UML for Embedded Systems

Exam FALL 2019

Software of a glucose monitoring system

Ludovic Apvrille

ludovic.apvrille@telecom-paris.fr

http://soc.eurecom.fr/UMLEmb/

During an exam, you are not supposed to talk with anyone else, by any means (including mobile phones, chat, etc.). Access to Internet is restricted to the website of the UMLEmb course only. You may consult your own UML/SysML models made in the scope of the labs, but not other models. Electronic devices are not allowed at all, apart from the desktop computers of the laboratory room ;-).

A grade is provided for each question. 1 bonus point is awarded for writing quality (report and models).

1 Objective

Your objective is to model the **software of a glucose monitoring system**. This system comprises a glucose sensor and a mobile application: the software of both components must be modeled. This system is simplified with regards to the real system.

You have exactly 3 hours to model this system and answer various questions: the time is very short. This means that **you have to make modeling assumptions**. **Keep your diagrams simple and readable**, in particular the analysis diagrams.

Your grade takes into account your report and your models. At the end of the exam, **reports** (in pdf format) and **models** (in TTool format) **must be sent to me by email**. Also, **the report must be printed** and **given to Alexia Cepero right after the end of the exam session**. The report should contain explanations concerning your models, as well as relevant screen captures of models (e.g., interesting simulation traces, formal verification results).

2 System specification

Again, the system to model is the software which monitors blood sugar level, i.e. the software embedded into the glucose sensor and the one of the mobile application.

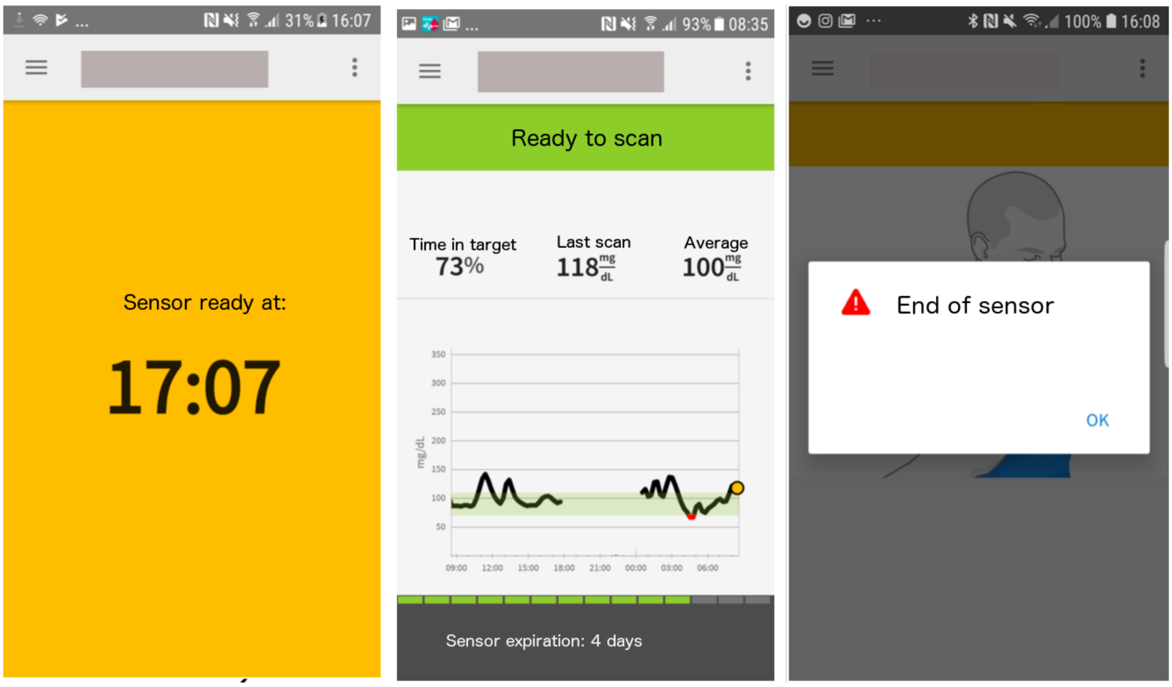
2.1 Description

2.1.1 Overall description

The following system intends to simplify the life of diabetics by helping them in determining their blood sugar level, without having to regularly analyse a blood drop. The system is built upon two main components:

1. A glucose sensor that must be stuck on the skin. The sensor has one electrode that measures blood sugar level. Moreover, this sensor can transmit every 1 mn, via Bluetooth, glycemia level in mg/dL. The battery of this sensor can last up to 7 days after which the whole sensor must be replaced.

2. A mobile application can be downloaded from major marketplaces. The mobile app records all glycemia values sent by the sensor. It also monitors battery level by informing on the number of remaining days before the sensor is deactivated. Glycemia values are stored locally in the application for one month. Also, if the user registers into the website of the company, values can be stored remotely for as long as desired. Statistics are displayed in the app, as shown below in the middle figure.



Sensor ready at:

17:07

Ready to scan

Time in target 73%

Last scan 118 mg/dL

Average 100 mg/dL

Sensor expiration: 4 days

End of sensor

Screen captures of the mobile app

- Right figure: sensor is initializing, so it cannot yet be used to monitor blood sugar level.
- Middle figure: the curve has a discontinuity because the sensor cannot internally save more than 4h of measurements: since the user hasn't connected to the sensor for more than 4h, data has been lost. The "time in target" indicates that the user was in the correct blood sugar level for 73% of the time.
- Right figure: a message informs that the sensor has expired. A new one must therefore be installed before new measures can be performed.

3 Assignments

I. Assumptions

1. Your assumptions should be clear. Do list them in the report: that list might evolve according to the models you make afterwards. Make a clear separation between environment and modeling assumptions. [2 points]

II. Requirements

1. Create a requirement diagram. [3 points]

III. Analysis

1. Make a use case diagram. [3 points]

2. Continue the analysis in the form you want: activity diagrams, nominal scenario, error scenarios, . . . : you are free to use the diagrams you want. Of course, the idea here is to show important points of the specification. [3 points]

IV. Design and validation

1. Make a block diagram. Put the emphasis on which blocks are used to model the system being designed, and which ones are used either to model the environment, or to prove properties (observers). [2 points]

2. Draw state machines, and provide a nominal simulation trace, as well as an error trace. [3 points]

3. Prove that whenever a new glycemia value is available in the sensor and there is an active connection between the sensor and the mobile application, then the glycemia value is updated in the mobile application within one minute (i.e. before a new glycemia value is available in the sensor). Also, from requirements, define a property of your choice, and prove whether it is satisfied (or not!). And obviously, explain how you have modeled those properties [3 points]

Good luck, have fun!