

## Clean SAPUI5

Lesbarer und wartbarer SAPUI5-Code

» Hier geht's  
direkt  
zum Buch

# DAS VORWORT

# Einleitung

JavaScript ist eine sehr mächtige Sprache, aber diese Mächtigkeit bringt auch Probleme mit sich. Daher ist es für eine nachhaltige Entwicklung unerlässlich, eine Codebasis zu haben, die lesbar, wartbar und vor allem testbar ist. Das Buch »Clean Code: A Handbook of Agile Software Craftsmanship« von Robert C. Martin und mehrere andere Bücher enthalten viele Best Practices, die helfen, dieses Ziel in Java zu erreichen, aber viele der beschriebenen Konzepte sind unabhängig von der Programmiersprache. Sauberer Code ist ein Kernbestandteil der agilen Softwareentwicklung und stellt sicher, dass Sie langfristig agil bleiben und Ihre Projekte problemlos an sich ständig ändernde Anforderungen anpassen können.

Da die Zeit, die mit dem Lesen von Code verbracht wird, viel höher ist als die Zeit, die für das Schreiben desselben Codes benötigt wird, ist ein wichtiges Ziel, dass Ihr Code leicht zu lesen ist und klar vermittelt, was er zu erreichen versucht. Mit einem unermüdlichen Fokus auf Lesbarkeit wirkt sich sauberer Code letztlich auf die Wartbarkeit der Codebasis aus.

In diesem Buch beschreiben wir Regeln, Praktiken, Prinzipien, Muster, Richtlinien und Empfehlungen, die Ihren SAPUI5-/JavaScript-Code bereinigen – d. h. lesbar, verständlich, wartbar und testbar machen. In diesem Buch erläutern wir Codemuster und Idiome und behandeln übergeordnete Designüberlegungen. Wir werden auch Gewohnheiten, Prozesse und organisatorische Aspekte untersuchen, die beeinflussen, wie Sie als einzelne Entwicklerinnen und Entwickler und im Team Clean Code implementieren.

## Hinweis

Viele Entwicklerinnen und Entwickler wechseln zwischen Sprachen und Frameworks. Dieses Buch verwendet dort, wo es sinnvoll ist, die gleiche Sprache und Terminologie wie das Buch »Clean ABAP: Lesbarer und wartbarer ABAP-Code« (SAP PRESS, 2022, [sap-press.de/5403](https://sap-press.de/5403)). Leserinnen und Leser von »Clean ABAP: Lesbarer und wartbarer ABAP-Code« profitieren daher von einem einfacheren Lernprozess, der sich auf bekannte Konzepte konzentriert und dann die Unterschiede untersucht. Wenn Sie mit diesem Buch beginnen, können Sie die gemeinsame Struktur der beiden Bücher nutzen, um sich mit »Clean ABAP« vertraut zu machen. Viele der Konzepte in »Clean ABAP« sollten Ihnen bekannt sein, sodass Sie sich auf die Anwendung dieser Prinzipien auf ABAP konzentrieren können. Beachten Sie jedoch, dass es große Unterschiede zwischen ABAP und JavaScript gibt und dass es keine konzeptionelle Überschneidung zwischen den beiden Büchern gibt.



## Zielgruppe des Buchs

Dieses Buch ist eher technischer Natur und richtet sich hauptsächlich an SAPUI5-/JavaScript-Entwicklerinnen und -Entwickler. Wir haben viele Codebeispiele aufgenommen, um es Ihnen zu erleichtern, die Prinzipien und Praktiken zu übernehmen. Wir bieten ausreichend Erklärungen für Leserinnen und Leser, die noch auf dem Anfänger-Level sind, haben aber auch viele interessante Inhalte für Entwicklerinnen und -Entwickler auf Senior- bis Expert-Level aufgenommen. Abhängig von Ihrem Kenntnisstand können Sie von verschiedenen Themen profitieren oder sich auf bestimmte Schwerpunktbereiche konzentrieren. Die allgemeinen Ideen in diesem Buch richten sich jedoch an alle Entwicklerinnen und Entwickler, unabhängig von ihrer Erfahrung.

Wir empfehlen auch Softwarearchitektinnen und -architekten, dieses Buch zu lesen, um Clean SAPUI5 zu verstehen. Einige Themen in diesem Buch überschneiden sich mit der Softwarearchitektur (z. B. Modularisierung). Auch wenn Sie gar nicht programmieren, können Sie von den Konzepten in diesem Buch profitieren, einige dieser Praktiken lernen und anwenden. Insbesondere das letzte Kapitel ist nicht nur für Ingenieurinnen und Ingenieure relevant, sondern auch für Scrum Master und Manager, die sich mit der Team- und Organisationsentwicklung beschäftigen müssen.

In dieses Buch haben wir Themen aufgenommen, die Ihnen helfen sollen, jederzeit mit sauberem SAPUI5 in jeder Art von Projekt mit jeder Art von Team zu starten.

## Aufbau des Buchs

Dieses Buch beginnt mit einer Einführung in Clean SAPUI5 und einer allgemeinen Diskussion der JavaScript-Sprache und des SAPUI5-Frameworks in Kapitel 1 und Kapitel 2. Anschließend gehen wir in Kapitel 3 und Kapitel 4 sowie in Kapitel 5 auf die übergreifenden Themen Klassen, Module und Funktionen ein und diskutieren Clean-SAPUI5-Praktiken für diese übergeordneten Konstrukte. Das zentrale Thema der Namensgebung behandeln wir in Kapitel 6. Anschließend stellen wir in Kapitel 7 bis Kapitel 11 weitere Details zu spezifischen Clean-SAPUI5-Themen vor, von Variablen und Literalen bis hin zu Fehlerbehandlung und Kommentaren. In Kapitel 12 erfahren Sie mehr über statische Codeprüfungen und Codemetriken. Modultests und Integrationstests werden in Kapitel 13 erörtert, gefolgt von TypeScript und Technologien in Zusammenhang mit SAPUI5 in Kapitel 14. Schließlich erfahren Sie in Kapitel 15, wie Sie Clean SAPUI5 auf individueller, Team- oder Organisationsebene erlernen und implementieren können.

Kapitel 1 und Kapitel 2 sind wichtig, da sie den Kontext für das gesamte Buch liefern. Kapitel 4 und Kapitel 5 werden am besten zusammen gelesen, da der Themenbereich Klassen und Module natürlich in das Thema Funktionen übergeht. Die anderen Kapi-

tel sind in sich abgeschlossen und können in beliebiger Reihenfolge gelesen oder als Referenz verwendet werden. Achten Sie jedoch darauf, Abschnitt 1.2, »Erste Schritte mit Clean SAPUI5«, zu lesen, um zu erfahren, welche Themen zu welchem Zeitpunkt behandelt werden sollten. Kapitel 15 ist wichtig für alle, die bereit sind, Clean SAPUI5 auf allen Ebenen im Unternehmen zu lernen und zu implementieren.

Sehen wir uns kurz an, was in den einzelnen Kapiteln behandelt wird:

**Kapitel 1, »Einführung«**, stellt das Clean-SAPUI5-Konzept vor, beschreibt, warum es wichtig ist, Clean Code zu schreiben, und zeigt Ihnen, wie Sie damit beginnen können. Außerdem erfahren Sie mehr über andere Stilleitfäden und Lernressourcen, mit denen Sie Ihr Verständnis über dieses Buch hinaus erweitern können.

**Kapitel 2, »JavaScript und SAPUI5«**, stellt einige aktuelle Ergänzungen zu JavaScript vor und zeigt anhand konkreter Beispiele, wie diese neue Syntax Entwicklerinnen und Entwicklern dabei hilft, Clean-SAPUI5-Code zu schreiben.

In **Kapitel 3, »Projektstruktur«**, erfahren Sie, wie Sie ein Projekt für verschiedene Projektarten strukturieren.

Eine saubere und wartbare Architektur erfordert eine umfassende Trennung der Aspekte (*Separation of Concerns*) in Module und Klassen. In **Kapitel 4, »Module und Klassen«**, untersuchen wir zunächst, warum Controller-Module in SAPUI5-Anwendungen schnell wachsen. Anschließend schlagen wir Strategien für eine angemessene Entkopplung vor.

**Kapitel 5, »Funktionen«**, befasst sich mit JavaScript-Funktionen. Wir werden verschiedene Schlüsselkonzepte durchgehen, z. B. verschiedene Arten von Funktionen und Funktionsparametern und wie der Funktionskörper strukturiert wird. Außerdem besprechen wir Funktionsaufrufe, Promises und Closures.

**Kapitel 6, »Namensgebung«**, enthält Richtlinien und Empfehlungen zur Namensvergabe, einschließlich der Auswahl guter und konsistenter Namen, der Behandlung von Domänenbegriffen und der Verwendung von Präfixen und Suffixen.

In **Kapitel 7, »Variablen und Literale«**, erläutern wir, wie Sie Variablen und Literale deklarieren und verwenden, wie Scoping-Regeln funktionieren und wie Sie die Destrukturierung und die Spread-Syntax nutzen.

**Kapitel 8, »Kontrollfluss«**, erklärt den Kontrollfluss und gibt einige Clean-SAPUI5-Empfehlungen für die Verwendung verschiedener Arten von Kontrollflussanweisungen.

**Kapitel 9, »Fehlerbehandlung«**, erläutert die Fehlerbehandlung, warum sie notwendig ist und wie Fehler auf die Clean-SAPUI5-Art behandelt werden.

Viele Aspekte der Codeformatierung zielen darauf ab, dass Code, der von einem Team geschrieben wird, in seinem Design konsistent ist. In **Kapitel 10, »Formatierung«**,

stellen wir diese Aspekte vor, wobei wir die Bedeutung von guter Codeformatierung und Richtlinien für den Codestil hervorheben. Außerdem beschreiben wir, wie Sie die Formatierung durch moderne Werkzeuge erzwingen und automatisieren können.

**Kapitel 11**, »Kommentare«, befasst sich mit der Frage, ob, warum und wie Kommentare effektiv in Ihrem Code verwendet werden sollten. Gut platzierte Kommentare sollten das Verständnis des Codes verbessern und ihn nicht behindern.

Statische Codeanalysewerkzeuge können Ihnen helfen, die Codequalität objektiv zu analysieren und zu bewerten. In **Kapitel 12**, »Statische Codeprüfungen und Code-metriken«, stellen wir ESLint vor und beschreiben, wie der Linter Ihnen helfen kann, saubereren Code zu erstellen.

**Kapitel 13**, »Testen«, behandelt Testprinzipien und zeigt Ihnen, wie Sie Modul-, Komponenten- und Integrationstests für Ihren JavaScript-Code entwerfen und ausführen. SAPUI5 wurde im Dezember 2013 veröffentlicht und hat sich seitdem erheblich weiterentwickelt. Es wird eine weitere Evolution von SAPUI5 geben. **Kapitel 14**, »TypeScript und verwandte Technologien«, enthält die Motivation und weitere Informationen zu TypeScript, Webkomponenten und der Fundamental Library, einigen der wichtigsten und wirkungsvollsten Neuheiten im SAPUI5-Ökosystem.

In **Kapitel 15**, »Wie Sie Clean SAPUI5 umsetzen«, erläutern wir Herausforderungen und Ansätze für die Einführung in Clean-SAPUI5-Verfahren.

Um Sie auf wichtige Informationen hinzuweisen und Ihnen so die Arbeit mit diesem Buch zu erleichtern, verwenden wir im Text Kästen mit den folgenden Symbolen:

-  Mit diesem Symbol sind *Tipps* und *Hinweise* aus der Berufspraxis markiert, die praktische Empfehlungen geben, die Ihnen die Arbeit erleichtern können.
-  Dieses Symbol weist Sie auf *Besonderheiten* hin, die Sie beachten sollten. Es *warnt* Sie außerdem vor häufig gemachten Fehlern oder Problemen, die auftreten können.
-  In Kästen, die mit diesem Symbol gekennzeichnet sind, finden Sie Informationen zu *weiterführenden Themen* oder wichtigen Inhalten, die Sie sich merken sollten.

## Danksagung

Ich möchte mich bei Klaus Häuptle und allen anderen Mitautoren und Rezensenten dieses Buches für die Gelegenheit, die Zusammenarbeit und die lebhaften Diskussionen zu Clean SAPUI5 und Clean-Code-Themen im Allgemeinen bedanken. Dieses Buch wäre auch nicht möglich ohne die Unterstützung von SAP, mit ihrer Kultur, die das ständige Lernen fördert und die uns täglich zur Verbesserung auffordert. Schließlich wäre ich nicht dort, wo ich heute bin, ohne die monumentale Unterstützung der Menschen, die ich in meinem Leben am meisten liebe – meine Eltern, meine Söhne

und meine wunderbare Frau Nara. Du inspirierst mich jeden Tag, eine bessere Version von mir zu sein!

— Rodrigo Jordão

Ein Buch zu schreiben, steht schon seit geraumer Zeit auf meiner To-do-Liste. Aber ich hätte nie gedacht, dass es so früh in meinem Leben dazu kommen würde. Ich bin Klaus Häuptle dankbar, dass er mich zusammen mit meinen wunderbaren Mitautoren zu diesem Projekt eingeladen hat. Ich habe viel gelernt, als ich meine Seiten dieses Buches geschrieben habe. Ein großes Dankeschön geht an meine Eltern, die mich ermutigt haben, das zu verfolgen, woran ich glaube, und ein besonderer Dank geht an meine geliebte Frau Vinodhini, die mich auf dieser Reise unglaublich unterstützt und immer wieder motiviert hat.

Für die Leserinnen und Leser dieses Buches ist dies nur der Anfang, um Dinge mit einer neuen Perspektive zu tun. Teilen Sie Wissen, verbreiten Sie die Liebe. Frieden.

— Narendran N V

Die Einladung, an diesem Buch mitzuwirken und Wissen mit der Community zu teilen, ist sicherlich ein Grund, sich heute dankbar zu fühlen. SAPUI5 ist seit Jahren in aktiver Entwicklung und hat es Entwicklerinnen und Entwicklern ermöglicht, Geschäftsanwendungen bereitzustellen, indem sie sich nur auf das konzentrieren, was wirklich zählt. Unsere Verantwortung als Entwicklerinnen und Entwickler besteht jedoch weiterhin darin, nach dem Besten zu streben – für Kolleginnen und Kollegen, die in Zukunft von unserem Code lernen und ihn lesen oder ändern werden, und für Kunden, die unsere Arbeit in ihrer täglichen Arbeit nutzen werden. Wissen ist etwas, das uns niemand nehmen kann, und egal wie umfangreich unsere Erfahrung ist, es gibt immer etwas Neues zu lernen oder zu verbessern. Ein großer Dank geht an Megan Fuerst, Klaus Häuptle und meine Mitautoren für ihre Unterstützung und dafür, dass sie dies möglich gemacht haben. Danke auch an meine Familie und meine Lieben, die immer für mich da sind, in all unseren Momenten der Freude und der Anstrengung.

— Daniel Bertolozzi

Als ich zum ersten Mal kontaktiert und gefragt wurde, ob ich an einem Buch über Clean SAPUI5 mitarbeiten wolle, war ich erstaunt und dankbar, die talentierten Leute kennenzulernen, die ihr Wissen – oder sollte ich »ihre Weisheit« sagen – mit der SAPUI5-Community teilen wollten. Die SAPUI5-Teams haben große Anstrengungen unternommen, um ein umfassendes Framework bereitzustellen, mit dem sich Entwicklerinnen und Entwickler auf das Was konzentrieren können, anstatt auf das Wie. Obwohl es möglich ist, einen Nagel mit einem Schraubendreher in die Wand zu hauen, geht es in diesem Buch vor allem darum, mit den richtigen Richtlinien und

Ratschlägen die passenden Werkzeuge für die richtigen Aufgaben zu finden. Ich hoffe, dass Clean-Code-Liebhaber die Antworten auf ihre Fragen finden werden. Vielen Dank an meine Mitautoren für ihre Mitwirkung und Unterstützung, und ein großes Dankeschön an meine Frau Nausicaa und meinen Sohn Maxime, die mich während der Nachtschichten unterstützt haben.

— Arnaud Buchholz

Die Arbeit mit SAPUI5, der Testautomatisierung und der SAP Community ist schon seit einigen Jahren ein spannender Weg, und ich habe es sehr genossen, mein erworbenes Wissen in diesem Buch zusammenzustellen. Ich habe nie erwartet, dass ich durch das Schreiben eines Kapitels zu einem Buch beitragen würde, aber ich bin dankbar für die Gelegenheit, die mir gegeben wurde. Ein großer Dank an Megan Fuerst, Klaus Häuptle und meine Mitautoren für die Unterstützung während dieses Prozesses. Auch ein großes Dankeschön an meine Liebste, dafür, dass sie da war und mich diese ganze Zeit unterstützt hat.

— Christian Lehmann

Zunächst einmal muss ich meiner Partnerin, Verena, danken, die während dieses Prozesses als Freundin, Rezensentin und Partnerin unentbehrlich war. Zweitens ist das Schreiben von lesbarem, testbarem und wartbarem Code etwas, was einem professionellen Entwickler sehr am Herzen liegt. Danke an all die Entwicklerinnen und Entwickler, die sich um dieses Thema kümmern, in ihrer Codebasis anwenden, ihr Wissen mit anderen teilen und helfen, zukünftige professionelle Entwicklerinnen und Entwickler heranzuziehen. Ich möchte Florian Hoffmann besondere Dankbarkeit aussprechen, der die Gesamtarbeit an der Anwendung von Clean Code auf ABAP mit mir begonnen hat – ohne sein Engagement gäbe es *Clean ABAP* und *Clean SAPUI5* nicht. Außerdem wäre dieses Buch ohne die Unterstützung der großen Entwickler-Community, die das InnerSource-Clean-SAPUI5-Projekt unterstützt hat, nicht möglich gewesen. Die Diskussion um Clean SAPUI5 schuf ein gemeinsames Verständnis, auf dem das Buch aufbauen konnte. Darüber hinaus weiß ich die Unterstützung meiner Vorgesetzten Gregor Tielsch und Daniel Stephan von SAP Procurement sehr zu schätzen. Ein großes Dankeschön geht an meine Mitautoren Rodrigo Jordão, Christian Lehmann, Arnaud Buchholz, Daniel Bertolozzi und Narendran N V, die so viel Zeit und Hingabe in das Buch investiert haben. Ich habe diese vertrauensvolle und inspirierende Zusammenarbeit immer gewürdigt. Ich möchte mich bei den Rezensenten Peter Muessig, Andreas Kunz und Damian Lion Maring bedanken. Ihr Feedback war äußerst wertvoll. Dank Megan Fuerst von SAP PRESS, die uns bei diesem Buchprojekt unterstützt und uns bei all unseren Fragen geholfen hat.

— Klaus Häuptle

## Fazit

Wenn Sie dieses Buch lesen, erfahren Sie alles über wichtige Idiome, Muster, Praktiken und Prinzipien, die Ihnen dabei helfen, Ihren SAPUI5/JavaScript-Code wartbar, lesbar und testbar zu machen. Sie erfahren, warum es wichtig ist, sich um sauberen Code zu kümmern, was er im Kontext Ihrer SAPUI5- und JavaScript-Projekte bedeutet und wie Sie ihn anwenden. Da es sich bei der Anwendung von Clean Code um eine Teamaktivität handelt, beschreibt das letzte Kapitel mehrere Ansätze, diese Konzepte in Ihr Team einzuführen und eine kontinuierliche Verbesserung als Team zu fördern.