

### Peer Response Worksheet

I reviewed the code, design documents and doxygen work of Yao Sun and she reviewed mine.

---

1. In the first paragraph of the mainpage, what is the most effective sentence with respect to orienting the reader to the project?

When I read the first paragraph, I found that the most effective sentence about the project is “The goal of this software project is to develop a rudimentary robot simulator in which robot behavior is visualized within a graphics window. The exact use of the application is yet to be determined.” I think this sentence is very useful because these short two sentences explain the purpose of the software very simple and clear way. Also, it gives the reader a general idea where the developer stopped to help them decide to continue working on this project or not.

---

2. Identify a sentence in the first paragraph that needs to be reworked and state what you think is problematic about that sentence. (Do not edit it.)

The sentence: “The intent is for the robot to move around the space searching for robots to freeze.”

My thought: I think the author means the player is moving around the space searching for the robot to freeze because it is a player function, not robot function.

The sentence: “Energy is depleted constantly and is depleted even more when moving or when it bumps into obstacles, but it can renew its energy by going to the charging station.”

My thought: I think the battery will recharge by colliding the recharge station not just going.

---

3. Identify a sentence or two in any of the paragraphs that provides the "big picture" with respect to the software, design, or class structure, AND is accompanied by low-level details that help the reader better understand the "big picture."

"The robot arena has a user-controlled robot, autonomous robots, obstacles, home base and a charging station." This sentence gives the user a big picture of what he/she will see in the arena. The second sentence is "RobotMotionHandler manages the modification to the velocity based on user input and Events." This sentence will show the reader how the mobile entity is moving and how it accepts the action from the sensor. It gives the user a hint that this class one of the important classes of the project to look at.

---

4. Comment on the effectiveness of this technique in the example from (3). If it is effective, analyze why you think it works here. If you think there are other details that would be more elucidating, state those.

I have some comments on both two sentences. The first sentence will be more effective if the author separates the entities into two categories and tells the reader what is the mobile and immobile entities. The second sentence will be more clear if the author writes that the angle of the mobile entity will change after any action by passing the information from the touch sensor to the motion handler. This editing will make the background of the relation between the classes more clear and obvious.

---

5. Identify a topic in the writing that is either underspecified or is discussed too in-depth. If underspecified, what is the most important idea that is missing? If too in-depth, what can be removed?

The author provides very general idea about the software and she did not go in-depth to the details. Moreover, she did not talk about the proximity sensor at all and did not mention how this sensor work and react with other entities. She said, “Autonomous robots that use sensors to avoid objects will move around the environment interfering with play”, I assume that she points to proximity sensor by saying “sensor”. So, if it is the first time for the reader to read this summary, he/she will not understand that it is a proximity sensor.

---

6. What do you think would be the single most impactful change to this document - in other words, what would you recommend to the author as the one area on which to focus? It could be related to the content (e.g. level of detail, more or less technical information, highlight more or fewer classes, etc.) or to the writing (e.g. reorganize paragraph or sentence order, condense text, improve sentence structure, etc.).

The document is very short and simple and gives the reader a short and general background about the project. However, I have some comments and suggestions. Firstly, I recommend her to remove any not important information. For example, writing a brief paragraph about the params and position could make the reader confused and they do not have enough details which could add information to reader's background. Secondly, I really recommend her to talk about what will appear exactly when the game just starts and list the number of the entities. After that, she can provide a detailed explanation of the functionality of each entity, mobile characteristic and included sensors of each entity. Furthermore, if you write a paragraph about a specific entity, it is better to not include any information about other entities. Finally, writing a brief explanation of all event scenarios which could happen in the arena as a list makes the document worthwhile.

---

7. As a programmer new to this project, which class do you think the document is emphasizing as the place to begin to engage the code? This might be explicit or implicit. What part of the writing made you think you should start with that class?

The author mention arena many times, so I think this is the most important class because it has access to all information of the game such as events, entities, and sensors. "Arena: The arena has a user-controlled player, autonomous robots, obstacles, home base, and a charging station. The arena is an area that the player and other moving entities can move around the space," this sentence shows me that the arena is the base of the game and has all essential information.

---

8. What do you consider to be the best and worst documented method in that class and why. OR, if you think they are all of equal quality, comment on the level of detail provided in the documentation. Is it sufficient, clear, and correct? If it is excellent, state what makes it excellent.

In general, the arena class looks like the arena class of iteration 1. The robot is created but commented. Also, she did not provide the function to check the proximity and the distress events. Finally, I notice she uses the `event_commands` enumeration to change the angle of the home base randomly. I did not really understand the how the function works, but I believe it is a creative way to change the angle. Otherwise, she provides very detailed comments explaining to the reader what each function does.

---

9. Skim through all the brief comments on the main classes page. What strikes you as you look at the collection? Is there an effective pattern in the comments? Is there something consistently lacking?

Generally, she did a great job commenting the code. Most of the comment explains what the function exactly does. However, there are some comments not belong to its class. For example, the player class has a brief comment talking about the robot. Also, the motion handler of superbots has a brief comment about the robot too. I believe this mismatch

because of common behavior of most programmer which is copying a code from other classes. Secondly, there are some code and function is listed in the class while it is not used in that class. For example, the robot class has an information about accepting recharge event and some information about the battery. Thirdly, the motion handler class is not abstract class as required. Furthermore, the parent motion handler has some extra information is not used by all child classes. Parent class accepts keypress event while this event is not used by all child. It is just used by the player. So, it is better if we delete this details and make it just inside the player motion handler.

---

10. Where did your eye go? What jumps out at you on the page? Is this an important element, thus warrants the attention? If not, offer a suggestion on how to make it less visually prominent.

Once I looked at her UML diagram, my eyes jumped to the player, home base, and robot. All these classes are in the center of the page. I believe she chose the right position to locate these main classes but I really recommend her to put the Arena on the upper side of the center of the page.

---

11. What did the author do in her/his UML diagram that you would like to incorporate into your UML? Why do you like that part of the UML and how does it differ from what you did?

Her UML diagram is very simple and clear, the reader does not need a long time to recognize what are the main classes and understand the relationship between the classes. However, she did not mention all .h files of the project. For example, the UML diagram does not show any information about the struct and the enumeration classes. So, if the reader wants to build a new project of the Robot Simulator, he/she does not have enough information to do that. I really recommend her to mention the struct file and show the relationship between them. Also, I really recommend her to change the yellow background of the classes because it is one of flame color and make the reading process not comfortable. On the other hand, my UML diagram is very detailed and complex. So, the reader needs to spend significant time to figure out the relationship between the classes.

12. Try to recall your sense of your first attempt to engage the base code, and think of how it is even more complex now. Keeping that in mind, what do you think was the most successful part of the author's writing (in doxygen and UML) with respect to helping a programmer get acclimated to the code? What do you think could be very helpful but needs some rework?

Her UML diagram is very understandable and not messy, but if she adds some details about the relationship as what she did with inheriting relationship. Moreover, if she shows the reader struct and enum classes will make the UML diagram more helpful and worthful. Also, she needs to review the comment of each class to be sure that all comments belong to their classes. Finally, removing the non-important function such as the accepting recharge event of the robot makes the running time of the software faster and the future developers do not need to review cleaning the program before they start working.

---