

Eric A. Nyamsi

# Projektmanagement mit Scrum

Tools zur Entwicklung von Software

 Springer Vieweg



# Projektmanagement mit Scrum

---

Eric A. Nyamsi

# Projektmanagement mit Scrum

Tools zur Entwicklung von Software

Eric A. Nyamsi  
Karlsruhe, Deutschland

ISBN 978-3-658-27485-6      ISBN 978-3-658-27486-3 (eBook)  
<https://doi.org/10.1007/978-3-658-27486-3>

Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über <http://dnb.d-nb.de> abrufbar.

Springer Vieweg

© Springer Fachmedien Wiesbaden GmbH, ein Teil von Springer Nature 2019

Das Werk einschließlich aller seiner Teile ist urheberrechtlich geschützt. Jede Verwertung, die nicht ausdrücklich vom Urheberrechtsgesetz zugelassen ist, bedarf der vorherigen Zustimmung des Verlags. Das gilt insbesondere für Vervielfältigungen, Bearbeitungen, Übersetzungen, Mikroverfilmungen und die Einspeicherung und Verarbeitung in elektronischen Systemen.

Die Wiedergabe von allgemein beschreibenden Bezeichnungen, Marken, Unternehmensnamen etc. in diesem Werk bedeutet nicht, dass diese frei durch jedermann benutzt werden dürfen. Die Berechtigung zur Benutzung unterliegt, auch ohne gesonderten Hinweis hierzu, den Regeln des Markenrechts. Die Rechte des jeweiligen Zeicheninhabers sind zu beachten.

Der Verlag, die Autoren und die Herausgeber gehen davon aus, dass die Angaben und Informationen in diesem Werk zum Zeitpunkt der Veröffentlichung vollständig und korrekt sind. Weder der Verlag, noch die Autoren oder die Herausgeber übernehmen, ausdrücklich oder implizit, Gewähr für den Inhalt des Werkes, etwaige Fehler oder Äußerungen. Der Verlag bleibt im Hinblick auf geografische Zuordnungen und Gebietsbezeichnungen in veröffentlichten Karten und Institutionsadressen neutral.

Springer Vieweg ist ein Imprint der eingetragenen Gesellschaft Springer Fachmedien Wiesbaden GmbH und ist ein Teil von Springer Nature.

Die Anschrift der Gesellschaft ist: Abraham-Lincoln-Str. 46, 65189 Wiesbaden, Germany

---

## Vorwort

Scrum ist das aktuelle gefragte Vorgehensmodell zum Erstellen von Software. Die Grundlagen eines Software-Produkts auf Basis von Scrum basieren auf der Erstellung der Anforderungen. Projektmanagement auf Basis von Scrum ermöglicht Lieferungen von Produkten mithilfe von agilen Praktiken.

Bei der Erstellung von Projekten mithilfe von Scrum stehen hauptsächlich Werte und Prinzipien im Mittelpunkt. Ziel der Anwendung des Scrum-Vorgehensmodells in der Entwicklung der Software ist es, eine kürzere Zeitspanne zwischen Produktidee und Auslieferung zu ermöglichen.

Anforderungen bestehen aus dem „Product Backlog“, welches der Darstellung des Produktplans dient. Die Produktplanung ermöglicht das Erstellen des Produkts. Wobei ein agiles Projekt mithilfe eines Produkts definiert wird.

Das Buch fokussiert auf die innere Qualität von Software mithilfe der Programmierung der Anwendungen für die Automobilbranche auf Basis von der Programmiersprache Java, deren Entwurfsmuster genannt Design Pattern wie z. B. Model View Controller(MVC)- oder Delegate-Pattern zur Anwendungsentwicklung implementiert werden. Einerseits zeigt die innere Qualität der Software den Programmieraspekt und andererseits hängt sie von Lesbarkeit, Wartbarkeit, Erweiterung und Wiederverwendbarkeit des Produkts ab. Die Codierung ermöglicht die Analyse der inneren Qualität der Software. Das Design Pattern MVC ist ein der meist benutzten Entwurfsmuster bei der Software-Entwicklung. Seine Implementierung mithilfe des Scrum-Vorgehensmodells ermöglicht seine Wiederverwendbarkeit bei der Softwareentwicklung. Dies erhöht die innere Qualität der Software. Die Anwendungen auf Basis von Oberflächen sind meist mit dem Entwurfsmuster MVC realisiert, weil sie eine höhere innere Qualität hinweisen.

Die Entwicklung von Software hängt auch von der Modellierung ihrer Komponenten oder Klassen ab. Wobei diese Modellierung Beziehungen zwischen Komponenten oder Klassen darstellt. Die Analyse der inneren Qualität der Software liegt in der Umsetzung des Konzepts „Modeling4Programming“, welches das Erzeugen der Codes aus dem Klassendiagramm von Modellen ermöglicht. Hierbei verfügt jedes Klassendiagramm über Codes. Vom Modellieren zum Programmieren auf Basis vom Eclipse Modeling Framework(EMF) ist ein Trend und trägt zur Analyse der Qualität der Entwicklungsarbeit bei.

Das Konzept zeigt, dass aus einem Klassendiagramm mehrere Zeilen von Codes erzeugt werden. EMF ermöglicht die Generierung von Codes mithilfe des Java Frameworks Ecore. Wie sieht die innere Software aus? Die Modellierung des Klassendiagrammes auf Basis von dem Ecore Framework beantwortet diese Frage. Welche Funktionalitäten sind implementiert? Die Modellierung des Klassendiagrammes auf Basis von EMF beantwortet diese Frage. Das Beseitigen der Fehler bei der Softwarereparatur erfordert das Analysieren der Struktur des Klassendiagrammes. Wobei das Modellieren sowohl des Klassendiagrammes als auch dessen Entwurfsmusters eine höhere Qualität der Software erzeugt. Ein geeignetes Entwurfsmuster ermöglicht ein effizientes Beseitigen der Fehler bei der Wartbarkeit der Software und damit eine höhere Qualität. Ohne Modellierung der inneren Teile der Software gibt es keine effiziente agile Software-Entwicklung. Modellierungen der Beziehungen zwischen den Klassen der Software können Hinweise über die Akzeptanz des Produkts geben, weil die inneren Teile der Software nur von den Programmierern sichtbar sind und damit von ihnen bei der Wartungsarbeit analysiert werden. Die Entwickler entscheiden selbst welche Entwurfsmuster für die Software-Entwicklung geeignet sind.

Java ist eine der starken Programmiersprachen zur Entwicklung von Anwendungen sowohl für Business als auch für Forschungen oder Technik mithilfe von verschiedenen Entwurfsmustern bei Vorgehensmodellen. Scrum ist auf dem Vormarsch und überzeugt die Entwickler bei der Suche nach Vorgehensmodellen, die ihnen die Freiheit zur Auswahl geeigneter Entwurfsmuster wie z. B. MVC- oder Delegate-Pattern für die Entwicklung der Software geben.

Karlsruhe, Deutschland, März 2019

Eric Aristhide Nyamsi

---

## Danksagung

Ich möchte mich für die Zusammenarbeit mit dem Springer-Vieweg Verlag bedanken, insbesondere bei Frau Andrea Broßler und Herrn Reinhard Dapper.

Karlsruhe, März 2019

Eric Aristhide Nyamsi

---

# Inhaltsverzeichnis

<b>1</b>	<b>Projektmanagement mit Scrum</b> . . . . .	1
1.1	Einleitung . . . . .	1
1.1.1	Anforderungsmanagement . . . . .	2
1.1.2	Oberflächenentwicklung für das Anforderungsmanagement . . . . .	2
1.1.3	Design Pattern Delegate mit Java . . . . .	3
1.1.4	Webentwicklung mit dem Open Source RapidClipse . . . . .	4
1.1.5	Webentwicklung mit JavaServer Faces(JSF) . . . . .	5
1.1.6	Ecore Modellierung . . . . .	5
	Fazit . . . . .	6
<b>2</b>	<b>Entwicklung von Automobilbranche-Anwendungen mit Java</b> . . . . .	7
2.1	Anforderungsmanagement für die Automobilebranche . . . . .	7
2.1.1	User Story zum Anforderungsmanagement . . . . .	8
2.1.2	Storyboard . . . . .	9
2.1.3	Product Backlog . . . . .	10
2.2	Rolle des Anforderungsmanagements . . . . .	10
2.2.1	ReqIF als Model für Anforderungen . . . . .	11
2.2.2	Das ReqIF-Datenmodell . . . . .	12
2.2.3	Modellieren der Automotive-Anwendungen mit Requirement Toolings . . . . .	13
2.3	Entwicklung von Automotive-Anwendungen mit Java XDEV für Feature-Modellierung . . . . .	29
2.3.1	Anforderungsmanagement der Softwareentwicklung für Fahrzeuge . . . . .	29
2.3.2	GUI-Entwicklung für das Anforderungsmanagement . . . . .	30
2.3.3	Design im Projektmanagement . . . . .	38
2.4	Design-Pattern Delegate auf Basis von Java für die Energieversorgung . . . . .	59
2.4.1	Nutzung der Design-Patterns mit Java in der Energieversorgung . . . . .	60
2.4.2	Design-Pattern „Delegate“ für die Berechnungen der Energieverluste . . . . .	61
2.4.3	Parallele Programmierung für das Design Pattern „Delegate“ . . . . .	85

Fazit.....	91
Literatur.....	92
<b>3 Anwendung der 3-Schichte-Architektur vom Java Framework</b>	
<b>Rapidclipse in den Business Applikationen.....</b>	<b>93</b>
3.1 Warum Rapidclipse für Java Framework? .....	93
3.2 Drei Schichten-Architektur des Java Frameworks Rapidclipse .....	94
3.2.1 Client Layer .....	96
3.2.2 Application Layer .....	102
3.2.3 Persistence Layer.....	107
Fazit.....	109
Literatur.....	109
<b>4 Visualisierung der Anwendungsentwicklung mit dem Java Framework</b>	
<b>Rapidclipse.....</b>	<b>111</b>
4.1 Rapidclipse als Teil vom Eclipse Java Framework .....	111
4.1.1 Überblick über das Erbe des Java Frameworks Eclipse.....	111
4.1.2 Komponenten des Java Frameworks Rapidclipse.....	113
4.2 Erste Schritte mit dem Java Framework Rapidclipse .....	114
4.2.1 Download.....	115
4.2.2 Start .....	116
4.3 Java EE mit Rapidclipse .....	116
4.3.1 JSF zur Web-Entwicklung.....	116
4.3.2 Managed Bean oder Java Bean .....	117
4.3.3 Facelets oder XHTML-Dokumente.....	122
4.3.4 Integration von HTML in JSF .....	124
Fazit.....	129
Literatur.....	129
<b>5 Modellgetriebene Software-Entwicklung mit dem Java</b>	
<b>Framework Rapidclipse.....</b>	<b>131</b>
5.1 Modellieren der Automotive-Anwendungen mit Eclipse	
Model Framework .....	132
5.2 Modellierung des Motors mit Ecore Model.....	136
5.2.1 Struktur eines EMF-Modells zur Modellierung des Motors .....	136
5.2.2 Grafischer Editor für EMF-Klassenmodelle .....	138
5.2.3 Codegenerierung der EMF-Teile.....	142
Fazit.....	286
Literatur.....	286
<b>Stichwortverzeichnis.....</b>	<b>287</b>



# Projektmanagement mit Scrum

1

## Tools zur Entwicklung von Software

### 1.1 Einleitung

Agiles Engineering in der Softwareentwicklung ist im Trend. Die Industrie setzt sich mit prozessorientierten Methoden wie Scrum auseinander und diese werden erfolgreich verbreitet. Die Automobilbranche ist sicher, ob und wie Softwareentwicklungsprozesse agil gestaltet werden können. Der Fakt ist: viele Änderungsanfragen während des Projektverlaufs werden mithilfe der agilen Methoden genannt „agile transformation processes“ bearbeitet. Deshalb ist die Agilität die Zukunft der Automobilindustrie, die Kosten sind in Ordnung, die Qualität ist gut und die Kunden sind nicht enttäuscht.

Project Owner wollen durch die Nutzung der prozessorientierten Methodik „Scrum“ eine effiziente und effektive Softwareentwicklung ermöglichen. Geschäftsführer, Entwicklungsleiter, Produktmanagement und Vertrieb interessieren sich für ein effizientes Projektmanagement in Form von priorisierten Anforderungen. Die höchste Priorität ist es, den Kunden durch frühe und kontinuierliche Auslieferung der funktionierenden Software zufrieden zu stellen. Wobei funktionierende Software das wichtigste Fortschrittsmaß ist. Weil die Aufgabe des Scrum Teams in der Herstellung des Produktes liegt, sollte das Scrum Team interdisziplinär zusammengesetzt sein, damit möglichst viele Ansichten bei der Lösung der einzelnen Aufgaben berücksichtigt werden.

Das Buch zeigt auf, wie diese Prozesse mit agiler Methodik, basierend auf Scrum-Prozessen, in der Automobilbrancherealisiert werden können. Es gibt einen Überblick über die Nutzung von Scrum-Prozessen zur Softwareentwicklung in diesem Bereich.

Es fokussiert auf die Webanwendungen mit Java in Bezug auf die Anforderungen der Features mithilfe des Scrum-Projektmanagements. Das Buch zeigt anhand eines Projekts für die Automobilbranche, wie das Scrum-Projektmanagement die Anforderungen der Features fertigstellt. Es gibt einen Überblick über das Java Framework RapidClipse in Bezug auf Web-Oberflächen, Datenmodell, MySQL mit Xampp, Datenbankanbindung,

Data-binding, ER-Diagramm, Administrations-Frontend, Testumgebung, Deployment, Java Webanwendung, Datenbankzugriffe, Drag-und-Drop, Javabeau-Schnittstelle, Formulare, Master-Detail-Generatoren, Automatisierte Testumgebung und Webanwendung mit Tomcat und IBM WebSphere Liberty und Ecore-Modellierung.

Das Ziel dieses Buches besteht darin, seinen Lesern möglichst verständlich die Fortschritte der agilen Softwareentwicklung näher zu bringen. Es soll den Lesern die Rollen, den Prozess und die Praktiken der Scrum-Methodik bezüglich der Softwareentwicklung mit der Programmiersprache Java, der Datenbank MySQL und den Servern Tomcat und IBM WebSphere Liberty anhand eines konkreten Projektes vermitteln.

### **1.1.1 Anforderungsmanagement**

Das Anforderungsmanagement bei der Scrum-Methodik stellt die Essenz für das Projekt dar. Hierbei werden Darstellungen des Variantenmanagements zum Erfassen der Anforderungen mithilfe der Programmiersprache Java ermöglicht. Das Open Source ReqIF auf Basis von dem Framework ProR ermöglicht das Erstellen der Anforderungen. Der Editor „Obeo Network-Requirement Tooling“ des Frameworks SysML-Designer ermöglicht das Erstellen von Modellen zum Anforderungsmanagement. Anforderungen genannt „Requirements“ vom diesem Editor bestehen aus einer Repository, der über eine Category verfügt. Anforderungen werden mit REQ gekennzeichnet. Der Editor zeigt Eigenschaften und Werte von Anforderungen REQ, welche mithilfe der Aktionen über „View Linked Requirement“ erstellt werden. Mit dem Assistenten von Obeo Network-Requirement Tooling werden neue Anforderungen erstellt und verbunden. Das Tooling ermöglicht sowohl das Editieren der Properties eines Requirement-Objekts als auch das Editieren der Eigenschaften einer-Objekt-Anforderung. Eine Category wird zum Editieren einer Anforderung selektioniert, wobei die editierte Anforderung auch entfernt werden kann. Der Editor gibt einen Überblick über Beziehungen zwischen Elementmodellen und Anforderungen. Die letzteren werden mithilfe der Requirement Traceability Matrix nachverfolgt. Hierbei wird die Nachverfolgbarkeit für das Anforderungsmanagement mithilfe von Menu->New Representation des Editors erstellt. Die Nachverfolgbarkeit gibt einen Überblick über die Anforderungselemente der Matrix. Die Use Case Traceability Matrix ermöglicht das Visualisieren der Anforderungen sowohl in Spalten als auch in Zeilen. Hierbei sind die Beziehungen der Nachverfolgbarkeit zwischen den Anforderungen mithilfe von der Use Case Traceability Matrix zu visualisieren.

### **1.1.2 Oberflächenentwicklung für das Anforderungsmanagement**

Das Java Framework XDEV mit Version 3-5 ermöglicht das Erstellen von grafischen Oberflächen auf Basis von Swing. Die Entwicklung der Anwendungen wird nach dem

Prinzip des Rapid Application Development(RAD) entsprechend der agilen Methodik realisiert. Anwendungen auf Basis von grafischen Oberflächen ermöglichen eine Usability-Analyse durch den Product Owner. Java-Anwendungen auf Basis von Graphical User Interface(GUI) sind leistungsfähiger, schneller, ausgereifter und stabiler. Das Erstellen der grafischen Oberflächen mithilfe des Frameworks Java XDEV4 ist einfach. Nach dem Starten des Java Frameworks XDEV4 wird über die Menüauswahl „Datei“ ein neues Projekt und anschließend verschiedene Fenster angelegt. Das Erstellen eines Projekts ermöglicht das Designen eines Formular-Prototyps mithilfe von Formularfeldern aus Swing wie z. B. XdevLabel, XdevComboBox oder XdevTextField. Formularprototype ermöglichen den Entwicklern die Erstellung von Datenmodellen mithilfe einer relationalen Datenbank wie z. B. MySQL zum Verwenden von Swing-Komponenten. Hierbei wird eine Haupttabelle mit verschiedenen Datenfeldern wie z. B. Id erstellt. Zur Füllung einer Combobox benötigt man eine Datenbanktabelle. Mithilfe der Entwicklung des Datenmodells werden Kapselungen der Daten der Tabellen realisiert. Mit Open Source Xampp wird die relationale Datenbank MySQL gestaltet. Anschließend wird sie an das Projekt angebunden. Dies erfolgt im Menü Datei → Neu → Datenquelle. In einem Dialog wird der Name der Datenquelle festgelegt. Danach wird die MySQL-Schnittstelle im Ordner Datenquellen und Plugin ausgewählt. Data Binding mit Java XDEV 3-5 ermöglicht das Ziehen der erstellten verschiedenen Datenbanktabellen auf Basis von MySQL per Drag-und-Drop auf den Ordner „Virtuelle Tabellen“ im Projektmanagement. Virtuelle Tabellen bilden in den Anwendungen Gegenstücke der erstellten Datenbanktabellen und werden mit GUI-Komponenten verknüpft. Das Erstellen der ER-Diagramme ermöglicht das Visualisieren der Beziehungen zwischen verschiedenen virtuellen Tabellen. Das Erstellen von Administrations-Frontend stellt die Anwendungen der Oberflächenentwicklung in der Technik von RAD dar. Hierbei gibt die GUI-Entwicklung einen Überblick über die Anwendungen der Layout-Manager in der Konstruktion von Formularen, wobei Formulare mithilfe der virtuellen Tabellen erstellt werden. Dies ermöglicht das Verknüpfen von GUI-Komponenten wie z. B. XdevTable mit dem Formular. Fensteraufrufe werden über GUI-Komponenten wie z. B. Buttons mithilfe des Ereignisses *actionPerformed* realisiert. Dies erzeugt eine Registrierung eines Mausklick-Events mithilfe des Aufrufes der Methode `OpenWindow()` mit der Anweisung `XDEV.OpenWindow()`.

### 1.1.3 Design Pattern Delegate mit Java

Eine Software-Architektur zeigt die Strukturen der Funktionalitäten und deren Zusammenhänge, damit eine Wartung des Systems gewährleistet werden kann. Die Software soll später mittels Design Pattern repariert oder getestet werden können. Ziel der Entwicklung der Funktionalitäten mit Design Pattern ist es, den Entwicklern die Programmieraufgaben zu erleichtern. Hierbei wird das Zusammenspiel von Klassen, Interface, Objekten und Methoden dargestellt. Dies ermöglicht die Erweiterung von gewünschten Funktionalitäten. Es gibt verschiedene Arten von Design Pattern zum Programmieren bestimmter An-

wendungen, z. B. Model View Controller, Strategy, Proxy, Interface, Brücke, Broucker, Delegate, Singleton, Iterator, usw... Das Delegate Pattern ist eines der wichtigsten Entwurfsmuster zum Nutzen von Diensten von anderen implementierten Klassen (Hilfsklassen). Dies bedeutet, die eigene Klasse braucht keine Implementierungen zum Realisieren, sondern nutzt die implementierten Methoden anderer Klassen, die als Hilfsklassen dargestellt werden. Anwendungen auf Basis von Java Design-Pattern Delegate verfügt auch über ein Interface, welche die Abstraktion des Programm-Codes darstellen. Ein Interface trennt die Beschreibung von Eigenschaften einer Klasse von ihrer Implementierung. Dabei wird ein Interface mit Hilfe von Klassen in Bezug auf das Thema Design-Pattern für die Energieversorgung implementiert. Zum einen soll die Delegate-Klasse die Dienste von nicht abgeleiteten Objekten verwenden und zum anderen sollen Interface, Basisklassen der Datenstruktur und implementierende Operationen zur Verarbeitung von Algorithmen implementiert werden.

Hier wird das Delegate-Pattern anhand der Verwendung von Interfaces und abstrakten Klassen in Bezug auf die Energieversorgung entwickelt.

### 1.1.4 Webentwicklung mit dem Open Source Rapidclipse

Rapidclipse ist ein neues Open Source Java Framework auf Basis von Java Eclipse. Rapidclipse (Version 3-4) und stellt die Eclipse Distribution für Rapid Cross-Platform-Development mit der Programmiersprache Java dar. Hierbei besteht das Java Framework Rapidclipse aus Eclipse-Plugins, die sowohl vorinstalliert als auch vorkonfiguriert sind. Rapidclipse ist auch ein sogenanntes Klone-Framework, weil es über zusätzliche Tools für Eclipse verfügt wie z. B. Vaadin GUI-builder für HTML5 Frontends, Hibernate, JPA-SQL, Open Source Framework und Cross-Platform Deployment. Letzteres ermöglicht das Deployment von Projekten als Webanwendungen, hybride mobile Apps oder klassische Java Desktop Applikation für Windows, Linux und Mac. Hierbei unterstützen verschiedene Webbrowser das Deployment des Projekts wie z. B. Firefox, Opera, Safari, Internet Explorer oder Microsoft Edge. Linux, Windows und Mac unterstützen die Projektentwicklung für Desktop-Anwendungen. Android, iOS und Windows sind für die mobile Anwendungen dabei. Vaadin ermöglicht den Entwicklern die Entwicklungen von HTML5-Oberflächen. Wobei das GUI-Framework Vaadin auf Basis von Google Web-Toolkit (GWT) entwickelt wurde. Das Graphical User Interface (GUI)-Framework Vaadin stellt ein komponentenbasiertes Model View Controller (MVC)-Framework wie JavaServer Faces (JSF) von Java Enterprise Edition (JavaEE) zum Entwickeln von Oberflächen-Anwendungen mit HTML auf Basis von Java dar. Hierbei besteht ein User Interface-Widgetset aus GWT. Beim Entwickeln der GUI-Anwendungen mit Vaadin wird die Generierung des Clients dynamisch erstellt. Es gibt eine vollautomatisierte Verbindung zwischen Client und Server. Der letzte verfügt über eine optimale Sicherheit. Rapidclipse GUI Builder für Vaadin ermöglicht zum einen das Designen von HTML-Oberflächen mithilfe der

Drag&Drop Technik und zum anderen das Verknüpfen der Datenbank. Hierbei wird der gesamte Code für die Oberflächen automatisch generiert. Das komponentenbasierte MVC-Framework Vaadin verfügt über zusätzliche Erweiterungen u. a. die Databinding-Verbindung von UI und JPA Entities per Drag&Drop, die Generierung von Formularen, die Authentifizierung/Autorisierung oder die GUI-Persistierung. Hibernate-Tools ermöglichen die Entwicklung der Datenbanken mit JPA. Mithilfe von Hibernate werden sowohl Datenbank-Import als auch Generierung von Entity-Klassen realisiert. Das RapidClipse Framework basiert auf Java Standard Edition(Java SE).

### **1.1.5 Webentwicklung mit JavaServer Faces(JSF)**

JavaServer Faces (JSF) ist ein komponentenbasiertes Model View Controller(MVC)-Framework von Java Enterprise Edition(Java EE) zum Entwickeln von Webframework im Java-Umfeld. Hierbei liegen sowohl die Model-Schicht als auch die Controller-Schicht auf der Server-Seite. Außerdem stellt der Browser die View-Schicht dar. Mithilfe vom Server verfügt die View-Schicht über das erzeugte HTML. Ziel der Verwendung von JSF als Webframework ist es, die Erstellung der webbasierten Benutzeroberflächen mit wiederverwend- und konfigurierbaren Komponenten wie z. B. Schaltflächen oder Auswahllisten zu ermöglichen. Hierbei hat das Framework JSF seine Aufgaben in Model, Controller und View verteilt. Das heißt erstens werden aus dem Model die beiden Elemente Entities und Business verwaltet, zweitens werden aus dem Controller die drei Elemente Dispatch, Request und Response erstellt und drittens werden aus der View die Elemente HTML und JavaScript erstellt und Komponenten verwaltet. Von daher ist zu erkennen, dass Komponenten beim Webframework JSF Konstrukte aus dem Server darstellen. Ein wichtiger Aspekt ist die Darstellung der Abstraktionsschicht zum Umsetzen von Webanwendungen durch das serverseitige Programmiermodell von JSF. Der Einsatz von JSF als Webframework ermöglicht den Programmierern, sich nicht um die Verbindung zwischen dem Client und dem Server kümmern zu müssen. Deshalb ist das Erlernen der Webtechnologie für die Programmierer nicht mehr erforderlich. Außerdem sind Formularanwendungen mit dem Framework JSF einfach zu erstellen. JSF verfügt über verschiedene Komponentenbibliotheken wie z. B. PrimeFaces zum Entwickeln von UI-Logik mit Interaktionen.

### **1.1.6 Ecore Modellierung**

Ecore-Modellierung stellt die Umwandlungen des Models eines Klassendiagrammes in Java-Code und damit das Konzept „Modeling4Programming“ dar. Hierbei ermöglichen Modelle die Generierung von Java-Code. Außerdem basiert Ecore auf Objektorientierung. EcoreTools verfügen über Klassen, Attribute, Operationen, Referenzen oder Attribute. Einerseits basieren die Datentypen auf denen von Java. Andererseits werden binäre Asso-

ziationen durch zwei Referenzen dargestellt. Klassen von Ecore-Modellen verfügen über Attribute, Operationen und Referenzen. EClassifier stellt die Oberklasse von EClass und EDataType dar.

---

## **Fazit**

Das Anforderungsmanagement bei der Scrum-Methodik stellt die Essenz für das Projekt dar. Hierbei werden Darstellungen des Variantenmanagements zum Erfassen der Anforderungen mithilfe der Programmiersprache Java ermöglicht. Das Java Framework XDEV mit Version 3-5 ermöglicht das Erstellen von grafischen Oberflächen auf Basis von Swing. Die Entwicklung der Anwendungen wird nach dem Prinzip des Rapid Application Development(RAD) entsprechend der agilen Methode realisiert. RapidCclipse(Version 3-4) stellt die Eclipse Distribution für Rapid Cross-Platform-Development mit der Programmiersprache Java dar. Hierbei besteht das Java Framework RapiCclipse aus Eclipse-Plugins, die sowohl vorinstalliert als auch vorkonfiguriert sind. RapidCclipse ist auch ein sogenanntes Klone-Framework, weil es über zusätzliche Tools für Eclipse verfügt wie z. B. Vaadin GUI-builder für HTML5 Frontends, Hibernate, JPA-SQL, Open Source Framework und Cross-Platform Deployment. JavaServer Faces (JSF) ist ein komponentenbasiertes Model View Controller(MVC)-Framework von Java Enterprise Edition(Java EE) zum Entwickeln von Webframework im Java-Umfeld. Hierbei liegen sowohl die Model-Schicht als auch die Controller-Schicht auf der Server-Seite. Außerdem stellt der Browser die View-Schicht dar. Mithilfe vom Server verfügt die View-Schicht über das erzeugte HTML. Ziel der Verwendung von JSF als Webframework ist es, die Erstellung der webbasierten Benutzeroberflächen mit wiederverwend- und konfigurierbaren Komponenten wie z. B. Schaltflächen oder Auswahllisten zu ermöglichen. Ecore -Modellierung stellt die Umwandlungen des Models eines Klassendiagrammes in Java-Code und damit das Konzept „Modeling4Programming“ dar.



# Entwicklung von Automobilbranche- Anwendungen mit Java

# 2

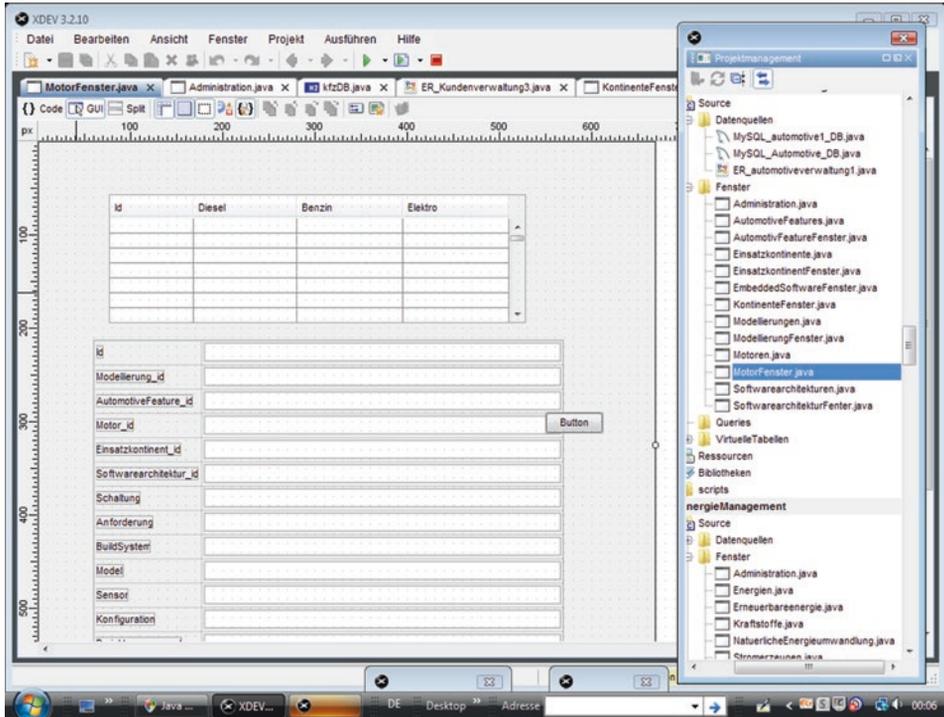
IT-Lösungen in Bezug auf das Anforderungsmanagement von Fahrzeugen und anderen Energieverbrauchern werden zunehmend nachgefragt. IT-Lösungen auf Basis von Java für Feature-Modellierung ermöglichen die Umsetzung der Anforderungen. Das Variantenmanagement der Softwareentwicklung befasst sich mit der Erfassung der Features des Produkts, der Systematik, der Konfiguration und der Entwicklungsartefakte wie z. B. Anforderungen, Modelle, Software, Hardware und Testfälle. Ziel des Anforderungsmanagements ist die systematische Erfassung der zu programmierenden Funktionen um das Management und die Erzeugung des Produktes zu ermöglichen .

Das Kapitel fokussiert auf die Schnittstelle von IT und Automobilbranche in Bezug auf das Anforderungsmanagement. Das Kapitel zeigt, wie Anwendungen auf Basis vom Framework Java RapidClipse entwickelt wurden, welche das Erfassen und Verwalten der Anforderungen bei Fahrzeugen ermöglichen.

## 2.1 Anforderungsmanagement für die Automobilebranche

Der zunehmende Anteil von Software im Fahrzeug stellt neue Herausforderungen an das Anforderungsmanagement in der Fahrzeugindustrie: Dokumentation und Management unterschiedlicher Anforderungsartefakte auf unterschiedlichen Abstraktionsebenen, sowie die Wiederverwendung von Anforderungen unter Berücksichtigung gegebener Abhängigkeiten [1].

Beim Dokumentieren der Anforderungen setzt sich der Programmierer mit der Variantenkomplexität auseinander: Heutzutage werden Systeme als Teil eines Pakets mit verschiedenen Ausprägungen entwickelt. Die Automobilbranchestellt das Management der Produktlinien zum Erfassen der Anforderungen in Bezug auf die Konfigurierbarkeit der Fahrzeuge dar (siehe Abb. 2.1). Abb. 2.1 stellt das Variantenmanagement zum Erfassen der Anforderungen dar. Die Abb. 2.1 zeigt die relevanten Merkmale der Features für die



**Abb. 2.1** Darstellung des Variantenmanagements zum Erfassen der Anforderungen mit Java

Fahrzeugherstellung. Hierbei wird ein Fahrzeug entweder mit Benzin- oder Dieselmotor mithilfe des Features „Motor“ hergestellt. Die Entwickler programmieren das Feature. Anschließend werden Variantengetriebe für das Feature „Schaltung“ erfasst. Das Feature „Einsatzkontinent“ wird in Kombination mit den beiden Features „Motor“ und „Schaltung“ realisiert.

Das Anforderungsmanagement in der agilen Methodik wie z. B. in Scrum fokussiert auf die Erstellung von *User Stories* bezüglich der Vorstellungen und Wünschen des neuen Produktes. Diese Anforderungen sind wichtig, weil sie die Basis für die Projektplanung bilden, welche „Product Backlog“, „Sprint Backlog“ und „Tasks“ darstellen.

Ein wichtiger Aspekt im Anforderungsmanagement ist die Unterstützung einer Methodik. Scrum, eine der bekannten agilen Methoden wird mit dem Anforderungsmanagement zum Realisieren des Projektmanagements kombiniert.

### 2.1.1 User Story zum Anforderungsmanagement

Die User-Stories sind kurze Features und implementieren die Elemente des „Product Backlog“. Diese beschriebenen Features stellen eine Zusammenarbeit zwischen dem

Kunden (oder dem Product Owner) und dem Entwicklungsteam dar. Der Product Owner im Auftrag des Kunden erstellt die User-Stories, welche als Product Backlog dem Entwicklerteam vorgestellt werden.

Der Product Owner ist der Vermittler zwischen dem Kunden und dem Entwicklerteam und damit für die Applikationen des Anforderungsmanagements verantwortlich.

Die Anwendung der User-Stories in der Software-Entwicklung ermöglicht die Analyse der Anforderungen, deren Ermittlungen mithilfe dieser Methode vereinfacht werden.

In der Praxis werden User-Stories auf einer Karteikarte oder einem Post-IT strukturiert in der Form

„Als (*Anwender einer konkreten Rolle*) möchte ich (*tun*)“. Das Listing 2.1 zeigt den Aufbau der User Storys.

---

**Listing 2.1: Aufbau der User Storys**

- „Als Anwender des Motors möchte ich die Anforderungen für das Motorkonzept erfüllen“
- „Als Anwender des Motors möchte ich das Design für das Motorkonzept gestalten“
- „Als Anwender des Motors möchte ich die Funktionalitäten mit den entsprechenden Tools verbinden“
- „Als Anwender des Motors möchte ich die Fenster der Motorsysteme öffnen“
- „Als Anwender des Motors möchte ich die drei Phasen des Scrum-Prozessmodells simulieren“
- „Als Anwender des Motors möchte ich die Verbrennungsdiagnostik für das Motorkonzept realisieren“

## 2.1.2 Storyboard

Das Ziel der Nutzung des Storyboards ist es, die Reihenfolge der User-Storys zu ermöglichen.

Hierbei treffen Product Owner und Scrum-Team eine Entscheidung zur Bearbeitung der Funktionalitäten. Zuerst skizziert der Product Owner die Anforderungen der Software auf Kärtchen, die „Story Cards“ genannt werden. Anschließend werden die Anforderungen priorisiert, wobei das Scrum-Team den Aufwand der Implementierung dieser Anforderungen und deren Relevanz schätzt. Die Entscheidung zum Programmieren dieser Anforderungen werden mithilfe dieser Schätzungen und Prioritäten getroffen. Falls es Veränderungen während der Entwicklungsarbeit gibt, wird das Storyboard erneut ausgeführt. Das Listing 2.2 zeigt den Aufbau eines Storyboards.

---

**Listing 2.2: Aufbau eines Storyboards**

0. As a developer I want to register
1. As a developer I want to sign-on
2. As a developer I want to analyse the program
3. As a developer I want to search the tools

### 2.1.3 Product Backlog

Das Product Backlog stellt die Verwaltung der Priorisierung der Anforderungen mithilfe einer Tabelle dar. Diese Tabelle beschreibt sowohl Funktionalitäten implementierter Endprodukte als auch die Dokumentationsaufforderungen. Das Product Backlog enthält sowohl Elemente des Storyboards als auch Wünsche der Entwickler und technologische Aspekte.

Laut der Standardisierung der Scrum-Methodik sollte die englische Sprache als Standard zum Product Backlog eingesetzt werden, damit die Anforderungen einfach formuliert werden können.

Ansonsten kann die deutsche Sprache für die Strukturierung des Product Backlogs auch als Standard eingesetzt werden. Hierbei werden die Informationen mithilfe der technischen Begriffe zum Erstellen eines Projekts vereinfacht. Wichtig ist es, alle Einträge mit einer Priorität und einem Status zu versehen.

Die Tab. 2.1 zeigt die Struktur eines Product Backlogs zum Beschreiben der priorisierten Anforderungen. Der Product Owner erstellt diese ToDo-Listen zu Teilanforderungen mit bestimmten Prioritäten. In der Struktur der Tabelle ist zu bemerken, dass die Priorisierungen mithilfe der Identifikationsnummer (ID) gekennzeichnet wurden. Die Spalte mit dem Titel „Module“ wurde „SUBJECT“ genannt, weil ähnliche inhaltliche Priorisierungen zusammengruppiert werden. Die Spalte mit Titel „Item“ stellt die Beschreibungen der Priorisierungen mithilfe der User-Stories dar. Wobei die Spalte mit dem Titel „Requirement“ die Struktur dieser Beschreibungen zeigt. Die Spalte „Priority“ zeigt die Reihenfolge der Implementierungen der Teilanforderungen bezüglich der Umsetzungen eines Sprints. Die letzte Spalte mit dem Titel „Status“ verdeutlicht die Aufgaben während der Priorisierung der Teilanforderungen. Der Product Owner ist für Änderungen der Priorisierungen verantwortlich, aber letztlich tragen sowohl der Master als auch das Scrum-Team zu dem Inhalt des Product Backlogs bei.

## 2.2 Rolle des Anforderungsmanagements

Laut den agilen Prinzipien ist das Anforderungsmanagement ein Teil der agilen Entwicklung [2]. Dies bedeutet dass, die Anforderungen nicht umfassend dokumentiert werden müssen, sondern nur insoweit ausgeführt werden müssen, dass sich die Projektbeteiligten

**Tab. 2.1** Das Product Backlog

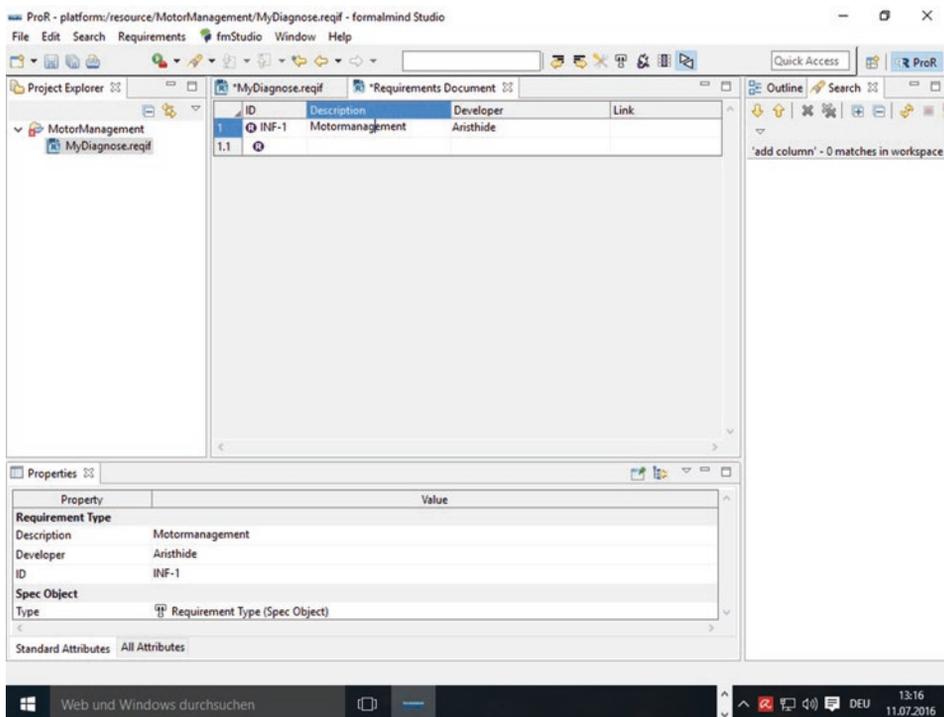
ID	Module	Requirement	Item	Priority	Status
1	Signin	Story	As a developer I want to register	1	ToDo
2	Signin	Story	As a developer I want to signin in	1	ToDo
3	Develop	Story	As a developer I want to analyse the program	2	ToDo
4	Develop	Story	As a developer I want to search the tools	3	ToDo

Aufbau vom Product Backlog: Priorisierung der Teilanforderungen

es verstehen können. Das Werkzeug zum Dokumentieren der Anforderungen basiert auf den Standard ReqIF (Requirements Interchange Format) vom Eclipse Framework. ReqIF stellt die Verwaltung der Anforderungen mit Eclipse dar. Hierbei spielt das Anforderungsmanagement eine essenzielle Rolle in der Systementwicklung. Anforderungen unterstützen die Kommunikation zwischen den Projektbeteiligten. Darüber hinaus ermöglichen die Anforderungen die Messung des Projektfortschrittes.

### 2.2.1 ReqIF als Model für Anforderungen

Das Werkzeug ReqIF ist ein mit XML umgesetztes Dateiformat und besteht aus einer Baumstruktur, die mit dem Tag <REQ-IF> ihre Wurzel hat [3]. Das Open Office von Eclipse *ProR* stellt die Anwendungen der Anforderungen im ReqIF dar. Es gibt auch kommerzielle Produkte für ReqIF wie z. B. ReqIF Server enso Managers und QuakeSafe von Orange-Moon. ReqIF ist ein aus der Automobilbranche stammender Standard zum Austausch von Anforderungen. Es besteht aus einem Datenmodell mit einem XML-basierten Persistenz-Format. ReqIF ist nicht nur ein Format für den Austausch der Anforderungen sondern auch für das Realisieren des Datenmodells geeignet. Abb. 2.2 zeigt das Erstellen der Anforderungen mit dem Werkzeug ReqIF von Eclipse.



**Abb. 2.2** Erstellen von Anforderungen mit Open Source ProR für ReqIF

Der Open Source formalmind Studio ist eine kostenlose Version für das Werkzeug ProR von Eclipse, dessen Programm zur Beschreibungen von Anforderungen geeignet ist. ProR ist eine OpenSource Software zum Arbeiten mit effizienten Anforderungen. Es basiert auf dem Requirements Interchange Format (ReqIF), womit die Interoperabilität mit anderen Werkzeugen gewährleistet wird [3].

## 2.2.2 Das ReqIF-Datenmodell

Das Werkzeug ReqIF orientiert sich an Datenmodell, welches auf das Element ReqIFContent fokussiert. Sieben wichtige Bestandteile eines ReqIF-Modells sind Subelemente vom ReqIFContent: *Datatypesdefinition*, *SpecType*, *SpecObject*, *SpecRelation*, *Specification* und *RelationsGroup*. Die Tab. 2.2 stellt die Funktionalität des ReqIFContent dar.

Gemäß der Tab. 2.2 verwendet ReqIF zur Beschreibung der Anforderungen („requirements“) *specification objects* mit definierbaren *Attributen*. Jedes Attribut eines „specification object“ hat jeweils einen Datentyp, z. B. *Boolean*, *Integer*, *Real*, *String*, *Auswahltypen* mit definierten Werten. Formatierte Texte und eingebettete Bilder werden als Elemente der Anforderungen dargestellt. Optional können numerische Datentypen je nach dem Projekt in ihrem Wertebereich integriert werden. Anschließend werden hierarchische Bäume (*specifications*) verwendet, um gegliederte Sichten auf die Anforderungen zu bilden, wobei Objekte in einem Baum mehrfach oder in mehreren Bäumen vorkommen können. Am Schluss werden Relationen zwischen Objekten dargestellt, z. B. Abhängigkeiten oder Widersprüche. Ebenso enthalten auch Relationen wie Objekte Attribute.

**Tab. 2.2** Wichtige Bestandteile eines ReqIF

Bestandteile	Struktur	Funktion
<i>SpecObject</i>	Enthält Metadata und interne genutzte ID	Anforderungsobjekte
<i>SpecType</i>	Enthält viele Elemente in ReqIF und <i>SpecObject</i> (Attribut <i>Description</i> vom Typ <i>String</i> )	Definition beliebiger Anzahl der Attributen
<i>Datatypesdefinition</i>	Text, Integer, Float, Date, Enumeration und XHTML	Instanziierung der Datentypen z. B.: generischer Typ <i>String</i> für die maximale Textlänge
<i>Specification</i>	Baumstruktur der Elemente: <i>SpecObjects</i> und <i>SpecTypes</i>	Referenzieren von <i>SpecObjects</i> , Typisieren über <i>SpecTypes</i>
<i>SpecRelation</i>	Verbindung	Verlinken der Elemente Typisieren mit <i>SpecTypes</i>
<i>RelationsGroup</i>	Verbindung	Beziehungen zwischen Objekten Attribute als Elemente
<b>Bestandteile</b>	<b>Struktur</b>	<b>Funktion</b>

Komponente vom ReqIFContent

## 2.2.3 Modellieren der Automotive-Anwendungen mit Requirement Toolings

Grafische domänenspezifische Sprachen(DSL) zum Anforderungsmanagement sind dank der Eclipse-Projekte gefragt. Die Nachfragen kamen als die Französische Firma Obeo im Eclipse-Projekte Teile ihres Werkzeuges Obeo Designer veröffentlicht hat. Obeo Designer ist eine starke Umgebung zum Realisieren von sowohl Modellierungen als auch von Anforderungen [4]. Das Tool Obeo Designer ist ein leistungsfähiges Werkzeug für grafische DSL. Die Firmen Obeo und Thales haben das Werkzeug Obeo Designer für die Realisierungen der Modellierungen und Anforderungen entwickelt.

Das von der Firma Obeo Projekt Sirius ist mit Eclipse, RapidClipse und formalmind-Studio kompatibel. Von daher kann man Obeo Designer mithilfe der Suchmaschine Marketplace in diesen Umgebungen installieren.

### 2.2.3.1 Anwendungen der Requirement-Toolings von Obeo Designer in der Umfassung des Anforderungsmanagements

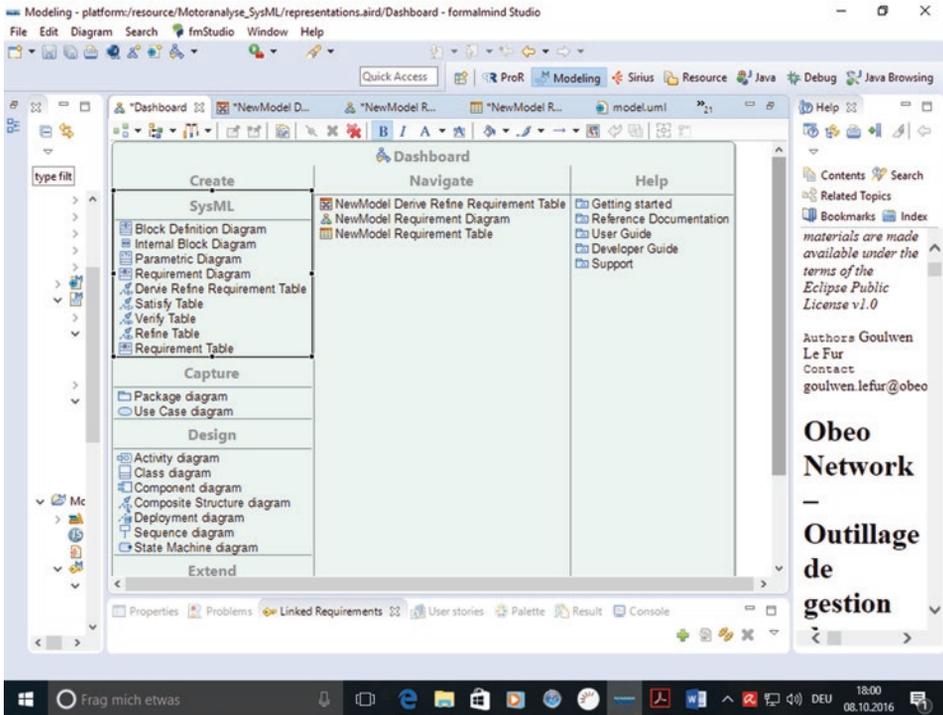
Das Werkzeug Obeo Designer fokussiert sowohl auf die Systemmodellierung(SysML) als auch auf die Software- oder Hardwaremodellierung(UML) zum Erstellen der Anforderungsanalyse, Spezifikation, Systemanalyse, Modellierung und Testfälle. SysML ist die grafische Modellierungssprache für die Systemmodellierung. SysML hilft den Entwicklern bei Analyse, Spezifikation, Design, Verifikation und Validation von Systemen, die aus Hardware, Software, Daten, Personal, Prozessen und technischen Hilfsmitteln (Anlagen) bestehen [4]. Obeo Designer bietet zur Modellierung viele Perspektive wie zum Beispiel Anforderungen, statische und dynamische Architektur und Interaktionen.

Laut der Definition von OMG bietet SysML Designer fünf Komponente:

- Block Definition Diagram(BDD)
- Internal Block Diagram(IBD)
- Parametric Diagram(PAR)
- Requirement Diagram(REQD)
- Requirement Table(REQT)

Das Abb. 2.3 zeigt das Dashboard des SysML Designers vom Obeo Designer. Das Dashboard besteht aus fünf Komponenten:

- Create SysML
- Capture
- Navigate
- Design
- Help

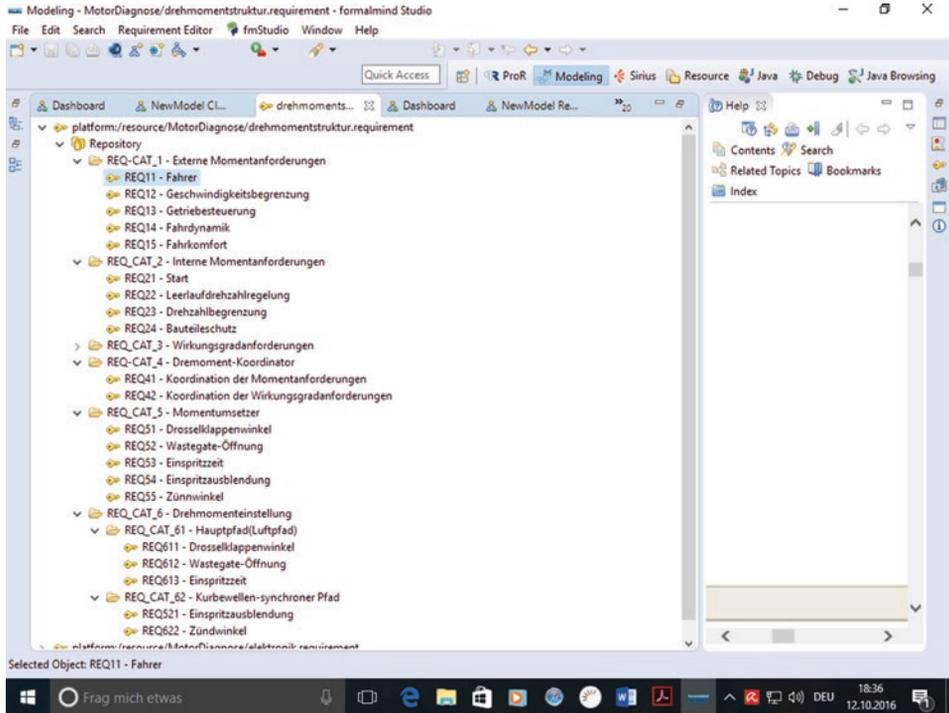


**Abb. 2.3** Überblick über das Dashboard vom SysML Designer

Die Komponente create SysML fokussiert auf Anforderungen und enthält vier Diagramme und eine Tabelle. Die zweite heißt *capture* und besteht aus den Unter-Komponenten *Package diagram* und *Use case diagram*. Die dritte Komponente ist *Design* genannt und stellt die Verhaltensdiagramme dar wie z. B. Sequenz-, Aktivitäts-, Klassen- und Komponentendiagramm. Die vierte ist zum Erstellen von Anforderungsdiagrammen zuständig. Die letzte ist *Help* genannt und besteht aus Dokumenten für Tutorials, Support und Anwendung zum Starten.

Obeo Designer verfügt über Tools zum Anforderungsmanagement: „Requirement Toolings“. Das Konzept ermöglicht den Entwicklern Anforderungen mithilfe eines Assistenten „Requirement Model (Catégorie Obeo Designer For IS)“ zu erstellen. Mithilfe der Umgebung formalmindStudio ist dieser Assistent via das Menü „File > New > Other ... > Requirement Model (Catégorie Obeo Designer For IS)“ erreichbar. Einerseits enthält das Werkzeug „Requirement Toolings“ vom Obeo Designer einen Editor zur Erstellung der Anforderungen und andererseits ein Fenster, das die Verbindung zwischen den Anforderungen des Projekts und die Modelle ermöglicht. Die Abb. 2.4 zeigt die Struktur des „Obeo Network-Requirement Toolings“ mit der Umgebung formalmindStudio.

Anhand der Abb. 2.4 ist zu bemerken, dass das Erstellen eines Anforderungsmodells mit der Erscheinung eines Editors beginnt. Auf der Abb. 2.4 ist zu sehen einen Editor, der

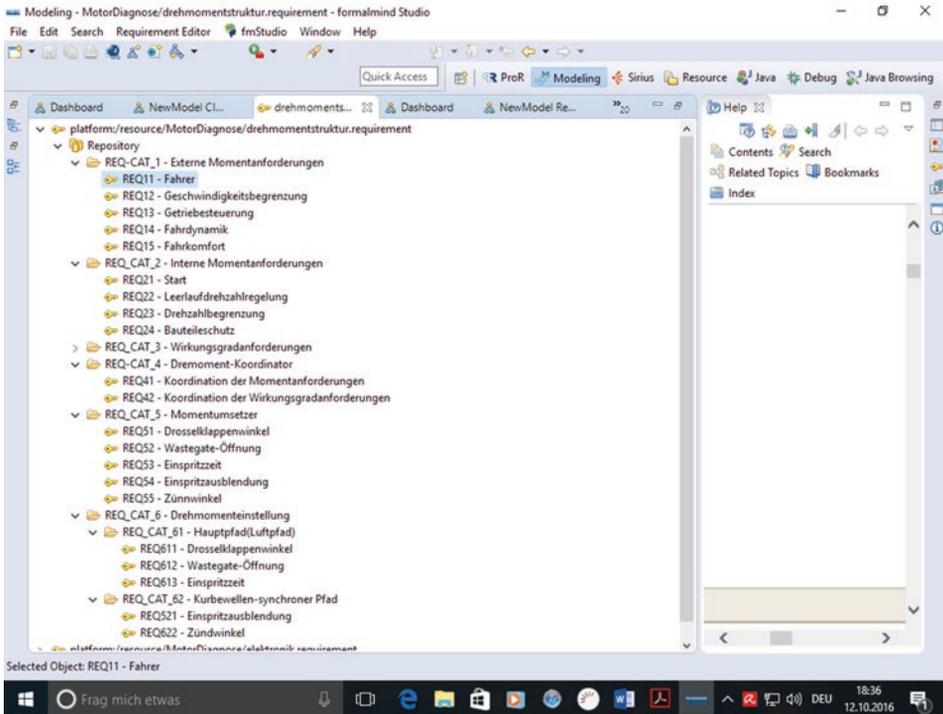


**Abb. 2.4** Erstellen eines Modelles zum Anforderungsmanagement mit Obeco Network-Requirement Tooling

zwei Teile enthält: Auf der linken Seite stellt ein Baum das Anforderungsmodell mit seinen Objekten dar. Auf der rechten Seite des Editors sind die Eigenschaften(Properties) des selektionierten Elementes des Baumes zu sehen. Das Erstellen der neuen Elemente im diesem Anforderungsbaum wird mithilfe des Menus „NewChild“ des Containers realisiert. Hierbei wird das gewünschte Elemente erstellt. Wobei das Erstellen einer neuen Anforderung in einer Kategorie über die Menüauswahl „New Child > Requirement“ erfolgt.

Das neue Element erscheint in diesem Anforderungsbaum und nachdem es selektioniert wurde, können seine Eigenschaften (Properties) auf der linken Seite erstellt werden.

Die Abb. 2.5 gibt einen Überblick über die Struktur des Anforderungsbaumes mit dem Editor „Obeco Network-Requirement Tooling“. Das Beispiel zeigt einen typischen Baum mit Elementen der Anforderungen bezüglich der Drehmomentstruktur des Motormanagements im Bereich Automobilelektronik. Anhand der Abb. 2.5 ist die Funktionsstruktur der Motorsteuerungen bezüglich der drehmomentbasierten Struktur erläutert. Die Abb. 2.5 zeigt, dass sowohl Drehmoment als auch Wirkungsgrad eine physikalische Drehmomentanforderung definieren. Beispielerweise interpretiert die Anforderung REQ11 bezüglich des Fahrers den Winkel des Fahrpedals als Drehmomentanforderung. Abb. 2.4 und 2.5 stellen die Hauptaufgaben der Momentstruktur dar, wobei Radmoment,



**Abb. 2.5** Überblick über die Struktur des Anforderungsbaumes des Editors „Obeo Network-Requirement Tooling“

Getriebeausgangsmoment, Kupplungsmoment und Kurbelwellenmoment (oder inneres Moment) im Antrieb liegen.

### 2.2.3.2 Bereiche der Anforderungsmodelle

Requirement Tooling von Obeo Designer stellt drei Bereiche zur Anforderungsmodellierung: Repository, Categories und Requirements.

Der erste Bereich ist Repository genannt und ist das Wurzelement des Models. Ein Repository hat einen Namen und enthält zentrale *Categories*. Außerdem kann eine Repository ein Modellelement referenzieren. Abb. 2.6 zeigt den Aufbau eine Repository für die Drehmomentstruktur des Motormanagements. Hierbei ist fünf Categories z. B. REQ\_CAT\_1 bis REQ\_CAT\_6 in einer Baumstruktur zu sehen.

Der zweite Bereich des Requirement Toolings heißt *Categorie* und ist ein Container für die Anforderungen. Dieser Bereich verschreibt die Anforderungen je nach verschiedenen Kriterien. Eine Category verfügt über eine Schlüsselnummer und einen Namen. Außerdem kann eine Category eine Unter-Category enthalten und auch ein Modellelement refe-

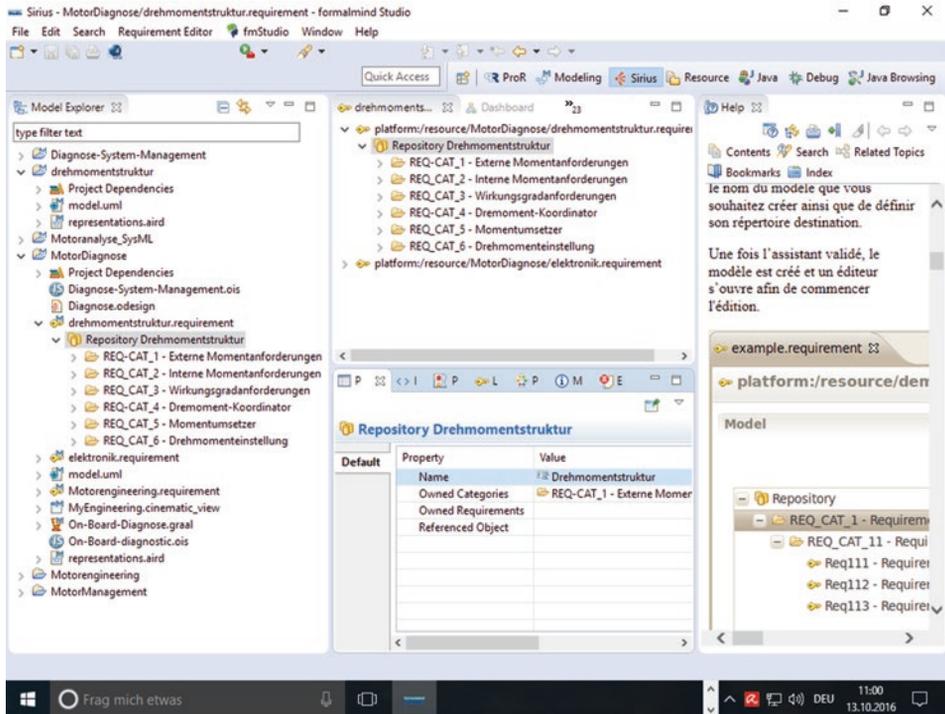
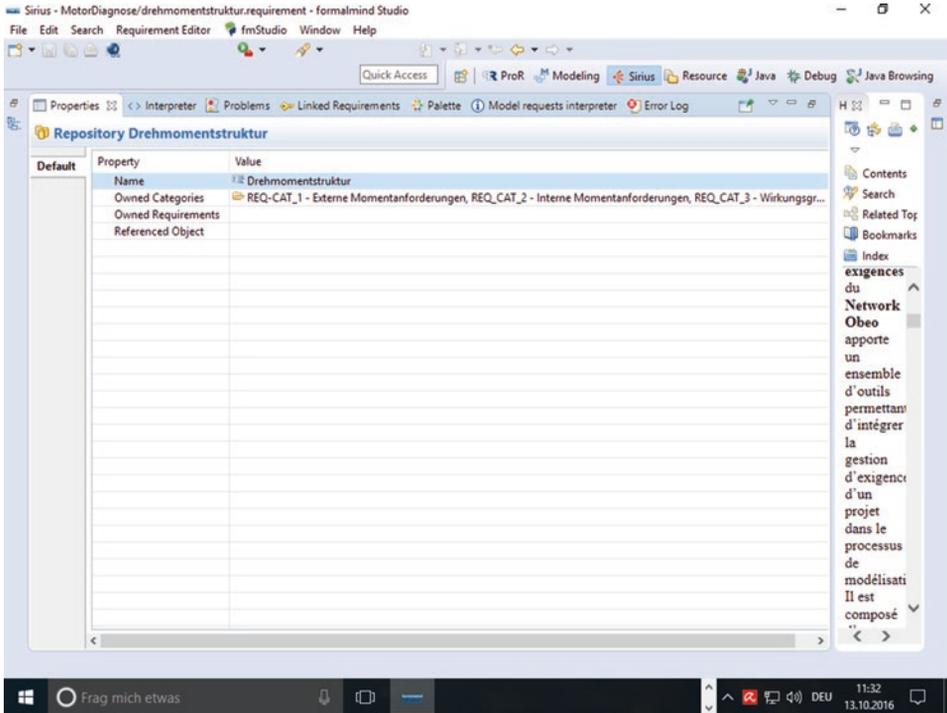


Abb. 2.6 Aufbau einer Repository

renzieren. Die Abb. 2.7 stellt den Aufbau einer Category bezüglich der Schlüsselnummer und des Namens. Hierbei ist zu erkennen, dass die Schlüsselnummer die Begriffe REQ und CAT bzw. für Requirement und Category wie z. B. REQ\_CAT\_1 bis REQ\_CAT\_6. Die Ziffer 1 bis 6 stehen nach CAT.

*Requirement* ist der dritte Bereich des Obeo Network-Requirement Toolings zum Modellieren der Anforderungen. Dieser Bereich ist je nach funktionalen oder technischen Kriterien eingeordnet. *Requirement* kann ein Unter-Typ enthalten. Hierbei verfügt ein *Requirement* über eine Schlüsselnummer, einen Namen, einen Status und eine Beschreibung (*statement*) hin. Es ist möglich, den Ursprung (*Rational*) eines Requirements sowie seine Akzeptanzkriterien (*AcceptanceCriteria*) zu definieren. Letztendlich ist es möglich dem Requirement ein Modellelement zu referenzieren. Abb. 2.8 und 2.9 geben einen Überblick über die Struktur des Bereichs *Requirement* bezüglich der Modellierung der Anforderungen. Das Beispiel zeigt das Requirement REQ11 bezüglich des Verhaltens der Anforderung zum *Fahrer* in der Drehmomentstruktur für das Motormanagement. Die Abb. 2.8 zeigt die Schlüsselnummer, den Namen und die Category der Anforderung



**Abb. 2.7** Aufbau einer Category

REQ11. Die Abb. 2.9 detailliert die Struktur der Anforderung in Bezug auf die Eigenschaft(*Properties*) und den Wert(*Value*).

### 2.2.3.3 Das Editieren der Anforderungen innerhalb des Obeo Designers

Anforderungsmanagement mit Obeo Designer ermöglicht das Verbinden der Anforderungen mit den Modellelementen mithilfe der View *Linked Requirement*. Der Zugang zu dieser View ist mithilfe der Menüauswahl folgendend dargestellt:

Window > Show View > Other ... > Linked Requirement (Catégorie Obeo Designer For IS)

Dieser View ist mit dem aktiven Editor im Obeo Designer synchronisiert. Wenn ein Modellelement selektioniert wird, zeigt die View alle Anforderungen zu diesem Modellelement. Von daher wird diese View automatisch aktualisiert. Ein Merkmal dieser View ist ein gelber Schlüssel auf der oberen linken Seite dieser View vor dem Begriff „*Linked Requirement*“. Auf der oberen rechten Seite der View sind ein Plus-Zeichen, einen Kugelschreiber mit einem Blatt, zwei Büroklammer und ein Multiplikation-Zeichen zu sehen. Diese Symbole stellen die Aktionen zum schnellen Bearbeiten der Anforderun-

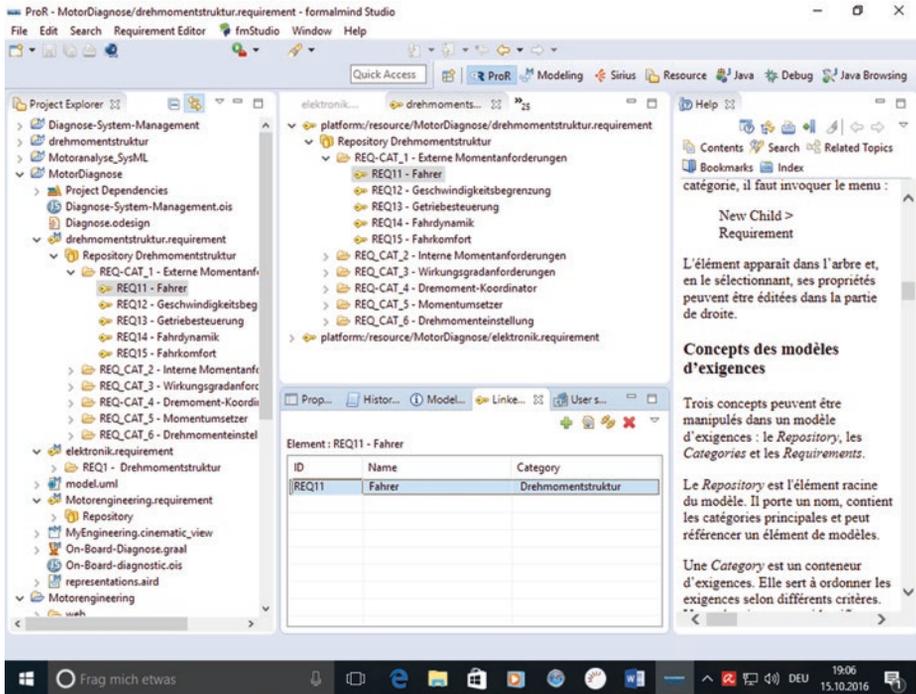


Abb. 2.8 Überblick über die Struktur einer Anforderung am Beispiel vom REQ11

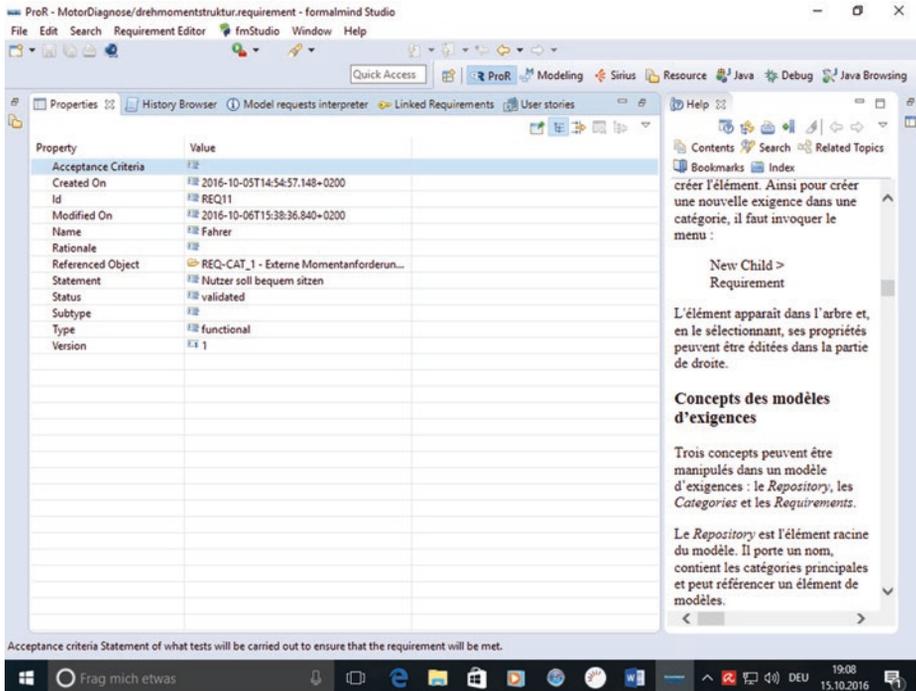
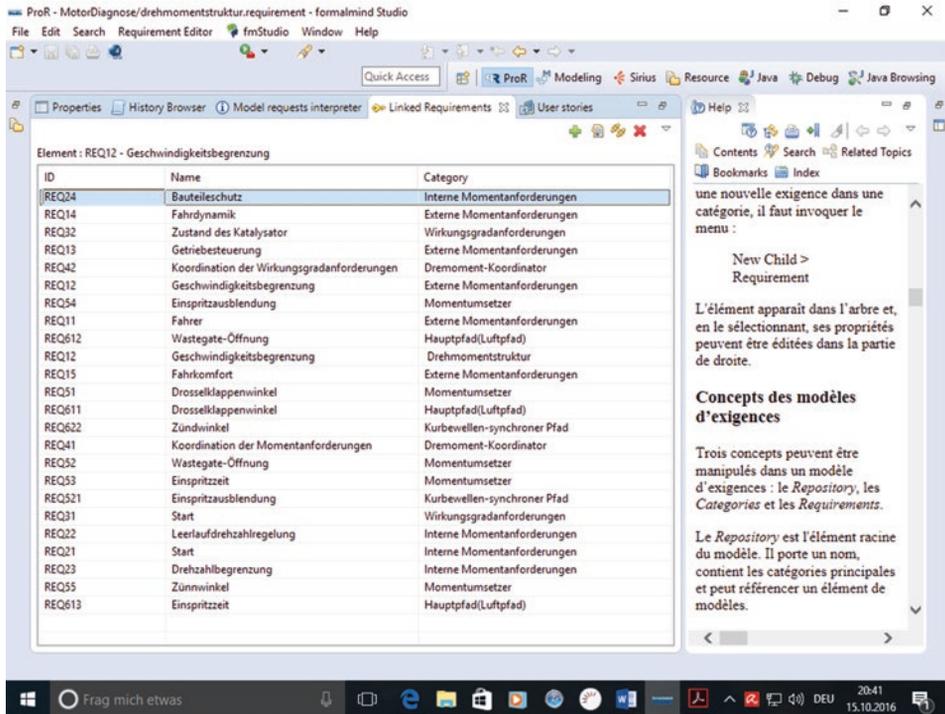


Abb. 2.9 Eigenschafthen und Werte der Anforderung REQ11



**Abb. 2.10** View Linked Requirement zur Erstellung der Anforderungen mithilfe der Aktionen

gen dar. Einerseits zeigt Abb. 2.10 die View *Linked Requirement* mit den Aktionen zur Erstellung der Anforderungen und andererseits die Schlüsselnummer, die Namen und die Categories der Anforderungen.

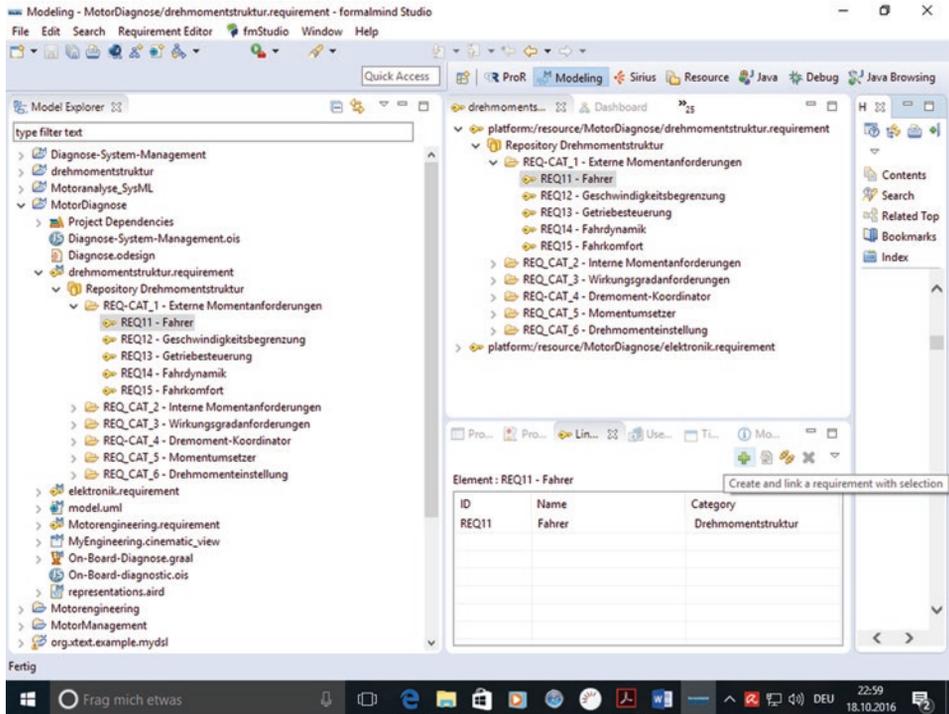
### Erstellen von Anforderungen

Das grüne Plus-Zeichen ermöglicht sowohl eine Erstellung als auch eine direkte Verbindung einer neuen Anforderung sobald eine bisherige Anforderung selektioniert wird. Ein Assistent wird geöffnet und zeigt die Funktion dieser Aktion mit der folgenden Anweisung:

„*Create and link a requirement with selection*“

Die Abb. 2.11 zeigt das Erstellen einer neuen Anforderung mithilfe des Assistenten.

Die Anweisung „*Create and link a requirement with selection*“ wird mithilfe des Klickens auf den Button für das medizinische Kreuz gezeigt. Sobald dieser Button geklickt wird, öffnet sich ein Assistent zur Erstellung neuer Anforderung, wobei die Eigenschaften dieser neuen Anforderung auf der ersten Seite mit Linked Requirement sichtbar sind. Die zweite Seite ermöglicht die Auswahl der Category der neuen Anforderung. Falls diese nicht existiert, kann man mithilfe des Buttons *New Category* eine neue Anforderung er-



**Abb. 2.11** Erstellen und Verbinden einer neuen Anforderung mit dem Assistenten von Obeo Network-Requirement Tooling

stellen. Die Abb. 2.12 und 2.13 zeigen das Erstellen der Eigenschaften eines Anforderungsobjektes zum einen und zum anderen das Erstellen einer neuen Category.

### Editieren der Anforderung

Die zweite Aktion stellt ein Blatt und einen Kugelschreiber und ermöglicht das Editieren einer schon anwesenden Anforderung im Assistenten dar. Beim Anklicken dieses Buttons wird eine selektionierte Anforderung im Assistenten mit der Anweisung „Edit the selected requirement“ editiert. In dem Assistenten kann man die Eigenschaften der Anforderungen verändern. Die Abb. 2.14 und 2.15 geben einen Überblick über den Assistenten zum Editieren einer selektionierten Anforderung.

### Verbindung von Anforderungen

Die dritte Aktion ermöglicht die Verbindung von Anforderung mithilfe der zwei Büroklammer-Zeichen, wobei Beim Anklicken auf das Zeichen ein Assistent mit der Anweisung „Link requirement“ mit verschiedenen Repositories erscheint. Der Assistent ermöglicht das Verwalten der Beziehung zwischen Elementmodellen und Anforderungen. Die Abb. 2.16 zeigt den Dialog-Assistent zum Selektionieren von Anforderungen für gewünschte Categories.