

Heuristic Analysis

Team 6: micro:bit

Undergraduate Students:

Brandon Searle
Jared Schrauben
Liam Andersen
Ryan Philipps
Sam Stevenson
Sarah Lindbeck
Scott Sviland
Ted Nachazel
Tyler Marenger
Wiley Roemer

Graduate Student:

Steven Ding

Design Description:

This system has one main view. Students will be using the device/app to log data and visualize it in a graph format. The application will allow users to graph a wider range of data than Makecode enables. Additionally, it will allow the user to interact with the data in ways that are not supported by Makecode such as finding features of interest.

Identification of UI Domain:

It is a web-based user interface. It allows users to connect their micro:bit onto the system. It begins logging data by pressing start with the micro:bit set to collect data. The data is collected and stored in the browser in increments specified by the user. A graph displays the data gathered and provides some statistical information. The user can download the data result they collected.

Usability Principles:

1: Visibility of system status

The system should inform the user what is currently happening. The status of the data recording and the device connectivity should be clearly identified. The data coming from the micro:bit device should update in real-time.

2: Match between system and the real world

The system should be easy to understand for the basic operations. The graphing should be similar to their real-world experience.

3: User control and freedom

Users might make mistakes during the runtime, such as pausing or end the recording unintentionally. There should be a confirmation button to let the user double check their decision.

4: Consistency and standards

The system should be consistent in terms and operations to prevent users from wondering whether they mean the same.

5: Error prevention

The system can reduce the probability of errors with a good design. Also before any confirmation and finalization of actions there should be a message of potential errors.

6: Recognition rather than recall

Make objects, options and actions visible from one page to another. Instructions for use of the system should also be visible.

7: Flexibility and efficiency of use

Both new users or experts should have easy access. The simple tasks should be done efficiently.

8: Aesthetic and minimalist design

Be as simple as possible, all information provided by the system should be helpful to the user.

9: Help users recognize, diagnose, and recover from errors

The system should pop out a message when error occurs and also possible solutions.

10: Help and documentation

The system should provide the user with well designed documentation. The information about the system should be covered and easy to locate.

This evaluation is based on the Jakob Nielsen's previous work:

<https://www.nngroup.com/articles/tenusability-heuristics/>

Usability problems:

1. There is no other view design such as confirmation pop-ups that prevents error actions.
2. There is no indication of current status of device (connectivity) or data (collection status).
3. There is little information about the system's usage and capability or a first-run guide.
4. There is no instruction for how to run multiple micro:bits.
5. No restore data/read from local data.

Critical Usability Concerns:

1. Data handling

This includes how to handle upcoming data from the micro:bit device. How to set the precision or the range of the graph. Can users have control over them? Considering the amount of the data expected to be collected to be huge, is the graph plotted still viewable and make sense to users?

2. Error handling

a. Disconnection

How to handle the data collected after a disconnection? Is it possible to continue the data collecting? How to manage the time axis if so?

b. Data error

Imagine when collecting temperature data, a reading of 1000 shows up because of hardware issues. How to plot and handle the error properly?