



KURSPLAN

Introduktion till programvarudesign och arkitektur

Introduction to Software Design and Architecture

5 högskolepoäng (5 ECTS credit points)

Kurskod: PA1460
Nivå: Grundnivå
Fördjupning: G1F
Utbildningsområde: Teknik
Ämnesgrupp: Datateknik

Huvudområde: Datavetenskap, Programvaruteknik
Version: 4
Gäller från: 2019-01-19
Fastställt: 2017-11-30

1. Kursens benämning och omfattning

Kursen benämns Introduktion till programvarudesign och arkitektur / Introduction to Software Design and Architecture och omfattar 5 högskolepoäng. En högskolepoäng motsvarar en poäng i European Credit Transfer System (ECTS).

2. Beslut om fastställande av kursplan

Denna kurs är inrättad av dekan 2017-10-20. Kursplanen har reviderats av prefekten vid institutionen för programvaruteknik och gäller från 2019-01-19. Kursen ersätter PA1443 Dnr: BTH-4.1.1-1973-2017

3. Syfte

Arkitekturen och designen av ett mjukvarusystem påverkar i hög grad kvaliteten på systemet och kostnaden för utvecklingen. Utgående från grundläggande objektorienterade begrepp och designprinciper analyseras och modelleras ett systems struktur och beteende med hjälp av modelleringsspråket UML (Unified Modelling Language) i en strukturerad arbetsmetodik, Unified Process. Arbetsmetodiken utgår från en beskrivning av kundkrav och bygger en spårbar kedja via olika UML-modeller hela vägen fram till implementation och testning. Den strukturerade arbetsmetodiken och modelleringsspråket UML ger ett stöd till designarbetet, men för att skapa en hållbar programvarudesign krävs också en förståelse för grundläggande designprinciper och designmönster. Designmönster är generella lösningsförslag på vanligt förekommande problem, och som mjukvaruutvecklare förväntas man känna till och kunna anpassa dessa generella lösningar till de konkreta utmaningar man försöker lösa. Grundläggande designprinciper beskriver hur man fördelar olika typer av ansvar mellan klasser för att få löst kopplad och lättunderhållen programvarukod. I takt med att system växer måste

man också ägna uppmärksamhet åt dess arkitektur, dvs. dess övergripande struktur av delsystem och komponenter, ur såväl ett konstruktionsperspektiv som ett runtime-perspektiv. Analogt med designmönster finns det arkitekturmönster som underlättar skapandet och underhållet av större system med olika önskade kvalitetsegenskaper.

4. Innehåll

Kursen omfattar följande:

- Grundläggande begrepp inom objektorienterad modellering
- Introduktion till modelleringsspråket UML
- Introduktion till arbetsmetodiken Unified Process
- Grundläggande designprinciper, t.ex. låg koppling, hög sammanhållning, inkapsling, och polymorfism
- Introduktion till, och användning av Designmönster
- Testning, i synnerhet ur mjukvarudesignprocess-perspektivet
- Introduktion till mjukvaruarkitektur
- Introduktion till mjukvaruarkitekturstilar

5. Mål

Kunskap och förståelse

Efter genomförd kurs ska studenten:

- kunna visa förståelse för grundläggande principer i objektorienterad programvaruutveckling
 - kunna visa förståelse för UML som modelleringsspråk
 - kunna visa kunskap om grundläggande designprinciper
 - kunna visa kunskap om grundläggande designmönster
 - kunna visa kunskap om grundläggande mjukvaruarkitekturstilar
 - kunna visa förståelse för hur designarbete skapar förutsättningar för testning av mjukvarusystem.
- Färdighet och förmåga
- Efter genomförd kurs ska studenten:

- kunna uttrycka strukturen och beteendet hos ett system i termer av objektorienterade koncept
 - kunna korrekt använda UML för att uttrycka struktur och beteende hos ett system
 - kunna tillämpa grundläggande designprinciper för en objektorienterad design
 - kunna tillämpa grundläggande designmönster i en objektorienterad design
 - kunna tillämpa grundläggande arkitekturstilar för ett mjukvarusystem
 - kunna resonera om de kvalitetsegenskaper ett system med en viss arkitekturstil har eller bör ha
 - kunna resonera om och skapa en grundläggande testplan för ett objektorienterat system.
- Värderingsförmåga och förhållningssätt
Efter genomförd kurs ska studenten:
- kunna analysera källkod för eventuella förbättringar
 - kunna analysera och kritiskt diskutera en design för eventuella förbättringar.

6. Lärande och undervisning

Undervisningen består av föreläsningar och laborationer. Föreläsningarna behandlar centrala begrepp, tekniker och metoder inom objektorienterad design. Laborationerna syftar till att ge förståelse för hur begrepp, metoder och tekniker kan tillämpas i praktiken. Svenska men undervisning på engelska kan förekomma.

7. Bedömning och examination

Examinationsmoment för kursen

Kod	Benämning	Omfattning	Betyg
1905	Saltentamen	1 hp	A-F
1915	Laboration	4 hp	A-F

Kursen bedöms med betygen A Utmärkt, B Mycket bra, C Bra, D Tillfredsställande, E Tillräckligt, FX Underkänd, något mer arbete krävs, F Underkänd. Betyget på Saltentamen bestämmer kursens slutbetyg vilket utfärdas först när samtliga examinationsmoment godkänts. Slutbetyget på kursen kan höjas om betyget på laborationsserien är A eller B.

8. Kursvärdering

Kursansvarig ansvarar för att studenternas synpunkter på kursen systematiskt och regelbundet inhämtas och att resultaten av utvärderingar i olika former påverkar kursens utformning och utveckling.

9. Förekunskapskrav

För tillträde till kursen krävs att den studerande har avklarat programmering, motsvarande 7,5 högskolepoäng.

10. Utbildningsområde och huvudområde

Kursen tillhör utbildningsområdet Teknik och ingår i huvudområdet Datavetenskap och huvudområdet Programvaruteknik.

11. Begränsningar i examen

Kursen kan inte ingå i examen med annan kurs, vars innehåll helt eller delvis överensstämmer med innehållet i denna kurs.

12. Kurslitteratur och övriga läresurser

huvudbok:

- C. Larman, /Applying UML and Patterns/, Prentice Hall, 3rd Edition.

referenslitteratur:

- Gamma, Helm, Johnson, Vlissides, /Design Patterns, Elements of Reusable Object-Oriented Software/, Addison-Wesley Professional.

- L. Bass, R. Kazman, P. Clements, /Software Architecture in Practice/, Addison-Wesley, 2012.

