NORGES TEKNISK-
NATURVITENSKAPELIGE UNIVERSITET
INSTITUTT FOR DATATEKNIKK OG INFORMASJONSVITENSKAP

Faglig   kontakt   under
eksamen:
Letizia Jaccheri
Tlf: 73593469 / 91897028

EKSAMEN I FAG SIF 8056   PROGRAMVAREARKITEKTUR

OPPGAVE 1 (25%)
Du må lage et programvaresystem som støtter "instant message" kommunikasjon mellom studenter og lærere i et kurs. Kurset varer i et semester og har minst 1000 studenter. Systemet må støtte forkjellige typer enheter som PC'er, PDA'er og mobiltelefoner.
- Skriv ned de funksjonelle krav for systemet (20 ord maks for hvert krav)
1. Register (mandatory)
2. Send 1:1 and 1:n (mandatory)
3. Receive (mandatory)
4. Manage groups (almost mandatory)
5. Login (optional)
6. Logout (optional, better not to have explicit login logout on mobile telephone)
- Skriv en prioritert list av ikke funsjonelle krav
1. Portability
2. Performance (at least 1000 users, delivery time circa 5 sec.)
3. Availability
4. Security
5. Usability
6. Reliability
7. Maintenaibility (not so important since we ask for a course)
8. Safety (NA)
- Lag et "use case" diagram
1. Here it is important that it is consistent with functional reqs
- Lag et "deployment" diagram
1. Typically oracle dbms, and an application server like j2ee and java on client, j2me on pda and telephone. If jabber so they must mention XML. If not jabber they should say something about which protocol.
- Motiver kort valget av hver enkelt COTS (inkludert DBMS)
1. Here it is good if they refer to not functional reqs

Oppgave 2 (30%)
1. Lag et UML klassediagram og et interaksjonsdiagram som beskriver "Observer pattern".
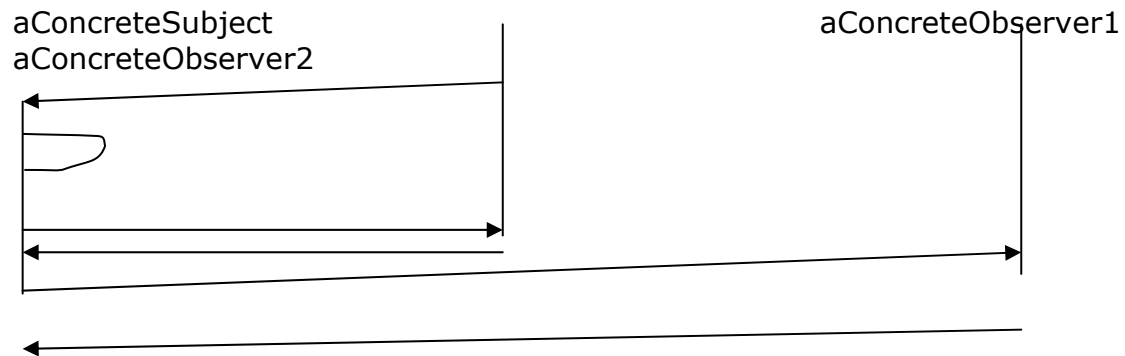Class diagram 4 classes:
- Subject (methods attach, detach, notify)
- Observer(method update)
- ConcreteSubject isA Subject (methods getstate, setstate)

❑ ConcreteObserver isA Observer (method update)
2 associations
observers(Subject, Observer) 1 to many
use(ConcreteObserver, ConcreteSubject) 1 to 1

aConcreteSubject   aConcreteObserver1
aConcreteObserver2

1. setstate()
2. notify()
3. update()
4. gestate ()
5. update()
6. gestate ()

2. Ericsson
2.1
Based on interfaces, domains, abstraction layers or archetypes. The first decomposition was mainly based on interfaces plus a middleware subsystem. Reuse based layering is done later.
2.2
The application framework contains:
Core services: object factory, transaction manager, garbage collector, database wrapper, platform wrapper for start of applications, extended orb, resource manager, performance monitoring
Design patterns and design rules
Libraries for application developers
Tools
Framework and components for fulfilment of quality requirements

3. facade

Oppgave 3 (45%)
For hvert spørsmål, svar "true" eller "false" og motiver svaret med maks 30 ord.

1. If you refer to pattern presentation so it is false.

An idiom expresses low level design decisions. Fundamental architectural choices are expressed by architectural patterns

If you refer to Mary Shaw presentation so it is true, but they should say that it is quite obsolete.

2. Det er meningsløst å evaluere "safety" attributen til eCourse systemet
It does not make sense to evaluate the safety attribute of the eCourse system (true, false)
Because it is not a system the life of human beings can depend.

3. Software design er en prosess som transformerer brukerkrav (funskjonelle og ikke funksjonelle) til en programvare arkitektur og en lavnivå spesifikasjon
Software design is the process of transforming user requirements (functional and not functional) into a software architecture and a detailed specification (true or false)

4. Et design dokument er det samme som et programvarearkitektur dokument
Software design is a document equivalent to a software architecture document (true or false)
it also contains a detail specification or low level design.

5. Arkitektonisk stil "implicit invocation" har ulemper i forhold til vedlikeholdbarhet
The architectural style implicit invocation has disadvantages with respect to maintenaibility (True, False)
It is positive to maintenaibility as it enables run time addition of components.

6. Kvalitetskrav kan kategoriseres i utviklingskrav (som vedlikeholdbarhet og safety) og operasjonelle krav (som ytelse og pålitelighet)
Quality requirements can be categorised into development QRs, e.g., maintenability and safety and operational QRs, e.g., performance and reliability (True or false)
Safety can be observed at run time, so there is a mistake there.

7. "Architectural design" av et programvaresystem er en del av utviklingsprosessen. Den kommer før kravspesifikasjonen.
Architectural design of an individual software system is part of the development cycle. It precedes requirement (functional and not functional) specification and it is followed by detailed design and implementation. (true or false)

8. En programvarearkitektur kan brukes som en "product line" arkitektur.
A software architecture can be used as a product line architecture, i.e. the common architecture for a set of related products or systems developed by the organization (true, false)

9. En arkitektur kan brukes for å beskrive et domene hvor komponenter skal integreres
An architecture can be used to describe a particular domain in which to integrate components (<u>true</u>, false)

10. Et objektorientert rammeverk som Enterprise java Beans er ikke en arkitektonisk komponent
An object oriented framework, such as Enterprise Java Beans is not an architectural component (true, <u>false</u>).
You can build a complex system by using EJB as one component.

11. Evaluering (assesment) av eCourse systemet som ble kjørte i løpet av kurset var en kvalitativ evaluering (true/false).
The assessment session of the eCourse system was a qualitative assessment (<u>true</u>, false)
Because one architecture out of two was supposed to win.

12. Evaluering av det .net baserte media42 systemet som ble kjørte i løpet av kurset var en kvalitativ evaluering (true/false).
The assessment session of the .net media42 system was a qualitative assessment (true, <u>false</u>)
There was only an architecture to be improved, quantitative improving of the quality attributes.

13. Microsoft .net er en spesifikk programvarearkitektur.
is a specific software architecture (True, <u>false</u>)
It is a generic software architecture or framework. A framework is a set of classes that embodies an abstract design for solutions to a family of related problems.

14. Følgende scenario for IBIS systemet har som mål å evaluere kvalitetsattributen ytelse: En million brukere skal spille samme videon samtidig
true, good if they mention that it is also a reliability matter.