Peer Reflection of Doxygen Documentation

As the author of the documentation, please compile Doxygen and provide the html folder to your peer responder AND your UML document. You may provide it by posting it on the web (you all have a cselabs web page or you can use some other web server).

RESPONDER: <Saksham Goel>

AUTHOR: <Danyang Wang>

As you reflect on the author's writing, keep in mind the intended audience and purpose of the writing.
- AUDIENCE: A programmer joining the project team who has no familiarity with the project.
- PURPOSE: Provide sufficient information to this programmer to reduce her/his time and effort to be productive and contributing to the project.

Load index.html and read the mainpage text. Answer these questions corresponding to mainpage, considering that the audience would be a programmer unfamiliar with this project. The go:

1. In the first paragraph of the mainpage, what is the most effective sentence with respect to orienting the reader to the project?
   In the first paragraph the most effective sentence would be as follows: "A video game like robot simulator with autonomus, intelligent robot behavior". This sentence clearly states what the objective of the whole project is and what is it that we are trying to achieve for the iteration 2. Also for someone who is reading this design document, they will immediately be able to form a rough image of what the actual work looks like and having this vision when someone joins a team helps a lot in overcoming problems afterwards because of misunderstanding.

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 in question would be as follows: "Battery level is depleted constantly and is depleted even more when moving or when it bumps into obstacles, but it can recharge its battery by going to the recharge station.". I think the wording for this sentence could be improved, but the most problematic thing about this sentence is the placement. This sentence is placed as the second sentence just afterwards the sentence that tells what the project overall is. This sentence definitely needs some context, where we have covered some material regarding user being a player and how the player entity is being depicted in this arena. If this is placed afterwards, when a brief discussion about player has been done, then it would fit perfectly.

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 sentence that helps me understand the bigger picture of the whole project is as given: "In this project, The arena is the class which has access to all the data items. It keeps some vectors to store robots, sensors, obstacles, mobile-entities and entities. In construct we push back various objects into the vectors so that later we can use for loops to go through all the components of the arena. Here, sensors are observers which get information from arena and change the behavior of various entities.".

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.

The sentence in (3) is effective because in a sentence or two it is able to lay out the most important components of the whole game that includes: Arena, "Entities", "Sensors". Also the sentence is able to provide some details about the observer patterns and how it being used to implement the different behaviors of the entities through the arena. This sentence also contains some lower level details regarding the set up of observer pattern using vectors. However there could be some more details that could be added regarding the flow of information from the arena to sensors using the events that is elucidated in the later section. Also there could be have been some mention of how the entities get information from the updated sensors, and how using that the entities get updated. This involved hinting towards the use of entities motion handlers.

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?

Motionhandlers for the different entities are an important part of how the update system works. The passage of information from the Sensors to Entities to their motionhandlers hasn't been discussed in enough detail. I think that this part is essential because it is necessary to understand how updating the sensors itself updates the next state of the entities. Also this information is critical in understanding how the observer pattern between arena and sensor is able to update entities itself. The flow of information is also what helps some other programmer to understand the project and look for possible bugs.

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 most impactful change for the document would be to work on the formatting for the document, as it doesnt seem that professional, and has some inconsistencies like the first characters of the headings are lowercase. Also there is a lot of empty space which doesn't give the right impression to the person who is reading the document. Other than that the author could work somewhat on content. In the discussion of all the sensors and how the events are related to the sensors there are repetitive statements offering clarification. I think it would be better to actually incorporate this information in the writeup and do not offer as a clarification.

  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 really explicit place to begin to engage the code through the writeup is to start understanding the vectors in the arena. These vectors are bring into discussion again and again along with special attention being paid to them in different sections. According to my reading of the design document, these vectors are like the most important part of the code which helps to actually bring everything together and provide a base for everything to work on. After that the other most important part of the code is sensors according to the design document. They are also discussed in length with each type of them being paid special attention. This makes sense because both of these classes make the observer pattern for this project.

  Now explore the documentation of the classes. Go to the class that you identified in (7).

  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.
  The best documented function in the class would be TimestepUpdate(). It has been referenced at a lot of places in the whole design document each focusing on a particular aspect of the function and explaining how the function handles the input to give the desired output for that particular case. Also the function is provided with information about what are the parameters for the function and how it uses these parameters to get information and implement its functionality. I think most of the functions are well documented, and have good enough information for a person to understand how the information is being passed and how is it being processed to give the correct return value or correctly set the state of some other variable.

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?

The documentation seems to be consistent in the sense that whenever it introduces some new terminology, it provides background information about that thing (class, function, member variable) and then build upon the earlier discussion to explain what is the significance of that class/function/variable. Also the document seems to effectively map out the different relationships between classes by introducing these relationships whenever the two classes are referenced for a particular case/context. Like when discussing about different events the document also discusses about the relationship between all the classes with respect to that event (for instance collision and how robot, player and superbot needs to handle the collision event differently).

Now look at the UML - be conscious of your first reaction!

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.

My eyes goes to the top left corner of the UML diagram which contains the biggest class container in the whole UML diagram. Yes this is one of the most important parts of the whole project code base and thus warrants the attention of the person looking at the UML diagram. Other thing that attracts my attention is the Player class container. It is the second biggest container and also is connected to a lot of other classes. This is also good, but I think that the amount of attention that player is getting should be less and on almost equal level of HomeBase, Robot and Superbot

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?

The author was somehow able to portray all the classes and their relationship in one page only which is very effective because the reader is able to draw a rough relationship diagram in their head immediately which includes all the classes and their interaction. I was not able to do this which made my UML spread in 4-5 pages and each page dedicated to special type of relationships between classes. Other than that I like how the UML divides the focus into different parts with each different type of entity taking some different part of the page, which helps draw clear distinction between the classes and still map out their interactions with other 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?

I think the most important part of the authors documentation, design document and UML is the way in which author was able to provide some visual images in the design document which focus in each sensor individually and how it is triggered and updated. These visuals clearly show how the sensors and events are interacting with each other and also how the entities are involved to some degree in this interaction. Other than that these diagrams also help the person understand what the role of each type of sensor is, which clears up a lot of things. Also the discussion which follows the diagrams is very helpful as it provides lower level information about how this type of interaction is being modelled in the project. This could be made much more helpful if the whole section has better formatting such that the person is able to identify the important bits quite distinctly from just looking at the discussion. Also it could be made more effective by adding the discussion on how the motionhandlers are passed the information and how they are using to implement the correct behavior for the entities.