

Dear Friction Force Explorers:

We've received your requirements specification for the multi-agent drawing system and are quite impressed with the amount of work put into the document. The entire project is well motivated; we especially appreciated the sources you cited to strengthen the presentation of your vision. The document is laid out in a meaningful and organized fashion, which makes parsing simple and efficient. We particularly appreciated the table of contents at the beginning which gave a broad overview of the document structure and content. The scenarios and use cases presented were very clear and done in great detail, and we were able to understand many of the interactions that the system should support. The "Reload Writing Implement" use case was particularly well explained and justified. The presence of supporting figures and images helped clear up some other questions that arose from either the wording or general presentation of the material.

That said, we did have a few questions concerning some of the specifications. Generally speaking, for some of the requirements, we weren't quite sure of the motivation behind them, or of their exact meaning. While each requirement had a priority listed, we weren't exactly sure how the priority related to the necessity of the given requirement, nor what the specific priority number actually implied. Even after reading through the document and assuming sensible answers to the aforementioned, there were a number of points about the system which we felt were unclear or unaddressed. We also weren't entirely sure of the justification behind the assumptions, nor how each particular assumption ties into the proposed requirements. Finally, there were a few places throughout the document where we felt the grammar and formatting should be improved to ease readability.

The assumptions listed were mostly reasonable, but we felt that a few things were missing. First, there was no assumption or requirement about the presence of obstacles. There should either be an assumption stating that the environment is obstacle-free or a requirement stating that the robot must be able to deal with obstacles. Additionally, there were no assumptions or requirements about the weather that the robot should operate in. Again, you should either assume that there won't be rain or snow or extreme temperatures, or there should be a requirement stating that the robot can operate in those conditions. The one assumption that did not seem reasonable was assumption A4, which assumed near perfect communication between the robots and the controlling host. This should be a requirement of the system instead of an assumption, since the design of the system will determine whether this is true.

A couple aspects of the assumptions and scenarios seem to contradict one another. For example, assumption A3 rules out using spray paint, but the product scope specifically mentions spray paint as a possible medium. Assumption A1 assumes the surface is flat and homogenous, but in the sport lines scenario it says that the robot must be able to travel on uneven surfaces. These conflicts make it unclear what the robot is expected to do. Is it expected to do everything mentioned in each scenario, or should it follow the assumptions? The design of the robot will need to be very different depending on which scenario it is expected to perform.

There were a few elements that we thought needed additional details or clarifications. Particularly, there could be more clarification on the scale of the system. All of the use cases are varied enough that they could fit multiple robot designs, in terms of size, number of robots in the system, and other properties of scale. It wasn't very clear if the proposed system would contain one singular robot design and only vary in number of robots, or if there would be multiple robots at multiple scales. For example, the lines being drawn on an airport would be of a much different scale than lines being drawn in

a sports arena or a child's drawing, and it wasn't very clear whether the system will handle all those scales, or how exactly the system would scale to its different operating scenarios.

A few items were ambiguously worded. The words "system", "robot", and "agent" should be clarified to refer to specific items, rather than being used relatively interchangeably, to avoid confusion. Also, it would help to clarify whether the "kill switch" is killing a single robot or the entire system, and where the kill switch will be operated from.

The requirements listed and described are fairly comprehensive, and provide a comprehensive overview for what is expected for the robot system. However, there are a few things that are not clear, and a few changes can improve the clarity and the coherence of this section. First, the Likert scale you employed for the priorities is able to clearly describe the relative importance of each requirement; it, on the other hand, does not clearly describe the importance of each requirement to the function of the robotic system. With just one or two sentences explaining the meaning and definition of each priority number, the requirements will be more coherent with respect to the entire project. For example, you can say, "Priority 7 means that the requirement is critical to the mission, and must be accomplished for the robotic system to function properly", or "Priority 4 means that the robot should fulfill the requirement to function optimally, but it is not required to perform the function". Then in each requirement, briefly why it fits the definition of the given priority. That way, the requirements will relate to the overall project better, and will also define the scope of the project a lot better. Another thing that can help with the organization of this section, is to sort the requirements by their given priority. That way, the readers will be able to easily tell which requirements are critical to the robot function, and as a result will be able to better comprehend the overall purpose of the robot.

Another aspect of the requirement section that can be improved is the categorization of functional and nonfunctional requirements. Functional requirements refer to the requirements that the system must do, or characteristics that it must contain in order to perform its intended function, while non-functional requirements specify the quality or attributes with which the system functions. Some of the requirements in the functional section, such as safe, cost under budget, or documentation, do not fall under functional requirements. These requirements are not essential to the robot performing its task; and therefore should be put under non functional requirements. Another example is that mobile app should be a functional requirement, as it describes a requirement directly related to the robotic system performing its mission, albeit its low given priority. Reevaluating and recharacterizing the requirements, together with detailed priorities and justifications of why they are given their priorities, will do wonders in describing the functions of the system, and also setting up the scope of the project.

We also noticed a few grammatical mistakes while reading the specification. For example, in the product goal section, it says "The goal of our project is to use robotic to bring..." when it should say "robots" or "robotics". Additionally, the reload writing implement use case references figure 4.1, which does not exist. Fixing small details like this will greatly improve the readability of the document. We also felt that the executive summary at the beginning did not add anything to the document and therefore could be removed.

Sincerely,  
*Vasu, Vivek, Kim & Tommy*  
We're Bad At This