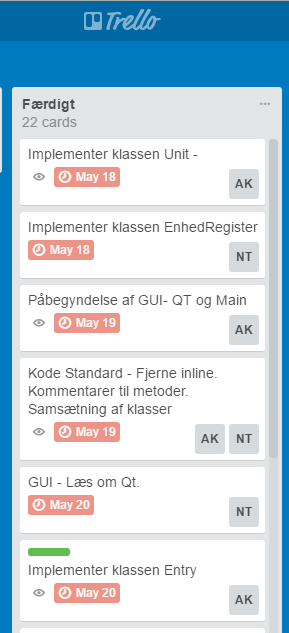
## Software design PC (AK og NT)

Design og implementering af PC softwaren blev foretaget ud fra systembeskrivelse, kravspecifikation og systemarkitektur. Formålet for softwaren på PC’en blev i systembeskrivelse, beskrevet som brugerens interaktion med Home Automation. Via brugergrænsefladen kunne brugeren konfigurerer hele systemet, og bestemme hvornår systemet skulle være aktivt.

Med kravspecifikation og systemarkitektur, kunne implementering påbegyndes. Fra systemarkitekturen blev brugerens interaktion kortlagt, som kunne ses i sekvensdiagrammerne i projektdokumentation. Da vi havde opstillet et klassediagram, hvor der var tilføjet metoder, så var nogle af metoder allerede kortlagt inden implementering var påbegyndt.

For at kunne kommunikerer med Styreboksen, blev der aftalt en protokol som skulle overholdes.

Hver dag under implementeringsfasen, blev der afholdt et møde hvor dagens sprints blev planlagt. De vigtigste opgaver blev priotereret, samt hvor længe de enkle forventede at tage. Da vi aftalte sprints hver eneste dag, så blev opgaverne forventet senest færdig til dagen efter. Alt planlægning blev foretaget via hjemmesiden Trello vist på Figur 1.

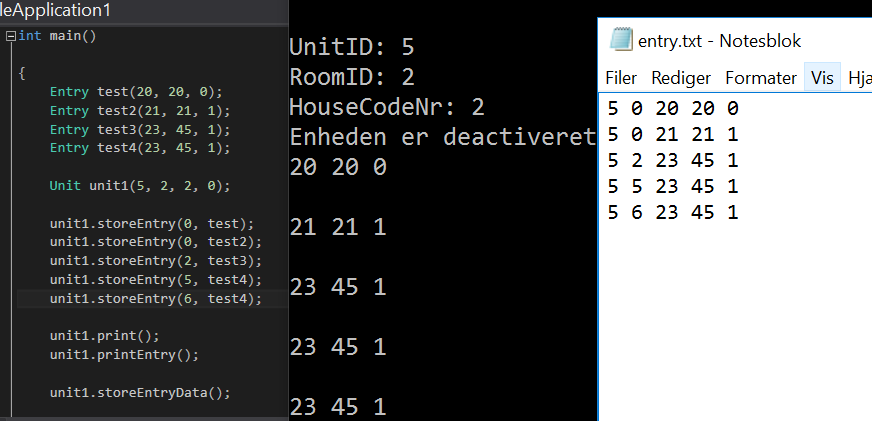


Figur - Billede af færdiggjorte opgaver

Klassediagrammer blev løbende opdateret under implementering af softwaren for PC. Da vi foretog løbende test af klasser og metoder, så blev metoder som at printe enheder ud implementeret, selvom vi på forhånd havde en ide om at disse ikke ville blive benyttet i det endelig program.

De første klasser UnitRegister, Units og Entry blev implementeret i Microsoft Visual Studio. Den grafiske brugerflade er blevet implementeret i QT creator.

En af de løbende test for overstående klasser, ses på Figur 2



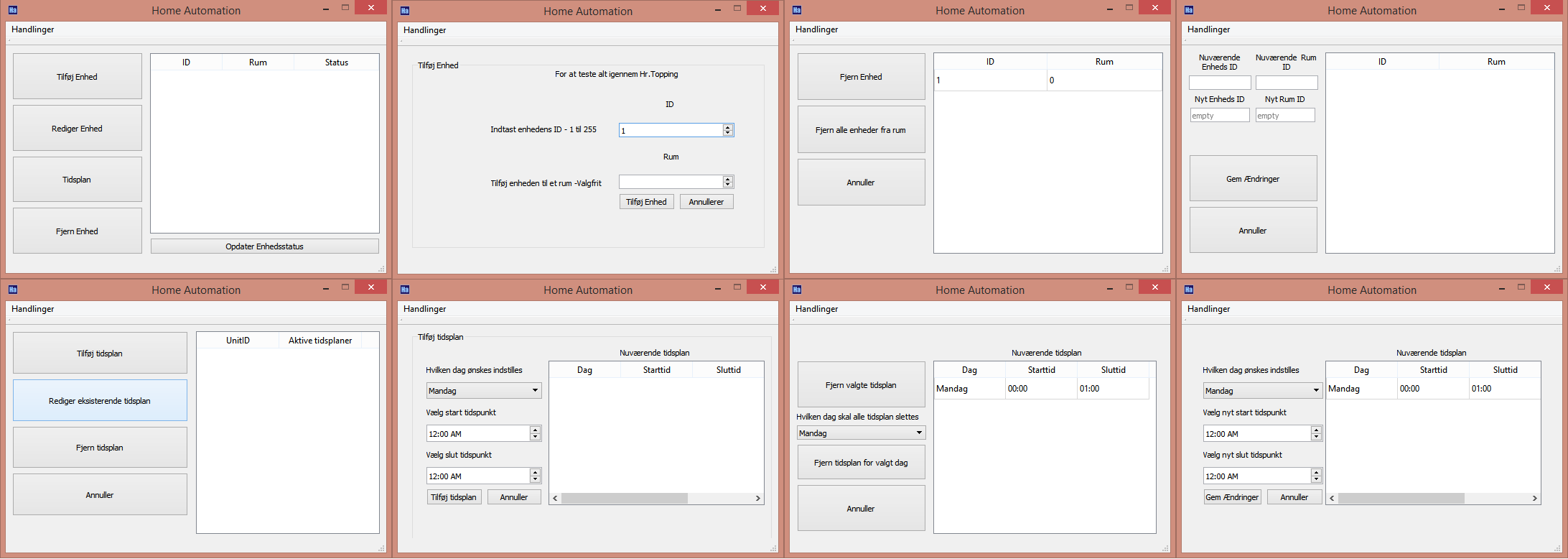
Figur - Test af Entry og Unit

På Figur 2 ses testen. Her ses at implementering tester Entry og Unit oprettelse. Da styreboksen blev implementeret af en gruppe, så foretog vi test hvor vi erstattede det modul med en tekst fil. Erstatning med en tekst fil var for at simulerer at der blev skrevet og læst til et ekstern modul. De løbende test blev benyttet for at kommunikerer om en opgave var færdig.

### Brugergrænseflade i Qt Creator

Brugergrænsefladen blev oprettet i Qt Creator. Hverken AK eller NT havde tidligere erfaring med programmet, hvilket gav en langsom start. Det overordnede formål med brugergrænseflade var at skabe en let måde for brugeren at konfigurerer Home Automation, og samtidig blive benyttet som systemets main.

Brugergrænsefladen er oprettet ud fra use case 1-6 samt use case 8-9, og tog udgangspunkt i sekvensdiagrammer, som kan ses i projektdokumentation. Under selve implementering blev ændringer som forhøjede brugervenlighed, hvilket betød opdatering af sekvensdiagrammerne.



Figur 3 – Hele brugergrænsefladen

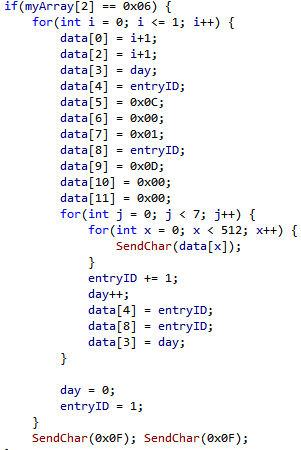
Et problem der opstod under fremstilling af software, var behovet for at begrænse brugerens mulighed for input. Dette skyldes at systemet maks skal kunne indeholde 256 enheder per rum, og ID 0 er reserveret til styreboksen. Et eksempel på hvordan dette løses vises på Figur 4, hvor et indbygget objekt fra QT, QintValidator benyttes til at begrænse tekst input i boksene. Dette begrænser input til tal fra 1 til 255.



Figur QIntValidator for EditUnit

Efter at PC softwaren var færdig implementeret, blev det testet ud fra accepttesten. Accepttesten er blevet udarbejdet med udgangspunkt på use casene, og indeholder præcise test som softwaren skulle overholde. Testen for PC softwaren blev udført med kodelås og en erstatning for styreboksen, da denne ikke var færdig implementeret. Største delen af PC softwaren testen blev udført med forventede resultater, hvor de fejlede resultater hovedsageligt bestod i at accepttesten ikke tog højde for foregående test resultater.

Under accepttesten fremstillede NT et stykke testsoftware der kunne emulere styreboksen. Denne software blev brugt udførelse af Use Case 1, hvori pinkoden fra kodelåsen valideres, og et par dummy enheder sendes til computeren. Koden vist i Figur 4 læser kommando fra PC og sender to fiktive enheder. Disse enheder benyttes til accepttesten. Data er et array med 512 pladser, der simulerer de data blokke der findes på SD-kort Modulet.



Figur Kodeudsnit til afsendelse af dummy enheder.