



## OADE- Arkitektur for mobile apps

**Kent Arne Bjerke**  
**Per Stian Hoff**  
**Andrzej Thingstad**

Master i datateknologi

Innlevert: juni 2015

Hovedveileder: John Krogstie, IDI

Norges teknisk-naturvitenskapelige universitet  
Institutt for datateknikk og informasjonsvitenskap





Institutt for datateknikk  
og informasjonsvitenskap

# OADE - Arkitektur for mobile applikasjoner

Datateknologi, Masteroppgave

Per Stian Hoff  
Kent Arne Bjerke  
Andrzej Thingstad

Veiledet av  
John Krogstie

Juni 2015

Institutt for datateknikk og informasjonsvitenskap  
Norges tekniske-naturvitenskapelige universitet  
Trondheim, Norge

## **Abstract**

In recent times, social networking has become more common in today's society. Social networking uses various techniques to inform users and give them the opportunity to communicate information to other users. Newsfeed, push notifications and the ability to follow content and users are examples of techniques that are used.

This thesis has developed a mobile service in the form of a prototype application for Android. The application allows users to create events that other users can participate in without having to depend on knowing the user who created the event or hold an invitation. The events will be sorted and presented to each user based on the interests they have chosen. Whether there will be created an event that matches the interests of a user, the application will send a push notification to the appropriate users for making those aware of the event. There are also opportunities to communicate with others users of an event through a chat.

The prototype has been developed in accordance with the general principles of usability and end-user feedback to achieve a result that satisfies the involved stakeholders. It was also conducted a survey in which participants were able to test and give feedback on the final prototype. The results of this evaluation show an interest in an application that allows users to create events across acquaintance.

## Sammendrag

I senere tid har sosiale nettverk blitt mer vanlig i dagens samfunn. Sosiale nettverk benytter seg av ulike teknikker for å informere brukere og gi de muligheten til å formidle informasjon til andre brukere. Aktivitetsstrøm, push-notifikasjoner og muligheten til å følge innhold og brukere er eksempler på teknikker som blir brukt.

I denne oppgaven er det blitt utviklet en mobiltjeneste i form av en prototype applikasjon for Android. Applikasjonen gir brukere mulighet til å opprette uformelle arrangementer som andre brukere kan delta på uten å være avhengig av å kjenne brukeren som har opprettet arrangementet eller inneha en invitasjon. Arrangementene blir sortert og presentert for hver bruker basert på interesser de har valgt. Om det blir opprettet et arrangement som samsvarer med interessene til en bruker, vil det bli sendt ut en push-notifikasjon til de aktuelle brukerne for å gjøre de oppmerksomme på arrangementet. Det er også muligheter for å kommunisere med andre brukere av et arrangement gjennom en chat.

Prototypen er blitt utviklet i samsvar med generelle prinsipper for brukervennlighet og sluttbrukerens tilbakemeldinger for å oppnå et resultat som tilfredstiller de involverte interessentene. Det ble også gjennomført en spørreundersøkelse hvor deltakere fikk teste og komme med tilbakemeldinger på den endelige prototypen. Resultatene fra denne evalueringen viser en interesse for en applikasjon som tillater brukere å opprette arrangementer på tvers av bekjentskap.

## **Problemstilling**

Ulike sosiale nettverk gir brukere muligheten til å opprette arrangementer som andre brukere kan bli invitert til eller bli med på om arrangementet er åpent. Brukere som ikke er i bekjentskap med brukeren som har opprettet et arrangement som ikke er åpent, vil ikke være i stand til å søke seg fram til dette arrangementet. Arrangement som er åpent for alle er i mindretall på store sosiale nettverk som Facebook.

Opgaven vil bestå av å utvikle og evaluere en prototype som tillater brukere å opprette arrangementer som andre brukere kan delta på uten å være avhengig av en invitasjon eller ha bekjentskap til brukeren som har opprettet arrangementet. Prosjektet skal følge en design science research-prosess, samtidig som en artefakt skal utvikles og evalueres.

## Preface

Dette dokumentet inneholder vår masteroppgave i Datateknikk ved Institutt for datateknikk og informasjonsvitenskap ved Fakultetet for informasjonsteknologi, matematikk og elektroteknikk ved Norges teknisk-naturvitenskapelige universitet. Oppgaven ble skrevet våren 2015 i Trondheim med veiledning fra professor John Krogstie. Vi ønsker å takke Krogstie for hans gode veiledning gjennom masteroppgaven. Videre vil vi også rette en takk til Dr Shang Gao på IDI for hans veiledning knyttet til MSAM-analysen.

Per Stian Hoff  
Kent Arne Bjerke  
Andrzej Thingstad

# Innholdsfortegnelse

|          |  |           |
|----------|--|-----------|
| <b>1</b> | <b>Innledning</b>                              | <b>16</b> |
| 1.1      | Motivasjon . . . . .                           | 16        |
| 1.2      | Prosjektdefinisjon . . . . .                   | 17        |
| 1.3      | Leserguide . . . . .                           | 17        |
|          | Kapittel 2: Bakgrunn . . . . .                 | 17        |
|          | Kapittel 3: Forskningsmetode . . . . .         | 17        |
|          | Kapittel 4: Problem . . . . .                  | 17        |
|          | Kapittel 5: Applikasjonen . . . . .            | 17        |
|          | Kapittel 6: Resultat . . . . .                 | 17        |
|          | Kapittel 7: Diskusjon . . . . .                | 18        |
|          | Kapittel 8: Konklusjon . . . . .               | 18        |
| <b>2</b> | <b>Bakgrunn</b>                                | <b>19</b> |
| 2.1      | Sosiale nettverk . . . . .                     | 19        |
|          | 2.1.1 Meetup . . . . .                         | 20        |
|          | 2.1.2 Facebook . . . . .                       | 21        |
|          | 2.1.3 Google Plus + . . . . .                  | 22        |
|          | 2.1.4 VK . . . . .                             | 22        |
| 2.2      | Sosiale teknikker . . . . .                    | 23        |
| 2.3      | Utviklingsverktøy . . . . .                    | 24        |
|          | 2.3.1 Xamarin . . . . .                        | 24        |
|          | 2.3.2 Google App Engine . . . . .              | 25        |
| <b>3</b> | <b>Forskningsmetode</b>                        | <b>26</b> |
| 3.1      | Forskningsspørsmål . . . . .                   | 26        |
|          | Spørsmål 1: . . . . .                          | 26        |
|          | Spørsmål 2: . . . . .                          | 26        |
|          | Spørsmål 3: . . . . .                          | 26        |
| 3.2      | Design Science Research . . . . .              | 27        |
| 3.3      | Akseptansemodeller . . . . .                   | 30        |
|          | 3.3.1 Theory of Reasoned Action . . . . .      | 31        |
|          | 3.3.2 Theory of Planned Behavior . . . . .     | 31        |
|          | 3.3.3 Diffusion of Innovation Theory . . . . . | 32        |



|          |  |           |
|----------|--|-----------|
| 3.3.4    | Unified Theory of Acceptance and Use of Technology . . . . .                   | 34        |
| 3.3.5    | Technology Acceptance Model . . . . .  | 34        |
| 3.3.6    | Mobile Services Acceptance Model . . . . .                                     | 35        |
| 3.4      | Brukervennlighet . . . . .   | 37        |
| 3.4.1    | Heuristisk Evaluering . . . . .  | 37        |
| 3.4.2    | System Usability Scale . . . . .   | 41        |
| 3.5      | Metoder for innhenting av data . . . . .                                       | 41        |
| 3.5.1    | Nettundersøkelse . . . . .   | 41        |
| 3.5.2    | Spørreundersøkelse . . . . .   | 42        |
| 3.5.3    | Observasjon . . . . .  | 42        |
| 3.5.4    | Intervju . . . . .   | 43        |
| 3.5.5    | Prototype testing . . . . .  | 43        |
| <b>4</b> | <b>Problem</b>   | <b>44</b> |
| 4.1      | Personas . . . . .   | 44        |
| 4.1.1    | Mental helse, Jacob . . . . .  | 44        |
| 4.1.2    | Student ved Gløshaugen, Jackueline . . . . .                                   | 45        |
| 4.1.3    | Student ved Dragvoll, Ida . . . . .  | 46        |
| 4.1.4    | Jobber hos Statoil, Per . . . . .  | 47        |
| 4.2      | Scenario . . . . .   | 48        |
| 4.2.1    | Scenario 1: Finne studievenner, uten applikasjonen . . . . .                   | 48        |
| 4.2.2    | Scenario 2: Finne studievenner, med applikasjonen . . . . .                    | 48        |
| 4.2.3    | Scenario 3: Finne personer å være med på fritiden, med applikasjonen . . . . . | 49        |
| 4.2.4    | Scenario 4: Finne turkamerat, med applikasjonen . . . . .                      | 49        |
| 4.3      | Lage test oppgaver . . . . .   | 50        |
| 4.3.1    | System Usability Scale Test . . . . .  | 50        |
| 4.3.2    | Mobile Service Acceptanse Model Test . . . . .                                 | 51        |
| 4.4      | Krav . . . . .   | 51        |
| 4.4.1    | Bestemme krav . . . . .  | 51        |
| 4.4.2    | Funksjonelle krav . . . . .  | 52        |
| 4.4.3    | Ikke funksjonelle krav . . . . .   | 53        |
| 4.5      | Use Case . . . . .   | 53        |
| 4.5.1    | Use Case 1: Logg Inn . . . . .   | 55        |
| 4.5.2    | Use Case 2: Legg Til Mangefull Informasjon . . . . .                           | 56        |

|          |   |           |
|----------|---|-----------|
| 4.5.3    | Use Case 3: Registrer Interesser . . . . .                    | 57        |
| 4.5.4    | Use Case 4: Vis Informasjon . . . . .                         | 57        |
| 4.5.5    | Use Case 5: Opprett Arrangement . . . . .                     | 57        |
| 4.5.6    | Use Case 6: Finn Arrangement . . . . .                        | 58        |
| 4.5.7    | Use Case 7: Min Side . . . . .                                | 58        |
| 4.5.8    | Use Case 8: Endre Interesser . . . . .                        | 59        |
| 4.5.9    | Use Case 9: Valgt Arrangement . . . . .                       | 59        |
| 4.5.10   | Use Case 10: Slett Arrangement . . . . .                      | 60        |
| 4.5.11   | Use Case 11: Chat . . . . .                                   | 60        |
| 4.5.12   | Use Case 12: Delta På Arrangement . . . . .                   | 61        |
| 4.6      | Demografi . . . . .   | 61        |
| 4.6.1    | Informasjon om respondentene . . . . .                        | 62        |
| 4.6.2    | Respondentenes forhold til smarttelefoner og sosiale nettverk | 62        |
| 4.7      | Brukerinteresse og behov . . . . .                            | 63        |
| 4.7.1    | Respondentenes interesse for en aktivitetsportal . . . . .    | 63        |
| <b>5</b> | <b>Applikasjon</b>  | <b>65</b> |
| 5.1      | Tekniske detaljer . . . . .                                   | 65        |
| 5.1.1    | Xamarin Forms . . . . .                                       | 65        |
| 5.1.2    | Google App Engine . . . . .                                   | 66        |
| 5.1.3    | Programmeringsspråk . . . . .                                 | 67        |
| 5.1.4    | Arkitektur . . . . .  | 67        |
| 5.1.5    | Støtte for flere skjermstørrelser . . . . .                   | 69        |
| 5.1.6    | Javascript Object Notation . . . . .                          | 69        |
| 5.2      | Oversikt over funksjonalitet . . . . .                        | 71        |
| 5.2.1    | Innlogging med Facebook . . . . .                             | 79        |
| 5.2.2    | Presentere arrangement i applikasjonen . . . . .              | 83        |
| 5.2.3    | Chat funksjonalitet . . . . .                                 | 85        |
| 5.2.4    | Søkefelt . . . . .  | 87        |
| <b>6</b> | <b>Resultat</b>   | <b>90</b> |
| 6.1      | Demografi . . . . .   | 90        |
| 6.1.1    | Informasjon om respondentene for den heuristiske evalueringen | 90        |
| 6.1.2    | Informasjon om respondentene for MSASM . . . . .              | 91        |
| 6.2      | Evaluering av brukergrensesnitt . . . . .                     | 92        |
| 6.2.1    | Resultat av heuristisk evaluering . . . . .                   | 92        |

|          |  |            |
|----------|--|------------|
| 6.2.2    | Resultat av SUS . . . . .                              | 94         |
| 6.3      | Resultat av MSAM . . . . .                             | 96         |
| 6.3.1    | Deskriptive resultater . . . . .                       | 96         |
| 6.3.1.1  | Perceived Usefulness . . . . .                         | 96         |
| 6.3.1.2  | Perceived Ease of Use . . . . .                        | 97         |
| 6.3.1.3  | Trust . . . . .  | 98         |
| 6.3.1.4  | Personal Initiatives and Characteristics . . . . .     | 98         |
| 6.3.1.5  | Context . . . . .                                      | 99         |
| 6.3.1.6  | Intention of Use . . . . .                             | 100        |
| 6.3.2    | Statistisk analyse og resultater . . . . .             | 100        |
| 6.3.2.1  | Reliabilitet og Validitet . . . . .                    | 101        |
| 6.3.2.2  | Den strukturelle modellen . . . . .                    | 103        |
| <b>7</b> | <b>Diskusjon</b>                                       | <b>106</b> |
| 7.1      | Interesse for en slik applikasjon . . . . .            | 106        |
| 7.2      | Oppnå aksept for å bruke en slik applikasjon . . . . . | 107        |
| 7.2.1    | Oppnå høy brukervennlighet . . . . .                   | 109        |
| <b>8</b> | <b>Konklusjon og videre arbeid</b>                     | <b>111</b> |
| 8.1      | Konklusjon . . . . .                                   | 111        |
| 8.2      | Videre arbeid . . . . .                                | 112        |
| <b>A</b> | <b>Nettundersøkelse</b>                                | <b>113</b> |
| <b>B</b> | <b>Spørreskjema om brukervennlighet</b>                | <b>117</b> |
| <b>C</b> | <b>Spørreskjema MSAM</b>                               | <b>120</b> |
| <b>D</b> | <b>Spørreskjema SUS</b>                                | <b>128</b> |
|          | <b>Kilder</b>  | <b>131</b> |

## Liste av figurer

|    |   |     |
|----|---|-----|
| 1  | Applikasjonen Meetup . . . . .  | 20  |
| 2  | Applikasjonen Facebook . . . . .  | 21  |
| 3  | Applikasjonen Google Plus+ . . . . .                                    | 22  |
| 4  | Applikasjonen VK . . . . .  | 23  |
| 5  | Generell DSR-prosess . . . . .  | 28  |
| 6  | Peffers DSR-modell fra 2008. . . . .                                    | 28  |
| 7  | Theory of Reasoned Action . . . . .                                     | 31  |
| 8  | Theory of Planned Behavior . . . . .                                    | 32  |
| 9  | Adopsjonskategoriene . . . . .  | 33  |
| 10 | Unified Theory of Acceptance and Use of Technology . . . . .            | 34  |
| 11 | Technology Acceptance Model . . . . .                                   | 35  |
| 12 | Mobile Services Acceptance Model . . . . .                              | 37  |
| 13 | Heuristisk evaluering . . . . .   | 38  |
| 14 | Jacob, 29 år . . . . .  | 45  |
| 15 | Jackueline, 24 år . . . . .   | 46  |
| 16 | Ida, 19 år . . . . .  | 47  |
| 17 | Per, 56 år . . . . .  | 48  |
| 18 | Første gang brukeren åpner applikasjonen . . . . .                      | 54  |
| 19 | En bruker åpner applikasjonen etter å ha logget inn tidligere . . . . . | 55  |
| 20 | Lik kode kjørt på forskjellige plattformer . . . . .                    | 66  |
| 21 | Arkitekturen MVVM . . . . .   | 68  |
| 22 | Applikasjonens interesseside og hovedside . . . . .                     | 72  |
| 23 | Applikasjonens veiledningssider . . . . .                               | 73  |
| 24 | Side for å opprette et arrangement . . . . .                            | 74  |
| 25 | Liste med alle tilgjengelige arrangement . . . . .                      | 75  |
| 26 | Applikasjonens profilside . . . . .                                     | 77  |
| 27 | Informasjon om valgt arrangement . . . . .                              | 78  |
| 28 | Kommunikasjon med andre brukere . . . . .                               | 86  |
| 29 | Liste over arrangement . . . . .  | 88  |
| 30 | Den strukturelle modellen . . . . .                                     | 104 |
| 31 | Respondentenes svar fra nettundersøkelsen . . . . .                     | 107 |

## Liste av tabeller

|    |   |    |
|----|---|----|
| 1  | Topp 15 mest populære sosiale nettverk av eBizMBA [5] . . . . .         | 19 |
| 2  | Retningslinjer for DSR . . . . .  | 30 |
| 3  | Funksjonelle krav . . . . .   | 52 |
| 4  | Ikke funksjonelle krav . . . . .  | 53 |
| 5  | Use Case 1: Logg inn . . . . .  | 56 |
| 6  | Use Case 2: Legg Til Mangelfull Informasjon . . . . .                   | 56 |
| 7  | Use Case 3: Registrer Interesser . . . . .                              | 57 |
| 8  | Use Case 4: Vis Informasjon . . . . .                                   | 57 |
| 9  | Use Case 5: Opprett Arrangement . . . . .                               | 58 |
| 10 | Use Case 6: Finn Arrangement . . . . .                                  | 58 |
| 11 | Use Case 7: Min Side . . . . .  | 59 |
| 12 | Use Case 8: Endre Interesser . . . . .                                  | 59 |
| 13 | Use Case 9: Valgt Arrangement . . . . .                                 | 60 |
| 14 | Use Case 10: Slett Arrangement . . . . .                                | 60 |
| 15 | Use Case 11: Chat . . . . .   | 61 |
| 16 | Use Case 12: Delta På Arrangement . . . . .                             | 61 |
| 17 | Informasjon om respondentene . . . . .                                  | 62 |
| 18 | Respondentenes forhold til smarttelefoner og sosiale nettverk . . . . . | 63 |
| 19 | Respondentenes interesse for en aktivitetsportal . . . . .              | 63 |
| 20 | Respondentenes svar på applikasjonens funksjoner . . . . .              | 64 |
| 21 | Beskrivelse av visninger . . . . .                                      | 75 |
| 22 | Beskrivelse av indikatorer . . . . .                                    | 76 |
| 23 | Beskrivelse av knappene . . . . .                                       | 76 |
| 24 | Informasjon om respondentene for den heuristiske evalueringen . . . . . | 91 |
| 25 | Informasjon om respondentene for MSASM . . . . .                        | 92 |
| 26 | Problemer knyttet til brukergrensesnittet . . . . .                     | 93 |
| 27 | Resultat av SUS-test . . . . .  | 95 |
| 28 | Respondentenes SUS poengsum . . . . .                                   | 95 |
| 29 | Oppsummering av svar for PU . . . . .                                   | 97 |
| 30 | Statisk oppsummering av spørsmålene knyttet til PU . . . . .            | 97 |
| 31 | Oppsummering av svar for PEOU . . . . .                                 | 97 |
| 32 | Statisk oppsummering av spørsmålene knyttet til PEOU . . . . .          | 97 |
| 33 | Oppsummering av svar for Tillit . . . . .                               | 98 |

|    |  |     |
|----|--|-----|
| 34 | Statisk oppsummering av spørsmålene knyttet til Tillit . . . . .         | 98  |
| 35 | Oppsummering av svar for PIAC . . . . .                                  | 99  |
| 36 | Statisk oppsummering av spørsmålene knyttet til PIAC . . . . .           | 99  |
| 37 | Oppsummering av svar for Kontekst . . . . .                              | 100 |
| 38 | Statisk oppsummering av spørsmålene knyttet til Kontekst . . . . .       | 100 |
| 39 | Oppsummering av svar for Bruksintensjon . . . . .                        | 100 |
| 40 | Statisk oppsummering av spørsmålene knyttet til Bruksintensjon . . . . . | 100 |
| 41 | Analyse av reliabilitet for hvert tema . . . . .                         | 101 |
| 42 | Factor loadings, composite reliability og AVE for hvert tema . . . . .   | 102 |
| 43 | Discriminant Validity . . . . .  | 103 |
| 44 | Test av hypotesene basert på P-verdi og Sti koeffisient . . . . .        | 105 |

## Liste av kode

|    |   |    |
|----|---|----|
| 1  | Google Cloud Endpoint metode . . . . .                          | 67 |
| 2  | Applikasjonen henter alle arrangement fra serveren . . . . .    | 70 |
| 3  | Egentilpasset renderer for Android . . . . .                    | 79 |
| 4  | OnActivityResult i MainActivity.cs . . . . .                    | 80 |
| 5  | Applikasjonsinformasjon som brukes av Facebook (AppDelegate.cs) | 80 |
| 6  | OnActivated og OpenUrl i AppDelegate.cs . . . . .               | 81 |
| 7  | Hele FacebookLoginRendererIos.cs som arver fra ButtonRenderer . | 81 |
| 8  | GetUserInfoAsync i Backend.cs . . . . .                         | 82 |
| 9  | ResizeImage i EventBox.cs . . . . .                             | 83 |
| 10 | ResizeImage i ImageResizer.cs . . . . .                         | 84 |
| 11 | ResizeImageAndroid i ImageResizer.cs . . . . .                  | 84 |
| 12 | ResizeImageIOS i ImageResizer.cs . . . . .                      | 85 |
| 13 | CheckForNewMessages i ChatPage.cs . . . . .                     | 86 |
| 14 | SearchClicked i EventPage.cs . . . . .                          | 88 |

## Ordliste

**API** - Application Programming Interface  
**AVE** - Average Variance Extracted  
**C** - Context  
**CR** - Composite Reliability  
**DOI** - Diffusion of Innovation Theory  
**DP** - Density-independent pixel  
**DSR** - Design Science Research  
**FK** - Funksjonelle krav  
**FL** - Factor Loading  
**GPS** - Global Positioning System  
**HE** - Heuristisk Evaluering  
**ICR** - Internal Consistency Reliability  
**IFK** - Ikke funksjonelle krav  
**ITU** - Intention to Use  
**JSON** - JavaScript Object Notation  
**MSAM** - Mobile Services Acceptance Model  
**MVVM** - Model-View-ViewModel  
**NTNU** - Norges tekniske-naturvitenskapelige universitet  
**OOM** - Out Of Memory  
**PEOU** - Perceived Ease of Use  
**PIAC** - Personal Initiatives and Characteristics  
**PLS** - Partial Least Square  
**PU** - Perceived Usefulness  
**PX** - Pixel  
**RA** - Reasoned Action  
**SDK** - Software Development Kit  
**SUS** - System Usability Scale  
**T** - Trust  
**TAM** - Technology Acceptance Model  
**TPB** - Theory of Planned Behavior  
**TRA** - Theory of Reasoned Action  
**UC** - Use Case  
**UML** - Unified Modeling Language



**UTAUT** - Unified Theory of Acceptance and Use of Technology

**WTLL** - Wireless Trondheim Living Lab

**XAML** - Extensible Application Markup Language

# 1 Innledning

Dette kapitlet inneholder en introduksjon til oppgaven. Den vil beskrive hvordan og hvorfor denne masteroppgaven er blitt gjort. Dette arbeidet er blitt gjort i forbindelse med Wireless Trondheim Living Lab (WTLL) [24]. WTLL er en living lab som blir drevet delvis alene av NTNU og delvis i samarbeid med andre selskaper inkludert Wireless Trondheim.

## 1.1 Motivasjon

Sosiale nettverk har blitt en viktig del av hverdagen til veldig mange mennesker, da de hjelper oss å kommunisere og holde kontakten med mennesker uavhengig av hvor de er i verden. Populære nettsider og applikasjoner som Facebook, Twitter, Instagram og LinkedIn er alle eksempler på sosiale nettverk som er informative, underholdende og som gjør oss oppmerksomme på ulike hendelser, situasjoner og arrangementer som skjer i samfunnet eller verden forøvrig.

Facebook tilbyr brukere blant annet muligheten for å opprette arrangementer. Et arrangement består av en overskrift, en beskrivelse, sted og et tidspunkt. Videre har brukeren bak arrangementet muligheten for å invitere andre Facebook-brukere til sitt arrangement. Brukeren kan også velge å holde arrangementet åpent for alle. Gjennom dette arrangementet kan alle aktuelle brukere bli oppdatert på endringer i tid, sted eller generell kommunikasjon vedrørende arrangementet.

Mennesker er sosiale vesener. Enten man er en introvert eller ekstrovert må vi vite at vi er del av et fellesskap. Vi trenger å føle at vi har andre som vi kobler sammen med og som deler våre fellestrekk. I sin forskningsartikkel “A Theory of Human Motivation” [16] plasserer psykolog Abraham Maslow behovet for tilhørighet like etter grunnleggende sikkerhets - og fysiologiske behov i hans hierarki av menneskelige behov.

Tall fra en undersøkelse gjennomført av Røde Kors i 2012 [13] viser at 1 av 3 ungdom mellom 18 og 29 år er plaget av ensomhet ofte eller av og til. Dette er tall høyere enn i andre aldersgrupper. Ensomhet kan handle om at man har få venner, men kan også handle om at man ikke føler seg inkludert i et fellesskap. Ensomhet er heller ikke en permanent tilstand, men en følelse som kan komme og gå.

Denne masteroppgaven kommer til å utforske muligheten for hvordan en ny applikasjon med fokus på å opprette arrangementer som hvem som helst kan bli med på, kan bidra til å bringe sammen personer med like interesser og på det viset

hjelpen individer som opplever ensomhet på grunn av mangel på et fellesskap.

## 1.2 Prosjektdefinisjon

Det er blitt utviklet en mobil applikasjon som tilbyr muligheten til å opprette og bli med på uformelle arrangementer. Arrangementene kan bli laget av enkelt personer, bedrifter, organisasjoner eller andre grupper.

Et forskningsprosjekt, se digitalt vedlegg 1, ble utført for å få en forståelse av hvilke tilgjengelige teknologier som kunne benyttes i forbindelse med utviklingen av applikasjonen. Det ble også vurdert hvilke teknologier som egnet seg best med tanke på utviklingen av applikasjonen. Dette medførte i en applikasjon som gjør det mulig å opprette og bli med på uformelle aktiviteter. Hensikten med denne applikasjonen er å få mennesker til å bli kjent med andre og danne et større sosialt nettverk. Deretter ble det gjort en evaluering av applikasjonen hvor innhentede data ble brukt for å bestemme interessen og for å evaluere faktorer som påvirker brukernes akseptanse av applikasjonen. Resultatet av dette ble brukt for å trekke en konklusjon i denne rapporten.

## 1.3 Leserguide

**Kapittel 2: Bakgrunn** inneholder en beskrivelse av de mest populære sosiale nettverkene og teknikker de benytter seg av. Kapitlet beskriver også kort om utviklingsverktøy som er blitt benyttet under utvikling av applikasjonen.

**Kapittel 3: Forskningsmetode** inneholder forskningsspørsmål og beskrivelse av benyttet forskningsmetode.

**Kapittel 4: Problem** inneholder en oversikt over problemet gjennom bruk av personas og scenarioer.

**Kapittel 5: Applikasjonen** inneholder tekniske detaljer om applikasjonen og applikasjonens funksjonalitet, både synlig og usynlig for brukeren.

**Kapittel 6: Resultat** inneholder resultater knyttet til brukervennlighetstestene som er blitt benyttet i evalueringen av applikasjonen, sammen med resultatene av akseptansetestingen som er blitt foretatt.

**Kapittel 7: Diskusjon** inneholder en diskusjon knyttet til resultatene som ble presentert i kapittel 6.

**Kapittel 8: Konklusjon** inneholder en konklusjon av masteroppgaven sammen med ulike forbedringer og videre arbeid som kan bli utført.

## 2 Bakgrunn

Dette kapitlet tar for seg de mest populære sosiale nettverkene og teknikker de benytter seg av. Det er også beskrevet kort om utviklingsteknologi og verktøy som er blitt benyttet under utviklingen av applikasjonen.

### 2.1 Sosiale nettverk

Et sosialt nettverk er et nettverk av menneskelige forhold som danner en forbindelse til andre mennesker. Sterke og svake bånd avspeiler graden av tilgjengelighet, bekjentskap og fortrolighet mellom personene, organisasjonene eller gruppene.

Det finnes mange populære sosiale nettverk. Den største er Facebook med sine 900 millioner unike månedlige besøkende etterfulgt av Twitter som har 310 millioner månedlige besøkende (måling gjort i april 2015) [5]. I tabell 1 vises de 15 mest populære sosiale nettverkene i april 2015.

**Tabell 1:** Topp 15 mest populære sosiale nettverk av eBizMBA [5]

| Navn          | Unike månedlige besøkende |
|---------------|---------------------------|
| Facebook      | 900 000 000               |
| Twitter       | 310 000 000               |
| LinkedIn      | 255 000 000               |
| Pinterest     | 250 000 000               |
| Google Plus + | 120 000 000               |
| Tumblr        | 110 000 000               |
| Instagram     | 100 000 000               |
| VK            | 80 000 000                |
| Flickr        | 65 000 000                |
| Vine          | 42 000 000                |
| Meetup        | 40 000 000                |
| Tagged        | 38 000 000                |
| Ask.fm        | 37 000 000                |
| MeetMe        | 15 500 000                |
| ClassMates    | 150 000 000               |

Facebook, Google Plus + og VK er alle sosiale nettverk som tilbyr funksjonalitet for å opprette og å bli med på aktiviteter andre har laget. Nettverkene er laget for at mennesker skal kunne legge til venner. På den måten kan de holde kontakten selv om de ikke befinner seg på samme grafiske område. Det er også lagt til funksjonalitet som gjør det mulig å opprette grupper, arrangementer eller sider som kan brukes som en informasjonsportal for en gitt aktivitet. Meetup er et sosialt nettverk som gjør det mulig å opprette uformelle aktiviteter som alle kan

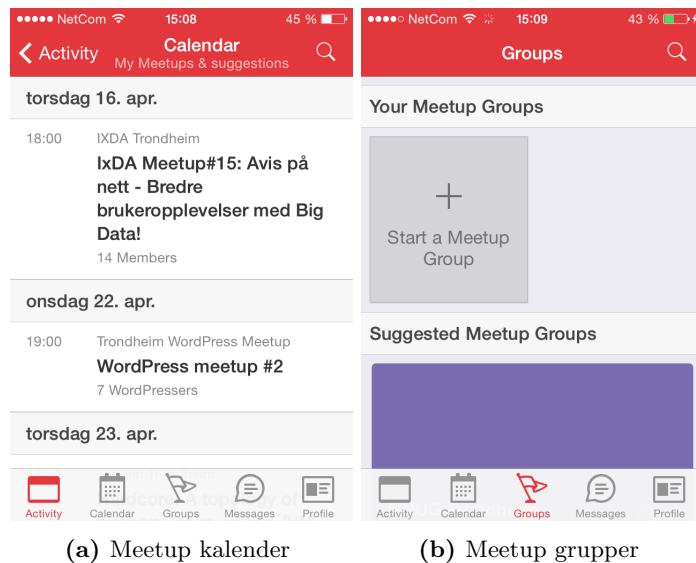
være med på. Personen som lager aktiviteten trenger ikke være venn med de som deltar. Meetup er utviklet rundt konseptet aktiviteter og tilpasset denne funksjonaliteten.

### 2.1.1 Meetup

Meetup er et sosialt nettverk hvor det kan opprettes grupper som andre mennesker kan bli medlem av. Gruppene kan annonsere aktiviteter som medlemmene av gruppen kan melde seg på, se figur 1b.

For å kunne opprette en gruppe må man betale et månedlig beløp til Meetup. De har to betalingsplaner, Basic og Unlimited, som kan kjøpes. Basic gjør det mulig å ha inntill 50 medlemmer og 4 personer som organiserer en gruppe. Planen Unlimited gjør det mulig å ha et ubegrenset antall medlemmer og organisatorer av gruppen. De jobber også med en betalingsplan, Premium, som også gir gruppen ekstra forfremmelse, samt tilgang til statistikk og data [17].

Den mobile applikasjonen Meetup har ikke implementert all funksjonalitet i applikasjonen. Ved oppretting av et arrangement blir brukeren sendt til en nettside hvor han/hun kan lage arrangementet.



Figur 1: Applikasjonen Meetup

### 2.1.2 Facebook

Facebook er et sosialt nettverk der brukeren lager en profil med personlig informasjon. Ved å ha en profil på Facebook kan man opprette relasjoner til andre brukere. Det er mulighet for å opprette og å bli med på arrangementer som andre har laget. Nettstedet ble opprinnelig laget for universitetsstudenter i USA, men ble i senere tid åpent for hele verden.

Ved oppretting av et arrangement kan brukeren selv bestemme hvem som skal ha lov til å melde seg på arrangementet. Det finnes også andre måter å invitere brukere til arrangementet, se figur 2b. På offentlige arrangement kan alle som er medlem av facebook bli med. Arrangement som har en åpen invitasjon kan alle inviterte brukere og venner til de inviterte brukerne bli med. Det er også mulig å lage arrangement som kun er tilgjengelig for personer som er invitert av verten eller gjestene.

Facebook bruker standard GPS posisjonering for å finne arrangement som er i nærheten av brukeren. Dette blir presentert i en liste med arrangementer i applikasjonen, se figur 2a. Det blir også presentert en liste med arrangement som er populære i brukerens nettverk.



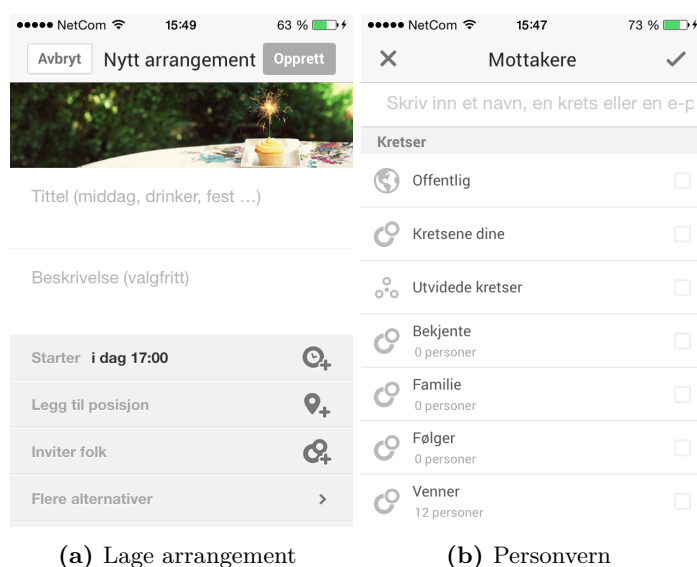
Figur 2: Applikasjonen Facebook

### 2.1.3 Google Plus +

Google Plus + er et sosialt nettverk som tilbyr mye av den samme funksjonaliteten som Facebook. Nettverket ble utviklet av Google og lansert 28. juni 2011 for enkelte brukere. 20 September 2011 ble nettverket åpnet for alle over 18 år. Fire måneder etter, 26. januar 2012, ble Google Plus + tilgjengelig for alle over 13 år.

Det er mulig å opprette arrangement på Google Plus +. Verten kan selv bestemme hvem som skal bli invitert til arrangementet, se figur 3b. Google Plus + bruker GPS-teknologi for å legge til adressen til et arrangement.

Funksjonalitet for å kunne snakke med andre medlemmer av arrangementet er implementert ved hjelp av en chat.



Figur 3: Applikasjonen Google Plus+

### 2.1.4 VK

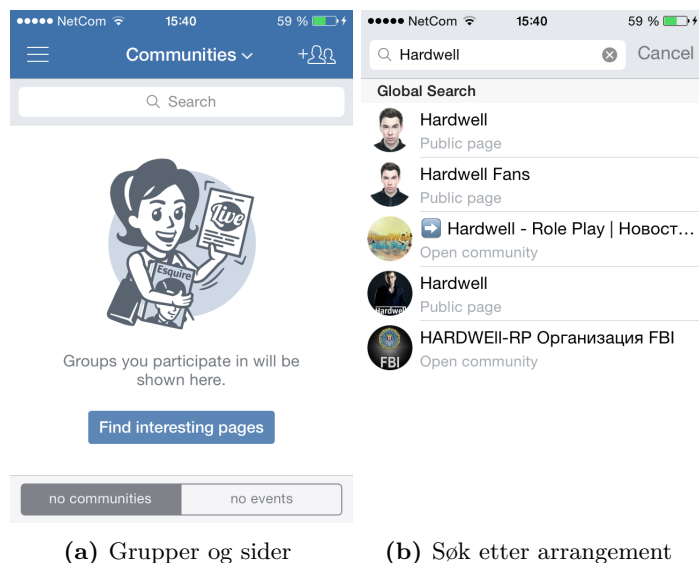
VK er et russisk sosialt nettverk som har oppnådd stor popularitet i den russiskspråklige delen av verden. I April 2015 var VK i følge alexa.com sin statistikk den 36 mest besøkte nettsiden i verden og nummer to i Russland [1]. Ifølge nettstedet selv er de det største sosiale nettverket i Europa.

*“VK is the largest European social network with more than a 100 million active users.”*

– www.VK.com



En bruker kan lage et arrangement ved å gå inn på [www.vk.com](http://www.vk.com) i nettleseren eller benytte seg av Android applikasjonen til VK. Det er ikke mulig å lage et arrangement i iOS applikasjonen som er blitt utviklet for vk, se figur 4. Når et arrangement skal opprettes må verten skrive inn generell informasjon om arrangementet og ved hjelp av GPS posisjonering finne adressen til arrangementet. I likhet med Meetup, Facebook og Google Plus+ har også VK implementert funksjonalitet for å snakke med andre medlemmer av et arrangement.



Figur 4: Applikasjonen VK

## 2.2 Sosiale teknikker

Sosiale nettverk bruker teknikker for å gjøre brukeropplevelsen bedre. Teknikkene oppdaterer og gir brukeren mulighet til å holde kontakten med andre brukere. Aktivitetsstrøm, push-notifikasjoner og muligheten til å følge innhold og brukere er teknikker som ofte blir brukt på sosiale nettverk.

En aktivitetsstrøm er en liste av nylige aktiviteter publisert av andre brukere. Facebook sin “News Feed” er et eksempel på en slik aktivitetsstrøm. Ved bruk av aktivitetsstrømmer kan brukerne enkelt holde seg oppdatert på hva som skjer i nettverket.

En push-notifikasjon er en melding som blir sendt til brukeren for å holde han/hun oppdatert på hva som skjer i applikasjonen.

Muligheten for å følge brukere eller melde seg på arrangementer er en mye

brukt teknikk på sosiale nettverk. Ved å koble en bruker til et arrangement kan man lett vise informasjon om arrangementet senere. Når brukerne er koblet til et arrangement er det enkelt å sende push-notifikasjoner ved eventuelle oppdateringer eller endringer i arrangementet.

## 2.3 Utviklingsverktøy

Valg av riktig utviklingsverktøy er viktig for å sikre god ytelse, lave kostnader og minst mulig vedlikehold. I dette kapitlet beskrives kort utviklingsverktøy som ble brukt under utvikling av applikasjonen. Xamarin og Google App Engine er verktøy som begge sikrer god ytelse, lave kostnader og lite vedlikehold, se digitalt vedlegg 1.

### 2.3.1 Xamarin

Xamarin er en plattform som tillater brukere å skrive applikasjonene sine i *C#* og dele kodebasen over flere plattformer som blant annet iOS, Android, Windows Phone og Mac. Dette sørger for at en kan bruke det samme språket, de samme API'ene og den samme strukturen for hver plattform som applikasjonen blir utviklet for. Alt som kan gjøres med Objective-C, Swift eller Java, kan også gjøres med *C#* i Xamarin.

Xamarin tilbyr native brukergrensesnitt som sørger for at applikasjoner blir utviklet med native brukergrensesnitt elementer for den plattformen det utvikles for. Det tilbys også aksess til native API'er og native ytelse som bidrar til at applikasjoner ikke bare ser native ut, men også oppleves som det.

Xamarin er blitt et populært verktøy som mange benytter seg av når det kommer til utvikling av applikasjoner som skal fungere på flere forskjellige plattformer [30]. Mye av grunnen til dette er blant annet at det er mulig å gjenbruke store deler av kodebasen på tvers av flere plattformer og det er mulig å skrive applikasjoner i *C#* som er et språk med høy popularitet verden over og med et stort økosystem rundt seg i form av .NET rammeverket til Microsoft.

Xamarin tilbyr også en omfattende nettportal hvor utviklere kan finne guider, videoer og eksempelapplikasjoner som viser hvordan man kommer i gang med utvikling, men også hvordan ulik funksjonalitet skal implementeres i Xamarin. Videre har Xamarin en Component-Store hvor det er mulig å finne en haug med ulike komponenter som kan implementeres inn i ulike Xamarin-applikasjoner. Hvem som helst kan utvikle disse komponentene og selv om noen av dem koster et par

kroner er mesteparten av de gratis. Personer kan også skrive ønsker til hvilke komponenter de mener mangler på plattformen, slik at andre utviklere kan utvikle disse komponentene. Xamarin tilbyr også same-day support for systemoppdateringer på blant annet Android og iOS.

### **2.3.2 Google App Engine**

Google App Engine opererer med en oppetid på 99.95%, noe som tilsvarer rundt 4,38 timer i året. Her er det viktig å bemerke seg at Google opererer med spesielle betingelser i sin Service Level Agreement som sier at all nedetid som ikke overskrider fem minutter med kontinuerlig nedetid, ikke teller mot den totale nedetiden til applikasjonen. Dermed vil man ikke få økonomisk godtgjørelse på bakgrunn av denne type nedetid [6].

Det er også mulig med automatisk skalering av server instanser. Dette løses med et orchestration rammeverk som involverer et verktøy implementert som en App Engine applikasjon. Verktøyet spør periodisk alle Compute Engine instanser for status [7]. Avhengig av applikasjonen, så kan denne statusen være belastning på CPU, minnebruk eller hvor mange oppgaver som venter på å bli gjort. Basert på brukerdefinerte kriterier, starter rammeverket opp nye instanser om det er behov for det og slår de av når det ikke er behov.

Det er også mulig å endre skalering manuelt ved blant annet å estimere et gjennomsittlig behov for ressurser eller ved å prøve å estimere når belastningen er på sitt høyeste og planlegge utifra det. Problemet med begge disse tilnærmingene er at det er vanskelig å beregne slik bruk, samtidig som det er kostbart om spådommene ikke slår til. Google App Engine har støtte for Java, Python, PHP og GO.

Google App Engine tilbyr en omfattende mengde ulike konfigurasjoner og er å finne på Google sine hjemmesider [8].

### 3 Forskningsmetode

Dette kapitlet presenterer et sett med forskningsspørsmål og hvordan forskningen ble gjort ved å beskrive den valgte forskningsmetoden. Videre vil en rekke brukervennlighetsprinsipper og andre evalueringsmetoder bli brukt for å vurdere prototypen som skal utvikles, sammen med akseptansesmodeller.

#### 3.1 Forskningsspørsmål

**Spørsmål 1:** Finnes det interesse for en applikasjon som tillater brukere å opprette arrangementer som andre personer en nødvendigvis ikke kjenner kan bli med på?

Facebook er det største sosiale nettverket som eksisterer, med 900 millioner unike besøkende hver måned [5]. Noen av utfordringene ved å lete etter arrangementer på sosiale nettverk som Facebook er blant annet at ikke alle arrangementer er tilgjengelige for alle brukere. Dermed vil oppgaven med å forsøke og oppdrive aktiviteter en kan bli med på være vanskeligere for en bruker, da vedkommende ikke har tilgang til alle arrangementene som er blitt registrert. Videre vil det være en utfordring for en bruker som befinner seg på et nytt sted å invitere brukere til sitt arrangement om vedkommende sin sosiale krets på Facebook ikke befinner seg på det samme stedet.

Ved å foreta en spørreundersøkelse på en relevant brukergruppe og spørre dem om de hadde vært interessert i å bruke en slik tjeneste, kan det bidra til større innsikt om det er behov for en slik applikasjon.

**Spørsmål 2:** Hvis det er en interesse for en slik applikasjon, hvordan kan man oppnå brukerenes aksept for å bruke en slik applikasjon?

Det finnes flere akseptansesmodeller basert på ulike teorier i sosialpsykologien som skal gjøre det mulig å teste brukere sin antatte holdning til adopsjon og bruk av ny teknologi. Ved å bruke en akseptansesmodell som er tilpasset denne masteroppgaven kan en måle ulike faktorer som påvirker brukerens intensjon om å bruke applikasjonen og dermed presentere frem et resultat på dette spørsmålet.

**Spørsmål 3:** Hvordan skal en presentere informasjonen knyttet til en slik applikasjon på en mobil enhet, slik at brukervennligheten blir høy?

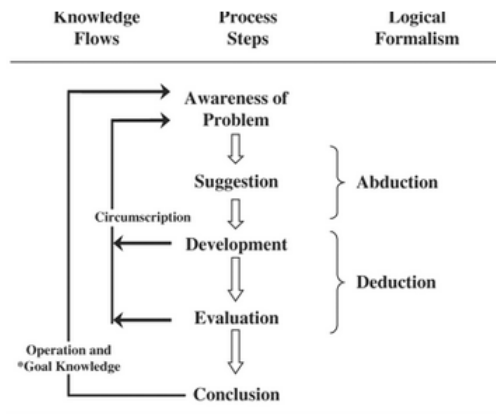
Det er flere faktorer som må tas hensyn til når en skal utvikle en applikasjon for en mobil enhet. Faktorer som responsivitet, naivgeringsmodeller, presentasjon av data og brukervennlighet er viktige med tanke på applikasjonen som skal utvikles gjennom denne masteroppgaven.

Ved å følge ulike prinsipper vedrørende brukervennlighet under utvikling av applikasjonen kan en minimere antall problemer i brukergrensesnittet og hjelpe brukere å finne arrangementer som er aktuelle for dem. Deretter kan dette brukergrensesnittet testes mot brukere for å avgjøre om brukervennligheten er tilfredstillende.

### 3.2 Design Science Research

Denne masteroppgaven følger en design science research (DSR) metodikk. Denne metodikken er et sett av analytiske teknikker for å utføre forskning på informasjonssystemer [29]. DSR innebærer utvikling av en ny eller en innovativ artifakt og analyse av bruken og ytelsen til denne artifakten. Figur 5 viser den generelle prosessen som blir fulgt i DSR:

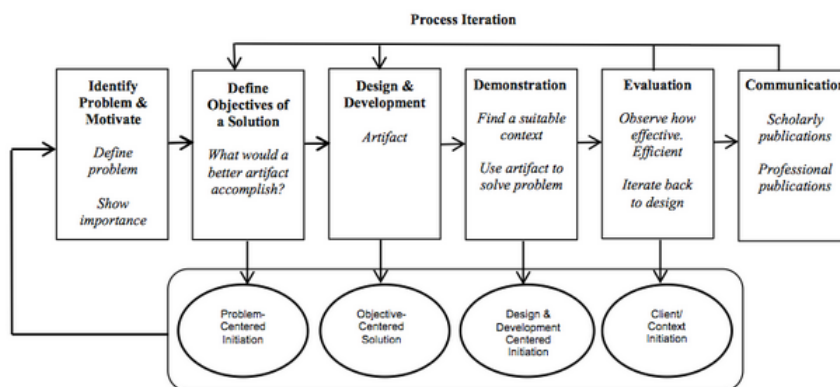
- **Problem:** En bevissthet rundt et problem som en ønsker å finne en løsning på eller innføre ny informasjon som videre kan brukes til å løse problemet. Resultatet av denne fasen er et forslag på hva som skal eller burde gjøres.
- **Forslag:** Forslagsfasen følger umiddelbart bak resultatet av problemfasen. Hvis en har investert betydelig innsats i et problem, men aktuelle individer ikke anser et forslag som et steg nærmere i å løse det gitte problemet vil forslaget bli satt til side. Denne fasen er et viktig kreativt steg mot en løsning.
- **Utvikling:** Løsningen foreslått i forslagsfasen blir videre utviklet og implementert i denne fasen. Teknikker og verktøy som skal brukes i denne fasen vil selvfølgelig variere avhengig av artifakten som skal produseres.
- **Evaluering:** Når artifakten er utviklet vil den bli evaluert i forhold til implisitte kriterier etablert i forslagsfasen. Kvantitative og kvalitative avvik skal noteres og forsøksvis forklares.
- **Konklusjon:** Denne fasen er slutten på en spesifikk forskningscyklus hvor resultatet blir beskrevet.



**Figur 5:** Generell DSR-prosess

Det eksisterer en rekke DSR-modeller som kan brukes til utvikling av informasjonssystemer. Modellen som blir brukt under denne avhandlingen følger Peffers DSR-modell fra 2008, se figur 6. Denne modellen har noen endringer i forhold til den generelle DSR-prosessen:

- Problemfasen blir delt opp i to faser; “Identifiser problemet & motivasjon” og “Definer krav til løsning”.
- Forslags - og utviklingsfasen blir slått sammen til en fase; “Design & utvikling”.
- Evalueringsfasen blir delt opp i to faser; “Demonstrasjon” og “Evaluering”.
- Konklusjonsfasen endrer navn til “Kommunikasjonsfasen”.



**Figur 6:** Peffers DSR-modell fra 2008.

Det grunnleggende prinsippet bak DSR er at kunnskap og forståelse av et problem og dens løsning er viktig med tanke på utviklingen og bruken av artifakten som skal munne ut på slutten av forskningssyklusen. På bakgrunn av dette ble Hevners sju retningslinjer benyttet under denne masteroppgaven [9], i tillegg til Peffers DSR-modell. Oppsummering av Hevners sju retningslinjer er å finne i tabell 2.

- Retningslinje nummer 1: En hensiktsmessig artifakt må bli utviklet. I denne masteroppgaven vil artifakten bli utviklet i form av en mobil applikasjon på både Android og iOS. Applikasjonen gjør det mulig for brukere å opprette og melde seg på arrangementer opprettet av andre brukere. Brukergrensesnitt presenterer tilgjengelig informasjon på en oversiktlig måte og gjør det lett tilgjengelig for alle brukerne.
- Retningslinje nummer 2: Målet med forskning på informasjonssystemer er å tilegne seg kunnskap og forståelse som gjør det mulig å utvikle og implementere teknologibaserte løsninger på uløste problemer. Målet med applikasjonen er å tilby brukere muligheten for å finne andre personer som ønsker å bli med på ulike arrangementer basert på interesser eller lokasjon. Videre skal det være mulig for brukere å bli med på arrangementer som befinner seg i nærområdet uten å være avhengig av å kjenne brukeren som har opprettet arrangementet eller inneha en invitasjon.
- Retningslinje nummer 3: Nyttens, kvaliteten og effektiviteten til artifakten som skal utvikles må bli evaluert på bakgrunn av gode evalueringsmetoder. Velkjente prinsipper innenfor utvikling av brukergrensesnitt som Nielsens 10 heuristikker, heuristisk evaluering og SUS ble brukt for å bedømme designet på applikasjonen. Videre ble MSAM brukt for å forutsi hvorvidt fremtidige brukere vil akseptere og bruke applikasjonen.
- Retningslinje nummer 4: Forskningen må sørge for å gi klare og verifiserbare bidrag. Dette kan man forsikre gjennom resultater av akseptansetestingen.
- Retningslinje 5: Strenge fremgangsmåter for hvordan forskningen er blitt gjennomført, både med tanke på utviklingen og evalueringen av artifaktet. Videre i dette kapittelet vil de ulike evaluerings - og utviklingsmetodene som blir brukt i denne masteroppgaven bli forklart.
- Retningslinje 6: Design er en søkeprosess for å finne en effektiv løsning på et problem. Problemløsning kan dermed bli sett på som å benytte seg av

tilgjengelige ressurser for å nå ønskede målsetninger. I denne masteroppgaven har det blitt benyttet en iterativ prosess ved å først identifisere et problem og dermed komme fram til en løsning ved å benytte seg av Peffers DSR-modell, se figur 6.

- Retningslinje nummer 7: Design science research må bli formidlet til både et teknologi - og ledelsesorientert publikum. Denne masteroppgaven vil være tilgjengelig i form av en rapport på Norges teknisk- og naturvitenskapelige universitet.

**Tabell 2:** Retningslinjer for DSR

| Retningslinje                                 | Beskrivelse  |
|---|--|
| Retningslinje 1: Designe en artifakt          | DSR må produsere en artifakt i form av en konstruksjon, en modell, en metode eller en instans.                 |
| Retningslinje 2: Problem relevanse            | Målsetningen med DSR er å utvikle teknologibaserte løsninger som løser viktige og relevante problemer.         |
| Retningslinje 3: Designevaluering             | Nytten, kvaliteten og effekten av en artifakt må bli evaluert gjennom velkjente og gode evalueringmetoder.     |
| Retningslinje 4: Forskningsbidrag             | Effektiv DSR må gi klare og verifiserbare bidrag.  |
| Retningslinje 5: Grundig forskning            | DSR er avhengig av strenge metoder for både evaluering og utvikling av artifaktet.                             |
| Retningslinje 6: Design som en søkeprosess    | Søken etter en effektiv artifakt er avhengig av at en utnytter alle tilgjengelige midler for å nå ønskede mål. |
| Retningslinje 7: Kommunikasjon av forskningen | DSR må presenteres til både et teknologi- og organisasjonsbasert publikum.                                     |

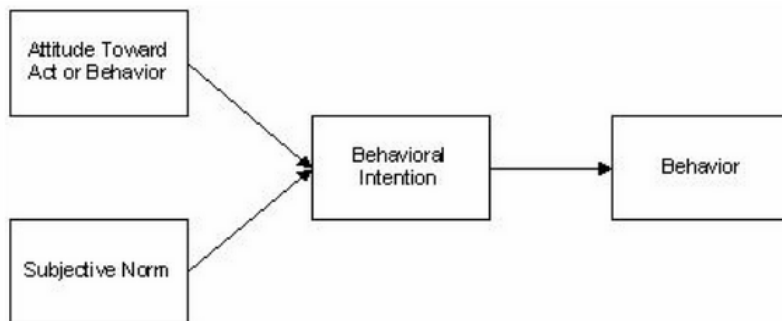
### 3.3 Akseptansemodeller

Det andre forskningsspørsmålet som ble stilt i starten av denne rapporten var hvorvidt en kunne forutsi brukerens aksept for applikasjonen som blir utviklet i denne masteroppgaven. Ved å følge en akseptansemodell som er blitt spesielt utviklet for bruk med mobile applikasjoner, kan en oppnå et bedre innblikk i om applikasjonen har den nytteverdien som er nødvendig for at den skal løse problemet den blir utviklet for. Videre i dette kapittelet vil enkelte eksisterende teorier knyttet til dette temaet bli nærmere forklart.



### 3.3.1 Theory of Reasoned Action

The theory of Reasoned Action (RA) ble utviklet av Martin Fishbein og Icek Ajzen som en forbedring i forhold til Information Integration-teorien [3]. Hovedsaklig er det to fundamentale forskjeller mellom de to teoriene. RA legger til et nytt element i overtalelseprosessen; atferdsmessig intensjon. Istedenfor å forutsi holdninger, forsøker RA eksplisitt å fokusere på atferd. Videre gjenkjenner også RA at det finnes situasjoner som begrenser i hvilken grad holdning har en påvirkning på atferden. Et eksempel på dette er hvis holdningen vår ønsker at vi skal gå på kino, men vi har ingen penger, vil vår mangel på penger endre vår holdning og hindre oss ifra å gå på kino. RA spår derfor atferdsintensjon, som er et kompromiss mellom å spå holdninger og faktisk forutsi atferd. Den andre endringen er at RA bruker ytterligere to elementer for å forutsi atferdsintensjon; holdninger og normer. Det vil si at våre holdninger ønsker at vi skal gjøre en ting, men holdningen våre kan bli endret på bakgrunn av den generelle normen. Et eksempel på dette er at Per sine holdninger oppmuntrer han til å se den nye Disney filmen på kino, men vennene hans syns denne filmen er barnslig. Dermed kan Per sine holdninger mot å se denne filmen endres. Spesifikt spår RA at atferdshensikt blir kontrollert av to faktorer; holdninger og subjektive normer, se figur 7.

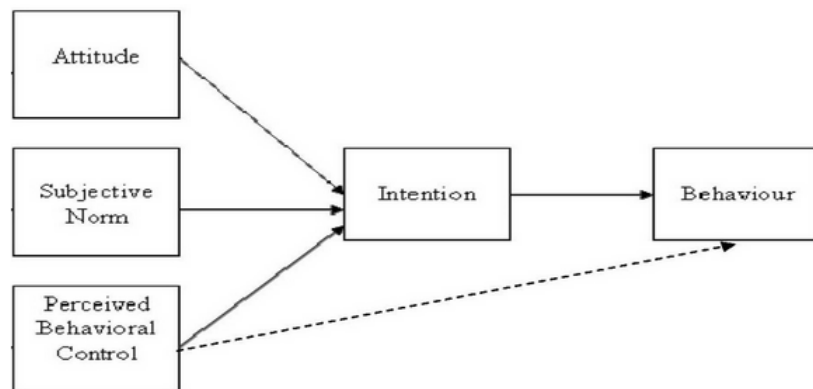


Figur 7: Theory of Reasoned Action

### 3.3.2 Theory of Planned Behavior

The theory of Planned Behavior (TPB) er en teori som knytter tro og atferd, og er en forlengelse av TRA. Teorien forteller at holdningen til en atferd, subjektive normer og oppfattet atferdskontroll former et individ sin atferdsintensjon og dermed dens atferd, se figur 8. Atferdskontroll refererer til en persons oppfatning av hvor enkelt eller vanskelig det er å utføre en ønskelig atferd. På bak-

grunn av dette vil atferdskontrollen til et individ variere på tvers av situasjoner og handlinger, noe som resulterer i at en person vil ha varierende oppfatning av atferdskontrollen avhengig av situasjonen vedkommende befinner seg i [22].



**Figur 8:** Theory of Planned Behavior

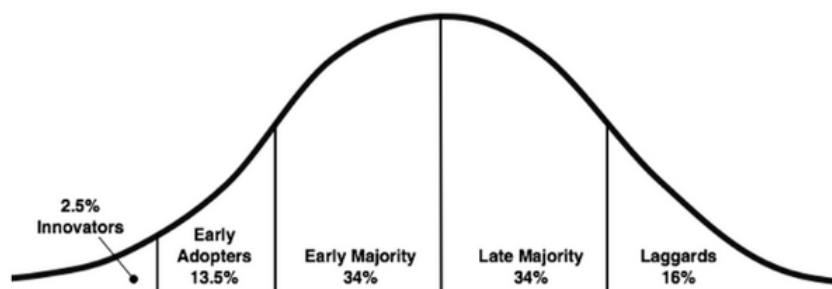
### 3.3.3 Diffusion of Innovation Theory

Diffusion of Innovation Theory (DOI) ble utviklet av EM Rogers i 1962 og prøver å forklare hvordan, over tid, en ide eller et produkt kan oppnå momentum og spre seg gjennom en bestemt populasjon eller et sosialt system. Resultatet av denne spredningen er at individer i populasjonen adopterer bruken av den nye ideen eller produktet. Hovedgrunnen til adopsjon er at individet må oppfatte ideen eller produktet som nytt eller nyskapende. Det er gjennom dette at diffusjon er mulig [21].

Adopsjon av en ny ide eller et produkt skjer ikke samtidig i en populasjon. Forskere har funnet ut at personer som adopterer en innovasjon tidlig har ulike egenskaper i forhold til personer som adopterer en innovasjon senere. Det eksisterer fem forskjellige adopsjonskategorier som individer fra en populasjon kan befinne seg i når det kommer til adopsjon av en ny ide eller et nytt produkt. Selv om flertallet av befolkningen vil falle i en av de midterste kategoriene, se figur 9, er det likevel viktig å forstå karakteristikene til målgruppene med tanke på ulike strategier som kan brukes for å appellere til de ulike adopsjonskategoriene.

- **Innovators:** Personer som ønsker å være de første til å prøve en innovasjon. Disse individene er villige til å ta sjanser og er ofte de første til å adaptere nye ideer. Veldig lite behøves å gjøre for å appellere til denne populasjonen.

- Early adopters: Mennesker som representerer opinionsledere og har et avslappet forhold til å ta i bruk nye ideer. Strategier for å appellere til denne populasjonen er gjerne bruksanvisninger.
- Early Majority: De er sjeldent ledere, men de adopterer nye ideer tidligere enn gjennomsnittet. De er ofte avhengig av bevis på at en innovasjon fungerer før de er villige til å vedta den. Strategier for å appellere til denne gruppen er blant annet suksesshistorier vedrørende innovasjonens effektivitet.
- Late Majority: De er skeptiske til forandringer og vil bare vedta en innovasjon etter at den har blitt prøvd ut av flertallet. Strategier for å appellere til denne populasjonen er gjerne informasjon om hvor mange som har vedtatt innovasjon med suksess.
- Laggards: Disse menneskene er bundet av tradisjon, svært konservative og veldig skeptisk til endringer. Strategier for å appellere til denne gruppen omfatter statistikk og press ifra folk i de andre adopsjonsgruppene.



**Figur 9:** Adopsjonskategoriene

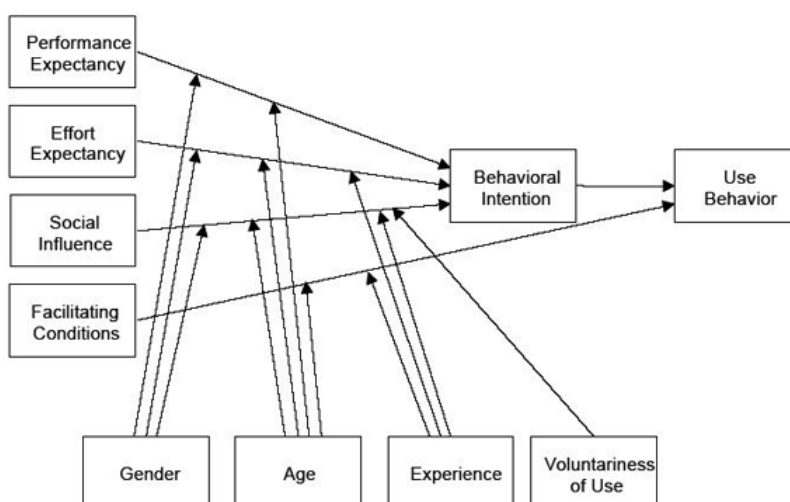
Videre er det fem hovedfaktorer som påvirker adopsjon av en innovasjon og hver av disse faktorene spiller en ulik rolle i de ulike adopsjonskategoriene:

- Relativ fordel: I hvilken grad en innovasjon blir sett på som bedre enn ideen eller produktet den erstatter.
- Kompabilitet: Hvor konsekvent innovasjonen er med tanke på verdiene, erfaringene og behovene til de potensielle brukerne som skal adoptere innovasjonen.
- Kompleksitet: Hvor vanskelig innovasjonen er å forstå og/eller bruke.

- Testbarhet: I hvilken grad innovasjonen kan testes og eksperimenteres med før en forpliktelse til adopsjonen blir gjort.
- Observerbarhet - I hvilken grad innovasjonen gir konkrete resultater.

### 3.3.4 Unified Theory of Acceptance and Use of Technology

Unified Theory of Acceptance and Use of Technology (UTAUT) er en annen modell som forsøker å forklare en bruker sin intensjon til å bruke et informasjonssystem [2]. UTAUT hevder at det er fire sentrale elementer som er direkte determinanter for brukerintensjon og atferd: forventet ytelse, forventet innsats, sosial innflytelse og tilrettelagte forhold. Videre blir kjønn, alder, erfaring og frivillighet av bruk brukt for å moderere effekten av de fire sentrale elementene som beskrevet i figuren 10. Teorien ble utviklet gjennom en konsolidering av åtte ulike modeller som tidligere forskning har konkludert med forklarer brukeratferden med tanke på informasjonssystemer.



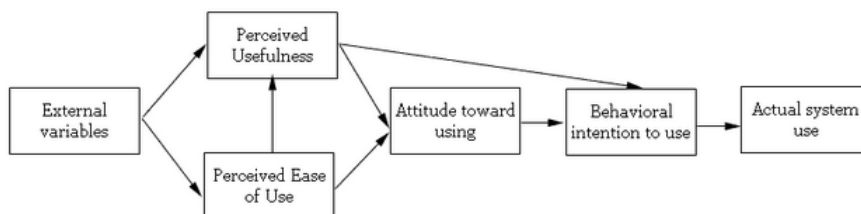
Figur 10: Unified Theory of Acceptance and Use of Technology

### 3.3.5 Technology Acceptance Model

Technology Acceptance Model (TAM) er en modell som spesifikt fokuserer på å prediktere akseptanse av et informasjonssystem. Hensikten med modellen er å forutsi akseptansegraden på et informasjonssystem, for deretter å identifisere mulige modifikasjoner som kan bli gjort for at brukere skal akseptere systemet [20].

Modellen beskriver to hovedfaktorer som er avgjørende for aksepten av et informasjonssystem; opplevd nytte (Perceived Usefulness) og opplevd brukervennlighet (Perceived Ease of Use). Perceived Usefulness (PU) defineres som graden av hvor mye et individ tror at bruken av et system kan forbedre vedkommende sin ytelse ved utførelse av en bestemt oppgave. Perceived Ease of Use (PEOU) referer til i hvilken grad en person tror at bruken av et system vil være enkelt og uanstrengende.

TAM beskriver at bruken av et informasjonssystem bestemmes av atferdsintensjon, men også at atferdsintensjonen er bestemt av personens holdning til systemet sammen med vedkommende sin oppfatning av nytteverdien til nevnte system. Dermed vil ikke holdningen til en person være den eneste faktoren som avgjør vedkommende sitt bruk av et system, men også hvorvidt systemet har en innvirkning på denne personen sine prestasjoner. Selv om en person er imot et informasjonssystem, er sannsynligheten høy for at han vil bruke systemet om det forbedrer ytelsen hans på jobb. TAM forteller også om en direkte kobling mellom PEOU og PU. Om to systemer tilbyr de samme funksjonene, vil en bruker benytte seg av systemet som vedkommende finner enklest å bruke, se figur 11.



**Figur 11:** Technology Acceptance Model

### 3.3.6 Mobile Services Acceptance Model

I dette kapittelet vil en akseptansmodell kalt Mobile Services Acceptance Model (MSAM) bli nærmere forklart. Denne modellen er blitt utviklet for den type artfakt som denne masteroppgaven skal resultere i og er mer passende enn de tidligere modellene som er blitt forklart i kapittel 3.3. Modellen er basert på TAM (kapittel 3.3.5) og er utvidet med ytterligere tre faktorer; kontekst, tillit og personlige initiativ og egenskaper [27], i tillegg til PEOU, PU (beskrevet i kapittel 3.3.5) og brukerintensjon til å bruke en applikasjon.

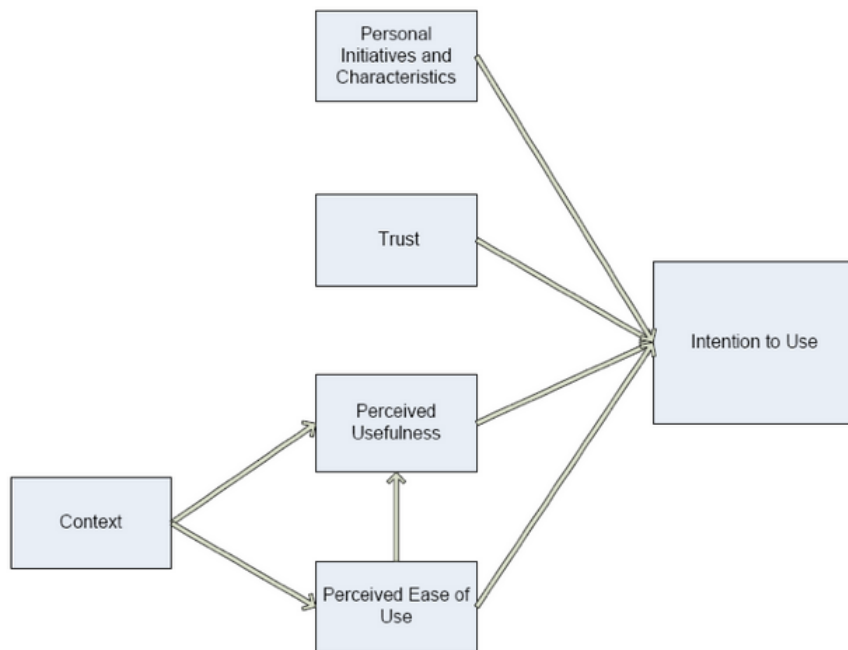
I denne sammenhengen blir kontekst definert som all informasjon som kan brukes til å karakterisere situasjonen for enheter (et sted, en person eller et objekt)

som anses relevante for samspillet mellom en bruker og en applikasjon. Kontekst omfatter vanligvis plassering, identitet og tilstanden av personer, grupper og objekter. Med tanke på mobile applikasjoner kan denne konteksten klassifiseres i to ulike kategorier: kontekst sentrert rundt personer (People Centered Context) og kontekst sentrert rundt plassering (Place Centered Context). Konteksten sentrert rundt plassering refererer blant annet til lokasjonen personer befinner seg i og hvilke ressurser denne lokasjonen besitter, mens den persons sentrerte konteksten refererer til statusen til brukeren, brukerens intensjoner, informasjonen brukeren er i besittelse av, etc.

Forståelsen for den mobile konteksten er også viktig med tanke på design av mobile tjenester. Basert på kontekstinformasjon kan en bruker bestemme om en mobil applikasjon er nyttig eller enkel å bruke. I MSAM foreslås det at konteksten påvirker PEOU og PU direkte.

Personlige initiativ kan defineres som brukerens vilje til å prøve ut nye applikasjoner. Personlige egenskaper inkluderer: alder, kjønn, utdanningsbakgrunn, kunnskap og ferdigheter, kultur og preferanser. En bruker sin vilje og behov spiller også en viktig rolle når det kommer til adopsjon av mobile tjenester. Ulike brukergrupper kan ha ulike perspektiver på en mobil tjeneste (beskrevet i kapittel 3.3.3). På bakgrunn av dette foreslås det i denne modellen at personlige initiativ og egenskaper påvirker brukerintensjonen direkte.

Med tanke på mobile tjenester, kan tillit defineres som brukerens tro på i hvor stor grad en mobilapplikasjon kan ha sikkerhets- og personvernstrusler. For leverandører av mobile applikasjoner er det å tilegne seg forbrukeren sin tillit en tidskrevende oppgave. Det er flere faktorer som er med på å påvirke forbrukere sin tillit når det kommer til kommersielle mobile applikasjoner. Merkevarer og ryktet til selskapet som utgir applikasjonen kan bidra til å enten styrke eller svekke tillit. Som et resultat av dette vil tillit påvirke brukerens intensjon om å bruke en mobile applikasjon, se figur 12.



**Figur 12:** Mobile Services Acceptance Model

### 3.4 Brukervennlighet

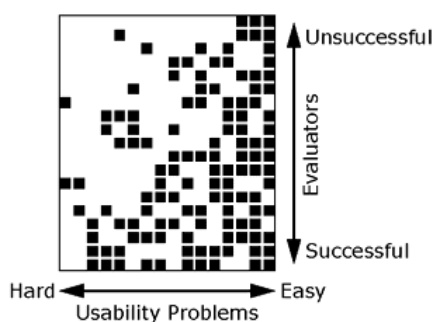
Denne avhandlingens tredje forskningsspørsmål spør om hvordan en kan utvikle applikasjonen og presentere informasjonen på en slik måte at brukervennligheten blir høy. Det eksisterer flere teknikker og retningslinjer for hvordan en skal forholde seg til utvikling av brukergrensesnitt for å forsøke å opprettholde høy brukervennlighet, men denne masteroppgaven skal forholde seg til en heuristisk evaluering for å avdekke svakheter og styrker ved brukergrensesnittet.

#### 3.4.1 Heuristisk Evaluering

Heuristisk evaluering (HE) er en metode for å avdekke problemer i et brukergrensesnitt, slik at de kan bli ivaretatt som en del av en iterativ designprosess. Metoden innebærer å ha et begrenset sett med evaluatorene bedømme et grensesnitt og vurdere hvorvidt det er noe samsvar med kjente prinsipper innenfor brukbarhet [18].

Generelt vil det være utfordrende for en person å gjøre en HE alene, da det ikke er gitt at denne personen vil finne alle problemene i et grensesnitt. Erfaring fra ulike prosjekter har vist at ulike personer finner ulike problemer og dermed er det mulig å øke effektiviteten av evalueringen ved å involvere flere evaluatorene.

Figur 13 viser en HE hvor 19 ulike personer ble brukt for å finne 16 ulike brukervennlighetsproblemer i et system. Hver av de sorte kvadratene representerer et funn av et brukervennlighetsproblem. Figuren viser også at det eksisterer en del ikke-overlapp over hvilke problemer de ulike evaluatorene har funnet i systemet. Enkelte problemer er enkle å avdekke, mens noen av de vanskelige problemene er kun avdekket av noen få evaluatorene. Dermed er det ikke nødvendigvis slik at den samme personen er den beste evaluatoren hver gang, da noen av brukervennlighetsproblemene som var mest utfordrende å finne ble funnet av evaluatorene som ellers ikke fant mange problemer. På bakgrunn av dette er det nødvendig å involvere flere evaluatorene i en heuristisk evaluering og det anbefales at dette tallet befinner seg et sted mellom tre og fem da en ikke vil få så mye mer informasjon ved å bruke et større sett med evaluatorene.



**Figur 13:** Heuristisk evaluering

Heuristisk evaluering utføres ved at hver enkelt evaluator inspiserer brukergrensesnittet alene. Etter at alle evalueringene er gjennomført har evaluatorene lov til å kommunisere, samtidig som alle funnene blir samlet sammen. Dette sikrer også uavhengige og objektive evalueringer fra hver evaluator. Resultatene av evalueringene kan registreres enten som skriftlige rapporter fra hver evaluator eller ved å ha evaluatorene muntlig videreføre resultatene til en observatør samtidig som de går gjennom brukergrensesnittet.

Forskjeller mellom heuristisk evaluering og tradisjonell brukertesting er hvorvidt observatøren er villig til å svare på spørsmål fra evaluatorene under testøkten og i hvilken grad evaluatorene kan få hint om hvordan de bruker grensesnittet. I tradisjonell brukertesting ønsker en å oppdage feilene en bruker gjør når vedkommende interagerer med grensesnittet. Dermed er observatørene svært restriktive med tanke på å gi noe mer hjelp til evaluatorene enn hva som er absolutt nød-



vendig. Dette på grunn av at evaluatorene skal finne svar på spørsmålene sine ved å bruke grensesnittet fremfor å få spørsmålene besvart av en observatør. I en heuristisk evaluering av en domenespesifikk applikasjon vil det være urimelig å nekte å svare evaluatorene på spørsmål knyttet til dette domenet, spesielt om det er evaluatører som har svært lite kjennskap til domenet. Ved å besvare spørsmålene vil dette føre til at evaluatorene er bedre rustet til å vurdere brukbarheten av grensesnittet med hensyn til de ulike egenskapene til domenet. Det er viktig å merke seg at evaluatører ikke bør gis hjelp før de virkelig sitter fast i brukergrensesnittet og har kommentert om hva som er problemet med brukervennligheten i akkurat det tilfellet.

Lengden på en heuristisk evaluering for en evaluator varer vanligvis mellom en eller to timer. Det vil være nødvendig med lengre økter for større og mer kompliserte grensesnitt, men da er det en fordel å dele evalueringen i flere økter og fokusere på forskjellige deler av grensesnittet for hver økt. Under evalueringsøkten vil evaluatoren gå gjennom grensesnittet flere ganger og vurdere de ulike elementene i forhold til en liste med ulike prinsipper for god brukbarhet (heuristikker). I tillegg til denne sjekklisten med heuristikker har evaluatoren også tillatelse til å vurdere ulike grensesnittelementer utifra ytterligere brukervennlighetsprinsipper som vedkommende mener er aktuelt for et bestemt element.

I prinsippet så kan evaluatører bestemme på egen hånd hvordan de ønsker å gå frem når de skal evaluere grensesnittet, men en generell anbefaling vil være at de går gjennom det minst to ganger. Den første gangen skal brukes for å få en følelse av flyten i grensesnittet sammen med det generelle omfanget av applikasjonen. Den andre tillater da evaluatoren å fokusere på spesifikke elementer, samtidig som de vet hvordan disse elementene passer inn i den store sammenhengen. Siden evaluatorene ikke bruker systemet for å utføre en reel oppgave, er det mulig å utføre en heuristisk evaluering av grensesnitt på en papirprototype. Dette gjør en heuristisk evaluering egnet for bruk tidlig i utviklingsfasen.

Resultatene av den heuristiske evalueringen er en liste over ulike problemer i brukergrensesnittet med referanser til de heuristikkene som blir brutt. Evaluatorene bør være så spesifikke som overhodet mulig og liste hvert eneste brukervennlighetsproblem. Om det er flere ting som er galt med et element, skal alle tre noteres med henvisning til heuristikkene og hvorfor de er et brukervennlighetsproblem. Det er to grunner til det; for det første er det en viss risiko for å repetere andre brukervennlighetsproblemer når en endrer på designet til et element om en

ikke er klar over alle problemene til elementet fra starten av. For det andre er det ikke nødvendigvis mulig å løse alle brukervennlighetsproblemene til et element, men en kan da løse noen av de som allerede er kjent.

Nielsens ti prinsipper for brukervennlighet som blir brukt i en heuristisk evaluering:

- Synlighet av systemstatus: Systemet skal til enhver tid holde brukeren informert om dens status gjennom feedback innenfor et hensiktsmessig tidsrom.
- Sammenheng mellom systemet og den virkelige verdenen: Systemet skal snakke brukeren sitt språk gjennom ord, fraser og konsepter som er kjent for brukeren, istedenfor å bruke systemorientert terminologi. Følg virkelige konvensjoner og sørg for at informasjon blir presentert i en naturlig og logisk rekkefølge.
- Brukerkontroll og frihet: Brukere velger ofte systemfunksjoner ved en feil og vil være avhengig av en tydelig merket "nødutgang", slik at de kan forlate en uønsket systemtilstand. Støtt angrefunksjoner.
- Konsistens og standarder: Brukere skal ikke behøve å lure på om forskjellige ord, situasjoner eller handlinger betyr det samme. Følg plattformkonvensjoner.
- Forebygge feil: Det er bedre med et design som hindrer at problemer oppstår til å starte med enn gode feilmeldinger. Eliminer situasjoner som er svært utsatt for feil eller presenter brukere med en bekreftelsesdialog før de forplikter seg til en handling.
- Gjenkjennelse fremfor hukommelse: Sørg for å synliggjør objekter, handlinger og alternativer for brukeren. Vedkommende skal ikke behøve å huske informasjon fra en del av grensesnittet til et annet. Instruksjoner for bruk av systemet skal være synlig og lett å hente fram når det er nødvendig.
- Fleksibelt og effektivt i bruk: Akseleratorer kan ofte fremskynde samhandlingen for ekspertbrukere, slik at systemet kan tilfredstille behovene for både erfarne og uerfarne brukere. Tillat brukere å skreddersy handlinger de bruker ofte.
- Estetisk og minimalistisk design: Dialoger skal ikke inneholde informasjon som er irrelevant eller som er sjeldent nødvendig. Hver ekstra enhet med

urelevant informasjon konkurrerer med de mer relevante enhetene med informasjon og reduserer deres synlighet.

- Hjelp brukerne å gjenkjenne, diagnostisere og løse feil. Feilmeldinger bør uttrykkes i vanlig språk, presist indikere hva problemet er og foreslå en løsning.
- Hjelp og dokumentasjon: Selv om det er bedre dersom systemet kan brukes uten noen form for dokumentasjon, kan dokumentasjon og hjelp være nødvendig. Dokumentasjon og hjelp skal være enkel å søke i, fokusere på brukers oppgaver, liste konkrete steg som skal utføres og ikke være for stor.

### **3.4.2 System Usability Scale**

System Usability Scale (SUS) er en enkel, ti-punkts skala som gir en oversikt over subjektive vurderinger av brukervennligheten til et system [28]. Metoden ble opprinnelig utviklet av John Brooke i 1986 som et verktøy for å måle graden av brukervennlighet. Skalaen metoden benytter seg av består av ti spørsmål med fem ulike svaralternativer som spenner seg fra sterkt uenig (1) til helt enig (5). SUS skiller effektivt mellom brukbare og ubrukelige systemer og gir pålitelige resultater. Resultatet av en SUS-test er en score mellom 0 - 100. Det er viktig å ikke tolke denne skalaen som prosenter, men å normalisere resultatet og skape en persentil rangering. Hvert spørsmål gir en sum og summene fra hvert spørsmål skal summeres, før de deretter skal multipliseres med 2,5 for å få den endelige poengsummen. En poengsum under 68 er under gjennomsnittet og over 68 er over gjennomsnittet god brukervennlighet. SUS brukes også som et evalueringsverktøy i ulike utviklingsprosjekter.

## **3.5 Metoder for innhenting av data**

For å samle inn data til å svare på forskningsspørsmålene og evaluere applikasjonen er det blitt benyttet flere forskjellige innsamlingsteknikker. Nettundersøkelser, tilbakemeldings-, SUS- og MSAM-skjemaer er alle metoder som ble benyttet for å samle informasjon. Denne seksjonen presenterer innsamlingsmetodene som ble benyttet.

### **3.5.1 Nettundersøkelse**

En nettundersøkelse er en av de mest kosteffektive metodene for innsamling av data. Metoden kan tilby både kvantitativ og kvalitativ informasjon, avhengig av

spørsmålene som blir stilt. Nettundersøkelser er lite tidskrevende, da respondentene får en nettadresse som gir de tilgang til undersøkelsen. Respondenten kan derfor svare på undersøkelsen når det passer, uten å måtte være avhengig av en som stiller spørsmålene. Muligheten til å forholde seg anonym er også en positiv ting med nettundersøkelser.

Selv om nettundersøkelser er både tids- og kosteffektivt er det ikke alltid den mest pålitelige kilden til informasjon. Respondentene kan ha misforstått spørsmål eller svarer useriøst på undersøkelsen. Det kan også være vanskelig å finne riktig testgruppe som kan medføre i mange feilaktige svar.

I dette studiet ble det gjennomført en nettundersøkelse for å finne interessen for en applikasjon som gjør det mulig for en bruker å lage/delta på uformelle arrangement. Nettundersøkelsen ble også brukt for å få tilbakemelding på designet til applikasjonen.

### **3.5.2 Spørreundersøkelse**

En spørreundersøkelse er nesten det samme som en nettundersøkelse, se kapittel 3.5.1. Den største forskjellen er at en representant fra prosjektet er tilstede og kan derfor se hvem som svarer. Respondenten er derfor ikke anonym lenger. Dette kan medføre at respondenten svarer mer positivt på spørsmålene enn hva han/hun hadde gjort hvis ikke utvikleren var til stede. En positiv ting med å ha representanten fra prosjektet til stede når respondenten svarer på spørsmålene er at uklare spørsmål kan bli forklart av utvikleren. På den måten vil ikke så mange respondenter misforstå spørsmålene.

I dette studiet ble det gjennomført to spørreundersøkelser, SUS og MSAM.

### **3.5.3 Observasjon**

Observasjon er en viktig metode for å samle inn informasjon på. Bruk av metoden gir kvalitativ informasjon som kan brukes for å beskrive respondentenes oppførsel under utførelse av oppgaver og spørsmål som er gitt. Det er viktig å ikke avbryte, veilede eller spørre spørsmål til deltakeren under testen, da dette kan endre deltakerens oppførsel. Bruk av metoden sørger for mer innsikt i respondentens opplevelse. En ulempe med metoden er at en representant fra prosjektet må være tilstede under hele undersøkelsen for å observere og dette kan bidra til å gjøre metoden svært tidkrevende.

I dette studiet ble metoden benyttet under brukergrensesnitt - og akseptansetestin-

gen for å observere hvordan deltakerne reagerte på designet og hvilke funksjoner som var enkle eller vanskelige å bruke.

#### **3.5.4 Intervju**

Intervju er også en kvalitativ metode som gir en bredere og dypere kunnskap om brukerens oppfatning av emnet. Det finnes tre ulike typer intervju [19]:

- Strukturert
- Delvis strukturert
- Ustrukturert

I et strukturert intervju blir et sett med spørsmål presentert til respondenten. Spørsmålene er satt i forkant av intervjuet og alle respondentene får de samme spørsmålene.

I et delvis strukturert intervju er det mulig å legge til og endre rekkefølgen på spørsmålene.

Ustrukturerte intervju gir intervjuobjektene friheten til å snakke fritt rundt et tema som er blitt presentert av intervjueren.

Strukturerte intervju blir ofte brukt for å sjekke informasjon, mens delvis strukturerte og ustrukturerte intervju er gode metoder for å tilegne seg ny informasjon om et emne.

I dette studiet ble både ustrukturerte og delvis strukturerte intervjuer benyttet. Det ble holdt to ustrukturerte møter med Mental Helse Trondheim for å snakke fritt rundt konseptet. Delvis strukturerte intervju ble benyttet under testing av brukergrensesnitt. Dette ga intervjueren mulighet til å tilpasse seg personen som ble intervjuet og miljøet.

#### **3.5.5 Prototype testing**

Prototype-testing er en mye brukt metode i programvareutvikling for å få tilbakemeldinger på brukernes erfaringer av systemet. Metoden blir brukt for å få tilbakemelding på design, brukervennlighet og funksjonalitet. Deltakerne kan komme med forslag til funksjonalitet som bør implementeres, funksjonalitet som bør endres og forbedringer i brukergrensesnittet.

I dette studiet ble det utført to prototype-tester, en for brukergrensesnittet og en for akseptansetesting. Testene ble utført på forskjellige typer mennesker for å dekke de mest relevante sluttbrukerne.

## 4 Problem

Dette kapitlet tar sikte på å gi en bedre forståelse av problemet som skal løses gjennom denne masteroppgaven. Videre blir personas og ulike scenarioer benyttet for å gi mer innsikt i problemet.

### 4.1 Personas

Personas er en metode som benyttes for å gjøre seg kjent med eller presentere en gruppe brukere på en menneskelig måte. Personas er ikke ekte individer, men oppdiktete portretter av aktuelle personer som ses på som en aktuell brukergruppe for et system som er under utvikling. Bruken av personas kan benyttes i forbindelse med kravspesifikasjoner, utvikling og testing av blant annet IKT-baserte tjenester.

#### 4.1.1 Mental helse, Jacob

**Alder:** 29 år

**Bosted:** Trondheim

**Interesser:** Hund, friluftsliv, trening og ungdom

**Kjerneverdier:** "Det er aldri så vondt at det ikke er godt for noe"

Jacob jobber hos mental helse i Trondheim med ungdom som er ensomme. Han blir beskrevet som en trivelig og omsorgsfull person av sine nærmeste venner. Jacob er veldig glad i dyr og er en stolt eier av en Staffordshire Bull Terrier som heter Fido. Av ulike aktiviteter som fyller hverdagen hans er sjakk, styrketrening og friluftsliv de som er mest fremtredende.

Jacob har mange venner, men få av vennene hans deler samme interesse for friluftsliv. Derfor bli det ofte mange turer alene i skogen med hunden Fido.

Jacob er ikke veldig teknisk interessert. Han har en gammel iPhone 4 og bruker mye Facebook til å oppdatere medlemmene i Mental Helse om arrangementer som er blitt planlagt. Utover det bruker Jacob kun mobilen til å ringe og sende meldinger med.



**Figur 14:** Jacob, 29 år

#### 4.1.2 Student ved Gløshaugen, Jackueline

**Alder:** 24 år

**Bosted:** Trondheim

**Interesser:** Venner, film og data

**Kjerneverdier:** "Ingenting blir bedre om man ikke jobber for det"

Jackueline er en fransk jente som har kommet til Trondheim for å studere datateknikk på NTNU. Hun flyttet til Norge for to år siden og går i andre klasse. På fritiden liker Jackueline å være med venner, se film og å sitte på datamaskinen. Da Jackueline kom til Norge syntes hun det var vanskelig å bli kjent med nordmenn. Derfor holdt hun seg til de andre studentene som også hadde flyttet til Norge fra andre land. Jackueline er en sosial jente som liker å bli kjent med andre personer og syntes derfor det er trist at nordmenn er så vanskelig å bli kjent med.

Jackueline liker å holde seg oppdatert på den nyeste teknologien. Hun eier en iPhone 6, iPad Air 2 og er en daglig bruker av Facebook og Instagram. Twitter og LinkedIn er også applikasjoner/nettsider hun ofte bruker.



**Figur 15:** Jackueline, 24 år

#### 4.1.3 Student ved Dragvoll, Ida

**Alder:** 19 år

**Bosted:** Trondheim

**Interesser:** Dataspill, lesing og matte

**Kjerneverdier:** "Stå på for det en tror på, til det motsatte er bevist"

Ida er en student som bruker mye av tiden sin til å spille dataspill, lese bøker og løse matteproblemer. Hun liker å være med andre mennesker, men har ikke funnet noen som deler de samme interessene som henne. Dette har gjort at Ida ikke har fått så mange venner i Trondheim. Ida begynte å spille dataspill på grunn av mangelen på venner. Nå sliter hun med depresjon og har gått til mental helse i Trondheim for hjelp. Hos mental helse deltar Ida på forskjellige aktiviteter som blir arrangert. Hun har blitt kjent med mange nye mennesker gjennom aktivitetene, men har ikke funnet noen som deler de samme interessene.

Ida er interessert i teknologi. Hun bruker mye tid på forum og har konto på både Facebook og Instagram. Ida eier en Samsung Galaxy S5 og laster stadig ned nye applikasjoner.





**Figur 16:** Ida, 19 år

#### **4.1.4 Jobber hos Statoil, Per**

**Alder:** 56 år

**Bosted:** Skien

**Interesser:** Friluftsliv, ski og familie

**Kjerneverdier:** "Ingenting blir bedre om man ikke jobber for det"

Per er en aktiv mann på 56 år som liker å være ute i naturen. Han jobber hos Statoil som plattformsjef. Han blir av de nærmeste rundt seg beskrevet som en morsom og aktiv person. Per har både kone og to barn. Når Per har fri liker han å være ute i naturen. Hans kone Ingrid pendler til Oslo i hverdagen for å dra på jobb og hans barn har flyttet til Trondheim for å studere. Derfor har Per mye alenetid og drar ofte på toppturer.

Per er ikke glad i tekniske komponenter. Han har kjøpt seg både iPhone 6 og en ny MacBook Pro, men sliter med å sette seg inn i teknologien. Per har en konto på Facebook som han bruker til å holde seg oppdatert på hva som skjer i vennekretsen.



**Figur 17:** Per, 56 år

## 4.2 Scenario

Scenarier blir brukt som et verktøy for å beskrive hvordan ulike brukere interagerer med et produkt eller en tjeneste. De bidrar til å skape en bedre forståelse av problemet ved å få frem risikable scenarier og oppdage nye brukerhistorier. Scenariene består av en personas og en relevant historie i relasjon til applikasjonen under utvikling.

### 4.2.1 Scenario 1: Finne studievenner, uten applikasjonen

Studenten Ida er lei av å sitte på hybelen og løse matteproblemer alene. Det nærmer seg eksamenstid og hun vil finne en studievenninne hun kan lese med. Ida lærer best ved å diskutere løsninger og sammen med andre finne svaret på et gitt problem.

Ida kjenner ikke mange av de som går i klassen hennes og de få hun vet hvem er har sagt at de skal være hjemme å studere. Hun går derfor inn på Facebook-siden til linjeforeningen Nabla og skriver en melding til alle medlemmene. I meldingen skriver hun litt om seg selv og forklarer at hun ønsker å møte en person som har lyst til å studere sammen med hun. Etter en uke har hun enda ikke fått noe svar og føler nå at ingen vil være med henne. Hun blir derfor sittende på hybelen sin å studere alene.

### 4.2.2 Scenario 2: Finne studievenner, med applikasjonen

Ida har få kontakter i Trondheim by og føler seg veldig alene. Hun har derfor gått til Mental Helse i Trondheim hvor hun har deltatt på flere arrangement. Nå som

eksamenstiden begynner å nærme seg ønsker hun å finne en studievenninne.

Ida har gjennom Mental Helse hørt om en applikasjon som heter GatherUp hvor mennesker kan finne og lage uformelle arrangement. Hun går inn i applikasjonen og lager et nytt arrangement med tittel “Studietid på Gløshaugen”. I beskrivelsen skriver Ida at hun trenger en studiepartner som skal ta samme eksamen som henne.

Etter en dag blir den franske studenten Jackueline medlem av arrangementet. Hun skriver i chatten at hun går datateknikk, men at hun har samme matte eksamen som Ida. De bestemmer seg for å møtes på skolen for å lære mattefaget sammen. Når Ida og Jackueline møtes på skolen dagen etter forteller Jackueline at hun har hatt problemer med å bli kjent med nordmenn. Jackueline syntes derfor at det var koselig og endelig møte en norsk person som var villig til å snakke engelsk med hun.

Hvis ingen hadde meldt seg på arrangementet ville ikke dette gjort noe. Det på grunn av at ingen kunne vist at det var hun som hadde lagd arrangementet med mindre de ble medlem. Derfor kunne Ida dratt på skolen uten å føle at alle så på henne som en ensom person ingen ville være med.

#### **4.2.3 Scenario 3: Finne personer å være med på fritiden, med applikasjonen**

Jacob jobber med ungdom i Mental Helse. Han arrangerer stadig aktiviteter for ungdom som sliter med depresjon på grunn av ensomhet. For å invitere medlemmer til arrangementene har han tidligere brukt facebook for å lage arrangementer. Nå bruker han applikasjonen GatherUp for å lage arrangementer. I applikasjonen kan medlemmene velge å være anonyme. Dette gjør at toleransen for å melde seg på er mye lavere. Han har også opprettet et arrangement som heter “Kaffe på solsiden” hvor medlemmene møtes og snakker om alt fra studentlivet til familielivet. Han har valgt å bruke applikasjonen slik at alle kan melde seg på og ikke bare de som har tatt kontakt med Mental Helse i Trondheim. Arrangementet “Kaffe på solsiden” blir automatisk oppdatert hver uke med en ny dato. Dette gjør at Jacob slipper å lage nye arrangement hver søndag.

#### **4.2.4 Scenario 4: Finne turkamerat, med applikasjonen**

Per har mye fritid når han ikke er på jobb. Han leste i Teknisk Ukeblad om en ny applikasjon hvor man kunne lage et arrangement eller bli med på arrangementer andre har laget. Arrangementene kunne være alt fra piknik i parken til ekstrem-sport. Per lastet derfor ned applikasjonen og laget et arrangement hvor han søkte

etter andre til å gå på topptur med. Senere den uken fikk Per en varsling på telefonen. På varslingen stod det: “Jacob har blitt med på ditt arrangement”. Per gikk derfor inn i applikasjonen og så at Jacob hadde skrevet til han i chatten. Etter et par dager med planlegging møtes de for å gå på Galdhøpiggen.

### 4.3 Lage test oppgaver

Denne seksjonen viser sammenhengen mellom personas, scenario og oppgavene i SUS og MSAM. Det er vanligvis tre forskjellige type brukere som må tas hensyn til når man lager oppgavene [26].

- Nybegynner eller førstegangsbrukere
- Kunnskapsrike brukere
- Hyppige brukere med mye kunnskap.

I kapittel 4.1 ble fire personas presentert. Jackueline og Ida er hyppige brukere av sosiale nettverk med mye kunnskap innenfor teknologiske enheter. Per er en kunnskapsrik bruker som er interessert i teknologi og kan en del om sosiale nettverk. Jacob er nybegynner og er ikke interessert i tekniske enheter. Han bruker sjeldent sosiale nettverk.

#### 4.3.1 System Usability Scale Test

De ulike oppgavene som skal bli gjennomført i SUS-testen er beskrevet i vedlegg D. Oppgavene er blitt utviklet i henhold til problemet denne masteroppgaven skal løse og gjenspeiler seg i grunnfunksjonaliteten til applikasjonen som er blitt utviklet. Den første oppgaven (“Anta at du ønsker å spille fotball. Finn en aktivitet hvor du kan bli med og spille fotball. Hvor mange deltar på dette arrangementet?”) er laget på bakgrunn av scenario 4. Der har du Jacob som ønsket å bli med på en topptur og brukte dermed applikasjonen sin søkefunksjon for å finne et arrangement som passet hans ønsker.

Den andre oppgaven (“Opprett et arrangement ved navn “Grilling på Kalvskinnet”. Tillatt at begge kjønn får lov til å bli med og maks 30 personer fra alderen 18 år og oppover”) er blitt laget på bakgrunn av scenario 2 hvor Ida ønsker å opprette et arrangement hvor hun er på leting etter en studiekamerat på Gløshaugen. Den siste oppgaven (“Meld deg på et arrangement som heter “Kents Opplevelse” og bli med i chatten. Hva er den siste meldingen som er blitt skrevet i chatten på dette

arrangementet?”) er blitt opprettet med scenario 2 i tankene, når Jackueline skal skrive på chatten til Ida om at de har det samme mattefaget. Dermed kan de bli studiekamerater.

#### **4.3.2 Mobile Service Acceptanse Model Test**

Under akseptansetesten ble det benyttet en Samsung Galaxy S4 Active, Samsung Galaxy S2 eller en Samsung Galaxy Tab 8 med en prototype av programvaren allerede installert på enheten. Mesteparten av testdeltagerne ble funnet rundt omkring på NTNU sitt campus på Gløshaugen og de ble tildelt informasjon rundt applikasjonen slik at de hadde en ide om hovedfunksjonene til appen. Deretter ble testdeltagerne bedt om å fullføre tre forskjellige arbeidsoppgaver som er knyttet til funksjonaliteten til applikasjonen:

- Anta at du ønsker å spille fotball. Finn et arrangement hvor du kan bli med og spille fotball. Hvor mange deltar på dette arrangementet?
- Opprett et arrangement ved navn “Grilling på Kalvskinnet”. Tillatt at begge kjønn får lov til å bli med og maks 30 personer fra alderen 18 år og oppover.
- Meld deg på et arrangementsom heter “Kents opplevelse” og bli med i chatten. Hva er den siste meldingen som er blitt skrevet i chatten på dette arrangementet?

Etter at testdeltagerne var ferdige med testoppgavene ble de tildelt et spørreskjema som de skulle fylle ut. Resultatene av dette spørreskjemaet ble analysert for å vurdere testdeltagerne sin aksept av en slik type mobil applikasjon.

### **4.4 Krav**

Krav er utviklet for å beskrive oppførselen til et spesielt system eller en applikasjon. Kravene skal beskrive all den funksjonaliteten som er forventet at systemet skal inneha og hvordan denne funksjonaliteten skal fungere. Disse kravene gjenspeiler de ulike behovene som brukerne har til et gitt system og blir ofte gitt en prioritet slik at de viktigste funksjonene blir tatt hånd om først under utviklingsfasen.

#### **4.4.1 Bestemme krav**

For å bestemme de ulike kravene til applikasjonen som skulle utvikles ble det gjennomført en kravsanalyse. Enkelte av kravene ble utformet på bakgrunn av

svarene hentet fra spørreundersøkelsen som ble tatt av individer som anses som sluttbrukere av applikasjonen. Ved å stille spørsmålet “Er det flere funksjoner du skulle ønske fra en slik applikasjon?”, ble det avdekket ytterligere funksjonalitet som ble ansett som viktig å implementere og dermed utarbeide relevante krav på bakgrunn av dette. Det ble også samlet inn krav etter flere ustrukturerte intervju med representanter fra Mental Helse i Trondheim, da de sitter på betydningsfull informasjon vedrørende en andel av sluttbrukerne denne applikasjonen ønsker å sikte seg inn på.

De funksjonelle kravene som er blitt utarbeidet har blitt merket med prioriteten høy, middels eller lav, alt ettersom hvor viktige de er for grunntanken i applikasjonen, men også hvorvidt de lar seg implementere med tanke på det begrensede tidsaspektet denne masteroppgaven står ovenfor.

#### 4.4.2 Funksjonelle krav

Funksjonelle krav er konkrete krav som beskriver hva et system skal gjøre, hvordan systemet skal reagere på ulike inputs eller hvordan et system skal reagere i forskjellige situasjoner. Tabell 3 inneholder de funksjonelle kravene som er blitt utarbeidet:

**Tabell 3:** Funksjonelle krav

| ID  | Beskrivelse   | Prioritet |
|-----|---|-----------|
| FK1 | Applikasjonen må støtte innlogging med Facebook   | Høy       |
| FK2 | Applikasjonen må vise en liste med alle tilgjengelige arrangement   | Høy       |
| FK3 | Det må være mulig for en bruker å melde seg på og melde seg av et gitt arrangement                                    | Høy       |
| FK4 | Det må være mulig for en bruker å lage et arrangement som andre brukere kan melde seg på                              | Høy       |
| FK5 | Det må være mulig å kommunisere med andre deltakere på et bestemt arrangement   | Høy       |
| FK6 | Det bør være mulig for en bruker å endre profilbildet sitt.   | Middels   |
| FK7 | Brukeren bør få varsling på telefonen hvis et arrangement han/hun er interessert i blir opprettet.                    | Middels   |
| FK8 | Det bør være mulig å finne arrangement kun i nærheten av en bruker sin lokasjon.                                      | Middels   |
| FK9 | Det bør være mulig for brukere å rapportere brukere som melder seg på et arrangement for så å ikke dukke opp likevel. | Lav       |

### 4.4.3 Ikke funksjonelle krav

Ikke-funksjonelle krav spesifiserer oppførselen til et system og dekker de resterende områdene av applikasjonen som de funksjonelle kravene ikke dekker. Tabell 4 inneholder de ikke-funksjonelle kravene som er blitt utarbeidet:

**Tabell 4:** Ikke funksjonelle krav

| ID   | Beskrivelse   | Prioritet |
|------|---|-----------|
| IFK1 | Applikasjonen må være brukervennlig og ha et SUS resultat på over 68        | Høy       |
| IFK2 | Data til applikasjonen må være tilgjengelig 95% av tiden                    | Høy       |
| IFK3 | Brukeren skal ikke vente mer enn 2 sekunder på en respons fra applikasjonen | Høy       |

## 4.5 Use Case

Use Case blir brukt for å analysere, identifisere og organisere funksjonelle krav. De funksjonelle kravene blir bestemt ved å se på flyten og forholdet mellom de forskjellige brukerne og use casene. Dette blir illustrert i UML-diagram [14]. Det er tre elementer som utgjør en use case; aktører, system og mål [11].

Use-Case beskrivelsene bygger på UML-diagrammene i figur 18 og 19. Figur 18 viser alle use casene første gang en bruker åpner applikasjonen, mens figur 19 viser alle use casene til applikasjonen når brukeren allerede har logget inn. De tekstlige use case elementene er tatt fra Derek Coleman's template [4] og inneholder:

**Use Case:** Use Case identifikator og navn

**Beskrivelse** Kort beskrivelse av mål og hensikten med use casen

**Aktører:** Liste av aktører som er involvert i use caset

**Forutsetninger:** Vilkår som må være oppfylt for at use caset skal være vellykket

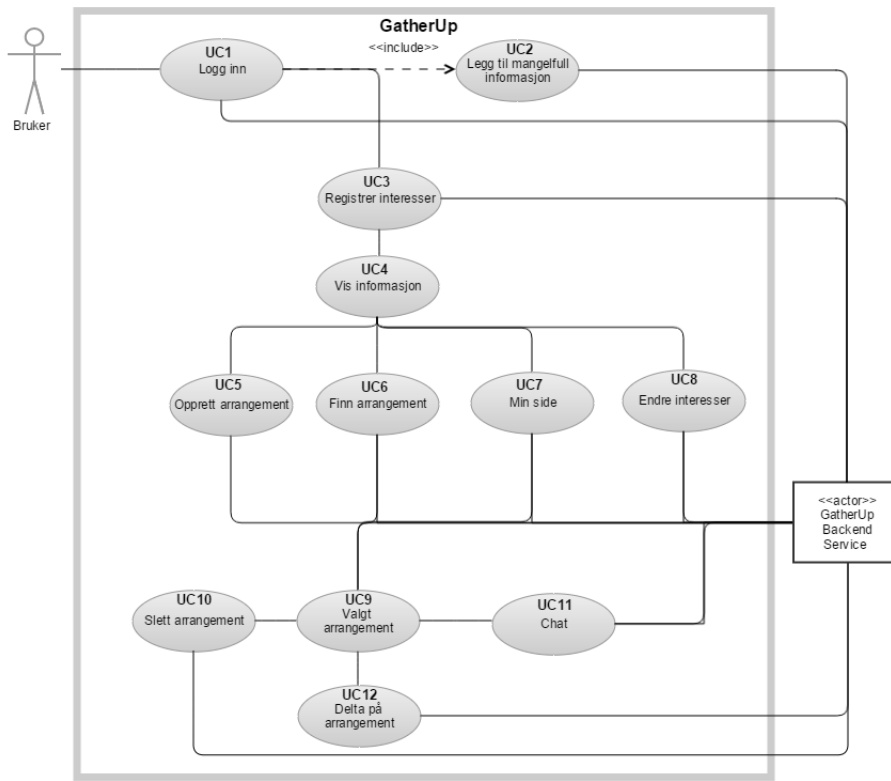
**Steg:** Interaksjon mellom aktøren og systemet som er nødvendig for å nå målet

**Variasjoner:** (*Valgfri*) Alle variasjoner i stegene til en use case

**Ikke funksjonelle krav:** (*Valgfri*) En liste av ikke funksjonelle krav som use caset må oppnå

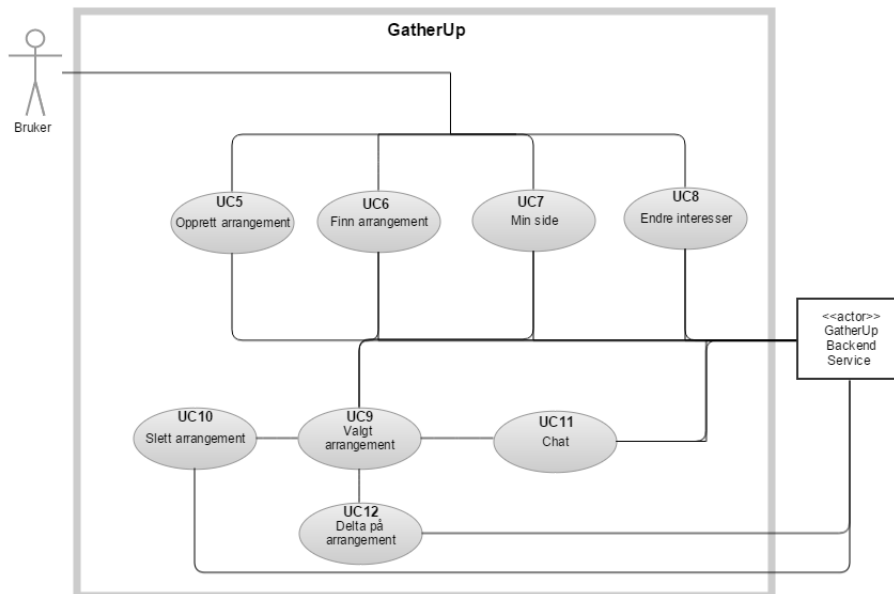
**Problemer:** Liste av problemer som må bli løst

Use Case UML-Diagram



Figur 18: Første gang brukeren åpner applikasjonen





**Figur 19:** En bruker åpner applikasjonen etter å ha logget inn tidligere

#### 4.5.1 Use Case 1: Logg Inn

Brukeren logger inn med Facebook og blir sendt til en ny side. Applikasjonen får data fra Facebook som blir lagret i en database. Use case er beskrevet i tabell 5.

**Tabell 5:** Use Case 1: Logg inn

| Use Case 1             | Logg Inn  |
|------------------------|---|
| Beskrivelse            | Applikasjonen henter og lagrer informasjon fra Facebook.  |
| Aktører                | Brukeren  |
| Forutsetninger         | Den mobile enheten er skrudd på og har internett forbindelse.   |
| Steg                   | <ol style="list-style-type: none"> <li>1. Applikasjonen henter brukerens Facebook id</li> <li>2. Applikasjonen henter tilgjengelig informasjon for Facebook id'en</li> <li>3. Applikasjonen sender tilgjengelig informasjon til serveren</li> <li>4. Serveren lagrer informasjonen i en database</li> <li>5. Applikasjonen sender brukeren til en ny side</li> </ol>              |
| Variasjoner            | <p>1a Applikasjonen får ikke tilbake Facebook id'en på grunn av feil hos Facebook</p> <p>2a Applikasjonen får ikke tilbake tilgjengelig informasjon fra Facebook på grunn av feil hos Facebook.</p> <p>3a Problemer med lagring av data på grunn av feil med forbindelse til serveren</p> <p>4a Problemer med lagring av data på grunn av feil med forbindelsen til databasen</p> |
| Ikke funksjonelle krav | #IFK1, #IFK2 og #IFK3   |
| Problemer              | Ingen   |

#### 4.5.2 Use Case 2: Legg Til Mangefull Informasjon

Etter innlogging med Facebook blir brukeren sendt til en ny side hvor han/hun må legge til eventuell informasjon som ikke var tilgjengelig fra Facebook. Dette kan for eksempel være fødseldatoen til brukeren. Use caset er beskrevet i tabell 6.

**Tabell 6:** Use Case 2: Legg Til Mangefull Informasjon

| Use Case 2             | Legg Til Mangefull Informasjon   |
|------------------------|--|
| Beskrivelse            | Brukeren skriver inn informasjonen som ikke var tilgjengelig fra Facebook. Applikasjonen lagrer så informasjonen på serveren   |
| Aktører                | Brukeren   |
| Forutsetninger         | Use Case 1: "Logg Inn". Internett forbindelse  |
| Steg                   | <ol style="list-style-type: none"> <li>1. Brukeren skriver inn manglende personlig informasjon</li> <li>2. Applikasjonen sender informasjonen til serveren</li> <li>3. Serveren lagrer informasjonen i en database</li> <li>4. Applikasjonen sender brukeren til en ny side</li> </ol> |
| Variasjoner            | <p>2a Problemer med sending av data på grunn av feil med forbindelse til serveren</p> <p>3a Problemer med lagring av data på grunn av feil med forbindelsen til databasen</p>  |
| Ikke funksjonelle krav | #IFK1, #IFK2 og #IFK3  |
| Problemer              | Ingen  |

### 4.5.3 Use Case 3: Registrer Interesser

Systemet skal være i stand til å finne arrangement som kan være aktuelle for brukere. Brukeren må derfor fortelle systemet hvilke interesser han/hun har. Use caset er beskrevet i tabell 7.

**Tabell 7:** Use Case 3: Registrer Interesser

| Use Case 3             | Registrer Interesser  |
|------------------------|---|
| Beskrivelse            | Brukeren velger interesser som blir lagret og koblet til brukeren   |
| Aktører                | Brukeren  |
| Forutsetninger         | Use Case 1: "Logg Inn". Internett forbindelse   |
| Steg                   | 1. Brukeren velger interesser<br>2. Applikasjonen sender interessene til serveren<br>3. Serveren lagrer informasjonen i en database<br>4. Brukeren blir sendt til en informasjonsside |
| Variasjoner            | 2a Problemer med sending av interesser til serveren<br>3a Problemer med lagring av data på grunn av feil med forbindelsen til databasen   |
| Ikke funksjonelle krav | #IFK1, #IFK2 og #IFK3   |
| Problemer              | Ingen   |

### 4.5.4 Use Case 4: Vis Informasjon

Første gangen brukeren logger inn i applikasjonen blir brukeren informert om hvordan applikasjonen fungerer. Use caset er beskrevet i tabell 8.

**Tabell 8:** Use Case 4: Vis Informasjon

| Use Case 4             | Vis Informasjon  |
|------------------------|--|
| Beskrivelse            | Informasjon om applikasjonen blir vist til brukeren  |
| Aktører                | Brukeren   |
| Forutsetninger         | Use Case 1: "Logg Inn". Internett forbindelse  |
| Steg                   | 1. Brukeren går igjennom informasjonssidene<br>2. Brukeren blir sendt til hovedsiden i applikasjonen |
| Variasjoner            |  |
| Ikke funksjonelle krav | #IFK1, #IFK2 og #IFK3  |
| Problemer              | Ingen  |

### 4.5.5 Use Case 5: Opprett Arrangement

Brukeren skal ha mulighet til å opprette arrangement som andre kan delta på. For å opprette et arrangement må brukeren skrive inn relevant informasjon som blir sendt til serveren. Informasjonen blir så lagret på en database som gjør arrangementet tilgjengelig for andre. Use caset er beskrevet i tabell 9.

**Tabell 9:** Use Case 5: Opprett Arrangement

| Use Case 5             | Opprett Arrangement  |
|------------------------|--|
| Beskrivelse            | Brukeren skriver inn informasjon om arrangementet som applikasjonen lagrer i en database   |
| Aktører                | Brukeren   |
| Forutsetninger         | Use Case 1: "Logg Inn". Internett forbindelse  |
| Steg                   | <ol style="list-style-type: none"> <li>1. Brukeren skriver inn informasjon om arrangementet</li> <li>2. Applikasjonen sender data om arrangementet til serveren</li> <li>3. Serveren lagrer informasjonen i en database</li> <li>4. Serveren sender tilbake en respons til applikasjonen</li> <li>5. Brukeren får tilbakemelding om arrangementet er lagret</li> </ol> |
| Variasjoner            | 2a Problemer med forbindelsen til serveren<br>3a Problemer med lagring av data til databasen<br>4a Problemer med sending av data fra serveren til applikasjonen  |
| Ikke funksjonelle krav | #IFK1, #IFK2 og #IFK3  |
| Problemer              | Ingen  |

#### 4.5.6 Use Case 6: Finn Arrangement

I use case 6 blir alle tilgjengelige arrangement vist. Ved å trykke på et arrangement blir man sendt til en ny side hvor utvidet informasjon om arrangementet blir vist. Use caset er beskrevet i tabell 10.

**Tabell 10:** Use Case 6: Finn Arrangement

| Use Case 6             | Finn Arrangement  |
|------------------------|---|
| Beskrivelse            | Applikasjonen henter og viser frem alle tilgjengelig arrangement  |
| Aktører                | Brukeren  |
| Forutsetninger         | Use Case 1: "Logg Inn". Internett forbindelse   |
| Steg                   | <ol style="list-style-type: none"> <li>1. Applikasjonen spør serveren om alle tilgjengelig arrangement, sortert etter alder og kjønn</li> <li>2. Serveren henter alle arrangement fra databasen</li> <li>3. Serveren sender alle arrangementene tilbake til applikasjonen</li> <li>4. Applikasjonen sorterer alle arrangementene etter brukerens interesser</li> <li>5. Applikasjonen viser alle arrangementene til brukeren</li> </ol> |
| Variasjoner            | 1a Problemer med forbindelsen til serveren<br>2a Problemer med henting av data fra databasen<br>3a Problemer med sending av data fra serveren til applikasjonen   |
| Ikke funksjonelle krav | #IFK1, #IFK2 og #IFK3   |
| Problemer              | Ingen   |

#### 4.5.7 Use Case 7: Min Side

Brukeren skal ha mulighet til å se arrangement han/hun har meldt seg på og arrangement de har laget. Use caset er beskrevet i tabell 11.

**Tabell 11:** Use Case 7: Min Side

| Use Case 7             | Min Side   |
|------------------------|--|
| Beskrivelse            | Applikasjonen henter brukerens arrangement og alle arrangement brukeren har meldt seg på   |
| Aktører                | Brukeren   |
| Forutsetninger         | Use Case 1: “Logg Inn”. Internett forbindelse  |
| Steg                   | 1. Applikasjonen spør serveren om alle arrangement knyttet til brukeren<br>2. Serveren henter alle arrangementene fra databasen<br>3. Serveren sender alle arrangementene tilbake til applikasjonen<br>4. Applikasjonen viser alle arrangementene til brukeren |
| Variasjoner            | 1a Problemer med forbindelsen til serveren<br>2a Problemer med henting av data fra databasen<br>3a Problemer med sending av data fra serveren til applikasjonen  |
| Ikke funksjonelle krav | #IFK1, #IFK2 og #IFK3  |
| Problemer              | Ingen  |

#### 4.5.8 Use Case 8: Endre Interesser

Brukeren skal ha mulighet for å endre sine interesser. Use caset som tar for seg endring av interesser er beskrevet i tabell 12.

**Tabell 12:** Use Case 8: Endre Interesser

| Use Case 8             | Endre Interesser   |
|------------------------|--|
| Beskrivelse            | Brukeren velger nye interesser som blir lagret i databasen   |
| Aktører                | Brukeren   |
| Forutsetninger         | Use Case 1: “Logg Inn”. Internett forbindelse  |
| Steg                   | 1. Brukeren velger nye interesser<br>2. Applikasjonen sender interessene til serveren<br>3. Serveren oppdaterer interessene til brukeren i databasen<br>4. Serveren sender en respons tilbake til applikasjonen<br>5. Brukeren får tilbakemelding om interessene ble oppdatert |
| Variasjoner            | 2a Problemer med forbindelsen til serveren<br>3a Problemer med oppdatering av data fra databasen<br>4a Problemer med sending av data fra serveren til applikasjonen  |
| Ikke funksjonelle krav | #IFK1, #IFK2 og #IFK3  |
| Problemer              | Ingen  |

#### 4.5.9 Use Case 9: Valgt Arrangement

Når brukeren har trykket på et arrangement blir han sendt til en ny side. På denne siden kan brukeren lese informasjon om arrangementet. Use caset er beskrevet i tabell 13.

**Tabell 13:** Use Case 9: Valgt Arrangement

| Use Case 9             | Valgt Arrangement  |
|------------------------|--|
| Beskrivelse            | Applikasjonen viser all informasjon om et gitt arrangement             |
| Aktører                | Brukeren   |
| Forutsetninger         | Use Case 1: “Logg Inn”. Internett forbindelse                          |
| Steg                   | 1. Applikasjonen skriver ut all informasjon som er lokalt mellomlagret |
| Variasjoner            | 1a Feil med henting av lokalt mellomlagret data                        |
| Ikke funksjonelle krav | #IFK1, #IFK2 og #IFK3  |
| Problemer              | Ingen  |

#### 4.5.10 Use Case 10: Slett Arrangement

Brukeren skal ha mulighet til å slette arrangement han/hun har opprettet. Use caset som tar for seg sletting av arrangement er beskrevet i tabell 14.

**Tabell 14:** Use Case 10: Slett Arrangement

| Use Case 10            | Slett Arrangement   |
|------------------------|---|
| Beskrivelse            | Brukeren sletter sitt eget arrangement  |
| Aktører                | Brukeren  |
| Forutsetninger         | Use Case 1 og 5: “Logg Inn” og “Opprett Arrangement”. Internett forbindelse   |
| Steg                   | 1. Applikasjonen sender en forespørsel til serveren<br>2. Serveren sletter arrangementet fra databasen<br>3. Serveren sender en tilbakemelding til Applikasjonen<br>4. Applikasjonen skriver ut tilbakemeldingen til brukeren |
| Variasjoner            | 1a Problemer med forbindelsen til serveren<br>2a Problemer med sletting av data i databasen<br>3a Problemer med sending av data fra serveren til applikasjonen  |
| Ikke funksjonelle krav | #IFK1, #IFK2 og #IFK3   |
| Problemer              | Ingen   |

#### 4.5.11 Use Case 11: Chat

For å kommunisere med medlemmene av et arrangement skal det implementeres en chat. Use caset er beskrevet i tabell 15.

**Tabell 15:** Use Case 11: Chat

| Use Case 11            | Chat  |
|------------------------|---|
| Beskrivelse            | Brukeren skriver meldinger med brukere av et arrangement  |
| Aktører                | Brukeren  |
| Forutsetninger         | Use Case 1 og 9: “Logg Inn” og “Valgt Arrangement”. Internett forbindelse   |
| Steg                   | <ol style="list-style-type: none"> <li>1. Applikasjonen sender meldingen til serveren</li> <li>2. Serveren lagrer meldingen i databasen</li> <li>3. Serveren sender en tilbakemelding til Applikasjonen</li> <li>4. Applikasjonen henter nye meldinger vært 5 sekund</li> </ol> |
| Variasjoner            | 1a Problemer med forbindelsen til serveren<br>2a Problemer med lagring av data i databasen<br>3a Problemer med sending av data fra serveren til applikasjonen   |
| Ikke funksjonelle krav | #IFK1, #IFK2 og #IFK3   |
| Problemer              | Ingen   |

#### 4.5.12 Use Case 12: Delta På Arrangement

En bruker skal ha muligheten til å melde seg på et arrangement. Use caset for å melde seg på et arrangement er beskrevet i tabell 16.

**Tabell 16:** Use Case 12: Delta På Arrangement

| Use Case 12            | Delta På Arrangement   |
|------------------------|--|
| Beskrivelse            | Brukeren melder seg på et arrangement  |
| Aktører                | Brukeren   |
| Forutsetninger         | Use Case 1 og 9: “Logg Inn” og “Valgt Arrangement”. Internett forbindelse  |
| Steg                   | <ol style="list-style-type: none"> <li>1. Applikasjonen sender en forespørsel til serveren</li> <li>2. Serveren linker brukeren til arrangementet i databasen</li> <li>3. Serveren sender en tilbakemelding til Applikasjonen</li> <li>4. Applikasjonen skriver tilbakemeldingen til brukeren</li> </ol> |
| Variasjoner            | 1a Problemer med forbindelsen til serveren<br>2a Problemer med lagring av data i databasen<br>3a Problemer med sending av data fra serveren til applikasjonen  |
| Ikke funksjonelle krav | #IFK1, #IFK2 og #IFK3  |
| Problemer              | Ingen  |

## 4.6 Demografi

Dette kapittelet presenterer den demografiske profilen av respondentene til nettundersøkelsen. Nettundersøkelsen tar sikte på å få en oversikt over deltakernes erfaring og eierskap til smart-telefoner, samt deres forhold til aktivitetsportaler.

Attributtene *karakteristikk*, *beskrivelse* og *element* forklarer detaljene av de demografiske spørsmålene. *Antall* er antall deltakere som har valgt dette svaralter-

nativet.

#### 4.6.1 Informasjon om respondentene

Tabell 17 viser den demografiske profilen av deltagerne. Nettundersøkelsen ble sendt til studenter ved NTNU og representanter for Mental Helse. Over halvparten av respondentene (57.8%) er ikke studenter. Det kan antas at den høye prosentandelen skyldes Mental Helse som har sendt undersøkelsen til eventuelle brukere av applikasjonen som ikke er studenter.

**Tabell 17:** Informasjon om respondentene

| Karakteristikk | Element              | Antall | %    |
|----------------|----------------------|--------|------|
| Kjønn          | Kvinne               | 19     | 42.2 |
|                | Mann                 | 26     | 57.8 |
| Student        | Ja                   | 19     | 42.2 |
|                | Nei                  | 26     | 57.8 |
| Årstrinn       | Ungdomskole          | 0      | 0    |
|                | Videregående         | 4      | 18.2 |
|                | Universitet/Høgskole | 17     | 77.3 |
|                | Doktorgradsstudent   | 1      | 4.5  |

#### 4.6.2 Respondentenes forhold til smarttelefoner og sosiale nettverk

Tabell 18 viser statistikk over respondentenes forhold til smarttelefoner og sosiale nettverk. Alle respondentene eier en smarttelefon eller tablet. Dette indikerer at respondentene kan være en del av sluttbrukermassen til applikasjonen. Respondentene er aktive telefonbrukere. 52.2% bruker smarttelefonen mer enn 3 timer daglig, mens 19.6% bruker telefonen mindre enn 1 time daglig. På spørsmål Q7 (“Har du brukt Facebook eller lignende for å prøve og finne ulike aktiviteter å gjøre alene eller med andre på fritiden din?”), svarte 71.7% at de hadde brukt sosiale nettverk for å finne uformelle aktiviteter. Av alle som svarte ja på spørsmål Q7 syntes 60.6% at det var vanskelig å finne relevante aktiviteter.



**Tabell 18:** Respondentenes forhold til smarttelefoner og sosiale nettverk

| Beskrivelse   | Element           | Antall | %    |
|---|-------------------|--------|------|
| Eier smarttelefon eller tablet                                  | Ja                | 46     | 100  |
|   | Nei               | 0      | 0    |
| Bruk av smarttelefon per dag                                    | Mindre enn 1 time | 9      | 19.6 |
|   | 1-2 timer         | 13     | 28.3 |
|   | 3-5 timer         | 16     | 34.8 |
|   | Mer enn 5 timer   | 8      | 17.4 |
| Brukt sosiale nettverk for å finne ulike aktiviteter å delta på | Ja                | 33     | 71.7 |
|   | Nei               | 13     | 28.3 |
| Var det vanskelig å finne relevante aktiviteter                 | Ja                | 20     | 60.6 |
|   | Nei               | 13     | 39.4 |

## 4.7 Brukerinteresse og behov

Denne seksjonen tar for seg de viktigste punktene i nettundersøkelsen - brukerinteresse og behov.

### 4.7.1 Respondentenes interesse for en aktivitetsportal

Tabell 19 inneholder de viktigste dataene fra nettundersøkelsen. 87% av respondentene var interessert i en applikasjon for å lage og delta på uformelle aktiviteter.

**Tabell 19:** Respondentenes interesse for en aktivitetsportal

| Beskrivelse  | Element | Antall | %  |
|--|---------|--------|----|
| Interessert i en applikasjon for å lage/delta på uformelle aktiviteter | Ja      | 40     | 87 |
|  | Nei     | 6      | 13 |

Respondentene ble også spurt om hva de syntes om funksjonene chat, innlogging med Facebook, push-notifikasjoner og bruk av lokasjonsdata. I tabell 20 blir resultatet presentert. Resultatet viser at alle funksjonene er viktige for applikasjonen. Muligheten for å kommunisere med andre deltagere i arrangementet viste seg å være svært viktig med hele 41.3% som sa det var “Veldig nyttig” og 34.8% som sa det var “Nyttig”. Funksjonen som kom dårligst ut var lokasjonstjenesten. Denne funksjonen skulle bruke GPS til å kun vise arrangement rundt brukeren. 17.8% sa at denne funksjonen ikke var så nyttig og 10.9% sa det var helt unyttig.

**Tabell 20:** Respondentenes svar på applikasjonens funksjoner

| Beskrivelse   | Element            | Antall | %    |
|---|--------------------|--------|------|
| Mulighet for å chatte med andre deltagere av en aktivitet           | Veldig nyttig      | 19     | 41.3 |
|   | Nyttig             | 16     | 34.8 |
|   | Ikke veldig nyttig | 4      | 8.7  |
|   | Helt unyttig       | 3      | 6.5  |
|   | Ikke sikker        | 4      | 8.7  |
| Logge inn med Facebook  | Veldig nyttig      | 12     | 26.1 |
|   | Nyttig             | 17     | 37   |
|   | Ikke veldig nyttig | 10     | 21.7 |
|   | Helt unyttig       | 5      | 10.9 |
|   | Ikke sikker        | 2      | 4.3  |
| Få notifikasjon når en aktivitet de er interessert i blir opprettet | Veldig nyttig      | 14     | 30.4 |
|   | Nyttig             | 23     | 50   |
|   | Ikke veldig nyttig | 4      | 8.7  |
|   | Helt unyttig       | 3      | 6.5  |
|   | Ikke sikker        | 2      | 4.3  |
| Hvis kun aktiviteter i nærheten av brukeren                         | Veldig nyttig      | 12     | 26.1 |
|   | Nyttig             | 14     | 30.4 |
|   | Ikke veldig nyttig | 8      | 17.8 |
|   | Helt unyttig       | 5      | 10.9 |
|   | Ikke sikker        | 7      | 15.2 |

## 5 Applikasjon

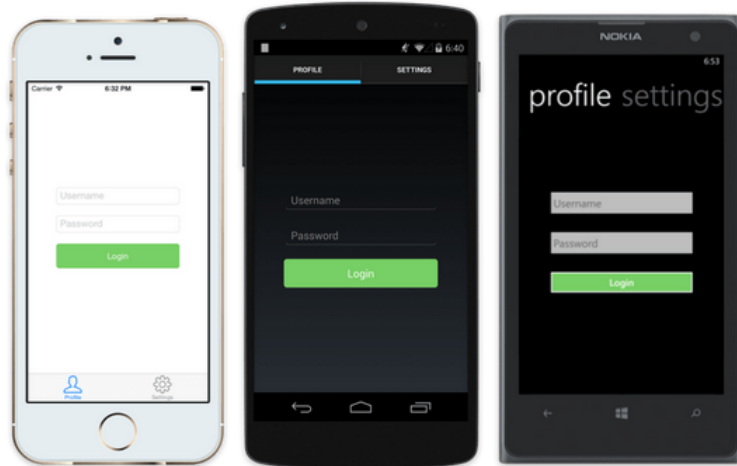
Dette kapitlet presenterer applikasjonen som er blitt utviklet gjennom denne masteroppgaven. Applikasjonen tilbyr brukerne muligheten til å opprette uformelle arrangementer, som andre brukere kan velge å bli med på uten å være avhengig av å kjenne personen som har opprettet arrangementet eller ha tilgang til en invitasjon. Personer som oppretter arrangementer kan velge hvilke brukere et arrangement passer for basert på interesser, kjønn eller alder. Målet til applikasjonen er å gjøre det enkelt å finne andre personer å gjøre en ønsket aktivitet med. Kapitlet vil begynne med en beskrivelse av brukt teknologi etterfulgt av en oversikt over implementert funksjonalitet.

### 5.1 Tekniske detaljer

I denne seksjonen kommer en kort beskrivelse av de tekniske aspektene til applikasjonen. Etter at teknologiene som er brukt er gjennomgått, kommer en beskrivelse av arkitekturmønster og hvordan det ble brukt.

#### 5.1.1 Xamarin Forms

Xamarin ble valgt for å utvikle applikasjonen. Undersøkelsene gjort i fordypningsprosjektet, se digitalt vedlegg 1, er grunnlaget for valget. Xamarin gir utviklere muligheten til å dele kode mellom alle plattformer, som inkluderer Android, Windows phone og iOS. Utviklingen ble utført i Visual Studio sammen med GitHub for å håndtere kildekode. Xamarin Forms sitt brukergrensesnitt blir renderet ved hjelp av de native kontrollene på hver plattform, slik at applikasjonen får riktig utseende i forhold til de retningslinjene som er satt for plattformen den kompiles til. Dette medfører at en tekstboks i Xamarin Forms kan bli seende annerledes ut på iOS og Android.



**Figur 20:** Lik kode kjørt på forskjellige plattformer

Figur 20 er et eksempel hvor samme kode kjører på de ulike plattformene, men med ganske ulikt utseende. Dersom man vil oppnå det samme resultatet på hver plattform, kan man lage egentilpassede rendere som lar utviklerne overstyre de native kontrollene med sin egen plattform-spesifikke kode. Egentilpassede rendere gir full fleksibilitet på hvordan kontrollene ser ut og oppfører seg, og kan omfatte plattformsspesifikk kode for å implementere native sdk-funksjoner som for eksempel push-notifikasjoner. Applikasjonen inneholder flere slike rendere for å lage en applikasjon som opplevdes forholdsvis likt på de ulike plattformene. Applikasjonen inneholder også en del funksjonalitet som krever plattformspesifikk kode. Under kapittel 5.2 vil det komme noen kodeeksempler med tilpassede rendere.

Applikasjonen består av følgende prosjekter i en Visual Studio Solution:

- Shared Project hvor alt av delt kode som kjører på alle plattformene ligger. Prosjektet inkluderer brukergrensesnitt, logikk og datamodeller.
- Android-prosjekt hvor all plattformsspesifikk kode for Android ligger.
- iOS-prosjekt hvor all plattformsspesifikk kode for iOS ligger.

### 5.1.2 Google App Engine

Applikasjonen benytter en server og en database som er plassert på Google sin sky-løsning, Google App Engine. Serveren er skrevet med Java kode og bruker teknologi som Google Cloud Endpoints. Klientbibliotekene som blir generert av Google Cloud Endpoint gir applikasjonen mulighet til å kjøre direkte API-kall

mot serveren. I kode 1 er det vist et eksempel på hvordan Google Cloud Endpoint er benyttet. Linje 1 inneholder informasjon om metoden som blir kalt. Navnet “saveUser” blir satt for å kunne kalle riktig API metode fra klienten. Utvikleren kan også bestemme hvilken http metode som skal benyttes. I dette eksempelet blir det benyttet POST som http metode. Metoden som blir kjørt er vist i linje 2, 3 og 4.

```
1 @ApiMethod(name="saveUser", httpMethod = HttpMethod.POST)
2 public User saveUser(User u) throws NotFoundException{
3     //Add user to the cloud.
4     return u;
5 }
```

**Kode 1:** Google Cloud Endpoint metode

Google Cloud SQL er benyttet for å lagre og hente data til applikasjonen. Databasen er basert på vanlig SQL teknologi og skalerer etter applikasjonens behov.

### 5.1.3 Programmeringsspråk

All applikasjonskode er skrevet i C#. Xamarin Forms tilbyr muligheten å skrive brukergrensesnitt i XAML, men er enklere å debugge med C# i Visual Studio. XAML har også vist seg å ha dårligere ytelse, da den blir inkludert som en ressurs og blir derfor tolket under kjøring uten noen forbehandling. C# er et velkjent objekt-orientert programmeringsspråk som ble gitt ut som en del av .NET plattformen i 2002. Det omfatter sterk typing, imperativ, deklarativ, funksjonell, generisk og komponentorienterte programmeringsdisipliner [10].

### 5.1.4 Arkitektur

Applikasjonen sin arkitektur er sterkt inspirert av Model-View-ViewModel mønsteret. MVVM består av tre hovedkonsepter:

- Separere brukergrensesnitt bekymringene (Visningen), forretningsobjektene og behaviors (Visningsmodellen) og datalaget (Modellen).
- Man kan konseptualisere Visningen (brukergrensesnittobjektene) på toppen, Visningsmodellen (objekter som gir brukergrensesnittet sine data og atferd) i midten og Modellen nederst, se figur 21 . Visningen kan vite om Visningsmodellen og Visningsmodellen kan vite om Modellen.

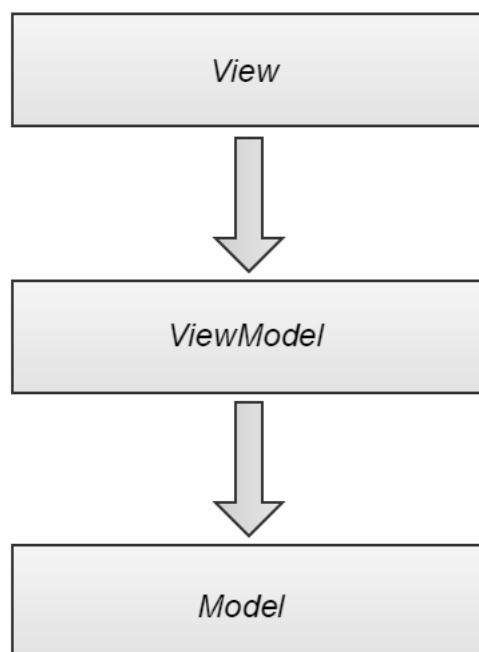
- I MVVM er mekanismen for å få Visningsmodellen til å gi data til Visningen, at Visningen angir Visningsmodellen som sin datakontekst. Vi knytter altså både data og kommandoer på denne måten mellom Visningen og Visningsmodellen.

MVVM frakobler også systemlogikk fra data, og reduserer dermed avhengigheter i koden. Det betyr at deler av systemet kan lettere oppdateres uten å nødvendigvis måtte endre på hele systemet. En skisse med delene i MVVM er illustruert i figur 21.

Ved å bruke bindinger og gjøre Visningsmodellen til datakonteksten til Visningen er koden mye lettere å vedlikeholde, og man øke testbarheten av applikasjonen. Visningsmodellen har ingen brukergrensesnitt, men inneholder kun det man vil teste. Noe som forenkler testingen radikalt.

Model-View-ViewModel

---



**Figur 21:** Arkitekturen MVVM

### 5.1.5 Støtte for flere skjermstørrelser

For å øke brukeropplevelsen er det fire punkter som må følges. Punktene sørger for at designet er tilpasset brukerens enhet og er listet opp under.

- **Skjermstørrelse** er den fysiske størrelsen, målt som skjermens diagonal.
- **Orientering** av skjermen fra brukerens synspunkt. Dette er enten liggende eller stående, noe som betyr at skjermens størrelseforhold er enten bredt eller høyt.
- **Oppløsning** er det totale antall fysiske piksler på en skjerm.
- **Density-independent pixel** også kjent som dp er en virtuell pikselenhet som brukes for å definere UI layout, for å uttrykke layout dimensjoner eller posisjoner på en tetthetsuavhengig måte.

På bakgrunn av at applikasjonen skal kjøres både på iOS og Android innebefatter det mange enheter med ulike skjermstørrelser, oppløsninger og piksel tettheter. Spesielt på Android siden det er et gratis operativsystem for hardware produsenter. Å skalere bilder, kontrollere og ikoner for ulike skjermstørrelser, oppløsninger og piksel tettheter øker brukeropplevelsen siden det virker som om applikasjonen var designet spesielt for deres enhet. Det ble implementert forskjellige layouts for ulike skjermer. Skjermens høyde og bredde blir hentet ut i dp (density-independent pixels), før ulike konstanter som bestemmer størrelsen til bildene, ikoner og kontrollene i de ulike layoutene blir regnet ut basert på målingene av skjermen.

Applikasjonen har blitt testet på syv ulike Android enheter og fem iOS enheter. Det inkluderer alt fra Samsung S2 (480x800px) til Samsung Tablet Pro 10.1 (2560x1600px) og iPhone 4 (640x920px) til iPad Air (1536x2048px). Applikasjonen var utviklet til å brukes i portrett modus hvor hovedgrunnen er alle bildene som er designet til å se bra ut i portrett. Disse ville blitt strukket i landskapsmodus og ødelagt brukeropplevelsen.

### 5.1.6 Javascript Object Notation

Javascript Object Notation også kjent som JSON er et uavhengig tekst-data format som er lesbart både for mennesker og maskiner. Språket er designet for rask lagring og utveksling av tekstlig informasjon. I applikasjonen bruker .NET APIet Newtonsoft.Json som er et bibliotek som brukes for å tolke data. All data som utveksles

mellom applikasjonen og webtjenesten er i JSON format. Newtonsoft.Json blir brukt i alle metoder som kommuniserer med webtjenesten. Kode 2 er et eksempel hvor alle arrangement til en bruker hentes fra webtjenesten. Dataen som blir sendt tilbake fra webtjenesten blir tolket med metoden DeserializeObject.

```
1 public static async Task<bool> GetGatheringsAsync()
2 {
3     List<Event> events = new List<Event>();
4     try
5     {
6         var url = "https://river-sky-851.appspot.com/_ah/api/socialactapi/v1/
7         eventcollection/" + SharedData.User.Gender + "/" + ConvertBirthday(SharedData.
8         User.Birthday) + "/" + SharedData.User.Id;
9         var request = HttpRequest.Create(string.Format(@url));
10        request.Proxy = null;
11        request.ContentType = "application/json";
12        request.Method = "GET";
13        using (HttpResponse response = await request.GetResponseAsync() as
14        HttpResponse)
15        {
16            using (StreamReader reader = new StreamReader(response.GetResponseStream()))
17            {
18                var content = reader.ReadToEnd();
19                var data = JsonConvert.DeserializeObject<EventRootobject>(
20                content);
21                foreach (Item e in data.items){
22                    List<User> userList = new List<User>();
23                    foreach (JsonUser u in e.users){
24                        userList.Add(new User(u.uid, u.first_name, u.gender, u.name, u.email,
25                        u.image_url, u.birthday));
26                    }
27                    events.Add(new Event(e.id, e.name, e.desc, e.date, e.time, e.
28                    participants, e.location, e.creatorId, e.interest.value, userList, e.gender, e.
29                    ageFrom, e.ageTo));
30                }
31            }
32            SharedData.events = events;
33            response.Close();
34            response.Dispose();
35            return true;
36        }
37    } catch (Exception e){
38        System.Diagnostics.Debug.WriteLine("Exception in saveEvent" + e);
39        return false;
40    }
41 }
```

**Kode 2:** Applikasjonen henter alle arrangement fra serveren

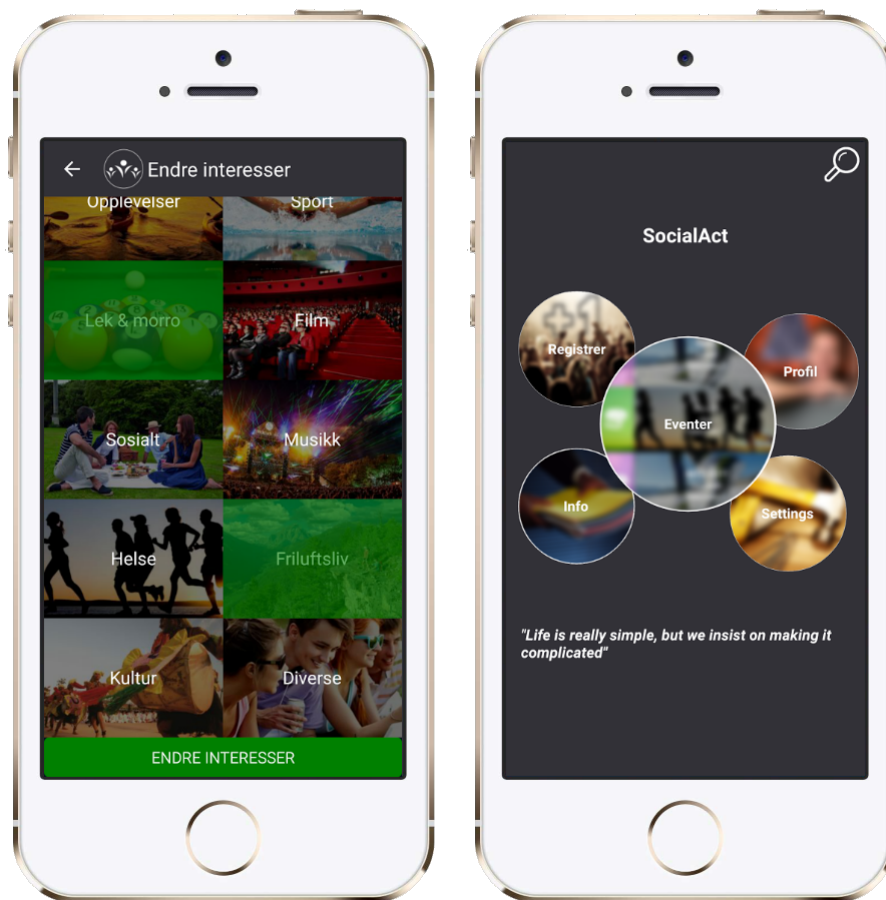


## 5.2 Oversikt over funksjonalitet

Denne seksjonen vil inneholde all funksjonalitet, som inkluderer både synlig og usynlig funksjonalitet for brukeren. Funksjonalitet som er synlig for brukeren vil være alt hvor brukeren interagerer med applikasjonen, som for eksempel å lage et arrangement, søke etter arrangement, opprette bruker, slette et arrangement m.m. Usynlig funksjonalitet er oppgaver applikasjonen håndterer uten at brukere kan endre eller gjøre noe med det. Sortering av listevisningen basert på interesser, sende push-varslinger, hente ut data fra serveren i et bestemt tidsintervall er eksempler på dette. Aktivitetene er automatisk gjort i bakgrunnen og kan ikke bli kontrollert av brukeren sine handlinger.

Brukervennligheten til applikasjonen har vært i fokus gjennom iterative tester. Nielsen's heuristiske prinsipper (kapittel 3.4.1) og System Usability Scale(kapittel 3.4.2) har blitt fulgt gjennom utviklingen av applikasjonen. Applikasjonen starter med en splash skjerm som vises i kort tid dersom brukeren ikke er innlogget. Om brukeren tidligere har laget en bruker og logget inn, vil den vare lengre da en asynkron bakgrunnsprosess henter nødvendig data fra serveren som skal brukes til å vise frem brukerinformasjon, ulike arrangementer, hvem som deltar med mer.

Ved Facebook-innlogging blir all offentlig informasjon brukeren har på facebook hentet ut ved å bruke APIet Xamarin.Facebook. Da applikasjonen trenger å vite kjønn samt alder, noe man kun kan hente ut fra facebook dersom brukeren har informasjonen som offentlig, vil en ExtraInformationNeeded skjerm dukke opp dersom vi ikke kan hente all ønsket informasjon. Denne skjermen ber kun om at brukeren fyller inn kjønn og alder.

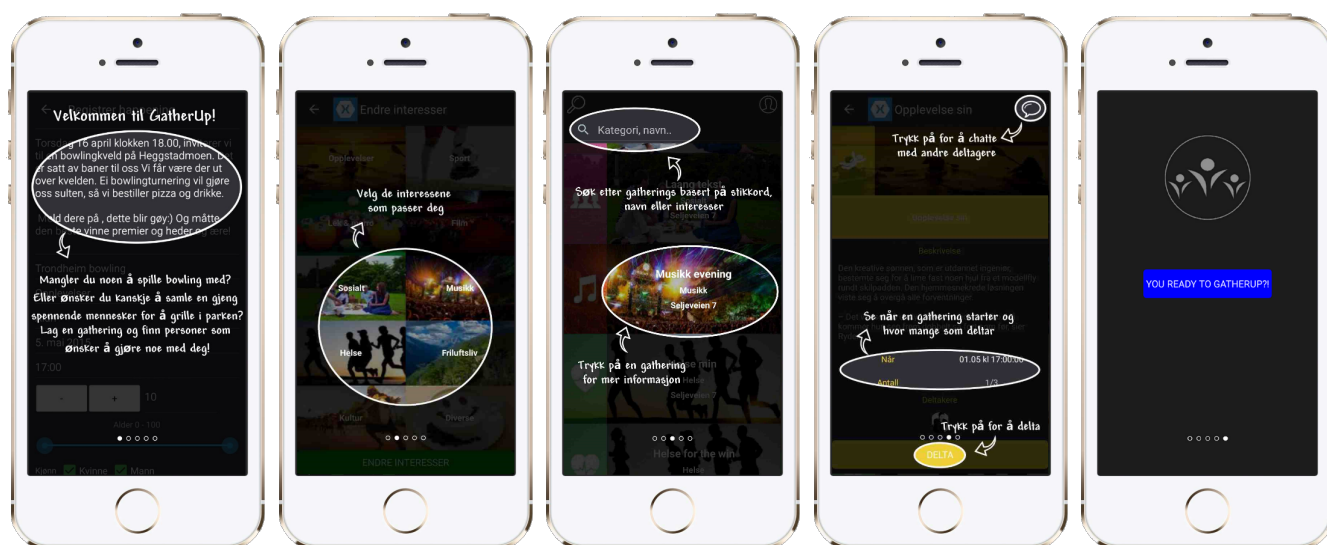


(a) Interesseside

(b) Hovedsiden

**Figur 22:** Applikasjonens interesseside og hovedside

Siste steg i selve registreringen er å velge interesser. Her må brukeren velge minimum en interesse, og maksimum tre. På figur 22a er “Lek & Morro” og “Friluftsliv” valgt. Før brukeren kommer inn på hovedsiden vil en CarouselPage med fem sider som viser hvordan applikasjonen skal brukes dukke opp, se figur 23. Dette skjer kun ved førstegangsregistrering.



(a) Hjelpeskjerm 1

(b) Hjelpeskjerm 2

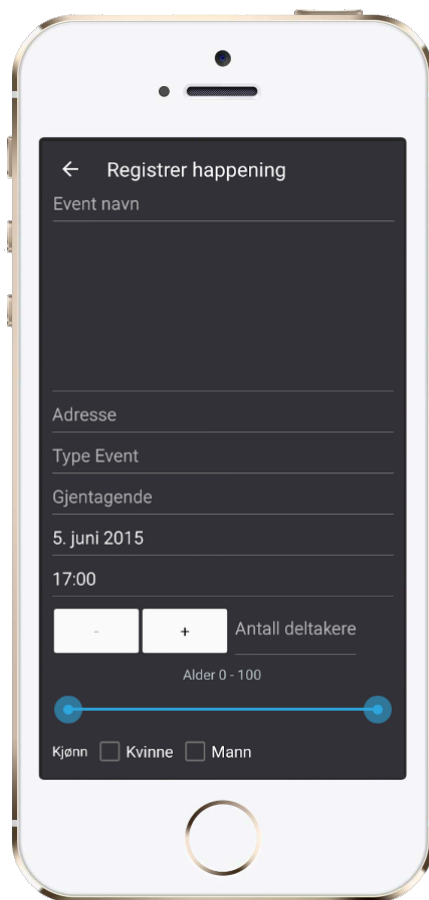
(c) Hjelpeskjerm 3

(d) Hjelpeskjerm 4

(e) Hjelpeskjerm 5

**Figur 23:** Applikasjonens veiledningssider

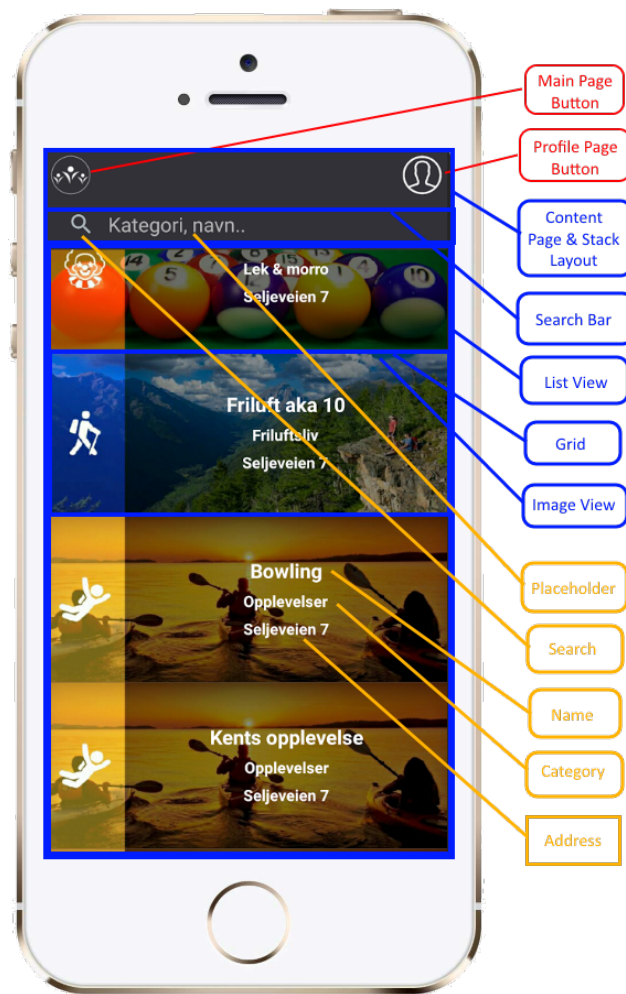
Hovedsiden, som er det første brukeren ser om man er logget inn og åpner applikasjonen, inneholder fem knapper i form av bilder som har en animasjon for at applikasjonen skal oppleves levende. Nederst befinner det seg et sitat som kan oppdateres ved å forandre et felt i databasen som ligger på serveren, se figur 22b. Designet gjenspeiler målet om å vise informasjonen på en enkel måte, men samtidig med et tiltalende design. Knappene på hovedskjermen sørger for at alle finner det man trenger, det er også mulig å sveipe mellom sidene. Både de med begrenset og avanserte kunnskap til smarttelefon er tenkt på under utviklingen.



**Figur 24:** Side for å opprette et arrangement

Siden for oppretting av et arrangement er ren uten mange distraksjoner, se figur 24. Her er brukerinputen i sentrum, og det skal være enkelt å se hva som skal fylles inn i de ulike feltene. Plassholdere er brukt for å minske skjermområde brukt for å vise hva som skal hvor, og det er noe som blir brukt i de fleste forms på både nettsider og applikasjoner. Når en bruker er ferdig med å fylle inn informasjonen og trykker på “Opprett” så valideres all brukerinputen og dersom noe mangler eller er feil kommer det opp en spesifikk melding som gir beskjed om dette.

Skjermen over alle arrangementene hvor brukeren vil være mesteparten av tiden, vist i figur 25, viser alle tilgjengelige arrangement. Oransje elementer er indikatorer for å hjelpe brukeren å forstå den viste informasjonen. Blå elementer er visninger og røde elementer er knapper for å vise hva brukeren kan gjøre. I tabell 21, 22 og 23 følger en beskrivelse av de forskjellige elementene for å vise hvordan en typisk skjerm i appen er bygd opp.



**Figur 25:** Liste med alle tilgjengelige arrangement

**Tabell 21:** Beskrivelse av visninger

| Visninger                  | Beskrivelse   |
|----------------------------|---|
| Content Page & Stacklayout | Skjermen er en del av en CarouselPage, hvor hver skjerm er en Content Page. Hver Content Page har en Stacklayout, som er en layout som kan fylles med ulike visninger |
| SearchBar                  | Viser tastaturet når brukeren trykker på visningen  |
| ListView                   | Scrollbar og hver rad kan trykkes på. Inneholder flere elementer  |
| Grid                       | Hvert element i listevisningen er en grid (rutenett) som inneholder ulike visninger med innhold   |
| ImageView                  | Visning som har et bilde som kilde  |

**Tabell 22:** Beskrivelse av indikatorer

| Indikatorer | Beskrivelse  |
|-------------|--|
| Placeholder | Hjelpetekst som indikerer hvilke nøkkelord brukeren kan søke med |
| Search      | Indikerer at det er en søkebar tilgjengelig her                  |
| Name        | Indikerer navnet til arrangementet                               |
| Category    | Indikerer kategorien til arrangementet                           |
| Address     | Indikerer adressen til arrangementet                             |

**Tabell 23:** Beskrivelse av knappene

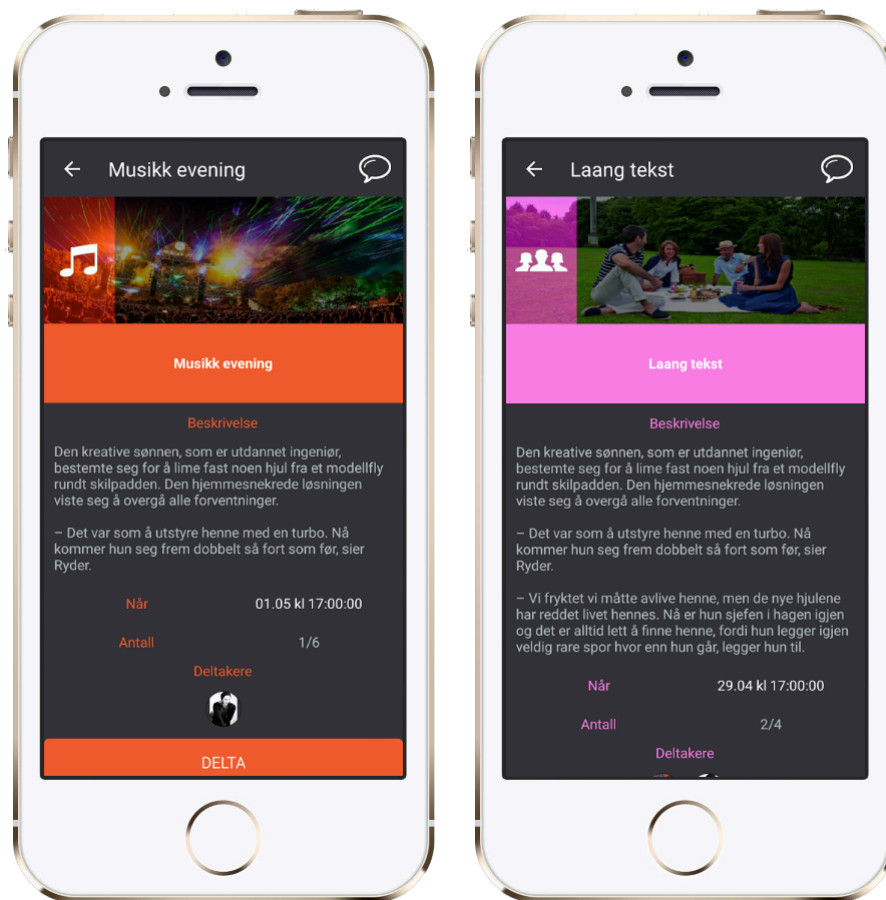
| Knapp             | Beskrivelse  |
|-------------------|--|
| MainPage knapp    | Tar brukeren tilbake til hovedsiden med en animasjon                                   |
| ProfilePage knapp | Tar brukeren til profilsiden hvor man ser alle sine kommende og opprettede arrangement |

Profilskjermen vist i figur 26 består av brukerens navn og bilde, samt to listervisninger. Den ene viser arrangement som brukeren har opprettet, og den andre viser arrangement hvor brukeren deltar. Dersom brukeren velger et arrangement han/hun har opprettet er det mulig å slette den. Velger brukeren et arrangement han/hun deltar på, er det mulig å slette deltakelsen sin.



**Figur 26:** Applikasjonens profilside

Skjermen som viser informasjonen til et arrangement, vist i figur 27, har sterke farger som gjenspeiler kategorien arrangementet er under. Det vil si, et arrangement under kategorien musikk har andre farger enn et arrangement under friluftsliv. Viktig informasjon fra arrangementet vises på en tiltalende måte. Øverst til høyre finnes en knapp, som gir tilgang til en chat, hvor man kan utveksle meldinger med de andre deltakerne. Mer om chat funksjonaliteten i kapittel 5.2.3.



(a) Arrangement 1

(b) Arrangement 2

**Figur 27:** Informasjon om valgt arrangement

Brukervennligheten til applikasjonen har vært i fokus gjennom iterative tester. Det er enkelt å se Nielsen's heuristikker i applikasjonens brukervennlighet. Første prinsippet "Visibility of System Status" handler om å holde brukeren oppdatert om hva som foregår i applikasjonen. Noen eksempler: Alt som tar tid er asynkrone bakgrunnsprosesser slik at brukergrensesnittet aldri er låst, men heller viser en prosessindikator eller at man fortsatt kan interagere med brukergrensesnittet. Når en bruker velger å delta eller opprette et arrangement vil en midlertidig tekstboks bli presentert til brukeren og gi bekreftelse på at dette er blitt gjort, før brukeren blir tatt til profilskjermen hvor man kan se de nye endringene.

Brukere gjør ofte feil og vil forlate den uønskete tilstanden (User Control and Freedom). Hensynet til denne retningslinjen kan sees ved at man alltid har en tilbakeknapp øverst i venstre hjørne. Eller at man i carousel pagen bare kan sveipe



seg tilbake. Alt av deltakelse, oppretting og endringer krever også bekreftelse fra brukeren. Dette for å være sikker på at brukeren ikke gjør noe han/hun ikke har lyst til.

Designet på applikasjonen går mye ut på å bruke bilder fra virkeligheten med skarpe farger som går igjen i applikasjonen. Ikonene går også igjen, og alle bildene er designet for å skape en helhet. Dette oppfyller Nielsen's fjerde prinsipp ("Consistency and standards"). Bilder fra virkeligheten som illustrerer de forskjellige interessene følger retningslinjen om at systemet skal snakke brukerens språk ("Match between system and the real world"). Brukeren kjenner seg igjen og skjønner hva interessen betyr uten å nødvendigvis måtte lese teksten. Farger på knappene forandres også ettersom man skal delta eller slette et arrangement. Fargene justeres til noe brukere kan relatere seg til, som for eksempel trafikklys.

Dette er en sosial applikasjon med mange brukere som må oppdateres dersom en bruker gjør noe, internett kreves dermed til de aller fleste handlinger. Dersom brukeren ikke har internettforbindelse vil en meldingsboks gi beskjed om hva som må gjøres for at applikasjonen skal fungere som forventet. Det er gjort intensiv testing og feilhåndtering som sørger for at applikasjonen aldri låser seg, men heller gir feilmelding dersom noe går galt. Dette samsvarer med prinsippene "Error Prevention and Help Users Recognize" og "Diagnose and Recover From Error".

Applikasjonen gjenspeiler å ha fulgt Nielsen's heuristikker, hvor de aller fleste prinsippene kommer godt frem som diskutert.

### 5.2.1 Innlogging med Facebook

Skjermen "Logg inn" presenterer en knapp for innlogging med Facebook. Dersom brukeren har Facebook-applikasjonen installert på telefonen og er logget inn, vil Facebook spørre om brukeren tillater at applikasjonen mottar data. Om Facebook-applikasjonen ikke finnes på telefonen, vil en webvisning åpnes hvor brukeren må logge inn på Facebook. Dette er tilfellet på både iOS og Android. Funksjonaliteten oppleves likt på begge plattformene, men implementeringen er forskjellig.

En egentilpasset renderer er utviklet til begge plattformene, og de arver begge fra ButtonRenderer. Android sin OnClick metode starter en ny aktivitet som heter FacebookActivity, se kode 3.

```
1 private class ButtonClickListener : Object, IOnClickListener  
2 {
```

```

3 public static readonly Lazy<ButtonClickListener> Instance = new Lazy<
4   ButtonClickListener>(() => new ButtonClickListener());
5
6 public void OnClick(View v)
7 {
8     var myIntent = new Intent(_activity, typeof(FacebookActivity));
9     _activity.StartActivityForResult(myIntent, 0);
10 }

```

**Kode 3:** Egentilpasset renderer for Android

Den nye aktiviteten kjører kode som finnes i API'et Xamarin.Facebook, som sørger for å åpne Facebook-applikasjonen eller en webvisning for å be brukeren om godkjenning. Når det er gjort kalles OnActivityResult, se kode 4.

```

1 protected override void OnActivityResult(int requestCode, Result resultCode,
2   Intent data)
3 {
4     base.OnActivityResult(requestCode, resultCode, data);
5
6     if (resultCode == Result.FirstUser)
7     {
8         var token = data.GetStringExtra("facebookToken");
9         SharedData.LoginToken(token);
10    }

```

**Kode 4:** OnActivityResult i MainActivity.cs

I metoden OnActivityResult mottar programmet en nøkkel (access token) som må brukes for å hente ut brukerinformasjonen ved å kjøre en get-forespørsel til Facebook. For å hente nøkkelen på iOS er det nødvendig i AppDelegate-klassen å registrere både FacebookAppID og FacebookAppName. Disse verdiene trenger å stemme overens med verdiene som er registrert på Facebook sin utviklerside, se kode 5.

```

1 [Register ("AppDelegate")]
2 public partial class AppDelegate : global::Xamarin.Forms.Platform.iOS.
3   FormsApplicationDelegate
4 {
5     private const string FacebookAppId = "151159766685255";
6     private const string FacebookAppName = "GatherUp";

```

**Kode 5:** Applikasjonsinformasjon som brukes av Facebook (AppDelegate.cs)

Videre er det nødvendig å override ytterligere to metoder i AppDelegate, se kode 6.

```

1 public override bool OpenUrl(UIApplication application, NSURL url, string
   sourceApplication, NSObject annotation)
2 {
3     base.OpenUrl(application, url, sourceApplication, annotation);
4     return FBSession.ActiveSession.HandleOpenURL(url);
5 }
6
7 public override void OnActivated(UIApplication application)
8 {
9     base.OnActivated(application);
10    FBSession.ActiveSession.HandleDidBecomeActive();
11 }

```

**Kode 6:** OnActivated og OpenUrl i AppDelegate.cs

Det må også implementeres en tilpasset renderer på FacebookLoginButton som kommuniserer med det native FacebookAPIet til iOS i Xamarin, se kode 7.

```

1 [assembly: ExportRenderer(typeof(FacebookLoginButton), typeof(
   FacebookLoginButtonRendererIos))]
2 namespace socialact.iOS
3 {
4     public class FacebookLoginButtonRendererIos : ButtonRenderer
5     {
6         protected override void OnElementChanged(ElementChangedEventArgs<Button> e)
7         {
8             base.OnElementChanged(e);
9
10            if (Control != null)
11            {
12                UIButton button = Control;
13
14                button.TouchUpInside += delegate
15                {
16                    HandleFacebookLoginClicked();
17                };
18            }
19        }
20
21        private void HandleFacebookLoginClicked()
22        {
23            if (FBSession.ActiveSession.IsOpen)
24            {
25                App.PostSuccessFacebookAction(FBSession.ActiveSession.AccessTokenData.
                AccessToken);
26            } else {
27                FBSession.ActiveSession.Open(FBSessionLoginBehavior.
                UseSystemAccountIfPresent, (aSession, status, error) =>
28                {
29                    if (error == null)
30                    {

```

```

31     App.PostSuccessFacebookAction(aSession.AccessTokenData.AccessToken);
32     }
33     });
34 }
35 }
36 }
37 }

```

**Kode 7:** Hele FacebookLoginRendererIos.cs som arver fra ButtonRenderer

I HandleFacebookLoginClicked åpnes en session som kjører kode som finnes i Xamarin.Facebook, se kode 7. Når nøkkelen er mottatt, blir metoden GetUserInfoAsync i SharedProject kalt både fra iOS prosjektet og Android prosjektet, se kode 8.

```

1 public static async Task<bool?> GetUserInfoAsync(string token)
2 {
3     try
4     {
5         var url = "https://graph.facebook.com/me/?access_token=" + token;
6         var request = HttpWebRequest.Create(string.Format(@url));
7         request.Proxy = null;
8         request.ContentType = "application/json";
9         request.Method = "GET";
10        using (HttpWebResponse response = await request.GetResponseAsync() as
11        HttpWebResponse)
12        {
13            using (StreamReader reader = new StreamReader(response.GetResponseStream()))
14            {
15                var content = reader.ReadToEnd();
16                var data = Newtonsoft.Json.JsonConvert.DeserializeObject<Rootobject>(
17                content);
18                SharedData.UserExists = await UserExists(data.id);
19
20                var info = CheckDataValuesIfNull(data);
21                string birthday = info[0];
22                string email = info[1];
23                string firstname = info[2];
24                string gender = info[3];
25                string name = info[4];
26
27                if (data.gender == null || data.birthday == null && (SharedData.UserExists
28                .HasValue && !SharedData.UserExists.Value)){
29                    SharedData.temporaryUserInfo = new TemporaryUserInfo(data.id, firstname,
30                    name, email, "https://graph.facebook.com/"+data.id+"/picture?type=large");
31                    return null;
32                }
33
34                var _result = await SaveUserInfoAsync(data.id, firstname, gender, name,
35                email, birthday);
36            }
37        }
38    }
39 }

```

```

31     if (!_result){
32         throw new Exception();
33     }
34 }
35 response.Close();
36 response.Dispose();
37 }
38 }
39 catch (Exception e)
40 {
41     System.Diagnostics.Debug.WriteLine("Exception i GetUserInfo: " + e);
42     return false;
43 }
44 return true;
45 }

```

**Kode 8:** GetUserInfoAsync i Backend.cs

Data i responsen fra Facebook blir sjekket for hvilken informasjon som følger med. Dersom kjønn eller bursdag ikke har en verdi blir kun et midlertidig brukerinfo-objekt opprettet. Deretter vil en ExtraInformationNeeded-skjerm dukke opp hvor brukeren må fylle ut den manglende informasjonen. Disse funksjonene ble implementert for å løse FK1(“Applikasjonen må støtte innlogging med Facebook”).

### 5.2.2 Presentere arrangement i applikasjonen

Største utfordringen med å vise arrangementene er alle bildene med god oppløsning som tar mye minne. Applikasjonen skal fungere på eldre telefoner, helt ned til Samsung S2 og iPhone 4S. Å programmere mobile applikasjoner kan sees på som å programmere for datamaskiner for 10-20 år tilbake, man må være forsiktig med å allokere for mye minne da det er lett å få Out Of Memory (OOM)-unntak. Det finnes veldig mange forskjellige oppløsninger, spesielt på Android. Et bilde som skal vises på en tablet med 2550x1820px oppløsning trenger å ha høyere oppløsning enn om det skal vises på en 960x640px skjerm. Mye arbeid er blitt gjort med tanke på minnehåndtering, spesielt når det kommer til bildene. Applikasjonen sjekker oppløsningen på mobilen før alle bildene blir kjørt gjennom en Resize-metode, se kode 9.

```

1 protected void ResizeImage()
2 {
3     //OnAppearing();
4     var assembly = typeof(EventBox).GetTypeInfo().Assembly;
5     byte[] imageData;

```

```

6 Stream stream = assembly.GetManifestResourceStream(ResourcePrefix +src);
7 using (MemoryStream ms = new MemoryStream())
8 {
9     stream.CopyTo(ms);
10    imageData = ms.ToArray();
11 }
12 byte [] resizedImage = ImageResizer.ResizeImage(imageData, (float)SharedData.
13     WidthPictures, (float)SharedData.HeightPictures);
14
15 this.img.Source = ImageSource.FromStream(() => new MemoryStream(resizedImage));
16 }

```

**Kode 9:** ResizeImage i EventBox.cs

WidthPictures og HeightPictures er den bestemte oppløsningen bildene skal ha basert på oppløsningen på telefonen. Tester har blitt gjort for å finne passende oppløsninger for de ulike skjermene. Metoden kaller “ImageResizer.ResizeImage” som velger to ulike metoder for å endre størrelse på bilde på, se kode 10.

```

1 public static byte [] ResizeImage(byte [] imageData, float width, float height)
2 {
3     #if __IOS__
4         return ResizeImageIOS ( imageData, width, height );
5     #endif
6     #if __ANDROID__
7         return ResizeImageAndroid(imageData, width, height);
8     #endif
9     #if WINDOWS_PHONE
10        return ResizeImageWinPhone ( imageData, width, height );
11    #endif
12 }

```

**Kode 10:** ResizeImage i ImageResizer.cs

På Android blir et Bitmap-objekt av det originale bildet opprettet, før metoden CreateScaledBitmap skalerer ned bildet til ønsket oppløsning og returnerer et nytt bitmap-objekt. Bitmap-objektet komprimeres til slutt for å ta mindre minne, se kode 11.

```

1 public static byte [] ResizeImageAndroid(byte [] imageData, float width, float
2     height)
3 {
4     // Load the bitmap
5     Bitmap originalImage = BitmapFactory.DecodeByteArray(imageData, 0, imageData.
6     Length);
7     Bitmap resizedImage = Bitmap.CreateScaledBitmap(originalImage, (int)width, (int)
8     height, false);
9 }

```

```

6
7     using (MemoryStream ms = new MemoryStream())
8     {
9         resizedImage.Compress(Bitmap.CompressFormat.Jpeg, 100, ms);
10        return ms.ToArray();
11    }
12 }

```

**Kode 11:** ResizeImageAndroid i ImageResizer.cs

iOS løsningen inneholder en del ekstra kode, se kode 12. Her blir først byte-tabellen med bildet kjørt gjennom en ImageFromByteArray-metode som returnerer et UIImage-objekt. Et rektangel blir laget for å kunne kjøre CGContext sin DrawImage metode. Til slutt må metoden ToArray() kjøres på det endrede bilde for å returnere en bytetabell.

```

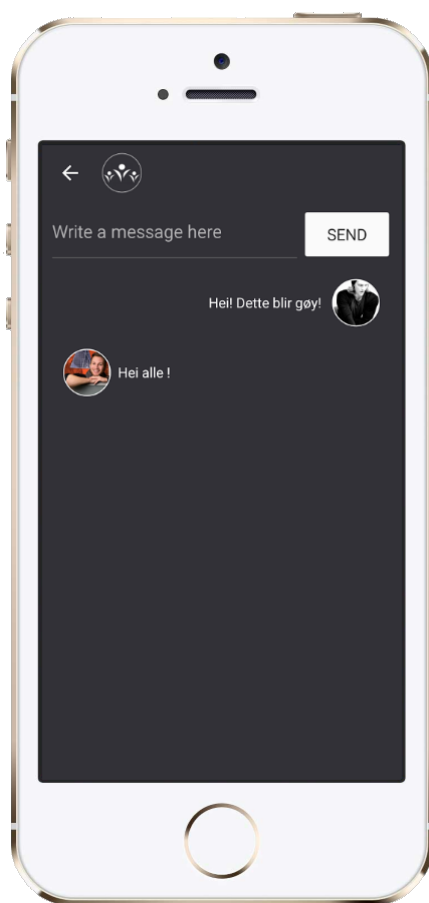
1 public static byte[] ResizeImageIOS (byte[] imageData, float width, float height)
2 {
3     UIImage originalImage = ImageFromByteArray (imageData);
4     //create a 24bit RGB image
5     using (CGContext context = new CGContext (IntPtr.Zero,
6         (int)width, (int)height, 8,
7         (int)(4 * width), CGColorSpace.CreateDeviceRGB (),
8         CGImageAlphaInfo.PremultipliedFirst)) {
9         RectangleF imageRect = new RectangleF (0, 0, width, height);
10        // draw the image
11        context.DrawImage (imageRect, originalImage.CGImage);
12        MonoTouch.UIKit.UIImage resizedImage = MonoTouch.UIKit.UIImage.FromImage (
13            context.ToImage ());
14        // save the image as a jpeg
15        return resizedImage.AsJPEG ().ToArray ();
16    }
17 }

```

**Kode 12:** ResizeImageIOS i ImageResizer.cs

### 5.2.3 Chat funksjonalitet

Xamarin hadde ikke ekstra støtte for chat-funksjonalitet som gruppen syntes fungerte tilfredsstillende. Det ble derfor utviklet en egen løsning. Figur 28 består av en ContentPage som har en StackLayout med en grid som inneholder en tekstboks hvor brukeren kan skrive en beskjed, samt en knapp for å sende meldingen. Stack-Layouten har også en listevissning som består av ChatBox objekter. Et slikt objekt arver fra en grid, og har en ImageCircle og en label.



**Figur 28:** Kommunikasjon med andre brukere

For å sjekke for nye meldinger i chatten kjøres en `CheckForNewMessagesAsync` metode som har en `while`-løkke som går uendelig, se kode 13. Ved `await Task.Delay(2000)` så venter tråden 2 sekunder før den går videre for å kalle databasen. Serveren får tilsendt ID'en til arrangementet og IDen til den siste meldingen applikasjonen har. Den sjekker så om det er kommet nye meldinger, er dette tilfellet vil resultatet ikke være null, og inneholde en liste med de nye meldingene. De nye meldingene blir lagt inn i listen som inneholder alle de gamle, for så at listevisningen oppdateres med nye `ChatBox`-objekter. Dette skjer hvert andre sekund, siden `Task.Delay` er asynkron og de nye meldingene også blir hentet asynkront så merker ikke brukeren noen forsinkelser i appen og kan bruke den som vanlig mens dette kjøres.

```
1 private async void CheckForNewMessagesAsync ()
2 {
3     while (true){
4         await Task.Delay(2000);
```



```

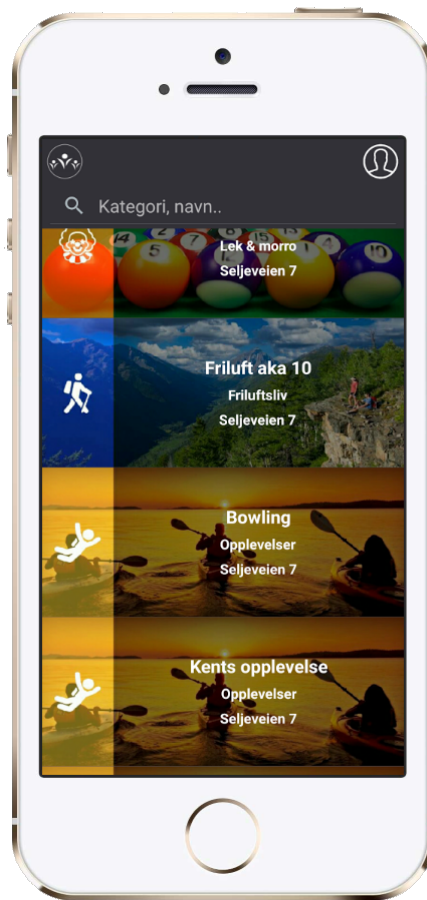
5     var result = await Backend.CheckForNewChatMessagesOnAEvent(aEventId, Messages
6     [0].Id);
7     if (result != null && result.Count > 0){
8         var doesExist = false;
9         foreach (ChatMessage message in result){
10            foreach (ChatMessage oldMessages in KeepMessages){
11                if (oldMessages.Id == message.Id){
12                    doesExist = true;
13                    break;
14                }
15            }
16            if (!doesExist){
17                KeepMessages.Insert(0, message);
18            }
19            doesExist = false;
20        }
21        Messages.Clear();
22        chatCounter = 0;
23        KeepMessages.Where(t => t.ImageUrl != null).ToList().ForEach(t => Messages.
24        Add(t));
25    }

```

**Kode 13:** CheckForNewMessages i ChatPage.cs

#### 5.2.4 Søkefelt

På figur 29 finnes en listevisning som viser alle tilgjengelig arrangement for brukeren, som samsvarer med brukerens kjønn og alder. Listen blir sortert slik at arrangementene som er i samme kategori som brukerens interesser havner øverst. Det kan være mange arrangement som vises i listevisningen. Derfor ble det implementert en søkebar som gjør det lettere for brukeren å finne ønsket arrangement, se kode 14 . Søkebaren har en placeholder som sier hvilke nøkkelord brukeren kan søke på, navn, kategori og adresse. Søkefunksjonaliteten leter også i substrenger, som vil si at brukeren ikke må skrive inn hele navnet/kategorien/adressen. Søkningen skjer også i sanntid slik at brukeren kan umiddelbart se hvilke arrangement som dukker opp (eller forsvinner) for hvert tegn som blir skrevet inn. Når noe er skrevet i søkefeltet, dukker det opp et kors på høyre side. Denne x-knappen sletter tekst i søkefeltet, slik at sletting av lange ord er en enkel oppgave.



**Figur 29:** Liste over arrangement

```

1 private void SearchClicked(object sender, EventArgs e)
2 {
3     if (searchBar.Text == null || searchBar.Text.Equals("")){
4         foreach (Event ev in SharedData.KeepEvents){
5             SharedData.Eventer.Add(ev);
6         }
7         return;
8     }
9     var searchText = searchBar.Text.Trim();
10    SharedData.KeepEvents.Where(t => (t.Name.ToLower().Contains(searchText.ToLower())
11    || SharedData.Interests[t.Interests].name.ToLower().Contains(searchText.
    ToLower()))).ToList().ForEach(t => SharedData.Eventer.Add(t));

```

**Kode 14:** SearchClicked i EventPage.cs

SearchClicked er en metode som kjøres dersom søkeknappen blir trykt, eller teksten i søkefeltet forandres. Listen som inneholder alle arrangement, som vises i lis-

tevisningen heter `SharedData.Eventer`. Mens `SharedData.KeepEvents` inneholder alle arrangement som brukeren har. Det sjekkes først om teksten i søkfeltet er tom, dersom dette er tilfellet skal alle arrangement vises. Om søkfeltet inneholder noe, kjøres en linq-spørring som finner alle arrangement i `KeepEvents` som matcher enten navn, kategori eller adresse på det som står i søkfeltet.

## 6 Resultat

I dette kapitlet vil resultatene fra testene som ble utført på brukergrensesnittet presentert, sammen med akseptansetestene. På bakgrunn av disse resultatene kan brukergrensesnittet og akseptansen av applikasjonen bli evaluert. Kapitlet vil også ta for seg demografien til de ulike deltakerne av begge testene, før resultatene vil bli utdypet ytterligere.

### 6.1 Demografi

Dette underkapitlet vil presentere to ulike demografiske profiler. En for deltakerne som ble brukt for å teste brukergrensesnittet og en for brukerne av akseptansetesten. Begge testene har et spørreskjema som skal fylles ut for å gi en bedre oversikt over deltakerne sin profilinformasjon, erfaring med smarttelefoner og bruk av sosiale medier for å finne aktiviteter å bli med på. Det er blitt brukt en SUS-test for å vurdere brukervennligheten og oppdage eventuelle feil ved brukergrensesnittet. En MSAM ble brukt for å vurdere hvorvidt applikasjonen som blir utviklet løser det opprinnelige problemet beskrevet i kapittel 4.

#### 6.1.1 Informasjon om respondentene for den heuristiske evalueringen

Tabell 24 gir en oversikt over de ulike deltakerne i den heuristiske evalueringen basert på alder, kjønn, samt tidsbruk og erfaring med bruk av smarttelefoner og/eller tablets. Alle deltakerne eier en smarttelefon eller en tablet. Flertallet av deltakerne (60%) har et relativt høyt smarttelefonbruk på mellom 3 og 5 timer om dagen. 20% av deltakerne bruker enheten mer enn 5 timer om dagen og 20% bruker enheten mellom 1 til 2 timer om dagen.

**Tabell 24:** Informasjon om respondentene for den heuristiske evalueringen

| Karakteristikk                           | Element           | Antall | %   |
|--|-------------------|--------|-----|
| Kjønn                                    | Mann              | 3      | 60  |
|  | Kvinnen           | 2      | 40  |
| Alder                                    | 23                | 1      | 20  |
|  | 24                | 1      | 20  |
|  | 25                | 1      | 20  |
|  | 27                | 1      | 20  |
|  | 54                | 1      | 20  |
| Eierskap av smarttelefon/tablet          | Ja                | 5      | 100 |
|  | Nei               | 0      | 0   |
| Erfaring ved bruk av smarttelefon/tablet | 1 (Nybegynner)    | 0      | 0   |
|  | 2                 | 0      | 0   |
|  | 3                 | 2      | 40  |
|  | 4                 | 1      | 20  |
|  | 5 (Ekspert)       | 2      | 40  |
| Bruk av smarttelefon/tablet per dag      | Mindre enn 1 time | 0      | 0   |
|  | 1-2 timer         | 1      | 20  |
|  | 3-5 timer         | 3      | 60  |
|  | Mer enn 5 timer   | 1      | 20  |

### 6.1.2 Informasjon om respondentene for MSASM

Tabell 25 viser en oversikt over den demografiske profilen til de ulike deltakerne som var med i akseptansetestingen. Elementer som kjønn, alder, erfaring med smarttelefoner og gjennomsnittlig bruk av smarttelefon gjennom hverdagen er presentert. Alle deltakerne eier en smarttelefon eller en tablet. Den største andelen av deltakerne bruker smarttelefonen og/eller tableten mellom 1 til 5 timer om dagen. 7.5% av deltakerne bruker enheten mindre enn 1 time om dagen, mens 5% av deltakerne bruker enheten mer enn 5 timer om dagen.

**Tabell 25:** Informasjon om respondentene for MSASM

| Karakteristikk                           | Element           | Antall | %    |
|--|-------------------|--------|------|
| Kjønn                                    | Mann              | 25     | 62.5 |
|  | Kvinnen           | 15     | 37.5 |
| Alder                                    | 20-25             | 27     | 67.5 |
|  | 26-30             | 7      | 17.5 |
|  | 31-40             | 2      | 5    |
|  | 41+               | 4      | 10   |
| Eierskap av smarttelefon/tablet          | Ja                | 40     | 100  |
|  | Nei               | 0      | 0    |
| Erfaring ved bruk av smarttelefon/tablet | 1 (Nybegynner)    | 1      | 2.5  |
|  | 2                 | 1      | 2.5  |
|  | 3                 | 16     | 40   |
|  | 4                 | 12     | 30   |
|  | 5 (Ekspert)       | 10     | 25   |
| Bruk av smarttelefon/tablet per dag      | Mindre enn 1 time | 3      | 7.5  |
|  | 1-2 timer         | 18     | 45   |
|  | 3-5 timer         | 17     | 42.5 |
|  | Mer enn 5 timer   | 2      | 5    |

## 6.2 Evaluering av brukergrensesnitt

Dette kapittelet tar for seg testresultatene av den heuristiske evalueringen som ble utført på brukergrensesnittet, sammen med resultatene av SUS-testen. Under SUS-testen ble testdeltakerne bedt om å vurdere et sett med spørsmål knyttet til applikasjonen og dens brukergrensesnitt på en skala fra 1 til 5, hvor 1 er “sterkt uenig”, mens 5 er “sterkt enig”. Det er også viktig å påpeke at SUS-testen ble utført mens utviklingen av brukergrensesnittet ikke var ferdig, så forbedringer er blitt gjort i etterkant.

Det ble også foretatt en spørreundersøkelse for å undersøke hvorvidt testdeltagerne savnet noe funksjonalitet i forhold til det som allerede var implementert, sammen med generelle kommentarer vedrørende applikasjonen, brukergrensesnittet og/eller eventuelle problemer de støtet på under testene.

### 6.2.1 Resultat av heuristisk evaluering

En heuristisk evaluering av applikasjonen ble utført for å finne svakheter knyttet til brukergrensesnittet på bakgrunn av Nielsens 10 heuristikker som beskrevet i kapittel 3.4.1. Testdeltagerne ble bedt om å utføre tre forskjellige oppgaver:

- Finn et arrangement hvor du kan være med og spille fotball. Hvor mange personer deltar på dette arrangementet?
- Opprett et arrangement ved navn “Grilling på Kalvskinnet”. Tillat at begge kjønn får lov til å melde seg på arrangementet og sett aldersgrensen mellom 18 og 30 år.
- Meld deg på et arrangement ved navn “Kents Opplevelse” og bli med i chatten. Hva er den siste meldingen som er blitt skrevet i chatten på dette arrangementet?

Hver testedeltaker utførte oppgavene individuelt flere ganger og skrev ned brukervennlighetsproblemer og hvilke heuristikker disse problemene brøt. Denne evalueringen registrerte 8 ulike problemer knyttet til brukergrensesnittet som beskrevet i tabell 26:

**Tabell 26:** Problemer knyttet til brukergrensesnittet

| # | Problem   | Krenket heuristikker |
|---|---|----------------------|
| 1 | Vanskeligheter for brukeren å forstå at vedkommende er inne i hjelpeskjermen.                     | #1                   |
| 2 | Uklart hvor man skal trykke for å komme seg videre gjennom hjelpeskjermen                         | #1                   |
| 3 | Ingen bekreftelse når en bruker sletter en event  | #9                   |
| 4 | Bruker forstår ikke at knappen “Registrer” oppretter et nytt event                                | #4                   |
| 5 | Burde være en placeholder i tekstfeltet hvor en bruker kan skrive inn en beskrivelse for et event | #6                   |
| 6 | Misforståelser rundt hva “Dine eventer” er i profilen.  | #2                   |
| 7 | Ordet “Setting” blir brukt for menyvalget hvor en bruker kan endre interesser.                    | #2                   |
| 8 | Misforståelser rundt hva 2/4 deltakere betyr inn på et arrangement.                               | #2                   |

- Problem #1 og #2: Første gangen en bruker starter applikasjonen vil vedkommende bli presentert med et sett av hjelpebilder som illustrerer ulike funksjoner applikasjonen innehar og hvordan disse funksjonene kan brukes. Et navigeringselement er plassert på bunnen av bildene for å vise brukeren hvor han/hun befinner seg i settet med bilder. Brukerne hadde vanskeligheter med å forstå at det var en hjelpeskjerm og hvordan man skulle navigere seg gjennom bildene.
- Problem #3: Om en bruker befinner seg inne på et arrangement som brukeren

selv har opprettet eksisterer det en knapp som tillater brukeren å slette valgt arrangement. Når en bruker trykker på denne knappen slettes arrangementet uten at en ekstra dialogboks dukker opp for å bekrefte at brukeren virkelig ønsker å slette arrangementet. Dette medfører at det er mulig for en bruker å slette et arrangement ved en feiltakelse.

- Problem #4: For å opprette et nytt arrangement, må brukeren trykke på knappen merket “Registrer”. Brukeren var i tvil om dette var for å registrere et arrangement eller om det var for å registrere en ny bruker. Dette bidrar til tvil og usikkerhet rundt bruk av applikasjonen og bryter med den fjerde heuristikken som sier at systemet skal snakke brukerens språk og bruke ord, uttrykk og begreper som er kjent for brukeren.
- Problem #5: Når en bruker skal skrive inn informasjon vedrørende et nytt arrangement han eller hun ønsker å opprette, er det ingen informasjon som forklarer at det er arrangementbeskrivelsen som skal stå inne i tekstfeltet.
- Problem #6: Under en bruker sin profil finnes det en liste med arrangementer som er merket med “Dine eventer”. Det var knyttet usikkerhet om det betydde arrangementer som brukeren hadde opprettet eller arrangementer som brukeren hadde meldt seg på.
- Problem #7 og #8: Ordet “Setting” er blitt brukt for å beskrive knappen som tillater en bruker å endre interesser. Dette burde bli endret med mindre applikasjonen skal ha flere innstillinger som kan endres på under dette menyvalget. I tillegg var enkelte av testdeltakerne usikre på om etiketten merket “2/4” inne på et arrangement betydde at det var 2 av 4 deltakere som hadde meldt seg på et arrangement.

### 6.2.2 Resultat av SUS

SUS-testen består av en spørreundersøkelse med 10 ulike spørsmål. Oddetall-spørsmålene er positivt formulert, mens partall-spørsmålene er negativt formulert. På bakgrunn av dette må Brookes kalkuleringsmetode benyttes for å gjøre resultatene sammenlignbare. Det må dermed subtraheres et poeng fra brukerens poengsum for hvert oddetalls-spørsmål, mens på hvert partalls-spørsmål må brukerens poengsum subtraheres fra 5. Dette gir hvert spørsmål en minimum score på 0 og en maksimum score på 4.



Tabell 27 viser poengsummen de ulike spørsmålene i SUS-testen endte opp med, samtidig som det er verdt å bemerke seg at hvert spørsmål oppnådde en over gjennomsnittlig poengsum. Spørsmål 10 (Jeg vil være avhengig av å lære meg masse før jeg kan begynne å bruke dette systemet) hadde den høyeste poengsummen på 3.4, noe som også indikerer en lav terskel for å komme i gang med å bruke applikasjonen som er blitt utviklet.

**Tabell 27:** Resultat av SUS-test

|         | 0      | 1      | 2      | 3      | 4      |              |
|---------|--------|--------|--------|--------|--------|--------------|
| Spm.nr. | Antall | Antall | Antall | Antall | Antall | Gjennomsnitt |
| 1       | 0      | 0      | 1      | 4      | 0      | 2.6          |
| 2       | 0      | 0      | 1      | 3      | 1      | 2.8          |
| 3       | 0      | 0      | 0      | 4      | 1      | 2.6          |
| 4       | 1      | 0      | 0      | 1      | 3      | 2.6          |
| 5       | 0      | 0      | 2      | 1      | 2      | 2.6          |
| 6       | 0      | 0      | 2      | 3      | 0      | 2.6          |
| 7       | 0      | 0      | 1      | 2      | 2      | 3.2          |
| 8       | 0      | 1      | 0      | 4      | 0      | 2.6          |
| 9       | 0      | 0      | 1      | 2      | 2      | 3.2          |
| 10      | 0      | 1      | 0      | 0      | 4      | 3.4          |

Ved å summere de normaliserte resultatene fra hver deltaker og multiplisere dette resultatet med 2.5, vil en få den totale poengsummen til den aktuelle deltakeren på SUS-testen. Tabell 28 viser poengsummen til hver deltaker, sammen med den gjennomsnittlige poengsummen på bakgrunn av alle testdeltakerne.

**Tabell 28:** Respondentenes SUS poengsum

| Testdeltaker | 1    | 2    | 3  | 4  | 5  | Gjennomsnitt |
|--------------|------|------|----|----|----|--------------|
| SUS poengsum | 57.5 | 77.5 | 80 | 75 | 85 | 75           |

I følge Jeff Sauro [25] er den gjennomsnittlige SUS-poengsummen på 68. En poengsum over 68 vil bli klassifisert som over gjennomsnittlig og alt under 68 er under gjennomsnitt. Prototypen utviklet i denne masteroppgaven oppnådde en gjennomsnittlig poengsum på 75. Dette gir en indikasjon på at brukervennligheten med tanke på brukergrensesnittet ligger på et tilfredstillende nivå, samtidig som det er viktig å tenke på at dette er en prototype. Dermed er det fremdeles mulig å forbedre poengsummen ytterligere. GPS og kartfunksjonen for de ulike arrangementene ble ikke lagt vekt på under brukervennlighetstestene da denne funksjonaliteten ikke var ferdig implementert på det aktuelle tidspunktet.

## 6.3 Resultat av MSAM

Dette kapittelet inneholder resultatene fra akseptansetestingen som er blitt gjennomført med et spesielt tilpasset MSAM. Programvaren SmartPLS er blitt brukt for å måle gyldigheten og påliteligheten av tilbakemeldingene.

### 6.3.1 Deskriptive resultater

Totalt 40 personer testet en prototype av applikasjonen og svarte på spørreskjemaet. Spørreskjemaet inneholdt spørsmål for å hente ut profilinformasjonen fra testdeltakeren, sammen med 18 andre spørsmål knyttet til MSAM. De 18 spørsmålene er fordelt på seks forskjellige variabler: Perceived Usefulness (PU), Perceived Ease of Use (PEOU), Trust (T), Personal Initiatives and Characteristics (PIAC), Context (C) og Intention to Use (ITU). Alle spørsmålene kan bli funnet i vedlegg C.

Hver variabel består av 2 til 4 spørsmål som måler ulike faktorer relevante for variabelen de er innenfor. Svarene på spørsmålene rangeres fra 1 til 7 hvor 1 representerer “Sterkt uenig”, mens 7 representerer “Sterkt enig”. Attributtene *Min* og *Max* er den minste og maksimale verdien som er blitt registrert på et spørsmål, mens *Mean* er gjennomsnittsverdien på spørsmålet som blir kalkulert ved å legge sammen poengsummen til alle svarene for så å dele det på antall svar som er blitt registrert. Videre er *N* brukt for å beskrive antall forekomster av et svar og *%* er prosentvis forekomst av et svar.

#### 6.3.1.1 Perceived Usefulness

Tabell 29 oppsummerer svarene på spørsmålene som omhandler variabelen PU. Respondentene benyttet seg ikke av hele poengskalaen, da svarene befant seg mellom intervallet 2 og 7. Spørsmål PU1 (“Jeg finner raskt og enkelt en happening å være med på”) oppnådde en poengsum mellom 2 og 7, mens spørsmål PU3 (“Det er enkelt å lage arrangementer som jeg ønsker at andre skal få være med på”) sin poengsum varierte mellom 4 og 7. PU2 (“Det er større sannsynlighet for at jeg finner arrangementer jeg ikke visste eksisterte”) scorete en poengsum mellom 3 og 7, men hadde den høyeste gjennomsnittsverdien med 6.03. PU1 og PU3 hadde en gjennomsnittsverdi på henholdsvis 5.35 og 5.4 som oppsummert i tabell 30.

**Tabell 29:** Oppsummering av svar for PU

|     | 1 |   | 2 |     | 3 |     | 4 |      | 5  |    | 6  |      | 7  |      |
|-----|---|---|---|-----|---|-----|---|------|----|----|----|------|----|------|
|     | N | % | N | %   | N | %   | N | %    | N  | %  | N  | %    | N  | %    |
| PU1 | 0 | 0 | 1 | 2.5 | 1 | 2.5 | 5 | 12.5 | 16 | 40 | 10 | 25   | 7  | 17.5 |
| PU2 | 0 | 0 | 0 | 0   | 1 | 2.5 | 3 | 7.5  | 6  | 15 | 14 | 35   | 16 | 40   |
| PU3 | 0 | 0 | 0 | 0   | 0 | 0   | 9 | 22.5 | 12 | 30 | 13 | 32.5 | 6  | 15   |

**Tabell 30:** Statisk oppsummering av spørsmålene knyttet til PU

| Spørsmål | N  | Min | Max | Mean |
|----------|----|-----|-----|------|
| PU1      | 40 | 2   | 7   | 5.35 |
| PU2      | 40 | 3   | 7   | 6.03 |
| PU3      | 40 | 4   | 7   | 5.40 |

### 6.3.1.2 Perceived Ease of Use

Tabell 31 oppsummerer svarene på spørsmålene som omhandler variabelen PEOU. Spørsmål PEOU1 (“Jeg synes appen er forståelig”) og PEOU3 (“Jeg synes det er enkelt å få appen til å gjøre det jeg vil”) benyttet seg ikke av hele poengskalaen da svarene befant seg mellom 3 og 7. 85% av svarene var mellom 4 og 6 for PEOU1, mens dette tallet steg til 90% for PEOU3. PEOU2 (“Bruk av appen krever lite konsentrasjon”) sin poengskala varierte mellom 2 og 7 med en gjennomsnittsverdi på 5.10. PEOU1 oppnådde den høyeste gjennomsnittsverdien med 5.20, mens den tilsvarende verdien for PEOU3 var 5.18. Disse gjennomsnittsverdiene, oppsummert i tabell 32, indikerer at mesteparten av brukerne mener at brukervennligheten til applikasjonen er over gjennomsnittet. Noe som også blir bekreftet av SUS-testen i kapittel 6.2.2.

**Tabell 31:** Oppsummering av svar for PEOU

|       | 1 |   | 2 |     | 3 |     | 4  |      | 5  |      | 6  |    | 7 |      |
|-------|---|---|---|-----|---|-----|----|------|----|------|----|----|---|------|
|       | N | % | N | %   | N | %   | N  | %    | N  | %    | N  | %  | N | %    |
| PEOU1 | 0 | 0 | 0 | 0   | 2 | 5   | 8  | 20   | 14 | 35   | 12 | 30 | 4 | 10   |
| PEOU2 | 0 | 0 | 1 | 2.5 | 0 | 0   | 13 | 32.5 | 11 | 27.5 | 10 | 25 | 5 | 12.5 |
| PEOU3 | 0 | 0 | 0 | 0   | 1 | 2.5 | 11 | 27.5 | 11 | 27.5 | 14 | 35 | 3 | 7.5  |

**Tabell 32:** Statisk oppsummering av spørsmålene knyttet til PEOU

| Spørsmål | N  | Min | Max | Mean |
|----------|----|-----|-----|------|
| PEOU1    | 40 | 3   | 7   | 5.20 |
| PEOU2    | 40 | 2   | 7   | 5.10 |
| PEOU3    | 40 | 3   | 7   | 5.18 |

### 6.3.1.3 Trust

Tabell 33 oppsummerer svarene på spørsmålene som omhandler variabelen T. Testdeltakerne benyttet seg av hele poengskalaen, men mesteparten av svarene er innenfor intervallet 5-7. Spørsmål T2 (“Jeg vil bruke appen hvis den beskytter brukers personvern”) og T3 (“Jeg vil bruke appen hvis jeg føler at data fra appen er til å stole på”) hadde en varierende poengsum på mellom 2 og 7, med en gjennomsnittsverdi på henholdsvis 5.33 for T2 og 5.30 for T3. Spørsmål T1 (“Jeg vil bruke appen hvis jeg har en klar oppfatning av all funksjonalitet appen innehar”) var det eneste spørsmålet i denne kategorien som benyttet seg av hele poengskalaen. 70% av svarene på spørsmålet var innenfor poengene 5-7 med en gjennomsnittsverdi på 4.90 som oppsummert i tabell 40.

**Tabell 33:** Oppsummering av svar for Tillit

|    | 1 |     | 2 |     | 3 |     | 4 |      | 5  |    | 6  |      | 7 |      |
|----|---|-----|---|-----|---|-----|---|------|----|----|----|------|---|------|
|    | N | %   | N | %   | N | %   | N | %    | N  | %  | N  | %    | N | %    |
| T1 | 1 | 2.5 | 1 | 2.5 | 3 | 7.5 | 7 | 17.5 | 14 | 35 | 12 | 30   | 2 | 5    |
| T2 | 0 | 0   | 1 | 2.5 | 2 | 5   | 7 | 17.5 | 12 | 30 | 9  | 22.5 | 9 | 22.5 |
| T3 | 0 | 0   | 1 | 2.5 | 3 | 7.5 | 5 | 12.5 | 14 | 35 | 8  | 20   | 9 | 22.5 |

**Tabell 34:** Statisk oppsummering av spørsmålene knyttet til Tillit

| Spørsmål | N  | Min | Max | Mean |
|----------|----|-----|-----|------|
| T1       | 40 | 1   | 7   | 4.90 |
| T2       | 40 | 2   | 7   | 5.33 |
| T3       | 40 | 2   | 7   | 5.30 |

### 6.3.1.4 Personal Initiatives and Characteristics

Tabell 35 oppsummerer svarene som omhandler variabelen PIAC. Nesten hele poengskalaen ble tatt i bruk av testdeltagerne på spørsmål PIAC2 (“Jeg vil gjerne være blant de første til å bruke appen”) hvor 5% benyttet seg av scoren 1. Derimot var det 85% av deltagerne som avla en score mellom 4 og 6. Spørsmål PIAC1 (“Jeg er i stand til å bruke denne appen”) hadde et svar intervall mellom 4 og 7, noe som betyr at deltakerne av testen ikke opplevde store vanskeligheter med å bruke applikasjonen. PIAC3 (“Bruk av denne appen er en god ide”) hadde mesteparten av svarene i det samme intervallet som PIAC1, bortsett fra én deltager som benyttet seg av poengsummen 2. Gjennomsnittsverdien for henholdsvis PIAC1, PIAC2 og PIAC3 er listet opp i tabell 36 med verdiene 5.85, 4.50 og 5.80. De høye gjennom-

snittsverdiene for PIAC1 og PIAC3 indikerer at brukerne har en positiv holdning til å være blant de første til å bruke denne applikasjonen. Det indikerer også at deltakerne mener applikasjonen er en god idé.

**Tabell 35:** Oppsummering av svar for PIAC

|       | 1 |   | 2 |     | 3 |     | 4  |    | 5  |      | 6  |      | 7  |      |
|-------|---|---|---|-----|---|-----|----|----|----|------|----|------|----|------|
|       | N | % | N | %   | N | %   | N  | %  | N  | %    | N  | %    | N  | %    |
| PIAC1 | 0 | 0 | 0 | 0   | 0 | 0   | 4  | 10 | 11 | 27.5 | 12 | 30   | 13 | 32.5 |
| PIAC2 | 2 | 5 | 0 | 0   | 3 | 7.5 | 14 | 35 | 14 | 35   | 6  | 15   | 1  | 2.5  |
| PIAC3 | 0 | 0 | 1 | 2.5 | 0 | 0   | 6  | 15 | 8  | 20   | 9  | 22.5 | 16 | 40   |

**Tabell 36:** Statisk oppsummering av spørsmålene knyttet til PIAC

| Spørsmål | N  | Min | Max | Mean |
|----------|----|-----|-----|------|
| PIAC1    | 40 | 4   | 7   | 5.85 |
| PIAC2    | 40 | 1   | 7   | 4.50 |
| PIAC3    | 40 | 2   | 7   | 5.80 |

### 6.3.1.5 Context

Tabell 37 oppsummerer svarene på spørsmålene som omhandlet variabelen C. Svarene som ble avgitt rundt denne variabelen var mer spredt, hvor blant annet spørsmål C1 (“Jeg vil bruke denne appen når jeg ønsker å finne på noe, men mangler en aktivitet å gjennomføre”) og C4 (“Jeg vil bruke denne appen hvis den finnes til PC når det er enklest for meg å bruke PC”) benyttet seg av hele poengskalaen. Gjennomsnittsscore på disse to spørsmålene var derimot 4.68 og 3.95 som vist i tabell 38. Spørsmål C2 (“Jeg vil bruke denne appen når jeg mangler personer for å gjennomføre en aktivitet eller et arrangement”) sine svar varierte i intervallet 2 til 7 med en gjennomsnittsscore på 4.75. Spørsmål C3 (“Jeg vil bruke denne appen hvis den finnes til mobil når det er enklest for meg å bruke mobilen”) sitt laveste registrerte svar var 1, mens det høyeste registrerte svaret var 7. Derimot benyttet ikke respondentene seg av hele poengskalaen som vist i tabell 37. Gjennomsnittsscoren for spørsmål C3 var også den høyeste gjennomsnittsscoren av samtlige spørsmål knyttet til kontekst med 5.20. Dette indikerer at mesteparten av brukerne ønsker å benytte seg av applikasjonen om den er tilgjengelig på mobil.

**Tabell 37:** Oppsummering av svar for Kontekst

|    | 1 |     | 2 |     | 3 |    | 4  |      | 5  |      | 6  |      | 7 |    |
|----|---|-----|---|-----|---|----|----|------|----|------|----|------|---|----|
|    | N | %   | N | %   | N | %  | N  | %    | N  | %    | N  | %    | N | %  |
| C1 | 1 | 2.5 | 1 | 2.5 | 4 | 10 | 11 | 27.5 | 14 | 35   | 5  | 12.5 | 4 | 10 |
| C2 | 0 | 0   | 3 | 7.5 | 2 | 5  | 12 | 30   | 10 | 25   | 11 | 27.5 | 2 | 5  |
| C3 | 1 | 2.5 | 1 | 2.5 | 0 | 0  | 5  | 12.5 | 15 | 37.5 | 16 | 40   | 2 | 5  |
| C4 | 3 | 7.5 | 4 | 10  | 6 | 15 | 15 | 37.5 | 5  | 12.5 | 5  | 12.5 | 2 | 5  |

**Tabell 38:** Statisk oppsummering av spørsmålene knyttet til Kontekst

| Spørsmål | N  | Min | Max | Mean |
|----------|----|-----|-----|------|
| C1       | 40 | 1   | 7   | 4.68 |
| C2       | 40 | 2   | 7   | 4.75 |
| C3       | 40 | 1   | 7   | 5.20 |
| C4       | 40 | 1   | 7   | 3.95 |

### 6.3.1.6 Intention of Use

Tabell 39 oppsummerer svarene på spørsmålene som omhandler variabelen ITU. Både spørsmål ITU1 (“Jeg har lyst til å bruke appen hvis jeg får tilgang til den”) og ITU2 (“Jeg kommer til å bruke appen hvis jeg får tilgang til den”) benyttet seg av hele poengskalaen, med et gjennomsnittssvar på henholdsvis 5.20 for ITU1 og 5.10 for ITU2 som oppsummert i tabell 40.

**Tabell 39:** Oppsummering av svar for Bruksintensjon

|      | 1 |     | 2 |     | 3 |     | 4 |     | 5  |      | 6  |      | 7 |    |
|------|---|-----|---|-----|---|-----|---|-----|----|------|----|------|---|----|
|      | N | %   | N | %   | N | %   | N | %   | N  | %    | N  | %    | N | %  |
| ITU1 | 1 | 2.5 | 1 | 2.5 | 2 | 5   | 4 | 10  | 13 | 32.5 | 15 | 37.5 | 4 | 10 |
| ITU2 | 1 | 2.5 | 1 | 2.5 | 3 | 7.5 | 3 | 7.5 | 16 | 40   | 12 | 30   | 4 | 10 |

**Tabell 40:** Statisk oppsummering av spørsmålene knyttet til Bruksintensjon

| Spørsmål | N  | Min | Max | Mean |
|----------|----|-----|-----|------|
| ITU1     | 40 | 1   | 7   | 5.20 |
| ITU2     | 40 | 1   | 7   | 5.10 |

### 6.3.2 Statistisk analyse og resultater

Dette kapittelet inneholder vurderinger av validiteten og reliabiliteten på bakgrunn av dataene som ble innhentet gjennom spørreskjemaet under akseptansetestingen.

### 6.3.2.1 Reliabilitet og Validitet

For å teste de ulike forskningshypotesene ble en Partial Least Square (PLS)-analyse foretatt. PLS er en regresjonsbasert teknikk som har vist seg å være en god tilnærming for å studere modeller som MSAM. I denne oppgaven ble data analysert ved hjelp av den statistiske programvaren PLS 2.0. For å teste reliabiliteten til de ulike variablene i MSAM-modellen ble Internal Consistency Reliability (ICR) testet med Cronbachs alpha-koeffisient. Tabell 41 viser reabilitetskoeffisientene for hver variabel i vår modell.

**Tabell 41:** Analyse av reliabilitet for hvert tema

| Variabel | Antall spørsmål | Cronbach's alfa |
|----------|-----------------|-----------------|
| C        | 4               | 0.595           |
| ITU      | 2               | 0.979           |
| PEOU     | 3               | 0.915           |
| PU       | 3               | 0.715           |
| PIAC     | 3               | 0.683           |
| T        | 3               | 0.904           |

Cronbach's alfa-verdiene varierer mellom 0.595 og 0.979. I følge J.P. Robinson [12] er en Cronbach's alfa-verdi på 0.6 den laveste aksepterte alfa-verdien, mens PLS 2.0 opererer med en terskel på 0.7. Alfa-verdiene til de ulike variablene er over terskelen på 0.6 bortsett fra C, som er på 0.595. Den lavere reliabiliteten til C kan delvis tilskrives et høyere antall spørsmål enn hva tilfellet er for de andre variablene.

**Tabell 42:** Factor loadings, composite reliability og AVE for hvert tema

| Variabel | Spørsmål | Factor Loading | Composite Reliability | Average Variance Extracted |
|----------|----------|----------------|-----------------------|----------------------------|
| C        | C1       | -0.37          | 0.729                 | 0.487                      |
|          | C2       | 0.920          |                       |                            |
|          | C3       | 0.846          |                       |                            |
|          | C4       | 0.619          |                       |                            |
| ITU      | ITU1     | 0.989          | 0.989                 | 0.979                      |
|          | ITU2     | 0.990          |                       |                            |
| PEOU     | PEOU1    | 0.942          | 0.946                 | 0.853                      |
|          | PEOU2    | 0.930          |                       |                            |
|          | PEOU3    | 0.899          |                       |                            |
| PIAC     | PIAC1    | 0.655          | 0.840                 | 0.636                      |
|          | PIAC2    | 0.826          |                       |                            |
|          | PIAC3    | 0.845          |                       |                            |
| PU       | PU1      | 0.838          | 0.822                 | 0.608                      |
|          | PU2      | 0.817          |                       |                            |
|          | PU3      | 0.734          |                       |                            |
| T        | T1       | 0.914          | 0.939                 | 0.837                      |
|          | T2       | 0.929          |                       |                            |
|          | T3       | 0.902          |                       |                            |

Konvergent validitet ble vurdert gjennom Composite Reliability (CR), Factor Loadings (FL) for de ulike spørsmålene og Average Variance Extracted (AVE). For å måle dette har [23] foreslått følgende kriterier:

- FL for de ulike spørsmålene skal overstige 0.5
- AVE for de ulike variablene skal overstige 0.5
- CR for de ulike variablene skal overstige 0.7

Tabell 42 viser at samtlige resultater oppfyller de ovennevnte kriteriene bortsett fra FL til spørsmål C1 som er på -0.37. Grunner til dette vil bli nærmere diskutert i kapittel 7.2. Foruten FL til C1, så varierer FL for hvert av spørsmålene mellom 0.619 og 0.990, mens CR-verdiene varierer fra 0.729 til 0.989. AVE for nesten samtlige variabler oversteg 0.5 med verdier varierende mellom 0.608 og 0.979, bortsett fra C som oppnådde et resultat på 0.487. Dette kan ha en sammenheng med spørsmål C1 som også scoret et FL-resultat som ikke oversteg minimumsverdien. Dette vil bli nærmere diskutert i kapittel 7.2. Ettersom de tre verdiene av pålitelighet er over de anbefalte minimumsverdiene, med unntak for C, tilfredstiller variablene konvergens pålitelighet.



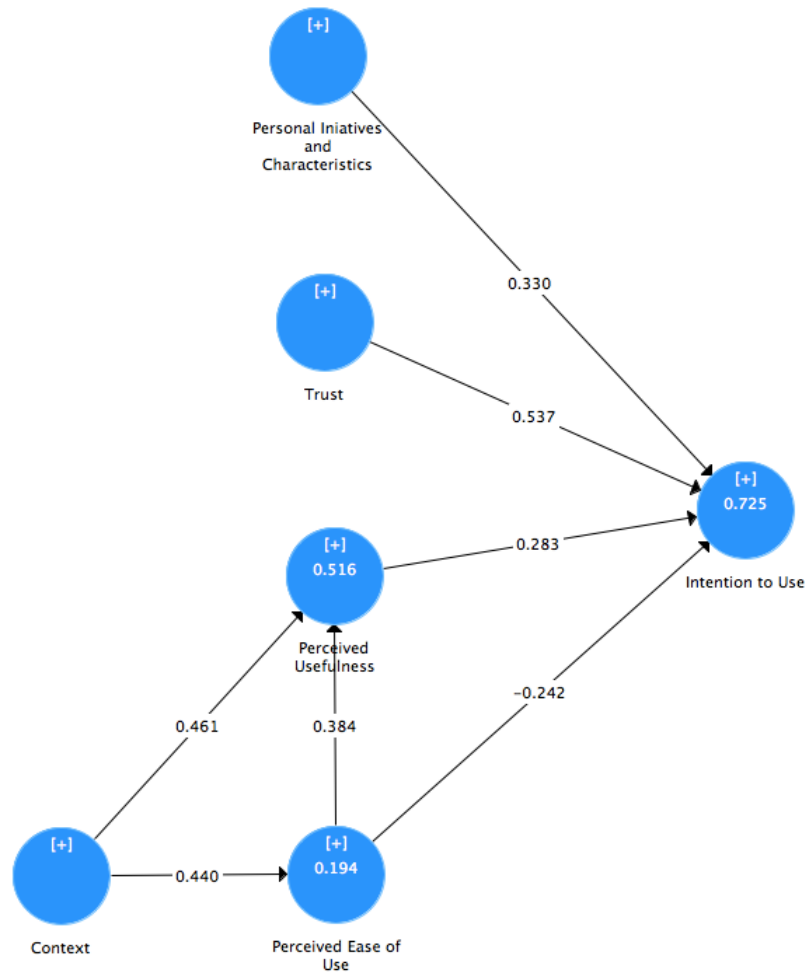
**Tabell 43:** Discriminant Validity

| Variabel | C     | ITU   | PEOU  | PU    | PIAC  | T     |
|----------|-------|-------|-------|-------|-------|-------|
| C        | 0.698 |       |       |       |       |       |
| ITU      | 0.764 | 0.990 |       |       |       |       |
| PEOU     | 0.440 | 0.229 | 0.924 |       |       |       |
| PU       | 0.630 | 0.459 | 0.587 | 0.798 |       |       |
| PIAC     | 0.687 | 0.703 | 0.608 | 0.570 | 0.780 |       |
| T        | 0.549 | 0.779 | 0.195 | 0.243 | 0.668 | 0.915 |

Tabell 43 viser målinger gjort på diskriminant validitet. Vi ser at variansen ekstrahert fra variablene er større enn kvadrat korrelasjonen på de undergående oppføringene i tabellen. På bakgrunn av dette kan vi si at de ulike variablene er empirisk distinkte. Dette er tilfellet for samtlige variabler bortsett fra C som har en lavere varians enn verdiene til de undergående oppføringene i tabellen. Bortsett fra dette kan vi si at testerresultatene av målingsmodellen er tilfredstillende da konvergent og diskriminant validitet er oppnådd.

### 6.3.2.2 Den strukturelle modellen

Figur 30 representerer den strukturelle modellen. Nummeret inne i hver sirkel står for  $R^2$  og er et nummer som indikerer hvor godt data er tilpasset en statistisk modell. Den indikerer også hvor sannsynlig fremtidige utfall kan bli spådd utifra modellen. Verdiene til de ulike stiene i figuren er sti-koeffisientene som beskriver i hvor stor grad en variabel påvirker en annen.



**Figur 30:** Den strukturelle modellen

Hypotesene, sti-koeffisientene og p-verdiene er oppsummert i tabell 44. Tabellen viser at det er kun hypotese 7 (PU → ITU) som ikke støttes. Dette på bakgrunn av at p-verdien ikke er lavere enn 0.05. Det er derimot ikke uvanlig at H7 ikke støttes i denne type modeller. Blant 101 avhandlinger som ble vurdert i [15], viste kun 58 av de en signifikant sammenheng mellom PEOU og ITU. Dette bidrar til å indikere at PEOU er en ustabil måleenhet når det kommer til å forutsi bruksintensjon. De øvrige hypotesene C → PU, C → PEOU, PEOU → PU, PIAC → ITU og PU → ITU kan alle støttes ved  $p < 0.05$ , men samtidig kan C → PU, C → PEOU og T → ITU støttes ved  $p > 0.001$ .

**Tabell 44:** Test av hypotesene basert på P-verdi og Sti koeffisient

| Hypotese        | Sti koeffisient | P Value | Hypoteseresultat |
|-----------------|-----------------|---------|------------------|
| H1:C -> PU      | 0.461           | 0.000   | Ikke avgjørbart  |
| H2: C -> PEOU   | 0.440           | 0.000   | Ikke avgjørbart  |
| H3: PEOU -> PU  | 0.384           | 0.018   | Støttet          |
| H4: PIAC -> ITU | 0.330           | 0.017   | Støttet          |
| H5: T -> ITU    | 0.537           | 0.000   | Støttet          |
| H6: PU -> ITU   | 0.283           | 0.022   | Støttet          |
| H7: PEOU -> ITU | -0.242          | 0.180   | Ikke støttet     |

## 7 Diskusjon

Dette kapittelet vil inneholde en diskusjon knyttet til de ulike forskningsspørsmålene som ble stilt i begynnelsen av denne masteroppgaven. Det er blitt foretatt tester og analyser sammen med spørreundersøkelser for å komme fram til et resultatet på følgende spørsmål:

**Spørsmål 1:** *Finnes det interesse for en applikasjon som tillater brukere å opprette arrangementer som andre personer en nødvendigvis ikke kjenner kan bli med på?*

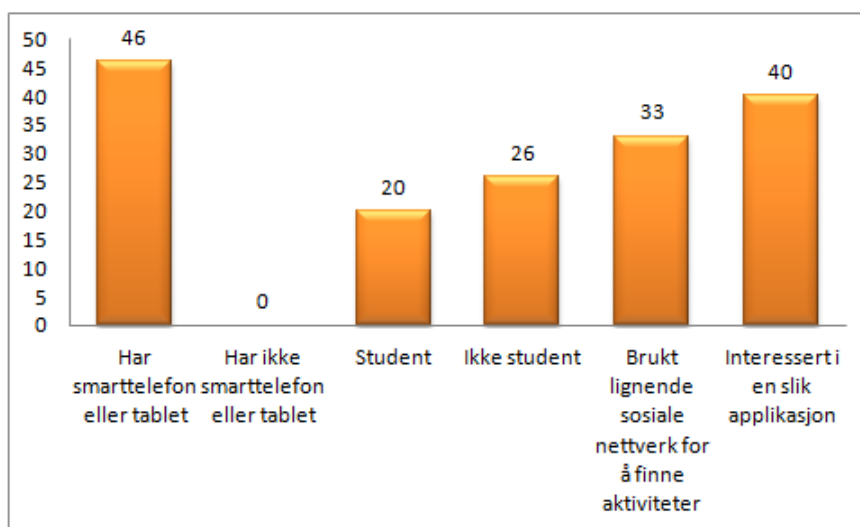
**Spørsmål 2:** *Hvis det er en interesse for en slik applikasjon, hvordan kan man oppnå brukerens aksept for å bruke en slik applikasjon?*

**Spørsmål 3:** *Hvordan skal en presentere informasjonen knyttet til en slik applikasjon på en mobil enhet slik at brukervennligheten blir høy?*

### 7.1 Interesse for en slik applikasjon

Resultatet fra nettundersøkelsen indikerte en sterk interesse for en applikasjon som gir brukeren mulighet til å opprette og delta på uformelle arrangement laget av andre. 87% av respondentene fra nettundersøkelsen sa de var interessert i å bruke en slik applikasjon. 85% av respondentene som var interessert i applikasjonen ville også brukt den selv om de ikke kjente personene som deltok på arrangementene.

Figur 31 viser noen av resultatene fra undersøkelsen som ble gjort. Det ble spurt 46 personer fra 18 til 77 år som alle hadde en smarttelefon eller tablet. Nettundersøkelsen ble utført av studenter ved NTNU og mennekser som har tilknytning til Mental Helse Trondheim. Mental Helse Trondheim har vist stor interesse for en slik applikasjon. NTNU studenter er også en stor eventuell brukergruppe og må derfor taes i betraktning under utviklingen av applikasjonen.



**Figur 31:** Respondentenes svar fra nettundersøkelsen

Mental Helse Trondheim har kommet med mange gode tilbakemeldinger og hjulpet med å informere mulige fremtidige brukere om applikasjonen. Det har blitt holdt to uformelle møter med tre representanter fra Mental Helse Trondheim. Mental Helse har kommet med direkte tilbakemeldinger på applikasjonens konsept og gitt uttrykk for funksjonalitet de savnet gjennom ustrukturerte intervjuer.

## 7.2 Oppnå aksept for å bruke en slik applikasjon

Dette kapittelet inneholder en diskusjon knyttet til resultatene av akseptansetestingen som er blitt gjennomført i denne masteroppgaven ved bruk av et MSAM-spørreskjema spesielt laget for applikasjonen som er blitt utviklet. Akseptansmodellen og dens hypoteser ble testet gjennom resultatene fra spørreskjemaet ved bruk av PLS 2.0 som beskrevet i kapittel 6.3.2. Ved å benytte seg av figur 30 i kapittel 6.3.2.2, ser vi en total varians på 72.5% når det kommer til Intention to Use, en total varians på 19.4% for Perceived Ease of Use og en total varians på 51.6% for Perceived Usefulness.

Resultatene viser at det er en positiv relasjon mellom (H3) Perceived Ease of Use og Perceived Usefulness, (H4) Personal Initiatives and Characteristics og Intention to Use, (H5) Trust og Intention to Use og (H6) Perceived Usefulness og Intention to Use. Derimot var det ingen positiv relasjon mellom (H7) Perceived Ease of Use og Intention to Use. Dette kan ha noe med den altfor høye P-verdien på 0.180, sammen med den negative stikoeffisienten på -0.242 som bidrar til at resultatet

ikke er signifikant nok og kan ikke bli brukt til å støtte opp om hypotesen.

Videre kunne ikke hypotesene (H1) Context og Perceived Usefulness og (H2) Context og Perceived Ease of Use vurderes om de hadde en positiv relasjon på bakgrunn av at Context hadde en Cronbach's alfa-verdi på 0.595. Som forklart i kapittel 6.3.2 er det anbefalt med en verdi på over 0.7, mens noen studier [12] har kommet fram til at en verdi på 0.6 også er akseptert. Dermed kunne ikke reliabiliteten til Context bekreftes. Det ble også vurdert å avrunde testresultatene til to signifikante siffer, da det ville ha resultert i at Context oppnådde en alfa-verdi på 0.6 og dermed kunne hypotesene vurderes. Det ble derimot bestemt å ikke gjøre dette.

Resultatene viste også at den sterkeste determinantent for bruksintensjon var tillit. Tillit spilte en betydelig større rolle når det kom til å forutsi brukernes adopsjon av applikasjonen enn både Perceived Usefulness og Perceived Ease of Use. Dette betyr ikke nødvendigvis at applikasjonen ikke er brukervennlig, noe resultatene av brukervennlighets-testene kan bekrefte, ei heller at applikasjonen ikke er nyttig. Det er kun en indikasjon på at Perceived Usefulness muligens ikke er en avgjørende faktor for bruksintensjon for respondentene i testen som ble gjennomført. Dette kan ses i sammenheng med hvilken type applikasjon som ble testet. Applikasjonen sitt kjernekonsept baserer seg på at mennesker som ikke nødvendigvis kjenner hverandre, skal oppmuntres til å delta på og opprette arrangementer for hverandre. Dette krever en stor andel tillit både til vedkommende som oppretter arrangementet, men også til personen som bekrefter at han eller hun ønsker å bli med. Om personen som oppretter arrangementet opplever at personer som sier ja til å bli med ikke dukker opp, vil dette være med på betvile applikasjonen sin integritet og muligens bidra til at vedkommende ikke oppretter andre arrangementer i fremtiden. Disse resultatene indikerer at tillit har en direkte positiv effekt på bruksintensjon.

En begrensning ved denne studien som kan bidra til å påvirke resultatene er den relative utvalgsstørrelsen som er noe liten. Det er blitt anbefalt med en minimum utvalgsstørrelse som er ti ganger større en antall spørsmål på den mest komplekse variabelen. Context er den mest komplekse variabelen med fire spørsmål, dermed ble det på grunn av naturen i studien og tiden som var til rådighet ikke innhentet svar fra mer enn 40 testdeltagere. Dette kan være noe av grunnen til at blant annet Context oppnådde en for lav Cronbach's alfa-verdi til at reliabiliteten kunne bekreftes, da det er mulig at useriøse respondenter svarer svært negativt

på spørreundersøkelsen av ulike grunner. Dette kan bli mindre utslagsgivende om utvalgsstørrelsen blir større og kunne ha ført til at reabiliteten til Context hadde vært gyldig og ytterligere hypoteser kunne ha blitt mer signifikante.

### 7.2.1 Oppnå høy brukervennlighet

Resultatene fra SUS-testen viste at alle spørsmålene knyttet til testen oppnådde en høyere poengsum enn gjennomsnittet som er 2. Spørsmålene som skilte seg ut på en positiv måte ved å ha en høyere poengsum enn gjennomsnittet for den aktuelle testen (2.82) er:

- Spørsmål 7 ("Jeg tror de fleste vil lære å bruke dette systemet svært raskt") med en poengsum på 3.2
- Spørsmål 9 ("Jeg følte meg veldig sikker ved bruk av systemet") med en poengsum på 3.2
- Spørsmål 10 ("Jeg vil være avhengig av å lære meg masse før jeg kan begynne å bruke dette systemet") med en poengsum på 3.4

Applikasjonen oppnådde en poengsum på 75 av 100 poeng noe som gir en indikasjon på at brukerne som testet applikasjonen opplevde brukervennligheten som god og over gjennomsnittet. SUS-testen ble derimot utført mens applikasjonen fremdeles var i et tidlig prototype-stadiet, med oppdiktet data og uten deler av funksjonaliteten som den skulle inneha. Dermed er det mulig å forbedre denne poengsummen ytterligere. Derimot er det viktig å brukerteste en applikasjon tidlig i utviklingsfasen slik at eventuelle forbedringer og endringer kan bli gjort forløpende. Den ferdigutviklede prototypen har blitt tilført ytterligere funksjonalitet som er blitt foreslått av eventuelle sluttbrukere av denne typen applikasjon.

Det er også blitt utført en heuristisk evaluering av brukergrensesnittet og funnene av denne evalueringen kan bli funnet i tabell 26 under kapittel 6.2.1. Nielsens andre heuristikk var en svakhet ved brukergrensesnittet som ble avdekket under testperioden. Det var enkelte designelementer som bidro til at testdeltagerne ble usikre. Slike situasjoner er ikke fordelaktige da de bidrar til misforståelser rundt bruken av applikasjonen. En av endringene som ble gjort er at etiketten som beskrev hvor mange deltakere som allerede har meldt seg på et arrangement og maks antall tillatte deltagere for arrangementet ble endret fra "2/4" til "2 av 4". Dette siden enkelte av testdeltagerne ikke var sikker på hva "2/4" inne på et arrangement stod

for. Andre endringer som ble gjort er blant annet knappen “Setting”, som tillot brukere å endre sine interesser, endret navn til “Interesser”. Etiketten “Dine Eventer” som viste hvor brukeren sine opprettede arrangementer var på profil-siden er blitt endret til “Mine Eventer”.

Et annet element som bød på utfordringer for testdeltagerne var hjelpeskjermen som kommer til syne første gangen applikasjonen blir startet opp. De var usikre på hvordan de skulle navigere i hjelpeskjermen med det første da den ikke reagerer på tastetrykk, men heller horisontale sveipebevegelser for å komme seg til neste skjerm bilde. Dette til tross for at det er et navigeringselement på bunnen av hjelpeskjermen som forteller brukeren hvilket skjerm bilde de er på og hvor mange som er igjen.

Andre endringer som er blitt gjort på brukergrensesnittet som et direkte resultat av den herustiske evalueringen er blant annet at en bekreftelse blir presentert for brukeren om vedkommende ønsker å slette et allerede opprettet arrangement. Knappen “Registrer” som er å finne på hovedsiden og hvis funksjon er å opprette et nytt arrangement har byttet navn til “Opprett arrangement”. Dette på grunn av at enkelte av testdeltagerne var usikre på hvorvidt “Registrer” betydde å registrere en ny bruker eller registrere et nytt arrangement. Det er også blitt plassert en placeholder i tekstfeltet “Beskrivelse” når en bruker skal opprette et nytt arrangement slik at brukeren får beskjed om at en liten beskrivelse om arrangementet skal skrives inn der.



## 8 Konklusjon og videre arbeid

Dette kapitlet inneholder en konklusjon på de ulike forskningsspørsmålene som er blitt evaluert i forhold til resultatene, sammen med elementer som det kan arbeides videre med.

### 8.1 Konklusjon

Denne masteroppgaven har undersøkt mulige faktorer som vil være avgjørende for bruksintensjonen av en ny applikasjon som tillater brukere å opprette arrangementer som ukjente mennesker kan velge å bli med på. Brukervennligheten har vært i fokus da den potensielle brukermassen kan være av svært forskjellig karakter og med ulike erfaringer med smarttelefoner og mobile applikasjoner. En artifakt i form av en mobil applikasjon ble utviklet og det ble benyttet ulike metoder og verktøy for å samle inn data og analysere resultatet. Tre forskningsspørsmål har blitt satt fokus på i denne prosessen:

**Spørsmål 1:** *Finnes det interesse for en applikasjon som tillater brukere å opprette arrangementer som andre personer en nødvendigvis ikke kjenner kan bli med på?*

**Spørsmål 2:** *Hvis det er en interesse for en slik applikasjon, hvordan kan man oppnå brukerenes aksept for å bruke en slik applikasjon?*

**Spørsmål 3:** *Hvordan skal en presentere informasjonen knyttet til en slik applikasjon på en mobil enhet slik at brukervennligheten blir høy?*

Resultatet fra kapittel 7.1 konkluderer med at det er en interesse for en applikasjon hvor brukerne kan lage og bli med på uformelle arrangement. Kapittel 7.2 oppsummerer at brukere er interessert i å bruke en slik applikasjon og at variabler som “perceived usefulness”, “personal characteristics and initiatives” og “trust” er viktige faktorer for brukere sin intensjon til å benytte seg av en slik mobil applikasjon. Ved å benytte potensielle sluttbrukere av applikasjonen tidlig i utviklingsfasen, har det vært mulig å analysere ulike problemer og utfordringer knyttet til brukervennlighet og det avgjørende resultatet. Velkjente evalueringsmetoder og retningslinjer knyttet til brukervennlighet har også blitt brukt for å utviklet et intuitivt brukergrensesnitt.

## 8.2 Videre arbeid

Implementasjon av et ratingsystem som kan tillate brukere å vurdere andre brukere av applikasjonen om det oppstår uhyggelige situasjoner, om de oppretter arrangementer som ikke er slik de blir beskrevet eller om de ikke møter opp på avtalte arrangementer. Akseptansetestingen viste at tillit var en viktig faktor for testdeltakerne og dermed vil et slikt ratingsystem bidra til å øke tillitsforholdet brukerne har til hverandre og applikasjonen.

Algoritmen som bestemmer hvilke arrangementer som blir presentert for en bruker kan bli bedre optimalisert når antall arrangementer begynner å øke. Per juni 2015 er det kun arrangementene som samsvarer med interessene, alder og kjønn til brukeren som blir lastet inn på applikasjonen. Dette kan bli forbedret ved å for det første ta hensyn til hvor mange arrangementer som skal innledningsvis lastes inn i applikasjonen. Om det er et stort antall arrangementer tilgjengelig i databasen, kan det oppstå minneproblemer om hver eneste arrangement blir lastet inn. Dette kan løses ved å sortere de aktuelle arrangementene til å passe inn med brukerens lokasjon.

På bakgrunn av tiden som var til rådighet ble ikke iOS-versjonen av applikasjonen slutført i tide til innleveringsfristen. Kjernefunksjonaliteten er blitt implementert, men enkelte elementer av brukergrensesnittet ble ikke implementert i tide. Kildekoden som er blitt skrevet for plattformspesifikke iOS-funksjoner ligger i digitalt vedlegg 2.

Muligheter for å se arrangementer som er i nærheten av brukeren sin lokasjon er ikke blitt implementert i prototypen. Dette er en forbedring som vil bidra til å gi brukere mer relevante søkeresultater i tillegg til kjønn, alder og interesser. Kildekode for å hente ut koordinater fra en adresse, hente ut adresse fra et sett med koordinater og hente ut lokasjonen til en mobil enhet for både iOS og Android er blitt utviklet og er å finne i digitalt vedlegg 2.

## A Nettundersøkelse

# Ny App - Aktivitetsportal

\*Må fylles ut

### Q1: Kjønn \*

- Mann
- Kvinne

### Q2: Alder \*

### Q3: Går du på skole? \*

- Ja
- Nei

### Q4: Hvis ja på spørsmål 3, hvilket trinn?

- Ungdomskole
- Videregående
- Universitet/Høyskole
- Andre (PhD etc)

Fortsett

# Ny App - Aktivitetsportal

\*Må fylles ut

## Ny App - Aktivitetsportal

**Q5: Eier du en smarttelefon eller en tablet? \***

E.g. iPhone, Android-telefon, iPad, Android-tablet

- Ja
- Nei

**Q6: Hvis ja på Q5, hvor mye tid bruker du på telefonen/tableten din hver dag? \***

- Mindre enn 1 time
- 1-2 timer
- 3-5 timer
- Mer enn 5 timer

**Q7: Har du brukt Facebook eller lignende sider for å prøve å finne ulike aktiviteter å gjøre alene eller med andre på fritiden din? \***

- Ja
- Nei

**Q8: Hvis ja på Q7, var det vanskelig å finne relevante aktiviteter å gjøre alene eller med andre? \***

- Ja
- Nei

**Q9: Om det fantes en enkel mobilapp som kunne vise deg ulike aktiviteter og eventer som du kunne gjøre sammen med andre der du bor, hadde du ønsket å bruke den? \***

- Ja
- Nei

**Q9.1: Hvis ja på Q9, ville du ha brukt den selv om du ikke kjente personene som skulle på en aktivitet personlig?**

- Ja
- Nei

**Q10: Hvor nyttig tror du en app som kunne vise deg alle de ulike aktivitetene som du og/eller vennene dine kunne bli med på? \***

- Veldig nyttig
- Litt nyttig
- Ikke så veldig nyttig
- Unyttig

**Q11: Hva syns du om noen av disse aktuelle funksjonene en slik app kan ha? \***

|  | Helt unyttig          | Ikke veldig nyttig    | Nyttig                | Veldig nyttig         | Ikke sikker           |
|--|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|
| Brukere som blir med på samme aktivitet skal ha mulighet til å chatte med hverandre                      | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| Brukere skal ha muligheten for å enkelt logge seg inn med Facebook                                       | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| Brukere bør få en notifikasjon når en aktivitet som de sannsynligvis er interessert i er blitt opprettet | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| Hvis bare ulike aktiviteter som er i nærheten av deg   | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |

**Q12: Er det andre funksjoner du skulle ønske en slik app hadde?**

Tilbake

Send

Send aldri passord via Google Skjemaer.



## B Spørreskjema om brukervennlighet

### Spørsmål til testdeltaker

\*Må fylles ut

**Kjønn? \***

- Mann  
 Kvinne

**Alder \***

**Eier du en smarttelefon eller en tablet? \***

- Ja  
 Nei

**Hvor mange timer om dagen benytter du deg av den? \***

- Mindre enn 1 time  
 1-2 timer  
 3-5 timer  
 Mer enn 5 timer

**Hvor mye erfaring har du med bruk av smarttelefoner/tablet? \***

Skalaen går fra 1 (Nybegynner) til 5 (Ekspert)

- 1  
 2  
 3  
 4  
 5

**Benytter du deg av Facebook eller andre sosiale medier for å finne aktiviteter å bedrive med på fritiden din? \***

- Ja  
 Nei

**Hvis svaret er nei på forrige spørsmål, hvorfor ikke?**

Fortsett

# Spørsmål til testdeltaker

\*Må fylles ut

## Avslutningsspørsmål

Savner du noe funksjonalitet?

Syns du appen viser nok detaljert informasjon om arrangementene? \*

- Ja  
 Nei

Hvis nei på forrige spørsmål, hva kan bli bedre?

Savner du noe informasjon? \*

- Ja  
 Nei

Hvis nei på forrige spørsmål, hva slags informasjon savner du?



Er det noen designendringer du skulle ønske ble gjort? \*

- Ja
- Nei

Hvis ja, forklar hvilke endringer du skulle ønske ble gjort?

Ville du brukt denne appen når den er ferdig? \*

- Ja
- Nei

Hvis nei på forrige spørsmål, hvorfor ikke?

Tilbake

Send

Send aldri passord via Google Skjemaer.

Drevet av  
 Google Forms

Dette innholdet er ikke laget eller godkjent av Google.

[Rapporter misbruk](#) - [Vilkår for bruk](#) - [Ytterligere vilkår](#)

## C Spørreskjema MSAM

---

# Mobile Service Acceptance Model 1

### Introduksjon

Tenk deg at du er en person som sitter hjemme, på skolen eller hvor enn du skulle ønske du var i verden. Du ønsker å gjøre noe. Spille fotball, stikke på kafe eller gå en tur med noen. Av ulike grunner så har du ingen i din nærmeste omgangskrets som kan bli med deg og du bruker en app på telefonen din som viser ulike aktiviteter du kan bli med på, opprettet av personer i samme situasjon som deg.

Oppgave 1: Anta at du ønsker å spille fotball. Finne en aktivitet hvor du kan bli med og spille fotball. Hvor mange deltar på det arrangementet?

Oppgave 2: Opprett et arrangement ved navn «Grilling på Kalvskinnet». Tillatt at begge kjønn får lov til å bli med og maks 30 personer fra alderen 18 og oppover.

Oppgave 3: Meld deg på et arrangement som heter Kents opplevelse og bli med i chatten. Hva er den siste meldingen som er blitt skrevet i chatten på dette arrangementet?

\*Må fylles ut

### Kjønn? \*

- Mann  
 Kvinne

### Alder? \*

### Studerer du? \*

- Ja  
 Nei

### Hvor mange timer om dagen bruker du en smarttelefon eller en tablet?

- Mindre enn 1 time
- 1-2 timer
- 3-5 timer
- Mer enn 5 timer

### Hvor erfaren er du med bruk av en smarttelefon eller en tablet?

1 2 3 4 5

Nybegynner      Ekspert

### Bruker du sosiale medier eller andre apper for å finne arrangementer som du kan bli med på?\*

- Ja
- Nei

### Hvis nei på forrige spørsmål, hvorfor ikke?

### Hvis ja på forrige spørsmål, hvilken app eller sosialt medie bruker du?

Fortsett

# Mobile Service Acceptance Model 1

\*Må fylles ut

## Opplevd nytteverdi

Jeg finner raskt og enkelt en happening å være med på?\*

1 2 3 4 5 6 7

Sterkt uenig        Sterkt enig

Det er større sannsynlighet for at jeg finner arrangementer jeg ikke visste eksisterte?\*

1 2 3 4 5 6 7

Sterkt uenig        Sterkt enig

Det er enkelt å lage arrangementer som jeg ønsker at andre skal få være med på?\*

1 2 3 4 5 6 7

Sterkt uenig        Sterkt enig

Tilbake

Fortsett

Drevet av



Dette innholdet er ikke laget eller godkjent av Google.

[Rapporter misbruk](#) - [Vilkår for bruk](#) - [Ytterligere vilkår](#)

# Mobile Service Acceptance Model 1

\*Må fylles ut

## Opplevd brukervennlighet

Jeg synes appen er forståelig \*

1 2 3 4 5 6 7

Sterkt uenig        Sterkt enig

Bruk av appen krever lite konsentrasjon \*

1 2 3 4 5 6 7

Sterkt uenig        Sterkt enig

Jeg synes det er enkelt å få appen til å gjøre det jeg vil \*

1 2 3 4 5 6 7

Sterkt uenig        Sterkt enig

Tilbake

Fortsett

Drevet av

 Google Forms

Dette innholdet er ikke laget eller godkjent av Google.

[Rapporter misbruk](#) - [Vilkår for bruk](#) - [Ytterligere vilkår](#)

# Mobile Service Acceptance Model 1

\*Må fylles ut

## Tillit

Jeg vil bruke appen hvis jeg har en klar oppfatning av all funksjonalitet appen innehar \*

1 2 3 4 5 6 7

Sterkt uenig        Sterkt enig

Jeg vil bruke appen hvis den beskytter brukerens personvern \*

1 2 3 4 5 6 7

Sterkt uenig        Sterkt enig


Jeg vil bruke appen hvis jeg føler at data fra appen er til å stole på \*

1 2 3 4 5 6 7

Sterkt uenig        Sterkt enig

Tilbake

Fortsett

Drevet av  
 Google Forms

Dette innholdet er ikke laget eller godkjent av Google.  
[Rapporter misbruk](#) - [Vilkår for bruk](#) - [Ytterligere vilkår](#)

# Mobile Service Acceptance Model 1

\*Må fylles ut

## Personlig initiativ

Jeg er i stand til å bruke denne appen \*

1 2 3 4 5 6 7

Sterkt uenig        Sterkt enig

Jeg vil gjerne være blant de første til å bruke appen \*

1 2 3 4 5 6 7

Sterkt uenig        Sterkt enig

Bruk av denne appen er en god ide \*

1 2 3 4 5 6 7

Sterkt uenig        Sterkt enig

Tilbake

Fortsett

Drevet av



Dette innholdet er ikke laget eller godkjent av Google.

[Rapporter misbruk](#) - [Vilkår for bruk](#) - [Ytterligere vilkår](#)

---

# Mobile Service Acceptance Model 1

\*Må fylles ut

## Kontekst

**Jeg vil bruke denne appen når jeg ønsker å finne på noe, men mangler en aktivitet å gjennomføre \***

1 2 3 4 5 6 7

Sterkt uenig        Sterkt enig

**Jeg vil bruke denne appen når jeg mangler personer for å gjennomføre en aktivitet eller et arrangement \***

1 2 3 4 5 6 7

Sterkt uenig        Sterkt enig

**Jeg vil bruke denne appen hvis den finnes til mobil når det er enklest for meg å bruke mobilen \***

1 2 3 4 5 6 7

Sterkt uenig        Sterkt enig

**Jeg vil bruke denne appen hvis den finnes til PC når det er enklest for meg å bruke PC \***

1 2 3 4 5 6 7

Sterkt uenig        Sterkt enig

Tilbake

Fortsett



# Mobile Service Acceptance Model 1

\*Må fylles ut

## Bruksintensjon

Jeg har lyst til å bruke appen hvis jeg får tilgang til den \*

1 2 3 4 5 6 7

Sterkt uenig        Sterkt enig

Jeg kommer til å bruke appen hvis jeg får tilgang til den \*

1 2 3 4 5 6 7

Sterkt uenig        Sterkt enig

Tilbake

Send

Send aldri passord via Google Skjemaer.

Drevet av  
 Google Forms

Dette innholdet er ikke laget eller godkjent av Google.

[Rapporter misbruk](#) - [Vilkår for bruk](#) - [Ytterligere vilkår](#)

## D Spørreskjema SUS

|  | Strongly disagree        |                          |                          |                          |                          | Strongly agree |
|--|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|----------------|
| 1. I think that I would like to use this system frequently                                   | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |                |
|  | 1                        | 2                        | 3                        | 4                        | 5                        |                |
| 2. I found the system unnecessarily complex  | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |                |
|  | 1                        | 2                        | 3                        | 4                        | 5                        |                |
| 3. I thought the system was easy to use  | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |                |
|  | 1                        | 2                        | 3                        | 4                        | 5                        |                |
| 4. I think that I would need the support of a technical person to be able to use this system | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |                |
|  | 1                        | 2                        | 3                        | 4                        | 5                        |                |
| 5. I found the various functions in this system were well integrated                         | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |                |
|  | 1                        | 2                        | 3                        | 4                        | 5                        |                |
| 6. I thought there was too much inconsistency in this system                                 | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |                |
|  | 1                        | 2                        | 3                        | 4                        | 5                        |                |
| 7. I would imagine that most people would learn to use this system very quickly              | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |                |
|  | 1                        | 2                        | 3                        | 4                        | 5                        |                |
| 8. I found the system very cumbersome to use   | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |                |
|  | 1                        | 2                        | 3                        | 4                        | 5                        |                |
| 9. I felt very confident using the system  | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |                |
|  | 1                        | 2                        | 3                        | 4                        | 5                        |                |
| 10. I needed to learn a lot of things before I could get going with this system              | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |                |
|  | 1                        | 2                        | 3                        | 4                        | 5                        |                |

## Kilder

- [1] Alexa. How popular is vk.com? Tilgjengelig på <http://www.alexa.com/siteinfo/vk.com>. Accessed 16/04-15.
- [2] John E. Anderson and Paul H. Schwager. SME Adoption Of Wireless LAN Technology: Applying The UTAUT Model. pages 1–5, 2004.
- [3] Cios. Theory of reasoned action. Tilgjengelig på [http://www.cios.org/encyclopedia/persuasion/Gtheory\\_1reasoned.htm](http://www.cios.org/encyclopedia/persuasion/Gtheory_1reasoned.htm). Accessed 20/04-15.
- [4] Derek Coleman. A use case temmplate. Tilgjengelig på [http://www.bredemeyer.com/pdf\\_files/use\\_case.pdf](http://www.bredemeyer.com/pdf_files/use_case.pdf). Accessed 14/05-15.
- [5] eBizMBA. Top 15 most popular social networking sites. Tilgjengelig på <http://www.ebizmba.com/articles/social-networking-websites>. Accessed 02/04-15.
- [6] Google. App engine service level agreement. Tilgjengelig på <https://cloud.google.com/appengine/sla>. Accessed 17/04-15.
- [7] Google. Autoscaler. Tilgjengelig på <https://cloud.google.com/compute/docs/autoscaler/>. Accessed 17/04-15.
- [8] Google. Pricing. Tilgjengelig på <https://cloud.google.com/pricing/>. Accessed 17/04-15.
- [9] Alan R. Hevner. Design Science in Information System Research. pages 1–32, Mars 2004.
- [10] Ecma International. C# Language Specification 4th edition. 2006.
- [11] Cory Janssen. Use case. Tilgjengelig på <http://www.techopedia.com/definition/25813/use-case>. Accessed 14/05-15.
- [12] P.R. Shaver J.P. Robinson and L.S. Wrightsman. Criteria for scale selections and evaluation, Academic Press, San Diego, CA. 1991.
- [13] Røde Kors. Jeg er ensom bryr meg. Tilgjengelig på [https://www.rodekors.no/distriktsider/oppland\\_rode\\_kors/ungdom1/jeg-er-ensom-bryr-meg/](https://www.rodekors.no/distriktsider/oppland_rode_kors/ungdom1/jeg-er-ensom-bryr-meg/). Accessed 10/02-15.
- [14] Craig Larman. Applying UML and patterns 3rd edition. pages 89–100, 2009.

- [15] K.A. Kozar Lee, Y. and K.R. Larsen. The technology acceptance model: Past, present, and future. *Communications of the Association for information systems*. page 50, 2003.
- [16] Abraham H. Maslow. A theory of human motivation. Tilgjengelig på <http://www.altruists.org/f62>. Accessed 12/02-15.
- [17] Meetup. Fairer meetup price plans. Tilgjengelig på <http://blog.meetup.com/fairer-meetup-price-plans/>. Accessed 02/03-15.
- [18] Jakob Nielsen. How to conduct a heuristic evaluation. Tilgjengelig på <http://www.nngroup.com/articles/how-to-conduct-a-heuristic-evaluation/>. Accessed 14/05-15.
- [19] Briony J. Oates. *Researching Information Systems and Computing*. pages 43–51, 108–111, 186–196, 202–212, 219–228, 236–241, 245–277, 2006.
- [20] University of Geneva. Technology acceptance model. Tilgjengelig på [http://edutechwiki.unige.ch/en/Technology\\_acceptance\\_model](http://edutechwiki.unige.ch/en/Technology_acceptance_model). Accessed 23/04-15.
- [21] Boston University School of Public Health. Diffusion of innovation theory. Tilgjengelig på <http://sphweb.bumc.bu.edu/otlt/MPH-Modules/SB/SB721-Models/SB721-Models4.html>. Accessed 21/04-15.
- [22] Boston University School of Public Health. The theory of planned behavior. Tilgjengelig på <http://sphweb.bumc.bu.edu/otlt/MPH-Modules/SB/SB721-Models/SB721-Models3.html>. Accessed 21/04-15.
- [23] Bagozzi RP and Yi Y. Specification, evaluation, and interpretation of structural equation models. *Journal of the Academy of Marketing Science* 40, pages 8–34, 2012.
- [24] J. Krogstie S. Andresen and T. Jelle. Lab and research activities at Wireless Trondheim. In *Wireless Communication Systems*. pages 4th International Symposium on pages 385–389, 2007.
- [25] Jeff Sauro. Measuring usability with the system usability scale(sus). Tilgjengelig på <http://www.measuringu.com/sus.php>. Accessed 15/05-15.
- [26] Ben Schneiderman and Catherine Plaisant. *Designing the User Interface*. Pearson, 5th edition. pages 80–84, 88–91, 156–183, 2010.

- [27] John Krogstie Shang Gao and Per Anton Gransæther. Mobile Services Acceptance Model. pages 446–453, 2008.
- [28] usability.gov. System usability scale(sus). Tilgjengelig på <http://www.usability.gov/how-to-and-tools/methods/system-usability-scale.html>. Accessed 15/05-15.
- [29] Bill Kuechler Vijay Vaishnavi. Design science research in information systems. Tilgjengelig på <http://desrist.org/desrist/content/design-science-research-in-information-systems.pdf>. Accessed 08/03-15.
- [30] Xamarin. Om xamarin. Tilgjengelig på <https://xamarin.com/about>. Accessed 16/04-15.