

Assignment #2 Writeup  
Brandon T. Wood  
CSCI E97  
September 30th 2018

## Did creating the design help make the implementation easier?

I believe designing before building is important and possibly the key to making good software. In this case I think planning out in advance was great in a broad overview, but when actually implementing it turned out to be more work. The broad overview glanced over major concerns with the implementation that was going under it.

## How could the design have been better, clearer, or made the implementation easier?

The design gave me a clear goal in my head and really did not change very much. The main problems came from redesigning after finding more information in the documents, special cases in the API and via the lecture. Things like not having to worry about implementing removal of objects which I had already dedicated time to implementing and when I found that the TestDriver would need to take a file that runs CLI queries extended my time to delivery greatly.

## Describe any implementation changes that you made to your design and how they continue to support the requirements.

I did add many small, private, helper methods like `getValue`, `getEntity`, `setAssociation`, etc that smoothed over and abstracted out a lot of the trouble of my datasets manipulation. I did not realize until I had completed most of the work that names had to be unique to the house level. I misheard and thought the opposet so names MUST be unique for all entities(rooms, houses, etc). Recreating this was difficult. Additionally the implementation is case insensitive and quoted variables had to have their spaces replaced with underscores to prevent trouble.

## Did the design review help improve your design?

A bit, my teammates designed their software in a very object-oriented way where I made my data oriented so their input was limited. I think they mostly did not like my diagrams, so much time was spent improving them. Also I found that the only really useful part of the design doc

were the UML diagrams which gave me some idea of the behavior of their programs. The dictionary, as just a list of inputs and outputs is hard to pull any useful information out of.

## Comments about your design from your peer design review partners

Design Review comments for Brandon

- 1) Consider adding the domain classes (house, room, sensor and appliance) to the class diagram and show the relationship between the classes.
- 2) Consider adding some information about the command API and TestDriver class.
- 3) The inclusion of webserver and database information adds more details about the Implementation

(See if you look here you will notice that Sathish did not understand that there are no classes for house, room or sensors as they are just stored as Triples in the knowledge graph.)

Stephen's reviews were via voice chat so let me paraphrase and summarize. He said that my document lacked details, my UML diagram was half baked and that when compared to Sathish's it lacked a lot of detail in the first four sections.

## Comments provided by you for each of your peer design review partners.

I told Sathish over voice chat that his Design Document was excellent and well documented. I suggested that he add more than just the administrator to his uses cases. Stephen's Design Document was going through an overhaul, I pointed out that his introduction and first few sections needed work.