Heuristics evaluation template

Name: Lewis Moroney - student .

|  |  |  |
| --- | --- | --- |
| Heuristic | Why | Severity (0 non - 4 very bad) |
| Visibility of system status | I think that I always knew what was going on when I was pressing each button, there was no delay moving between screens | 0 |
| Match between system and the real world | I liked how pairing especially was similar to pairing my phone to my speaker and the like. I don’t use online banking apps though so I can’t really compare the process of taking out money. | 0 |
| User control and freedom | I like the goals and the getting money out, I do however think the 24 hour lock limits user control and freedom, but I guess that’s the point. Maybe give them more freedom, but being able to set the lock? | 2 |
| Consistency and standards | Given that it’s a prototype, the gray is a bit rough but the placement of the buttons were effective, and the use of blue and those colours were nice. I noticed the colours matched with the goals on the wearable. I knew where to click based on the shape and position of text on a button and the colours, like I previously said | 1 |
| Error prevention | I think that the prototype itself was telling me when I was clicking the areas that didn’t have anything active. | 2 |
| Recognition rather than recall | I don’t think I ever needed to stop and think about what I had to do, after I’d seen each screen once. | 0 |
| Flexibility and efficiency of use | Making this product more flexible with more options to do might bloat it and make users forget that its an anti-addiction thing by making it too fun to use. | 1 |
| Aesthetic and minimalist design | I like the way the screens arranged. Theres not a lot going on. I think you use that blue gradient way to sparingly | 2 |
| Help users recognize, diagnose, and recover from errors | A lot of the back spaces didn’t work. The navigation elements were there but they couldn’t be used. | 4 |
| Help and documentation | N/A | 0 |

**Facilitatory help notes – what the heuristics mean:**

**Visibility of system status:** They know what’s going on.

**Match between system and the real world:**The product should be familiar, and not too confusing and new.

* Speak the user’s language with words, phrases, - make sure info appears in a logical order.

**User control and freedom:**  Support undo and redo.

**Consistency and standards:** The product should be consistent, and follow a similar standard across the whole product to minimise confusion

**Error prevention:**Self explanatory

**Recognition rather than recall:** Product should allow learnt behaviour to be implemented so they don’t have to think as much.

**Flexibility and efficiency of use:** Learnt behaviour can make the product more efficient, so they don’t have to spend so long figuring it out.

**Aesthetic and minimalist design:** Self-explanatory.

**Help users recognize, diagnose, and recover from errors**: Error messages – assisting the users to recover from mistakes

**Help and documentation:** Allowing the product to teach.

Added extras: