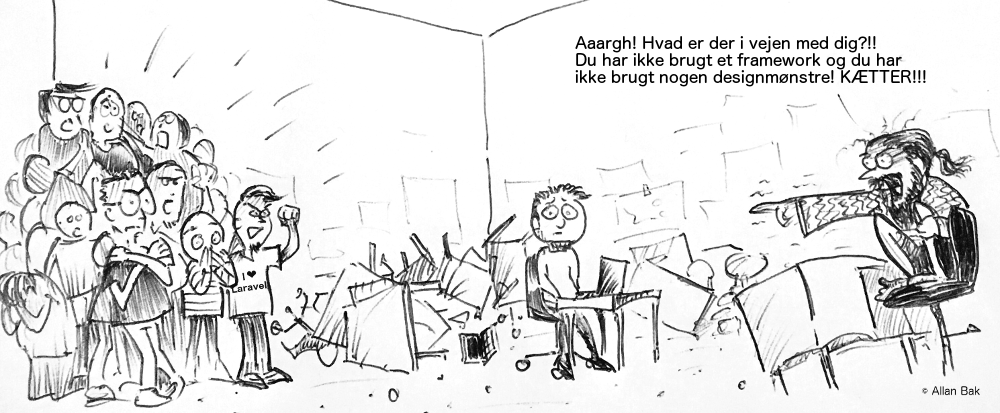
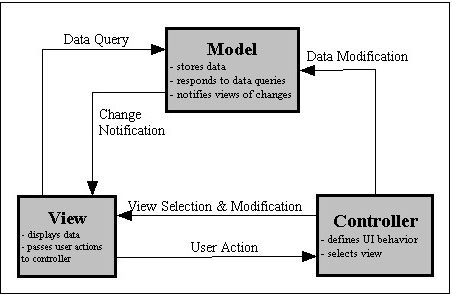
Her er en, der ikke har fulgt med tiden. Don’t be him!

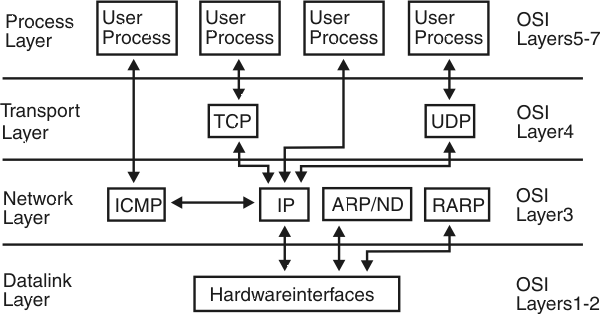


Herunder er det skitseret, hvordan sammenhængen mellem de tre dele af MVC kan opfattes.

 [18]

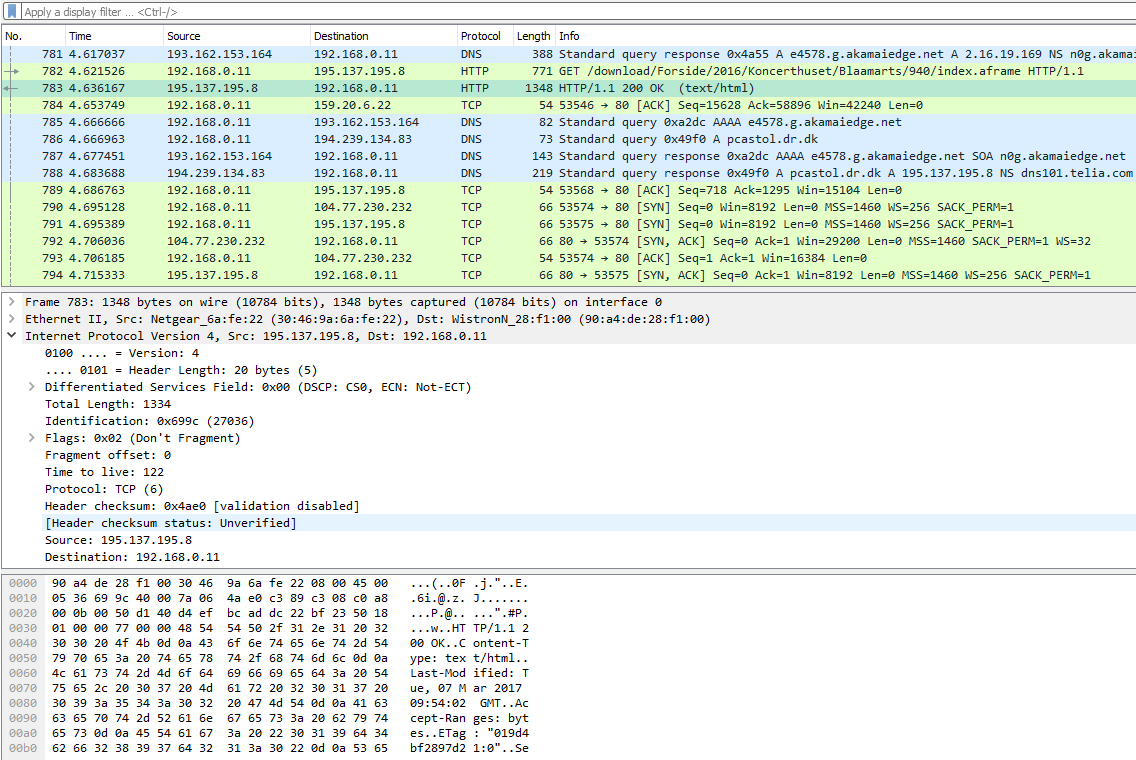
” A controller is the means by which the user interacts with the application.” (ootips.org [34])

Denne type kryptering vil f.eks. foregå med TLS, som gennemføres i transportlaget

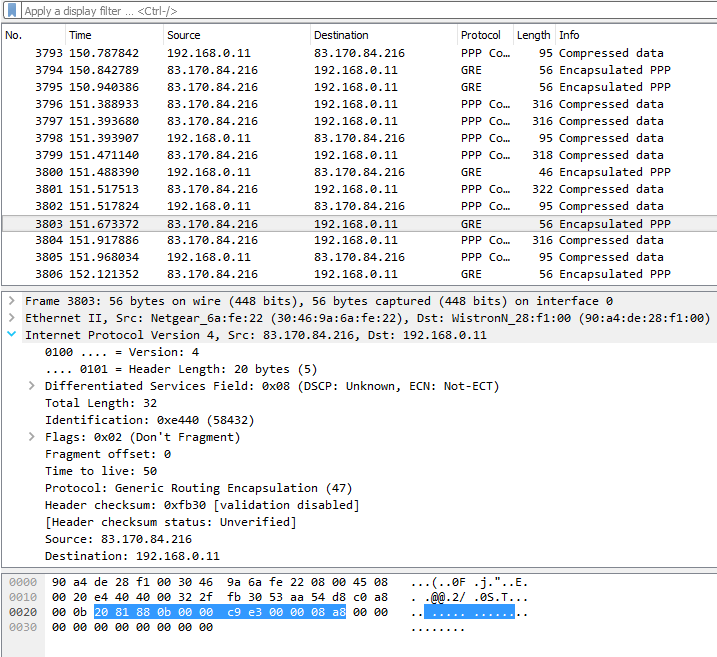
 [23]

VPN og krypterer hele pakken, der skal sendes, dvs. header og payload. Den arbejder på linklaget og gør, at

Et opslag på dr.dk uden VPN viste Wireshark sådan med forskellige protokoller og IP-adresser:

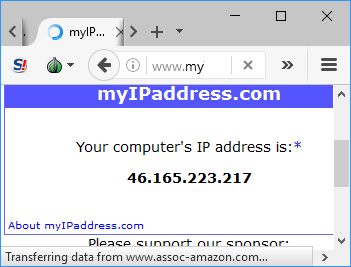
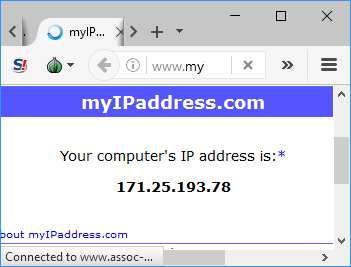


I denne session blev VPN anvendt. Der var et par ARP requets, men ellers var al trafikken mellem pc'en (192.168.0.11) og VPN-severen (83.170.84.216), og protokollerne ikke afslørede meget.

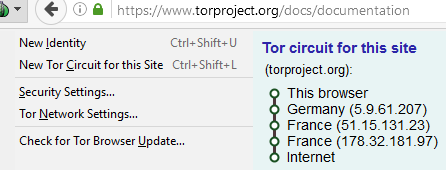
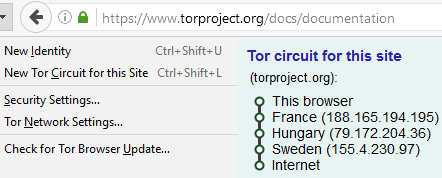


Et circuit opbygges ved at sende en Create-kommando til første node. Herefter bygges én nøgle, der skal anvendes til kryptering på første delstrækning. Efterfølgende sendes en Extend-kommado til sidste node i det partielt byggede circuit. Denne node vil så sende en Create til næste node.

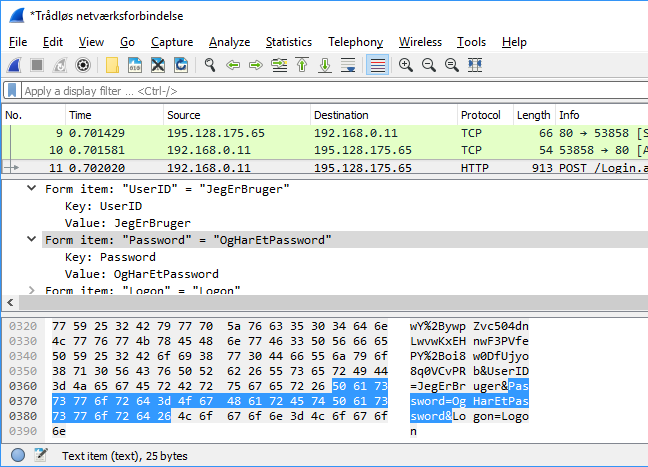
For hvert HTTP-request bygges et nyt circuit, hvilket hæmmer mulighederne for at følge trafikken. Det eneste der er sket mellem disse billeder, er en refresh (Ctrl-R)



På samme måde er her to circuits for at ramme Torprojects dokumentation to gange med refresh.



I password input text feltet i på websiden vises det indtastede password som stjerner eller cirkler, men de optræder eller i klartekst. I nedenstående billede har Wireshark opsnappet en logon-dialog over en ikke-krypteret forbindelse.



|  |  |  |
| --- | --- | --- |
| **Logon** |  |  |
| brugerId | lastLogon | logonCnt |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **User** |  |  |  |  |  |  |
| brugerId | createTs | password | salt | email | prim | erAdmin |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Arrangement** |  |  |  |  |  |
| arrangementId | arrangementBeskrivelse | createTs | arrangementWeb | arrangementDt | arrangementTid |

|  |  |
| --- | --- |
|  |  |
| spiseSted | spiseTid |

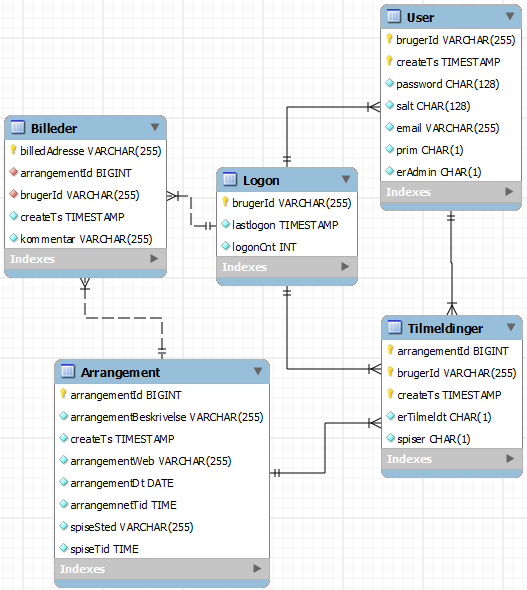
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tilmeldinger** |  |  |  |  |
| arrangementId | brugerId | createTs | erTilmeldt | spiser |

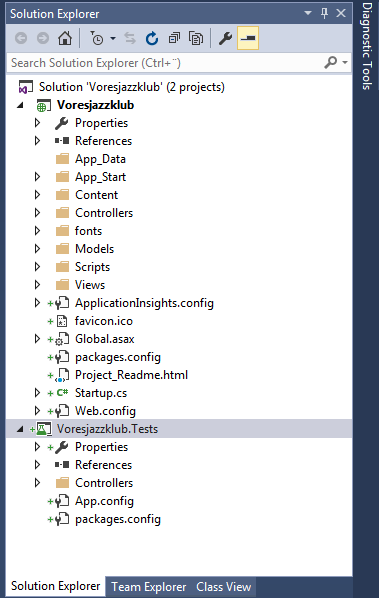
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Billeder** |  |  |  |  |
| billedAdresse | arrangementId | brugerId | createTs | kommentar |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **UserPassword** | |  |  |  | **Email** |  |  |  | **Admin** |  |  |
| brugerId | createTs | password | salt |  | brugerId | createTs | email |  | brugerId | createTs | erAdmin |

Af praktiske grunde accepteres redundans, og normalisering ud over 3NF (og evt. Boyce Codd) vil ikke blive gennemført.

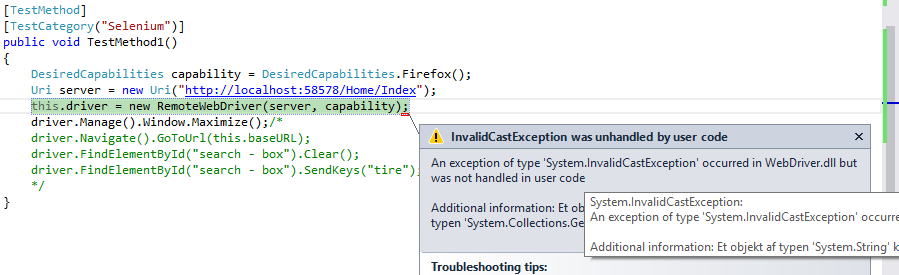
Det resulterende EER-diagram importeret i MySQL-Workbench.



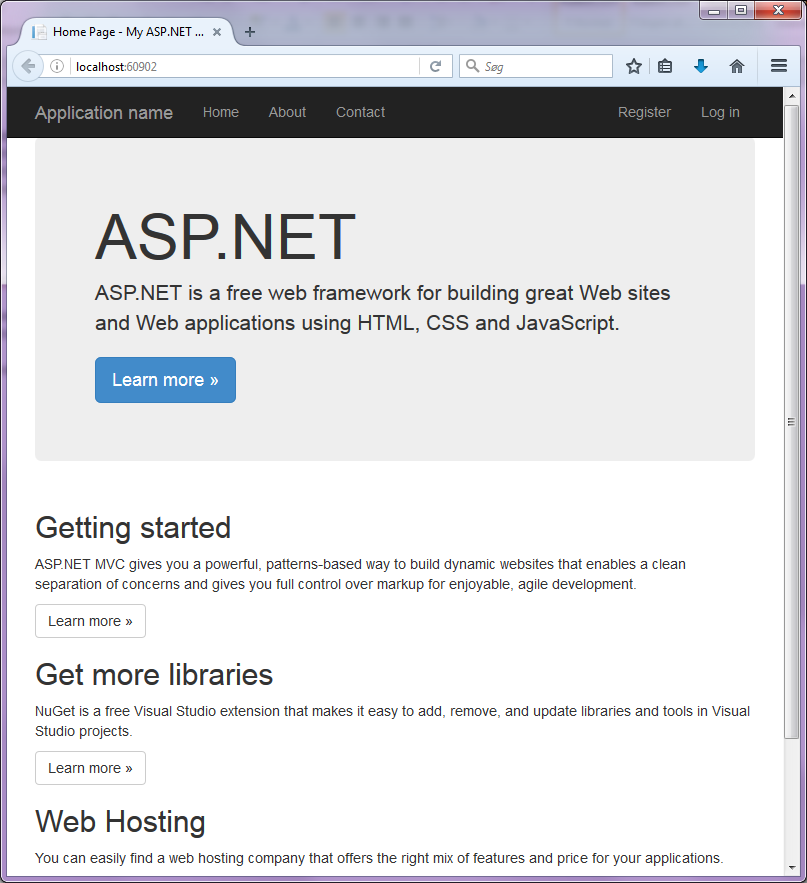


Der er små 600 filer og over 800 mapper i det automatisk genererede projekt.

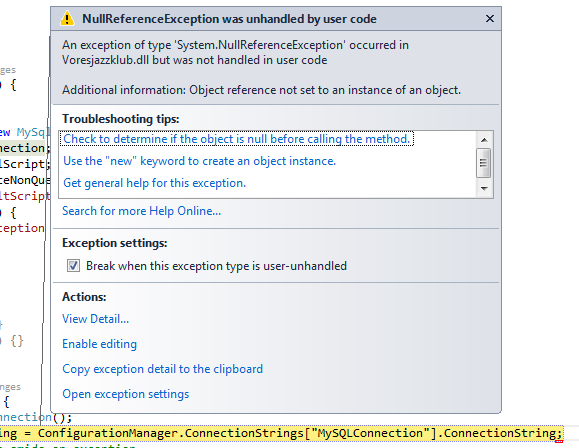
Selenium skulle selvfølgeligt prøves, men der kom flere udfordringer ud af det. Google kunne ikke umiddelbart komme med nogen brugbar forklaring, så tilbage er kun at grave sig ned i problemet, hvilket formentlig er ganske tidskrævende.



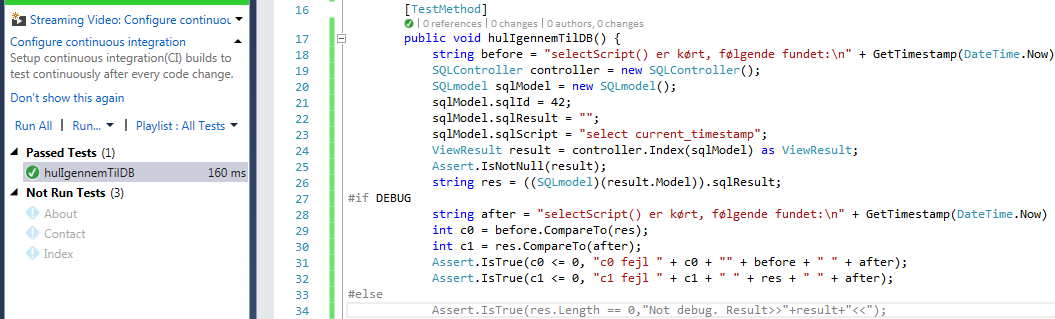
Den kørende web-applikation right out of the box.

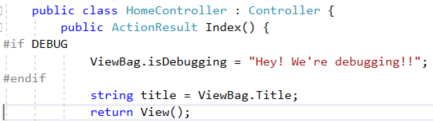


Den første test gik helt galt da testprojektet ikke kunne finde MySQLConnection.



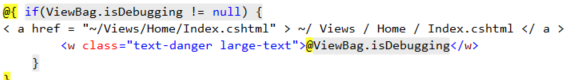
I Testprojektets App.config indsættes connection strings fra hovedprojektets Web.config, og så kører den første test pænt igennem

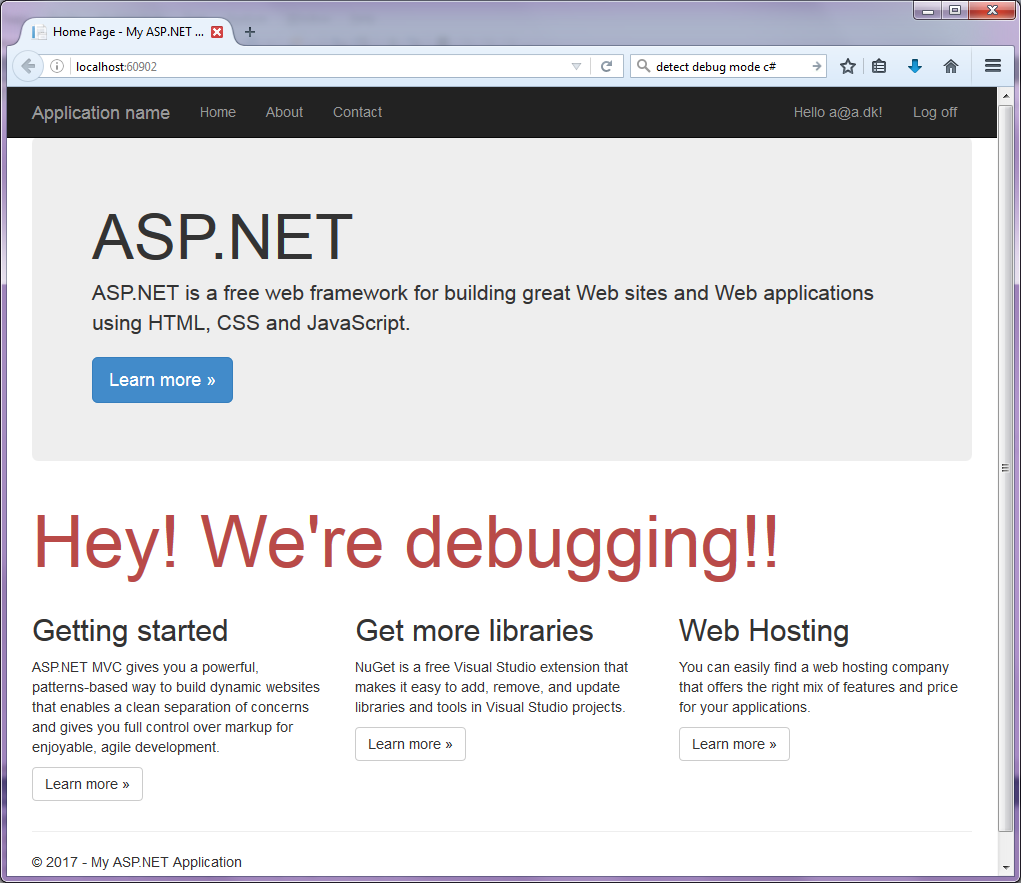




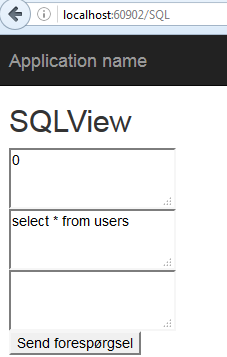
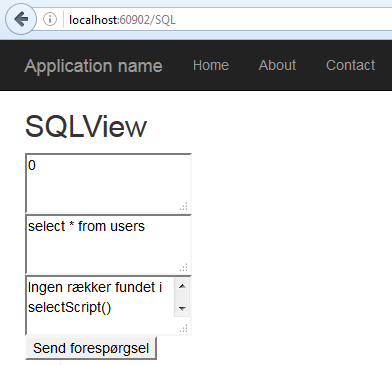
DEBUG er en variable, som VS sætter ud fra om den arbejder i Debug eller Release modus. Den kan så anvendes af C#’s precompiler til at inkludere eller ekskludere kode.

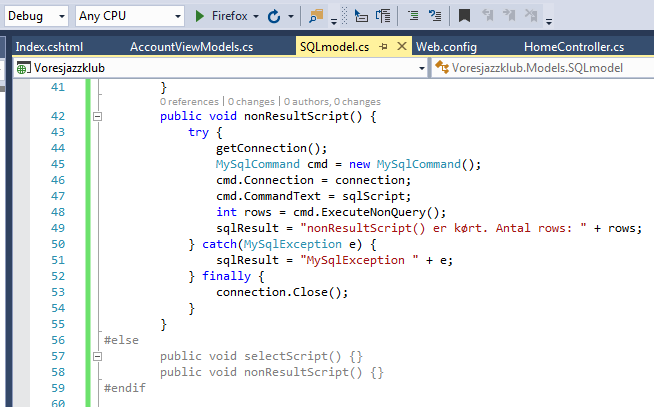
Her sættes en tekststreng i ViewBag, som er en ustruktureret samling objekter, der kan sendes til View’et. Teksten vises i rød og med en stor skrifttype





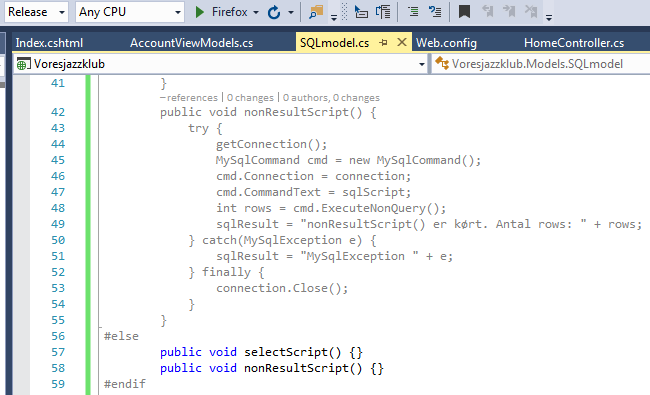
Ved at være i Debug-modus, kan man køre SQL-scripts. Det kan man ikke i release-modus



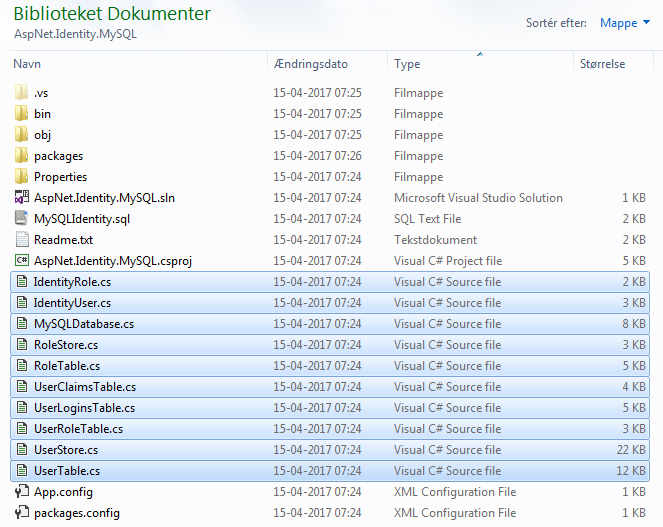


I øverste venstre hjørne står ”Debug” og i kode testes på om DEBUG er sat. Derfor kan SQL-scriptet køres

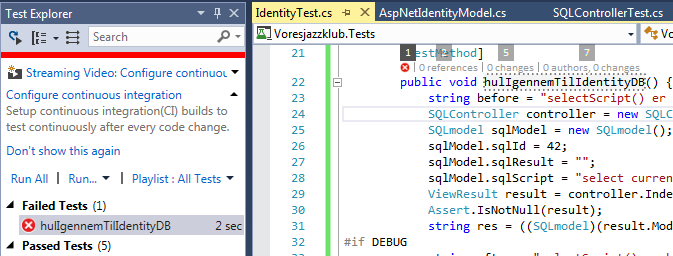
Tilsvarende er VS i Release-modus og derfor anvendes to absolut tomme metoder, hvorfor der intet sker.

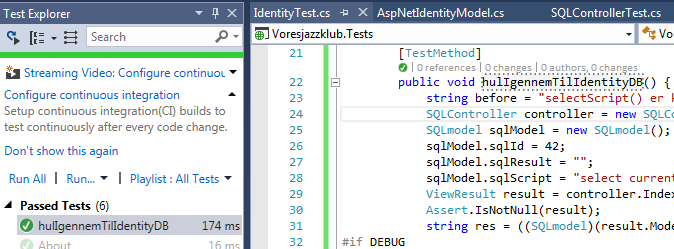


Disse klasser kommer der ud af at hente AspNet.Identity.MySQL:

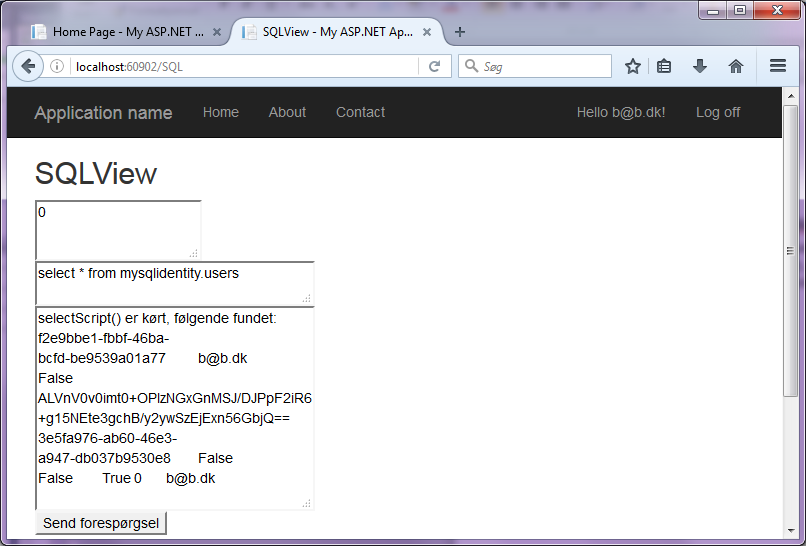


Når der bliver lavet noget nyt, skal der også være kørt en test. Første gang den køres, forventes det, at den fejler, og når der er kodet tilpas meget, og testen køres igen, skal den blive grøn. Det er naturligvis også tilfælde her, hvor det tjekkes om der er forbindelse til den nye database.



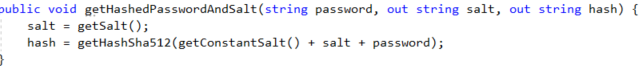


Ved at gå ind på siden SQL, hvor et vilkårligt SQL-script kan køres fra, hvilket muligvis kan opfattes som en sikkerhedstrussel, vælges alle registrerede brugere. Her genfindes så den nyligt registrerede bruger.



Downside ved at anvende Microsfts identitests framework er, at det kompromitterer den valgte datamodel. Det betyder så også, at applikationens funktionalitet bliver lidt anderledes en oprindeligt tænkt.

Der er fremstillet to metoder til brug, hvor brugerens password skal håndteres. Det er ved oprettelse, ændring og tjek af validitet. Den ene får et password i klartekst ind og afleverer hash og salt. Som det ses, er hashen genereret ud fra en konstant værdi, en tilfældig værdi og password. Disse tre værdier er tekststrenge, der konkateneres.

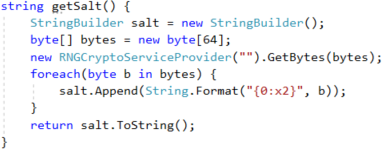


Den anden metode tager password i klartekst og det salt, der er persisteret i databasen, og returnere samme værdi som i metoden ovenfor.

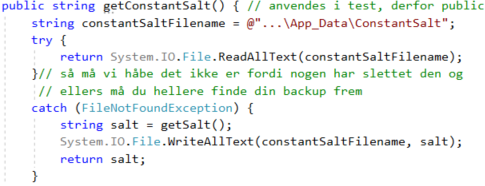


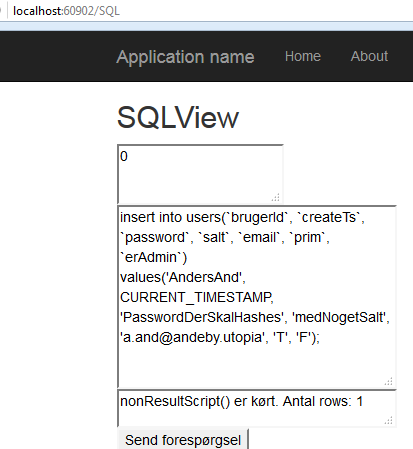
Selve hashingen står .Net for i klassen SHA512Managed i metoden ComputeHash. Teksten konverteres til bytes, og outputtet konverteres til en hexadecimal streng.





Den sidste metode er til det konstante salt. Det bliver gemt i maskinens filsystem. Fordelen er, at salt kan beskyttes, så det kun er en administrator, der har adgang til det. Der er ingen garantier mod tyveri, men det er i hvert fald gjort besværligt.

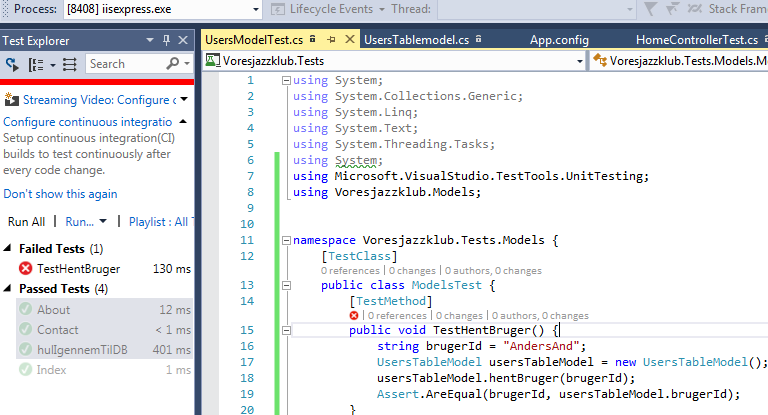


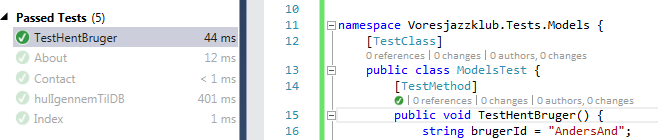


insert into users(`brugerId`, `createTs`, `password`, `salt`, `email`, `prim`, `erAdmin`)

values('AndersAnd', CURRENT\_TIMESTAMP, 'PasswordDerSkalHashes', 'medNogetSalt', 'a.and@andeby.utopia', 'T', 'F');

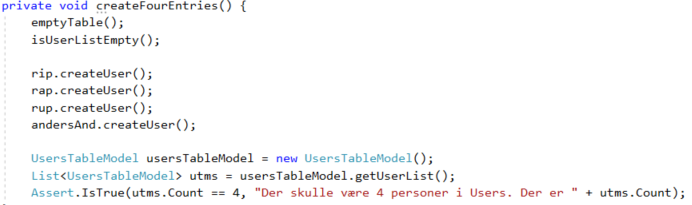
Man starter med testmetoden. IDE’et vil ikke kunne kompilere testmetoden, da der hverken er en klasse eller metode IDE’et kan finde referencer til. Så laves det minimum, der skal til at kunne kompilere, hvorefter testen køres.



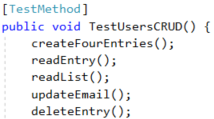


De næste trin er at tilføre og ændre koden, i den testede metode, indtil testen bliver grøn. Alle test, til de berørte komponenter, skal også køres igen, for at se, om man har fået ødelagt noget.

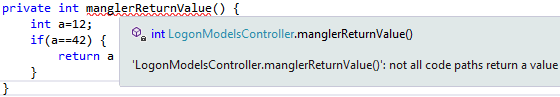
Her er en af de underrutiner, der anvendes som et af skridtene i test af alle CRUD-operationerne. Først tømmes tabellen og det tjekkes om den er tom. Herefter oprettes fire rækker i tabellen, og det tjekkes om der så er fire elementer i listen, der hentes i databasen.

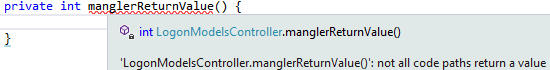


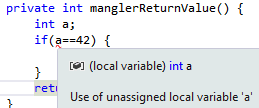
Her en af de testmetoder, der gennemløber alle de mulige CRUD-operationer. Først oprettes fire rækker i tabellen med ovenstående metode.



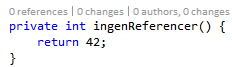
Statisk test bliver udført af IDE'et. Her er en metode, der ikke altid returnerer noget. Det tæller som fejl.



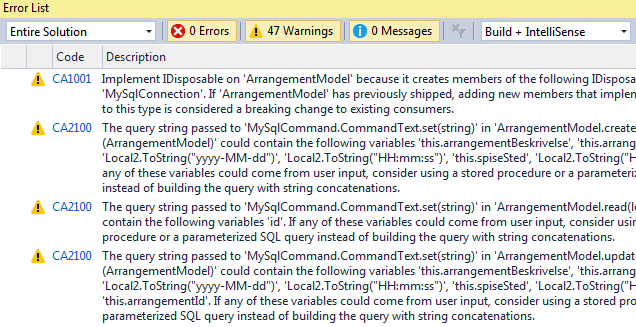




Nu returneres der rent faktisk noget, men variablen, der opereres på, har ikke fået tildelt en værdi



Dette er så endeligt en metode, der ikke får røde bølgelinjer, men så ikke har nogle referencer.



Der er en mere dybdegående statisk analyse, som giver en længere række warnings. I en produktionsversion af produktet er hverken fejl eller advarsler acceptable. Der er to muligheder: Enten at rette i koden ud fra beskrivelsen, eller at konfigurere analyseværktøjet til ikke give bestemte advarsler. Det sidste skal naturligvis kun ske, hvis der kan argumenteres forsvarligt for det.

Deployment gik tilsyneladende godt, men applikationen kunne ikke køre på test.voresjazzklub.dk, kun lokalt på udviklingsmaskinen. Det skal undersøges, hvorfor der er forskellige opfattelser.

