



Lunds Universitet
LTH
Ingenjörshögskolan, Helsingborg

Skriftlig tentamen den 21 oktober 2008

Kravhantering, ETS672, 7,5 hp

Kursansvarig: Christin Lindholm

Skrivtid: 8.00-13.00

Inga hjälpmedel är tillåtna

Maximalt antal poäng: 90 poäng
För betyget godkänt krävs 45 poäng

Tentamen innehåller två delar: Del 1 Teori 40 poäng, Del 2 Uppsatsämnen 50 poäng

Del 1 består av kryssfrågor och flervalsfrågor och fylls i direkt i detta frågehäfte.

Del 2 innehåller öppna frågor som besvaras i uppsatsform och lämnas in på separata papper

NAMN: _____

Skriv namn på varje inlämnat papper!

Denna del innehåller frågor som efterfrågar kryss. Frågor som kräver ställningstagande mellan två alternativ. Ställningstagandet anges med **ett** kryss i **en** av rutorna. Ett korrekt satt kryss ger ½ poäng, ett felaktigt satt kryss ger minus ½ poäng. Om inget av alternativen kryssats ges 0 poäng.

Del 1 kan totalt inte ge mindre än 0 poäng.

A1. Vad gäller för dessa påståenden ($\pm 11,5$ p)

- | | |
|--|---|
| Vilken kravnivå man väljer beror i huvudsak på typen av användargränssnitt | <input type="checkbox"/> Stämmer bra |
| | <input type="checkbox"/> Stämmer dåligt |
| Projekttypen påverkar sällan ansvarsfördelningen mellan intressenter | <input type="checkbox"/> Stämmer bra |
| | <input type="checkbox"/> Stämmer dåligt |
| Estimering på ratio-skala innebär parvis jämförelse på graderad skala | <input type="checkbox"/> Stämmer bra |
| | <input type="checkbox"/> Stämmer dåligt |
| Parvisa jämförelser innebär högst en jämförelse per krav | <input type="checkbox"/> Stämmer bra |
| | <input type="checkbox"/> Stämmer dåligt |
| Krav på målnivå gör att leverantörer slipper ta ansvar även för omstrukturering av verksamheten kring produkten. | <input type="checkbox"/> Stämmer bra |
| | <input type="checkbox"/> Stämmer dåligt |
| Spårbarheten underlättas om målnivån är produktorienterad. | <input type="checkbox"/> Stämmer bra |
| | <input type="checkbox"/> Stämmer dåligt |
| För att kunna utföra ett användbarhetstest krävs ett fungerande system | <input type="checkbox"/> Stämmer bra |
| | <input type="checkbox"/> Stämmer dåligt |

- Kategorisering vid prioriteringar är snabbast och ger mest. Stämmer bra
 Stämmer dåligt
- En kravingenjör förväntas hjälpa intressenterna att hitta realistiska och kostnadseffektiva produkter. Stämmer bra
 Stämmer dåligt
- Enligt Moores modell är kunderna i den tidiga majoriteten (early majority) teknik glada och är intresserade av att prova de nya teknikerna. Stämmer bra
 Stämmer dåligt
- Perspektiv baserad granskning innebär att olika perspektiv kombineras t.ex. från användare, designers, testare Stämmer bra
 Stämmer dåligt
- Följande krav:
R1: Tre prototypversioner ska göras och användbarhetstestas under designen av systemet är mer riskfyllt för leverantören än för kunden. Stämmer bra
 Stämmer dåligt
- Ofullständiga datakrav ger större problem i praktiken än ofullständiga kvalitetskrav. Stämmer bra
 Stämmer dåligt
- Elicitering, analys och specifikation är de tre första stegen i kravprocessen Stämmer bra
 Stämmer dåligt
- Marknadsdriven kravhantering innebär många konkurrenter, ofta mindre formella specifikationer och liten närhet till kunderna. Stämmer bra
 Stämmer dåligt

- Card Sorting, Laddering och Scenario-analys är exempel på prioriteringstekniker Stämmer bra
 Stämmer dåligt
- Användningsfall (use case) är måluppfyllande användningssituationer Stämmer bra
 Stämmer dåligt
- Granskning ad hoc innebär att man följer givna riktlinjer Stämmer bra
 Stämmer dåligt
- Först görs specificering, sedan elicitering och sist validering när det gäller kravhantering Stämmer bra
 Stämmer dåligt
- Noggrannhet (accuracy), Autenticitet (authenticity) och integritet (integrity) är kvalitetsegenskaper som ingår i säkerhet (security)? Stämmer bra
 Stämmer dåligt
- Virtuella fönster (virtual windows) ska inte innehålla menyer och knappar Stämmer bra
 Stämmer dåligt
- För uppgiftsbeskrivningar (task descriptions) ska varje uppgift helst vara öppen så att måluppfyllnaden blir flexibel. Stämmer bra
 Stämmer dåligt
- Komplett, inkonsistent och modifierbar är exempel på tre kvalitetskriterier för en bra kravspecifikation. Stämmer bra
 Stämmer dåligt

A2. Vilken eliciteringsteknik passar bäst för att... (±2,5p)

Lösa konflikter	<input type="checkbox"/>	intervjuer	<input type="checkbox"/>	förhandlingar (negotiations)
hitta nuvarande problem?	<input type="checkbox"/>	prototypframställning	<input type="checkbox"/>	Observationer
ta fram förslag till framtida system?	<input type="checkbox"/>	observationer	<input type="checkbox"/>	Fokusgrupper (focus groups)
finna konsekven- ser och risker	<input type="checkbox"/>	pilotexperimet p	<input type="checkbox"/>	observationer
avgöra prioriteter?	<input type="checkbox"/>	prototypframställning	<input type="checkbox"/>	Fokusgrupper

A4. Vilken valideringsteknik passar bäst för att hitta... (±2,5p)

inkonsekvenser mellan olika krav	<input type="checkbox"/>	CRUD (skapa, läsa uppdatera, tabort)	<input type="checkbox"/>	Strukturkontroll (<i>structure check</i>)
spårbarhets- problem?	<input type="checkbox"/>	Pilottest (<i>pilot test</i>)	<input type="checkbox"/>	Strukturkontroll (<i>structure check</i>)
missade krav?	<input type="checkbox"/>	Mål-krav-spårning (<i>goal-requirements tracing</i>)	<input type="checkbox"/>	Strukturkontroll (<i>structure check</i>)
läsbarhets- problem?	<input type="checkbox"/>	Simulering (<i>simulation</i>)	<input type="checkbox"/>	Strukturkontroll (<i>structure check</i>)
orealistiska krav?	<input type="checkbox"/>	Prototyptestning (<i>prototype test</i>)	<input type="checkbox"/>	Strukturkontroll (<i>structure check</i>)

A5. Ange i vilken riktning spårningen går om man spårar... (±2,5p)

- | | | | | |
|---|--------------------------|---|--------------------------|---|
| ...från krav till mål? | <input type="checkbox"/> | spårning bakåt
(<i>backward tracing</i>) | <input type="checkbox"/> | spårning framåt
(<i>forward tracing</i>) |
| ...från krav till programkod? | <input type="checkbox"/> | spårning bakåt
(<i>backward tracing</i>) | <input type="checkbox"/> | spårning framåt
(<i>forward tracing</i>) |
| ...från användargränssnitt till målbeskrivningar? | <input type="checkbox"/> | spårning bakåt
(<i>backward tracing</i>) | <input type="checkbox"/> | spårning framåt
(<i>forward tracing</i>) |
| ...från dataflödesdiagram till testfall? | <input type="checkbox"/> | spårning bakåt
(<i>backward tracing</i>) | <input type="checkbox"/> | spårning framåt
(<i>forward tracing</i>) |
| ...från sekvensdiagram till kontextdiagram? | <input type="checkbox"/> | spårning bakåt
(<i>backward tracing</i>) | <input type="checkbox"/> | spårning framåt
(<i>forward tracing</i>) |

A6. Vad gäller för dessa påståenden angående: (±3p)

Analytical Hierarchy Process (APH)
Bygger på en ratioskala

- Stämmer bra
 Stämmer dåligt

Analytical Hierarchy Process (APH)
Måste gör minst en parvis jämrörelse per krav

- Stämmer bra
 Stämmer dåligt

Analytical Hierarchy Process (APH)
Parvisa jämförelser innebär maximal
redundant $n*(n-1)/2$

- Stämmer bra
 Stämmer dåligt

Hårvarukrav
Funktionskraven ska specificera vilka
komponenter som ska användas

- Stämmer bra
 Stämmer dåligt

Funktionella krav
Anger vad systemet ska göra

- Stämmer bra
 Stämmer dåligt

Funktionella krav
Dataordlista (data dictionary) är en
datakravstil

- Stämmer bra
 Stämmer dåligt

A7. Använd alternativen A-I för att ange en typisk fördel respektive en typisk nackdel med var och en av följande tekniker (7p)

I frågorna efterfrågas en bokstav. Alla kvadrater ska fyllas i men en exakt bokstav. Vissa bokstäver kan förekomma flera gånger och det är inte säkert att alla bokstäver behövs. Rätt ifylld ruta ger ½ poäng medan felaktigt ifylld yta ger 0 poäng.

A: Lätt för utvecklare att använda vid verifiering

B: Svårt för kunder att validera att arbetsuppgifter stöds på ett bra sätt

C: Passar dåligt vid inköp av generisk hyllprogramvara (COTS)

D: Ger inte information om krav på data

E: Passar bra att implementeras direkt och de flesta utvecklare känner till dem

F: Om de är stora blir de svåra att läsa och förstå

G: Kunden kan lätt validera vilken data som ska lagras

H: Oerfarna utvecklare lägger gärna onödig tid på design av användargränssnittet

I: Lätt att drömma ihop så många att systemet blir orealistiskt

Händelselista på produktnivå (<i>product event list</i>)	Fördel	<input type="checkbox"/>	Nackdel	<input type="checkbox"/>
Produktegenskapskrav (<i>feature requirements</i>)	Fördel	<input type="checkbox"/>	Nackdel	<input type="checkbox"/>
Skärmbilder och prototyper (<i>screens and prototypes</i>)	Fördel	<input type="checkbox"/>	Nackdel	<input type="checkbox"/>
Uppgiftsbeskrivningar (<i>task descriptions</i>)	Fördel	<input type="checkbox"/>	Nackdel	<input type="checkbox"/>
Tillståndsdigram (<i>state diagrams</i>)	Fördel	<input type="checkbox"/>	Nackdel	<input type="checkbox"/>
Datamodeller med E/R-diagram (<i>data model with E/R-diagrams</i>)	Fördel	<input type="checkbox"/>	Nackdel	<input type="checkbox"/>
Virtuella fönster (<i>virtual windows</i>)	Fördel	<input type="checkbox"/>	Nackdel	<input type="checkbox"/>

A8. Påstående/anledning-frågor. (7p)

För varje par av påstående/anledning svara med ett av följande alternativ:

- A: Både påståendet och anledningen är korrekta uttalanden OCH anledningen förklarar påståendet på ett korrekt sätt.
- B: Både påståendet och anledningen är korrekta uttalanden, men anledningen förklarar inte påståendet.
- C: Påståendet är korrekt, men anledningen är ett felaktigt uttalande.
- D: Påståendet är felaktigt, men anledningen är ett korrekt uttalande.
- E: Både påståendet och anledningen är felaktiga uttalanden.

Påstående: Krav på domännivå innehåller oftast vad varje enskild aktör kräver var för sig.

Anledning: Designnivåkrav uppkommer ofta då befintliga system ingår i domänen.

A|B|C|D|E

Påstående: Vid interna projekt (*in-house projects*) används sällan formella kravspecifikationer.

Anledning: Formella kravspecifikationer gör det svårt att verifiera att marknaden är nöjd med produkten.

A|B|C|D|E

Påstående: Om leverantören inte har domänkunskap och ej kan ta ansvar för omstrukturering av verksamheten är det lämpligt att kraven är på målnivå.

Anledning: Formella kravspecifikationer gör det svårt för lekmän att validera kraven.

A|B|C|D|E

Påstående: För hyllprogramvara (COTS) är det mindre lämpligt att ställa krav på designnivå.

Anledning: Kravhanteringen för hyllprogramvara handlar till stor del om att välja mellan befintliga produkter med redan existerande användargränssnitt.

A|B|C|D|E

Påstående: Eliciteringstekniken ”design workshops” riskerar att helt förbise övergripande affärs mål.

Anledning: Deltagarna blir lätt uppslukade av tekniska detaljer.

A|B|C|D|E



Påstående: Vid införande av informationssystem i stora organisationer är det lämpligt att genomföra pilotexperiment.

Anledning: Om en stor verksamhet berörs av informationssystemet är införandekostnaden ofta högre än systemutvecklingskostnaden.

A|B|C|D|E



Påstående: CRUD-matrisen är bra att använda för att finna krav som saknas i kravspecifikationen.

Anledning: När införandekostnaden är högre än utvecklingskostnaden är det lämpligt att genomföra pilotexperiment.

A|B|C|D|E



Påstående: I uppgiftsbeskrivningar (*task descriptions*) är den exakta ordningsföljden hos deluppgifterna inte nödvändigtvis den enda rätta.

Anledning: För domännivåkrav är uppdelningen mellan vem som gör vad av aktörer och system inte avgjord.

A|B|C|D|E



Påstående: Dataflödesdiagram är bra för att beskriva användaraktiviteter.

Anledning: Dataflödesdiagram beskriver krav på målnivå

A|B|C|D|E



Påstående: Kravhanteringen bör inte sluta förrän kraven är fullständiga.

Anledning: En fullständig kravspecifikation underlättar kravvalidering

A|B|C|D|E



Påstående: Varje uppgiftsbeskrivning (*task description*) bör helst vara öppen så att måluppfyllnaden inte är avgjord.

Anledning: Det är bättre att dela upp relaterade deluppgifter.

A|B|C|D|E



Påstående: Prioritering med betygsättning kan försvåra ändringshanteringen.

Anledning: Vid betygsättning får många krav samma prioritet vilket inte ger en total rangordning och det är därmed inte lätt att avgöra vilka krav som bör strykas först.

A|B|C|D|E



Påstående: Om användbarhetsproblem upptäcks vid systemtest är de ofta svåra att hantera.

Anledning: Användbarhetsproblem kräver ofta små förändringar av begränsade delar av användargränssnittet.

A|B|C|D|E



Påstående: Ansvar för kravuppfyllelse bör i minsta möjliga mån kopplas till systemkomponenter

Anledning: Kravuppfyllelse kopplat till systemkomponenter innebär att ansvarighet och påverkansanalys kan utrönas.

A|B|C|D|E



A9. Välj för varje beskrivning den projekttyp som passar bäst. (4p)

Projekttyp

A = Produktutveckling (product development)

B = Internutveckling (in-house development)

C = Kontraktutveckling (contract development)

D = Offertförfrågan (tender)

E = Utveckling på löpande räkning (time and materials)

F = Utveckling utlagd på underleverantör (sub-contracting)

G = Inköp av hyllprogramvara (COTS purchase)

- | | | |
|---------------|--------------------------|---|
| A B C D E F G | <input type="checkbox"/> | Kunden betalar utvecklingskostnaden till leverantören, ofta månadsvis. Kostnaden varierar och slutsumman är ofta inte i förväg känd. |
| A B C D E F G | <input type="checkbox"/> | Kunden behöver ett system för en viss typisk uppgift och köper det som passar bäst av de system som finns på marknaden. |
| A B C D E F G | <input type="checkbox"/> | Flera leverantörer får chans att visa vad de kan leverera i ett anbudsförfarande. Denna projekttyp ingår ofta som en lagstadgad del i en offentlig upphandling. |
| A B C D E F G | <input type="checkbox"/> | Utveckling för en öppen marknad där marknadsavdelningen har kundkontakt och samtidigt ofta agerar intern kund åt utvecklingsavdelningen. |
| A B C D E F G | <input type="checkbox"/> | En avgränsad del av utvecklingen lämnas vidare till en tredje organisation. En integratör ansvarar sedan för helheten och leveransen till kunden. |
| A B C D E F G | <input type="checkbox"/> | En uppdragsgivare och en leverantör reglerar genom styrdokument vad som ska levereras, ofta innefattande kundspecifik utveckling med överenskommen prismodell. |
| A B C D E F G | <input type="checkbox"/> | Utvecklingen sker för eget bruk och baseras inte på kontrakt mellan skilda juridiska parter. Ofta genomförs utvecklingen utan kravspecifikation, inte sällan med ödesdigra följder. |
| A B C D E F G | <input type="checkbox"/> | Denna typ av utveckling kännetecknas av många kunder och konkurrenter, evolution genom releaser, och fokus på lönsamhet och marknadsandelar. |

Del 2 Uppsatser 50 p

Skriv korta uppsatser utifrån följande rubriker. Max 2 A4-sidor per uppsats. Skriv läsligt. Svårlästa uppsatser ger poängavdrag. Börja på nytt blad för varje ny uppsats.

2.1 Icke-funktionella krav: exempel, kategorier, tekniker och teknikval, utmaningar (15p)

2.2 Validering: kravkvalitet och metoder (10p)

2.3 En bra kravspecifikation: innehåll, egenskaper, situationsanpassning och teknikval (15p)

2.4 Användbarhet: hur uppnås användbarhet, användbarhetsfaktorer, tekniker och metoder (10p)