SCHUDDINCK | Firebase | 15 augustus 2017

Firebase

Nieuwe technologie voor applicaties en proof of concept

# Voorwoord

Na de opkomst van de eerste personal computer in de jaren tachtig van de vorige eeuw is de informatie technologie er in een ijltempo op vooruitgegaan.

Onze ouders behoorden tot de eerste generatie op personal computers op kantoor en thuis in gebruikt te nemen. In het begin waren personal computers grote apparaten die werkten met floppydisks. Men programmeerde met Cobol en Pascal en e-mail bestond nog niet. Pas in de jaren negentig werd het internet voor het grote publiek toegankelijk gemaakt en vanaf toen was de IT-trein niet te stoppen.

Mijn generatie zag de opkomst van computerspellen, sociaalnetwerksites zoals Netlog en Facebook maar ook portaalsites voor bijvoorbeeld scholen. Tijdens mijn tienerjaren was er vooral twijfel of al dat “computerspelgedoe” wel goed was voor kinderen. Nu weten we al dat het kinderen kan helpen met het sneller maken van keuzes en het meer inzicht verwerven in dingen. Of het goed is elke dag voor een computerscherm te zitten, is dan weer een andere vraag. Want tegenwoordig zijn we voortdurend op een scherm aan het kijken. Is het geen computerscherm dan is het wel een tablet of smartphone. En nu de roamingkosten afgeschaft zijn in de Europese Unie staat er echt geen rem meer op. Oma kan elke dag onze vakantie volgen. Of hoe citytrippen whatsappen is geworden.

Wij jongeren kunnen ons geen leven meer inbeelden zonder apps. Mijn interesse voor het ontwikkelen van apps speelde een grote rol in de uiteindelijke keuze voor mijn studierichting. De keuze van mijn stage was dan ook voor de hand liggend.

Tot slot nog een woord van dank aan Hogeschool Gent en mijn stagebegeleiders die mij de kans boden om een stage te volgen in een mobile app development agency.

Alsook wil ik mijn familie en vrienden bedanken voor hun steun.

# Samenvatting

De opkomst van smartphones en de bijhorende apps heeft een revolutie ontketend. Apps zijn big business geworden. Echter het ontwikkelen hiervan was tot nu toe een tijdrovend, moeizaam en duur proces. Firebase werd dan ook door Google gecreëerd om snel kwaliteitsvolle applicaties te ontwikkelen zonder verplicht te zijn om een eigen backend te schrijven. Het biedt talloze andere mogelijkheden van het ontwikkelen tot het groeien van de app.

De interesse in Google Firebase werd aangewakkerd tijdens de stage en de nieuwsgierigheid om ermee te werken en te zien hoe het zijn weg vindt in de bedrijven heeft geleid tot de keuze van dit onderwerp.

In eerste instantie werd de door Firebase aangeboden documentatie beknopt uitgelegd en vergezeld van screenshots die zorgen voor een visuele voorstelling.

Vervolgens werd er een proof-of-concept gemaakt om de performantie, snelheid van creatie en gebruiksvriendelijkheid na te gaan. De ervaring leerde dat Firebase haar beloftes waarmaakt

In een vergelijkende studie werd nagegaan of het mogelijk is een standaard applicatie te ontwikkelen zonder Firebase door het opzoeken van alternatieven voor de verschillende componenten van Firebase. Hieruit bleek dat het schrijven van een eigen backend of het gebruik maken van alternatieven enerzijds veel ervaring zou vergen en anderzijds veel tijd en geld zou kosten.  
De voor- en nadelen van het schrijven van een applicatie met en zonder Firebase werden toegelicht. Het gebruiksgemak, de snelheid, de lage kostprijs en een groeiende Firebase community pleitten allen in het voordeel van Firebase.

Tot slot werden de bedrijven bevraagd. Hieruit bleek dat Firebase onder de bevraagde bedrijven al goed gekend was maar nog niet door allen effectief geïmplementeerd werd. De meesten waren enthousiast over de gebruiksvriendelijkheid, de snelle opbouw, de lage instapdrempel en kostprijs en het om deze redenen aan klanten en andere bedrijven zouden aanraden.

Firebase biedt heel wat voordelen. Denk maar aan de tijdswinst, de gebruiksvriendelijkheid, de lage kostprijs, de lage instapdrempel, de komende updates, het gemakkelijk onderhoud en een betere controle over de applicatie.   
Extra voordelen die in de kijker gezet moeten worden zijn toch wel dat een junior ontwikkelaar geen moeite heeft om met Firebase aan de slag te gaan en dat kleine applicaties voor bijvoorbeeld beurzen heel snel opgezet kunnen worden.   
Uit dit alles kan men afleiden dat Firebase meer en meer gebruikt zal worden door de Belgische ontwikkelaars van mobiele applicaties.   
Het valt te verwachten dat er met Firebase een waardige concurrent bijgekomen is op de markt voor mobiele applicaties.

Inhoudstafel

[1. Inleiding 1](#_Toc490594468)

[2. Wat is Firebase? 2](#_Toc490594469)

[2.1. Beschrijving Firebase 2](#_Toc490594470)

[2.2. Gebruik Firebase 2](#_Toc490594471)

[3. Componenten van Firebase 3](#_Toc490594472)

[3.1. Analyse 3](#_Toc490594473)

[3.1.1. Firebase Analytics 3](#_Toc490594474)

[3.1.2. Onderdelen van Firebase Analytics: 3](#_Toc490594475)

[3.2. Ontwikkeling 5](#_Toc490594476)

[3.2.1. Authentication (iOS, Android, Web, C++, Unity) 5](#_Toc490594477)

[3.2.2. Cloud functions (iOS, Android, Web, C++, Unity) 7](#_Toc490594478)

[3.2.3. Cloud Messaging 7](#_Toc490594479)

[3.2.4. Crash Reporting (iOS, Android) 7](#_Toc490594480)

[3.2.5. Hosting 9](#_Toc490594481)

[3.2.6. Realtime Database 9](#_Toc490594482)

[3.2.7. Storage 10](#_Toc490594483)

[3.2.8. Test Lab 10](#_Toc490594484)

[3.3. Groeimogelijkheden 10](#_Toc490594485)

[3.3.1. Acquisatie 10](#_Toc490594486)

[3.3.2. Hernieuwde betrokkenheid 12](#_Toc490594487)

[3.4. Winst 13](#_Toc490594488)

[3.4.1. In-App advertenties 13](#_Toc490594489)

[4. Proof-of-Concept 14](#_Toc490594490)

[4.1. Authentication 14](#_Toc490594491)

[4.1.1. Wat houdt dit in en hoe werkt het 14](#_Toc490594492)

[4.1.2. Hoe werd het getest 15](#_Toc490594493)

[4.1.3. Besluit 17](#_Toc490594494)

[4.2. Realtime Database 17](#_Toc490594495)

[4.2.1. Wat houdt dit in en hoe werkt het 17](#_Toc490594496)

[4.2.2. Hoe werd het getest 17](#_Toc490594497)

[4.2.3. Besluit 18](#_Toc490594498)

[4.3. Notifications 18](#_Toc490594499)

[4.3.1. Wat houdt dit in en hoe werkt het 18](#_Toc490594500)

[4.3.2. Hoe werd het getest 19](#_Toc490594501)

[4.3.3. Besluit 21](#_Toc490594502)

[4.4. Crash reporting 22](#_Toc490594503)

[4.4.1. Wat houdt het in en hoe werkt het 22](#_Toc490594504)

[4.4.2. Hoe werd het getest 22](#_Toc490594505)

[4.4.3. Besluit 24](#_Toc490594506)

[4.5. Performance 24](#_Toc490594507)

[4.5.1. Wat houdt het in en hoe werkt het 24](#_Toc490594508)

[4.5.2. Hoe werd het getest 24](#_Toc490594509)

[4.5.3. Besluit 26](#_Toc490594510)

[4.6. Analytics 27](#_Toc490594511)

[4.6.1. Wat houdt het in en hoe werkt het 27](#_Toc490594512)

[4.6.2. Hoe werd het getest 27](#_Toc490594513)

[4.6.3. Besluit 28](#_Toc490594514)

[4.7. Remote Config 29](#_Toc490594515)

[4.7.1. Wat houdt het in en hoe werkt het 29](#_Toc490594516)

[4.7.2. Hoe werd het getest 29](#_Toc490594517)

[4.7.3. Besluit 32](#_Toc490594518)

[4.8. Conclusie Proof-of-Concept 32](#_Toc490594519)

[5. Het schrijven van applicatie met en zonder Firebase 34](#_Toc490594520)

[5.1. Voordelen 34](#_Toc490594521)

[5.1.1. Met Firebase: 34](#_Toc490594522)

[5.1.2. Zonder Firebase: 35](#_Toc490594523)

[5.2. Nadelen 35](#_Toc490594524)

[5.2.1. Met Firebase: 35](#_Toc490594525)

[5.2.2. Zonder Firebase: 36](#_Toc490594526)

[6. Bevraging bij bedrijven 37](#_Toc490594527)

[6.1. Vraagstelling 37](#_Toc490594528)

[6.2. Respons bevraagde bedrijven 38](#_Toc490594529)

[6.3. Analyse van de antwoorden 39](#_Toc490594530)

[Vraag 1: Heeft u ooit gehoord van Firebase? Zo ja, waar? 39](#_Toc490594531)

[Vraag 2: Indien neen op de eerste vraag: Met de gegeven uitleg zouden jullie overwegen Firebase uit te testen en bij positieve testen het toepassen ervan aan potentiële klanten voor te stellen? 39](#_Toc490594532)

[Vraag 3: Heeft u Firebase ooit gebruikt in 1 of meerdere projecten? 39](#_Toc490594533)

[Vraag 4: Van welke features, aangeboden door Firebase, heeft u al gebruik gemaakt? 39](#_Toc490594534)

[Vraag 5: Welke features, aangeboden door Firebase, zijn voor u en uw bedrijf het meest interessant? 40](#_Toc490594535)

[Vraag 6: Heeft u interesse om Firebase in toekomstige projecten te gebruiken? Zo ja, waarom? 40](#_Toc490594536)

[Vraag 7: Waarom is Firebase, volgens u, wel of niet beter dan soortgelijke technologieën (Azure, AWS)? 41](#_Toc490594537)

[Vraag 8: Zou u het gebruik van Firebase aanraden bij andere bedrijven en klanten? Waarom (niet)? 41](#_Toc490594538)

[6.4. Besluit van de bevraging bij bedrijven 42](#_Toc490594539)

[7. Algemeen besluit 43](#_Toc490594540)

[8. Referentielijst 45](#_Toc490594541)

[9. Bijlages 46](#_Toc490594542)

# 

# Inleiding

Entertainment, gezondheid, business, sociale media. Apps, vandaag kan niemand meer zonder. Ze vormen een essentieel onderdeel van ons persoonlijk en professioneel leven. Van het spelen van spelletjes tot een organisatietool voor agendaplanning, er bestaat wel een app voor. De jongste tien jaar is het ontwikkelen van software duidelijk verschoven van complexe systemen naar eenvoudigere en modulaire applicaties. Tegenwoordig willen gebruikers immers een applicatie op maat die overal en te allen tijde toegankelijk is.  
Met andere woorden, apps zijn big business geworden. Sinds de allereerste smartphone en de bijhorende apps is de markt voor mobiele applicaties wereldwijd een miljardenindustrie geworden. Met een jaarlijkse groei van 20% sinds 2010, volgt de mobiele applicatiemarkt in België deze globale trend. Het aantal professionele applicatieontwikkelaars in België stijgt nog elke dag. De meesten werken voor startups, grote bedrijven of voor eigen rekening.

Omdat het ontwikkelen van applicaties tot voor kort een tijdrovend, moeizaam en duur proces was, kwam Google met een oplossing, Firebase genaamd. Firebase werd door Google gecreëerd om snel kwaliteitsvolle applicaties te ontwikkelen zonder de verplichting een eigen backend op te bouwen.

Deze bachelorproef zal uitleggen wat Firebase concreet is, welke componenten er deel van uitmaken en hoe ze werken. Het is niet enkel de bedoeling om een introductie te geven van Firebase maar ook om deze technologie en haar componenten uit te testen door middel van een proof-of-concept.  
De ontwikkelde demo-applicatie verschaft duidelijkheid over de voornaamste componenten, hoe de werking van elk van hen getest wordt en wat er gemakkelijk en of moeilijk aan is qua implementatie en wat eventueel een alternatief voor Firebase zou kunnen zijn. Verder worden de voor- en nadelen van Firebase toegelicht. Een beknopte vergelijking met het schrijven van een applicatie zonder Firebase moet inzicht bieden in de beperkingen en obstakels die zich hierbij stellen.

In een volgende fase worden bedrijven aangeschreven en bevraagd om na te gaan hoe bekend Firebase is sinds het eind 2016 gelanceerd werd door Google en of het al effectief toegepast wordt in het bedrijfsleven.  
Is Firebase inderdaad de oplossing die het voorhoudt te zijn om snel kwaliteitsvolle applicaties te ontwikkelen en zal het nog uitbreiding kennen? Op het einde zal duidelijk worden of het een goede zaak is voor bedrijven om Firebase te gebruiken voor het ontwikkelen van hun applicaties of niet.

# Wat is Firebase?

## Beschrijving Firebase

Firebase is een ontwikkelplatform voor mobiele- en webapplicaties bestaande uit meerdere ondersteunende componenten die door ontwikkelaars kunnen gebruikt worden.  
Het bedrijf, Firebase, werd opgericht in 2011 door Andrew Lee en James Tamplin. Het initiële product van Firebase was een *Realtime Database*. Dit liet ontwikkelaars toe om data op te slaan en te synchroniseren over verschillende soorten toestellen (tablets, smartphones, computers, …).  
In 2014 werd Firebase door Google overgenomen en in oktober 2016 werden er vele nieuwe componenten geïntroduceerd tijdens de Google I/O, een jaarlijkse ontwikkelaarsconventie gehouden te San Francisco.

## Gebruik Firebase

Ontwikkelaars kunnen Firebase zelf toevoegen aan hun applicatie (Android, iOS, C++, Unity, web) door een stappenplan te volgen op de site van Google Firebase. Eens toegevoegd kan de ontwikkelaar kiezen uit een reeks aangeboden componenten die aan zijn noden voldoen.

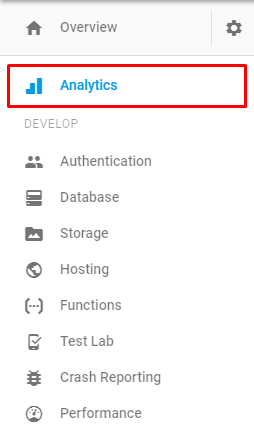
# Componenten van Firebase

## Analyse

### Firebase Analytics

Firebase Analytics is de kern van Firebase. Het is een gratis analyse middel dat inzicht biedt op het gebruik van de applicatie en op de gebruikersbetrokkenheid. Firebase Analytics is geïntegreerd over de verscheidene Firebase componenten en laat tot 500 verschillende events toe.  
De klant kan met behulp van Firebase Analytics een analytisch rapport laten opstellen om het gedrag van de gebruikers te begrijpen door middel van het opschrijven van data (events) in het logboek en door het zelf aanmaken van doelgroepen (audiences). Dit analytisch rapport staat op zijn beurt toe om geïnformeerde beslissingen omtrent de applicatie te maken.

Het analytisch rapport is te vinden op het linkerpaneel in de Firebase Console (screenshot 3.1).

  
(screenshot 3.1)

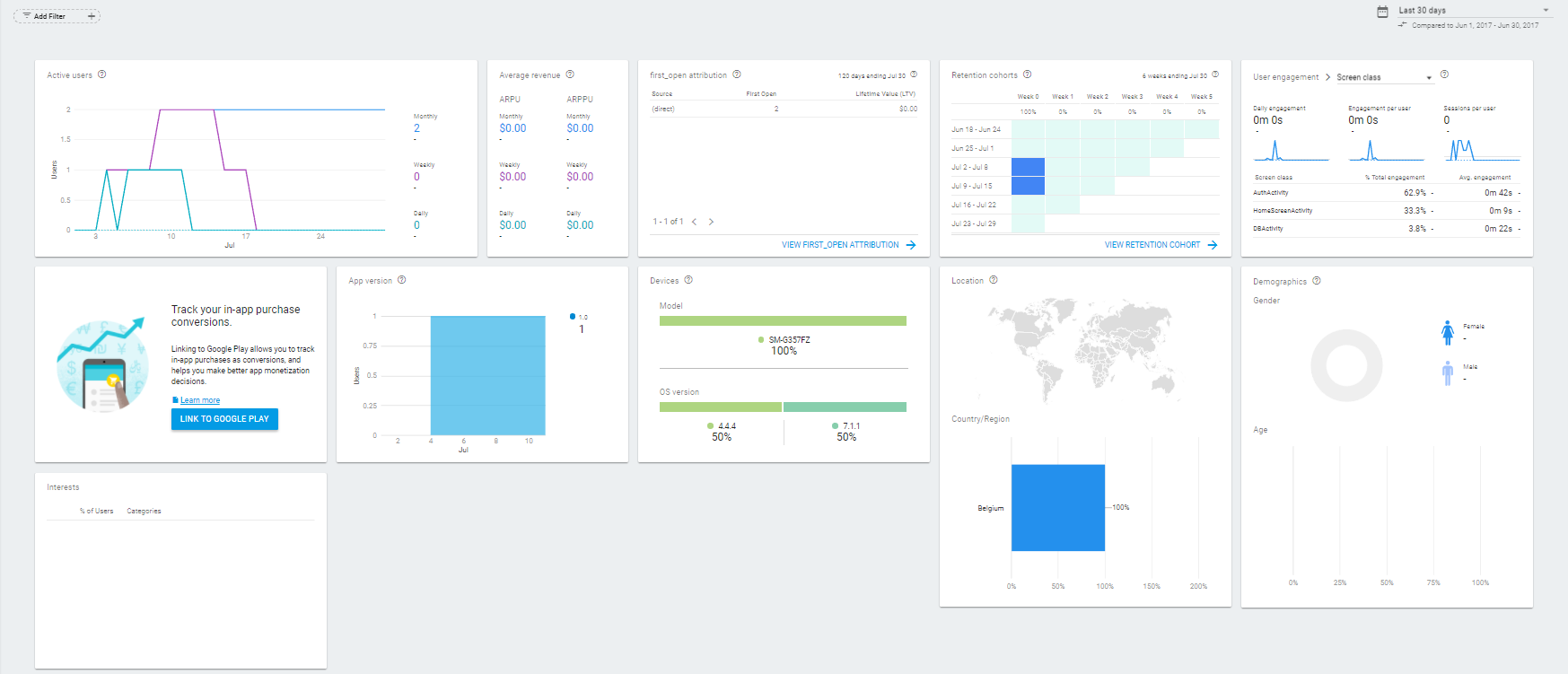
Firebase Analytics is geïntegreerd met andere Firebase componenten:

* BigQuery
* Firebase Crash Reporting
* Firebase Cloud Messaging
* Firebase Remote Config
* Google Tag Manager

### Onderdelen van Firebase Analytics:

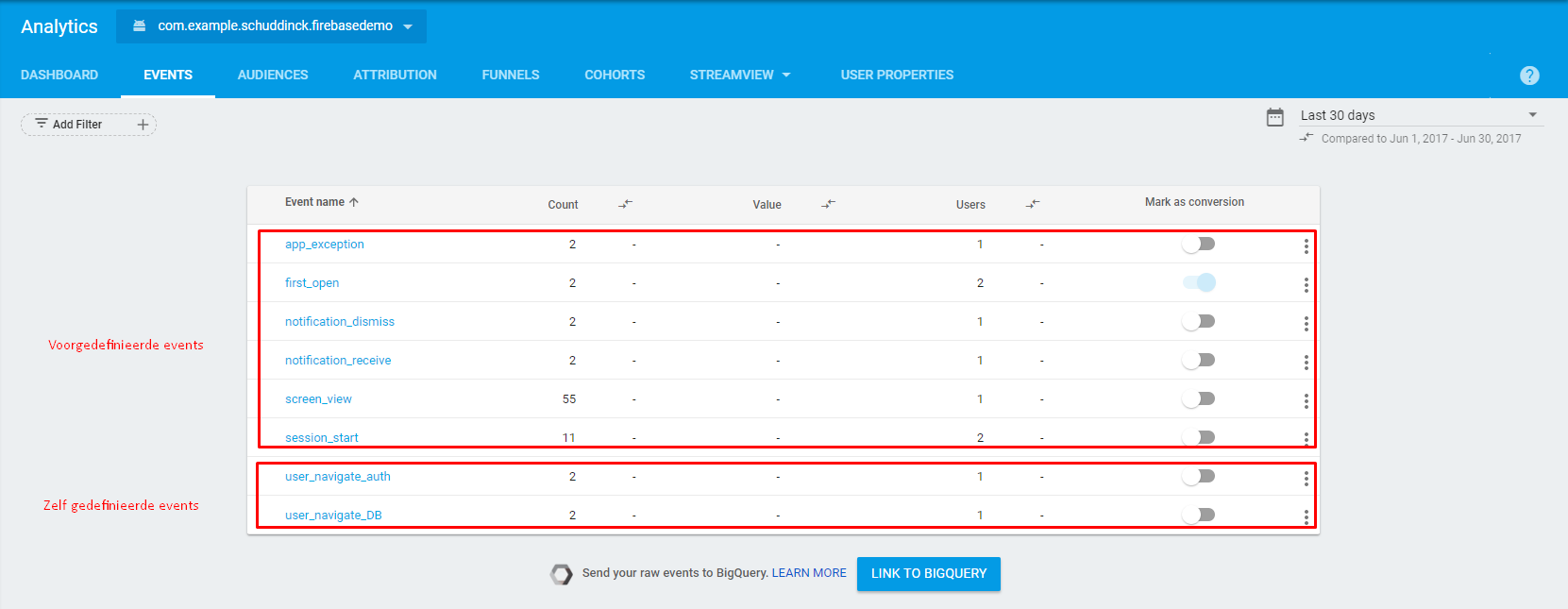
* Dashboard:

Op het dashboard (screenshot 3.2) is een overzicht te zien van de verschillende onderdelen (cohorts, attributen) en extra informatie. De extra informatie geeft de klant meer inzicht over zijn gebruikers (geslacht, locatie, type smartphone en type ondersteunings systeem, hoe actief de app gebruikt wordt, welke app versies gebruikt worden, welke schermen het vaakst gebruikt worden, hoeveel geld er verdiend is door de app)

  
(screenshot 3.2)

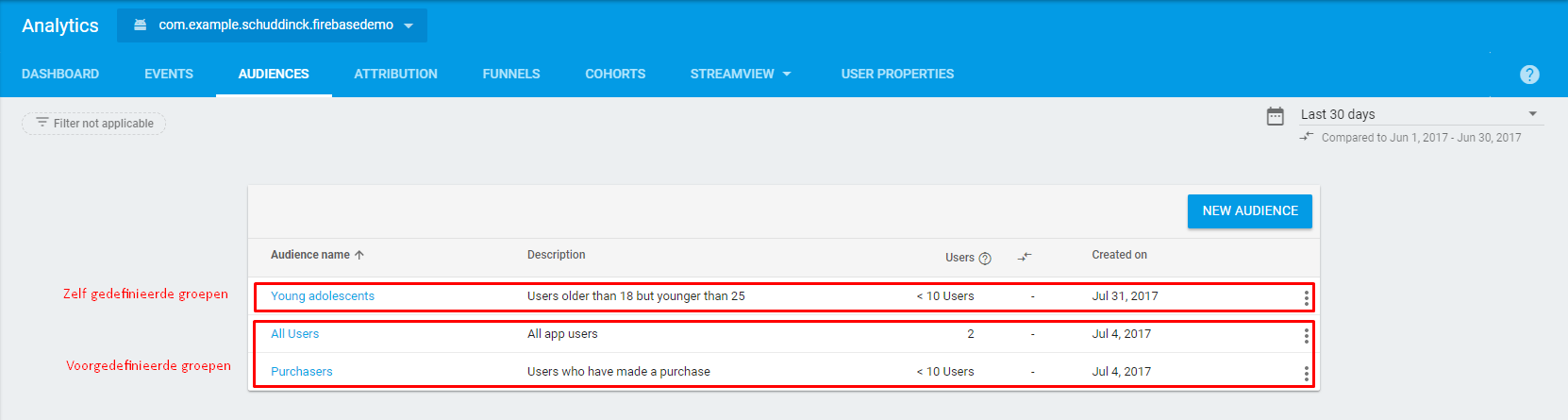
* Events:

Events zijn gebruikershandelingen of gebeurtenissen in de applicatie zelf. Er zijn een aantal voorgedefinieerde events zoals het eerst openen van de applicatie (op de afbeelding “first\_open” event) of het ontvangen van een notificatie (op de afbeelding “notification\_receive”). Daarnaast kan de ontwikkelaar zelf nog andere events toevoegen (screenshot 3.3) door gebruik te maken van de Firebase Analytics SDK (Software Development Kit).

  
(screenshot 3.3)

* Audiences:

Audiences (screenshot 3.4) zijn gegroepeerde gebruikers. Standaard zijn er 2 voorgedefinieerde groepen, “alle gebruikers” en “kopers” (dit zijn gebruikers die al aankopen gemaakt hebben in de applicatie). Ook kunnen er groepen gebruikers door de klant gedefinieerd worden. Bijvoorbeeld een groep gebruikers die ouder zijn als 18 jaar of gebruikers die een bepaalde versie van de app hebben.

  
(screenshot 3.4)

* Attribution, Funnels, Cohorts, StreamView/DebugView, User Properties
  + Attribution:  
    Toont hoeveel events gestart zijn door iedere bron en advertentie netwerk.
  + Funnels:  
    Visualiseert hoe gebruikers navigeren doorheen de applicatie.
  + Cohorts:  
    Is een set van gebruikers die de applicatie gelijktijdig (zoals dezelfde dag of dezelfde week) beginnen gebruiken zijn.
  + StreamView/DebugView:

StreamView geeft een live overzicht van de analytics events die door de Analytics servers worden verzameld.  
DebugView laat toe om events in realtime te volgen.

* + User Properties  
    Zijn eigenschappen van gebruikers die enerzijds voorgedefinieerd zijn, zoals leeftijd en geografische locatie, en anderzijds door de klant kunnen bepaald worden.

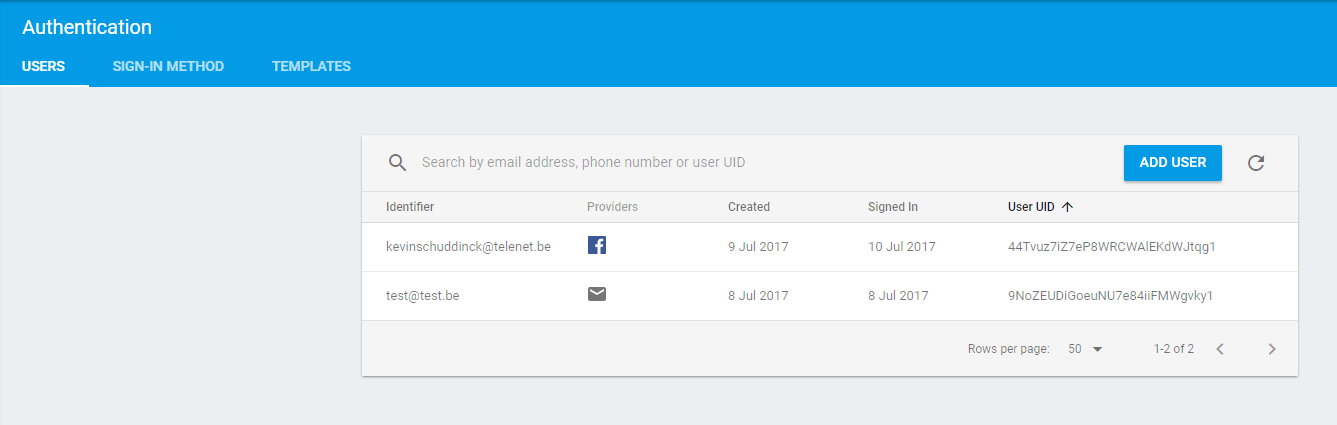
Firebase Analytics kan gebruikt worden voor iOS, Android, C++ en Unity applicaties.

## Ontwikkeling

### Authentication (iOS, Android, Web, C++, Unity)

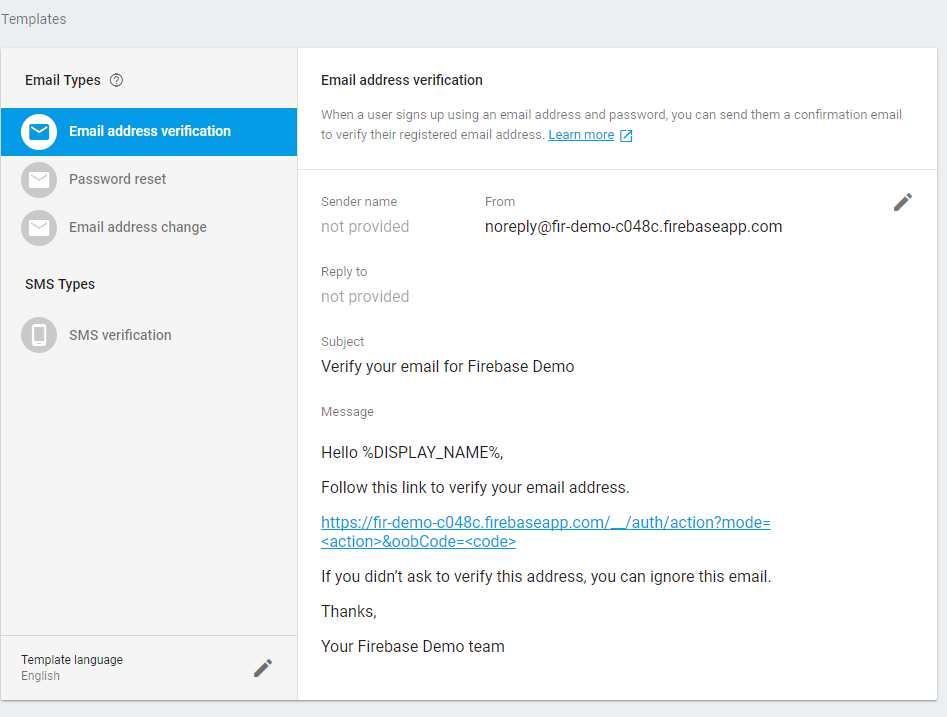
Firebase Authentication geeft backend services (backend services ondersteunen de frontend en frontend is wat de gebruiker ziet en waarmee hij kan interageren), gemakkelijk toepasbare SDK’s en voorgeprogrammeerde UI libraries (kant en klaar geschreven code om te gebruiken in de layout) om gebruikers te verifiëren.   
Ten tweede voorziet het ook mogelijkheden om in te loggen met behulp van Facebook, Twitter, Github en Google.   
Voor meer flexibiliteit is er altijd een derde mogelijkheid, nl. een eigen authenticatiesysteem uitwerken, die dan kan gekoppeld worden aan de Firebase SDK. Een laatste optie is om de gebruiker anoniem te laten inloggen. Hierbij wordt er tijdelijk een anoniem account aangemaakt. De ontwikkelaar beslist dan uiteraard zelf wat een anoniem account kan en niet kan.

Op de screenshot hieronder (screenshot 3.5) kan men de gebruikers zien, wanneer deze aangemaakt zijn, het laatst aangemeld waren en user UID.



(screenshot 3.5)

Verder zijn er ook nog templates voorzien (screenshot 3.6) door Firebase Authentication. Deze kunnen gebruikt worden om een e-mail of sms te versturen naar de gebruiker om zijn e-mailadres te verifiëren, wachtwoord te resetten of e-mailadres te veranderen.



(screenshot 3.6)

Firebase Authentication kan gebruikt worden voor iOS, Android, C++, web en Unity applicaties.

### Cloud functions (iOS, Android, Web, C++, Unity)

Niet alle logica mag zomaar in de applicatie geschreven worden. Als dit wel gedaan wordt zorgt dit voor gevaren zoals reverse engineering (reverse engineering is het uiteen halen van een object om te zien hoe het werkt en het zo te reproduceren of te veranderen voor andere doeleinden). Daarom heeft Firebase ook Cloud Functies geïntroduceerd.  
Belangrijke code of intensieve operaties kunnen geüpload worden naar de Firebase servers zodat deze beveiligd zijn tegen hackers.

Firebase Cloud Functions kan gebruikt worden voor iOS, Android, C++, web en Unity applicaties.

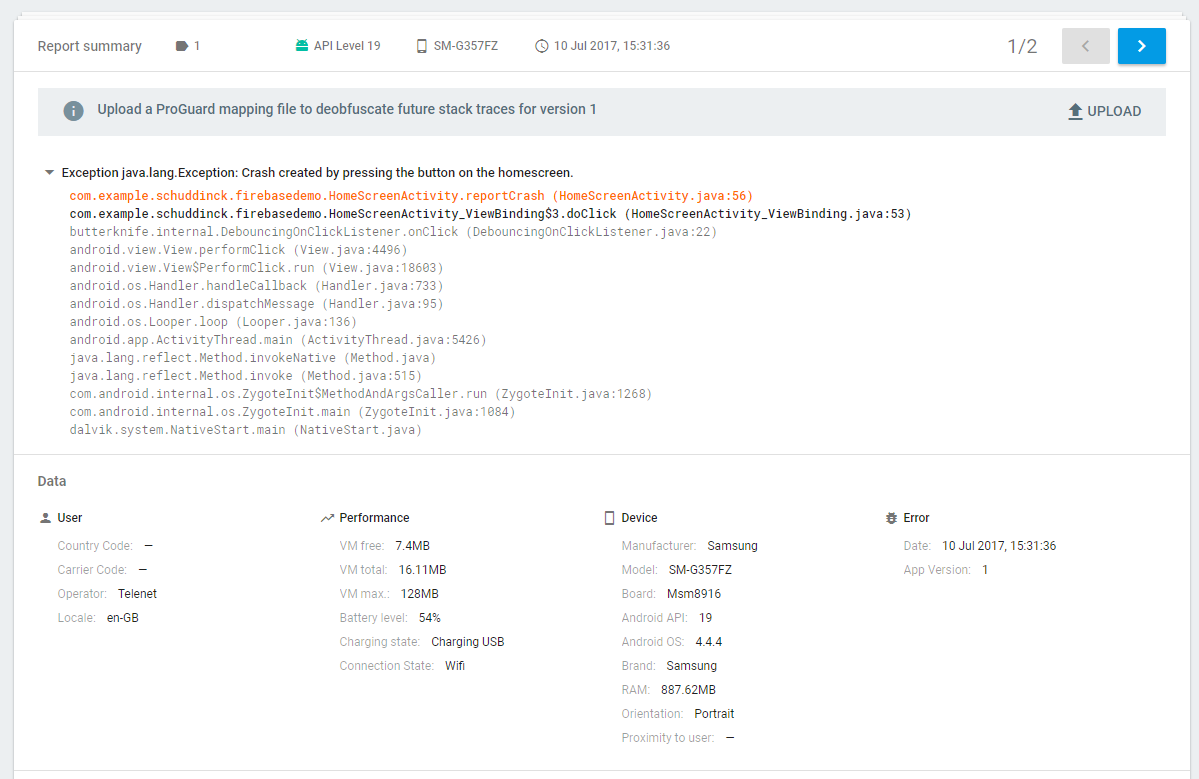
### Cloud Messaging

Firebase Cloud Messaging laat toe om berichten (notificaties), van maximaal 4KB, over meerdere platformen (iOS, Android, Web, C++, Unity) te versturen. Berichten kunnen behandeld worden voor enkele toestellen, groepen toestellen (bv. enkel iOS of enkel Android) of topics. Topics zijn onderwerpen waarop toestellen zich kunnen aboneren.

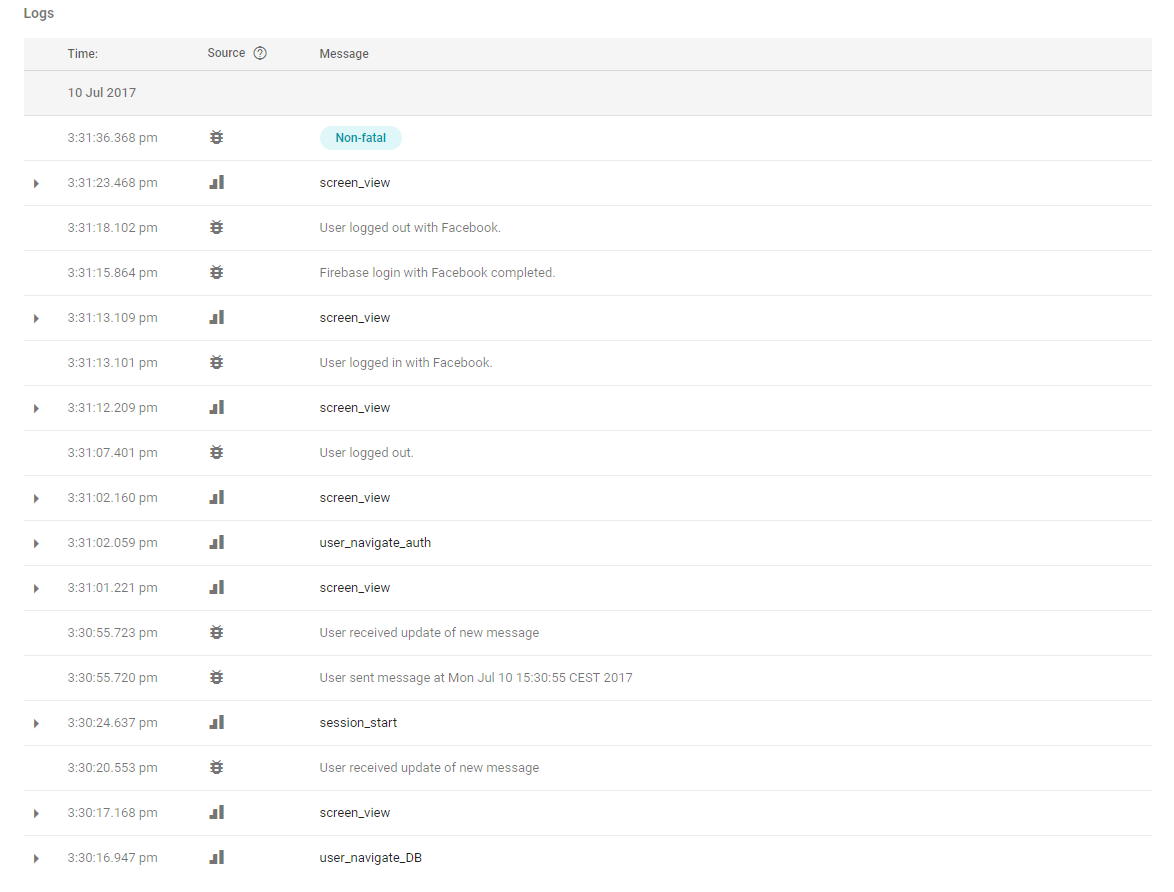
Firebase Cloud Messaging kan gebruikt worden voor iOS, Android, C++, web en Unity applicaties.

### Crash Reporting (iOS, Android)

Firebase Crash Reporting stelt gedetailleerde rapporten op over fouten in de applicatie. Fouten worden gegroepeerd per gelijkenis en geordend volgens impact op het gebruik van de applicatie. Bij elk rapport komt er informatie zoals stacktrace (een lijst van methodes die de applicatie aan het uitvoeren was voor er zich een exceptie voordeed), type toestel, app versie, … maar ook een log (screenshot 3.7 en 3.8) die elke gebeurtenis registreert en bijhoudt.



(screenshot 3.7)



(screenshot 3.8)

Firebase Crash Reporting kan gebruikt worden voor iOS en Android applicaties.

### Hosting

Firebase Hosting biedt snelle en veilige webhosting. Snel door het gebruik van Content Delivery Network (CDN) en veilig door een 2048-bit SSL certificaat (Digicert, n.d.).  
Alle static assets (zaken die niet veranderen) zoals HTML, CSS, JS, … kunnen dus hierop gehost worden.  
Custom domains worden ook gratis aangeboden door Firebase, dit wordt voorzien door een subdomain dat wordt aangemaakt voor het project.

Firebase Hosting kan natuurlijk enkel gebruikt worden voor het webapplicaties.

### Realtime Database

De realtime database van Firebase is een cloud hosted NoSQL (Not Only Structured Query Language) database.   
Voordelen van een NoSQL database zijn o.a.:

* Vermogen tot schaalbaarheid
* Kan omgaan met veranderingen
* Kan omgaan met Big Data
* Minder nood aan onderhoud

Wanneer er data wordt weggeschreven naar de database wordt dit uiteraard opgeslagen maar tevens wordt het ook gesynchroniseerd naar meerdere clients in geen tijd (screenshot 3.9). Dit is onmisbaar voor bijv. collaboratieve apps.



(screenshot 3.1)

De realtime database heeft nog een andere kernfunctionaliteit, nl. het opslaan van data in het toestel zelf. Het nut hiervan is dat de app responsief blijft, zelfs als de internetconnectie onderbroken wordt. Als er terug connectie met internet gemaakt is wordt alles terug gesynchroniseerd.

Firebase Realtime Database kan gebruikt worden voor iOS, Android, C++, web en Unity applicaties.

### Storage

Cloud Storage geeft aan applicatie ontwikkelaars de mogelijkheid om inhoud gegenereerd door de gebruikers zoals foto's of videos op te slaan en te beheren. Het uploaden van foto's of videos wordt opgeslaan in de cloud storage.   
De could storage is schaalbaar naargelang er meer ruimte nodig is of niet. Elke connectie met de cloud storage is ook beveiligd.  
Wanneer er zich tijdens het uploaden een storing voordoet bij de connectie met het internet wordt deze tijdelijk gepauzeerd tot wanneer er terug een stabiele connectie is.   
Dit component zal vooral zijn nut bewijzen voor sociale media applicaties.

Firebase Cloud Storage kan gebruikt worden voor iOS, Android, C++, web en Unity applicaties.

### Test Lab

Android is één van de grootste ondersteuningssystemen voor smartphones en wordt dus door vele merken gebruikt. Niet elk merk brengt een toestel met dezelfde schermdimensies uit, daarom kan het ontwikkelen voor Android soms een hardnekkige karwei zijn.  
Hiervoor biedt Firebase een oplossing, nl. Test Lab.  
Test Lab staat toe om het uitzicht op verschillende toestellen te evalueren door middel van screenshots en tips. Daarbij biedt Test Lab ook nog voorgedefinieerde (Robo) en eigen tests.  
Test resultaten geven een gedetailleerd verslag voor elk gebruikt toestel (inclusief screenshots, logs en een lijst van crashes).

Firebase Test Lab kan enkel gebruikt worden voor Android applicaties.

## Groeimogelijkheden

### Acquisatie

#### Dynamische links

Apps worden ontdekt door mond-tot-mond reclame, via tekstberichten of tweets. Vaak wordt dan een link voorzien die verwijst naar een statische pagina die niet weet wat voor toestel er gebruikt wordt.  
Met Firebase kan er gebruik gemaakt worden van dynamische links die de gebruikers verwijzen naar de gelinkte content op de native app. Deze dynamische link werkt cross-platform en de toestand wordt bijgehouden tijdens het installatieproces (voorbeeld 1.1).  
Ook wordt er een analytisch inzicht gegeven voor het gebruik van deze links bij Firebase Analytics.

Voorbeeld 1.1:  
Persoon A daagt persoon B uit voor een spel, maar persoon B heeft de app om het spel te spelen nog niet geïnstalleerd. De link verwijst hem naar de Google Play Store of de iTunes Store (afhankelijk van welk toestel er gebruikt wordt) en slaat de toestand van de uitdaging op. Wanneer de app geïnstalleerd is zal de uitdaging verdergaan.

Firebase Dynamic Links kan gebruikt worden voor iOS, Android, C++, web en Unity applicaties.

#### 3.3.1.2. AdWords

AdWords is een project van Google en wordt gebruikt om producten en diensten te adverteren. Het steunt op twee belangrijke netwerken: het zoeknetwerk (google search) en het Display Netwerk. Het zoeknetwerk zal mensen bereiken die op zoek zijn naar specifieke producten of diensten terwijl het Display Netwerk eerder mensen zal bereiken die algemene diensten van google gebruiken (Google Play Store, YouTube, Gmail).

Gebruikers van AdWords bepalen een budget dat mag gebruikt worden en Google AdWords zal bepalen welk kanaal het beste is om advertenties te tonen.  
AdWords is geïntegreerd in Firebase. Zo kan er gebruik gemaakt worden van de statistieken om te bepalen hoe het project het best geadverteerd wordt.

#### Remote config

Firebase Remote Config is een cloud service dat het gedrag en uitzicht van een applicatie veranderd zonder een update van de app uit te brengen. Een update moet doorheen een heel proces van stappen lopen vooraleer het beschikbaar is voor de gebruikers.  
Remote Config geeft de mogelijkheid om in-app waarden te definiëren die het toestaan om het gedrag en uitzicht van de app te beheren. Later kan de Firebase Console gebruikt worden om standaard waarden aan te passen voor alle gebruikers of naargelang groepen van gebruikers. Bijvoorbeeld er is een applicatie ontwikkeld maar er is een taalfout ingeslopen. In dit geval kan de Remote Config gebruikt worden om de fout te corrigeren, mits de tekst als waarde hierin opgeslagen is, zonder een update uit te brengen.   
Een ander voorbeeld is een applicatie van een spel waarin men de moeilijkheidsgraad van een bepaald niveau kan aanpassen naargelang de groep van gebruikers om iedereen tevreden te houden. Zo kan er een onderscheid gemaakt worden tussen "hardcore" en "casual" spelers.

Firebase Remote Config kan gebruikt worden voor iOS, Android, C++ en Unity applicaties.

### Hernieuwde betrokkenheid

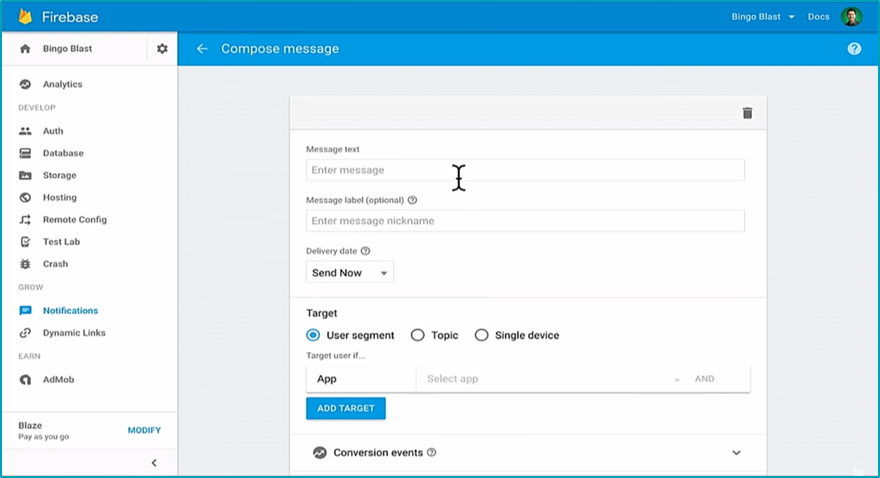
#### 3.3.2.1. App Indexing

App indexing is een goede manier om apps naar de mensen te brengen d.m.v. Google Search. Wanneer er een google search wordt uitgevoerd, kunnen apps worden weergegeven als zoekresultaten. Hoe meer er op de link naar de app geklikt wordt vanuit de Google Search hoe hoger de app in de Google Search ranking komt te staan.

App Indexing kan gebruikt worden voor iOS en Android applicaties.

#### 3.3.2.2. Notificaties

Firebase notifications met als kern Firebase Cloud Messaging (FCM) laat toe om doelgerichte notificaties naar gebruikers te sturen a.d.h.v. een grafisch interface (screenshot 3.10). Wanneer de notificatie verstuurd is, kan er terug analytische data verzameld worden.



(screenshot 3.10)

## Winst

### In-App advertenties

#### 3.4.1.1. AdMob

AdMob is een gemakkelijke manier om geld te verdienen met doelgerichte, in-app reclame. Samen met Firebase Analytics verschaft dit meer inzicht in het gebruik van de app bijv. om trends te spotten en daarop in te spelen.

AdMob kan gebruikt worden voor iOS, Android, C++ en Unity applicaties.

# Proof-of-Concept

Uit het vorig hoofdstuk blijkt duidelijk wat Firebase is en kan. Aangezien er nog geen volwaardig onderzoek naar verricht is, is er voor deze bachelorproef een proof-of-concept uitgewerkt met als doel bepaalde toepassingsmogelijkheden (componenten) met en zonder Firebase uit te testen. Iedere app is tenslotte opgebouwd uit een serie componenten.

De demo applicatie gemaakt voor deze bachelorproef belicht de volgende voornaamste componenten:

* Authentication
* Realtime Database
* Notifications
* Crash Reporting
* Performance monitoring
* Analytics
* Remote Config

Deze componenten kunnen aan de applicatie toegevoegd worden door ze te compileren (bv. compile ‘com.google.firebase:firebase-auth:11.0.2’).  
Implementatie is vereist voor de volgende componenten:

* Authentication
* Realtime Database
* Crash Reporting (extra code hierbij is geen noodzaak, maar wel een best practice)
* Analytics (extra code hierbij is geen noodzaak, maar wel een best practice)
* Remote Config

## Authentication

### Wat houdt dit in en hoe werkt het

Authentication is het proces om te bepalen of iemand wel degelijk is wie hij zegt te zijn. Dit komt er op neer dat er verschillende inlogmogelijkheden zijn, nl.

1. Het aanmaken van een account bij Firebase
2. Het gebruiken van een anoniem account bij Firebase
3. Inloggen via een account van Facebook, Twitter, Github of Google
4. Het gebruiken van een eigen inlogsysteem dat gekoppeld kan worden aan Firebase

Met behulp van het creëeren van accounts kunnen applicaties de identiteit van een gebruiker verifiëren. Een account wordt geregistreerd door middel van identiteitsgegevens (dit kan bv. onder de vorm van de naam van de gebruiker, een alias of een e-mailadres) en een wachtwoord.

### Hoe werd het getest

Er werden drie login methodes geïmplementeerd.  
De eerste methode is het aanmaken van een standaard account bij Firebase met een e-mailadres en wachtwoord.  
De tweede methode is het inloggen bij Firebase via een account van Facebook. In dit geval is het wel vereist de applicatie eerst te registreren bij Facebook zelf.  
De laatste methode is het creëeren van een eigen inlogsysteem zonder te koppelen aan Firebase.  
Hier komen er natuurlijk heel wat meer zaken bij kijken zoals het encrypteren van wachtwoorden, de accounts met geëncrypteerde wachtwoorden opslaan in een database, het aanmaken van een eigen database en deze onderhouden en het terug decrypteren van wachtwoorden om deze te kunnen vergelijken met het door de gebruiker ingegeven wachtwoord. Als een inlogsysteem gekoppeld moet worden aan Firebase komt er nog een extra stap bij, nl. het genereren van een Firebase token eens de gebruiker geverifiëerd is.

De eerste rij afbeeldingen toont de login sequentie waar er met een e-mailadres en wachtwoord wordt aangemeld en afgemeld  
De tweede rij afbeeldingen toont de registratie sequentie waar er een account aangemaakt wordt met het e-mailadres test2@test.be. Na het registreren wordt het account ook direct aangemeld  
De derde rij afbeeldignen toont de facebook login sequentie waar er aangemeld wordt met “kevinschuddinck@telenet.be”.

|  |  |  |
| --- | --- | --- |
| C:\Users\SCHUDDINCK\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Screenshot_1.png  Login sequentie | C:\Users\SCHUDDINCK\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Screenshot_2.pngaangemeld met test@test.be na ingeven van e-mail en wachtwoord | C:\Users\SCHUDDINCK\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Screenshot_3.pngafgemeld met test@test.be |
| C:\Users\SCHUDDINCK\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Screenshot_4.pngregisteren van test2@test.be | C:\Users\SCHUDDINCK\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Screenshot_5.pngna registreren wordt het account direct aangemeld | C:\Users\SCHUDDINCK\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Screenshot_3.pngafgemeld met test2@test.be |
| C:\Users\SCHUDDINCK\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Screenshot_6.pngInloggen met facebook | C:\Users\SCHUDDINCK\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Screenshot_7.pngHet accepteren van de voorwaardes (is hierop al gebeurd) | C:\Users\SCHUDDINCK\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Screenshot_8.pngAangemeld met facebook, de log out button van facebook verschijnt |

### Besluit

Alle methodes zijn zowel met als zonder Firebase op zich gemakkelijk aan te maken.  
Een groot voordeel echter van Firebase is de ingebouwde beveiliging ervan.   
Immers als er gebruik gemaakt wordt van een eigen inlogsysteem om een account aan te maken moet er voor gezorgd worden dat wachtwoorden geëncrypteerd worden alvorens ze opgeslagen worden. Dit om ze te beveiligen tegen hackers. Bovendien moet er een opslagruimte voorzien worden voor de accounts. Zowel de opslagruimte als de encryptie wordt bij Firebase automatisch voorzien.

## Realtime Database

### 4.2.1. Wat houdt dit in en hoe werkt het

Een realtime database is een database dat gebruik maakt van realtime bewerkingen (dit zijn bewerkingen uitgevoerd in een gespecifieerde tijd) om te kunnen omgaan met taken wiens toestand voortdurend verandert. Realtime bewerkingen worden zo snel uitgevoerd zodat het resultaat dat terugkomt direct kan gebruikt worden in volgende stappen.  
Het verschilt van de traditionele databases die persistente data bevatten en niet aangetast worden door tijd.   
De realtime database van Firebase voorziet nog een extra component, nl. het lokaal opslaan van data zodat de applicatie responsief blijft. Als dit niet gedaan wordt, bevriest de applicatie tijdens het ophalen van nieuwe data wanneer deze geen connectie heeft met het internet.

### Hoe werd het getest

Het testen van de realtime database van Firebase ging als volgt:

1. De gebruiker verstuurt data (in de vorm van een bericht) naar de bestaande databank van Firebase. Vooraleer de data naar de databank kan verstuurd worden, wordt dit omgezet naar JSON formaat. Een JSON formaat van data is alle data omzetten naar tekst zodat dit gemakkelijk kan opgeslaan worden in de databank.
2. Deze realtime databank slaat de verandering op
3. De realtime databank verstuurt alle of de laatste (naargelang de instellingen) data in JSON formaat inclusief veranderingen door naar de clients.
4. De clients zetten de gekregen data (JSON formaat) om naar de oorspronkelijke data en handelen de oorspronkelijke data verder intern af.

|  |  |  |
| --- | --- | --- |
| C:\Users\SCHUDDINCK\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Screenshot_10.pngDe reeds verzonden bericht worden opgehaald wanneer er naar dit scherm genavigeerd wordt | C:\Users\SCHUDDINCK\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Screenshot_11.pngHet verzenden van een nieuw bericht “derde test” wordt hier klaargemaakt | C:\Users\SCHUDDINCK\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Screenshot_12.pngNa op de knop “Send” gedrukt te hebben wordt het bericht verzonden en krijgt de applicatie een update dat de realtime database een nieuw bericht heeft |

### Besluit

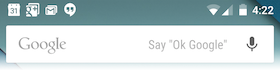
Het gebruik van de realtime database van Firebase is een vrij gemakkelijk proces. Verdere mogelijkheden met de databank is dat de nieuwste data gesorteerd (op sleutel [een sleutel is een uniek gegeven waarmee de databank data opslaat], waardes of een specifieke waarde) vestuurd kan worden naar de clients.

Om een realtime database te gebruiken zonder Firebase kan *Realm mobile platform* gebruikt worden (Realm, n.d.) maar bij de realtime database van Realm moet de ontwikkelaar zelf nog een server opzetten om daarop de databank te plaatsen.

## Notifications

### Wat houdt dit in en hoe werkt het

Een notificatie is een bericht van een applicatie dat getoond kan worden aan de gebruiker buiten diezelfde applicatie (Android, n.d.). Wanneer het systeem een notificatie ontvangt wordt dit eerst getoond als een icoon in het notificatie gebied (standaard een balk bovenaan het scherm zie afbeelding 4.1). Om de notificatie te bekijken, opent de gebruiker de "notification drawer". De gebruiker kan deze acties om het even wanneer verrichten, dus ook tijdens het gebruiken van een andere applicatie.



(afbeelding 4.1 : https://developer.android.com/images/ui/notifications/notification\_area.png)

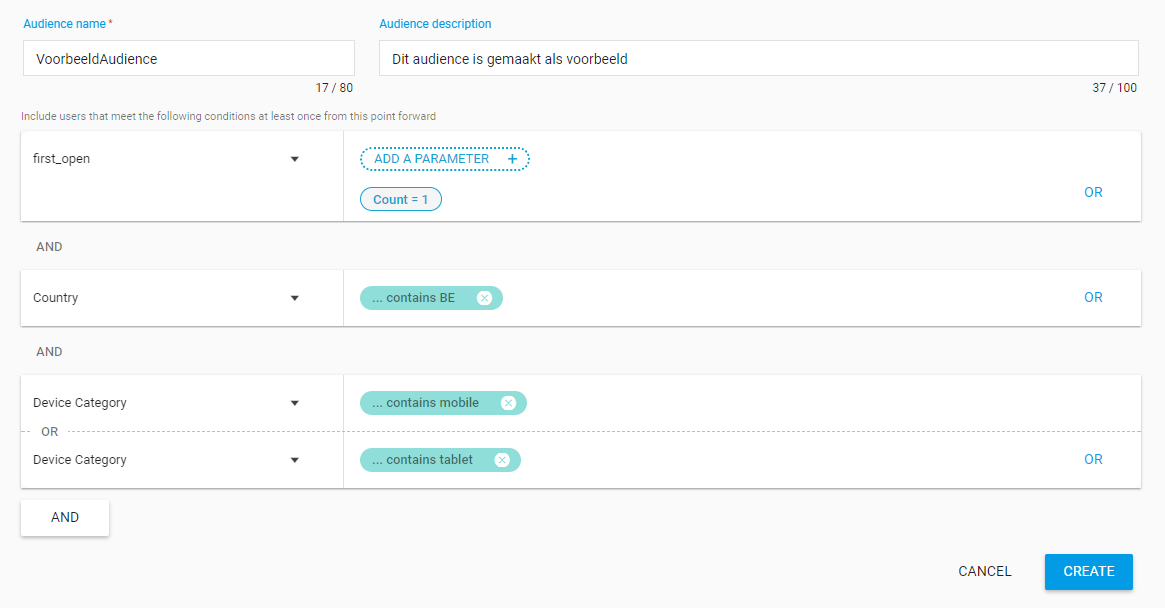
De iconen boven de google balk zijn notificaties.

### Hoe werd het getest

Door het compileren van Firebase Cloud Messaging in de applicatie, kan de gebruiker van het toestel waarop de applicatie is geïnstalleerd notificaties ontvangen.

Notificaties kunnen verstuurd worden naar doelgroepen, naar enkele toestellen of enkel naar toestellen die geabonneerd zijn op specifieke topics.

Doelgroepen zijn groepen bepaald door de klant zelf, deze groepen kunnen gemaakt worden in de “audiences” tab van Firebase Analytics. Er zijn 2 voorgedefinieerde doelgroepen, nl. alle gebruikers en alle kopers. Om specifieke doelgroepen aan te maken kunnen er verschillende “user properties” en “events” geselecteerd worden (screenshot 4.1). Firebase voorziet zelf een aantal user properties o.a. leeftijd en geslacht van de gebruiker, land waarin het toestel zich momenteel bevindt, applicatie versie, ingestelde taal op het toestel, app store versie, merk en model van het toestel, categorie van het toestel, interesses van de gebruiker, LTV van de gebruiker (Life Time Value, dit geeft de waarde weer van de gebruiker gebaseerd op zijn koopgedrag en gebruik van de applicatie) en OS versie (versie van het ondersteuningssysteem van het toestel). De klant is niet gelimiteerd tot de voorgedefinieerde user properties maar kan er ook specifieke door de ontwikkelaar aan laten toevoegen. Een nadeel hiervan is dat eens een user property is aangemaakt ze wel aangepast kan worden maar momenteel nog niet verwijderd kan worden.



(screenshot 4.1)

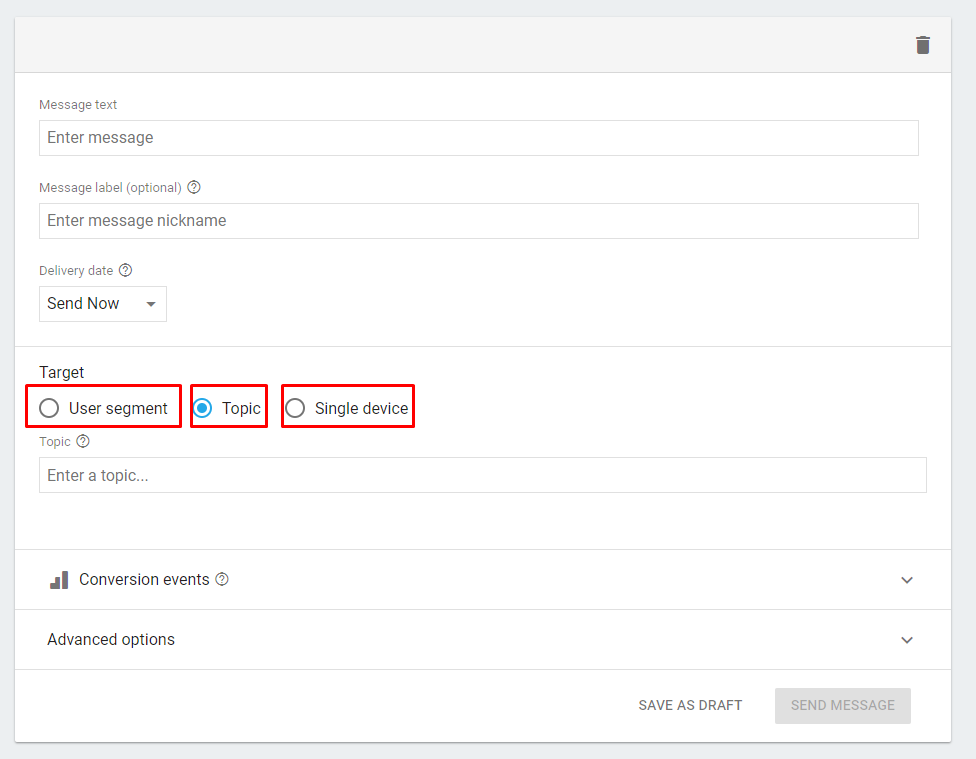
Events aan de andere kant zijn situaties die zich voorgedaan hebben in de applicatie. Bijvoorbeeld een eerste keer de app geopend of een app update of het navigeren naar een bepaald scherm.

Notificaties kunnen verstuurd worden naar enkele toestellen door gebruik te maken van hun FCM ID (Firebase Could Messaging Identification). Dit is vanzelfsprekend uniek voor ieder toestel.

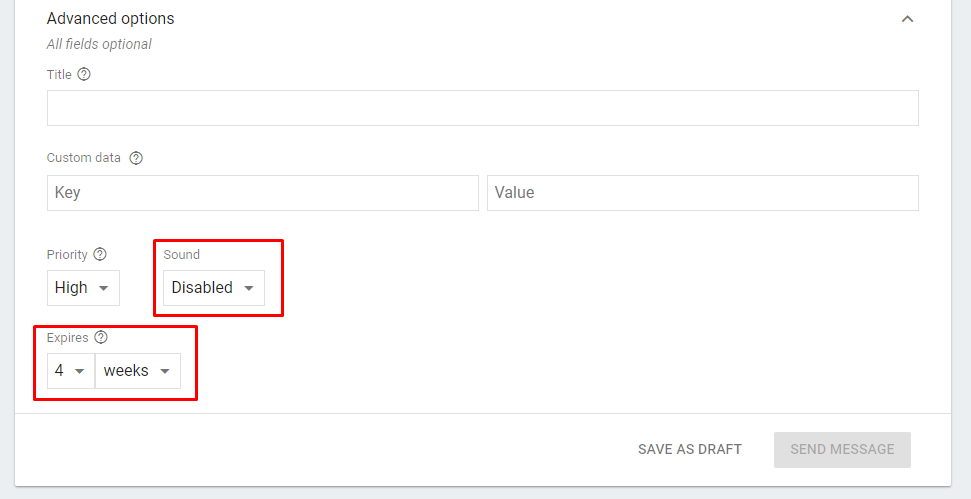
De laatste mogelijkheid om notificaties te versturen is om gebruikers te laten abonneren op bepaalde “topics” (onderwerpen).

Vanuit het Notificatie interface van Firebase kan de klant notificaties versturen naar zijn gebruikers (screenshot 4.2 en 4.3).

Wanneer de klant een notificatie wil verzenden vult hij een titel en een bericht in en kiest naar wie deze notificatie verzonden moet worden. Verder kan er nog gespecifieerd worden of deze notificatie nu of later moet verzonden worden en of dit met of zonder geluid gebeurt. Verder in de Advanced options is er nog de mogelijkheid om een tijd mee te geven wanneer de notificaties zullen vervallen.



(screenshot 4.2)



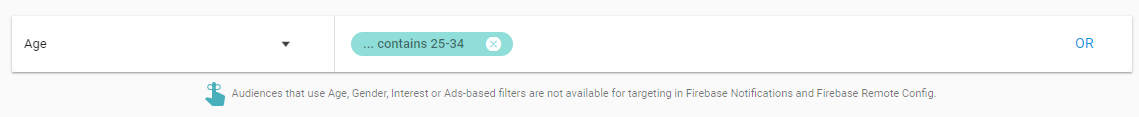
(screenshot 4.3)

Volgende gsm-afbeelding toont een notificatie vestuurd met Firebase naar een topic waarop de gsm is geabonneerd:

|  |  |
| --- | --- |
| C:\Users\SCHUDDINCK\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Screenshot_17.pngDe rode pijl wijst naar de notificatie links bovenaan | C:\Users\SCHUDDINCK\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Screenshot_16.pngDe notificatie toont de melding “test” |

### Besluit

Het gebruik van Notificaties met Firebase is heel eenvoudig en gebruiksvriendelijk, alles wijst zichzelf uit. Een iets complexer proces is het aanmaken van audiences en user properties. Bij te vermelden zijn de restricties op sommige doelgroepen. Doelgroepen die gedefinieerd zijn door een leeftijd, geslacht, interesse of advertentie gebaseerde filters kunnen niet gebruikt worden om notificaties naar te versturen (screenshot 4.4).



(screenshot 4.4)

Buiten Firebase is er altijd de mogelijkheid of “One Signal” te gebruiken. Dit platform is enkel gespecifieerd in het verzenden van push notifications op alle soorten toestellen (iOS, Android, Web).   
Binnenin de applicatie is er ook de mogelijkheid om notificaties te versturen naar de gebruiker maar dit vraagt extra code en gebeurt enkel in voorgedefinieerde situaties zoals bijvoorbeeld wanneer een gebruiker in een game applicatie een bepaald level uitgespeeld heeft, krijgt hij een notificatie van zijn overwinning.

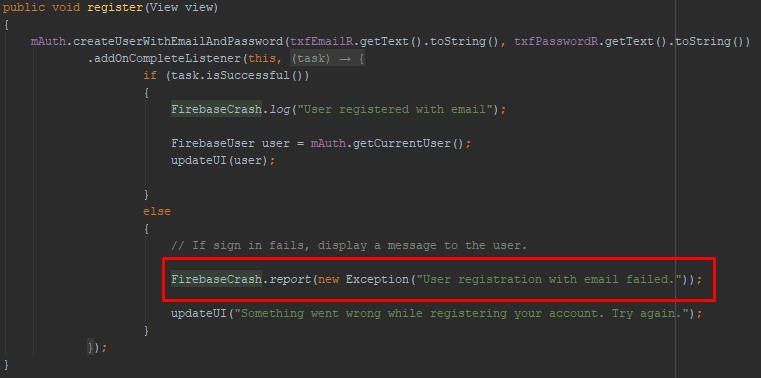
## Crash reporting

### Wat houdt het in en hoe werkt het

Crash reports zijn verslagen (een overzicht van stacktrace en logs) die opgemaakt worden wanneer er zich een crash voordoet in de applicatie. Een crash wordt meestal gezien als een app die ineens wordt afgesloten maar dat is niet alles. Een betere verwoording voor een crash is een onderdeel van een applicatie dat niet naar behoren werkt. Bijvoorbeeld een webpagina dat niet wil laden of een knop die niet werkt.

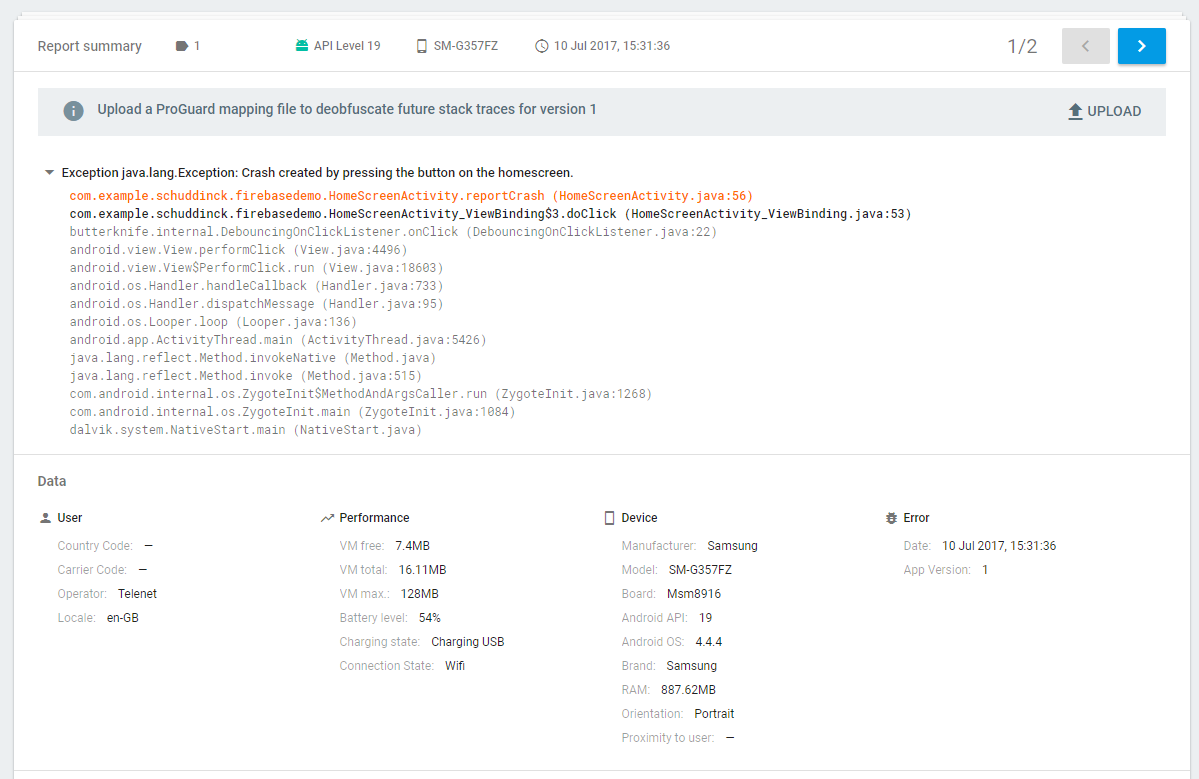
### Hoe werd het getest

In de demo applicatie gemaakt voor deze bachelorproef is er een knop voorzien die een crash report aanmaakt. Op het Crash report interface van Firebase kunnen alle crash reports terug gevonden worden. De ontwikkelaar moet een crash rapport aanmaken op elke mogelijke plaats waar zich een crash kan voordoen (screenshot 4.5).

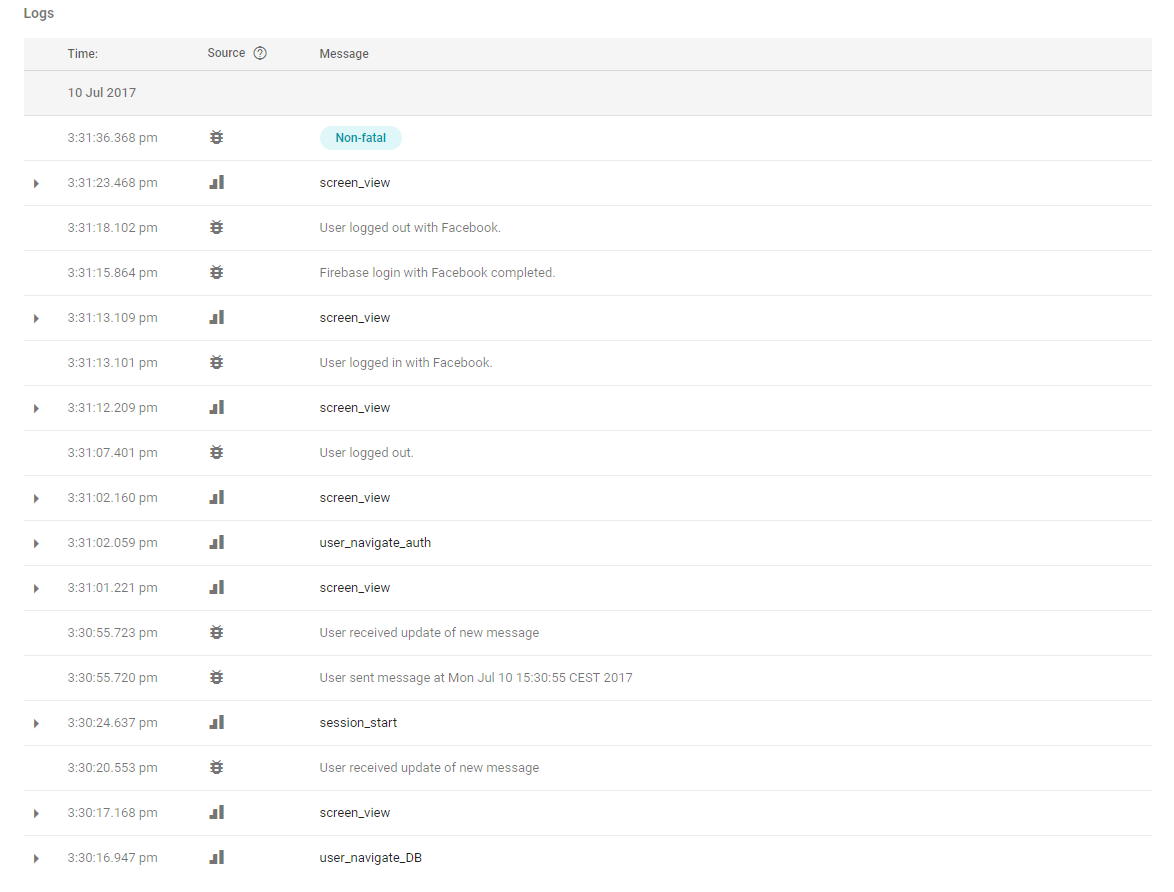


(screenshot 4.5)

Deze staan gesorteerd volgens impact op de gebruiker en samengevoegd volgens gelijkenis. In de applicatie worden er logs van de acties van de gebruiker bijgehouden. Als er zich later een crash voordoet, kunnen de logs overlopen worden in het crash rapport (screenshot 4.6 en 4.7).



(screenshot 4.6)



(screenshot 4.7)

### Besluit

Het is zeer voor de hand liggend om dit op te stellen met Firebase. De code in de applicatie moet wel logs en crash reports bevatten.   
Zonder Firebase is er nood aan een platform zoals bijvoorbeeld “HockeyApp” om crash rapporten te verzenden naar een plaats waar ze geraadpleegd kunnen worden.

## Performance

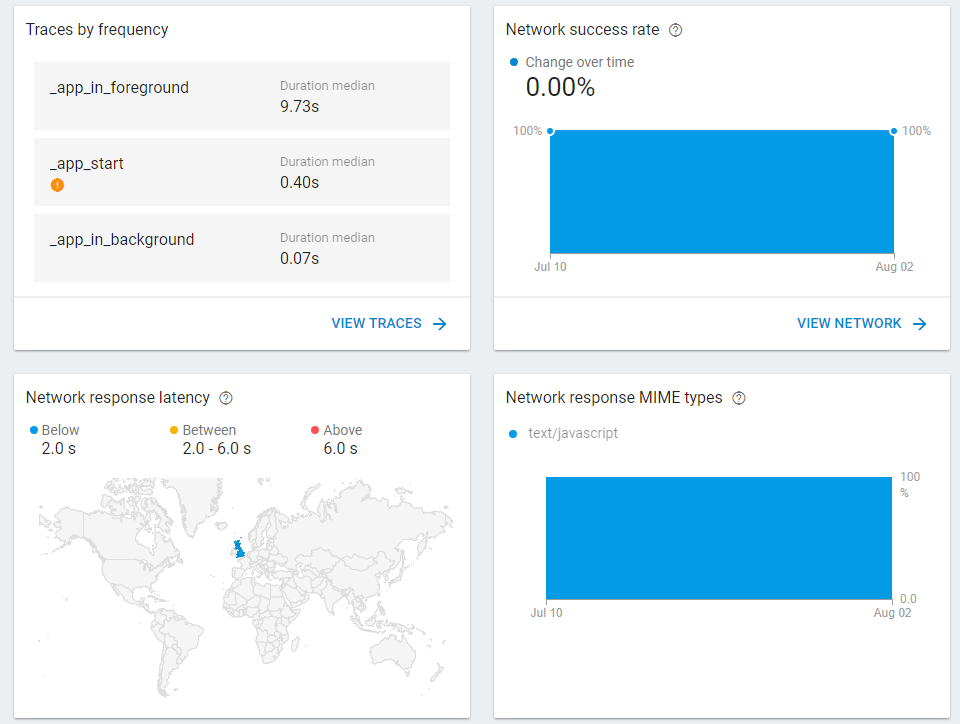
### Wat houdt het in en hoe werkt het

Performance monitoring wordt gebruikt om inzicht te krijgen in de werking van een applicatie. Deze informatie kan gebruikt worden om de uitvoering van de applicatie te optimaliseren. Bijvoorbeeld een optimalisatie bij het opstarten van de applicatie of bij het ophalen van data of bij het afhandelen van geldtransacties.

### Hoe werd het getest

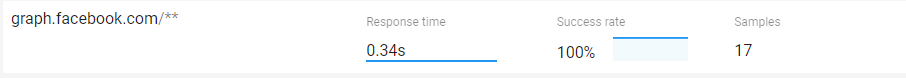
Om Firebase Performance Monitoring te gebruiken in een applicatie moet het gecompileerd worden in de applicatie (compile ‘com.google.firebase:firebase-perf:11.0.2’). Verder moet er niks extra geschreven worden.  
Van hier neemt Firebase Performance Monitoring het over en berekent alles vanzelf zoals de tijd die nodig is om de applicatie op te starten of het ophalen van data (screenshot 4.8) door middel van “traces”. Traces zijn rapporten van performantie gemeten tussen twee punten in tijd in de applicatie. Het maakt gebruik van method calls (zoals de OnCreate en OnResume methodes van een activity) en notifications om het start- en stopmoment van ieder type van automatische trace te bepalen. Volgende types van automatische traces worden door Firebase aangeboden:

* App start traces die de tijd meten tussen het punt waarop de gebruiker de applicatie opent en het punt waarop de applicatie responsief is.
* App in background traces die de tijd meten wanneer de applicatie op de achtergrond loopt.
* App in foreground traces die de tijd meten wanneer de applicatie op de voorgrond loopt en beschikbaar is voor de gebruiker.



(screenshot 4.9)

Naast traces worden er ook https network requests (bv het inloggen via Facebook) opgevolgd (screenshot 4.9).



(screenshot 4.9)

In de traces en network requests kan de ontwikkelaar een filter toepassen om gemakkelijk een bepaalde trace of network request terug te vinden. Er kan gesorteerd worden op:

* Country
* Device
* OS version
* App version
* Radio
* Carrier
* MIME type (Multipurpose Internet Mail Extensions, is een internet standaard die het formaat van emails uitbreidt naar bv. emails met een bijlage)

Alle informatie kan teruggevonden worden op het dashboard van Firebase onder de rubriek Performance.  
Firebase Performance Monitoring bevindt zich op dit moment nog in een beta fase en zal in de toekomst meer eigenschappen krijgen.

### Besluit

Alhoewel Firebase Performance Monitoring zich nog maar in een beta fase bevindt, kan deze tool al zeer handig zijn voor ontwikkelaars om te ontdekken waar bottlenecks voorkomen in applicaties.   
Om aan performance monitoring van applicaties te doen zonder Firebase, is er de optie om de applicatie te koppelen aan New Relic (Relic, Digital Performance Monitoring and Management | New Relic, n.d.). New Relic specialiseert zich in het meten van de uitvoeringen die een applicatie doet en heeft door zijn jaren ervaring, het bestaat immers al sinds 2008, een grote voorsprong op de Performance Monitoring van Firebase.  
New Relic biedt veel onder andere Application Monitoring en Database Monitoring aan om gebruik van te maken (Relic, Application Performance Monitoring Tools and Features | New Relic APM, n.d.).  
Eigenschappen van Application Monitoring:

* Response Time, Throughput en Error rates van web transacties
* Application Histograms and Percentiles
* Most Time-Consuming Transactions
* Performance of External Services (bv. inloggen met facebook)
* Thread Profile (vergelijkbaar met crash reporting van Firebase maar minder gedetailleerd)
* Transaction Metrics and Traces
* Access to Performance Data API (informatie over het ophalen van data)
* Track Key Business Transactions (pro)
* Cross Application Tracing (pro)
* Transaction Breakdown (pro)
* X-Ray Sessions (pro)
* Java Virtual Machine Performance Analyser (pro)
* Service Map (pro)
* Deployment Analysis, History, Comparis (pro)

Eigenschappen van Database Monitoring:

* Time Spent in Database Calls
* Database Call Response Time and Throughput
* Slow SQL Report
* SQL Queary Analysis
* Filter to a specific database/cache type
* See and search all database/cache operations

New Relic mag dan wel zeer veel aanbieden maar aan een hoge kostprijs. Er is wel nog een onderscheid tussen PRO en ESSENTIAL klanten maar het blijft duur.

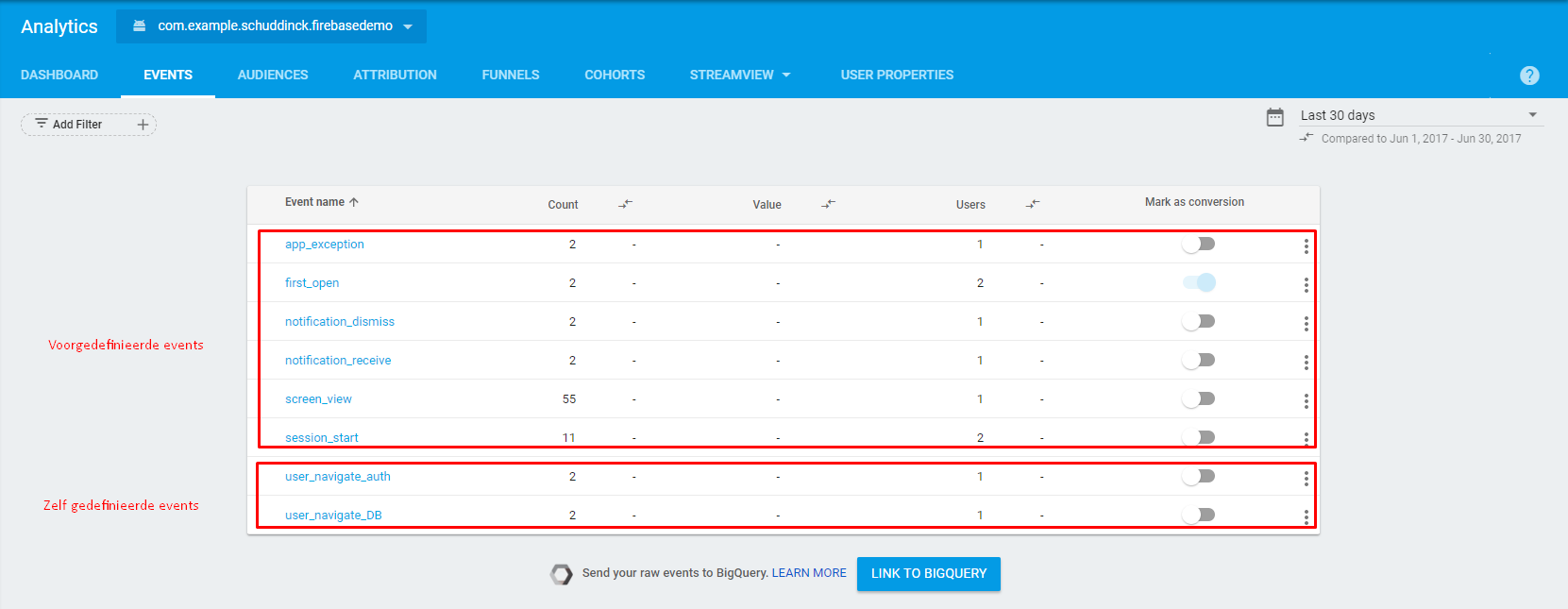
## Analytics

### Wat houdt het in en hoe werkt het

Application Analytics is een real-time analyse van verzamelde data over het gebruik van de applicatie en zijn gebruikers. Dit is nodig om inzicht te krijgen in het business aspect van de applicatie. Zo kan er ingespeeld worden op trends of kunnen er promoties gegeven worden voor een select aantal of alle gebruikers. Dit is dus een handige marketing tool.

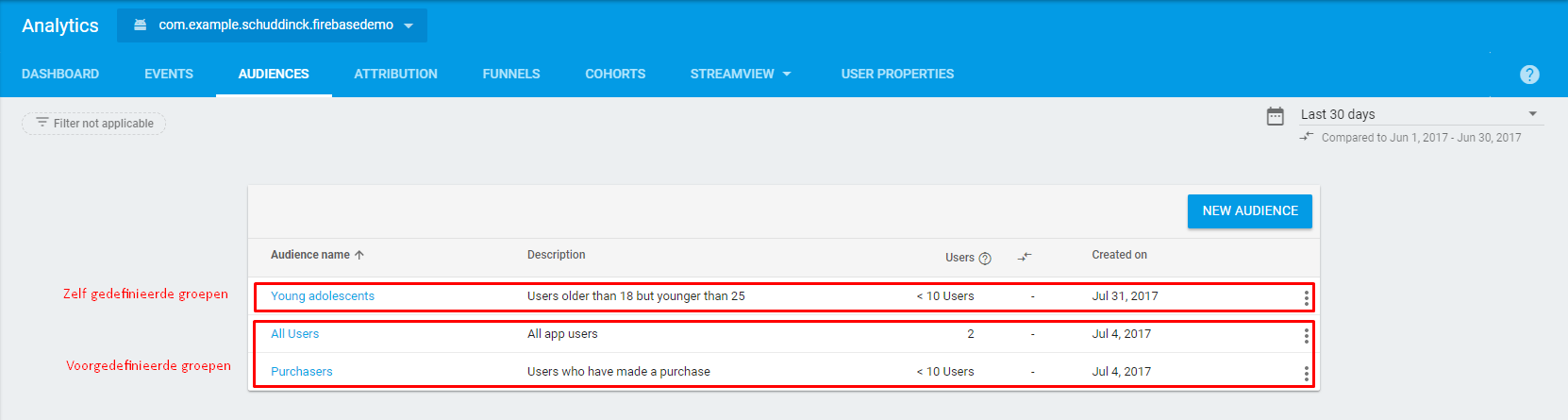
### Hoe werd het getest

Om Firebase Analytics te implementeren moet het enkel gecompileerd worden in de applicatie (compile 'com.google.firebase:firebase-core:11.0.2'). Dit geeft automatisch al toegang tot analytische informatie. Voor meer diepgaande informatie kunnen er custom events en audiences aangemaakt worden.  
Op volgende afbeelding (screenshot 4.10) staan er voorgedefinieerde en zelf gedefinieerde events. Zelf gedefinieerde events werden geprogrammeerd in de applicatie. Dit wordt gedaan door eigen events te loggen, bv. het navigeren naar een nieuw scherm.

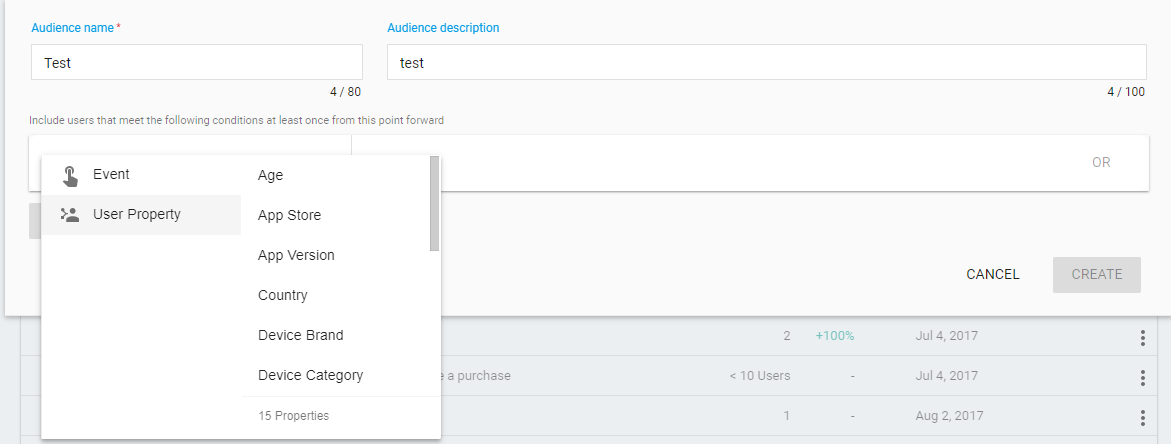


(screenshot 4.10)

Wat betreft audiences, deze moeten aangemaakt worden in de Firebase Console. Automatisch maakt Firebase 2 audiences aan, nl. All users en Purchasers (screenshot 4.11). Er kunnen custom audiences aangemaakt worden door op de knop “New audience” te klikken en daarna te specifiëren door middel van een keuzemenu (screenshot 4.12).



(screenshot 4.11)



(screenshot 4.12)

### Besluit

Analytics aangeboden door Firebase is een zeer uitgebreidde analytische tool en bovendien volledig gratis. Het is gemakkelijk in gebruik om een basis rapport op te stellen, voor een meer diepgaand rapport is enige kennis van analytische informatie natuurlijk vereist.

Een alternatief voor Firebase Analytics zijn Google Analytics maar hierop is Firebase Analytics gebaseerd en Firebase Analytics is over het algemeen beter en uitgebreider maar het grootste voordeel van Google Analytics is is dat het realtime data toont. Firebase Analytics heeft momenteel nog geen realtime analytische data. Google Analytics heeft componenten die Firebase Analytics niet heeft maar dit is te betalen ($12 500 / maand). (zie bijlage “vergelijking Firebase Analytics met Google Analytics)  
Een ander alternatief, buiten Google, is de analytische tool AppDynamics van Cisco. AppDynamics heeft geen optie om zelf analytische data aan te maken met eigen criteria (zoals events en audiences). AppDynamics is te betalen terwijl Google Analytics gratis gebruikt kan worden en als er nood is aan nog meer informatie kan er eventueel gekozen worden voor een betalende versie. Firebase Analytics daarintegen is volledig gratis.

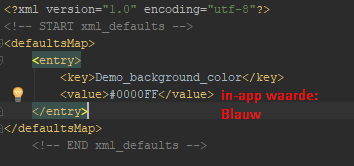
## Remote Config

### Wat houdt het in en hoe werkt het

Remote config kan men vertalen naar een configuratie die van op afstand kan worden veranderd. In het geval van Firebase kan men dus configuraties van de applicatie veranderen vanop afstand door middel van Firebase Console. Met andere woorden er zijn geen updates van de applicatie vereist maar een voorwaarde is wel dat er in de console nieuwe waardes moeten gepubliceerd worden.

### Hoe werd het getest

Om Firebase Remote Config te testen werd er een nieuw scherm in de demo applicatie aangemaakt. Dit nieuwe scherm heeft enkel een achtergrondkleur, die verandert naargelang de waarde die online gepubliceerd wordt in de Remote Config in de Firebase Console. Als de applicatie geen toegang heeft tot het internet (bv. doordat de gebruiker de verbinding met internet heeft afgesloten of simpelweg omdat er nergens een connectie te bespeuren is) wordt de in-app waarde genomen.  
Om te beginnen moet er een nieuw xml-bestand aangemaakt worden die de in-app waarde(s) bevat(ten) (screenshot 4.13).



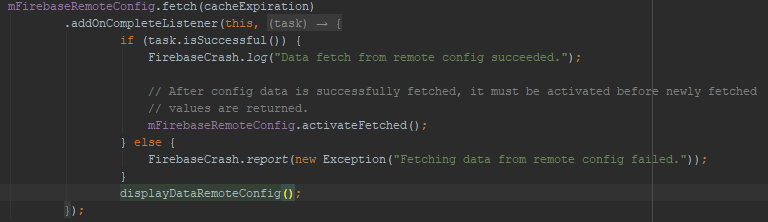
(screenshot 4.13)

Dit bestand heeft als naam “remote\_config\_defaults” gekregen. Dit is belangrijk omdat deze naam gebruikt moet worden tijdens de setup van de remote config in de applicatie (screenshot 4.14) zodat de standaard in-app waardes kunnen gelezen worden.

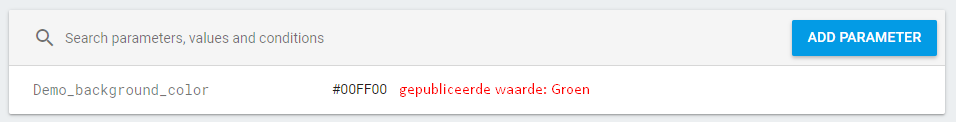
C:\Users\SCHUDDINCK\AppData\Local\Microsoft\Windows\INetCache\Content.Word\RemoteConfig_File.png

(screenshot 4.14)

De volgende stap in de setup (screenshot 4.15) zorgt ervoor dat de applicatie, iedere keer als er naar dit scherm genavigeerd wordt, de gepubliceerde waarde (screenshot 4.16) ophaalt.



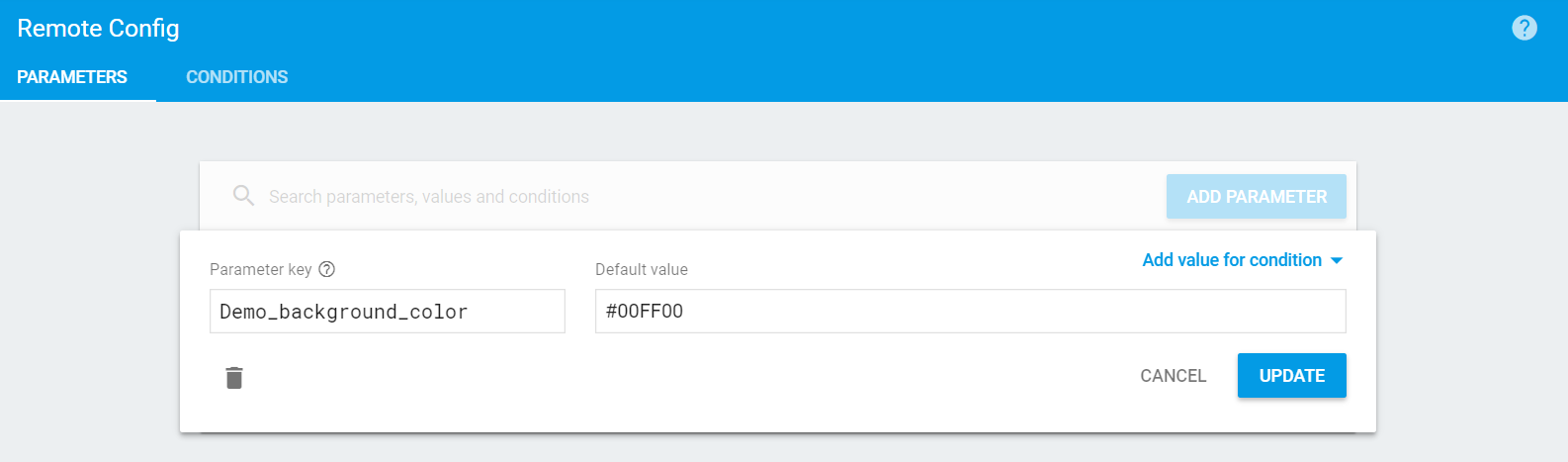
(screenshot 4.15)



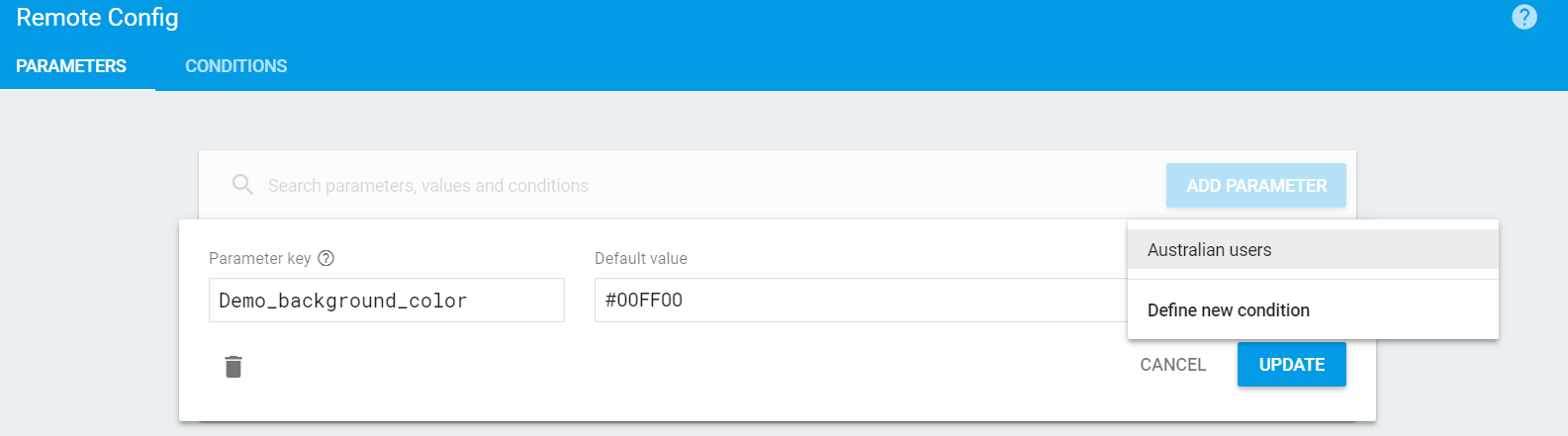
(screenshot 4.16)

Bij succes wordt de gepubliceerde waarde gebruikt om de kleur van het scherm in te stellen (naar groen), bij een mislukking wordt er gebruik gemaakt van de in-app waarde zoals vermeld in het xml-bestand (remote\_config\_defaults). In dit geval zal de kleur van het scherm blauw zijn.  
Niet te vergeten is dat eens er een parameter aangemaakt of veranderd is er nog op de “Publish changes” knop gedrukt moet worden.

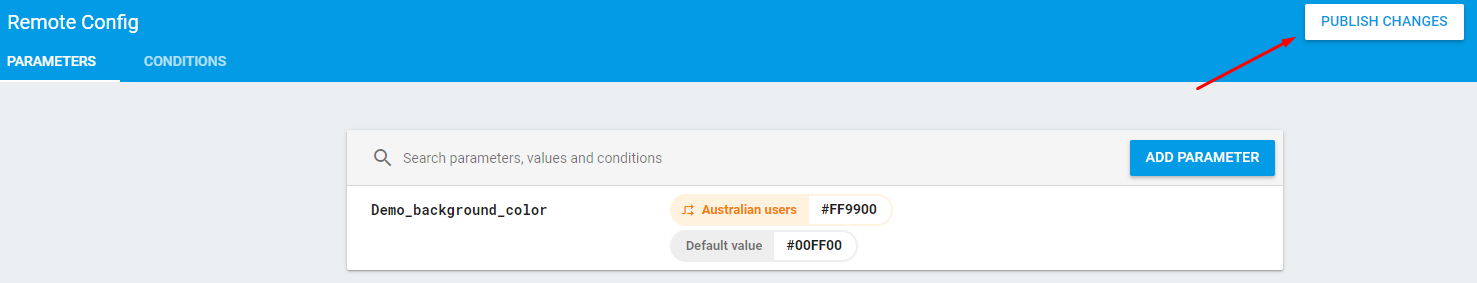
Op de volgende afbeeldingen (screenshot 4.17, 4.18 en 4.19) is een overzicht van de Remote Config te zien. Geselecteerd is de parameter die de kleur van het scherm in de demoapplicatie definieert nl. groen (#00ff00). Hieraan kan een conditie toegevoegd worden door op “add value for condition” te drukken. Dan kan er een andere waarde gepubliceerd worden voor bv. mensen die zich in Australië bevinden en mensen die zich daarbuiten bevinden. Condities moeten zelf aangemaakt worden en er kan gekozen worden uit Audiences of User Properties (standaard [enkel land of ingestelde taal van het toestel] of zelf aangemaakte) of een combinatie van beide. Er is ook de mogelijkheid om meerdere condities toe te voegen aan één parameter.



(screenshot 4.17)



(screenshot 4.18)



(screenshot 4.19)

De laatste stap is natuurlijk om de waardes te publiceren door op de “Publish changes” knop te drukken.

|  |  |  |
| --- | --- | --- |
| C:\Users\SCHUDDINCK\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Screenshot_13.pngHoofdscherm waar later op “Remote Config”-knop gedrukt wordt | C:\Users\SCHUDDINCK\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Screenshot_14.pngOp dit scherm wordt de in-app waarde van de kleur (blauw) van het scherm getoond omdat er geen verbinding is met internet | C:\Users\SCHUDDINCK\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Screenshot_15.pngAls men terug navigeert naar het hoofdscherm vervolgens connectie met het internet toelaat en dan weer navigeert naar het scherm dat remote config gebruikt, wordt de online waarde opgehaald en is het scherm groen |

### Besluit

Remote configuration voor applicaties is een component uniek aan Firebase en bovendien een zeer handige tool. Zo kan de ontwikkelaar van een applicatie bepaalde waarden online publiceren zodat de app voor ieder doelpubliek er anders uitziet en zelfs een ander gedrag kan vertonen.  
Bij de test met de achtergrondkleur van het scherm bleek dat bij het eerste gebruik van de applicatie de standaard in-app kleur getoond werd, nl. blauw, om daarna de gepubliceerde waarde, nl. groen, te tonen. Bij volgend gebruik met of zonder internetconnectie zal de kleur groen direct getoond worden zelf niet als de smartphone heropgestart wordt of the cache geleegd wordt.

## Conclusie Proof-of-Concept

Het maken van een Proof-of-Concept voor deze bachelorproef ging vrij vlot. Een eerste vereiste was het nodige opzoeken op Google Firebase, alles lezen en introductiefilmpjes beluisteren. Dit gaf al heel wat inzicht in de werking van Firebase en heeft geholpen bij de beslissing welke componenten in de demo applicatie zouden opgenomen worden.  
De volgende stap was het effectief uitwerken van een Proof-of-Concept. Hiervoor werd een demo applicatie gemaakt om de verschillende gekozen componenten uit te testen. Met behulp van de documentatie op de site van Google Firebase met daarin duidelijke voorbeelden van hoe de code gestructureerd moet worden werd het programmeren heel gemakkelijk gemaakt voor elk niveau van applicatie ontwikkelaar.

De site van Google Firebase is heel overzichtelijk, duidelijk en gebruiksvriendelijk. Dit maakt het gemakkelijk om componenten in de applicatie te implementeren.  
De tijdspanne om de componenten (Realtime database, authentication, crash reporting, performance monitoring, notifications en analytics) te concretiseren kwam neer op een aantal werkdagen (ongeveer drie tot vier dagen). Wat het meeste tijd vergde was het gebruik opstellen van de realtime database en de authentication, specifiek de connectie met facebook.

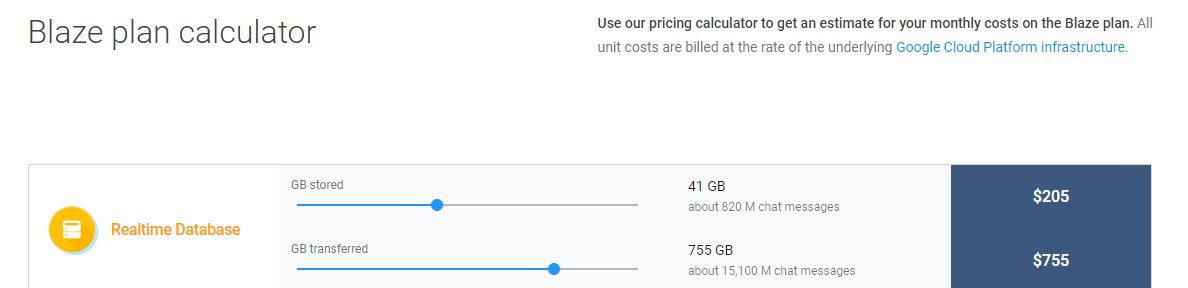
De wachttijd alvorens bepaalde componenten beschikbaar gesteld waren kon oplopen tot twee à drie uur. Bijvoorbeeld bij het aanmaken van topics voor notifications duurde het ongeveer twee tot drie uur alvorens het topic in de console aangemaakt was om notifications naar te versturen.  
Tijdens het wachten op het beschikbaar gesteld worden van bepaalde componenten, is het altijd mogelijk om al te beginnen werken aan de volgende component.  
Elke component werkt op zich maar kan ook samenwerken met andere componenten mits deze geïmplementeerd zijn. Bijvoorbeeld Firebase Analytics in combinatie met notifications.

# Het schrijven van applicatie met en zonder Firebase

Zoals met alle systemen zijn er voor- en nadelen verbonden aan het schrijven van een applicatie zowel met gebruik van Firebase als zonder.  
Het grootste voordeel van Firebase is dat een beginnend ontwikkelaar meteen aan de slag kan en dit zonder dat het één cent kost. Bewijs hiervan is de proof-of-concept.  
Het schrijven van een applicatie zonder Firebase vergt enerzijds meer ervaring en anderzijds zijn er ook heel wat kosten aan verbonden.

## 5.1. Voordelen

### 5.1.1. Met Firebase:

* Alle componenten zijn verzameld onder een dak, zowel qua ontwikkelen als wat het groeipotentieel en marketing betreft.
* Iedere component is duidelijk en overzichtelijk gedocumenteerd. Hij bevat een korte en meer gedetailleerde uitleg in de vorm van geschreven documenten en via korte videofilmpjes.
* Bij elke component wordt vermeld voor welke platformen deze kan gebruikt worden, nl. iOS, Android, C++, web en Unity.
* Er moet geen back-end code geschreven worden voor de databank, dit bespaart enorm veel tijd voor ontwikkelaars, bijgevolg bespaart dit geld voor de klant.
* Er wordt gebruik gemaakt van ingebouwde autorisaties, er moet geen code geschreven worden om de wachtwoorden te beveiligen (hashen).
* Er is een test lab voor Android voorzien, dit betekent dat er geen verschillende toestellen die op Android draaien aangekocht moeten worden om de applicatie te testen.
* Applicaties die gebruik maken van Firebase zijn cloud hosted, m.a.w. men hoeft ze niet zelf te beheren.
* Marketing wordt vergemakkelijkt door gebruik van Firebase Analytics.
* Het adverteren van applicaties (naast de klassieke reclamecampagnes zoals televisie, kranten, affiches en dergelijke) kan nu ook gebeuren door middel van dynamic links. Zo kunnen vrienden elkaar aansporen via de app zelf om applicaties te downloaden en te gebruiken.  
  Bovendien is er nog App Indexing om de applicatie te doen stijgen in de Google Search ranking.
* Code die tot de kern van de applicatie behoort en vatbaar is tot reverse engineering kan in de cloud opgeslagen worden zodat deze daartegen behoed is.
* Het gedrag en uitzicht van de applicatie kan aangepast worden door middel van Remote Config zodat verschillende groepen van gebruikers zich kunnen bedienen van een bij hen passende applicatie.
* Remote Config zorgt er ook voor dat ingeslopen fouten gemakkelijk kunnen gecorrigeerd worden zonder daarvoor een update uit te moeten brengen.
* Er zijn 3 tariefplannen, een gratis (Spark), een vast maandelijks bedrag van $25 (Flame) en een Pay-as-you-go (Blaze). Het laatste kan men zelf berekent met behulp van de Blaze Plan Calculator (screenshot 5.1) zodat er geen verborgen kosten zijn. (screenshot 5.1)
* Het gratis plan is perfect voorzien om de app uit te testen bij een klein publiek (maximaal 100 actieve gebruikers) zodat er geen extra kosten zijn tijdens het ontwikkelen van de applicatie. Eens de applicatie op de markt wordt uitgebracht is er natuurlijk de mogelijkheid om te upgraden naar één van de twee andere pakketten (Flame of Blaze).
* Firebase bestaat al sinds 2011 maar is overgenomen door Google en hernieuwt met een heleboel handige componenten. Aangezien het nog maar net gelanceerd is door Google mogen we ons verwachten aan nieuwe componenten of uitbreidingen.

### 5.1.2. Zonder Firebase:

* Bij nood aan diep gedetailleerde rapporten kunnen andere platformen gebruikt worden. Bijvoorbeeld voor performance monitoring bestaat er New Relic.

## 5.2. Nadelen

### 5.2.1. Met Firebase:

* Er is ondersteuning voorzien voor maar 10 vragen met troubleshooting per jaar (andere opties zijn fora stackoverflow, firebase slack, …). Er moeten dus prioriteiten gesteld worden op vlak van welke vragen aan de Firebase support gesteld worden en welke via andere kanalen kunnen opgelost worden.
* Het gratis plan geldt enkel voor de eerste 100 actieve gebruikers en kan dus enkel kleinschalig gebruikt worden. Eens de applicatie op de grote markt komt is er nood aan overschakeling naar één van de andere tarieven (Flame of Blaze).
* De cloud hosted data heeft ook een keerzijde. Als het cloud platform van Google problemen vertoont, is het buiten de controle van de klant of ontwikkelaar. Echter Google is een grote speler dus hebben ze hun naam hoog te houden en zullen zorgen dat eventuele downtime zo snel mogelijk verholpen wordt.
* Firebase door Google, zoals daarnet vermeld, is vrij recent gelanceerd. Dit kan zorgen voor veranderingen waardoor de applicatie moet vernieuwd worden.
* Het nieuwe Firebase platform is ook nog niet geoptimaliseerd voor third party platformen zoals Xamarin. Sommige componenten werken nog niet naar behoren zoals de realtime database voor Xamarin Android.
* Eens je over het limiet van een plan zoals Spark of Flame bent worden datatransfers stopgezet voor de rest van de lopend maand. Tijdig uitkijken naar een upgrade is dus nodig.
* Bij Firebase Analytics is er de keuze om “User properties” aan te maken. Echter deze kunnen niet verwijderd worden en er kunnen er maximaal 25 aan toegevoegd worden.
* Crash Reporting is er enkel voor iOS en Android. Wat crash reporting voor C++, web en Unity betref, moet er dus een alternatief gezocht worden.

### 5.2.2. Zonder Firebase:

* Zonder Firebase moet er wel een eigen backend geschreven worden en dit kost vanzelfsprekend veel extra tijd en geld.
* Alle componenten moeten apart op het internet moeten worden opgezocht om vervolgens te bestuderen en eventueel met elkaar te vergelijken. Dit neemt meer tijd en geld in beslag.
* Wanneer er toch alternatieven (Realm, New Relic, AppDynamics, …) gevonden zijn voor elke component zijn die niet noodzakelijk met elkaar compatibel.
* Ieder component apart heeft zijn eigen prijskaartje en dit drijft uiteraard de totale kost de hoogte in.
* AWS (Amazon Web Service) is een soortgelijk overkoepelend platform als Firebase maar is niet altijd een goed alternatief voor elke applicatie. AWS is meer gemaakt met de bedoeling om een ondersteuning te bieden voor webapplicaties (AWS, n.d.).

# Bevraging bij bedrijven

## 6.1. Vraagstelling

Firebase door Google is sinds eind 2016 op de Google I/O openbaar gemaakt. Een tweede luik van deze bachelorproef is om na te gaan hoe bekend Firebase al is bij bedrijven die applicaties ontwikkelen. Daarom is de volgende vragenlijst (opgesteld met Google Docs) verstuurd naar een vijftigtal bedrijven (Sortlist, n.d.).  
De bevraging van de bedrijven verliep volgens de volgende stadia:

1. Situering
   1. Heeft u ooit al gehoord van Firebase? Zo ja, waar?  
      Have you ever heard of Firebase? If so, where?
2. Introductie van Firebase (met behulp van een kennismakend filmpje)
   1. Introductie Firebase (Dit is een kort kennismakend filmpje van Firebase om de bedrijven die Firebase niet kennen op de hoogte te brengen).  
      Indien het bedrijf in kwestie Firebase niet kent werd er gevraagd of er een interesse gewekt was om Firebase uit te testen en later eventueel aan klanten voor te stellen.
3. Gebruik van Firebase (bedoelt voor bedrijven die al gebruik gemaakt hebben van Firebase)
   1. Heeft u Firebase ooit gebruikt in een of meerdere projecten  
      Have you ever used Firebase in one or more projects
   2. Van welke features, aangeboden door Firebase, heeft u al gebruik gemaakt?  
      Which features, offered by Firebase, have you used before?
   3. Welke features, aangeboden door Firebase, zijn voor u het meest interessant?  
      Which features, made available by Firebase, are the most interesting to you and your company
4. Algemeen
   1. Heeft u interesse om Firebase in toekomstige projecten te gebruiken? Zo ja, waarom?  
      Are you interested in using Firebase in the future? If so, why?
   2. Waarom is Firebase, volgens u, wel of niet beter dan soortgelijke technologieën (Azure, AWS)?  
      Why is Firebase, according to you, better (or worse) than similar technologies (Azure, AWS)?
   3. Zou u het gebruik van Firebase aanraden bij andere bedrijven? Waarom (niet)?  
      Would you recommend Firebase to other companies and clients? Why (not)?

Deze vragenlijst werd bewust kort gehouden zodat bedrijven niet zouden afgeschrikt worden door een ellenlange vragenlijst. Ook is ze blind zodat bedrijven in alle anoniemiteit en eerlijkheid deze vragenlijst zouden beantwoorden. Tevens werden de vragen naar het Engels vertaald omdat bleek dat toch in heel wat bedrijven in België, vooral Brussel en Wallonië, men het Nederlands niet altijd machtig is. Daar komt nog bij dat sommige bedrijven hun applicatieontwikkelaars vanuit het buitenland werken.

## 6.2. Respons bevraagde bedrijven

De respons van de bedrijven verliep tegen alle verwachtingen in heel moeizaam. De volgende tactieken werden toegepast.  
Eerst werden een twintigtal kleinere mobile app development bedrijven in Vlaanderen per telefoon gecontacteerd om een specifiek e-mailadres van één van hun programmeurs te verkrijgen waarna deze persoon de link naar de vragenlijst kreeg toegestuurd. Dit leverde een behoorlijke response op.

Vervolgens werd de bevraging uitgebreid naar de rest van België, inclusief grotere bedrijven. In totaal werden een zestigtal bedrijven gecontacteerd. Allen werden eerst telefonisch gecontacteerd. Om daarna de link naar de vragenlijst door te sturen. Wat opviel was dat grotere bedrijven minder bereid waren specifieke e-mailadressen vrij te geven maar enkel het algemene e-mailadres. Zo kon niet gecontroleerd worden of de vragenlijst effectief bij de juiste personen aankwam.

De voornaamste redenen waarom het proces van bevraging zo moeizaam verliep zijn als volgt:

* Enquête-moeheid: mensen worden dagelijks overstelpt met vragenlijsten allerhande en zijn het moe om nog maar eens een bevraging in te vullen.
* Geen tijd: programmeurs werken naar deadlines toe en hebben niet altijd de tijd om hun werktijd te stoppen in iets dat niet werk gerelateerd is.
* Outsourcing: Grotere bedrijven werken met IT-afdelingen in het buitenland en zijn niet bereid om hun e-mailadressen door te spelen of e-mails door te sturen.  
  Zelfs als de bedrijven willig zijn om e-mails door te sturen naar hun programmeurs in het buitenland zijn de programmeurs zelf niet altijd even bereid om de vragenlijst in te vullen.
* Overbevraging: sommige bedrijven krijgen zoveel aanvragen van studenten dat ze besloten hebben om geen enkele nog te beantwoorden.
* Vakantieperiode: veel programmeurs waren niet aanwezig door vakantieplannen.

De respons verliep enerzijds moeizaam maar anderzijds waren sommige bedrijven enorm geïnteresseerd in het resultaat van mijn bachelorproef en vroegen hiervan een kopie.

## Analyse van de antwoorden

### Vraag 1: Heeft u ooit gehoord van Firebase? Zo ja, waar?

De antwoorden hierop waren overwegend positief. De meeste bedrijven kennen Google Firebase door blogs, newsfeeds, hun werk (collega’s en concurrenten) of Google I/O.  
Één van de bedrijven werktte al met Firebase voor het overgenomen werd door Google omdat ze zochten naar een realtime database.

### Vraag 2: Indien neen op de eerste vraag: Met de gegeven uitleg zouden jullie overwegen Firebase uit te testen en bij positieve testen het toepassen ervan aan potentiële klanten voor te stellen?

Uit de antwoorden op deze vraag blijkt duidelijk dat over het algemeen wie het niet kent er ook niet voor open staat om het in de toekomst uit te testen of aan potentiële klanten voor te stellen. Dit kan liggen aan het feit dat er geen tijd genomen werd om het introductiefilmpje te bekijken of ze er zich vlug wouden van afmaken.  
Anderen waren iets positiever in de zin dat ze het gebruik van Firebase zouden overwegen als de voordelen zouden opwegen tegen het gebruik van andere systemen zoals AWS. Deze bedrijven hebben dan ook gevraagd naar de resultaten van deze bachelorproef.

### Vraag 3: Heeft u Firebase ooit gebruikt in 1 of meerdere projecten?

De meeste bedrijven die Google Firebase kennen, hebben Firebase al voor één of meerdere projecten gebruikt behalve diegene die het enkel intern gebruiken.

### Vraag 4: Van welke features, aangeboden door Firebase, heeft u al gebruik gemaakt?

De componenten van Google Firebase die het meest populair zijn in dalende volgorde:

1. Realtime database
2. Authentication
3. Crash reporting en notifications
4. Hosting en Cloud Messaging
5. Dynamic links en AdMob

Wat overduidelijk niet gebruikt wordt zijn devolgende:

* Invites
* AdWords
* App Indexing

### Vraag 5: Welke features, aangeboden door Firebase, zijn voor u en uw bedrijf het meest interessant?

De meest interessante componenten voor bedrijven zijn volgende in dalende volgorde:

1. Realtime database
2. Authentication
3. Notifications
4. Cloud Messaging
5. Crash reporting

Niet interessant voor de bedrijven zijn de volgende:

* Invites
* AdWords

Zoals te verwachten liepen de antwoorden op deze vraag bijna volledig gelijk met de antwoorden op de vorige vraag.

### Vraag 6: Heeft u interesse om Firebase in toekomstige projecten te gebruiken? Zo ja, waarom?

Uit antwoorden blijkt dat wie het enkel intern gebruikt, Firebase ofwel niet zal gebruiken voor toekomstige projecten ofwel heerst er nog twijfel.  
Andere bedrijven zitten mogelijks gebonden aan vaste partners voor bijvoorbeeld hosting (denk maar aan Azure) of hebben een eigen backend geschreven of maken gebruik van soortgelijke systemen zoals AWS.  
Het merendeel van de bedrijven is zeer positief en zou het zeker gebruiken voor toekomstige projecten om de volgende redenen:

* Kostprijs (goedkoop)
* Gebruiksvriendelijkheid
* Snelle opbouw
* Vele mogelijkheden
* Stabiel in gebruik
* Goede realtime database
* Duidelijke API

Verder blijkt wel dat het gebruik van de onderdelen afhangt van project tot project.

### Vraag 7: Waarom is Firebase, volgens u, wel of niet beter dan soortgelijke technologieën (Azure, AWS)?

De antwoorden op deze vraag liggen ongeveer in dezelfde lijn als de antwoorden op de vorige vraag. Wie het enkel intern gebruikt is moeilijk te overtuigen. Zij gebruiken ofwel een eigen backend of andere technologieën zoals AWS. Waar ze een probleem van maken is het feit dat AWS ruwweg dezelfde mogelijkheden biedt als Firebase maar ook nog SDK’s voor Windows.  
De bedrijven die het wel gebruiken, vinden het beter om de volgende redenen:

* De lage kostprijs
* Toegankelijkheid
* De instapdrempel is heel laag
* Het ruime aanbod aan componenten
* All-in-one platoform waar veel energie is ingestoken
* Overtuiging dat aangezien het een Google product is, dit nog verbeterd zal worden en dat het altijd zal ondersteund blijven

Verder zijn er nog bedrijven wiens antwoord meer genuanceerd is, deze schrijven dat alles afhangt van de eigen backend en hoe gemakkelijk of moeilijk het is om hiermee te integreren.  
Ten slotte waren er nog diegene die wel positief tegenover Google Firebase staan maar momenteel er momenteel niet genoeg kennis van hebben door het gebruik van soortgelijke systemen (AWS).

### Vraag 8: Zou u het gebruik van Firebase aanraden bij andere bedrijven en klanten? Waarom (niet)?

Ook deze antwoorden liggen in dezelfde lijn als de antwoorden op de vorige vragen. De meeste bedrijven waren positief. Vooral wat betreft kostprijs, betrouwbaarheid, gebruiksvriendelijkheid, goede ervaring met realtime database, hosting, crash reporting en tijdsbesparing.  
Bezwaren om het wel of niet aan te raden aan andere bedrijven zijn het wel of niet hebben van de mogelijkheid of de ervaring om een eigen backend te schrijven en de noden van de klant in het algemeen.  
Daarna zijn sommige bedrijven bezorgt om de privacy wetgeving van Europa (GDPR, General Data Protection Regulation, (Europese Unie, n.d.) ).  
Bedrijven die het niet gebruiken vinden dat ze te weinig ervaring hebben om het te kunnen aanbevelen aan andere bedrijven.

## Besluit van de bevraging bij bedrijven

Google Firebase is vrij bekend bij de bevraagde bedrijven hoewel het nog geen jaar geleden is dat het gelanceerd werd (oktober 2016).   
Een klein aantal bedrijven gebruiken Firebase niet of alleen intern.   
Bedrijven die reeds geïnvesteerd hebben in het opzetten van een eigen backend of werken met een soortgelijk systeem als Firebase (bijvoorbeeld AWS) zijn minder happig om over te schakelen naar Firebase tenzij ze zich er een voordeel mee kunnen doen. Toch staan ze open voor Google Firebase. Bewijs hiervan is de vraag naar de resultaten van deze bachelorproef.  
Het merendeel gebruikt de voornaamste componenten van Firebase zoals realtime database, authentication, crash reporting en cloud messaging. Zij hebben het al gebruikt voor meerdere projecten, hetzij klein of groot.  
Hun enthousiasme komt vooral door de snelle opbouw, lage instapdrempel, kostprijs en het feit dat er breed scala aan componenten onder één dak beschikbaar is.  
Firebase is ideaal voor het creëeren van demo applicaties die snel klaar moeten zijn voor bijvoorbeeld beurzen of klanten. Uit gesprekken bleek ook nog dat de Firebase community een extra voordeel kan betekenen voor het oplossen van problemen, ideëen en ervaringen uit te wisselen.  
Ontwikkelaars kijken uit naar de geplande updates van Firebase die aangekondigd werden op de Google I/O Extended die plaatsvond in Amsterdam in mei 2017.

Concreet kan vastgesteld worden dat wie Firebase al gebruikt heeft het zal blijven gebruiken en het ook aanraadt aan andere bedrijven om de bovengenoemde redenen. Natuurlijk wordt er wel bij vermeld dat Firebase niet de oplossing is voor alle projecten en dus moet het bekeken worden van project tot project.

Wat ook vermeld moet worden is dat de bevraging stroef verliep. Alle bedrijven moesten uiteindelijk eerst telefonisch gecontacteerd worden om enige reactie te verkrijgen. Ervaring heeft mij geleerd dat wanneer mij aangespoord werd om een mail te sturen naar het algemeen e-mailadres ik al met zekerheid kon zeggen dat hierop niet gereageerd zou worden. Dit was duidelijk te zien bij de opvolging van de antwoorden op Google Forms.  
Bij direct contact met de ontwikkelaar verliepen de gesprekken zeer vlot en kreeg ik ook meteen respons op Google Forms. Dikwijls werd er gevraagd waar ik hun bedrijf gevonden had aangezien dit belangrijk is voor hun marketing.

# Algemeen besluit

Deze bachelorproef heeft als bedoeling inzicht te verschaffen in Google Firebase. De beschrijving van elk van haar componenten maken duidelijk wat Google Firebase vandaag voor de ontwikkelaars van mobiele applicaties kan betekenen.

Uit de proof-of-concept blijkt dat Google Firebase goed gedocumenteerd is door middel van verklarende teksten, introductiefilmpjes en code voorbeelden om de ontwikkelaar alvast op weg te zetten. De site van Google Firebase is heel overzichtelijk en gebruiksvriendelijk, wat maakt dat het implementeren van de voornaamste componenten voor de proof-of-concept heel vlot en snel verliep.

Uit de vergelijkende studie blijkt dat het ontwikkelen van een standaard applicatie zonder Firebase meer ervaring, hulpmiddelen en kosten vergt. De beschikbare alternatieve componenten zijn zowel apart als in sommige pakketten (zoals AWS) te vinden. Aparte componenten zijn niet vanzelfsprekend met elkaar compatibel en een overkoepelend systeem zoals AWS heeft meer mogelijkheden maar vooral voor het opstellen van webapplicaties en is dus een minder goed alternatief voor mobiele applicaties. Of er nu gebruik gemaakt wordt van een eigen backend of gebruik gemaakt wordt van aparte componenten of overkoepelende systemen, allen kosten meer tijd, ervaring en geld dan Google Firebase.

Uit de bevraging van de bedrijven blijkt dat Google Firebase al goed gekend is onder de bedrijven in België. Velen hebben er van gehoord op de Google I/O van eind 2016 en mei 2017. De meesten gebruiken de voornaamste componenten zoals authentication, realtime database, crash reporting en cloud messaging. Bijgevolg hebben ze Firebase al toegepast in meerdere projecten al zij het klein of groot. Hun enthousiasme komt vooral door de snelle opbouw, lage instapdrempel, kostprijs en het feit dat er breed scala aan componenten onder één dak beschikbaar is.  
Ook het feit dat de Google Firebase community toeneemt, is een bonus.   
Een klein aantal werkt reeds met een eigen backend of met andere systemen zoals AWS en zij vertonen nog een afwachtende houding.

Firebase brengt heel wat voordelen met zich. Denk maar aan de tijdswinst, de gebruiksvriendelijkheid, de lage kostprijs, de lage instapdrempel, de komende updates, het gemakkelijk onderhoud en een betere controle over de applicatie.   
Extra voordelen die in de kijker gezet moeten worden zijn toch wel dat een junior ontwikkelaar geen moeite heeft om met Firebase aan de slag te gaan en dat kleine applicaties voor bijvoorbeeld beurzen heel snel opgezet kunnen worden.   
Uit dit alles kan men afleiden dat Firebase meer en meer gebruikt zal worden door de Belgische ontwikkelaars van mobiele applicaties.   
Het valt te verwachten dat er met Firebase een waardige concurrent bijgekomen is op de markt voor mobiele applicaties.

# Referentielijst

Android. (sd). *Notifications | Android*. Opgehaald van Android developers: https://developer.android.com/guide/topics/ui/notifiers/notifications.html

AWS. (sd). *Getting Started with AWS documentation*. Opgehaald van aws.amazon.com: https://aws.amazon.com/documentation/gettingstarted/

Digicert. (sd). *Just How Strong is 2048-bit SSL Certificate Encryption?* Opgehaald van Digicert: https://www.digicert.com/TimeTravel/math.htm

Europese Unie. (sd). *Home Page of EU GDPR*. Opgehaald van EUGDPR: http://www.eugdpr.org/

Google. (sd). *About the Google Display Network - AdWords Help*. Opgehaald van support.google.com: https://support.google.com/adwords/answer/2404190?hl=nl

Google. (sd). *Documentation | Firebase*. Opgehaald van Firebase: https://firebase.google.com/docs/

Realm. (sd). *Realm*. Opgehaald van Realm Mobile Platform: https://realm.io/products/realm-mobile-platform/

Relic, N. (sd). *Application Performance Monitoring Tools and Features | New Relic APM*. Opgehaald van New Relic APM: https://newrelic.com/application-monitoring/features

Relic, N. (sd). *Digital Performance Monitoring and Management | New Relic*. Opgehaald van New Relic: https://newrelic.com/

Sortlist. (sd). *De 10 Beste Mobile App Development Bureaus in België (2017) - Sortlist*. Opgehaald van Sortlist: https://www.sortlist.be/nl/mobile-app-development/belgie-be

trade, F. i. (sd). *Mobile application development in Flanders*. Opgehaald van Flanders investment and trade: https://www.flandersinvestmentandtrade.com/invest/en/sectors/digital-society/mobile-applications

# Bijlages