

'Stay connected'

Hidde Hensel, Freddy de Greef, Diederik Beker

Idee:

Het idee is vrij simpel, een geschikte match vinden tussen twee personen. Natuurlijk bestaat dit al enige tijd in de vorm van dating sites en apps. Bij een dating site vult men een profiel in en wordt aan de hand van dit zelf ingevulde profiel gekoppeld aan iemand met vergelijkbare interesses. Soms pakt dit goed uit, soms minder goed. De vraag blijft altijd hangen of mensen dit profiel wel eerlijk invullen. Het is vrij gemakkelijk om kwaliteiten die je bezit ietsje te overdrijven of jezelf net iets interessanter te laten klinken dan je eigenlijk bent. Dus hoe betrouwbaar is dit 'dating profiel' nou echt?

Bij deze app is het de bedoeling te kijken naar de echte situatie, niet een schets die iemand van zichzelf maakt. Het idee is vrij simpel, je zet de applicatie aan op je telefoon en dat is alles wat een gebruiker hoeft te doen. Op de achtergrond houdt de app meta informatie over de persoon in kwestie. Er wordt bijgehouden waar de persoon is en naartoe gaat, hoelang die op deze locatie is, er wordt door middel van geofencing bepaald wat er op die locatie gedaan wordt. Bovendien wordt er door middel van peer-to-peer communicatie tussen de verschillende, in de buurt van elkaar te bevinden, telefoons bijgehouden met wie deze activiteit wordt uitgevoerd. Het belpatroon en app gebruik kan worden bijgehouden en opgeslagen.

Neem al deze informatie bij elkaar over een bepaalde tijd en er ontstaat een vrij duidelijk patroon van wat een persoon doet, welke interesses hij/zij heeft en waar en wanneer deze persoon zich ergens bevindt. Door middel van vergelijking van dit patroon met het patroon van andere personen kan een match worden gevonden op basis van feitelijke informatie i.p.v. een zelf ingevuld profiel, en kan een grote 'match succes rate' worden behaald.

Als een match is gevonden kan er door middel van eerder verkregen informatie een plaats en tijd worden bepaald (voorspeld) waar deze personen kunnen ontmoeten. Zonder verdere informatie krijgen zij de tijd en plaats van de ontmoeting. De eerste ontmoeting zal dus de eerste keer zijn dat deze personen informatie over elkaar krijgen. Als het goed is hebben ze veel te bespreken aangezien ze dezelfde interesses, gedrags patronen, koffiedrink tijden, locaties, etc. delen.

Er wordt bij deze verschillende data opgeslagen. Deze data bestaat onder andere uit: locaties, dit wordt opgehaald m.b.v. Google Location Services, activiteiten, bel patronen, app gebruik, contact met andere personen. Dit wordt centraal bijgehouden in een database. Door middel van bluetooth kan er gecommuniceerd worden met andere telefoons en door middel van wifi en mobiel netwerk kunnen er gegevens worden gestuurd van en naar de server. Op de server wordt een patroon gemaakt van de data en deze metadata wordt het vergeleken met het patroon van andere mensen.

Het systeem kan worden versneld, voor de gebruiker, door rekening te houden met de waarschijnlijkheid van een ontmoeting tussen mensen. Als er een patroon wordt gevonden waarin een bepaald persoon vaak een ander bepaald persoon tegenkomt, kan de ene persoon data van de server meenemen voor de andere persoon. Op deze manier blijft iedereen (via-via) up-to-date, en zijn de updates sneller. Met genoeg mensen die gebruik maken van de applicatie zijn deze updates zelfs bijna real-time en werkt het kosten besparend, doordat minder mensen met het netwerk verbinding hoeven te maken, en meer communicatie over het gratis bluetooth plaatsvindt.

Het uitgangspunt is dus een applicatie die, op de achtergrond, informatie verzamelt over het gedrag van een persoon en deze, door middel van patroon herkenning, koppelt aan een ander persoon. Er wordt een ontmoeting gepland door middel van een voorspelling van tijd en plaats. Het enige wat de gebruiker hoeft te doen om een match te vinden is zijn of haar telefoon bij zich houden (wat toch

al gebeurt). Op deze manier is het vinden van een match kosteloos, makkelijk en kost het geen tijd. Bovendien kan er niet worden gelogen of overdreven of interesses en persoonlijkheid en worden er dus reële matches gevonden.

Sensoren:

Camera
Micorfoon
Wifi
Bluetooth
Compass
Gravitatie meter
GPS

Wanneer connectie:Bij elkaar in de buurt:

Connectie met bluetooth of wifi. Verstuur meest recente database naar andere gebruiker.

Matching gebeurt op server.

Bij een ontmoeting, afstand meting (door bluetooth?), daardoor weet je dat je de juiste persoon hebt.

Is er een ontmoeting plaatsgevonden na een match? → dus werkt het matching patroon?

Wifi → database:

Stuur meta data.

Haal meest recente database op.

Systeem weet dat mensen vaak in de buurt zijn van elkaar, dus kan gegevens van de database meesturen met iemand anders en dan peer-to-peer overdragen. (Als een persoon niet in de buurt is van wifi maar wel van een ander persoon, kan die nog steeds server data ontvangen door peer-to-peer).

Gebruikers gegevens:

Meta data → GPS, Bel patroon, connecties met andere personen, app gebruik

Door het kijken naar meta data wordt er alleen rekening gehouden met gedrag, niet user input. AI mogelijkheden.

Gps:

Waar ben je geweest? Uni, sportschool, etc. (Patroon)

Punten herkenning

Hoelang ben je ergens geweest?

Bel patroon:

Hoelang?

Wanneer?

Waar?

Connecties:

Met wie ben je ergens?

Hoelang ben je met iemand?

Met wie maak je connectie?

Wat wissel je uit?

App gebruik:

Welke app gebruikt iemand?
Hoelang gebruikt iemand deze app?

Match gevonden:

Indien mensen gematched kunnen worden, kan er worden bepaald (door middel van algoritme), wanneer deze twee mensen tijd hebben en op welke locatie het het handigst is om een ontmoeting te laten plaatsvinden, en wordt deze dus geplanned.

Waarom is dit anders dan een dating site:

Mensen kunnen geen nep profiel creëren of dingen overdrijven omdat ze niet zelf informatie kunnen invoeren, hierdoor krijg je real matching met mensen met dezelfde interesses. Natuurlijk is het altijd mogelijk dat mensen een ander gedrags patroon gaan vertonen, maar dit kost veel moeite, en iemand moet zijn gewone dagelijkse taken alsnog uitvoeren.

Deze app is dus niet als een online dating profiel, er wordt niet gekeken naar wat mensen zeggen dat ze doen, maar wat ze echt doen.

User specificatie:

Kan specificeren wat voor meeting die wil: Vriendschappelijk, blind-date, etc.

Data:

De manier waarop data wordt opgeslagen en verzonden is natuurlijk erg belangrijk. Ten eerste moet er genoeg data worden opgeslagen om een reeel en logisch patroon te kunnen creëren. Ten tweede moet de data makkelijk uit te lezen zijn op zowel de server als het android toestel. Ten derde moet het niet te veel ruimte innemen. En als laatste moet het makkelijk kunnen worden verzonden tussen de server en het android toestel, en tussen twee android toestellen. Een eventuele toevoeging kan zijn: encryptie. De data kan gevoelige informatie bevatten, zoals locatie data, die voor andere personen moet worden afgeschermd.

Het lijkt ons het handigst om de data in een *string* te verzenden. Een string is makkelijk op te slaan in de SQLite database van een android toestel, het is makkelijk te verzenden via bluetooth en naar en van de server. We kunnen een string zo indelen dat je in één string alle data kan verpakken en verzenden door middel van scheidings tekens.

Voorbeeld:

We willen graag de volgende data verzenden:

Location: Long:53.2352349, Lat:4.23482092

Address: Sciencepark 500, Amsterdam, The Netherlands

App usage: phone:54min20secs, WhatsApp:21mins08secs, SnapChat:5mins12secs

We kunnen deze data in één string zetten met het scheidingsteken ';'

"Location: Long:53.2352349, Lat:4.23482092;Address: Sciencepark 500, Amsterdam, The Netherlands;App usage: phone:54min20secs, WhatsApp:21mins08secs, SnapChat:5mins12secs"

Deze string kunnen we vervolgens op een ander toestel of op de server weer logisch opdelen in de individuele strings 'Location', 'Address' en 'App usage'.

Make the android application a service

For this application to work properly it obviously needs to continuously gather information. For this to be possible the application needs to always be running. If you keep the application running on the I/O-thread you would not be able to use your phone for other things, that is obviously not going to work. For this reason our app will be implemented as a *service*. A service is an application

running in the background on a different thread than the I/O-thread. This way it can continuously gather data without the user noticing the application being run. A problem could arise with a phone that is being restarted, the application has to be started manually for it to start getting the information again. For this we implemented a *BroadcastReceiver* that listens for *BOOT_COMPLETE*. Due to android being able to stop the application if it lacks memory we will have the *onStartCommand* return *START_STICKY* and our *onCreate* will call *startService* on itself, the application will restart as soon as enough memory is available.

Stappenplan:

1. Server maken
2. Connectie van android naar server (via wifi)
3. Connectie van server naar android (via wifi)
4. Connectie van android naar android (via bluetooth)
5. Data verzending van android naar server
6. Data verzending van server naar android
7. Data verzending van android naar android
8. App maken die locatie (via wifi) kan opslaan (google api voor locatie)
9. Meta data verzameling (google api voor locatie)
10. Database voor meta data
11. Meta data vergelijking voor match (algoritme)
12. Planning van ontmoeting op basis van gegevens (voorspelling van locatie en tijd)
13. GUI voor android

Extra features:

- Mensen die vaak bij elkaar in de buurt zijn kunnen data, van de server, voor elkaar meenemen en overdragen.
- Data verzending secure maken.
- User opties voor soort match (vriendschappelijk, blind-date, etc.).
- Bij ontmoeting, afstands meting tussen de twee mensen, hierdoor wordt bepaald of de ontmoeting werkelijk heeft plaats gevonden.
- Database update door middel van peer-to-peer. De meest recente database wordt naar andere gebruiker gestuurd door middel van peer-to-peer. Dus als iemand niet in de buurt is geweest van wifi kan die toch een geupdate database hebben.

Scenario:

Persoon fietst elke dag van huis (wifi connectie kwijt) naar het UvA science park (wifi connectie). De tijdsduur van deze reis wordt bijgehouden. Onderweg komt deze persoon langs de UvA faculteit op Roeterseiland (GPS of wifi connectie?) (Onderweg peer-to-peer?). De persoon blijft een aantal uur op het science park en vertrekt dan (wifi connectie kwijt) naar een cafe (wifi connectie). Hier spreekt die af met iemand anders waardoor een peer-to-peer connectie ontstaat. De tijdsduur van de ontmoeting en de persoon waarmee wordt ontmoet wordt onthouden. Hierna vertrekt de persoon en gaat naar de hockeyclub. De tijdsduur van de wedstrijd wordt bijgehouden en eventuele medespelers met peer-to-peer. Een telefoontje wordt gepleegd en er wordt bijgehouden hoelang dit duurt en op welke locatie. Door de dag heen worden een aantal apps gebruikt en de tijdsduur en soort app wordt bewaard. Thuis aangekomen krijgt de persoon een match met een hockeyer die in de buurt van het zojuist bezochte cafe ook vaak koffie drinkt. Er wordt daar in de buurt een ontmoeting geplanned. Ze leefde nog lang en gelukkig.

Visualisatie:

Visueel beeld van connecties, informatie verzending, etc.

Planning:

14 Juni	Bluetooth connectie Server en android connectie Google location api en lokale database Schrijven van verslag en contact met Technology Transfer Office	Freddy Hidde Diederik Diederik
19 Juni	Data logisch verpakken en verzenden Andere meta data verzameling Data vergelijking (algoritme)	Diederik
21 Juni	Onmoeting plannen (voorspellen van tijd, plaats) Database update via bluetooth (meest recente) Ontmoetings check (bluetooth afstand)	
26 Juni	Data voor server via andere persoon naar database Data beveiliging GUI opmaken, user opties	

To do:

- Benchmark bedenken

Dagelijke notulen:

Maandag 10-Jun:

- Bespreking project → wat gaan we doen?
- Conclusie: Stay Connected, Diederik verantwoordelijk voor git
- Taakverdeling (zie Planning hierboven)
- Eind van de dag → wie heeft wat afgekregen?
- Conclusie: Iets meer tijd nodig voor alles, verslag voor groot deel opgesteld

Dinsdag 11-Jun:

- Hidde is ziek
- Wat gaan we doen vandaag?
- Conclusie: Verder volgens planning
- Hidde werkt wanneer die kan, haalt in het weekend in wanneer beter
- Eind van de dag → wie heeft wat afgekregen?
- Conclusie: Location gevonden, morgen verder met database

Woensdag 12-Jun:

- Hidde is ziek
- Wat gaan we doen vandaag?
- Conclusie: Verder met taken
- Eind van de dag → wie heeft wat afgekregen?
- Conclusie: Bluetooth is stuk lastiger dan gedacht, database opslaan gelukt, maar morgen maken met Service zodat die concurrent kan, brief naar Technology Transfer Office gestuurd

Vrijdag 14-Jun:

- Hidde is nog steeds ziek
- Wat gaan we doen vandaag?
- Conclusie: Bluetooth afmaken, duurt nog wel even, anders in het weekend afmaken. Database schrijven en ophalen concurrent maken d.m.v. Services
- Eind van de dag → wie heeft wat afgekregen?
- Conclusie: Verslag geschreven en concurrency van database schrijven en lezen afgemaakt. Bluetooth is nog steeds lastig, maar voortgang, dit weekend afmaken.

Maandag 17-Jun:

- Hidde weer beter!
- Wat gaan we doen vandaag?
- Conclusie: Hidde gaat hard aan het werk met de server-android connectie, Diederik gaat werken aan data verzending (Wat? Hoe?), Freddy gaat weer verder met bluetooth want dat was nog niet gelukt.