

INGENIØRHØJSKOLEN AARHUS UNIVERSITET

BACHELORPROJEKT

RAMBØLL TILSYN



Modul- & Integrationstest

PROJEKT NR. 17103

Navn	Au ID	Studienummer
Ao Li	AU512161	201407737
Morten Sand Knudsen	AU463338	201270955

VEJLEDER: LARS CHRISTIAN JENSEN

DATO: 19/12-2017

Indholdsfortegnelse

Kapitel 1	Unit Test	4
1.1	Validator	4
Kapitel 2	Manuelle Test	7
2.1	Login test	7
2.2	Opret bruger test	7
2.3	Opret projekt test	8
2.4	Registrering på PDF test	8
2.5	Diverse test	8
Ordforklaring		8
Litteratur		10

Læsevejledning

Denne rapport redegør for de tests, som er lavet til projektet Rambøll Tilsyn.

Modul- og Integrationsdokumentationen er inddelt i to sektioner:

I første sektion forklares der omkring de automatiserede test som er lavet i projektet.

I anden sektion kigges nærmere på de manulle test cases som ville skulle valideres i forbindelse med systemet.

I slutningen af rapporten findes ordforklaring og litteraturliste.

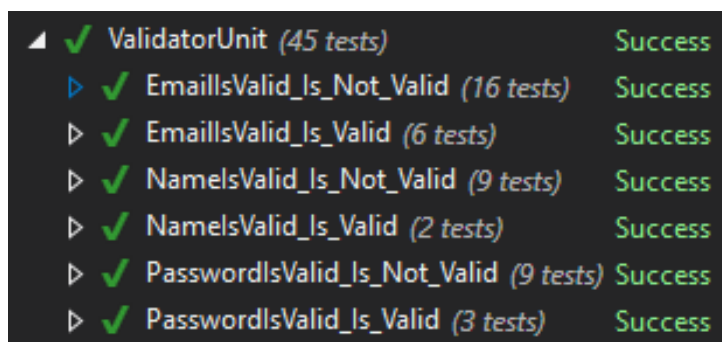
1 Unit Test

I følgende afsnit beskrives de unit tests, der bliver lavet. Der bruges Nunit[1], som er en testing framework til vores testing af klasser.

Diverse klasser vil blive testet i Testklassen af følgende syntaks "[Klasse]Unit". Metoderne skrevet i test klassen er skrevet med henblik på, at det skal være let læseligt, og være opsat med følgende syntaks: [Hvad der testes]_[I hvilke slags senarier]_[og den forventede resultat]

1.1 Validator

Validator-klassen er en statisk klasse, som har ansvar for at håndtere validering af input felterne i applikationen fra brugeren. For eksempel kan det være, når brugeren skal indtaste sin email og password. På Figur 1.1 ses følgende test der bliver udført.



Figur 1.1. Her ses et screenshot af test sessionen på ValidatorUnit

Der testes kun for Validator klassen fordi at det er den eneste klasse, der har noget algoritme. Da meget af koden, som der benyttes, ligger i forskellige frameworks, tages der højde for at disse er vel implementeret og gennemtestet, så derfor bliver der ikke udført yderligere tests.

På Figur 1.2 ses følgende test på e-mail, der bliver udført. Her ses hvilket format af e-mail vi tillader for at sikre klassen fungerer, som ønsket. Der bliver udført både positiv og negativ tests på e-mails for at tjekke, at både de ønskede og uønskede værdier bliver godkendt. Funktionen EmailIsValid tjekker, at e-mail opfylder blandt andet disse krav:

- E-mail må ikke have mellemsrumkarakter.
- Domæne navnet må ikke være tomt.
- Top level domain navnet[2] må ikke være tomt.
- Lokal navn må ikke være tomt.
- Lokal navn må ikke starte eller slutte med et punktum.

```
[TestCase("liao@live.dk")]
[TestCase("s@live.sccd")]
[TestCase("2314@live.dk")]
[TestCase("s@123.sd")]
[TestCase("s.d@123.sd")]
[TestCase("li@oo.dk")]
0 references | 0 changes | 0 authors, 0 changes
public void EmailIsValid_Is_Valid(string s)
{
    ... Assert.IsTrue(Ramboell.iOS.Validator.EmailIsValid(s));
}
[TestCase(".d@123.sd")]
[TestCase("s.@123.sd")]
[TestCase("s@live.s")]
[TestCase("s@s.123")]
[TestCase("s@s.shj")]
[TestCase("s@s.s")]
[TestCase("")]
[TestCase("s@s.s.")]
[TestCase("s@s")]
[TestCase("s@")]
[TestCase("sd")]
[TestCase("@")]
[TestCase("s.sd")]
[TestCase("s@123.sd ")]
[TestCase("s@123 .sd")]
[TestCase("s @123.sd")]
[TestCase("s@s.s")]
0 references | 0 changes | 0 authors, 0 changes
public void EmailIsValid_Is_Not_Valid(string s)
{
    ... Assert.IsFalse(Ramboell.iOS.Validator.EmailIsValid(s));
}
```

Figur 1.2. Her ses et screenshot af test sessionen på ValidatorUnit e-mail

På Figur 1.3 ses både positive og negative tests på funktionen PasswordIsValid. Der bliver givet et password, som funktionen vil godkende og et password, som ikke bliver godkendt. Der bliver også tjekket op mod at der bliver givet det forventede resultat. Funktionen PasswordIsValid tjekker at et password opfylder disse krav:

- Minimum 6 karakter lang.
- Minimum et stort bogstav.
- Minimum et lille bogstav.
- Minimum et tal.
- Speciel tegn er valgfrit.
- Må ikke indeholde mellemrum.



```
[TestCase("Aa2345")]
[TestCase("Abc123")]
[TestCase("Aa#345")]
0 references | 0 changes | 0 authors, 0 changes
public void PasswordIsValid_Is_Valid(string s)
{
    Assert.IsTrue(Ramboell.iOS.Validator.PasswordIsValid(s));
}
[TestCase("aa2345")]
[TestCase("123456")]
[TestCase("!\"'\"#%&/()=\"")]
[TestCase("")]
[TestCase("sdsdadsaf")]
[TestCase("12345")]
[TestCase("RTYVFUGHBNJK")]
[TestCase("DXFG")]
[TestCase("x&fbf83bgc")]
[TestCase("Aa#345dsavshw543I/T VIYI")]
0 references | 0 changes | 0 authors, 0 changes
public void PasswordIsValid_Is_Not_Valid(string s)
{
    Assert.IsFalse(Ramboell.iOS.Validator.PasswordIsValid(s));
}
```

Figur 1.3. Her ses et screenshot af test sessionen på ValidatorUnit på password

2 Manuelle Test

Dette kapitel vil indeholde en beskrivelse af de test cases, som er blevet kørt i forbindelse med testen af applikationen til Rambøll Tilsyn.

2.1 Login test

Denne sektion indeholder de manuelle tests for 'Login' viewet.

Pass/fail criteria:

Alle steps skal verificeres på Rambøll Tilsyn applikationen:

- Verificer at bruger kan logge ind med korrekte brugeroplysninger.
- Verificer at bruger ikke kan logge ind med forkert e-mail.
- Verificer at bruger ikke kan logge ind med forkert kodeord.

2.2 Opret bruger test

Denne sektion indeholder de manuelle tests for 'Opret bruger' viewet.

Pass/fail criteria:

Alle steps skal verificeres på Rambøll Tilsyn applikationen:

- Verificer at en bruger kan oprettes, hvis alle felter er udfyldte korrekt.
- Verificer at en bruger ikke kan oprettes, hvis e-mail feltet mangler at blive udfyldt.
- Verificer at en bruger ikke kan oprettes, hvis kodeord feltet mangler at blive udfyldt.
- Verificer at en bruger ikke kan oprettes, hvis fornavn feltet mangler at blive udfyldt.
- Verificer at en bruger ikke kan oprettes, hvis efternavn feltet mangler at blive udfyldt.
- Verificer at en bruger ikke kan oprettes, hvis telefonnummer feltet mangler at blive udfyldt.
- Verificer at en e-mail skal indeholde et @ for at blive godkendt.
- Verificer at kodeordet skal overholde kravene omkring minimum 6 cifre, et stort bogstav, et lille bogstav og et tal.

2.3 Opret projekt test

Denne sektion indeholder de manuelle tests for 'Opret projekt' viewet.

Pass/fail criteria:

Alle steps skal verificeres på Rambøll Tilsyn applikationen:

- Verificer at et projekt kan oprettes hvis alle felter er udfyldte korrekt.
- Verificer at et projekt ikke kan oprettes hvis projektnavn feltet mangler at blive udfyldt.
- Verificer at et projekt ikke kan oprettes hvis projektnummer feltet mangler at blive udfyldt.
- Verificer at et projekt ikke kan oprettes hvis projektadresse feltet mangler at blive udfyldt.
- Verificer at et projekt ikke kan oprettes hvis kundenavn feltet mangler at blive udfyldt.

2.4 Registrering på PDF test

Denne sektion indeholder de manuelle tests for 'Registrering på PDF' viewet.

Pass/fail criteria:

Alle steps skal verificeres på Rambøll Tilsyn applikationen:

- Verificer at der kan oprettes et fluebensobjekt.
- Verificer at der kan oprettes et minusobjekt.
- Verificer at der kan oprettes et cirkelobjekt.
- Verificer at kan navigeres en side tilbage i PDF'en ved at trykke på pil tilbage.
- Verificer at kan navigeres en side frem i PDF'en ved at trykke på pil frem.

2.5 Diverse test

Denne sektion indeholder de manuelle tests for grafisk udseende.

Pass/fail criteria:

Alle steps skal verificeres på Rambøll Tilsyn applikationen:

- Verificer at knapper har samme farve som baggrunden på Rambøll logoet.
- Verificer at tekst typerne på de forskellige tekster er ens.

Ordforklaring

Forkortelse	Forklaring
Fail criteria	Kravene for afvist test
Pass criteria	Kravene for godkendt test
Password	Kodeord
PDF	Portable Document Format

Litteratur

- [1] Unit testing framework for .net. Framework. URL <http://nunit.github.io/>.
- [2] Techopedia Inc. What is a top-level domain (tld)? - definition from techopedia. URL <https://www.techopedia.com/definition/1348/top-level-domain-tld>. Last Visited d. 15.12.2017.