Heuristics Evaluation of Team 3342 Date: 10/16/23

By Team #: 3348 **Names:** Abdullah Altaweel, Serene Belhadj-Yahya, Sophia Bowie, Sanskriti Gupta, Ajay Krishnaswamy, and Soongeol Kang

Part 2: General Heuristic Evaluation of Prototype Interface

Heuristic	***Evaluation*** In the space below, enter your observation and evaluation of the degree to which the heuristic has been satisfied. Use as much space as you see fit.
1. Visibility of system status • Always keep users informed about what is going on. • Provide appropriate feedback within reasonable time.	The app has helpful messages after submitting your weight/blood pressure that let you know your info has been properly submitted and sent to healthcare professionals.
2. Match between system and the real world • Speak the users' language, with words, phrases and concepts familiar to the user, rather than system-oriented terms. • Follow real-world conventions, making information appear in a natural and logical order.	The app was easy to use and log information, however it was confusing when it came to reporting data to medical professionals.
3. User control and freedom • Users often choose system functions by mistake. • Provide a clearly marked "out" to leave an unwanted state without having to go through an extended dialogue. • Support undo and redo.	The app lacks the ability to undo entering a weight or delete an incorrect entry. Adding this feature would help users handle making errors.
4. Consistency and standards • Users should not have to wonder whether different words, situations, or actions mean the same thing. • Follow platform conventions.	Across the app, design choices such as colors and fonts stay consistent. Additionally, the back button on every screen is in the same place, which allows users to quickly know what to expect while interacting with the app.

5. Error prevention • Even better than good error messages is a careful design which prevents a problem from occurring in the first place.	Although it was not an option in the prototype, it would be worth it to consider a check on the user's input for blood pressure and weight as those values may become unrealistic.
6. Recognition rather than recall • Make objects, actions, and options visible. • User should not have to remember information from one part of the dialogue to another. • Instructions for use of the system should be visible or easily retrievable whenever appropriate. • Buildings are all visible and clickable • Experienced users can directly check on the buildings they are interested in by clicking on them • Recent searches feature is useful for users that use the same buildings regularly	All the options available were clearly presented in the application's landing page, but was somewhat unclear if historical data would already be in the input page or just the history page, as the "report" feature appears to be in the input page. That mild inconsistency may be slightly confusing to new users.
7. Flexibility and efficiency of use • Accelerators -- unseen by the novice user -- may often speed up the interaction for the expert user so that the system can cater to both inexperienced and experienced users. • Allow users to tailor frequent actions.	Although there are no accelerators that obvious, the application is minimal enough and does not require the use of shortcuts or such accelerator features. The application has the most frequently used (and as far as we can tell, the only features) in the front of the landing page.
8. Aesthetic and minimalist design • Dialogues should not contain information which is irrelevant or rarely needed. • Every extra unit of information in a dialogue	The app is very minimalistic and does not include excess information that is irrelevant to the users' needs.

competes with the relevant units of information and diminishes their relative visibility.	
9. Help users recognize, diagnose, and recover from errors • Expressed in plain language (no codes) • Precisely indicate the problem • Constructively suggest a solution.	Although it is hard to judge from the prototype, it might be reasonable to expect a way to remove faulty inputs and data entries. Maybe a way to input data for the past as users may forget to input data for every day.
10. Help and documentation • Even though it is better if the system can be used without documentation, it may be necessary to provide help and documentation. • Help information should be easy to search, focused on the user's task, list concrete steps to be carried out, and not be too large.	Learn More page is very helpful for users to find answers to their questions. Since medical info can sometimes be confusing, consider adding more information to the history page. For example, displaying a healthy/goal blood pressure on the graph could help users better understand the context of their medical info.