Transactions Import

Inspiration

[redacted]

Brainstorming

Here's what we need from the Transactions Import view:

- A way to choose the account to import to.
- A way to choose the CSV file to import.
- An indicator of just how many transactions might have been found after uploading the CSV file.
- A way to match CSV columns.
- A way to create a profile of matched CSV columns.
- A way to adjust transactions.
- A way to bulk adjust transactions.
- A final review of the transactions to import, along with the other information (account name, balance change, number of transactions, etc).
- A way to navigate forwards and backwards between steps.
- A display of the current step and the overall progress through the import process.
- Short explanations for each step.

I think it will still suffice to have the 5 steps we had originally:

i. Choose account

ii. Choose CSV file

iii. Match CSV columns

iv. Adjust transactions

v. Finish import

Also, random other thought... would it have made more sense for `Import Profile` to be `Import Format`? ... shit

- I've made the decision to only expose the term "format" to the user. "Profile" should not be used. As such, the model name "Import Profile" is now **technical debt**, since it *should* get renamed to "Import Format".

Desktop Checklists

Something I've been debating about for the desktop layout is how (or, more precisely, whether or not) to constrain width. Currently, we have a fixed width for the entire import interface (as does for reference). As such, all elements stretch to fill the width and are consistently placed (like the back/next buttons).

However, this leaves some steps looking particularly stupid, namely the Choose Account and Choose CSV File steps, because there is lots of leftover whitespace on the side.

As such, I was thinking about if we just did a centered kind of layout where everything just takes up as much it needs (or at least, not an excessive amount). If we go with this approach, and use the horizontal stepper at the top of the layout that *is* a fixed width, then that allows us to go both inside *and* outside the bounds of the width of the stepper. This means the stepper only needs to be as large as it needs to be, while allowing steps like Adjust Transactions to spread out with as much space as possible.

Obviously there's a point where more space starts looking ridiculous (e.g. 4k), but I think this kind of layout is more responsive both upwards and downwards. Let's get it a shot.

Really, the only downsides are that the back/next buttons aren't in the same horizontal position between steps. However, I think the pros outweigh the cons here.

BTW, speaking of the back/next navigation buttons, I don't think it makes as much sense as it does on mobile to have duplicate back/next buttons in the header and footer of the layout, since users are working with bigger screens. So we'll go with just the footer buttons.

UPDATE: So having designed the first three steps (and some of the fourth) with this centered approach to the layout, I'm having second thoughts. On the one hand, it *is* obviously better for responsiveness. Additionally, it makes the readability of the step instructions better, since they have a max width.

However, it just looks *so weird* having the instructions at a limited width (with the navigation buttons aligned on that width) but then have something like the New Import Profile form expand outside that width. I think adding the dividers between the sections (i.e. the extra wide table and the navigation buttons) helps a bit, but still...

And honestly, I can't really think of any better alternatives. If everything's all left aligned, then that 1. also looks weird, but more importantly 2. sets a hard limit on the width of the content, which is what we're trying to avoid.

If we make everything in the same step the same width, then we run into making the instructional message too wide and reducing readability.

I think we just keep going on with the centered layout and re-evaluate once all the steps have been designed.

UPDATE: So having designed all five steps now, I think I've come up with some alternatives to the entirely centered layout.

i. Use a large-ish sidebar to hold the title, steps, and instructional messages. Then the rest of the space can be dedicated to the content. Kinda like this inspiration I had found for the onboarding flow:

   - Thinking on this more, I don't know how well this kind of "top to bottom" sidebar design works *inside* an app (as opposed to like above, where it's before the app), notably one that will probably have some sort of navigation at the top.

   - Eh, it'd probably be fine.

   - Might as well throw something together and see what it looks like.

   - The only thing I don't know about is how the stepper should be laid out. As it currently stands, it's obviously too wide to fit in a sidebar. So... make it vertical? Change it to the style of the mobile stepper?

   - So I just tried it out, but I'm not feeling it. Like, yes using a sidebar makes it look better, but I feel like it sacrifices too much in usability to be worth it.

     - Mainly as a result of having the user jump all over the place when scanning for information. If the steps and the step instructions are in the sidebar, then that means the user has to scan the sidebar first and then the main content. Whereas with the centered design, it's just a straight shot down.

     - I don't know. Since *I'm* not really the target of the import process, I don't really care. However, I'm sure once this thing is in other people's hands, they'll have their own opinions on it.

ii. Move the instructional message into the bar with the steps.

   - OK, thinking more on this, maybe less of a good idea.

Choose Account Step Checklist

- Need the title

- Need a back button

- Need to design the step indicator (with all step numbers and titles, and the current step)
- Need to design the Back/Next navigation buttons
- Need an instructional message
- Need inputs for choosing the account type and account

Choose CSV File Step Checklist

- Change the step indicator
- Change the instructional message
- Need a button to choose the file
- Need a message for the file name
- Need a message for the number of transactions found

Match CSV Columns Step Checklist

- Change the step indicator
- Change the instructional message
- Need a section for creating a new import profile
  - Need an input for the profile's name
  - Need a picker for setting the uFincs field of each column
  - Need to show sample data for each column
  - Need a "Save & Next" nav button instead of just "Next"
- Need a section for using an existing import profile
  - Need a picker for the import profile
  - Need to show the mapping of selected import profile along with some sample data

Adjust Transactions Step Checklist

- Change the step indicator
- Need an explanation message
- Need a transactions list with inputs
- Need to remove the delete actions from the transaction rows so that only edit is left
- Need a checkbox state for the transaction type icons

- Need a selected (highlighted) state for the transaction rows
- Need a bulk action buttons for the following actions
  - Change Type
  - Change Date
  - Change Description
  - Change Amount
  - Change Account
  - Exclude from Import
  - Include in Import
- Need forms for changing the various transaction fields after initiating a bulk action.
- Need an excluded (disabled) state for the duplicate transactions
- Need a duplicate transactions section (and message)
- Need a "select all" checkbox (one for each section -- non-duplicates and duplicates)
- Need a state where the screen is scrolled but the bulk actions are still visible
- Need pagination for the transactions lists

Finish Import Step Checklist

- Change the step indicator
- Change the "Next" button for "Import"
- Need a card with the metadata information
- Need a non-editable transactions table

Mobile Checklist

- A way to choose the account to import to.
- A way to choose the CSV file to import.
- An indicator of just how many transactions might have been found after uploading the CSV file.
- A way to match CSV columns.
- A way to create a profile of matched CSV columns.
- A way to adjust transactions.

- A way to bulk adjust transactions.
- A final review of the transactions to import, along with the other information (account name, balance change, number of transactions, etc).
- A way to navigate forwards and backwards between steps.
- A display of the current step and the overall progress through the import process.
- Short explanations for each step.

Adjust Transactions Step Checklist

- Change the title and next step
- Change the step indicator
- Need a transactions list
- Need an explanation message
- Need transaction rows with an edit button in place of the overflow button
- Need a checkbox state for the transaction type icons
- Need a selected (highlighted) state for the transaction rows
- Need a bulk actions dropdown with the following actions
    - Change Type
    - Change Date
    - Change Description
    - Change Amount
    - Change Account
    - Exclude from Import
    - Include in Import
- Need forms for changing the various transaction fields after initiating a bulk action.
- Need an excluded (disabled) state for the duplicate transactions
- Need a duplicate transactions section (and message)
- Need a "select all" checkbox (one for each section -- non-duplicates and duplicates)
- Need a state where the screen is scrolled but the bulk actions are still visible
- Need pagination for the transactions list

Complete Import Step Checklist

- Change the title
- Remove the next step
- Change the step indicator
- Change the "Next" button for "Import"
- Need a card with the metadata information
- Need a non-editable transactions table

Choose Account Step

- We just need the two select inputs: Type and Account.
- Something I'm still undecided on is whether or not each step should be encased in a card (like the current UI) or if they should be flat on the background.
- Considering we've tried to use as few cards as possible, and that the Transaction and Account forms are flat on the background, I think we should try for that approach here.

Choose CSV File Step

- We need to show a button to choose a file, the name of the file once uploaded, and the number of transactions that are 'found'.
- Based on how other apps do it, it seems like using just the default "Choose file" button is the most common thing.
  - I don't know if this is because it's a UX affordance (e.g. the button is different based on browser/OS) or if the other apps were just lazy.
  - I think we should still use a custom button; that way it's more coherent with the rest of the app's design. I also don't think using the default style is such a huge affordance that users won't know what to do -- especially on a page *dedicated* to the act of clicking this one button.
- Instead of showing a toast, we should probably just show a message with a checkmark (and the file name) to indicate that the file was uploaded.

Match CSV Columns Step

- The existing design is probably the most.. weird? not necessarily 'unintuitive' but.. way we could have done it: we pick the fields for the *columns* by setting them up as *rows* in a table.
- Really, the format should look more like what PocketSmith does, with the fields above the columns and the row data.. in rows.
- However, the actual layout and design of PocketSmith's solution is really ugly.

- I really do like approach (with the drag and dropping), but that doesn't really work well on mobile.
- Speaking of mobile, I suppose this kind of tabular approach to picking the fields doesn't really work (ya know, constrained horizontal space).

So how can we improve the existing design for mobile and then make it *even better* on desktop?

- I think something that's worth pointing out is that handles mapping by matching row data to its own fields (that is, the action that the user takes is choosing a set of row data for a field).
- Whereas uFincs currently does it by matching fields to row data (that is, the user chooses a uFincs field for a corresponding set of row data/column).
  - In fact, this is also how does it.
- I wonder which of these two approaches is a better user experience...
- I think, because we have optional fields, it's easier to do how we currently do it (that is, uFincs field → row data).

As far as a design goes, I think I've got an idea. On mobile:

- It could be structured in a column (duh) where there are multiple sections: one for each column.
- The label for the section would be the existing column name (or just the column number if the CSV doesn't use named columns).
- Then a `select` input for the uFincs field.
- Then a set of sample row data.

On desktop:

- We can take all of the sections and actually line them up as columns, like the user would find in the CSV.

As far as a design for splitting up the "New Profile" vs "Existing Profile" sections:

- On mobile, I think something like the tab bar would be sufficient for letting the user switch between them
  - Additionally, the tab bar can be hidden when the user hasn't created any profiles. This way they just get the "New Profile" form.
    - Actually, this might be a bad idea. Since then users might question *how* formats get reused if they don't see the option the first time around. Hmm...

- On desktop, I'll have to think about it.
  - My initial thoughts are that tabs wouldn't work as well on desktop, since we have so much more space to play with.
  - But I'd have to see it to see if it's stupid.
  - Just know that I don't really like the current design (where both are sections are laid out vertically with a separator between them).

Adjust Transactions Step

- We're just gonna need the Transactions table. Here's where we'll have to design the version of the table with inputs.
- I suppose on mobile, instead of having an 'overflow' action, it'll just be an edit icon. Then it'll open up a version of the Transactions form that is just for these importable transactions.

Bulk Actions

- We're obviously going to need some checkboxes on the far left side of the table rows.
  - Actually, I think I have a better idea for this. What if, like the gmail app, we used the icons for the transaction type as the 'checkbox'?
  - That is, clicking on the icon would transform the icon to a "check" and highlight the row as selected.
  - Then, as an extra affordance in case users don't pick up on this, clicking/tapping on the row itself (i.e. outside an input on desktop, anywhere on mobile) also selects the transaction.
  - An extra, extra affordance would be highlighting the row on hover on desktop.
    - Can enhance this even further by ... [thought never completed]
  - A triple extra affordance would be to have the bulk actions buttons/dropdown *visible* before the user has selected any transactions, so that they can kinda think "huh, if there's bulk actions available, then I must be able to select transactions somehow".
  - The final affordance would be the (actual) checkbox in the header. This gives users an obvious way to try and select transactions and see how the transaction type icons transform. From there, they should have figured it out.
  - Relevant article on bulk editing design: [Bulk Editing Design for Web Applications]
- We're also going to need one in the header row.
- As for how to handle the actual actions, I'm thinking the following:

- On desktop, we can have buttons (one each) for every action that can be taken on the set of selected transactions. That is, no dropdown.
- On mobile, we'll have little choice but to use a dropdown (not a select).
- As for the actions, we need one for setting each field.
  - That is, a button each for changing the type, date, description, amount and target account.
  - We also need buttons to include/exclude from the import.
  - For the text fields, it could pop down a little toast-like dialog with a text input and confirm/cancel buttons (please not an actual modal dialog).
  - For the date field, it can just pop open a date picker.
  - Finally, for the target account, it can be like the text inputs, but with a `select` instead of a text input in the dialog.
- Also, the actions should only show up once the user selects some transactions.
  - Nope, I don't like this anymore. See above.
- I don't know the best UX to lay this all out on mobile.
  - The user has the potential of having to scroll a lot to select transactions and then scroll all the way back to the top to select a bulk action.
  - Is there some way we could have the bulk actions dropdown float along with the user as they scroll? Will have to think on this one.
  - Maybe the bulk actions dropdown could just turn into an action bar that sticks to the top of the screen as the user scrolls?

Duplicates

- We currently do this by hiding the duplicates but then showing the user the option to show them in the transactions table.
- However, I kind of like the approach uses, where they have a separate table for duplicate transactions.
- Because we currently use checkboxes in uFincs to indicate that a transaction should be included in the import, I think we two options:
  - Either don't allow importing duplicates at all and remove the ability to explicitly include/exclude transactions from the import, or
  - Come up with a different way of representing "include/exclude".
  - Something to think on.

    - OMG, I had the greatest idea for this. Based on the idea of using the Transaction type icons as 'checkboxes' (ala gmail), what if one of the bulk actions was just "exclude" or "include" in import?
    - Then, to represent transactions as being excluded from import, they could just be highlighted differently with 'disabled' looking text.
    - I think this approach works best if we separate the duplicated and non-duplicated transactions. This way, we can have the duplicated transaction highlighted as 'excluded' by default, and we don't have to have a separate highlight state for 'duplicated' (like how we're currently doing it).

Review & Complete Import Step

- We currently show the following as part of the final step:
  - The account being imported to
  - The name of the file being imported from
  - The name of the import profile
  - The net change in account balance
  - The number of transactions
  - The table of transactions being imported
- I think we could separate out the metadata (i.e. everything that isn't the Transactions) into a sidebar card (kind of like how we do the account summaries for the redesigned Transactions table).
  - On mobile, this card would come first before the transactions.
- I don't think there's really much to think about in terms of design; it's mostly elements we already have.

Navigation

- We just need some "Back" and "Next" buttons.
- Obviously we'll use the 3px border radius style for the "Next" button (filled in with color), since it's a primary action.
- I think the "Back" button could be an outline button or just a link button. The current uFincs uses a link button, but I think I want to see what both look like before making a decision.
- For the disabled state, I went with just turning the background a lighter shade of grey -- aka reducing contrast, the classic for disabled states.

- However, I came across this: Disabled buttons suck - Axess Lab
- It basically says that, for a better UX, even though the button looks disabled, it should still be clickable to be able to inform the user *why* the button is disabled.
- For our case, that would mean showing a message that the current step hasn't been completely filled in (and then maybe showing some error messages on relevant fields).
- Seems like a good idea.

- Also, maybe we shouldn't use a gray state for disabled? Why You Shouldn't Gray Out Disabled Buttons
  - tl;dr decrease opacity to less than 40% instead

Stepper

I really like this:

Which is from How to Display Steppers on Mobile Forms.

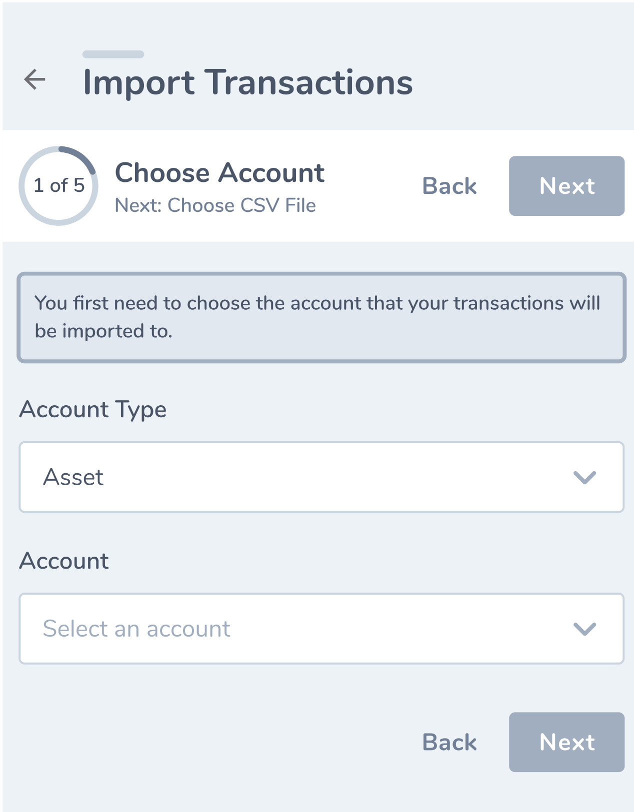
As such, I think using a radial stepper on mobile makes the most sense, while we can still get away with a linear stepper on desktop.

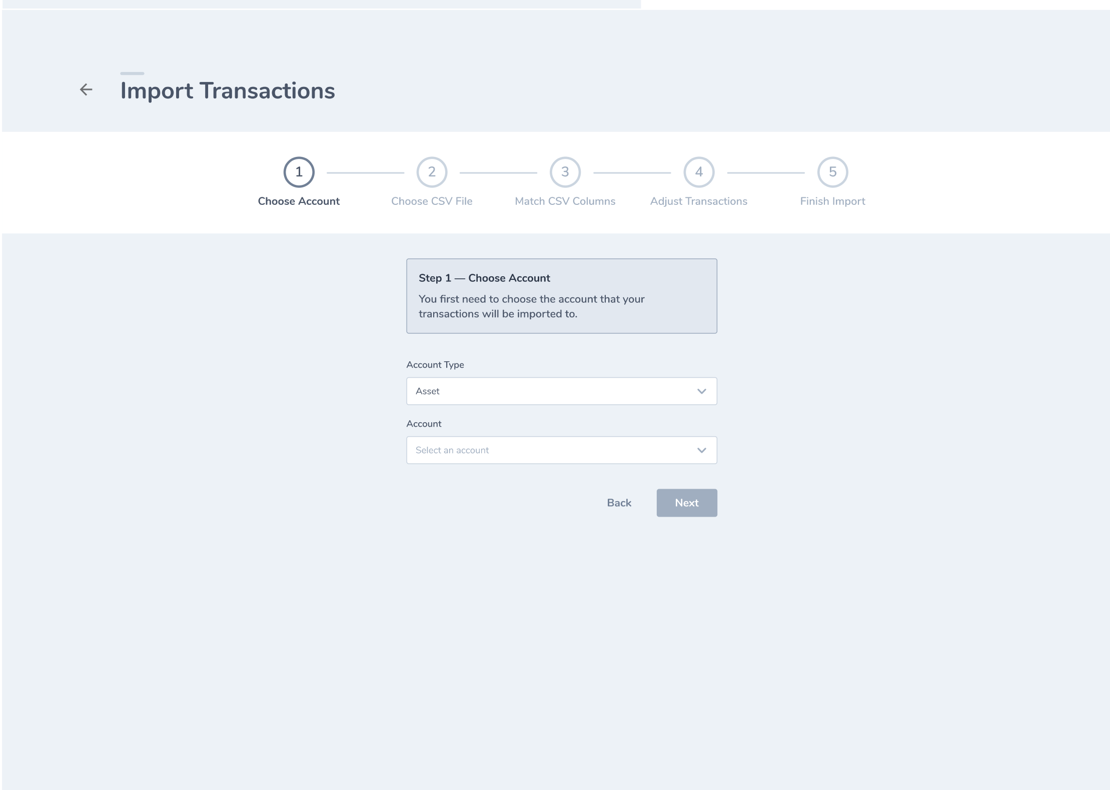
On desktop, in order to get as much horizontal space as possible (necessary for the CSV column matching and Adjust transaction steps), I think we should use a horizontal stepper over the entire page. The use of a horizontal stepper also makes more sense considering that the "Next" and "Back" buttons are laid out horizontally, so it kind of matches the user's mental model.

- The only detriment to using a horizontal stepper is that there'd be a lot of wasted horizontal space for the first two steps (choosing an account and a file).
  - Maybe we should combine the two steps?

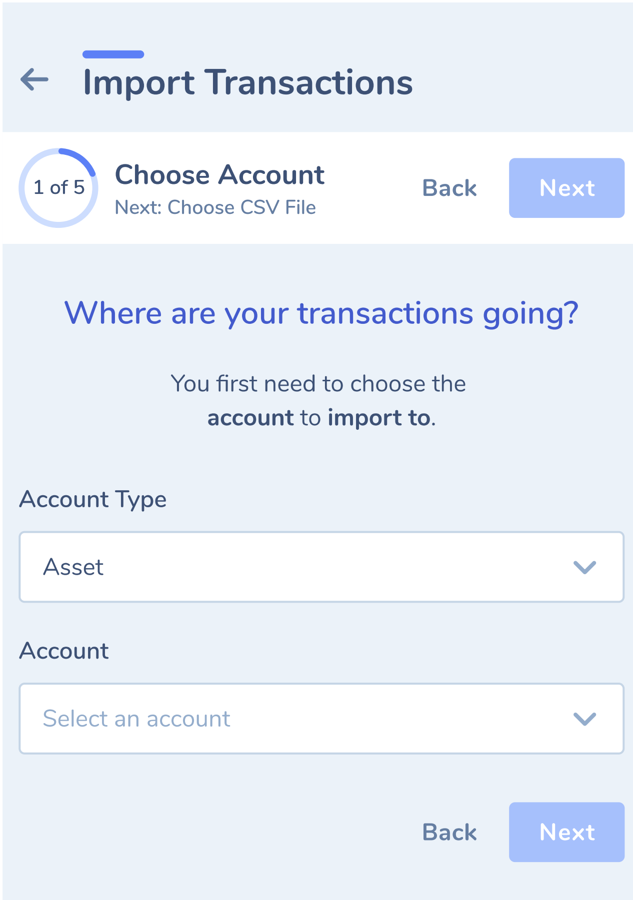
Designs

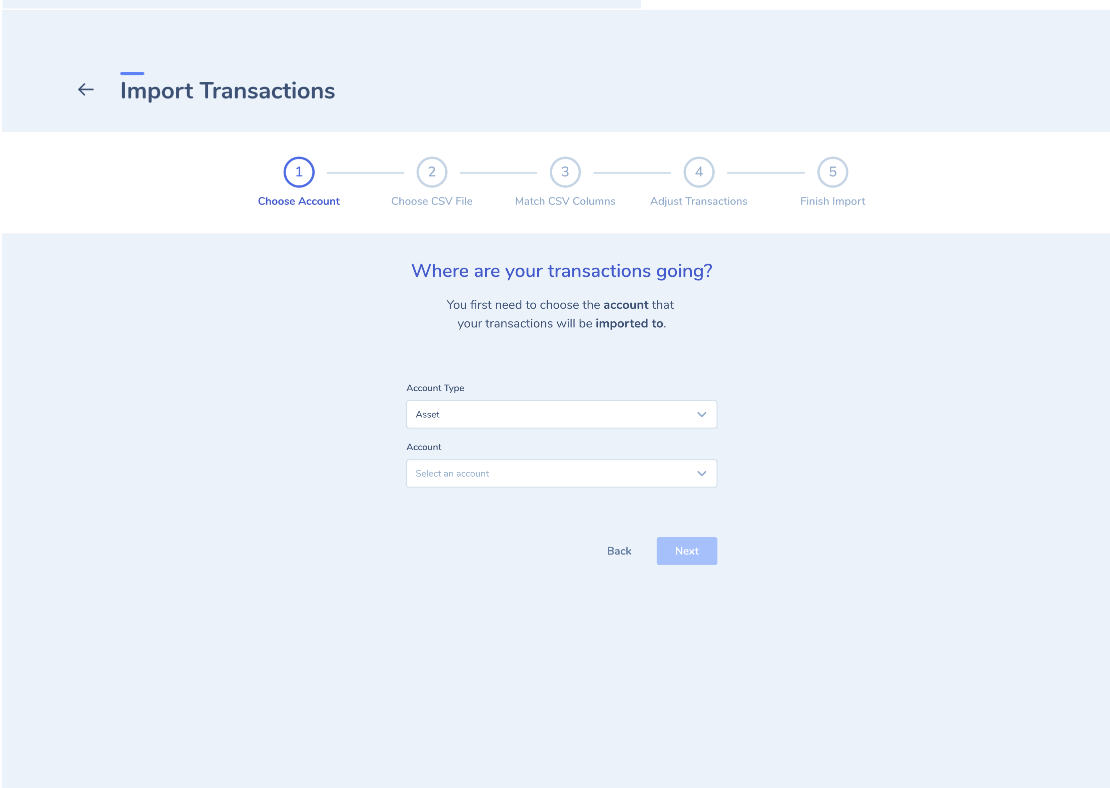
Grayscale


Import Transactions
1 of 5
Choose Account
Next: Choose CSV File
Back
Next
You first need to choose the account that your transactions will be imported to.
Account Type
Asset
Account
Select an account
Back
Next


Import Transactions
1
Choose Account
2
Choose CSV File
3
Match CSV Columns
4
Adjust Transactions
5
Finish Import
Step 1 — Choose Account
You first need to choose the account that your transactions will be imported to.
Account Type
Asset
Account
Select an account
Back
Next

Colored


Import Transactions
1 of 5
Choose Account
Next: Choose CSV File
Back
Next
Where are your transactions going?
You first need to choose the **account** to **import to**.
Account Type
Asset
Account
Select an account
Back
Next


Import Transactions
1
Choose Account
2
Choose CSV File
3
Match CSV Columns
4
Adjust Transactions
5
Finish Import
Where are your transactions going?
You first need to choose the **account** that your transactions will be **imported to**.
Account Type
Asset
Account
Select an account
Back
Next