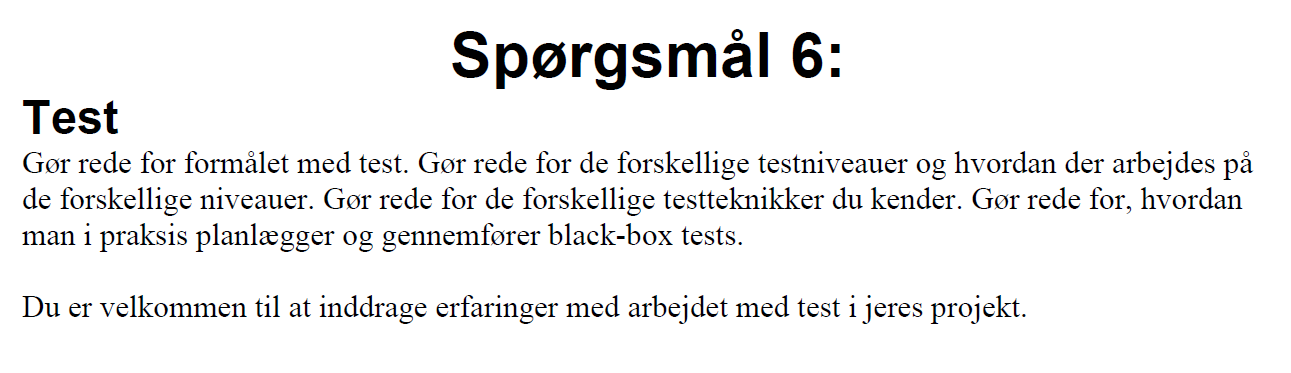
Spørgsmål 6



Formålet med test:

• Validering: laves det rigtige system – laves det system brugerne gerne vil have

• Verifikation: laves systemet rigtigt – opfører systemet sig som vi forventer

Generelt handler test om at

• **Planlægge** de nødvendige tests for hver fase/iteration, planlægge **hvilket system** eller hvilkedele af systemet der skal testes, planlægge **hvad** der skal testes, samt planlægge **hvornår** der skal testes

• **Designe og implementere** test ved at finde et passende og dækkende antal testdata ogudarbejde testcases på baggrund af disse

• **Udføre** de forskellige test, og systematisk håndtere resultatet af hver test, dvs. rapportere fejlpå en systematisk måde, så rette personer kan følge op

*Strategisk niveau:* mål og strategi

*Taktisk niveau:* planlægning

*Operationelle niveau:* udførelse

**Test niveauer**:

• Unittest: test af metoder, klasser (som helhed, dvs attributter, metoder, tilstande, osv), komponenter, osv, før de integreres med anden software

• Integrationstest: test af sammenhænge/samarbejde mellem komponenter og klasser

• Systemtest: test om systemet er komplet og korrekt, f.ex at use cases kan gennemføres og virker korrekt

• Brugertest: test af forskellige scenarier af use cases, om de fungerer tilfredsstillende og tilstrækkeligt brugervenligt for brugeren, afhængig af kravspecifikation

**Test teknikker** (kan bruges mere eller mindre uafhængig af situation, behov, og test niveau):

• Black-box: koden er ukendt, tager i stedet udgangspunkt i beskrivelser og modeller af systemet, komponent(er), metode(r), use case(s), eller andet

o Identificer gyldig/ugyldig værdiområde for input o Opdel de to værdiområder i *ækvivalensmængder* o Beskriv *grænseværdierne*

o Udarbejd testcases

• White-box: koden er kendt, strukturbaseret, dvs tager udgangspunkt i programmets interne struktur

o *Statementdækning*: hvis 100% statementdækning designes testcases så alle statements gennemløbes min én gang

o *Forgreningsdækning*: hvis 100% forgreningsdækning designes testcases så alle if/else, for- og while-statements identificeres og rammes

o *Stidækning*: hvis 100% stidækning designes testcases så alle mulige stier rundt i programmet identificeres og gennemløbes min én gang

o *Betingelsesdækning*: hvis 100% betingelsesdækning designes testcases så alle sammensatte betingelser afprøves, så alle mulige kombinationer af true/false er dækket ind

• Grey-box: en black-box test, hvor koden er kendt, hvilket nødvendigvis betyder at nogle ting fra white-box bliver blandet lidt i