

Implementierungsdokumentation

**Definition und Durchführung von
Messwertverarbeitung
für den Physikunterricht
auf Basis eines Raspberry Pis**

Version 1.0.0

David Gawron Stefan Geretschläger Leon Huck
Jan Küblbeck Linus Ruhnke

10. August 2019

Inhaltsverzeichnis

1 Ziel der Implementierungsdokumentation

2 Ausarbeitungsstand der Abnahmekriterien

Im Folgenden werden die Muss-, Soll- und Wunschkriterien aus dem Pflichtenheft herangezogen, das in der ersten Phase des Projekts entstanden ist. Es findet eine Bestandsaufnahme statt, inwieweit die Kriterien erfüllt sind.

Falls das Softwareprodukt ein Musskriterium nicht wie im Pflichtenheft beschrieben aufweist, so führt dieses Dokument detailliert die Ursachen und Gründe hierfür auf. Falls das Softwareprodukt ein Sollkriterium nicht wie im Pflichtenheft beschrieben aufweist, so beschreibt dieses Dokument zwar nicht in jedem Detail, aber hinreichend informativ die Ursachen und Gründe hierfür. Nicht umgesetzte Wunschkriterien werden lediglich benannt, aber nicht hinterfragt.

2.1 Entwurf von Messkonfigurationen

Es fand ein Fallback statt. Die Messkonfigurationen werden nicht wie gewünscht graphisch durch ein Drag- and Drop Feld erstellt, sondern müssen textuell eingegeben werden. Dadurch verändert sich auch die Betrachtung, wie und ob die folgenden Kriterien überhaupt erfüllt werden können.

MK 1 Das Musskriterium „Hinzufügen eines Bausteins aus dem Prototypen-Feld zu der Messkonfiguration“ ist TODO

MK 2 Das Musskriterium „Anpassen von wichtigen funktionalen Bausteineigenschaften“ ist TODO

MK 3 Das Musskriterium „Löschen eines Bausteins aus der Messkonfiguration“ ist erfüllt, da der Benutzer die Textuelle Repräsentation eines Bausteins aus der Messkonfiguration entfernen kann.

MK 4 Das Musskriterium „Erstellen einer Verbindung“ ist umgesetzt. Der Benutzer kann ein Kanaltupel der Liste an Verbindungen hinzufügen und somit eine Verbindung der Messkonfiguration hinzu fügen.

MK 5 Das Musskriterium „Löschen einer Verbindung“ ist erfüllt. Der Benutzer kann eine Verbindung aus der Liste der Verbindungen löschen, in dem er das entsprechende Kanaltupel löscht.

SK 1 Das Sollkriterium „Undo-Redo-Funktion“ ist nicht umgesetzt. Die Messkonfiguration wird textuell erstellt und der Editor unterstützt keine Undo-Redo-Funktion.

WK 1 Das Wunschkriterium „Hinzufügen, Bearbeiten und Löschen von ergänzender Informationen zu der Messkonfiguration durch den Benutzer“ ist nicht umgesetzt.

2.2 Handhabung von Bausteinprototypen

SK 2 Der Benutzer ist in der Lage die Eigenschaften der Bausteinprototypen einzusehen. Jedoch ist die Ansicht auf das Anzeigen der *todo* beschränkt. Der Grund hierfür ist die Anbindung von dem Model an die GUI. Dadurch ist es aktuell nur möglich mit den allgemeinen Bausteinprototyp-Informationen zu arbeiten. Dementsprechend ist das Anzeigen der speziellen Eigenschaften, der Bausteinprototypen, nicht möglich.

SK 3 Das Kopieren der Bausteinprototypen ist möglich. Auch das Anpassen der allgemeinen Eigenschaften ist möglich. Jedoch können keine generischen Bausteinprototypen erstellt werden. Dafür wäre die Erstellung einer allgemeinen Vorläge nötig gewesen. Diese Erweiterung hätte dem Nutzer jedoch keine weitere Funktionalität geboten. Aus diesem Grund haben wir uns dazu entschieden die Funktion erst in einer eventuellen Erweiterung der Anwendung zu integrieren.

WK 2 Die Verwendung von erweiternder Software, über eine Schnittstelle, ist nicht mehr vorgesehen. Externe Software kann weiterhin zur Erstellung von Yaml-Dateien genutzt werden. Die so entstandenen Yaml-Dateien können über die Lade-Funktion der Anwendung aufgerufen und anschließend verwendet werden.

SK 4 Die Anwendung unterscheidet in ihrer jetzigen Form nicht zwischen Benutzerdefinierten- und System-Bausteinen. Deshalb ist es auch hier nur möglich die allgemeinen Eigenschaften der Bausteinprototypen zu ändern.

SK 5

2.3 Gewährleisten von Persistenz

MK 6 Das Kriterium ist erfüllt. Die verwendeten Bausteine und ihre Anordnung, also die Messkonfiguration, kann in einer Datei gespeichert werden.

MK 7 Das Kriterium ist erfüllt. Messkonfigurationen können aus Dateien geladen werden.

SK 6 Das Kriterium ist nicht erfüllt, da keine neuen Prototypen in der Anwendung erstellt werden können. (Siehe SK 3)

SK 7 Bausteine werden aus Dateien geladen, die jedoch nur außerhalb der Anwendung erstellt werden können.

2.4 Bereitstellung vorgefertigter Teile

2.5 Handhabung von Messläufen

2.6 Benutzbarkeit der GUI

SK 11 Die Funktionalität Aktionen per Drag-and-Drop durchzuführen ist in der aktuellen Version der Anwendung nicht implementiert. Die Gründe hierfür sind die lange Einarbeitungszeit in ein externes Editor-Programm, welches ermöglicht Objekte über Drag-and-Drop zu bewegen. Dies würde das Auswählen eines Editor-Programms beinhalten, die Einarbeitungszeit, die Implementierung und Integration dieses Programms beinhalten. Durch ein zu spätes Festlegen auf ein Editor-Programm, *JHotDraw* hat die Zeit für die Einarbeitung, Implementierung und Integration gefehlt. Deswegen haben wir uns aus Zeitgründen dagegen entschieden ein Editor-Programm zu implementieren. Leider entfällt dadurch eine grundlegende Funktionalität unserer Anwendung, welche bereits fest vorgesehen war und die Benutzung der Anwendung vereinfachen und verbessern würde. Als Weiterentwicklung hat dieses Kriterium eine hohe Priorität.

SK 12 Die Anwendung enthält ein Hilfe-Fenster, welche dem Benutzer eine kurze Beschreibung der Funktionalität der Anwendung und Information über die Anwendung bieten. Die Informationen zu den GUI-Elementen lassen sich jedoch nur aus dem Hilfe Fenster-Text auslesen und nicht interaktiv über Bedienung der GUI-Elemente.

SK 13 Die Anwendung bietet dem Benutzer Information über Fehler-Rückmeldungen über ein Fehlerfenster. Vor Fehlverhalten wird nicht gewarnt sondern nur reaktionär auf Fehler reagiert. Die Implementierung des Vorwarnen vor Fehlverhalten hätte sich durch konstante Analyse der Benutzereingaben als zeitintensiv und komplex erwiesen und wurde deswegen nicht umgesetzt.

WK 6 Die Anwendung ist in deutscher Sprache. Das Sprach-Paket lässt sich in der jetzigen Version nicht ändern.

WK 7 Die Festlegung auf ein Konfigurationsfeld, welche eine Konfiguration in schriftlicher Form erstellen lässt führt dazu, dass es keine visuelle Repräsentation von Bausteinen und deren Ein- und Ausgänge gibt.

WK 8 Die Anwendung legt kein festes Farbschema fest und somit ist das Farbschema nicht anpassbar. Die Implementierung ist vorgesehen, aber war zeitlich nicht umsetzbar.

WK 9 Die Anwendung legt eine feste Schriftgröße fest, die sich durch die feste Implementierung der GUI-Tools festlegt und ist somit nicht änderbar. Die Implementierung ist vorgesehen, aber nicht umgesetzt.

2.7 Abgrenzungskriterien

3 Umsetzung des Entwurfs

Während der Entwurfsphase wurden sowohl UML-Klassendiagramme als auch UML-Sequenzdiagramme erstellt. Zusammen mit der textuellen Beschreibungen der zu erstellenden Software-Elemente bildeten diese die Basis für die Produktion des Quellcodes während der Implementierungsphase.

In aller Regel lassen sich abstrakte Entwurfsinhalte während der Implementierung nicht in allen Details exakt umsetzen, was verschiedene Gründe haben kann. Bisweilen entpuppt sich auch eine andere Umsetzung als vorteilhafter. Die folgenden Abschnitte halten für jedes Softwaremodul die Abweichungen der Implementierung gegenüber den im Entwurf beschriebenen Strukturen fest. Des Weiteren enthalten sie die Gründe für diese Abweichungen.

3.1 Model

3.1.1 Ersetzen des Entwurfsmuster Erbauer durch eine Zuständigkeitskette

Das Paket „Model.BuildingBlockBuilder“ im Entwurf wurde durch das Paket „model.block“ ersetzt. Das dort verwendete Entwurfsmuster Erbauer erfüllte nicht die notwendige Anforderung, dass der Benutzer leicht eigene Versionen von Bausteinen in die Anwendung einfügen konnte. Darum wurde der Erbauer durch eine Zuständigkeitskette ersetzt. Hier gibt es keine Methode für jeden Baustein im Director, sondern es gibt nur eine Anzahl von Bearbeitern, die einen Block eines Types erstellen. Wenn also der Benutzer eine eigene Transformation erstellen will, kann er die .yaml Datei einer bereits vorhandenen Transformation kopieren und einige Parameter (außer Typ und subtyp) verändern. Die resultierende Transformation wird dann von der Anwendung als eine erkannt und kann dann auch dort verwendet werden. Dadurch entfallen alle folgenden Klassen des Entwurfs:

Builder

TransformationBuilder

RepresentationBuilder

XYRepresentationBuilder

TableRepresentationBuilder

SensorBuilder

VirtualSensorBuilder

PhysicalSensorBuilder

SnakeYamlParser

java.util.hashmap

sowie all diese öffentlichen Methoden in der Director Klasse:

createSensorFromYaml

constructTransformation

`constructXYRepresentation`

`constructNTimeRepresentation`

`constructDS18B20TemperatureSensor`

`constructBMPx80PressureSensor`

`constructINA219CurrentAndVoltageSensor`

`constructMMA8451Accelerometer`

`constructTransformation`

Statt dessen wurden folgende Klassen hinzugefügt:

`GeneralBlockKvProcessor`

`KvProcessor`

`SensorKvProcessor`

`PhysicalSensorKvProcessor`

`VirtualSensorKvProcessor`

`RepresentationKvProcessor`

`TableRepresentationKvProcessor`

`XYRepresentationKvProcessor`

`TransformationKvProcessor`

und die Methode `constructBuildingBlock` zum Director und zu jedem Bearbeitern die Methode `processKvPair` hinzugefügt. Dabei unterscheiden sich die Methoden der einzelnen Bearbeitern zwar nicht im Namen, aber in ihrer Funktion. Jeder Bearbeiter leitet entweder die Anfrage weiter oder erstellt einen Blocktyp und gibt ihn zurück.

3.1.2 Auswirkungen des Fallbacks auf das Model

Nach der Hälfte der Implementierungszeit haben wir uns entschlossen die graphische Drag- and Drop-Funktion aufzugeben, um einen funktionierenden Prototypen entwickeln zu können, der einen Datenfluss zwischen allen Modulen der Anwendung realisiert. Wir haben die graphische Benutzeroberfläche stark vereinfacht und die Bearbeitung der Messkonfiguration erfolgt nun textuell. Dadurch sind viele Klassen und Methoden im Model nicht mehr notwendig oder wurden für die neuen Funktionen angepasst. Die Benutzbarkeit und Benutzerfreundlichkeit wird durch den Fallback zwar reduziert, aber ein funktionierender Prototyp hatte mehr Priorität.

Das Entwurfspaket `model.channellogic` hat nun in der Implementierung den Namen `model.channel`. Die Klassen `Channelstate`, `Connected`, `UnConnected` und `Value Ready` des Entwurfs haben keine Funktion mehr und werden daher verworfen. Dadurch sind auch die entsprechenden Methoden `connect`, `disconnect`, `receiveValue`, `eraseValue` in der Klasse `Channel` verworfen worden. Außerdem wurde die Klasse `Channel` abstract gemacht. Die Methode `establishConnection` wurde je einmal für jede Kanalart in den Unterklassen implementiert. Dadurch kann der Versuch, eine Verbindung zwischen zwei gleichen Kanaltypen zu erstellen, verhindert werden.

Die Entwurfspakete `model.sensorLogic`, `model.transformationLogic` und `model.representationLogic` wurden zu den Implementierungspaketen `model.sensor`, `model.transformation` und `model.representation` umbenannt. In den drei Paketen wurden alle Entwurfsklassen beibehalten. Allerdings haben die Klassen `VirtualSensor`, `PhysicalSensor`, `TableRepresentation` und `XYRepresentation` keine Funktion, außer um möglichen zukünftigen Bausteinen eine Grundlage zur erweiterten Kategorisierung zu bieten. Diese vier Klassen haben ihre explizite Methode `processValue` verloren, da sich diese von Implementierung der jeweiligen Oberklasse nicht unterscheidet. Außerdem sind die drei Klassen `Sensor`, `Transformation` und `Representation` in der Implementierung nicht mehr abstrakt. Alle Bausteinklassen haben außerdem einen Copy-Constructor erhalten um tiefe Kopien anlegen zu können.

Die Klasse `Function` aus dem Paket `model.transformation` hat ihre Methode `applyFunction` verloren und erstellt nun nur noch die benötigte Funktion aus den Daten der .yaml Datei. Die Funktionen der Methode `applyFunction` wurde in die Methode `processValue` der Klasse `Transformation` eingefügt, so dass der reinkommende Wert schneller verarbeitet und weiter gesendet werden kann.

Die Methode `processValue` aller Bausteine verarbeitet nun keine Pakete mehr sondern nur noch deren Werte.

3.2 GUI

3.2.1 GUI-Paket

3.2.2 Configuration-Paket

3.2.3 BlockProperties-Paket

3.2.4 Exception-Paket

3.2.5 HelpAndOption-Paket

3.2.6 Menu-Paket

3.2.7 Model-Interface-Paket

3.2.8 Controller-Interface-Paket

3.3 Controller

3.4 Backend

3.4.1 SSH-Verbindung

Für die Kommunikation mit dem Raspberry Pi wird eine SSH-Verbindung verwendet. So können über das lokale Netzwerk Python-Skripte auf dem Raspberry Pi ausgeführt und Dateien kopiert werden.

Die SSH-Verbindung ist mithilfe der Bibliothek *SSHJ*¹ implementiert. Infolge dieser Technologieentscheidung sind folgende Klassen überflüssig geworden: `SystemProcessCommandLine`, `SshCommandGetSensorIds`, `SshCommandCopyFromPi`, `SshCommandCopyToPi`.

Die Aufgaben dieser Klassen werden stattdessen durch ihre vorgesehenen Oberklassen (`CommandGetSensorIds`, `CommandCopyFromPi`, `CommandCopyTo`) und direkt durch `SshToPi` erfüllt.

¹<https://github.com/hierynomus/sshj>

Nach außen ist der Zugriff auf das Backend über die Schnittstellen `IAccessToSensorInfo` und `IAccessToMeasurementRun` und die implementierenden Klassen `SensorInfoAgent` und `MRunAgent`, sowie die Klasse `PickupPointForBackendAgents` möglich. Diese wurden wie im Entwurf vorgesehen umgesetzt.

3.4.2 Simulierte Verbindung

Anstelle einer echten SSH-Verbindung kann das Backend auch simuliert werden. Dazu wurden die Klassen `ComToFile` und `FileCommandFactory` eingeführt. Es können dadurch Messdaten aus einer zuvor erstellten Datei ausgelesen werden.

Die Simulation des Backends ist nur eingeschränkt funktionstüchtig, da die Verbindung mit einem echten Gerät in der Entwicklung eine deutlich höhere Priorität hatte.

3.4.3 Nebenläufigkeit

Das Auslesen von Daten läuft parallelisiert ab. Dabei wird für jeden Sensor ein eigener Thread aufgebaut. (Nicht, wie im Entwurf angedeutet, für jeden Kanal.)

Jeder dieser Threads verwendet eine Instanz der Klasse `MeasurementRunnable`, welche die Klassen `CommandMRun` und `SshCommandMRun` aus dem Entwurf ersetzt. Über diese *Runnables* wird der Messlauf gesteuert (unter anderem pausiert und angehalten). Die Klasse `MRunThread` ist dabei überflüssig geworden.

3.5 Cache

3.6 File-Service

4 Realer Implementierungsablauf

Dieser Abschnitt führt auf, inwieweit der tatsächliche zeitliche Implementierungsablauf vom geplanten Ablauf abgewichen ist, und beschreibt die Ursachen und Gründe für diese Abweichungen. Abhängigkeiten zwischen den Implementierungsschritten und kritische Pfade stehen hierbei besonders im Fokus.

Von Abweichungen betroffene Softwareelemente werden nicht im Einzelnen aufgeführt, sondern es werden lediglich in Bezug auf die Abweichungsgründe die Gruppen der betroffenen Softwareelemente benannt.

5 Anhang

	Woche	27							28							29							30							31							32						
	Tag	07. Jul	08. Jul	09. Jul	10. Jul	11. Jul	12. Jul	13. Jul	14. Jul	15. Jul	16. Jul	17. Jul	18. Jul	19. Jul	20. Jul	21. Jul	22. Jul	23. Jul	24. Jul	25. Jul	26. Jul	27. Jul	28. Jul	29. Jul	30. Jul	31. Jul	01. Aug.	02. Aug.	03. Aug.	04. Aug.	05. Aug.	06. Aug.	07. Aug.	08. Aug.	09. Aug.	10. Aug.	11. Aug.						
Module																																											
Abstract																																											
Verantwortung																																											
Controller																																											
Builder Interface																																											
Model Information																																											
Measurement Unit																																											
View Controller Interface																																											
Robot Action																																											
Block Action																																											
Connection Action																																											
Command Pattern																																											
Command Manager																																											
Command																																											
AddBlockToConfigCommand																																											
ModifyBlockConfigCommand																																											
RemoveBlockFromConfigCommand																																											
EnableBlockPropertyCommand																																											
DisableBlockPropertyCommand																																											
SaveConfigCommand																																											
LoadConfigCommand																																											
CreateChannelConnectionCommand																																											
ModifyChannelConnectionCommand																																											
StartRunCommand																																											
StopRunCommand																																											
ResumeRunCommand																																											
PauseRunCommand																																											
Service																																											
Service																																											
Service																																											
Service																																											
GUI																																											
Linux																																											
Menu																																											
MenuWindow																																											
PrototypeField																																											
TransformationBlockField																																											
RepresentationBlockField																																											
FieldHandler																																											
Configuration																																											
ConfigurationField																																											
BlockToBlockDiagramHandler																																											
BuildingBlockView																																											
SensorBlockView																																											
TransformationBlockView																																											
RepresentationBlockView																																											
RobotControllerView																																											
AddWindowAndDiagramHandler																																											
RemoveWindowAndDiagramHandler																																											
ChannelConnector																																											
ChannelDescriptor																																											
ChannelDescriptor																																											
Wire																																											
Building Block Properties																																											
BuildingBlockProperties																																											
BuildingBlockPropertiesHandler																																											
SensorBlockProperties																																											
TransformationBlockProperties																																											
ChannelBlockPropertiesHandler																																											
RepresentationBlockProperties																																											
RobotBlockPropertiesHandler																																											
Button																																											
Button																																											
ButtonHandler																																											
Exception																																											
ExceptionWindowHandler																																											
ExceptionWindow																																											
BuildingBlockExceptionWindow																																											
ConnectionExceptionWindow																																											
GeneralExceptionWindow																																											
RobotOption																																											
OptionWindow																																											
RobotWindow																																											
RobotWindowHandler																																											
RobotWindowHandler																																											
RobotWindowHandler																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
RobotAction																																											
BlockAction																																											
ConnectionAction																																											
FacadeMapView																																											
FacadeMapView																																											
David/Loren																																											
Core																																											
MeasurementUnit																																											
MeasurementUnit																																											
MeasurementConfiguration																																											
BuildingBlock																																											
RobotMessage																																											
TransformationLogic																																											
TransformationLogic																																											
Function																																											
TransformationLogic																																											
Representation																																											
Representation																																											
Representation																																											
Sensor Logic																																											
Sensor																																											
PhysicalSensor																																											
VirtualSensor																																											
Channel Logic																																											
Channel																																											
ChannelState																																											
Connected																																											
Disconnected																																											
ValueReady																																											
Channel																																											
OutChannel																																											
Building Block Builder																																											
Director																																											
Builder																																											
BuilderManager																																											
SensorBuilder																																											
PhysicalSensorBuilder																																											
VirtualSensorBuilder																																											
TransformationBuilder																																											
RepresentationBuilder																																											
RobotRepresentationBuilder																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											
FacadeControllerView																																											

6 Glossar