

# Proj-Iteration 4&5

PVG/Coaching

**Boris Magnusson**  
Datavetenskap LTH

PVG/Coach 2011

Proj-Iter4&5 : 1

## Grov plan för releaser

Release 1/1B innehåller den funktionalitet (eller nästan) som behövs för att hantera tävlingen i Påsk.

Release 2 skjuts en vecka till iteration 5

Inför release 3 tillkommer stöd för funktioner före och efter själva tävlingen, om tiden och ekonomin medger.

PVG/Coach 2011

Proj-Iter4&5 : 2

# Leveranser

Efter iteration:

- 4 - Teknisk dokumentation (första version)
  - Granska andra gruppens tekniska dokumentation ver 1.
- 5 - Release 2 (system, teknisk dokumentation, källkod)
  - Granska andra gruppens R2 och förbered presentation
- 6 - Release 3 ( -"-)
  - Granska andra gruppens R3 före ev justeringar av presentationen

Ett annat team skall granska ert system och presentera det på det avslutande onsdagsmötet.

Allt ovan skall levereras dels till kunden och dels till det andra teamet (se hemsidan som har en tabell så att allt skickas till rätt team och kund!).

## Plan för granskning

Efter iteration 4 får ni från ett annat team en första version av teknisk dokumentation. En uppdaterad version av tekn dok lämnas sedan med R2 och kommer att vara en del av redovisningen i kursen.

Mellan iteration 4 och 5 skall ni preliminärgranska det andra teamets preliminära tekniska dokumentation. Se hemsidan för detaljer och hur ni skickar er rapport.

Mellan iteration 5 och 6 skall ni granska det andra teamets release 2 och förbereda en presentation.

Efter iteration 6 får ni även det andra teamets slutrelease för ev beaktande inför presentationen.

Sista ons/torsdagsmötet skall ni presentera deras system och redovisa era synpunkter.

# Release 2

Prioriteringen för release 2 är:

- A) Avhjälp ev brister i release 1B - baserat på feedback från kunden
- B) Övriga stories t o m 25.
- C) Html- formatterat resultat
- D) Integrerad systemuppläggnig

Om ni bedömer att inte både C och D kommer att hinnas med diskutera med kunden vilken ni börjar med.

Dessutom skall alltså två versioner av teknisk dokumentation göras.

Om det finns behov av ytterligare arbetsuppgifter finns story (33,34) inför Release 3 med på slutet.

# Release 3

Stöd för webb-baserad anmälan och insamling av uppgifter före tävlingen (story 34).

Story 33 Kodkvalité, Stabilitet

Stöd för andra resultat från systemet: ställning i tävlingen, information till förarna.

# Story 29 Integrera html-baserad resultatlista.

Efter en lyckad Story 28 återstår att integrera lösningen med det existerande sorteringsprogrammet.

Observera att båda typerna av resultat (med och utan sortering) skall kunna genereras i båda typerna av format.

Kunden ser gärna att koden som genererar html isoleras i egen/egna klasser liksom den existerande koden som genererar utmatning i Excelformat.

Acceptanstest - jämför t ex med resultatet från Pansartrofén. (fmck.malmo.enduro.just.nu Pansartrofén Resultat)

Prova med etapplopp-full

PVG/Coach 2011

Proj-Iter4&5 : 7

Estimate: \_\_\_\_\_

Depends on story:

Affects story:

Classes to be changed:

Classes to be updated:

Analysed by:

Coded by:

Comments/clarifications (see back).

Tasks:

PVG/Coach 2011

Proj-Iter4&5 : 8

# Story 30 Integrera med Klient/Server-lösning

Efter en lyckad utveckling av Serverlösning(story 27) återstår att integrera med existerande program. Dock finns det ett antal saker man bör observera:

- Välj om lösningen baseras på applets eller java-program
- Klient/registreringsprogrammet måste spara registreringar på fil lokalt även om de skickas över nätverket
- Serverprogrammet måste spara de registreringar det mottagit på fil
- Efter program/nätverkscrash måste förlorade registreringar kunna skickas över till Servern (synkronisering)

Vad som än händer får registreringar inte gå förlorade!

Estimate: \_\_\_\_\_

Depends on story:

Affects story:

Classes to be changed:

Classes to be updated:

Analysed by:

Coded by:

Comments/clarifications (see back).

Tasks:

# Story 31: Tekn dokumentation

Systemet skall dokumenteras så att det kan vidareutvecklas av ett annat team. All dokumentation skall finnas incheckad i repositoryt. Följande dokumentation skall finnas:

En textfil som heter README och som ligger i översta katalogen och som innehåller information om hur man hittar i filstrukturen.

Dokumentation av hur man bygger systemet.

Ett dokument som beskriver arkitekturen hos systemet. Till exempel: Vilka program och filer finns? Vilka huvuduppgifter har de olika programmen och filerna? Hur samverkar de (exemplifiera med några centrala stories)? Hur kan programmen distribueras på olika datorer? För varje program, beskriv översiktligt den interna strukturen med de viktigaste klasserna.\* För varje typ av fil, beskriv filformatet noggrant.

och vilka designmönster som har använts.

PVG/Coach 2011

Proj-Iter4&5 : 11

Estimate: \_\_\_\_\_

Depends on story:

Affects story:

Classes to be changed:

Classes to be updated:

Analysed by:

Coded by:

Comments/clarifications (see back).

Tasks:

# Story 32 Källkodsrelease 2

Arkitekturen för ert system skall granskas av ett annat team. Ni skall ta fram en källkodsrelease, dvs ett utcheckat repository med all källkod, makefiler, teknisk dokumentation, etc. Filerna behöver inte vara kompilerade.

Källkodsrelease 2 skall utgöras av en tar-fil som innehåller alla ovanstående filer. Källkodsrelease 2 skall motsvara (användar-)release 2. (Det är därför den numreras 2 trots att vi aldrig gjorde någon källkodsrelease 1).

Källkodsreleasen skall innehålla teknisk dokumentation enligt story 31.

PVG/Coach 2011

Proj-Iter4&5 : 13

Estimate: \_\_\_\_\_

Depends on story:

Affects story:

Classes to be changed:

Classes to be updated:

Analysed by:

Coded by:

Comments/clarifications (see back).

Tasks:

PVG/Coach 2011

Proj-Iter4&5 : 14

# Story 33 Kodkvalitet och Stabilitet

Gå igenom källkoden och identifiera kod som har bristfälliga enhetstester. Implementera dessa tester.

Identifiera felsituationer, t ex fel i användardata, parametrar, fel i namnfilen, filer som inte finns. Förbättra programmet så att det inte kraschar utan ger utmatning som hjälper användaren att hitta och korrigera misstaget.

Provkör med acceptans-varv och acceptans-etapp

PVG/Coach 2011

Proj-Iter4&5 : 15

Estimate: \_\_\_\_\_

Depends on story:

Affects story:

Classes to be changed:

Classes to be updated:

Analysed by:

Coded by:

Comments/clarifications (see back).

Tasks:

PVG/Coach 2011

Proj-Iter4&5 : 16



# Story 34 Webb-anmälan

Innan tävlingen anmäler sig de tävlande på en web-blankett. De fyller i namn, klubb, klass, etc. och skickar iväg. På något sätt når denna info en av de tävlingsansvariga och filen Personuppgifter produceras. Se kursens hemsida för olika förslag på lösningar. Lämpligen struktureras lösningen i två delar:

- Konstruera webb-blanketten. Gör ett enkelt exempel som är lätt att modifiera för olika slags tävlingar (enduro, cykel, ridsport, ...) med olika klasser.
- Konstruera systemet som skapar filen Personuppgifter (denna uppgift behöver troligen delas in i flera mindre uppgifter, beroende på vald lösning)

Installationsmanual och användarhandledning (både för web-blanketten och för systemet som skapar filen Personuppgifter) behövs. Det skall vara enkelt att installera för en ny tävling med ny anmälsansvarig. Gör en release av detta lilla system (med html-sida, jar-fil, installationsmanual och användarhandledning).

PVG/Coach 2011

Proj-Iter4&5 : 17

Estimate: \_\_\_\_\_

Depends on story:

Affects story:

Classes to be changed:

Classes to be updated:

Analysed by:

Coded by:

Comments/clarifications (see back).

Tasks:

PVG/Coach 2011

Proj-Iter4&5 : 18

# Tracker

Story	Vem	Tid?	Tid!
29 Integrera html-lös.			
30 Integrera serverlös.			
31 Teknisk dokumentat.			
32 Källkodsrelease2			
33 Kodkvalité, Stabilitet			
34 Webb-anmälan			