

---

# Vorwort

Wie viele Artikel und Bücher habe ich schon gelesen, in denen im ersten Satz steht, dass Computersysteme immer komplexer werden oder Entwicklungszyklen immer kürzer und dass man deshalb von nun an Dinge besser/schneller machen muss als früher? Ich kann es nicht sagen. Diese Ausrede und Einleitung ist aber schon seit mehreren Jahrzehnten gültig, denn Techniken werden unentwegt weiterentwickelt, Technologien wandeln sich durch neue Einflüsse und die Zeit steht niemals still. Schon gar nicht rund um die Themen der Software-Entwicklung. Dieses Buch wurde nicht geschrieben, weil gerade jetzt alles komplexer wird und ich, als Autor, der Welt sagen will, wie man damit umgeht, sondern weil es im deutschsprachigen Raum noch kein brauchbares Buch zum Thema Test von Software eingebetteter Systeme gab und ich oft genug gesehen habe, wie schwer sich manche Leute damit tun.

## Praxisbezug des Buchs

Das Buch vermittelt praxisnahes Wissen zum Test von Software eingebetteter Systeme, so wie es jetzt und heute gemacht wird oder gemäß dem industriellen Stand der Technik gemacht werden sollte. Auch wenn der Stand der Wissenschaft und Forschung für das Schreiben des Buchs wichtig war, so nehmen forschungsnahe Themen nur wenig Platz ein, wenn deren industrielle Umsetzung Probleme bereitet. Das Buch präsentiert viele Beispiele, persönliche Erfahrungsberichte und enthält praxisnahe Fragen *und Lösungen* zum Selbsttest. Die Terminologie des Buchs orientiert sich weitgehend an der des ISTQB-Certified-Tester-Schemas und der Normenreihe ISO 29119. Das Buch ist dabei an vielen Stellen komplementär zum Curriculum des ISTQB, speziell dann, wenn es um Echtzeit, Concurrency und maschinennahe Themen geht, und es ist – unvermeidbar – zum Teil redundant dazu.

Software für eingebettete Systeme unterscheidet sich von anderer Software dadurch, dass die Software Teil des Produkts ist, das der Kunde kauft, und nicht das Produkt selbst. Meist wird das Produkt, in das ein Prozessor mit zugehöriger Software eingebettet ist, wertlos, wenn die Software nicht zuverlässig funktioniert. Im schlimmsten Fall muss es vom Markt genommen werden. Kein Wunder

also, dass man sich in dieser Sparte der Software-Entwicklung besonders viele Gedanken zur Korrektheit von Software macht.

Aber durch Testen alleine erhält man kein fehlerfreies Produkt. Wie ein alter Freund von mir zu sagen pflegte: »Man kann Software nicht am Ende der Produktentwicklung ›gesund testen‹, wenn es in so manchen Entwicklungsschritten zuvor krankt.« Das vorliegende Buch beschäftigt sich daher auch mit Praktiken links und rechts vom Test, die die Entwicklung begleiten und dazu beitragen, mit höherer Wahrscheinlichkeit »gesunde« Software zu erhalten.

### **Wie man dieses Buch liest**

Das Spektrum eingebetteter Systeme reicht von batteriebetriebenen 16-Bit-Controllern ohne Betriebssystem und mit geringen Anforderungen an die Systemintegrität bis hin zu im Internet Of Things vernetzten Multiprozessor-Systemen mit Echtzeitbetriebssystem, Mensch-Maschine-Schnittstelle und Sicherheitsrelevanz. Entsprechend viele Zielgruppen hat dieses Buch. Daher gibt es vermutlich Kapitel im Buch, die für Ihre speziellen Aufgaben irrelevant sein könnten. Sie dürfen diese Kapitel gerne überspringen, denn ich habe mir größte Mühe gegeben, jedes Kapitel weitgehend unabhängig von den anderen zu schreiben, und das Letzte, was dieses Buch tun sollte, ist, Sie zu langweilen. So können Systemtester zum Beispiel getrost die Kapitel über Reviews und statische Analyse auslassen. Die Unabhängigkeit der Kapitel macht es nötig, dass Inhalte teilweise wiederholt werden müssen.

Fans moderner Programmiersprachen sollten sich nicht daran stoßen, dass fast alle Beispielprogramme im Buch in der Programmiersprache C geschrieben sind. C ist weder komfortabel noch modern, doch hat C noch immer seinen festen Platz in der Entwicklung von Produkten mit höchsten Qualitätsanforderungen, und die meisten Entwickler verstehen die Basissyntax dieser Sprache ohne Probleme.

Ich wünsche Ihnen viel Freude beim Lesen und viel Erfolg bei der Umsetzung der in diesem Buch präsentierten Ideen.

Herzlichst

Stephan Grünfelder, Oktober 2016