



**HoGent**

Faculteit Bedrijf en Organisatie

Vergelijkende studie en proof-of-concept in functie van een doelbewuste keuze tussen een cross platform mobiele applicatie en een mobiele website

Yannick Van Hecke

Scriptie voorgedragen tot het bekomen van de graad van  
professionele bachelor in de toegepaste informatica

Promotor:  
Koen Hoof  
Co-promotor:  
Jeroen Gevenois

Instelling: Politiezone Gent - Dienst ICT

Academiejaar: 2016-2017

Tweede examenperiode



Faculteit Bedrijf en Organisatie

Vergelijkende studie en proof-of-concept in functie van een doelbewuste keuze tussen een cross platform mobiele applicatie en een mobiele website

Yannick Van Hecke

Scriptie voorgedragen tot het bekomen van de graad van  
professionele bachelor in de toegepaste informatica

Promotor:  
Koen Hoof  
Co-promotor:  
Jeroen Gevenois

Instelling: Politiezone Gent - Dienst ICT

Academiejaar: 2016-2017

Tweede examenperiode

## **Samenvatting**

# Voorwoord

# Inhoudsopgave

<b>1</b>	<b>Inleiding</b>	<b>5</b>
1.1	Stand van zaken . . . . .	5
1.2	Probleemstelling en Onderzoeksvragen . . . . .	6
1.3	Opzet van deze bachelorproef . . . . .	7
<b>2</b>	<b>Methodologie</b>	<b>8</b>
2.1	Literatuuronderzoek . . . . .	8
2.2	Vastleggen casus . . . . .	8
2.3	Framework kiezen voor cross platform mobiele applicatie . . . . .	8
2.4	Framework kiezen voor responsive webapplicatie . . . . .	9
2.5	Opzetten REST api . . . . .	9
2.6	De ontwikkeling van een cross platforme mobiele applicatie . . . . .	9
2.7	Ontwikkelen responsive webapplicatie . . . . .	10
2.8	Vergelijken van de tijd die nodig is om beide vormen van toepassingen te ontwikkelen . . . . .	10
2.9	Metten van gegevensverbruik . . . . .	10
2.10	Metten van gebruiksvriendelijkheid . . . . .	10
2.11	Extra kosten bij de realisatie en publicatie . . . . .	11
2.12	Besluitvorming . . . . .	11
<b>3</b>	<b>Framework cross platforme mobiele applicatie</b>	<b>12</b>
3.1	Twee grote categorieën . . . . .	12
3.2	Architectuur van de cross platform mobiele applicatie . . . . .	12
3.3	Voordelen van de gekozen architectuur . . . . .	13
3.3.1	Ontwikkeling en project structuur . . . . .	13
3.3.2	User Interface . . . . .	13
3.4	Nadelen van de gekozen architectuur . . . . .	13
3.5	Kosten . . . . .	13
3.5.1	Kosten voor testen . . . . .	13
3.5.2	Kosten voor publicatie . . . . .	14

<b>4</b>	<b>Ontwikkeling cross platforme mobiele applicatie</b>	<b>15</b>
4.1	Analyse van het project . . . . .	15
4.2	Ontwikkeling van de ASP.NET MVC Web api . . . . .	15
4.3	Ontwikkeling van de cross platforme mobiele applicatie . . . . .	15
4.3.1	Android . . . . .	15
4.3.2	iOS . . . . .	15
4.3.3	windowsphone . . . . .	15
<b>5</b>	<b>Conclusie</b>	<b>16</b>

# Hoofdstuk 1

## Inleiding

### 1.1 Stand van zaken

In de vakliteratuur zijn er vandaag diverse artikelen te vinden over een vergelijkende studie tussen mobiele applicatie en responsive webapplicaties. Meeste onderzoeken die tot nu toe gevoerd zijn, zijn echter een beperkte lijst van kenmerken die aangeven wanneer men het best voor welke optie kiest. Dit zonder dat hier wetenschappelijk onderzoek aan vooraf ging.

Toch zijn deze wetenschappelijke artikelen beschikbaar. Zo onderzochten Paulo R. M. de Andrade (2015) in het artikel 'CROSS PLATFORM APP A COMPARATIVE STUDY' de voornaamste reden om te kiezen voor enerzijds de mobiele applicatie en anderzijds de responsive website.

De belangrijkste redenen uit het onderzoek om te kiezen voor een cross-platform mobiele applicatie zijn de volgende:

- Performantie
- User Interfaces in de lijn van het besturingssysteem
- Distributie via de app stores
- Push notificaties

In de vakliteratuur haalt men voor de hoofdzakelijkste redenen om te kiezen voor een responsive webapplicatie, de volgende aan:

- Eenvoudige ondersteuning voor meerdere systemen
- Het publiceren van de toepassing



## 1.2 Probleemstelling en Onderzoeksvragen

Momenteel worden er bij de ICT-afdeling van de Gentse Politie aan verschillende projecten gewerkt waarbij het mobiel beschikbaar maken van interne gegevens een vereiste is. Om dit naar een gebruiksklaar eindproduct voor een mobiel apparaat om te zetten, bestaan er vandaag de twee opties.

Enerzijds kan er gekozen worden voor een cross-platform mobiele applicatie. Dit houdt in dat men binnen 1 ontwikkelomgeving een applicatie ontwikkelt die bruikbaar is op de 3 grote mobiele platformen: Android van Google, iOS van Apple en Windows Universal Platform van Microsoft. Hierbij wordt er code gedeeld tussen de 3 platformen.

Het weergeven van de data gebeurt op een platform-specifieke manier. Dit omdat men binnen de verschillende mobiele besturingssystemen werkt met andere UI elementen om de data in te tonen.

Anderzijds bestaat de mogelijkheid om te kiezen voor een responsive webapplicatie. Dit wil zeggen dat de inhoud van de webapplicatie aangepast wordt, naargelang het scherm waarbij men de mobiele website bekijkt. Zo wordt er op een smartphone mogelijks minder informatie weergegeven dan indien men de website bekijkt op een computer of wordt in het geval van de smartphone de positie en afmetingen van de elementen aangepast.

In dit werk wordt onderzocht in welke gevallen de webapplicatie de beste optie is alsook in welke gevallen de mobiele applicatie een beter alternatief vormt.

Hierbij zullen volgende onderzoeksvragen beantwoord worden:

- Bij welke vorm van applicatie wordt het snelst resultaat behaald?
- Welke optie is het meest gebruiksvriendelijk?
- Welke optie is het efficiëntst?
  - Welke optie verbruikt de meeste hoeveelheid gegevens?
  - Welke optie geeft de eindgebruiker het snelst de gevraagd informatie?
- Welke optie is het veiligst?
- Welke optie kan het snelst ontwikkeld worden?

## 1.3 Opzet van deze bachelorproef

In functie van de snelheid van de ontwikkeling van beide producten, moet er een ontwikkelomgeving gekozen worden. Om een cross-platform mobiele applicatie te ontwikkelen, bestaat de keuze uit 2 technologieën. Enerzijds zijn er ontwikkelomgeving met C# als achterliggende programmeertaal. Hierbij wordt de UI gedefinieerd in xml voor Android, xaml van windowsphone en Universal Windows Platform en storyboards voor iOS. Hierbij wordt de oorspronkelijke interactie en structuur van de UI en de logica van het mobiele besturingssysteem behouden. Visual Studio en Xamarin zijn 2 voorbeelden die gebaseerd zijn op deze technologie. Anderzijds kan men ook opteren voor een cross-platforme mobiele applicatie te bouwen op basis van html, css en javascript. Hierbij wordt er eerst een webservice van het te hosten project gemaakt, om vervolgens deze webservice op te starten in een emulator. De ontwikkelomgeving PhoneGap is op deze manier van werken gebaseerd.

In dit onderzoek zal gekozen worden voor de eerste manier van werken, nl. de ontwikkelomgeving met C# als achterliggende programmeertaal. Vanwege de huidige kennis van het ontwikkelen van cross-platforme mobiele applicatie geniet de eerste optie de voorkeur.

Nadat de gekozen ontwikkelomgeving vastligt, wordt er aan de hand van de functionele requirements beslist welke de minimale versies van de besturingssystemen zijn. Hierbij wordt getracht het aantal ondersteunende apparaten zo hoog mogelijk te houden. Tijdens het onderzoek wordt, zoals in de onderzoeksvragen reeds vermeld, ook rekening gehouden met de tijd en kosten die de ontwikkeling en publiek stellen van beide applicaties met zich meebrengen.

Vervolgens wordt de gebruikte methodologie verklaard, waarna de casus uitgelegd wordt. Deze casus is de basis die zal gebruikt worden in het onderzoek naar efficiëntie en gebruiksvriendelijkheid van de cross-platforme mobiele applicatie en de responsive webapplicatie.

Nadat de casus duidelijk verklaard is, worden de gemaakte applicaties verklaard aan de hand van de geschreven code. Vervolgens worden de resultaten getoond en wordt de conclusie die uit het onderzoek voortvloeit, medegedeeld. Deze conclusie wordt ook gebruikt door de Gentse Politie in hun keuze tussen een mobiele applicatie of een responsive website.

Na de conclusie worden er nog richtlijnen voor toekomstig onderzoek gegeven in deze materie.

# Hoofdstuk 2

## Methodologie

### 2.1 Literatuuronderzoek

Ter voorbereiding van deze vergelijkende studie werd er een literatuuronderzoek verricht. Dit hield in dat er op internet gezocht werd naar verschillende wetenschappelijke artikelen over voorgaande vergelijkende studies tussen cross platforme mobiele applicatie en responsive website. Dit literatuuronderzoek heeft al voornaamste reden het geven van een stand van zaken in het ontwikkelen van cross-platforme mobiele applicaties en responsive webapplicatie.

### 2.2 Vastleggen casus

De eerste stap in het onderzoek naar de vergelijkende studie tussen een cross-platforme mobiele applicatie en een responsive website, was het vastleggen van de casus. Dit bepaalde de functionele requirements van beide applicaties en gebeurde in overleg met de co-promotor. De casus omvatte een toepassing rond personeelsgegevens binnen de gentse politie, waarbij de gegevens momenteel enkel intern beschikbaar zijn. Het doel van de casus is om deze, mits de nodige beveiliging, beschikbaar te stellen van het personeel dat niet altijd met het intern netwerk verbonden is. Hierbij word er vooral gedacht aan mensen op het terrein.

### 2.3 Framework kiezen voor cross platform mobiele applicatie

Omdat vanuit de stage reeds gekozen werd voor de ontwikkelomgeving van Microsoft en dit hierdoor reeds een vertrouwde manier van werken was, heb ik gekozen om de cross platform mobiele applicatie te ontwikkelen met Visual Studio 2015. Hierbij

wordt er gebruik gemaakt van een apart deelproject binnen de applicatie waarin code gemeenschappelijk wordt gemaakt voor de 3 platform-specifieke applicaties. Op deze manier kan men code hergebruiken, waardoor de efficiëntie van de geschreven code stijgt.

### 2.4 Framework kiezen voor responsive webapplicatie

Het framework, dat voor de responsive webapplicatie gebruikt werd, is het MVC-framework van Microsoft. Hierdoor kan de ontwikkeling van de webapplicatie ook in Visual Studio gebeuren, hetgeen voor de vergelijking van de ontwikkeltijd een eerlijker resultaat opleverde, dan wanneer men zich nog dient in te werken in nieuwe materie.

### 2.5 Opzetten REST api

Het opzetten van de REST api was de volgende stap in het onderzoeksproces. Aanvankelijk werd er gekozen om eerst de basisfunctionaliteiten te implementeren en pas nadien werd de authenticatie toegevoegd. Dit zorgt ervoor dat de data niet ongeoorloofd beschikbaar is voor derden. Hierbij dient men zich eerst te authenticeren aan de hand van de gebruikersnaam en het wachtwoord van het gebruikersaccount binnen de Politiezone Gent. Indien men zich succesvol geauthenticeerd heeft bij het domein aan de hand van de Active Directory, krijgt men van de REST api een token terug. De geldigheid van deze token kan men vrij kiezen. Deze token is verplicht bij te voegen bij alle requests die verstuurd worden naar de server. Het testen van de REST api zal eerst gebeuren aan POSTMAN, een REST-client. Zo is men zeker dat de REST-api correct werkt, alvorens over te gaan te het ontwikkelen van de mobiele applicatie.

### 2.6 De ontwikkeling van een cross platforme mobiele applicatie

Nadat het testen van de REST api voltooid is, begint de ontwikkeling van de cross platforme applicatie. Hierbij is het de bedoeling om de code die gemeenschappelijk is voor de 3 deelprojecten, nl. het ophalen van gegevens uit de webservice en deze op te slaan in een lokale databank, alsook het mappen naar objecten van deze gegevens af te zonderen van de platform-specifieke deelprojecten in een gemeenschappelijk deelproject. Op deze manier wordt de ontwikkeltijd gereduceerd ten opzichte van het ontwikkelen van 3 afzonderlijke applicaties. Het weergeven van de opgehaalde data wordt, samen met de ontwerp van de schermen voor de platforme-specifiek user interfaces, in de

platform-specifieke projecten gencodeerd. Dit omdat er geen uniforme manier bestaat om de user interfaces te voorzien binnen de gekozen ontwikkelomgeving.

## **2.7 Ontwikkelen responsive webapplicatie**

De volgende stap in deze vergelijkende studie tussen een cross platforme mobiele applicatie en responsive webapplicatie is het ontwikkelen van de responsive webapplicatie. Hierbij wordt dezelfde databank gebruikt die de webservice van de cross platforme mobiele applicatie. Verder wordt ook het domeinmodel van de webservice en de mobiele applicatie overgenomen. Hierna worden de controllers geïmplementeerd met authenticatie en autorisatie aan de hand van de Active Directory. Dit om vervolgens het responsive design van de webpagina's af te werken.

## **2.8 Vergelijken van de tijd die nodig is om beide vormen van toepassingen te ontwikkelen**

Wanneer beide vormen van de toepassing ontwikkeld zijn, zal er gekeken worden naar tijd die nodig was om deze tot stand te brengen. Dit is overigens een onderzoeksvraag van de casus en zal gebeuren aan de hand van een versiebeheersysteem. Hierbij de tijd tussen het verschil tussen de eerste en de laatste commit zal berekend worden.

## **2.9 Meten van gegevensverbruik**

Het meten van de verbruikte hoeveelheid gegevens zal gebeuren door te bekijken van de Debugger tijdens het uitvoeren van beide applicaties. Dit zal een vooraf vast aantal keer gebeuren, hetzij eerst zonder en erna met caching van informatie. Nadien zal dit proces herhaalt worden bij de responsive website. Indien alle nodige gegevens verzamelt zijn, zal er een conclusie getrokken worden over welke vorm van applicatie het meest zuinig omgaat met het internetgebruik.

## **2.10 Meten van gebruiksvriendelijkheid**

Nadat het gegevensverbruik gemeten is, zal ook de gebruiksvriendelijkheid gemeten worden aan de hand van enkele vragen. Zo zal er gevraagd worden welke vorm van applicatie er het meest gebruiksvriendelijk is en waarom dit het geval.

## **2.11 Extra kosten bij de realisatie en publicatie**

De kosten die nodig zijn voor de realisatie en publicatie van beide applicaties zijn ook een factor die meetelt in de vergelijking van enerzijds de cross-platforme mobiele applicatie en anderzijds de responsive website. Hierbij wordt hoofdzakelijk gedacht aan de kosten die de publicatie in de app stores met zich meebrengt, alsook de kosten voor de hosting van enerzijds de webservice en anderzijds de webapplicatie.

## **2.12 Besluitvorming**

Eens alle bovenvermelde gegevens bekend zijn, zullen met elkaar vergeleken worden om zo tot een gefundeerde lijst van argumenten te komen. Deze lijst zal vermelden in welke gevallen de cross-platforme mobiele applicatie de betere oplossing voor de klant is. Overigens wordt er ook een lijst met karakteristieken voor de responsive webapplicatie bekend gemaakt.

## **Hoofdstuk 3**

# **Framework cross platforme mobiele applicatie**

### **3.1 Twee grote categorieën**

Ter voorbereiding naar het vergelijkende onderzoek tussen enerzijds de cross platforme mobiele applicatie en anderzijds de responsive website, is er voor beide mogelijke manier gezocht om deze vormen van applicaties te ontwikkelen. Voor de cross-platforme mobiele applicatie zijn deze mogelijke vormen enerzijds een methode die zeer sterk aansluit bij het ontwikkelen van html-website. Hierbij wordt de html code in combinatie met css voor de opmaak en JavaScript voor de logica door de compiler omgezet naar een cross- platforme mobiele applicatie. Op deze laatste manier wordt er gedurende dit werk niet verder op ingegaan.

### **3.2 Architectuur van de cross platform mobiele applicatie**

Voor de opbouw van de cross platforme mobiele applicatie is er geopteerd voor een architectuur waarbij men zoveel mogelijk code gemeenschappelijk kan hergebruiken. Concreet houdt dit in de men in het Shared-project zowel de definitie van de domeinklassen, als de code die verantwoordelijk is voor het ophalen van de data uit de REST-service en de mapping definieert.

## **3.3 Voordelen van de gekozen architectuur**

### **3.3.1 Ontwikkeling en project structuur**

Het grote voordelen van de architectuur van de cross platforme mobiele applicatie, nl. specifieke deelprojecten voor de afzonderlijke mobiele besturingssystemen en het gemeenschappelijk stellen van bepaalde delen van de programmacode, heeft als voorname voordeel voor de ontwikkelaar dat binnen de gekozen ontwikkelomgeving, nl. Visual Studio, ervoor gekozen heeft om de structuur voor de mobiele applicaties van de deelprojecten over te nemen van de oorspronkelijke structuur van de Android-, iOS- en windowsphone-applicatie. Deze manier van werken vereenvoudigt de overstap van de oorspronkelijke ontwikkeling, Android Studio voor Android en xcode voor iOS, naar Visual Studio. Ook de syntax van de code van de deelprojecten werd hierbij overgenomen en vertaald naar C#.

### **3.3.2 User Interface**

## **3.4 Nadelen van de gekozen architectuur**

## **3.5 Kosten**

De ontwikkeling van een cross platforme mobiele applicatie is een project waaraan de nodige kosten verbonden zijn. Hieronder volgt een oplist van de kosten per fase in het softwareontwikkelingsproces.

### **Kosten voor de ontwikkeling**

Reeds bij de ontwikkeling van de cross platforme mobiele applicatie zijn mogelijke kosten. Zo dient men een licentie op een ontwikkelomgeving aan te schaffen.

### **3.5.1 Kosten voor testen**

Voor een applicatie in de applicatiewinkel van een mobiel besturingssysteem terecht komt, dient deze eerst uitvoering getest te worden. Op deze manier kunnen de mogelijke fouten uit de software gehaald. Fouten in de applicatie kunnen ervoor zorgen dat de applicatie onverwacht beëindigd wordt of vastloopt. Dit beëindigen of vastlopen kan tot gevolg hebben dat de applicatie slechte reviews krijgt in de applicatiewinkels en bijgevolg minder downloads en inkomsten voor de ontwikkelaar opleveren.

Indien men ervoor kiest om voor de drie grote platforme te ontwikkelen, heeft men naast de kost van de licentie voor Visual Studio of een ander ontwikkelomgeving,



ook de kost van een computer waar het besturingssysteem macOS op geïnstalleerd is in combinatie met Xcode, de ontwikkelomgeving van Apple. Xcode is een IDE<sup>1</sup> waarin men applicatie kan ontwikkelen voor iOS, macOS, tvOS en watchOS, de besturingssystemen voor respectievelijk iPhone en iPad, iMac en de MacBook in alle versie, de Apple TV en de Apple Watch. De ontwikkelomgeving is gratis beschikbaar de App Store in macOS xco.

#### 3.5.2 Kosten voor publicatie

De keerzijde van deze keuze is, dat men echter wel nog kennis nodig heeft van Android, iOS en Windows Phone om deze applicatie te kunnen ontwikkelen. Zo moeten er nog steeds 3 applicaties ontwikkeld worden, hetzij wel in een afgeslankte versie. Dit in tegenstelling tot de responsive webapplicatie waarbij men maar 1 keer de gebruikers-interface dient te definiëren.

Verder dient ook men ook de applicatie voor elk mobiel besturingssysteem te publiceren in de appstores van de verschillende mobiele besturingssystemen. Hieraan zijn ook ofwel eenmalige ofwel jaarlijkse kosten aan verbonden. Zo dient men voor de publicatie in de App Store van Apple zich eerst inschrijven voor het Apple Developer Program van Inc (2017). Deze inschrijving kost 99 Amerikaanse dollar per jaar. Verder kunnen bedrijven of organisaties zich hier ook bij aansluiten, om zo meerdere ontwikkelaars aan te laten sluiten op 1 ontwikkelaarsaccount voor de App Store. Dit bespaart uiteraard op de kosten.

Voor Android dient men bij de publicatie van de app Google (2017) eenmalig 25 Amerikaanse dollar betalen voor de inschrijving. Overigens zijn er voor de publicatie van de Android-app geen kosten gevonden op de ontwikkelaarswebsite van Android. Verder kan men aan de hand van Developer Console het aantal downloads van de app, het aantal gerapporteerde crashes en de gemiddelde rating die de gebruikers gaven bekijken. Aan de hand van de recensies die de gebruikers op de app, kan de ontwikkelaar de feedback met eventuele verbeterpunten bekijken.

Indien men de applicatie wenst te publiceren in de 'Microsoft Marketplaces' van Microsoft (2017) zoals de Windows Store, Office Store, Azure Marketplaces en nog meer aan te kondigen winkels, dient men voor een individueel account eenmalig ongeveer 19 Amerikaanse dollar te betalen aan Microsoft. Voor bedrijfsaccounts komt deze kost neer op +/- 99 dollar.

---

<sup>1</sup>Integrated Develop Environment

## **Hoofdstuk 4**

# **Ontwikkeling cross platforme mobiele applicatie**

### **4.1 Analyse van het project**

### **4.2 Ontwikkeling van de ASP.NET MVC Web api**

### **4.3 Ontwikkeling van de cross platforme mobiele applicatie**

#### **4.3.1 Android**

#### **4.3.2 iOS**

#### **4.3.3 windowsphone**

## **Hoofdstuk 5**

### **Conclusie**

# Bibliografie

Google (2017). Get started with publishing | android developers.

Inc, A. (2017). Apple developer program.

Microsoft (2017). Register as an app developer.

Paulo R. M. de Andrade, A. B. A. (2015). Cross platform app a comparative study.  
*International Journal of Computer Science & Information Technology (IJCSIT)*,  
7(1):33–40.

## **Lijst van figuren**

## **Lijst van tabellen**