

TRACKABUS

BACHELORPROJEKT

Procesrapport
for
TrackABus

Author:

Gruppe 13038

Supervisor:

Michael Alrøe

13. december 2013

Versionshistorie:

Ver.	Dato	Initialer	Beskrivelse
1.0	12-12-2013	13038	Start på rapportskrivning

Godkendelsesformular:

Forfatter(e):	Christoffer Lousdahl Werge (CW) Lasse Sørensen (LS)
Godkendes af:	Michael Alrøe.
Projektnr.:	bachelorprojekt 13038.
Filnavn:	Procesrapport.pdf
Antal sider:	5
Kunde:	Michael Alrøe (MA).

Sted og dato: _____

10832 _____
Christoffer Lousdahl Werge

MA _____
Michael Alrøe

09421 _____
Lasse Lindsted Sørensen

1 Resumé

Indhold

1	Forord	3
2	Projektafgrænsning	4
3	Projektets fortræffeligheder	4
4	Forslag til forbedringer af projektet eller produktet	5

2 Forord

Dette projekt er udarbejdet af otte studerende på Ingeniørhøjskolen i Aarhus. Projektet er et eksamensprojekt på fjerde semester for IKT-linjen. Det er primært udarbejdet på baggrund af undervisning fra fag på fjerde semester; Windowsprogrammering, Databaser, Softwaretest og Interfacing. Således danner viden fra disse fire fag, samt fag fra de tre foregående semestre, baggrund for udviklingen af dette semesterprojekt.

3 Indledning

Sortering af elementer kan være en vigtig opgave i industriverdenen, og en automatisering af denne mekanisme kan spare meget arbejdskraft og dermed spare både hænder, penge, fejl, arbejdstimer og arbejdsskader. Dette dokument omhandler udviklingen af et softwareprodukt, der kan kontrollere en robot, samt gøre denne i stand til at sortere en række klodser alt efter deres materialetype. Det er en forholdsvis lille robot, der kun kan sortere små klodser, men det samme software ville kunne modificeres til brug på en større robot, som er i stand til at sortere så store elementer, at de ikke ville kunne flyttes med håndkraft.

Program, Silver Bullet Sort, er blevet udviklet til at håndtere ovenstående problemstilling for et firma, der har indkøbt en Scorbot ER-4u¹ med tilhørende programmel. Systemet skal fungere således, at en klods registreres på et transportbånd, hvorpå robotten måler og vejer klodsen. Herefter placerer klodsen i et materialespecifikt rum i en boks, der indeholder klodser med den givne materialetype. Programmet er lavet på baggrund af en række krav fra et udstedt produktoplæg. Desuden har en række møder med virksomheden også været med til at danne grundlag for fastlæggelse af programmets endelige funktionalitet. Programmet skal indeholde en database, som kan persistere diverse data i systemet, herunder data fra de elementer, der er blevet sorteret samt logge informationer om, hvordan processen er forløbet. Ligeledes skal programmet indeholde en IDE, som gør det muligt at lave brugerdefinerede programmer til sortering af elementer. Denne kan implementeres på flere måder, men den anbefalede metode indbefatter at omprogrammere robotten ved brug af en række funktioner, som firmaet har udleveret til udviklerne. Dog er disse

¹http://www.intelitek.com/admin/Products/uploads/File/File1_17.pdf

funktioner meget dårligt dokumenteret, så det har været op til udviklerne at undersøge anvendelsen af dem. Derudover skal det være muligt, at simulere robotten således, at et program kan testes, når robotten ikke er til rådighed. Desuden var det et krav, at der skulle laves en brugergrænseflade, hvorigennem alt kommunikation med robotten kunne kontrolleres. Det eneste hardware-mæssige krav kunden havde var, at få udviklet en vægt på baggrund af en vejecelle, som skulle indgå i processen, når materialetypen skulle findes.

Inden selve koden blev påbegyndt udviklet, blev der brugt en del ressourcer på at få tilegnet sig en fundamental viden om det, fra producenten, udleverede hardware og software. En domænemodel blev lavet på baggrund af kravene, hvilket skulle give et overblik over de specifikke dele i projektet. Herefter blev en kravspecifikation udarbejdet. Opbygning med at fastlægge de fundamentale krav før der blev udviklet kode, er primært kendt fra udviklingsmetoden *Unified Process*.²

For at det endelige produkt skulle stemme overens med firmaets krav og ønsker, er der gennem hele arbejdsprocessen holdt mange møder med virksomheden samtidig med, at virksomheden undervejs er blevet præsenteret for noget fungerende software, så det fra firmaets side har været muligt, at komme med ønsker til forbedringer og modificeringer af det endelige produkt. Da projektet er udviklet ved hjælp af agile arbejdsmetoder og principper, herunder primært *Scrum*³ og *Extreme programming*⁴, er det en naturlig proces at ændre kravene undervejs og revurdere projektet alt efter kundens behov.

Begreber og forkortelser

- UC: Use Case
- IDE: Integrated Development Environment
- UP: Unified Process.
- DLL: Dynamic-link library

²http://en.wikipedia.org/wiki/Unified_Process

³[http://en.wikipedia.org/wiki/Scrum_\(development\)](http://en.wikipedia.org/wiki/Scrum_(development))

⁴http://en.wikipedia.org/wiki/Extreme_programming

- WPF: Windows Presentation Foundation
- MVVM: Model View ViewModel

Læsevejledning

Nedenfor er listet en kort beskrivelse af hvert afsnit i dette dokument:

- Abstract og Resumé
 - Disse afsnit giver en kort beskrivelse af projektet på både dansk og engelsk.
- Indledning
 - Dette afsnit fortæller baggrunden for projektet, kravene til projektet, samt hvilke arbejdsmetoder og processer, der er anvendt. Herefter følger begreber og forkortelser samt en læsevejledning.
- Projektafgrænsning
 - Afsnittet fortæller kort om de begrænsninger, der er blev lavet, da projektet blev fastlagt.
- Projektgennemførelsen
 - Her præsenteres de forskellige iterationer, hvorigennem projektet er udviklet, samt erfaringerne med disse.
- Metoder
 - Dette afsnit beskriver de forskellige arbejdsmetoder, der er anvendt i udarbejdelsen af det endelige produkt.
- Specifikation- og analysearbejdet
 - Analysearbejdet, der ligger til grund for opbygningen af projektet, præsenteres i dette afsnit. Til dette hører kravspecifikation, domænemodel osv.
- Designprocessen

- Her beskrives selve designprocessen og erfaringerne med denne. Hertil hører lagdeling, klassediagrammer osv.
- Udviklingsværktøjer
 - Dette afsnit giver en beskrivelse af de mest væsentlige udviklingsværktøjer, der er blevet brugt til udviklingen af projektet.
- Resultater
 - De mest væsentlige resultater præsenteres objektivt i dette afsnit.
- Diskussion af opnåede resultater
 - I dette afsnit diskuteres der på baggrund af de opnåede resultater.
- Opnåede erfaringer
 - Dette afsnit giver en beskrivelse af de opnåede erfaringer gruppen har gjort sig, primært på baggrund af selve arbejdsprocessen.
- Projektets fortræffeligheder
 - Her præsenteres de dele af projektet, som er blevet fundet særdeles velfungerende.
- Forslag til forbedringer af projektet eller produktet
 - Afsnittet giver en beskrivelse af, hvad der kunne have været forbedret i selve produktet eller produktudviklingen.
- Konklusion
 - Der laves her en konklusion på baggrund af de opnåede resultater og diskussionen af disse, samt på baggrund af de opnåede erfaringer.
- Referencer
 - Afsnittet lister de forskellige materialer og værker, der refereres til.

4 Projektafgrænsning

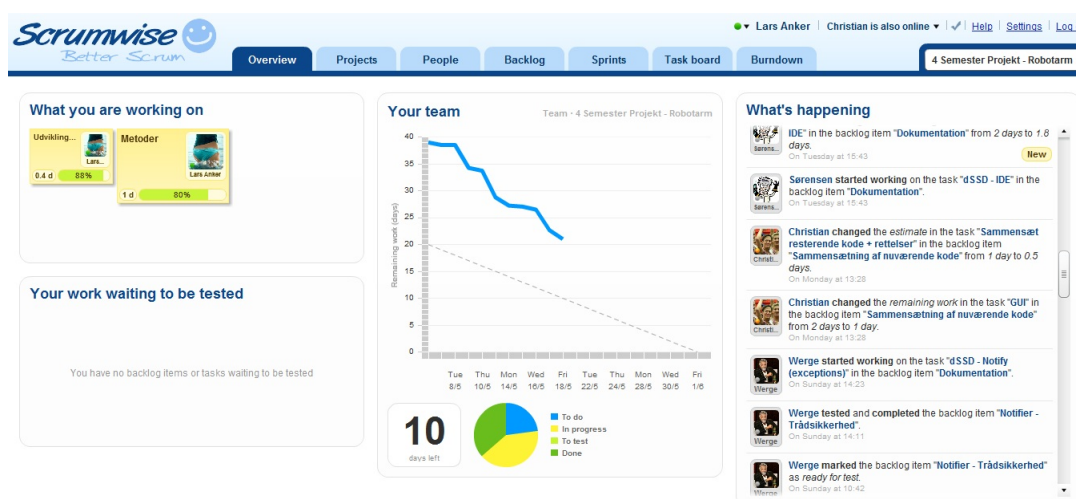
Da projektet har været udviklet selvstændigt, uden et projektoplæg eller noget fastsatte krav fra en kunde, har der ikke været mange afgrænsninger. En afgrænsning der hurtig opstod, var muligheden for at få GPS-koodinator for busserne, fra midttrafik. Dette gjorde det endnu vigtigere, at hurtigt få lavet en bus simulator der kunne bruges som alternativ.

5 Udviklingsværktøjer

Som nævnt har den primære udviklingsprocess til dette projekt været *Scrum*.

Under hele forløbet er værktøjet *Scrum Wise*⁵ blevet brugt til at håndtere projektbackloggen og sprintbackloggen. Værktøjet er online-baseret, og det er meget nemt at gå til. Der er mulighed for at oprette, redigere, afslutte og sætte statussen på de forskellige opgaver, via det indbyggede taskboard. Derudover er der en burndown-chart, der gør det nemt at overskue, hvordan det går med det aktuelle sprint.

Dette værktøj har gjort det nemt at benytte *Scrum* som udviklingsprocess og virkelig givet den gennemsikuelighed i projektetprocessen, hvilket er en af grundpillerne i *Scrum*. Et lille eksempel på *Scrum Wise*, kan ses på nedenstående billede(Ligger i stor størrelse i bilaget under Billeder/Scrum):



Figur 1: Scrum Wise Overblik

⁵<http://www.scrumwise.com/> - for nærmere indblik i Scrum Wise

Til versionsstyring og revisionskontrol er der brugt *SVN*. Programmerne *SmartSVN* / *TortoiseSVN* er benyttet til at styre den overordnede filhåndtering. Derudover er *AnkhSVN* (et plugin til *Microsoft Visual Studio*) blevet benyttet, hvilket sikrer en nem styring af solution-filer. Dette har givet en flydende versionsstyring uden store problemer.

Til test af koden er *NUnit* blevet benyttet, sammen med frameworket *Rhino Mocks*, der var en hjælp til testskrivning. For at overskueliggøre hvor meget af koden der var testet, er *DotCover* blevet brugt sammen med de ovenstående programmer.

Derudover har hjælpepluginet *ReSharper* ligeledes hjulpet med syntax, opbygning af programmer og rettelser af navne på attributter, variabler osv.

Alt tekstredigering er foregået i *LaTeX*, hvilket har været meget brugbart, da større dokumenter kunne opdeles i forskellige filer, som kunne arbejdes på individuelt. Dette har tilladt, at flere gruppemedlemmer kunne redigere samme dokument uden der opstod konflikter. Desuden sikres, at de nyeste billeder og lignende altid er opdateret, da *LaTeX* linker til disse i stedet for at indsætte dem direkte ligesom i fx word.

Brugen af *LaTeX* har dog givet en del ekstra arbejde, da det var et nyt program for de fleste gruppemedlemmer. I sidste ende er den tabte tid dog blevet genvundet, da samlingen har fungeret smertefrit, og den grundlæggende opsætning ikke er blevet ødelagt i takt med, at der er tilføjet mere tekst.

Til brug til at lave diagrammer er programmet *Visual Paradigm For UML*, samt *Star UML* blevet benyttet. Disse programmer er fundet rigtig gode og nemme at arbejde med og dermed blevet de primære valg som diagramværktøjer frem for for eksempel *Microsoft Visio*.

Microsoft Visio er dog blevet benyttet, da det har været glimrende til relationelle skemaer. Derudover er programmet *DIA* blevet brugt til udformning af ER-diagrammer.

6 Resultater

Nedenfor er de mest væsentlige resultater listet, hvorefter de kort beskrives. Der refereres til accepttestspecifikationen, hvis det ønskes at få et større overblik over de udførte tests.

Overordnede resultater

- Få en præcis tid til bus ankommer til stoppested.
- En måde at administrere busruter.
- relationel database, til persistering af data.
- Web service. til database adgang for mobil applikationen.
- Simulator af bus.

Mobil applikation

Mobil applikationen kan vise busruter, med sens stoppesteder og busser der er på ruten. samt det er muligt at få vist tiden til den næste bus ankommer ved et valgt stoppested.

Administrations værktøj

Det kan lade sig gøre, at administrer busruter, dette indebærer at lave, slette og ændre i ruter, stoppesteder og busser, samt ændre hvilken rute de forskellige busser køre op.

Database

Databasen er skabt med MySQL og består af 8 tabeller til at holde alt data for systemet. Systemet gemmer en del data, herunder busruter, stoppesteder, busser og GPS-koordinater for de forskellige busser.

Web service

Web servicen bruges som mellemlid mellem mobil applikationen og MySQL databasen, samt fjerner meget af det tunge arbejde fra mobil applikationen og flytter det over på en server.

Simulator

Det er muligt at simulere en eller flere busser der køre på sin busrute.

7 Projektets fortræffeligheder

Nedenfor er en række funktionaliteter og arkitekturer i projektet, som gruppen er specielt stolt af, listet og beskrevet.

Mobil applikation

Da mobil applikationen var den vigtigste del af projektet, blev der sat stort fokus på at få bygget den op med godt design. Der blev også sat fokus på at få gjort applikationen brugervenlig og intuitiv for brugeren. Det blev opnåede ved at brugeren skal fortage så få klik som muligt for at få vist det information han ønsker, samt ved at gøre det tydeligt hvad de forskellige knapper bruges til.

WebService

Der blev hurtigt fastslået at der ikke skulle ske meget arbejde på mobil telefonen, dette blev løst ved at der blev udviklet en webserive. Dette har gjort det muligt at udføre tunge udregninger på en server i stedet for på telefonen. Desuden gør den det muligt at nemmere at kunne udvikle lignende applikationer til andre mobil styresystemet så som IOS og Windows Phone.

8 Forslag til forbedringer af projektet eller produktet

Under udarbejdelsen af et softwareprojekt er det næsten altid plads til forbedringer, både mht. til arbejdsprocessen og selve produktet. Det gælder også for dette projekt. Nedenfor er en række elementer i projektet, som kunne have været optimeret, listet og beskrevet.

GUI - Administrations hjemmeside

Brugergrænsefladen på administrations hjemmesiden kunne være bedre, vis mindsker vinduet på hjemmesiden vil de forskellige elementer ikke længere passe i deres respektive felter. Det kan nemt blive løst ved at lave et nyt .css stylesheet, der vil gøre hjemmesiden til en mere behagelig brugeroplevelse.

Server

Serveren der hoster både administrations hjemmeside, webservicen og MySQL databasen, er en billige, ikke særlig kraftig server og kan derfor, sandsynligvis, ikke håndtere en særlig stort antal samtidige brugere af mobil aoliaktionen. Dog er det nemt at flytte det hele over på en ny server, når det bliver nødvendigt

Administrator login

Det skal være nødvendigt at logge ind på administratorens hjemmesiden før den kan bruges, eller vil det være muligt for hvemsomhelst at tilgå administrations værktøjerne.