

Projekthandbuch

Jeremias Eppler, Jochen Morent, Georgi Georgiev

8. Dezember 2013



Inhaltsverzeichnis

1	Projektstart	3
1.1	Sachlicher Kontext	3
1.2	Zeitlicher Kontext	4
1.3	Sozialer Kontext	5
1.4	Projektorganisation	6
1.5	Projektstrukturplan	7
1.6	Arbeitspakete Spezifikation	8
1.7	Projektmeilensteinplan	16
1.8	Projektbalkenplan	17
1.9	Projektfortschrittsbericht	18
1.10	Projekt-Score-Card	20

1 Projektstart

1.1 Sachlicher Kontext

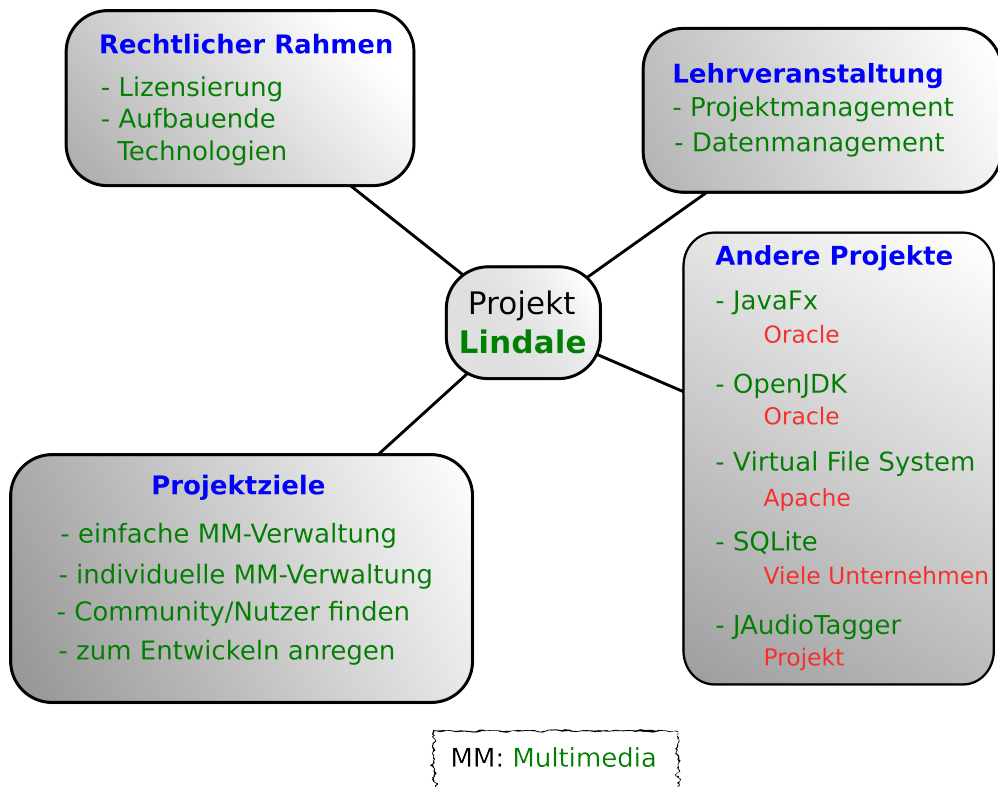


Abbildung 1.1: Sachlicher Kontext

1.2 Zeitlicher Kontext

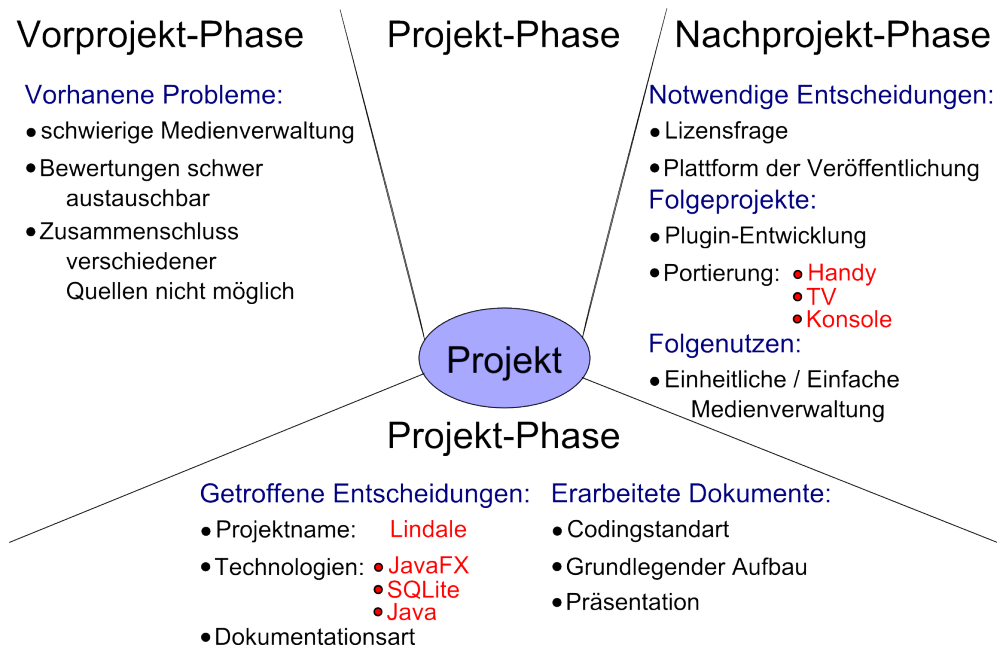


Abbildung 1.2: Zeitlicher Kontext

1.3 Sozialer Kontext

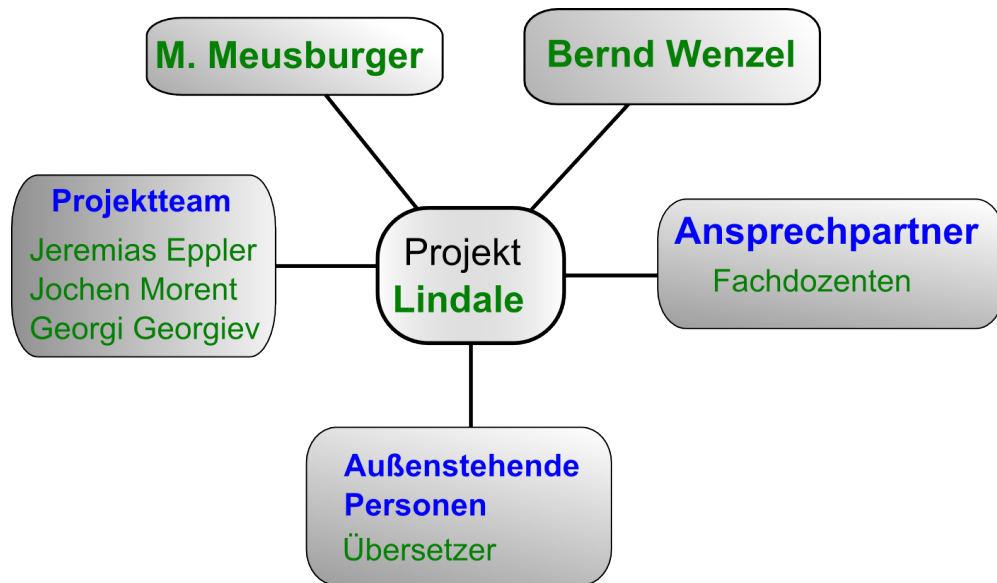


Abbildung 1.3: Sozialer Kontext

1.4 Projektorganisation

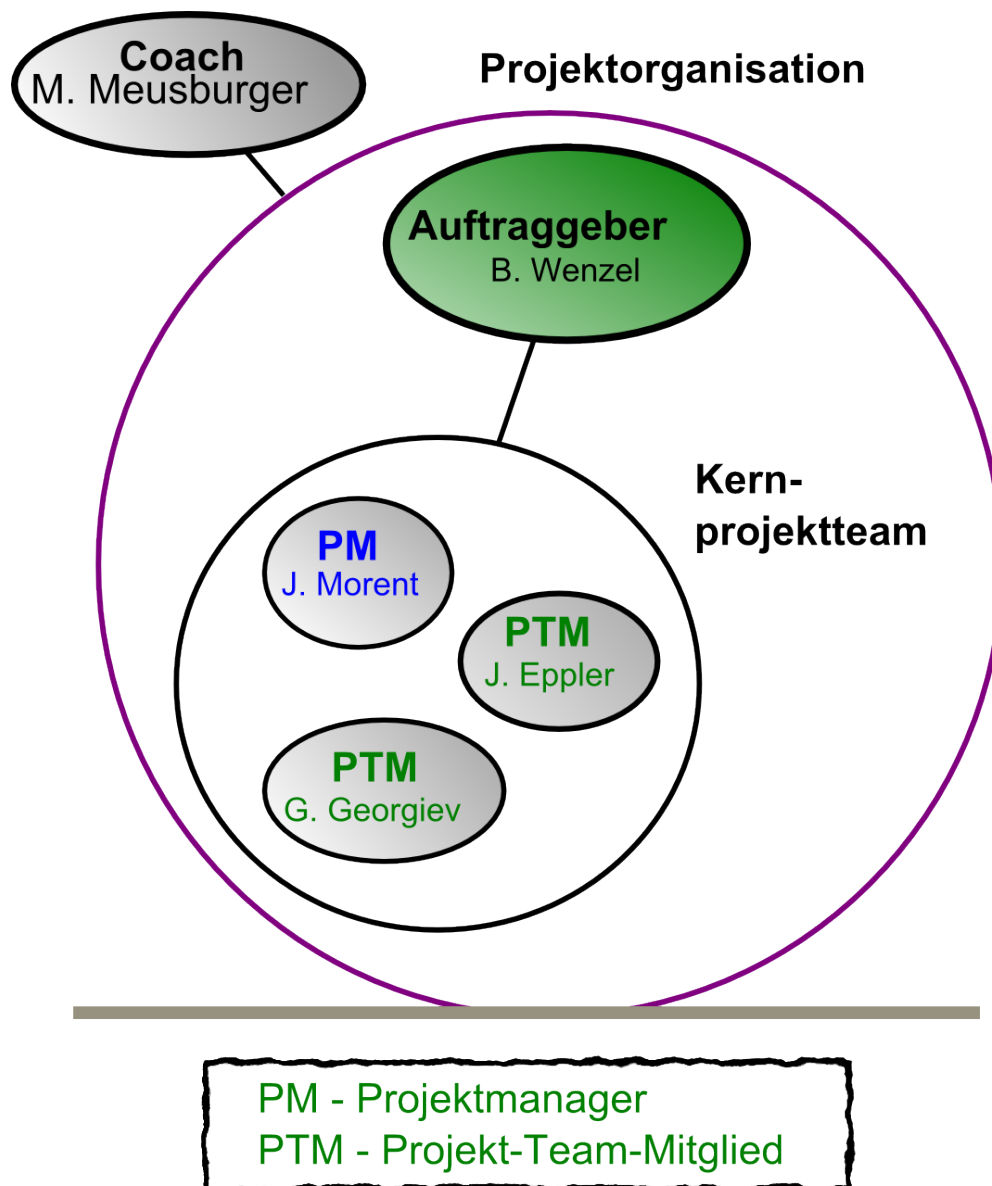


Abbildung 1.4: Projektorganisation

1.5 Projektstrukturplan

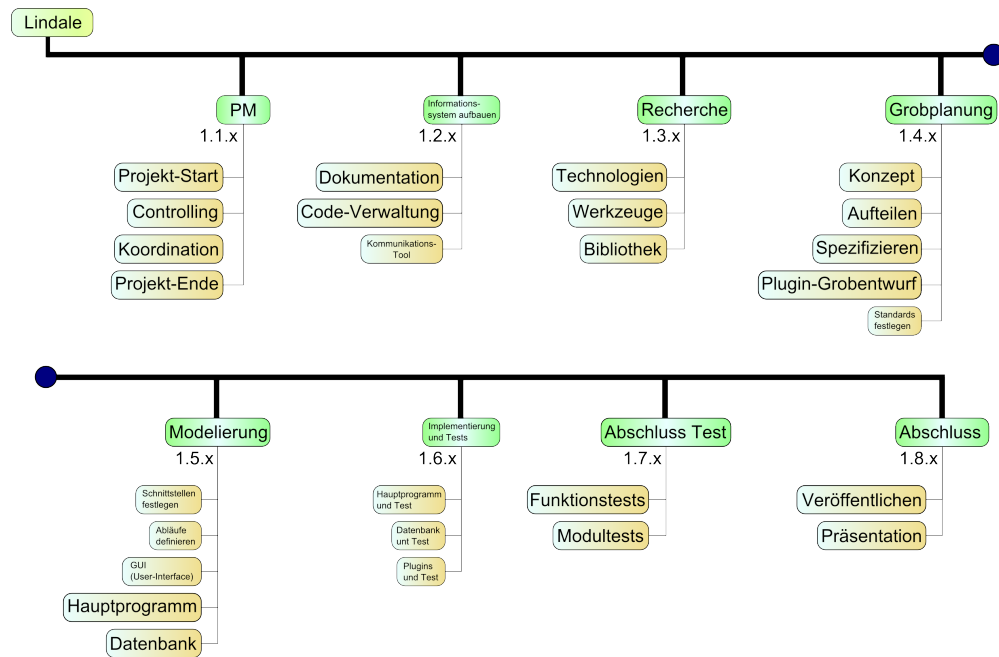


Abbildung 1.5: Projektstrukturplan

1.6 Arbeitspakete Spezifikation

Informationssystem aufbauen

Dokumentation (1.2.1)	
Vorgehensweise / Inhalte	<ul style="list-style-type: none"> • Linksammlung • Projektdokumentation • Protokolle • Dateien, die während des Projekts erstellt werden, sammeln • Recherche Ergebnisse
Ergebnisse	<ul style="list-style-type: none"> • Durchgehende Dokumentation • Einheitlicher Informationsstand
Code-Verwaltung (1.2.2)	
Vorgehensweise / Inhalte	<ul style="list-style-type: none"> • Code Versionsverwaltung (mit GIT) • Comitten / Teamwork sammeln • Recherche Ergebnisse
Ergebnisse	<ul style="list-style-type: none"> • Separat an verschiedenen oder gleichen Codestücken arbeiten • Nicht beeinflussen durch separate Entwicklungsschritte
Kommunikations-Tool (1.2.3)	
Vorgehensweise / Inhalte	<ul style="list-style-type: none"> • Facebook-Gruppe • Facebook-Gruppen-Chat
Ergebnisse	<ul style="list-style-type: none"> • Absprachen, Treffen, Neuigkeiten • Schneller Informationsaustausch

Recherche

Technologien (1.3.1)	
Vorgehensweise / Inhalte	<ul style="list-style-type: none"> • Abschätzen des Implementierungsaufwand • Moderner Technologien Einsatz (FXML, JavaFX)
Ergebnisse	<ul style="list-style-type: none"> • Effiziente Implementierung • Richtiger Einsatz der Technologie
Werkzeuge (1.3.2)	
Vorgehensweise / Inhalte	<ul style="list-style-type: none"> • Ermitteln von nützlichen Tools • Ausprobieren der Werkzeuge • Dokumentieren • Auswahltreffen
Ergebnisse	<ul style="list-style-type: none"> • Erfahrungen sammeln • Geeignete Tools finden
Bibliothek (1.3.3)	
Vorgehensweise / Inhalte	<ul style="list-style-type: none"> • Suchen • Experimentieren • Auswahltreffen
Ergebnisse	<ul style="list-style-type: none"> • Erleichterte Programmierung • Richtiger Einsatz der Bibliotheken

Grobplanung

Aufteilung (1.4.1)	
Vorgehensweise / Inhalte	<ul style="list-style-type: none"> • Aufsplitten der Module <ul style="list-style-type: none"> – Startprogramm – GUI – Plugin-Manager – Datenbank – Plugins
Ergebnisse	<ul style="list-style-type: none"> • Leichtere Verteilung der Aufgaben
Spezifizieren (1.4.2)	
Vorgehensweise / Inhalte	<ul style="list-style-type: none"> • Tests definieren • API / Plugin-API • SQL-Wrapper
Ergebnisse	<ul style="list-style-type: none"> • Einheitliche Schnittstellen
Plugin-Grobentwurf (1.4.3)	
Vorgehensweise / Inhalte	<ul style="list-style-type: none"> • Arten definieren • Start-/Stop-/Instalations-/Deinstalations- Verhalten definieren • Modi (Automatisch/Manuel) • Zugriffe / Zugriffsarten • Bereitstellung
Ergebnisse	<ul style="list-style-type: none"> • Modulare Architektur
Standards festlegen (1.4.4)	
Vorgehensweise / Inhalte	<ul style="list-style-type: none"> • Coding-Standards • Sprach-Standards • Formatierungs-Standards
Ergebnisse	<ul style="list-style-type: none"> • Vereinheitlichung der Entwicklung

Modellierung

Schnittstellen festlegen (1.5.1)	
Vorgehensweise / Inhalte	<ul style="list-style-type: none"> • UML (Klassen-Diagramm) Modellierung • APIs Implementierungs nah definieren <ul style="list-style-type: none"> – Plugin-API – Interne API <ul style="list-style-type: none"> * SQL-Execution * GUI • Testfälle definieren
Ergebnisse	<ul style="list-style-type: none"> • Implementierungsnahes Model • Testdefinitionen
Abläufe definieren (1.5.2)	
Vorgehensweise / Inhalte	<ul style="list-style-type: none"> • UML (Sequenz-Diagramm) • Flussdiagramme • Programmstart • Plugin: Start-/Stop-/Instalations-/Deinstalations-Verhalten • Quellen Hinzufügen • Datenbankzugriffe (Wrapper ↔ SQLite) • Medien laden / abspielen • Laden von Plugins in die GUI • Fehlermeldungen
Ergebnisse	<ul style="list-style-type: none"> • Definierter Programmablauf

GUI (1.5.3)	
Vorgehensweise / Inhalte	<ul style="list-style-type: none"> • Grafische Modellierung • Dialoge Modellieren / Festlegen • Aussehen (Farbe, Steuerelemente, Schnittstellen zum User)
Ergebnisse	<ul style="list-style-type: none"> • Strukturierte und moderne Oberfläche
Hauptprogramm (1.5.4)	
Vorgehensweise / Inhalte	<ul style="list-style-type: none"> • UML (Klassen-Diagramme) • Schnittstellen zu Bibliotheken • Schnittstellen zur Datenbank • Statement-Exekution-Parser
Ergebnisse	<ul style="list-style-type: none"> • Ein Model des Hauptprogrammes
Datenbank (1.5.5)	
Vorgehensweise / Inhalte	<ul style="list-style-type: none"> • E3R-Methode • 5. Normalform • Seminar Wenzel • Test-Transaktionen definieren
Ergebnisse	<ul style="list-style-type: none"> • Datenbankmodel nahe Implementierung • Testdefinitionen

Implementierung und Tests

Hauptprogramm und Test (1.6.1)	
Vorgehensweise / Inhalte	<ul style="list-style-type: none"> • Umsetzen der Klassen-Diagramm (Dummys) • Umsetzung definierter Test (JUnit) • Implementierung der Funktionalität • GUI-Test
Ergebnisse	<ul style="list-style-type: none"> • Lauffähiges Hauptprogramm (nicht mehr grabbeln)
Datenbank und Test (1.6.2)	
Vorgehensweise / Inhalte	<ul style="list-style-type: none"> • Implementieren • Abfragen Tests / Transaktions Tests
Ergebnisse	<ul style="list-style-type: none"> • Super Datenbank
Plugins und Test (1.6.3)	
Vorgehensweise / Inhalte	<ul style="list-style-type: none"> • Implementieren • Testen • Schnittstellen Tests
Ergebnisse	<ul style="list-style-type: none"> • Funktionierende Plugins

Abschluss Test

Funktionstests (1.7.1)	
Vorgehensweise / Inhalte	<ul style="list-style-type: none"> • Alle Funktionalitäten überprüfen • Fehler provozieren
Ergebnisse	<ul style="list-style-type: none"> • Programm kann veröffentlicht werden
Modultests (1.7.2))	
Vorgehensweise / Inhalte	<ul style="list-style-type: none"> • Jedes Modul separatem Test unterzogen • Modulgruppen testen • Teil-Programm-Tests z.B.: Hauptprogramm
Ergebnisse	<ul style="list-style-type: none"> • Funktionalität der Programmteile gewährleistet

Abschluss

Veröffentlichen (1.8.1)	
Vorgehensweise / Inhalte	<ul style="list-style-type: none"> • Lizenzfrage klären • Webpräsenz • Webhoster finden
Ergebnisse	<ul style="list-style-type: none"> • Veröffentlichen
Präsentieren (1.8.2)	
Vorgehensweise / Inhalte	<ul style="list-style-type: none"> • Projekthandbuch • Präsentation vorbereiten / üben / halten
Ergebnisse	<ul style="list-style-type: none"> • Präsentations Note

1.7 Projektmeilensteinplan

Tabelle 1.1: Projektmeilensteinplan

Projektmeilensteinplan			
PSP-Code	Meilenstein	Plantermin	Isttermin
	Projektstart	05.10.2013	
1.3	Start Recherche	05.10.2013	
1.4	Grobplanung	05.10.2013	
1.4.1	Konzeptions beginn	21.11.2013	
1.6	Modellierung Ende	06.12.2013	
1.7.1	Start Hauptprogramm Entwicklung	07.12.2013	
1.7.2	Beginn Plugin-Entwicklung	21.12.2013	
1.8	Start Testen	06.12.2013	
1.9	Projektabschluss	10.01.2013	

1.8 Projektbalkenplan

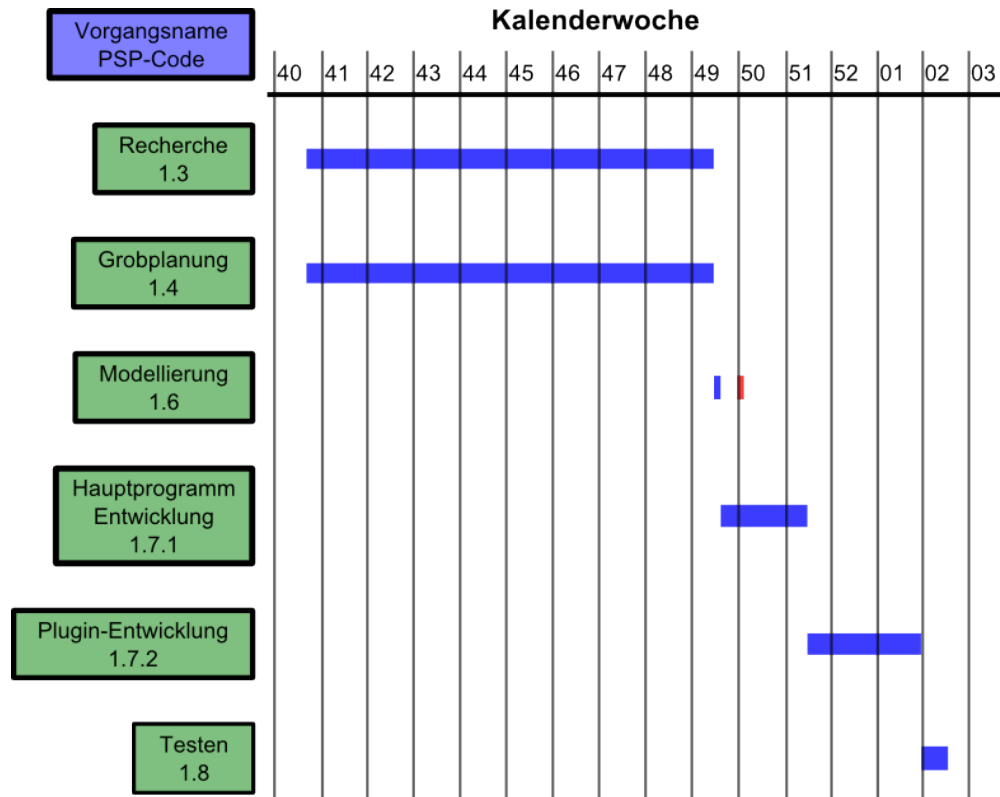


Abbildung 1.6: Projektbalkenplan

1.9 Projektfortschrittsbericht

Lindale Projektfortschrittsbericht				
sehr gut	gut ✓	okay	schlecht	sehr schlecht
Gestamtstatus des Projektes				
<ul style="list-style-type: none"> • Projekt liegt im Zeitrahmen • Hinzugekommenes Know-how durch Datenbankprogrammierungs Lehrveranstaltung und Seminar • Angespante Arbeitssituation aller Teammitglieder • Verteilung von Inhalten des Projekthandbuches → Informations und Datenmisstand innerhalb des Projektes 				
Status: Projektziele und Betrachtungsobjekte				
<ul style="list-style-type: none"> • Zieladaptierung: weglassen von Features wie: <ul style="list-style-type: none"> – Logische und Physikalische Collection – Bewertungssystem 				
Status: Projektleistungsfortschritt				
<ul style="list-style-type: none"> • Arbeitspakete fortschritt: <ul style="list-style-type: none"> 1.2 Erfolgreich umgesetzt 1.3 Recherche für die Hauptprogramm Entwicklung abgeschlossen 1.4.3 Spezifizieren für die Entwicklung des Hauptprogramms abgeschlossen 1.4.5 Standards soweit festgelegt, das entwickelt werden kann 1.5.5 Datenbank Modellierung abgeschlossen 				
Status: Projekttermine				
<ul style="list-style-type: none"> • Verzögerung bei der Entwicklung und Modellierung des Hauptprogramms durch neu hinzugewonnes Wissen 				
Status: Projektumwelt, Beziehungen zu anderen Projekten				

<ul style="list-style-type: none"> • Hinzugekommener Dozent als Experte zwischen Datenbank und Anwendungsprogramm
Status: Maßnahmen
<ul style="list-style-type: none"> • Adaptieren der Nebenziele • Neu ausrichtung des Projektes mithilfe des gewonnenen Informationen • Zusammenführen der Projekthandbuch Informationen • Einpflegen von Informationen in das Informationssystem
Status: Anhang
<ul style="list-style-type: none"> • Projekt Score Card
Version 1.0 Datum 6 Dez, 2013 Ersteller: Lindale-Team

1.10 Projekt-Score-Card

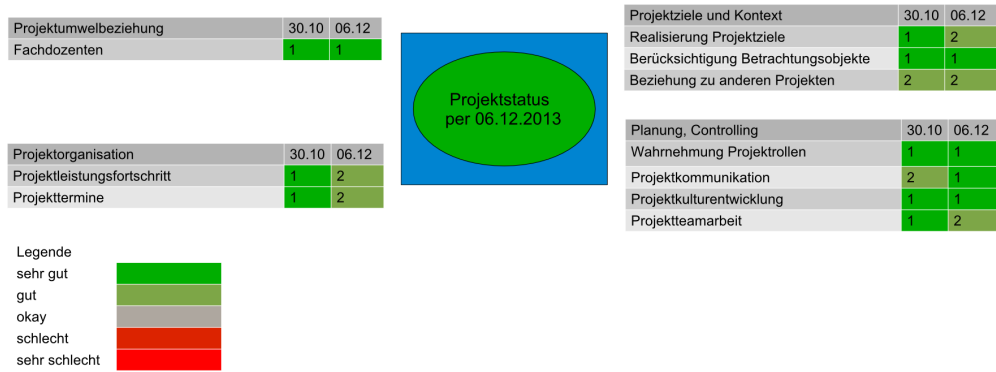


Abbildung 1.7: Projekt-Score-Card