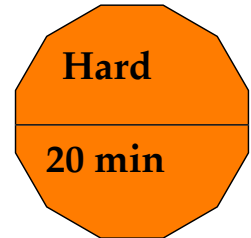# Task #2 (In-Class Exercise) using JUnit, Easy Mock and Guice

**Hard**

**20 min**

- Task #2:
    - 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.

# Meteorological Station Application Model

**MeteorologicalStation**

+main()

**«interface»
IMeteorologicalStationGUI**

+displayTemperature(temperature : int)
+displayWindspeed(windspeed: int)
+displayHumidity(humidity: int)

**Collaborators**

**MeteorologicalSensorArray**

+getTemperatureData(): int
+getWindspeedData(): int
+getHumidityData(): int

**SUT**

**MeteorologicalStationController**

-currentTemperature: int
-currentWindspeed: int
-currentHumidity: int
+measure()
+save()

**MeteorologicalStationGUI**

+displayTemperature(temperature : int)
+displayWindspeed(windspeed: int)
+displayHumidity(humidity: int)

**MeteorologicalFileStorage**

+setTemperature(temperature: int)
+setWindspeed(windspeed: int)
+setHumidity(humidity: int)
+save()

MeteoStation

Temperature: 38 °C
Windspeed: 90 kn
Humidity 33 %

Measure    Save