CONCORDIA UNIVERSITY

DEPARTMENT OF COMPUTER SCIENCE AND SOFTWARE ENGINEERING

SOEN 331 – Sections S and U:

Formal Methods for Software Engineering

Assignment 3 on UML state machines

Instructor: **Dr. Constantinos Constantinides**, P.Eng.

General information

Date posted: Saturday, 30 March, 2024.

Date due: Wednesday, 10 April, 2024, by 23:59.

Weight: 10% of the overall grade.

Introduction

You must find a partner and between the two of you should designate a team leader who will submit the assignment electronically. The total weight of the assignment is 100 points.

Ground rules

1. This is an assessment exercise. You may not seek any assistance while expecting to receive credit. You must work strictly within your team and seek no assistance for this assignment (e.g. from the teaching assistants, fellow classmates and other teams or external help). You should not discuss the assignment during tutorials. I am available to discuss clarifications in case you need any.
2. Both partners are expected to work relatively equally on each problem. Accommodating a partner who did not contribute will result in a penalty to both. You cannot give a "free pass" to your partner, with the promise that they will make up by putting more effort in a later assignment.
3. If there is any problem in the team (such as lack of contribution, etc.), you must contact me as soon as the problem appears.
4. Make sure you have an up-to-date VPN client on your local machine and you are able to access our electronic submissions system.
5. Late submissions will not be accepted, and no submissions will be accepted by email.

Your assignment

Given the system requirements below, produce the state transition diagram of a UML state machine. A PowerPoint template is provided that gives a partial insight of the hierarchy of states. You must construct your model according to this template.

The system monitors temperature and humidity levels in some facility, going into an emergency mode should any of these levels exceed their corresponding threshold values that the system has been configured with.

Initially ON, at the highest level of abstraction the system can be in one of three possible modes: ON, OFF, or PAUSE.

While ON, the system can become OFF if it receives a 'shut off' event. Reaching the OFF mode concludes the functionality of the system.

If at any moment, while ON, the power goes out, then the system goes into a PAUSE mode while switching into battery power. The system will stay in PAUSE for 10 minutes, after which time if the power has been restored, it will go back to ON, otherwise it will become OFF.

Once the system is ON, it initially lies in an IDLE mode. The system cannot stay in IDLE mode for more than 2 minutes. If this ever occurs, then the system will exit the ON mode.

While at IDLE, the system will automatically become ACTIVE at 6AM, but only if it is already configured. Otherwise, if the system has not already been configured, then it can only become ACTIVE once it receives event 'activate.' There should also be an option to activate the system while skipping the configuration mode, provided that the system has already been configured. Once ACTIVE, the system will automatically go back to IDLE at 11PM. The system can become IDLE anytime if while at ACTIVE mode it receives event 'deactivate.' While being ACTIVE, the system lights a green led. If ever the system leaves the ACTIVE mode, then the green led goes off.

In ACTIVE, the default initial mode is CONFIGURING mode which, once completed, will lead to the core operation of the system, namely MONITORING. During this core functionality, the system can go back to CONFIRURING once it receives a 'set' event. This 'set' event would allow an actor to modify the configuration of the system while the system is executing its core operation.

Configuring the system entails the reception of a sequence of events to set up thresholds for the two variables that the system is expected to monitor: temperature and humidity. If while in CONFIGURING mode, the system detects an inactivity for more than 2 minutes, then the system produces a 'beep' sound and goes into an ERROR mode. The system can return to CONFIGURING upon reception of event 'reset.' This ERROR mode can also serve another purpose: to serve as an alternative exit from the CONFIGURING mode if, upon validating the two variables, any of them is found to be null.

~~Upon entering the CONFIGURING mode, the system must set both variables to null.~~

The system is now at a READY mode, waiting for a value of the first of the two variables. Below is a description of the sequence of events that must occur if the actor decides to first enter a value for the temperature threshold. A similar sequence will occur if the humidity threshold is entered first:

1. The actor sets the threshold level for temperature. At this point, the system goes into a new mode, called WAIT_FOR_H whereby it expects a value for the humidity threshold. If the actor sets yet again a value for temperature, the system will remain in that mode, while having overwritten the current value of temperature threshold with its new value. If the actor now enters a value

for humidity threshold, then the system moves to a VALIDATING mode. Having stayed in VALIDATING for 5 seconds, the system performs a check to see if any of the values are null. If so, then the system must abruptly exit the CONFIGURING mode. Otherwise, if both values are non-null, then the system concludes the configuration process.

2. It has been described above that the system provides the option to enter the configuration process while the system is performing its core operation. This means that the system should not enforce the setting of both variables, but it should allow the setting of just one in cases where the actor wishes to modify only one. This is a description of the sequence of events that must occur if the actor decides to modify the existing value for the temperature threshold. The actor enters a value (as in [1] above), but does not enter a subsequent value for the humidity threshold. Instead, the actor generates a 'done' event, where the system will proceed to validate the value as described in [1] above.

The core operation of the system is done under a MONITORING mode whereby the system starts in mode READING. This is where the system reads the two variables from its external environment and determines normal or critical ranges. If at any moment any of the readings reaches or exceeds its corresponding threshold, then the system moves to an EMERGENCY mode which for the sake of simplicity is not elaborated in this specification.

While in EMERGENCY, the system can resume operation by going into READING upon receiving event 'reset.'

The assumption here is that while in EMERGENCY, external factors will change the status of the environment parameters (temperature and humidity). This means that while in EMERGENCY, once the actor generates a 'reset' event, we should not expect the system to immediately loop to READING and then back to EMERGENCY.

Once in READING mode, the system starts in some IDLE mode. The moment it enters this mode the system reads the values of the two environment parameters and after 10 seconds it must determine if any of the values lie in some medium or high range (not specified). If any of the readings lie in the medium range, the system lights some orange led, otherwise if in the high range, the system will light a red led.

For simplicity, let us take the highest reading into consideration between the two values, i.e. if normal and medium then medium. If medium and high, then high.

If both readings are in a normal range, then there will be no special indication. After the environment parameter levels are determined, the system will go into a ~~READING~~ PROCESSING mode only to repeat the cycle of previous events after 10 seconds. Upon entering ~~READING~~ PROCESSING, the values of the two environment parameters are read but the leds have to be put to off before determining the new ranges (and any subsequent led action can take place).

What to submit

You may use any drawing tool to produce the state transition diagram of the state machine and produce a single pdf file, named after you and your partner, e.g. if Lisa Gerrard and Nick Cave were partners and Lisa were the one to submit, then the submission file is called gerrard-cave.pdf. Submit your assignment at the Electronic Assignment Submission portal at

https://fis.encs.concordia.ca/eas

under **Assignment 3**.

END OF ASSIGNMENT