Heuristic Evaluation of PayAbility

1. Problem

The app we are evaluating is Payability, a mobile app that is designed to help empower people with disabilities in their shopping experiences by guiding them through the shopping process with tools that allow price comparison, budget tracking, and cash calculation.

2. Violations Found

1. [H2-3 User Control and Freedom] [Severity 2] [Found by A & C]

When initially setting the budget, there should be a way to fix the budget in case the user typed it in incorrectly, which can happen if they have a math disability. There should be an "change budget" button underneath "start shopping" on screen 4. There is a back arrow, but the "change budget" button would be clearer.

2. [H2-5 Error Prevention] [Severity 1] [Found by C]

The user could accidentally type in a budget that is much too high/low for the intended activity. There should be a way to notify the user of a typical budget for their intended activity in case their entered amount is abnormal. A fix could be selecting the type of shopping they are doing from a list and having the app show on the screen what a "typical" budget is for that activity.

3. [H2-4 Consistency and Standards] [Severity 0] [Found by C]

On the barcode screen it says "Manually input the items description and price", but the input text field says "product description" which could be confusing. To standardize, rename the text field to "item description".

4. [H2-7 Flexibility and Efficiency of Use] [Severity 3] [Found by C]

Ended up going in a loop trying to figure out how to compare the two items in the prototype. I would click the manual tab and enter the pencil, and then press continue shopping and add the laptop but after I added the laptop, the compare prices screen didn't

come up and it was back to having no items in the cart. The comparison only came up if I clicked "add another item" under the barcode, which isn't on the manual page and I overlooked it several times. Make this button more prominent and on the manual input page.

5. [H2-8 Aesthetic and Minimalist Design] [Severity 2] [Found by C]

O n the barcode screen, the tabs for "barcode" and "manual" are extremely small and difficult to read. C suggests taking away both buttons from the top and add a "manual" button right above "add another item". On the manual page, put a "barcode" button right below price. A suggests keeping the buttons the same and just increasing the font size for readability.

6. [H2-2: Match between System and the Real World] [Severity 2] [Found by C]

O n the manual screen where it has the tax and tip options, it is not very common to look at individual items and put a tax on them before putting them in your cart. Usually tax is calculated on the cart as a whole, not on individual items. Get rid of the tip/tax page and then when the cart is full and the user is done shopping, have them enter a sales tax percent / % tip / list common sales taxes and how they would affect the cart's total price.

7. [H2-5 Error Prevention] [Severity 2] [Found by C]

H aving the option to add tip to an individual item is risky because the user can accidentally tip at a store or in a situation where they don't need to, but it is ambiguous. Get rid of the tip button and at the end when they are done have a list of suggested places where tip is normal / or have them select where they are shopping and the app can recommend whether or not a tip is necessary.

8. [H2-3 User Control and Freedom] [Severity 4] [Found by A & B & C]

T he screen where the user can swipe to select a payment option can be overwhelming as the possibilities of how to pay for something increases to extreme amounts as the total price goes up (so many different UI screens). To simplify this, the UI should ask the user what bills they have and how many of each they have, and then display the possibilities from there in order to prune the seemingly infinite combinations of payment options.

9. [H2-1 Visibility of System Status:][Severity 1][Found by A]

The screen for scanning items says compare for every stage, which is not necessarily true in every case, such as when the user is shopping

10. **[H2-4 Consistency and Standards] [Severity 1] [Found by A]**

The fact that the "shop" button does the same thing as "add an item" is somewhat unclear. The phrase "shop" doesn't necessarily lead me to believe that the button will lead to immediately scanning an item.

11. **[H 2-4 User Control and Freedom] [Severity 3] [Found by A]**

Users may not always want to add an item immediately after scanning it. There is no way to "undo" an added item or to check before adding it.

12. **[H2-3 User Control and Freedom] [Severity 2][Found by A]**

After manually entering an item, there is no immediately easy way to correct entry or undo the action.

13. **[H 2-5 Error Prevention] [Severity 3] [Found by A]**

On the compare screen where the comparison between items is shown, it is unclear what the "add item" button will do: is it adding one of these two items to the cart or is it adding another item to the comparison?

14. **[H 2-5 Error Prevention] [Severity 2] [Found by A and B]**

On the barcode compare screen, the "add another item" button is unclear in its use since the user can also enter information or take a picture of an item from that screen.

1 5 . [H 2 - 3 U s e r C o n t r o l a n d F r e e d o m] [S e v e r i t y 2] [F o u n d b y A]
After going over budget, there is no way for the user to adjust the budget, which could be desired instead of removing items.

16. **[H 2-1 Visibility of System Status] [Severity 1] [Found by A]**

Unclear why the pay screen has the word, "compare" on the top.

17. **[H2-8 Aesthetic and Minimalist Design][Severity 1] [Found by B]**

On the home screen, the similar coloring makes "Home" look like a button.

18. [H2-2 Match between System and Real World][Severity 2][Found by B]

Set budget should be at the top since it would chronologically be first.

19. [H2-6RecognitionratherthanRecall][Severity2][FoundbyBandA]

It's unclear how many items can be compared or how many should be allowed to be compared.

20. [H2-7 Flexibility and Efficiency of Use][Severity 3][Found by A and C]

The compare items page doesn't have a clear way to add items from the comparison to your cart, which forces the users into unnecessary additional actions.

21. [H2-7 Flexibility and Efficiency of Use][Severity 3][Found by A and C]

The compare items page "remove item" button seems to force more clicks than necessary. Removing items could be as simple as pressing an "x" next to the item.

22. [H2-1 Visibility of Status][Severity 2][Found by C]

On the "compare" page, users cannot see the current status of their budget, which could be very relevant.

23. [H2-1 Visibility of Status][Severity 1][Found by A and C]

On the scanning items page, users cannot see the current status of their budget, which could be very relevant.

24. [H2-1 Visibility of Status][Severity 1][Found by A and C]

On the calculations page, showing the status of budget could be very helpful.

25. [H2-7 Flexibility and Efficiency of Use][Severity 3][Found by B]

Not having a login, hinders the user in a number of ways: If they accidentally close the app, they may lose all of their information, and the Shop button can't "learn" what dollars and coins they use most often.

26. [H2-2 Match between System and Real World][Severity 2][Found by A and C]

In the shopping context, tipping is probably not too common to the point where having an option to add tip may be unnecessary in the first place.

27. [H2-7 Flexibility and Efficiency of Use] [Severity 2] [Found by C]

On the manual screen with tax and tip, if you do prompt for tax, add how much % tax.

28. [H2-3 User Control and Freedom][Severity 2] [Found by B]

I should be able to use the other functionalities while I'm shopping; when I return to the home screen, I should be able to keep my progress when shopping and even checkout/look in my cart.

29. [H2-1 Visibility of System Status] [Severity 2] [Found by A and C]

The screen shown after manual entering of item should show the original price that was entered.

30. [H2-3 User Control and Freedom] [Severity 4] [Found by B]

When I'm inputting a price manually, I can't erase any of the numbers I've inputted if I make a mistake; you need a backspace button.

31. [H2-4 Consistency and Standards] [Severity 1] [Found by B]After I click "yes" to add a tip, the screen looks very similar to the previous, but everything is shifted up which is very jolting to the user.

32. [H2-1 Visibility of System Status] [Severity 2] [Found by B]

After I remove an item from the cart because I've gone over budget, there are no recommended things to remove, but you don't inform me that I no longer need to remove items which may lead me to remove more than I need to.

33. [H2-5 Error Prevention] [Severity 2] [Found by B]

On the screen where you display the dollar bills, the dollars will be hard to see, so it would be helpful to include large, clear text that tells the user how much each dollar is that they're supposed to be using.

34. [H2-8 Aesthetics and Minimalist Design] [Severity 1] [Found by A & B & C]

All the headers are misaligned, need to be centered.

3. Summary of Violations

Category	# Viol. (sev 0)	# Viol. (sev 1)	# Viol. (sev 2)	# Viol. (sev 3)	# Viol. (sev 4)	# Viol. (total)

[H2-1: Visibility of Status]	0	4	3	0	0	**7**
[H2-2: Match Sys & World]	0	0	3	0	0	**3**
[H2-3: User Control]	0	4	0	0	2	**6**
[H2-4: Consistency]	1	2	0	1	0	**4**
[H2-5: Error Prevention]	1	3	1	0	0	**5**
[H2-6: Recognition not Recall]	0	0	1	0	0	**1**
[H2-7: Efficiency of Use]	0	0	1	4	0	**5**
[H2-8: Minimalist Design]	2	1	0	0	0	**3**
[H2-9: Help Users with Errors]	0	0	0	0	0	**0**
[H2-10: Documentation]	0	0	0	0	0	**0**
Total Violations by Severity	**4**	**14**	**9**	**5**	**2**	**34**
Note: check your answer for the green box by making sure the sum of the last column is equal to the sum of the last row (not including the green box)						

4. Evaluation Statistics

Severity / Evaluator	Evaluator A	Evaluator B	Evaluator C	Total
sev. 0	0%	0%	100%	1
sev. 1	43%	29%	29%	14
sev. 2	32%	27%	41%	22

sev. 3	50%	13%	38%	8
sev. 4	25%	50%	25%	4
total (sev. 3 & 4)	42%	25%	33%	12
total(all severity levels)	53%	38%	53%	34

5. Summary Recommendations

I think the PayAbility UI is incredibly straightforward, easy to use, and addresses a need that many apps seem to skip over. I think all of the task flows are very doable and appropriate for the app. To improve the app, there needs to be an improvement in user control and freedom when going through each task flow. Since there is a lot of chance for user input error, make sure to go through the task flows and go through every scenario where a user can have an input error and make sure it is addressed clearly and there are straightforward options to fix it. Having "undo buttons" in many cases would be helpful. It would be helpful to have an initial screen that the user can either select what kind of shopping they are doing or manually enter it into the app so the app can provide rough budget estimates and whether or not tip/tax are included as guidelines for the user. The goal is to make this process as simple and unintimidating as possible, and the motion of entering tip/tax for every item is daunting. In addition, in most cases, tax should be calculated at the very end since the tax is usually applied across all items equally. Additionally, most shopping situations would not require tipping, so that could be unnecessary. For the payment UI, it is very rare that the randomly suggested bill combinations will match the cash the user has, so there should be some redesign to make that more flexible. We would suggest a system that focuses on single bills at a time that the user can input that they paid and that narrows down which bills can be paid at a given time. One thing that could be helpful in addition is adding functionality to handle shopping at different stores. It's unclear whether or not this is supported currently, but allowing the user to input this would help customize the experience to their needs in certain areas Overall the app is very intuitive, but some redesign on button sizes and placement, the payment UI method, and escape/backtracking routes will make the app even better.