r to result
    set outputText to ""
    tell application "System Events"
        set p to make new property list item with data r
        set p to property list items of property list item "transactions" of p
        repeat with ln in p
            set pp to property list item "booked" of ln
            set pp to value of pp
            if pp = true then
                set pp to property list item "bookingDate" of ln
                set pp to value of pp as date
                set theDate to short date string of pp
                set pp to property list item "name" of ln
                set theTitle to value of pp
                set pp to property list item "category" of ln
                set categoryItem to "Others"
                if (exists property list item "category" of ln) then
                    set pp to property list item "category" of ln
                    set categoryItem to value of pp
                end if
                set details to ""
                if (exists property list item "purpose" of ln) then
                    set pp to property list item "purpose" of ln
                    set details to value of pp
                end if
                if (exists property list item "bookingText" of ln) then
                    set pp to property list item "bookingText" of ln
                    set bText to value of pp as text
                    set details to details & ", Transaction type: " & bText
                end if
                if (exists property list item "endToEndReference" of ln) then
                    set pp to property list item "endToEndReference" of ln
                    set bText to value of pp as text
                    set details to details & ", Reference: " & bText
                end if
                if (exists property list item "mandateReference" of ln) then
                    set pp to property list item "mandateReference" of ln
                    set bText to value of pp as text
                    set details to details & ", Mandate: " & bText
                end if
                if (exists property list item "creditorId" of ln) then
                    set pp to property list item "creditorId" of ln
                    set bText to value of pp as text
                    set details to details & ", Creditor ID: " & bText
                end if
                set pp to property list item "amount" of ln
                set amount to value of pp as text
            end if
        end repeat
    end tell
end run
```

```

set AppleScript's text item delimiters to "\""
set categoryItems to text items of categoryItem
if (count of categoryItems) > 1 then
    set category to item 2 of categoryItems
else
    set category to categoryItem
end if
set theType to ""
if details contains "Kartenzahlung" then
    set theType to "Credit Card"
end if
if details contains "Überweisung" then
    set theType to "Transferral"
end if
if details contains „Standing Order" then
    set theType to "Dauerauftrag"
end if
if amount > 0 then
    set theType to "Debit Entry"
end if
set AppleScript's text item delimiters to ""
set nl to theTitle & " " & details & " " & theDate & " " & amount & " " & ac-
countname & " " & category & " " & theType & "
"
        set outputText to outputText & nl
    end if
end repeat
end tell
return outputText
end run

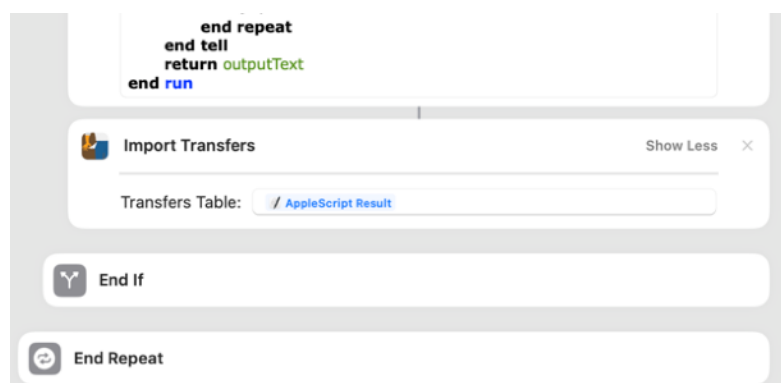
```

First we get the parameters from MoneyInput and store them in explicit variables. Then we talk to the application “MoneyMoney” and have all transactions for the `iban` account exported from the start date to the end date. We get the export result reported in `result` as a “plist”. This can go wrong, especially if the MoneyMoney database has not yet been unlocked. We then report this to the user with a request to unlock it.

If everything went well, we need to decode the `plist`. This could be quite time-consuming, but fortunately the “Property List Suite” from “System Events” can help us here. Although this is not an actual application, we have to address it as such. First we create a `property list item` from the result of the query from MoneyMoney. Then we get the `transactions` from them and loop over all of them. First we make sure that the `transaction` is really booked and get the fields we need for Himp.

The rest is pretty simple. The fields are sorted slightly differently in MoneyMoney than in Himp. The account name may be different in MoneyMoney than in Himp; the IBAN is crucial to identify the correct account.

MoneyMoney also has categories, but only in two levels: main category and subcategory. If a booking only has a main category, only that will be displayed; For subcategories, it is always output in the format “main category\subcategory”. That's why we have to



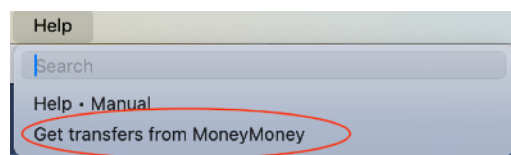
change AppleScript's text item delimiter to "\"" again to break this down and only pass the last word to Himp.

Then we build the line for Himp with tab-separated parameters, don't forget the character for a new line at the end (\n) and append it to our outputText. Then when we have finished all the loops, we pass the outputText to the next action in the shortcut.

Wow, that was pretty intense, wasn't it? This would certainly look different for another banking system, but we can use this example as a template.

Then we pass the AppleScript result to "Import Transfers" from Himp and everything will be fine.

The Mac version of Himp also provides this example as a file for integration with a banking system. You can add this shortcut to your own shortcuts under "Help" and modify it.



## 6. The Analysis View

So now we come to the analysis view. It makes the most sense on the Mac; that's why it's the default view here. But first we need a few bookings to make the view a little more interesting. Here are the test bookings:

Title	Date	Amount	Account	Category	Transfer Type
Electricity	01/03/2024	-\$78.00	XY Bank	Electricity	Direct Debit
Vegetables + Bowl	01/16/2024	-\$12.45	XY Bank	Household & Groceries	Debit Card
Rolls	01/27/2024	-\$3.40	Cash	Groceries	Cash Payment
Soap	01/27/2024	-\$2.20	Cash	Drugstore	Cash Payment
Electricity	02/01/2024	-\$78.00	XY Bank	Electricity	Direct Debit
ATM	02/17/2024	\$200.00	Cash	Others	Cash Payment
Credit Card xyz	02/19/2024	-\$200.00	XY Bank	Others	Credit Card

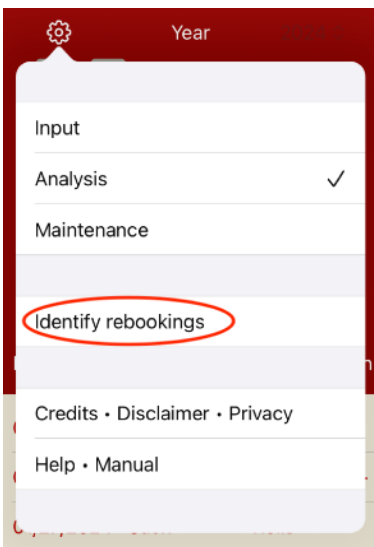
When we enter the analysis view after entering these bookings, we see something like this (next page):

The upper red area is used to control the analysis, while the bookings found are displayed in the lower half of the view. The first two lines in the control part are used to specify the analysis period. Using the drop-down list next to Year, we select the year or "All" years when we click or tap on the "X" on the right. The same applies to the month below. We can use the arrows to the left of the month (◀ and ▶) to scroll through the months or years if Month is set to "All".

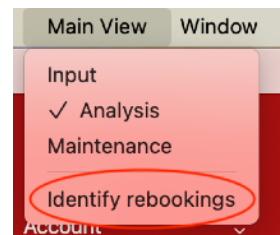
In our example we see all bookings from 2024.

Date	Account	Title / Foreign A...	Details	Transfer Type	Category	User	Group	Amount
01/03/2024	XY Bank	Electricity		Direct Debit	Electricity	MeMyselfA...		-\$78.00
01/16/2024	XY Bank	Vegetables + Bo...		Debit Card	Household...	MeMyselfA...		-\$12.45
01/27/2024	Cash	Rolls		Cash Paym...	Groceries	MeMyselfA...		-\$3.40
01/27/2024	Cash	Soap		Cash Paym...	Drugstore	MeMyselfA...		-\$2.20
02/01/2024	XY Bank	Electricity		Direct Debit	Electricity	MeMyselfA...		-\$78.00
02/17/2024	Cash	ATM		Cash Paym...	Others	MeMyselfA...		\$200.00
02/19/2024	XY Bank	Credit Card xyz		Credit Card	Others	MeMyselfA...		-\$200.00
<b>Sum</b>								<b>-\$174.05</b>
Balance								

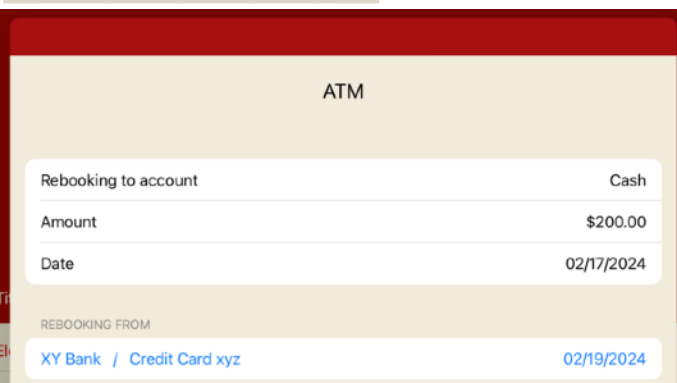
## 6.1. Identify Rebookings



But first we want to clear things up a bit. Above we described that we had gotten €200 for our wallet at the ATM. The amount was then debited from the account two days later. So we now have two bookings, a credit and a debit, with the same amount and a different sign. This could falsify our statistics somewhat, since the “income” on our cash account was not real income but just a transfer.

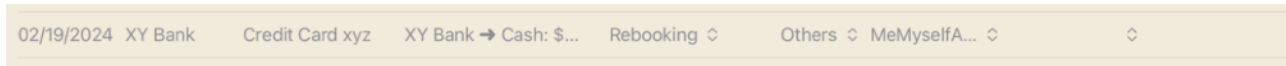


This is where Himp helps us with the cleanup. In the gear menu on iOS and in the “View” menu on macOS, there is the “Identify rebookings” function in the analysis view if we have not selected a specific account here. This identifies candidates for bookings that could belong together (different accounts, same amount but different sign, within a few days). This is what we want to do:



OK, a dialog appears with the incoming payment and a list of outgoing payment

candidates that would be suitable as a rebooking. This is just one booking (at XY Bank) and that is actually it. If we select it, the two bookings become just one and in our view for “All Accounts” (see below) the amount disappears (the two bookings cancel each other out).

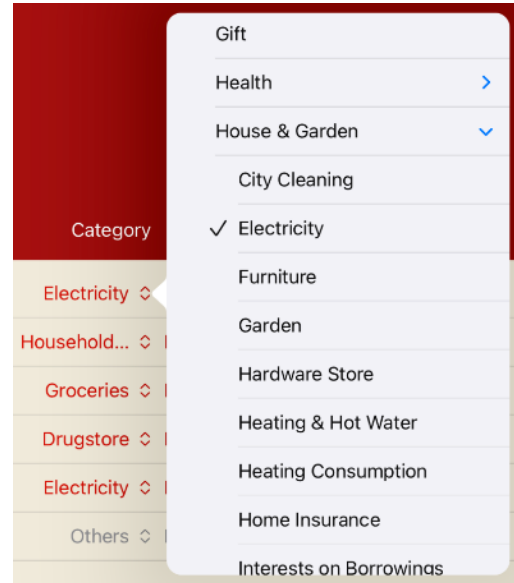
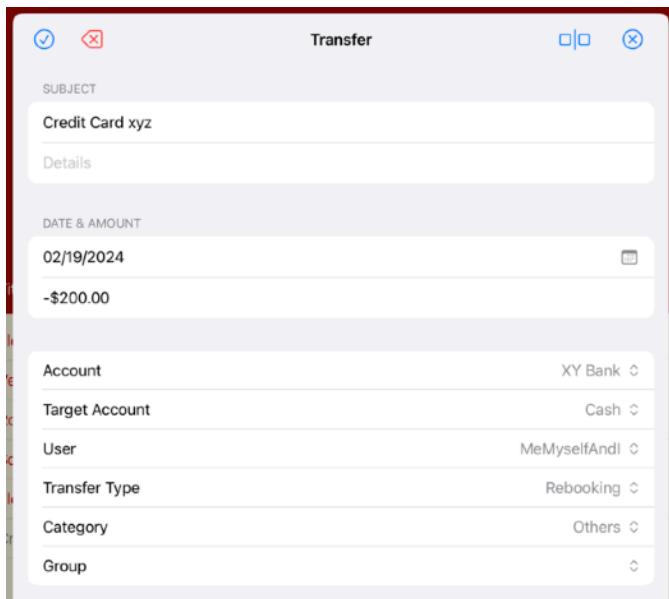


In addition, the color of the booking becomes a drab gray.

## 6.2. Bookings in the Analysis View

In the analysis view, we can quickly change the classifications of a booking (transfer type, category, user and group) using the drop-down lists.

If we want to change more, we can simply click or tap on the booking outside the drop-down lists and come back to the familiar details window from the maintenance view.



When the transfer is a rebooking, we can also see clearly what the “Target Account” is needed for.

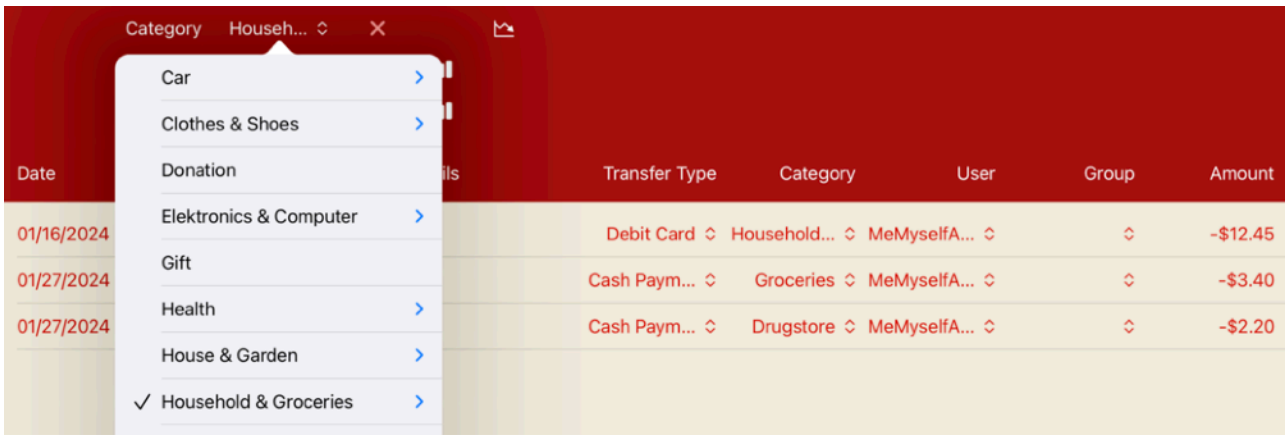
At the bottom right we see the total of the bookings listed and the balance of the currently selected account (here: XY Bank) at the end of the analysis period.

Sum	-\$368.45
Balance	\$745.34

## 6.3. Control of the Analysis View


The upper part of the analysis view is used for control. We have already gotten to know “year” and “month”. This determines the analysis period. The drop-down lists below are used to further filter bookings. So we can look at bookings from a specific account or a specific category. So if we select “Household & Groceries” under “Category”, we will see all bookings in the selected period that belong to this category or one of its subcategories. We see three bookings here with the categories “Household & Groceries”, “Drugstore” and “Groceries”.



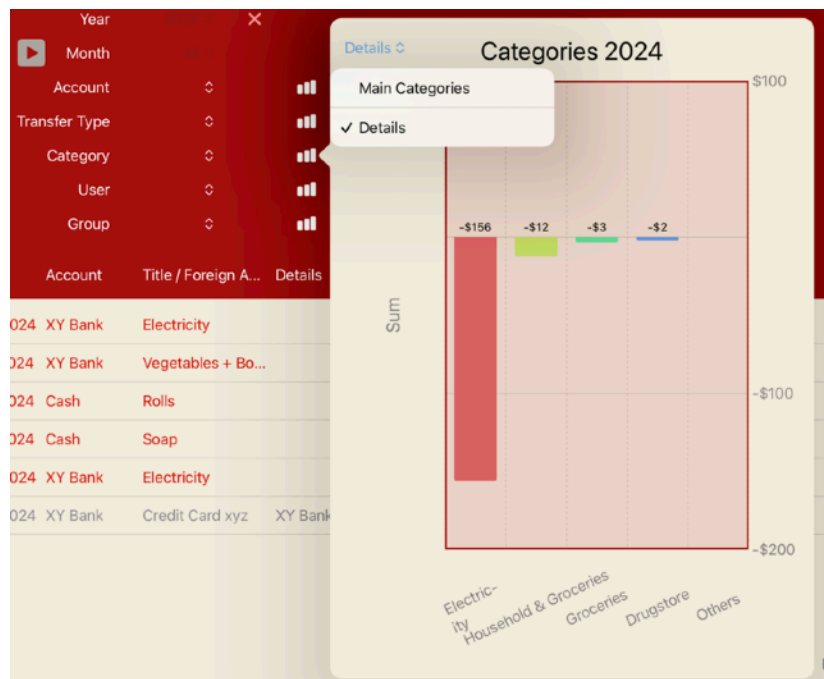



The “X” to the right of the selection deletes the corresponding filter again. We want to do that now.

### 6.3.1. The Distribution Graphs

Because the sign to the right of it  leads to the graph of the distribution of the corresponding classification. Let’s tap or click on it.

Oh, a distribution graph of the categories of our bookings in 2024 appears. For the categories, we can still choose whether we only want to see the main categories or all subcategories. Since we only have so few bookings in Himp, the detailed view is more interesting.



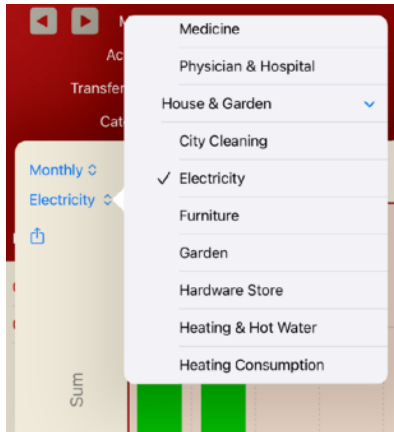
We can export the values in the graphs as a text file for further processing using the character “”. If we click or tap on a category of this graph, the corresponding category will be selected and we will go directly to the history graph. We want to do that and click or tap on the first column “Electricity”.

### 6.3.2. The Trend Graphs

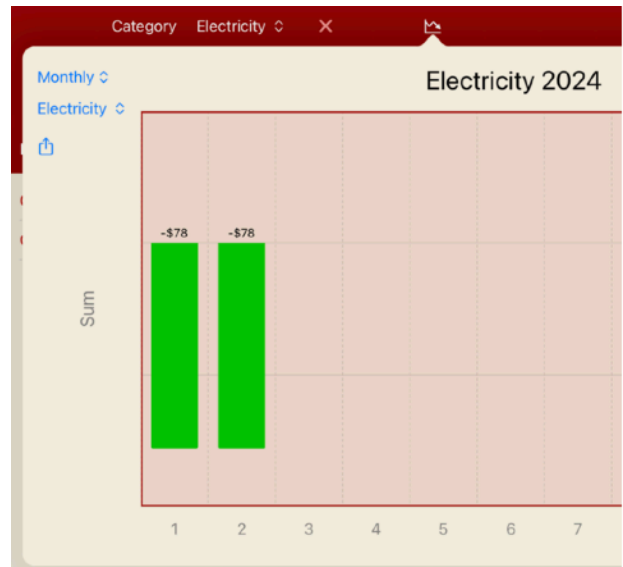
This brings us directly to the trend graph of the corresponding criterion (here the “Electricity” category). There we can choose whether we want to see the monthly trend in the current year or the annual trend over all years.

If we have set a filter, we can always access the corresponding trend graph using the “📈” symbol. We can also quickly change the category using the drop-down list at the top left of the trend graph.

We can also overlay several trends using the symbol “⊕” at the top right of the graphic.

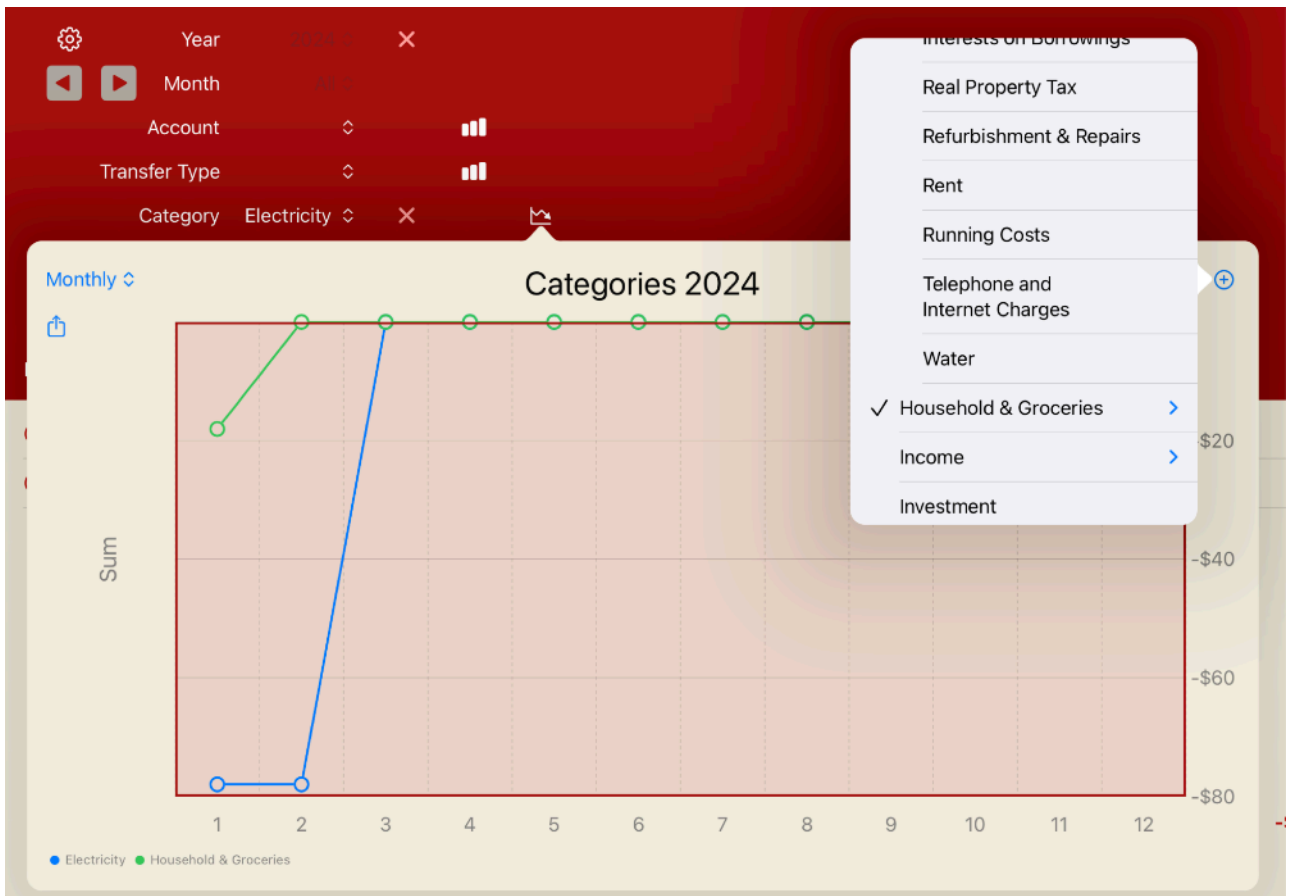


They are then displayed with different colors as a line graphic.



The legend of the graphic can be found at the bottom left. If we click or tap on a criterion again in the overlay selection, it will be deselected.

When we tap or click on a column of a single trend graph, the corresponding analysis period is set and the graph is closed.



---

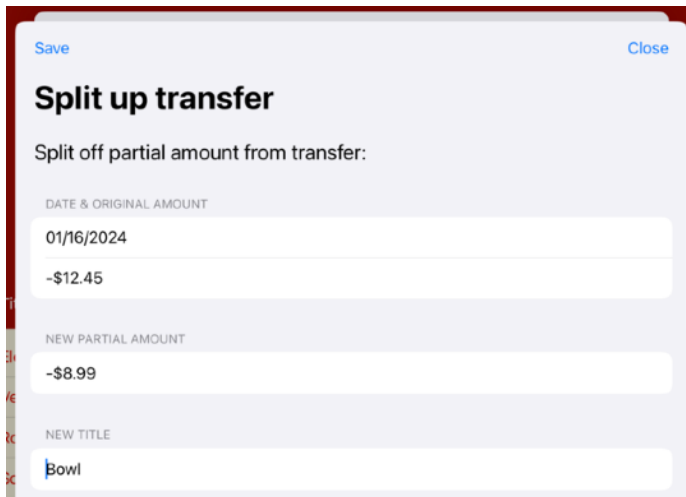
## 6.4. Split Up Transfer

In the details view of a transfer we can also split it into two bookings. In our example, we had bought vegetables and a bowl in a supermarket. This actually belongs in different categories

and we have managed to assign this booking to the common main category. But we can also split the transfer.

To do this, we click or tap on it will be taken to the details view. We can split off a partial booking using the symbol “□□” at the top right.

Of the \$12.45 for the entire purchase, \$8.99 was for the bowl. We enter this as a partial amount (with a negative sign because it is an expense) and also adjust the title.



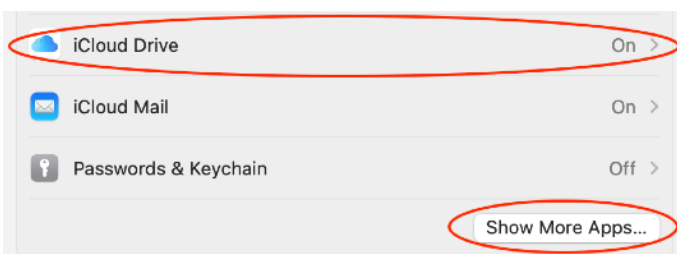
And now we have two bookings that we can assign to different categories: “Groceries” and “Household Goods”. The other classifications remain the same.

01/16/2024	XY Bank	Vegetables	Debit Card	Groceries	MeMyselfA...	◇	-\$3.46
01/16/2024	XY Bank	Bowl	Debit Card	Hosehold...	MeMyselfA...	◇	-\$8.99

## 7. Synchronization on multiple Devices

By default, Himp syncs data between multiple devices. The following requirements must be met:

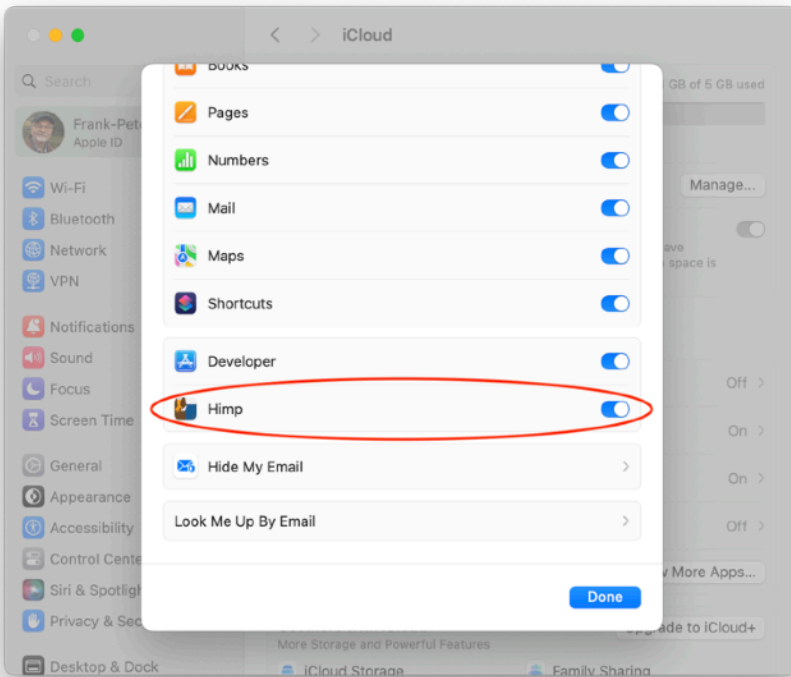
- All devices must be signed in with the same Apple ID.
- The iCloud Drive must be switched on, even if Himp does not store any data directly on the iCloud Drive.
- Himp must be allowed to use iCloud on the devices.



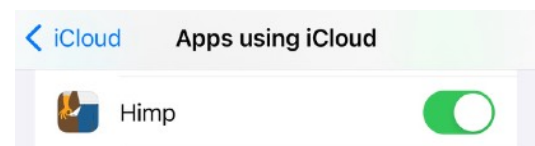
On the Mac we find these settings under “System Preferences” in the first line for the Apple ID and then under “iCloud”.

The settings for using iCloud through Himp are then listed under “Show More Apps...”. Here we can also prohibit the synchronization of Himp on this special

device.



The settings look very similar on iOS. Here too we find the Apple ID at the top in the “Settings” and the use of iCloud Drive under “iCloud”. Permission to use iCloud by Himp then follows under “Show All”.



So that's it. Have fun with Himp.