

KursPM - Projektkurs inom datateknik och nätverksteknik HI1038 VT18 9,0 hp

Kursmoment

Kursen har två moment: RED1 - Redovisning, 3,0 hp med betygsskala P/F samt PRO1 - Projekt, 6,0 hp med betygsskala P/F

Projektmetodik, RED1 (3,0 hp)

I kursen ingår en del om projektmetodik. Grundstommen i detta material består av 3 föreläsningar samt en bok om agilt projektarbete.

Även om nivån inte är så djup, då det är en grundkurs, är det ett ganska lästungt material. Eftersom det är ett stort material kommer det krävas att du läser kontinuerligt. Internetteknikdelen kommer att ligga koncentrerat i början av kursen för att skapa utrymme för projektet i slutet, då ofta mycket av arbetet på projektet hamnar där.

Detta moment kommer att examineras kontinuerligt som en del parallellt med projektet genom att ni efter varje retrospektivmöte måste lämna in er personliga reflektioner. Detta ska ske minst 5 gånger under kursen.

Utöver reflektioner ni gör i samband med projektuppgiften tillkommer ett mindre teoretisk moment där ni individuellt ska läsa projektboken om Agil Utveckling av Anders Cajander. Denna finns tillgänglig via kurshemsida på canvas.

Ni ska skriva en sammanfattande reflektion. Där ni även berättar hur ni tillämpad kunskapen från boken på ert projekt. Följande ska vara inkluderade i er reflektion:

- Hur ni försäkrar er om att en task "verkligen" är klar?
- Hur ni har tillämpat olika metodstöd under projektet så som digital projektrum och versionshanteringsystem.
- Hur ni tillämpad och ser på ADT (abstract data types), source och header filer med avseende på uppdelning, oberoende och gränsdragning.
- Sist hur ni tillämpat och ser på de idéer om loggning som finns i boken.

Deadline 14 april ca 1000 ord. Er sammanfattande reflektion ska innehålla källhänvisning till boken olika delar där de så är lämpligt.

Ni kommer efter strax efter deadline få två andra rapport att för peer-reviewing med deadline den 17 april.

Utöver detta så ingår även en muntlig och skriftlig presentations övning med Gunnilla Nuclér. Denna kommer vara behövlig då de kommer att lottas vem i gruppen som kommer genomföra slutpresentationen olika delar.

Examination

RED1 examineras med kontinuerlig under kursen och avslutat med en individuell reflektion.

Detaljplanering

17/3	Föreläsning Introduktion
18/3	Föreläsning Projektarbete
24/3	Föreläsning Projektarbete
24/3	Föreläsning SDL
27/3	Föreläsning Stora program
30/4	Föreläsning SDL net
31/4	Föreläsning Projektarbete
1/4	Föreläsning Versionshantering och Testning
7/4	Föreläsning i Presentationsteknik
	Handledning varje vecka
5/5 – 7/5	Övning i Presentationsteknik
25/5	Redovisning
1/6	Examination

Projekt, PRO1 (6,0 hp)

Denna del av kursen ska ge en fördjupning och praktisk tillämpning av kunskaper inhämtade från tidigare kurser om ingenjörsmetodik, programmering, operativsystem samt digital- och mikrodator teknik.

Förutom de mål som anges i kursplanen, se <http://www.kth.se/student/kurser/kurs/HI1038>, tillkommer specifika mål som kan se något olika ut beroende på om du väljer programmerings eller nätverksinriktning för projektet.

Tillkommer i detta projekt

Följande ska vara implementerat i den färdiga produkten:

- Prototypen ska vara en nätverksapplikation enligt Klient-Server-modell, med multipla klienter. Det är önskvärt att servern skrivs för plattformen Unix/Linux.
- Ett genomtänkt applikationsnivåprotokoll för nätverkskommunikationen ska upprättas. Val av TCP och/eller UDP för överföringen ska motiveras.
- Koden ska huvudsakligen vara skriven i programmeringsspråket C.
- Koden ska vara väl skriven m.a.p.
 - Koddokumentation i form av kommentarer
 - Sammanhörande data ska organiseras med hjälp av egendefinierade datatyper, struct, tillsammans med tillhörande funktioner.

- Datatyper och tillhörande funktioner ska vara organiserade i återanvändbara moduler; i C i form av header- och källkods-filer (.h- och .c-filer).

- Testning av prototypen ska utföras och dokumenteras.
- En användarmanual skriven enligt anvisningar ska produceras.

Det är dessutom önskvärt att:

- Klienten har ett grafiskt användargränssnitt tillsammans med någon form av händelsehantering för tangenttryckningar och/eller mus-händelser.
- Parallell programmering med trådar och/eller processer på server och/eller klient-sidan. Hanteringen av gemensamma resurser ska då vara synkroniserad.

För fullständig information, se betygskriterierna för momentet PRO1.

Examination, PRO1

PRO1 - Projekt, 6,0 hp, betygsskala: P/F

Godkänd projektrapport, muntlig presentation av projektet samt godkänd prototyp (PRO1; 6,0 hp),

Litteratur

Projekthandboken från projektarbetet.

Advanced Linux Programming, kapitel 1 - 5, <http://www.advancedlinuxprogramming.com/>

Övrigt material, t.ex. i anslutning till föreläsningar, finns under kursens aktivitet på KTH canvas, kth.instructure.com

Projektuppgifter

Förslag på projektuppgifter presenteras i separat dokument på KTH canvas. Gruppen kan också föreslå en egen projektuppgift som uppfyller kraven ovan. Förslaget ska godkännas av examinator i god tid före arbetets början.

Handledare

Varje grupp kommer att tilldelas en handledare. Handledare är:

Mohammed Mutaliev, mutaliev@kth.se

Robin Eriksson Ståhl, rassta@kth.se

Christian Säldebring, sald@kth.se

Torgny Forsberg, torgny@kth.se

Projektarbetet

Protokollen från etappmötena ska bl.a. dokumentera vad var och en av medlemmarna gjort under den gångna perioden. Vi ska alltid eftersträva att arbetet fördelas rättvist och i en bilaga till etapprapporten ska detta följas upp, se nedan om tidsrapportering.

Gruppen ska de dessutom presentera en plan över de arbetsdagar de arbetar med projektet. Dessa dagar ska inledas med ett kort stå-upp-möte ("daily scrum").

Förutom detta ska vissa delar av projektmetoden "Extreme Programming" implementeras i arbetet, se separat dokument.

Tidrapportering

Ni ska fortlöpande ha kontroll på hur mycket tid ni lägger ned på kursen. Projektdelen i kursen omfattar 7 hp, vilket motsvarar 4,67 veckors heltidsstudier per elev. Om ni är 4 personer i projektet har ni då totalt $4 \cdot 4,67 \cdot 40 = 747$ timmar till ert förfogande. Som nämnt ovan ska detta löpande dokumenteras i en bilaga till etapprapporterna. Total nedlagd tid för varje deltagare ska redovisas individuellt och ett medelvärde på totalt nedlagd tid under hela projektet ska beräknas. Gruppens medlemmar får inte avvika mycket från detta medelvärde, arbetet ska som sagt fördelas jämnt över gruppmedlemmarna.

Gruppkontrakt

Ni bör internt i gruppen skapa ett kontrakt där ni reglerar vem som gör vad, och vad konsekvenserna ska bli om någon inte uppfyller sina åtaganden. Det är t.ex. viktigt att handledarna informeras om någon deltagare faller i från projektet. Skriv kriterier för detta i kontraktet så blir det inga diskussioner om det inträffar.

Dokumentation

All dokumentation, inklusive källkod, ska publiceras så att alla, inklusive handledare, kan ta del av denna.

Versionshantering

Gruppen ska använda ett datorbaserat system för versionshantering, så att olika versioner av programkod, rapport och manual blir spårbara.

Applikationerna

Till den slutliga versionen av applikationerna ska bifogas färdigkompileerade och exekverbara enheter (exempelvis klient och server), samt en manual, som innehåller installationsinstruktioner, användningsinstruktioner och hur vanliga fel åtgärdas.

Möten

I projektet finns flera sorters möten.

Daily scrum

Ett kort möte varje dag där alla i teamet berättar vad de gjort, vad de ska göra och om de behöver hjälp. Kund får vara med (inte yttra sig), men detta är ett möte för team medlemmarnas planering av dagens arbete (kallas ibland stå-upp-möte).

Etappmöte (Sprintmöte)

Ett planeringsmöte som sker i början på varje sprint.

Product backlog för den avslutade sprinten diskuteras och analyseras, därefter demonstreras klar funktionalitet.

Sedan planerar alla deltagare i projektet vad man ska göra under den kommande sprinten och en ny sprint backlog skapas.

Sprint möten ska protokollföras.

Handledning

När det är bestämt hur ert projekt ska se ut får ni en handledare och en uppdragsgivare. Ni kommer överens med er uppdragsgivare om en kravspecifikation och skall agera mot uppdragsgivaren så som en professionell leverantör kan förväntas göra. Det innebär bl.a. att:

- Ni kommer ganska snart att veta mer om detaljerna i projektet än er uppdragsgivare. Uppdragsgivaren vill bara "ha en sak som funkar", det är er uppgift att lösa det.
- Det finns ingen lärare som planerar ert projekt på detaljnivå. Ni får själva leta reda på den information som ni behöver.
- Ni bör så snart som möjligt göra upp en tidsplan. Räkna realistiskt och med goda marginaler.
- Det är naturligt att det uppstår problem under ett projekt och det är förstås bäst om de kan undvikas, men viktigast är att de som ändå uppstår hanteras på ett bra sätt. Att vänta och se vad som händer är fel sätt.
- Professionalism innebär också att man meddelar sin uppdragsgivare om man inte blir klar i tid, gärna så snart man känner till det. Det innebär också att man frågar uppdragsgivaren om något är oklart i kravspecifikationen och reagerar om något i kravspecifikationen är ogenomförbart eller om man behöver ändra den. Ansvar för att slutresultatet blir bra ligger alltså på er, inte på den som har skrivit kravspecifikationen.

Ovanstående är typiska grundförutsättningar vid projektarbete i industrin. Kör ni fast finns givetvis handledarna där även som lärare.

Deadlines

24/3	Projektgrupperna presenteras
27/4	Projektstart. Val av projektuppgift och scrum master klart
1/4	Kravspecifikation med "ambitionsnivå" lämnas till handledare. Specifikation av arbetsdagar lämnas även det till handledare.
14/4	Inlämning av sammanfattning på projektmetodik
17/4	Sista datum för inlämnande av kravspecifikation justerad enligt handledares anvisningar. Peer-Review av sammanfattningar.
Vecka 16 – 21	Arbete med projektuppgift och rapport. Sprintmöten, varav minst 3 med handledare, samt schemalagda "dailyscrums".
17/5	Rapporten klar. Lämnas in på canvas samt till opponerande grupp.
25/5	Slutseminarium med muntlig presentation av projektet samt opposition. Skriftlig opponering lämnas till gruppen och på canvas.
1/6	Slutgiltigredovisning och slutrapport lämnas in via canvas.

Sprint möte

Resultatet, så långt, presenteras under 20 minuter för handledaren.

Slutseminarium

Varje grupp kommer att presentera sitt projekt under 12 minuter. Därefter följer 3 minuter med kommentarer och frågor från examinator och opponerande grupp (se nedan).

Det kommer ske låtning vid redovisningstillfället om vem i gruppen som kommer presentera vad. Det innebär att samtliga gruppmedlemmar skall kunna svara vara förberedde på hela projektets genomförande och rapportens innehåll.

Projektor som kan kopplas till laptop finns i salen.

Obligatorisk närvaro gäller hela dagen.

Om opposition

Vid slutseminariet ska varje grupp opponera på en annan grupps rapport. Den föreliggande rapporten ska noggrant gås igenom, och opponenter ska väcka diskussion i frågor som behandlas, visa på tänkbara alternativa lösningar, påpeka eventuella oklarheter och svagheter i rapporten, etc. Oppositionen ska genomföras i positiv anda, och syftar inte till att hitta så många felaktigheter som möjligt i den andra gruppens arbete. Opponeringen ska dels ges skriftligen till gruppen och muntligen presenteras i samband med redovisningen.

För att opponenter ska ha rimlig förberedelsestid ska varje grupp lämna sin preliminära skriftliga rapport till sina opponenter och bilda senast 17/5.

Rapporten

Projektrapporten syftar till att träna er förmåga att uttrycka er i skrift och den kommer därför att granskas inte bara för sitt innehåll utan även med avseende på struktur och språkbruk.

Rapporten lämnas in via Bilda på utsatt tid. Det finns två deadlines för rapporten, en första version, som ni kommer att få kommentarer på, samt en slutlig version. Båda versionerna av rapporten kommer att plagiatkontrolleras.

Strukturen på rapporten ska likna den som används för teknisk dokumentation. Typiskt innehåller en sådan:

Sammanfattning (Abstract): Här ges en kort sammanfattning av hela rapporten, med frågeställning och de väsentligaste svaren. Denna text klipps ofta ut ur rapporten och görs tillgänglig i diverse databaser, så den ska "stå på egna ben".

Introduktion (Introduction): Här ger man bakgrunden till projektet, vilka problem det ska lösa och vad andra har gjort för att lösa det.

Produktsbeskrivning (Product description): Här beskrivs ni produkten ni har gjort

Systemarkitektur (System architecture): Här beskrivs hur system är byggda inklusive övervägande. Hur har ni tänkt?

Specifikation (Specifikation): Vilka egenskaper har er produkt? Vad blev slutresultatet?

Referenser (References): Här listas alla källor man har använt. Listan ska vara numrerad och varje nummer ska refereras till i rapportens text.

Appendix (Appendices): Rådata och programkod och annat utrymmeskrävande och/eller svårläst material läggs lämpligen sist i rapporten i ett appendix.

Notera att den löpande texten i rapporten bör endast i undantagsfall innehålla ren kod. Viktiga delar av koden kan publiceras i ett appendix. Hela koden ska när projektet är färdigt, tillsammans med rapporten, användarmanual och binärer, publiceras på en webbsida. En referens till denna sida ska finnas med i rapporten.

Fusk

Med fusk menas att man med otillåtna hjälpmedel eller på annat sätt försöker vilseleda vid prov eller när en studieprestation annars ska bedömas. (Högskoleförordningen 10 kap § 1).

Det finns mycket material på Internet som kan hjälpa er att utföra ert projekt. Det är bra att titta på vad andra har gjort för att lära sig hur man kan göra. Den rika tillgången på material är samtidigt problematisk eftersom den kan fresta till fusk. Om ni använder eller låter er inspireras av material som ni inte har producerat själva så måste ni därför hänvisa till källan och ange på vilket sätt ni har använt den. Det gäller bland annat kretslösningar, programkod, bilder och text. Endast citattecken och hänvisning skiljer citatet från plagiatet.

Det djupare syftet med alla kurser är att vinna kunskap. En elev som fuskar och tar en färdig lösning från nätet genererar åtminstone två problem. För det första minskar inläringen, man brukar tala om att "fusk kortsluter lärandet". För det andra vilseleder man examinator som ska bedöma elevens prestation och sätta betyg. Betyget kommer inte att reflektera elevens kunskap utan elevens förmåga att fuska. Båda problemen är dåliga för både KTH och elevens förutsättningar till framgång i fortsatta studier och senare i arbetslivet.

Ibland hörs frågan "Hur mycket måste man ändra för att det inte ska räknas som fusk?" Frågan är felställd. Tänker man på det sättet fuskar man redan i praktiken. Man vill använda någon annans arbete och ändra det precis så mycket att det passerar en plagiatkontroll. Man fokuserar på ändringen snarare än innehållet. Därmed är lärandet kortslutet. Man bör i stället ta till sig materialet, använda det och när man har lärt sig vad det innebär formulera en text med egna ord och ange en referens till ursprungsmaterialet. Man kan också använda en ursprungstext i ett citat, med en tydlig hänvisning. Man måste dock själv förstå citatet om det ska kunna fylla någon funktion i rapporten.

Om plagiering upptäcks meddelas grupperna detta före den muntliga presentationen. Handledarna talar dock inte om *vad* som har hittats, gruppen får leta upp det själv och presentera en ändring vid presentationen. Allvarliga fall resulterar i disciplinnämndsansökan, lindriga fall med enstaka plagierade meningar eller bilder resulterar i nedsatt betyg eller underkännande.