# Changes 3rd iteration:

1. Overhaul of tasks including a major simplification of the class and decoupling them from workstations completely. We also made Task an abstract class and introduced subclasses of Task to represent de different types of tasks.

2. Car now only holds 1 list of tasks (instead of tracking uncompletedTasks seperatly). Tasks have a boolean taskComplete() to represent if the task is complete. Getting all uncompleted tasks now works by getting all tasks of a car that have this boolean set to false. When calling completeTask() you set this boolean in task to True instead of removing it from the uncompleted tasks list.

3. Seperated the UI from the domain layer. Users in the domain layer don't have any knowledge of the UI anymore. Instead, the UI calls functions in the domain layer to request the desired information you want to display.

4. Add defensive programming in places where it was lacking. Also added new unit tests to test the defensiveness of our program.

5. Refactored long methods into more concise and reusable methods. Refactoring also included namechanges for methods to make it more clear what their functionality is.

6. Added documentation in places where it was lacking.

7. Overhaul of the class and sequence diagrams.

8. Bugfixes, mainly in the scheduler and carManufacturingController.