



Afprøvning af Cross-platform frameworks

4. SEMESTER EKSAMENS RAPPORT 27/5-2016

MARTIN HERODT

Indhold

| | |
|---|----|
| Introduktion..... | 2 |
| Problemstilling..... | 3 |
| Problemformulering..... | 3 |
| Tidsplan | 4 |
| Cross-platform udviklingsmiljøer | 5 |
| Afprøvninger..... | 6 |
| jQuery Mobile..... | 7 |
| Hvad er Phonegap/Cordova egentlig?..... | 8 |
| Microsoft Xamarin | 10 |
| Ionic | 12 |
| Konklusion | 13 |
| Litteraturliste..... | 14 |

Introduktion

Jeg har fået til opgave at finde et spændende eksamensprojekt, som en del af datamatiker uddannelsen på 4. semester, til dette skal vi lave en synopsis på max 10 sider.

Jeg har valg at gå i dybden inden for udviklingen af mobile løsninger. Jeg vil gerne vide hvordan man vælger det helt rigtige crossplatform framework, samt finde ud af hvor dens styrker og svagheder er. Her vil jeg udføre ud fra nogle simple test.

Som en del af 4. semester, valgte jeg android som en af mine 3 valgfag. Jeg synes det var et af de mest spændende udviklingsmiljøer jeg nogle sinde har arbejdet med. Jeg begyndte derfor at tænke på om der var andre og nemmere måder at udvikle apps på.

Jeg vil gerne takke min vejleder Anders Kristian Børjesson for de 3 indholdsrige og inspirerende vejledninger, jeg har modtaget i forløbet.

Problemstilling

I en digital tid hvor alle sidder naglet fast til mobiltelefoner og tablets, sidder mange it-udviklere og arbejder hårdt for at udvikle apps for store og små virksomheder. Når firmaer så står over for en problemstilling, og skal søge efter en it-udvikler der kan løse dette, bliver de tit præsenteret for en meget høj pris, samt en lang tidsperiode ind til deadline.

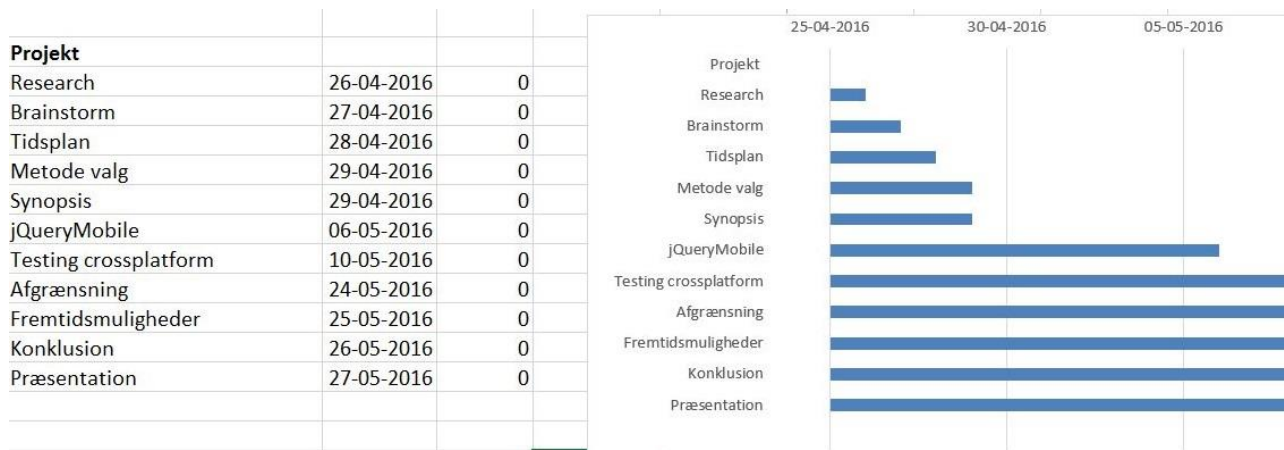
Jeg vil i dette projekt undersøge og afprøve de forskellige muligheder der er for at vælge den helt rigtige løsning.

Problemformulering

- Hvordan udvikler jeg mobil/tablet apps til de mest brugte platforme?
- Hvad er sværhedsgraden for at udvikle en cross-platform app?
- Hvilket kodesprog bruges der til de forskellige løsninger?
- Er der fremtid i cross-platform apps?

Tidsplan

Efter første evalueringsmøde med Anders Børjesson "min vejleder", fastlagde jeg en tidsplan for hele projektførløbet. Tidsplanens formål var at mindske risikoen for kommende og uventede problemer, samt at planlægge og organisere det ind i min dagligdag. På nedenstående billede ses min første tidsplan.

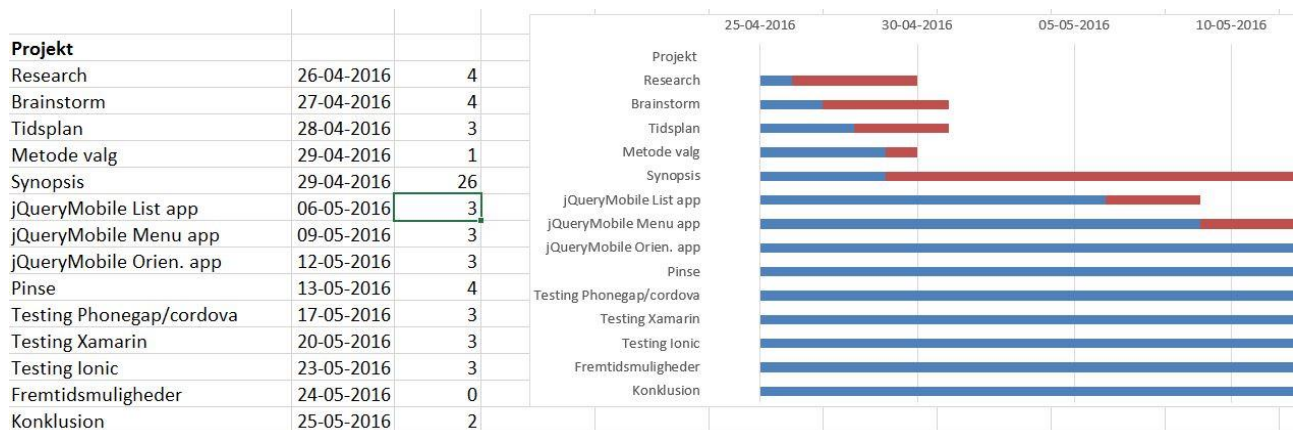


Efter andet evalueringsmøde, kom vi frem til at lave en smule luft op til nogle af milepælene, for at være imødekommende over for evt. ændringer, samt bryde nogle af de lange tidsperioder ned i mindre, og mere præcise dage, som ses på nedenstående billede.



Mit sidste evalueringsmøde før afleveringen har været meget vigtig for mig. Her fik vi diskuteret hvordan jeg får finpustet min synopsis, dvs. få gjort billederne større og mere tydelige, samt fjerne alt overflødige tekst og rettet rapporten for fejl.

Angående min tidsplan, så går alt efter skemaet som ses på nedenstående billede. Plan er fra i morgen at kigge på en forside og indholdsfortegnelse, samt en konklusion.



Cross-platform udviklingsmiljøer

I en tid med flere forskellige mobiler, tablets, laptops og nu også smartwatches, er der stor fremtid i cross-platform udviklingen. Der er mange frameworks på markedet i dag, som er i konstant udvikling. Alle har med det i fokus at gøre cross-platform appen, så tæt på en native app som overhoved muligt.

Her er et par af de mange frameworks:

- Phonegap/Cordova
- Xamarin
- Ionic
- Mobile Angular UI
- Framework 7
- Kendo UI
- NativeScript

I dette projekt har jeg valgt at fokusere på Phonegap, Xamarin og Ionic. Jeg vil også kigge på andre frameworks som jQueryMobile og ionic creator, det skal lige siges at de web apps frameworks og ikke er cross-platforms frameworks.

De 2 mest brugte udviklingsmiljøer er Apples IOS(Xcode) og Googles Android (Andriod studio). Herefter kommer der andre som Microsoft Windows (Visual Studio) og Blackberry. Jeg er i besiddelse af en iPhone og en Android mobiltelefon, derfor er det kun muligt for mig at afprøve på disse enheder.

Afprøvninger

For at skabe et overblik over de tests jeg ønsker at udføre, har jeg valgt at lave en tabel, som ses på nedenstående billede. Tabellens horisontale række beskriver de funktioner jeg ønsker at afprøve og det kodesprog servicen acceptere.

På grund af at hybrid apps benyttes af forskellige kodesprog og services, vælger jeg at lave en sværhedsgrad kolonne i en skala fra 1 til 5, hvor 1 er nemt og 5 er svært. Sværhedsgraden vurderes ud fra dokumentationen på servicen, installationen og kodesproget.

Tabellens første kolonne er den service jeg ønsker at afprøve funktionerne i. De hvide felter indikere hvad jeg først vil afprøve og de grå felter indikere hvis der er tid til overs.

Det er med vilje at jeg afprøver lister, knapper og orienteringer frem for native funktioner, fordi at der typisk findes dokumentationer over servicens Api, som f.eks. Phonegaps native Api¹. Så for at holde det helt simpelt, afprøver jeg de mest brugte funktioner efter min mening.

| | Lister | Menu knapper | Orientation | Swipe | Kodesprog | Sværhedsgrad |
|-------------------|--------------------------------------|--------------------------------------|--------------------------------------|--------------------------|----------------------------|--------------|
| Phonegap | IOS -OK Android -OK | OS -OK Android -OK | OS -OK Android -OK | OS - OK Android -Sløv | HTML,CSS & JS | 2 |
| Xamarin | IOS - kan ikke testes Android -OK | IOS - kan ikke testes Android -OK | IOS - kan ikke testes Android -OK | | C# & XAML | 5 |
| Ionic | OS -OK Android -OK | OS -OK Android -OK | OS -OK Android -OK | | HTML,CSS & Angular (JS) | 4 |
| Mobile Angular UI | | | | | | |

Eftersom jeg kommer med en webudvikler baggrund, er der ingen tvivl om at Phonegap er min favorit. Jeg har givet den en sværhedsgrad på 2, fordi jeg følte det var en nem proces, lige fra installation til at se noget visuelt. Jeg havde dog lidt besvær med afprøve Phonegap appen på mine enheder, lige ind til jeg fandt en app på App Store² og Google Play³.

¹ <http://docs.phonegap.com/plugin-apis/>

² <https://itunes.apple.com/app/id843536693>

³ <https://play.google.com/store/apps/details?id=com.adobe.phonegap.app>

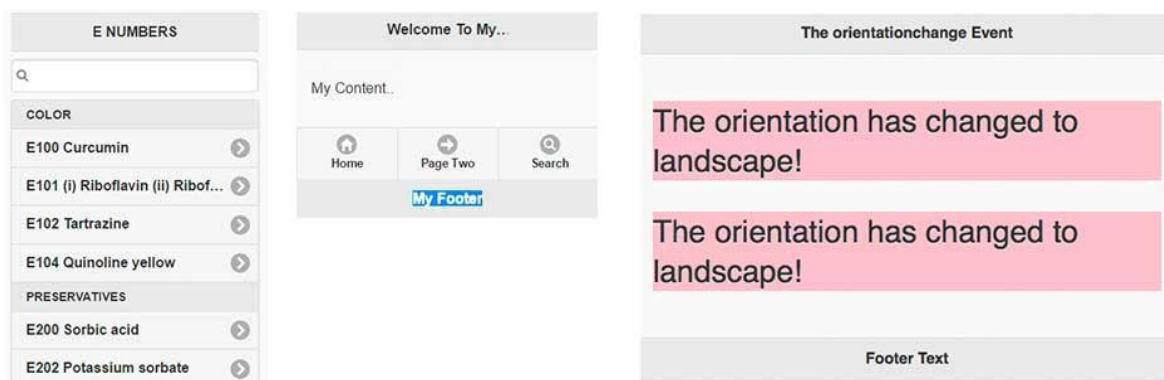
jQuery Mobile

For at kan komme i gang med at teste de forskellige cross-platform frameworks, har jeg valgt at benytte mig af jQuery Mobile⁴. Det gør jeg fordi jeg i forvejen har kendskab til det fra et fritids projekt⁵ og fordi det er nemt at finde dokumentation og eksempler på kode.

jQuery Mobile er en HTML5-baseret brugerplatform, der har til formål at gøre responsive hjemmesider tilgængelige på alle smartphones, tablets og stationære enheder. JQuery Mobile er bygget på HTML5, CSS3 og jQuery (JavaScript). Frameworket tilbyder en masse funktioner, som f.eks. Ajax navigation med side animation, touch events, tema styling og meget mere.

Men da mit projekt ikke omhandler jQuery Mobile, vil jeg henvise til jQuery Mobile⁶ og themeroller⁷.

På nedenstående billeder ses 3 simple jQuery Mobile eksempler. Jeg har valgt at udvikle en applikation for hvert afprøvningsformål, det vil sige at jeg først vil teste hvordan en liste fungerer, derefter menu, skræm orienteringen og evt. swipe, hvis jeg får tid. Jeg tror dette er den bedste måde at finde frem til hvordan de forskellige cross-platform apps reagerer på hver sin enhed.



⁴ <https://jquerymobile.com/>

⁵ <http://stylez.dk/iOSE/>

⁶ <http://jquerymobile.com/>

⁷ <http://themeroller.jquerymobile.com/>

Hvad er Phonegap/Cordova egentlig?

Apache Cordova⁸ tidligere kaldt Phonegap og Apache Callback stammer oprindeligt fra Adobe. Adobe valgte at udgive en open-source version af softwaren ved navn Apache Cordova, og så selv beholde navnet Phonegap. Phonegap og Cordova Api minder rigtig meget om hinanden, men Phonegap er bygget ovenpå Cordova, ligesom Ionic⁹, som jeg senere vil komme meget mere ind på. Cordova giver stor mulighed for at tilføje ekstra plugins¹⁰ til næsten alle platforme, så man kan tilpasse projekt lige efter eget valg. Apache Cordova bliver sponsoreret af bl.a. Adobe, BlackBerry, Google, IBM, Intel, Mozilla og mange andre.

I dag tilbyder Adobe en service "Phonegap Build" mod betaling, hvor man nemt og enkelt transformere sin web app om til native app.

Phonegap og Cordova er en service som omdanner en web applikation til et web view¹¹. Et web view er et native komponent, der viser web indhold i en native applikation, lidt ligesom alle andre web applikationer som vises i en browser. Det kan åbne hyperlinks, loade JavaScript og afspille mediefiler, det kaldes en hybrid applikation. Phonegap tilbyder flere JavaScript API'er der muliggøre at bruge telefonens eller tabellens indbyggede funktioner, som kamera, mikrofon, gyrometer osv. Phonegap er det hul imellem web applikation og den håndholde enhed. Nedenstående billede viser en visuel forståelse af Phonegap.



Billede er fra lynda.com - up and running with phonegap build

Det skal dog lige siges at Phonegap ikke giver adgang til selve design-interfacet, som knapper og faner. Alt dette skal man selv sørge for at style i sin html og css.

Med Phonegap build kan man nemt omdanne sin web app til native cross-platform app, ved at komprimere sin HTML5, CSS3 og JavaScript til en zip fil og så uploade den.

For at benytte sig af phonegap build kræver det en Adobe konto. Når man er logget ind har man enten mulighed for at uploade fra github eller fra en zip. fil. Hvis en zip. fil bliver uploadet korrekt vises der detaljer om dit projekt som ses på nedenstående billede.

⁸ <https://cordova.apache.org/>

⁹ <http://ionicframework.com/>

¹⁰ <http://ionicframework.com/>

¹¹ <http://developer.telerik.com/featured/what-is-a-webview/>

The screenshot shows the PhoneGap Desktop interface for an application named "Test List App". The interface includes a header with the app name and a QR code, a navigation bar with "Builds", "Plugins", "Collaborators", and "Settings", and a table of build records. Below the table are controls for building and logging for iOS, Android, and Windows.

| App ID | Version | Owned by | PhoneGap (iOS / Android / Windows) | Source | Tag / Branch | Commit | Last built (2) |
|---------|---------|--------------------|------------------------------------|---|--------------|---------|----------------|
| 2082153 | n_a | marthero@gmail.com | cli-5.2.0 (3.91 / 4.11 / 3.8.1) | https://github.com/MARTIN-HERO/hybrid.git | master | 82a0ac6 | 2 hours |

Der vil være mulighed for at downloade en apk eller en xap fil til at teste på en android eller en Windows iPhone. Desværre giver IOS en fejl pga. manglende Apple certifikater, dette vil kræve et Apple udvikler medlemskab til \$99¹² om året.

Det har PhoneGap Desktop heldigvis en løsning på. Den opretter en lokal server på din computer, som gør det muligt at teste sine Apps direkte på mobilene. Man kan hente Phonegap developer app fra app store, Google play eller Windows store, ved at installere appen på mobilen/tabletten og benytte sig af samme netværk, giver det dig adgang til at teste dine apps.

Jeg er overrasket over hvor nemt det er at afprøve sine apps i denne service og endnu mere chokeret over hvor meget det minder om en native app.

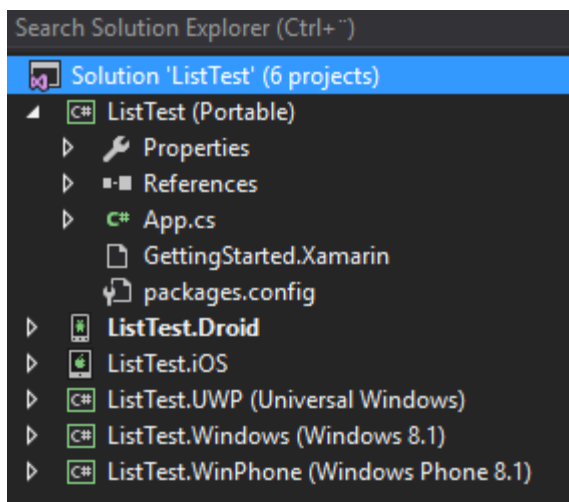
¹² <https://developer.apple.com/support/compare-memberships/>

Microsoft Xamarin

Xamarin har lange haft et samarbejde med Microsoft, men er nu blevet opkøbt. Det virker som om at Microsoft vil ikke for solgt nok Windows iPhones pga. af manglende apps, derfor tror jeg de satser på et marked for cross-platform løsninger. Jeg forstiller mig at de håber på der bliver udviklet flere apps til Windows Phones, hvis de tilbyder .Net udviklerne denne service.

Det gode ved Xamarin og for andre cross-platform app frameworks, er at udviklerne kun skal fokusere på at skrive kode og opdatere i et projekt af gangen, hver gang de udvikler til IOS, Android og Windows Phone. Så en kombination af Xamarin og Visual Studio giver alle .Net udviklere muligheden for at lave en samlede app løsning i deres ynglings IDE, hvor der kan styles, teste, debugge og publicere.

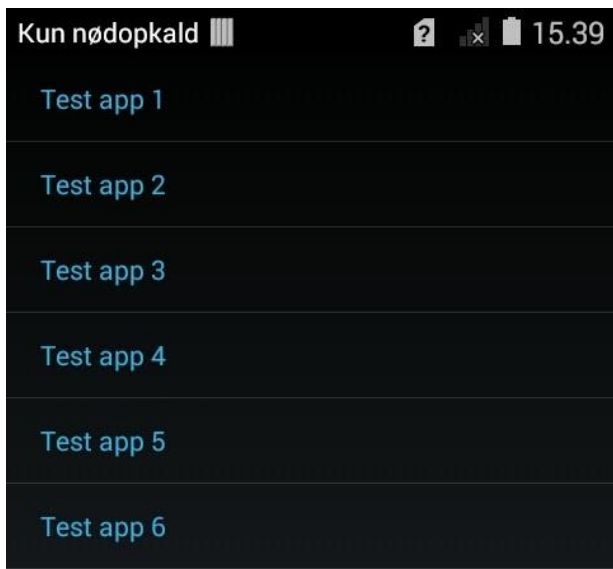
Det kræver en del opdateringer, tid og plads at installere Xamarin. Jeg opretter et simpelt projekt under C#->Cross-platform->Blank App (Xamarin.Form Portable). Her kan man se i Solution Explorer (Se nedenstående billede) at alle 3 platforme bliver samlet i et projekt.



Xamarin virker meget som et framework med mange muligheder, forstået på den måde, at man kan gå i dybden med lige præcis den enhed man ønsker at udvikle til.

Efter at jeg har brugt halvanden dag på installation som fylder 30 GB, samt oprettelse af liste app, må jeg konstatere at dette framework har mange fejl og mangler. Allerede første gang jeg ønsker at teste projektet på min android enhed, får jeg denne fejl i installationen "Xamarin.Android.Support.Design". Jeg googler det, og finder hurtigt fra til at mange andre har samme problem. Her viser det sig, at man selv til opdatere flere fil pakker, som derfor gør det til en langt proces fra installation til at man ren faktisk kan afprøve sine test projekter.

Fordi jeg er i besiddelse af en Android telefon, er der dog ingen problem med at afprøve mit projekt. Ved blåt at tilslutte telefonen via usb, åbner projektet op automatisk efter kort tid, når men kører projektet i Visuel Studio. På nedenstående billede ses et skræm dump fra min Android telefon.



Næste afprøvning jeg ønskede at udføre, var i en Windows Phone Emulator, da jeg desværre ikke er ejer en Windows iPhone. Men her støder jeg på en ny fejl som ses på nedenstående billede.

| | Code | Description | Project | File | Line | Suppression St... | Filter |
|---|---------|---|-------------------|------|------|-------------------|--------|
| ✖ | DEP0001 | Unexpected Error: -1988945906 | ListTest.WinPhone | | | | |
| ⚠ | IDE0006 | Error encountered while loading the project. Some project features, such as full solution analysis for the failed project and projects that depend on it, have been disabled. | ListTest | | 1 | Active | |

Jeg føler lidt det er et most at eje samtlige enheder man ønsker at udvikle til, samt et Apple udvikler abonnement ¹³. Det er både hurtigere at teste på rigtige enheder frem for emulatorer, samtidig med at man får den helt rigtige følelse af hvordan appen fungerer på den udvalgte enhed.

¹³ <https://developer.apple.com/support/compare-memberships/>

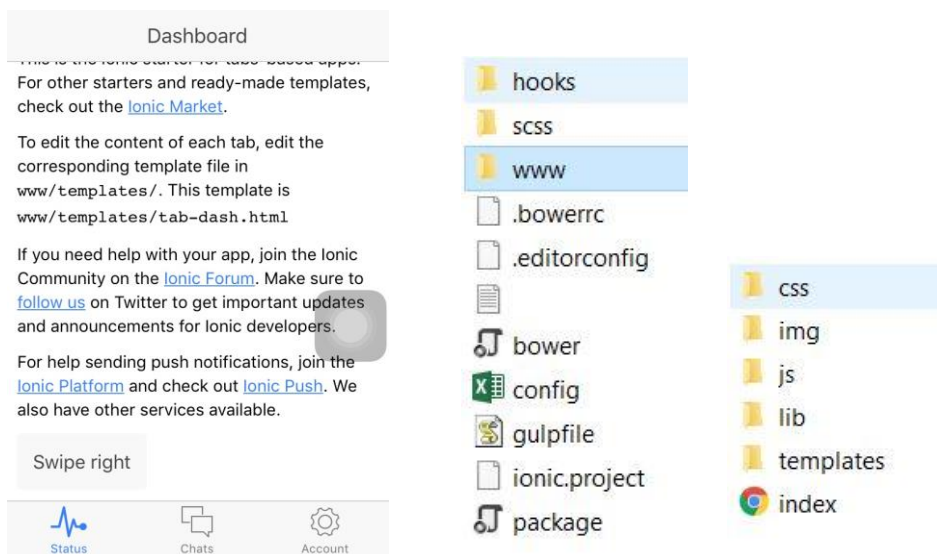
Ionic

Ionic¹⁴ er et open-source SDK projekt til at udvikle cross-platform apps. Som jeg nævnte tidligere, så er Ionic bygget ovenpå Apache Cordova og Angular JS¹⁵. Ionic tilbyder også en masse værktøjer og services til web udvikling baseret på HTML, CSS, JS og Sass¹⁶. De tilbyder bl.a. et prototype værktøj¹⁷, hvor man nemt kan designe sin app ved hjælp af en drag and drop funktion.

Det er nemt komme i gang med Ionic, det kræver dog lidt command line kendskab. Først i processen installer man Nodejs¹⁸, som er et open source cross-platform runtime system. Det at systemet kører i runtime betyder bare at det kører samtidig med applikationen. Node.js er ikke et JavaScript framework selvom mange af modulerne er skrevet i JavaScript.

Når Nodejs er installeret, er man klar til at installere Cordova og Ionic via. commando prompten. Det gøres meget simpelt ved at indtaste følgende tekst " npm install -g cordova ionic". Efter 3-4 min. er man klar til at oprette sit første projekt.

Hvis man navigere til den ønsket placering, hvor man ønsker projekt skal være, så skriver man bare "ionic start Menu tabs" i prompten. Denne command line opretter så et tabs projekt med mappenavnet Menu. På nedenstående billeder ses mappestrukturen og selve web appen.



I løbet af ingen tid har man et test projekt kørende. Under www som ses på det sidste billede, ligger alt HTML, CSS og JavaScripten. Her er det en klar fordel at herske Angular JS¹⁹, da det er frameworkets måde binde controller sammen med ens HTML filer.

¹⁴ <http://ionicframework.com/>

¹⁵ <https://angularjs.org/>

¹⁶ <http://sass-lang.com/guide>

¹⁷ <https://creator.ionic.io/app/designer/bbd66bd5ee49>

¹⁸ <https://nodejs.org/en/>

¹⁹ <https://angularjs.org/>

Konklusion

Cross-platform universet er meget større og mere avanceret end jeg først forstillede mig. Der er ingen tvivl om at denne udviklingsmetode er en nem og billig måde at komme i gang på. Jo flere gode apps som kommer i omløb, desto mere salg af håndholde enheder er der for de store virksomheder. Jeg forstår nu godt hvorfor bl.a. Microsoft har købt Xamarin og Adobe er gået Phonegap build vejen.

Ligesom jeg også forstår hvorfor der ikke er noget framework som er bedre end et andet. De forskellige frameworks er design og udviklet til specifikke programmører, alt efter det kodesprog de hersker i. Derfor var det bl.a. nemt for mig at komme i gang med Phonegap, fordi jeg har et stort kendskab til web programmering, fra mit tidligere arbejde.

Litteraturliste

<http://www.lynda.com/Creative-Cloud-tutorials/What-PhoneGap/134924/170242-4.html>

<https://developer.apple.com/support/compare-memberships/>

<https://build.phonegap.com/>

<http://developer.telerik.com/featured/what-is-a-webview/>

<https://www.linux.com/news/10-best-free-mobile-application-development-frameworks-support-android>

<http://phonegap.com/getstarted/>

http://www.w3schools.com/jquerymobile/jquerymobile_list_views.asp

http://www.w3schools.com/jquerymobile/tryit.asp?filename=tryjqmob_navbars

http://www.w3schools.com/jquerymobile/jquerymobile_events_orientation.asp

http://www.w3schools.com/jquerymobile/jquerymobile_events_touch.asp

<https://www.youtube.com/watch?v=I9kjeKpsoj0>

<http://www.lynda.com/Development-Tools-tutorials/Visual-elements-explained-Page-view-cell-layout/449034/483499-4.html>

<https://www.version2.dk/artikel/microsoft-koeber-xamarin-og-sikrer-sig-brobygningsvaerktoej-til-ios-og-android-623597>

<https://www.xamarin.com/forms>

<http://stackoverflow.com/questions/36489766/cannot-upgrade-nuget-packages-for-xamarin-android>

<https://forums.xamarin.com/discussion/56499/empty-android-support-23-0-1-3-directories>

<http://stackoverflow.com/questions/34199003/xamarin-forms-could-not-be-updated>

<https://www.addthis.com/blog/2014/10/27/7-things-to-consider-when-making-ios-and-android-apps-with-cordova-or-phonegap/#.V0Mei5GLR2R>

<https://cordova.apache.org/>

https://en.wikipedia.org/wiki/Apache_Cordova

<https://en.wikipedia.org/wiki/Node.js>

