2-Hour UX Test

Background
We are interested in understanding your thought process and approach to interrogating UX problems. We will not be looking too much into the design details, we are more interested to understand your approach. Do whatever is needed to communicate your analysis and ideas for solving this problem (text, wireframes, mock-ups etc).

Problem: We're seeing less of physical money in the world. Which in turn means parents are no longer giving their kids weekly allowance with physical money. The parents are struggling to find a balance of giving them a card and still get an overview of their spending.

Brief: You work for a bank that is currently working on bank accounts for teens (13 - 17 year olds). The team you're working with consist of developers and a Product Manager. We already have parents and teens who very much would like to be involved in the process. How would you break down this project?

Things to bear in mind:

1. 1. We don't expect you to do research into this to understand how teens think. You're more than welcome to make up assumptions, you will not be judged on incorrect assumptions. This is more to see what approach you would take.
2. Try to limit your time to a couple of hours at most

Deliverables: The approach for the project (in any shape you're comfortable communicating your idea) and at least **one visual layout** that shows your design style.

The layout could be part of the thought process i.e a user flow, user persona or it could be a mockup of the bank card, balance overview.

Introduction

My approach to this design challenge was to go about it as a free writing + sketching exercise. My hope is that this approach will give anyone reading a clear picture of how I work.
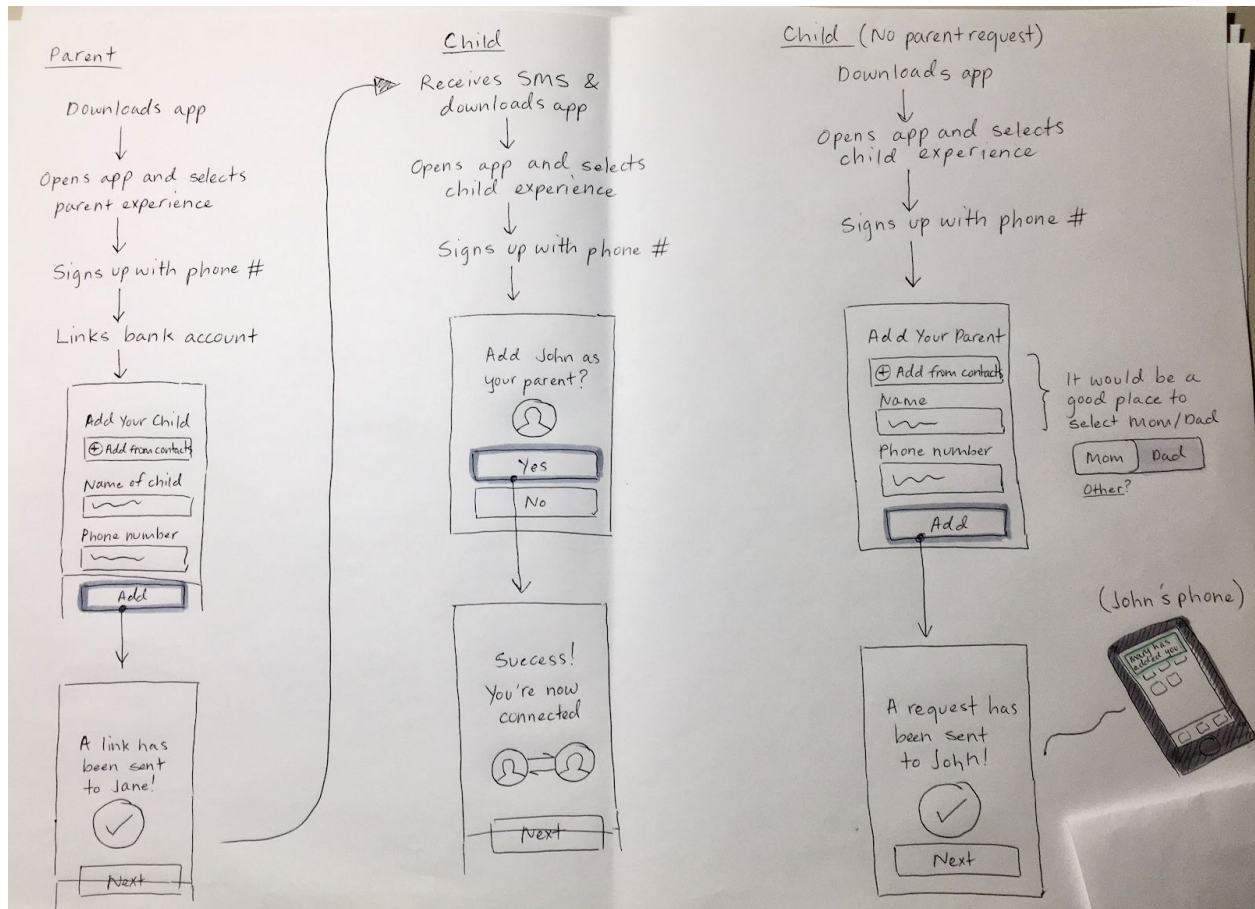
My Process

I begin the design challenge by defining some assumptions which will help me narrow in on product requirements:

- Parents must sign up for an account in the app because no integrations will be made into any of the bank's other products (e.g. an existing customer website)
- When parents sign up, they need to link a single existing bank account
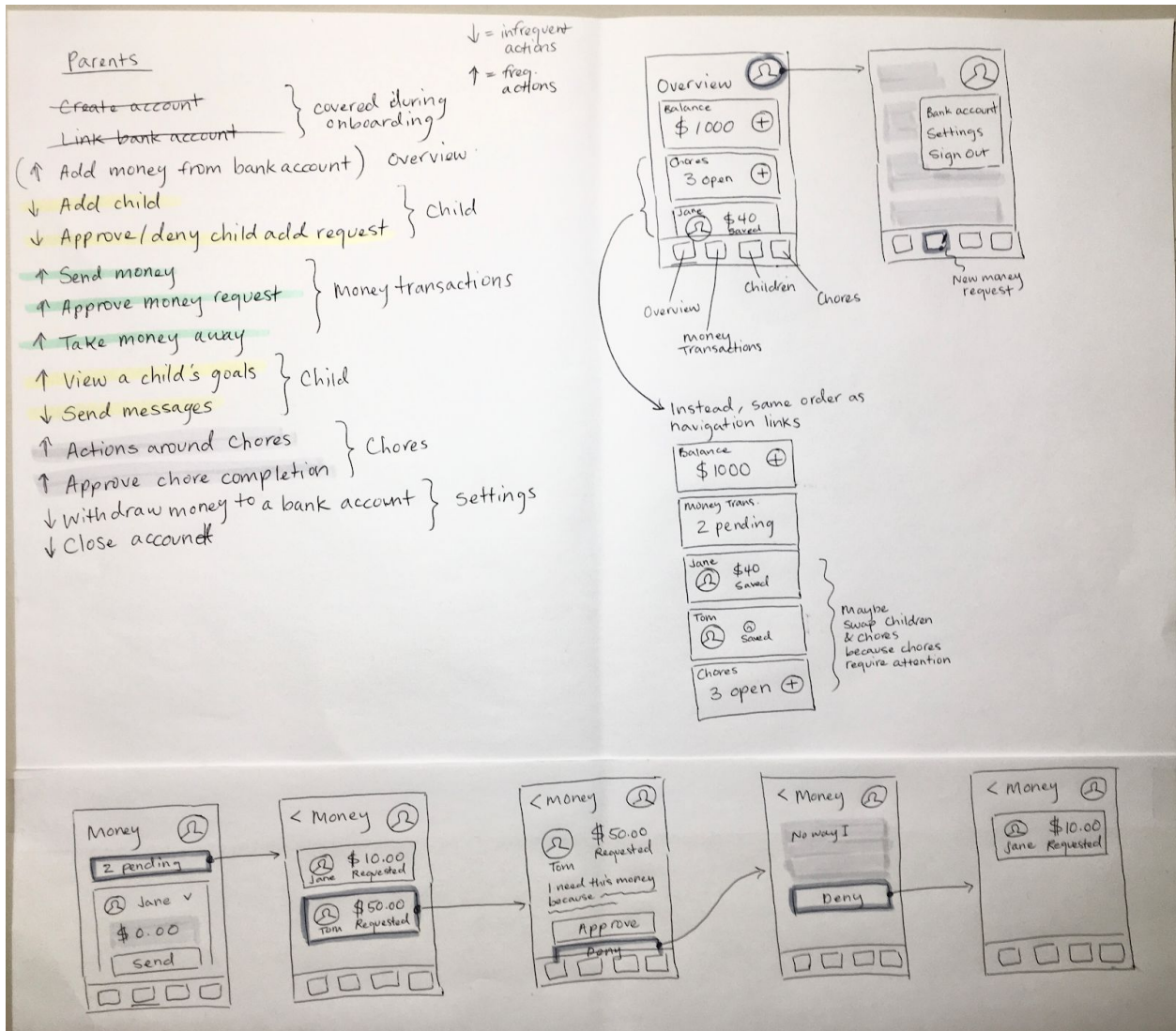
Through making these assumptions, I limit the scope of any user actions to just within the app that I'm designing for. Next, I begin to brainstorm the actions for the two user roles: parents and children. How I work is that I brainstorm both in parallel: after figuring out a parent action, I move to the child column to add any required actions that are necessary to complete the "circle of transaction".

Parents	Children
● Create account ● Link bank account ● Add money from bank account ● Add child ● Approve/deny child add request ● Send money ● Approve money request ● Take away money ● View a child's goals ● Send praise/suggestion message ● Create/edit/delete chores ● Approve chore completion ● Withdraw money to a bank account ● Close account	● Create account ● Approve/deny parent add request ● Request to be added as a child ● Request money ● Create/edit/delete goals ● Accept/cancel/complete chore ● Close account

Because it's the first step in the user's experience, I decide to sketch a bit on the onboarding experience. It's interesting to think through the different onboarding flows: younger children might be walked through account creation with their parents whereas older children would sign up on their own. A child might sign up via an SMS link triggered from a parent's account or they could simply download the app themselves and then look up their parents. Regardless of whoever initiates the connection, parent or child, the other should have to approve the request.

Parent

Downloads app

↓

Opens app and selects parent experience

↓

Signs up with phone #

↓

Links bank account

↓

Add Your Child
⊕ Add from contacts
Name of child
〰
Phone number
〰
Add

↓

A link has been sent to Jane!
✓
Next

Child

Receives SMS & downloads app

↓

Opens app and selects child experience

↓

Signs up with phone #

↓

Add John as your parent?
👤
Yes
No

↓

Success! You're now connected
👤🔗👤
Next

Child (No parent request)

Downloads app

↓

Opens app and selects child experience

↓

Signs up with phone #

↓

Add Your Parent
⊕ Add from contacts
Name
〰
Phone number
〰
Add

It would be a good place to select Mom/Dad
Mom | Dad
Other?

↓

(John's phone)

A request has been sent to John!
✓
Next

What I sketched above is just a jumping off point since it only covers the happy path. But rather than just focusing on onboarding, I decide to brainstorm a bit on how I can group my user actions (from my earlier table). This will help me to determine the app's IA and also it will give me a basis to build an overview page for parents.

Parents

↓ = infrequent actions
↑ = freq. actions

~~Create account~~
~~Link bank account~~ } covered during onboarding

(↑ Add money from bank account) Overview
↓ Add child
↓ Approve/deny child add request } Child

↑ Send money
↑ Approve money request } Money transactions
↑ Take money away

↑ View a child's goals } Child
↓ Send messages

↑ Actions around chores
↑ Approve chore completion } Chores

↓ Withdraw money to a bank account } Settings
↓ Close account

Overview

Balance
$1000 ⊕

Chores
3 open ⊕

Jane $40 Saved

Children   Chores
Overview
money Transactions

Bank account
Settings
Sign Out

New money request

Instead, same order as navigation links

Balance
$1000 ⊕

Money Trans.
2 pending

Jane $40 Saved

Tom @ Saved

Chores
3 open ⊕

Maybe swap children & chores because chores require attention

Money
2 pending
Jane ∨
$0.00
Send

< Money
Jane $10.00 Requested
Tom $50.00 Requested

< money
$50.00 Requested
Tom
I need this money because ~
Approve
Deny

< Money
No way I
Deny

< Money
$10.00
Jane Requested

Time's up… what would I work on next?
In my user actions table, there's a lot of transactions happening around chores. To me that screams a treasure trove for edge cases! Especially when you think about multiple parents and children.
If I had more than 2 hours, I would definitely ideate around this. First by simplifying the problem to interactions between just one parent and one child and then expanding the problem to one parent and two children. Then two parents and two children. Testing edge cases like these would help me to strengthen my design for scale.