

# flashback

**Organisatie**  
Erasmushogeschool  
Brussel

**Opleiding**  
Multimedia & Creative  
Technologie



**Uitgave**  
Anderlecht, Brussel

**Academiejaar**  
2022 - 2023

**Promotor**  
Bert Heyman



## **Bachelorproef 3BaMCT**

Auteur: Somrad Sharma

Studentnummer: 201888178

E-mailadres: somrad.sharma@student.ehb.be

Opleiding: Multimedia & Creative Technologie

Organisatie: Erasmushogeschool Brussel

Promotor: Bert Heyman

19 juni 2023, Anderlecht, Brussel

# Abstract

Deze paper onderzoekt de behoefte aan verbeterde familiale communicatie in een tijdperk waarin sociale media ons meer dan ooit met elkaar verbinden. Hoewel dit connectiviteit en sociale interactie vergemakkelijkt, worden er ook onopgemerkte problemen en tekortkomingen blootgelegd, met name op het gebied van familiegerichte communicatie.

Bestaande platformen zijn vaak gericht op een mondial publiek en missen de specifieke functionaliteiten die families nodig hebben. Dit gat in de markt vormde de motivatie voor mijn onderzoek en de ontwikkeling van een oplossing.

Dit paper maakt deel uit van mijn bachelorproef in de studierichting “Multimedia & Creative Technologie”. Het

beschrijft het onderzoek, het proces en het eindresultaat van mijn werk. Tijdens mijn onderzoek heb ik ontdekt dat het ontwikkelen van een mobiele applicatie met behulp van React Native de beste aanpak was.

Ik heb ook gebruikgemaakt van het Firebase-platform, waarbij ik maakte gebruik van de real-time database, authenticatie en storage.

Gezien mijn beperkte ervaring met deze technologieën, moest ik uitgebreid onderzoek doen om ze effectief toe te passen in de ontwikkeling van de applicatie.

Na de analyse van verschillende aspecten begon ik het ontwerpen van de functionaliteiten van de app. Ik creëerde een samenhangende

huisstijl, wireframes, een logo en uiteindelijk het definitieve design.

Het resultaat van mijn werk is een mobiele applicatie die familieleden in staat stelt om op een verbeterde en meer gepersonaliseerde manier met elkaar te communiceren.

Door de specifieke behoeften van families te adresseren en gebruik te maken van de technologieën, hoop ik een positieve impact te hebben op de manier waarop gezinsleden met elkaar verbonden blijven.

# Voorwoord

Deze bachelorproef markeert voor mij een project waarvoor ik tegelijkertijd angst en enthousiasme voelde. Dit is één van de eerste dingen die in je opkomt als je denkt aan hogere studies, en aangezien het ook mijn eerste ervaring is met het ontwikkelen van een mobiele applicatie, veroorzaakt dat een mix van angst en opwinding.

Hoewel dit gezien wordt als mijn bachelorproef, is het niet slechts mijn individuele inspanning. Ik had dit project niet kunnen volbrengen zonder de hulp en ondersteuning van de mensen om mij heen.

Ik wil iedereen bedanken die heeft bijgedragen aan het succesvol afronden van mijn afstudeerproject. Allereerst wil ik mijn docenten van de Erasmushogeschool Brussel

bedanken. Door hun lessen en begeleiding heb ik de kennis en vaardigheden verworven die ik heb toegepast in mijn bachelorproef. Speciale dank gaat uit naar mijn begeleider, Bert Heyman. Zijn constructieve feedback en begeleiding hebben me in staat gesteld om mijn bachelorproef efficiënter en doeltreffender uit te werken.

Daarnaast wil ik mijn medestudenten bedanken die deel uitmaakten van mijn groep voor de roundtable gesprekken. Hun waardevolle feedback was enorm behulpzaam.

Ik wil ook Hajar Lamri Saifi, Widad Annoud, Rayan Khare en Nadia Saghir bedanken voor het testen van mijn applicatie en het geven van hun feedback. Hun inzichten waren zeer waardevol en hebben me

enorm geholpen.

Mijn ervaring met het maken van deze bachelorproef heeft me zowel uitgedaagd als beloond, en ik ben dankbaar voor iedereen die deel heeft uitgemaakt van deze reis.

# Con

8

Inleiding

Probleemstelling

Oplossing

flashback

12

Analyse

Concurrentie

Doelgroep

Functionele

MoSCoW &  
Technische

18

Design

Huisstijl

Logo

Sketch & Mockup

# tent

## 24

Development

Tools

Packages

Moeilijkheden &  
Oplossingen

Technisch Rapport

## 30

Besluit

## 31

Literatuurlijst

# Inleiding

## Probleemstelling

In de moderne wereld van vandaag verbindt technologie ons op manieren die we ons vroeger niet konden voorstellen. Sociale media en andere digitale communicatiemiddelen zijn de norm geworden. Maar deze technologieën zijn meestal gericht op algemene communicatie en houden vaak geen rekening met wat families echt nodig hebben. Ze zorgen soms zelfs voor droge, oppervlakkige interacties.

Terwijl onze samenleving snel groeit en steeds meer vertrouwt op technologie, wordt het steeds duidelijker dat we behoefté hebben aan meer persoonlijke en intieme communicatiemiddelen.

## Onderzoeksraag

**“Hoe kan technologie bijdragen aan het verrijken van familie-interacties en het beperken van digitale overconsumptie?”**

Studies tonen aan dat te veel gebruik van digitale apparaten bij kinderen kan leiden tot mentale problemen zoals depressie en angst (Anxiety Centre, 2021). Tegelijkertijd kunnen ouders waardevolle adviezen vinden via sociale media, maar soms kan dit leiden tot gevoelens van minderwaardigheid wanneer ze zichzelf vergelijken met anderen.

Dit alles benadrukt de noodzaak van technologie die specifiek is ontworpen voor families. We hebben technologie nodig die de kwaliteit van interacties tussen familieleden kan verbeteren, en die de ervaring van communiceren in de familie kan verrijken, in plaats van vervangen.

Ook is het belangrijk dat deze technologie helpt om de negatieve effecten te verminderen die kunnen ontstaan door overmatig gebruik van digitale apparaten en sociale media.

## Oplossing

Vanuit de probleemstelling en het onderzoek dat ik heb uitgevoerd heb ik besloten om een app te ontwikkelen. Hoewel mijn achtergrond in webontwikkeling heb, zag ik meer waarde in het maken van een app, aangezien apps veel worden gebruikt voor communicatie. Mijn aanpak is om het probleem aan te pakken met een vergelijkbaar middel – een beetje zoals ‘vuur met vuur bestrijden’.

Het hoofddoel van mijn app is om gebruikers en hun familie in staat te stellen boeiende gesprekken aan te gaan en met elkaar in contact te blijven tijdens drukke tijden. Daarnaast wil ik het gebruik van mobiele apparaten helpen verminderen.

In de ontwikkelingsfase heb ik verschillende concepten overwogen. Eerst wilde ik een familietool-app maken met functies zoals kalenders, agenda's en chatmogelijkheden.

Na een grondige evaluatie besefte ik dat dit geen betekenisvolle interacties bevorderde, niet origineel was en niet hielp om de negatieve effecten van overmatig telefoongebruik te beperken.

## **flashback**

Dit bracht me bij mijn uiteindelijke concept: een foto-sharing social media app speciaal voor families. Foto's hebben de kracht om duizenden woorden, gebeurtenissen en emoties over te brengen met één enkel beeld, en ze stimuleren vaak gesprekken.

Door trends te onderzoeken, merkte ik dat tijdelijke foto's zoals Stories en Snaps populair zijn. Bijgevolg, heb ik mijn finale concept vormgegeven rond dit idee: een social media app waarin familieleden één keer per dag foto's kunnen plaatsen op een vastgesteld tijdstip.

De foto's worden pas later op de dag onthuld, wanneer iedereen tijd heeft om samen te komen. Dit creëert een gevoel van anticipatie gedurende de dag en bevordert kwalitatieve conversaties.

Door het beperken van de mogelijkheid om slechts één keer per dag te posten, pak ik ook het probleem van overmatig telefoongebruik aan.

# Analyse

## Concurrentieanalyse

Er zijn veel sociale media apps zoals Instagram, Snapchat en BeReal die op het delen van foto's en video's focussen. Ze zijn echt niet specifiek ontworpen om familiecommunicatie te bevorderen.

Instagram, bekend om zijn visuele content, stelt gebruikers in staat om hun levensmomenten te delen door middel van foto's en video's. Het dient ook als een krachtig platform voor merken om hun producten te promoten, wat helaas soms de sociale connectie tussen gebruikers kan verstoren. Ondanks deze kenmerken biedt Instagram geen functies die specifiek ontworpen zijn om familiecommunicatie te vergemakkelijken.

Snapchat, geliefd bij de jongere generatie, stimuleert spontane en authentieke communicatie door het gebruik van tijdelijke content. Net als Instagram, houdt het platform echter geen rekening met de nuances van familie-interacties en communicatie.

BeReal onderscheidt zich doordat het geen filters of advertenties toestaat en gebruikers beperkt tot het plaatsen van slechts één foto per dag. Deze beperkingen zijn bedoeld om oprechte interacties aan te moedigen, maar ook BeReal richt zich niet specifiek op familiecommunicatie.

Naast deze platforms zijn er ook apps die specifiek op families

zijn gericht. Echter, deze apps zijn vaak meer pragmatisch en bedoeld om het dagelijks leven te organiseren, zoals het delen van kalenders en takenlijsten, in plaats van het dienen van de sociale en communicatieve behoeften van families.

Kortom, veel van de concurrenten die ik heb onderzocht richten zich op een algemeen publiek. Ze doen dit succesvol door een breed assortiment aan features, functies en tools aan te bieden. Dit bevestigt mijn eerdere constatering dat er geen apps specifiek gericht zijn op families of het verbeteren van familiecommunicatie.

Dit toont aan dat, hoewel verbeterde communicatie tussen familieleden een waardevolle aanvulling zou kunnen zijn op bestaande apps, dit geen prioriteit is voor deze concurrenten. Er is dus een duidelijke behoefte aan een platform dat zich specifiek richt op het verbeteren van familiecommunicatie en het versterken van familiebanden.

## Doelgroepanalyse

Mijn doelgroep omvat families, met speciale aandacht voor jonge ouders en tieners. Ik kies voor deze groepen omdat deze algemeen gezien handiger zijn met technologie en vaak op zoek zijn naar nieuwe manieren om te communiceren.

Na het bekijken van andere sociale media apps, zie ik dat de gebruikers vaak jonger zijn. Ze willen graag op een leuke en andere manier contact hebben met vrienden. Oudere mensen gebruiken meestal simpele chats of praten gewoon in persoon.

Hieruit volgt mijn keuze om een app te maken voor mensen tussen de 12/13 en 45 jaar. Deze groep is breed genoeg om de gemiddelde familie te omvatten, en deze leeftijdsgroep is meestal ook open voor het proberen van nieuwe dingen.

Het doel is om een makkelijk te gebruiken app te maken die leuk is voor de jongeren, maar ook handig voor de hele familie.

# Functionele Analyse

Ik voerde mijn functionele analyse uit door het vragen aan vrienden en collega's of ze voornamelijk via sms of Whatsapp communiceren. Toen ik daarnaast informeerde over de manieren waarop ze met elkaar verbonden zijn. kreeg ik vooral het antwoord foto's.

Door foto's kunnen ze in een ogenblik met elkaar meelevens en erover praten, wat de kwaliteit van de conversaties kan verbeteren.

Er zijn ook foto's die alleen bedoeld zijn voor familieleden, maar toch in de galerij te zien zijn. Dit betekent dat mensen die niet tot je familie behoren, toch in staat zijn om deze foto's te zien als ze door je galerij scrollen.

Met mijn must-have wil ik een functie toepassen waarbij elk familielid zich eerst moet aanmelden. Zo krijgt iedereen een rol binnen de familie, zoals

vader, moeder, zus, enzovoort.

Door het aanmelden krijgt ieder familielid dus een rol. Gedurende de dag moet elk familielid een foto maken van zichzelf, die we de "flashback" noemen.

Zo moet ieder familielid dus een "flashback" maken, dat is een foto van hem of haar. Onder elke flashback moeten/kunnen de familieleden liken, wat elkaar stimuleert en goed laat voelen en later conversaties kan opleveren.

Er is een bepaalde tijdslimiet (bijvoorbeeld twee uur) om de flashbacks te plaatsen en een bepaalde tijd dat de flashback te zien zijn.

# MoSCoW-Analyse

Met deze analyse kon ik vaststellen welke functies absoluut essentieel zijn voor mijn app en welke niet. Deze inzichten hebben me in staat gesteld mijn prioriteiten te stellen tijdens het ontwikkelingsproces.

## **Must have**

Functies die ik moet hebben in mijn app

Rollenfunctie

Foto's zijn alleen zichtbaar voor familieleden

## **Should have**

Functies die ik zou moeten hebben in mijn app

User Login

Like button bij elke post

Foto post met time limit functie

Familie functie

Foto zichtbaar op een specifieke tijdstip

## **Could have**

Functies die ik zou kunnen hebben in mijn app

Locatie kunnen delen

Timelinefunctie

Profielpagina

E-mailverificatie na registratie

Forgot password functie

## **Won't have**

Functies die ik niet wil hebben in mijn app

Kalenderfunctie

Chatfunctie

Comments kunnen plaatsen

## Technische Analyse

Als frontend webontwikkelaar liggen mijn vaardigheden en ervaring voornamelijk in webontwikkeling. Daarom heb ik verschillende technologieën in overweging genomen voor de ontwikkeling van mijn applicatie, waaronder Flutter, Swift/Xcode en React Native.

Laten we eerst kijken naar Flutter. Flutter is een toolkit ontwikkeld door Google. Het is ontworpen om het voor ontwikkelaars gemakkelijker te maken om apps te bouwen die werken op mobiele, desktop- en webplatforms. Het handige aan Flutter is dat het open-source is, wat betekent dat iedereen het gratis kan gebruiken en verbeteren. Een nadeel van Flutter is echter dat het een relatief nieuwe taal is, dus er is nog niet zoveel ondersteuning of bestaande modules beschikbaar als voor sommige andere talen.

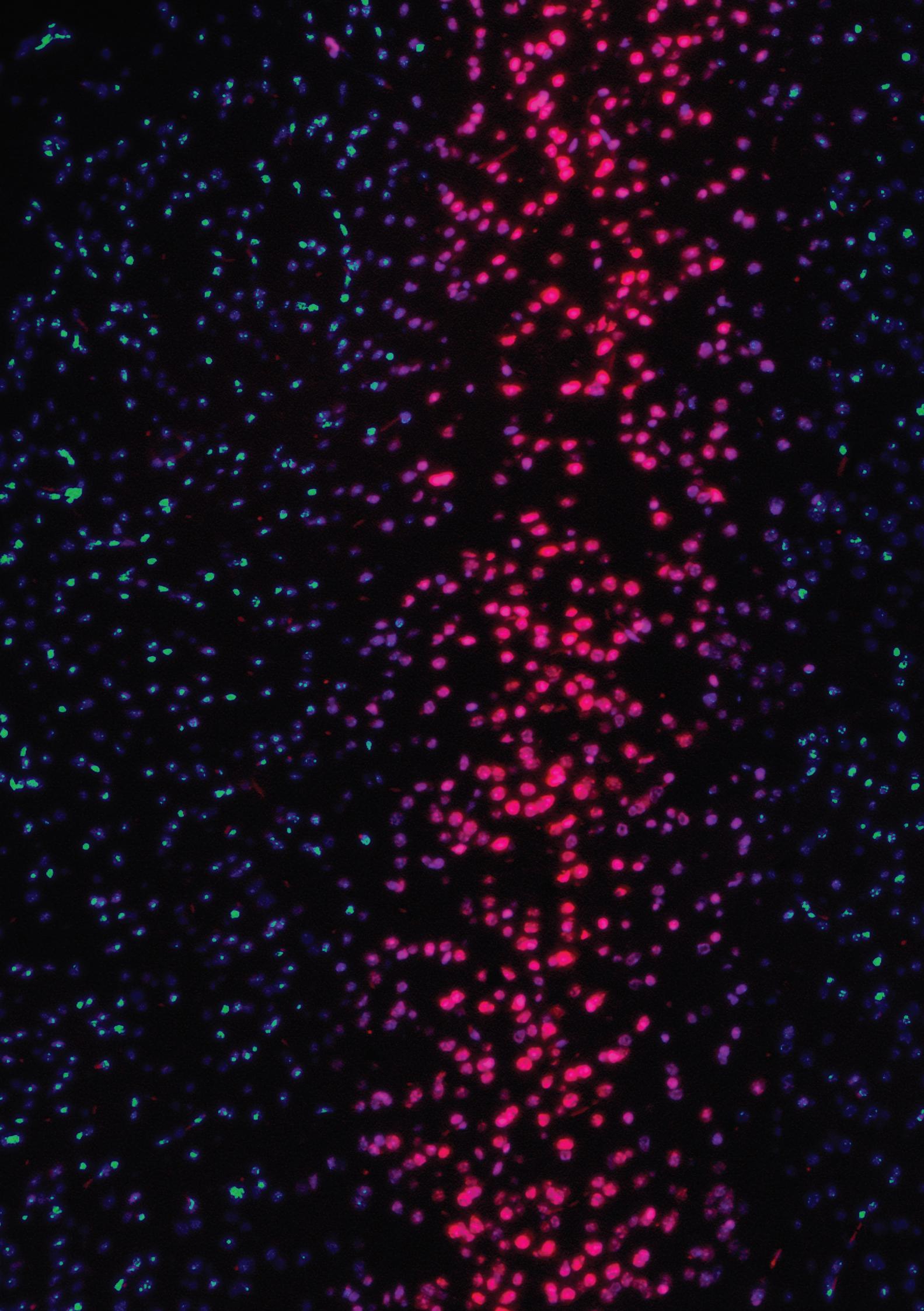
Swift is de programmeertaal

die door Apple is ontwikkeld voor het maken van iOS-apps. Je gebruikt het samen met Xcode, dat fungeert als een geïntegreerde ontwikkelomgeving (IDE) waarin je je apps bouwt. Het voordeel van Swift is de hoge uitvoeringssnelheid en krachtige prestaties, die het mogelijk maken om krachtige apps te bouwen. Het nadeel is dat het zich uitsluitend richt op Apple-apparaten, dus het ontwikkelen van apps voor Android is niet mogelijk met deze technologie.

React Native, de technologie die ik uiteindelijk heb gekozen, is een open-source framework ontwikkeld door Facebook. Het maakt het voor webontwikkelaars mogelijk om native mobiele apps te schrijven in JavaScript, een taal die ze al kennen, en tegelijkertijd de snelheid en efficiëntie van native apps te behouden. Het grootste voordeel van React

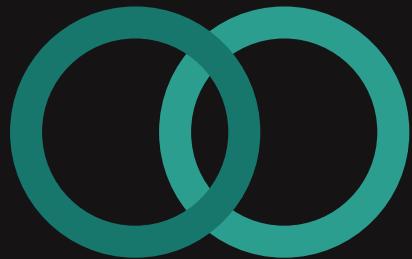
Native is de mogelijkheid om één keer te schrijven en overal uit te voeren - je schrijft één codebase die werkt op zowel Android als iOS.

Het gebruik van React Native leek voor mij een logische keuze, gezien mijn bestaande ervaring met ReactJS, waarmee de syntax veel overeenkomsten vertoont. Dit betekent dat ik, ondanks dat app-ontwikkeling nieuw voor mij is, niet helemaal vanaf nul hoeft te beginnen. De leercurve is minder steil en ik kan efficiënter werken. Bovendien stelt het gebruik van React Native mij in staat om een app te ontwikkelen die zowel op Android als iOS kan worden uitgevoerd. Bij andere technologieën is dit niet mogelijk, wat hun gebruik minder flexibel maakt.



# Huisstijl

## Colors



#24776F



#303030



#580020

#2A9D8F

#DCDCDC

#9D2A38

Ubuntu regular  
#DCDCDC

flash

## Fonts

Aa

Nunito Bold

ABCDEFGHIJKLMNO  
PQRSTUVWXYZ

abcdefghijklmno  
qrstuvwxyz

1234567890

Aa

Nunito Medium

ABCDEFGHIJKLMNO  
PQRSTUVWXYZ

abcdefghijklmno  
qrstuvwxyz

1234567890

Aa

Nunito Regular

ABCDEFGHIJKLMNO  
PQRSTUVWXYZ

abcdefghijklmno  
qrstuvwxyz

1234567890

## Logo



### Symbol logo

Dit logo bestaat uit handen die elkaar vasthouden, gevormd in een cirkel tegen de klok in. De handen symboliseren de familielieden die met elkaar willen verbinden, terwijl de tegengestelde richting van de klok een knipoog is naar het idee van terug in de tijd gaan, zoals de naam “flashback” suggereert.

# back

### Main logo

De naam ‘flashback’ verwijst naar het herbeleven van een scène uit het verleden. Dit concept is ook de kern van mijn app. Gebruikers maken een foto en bekijken deze pas later op de dag, waardoor ze in feite een ‘flashback’ ervaren.

# Design

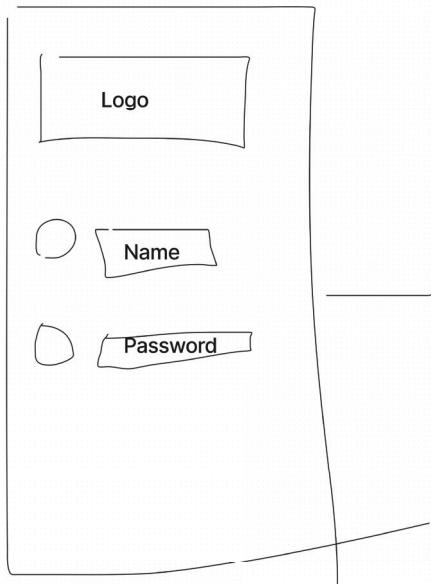


## Sketch

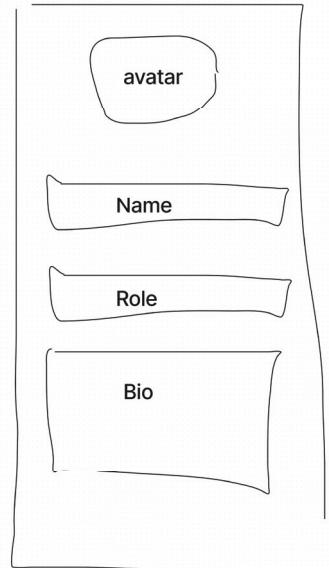
Voordat ik aan het ontwerp van mijn applicatie begon, moest ik eerst een idee krijgen van de gewenste structuur. Dit heb ik bereikt door een wireframe te creëren. Hoewel de meeste mensen dit proces zouden voltooien met een softwareprogramma, gaat mijn voorkeur naar de “old school” manier: ik heb de begin lay-out van mijn app manueel getekend met behulp van Figma. Uiteindelijk heeft mijn finale ontwerp deze lay-out overgenomen.



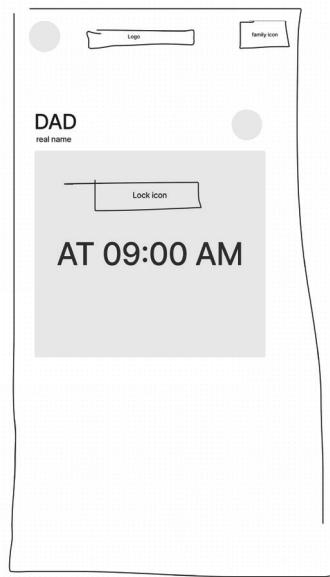
#### Log in / Sign up



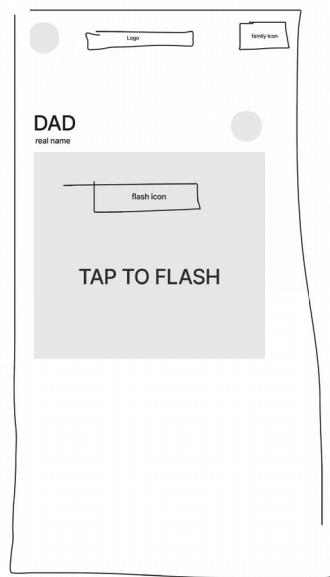
#### Profile



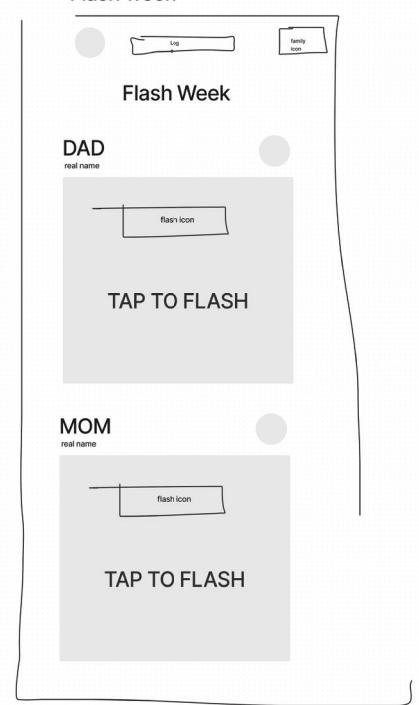
#### Home / Locked

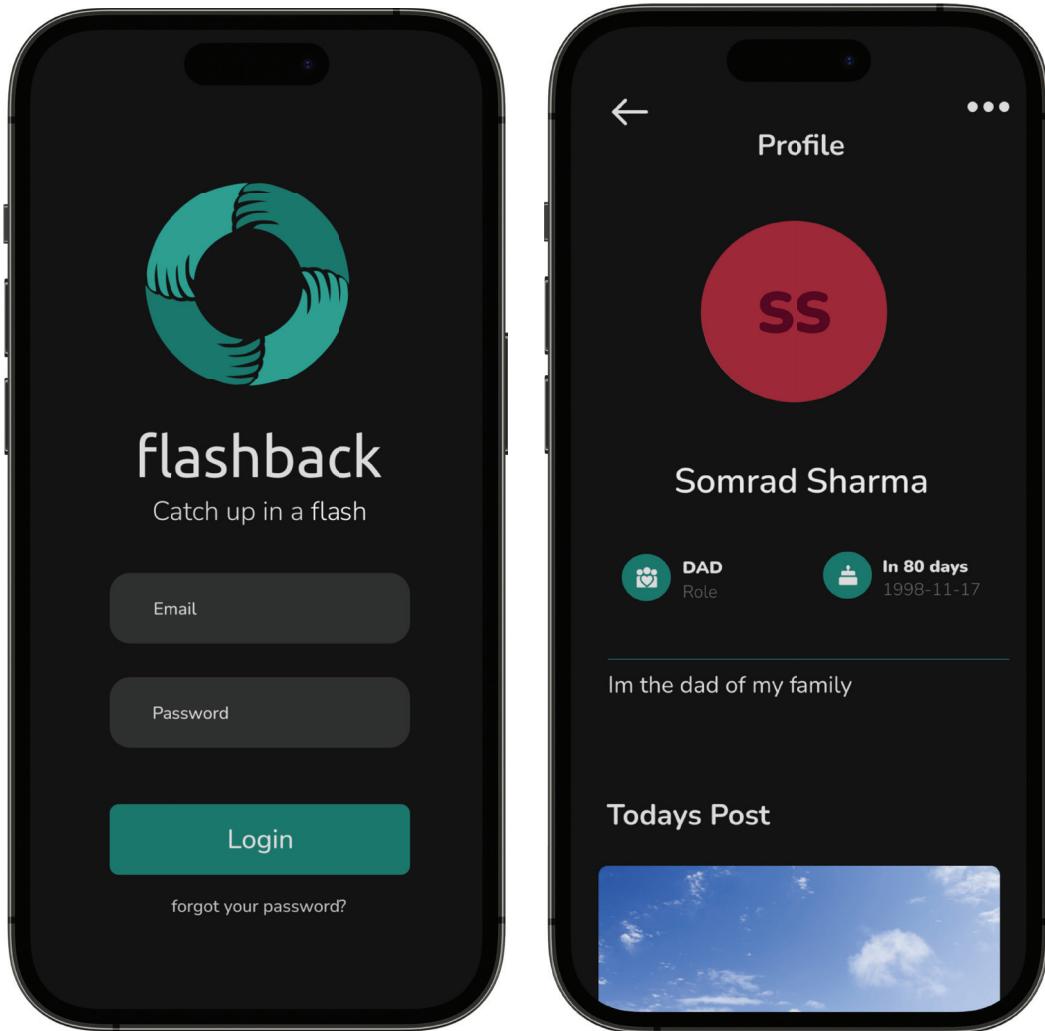


#### Home / UnLocked



#### Flash week

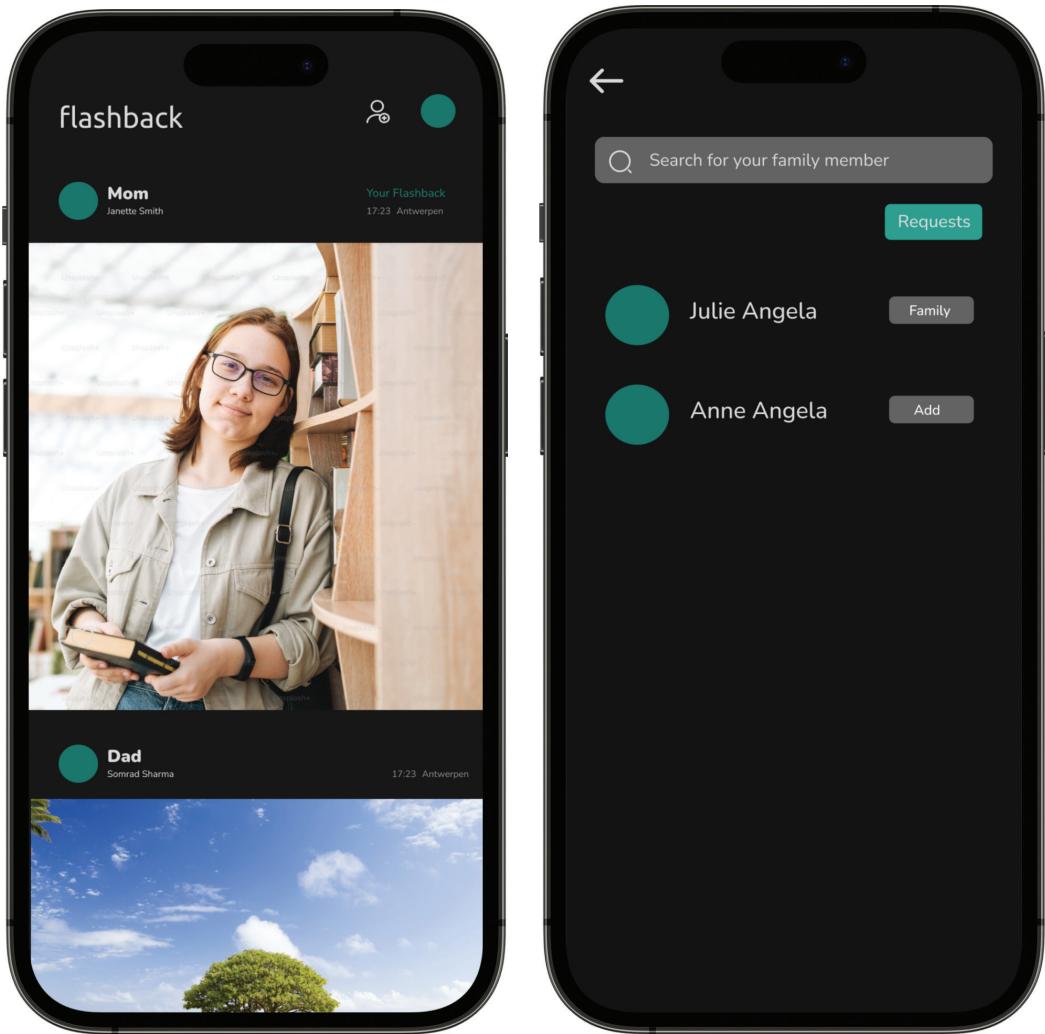




## Mockup

Na het trial en error bij het schetsen en creëren van mijn wireframes, bereikte ik het uiteindelijke product bereikt. Dit is het definitieve ontwerp dat ik uiteindelijk in mijn app gebruik. Tijdens het ontwikkelproces zijn er enkele veranderingen doorgevoerd om de ontwikkeling te vergemakkelijken, maar het ontwerp wijkt niet veel af van het oorspronkelijke concept.





# Development

## Tools

Voor mijn bachelorproef heb ik een aantal tools en packages gebruikt die essentieel waren voor mij. Zonder deze tools zou mijn applicatie “flashback” nooit tot stand zijn gekomen. In dit deel ga ik daar verder op in.

### React Native

Zoals eerder genoemd heb ik gekozen om mijn applicatie te ontwikkelen met het React Native framework. Een framework is een gestructureerde omgeving die helpt bij het coderen van software. React Native is bijzonder omdat het gebruik maakt van JavaScript en de syntax zeer vergelijkbaar is met die van ReactJS.

Vóór dit project had ik alleen een kleine workshop over

React Native gevolgd. Echter, in het eerste semester had ik uitgebreid gewerkt met ReactJS, waardoor ik een solide basis had opgebouwd in het werken met React syntax. Dankzij deze ervaring kon ik de kennis gemakkelijk toepassen in mijn werk met React Native.

Een ander voordeel van het kiezen voor React Native is de omvangrijke en actieve gemeenschap. Er zijn veel bibliotheken en uitgebreide

documentatie beschikbaar, wat het ontwikkelingsproces aanzienlijk vergemakkelijkt. Deze rijke bronnen waren een grote hulp bij het bouwen van mijn applicatie, waardoor het ontwikkelproces soepeler verliep.

## Visual Studio Code

Visual Studio Code is een veelgebruikt softwareprogramma voor het schrijven van code. Het is een zogenaamde IDE, ofwel Integrated Development Environment (Geïntegreerde Ontwikkelomgeving), een alles-in-één tool voor programmeurs waarin je code kan schrijven, testen en debuggen.

Deze tool is geliefd onder ontwikkelaars waaronder ikzelf, vanwege zijn moderne, gebruiksvriendelijke ontwerp. Een van de grootste voordelen is de uitgebreide bibliotheek met plugins waarmee je Visual Studio Code kan personaliseren.

## Xcode

Xcode is speciaal bedoeld voor het ontwikkelen van iOS-apps die alleen in de App Store kunnen worden gepubliceerd. Ik heb Xcode echter op een iets andere manier gebruikt. In plaats van het te gebruiken om apps te bouwen, heb ik het gebruikt om toegang te krijgen tot de iPhone-simulator. Dit is eigenlijk een virtuele telefoon die ik op mijn computer kon laten verschijnen. Hierdoor kon ik mijn applicatie testen alsof het op een iPhone draaide, maar dan gewoon op mijn computer.

## Expo

Expo is een open-source platform dat je kan inzetten voor het ontwikkeling en het testen van applicaties. Bij het opstarten van een React Native project kan je ervoor kiezen om Expo direct in je project te integreren. Voor mij bleek Expo bijzonder nuttig, zelfs onmisbaar, tijdens de ontwikkelingsfase van mijn app. Het zorgde voor een soepel en efficiënt testproces, waardoor ik me volledig kon richten op het verbeteren van de functionaliteiten van de app.

## Trello

Trello is een webgebaseerde tool waarmee je gedetailleerde planningen kan maken en deze volledig kan aanpassen naar je eigen wensen. Bovendien biedt het de mogelijkheid om je planning te delen met je team, mocht dit nodig zijn.

Ik heb Trello uitgebreid gebruikt tijdens de ontwikkeling van mijn applicatie. Gezien het feit dat de ontwikkeling van mijn applicatie een groot project met een strakke deadline was, was er veel te beheren en te organiseren. Dankzij Trello kon ik mijn taken efficiënt opsplitsen in kleinere, beheersbare stukjes en elk van deze taken koppelen aan een specifieke deadline.

## Firebase Real Time DB

Firebase, een Google-product, is een veelzijdig platform dat meerdere diensten aanbiedt. Een daarvan is Real Time Database, die ik in mijn app gebruik om gebruikersgegevens op te slaan. Het grootste voordeel van deze database is de realtime werking; veranderingen in de gebruikersgegevens worden onmiddellijk bijgewerkt in de database. Dit zorgt voor een vlotte en reactieve ervaring voor de gebruikers, wat essentieel is voor het succes van mijn app.

## Firebase Authentication

Een andere dienst van Firebase die ik gebruik, is Firebase Authentication. Hiermee kunnen gebruikers zich registreren, inloggen en hun wachtwoord herstellen, allemaal met hun e-mailadres en wachtwoord. Deze functie houdt de gebruikersgegevens veilig en makkelijk te beheren.

## Firebase Storage

Een andere belangrijke functie is Firebase Storage, waar ik de foto's die gebruikers in mijn app posten opslaan. Elke opgeslagen afbeelding krijgt een unieke URL, die ik in mijn Firebase Realtime Database bewaar, gekoppeld aan de juiste gebruiker. Dit maakt het bijhouden en terugvinden van de geüploade foto's zeer efficiënt.



## Packages

Dit onderdeel bespreekt de softwarebibliotheeken (ook wel ‘packages’ genoemd) die ik heb gebruikt voor mijn app. Door deze packages te gebruiken, kon ik bepaalde functionaliteiten implementeren zonder ze vanaf nul te moeten ontwikkelen.

```
expo/vector-icons - Versie 13.0.0
react-native-community/datetimepicker - Versie
7.1.0
react-navigation/native - Versie 6.1.6
react-navigation/stack - Versie 6.3.16
expo - Versie 48.0.15
expo-camera - Versie 13.2.1
expo-font - Versie 11.1.1
expo-image-manipulator - Versie 11.1.1
expo-image-picker - Versie 14.1.1
expo-status-bar - Versie 1.4.4
firebase - Versie 9.22.0
from - Versie 0.1.7
import - Versie 0.0.6
react - Versie 18.2.0
react-firebase-hooks - Versie 5.1.1
react-native - Versie 0.71.7
react-native-picker-select - Versie 8.0.4
react-native-svg-transformer - Versie 1.0.0
react-native-vector-icons - Versie 9.2.0
tinycolor2 - Versie 1.6.0
expo-blur - Versie 12.2.2
```

# Moeilijkheden & Oplossingen

Elke uitdaging in een project is een kans voor ons als mensen om te leren en te groeien. Dit project, mijn eerste ervaring met React Native en app-ontwikkeling, was hierop geen uitzondering. Hoewel de overgang van webdevelopment naar app-ontwikkeling in het begin wat lastig was, hielp elke hindernis me om te evolueren. Het proces ging uiteindelijk vlotter dan verwacht.

Met de tijd en door vallen en opstaan, wist ik React Native volledig te doorgronden en te beheersen.

## React Native Svg

Het importeren van een SVG lijkt een eenvoudige taak, maar voor mij was dit helaas niet het geval. Voor iets dat zo simpel zou moeten zijn, heb ik er relatief veel tijd aan besteed. Het probleem was dat React Native standaard geen ondersteuning biedt voor SVG's. Na verschillende pakketten te hebben uitgeprobeerd, vond ik uiteindelijk "react-native-svg-transformer" waarmee ik succesvol mijn SVG's kon importeren. SVG's zijn cruciaal voor het UI/UX-design van mijn app, dus ik bleef doorzetten tot het werkte.

## Firebase Syntax

Mijn eerste ervaring met Firebase verliep grotendeels soepel, met als uitzondering van het implementeren van het optiescherm. Hier konden gebruikers hun informatie bijwerken. Data ophalen lukte, maar deze aanpassen en terugsturen naar de database was een uitdaging. Ik gebruikte een verouderde Firebase syntax (versie 8), die het bewerken van gegevens belemmerde. Door alle Firebase code in mijn project bij te werken naar de nieuwere versie 9 syntax, kon ik deze hindernis overwinnen en het optiescherm succesvol voltooien.

## App Crash

Bij het ontwikkelen van de foto-uploadfunctie liep ik tegen een probleem aan - de app crashte voortdurend. Met geen specifieke foutmeldingen was het moeilijk om het probleem te vinden. Na een uitgebreide controle van de code en functietests, kwam ik erachter dat de foto's te geheugenintensief waren, wat tot crashes leidde.

Ik loste dit op door een compressiemethode in te voeren die de grootte van de foto's verminderde, waardoor de geheugenbelasting afnam en de crashes stopten.

## Stap 1

Ga naar mijn GitHub-pagina: <https://github.com/somrads/flashback.git> en download de code van het project.

## Stap 2

Open de gedownloade code in Visual Studio Code en navigeer naar de frontend-map. Voer het commando ‘npm install’ uit in de terminal om alle benodigde packages te downloaden en te installeren.

## Stap 3

Ga naar de db-map en open het bestand ‘firebase.js’. Hier moet je jouw eigen Firebase-credentials invoeren als je deze nog niet hebt. Indien je met mijn credentials wilt werken, kan je deze stap overslaan.

## Stap 4

Je krijgt een QR-code te zien in de terminal. Om de app op je telefoon te krijgen, download je de Expo-app en scan je de QR-code met de camera van je telefoon.

Als je de app op een emulator wilt uitvoeren, druk je op ‘Shift + i’ om de emulators te openen. Kies de iOS-emulator en selecteer het gewenste iPhone-model.

## Belangrijk

In de camera- en postcomponenten kan je de timingfuncties naar wens aanpassen, zodat je niet hoeft te wachten op het exacte tijdstip om een foto te maken of om de foto te bekijken.

Houd er rekening mee dat voor het gebruik van de iOS-

emulator en het testen van de app op iOS-apparaten, Xcode moet worden gedownload en geïnstalleerd. Houd er ook rekening mee dat de app momenteel alleen compatibel is met iOS-toestellen.

# Technisch Rapport

Op dit moment is de app alleen lokaal beschikbaar en nog niet online gepubliceerd. Zodra de app beschikbaar is in de Expo App Store, kan je deze downloaden door te zoeken naar “flashback”. Zorg ervoor dat je de Expo-app hebt gedownload voordat je de app gebruiken.

Om de app te gebruiken,

meld je je aan en verifieer je je e-mail. Zodra je bent ingelogd, kan je foto's maken op het aangegeven tijdstip in de app. Wanneer je een foto plaatst, ontvangen jij en je familieleden een melding met de tekst “flashback created” in je feed. Je kan ook je profiel bekijken en je informatie en profielfoto aanpassen. Verder kan je familieleden toevoegen

om hun foto's te zien. Houd de Expo App Store in de gaten voor de officiële release van “flashback” en geniet van het gebruik van de app zodra deze beschikbaar is.

# Besluit

Het ontwikkelen van een applicatie is een complex en tijdrovend proces. Het is onrealistisch om binnen een korte periode een perfect functionele app te creëren. Ondanks deze uitdagingen ben ik tevreden met de resultaten die ik heb behaald, vooral gezien mijn beperkte kennis van app-ontwikkeling in het begin.

Tijdens dit project heb ik veel geleerd over het werken met Firebase, zoals het implementeren van een loginpagina met behulp van Firebase en het werken met databases om gegevens op te halen en te verzenden tussen verschillende delen van de app.

Mijn kennis van React Native is ook aanzienlijk gegroeid

en het is nu een belangrijke vaardigheid geworden. Daarnaast heb ik veel geleerd over app-ontwerp, aangezien ik voornamelijk een webdesigner was. Over het algemeen heb ik enorm veel geleerd en dingen bereikt die ik eerder niet voor mogelijk had gehouden.

Ik ben vooral trots op hoe mijn onderzoek en mijn product mijn onderzoeksraag effectief hebben beantwoord. Met de app heb ik de mogelijkheid gecreëerd voor kwalitatieve interacties binnen families door middel van foto's die alleen op een geschikt moment voor iedereen zichtbaar zijn. Bovendien ben ik ervan overtuigd dat de beperking van het eenmaal per dag plaatsen van foto's bijdraagt aan de verminderen smartphonegebruik.

Hoewel er nog enkele functies ontbreken en ik graag zou willen toevoegen - iets wat ik in de toekomst ook zeker zal doen - beschouw ik dit project als een succes. Ik ben trots op wat ik heb bereikt en heb een degelijk product kunnen leveren.

# Literatuurlijst

Anxiety Centre. (2021, 15 juni). Screen Time Linked To Anxiety And Depression In Children - AnxietyCentre.com. <https://www.anxietycentre.com/research/screen-time-linked-to-anxiety-and-depression-in-children/>

Bug Ninja. (2022a, 23 april). How To Add Custom Google Fonts In React Native Expo Applications | React Native Tutorial | JS [Video]. YouTube. <https://www.youtube.com/watch?v=GeySpTa0EXE>

Bug Ninja. (2022b, 19 juli). Email Authentication With Verification In React Native and Firebase App | React Native Tutorial | [Video]. YouTube. <https://www.youtube.com/watch?v=lBUjggiWQ1U>

Bug Ninja. (2023, 9 januari). How To Fetch or Retrieve Data From Firebase No SQL Real-time Database Using React Native & Expo Apps [Video]. YouTube. <https://www.youtube.com/watch?v=6wk1OrGKE1w>

Camera. (z.d.). Expo Documentation. <https://docs.expo.dev/versions/latest/sdk/camera/>

Garcia, J. (2021). What Are Social Media Stories and Why Are They Everywhere? MUO. <https://www.makeuseof.com/social-media-stories-everywhere/>

Gledhill, N. (2020). Social media affects family dynamics. The Daily Universe. <https://universe.byu.edu/2020/01/21/the-effects-of-social-media-on-family-dynamics/>

MissCoding. (2022, 8 oktober). Using Custom Fonts in Expo React Native Apps with Splash Screen Loading - useFonts Hook [Video]. YouTube. <https://www.youtube.com/watch?v=vilkcDYSBrl>

npm: react-native-camera. (z.d.). npm. <https://www.npmjs.com/package/react-native-camera>

React Native · Learn once, write anywhere. (z.d.). <https://reactnative.dev/>

Safes. (2023). Effect of social media on parents' parenting. <https://www.safes.so/blogs/effect-of-social-media-on-parents-parenting/>

Timestamp | React Native Firebase. (z.d.). <https://rnfirebase.io/reference/firestore/timestamp>

William Candillon. (2020, 22 september). #20 Edit Profile — React Native Fashion [Video]. YouTube. <https://www.youtube.com/watch?v=2hveuwFtVZ0>

flashback

flashback

## **Bachelorproef 3BaMCT**

Auteur: Somrad Sharma

Studentnummer: 201888178

E-mailadres: somrad.sharma@student.ehb.be

Opleiding: Multimedia & Creative Technologie

Organisatie: Erasmushogeschool Brussel

Promotor: Bert Heyman

19 juni 2023, Anderlecht, Brussel





Catch up in a  
flash