

Usability Test Report

Team-6

Micro:Bit

UX Consultant: Yogi Koteru

24 April 2023

Introduction:

The undergrad development team that I was responsible as the usability consultant for was tasked to develop an application called 'Micro:Bit'. The goal of the undergrad student's application is to develop a web application and Micro:Bit that helps students and teachers from high school and middle school to display the results using micro bit which is connected via Bluetooth. The results are shown in the dynamic graph which will display the live data to the user. When looking at the graph a user will be able to zoom in, or out, to closely observe the data points. All the information collected by the micro: bits can be downloaded by the user. When a user collects the data in his/her micro: bit after executing the experiment they can store the results in the micro: bit. The application undergrads developing helps the students to connect their micro: bit connect to web using Bluetooth, here the data is displayed in graph format A sensor will be connected to micro: bit. The micro: bit then record the data and send it off to the web application via a Bluetooth connection. After receiving the data from the micro: bit, the web application will continuously plot points in the form of a line graph. Once the data points have been plotted users can zoom in on points or filter data sets to match their needs. Data collected during usage will also be stored within the application and can be downloaded once the user ends the data stream. App data from the micro: bit will be streamed in as either floats or integers in the form of key value pairs. There are different functionalities to work with Micro:Bit users can change the name of their micro:bit as he/she wishes. Connect and disconnect buttons and Legend which is used to hide the data from X or Y axis based on the user preference. Download button which is helpful. In download we have 2 options where user can download Raw csv file or an altered csv file. Before using the micro:bit the code for the desired experiment needed to be integrated into the micro:bit so it can act properly without any trouble.

The test sessions had two objectives. The first step was to evaluate the application's functionalities as they are now implemented. The user must be able to perform each task that the project's scientist wanted. The application's usability was tested as the second goal, which was also my main focus as a usability consultant. My job during the tests was to explain what we needed the participant to perform, discuss our expectations for the tests, and respond to any queries they might have. I had at least two members of the development team present for each test. It was their duty to record all they observed during the test in the form of notes. These observations and notes were used while completing the bug report and testing challenge forms. We gave the participants a warm welcome and went over everything that will occur during the test before allowing them to begin. I gave them instructions on how to fill out the preliminary data during the exam and addressed any technical queries they had. If the query was "What does micro:bit mean?" or a similar query regarding a topic that a real user of the application would be familiar with. I would respond to it. I would attempt to let them figure it out on their own if it were something different, like asking what button they should press. This was done because the bigger the problem that our development team had to look at became, the longer it took them to figure it out. once the test was finished by the user. After asking them a few quantitative

questions, I would follow up by asking if they had any other queries, remarks, or issues with the application. The observers would note all the responses to these queries.

Test Plan:

Here are the initial test designs that were part of the assignment for the usability test plan. The second test, while mostly identical to the first, also requests that the user test out the application's pause/resume feature. All six of our test sessions ultimately used only the second test. We felt there was no justification for anyone to forego testing that aspect of the feature during their test session after discussing it with the development team.

For each test case, the following will be the same:

Equipment:

Micro:bit and Laptop/Desktop.

Quantitative Measurement:

How long it took them to map their "stream" in comparison to how long it takes one of the developers to map a "stream" is one factor that will be considered. Since they created it and are familiar with its workings, it would be safe to expect that the development will move more quickly, but observing how much of a difference in time exists would be a useful indicator of how usable the app is. They must roughly be divided into equal length "streams".

Qualitative Measurement:

There will be a short survey after the test is conducted where the user will answer a few different questions. The questions will be answered on a scale of 1-5 where 1 strongly disagrees and 5 strongly agree. These will be used for qualitative measurements.

Post Scenario:

The following inquiries will be made of the user before the post-scenario interview and questionnaire, and they will be graded on the scale:

- On the application, you were able to complete all of your tasks.
- You never doubted the next step of action.
- The application's structure and flow were straightforward and understandable.
- Please score the application's overall usability. (1 being bad and 5 being excellent).

They will then be asked to provide an explanation for each of their answers. After that we will open it up to them and ask if they have any additional questions, comments, or concerns.

Test Set Up Details:

The test will be conducted in-person. The location for the test will be sent to the mail before 2 days for the test.

1. Test

a. Goals:

Straightforward walkthrough to test the basic functionalities of the web application.

b. Scenario Description:

This is an application for streaming data. You will open the application and use micro:bit to the best of your ability follow the instructions in help button and connect the micro:bit. While the data is streaming you can zoom in and out pause the data flow, remove x or y axis data.

c. Task List:

- Access the website.
- Use Micro: bit to connect Bluetooth.
- Perform the experiment.
- Play/Pause the graph.
- Rename the device.
- Use Legend options.
- Download the csv file.
- Use the Help Button in case of confusion.

d. Potential Observations:

- Whether or not the micro: bit is giving output smoothly.
- Whether the user can easily find their way around the application.
- Whether confusion causes the user to hesitate or delay what they are doing.

2. Test 2

a. Goals:

Same as test 1, but we will also test the pause/resume functionality and ask the user to connect and disconnect the micro:bit.

b. Scenario Description:

This is an application for streaming data. You will open the application and use micro:bit to the best of your ability follow the instructions in help button and connect the micro:bit. While the data is streaming you can zoom in and out pause the data flow, remove x or y axis data.

c. Task List:

- Access the website.
- Use Micro: bit to connect Bluetooth.
- Perform the experiment.
- Play/Pause the graph.

- Rename the device.
- Use Legend options.
- Download the csv file.
- Use the Help Button in case of confusion.

d. Potential Observations:

- Whether or not the micro: bit is giving output smoothly.
- Whether the user can easily find their way around the application.
- Whether confusion causes the user to hesitate or delay what they are doing.
- Whether pause/resume functionalities work.
- Whether they can download csv file.

Results

I'll go over the testing procedures and their outcomes in this part. The tests went rather smoothly overall. We were able to complete 8 tests despite having one person change times and arrive a bit late. As was already indicated, all 8 participants thoroughly explored the application and all its functions that were already in place. The four questions included in the test plans were designed to evaluate three distinct areas, with the last question evaluating the application. The participants were asked to score their level of agreement with the first three statements on a scale of one to five, with one being a strong disagreement and five representing a strong agreement.

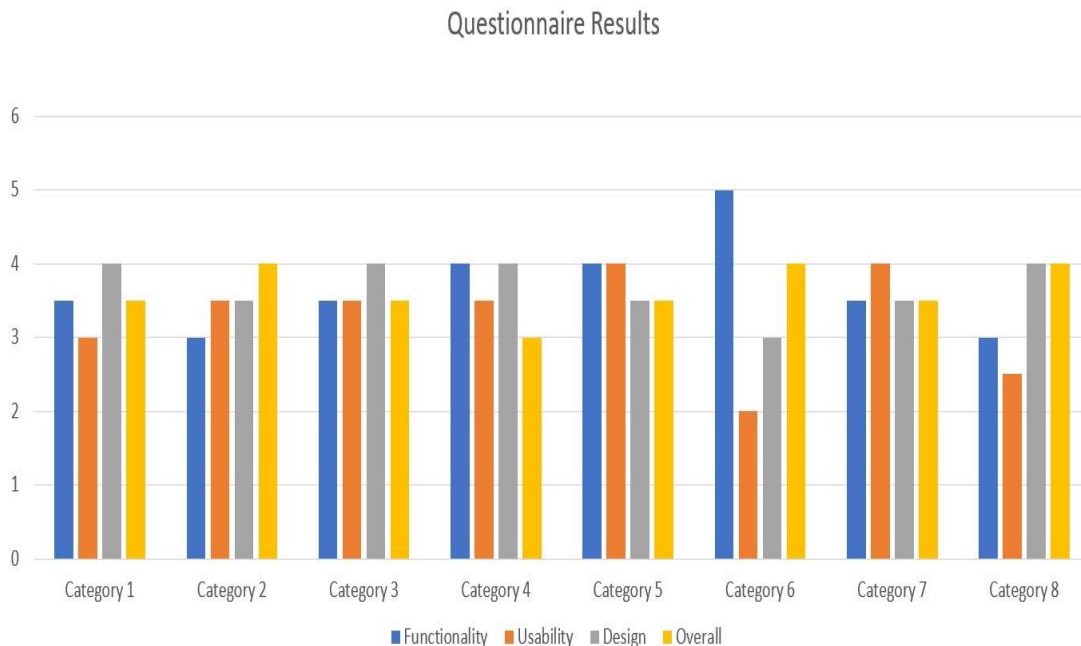
"You were able to do everything you needed to do on the application," was the first claim made. This evaluation of the application's technological capabilities was the goal of this statement. Although we were aware that none of our participants knew much about the subject matter they were testing, we nevertheless wanted to know if they believed they could complete the task after hearing the task description read out to them beforehand. This question received a 4.33 out of 5 on average. This relates to what we hoped and expected. There was no reason to anticipate many low results on this one since we only asked the participants to test fully developed functionality.

"You were never unsure of what you should be doing next," was the second statement made in reference to the application's usability. Since I didn't anticipate that most of our participants would have taken a usability course yet, I didn't want to use a general term like "usability" when I was attempting to evaluate a particular aspect of the program. I reasoned that it would be simpler for a participant to recall instances during the run-through when they were a bit perplexed than it would be to recall instances when they didn't find it useful. I specifically instructed the observers to take note of and record any instances where it appeared that the user was pausing or was unsure about what to do. The overall score for this assertion was 3.33. This was the lowest average score for any of the questions, which was lower than I had hoped. I will

go into all the factors that contributed to this lower score as well as what needs to be done to improve it in the conclusion section.

The line "The layout and flow of the application was simple and intuitive" appeared in the third question, which focused on the application's design. Once more, the participants were questioned on their level of agreement with that assertion. The highest average score of any question was 4.5, which was given to this one. The development team was able to create a good design without having to spend a lot of time meticulously developing numerous pages because this was a very modest application with only five or six pages. A few minor issues, including pop-up boxes overlapping and potentially repositioning certain buttons, were mentioned, but the participants didn't express any significant worries about the design of the application.

The participant was asked to rank their overall application in the fourth and final question. This question's average score was 3.75, which is not horrible but a little below what I had hoped for. As I said before, I'll go into more depth about particular items that must be addressed in the application in the section after this. The scores from each test for each question are listed in the table below.



As I mentioned at the beginning of this section, I think the development team and I received the best results possible from these tests. All the participants gave truthful responses and performed admirably during their run-throughs.

Conclusion:

I'll go over some of the specific issues that some of the participants had in this section and talk about what needs to happen to fix them. This program is not particularly complicated. When utilizing this program, the user will typically spend more than 90% of their time on only one page (the data streaming page). Because of this, the participants didn't encounter a very extensive list of defects or issues, but I do want to share the most significant ones here. On the bug report form in the appendix, all the noteworthy bugs are recorded.

One stands out since it was brought up by each of the six participants at least once, if not more, during the test sessions. They cannot identify whether the micro:bit is connected or disconnected as it does not show any notification when connected or disconnected. Fortunately, the solution to this problem is simple. The development team will need to add functionality so that it gives pop-up when connected or disconnected. This will lead to a much smoother understanding of the data.

Another major bug we have found out is when the micro:bit is connected and streaming the data when we click on pause and then after 5 seconds (assumption) when we click on resume it displaying the data that it has collected in that 5 seconds and adding the data to graph which making the graph clumsy and creating confusion.

Another challenge we face is that the user is not able to understand the usage of micro:bit in the real world. Many users have confusion on how to use micro:bit and the buttons on it.

Sometimes while performing the experiment the micro:bit couldn't display the graph.

Users never used the legend button, and they find it not useful.

There were about as many bugs as I expected throughout the tests. I had meetings with the development team after each round of testing to go over any bugs that were found and what exactly needed to be fixed. I believe that the development team has all the tools and skills to make the necessary changes to complete the application.

Appendix A:

Undergraduate Session Attendance

| | Test Date | Test Time | Development Team Members Assisting |
|---|-----------|-----------|------------------------------------|
| 1 | 4/10/23 | 3:00pm | Noah, Roger, Micheal |
| 2 | 4/10/23 | 4:00pm | Noah, Roger, Josh |
| 3 | 4/11/23 | 1:00pm | Tyler, Parker, Roger, Micheal |
| 4 | 4/11/23 | 4:00pm | Tyler, Thanuja, Roger, Josh |
| 5 | 4/11/23 | 5:00pm | Parker, Thanuja, Roger, Micheal |
| 6 | 4/13/2023 | 1:00pm | Tyler, Roger, Parker, Micheal |
| 7 | 4/14/23 | 12:00pm | Noah, Josh, Roger |
| 8 | 4/14/23 | 5:00pm | Parker, Noah, Roger |

Appendix B:

Bug report form

| Bug # | Bug Name | Bug Description |
|-------|-----------------|--------------------------------------|
| 1 | Data Flow | Micro:bit can't flow data to graph |
| 2 | Data Collection | When in pause still collecting data. |
| 3 | Values | Values on x&y axis |
| 4 | Save State | |
| 5 | Pop-up | |
| 6 | Live | |

Appendix c:

Testing Challenges

| Number | Name | Challenge Description |
|--------|-------------------------|--|
| 1 | Talking | We asked the participants to talk us through their thoughts while they were using the app so we could get better feedback, only a couple did this. |
| 2 | Too Nice | Seemed like some of the participants didn't really want to give us a bunch of negative feedback even though that is what we are looking for. |
| 3 | Explanation | Many users did not understand properly about the micro:bit first time. |
| 4 | Late | Some of the user showed my late to the test. |
| 5 | Understanding the graph | Most of the users taking time to understand the graph and its values. |
| 6 | Units | When doing experiment, they can't understand the values without units. |