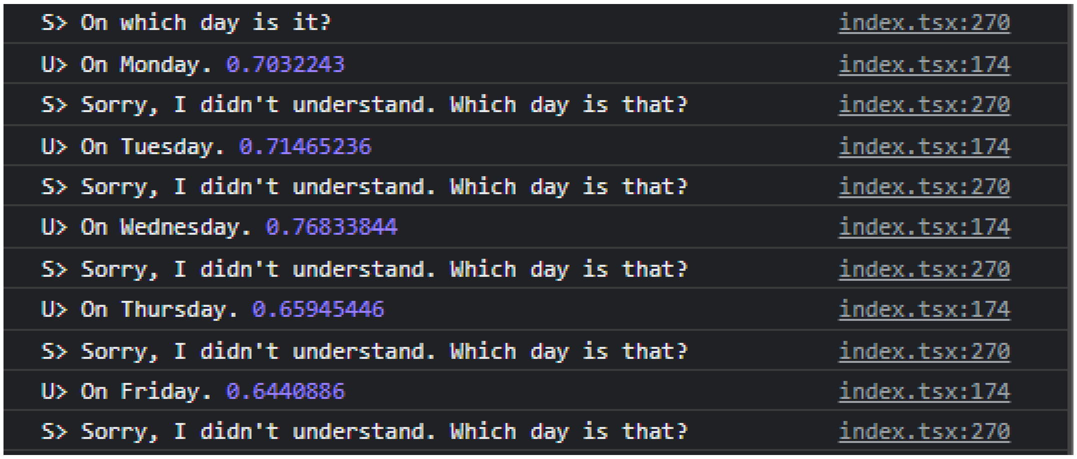
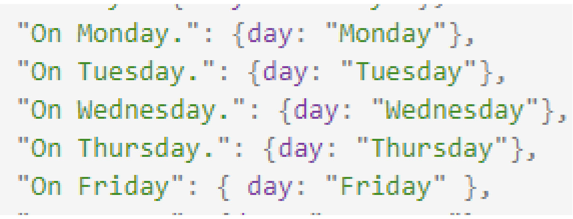
While working on my app, I have found that the most frequent and frustrating errors are due to the inconsistencies in how the app registers some words provided in its dictionary. Most of the time, it appears unpredictable, with the user needing to test every instance of their dictionary in order to make sure that it is written in the exact manner the app will register it. For example, regarding the days of the week, it appears that all of them have to be written with a final period because that is what the app registers if one is to check the developer console in browser, except for Monday and Friday, which are registered as not having a period, and have to be annotated accordingly in the dictionary. While tedious, this can be easily fixed. A different set of errors which I have tried to fix, but have been unsuccessful in doing so, are the instances in which the user provides the app with a two-word composite, such as “On Monday.”, “On Tuesday” and so on, although this behaviour is, again, inconsistent, seeing as how other two-word entries, such as “Next week” work fine. Although I have provided it with the seemingly suitable dictionary entries, the app fails to accept them, and I removed them altogether in the end. I am providing examples in the form of screenshots:



```
S> On which day is it?                                   index.tsx:270
U> On Monday. 0.7032243                                  index.tsx:174
S> Sorry, I didn't understand. Which day is that?        index.tsx:270
U> On Tuesday. 0.71465236                                index.tsx:174
S> Sorry, I didn't understand. Which day is that?        index.tsx:270
U> On Wednesday. 0.76833844                              index.tsx:174
S> Sorry, I didn't understand. Which day is that?        index.tsx:270
U> On Thursday. 0.65945446                               index.tsx:174
S> Sorry, I didn't understand. Which day is that?        index.tsx:270
U> On Friday. 0.6440886                                  index.tsx:174
S> Sorry, I didn't understand. Which day is that?        index.tsx:270
```

Fig 1. Dev console



```
"On Monday.": {day: "Monday"},
"On Tuesday.": {day: "Tuesday"},
"On Wednesday.": {day: "Wednesday"},
"On Thursday.": {day: "Thursday"},
"On Friday": { day: "Friday" },
```

Fig 2. Dictionary

Therefore I believe that this poses quite a big limitation for the person who is developing the app. The dictionary entries can't be written in just any way, and it is seemingly impossible to confidently predict how the app will act, which leads to a great deal of time being allocated to this aspect, due to the high level of unreliability.

Aside from this, one of the greatest limitations that I think the app has is the fact that it relies on dictionaries with specific entries in order for the user to be able to communicate with it. If the user has no access to what goes on in the code, it can be close to impossible to know what needs to be said for the app to interact with them. This is why ambiguity needs to be avoided as much as possible. In the 'menu' prompt, the question "What do you want to do?" is entirely too open ended for the capabilities that the app actually possesses. Therefore, a more straightforward and contextually revealing question, such as "Do you want to create a meeting or search for someone?" would be more suitable. Even so, this question invites a multitude of different variations and possible formulations of the two possible answers, of which I have included a few, in order to make sure that the user will successfully make it to the next step.