

# Indholdsfortegnelse

---

<b>1</b>	<b>App vision</b>	<b>2</b>
<b>2</b>	<b>Personlig vision</b>	<b>3</b>
<b>3</b>	<b>Context</b>	<b>4</b>
<b>4</b>	<b>Krav</b>	<b>5</b>
4.1	Use Cases . . . . .	5
4.1.1	Use case 1: Log in . . . . .	5
4.1.2	Use case 2: Add friend . . . . .	5
4.1.3	Use case 3: Request location . . . . .	6
4.1.4	Use case 4: Send location . . . . .	6
4.2	User story . . . . .	6
<b>5</b>	<b>Design</b>	<b>7</b>
5.1	Implementering . . . . .	7
5.2	Primære risici . . . . .	7

# App vision 1

---

Det overordnede formål med appen er at kunne lokalisere sine venner når man vil og når de vil. Appen adskiller sig fra de allerede eksisterende lokations apps ved at det ikke er en vedvarende lokations tracking, men blot aktuel lokations tracking.

Appen er tiltænkt på den måde at man starter med at tilføje nogle venner til appen. Herefter har man to muligheder; enten så kan man bede sine venner om deres lokation eller man kan sende sin egen lokation til dem. Vennen i den anden ende, som også har appen, skal så vælge hvorvidt vedkommende vil sende sin lokation og om han vil åbne den tilsendte lokation. På denne måde har brugeren af appen meget kontrol over hvornår lokationen deles og hvem der får den. Appen gør det nemt at finde din ven uden en masse klik og uden at venen skal bekymre sig om at blive overvåget i længere tid.

# Personlig vision 2

---

Som inspiration til opgaven er det blevet besluttet at benytte *Google Firebase* til persistering af data. Dette benyttes for at prøve kræfter med Firebase, som kan benyttes til mange mål. Derudover vil der blive brugt GPS-lokalitet for at få den sidste kendte lokalitet.

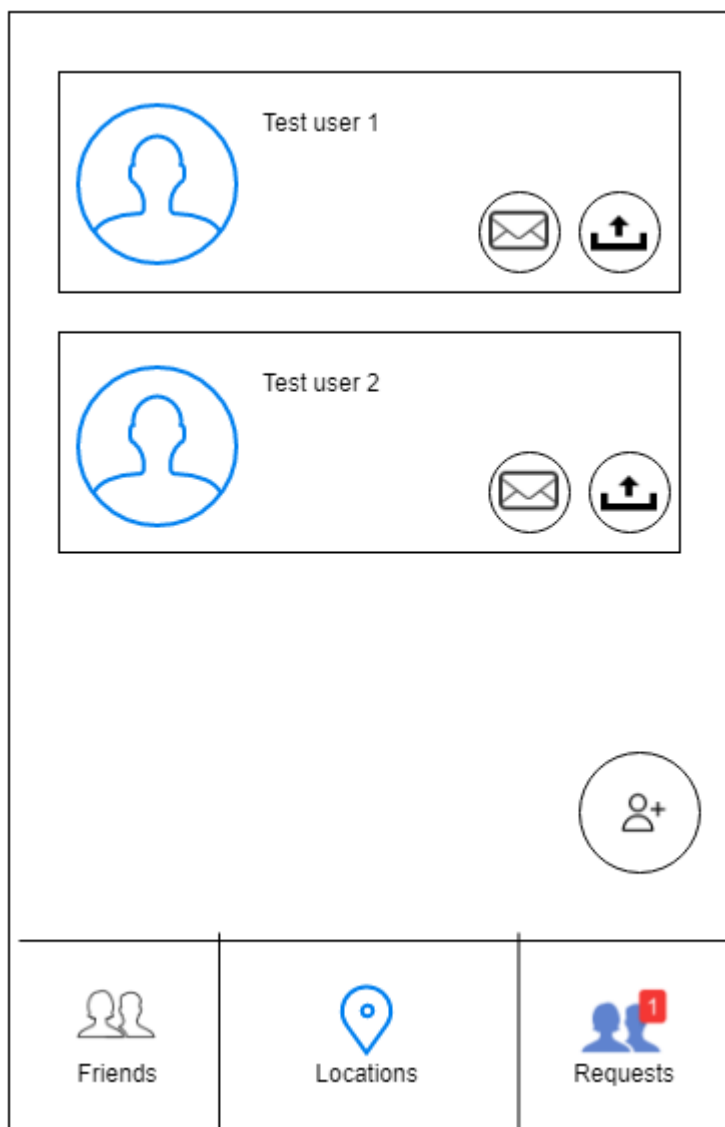
I dette projekt vil Firebase blive benyttet til at opbevarere brugerdata, heriblandt brugernavne og sidst kendte lokalitet.

# Context 3

---

Herunder ses et mockup af den primære sektion af brugergrænsefladen. Yderligere design kan ses under *Design Overview*. Billedet er genereret igennem draw.io, og ressourcerne er også taget derfra.

Figur 3.1: Mockup af friendslist



## 4.1 Use Cases

### 4.1.1 Use case 1: Log in

Use Case	Connect
Precondition	Application cache is cleared
Postcondition	User is logged in and ready to add friends
Path	<ol style="list-style-type: none"><li>1. User opens APPLICATION</li><li>2. User clicks Connect with..<ol style="list-style-type: none"><li>2.1 Facebook<ol style="list-style-type: none"><li>2.1.1 User selects User Name</li><li>2.1.2 User clicks OK.</li></ol></li><li>2.2 Google<ol style="list-style-type: none"><li>2.2.1 User selects User Name</li><li>2.2.2 User clicks OK.</li></ol></li><li>2.3 Custom login<ol style="list-style-type: none"><li>2.3.1 User clicks OK</li></ol></li></ol></li><li>3. User is logged in</li></ol>

### 4.1.2 Use case 2: Add friend

Use Case	Add friend
Precondition	User is logged in
Postcondition	User 2 is added as a friend
Path	<ol style="list-style-type: none"><li>1. User 1 presses button in bottom right corner</li><li>2. User 1 searches for username on user 2<ol style="list-style-type: none"><li>2.1 [<i>User 2 not existing in database</i>]</li></ol></li><li>3. User 1 adds user 2 as friend<ol style="list-style-type: none"><li>3.1 [<i>User 1 cancels operation</i>]</li></ol></li><li>4. User 2 accepts friend request<ol style="list-style-type: none"><li>4.1 [<i>User 2 declines friend request</i>]</li></ol></li></ol>
Extensions	<p>Extension 2.1 <i>User 2 not existing in database</i></p> <ol style="list-style-type: none"><li>2.1.1 Text telling that user isn't found</li><li>2.1.2 Use case continues from 3</li></ol> <p>Extension 3.1 <i>User 1 cancels operation</i></p> <ol style="list-style-type: none"><li>3.1.1 User 1 cancels operation</li><li>3.1.2 Use case ends</li></ol> <p>Extension 4.1 <i>USer 2 declines friend request</i></p> <ol style="list-style-type: none"><li>4.1.1 User 2 declines friend request</li><li>4.1.2 User 2 is not added as a friend</li></ol>

### 4.1.3 Use case 3: Request location

Use Case	Request Location
Precondition	User 1 is logged in User 1 and user 2 are connected
Postcondition	User 2 receives notification
Path	1. User 1 clicks request location on user 2 2. User 2 receives notification 3. User 2 accepts Extension 3.1 <i>User 2 declines</i> 4. User 1 receives notification with user 2's location
Extensions	Extension 3.1 <i>User 2 declines</i> 3.1.1. User 2 declines

### 4.1.4 Use case 4: Send location

Use Case	Send Location
Precondition	User is logged in. At least 2 minutes have gone by since this Use Case was performed.
Postcondition	Friend receives notification with location
Path	1. User clicks send location on desired friend 2. Friend receives notification 3. Friend clicks the notification 4. Google Maps opens and pinpoints the location

## 4.2 User story

Et scenarie hvor man kan benytte appen er f.eks. hvis man er på festival med nogle venner. I stedet for at skulle bruge en masse tid på at sende en masse sms'e for at finde hinanden så kan man bruge denne app og få den præcise lokation på ens venner med ganske få klik.

## 5.1 Implementering

Firestore vil blive benyttet Firestore som database som også skal stå for at sende notifikationer brugerne imellem. Der vil blive benyttet en service til alle firestore kald. Disse kald vil også være asynchrone. Der vil potentielt blive benyttet fragments til opbygningen af de forskellige faner; *Friends*, *Locations*, *Requests* på main aktiviteten som man kan se i afsnit 3.1.

## 5.2 Primære risici

Firestore samt fragments er nok de primære risici når appen skal udvikles, da det er disse to områder vi har mindst erfaringer med, men det bliver spændende at prøve at arbejde med.