

Kapitel Zusammenfassung ÜB 5 || 18.11.2023

Kapitel 22.1-Kapitel 23.3-Kapitel 23.5

Kapitel 22 Vorblick

- **Ziel:** Überblick über das Management von Softwareprojekten & über zwei wichtige Managementaktivitäten (Risiko- und Personalmanagement)
 - **Kenntnisse nach dem Lesen:**
 - Hauptaufgaben eines Softwareprojektmanagers kennen
 - Verständnis von Risikomanagement und Kenntnis der Risiken in Softwareprojekten.
 - Verständnis der Faktoren, die die Personalmotivation beeinflussen, für Softwareprojektmanager.
 - Kenntnis der Schlüsselfaktoren für Teamarbeit, einschließlich Teamzusammensetzung, -organisation und -kommunikation.
 - **Kriterien für erfolgreiches Projektmanagement:**
 - Software fristgerecht an den Kunden auszuliefern
 - Gesamtkosten im vorgegebenen Rahmen zu halten
 - Software die den Erwartungen des Kunden entspricht
 - Zufriedenes und gut funktionierendes Entwicklerteam aufzubauen
-

Kapitel 22.1 Risikomanagement

Projektmanager müssen Risiken voraussehen und Maßnahmen zur Vermeidung von Auswirkungen auf Zeitplan und Softwarequalität ergreifen.

- **Arten des Risikos:**
 - **Projektrisiken:**
 - Wirken sich auf Projektzeitplan oder die Ressourcen aus
 - *Beispiel:* Verlust eines erfahrenen Systemarchitekten → Diesen zu ersetzen kann lange dauern und somit die Entwicklung über den geplanten Zeitlauf verzögern
 - **Produktrisiken:**
 - Wirken sich auf die Qualität oder die Leistung der entwickelten Software aus
 - *Beispiel:* käuflich erworbene Komponente, die nicht die erwartete Leistung bringt → kann die Gesamtleistung des Systems beeinträchtigen
 - **Geschäftsrisiken:**
 - Risiken welches das Unternehmen trägt die das Projekt entwickelt
 - *Beispiel:* die Einführung eines Konkurrenzprodukts durch einen Mitbewerber, die die Verkaufsprognosen bestehender Softwareprodukte beeinflussen kann.
 - **Risikokategorien können sich überschneiden**

- Der Verlust eines erfahrenen Programmierers kann ein **Projektrisiko (Zeitplanverzögerung)**, ein **Produktrisiko (Programmierfehler durch weniger Erfahrung)** und ein **Geschäftsrisiko (Ruf und Erfahrung beeinflussen die Auftragsakquisition)** darstellen. Bei großen Projekten sollten Risikoanalysenergebnisse und Folgenabschätzungen in einem Risikoregister festgehalten werden.

Einige universelle Risiken:

Risiko	Art des Risikos	Beschreibung
Personalveränderung	Projekt	Erfahrenes Personal verlässt das Projekt vor seiner Fertigstellung.
Managementwechsel	Projekt	Es gibt einen Wechsel zu einem neuen Management, das neue Prioritäten setzt.
Nichtverfügbarkeit von Hardware	Projekt	Für das Projekt unverzichtbare Hardware wird nicht pünktlich geliefert.
Veränderung der Anforderungen	Projekt und Produkt	Die Anzahl der Änderungen bei den Anforderungen ist größer als erwartet.
Verzögerungen in der Spezifikation	Projekt und Produkt	Die Spezifikationen wichtiger Schnittstellen sind nicht pünktlich verfügbar.
Unterschätzen des Umfangs	Projekt und Produkt	Der Umfang des Projekts wurde unterschätzt.
Unzureichende Leistung der Softwarewerkzeuge	Produkt	Softwarewerkzeuge, die das Projekt unterstützen, bringen nicht die erwartete Leistung.
Technologieveränderung	Geschäft	Die dem System zugrunde liegende Technologie wird durch eine neue Technologie verdrängt.
Produktkonkurrenz	Geschäft	Ein Konkurrenzprodukt wird auf den Markt gebracht, bevor das System fertiggestellt ist.

Abbildung 22.1: Beispiele für häufige Projekt-, Produkt- und Geschäftsrisiken.

- Ablauf des Risikomanagements:**
 - Risikoerkennung: Liste potenzieller Risiken** identifizierung möglicher Projekt-, Produkt- und Geschäftsrisiken.
 - Risikoanalyse: Priorisierte Liste der Risiken** Beurteilung der Eintrittswahrscheinlichkeit und Folgen dieser Risiken
 - Risikoplanung: Risikovermeidung und Notfallpläne** Erstellen von Plänen, die Maßnahme zur Vermeidung und Verringerung der Auswirkungen definieren
 - Risikoüberwachung: Risikobewertung** Regelmäßiges Bewerten der Risiken und Pläne der Risikominimierung und deren Überarbeitung, sobald mehr Informationen über ein Risiko verfügbar sind
 - Nach **Risikoüberwachung** Loopback zu **Risikoanalyse**

Kapitel 22.1.1 Risikoerkennung

Risikoerkennung ist der erste Schritt des Risikomanagements und umfasst das Identifizieren der Risiken, die eine Gefahr für den Softwareentwicklungsprozess, die zu entwickelnde Software oder das Entwicklungsunternehmen darstellen können.

- **Checkliste sechs verschiedener Risikoarten:**
 - **Technologische Risiken:** Risiken die sich aus dem Hard- oder Softwaretechnologien ergeben, die im entwickelten System Verwendung finden
 - **Personalbezogene Risiken:** Risiken, die mit Personen aus Entwicklerteam zusammenhängen
 - **Unternehmensbezogene Risiken:** Risiken aus dem Unternehmensumfeld
 - **Risiken durch Werkzeuge:** Risiken, die sich aus den verwendeten Softwarewerkzeugen und Hilfssoftware bei der Systementwicklung ergeben.
 - **Anforderungsrisiken:** Risiken, die auf geänderte Kundenanforderungen und den Umgang mit diesen Änderungen zurückzuführen sind.
 - **Schätzrisiken:** Risiken, die sich aus den Management-Schätzungen der für die Systemerstellung benötigten Ressourcen ergeben.

Einige Beispiele möglicher Risiken aus jeder dieser Kategorien:

Art des Risikos	Mögliche Risiken
Technologisch	1. Die im System verwendete Datenbank kann nicht so viele Transaktionen pro Sekunde durchführen wie erwartet. 2. Fehler in wiederverwendbaren Softwarekomponenten müssen vor ihrer Wiederverwendung behoben werden.
Personenbezogen	3. Es ist unmöglich, genügend Personal mit den benötigten Fähigkeiten zu rekrutieren. 4. Wichtige Mitarbeiter sind krank oder zu kritischen Zeitpunkten nicht verfügbar. 5. Es gibt keine Möglichkeit, erforderliche Schulungen durchzuführen.
Unternehmensbezogen	6. Das Unternehmen wird umstrukturiert, sodass ein anderes Management für das Projekt verantwortlich ist. 7. Finanzielle Probleme des Unternehmens zwingen zu Kürzungen des Projektbudgets.
Werkzeuge	8. Der durch entsprechende Softwarewerkzeuge generierte Code ist ineffizient. 9. Softwarewerkzeuge arbeiten nicht wie erwartet durchgängig zusammen.
Anforderungen	10. Es werden Änderungen der Anforderungen vorgeschlagen, die eine beträchtliche Nachbearbeitung des Entwurfs nach sich ziehen. 11. Kunden verstehen die Auswirkungen von Anforderungsänderungen nicht.
Schätzung	12. Die zur Entwicklung der Software benötigte Zeit wird unterschätzt. 13. Die Anzahl der Fehlerbehebungen wird unterschätzt. 14. Der Umfang der Software wird unterschätzt.

Kapitel 22.1.2 Risikoanalyse

Während der Risikoanalyse werden alle erkannten Risiken der Reihe nach betrachtet und hinsichtlich ihrer Eintrittswahrscheinlichkeit und Konsequenzen beurteilt. Es gibt hierzu keinen einfachen Weg – Sie müssen auf Ihr eigenes Urteilsvermögen und auf Ihre Erfahrungen mit Problemen in früheren Projekten vertrauen. Es ist nicht möglich, die Wahrscheinlichkeit und das Ausmaß des Risikos präzise und in konkreten Zahlen vorherzusagen. Stattdessen sollten Sie das Risiko einem Wertebereich zuweisen:

- **Wahrscheinlichkeit des Risikos:**
 - sehr gering
 - niedrig
 - mittel
 - hoch
 - sehr hoch
- **Auswirkung des Risikos:**
 - katastrophal (bedroht den Fortbestand des Projekts)
 - kritisch (würde zu wesentlicher Verzögerung führen)
 - geringfügig (Verzögerungen sind innerhalb der erlaubten Möglichkeiten)
 - unwesentlich

Resultate sollten nach Analyseprozess tabellarisch in der Reihenfolge der Auswirkungen angeordnet werden:

Risiko	Wahrscheinlichkeit	Auswirkungen
Finanzielle Probleme des Unternehmens zwingen zu Kürzungen des Projektbudgets. (7)	Niedrig	Katastrophal
Es ist unmöglich, genügend Personal mit den benötigten Fähigkeiten zu rekrutieren. (3)	Hoch	Katastrophal
Wichtige Mitarbeiter sind krank oder zu kritischen Zeitpunkten nicht verfügbar. (4)	Mittel	Kritisch
Zur Wiederverwendung vorgesehene Softwarekomponenten enthalten Fehler, die ihre Funktionalität einschränken. (2)	Mittel	Kritisch
Es werden Änderungen der Anforderungen vorgeschlagen, die eine beträchtliche Nachbearbeitung des Entwurfs nach sich ziehen. (10)	Mittel	Kritisch
Das Unternehmen wird umstrukturiert, sodass ein anderes Management für das Projekt verantwortlich ist. (6)	Hoch	Kritisch
Die im System verwendete Datenbank kann nicht so viele Transaktionen pro Sekunde durchführen wie erwartet. (1)	Mittel	Kritisch
Die zur Entwicklung der Software benötigte Zeit wird unterschätzt. (12)	Hoch	Kritisch
Softwarewerkzeuge arbeiten nicht wie erwartet durchgängig zusammen. (9)	Hoch	Geringfügig
Kunden verstehen die Auswirkungen von Anforderungsänderungen nicht. (11)	Mittel	Geringfügig

Risiko	Wahrscheinlichkeit	Auswirkungen
Es gibt keine Möglichkeit, erforderliche Schulungen durchzuführen. (5)	Mittel	Geringfügig
Die Anzahl der Fehlerbehebungen wird unterschätzt. (13)	Mittel	Geringfügig
Der Umfang der Software wird unterschätzt. (14)	Hoch	Geringfügig
Der durch entsprechende Softwarewerkzeuge generierte Code ist ineffizient. (8)	Mittel	Unwesentlich

Abbildung 22.4: Risikoarten und Beispiele.

- Tabelle bei jedem Durchlauf durch den Risikomanagementprozess auf den neuesten Stand bringen.
- Entscheidung über die wichtigsten Risiken basierend auf der Kombination von Eintrittswahrscheinlichkeit und Auswirkungen.
- Eintrittswahrscheinlichkeit und Auswirkungen von Risiken können sich ändern, wenn mehr Informationen verfügbar sind und Risikomanagementpläne umgesetzt werden.
- Im Allgemeinen zählen alle katastrophalen Risiken und alle kritischen Risiken mit überdurchschnittlicher Eintrittswahrscheinlichkeit zu den wichtigsten Risiken.
- Boehm (1988) schlägt vor, die zehn wichtigsten Risiken zu identifizieren und zu überwachen, aber die genaue Anzahl kann projektspezifisch sein und zwischen fünf und fünfzehn liegen.
- Die Anzahl der zur Überwachung ausgewählten Risiken sollte sich an der Praxis orientieren.
- Es wird empfohlen, alle Risiken zu beachten, die katastrophale oder kritische Konsequenzen haben.

Risiko	Strategie
Finanzielle Probleme des Unternehmens	Erstellen Sie eine Zusammenfassung für das höhere Management, in der Sie aufzeigen, welchen wichtigen Beitrag das Projekt zum Erreichen der Ziele des Unternehmens leistet und warum Kürzungen des Projektbudgets nicht unbedingt kosteneffizient sind.
Rekrutierungsprobleme	Alarmieren Sie den Kunden bei potenziellen Schwierigkeiten und der Möglichkeit von Verzögerungen. Überprüfen Sie den Zukauf von Komponenten.
Krankheit des Personals	Reorganisieren Sie das Team so, dass es mehr Überschneidungen bei den Arbeiten gibt, damit die Mitarbeiter die Aufgabe der anderen besser verstehen.
Fehlerhafte Komponenten	Ersetzen Sie potenziell fehlerhafte Komponenten durch neue Komponenten, die als zuverlässig bekannt sind.
Änderungen der Anforderungen	Leiten Sie Information zur Rückverfolgbarkeit ab, um die Auswirkungen von Anforderungsänderungen zu beurteilen. Trennen Sie beim Entwurf die Daten so weit wie möglich von der Implementierung.
Umstrukturierung des Unternehmens	Erstellen Sie eine Zusammenfassung an das höhere Management, in der Sie aufzeigen, welchen wichtigen Beitrag das Projekt zum Erreichen der Ziele des Unternehmens leistet.

Risiko	Strategie
Datenbankleistung	Untersuchen Sie die Möglichkeit des Erwerbs einer Datenbank mit höherer Leistung.
Unterschätzte Entwicklungszeit	Untersuchen Sie, ob es sich lohnt, Komponenten zuzukaufen oder automatisch Code zu erzeugen.

Abbildung 22.5: Strategien zum Risikomanagement.

Kapitel 22.1.3 Risikoplanung

Die Risikoplanung entwickelt Strategien zur Bewältigung der Hauptbedrohungen für das Projekt. Für jedes Risiko werden Maßnahmen überlegt, um die Auswirkungen auf das Projekt zu minimieren. Es wird auch berücksichtigt, welche Daten zur Überwachung des Projekts erfasst werden müssen, um aufkommende Probleme frühzeitig zu erkennen. In der Risikoplanung werden "Was-wenn"-Fragen gestellt, die sowohl einzelne Risiken als auch Risikokombinationen und externe Faktoren berücksichtigen, die Auswirkungen auf diese Risiken haben könnten.

Beispielfragen:

- Was, wenn mehrere Entwickler gleichzeitig krank werden?
- Was, wenn eine Konjunkturschwäche zu Budgetkürzungen von 20% für das Projekt führen
- Was, wenn die Leistung von Open-Source-Software nicht stimmt und der einzige Experte für die Open-Source-Software das Unternehmen verlässt?
- Was, wenn das Unternehmen, das Softwarekomponenten liefert und wartet, sein Geschäft aufgibt?
- Was, wenn der Kunde die überarbeiteten Anforderungen nicht wie angekündigt liefert?

→ Die Antworten zu diesen Fragen liefern die **Risikomanagementsstrategien**

Risikomanagementsstrategien:

- **Vermeidungsstrategien:** Strategien, die darauf abzielen, die Eintrittswahrscheinlichkeit des Risikos zu verringern, z.B. Umgang mit fehlerhaften Komponenten.
- **Minimierungsstrategien:** Strategien, die darauf abzielen, die Auswirkungen des Risikos zu reduzieren, z.B. Strategie im Krankheitsfall des Personals.
- **Notfallpläne:** Strategien, die sich auf den schlimmsten Fall vorbereiten und eine Strategie zur Problemlösung parat haben, z.B. Strategie bei finanziellen Problemen des Unternehmens.

Kapitel 22.1.4 Risikoüberwachung

Die Risikoüberwachung stellt sicher, dass die Risikoannahmen noch gültig sind. Die Risikoüberwachung beurteilt regelmäßig die Wahrscheinlichkeit und die Auswirkungen der Risiken und passt sie an. Die Risikoüberwachung berücksichtigt andere Faktoren, die die Risiken beeinflussen können, wie z.B. die Anzahl der Anforderungsänderungen & hängt von der Art des Risikos ab

Einige Beispiele von Faktoren, die bei der Beurteilung dieser Risikoarten hilfreich sein können:

Art des Risikos	Mögliche Indikatoren
Technologisch	Verspätete Lieferung von Hardware oder Hilfssoftware; viele gemeldete technische Probleme
Personenbezogen	Schlechte Arbeitsmoral der Mitarbeiter; schlechtes Betriebsklima; hohe Mitarbeiterfluktuation
Unternehmensbezogen	Gerüchte im Unternehmen; zu geringe Aktivität des höheren Managements
Werkzeuge	Abneigung der Teammitglieder, Werkzeuge zu benutzen; Beschwerden über Softwarewerkzeuge; Forderungen nach schnelleren Computern/mehr Speicher usw.
Anforderungen	Viele Anfragen zur Änderung der Anforderungen; Beschwerden der Kunden
Schätzung	Überschreiten des vereinbarten Zeitplans; Versäumnisse, gemeldete Fehler zu beseitigen

Abbildung 22.6: Risikoindikatoren.

- Die Risikoüberwachung ist ein wichtiger Teil des Risikomanagements.
- Die Risikoüberwachung überprüft regelmäßig die Wahrscheinlichkeit und die Auswirkungen der Hauptrisiken in jedem Projektstadium.
- Die Risikoüberwachung passt die Risikoplanung an, wenn sich die Situation ändert.

Kapitel 22.2 Personalmanagement

Mitarbeiter einer Softwarefirma sind deren größtes Kapital und sollten respektiert werden und passend zu ihren Fähigkeiten eingesetzt werden. Softwareprojektmanager müssen die technischen Probleme der Softwareentwicklung verstehen, aber auch gute Personalführungskompetenzen haben.

- **Gleichheit:** Alle Mitarbeiter sollten gleich behandelt werden und ihren Beitrag zum Projekt wertgeschätzt werden.
- **Respekt:** Jeder Mitarbeiter sollte seine individuellen Qualifikationen und Kompetenzen respektiert bekommen und die Möglichkeit haben, seine Ideen einzubringen.
- **Integration:** Mitarbeiter sollten sich gehört und ernst genommen fühlen und in einem offenen und kooperativen Arbeitsumfeld arbeiten.
- **Ehrlichkeit:** Als Manager sollten Sie immer ehrlich über die Stärken und Schwächen des Teams sprechen und auch Ihre eigenen technischen Grenzen anerkennen und Hilfe annehmen, wenn nötig.

→ Praxisbezogenes Personalmanagement sollte auf Erfahrungen basieren

Kapitel 22.2.1 Mitarbeitermotivation

- Ein Projektmanager muss die Mitarbeiter motivieren, damit sie effektiv arbeiten.
- Motivation hängt von der Befriedigung der Bedürfnisse ab, die Maslow (1954) in mehreren Ebenen angeordnet hat.

- Die **unteren Ebenen sind die grundlegenden Bedürfnisse** wie Nahrung, Schlaf, Sicherheit usw.
- Die **höheren Ebenen sind die sozialen Bedürfnisse** wie Zugehörigkeit, Anerkennung, Selbstverwirklichung usw.
- Die **grundlegenden Bedürfnisse haben Vorrang vor den abstrakteren Bedürfnissen.**

Hierarchie menschlicher Bedürfnisse:

↓ (**sozialen Bedürfnisse von oben nach unten**) ↓

1. Bedürfnis nach Selbstverwirklichung
2. Bedürfnis nach Anerkennung
3. Soziale Bedürfnisse
4. Sicherheitsbedürfnisse- **Soziale Bedürfnisse:**

- Um Mitarbeiter zu motivieren sollten Kollegen untereinander reden und sich treffen.
- Sozialräume im Office um Interaktionen und Teamgefühl zu steigern
- **Bedürfnis nach Anerkennung:**
 - öffentliche Anerkennung von Erfolgen als einfacher aber wirksamer Weg
 - passende Bezahlung
- **Bedürfnis der Selbstverwirklichung**
 - Mitarbeitern Verantwortung für ihre Arbeit übertragen und anspruchsvolle (aber erfüllbare) Aufgaben zuweisen
 - Weiterbildung als Motivationsfaktor

5. Physiologische Bedürfnisse

↑ (**grundlegende Bedürfnisse von unten nach oben**) ↑

→ Aus Sicht der Managements sind soziale Bedürfnisse und Bedürfnis der Anerkennung am wichtigsten

- **Soziale Bedürfnisse:**
 - Um Mitarbeiter zu motivieren sollten Kollegen untereinander reden und sich treffen.
 - Sozialräume im Office um Interaktionen und Teamgefühl zu steigern
- **Bedürfnis nach Anerkennung:**
 - öffentliche Anerkennung von Erfolgen als einfacher aber wirksamer Weg
 - passende Bezahlung
- **Bedürfnis der Selbstverwirklichung**
 - Mitarbeitern Verantwortung für ihre Arbeit übertragen und anspruchsvolle (aber erfüllbare) Aufgaben zuweisen
 - Weiterbildung als Motivationsfaktor

Die Kernessenz des Textes ist, dass die Motivation der Mitarbeiter von verschiedenen Faktoren abhängt und dass die Manager darauf achten müssen, sowohl die individuellen als auch die kollektiven Bedürfnisse zu erfüllen.

Drei Arten von Mitarbeitern:

- **Aufgabenorientierte Mitarbeiter:**
 - Beziehen ihre Motivation aus ihrer Arbeit

- Oftmals Informatiker, die von der intellektuellen Herausforderung des Softwareentwicklung motiviert werden
- **Selbstorientierte Mitarbeiter:**
 - lassen sich prinzipiell von persönlichem Erfolg und Anerkennung motivieren
 - an der Softwareentwicklung als ein Mittel zum Erreichen ihrer persönlichen Ziele interessiert
 - Oft werden sie eher von langfristigeren Zielen wie angestrebten Beförderungen motiviert → Um diese Ziele umsetzen zu können, möchten sie bei der Arbeit erfolgreich sein.
- **Interaktionsorientierte Mitarbeiter:**
 - beziehen ihre Motivation aus der Anwesenheit und den Handlungen ihrer Kollegen

Jeder Mitarbeiter hat eine Dominante Motivationskategorie, welche sich aber im Laufe der Zeit ändern kann je nach den Umständen und der Gruppendynamik

Kapitel 22.3 Teamwork

- Wie Teams organisiert werden und welche Vorteile ein gutes Teamwork hat
- Teams sollten klein und ausgewogen sein um *Kommunikation und Motivation* zu fördern

Vorteile einer Gruppe:

- **eigener Qualitätsstandard wird entwickelt:**
 - intern erstellte Standards werden eher eingehalten als extern erstellte
- **Einzelnen lernen voneinander und unterstützen sich gegenseitig**
- **Wissen wird geteilt**
- **Umgestaltung und ständige Verbesserung wird gefordert**
 - Gruppenmitglieder verfolgen gemeinsam das Ziel, erstklassige Ergebnisse zu liefern und Fehler zu beheben

Gute Projektmanager fördern den Gruppenzusammenhalt durch die Schaffung einer Gruppenidentität, die Durchführung gemeinsamer Veranstaltungen und die Einbeziehung aller Mitglieder. Sie sorgen dafür, dass alle auf dem gleichen Informationsstand sind und schaffen so ein Klima des Vertrauens. Die Effektivität einer Gruppe hängt jedoch auch von der Art des Projekts und der Stabilität des Unternehmens ab.

- **Größter Einfluss auf Teamarbeit:**
 - Die Menschen in der Gruppe
 - Die Organisation der Gruppe
 - Kommunikation zwischen Management und Mitarbeitern
-

Kapitel 22.3.1 Teammitglieder auswählen

Wie man ein geeignetes Team auswählt und wie die Persönlichkeiten und Motivationen der Teammitglieder die Teamleistung beeinflussen

- Teams sollten aus Mitarbeitern mit unterschiedlichen Fähigkeiten, Fachkenntnissen und Persönlichkeiten bestehen, um eine Vielzahl von Aufgaben zu erfüllen
- **Faktoren, die die Teamleistung beeinflussen:**
 - Qualifikationen und Fachkenntnisse : sollten Qualifikationen und Fachkenntnisse verfügen

- Persönlichkeiten : *sollten einander ergänzende Persönlichkeiten haben (selbstorientiert, aufgabenorientiert, interaktionsorientiert)*
- Motivationen : *sollten motiviert sein Software zu verbessern und Ziele zu erreichen*
- Interaktionen: *sollten effektiv miteinander kommunizieren und zusammenarbeiten*

Fallstudie :

- Die Projektmanagerin Alice hat versucht, eine Gruppe aus einander ergänzenden Persönlichkeiten zusammenzustellen, wie in Abbildung 22.10 dargestellt
- Die Gruppe besteht aus einer guten Mischung aus interaktions- und aufgabenorientierten Mitarbeitern, aber es gibt auch einige Probleme
- Dorothea ist eine selbstorientierte Mitarbeiterin, die nicht zufrieden mit ihrer zugewiesenen Aufgabe ist und versucht, ihre eigenen Ziele zu verfolgen
- Fred ist ein Teilzeit-Domänenexperte, der hauptsächlich an den technischen Herausforderungen interessiert ist und möglicherweise die Interaktion mit anderen Gruppenmitgliedern vernachlässigt
- Alice muss darauf achten, dass diese Probleme nicht die Teamleistung beeinträchtigen und dass alle Gruppenmitglieder an den Projektzielen beteiligt sind

Kapitel 22.3.2 Organisation der Gruppe

- Projektmanager sollte nicht gleichzeitig technischer Leiter der Gruppe sein
- Organisation wer wichtige technische Entscheidungen treffen sollen (Systemarchitekt, Projektmanager, Teammitglieder)
- Interaktion mit externen beteiligten sollte über Projektmanager ggf. auch spezielle externe Anlaufpersonen passieren
- Organisation wie externe Mitglieder in Gruppe interagiert werden
- Organisation wie Informationen weitergegeben werden, vorallem ist der Schutz vor Überinformation wichtig
- Änderungen an der Software bedingen oft Änderungen an mehreren Teilen des Systems und folglich Diskussionen und Verhandlungen auf allen Ebenen der Hierarchie
- Weil Software sich so schnell entwickelt, wissen junge Nachwuchskräfte manchmal mehr als erfahrenere Mitarbeiter. Wenn die Kommunikation nur von oben nach unten stattfindet anstatt auf einer flachen Ebene kann es sein, dass erfahrene Mitarbeiter die Chance verpassen mit neuer Technologie zu arbeiten. Die jungen Mitarbeiter können dann genervt davon sein, veraltete Techniken zu benutzen.

Kapitel 22.3.3 Kommunikation in der Gruppe

- Kommunikation in der Gruppe ist essenziell für die Effektivität und Effizienz
- **Gruppengröße:**
 - Desto größer die Gruppe desto schwieriger die Kommunikation (**Formel für einseitige Kommunikationsverbindung ist $n \cdot (n-1)$ n : Gruppengröße**)
- **Gruppenstruktur**
 - informelle Gruppen sind besser als hierarchisch strukturierte Gruppen (PS4 Lobbies better than Work Groupchats)
- **Zusammensetzung der Gruppe**

- Gruppenmitglieder mit selben Persönlichkeitstypen streiten öfter miteinander, männlich weiblich gemischte Gruppen sind besser
 - **Arbeitsumgebung**
 - Kann sich je nach Umgebung positiv oder negativ auf Kommunikation auswirken
-

ZUSAMMENFASSUNG:

- Gutes Projektmanagement ist im Bereich Softwareentwicklung unerlässlich, wenn die Projekte innerhalb eines vorgegebenen Zeitplans und Budgets realisiert werden sollen.
 - Softwaremanagement unterscheidet sich von dem Management in anderen Entwicklungsbereichen insofern als Software nicht greifbar ist. Projekte können neuartig oder innovativ sein, sodass es keine Erfahrungswerte gibt, auf die beim Management zurückgegriffen werden kann. Softwareprozesse sind noch nicht so ausgereift wie klassische Fertigungsprozesse.
 - Risikomanagement bedeutet, die Hauptrisiken des Projekts zu ermitteln und zu bewerten, um ihre Eintrittswahrscheinlichkeit und ihre potenziellen Auswirkungen auf das Projekt festzulegen. Für sehr wahrscheinliche und möglicherweise ernste Risiken sollten Pläne erstellt werden, um sie zu vermeiden, zu bewältigen oder zu behandeln.
 - Personalmanagement umfasst die Wahl der richtigen Mitarbeiter für ein Projekt sowie die Organisation des Teams und seiner Arbeitsumgebung, damit alle so produktiv wie möglich arbeiten können.
 - Mitarbeiter lassen sich durch Interaktion mit anderen motivieren sowie durch Anerkennung von Seiten des Managements und durch die Möglichkeit der persönlichen Weiterentwicklung.
 - Softwareentwicklungsgruppen sollten eher klein sein und sich durch ihren Zusammenhalt auszeichnen. Die Schlüsselfaktoren, die die Effektivität einer Gruppe beeinflussen, sind die Mitarbeiter der Gruppe, die Organisationsform der Gruppe und die Kommunikation zwischen den Gruppenmitgliedern.
 - Die Kommunikation innerhalb einer Gruppe wird von Faktoren wie den Statusunterschieden der Mitglieder, der Gruppengröße, der Geschlechterzusammensetzung, den Persönlichkeiten sowie den verfügbaren Kommunikationskanälen beeinflusst
-

ÜBUNGEN

1. Beschreiben Sie, warum die Nichtgreifbarkeit von Softwaresystemen beim Management von Softwareprojekten spezielle Probleme bereitet.
2. Beschreiben Sie, warum die besten Programmierer nicht automatisch die besten Softwaremanager sind. Es könnte hilfreich sein, für Ihre Antwort die Liste der Managementaufgaben vom Anfang dieses Kapitels (direkt vor Abschnitt 22.1) heranzuziehen.
3. Suchen Sie in der einschlägigen Literatur nach dokumentierten Softwareprojektproblemen und erstellen Sie eine Liste der Management-Schwierigkeiten und -Fehler in diesen gescheiterten Projekten. (Ich empfehle Ihnen, mit dem Buch The Mythical Man-Month von Fred Brooks zu beginnen.)
4. Nennen Sie sechs weitere Risiken, die zusätzlich zu denen in Abbildung 22.1 in Softwareprojekten auftreten können.
5. Nennen Sie Gründe, warum sich die Eintrittswahrscheinlichkeit von Risiken und die Risikofolgen während der Entwicklung eines Projekts meistens ändern.
6. Festpreisverträge, bei denen der Auftragnehmer ein festes Preisangebot für die Entwicklung eines Softwaresystems macht, können dazu dienen, das Projektrisiko vom Kunden auf den Auftragnehmer zu

verlagern. Wenn bei dem Projekt etwas schiefgeht, muss der Auftragnehmer zahlen. Überlegen Sie, wie der Einsatz solcher Verträge die Wahrscheinlichkeit von Produktrisiken erhöhen kann.

7. Erklären Sie, warum es den Zusammenhalt einer Gruppe stärken kann, wenn alle Gruppenmitglieder über den Fortschritt und die technischen Entscheidungen eines Softwareentwicklungsprojekts auf dem Laufenden gehalten werden.
8. Welche Probleme sehen Sie für XP-Teams, bei denen viele Managemententscheidungen den Teammitgliedern überlassen bleiben?
9. Schreiben Sie eine Fallstudie im Stil dieses Buches, um die Bedeutung der Kommunikation in einem Projektteam zu veranschaulichen. Gehen Sie davon aus, dass einige Teammitglieder nicht vor Ort arbeiten und dass es nicht möglich ist, das gesamte Team kurzfristig zusammenzurufen.
10. Sie werden von Ihrem Manager aufgefordert, Software in einem Zeitraum abzuliefern, der, wie Sie wissen, nur eingehalten werden kann, wenn Sie von Ihrem Projektteam verlangen, unbezahlte Überstunden zu leisten. Alle Teammitglieder haben kleine Kinder. Diskutieren Sie, ob Sie die Aufforderung Ihres Managers annehmen und Ihr Team überzeugen sollten, mehr Zeit dem Unternehmen und weniger den Familien zu widmen. Welche Faktoren sind für Ihre Entscheidung ausschlaggebend?

Kapitel 23.3 Zeitplanung

Abschätzung wie lange welche Prozesse & Aufgaben brauchen

Eine kurze Einführung in die Zeitplanung von Softwareprojekten ist, dass es sich um einen Prozess handelt, bei dem die Projektarbeit in mehrere voneinander unabhängige Aufgaben zerlegt wird, die dann geschätzt und geplant werden. Die Zeitplanung berücksichtigt sowohl die Anforderungen des Kunden als auch die Ressourcen und Fähigkeiten des Entwicklungsteams. Die Zeitplanung kann je nach dem gewählten Softwareprozess unterschiedlich detailliert und flexibel sein. Die Zeitplanung hilft, die Abhängigkeiten und Risiken der Aufgaben zu identifizieren und zu verwalten, um Verzögerungen und Änderungen zu vermeiden oder zu bewältigen. Die Zeitplanung kann auch grafisch dargestellt werden, um den Überblick und die Kommunikation über den Projektfortschritt zu erleichtern.

Kapitel 23.3.1 Darstellung des Zeitplans

- **Kalenderbasierte Balkendiagramme**
 - Zeigen wer für welche Aktivität verantwortlich ist
 - die erwartete Laufzeit sowie Anfangs- und Endzeitpunkt der Aktivität
 - werden manchmal auch **Gantt-Diagramm** genannt
- **Ablaufdiagramme**
 - Zeigen Abhängigkeit zwischen den verschiedenen Aktivitäten, aus denen das Projekt besteht

Jede Aktivität besteht aus folgenden Angaben:

- eine Dauer in Kalendertagen oder -monaten
- einem Schätzwert für den Aufwand in Personentagen oder Personenmonaten zur Fertigstellung der Arbeit
- einem Abgabetermin, an dem die Aktivität erledigt sein sollte

- einem definierten Endpunkt. Es kann sich dabei um ein Dokument, ein Review-Meeting, die erfolgreiche Ausführung aller Tests usw. handeln

→ Zudem sollten auch Meilensteine gesetzt werden, welche als kuzrer Bericht dokumentiert werden

- Projektmanager erstellen Zeitplan und verteilen Ressourcen (vor allem Softwareentwickler) auf die Aufgaben
- Ressourcenzuweisungen werden in Projektverwaltungswerkzeugen eingegeben, die Balkendiagramme erstellen
- Teilzeitzuweisungen, Spezialisten und Verzögerungen können zu Terminproblemen führen
- Projektmanagementtools verwenden Tabellen und Datenbanken, um Balken- und Ablaufdiagramme zu erzeugen

23.4 Agile Planung

- Agile Methoden sind iterative Ansätze, die Software in Inkrementen entwickeln und ausliefern (**Step by Step**)
- Die Funktionalität der Inkremente wird während der Entwicklung beschlossen, um auf Änderungen der Anforderungen und Prioritäten des Kunden zu reagieren
- Agile Methoden wie Scrum und Extreme Programming haben einen zweistufigen Planungsansatz: Anfangsphase und Entwicklungsplanung
- am meisten benutzen agile Ansätze sind Scrum und Extreme Programming

Entwicklungsplanung:

- **Releaseplanung**: die auf Monate vorausschaut und entscheidet, welche Merkmale Teil eines Systemrelease sein sollten
- **Iterationsplanung**: betrachtet einen kürzeren Zeitabschnitt und konzentriert sich auf die Planung des nächsten Inkrements eines Systems. Diese umfasst normalerweise zwei bis vier Wochen Teamarbeit.

Das Planungsspiel ist eine Methode der agilen Softwareentwicklung, die auf User-Stories basiert. User-Stories sind kurze Beschreibungen der gewünschten Funktionen des Systems, die vom Entwicklungsteam und dem Kunden gemeinsam erarbeitet werden. Das Ziel des Planungsspiels ist es, den Zeit- und Ressourcenaufwand für die Umsetzung der User-Stories zu schätzen und einen Release- und Iterationsplan zu erstellen.

- User-Stories werden nach Aufwand sortiert und mit Punkten bewertet.
- Geschwindigkeit des Teams wird geschätzt und in Personentage umgerechnet.
- Releaseplanung wählt und priorisiert Storys für ein Release mit Kundenbeteiligung.
- Iterationsplanung wählt und implementiert Storys für eine Iteration mit Geschwindigkeitsanpassung.
- Aufgabenplanung zerlegt Storys in Entwicklungsaufgaben und verteilt sie an Entwickler.

Techniken der Aufwandsabschätzung:

- **Erfahrungsbasierte Techniken**: Die Schätzung der zukünftigen Aufwandsanforderungen basiert auf den Erfahrungen des Managers, die er im Fachbereich und in vorherigen Projekten gesammelt hat
- **Aufwandsschätzung mithilfe eines Algorithmus**: Abschätzung wird mithilfe von mathematischen Formeln und Projektattribute (wie Größe) und Prozessmerkmale (wie die Erfahrung der Mitarbeiter) berechnet

→ bei beiden fällen entscheiden ob Aufwand direkt schätzen oder lieber die Projekt und Produktmerkmale →
In der Anfangsphase eines Projekts weisen diese Schätzungen eine erhebliche Fehlermarge auf