
Ludewig Lichter Software Engineering

Software Engineering
Funktionale Sicherheit nach ISO 26262
Softwareevolution
Ich bin dann mal still
Workshops im Requirements Engineering
Software Engineering - kompakt
Technische Unterstützungssysteme
Software Engineering
Prototypen in der Softwareentwicklung
Modernes Software Engineering
Requirements-Engineering und -Management
UML @ Classroom
Auswahl und Einführung eines Vorgehensmodells für die Softwareentwicklung
Alles in Butter
Software-Engineering
Software Engineering für verteilte Anwendungen
Software Engineering
Grundkurs Informatik
Analyse und Design mit der UML 2.5
Die Software-Krise und ihre Macher
Software engineering
Benutzerorientierte Software-Entwicklung
Security by Design. Security Engineering informationstechnischer Systeme
Software-Engineering
Einführung in Software Engineering
Integrierte Software Engineering Services zur effizienten Unterstützung von Entwicklungsprojekten
Software Engineering
Software Engineering
Best Practice Software-Engineering
Digitalisierung
Studienarbeiten
Programmieren lernen von Kopf bis Fuß
Lehrbuch der Softwaretechnik: Basiskonzepte und Requirements Engineering
Fallstudie Software-Engineering. Kernkonzept und Veranschaulichung
Software Engineering
Modernes Software Engineering
Entwurfsprinzipien und Konstruktionskonzepte der Softwaretechnik
Grundlagen der Mensch-Computer-Interaktion
Software Engineering

MADDEN LISA

Springer-Verlag

Essay aus dem Jahr 2021 im Fachbereich Informatik - Software, Note: 1,7, FOM Hochschule für Oekonomie und Management gemeinnützige GmbH, Hochschulstudienzentrum Hamburg, Veranstaltung: Softwareentwicklung, Sprache: Deutsch, Abstract: Den Gedanken die Entwicklung eines Produktes, von der Idee bis zur Fertigstellung, zu unterstützen, gibt es nicht erst seit Software entwickelt wird. Speziell im Bereich des Maschinenbaus ist es etabliert und allgemein bekannt, dass mit Prototypen gearbeitet wird. In der Automobilbranche werden die Prototypen genutzt, um das Auto vollumfänglich zu testen. Dabei wird betrachtet, ob alle Funktionalitäten wie angedacht arbeiten, bevor das Produkt in Produktion geht und an Kunden verkauft wird. Dies ermöglicht das Aufdecken von unerwarteten Problemen, die beim Entwurf und der Planung des Produktes unter Umständen nicht bedacht wurden. Ebenso besteht im Rahmen der Softwareentwicklung das Risiko, dass das fertige Programm am eigentlichen Bedarf vorbei entwickelt wurde. Um solch ein Risiko zu minimieren, werden Prototypen unter anderem in diesem Bereich eingesetzt. Als Vorteil gilt zudem, dass kritische Funktionen bereits frühzeitig erkannt und durch einen Prototypen realitätsnah getestet werden können. Durch die oft aufgezeigten Vorteile wird in dieser Arbeit folgende Forschungsfrage beleuchtet: „Ist der Einsatz von Prototypen generell zu empfehlen und können die Versprechungen gehalten

werden oder existieren auch relevante Nachteile und Herausforderungen bei der Verwendung von Prototypen?“
Software Engineering Mosaik Verlag
Dieses Lehrbuch vermittelt die Grundlagen der objektorientierten Modellierung anhand von UML und bietet eine kompakte Einführung in die fünf Diagramme Klassendiagramm, Anwendungsfalldiagramm, Zustandsdiagramm, Sequenzdiagramm und Aktivitätsdiagramm. Diese decken die wesentlichen Konzepte ab, die für die durchgängige objektorientierte Modellierung in einem kompletten Softwareentwicklungsprozess benötigt werden. Besonderer Wert wird auf die Verdeutlichung des Zusammenspiels unterschiedlicher Diagramme gelegt. Die präsentierten Konzepte werden anhand von illustrativen Beispielen erklärt.
Funktionale Sicherheit nach ISO 26262
dpunkt.verlag
Softwareevolution bedeutet Wartung plus Weiterentwicklung eines bestehenden Systems. In den bestehenden Systemen steckt die akkumulierte Erfahrung eines Unternehmens und die Arbeit mehrerer Personen über viele Jahre. Das Buch unterstreicht den immensen Wert bestehender Softwaresysteme und die Notwendigkeit, sie zu bewahren. Sie müssen ständig ausgebaut und regelmäßig renoviert werden. Das alles verlangt nach anderen Techniken und Methoden als bei der Entwicklung eines neuen Systems. Die Autoren behandeln in diesem Grundlagenwerk Themen wie Wartungs- und Wiederaufbereitungsprozesse, Wiederverwendung, Softwareanalysemethoden, Reverse Engineering, Nachdokumentation und Wirtschaftlichkeitsaspekte der Softwaresystemerhaltung. Auch auf

Aspekte bei agilen Entwicklungsprojekten wird eingegangen. Die Kapitel des Buches sind nach den Tätigkeiten in einem Softwareevolutionsprozess gegliedert. Aus dem Inhalt: - Die Gesetze der Softwareevolution - Der Evolutionsprozess - Softwaresystemanalyse - Softwareevolutionsplanung - Fehlerbehebung - Änderungen - Sanierung - Softwareweiterentwicklung - Systemregressionstest Der Stoff, der in diesem Buch zusammengefasst ist, basiert auf mehr als 30 Jahren Erfahrung des Hauptautors Harry Sneed in der Wartung und Weiterentwicklung bestehender Software.

Softwareevolution Hanser Verlag
Dieses Buch führt Sie in die wichtigsten Entwurfsprinzipien und Konzepte der Softwaretechnik ein Entwurfsprinzipien, auch Design Patterns, sind bewährte, einfache und klare Denkkonzepte des Software Engineering, welche Entwicklern helfen, hochwertige Softwaresysteme zu konstruieren. Sie greifen in die Konstruktion eines Systems ein und betreffen die für den Entwickler sichtbare Qualität des Quellcodes. Qualitätsziele für den Entwurf sind beispielsweise: · das Vermeiden unerwünschter Kopplungen von Softwareteilen · die Einfachheit und Verständlichkeit · die Testbarkeit · die Stabilität bei Programmweiterungen
Dieses Buch führt Sie in die Grundlagen der Softwaretechnik ein und stellt Ihnen die wichtigsten Entwurfsprinzipien und Konstruktionskonzepte vor. Es gibt keine allgemein anerkannten Kataloge von Entwurfsprinzipien für die Softwareentwicklung. Daher enthält dieses Buch eine Auswahl von praxisrelevanten Entwurfsprinzipien und Konzepten, die zudem in der Clean-

Code-Bewegung eine große Rolle spielen. Das Buch eignet sich nicht nur für die tägliche Arbeit in Bereich der Softwaretechnik, sondern auch als Lehrbuch für Studierende der Informatik. Die folgenden Inhalte behandelt der Autor In seinem Buch beschäftigt sich Joachim Goll unter anderem mit den folgenden Thematiken: · Das Konzept der Software-Module · Unerwünschte Abhängigkeiten und deren Abschwächung in Programmen · Erzeugung einer modularen Struktur in Programmen · Reduktion der Komplexität von Programmen · Konstruktion schwach gekoppelter Teilsysteme · Korrektheit von polymorphen Programmen · Stabilität von Programmen und Erweiterbarkeit bei Programmänderungen · Inversion of Control

Ich bin dann mal still Springer-Verlag
Keine ausführliche Beschreibung für "Einführung in Software Engineering" verfügbar.

Workshops im Requirements Engineering
dpunkt.verlag

Die Beiträge im vorliegenden Buch stammen von Praktikern der Objektorientierung, die ihren Erfahrungsschatz dokumentieren und anderen Anwendern zugänglich machen. Dabei wird vor allem auf Themen eingegangen, die in Grundlagenbüchern nicht oder nur am Rande behandelt werden, wie z.B. der Entwurf und die Dokumentation von Softwarearchitekturen, Aspekte des Prozess- und Projektmanagements, Geschäftsmodellierung im Kontext objektorientierter Modelle und die Verwendung prädikativer Techniken.

Software Engineering - kompakt
Springer-Verlag

Software wird auch heute im wesentlichen noch so hergestellt wie vor

20 Jahren, die erzielten Produktivitäts- und Qualitätsverbesserungen sind eher bescheiden. Deshalb und wegen des allgemein beklagten "Anwendungsstaus" - der Bedarf an neuer Anwendungssoftware übersteigt bei weitem die Verfügbarkeit - spricht man von einer weltweiten Software-Krise. Hinzu kommt, da insbesondere die deutsche Software-Industrie bisher wenig Erfolg in der weltweiten Vermarktung ihrer Produktion hat. Entgegen gängiger Expertenmeinung macht der Autor deutlich, da die vielzitierte Software-Krise vor allem auf Unzulänglichkeiten und Nachlässigkeiten in der Entwicklung von Software beruht, die durchaus überwindbar sind: Die Schwächen der heutigen Softwaretechnologien wiegen in ihrer Wirkung weit geringer als die bei den Beteiligten bestehenden psychologischen Barrieren gegen ihren Einsatz. Die Bewältigung der Softwareprobleme hängt somit stärker von einer allgemeinen Änderung des Bewusstseins ab als von technologischen Fortschritten. Mit diesem Buch sollen deshalb insbesondere die Entscheidungsträger in der Industrie angesprochen werden. Es wendet sich aber auch an eine breitere Öffentlichkeit, um für die bevorstehende öffentliche Diskussion Sachverstand zu vermitteln. Nicht zuletzt richtet es sich natürlich an die Softwareentwickler selbst und versetzt sie in die Lage, ihre Rolle in der vermeintlichen Krise richtig einzuschätzen.

Technische Unterstützungssysteme Springer-Verlag

Das Buch ist als Lehrbuch abgefaßt und beinhaltet zu jedem Abschnitt Übungsaufgaben sowie (im Anhang) einige Lösungshinweise. Ausgehend von den elementaren Grundkenntnissen im

Software Engineering, wie strukturierte Programmierung, Modularisierung usw. wird am Beispiel der objektorientierten Analyse, dem Design und der Implementation (im Smalltalk und C++) ein vollständiger Softwarelebenszyklus mit Metriken bewertet und beschrieben. *Software Engineering* dpunkt.verlag
Dieses Buch behandelt alle Aspekte des funktionalen Sicherheitsmanagements und beschreibt die Anforderungen der ISO 26262 im Detail. Es wird nicht nur dargestellt, was in der Norm gefordert wird, sondern auch wie die Anforderungen erfüllt werden können. Dies geschieht anhand eines durchgängigen Praxisbeispiels aus dem Automotive-Bereich. Umfangreiche Umsetzungsbeispiele, hilfreiche Vorlagen und praktische Anwendungstipps begleiten den Leser durch alle behandelten Phasen des Sicherheitslebenszyklus und fördern das Verständnis für den Aufbau eines funktionalen Sicherheitsmanagements. *Prototypen in der Softwareentwicklung* MITP-Verlags GmbH & Co. KG
Mit dem vorliegenden Werk wird ein transdisziplinärer und partizipativer Ansatz zur bedarfsorientierten Technikentwicklung beschrieben. Ausgehend von der Vorstellung, dass der Mensch auch in Zukunft in vielen Bereichen durch Technik weder ersetzt werden kann noch sich ersetzt zu werden wünscht, richtet sich das Werk an Wissenschaftler und Praktiker gleichermaßen. Die gesellschaftliche Akzeptanz technischer Unterstützungssysteme stellt den Kern und Ausgangspunkt des Buches dar. Neben den Grundlagen und Ansätzen für die Entwicklung von Unterstützungssystemen werden im Buch innovative Technologien und Anwendungsbeispiele akzeptierter

Systeme vorgestellt.

Modernes Software Engineering GRIN Verlag

Im Software-Engineering geht es um die Modellierung und Entwicklung komplexer, qualitativ hochwertiger Software und die für einen erfolgreich durchgeführten Realisierungsprozess geeigneten Methoden, Werkzeuge und Standards. In diesem kompakten Lehrbuch werden die wichtigsten Themen rund um Software-Engineering erklärt, zusammengefasst und mit kleinen Praxisbeispielen vertieft. Von zentraler Bedeutung für das Software-Engineering ist der Software-Lebenszyklus. Gemeint ist damit der gesamte Prozess, der zur Erstellung und Erhaltung eines Softwaresystems führt. Sowohl in traditionellen als auch in agilen Softwareerstellungsprozessen läuft dieser Lebenszyklus ab. Bewährt hat sich in der Praxis die Einteilung in sogenannte Phasen, denen die Gliederung folgt. Nach einer kurzen Einführung werden in Kapitel 2 vorab phasenübergreifende Verfahren wie divergierende Vorgehensmodelle und Projektmanagement besprochen. Kapitel 3 behandelt die Planungsphase; Kapitel 4 ist dem Requirements-Engineering gewidmet, bei dem die Software-Anforderungen kreativ konstruiert, analysiert und – traditionell oder agil – dokumentiert werden. In Kapitel 5 folgt die Besprechung der Verfahren für die Designphase der Software. Hier wird hinterfragt, wie gute Software-Architekturen Erfolg versprechend erdacht, mit der UML-Notation geeignet modelliert und in späteren Projekten wiederverwendet werden können. Kapitel 6 widmet sich der Test- und Abnahmephase und damit den wichtigen Qualitätssicherungsfragen. Abschließend wird in Kapitel 7 die Wartung – zur

wirksamen Erhaltung von Softwaresystemen – erklärt. Anfänger erhalten eine schnelle Orientierung und kompaktes, fundiertes Grundwissen. Fortgeschrittene Leser finden hier ein aktuelles, gut strukturiertes Nachschlagewerk. Unter <https://www.hanser-fachbuch.de/buch/Software+Engineering+kompakt/9783446459496> finden interessierte Leser weitere Übungsaufgaben zum Thema Software-Engineering.

Requirements-Engineering und -Management Springer-Verlag

Essay aus dem Jahr 2020 im Fachbereich Informatik - Software, Note: 2,0, FOM Hochschule für Oekonomie & Management gemeinnützige GmbH, Köln, Sprache: Deutsch, Abstract: In dieser Arbeit werden Anforderungen mittels eines Use Case Diagramms dargestellt und anschließend formell beschrieben. Anschließend erfolgt die Modellierung des beispielhaften Use Cases als Klassen- und Aktivitätendiagramm. Außerdem wird ein erster Entwurf der Architektur des zu entwickelnden Software Systems dargestellt. Zum Abschluss dieser Arbeit gehört ein Fazit und ein Ausblick auf das, was folgen kann. Viele Unternehmen entdecken zurzeit die Agile Entwicklung für sich. In dieser ist es üblich, die in dieser Arbeit dargestellten Vorgehensweise nicht zu berücksichtigen. In der agilen Entwicklung wird auf schnelle, inkrementelle Auslieferung von kleinen Softwarepaketen Wert gelegt. D.h., dass große Analysen und Anforderungserhebungen nicht im Vorfeld stattfinden. Dabei können Anforderungen, die im ersten Augenblick nicht offensichtlich sind, vergessen werden. In großen Projekten wird dieses Vorgehen wohl kaum umzusetzen sein.

Damit Fehler vermieden werden, sollte an der strukturierten Analyse festgehalten werden.

UML @ Classroom vdf Hochschulverlag AG

- Praktische und innovative Lösungen für die Anforderungsanalyse sichern den Know-how-Transfer in Ihrem Entwicklungsvorhaben. - Erlernen Sie das Ermitteln, Vermitteln, Herleiten und Verwalten von qualitativ hochwertigen Anforderungen. - Meistern Sie Ihre Anforderungen in agilen Frameworks sowie in klassischen Vorgehensweisen. - Tauchen Sie ein in die Welt der Smart Ecosystems. - Lernen Sie das Zusammenspiel von Anforderungen und Architektur im Systems-Engineering kennen. Der Erfolg von Systementwicklungen entscheidet sich bereits in der Anforderungsanalyse! Sie ist das Fundament für viele weitere Tätigkeiten. Dieses Buch liefert Ihnen Hintergründe, Strategien, klare Konzepte und umfangreiche Praxistipps zur pragmatischen Umsetzung Ihrer Anforderungen - von der Erhebung bis hin zur Verwaltung. Als neue Themen werden in der 7. Auflage Requirements-Engineering im agilen Umfeld, Systems-Engineering und Smart Ecosystems betrachtet. Zusätzlich bietet diese Auflage Einblicke in den Einsatz von Videos im Requirements-Engineering, Crowd-RE und die Besonderheiten im Variantenmanagement. Durch die Buchkapitel begleiten Sie ein durchgehendes Beispiel mit einer eigenen Rahmenhandlung und eine von Kapitel zu Kapitel aufbauende Bauanleitung für einen Requirements-Engineering-Leitfaden. Im Internet finden Sie unter www.sophist.de/re7 zusätzliche Formulare, Checklisten, Hintergrundinformationen und vieles mehr. AUS DEM INHALT //

Vorgehensweisen klassisch und agil/Anforderungsermittlung/SOPHIST-Regelwerk/Anforderungsschablonen/Anforderungsanalyse/Geschäftsprozesse/Systems-Engineering/Smart Ecosystems (Industrie 4.0)/Anforderungsdokumentation/klassisch und agil/Nichtfunktionale Anforderungen/Prüftechniken für Anforderungen/Anforderungskonsolidierung/Requirements-Management, Change- & Release-Management/Einführungsstrategien/Produktlinien und Produktfamilien/Videos im Requirements-Engineering/Requirements-Engineering mit der Crowd
Auswahl und Einführung eines Vorgehensmodells für die Softwareentwicklung dpunkt.verlag
 Das Handbuch fürs Selbststudium, für den Job oder vorlesungsbegleitend erfahrungsbasierter Über- und Einblick ins Software Engineering, der sowohl die Theorie als auch die Praxis abdeckt umfassend, verständlich und praxiserprobt Das Buch vermittelt die Grundlagen, Erfahrungen und Techniken, die den Kern des Software Engineerings bilden. Es ist als Material zu Vorlesungen über Software Engineering konzipiert. Auch für Praktiker, die mit der Softwareentwicklung und -bearbeitung und den dabei auftretenden Problemen vertraut sind, ist das Buch sehr gut geeignet, um die Kenntnisse im Selbststudium zu ergänzen und zu vertiefen. Der Inhalt des Buches ist in fünf Hauptteile gegliedert: - Grundlagen - Menschen und Prozesse - Daueraufgaben im Softwareprojekt - Techniken der Softwarebearbeitung - Verwaltung und Erhaltung von Software Auch auf die Ausbildung zukünftiger Software Engineers wird eingegangen. Ergänzende Informationen sind auf der

Webseite der Autoren verfügbar:
<https://se-buch.de>.

Alles in Butter Carl Hanser Verlag GmbH
 Co KG

Was lernen Sie in diesem Buch? Es ist kein Geheimnis, dass die Welt um Sie herum immer computerbasierter, vernetzter, konfigurier- und programmierbarer wird. Sie können passiv daran teilhaben – oder Sie können lernen zu programmieren. In diesem Buch erfahren Sie, wie Sie Code schreiben, mit dem Sie Ihren Computer, Ihr Mobilgerät oder alles, was sich programmieren lässt, dazu bringen, Dinge für Sie zu tun. Mit der Programmiersprache Python erarbeiten Sie sich Schritt für Schritt die Grundkonzepte der Programmierung sowie viele grundlegende Themen der Informatik wie Speicherung, Datenstrukturen, Abstraktion, Rekursion oder Modularität. Warum sieht dieses Buch so anders aus? In dieses Buch sind die neuesten Erkenntnisse der Kognitionsforschung und der Lerntheorie eingeflossen, um Ihnen das Lernen so einfach wie möglich zu machen. Statt einschläfernder Bleiwüsten verwendet dieses Buch eine Vielzahl von Abbildungen und Textstilen, die Ihr Gehirn auf Trab halten und Ihnen das Wissen direkt ins Hirn spielen – und zwar so, dass es sitzt.

Software-Engineering Springer-Verlag
 Der Koch, der sein Herz auf der Zunge trägt Jetzt ist es so weit – das wahre Horst-Lichter-Kochbuch kommt! Kaum ein Koch ist im deutschen Fernsehen in den letzten Jahren so populär geworden wie Horst Lichter. Ob als Gastkoch oder als Moderator – der Mann mit dem Bart hat die rheinische Frohnatur in unsere Küchen gebracht. Wo Lichter kocht, wird gelacht. Sein Geheimnis sind sein Humor und seine Leidenschaft für Deftiges:

Lange hat kein Spitzenkoch so offen von Butter und Sahne geschwärmt und Hausmannskost sowie vergessene Klassiker der deutschen Küche auf unsere Teller zurückgeholt. »Der Horst kann viel mehr, als er uns manchmal zeigt, der weiß genau, wie man auf höchstem Niveau kocht!« sagt über ihn kein geringerer als Sternekoch Alfons Schuhbeck. Alles in Butter ist ein wundervolles Kochbuch und mehr als das: In Kapiteln wie »Aus dem Nähkästchen«, »Klassiker einmal anders« oder »Was ich heimlich von Johann Lafer gelernt habe« serviert Horst Lichter skurrile Storys aus seinem Leben, Sprüche zum Schmunzeln, Hintergründiges über Zutaten und Zubereitung, Tipps und Tricks. • Das wahre Horst-Lichter-Kochbuch mit vielen neuen Rezepten und exklusiven Fotos • Klassiker, mal ganz anders und zwar so, dass man sie wirklich nachkochen kann und keine Tellermalerei! • ... das wird so sein wie ich, wie man mich kennt: mit Anekdotchen, mit Geschichten und mit Blödsinn! • Tausend tolle Geheimnisse aus der Welt der Küche

Software Engineering für verteilte Anwendungen Walter de Gruyter GmbH & Co KG

Literaturverz. S. 394 - 397. - Angek. u.d.T. "Analyse und Design mit der UML 2.4.1"

Software Engineering Carl Hanser Verlag GmbH Co KG

Ziel dieses Leitfadens ist die Vermittlung von Regeln und Techniken für die Durchführung wissenschaftlicher Arbeiten. Dazu gehören Seminar-, Bachelor-, Master- und Doktorarbeiten, also alle Prüfungsleistungen mit wissenschaftlichem Anspruch, die von Studierenden, Assistentinnen und Assistenten unter Anleitung, aber in gewisser Selbständigkeit und über

längere Zeit hinweg erbracht werden. Im Mittelpunkt stehen Bachelor- und Masterarbeiten, doch auch auf Publikationen und auf Dokumentationen ohne wissenschaftlichen Anspruch lassen sich die meisten Aussagen anwenden. Der Leitfaden ist für Arbeiten in allen technischen und naturwissenschaftlichen Disziplinen hilfreich, die Informatik dient lediglich als Beispiel. Er wendet sich einerseits an Studierende, die lernen wollen, ein Projekt auszuwählen, vorzubereiten, durchzuführen und zu präsentieren. Andererseits sind die Betreuerinnen und Betreuer angesprochen, die die Arbeiten definieren, unterstützen und schließlich beurteilen.

Grundkurs Informatik Software Engineering

Horst Lichter, Deutschlands beliebtester TV-Koch und Moderator, begibt sich auf eine Reise in die Stille und zu sich selbst: Ein Lebensratgeber voller Ruhe, Kraft und Entschleunigung, aber auch voller Humor, Augenzwinkern und Lebensfreude. Die Sehnsucht nach einem Moment des Innehaltens in einer Welt, die uns alles abverlangt, die Stress und Burnout, Überforderung und Anstrengung mit sich bringt, kennen wir alle. Schweigen, still sein, einatmen, ausatmen, in der Meditation ganz für sich sein – dieses Abenteuer, diese Selbstfindung wagt der Mann, der nie um einen flotten Spruch verlegen ist. Was macht die plötzliche Stille mit ihm? Die Pause vom schnellen Leben? Ist Meditation der Schlüssel zum Glück? Und was hat es eigentlich mit diesem Buddhismus auf sich? Horst Lichter erzählt auf sympathische und humorvolle Weise, was ihn zum Schritt in die Stille bewogen hat, warum er sich die Pause vom Alltagsstress gönnt. Er berichtet von der Zeit der Ruhe und was

passiert, wenn man plötzlich allein in seinem Kopf ist. Wenn die laute Welt durch die eigenen Gedanken ersetzt wird. Welche Fragen man sich selbst stellt und auch selbst beantworten muss. Und er verrät, was am Ende bleibt von dieser Erfahrung. Wie es gelingt, seine Kraft und innere Mitte zu bewahren, wenn man wieder zurück im Alltagsstress ist. Dieses Buch ist ein spannendes Experiment, berührend, inspirierend und zum Schmunzeln, aber auch echte Lebenshilfe für den Umgang mit der Überforderung und der Belastung, die wir alle täglich spüren.

Analyse und Design mit der UML 2.5 dpunkt.verlag

Software-Komponenten tragen durch einen hohen Grad an Wiederverwendbarkeit, bessere Testbarkeit und Wartbarkeit zur effizienten Herstellung komplexer Software-Anwendungen bei. Diese Vorteile bedingen jedoch oft eine aufwendigere Einarbeitung beim Einstieg in diese Materie durch die Vielzahl an komplexen Komponenten-Frameworks, Werkzeugen und Entwurfsansätzen. Das vorliegende Buch „Best-Practice Software Engineering“ bietet Neu- und Wiedereinsteigern in die komponentenorientierte Software-Entwicklung eine Einführung in die Materie durch eine abgestimmte Zusammenstellung von praxiserprobten Konzepten, Techniken und Werkzeugen für alle Aspekte eines erfolgreichen Projekts. Für moderne Software-Entwicklung sind eine Vielzahl von unterschiedlichen Fähigkeiten erforderlich, die nur im richtiger Kombination zu einem erfolgreichen Ergebnis führen. Daher wird in diesem Buch besonderer Wert darauf gelegt, nicht einzelne Techniken des Software Engineerings isoliert zu betrachten,

sondern das effiziente Zusammenspiel verschiedener Aspekte darzustellen. Schwerpunkte liegen auf Vorgehensstrategien im Software-Lebenszyklus, Projektmanagement, Qualitätssicherung, UML-Modellierung, Entwurfsmustern und Architekturen, komponentenorientierter Software-

Entwicklung sowie ausgewählten Techniken und Werkzeugen. Zu den Beispielen im Buch finden Sie den vollständigen Source Code sowie umfangreiche Fallbeispiele zu Artefakten aus dem Projektverlauf auf der Webseite zum Buch.

Related with Ludwig Lichter Software Engineering:

[© Ludwig Lichter Software Engineering The Lion King And Hamlet Comparison Worksheet](#)

[© Ludwig Lichter Software Engineering The Law Of Timing](#)

[© Ludwig Lichter Software Engineering The Living Environment Prentice Hall Brief Review Answer Key](#)