

► chefredakteur



Dokumentation ist ein traditionell ungeliebter Bereich der Softwareentwicklung – zumindest sehen das die meisten Entwickler so. 40 Jahre lang hat man ihnen eingebläut, die Qualität ihrer Software könne man an der Qualität der Dokumentation erkennen, aber diese Vorhaltungen haben kaum zu mehr geführt als einem schlechten Gewissen. Dabei wissen die meisten Entwickler eine gute Dokumentation zu schätzen – nämlich die fremder Software, die sie selbst nutzen möchten.

Dokumentation ist bei vielen Softwaresysteme unabdingbar für ihren sicheren Einsatz. Aber wie bei jedem Dokument gilt auch hier, dass Dokumentation für einen bestimmten Leserkreis geschrieben ist, der bestimmte Bedürfnisse hat: sowohl bei der Information als auch bei deren Aufbereitung. Schlechte Dokumentation krankt in der Regel daran, dass Zielgruppe und Aufbereitung nicht beachtet wurden. Entwickler erwarten zum Beispiel Konzepte und Beispielcode in der Dokumentation von Bibliotheken, nicht nur Tautologien á la „getX() – diese Methode holt X“ und Hinweise zur Bedienung der Maus.

Gute Dokumentation wird heute häufig von technischen Autoren geschrieben, die ebenso dafür ausgebildet wurden, Dokumentationen zu schreiben, wie Entwickler ausgebildet wurden, Software zu schreiben. Dass deutlich mehr dazu gehört als „etwas mehr Anstrengung“ demonstrieren Ihnen die Autoren der Schwerpunkt-Artikel in dieser Ausgabe von OBJEKTSpektrum.

Mit „Stille und Rauschen in der Dokumentation“ beschäftigt sich **Jan-Christian Krause**: Stille ist Information, die vorhanden sein sollte, die man aber vergebens sucht, während Rauschen für überflüssige Information steht, die das Verständnis eher stört als fördert. An einem Beispiel aus der „eBay Trading API“ zeigt er, wie man beides systematisch vermeiden kann.

Eine Serie von Beiträgen zur Dokumentation in agilen Teams eröffnet **Uwe Friedrichsen** mit seinem Artikel „Was muss, was kann, was geht gar nicht?“. Er plädiert für eine klare Zielorientierung bei der Dokumentation und für die strikte Trennung zwischen Projekt- und Produktdokumentation. Während Projektdokumentation vermieden wird, wo es möglich ist, schlägt Friedrichsen für die Produktdokumentation vor, nach Stakeholdern zu differenzieren: früh feststellen, wer welche Art von Dokumentation braucht, und daraus die richtige Dokumentation ermitteln.

Mit der Integration der Dokumentationserstellung in den Entwicklungsprozess beschäftigt sich **Andreas Rüping** in seinem Artikel „Von agilen Verfahren lernen“. Auch er plädiert dafür, nur die Dokumentation zu schreiben, die wirklich notwendig ist. Insbesondere beschäftigt Rüping sich aber mit der Frage, *wann* die Dokumentation sinnvollerweise geschrieben werden sollte. Geht man sie zu früh an, riskiert man wegen der evolutionären Natur agiler Verfahren erhöhte Aufwände, geht man sie zu spät an, liegt sie möglicherweise nicht mehr rechtzeitig vor.

Bei erhöhten Anforderungen an Qualität oder Umfang der Dokumentation übernehmen in der Regel technische Redakteure das Ruder. **Marc Beckers** beschäftigt sich in seinem Beitrag „Darf’s ein bisschen weniger sein?“ mit der Integration technischer Redakteure in ein Scrum-Team. Je besser der Informationsfluss zwischen Entwicklern und Redakteuren ist, umso besser ist auch das Ergebnis. Agile Verfahren bieten hier optimale Möglichkeiten – vorausgesetzt, die Teams arbeiten gut zusammen und verfallen nicht in Zuständigkeitsdenken.

Mit der „Dokumentation von technischen Schnittstellen mit Wikis“ beschäftigt sich **Ulrike Parson** in ihrem Artikel. „Wikis bieten eine ideale Plattform, um gemeinsam Wissen zu erarbeiten und weiterzugeben [...]. Aber ohne Pflege verkommt auch das schönste Wiki und endet als Informationshalde“, schreibt sie. Um ein solch trauriges Ende zu vermeiden, bietet sie sieben Schritte an, die eine gute Zusammenarbeit zwischen Entwicklern und Redakteuren sicherstellen.

Mit einem ganz anderen Thema setzen sich im letzten Schwerpunkt-Beitrag **Michael Engler, Jens Palluch** und **Jörg Schmidt** unter dem Titel „Gesund geschrieben: Normenkonforme Dokumentation medizinischer Software“ auseinander. Um solche Software überhaupt auf den Markt bringen zu dürfen, muss die Dokumentation einer Reihe von Normen genügen – je nach Einsatzart und anvisiertem Markt. Es braucht schon Experten wie die Autoren, um sich in diesem Dschungel zurechtzufinden.

Ihr

Jens Coldewey