

Uvod u softversko inženjerstvo

Čiste aplikacije

Nikola Luburić
nikola.luburic@uns.ac.rs

```
        message =  
        if not hasattr(self, '_headers_buffer'):  
            self._headers_buffer = []  
        self._headers_buffer.append((" %s %d %s\r\n" %  
            (self.protocol_version, code, message)).en  
            'latin-1', 'strict'))  
  
    def end_header(self, keyword, value):  
        """Send a MIME header to the headers buffer."""  
        if self.request_version != 'HTTP/0.9':  
            if not hasattr(self, '_headers_buffer'):  
                self._headers_buffer = []  
            self._headers_buffer.append(  
                ("%s: %s\r\n" % (keyword, value)).encode('lati  
            keyword.lower() == 'connection':  
                if value.lower() == 'close':  
                    self.close_connection = True  
                elif value.lower() == 'keep-alive':  
                    self.close_connection = False
```

Kako organizovati pakete?

Kako postaviti arhitekturu?

Čemu sve ovo?

Kako izgleda u praksi?

Čiste aplikacije

Primeri

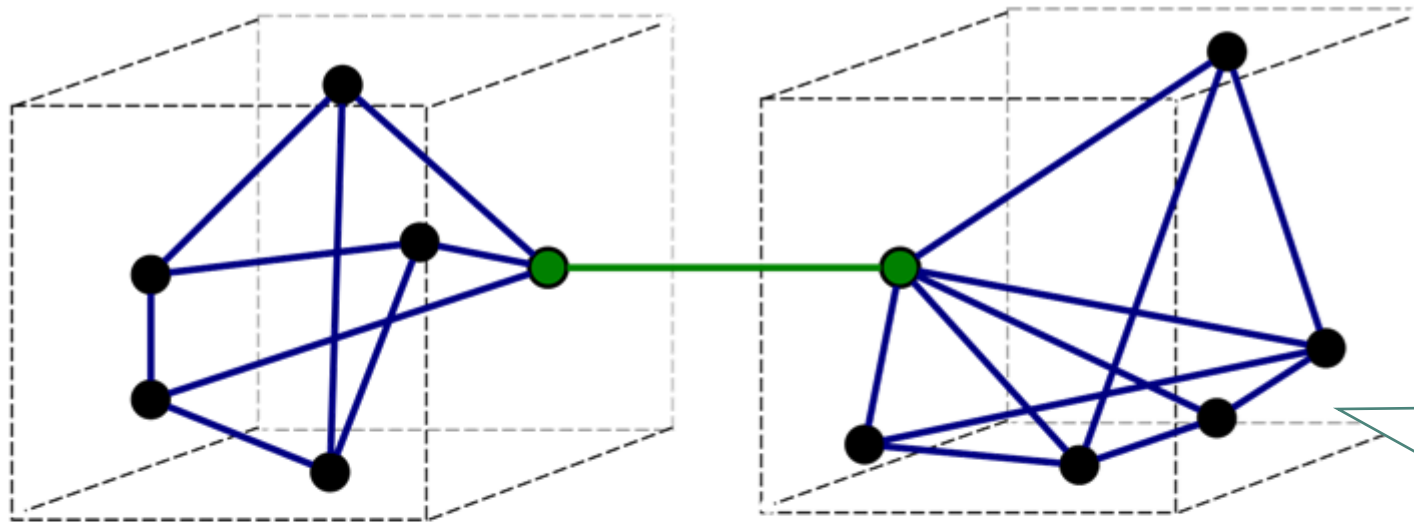
org.hibernate.sessions.exceptions
Zdravo.PatientSystem.HealthMonitor

Struktura

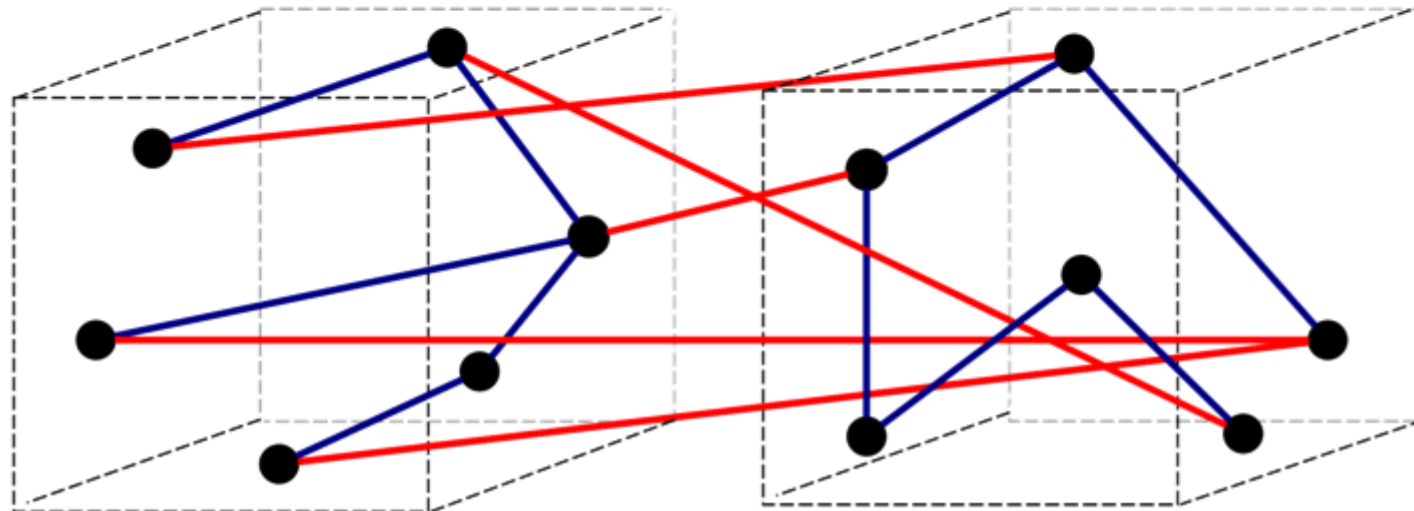
(<Company>|<Domain>).[(<Product>|<Technology>)]
[.<Feature>][.<Segment>]

Resursi












<https://docs.oracle.com/javase/tutorial/java/package/namingpkgs.html>
<https://docs.microsoft.com/en-us/dotnet/standard/design-guidelines/>


















*da li važi za
trenutne
pakete?*



ZdravoKorporacija



































- ▶  Dependencies
- ▶  Controller
- ▶  DTO
- ▶  Model
- ▶  Repository
- ▶  Resources
- ▶  Service
- ▶  View
- ▶  App.xaml
- ▶  AssemblyInfo.cs
- ▶  MainWindow.xaml

Service

- ▶  `AbsenceRequestService.cs`
- ▶  `AdvancedRenovationJoiningService.cs`
- ▶  `AdvancedRenovationSeparationService.cs`
- ▶  `AnamnesisService.cs`
- ▶  `AppointmentService.cs`
- ▶  `BasicRenovationService.cs`
- ▶  `DoctorService.cs`
- ▶  `EquipmentService.cs`
- ▶  `ManagerService.cs`
- ▶  `MedicalRecordService.cs`
- ▶  `MedicationService.cs`
- ▶  `NotificationService.cs`
- ▶  `PatientService.cs`
- ▶  `PrescriptionService.cs`
- ▶  `RatingService.cs`
- ▶  `RoomService.cs`
- ▶  `SecretaryService.cs`

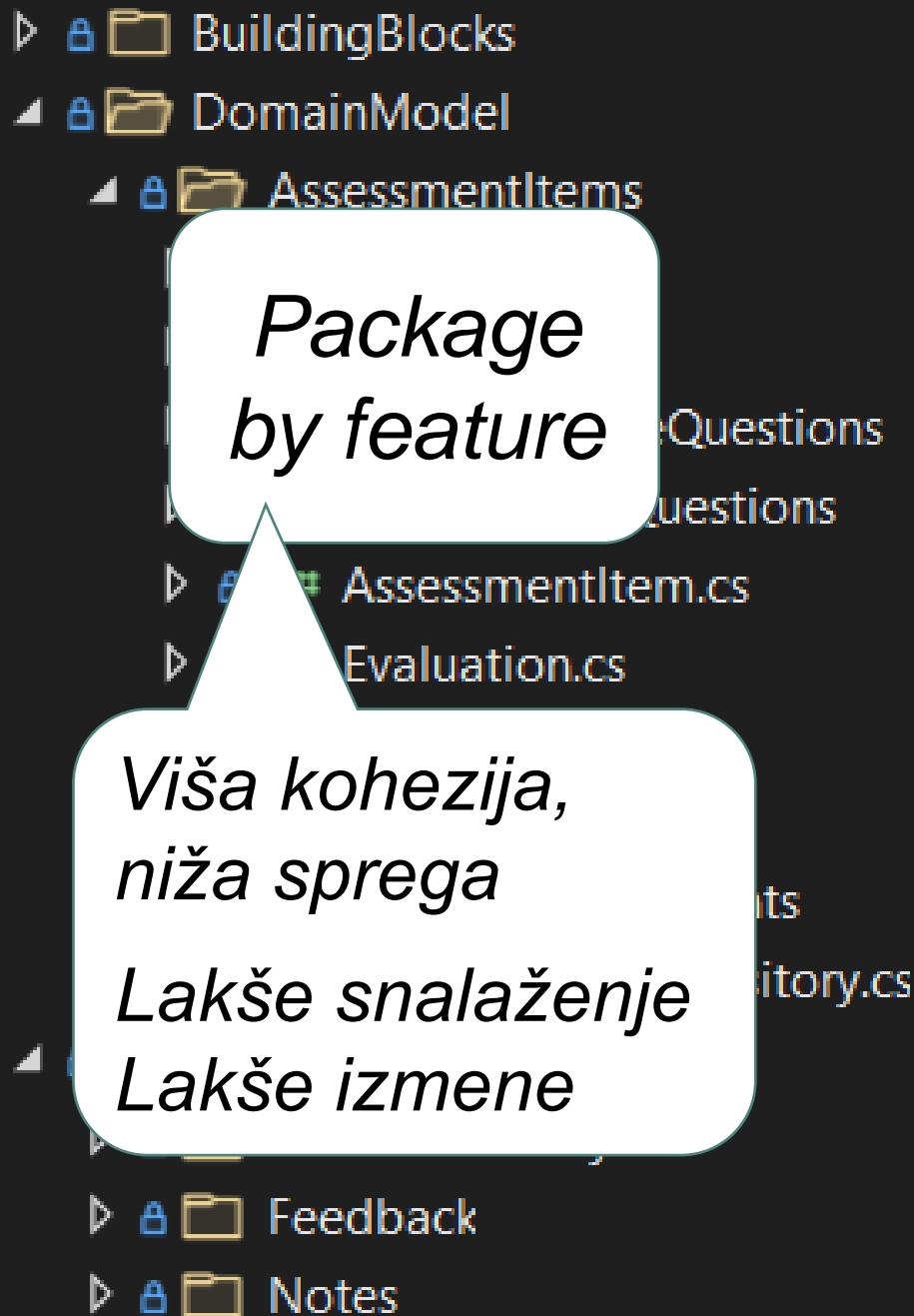
*šta otvaram
ako dodajem
novu funkciju
za pacijente?*

Paketiranje – visoka kohezija

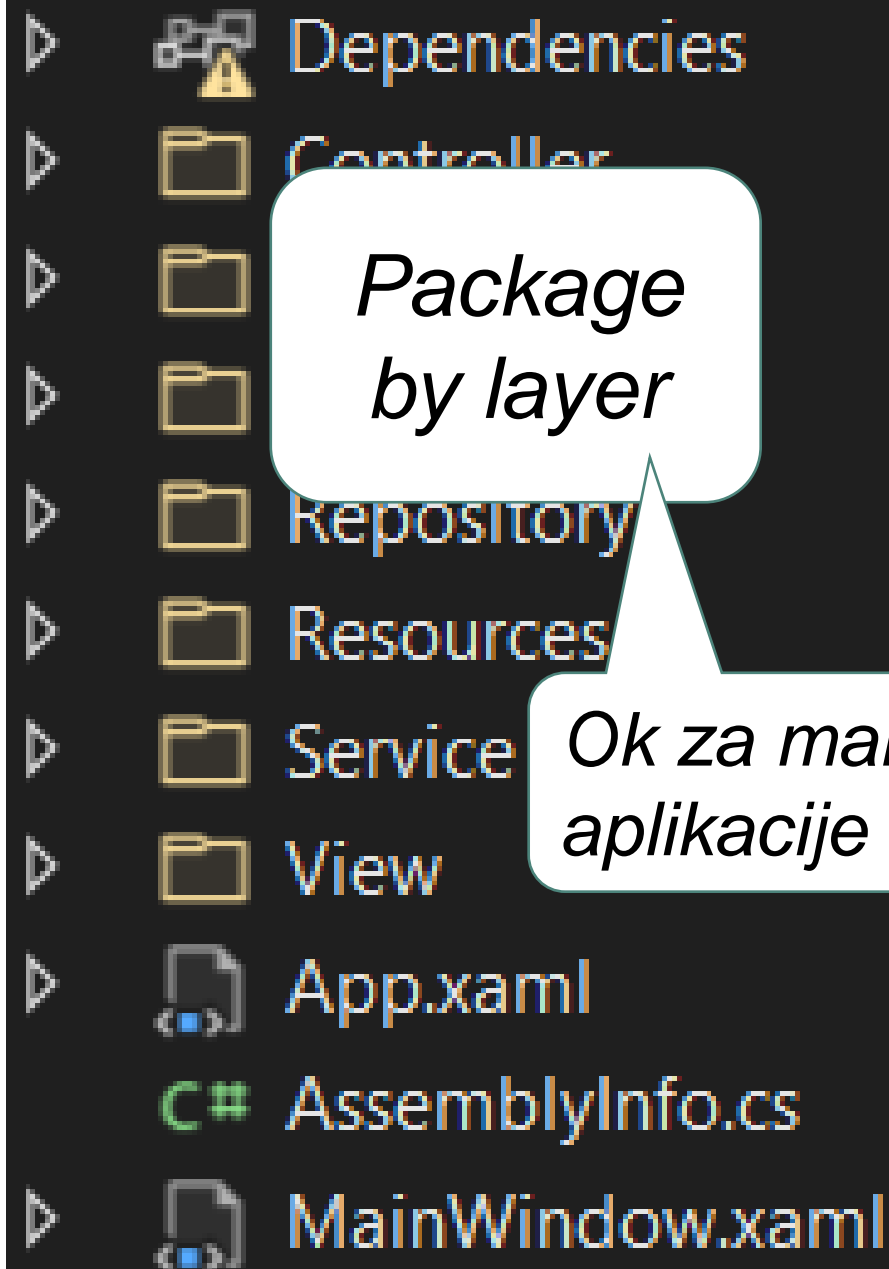
-   BuildingBlocks
-   DomainModel
 -   AssessmentItems
 -   ArrangeTasks
 -   Challenges
 -   MultiResponseQuestions
 -   ShortAnswerQuestions
 -   AssessmentItem.cs
 -   Evaluation.cs
 -   Submission.cs
 -   InstructionalItems
 -   KnowledgeComponents
 -   IKnowledgeUnitRepository.cs
-   LearnerModel
 -   DomainOverlay
 -   Feedback
 -   Notes

*šta otvaram
ako dodajem
novo pravilo za
izazove?*

Paketiranje – visoka kohezija



C# ZdravoKorporacija



Locate

Intuitivno i jednostavno pronađi šta ti treba

Identify

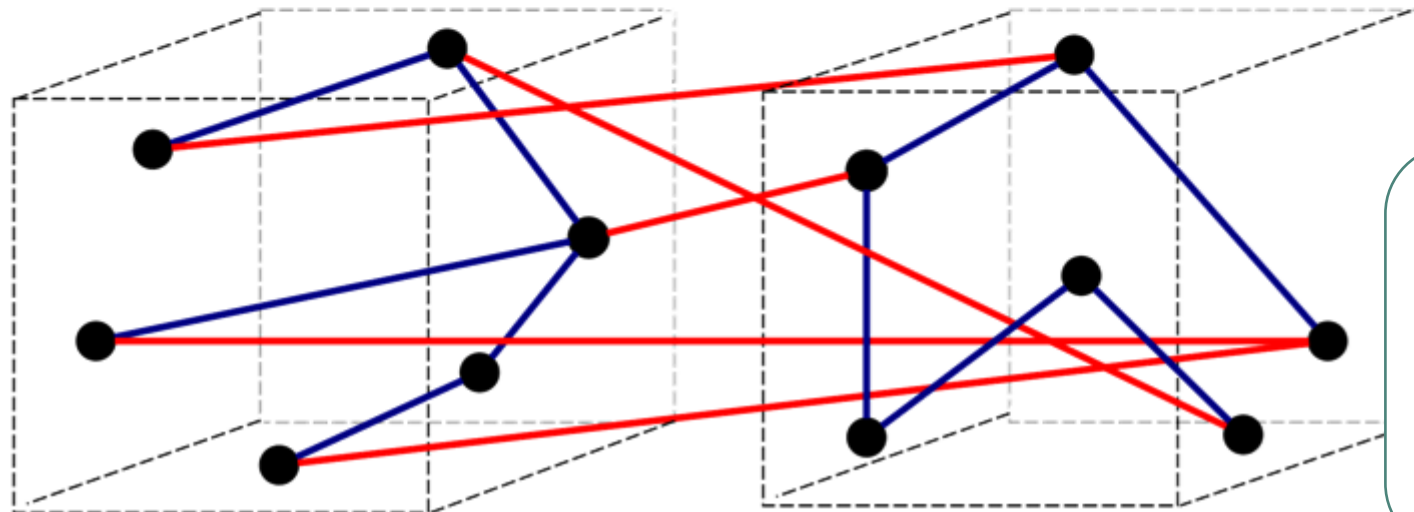
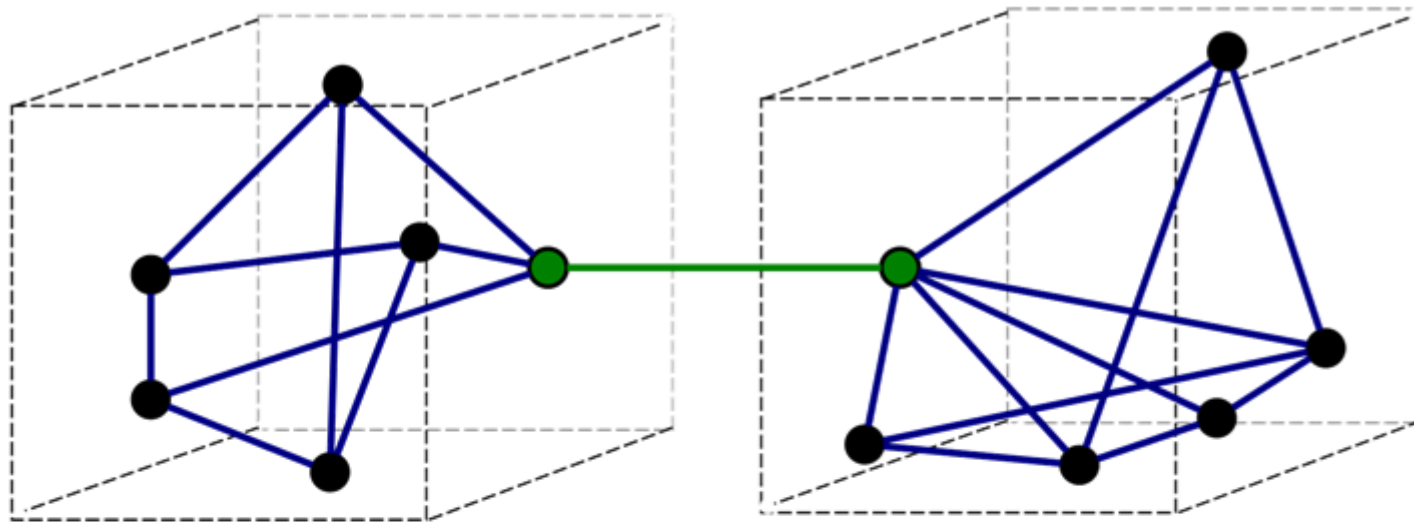
Razumi sadržaj spram naziva

Flat

Izbegavati duboko ugnježdavanje

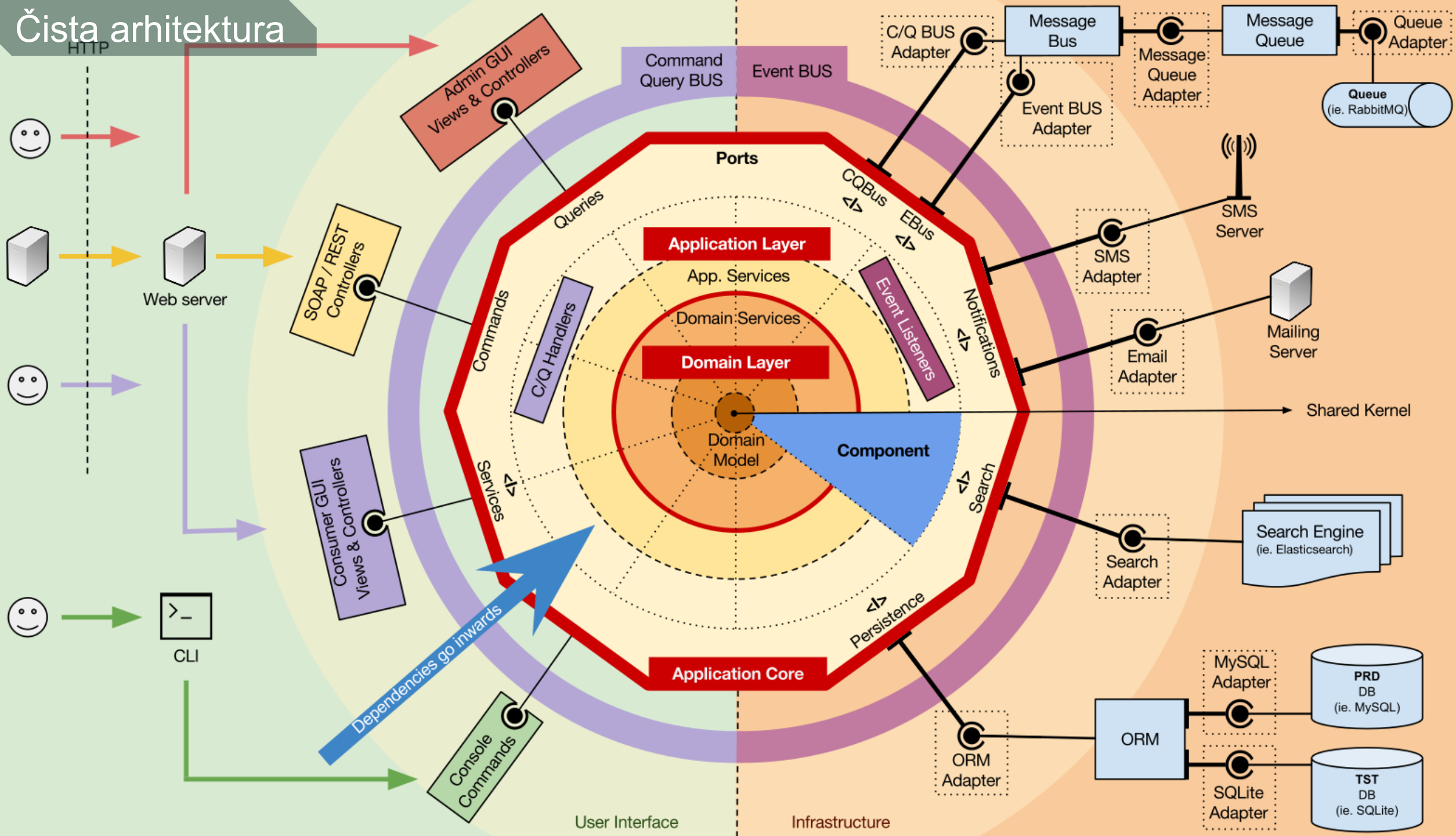
Try to be DRY

Izbegavati redundantnost u nazivu

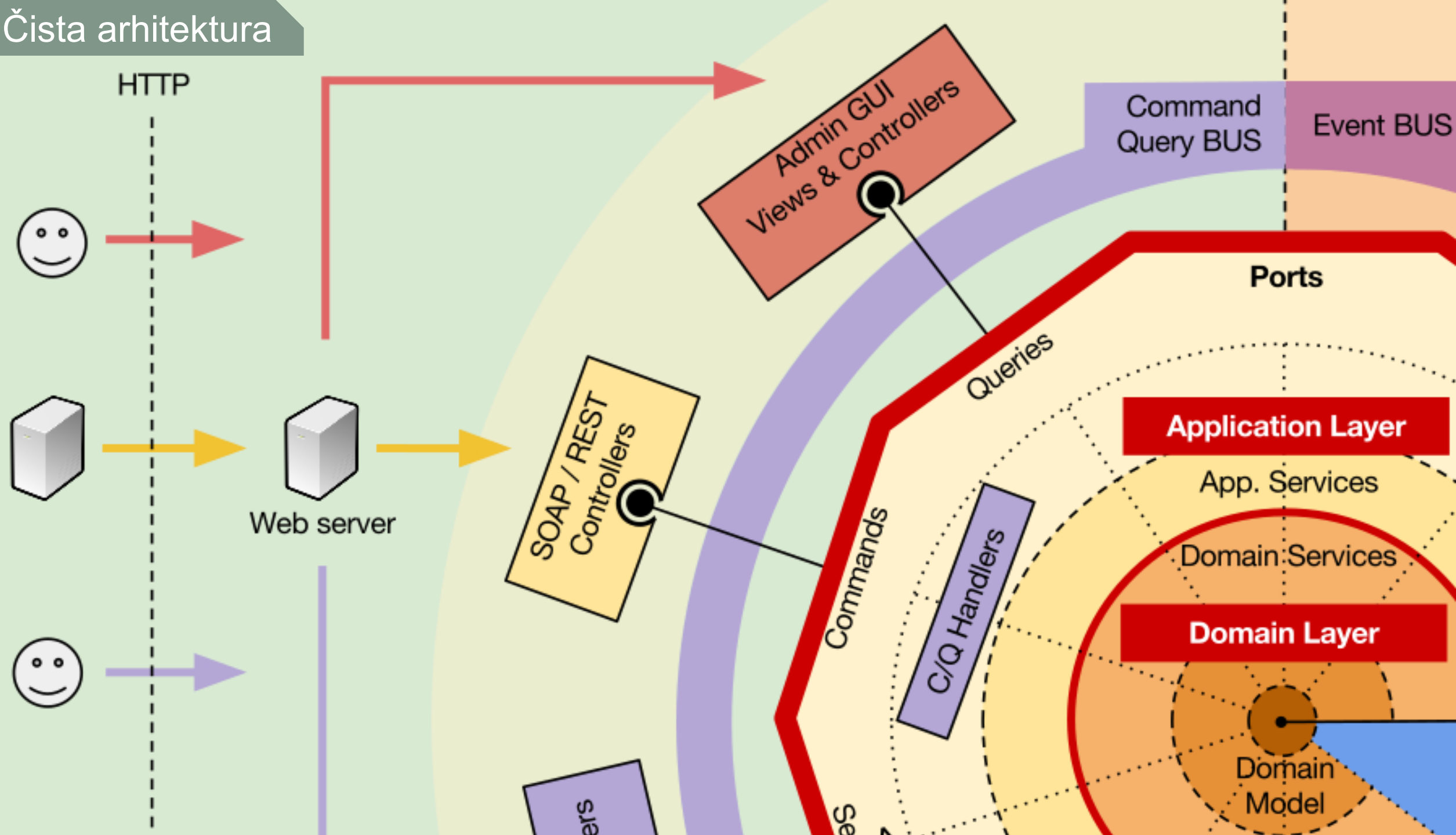


*Avoid
namespace
mutually
dependent*

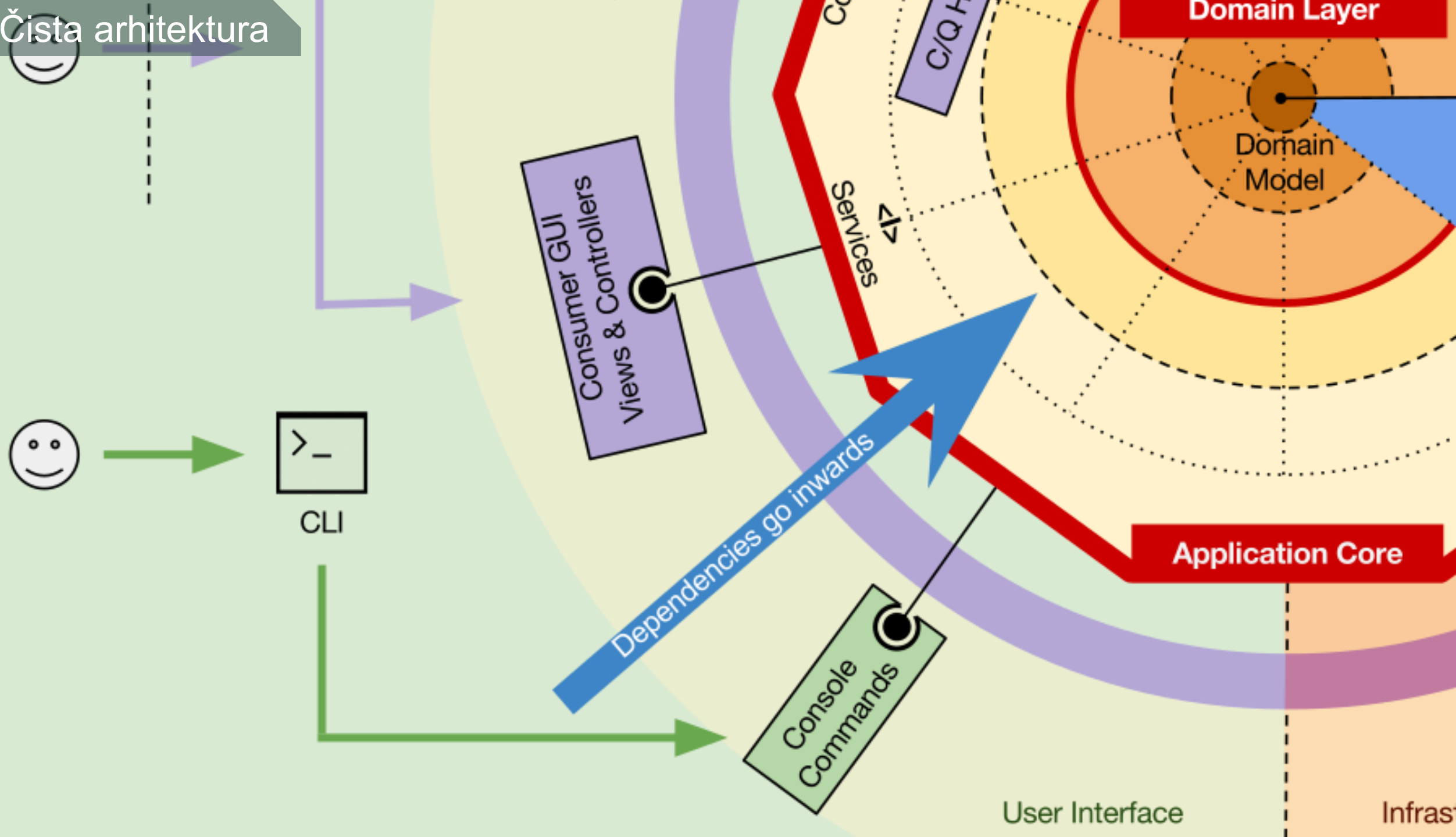
Čista arhitektura



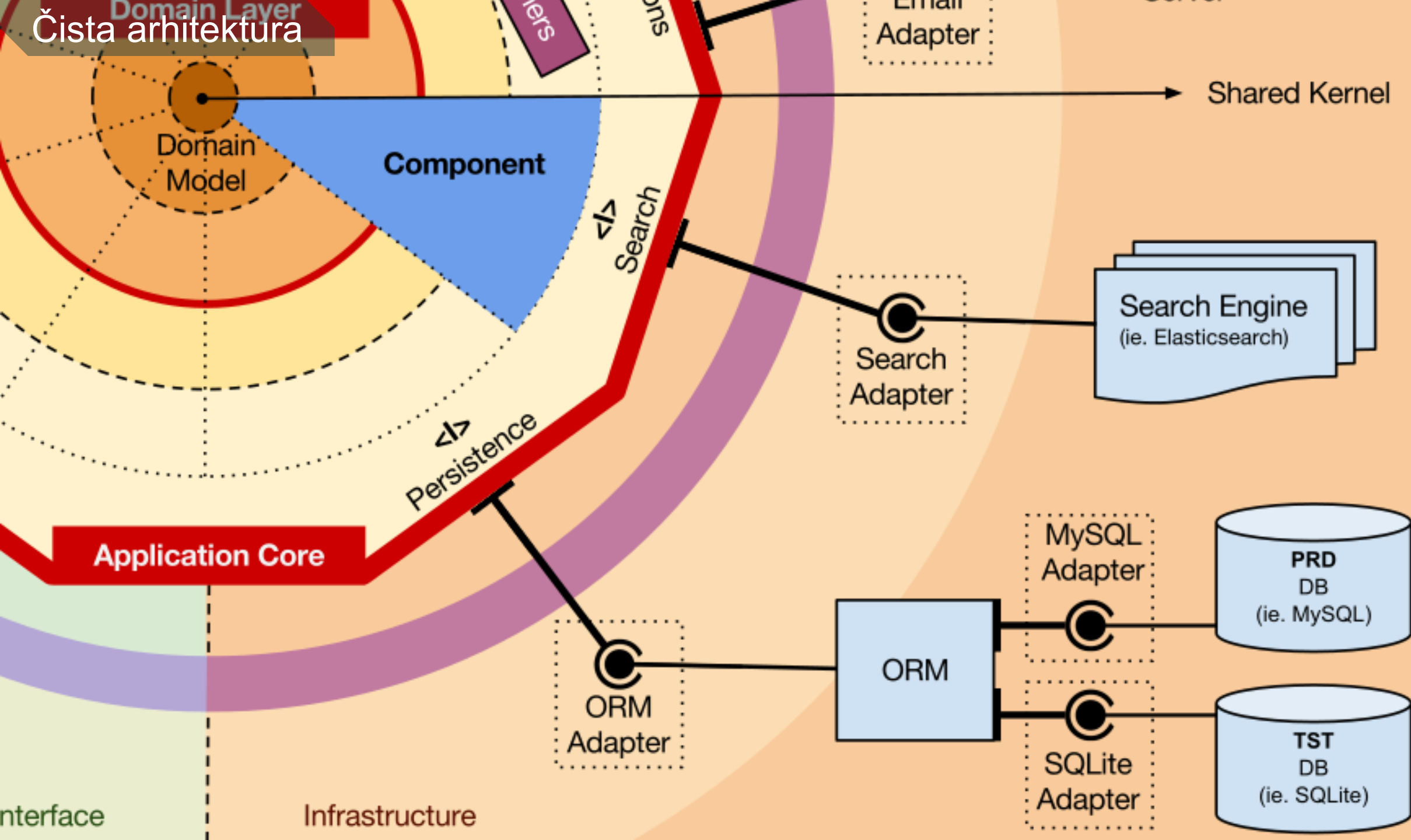
Čista arhitektura



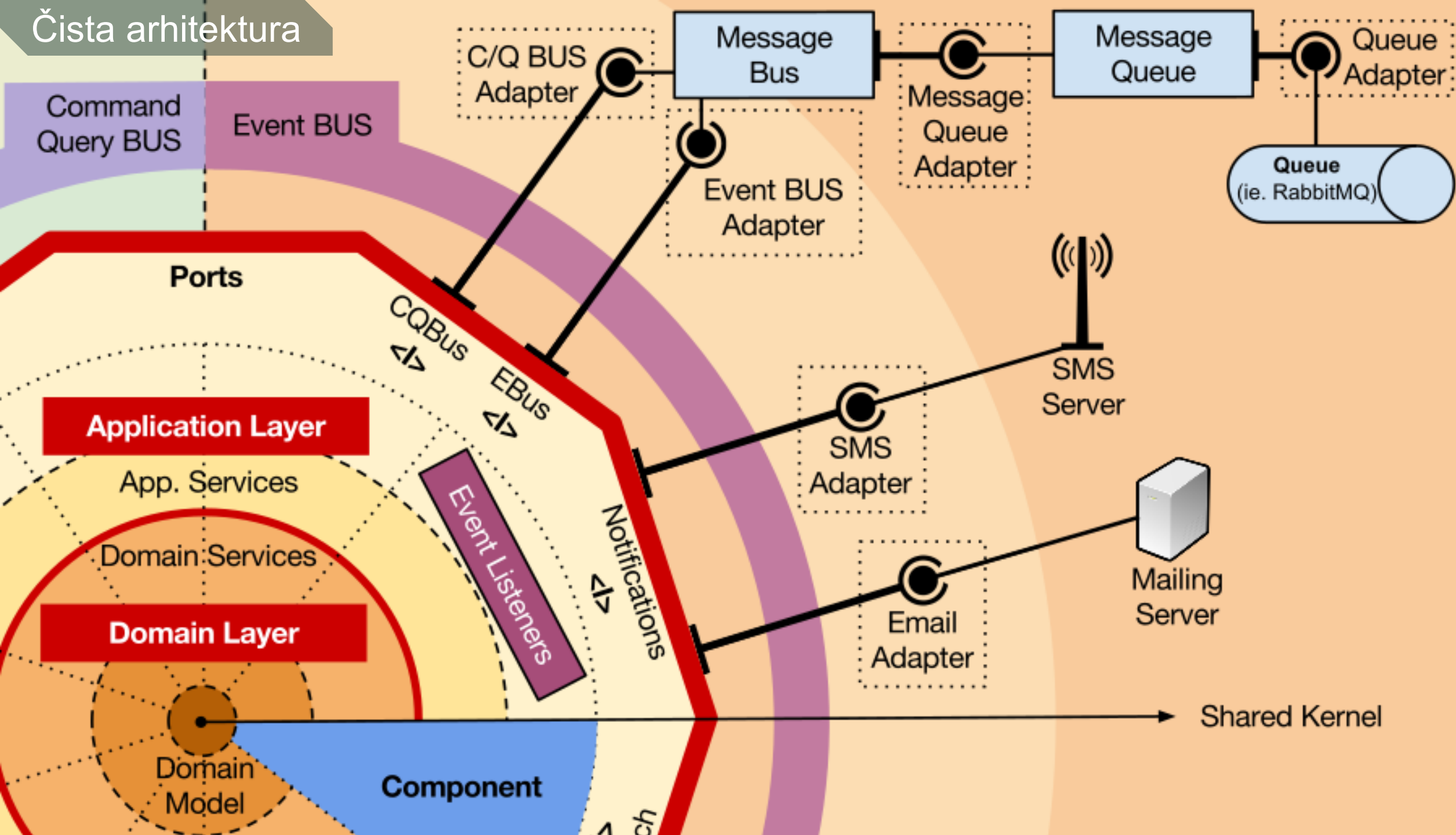
Čista arhitektura

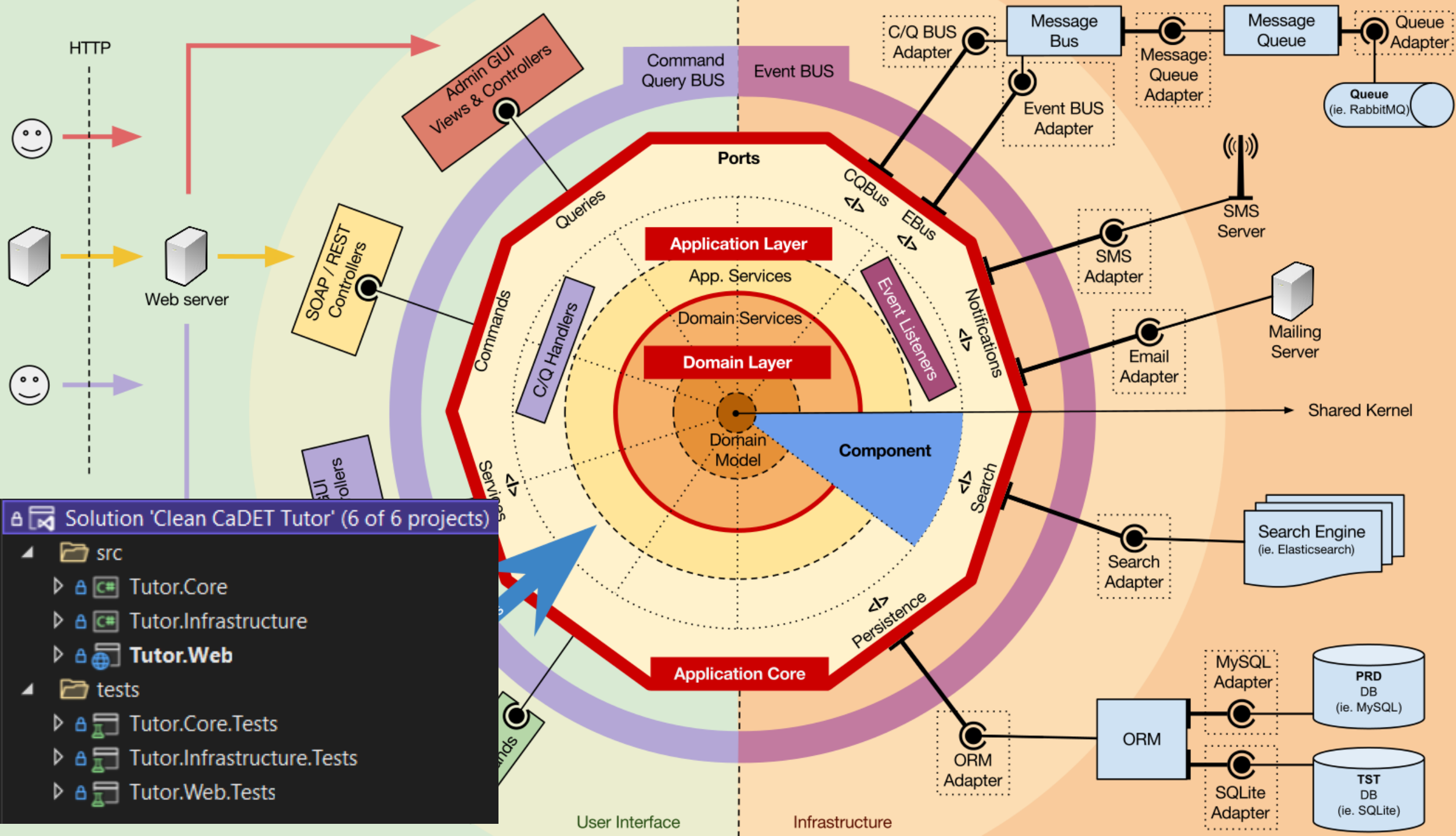


Čista arhitektura



Čista arhitektura

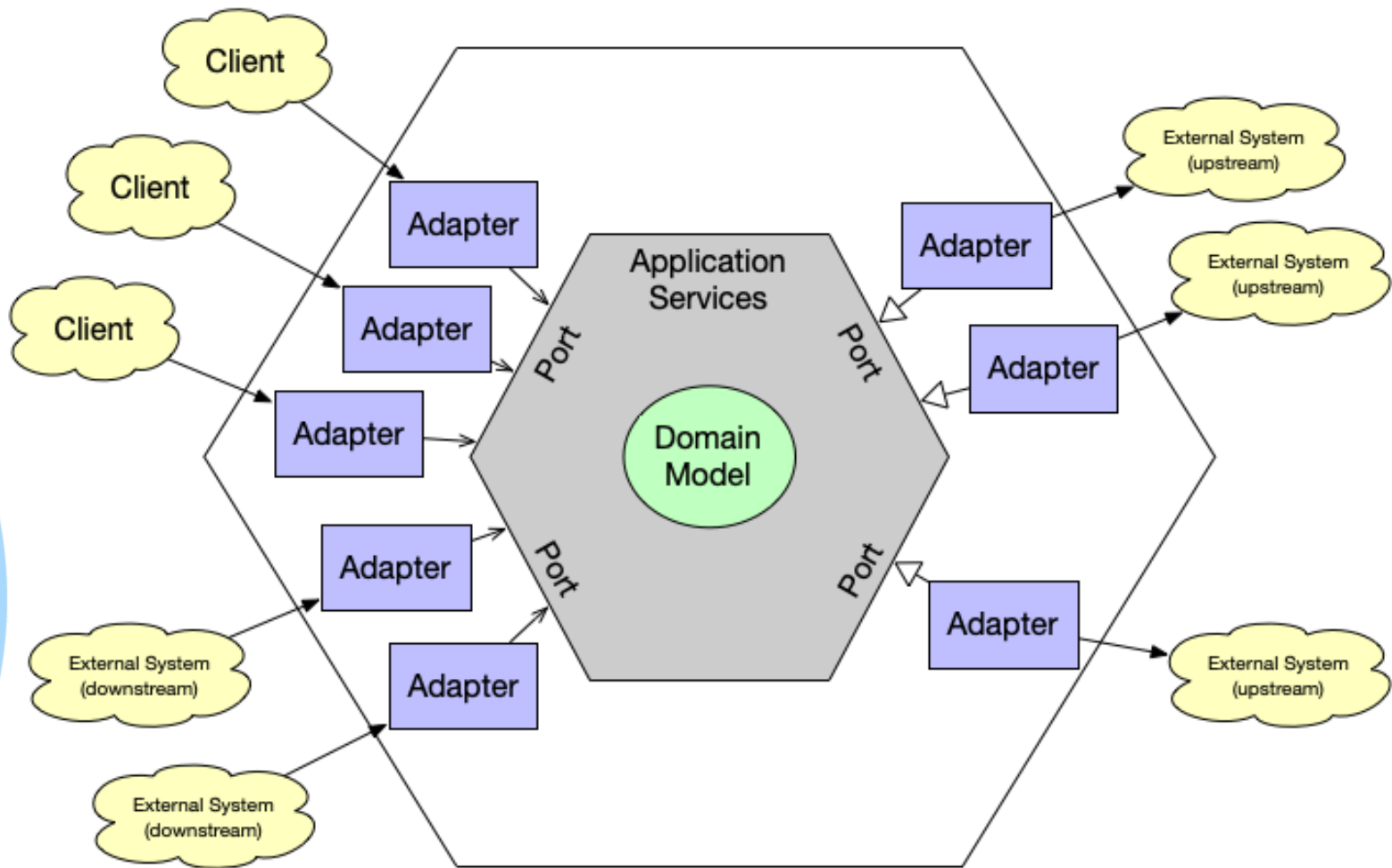
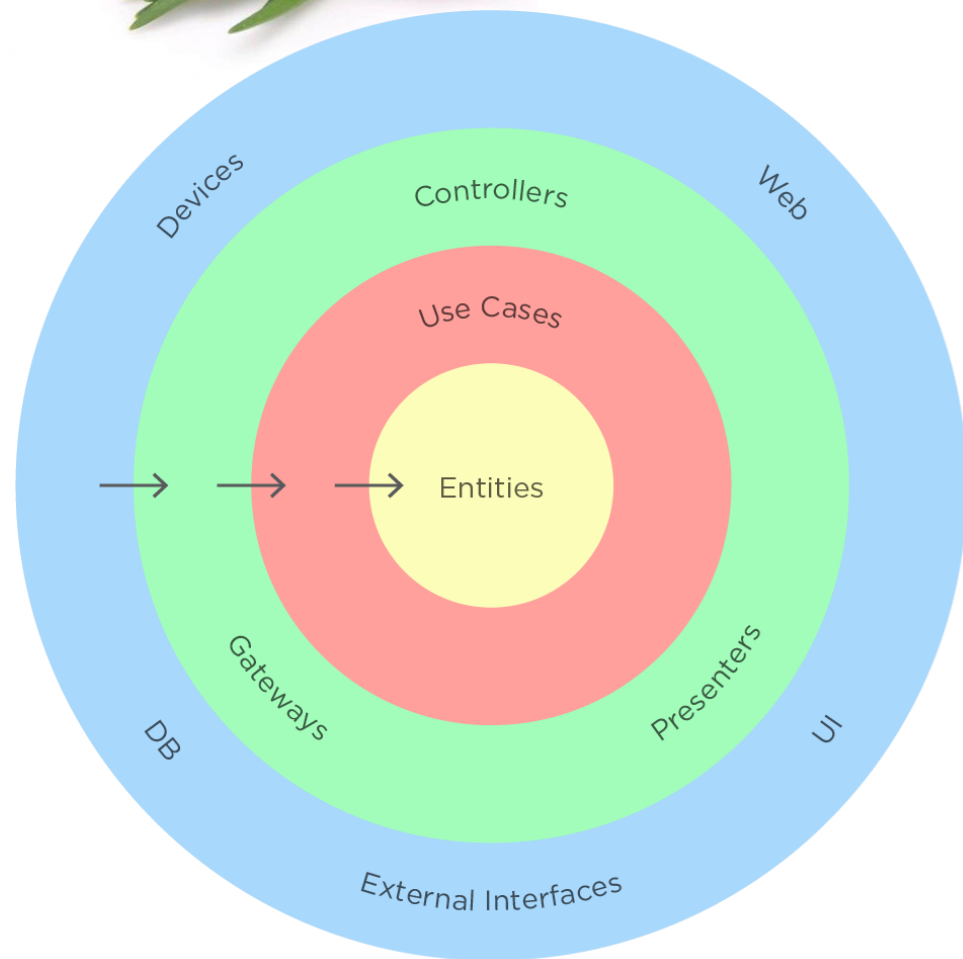




Solution 'Clean CaDET Tutor' (6 of 6 projects)

- src
 - Tutor.Core
 - Tutor.Infrastructure
 - Tutor.Web
- tests
 - Tutor.Core.Tests
 - Tutor.Infrastructure.Tests
 - Tutor.Web.Tests

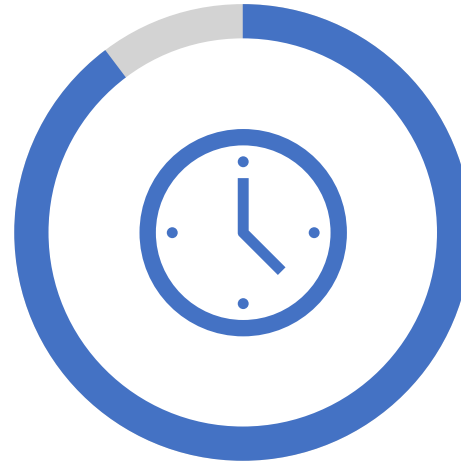
Clean / Hexagonal / Onion Architecture





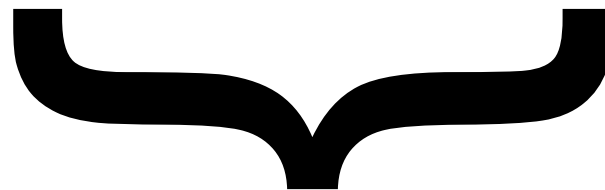
Dogovorite timske konvencije

- ❖ Čist kod pospešuje komunikaciju
- ❖ Kod koji prati timska pravila pospešuje timsku komunikaciju
- ❖ Paketiranje, imenovanje, indentacija, formatiranje, širina koda, dužina datoteka...



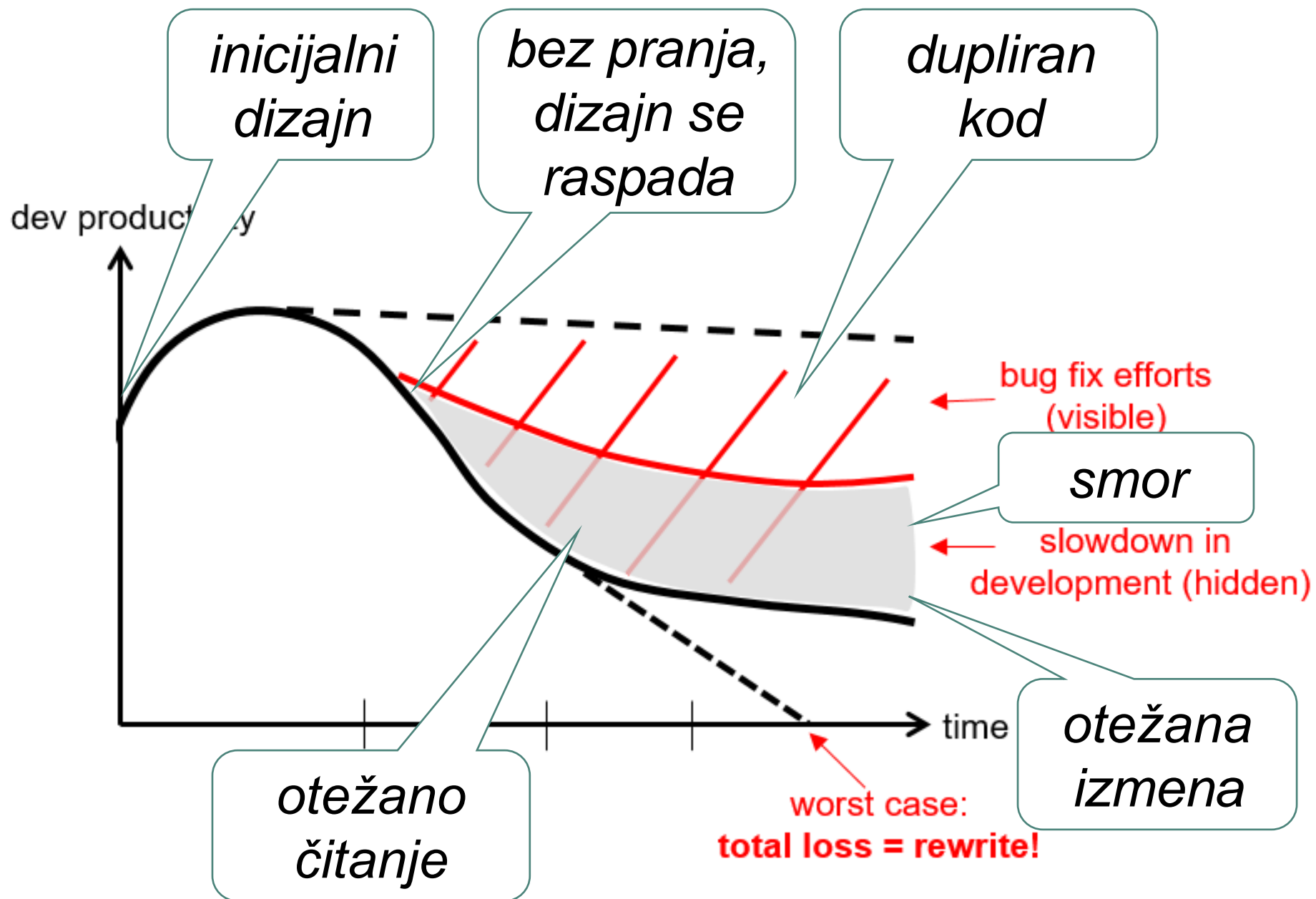
Održavanje košta
40 do 80%
ukupnog troška

Na zrelom projektu
odnos čitanja i
pisanja je 10:1

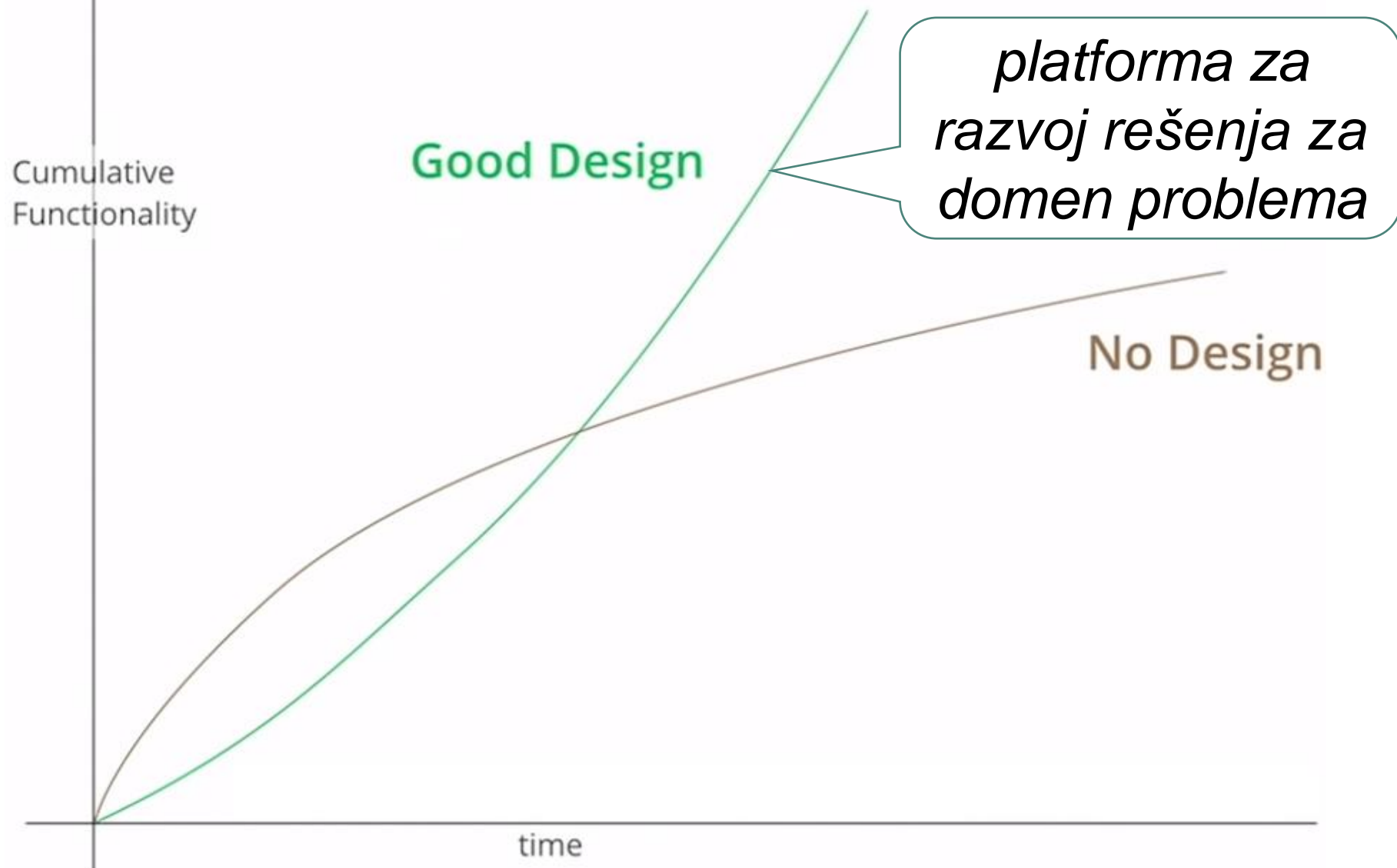


Većina troška razvoja je u čitanju koda
Čist dizajn koda **značajno** smanjuje troškove

Značaj čistog koda



Design Stamina Hypothesis



Zašto pišemo prljav kod?

Nedostatak
podrške
menadžmenta

Nedostatak
veštine i brige inženjera

Zašto pišemo prljav kod?

Nedostatak
veštine i **brige** inženjera

Zašto pišemo prljav kod?

Kako rešiti nedostatak veštine i brige inženjera?

~5 predavanja

~5 vežbi

~5 knjiga

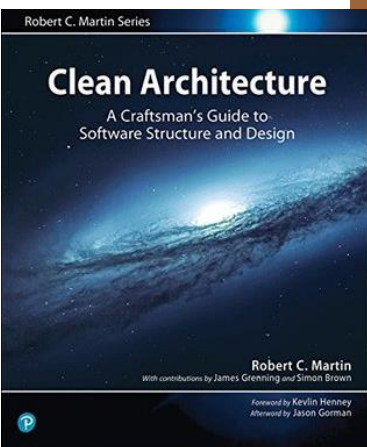
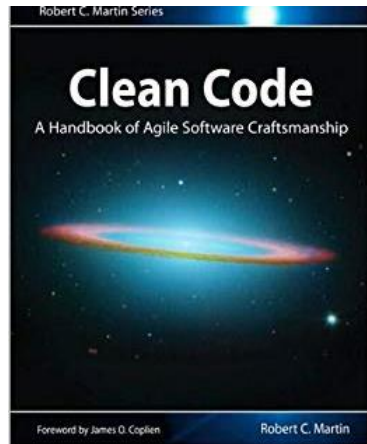
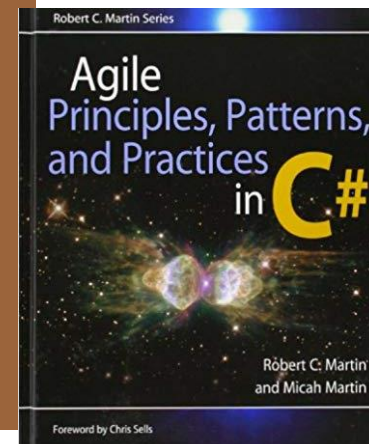
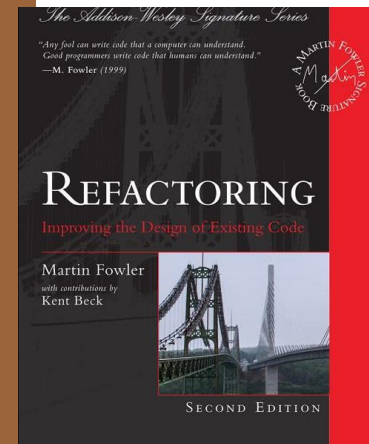
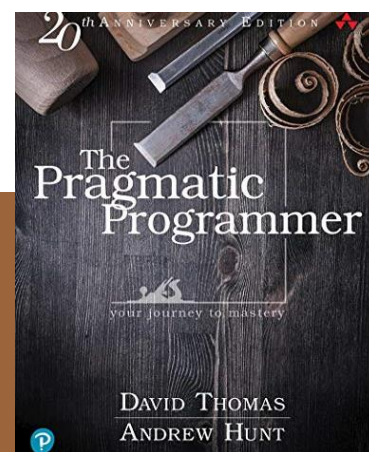
Ozbiljan projekat
(i svi budući...)

Novac
Ugled

Zabava
Izazovi

Ispunjenost
Smisao

Sloboda



Kako da integrišemo čišćenje koda u posao?

Izdvoj vreme za refaktorisanje u procene

Izbori se za karticu „tehničko unapređenje“

Kako da integrišemo čišćenje koda u posao?

refactor: LearnerId derivation redesign ✓

#43 by Luburic was merged 17 days ago

1

feat: Learner groups and group membership. ✓

#44 by Luburic was merged 18 days ago

1

refactor: assessment item mastery ✓

#43 by Luburic was merged 25 days ago

15

feat: Adds endpoint for serving events. ✓

#42 by Luburic was merged 24 days ago

1

feat: users and roles ✓

#41 by Luburic was merged on Apr 26

1

refactor: domain overlay restructuring ✓

#40 by Luburic was merged on Apr 26

1

feat: User is able to save/update/delete notes. ✓

#39 by lukaDoric was merged on Apr 7 • Approved

3

feat: Record instructor feedback and fetch learner name/surname. ✓

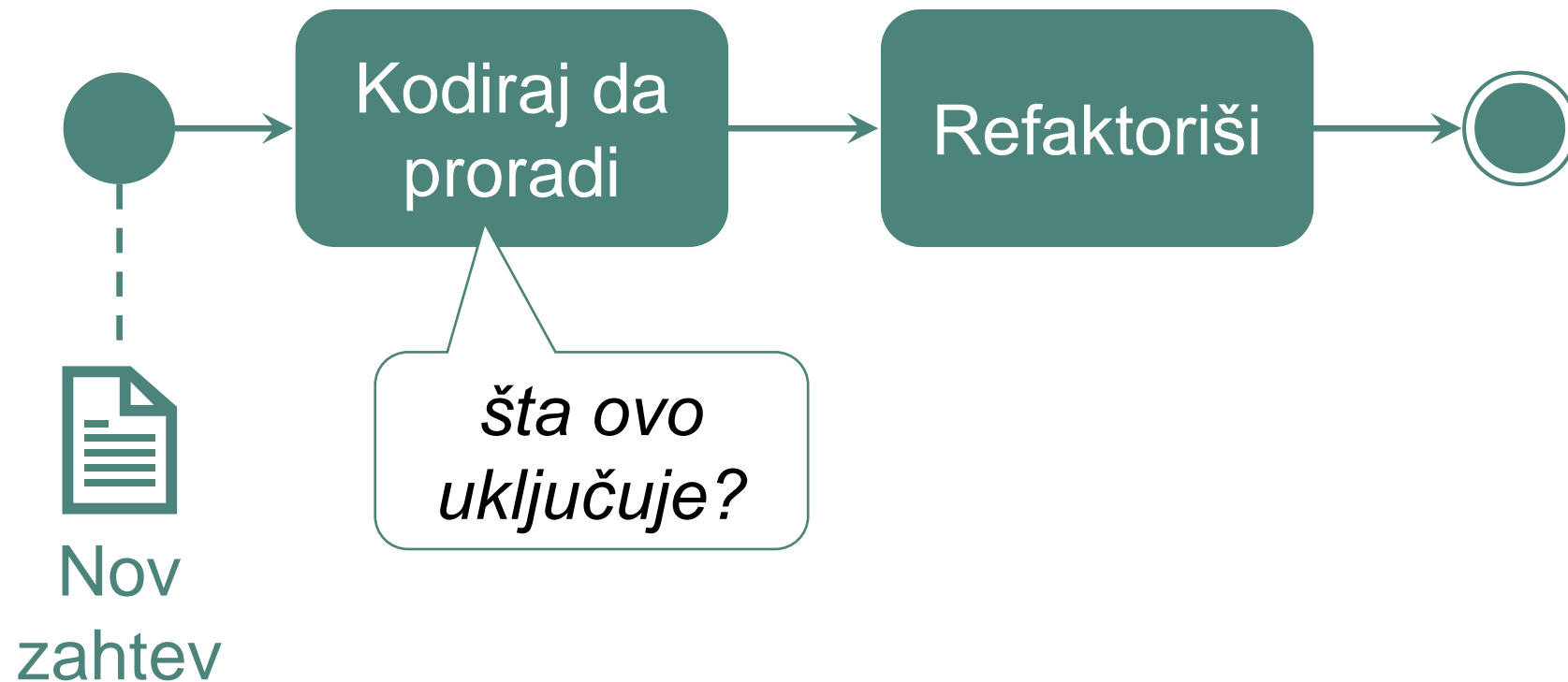
#38 by lukaDoric was merged on Apr 6 • Changes requested

7

Major refactor: Renames Assessment and Instructional Events to Items; moves KCM to Learner/DomainOverlay namespace. ✓

1

Kako da integrišemo čišćenje koda u posao?

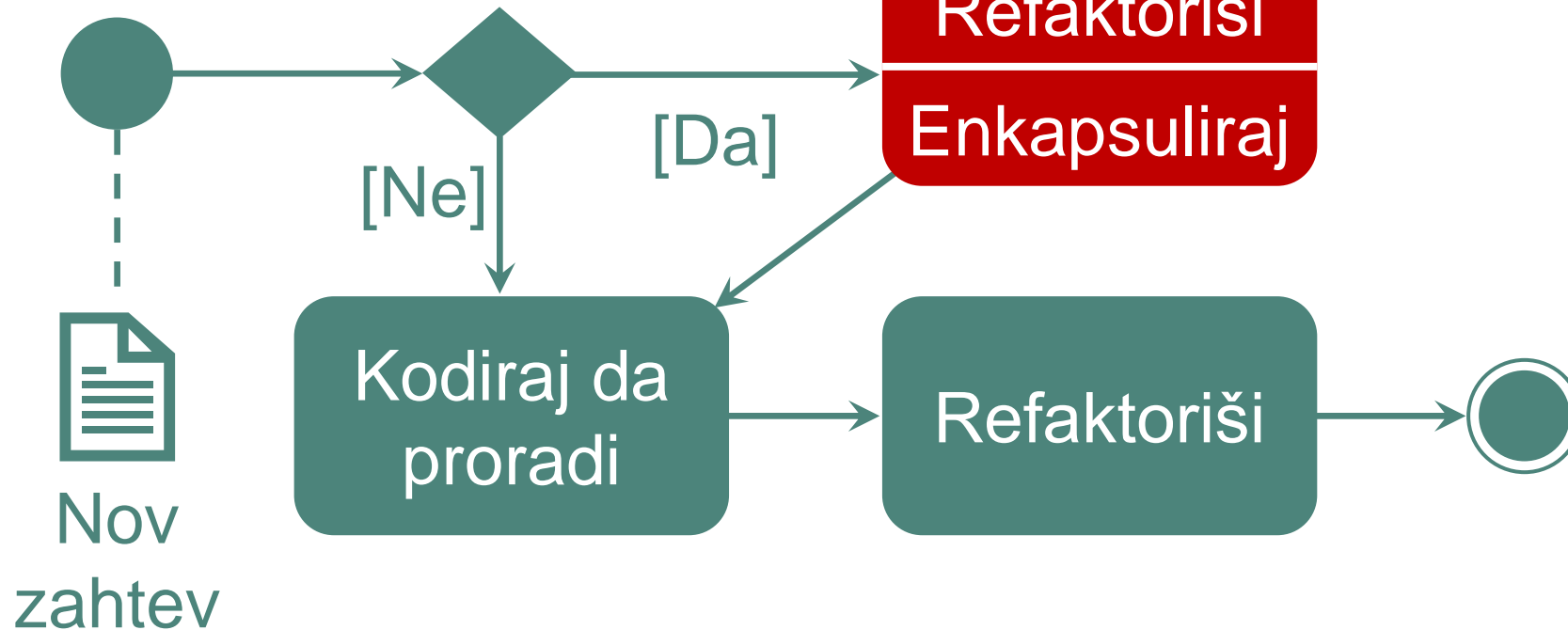


Kako da integrišemo čišćenje koda u posao?

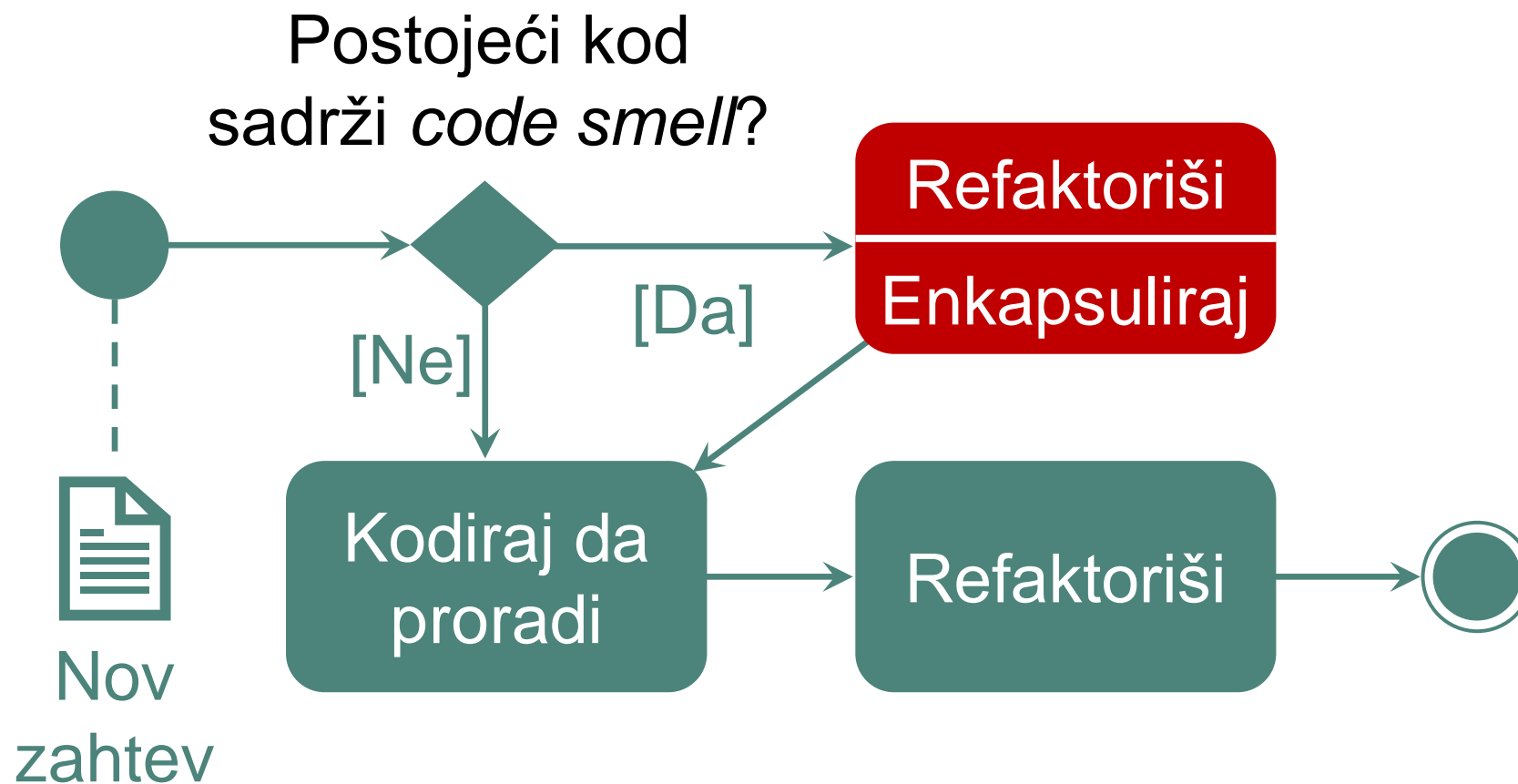
*kako ovo
odrediti?*

Postojeći kod
treba refaktorirati?

*šta ako nema
testova?*



Kako da integrišemo čišćenje koda u posao?



Šta je *code smell*?

Struktura u kodu koja ukazuje
na potrebu za refaktorisanjem

MYSTERIOUS
NAME

COMMENTS

LONG FUNCTION

LONG PARAMETER
LIST

MUTABLE DATA

DIVERGENT
CHANGE

SHOTGUN
SURGERY

FEATURE ENVY

PRIMITIVE
OBSESSION

REPEATED
SWITCHES

DUPLICATED CODE

LAZY ELEMENT

TEMPORARY FIELD

MESSAGE CHAINS

MIDDLE MAN

INSIDER TRADING

CLASSES WITH
DIFFERENT
INTERFACES

DATA CLASS

REFUSED
BEQUEST

LOOPS

GLOBAL DATA

DATA CLUMPS

SPECULATIVE
GENERALITY

LARGE CLASS

Šta ako menadžment ne
podržava refaktorisanje?

Šta ako kolege ne
podržavaju refaktorisanje?

Menjaj firmu ili menjaj firmu