

Eksamiküsimuste vastused aines Tarkvaratehnika (MTAT.03.094)

Aeg: 15. jaanuar 2008, 09:00 – 13:00

Küsimused

1. (4 p.)

Küsimus:

Nimetage väledate (agile) metoodikate põhilised väärtused (neli) ning tehnikad (vähemalt kuus). Andke neile lühike kirjeldus.

Hindamine:

- Iga väärtuse eest 0.25 punkti
- Iga tehnika eest 0.5 punkti

Materjal:

http://courses.cs.ut.ee/2007/tvt/uploads/Main/se_07_17.pdf

Slaidid 40 – 41

2. (4 p.)

Küsimus:

Explain the fundamental difference between functional and non-functional requirements. List at least three types of non-functional requirements and explain briefly the essence of each one.

Vastus & Hindamine:

One point for the first part, one point for each type of non-functional requirements adequately described:

- Functional requirements: what?
- Non-functional requirements: how?
- Non-functional requirements: usability; fault tolerance, reliability, availability, performance; supportability, manageability, configurability, portability; documentedness.

Materjal:

http://courses.cs.ut.ee/2007/tvt/uploads/Main/se_07_05.pdf

Slides 15-16

3. (4 p.)

Küsimus:

Name and briefly describe three principles of service-oriented architectures.

Vastus & Hindamine:

- (4/3 p.) Abstraction: Services share standards and contracts and nothing else.
- (4/3 p.) Autonomy (or loose-coupling): One should minimize dependencies between services and make dependencies explicit. This is motivated in part by the fact that services have owners.
- (4/3 p.) Reusability / composability: Services are intended to be reused / composed.

4. (4 p.)

Küsimus:

In the following are given two lists. One list of descriptions of analysis/modeling tasks and one list of notations. Match each task to the the notation(s) suitable for performing it. Write the answer in the form:

taskno. letter-notation1, letter-notation2, ...

Matter to describe:

1. The Computer Network Structure of the University of Tartu
2. Checking out a book from library
3. Employee list
4. Executive overview of systems analysis report
5. Monthly sales report by product groups
6. Database structure for an online music shop
7. Business Process Flow of checking out at Amazon.com
8. The HTTP Protocol

Notations:

- a) Text
- b) Table(s)
- c) IDEF0
- d) UML use case diagram(s)
- e) UML object diagram(s)
- f) UML class diagram(s)
- g) UML activity diagrams(s)
- h) UML sequence diagram(s)
- i) Fujaba story diagrams

Vastus & Hindamine:

1. (0.5 pt) The Computer Network Structure of the University of Tartu [e]
2. (0.5 pt) Checking out a book from library [c,g,i,(d)]
3. (0.5 pt) Employee list [a,b]
4. (0.5 pt) Executive overview of systems analysis report [a]
5. (0.5 pt) Monthly sales report by product groups [b]
6. (0.5 pt) Database structure for an online music shop [f]
7. (0.5 pt) Business Process Flow of checking out at Amazon.com [c,g,h]
8. (0.5 pt) The HTTP Protocol [h,(g)]

5. (10 p.)

Ülesanne:

Situatsiooni kirjeldus:

Eksisteerib tootmisettevõtte, mille olemasolevate osakondade infosüsteemid ja nendevahelised seosed on kirjeldatud järgmiselt:

Tarnijad: tarnivad toorme lattu.

- Eksisteerib tarnijaid kes saadavad meie süsteemi informatsiooni ning tarnijaid kelle süsteemidest meie lahendus käib infot pärimas.
- Tarnijatel lahendused on realiseeritud erinevatel tehnoloogilistel platvormidel.
- Tarnijate lahendused on kõikuva kvaliteediga

Tootmine: võtab laost toorme ning asetab lattu valmistoodangu

- Tootmishooneid paikneb mitmes linnas
- Tootmine kasutab keeles C++ realiseeritud süsteeme
- Tootmine broneerib laost tooraine ühe nädala jagu ette
- Tootmine saadab valmistoodangu erinevatesse laohoonetesse

Ladu: toorme ja valmistoodangu hoidla

- Laod asuvad erinevates linnades.

- Laosüsteemid on realiseeritud põhinevana Oracle 9i andmebaasil.
- Tehingute mahud suuremates tabelites on miljonites kirjetes aastas.

Müük: müüb laost kauba klientidele

- Müük kasutab Microsoft CRM (customer relationship management) tarkvara.
- Andmehoidlana kasutatakse Microsoft SQL Server 2000.

Olete kirjeldatud ettevõtte laosakonna uus IT juht ning planeerite IT investeeringuid ning tegevusi järgmiseks kolmeks aastaks. Eesmärk on vähendada inimtööjõukulusid viies kogu tarneahela protsessi automaatseks. Tänapäevane osakondade vaheline süsteemiintegratsioon sisuliselt puudub.

Tööülesanne:

Kirjelda millises äriprotsessi etapis milliseid erinevaid liidestamise lahendusi kasutad ning miks:

- Tarnijad – Ladu
- Ladu – Tootmine
- Ladu – Müük

Lahenduse loomisel arvestada liidestuse elementides:

- Kui kasutatakse jagatud andmeid siis kirjeldada miks see on kasulik
- Kui kasutatakse RPC tehnoloogiat siis
 - Kas kutsed on sünkroonsed või asünkroonsed?
 - Millist konkreetset RPC lahendust soovitatakse?
- Kui kasutatakse messaging lahendust siis
 - Kas kasutatakse point-to-point või publish-subscribe mudelit?
- Millistes osades protsessis võiks osutada otstarbekaks batch-processing?
- Kirjeldada üldisel tasemel kuidas kavatsetakse tagada
 - Transaktsioonide tugi
 - Autoriseerimislahenduse tugi
 - Protsessi jälgitavus ja läbipaistvus

Hindamine:

Kokku on küsimuse eest võimalik saada 10pt mis jaguneb järgmiselt:

- (2.5 p.) Lao ja tarnijate integratsioon. Lahendus peab kirjeldama kuidas realiseeritakse push ning pull mudel. Peab olema selge kuidas käsitletakse juhte kui tarnijate süsteemid ei tööta. Pakutav integratsioonilahendus ei tohi olla platvormist sõltuv. Lahendusest peab olema näha kas kasutatakse statistilist lähenemist (tellitakse nii palju kui minevikus tavaliselt kulunud on) või tellitakse ainult sel juhul kui tootmisest tellimus peal on. Peab olema selge kuidas kontrollitakse kas esitatud tellimus on õigeaegselt täidetud. Kuidas käsitatakse situatsioonis kui tootmine nõuab tooret aga tarnijad ei ole seda õigeaegselt tarninud.
- (2.5 p.) Tootmise ja lao integratsioon. Lahendus peab kirjeldama kuidas realiseeritakse tellimuse esitamine ning tellimuse täitmine. Peab olema näha kuidas kontrollitakse tellimuse täidetust (listener, callback, pollimine, vms). Selge peab olema kuidas käsitletakse mitu-mitmele integratsiooni:
 - Tellitavat tooret võib olla ainult mõnes laos.
 - Ladu peab saatma välja toorme kindlale tehasele.
 - Transpordikulude minimeerimiseks tuleks eelistada tootmisele lähimat ladu (ideaalis samas linnas asuvat).

Kuidas käsitletakse juhtu kui laod ei suuda toormetellimust rahuldada? Kuhu ja milliste reeglite järgi saadetakse valmistoodang (müügitellimuste maht)? Lahendus peab kirjeldama kuidas pakutakse müügisüsteemidele piisavalt kiiresti (loe: reaajas) informatsiooni (arvestades kirjete suurt mahtu operatiivbaasis).

- (2.5 p.) Müügi ja lao integratsioon. Lahendus peab näitama kuidas on müük organiseeritud: kas kasutatakse tsentraalset müügikanalit või on müügiesindusi mitu. Lahendus peab näitama kuidas käituda kui soovitatavat toodet hetkel laos ei ole. Lahendus peaks kirjeldama kuidas tagatakse optimaalne kauba liikumine laost kliendini. Lahendus võiks käsitleda kauba tagastamist.
- (0.5 p.) Transaktsioonide lahendus. Lahenduses peab olema näha et tudeng saab aru mis on transaktsioon ning selgitama kuidas tagatakse andmekoosseisu terviklikkus erinevate süsteemide vahel. Ükskõik kas käsitletakse hajustransaktsioone, kompenseerivaid transaktsioone või mõeldakse ise mingi loomingu lahendus välja.
- (0.5 p.) Autoriseerimislahendus. Lahenduses peab olema näha et tudeng saab aru mis on autoriseerimine ning selgitama kuidas tagatakse süsteemide vahel autoriseerimisreeglite propageerimine või näitama kuidas ta kasutab kesket reeglite komplekti. SSO mõiste tundmise ning selgitamise eest antud probleemi kontekstis võib anda lisapunkte.
- (0.5 p.) Monitooringulahendus. Lahenduses peab näitama kuidas ja mis vahenditega tagatakse protsessi läbipaistvus. Ühelt poolt peab olema näha kuidas infot kogutakse ning teisalt kuidas kogutud infot esitatakse. Juhul kui kirjeldatakse alarmide tekitamise mõistet audittraili pealt võib anda lisapunkte.
- (1 p.) Lahenduse üldine arusaadavus, mõistlikkus ning selgus. Hinnata seda kas tudeng on ülesande läbi mõelnud ning saanud ärist aru. Jälgida kas koostatud protsess käsitleb eriolukordi.

6. (11 p.)

Küsimus:

Assume you have HTTP API for creating new users with this specification:

URL: <https://www.example.com/index.jsp>

Parameter	Type	Accepted values	Notes
action	enumeration	newuser	Required field
username	string	Alphanumeric, 4-30 characters	Required field
password	string	Alphanumeric, 1-32 characters	Hashed with SHA1 algorithm
firstname	string	Letters only, 2-30 characters	
lastname	string	Letters only, 4-30 characters	Required field

Email	string	1-50 characters, @ sign, 2-50 characters, at least 1 dot, 2-3 characters	Optional
phone	integer	Numeric, 3-20 numbers	Optional
ssn	integer	Numeric, 0-10 numbers	Required field, Social Security Number
weight	float	Floating point number	Optional

Example call:

<https://www.example.com/index.jsp?action=newuser&username=test1>

How would you test an implementation of this API and what do you find wrong/unspecified in this specification.

Lahendus:

- (1 p.) Try to alter the parameter action to something else, for example changeuser
- (2 p.) Try to break the lower limit of field specifications
- (2 p.) Try to break the upper limit of field specifications
- (1 p.) Try to make the call with wrong type of parameter
- (1 p.) Estonian SSN's are 11-character long, but the API will reject these
- (1 p.) SHA1 hashes are 40-characters long
- (1 p.) What about ÕÜÄÖöüäö? Other special characters?
- (1 p.) What about charset? Is UTF8, ISO-8859-1, etc accepted?
- (1 p.) What if you make the call over HTTP?
- (1 p.) What if you use GET/POST method?
- (1 p.) Are the error messages informative?
- (1 p.) What about authentication?
- (1 p.) Email validation: is dot(s) allowed before @ sign?
- (1 p.) What about longer than 3-character top-level domains (info, aero, mobi, etc)?
- (1 p.) What if you have more than 1 dot after @ sign?
- (1 p.) What about phone number with + signs?
- (1 p.) Why is the email parameter with capital E? Do we accept "e" too? What happens if you capitalize other parameter names?
- (1 p.) Is , or . accepted in weight? Or both?
- (1 p.) How many digits after decimal separator are accepted in weight?
- (1 p.) Can I use SQL injection?

7. (13 p.)

Ülesanne:

Game Manual:

Estonian version:

Nimi: Reis ümber maailma

Vahendid: 1 mängulaud, 16 nappu 4 erinevas värvitoonis, 1 täring

Reeglid: Igal mängijal on neli ühevärvilist nappu; kolm neist asetatakse

nurkades olevatele samavärvilistele platsidele (pessa), üks sinna juurde kuuluvale noolega stardiväljale. Täringuga loositakse mängujärjekord: alustab see, kes saab suurima silmade arvu; edasi käiakse päripäeva. Iga mängija veeretab täringut ja liigutab üht oma nuppu edasi vastavalt saadud silmade arvule. Veeretades 6 silma, saab mängija õiguse uuesti veeretada. 6 silmaga võib mängija ühe nupu pesast stardiväljale tuua ja järgmise veeretamisega seda edasi liigutada. Kui mõni nupp peatub väljal, kus juba on vastase nupp, lööb ta eesoleva nupu oma pessa tagasi. Neljale koduväljale võib astuda ainult täpselt sobiva silmade arvuga. Võitja on see, kes oma neli nuppu esimesena koduväljadele viib.

English version:

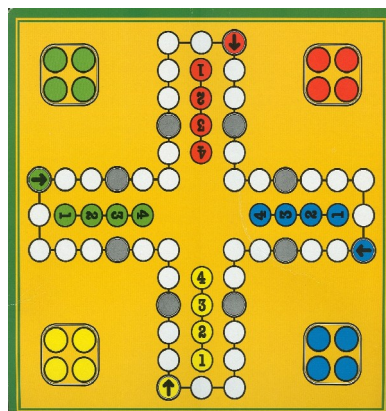
Name: TRIP AROUND THE WORLD (LUDO)

Material: 1 game board, 16 game pieces of 4 different colors, 1 die

Rules: Each player has four pieces of the same color; three of them are placed onto the matching fields in the corner of the board (nest), one to the corresponding starting field (marked with an arrow). The die is rolled to determine the playing order: the player with the highest result starts, after that players take turns in clockwise order. Each player rolls the die and moves one of his/her pieces corresponding number of fields forward. A player who rolls a 6 gets another roll. The player who rolls a 6 can bring one piece from the nest to the starting field and move it on according to the next roll. If a piece finishes a move on a field where another player's piece already stands, the other player's piece is kicked back to the nest. A player has to get the exact number on the die to step onto the four home fields. The player to first move all his/her pieces to the home fields is the winner.

Game board:

The used colors are green, red, blue, and yellow.



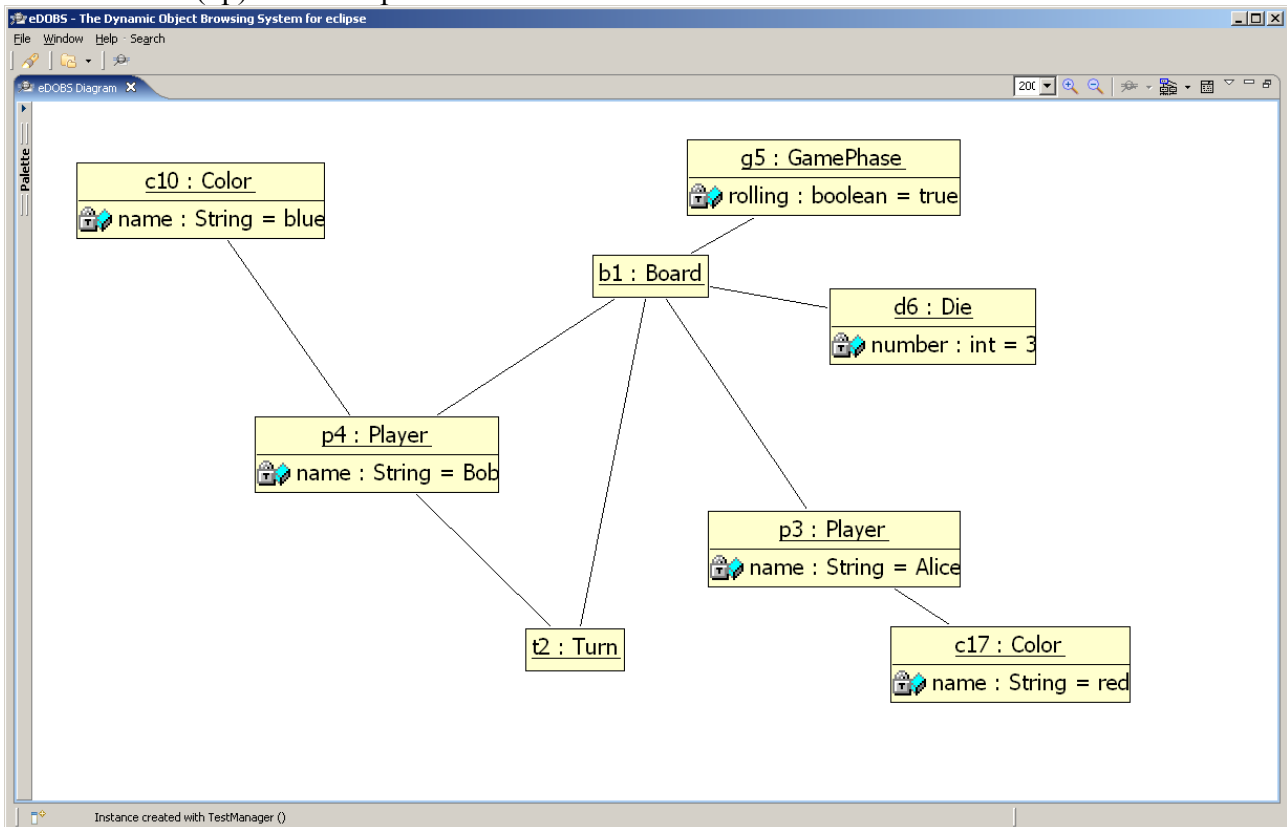
Task:

- Write down 3 textual specific (use names and numbers) usecases/scenarios (imagine the game is played by Alice -red- and Bob -blue-):
 - roll the die,
 - bring figure in game,
 - move figure.
- Draw 6 object diagrams for these usecases (one for each precondition and one for each postcondition). Only draw relevant objects, not all fields on the game board.
- Derive the class diagram (methods not needed) for it (from object diagrams).

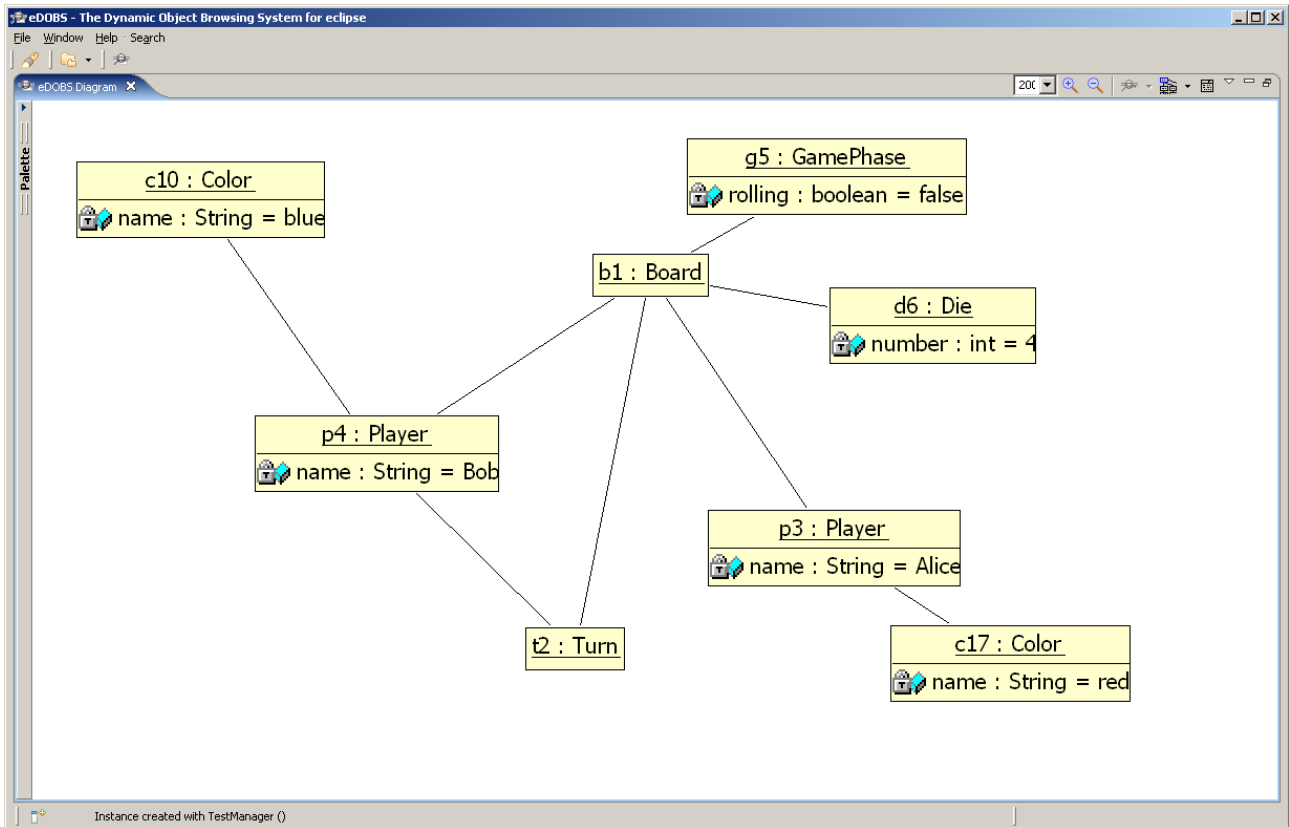
Lahendus & Hindamine:

- Usecases:
 - (1p) Throw dice
 - Precondition: Two players play the game Ludo: Alice and Bob
 - Alice and Bob are sitting at the Ludo game board.
 - Alice has color red.
 - Bob has color blue.
 - It is Bob's turn.
 - The game phase is roll the dice.
 - Action:
 - Bob presses the "roll the dice" button.
 - The die gets the new value 4.
 - The phase of the game changes to moving.
 - Postcondition:
 - The die shows the value 4.
 - The phase of the game is now moving.
 - It is Bob's turn.
 - (1p) Bring figure in the game
 - Precondition:
 - Two players play the game Ludo: Alice and Bob.
 - Alice and Bob are sitting at the Ludo game board.
 - Alice has color red.
 - Bob has color blue.
 - All four figures of Alice are in the red home location on the board
 - All four figures of Bob are in the blue home location on the board
 - It is Alice' turn.
 - Game is in phase rolling dice
 - Action:
 - Alice throws a 6
 - The game phase switches to moving
 - Alice moves the first of her figures in the home location on the red start-field
 - The game phase switches to rolling dice.
 - The turn stays at Alice' (as it was a 6)
 - Postcondition:
 - It is Alice's turn.
 - The game phase is rolling dice.
 - Three of Alice figures are in the red home location.
 - One of Alice figures is on the start field.
 - All four figures of Bob are in the blue home location on the board.
 - (1 p.) Move figure
 - Precondition:
 - Two players play the game Ludo: Alice and Bob.
 - Alice and Bob are sitting at the Ludo game board.
 - Alice has color red
 - Bob has color blue
 - It is Bob's turn.
 - Bob has rolled the dice. he has rolled a 2.
 - Game is in phase moving.
 - Bob has one figure on the game board (one field adjacent to his start filed)

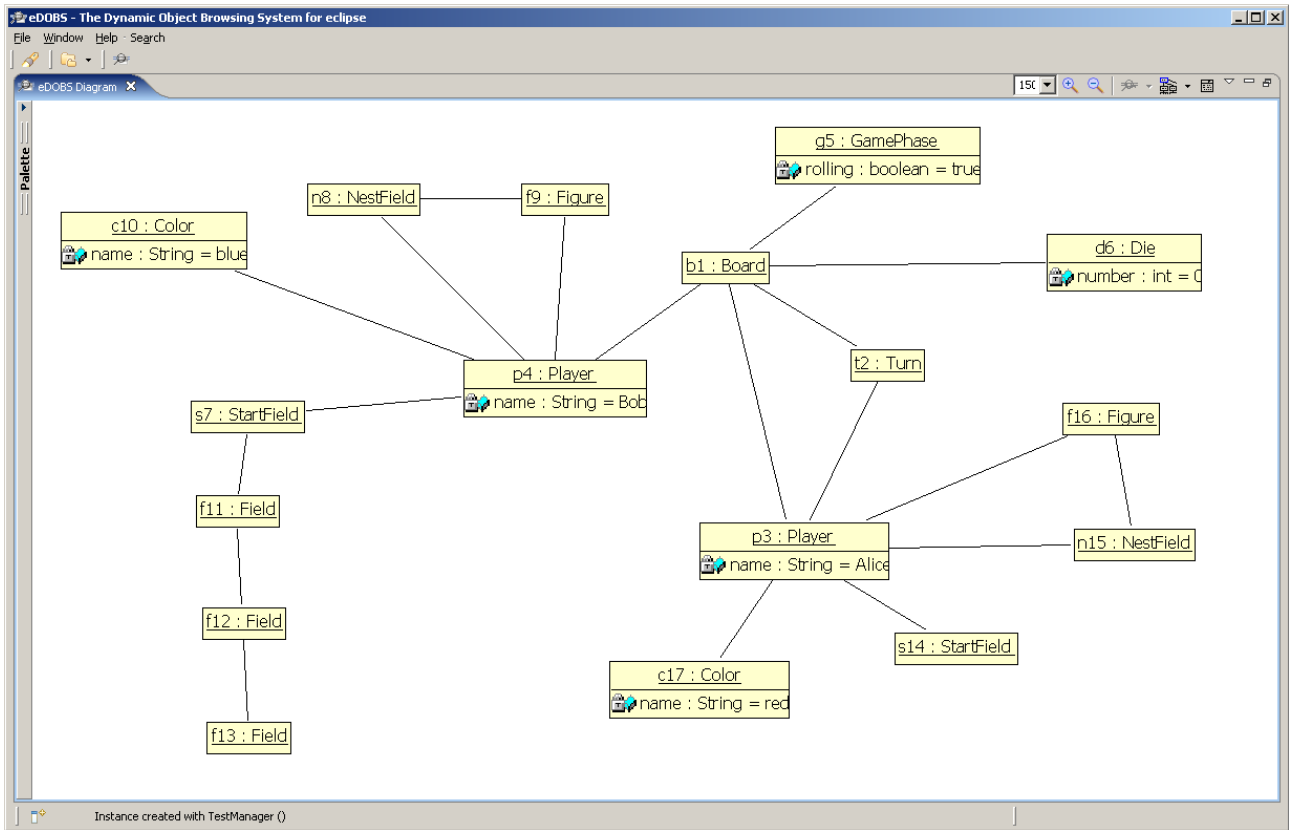
- Action:
 - Bob selects his figure and presses the move button.
 - The figure is moving 2 times one step forward.
 - The phase of the game switches to rolling dice.
 - The turn switches to Alice.
- Postcondition:
 - It is Alice' turn.
 - The game phase is rolling dice.
 - One of Bob's figures is on the game board 3 fields adjacent to his start field.
- Object diagrams
 - (1p) Roll dice pre



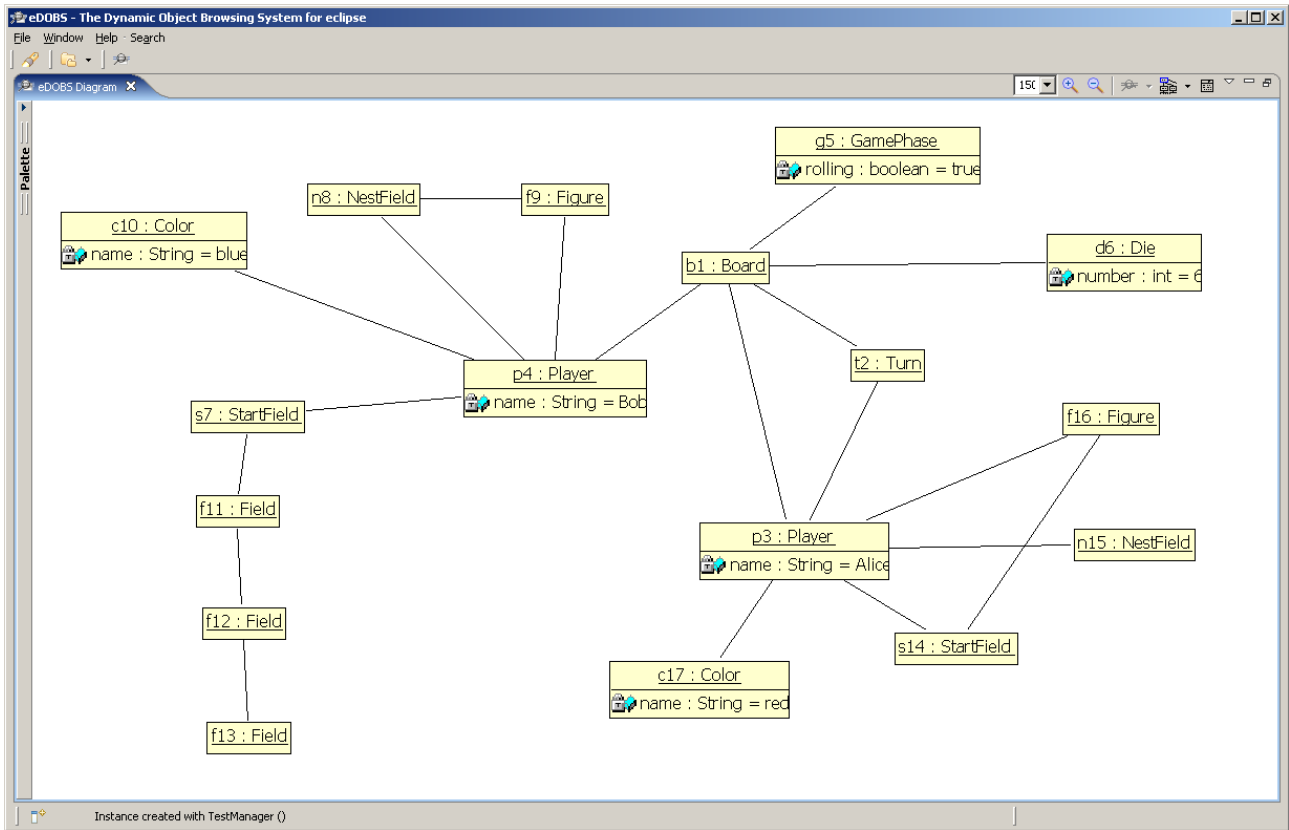
■ (1 p.) Roll dice post



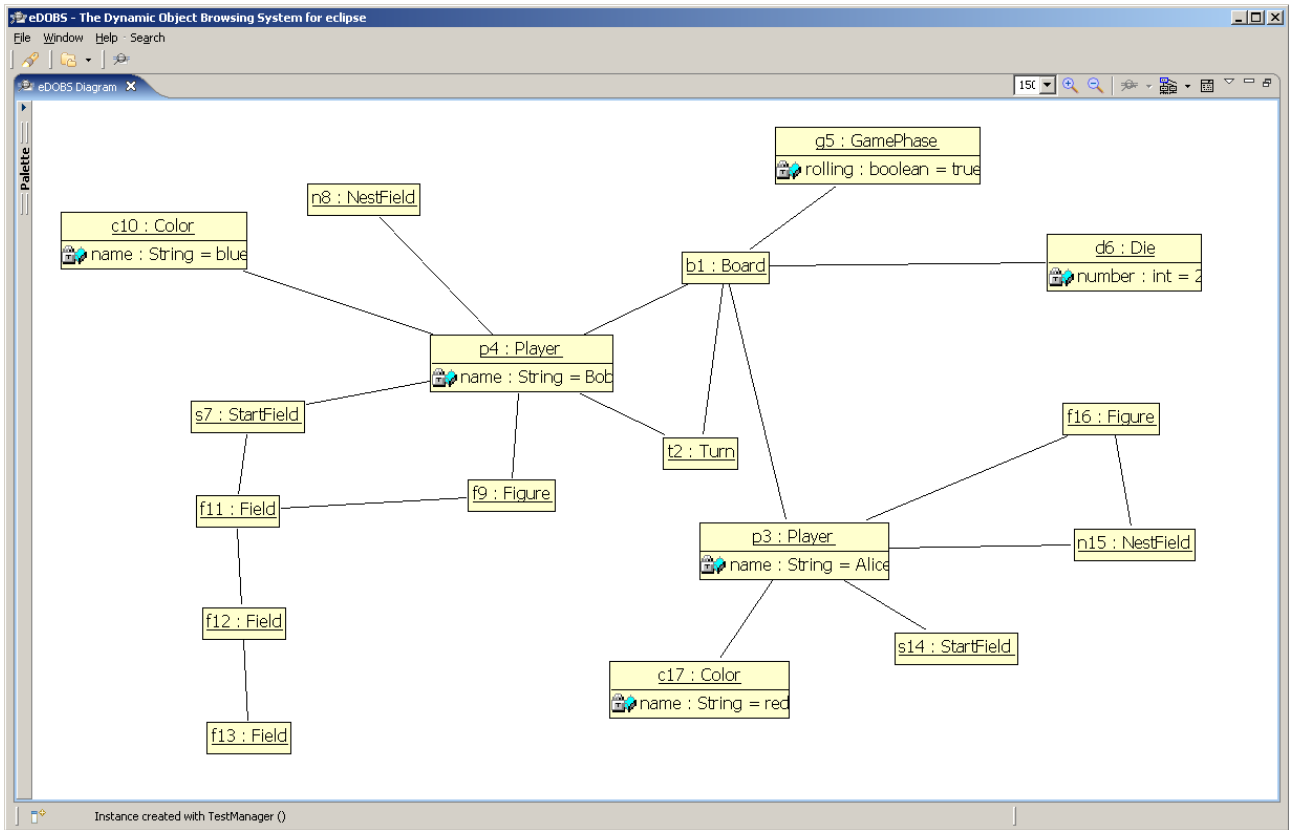
■ (1 p.) Bring in figure pre



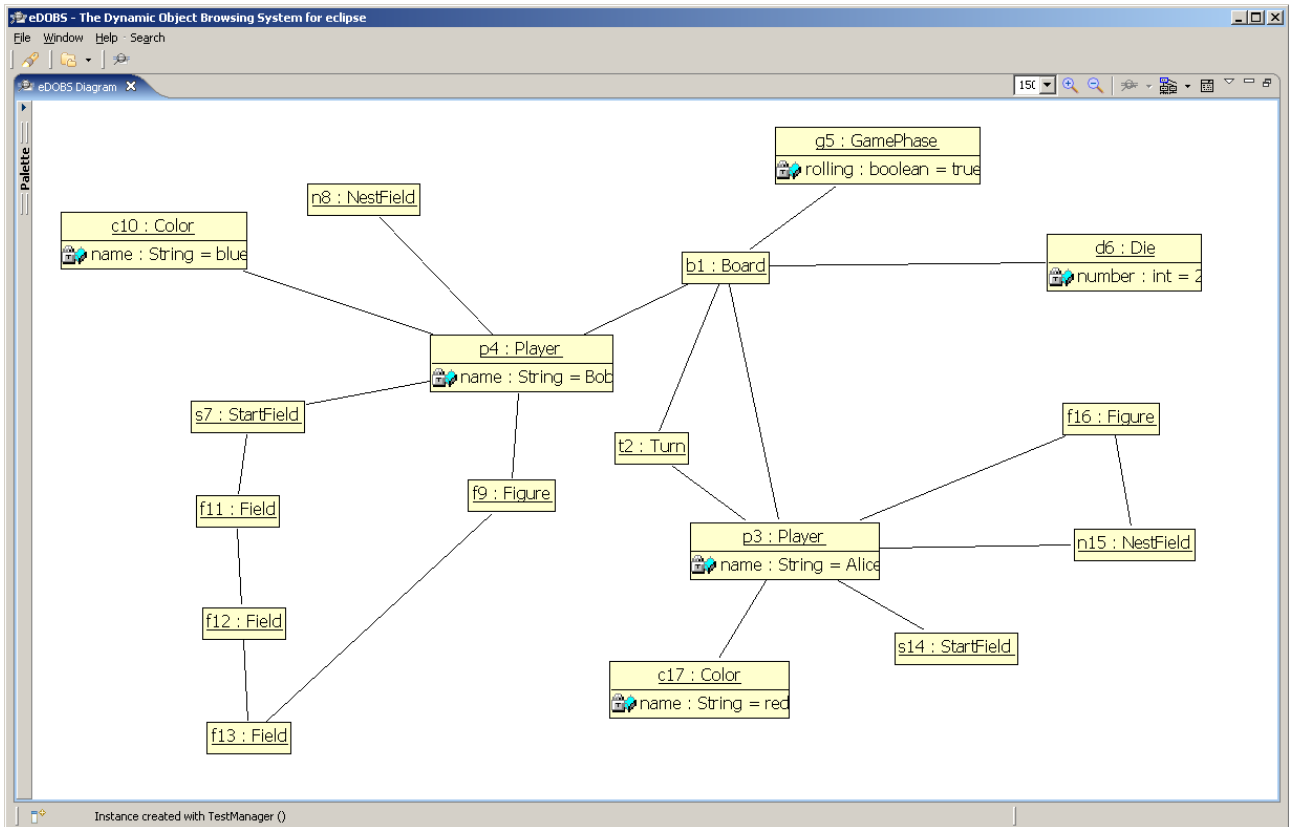
■ (1 p.) Bring in figure post



■ (1 p.) Move figure pre

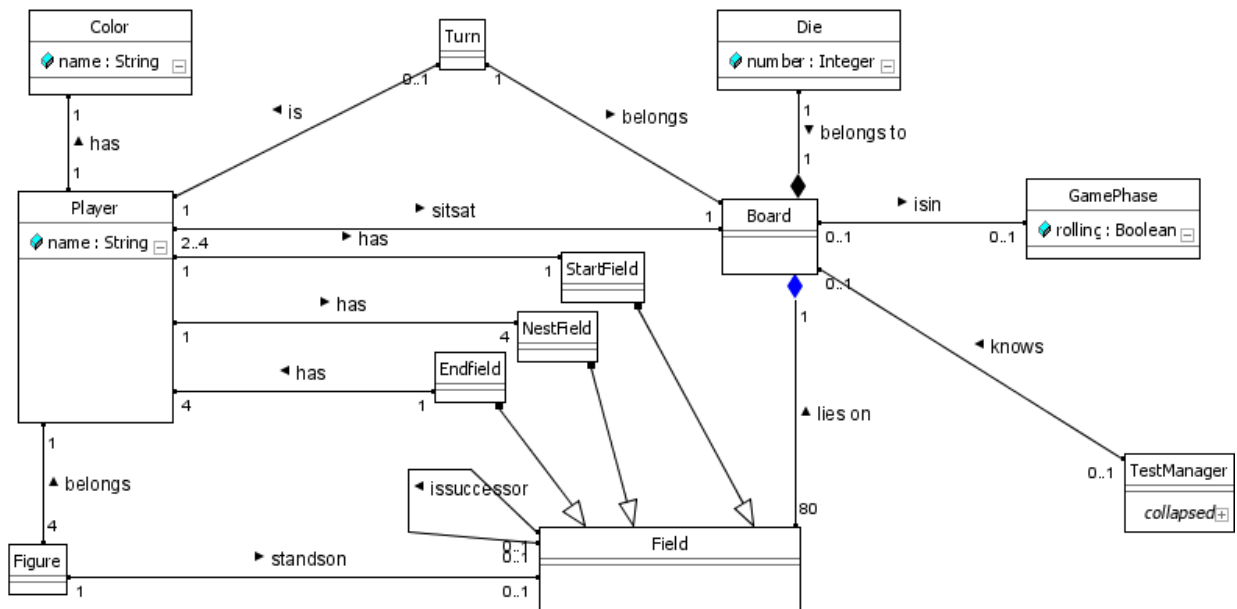


- Move figure post (1p):



- The students only need to draw the relevant objects. The associations must be named similar to the following class diagram. Unnamed associations are usually not allowed (maybe for *is* and *has* links)

- (4 p.) Class diagram



- Accept other (complete and overcomplete) solutions. (Ignore the test-manager class.)

Materjalid:

http://courses.cs.ut.ee/2007/tvt/uploads/Main/se_07_06.pdf

http://courses.cs.ut.ee/2007/tvt/uploads/Main/se_07_21.pdf