

Univerza  
v Ljubljani

Fakulteta za *gradbeništvo*  
in *geodezijo*



# Agile metodologije razvoja programske opreme

februar 2016

doc.dr. Matevž Dolenc  
mdolenc@itc.fgg.uni-lj.si

# Vsebina

## Klasične metodologije

Model vodnega slapa (waterfall model)

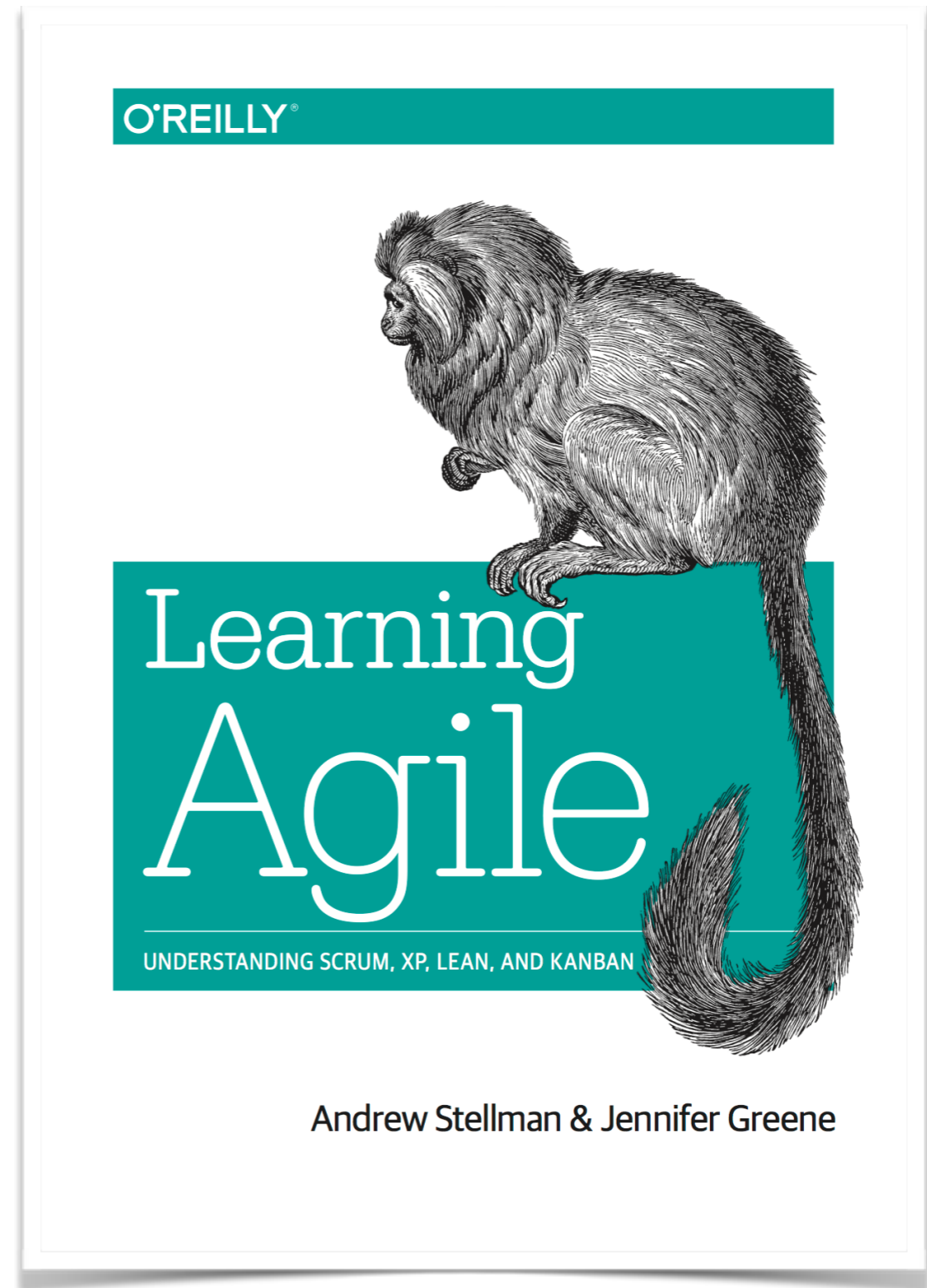
## Agilne metodologije

Scrum

eXtreme Programming

Lean

Kanban



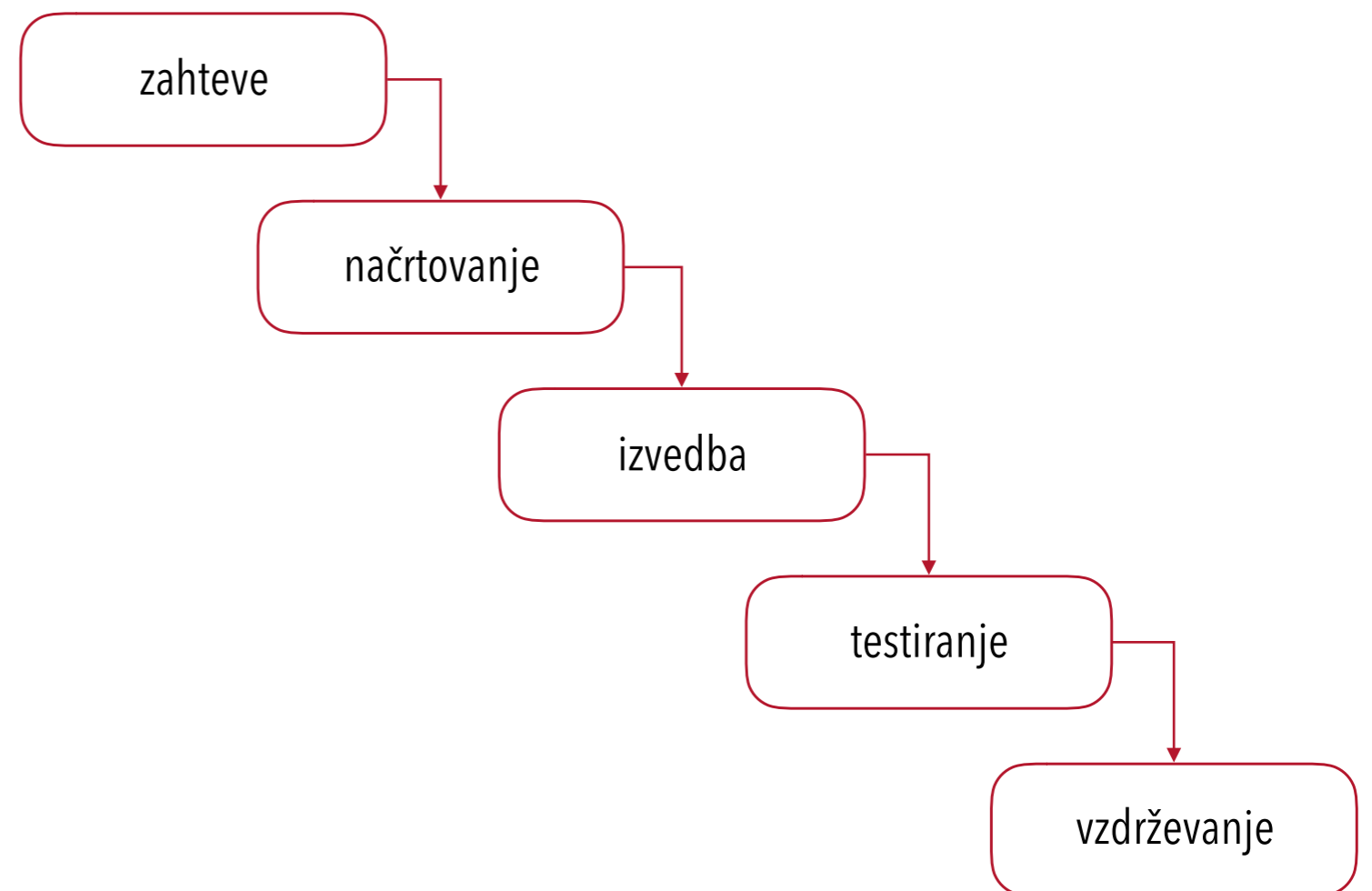
# Model vodnega slapa (waterfall model)

## Značilnosti

**Zaporedno** izvajanje faz razvoja

Zelo redko težimo k vračanju razvoja nazaj na **prejšnje faze**

Velik poudarek na pravilnosti/celovitosti že v **prvem koraku**



# Model vodnega slapa (waterfall model)

## Značilnosti

**Zaporedno** izvajanje faz razvoja

Zelo redko težimo k vračanju razvoja nazaj na **prejšnje faze**

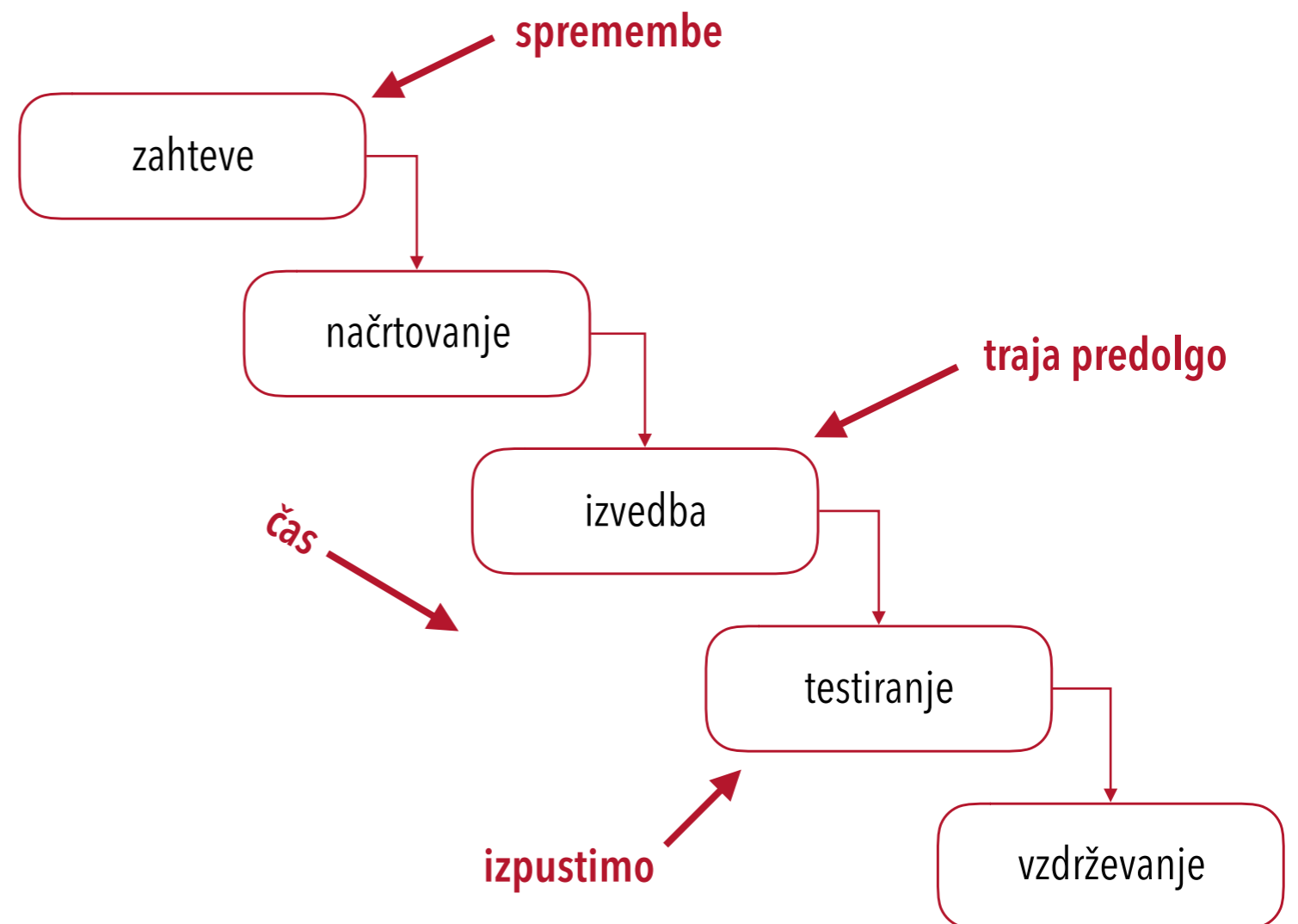
Velik poudarek na pravilnosti/celovitosti že v **prvem koraku**

## Problemi

**Produkt** dobimo na **koncu projekta**

**Testiramo** proti koncu projekta (običajno **zmanjka časa**)

**Potrditev** rešitev dobimo pozno (običajno **prepozno**)



Razvoj programske opreme po  
metodi vodnega stolpa vsebuje  
**veliko nevarnosti**, je pogosto **dražji** in  
običajno **manj učinkovit** od  
**agilnih metodologij** razvoja.

**HITROST** prilagodljivost  
**AGILNE metodologije**  
sodelovanje **KVALITETA** iterativnost  
odprtost

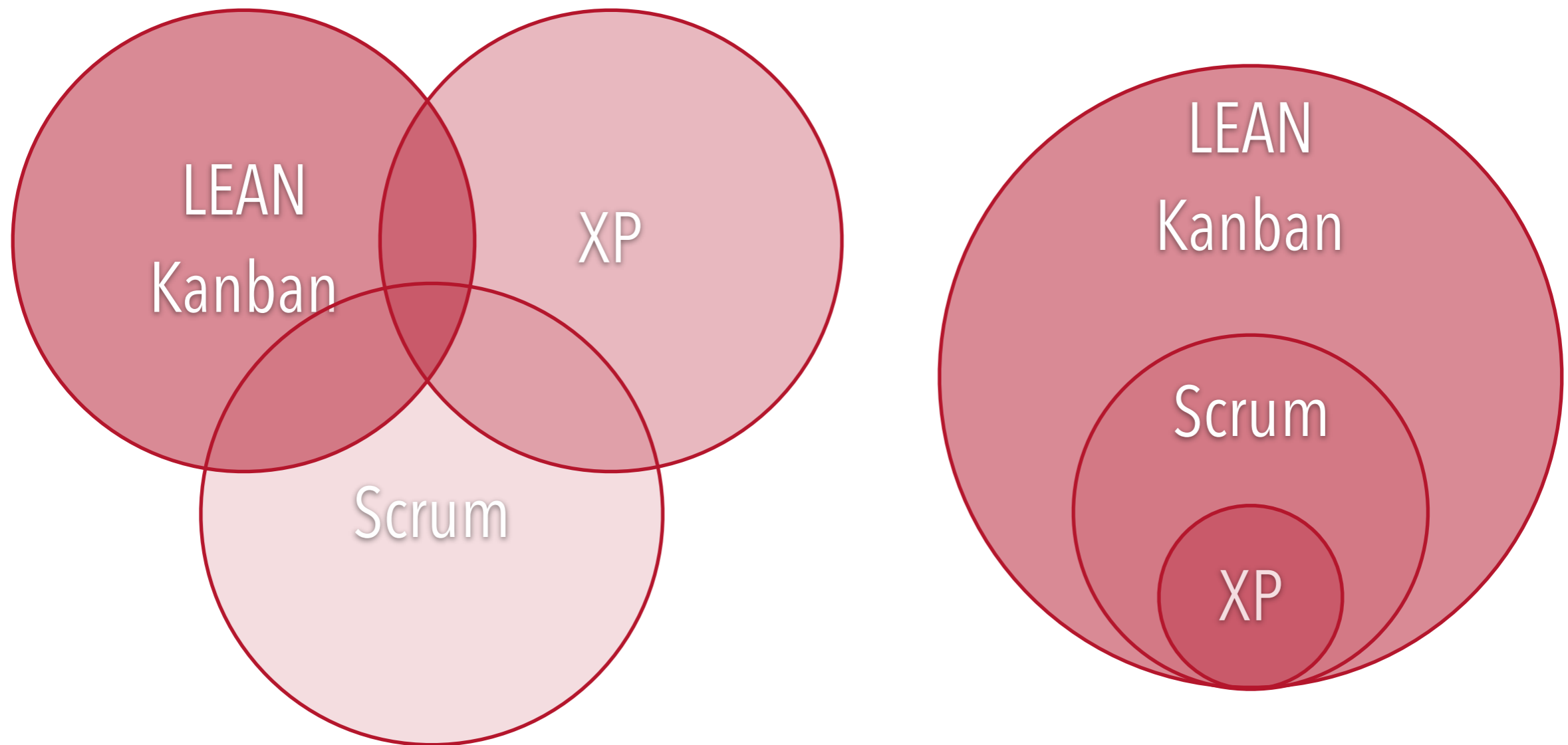
To ni proces, temveč **filozofija** oziroma **nabor vrednot**

# Agilni manifest

- Posamezniki in interakcije pred procesi in orodji
- Delujoča programska oprema pred vseobsežno dokumentacijo
- Sodelovanje s stranko pred pogodbenimi pogajaji
- Odziv na spremembe pred toga sledenjem načrtom

Z drugimi besedami, četudi **cenimo dejavnike na desni**, vseeno **bolj cenimo** tiste **na levi**.

# Različne agilne metodologije

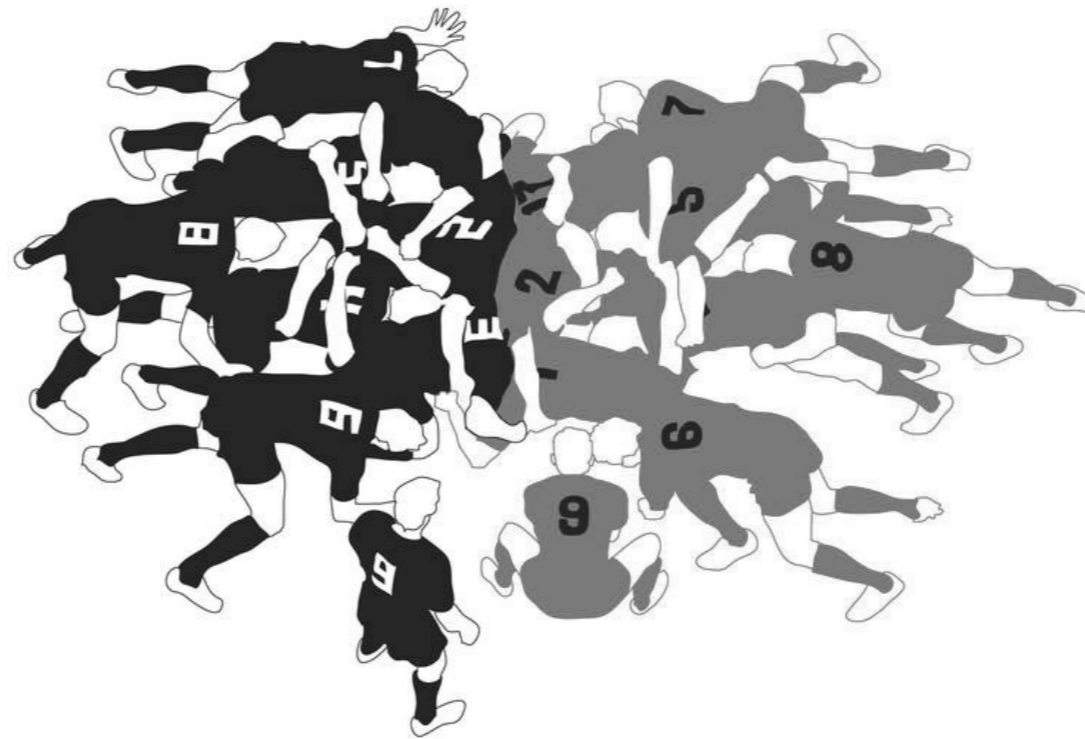


Vrednote, zapisane v Agilnem manifestu, so skupne vsem agilnim metodologijam.



# Scrum

Scrum je okvir znotraj katerega lahko **rešujemo kompleksne probleme** pri tem pa **produktivno** in **kreativno** razvijamo izdelke **najvišje** možne **kvalitete**.



<http://scrum.org>

SCRUM vodič: pravila igre (2001), <http://www.scrumguides.org/docs/scrumguide/v1/Scrum-Guide-SI.pdf>

# Scrum temelji

## Empirični nadzor procesov

Znanje se pridobi z izkušnjami in odločitvami na podlagi znanega

Iterativni, inkrementalni pristop za optimizacijo predvidljivosti in nadzor tveganja

## Trije osnovni principi

Transparentnost

Pregled

Prilagoditev

# Scrum ekipa

## Scrum Roles

### Lastnosti

Navzkrižno funkcionalne

Samo-organizirane

Fleksibilnost, kreativnost in produktivnost


Agilnost / celovitost / 3 - 9 oseb

### Vloge

Lastnik izdelka

Razvojna ekipa

Scrum učitelj




**Pigs**

Roles and RACI -VS governance matrix:

- Product Owner (accountable, verifier)
- Scrum Master (responsible)
- Scrum Team (responsible)

Committed to:

- Adhering to the Scrum process
- Collaboration to get work done
- "having their bacon on the line"



**Chickens**

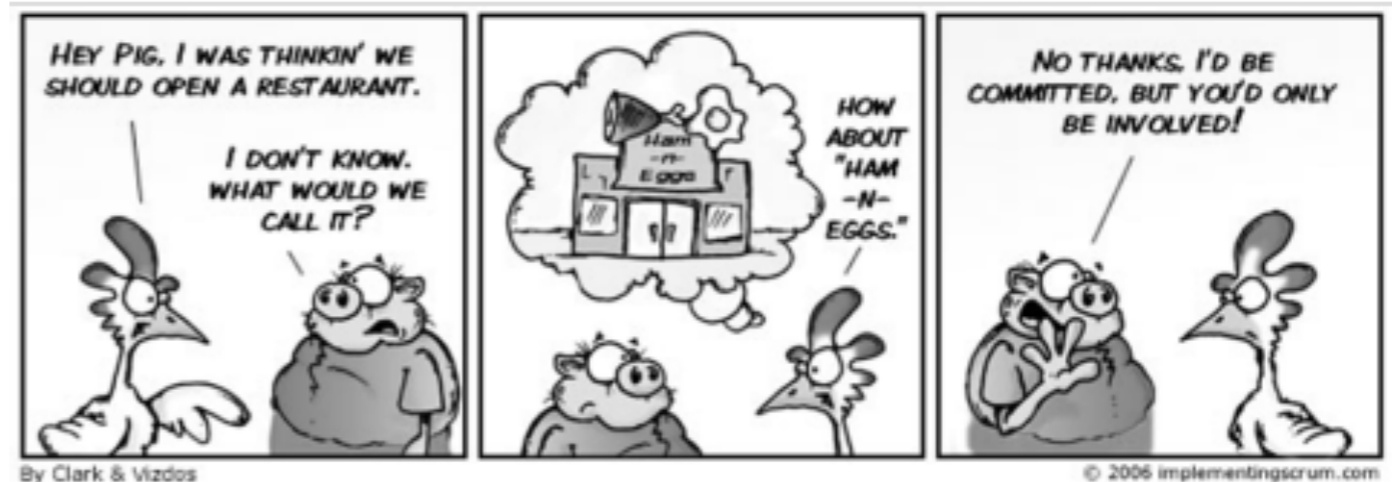
Roles:

- Business owner (accountable)
- Project committee (signatory)
- Stakeholders
- End-users
- Consulting experts

Everyone else who is involved, engaged, consulted and interested in the project.

RACI-VS governance matrix:

- Consulted, informed



# Scrum proces

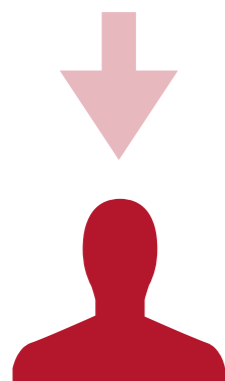
Zahteve vodstva, ekipe,  
interesnih skupin,  
naročnikov, uporabnikov ...



Scrum učitelj



Razvojnna ekipa

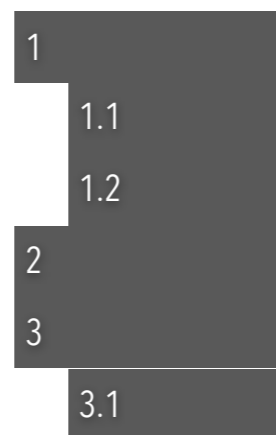
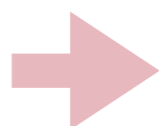


Lastnik izdelka

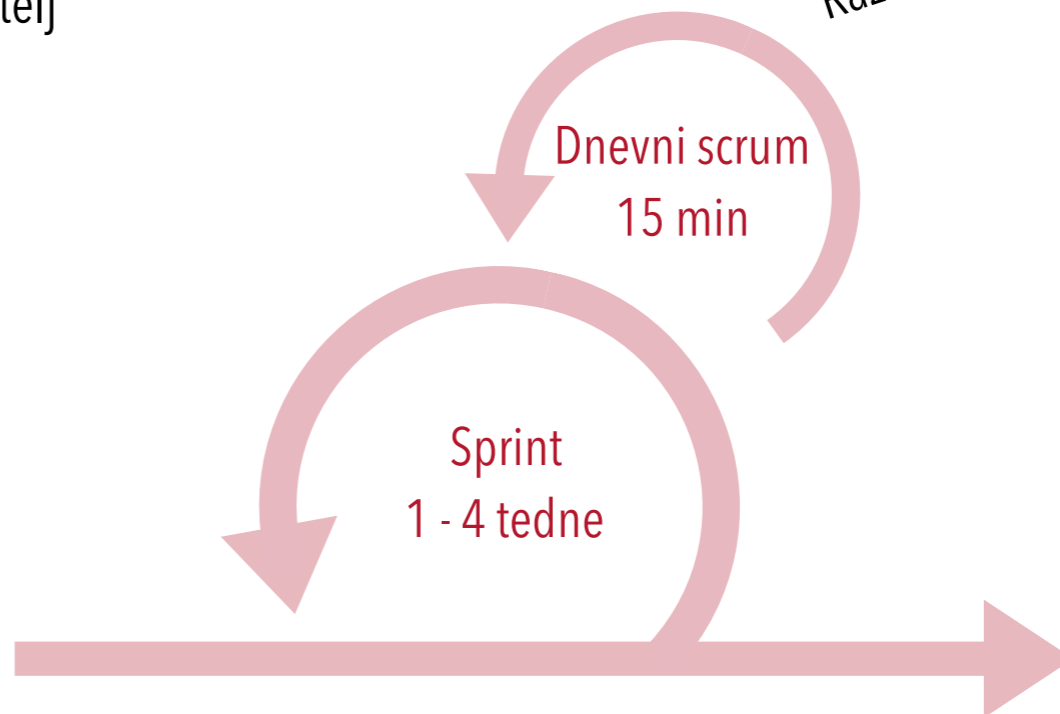
Načrtovanje sprinta  
4 - 8 ur



Seznam zahtevkov  
izdelka



Seznam zahtevkov  
sprinta



Revizija sprinta  
2 - 4 ure

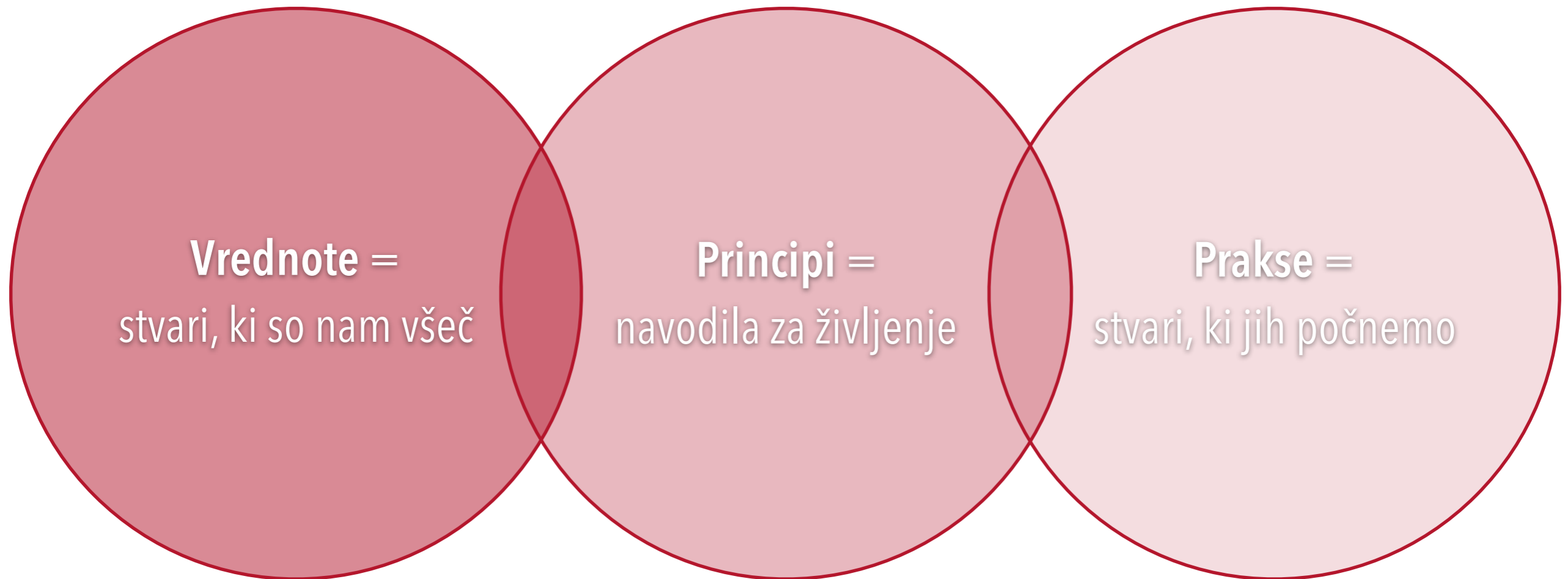


Končni izdelek



Retrospektiva sprinta  
1 - 3 ure

# eXtreme Programming (XP)



komunikacija, enostavnost, pogum,  
povratne informacije, spoštovanje

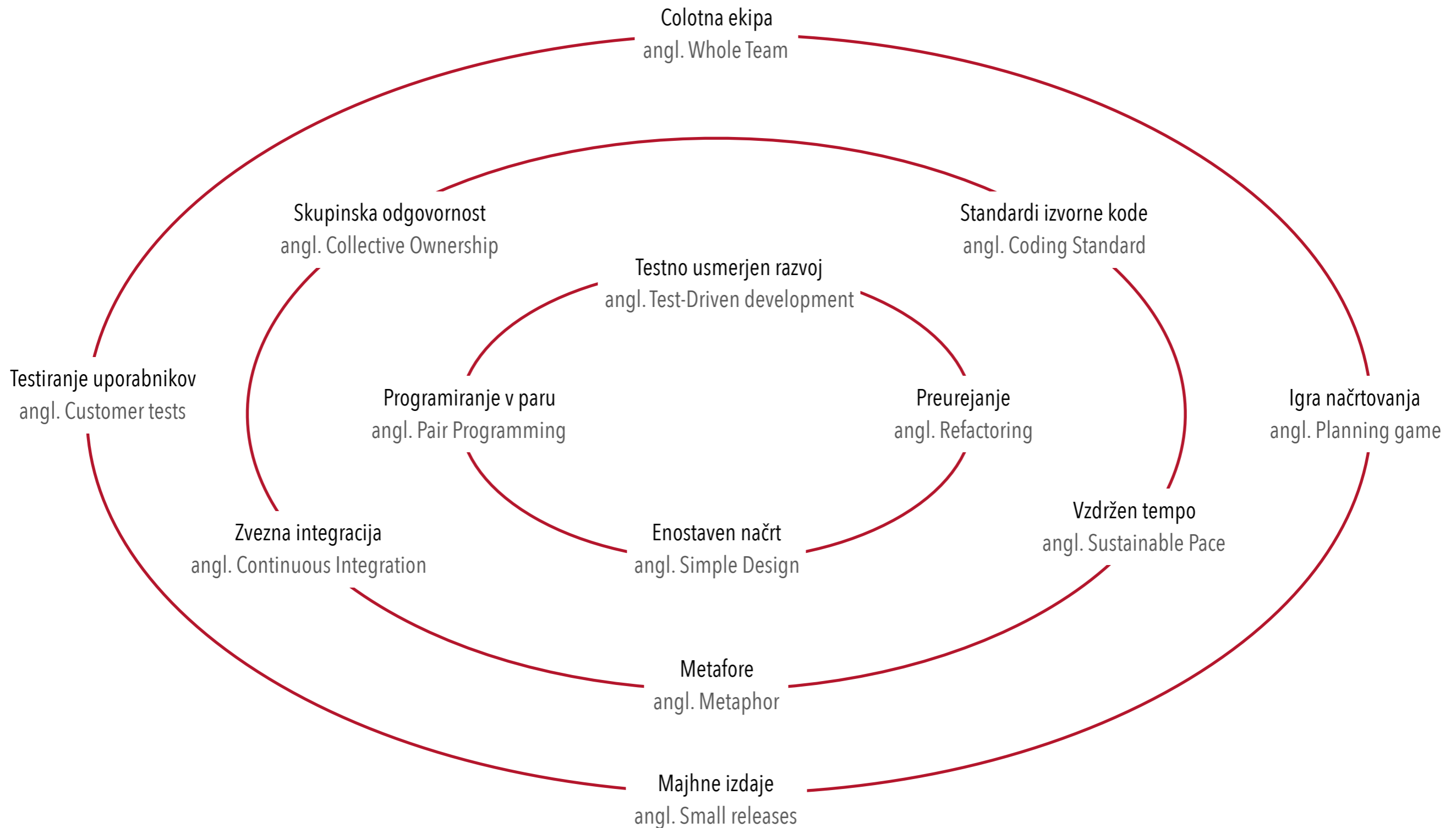
Cilj je razvoj razširljivih  
in prilagodljivih rešitev.

humanost, ekonomičnost, obojestranska  
korist, izboljšave, različnost, refleksija,  
konstanten tok, redundantnost,  
sprejemanje napak, kvaliteta,  
sprejemanje odgovornosti,  
majhne spremembe, ...

vključevanje uporabnikov,  
postopen razvoj/namestitev,  
skupna izvorna koda, deljenje izvorne  
kode, kontinuiteta razvojne ekipe, ...

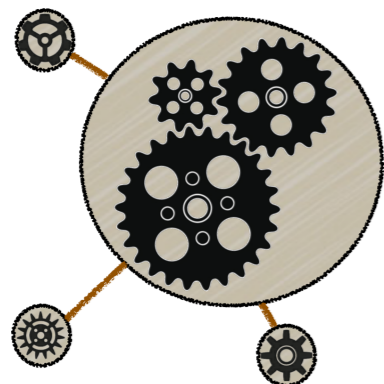
Kent Beck, Cynthia Andres, "Extreme Programming Explained: Embrace Change", Addison-Wesley, 2004.

# Razvojne prakse

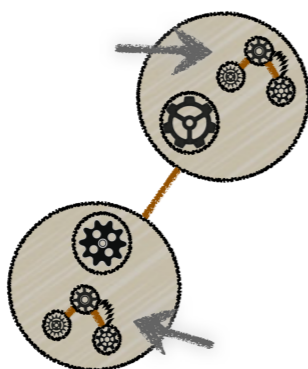


# Napake v izvorni kodi

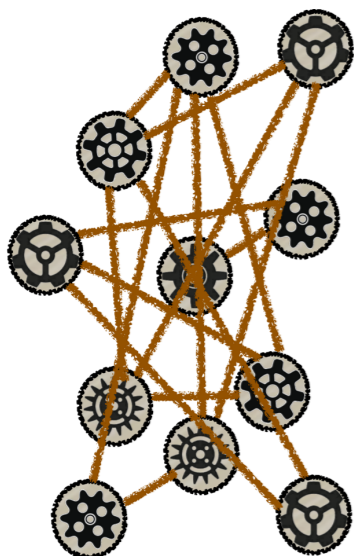
Zelo veliki razredi



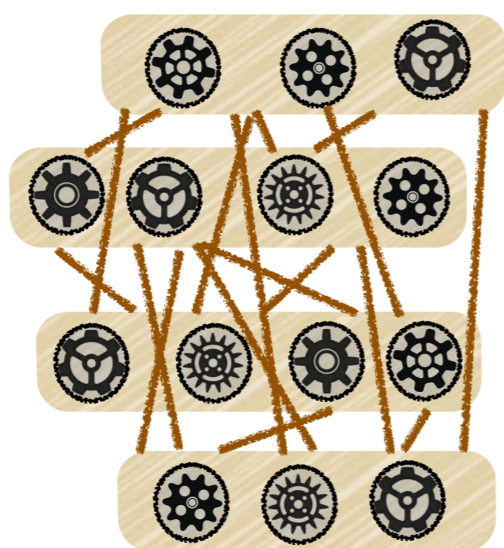
Podvajanje kode



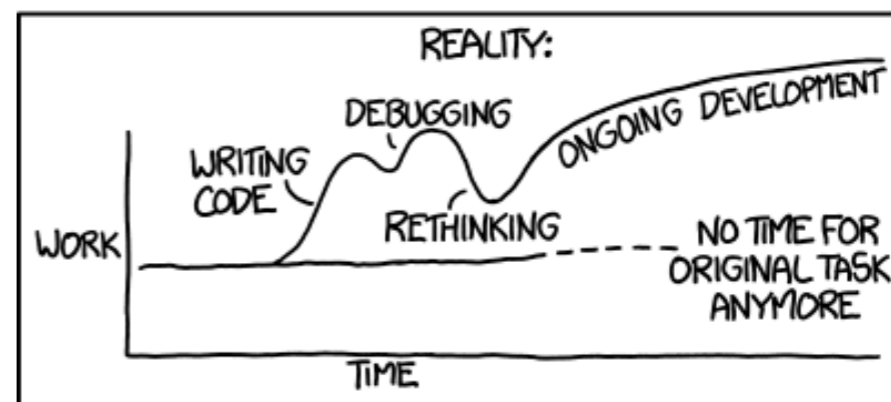
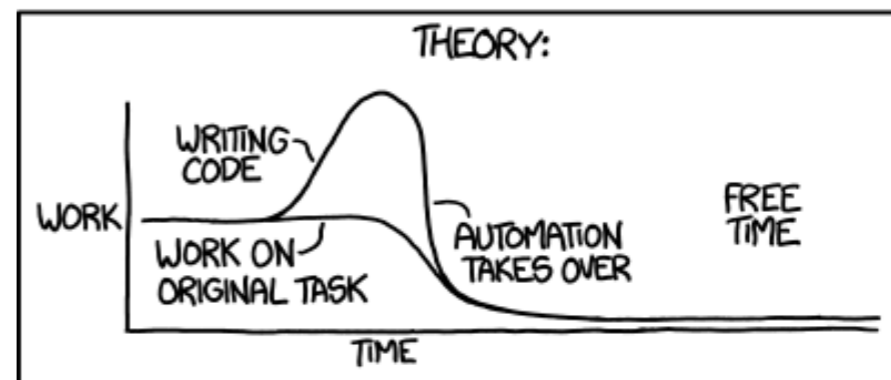
Špageti izvorna koda



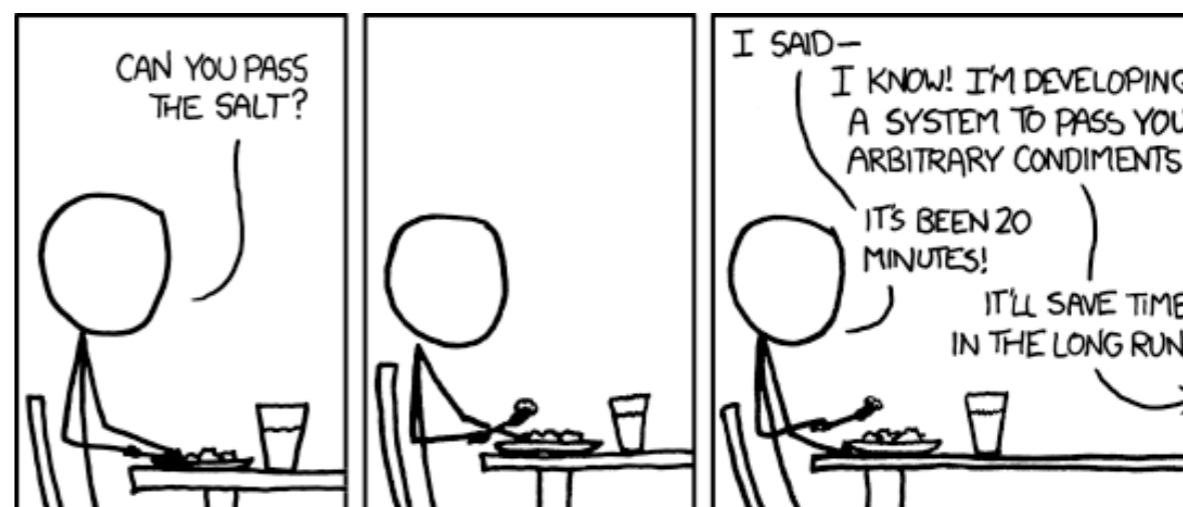
Lazanja izvorna koda



"I SPEND A LOT OF TIME ON THIS TASK.  
I SHOULD WRITE A PROGRAM AUTOMATING IT!"

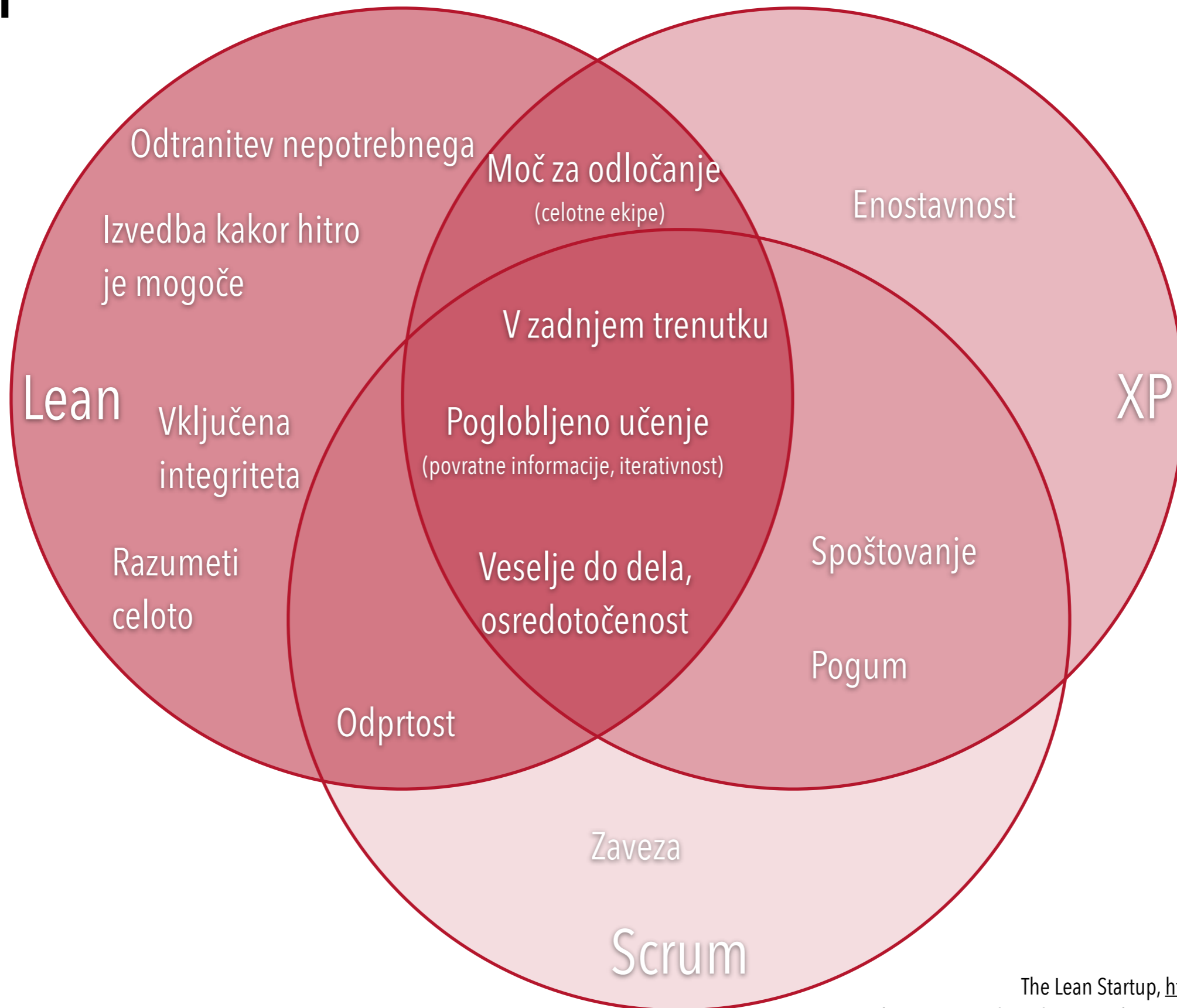


Automation, <http://xkcd.com/1319>



The general problem, <http://xkcd.com/974>

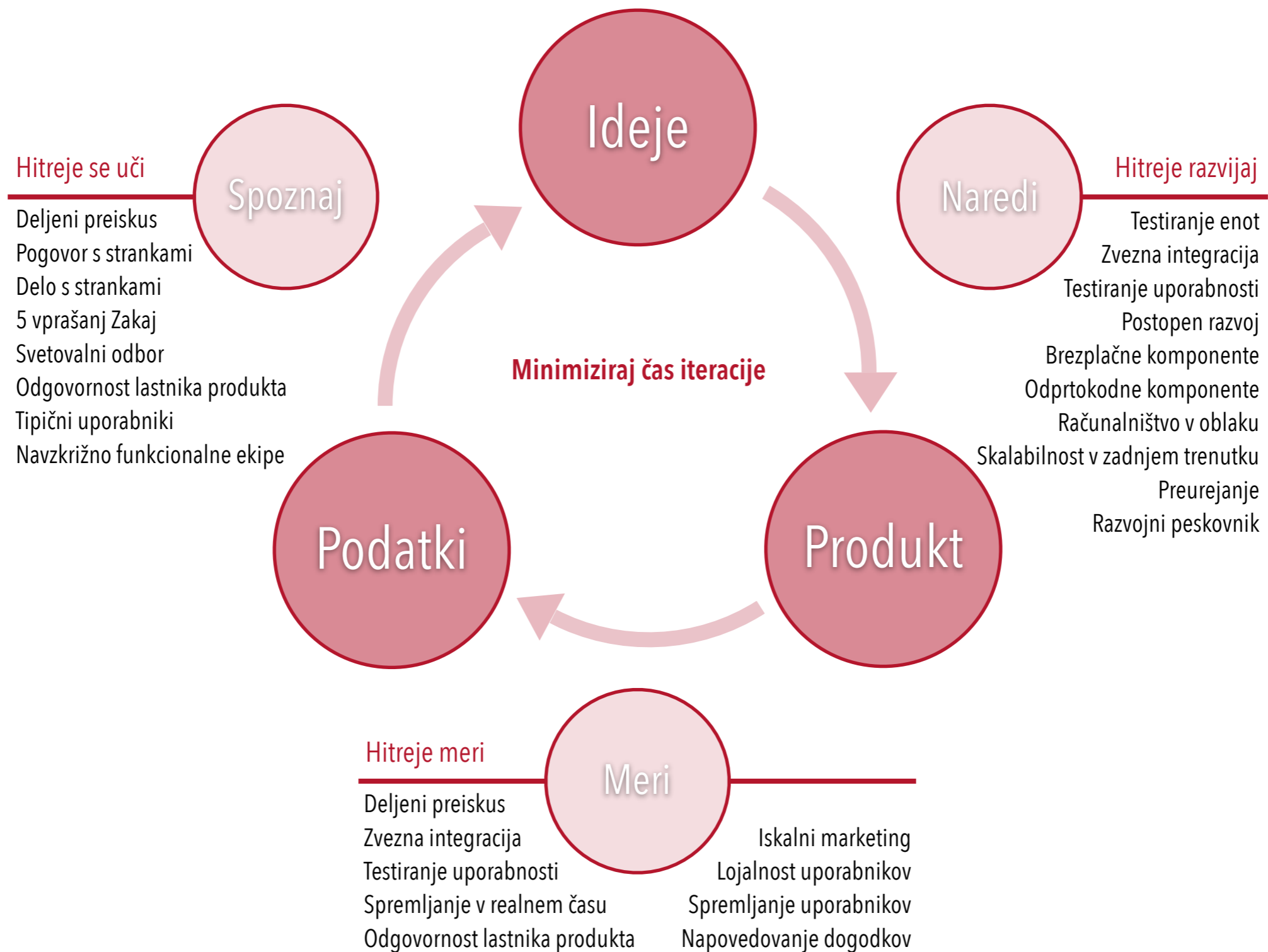
# Lean



The Lean Startup, <http://theleanstartup.com>  
Ash Maurya, "Delaj vitko", Pasadena, 2014, <http://delajvitko.si>



# Vitek proces



# Kanban

... je metoda za **upravljanje načrtovanja** produkta, s poudarkom na **iterativnem razvoju** in **usklajenem delu** razvojne ekipe.

## Osnovna načela

Začnemo s tem kar vemo

Uvajamo postopne izboljšave

V začetku ohranimo obstoječe naslove, vloge in odgovornosti

## Osnovne prakse

Vizualizacija dela/nalog

Omejitev tekočega dela (Work In Progress - WIP)

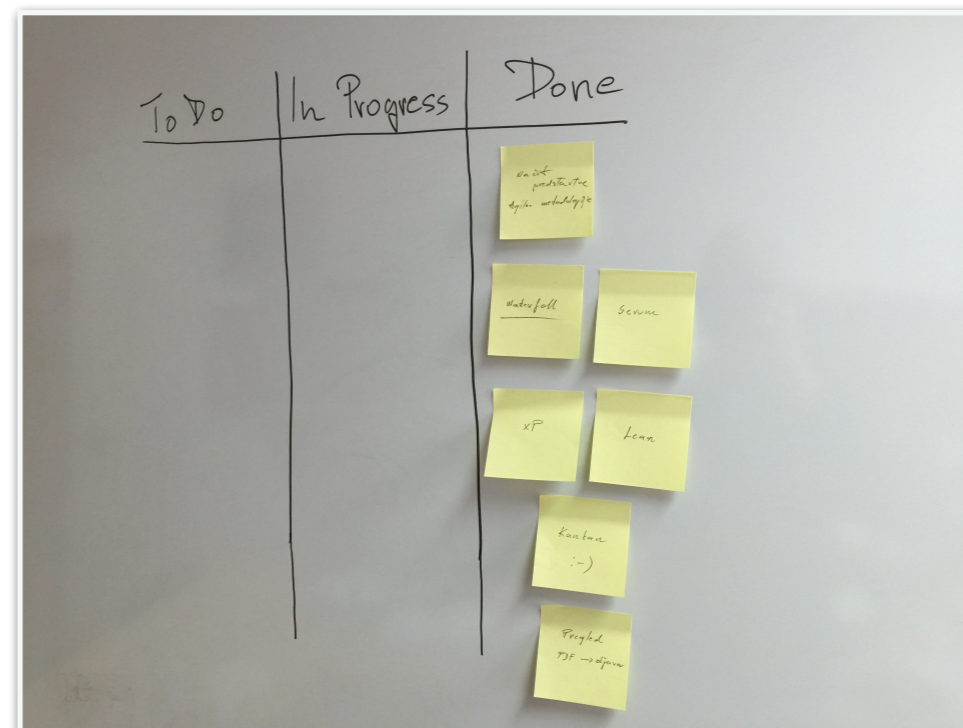
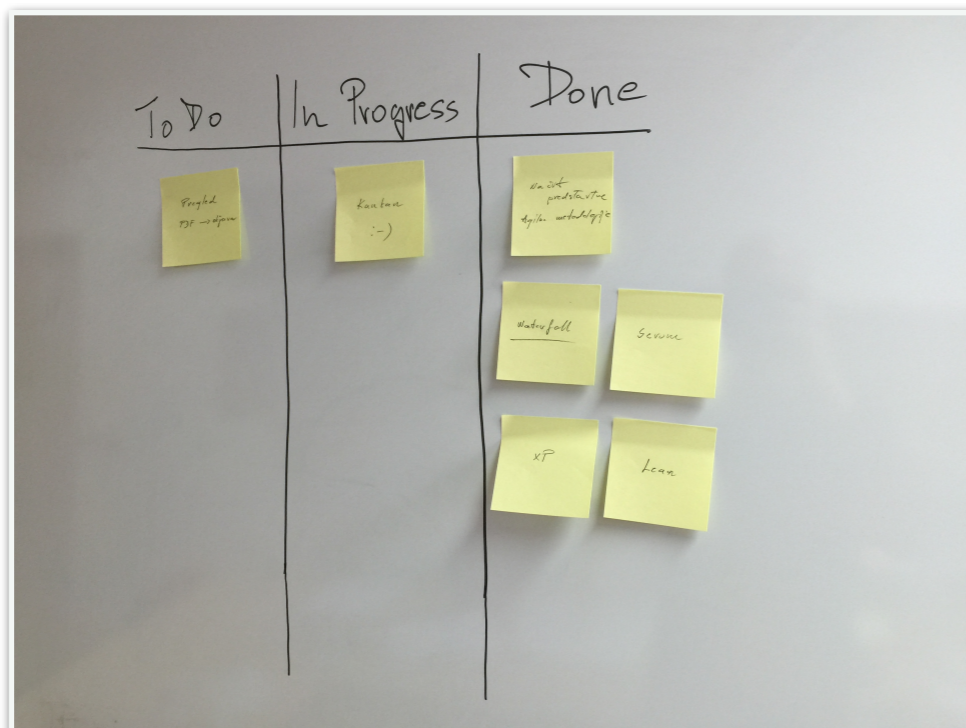
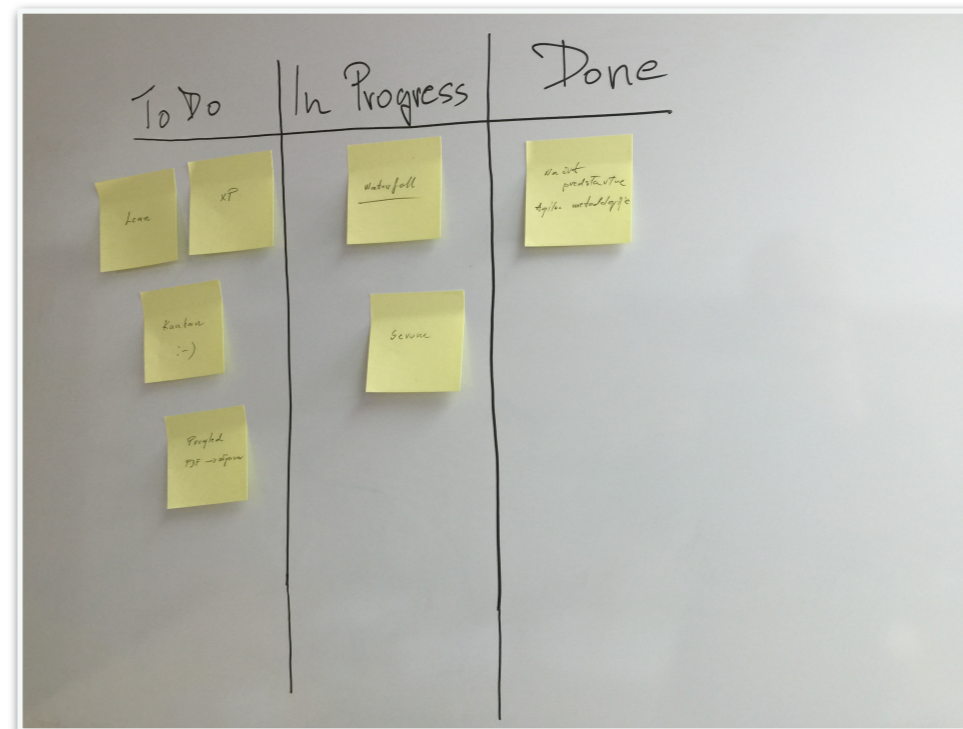
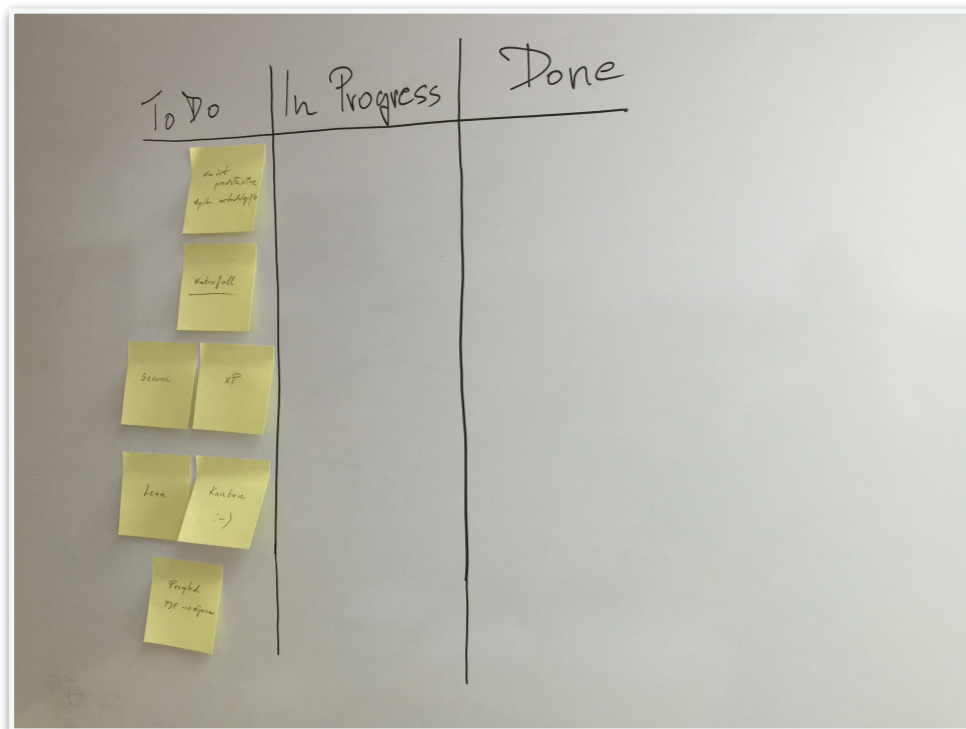
Nadzor nad tokom dela

Eksplicitne procesne politike

Vzpostavimo zanko za povratne informacije

Izboljšamo sodelovanje, eksperimentalni razvoj

# Kanban tabla



**HITROST** prilagodljivost  
**AGILNE metodologije**  
sodelovanje **KVALITETA** iterativnost  
odprtost

To ni proces, temveč **filozofija** oziroma **nabor vrednot**