

Business Patterns for Software Developers

Allan Kelly

*ebooks | Download PDF | *ePub | DOC | audiobook*



[Download](#)

[Read Online](#)

#1558106 in eBooks 2012-04-10 2012-04-10 File Name: B007U2ZT7K | File size: 68.Mb

Allan Kelly : Business Patterns for Software Developers before purchasing it in order to gage whether or not it would be worth my time, and all praised Business Patterns for Software Developers:

1 of 1 people found the following review helpful. Developers, yes, but for all of you By AcetoMr. Kelly begins his book on Business Patterns with a survey of software company types. This approach struck me oddly because his title (if it is in fact his choice) BUSINESS PATTERNS FOR SOFTWARE DEVELOPERS suggests a technical work on recurring patterns that developers (a.k.a. programmers, software engineers, etc.) can employ in designing and writing code. This book is for developers, to be sure. But it ought have a wider audience. Everybody in a software company, and many in business units will benefit from this book. I tried to think of some role that would not. I could not. The evolution of software development describes an arc that has moved from the singular creation of software for a single purpose to the notion of a real product, with all that entails. In my experience, we are so good at developing crack software that we ignore its larger context as a product. Foreshadowing further, products sit in the context of the entire business world and tradition. Hang on just a bit if I have not turned you away. Developers, or at least their keepers, are forever talking about the SDLC. Good as it is, the software development life-cycle is not to be confused with the

product life-cycle, and neither with the life-cycle of companies. So Mr. Kelly yanks us back to consider the larger context of a software company. So, instead of being project centric, he expands the scope to product centric. Later he will spiral again to the context of running a company. He is looking for analogous patterns at each level of the spiral. Finance jerks always want to express the larger problem of what to do with software products as maximizing immediate returns. They always have new and better jobs after you are already dead. Finance champs all know that the area under the curve is the whole life of the company. That curve is the only non-destructive maximization. The rest is grift. Most start-ups that do not fail quickly do not go past the initial phase of the early success attributed to a good idea. It may sometimes be given something resembling money. But, rather than using this initial success to enable, and more importantly to adjust to the next juicy market, the natural, uninformed tendency is to do more of the same, thereby missing the boat. The perfect balance is the dynamic dance between innovative risk and mundane financial muddling. By the time Mr. Kelly gets to his discussion of strategy, it seems that developers are nearly always better assets when they know what their firm is doing, and thereby they align, however tangentially, to the company goals: "Without shared understanding between individuals of what the company strategy is and its intentions are, there is risk of no coordination between apartments, teams and individuals: each runs in a different direction and then there is little overall forward movement" Strategy, it turns out, is not only intentional, it emerges from the forces of the past. Therefore, it is foolish to embark on a strategy that you think is a good idea but divorced from the past. As emergent strategy cannot successfully drive the future, it does offer the real context to ground a real strategy forged from the intentional and the emergent. He refers us to Mintzberg from his "Strategy as a pattern; a consistent pattern over time. "Strategy is not the plan; it is the thinking behind the plan. It follows that, if patterns exist, they have an inductive aspect to them. Inductive reasoning of pattern progression can be a powerful analytical tool, but it is not enough to effect a brake through. In another curious analogy to pure software development, he says, "...as strategy is implemented it must allow for emergent properties." Formal change control is an important discipline if the product is to fruit. Agile development needs agility strategy, if you will excuse the methodological pun. Mr. Kelly provides a basic survey of funding start-ups. Ye who dance with Venture Capitalists best learn the steps before the music begins. If you play Faust, you better know how high you will be made to jump at each round of VC funding. Next comes an extended discussion about classes or categories of products, and of how they live in the marketplace. He moves away from strictly software companies, and for good reason. Lifecycle patterns are easier to see when software people are not just looking in their own yard. Better to get out and see that product types, their characteristics and attributes, without the unintentional distraction caused by the familiar. Mr. Kelly is a superior pattern analyst and a superior strategist. Surely as night follows day, marketing follows product in the developmental context (as opposed to the mature company). Thusly are going concerns built from start-up software companies. To move from a generic, or core product to an optimized company that grows nicely, there are more aspects to the grand strategy, or, as he names it, "the whole product". Sound like quite a lot for one book, and one that is but 300 pages. We have come a long way, yet something is missing. The customer has gone missing. Just as in product strategy, where he is loudly against the temptation to divert vital company energy into engineering one-offs, so he is likewise bold with customer focus: ASSUME ALL CUSTOMERS ARE SIMILAR It is not that you are not customer oriented, but that your product management does not dilute itself over functionality not common to your target customers. Get to market with your product and learn how to improve while growing over time. Once you are in the marketplace, it is time to consider distribution models, which in turn shape sales strategy. He favors account management by pairs of business with technical. Only in the final third does he tackle the tricky world of services - hard to manage; hard to track - and the hardest to sustain. The bankruptcy courts are strewn with the charred corpses of successful product companies that let this tail wag the dog and ruin the purse. Finally, he returns to his opening theme, pattern, for a summary yet philosophical discussion. Again, he turns to his architect mentor, Christopher Alexander. His "The Nature of Order" is a worthwhile study for the ambitious reader. This book is for all sorts of technology people, from those in start-ups to those wanting a make-over, from skunk-works to seasoned developers. For a senior manager, save money on consulting fees. This book is a guard against their invasive capture of your vision and your money, and a good companion to your strategic planning. This book is a good guide for managers at all levels, whether writing a business plan or developing staff. Use this book to plan your year as a small part of your staff meetings. Assign chapters or sections, perhaps on a monthly schedule, or at least a quarterly one. They make for good team working sessions. Align and conquer. 3 of 3 people found the following review helpful. Wow, the best reference set I have seen for people working on business issues. By Mark P. McDonald The title "Business patterns for software developers" sounds academic, technical and dry as a bone. But once you get past the title, this book is a uniquely structured and presentation of business design and operations I have found in a long time. Based on the idea of design patterns, from Christopher Alexander, this book provides a reference tour de force of business information and knowledge. This book is so much more than a software book and in fact the software parts of it are not the real value here. The real value starts with the patterns -- chapter 7 and how they are structured. You do not need the first 124 pages of this book, particularly if you are a business person to get extreme value from this book. The chapters containing the patterns are diverse and detailed covering the following business areas: Marketing Distribution Models Direct distribution Indirect

Distribution Services (Professional, customer service, self service, etc) Product organizations Patterns provide a succinct and clear language for representing complex ideas in common ways. Understanding how to use a pattern and this book becomes a great reference as well as education resource. Each of the patterns is presented in the following format: Description Context Problem Forces Solution Consequences Variations Related work Examples Each of these characteristics enable the authors to explain business activities, processes and goals in no more than three or four pages. It provides enough information to get you started without too much information that artificially limits your design or innovation. This book is **HIGHLY RECOMMENDED** for Consultants, IT professionals, Business Process Designers or anyone who has to learn a new area or wants to understand how business activities work. This is not a book you will read cover to cover, but neither is Alexanders two pattern books. Rather this is an active reference book, one that you will dip into, explore and learn from. It provides a great check to see if you are considering the right solution, covering all your bases, etc. 0 of 0 people found the following review helpful. **Incisive Guide to Business Strategy for Software Developers** By Jessica Weissman I started in software development long before the patterns concept became popular. The software and design patterns I've seen often look like clever names for standard and obvious ideas or ways of doing things...and maybe that's what they are. Anyway, this book has finally made me see what use patterns can be. Allan Kelly has done a superb job of laying out the forms of business strategy and methods of operation available to software companies. Developers who have been around a while will recognize the life cycles, marketing strategies, types of support, product development choices and more of the companies we've been involved with. If you've ever wondered why the company doesn't just make the software better instead of selling professional services, Kelly explains it. Yes, the bosses did it on purpose. They also skimmed on the documentation on purpose. Kelly not only describes the patterns, he explains them clearly and makes sense of the reasoning behind them. If you're a developer, this book can help you understand what the marketers and and distribution people and product managers and operations people are up to. If you are a developer who wants to move into the business end of things, this book is as good as an MBA (except for the details of finance and personnel management and all that generic stuff you also need). As a bonus, Kelly includes a section defining and explaining what patterns themselves are. I can't recommend this book highly enough for its target audience. It does what any book of its type should do: organizes knowledge and understanding and experience you already have, puts it together, gives concepts names, and makes the whole thing make sense. One caution: Kelly is British, so some of his terminology reflects British rather than American usage. "Poacher turned gamekeeper" makes a lot more sense in an English context than in an American one, but you'll understand the pattern anyway.

A must-have recipe book for building software Perhaps you can relate to this all-too common scenario: you know all about your software product?but could do with some help in understanding the strategic side of things. If so, this book is the one-stop resource you'll need in order to become a successful software entrepreneur. Patterns expert Allan Kelly provides you with the step-by-step route that needs to be followed in order to understand business strategy and operations. Each chapter starts out with a solid introduction and theoretical overview, which is then further illustrated with patterns and case studies, all aimed at helping you move into the management of software. Teaches you the ropes of business strategy and operations for software Places special emphasis on the patterns for those who make software for sale Addresses patterns philosophy, patterns strategies, business strategy patterns, and software company lifecycle Shares practical tools, tips, and examples of best practices so you can see how each specific pattern fits in and needs to be implemented. **Business Patterns for Software Development** divulges strategies, operations, and structures for building successful software.

About the Author Allan Kelly has seen the software industry from many different angles - has held just about every job in the software world and has worked with just about every type of company there is. As a result he has seen the many different ways of doing business the industry has, and seen many of the same ways repeated.