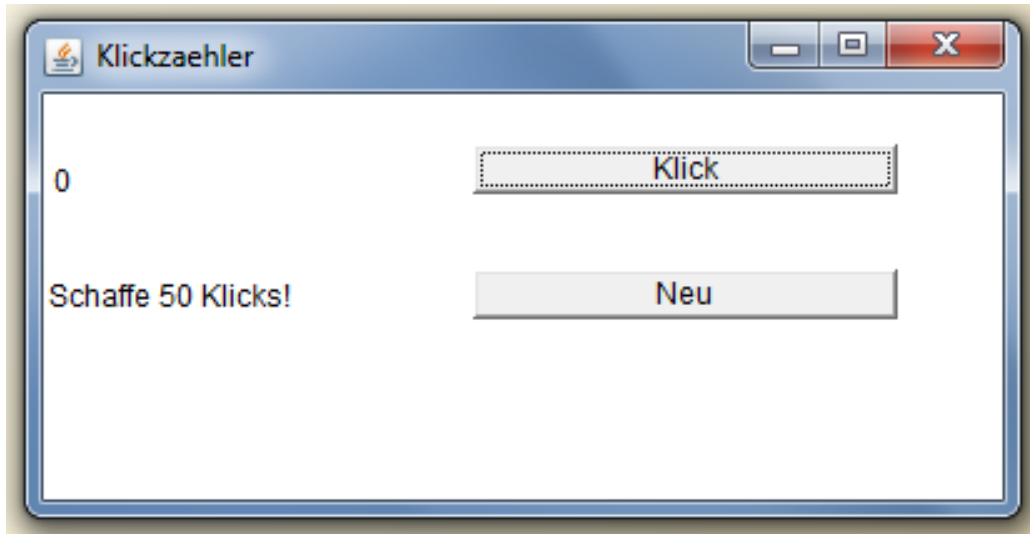




Entwurfsmuster – Modell-View-Controller



GUI-Programmierung

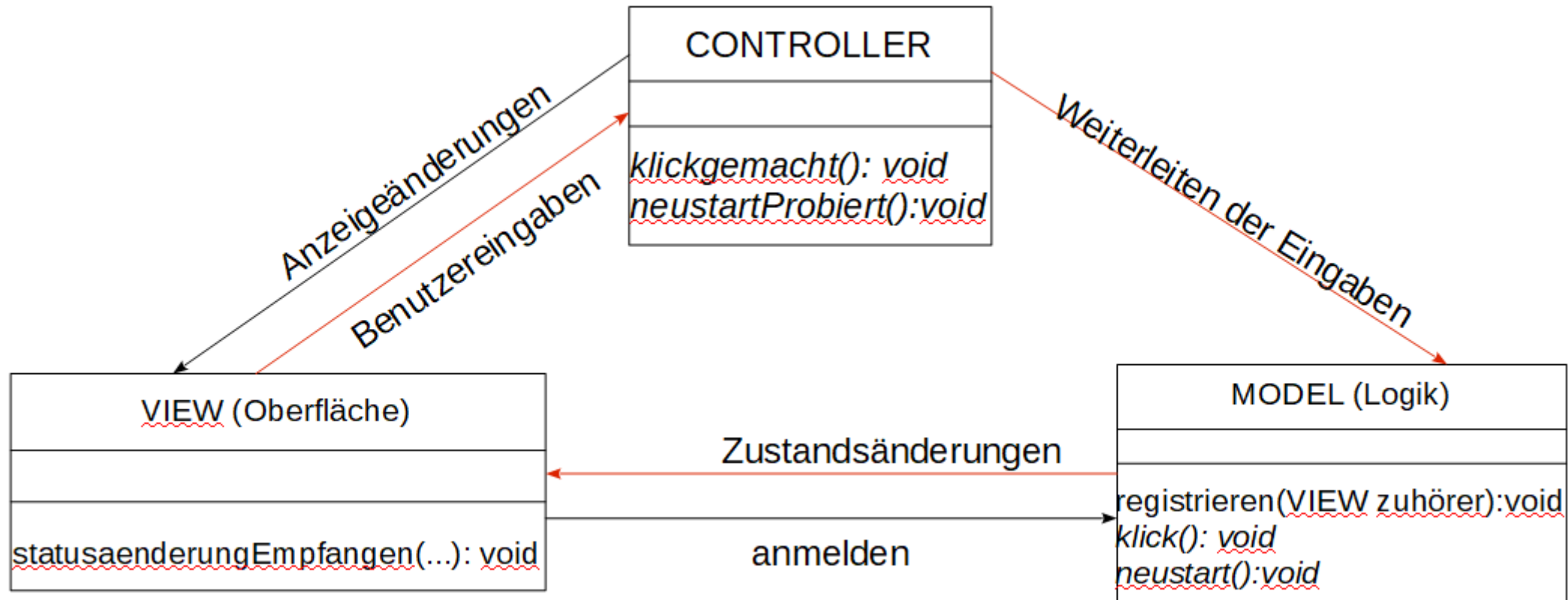


KLICKZAEHLER
<ul style="list-style-type: none">- klick: JButton- neu: JButton- zeit: JLabel- zaehler: JLabel
<ul style="list-style-type: none">+ KLICKZAEHLER()+ actionPerformed(e:(ActionEvent)) : void

- Schreibe eine Klasse Klickzähler. Nutze die GUI-Programmierung, um ein Fenster gemäß dem Bild zu gestalten.

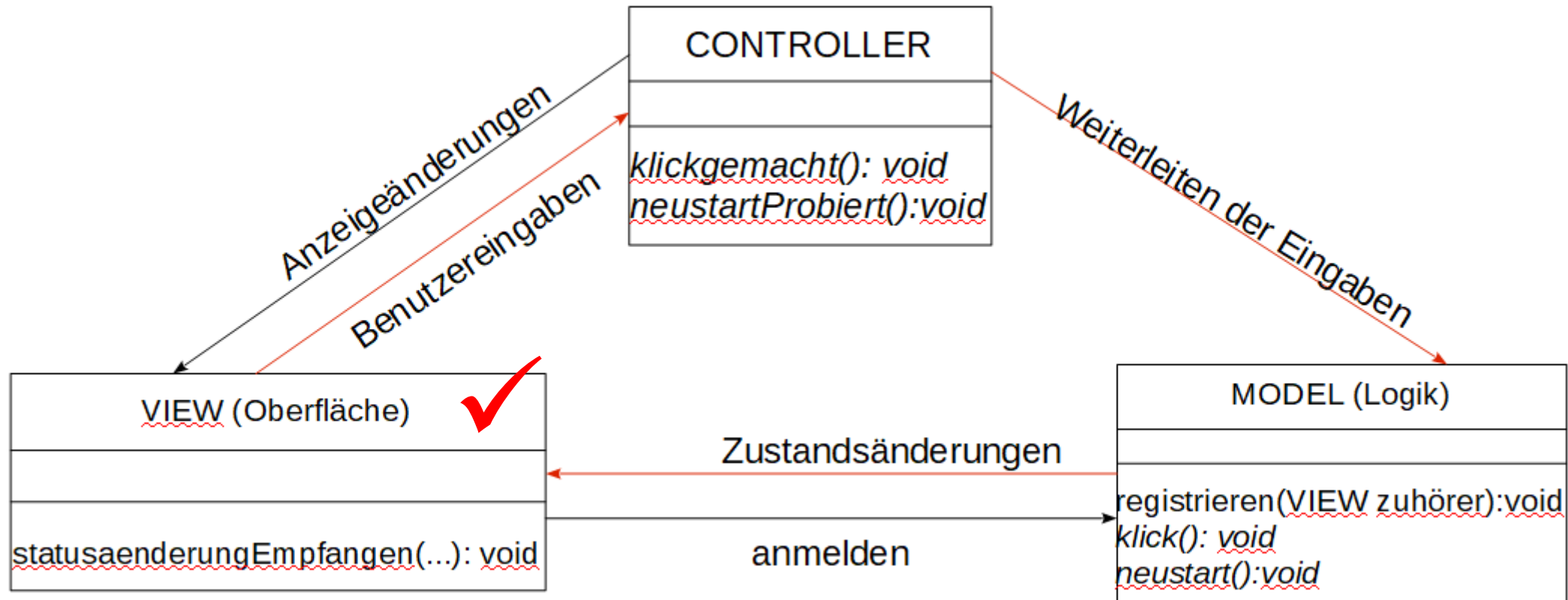
Entwurfsmuster MVC

- Das Entwurfsmuster MVC trennt MODEL (Logik), VIEW (Oberfläche) und CONTROLLER (Kontrolleur) voneinander.



Entwurfsmuster MVC

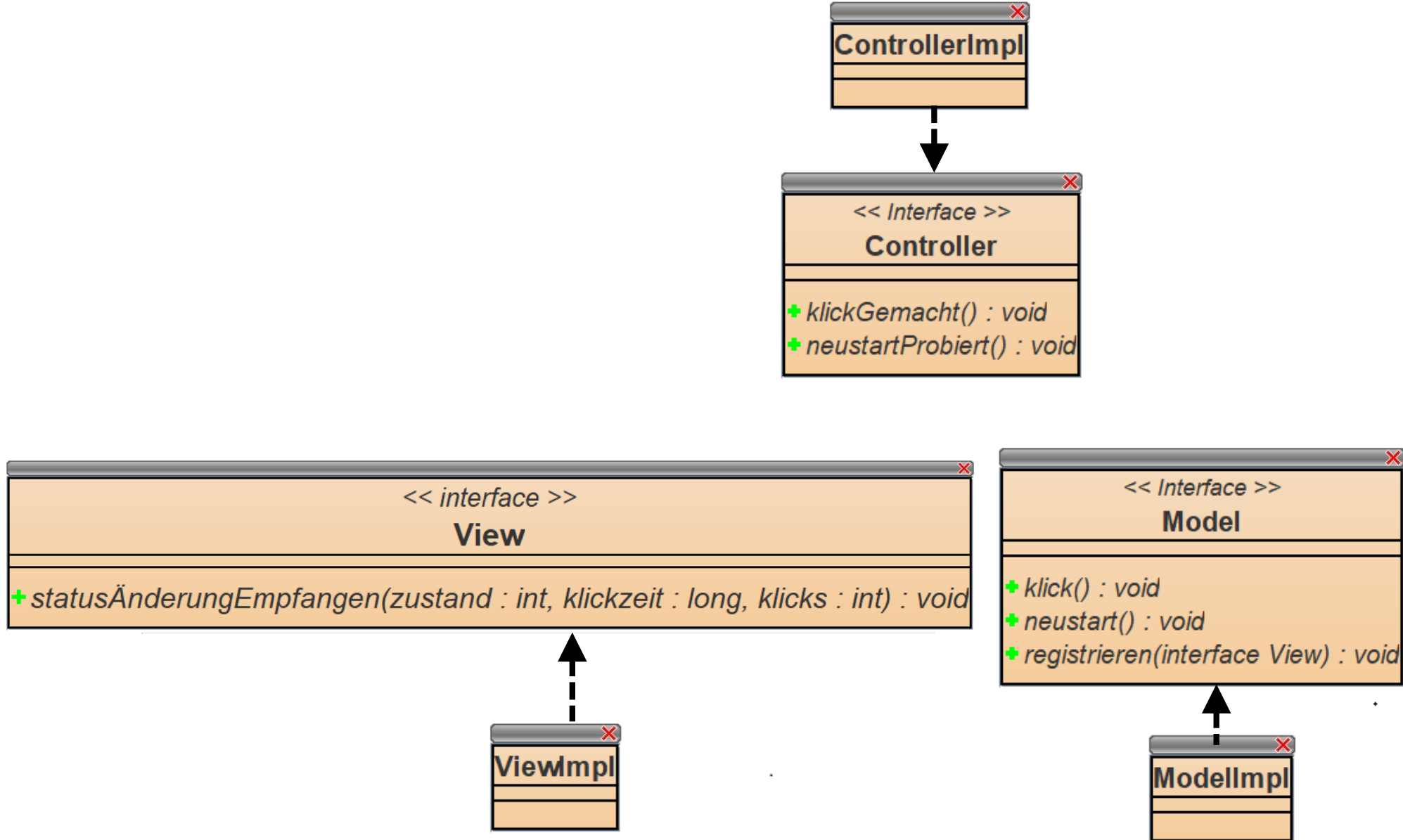
- Das Entwurfsmuster MVC trennt MODEL (Logik), VIEW (Oberfläche) und CONTROLLER (Kontrolleur) voneinander.



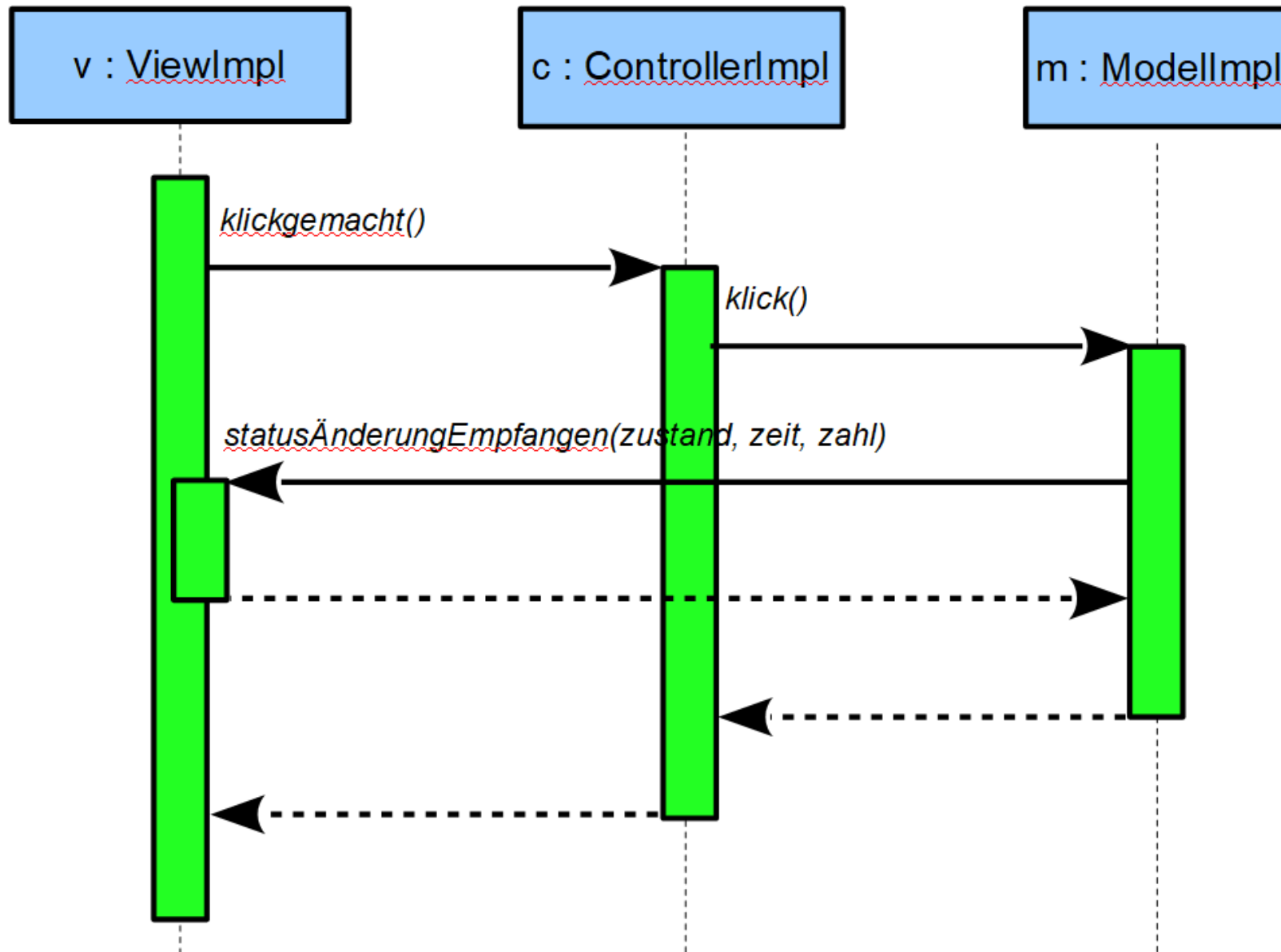
Aufgabe: MVC-Grundgerüst

- Erstelle die drei Interfaces View, Model und Controller.
- Benenne die Klasse Klickzähler zu ViewImpl um. Erstelle die Klasse ModellImpl und ControllerImpl, die die entsprechenden Interfaces einbinden.

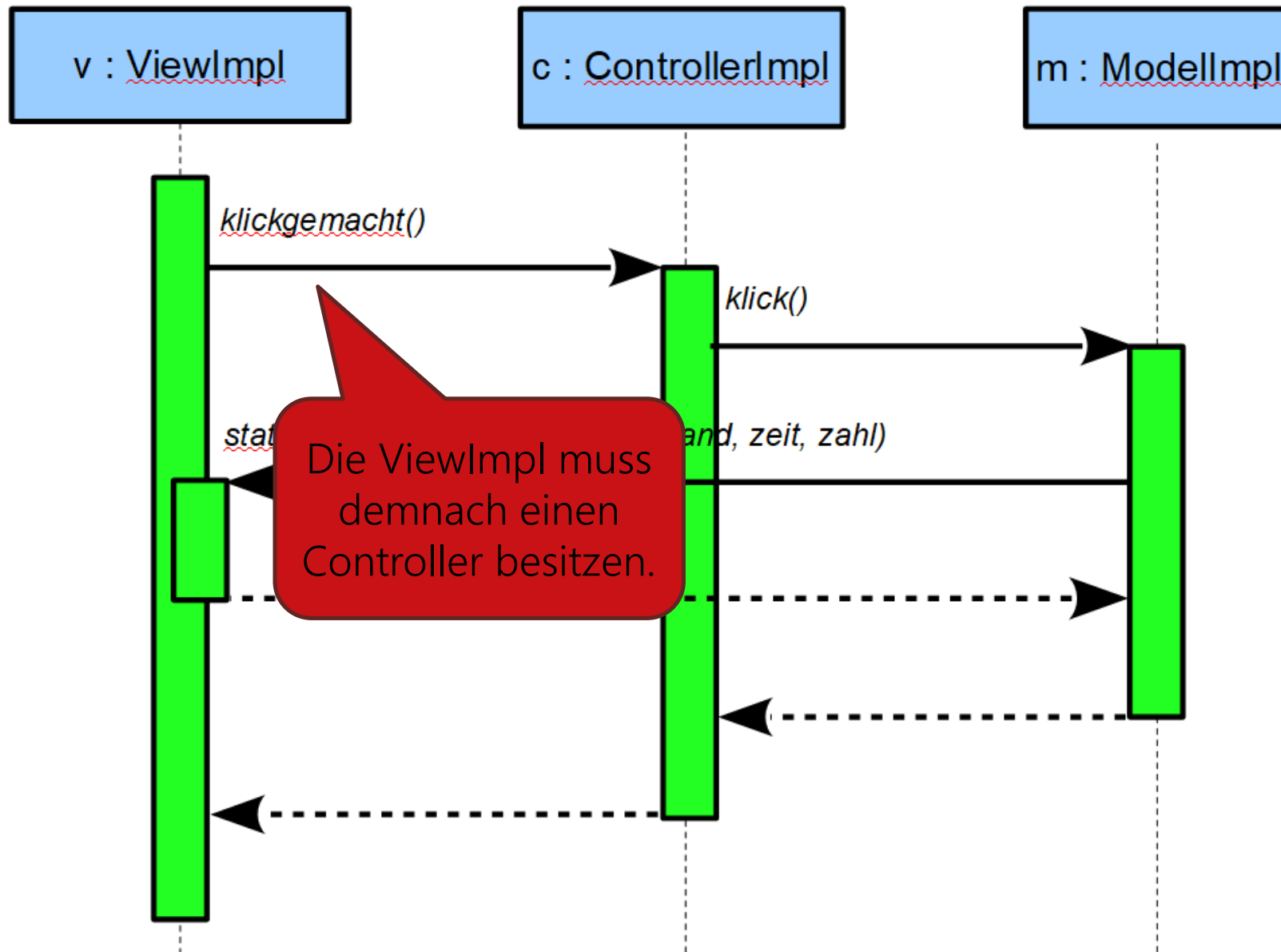
Aufgabe: MVC-Grundgerüst



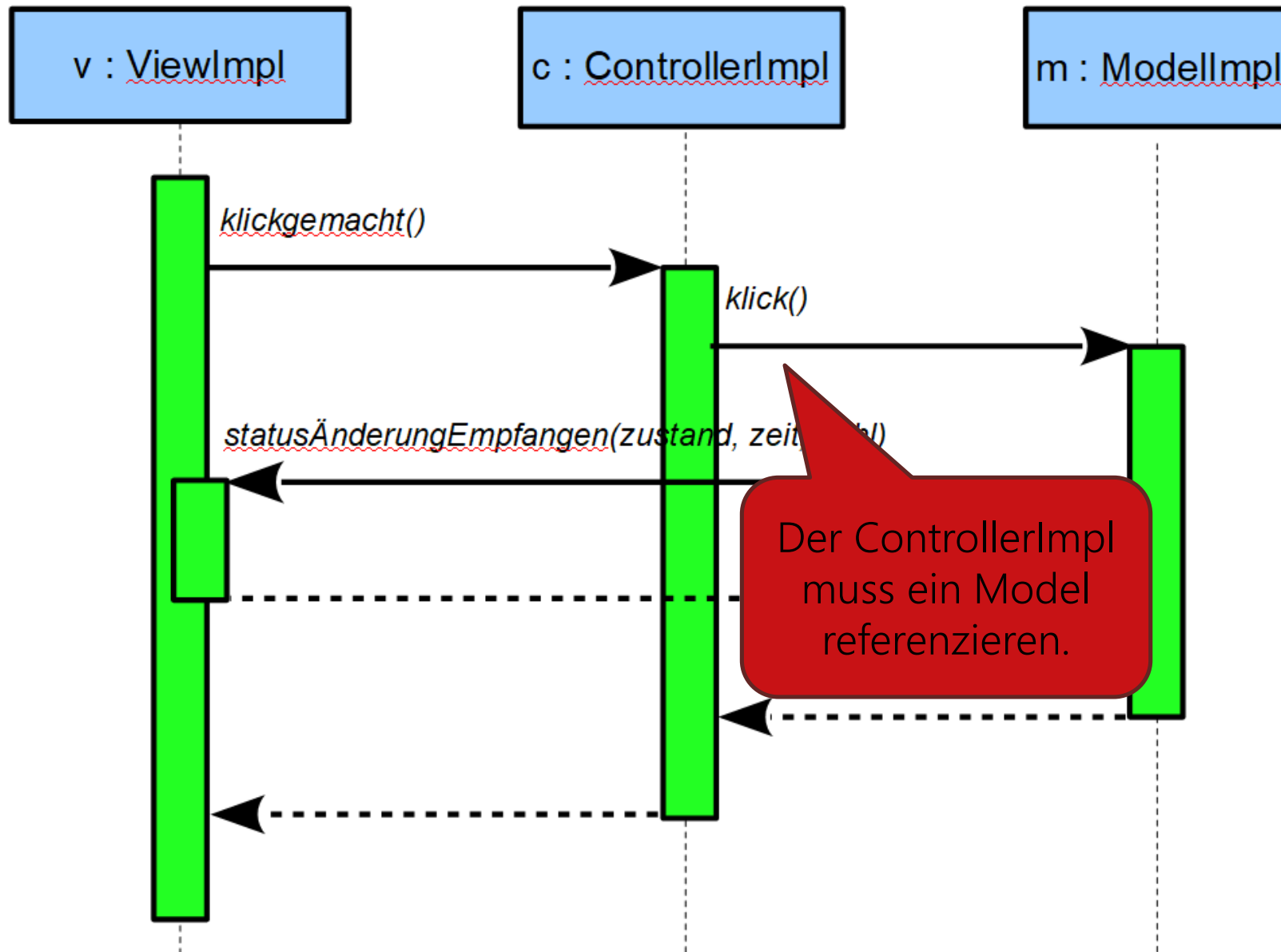
Funktionsweise MVC: Was passiert bei einem Klick?



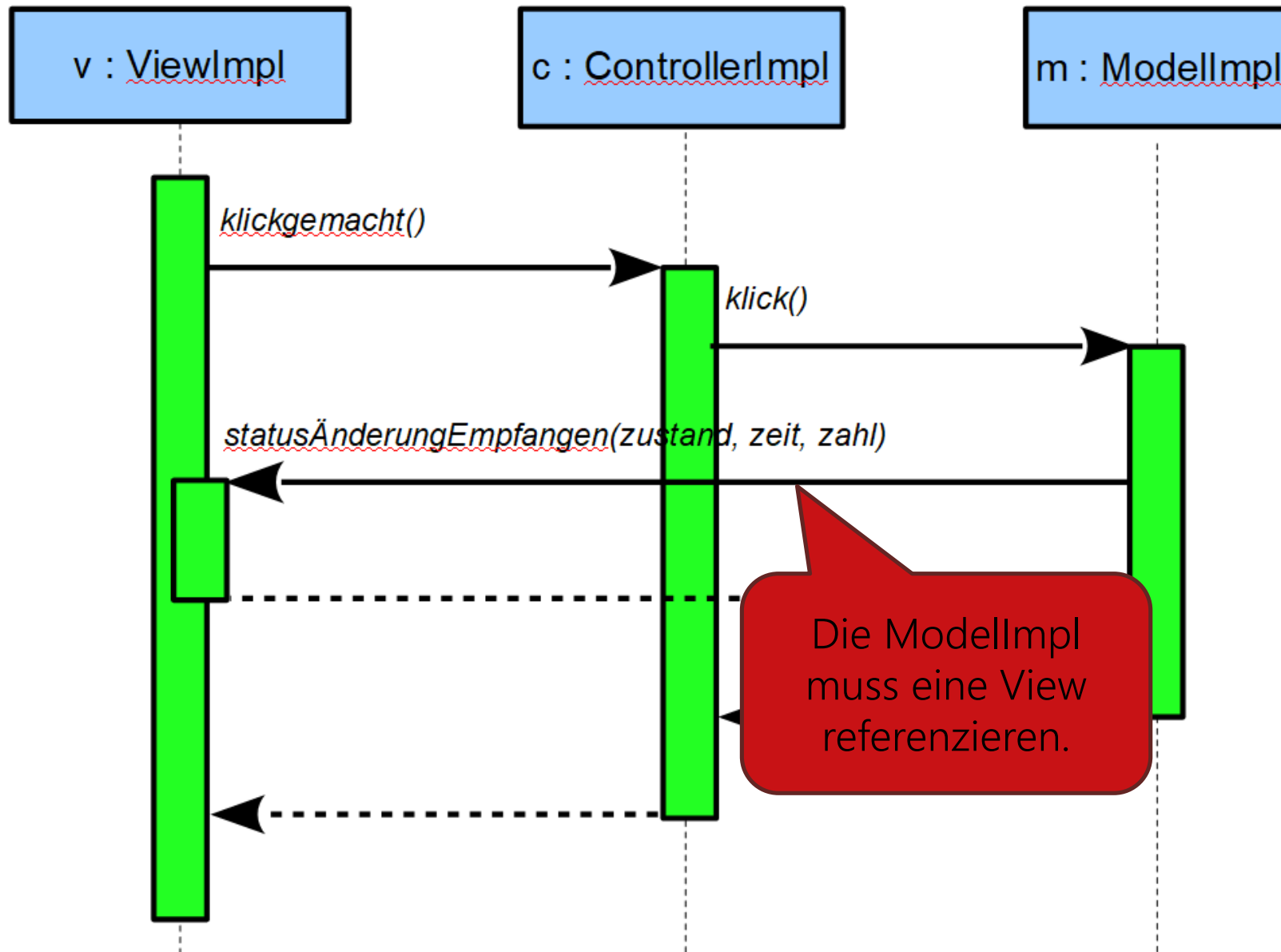
Funktionsweise MVC: Was passiert bei einem Klick?



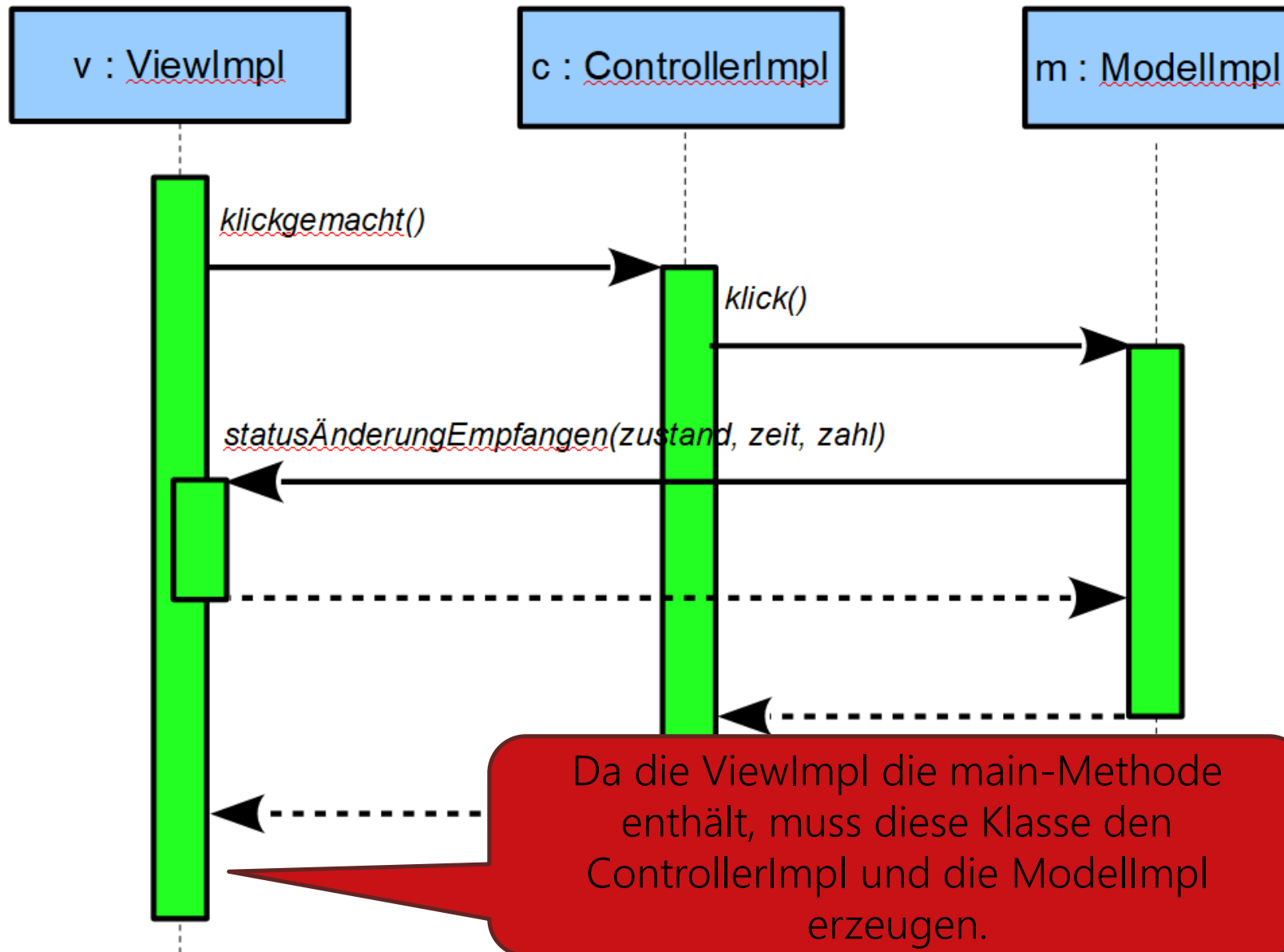
Funktionsweise MVC: Was passiert bei einem Klick?



Funktionsweise MVC: Was passiert bei einem Klick?



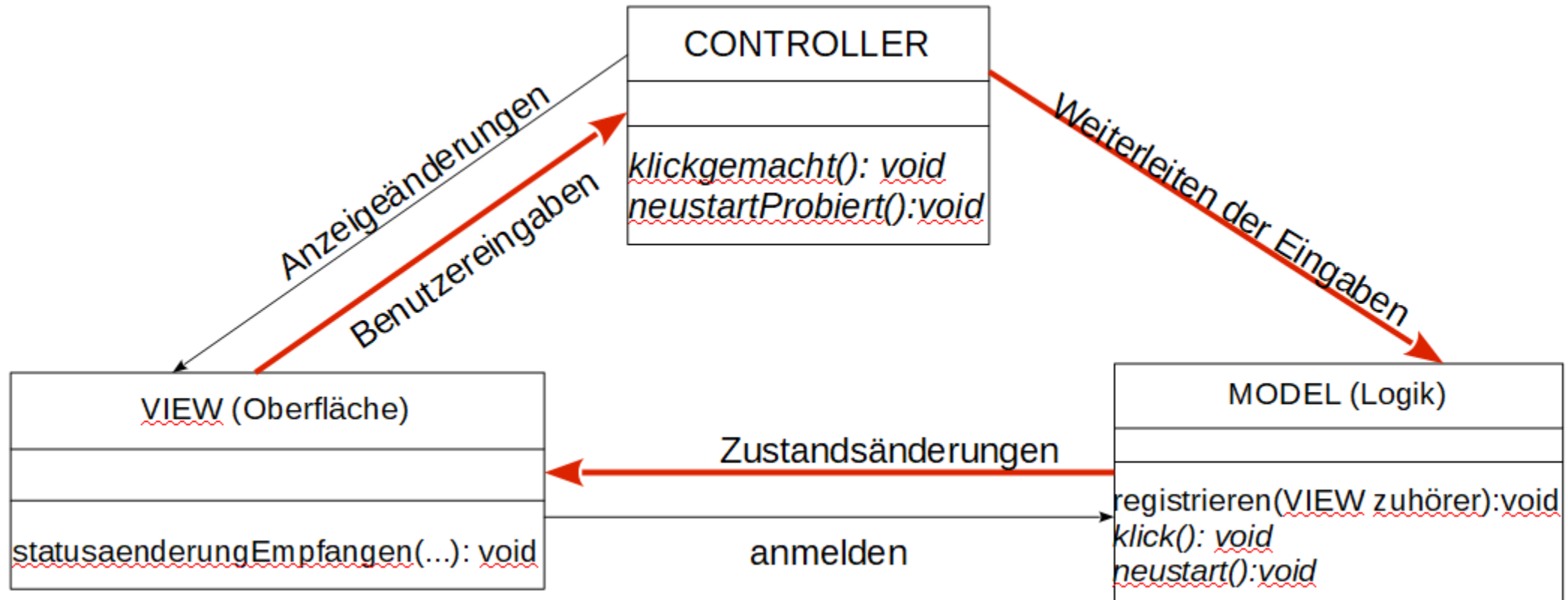
Funktionsweise MVC: Was passiert bei einem Klick?



MVC – Realisierung der Funktionsweise

- Die ModellImpl benötigt ein View-Referenzattribut. Dieses kann über die Methode des Interfaces registriert werden (Observer-Pattern).
- ControllerImpl referenziert ein Model-Objekt, das im Konstruktor übergeben und initialisiert wird.
- ViewImpl bleibt weiterhin Hauptklasse inklusive main-Methode
 - Sie referenziert ein Controller-Objekt
 - Im Konstruktor wird ein neues ModellImpl erzeugt und dem ControllerImpl beim Erzeugen zugewiesen
 - Die ViewImpl registriert sich selbst beim Model-Objekt

Programmierung der Interaktion und Logik



Zustandsänderungen mitteilen: Welche Zustände nimmt der Klickzähler an?

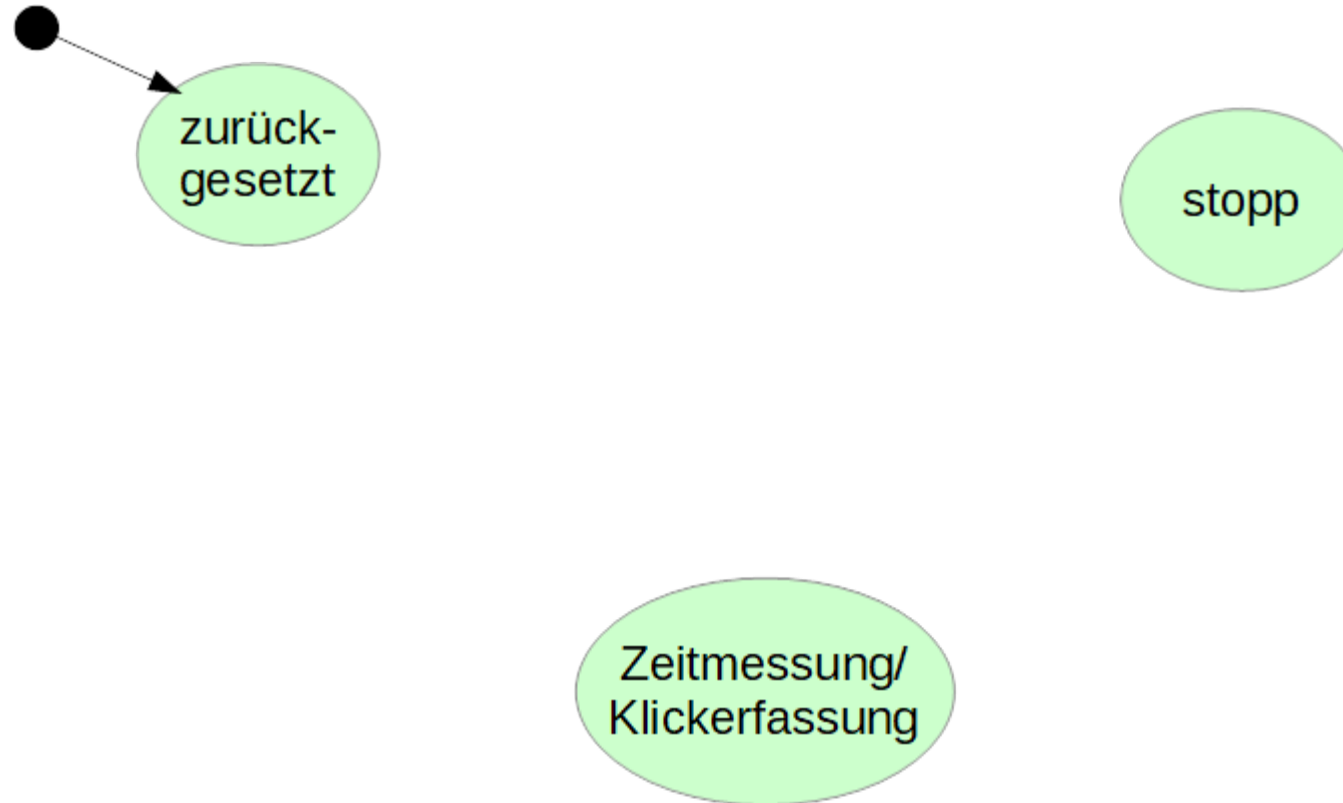
Zustandsänderungen mitteilen: Welche Zustände nimmt der Klickzähler an?

zurück-
gesetzt

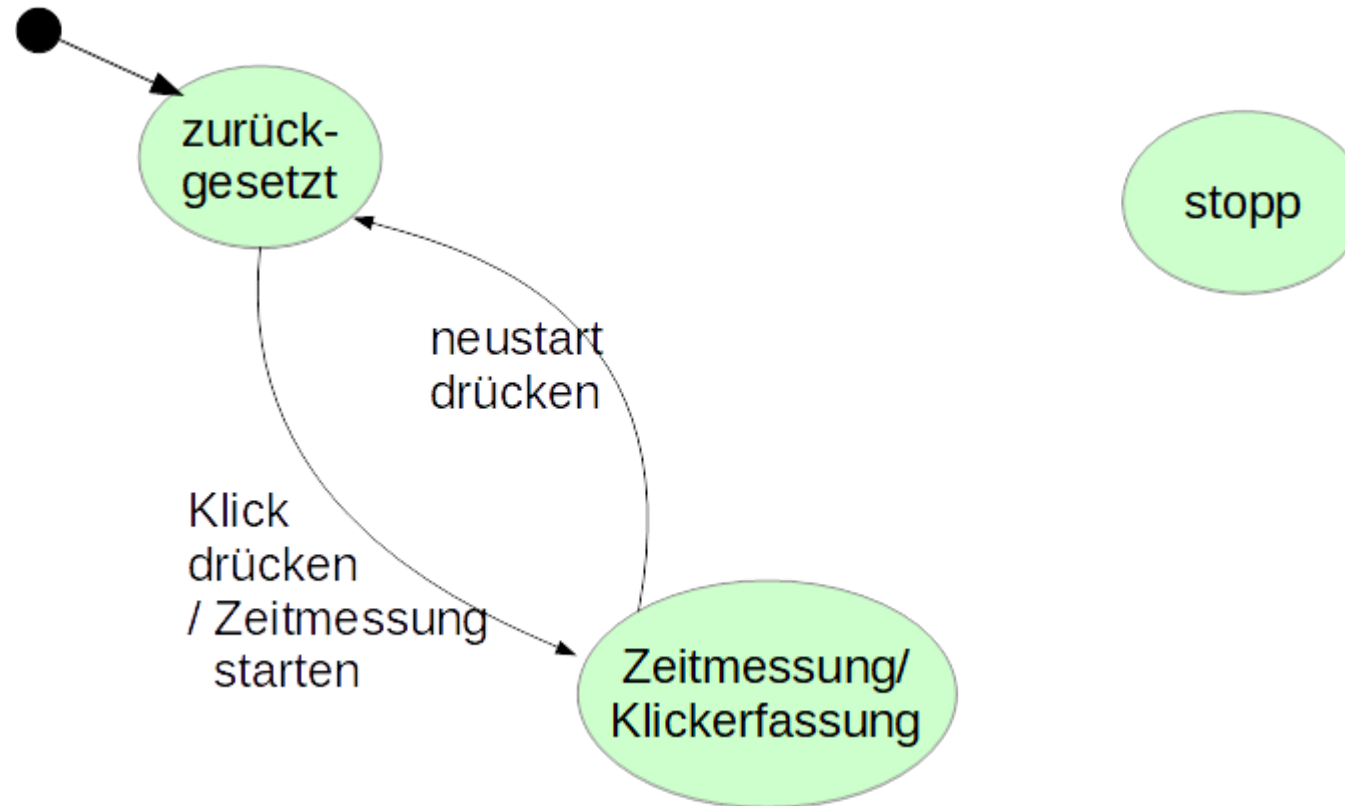
stopp

Zeitmessung/
Klickerfassung

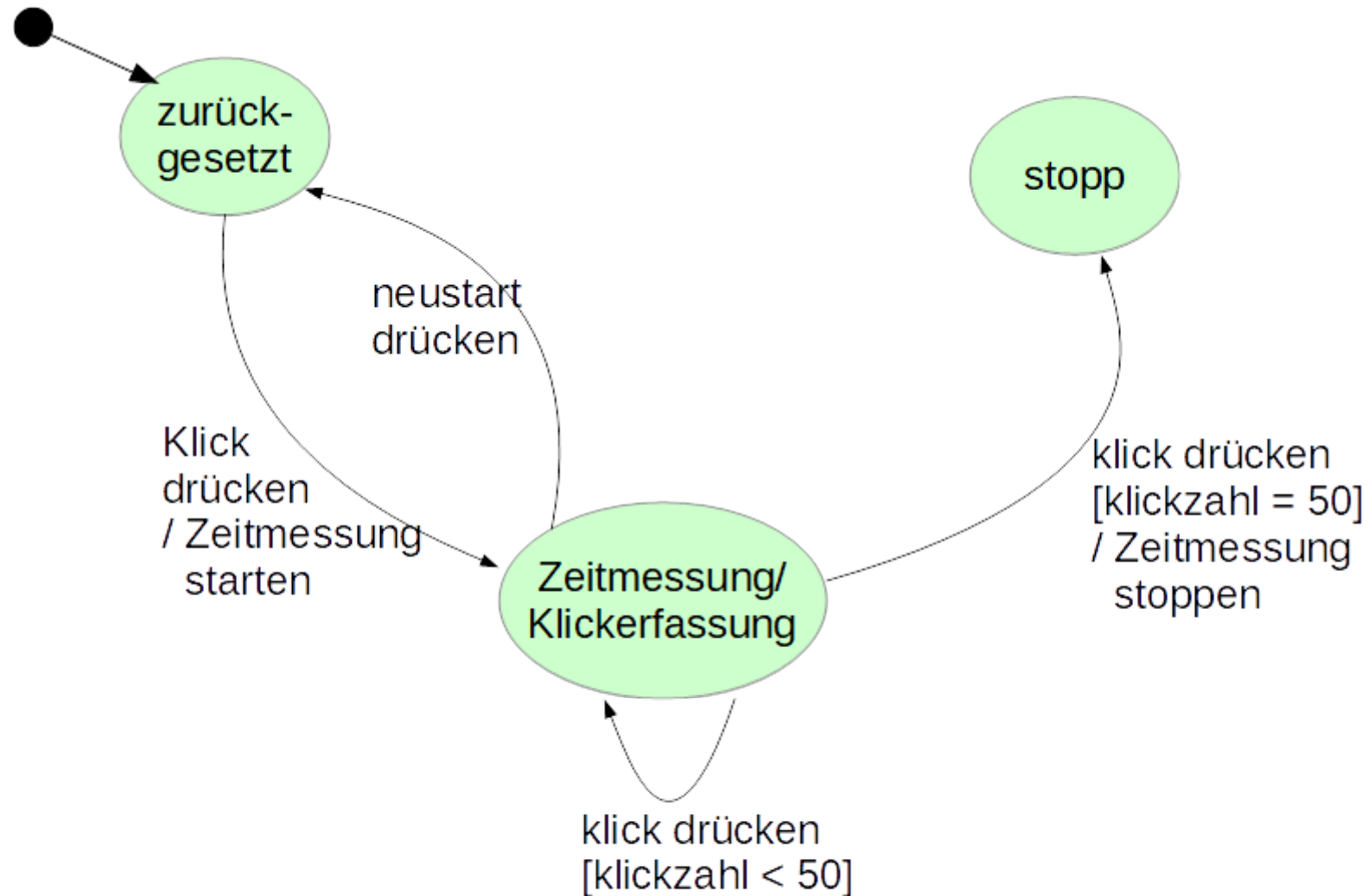
Zustandsänderungen mitteilen: Welche Zustände nimmt der Klickzähler an?



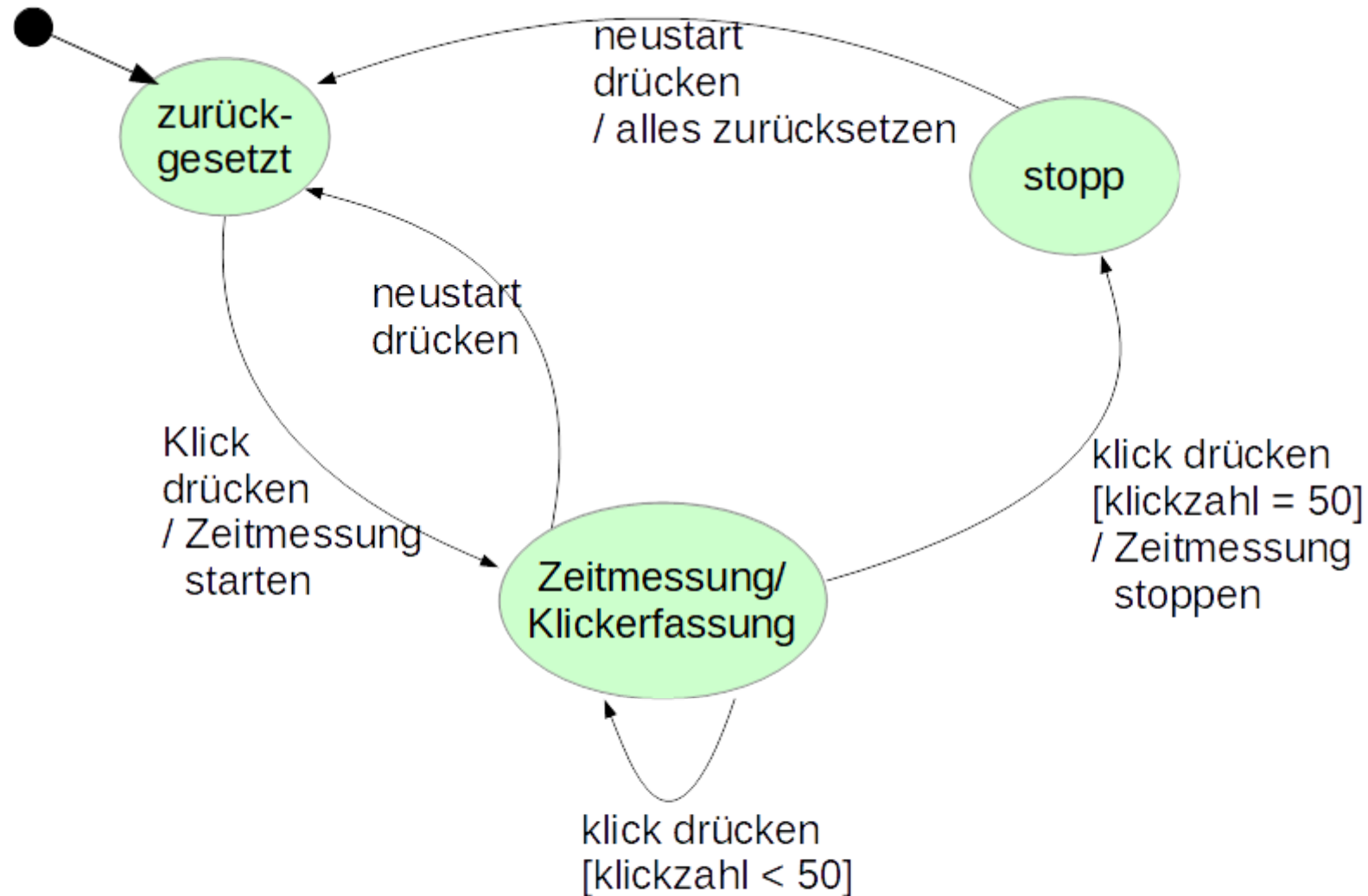
Zustandsänderungen mitteilen: Welche Zustände nimmt der Klickzähler an?



Zustandsänderungen mitteilen: Welche Zustände nimmt der Klickzähler an?



Zustandsänderungen mitteilen: Welche Zustände nimmt der Klickzähler an?



Hilfestellung zum Programmieren

- Das Model verwaltet die Logik des Programm. Es speichert den Zustand und die Variablen für den Ablauf des Programms.

```
private int klickzahl;  
private long startzeit;  
private long messzeit;  
/**  
 * 1: Ursprungszustand - alle Buttons aktiv, Aufforderung zum klicken  
 * 2: laufender Zustand (klickzahl < 50)  
 * 3: Messung beendet, noch nicht zurück gesetzt, Klick-Button inaktiv  
 */  
private int zustand;
```

- Implementiere anschließend gemäß dem Datenfluss:
View.actionPerformed() -> Controller.klick()/neustart() ->
Model.klick()/neustart() -> View.statusÄnderungEmpfangen()