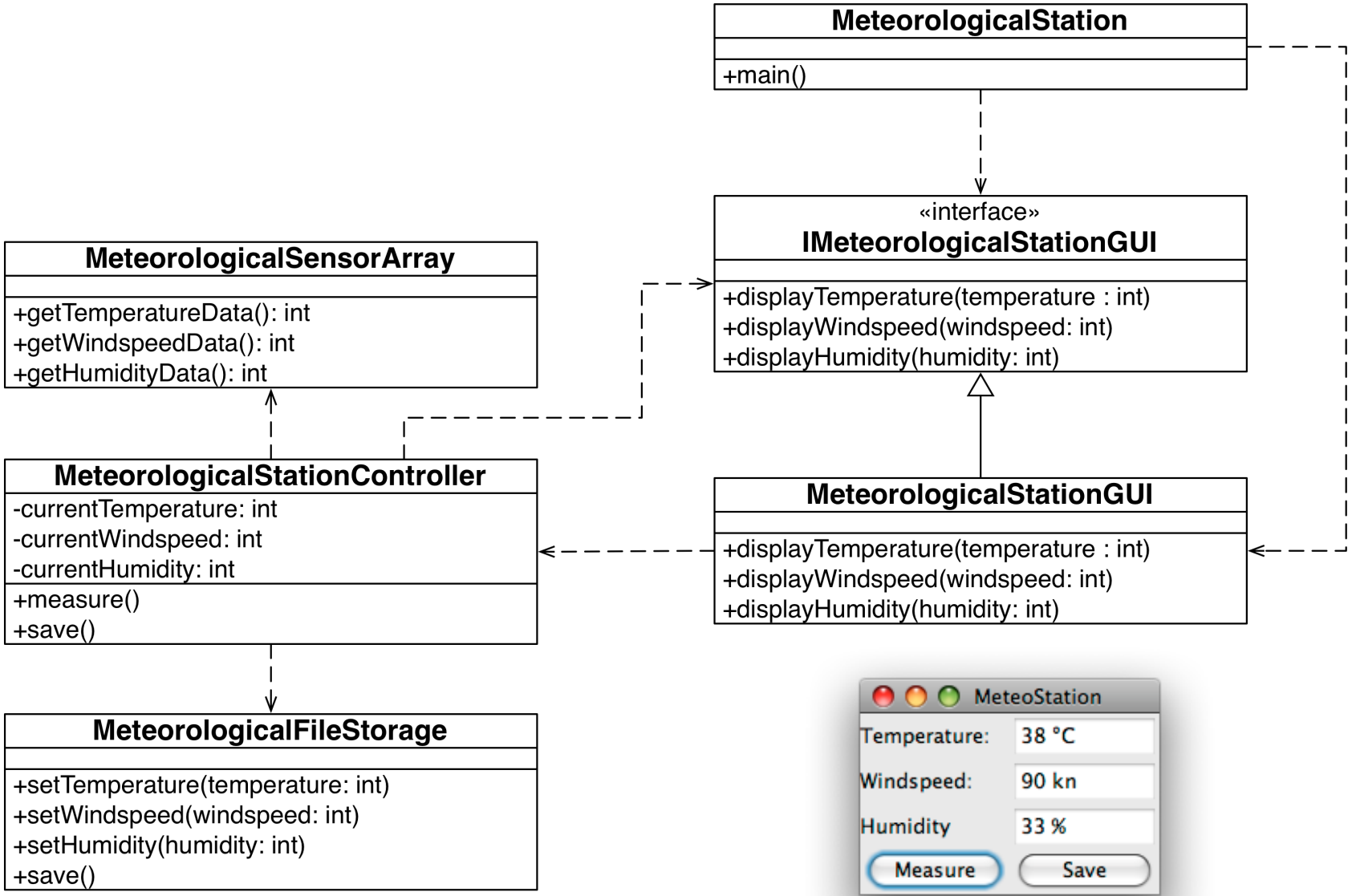
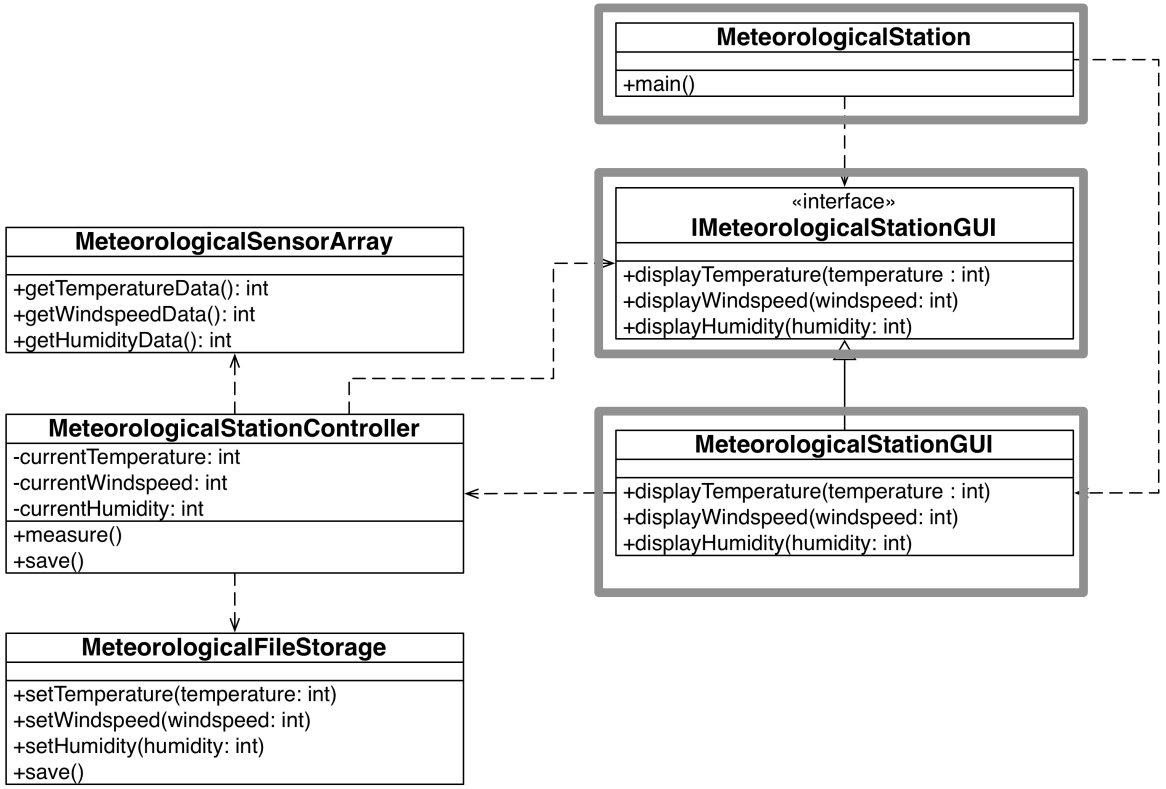
Meteorological Station Application Model



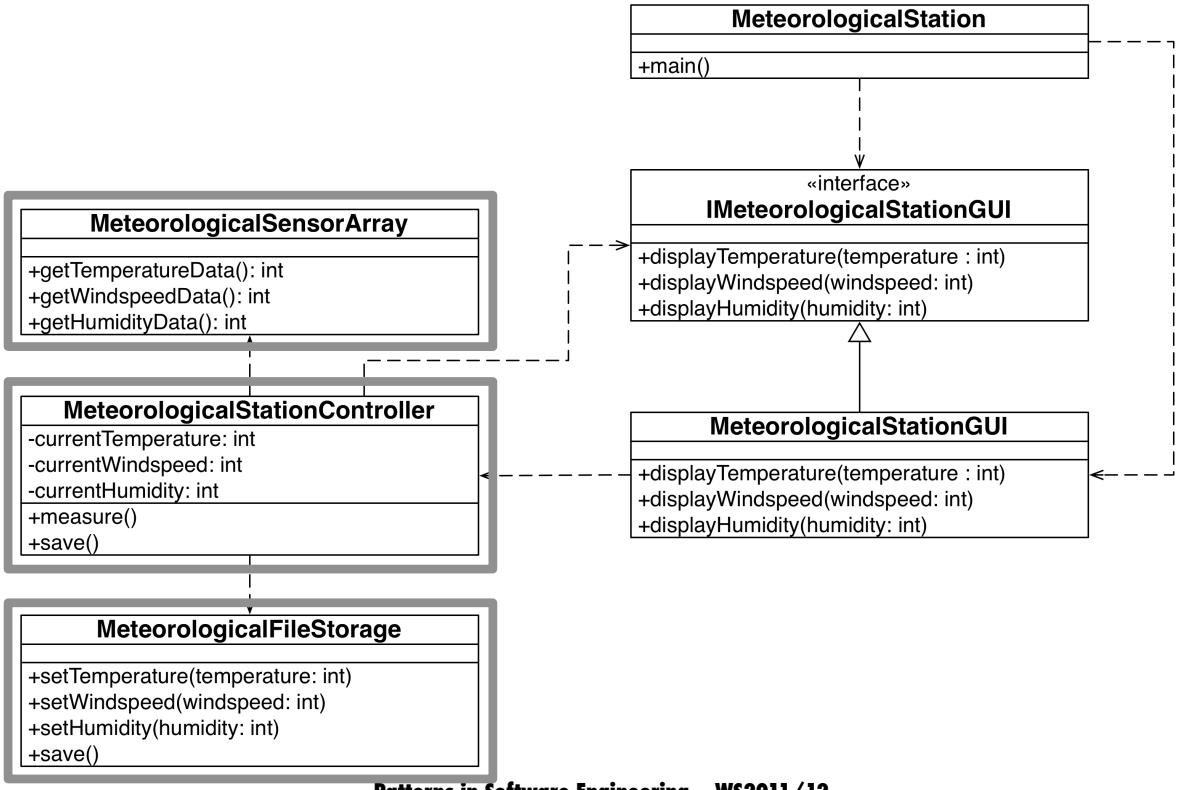
Meteorological Station Implementation Details

- MeteorologicalStation contains the main method and starts up the application
- IMeteorologicalStationGUI declares an interface used to decouple the concrete GUI implementation from the controller. It provides methods that the controller can use to update the GUI
- MeteorologicalStationGUI is a Java Swing implementation of the application's GUI.



Meteorological Station Implementation Details ctd.

- MeteorologicalSensorArray provides us with meteorological data
- MeteorologicalFileStorage can save the meteorological data to a text file
- MeteorologicalStationController contains the application logic and mediates between the model (MeteorologicalSensorArray) and the view (MeteorologicalStationGUI).



Exercises using jUnit, Easy Mock and Guice

- Task #1: Hard (20 minutes):
  - Download the handout code Moodle
  - Remove compile time dependencies in MeteorologicalStationGUI and MeteorologicalStationController where it makes sense
  - Create a "Production" Module to configure the bindings
- Task #2: Harder (Homework):
  - Use the refactored code from Task #1 to unit test MeteorologicalStationController (both the measure() and the save() method)
  - Use EasyMock to mock the collaborators of MeteorologicalStationController (remember to introduce interfaces where necessary)
  - Create a "Test" Module to configure the test bindings
- Post your solution in Moodle.