

# Tekisinkö sovelluksen? 9 vinkkiä järjestöille sovelluskehityksen tueksi



Materiaali on osa Eläkeliitto ry:n ja EHYT ry:n yhteisen  
LähiVerkko-projektin materiaalityöntantaa.  
Kiitämme kaikkia oppaan kommentointiin osallistuneita!

Teksti: Maarit Nyqvist ja Marja Pakarinen

Kuvat: LähiVerkko-projekti

Graafinen suunnittelu: Sopiva Design

Ilmestymisvuosi 2018

© Eläkeliitto

# Johdanto

Älypuhelinien yleistyttyä mobiilisovellukset ovat tulleet luonnolliseksi osaksi ihmisten arjen toimintaa ja vuorovaikutusta. Sovellukset eli appsit ovat tätä päivää – myös meille järjestöille. Sovelluksilla pyritään vastaamaan erilaisiin tarpeisiin: ihmisten tavoittamiseen tai auttamiseen, yhteydenpitoon jäsenjärjestöjen kanssa tai järjestön oman sanoman eteenpäin viemiseen. Jotkut sovellukset on tehty vain viihtymistä varten. Moni järjestö, joka haluaa kokea kuuluvansa edelläkävijöihin ja pysyä mukana digitalisaatiossa, pohtii sovelluksen tekemistä. Osa järjestöistä on jo ottanut sovellukset osaksi toimintaansa. Vuoden 2016 lopulla järjestöjen mobiilisovelluksia löytyi markkinoilta noin 30 (Toikkanen 2016).

Sovelluskehitys on oma maailmansa, josta on tullut miljardiluokan liiketoimintaa. Miten järjestöt voisivat hyödyntää mobiiliteknologiaa toiminnassaan? Mitkä ovat ensi askeleet ideasta sovellukseen? Mitä kaikkea pitäisi ottaa huomioon? Millainen sovellus voisi menestyä, kun sovelluskaupat pursuavat jo valmiiksi monenlaisiin tarpeisiin tehtyjä tuotteita?

Eläkeliiton ja Ehkäisevä päihdetyö EHYT ry:n LähiVerkko-projektissa käynnistettiin +60-nimisen mobiilisovelluksen kehittäminen vuoden 2014 lopulla. +60-sovellus suunnattiin ikäihmisille, ja sen tarkoitus oli mahdollistaa käyttäjän yhteydenpito tuttuun lähipiiriin sekä tutustuttaa käyttäjä uusiin ihmisiin omassa lähiympäristössä. Matka ideasta valmiiseen sovellukseen oli haastava, mutta samalla täynnä oppimisen kokemuksia ja oivalluksia. Vastaan tuli monia kysymyksiä, joita emme osanneet edes ajatella lähtiessämme ideoimaan sovellusta. Tässä oppaassa haluamme jakaa kokemuksiamme ja havaintojamme ja näin auttaa toisia järjestöjä omalla matkallaan sovellusten maailmassa. Opas on jaettu yhdeksään kysymykseen, joihin vastaamalla lukija voi pohtia, mitä suunnittelussa, budjetoinnissa, viestinnässä ja ideoinnissa on hyvä ottaa huomioon.

# 9 kysymystä sovelluksen kehittämisen tueksi

1. Miksi teemme sovelluksen ja kenelle? Tarvekartoitus
2. Sovelluksen ominaisuudet? Ideasta suunnitelmaan
3. Miten sovelluksen pitäisi toimia? Tekniset vaatimukset
4. Mitä sovelluksen tekeminen maksaa? Budjetointi
5. Ketkä kehittävät sovelluksen? Kilpailutus ja sopimus
6. Miten kohderyhmä otetaan mukaan? Testaus
7. Miten sovellukselle saadaan käyttäjiä? Tiedottaminen ja markkinointi
8. Millainen on sovelluksen elinkaari? Kehittämisen jatkuva prosessi
9. Mitä vielä pitäisi ottaa huomioon? Sovelluksen ylläpito

Lopuksi

# 1. Miksi teemme sovelluksen ja kenelle? Tarvekartoitus

*Syksyllä 2014 ryhdyimme miettimään, olisiko nyt aika tehdä sovellus. Siitähän oli ollut puhetta jo projektin alusta asti. Kohderyhmänämme ovat ikäihmiset, eikä heille suunnattuja sovelluksia ole juuri olemassa – varsinkaan suomenkielisiä. Sovellus voisi vastata projektin tavoitteisiin, yhteydenpidon lisäämiseen ja teknologian mahdollisuuksien välittämiseen. Aloitimme sovelluksen suunnittelun, ja sen nimeksi tuli ”+60”.*

Sovelluksen suunnitteleminen on haastava prosessi, ja päätöstä sovelluksen tekemisestä kannattaa harkita tarkkaan ja monista näkökulmista. Tarpeen huolellinen pohtiminen on keskeinen vaihe sovelluksen kehittämisessä. Hyvin suunniteltu teknologia lähtee ihmisen aidosta tarpeesta. *Elämälähtöisen suunnittelun* (Life based design, LBD) ideana on selvittää, mikä on teknologian rooli ihmisen elämässä. Millaisia tarpeita ihminen pyrkii elämässään tyydyttämään? Aidossa tarvelähtöisessä suunnittelussa ratkaisuja ei välttämättä ole vielä olemassa, vaan ne luodaan. Elämälähtöiseen suunnitteluun kuuluu ihmisen fyysisten ja psyykkisten ominaisuuksien määrittelyä, elämänpiirin ja -muodon kartoittamista ja sosiokulttuurisen aseman pohtimista (Saariluoma ym. 2010; Leikas 2010).

Millä tavalla sovelluksen tarvetta voisi sitten selvittää? Omassa projektissa huomasimme, että tarvetta kannattaa pohtia monesta näkökulmasta, joita kaikkia ei tule edes alkuun ajatelleeksi. Kokosimme tähän muutamia vinkkejä näkökulmiin:

- Tutkikaa sovelluskauppojen valikoimaa ja miettikää, ratkaiseeko jo jokin olemassa oleva sovellus tarpeen. Onko mobiilisovellus juuri oikea väline tarpeen ratkaisemiseksi? On hyvä tuntea eri sovelluksia ja sitä, miten kohderyhmänne kuuluvat ihmiset käyttävät niitä. Onko joku kustannustehokkaampi tapa ratkaista tarve? Tällainen voi olla esimerkiksi responsiivinen mobiilisivusto. Nykyään ihmiset ovat tarkkoja siitä, mitä sovelluksia lataavat puhelimiinsa. Moni käyttää mieluummin verkkosivua kuin lataa harvoin käytetyn sovelluksen, joka vie puhelimesta muistitilaa.

- Määrittäkää tarkkaan kohderyhmä, jolle teette sovellusta. Kuka sovellusta käyttää, ja mitä hyötyä siitä on juuri hänelle? Minkä tarpeen sovellus tyydyttää? Tuottaako se lisäarvoa ihmisen elämään? Miksi hän lataisi sovelluksen? Millä muulla tavalla hän yrittää ratkaista tarvettaan, tai miten se tällä hetkellä ratkaistaan? Miksi hän hylkäisi nykyisen keinonsa ja ottaisi sovelluksen käyttöön?
- Onko kohderyhmällä käytössään mobiililaitteita, internetiä ja sovelluksia, ja voivatko he ylipäätään käyttää sovelluksia? Mitä sovelluksen käyttäminen vaatii kohderyhmänne näkökulmasta? Kuinka käyttäjäryhmä otetaan mukaan suunnitteluun? Onko tarve lähtöisin käyttäjiltä vai järjestöltä?
- Millaista käyttäjämäärää tavoitellaan? Tarvitseeko sovellus tietyn määrän ja tietyllä alueella olevia ihmisiä toimiakseen (vrt. esim. sosiaalisen median sovellukset). Mitä teiltä järjestönä vaatii, että saatte riittävän määrän ihmisiä käyttämään sovellusta?
- Mitä tarpeita organisaatiollanne on, ja mitä arvoa sovellus tuottaa järjestölle? Sovellusten kehittäminen ja ylläpito vaativat resursseja, niin ajallisia kuin rahallisia. Millaisia resursseja sovellus vaatii? Pitääkö sovellukseen tuottaa ajankohtaista sisältöä, kuinka usein sitä tuotetaan ja kuka tästä on vastuussa? Tarvitseeko jonkun vastata sovelluksen kautta tulleisiin kysymyksiin? Lisääkö sovellus resurssien tarvetta vai vähentääkö sitä? Sovellukset, joissa autetaan ihmisiä eli vaikkapa chat-tyyppiset palvelut, vaativat joustavaa resursointia järjestöltä.

## 2. Sovelluksen ominaisuudet? Ideasta suunnitelmaan

*Korostimme alusta asti +60:n tulevaa kohderyhmää. Näimme kuvia tulevasta sovelluksesta ja saimme antaa arvioita. Halusimme enemmän kontt-rastia teksteihin. Muutamia toiminnallisuuksia poistettiin, mikä paransi käytettävyyttä. Muutoin annoimme vapaat kädet sovellusta tekeville yritykselle.*

Ennen kuin sovellusta lähdetään suunnittelemaan toteuttajaksi valitun yrityksen kanssa, on hyvä tietää jotain teknologian suunnittelusta. Tällöin pystyy paremmin arvioimaan tehtyjä ratkaisuja kriittisesti. Määrittäkää, mitä sovellus konkreettisesti tekee, mikä on sen päätoiminnallisuus. Voitte tehdä sovelluksesta alkuun yksinkertaisen paperimallin ja piirtää valikoita: mistä pääsee minne, tarvitseeko rekisteröityä, miltä sovellus voisi näyttää, onko kuvakkeita vai tekstiä, mistä saa ohjeita, millaisia erilaisia näyttökuvia tarvitaan, onko vaatimuksia logoille tai väreille jne. Tällä tavalla hahmottuu paremmin, miten sovellusta käytetään, ja ideaa on helpompi esitellä suunnittelijalle. Paperimallia kannattaa esitellä myös kohderyhmälle ja pyytää siitä kommentteja. Voitte miettiä sovelluksellenne nimeä jo etukäteen. Sovelluksen nimeä tarvitaan jo varsin aikaisessa vaiheessa, joten siitä on hyvä olla ideoita olemassa. Pyytäkää suunnittelijaa esittämään teille kuvia suunnittelun edetessä.

Itse opimme sen, että liikkeelle kannattaa lähteä perusideasta, joka toteutetaan ensin ja jota pystytään testaamaan käytännössä mahdollisimman aikaisessa vaiheessa. Kehittämisen aikana voi tulla mieleen monenlaisia toiveita, lisätoimintoja ja jatkokehittämisen kohteita, mutta kannattaa keskittyä olennaiseen ja saada päätoiminnallisuudet ensin kuntoon. Toiminnallisuuksia voi lisätä myöhemmin. Mitä yksinkertaisempi toiminta-ajatus ja mitä vähemmän siirtymiä ja valintoja, sitä parempi sovellus on käyttäjän kannalta. Parhaimmillaan sovelluksen käyttäminen on intuitiivista, jolloin se ei vaadi käyttöohjeiden tai opasteiden lukemista. Suunnittelussa pohditaan tällöin, miten ihminen käyttäytyy teknologiaa käyttäessään ja miten voimme vastata siihen (Garrett 2010, 81).

Nykyään puhutaan vuorovaikutussuunnittelusta (interaction design), jossa keskiössä ovat teknologiaa käyttävä ihminen, hänen tarpeensa, odotuksensa sekä tavat ja osaaminen. Opittavuus ja muistettavuus ovat keskeisiä vuorovaikutussuunnittelun kriteerejä. Tiedon esittäminen graafisesti ja tiedon hyvä organisointi ovat siinä keskeisiä (Saariluoma 2016; Nielsen 1993). Usein toteutuksessa käytetään niin sanottuja standardiratkaisuja. Tällä tarkoitetaan sitä, että tuotteen logiikka toimii samalla tavoin kuin muiden vastaavien palveluiden logiikka, eli siinä on käyttäjälle entuudestaan tuttuja elementtejä. Standardiratkaisuja voivat olla esimerkiksi graafiset käyttöliittymät, valikot, jotka on järjestetty ruudukkoon, ja erilaiset tutut kuvakkeet. Toisinaan kuitenkin yliarvioidaan sitä, mitkä kuvakkeet ovat kaikille selviä. Suurennuslasi-kuvake esimerkiksi voi edustaa tiedon etsimisen lisäksi myös kuvaketta, jolla voi suurentaa tekstiä tai kuvia. Mikäli päädytte tekemään poikkeavia suunnitteluratkaisuja, kannattaa varmistaa, että kohderyhmä ymmärtää uuden ratkaisun merkityksen. Visuaalisuudella on suuri merkitys. Visuaalisuuteen liittyy muun muassa kuvat, selkeys, kontrastit, värit, muodot mutta myös esteettisyys. Kauniiksi koetut tuotteet koetaan myös käytettäväksi. (Leikas & Laukka 2014.)

Mitä sitten on käytettävyyttä? ISO 9241-11 -standardi määrittelee käytettävyyden seuraavalla tavalla: ”Tarkkuus, tehokkuus ja tyytyväisyys, jolla määritellyt käyttäjät saavuttavat määritellyt tavoitteet tietystä ympäristössä.” Tarkkuudella tarkoitetaan, että järjestelmässä tai laitteessa on käyttäjän kannalta oikeat ja tarvittavat ominaisuudet. Tehokkuudella tarkoitetaan sitä, miten helppoa ja nopeaa järjestelmän tai laitteen käyttäminen käyttäjän näkökulmasta on. Tyytyväisyydellä tarkoitetaan käyttäjän tyytyväisyyttä järjestelmään tai laitteeseen eli sitä, onko järjestelmän tai laitteen käyttäminen miellyttävää ja pitääkö käyttäjä siitä. (Teknologian tutkimuskeskus 2017.) Käytettävyyttä voidaan mitata erilaisilla käyttäjätestauksilla (ks. luku 6).



### 3. Miten sovelluksen pitäisi toimia? Tekniset vaatimukset

Sovelluksen hintaan vaikuttavat tekniset vaatimukset, ja niitä on hyvä miettiä etukäteen, jotta asiasta voi keskustella kilpailutukseen osallistuvien toimijoiden kanssa. Tällä hetkellä Suomessa käytetään lähinnä kolmea käyttöjärjestelmää, joista hallitsevina ovat Android ja IOS sekä pikkuhiljaa poistuva Windows. Tätä opasta kirjoitettaessa (syksy 2017) Windows Phone -käyttöjärjestelmän osuus kaikista puhelimista oli Suomessa alle 5 % ja maailmanlaajuisesti alle 1 %. Sovellusta tehtäessä on päätettävä, millä käyttöjärjestelmillä sovellus toimii. Jokainen käyttöjärjestelmä lisää sovelluksen hintaa, koska sovelluksesta on tehtävä omat versionsa eri sovelluskauppoja varten.

Eri käyttöjärjestelmillä on myös omia teknisiä sovellusvaatimuksia. Esimerkiksi Apple vaatii, että sovelluksessa on oltava mekanismi, jolla on mahdollisuus ilmoittaa asiattomasta sisällöstä. Asiaton sisältö on poistettava vuorokauden kuluessa ilmoituksesta. Vastaavanlaisia uusia ehtoja saattaa tulla ajan kuluessa lisää, ja niihin pitää reagoida tai sovellus poistetaan kaupasta. Sovellus ei ole koskaan varsinaisesti valmis, vaan sitä voi joutua kehittämään lisää koska tahansa.

Kohderyhmä määrittää sovelluksen teknisiä vaatimuksia. Minkälaista teknologiaa kohderyhmä käyttää? Käyttöjärjestelmän lisäksi myös näytön koko luo rajoitteita sovelluksen kehittämiseen. Ikäihmisille tekstin näkyvyys voi esimerkiksi olla ongelma, jos laitteessa on liian pieni näyttö, mikä tulee ottaa huomioon jo teknisen suunnittelun alkuvaiheessa.

## 4. Mitä sovelluksen tekeminen maksaa? Budjetointi

*+60-sovellus on käyttäjälle ilmainen. Koska sen käyttö perustuu tunniste-tietoihin, sovellus vaatii lataamisen jälkeen tunnistautumista, jota varten lataaja saa koodin tekstiviestillä. Jokainen tekstiviesti on ylläpitäjälle maksullinen. Käytännössä tästä voisi muodostua suurikin lasku, mikäli sovellus olisi yhtäkkiä alkanut kiinnostaa suurta yleisöä.*

Sovellusten hintahaarukka on luonnollisesti todella suuri, ja hintaan vaikuttavat sovelluksen sisältö ja se, kuinka monimutkainen sovellus on, sisältääkö se paljon grafiikkaa, toimiiko se useilla kielillä ja useissa käyttöjärjestelmissä, kuinka nopea se on ja kuka sen tekee. Esimerkiksi LähiVerkon +60-sovellus maksoi hieman alle 30 000 euroa ja EHYT ry:n OttoMitta noin 25 000 euroa.

Jos järjestön toimintaa rahoitetaan Veikkauksen tuotoilla, on syytä tarkastaa rahoittajan suhtautuminen sovelluksen kehittämiseen. STEAn avustusvalmistelija ja ICT-hankkeiden yhteyshenkilö Vesa Kurikka toteaa:

*Osana kehityshankkeita STEAn rahoitusta voi käyttää sovelluskehitykseen, jos se nähdään tarpeelliseksi hankkeen tavoitteiden saavuttamiseksi. Ensisijaisesti tulee kuitenkin selvittää, miten hyvin jo olemassa olevat sovellukset soveltuvat joko sellaisinaan tai mukautettuina suunniteltuun käyttöön. Kehitettäessä uutta sovellusta tulee hankinta kilpailuttaa, jos kehitystyö tilataan ulkopuoliselta toimittajalta. Toteutuksen on syytä pohjautua avoimen lähdekoodin ratkaisuihin, jotta sovelluksen ylläpito ja jatkokehitysmahdollisuudet turvataan. Sovelluksen tekijänoikeudet tulee määrittää niin, että sovelluksen käyttö on mahdollista myös muille sosiaali- ja terveysalan järjestöille. Jos kyseessä on pelkkä sovelluskehitys ilman kehitysprojehtia, voidaan kehitystyöhön hakea STEAn investointiavustusta. Tällöin avustusosuus on 70 % kokonaiskustannuksista. STEAn rahoitus voi tulla kyseeseen lähinnä suurissa useampia toimijoita palvelevien ratkaisujen toteutuksissa. Avustuskriteerit ovat samat kuin STEAn muissa avustuskohteissa, eli tarve ja tavoitteet tulee perustella yksityiskohtaisesti. (Tiedoksianto Vesa Kurikka, STEA 29.9.2017.)*

Rahoitusta sovelluksen tekoon voi hakea myös säätiöiltä ja opetus- ja kulttuuriministeriöstä tai yritysten kanssa voi tehdä yhteistyötä. Sovelluksen toteutuksessa kannattaa pohtia myös yhteistyötä muiden järjestöjen kanssa, joilla on samanlaisia tarpeita. Järjestöjen yhteisissä sovelluksissa resursseja saadaan jaettua, mutta toisaalta on sovittava etukäteen monia käytännön asioita, kuten omistajuus, käyttö- ja tekijänoikeudet, päivityksistä huolehtiminen, työnjako ja organisointi ja kustannusten jakautuminen, myös sovelluksen valmistumisen jälkeen.

Usein budjetin ajatellaan koostuvan pääosin sovelluksen teknisestä kehittämisestä, mutta sen lisäksi kannattaa budjetoida rahaa sovelluksen elinkaarta ajatellen. Budjetoinnissa on hyvä huomioida ainakin seuraavia kustannuksia:

- Sovelluksen ylläpidosta syntyy kuluja, joihin ei aina osata varautua. Näistä lisää ylläpito-osiossa luvussa kahdeksan. Alkuvaiheessa kannattaa jo päättää, kuka tekee sovelluksen päivitykset ja miten ne kustannetaan?
- Sovellukset ovat yleensä käyttäjille ilmaisia, mutta sovelluksen käytöstä voi aiheutua kuluja järjestölle. Sovelluksen toimiminen saattaa vaatia esimerkiksi tekstiviestien käyttöä, jotka ovat maksullisia sovelluksen omistavalle organisaatiolle.
- Markkinointiin on varattava rahaa ja mietittävä, käytetäänkö esimerkiksi mainostoimistoa ja tarvitaanko mainosmateriaaleja. Tarvitaanko sovellukseen grafiikkaa tai visuaalisia elementtejä? Esimerkiksi Apple käyttää vain pienen osan liikevaihdostaan tuotekehitykseen, mutta markkinointi sitä vastoin lohkaisee arvattavasti melkoisen siivun.

## 5. Ketkä kehittävät sovelluksen? Kilpailutus ja sopimus

*Pyysimme tarjouksia neljältä eri sovellusyrytykseltä. Tapasimme tarjouksen tehneiden yritysten edustajia erikseen noin tunnin kestävässä palaverissa, joissa kerrottiin projektistamme ja sen tavoitteista sekä ideoitii sovelluksen toimintaa. Saimme neljä tarjousta, joissa sovelluksen toimintaan ehdotettiin hyvin erilaisia ratkaisuja. Ainoastaan hinta oli kaikissa lähes sama. Olimme painottaneet tarjouspyynnössä luovuutta ja tämän painotuksen perusteella valitsimme palveluntarjoajan.*

Sovelluksen perusidean suunnittelu ja toiminnallisuuksien miettiminen etukäteen auttavat kilpailutuksessa. Aina ei kuitenkaan tarkalleen tiedetä, mitä ollaan tilaamassa ja millaisia toteutusmahdollisuuksia on käytettävissä. Sovellusten maailma on lähes rajaton, eikä tilaajan luovuus tai osaaminen välttämättä riitä siihen, että hän osaisi määritellä tai arvioida sovelluksen toimintaa kovin tarkkaan. Seuraava kysymyslista voi auttaa tarjousten vertailussa:

- Mitä referenssejä yrityksellä on? Millaisia toteutuksia he ovat tehneet aikaisemmin?
- Onko kohderyhmä yritykselle tuttu?
- Onko yritys tehnyt aiemmin järjestöille sovelluksia?
- Mille käyttäjärjestelmille yritys on tehnyt sovelluksia?
- Onko yrityksen talous vakaa?
- Kuka omistaa grafiikat?
- Mitä sitten, jos firma lopettaa toimintansa? Mistä löytyvät kuvat, koodit, grafiikat, fontit jne.?

Sopimusta ja työtilausta tehtäessä kannattaa miettiä ja keskustella palveluntuottajan kanssa mm. seuraavista asioista:

- Miten aikataulu etenee, millaisia etappeja siihen sisältyy? Mitä aikataulu tarkoittaa tilaavan järjestön kannalta, milloin pitää varata aikaa esimerkiksi ideoiden testaukseen ja kommentointiin? Aikatauluttakaa kehittämisen etukäteen, mutta muistakaa myös jättää siihen väljyyttä ja varautukaa viivästyksiin.
- Miten hoidetaan tiedonkulkua? Millä kanavalla sovitaan asioista? Keiden kesken? Kuka on yhteydessä ja kenen kanssa?
- Keskustelkaa siitä, miten otetaan huomioon palautteet ja muokkaukset, joita tulee kehittämisen aikana esimerkiksi kohderyhmätestausten kautta?
- Miten käyttäjät otetaan mukaan ja miten käyttäjättestaukset suoritetaan? Kuka hankkii testaajat? Ketkä testaavat ja miten?
- Missä ja miten säilytetään mahdolliset henkilötiedot, joita sovelluksen käytön ja rekisteröitymisen aikana kerätään?
- Muistakaa pyytää sopimukseen ehto, että omistatte itse koodin ja voitte jatkaa sen kehitystä myös kolmannen osapuolen kanssa. Näin ette ole sidottuja alkuperäiseen yritykseen, kun sovellusta jatkokehitetään, tai joudu tilanteeseen, jossa sovellus on kesken mutta palveluntuottaja ei pysty sitä viemään eteenpäin.
- Sovelluskehityksen myöhästymisistä voidaan sopia sakkomaksuja. Usein sopimukseen sisältyy myös maksuun liittyviä ehtoja. Sovelluksen kokonaislasku voidaan maksaa esimerkiksi kolmessa osassa: alussa, tietyn kehitysvaiheen jälkeen ja sovelluksen luovutuksen yhteydessä. Sopimukseen kannattaa kirjata myös takuu-aika ja sen aikana löytyvien virheiden korjaus.

## 6. Miten kohderyhmä otetaan mukaan? Testaus

*+60-sovelluksen kehittäminen alkoi joulukuussa 2014. Tammikuussa 2015 järjestimme ensimmäisen käyttäjätestaustilaisuuden vapaaehtoisille testaajille. Saimme toimistolle kuusi vapaaehtoista, jotka olivat iältään 40–70 vuotta. Testauksen suoritti sovelluksen tehnyt taho. Testaus sisälsi lähinnä sovelluksen käytettävyyden testausta. Järjestön työntekijät taas kommentoivat sovelluksen ulkoasuun liittyviä kysymyksiä. Myöhemmin sovellusta testasi ryhmä vapaaehtoisia palvelutalon asukkaita, jotka kokeilivat sovelluksen käyttöä viikon ajan erilaisten tehtävien avulla ja antoivat suullista ja kirjallista palautetta kokemuksistaan.*

Järjestöillä on usein se etu, että oma jäsenistö tai kohderyhmä on tuttu ja aktiivisesti mukana järjestön toiminnassa. Sovelluksen suunnitteluun ja kehittämiseen on tärkeää ottaa mukaan kohderyhmän edustajia jo heti alusta alkaen ja osallistaa heitä koko sovelluksen kehittämisen ajan suunnitteluun, kommentointiin ja testaukseen.

Käyttäjäkokeemusta kannattaa testata kohderyhmän kanssa kehittämisen eri vaiheissa. Ennen käyttäjätestausta on hyvä pohtia, mitä asioita testataan: sovelluksen toimivuutta, ulkoasua, selkeyttä, helppokäyttöisyyttä, miellyttävyyttä jne. Eri tarpeisiin on olemassa erilaisia testipatteristoja, joita löytyy esimerkiksi sivustolta <http://www.allaboutux.org/>, ja niitä voi yhdistellä tarpeen mukaan. Muistakaa kysyä sovellusta tekevältä taholta, miten käyttäjätetaus on huomioitu. Käyttäjäkokeemuksen testaamiseen on monta tapaa. SUS (System Usability Scale) -kysely on yksi eniten käytetyistä käyttäjätetaamisen mittaristoista. Sillä voidaan mitata tuotteiden, yleensä verkkosivujen, käytettävyyttä. (Tullis & Albert 2013, 148.) Testikysymykset löytyvät esimerkiksi teoksesta *Measuring the User Experience*.

Käyttäjätetien tekeminen ei ole välttämättä vaikeaa ja/tai kallista. Helppoimmillaan se voidaan toteuttaa niin, että kutsutaan kohderyhmään kuuluvia henkilöitä sopivaan tilaan kokeilemaan sovelluksen prototyyppiä. Samalla havainnoidaan, miten ihmiset sovellusta käyttävät, ja keskustellaan siitä, mikä sovelluksessa toimii ja mikä ei. Nielsen Norman Groupin mukaan viisi testaajaa riittää paljastamaan 85 % käytettävyysongelmista. Testaajia



valittaessa kannattaa miettiä kohderyhmää ja valita testaajiksi mahdollisimman lähellä kohderyhmää olevia henkilöitä. Testaajien tulee olla ulkopuolisia, jotta heillä ei ole sidosta itse tuotteeseen. Näin saadaan mahdollisimman todenmukainen palaute sovelluksesta. Koska testaajia voi olla hankala saada osallistumaan ilman palkintoa, kannustimena voi toimia esimerkiksi lahjakortti, lounas tai pullakahvit.

On suositeltavaa tehdä pieniä testejä useasti sen sijaan, että tekisi vain yhden ison testauksen. Hyvän sovelluskehittäjän tunnistaa siitä, että hän sisällyttää käyttäjätestauksen iteratiivisesti osaksi kehittämistä. Iteratiivinen kehittäminen tarkoittaa sitä, että kun käyttäjätesteissä havaitaan parannettavaa, parannus tehdään seuraavaan versioon ja sitä testataan uudelleen. Jos käyttäjätestausta ei esimerkiksi budjetti- tai aikataulusyistä voida toteuttaa, kannattaa testausta tehdä itse edes jollakin tasolla. Jo yhdellä käyttäjätestauksella saat enemmän ymmärrystä kuin tekemättä yhtään testausta.

## 7. Miten sovellukselle saadaan käyttäjiä? Tiedottaminen ja markkinointi

*Palkkasimme mainostoimiston toteuttamaan sovelluksen markkinointia kanssamme. Heidän kanssaan keksimme sovelluksen nimen ja saimme sovellukselle logon ja graafisen ohjeistuksen sekä mainosjulisteen ja -esitteen. Pohdimme suuren lanseeraustilaisuuden järjestämistä, mutta se jätettiin pitämättä, koska sovelluksen aikataulu viivästyi. Sovellus ei ollut siinä muodossa, että sitä olisi voinut markkinoida suurelle yleisölle.*

Markkinointi on asia, joka helposti unohtuu tai jää suunnittelematta. On hyvä varautua ajallisesti ja resurssien puolesta siihen, että markkinointia täytyy suunnitella ja käynnistää samaan aikaan, kun sovellusta vielä suunnitellaan ja kehitetään. Tehkää suunnitelma sovelluksen markkinoinnista. Vaikka idea ja itse sovellus olisivat kuinka hyviä tahansa, tieto ei leviä puskaradion kautta kaikille. Usein markkinointiin kannattaa panostaa. Miksi tehdä kallis sovellus, jos siitä ei ole varaa kertoa kenellekään? On hyvä varautua myös siihen, että markkinointi onnistuu ja sovelluksesta tuleekin menestys. Mitä tehdään sitten, jos esimerkiksi nuorten mobiilichattiin ilmestyy 500 nuorta markkinointitempauksen jälkeen?

Sovellukselle tarvitaan jo aikaisessa vaiheessa tiettyjä elementtejä. Näitä ovat sovelluksen nimi ja logo/tunnus sekä visuaalinen ilme ja värimaailma, joita käytetään sovelluksen toteutuksessa.

- Nimen olisi hyvä olla mieleen jäävä. Pohtikaa, riittääkö suomenkielinen nimi vai onko kohderyhmä monikielinen? Kertooko nimi sisällöstä jotakin? Onko se lyhyt tai muutoin helposti muistettava? Onko se hauska? Tarkastakaa sovelluskaupasta, onko nimi ainutlaatuinen eikä se ole toisen sovelluksen käytössä.
- Sovelluksen logo/tunnus näkyy puhelimen näytöllä, kun sovellus on ladattu. Joskus sovelluksen tekevä yritys voi suunnitella myös logon, etenkin jos sovellus sisältää muutakin grafiikkaa.



Eri sovelluskaupoissa on erilaisia vaatimuksia sovelluksille. Sovelluksen lanseeraustilaisuus kannattaakin järjestää vasta, kun sovellus on varmasti läpäissyt kaikki tekniset vaatimukset ja on valmis käyttöön otettavaksi. Yllättäviä viivästyksiä voi tulla aivan viime metreillä. Sovelluskehittäjät kertovat, että ensimmäinen versio sovelluksesta menee vain todella harvoin sovelluskauppojen seulan läpi ja muutoksia tai lisäyksiä joudutaan tekemään.

## 8. Millainen on sovelluksen elinkaari? Kehittämisen jatkuva prosessi

Juuri kun luulimme, että nyt kaikki on valmista, työ vasta alkoikin. Tuntui, että sovellusta piti päivittää jatkuvasti. Kuulimme lataajilta ongelmatilanteista sovelluksen toiminnassa ja havaitsimme itsekin, että joitakin ratkaisuja olisi hyvä muuttaa, mutta rahoitusta isompien päivitysten tekemiseen ei enää loppuvaiheessa ollut budjetoitu. Lisäksi pohdimme, mitä sovellukselle tapahtuu, kun projektimme päättyy. Kuka vastaa päivityksistä ja niiden rahoittamisesta? Arviointia emme olleet miettineet ollenkaan muuta kuin omien projektitavoitteidemme kannalta. Myös viestinnän vaatavuus yllätti: sovellus oli toiminnallisuudeltaan sellainen, että sen käyttö olisi vaatinut varsin suuren käyttäjäjoukon pienen alueen sisälle.

Sovelluksella on elinkaari, joka alkaa ideasta ja päättyy johonkin. On houkuttelevaa ajatella, että sovelluksen elinkaaren muodostavat ideointi, suunnittelu, toteutus ja lopussa valmiin tuotteen käyttöönotto. Mutta todellisuudessa sovelluksen valmistumisen jälkeen sovelluksen elinkaari on vasta alussa. Ensimmäistä versiota sovelluksesta kutsutaan beta-versioksi. On hyvin todennäköistä, että tässä vaiheessa löytyy teknisiä vikoja, joita täytyy korjata. Käyttäjiltä voi tulla palautetta, jonka perusteella sovellusta on korjattava. Käyttäjärjestelmiin tulee päivityksiä, joiden takia sovellustakin on päivitettävä. Tähän on hyvä varautua ja valmistautua siihen, että kehittäminen prosessi jatkuu ensimmäisen version julkaisun jälkeenkin.

Kun sovellus on hyväksytty sovelluskauppoihin ja julkaistu, on markkinointi täytynyt viimeistään käynnistää. Toteuttajia tietenkin kiinnostaa, kuinka paljon sovellusta ladataan. Tässä kannattaa hyödyttää App Anniea. App Annie on sovellusmarkkinatietoyhtiö, jonka sivulta ([www.appannie.com](http://www.appannie.com)) voit seurata sovelluksen latausmääriä. Kun rekisteröidyt App Annieen ja syötät omalle sivullesi sovelluskauppojen tunnuksesi, näet kätevästi samasta taulukosta kaikkien kauppojen latausten lukumäärän. Voit halutessasi seurata latauksia tietyltä aikaväliltä tai tietyn sovelluskaupan osalta. Näet myös, missä maissa sovellusta on ladattu. Mikäli sovellus on kokonaan tai joiltain osin maksullinen, myös tuottojen muodostumisen näkee kätevästi samalta sivulta.

Jos ajatellaan sovelluksen käyttöä, kiinnostavampaa olisi tietää, kuinka kauan tai kuinka useasti henkilöt käyttävät sovellusta. Lukumäärien lisäksi kiinnostavaa on tietää, keitä käyttäjät ovat, miksi he käyttävät sovellusta ja mitä lisäarvoa se tuo heidän elämäänsä. Yksi mahdollisuus tarkastella kokemusta on käyttäjien antama palaute. Jos sovellukseen on mahdollista lisätä esimerkiksi linkki palautekyselyyn, se kannattaa tehdä. Vastausprosenttia lisäävät vastaajien kesken tietyin väliajoin suoritettavat arvonnat. Suppeampana vaihtoehtona on sovelluksen luokituksen seuraaminen. Sekä Google Playssa että Applen sovelluskauppa Appstoressa käyttäjät voivat arvioida sovelluksia antamalla niille tähtiluokituksia yhdestä viiteen ja lisäksi sanallisia arviointia. Näitä kannattaa seurata, palaute on arvokasta.

Tuotteen kehittämisestä voi tehdä jatkuvan sopimuksen palveluntuottajien kanssa. Jotkut yritykset tarjoavat esimerkiksi kuukausimaksullista sopimusta, jossa he toteuttavat kaikki sovelluskaupoista tai käyttöjärjestelmien muutoksista johtuvat muutokset ja korjaavat käyttäjien löytämät bugit.

## 9. Mitä vielä pitäisi ottaa huomioon? Sovelluksen ylläpito

EU:n tietosuojasetus (General Data Protection Regulation, GDPR) astuu voimaan koko EU:n alueella keväällä 2018. Se koskee myös sovelluksia. Asetus muuttaa yritysten ilmoitusvelvollisuutta mahdollisten tietosuojamurtojen yhteydessä sekä vähentää yritysten oikeuksia kerätä henkilötietoja. Asetuksesta kannattaa itsekin olla tietoinen, koska sovelluksen ylläpito usein siirtyy jossain kohtaa kokonaan sen tilanneelle taholle. Miten sitten tietosuojasetukseen tulee varautua? Ensimmäisenä kannattaa kysyä sovellusta tekevältä yritykseltä, miten he ovat asian huomioineet. Toiseksi kannattaa miettiä, mitä henkilötietoja sovellus kerää. Usein esimerkiksi puhelinnumero on sellainen tieto, joka tallentuu rekisterin pitäjälle. Kerääkö sovellus jotain muuta tietoa? Onko ikä esimerkiksi olennainen tieto (se voi olla, mikäli sovelluksessa on ikäraja).

Sovelluksista löytyy yleensä tietosuojaseloste (tai rekisteriseloste) ja käyttöehdot. Ensimmäisessä kerrotaan, mitä tietoja lataajista kerätään, mikä taho niitä kerää ja mikä rekisterin käyttötarkoitus on. Asianmukaisessa rekisteriselosteessa kerrotaan myös, mihin tietoja luovutetaan ja kuka niitä pääsee tarkastelemaan. Käyttöehdoissa määritetään vastuut ja oikeudet, esimerkiksi kenelle kuuluu vastuu, jos sovellus ei toimi odotetunlaisesti ja mitä vastuita palvelun käyttäjällä on. Käyttöehtoja laadittaessa kannattaa tutustua muiden sovellusten käyttöehtoihin.

# Lopuksi

Projektimme on päättymässä, ja katselemme tekemiämme valintoja nyt jo matkan päästä. Omassa kehittämissä prosessissamme päädyimme lopulta poistamaan sovelluksen sovelluskaupoista. Sovelluksessa ilmeni liikaa teknisiä ongelmia ollakseen helppokäyttöinen, ja ongelmien korjaaminen olisi vaatinut paljon uutta testaamista, teknistä kehittämistä ja muokkaamista. Huomasimme myös, että sovelluksemme kilpaili liian vaativilla markkinoilla. Se oli ikääntyville suunnattu sosiaalisen median sovellus, jonka toiminta olisi vaatinut varsin suuria alueellisia käyttäjämääriä.

Sovelluksen teko oli kuitenkin erittäin opettavainen kokemus, joka sai aikaan paljon hyvää. Teimme matkan varrella sovelluksia ja mobiiliteknologiaa tutuiksi sadoille ikäihmisille ja välitimme viestiä siitä, että nykyaikainen mobiiliteknologia kuuluu kaikille, myös ikäihmisille. Teknologian kehittämissä tulee huomioida ikäihmisten näkökulma, tarpeet, toiveet ja elämäntapa.

Olemme huomanneet, että moni järjestö pohtii oman sovelluksen kehittämistä ja kaipaa siihen tietoa ja tukea. Yksi tuotos omasta sovelluskehittämisen prosessistamme on tämä opas, josta toivomme olevan hyötyä niille järjestöille, jotka miettivät sovelluksen kehittämistä. Uskomme edelleen, että mobiilisovelluksissa on merkittävä potentiaali hyödyttää järjestöjen työtä, kun sovelluksen tekemisen järjestyminen ja hyödyllisyys on tarkkaan punnittu. Seuraavaan sovellukseen olemme jo paremmin valmistautuneita, toivottavasti myös te!

## Lähteet:

- Leikas, Jaana (2010). Life-Based Design. A holistic approach...  
Saatavilla: [http://www.vtt.fi/Documents/2010\\_P726.pdf](http://www.vtt.fi/Documents/2010_P726.pdf)
- Leikas, Jaana; Laukka, Maria (2014). Esteettisyys ikäteknologiassa, teoksessa *Ikäteknologia* (toim. Jaana Leikas). Raisio: Newprint Oy.
- Nielsen, J. (1993). *Usability Engineering*. New York: Academic Press.
- Nielsen Norman Group. Verkkosivu: <https://www.nngroup.com/articles/why-you-only-need-to-test-with-5-users/>, viitattu 26.9.2017
- Saariluoma, Pertti; Cañas, Jose, J., Leikas, Jaana (2016). *Designing for life. A human perspective on Technology Development*. Lontoo: Macmillan Publishers Ltd.
- Saariluoma, Pertti; Kujala, Tuomo; Kuuva, Sari; Kymäläinen, Tiina; Leikas, Jaana; Liikkainen, Lassi A.; Oulasvirta, Antti (2010). *Ihminen ja teknologia. Hyvän vuorovaikutuksen suunnittelu*. Tampere: Tammerprint Oy.
- Teknologian tutkimuskeskus (2015). *Mitä käytettävyyys tarkoittaa?*
- Toikkanen (2016). <http://vapaaehtoisestiparas.fi/jarjestojen-mobiilisovellukset-osa-1-lista-olemassa-olevista-appeista/>
- Tullis, Tom; Albert, Bill (2013). *Measuring the user experience: collecting, analyzing, and presenting usability metrics* (toinen painos). Elsevier.

Moni järjestö, joka haluaa kokea kuuluvansa edelläkävijöihin ja pysyä mukana digitalisaatiossa, pohtii mobiilisovelluksen tekemistä. LähiVerkko-projektissa käynnistettiin sovelluksen kehittäminen vuoden 2014 lopulla. Matka ideasta valmiiseen sovellukseen oli haastava, mutta samalla täynnä oppimisen kokemuksia ja oivalluksia. Tässä oppaassa jaamme näitä kokemuksiamme. Toivomme, että tämä opas auttaa muita sovelluksen tekemistä pohtivia ja suunnittelevia järjestöjä omalla matkallaan kohti mobiiliteknologiaa.

LähiVerkko-projekti on Eläkeliiiton ja EHYT ry:n yhteisprojekti, joka toteutettiin 2013–2017. Projektissa on tuotettu runsaasti materiaalia ja tietoa ikäihmisten tietotekniikan käytön tueksi. Tutustu: [www.lahiverkko.fi](http://www.lahiverkko.fi)

