



Teemu Olkkonen

# Ihmislähtöinen ohjelmistokehitys: käyttäjäkeskeisen suunnittelun vaikutus ja sovellukset

Metropolia Ammattikorkeakoulu

Insinööri (AMK)

Tieto- ja viestintätekniikka

Insinöörityö

18.12.2024

# Tiivistelmä

Tekijä:	Teemu Olkkonen
Otsikko:	Ihmislähtöinen ohjelmistokehitys: käyttäjäkeskeisen suunnittelun vaikutus ja sovellukset
Sivumäärä:	47 sivua
Aika:	18.12.2024
Tutkinto:	Insinööri (AMK)
Tutkinto-ohjelma:	Tieto- ja viestintätekniikka
Ammatillinen pääaine:	Ohjelmistotuotanto
Ohjaajat:	Lehtori Jorma Rätty

---

Tämän opinnäytetyön tarkoituksena on tutkia käyttäjäkeskeisen suunnittelun periaatteita ja niiden soveltamista ohjelmistokehitykseen. Työ tarkastelee, kuinka käyttäjäkeskeinen suunnittelu voi parantaa ohjelmistotuotteiden käytettävyyttä ja laatua. Erityisesti tutkimus käsittelee iteratiivisen suunnitteluprosessin ja käyttäjäpalautteen hyödyntämisen merkitystä käytännön esimerkkien kautta, kuten Applen iPhone:n kehityksessä. Lisäksi työssä käsitellään käyttäjäkokemuksen roolia nykypäivän teknologiaratkaisuissa ja tuodaan esiin opit, joita voidaan hyödyntää tulevaisuuden ohjelmistoprojekteissa.

Työn menetelminä käytettiin kirjallisuuskatsausta, jossa tarkasteltiin käyttäjäkeskeisen suunnittelun ja käytettävyyden periaatteita ohjelmistokehityksessä sekä omia kokemuksiani käyttäjärajapinnasta, erityisesti palvelupistekokemuksen näkökulmasta. Keskeisinä lähteinä toimivat Donald A. Normanin teos *The Design of Everyday Things* sekä Nielsen Norman Groupin julkaisut, jotka tarjoavat teoreettista pohjaa käyttäjäkeskeisen suunnittelun ja käytettävyyden merkityksen ymmärtämiselle.

Työn tuloksena havaittiin, että käyttäjäkeskeinen suunnittelu parantaa merkittävästi ohjelmistojen käytettävyyttä ja käyttäjätyytyväisyyttä. Käyttäjien osallistuminen suunnitteluprosessin eri vaiheissa vähentää käyttöön liittyviä virheitä ja lisää käyttäjien sitoutumista tuotteeseen. Esimerkkinä toimineen iPhone:n kehityksessä käyttäjäpalautteen jatkuva hyödyntäminen johti innovatiivisiin ja intuitiivisiin käyttöliittymäratkaisuihin, jotka asettivat uusia standardeja alalle. Tulokset vahvistavat, että käyttäjäkeskeisen suunnittelun periaatteet, kuten iteratiivinen kehitys ja käyttäjätestaus, ovat olennaisia ohjelmistokehityksen onnistumiselle ja käyttäjäystävällisten tuotteiden luomiselle.

Avainsanat: Käyttäjäkeskeinen suunnittelu, käytettävyys, iteratiivinen suunnittelu, käyttäjäkokemus, ohjelmistokehitys, design thinking, käytettävyydestaus, affordanssit, ekosysteemit ja tekoäly.

---

Tämän opinnäytetyön alkuperä on tarkastettu Turnitin Originality Check -ohjelmalla.

## Abstract

Author: Teemu Olkkonen  
Title: Human-Centered Software Development: Impact and Applications of User-Centered Design  
Number of Pages: 47 pages  
Date: 18 December 2024

Degree: Bachelor of Engineering  
Degree Programme: Information and Communication Technology  
Professional Major: Software Development  
Supervisors: Jorma Rätty, Senior Lecturer

---

The purpose of this thesis is to study the principles of user-centered design and their application to software development. The paper examines how user-centered design can improve the usability and quality of software products. In particular, the research deals with the importance of the iterative design process and the utilization of user feedback through practical examples, such as in the development of Apple's iPhone. In addition, the thesis discusses the role of user experience in today's technology solutions and brings out lessons that can be used in future software projects.

The research methods include a literature review focused on the principles of user-centered design and usability in software development, supplemented by personal experience from the user interface perspective in a Service Desk context. Key sources include Donald A. Norman's *The Design of Everyday Things* and publications from the Nielsen Norman Group, providing a theoretical foundation for understanding the importance of user-centered design and usability.

The findings show that user-centered design significantly enhances software usability and user satisfaction. Involving users at various stages of the design process reduces usage errors and strengthens user engagement with the product. In the case of iPhone development, the continuous integration of user feedback led to innovative and intuitive interface solutions, setting new industry standards. These results confirm that principles of user-centered design, such as iterative development and user testing, are essential to successful software development and the creation of user-friendly products.

Keywords: User-centered design, usability, iterative design, user experience, software development, design thinking, usability testing, affordances, ecosystems and artificial intelligence

# Sisällys

## Lyhenteet

1	Johdanto	1
2	Käyttäjäkeskeisen suunnittelun periaatteet	2
2.1	Käyttäjäkeskeisen suunnittelun periaatteet	2
2.2	Käytettävyyden merkitys ohjelmistokehityksessä	4
2.3	Design Thinking -lähestymistapa	6
3	Teoreettinen viitekehys	9
3.1	Affordanssit ja signaalit suunnittelussa	9
3.2	Palaute ja rajoitteet käytettävyydessä	11
3.3	Virheiden hallinta ja ehkäisy	13
4	Case tutkimus: Applen iPhone	15
4.1	Käyttäjätutkimuksen rooli iPhonen kehityksessä	15
4.2	Iteratiivinen suunnittelu ja prototypointi	17
4.3	Käytettävyydestä iPhonen kehitysprosessissa	20
4.4	Uusien standardien luominen ja innovatiiviset suunnittelupäätökset	22
5	Tulokset ja analyysi	25
5.1	Käyttäjäkeskeisen suunnittelun vaikutukset iPhonen menestykseen	25
5.2	Haasteiden voittaminen ja oppimiskokemukset	28
5.3	Vaikutukset laajempaan ohjelmistoteollisuuteen	30
6	Pohdintoja tulevaisuuteen	33
6.1	Opit ja parhaat käytännöt	33
6.2	Käyttäjäkeskeisen suunnittelun tulevaisuuden näkymät	35
6.3	Suosituksia suunnittelijoille ja ohjelmistokehittäjille	37
7	Yhteenveto	40
7.1	Tutkimuksen keskeiset johtopäätökset	40
7.2	Tutkimuksen rajoitukset ja jatkotutkimusaiheet	41
7.3	Loppupäätelmät	43



## Lyhenteet

- UCD: User-Centered Design, käyttäjäkeskeinen suunnittelu. Suunnittelufilosofia, joka asettaa käyttäjät keskiöön kehitysprosessin kaikissa vaiheissa parantaakseen tuotteen käytettävyyttä ja käyttäjäkokemusta.
- AR: Augmented Reality, lisätty todellisuus. Teknologia, joka yhdistää reaali maailman ja virtuaalisen sisällön käyttäjän näkymään reaaliaikaisesti.
- VR: Virtual Reality, virtuaalitodellisuus. Simuloitu ympäristö, joka tarjoaa käyttäjälle immerstiivisen kokemuksen hyödyntämällä tietokonetta ja erityisiä laitteita kuten VR-laseja.
- UX: User Experience, käyttäjäkokemus. Käyttäjän kokemus ja tunteet, jotka syntyvät tuotteen tai palvelun käytön yhteydessä.
- UI: User Interface, käyttöliittymä. Tuotteen tai järjestelmän visuaaliset elementit ja komponentit, joiden avulla käyttäjä on vuorovaikutuksessa sen toiminnallisuuden kanssa.

# 1 Johdanto

Käyttäjäkeskeinen suunnittelu (User-Centered Design, UCD) on suunnittelufilosofia, joka keskittyy käyttäjien tarpeiden, odotusten ja rajoitusten ymmärtämiseen sekä niiden huomioimiseen koko suunnittelu- ja kehitysprosessin ajan. Sen tavoitteena on varmistaa, että tuotetut ohjelmistot ja palvelut eivät pelkästään täytä teknisiä vaatimuksia, vaan myös tarjoavat käyttäjäystävällisen ja tehokkaan käyttökokemuksen. Käyttäjäkeskeinen suunnittelu eroaa perinteisistä suunnittelumenetelmistä painottamalla käyttäjän aktiivista osallistumista ja iteratiivista prosessia, jossa suunnitelmia testataan ja parannetaan jatkuvasti käyttäjäpalautteen perusteella. (Norman.)

Nykyisessä digitaalisessa ympäristössä käyttäjäkokemus on noussut keskeiseksi kilpailutekijäksi, joka voi vaikuttaa merkittävästi ohjelmistotuotteiden menestykseen tai epäonnistumiseen. Teknologian kehittyessä ja digitaalisten palveluiden monipuolistuessa käyttäjät odottavat yhä enemmän intuitiivisuutta, saumattomuutta ja miellyttävyyttä tuotteilta, joita he käyttävät. Esimerkiksi Applen iPhone on erinomainen esimerkki siitä, miten käyttäjäkeskeinen suunnittelu voi johtaa innovatiivisiin ratkaisuihin ja merkittävään kaupalliseen menestykseen. Toisaalta epäonnistumiset, kuten Microsoftin Windows 8, korostavat, miten tärkeää on ymmärtää ja vastata käyttäjien odotuksiin.

Työssäni Technical Support Specialistina olen huomannut, kuinka tärkeä rooli käyttäjäkeskeisellä ajattelulla on päivittäisessä työssäni. IT-palvelumme (Service Desk) on usein ensimmäinen yhteyspiste käyttäjien ja teknologian välillä, ja olemme jatkuvasti vuorovaikutuksessa käyttäjien kanssa, mikä tarjoaa meille ainutlaatuisen näkökulman heidän tarpeisiinsa ja haasteisiinsa. Tämän vuorovaikutuksen kautta voimme kerätä arvokasta palautetta, joka auttaa meitä parantamaan ohjeistuksia, tukiprosesseja ja jopa vaikuttamaan tulevaan ohjelmistokehitykseen.

Työn merkitys korostuu erityisesti nykyisessä teknologiaympäristössä, jossa käyttäjien odotukset ja vaatimukset kasvavat jatkuvasti. On tärkeää, että

suunnittelijat ja kehittäjät omaksuvat käyttäjäkeskeisen lähestymistavan, jotta he voivat luoda tuotteita, jotka eivät ainoastaan vastaa käyttäjien tarpeisiin, vaan myös ylittävät heidän odotuksensa, luoden siten vahvemman asiakasuskollisuuden ja kilpailuedun.

## **2 Käyttäjäkeskeisen suunnittelun periaatteet**

### **2.1 Käyttäjäkeskeisen suunnittelun periaatteet**

Käyttäjäkeskeinen suunnittelu on suunnittelufilosofia, joka keskittyy loppukäyttäjän tarpeiden, odotusten ja rajoitusten ymmärtämiseen ja niiden asettamiseen suunnitteluprosessin keskiöön. Käyttäjäkeskeisen suunnittelun peruseriaatteet ovat jatkuva käyttäjien osallistaminen, iteratiivinen suunnittelu, käyttäjäpalautteen kerääminen ja analysointi sekä suunnittelun optimointi käyttäjien palautteen pohjalta.

Yksi käyttäjäkeskeisen suunnittelun keskeisimmistä periaatteista on käyttäjän aktiivinen osallistaminen kehitysprosessin kaikissa vaiheissa. Tämä voidaan toteuttaa erilaisten käyttäjä tutkimusten, kuten haastattelujen, kyselyiden ja käytettävyydestien avulla. Käyttäjän osallistuminen auttaa varmistamaan, että suunnittelu vastaa käyttäjien todellisia tarpeita eikä vain kehitystiimin oletuksia. Käyttäjien osallistaminen on erityisen tärkeää ohjelmistokehityksessä, sillä käyttäjien kokemukset ja palaute ovat arvokasta dataa, jonka avulla voidaan tehdä parannuksia ja vältetään virheitä varhaisessa vaiheessa. (Norman; Gibbons.)

Omassa työssäni Service Deskissä käyttäjäkokemus on tärkeä osa päivittäistä työtä, sillä olemme suoraan yhteydessä loppukäyttäjiiin, jotka raportoivat ongelmista ja esittävät erilaisia työpyyntöjä. Tämä antaa meille arvokasta tietoa siitä, miten käyttäjät kokevat ohjelmistot ja mitä ongelmia he kohtaavat käytön aikana. Vaikka emme varsinaisesti osallistu käyttöliittymäsuunnitteluun, meillä on mahdollisuus vaikuttaa käyttäjäkokemukseen tarjoamalla selkeitä ohjeita ja ratkaisuja käyttäjien haasteisiin. Näin ollen käyttäjien osallistuminen suunnitteluun tapahtuu epäsuorasti palautteen ja tukipyyntöjen kautta.

Iteratiivinen suunnittelu on toinen tärkeä käyttäjäkeskeisen suunnittelun periaate, jossa suunnittelua ja kehitystä toistetaan käyttäjäpalautteen perusteella. Prosessi alkaa yleensä prototyypin kehittämällä, jota testataan käyttäjillä. Saadun palautteen avulla tehdään parannuksia ja seuraava prototyyppi testataan uudelleen. Tätä sykliä toistetaan, kunnes tuote vastaa optimaalisesti käyttäjien tarpeisiin. Iteratiivisen suunnittelun tarkoituksena on jatkuvasti parantaa tuotetta ja ratkaista mahdolliset ongelmat ennen lopullista julkaisua.

Esimerkiksi iPhoneen kehityksessä iteratiivinen suunnitteluprosessi oli avainasemassa. Apple testasi jatkuvasti uusia käyttöliittymän eleitä, kuten pyyhkäisyä ja nipistystä, ja iteratiivisen palautteen kautta nämä eleet hiottiin mahdollisimman intuitiivisiksi käyttäjille. Samoin omassa työssäni Service Deskissä olemme huomanneet, että ongelmien toistuva esiintyminen antaa meille mahdollisuuden päivittää ohjeitamme ja prosessejamme, jolloin käyttäjät oppivat paremmin käyttämään ohjelmistoja.

Käyttäjäpalautteen systemaattinen kerääminen ja analysointi on välttämätöntä käyttäjäkeskeisen suunnittelun periaatteiden mukaisessa kehitysprosessissa. Käyttäjätestauksen, kyselyiden ja tukipalveluiden kautta saatu palaute tarjoaa arvokasta tietoa siitä, miten käyttäjät kokevat tuotteen ja mitä parannuksia voitaisiin tehdä. Tämä tieto mahdollistaa iteratiivisen kehityksen ja varmistaa, että lopullinen tuote vastaa käyttäjien odotuksia ja tarpeita. (Norman, Gibbons)

Service Deskissä käyttäjäpalaute on päivittäinen osa työtäni. Jokaisesta ratkaistusta työpyynnöstä käyttäjä voi antaa palautetta, jonka perusteella arvioimme, kuinka hyvin onnistuimme ongelman ratkaisussa ja kuinka selkeät ohjeet olivat. Palautteen avulla voimme kehittää tukiprosessejamme ja päivittää ohjeitamme, jotta käyttäjäkokemus paranee. Tämä on eräänlaista käyttäjäkeskeisen suunnittelun soveltamista käytännössä, sillä pyrimme jatkuvasti parantamaan palvelun laatua käyttäjiltä saadun palautteen avulla.

Käyttäjäkeskeinen suunnittelu eroaa perinteisistä suunnittelumenetelmistä erityisesti siinä, että se painottaa käyttäjän aktiivista osallistumista

suunnitteluprosessin kaikissa vaiheissa. Perinteiset suunnittelumallit saattavat keskittyä enemmän teknisiin vaatimuksiin tai kehittäjien näkemyksiin, mutta käyttäjäkeskeinen suunnittelu asettaa käyttäjän keskiöön. Tämä lähestymistapa varmistaa, että ohjelmistot ja palvelut vastaavat paremmin loppukäyttäjien tarpeisiin ja tarjoavat paremman käyttökokemuksen. Normanin mukaan käyttäjäystävällisen suunnittelun ytimessä on se, että käyttäjällä on selkeät signaalit siitä, miten tuotetta käytetään, ja että tuotteet ovat intuitiivisia sekä helppokäyttöisiä. Tämä näkyy selkeästi myös käytännön työssäni, kun autamme käyttäjiä ratkaisemaan ongelmia ja annamme selkeitä ohjeita portaaleissa. (Norman.)

Käyttäjäkeskeinen suunnittelu on elintärkeä osa onnistunutta ohjelmistokehitystä. Se varmistaa, että tuotteet eivät ainoastaan täytä teknisiä vaatimuksia vaan myös tarjoavat sujuvan ja positiivisen käyttäjäkokemuksen. Käyttäjien osallistaminen, iteratiivinen suunnittelu ja palautteen hyödyntäminen ovat avaintekijöitä, joiden avulla ohjelmistot voidaan optimoida vastaamaan loppukäyttäjien todellisia tarpeita. Tämä lähestymistapa on osoittautunut toimivaksi niin omassa työssäni Service Deskissä kuin myös suurissa menestystarinoissa, kuten Applen iPhonen kehityksessä.

## 2.2 Käytettävyyden merkitys ohjelmistokehityksessä

Käytettävyys on keskeinen osa käyttäjäkokemusta ja määrittää pitkälti sen, kuinka onnistuneesti käyttäjät pystyvät suorittamaan ohjelmistossa vaaditut tehtävät. Se ei tarkoita vain sitä, että ohjelmisto toimii teknisesti oikein, vaan että se on intuitiivinen, helppokäyttöinen ja vastaa käyttäjien tarpeita. Hyvä käytettävyys lyhentää oppimiskäyrää, parantaa käyttäytyvyyttä ja vähentää tarvetta asiakastukeen, mikä lopulta voi kasvattaa liiketoiminnan tuottoa ja käyttäjäuskollisuutta.

Käytettävyys voidaan jakaa useisiin osa-alueisiin, joista tärkeimmät ovat tehokkuus, opittavuus, muistettavuus, virheiden minimointi ja käyttäjien

tyytyväisyys. Käyttäjän on pystyttävä suorittamaan tehtävät nopeasti ja sujuvasti. Uudelle käyttäjälle tulisi olla helppoa oppia käyttämään ohjelmistoa ilman pitkiä koulutuksia tai manuaaleja. Käyttäjien on pystyttävä muistamaan ohjelmiston toiminnot, kun he palaavat sen pariin uudelleen. Haluamme päästä tilanteeseen, jossa käyttäjä ei tee virheitä, jos tekee, ne voidaan korjata helposti, käyttäjän kokemus on miellyttävä ja täyttää hänen odotuksensa. Esimerkiksi Nielsen Norman Groupin "Usability 101: Introduction to Usability" -artikkelissa tuodaan esiin, kuinka nämä osa-alueet yhdessä rakentavat kokonaisvaltaisen käyttäjäkokemuksen, joka johtaa parempaan ohjelmiston hyväksyntään ja asiakasuskollisuuteen. (Nielsen.)

Käytettävyys ei ole pelkästään tekninen käsite, vaan sillä on suora vaikutus liiketoimintaan. Hyvä käytettävyys parantaa käyttäjäkokemusta, mikä johtaa pidempään ohjelmiston käyttöaikaan, vähäisempään tukipalvelun tarpeeseen ja lisää asiakastyytyvää asiakastytyvää. Käyttäjäystävällinen ohjelmisto lisää myös asiakasuskollisuutta ja vähentää siirtymistä kilpailijan palveluihin. McKinseyn tutkimuksen mukaan yritykset, jotka panostavat käyttäjäkokemukseen, voivat kasvattaa liikevaihtoaan jopa 200%. Samoin Google on havainnut, että huonosti suunnitellut mobiilisovellukset menettävät jopa 61% käyttäjistä, jos ensimmäinen käyttökokemus on negatiivinen. Tämä korostaa, kuinka käytettävyydellä on suora yhteys ohjelmiston kaupalliseen menestykseen. (Sheppard.)

Omassa työssäni Service Deskissä olen huomannut, kuinka usein käyttäjien ongelmat johtuvat puutteellisesta käytettävyydestä. Monet tukipyynnöt, kuten salasanoihin tai käyttöliittymän hankaluuksiin liittyvät ongelmat, voitaisiin välttää paremmalla käyttöliittymäsuunnittelulla ja selkeämmillä ohjeilla. Esimerkiksi mainitsemani järjestelmä, jossa käyttäjien täytyy avata ikkuna uudessa ikkunassa ja valita "Show all content" voidakseen vaihtaa salasanan, on hyvä esimerkki käytettävyysoongelmasta. Ongelma ei ole tekninen vika, vaan käyttöliittymän epäintuitiivisuus, joka aiheuttaa tarpeetonta turhautumista ja lisää työmäärää asiakastukipalveluille.

Hyvä käytettävyys voi olla ratkaiseva kilpailutekijä. Esimerkiksi Applen iPhone on erinomainen esimerkki onnistuneesta käytettävyydestä. iPhoneen intuitiivinen käyttöliittymä, eleohjaukset ja selkeä palaute ovat tehneet siitä helppokäyttöisen niin teknisesti taitaville kuin vähemmän kokeneille käyttäjille. Käytettävyys oli keskeinen tekijä iPhoneen menestyksessä ja sen vaikutus näkyy edelleen teknologia-alalla. Toisaalta huono käytettävyys voi johtaa merkittäviin ongelmiin. Microsoftin Windows 8 -käyttöjärjestelmä on klassinen esimerkki siitä, kuinka radikaalit muutokset käyttöliittymässä ilman käyttäjien tarpeiden huomioimista voivat johtaa epäonnistumiseen. Huomasin tuttavieni keskuudessa, että he kokivat uuden aloitusnäytön ja puuttuvan Käynnistä-valikon sekavina ja vaikeakäyttöisinä, mikä johti tyytymättömyyteen ja kielteiseen vastaanottoon.

Käytettävyys on kriittinen osa ohjelmistokehitystä, joka vaikuttaa suoraan ohjelmiston onnistumiseen tai epäonnistumiseen markkinoilla. Hyvä käytettävyys parantaa käyttäjäkokemusta, vähentää tukipalvelujen tarvetta ja lisää asiakasuskollisuutta. Käytettävyys ei kuitenkaan rajoitu vain tekniseen toteutukseen, vaan se vaatii jatkuvaa käyttäjäpalautteen keräämistä, analysointia ja parannusten tekemistä.

Kokemukseni Technical Support Specialistina Service Deskissä ovat osoittaneet, kuinka käytettävyys vaikuttaa suoraan loppukäyttäjien ongelmiin ja työpyyntöihin. Tukitiimin jäsenenä minulla on mahdollisuus nähdä, miten ohjelmistojen käytettävyys tai sen puute vaikuttaa käyttäjien päivittäiseen työhön. Selkeät ohjeet ja käyttöliittymäsuunnittelun parannukset voivat vähentää toistuvia ongelmia ja parantaa merkittävästi käyttäjäkokemusta.

### 2.3 Design Thinking -lähestymistapa

Design Thinking on käyttäjäkeskeinen ongelmanratkaisun menetelmä, joka keskittyy syvälliseen ymmärrykseen käyttäjien tarpeista ja toiveista. Se koostuu viidestä päävaiheesta, jotka ovat empatia, määrittely, ideointi, prototypointi ja testaus. Tämän lähestymistavan avulla kehitystiimit voivat tuottaa innovatiivisia ratkaisuja, jotka vastaavat paremmin loppukäyttäjien tarpeisiin.

Ensimmäinen vaihe, empatia, keskittyy käyttäjien tarpeiden ja käyttäytymisen syvälliseen ymmärtämiseen. Tämä vaihe on keskeinen käyttäjäkeskeisessä suunnittelussa, sillä ilman ymmärrystä siitä, mitä käyttäjät todella tarvitsevat, on mahdotonta kehittää toimivia ja käyttäjäystävällisiä ratkaisuja. Empatiaan voidaan päästä erilaisten käyttäjätutkimusten, haastattelujen ja havainnoinnin kautta, jolloin suunnittelijat saavat suoran käsityksen käyttäjien haasteista ja toiveista. (Gibbons.)

Omassa työssäni Service Deskissä kohtaan päivittäin käyttäjiä, joilla on erilaisia teknisiä haasteita. Yksi tärkeä osa työtäni on yrittää ymmärtää, mitä käyttäjä todella tarvitsee tai kokee ongelmaksi, sillä käyttäjän esittämä ongelma ei aina ole teknisesti oikea. Tämän ymmärryksen avulla pystyn ohjaamaan käyttäjiä paremmin ja kehittämään selkeitä ohjeita, jotka auttavat heitä ratkaisemaan ongelmansa itsenäisesti esimerkiksi itsepalveluportaalissa.

Toisessa vaiheessa, määrittelyssä, tunnistetaan ja määritellään ongelmat, joita halutaan ratkaista. Tämän vaiheen tarkoituksena on koota kerätty tieto yhteen ja selventää, mitkä ovat keskeiset ongelmat ja mitä niille voidaan tehdä. Määrittelyvaihe on tärkeä, sillä väärin määritellyt ongelmat voivat johtaa epäonnistuneisiin ratkaisuihin. Esimerkiksi mainitsemani tapaus, jossa käyttäjät eivät onnistuneet vaihtamaan salasanaa ilman selkeitä ohjeita, on hyvä esimerkki määrittelyvaiheen tärkeydestä. Ongelma ei ollut itse järjestelmän tekninen toiminta, vaan käyttöliittymän epäintuitiivisuus, joka esti käyttäjiä onnistumasta tehtävässä ilman apua. Kun ongelma määriteltiin tarkasti, pystyin kehittämään toimivat ohjeet, jotka auttoivat käyttäjiä ratkaisemaan ongelman itsenäisesti.

Ideointivaiheessa tiimit kehittävät erilaisia ratkaisuvaihtoehtoja. Tässä vaiheessa luovuus on tärkeää, sillä tavoitteena on tuottaa mahdollisimman monta ratkaisuehdotusta. Ideointivaiheessa ei keskitytä vielä yksityiskohtiin, vaan tavoitteena on luoda innovatiivisia ja mahdollisesti rohkeita ratkaisuja. (Gibbons.)

Esimerkkinä voin mainita tilanteen, jossa käyttäjillä oli toistuvasti ongelmia tietyn portaalin käytössä. Service Desk -tiimimme yhdessä asiakkuuden kanssa mietti erilaisia keinoja parantaa portaalin käytettävyyttä. Yksi ratkaisu oli luoda yksityiskohtaiset ohjeet, jotka julkaisimme itsepalveluportaalissa. Tämä oli yksinkertainen mutta tehokas ratkaisu, joka auttoi käyttäjiä selviämään ongelmista ilman, että heidän täytyi soittaa tukeen tai laajempaa muutosta itsepalveluportaaliin, mikä tulisi huomattavasti kalliimmaksi. Onkin tärkeä tunnistaa, milloin ohjeet ovat tehokkaampi ratkaisu kuin käyttöliittymän tai järjestelmän uudistaminen.

Design Thinking -lähestymistavan neljäs vaihe keskittyy ratkaisujen konkreettiseen kehittämiseen ja testaamiseen. Prototyyppejä kehitetään ja niitä testataan todellisilla käyttäjillä. Testauksen avulla saadaan arvokasta palautetta, jonka perusteella prototyyppiä voidaan parantaa. Tämä vaihe on tärkeä, sillä se mahdollistaa suunnitelman käytännön toimivuuden arvioinnin ennen lopullista toteutusta. (Gibbons.)

Esimerkkinä tästä voin mainita käyttäjille laatimani ohjeet salasanan vaihtamiseen portaalissa. Aluksi kokeilin useita eri lähestymistapoja ohjeiden kirjoittamiseen ja testasin niitä käyttäjien kanssa. Testauksen jälkeen sain palautetta siitä, mikä ohjeiden osio oli selkeä ja mikä vaati tarkennuksia. Tämä iteratiivinen testausprosessi johti lopulta selkeään ja helppokäyttöiseen ohjeistukseen.

Viimeinen vaihe on iterointi ja jatkuva parantaminen. Tämä vaihe on jatkuva, ja siinä kerätään palautetta käyttäjiltä ja parannetaan ratkaisua edelleen. Iterointi on keskeinen osa Design Thinking -lähestymistapaa, sillä suunnittelun on vastattava käyttäjien muuttuviin tarpeisiin ja haasteisiin. (Gibbons.)

Service Desk -työssäni tämä näkyy siinä, että jatkuvasti päivitetään ohjeistuksia ja tukiprosesseja käyttäjiltä saadun palautteen perusteella. Iteratiivinen parantaminen varmistaa, että tarjoamamme tuki ja ohjeistus vastaa paremmin käyttäjien odotuksia ja tarpeita. Esimerkiksi käyttäjäpalautteen perusteella

olemme muokanneet portaalin ohjeita useita kertoja, jotta ne olisivat selkeämpiä ja helpommin ymmärrettäviä.

Design Thinking on tehokas lähestymistapa ohjelmistokehitykseen, jossa käyttäjäkokemus asetetaan suunnittelun keskiöön. Tämän lähestymistavan avulla voidaan kehittää innovatiivisia ratkaisuja, jotka vastaavat käyttäjien tarpeisiin ja parantavat ohjelmistojen käytettävyyttä. Design Thinking korostaa empatiaa, luovuutta, iteratiivisuutta ja käyttäjien aktiivista osallistamista suunnitteluprosessiin, mikä tekee siitä erityisen soveltuvan monimutkaisten ongelmien ratkaisuun. Käytännön työssäni olen nähnyt, kuinka tärkeää on jatkuvasti kerätä palautetta ja parantaa ratkaisuja käyttäjien tarpeiden pohjalta.

### **3 Teoreettinen viitekehys**

#### **3.1 Affordanssit ja signaalit suunnittelussa**

Affordanssit ja signaalit ovat keskeisiä käsitteitä käyttäjäkeskeisessä suunnittelussa, erityisesti käyttöliittymä- ja ohjelmistokehityksessä. Donald A. Normanin *The Design of Everyday Things* -kirjassa esittelemät termit auttavat suunnittelijoita ymmärtämään, miten käyttäjät tulkitsevat ja vuorovaikuttavat käyttöliittymän kanssa. Affordanssit viittaavat objektin tai käyttöliittymän ominaisuuksiin, jotka viestivät käyttäjälle, kuinka kyseistä objektia tulisi käyttää. Affordanssit eivät ole pelkästään visuaalisia vihjeitä, vaan ne ovat osa objektin sisäistä logiikkaa, joka tekee sen käytön intuitiiviseksi. Esimerkiksi ovenkahva affordoi vetämistä, kun taas nappi affordoi painamista. Käyttöliittymissä esimerkiksi painikkeet affordoi napsauttamista.

Käytettäessä ohjelmistoja, affordanssien tulee olla selkeitä, jotta käyttäjän ei tarvitse arvailla, miten tuotetta käytetään. Esimerkiksi painikkeiden muotoilu, koko ja sijainti voivat viestittää käyttäjälle, että niitä voi painaa. Jos affordanssit eivät ole selkeitä, käyttäjä saattaa tehdä virheitä tai olla käyttämättä ohjelmaa oikein. (Norman.)

Omasta työstäni Service Deskissä voin antaa esimerkin affordansseista, jossa eräässä asiakkuuden järjestelmässä käyttäjät eivät osanneet avata tiettyjä näkymiä, koska käyttöliittymän painikkeet eivät affordoineet selkeästi, mitä toimintoa ne edustivat. Tämä johti siihen, että käyttäjät joutuivat ottamaan meihin yhteyttä ja pyytämään apua toistuvasti. Käyttöliittymän epäselvät affordanssit aiheuttivat näin turhautumista ja lisäsivät tukipyyntöjen määrää. Selkeät ohjeet ja tukiprosessit autoivat osittain, mutta ongelma olisi voitu ratkaista paremmin suunnitelluilla affordansseilla.

Signaalit ovat ne visuaaliset tai auditiiviset vihjeet, jotka ohjaavat käyttäjää käyttämään tuotetta oikein. Ne toimivat selkeinä merkkeinä siitä, mitä käyttäjän tulisi tehdä, ja ne auttavat käyttäjää ymmärtämään tuotteen affordanssit. Esimerkiksi käyttäjä voi ymmärtää, että nappia tulee painaa, kun sen päällä lukee "Klikkaa tästä" tai se on korostettu selkeällä värillä.

Signaalit ovat tärkeitä erityisesti digitaalisissa ympäristöissä, kuten ohjelmistojen käyttöliittymissä, joissa käyttäjän on vaikea päätellä pelkän objektin perusteella, miten sitä käytetään. Esimerkiksi verkkosivuilla sininen alleviivattu teksti toimii signaalina siitä, että kyseessä on linkki, jota voi napsauttaa. (Norman.)

Omassa työssäni Service Deskissä olen usein nähnyt tilanteita, joissa signaalit puuttuvat tai ovat epäselviä. Esimerkiksi ohjeissa, jotka olen laatinut salasanan vaihtoa varten, selkeä signaali "Show All Content" oli tärkeä osa ohjeiden onnistumista. Jos signaalit eivät olisi olleet selkeitä, käyttäjät olisivat todennäköisesti edelleen tehneet virheitä, koska järjestelmä ei itsessään antanut tarpeeksi ohjeita siitä, mitä tehdä.

Affordanssit ja signaalit täydentävät toisiaan. Affordanssi tarjoaa mahdollisuuden toimintaan, ja signaali varmistaa, että käyttäjä ymmärtää, mitä toimintoja on tarjolla. Hyvä suunnittelu ottaa huomioon molemmat elementit ja varmistaa, että käyttäjät voivat suorittaa halutut tehtävät ilman sekaannusta tai tarvetta ulkopuoliselle tuelle. Normanin kirjassa korostetaan, että huono suunnittelu johtaa usein siihen, että käyttäjät tekevät virheitä, koska he eivät

tunnista oikeita affordansseja tai koska signaalit puuttuvat tai ovat epäselviä. Tämä korostuu erityisesti teknisissä ympäristöissä, kuten ohjelmistojen käytössä, jossa käyttäjän täytyy nopeasti ymmärtää, mitä toimintoja hänellä on käytettävissä ja kuinka niitä voidaan käyttää.

Affordanssit ja signaalit ovat kriittisiä käyttäjäkeskeisessä suunnittelussa, erityisesti silloin, kun tavoitteena on tehdä ohjelmistosta intuitiivinen ja helppokäyttöinen. Ilman selkeitä affordansseja ja signaaleja käyttäjät joutuvat arvailemaan, miten käyttöliittymää käytetään, mikä johtaa virheisiin, turhautumiseen ja lisääntyneisiin tukipyyntöihin. (Norman.)

Kokemukseni Service Deskissä on opettanut, että hyvin suunniteltu käyttöliittymä, joka hyödyntää selkeitä affordansseja ja signaaleja, vähentää merkittävästi tukipyyntöjen määrää ja parantaa käyttäjäkokemusta. Täten käyttäjäkeskeinen suunnittelu, jossa huomioidaan nämä periaatteet, ei ainoastaan paranna käytettävyyttä, vaan myös vähentää organisaation resursseja käyttäjätuen tarjoamiseen.

### 3.2 Palaute ja rajoitteet käytettävyydessä

Palaute on olennainen osa käyttäjäkeskeistä suunnittelua ja käytettävyyttä, sillä se antaa käyttäjälle välittömän tiedon siitä, ovatko heidän toimensa onnistuneet. Palaute voi olla visuaalista, auditiivista tai tuntoaistiin perustuvaa. Sen avulla käyttäjät saavat ymmärryksen siitä, mitä ohjelmistossa tapahtuu, mikä vähentää epävarmuutta ja mahdollistaa tehokkaamman toiminnan. Selkeä palaute lisää käyttäjän hallinnan tunnetta ja vähentää virheiden tekemisen riskiä.

Normanin kirjassa *The Design of Everyday Things* palautetta pidetään olennaisena osana käyttöliittymäsuunnittelua. Käyttäjille annettava palaute auttaa heitä ymmärtämään, onko heidän toimintansa onnistunut ja mitä seuraavaksi tulee tehdä. Ilman riittävää palautetta käyttäjät voivat tuntea olonsa epävarmoiksi ja tehdä virheitä, koska he eivät tiedä, miten järjestelmä reagoi heidän toimintoihinsa. Palaute toimii myös oppimisvälineenä, jonka avulla käyttäjät voivat parantaa osaamistaan ohjelmiston käytössä. (Norman.)

Service Desk -kokemukseni perusteella olen nähnyt monia tilanteita, joissa palautteen puuttuminen on aiheuttanut käyttäjille turhautumista. Esimerkiksi salasanan vaihtoportaalissa käyttäjät eivät aina tiedä, mitä heidän pitäisi tehdä seuraavaksi, jos portaalissa ei ole selkeää palautetta siitä, onko toimenpide onnistunut vai ei tai miksi toimenpide ei ole onnistunut ollenkaan. Tällaiset ongelmat aiheuttavat turhia tukipyyntöjä, jotka olisi voitu välttää tarjoamalla parempaa palautetta käyttöliittymässä. Kun olen kirjoittanut ohjeita tällaisten tilanteiden varalle, selkeät ohjeet ja palautteen visualisointi ovat olleet keskeisiä elementtejä.

Rajoitteet ovat toinen keskeinen osa käytettävyyssajattelua. Ne auttavat käyttäjää ymmärtämään, mitä toimintaa ohjelmistolla voi tehdä ja mitä ei. Rajoitteet voivat olla fyysisiä rajoitteita, kuten fyysisten laitteiden tai käyttöliittymän osien toimintatavat, kulttuurisia rajoitteita, jotka perustuvat yhteiskunnan tai kulttuurin käytäntöihin tai loogisia rajoitteita, jotka perustuvat ohjelmiston sisäiseen logiikkaan ja toimintamalleihin.

Normanin mukaan hyvät rajoitteet ohjaavat käyttäjää kohti oikeita toimintatapoja ja estävät virheiden tekemisen. Ne voivat auttaa käyttäjää navigoimaan järjestelmässä intuitiivisesti ja estää väärinkäytön. Esimerkiksi jos käyttöliittymässä tietyt painikkeet eivät ole käytettävissä ennen kuin vaaditut tiedot on syötetty, tämä looginen rajoite estää käyttäjää etenemästä liian pitkälle ennen kuin tarvittavat toimenpiteet on suoritettu. (Norman.)

Omassa työssäni olen nähnyt esimerkkejä siitä, kuinka rajoitteet voivat joko tukea tai haitata käyttäjäkokemusta. Eräessä tapauksessa käyttäjä ei pystynyt vaihtamaan salasanaa, koska käyttöliittymä ei tarjonnut riittäviä signaaleja siitä, mitä toimintoa hänen piti käyttää. Ohjeistus auttaa tämänkaltaisessa tilanteessa tiettyyn pisteeseen asti, mutta lähtökohtaisesti käyttöliittymä olisi pitänyt suunnitella selkeämmin ja poistaa samankaltaiset, mutta eri komponentit.

Rajoitteet ja palaute toimivat yhdessä tehokkaasti parantaakseen käytettävyyttä. Rajoitteet estävät käyttäjää tekemästä virheitä tai toimintavirheitä, kun taas palaute kertoo käyttäjälle, että toimenpide on joko

onnistunut tai epäonnistunut. Ilman näiden kahden elementin tasapainoista käyttöä käyttäjäkokemus voi kärsiä, ja käyttäjät saattavat tehdä enemmän virheitä, jotka johtavat tukipyyntöihin tai ohjelmiston käyttöönoton epäonnistumiseen.

Hyvä esimerkki tästä on tilanteessa, jossa käyttäjä yrittää käyttää ohjelmistoa väärin. Jos ohjelmistossa on hyvin suunnitellut rajoitteet, se estää käyttäjän etenemisen väärällä tavalla. Ja kun käyttäjä saa välitöntä palautetta, hän tietää, mitä on tehtävä seuraavaksi. Tämä on erityisen tärkeää teknisissä ympäristöissä, kuten monimutkaisissa tietojärjestelmissä, joissa käyttäjän toiminta voi helposti johtaa virheisiin. (Norman.)

Palaute ja rajoitteet ovat keskeisiä elementtejä käyttäjäkeskeisessä suunnittelussa, jotka vaikuttavat suoraan ohjelmistojen käytettävyyteen. Hyvin toteutettuna ne auttavat käyttäjää suorittamaan tehtäviä tehokkaasti ja vähentävät virheiden tekemisen riskiä. Service Desk -kokemukseni kautta olen havainnut, kuinka näiden periaatteiden huomioiminen käyttöliittymissä voi vähentää tukipyyntöjä ja parantaa käyttäjäkokemusta. Hyvin suunnitellut palaute- ja rajoitemekanismit eivät pelkästään helpota käyttäjien työtä, vaan myös vähentävät organisaation kustannuksia asiakastuen tarjoamisessa.

### 3.3 Virheiden hallinta ja ehkäisy

Virheiden hallinta ja niiden ehkäisy ovat keskeisiä osa-alueita käyttäjäkeskeisessä suunnittelussa. Donald A. Normanin mukaan hyvin suunniteltu järjestelmä pyrkii estämään virheiden syntymisen ja tarjoaa selkeät keinot virheiden korjaamiseen, mikä parantaa käyttäjäkokemusta merkittävästi. Virheiden hallinnassa tärkeää on sekä ennakoida mahdolliset virheet että tarjota käyttäjälle selkeät ohjeet niiden välttämiseksi ja ratkaisemiseksi. Normanin kirjassa *The Design of Everyday Things* virheiden ehkäisyyn liittyy erityisesti käyttöliittymän suunnittelun selkeys ja ennakoivat mekanismit. Kun käyttöliittymä on selkeä ja antaa käyttäjälle tarpeeksi vihjeitä oikeista toimenpiteistä, virheiden määrä vähenee huomattavasti. Tämän takia

käyttöliittymän on tarjottava riittävästi vihjeitä ja selkeitä affordansseja, jotta käyttäjä voi helposti ymmärtää, miten ohjelmaa käytetään. (Norman.)

Esimerkiksi Service Desk -kokemuksessani olen nähnyt, kuinka selkeät käyttöliittymät vähentävät tukipyyntöjä merkittävästi. Eräessä tapauksessa salasanan vaihtamiseen liittyvät ongelmat olisivat olleet vältettävissä, jos käyttöliittymä olisi selkeämmin ilmaissut, mitä käyttäjän tulisi tehdä. Ohjeiden lisääminen ja visuaaliset vihjeet auttoivat kuitenkin vähentämään virheiden määrää, mikä osoittaa, että ennakoivilla suunnitteluratkaisuilla voidaan estää virheitä.

Virheiden ehkäisy voidaan jakaa kahteen päästrategiaan, jotka ovat virheiden ennaltaehkäisy ja virheiden hallinta niiden ilmetessä. Virheiden ennaltaehkäisy tarkoittaa, että järjestelmä on suunniteltu estämään käyttäjiä tekemästä virheitä ennen kuin ne tapahtuvat. Esimerkiksi toimintojen rajoittaminen niin, että käyttäjä ei voi edetä ilman oikeita tietoja, voi estää virheitä. Myös rajoitteiden käyttäminen käyttöliittymässä, kuten harmaantuneet painikkeet ennen kuin kaikki kentät on täytetty, voi estää käyttäjää tekemästä virheitä.

Jos virheitä tapahtuu, tärkeää on tarjota käyttäjille selkeät keinot niiden korjaamiseksi. Tähän kuuluvat muun muassa virheilmoitukset, jotka kertovat käyttäjälle, mitä meni pieleen ja kuinka virhe voidaan korjata. Hyvin suunnitellut virheilmoitukset vähentävät käyttäjän turhautumista ja parantavat heidän mahdollisuuksiaan ratkaista ongelma itsenäisesti. (Laubheimer.)

Service Deskissä virheiden hallinta on erityisen tärkeää. Monissa tapauksissa käyttäjät joutuvat ottamaan yhteyttä tukeen, koska he eivät tiedä, miten ratkaista virheellinen tilanne. Selkeät virheilmoitukset ja ohjeet siitä, kuinka korjata virheet, vähentäisivät merkittävästi tukipyyntöjen määrää. Tämän vuoksi ohjeiden selkeyttäminen on ollut yksi tärkeimmistä parannuksista, joita olen tehnyt omassa työssäni. Kun esimerkiksi kirjoitin tarkempia ohjeita salasanan vaihtamisesta, huomasin, että käyttäjät tekivät vähemmän virheitä ja pystyivät ratkaisemaan ongelmat itsenäisesti.

Virheiden hallinta ei rajoitu pelkästään virheilmoituksiin ja ennaltaehkäisyyn. Käyttäjien tukeminen virhetilanteissa tarkoittaa myös, että järjestelmä tarjoaa käyttäjälle mahdollisuuden palauttaa virheelliset toimenpiteet tai aloittaa uudelleen ilman suurta turhautumista. Esimerkiksi "peruuta"-toiminto, joka mahdollistaa käyttäjän kumoamaan virheellisen toiminnon, parantaa käyttökokemusta ja vähentää stressiä. (Laubheimer.)

Service Deskissä näen usein, kuinka käyttäjät tarvitsevat tukea, kun he ovat tehneet virheen. Hyvät ohjeet ja selkeät palautemekanismit käyttöliittymässä vähentävät heidän tarvettaan ottaa yhteyttä tukeen ja parantavat heidän kokemustaan ohjelmiston käytöstä. Tämän takia virheiden ehkäisy ja käyttäjien tukeminen ovat avainasemassa, kun pyritään vähentämään turhautumista ja parantamaan ohjelmiston käytettävyyttä.

Virheiden hallinta ja ehkäisy ovat keskeisiä elementtejä käyttäjäkeskeisessä suunnittelussa. Ennaltaehkäisevät mekanismit ja selkeät virheilmoitukset ovat välttämättömiä, jotta käyttäjät voivat toimia tehokkaasti ja minimoida virheiden tekemisen riskit. Omassa työssäni Service Deskissä olen nähnyt, kuinka hyvin suunnitellut käyttöliittymät ja ohjeet voivat vähentää virheiden määrää ja tukea käyttäjiä virhetilanteissa. Käyttäjäkeskeisessä suunnittelussa on tärkeää, että virheiden hallinta on integroitu osaksi suunnitteluprosessia, jotta lopputuote on mahdollisimman intuitiivinen ja virheettömästi käytettävissä.

## **4 Case tutkimus: Applen iPhone**

### **4.1 Käyttäjätutkimuksen rooli iPhonen kehityksessä**

Applen iPhonen kehitysprosessi on erinomainen esimerkki siitä, kuinka käyttäjäkeskeinen suunnittelu voi ohjata menestyksekkään tuotteen luomista. Alusta alkaen Apple keskittyi käyttäjien tarpeiden ja odotusten ymmärtämiseen sekä heidän käyttäytymisensä havainnointiin, mikä mahdollisti innovatiivisten ratkaisujen kehittämisen, kuten kosketusnäytön käyttöliittymän ja helppokäyttöisyyden korostamisen. Applen käyttäjätutkimukset ja havainnot auttoivat määrittelemään iPhonen keskeiset ominaisuudet. Perinteisistä

matkapuhelimista poiketen iPhone ei sisältänyt fyysistä näppäimistöä, vaan Apple päätti luoda täysin uudenlaisen käyttöliittymän, joka perustui kosketusnäyttöön ja eleisiin, kuten pyyhkäisyyn ja nipistykseen. Tämän päätöksen pohjalta Apple suoritti laajoja käyttäjätestejä varmistaakseen, että uusi teknologia olisi käyttäjien kannalta intuitiivinen ja helposti omaksuttavissa. (Made In Tandem)

Service Desk -kokemuksistani tiedän, kuinka tärkeää on ymmärtää käyttäjien tarpeet ja heidän kohtaamansa haasteet. Käyttäjätutkimusten avulla voidaan saada selkeä käsitys siitä, miten loppukäyttäjät käyttävät tuotetta ja mitä ongelmia he kohtaavat. Tässä iPhone tapauksessa Apple havaitsi, että monet käyttäjät kokivat fyysisen näppäimistön rajoittavana ja epämukavana, mikä johti täysin uuden, eleohjattavan kosketusnäytön suunnitteluun. Tämä osoittaa, kuinka käyttäjien kuunteleminen ja heidän tarpeidensa ymmärtäminen voi muuttaa koko tuotekehityksen suuntaa.

Käyttäjätutkimuksen merkitys näkyy myös siinä, kuinka Apple testasi toistuvasti tuotteitaan. Normanin teorian käyttäjäkeskeisestä suunnittelusta ja käytettävyydestä tulivat selkeästi esille tässä prosessissa. iPhone ensimmäisiä prototyyppejä testattiin laajasti, ja palautteen perusteella Apple teki useita muutoksia varmistaakseen, että laitteen käyttö olisi mahdollisimman intuitiivista. Tätä prosessia voidaan kutsua jatkuvaksi oppimiseksi ja parantamiseksi, jossa käyttäjän tarpeet ja odotukset ovat keskiössä. Applen tutkimukset ja testit kohdistuivat erilaisiin käyttäjäryhmiin, mikä mahdollisti tuotteen optimoinnin laajalle käyttäjäkunnalle. Tämä oli tärkeää erityisesti siksi, että iPhone ei ollut suunnattu vain teknisesti kokeneille käyttäjille, vaan myös vähemmän teknisesti taitaville. Näin Apple pystyi laajentamaan markkinoitaan ja tarjoamaan tuotteen, joka oli sekä teknisesti edistynyt että helposti omaksuttava laajalle käyttäjäkunnalle.

Käyttäjätutkimukset ja havainnot eivät vain ohjanneet teknisiä ratkaisuja, vaan myös koko iPhone suunnittelun filosofiaa. Esimerkiksi eräässä tutkimuksessa Apple havaitsi, että käyttäjät eivät pitäneet liian monimutkaisista valikoista ja preferoivat selkeyttä ja yksinkertaisuutta. Tämä johti käyttöliittymän

minimalistiseen muotoiluun ja yksinkertaistettuihin toimintoihin, mikä sittemmin muodostui iPhoneen tavaramerkiksi. (Tibidavid.)

Omassa työssäni olen usein kohdannut tilanteita, joissa käyttäjän ongelmat johtuvat ohjelmiston liian monimutkaisesta käyttöliittymästä tai selkeiden ohjeiden puutteesta. Applen tapa integroida käyttäjätutkimusten havainnot osaksi koko suunnitteluprosessia osoittaa, kuinka käyttäjäkokemuksen yksinkertaistaminen voi parantaa tuotteen käytettävyyttä ja vähentää käyttäjien turhautumista. Käyttäjäkeskeinen tutkimus ei siis pelkästään paranna teknistä ratkaisua, vaan myös koko käyttökokemusta, mikä on ratkaisevaa tuotteen menestykselle.

Käyttäjätutkimus oli ratkaisevan tärkeässä roolissa iPhoneen kehitysprosessissa. Apple ymmärsi, että loppukäyttäjän kokemukset ja tarpeet ovat keskeisiä tuotekehityksessä, ja käytti tätä tietoa hyväkseen kehittäessään innovatiivisia ratkaisuja, kuten kosketusnäytön käyttöliittymän ja minimalistisen muotoilun. Tämä filosofia perustui vahvasti käyttäjäkeskeiseen suunnitteluun, jonka avulla tuotteesta saatiin käyttäjäystävällinen, helposti omaksuttava ja teknologisesti edistyksellinen. Näiden oppien soveltaminen ohjelmistokehitykseen voi auttaa luomaan tuotteita, jotka eivät ainoastaan vastaa teknisiä vaatimuksia, vaan myös ylittävät käyttäjien odotukset ja parantavat käyttäjäkokemusta.

## 4.2 Iteratiivinen suunnittelu ja prototypointi

Iteratiivinen suunnittelu ja prototypointi ovat keskeisiä osia käyttäjäkeskeisessä suunnittelussa. Iteratiivisessa prosessissa tuotetta kehitetään ja parannetaan jatkuvasti käyttäjäpalautteen ja testaustulosten pohjalta. Tämä lähestymistapa mahdollistaa tuotteen jatkuvan kehityksen ja optimoinnin siten, että lopputulos vastaa mahdollisimman hyvin käyttäjien tarpeita.

Iteratiivinen suunnittelu tarkoittaa suunnitteluprosessia, jossa tuotetta kehitetään asteittain tekemällä pieniä muutoksia ja parannuksia palautteen perusteella. Tämä prosessi varmistaa, että kehittäjät eivät jää kiinni alunperin tehtyihin virheisiin, vaan he voivat oppia jokaisesta versiosta ja tehdä korjauksia.

Jokainen iteraatio testataan käyttäjillä, ja testitulosten perusteella tehdään muutoksia seuraavaan versioon. (Norman.)

Omassa työssäni Service Deskissä olen usein huomannut, että prosessin jatkuva parantaminen on tärkeää. Esimerkiksi ohjeita kirjoittaessani olen huomannut, että ensimmäinen versio ei aina ole riittävän selkeä kaikille käyttäjille, joten olen parantanut ohjeita käyttäjiltä saamani palautteen perusteella. Tämä iteratiivinen lähestymistapa on myös keskeistä ohjelmistojen kehityksessä, jossa lopullinen tuote saadaan optimoitua vasta useiden testikierrosten jälkeen.

Applen iPhoneen kehityksessä prototyyppejä käytettiin laajasti osana iteratiivista suunnittelua. Ensimmäisten prototyyppien tarkoituksena oli testata, kuinka käyttäjät suhtautuvat täysin uuteen tapaan käyttää matkapuhelinta, erityisesti kosketusnäyttöön perustuvaan käyttöliittymään. Prototyyppejä testattiin käyttäjillä, ja heidän kokemuksiensa ja palautteidensa perusteella käyttöliittymää parannettiin. Tämä prosessi toistui useita kertoja, kunnes iPhoneen käyttöliittymä oli optimoitu vastaamaan mahdollisimman hyvin käyttäjien tarpeita. (Made In Tandem.)

Normanin teorian mukaisesti prototyypit ovat olennainen osa käyttäjäkeskeistä suunnittelua, koska ne antavat kehittäjille konkreettisen mahdollisuuden testata suunnitteluratkaisuja käytännössä. Tämä ei ainoastaan helpota ohjelmiston käytettävyyden testaamista, vaan myös mahdollistaa nopeiden muutosten tekemisen ennen tuotteen lopullista julkaisua. (Norman.)

Iteratiivinen suunnittelu näkyy myös omassa työssäni, erityisesti ohjeiden ja tukiprosessien kehittämisessä. Kirjoittaessani ohjeita esimerkiksi salasanojen vaihtamiseen liittyen, olen huomannut, että käyttäjäpalautteen perusteella ohjeita on jouduttu parantamaan useita kertoja. Ensimmäinen versio ohjeista ei ollut riittävän selkeä kaikille käyttäjille, mutta jokaisen iteraation jälkeen saimme tarkempaa tietoa siitä, mikä osa ohjeista oli epäselvä. Tämän jälkeen teimme parannuksia, kunnes lopputulos oli mahdollisimman käyttäjäystävällinen. Iteratiivinen lähestymistapa varmistaa, että ohjeet ja tukiprosessit vastaavat

paremmin käyttäjien tarpeisiin ja vähentävät virheiden määrää. Tämä sama prosessi on avainasemassa ohjelmistokehityksessä, jossa jokainen iteraatio vie tuotteen lähemmäksi lopullista, optimoitua versiota.

Prototyypointi mahdollistaa myös uusien ideoiden testaamisen ilman, että kaikki resurssit sidotaan lopulliseen tuotteeseen. Prototyyppejä voidaan käyttää riskienhallintatyökaluna, sillä ne antavat mahdollisuuden testata, kuinka käyttäjät reagoivat uusiin ominaisuuksiin tai käyttöliittymäratkaisuihin, ennen kuin ne otetaan laajemmin käyttöön. Apple hyödynsi tätä prosessia kehittäessään eleohjausta ja kosketusnäytön käyttöliittymää iPhonessa. Prototyyppien avulla Apple pystyi keräämään arvokasta tietoa käyttäjien vuorovaikutuksesta uuden teknologian kanssa ja tekemään tarvittavat muutokset varhaisessa vaiheessa, mikä johti lopulta menestykselliseen tuotteeseen.

Iteratiivinen suunnittelu ja prototyypointi olivat keskeisessä asemassa iPhonen kehitysprosessissa. Applen kyky testata jatkuvasti käyttäjiä ja kerätä palautetta varmisti, että jokainen uusi iteraatio oli parempi kuin edellinen. Tämä prosessi mahdollisti sen, että iPhone ei ollut pelkästään teknisesti edistynyt tuote, vaan myös käyttäjäystävällinen ja helposti omaksuttava. Jokainen prototyyppi toi mukanaan uusia oivalluksia siitä, miten käyttäjät vuorovaikuttavat tuotteen kanssa, mikä johti lopulta siihen, että iPhonen käyttöliittymästä tuli yksi kaikkien aikojen parhaiten suunnitelluista.

Iteratiivinen suunnittelu ja prototyypointi eivät ainoastaan parantaneet tuotteen käytettävyyttä, vaan myös lyhensivät kehitysaikaa ja vähensivät kustannuksia, koska suurempia suunnittelumuutoksia ei tarvinnut tehdä myöhäisessä vaiheessa. Tämä prosessi on olennainen osa modernia ohjelmistokehitystä ja osoittaa, kuinka tärkeää on kerätä jatkuvasti palautetta ja tehdä pieniä, hallittavia muutoksia tuotteen kehityksen aikana. (Moran.)

Iteratiivinen suunnittelu ja prototyypointi ovat kriittisiä tekijöitä onnistuneessa käyttäjäkeskeisessä suunnittelussa. Ne mahdollistavat tuotteen jatkuvan parantamisen ja optimoinnin käyttäjäpalautteen perusteella. Omassa työssäni

Service Deskissä olen nähnyt, kuinka tärkeää on, että tukiprosessit ja ohjeet kehittyvät jatkuvasti käyttäjiltä saadun palautteen perusteella. Tämä sama prosessi on ollut keskeinen myös iPhoneen menestyksessä, sillä Apple hyödynsi käyttäjien antamaa palautetta ja paransi tuotetta jokaisen iteraation myötä. Tämä johti lopulta siihen, että iPhone ei ollut vain teknologisesti edistynyt, vaan myös intuitiivinen ja helppokäyttöinen tuote.

### 4.3 Käytettävyydestaus iPhoneen kehitysprosessissa

Käytettävyydestaus oli olennainen osa Applen iPhoneen kehitysprosessia, sillä sen avulla voitiin varmistaa, että laitteen käyttöliittymä oli käyttäjille intuitiivinen ja helposti omaksuttava. Apple testasi prototyyppejä laajasti todellisilla käyttäjillä, mikä auttoi havaitsemaan käyttöliittymän ongelmakohdat ja tekemään tarvittavat parannukset ennen lopullista julkaisua.

Käytettävyydestaus jakautui useaan vaiheeseen. Ensimmäiset testit keskittyivät siihen, miten käyttäjät suhtautuivat täysin uuteen kosketusnäytön käyttöliittymään, joka ei sisältänyt perinteistä fyysistä näppäimistöä. Apple halusi varmistaa, että kosketusnäytön eleohjaukset, kuten pyyhkäisy ja nipistys, olivat helposti ymmärrettäviä ja luonnollisia käyttää. Käyttäjätestauksen avulla havaittiin, että monet käyttäjät kokivat fyysisen näppäimistön tarpeettomana ja pitivät kosketusnäyttöä innovatiivisena tapana käyttää puhelinta. Apple keräsi käyttäjäpalautetta jokaisesta prototyypistä ja iteratiivisen prosessin avulla paransi jatkuvasti käyttöliittymää. Tämä testausprosessin toistaminen antoi Applelle mahdollisuuden hioa yksityiskohtia, kuten ikonien sijoittelua, eleiden tarkkuutta ja painikkeiden käyttöä, jotta ne vastaisivat mahdollisimman hyvin käyttäjien odotuksia ja tarpeita.

Käytettävyydestauksessa Apple hyödynsi useita eri menetelmiä.

Käyttäjähastattelut ja käytettävyydestetit prototyyppien kanssa olivat tärkeä osa prosessia. Testien aikana käyttäjät saivat käyttää laitetta erilaisissa todellisissa skenaarioissa, mikä auttoi Applea keräämään dataa siitä, miten ihmiset todellisuudessa vuorovaikuttivat laitteen kanssa. Tämä antoi mahdollisuuden

nähdä, mitkä osat käyttöliittymästä toimivat odotetusti ja missä oli parannettavaa. (Sush Labs.)

Service Desk -kokemukseni on opettanut minulle, kuinka tärkeää on nähdä käyttäjän todellinen vuorovaikutus ohjelmiston tai laitteen kanssa.

Käytettävyydestit eivät ainoastaan paljasta teknisiä ongelmia, vaan myös sen, miten käyttäjät ymmärtävät ja kokevat järjestelmän. Tämä oli ratkaisevan tärkeää myös Applen tapauksessa, kun se pyrki luomaan laitteen, joka oli sekä teknisesti edistynyt että intuitiivinen.

Apple testasi iPhonea monenlaisilla käyttäjäryhmillä varmistaakseen, että laite oli helppokäyttöinen laajalle käyttäjäkunnalle. Testiryhmiin kuului sekä teknisesti taitavia että vähemmän teknisesti orientoituneita käyttäjiä. Tämä laaja käyttäjäjoukko mahdollisti sen, että Apple sai monipuolista palautetta, jonka avulla se pystyi kehittämään tuotteen, joka oli helposti omaksuttavissa kaikille käyttäjille. Testitulosten analysointi oli keskeinen osa Applen kehitysprosessia. Käyttäjien tekemät virheet ja heidän kommenttinsa antoivat arvokasta tietoa siitä, mitkä osat käyttöliittymästä olivat intuitiivisia ja mitkä tarvitsivat parannusta. Iteratiivisen suunnitteluprosessin ansiosta Apple pystyi tekemään nopeita parannuksia ja testaamaan ne uudelleen käyttäjillä, mikä johti parempaan lopputuotteeseen jokaisessa iteraatiossa.

Esimerkiksi iPhoneen ikonien sijoittelua ja eleitä parannettiin useaan otteeseen palautteen perusteella. Alkuperäiset suunnitelmat eivät täysin vastanneet käyttäjien odotuksia, mutta kun käyttäjäpalautetta analysoitiin, Apple muutti esimerkiksi eleiden herkkyyttä ja ikonien sijaintia helpottamaan käyttäjien toimintaa. Tämä prosessi mahdollisti tuotteen hionnassa sen, että käyttöliittymä oli mahdollisimman selkeä ja yksinkertainen. Samalla se tarjosi täyden toiminnallisuuden. Käytettävyydestaus toi esiin myös tärkeän periaatteen. Apple ei halunnut vain testata teknisiä ratkaisuja vaan myös sitä, miten käyttäjät vuorovaikuttavat laitteen kanssa ja miten he ymmärtävät sen toiminnallisuudet. Tämä käyttäjäkeskeinen lähestymistapa oli keskeinen iPhoneen menestyksen kannalta. Testaamalla prototyyppisiä laajasti ja hyödyntämällä saatuja

palautteita Apple pystyi luomaan laitteen, joka täytti käyttäjien tarpeet ja ylitti heidän odotuksensa. (Moran.)

Omassa työssäni käytettävyystestaus on tärkeä osa tukiprosesseja. Näen päivittäin, kuinka käyttäjät reagoivat ohjelmistojen käyttöliittymiin ja millaisia ongelmia he kohtaavat. Kun ongelmat toistuvat, ne usein viittaavat käyttöliittymän epäselvyyteen tai käytettävyyden puutteisiin. Tämän vuoksi käytettävyystestaus ja palautteen kerääminen ovat kriittisiä ohjelmistojen kehityksessä. Kun nämä ongelmat analysoidaan ja niihin reagoidaan iteratiivisen suunnittelun avulla, ohjelmiston käytettävyys paranee, käyttäjäkokemus paranee merkittävästi.

Käytettävyystestaus oli olennainen osa Applen iPhoneen kehitysprosessia, ja se varmisti, että laitteen käyttöliittymä oli intuitiivinen ja käyttäjäystävällinen. Apple käytti laajasti käyttäjäpalautetta ja testituloksia parantaakseen laitteen käytettävyyttä ja muuttaakseen suunnittelua, jotta se vastasi paremmin käyttäjien odotuksia. Iteratiivinen testaus ja jatkuva palauteprosessi mahdollistivat sen, että jokainen iPhoneen versio oli parempi kuin edellinen, mikä johti lopulta erittäin menestykselliseen ja laajasti hyväksytyyn tuotteeseen. Käytettävyystestaus ei pelkästään parantanut iPhoneen teknistä suorituskykyä vaan myös varmistui siitä, että käyttäjien kokemus oli positiivinen ja sujuva, mikä teki iPhoneesta yhden kaikkien aikojen menestyneimmistä teknologisista tuotteista.

#### 4.4 Uusien standardien luominen ja innovatiiviset suunnittelupäätökset

Applen iPhone asetti uusia standardeja niin käyttöliittymäsuunnittelussa kuin koko teknologia-alalla. Sen innovatiiviset suunnittelupäätökset, kuten fyysisen näppäimistön poistaminen ja kosketusnäytön käyttöönotto, mullistivat älypuhelimien käytettävyyden. Tämä rohkea päätös oli yksi merkittävimmistä iPhoneen kehityksen aikana, ja se loi täysin uuden tavan käyttää puhelinta eleiden avulla.

Ennen iPhonea useimmat älypuhelimet sisälsivät fyysisen näppäimistön, jonka avulla käyttäjät syöttivät tekstiä ja navigoivat käyttöliittymässä. Apple päätti kuitenkin poistaa näppäimistön kokonaan ja korvata sen kosketusnäytöllä, jossa eleet, kuten pyyhkäisy ja nipistys, mahdollistivat navigoinnin ja tekstinsyötön. Tämä oli radikaali muutos, mutta käyttäjien suorittamat käytettävyytestit osoittivat, että kosketusnäytön käyttö oli monille käyttäjille intuitiivinen ja luonnollinen tapa olla vuorovaikutuksessa laitteen kanssa. Applen päätös luottaa täysin kosketusnäyttöön ja hylätä perinteinen näppäimistö vaikutti koko teknologia-alaan, ja pian muut valmistajat seurasivat sen jalanjälkiä. Applen kosketusnäyttö asetti uuden standardin älypuhelimille ja tarjosi käyttäjille entistä enemmän mahdollisuuksia mukauttaa laitteen käyttöä. Tämän suunnittelupäätöksen avulla Apple erottui kilpailijoistaan ja muutti sen, miten älypuhelimia käytetään. (Bootcamp.)

Käytännön työkokemuksessani Service Deskissä olen nähnyt, kuinka tärkeää on, että käyttöliittymät ovat helppokäyttöisiä ja intuitiivisia, sillä nämä tekijät voivat vähentää käyttäjien virheitä ja parantaa heidän kokemustaan ohjelmiston kanssa.

Toinen merkittävä innovaatio oli App Storen perustaminen, joka loi täysin uuden ekosysteemin sovelluskehitykselle. App Store avasi mahdollisuuden kolmansien osapuolten kehittäjille luoda sovelluksia iPhoneille, mikä lisäsi huomattavasti laitteen käyttötapoja ja mahdollisti monipuolisen sovellusvalikoiman. Tämä strategia ei ainoastaan lisännyt iPhoneen houkuttelevuutta käyttäjien keskuudessa, vaan myös loi kokonaan uuden liiketoimintamallin, jossa sovelluskehittäjät voivat ansaita rahaa myymällä sovelluksia suoraan käyttäjille.

App Store on ollut yksi Applen menestyneimmistä päätöksistä, ja sen vaikutukset näkyvät laajasti koko teknologia-alalla. Monet muut älypuhelinvalmistajat, kuten Samsung ja Google, seurasivat Applen mallia ja perustivat omat sovelluskauppansa. App Storen avulla Apple pystyi luomaan vahvan ekosysteemin, joka sitoutti käyttäjät laitteeseen ja lisäsi heidän tyytyväisyyttään. Apple on myös tunnettu laitteiston ja ohjelmiston saumattomasta integraatiosta. iPhoneen suunnittelussa Apple keskittyi siihen,

että laitteiston ja ohjelmiston yhdistäminen toimisi niin hyvin, että käyttäjien ei tarvitsisi kohdata teknisiä ongelmia tai ristiriitoja. Tämä päätös oli keskeinen osa Applen strategiaa ja se mahdollisti sen, että laite tarjosi sujuvan ja johdonmukaisen käyttökokemuksen, jota kilpailijat eivät pystyneet tarjoamaan yhtä hyvin. (Bootcamp.)

Omassa työssäni Service Deskissä olen huomannut, että yksi suurimmista ongelmista käyttäjille on teknisten ongelmien ja ohjelmistojen yhteensopimattomuus. Suurin asiakkuutemme käyttää sekaisin Windows-, Mac-, Android- ja iPhone-laitteita. Kun laitteisto ja ohjelmisto eivät toimi yhteen saumattomasti, käyttäjät kohtaavat toistuvia virheitä ja ongelmia, jotka heijastuvat palvelupyynnöinä tukitiimeille. Applen lähestymistapa vähensi merkittävästi tällaisia ongelmia ja tarjosi käyttäjille sujuvan kokemuksen, mikä paransi heidän tyytyväisyyttään.

Applen suunnittelupäätökset eivät olleet vain teknisiä, vaan myös strategisia. Päätös poistaa fyysinen näppäimistö ja korvata se kosketusnäytöllä oli rohkea, mutta myös kaupallisesti onnistunut. Samoin App Storen perustaminen ja laitteiston sekä ohjelmiston integrointi asettivat uusia standardeja, joita muut teknologiayritykset seurasivat. Nämä päätökset ovat osoitus siitä, että innovatiiviset ratkaisut voivat muuttaa koko toimialaa ja määritellä uusiksi sen, miten käyttäjät vuorovaikuttavat teknologian kanssa.

Monet älypuhelinvalmistajat, kuten Samsung, Xiaomi ja Huawei, ovat seuranneet Applen esimerkkiä ja ottaneet käyttöön samankaltaisia suunnitteluratkaisuja, kuten kosketusnäytöt ja sovelluskaupat. Näiden päätösten vaikutukset näkyvät edelleen, ja ne ovat olleet kriittisiä älypuhelimien kehitykselle. (Made In Tandem.)



Kuva 1. iPhone-käyttöliittymän kehitys versiosta iOS 1 versioihin 6 ja 7.

Applen rohkeat ja innovatiiviset suunnittelupäätökset iPhoneen kehityksessä asettivat uusia standardeja käyttöliittymäsuunnittelussa ja koko teknologia-alalla. Kosketusnäytön käyttöönotto, App Storen perustaminen ja laitteiston sekä ohjelmiston saumaton integraatio osoittivat, kuinka käyttäjäkeskeinen suunnittelu voi johtaa merkittäviin innovaatioihin ja menestykseen. Service Desk -kokemukseni perusteella tiedän, kuinka tärkeää on, että tekniset ratkaisut ovat paitsi edistyksellisiä myös helppokäyttöisiä ja intuitiivisia. Applen päätökset eivät ainoastaan parantaneet teknologiaa, vaan myös muuttivat sitä, miten ihmiset käyttävät teknologiaa arjessaan.

## 5 Tulokset ja analyysi

### 5.1 Käyttäjäkeskeisen suunnittelun vaikutukset iPhoneen menestykseen

Käyttäjäkeskeinen suunnittelu oli keskeinen tekijä iPhoneen kaupallisessa menestyksessä. Alusta alkaen Applen tavoitteena oli luoda laite, joka vastasi käyttäjien tarpeisiin ja tarjosi intuitiivisen käyttökokemuksen. Tämä saavutettiin käyttäjätutkimusten, prototyyppien kehittämisen ja käytettävyydestestauksen avulla, mikä mahdollisti laitteen parantamisen jatkuvasti ennen sen julkaisua.

iPhoneen suurimpia innovaatioita oli kosketusnäytön ja eleohjauksen käyttöönotto. Aiemmin puhelimissa käytettiin fyysisiä näppäimistöjä, jotka

rajoittivat käyttöliittymän joustavuutta. Applen valitsema kosketusnäyttö, jossa eleet, kuten pyyhkäisy ja nipistys, mahdollistivat navigoinnin, oli radikaali mutta onnistunut ratkaisu. Tämä käyttöliittymä oli käyttäjille luonnollinen ja helposti omaksuttava, ja se johti siihen, että puhelimen käyttö oli selkeää myös teknisesti vähemmän taitaville käyttäjille. Käyttäjäkeskeisen suunnittelun avulla Apple varmisti, että laitteen käyttöliittymä oli paitsi teknisesti edistynyt, myös intuitiivinen. Tämä oli kriittinen tekijä, joka johti iPhoneen menestykseen. (Bootcamp.)

Service Desk -kokemukseni on osoittanut, että jos käyttöliittymä ei ole intuitiivinen, se johtaa usein käyttäjien turhautumiseen ja lisää tukipyyntöjä. iPhoneen tapauksessa Apple onnistui minimoimaan nämä ongelmat, mikä paransi merkittävästi käyttäjäkokemusta.

Käyttäjäkeskeinen suunnittelu vaikutti myös siihen, miten Apple rakensi laajemman ekosysteemin iPhoneelle. App Storen perustaminen mahdollisti kolmansien osapuolten sovelluskehityksen, mikä laajensi huomattavasti laitteen käyttötapoja ja lisäsi sen arvoa käyttäjille. Sovelluskauppa oli intuitiivinen ja helppokäyttöinen, ja se mahdollisti käyttäjille nopean pääsyn tuhansiin eri sovelluksiin. Tämä lisäsi iPhoneen houkuttelevuutta ja kasvatti merkittävästi käyttäjäkantaa. Käyttäjäkeskeinen lähestymistapa näkyi myös siinä, miten Apple teki yhteistyötä sovelluskehittäjien kanssa varmistaakseen, että sovellukset toimivat saumattomasti laitteen käyttöliittymässä. Tämä strategia ei ainoastaan parantanut iPhoneen käyttöarvoa, vaan myös loi sitoutuneen käyttäjäyhteisön, joka hyödynsi laitteen täyttä potentiaalia. (Made In Tandem)

Omassa työssäni olen huomannut, että käyttäjäkeskeinen ohjeistus ja ekosysteemien rakentaminen voi helpottaa käyttäjien työskentelyä ja vähentää tukitarpeita.

Apple käytti laajasti iteratiivista suunnittelua ja käytettävyydestausta varmistaakseen, että jokainen iPhoneen versio paransi edellistä. Tämä prosessi mahdollisti sen, että jokainen iterointi perustui käyttäjäpalautteeseen ja testituloksiin, mikä johti optimoituun käyttöliittymään ja parannettuihin

ominaisuuksiin. Esimerkiksi ikonien sijoittelu ja eleiden herkkyys optimoitiin useiden testikierrosten perusteella, mikä teki laitteesta käyttäjäystävällisemmän. (Made In Tandem.)

Iteratiivinen suunnittelu on myös keskeinen osa omaa työskentelyäni Service Deskissä, jossa ohjeita ja tukiprosesseja parannetaan jatkuvasti käyttäjäpalautteen perusteella. Tämä lähestymistapa on tärkeä, sillä se mahdollistaa jatkuvan parantamisen ja varmistaa, että tuote tai palvelu vastaa aina käyttäjien tarpeisiin. Applen menestys iPhonella osoittaa, kuinka tehokasta iteratiivinen suunnittelu voi olla ohjelmistojen ja laitteiden kehittämisessä.

Käyttäjäpalautteen kerääminen ja sen hyödyntäminen oli keskeistä iPhonen suunnittelussa. Apple otti aktiivisesti vastaan palautetta käyttäjiltä ja hyödynsi sitä parannusten tekemiseen. Tämä auttoi varmistamaan, että iPhone vastasi käyttäjien odotuksia ja tarjosi heille positiivisen kokemuksen. Applen kyky kuunnella käyttäjiä ja tehdä tarvittavia muutoksia johti siihen, että iPhone kehittyi nopeasti ja vastasi paremmin markkinoiden tarpeita. (Made In Tandem; Norman.)

Käyttäjäpalautte on tärkeä osa myös omaa työskentelyäni, ja sen avulla voin parantaa esimerkiksi ohjeita ja tukiprosesseja, jotta ne vastaavat paremmin käyttäjien tarpeita. Applen tapa käyttää palautetta suunnitteluprosessissa on erinomainen esimerkki siitä, kuinka palautteen tehokas hyödyntäminen voi johtaa menestykseen.

Käyttäjäkeskeinen suunnittelu oli yksi tärkeimmistä tekijöistä, joka johti iPhonen menestykseen. Apple panosti käyttäjien tarpeiden ymmärtämiseen ja optimoi laitteen käytettävyyttä jatkuvasti käyttäjäpalautteen ja testitulosten perusteella. Tämä lähestymistapa johti innovatiivisiin ratkaisuihin, kuten kosketusnäytön ja App Storen käyttöönottoon, jotka asettivat uusia standardeja koko alalle. (Made In Tandem.)

Omassa työssäni olen nähnyt, kuinka tärkeää on, että suunnitteluprosessit perustuvat käyttäjäkokemukseen ja jatkuvaan palautteen hyödyntämiseen, ja

iPhone on erinomainen esimerkki siitä, kuinka käyttäjakeskeinen suunnittelu voi johtaa merkittävään kaupalliseen menestykseen.

## 5.2 Haasteiden voittaminen ja oppimiskokemukset

iPhonen kehityksessä Apple kohtasi monia haasteita, jotka olivat ratkaisevan tärkeitä laitteen lopulliselle menestykselle. Yksi merkittävimmistä haasteista oli fyysisen näppäimistön poistaminen ja sen korvaaminen täysin kosketusnäytöllä. Tuohon aikaan markkinoilla olevat puhelimet, kuten BlackBerry, olivat tunnettuja fyysisestä näppäimistöstään, ja siirtyminen pelkkään kosketusohjaukseen herätti aluksi epäilyksiä.

Kosketusnäytön käyttöön liittyi aluksi monia epäilyksiä, sillä markkinoilla ei ollut aiemmin täysin kosketusohjattuja puhelimia, ja käyttäjät olivat tottuneet perinteisiin näppäimistöihin. Apple käytti laajoja käyttäjätutkimuksia ja prototyyppointia varmistaakseen, että kosketusnäyttö olisi mahdollisimman intuitiivinen ja helposti käytettävä. Käytettävyytestauksissa havaittiin esimerkiksi, että monet käyttäjät olivat epävarmoja eleohjauksista, kuten pyyhkäisystä ja nipistyksestä, mutta iteratiivisen testauksen kautta näitä eleitä kehitettiin niin, että ne olivat helppoja oppia ja käyttää. (Bootcamp)

Service Desk -kokemuksistani tiedän, kuinka tärkeää on varmistaa, että uudet käyttöliittymät ovat käyttäjille helposti omaksuttavia. Esimerkiksi ohjeita kirjoittaessani olen kohdannut samanlaisia haasteita, kun uudet järjestelmät tai käyttöliittymät aiheuttavat aluksi hämmennystä käyttäjissä. Tämän vuoksi selkeät ohjeet ja käyttäjäystävälliset ratkaisut ovat olleet olennaisia uusien järjestelmien käyttöönotossa.

Toinen suuri haaste oli luoda toimiva ekosysteemi iPhonelle, joka ei perustunut pelkästään laitteistoon, vaan myös sovelluksiin ja ohjelmistoihin. Tämän ratkaisemiseksi Apple loi App Storen, joka mahdollisti kolmansien osapuolten sovelluskehityksen. Tämä päätös loi täysin uuden tavan käyttää älypuhelimia ja muutti samalla koko mobiilialan kehitysmallia. Haasteena oli varmistaa, että App Store oli riittävän helppokäyttöinen niin kehittäjille kuin käyttäjillekin. Apple

onnistui tässä, koska se keskittyi käyttäjäkokemukseen sekä sovelluskehittäjien että loppukäyttäjien näkökulmasta. (Made In Tandem.)

Omassa työssäni olen nähnyt, kuinka tärkeää on luoda selkeitä ja käyttäjäystävällisiä järjestelmiä, jotka palvelevat monia eri käyttäjäryhmiä. App Store on hyvä esimerkki siitä, kuinka monimutkaisen järjestelmän voi tehdä helposti käytettäväksi sekä kehittäjille että kuluttajille, ja sen onnistuminen perustui vahvasti käyttäjäkeskeiseen suunnitteluun.

Kehitysprosessissa Apple sai paljon palautetta ja kohtasi kriittisiä kysymyksiä siitä, oliko kosketusnäytön käyttö oikea valinta. Iteratiivinen testaus ja jatkuva palautteen kerääminen kuitenkin osoittivat, että käyttäjät tottuivat nopeasti uuteen käyttöliittymään, kunhan ohjeistus ja visuaaliset vihjeet olivat riittävän selkeitä. Apple teki merkittäviä muutoksia kehitysprosessin aikana esimerkiksi ikonien sijoitteluun ja eleiden tarkkuuteen perustuen saatuun palautteeseen. (Bootcamp.)

Service Desk -työssäni käyttäjäpalaute on ollut olennaista prosessien parantamisessa. Usein käyttäjiltä tuleva negatiivinen palaute on arvokasta, sillä sen avulla voidaan korjata suunnittelun puutteita ja parantaa käytettävyyttä. Applen tapa reagoida käyttäjiltä saatuun palautteeseen ja muokata tuotetta sen perusteella on hyvä esimerkki siitä, kuinka käyttäjien tarpeisiin voidaan vastata tehokkaasti.

Yksi suurimmista oppimiskokemuksista iPhoneen kehitysprosessissa oli ymmärtää, kuinka tärkeää on testata jatkuvasti uutta teknologiaa käyttäjillä. Uudet ominaisuudet, kuten eleohjaus ja kosketusnäyttö, eivät olleet aluksi intuitiivisia kaikille käyttäjille, mutta Apple paransi niitä iteratiivisen testauksen avulla. Tämä oppimiskokemus osoittaa, että käyttäjien tarpeet voivat muuttua kehityksen aikana, ja siksi suunnitteluprosessin on oltava joustava ja avoin muutoksille. (Bootcamp.)

Omassa työssäni olen oppinut, että uudet järjestelmät ja ohjelmistot aiheuttavat usein käyttäjissä epävarmuutta, mutta iteratiivisen kehityksen avulla voidaan

tunnistaa ja ratkaista yleisimmät ongelmat. Tämän vuoksi on tärkeää, että suunnitteluprosessi on jatkuva ja reagoi käyttäjäpalautteeseen nopeasti.

Applen haasteiden voittaminen iPhone'n kehitysprosessissa osoittaa, kuinka tärkeää on, että käyttäjäkeskeinen suunnittelu on keskeinen osa kehitystä. Innovatiiviset ratkaisut, kuten kosketusnäytön ja App Storen käyttöönotto, vaativat rohkeutta ja jatkuvaa testausta, mutta iteratiivisen suunnittelun avulla Apple pystyi parantamaan tuotettaan käyttäjäpalautteen perusteella.

Omassa työssäni Service Deskissä olen nähnyt saman periaatteen toimivan. Käyttäjäpalautte ja jatkuva prosessien parantaminen auttavat voittamaan haasteet ja luomaan paremman käyttäjäkokemuksen.

### 5.3 Vaikutukset laajempaan ohjelmistoteollisuuteen

Applen iPhone on jättänyt merkittävän jäljen koko ohjelmistoteollisuuteen, erityisesti käyttäjäkeskeisen suunnittelun, käyttöliittymäsuunnittelun ja mobiilisovellusten ekosysteemin kehityksen osalta. Applen innovatiiviset suunnittelupäätökset ja rohkeat standardit ovat asettaneet uusia vaatimuksia kaikille teknologian ja ohjelmistokehityksen toimijoille. Monet yritykset ovat omaksuneet Applen lähestymistavan, mikä on muuttanut tapaa, jolla ohjelmistoja ja laitteistoja kehitetään.

iPhone'n käyttöliittymän innovatiivinen ja intuitiivinen kosketusnäyttö mullisti käyttöliittymäsuunnittelun. Aikaisemmin monet puhelimet ja muut mobiililaitteet perustuivat fyysisiin näppäimistöihin, ja niiden käyttöliittymät olivat usein monimutkaisia ja vaikeasti navigoitavia. iPhone'n tuoma yksinkertaisuus ja selkeys, jossa käyttäjän toiminnot perustuivat luonnollisiin eleisiin, kuten pyyhkäisyyn ja nipistykseen, asetti täysin uuden standardin, jota muut valmistajat ja ohjelmistokehittäjät seurasivat. (Bootcamp.)

Service Desk -kokemuksissani olen nähnyt, kuinka käyttöliittymän selkeys ja intuitiivisuus vaikuttavat suoraan käyttäjäkokemukseen ja vähentävät tukipyyntöjä. Hyvin suunnitellut käyttöliittymät, kuten iPhone'ssa nähtiin,

mahdollistavat sen, että käyttäjät voivat käyttää järjestelmää ilman turhautumista ja ilman tarvetta tukeen. Tämä vähentää merkittävästi virheiden ja tukipyyntöjen määrää, mikä näkyy parempana asiakastyytyvyytenä ja tehokkaampana resurssien käyttönä.

App Storen perustaminen oli merkittävä innovaatio, joka ei ainoastaan kasvattanut iPhoneen houkuttelevuutta, vaan muutti myös sovellusekosysteemin perusrakenteen. Ennen App Storea sovellusten hankkiminen oli monimutkaista ja hajanaista, eikä ollut yhtenäistä tapaa löytää ja asentaa sovelluksia turvallisesti. Applen luoma App Store toi käyttäjille helpon ja turvallisen tavan ladata ja asentaa sovelluksia, ja se loi uuden liiketoimintamallin, jossa kolmannet osapuolet voivat kehittää sovelluksia ja tavoittaa suoraan kuluttajat.

App Storen menestys kannusti myös muita teknologia- ja ohjelmistoyrityksiä, kuten Googlea ja Microsoftia, luomaan omia sovelluskauppojaan, jotka perustuivat Applen esimerkkiin. Tämä muutos ei ainoastaan vaikuttanut mobiililaitteiden ekosysteemeihin, vaan myös laajensi ohjelmistokehitystä ja loi uusia liiketoimintamahdollisuuksia kehittäjille ympäri maailmaa. (Bootcamp)

Omassa työssäni näen, kuinka ekosysteemien rakentaminen voi helpottaa käyttäjien työskentelyä ja tarjota heille laajemman valikoiman työkaluja ja ratkaisuja. Applen App Store osoitti, että käyttäjien sitouttaminen on tärkeää ekosysteemin menestykselle. Samoin itsepalveluportaaleissa tarjoamani ohjeet ovat osa laajempaa ekosysteemiä, joka helpottaa käyttäjien ongelmien ratkaisua ja vähentää tukitarvetta.

iPhoneen myötä käyttäjäkeskeinen suunnittelu nousi merkittäväksi suunnitteluperiaatteeksi, ei vain mobiililaitteissa, vaan myös laajemmin ohjelmistoteollisuudessa. Applen lähestymistapa korosti käyttäjien tarpeiden asettamista etusijalle ja iteratiivista suunnitteluprosessia, jossa käyttäjäpalaute ohjasi kehitystä. Tämä malli on sittemmin omaksuttu laajasti eri aloilla, kuten verkkosivujen ja sovellusten suunnittelussa. (Sush Labs.)

Nielsen Norman Groupin artikkelit, kuten Usability 101 ja UX Research Cheat Sheet, osoittavat, kuinka käyttäjakeskeisen suunnittelun periaatteita voidaan soveltaa laajasti eri konteksteissa. iPhoneen menestys toimi esimerkkinä siitä, miten käyttäjäpalautteen hyödyntäminen ja suunnitteluprosessin iteratiivinen parantaminen voivat johtaa sekä parempaan käytettävyyteen että kaupalliseen menestykseen.

Applen päätökset iPhoneen kehityksen aikana loivat pitkän aikavälin vaikutuksia ohjelmistoteollisuuteen. Käyttöliittymän selkeys, App Storen luoma ekosysteemi ja käyttäjakeskeisen suunnittelun nousu standardiksi ovat edelleen voimissaan. Monet yritykset ovat omaksuneet nämä periaatteet omiin kehitysprosesseihinsa, ja ne ovat asettaneet uuden lähtökohdan ohjelmistojen käytettävyydelle ja suunnittelulle. (Nielsen; Farrell.)

Esimerkiksi Service Desk -työssäni käyttäjäystävällisyys ja selkeät, helposti navigoitavat käyttöliittymät ovat keskeisiä elementtejä, jotka vähentävät virheiden määrää ja parantavat käyttäjien kokemusta. Applen tapa asettaa käyttäjäkokemus suunnittelun keskiöön osoittaa, kuinka käyttäjakeskeinen lähestymistapa voi vaikuttaa sekä yksittäisten tuotteiden että koko toimialan kehitykseen.

iPhoneen vaikutukset laajempaan ohjelmistoteollisuuteen olivat valtavat. Apple asetti uusia standardeja käyttöliittymäsuunnittelulle, sovellusekosysteemien rakentamiselle ja käyttäjakeskeiselle suunnittelulle, jotka ovat muokanneet koko teknologia-alaa. Applen menestys osoittaa, kuinka tärkeää on asettaa käyttäjien tarpeet suunnittelun keskiöön ja hyödyntää käyttäjäpalautetta suunnitteluprosessin jokaisessa vaiheessa. Tämä lähestymistapa on muuttanut tapaa, jolla ohjelmistoja ja laitteistoja kehitetään, ja sen vaikutukset näkyvät edelleen nykypäivän ohjelmistokehityksessä.

## 6 Pohdintoja tulevaisuuteen

### 6.1 Opit ja parhaat käytännöt

iPhonen kehitysprosessista on opittu monia arvokkaita asioita, jotka voidaan soveltaa tuleviin projekteihin. Applen menestyksen taustalla oli vahvasti käyttäjäkeskeinen suunnittelu, joka perustui jatkuvaan käyttäjäpalautteen keräämiseen, iteratiiviseen suunnitteluun ja rohkeisiin innovaatioihin. Näiden oppien avulla voidaan kehittää tulevaisuuden ohjelmistoprojekteja tehokkaammin ja käyttäjäystävällisemmin.

Yksi keskeisistä opeista on jatkuvan käyttäjäpalautteen merkitys ja sen hyödyntäminen iteratiivisessa suunnittelussa. iPhonen kehitysprosessissa käyttäjien palaute ohjasi jatkuvia parannuksia, erityisesti käyttöliittymän ja eleohjausten osalta. Apple ei pelkästään testannut laitetta kerran, vaan iteratiivinen lähestymistapa mahdollisti käyttäjien tarpeiden ja odotusten huomioimisen jokaisessa kehitysvaiheessa.

Tulevissa projekteissa on tärkeää, että käyttäjät otetaan mukaan suunnitteluprosessiin varhaisessa vaiheessa ja heidän palautteensa analysoidaan jokaisessa kehityssyklissä. Tämä mahdollistaa sen, että mahdolliset ongelmat voidaan ratkaista varhaisessa vaiheessa ja tuote optimoida käyttäjien tarpeita vastaavaksi. Iteratiivinen suunnittelu ei ainoastaan paranna tuotteen käytettävyyttä vaan myös vähentää suurten virheiden ja muutosten riskiä myöhäisessä vaiheessa. (Made In Tandem.)

Tämä käytäntö on myös olennainen osa omia työtehtäviäni Service Deskissä, missä ohjeistuksia ja prosesseja päivitetään jatkuvasti käyttäjien antaman palautteen perusteella.

Toinen oppi on käyttäjäkeskeisen ajattelun keskeinen merkitys. Applen suunnitteluprosessi osoitti, että käyttäjän tarpeiden asettaminen suunnittelun keskiöön voi tuottaa innovatiivisia ja käytännöllisiä ratkaisuja. Esimerkiksi kosketusnäytön käyttöönotto ja fyysisen näppäimistön poistaminen olivat

rohkeita päätöksiä, jotka perustuivat syvälliseen ymmärrykseen käyttäjien tarpeista ja käyttäytymisestä.

Tulevissa projekteissa on tärkeää, että suunnittelutiimit keskustelevat jatkuvasti siitä, miten käyttäjät tulevat vuorovaikuttamaan tuotteen kanssa.

Käyttäjakeskeisyys ei tarkoita vain teknisten vaatimusten täyttämistä, vaan myös käyttäjien emotionaalisten ja kognitiivisten tarpeiden ymmärtämistä.

Käyttöliittymien ja prosessien on oltava intuitiivisia ja selkeitä, jotta käyttäjäkokemus olisi sujuva ja miellyttävä. (Bootcamp.)

Omassa työssäni olen havainnut, että käyttäjäkokemus paranee merkittävästi, kun käyttäjän näkökulma otetaan huomioon. Esimerkiksi itsepalveluportaalin ohjeiden parantaminen käyttäjien palautteen perusteella on auttanut vähentämään tukipyyntöjä ja parantamaan käyttäjien itsepalvelumahdollisuuksia.

Applen päätös luopua fyysisestä näppäimistöstä ja siirtyä kosketusnäytön käyttöön oli innovatiivinen ja riskialtis ratkaisu, mutta se osoittautui menestykselliseksi. Tämä oppi korostaa sitä, että teknologian kehityksessä on tärkeää kokeilla uusia ja rohkeita ratkaisuja, vaikka ne saattavat aluksi herättää epäilyksiä.

Tulevissa projekteissa kehitystiimien tulisi olla avoimia uusille ideoille ja valmiita kokeilemaan vaihtoehtoisia ratkaisuja. Uuden teknologian tai toimintatavan käyttöönotto vaatii usein iteratiivista testausta ja käyttäjäpalautetta, mutta ilman rohkeutta ja innovatiivisuutta todellista edistystä ei saavuteta. Tämä on tärkeä oppi, joka voidaan siirtää kaikkiin ohjelmistoprojekteihin, olipa kyseessä pieni käyttöliittymän muutos tai suurempi teknologinen läpimurto. (Bootcamp.)

iPhonen kehitysprosessi osoitti myös ekosysteemin rakentamisen tärkeyden. App Storen avulla Apple loi alustan, joka ei ainoastaan tarjonnut käyttäjille sovelluksia, vaan myös mahdollisti kolmansien osapuolten kehittäjille mahdollisuuden luoda arvoa laitteelle. Tämä malli on sittemmin omaksuttu

laajasti koko teknologia-alalla, se on osoittautunut menestykseksi niin käyttäjille kuin kehittäjille.

Tulevissa projekteissa tulisi pohtia, miten ekosysteemin rakentaminen voi auttaa tuotteiden laajentamisessa ja niiden elinkaaren pidentämisessä. Innovatiivisten ratkaisujen lisäksi on tärkeää, että käyttäjille tarjotaan mahdollisuuksia laajentaa ja personoida käyttökokemustaan, mikä sitouttaa heitä pitkäaikaisesti tuotteeseen. (Made In Tandem.)

Applen iPhoneen kehitysprosessista opitut käytännöt, kuten jatkuva käyttäjäpalaute, iteratiivinen suunnittelu, rohkeat innovaatiot ja ekosysteemin rakentaminen, tarjoavat arvokkaita oppeja tuleviin ohjelmistoprojekteihin. Omassa työssäni Service Deskissä olen nähnyt, kuinka nämä periaatteet voivat parantaa sekä tuotteiden että palvelujen käytettävyyttä ja käyttäjäkokemusta. Kun nämä opit viedään käytäntöön tulevissa projekteissa, voidaan saavuttaa merkittäviä parannuksia sekä teknisesti että käyttäjäkokemuksen näkökulmasta.

## 6.2 Käyttäjäkeskeisen suunnittelun tulevaisuuden näkymät

Käyttäjäkeskeinen suunnittelu kehittyy jatkuvasti ja mukautuu teknologian ja käyttäjien tarpeiden muuttuessa. Tulevaisuuden näkymät osoittavat, että käyttäjäkeskeinen suunnittelun tulee olemaan yhä tärkeämpi osa ohjelmistokehitystä, erityisesti uusien teknologioiden, kuten tekoälyn, lisätyn todellisuuden (AR) ja virtuaalitodellisuuden (VR) kehityksessä. Nämä teknologiat avaavat uusia mahdollisuuksia käyttäjäkokemuksen parantamiseksi, mutta samalla ne asettavat suunnittelijoille uusia haasteita. (Nielsen.)

Tekoäly ja koneoppiminen ovat nousevia teknologioita, joilla on valtava potentiaali parantaa käyttäjäkokemusta. Tekoälyn avulla järjestelmät voivat oppia käyttäjien tottumuksia ja tarjota entistä henkilökohtaisempia ja mukautettuja käyttökokemuksia. Esimerkiksi suosittelualgoritmit ja käyttäjän käyttäytymiseen perustuvat automatisoidut toiminnot voivat helpottaa käyttäjien arkea ja tehdä järjestelmistä intuitiivisempia. Samalla kun tekoäly tuo mukanaan mahdollisuuksia, se asettaa myös uusia haasteita suunnittelijoille. On tärkeää

varmistaa, että tekoälytoiminnot ovat läpinäkyviä ja helposti ymmärrettäviä käyttäjille. (Farrell.)

Service Desk -kokemukseni perusteella tiedän, että monille käyttäjille uudet ja monimutkaiset teknologiat voivat tuntua pelottavilta tai vaikeasti omaksuttavilta. Siksi tekoälyn ja koneoppimisen integroinnissa on tärkeää säilyttää selkeys ja käyttäjäystävällisyys, jotta käyttäjät voivat tuntea hallitsevansa järjestelmää.

Lisätty todellisuus ja virtuaalitodellisuus ovat seuraava merkittävä kehitysaskel käyttäjäkokemuksessa. Näiden teknologioiden avulla käyttäjät voivat kokea digitaaliset ympäristöt aivan uudella tavalla, mikä avaa valtavia mahdollisuuksia esimerkiksi opetukseen, viihteeseen ja terveydenhuoltoon. Käyttäjäkeskeisen suunnittelun rooli näissä teknologioissa on ratkaiseva, sillä AR- ja VR-sovellukset vaativat intuitiivisia ja helppokäyttöisiä käyttöliittymiä, jotta käyttäjät voivat navigoida monimutkaisissa virtuaaliympäristöissä. Lisätyn todellisuuden avulla voidaan esimerkiksi tarjota käyttäjille visuaalisia ohjeita ja opastusta todellisissa ympäristöissä, mikä voi vähentää tarvetta perinteiseen asiakastukeen. (Farrell.)

Service Deskissä olen usein nähnyt, kuinka selkeät visuaaliset ohjeet voivat auttaa käyttäjiä ratkaisemaan ongelmia itsenäisesti. AR-teknologia tarjoaa mahdollisuuden viedä tämä uudelle tasolle, jossa käyttäjät voivat saada reaaliaikaista visuaalista opastusta laitteidensa käyttöön.

Käyttäjäkeskeinen suunnittelu ja iteratiivinen kehitys tulevat olemaan yhtä tärkeitä uusien teknologioiden, kuten tekoälyn ja lisätyn todellisuuden, kehityksessä kuin ne olivat perinteisissä ohjelmistoissa. Käyttäjien tarpeet muuttuvat jatkuvasti, ja uusien teknologioiden käyttöönotossa iteratiivinen suunnittelu mahdollistaa jatkuvan optimoinnin ja parannusten tekemisen käyttäjäpalautteen perusteella. Tämä prosessi auttaa vähentämään käyttäjien kokemaa oppimiskynnystä ja parantamaan heidän kokemustaan uusien teknologioiden kanssa. (Gibbons.)

Service Desk -työssäni olen nähnyt, kuinka tärkeää on testata ja parantaa jatkuvasti uusia järjestelmiä, kun ne otetaan käyttöön. Samalla tavalla tekoälyn ja AR/VR-tekniikoiden käyttöönotto vaatii jatkuvaa käyttäjäpalautteen keräämistä ja kehitystyötä, jotta teknologiat saadaan toimimaan mahdollisimman tehokkaasti ja käyttäjäystävällisesti.

Tulevaisuuden käyttäjäkeskeinen suunnittelu ei koske vain teknisiä ratkaisuja, vaan myös eettisiä kysymyksiä. Kun tekoäly ja koneoppiminen keräävät ja analysoivat valtavia määriä käyttäjätietoa, on suunnittelijoiden otettava huomioon yksityisyyden suoja ja tiedonhallinnan läpinäkyvyys.

Käyttäjäkeskeisessä suunnittelussa on varmistettava, että käyttäjät ymmärtävät, miten heidän tietojensa käytetään, ja että heillä on valta hallita omaa digitaalista jalanjälkeään. Tämä on erityisen tärkeää teknologioiden, kuten tekoälyn ja koneoppimisen, kohdalla, joissa käyttäjäkokemus perustuu pitkälti käyttäjädatan analysointiin ja automaattisiin päätöksiin. Omassa työssäni näen usein käyttäjien huolenaiheita, jotka liittyvät tietoturvaan ja yksityisyyteen. Näitä huolia voidaan lieventää selkeällä viestinnällä ja läpinäkyvillä käytännöillä, jotka antavat käyttäjille hallinnan omasta datastaan. (Nielsen.)

Käyttäjäkeskeisen suunnittelun tulevaisuuden näkymät ovat lupaavat, mutta samalla ne asettavat uusia haasteita suunnittelijoille. Uusien teknologioiden, kuten tekoälyn, lisätyn todellisuuden ja virtuaalitodellisuuden, kehittyessä on tärkeää säilyttää käyttäjäkeskeisen suunnittelun peruseräperiaatteet, jotka ovat käyttäjäystävällisyys, läpinäkyvyys ja jatkuva iteratiivinen kehitys. Service Desk -kokemukseni perusteella tiedän, että käyttäjien tarpeet ja odotukset kehittyvät jatkuvasti, ja näihin muutoksiin vastaaminen edellyttää joustavuutta ja kykyä sopeutua uusiin teknologioihin.

### 6.3 Suositukset suunnittelijoille ja ohjelmistokehittäjille

Käyttäjäkeskeisen suunnittelun periaatteiden soveltaminen ohjelmistokehityksessä on avain onnistuneiden ja käyttäjäystävällisten tuotteiden luomiseen. Käyttäjäpalautte, iteratiivinen suunnittelu ja selkeät käyttöliittymät ovat olennaisia elementtejä, jotka vaikuttavat suoraan

käyttäjäkokemukseen. Applen iPhoneen kehitysprosessista voidaan oppia monia tärkeitä asioita, jotka ovat sovellettavissa laajasti erilaisiin projekteihin.

Yksi keskeisistä suosituksista on, että käyttäjäpalautteen keräämistä ei tule nähdä yksittäisenä vaiheena, vaan jatkuvana prosessina koko kehityksen ajan. Applen iPhone-kehityksessä palaute ohjasi monia keskeisiä suunnittelupäätöksiä, kuten eleohjauksen optimoinnin ja ikonien sijoittelun. Käyttäjät olivat osallisina koko suunnitteluprosessissa, ja heidän palautteensa mahdollisti käyttöliittymän parantamisen ennen julkaisua. Suunnittelijoiden ja ohjelmistokehittäjien tulisi ottaa käyttöön käyttäjätestaukset varhaisessa vaiheessa ja integroida ne jatkuvaksi osaksi kehitystä. Tämä auttaa tunnistamaan ongelmat aikaisessa vaiheessa ja parantaa tuotteen käytettävyyttä lopullisessa versiossa. (Norman.)

Service Desk -kokemukseni kautta olen huomannut, kuinka tärkeää on kuunnella käyttäjien palautetta, jotta voidaan kehittää selkeämpiä ohjeita ja ratkaisuja heidän tarpeisiinsa. Tämä filosofia on suoraan sovellettavissa ohjelmistokehitykseen.

Toinen suositus on iteratiivisen suunnittelun omaksuminen. Apple käytti laajasti iteratiivista suunnitteluprosessia, jossa jokainen iPhoneen versio parani käyttäjäpalautteen perusteella. Tämä vähensi suurten virheiden ja muutosten tarvetta myöhäisessä vaiheessa ja mahdollisti nopeamman kehityksen. Suunnittelijat ja ohjelmistokehittäjät voivat hyötyä pienistä, säännöllisistä päivityksistä ja parannuksista. Iteratiivinen lähestymistapa mahdollistaa tuotteen jatkuvan kehityksen ja optimoinnin sen sijaan, että kaikki parannukset tehdään yhdellä suurella muutoksella. (Norman.)

Service Desk -kokemuksissani olen nähnyt, kuinka pienet muutokset ohjeistuksiin voivat helpottaa käyttäjien ongelmia huomattavasti, mikä heijastuu parempana käyttäjätyytyväisyytenä.

Käyttäjäkeskeinen ajattelu on tärkeä suunnitteluperiaate, joka tulee säilyttää kaikissa projekteissa. Suunnittelijoiden ja kehittäjien tulisi kysyä jatkuvasti,

miten käyttäjät tulevat vuorovaikuttamaan ohjelmiston kanssa ja mitkä ovat heidän keskeiset tarpeensa ja toiveensa. Tämä voidaan saavuttaa käyttäjätestauksilla ja prototypoinnilla, kuten Apple teki iPhoneen kehityksen aikana. (Norman.)

Omassa työssäni olen havainnut, että kun käyttäjien tarpeet ja odotukset otetaan huomioon, lopputulos on aina parempi. Esimerkiksi selkeät ja helposti seurattavat ohjeet voivat vähentää virheiden määrää ja parantaa käyttäjäkokemusta. Suunnittelijoiden tulisi muistaa, että käyttäjät eivät aina ole teknisesti taitavia, joten käyttöliittymien ja prosessien tulee olla intuitiivisia ja selkeitä. Viestintä on keskeinen tekijä onnistuneessa ohjelmistokehityksessä. Applen kehitysprosessissa tiivis yhteistyö suunnittelijoiden, kehittäjien ja käyttäjien välillä mahdollisti innovatiivisten ratkaisujen toteuttamisen. Suunnittelijoiden ja kehittäjien tulisi tehdä tiivistä yhteistyötä läpi koko projektin ja olla avoimia kommunikoimaan käyttäjäpalautteen perusteella tehtävistä muutoksista. (Made In Tandem.)

Service Desk -työssäni olen usein joutunut toimimaan viestinvälittäjänä käyttäjien ja teknisen tiimin välillä, ja olen nähnyt, kuinka tärkeää on varmistaa, että kaikki osapuolet ymmärtävät toistensa tarpeet ja odotukset. Vahva viestintä ja monialainen yhteistyö ovat avainasemassa ohjelmistokehityksessä, ja ne varmistavat, että tuote täyttää käyttäjien tarpeet ja liiketoiminnan tavoitteet.

Käyttäjäkeskeisen suunnittelun periaatteet, kuten jatkuva käyttäjäpalautte, iteratiivinen suunnittelu, käyttäjän asettaminen suunnittelun keskiöön sekä vahva viestintä, ovat keskeisiä elementtejä menestyksellisen ohjelmistotuotteen kehittämisessä. Omassa työssäni näiden periaatteiden soveltaminen on johtanut parempaan käyttäjätyytyväisyyteen ja tehokkaampaan ongelmanratkaisuun. Suunnittelijoiden ja ohjelmistokehittäjien tulisi hyödyntää näitä oppeja kaikissa projekteissaan, jotta he voivat tuottaa tuotteita, jotka ovat sekä teknisesti edistyneitä että käyttäjäystävällisiä.

## 7 Yhteenveto

### 7.1 Tutkimuksen keskeiset johtopäätökset

Tässä tutkimuksessa tarkasteltiin käyttäjäkeskeisen suunnittelun merkitystä ohjelmistokehityksessä ja sen vaikutusta käyttäjäkokemuksen parantamiseen.

Tutkimus vahvistaa, että käyttäjäkeskeinen suunnittelu on olennainen osa ohjelmistokehitystä, joka parantaa ohjelmiston käytettävyyttä ja loppukäyttäjien tyytyväisyyttä. Iteratiivinen suunnittelu, jossa käyttäjiä osallistetaan kehitysprosessin eri vaiheissa, auttaa kehittämään tuotteita, jotka vastaavat paremmin käyttäjien todellisiin tarpeisiin. Tämä johtopäätös ilmeni selvästi Applen iPhone:n kehitysprosessista, jossa käyttäjättestaukset ja jatkuva palaute auttoivat optimoimaan käyttöliittymän intuitiiviseksi ja helppokäyttöiseksi.

Iteratiivinen suunnitteluprosessi, jossa tuotetta kehitetään vaiheittain käyttäjäpalautteen perusteella, osoittautui tehokkaaksi tavaksi vähentää käyttöön liittyviä virheitä ja parantaa käyttäjäkokemusta. Tämä iteratiivinen lähestymistapa mahdollisti paremman käytettävyyden ja vähensi tarvetta suurille muutoksille myöhäisessä kehitysvaiheessa. Myös omassa työssäni Service Deskissä olen huomannut, että jatkuva prosessien parantaminen käyttäjäpalautteen perusteella johtaa selvästi parempaan käyttäjäkokemukseen.

Applen päätös kehittää App Store -sovelluskauppa osoitti, että ekosysteemin rakentaminen voi lisätä merkittävästi tuotteen arvoa ja sitouttaa käyttäjiä. Tämä ekosysteemiajattelu on laajasti sovellettavissa ohjelmistokehitykseen tarjoamalla käyttäjille mahdollisuuksia personoida ja laajentaa käyttökokemustaan voidaan lisätä käyttäjätyytyväisyyttä ja käyttökollisuutta.

Tutkimus korostaa käyttäjien osallistamisen tärkeyttä suunnitteluprosessin kaikissa vaiheissa. Käyttäjien tarpeiden ja odotusten ymmärtäminen auttaa suunnittelutiimiä tekemään ratkaisuja, jotka eivät ainoastaan täytä teknisiä vaatimuksia, vaan myös tarjoavat miellyttävän ja intuitiivisen käyttökokemuksen.

Tämä on tärkeää erityisesti sellaisten tuotteiden ja palveluiden kehittämisessä, jotka ovat monimutkaisia ja joiden käyttäjillä voi olla erilaisia taitotasoja.

Hyvä käytettävyys vähentää merkittävästi tukipalveluiden tarvetta ja virheiden määrää. Tutkimuksessa tuli ilmi, että selkeät käyttöliittymät, joissa affordanssit ja signaalit tukevat toisiaan, helpottavat käyttäjän vuorovaikutusta ohjelmiston kanssa ja vähentävät virheiden riskiä. Käyttäjäystävällinen suunnittelu ei ainoastaan lisää käyttäjätyytyväisyyttä, vaan myös vähentää organisaation resursseja asiakastuen tarjoamisessa.

Nämä johtopäätökset osoittavat, että käyttäjäkeskeinen suunnittelu on tärkeä strategia, jolla voidaan parantaa ohjelmistojen käytettävyttä ja lisätä asiakastyytyväisyyttä. Sen soveltaminen käytännössä vaatii käyttäjien osallistamista, iteratiivista kehitystä ja jatkuvaa palautteen analysointia, mikä takaa, että tuote tai palvelu vastaa käyttäjien todellisia tarpeita ja odotuksia.

## 7.2 Tutkimuksen rajoitukset ja jatkotutkimusaiheet

Tutkimuksessa käytetty aineisto on rajallinen ja perustuu pitkälti omiin kokemuksiini Technical Support Specialistina Service Deskissä sekä kirjallisuudesta, kuten Donald A. Normanin ja Nielsen Norman Groupin julkaisuista, saatuun tietoon. Näin ollen tutkimuksen tulokset saattavat olla jossain määrin sidottuja tämän kontekstin erityispiirteisiin, eikä niiden yleistettävyys välttämättä kata kaikkia käyttäjäkeskeisen suunnittelun sovellusalueita tai ohjelmistokehityksen ympäristöjä. Erityisesti käyttäjätutkimuksen näkökulmasta tämä tarkoittaa, että tutkimuksen havainnot ei voida täysin yleistää laajemmalle käyttäjäryhmälle ilman lisätutkimusta.

Tutkimus nojautuu pitkälti laadulliseen eli kvalitatiiviseen aineistoon, kuten henkilökohtaisiin havaintoihin ja kokemuksiin, mikä voi asettaa haasteita objektiivisuuden ja tulosten yleistettävyyden suhteen. Vaikka laadullinen tutkimus tarjoaa syvällistä ymmärrystä käyttäjäkokemuksista, se ei mahdollista määrällistä arviointia tai selkeitä tilastollisia johtopäätöksiä. Kvantitatiivisen

aineiston puute rajoittaa tutkimuksen kykyä esittää yleispäteviä, numeerisesti vahvistettuja tuloksia käyttäjäkeskeisen suunnittelun vaikutuksista.

Tämän tutkimuksen ajallinen rajaus voi vaikuttaa sen tulosten kestävyYTEEN. Käyttäjäkeskeinen suunnittelu on ala, joka kehittyy jatkuvasti teknologian ja käyttäjien tarpeiden muuttuessa. Tämä tarkoittaa, että nyt tehdyt havainnot ja suositukset voivat vanhentua suhteellisen nopeasti, kun uudet teknologiat, kuten tekoäly, lisätty todellisuus ja virtuaalitodellisuus, alkavat vaikuttaa käyttäjäkokemuksen suunnitteluun yhä enemmän.

Tutkimus nojaa vahvasti yksittäisiin esimerkkeihin, kuten Applen iPhone'n kehitysprosessiin, jotka toimivat konkreettisina esimerkkeinä käyttäjäkeskeisen suunnittelun merkityksestä. Tämä lähestymistapa on hyödyllinen havainnollistamaan tiettyjä periaatteita, mutta samalla se rajoittaa tutkimuksen näkökulmaa. Yksittäisten tapausten analysointi ei välttämättä anna kattavaa kuvaa siitä, miten käyttäjäkeskeinen suunnittelu soveltuu muihin organisaatioihin ja konteksteihin.

Jatkotutkimus voisi hyötyä kvantitatiivisten menetelmien soveltamisesta käyttäjäkeskeisen suunnittelun vaikutusten arviointiin. Esimerkiksi käyttäjätyytyväisyyden mittaaminen tilastollisin menetelmin, kyselytutkimusten ja käytettävyydestien avulla, voisi tarjota määrällistä tukea tämän työn laadullisille havainnoille ja auttaa vahvistamaan, missä määrin UCD-parannukset vaikuttavat käyttäjäkokemukseen ja ohjelmiston menestykseen.

Tämän tutkimuksen havainnot perustuvat pääasiassa yhteen työympäristöön ja tiettyihin esimerkkitapauksiin. Jatkotutkimuksessa olisi hyödyllistä laajentaa käyttäjäryhmää eri ikäisiin, kulttuurisiin ja teknologisiin ympäristöihin, jotta saataisiin kattavampi kuva käyttäjäkeskeisen suunnittelun vaikutuksista. Erityisesti olisi mielenkiintoista tutkia, miten eri demografiset tekijät vaikuttavat käyttäjien odotuksiin ja käyttäjäkokemukseen.

Uudet teknologiat, kuten tekoäly, lisätty todellisuus ja virtuaalitodellisuus, ovat nostamassa käyttäjäkokemuksen suunnittelun vaatimuksia uudelle tasolle.

Jatkotutkimus voisi keskittyä siihen, miten käyttäjakeskeisen suunnittelun periaatteita voidaan soveltaa näihin teknologioihin ja mitä uusia haasteita ja mahdollisuuksia ne tuovat mukanaan. Esimerkiksi tekoäly voi tuoda mukanaan henkilökohtaisempia kokemuksia, mutta samalla se asettaa uusia vaatimuksia läpinäkyvyydelle ja eettisyydelle.

Tutkimus voisi myös keskittyä tarkastelemaan käyttäjakeskeisen suunnittelun taloudellisia vaikutuksia ohjelmistokehitykseen ja liiketoimintaan. Esimerkiksi kvantitatiivinen tutkimus siitä, kuinka paljon käyttäjakeskeinen suunnittelu vähentää tukipyyntöjen määrää tai parantaa käyttäjätyytyväisyyttä ja asiakasuskollisuutta, voisi tarjota tärkeää tietoa yrityksille, jotka harkitsevat käyttäjakeskeisen suunnittelun implementointia.

Käyttäjakeskeinen suunnittelu perustuu käyttäjien tarpeiden ja odotusten ymmärtämiseen, mutta nämä tarpeet voivat vaihdella merkittävästi kulttuurien välillä. Jatkotutkimus voisi tutkia, kuinka käyttäjakeskeisen suunnittelun periaatteet tulisi mukauttaa eri kulttuurien ja markkinoiden erityispiirteisiin. Tällainen tutkimus voisi auttaa kansainvälisiä yrityksiä kehittämään tuotteita, jotka ovat paremmin kohdennettuja eri käyttäjäryhmille.

Yhteenvetona voidaan todeta, että tutkimuksella on useita rajoituksia, jotka liittyvät aineiston laajuuteen, kvalitatiiviseen näkökulmaan ja esimerkkitapausten rajallisuuteen. Jatkotutkimusaiheita on runsaasti, ja ne tarjoavat mahdollisuuden syventää ymmärrystä käyttäjakeskeisen suunnittelun vaikutuksista eri konteksteissa sekä laajentaa näkökulmaa uusien teknologioiden ja eri kulttuurien vaikutukseen käyttäjäkokemukseen. Näiden aiheiden tutkiminen voisi vahvistaa ja täydentää tämän työn johtopäätöksiä ja tarjota entistä kattavampaa tietoa käyttäjakeskeisen suunnittelun merkityksestä ohjelmistokehityksessä ja liiketoiminnassa.

### 7.3 Loppupäätelmät

Tutkimuksessa tarkasteltiin käyttäjakeskeisen suunnittelun merkitystä ohjelmistokehityksessä ja sen vaikutusta ohjelmistojen käytettävyyteen ja

käyttäjäkokemukseen. Käyttäjäkeskeinen suunnittelu on noussut keskeiseksi suunnitteluperiaatteeksi nykyisessä teknologia-alan kehityksessä, jossa käyttäjäkokemus on tullut tärkeäksi kilpailutekijäksi. Tämä tutkimus on tuonut esiin useita keskeisiä havaintoja ja suosituksia, jotka voivat auttaa organisaatioita kehittämään käyttäjäystävällisiä ja käyttäjien tarpeita vastaavia ohjelmistotuotteita.

Tutkimuksen perusteella voidaan todeta, että käyttäjäkeskeinen suunnittelu on tehokas lähestymistapa ohjelmistojen käytettävyyden parantamiseksi ja käyttäjätyytyväisyyden lisäämiseksi. Käyttäjäkeskeinen suunnittelu eroaa perinteisistä suunnittelumalleista siinä, että se asettaa käyttäjän suunnitteluprosessin keskiöön ja painottaa käyttäjien aktiivista osallistumista kehityksen eri vaiheissa. Tämä tutkimuksessa korostunut iteratiivinen lähestymistapa, jossa tuotetta kehitetään jatkuvasti käyttäjäpalautteen perusteella, mahdollistaa paremman käytettävyyden, käyttäjäkokemuksen ja siten myös ohjelmistotuotteen menestyksen markkinoilla.

Erytisesti iteratiivisen suunnittelun ja käyttäjätestauksen merkitys korostui tutkimuksen havainnoissa. Apple iPhonen kehitysprosessin analyysi toi esille, kuinka jatkuva käyttäjätestaus ja käyttäjäpalautteen hyödyntäminen voivat johtaa käyttäjäystävällisiin ja innovatiivisiin ratkaisuihin. Applen päätös luottaa iteratiiviseen suunnitteluun ja käyttäjäkeskeiseen lähestymistapaan osoitti, että käyttäjäkokemuksen parantaminen on mahdollista, kun käyttäjät ovat aktiivisesti mukana suunnitteluprosessissa. Tämä periaate voidaan yleistää laajemminkin ohjelmistokehitykseen, ja sen soveltaminen voi parantaa tuotteiden käytettävyyttä ja vähentää virheiden määrää.

Käytettävyydestä, käyttäjäpalaute ja iteratiivinen kehitys eivät ainoastaan paranna käyttäjäkokemusta, vaan myös vähentävät asiakastuen tarvetta ja parantavat asiakasuskollisuutta. Omassa työssäni Service Deskissä on nähtävissä, kuinka käyttöliittymän selkeys ja käyttäjien tarpeisiin vastaaminen vähentävät toistuvia tukipyyntöjä ja auttavat käyttäjiä ratkaisemaan ongelmia itsenäisesti. Tämän tutkimuksen perusteella voidaan siis todeta, että käyttäjäkeskeisen suunnittelun periaatteet ovat olennaisia myös käytännön

teknisessä tuessa, jossa käyttäjäkokemuksen parantaminen johtaa parempaan palvelun laatuun ja käyttäjien tyytyväisyyteen.

Tutkimus toi esiin myös käyttäjäkeskeisen suunnittelun rajat ja haasteet. Käyttäjäpalautteen ja käytettävyydestauksen merkitys on kiistaton, mutta on tärkeää huomata, että käyttäjäkeskeisen suunnittelun toteuttaminen vaatii resursseja ja sitoutumista koko organisaatiolta. Käyttäjien tarpeiden jatkuva huomioiminen ja iteratiivinen kehittäminen voivat olla aikaa vieviä prosesseja, ja ne vaativat tiivistä yhteistyötä suunnittelijoiden, kehittäjien ja muiden sidosryhmien välillä. Lisäksi käyttäjäkeskeisen suunnittelun soveltaminen uusissa teknologioissa, kuten tekoälyssä ja lisätyssä todellisuudessa, tuo mukanaan uusia haasteita ja vaatii uusia lähestymistapoja, jotta käyttäjäkokemus pysyy selkeänä ja intuitiivisena.

Jatkotutkimusaiheiden osalta tutkimus toi esiin tarpeen tarkastella käyttäjäkeskeisen suunnittelun vaikutuksia laajemmalla kvantitatiivisella aineistolla sekä tutkia sen soveltamista erilaisissa kulttuurillisissa ja teknologisissa konteksteissa. Esimerkiksi uusien teknologioiden, kuten tekoälyn ja AR:n tuomat haasteet ja mahdollisuudet käyttäjäkokemuksen parantamiseen vaativat lisää tutkimusta. Myös käyttäjäkeskeisen suunnittelun taloudellisten vaikutusten tarkempi tutkiminen voisi auttaa organisaatioita ymmärtämään käyttäjäkeskeisen suunnittelun potentiaalin liiketoiminnan kannalta.

Lopuksi voidaan todeta, että käyttäjäkeskeinen suunnittelu on olennainen osa nykyaikaista ohjelmistokehitystä, ja sen avulla voidaan parantaa tuotteiden käytettävyyttä, vähentää tukipalvelun tarvetta ja lisätä asiakastyytyväisyyttä. Tutkimus osoittaa, että käyttäjäkeskeisen suunnittelun periaatteiden, kuten iteratiivisen suunnittelun ja käyttäjätestauksen, soveltaminen ohjelmistokehityksen eri vaiheissa voi johtaa merkittäviin parannuksiin niin teknisesti kuin käyttäjäkokemuksen näkökulmasta. Organisaatioiden tulisi ottaa nämä periaatteet huomioon kehittäessään uusia tuotteita ja palveluita, ja panostaa resursseja käyttäjäkokemuksen optimointiin, jotta ne voivat saavuttaa kilpailuetua ja luoda käyttäjäystävällisiä ratkaisuja, jotka vastaavat markkinoiden ja käyttäjien muuttuvia tarpeita.



## Lähteet

- 1 Norman, Donald A. 2013. The Design of Everyday Things. Pokkari.
- 2 Gibbons, Sarah. 2016. Design Thinking 101. Verkkoaineisto. NNGroup. Luettu 17.07.2024. <<https://www.nngroup.com/articles/design-thinking/>>
- 3 Nielsen, Jakob. 2012. Usability 101: Introduction to Usability. Verkkoaineisto. NNGroup. Luettu 18.07.2024. <<https://www.nngroup.com/articles/usability-101-introduction-to-usability/>>
- 4 Sheppard, Benedict, Sarrazin, Hugo, Kouyoumijan, Garen ja Dore, Fabricio. 2018. The business value of design. Verkkoaineisto. McKinsey & Company. Luettu 25.9.2024. <<https://www.mckinsey.com/capabilities/mckinsey-design/our-insights/the-business-value-of-design/>>
- 5 Nielsen, Jakob ja Laubheimer, Page. 2019. Top 10 Application-Design Mistakes. Verkkoaineisto. NNGroup. Luettu 17.07.2024. <<https://www.nngroup.com/articles/top-10-application-design-mistakes/>>
- 6 Made In Tandem. 2022. 14 Years: Here's How the iPhone Influenced UI Design. Made In Tandem Blog. Luettu 19.08.2024. <<https://madeintandem.com/blog/14-years-heres-iphone-influenced-ui-design/>>
- 7 Tibidavid. The History of iOS: A UI/UX Designer's Perspective. Verkkoaineisto. Luettu 19.08.2024. <<https://www.tibidavid.com/ux-design/the-history-of-ios-a-ui-ux-designer-s-perspective/>>
- 8 Moran, Kate. 2019. Usability Testing 101. Verkkoaineisto. NNGroup. Luettu 20.07.2024. <<https://www.nngroup.com/articles/usability-testing-101/>>
- 9 Wong, Lauren. 2018. iOS: The User Interface Design Journey. Verkkoaineisto. Sush Labs. Luettu 19.08.2024. <<https://www.sushlabs.com/blog/ios-the-ui-design-journey/>>
- 10 Bootcamp. The Apple Effect: How a Company Revolutionized UI/UX Design and What You Can Learn From Them. Verkkoaineisto. Luettu 22.08.2024. <<https://bootcamp.uxdesign-and-what-you-can-learn-from-them-cb2c41b26931/>>

- 11 Farrell, Susan. 2017. UX Research Cheat Sheet. Verkkoaineisto. NNGroup. Luettu 26.07.2024. <<https://www.nngroup.com/articles/ux-research-cheat-sheet>>