6.1800 Project Part 1: System Enhancement Report

Due: April 15, 2026, 11:59pm
Length: 2000-3000 words (see below)
Submit: On Canvas, as a team

The 6.1800 Project involves two deliverables: a system enhancement report (this deliverable) and a project presentation. This first deliverable has several purposes: to provide a framework within which you can articulate your preliminary design and justifications for design choices; to get feedback from technical staff that will guide the remainder of the design and/or modifications to the design; and to gain mastery of many of the skills needed to write a systems design paper. The report explains how your design works to solve the assigned problem. It further demonstrates that the design meets all the requirements and justifies design decisions.

Your report should be written for a broad range of systems experts from different fields, and the paper should build on key reasoning patterns and concepts (defining behaviors and properties in relevant use cases; naming techniques used to produce desired outcomes), using specific technical language as appropriate. It should build on our framework of key concepts such as modularity, simplicity, scalability, etc., using specific technical language as appropriate.

Late policy: Like all communication assignments in 6.1800, if you reach out to 6.1800-personal@mit.edu in advance of the deadline, we will give your team a 24-hour extension, no questions asked. Beyond that, if you submit late, we will penalize you one letter grade per 48 hours, starting from the time of the deadline.

Word count: The word count for this assignment is 2000-3000 words, which is an unusually large range for our assignments. We are expecting you to write at least 2000 words; if you are writing less than that, you are not giving us enough detail. We've given you a much larger upper bound because some teams end up front-loading the work on their project, and work out many of the details during the report time. These teams will likely write closer to 3000 words. Other teams do a reasonable amount of design work for the report and add in additional details between the report and the presentation; these teams will likely write closer to 2000 words.

Writing Guidelines

Your preliminary report should:

- Briefly introduce your design by identifying the purposes and goals, then summarize your subsystem's modules, behaviors, and innovative techniques or strategies. Describe who is *most* impacted by your subsystem, and how that impact influenced your design decisions. Give your subsystem a name for ease of reference.

- Include two system diagrams (titled, labeled, captioned, and referenced in the text) and include additional figures if appropriate to illustrate complex or challenging concepts. Your first system diagram should illustrate how your subsystem fits into the larger system described in the spec. Your second diagram should illustrate the design of your particular subsystem.

- Define your subsystem completely, clearly linking design choices, techniques, or strategies to the behaviors and properties they produce. Justify your choices with use cases, scenarios, design principles, and impact on stakeholders; you may also justify your choices by comparing to alternative designs that you considered. Identify major design decisions, tradeoffs, and choices.

- Conclude by summarizing how your design will meet the requirements, highlighting the novelty or specific focus of your subsystem, and note problems which remain to be resolved. (May be a separate section or may be the final paragraph.)

Technical Guidelines

Your preliminary report should contain enough information to convince the staff that your design is reasonable, but it does not need to describe every single detail of the subsystem. That means your report should:

- Clearly lay out the problem and explain the challenges your subsystem needs to overcome.

- Characterize the design of your subsystem at a high level and describe how it will meet the requirements. Doing so will require you to explain some of the implementation details. You should be describing *your* subsystem and *your* design decisions; you do not need to repeat the description(s) given in the spec.

- State any assumptions that you made in your design.

- Describe the use cases most important to your design, since those use cases should inform how you design your subsystem.Explain the impact of your subsystem on the people/groups who are most impacted by it.

You do not need to evaluate your design; an evaluation section will be part of your presentation, but not this report.

Grading + Feedback

You will receive a grade on this project from your WRAP instructor. You will receive technical feedback from your TA, including a summary of whether you need to make any major or minor

changes to your design ahead of the presentation (the vast majority of teams will!). Because this is your first time designing a system, we are not grading you on how successful your design might be in practice — we expect that your design will improve between this assignment and the final presentation, based partly on the feedback you receive from your TA. Your grade will be based on how clearly you describe your design, and how well you justify your choices using systems reasoning and real-world motivation; you can see a more detailed rubric here. A clear, complete, and well-justified design will make it easier to prepare for the presentation.

AI Policy

Like all systems, GenAI has benefits and harms. Using these tools can cause real harm in the world (theft of other's work, theft of your own work, environmental harms, etc.) and can also harm your own learning. In this class, you're allowed to use GenAI for the following:

- Spelling and grammar checks (e.g., Grammarly)
- Word-level suggestions
- Searching for sources (you must confirm and include the source)
- Looking for high-level information (e.g., Google's AI summary); you would need to track that back to a real source if you wanted to cite it in any classwork
- Research (you must confirm and include sources)

You are **not** allowed to use GenAI to create text (beyond a single word) or images. This means you should not use GenAI for drafting and revision. Using GenAI to create text/images runs counter to the learning objectives in this class, the majority of which center around developing, understanding, and expressing complex ideas.

Additional Guidance for this Project

This project will be overwhelming at first; there are a lot of choices to make, and the spec is large. Part of your first job is to identify where (in the spec) the information given to you is relevant to your work; the spec presents a relatively complete picture of a system, and you're unlikely to need to use every detail in it to design your subsystem.

Before you even begin to design your subsystem, you should make sure you can answer the following questions:

- What are the goals of the system?
- Who might be impacted by this system? Users, sure, but who else?
- What are the modules?
- What modules communicate directly? How (via what technology)?
- What parts of the modules exist already?
- What parts of the system do you need to design?
- What things from lecture/recitation seem relevant so far? It is *normal and expected* to not use every single thing you see in lecture in the project

Once your team is ready to begin your design, you should first decide what you want to prioritize. Simplicity? Reliability? Scalability? Etc. Then work to develop a design that works when nothing goes wrong — no network failures, no machine crashes, no bikes being taken out while a video is transferring, etc. From there, you can adjust your design to accommodate different types of failures and other situations.