

## Lab 2: Weather Mapping System-Logical View

### A. Use Case Realization

The realization of a use case is called *collaboration*. A use case is a collection of scenarios, each describing specific aspect of the use case. We consider two scenarios of the *Collect data* use case called *Report Weather Readings* and *Report Air Quality*:

- *Report Weather Readings*: encompasses functionality for reading weather data from remote sources. More specifically weather stations transmit the data collected from weather instruments to the area computer when they receive a request from that machine.
  - *Report Air Quality Readings*: the pollution readings are automatically collected when requested by a weather station and transmitted at the same time as the weather data.
1. The flow of events for a use case is captured in text, whereas scenarios are captured in interaction diagrams. There are two types of interaction diagrams: sequence diagrams that show object interactions arranged in time sequence, and collaboration diagrams, which show object interactions organized around the objects and their links to each other. Both represent an alternate way of describing a scenario.
    - a. Give using Rational Rose the sequence diagram corresponding to the *Report Weather Readings* and the *Report Air Quality* scenarios. (30%)
    - b. Give an alternate representation for the same scenarios using collaboration diagrams. (10%)
  2.
    - a. Identify the boundary, entity and control classes involved in these scenarios. Create these classes using Rational Rose and add if applicable appropriate stereotypes. (10%)
    - b. Class diagrams may also be attached to use cases and contain a view of the classes participating in the use case. Create using Rose a view of participating classes to the *Collect data* use case. (10%)
    - c. Identify the relationships among the classes involved in the scenarios, and create them in the corresponding class diagram using Rose. (10%)
  3. Create the attributes and operations involved in the classes participating in the *Collect data* use case and document them using Rose. The documentation for an operation should state briefly the functionality performed by the operation. It should also state any input values needed by the operation along with the return value of the operation. This information may not be known initially, in which case it should be added later when more is known about the class. The documentation for an attribute should state concisely the purpose of the attribute, not the structure of the attribute. (30%)