

AARHUS UNIVERSITET

FORÅR 2021

Plug-n-Play - Test Specifikation

Gruppe 2021F12

Navn	Studienummer
Tri Nguyen	201610974
Niclas Spas	20106235

Vejledt af
Torben Gregersen

1. juni 2021

Indhold

0.1	Indledning	1
1	Modultest	2
1.1	Frontend	2
1.2	Backend	2
1.2.1	Backend Dependency Tree	2
1.2.2	Backend - Modultest	3
1.2.3	AuthController	3
1.2.4	Modul-test resultater	9
2	Integrationstest	10
2.1	Backend - Integrationstest	10
2.1.1	Registrering integrationstest resultater	11
2.1.2	Login integrationstest resultater	12
3	Accepttest	13
4	Litteratur	32

0.1 Indledning

Kapitel 1

Modultest

1.1 Frontend

Der er i frontenden ikke gjort brug af automatiserede test. Alt testing er i frontenden er foregået manuelt. Årsagen til dette er mangel på tid. Havde der været tid til opsætning af test ville der have været gjort brug af af test frameworket Jasmine til unit testing af services og komponent scripts. Der var ikke planer om at lave dybere testing, såsom testing af html bindings.

1.2 Backend

For blokkende til backenden er der sat modultest op for at kunne teste de givende moduler hvert for sig uafhængigt af hinanden, hvilket isolerer testen til kun at teste for det enkelte modul. Modultesten har af den opgave at kunne teste den interne opførsel for modulerne, og for at se om de returnerer det ønskede i forskellige scenarier med forskellige variabler som kører gennem systemet. Der gøres brug af NSubstitute til at lave fakes af andre afhængigheder, samt XUnit til at udføre modultestene.

1.2.1 Backend Dependency Tree

Når der skal planlægges tests kan et dependency tree være et rigtig godt værktøj til at give et overblik over hvilket moduler har brug for hvilke andre moduler. Det gør det lettere at planlægge en integrationstest senere hen af processen.

1.2.2 Backend - Modultest

Til backend-modultest er der gjort brug af automatiserede såvel som manuelle modultests. De automatiserede modul-tests er udført med XUnit og NSubstitute for at kunne isolere moduler og tests fra andre instanser. Modul-testene skrives og evalueres i den samme solution som API'et, hvor hver test dedikeres til hvert modul i isolation, og laver Fakes af afhængighederne gennem NSubstitute.

Modul-testene udføres alle i Arrange Act Assert pattern[1], som arrangerer alt data så det er klar til brug, udføres der metoder som skal testes, og derefter tester for om de ønskede udfald sker. De følgende klasser vil have modul-tests dedikeret til en metode samt et skærmbillede af Test-Explorer fra Visual Studio for at vise testens kørsel.

Til planlægningen af modultest er der gjort brug af Top-Down-Integration-Testing[2] som roden til modul-testenes rækkefølge af prioritet. Der er derfor startet ud med at skrive tests for de øverste lag i backendenden før der nedre lage eller "stubbe".

1.2.3 AuthController

AuthController-klassen står for at være den udadgående kontakt til omverdenen, her i form af en klient. AuthController har til den specifikke opgave at tage sig autentificering og autorisering. Der vil derfor testes for de relevante metoder som har med login, registrering, og ændringer af login-data. Til start defineres objekter som der kan bruges som input og/eller tjek-værdier, som de kommende tests kan gøre brug af hver for sig uden at skulle skrive nye objekter i hver test.

Listing 1.1: AuthControllerTests setup

```
1 public AuthControllerTests()
2 {
3     authService = Substitute.For<IAuthService>();
4     configuration = Substitute.For<IConfiguration>();
5
6     existingUser = new User()
7     {
8         Username = "IExist",
9         Password = "ArbitraryPassword",
10        Email = "ArbitraryEmail"
11    };
12
13    nonExistingUser = new User()
14    {
```

```
15     Username = "IDoNotExist",
16     Password = "ArbitraryPassword",
17     Email = "ArbitraryEmail"
18 };
19
20 correctUser = new User()
21 {
22     Username = "IExist",
23     Password = "CorrectPassword",
24     Email = "ArbitraryEmail"
25 };
26
27 nonExistentTokenUser = new Token("")
28 {
29     JsonWebToken = "noUser"
30 };
31
32 nonExistentTokenPassword = new Token("")
33 {
34     JsonWebToken = "noPassword"
35 };
36 }
```

PostRegister

PostRegister metoden tjekker op for, om den givne bruger eksisterer, hvis brugeren eksisterer vil den så returnere en fejl. Findes brugeren ikke kan den oprettes i systemet uden problemer. Disse to scenarier testes i modul-testene med det første scenarie på listing 1.2:

Listing 1.2: PostRegister brugernavn er allerede i brug

```
1 [Fact]
2 public async Task PostRegister_UsernameTaken()
3 {
4     // Arrange
5     authService.Register("IExist", "ArbitraryPassword", "
        ArbitraryEmail").Returns(false);
6     AuthController systemUnderTest = new AuthController(authService,
        configuration);
7
8     // Act
```

```
9     var actionResult = await systemUnderTest.PostRegister(  
        existingUser);  
10  
11     // Assert  
12     var result = Assert.IsType<ConflictObjectResult>(actionResult);  
13     Assert.Equal("Username taken", result.Value);  
14 }
```

Til registreringen af en bruger skal der gøres brug af et brugernavn, et kodeord, og en e-mail adresse. Med brugernavnet `IExist` er det indforstået, at den der prøver at registrere sig i systemet prøver at gøre brug af et allerede eksisterende brugernavn, der forventes så at `PostRegister` metoden returnerer en relevant fejl, her `"Username taken"`.

Listing 1.3: PostRegister brugernavn er ikke allerede i brug

```
1 [Fact]  
2 public async Task PostRegister_UserRegistered()  
3 {  
4     // Arrange  
5     authService.Register("IDoNotExist", "ArbitraryPassword", "  
        ArbitraryEmail").Returns(true);  
6     AuthController systemUnderTest = new AuthController(authService,  
        configuration);  
7  
8     // Act  
9     var actionResult = await systemUnderTest.PostRegister(  
        nonExistingUser);  
10  
11     // Assert  
12     var result = Assert.IsType<OkObjectResult>(actionResult);  
13     Assert.Equal("User registered", result.Value);  
14 }
```

PostLogin

`PostLogin` metoden tjekker op for, om den givne bruger eksistere, og om den givne kodeord er korrekt i forhold til brugeren. De kommende tests for metoden vil gennemgå de forskellige scenarier der kan teste opførslen af metodens kald uafhængigt af klassens overordnede afhængigheder.

Listing 1.4: PostLogin test for forkert brugernavn

```
1 [Fact]
2 public async Task PostLogin_UsernameNotRegistered()
3 {
4     // Arrange
5     authService.Login("IDoNotExist", "ArbitraryPassword").Returns(
6         nonExistentTokenUser);
7
8     AuthController systemUnderTest = new AuthController(authService,
9         configuration);
10
11     // Act
12     var actionResult = await systemUnderTest.PostLogin(
13         nonExistingUser);
14
15     // Assert
16     var result = Assert.IsType<NotFoundObjectResult>(actionResult);
17     Assert.Equal("User not found", result.Value);
18 }
```

Testen på listing 1.4 tester for en bruger som ikke eksisterer, der forventes derved at serveren svarer igen med at der er givet forkerte legitimationsoplysninger. Da der kun testes for det enkelte modul er AuthService klassen Faked og tvunget til at returnere en bestemt værdi, her kan der så testes på modulet for at se om det opfører sig som forventet ud fra input og svar det får fra bruger og afhængigheder.

Listing 1.5: PostLogin test for forkert kode

```
1 [Fact]
2 public async Task PostLogin_WrongPassword()
3 {
4     // Arrange
5     authService.Login("IExist", "ArbitraryPassword").Returns(
6         nonExistentTokenPassword);
7
8     AuthController systemUnderTest = new AuthController(authService,
9         configuration);
10
11     // Act
12     var actionResult = await systemUnderTest.PostLogin(existingUser);
13
14     // Assert
15     var result = Assert.IsType<ConflictObjectResult>(actionResult);
16     Assert.Equal("Wrong credentials", result.Value);
17 }
```

Ligeledes som på listing 1.4 bliver der i listing 1.5 testet for forkerte legitimationsoplysninger. I denne test testes der for, hvordan systemet opfører sig med en forkert kodeord, hvor det forventede resultat ville være at returnerer "Wrong credentials". Der er valgt ikke at skelne direkte mellem forkert kodeord eller ikke-eksisterende bruger, for sikkerhedens skyld hvis nogen skulle prøve at komme ind i en konto der ikke tilhøre dem.

Listing 1.6: PostLogin korrekte legitimationsoplysninger

```
1 [Fact]
2 public async Task PostLogin_LoggedIn()
3 {
4     // Arrange
5     authService.Login("IExist", "CorrectPassword").Returns(
6         correctTokenResponse);
7
8     AuthController systemUnderTest = new AuthController(authService,
9         configuration);
10
11    // Act
12    var actionResult = await systemUnderTest.PostLogin(correctUser);
13
14    // Assert
15    var result = Assert.IsType<OkObjectResult>(actionResult);
16    Assert.Equal(correctTokenResponse, result.Value);
17 }
```

Som sidste led i testen med login testes der i listing 1.6 at adfærden af, at have et korrekt brugernavn og korrekt kodeord gør. Metoden forventes derved, at der skal returneres et Token-objekt, hvis det lykkedes at logge ind.

ChangePassword

ChangePassword metoden beskriver sig selv rimelig godt, den bruges når en bruger gerne vil skifte sit kodeord, og det kan være af flere forskellige årsager. Da kodeordene ikke persisteres i databasen som plain-text skal der mere til for at kunne sørge for, at et kodeord kan tjekkes op og ændres.

Listing 1.7: ChangePassword forkerte legitimationsoplysninger

```
1 [Fact]
2 public async Task ChangePassword_Failed()
3 {
4     // Arrange
```



```
5     authService.PasswordUpdate("IExist", "ArbitraryPassword").
        ReturnsNull();
6     AuthController systemUnderTest = new AuthController(authService,
        configuration);
7
8     // Act
9     var actionResult = await systemUnderTest.ChangePassword(
        existingUser);
10
11    // Assert
12    var result = Assert.IsType<ConflictObjectResult>(actionResult);
13    Assert.Equal("Something went wrong", result.Value);
14 }
```

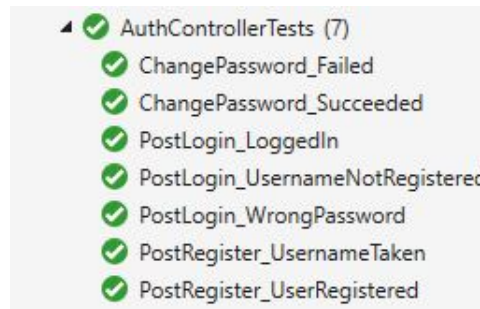
Listing 1.7 viser opførslen af metoden når en ikke eksisterende bruger prøver at ændre en kode. Da brugeren ikke eksisterer, vil der ikke komme noget svar fra AuthService-klassen, derved vil kodeordet ikke kunne blive opdateret. Der forventes så her, at der tilbagegives en fejl "Something went wrong". Det er højst usansynligt at dette ville blive et reelt problem, da den tænkte adgang til metoden sker fra en brugers egen profil, derfor bør brugeren eksistere når der kaldes på denne metode.

Listing 1.8: ChangePassword korrekte legitimationsoplysninger

```
1 [Fact]
2 public async Task ChangePassword_Succeeded()
3 {
4     // Arrange
5     User tempUser = new User();
6     authService.PasswordUpdate("IExist", "CorrectPassword").Returns(
        tempUser);
7     AuthController systemUnderTest = new AuthController(authService,
        configuration);
8
9     // Act
10    var actionResult = await systemUnderTest.ChangePassword(
        correctUser);
11
12    // Assert
13    var result = Assert.IsType<OkObjectResult>(actionResult);
14    Assert.Equal("Password has been changed", result.Value);
15 }
```

Der vises så på listing 1.8 en succesfuld kodeordsændring. Der forventes så her, at brugeren eksisterer og et opdateret User-objekt kan returneres. Da der modtages et User-objekt antages det at der er ændret på kodeordet, og der forventes en returbesked "Password has been changed" for at indikere den succesfulde kodeordsændring.

1.2.4 Modul-test resultater



Figur 1.1: Testresultater fra Test-Explorer for AuthControllers modul-test

De viste **Assert**-statements fra modul-testene viser, at de resultater som var forventede, blev modtaget korrekt.

Kapitel 2

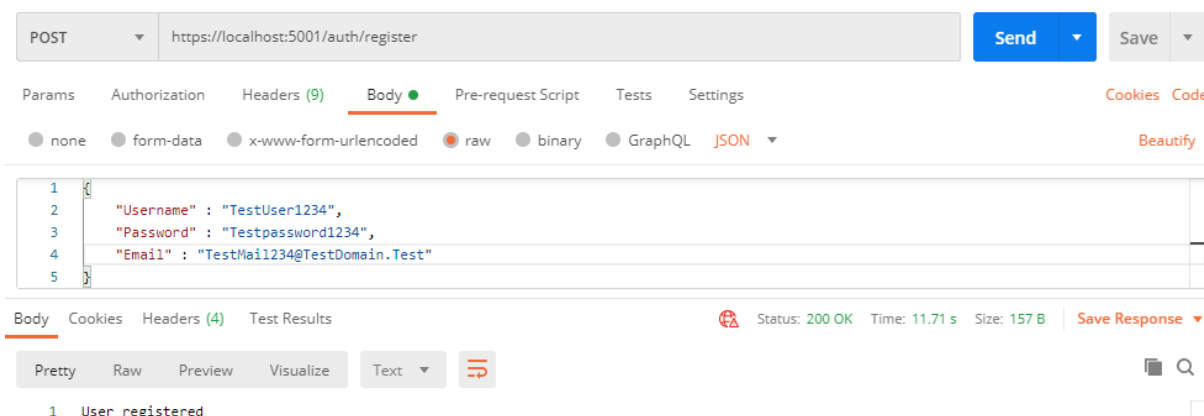
Integrationstest

2.1 Backend - Integrationstest

Integrationstest af backend er sket gennem manuelle tests. Der er gjort brug af PostMan til at teste API-endpoints for om de opfører sig som forventet. De manuelle test har klare forventninger, og tager sig af hele kæden fra server til database. Testene er udført lokalt, da testene skulle kunne gennemføres før det ville blive deployet på en internet server. Der bliver testet på 2 forskellige end-points for at kunne vurdere samarbejdet mellem de interne klasser, og serveren forbindelse til databasen. De to end-points der testes er:

- `https://localhost:5001/auth/register`
 - Registrerings-end-point, som tager sig af registrering af brugere til systemet. Der testes forbindelse fra controller, til service, til database. Der tjekkes ikke direkte i databasen, om data er persisteret, men der testes for hvorvidt en forventet returværdi modtages.
- `https://localhost:5001/auth/login`
 - Login-end-point, som tager sig af login for eksisterende brugere til systemet. Der testes forbindelse fra controller, til service, til database. Servicen vil efterspørge data fra databasen, sammenligne det med hvad den får fra klienten, og derved kunne returnere en forventet værdi gennem controlleren.

2.1.1 Registrering integrationstest resultater



Figur 2.1: Manuel test af brugerregistrering

Fig. 2.1 viser en API-test fra PostMan, hvor der kaldes til URL `https://localhost:5001/auth/register` med legitimationsoplysningerne som vist i listing 2.1. Serveren returnerer med responskoden `200 OK`, som indikerer at der succesfuldt er udført et kald til API'et. Endvidere ses der også responsen `"User registered"` som også indikerer at kaldet er fuldent.

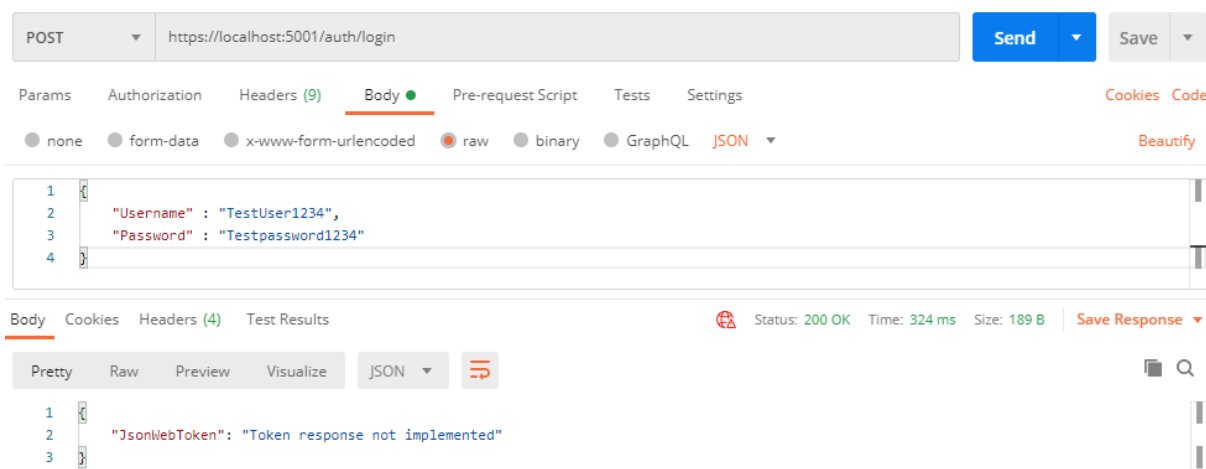
Listing 2.1: Legitimationsoplysninger for registrering af ny bruger

```

1  {
2      "Username" : "TestUser1234",
3      "Password" : "Testpassword1234",
4      "Email" : "TestMail1234@TestDomain.Test"
5  }
6  }

```

2.1.2 Login integrationstest resultater



Figur 2.2: Manuel test af brugerlogin

Listing 2.2: Legitimationsoplysninger for logging af eksisterende bruger

```
1  {
2      "Username" : "TestUser1234",
3      "Password" : "Testpassword1234"
4  }
5 }
```

Fig. 2.2 viser en API-test fra PostMan, hvor der kaldes til URL `https://localhost:5001/auth/login` med legitimationsoplysningerne som vist i listing 2.2. Serveren returnerer med responskoden 200 OK, som indikerer at der succesfuldt er udført et kald til API'et. Endvidere ses der også responsen på listing 2.3. Det returnerer et Token-objekt som består af en string som skulle indeholde en JWT, der skulle kunne autentificere en bruger, denne funktionalitet er dog ikke implementeret, og der returneres så derfor kun en string-response inde i objektet.

Listing 2.3: Token-objekt som respons for login

```
1  {
2      "JsonWebToken": "Token response not implemented"
3  }
4 }
```

Kapitel 3

Accepttest

Baseret på de funktionelle krav beskrevet i Krav Specifikationen, er der taget udgangspunkt i User Stories for at specificere en accepttest af funktioner. User Stories beskrev den overordnede gennemgang af scenarier meget løst, som nu bliver udspecificeret til accepttest. Accepttest

Accepttest User Story 1 - Gæst spiller spil			
Forudsætning: Platformen er åben i en browser på to computere			
Aktivitet	Forventet Resultat	Resultat	Vurdering
På den første computer: Tryk på "New Game"	En pop-op menu viser valgmuligheder for spil.	Der bliver dirigeret til køsiden.	Uacceptabelt
Tryk på "Nim"	Der bliver vist en pop-op menu som viser at man er sat i kø til "Nim".	Der vises kun, at man bliver sat i kø til 'Nim'	Delvist acceptabelt
På den anden computer: Tryk på "New Game"	En pop-op menu viser valgmuligheder for spil.	Der bliver dirigeret til køsiden.	Uacceptabelt
Tryk på "Nim"	Der bliver vist en pop-op menu som viser at man er sat i kø til "Nim". Der bliver startet et spil mod den første computer.	Der vises kun, at man bliver sat i kø til 'Nim'.	Acceptabelt / Uacceptabelt

Accepttest User Story 2 - Gæst benytter chat			
Forudsætning: Gæst har startet et spil & er matchet med sig selv i en anden browser på en anden computer.			
Aktivitet	Forventet Resultat	Resultat	Vurdering
Indtast "Hej, håber det bliver en god kamp" i beskedfeltet	Den indtastede tekst ses i beskedfeltet	Den indtastede tekst ses i beskedfeltet.	Acceptabelt
Tryk på "Send"	Den indtastede tekst ses i begge chatbokse i hver browser	Den indtastede tekst ses ikke i begge chatbokse i hver browser.	Uacceptabelt

Accepttest User Story 3 - Gæst registrer sig			
Forudsætning: Platformen er åben i browseren			
Aktivitet	Forventet Resultat	Resultat	Vurdering
Tryk på "Register"	Der navigeres til registreringssiden	Der navigeres som forventet.	Acceptabelt
Indtast Username: "TestBruger" E-mail: "email.com" Password: "pas" Confirm Password: "pas"	De indtastede værdier vises i deres respektive tekstbokse. Kordeordet vises som stjerner(*). Der vises fejl ved E-mail feltet "E-mail is not a valid e-mail". Der vises fejl ved Password feltet "The Passwords does not meet the security requirements. The password must be at least 6 characters long, contain at least 1 upper case letter, 1 lower case letter, 1 number, and 1 special character".	De indtastede værdier vises i deres respektive tekstbokse som forventet.	Acceptabelt
Ændre E-mail: "email@domain.com" Password: "Pas" Confirm Password: "Pas"	Der vises fejl ved Password feltet "The Passwords does not meet the security requirements. The password must be at least 6 characters long, contain at least 1 upper case letter, 1 lower case letter, 1 number, and 1 special character".	Fejl vises som forventet.	Acceptabelt

Accepttest User Story 3 - Gæst registrer sig			
Aktivitet	Forventet Resultat	Resultat	Vurdering
Ændre Password: "Pas1" Confirm Password: "Pas1"	Der vises fejl ved Password feltet "The Passwords does not meet the security requirements. The password must be at least 6 characters long, contain at least 1 upper case letter, 1 lower case letter, 1 number, and 1 special character".	Fejl vises som forventet.	Acceptabelt
Ændre Password: "Pas1!" Confirm Password: "Pas1!"	Der vises fejl ved Password feltet "The Passwords does not meet the security requirements. The password must be at least 6 characters long, contain at least 1 upper case letter, 1 lower case letter, 1 number, and 1 special character".	Fejl vises som forventet.	Acceptabelt
Ændre Password: "Password123!"	Kodeordet vises som stjerner(*). Der vises fejl ved Confirm Password feltet "The passwords must match.".	Der vises stadig fejl ved Password.	Uacceptabelt
Ændre Confirm Password: "Password123!"	Kodeordet vises som stjerner(*).	Kodeord vises som stjerner, men der er stadig fejl.	Delvist acceptabelt

Accepttest User Story 3 - Gæst registrer sig			
Aktivitet	Forventet Resultat	Resultat	Vurdering
Tryk på "Register"	Toast vises "You have successfully been registered". Der navigeres til loginsiden.	Der bliver kun navigeret til loginsiden.	Delvist acceptabelt
Tryk på "Register"	Der navigeres til registreringssiden.	Der navigeres som forventet.	Acceptabelt
Indtast Username: "TestBruger" E-mail: "email@domain.com" Password: "Password123!" Confirm Password: "Password123!"	De indtastede værdier vises i deres respektive tekstbokse. Kordeordet vises som stjerner(*).	De vises som forventet, dog med fejl i koden.	Delvist acceptabelt
Tryk på "Register"	Der vises fejl ved Username "The chosen username has already been taken."	Fejlen vises som forventet.	Acceptabelt

Accepttest User Story 4 - Gæst logger ind			
Forudsætning: Accepttest 3 er fuldført & platformen er åben i browseren			
Aktivitet	Forventet Resultat	Resultat	Vurdering
Tryk på "Login"	Der navigeres til loginsiden	Der navigeres som forventet.	Acceptabelt
Indtast Username: "TestBurger" Password: "Password123!"	Username vises i tekstboksen. Kodeordet vises som stjerner(*).	Teksten bliver vist i de relevante tekstfelter.	Acceptabelt
Tryk på "Login"	En toast vises "The username is not registered or the password is wrong"	Fejlene vises som forventet.	Acceptabelt
Indtast Username: "TestBruger" Password: "Password321!"	Username vises i tekstboksen. Kodeordet vises som stjerner(*).	Teksten bliver vist i de relevante tekstfelter.	Acceptabelt
Tryk på "Login"	En toast vises "The username is not registered or the password is wrong"	Der vises de relevante fejl, men der vises ikke nogen toast.	Delvist acceptabelt
Indtast Username: "TestBruger" Password: "Password123!"	Username vises i tekstboksen. Kodeordet vises som stjerner(*).	Teksten bliver vist i de relevante tekstfelter.	Acceptabelt
Tryk på "Login"	En toast vises "You are successfully logged in". Der navigeres til forsiden som bruger.	Der navigeres til forsiden som bruger, men der vises ingen toast.	Delvist acceptabelt

Accepttest User Story 5 - Bruger spiller spil			
Forudsætning: Logget ind som bruger på en computer & er gæst på en anden computer. Platformen er åben i en browser på begge computere			
Aktivitet	Forventet Resultat	Resultat	Vurdering
På den første computer: Tryk på "New Game"	En pop-op menu viser valgmuligheder for spil.	Der bliver navigeret til kø-siden.	Uacceptabelt
Tryk på "Nim"	Der bliver vist en pop-op menu som viser at man er sat i kø til "Nim".	Der vises kun, at man bliver-sat i kø til "Nim"	Delvist acceptabelt
På den anden computer: Tryk på "New Game"	En pop-op menu viser valgmuligheder for spil.	Der bliver navigeret til kø-siden.	Uacceptabelt
Tryk på "Nim"	Der bliver vist en pop-op menu som viser at man er sat i kø til "Nim". Der bliver startet et spil mod den første computer.	Der vises kun, at man bliver-sat i kø til "Nim" og spillet starter	Delvist acceptabelt

Accepttest User Story 6a - Bruger udfordrer til spil - Via Chat			
Forudsætning: Logget ind som bruger på en computer og TestBruger* på en anden computer, & platformen er åben i en browser på begge computere.			
Aktivitet	Forventet Resultat	Resultat	Vurdering
På den anden computer: Indtast "Fight me" i den globale chat.	Teksten vises i tekstboksen.	Den indtastede tekst ses i beskedfeltet.	Acceptabelt
Tryk på "Send"	Teksten vises i chatboksen på begge computere.	Beskeden vises som forventet.	Acceptabelt
På den første computer: Tryk på "TestBruger" i chatboksen	En menu vises ved brugernavnet.	Funktionalitet af at klikke på et navn i chatten er ikke implementeret.	Uacceptabelt
Tryk på "Challenge"	En pop-op menu viser valgmuligheder for spil.	Funktionalitet af udfordringer er ikke implementeret.	Uacceptabelt
Tryk på "Nim"	Et modal pop-op vindue viser, at den givende bruger er blevet udfordret i "Nim", og tæller ned fra 30 sekunder. Et modal pop-op vindue viser, at brugeren er blevet udfordret, og tæller ned fra 30 sekunder.	Funktionalitet af udfordringer er ikke implementeret.	Uacceptabelt
Tryk på "Cancel"	De modale pop-op vinduer lukkes på begge computere.	Funktionalitet af udfordringer er ikke implementeret.	Uacceptabelt

Accepttest User Story 6a - Bruger udfordrer til spil - Via Chat			
Aktivitet	Forventet Resultat	Resultat	Vurdering
Tryk på "TestBruger" i chatboksen	En menu vises ved bruger-navnet.	Funktionalitet af at klikke på et navn i chatten er ikke implementeret.	Uacceptabelt
Tryk på "Challenge"	En pop-op menu viser valgmuligheder for spil.	Funktionalitet af udfordringer er ikke implementeret.	Uacceptabelt
Tryk på "Nim"	Et modal pop-op vindue viser, at den givende bruger er blevet udfordret i "Nim", og tæller ned fra 30 sekunder. Et modal pop-op vindue viser, at brugeren er blevet udfordret, og tæller ned fra 30 sekunder.	Funktionalitet af udfordringer er ikke implementeret.	Uacceptabelt
På den anden computer: Tryk på "Decline"	De modale pop-op vinduer lukkes på begge computere.	Funktionalitet af udfordringer er ikke implementeret.	Uacceptabelt
På den første computer: Tryk på "TestBruger" i chatboksen	En menu vises ved bruger-navnet.	Funktionalitet af udfordringer er ikke implementeret.	Uacceptabelt

Accepttest User Story 6a - Bruger udfordrer til spil - Via Chat			
Aktivitet	Forventet Resultat	Resultat	Vurdering
Tryk på "Challenge"	En pop-op menu viser valgmuligheder for spil.	Funktionalitet af udfordringer er ikke implementeret.	Uacceptabelt
Tryk på "Nim"	Et modal pop-op vindue viser, at den givende bruger er blevet udfordret i "Nim", og tæller ned fra 30 sekunder. Et modal pop-op vindue viser, at brugeren er blevet udfordret, og tæller ned fra 30 sekunder.	Funktionalitet af udfordringer er ikke implementeret.	Uacceptabelt
På den anden computer: Tryk på "Accept"	Spillet "Nim" går i gang på begge computere.	Funktionalitet af udfordringer er ikke implementeret.	Uacceptabelt

Accepttest User Story 6b - Bruger udfordrer til spil - Via Venneliste			
Forudsætning: Logget ind som bruger og TestBruger*, de to brugere er venner & platformen er åben i en browser på begge computere.			
Aktivitet	Forventet Resultat	Resultat	Vurdering
På den første computer: Tryk på "Friends"	Chatboksen erstattes med vennelisten.	Chatboksen erstattes som forventet.	Acceptabelt
Udfør samme testscenarier som i Accepttest 6a.	De samme forventede resultater som specificeret i Accepttest 6a.	Da funktionalitet for udfordring ikke er implementeret fugnerer det ikke.	Uacceptabelt

Accepttest User Story 7a - Bruger benytter chatten - Globalchat			
Forudsætning: Logget ind som bruger & platformen er åben i en browser på to computere			
Aktivitet	Forventet Resultat	Resultat	Vurdering
Indtast "Hej alle sammen"	Teksten vises i tekstboksen.	Teksten vises som forventet.	Acceptabelt
Tryk på "Send"	Teksten vises i chatboksen på begge computere.	Teksten vises som forventet.	Acceptabelt

Accepttest User Story 7b - Bruger benytter chatten - In-game chat			
Forudsætning: Logget ind som bruger, har startet et nyt spil, platformen er åben i browseren, & er matchet med sig selv i en anden browser på en anden computer.			
Aktivitet	Forventet Resultat	Resultat	Vurdering
Indtast "Hej modstander"	Teksten vises i tekstboksen.	Teksten vises som forventet.	Acceptabelt
Tryk på "Send"	Teksten vises i chatboksen på begge computere.	Teksten vises ikke i chatten.	Uacceptabelt

Accepttest User Story 8 - Bruger benytter vennelisten			
Forudsætning: Logget in som bruger og TestBruger* & platformen er åben i en browser på begge computere. Computeren med TestBruger har sendt en besked i den globale chat.			
Aktivitet	Forventet Resultat	Resultat	Vurdering
Tryk på "TestBruger" i chatten	En menu vises ved brugernavnet.	Funktionalitet med venne- liste er ikke implemente- ret.	Uacceptabelt
Truk på "Send friend request"	Toast vises "Friend request sent".	Funktionalitet med venne- liste er ikke implemente- ret.	Uacceptabelt
På den anden computer: Tryk på "Friends"	Chatboksen erstattes med vennelisten. Der vises "Pending request from [brugernavn]" på listen.	Funktionalitet med venne- liste er ikke implemente- ret.	Uacceptabelt
Tryk på "Decline"	Anmodningen fjernes.	Funktionalitet med venne- liste er ikke implemente- ret.	Uacceptabelt
På den første computer: Tryk på "TestBruger" i chatten	En menu vises ved bruger- navnet.	Funktionalitet med venne- liste er ikke implemente- ret.	Uacceptabelt
Tryk på "Send friend request"	Toast vises "Friend request sent". Der vises "Pending request from [brugernavn]" på listen på den anden computer.	Funktionalitet med venne- liste er ikke implemente- ret.	Uacceptabelt

Accepttest User Story 8 - Bruger benytter vennelsiten			
Aktivitet	Forventet Resultat	Resultat	Vurdering
På den anden computer: Tryk på "Accept".	Den første bruger tilføjes som ven på den anden brugers venneliste.	Funktionalitet med venne- liste er ikke implemente- ret.	Uacceptabelt
På den første computer: Tryk på "Friends"	Chatboksen erstattes med vennelisten. Test-Bruger vises som ven på vennelisten.	Funktionalitet med venne- liste er ikke implemente- ret.	Uacceptabelt
Tryk på "Remove fri- end" ved TestBruger	En pop-op menu spørger efter bekræftelse om at fjerne TestBruger fra ens venneliste.	Funktionalitet med venne- liste er ikke implemente- ret.	Uacceptabelt
Tryk på "No"	Pop-op menuen lukkes.	Funktionalitet med venne- liste er ikke implemente- ret.	Uacceptabelt
Tryk på "Remove fri- end" ved TestBruger	En pop-op menu spørger efter bekræftelse om at fjerne TestBruger fra ens venneliste.	Funktionalitet med venne- liste er ikke implemente- ret.	Uacceptabelt
Tryk på "Yes"	Pop-op menuen lukkes og TestBruger fjernes fra vennelisten. Der kan på den anden computer ses at den første bruger forsvinder fra TestBrugers venneliste.	Funktionalitet med venne- liste er ikke implemente- ret.	Uacceptabelt

Accepttest User Story 8 - Bruger benytter vennelsiten			
Aktivitet	Forventet Resultat	Resultat	Vurdering
Tryk på "Add friend"	En pop-op menu spørger efter et brugernavn.	Funktionalitet med venne-liste er ikke implementeret.	Uacceptabelt
Tryk på "Cancel"	Pop-op menuen lukkes.	Funktionalitet med venne-liste er ikke implementeret.	Uacceptabelt
Tryk på "Add friend"	En pop-op menu spørger efter et brugernavn.	Funktionalitet med venne-liste er ikke implementeret.	Uacceptabelt
Indtast "TestBruger"	Det indtastede brugernavn vises i tekstfeltet.	Funktionalitet med venne-liste er ikke implementeret.	Uacceptabelt
Tryk på "Send"	Pop-op menuen lukker. Toast vises "Friend request sent". Der vises "Pending request from [brugernavn]" på den anden computer.	Funktionalitet med venne-liste er ikke implementeret.	Uacceptabelt
På den anden computer: Tryk på "Decline"	Anmodningen fjernes fra listen.	Funktionalitet med venne-liste er ikke implementeret.	Uacceptabelt

Accepttest User Story 8 - Bruger benytter vennelsiten			
Aktivitet	Forventet Resultat	Resultat	Vurdering
På den første computer: Tryk på "Add friend"	En pop-op menu spørger efter et brugernavn.	Funktionalitet med venne- liste er ikke implemente- ret.	Uacceptabelt
Indtast "TestBruger"	Det indtastede bruger- navn vises i tekstfeltet.	Funktionalitet med venne- liste er ikke implemente- ret.	Uacceptabelt
Tryk på "Send"	Pop-op menuen luk- ker. Toast vises "Friend request sent". Der vises "Pending request from [brugernavn]" på den anden computer.	Funktionalitet med venne- liste er ikke implemente- ret.	Uacceptabelt
På den anden computer: Tryk på "Accept"	Den første bruger tilføjes som ven på TestBrugers venneliste.	Funktionalitet med venne- liste er ikke implemente- ret.	Uacceptabelt

Accepttest User Story 9 - Bruger benytter egen profilside			
Forudsætning: Logget ind som TestBruger* & platformen er åben i en browser.			
Aktivitet	Forventet Resultat	Resultat	Vurdering
Tryk på "Profile"	Der navigeres til profilside.	Funktionalitet med profil er ikke implementeret.	Uacceptabelt
Tryk på "Change E-mail"	En pop up menu spørger om den nye e-mail.	Funktionalitet med profil er ikke implementeret.	Uacceptabelt
Indtast "forkert@.com"	Det indtastede vises i tekstfeltet. Der vises en fejlbesked at det ikke er en ordentlig e-mail.	Funktionalitet med profil er ikke implementeret.	Uacceptabelt
Indtast "rigtig@domain.com"	Det indtastede vises i tekstfeltet.	Funktionalitet med profil er ikke implementeret.	Uacceptabelt
Tryk på "Confirm"	Pop up menuen lukker. Toast vises "E-mail successfully changed.". Der ses at ens e-mail er ændret til "rigtig@domain.com".	Funktionalitet med profil er ikke implementeret.	Uacceptabelt
Tryk på "Change Password"	En pop-up menu spørger om det nuværende kodeord samt nyt kodeord.	Funktionalitet med profil er ikke implementeret.	Uacceptabelt

Accepttest User Story 9 - Bruger benytter egen profilside			
Aktivitet	Forventet Resultat	Resultat	Vurdering
Indtast Current Password: "Password123!" New Password: "Password321#" Confirm Password: "Password321#" Udfør samme testscenarier for password verifikation som i Accepttest 3	Det indtastede vises som stjerner(*) i deres respek- tive tekstfelter. Samme forventede resultatersom specificeret i Accepttest 3.	Funktionalitet med profil er ikke imple- menteret.	Uacceptabelt
Tryk på "Confirm"	Pop up menuen lukker. Toast vises "E-mail suc- cessfully changed.".	Funktionalitet med profil er ikke imple- menteret.	Uacceptabelt
Tryk på "Change Pictu- re"	En pop-op menu vises og efterspørger en sti til hvor billedet er lokaliseret lo- kalt på computeren.	Funktionalitet med profil er ikke imple- menteret.	Uacceptabelt
Tryk på "Cancel"	Pop-op menuen lukkes.	Funktionalitet med profil er ikke imple- menteret.	Uacceptabelt
Tryk på "Change Pictu- re"	En pop-op menu vises og efterspørger en sti til hvor billedet er lokaliseret lo- kalt på computeren.	Funktionalitet med profil er ikke imple- menteret	Uacceptabelt
Naviger til et billede og vælg det	Det valgte billede uploa- des og kan ses i en thumb- nail.	Funktionalitet med profil er ikke imple- menteret.	Uacceptabelt

Accepttest User Story 9 - Bruger benytter egen profilside			
Aktivitet	Forventet Resultat	Resultat	Vurdering
Tryk på "Confirm"	Pop-op menuen lukkes. Toast vises "Picture successfully changed." Det nye billede ses i billedrammen.	Funktionalitet med profil er ikke implementeret.	Uacceptabelt
Tryk på "Statistics"	Profilsiden erstattes med statistiksiden.	Funktionalitet med profil er ikke implementeret.	Uacceptabelt

Accepttest User Story 10a - Bruger ser på andres brugerprofil - Gennem Chat			
Aktivitet	Forventet Resultat	Resultat	Vurdering
På den anden computer: Send "Hej alle" i chatten jf. accepttest af User Story 7	Teksten vises i begge chatbokse.	Teksten vises som forventet.	Acceptabelt
På den første computer: Tryk på brugernavn i chatboksen.	En pop up menu vises ved brugernavnet.	Funktionalitet med profil er ikke implementeret.	Uacceptabelt
Tryk på "Profile"	Der navigeres til den anden brugers offentlige profilside.	Funktionalitet med profil er ikke implementeret.	Uacceptabelt

Accepttest User Story 10b - Bruger ser på andres brugerprofil - Gennem Venneliste			
Forudsætning: Logget ind som bruger, platformen er åben i browseren, man har mindst én ven på vennelisten.			
Aktivitet	Forventet Resultat	Resultat	Vurdering
Tryk på "Friends"	Chatboksen erstattes med vennelisten.	Chatboksen erstattes som forventet	Acceptabelt
Tryk på "Profile" under brugernavnet på listen.	Der navigeres til den anden brugers offentlige profilside.	Funktionalitet med profil er ikke implementeret.	Uacceptabelt
Tryk på "Profile"	Der navigeres til profilsiden for den anden bruger.	Funktionalitet med profil er ikke implementeret.	Uacceptabelt

Kapitel 4

Litteratur

- [1] Paulo Gomes. Unit testing and the arrange, act and assert (aaa) pattern. <https://medium.com/@pjbfgf/title-testing-code-ocd-and-the-aaa-pattern-df453975ab80>, 2021. Url date: 1. juni 2021.
- [2] tutorialspoint. Top down integration testing. https://www.tutorialspoint.com/software_testing_dictionary/top_down_integration_testing.htm, 2021. Url date: 1. juni 2021.