

Lunds Tekniska Högskola, Inst. för Datavetenskap

Skriftlig tentamen i ETS170 Kravhantering

Tid: 2010-12-16 kl. 8-13, **Plats:** Eden 25, 26

Hjälpmedel: Inga.

OBS! Tentamen innehåller två delar: Del A Teori 50 poäng, Del B Uppsatsämnen 50 poäng. Del A består av flervalsfrågor och kommer att bedömas schablonmässigt med mallar (ev. automatiskt) och fylls i direkt i detta häfte. Del B innehåller öppna frågor som besvaras i uppsatsform och lämnas in på separata papper.

NAMN: _____

Skriv namn på varje inlämnat papper!

DEL A. TEORI 50p

Denna del innehåller frågor i form av påståenden och anledningar.

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.

Alternativen anges med bokstäver A-E i anvisad kvadrat:

Svar A|B|C|D|E|_

Alla kvadrater ska fyllas i med exakt en bokstav. Rätt ifylld ruta ger 1 poäng medan felaktigt ifylld eller oifylld ruta ger 0 poäng.

Denna del innehåller även *para-ihop* -frågor där olika svarsalternativ (A-E) ska paras ihop med rätt alternativ i en lista. Svaren anges i anvisad kvadrat. Rätt ifylld ruta ger 1 poäng medan felaktigt ifylld eller oifylld ruta ger 0 poäng.

	<i>Påstående</i>	<i>Anledning</i>	<i>Svar</i> A B C D E _
1	Kvalitetskrav specificerar ofta kriterier som kan användas för att bedöma hur väl en viss funktionalitet i systemet fungerar.	Kvalitetskrav står i kontrast till de icke-funktionella kraven, då de senare ofta definierar ett icke fungerande systembeteende.	☐
2	Lättrörlig (<i>eng . Agile</i>) kravhantering är lämplig vid föränderliga kravmängder.	Hantering av kravändringar under utvecklingens gång möjliggör flexibel anpassning till förändrade kundbehov.	☐
3	Att klustra krav med många beroenden kan ofta underlätta releasplaneringen.	Krav som är beroende av varandra är sällan lämpliga att implementera i samma release.	☐
4	Krav på design-nivå ökar möjligheten att finna alternativa lösningar när det är dags för implementation.	Design-krav beskriver ofta specifika implementationslösningar.	☐
5	Vid prioritering med parvisa jämförelser kan man beräkna ett index som karaktäriserar mängden inkonsekvenser.	Om ett krav finns med i endast en jämförelse, kan denna redundans användas för att upptäcka cirkulära egenvärden.	☐
6	Det är sällan lämpligt att specificera maximala responstider i ett fleranvändarsystem.	Maxtiden är ofta väldigt dyr att garantera; det är lämpligare att t.ex. kravställa sannolikheten att en viss acceptabel responstid uppnås.	☐
7	Vid elicitering inom okända kravområden är strukturerade intervjuer ofta lämpliga.	Strukturen hjälper till att identifiera kravområden som ej legat till grund för intervjufrågorna.	☐
8	Om man förlitar sig enbart till designworkshops vid elicitering är det stor risk att affärs mål och uppgifter glöms bort.	Användarnas uppmärksamhet upptas vid designworkshops av tekniska frågeställningar med fokus lösningsförslagen.	☐

	Påstående	Anledning	Svar A B C D E _
9	I ett virtuellt fönster (<i>virtual window</i>) visas dataflödet (<i>data flow</i>) mellan användargränsnittet och databasen.	Datakrav berör relationen mellan input-output och den bearbetning som sker däremellan.	☐
10	Ett kontextdiagram är lämpligt om man vill beskriva vilka gränssnitt som finns mellan systemet och dess omgivning.	Kontextdiagram visar en överskådlig bild av direkta aktörer och deras kommunikation med systemet.	☐
11	Uppgiftsbeskrivningar (<i>task descriptions</i>) är ofta lättare för användare att validera jämfört med klassdiagram.	Uppgiftsbeskrivningar (<i>task descriptions</i>) definierar den specifika tidsordning som underuppgifter (<i>sub tasks</i>) sker i.	☐
12	I marknadsdriven kravhantering händer det att ett krav går till tillståndet "avslag" (<i>reject</i>) direkt efter initialtillståndet.	Det kan hända att ett krav inte är i linje med den övergripande strategin för mjukvaruprodukten.	☐
13	Spårbarhet (<i>traceability</i>) underlättar ofta underhåll (<i>maintenance</i>).	Kravändringars påverkan på kod kan följas om det finns spårbarhet från krav till implementation, vilket gör det lättare att hitta var uppdateringar krävs.	☐
14	Kontextdiagram (<i>context diagram</i>) är inte lämpligt vid utveckling med löpande räkning (<i>time-and-material</i>).	Vid utveckling med löpande räkning (<i>time-and-material</i>) betalar kunden kostnaderna allt eftersom.	☐
15	Fokusgrupper är mer strukturerade än idékläckningsmöten (<i>brainstorming</i>).	När man genomför brainstorming uppmanas intressenterna att närsomhelst kritisera idéerna.	☐
16	Öppen målsättning (<i>open target</i>) låter användaren bestämma det exakta värdet.	Öppet mått (<i>open metric</i>) finns ofta i en dataordlista (<i>data dictionary</i>)	☐

	Påstående	Anledning	Svar A B C D E _
17	CRUD-kontrollen kan med fördel användas för att leta efter saknade eller inkonsekventa krav.	Datakrav är mer lika kvalitetskrav än funktionella krav, varför inkonsekvenser ofta blir mindre riskfyllda.	☐
18	Enligt QUPER uppstår mättnadsbarriär (saturation barrier) när kostnaderna planar ut.	Innan en barriär kan kvaliteten förbättras till en förhållandevis låg kostnad, men vid en barriär krävs stora investeringar för att nå högre kvalitet.	☐
19	Det är svårt att få ett meningsfullt resultat från ett användbarhetstest om det utförs innan produkten levererats.	Heuristisk evaulering är ofta dyrare än användbarhetstestning, men har mindre risk att identifiera skenbara användbarhetsproblem.	☐
20	Produktledning för mjukvaruprodukter (<i>software product management</i>) kräver många skilda kompetenser, t.ex. kravprioritering och releasplanning.	Föreberedelse för lansering (<i>launch preparation</i>) är en kompetens som är extra relevant för de mjukvaruorganisationer som saknar interna intressenter.	☐
21	Krav på målnivå gör att leverantörer kan ta ett mindre ansvar för verksamheten kring produkten.	Vilken kravnivå man väljer beror i stor utsträckning på vem som ska utföra uppdraget	☐
22	Det är ofta lämpligt att ta med både domäninformation och illustrativa exempel i anslutning till ett krav.	Information om bakgrund till kravet samt hypotetiska lösningsförslag är sällan till hjälp när krav ska implementeras.	☐
23	Nuvärdesanalys (<i>net present value analysis</i>) tar inte hänsyn till osäkerheten i framtida vinster.	Framtida vinster är mer osäkra desto längre bort i tiden de inträffar.	☐

	Påstående	Anledning	Svar A B C D E _
24	Öppna mått (<i>open metric</i>) passar bra när man vill överlåta åt leverantören att definiera hur kvalitetskrav ska mätas.	Vissa kvalitetskrav är svåra att kvantifiera och leverantören kanske har specifik domänkunskap kring vanliga sätt att göra kvantifieringar i domänen.	☐
25	Kvalitetskrav är i praktiken ofta svårare att hantera än datakrav.	Kvalitetskrav rör oftast bara enskilda delar av systemet, medan datakrav oftare påverkar hela systemet.	☐
26	I marknadsdriven kravhantering kan man med fördel använda en databas där kravattributen specificeras successivt.	I en kravdatabas kan man lättare skilja på alfa- och betakrav.	☐
27	100\$-metoden ger prioriteter som är på en ordinal skala där endast ordningen och inte inbördes storleksrelationer kan avgöras.	Med 100\$-metoden kan man beräkna ett konsekvensindex, då varje jämförelse är med flera gånger.	☐
28	Användbarhetsproblem är ofta mer oväntade för utvecklarna än programmeringsproblem.	När man utvärderar ett systems användbarhet försöker man undvika att ta hänsyn till användarnas subjektiva upplevelse.	☐
29	En datamodell (E/R-diagram) är bättre på att ange kardinalitet (<i>cardinality</i>) jämfört med virtuella fönster.	En fallgrop med virtuella fönster är att överarbeta den grafiska utformningen, vilket kan öka risken att de tolkas som krav på designnivå.	☐
30	Att ha en speciellt ansvarig för integrationen av en produkt som består av delprodukter från flera leverantörer leder ofta till mer problem än om kunden ansvarar.	Kunden kan ofta själv ta ansvar för integrationen, speciellt om kunden kan verksamheten bättre än programvarutekniken.	☐
31	Att göra parvisa jämförelser mellan krav tar kortare tid än att prioritera varje krav var för sig.	Om kraven prioriteras var för sig får man inte någon information om deras inbördes relation.	☐

	Påstående	Anledning	Svar A B C D E _
32	Kapacitetskrav (<i>capacity requirements</i>) är ofta enkla att kvantifiera.	Kapacitetskrav (<i>capacity requirements</i>) är processkrav som t.ex. kvantifierar utvecklingskapaciteten (<i>development capacity</i>).	☐
33	Användbarhetskrav på designnivå utgör en större risk för kunden än för leverantören.	Leverantören kan bedöma och uppfylla ett designkrav oberoende av om det stödjer kundens verksamhet och användarens uppgifter.	☐
34	Varje uppgiftsbeskrivning (<i>task description</i>) bör helst bara ha en aktör.	Det är bättre att dela upp relaterade deluppgifter i olika uppgiftsbeskrivningar, en för varje aktör.	☐
35	Kravhanteringen bör inte sluta förrän kraven är fullständiga.	En fullständig kravspecifikation är lättare att överblicka.	☐
36	I en korrekt kravspec återspeglar alla krav existerande behov eller förväntningar.	Kravens korrekthet är ofta beroende av deras modifierbarhet.	☐
37	Risken för inkonsekventa krav ökar om det finns redundans i kravspecifikationen.	Redundans kan åtgärdas genom att istället göra korsreferenser.	☐
38	Unika nummer för varje krav stödjer modifierbarheten.	Unika nummer underlättar index och korsreferenser, som i sin tur gör det lättare att hitta och hålla reda på krav som ändras.	☐
39	Krav på utvecklingsprocessen syftar till att ersätta vaga krav med krav på lämpliga arbetsmetoder.	Iterativ utveckling delar upp kravhanteringen i flera steg.	☐

Påstående

Anledning

Svar

A|B|C|D|E|_

40 Om användbarhetsproblem upptäcks vid systemtest är de ofta lättare att hantera än om de upptäcks vid elicitering.

Användbarhetsproblem kan kräva stora förändringar av användargränssnittet.

☐

41 Ett fokusgruppsmöte används ofta i stället för semi-strukturerad validering (*semi-structured validation*).

Arbetet med fokusgrupper liknar brainstorming men är mer strukturerad och sker i flera steg, bland annat ingår framtidsvisioner och prioritering.

☐

Para ihop:

Nuvarande arbete (present work)	A
Framtida systemidéer (future system ideas)	B
Realistiska möjligheter (realistic opportunities)	C
Fullständighet (completeness)	D
Åtagande (commitment)	E

Para ihop *mest lämpliga* saken att elicitera (A-E) ovan med resp. eliciteringsteknik nedan: (samma bokstav kan ev. förekomma flera gånger; det är inte säkert att alla bokstäver ovan passar att para ihop med nedan alternativ.)

42	Observation	☐
43	Förhandling (negotiation)	☐
44	Mål-domän-analys (goal-domain analysis)	☐
45	Prototyper	☐
46	Brainstorming	☐

Para ihop:

<i>Lättrörlig kravpraxis</i>		<i>Agile RE practices</i>	
Hantering av kravändringar genom konstant planering	A	Managing requirements change through constant planning	A
Propotyputveckling	B	Prototyping	B
Test-driven utveckling	C	Test-driven development	C
Iterativ kravhantering	D	Iterative requirements engineering	D
Granskningsmöten och acceptanstest	E	Review meetings and acceptance tests	E

Para ihop *mest lämpliga* lättrörliga kravpraxis (A-E) ovan med utmaning (challenge) nedan: (samma bokstav kan ev. förekomma flera gånger; det är inte säkert att alla bokstäver ovan passar att para ihop med nedan alternativ.)

- | | | | |
|----|--|---|--------------------------|
| 47 | De har varit ovilliga att acceptera längre utvecklingscykler som är nödvändiga för att uppnå skalbara och robusta implementationer allt eftersom produkten mognar. | They have been unwilling to accept longer development cycles that are necessary to develop more scalable and robust implementations as the product matures. | ☐ |
| 48 | Detta kräver djup förståelse av kraven och omfattande samarbete mellan utvecklare och kund, då detta innefattar att förfinas lågnivåspecifikationer iterativt. | This requires a thorough understanding of the requirements and extensive collaboration between the developer and customer, because it involves refining low-level specifications iteratively. | ☐ |
| 49 | Att göra om designen av arkitekturen blev väsentligt dyrare. | Redesigning of the architecture added significantly to the cost | ☐ |
| 50 | Icke-funktionella krav är ofta dåligt definierade och ignoreras under tidiga utvecklingscykler. | NFRs are often ill defined and ignored during early development cycles. | ☐ |

DEL B UPPSATSER 50p

Utgå från följande rubriker och skriv korta uppsatser på max 2 A4-sidor per uppsats. Var noga med att skriva läsligt. Svårlästa uppsatser ger poängavdrag. Börja på nytt blad för varje ny uppsats.

B1. Funktionella krav: Ge konkreta exempel på 4 olika specifikationsstilar, beskriv hur de används samt när respektive stil passar bra och/eller dåligt. (20 p)

B2. Kvalitetskrav (icke-funktionella krav):
utmaningar, kategorier och specifikationsstilar (15 p)

B3. Prioritering och releaseplanering i marknadsdriven kravhantering:
begrepp, utmaningar och metoder (15 p)

Nr Svar Kap

1 C	1 & 6	B1	Funktionella krav	sum 20
2 A	[AGRE]	För varje teknik: ca 5 poäng (viss poängöverföring m. tekniker om större/mindre)		
3 C	[INTDEP]	exempel	ca 1 poäng för bra exempel	1*4 4
4 D	1	beskrivning	ca 2 poäng för bra beskrivning av tekniken	2*4 8
5 C	[PRIO2]	passar	ca 2 poäng för bra beskrivning av när den passar	2*4 8
6 A	6.5 p 244			
7 E	8	B2	Kvalitetskrav	sum 15
8 A	8.2.10 sid 344	Utmaning	1 p per bra svårighet/problem/utmaning	max ca 4
9 E	2	Kategorier	1p per bra typ av kvalitetskrav inkl. förklaring	max ca 5
10 A	3.2	Specifikationsstilar		max ca 6
11 C	3		1 p per bara beskrivning av eller exempel på stil	
12 A	[MDRE2]		0,5 p för open metric	
13 A	7 & 8		0,5 p för open target (om förstått skillnaden OM/OT)	
14 D	1 & 3			
15 C	8	B3	MDRE, speciellt RP och PRIO	sum 15
16 E	6	begrepp	1p per relevant, väl förklarat RP/PRIO begrepp	max ca 5
17 C	6, 2	utmaning	1 p per bra svårighet/problem/utmaning	max ca 4
18 D	[QUPER]	metoder	2 p per bra beskriven metod/angreppssätt	max ca 6
19 E	6.6		prioriteringsmetod, beroende, laxtrappa, etc	
20 C	[SPM]			
21 D	1			
22 C	1			
23 D	8 sid 360	p max ca B1-B3 indikerar prioritet och inriktning		
24 A	6.3	bedömning görs för varje individuellt fall och p är inte strikt		
25 C	1.4			
26 C	[MDRE2]			
27 E	[PRIO1]			
28 C	6			
29 B	2			
30 E	5			
31 D	[PRIO2]			
32 C	6			
33 A	6			
34 E	3			
35 E	9			
36 C	9			
37 B	9			
38 A	9			
39 B	3.16 sid 150			
40 D	6			
41 D	8			
42 A	8			
43 E	8			
44 D	8			
45 C	8			
46 B	8			
47 B	[AGRE]			
48 C	[AGRE]			
49 A	[AGRE]			
50 D	[AGRE]			