

Uppgift 1: Scrum (4p)

Scrum definierar en mängd roller, artefakt, möten och spelregler. Berätta vilka metoder Scrum erbjuder för att lösa problemsituationerna nedan. Det räcker inte att du listar termer. Berätta vad varje botemedel som Scrum erbjuder handlar om och hur den hjälper med problemet som beskrivs.

- Silva och Jussi grälar ofta om ifall enhetstest bör skapas för nya funktionaliteter.
- Jami, som är chef för företagets marknadsavdelning, får ofta briljanta ideér som bör implementeras genast oavsett vad som är på hälft.
- För ofta märker man att ett par programutvecklare, speciellt Kristian och Petrus, har båda kodat varsin version av samma user story. Många dagars arbete kan gå till spillo.
- Ibland är det oklart vilka storyn som borde utföras under sprinten. Det beror lite på vilken av företagets chefer man råkar fråga, speciellt vdn Luukkainens åsikt avviker från de andra chefernas linje.
- Samma problem uppstår i projektet från månad efter månad, t.ex. Git används hur som helst, testerna fungerar ibland och ibland inte. Namngivningen av klasser och metoder är ganska arbiträrt.
- Kalle tycker det är roligt att få programmera helt i fred. Den dåliga sidan är nog den, att ingen riktigt vet när något överhuvudtaget blir färdigt. Ibland tar det månader.
- Dom här sprinterna är roliga. Det är bara lite kaotiskt ibland när ingen riktigt vet vad man borde göra under sprinten. Det är ett mysterium hur utförandet av de olika user storyna påverkar den existerande koden.
- Vi har litet det problemet, att fast vi oftast får allt implementerat, kan det inte i slutet av sprinten sättas i produktion eftersom test-teamet i Indien fungerar med flera veckors fördröjning

Uppgift 2: Kravhantering (4p)

Din kamrat har fastställt följande user story som beskriver *betalningsfunktionaliteten för köp i en nätbutik*. Vad är bra och vad är dåligt och vad fattas från user storyn? Hur skulle man kunna förbättra user storyn?

När köpen i varukorgen betalas (förutom kreditkort fungerar MobilePay och andra möjliga betalningsalternativ), skall det i databasens ORDERS-tavla skrivas leveransadressen. Även tavlan PRODUCTS, som upprätthåller lagersaldot, skall uppdateras. Det lönar sig att observera, att försäljningen under Black Friday är väldigt. Programvaran skall ändå fungera snabbt även under en enorm användarpik. Det lönar sig att med samma programmera en möjlighet att bestämma hur leveransen arrangeras, genom posten eller någon annan leverantör t.ex. UPS. Denna story skall implementeras genast, den är livsviktig.

Uppgift 3: Termer (3p)

Är följande påståenden sanna eller falska?

- Programvarans validering sker genom enhetstestning
- Om användaren är väldigt nöjd, är troligtvis programvarans externa kvalitet i skick
- Testernas tillräckliga radtäckning kan säkras genom utforskande testning
- Med hjälp av ett Burn down -schema kan man uppskatta hur mycket programmerarna har arbetstimmar kvar till förfogande i sprinter som är på gång
- Att följa tillgänglighetsdirektivet (<https://vm.fi/sv/tillganglighetsdirektivet>) är ett så kallat icke-funktionellt krav
- Om man producerar en så kallad minimal viable product, skapar man inte oftast omfattande automatiserade tester för den
- Genom velociteten kan det vara möjligt att uppskatta när programvaran eller en del av den blir färdig
- Även i agila metoder dokumenteras kraven
- Kodgranskning är vattenfall, den hör inte till agil programvaruutveckling
- Scrum of Scrums är ett sätt att försöka se efter att ett företags olika teams kan arbeta enligt ett gemensamt mål
- Inkrementell arkitektur garanterar att kodens interna kvalitet hålls i skick
- Lean startup -metoden förbjuder att ta tekniskt skuld

Uppgift 4: Lean/DevOps (4p)

I samband med Lean pratar man om Just in Time (JIT) -produktion, samt om att minimera cykeltiden för produktionen.

Klargör vilken nytta en kort cykeltid har i programvaruproduktion, hur man försöker minska på cykeltiden i programvaruproduktion och vad DevOps har med saken att göra.

I ditt svar bör du även berätta vad begreppen Just in Time (JIT) -produktion, cykeltiden för produktion och DevOps betyder.

Uppgift 5 (del 1&2): Kodens kvalitet (5p)

Vad menas med programvarans interna kvalitet? (1p)

Vilka problematiska faktorer gällande programvarans interna kvalitet finns i koden nedan? (2p)

Förklara hur du skulle refaktorera koden nedan genom att tillämpa designmönster eller andra lämpliga lösningar. Svaret bör innehålla skiss av kod. Det räcker inte med att lista namn på designmönster. Skriv ändå inte "javaboilerplate" (dvs. klass+tillgänglighetsbestämning osv.) i sin helhet. (2p)