Teknikker og værktøjer i fokusområdet Test

## Teknikker:

Det er vigtigt at have kundens krav for øjet, når man skal teste ens system, så man kan sikre, at systemet opfylder deres krav, og man ikke bruger tid på at teste systemet for ting, som ikke er relevante for dem. Derved kan vi også sikre sporbarhed mellem kundens krav og vores tests af systemet.

**Unit testing:** Formålet med unit testing er at sikre, at en kodeblok fungerer som ønsket og forventet. Dette gøres ved hjælp af white box testing.

**Integration testing:** Formålet med integration testing er at sikre, at moduler spiller sammen som ønsket og forventet. Til dette kan man bruge både white box og black box teknikker.

**Code Review:** Formålet med code review er at finde fejl eller mangler i vores kode ved at lade en anden kigge på koden med friske øjne.

Mange test teknikker kræver, at man kan bruge ens kunder og/eller brugere til at hjælpe med testen, og derfor er de ikke mulige for os at bruge.

## Værktøjer:

Med vores teknikker valgt inden for fokusområdet er det vigtigt at forklare hvilke værktøjer vil bruge til at udføre teknikken. Til vores projekt har vi valgt at bruge **Visual Studio’s test suite**, til udførelse af unit testing og integration testing. Til code review har vi dog flere værktøjer, såsom **pull requests,** som kræver at en anden end der lavede requestet skal godkende det.

## Kvalitetskriterier:

En unit test skal kun teste én funktionalitet af en kodeblok i isolation.

En unit test er ikke en integration test og skal derfor ikke teste, hvordan kodeblokke taler sammen.

For at sikre, at to kodeblokke arbejder sammen som ønsket, skal der være en beskrivelse af, hvordan vi ønsker at kodeblokkene skal arbejde sammen.

Kodeblokkene skal være unit testet inden de bliver integration testet, så vi er sikre på, at vi kun tester, hvordan de taler sammen.

Både unit og integration tests skal være automatiserede, så de bliver kørt, hver gang vi laver en ændring i vores system.