

Praktikumsbeschreibung

Überblick:

Patientenmonitore sind technisch hoch-komplexe Systeme, die eine Reihe von (sicherheits-kritischen) Funktionen sicher durchführen müssen. Dies beinhaltet sowohl die Messung, die Verarbeitung der Patientendaten, als auch deren Darstellung (Visualisierung).

Generell bestehen die Funktionalität des Patientenmonitor aus 3 verschiedenen „Sub-Funktionalitäten“:

1. Messung (Measurement)
2. Alarm (Alarming)
3. Visualisierung (Display) der Vitalzeichen und potentieller Alarme

Das SuD (System under Development) Patientenmonitor hat die folgenden Input- und Outputwerte an der Systemschnittstelle, die für dieses Praktikum wichtig sind:

- Inputwerte: Vitalzeichen des Patienten in Form von systolischem und diastolischem Blutdruck
- Outputwerte: Display auf dem Patientenmonitor (Visualisierung der Patientendaten (systolische und diastolischer Blutdruck inkl. potentieller Alarme).

Das Systems Engineering erfolgt nicht ohne Anforderungen. Diese zu strukturieren, modellieren und dokumentieren ist die Aufgabe einer Anforderungsspezifikation (Requirements-Specification). Drei erlernte Modellierungstechniken zur Anforderungsbeschreibung werden in diesem Praktikum angewendet.



Abbildung 1 Patientenmonitor der Firma Philips

Praktikum 2

Ziel: Es werden unterschiedliche Anforderungen durch die SysML-Diagramme: Aktivitätsdiagramm (inkl. Swimlanes), Sequenzdiagramm, Use-Case Diagramm sowie Requirements-Diagramm entsprechend der Vorlesung im Werkzeug Cameo modelliert und untereinander (auch mit den Modellen der funktionalen Architektur aus Praktikum 1 verbunden).

Aufgabe 1 (Projekt Set-Up)

Importieren Sie das Projekt Patientenmonitor aus den vorherigen Praktika als Referenzprojekt. Erstellen Sie, falls nicht schon im ersten Praktikum geschehen, ein neues SysML- Package („Requirements“ oder „Anforderungen“) innerhalb des Package „System“. Hier sollten jetzt „Funktionaler Viewpoint“ und das neue „Requirements“ Package liegen. Innerhalb des neuen Packages legen Sie folgende weitere SysML-Package an: „01_Use-Case“, „02_Requirements“ an..

Aufgabe 2 (Use-Case Diagramm):

Erstellen Sie ein SysML Use-Case Diagramm das folgendes darstellt:

- Die Funktionen des Patientenmonitors als Use-Case beschrieben (vgl. oben und erstes Praktikum)
- die verschiedenen Aktoren (bspw. Patient, Arzt)
- Nutzen Sie neben Use-Cases mindestens eine <<extends>> und eine <<include>> - Beziehung.

Aufgabe 3 (Aktivitätsdiagramm):

Eine **textuelle Anforderung** aus der Anforderungsabteilung Ihres Unternehmens ist wie folgt beschrieben: „An der Systemschnittstelle kommt ein unverarbeiteter systolischer Blutdruckwert (Name: sysInput) und ein unverarbeiteter diastolischer Blutdruckwert (Name: dialInput) in das System. Aktionen zur „Glättung d. Blutdruckwertes“ arbeiten die noch rohen Inputwerte entsprechend auf. (Diese Aktion ist der Funktion „Messung“ der Funktionalen Architektur zuzuordnen) Diese aufgearbeiteten Blutdruckwerte werden anschließend jeweils durch eine Aktion „Überprüfung Kritikalität“ überprüft. Es wird jeweils entschieden, ob ein Alarm ausgegeben wird oder nicht. (Diese Aktivität entspricht in der Funktionalen Architektur der Funktion Alarming). Wenn der systolische Anteil des Blutdruckwertes größer als 160 mmHg ist und/oder der diastolische Blutdruckwert größer 100, wird jeweils ein Alarmsignal (Name: alarmSysOut) gesendet, bspw. durch die Aktion „Senden eines Alarmsignals“ (entspricht der Display-Funktionalität) und auf dem Display ausgegeben. An der Schnittstelle nach außen werden zudem immer der systolische, als auch der diastolische Blutdruckwert ausgegeben. (Name: displaySysOut, display DiaOut, alarmSys, alarmDia).“

- Beschreiben Sie diese textuelle Anforderung mit einem SysML Activity-Diagramm (Aktivitätsdiagramm), dass Sie für den in Aufgabe 2 beschriebenen Use-Case anlegen!
- Verwenden Sie bei der Modellierung: Aktionen (Actions), Kontroll- und Datenfluss, Decision- und Merge-Nodes, sowie Activity Parameter Nodes für Inputs und Outputs an der Systemschnittstelle.
- Erweitern Sie die Funktionalität, indem sie eine der Aktionen oben, durch ein weiteres Aktivitätsdiagramm verfeinert beschreiben. Welche Aktion erweitert wird, entscheiden sie selbst.

Aufgabe 4 (Swimlanes für Aktivitätsdiagramme)

Erzeugen Sie im Aktivitätsdiagramm sogenannte „Swimlanes“ und ordnen den Spalten der Swimlanes die Sub-Funktionen aus Ihrer Funktionalen Architektur (aus Praktikum 1) zu (SysML Blöcke per Drag-and-Drop). Ordnen Sie die Aktivitäten dann entsprechend den Swimlanes zu. Es sollten keine „gelb umrandeten“ Aktionen geben. Erklären Sie warum das so ist.

Aufgabe 5 (Szenario Diagramm 1)

Zur Beschreibung des oben genannten „Use-Cases“ beschreiben wir diesen zusätzlich mittels eines Szenarios. Dazu beschreiben Sie *eine einzelne Ausführung* der oben beschriebenen Alarm-Funktionalität mittels eines Sequenzdiagramms.

- Legen Sie ein SysML–Sequence Diagram an (Dieses ist in Cameo erst sichtbar, wenn sie „Create Diagram“ ausgewählt haben und dann unten im Fenster „Expert“ auswählen) ebenfalls unter dem Use-Case an.
- Nutzen Sie die Konzepte der Lebenslinie und Interaktion. Überlegen Sie, welche Lebenslinien Sie benötigen und wie Interaktionen für eine Ausführung aussehen könnten. Nutzen Sie ebenfalls die Blöcke aus der Funktionalen Architektur für die Lebenslinien.
- Als Interaktionen verwenden Sie z.B. *Call Messages* (wenn Sie die Sequenz als Methodenaufrufe beschreiben wollen) oder *Messages*, bei der dann konkrete Instanzen der Daten angegeben werden, wie bspw. „159“ (als systolischer Blutdruckwert).
- Zudem verwenden Sie eine *Alternative* mit *Bedingung*, sowie einen „*Duration Constraint*“ als auch einen „*Time Constraint*“ enthalten für Szenarien, die sie sich ausdenken.

Erstellen Sie zu diesem Sequenzdiagramm.

Aufgabe 6: SysML Requirements Diagramm und Relations

Erstellen Sie textuelle Anforderungen mit Hilfe eines SysML-Requirements Diagram für die verschiedenen Use-Cases, die Sie in Aufgabe 2 erstellt haben. Verwenden Sie im Requirements-Diagramm:

- containment-Relationen zw. Anforderungen (wo diese notwendig sind, vgl. Vorlesung)
- mind. eine <<deriveReq>> - Relation

Relations:

- Verbinden Sie den Use-Case zur Blutdruck-Messung (aus Aufgabe 2) im Use-Case Diagram mit der entsprechenden Anforderung im SysML Requirements-Diagramm aus Aufgabe 6 mit der <<refine>>-Relation. Achten Sie auf die richtige Richtung dieser Relation (vgl. Vorlesung).
- Verbinden Sie den Use-Case zur Blutdruck-Messung (aus Aufgabe 2) im Use-Case Diagram mit der entsprechenden System-Funktion aus Ihrer funktionalen Architektur (aus Praktikum 1). Verwenden Sie die <<satisfy>> - Relation. Achten Sie auf die richtige Richtung dieser Relation (vgl. Vorlesung)

Aufgabe 7:

Laden Sie Ihr Modell in den ilias-Abgabeordner hoch.