ICS 372

Prof. Timmothy Carlson

Group 6: GANBY

Refactoring Report

1. Persistence Issue:

The program from the earlier phase included a persistence feature that saved the program's state when exiting, but there were some problems due to our implementation of this feature.

"No dealer information or rental status is being persisted, nor is a dealer persisted if it doesn't have a vehicle." The dealer's acquisition status was also not being saved. These issues occurred because we used the JSONFileExporter to export the program's state, and the JSONFileExporter saved only vehicles and not dealers. Because of this, dealers with no vehicles were not saved, dealer name was not saved, and dealer acquisition status was also

not saved. We also did not account for the rent_status of vehicle when exporting vehicles.

To fix these issues, we used the XMLFileExporter to export the program's state instead of JSONFileExporter. Exporting to an XML file is better than a JSON file because our XMLFileExporter at the time could save both dealers and vehicles. We also included rent status of vehicle in the export. This way, dealers with no vehicles are saved, dealer name and acquisition status are saved, and vehicle rent status is also saved.

2. JSON file processing issues:

We had an issue with JSON processing files, mainly with exporting data to JSON file. JSONFileExporter did not export dealer name, dealer acquisition status, vehicle rent status, and also dealers which have no vehicles. We solved this by including these three attributes into JSON file exports: dealer_name, acquisition_status, and rent_status. When the JSONFileImporter imports JSON files now, it looks for those three attributes and accounts for them, but if they are null in the file they are passed into the builders as null and the vehicle and dealer classes will handle setting default values for them.

Another issue was when a dealer had no vehicles, exporting that dealer to a JSON file would not retain it. This issue happened because the dealer has no vehicles for the JSON to export, so it doesn't end up exporting anything. So now the JSONFileExporter checks if dealer has no vehicles, and if it has no vehicles, it exports a normal JSON file except all fields are null except dealer_name, dealer_id, and acquisition_status. When importing a file that has a dealer with no vehicles, the importer will check if vehicle_id is null, and if it is it will not create vehicle and will only create dealer.

3. DataExporter Class Issue:

"The DataExporter class converts the catalog to JSON, as does JSONFileExporter, but only JSONFileImporter is used for importing. That means code duplication is likely occurring (DataExporter and JSONFileExporter both have a prettyPrintJson() method)." We solved this issue by removing the DataExporter and keeping the JSON exporting functionality for the JSONFileExporter.

4. SportsCar Rental Status Issue:

"Another design issue is how you have implemented not allowing rental on sports cars. You could have either made the UI responsible for knowing that a sports car shouldn't be rented, or your could have made the SportsCar object responsible, but you ended up doing both, which means you have repeated yourself."

We fixed this by removing this functionality from the SportsCar object and giving it only to the UI.

5. GUI Issues:

a. "You have 2 representations of 'selection' in the table views: a checkbox and the selection highlight. These interacted in weird ways (selection highlighting was lost when the tableview changed, but checkboxes stayed checked). I this caused a lot of problems, so think about how that interaction should work for next time." We removed the selection highlight and now require user to check or uncheck checkboxes when wanting to select or deselect dealers and vehicles. If you click on a row it will still highlight it but the code does not listen except to the checkboxes.

b. The button to toggle rent status for vehicles was a bit funky before. It was one button that switched between "set as rented" and "set as available", which caused issues when vehicles selected had different rent statuses. We solved this

by splitting this up into two buttons: “Set as Rented” and “Set as Available” which do what they say to all vehicles selected.

c. The same issue as part(b) was present for the “Change Acquisition Status” button for dealers. We resolved this the same way by splitting that button into two buttons: “Disable Vehicle Acquisition” and “Enable Vehicle Acquisition.”
d. The whole GUI layout was changed. Right now the GUI is split into two horizontal sections, upper section for the dealer table and the lower section for the vehicle table. Because of the previous GUI layout, dealer inventory size could not show unless user scrolls to see, and ditto for vehicle rent status for the vehicle table

6. Alert Messages Issues:

Many UI classes used alert messages, and some classes had repeated definitions of showAlert() method. So we made a new class called AlertHelper that all UI classes call when an alert message needs to be made.