**DP Final Report Assignment**
*6.033 2021; due 5/11 at 11:59pm*

Having now had two rounds of feedback on your design, it's time to write your final report. Unlike the proposal document, the report should contain enough detail that it could feasibly be turned over to Facilities for implementation. It should also contain an evaluation of your design.

**Outline of the Report**

Your team's report should be approximately 6000 words and follow the basic outline below:

- **Title page:** Give your report a title that reflects the subject and scope of your project. Include your names, email addresses, the name of the recitation instructor who is grading your DP deliverables, and the date on the title page.

- **Introduction:** Summarize the problem to be solved and what your design is intended to achieve. When summarizing the problem, you should extrapolate and highlight the technical challenges that make this issue a hard systems problem from the design description. Outline your design and briefly outline why your design meets the requirements.

- **System Overview:** Provide a high-level description of your system's modules, behaviors, and innovating techniques or strategies. This should include a system diagram and serve to introduce definitions for key terms used in the Design section. The system overview also provides an opportunity to prioritize the main system objective(s).

- **Design:** Explain your design. Identify your design's main components and workflows. You should sub-divide the design, with corresponding subsections in the text, so that the reader can focus on and understand one piece at a time. Explain why your design makes sense as well as explaining how it works. Use diagrams, pseudo-code, and worked examples as appropriate.It should be clear from this section that your design meets the specifications of the assignment (e.g., that it does not exceed the storage available on the servers nor the capacity of the network, etc.). Leave any major calculations to the evaluation section, though it's fine to reference those calculations beforehand (e.g., "Our design results in a communication overhead of fewer than 1Kbit/sec; see Section 3.1 for an analysis.").

- **Evaluation:** Evaluate your design. There are more details about this section below.

- **Conclusion:** Briefly summarize your design, highlighting the novelty or specific focus of your system, and provide recommendations for further actions and a list of any problems that must be resolved before the design can be implemented.

- **Author contributions:** A brief statement (typically 1-3 sentences long) describing the contributions of each author. These contributions could include designing specific components of the system, research or investigation related to the design problem, qualitative or quantitative evaluation, writing the text of the report, editing the report, creating figures, etc.

- **Acknowledgments and references:** Give credit to individuals whom you consulted in developing your design. Provide a list of references if appropriate.

**Evaluation**

A good evaluation will do more than just calculate metrics relevant to your system; it will also use calculations to justify design decisions. For example, "Our method for transmitting data from the server results in an average upload time of two minutes, compared to a design without this method, which results in an average upload time of ten minutes."

At a minimum, your evaluation section should address the following questions:

- What is the communication overhead of your system? What does your system gain from any additional communication that you added? That additional communication might be administrative overhead, it might be duplication of certain messages, etc.

  When you report your network overhead, it should also be clear to the reader *which* type of network you're using (BLE, wifi, etc.), as well as whether there are any timing constraints on how you send data (for example, are you constantly sending a small amount of data, or sending larger amounts in bursts, etc.).[1]

- How long do various operations take on average? In the worst case? These operations could include the latency of getting an alert to a user, the amount of time it takes to locate a piece of data on local storage, etc.

- What parts of your system limit scale, and what are those limits? E.g., could your system handle the addition of more users? Of more data?

- How does your design handle the use cases given? Also speculate about large changes in the use cases that would or wouldn't be accommodated by your design.

Additionally, we'd like you to discuss the following issues: We've designed this project around the assumption that every user has a phone (though the phones may sometimes fail). Suppose some users *do not* have a phone. What happens in your system? Are there parts of it that still

[1] In the real world, sending data over BLE vs. sending data over wifi have different consequences when it comes to battery drain. Designers would think carefully about how much data was sent over each network — not just how much data was sent in total by the system — and even how frequently each radio was turned on.

work well?

In the context of this project, you are *not* required to accommodate users without a phone. We simply want you to explain how your system fares here, since this is an important issue that one would encounter in the real world.