Did the application of the design patterns help or hinder your design and implementation? Please explain how.

The use of the singleton pattern helped the design, because it made accessing some key data easier across all the methods. Those key data do not have to be passed from one method to another, as it can be accessed directly from the singleton.

How could the design have been better, more straightforward, or made the implementation easier?

The “ResourceRole” entitlement is very confusing. The implementation would be more straightforward without it.

Any implementation changes that you made to your design and how they continue to support the requirements

The test driver was modified to read a file that contains a list of script files, and then run all the script files in the order listed. This reorganization made keeping track of the scripts easier, because the single script file would be extremely long and difficult to work with.

Is the design process getting easier?

The design process was getting easier until the authentication assignment. In general, the simulation of a smart city as texts in a java program seems unusual. For example, we talk about sensors getting inputs, but we’re just sending a command from the api to the sensor, instead of sending the input directly to the sensor.

Did the design review help improve your design?

The design review for the authenticator did not really help improve my design.

Your comments for your review partners

My partners did not provide anything for me to review.

Comments from peer design review and optionally the functional review

“Hi Loi, looks great! I think if your role/permission/resource role entitlement instances will have a life cycle of their own you may want to make the association link between them and a user (the open diamond head). Also, I think Eric put in a slide that these would all implement an Entitlement interface and/or a 'visitable' interface. But your diagram overall is super clear”