

Obsah

KAPITOLA 1

Úvod

První zkušenosti z praxe	1
Založení vlastní firmy	2
Co najdete v této knize	2
Rozdělení obsahu a témat	3

ČÁST I

Tajemství tvorby softwaru 5

KAPITOLA 2

Zvláštnosti vývoje softwaru 7

Dodržení časového harmonogramu	7
Vývoj softwaru a informační ekonomika	8
Vývoj softwaru je především tvůrčí práce	9
Způsob použití informací a zkušeností	9
Spisovatelská a novinářská činnost	10
Aplikovaný výzkum	10
Projektové práce	10
Vysoká kvalifikace a práce v týmu jsou základem	11
Obecná pravidla pro úspěšný vývoj softwaru	11
Postup není vždy tou nejlepší odměnou	12

KAPITOLA 3

Příčiny problémů softwarových projektů 15

Nejčastější problémy	16
Problém první – ZPOŽDĚNÍ	16
Problém druhý – VYSOKÁ CHYBOVOST	16
Problém třetí – NESPLNĚNÍ POŽADOVANÉ FUNKČNOSTI	17
Problém čtvrtý – NEDOSTATEČNÁ VÝKONNOST	17
Problém pátý – SLOŽITÉ UŽIVATELSKÉ ROZHRANÍ	17
Problém šestý – OBTÍŽNÁ UDRŽOVATELNOST PROGRAMU	18
Základní příčiny problémů	18
Příčina první – PODCENĚNÍ PROJEKTU	19
Příčina druhá – ŠPATNÉ ZADÁNÍ	19
Příčina třetí – NEDOSTATEČNÁ ANALÝZA	20

Příčina čtvrtá – PŘÍLIŠNÁ SLOŽITOST PROJEKTU	20
Příčina pátá – PŘEHNANÝ DŮRAZ NA TECHNOLOGII	21
Příčina šestá – ŠPATNÁ KVALITA PROGRAMOVÉHO KÓDU	21
Příčina sedmá – NEVHODNÉ METODIKY, POSTUPY, TECHNOLOGIE	21
Příčina osmá – NEDOSTATEČNÉ TESTOVÁNÍ	22
Příčina devátá – ŠPATNÉ PROJEKTOVÉ ŘÍZENÍ	22

KAPITOLA 4

Poučení z historie 23

Počítačový pravěk	23
Rozvoj v 60. letech 20. století	23
Úsvit softwarového inženýrství	25
Základní metodika práce – metoda vodopádu	25
Rozšiřování programů v „pravěku“	26
Dnešní pohled na vývoj softwaru	27
Snadnost ovládnání a rychlost uvedení na trh	27

KAPITOLA 5

Nedostatky školního vzdělání 31

Vstup ze školy do praxe	31
Problémy začínají...	31
Znalosti je třeba doplnit praxí	32
Příliš technické zaměření	32
Většinou chybí metodika zaškolení nováčka	33
Vysoké školy kladou důraz na exaktní řešení	33
Programování NENÍ matematická disciplína	34
Co by měl softwarový inženýr také absolvovat	34
Základy psychologie pro práci v týmu	35
Základy ekonomie	35
Praktické nedostatky začínajících programátorů	35
Absence zkušeností řešení běžných situací	35
Neznalost procedur při vývoji softwaru	36
Podcenění základních metodik	36

ČÁST II

Jak psát a jak nepsat programy 37

KAPITOLA 6

Proč psát kód čitelně 39

Rychlost neznamená vyšší zkušenost	39
Absence komentáře jako nástroj zajištění nenahraditelnosti	40
Nedostatek času jako příčina nečitelnosti	41
Komentovaný zdrojový kód je nejlepší dokumentací	41

Problematické typy v týmu – „slabý článek“	222
Problematické typy v týmu – „dělnická mentalita“	223
Problematické typy v týmu – „snaživců“	223
Vhodná velikost týmu	223
Požadavky na řízení týmu	224
Složení týmu	225
Postava jedna – manažer produktu	225
Postava dvě – vedoucí projektu	226
Postava tři – technický leader, architekt	228
Postava čtyři – databázový specialista	228
Postava pět – analytik	229
Postava šest – designér	229
Postava sedm – programátor	230
Postava osm – tester	230
Postava devět – tvůrce dokumentace	230
Výběr členů týmu	231
Kritérium první – výběr podle schopnosti splnit úkol	231
Kritérium druhé – schopnost komunikace	232
Kritérium třetí – profesionalita a pečlivost	232
Kritérium čtvrté – dostatečný technický přehled	232
KAPITOLA 31	
Vedení vývojového týmu	235
Nutnost řízení týmu	235
Vedení týmu místo řízení	235
Zadávání úkolů a jejich kontrola	236
Písemné zadání úkolu	236
Oznámení o ukončení úkolu	237
Pravidelná kontrola postupu	237
Motivace členů týmu	238
Odměna při dosažení milníků	238
Různé druhy odměn	238
Řešení problematických situací	238
Komunikace uvnitř týmu	239
Sdílení zkušeností v týmu	239
Ostré diskuse a hádky v týmu	240
KAPITOLA 32	
Komunikace s okolím	241
Komunikace se zákazníkem	241
Úrovně komunikace se zákazníkem	242
Písemné zápisy komunikace	242
Situace při změně konkrétní osoby	242
Podpora projektu ve vedení zadavatele	243

Problémy při jednání s klienty	243
Zpoždění nebo nedosažení výsledků	244
Nepříznivý dojem z průběhu	244
Nepříznivý dojem z první verze	244
Změny strategie kvůli změně ve vedení zadavatele	245
Změny nad rámec původního plánu a rozpočtu	246
Jednání s nadřízenými	246
Interní jednání	247
Problém „střet civilizací“	247
Jednání s manažery	248
Bod první – manažeři většinou nerozumí technickým otázkám	248
Bod druhý – manažeři předpokládají, že přeháníte	248
Bod třetí – doplňte písemné zápisy z jednání	249
Bod čtvrtý – nepočítejte s logikou nebo základní aritmetikou	249
Bod pátý – neposkytujte irelevantní negativní informace	250

ČÁST VIII

Metodiky a technologie 251

KAPITOLA 33

Metodiky 253

Zaměření a rozsah metodik	254
Šíření metodik	254
Zaměření metodik na technickou část	254
CASE nástroje	255
Příklady metodik	255
Metodika jedna – Microsoft Solution Framework	255
Metodika dvě – extrémní programování	256
Metodika tři – Rational Unified Process	256
Slabiny metodik	257
Najímání programátorů v rámci projektu	257
Obecnost metodik	258
Metodika je vodící nití projektu	258
Certifikace	258
Typy certifikací	259

KAPITOLA 34

Vizuální modelování 261

Výhody vizuálního modelování	261
Diagramy	262
UML	262
Diagramy UML jsou intuitivní	263
Nástroje pro UML	263

Nevýhody UML	263
Jak začít s vizuálním modelováním	264
Pokročilý uživatel UML	264
Modelování nenahrazuje dokumentaci	265

KAPITOLA 35

Návrh struktury aplikace **267**

Rozdělení aplikace na části	267
Rozdělení většího týmu na menší skupiny	268
Rozdělení projektu na přesně definované komponenty	268
Jak aplikaci rozdělit	269
Rozdělení aplikace do tří vrstev	270
Datová vrstva	270
Vrstva aplikační, vrstva obchodních pravidel	270
Prezentační vrstva	270
Úskalí návrhu architektury	272
Obtížně viditelné přehnané požadavky	272
Experimenty v architektuře jsou nebezpečné	273

KAPITOLA 36

Použitá technologie, nástroje a komponenty **275**

Ani vývojářské nástroje nejsou bez chyb	275
Nové technologie	276
Problémy s kompatibilitou nové technologie	276
Volba programovacího jazyka	277
První požadavek – detailní znalost programátorů	277
Druhý požadavek – splnění požadavků	277
Třetí požadavek – vyšší úroveň abstrakce	277
Používání knihoven	278
Cizí knihovny mají svá úskalí	278
Vytvoření samostatné komponenty použitelné pro další projekty	278

ČÁST IX

Architektura aplikace **281**

KAPITOLA 37

Výkonnost aplikace **283**

Výkonnostní kritéria	283
Kritérium první – počet operací za časovou jednotku	284
Kritérium druhé – doba odezvy	284
Kritérium třetí – škálovatelnost propustnosti aplikace	284
Plánování výkonnosti	285

Propustnost proti škálovatelnosti	285
Stanovení výkonnostních požadavků	285
Vytipování slabých míst	286
Vliv datového modelu a použitých algoritmů	286
Obecné řešení může být na úkor výkonnosti	286
Škálovatelnost	287
1. Využití více počítačů	287
2. Rozdělení zátěže specializací počítačů	287
3. Určení priorit při zpracování úkolů	288
4. Spojení počítačů do clusteru	288
5. Programátorská pravidla pro zajištění škálovatelnosti	288
Kešování	288
6. Vhodné algoritmy	289
7. Vzájemné blokování zpracovávaných požadavků	289
Metody databázového zamykání	289
Příklad zpracování požadavku se zamykáním	289
Aplikace s více vlákny	290
Odezva	290
Zobrazení postupu operace	291
Výkonnostní testy	291
Identifikování výkonnostního problému	292
Hledání zúžených míst	292
Testování výkonnosti ve zkušebním provozu	293
KAPITOLA 38	
Návrh uživatelského rozhraní	295
Konzultace rozhraní s budoucími uživateli	295
Postup při návrhu rozhraní	296
Znalost způsobu práce s daty	296
Pravidla pro návrh uživatelského rozhraní	296
Optimalizace pro všechny uživatele	297
Intuitivní a ergonomické ovládání	297
Volba obsahu pro hlavní obrazovky	298
Snadnost ovládání	298
Grafický návrh	299
Konzistence rozhraní	299
Snadné pochopení celé aplikace	299
KAPITOLA 39	
Práce s databází	301
Role databáze v projektu	301
Databáze pro malé projekty	301
Návrh databáze	302
Výhody uložených procedur	303

Nevýhody uložených procedur	303
Zámky a vzájemné blokování uživatelů	303
Pravidla pro zamykání	304
Provoz a údržba databáze	304
Automatické testování struktury databáze	305
Zálohování databáze	305
Údržba databáze	305
KAPITOLA 40	
Rozhraní a protokoly	307
Chyby v komunikaci	307
Definice protokolu	308
Možnost snadného rozšiřování protokolu	308
Odolnost protokolu vůči chybám	308
Postup při implementaci	309
Co dělat v případě chyby	309
Testovací modul pro komunikaci	309
Změny v protokolu na jedné ze stran	310
ČÁST X	
Když projekt běží...	311
KAPITOLA 41	
Průběh vývoje	313
Plánování a řízení v každé fázi	313
Udržování projektového plánu	313
Pozor na „skoro hotové“ úkoly	314
Minimalizace rizik v průběhu realizace	314
Pravidelné verzování	314
Pravidelná komunikace s klientem	315
Zpoždění	315
Přesčasová práce a její limity	315
Cílová prémie	316
Posílení týmu	316
Omezení vedlejších činností	317
O zpoždění je třeba co nejdříve informovat	317
KAPITOLA 42	
Nasazení a údržba	319
Bod první: instalace aplikace do systému zákazníka	319
Bod druhý: překážka v podobě hardwaru nebo systémového softwaru	320
Bod třetí: dobrá spolupráce se správcem systému	320
Bod čtvrtý: Chyby mimo aplikaci a jejich řešení	321

Údržba a úpravy	321
Změny musí projít normálním procesem	322
Všechny změny musí být řádně otestovány	322
KAPITOLA 43	
Zavádění pravidel a postupů	323
Dohoda a kompromis	323
Volba vhodného pravidla záleží na situaci	324
Kontrola dodržování pravidel	324
PŘÍLOHA 1	
Příklad testu pro uchazeče na pozici programátora	325
Rejstřík	331

Čitelnost kódu je přímo spojena s životností aplikace	41
Komunikace v týmu šetří dvojí práci	42
Čitelnost kódu je nejen pro vývojáře	42
Nekomentovaná funkční část se často stává „černou skříňkou“	43
KAPITOLA 7	
Pravidla pro přehledný zápis kódu	45
Základní pravidla čitelného zápisu	46
Čtení bez přemýšlení	47
Identifikátory a jmenné konvence	50
KAPITOLA 8	
Pravidla pro psaní komentářů	53
Účelem komentáře je usnadnění práce	53
Co je třeba komentovat	54
Komentář složitějších algoritmů	55
Hlavičky modulů a procedur	59
Komentování změn v kódu	60
KAPITOLA 9	
Jasná struktura kódu a dat	61
Vývoj programovacích jazyků	61
Nástup objektového programování	62
Vizuální programování	62
Základní pravidla	63
Dobře napsaná procedura	66
Komunikace procedury s okolím	67
Ošetření chybových stavů	67
Reakce procedury na výjimky	70
Datové struktury	72
Reference místo ukazatelů	72
Uvolňování paměti po ukončení životnosti objektů	74
Metody „uklizení“ paměti	75
Přetečení bufferů	76
Objektový model	76
Třídy objektového modelu	77
Návrh hierarchie dědičnosti tříd	77
Moduly a globální proměnné	77
KAPITOLA 10	
Programátorské triky	79
Co je vlastně trik?	79

Proč se triky používají?	79
Příklady triků	80
Jak s triky zacházet	81
KAPITOLA 11	
Vstupní data	83
Data importovaná z externích systémů	83
Protokol přenášných dat	84
Změny v protokolu a možné důsledky	84
Reakce na chybná data	85
Data zadaná uživatelem	85
Ověření správnosti uživatelského vstupu	86
Problémem mohou být i různá časová pásma	87
<hr/>	
ČÁST III	
Proč dochází k chybám?	89
KAPITOLA 12	
Bezpečnostní díry	91
Bezpečnost závisí na architektuře systému	91
Jak je možné se „nabourat“ do aplikace	92
Přihlašování uživatelů	92
Přihlašovací modul a šifrování	92
Pravidla pro vytváření a změnu hesel	93
Podvržená data	93
Nevěřit ničemu, co přichází od uživatele	94
Vyhrazené znaky	94
Nahrazení řídicích znaků protokolu	95
Vstup dat do kódu HTML	96
Vyloučení bezpečnostních rizik	97
Bezpečnostní procedury uvnitř firmy	97
KAPITOLA 13	
Urychlování programu	99
Rychlost je otázkou správného návrhu systému	99
Paralelní zpracování instrukcí	100
Optimalizátor kódu	100
Kde se dá opravdu ušetřit	100
Návrh databáze	100
Volba algoritmů	101
Optimalizace bloků kódu	101

KAPITOLA 14

Nedodělané uživatelské rozhraní 103

Odstup uživatelů při zavádění aplikace	
Požadavky uživatelů na ovládání aplikace	104
Většina nedostatků se projevuje až při skutečné práci	104
Ovládání jen pomocí klávesnice	104
Správné rozvržení obrazovky	105
Snadný tisk informací na tiskárně	105
Rychlost aplikace je ovlivněna celým prostředím	105

KAPITOLA 15

Modifikace programu 107

Modifikace během vývoje nebo zkušebního provozu	107
Modifikace během ostrého provozu	108
Postup při provádění změn	108
Je nutné najít skutečnou příčinu chyby	108
Dokumentace každé změny	109
Jak se na změny připravit	109
Defenzivní programování	110

KAPITOLA 16

Programátorské slabosti 111

Kreativita versus rutina	112
Programátorská lenost	112
Zábavná forma vs. efektivní práce	113
Nejméně oblíbené činnosti	113
Odkládání nepřijemné práce	113

KAPITOLA 17

Chyby patří k programování 115

Boj s chybami začíná už na začátku projektu	116
Chyby nezatloukejte	116
Chyby musí být vidět	117
Procedury pro zajištění kvality	117
Hlavním cílem metodik je snížit rizika	118
Pravidla pro zajištění kvality	118
Specializovaný tester	118
Postup při testování	119

ČÁST IV

Co s chybami? 121

KAPITOLA 18

Co je to chyba	123
Pohled ze strany programátora	123
Pohled ze strany uživatele	123
Stanovení odpovědnosti za chyby a závady	124
Základní typy chyb	125
První typ – programátorské chyby	125
Druhý typ – nedostatky v zadání	125
Třetí typ – technologické nedostatky	126
Čtvrtý typ – ostatní	126

KAPITOLA 19

Příčiny chybovosti projektů	129
Kolik stojí chyba	129
Čím dříve se chyba objeví, tím stojí méně	130
Nejčastější příčiny neodhalených chyb	131
Příčina první – podcenění složitosti projektu	131
Příčina druhá – snaha o dokončení v termínu za každou cenu	132
Příčina třetí – použití zcela nových technologií	132
Příčina čtvrtá – nedostatečná kvalifikace nebo zkušenost	134
Příčina pátá – špatná komunikace mezi zadavatelem a uživatelem	134
Příčina šestá – podcenění analýzy a požadavků	135

KAPITOLA 20

Hledání chyb během vývoje	137
Příkazy v programu usnadňující ladění	137
Ověřovací podmínky (Asserts)	137
Záznam událostí (logging)	139
Trasování (tracing)	140
Špatné předpoklady generují chyby	140
Hledání chyb během psaní kódu	141
Neočekávaná chyba = stop	141
Chyba nesmí zmizet	141
Prvotní testování a breakpointy	142
Post-mortem debugger	142
Chyby mimo počítač vývojáře	143
Špatná konfigurace testovacího počítače	143
Špatná instalace OS a SW	143
Jiná verze operačního systému	144
Nedostatek systémových prostředků	144
Odlišné jazykové nastavení	144
Okrajové podmínky	144
Blokování dat uživateli	145

KAPITOLA 21

Revize kódu a dokumentů	147
Revize může odhalit skryté chyby	147
Co by měl kdo revidovat	148
Revize architektury a analýzy	148
Revize jádra aplikace	148
Revize uživatelského rozhraní	148
Kdo by měl provádět revizi	149
Průběh revize	149
Skupinová metoda (brainstorming)	150
Dokumentová metoda	150
Zpracování připomínek	150

ČÁST V

Testování aplikace **153**

KAPITOLA 22

Testování	155
Typy testování	155
Funkční testování	156
Uživatelské testování	157
Testování výkonnosti	158
Vytvoření správného testovacího prostředí	158
Testovací scénář	158
Testování zátěže	159
Automatické testování	159
Rozsah testovacího skriptu	160
Automatizované testovací nástroje	160
Ruční testování	161
Pisemný testovací scénář	161
Snaha simulovat skutečného uživatele	162
Bezpečnostní audit	163
Definice bezpečnostních požadavků	163

KAPITOLA 23

Kolotoč interního testování	165
Pravidelná tvorba verzí	166
Moderní metodiky vytváření aplikací	166
Buildy představují mezníky projektu	167
Odstraňování chyb v buildech	167
Zprávy o chybách	168
Správný popis chyb	168

Parametry buildu	168
Nebezpečí ohrožení komunikace v týmu	169
Proces vytvoření nové verze	169
Termín uzávěrky nové verze	169
Začlenění i rozdělaných částí	169
Automatizace sestavení buildu	169
Testování sestaveného buildu	170
Testování výkonnosti buildu	170
Komplexní testy všech modulů	170
KAPITOLA 24	
Zkušební provoz	173
Zkušební provoz	173
Pilotní provoz	174
Zahájení zkušebního provozu	174
Ovlivnění zkušebního provozu dalšími vlivy	174
Spolupráce je základem úspěchu testů	175
Komunikace s externími systémy	175
Průběh zkušebního provozu	175
Vlastní testování	176
Odstraňování chyb	176
Odstraňování závažných chyb	177
Zdroje informací o chybách	177
Ukončení zkušebního provozu	177
KAPITOLA 25	
Provoz u zákazníka	179
Akceptační testy	
Pilotní provoz	180
Definice pilotního provozu	180
Řešení problémů v ostrém provozu	181
Distribuce nových verzí uživatelům	182
ČÁST VI	
Řízení softwarových projektů	183
KAPITOLA 26	
Zahájení projektu	185
Vedoucí projektu	185
Typy projektů podle zadavatele	185
Typ první: Interní projekt	186
Typ druhý: Projekt pro volný trh	186
Typ třetí: Projekt softwaru na zakázku	187

Typ čtvrtý: Projekt jako subdodávka komplexního řešení	188
Chybějící přístup k zákazníkovi	188
Přesun rozhodnutí od hlavního dodavatele	188
Rozdílné testovací postupy	189
Autorská a uživatelská práva	189
První rozhovory s klientem	190
Pochopení zvyků zákazníka	190
Jaká je motivace zákazníka?	190
Vnitrofiremní politika a podpora projektu	191
Rozdílnost používaných jazyků	192
Diskuse nad rysy a rozdělení podle důležitosti	192
Jak zabránit budoucím průšvihům	193
Zhodnocení reálnosti cílů a přijatelnosti rizika	193
Chování zákazníka k jeho dodavatelům	193
Předčasné cenové odhady a nabídky	194
Uzavření obchodu za každou cenu	194
Smlouva se zákazníkem	195
Vyváženost smlouvy	195
Přesná specifikace produktu	195
Postup provádění změn specifikace	195
Komunikace změn smlouvy s vedoucím projektu	195
KAPITOLA 27	
Analýza	197
Role analýzy v projektu	197
Postup analýzy	198
Rozpracování specifikace	198
Ověření představ se zákazníkem	199
Metodika vývoje „shora dolů“	199
Metodika zaměřená na statickou strukturu	199
Kombinovaný přístup	199
Návrh zdola bez návaznosti na požadavky	200
Návrh architektury aplikace	200
Výstupy analýzy	201
Revize dokumentů zákazníkem	201
První dokument – podrobný popis požadavků	201
Druhý dokument – architektura a struktura aplikace	202
Dokument třetí – návrh struktury databáze	202
Dokument čtvrtý – řízení projektu	202
KAPITOLA 28	
Projektový plán	203
Projektový plán není dogma	203
Cíle projektového plánu	203

První účel – odhad času a nákladů	204
Druhý účel – určení závislostí prací	204
Třetí účel – rozvrh úkolů pro jednotlivce	204
Čtvrtý účel – kontrola průběhu a postupu	204
Vytvoření projektového plánu	205
Softwarové nástroje	206
První krok – rozdělení prací na úkoly	206
Druhý krok – určení návaznosti dílčích úkolů	206
Třetí krok – stanovení milníků	207
Čtvrtý krok – odhad potřebného času	207
Stanovení rezerv pro dokončení projektu	207
Započtení rezerv do projektového plánu	208
Krok pátý – výpočet nákladů a doby trvání	208
Jak urychlit celý projekt	209
Analýza rizik	209
Bod první – seznam možných rizik	210
Bod druhý – ohodnocení rizik podle kritérií	210
Nedostatek informací je zdrojem rizika	210
Revize projektového plánu	211
Význam aktuálnosti projektového plánu	211

KAPITOLA 29

Ekonomická stránka projektu **213**

Unikátní technologie nebo obchodní úspěch „Všeho s mírou“	213 214
Jak spočítat náklady	214
Náklady na práci vývojářů	215
Provozní režijní náklady	215
Nutné rezervy při neúspěchu projektu	215
Rentabilita projektu	216
Cenová jednání	216
První varianta – veškeré náklady plus přírážka	216
Druhá varianta – pevná cena	217

ČÁST VII

Lidské zdroje a komunikace **219**

KAPITOLA 30

Skladba vývojového týmu **221**

Jak poskládat tým	221
Znalost psychologie při sestavování a řízení týmu	222
Problematické typy v týmu – „beran“	222