

Transactions Page

Inspiration

A pinterest filter on table designs: [77 Best Tables UX Design images in 2020 | Dashboard design,...](#)

[redacted]

Brainstorming

- This component/page is gonna have a *lot* of shit going on. Here's a rough list of what we need to fulfill all of the acceptance criteria:
- Actions to delete or update transactions.
- A date range picker to choose between daily/weekly/monthly/yearly/custom.
- A way to filter by type.
- Aggregate values for each type.
- Absolute/percentage change amounts for each type (relative to the last period).
- Total net worth at the current point in time.
- Absolute/percentage change in net worth (relative to the last period).
- A search input.
- Help info for the search input to specify how to do custom filters.
- An editing view for inline table editing.
 - I didn't really bother creating a design for this. It just requires subbing out text fields for inputs, so it's not like a lot of thinking needs to go into it.
- A button to create a new transaction.
- A way to sort by transaction field.
- Pagination.
- A mobile list view.
- A desktop table view.
- Empty state.

- Income/expense breakdowns.
- In terms of overall design, it looks we have a couple of decisions to make:
 - Should the table be card style or flush on the page?
 - We did card style in the first uFincs design, but I didn't really like it. Maybe best to experiment with both?
 - Although I really think that flush would be good, since it'd match with what we've been doing with the forms (white inputs on an off-white background).
 - How should the rows be styled?
 - Should the rows be zebra striped, have borders between them, have spacing between them? Or just nothing?
 - Having just done a basic design for the small transactions list (see below), I think I want to try out the 'spacing between them' approach.
 - The second big decision to make is how to lay out all of the filtering/searching stuff.
 - I think I'll start by designing a basic table and then just go from there.
- Here are the fields that we need to display:
 - type
 - date
 - description
 - amount
 - from
 - to

Editing View

- What if, instead of something like an 'Editing' toggle, users could just click on a cell and edit it's value?
- Additionally, hovering over a cell would indicate that the cell is editable.
- On top of this, tabbing into a cell would automatically change it to editing mode.
- And then, just have changes persisted on a debounce or something.
- The only downside to this is that users might not notice that the cells are editable.

- Maybe make all inputs for a row appear editable when hovering on the row?

Search Input

I fear that once we implement specific property searching, users (including myself) might want to build complex queries.

For example, I was originally envisioning the syntax as being just `from:chequing`.

However, I then thought how this would apply to Amounts. Obviously, `amount:200` is *valid*, but it isn't exactly useful as something like `amount<200`.

- **Problem #1: We might need more operators.**
- Potential solution: Don't start using `:` as the separator but do use `=` instead. This gives us the ability to branch out in the future.

Additionally, how do we chain together a bunch of conditions? AND or OR?

- **Problem #2: Condition chaining is ambiguous if not spelled out, and is limiting if we only allow one or the other.**
- Potential solution: Just give users the ability to explicitly chain them with AND and OR operators. lol

So how should we actually deal with these problems? I see two meaningful solutions:

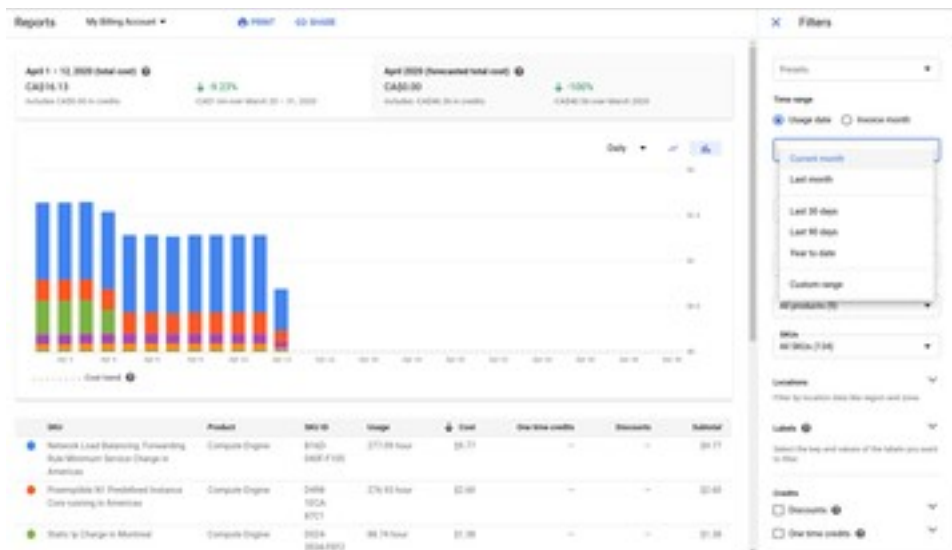
- i. Build out our own custom syntax for building queries and keep everything textual.
- ii. Build out an 'advanced search' solution akin to something like [this query builder](#).

Just some ideas for the future. I still want to implement the basic `"from=chequing"` syntax, but I know that'll throw us down a rabbit hole of custom search stuff.

Date Range Filter

- I think the first thing to figure out is how to display the date filter.
- [redacted]
- I quite like how they give the option to do an overarching filter ("Last 30 days", "This month", etc) and then allow to make it more specific (by changing the ends of the range).
- I think it's also worthy to note that this approach inherently gives custom date ranges as a first class argument.
- It's also worth pointing out that they don't give any default overarching filters for less than the current month. That is, there's no "today" or "this week" view. I think this makes sense for their product since it's more so about looking at large longer-term trends than very short-term trends.

- Now, whether that same kind of logic applies to uFincs is up for debate.
 - For example, how often would users want to see the list transactions of one specific day? Probably very rarely. One specific week? Probably more likely than a day.
- When I was thinking of date filtering for the transactions list/table (and really, everything else in uFincs), I was thinking of just doing default [Daily/Weekly/Monthly/Yearly] views, but looking at how does it has me reconsidering.
- Now that I'm thinking about alternatives, the Billing section of GCP springs to mind...



- They stick "Custom range" as a separate option, instead of supporting it as a first class option like .
- Now I remember why I wanted to do the whole [Daily/Weekly/Monthly/Yearly] thing: so that users could scrub between values in the interval.
 - That is, currently, when looking at Monthly Transactions, users can use arrows to switch between months.
 - My thinking was that users could do that switching but with other intervals.
 - But again, I don't know how useful it would be to have daily or weekly intervals.
 - So now I guess it comes down to a question of what provides the user more value:
 - The ability to quickly switch between values in an interval, or
 - The ability to specify custom ranges quickly.

- Because, even if we do this interval thing, there's nothing stopping us from allowing the user to specify a custom range. The only thing that we'd have limit is the switching between values in a custom range, since that could get out of hand.
- Although, I suppose there isn't any good reason why we couldn't have the best of both worlds:
 - Provide the user with an interval.
 - Then provide them the start and end dates of the range, like does.
 - Of course, if the interval is daily, then there'd only one end of the range...
 - Then provide them the switching capability.
- This way, the user can still switch between values in the interval, but still opt to change either end of the range. This would change the interval to 'Custom range' and prevent backwards/forwards value switching.
- Of course, we *could* support backwards/forwards value switching in a custom range - the interval would just be whatever the length is between the start and end date.
 - Whether that would be more or less confusing for the user than disabling the switching is up for debate.
 - But since we're all in on this, *might as well support it, amirite!?!*
- Just a note: If the user changes the date range manually, then it should only switch from whatever interval they had previously to 'Custom range' only if the new date range doesn't conform to an existing interval.
 - For example, if the user is on the 'Monthly' interval, but switches to viewing a week's worth of data, then the interval should change to 'Weekly'.
 - Now obviously, this calculation is simple for weekly (7 days difference) or daily (1 day), but monthly is a bit more tricky. I'd say that it should only get the monthly interval if the user happened to choose the a date range with the start and end of a month. Otherwise, I can't just guess '28 days or 29 days or 30 days or 31 days'.
 - Actually, I think monthly can just be whether or not the date is the same and there is a month's difference. i.e., "March 13" and "April 13" is considered a monthly interval.
 - Additionally, the default value for 'monthly' should be *this month* and not *the last 30 days*.
 - But when users traverse to a past (or future) month, it should show the full month.

- Perhaps we can add a **setting** so users can choose between showing "up till today" or "the whole month".
 - Additionally, maybe users might want a **setting** for "today, back a month". That is, if today is "April 21", then monthly would mean "March 21 to April 21".
- I think this concept of "show this period first" should apply to all of the intervals.
 - That is, weekly starts off with this week (starting on Sunday), yearly starts off with this year (January 1st), and daily... doesn't need a special exemption.
- Just a **note**: We might have to reconsider this decision once recurring transactions are implemented, because what if users want to see the future/upcoming transactions for the month? There'd be no easy way for them to see it if the interval started at "this month, up till today".
 - Therefore, we might just want to show the whole month and differentiate upcoming transactions somehow.
 - I'm thinking maybe something like gray them out? Or put them at the bottom of the list? Or hide them behind a toggle of some sort?
 - Yearly also works out nicely at just 365 days.
- As far as design goes, I'm thinking we could just rip off design and then put some arrows on either end of the date range.

Date Range Inspiration

- [redacted]
- I quite like how all the options for preset date ranges are just presented as segmented buttons.
- The fact that the date range calendar thing is presented with a dropdown arrow seems to indicate that the entire range is editable in one calendar view, as opposed to something like where each end of the range is independently editable.
 - I don't know which of these approaches is better. Probably
 - [redacted]

All Time?

Something I just thought of was should we give a user the ability to specify "All time" as one of the 'intervals'?

Obviously, it doesn't make much sense from an 'interval' perspective, but it's probably useful from a user perspective.

We can get around it not being an interval by disabling the backwards/forwards buttons.

Giving the user the ability to quickly access all of their transactions means that they can apply searches in a more effective manner. Especially if we decide to implement more advanced search abilities.

Switching between intervals?

When switching between intervals, the naive logic should be as follows:

- i. When going from a bigger to a smaller interval, align the smaller interval at the end of the bigger interval.
 - For example, going from Monthly of Mar 1 - Mar 31 to Weekly would set the interval to Mar 24 - Mar 31.
- ii. When going from a smaller interval to bigger interval, align the bigger interval at the start of the smaller interval.
 - For example, going from Weekly of Mar 1 - Mar 8 to Monthly would set the interval to Mar 1 to Mar 31.

However, this naive logic doesn't allow for pure switching interval switching. That is, if I go from one interval and then back, I expect to be where I was before I changed anything.

But, taking one of the above examples with the given logic, if we go Monthly Mar 1 - Mar 31, then Weekly to Mar 24 - Mar 31, then back to Monthly, we'd end up with Mar 24 - Apr ..24?.

As such, we almost have to take the start of the interval when switching, regardless. However, this then might confuse the user.

For example, if I default to the Monthly view when going to the Transactions page, then want to change to the Weekly view, I *expect* to get the transactions for the latest week in the month.

So maybe, instead of having naive logic that just always aligns to one end of the interval or another, we need some other logic. For example, maybe switching to Monthly always just switches to the full month of whatever month the user is currently in.

But this, again, breaks pure switching. So maybe pure switching is important to the user?

So maybe the logic is just this:

- Daily: Take the last day in the last interval.
- Weekly: Align the end with the end of the last interval.

- **Monthly:** Set the interval to the full month of whatever month the last interval was in.
 - If the last interval spanned two months, take the latest month.
 - If the month is the current month, set the interval only up to today.
- **Yearly:** Set the interval to the full year of whatever year the last interval was in.
 - If the last interval spanned two years, take the latest year.
 - If the year is the current year, set the interval only up to today.

Income/Expenses Summaries

(from Slack)

We could put a list/table tallying up the totals for each income/expense account.

Additional feature: Clicking on an account filters the current view to show only transactions of that type. In fact, we could split income and expenses to two separate lists and then add little progress bars showing how much of each account takes up the total amount.

Idea was inspired by something I read in.

Post Design

The account bars were designed standalone, but I came across this design where the bars are overlaid on a total bar:

[redacted]

I kinda also like this... will have to try both.

Mobile Design Brainstorming and Iterations

Here's a copy of the checklist, but specifically for the small design.









- Actions to delete or update transactions.
- A date range picker to choose between daily/weekly/monthly/yearly/custom.
- A way to filter by type.
- Aggregate values for each type.
- Total net worth at the current point in time.
- Absolute/percentage change in net worth (relative to the last period).
- A search input.

- A button to create a new transaction.
- Pagination.
- Empty state.
- Income/expense breakdowns.

Here's the things that aren't happening on mobile:

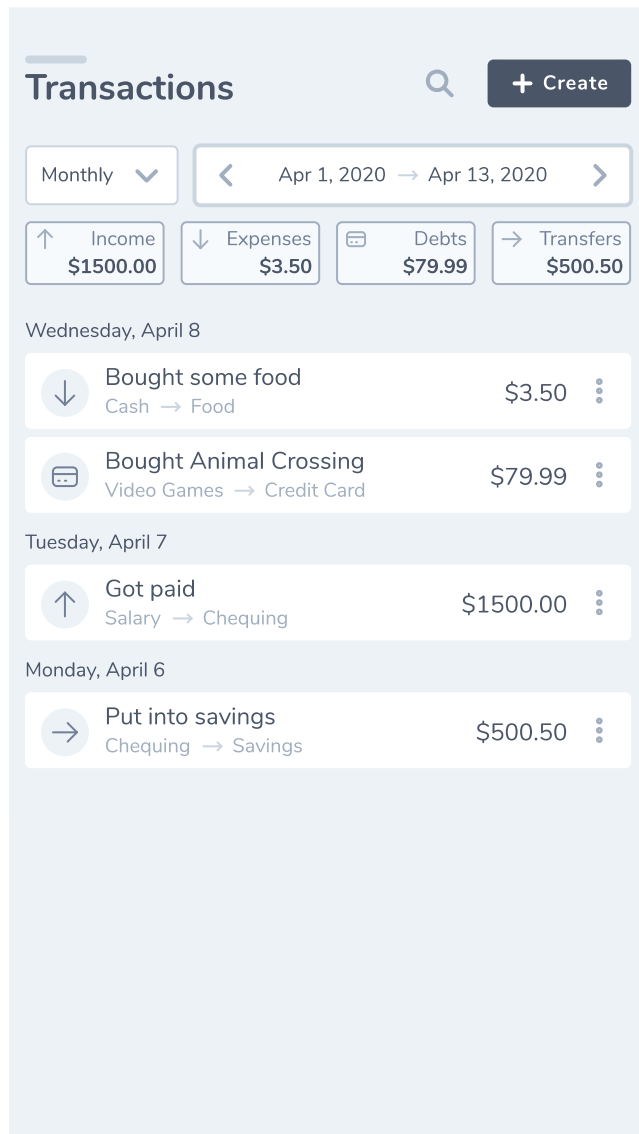
- Absolute/percentage change amounts for each type (relative to the last period).
 - Already ruled this out for not having enough space.
- Help info for the search input to specify how to do custom filters.
 - I don't think we'll have the room for this on mobile.
- An editing view for inline table editing.
 - Yeah, this ain't happening on mobile.
- A way to sort by transaction field.
 - Sorting is definitely not happening on mobile.

Here's the first iteration of the small design, just the list:

Transactions			
Wednesday, April 8			
	Bought some food Cash → Food	\$3.50	
	Bought Animal Crossing Video Games → Credit Card	\$79.99	
Tuesday, April 7			
	Got paid Salary → Chequing	\$1500.00	
Monday, April 6			
	Put into savings Chequing → Savings	\$500.50	

So now we need to figure out to display all of the filtering and aggregation stuff...

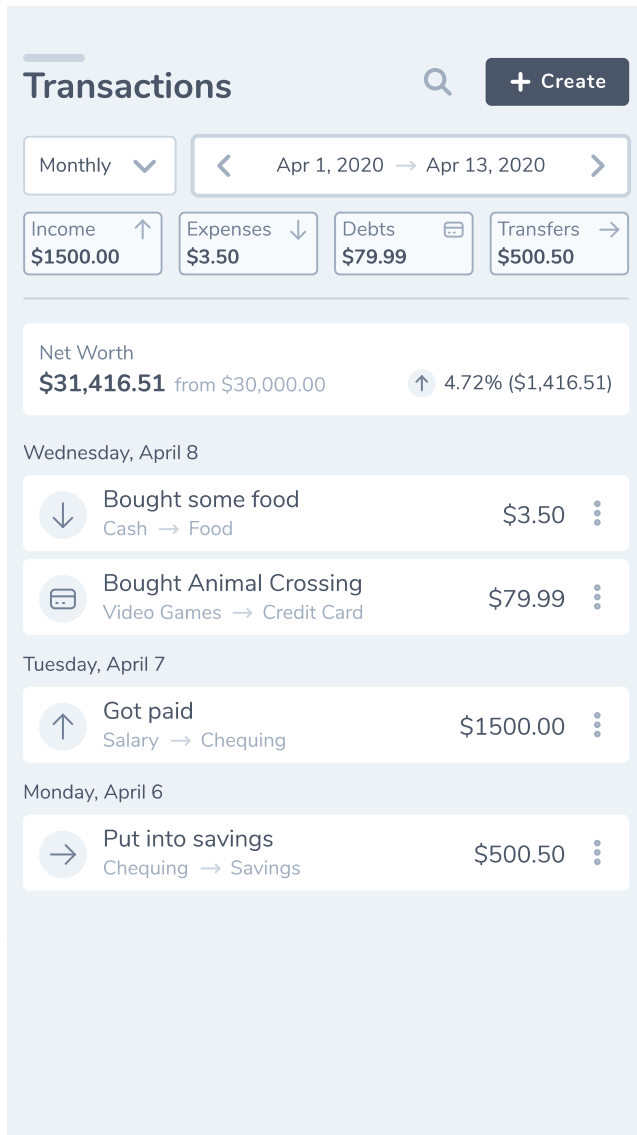
Here's an update with type filtering, search, create button, and date range stuff:



Note that the transaction type icons have been changed to being only 1px thick. This is because, at 2px thick, they were presenting themselves as too important on this small canvas. Thus, to get their visual hierarchy down, the thickness was reduced, along with the color.

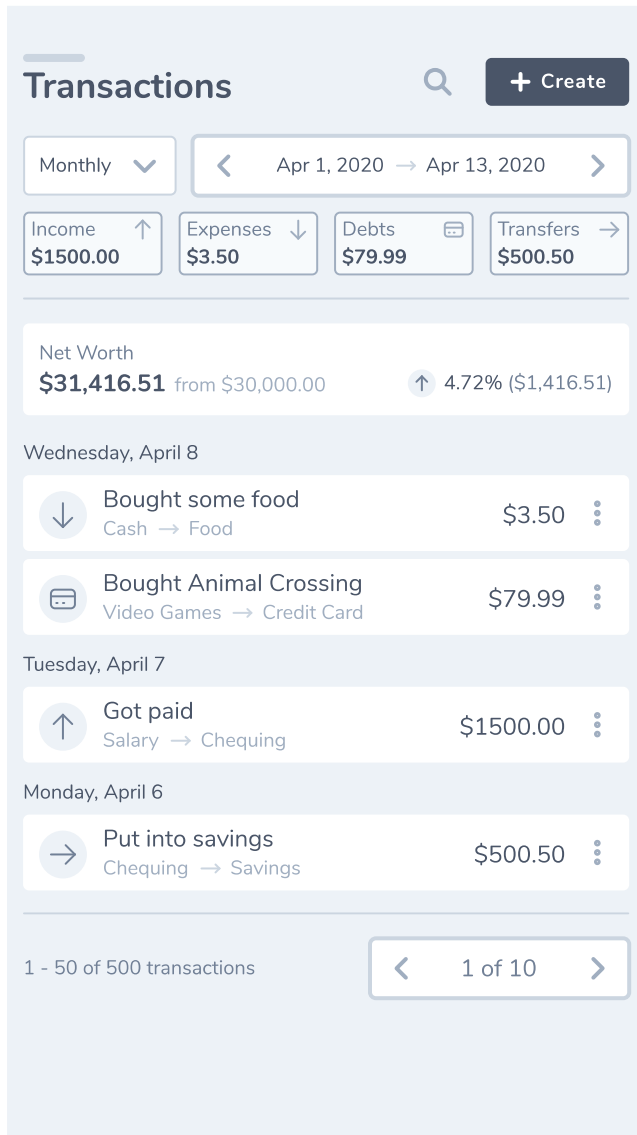
So now the biggest things left are how to fit in the net worth (and abs/% changes?) and pagination.

Another update, this time with the net worth display:



- One thing that's noteworthy here is that the alignment of the elements in the Type cards has been swapped around; now the text is on the left and the icon is on the right. This is mirror how the Net Worth section is laid out.

Another update, with pagination:



We're getting pretty close! All that's left is to put together a semi-sane empty state.

I think this is the final iteration for the mobile design:

Transactions

Create

Monthly

< Apr 1, 2020 → Apr 13, 2020 >

Income
\$0.00


Expenses
\$0.00

Debts
\$0.00

Transfers
\$0.00

Net Worth

\$30,000.00 from \$30,000.00



Welp, no transactions here

Doesn't look like you've made any transactions during this time period.

How about creating one now?

Create Transaction

Boom! Empty state. Everything on the checklist has now been completed.

Update/Delete Actions

- What I'm thinking for this, instead of a lame old dropdown, is to slide the row/list item in to left and expose the actions on a plane beneath the row.

Create and Search

- For these two actions on mobile, I'm thinking of putting them into the header alongside the title, kinda like what we do for the "New Transaction" form with the close button.
- Instead of a search bar, we'll use just the search icon and then have the input expand somehow.

- I'm thinking the input could expand either beneath the header or to fill the rest of the space to the right of the title and push the Create action out.
 - Hmm, I don't think having it expand to push the Create action out would be a good idea, because then how would a user create a new transaction while also searching?
 - Maybe it could just take up the extra header space *without* pushing out the Create action?
- Now, do I want to use "New" or "Create" for the 'create' action?
 - Well, I used "New" in the original uFincs, but I've seen some sites (like DigitalOcean) use "Create".
 - Just as a note on mobile, I'm going to omit "Transaction" from the button label (i.e. on mobile it'll be "New" or "Create", but on desktop it'd be "New Transaction" or "Create Transaction").
 - Well, "New" is shorter... so it gets some points for that.
 - But "Create" is the action at the end of the "New Transaction" form...
 - But the title of the form is "New" Transaction...
 - Hmm...
 - For reference, here's a related discussion on this topic: [How to phrase button labels?](#)
 - I think what makes "Create" beat out "New" is that it is a *verb*. And based on [this](#), buttons should use verbs
 - **Decision:** Use "Create".

Net Worth View

- Something I want to steal from is their use of.. metrics on their charts. Specifically, how they present the current value, add "from \$XXX" to indicate what's it changed from, and then show a percentage change as well.
- Obviously, I also want to add on the absolute change in addition to the percentage change.

Here's an example:

- [redacted]
- So for the Transactions page view, we can show the total net worth at the end of the given period, "from" the total net worth at the start, with the delta in percentage and absolute.

- Now, how we visually present this information on the mobile view is a bit more problematic, because of how much *stuff* is already clogging up the space above the transactions list.

Here's something that came to my mind when thinking of this design:

[redacted]

This sidebar is from one of the old Tailwind UI designs. Basically, what I was thinking was just sticking another box below what is currently the type filters, with the net worth information in it.

A little uninspired, but I think it'll get the job done. Whether the UI will look too cluttered afterwards is up for debate. However, that might be fixable with the use of some dividers, like we did with the Transactions form.

A side thought:

- If we want to display the net worth as being calculated at the end of the given date interval, then I think -- going back to the discussion on whether "Monthly" should be "this month, up till today" or "this whole month, to include upcoming transactions" -- that the "Monthly" view should be "this month, up till today" by default.
- This is to have the net worth calculation actually reflect where it's at *today*, as a default.
- I think this is a good compromise, since users can still manually change the end date to the end of the month to set their upcoming transactions and what their net worth would be like assuming all those transactions are calculated into it.

Some relevant design inspiration:

[redacted]

Pagination

I think my favourite example of pagination (so far) has to be from this:

[redacted]

I like this pagination because it has the three things I look for in pagination:

- Obviously, the pages and a way to switch between them.
- The ability to change how many items are displayed per page.
- The total number of items.

Now, my thinking was, with the transactions list/table, we could probably get away with showing a fairly hefty number of transactions on one page. I was thinking at least 50, but maybe even as much as 100.

As such, the average use case probably won't ever see the pagination. Only in the case where the user is trying to view multiple months or years worth of transactions will this happen.

However, since there is the possibility (and we know how poorly tons of transactions performs wrt rendering), we need pagination.

But I really don't want to do anything too complicated with the pagination... cause it can be a real pain.

Here's some more examples of pagination styles:

[redacted]

In terms of simplicity, I must save I prefer #3/12 (#3 is more applicable to our subdued rounded style) because it doesn't require any fancy logic for what page numbers to display when. It does come with the downside of not being able to quickly jump to first or last page, or specifying a specific page to jump to, but I think those trade-offs are worth it for the simplicity.

- Technically, we could implement it so that clicking the middle portion of the pagination component (i.e. the "3 of 10" part) could bring up some sort of page picker, but I think we should just keep it simple for now.

Ultimately, the user is probably much more likely to filter the list some more (through types, search, or dates) than they are to browse through pages of transactions.

And since we think we can get away with displaying 50 or 100 transactions on one page, we probably don't even need the input to change how many are displayed per page. I mean, how often is the user gonna want to display *fewer* items?

And I think we can still show the total number of transactions.

Empty State

Here's some inspiration:

[redacted]

And some articles on empty states:

- [Empty states](#)
- [Empty States Design Best Practices](#)

I found a pretty decent graphic for the transaction list empty state:



Taken from [Illustrations | unDraw](#). Seems like a pretty good resource for humanistic graphics.

Now to make it grayscale to fit in with the grayscale designs...

Then we just gotta slap on some 'empathetic' text, add a call-to-action button, and presto! Empty state.

Designs

Grayscale

Transactions

[Create](#)

Monthly

Apr 1, 2020 Apr 13, 2020

Income
\$1500.00

Expenses
\$3.50

Debts
\$79.99

Transfers
\$500.50

Net Worth

\$31,416.51 from \$30,000.00

+4.72% (\$1,416.51)

Transactions

[Summary](#)

Wednesday, April 8

Bought some food **\$3.50**

Cash Food

Bought Animal Crossing **\$79.99**

Video Games Credit Card

Tuesday, April 7

Got paid **\$1500.00**

Salary Chequing

Monday, April 6

Put into savings **\$500.50**

Chequing Savings

1 - 50 of 500 transactions

1 of 10

Transactions

Create Transaction

Search all transaction properties



Daily

Weekly

Monthly

Yearly

All Time

Custom

< Apr 1, 2020 → Apr 13, 2020 >

Income ↑ \$1,500.00 from \$1,400.00 +7.14% (\$100.00)	Expenses ↓ \$3.50 from \$15.60 -77.6% (\$12.10)	Debts 📅 \$79.99 from \$50.00 +60.0% (\$29.99)	Transfers → \$500.00 from \$1,000.00 -50.0% (\$500.00)	Net Worth \$31,416.51 from \$30,000.00 +4.72% (\$1,416.51)
---	---	---	--	--

Transactions Summary

Date ▾	Description	Amount	From	To	
↓ Wed, Apr 15	Bought a whole metric ton of food for the quarantine.	\$10,000.00	Uncategorized Income	Uncategorized Income	✎ 🗑
↓ Wed, Apr 15	Bought a whole metric ton of food for the quarantine.	\$10,000.00	Uncategorized Income	Uncategorized Income	
↓ Wed, Apr 15	Bought a whole metric ton of food for the quarantine.	\$10,000.00	Uncategorized Income	Uncategorized Income	
↓ Wed, Apr 15	Bought a whole metric ton of food for the quarantine.	\$10,000.00	Uncategorized Income	Uncategorized Income	
↓ Wed, Apr 15	Bought a whole metric ton of food for the quarantine.	\$10,000.00	Uncategorized Income	Uncategorized Income	
↓ Wed, Apr 15	Bought a whole metric ton of food for the quarantine.	\$10,000.00	Uncategorized Income	Uncategorized Income	
↓ Wed, Apr 15	Bought a whole metric ton of food for the quarantine.	\$10,000.00	Uncategorized Income	Uncategorized Income	

1 - 50 of 500 transactions < 1 of 10 >

Transactions

Create Transaction

Search all transaction properties



Daily

Weekly

Monthly

Yearly

All Time

Custom

< Apr 1, 2020 → Apr 13, 2020 >

Income ↑

\$1,500.00 from \$1,400.00
+7.14% (\$100.00)

Expenses ↓

\$3.50 from \$15.60
-77.6% (\$12.10)

Debts 📅



\$79.99 from \$50.00
+60.0% (\$29.99)

Transfers →

\$500.00 from \$1,000.00
-50.0% (\$500.00)

Net Worth

\$31,416.51 from \$30,000.00
+4.72% (\$1,416.51)

Date ▾	Description	Amount	From	To	
↓ Wed, Apr 15	Bought a whole metric ton of food for the quarantine.	\$10,000.00	Uncategorized Income	Uncategorized Income	 
↓ Wed, Apr 15	Bought a whole metric ton of food for the quarantine.	\$10,000.00	Uncategorized Income	Uncategorized Income	
↓ Wed, Apr 15	Bought a whole metric ton of food for the quarantine.	\$10,000.00	Uncategorized Income	Uncategorized Income	
↓ Wed, Apr 15	Bought a whole metric ton of food for the quarantine.	\$10,000.00	Uncategorized Income	Uncategorized Income	
↓ Wed, Apr 15	Bought a whole metric ton of food for the quarantine.	\$10,000.00	Uncategorized Income	Uncategorized Income	
↓ Wed, Apr 15	Bought a whole metric ton of food for the quarantine.	\$10,000.00	Uncategorized Income	Uncategorized Income	
↓ Wed, Apr 15	Bought a whole metric ton of food for the quarantine.	\$10,000.00	Uncategorized Income	Uncategorized Income	
↓ Wed, Apr 15	Bought a whole metric ton of food for the quarantine.	\$10,000.00	Uncategorized Income	Uncategorized Income	

1 - 50 of 500 transactions

< 1 of 10 >

Income Summary

Salary

\$1,500.00

Interest

\$500.00

Expenses Summary

Food

\$1,200.00

Video Games

\$600.00

Subscriptions

\$300.00

Groceries

\$150.00

Electronics

\$150.00

Net Income

-\$400.00

Colored

Transactions

Monthly

Apr 1, 2020 → Apr 13, 2020

Income
\$1500.00

Expenses
\$1500.00

Debts
\$1500.00

Transfers
\$1500.00

Net Worth
\$31,416.51 from \$30,000.00 +4.72% (\$1,416.51)

Transactions

Summary

Wednesday, April 8

↓

Bought some food
Cash → Food

\$3.50

Bought Animal Crossing
Video Games → Credit Card

\$79.99

Tuesday, April 7

↑

Got paid
Salary → Chequing

\$1,500.00

Monday, April 6

→

Put into savings
Chequing → Savings

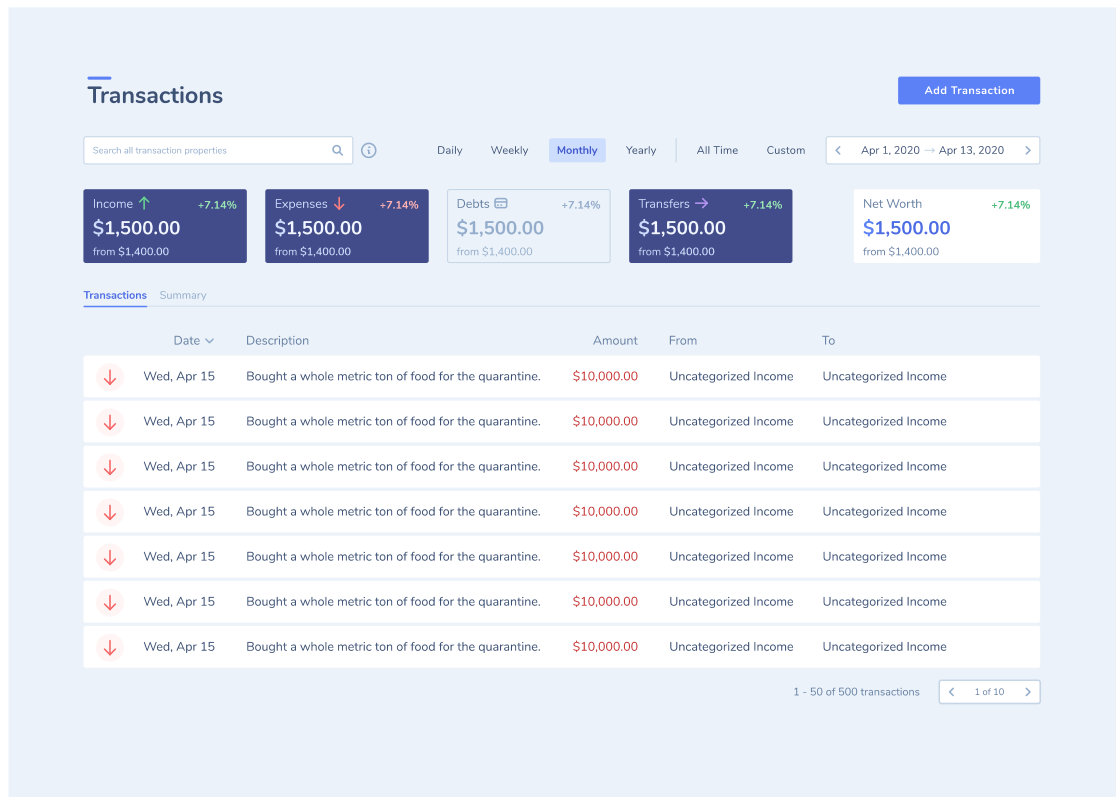
\$500.50

1 - 50 of 500 transactions

<

1 of 10

>



Post Design Ideas

Here's some things I've thought of since creating the above version of the design.

Type Card Carousel

I came across this:

[redacted]

I like, how on the mobile version, they have a swipeable list of cards. I don't know how well that works *on the web*, but I like the idea regardless (i.e., is swiping on the web even a thing?)

What we could also do is have our small type cards by default and then a button to expand them to the full size ones with all the information, but with the cards in a swiping carousel. That way users can have the best of both worlds.

Progress Bars

Also came across this:

[redacted]

They use overlaid bars to represent percentages instead of underline bars. They also actually show the percentage. Maybe showing the percentage would be good...

Could we show the percentage *inside* the bar? Or should it just be at the end like the below design?

Also this:

[redacted]

Another take on the percentage bars.

Type Card Shadows

Also this:

[redacted]

Might want to consider adding an inset drop shadow to the selected Transaction type cards.

Triangles instead of +/-

Also this:

[redacted]

I wonder if it's better to use these triangles for indicating gains/losses than plus/minus signs.

Collapsing the Transactions Table

What if we made the Transactions Table *really* responsive by having it collapse columns together as the screen size moves towards small?

For example, when the table gets small enough, we could collapse the From and To columns by removing the columns altogether and putting the From → To line beneath the description.

There are other optimizations we could make, but that's the main one I thought of.

- Oh, just thought of another one! We can transition the actions to be an overflow icon so that the actions are hidden under the row, like on mobile.

This would help fit the Transactions Table on something like the Account Details page, since the Details portion doesn't get the full screen width.

Other Pagination Ideas

Here's some alternative pagination ideas from [Tailwind UI](#).

Styling the Type Filters

I've been agonizing over how to style the type filters to make them not look like complete garbage when using colors.

When I converted them 1-to-1 from grey to colored, there was just too much going on and they looked really bad.

I've come across two kinds of ideas that I like so far.

The first is full colored backgrounds with white text (and shadows):

I tried to replicate this kind of gradient background, but I feel our colors aren't the right hue to get the effect right. Or I might just need more judicious shadows.

And the second is just a border (and shadows):

I think the second one would work better at conveying that these elements are clickable, since they are all selected by default.

Here's some others: