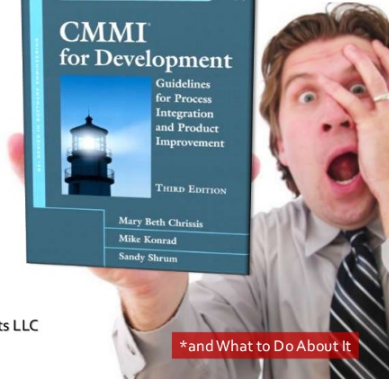
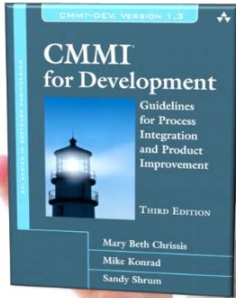
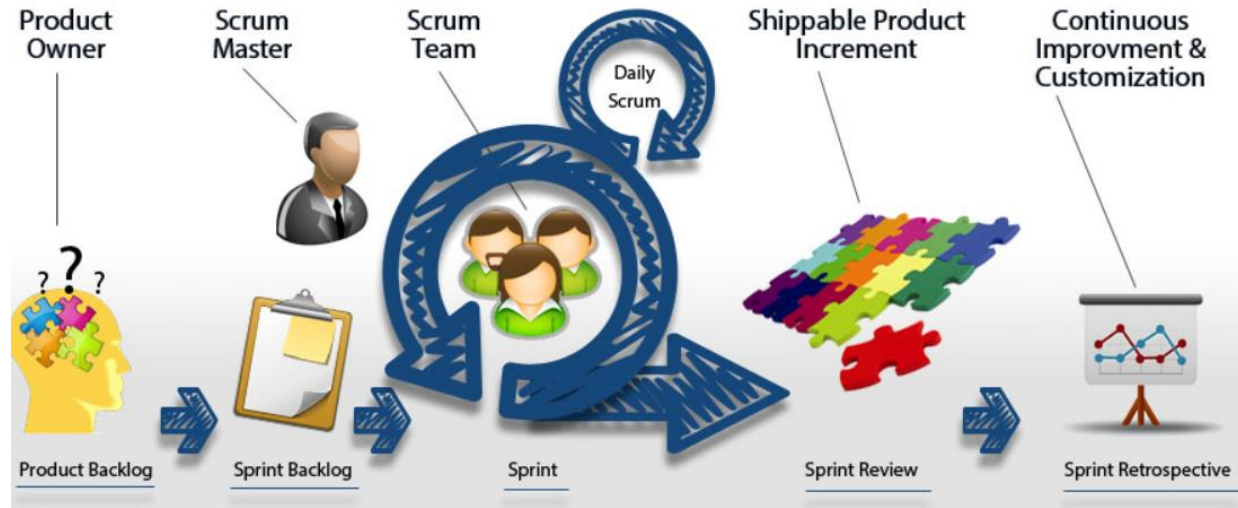


# Why Project Managers (Understandably) Hate the CMMI\*



Bill Smith CEO  
Leading Edge Process Consultants LLC  
www.CmmiTraining.com

\*and What to Do About It



# Razvojni put

Odmah nakon početka projekta

....

Pre nego se završi projekat

U.S. Small Business Administration



Your Small Business Resource

# Razlike izmedju efektivnosti i efikasnosti? Primer.

- ▶ Efektivan je učinkovit (od više puteva izabrati najbolji)
- ▶ Efikasan je (na izabranom putu stisnete papučicu za gas i prođete sto brže)
- ▶ Biti fokusiran(effectiveness?) znači raditi pravi posao, dok biti efikasan(efficiency?) znači raditi posao kako treba.

# Soon after the project starts

- ▶ A. To learn CMMI
- ▶ Aerodrom u Singapuru je radjen po CMMI metodi, snabdevanje delova fronta u Afganistanu, Rajfajzen banka u Becu je pravila CMMI plan za Indiju.
  
- ▶ B. To learn Another more focused approach
- ▶ (e.g., Microsoft Project, trello); HW = MP.local

# Before the project is over:

- ▶ a. To form a company (e.g., using the SBA guidelines);  
HW = BizPlan.sba
- ▶ b. To protect the company product with a patent  
(e.g., using the USA patent office guidelines); HW =  
PatentApplication.pto

# Soon after the project starts

- ▶ B. To learn Another more focused approach
- ▶ Agile Manifesto, definicija i principi
- ▶ Karakteristike agilnog razvoja
- ▶ Pojam “kaubojskog kodiranja”
- ▶ Aktuelne agilne metode

# The Best method - 10 celina

- ▶ a) Uvod
- ▶ b) Predstavljanje problema
- ▶ c) Postojeca resenja i njihova kritika
- ▶ d) Predloženo resenje i zasto se ocekuje da bude bolje
- e) Uslovi i pretpostavke koje idu uz predstojeće istrazivanje
- ▶ f) Analiticka analiza
- ▶ g) Analiza simulacija
- ▶ h) Analiza implementacije
- ▶ i) Zakljucak
- ▶

# Pojam agilnih metoda

- ▶ Grupa metodologija koje se zasnivaju na iterativnom i inkrementalnom razvoju
- ▶ Zahtevi i rešenja se razvijaju kroz saradnju između timova

# Pojam iteracije

- ▶ Jedan kratak period vremena, koji obično traje između jedne i četiri nedelje
- ▶ Svaka iteracija uključuje tim kroz celokupan ciklus razvoja softvera



# Agile Manifesto

- ▶ Osnovna definicija agilnog razvoja
- ▶ Nastao je 2001. godine kao odgovor na potrebu za kreiranjem razvojnih metoda koje je lako savladati

# Anekdota

<http://www.agilejedi.com/chickenandpig>

Pile i prase su živele na farmi i poželeti da razviju projekat koji će obradovati farmera.

Pile je predložilo: „Napravimo doručak za gazdu.“

Prase je upitalo: „Kako bismo ga nazvali?“. Pile je odgovorilo: „Šunka i jaja?“. Prase je razmislilo za trenutak i reklo: „Ne, hvala. Ja bih bio žrtvovan, a ti samo uključen.“

**AGILNI TIM:** prasići i pilići

**PRASIĆI** (Razvojni tim, Projektanti, Testeri tj. vođa, vlasnik proizvoda, razvojni tim).

Svi ostali su „pilići“. „Pilići“ ne mogu da govore „prasićima“ kako da rade svoj posao.

# Lista principa koji se moraju zadovoljiti da bi neka metoda bila agilna

1. Zadovoljstvo korisnika brzom isporukom korisnog softvera
2. Mogućnost promene zahteva, čak i u poodmakloj fazi razvoja
3. Česta isporuka softvera, u razmaku od par nedelja
4. Ispravan softver je osnovna mera napretka

# Agile Manifesto principi

5. Razvoj koji je u stanju da održi konstantan tempo
6. Bliska kooperacija između projektanata i poslovnih saradnika
7. Najbolji tip komunikacije je komunikacija “licem-u-lice”
8. Projekti se izvode u okruženju u kojem su motivisani pojedinci, u koje se može imati poverenja

# Agile Manifesto principi

9. Kontinualno usmeravanje pažnje ka tehničkoj veštini i dobrom dizajnu
10. Jednostavnost
11. Samoorganizovani timovi
12. Prilagođavanje promenljivim okolnostima

# Karakteristike

- ▶ Adaptivno planiranje
- ▶ Evolutivni razvoj i isporuka
- ▶ Iterativni pristup
- ▶ Brz i fleksibilan odgovor na promene
- ▶ Timski rad i kolaboracija
- ▶ Razbijanje zadatka na sitne korake
- ▶ Nema dugoročnog planiranja

# Karakteristike

- ▶ Minimalan broj funkcionalnosti po iteraciji
- ▶ Tim nema ni jedan od postojećih oblika hijerarhije
- ▶ Tim sam bira način realizacije zadatka u svakoj iteraciji
- ▶ Zastupnik klijenta proverava usklađenost projekta sa zahtevima

# Karakteristike

- ▶ Akcenat je na funkcionalnom softveru, a ne na dokumentaciji
- ▶ Agilne metode se nazivaju i „nedisciplinovanim hakovanjem“ ili „kaubojskim kodiranjem“



- ▶ „Kaubojsko kodiranje“ karakteriše odsudstvo definisane metode
- ▶ Agilni timovi slede definisani proces, koji je često vrlo rigorozan

# Maksime agilnog razvoja

- ▶ Veću vrednost ima:
  - ▶ Pojedinaac i interakcija od procesa i alata
  - ▶ Dobra programska podrška od sveobuhvatne dokumentacije
  - ▶ Saradnja s klijentima od pregovaranja oko ugovora
  - ▶ Brz odgovor na promene od praćenja plana

# Šta ih čini jedinstvenim?

- ▶ Primenom agilnih metoda razvoj softvera postaje:
  - ▶ Inkrementalan
  - ▶ Kooperativan
  - ▶ Direktan
  - ▶ Adaptivan

# Proces uvođenja

- ▶ Pre primene agilnih metoda odgovoriti na sledeća pitanja:
  - ◆ kako se odabrana metoda uklapa u okruženje
  - ◆ da li postoje timovi koji žele da probaju da rade na agilni način
  - ◆ da li postoje zainteresovani korisnici

# Istorijat

- ▶ Nastanak ranih agilnih metoda vezuje se za period pre 2000. godine:
  - ▶ 1986. - SCRUM u oblasti opšteg menadžmenta
  - ▶ 1995. - Adaptivni razvoj softvera, Razvoj vođen karakteristikama i Metod dinamičnog razvoja sistema
  - ▶ 1996. - Crystal clear i ekstremno programiranje

# Agilne metode danas

- ▶ Ekstremno programiranje (XP) i Industrijsko ekstremno programiranje (IXP)
- ▶ Scrum
- ▶ Adaptivni razvoj softvera (ASD)
- ▶ Rational Unified Process

# Agilne metode danas

- ▶ Crystal clear i ostale crystal metode
- ▶ Metode dinamičkog razvoja sistema (DSDM)
- ▶ Razvoj vođen karakteristikama (FDD)
- ▶ Agile Unified Process (AUP)
- ▶ Essential Unified Process

# Ekstremno programiranje - XP

- ▶ Cilj je da poboljša kvalitet softvera i pruži mogućnost brzog odgovora na promenljive zahteve korisnika
- ▶ Zasniva se na teoriji “manje je dobro, ali više je bolje”, pa otuda i naziv ove metodologije ekstremno programiranje



# XP karakteristike

- ▶ Programiranje u parovima
- ▶ Ekstenzivno proveravanje koda
- ▶ Funkcionalnosti se programiraju tek kada se pojavi potreba
- ▶ Ravna menadžment struktura
- ▶ Jednostavnost i jasnoća koda
- ▶ Česta komunikacija sa kupcem i među samim programerima

# Potencijalne mane

- ▶ Nestabilni zahtevi korisnika
- ▶ Nedostatak specifikacije i dokumenata
- ▶ Kompromisi usled konflikta sa korisnikom nisu dokumentovani

# Životni ciklus

- ▶ Istraživanje
- ▶ Planiranje
- ▶ Iteracije do nove distribucije
- ▶ Proizvodnja
- ▶ Održavanje i umiranje

# Industrijsko ekstremno programiranje - IXP

- ▶ Nastalo 2004. u konsultantskoj firmi Cutter, evolucijom ekstremnog programiranja
- ▶ Osnovne vrednosti mogu biti fleksibilne, odnosno razlikuju se od tima do tima

# Pet osnovnih vrednosti



# Poboljšanje postojeće prakse

- ▶ Neki od načina za poboljšanje su:
  - ▶ Razvoj vođen testovima priče
  - ▶ Dizajn vođen domenom
  - ▶ Programiranje u parovima
  - ▶ Iterativno testiranje korisnosti

# Scrum

- ▶ Iterativni, inkrementalni razvojni okvir
- ▶ Upravljanje projektima, pre svega za razvoj softvera
- ▶ Agilni razvoj softvera
- ▶ Termin dolazi iz ragbija
- ▶ Strategija „vraćanje lopte nazad u igru uz pomoć timskog rada“

# Scrum - karakteristike

- ▶ Sprint - osnovna jedinica u procesu razvoja
- ▶ Sprint traje između jedne nedelje i mesec dana i teži da ima konstantnu dužinu
- ▶ Definisanje skupa praksi i uloga



# Scrum uloge

1. "Scrum Master" održava procese, najčešće umesto projektnog menadžera
2. "Product Owner" predstavlja zainteresovane strane ili poslovanje
3. "Team" grupa čije su funkcije isprepletane, a zadatak je analiza, dizajn, implementacija, testiranje itd.

# Scrum proces



# Faze procesa

## 1. Faza pre igre:

- ▶ Planiranje - definisanje sistema koji će se razvijati
- ▶ Dizajn/arhitektura - kreiranje dizajna sistema na osnovu podataka iz prethodne faze

## 2. Faza razvoja (faza igre) - agilni deo Scrum pristupa, neprekidna kontrola svih bitnih parametara; “crna kutija” u kojoj se može očekivati nepredviđeno ponašanje

# Faze procesa

- 3. Faza posle igre** - do ove faze se dolazi kada se utvrdi da su svi zahtevi ispunjeni; nema novih zahteva, radi se priprema za izdavanje sistema
- ▶ Priprema uključuje testiranje, integraciju i dokumentaciju

# Crystal Clear

- ▶ Član Crystal porodice metodologija
- ▶ Namijenjen timovima koji se sastoje od 6 ili 8 projektanata
- ▶ Primena na sisteme koji nisu krucijalni
- ▶ Akcenat na efikasnosti i bezbednosti komponenti projekta
- ▶ Fokusira se na ljude, a ne na procese

# Adaptive Software Development

- ▶ Posledica brzog razvoja aplikacija
- ▶ Princip kontinualne adaptacije procesa
- ▶ Repetitivne serije ciklusa špekulacije, kolaboracije i učenja
- ▶ Paradoks planiranja - sve zainteresovane strane greše po pitanju nekih aspekata, dok pokušavaju da definišu misiju projekta

# ASD životni ciklus

- ▶ Osnovne karakteristike ASD životnog ciklusa su:
  - ▶ Fokusira se na misiju
  - ▶ Zasniva se na karakteristikama
  - ▶ Iterativan
  - ▶ Vođen je rizikom
  - ▶ Tolerantan na promene

# Feature-driven development

- ▶ Iterativni, inkrementalni proces za razvoj softvera
- ▶ Spaja najbolje prakse
- ▶ Blagovremena isporuka
- ▶ FDD proces se sastoji od 5 aktivnosti:
  1. Razvoj modela
  2. Kreiranje liste karakteristika
  3. Planiranje na osnovu karakteristika
  4. Dizajniranje prema karakteristikama
  5. Razvoj prema karakteristikama



# Dynamic systems development method

- ▶ Bazira na metodi brzog razvoja aplikacija
- ▶ Iterativni, inkrementalni pristup
- ▶ Kontinualno uključivanje korisnika odnosno kupca
- ▶ Definiše troškove, željeni kvalitet i vreme
- ▶ Koristi MoSCoW metodu za prioritete

# MoSCoW metoda

- ▶ Ova metoda definiše sledeće kategorije:
  - ▶ M - MUST - zahtevi koji moraju biti zadovoljeni
  - ▶ S - SHOULD - stavke koje bi trebalo uključiti u rešenje, ako je to moguće
  - ▶ C - COULD - poželjni zahtevi, ali ne i neophodni i biće uključeni ukoliko to vreme i resursi dozvole
  - ▶ W - WON'T - zahtevi o kojima se može razmišljati u budućnosti

# DSDM principi

1. Fokusirati se na potrebe poslovanja
2. Isporuka na vreme
3. Kolaboracija
4. Nikada ne praviti kompromis po pitanju kvaliteta
5. Inkrementalni razvoj koji se bazira na čvrstoj osnovi
6. Iterativni razvoj
7. Kontinualna i jasna komunikacija
8. Demonstracija kontrole

# Rational Unified Process

- ▶ Deo Rational Software korporacije, koja je bila sastavni deo IBM-a do 2003. godine
- ▶ Adaptibilan proces koji se može krojiti prema potrebama organizacije i timova
- ▶ Glavni elementi:
  - ▶ Uloge
  - ▶ Proivodi rada
  - ▶ Zadaci

# RUP životni ciklus

- ▶ Životni ciklus projekta se deli u četiri faze:
  1. Početna faza
  2. Faza razrade
  3. Faza konstrukcije
  4. Prelazna faza
- ▶ Svaka faza ima određeni cilj koji mora da zadovolji i na kraju prekretnicu koja označava da je cilj ostvaren

# Agile Unified Process

- ▶ Pojednostavljena verzija RUP-a
- ▶ AUP ima sedam disciplina:
  1. Model
  2. Implementacija
  3. Test
  4. Razvoj
  5. Konfiguracioni menadžment
  6. Projektni menadžment
  7. Okruženje

# Essential Unified Process

- ▶ Nastao je kako bi poboljšao Rational Unified Process
- ▶ Mogućnost biranja onih praksi koje su primenjive u konkretnoj situaciji i njihovo uključivanje -> poboljšanje u odnosu na RUP

# Agile Modeling

- ▶ Dodatak ostalim agilnim metodama kao što su XP, AUP, Scrum
- ▶ Osnovne vrednosti AM:
  - ▶ Hrabrost
  - ▶ Komunikacija
  - ▶ Povratna sprega
  - ▶ Skromnost
  - ▶ Jednostavnost



# Velocity

- ▶ Merenje tempa kojim tim radi kako bi se procenilo vreme potrebno za dodavanje nove funkcionalnosti
- ▶ Merenje brzine kako bi se obezbedile dodatne informacije o ostvarenom učinku tima tokom vremena

# Velocity terminologija

- ▶ **Jedinica posla** - jedinica koju je tim izabrao kao meru brzine izvođenja projekta
- ▶ **Interval** - trajanje svake iteracije u procesu razvoja softvera
- ▶ **Brzina** - određuje se na osnovu prethodna dva parametra

# Before the project is over:

- ▶ a. To form a company (e.g., using the SBA guidelines);  
HW = BizPlan.sba
- ▶ b. To protect the company product with a patent  
(e.g., using the USA patent office guidelines); HW =  
PatentApplication.pto

# Kapital, ugovori, savetovanja (capital, contracts and counselling)



# Koji su tipicni delovi biznis plana?

- ▶ Rezime,
- ▶ Poslovni opis i vizija,
- ▶ Definicija trzista,
- ▶ Opis proizvoda i usluga,
- ▶ Organizacija i menadzment,
- ▶ Strategije marketinga i prodaje,
- ▶ Finansijski menadzment.

Koja je razlika između pokretanja nove kompanije i traženja više novca za postojeću kompaniju?

- ▶ <http://home.etf.rs/~vm/os/uspi/sba.html>

# Definisati pojam “andjeoske” investicije

- ▶ Andjeo investitor ili andjeo je bogat pojedinac koji pruža kapital za pokretacki(start-up) biznis, obicno za zamenu trazi “konvertibilni” dug ili kapital u preduzecu.

# Definisati pojam rizični kapital

- ▶ Rizični kapital je novac obezbedjen da finansira kompanije u ranim fazama, nastajanju ili kompanije koje rastu.
- ▶ Fondovi rizičnog kapitala ulažu u preduzeća u zamenu za kapital u preduzeću u koje ulažu.
- ▶ Uglavnom ulažu u preduzeća koja imaju neku novu tehnologiju ili poslovni model u visoko tehnološkoj industriji, kao npr. biotehnologija ili IT.



# Koja je razlika izmedju marketinga i prodaje?

Prodaja je ubedjivanje, marketing je primenjeno razumevanje.