

Kurs-PM – Clean Code och testbar kod 30p

Beskrivning av kursen/innehåll:

Efter kursen ska den studerande behärska principer och tekniker för att skapa underhållsvänlig, skalbar och återanvändbar kod. Kursen ger kunskaper i att kunna använda sig av designmönster för att uppnå högre kodkvalitet. Den studerande lär sig moderna utvecklingsprocesser och att kunna tillämpa deras principer i utvecklingsarbetet. Syftet med kursen är att uppnå effektivitet och hög kvalitet i utvecklingsarbetet samt att kunna skapa kod av hög kvalitet som är lätt och enkelt att underhålla och bygga ut.

Teknologier/verktyg:

- Visual Studio 2022
- xUnit
- C#
- ASP.NET
- Docker
- RabbitMQ

Utbildare:

Niklas Hjelm – niklas.hjelm@iths.se

Datum & Veckoplanering:

2023-11-13 – 2023-12-23

Vecka	Dag - Plats	Ämne
46	Tisdag – På plats	Kursintroduktion – Lessons learned LIA
	Torsdag – Distans	TDD
	Fredag – På plats	UnitTester
47	Tisdag – På plats	Refactoring
	Torsdag - Distans	Handledning
	Fredag – På plats	Intro SOLID
48	Tisdag – På plats	Gästföreläsning – Bernt Johansson Design Patterns
	Torsdag – Distans	Handledning
	Fredag – På plats	Gästföreläsning – Bernt Johansson Design Patterns
49	Tisdag – På plats	Intro Microservices
	Torsdag - Distans	Handledning

	Fredag – På plats	Docker och Docker Compose
50	Tisdag - Distans	API Gateway
	Torsdag - Distans	Handledning
	Fredag – På plats	Message Queues (RabbitMQ)
51	Tisdag - Distans	Handledning
	Torsdag – Distans	Handledning
	Fredag - Distans	Handledning

Kurslitteratur:

Bloggar, handouts, kompendier och videoklipp som utbildaren tillhandahåller.

Rekommenderad läsning:

Clean Code: A Handbook of Agile Software Craftsmanship – Robert C. Martin

Design Patterns: Elements of Reusable Object-Oriented Software – Gang of Four

Examinationsformer:

Kunskapskontroller görs under kursen genom en tenta, en individuell inlämning samt ett grupparbete.

Betygssättning:

Studenten ska få sitt kursbetyg senast 3 veckor efter det att kursen avslutats

Betygskriterier:

Kursen bedöms med betygen Icke Godkänd, Godkänd eller Väl godkänd. Grunden för betyg avgörs av hur väl studenten har uppfyllt betygskriterierna.

Studenten ska få sitt kursbetyg senast 3 veckor efter det att kursen avslutats.

Icke godkänt (IG) Den studerande har fullföljt kursen men inte nått alla mål för kursen

Godkänt (G) Den studerande har nått samtliga mål för kursen

Kunskaper

1. Design patterns (factory, singleton, observer, solid etc)
2. TDD (Test Driven Development) / BDD (Behaviour Driven Development)
3. SOA (Service-Oriented Architecture) och Microservices
4. Enhetstester
5. Bygga applikationer med hög maintainability (underhåll) och skalbarhet (enkelt utöka funktionalitet)
6. Patterns och anti-patterns

Färdigheter

7. Skapa kod som är underhållsvänlig, lättläst, kommenterad, skalbar och återanvändbar
8. Skapa tester för att säkerställa en hög kvalitet på koden samt en hög test coverage
9. Använda design patterns för att lösa kodproblem
10. Identifiera och undvika defekta processer och implementationsmönster (anti-patterns)

Kompetenser

11. Skapa kod av hög kvalitet med hög test coverage
12. Skapa underhållsvänlig kod för att åstadkomma en hög maintainability
13. Skapa skalbar och återanvändbar kod för att enkelt kunna bygga ut funktionalitet
14. Skapa feltoleranta applikationer och system
15. Underlätta för testning och underhåll av kod genom TDD/BDD

Väl godkänt (VG) Den studerande har nått samtliga mål för kursen. Den studerande kan dessutom:

1. Själständigt och med säkerhet skapa kod som är underhållsvänlig, lättläst, kommenterad, skalbar och återanvändbar
2. Själständigt och med säkerhet använda design patterns för att lösa kodproblem samt kunna analysera och föreslå förbättringar

Examinerande moment (Labbar, projekt, tentor, rapporter etc. som är betygsgrundande):

Innehåll	Introduktion	Deadline	Täcker upp betygsgriterier
Labb 1 SOLID och TDD	V.46	V.48	1, 2, 4, 5, 6, 7, 8, 9, 11, 12, 13, 15
Labb 2 Grupp Microservices	V.49	V.52	1, 3, 5, 6, 7, 9, 10, 11, 12, 13, 14
Hemtenta	V.50	V.51	1,2,3,4,5,6