AARHUS UNIVERSITET

I4SWT-02 Software test

Handin 2

Gruppe: SWT E2018 9

ATM Del 1

Christoffer Kjellerup Jakobsen - 201610457 (au569759)

Simon Møller Hindkær - 201609970 (au570382)

Marius Møller - 201610460 (au566306)

Tobias Damm Henrichsen - 201611660 (au553636)

Link til Jenkins build job: http://ci3.ase.au.dk:8080/job/TeamSWT09AirTrafficMonitorBuildCoverage/

Link til GitHub repository: https://github.com/ChristofferKJ/I4SWT/tree/master/ATM

Indhold

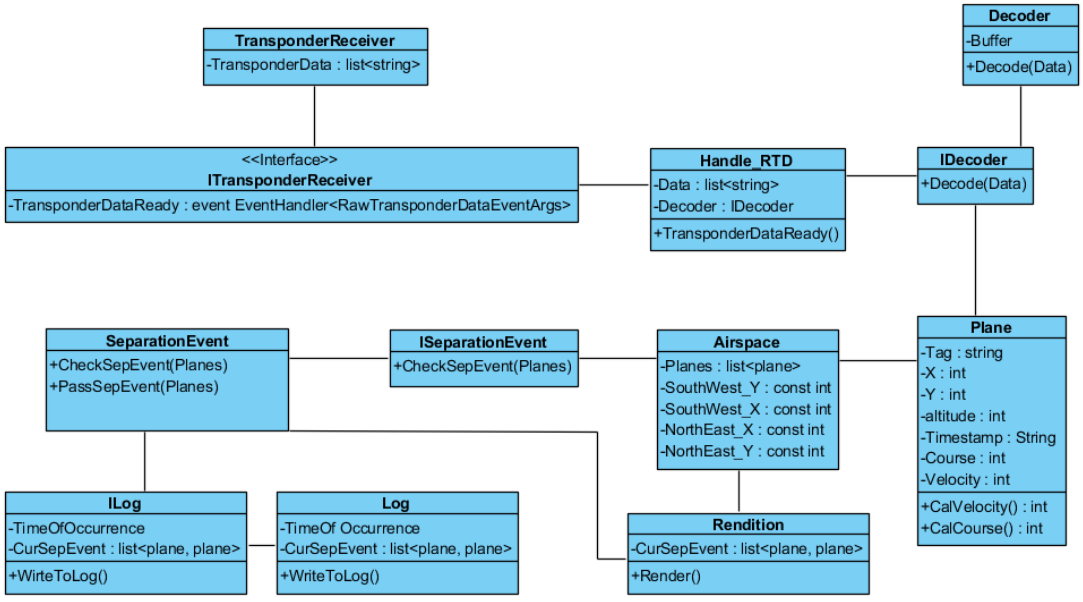
Air Traffic Monitor

Indledning

I forbindelse med Handin 2 i I4SWT, er der blevet udarbejdet en journal, samt en implementation, med tests, af en Air Traffic Monitor, som skulle overvåge og decode en række genererede flydata fra en ”Transponder Reciever”. Til opgaven er der blevet anvendt continuos integration, ved hjælp fra værktøjerne GitHub og Jenkins (Links på forsiden).

Design

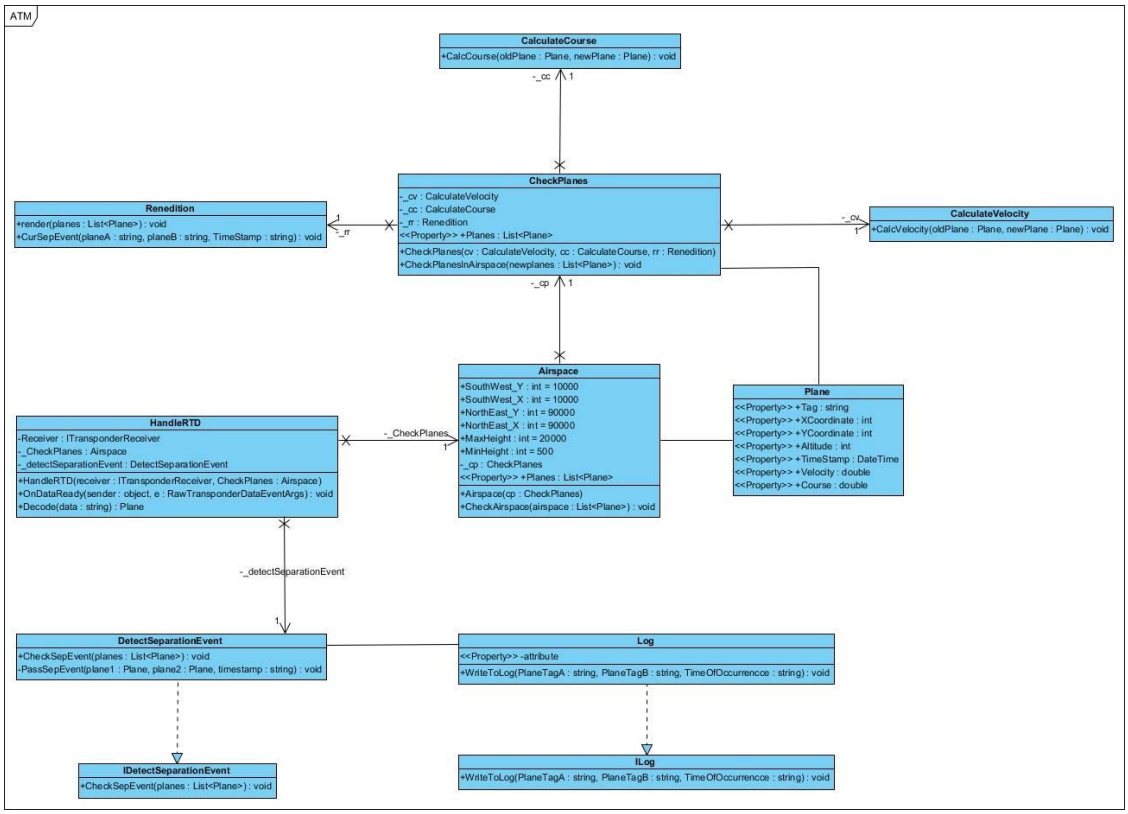
Da vi i gruppen modtog opgaven, var vores første tanke at få opdelt funktionaliteten for at kunne gøre vores design overskueligt og testbart. Hvorvidt vi opnåede dette er en anden diskussion, men vi fik lavet et system, som fungerer og kan kontrollere lufttrafikken i et givent område. Der blev altså i første omgang lavet et design som skulle gøre tingene mere overskuelige og testbare. Dette ses nedenunder.



Figur 1 Udkast til design

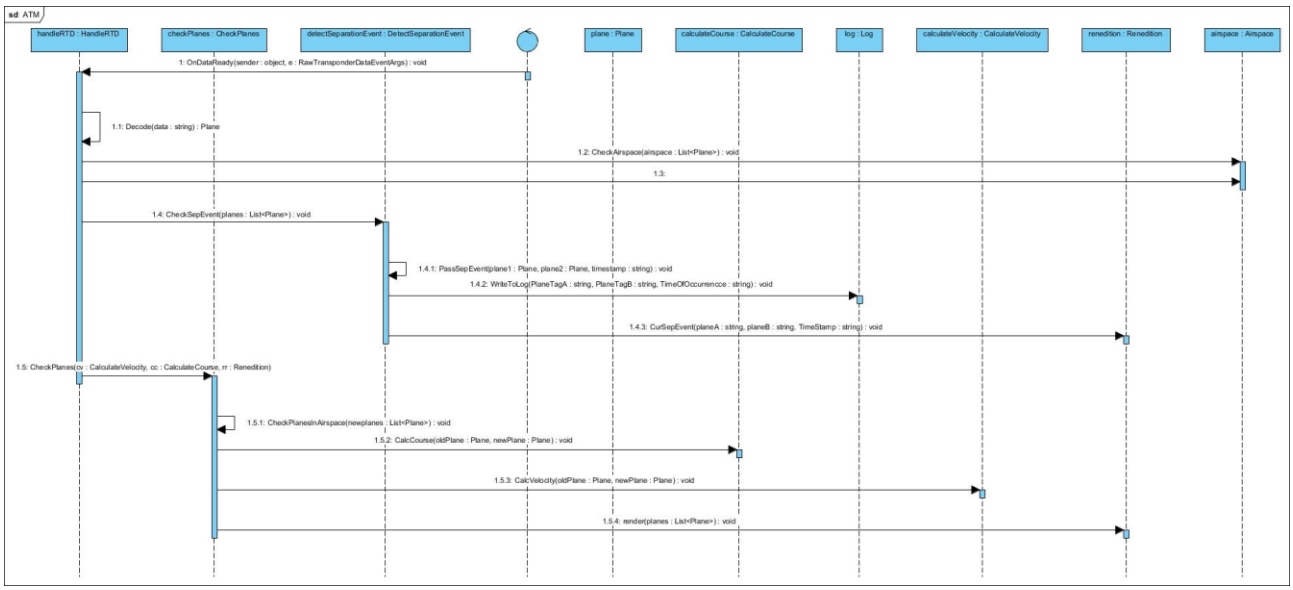
Vores design blev lavet med udgangspunkt i kravene til systemet og set i bagklogskabens lys skulle der nok have været mere fokus på testbarhed :-). Vi kunne under arbejdet med systemet konkludere at det første design ikke holdte helt, hvorfor der blev lavet et nyt klassediagram samt sekvensdiagram. Her ses vores løsning samt hvad der sker når vi modtager ”transponder” data.

Klassediagram



Figur 2 - Klassediagram for ATM (AirTrafficMonitor)

Sekvensdiagram



Figur 3 - Sekvensdiagram for ATM hvor vi modtager Transponderdata

Opdeling af implementering af klasser og test

I starten læste hele gruppen opgaven igennem og udarbejdede et klassediagram sammen. Herefter blev klasserne fordelt ud på gruppens medlemmer, og implementeringen gik i gang. Undervejs blev klasserne oprettet, rettet til og til sidst færdigudarbejdet af de medlemmer der var sat på de forskellige klasser. Som udgangspunkt, var det en person der stod for én eller flere bestemte klasser, så der ikke blev arbejdet i de samme filer, af flere forskellige medlemmer.

Ved implementeringen af vores tests, blev der nu byttet rundt, så vi lavede tests for hinandens klasser. På den måde, blev klasserne kigget igennem af et nyt sæt øjne.

Jenkins og GitHub (Continuos integration)

GitHub

Overordnet set, har GitHub hjulpet os positivt, idet vi er en gruppe på 4 mennesker, der typisk arbejder i den samme solution, på samme tid. Ved at uddele de forskellige klasser, har vi altså kunne arbejde på det samme projekt på samme tid, hvilket jo er en kæmpe fordel når man laver gruppearbejde. Idet vi har været gode til at ”pushe” vores filer op på vores repository, har det givet alle gruppens medlemmer et godt overblik over fremskridtet hos de andre personer, hvilket både er med til at indbyde til kritik og ros på det skrevne kode. Der opstod dog et problem, hvor to personer arbejdede på den samme fil, hvilket resulterede i at vi tabte noget kode. Her fulgte et større tidsspild, på at få genfundet den tabte kode, idet vi stadig er forholdsvist nye på GitHub.

Jenkins

Brugen af Jenkins, har i denne opgave hjulpet os med at få styr på om kvaliteten af de tests vi har lavet, har været tilstrækkelig. Der har dog været store problemer med at få selve jenkins siden op at stå, så den lavede coverage på de rigtige tests, hvilket har kostet meget tid for gruppen.

Alt i alt er CI noget vi mener burde bruges til ethvert gruppeprojekt, og der er rigtig mange positive aspekter, ved at anvende det. Dog har den største ulempe, været den anvendte tid der er blevet brugt på at fixe mindre problemer.

Kørsel af programmet

```
Plane tag: XOR568
Position: (34505,88495)
Altitude: 7600m
Speed: 262,895948534584m/s
Course: 92,2147551273192 degrees
Time: 04-10-2018 16:57:24

Plane tag: ASE275
Position: (10316,31337)
Altitude: 5800m
Speed: 250,399652328926m/s
Course: 158,938645538787 degrees
Time: 04-10-2018 16:57:24

Plane tag: XOR568
Position: (34511,88314)
Altitude: 7600m
Speed: 263,609054161428m/s
Course: 91,8986123706589 degrees
Time: 04-10-2018 16:57:25

Plane tag: ASE275
Position: (10477,31275)
Altitude: 5800m
Speed: 251,128617837889m/s
Course: 158,938645538787 degrees
Time: 04-10-2018 16:57:25
```

Figur 4 eksempel på kørsel