Development process 28/3 2017

Utvecklingsprocesser

Systematiskt

Är det framgångsrikt så ska man kunna repetera processen

är det mindre framgångsrikt ska man kunna ändra.

vi har en plan med vad som ska göras

vi gör ordentlig design och ordentlig analys för att bli mer framgångsrika

Disciplinerade

Har vi en plan, har vi en process så ska vi vara tillräckligt disciplinerad för att följa den

Vi är målfokuserade, vad är det viktigaste att få ut ur processen. Vad kunden vill.

Mätbarhet

hur nära målet kom vi. Måste analysera vad som gick mindre bra och varför

Development

\*this

Driftsättning(operation)

Man ändrar aldrig i en driftmiljö

man arbetar i en testmiljö

Fler och fler tjänster sker i online och kan potentiellt sätt skada verksamheten ekonomiskt, man ändrar aldrig i driftmiljön.

Underhåll

80 90 % av systemets livscykel är underhåll. Vi skriver koden för att vi själva ska förstå vad datorn ska göra.

Process vs Project vs Product

Att välja vad man ska fokusera optimeringen mot

Produkten måste bli klar i tid.

för Projektet antingen mer tid och resurser för att lyckas eller så skär man i vad som ska göras.

och påverkar produkten.

väljer man att optimera för ett område kommer det att påverka andra områden.

Use Case Diagrams.

vi har en spelare och en spelmaskin

aktörer ritas som streckgubbar och systemet ritas som lådor. lådan är “system boundry”

i lådan har vi “roll dice(play game)”, “login”, “logout”, dessa ritas i ovala cirklar

Aktören(spelaren) är inblandad i alla dessa use cases

Use cases.

Use case: roll dice

aktörer: Player

Description: kort beskrivning av vad som händer. Player rolls dices if >7 win else lose, score.

Conceptual Model(Domän modell) för roll dice

Player, Dice och Score är tre koncept (leta substantiv för att hitta koncept).

Skriv koncepten i rektanglar.

Player använder Dice(rolls).

Dice har ett Score.

Lyft in Score som ett attribut i Dice, passar det inte vid ett senare tillfälle kan man lyfta ut det.

Game är ett koncept.

game kan ha ett state som attribut som kan vara win eller lose.

Man fortsätter och lägger till i samma domän modell med resterande use cases.

man skaffar sig en ide om hur de olika identiteterna hänger samman.

SSD, system Sequence Diagram för Roll Dice

ett för varje use case.

Aktören som arbetar med systemet och vi har :Systemet

Tanken med ett SSD är att vi har en låda och vad kan man göra med systemet och hur reagerar systemet. Hur aktören uppfattar vad systemet gör.

Player -> roll Dice() -> :system

Player <- Score <- System

Player <- win/lose <- system

Sekvensdiagram, man tittar på vad som finns inuti systemet, vad som händer i systemet.

man tittar på konkreta instanser av klasser

:system

roll Dice ->

s1 = roll() -> d1:Dice

s2 = roll() -> d2:Dice

<-Score

Två Klasser, System och Dice

System har en relation till Dice

i System har vi en funktion rollDice() och variabeln score

i Dice finns funktionen roll() och variabeln value

Jag gissar att jag ska spara score i system och gissar att Dice ska hålla ett värde. utifrån sekvensdiagramen.

Projekt planering

Vad behöver vi veta

vad som ska göras och när det ska vara klart

hur stora resurser har vi

vilken process ska användas

har vi rätt resurser

hur lång tid ett visst moment tar

Work Breakdown Structure

Search for room - implementation, gui design, database design, testing

implementation - DB, Logic, GUI

testing - create test, run tests

Börja med use cases, prioritera dem.

GANTT Schema, en kalender för olika uppgifter, plocka löven ur WBS och lägg i en lista för vad som ska göras. Tips för att göra tids estimat, Gör dem ofta.

Tracking progress

Rapportera tid eller progress

Mängd tid/pengar spenderat

Antal uppgifter klara

Story Points

Fibonacci serien istället för att säga en viss tid.

Team tar olika lång tid på sig för uppgifter

ett verktyg för att kunna göra en hastighets uppskattning

Earned Value Chart

Tid på en axel och progress på andra axeln, kan vara story points.

lägga till en linje för kostnad. kan man se om man ligger efter

earned value, hur mycket är levererat. hur mycket har jag kunnat leverera. ex hur många story points som jag har kunnat avsluta

Burndown Chart

tidsaxel och progress axel, en linje - ursprungsplan för tidsåtgången, en linje för replaning som sjunker.

Risk planering

Identifiera risker

Utveckla en plan för att hantera risken

En risk är en sannolikhet att någon utomstående händelse inträffar och påverkar projektet

påverkar tiden och resurserna

påverkar quality of perfomance