Heuristics evaluation template

Name: Tanmay Kulkarni – (comp-sci) student .

|  |  |  |
| --- | --- | --- |
| Heuristic | Why | Severity (0 non - 4 very bad) |
| Visibility of system status | The Co-mit active status is hard to display in a prototype, with individual screens, but something animating, a small pulse or something to know the screen isn’t frozen would be helpful | 1 |
| Match between system and the real world | It was exactly like things I’ve used before, screens which you swipe side-to-side, and buttons look familiar. | 0 |
| User control and freedom | It was hard to see in the prototype, but I think overall, being able to choose what screen you want, no matter what money amount you’re up to isn’t restrictive, which I believe is good | 1 |
| Consistency and standards | Throughout the whole app, it looked the same throughout the whole thing and I wasn’t confused about why something looked a particular way | 0 |
| Error prevention | I think because it was a prototype again, we didn’t get to see all ways that people may make a mistake, so I think improving the prototype and having all buttons work, so its not so linear. | 2 |
| Recognition rather than recall | When I was spending it almost became instinctive seeing what was a pop-up (something I wanted to skip), and learning to figure out how to keep gambling | 0 |
| Flexibility and efficiency of use | I think it’s simple to use, accessing money makes you think but it’s not too hard which is good, and it also allows you to see how you’re tracking against your goals which I think worked well | 0 |
| Aesthetic and minimalist design | The screens never looked too busy, and I could see what I had to I believe. Font is really small though, expesially on the dark screen. | 4 |
| Help users recognize, diagnose, and recover from errors | I don’t think I had any errors moving through screens but I assume that you’d handle errors well. I think visability will be poor thou | 1 |
| Help and documentation | I never saw anything to help me because I had no errors, but I never saw where you can see history of app. I did see the little ? things which was effective. | 1 |

**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: