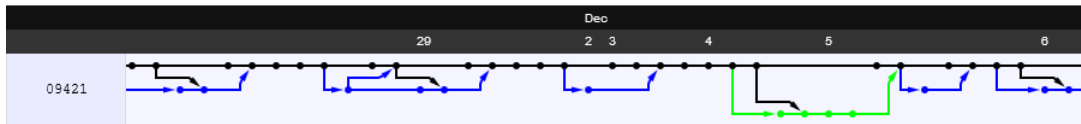


## 1 Udviklingsværktøjer

Under udførelsen af projektet er der blevet gjort brug af dele af Scrum, men da gruppens medlemsantal kun er to, blev det ikke set som nødvendigt at tage et scrum værktøj i brug. Der har været konstant kommunikation mellem gruppemedlemmerne, da der udelukkende er blevet arbejdet sammen. Under skriveprocessen af kravspecifikationen, accepttesten, system arkitekturen og dette dokument, er samtlige underafsnit blevet skrevet på et scrum-board, hvorpå samtlige gruppemedlemmer kan melde sig på en opgave.

Under udviklingen af systemet, blev der gjort brug af ansvarsområder, hvori der var klare linjer for, hvem der skulle udvikle hvilke dele.

Til versionsstyring er GitHub taget i brug, som er en versioncontroller, der implementerer Git. Dette er et standard versionsstyrings system, hvori medlemmer kan tilføje, fjerne eller ændre filer, hvorefter andre medlemmer kan tilgå ændringerne. Når et system af denne størrelse implementeres, genereres der ofte en del unødvendige filer. Git implementerer muligheden for, at kunne ignorere disse filer igennem en ignore protocol. Dette har specielt været nyttigt under udviklingen af mobilapplikationen, samt ved skrivningen af projektes dokumenter, da begge dele auto-genererer en del unødvendige filer, ved bygning. GitHub, og dermed versionstyrings redskaberne, tilgås igennem et commandline værktøj, kaldet Git Shell, som bygger ovenpå Windows Powershell. Heri kan der navigeres til den mappe, hvor repositoret er oprettet og herfra addes, committes, deletes og merges, præcis som et normalt versionsstyrings værktøj. Repositoret kan også tilgås online, hvori versionstyrings historikken kan tilgås. Dette har været nyttigt i den sammenhæng, at et tidligere commit nemt kunne hentes ned, hvis to versioner skulle holdes op mod hinanden. Commit og merge træet kan også følges. På figur ??, kan en del af dette træ ses. Den sorte linje repræsenterer masteren, eller trunken. De blå og grønne linjer repræsenterer en lokal branch, hvori mere end et commit er blevet tilføjet. Når den sorte linje peger ned på en af de blå eller grønne, betyder det at et merge er blevet udført. Prikkeren på de forskellige linjer repræsenterer et commit.



Figur 1: Git commit historik

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 begge 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.

Alle diagrammer er blevet udviklet i Microsoft Visio 2013, da dette havde indbygget værktøjer til alle de diagrammer, der blev set som nødvendige. Den eneste undtagelse er Use Case diagrammet, som blev udviklet på hjemmesiden [www.creately.com](http://www.creately.com). Visio understøtter "stencils", hvori forskellige komponenter relevant for det givne diagram, kan tilføjes. Dette gjorde det yderst nemt at arbejde med.