

TU DRESDEN

SOFTWAREMANAGEMENT

PROJEKTARBEIT

Indoor-Lokalisierungs-Framework

ILF-Prototyp

Autoren:

Wieland STRAUSS

Martin WELCKER

Betreuer:

Dr.-Ing. Karsten WENDT

18. Juli 2018



Inhaltsverzeichnis

1	Projektdefinition	2
1.1	Vision	2
1.1.1	Technische Umsetzung	2
1.1.2	Anwendungsfall	3
1.2	Stakeholder	4
1.2.1	Externe Stakeholder	4
1.2.2	Interne Stakeholder	4
1.2.3	Sonstige	5
1.3	Projektziele	5
2	Planung	6
2.1	Projektumfang	6
2.1.1	Projektstrukturplan	6
2.1.2	Rollen	6
2.1.3	Ressourcen	7
2.1.4	Allgemeine Risikoanalyse	7
2.1.5	Arbeitspakete und Ressourcenverteilung	7
2.1.6	Projektmodell	7
2.2	Zeitplan	8
2.2.1	Meilensteine	8
2.2.2	Netzplan	8
2.3	Kostenplanung	8
2.4	Controlling	8
2.5	Risikomanagement	9
2.6	Kommunikationsplan	9
2.7	Qualitätssicherung	10
3	Anhang	10

1 Projektdefinition

1.1 Vision

Die Zielstellung des Projektes ist es einen Prototypen eines Frameworks für Indoor-Lokalisierungslösungen zu erstellen.

Das Framework soll dazu die Verknüpfung verschiedenster Übertragungstechnologien (z.B. UWB, Bluetooth,..) ermöglichen. Entwicklern soll es ermöglicht werden, auf einfachen Weg komplexe Indoor-Lokalisierungs-Systeme zu realisieren. Das Framework unterstützt dazu eine Vielzahl von möglichen Schnittstellen. Dazu gehören Server- und Geräteschnittstellen, um Datenhaltung und Datenpräsentation zu ermöglichen.

Dieses Projekt hat die Zielstellung einen minimalen Prototypen des beschriebenen Frameworks zu entwickeln. Dieser beschränkt sich auf die Implementierung von zwei Kommunikationstechnologien. Es soll dadurch die Umsetzbarkeit des Frameworks evaluiert werden. Ein relevanter Anwendungsfall soll die Mächtigkeit des Prototypen demonstrieren. Die Menge der zu implementierenden Schnittstellen ist dabei von diesem Anwendungsfall abhängig. Nach der Entwicklung des Prototypen soll dieser in einer Live-Demo potentiellen Endkunden und Interessenten vorgestellt werden. Am Ende erhält der Auftraggeber eine Evaluation über die Umsetzbarkeit des Prototypen. Dazu werden parallel zur Entwicklung Dokumentationen geführt.

1.1.1 Technische Umsetzung

Sensorik Um einen möglichst repräsentativen Anwendungsfall zu entwickeln ist es notwendig, die möglichen Technologien für Indoor-Lokalisierung zu betrachten.

Die Wahl der beiden zu implementierenden Technologien soll dabei auf diejenigen fallen, die maximales Potential versprechen und sich von voneinander stark unterscheiden. Diese Entscheidung fällt hier einerseits auf UWB, da es eine sehr populäre Technologie ist, die besonders in der Industrie großen Andrang gefunden hat. Sie ist zwar kostenintensiv, aber vereint dafür hohe Reichweite und Genauigkeit. Die hohe Datenrate bis zu 1320 Mbit/s sorgt dafür, dass UWB auch für zukünftige Anwendungen eine gute technische Grundlage bietet.

Die Wahl der zweiten Technologie fällt auf RFID, die im Gegensatz zu UWB eher das punktuelle Tracking unterstützt. Im Vergleich ist sie wesentlich günstiger und

damit für eine andere Gruppe von Anwendungen interessant.

Eine tabellarische Übersicht der Technologien ist der Tabelle 1 auf Seite 10 im Anhang zu entnehmen.

Daten Zur Datenspeicherung wird eine **MySQL** Datenbank genutzt. Die klassische relationale Datenbank ist für unseren Anwendungsfall schnell genug. Sie ist zuverlässig und populär, so dass viele Lösungen für bestimmte Problemstellungen bestehen. Zudem besitzt sie viele praktische Features.

Um flexibel auf Kundenwünsche eingehen zu können wird eine Datenbankschnittstelle verwendet. Die Wahl fällt dabei auf **ODBC** (Open Database Connectivity).

Nutzergeräte Als Nutzergeräte werden Smartphones verwendet. Sie bilden die Schnittstelle zwischen System und Nutzer. Da die Smartphones die ausgewählten Sensortechnologien (RFID, UWB) unterstützen müssen fällt die Wahl auf **BeSpon-Phones**. Diese bieten ausreichend Leistung und dienen zum Testen des Frameworks, als Demonstrationsobjekt in der Präsentation und der Live-Demo.

1.1.2 Anwendungsfall

In der Präsentation soll das Framework anhand eines Museums-Szenarios demonstriert werden. Dazu agieren die potentiellen Endkunden als Museumsbesucher. Sie erhalten ein Smartphone, welches sowohl mit Sensoren für UWB und RFID ausgestattet ist. Für die Besucher ergibt sich die Möglichkeit mit dem Gerät und einer vorinstallierten Demo-Anwendung den Museumsraum zu erkunden. Verschiedene Präparate sind mit passiven **RFID**-Chips versehen. Diese werden automatisch von der Anwendung erfasst, damit diese zugehörige multimediale Inhalte präsentieren kann.

Die **UWB**-Tags in den Smartphones ermöglichen die Lokalisierung im Raum. Die Ortsinformationen werden über eine Schnittstelle zu einem Server übertragen, der diese in einer Datenbank speichert. In der Live-Demo soll damit eine virtuelle Karte mit den aktuellen Besucherströmen angezeigt werden (Heatmap).

1.2 Stakeholder

1.2.1 Externe Stakeholder

Auftrag- und Kapitalgeber: Der Auftraggeber wünscht eine zuverlässig funktionierenden Prototypen in Rahmen der möglichst geringen bewilligten finanziellen Mittel und den vereinbarten Zeitraum. Dabei steht nicht die vollständige Entwicklung des Frameworks im Vordergrund, sondern die Evaluation der Möglichkeiten von Technologien.

Potentielle Endkunden: Die potentiellen Endkunden sind die Zielgruppe unserer Entwicklung und sollen in der Präsentation des Prototypen vom Kauf des Frameworks überzeugt werden. Der Prototyp und die Vorstellung des Frameworks sollte mögliche Anwendungsszenarien abdecken.

1.2.2 Interne Stakeholder

Management: Das Management möchte alle externen Stakeholder möglichst zufriedenstellen, aber zugleich nur beschränkt Ressourcen und Mitarbeiter zur Verfügung stellen. Der Kapitalgeber soll weiterhin als beständiger Investor gewonnen werden, der nach dem Projekt die langfristige Weiterentwicklung des Prototypen zu einem großen Framework mit unterstützt.

Die Qualität des Produktes sollte eines Prototypen/Demoversion angemessen sein. (Eine Demonstration sollte fehlerfrei Möglich sein, mehr nicht.) Das Projek hat den gleichen Stellenwert wie jeder andere Auftrag.

Mitarbeiter: Die Mitarbeiter haben die Möglichkeit neue Technologien auszuprobieren und neue Lösungen zu entwickeln. Während ihres Mitwirkens an dem Projekt sind sie von anderen Aufgaben befreit, so dass kein Interessenkonflikt entsteht.

1.2.3 Sonstige

Standards Bei der Implementierung orientieren wir uns an den technischen Standards.

- UWB
IEEE 802.15.3a
IEEE 802.15.4a
- RFID
ISO 18000-1 ff

1.3 Projektziele

Prototyp: Zielstellung für den Prototypen ist das vollständige umsetzen der folgenden Funktionalitäten:

- Zwei unterstützte Sensortechnologien (UWB, RFID)
- Schnittstellen zu Geräten und Sensoren in Form von Geräte kapseln, die modulares Erweitern ermöglichen
- Schnittstellen zu Datenbankserver
- Implementierung des Anwendungskernes
- Evaluation über die Realisierbarkeit des Prototypen (Schwierigkeiten / Verbesserungsvorschläge)

Nicht-Ziele:

- Keine Entwicklung des vollständige Frameworks
- Keine graphische Benutzeroberfläche

Präsentation:

- Ausarbeitung eines Vortrages mit Livedemonstration über die Rahmenbedingungen des Frameworks und mögliche Anwendungsfälle
- Entwickeln einer Demonstrationsanwendung 'Museumsbesuch', in der die Endkunden das Framework selber testen können

2 Planung

2.1 Projektumfang

Nachfolgend werden die Arbeitspakete anhand des Projektstrukturplanes entwickelt und benötigte Ressourcen aufgeteilt um den zeitlichen Projektrahmen festzulegen.

2.1.1 Projektstrukturplan

Die jeweiligen Arbeitspakete leiten sich aus dem Projektstrukturplan ab (Abbildung 1 im Anhang). Sie entsprechen dabei den Endknoten des Diagrammes.

2.1.2 Rollen

Projektleiter Der Projektleiter ist für das gesamte Projekt verantwortlich und repräsentiert sein Team nach Außen. Er ist dem Management unterstellt. Seine Aufgabe ist die Koordinierung des Projektes und der Mitarbeiter. Zusätzlich ist er für die organisatorischen Belange zuständig und arbeitet bei einigen Aufgaben mit, wie beispielsweise den Entwurf.

Entwickler Die Entwickler sind für den Entwurf und die Implementierung des Frameworks zuständig.

Full-Stack-Entwickler Der Full-Stack-Entwickler ist als Allrounder vor allem für die Entwicklung der Demoanwendung zuständig.

Software-Ingenieur Der Software-Ingenieur entwickelt und plant das Framework. Dabei liegt seine Hauptaufgabe im Entwurf.

Datenbankadministrator Der Datenbankadministrator ist für die Bereitstellung einer Datenbank und einer Datenbankschnittstelle zuständig. Er unterstützt das Entwicklerteam bei der Funktionalität der Datenformate und Speicherung.

UI-Designer Der UI-Designer kümmert sich um die Oberflächen des Frameworks und der Demoanwendung.

Vortragender Der Vortragende ist für die Präsentation und Moderation bei der Demo-Anwendung Museumsbesuch sowie deren Erarbeitung zuständig.

2.1.3 Ressourcen

Die Auflistung der benötigten Ressourcen mit den entstehenden Kosten ist der Tabelle 2 auf Seite 12 im Anhang zu entnehmen.

2.1.4 Allgemeine Risikoanalyse

In Tabelle 3 (Anhang) ist eine Zusammenstellung der allgemeinen Risiken, die phasen- und vorgangs-unabhängig auftreten können zu sehen. Das Erreichen der Projektziele im definierten Zeitraum ist möglich sofern keines der Risiken eintritt.

2.1.5 Arbeitspakete und Ressourcenverteilung

In Ergänzung zur Allgemeinen Risikoanalyse, werden in Tabelle 4 ebenfalls Risiken in Form von Voraussetzungen formuliert. Sind diese im Projektverlauf erfüllt, können die einzelnen Arbeitspakete im geplanten Rahmen realisiert werden.

Für die Arbeitspakete siehe Tabelle 4 auf Seite 13 (Anhang)

2.1.6 Projektmodell

Der Projektstrukturplan gliedert die Arbeitspakete in beiden Bereiche „Prototyp“ und „Präsentation“. Die Entwicklung des Prototypen läuft dabei linear ab. Das bedeutet, dass Design, Implementierung und Test voneinander getrennt werden. Innerhalb der einzelnen Arbeitspakete soll jedoch **Scrum** und **Kanban** zum Einsatz kommen. Täglich sind dabei Scrum-Meetings geplant, die zur Kontrolle der Arbeitspakete dienen. Ein Kanban-Board soll die täglichen Aufgaben unterstützen und als Art Ticketsystem genutzt werden.

Die Entwicklung der Demoanwendung für das Präsentations-Szenario erfolgt nach dem Prinzip des **Extreme Programming**. Dieses Verfahren ist vorteilhaft, da die Anwendung nur für die Präsentation kreiert wird und hier das MVP im Vordergrund steht.

2.2 Zeitplan

2.2.1 Meilensteine

Die Projektmeilensteine erschließen sich aus dem Projektstrukturplan. Eine Etappe ist dabei mit dem Ende der Implementierung des Prototypen erreicht. Eine zweite kürzere Etappe endet mit der Fertigstellung der Präsentation und der Demoapplikation.

2.2.2 Netzplan

Netzplan siehe Abbildung 3 auf Seite 14 (Anhang)

Zusätzlich ist ein Gantt-Diagramm in Abbildung 4 auf Seite 15 (Anhang) zu sehen.

2.3 Kostenplanung

Anhand der Aufteilung von Ressourcen auf die verschiedenen Arbeitsschritte ermitteln sich folgende Projektkosten. Dabei wird davon ausgegangen, dass die Arbeitspakete, die nicht auf dem kritischen Pfad liegen, ohne Nutzung der Pufferzeit realisiert werden.

Die Gesamtkosten belaufen sich auf **233.800,00 €**. Eine Übersicht der im Projektverlauf kumulierten Kosten ist der Abbildung 2 im Anhang auf Seite 11 zu entnehmen.

2.4 Controlling

Die Projektgruppe trifft sich täglich um den aktuellen Stand festzuhalten, Probleme sowie deren Lösungsvorschläge zu besprechen und weitere Informationen auszutauschen. Des Weiteren wird vom Projektleiter die Möglichkeit genutzt, um eine Leistungsrückmeldung zu geben, mögliche Problemstellungen und Konflikte früh wahrnehmen zu können und auch neuen Input zu geben. Dabei wird das SCRUM-Prinzip genutzt.

Die Leistungsübersicht des Einzelnen wird über die Aufgabenorganisation realisiert. Vorab wurden wöchentliche Meilensteine definiert. Innerhalb dieser Meilensteine erfolgt die Aufgabenvergabe über ein agiles Ticket-Systeme nach Kanban.

Ein Ticket beinhaltet eine kleinste Teilaufgabe und kann von den Mitarbeitern selbst gewählt oder vom Projektleiter zugewiesen werden.

Dadurch ist eine kontinuierliche Übersicht des Arbeitsvorschlusses und der Leistung, sowie der Unterschied zwischen geplanten und tatsächlichen Zeitablauf sichtbar.

Zur Versionsverwaltung wird Git verwendet.

2.5 Risikomanagement

Arbeitsphasen im Aufwand und Zeit tendenziell eher pessimistisch geschätzt. Durch das Ticketsystem können - soweit möglich - Aufgaben vorgezogen werden. Wenn ein Meilenstein nicht zum geplanten Zeitpunkt erreicht wurde fallen die übrigen Tickets an den Nächsten, so dass der Arbeitsaufwand gut abzuschätzen bleibt.

Kleinere Risiken werden vom Projektleiter behandelt, größere mit der gesamten Projektgruppe (gegebenenfalls in einem Sondermeeting). Im Budget ist ein Teil für unvorhersehbare Ereignisse wie technische Ausfälle oder notwendige Überstunden vorgesehen. Durch Vorüberlegungen und laufende Prozessüberwachung sollen Risiken und deren Folgen minimiert werden. Das Nutzen von Unternehmensstandards soll die Vorüberlegungen und die Abhandlung von Problemen erleichtern.

Das Risikomanagement wird somit auf Stufe 2 betrieben. Auf dieser Stufe wird nicht erst mit Krisenmanagement auf zum Problem gewordene Risiken reagiert, sondern auf den Risikoeintritt an sich mittels der Vorüberlegungen.

2.6 Kommunikationsplan

Die Entwickler bilden mit dem Vortragenden ein Team und sollten direkte Fragen untereinander klären. Dazu steht ihnen der direkte Kontakt, E-Mails und die Plattform Slack zur Verfügung. Bei Uneinigkeiten oder Fragen, die in höherer Entscheidungsgewalt liegen, ist der Projektleiter zu konsultieren. Zusätzlich bietet das tägliche Meeting Plattform für diversen Austausch.

2.7 Qualitätssicherung

Die Entwickler sind angewiesen beim Schreiben des Frameworks eine Dokumentation zu pflegen, damit Fehler schneller gefunden werden, Schnittstellen einfacher zu implementieren sind und die Entwickler besser an die Arbeit anderer anknüpfen können. Zusätzlich soll dies die Mögliche Weiterentwicklung vom Prototypen zum vollwertigen Framework erleichtern.

Tests werden nur geringfügig geschrieben. Sie sollen nur die grundsätzlichen, elementaren Funktionen abdecken. Der Prototyp sollte zuverlässig funktionieren, aber muss noch nicht alle Funktionalitäten vollständig und sicher realisiert haben.

3 Anhang

Tabelle 1: Technologien

Technologie	Reichweite	Genauigkeit	Bemerkung
WLAN	••••	•	Schwierigkeiten bei Änderung der Raumstruktur
Bluetooth	••	••	Hohe Portabilität und Skalierbarkeit
RFID	•	••••	Einsatz bei Ortung von vielen Einzelobjekten (z.B. Logistik)
UWB	••••	•••	Hohe Kosten für Ausrüstung Großes Potenzial im industriellen Sektor

Mehr Punkte stehen für bessere Eigenschaften

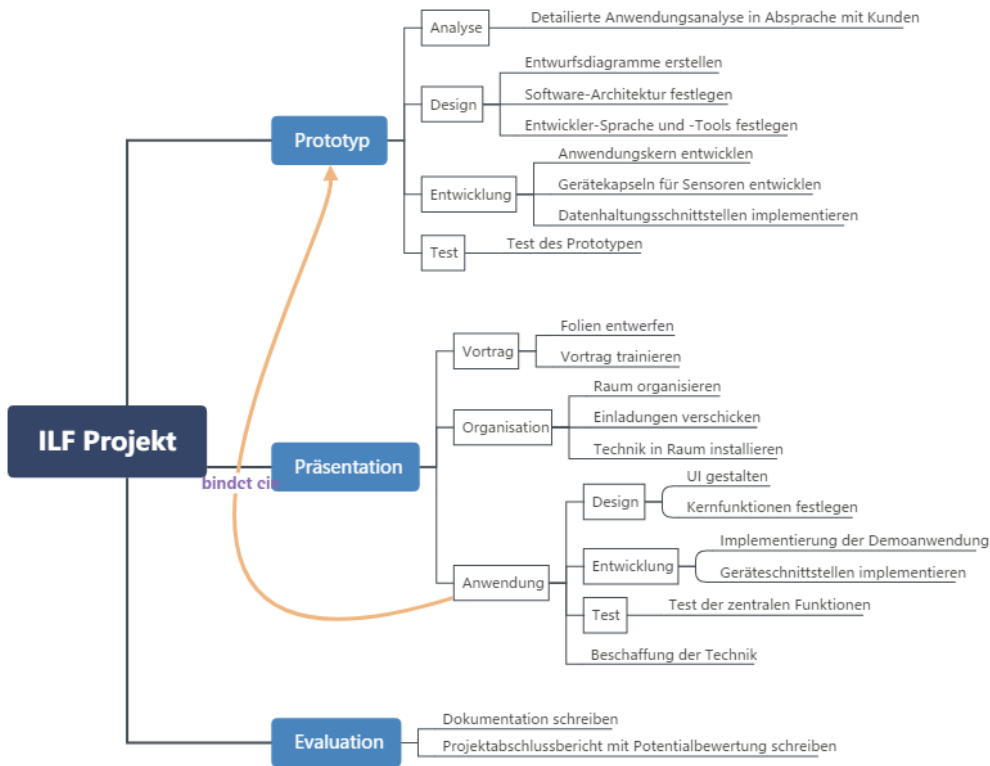


Abbildung 1: Projektstrukturplan nach produktorientierter Gliederung

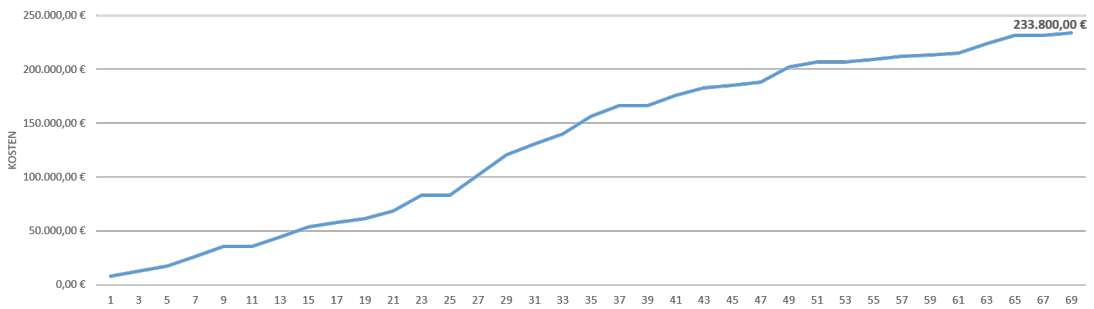


Abbildung 2: Kumulierte Kosten im Projektverlauf (Projektwochen)

Tabelle 2: Ressourcen

Name	Studenlohn	Kosten	Arbeit
Projektleiter	250,00 €/Std	61.833,33 €	247,33 Std.
Software-Ingenieur	180,00 €/Std	30.240,00 €	168 Std.
Full-Stack-Entwickler	170,00 €/Std	28.106,67 €	165,33 Std.
Datenbank-Administrator	150,00 €/Std	17.600,00 €	117,33 Std.
SW-Entwickler 1	150,00 €/Std	16.800,00 €	112 Std.
SW-Entwickler 2	150,00 €/Std	16.800,00 €	112 Std.
SW-Entwickler 3	150,00 €/Std	16.800,00 €	112 Std.
SW-Entwickler 5	150,00 €/Std	16.800,00 €	112 Std.
SW-Entwickler 4	150,00 €/Std	10.800,00 €	72 Std.
Tester	150,00 €/Std	9.600,00 €	64 Std.
UI-Designer	100,00 €/Std	1.600,00 €	16 Std.
Vortragender	80,00 €/Std	1.520,00 €	19 Std.
Entwicklerlizenzen	-	1.200,00 €	1
BeSpoon-Phone	-	1.000,00 €	1
Vorstellungsraum (Museum)	-	500,00 €	1
UWB-Tags	-	2.500,00 €	1
RFID-Technik	-	100,00 €	1

Tabelle 3: Allgemeine Risiken

Risiko	Folgen
Ausfall eines Entwicklers	Krankheit oder andere unabsehbare Ausfälle von Entwicklern resultieren, sofern nicht kompensierbar in einer Verlängerung der Projektdauer. Urlaubstage werden in der Planung berücksichtigt.
Zusätzliche Anforderungen des AG	Neue Idee des AG können nach Absprache in das Projekt einfließen. Diese führen jedoch, je nach Umfang zu einer Verspätung der Deadline und einem Anstieg der Projektkosten.

Tabelle 4: Arbeitspakete

Nr.	Vorgangname	Dauer	Vorgänger	Ressourcennamen	Risiken
1	<u>Prototyp</u>	32 Tage		<u>Entwicklerlizenzen[1]</u>	
2	Design	14 Tage			
3	Detaillierte Anwendungsanalyse in Absprache mit Kunden	2 Tage		Projektleiter;Software-Ingenieur	
4	Software Architektur festlegen	8 Tage	3	SW-Entwickler 1; SW-Entwickler 2;Full-Stack-Entwickler;Projektleiter	Es werden nur bis zu zwei Konzepte für die Software-Architektur vorgestellt.
5	Entwurfsdiagramme erstellen	2 Tage	4	SW-Entwickler 1; SW-Entwickler 2;Full-Stack-Entwickler;Projektleiter	
6	Entwickler-Sprache und -Tools festlegen	2 Tage	5	SW-Entwickler 1; SW-Entwickler 2;Full-Stack-Entwickler;Projektleiter	
7	Implementierung	14 Tage	2		
8	Anwendungskern entwickeln	7 Tage	6	SW-Entwickler 1; SW-Entwickler 2; SW-Entwickler 3;SW-Entwickler 4;SW-Entwickler 5;Projektleiter	Die geplante Software-Struktur kann wie gedacht umgesetzt
9	Geräte kapseln für Sensoren entwickeln	7 Tage	8	SW-Entwickler 1; SW-Entwickler 2; SW-Entwickler 3	Die verwendete Sensor-Technik wurde bereits geliefert.
10	Datenhaltungsschnittstelle implementieren	2 Tage	8	SW-Entwickler 4;SW-Entwickler 5;Projektleiter	
11	Test	4 Tage	7		
12	Testen des Prototypen	4 Tage	10;9	Tester;Software-Ingenieur	Die Qualitätsanforderungen an den Prototypen ändern sich nicht im Projektverlauf.
13	Prototyp fertiggestellt	0 Tage	1;12		
14	Evaluation	12 Tage			
15	Dokumentation schreiben	2 Tage	12	Full-Stack-Entwickler;Projektleiter;Software-Ingenieur;SW-Entwickler 1	
16	Projektabschlussbericht mit Potentialbewertung schreiben	2 Tage	15;28	Projektleiter;Software-Ingenieur;SW-Entwickler 1;Tester	Es wird ein Museum gefunden, dass für eine Präsentation einen Raum zur Verfügung stellt. Alternativ müssen Präparate für einen eigenen Vorführraum vorbereitet werden.
17	<u>Präsentation</u>	23,38 Tage			
18	Organisation	23 Tage			
19	Raum organisieren	2 Tage	3	Projektleiter	Der Raum beeinflusst das Signal nicht signifikant.
20	Einladungen versenden	2 Tage	19	Projektleiter	
21	Technik im Raum installieren	2 Tage	22;19;28;31	Full-Stack-Entwickler;Vortragender	
22	<u>Demoanwendung</u>	12 Tage	13		
23	Design	3 Tage			
24	Kernfunktionen der Demo festlegen	1 Tag	13	Full-Stack-Entwickler;Projektleiter	
25	UI gestalten	2 Tage	1;24	UI-Designer	Die UI wird nach spätestens zwei unterschiedlichen Vorschlägen akzeptiert.
26	Entwicklung	7 Tage			
27	Geräteschnittstellen implementieren	3 Tage	25;31	Full-Stack-Entwickler	
28	Implementierung der Demoanwendung	4 Tage	27	Full-Stack-Entwickler	
29	Test	2 Tage			
30	Testen der zentralen Funktionen	2 Tage	28	Full-Stack-Entwickler	
31	Beschaffung der Technik	2 Std.	24	Projektleiter	Es treten keine Lieferschwierigkeiten auf.
32	Vortrag	2,38 Tage	22		
33	Folien entwerfen	8 Std.	24;20	Vortragender	
34	Vortrag üben	3 Std.	33	Vortragender	
35	Präsentation und Projektabschluss	0 Tage	17;16;34;30;21		

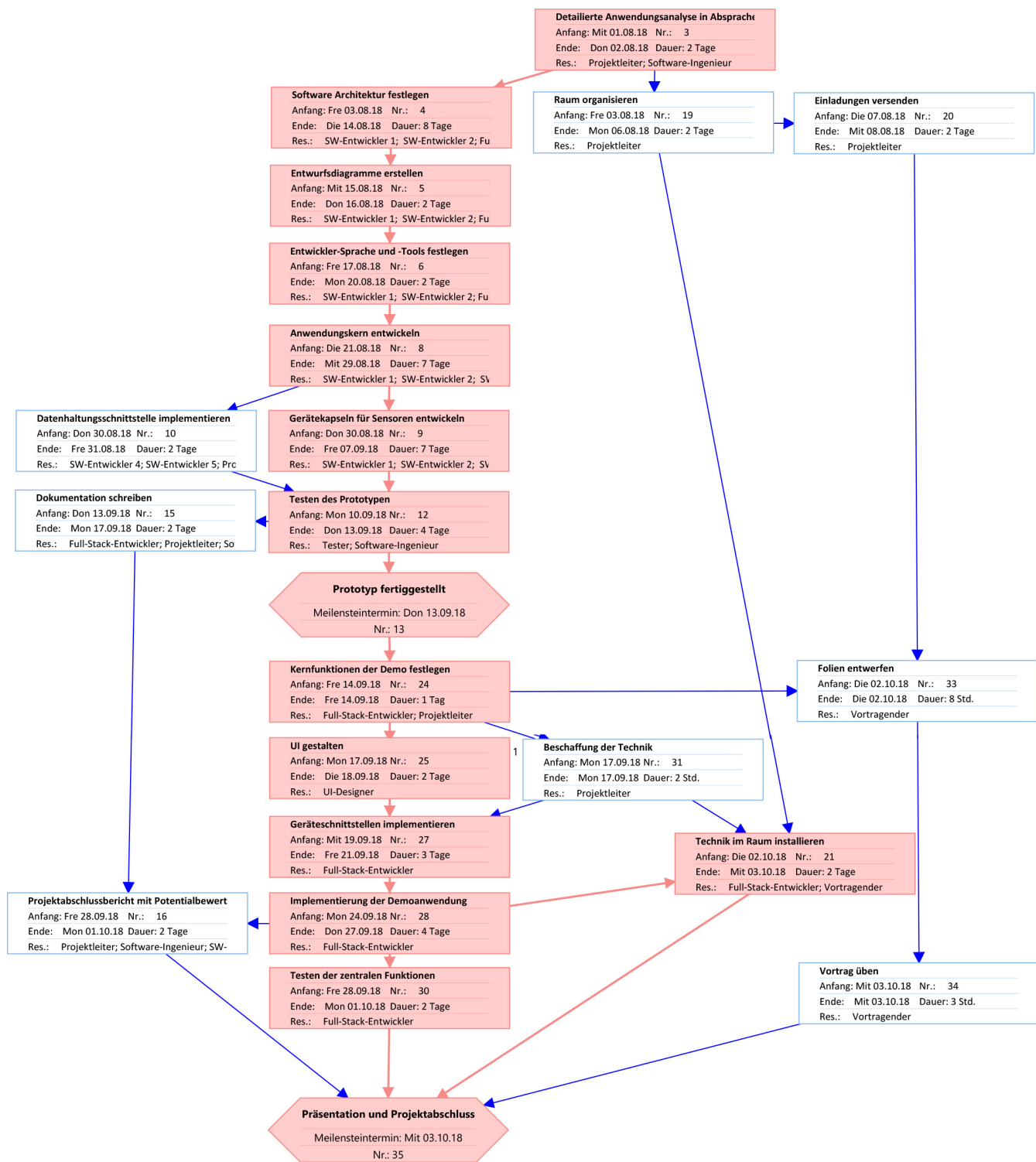


Abbildung 3: Netzplan

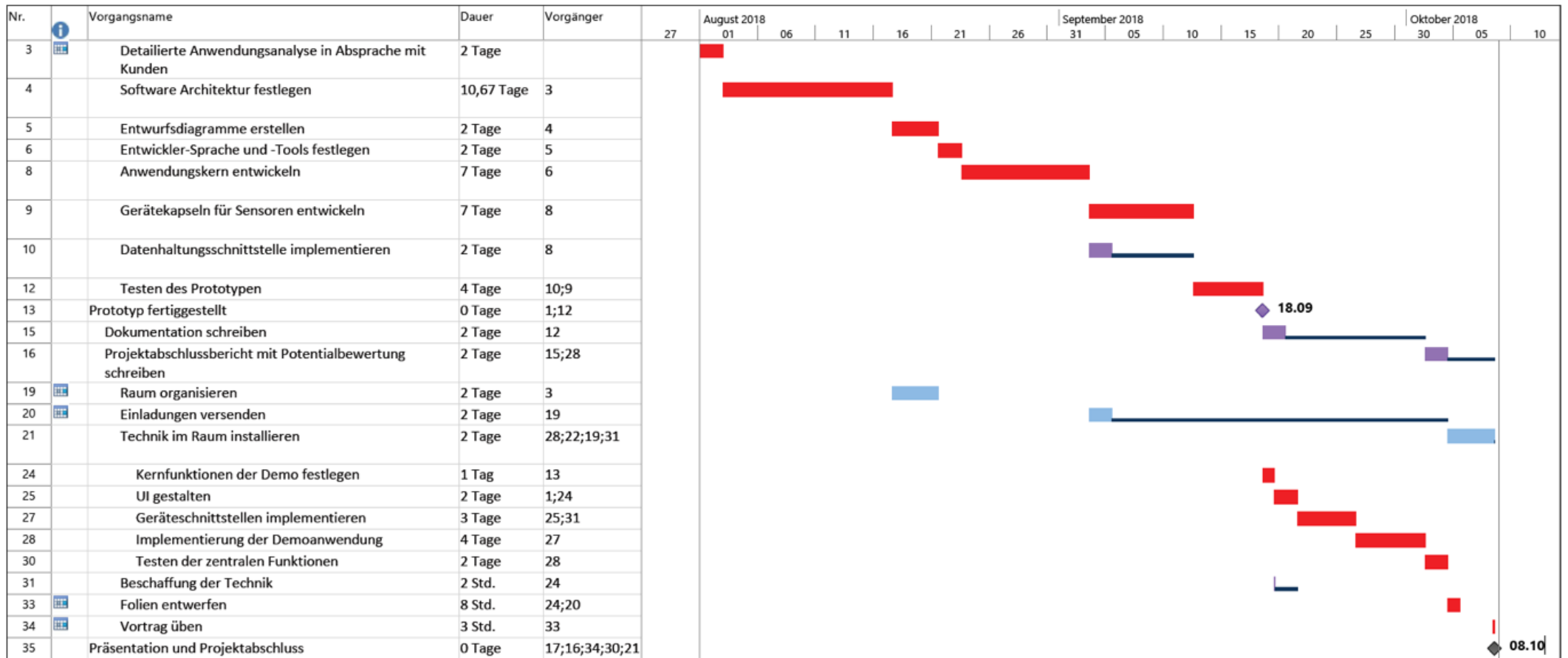


Abbildung 4: Gantt Diagramm