

Prinzipien der Ethnographie

- Natürliche Umgebung
 - Aktivitäten in Alltagsumgebung untersuchen
- Ganzheitlichkeit
 - Einzelverhalten im Kontext untersuchen
- Beschreiben, nicht bewerten
 - Ist-Verhalten, nicht Soll-Verhalten
- Sicht der Handelnden einnehmen
 - Verhalten beschreiben in Begriffen, die für den Handelnden relevant und bedeutungsvoll sind

- ist eine Form der ethnographischen Untersuchung
- nach dem Meister-Lehrling-Modell
- Lernen durch Vormachen und Beobachtungen und zwischenzeitliches Fragen und Klären
- geprägt durch
 - Bescheidenheit
 - Neugier
 - Aufmerksamkeit
 - konkrete (statt abstrakter) Fragen

Beispiel eines Ablaufes

① Einleitung (15 min)

- Vorstellung, Ziele, Dank
- Zustimmung zu Aufzeichnung, Vertraulichkeit
- Arbeit, nicht Person wird betrachtet!
- Meinungen zu technischer Unterstützung?
- Überblick gewinnen

Beispiel eines Ablaufes

① Einleitung (15 min)

- Vorstellung, Ziele, Dank
- Zustimmung zu Aufzeichnung, Vertraulichkeit
- Arbeit, nicht Person wird betrachtet!
- Meinungen zu technischer Unterstützung?
- Überblick gewinnen

② Übergang (1 min)

- Regeln, Rollen, Beziehung
- ich frage, Sie dürfen abwehren

Beispiel eines Ablaufes

① Einleitung (15 min)

- Vorstellung, Ziele, Dank
- Zustimmung zu Aufzeichnung, Vertraulichkeit
- Arbeit, nicht Person wird betrachtet!
- Meinungen zu technischer Unterstützung?
- Überblick gewinnen

② Übergang (1 min)

- Regeln, Rollen, Beziehung
- ich frage, Sie dürfen abwehren

③ Erhebung im Kontext (2 Std.)

- Beobachtung und Nachfragen
- Notizen machen, mitlaufen, sich unsichtbar machen
- Pausen nach Wunsch

Beispiel eines Ablaufes

① Einleitung (15 min)

- Vorstellung, Ziele, Dank
- Zustimmung zu Aufzeichnung, Vertraulichkeit
- Arbeit, nicht Person wird betrachtet!
- Meinungen zu technischer Unterstützung?
- Überblick gewinnen

② Übergang (1 min)

- Regeln, Rollen, Beziehung
- ich frage, Sie dürfen abwehren

③ Erhebung im Kontext (2 Std.)

- Beobachtung und Nachfragen
- Notizen machen, mitlaufen, sich unsichtbar machen
- Pausen nach Wunsch

④ Zusammenfassung (15 min)

- was die Beschäftigte tut, ihre Rolle
- was wichtig ist
- Ergänzungen, Korrekturen?

	Ist-Zustand	Soll-Zustand	Folgen
Auswertung vorhandener Daten/Dokumente	+	-	-
Beobachtungen	+	o	-
Befragung			
- geschlossene Fragen	+	o	-
- offene Fragen	+	o	-
- hybride Fragen	+	o	-
Prototyping	-	+	+
partizipative Entwicklung	-	+	+

Zielsetzung:

- Anforderungen anhand eines Beispiels erheben und überprüfen
- technische Möglichkeiten überprüfen und demonstrieren
- frühzeitig mögliche Lösungsansätze präsentieren

Idee:

- rasche und billige Entwicklung eines prototypischen Systems als Diskussionsgrundlage

Typen von Prototypen

Wegwerf-Prototyp:

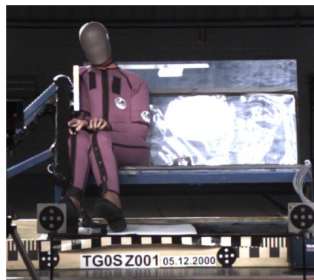
- beschreibt ein Softwaresystem exemplarisch
- dient zur Erhebung und Analyse von Anforderungen
- demonstriert die Funktionalität, die mit Stakeholdern diskutiert werden soll
- implementiert nicht notwendigerweise die gezeigte Funktionalität (z.B. GUI-Prototyp)
- ist als Komponente für das Endprodukt ungeeignet



Typen von Prototypen

Technischer Prototyp:

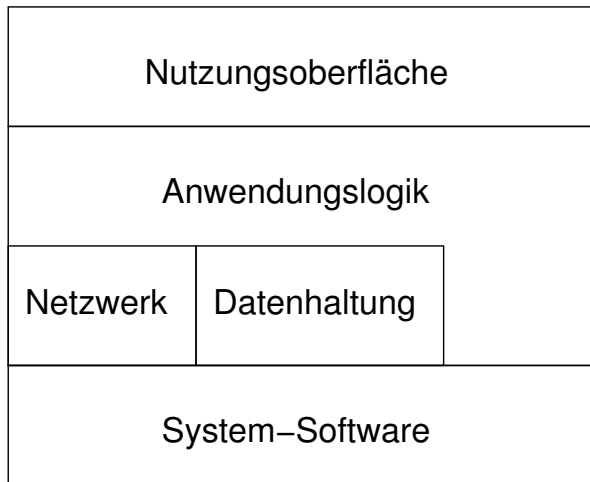
- zeigt die technische Umsetzbarkeit von Ansätzen zur Problemlösung
- implementiert einen (kleinen) Ausschnitt der Funktionalität des Systems
- wird eher zur Machbarkeitsabschätzung und -demonstration von Entwicklern eingesetzt



Evolutionärer Prototyp:

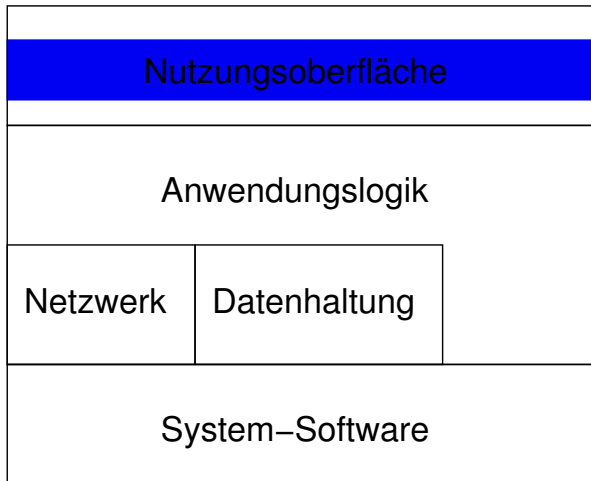
- dient zur schnellen Bereitstellung eines funktionsfähigen Systems im Rahmen von evolutionären Prozessmodellen zur Software-Entwicklung
- wird in weiteren Ausbaustufen zum endgültigen Produkt weiterentwickelt

Horizontaler vs. vertikaler Prototyp



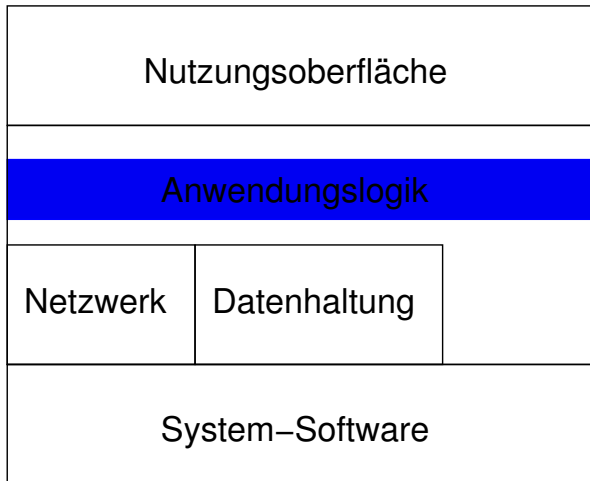
Horizontaler vs. vertikaler Prototyp

Horizontaler Prototyp: realisiert Aspekte einer spezifischen Ebene des Software-Systems; Bsp: Oberflächenprototyp



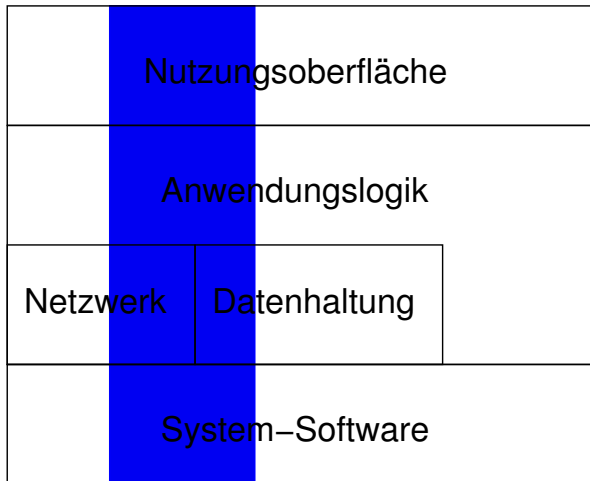
Horizontaler vs. vertikaler Prototyp

Horizontaler Prototyp: realisiert Aspekte einer spezifischen Ebene des Software-Systems; Bsp: Anwendungsprototyp



Horizontaler vs. vertikaler Prototyp

Vertikaler Prototyp: realisiert ausgewählte Aspekte des Softwaresystems vollständig



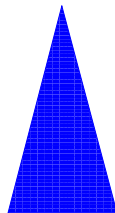
Prototypen: Storyboards

Storyboards

- dienen zur Implementierung von Prototypen
- demonstrieren das zu diskutierende Systemverhalten als „Geschichte“

Typen:

- passives Storyboard (Papierprototyp)
- aktives Storyboard (animierter Prototyp)
- interaktives Storyboard (ausführbarer Prototyp)



Aufwand

Passive Storyboards

Demonstration:

Analytiker spielt die Bedienung mit dem System durch, indem er entlang eines Anwendungsszenarios Eingabemöglichkeiten und Systemreaktionen demonstriert.

Mittel:

- Skizzen
- Bildschirm-Masken (Screenshots)
- mögliche Systemausgaben

Bemerkung:

- ermöglicht einfache und billige Prototyperstellung
- ermöglicht Interaktion mit Beteiligten am Beispiel

Beispiel passives Storyboard



Beispiel passives Storyboard



Beispiel passives Storyboard



Beispiel passives Storyboard



Aktive Storyboards

Demonstration:

Abspielen einer selbstablaufenden Präsentation des Systemverhaltens.

Mittel:

- Film, Diashow
- selbstablaufende PowerPoint-Präsentation

Bemerkung:

- ermöglicht einfache automatisierte Darstellung von typischen Anwendungsszenarien
- erfordert nicht unbedingt die Anwesenheit von Analytikern
- erlaubt keine Interaktion während der Präsentation

Interaktive Storyboards

Demonstration:

Prototyp ermöglicht dem Benutzer die frühzeitige Interaktion mit dem möglichen System; Funktionalität kann evtl. durch Analytiker „von Hand“ simuliert werden

Mittel:

- ausführbares Programm, das Teile der Funktionalität realisiert

Bemerkung:

- erfordert höheren Aufwand bei Prototyp-Erstellung
- ermöglicht Interaktion des Nutzers mit dem System
- erlaubt größtmögliche Nähe zum realen System

Vor- und Nachteile des Einsatzes von Prototypen

- + erlauben frühzeitige Demonstration von Lösungsansätzen
- + erlauben frühzeitige Beteiligung der Anwender
- + vermeiden das „Leere-Blatt-Syndrom“

Vor- und Nachteile des Einsatzes von Prototypen

- + erlauben frühzeitige Demonstration von Lösungsansätzen
- + erlauben frühzeitige Beteiligung der Anwender
- + vermeiden das „Leere-Blatt-Syndrom“
- + reduzieren Entwicklungsrisiken durch frühzeitige Diskussion mit Beteiligten
- + geeignete Werkzeuge ermöglichen die schnelle Erstellung von Prototypen

Vor- und Nachteile des Einsatzes von Prototypen

- + erlauben frühzeitige Demonstration von Lösungsansätzen
- + erlauben frühzeitige Beteiligung der Anwender
- + vermeiden das „Leere-Blatt-Syndrom“
- + reduzieren Entwicklungsrisiken durch frühzeitige Diskussion mit Beteiligten
- + geeignete Werkzeuge ermöglichen die schnelle Erstellung von Prototypen
- erfordern erhöhten Entwicklungsaufwand durch (zusätzliche) Prototyp-Entwicklung
- Gefahr, dass Wegwerf-Prototyp Teil des Produkts wird (z.B. aus Zeitdruck)

Definition

requirement: condition or capability needed by a user to solve a problem or achieve an objective.

Definition

requirement: condition or capability needed by a user to solve a problem or achieve an objective.

specification: document that specifies, in a complete, precise, verifiable manner, the requirements (, ...) of a system or component, and, often, the procedures for determining whether these provisions have been satisfied.

Definition

requirement: condition or capability needed by a user to solve a problem or achieve an objective.

specification: document that specifies, in a complete, precise, verifiable manner, the requirements (, ...) of a system or component, and, often, the procedures for determining whether these provisions have been satisfied.

software requirements specification (SRS): documentation of the essential requirements (functions, performance, design constraints, and attributes) of the software and its external interfaces.

Anforderungen sind gleichbedeutend mit Minimalbedingungen hinsichtlich Funktion und Qualität.

⇒ Wir müssen also die Funktion und Qualität definieren.

Anforderungen sind gleichbedeutend mit Minimalbedingungen hinsichtlich Funktion und Qualität.

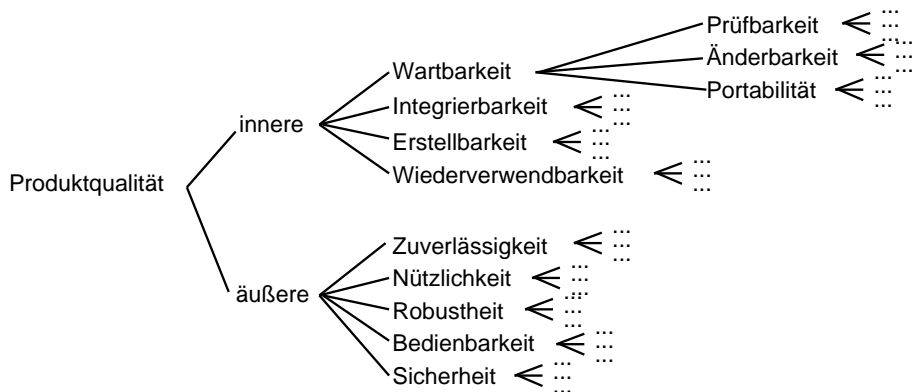
⇒ Wir müssen also die Funktion und Qualität definieren.

Definition

Funktion: in der Zeit ablaufende Transformation

- von Eingabedaten
- in Ausgabedaten
- unter Verwendung von Ressourcen

Produktqualitäten



Bedeutung einer Anforderungsspezifikation

Zweck

Abstimmung mit
Kunden

Folge von Mängeln

Die Anforderungen bleiben ungeklärt, Wünsche des Kunden bleiben unberücksichtigt.

Bedeutung einer Anforderungsspezifikation

Zweck

Folge von Mängeln

Abstimmung mit
Kunden

Die Anforderungen bleiben ungeklärt, Wünsche des Kunden bleiben unberücksichtigt.

Entwurf

Entwerfer fehlt Vorgabe, darum mehr Kommunikation / eigene Vorstellung als Vorgabe.

Bedeutung einer Anforderungsspezifikation

Zweck

Folge von Mängeln

Abstimmung mit
Kunden

Die Anforderungen bleiben ungeklärt, Wünsche des Kunden bleiben unberücksichtigt.

Entwurf

Entwerfer fehlt Vorgabe, darum mehr Kommunikation / eigene Vorstellung als Vorgabe.

Benutzerhandbuch

Basis für das Handbuch fehlt, es wird darum phänomenologisch verfasst.

Bedeutung einer Anforderungsspezifikation

Zweck

Folge von Mängeln

Abstimmung mit
Kunden

Die Anforderungen bleiben ungeklärt, Wünsche des Kunden bleiben unberücksichtigt.

Entwurf

Entwerfer fehlt Vorgabe, darum mehr Kommunikation / eigene Vorstellung als Vorgabe.

Benutzerhandbuch

Basis für das Handbuch fehlt, es wird darum phänomenologisch verfasst.

Testvorbereitung

systematischer Test ist unmöglich

Bedeutung einer Anforderungsspezifikation

Zweck

Folge von Mängeln

Abstimmung mit
Kunden

Die Anforderungen bleiben ungeklärt, Wünsche des Kunden bleiben unberücksichtigt.

Entwurf

Entwerfer fehlt Vorgabe, darum mehr Kommunikation / eigene Vorstellung als Vorgabe.

Benutzerhandbuch

Basis für das Handbuch fehlt, es wird darum phänomenologisch verfasst.

Testvorbereitung

systematischer Test ist unmöglich

Abnahme

Korrektheit ist subjektiv, Streit ist unvermeidbar.

Bedeutung einer Anforderungsspezifikation

Zweck	Folge von Mängeln
Abstimmung mit Kunden	Die Anforderungen bleiben ungeklärt, Wünsche des Kunden bleiben unberücksichtigt.
Entwurf	Entwerfer fehlt Vorgabe, darum mehr Kommunikation / eigene Vorstellung als Vorgabe.
Benutzerhandbuch	Basis für das Handbuch fehlt, es wird darum phänomenologisch verfasst.
Testvorbereitung	systematischer Test ist unmöglich
Abnahme	Korrektheit ist subjektiv, Streit ist unvermeidbar.
Wiederverwendung	nicht spezifizierte Systeme sind kaum durchschaubar, darum schwer wiederzuverwenden.

Bedeutung einer Anforderungsspezifikation

Zweck	Folge von Mängeln
Abstimmung mit Kunden	Die Anforderungen bleiben ungeklärt, Wünsche des Kunden bleiben unberücksichtigt.
Entwurf	Entwerfer fehlt Vorgabe, darum mehr Kommunikation / eigene Vorstellung als Vorgabe.
Benutzerhandbuch	Basis für das Handbuch fehlt, es wird darum phänomenologisch verfasst.
Testvorbereitung	systematischer Test ist unmöglich
Abnahme	Korrektheit ist subjektiv, Streit ist unvermeidbar.
Wiederverwendung	nicht spezifizierte Systeme sind kaum durchschaubar, darum schwer wiederzuverwenden.
spätere Reimplementierung	Kompatibilität setzt voraus, dass man weiß, womit die neue Software kompatibel sein soll.

inhaltlich

- (1) zutreffend (nicht „korrekt“!)
- (2) vollständig (relativ zu den Wünschen des Kunden)
- (3) widerspruchsfrei (oder konsistent, damit auch realisierbar)
- (4) neutral d.h. abstrakt (und damit offen für beliebigen Entwurf)

Angestrebte Eigenschaften der Spezifikation

inhaltlich

- (1) zutreffend (nicht „korrekt“!)
- (2) vollständig (relativ zu den Wünschen des Kunden)
- (3) widerspruchsfrei (oder konsistent, damit auch realisierbar)
- (4) neutral d.h. abstrakt (und damit offen für beliebigen Entwurf)

in der Darstellung

- (5) leicht verständlich (für alle Zielgruppen!)
- (6) präzise (schließt Umgangssprache aus)

Angestrebte Eigenschaften der Spezifikation

inhaltlich

- (1) zutreffend (nicht „korrekt“!)
- (2) vollständig (relativ zu den Wünschen des Kunden)
- (3) widerspruchsfrei (oder konsistent, damit auch realisierbar)
- (4) neutral d.h. abstrakt (und damit offen für beliebigen Entwurf)

in der Darstellung

- (5) leicht verständlich (für alle Zielgruppen!)
- (6) präzise (schließt Umgangssprache aus)

in der Form

- (7) leicht erstellbar (was die Notationen und Modelle betrifft)
- (8) leicht verwaltbar (also auch zweckmäßig strukturiert)
- (9) objektivierbar (auch – nicht sinnvoll – „testbar“ genannt)

Angestrebte Eigenschaften der Spezifikation

inhaltlich

- (1) zutreffend (nicht „korrekt“!)
- (2) vollständig (relativ zu den Wünschen des Kunden)
- (3) widerspruchsfrei (oder konsistent, damit auch realisierbar)
- (4) neutral d.h. abstrakt (und damit offen für beliebigen Entwurf)

in der Darstellung

- (5) leicht verständlich (für alle Zielgruppen!)
- (6) präzise (schließt Umgangssprache aus)

in der Form

- (7) leicht erstellbar (was die Notationen und Modelle betrifft)
- (8) leicht verwaltbar (also auch zweckmäßig strukturiert)
- (9) objektivierbar (auch – nicht sinnvoll – „testbar“ genannt)

Diese Merkmale konkurrieren, d.h. die Erfüllung des einen erschwert oder verhindert die Erfüllung des anderen.

- Ein Begriffslexikon anlegen und entwickeln

- Ein Begriffslexikon anlegen und entwickeln
- Von der Aufgabe ausgehen, nicht von ihrer Lösung

Regeln für Analyse und Spezifikation

- Ein Begriffslexikon anlegen und entwickeln
- Von der Aufgabe ausgehen, nicht von ihrer Lösung
- Daten suchen, nicht Programmabläufe beschreiben

Regeln für Analyse und Spezifikation

- Ein Begriffslexikon anlegen und entwickeln
- Von der Aufgabe ausgehen, nicht von ihrer Lösung
- Daten suchen, nicht Programmabläufe beschreiben
- Abstraktionsebene nicht in einer Darstellung wechseln

Regeln für Analyse und Spezifikation

- Ein Begriffslexikon anlegen und entwickeln
- Von der Aufgabe ausgehen, nicht von ihrer Lösung
- Daten suchen, nicht Programmabläufe beschreiben
- Abstraktionsebene nicht in einer Darstellung wechseln
- Die Spezifikation nach Aspekten organisieren

Regeln für Analyse und Spezifikation

- Ein Begriffslexikon anlegen und entwickeln
- Von der Aufgabe ausgehen, nicht von ihrer Lösung
- Daten suchen, nicht Programmabläufe beschreiben
- Abstraktionsebene nicht in einer Darstellung wechseln
- Die Spezifikation nach Aspekten organisieren
- Ein Mengengerüst bilden

Regeln für Analyse und Spezifikation

- Ein Begriffslexikon anlegen und entwickeln
- Von der Aufgabe ausgehen, nicht von ihrer Lösung
- Daten suchen, nicht Programmabläufe beschreiben
- Abstraktionsebene nicht in einer Darstellung wechseln
- Die Spezifikation nach Aspekten organisieren
- Ein Mengengerüst bilden
- Den Kunden (Anwender) einbeziehen

Regeln für Analyse und Spezifikation

- Ein Begriffslexikon anlegen und entwickeln
- Von der Aufgabe ausgehen, nicht von ihrer Lösung
- Daten suchen, nicht Programmabläufe beschreiben
- Abstraktionsebene nicht in einer Darstellung wechseln
- Die Spezifikation nach Aspekten organisieren
- Ein Mengengerüst bilden
- Den Kunden (Anwender) einbeziehen
- Geeignete Sprachen und Werkzeuge verwenden

Regeln für Analyse und Spezifikation

- Ein Begriffslexikon anlegen und entwickeln
- Von der Aufgabe ausgehen, nicht von ihrer Lösung
- Daten suchen, nicht Programmabläufe beschreiben
- Abstraktionsebene nicht in einer Darstellung wechseln
- Die Spezifikation nach Aspekten organisieren
- Ein Mengengerüst bilden
- Den Kunden (Anwender) einbeziehen
- Geeignete Sprachen und Werkzeuge verwenden
- Die Spezifikation so früh wie möglich prüfen und dem Konfigurationsmanagement unterstellen

Regeln für Analyse und Spezifikation

- Ein Begriffslexikon anlegen und entwickeln
- Von der Aufgabe ausgehen, nicht von ihrer Lösung
- Daten suchen, nicht Programmabläufe beschreiben
- Abstraktionsebene nicht in einer Darstellung wechseln
- Die Spezifikation nach Aspekten organisieren
- Ein Mengengerüst bilden
- Den Kunden (Anwender) einbeziehen
- Geeignete Sprachen und Werkzeuge verwenden
- Die Spezifikation so früh wie möglich prüfen und dem Konfigurationsmanagement unterstellen
- Die Spezifikation intensiv verwenden

Inhalt der Anforderungsspezifikation nach IEEE Standard 830.1998

1. Einführung

1.1 Zweck

1.2 Rahmen

1.3 Definitionen, Akronyme und Abkürzungen

1.4 Referenzen

1.5 Übersicht über das Dokument

2. Allgemeine Beschreibung

2.1 Produktperspektive

2.2 Produktfunktionen

2.3 Charakteristika der Benutzer

2.4 Einschränkungen

2.5 Annahmen und Abhängigkeiten

3. Detaillierte Beschreibung