## CI Server

Gennem udviklingen af projekter kan man komme ud for, at de automatiserede tests kan tage lang tid at køre igennem. Derfor kan man bruge en CI server til at køre testene, så der kan arbejdes videre, imens testene kører igennem. I større projekter der udvikles på, bliver testene kørt igennem flere gange om dagen, og derfor ville programmøren bruge lang tid på at sidde og vente på, at testene er færdige, før han kan arbejde videre.

Ingeniørhøjskolen har stillet en Jenkins CI server til rådighed, som er blevet benyttet gennem udviklingen af Pristjek220. Jenkins er et rigtig godt værktøj, da der spares tid på tests som forklaret før. Dog er det ikke blevet benyttet så meget til udviklingen af Pristjek220, da det ikke er et stort system, og der fås svar på testene inden for få sekunder, når det køres på computeren.



Figur 1: CI server Test/build graf.

Figur 1 viser antallet af tests pr. byg fra CI serveren for Pristjek220, hvor det blå er succesfulde tests, og de røde er tests, der er fejlet, CI serveren er indstillet til hvert 15 minut til at tjekke om der er lavet en ændring i projektet, hvis der er bygger den projektet og køre alle tests igennem. Gennem brugen af CI serveren fandt gruppen ud af, at Jenkins havde nogle uoverensstemmelser med dansk/engelsk. Nogle tests virkede fint på computeren, mens CI serveren fejlede på testene, da den ikke kunne punktum og komma for priserne for produkterne. Som det kan ses ud fra Figur 1, gik der noget tid, før dette blev opdaget. Dette skyldes, at testene var hurtigere at køre på computeren, og ikke fejlede der. Da dette så blev opdaget, ændrede gruppen den specifikke kultur for decimaltal, så det stemte overens både på computerne og CI serveren.