

# Softwarequalität

## Validierung:

Erfüllt das System alle Anforderungen (Vollständigkeit)?

***Wird das richtige Produkt entwickelt?***

## Verifikation:

Beweis der Korrektheit des Systems

***Wird das Produkt richtig erstellt?***

→ *Aufspüren von Fehlern*

# Testverfahren



## statisch

- Softwareinspektion  
(z.B. Typprüfung,  
fehlerhafte Initialisierung)
- jederzeit einsetzbar

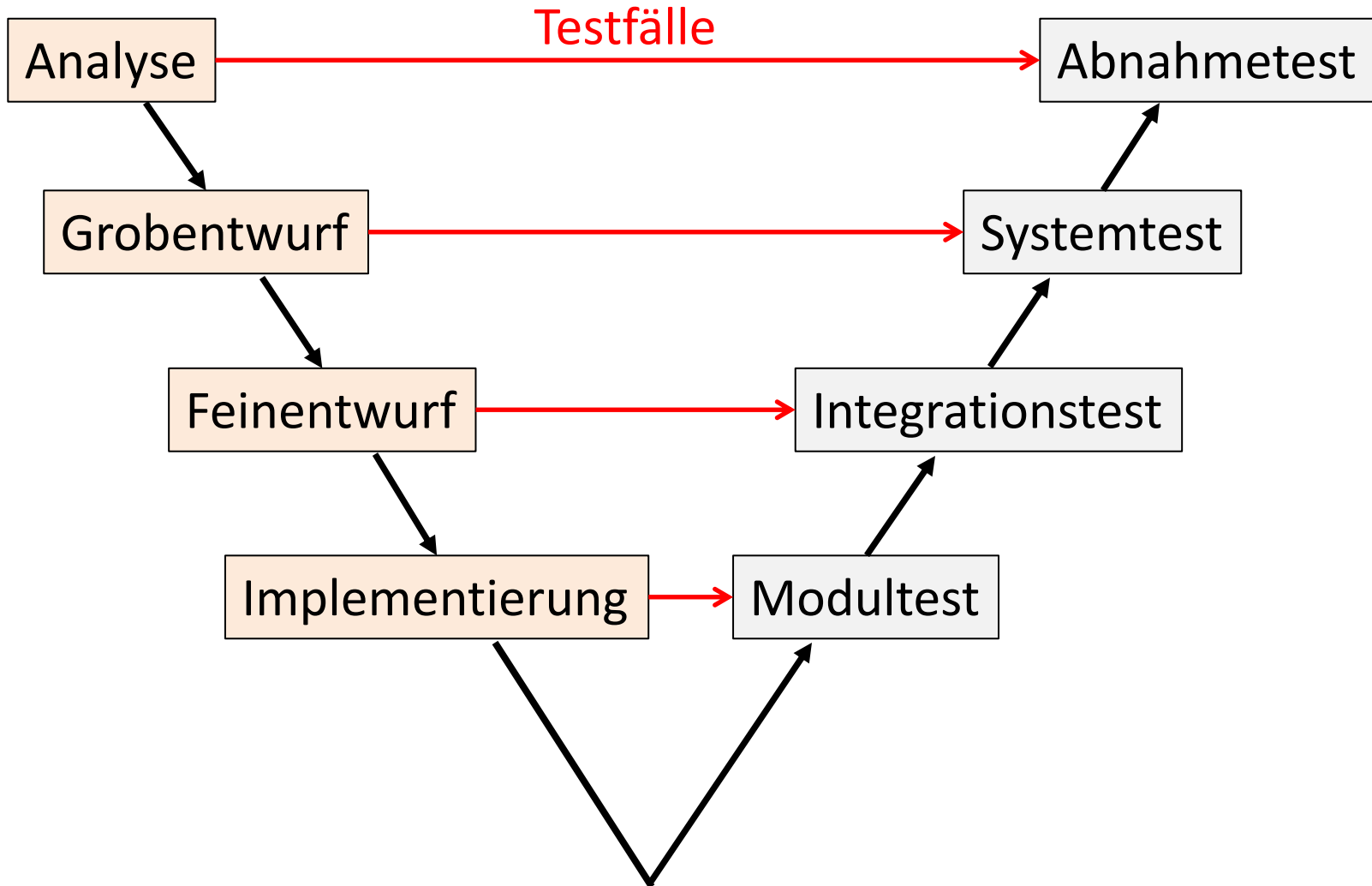
## dynamisch

- Softwaretests auch mit  
Eingaben, die Fehlverhalten  
auslösen
- erst nach der Implementierung  
einsetzbar

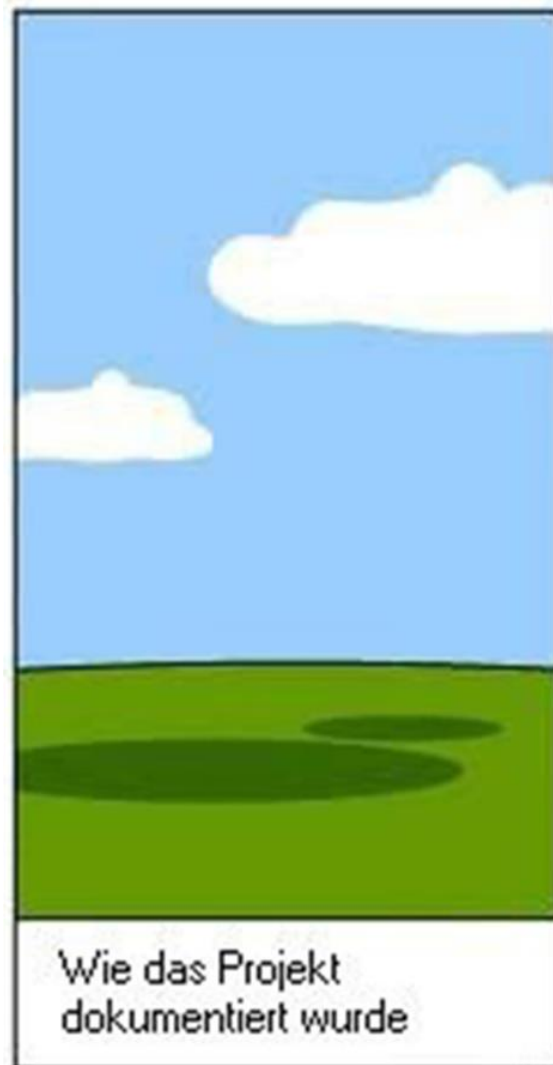
# Das V-Modell

<https://www.youtube.com/watch?v=P0TgRjj8hQg>

<https://www.youtube.com/watch?v=5ahlznlqFyo>



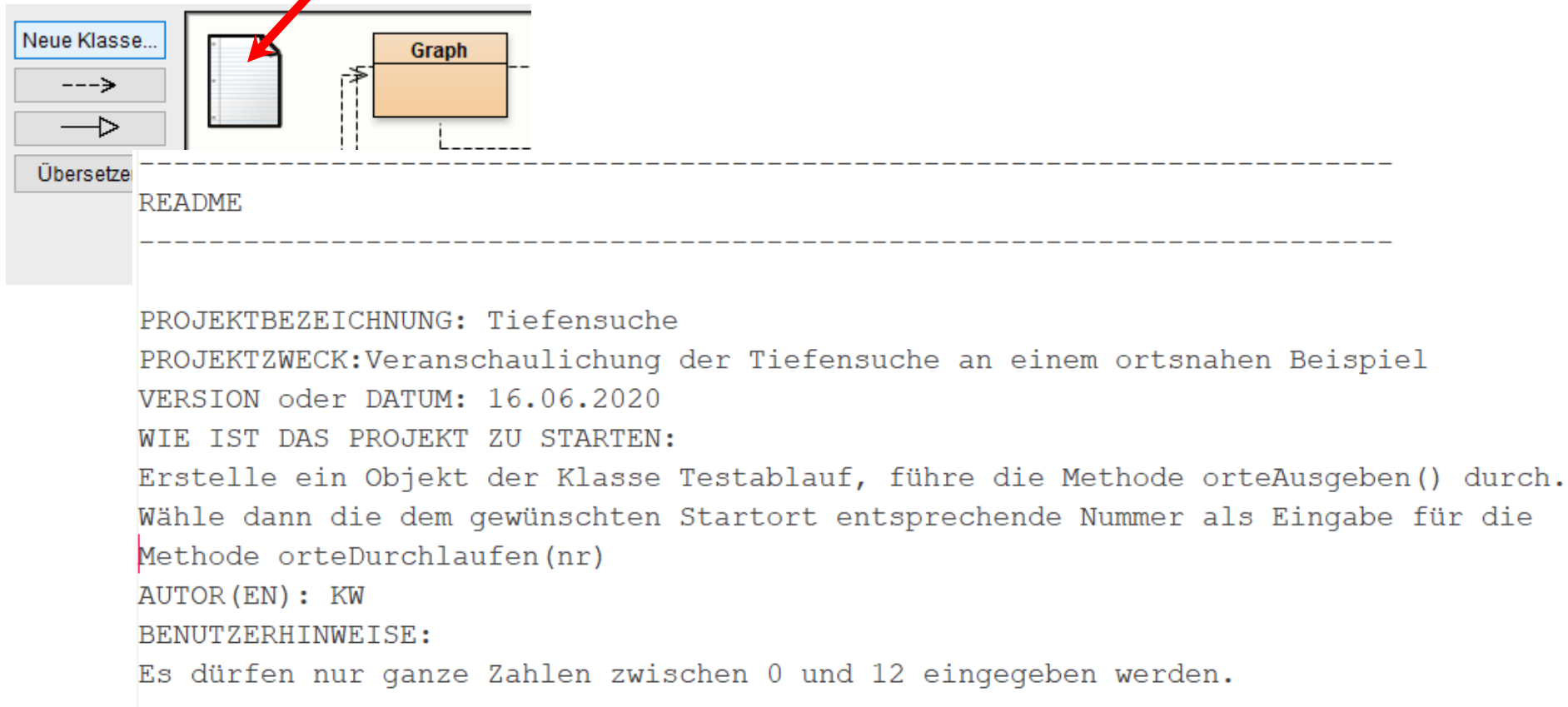
# Dokumentation von Softwareprojekten



# Dokumentation von Softwareprojekten

## Benutzer- oder Anwenderdokumentation:

Textblattsymbol in der linken oberen Ecke => readme.txt  
(= Bedienungsanleitung)



The screenshot shows a software development environment. On the left, there is a sidebar with buttons: "Neue Klasse...", a dashed arrow button, a solid arrow button, and "Übersetze". The main area displays a diagram with a document icon (representing a README file) and a box labeled "Graph". A red arrow points from the text "Textblattsymbol in der linken oberen Ecke" to the document icon. Below the diagram, the content of the README file is displayed, starting with "README" followed by a dashed line separator. The text in the README is as follows:

```
PROJEKTBEZEICHNUNG: Tiefensuche
PROJEKTZWECK: Veranschaulichung der Tiefensuche an einem ortsnahe Beispiel
VERSION oder DATUM: 16.06.2020
WIE IST DAS PROJEKT ZU STARTEN:
Erstelle ein Objekt der Klasse Testablauf, führe die Methode orteAusgeben() durch.
Wähle dann die dem gewünschten Startort entsprechende Nummer als Eingabe für die
Methode orteDurchlaufen(nr)
AUTOR (EN) : KW
BENUTZERHINWEISE:
Es dürfen nur ganze Zahlen zwischen 0 und 12 eingegeben werden.
```

# **System- oder Entwicklungsdokumentation:**

In Java javadoc

(in BlueJ: Werkzeuge → Dokumentation erzeugen)

Javadoc-Kommentare: `/** blablabla */`

Unmittelbar vor Klasse: Klassenkommentar

Vor Methode: Methodenkommentar.

Schlüsselwörter für Formatierung:

`@version, @author, @param, @return`

(innerhalb des Kommentars)

## Dokumentation einer **Klasse**:

- Klassenname
- allgemeiner Zweck und Eigenschaften
- Versionsnummer
- Autoren
- Dokumentation jeder Methode, auch Konstruktor:

## Dokumentation einer **Methode**:

- Name
- Ergebnistyp
- Parametername und –typ
- Zweck und Arbeitsweise
- Beschreibung jedes Parameters
- Beschreibung des Ergebnisses

Untersuche hierzu die  
Java-Klassenbibliotheken  
(unter BlueJ-Hilfe)!