

Exploratory Testing

VED BRUG AF MICROSOFT TEST MANAGER

Ian Mănebo Frost

ERHVERVSAKADEMIET SJÆLLAND, ROSKILDE | 4 SEMESTER EKSAMEN 2019

Indhold

Indledning.....	2
Motivation	2
Problemstilling.....	2
Metode	2
Planlægning	2
Hvad er Exploratory Testing?	3
Microsoft Test Manager	5
Microsoft Test Manager Struktur	5
Exploratory Testing funktion	6
Hvornår ville man vælge Exploratory Testing?.....	7
Konklusion	8
Refleksion	8
Litteraturliste	9

Indledning

Mange IT-systemer bliver testet vha. automatiseret testværktøjer. Vi har igennem vores studieforløb stiftet bekendtskab med bl.a. Selenium som tester GUI. Værktøjer som disse er yderst effektive da de er automatiseret hvilket resulterer i mindre udgifter og det er mindre tidskrævende. Men selvom automatiseret tests er populære, er de stadig ikke perfekte, de kan ikke skelne mellem de helt små detaljer, som kun det menneskelige har sans for. Jeg har valgt at sætte mit fokus på Exploratory Testing da det netop omhandler at teste systemet manuelt som bruger.

Motivation

Da jeg arbejdede med mit 2. semester projekt lavede vi hele tiden ændringer i programmet for at få det til at virke, fra de mindste bugs til de største bugs.

Vi fulgte Test Driven Development metoden, hvor vi havde lavet use cases og skulle derfra teste de enkelte metoder, hvilket viste sig som værende meget sløvslagt arbejde.

Jeg stiftede bekendtskab med Exploratory Testing gennem min vejleder Mohammed, og efter lidt research på nettet da så min første YouTube video, vidste jeg at det var et emne jeg gerne ville gøre mig i til eksamen.

Exploratory Testing adskiller sig fra hvad jeg har lært indtil videre under datamatiker uddannelsen. Som studerende ønsker jeg at få så bred en viden indenfor Testing da jeg ønsker at gøre en karriere indenfor området når jeg er færdig uddannelsen.

Problemstilling

- Hvad er Exploratory Testing?
- Hvordan udfører MTM Exploratory Testing og hvilke værktøjer kan man bruge?
- Hvornår ville man vælge Exploratory Testing?

Metode

Måden hvorpå jeg vil konkludere min problemstilling og i forlængelse heraf lave den bedst mulige synopsis. Har jeg valgt at min tilgang til arbejde skal forløbe sig således at jeg altid starter med min research. Min research består af læsning og forståelse af faglig relevant viden. Dette vil primært foregå gennem nøje udvalgte hjemmesider jeg føler kan underbygge lige netop det jeg har sat mig for at finde ud af. Dernæst vil jeg skifte mit fokus mod at kunne relatere min nyfundne viden til praksis. Jeg har valgt at benytte mig af programmet Microsoft Test Manager (MTM), for at stifte bekendtskab med et værktøj som giver mig en bedre forståelse af Exploratory Testing. Med de valg af hjemmesider jeg har truffet mig, vil jeg således også følge en vejledning for hvordan en virksomhed desuden bruger dette. Netop for at kunne perspektivere den viden jeg har fundet til et reelt scenarie for at understøtte min forståelse – dette vil også være en del af min mundtlige præsentation. Afslutningsvis i min arbejdsproces, lægger jeg al min koncentration i synopseskrivningen, Dokumentere min proces. Her bliver der lagt fokus på det mest relevante og vigtigste af hvad jeg har lavet research om. Dette gøres for at give læren og censoren den tilstrækkelig viden for at kunne forstå emnet samt forberede potentielle spørgsmål til eksamen.

For at udarbejde den bedst mulige synopsis har jeg valgt at benytte mig af følgende metoder:

Planlægning

Jeg har i dette afsnit sat mig for sinde at lave en visuel præsentation af mit planlagte arbejdsforløb. Dette ses nedenfor hvor jeg gennem projektforsløbet, løbende har skrevet synopsis i takt med min research samt

mit brug af Microsoft Test Manager. Dette er et bevidst valg, da jeg følte at dette ville resultere i den mest nøjagtige dokumentering af mit arbejde

MANDAG	TIRSDAG	ONSDAG	TORS DAG	FREDAG	LØRDAG	SØNDAG
26 Research om Exploratory testing	27 Research om Exploratory testing	28 Synopsis	29 Research	30 Synopsis	01 Arbejde	02 arbejde
03 Research om Microsoft Test Manager (MTM)	04 Udforskning af MTM	05 Synopsis	06 Udforskning af MTM	07	08	09 Synopsis
10 Synopsis	11 Research om MTM	12 Synopsis	13 Synopsis	14 Synopsis	15	16
17 Korrektur læsning af synopsis	18 Tilføje eller fjerne irrelevante indhold i synopsis	19 Synopsis	20 Aflevering af synopsis	21	22	23

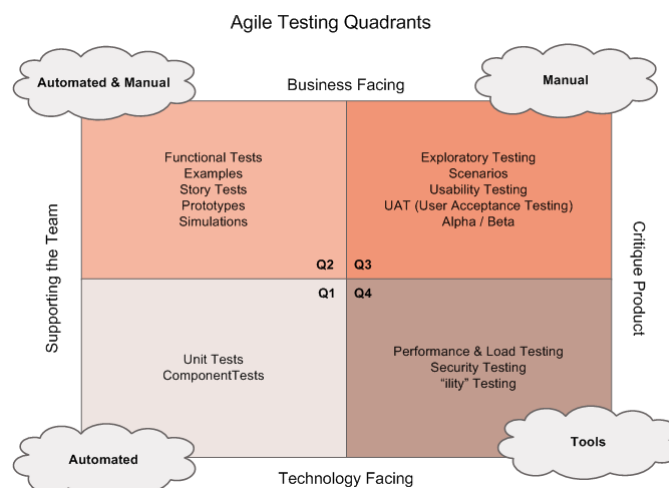
Hvad er Exploratory Testing?

Når man benytter sig af Exploratory Testing er fokusområdet lagt på selvlæring og frigørelse. Når man skal teste, skal der tænkes ud af boksen og der skal ikke følges testcases der er skrevet på forhånd, som desuden kun har ét enkelt formål. Når en tester følger Exploratory Testing benytter han sig af MTM, de finder ud af hvilke potentielle fejl og bugs der kan opstå, det er derfor vigtigt at man som tester benytter programmet som en klient ville gøre.

For at drage ligheder til vores undervisningsforløb i 4 semester, har jeg valgt at placere det i den Agile Testing Quadrant (ATQ). ATQ er en model der forsøger at kategorisere forskellige testmetoder samt opdele dem i forlængelse heraf. Dette er gjort for at skabe et langt mere overskueligt overblik samt, for at kunne finde de forskellige værdier den givne metode kan give til teamet og produktet.

Hvis man ser på Exploratory Testings placering i forhold til den Agile Testing Quadrant, er den placeret i Q3. Q3 er området, hvor alt forretningslogik starter, det er her man laver sine traditionelle user acceptance tests, det vil altså sige at Exploratory Testing ikke blot udføres af én aktør, men man tilhører et team som snakker sammen på daglig basis, hvor man skal dokumentere hvad der bliver testet, da det er med til at skabe feedback og gøre produktet bedre. Eftersom det er testeren selv, som skal gennemgå systemet, er det også oplagt at det bliver beskrevet som værende Manuel Testing.

Som tester, arbejder vi med specifikke områder på produktet og ikke bare helheden – da dette kunne gå hen og blive en alt for stor mundfuld for den enkelte tester at holde styr på. Et eksempel vi kunne benytte for videre at uddybe hvad dette helt præcist betyder. Kan vi forestille os at vi som tester skal teste en betalingservice for en given hjemmeside. Her skal vi tænke på mulige fejl, og vores tester prøvekører siden som en hver anden bruger ville gøre dette. Han indtaster sine personlige oplysninger og kommer ved en



fejl til at indtaste et ugyldigt postnummer – dette kunne være for langt, eller slet ikke eksistere i Danmark. Men alligevel bliver hans ordrer godtaget og færdiggjort. Dette er selvfølgelig ikke vores ønsket funktionalitet, derfor noteres denne fejl og rapporteres videre til resten af teamet. Altså er vi som sagt ikke fastlåste som ”generelle testere”, vi har hvert et fokusområde som vi har et delt ansvar for.

Exploratory Testing adskiller sig derfor meget fra Scripted Testing som jeg har arbejdet med under mit 2. semester projekt, her er der til formål kun at teste det som test cases beskriver.

Ifølge flere eksperter som benytter sig af Exploratory Testing følger proceduren fem stadier som er følgende: ¹

1. **Bug kategorisering** - Først kategoriseres bugs i forhold til hvor vigtige de kan være for programmet
2. **Test Charter** - Herefter bliver der skabt en Test Charter, som skitserer områder man skal teste indenfor
3. **Time box** -Testerne år sammen i parvis og arbejder intensivt op til 90 minutter, hvor man kører alle scenarier igennem. Dette kaldes for timebox
4. **Result review** - Her tjekkes der for hvilke resultater man har fået, og hvilken læring man har fået ud af programmet
5. **Debriefing** - Her briefes der i teams, om de fejl man har fundet, har stemt overens med de test charter man har skabt tidligere, samt om der er brug for yderligere tests.

Der er forskellige værktøjer som er tilgængelige for at benytte sig af Exploratory Testing. Jeg har givet to eksempler på værktøjer udover det jeg vil benytte mig af.

Tosca Tricentis - Tosca Tricentis benytter sig af samme funktioner som MTM, her kan der skabes test cases til automatiske test funktioner samt Exploratory Testing, her der også lagt fokus på at finde bugs og skabe konstant feedback for at få det bedste produkt Tosca Tricentis har til forskel for MTM at det er et abonnement man skal abonnere på, for at få adgang til deres produkter, hvor jeg gennem mit studie har adgang til alle Microsofts produkter i Imagine. Tricentis beskæftiger sig ikke kun med Exploratory Testing, deres produkter strækker sig bredt indenfor programmering, de har værktøjer fra web API til mobile programmering². Tricentis tilbyder ydermere også et træningsforløb med programmet, hvor man har til formål at blive specialist med Exploratory Testing, dette munder ud i en online eksamen hvor man kan få et bevis og certifikat, der kan vedhæftes på ens CV.

Azure Test and Feedback extension – Azure har udgivet et udvidelsesprogram til Azure DevOps, hvilket vil sige at man skal benytte sig af Azure DevOps for at kunne gøre brug af deres udvidelsesprogram. Det smarte ved dette er, du ikke skal have to programmer åbne for at se hvilke bugs man har fundet. Når man laver Exploratory Testing i dette program, optager den din skærm og alt hvad du klikker på, du kan endda indsætte screenshots alt efter behov, når dette er færdigt, bliver det direkte sendt over til Azure DevOps. Men eftersom jeg ikke har kendskab til Azure DevOps har jeg ikke valgt at benytte mig af deres produkter.

Jeg vil tage udgangspunkt i Microsoft Test Manager, eftersom vi igennem Datamatiker studiet har stiftet bekendtskaber med andre Microsoft produkter, herunder Visual Studio og Visio, derfor følte jeg det lå meget nærliggende. En anden faktor som gjorde sig gældende for mit valg, var informationen til rådighed på

¹ <https://www.techarcis.com/exploratory-testing-should-be-performed-when-why/>

² <https://www.tricentis.com/software-testing-tools/>

nettet om hvordan det skulle sættes op og hvor dybdegående informationen var.

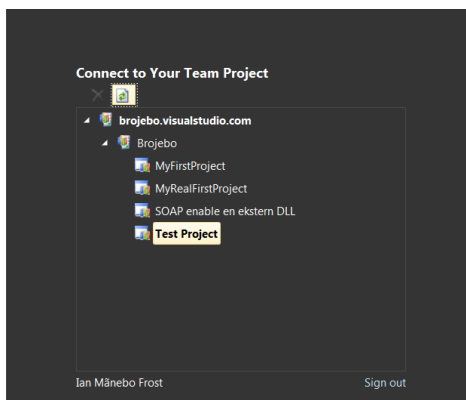
Jeg har derfor i løbet af min synopsis valgt at give en kort introduktion til hvad MTM er og hvordan en session³ udføres.

Microsoft Test Manager

Microsoft udbyder en masse programmerings produkter og da testing indgår i softwareudvikling har de udviklet deres eget program, Microsoft Test Manager, til firmaer som ønsker at teste i et agilt miljø.

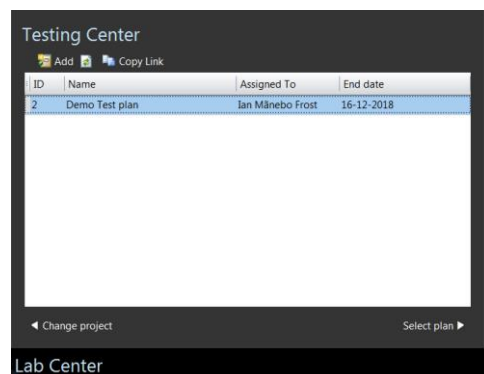
Microsoft Test Manager har primær til årsag at dokumentere den manuelle testers proces igennem systemet, ved hjælp af dets mange funktioner gør det arbejdet nemmere for testerens udviklingsteam at forstå hvor langt man er, samt hvilke opgaver der er fundet frem til.

Microsoft Test Manager Struktur



Microsoft Test Manager er forbundet til ens Visual Studio profil, der besidder offentliggjorte projekter. Første skridt er at vælge det specifikke projekt man ønsker at skabe sin tests til, som ses på ovenstående billede, har jeg skabt et helt nyt projekt kaldet "Test Project".

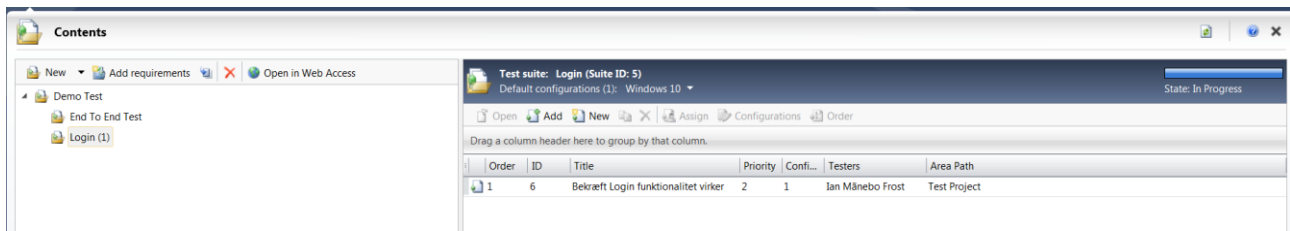
Efterfølgende skal der skabes en test plan. En test plan bliver delt op i tre kategorier: Et navn, hvilket område det kan dække og hvilken iteration den tilhører. Jeg har på nedenstående billede skabt et projekt der opfylder disse tre kriterier.



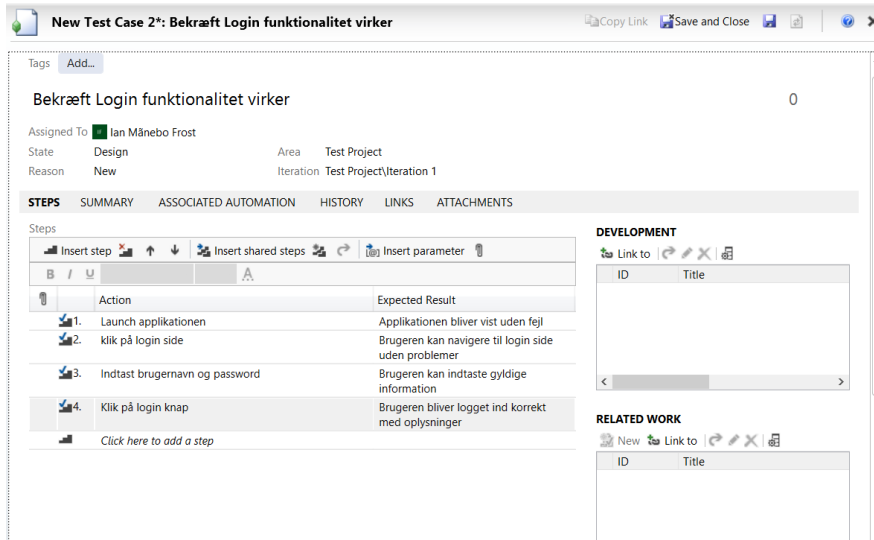
Test plan er stedet man specificer de områder ens test cases skal dækkes. Det er her test teamet kan se hvilke der skal laves. En test plan indeholder suites, som er det overordnede område hvori de forskellige test cases ligger, her er der også beskrevet prioritering af test case, samt hvor mange ændringer der er blevet lavet på test casen. Ved hjælp af statusprocesbaren i højre hjørne, kan man se om det er en aktiv suite (blå farve) eller en færdig suite (rød farve).

Her er det nemt og overskueligt for testeren at se indenfor hvilket område personen skal teste. Testeren kan enten tilføje bugs indenfor visse områder, eller blot udføre en af test cases.

³ En test case udførelse bliver betegnet som en session



En test case i MTM består af en beskrivelse og et forventet resultat som ses på nedenstående billede



Exploratory Testing funktion

Som tidligere beskrevet i synopsen, er frihed essentielt for at kunne benytte sig af Exploratory Testing. Frihed er en nødvendighed når man som tester skal anvende systemet som en bruger ville gøre.

Microsoft Test Manager indeholder en funktion specifikt henvendt til Exploratory Testing. Når man benytter sig af den funktionalitet, forsvinder ens store klodsede Test Manager program og ind i stedet optræder en lille applikation, hvor man dokumenterer ens forløb med funktionen.

Exploratory mode optager i baggrunden hvilke funktioner man benytter sig af, af den pågældende applikation, herefter kan der oprettes en bug, og ved hjælp af optagelserne opretter MTM selv steps ud fra hvilke handlinger man har gjort sig brug af.

Udover at kunne optage kan der også tages screenshots, for at gøre det visuelt nemmere for testeren at dokumentere fejlen.

Alt dette skal munde ud i en test case, der er autogeneret ud fra de optagelser fra Exploratory funktionen, og som er veldokumenteret for udvikleren, så han konkret ved hvilke fejl der skal kigges på og gøres ved.

På nedenstående billede ses Exploratory funktionen i efter den har optaget en session og ens steps er blevet autogeneraliseret

New bug

Tags [Add...](#)

Title:

STATUS

Assigned To:

State:

Reason:

Resolved Reason:

CLASSIFICATION

Area:

Iteration:

PLANNING

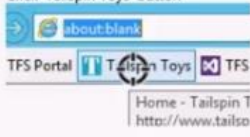
Stack Rank: Priority: Severity:

DETAILS **SYSTEM INFO** **TEST CASES** **ALL LINKS** **ATTACHMENTS**

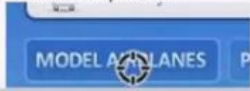
Steps to Reproduce:

The 6 steps you performed are listed below. To change which steps to include in the bug, click [Change steps](#)

1. Go to web page 'about:blank' using new browser instance
2. Click 'Tailspin Toys' button



3. Click 'Model Airplanes' link



History:

Type your comment here.

Hvornår ville man vælge Exploratory Testing?

Jeg har i sinde at undlade at svare på dette spørgsmål i min synopsis, da jeg ønsker at fremlægge min teori om, hvornår man benytter sig af Exploratory Testing samt hvilke fordele og ulemper det har, til eksamen.

Konklusion

Jeg har igennem dette forløb lært at Exploratory Testing handler om at tænke ud af boksen og tænke på de fejl en automatiseret test ikke kan gennemskue. Exploratory Testing bygger på at teste indenfor et givet område men ikke en specifik case, hvilket er meget essentielt at man har friheden indenfor et område. Exploratory er derfor et manuelt værktøj, hvilket også bliver illustreret i vores ATQ, som viser et overblik over de forskellige værktøjer indenfor testing verdenen.

For at kunne udføre Exploratory Testing er der udgivet forskellige udviklingsprogrammer, jeg gjorde mig bekendt med Microsoft Test Manager da vi har arbejdet med deres produkter ud igennem uddannelsen. Forskellen mellem de andre værktøjer ligger i produktet er tilhørende af et andet programmeringsprodukt, og eftersom vi har arbejdet med Visual Studio igennem vores uddannelse, har jeg haft byggeservere på forhånd, dette gjorde det endnu nemmere for min beslutning om hvilket værktøj jeg skulle gøre mig i. MTM har vist sig værende et fremragende værktøj indenfor den agile test verden, det er nemt overskueligt at skabe test cases og tildele aktører på pågældende case. Ydermere indeholder MTM den specielle funktion som er specifik for Exploratory Testing, som gør at man kan teste systemet igennem som en bruger ville gøre det. Når man benytter sig af exploratory funktionen går man automatisk igennem de 5 faser, som kendetegner exploratory testing. Ved at gennemføre disse 5 faser har man udført Exploratory Testing, men det er op til teamet selv at afgøre hvornår man føler man har testet produktet nok igennem. Som sagt tidligere har jeg undladt at svare på den sidste problemstilling da jeg ønsker at inkludere det i min fremlæggelse til eksamen.

Refleksion

Dette afsnit har til formål at dokumentere de tanker jeg har skabt mig over hvad jeg kunne have gjort anderledes, samt om jeg har fået det udbytte som jeg havde forventet jeg ville få, under mit forløb med denne opgave.

Planlægning: Jeg startede forløbet op med at planlægge et skema som skulle diktere hvordan jeg skulle udnytte min tid bedst muligt. Jeg har efterfølgende fundet ud af at det ikke er en metode jeg vil følge i samme stil til fremtidige projekter, da jeg følte mig fastlagt og det hele skulle gå en smule hurtigere med dokumentering og forståelse for at kunne overholde tidsplanen. Dette fandt jeg hurtigt ud af ikke var ideelt og har derfor ændret i min tidsplan for at få den til at passe til mine behov.

Min ide med at dokumentere løbende mens jeg gjorde mig bekendt med MTM har været en positiv ting, da jeg føler at det har hjulpet på min dokumentering.

På trods af jeg ikke har fulgt planlægningen føler jeg det udbytte jeg har fået igennem forløbet har været tilfredsstillende, jeg er blevet klogere på hvad Exploratory Testing har af grundværdier og hvordan det kunne fungere i en virksomhed.

Automatisering: Havde min synopsis skulle indeholde noget yderligere ville jeg gerne gå i dybden med hvordan man vha. MTM kan lave automatiserede tests og hvordan de virker, da jeg føler at det er relevant når jeg allerede omtaler MTM.

Spørgeskemaer: Jeg kunne have henvendt mig eksperter og firmaer for at høre hvilke testmetoder de benytter sig af, og høre hvilke erfaringer de har gjort sig med Exploratory Testing.

Litteraturliste

Kilder:

// Denne kilde har jeg benyttet mig for at forstå de basale elementer indenfor Exploratory Testing
Forfatter ikke opgivet – Guru99 (År ikke opgivet) "**Exploratory Testing Tutorial: Process, Techniques & Examples**"

<https://www.guru99.com/exploratory-testing.html>

// Denne kilde har givet mig viden indenfor andre relevante værktøjer der kan benyttes til Exploratory Testing

Forfatter ikke opgivet – **Software Testing Help** (2018) "**17 Best Exploratory Testing Tools**"

<https://www.softwaretestinghelp.com/tools/top-17-exploratory-testing-tools/>

// Denne kilde har jeg benyttet mig af for at forstå ATQ og dens hovedområder.

Lisa Crispin (2011) "**Using the Agile Testing Quadrants**"

<https://lisacrispin.com/2011/11/08/using-the-agile-testing-quadrants/>

// Denne kilde har jeg benyttet for at få en større viden indenfor et andet værktøj man kan benytte til Exploratory Testing

Forfatter ikke opgivet - **Tricentis** (År ikke angivet) "**Exploratory TestingFact Sheet**"

<https://www.tricentis.com/resource-assets/exploratory-testing/>

// Denne hjemmeside er blevet benyttet til at forstå Tricentis produktet og hvad det kan tilbyde.

Forfatter ikke opgivet - **Tricentis** (År ikke angivet) "**Exploratory Testing for Jira**"

<https://marketplace.atlassian.com/apps/1216562/exploratory-testing-for-jira?hosting=server&tab=overview>

// Kilden har til formål at give mig den nødvendig viden for at kunne sige hvornår man ville gøre brug af Exploratory Testing.

Forfatter ikke opgivet – **TechArcis** (År ikke angivet) "**Exploratory Testing – Why and when do we require exploratory testing?**"

<https://www.techarcis.com/exploratory-testing-should-be-performed-when-why/>

Video Kilder:

// Denne video har jeg brugt som støttehjul til at stifte bekendskab med MTM.

OneTest Solution (2014) "**Microsoft Test Manager 2013 (MTM) - Configure and Create Test Plan in MTM**"

<https://www.youtube.com/watch?v=G7-gAaoGv0>

// Denne video har jeg benyttet mig for at blive bedre til at bruge programmet MTM.

OneTest Solution (2015) "**Microsoft Test Manager 2013 (MTM) - Creating Test Suites**"

https://www.youtube.com/watch?v=T5gV3C_AQ74

// Denne video har jeg benyttet mig af for at få den fulde gennemgang af MTM og forstå alle dens funktioner.

Microsoft (2013) "**Case Management with Microsoft Test Manager 2013**"

https://www.youtube.com/watch?v=O_FQJ5Qx4k