

Teil III

**Feldstudie: Fachliche Integration des *EMI*-Systems
in das SSELab**

III.1 Anforderungen zur fachlichen Integration ins SSELab

In der Wissenschaft gebührt der Ruhm demjenigen, der die Welt von einer Idee überzeugen kann, nicht demjenigen, der die Idee zuerst hatte.

SIR FRANCIS DARWIN

Inhalt

III.1.1 Organisationsprofil	97
III.1.2 Anforderungsanalyse	98
III.1.3 Fazit der Anforderungsanalyse	103

Zu Beginn dieser Arbeit stand das *SSELab* als einzige, geeignete Feldstudie zur Verfügung. Die Vorteile dieses Forschungsprojekts sind neben der universitären und fachlichen Nähe zu der Arbeit am Lehr- und Forschungsgebiet für Software-Konstruktion, eine große Anzahl an Nutzer, wie auch der Plan einer *EMI*-Integration innerhalb des SSELabs. Im weiteren Verlauf der Arbeit wurde jedoch festgestellt, dass die Ergebnisse dieser Evaluation nicht ausreichen, um die Informationsbedürfnisse an ein Kennzahlen-System wie *EMI* beziehungsweise dessen Dashboard *SCREEN* zu verallgemeinern. Aus diesem Grund wurden des Weiteren zwei Unternehmen in den Betrachtungsfokus hinzugezogen, dessen Ergebnisse in Teil IV und Teil V präsentiert werden.

III.1.1 Organisationsprofil

Das *SSELab* wurde vom *Lehrstuhl für Software Engineering der RWTH Aachen* unter der Leitung von Prof. Dr. Bernhard Rumpe im Rahmen einer Forschungsarbeit von Christoph Herrmann und anderen wissenschaftlichen Mitarbeitern entwickelt [HKR12]. Es dient als Unterstützungsplattform zur Entwicklung von Software-Projekten und stellt typische Basisdienste, wie beispielsweise Versionsverwaltungssysteme, Change-Request-Systeme, Wiki-Seiten oder Mailinglisten zur Verfügung. Das SSELab wird vor allem innerhalb des universitären Bereichs für diverse Projekte mit unterschiedlichen Schwerpunkten genutzt, findet jedoch auch in Kooperationen mit Industrieprojekten Verwendung. Die unterschiedlichen Möglichkeiten der Verwendung werden im folgenden Abschnitt III.1.2 genauer erläutert.



III.1.2 Anforderungsanalyse

Die Anforderungsanalyse für ein SSELab-Dashboard fand im Rahmen von elf Interviews mit entsprechenden Projektleitern diverser SSELab-Projekte statt. Innerhalb der 30-45 minütigen Interviews wurden die Projekte, ihre Ziele, wie auch aktuelle Maßnahmen zur Ermittlung und Verbesserung der Projektqualität durch einen kurzen Fragenkatalog à fünf Fragen erörtert, welche in der folgenden Auflistung dargestellt werden.

1. Welche Art von Projekten enthält das benutzerspezifische SSELab der einzelnen Interview-Partner? Welche Prozesse und Werkzeuge werden verwendet? Wie sieht die Datenanbindung aus?
2. Welche Maße und Mittel zur Qualitätssicherung werden aktuell verwendet?
3. Welche Wünsche an die Qualität (sowohl von Produkt als auch Prozess) existieren?
4. Werden spezielle Metriken, wie beispielsweise Projektfortschritt, verbrauchte Mitarbeiterstunden, Testabdeckung oder ähnliches gewünscht beziehungsweise benötigt?
5. Existieren weitere Anmerkungen oder Wünsche an das Dashboard?

Der folgende Teil beschäftigt sich mit der Auswertung und Evaluation dieser fünf Fragen.

III.1.2.1 Frage 1: Welche Projekte sind im SSELab vorhanden?

Da das SSELab ein Forschungsprojekt ist, findet auch in dieser Umgebung der größte Gebrauch statt. Die elf befragten Projektleiter sind wissenschaftliche Mitarbeiter diverser Lehrstühle des Fachbereiches Informatik der RWTH Aachen. Jedoch nutzen diese das SSELab nicht allein für typische Software-Entwicklungsprojekte, sondern erweitern den Nutzungsfokus um weitere Aspekte. Die unterschiedlichen Projektarten, ihre Datenanbindungen, die verwendeten Werkzeuge und die unterschiedlichen Benutzerrollen werden folglich erörtert.

III.1.2.1.1 Projektarten

Wie eingangs erwähnt, wird das SSELab nicht ausschließlich für Software-Projekte verwendet. Zwar ist dies die Hauptnutzungsart und die bereitgestellten Basisdienste sind hierauf ausgelegt, jedoch werden zusätzlich Organisationsprojekte, wie auch studentische und wissenschaftliche Arbeiten mit dieser Plattform verwaltet.

Software-Entwicklungsprojekte: Sie decken den Großteil der im SSELab vorhandenen Projekte ab. Typischerweise sind es Java-EE-Projekte, welche die Eclipse-Entwicklungsumgebung und einige der im SSELab zur Verfügung gestellten Dienste nutzen. Sie reichen von Client-Server-Anwendungen über Code-Generatoren bis hin zu einer komponentenbasierten Entwicklung.

Organisationsprojekte: Diese Projektes dienen der Strukturierung und Organisation des Lehrstuhlbetriebs, der (Lehr-)Veranstaltungen oder vorhandener Verwaltungsprojekte, wie beispielsweise der Finanzverwaltung. Ebenso werden mithilfe des SSE Labs interne Dokumente, Mitarbeiter oder eigene Webseiten verwaltet.

Wissenschaftliche Arbeiten: Typische Schreibprojekte wie beispielsweise Veröffentlichungen oder Bücher fallen unter diesen Begriff. Hierfür wird das SSE Lab als Dokumentenablage verwendet, um die Schriftstücke an einem lokalen Ort zu sammeln und gegebenenfalls für mehrere Editoren verfügbar zu machen.

Studentische Arbeiten: Sie umfassen sowohl studentische Abschlussarbeiten, wie auch diverse Software-Praktika. Die Praktika werden in der Regel wie Entwicklungsprojekte betrachtet, wohingegen die Abschlussarbeiten wie wissenschaftliche Arbeiten gehandhabt werden können.

Im Laufe dieser Arbeit werden hauptsächlich Software-Entwicklungsprojekte betrachtet. Ein Kernpunkt dieser Entwicklungsprojekte ist die Tatsache, dass annähernd alle Projekte der Forschung an den jeweiligen Lehrstühlen dienen. Im Laufe der Interviews stellte sich heraus, dass innerhalb dieser Projekte nahezu kein definierter Prozess gelebt, sondern das Vorgehen je nach Bedarf agil angepasst wird. Hieraus ergibt sich die Tatsache, dass die Informationsbedürfnisse an ein Kennzahlen-Dashboard nicht anhand der verwendeten Prozesse verallgemeinert werden können, sondern je nach Projektziel individuell angepasst werden müssen.

III.1.2.1.2 Datenanbindung und Tools

Wie bereits erwähnt stellt das SSE Lab seinen Benutzern bestimmte Dienste zur Verfügung, welche folglich knapp erläutert werden.

Versionsverwaltungssysteme: Die meisten Projekte¹ nutzen neben der Eclipse-Entwicklungsumgebung *subversion (svn)* [6] als Versionsverwaltungssystem. Weiterhin steht auch *git* [1] neben *svn* als integrierter Basisdienst des SSE Labs zur Verfügung.

Change-Request-Management-Systeme: Ein weiteres Werkzeug, welches häufig genutzt wird, ist das Change-Request-Management-System *Trac* [13], mit dessen Hilfe Fehler, wie auch neue Anforderungen erhoben und zur Abarbeitung bereit gestellt werden können.

Speichersysteme: Ein internes Speichersystem wird ebenfalls vom SSE Lab als Dienst zur Verfügung gestellt, in welchem Daten und Dokumente anderen SSE Lab-Mitgliedern zur Verfügung gestellt werden können. Zusätzlich besteht die Möglichkeit einer Einbindung dieses Systems in das Dateisystem des eigenen Rechners.

¹wenn nicht anders definiert, so gilt ab jetzt: *Projekt = Software-Entwicklungsprojekte*

Wikis und Mailinglisten: Zur Organisation und Informationsaufbereitung von internen oder externen Daten werden auch diese Dienste innerhalb von Projekten genutzt. Mittels der Mailinglisten können beispielsweise wichtige Informationen, wie auch der Status aktueller Commits oder ähnliche Mitteilungen an Mitglieder des Projekt-Labs vermittelt werden. Innerhalb von Wikis können Informationen zum Projekt, Dokumentationen oder beliebige andere Daten sowohl projektintern, als auch mit externen Projektinteressenten geteilt werden.

Code-Generatoren: Einige Projekte nutzen des Weiteren die Zusatzdienste der intern vorhandenen Code-Generatoren wie beispielsweise *MontiCore* [27] oder *MontiArc* [26], auf die an dieser Stelle nicht weiter eingegangen wird.

Neben den vorhandenen SSELab-Diensten, werden innerhalb der Entwicklungsprojekte oftmals zusätzliche externe Werkzeuge, wie typische CASE-Tools welche in Abschnitt I.2.3.2 dargestellt wurden, genutzt. An dieser Stelle werden die im SSELab verwendeten Werkzeuge kurz umrissen.

Continuous-Integration-Systeme: Diese Werkzeuge dienen der kontinuierlichen Prüfung auf Ausführbarkeit und Fehler der Software nach dem Commit eines Entwicklers. Innerhalb des SSELabs werden hauptsächlich die beiden Systeme *Jenkins* [2] und *Hudson* [12] verwendet.

Build-Management-Tools: Sie dienen dazu den Build-Prozess innerhalb des Software-systems zu automatisieren. Diese Vereinfachung geschieht innerhalb der SSELab-Projekte mittels *Apache Maven* [32] oder *Apache Ant* [28].

(Microsoft) Office Tools: Für Datenaufbereitungen, wie auch die Projektplanung werden sowohl das Microsoft Office Tools, wie *MS Project* [30], als auch Text- oder Tabellenkalkulationsdokumente genutzt.

III.1.2.1.3 Benutzerrollen

Da die Projekte im SSELab, wie eingangs beschrieben, hauptsächlich agile Forschungsprojekte in kleinem Rahmen sind, existieren nur zwei explizite Benutzerrollen: User und Manager.

User: Sie sind die normalen Benutzer, welche das SSELab hauptsächlich für Projektarbeiten oder Informationsgewinnungen nutzen. Sie können in Softwareprojekten die *Entwickler* sein oder in Organisationsprojekten reine Beobachter oder *Nutzer der dargebotenen Inhalte*.

Manager: Ihre Aufgabe hingegen ist die Leitung der Organisation und Projekte. Innerhalb der Rolle des Projektmanagers existieren zwei Ausprägungsarten: der *reine Projektmanager eines Projektes* und der *Multi-Projektmanager*, welcher mehr als ein einzelnes Projekt leitet und organisiert. In kleinen SSELab-Projekten fällt die Rolle des Projektmanagers und des Entwicklers oftmals zusammen.

III.1.2.2 Frage 2: Welche Maße und Mittel werden zur Qualitätssicherung aktuell verwendet?

Die Bandbreite der Qualitätssicherung ist wie in Abschnitt I.3.4 beschrieben groß. Sie startet bei der Produktqualität im Rahmen von Codemetriken und Testumgebungen und geht über zur Projektqualität, welche Zeiteinhaltung oder Ticketabarbeitung beinhalten kann. Im folgenden werden aktuelle Maßnahmen zur Qualitätssicherung innerhalb der SSELab-Projekte genauer erörtert.

Codemetriken werden in fast allen Projekten zur Qualitätssicherung angewendet. Der Unterschied innerhalb der Projekte besteht in der Priorisierung und Verwendung der Metriken. Zur Erhebung der Code-Metriken wird hauptsächlich das Tool *Sonar* [4] verwendet, welches Kennzahlen der Vermessung von *Lines of Code* über den *Testerfolg* bis hin zu *Verletzungen von Code-Konventionen* bereit stellt.

Test-Infrastrukturen sind nach den Code-Metriken die zweit häufigste Methode zur Sicherung der Qualität. In den meisten Fällen werden Funktionalitäten mittels JUnit-Tests [3] geprüft. Weiterhin verwenden wenige Projekte auch andere Testinfrastrukturen zur Überprüfung von Benutzeroberflächen oder die Erfüllung der funktionalen Anforderungen.

Continuous-Integration- oder Nightly-Build-Systeme werden ebenfalls in vielen Projekten als Qualitätssicherungsmaßnahme verwendet. Nightly Builds dienen der täglichen Überprüfung, ob das System stabil gebaut werden kann und keine Tests fehlschlagen. CI geht noch einen Schritt weiter, indem sie bei jedem Commit eines Entwicklers obige Faktoren überprüfen.

Ticketsysteme sind auch innerhalb des SSELabs durch das zuvor genannte das Tool *Trac* integriert. Dieses System wird in einigen Projekten genutzt um Fehler, Bugs und Features zu sammeln, aber auch Aufgaben und Zuständigkeiten an Projekt-Mitarbeiter zu adressieren.

Reviews dienen der Überprüfung von Artefakten. So werden innerhalb der Projekte Modelle, aber auch Code beispielsweise durch die Projektmanager oder erfahrene Mitarbeiter geprüft.

III.1.2.3 Frage 3: Welche Wünsche an die Qualität existieren?

Die grundsätzlichen Wünsche an die Projektqualität waren bei nahezu allen Projektmanagern gleich:

- hoher Testerfolg
- gute Testabdeckung
- erfolgreicher Buildstatus
- Einhaltung der Kodierrichtlinien

Des Weiteren unterschieden sich weitere Qualitätsaspekte je nach Größe und Mitarbeiterzahl:

- adäquater Projektfortschritt
- Einhaltung von Projektterminen
- Reviews von Code-Commits
- anforderungsnahe Entwicklung
- geringe Architekturverletzungen durch die Entwickler

III.1.2.4 Frage 4: Welche „speziellen Metriken“ werden gewünscht?

Allgemeine Dashboard-Kennzahlen für SSELab-Projekte werden in Abschnitt III.2.2.3 erläutert und sollen an dieser Stelle nicht vorweg genommen werden. Weiterhin traten in den Interviews Sonderwünsche auf, welche vorerst nicht weiter berücksichtigt wurden.

III.1.2.5 Frage 5: Existieren weitere Anregungen oder Wünsche?

Viele Anregungen und Wünsche liefen auf die Weiterentwicklung des SSELabs hinaus, welche allerdings nicht Teil dieser Arbeit ist. Diese Anmerkungen wurden jedoch an die entsprechenden SSELab-Entwickler weitergeleitet.

Des weiteren wurden typische Anforderungen an das Dashboard dargelegt, die in die Konzept-Anforderungen aus Abschnitt II.2.3 mit eingeflossen sind. Speziell für das SSELab-Dashboard werden an dieser Stelle die wichtigsten genannten Anforderungen kompakt erläutert.

III.1.2.5.1 Konfigurabilität

Einer der wichtigsten Punkte der befragten Projektmanager war der Wunsch nach großer Anpassungsfähigkeit. So soll eine schnelle und einfache Konfiguration möglich sein, Änderungen innerhalb des konfigurierten Dashboards müssen schnell durchführbar sein und die Variabilität der Datenansicht soll ebenfalls zur Verfügung stehen.

III.1.2.5.2 Dokumentation

Des weiteren besteht das Bedürfnis an einer guten Dokumentation. So sollen einzelne Widgets detailliert mit ihren Informationsbedürfnissen und Interpretationshilfen dokumentiert sein. Weiterhin besteht der Wunsch einzelne Widgets durch den Dashboard-Nutzer mit Kommentaren zu versehen, um beispielsweise schlechte Testwerte als „false-positive“ zu markieren oder Hinweise bezüglich der Interpretation anzumerken.

III.1.2.5.3 Multi-Projekt-Dashboard

Insbesondere Manager mehrerer Projekte wünschten sich ein allumfassendes Überblicks-Dashboard mit den wichtigsten Kennzahlen aller Projekte. Dieses soll nicht der genauen Erfassung der einzelnen Projekte, sondern nur einer kurzen Übersicht des Projektstatus dieser dienen.

III.1.2.5.4 Informationen aus Nutzungsanalyse

Der Wunsch nach Informationen bezüglich der Nutzung von Kennzahlen durch andere Dashboard-Nutzer gleicher Rolle wurde ebenfalls ausgesprochen. So existiert der Wunsch der Projektleiter anhand der Nutzung anderer Manager ihr Messverhalten zu überprüfen und zu verbessern. An dieser Stelle setzen die in dieser Arbeit entwickelten Dashboard-Vorlagen an, welche den Benutzern eine Hilfestellung für die Verwendung von rollentypischen Kennzahlen geben sollen.

III.1.3 Fazit der Anforderungsanalyse

Wie innerhalb der Anforderungsanalyse festgestellt wurde, besteht ein Großteil der SSELab-Projekte aus agilen Forschungsprojekten, welche hauptsächlich die Rollen des Projektmanagers und des Entwicklers beinhalten. Neben den einfachen Projektmanagern existieren auch Multi-Projektmanager, welche mehrere Projekte gleichzeitig leiten und somit eine zusätzliche Übersicht der relevanten Kennzahlen aller Projekte benötigen.

Der Fokus dieser Arbeit liegt bei den Kennzahlen für die Rolle des Projektmanagers. Entwickler-Kennzahlen können mithilfe von Werkzeugen wie *Sonar* [4] erhoben und somit einfach in das *EMI*-System integriert werden. Kennzahlen für die Erhebung des Projektstatus sind unter Umständen schwerer zu ermitteln, weswegen diese in den Mittelpunkt dieser Arbeit rücken.

III.2 Dashboard-Vorlagen für das SSELab

Miß alles, was sich messen läßt,
und mach alles meßbar, was sich
nicht messen läßt.

ARCHIMEDES

Inhalt

III.2.1 Qualitätsziele und Informationsbedürfnisse der SSELab-Projekte . .	105
III.2.2 Dashboard-Vorlage: Projektmanager	109
III.2.3 Fazit der Erstellung von Dashboard-Vorlagen	113

III.2.1 Qualitätsziele und Informationsbedürfnisse der SSELab-Projekte

Durch die in Abschnitt III.1.2.3 beschriebene dritte Frage, wurden Qualitätsziele der SSELab-Projektmanager grob erläutert. In diesem Abschnitt werden diese Ziele anhand eines angepassten Qualitätsmodells (siehe Abbildung III.2.1) definiert, welches sich von dem in Abbildung I.3.3 (S. 23) dargestellten Qualitätsmodell von Ludewig und Lichter ableiten lässt. Anstatt der in [LL07] beschriebenen *Prozessqualität*, wird innerhalb der SSELab-Projekte der Begriff der *Projektqualität* eingeführt, da innerhalb der evaluierten Forschungsprojekte keine klar definierten und strukturierten Prozesse gelebt werden.

In den folgenden Abschnitten werden sowohl die Projekt- als auch Produktqualitätsziele der SSELab-Projekte, wie auch die Informationsbedürfnisse an das zu integrierende EMI-Dashboard SCREEN in Teilen dargestellt. Eine komplette Übersicht der Informationsbedürfnisse befindet sich in Anhang B.

III.2.1.1 Projektqualität

Durch die unterschiedliche Nutzungsmöglichkeit des SSELabs innerhalb verschiedener Projektbereiche, ist das Interesse einer hohen Projektqualität bei den Projektmanagern von unterschiedlich starker Ausprägung. Projekte, eines kleinen Umfangs mit wenig entwickelnden Mitarbeitern, setzen ihren Fokus oftmals auf die in Abschnitt III.2.1.2 beschriebene Produktqualität. Projektmanager größerer Projekte achten hingegen auch auf die Qualität der Projektleistung und Planungssicherheit.

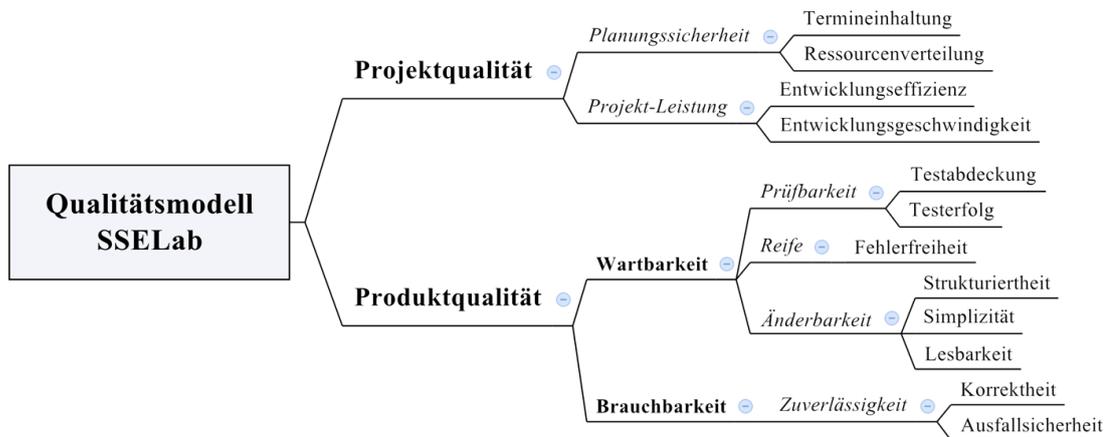


Abbildung III.2.1: Qualitätsmodell der SSELab-Softwareentwicklungsprojekte

III.2.1.1.1 Planungssicherheit

Vor allem innerhalb Projekte, deren Meilensteine beziehungsweise Releases eingehalten werden müssen, ist der Überblick über die *Termeinhaltung* von großem Interesse. Zusätzlich kann für die Projektmanager auch die *Verteilung der Ressourcen* (in diesem Fall Entwickler) von Interesse sein, insbesondere wenn diese für mehrere Aufgaben eingeteilt sind oder nur ein begrenztes Kontingenz an Arbeitszeit zur Verfügung haben. Aus diesen Qualitätszielen lassen sich folgende Fragen ableiten:

- Verläuft das Projekt nach Plan?
- Können Projekttermine eingehalten werden?
- Werden alle geplanten Features bis zum Meilenstein fertig gestellt?
- Werden alle offenen Tickets bis zum Meilenstein fertig gestellt?
- Wie sieht der aktuelle und der voraussichtliche Arbeitsaufwand bis zum nächsten Meilenstein aus?
- Welche Mitarbeiter sind für das Projekt verfügbar?
- Wie verteilt sich die Verfügbarkeit der einzelnen Mitarbeiter auf mehrere Aufgaben?
- Wie viel Arbeitszeit steht ein bestimmten Mitarbeiter noch zur Verfügung?

III.2.1.1.2 Projektleistung

Unter dem Begriff der *Projektleistung* verbergen sich Fragen an die *Entwicklungsgeschwindigkeit* und die *Entwicklungseffizienz*. So muss ein Projektmanager Aufwände einschätzen und anhand der Entwicklungsgeschwindigkeit Fertigstellungsgrade zu bestimmten Meilensteinen bestimmen können. Aus diesem Grund leiten sich daraus unter anderem folgende Fragen ab:

- Wie schnell werden Tickets innerhalb des Projekts abgearbeitet?
- Wie lange dauert die durchschnittliche Ticketabarbeitung?
- Wie verläuft die Abarbeitung der Aufgaben/ To Dos?
- Wie viel Entwicklungsaufwand ist für eine bestimmte Aufgabe prognostiziert?
- Wie viel Aufwand haben die Aufgaben tatsächlich verbraucht?

III.2.1.2 Produktqualität

Wie bereits in Abschnitt III.2.1.1 erläutert, spielt die Produktqualität bei vielen Projektmanagern der SSELab-Projekte eine größere Rolle als die Qualität des Projektablaufs. Vor allem die Qualitätsmerkmale aus den Bereichen der *Wartbarkeit* (Prüfbarkeit, Reife, Änderbarkeit) und der *Brauchbarkeit* (Zuverlässigkeit) haben eine hohe Relevanz.

III.2.1.2.1 Prüfbarkeit

Fragen bezüglich des *Testerfolgs* und der *Testabdeckung* stehen im Mittelpunkt der Prüfbarkeit eines Software-Projekts:

- Wie hoch ist der aktuelle Testerfolg innerhalb des Projekts?
- Wie hat sich die Anzahl der Tests und der Testerfolg im Laufe der Zeit entwickelt?
- Hat sich der Testerfolg nach dem letzten Commit verändert? Und wenn ja, wie?
- Wie hoch ist die momentane Testabdeckung?
- Wie sieht die Entwicklung der Testabdeckung innerhalb des Projekts aus?

III.2.1.2.2 Reife

Die Reife eines Software-Produkts lässt sich vor allem durch die *Anzahl und schwere der Fehler* innerhalb der Entwicklung abbilden. Zwar ist eine fehlerfreie Software nicht garantierbar, jedoch lässt die Kombination von vorhandenen Fehlern mit der existierenden Testabdeckung Rückschlüsse über die Produkt- und Prozessreife zu. Wurden beispielsweise viele Fehler und eine niedrige Testabdeckung aufgedeckt, so könnte diese Aussage zur Interpretation eines niedrigen Reifegrad innerhalb des Entwicklungsprozesses führen,

in dem das Produkt durch zu wenige Tests geprüft wurde. Weiterhin könnte aufgrund der niedrigen Testabdeckung auf weitere, noch nicht entdeckte Fehler geschlossen werden, was die Produktreife als gering erscheinen lässt. So stellen sich im Bereich der Reife unter anderem folgende Fragen:

- Wie hoch ist die Anzahl der offenen Fehler?
- Welche Aussagen lassen sich durch die Kombination aus Testabdeckung und Testerfolg treffen?
- Wie verläuft die Abarbeitung der offenen Fehler?
- Wie entwickeln sich die Fehler innerhalb des gesamten Projekts?
- Wie sieht die Verteilung der Fehler nach Härtegrad beziehungsweise Priorität aus?

III.2.1.2.3 Änderbarkeit

Das Qualitätsmerkmal der Änderbarkeit beinhaltet eine gute *Lesbarkeit*, wie auch eine gewisse *Simplizität* des Quellcodes. Hierfür wünschen sich die Projektmanager der SSELab-Projekte eine gute Kommentardichte, die Einhaltung von Kodierrichtlinien, wie auch die Kapselung der Funktionen. Zusätzlich soll die *Strukturiertheit* innerhalb des Quellcodes (beispielsweise in Komponenten, Pakete, Klassen), wie auch innerhalb des Entwicklungsprozesses, etwas durch die Verwendung von Tickets zur Aufgabenverteilung und daraus folgend die Benennung einer Ticket-ID bei jedem Code-Commit, überprüft werden. Aufgrund dieser Eigenschaften gehen folgende Fragen hervor:

- Wie hoch ist die Dichte der Kommentare innerhalb des Quellcodes?
- Besitzt jede Klasse/Methode einen Kommentar?
- Wie hoch ist der Verletzungsgrad der Kodierrichtlinien?
- Welches sind die Top-5-Regelverletzungen innerhalb dieser Richtlinien?
- Wurden Entwurfsmuster korrekt verwendet?
- Wie hoch ist die Abhängigkeit der Klassen/Pakete/Komponenten innerhalb des Projekts?
- Wie viele Commits enthalten keine Ticket-ID?

III.2.1.2.4 Zuverlässigkeit

Die Zuverlässigkeit eines Software-Produkts beinhaltet neben der *korrekten Implementierung der Anforderungen*, auch eine hohe *Ausfallsicherheit*. Diese kann mittels Tests bezüglich der Anforderungen und der regelmäßigen Überprüfung des Produkt-Builds (beispielsweise einmal täglich via *Nightly Build* oder gar bei jedem Commit via *Continuous Integration*) überprüft werden. Aus diesen Anforderungen stellen sich folgende Fragen:

- Wie hoch ist die Testabdeckung bezüglich der vorhandenen Anforderungen?
- Wie verlaufen die Tests?
- Welcher Commit hat Probleme innerhalb der Tests verursacht?
- Wie verläuft der Build?
- Welche Probleme treten beim Build auf?
- Welcher Commit hat Probleme in Build verursacht?

Die Auswahl der Fragen spiegelt das Informationsbedürfnis von Projektmanagern der evaluierten SSELab-Projekten wider. Mithilfe dieser wird eine Dashboard-Vorlage erstellt, welche die Qualitätsziele und Informationsbedürfnisse mit möglichen Kennzahlen verbindet. Diese Vorlage entspricht einem rollenspezifischen Katalog an Metriken, welcher dem Dashboard-Nutzer innerhalb des initialen Konfigurationsprozesses die Möglichkeit der Auswahl passabler Kennzahlen liefern soll. Folglich wird eine erste Vorlage erstellt, welche Metriken beinhaltet, die in der ersten Integrationsiteration einfach zu erheben und zu implementieren sind.

III.2.2 Dashboard-Vorlage: Projektmanager

Aus den obigen Qualitätskriterien gepaart mit den Fragestellungen zur Abdeckung der Informationsbedürfnisse können nun Dashboard-Vorlagen erstellt werden. Innerhalb dieser Arbeit wird die Vorlage zu der Rolle des Projektmanagers exemplarisch erstellt, welche in einem ersten Iterationsschritt in das *EMI*-System integriert werden kann. Die komplette Vorlage kann in Anhang C betrachtet werden, an dieser Stelle der Arbeit werden lediglich Teilelemente der Vorlage besprochen.

III.2.2.1 Schritt 1: Auswahl der Informationsbedürfnisse

Durch die Erhebung der Anforderungsanalyse wurde klar, dass innerhalb des SSELabs hauptsächlich zwei Projektrollen vorhanden sind: Entwickler und Projektmanager. Durch die Forschungsnähe und der Anhäufung vieler kleiner Projekte ist eine tatsächliche Abgrenzung zwischen Managern und Entwicklern schwer herzustellen, da viele Projektmanager innerhalb dieser Projekte selber mitentwickeln. Aus diesem Grund wird an dieser Stelle eine Vorlage erstellt, die auf Basis der Benutzerrolle des Projektmanagers basiert, jedoch auch wenige Entwickleraspekte beinhaltet.

Vor allem Fragestellungen zur Planungssicherheit und der Projektleistung liegen im Fokus einer Projektmanagers. Des weiteren muss jedoch auch die Zuverlässigkeit, Reife, Prüfbarkeit und Änderbarkeit gegeben sein um ein komplettes Bild des aktuellen Projektstatus zu erhalten.

Tabelle III.2.1: Gliederungsmerkmale der SSELab-Informationsbedürfnisse

Qualitätsziel	Qualitätskriterium	Merkmal
Projektqualität	Planungssicherheit	Zeit Kosten Personal
	Projektleistung	Fortschritt Qualität
Produktqualität	Prüfbarkeit	Testabdeckung
	Reife	Fehler
	Änderbarkeit	Dokumentation Architektur
	Zuverlässigkeit	Build

III.2.2.2 Schritt 2: Gliederung der Fragestellungen

Die reine Auflistung der Fragestellungen, welche zu den zuvor genannten Qualitätskriterien passend sind, führt zu einem großen Katalog an Informationsbedürfnisse, die für den Benutzer des Dashboards nur schwer eine Auswahl passender Fragen ermöglichen. Aus diesem Grund werden diese Qualitätskriterien in weitere Merkmalsklassen unterteilt, welche in Tabelle III.2.1 dargestellt werden.

So können beispielsweise für die Projektleistung folgende Fragestellungen den Merkmalen zugeordnet werden:

- Fortschritt
 1. Wie verläuft die Ticketabarbeitung innerhalb des Projekts?
 2. Wie lange dauert die durchschnittliche Ticketabarbeitung?
 3. Wie verläuft die Abarbeitung der To Dos?
 4. Wie verläuft die Abarbeitung offener Fehler?
- Qualität
 6. Wie entwickeln sich die Fehler innerhalb der Projektlaufzeit?
 7. Wie hat sich die Anzahl der Tests innerhalb des Projekts entwickelt?
 8. Wie verläuft die Entwicklung des Testerfolgs?
 9. Wie entwickelt sich die Testabdeckung?
 10. Wie verlaufen die Builds im Allgemeinen?

Durch die zuvor getätigte Aufarbeitung der Informationsbedürfnisse, welche im gesamten Umfang in Anhang B zu finden sind, lässt sich feststellen, dass beispielsweise Fragestellung 3 anhand der vorhandenen Datenquellen nicht direkt beantwortet werden können. Dieses und weitere Informationsbedürfnisse werden im Zuge der Dashboard-Vorlage des ersten Iterationsschrittes vernachlässigt, können zu einem späteren Zeitpunkt jedoch in die Vorlage integriert werden. Eine vollständige Übersicht der in dieser Arbeit erstellten Vorlage ist in Anhang C zu finden.

III.2.2.3 Schritt 3: Auswahl passender Metriken

Nach der Auswahl und Gliederung der benötigten Informationsbedürfnisse, müssen Metriken zur Berechnung und Darstellung der Kennzahlen evaluiert werden. An dieser Stelle beispielhaft Fortschrittsmetriken der Projektleistung detailliert besprochen.

So sind die für diese Vorlage ausgewählten Fragen der Fortschrittsmessung folgende:

- Wie verläuft die Ticketabarbeitung innerhalb des Projekts?
- Wie lange dauert die durchschnittliche Ticketabarbeitung?
- Wie verläuft die Abarbeitung offener Fehler?

Zur Beantwortung dieser Fragestellungen kann das vorhandene Ticketsystem *Trac* [13] benutzt werden, unter der Voraussetzung dass die Daten in diesem System gepflegt und auf aktuellem Stand gehalten werden. So kann eine ausführliche Ticketanalyse bezüglich der Ticketabarbeitung den Projekt-Fortschritt darstellen. Drei mögliche Kennzahlen hierzu wären:

- Historie der Anzahl an abgearbeiteten Tickets
- Entwicklung der Statusanzahl vorhandener Tickets
- Durchschnittliche Ticketabarbeitungsdauer

Die Aussagekraft dieser Kennzahl kann nur durch die richtige und regelmäßige Verwendung des Ticketsystems gegeben sein. So müssen Tickets beispielsweise nach der Fertigstellung geschlossen werden um sie als abgeschlossen zu betrachten. Des weiteren stellt sich die Frage, wie die durchschnittliche Abarbeitungsdauer ermittelt werden kann. Aktuell bietet die integrierte *Trac*-Version nur die Möglichkeit der Setzung eines Start- und Enddatums. Aus diesem Grund muss im ersten Iterationsschritt von einer Abarbeitung in Tagen statt in Stunden ausgegangen werden. Für eine genaue Abarbeitungsdauer in Stunden muss das Ticketsystem an die Bedürfnisse der genaueren Evaluation angepasst werden.

Die gesamte Dashboard-Vorlage für die Rolle des Projektmanagers im SSELab ist in Anhang C dargestellt.

III.2.2.4 Schritt 4: Auswahl geeigneter Darstellungsformen

Nach einer erfolgreichen Ausarbeitung möglicher Kennzahlen und dem Abgleich dieser mit den vorhandenen Datenquellen, müssen diese Metriken mittels Dashboard-Widgets visualisiert werden. Für eine passende Auswahl der Visualisierungsform wird an dieser Stelle, wie zuvor auch im allgemeinen Konzept (Abschnitt II.4.2.4), auf das Werk von Stephen Few [Few12] verwiesen.

Für die zuvor genannten Metriken wurden beispielhaft zwei Visualisierungen erstellt, welche sich an die Richtlinien des Dashboard-Designs (siehe Abschnitt I.4.2.1.3) orientieren. Abbildung III.2.2 stellt die visualisierte Darstellung der Status-Entwicklung von Projekt-Tickets dar. Der aktuelle Zeitpunkt wird anhand der blauen Linie dargestellt, die Anzahl aller Tickets durch die graue und die Säulen innerhalb des Diagramms stellen die Anzahl der gesamten Tickets aufgeteilt nach ihrem Status dar.

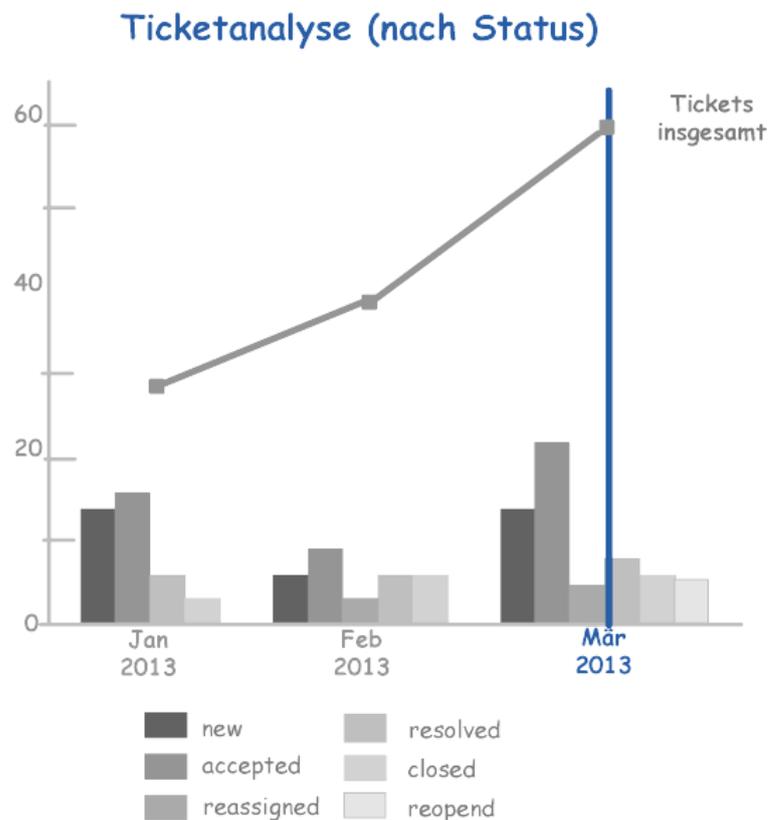


Abbildung III.2.2: Schritt 2: Prototyp-Erstellung des *EMI*-Dashboards

Des weiteren wird bildet Abbildung III.2.3 die Visualisierung der durchschnittlichen Abarbeitungsdauer der Tickets in Tagen ab. Auch hier spiegelt die blaue Linie den aktuellen Erhebungszeitpunkt wider. Die durchschnittliche Dauer der Ticketabarbeitung

wird in diesem Dashboard-Widget als Boxplot (siehe auch Abbildung I.4.4 (S. 43) für weitere Erläuterungen) dargestellt.

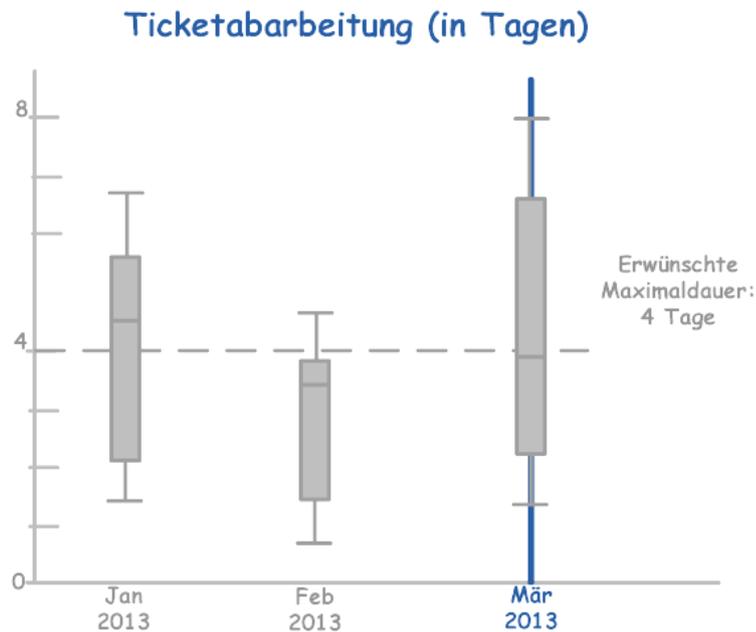


Abbildung III.2.3: Schritt 2: Prototyp-Erstellung des *EMI*-Dashboards

III.2.3 Fazit der Erstellung von Dashboard-Vorlagen

Innerhalb dieser Integrationsschritte wurde aus den evaluierten Informationsbedürfnissen der Anforderungsanalyse eine Auswahl der wichtigsten Fragestellungen und Kennzahlen zur Erstellung einer Projektmanager-Vorlage getroffen. Die Durchführung dieser Schritte zeigte, dass unter Umständen die Notwendigkeit der Anpassung vorhandener Datenquellen existiert, da gewünschte Informationsbedürfnisse andernfalls nicht zu erhalten sind. Weiterhin spielt für die Erhebung aussagekräftiger Kennzahlen auch die Nutzung der entsprechenden Messdatensysteme aus Sicht des Messkunden eine große Rolle. Nur durch die regelmäßige Aktualisierung und Verwendung dieser Systeme können effektiv Daten zur Projektsteuerung gewonnen werden.

III.3 Prototyp eines *EMI*-Dashboards für das SSELab

Das Beispiel ist einer der erfolgreichsten Lehrer, obgleich es wortlos lehrt.

SAMUEL SMILES

Inhalt

III.3.1 Schritt 1: Betrachtung der Prototyping-Ziele	115
III.3.2 Schritt 2: Erstellung der Prototypen	115
III.3.3 Schritt 3: Evaluation durch den Benutzer	117
III.3.4 Fazit des Prototypings	118

Nachdem in vorigen Kapitel die Erstellung einer Dashboard-Vorlage für SSELab-Projektmanager beschrieben wurde, wird in diesem Kapitel die Erstellung des Prototyps umrissen. Hierbei werden die drei Vorgehensschritte der Zielbetrachtung, Entwicklung des Prototyps und der Evaluation durch den Benutzer des in Abschnitt II.5.3 vorgestellten Integrationskonzepts beachtet.

III.3.1 Schritt 1: Betrachtung der Prototyping-Ziele

Im Laufe der Anforderungsanalyse wurde ersichtlich, dass nicht alle Projektmanager mit dem Aspekt der Kennzahlen-Erhebung zur Qualitätssicherung vertraut waren. Vor allem der Begriff des Kennzahlen-Dashboards war nicht allen geläufig. Aus diesem Grund dient der folglich erstellte Prototyp der Darstellung eines möglichen Dashboard mittels beispielhafter Kennzahl-Darstellungen.

Ein zweites Ziel des prototypischen Ansatzes ist die Ermittlung der Einfachheit und Handhabbarkeit der Konfiguration, welche einen zweiten Prototypen benötigt. Folglich wird die Erstellung dieser Prototypen beschrieben.

III.3.2 Schritt 2: Erstellung der Prototypen

Innerhalb dieses Schrittes werden die beiden entwickelten Prototypen genauer erläutert.

III.3.2.1 Dashboard-Prototyp

Aufgrund der Erläuterungen innerhalb der Interviews zur Anforderungsanalyse, wurde den Projektmanagern typische Grundlagen im Bereich der Dashboard-Widgets und des Dashboard-Designs erläutert. Um diese Informationen zu visualisieren wurde ein digitaler Papier-Prototyp entwickelt, welcher die wichtigen Design-Aspekte nach Stephen Few (siehe auch Abschnitt I.4.2.1.3) beinhalten und der Optik des *SCREEN*-Dashboards nahe kommt. Da das Prototyping-Werkzeug von Matthias Gora [Gor13] zu diesem Zeitpunkt der Arbeit in der Entwicklung steckte, wurde *Microsoft Visio* zur Erstellung verwendet.

In einem ersten Schritt wurden verschiedene Kennzahlen mit beispielhaften Visualisierungen ausgewählt. Diese werden in Tabelle III.3.1 vorgestellt.

Tabelle III.3.1: Kennzahlen und ihre Darstellungsform innerhalb des SSELab-Dashboard-Prototypen

Kennzahl	Darstellungsform
Allgemeine Projektkennzahlen	Bulletgraphen
Buildstatus	Tabelle inklusive Trendanzeige
Top-5-Regelverletzungen	Tabelle inklusive Trendanzeige
Ticketanalyse nach Status	Kombination aus Säulen- und Liniendiagramm
Verteilung der offenen Tickets nach Priorität	Kombination aus Säulen- und Liniendiagramm
Durchschnittliche Ticketbearbeitung in Tagen	Boxplots
Fortschrittsanzeige	Meilensteinplan

In einem weiteren Schritt wurden die erstellten Dashboard-Widgets auf einem Hintergrund des *SCREEN*-Dashboards wie in Abbildung III.3.1 (S. 119) angeordnet. Zusätzlich wurden die Informationsbedürfnisse, welche durch das angegebene Dashboard beantwortet werden, als Übersicht integriert. An dieser Stelle fällt die Verwendung der Comic Sans Schriftart auf, welche benutzt wurde um den Charakter eines Prototyps darzustellen. Da dieser Prototyp nicht in erster Linie für das Verständnis der Benutzerinteraktion, sondern für die Vermittlung des Designs erstellt wurde, wurde in diesem Fall auf eine skizzenhafte Darstellung der Dashboard-Widgets verzichtet.

und den Evaluationspartnern vermittelt.

III.3.2.2 Konfigurationsprototyp

Neben der prototypischen Darstellung des Kennzahlen-Dashboards, wurde auch ein Prototyp für die Konfiguration durch den Benutzer erstellt. Aufgrund der vorhandenen *Microsoft Visio* Vorlage des Dashboards, wurde dieses Werkzeug auch für die Erstellung der Konfigurationsschritte benutzt und anhand einer Datei aus mehreren Folien an die entsprechenden Benutzer geschickt.

Die Evaluation des Feedbacks zeigte, dass diese Art der prototypischen Darstellung für die Konfigurationsbedienung ungeeignet ist. Vor allem stellte sich heraus, dass dieser Prototyp trotz weitergehenden Erläuterungen nicht selbsterklärend war. Aus diesem Grund konnte Feedback nur durch die Möglichkeit von Interviews eingeholt werden, welche sowohl für die Interviewer als auch die Interviewten Zeitaufwand bedeutete.

Anhand der erhaltenen Rückmeldungen wurde ein verbesserter Papier-Prototyp erstellt, der durch die manuelle Verwendung von Papier und Stiften Interaktionen mit dem Benutzer zulässt. Abbildung III.3.2 (S. 120) und Abbildung III.3.3 (S. 120) zeigen zwei exemplarische Bilder des Papier-Prototypen.

III.3.3 Schritt 3: Evaluation durch den Benutzer

Anhand der Prototypen wurde Nutzer-Feedback gesammelt, welches zur Verbesserung des Designs und der Konfiguration genutzt wurde. Folglich werden die wichtigsten Punkte des Feedbacks erläutert.

III.3.3.1 Dashboard-Prototyp

Die Evaluation des Dashboard-Prototyps hat hauptsächlich positives Feedback ergeben. Die Möglichkeit und die Art der Visualisierung hat Anklang gefunden, lediglich die nicht integrierte Option an weiteren Informationen bezüglich der Dashboard-Widgets durch ein Mouse-Over wurde kritisiert.

III.3.3.2 Konfigurationsprototyp

Der erste Konfigurationsprototyp stieß auf wenig Anklang aufgrund eines Versuchs der Fern-Evaluation. Daraufhin wurde nach einigen Feedback-Interviews ein angepasster Papier-Prototyp erstellt. Folglich werden die wichtigsten Kritikpunkte aufgegriffen.

III.3.3.2.1 Digitaler Papier-Prototyp

Im ersten Schritt der Feedback-Runde, der Fern-Evaluation durch die Zusendung des Prototyps via E-Mail, ergaben sich keine Rückmeldungen durch die Ansprechpartner der Anforderungsanalyse. Erst durch Feedback-Interviews, bei denen der Prototyp erklärt wurde, kam es zu Anmerkungen und Kritik. Die Kritikpunkte werden mit der Visualisierung dieses Prototyps in Anhang D weitgehend erläutert, an dieser Stelle sei nur erwähnt,

dass die Informationsdichte innerhalb des Prototyps zu hoch war, was zur Überforderung des Nutzers, wie auch einer Abneigung der Benutzung dieser Konfiguration führte. Aufgrund dieser Erkenntnisse wurde ein zweiter händisch gezeichneter Prototyp erstellt, in dessen die Kritikpunkte und Verbesserungsvorschläge integriert wurden.

III.3.3.2 Echter Papier-Prototyp

Durch den gebastelten Papier-Prototypen konnte das Feedback der Benutzer interaktiv eingeholt werden. Die verbesserte Visualisierung durch die Gliederung und Strukturierung der Informationsbedürfnisse, wie auch andere Interaktionsmöglichkeiten führten zu positivem Feedback, welches nur marginale Änderungen im Design hervorbrachte.

III.3.4 Fazit des Prototypings

Im Allgemeinen ist das Prototyping für den Abgleich der Vorstellungen bezüglich des Designs, der Funktionalität und der Konfiguration sehr wichtig. Wäre in diesem Fall kein Prototyp erstellt worden, so wäre eine Dashboard-Konfiguration implementiert worden, welche unbenutzbar gewesen wäre. Aus diesem Grund ist das Prototyping für die Einsparung an Aufwand (vor allem im Bereich der Änderungen) essentiell.

Des weiteren konnte festgestellt werden, dass ein digitaler Prototyp zwar sinnvoll für ein originalgetreues Design ist, jedoch einen großen Zeitaufwand bezüglich der Entwicklung mit sich bringt. Aus diesem Grund empfiehlt es sich vor allem im Bereich von möglichen Benutzerinteraktionen Papier-Prototypen zu nutzen, welche in Feedback-Gesprächen mit den Benutzern evaluiert werden. Sollte eine direkte Evaluation nicht möglich sein, so kann ein Video mit allen Interaktionsmöglichkeiten erstellt werden, auf dessen Basis der Benutzer sein Feedback abgeben kann. Diese Möglichkeit empfiehlt sich vor allem bei einer großen Anzahl, als auch bei lokal verteilten Feedback-Gebern.

Der größte Vorteil eines händisch gezeichneten Prototypen ist die Möglichkeit der schnellen Erstellung und Änderung. Es wird dem Nutzer sofort ersichtlich, dass dieser Prototyp zur reinen Evaluation dient und innerhalb des Feedback-Gesprächs können Änderungen in den Prototypen eingezeichnet und evaluiert werden.

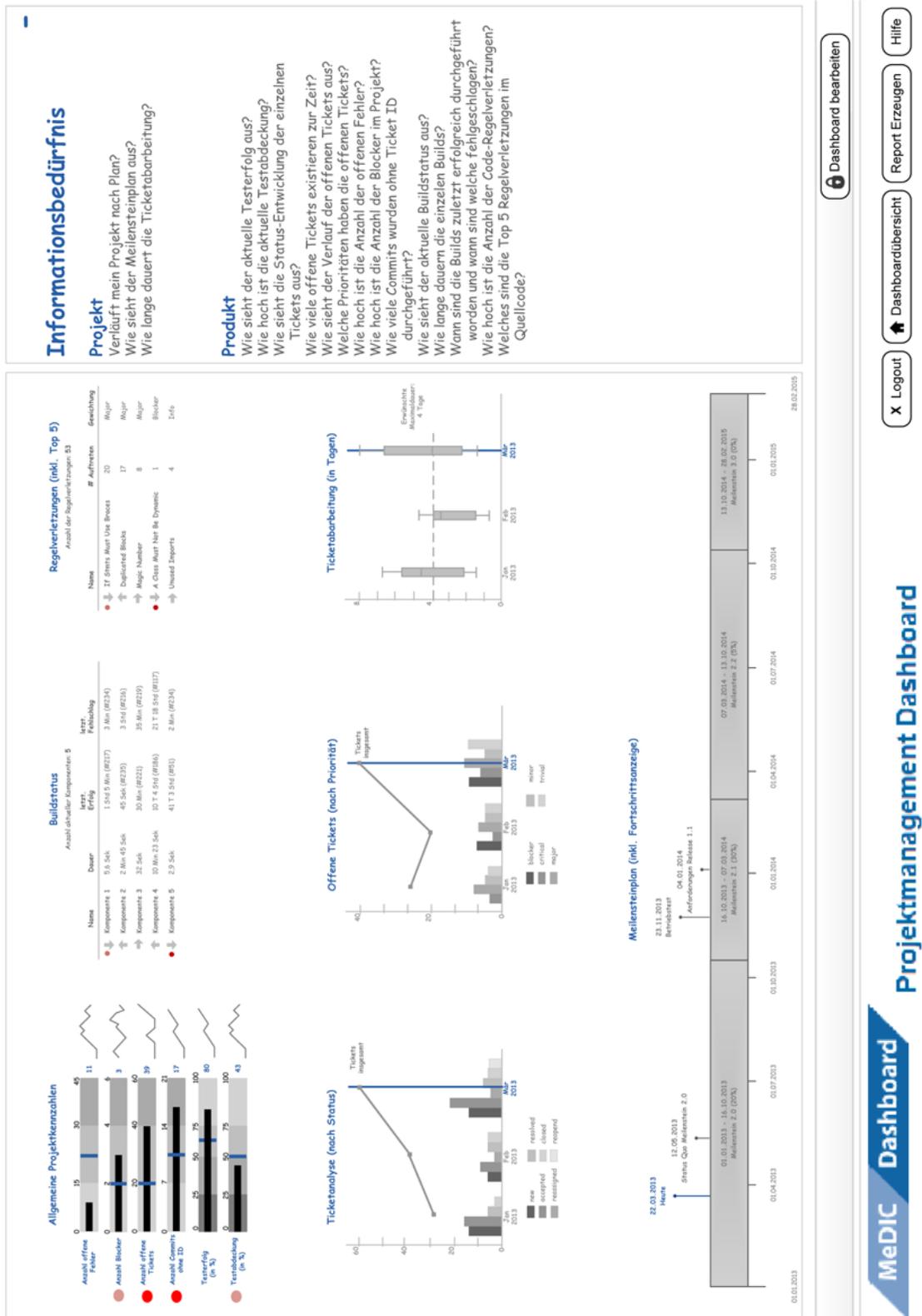


Abbildung III.3.1: Dashboard-Prototyp des SCREEN-Dashboards für das SSELab

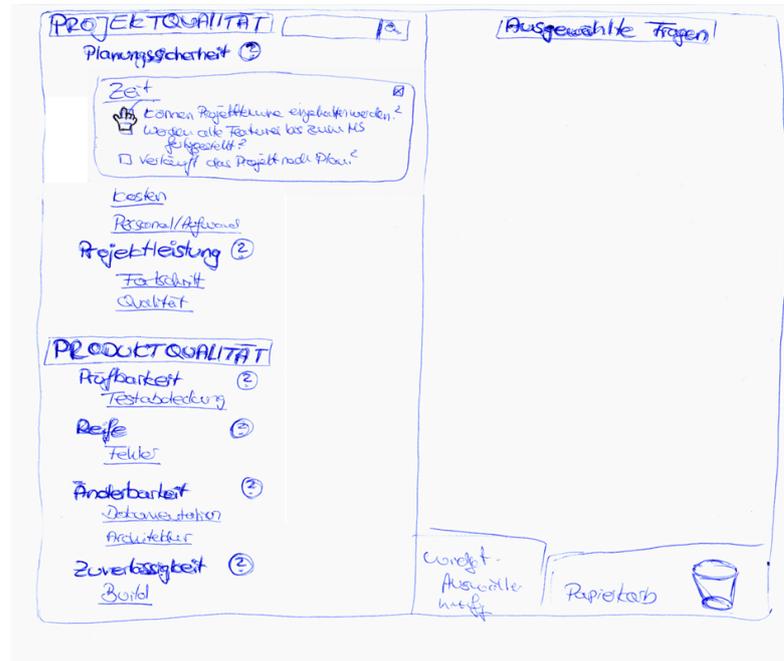


Abbildung III.3.2: Ausgangsbasis zur Konfiguration der Informationsbedürfnisse

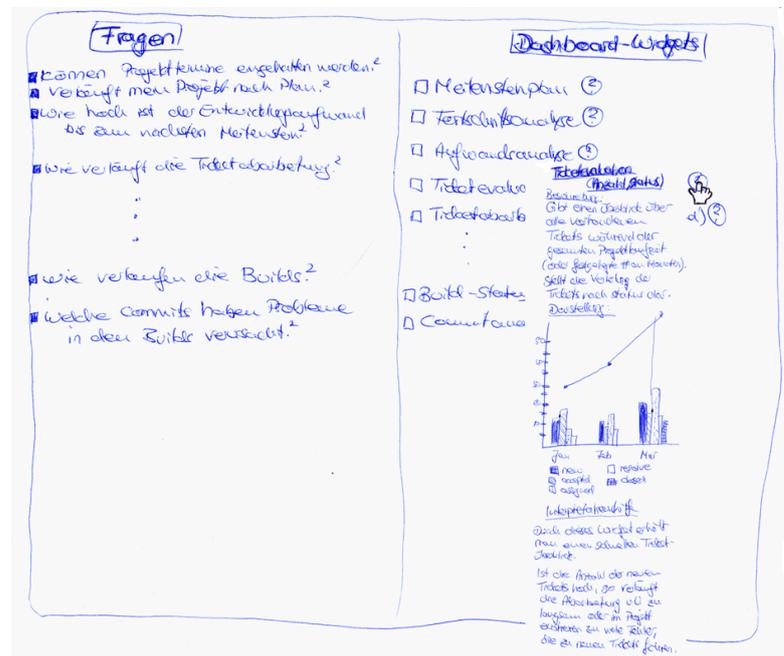


Abbildung III.3.3: Ansicht zur Auswahl der Dashboard-Widgets

III.4 Weiteres Vorgehen

Wer für die Zukunft sorgen will,
muß die Vergangenheit mit
Ehrfurcht und die Gegenwart mit
Mißtrauen aufnehmen.

JOSEPH JOUBERT

Inhalt

III.4.1 Technische Integration	121
III.4.2 Nutzung, Wartung und Weiterentwicklung	122
III.4.3 Fazit der Feldstudie im SSELab	122
III.4.4 Ausblick	123

In diesem letzten Kapitel des Teil III wird ein Ausblick für das weitere Vorgehen der technischen Integration, so wie der Nutzung und Verbesserung des Messsystems gegeben.

III.4.1 Technische Integration

Für die Feldstudie des SSELabs wurde innerhalb dieser Arbeit der Teilbereich der fachliche Integration eines Messsystems durchgeführt und evaluiert. Nach den voran gegangenen Schritten steht die technische Integration des Systems in das SSELab an, welche durch eine weitere Abschlussarbeit von Arthur Otto [Ott13] durchgeführt wird. Folgende Schritte sind unter anderem dafür notwendig:

III.4.1.1 Integration des Messsystem

Das Messsystem muss in die Systemstruktur des SSELabs integriert werden. Hierfür muss nicht nur das Messsystem implementiert werden, sondern zusätzlich die Koppelung der Datenquellen an die *EMI* geschehen.

III.4.1.2 Anpassung der Datenadapter

Die vorhandenen Datenadapter des *EMI*-System können zum großen Teil für die Datenquellen des SSELabs genutzt werden. Sollten weitere Datensysteme benötigt werden zu denen keine Datenadapter existieren, so müssen diese vor der Benutzung implementiert und integriert werden.

III.4.1.3 Konfiguration des Dashboards

Nach der technischen Integration des Systems muss das Dashboard für die Benutzung durch verschiedene Nutzer konfiguriert werden. So müssen die erstellten Dashboard-Vorlagen mit ihren Benutzerrollen, Informationsbedürfnissen, Metriken und Dashboard-Widgets in das System integriert und miteinander verbunden werden. Sollte der Aspekt der Datensicherheit innerhalb des Systems integriert werden, so müssen ebenso Zugriffsrechte bezüglich bestimmter Dashboard-Widgets oder Datenquellen für die Benutzer verteilt werden.

III.4.1.4 Inbetriebnahme

Nach der Konfiguration des Dashboards müssen innerhalb des Dashboards Benutzerkonten erstellt werden, welche den späteren Verwendern die Möglichkeit einer individuellen Nutzung des Dashboards ermöglicht. Wurden all diese Faktoren berücksichtigt, so kann das System in Betrieb genommen werden

III.4.2 Nutzung, Wartung und Weiterentwicklung

Nach der Integration des Systems sollte die Wartung und Weiterentwicklung nicht vernachlässigt werden. So können beispielsweise Nutzungsstatistiken erhoben werden, die Auskunft über Nutzungshäufigkeiten und verwendete Dashboard-Widgets geben. Anhand dieser Informationen können nicht nur häufig verwendete Widgets erkannt und als typisch genutzte Metriken weitergereicht werden, sondern auch Probleme durch nicht-genutzte Widgets oder gar Dashboards ersichtlich werden. Des weiteren sollten regelmäßige Erhebungen der Informationsbedürfnisse stattfinden, um neue Datenquellen, Fragestellungen oder Prozessänderungen innerhalb des Systems einzupflegen.

III.4.3 Fazit der Feldstudie im SSELab

Die Feldstudie innerhalb des SSELabs zeigte die Möglichkeit einer fachlichen Integration von Messsystemen ohne vorhandene Vorkenntnisse oder Durchführungen von Kennzahlmessungen. Aufgrund der Vielzahl von Forschungsprojekten konnten typische Kennzahlen für forschungsnahe Projektleiter erhoben werden.

Die Anforderungsanalyse durch Interviews hat sich als detailreich, jedoch auch zeintensiv herausgestellt. Im Rahmen der intervieweten Projektleiter war dieser noch machbar, bei einer größeren Anzahl an Interview-Partnern empfiehlt sich jedoch die Durchführung der Analyse in Workshops mit mehreren Teilnehmern.

Die Evaluation der Dashboard-Ansichten zeigte, dass Prototyping ein unerlässliches Werkzeug für eine solche Integration ist. Ebenso wurde festgestellt, dass vor allem die Nutzung von Papier-Prototypen für Konfigurationsansichten empfohlen wird.

III.4.4 Ausblick

Innerhalb der Anforderungsanalyse wurde festgestellt, dass einige Projektmanager mehr als ein Projekt zu leiten haben. Diese wünschten ein Multi-Projekt-Dashboard, welches einen allgemeinen Überblick über alle vorhandenen Projekte liefert. Dieses Dashboard wurde innerhalb dieser Arbeit nicht weiter betrachtet, stellt jedoch viele neue Anforderungen an Design und Funktionalität des *EMI*-Dashboard *SCREEN*.

Des Weiteren wurden nur Softwareentwicklungsprojekte betrachtet. Innerhalb des SSE Labs werden jedoch auch weitere Projektarten verwaltet, welche unter Umständen ebenfalls via Kennzahlen-Dashboard gesteuert werden könnten.

Auch die Projektrollen wurden nicht komplett abgedeckt. Neben der hier betrachteten Rolle des Projektmanagers, existiert noch eine weitere Rolle des Entwicklers. Auch dieser kann für seine Aufgaben innerhalb des SSE Labs ein Dashboard zur Steuerung der Produktqualität nutzen. Die Vorlage mit typischen Kennzahlen hierfür ist in einem weiteren Schritt zu erstellen und zu integrieren.

