

Översikt

För mer detaljer om kurserna, se sida 2-5 – klicka på rubrikhuvuden för snabb ta dig till detaljerna för respektivekurs. För information du inte hittar i detta blad, kontakta oss via e-post info@thin2k.com eller ring oss på nummer 0046 708-36 15 32.

VIKTIGT: Kursmaterial för samtliga kurser är på engelska men kursen i sig hålls på svenska förutsatt att alla elever talar svenska.

[Feature driven requirement management \(2 dagar\)](#)

De som går denna kurs får gedigen kunskap inom agilt kravarbete och konkreta verktyg för att arbeta med krav baserade på features som bryts ned user stories och hela vägen till testfall. Kursen tar sin utgångspunkt i skapande av ett visionsdokument som bryts ned och vars innehåll prioriteras för effektivast möjliga genomförande.

[Feature driven design and documentation \(1 dag\)](#)

Du lär dig på kursen att på ett strukturerat sätt bryta ned features till designdokumentation och gör att du enkelt hittar designdokumentationen senare. Kursen tar upp spårbarhet till krav, UML, modellering och modelleringsverktyg

[IT architecture from requirements to SAD \(8 dagar\)](#)

Om du önskar ta steget och arbeta som arkitekt är detta kursen för dig! Kursen börjar med vad arkitektur är och de olika arkitekturrollerna och tar dig över vad som driver arkitektur till hur du faktiskt skapar sunda och hållbara arkitekturer och dokumenterar den.

[Sparx EA course \(2 dagar\)](#)

Kursen ger dig en stabil grund att stå på när du ska börja modellera med [Sparx EA](#). Du går lära dig allt ifrån hur du installerar, säkrar modellen från otillåten uppdatering, arbetar i gränssnittet, konfigurerar och utbyter modellinformation till andra med modellverktyget samt personer som inte har det genom rapportframställan.

Feature driven requirement management (2 dagar)

Kursmål

Efter kursen har du fått kunskaper om hur du arbetar agilt med krav baserat på konceptet "feature" som är ett centralt begrepp i vitt spridda och erkända agila ramverk som exempelvis SAFe, Scaled Agile Framework. Kursen ger den som går verktyg för att på ett effektivt sätt arbeta med features – med eller utan andra ramverk. För de som inte arbetar med SAFe eller liknande ramverk erhålls kunskap om hur man löpande skapar värde med bibehållen översikt och kontroll och för de som arbetar med SAFe eller liknande ramverk erhålls kunskap om hur du bygger upp en kravmassa som på ett sömlöst sätt passar in i det featuredrivna ramverket.

Elever får lära sig vad agilitet är och förstå det agila manifestet, att skapa visionsdokument, modellera intressenter, fånga behov, nyttor, features, user stories, kvalitetsegenskaper och testfall samt hur man bör gå tillväga för prioritering och planering av krav i agila projekt. Du får också en insikt av vad som kan förväntas av strukturerad design av varje feature.

Vem bör deltaga?

Kursen vänder sig främst till dig som arbetar som kravanalytiker eller produktägare och önskar få praktisk kunskap om olika typer av krav samt hur du prioriterar och planerar dem och över-sätter till testfall.

Övningar

Genom ett flertal praktiska övningar i grupp och vid PC befästs dina kunskaper inom featuredriven kravfångst.

Lärare

Thomas Gericke är Systemvetare, utbildad vid Stockholms Universitet med examen 1995. Thomas har arbetat operativt samt varit lärare i både öppna och kundorienterade kurser. Hans drivkrafter är att skapa ordning och struktur och maximera kundnytta genom effektiva processer för allt från verksamhetsmodellering över krav på system till lösning.

Förkunskaper

Du bör ha allmänna kunskaper kring IT och grundläggande förståelse eller erfarenhet av kravhantering av IT-system.

Sammanfattning

Features ("systemegenskaper") är idag ett centralt begrepp för att skapa ordning och tydligt mätbart värde i agila projekt. Begreppet definieras enligt det erkända och vitt spridda ramverket SAFe, Scaled Agile Framework, som "IT-tjänster som täcker användares behov". Features är den minsta enhet som på egna ben levererar verkligt värde – när en feature levereras/omformas till IT-stöd så förbättras en organisations förmåga inom något område. Features behövs för att skapa kontroll över stora mängder krav – features är tillräckligt stora för att ge överblick och förståelse jämfört med exempelvis user stories som ofta blir för små och många till antalet för att total kontroll ska kunna skapas.

De som går denna kurs får en gedigna och konkreta verktyg för att arbeta sant agilt med krav baserat på konceptet "feature" som inför utveckling bryts ned user stories och hela vägen till testfall. Kursen inleds med en genomgång av det agila manifestet och vad agilt arbete innebär.

Kursen fortsätter sedan med framtagande av en vision för ett tänkt projekt för att ena projektmedlemmar och skapa samsyn. Visionen konkretiseras genom att varje grupp skapar ett visionsdokument innehållande visionsformulering, intressenter, intressentbehov och en lista över features som täcker de olika behoven som har fångats för respektive intressent. I samband med framtagande av vision initieras även arbete med kvalitetsegenskaper och eventuella begränsningar som finns för systemutvecklingsarbetet.

Efter att en vision har skapats får elever lära sig hur features prioriteras på ett maximalt effektivt sätt så att verksamhetsnytta, risk mm utgör en bas för en rationell planering för både funktionalitet och arkitekturarbete. Prioriterade features bryts sedan ned i user stories som i sin tur passas in i iterationer/sprintar för att till sist detaljeras i form av testfall. Kursen tar även informationsmodellering och hur den förhåller sig till olika typer av informationsmodeller inom en verksamhet.

Notering: Kursen är en systerkurs till *Feature driven design and documentation* som är en kurs som tar upp hur features på ett strukturerat sätt kan brytas ned till spårbar designdokumentation i modeller. Båda kurserna kan dock gås separat. Notera även i detta sammanhang att de olika kurserna har olika målgrupper - *Feature driven requirement management* riktar sig till produktägare och kravanalytiker och *Feature driven design and documentation* riktar sig till designers/utvecklare.

Kursinnehåll/kapitel

- Introduction
- Agile
- Create a vision
- Detail features
- Quality attributes
- Prioritize features
- User stories
- Information modeling
- Test cases
- Wrap up (and what to expect from design)

Utlåtande från kund som gått kursen

"Jag gick kursen feature driven requirement management för Thomas Gericke. Kursen var väldigt pedagogisk och strukturerat upplagd. Efter kursen hade jag både verktyg att direkt införa förbättringar i vår verksamhet men jag hade även fått en större förståelse för hela området och inom vilka delar jag och vår verksamhet behövde utöka kunskap. En givande och intressant kurs som öppnat nya dörrar."

Sara Schill

VD, Winassist System AB

Feature driven design and documentation (1 dag)

Kursmål

Efter kursen har du fått kunskaper om hur du arbetar agilt med design och formgivning ("design") baserat på konceptet "feature" som är ett centralt begrepp i vitt spridda och erkända agila ramverk som exempelvis SAFe, Scaled Agile Framework. Kursen ger den som går verktyg för att på ett effektivt sätt arbeta med lösningssidan av features och featuredriven utveckling – med eller utan andra ramverk.

Efter kursen har du fått kunskaper om hur du på ett strukturerat sätt skapar design och dokumentation i ett featuredrivet projekt. Du får insikt om vad som bör dokumenteras, spårning och vad som kan uppnås med ett modelldrivet angreppssätt.

Vem bör delta?

Kursen vänder sig främst till dig som arbetar som designer och som önskar ökad kunskap om hur man dokumenterar design och gör att dokumentation enkelt kan hittas senare.

Övningar

Genom ett flertal praktiska övningar i grupp och vid PC befästs dina kunskaper inom design, UML och modellering.

Lärare

Thomas Gericke är Systemvetare, utbildad vid Stockholms Universitet med examen 1995. Thomas har arbetat operativt samt varit lärare i både öppna och kundorienterade kurser. Hans drivkrafter är att skapa ordning och struktur och maximera kundnytta genom effektiva processer för allt från verksamhetsmodellering över krav på system till lösning.

Förkunskaper

Du bör ha allmänna kunskaper kring IT och grundläggande förståelse eller erfarenhet av utveckling av IT-system.

Sammanfattning

Features ("systemegenskaper") är idag ett centralt begrepp för att skapa ordning och tydligt mätbart värde i agila projekt. Begreppet definieras enligt det erkända och vitt spridda ramverket SAFe, Scaled Agile Framework, som "IT-tjänster som täcker uppfyller användares behov". Features är den minsta enhet som på egna ben levererar verkligt värde – när en feature levereras/omformas till IT-stöd så förbättras en organisations förmåga inom något område. Features behövs för att skapa kontroll över stora mängder krav – features är tillräckligt stora för att ge överblick och förståelse jämfört med exempelvis user stories som ofta blir för små och många till antalet för att total kontroll ska kunna skapas.

Denna kurs ger eleven ett strukturerat sätt bryta ned features till designdokumentation som sedan enkelt kan hittas. Ett stort mål inom kursen är att skapa en grund för att kunna spåra designdokumentation till de features som var upphovet till designen eftersom det ofta är svårt att hitta teknisk dokumentation och koppla den till ett givet sammanhang senare. Denna spårning skapas genom modelldrivet arbete och länkar från design till features vilket säkrar att teknisk dokumentation alltid kan hittas och spåras till sitt krav/ursprung.

Kursen inleds med behovet av kravspårning och hur den på ett strukturerat sätt kan upprättas i modelleringsverktyg. Du får där lära dig hur vad features är, vikten av spårning till features och ”upptäckbarhet” av design, dvs att den går att hitta senare.

Kursen fortsätter sedan med vikten av modellering av design och stora notationer för att sedan fortsätta med informationsmodellering – vad det är och hur databasmodeller förhåller sig till andra typer av informationsmodeller.

På slutet av kursen summeras erhållna kunskaper inom respektive område genom att deltagaren får lära sig tekniker för att samla näraliggande designdokumentation till en enhet som är spårbar till en feature.

Denna kurs är mycket bra för organisationer som känner att det saknas strategi för att dokumentera systemlösningar och design och/eller att gjord dokumentation är svår att hitta senare alternativt saknar ett sammanhang till annan dokumentation som krav eller verksamhetsprocesser.

Notering: kursen är en systerkurs till kursen *Feature driven requirement management* på det sätt att den designdokumentation som beskrivs utgår från konceptet *Feature*. Organisationen kan satsa på designkursen utan *Feature driven requirement management* förutsatt att man arbetar med konceptet *Feature*. Notera även i detta sammanhang att det är olika målgrupper för de olika kurserna. Denna kurs riktar sig till designers/utvecklare och kravkursen mer till produktägare och kravanalytiker.

Kursinnehåll/kapitel

- Introduction
- Requirements and tracing
- Modeling and notations
- Information and other models
- Feature realizations
- Possibilities with model driven approach

IT-arkitektur från krav till SAD (8 dagar)

Kursmål

Efter kursen har du fått kunskaper om hur du identifierar och arbetar med arkitekturdrivande krav, skapar sunda arkitekturer och dokumenterar IT-arkitektur i formen av en SAD (Software Architecture Document).

Notering: kursen kommer att ges ut i en universitetversion mot en extra kostnad för de som önskar bli certifierade samt få bevis på sin kunskap i form av universitetspoäng.

Vem bör delta?

Kursen vänder sig främst till dig som arbetar som utvecklare, designer och som önskar arbeta med arkitektur. Kursen vänder sig också till dig som redan idag arbetar med arkitektur men önskar förkovra sig i ämnet.

Övningar

Genom ett flertal praktiska övningar i grupp och vid PC befästs dina kunskaper inom arkitektur. Kursen avslutas genom att studenter skapar en SAD som examineras vid separat tillfälle.

Om läraren Thomas Gericke

Thomas är Systemvetare, utbildad vid Stockholms Universitet med examen 1995. Thomas har arbetat med arkitektur och metodik i över tjugo år och har utbildat ett par hundra arkitekter genom åren – privat och inom Universitetet. Thomas är certifierad inom både IT-arkitektur och koncernarkitektur/-TOGAF och har lång erfarenhet av arkitekturverktyg och modellering.

Förkunskaper

Du bör ha goda kunskaper om IT och ha arbetat med utveckling i ett antal år.

Sammanfattning

Om du önskar ta steget och arbeta som arkitekt är detta kursen för dig! Kursen är en längre version än introduktionskursen och ger dig en bredare kunskap i syfte att faktiskt direkt kunna arbeta praktiskt inom området.

Kursen inleds med skillnaden på arkitektur, design och utveckling för att positionera ämnet och klargöra vad en arkitekt primärt arbetar med och vad han/hon primärt inte arbetar med. Kursen går sedan över i modellering och centrala notationer för att åskådliggöra arkitekturmodeller.

Kursen tar upp krav som driver arkitektur mer än andra krav – kvalitetsattribut och hur man som arkitekt förväntas säkra att alla krav på kvalitet är fångade, mät- och uppföljbara.

Deltagare får lära sig att prioritera krav för att eliminera risk men samtidigt skapa verksamhetsnytta samt arbeta med arkitekturmönster, designmönster och mekanismer för att uppnå kvaliteter som ställs på lösningar.

Kursen tar även upp arkitekturstilar och tjänsteorienterade arkitekturer som traditionell SOA och REST och integrationer mellan system.

Mot slutet av kursen behandlas processmodellering, driftsättningsvyer mm för att få alla delar som behövs för att dokumentera ett traditionellt mjukvarudokument (SAD).

På övrigt tid efter/mellan kursdagarna ska elverna som går kursen göra ett antal hemmauppgifter inklusive ta fram ett mjukvaruarkitekturdokument som kommer att bedömas av läraren.

Områden som denna kurs behandlar utöver introduktionskursen är målarkitekturarbete, IT governance, hur man uppnår sunda arkitekturer, informationsmodellering, modelleringsverktyg, databastyper/persistens, virtualisering, namnkonventioner, masterdata, BI, säkerhet, designdokumentation.

VIKTIGT: Denna kurs ges inte ut i samarbete med Mittuniversitetet och ger således inte högskolepoäng.

Kursupplägg

Kursen är totalt på fem dagar och hålls i två teoretiska block med en examinationsdag. Inom varje block ses studenter och lärare varje dag och mellan blocken är det luft/tid för övning, repetition och eget studerande.

- Block 1: Tre dagar
- Block 2: Tre dagar
- Block 3: Två dagar

På kursen kommer det att finnas datorer att tillgå, men studenten förväntas arbeta med övningar utanför kurstid varför det även där måste finnas tillgång till dator med Office-paket.

Kursinnehåll/kapitel

- Introduction
- What is architecture and what is it not?
- What an architect works with
- Modeling and notations
- Target and reference architectures
- IT governance
- Architecture driving requirements
- Architecture standards and frameworks
- Prioritization of architecture work
- Sound architectures and how to achieve it
- The canonical model
- Modeling tools
- Architecture and design patterns
- Mechanisms
- Service oriented architectures
- Integration
- Processes and threads
- Nodes and network topologies
- Persistence types
- Agile architectures and agile architecture work
- Virtualization and the cloud
- Naming conventions
- Test and quality assurance

- Master data
- Business intelligence
- Security
- Standard systems
- Organizing architecture work
- To document IT architecture
- IT design documentation
- SAD examination (valfri 9:e dag)

Praktiska övningar

- Create groups
- Three words
- Architectural goals and principles
- Most costly mistake
- Application for a solution architect
- Create an activity diagram
- Create a system landscape
- Create an architecture board
- Identify and quantify quality
- Choose an interesting standard, protocol or framework
- Prioritize requirements
- Choose tactics
- Create an information model
- Create a message
- Choose patterns
- Create mechanism chain
- Create a service architecture
- Choose integration strategy
- Process view
- Deployment view
- Choose database type
- Plan for resource allocation
- Decide virtualization strategy
- Decide naming conventions
- Automation of organizations
- Discuss the need for master data
- Integration strategy for BI
- Decide security architecture
- Staffing of architecture work
- Pros and cons of standard systems
- Demo of feature realization
- Create a SAD

Utlåtande från kund som gått arkitekturutbildningen

"Jag gick THIN2Ks kurs i arkitektur under våren och sommaren 2019 med en tanke att snabbt få en förståelse för hur vi på företaget bör strukturera vår mjukvara för att det ska bli bra. Kursen visade sig vara något mycket, mycket större än jag trott från början och jag fick förståelse för vad arkitektur går ut på och lärde mig ett helt yrke kan man säga. Det som var mest intressant var kapitlet med tjänsteorienterad arkitektur och i synnerhet delarna som berörde mikrotjänster. Läraren, Thomas Gericke, har en enorm kunskap i ämnet samt en pedagogisk läggning som gör att jag numera tänker på ett helt annat sätt än jag gjort tidigare. Vidare har jag fått ett helt vokabulär att tala med våra utvecklare med.

Gå denna suveräna kurs om du planerar att arbeta som IT-arkitekt!"

Stefan Jonsson

Chefsarkitekt, WinAssist System AB

Sparx EA course (2 dagar)

Kursmål

Efter kursen har du fått kunskaper om hur du installerar, anpassar, arbetar säkert, modellerar, analyserar modeller, samt utbyter information mellan modeller och andra delar inom verktyget Sparx EA.

Vem bör delta?

Kursen vänder sig främst till dig som arbetar med modellering i någon form och önskar öka spårbarheten och samspelet mellan olika delar av arkitekturinformation.

Övningar

Genom ett flertal praktiska övningar i grupp och vid PC befästs dina kunskaper inom modellering och verktyget Sparx EA.

Lärare

Thomas Gericke är Systemvetare, utbildad vid Stockholms Universitet med examen 1995. Thomas har arbetat operativt samt varit lärare i både öppna och kundorienterade kurser. Hans drivkrafter är att skapa ordning och struktur och maximera kundnytta genom effektiva processer för allt från verksamhetsmodellering över krav på system till lösning.

Förkunskaper

Du bör ha allmänna kunskaper kring IT och grundläggande förståelse för vikten av modellering.

Sammanfattning

Kursen ger dig en stabil grund att stå på när du ska börja modellera med Sparx Enterprise Architect (EA) – det troligen mest spridda modelleringsverktyget i Sverige och världen. Ett stort mål inom kursen är att lära ut hur Sparx EA kan skapa en "single-point-of-documentation" för hela organisationer, dvs en dokumentation som lagrar en sanning som föds av många deltagare genom kollaborativt arbete.

Du får inom kursen lära dig hur Sparx EA installeras, ställs in, viktiga aspekter när du arbetar delar på och arbetar med gemensamma modeller, hur du arbetar säkert i gemensamma resurser.

Kursen tar även upp kända notationer och modelleringsstandarder samt hur du kan anpassa verktyget för att lägga till modelleringsstandarder just din organisation behöver eller använder. Deltagare får även lära sig vikten av spårbarhet mellan modellelement och nivåer samt hur du gör analyser mellan de spårade arkitekturelementen.

Vidare tar kursen upp viktiga funktioner som gemensamma ordlistor och begreppsmodeller, planeringsstöd för projekt, hur verktyget kan byggas ut.

Deltagare får lära sig att byta ut information mellan modeller och hantera resurser, hur verktyget kan kopplas till andra typer av verktyg som databaser samt hur verktyget kan sammanställa information med hjälp av det inbyggda rapportverktyget.

Denna kurs vänder sig till organisationer som känner att de måste ta till nya tag för att ta kontroll över verksamhetsdokumentation och spårning mellan olika typer av dokumentation genom en

modellbaserad strategi med hjälp av det kostnadseffektiva, populära och kraftfulla verktyget Sparx EA.

Kursinnehåll/kapitel

- Introduction
- Installation and setup
- About modeling tools and repositories
- The workbench
- Security and concurrent work
- Notations and modeling standards
- Modeling architectural elements
- Architecture drivers and traceability
- Model analysis
- Glossary and canonical modeling
- Planning and collaboration support
- Tool configuration and stereotypes
- Forward and reverse engineering
- Manage resources
- Model-to-model exchange
- Model content communication and reporting
- Extending the tool capabilities

Utlåtande av kund för Sparx EA-utbildningen

"En grundkurs i EA som ger en god grund att bygga vidare på, men det var engagemanget och entusiasmen hos kursledaren Thomas Gericke som lyfte kursen, och smittade av sig på oss kursdeltagare. Jag kan varmt rekommendera denna kurs!"

Joakim Sjölin

Flintab AB