

SWI129 / Softwarové inženýrství pro praxi - sylabus předmětu

Tomáš Krátký

Profinit, s.r.o.
tomas.kratky@profinit.eu1 Osnova přednášek a cvičení¹

#	Přednáška	Vybraná témata – "sylabus" pro přednášky	Ilustrace příklady ² - "sylabus" pro cvič.
1	Úvod do disciplíny softwarového inženýrství v jejích souvislostech	<ul style="list-style-type: none"> - Nominální obsah disciplíny - Primární činnosti sw eng - Podpůrné činnosti sw eng - Souvislosti s vedením sw projektu, softwarovým procesem organizace a sw eng v době rozvoje a údržby systému 	Průběžný příklad: <ul style="list-style-type: none"> - Představení projektu a systému pro průběžný příklad - Poptávka (RFP) a nabídka (Proposal) Doplňkové příklady: <ul style="list-style-type: none"> - Představení typů projektů a systémů, ze kterých budou příklady čerpány
2	Requirements Engineering	<ul style="list-style-type: none"> - Zjišťování, analýza, specifikace a verifikace požadavků - Formy specifikace - Role strukturovaného textu a jeho vztah ke všem jiným formám - Specifikace v době vymýšlení, vývoje a údržby sw systému - Role modelu GUI, modelu dat a jiných modelů - Vztah k architektuře a vedení sw projektu - Co v praxi téměř vždy funguje a co zcela jistě nefunguje - Jak nepoužívat use cases - Nástroje 	Průběžný příklad: <ul style="list-style-type: none"> - jak probíhá zjišťování u zákazníka, analytické schůzky - jak zapisovat co se domluvilo, jak zákazník přezkoumává specifikaci - kdo vlastně vymýšlí systém? - jak je architektura nutná už při iniciální specifikaci požadavků - jak nepoužívat CASE a UML - jak nepoužívat Use Cases - jak strukturovaný text musí být master - jak strukturovat stovky stran specifikace - podle čeho dekomponovat specifikaci - změna dekompozice specifikace od vymýšlení systému po údržbu - rozložení výdajů zdrojů na Reqs. Eng. v průběhu SDLC Doplňkové příklady: <ul style="list-style-type: none"> - ilustrace použití vzoru od Wieger z Process Impact - jak nefunguje katalog požadavků
3	Softwarová architektura a design	<ul style="list-style-type: none"> - Základní strukturální pohledy na systém (fyzický, logický atd.) - Projev architektury a designu v různých prg. modelech (svět imperativních jazyků, dbs atd.) - Design a prg techniky a jejich vztah k architektuře - Vztah architektury k organizaci a vedení sw projektů - Vztah specifikace požadavků k architektuře - Design sw v různých kontextech (e.g. dbs, java, web, dwh/etl, 4GL atd.) - Design patterns v praxi - Web Engineering v praxi 	Průběžný příklad: <ul style="list-style-type: none"> - které jsou základní požadavky vzhledem k architektuře - co jsou vlastně architektonická rozhodnutí - ukázky architektury systému - vztah logické, fyzické architektury se specifikací a okrajovými podmínkami - návrhu dbs, java, MVC - co předem zkoušet - mapování GoF DP na skutečný systém Doplňkové příklady: <ul style="list-style-type: none"> - ukázky návrhu a jeho dokumentace - ukázky nepovedených DM - ukázky povedených a nepovedených designů a diskuse příčin
4	Konstrukce	<ul style="list-style-type: none"> - Detailní design, programování, ladění a unit testování - Konstrukce a programátorské techniky - Konstrukce v prostředí dbs, imperativních jazyků - Reálné možnosti použití zásad XP, resp. kdy se hodí a kdy nikoli - Konstrukce v době vývoje na "zelené louce" vs. v době rozvoje a údržby systému - Nástroje 	Průběžný příklad: <ul style="list-style-type: none"> - programátorské a design techniky pro View Controller na bázi Struts - programátorské a design techniky pro vlastní MVC framework - programátorské techniky pro t-sql - konvence pro psaní programů v java, t-sql - použité nástroje Doplňkové příklady:

¹ Potenciální mírné přeskupení a změnu důrazu na jednotlivá témata považují za možnou.

² Pro ilustraci příklady budou použity i) *Průběžný příklad* založený na reálném systému a projektu, jehož vývoj autor tohoto dokumentu několik let vedl od nabídky až po údržbu (parametry: cca 1M SLOC, 5000 MD, ve špičce 13 lidí, informační systém v pojišťovnictví); ii) *Doplňkové příklady* z jiných projektů, které autor buď vedl nebo supervizoval nebo, které důvěrně zná jiným způsobem; doplňkové příklady budou vedeny snahou o dostatečnou ilustraci pestrosti situací reálného světa. Výčet věcí k ilustraci může být upraven pro větší edukativní efekt. Pro ilustraci příklady budou dále využita cvičení.

			<ul style="list-style-type: none"> - výsledky přezkoumání java a t-sql programů; na co dávat ve škole pozor když se učí programování, paralelismus, oo design, dbs ...
5	Testování	<ul style="list-style-type: none"> - Od unit testování až po akceptační - Organizace testování (plán testů, záznam testů, testovací případy, záznam chyb) - Organizace v době kvalifikačního a akceptačního testování - Testy nefunkční: výkon, memory leakage atd. - Průběžné testování a jeho organizace - Automatické testy, regresses - Organizace a ekonomika testování v různých fázích SDLC - Nástroje 	<p>Průběžný příklad:</p> <ul style="list-style-type: none"> - automatické testy od designu, přes údržbu po ekonomiku - plány testů - záznamy testů - test cases - průběžné testování - technická infrastruktura pro automatické testy - jak se liší kvalifikační testování v době vývoje systému a v době údržby systému - únava z testování - organizace a plánování v době akceptačního testování - rozložení výdajů zdrojů na testování v průběhu SDLC <p>Doplňkové příklady:</p> <ul style="list-style-type: none"> - projekty, kde není akceptační testování - projekty, kde není nezávislé testování jiným členem týmu
6	Dokumentace, validace, verifikace a Q&A	<ul style="list-style-type: none"> - Forma a obsah dokumentace - Logické a fyzické členění dokumentace - Role textu a grafické notace - Dokumentace a její struktura v různých fázích SDLC - Tvorba, údržba, používání dokumentace - Ekonomika dokumentace - Validace a verifikace technických artefaktů odpovídajícím jednotlivým činnostem sw eng - Ekonomika V&V/ Q&A - V&V/ Q&A v různých fázích životního cyklu - Nástroje 	<p>Průběžný příklad:</p> <ul style="list-style-type: none"> - strukturování dokumentace logické, fyzické a její život a údržba - použití textu a modelů - ukázky dokumentů, které zůstaly léta stejné, které zastaraly, které se musely dopsat - ukázky ex post summary dokumentů - ukázky jak v praxi vypadá V&V/ QA a k čemu se hodí <p>Doplňkové příklady:</p> <ul style="list-style-type: none"> - ukázky doc, javadoc, forest, html - jak spojit dokumentaci papírovou, elektronickou, help a zdrojáky tak aby nezastaraly
7	Konfigurační řízení	<ul style="list-style-type: none"> - Konfigurační jednotky - Verze systému - Konfigurační řízení vzhledem k vývoji, akceptaci a produkci - Konfigurační řízení vzhledem k různým typům konfiguračních jednotek (programy, dbs schema, data, konfigurace) - Údržba několika souběžných verzí systém - Konfigurační řízení v době vývoje a v době údržby - Vztah k různým prostředím a dodávkám systému - Source code policy - Nástroje 	<p>Průběžný příklad:</p> <ul style="list-style-type: none"> - ukázky všech typů konfiguračních jednotek (java, t-sql, schema, data, konfigurace, replikační definice, jsp, xml, ...) - konkrétní ukázky zdrojákové politiky v různých situacích (Build n ve vývoji, Build n-1 v akceptaci, Build n-2 v produkci atd.) - diskuse používání možností CVS/ SVN <p>Doplňkové příklady:</p> <ul style="list-style-type: none"> - typické problémy s CM: není jasné kde je jaký zdroják; není jasné kam kdy commit; kopie lokálních kopií CVS, nepoužívání tags a branches atd.
8	Vývojové prostředí, dodávky systému, akceptační a produkční prostředí	<ul style="list-style-type: none"> - Releases systému, patche systému - Nároky na vývojové prostředí - Build, deploy a smoked testing - Vztah ke konfiguračnímu řízení - Vztah k prostředí akceptačnímu a produkčnímu - Zálohování - Nástroje 	<p>Průběžný příklad:</p> <ul style="list-style-type: none"> - co vše se děje v noci: build, deploy, smoked test - zálohování - co když vývojové prostředí je jiné než produkční - co když vývojové prostředí používám jinak než produkční - co když nekontroluji výsledky automatických testů - ucelený postup pro integraci CM & DevEnv & CR & Delivery - dodávka systému a jak vypadá "velký" Release a "malý" Patch (EBF) - Instalační sety <i>en bloc</i> a inkrementální - Instalační dokumentace a postupy - Release notes - Release management - Aktualizace software off-line částí systému (cca 1000 laptops) <p>Doplňkové příklady:</p> <ul style="list-style-type: none"> - příklady různých způsobů dodávání zákazníkovi a všeho co s tím souvisí
9	Maintenance	<ul style="list-style-type: none"> - Specifika údržby - Výzvy a možnosti údržby - Typické problémy k řešení 	<p>Průběžný příklad:</p> <ul style="list-style-type: none"> - kompletní definovaný a praktikovaný proces údržby, který se vyjadřuje ke všem primárním

		<ul style="list-style-type: none"> - Ucelený postup údržby pro typický informační systém - Údržba vs. specifikace, architektura, testování, konfigurace, dokumentace, měření ... - Role měření před a při údržbě - Jak napsat kontrakt na údržbu sw systému - Nástroje 	<ul style="list-style-type: none"> - a podpůrným činnostem softwarového inženýrství v době údržby - maintenance checklist pro odhady - nutnost měření - kontrakt se zákazníkem <p>Doplňkové příklady:</p> <ul style="list-style-type: none"> - příklady různých systémů v údržbě a co to u nich znamená - co když není dokumentace - co když nejsou autoři systému - co když není konfigurační řízení - co když nevím co jsem dodal zákazníkovi - co když nevím co je v produkci - co když nejsou prg/ des. techniky - je těžké si pokazit elegantně navržený systém pro rozvoj? - je možné mít téměř dokonalý proces údržby, který ještě navíc podporuje ekonomiku a <i>vice versa</i>
10	Vedení, organizace projektu a modely životního cyklu	<ul style="list-style-type: none"> - Základní metoda vedení sw projektu - Rozsah, jeho stanovení a řízení - Modely SDLC a jejich rekurentní podstata - Vztah vedení s architekturou, specifikací a modely SDLC - Podstat debaty o modelech SDLC, kdy se hodí waterfall, XP, architecture-based etc. - Generalizovaný pohled na problematiku modelů SDLC Anchoring the Software Process (LCO, LCA, IOC) 	<p>Průběžný příklad:</p> <ul style="list-style-type: none"> - ukázky plánů, odhadů, rizik, problémů, status reportů, zápisů ze schůzí - typické problémy, typická rizika - řízení rozsahu - ukázky jak je nutné znát specifikaci a architekturu systému pro efektivní řízení jeho vývoje - důležitost otevřeného pravdivého vztahu se zákazníkem - důležitost nalezení partnerů pro řízení projektu na straně zákazníka - plány krátkodobé, dlouhodobé - projektové schůzky interní a se zákazníkem <p>Doplňkové příklady:</p> <ul style="list-style-type: none"> - ilustrace problémů s vedením projektů - ilustrace různých forem jak zachytit plán, rizika, problémy
11	Odhadování, plánování, historie projektů a nabídky	<ul style="list-style-type: none"> - Nutnost, možnost a realizovatelnost základních měření - Odhadování od chvíle nabídky po údržbu - Plán dlouhodobý a krátkodobý - Jaké všechny aspekty plánovat (výstup, čas kalendářní, lidi, metodu, výdaj zdrojů) - Každodenní monitorování situace, rutinní řízení a přeplánování projektu. - Nároky na historii projektu a co to vlastně je - RFP a následný Proposal; nároky na nabídku 	<p>Průběžný příklad:</p> <ul style="list-style-type: none"> - historie projektu - příklady měření - příklady nástrojů pro měření - vysvětlení jak strukturovat měření - názorně ukázat jak při psaní nabídky/ kontraktu pro rozvoj a údržbu, plánování v pozdějších fázích projektu potřebují předchozí měření - ilustrace jak bezbolestně zavést měření - typické rozložení typických činností sw eng. v čase a jejich proporce - ukázka jak používat bugzilla a jiné nástroje pro toto <p>Doplňkové příklady:</p> <ul style="list-style-type: none"> - historie mnoha projektů - různé způsoby jak získat základní metriky - ukázka nabídky
12	Proces vývoje projektu a organizace	<ul style="list-style-type: none"> - Postupy práce pro jednotlivý projekt - Co předepisovat a co nepředepisovat pro jednotlivý projekt - Minimální praxe na projektu a jak ji stanovit a prosadit - Jak a kdy stanovit sw proces organizace - Vztah sw procesu organizace a jednotlivých projektů - Jak prosadit sw proces organizace - Minimální sada nástrojů - Komentář k CMM, SPICE, TickIT, ISO, MIL, DoDs, ESA a podobným standardům, resp. <i>de facto</i> standardům na vývoj software 	<p>Průběžný příklad:</p> <ul style="list-style-type: none"> - jak budovat proces organizace z procesu projektu(ů) - jak použít projekt jako <i>Experience Factory</i> <p>Doplňkové příklady:</p> <ul style="list-style-type: none"> - jak na konkrétních projektech prosazovat proces vývoje organizace
13 ³	Shrnutí, příklady, diskuse, literatura, odborné zdroje, příprava na test	<ul style="list-style-type: none"> - Toto pouze říká, že pokud vyjde 13. přednáška tak se nalezen smysluplná náplň - Jak psát nabídku - Jaká bývá situace v době psaní nabídky 	<p>Průběžný příklad:</p> <ul style="list-style-type: none"> - cokoli se bude hodit <p>Doplňkové příklady:</p> <ul style="list-style-type: none"> - příklady nabídek a situací, za kterých vznikaly

³ Nemusí se uskutečnit.

2 Struktura přednášky

Každá přednáška bude vhodně obsahovat:

1. základní přehled nominální učebnicové látky
2. minimální praxi, pod kterou nelze bez velkého rizika klesnout
3. ilustrace z reálných projektů
4. praktické závěry
5. kvalitní a pragmatické zdroje⁴ (knihy, články, reporty, templates dokumentů, cheklists etc.)

⁴ Typicky z NASA, DoD a pár amerických firem a universit.