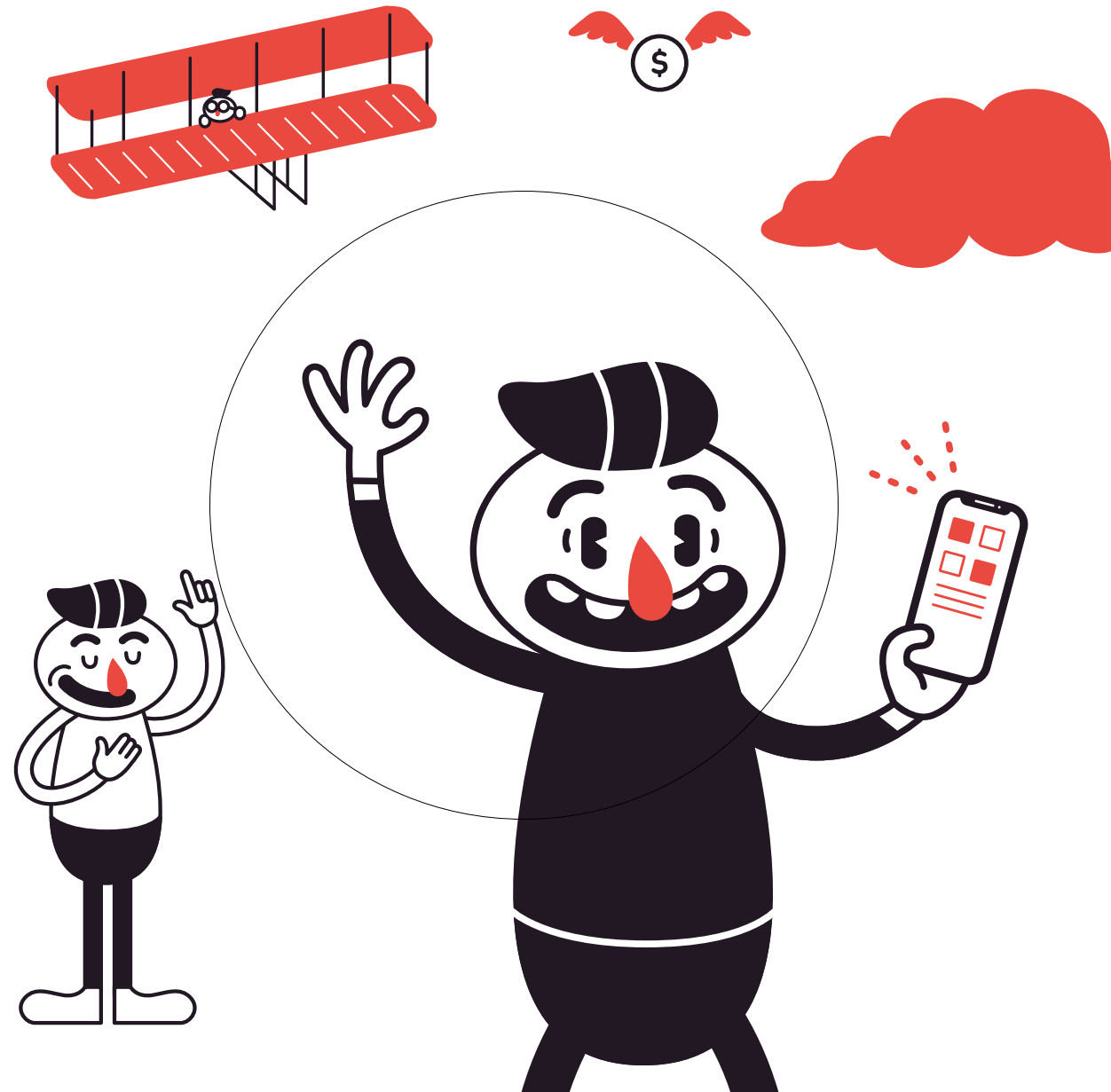


**SOFTAPALJASTUKSIA**





## **KONSEPTI JA SUUNNITTELU**

Kisko Labs Oy  
[www.kiskolabs.com](http://www.kiskolabs.com)

## **KUVITUKSET**

Nick Hoong  
Cartoon Dumpling

ISBN 978-952-94-1477-2  
ISBN 978-952-94-1478-9 (PDF)

## **SOFTAPALJASTUKSIA**

44 ohjetta onnistuneeseen  
softaprojektiin

**Esipuhe**

### **Yksin - tarkoittaa minkään arvokkaan toteutumista harvoin.**

Yrityksen arvoa kasvattavat ja siis asiakkaalle arvoa tuottavat toimenpiteet tehdään lähes aina yhdessä. On todettu, että jatkossa kaikki yritykset ovat softayrityksiä. Palveluihin liittyy usein softaa, samoin tuotteisiin. Harva ajattelee sykemittaria vain fyysisenä tuotteena, vaan tuotteen ja softan yhdistelmänä, käyttökokemuksena. Lukuunottamatta yhdysvaltalaisia alustatalouden jättejä, harvalla yrityksellä tänä päivänä on kaikki tarvittava osaaminen toimivien palvelujen kehittämiseen.

Kaikilla toimialoilla on mahdollista löytää arvonlisää digitalisaatiosta. Asiakkaalle usein arvo tiivistyy palvelun käytettävyytenä ja nopeutena. Siksi nopeuden ja ketteryyden hyödyllisyys kehityksessä korostuu. On syytä tutkia ja kokeilla mitä olemassa oleva data kertoo. Samalla voin helpottaa tuskaa: Data ei koskaan ole ”riittävän hyvää”, aina on parannettavaa. Siksi palvelua kannattaa lähteä kehittämään.

Mitä palvelua kehittäviin kumppaneihin tulee, Etolan perustajan Nils Etolan vinkki pätee edelleen:

”Kun tarvitset apua, ota paras sillä se tulee edullisimmaksi.”

Markkinoilla ei ole tapana odottaa. Kun idea (jonka arvo ensimmäisen luvun mukaan on nolla...) on syntynyt, miten siitä saadaan nopeasti aikaan toimiva palvelu, jota asiakkaat haluavat käyttää? Jopa verottaja on muokannut palveluitaan aidosti käytettäväksi ja heillä sentään on monopoli!

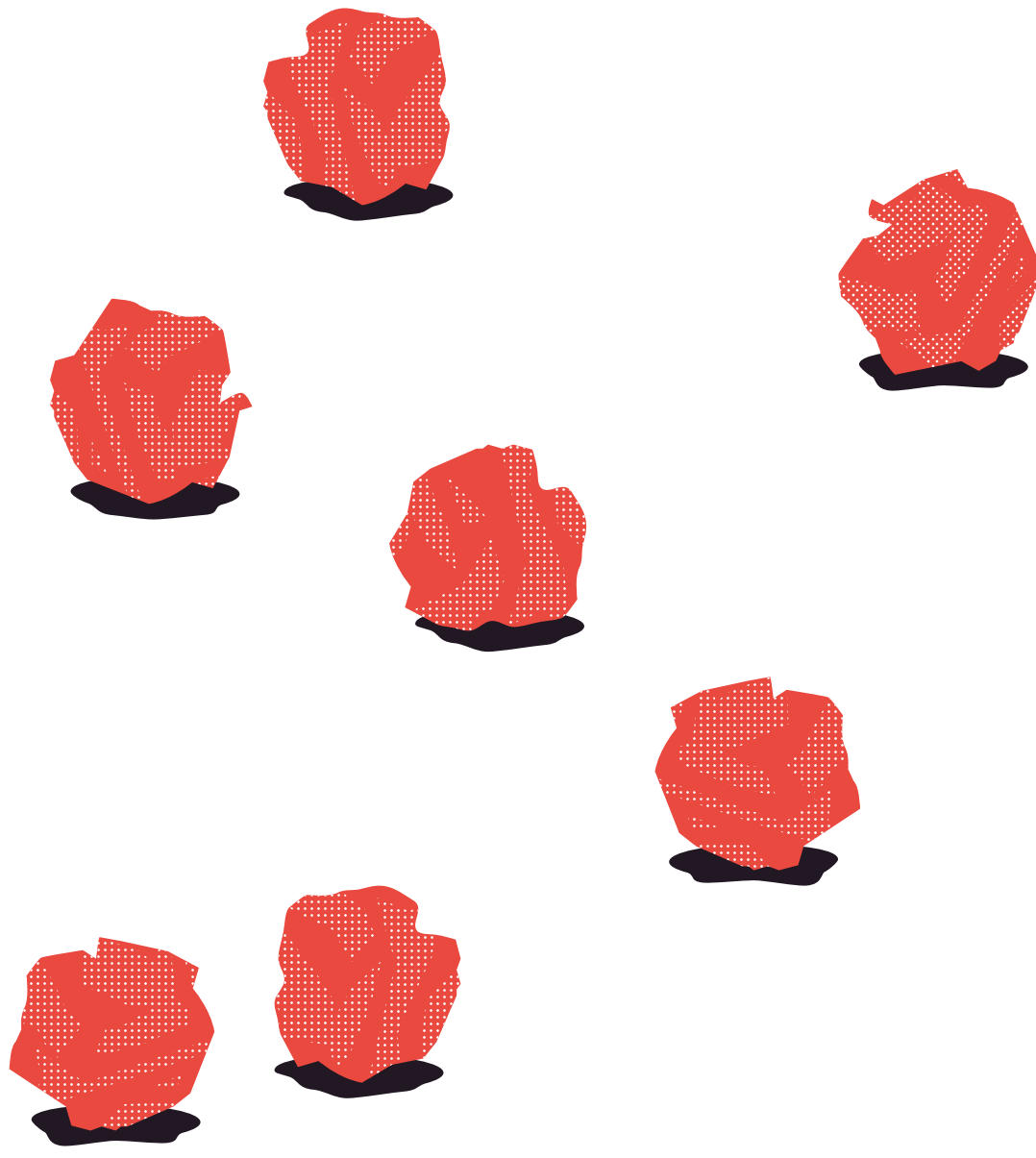
Kehotan avoimuuteen. Olen saanut tehdä Kiskon kanssa töitä eri rooleissa vuosien ajan ja olen nähnyt, että he kertovat jos tarpeellista osaamista ei juuri nyt ole saatavilla. Ensimmäinen 10 vuotta olivat valmistautumista tähän hetkeen. Pitkää peliä, jossa asiakkaan asiakas on aina etusijalla. Hauskaa matkaa!

**— Antti Merilehto**

*Tekoäly - matkaopas johtajalle kirjan kirjoittaja*



Idea



## Idean arvo on nolla

Jos sinulla on käsittämättömän huono idea ja toteutat sen äärettömän hyvin, sinulla saattaa olla käsissäsi mainio tuote. Vastaavasti jos sinulla on hyvä idea, jota et toteuta tai toteutat sen huonosti, sinulla ei ole oikeastaan mitään. Keskity siis toteutukseen sen sijaan, että vaalit ja hinkkaat ideaasi yksin pimeässä.





## Yksinkertaiset ideat eivät ole yksinkertaisia

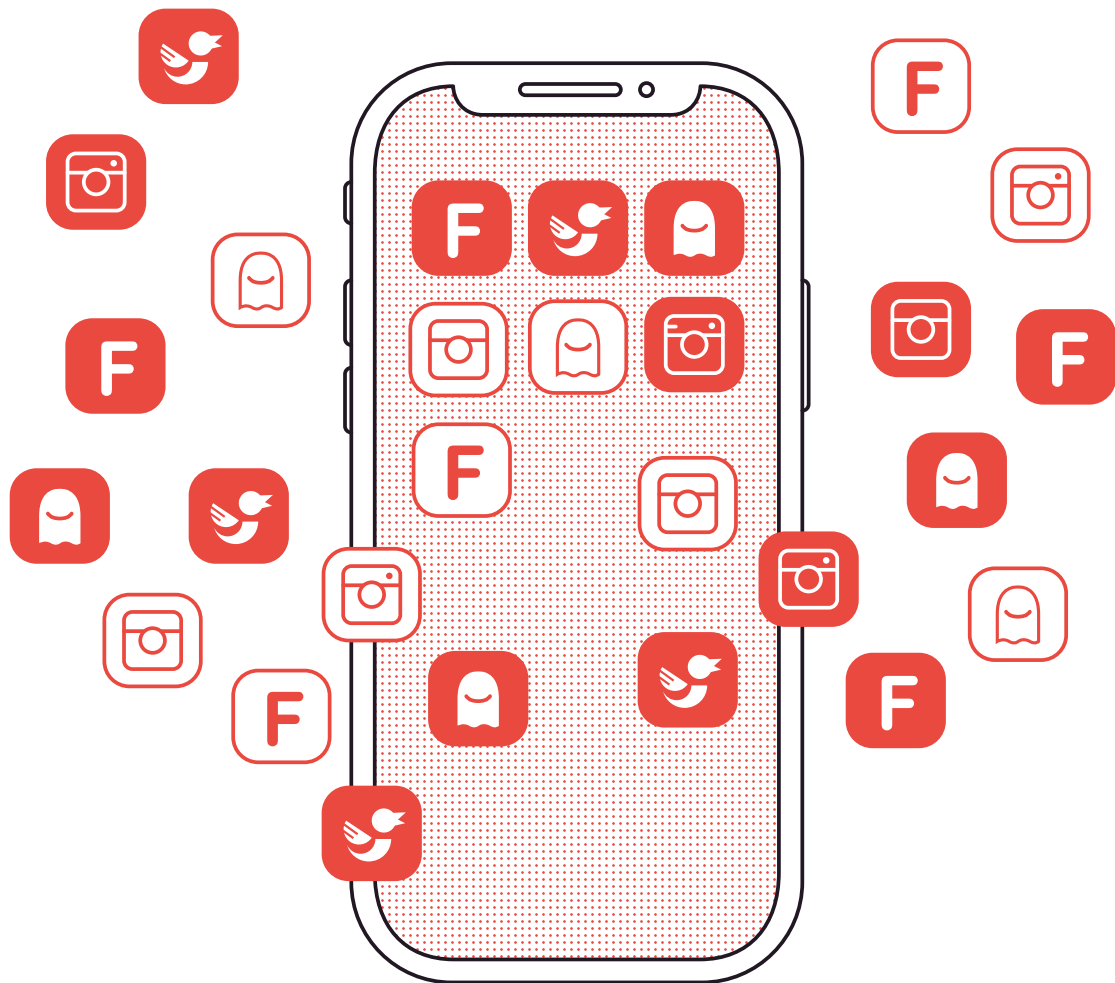
Ensimmäinen syy tälle on se, että usein loppukäyttäjä saattaa haluta jotain muuta tai enemmän kuin alunperin on ajateltu. Toinen syy on se, että harvalla on todellisuudessa käsitystä sovellusten teknisestä toteutuksesta. Esimerkiksi suljetulle pienelle käyttäjäryhmälle voi olla suhteellisen yksinkertaista rakentaa sovellus, mutta mikäli tarkoitus on tehdä sovellus isommalle käyttäjäryhmälle, on otettava huomioon monia asioita, joita harva on osannut miettiä. Tällöin ollaan jo monimutkaisessa maailmassa, jolloin paperilla yksinkertaiselta kuulostava idea ei olekaan enää niin yksinkertainen.





## Keskustele loppukäyttäjän kanssa

Ihmisten mielet toimivat eri tavoin. Esimerkiksi insinööri keksii ongelmaan kuin ongelmaan ratkaisun (johon usein liittyy nippusiteet), mutta ovatko nuo ratkaisut oikeasti hyviä? Parhaat ideat löytyvät niin, että loppukäyttäjän kanssa keskustellaan. Annetaan testiversioita kokeiltavaksi, sparrataan ratkaistavan ongelman kanssa ja pyydetään rehellistä palautetta. Ajatellaan vaikka menestynyttä start-up yrittäjää - aika harva on onnistunut tekemään miljoonansa ilman loppukäyttäjän kanssa käytyjä keskusteluja palvelun kehitysvaiheessa. Hae palautetta aikaisessa vaiheessa ja uskalla näyttää keskeneräisiäkin suunnitelmia loppukäyttäjälle. Sitä kautta löytyy oikeasti hyvät ideat.



## Äppi-ideasi on toteutettavissa, mutta onko sille tarvetta?

Nykyään meillä kaikilla on puhelimet täynnä erilaisia sovelluksia, minkä myötä meistä jokaisesta on tullut appikeksijöitä jos jonkinmoisiin tarkoituksiin. Usein kysytään, onko tämä idea toteutettavissa? Tämän päivän teknologioilla suurin osa esitetyistä appi-ideoista on täysin toteutettavissa.

Tärkeämpi kysymys on se, onko ideoimallesi sovellukselle tarvetta? Toteutettavien ideoiden määrä on pohjaton ja paljon suurempi kuin ihmisten kipeitä tarpeita ratkaisevien ideoiden määrä. Vielä rajatumpi on ihmisten aktiivisesti käyttämien palveluiden joukko.

Eli kun lähdet kehittämään sovelluksia, aloita loppukäyttäjien tarpeista. Todellisten tarpeiden löytäminen on kuitenkin loppujen lopuksi niin paljon vaikeampaa.



## Kun softasi pitäisi tehdä sitä sun tätä

Siinä vaiheessa kun huomaat, että olet tilaamassa ohjelmistoa, jonka pitäisi tehdä vähän kaikenlaista ja kaiken pitäisi pystyä muuttumaan, pysähdy hetkeksi. Mieti läpi onko palveluusi määritetty vielä se, mitä oikeasti halutaan tehdä. Koodin kirjoittaminen on minkä tahansa digitaalisen palvelun rakentamista pienin askelin. Tämän vuoksi suunnitelman tulee olla kirkas ennen kuin riviäkään koodia kirjoitetaan.

Kysy itseltäsi, haluatko Excelin vai sinulle räätälöidyn työkalun? Jos haluat monitoimityökalun, joka tekee mitä vain, se todennäköisesti tekee kaikkea keskinkertaisesti, mutta ei mitään todella hyvin.





**Kumppanin  
valinta**



## Hyvä IT-myyjä osaa myös koodata

Melko harvalla IT-alan myyjällä on tarkempaa käsitystä siitä, mitä projekteissa tapahtuu käytännön tasolla tai mitä ohjelmoija tai suunnittelija päivittäisessä työssään tekee. Heillä ei välttämättä ole ymmärrystä siitä, mitä jokin muutos tarkoittaa ohjelmoijan tai suunnittelijan työpöydällä. Hyöty siitä, että ostat suoraan osaaajilta on siinä, että näin pystytään antamaan tarjous tai arvio toiveistasi saman tien. Palautteen sykli on näin monta kertaa nopeampi.



## Varo asiakasta pilkkaavaa myyjää

Oletko saanut vierailuja myyjiltä, jotka ovat lytänneet aiempia teknologia-hankintoja, -järjestelmiä ja -valintoja? Näin tehdessään he käytännössä haukkuvat sinua – ostajaa. Vältä tällaisia toimittajia.

Toimittajalta vaatii tietyn kypsyyden asteen ymmärtää, että projekteissa tapahtuu kaikenlaista. Softamyyjä ei voi koskaan tietää mikä asiakkaan aiemmissa hankinnoissa on ollut onnistumista ja mikä epäonnistumista. Meidän tehtävä on auttaa asiakas nykytilasta eteenpäin, ei katsella peräpeiliin ja lytätä aiempi tekeminen.



## Vaadi toimittajaa selvittämään ohjelmistoprojektisi todellinen luonne

Suurin syy budjettien paukkumiseen ja aikataulujen venymiseen ohjelmistoprojekteissa on se, että toimittaja ei ole ymmärtänyt asiakkaan haluamaa laajuutta tuotteelle. Voi olla, että tarjouspyynnössä on kuvattu asiakkaan tarpeet hyvin ylimalkaisesti tai maailmassa on tapahtunut muutoksia, jotka ovat muuttaneet asiakkaan tarpeita. Voi myös olla, että kilpailija on tullut markkinoille tai uudet teknologiat ovat asettaneet uusia vaatimuksia palvelulle.

Resepti budjetti- ja aikatauluriskin poistamiseksi on kohtuullisen yksinkertainen: pilkotaan projekti pienempiin osiin. Mallimme on yleensä, että ensin pureskellaan asiakkaan kanssa idea ja sen liiketoimintamahdollisuudet läpi. Tämän jälkeen esitämme tiukasti rajatut ensimmäisen version toiminnallisuudet ja voimme antaa tälle selkeän aikataulun. Kun ensimmäinen versio on saatu maaliin ja ulos loppukäyttäjille, on aika tarkastella sen hetkisiä tarpeita ja tehdä priorisointia jatkokehitykselle. Pilkottaessa projekti konkreettiseen, lopputulokseen tähtäävään, pienempään kokonaisuuteen, on mahdollista reagoida ulkopuolelta tuleviin muutoksiin nopeammin kuin jättimäisessä pitkän aikavälin hankkeessa.

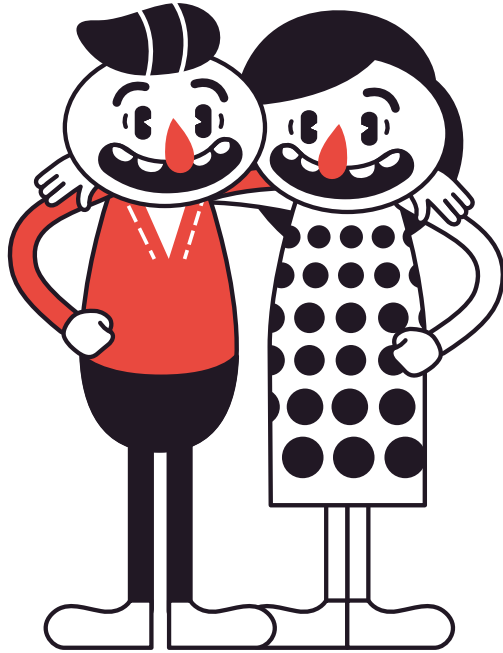


## Oletko osannut pohtia softahankettasi riittävän monelta kantilta?

Fysioterapeutti lähtee hoitamaan ongelmaa fysioterapialla, kirurgi veitsellä ja lääkäri lääkkeillä. Voi kuitenkin olla, että ongelman ratkaisuun tarvitaan kaikkia. Oletko koskaan huomannut, että kaikki, jopa sinä, lähtevät ratkaisemaan ongelmia oman osaamisensa kautta?

Olemme todenneet usean osaamisalueen yhdistämisen toimivan myös softaprojekteja toteutettaessa. Meillä tiimeihin voi kuulua kehittäjiä, suunnittelijoita ja markkinoinnin osaajia. Näin pystymme tarkastelemaan toteutettavaa tuotetta mahdollisimman laaja-alaisesti.





## Ihminen koodin takana

Olet varmaan tottunut kivijalkamyymälöissä ja verkossa asioidessasi hyvään palveluun. Ravintolaankaan mennessäsi et osta vain ravintoa vaan kokemuksen. Miksi et vaatisi tätä myös softaa ostaessasi?

Tulevaisuudessa softakehitys tulee keskittymään enemmän ja enemmän olemassa olevien palveluiden yhdistämiseen. Tekoäly muuttaa tavan tehdä myös softaa. Softan kehittäminen on jo nykyisellään 92%\* kommunikaatiota ja tulevaisuudessa vuorovaikutuksen merkitys korostuu entisestään. Tällöin viimeistään digikumppanisi tarjoama palvelukokemus nousee merkittäväksi kilpailutekijäksi.

Sillä ei ole ainoastaan väliä, mitä ostetaan teknisesti vaan myös kuka koodin toimittaa. Softakumppania valitessasi valitset myös tulevien kuukausien, tai kenties vuosien, työkaverin—kumppanin. Valitse niin, että kumppanisi tekee elämästäsi parempaa.

*\*kokemuksen tuoma arvio*



**Ominaisuuksien  
valinta**



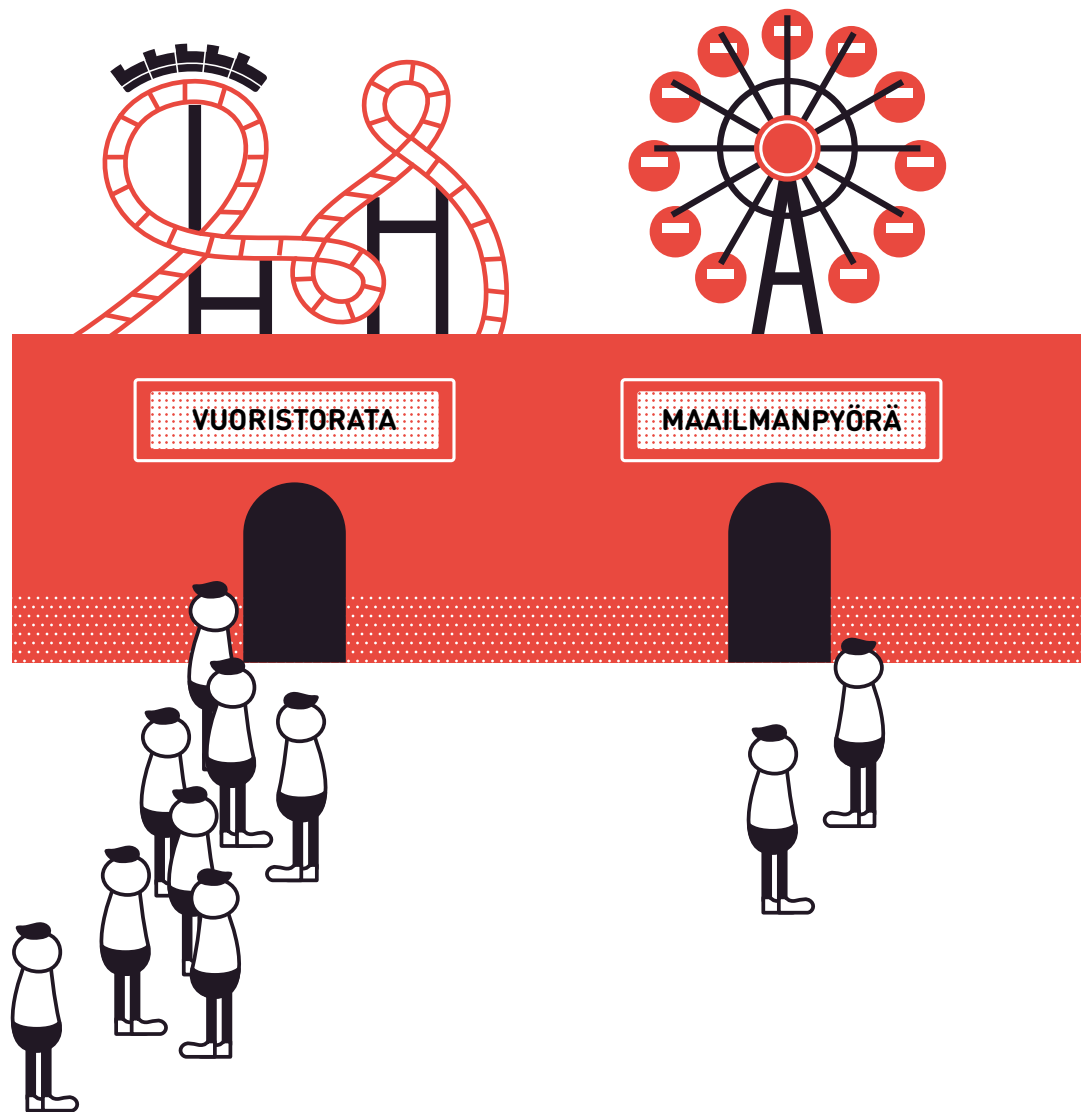
## Testaamalla tiedät mitä ominaisuuksia kannattaa kehittää

Jos et ole aivan varma kannattaako jotain ominaisuutta tehdä, kannattaa asiaa lähteä testaamaan tekemällä koe. Usein yrityksissä testataan paljon asioita. Testailun ja tällaisen muodollisen kokeen ero on se, että kokeelle määritetään tietyt asiat, joihin pyritään vastaamaan.

Olemme ottaneet oppimme Lean-valmentaja Tristan Kromerilta. Kromerin mukaan kokeessa tulee olla määritettynä seuraavat kolme asiaa:

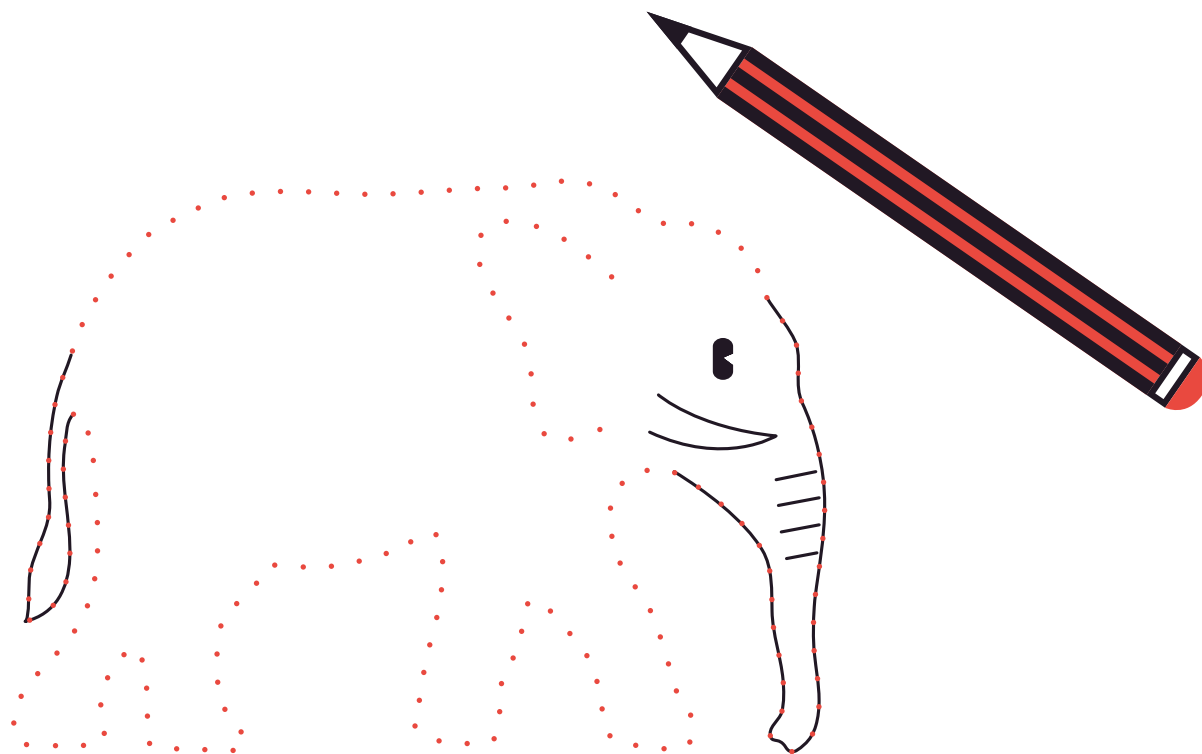
1. Hypoteesi eli oletus siitä, miten asioiden tulisi toimia
2. Oppimistavoite
3. Mittarit. Milloin koe on onnistunut tai epäonnistunut?

Nämä asiat tulee määrittää ennalta, jotta emme päädy muuttamaan seuraamiamme asioita lennosta. Näin saavutettavat tulokset eivät olisi vertailukelpoisia.



## Anna käyttäjän tehdä priorisointi puolestasi

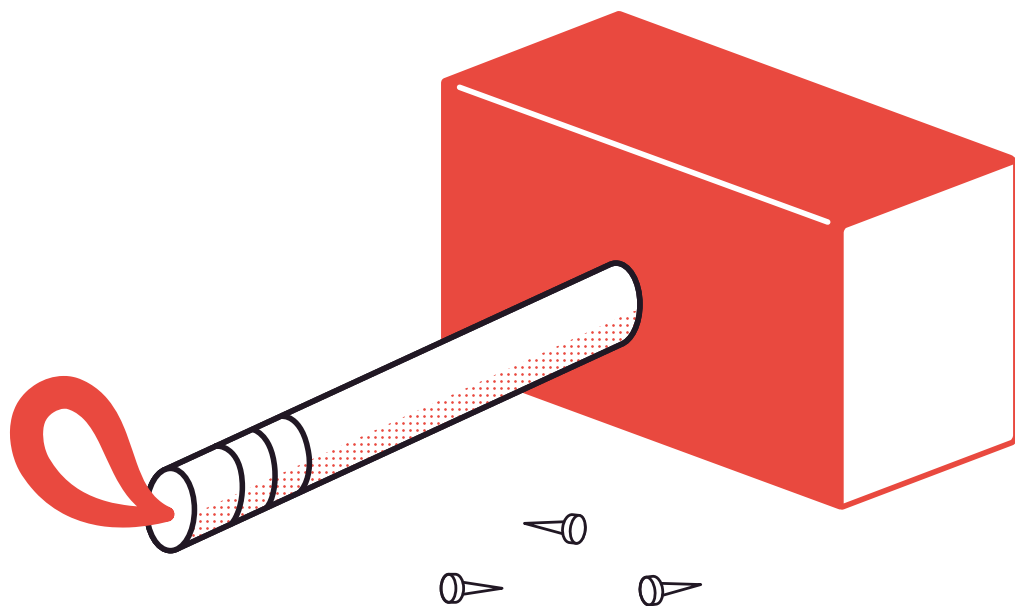
Jos priorisointi tuntuu itsestä vaikealta, anna käyttäjien tehdä se puolestasi. Antamalla loppukäyttäjälle vapaat kädet käyttää prototyyppiä, näemme mitä toiminnallisuuksia hän luontaisesti käyttää. Jotkin ominaisuudet voivat jäädä tyystin käyttämättä, vaikka ne olisivatkin paperilla priorisoitu korkealle. Jos käyttäjät eivät syystä tai toisesta käytä jotakin ominaisuutta, voidaan se tiputtaa kokonaan pois tai alemmas prioriteettilistalla, jolloin joko säästät osan budjetista tai voit käyttää säästetyn osan suosittumpien ominaisuuksien toteutukseen tai jatkokehitykseen.



## Elefantin tarina - Priorisointi on myös jostain luopumista

Toiminnallisuuden priorisointi ei tarkoita ainoastaan niiden listaamista järjestykseen. Priorisointi tarkoittaa järjestämisen lisäksi myös ominaisuuksien karsimista. Priorisoinnilla pyritään löytämään kaksi-kolme sellaista ominaisuutta, jotka luovat tuotteen ytimen.

Kuvitellaan, että projekti olisi elefanti. Jos elefanttia lähdetään toteuttamaan rajallisella määrällä aikaa, mistä ominaisuuksista pitäisi aloittaa? Aloittamalla hännästä ja takaosasta aika loppuu kesken ja elefanti jää tunnistamattomaksi. Sen sijaan aloittamalla kärsästä päästään jo pitkälle, sillä elefanti on tunnistettavissa pelkästä kärsästä. Tämän takia on erittäin tärkeää löytää ne kaksi-kolme kriittistä toiminnallisuutta, joiden ympärille rakennat oman tuotteesi. Kun sinulla on lisää aikaa ja rahaa, voit jatkaa kehitystä korvista ja muista osista.



## Valitse teknologia ongelman suuruuden mukaan

Kuulemme usein lauseita kuten: “Ai te käytätte Railsia, eikö Twitter juuri luopunut siitä?” tai “Facebook käyttää teknologiaa X, miksi te ette käytä sitä?”

Skaala, jolla Facebook tekee töitä on paljon suurempi sekä työntekijöiden että käyttäjien määrän suhteen. Facebookilla on tällä hetkellä noin kaksi miljardia käyttäjää. Jos jokainen suomalainen vauvasta vaariin käyttäisi palveluasi, käyttäjiesi määrä vastaisi silti vain noin 0,2% Facebookin käyttäjistä.

Tosiasia on, että sinun palvelullasi tulee todennäköisesti olemaan vähemmän käyttäjiä kuin Facebookilla on työntekijöitä. Silloin sinun ongelmasi ja Facebookin ongelmat ovat aivan eri mittaluokassa, jolloin käytettävät teknologiat tulisi valita sen mukaisesti.



**Investointi,  
hankinta ja  
kilpailutus**



## iOS, Android vai sittenkin jotain muuta?

Mobiiliapplikaatioita suunnitellessa halutaan sovellus monesti sekä Androidille että iOS -alustoille. Teknisesti tällöin puhutaan natiivi-applikaatioista. Natiivikehitysprojektien alkuinvestointi ja jatkokehitys ovat usein suuria investointeja. Siksi natiiviapplikaation kehitykseen ei kannata lähteä suin päin vaan pohtia onko se paras ratkaisu tämän hetken tarpeeseen.

Markkinat elävät ja emme tiedä, mitä puhelimia käytämme viiden vuoden päästä. Sen takia voi olla ennenaikaista panostaa juuri nyt kahteen erilliseen natiivi-sovellusversioon. Sen sijaan Internet ei suurella varmuudella tule katoamaan. Selainpohjaiseen kehitykseen ei vaikuta se, millainen puhelinmalli kullakin on käytössä nyt tai vuosien päästä. Jos sovelluksesi olisikin web-palvelu, olisi se käytettävissä laajalle joukolle heti julkaisusta lähtien, puhelimilla sekä työpöytaselaimilla. Lisäksi on teknologiaratkaisuja, joilla pystymme paketoimaan web-pohjaisia sovelluksia mobiiliapplikaatioiksi. Näin pystymme kustannustehokkaasti hyödyntämään isoja jakelukanavia kuten App Storea tai Googlen Play Storea.

Joskus natiivisovelluksen kehittäminen on tarpeellista, mutta on tärkeää ymmärtää tarpeet ennen kuin valitaan teknologiaa.



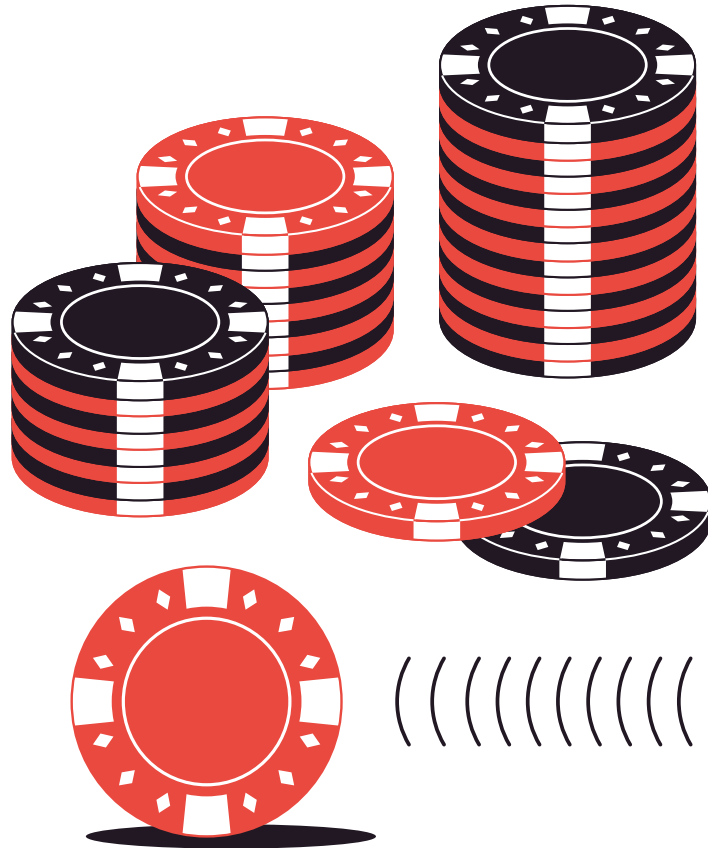


## Kannattaa valita kalliimpi mutta nopeampi softatoimittaja

Maksa korkeampaa tuntihintaa toimittajalle, joka saa palvelusi ulos huomattavasti muita nopeammin. Miksi? Pohditaan tätä esimerkin kautta:

Kilpailijasi kehittää 12 kuukautta tuotetta samalla idealla kuin sinä. Kehität toimittajasi kanssa oman tuotteesi hieman karsituilla ominaisuuksilla markkinoille kolmessa kuukaudessa. Sinulla on yhdeksän kuukautta aikaa saada palautetta ja oppia markkinoilta verrattuna kilpailijaan.

Nyrkkisääntönä on, että start-upeilla on kaksi tilaa: *earn or learn* – tienaa tai opi. Todennäköisesti et vielä palvelusi julkaisuvaiheessa tienaa mitään, koska olet hankkimassa ensimmäiset asiakkaasi, mutta sinulla on mahdollisuus oppia. Kilpailijasi sen sijaan on menettänyt yhdeksän kuukauden opit.



## Budjetin paljastaminen hyödyttää ohjelmistoprojektin tilaajaa

Saamme todella paljon tarjouspyyntöjä, joissa budjettia ei ole kerrottu. Tietyissä tapauksissa budjetti saadaan selville kysymällä, mutta harvoin se meille avoimesti kerrotaan. Jos budjetti kerrotaan heti, pystymme kertomaan asiakkaallemme tarkemmin mikä annetun budjetin rajoissa on mahdollista ja mikä ei.

Budjetin pimittämisessä voi usein olla taustalla se, että halutaan saada mahdollisimman paljon arvoa omalle sijoitukselle. Tietyissä tilanteissa kilpailutus voi ehkä toimia näin, mutta meidän näkökulmasta on selkeämpää pelata avoimin kortein, jotta pystymme vastaamaan asiakkaan tarpeisiin ja tekemään mahdollisimman hyvän tarjouksen.



## Haluatko tietää mitä verkkopalvelun tekeminen maksaa?

Ei ole ollenkaan tyhmää kysyä, mitä verkkopalvelu maksaa. Vastauksen antamiseksi tarvitsemme kuitenkin huomattavan paljon tietoa, koska tuottamamme palvelut eivät ole hyllystä poimittavia ratkaisuja.

Kun haluat vastauksen kysymykseen, ole esimerkiksi valmis kuvailemaan omin sanoin mitä käyttäjä palvelussa tekee, tuleeko palvelun olla yhteydessä muihin palveluihin tai toimiiko se puhelimessa. Pyri antamaan mahdollisimman laaja kuva siitä, mikä toivetilasi palvelulle on. Tällöin voimme antaa jo aika hyvän hintahaarukan sille, mitä toteuttaminen kustantaa.



## Oikeudet lähdekoodiin kuuluvat tilaajalle

Kannattaa pitää huolta, että kun palvelua lähdetään kehittämään sinun ideastasi, myös ohjelmiston lähdekoodi kuuluu sinulle. Meistä on absurdi ajatus, että sinun tilaamasi ja ideoimasi palvelun ohjelmistokoodit kuuluisivat meille.

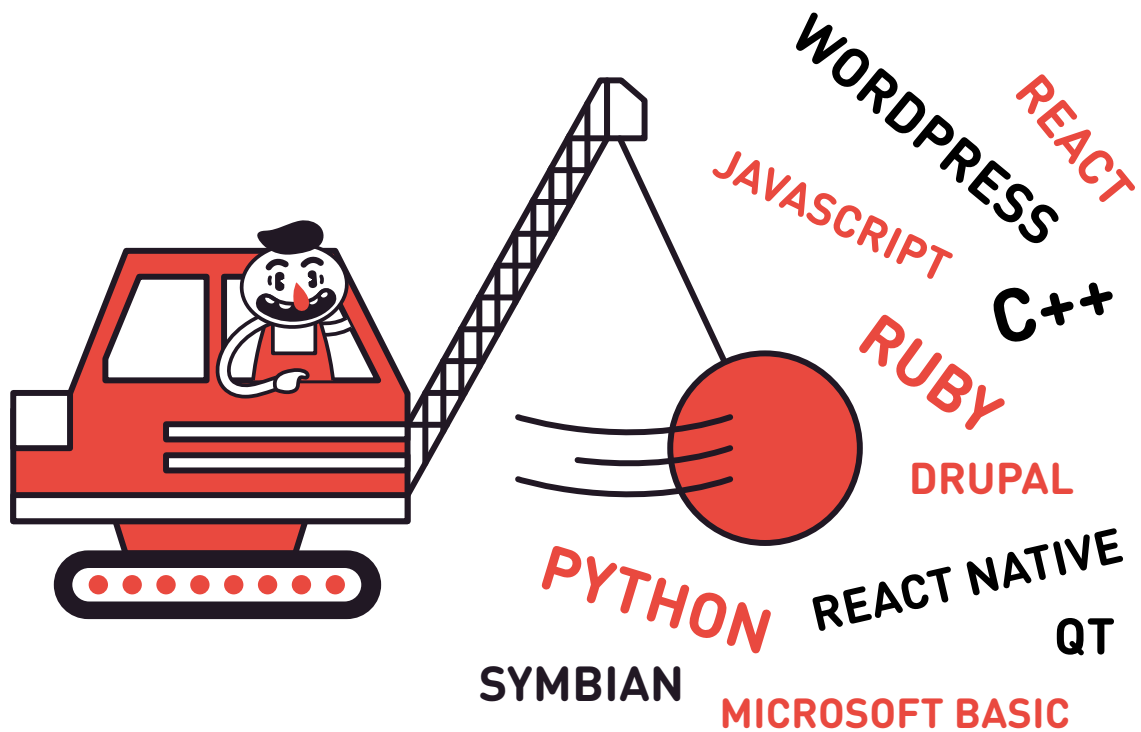
Kun sinulla on oikeudet lähdekoodiin, voit viedä koodin toiselle toimittajalle jatkokehitykseen, rekrytoida oman tiimin kehittämään palvelua tai jopa myydä palvelusi. Tämä ei välttämättä ole mahdollista mikäli alihankkijalla on oikeudet tuottamaansa koodiin. Pidä siis huoli siitä, että tilaamasi koodin oikeudet kuuluvat sinulle.



## Ohjelmiston ostajan paradoksi

Olet saattanut joskus ostaa organisaatioon ohjelmiston ajatellen, että nyt tuli tehtyä hyvät kaupat. Pian on kuitenkin alkanut kuulumaan soraääniä, ettei ohjelmisto olekaan hyvä eikä sitä ole kiva käyttää. Kyse on ilmiöstä, jossa ostaja ei käytä ja käyttäjä ei ole ostanut.

Miten siis ostaa ohjelmisto, jota käyttäjä haluaa käyttää? Ottamalla heidät mukaan jo alkuvaiheessa suunnittelu- ja testaustyöhön muun muassa antamalla käyttöliittymäprototyyppi käyttäjille testattavaksi ja keräämällä tästä palautetta. Totuus on, että ihmiset käyttävät paljon mieluummin järjestelmää, joka on kehitetty kuulemalla heitä ja joka palvelee heidän työtehtäviään täydellisesti.



## Vanhan lähdekoodin saneeraus, vai uuden rakentaminen?

Kun mietit softan tilaamista, sinulle saattaa tulla houkutus hyödyntää vanhaa lähdekoodia. Useimmiten tämä on erittäin suuri virhe.

Koodi ei itsessään ole itseisarvo vaan liiketoimintasi ja asiakkaasi ovat. Parempi tapa toimia ja luoda menestyvä tuote on pohtia sitä, minkälainen tuote rakennetaan. Pohdinnan lopputuloksena tehdään projektin pienin vaihe eli koodaaminen.

Jos liiketoiminta lähtee muuttumaan, niin kuin kasvuyrityksissä usein huomataan, on osattava päästää irti lähdekoodista, sillä se ei ole arvokainta. Matkan varrella kertyneet opit ja kokemukset ovat niitä, jotka haluat ottaa mukaasi. Älä siis väkisin väännä vanhaa palvelemaan uutta vaan uskalla päästää irti toimimattomasta lähdekoodista.



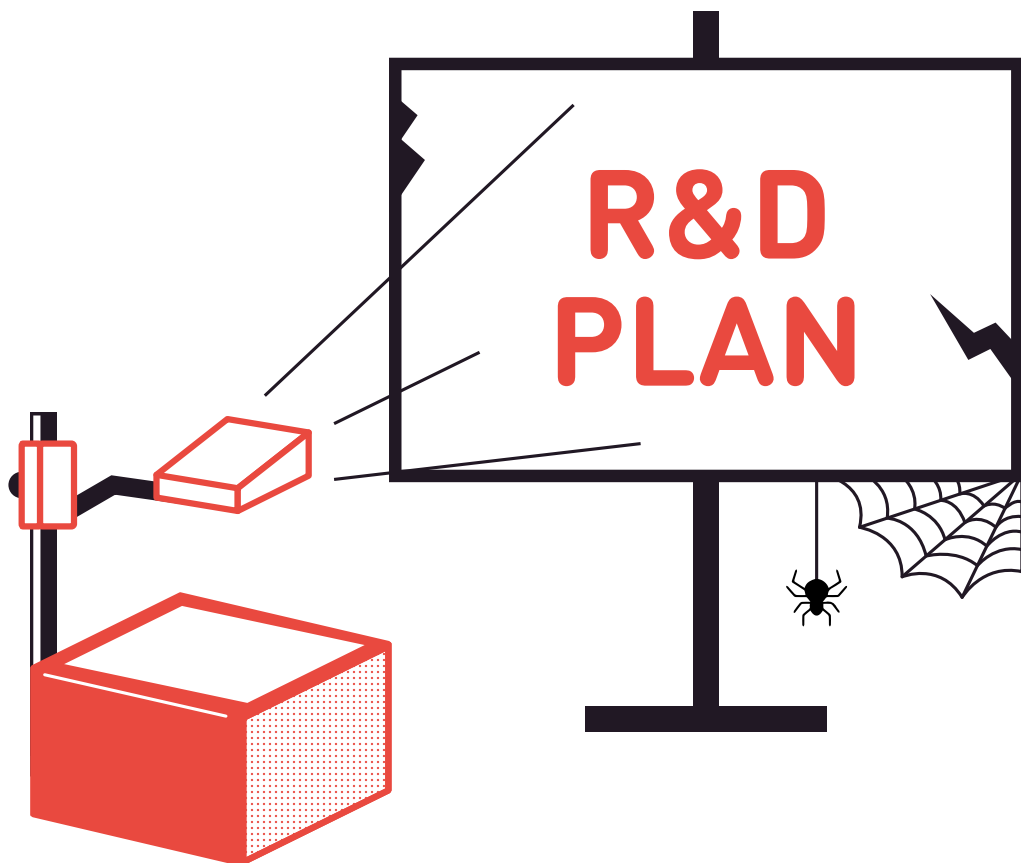
**Prototyyppi**



## Prototyypillä saat esimiehesi innostumaan ohjelmistoprojektista

Jos sinun täytyy myydä omalle esimiehellesi ajatus tai sinulla on idea uudesta liiketoiminnasta tai sovelluksesta, lähetkö esittelemään sitä PowerPointilla? Se on tylsä tapa esitellä ja ihmiset eivät yleensä innostu kalvosulkeisista. Jos taas esittelet ideasi prototyypillä, joka on jo puhelimesta oikean näköinen sovellus, herätät ihmisissä tunteita ja inspiroit heitä näkemään kuinka ideasi todella toimii. Tämä on todella hyvä tapa lähteä viemään yrityksesi johdolle uutta ideaa, koska visio-  
tulvan sijaan he saavat ideasi käsin kosketeltavaksi.





## Tekstidokumentti spekseistä vai prototyyppi?

“Me ei tarvita prototyyppiä” on usein asiakkailta tuleva ajatus. Lähtötilanteessa uskotaan, että tiedetään miten sovellus tulee toimimaan ja mitä tarpeita sen käyttäjillä on. Mikä prototyypin hyöty sitten oikeasti on? Huomioi erityisesti nämä kaksi asiaa:

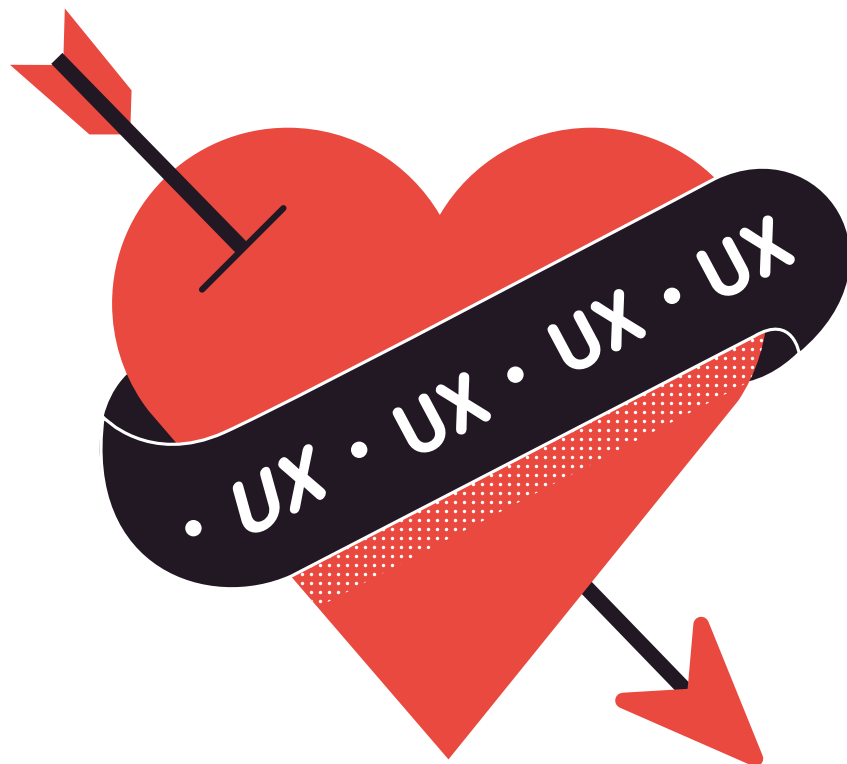
1. Jos meillä on tekstimuotoinen määritelmä palvelusta, jokaiselle muodostuu oma näkemyksensä siitä, miltä lopputulos tulee näyttämään. Visualisoimalla palvelun käyttöliittymän saamme yhtenäistettyä ajatuksemme tavoitellusta lopputuloksesta, mikä vähentää projektin epäonnistumisen riskiä huomattavasti.
2. Prototyyppi tuo ajatukset lähemmäksi todellisuutta. Usein kun palvelun ensimmäinen versio annetaan johdolle testattavaksi, syntyy uusia ideoita tai tarpeita, joihin ei ennalta oltu varauduttu. Prototyypaamalla voidaan siis kerätä palautetta ja reagoida yllättäviin tarpeisiin ennen kuin palvelua on alettu kehittämään ja resurssit voidaan keskittää niihin asioihin, jotka oikeasti ovat käyttäjille tärkeitä.



## Prototyyppi auttaa priorisoinnissa

Mitä on priorisointi? Olemme usein nähneet priorisoitavan asioita niin, että palvelulle määritetään 20 tarvittavaa ominaisuutta, jotka laitetaan toteuttamisjärjestykseen listalle. Meidän näkemyksemme palvelun ensimmäistä versiota tehtäessä on, että tuon ominaisuuslistan pohjalta tulee olla valmis jättämään 30% pois palvelusta. Monelle tämä tulee uutena asiana: priorisointi on opittu näkemään järjestyksenä asioiden toteutumiselle.

Meidän mielestämme priorisoinnin tulee olla prosessi, jossa karsit palvelustasi jotain pois. Mutta mistä tiedetään, mitkä ovat niitä pois jätettäviä asioita? Prototyypit ovat yksi keino. Prototyypeillä kysytään todellisuudelta palautetta, eli testataan palvelua kohdeyleisölläsi ja selvitetään eri ideoiden toimivuutta. Toimimattomat ideat voimmekin laittaa surutta silppuriin ja keskittyä niihin, jotka todella palvelevat loppukäyttäjiä.

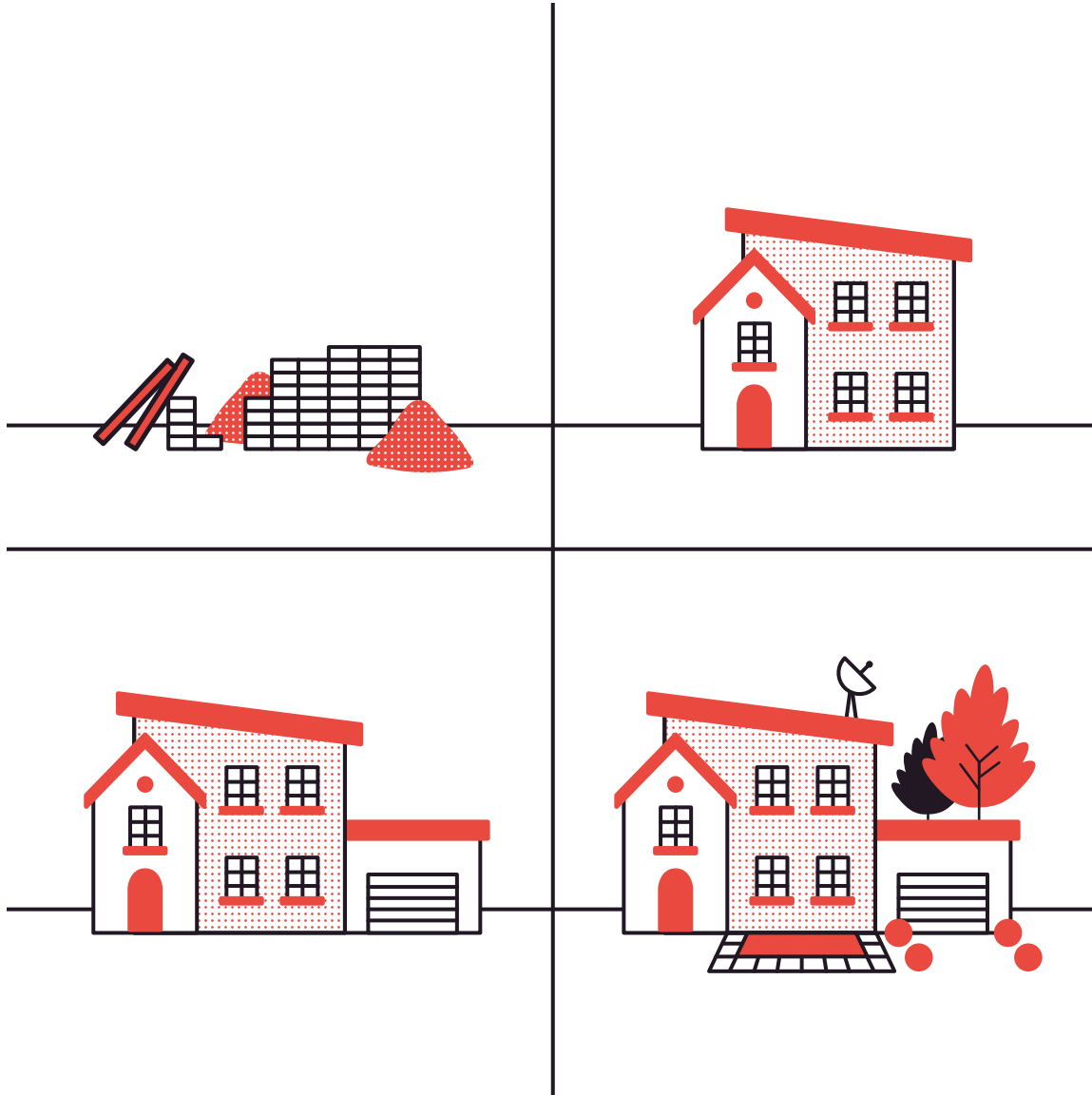


## Käytettävyys ennen ulkonäköä

Uutta sovellusta suunniteltaessa tulisi ensisijaisesti miettiä, miltä sovellus tuntuu, ei niinkään miltä se näyttää. Graafinen suunnittelu on toki tärkeää, mutta se ei ole ensimmäinen prioriteetti.

Mietitään vaikka kyytipalvelu Uberia. Taikuus ei heidän tapauksessa tule siitä, miltä heidän sovellus näyttää vaan palvelun käytön vaivattomuudesta. Eli toisin sanoen siitä, että nappia painamalla joku tulee hakemaan sinut, vie sinut haluamaasi kohteeseen ja maksut hoituvat automaattisesti taustalla.

Käytettävyyden suunnittelu vaatii muutakin kuin hahmottelua paperille. Tarvitset elävämpiä työkaluja kuten prototyyppejä, jotta pystyt luomaan palvelun, jota käyttäjät oikeasti rakastavat käyttää.



## Prototyypillä etumatkaa varsinaisen palvelun kehittämiseen

Yleinen ajattelumalli prototyyppejä tehtäessä on, että kun prototyyppi on saatu valmiiksi ja sen viestintäarvo on saavutettu, sille ei nähdä enää arvoa ja se heitetään pois. Kiskolla asiat tehdään toisin.

Kun me teemme prototyypin, se on jo ensimmäinen versio valmiin soveluksen käyttöliittymästä, johon suunnitteluprosessin aikana lisätään uusia ominaisuuksia ja joka lopulta muotoutuu oikeaksi palveluksi loppukäyttäjien käsissä. Tällöin ei tehdä mitään “turhaan” vaan saadaan etumatkaa palvelun kehittämiseen heti alussa: kun tehdään päätöksiä prototyypissä toimivien ominaisuuksien säilyttämisestä lopullisessa tuotteessa, voidaan hyödyntää jo tehtyä työtä.



## Puistosuunnittelijan prototyyppi

Kuvittele, että olet suunnittelemassa puistoa ja mietit kulkureittejä puistoon. Teet esimerkiksi haastatteluja ja juttelet ihmisille siitä, minne he haluaisivat kulkureitit sijoitettavan. Rakennetaan puisto ja sijoitetaan kulkureitit haastattelujen perusteella. Parin kuukauden kuluttua huomaat kuitenkin, että kulkureittien lisäksi puistoon on alkanut syntyä polkuja muuallekin kuin ennalta määriteltyihin sijainteihin. Rakentamalla aluksi pelkän nurmikentän ilman kulkureittejä, saisimme ihmisten todellisen käytön perusteella tehtyä reitit heti sinne, missä niille on tarvetta.



## Projekti­kommunikaatio



## Arvasitko jo ennalta miksi projektisi meni pieleen?

Onko sinulle joskus käynyt niin, että projekti on mennyt pieleen juuri sillä tavalla kuin arvelit? Tai onko joku muu sanonut, että he arvasivat tietyn asian menevän projektissa päin mäntyä? Meillä on hankkeissa eri ihmisillä piilevää tietoa siitä, mikä projektissa voi mennä pieleen. Kuten lentoliikenteessä ja laskuvarjohypyissä, riskien minimointi on ykkösjuttu. Siihen pyrimme myös softaprojekteissamme. Onkin siis tärkeää, että tuo piilevä tieto saadaan hyödynnettyä heti alusta asti.

Ennen hanketta istumme asiakkaamme kanssa alas ja teemme niin sanotun *pre mortem*-harjoituksen. Monille tuttu harjoite on *post mortem*, jossa hankkeen jälkeen reflektoidaan yhdessä mikä meni hyvin tai huonosti. *Pre mortemissa* tämä tehdään ennen kuin työt aloitetaan ja käydään läpi isoimmat projektiin liittyvät pelot ja riskit. Näin nämä tulevat koko kehitysryhmälle näkyväksi ja voidaan tietoisesti välttää niiden realisoituminen projektin aikana.



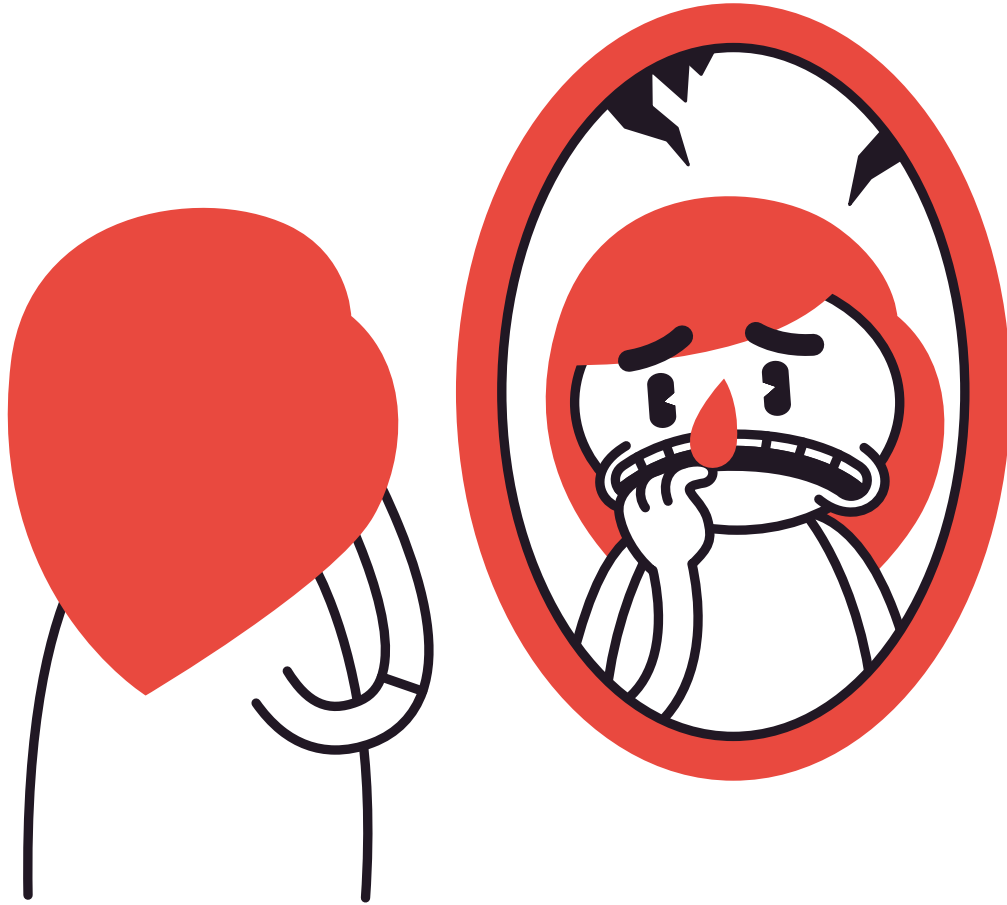
## Oikea aika vaikeille kysymyksille

Vaikeat kysymykset pitäisi esittää heti projektin alussa tai kun niitä tulee eteen. Tämä tarkoittaa sitä, että vaikeita, jopa ikäviä kysymyksiä saatetaan kysyä jo myyntivaiheessa. Saatat joutua ottamaan selvää asioista, joista et tiedä, kysymään talousasioista ja niin edelleen. Näiden kysymysten tarkoitus on se, että voimme luvata ja toteuttaa asioita jossakin aikataulussa. Haluamme tiedostaa kaikki möröt jo heti alkuvaiheessa, jotta voimme luvata asiakkaalle oikeat aikataulut ja pysyä niissä.





**Julkaisu**



## Rauhoittavia sanoja julkaisukammoiselle

Siinä vaiheessa kun projekti saadaan suurin piirtein maaliin ja päästään keskustelemaan julkaisusta, todella useilla asiakkailla tulee eteen julkaisukammo. Kaiken tulee olla viimeisen päälle viilattua ja loppuun hiottua ennen kuin palvelu voidaan julkaista. Sanamuotoa täytyy hieman muuttaa, valittu kuva alkaa epäilyttämään – aina löytyy jotain. Usein ollaan kuitenkin vielä siinä pisteessä, että käyttäjät lasketaan kymmenissä miljoonien sijaan eikä heidän käyttäytymisestään palvelussa ole opittu vielä tarpeeksi.

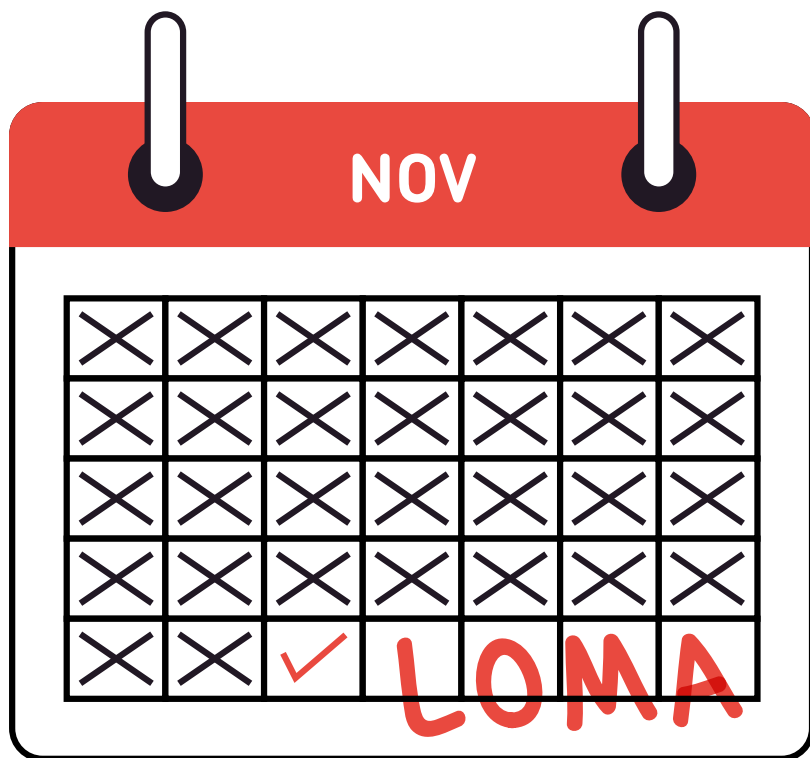
Viimeistelyn määrästä huolimatta palvelusi käyttäjät tulevat antamaan palautetta, niin positiivista kuin negatiivistakin. Viivästyttämällä palvelusi julkaisua viivästyttät myös palautteen keräämistä, jolla voidaan oppia lisää ja kehittää palvelua eteenpäin.



## Valmiiksi saamisen vimma voi pilata lopputuloksen

Osa ihmisistä vaivaa valmiiksi saamisen vimma – tarve, joka saa meidätkin joskus valtaansa. Ennen joulua ja juhannusta on usein kova paine saada projektit valmiiksi.

Vaikka kuinka on julkaisuvimma tai lomat painaa päälle, ei tule ikinä unohtaa loppukäyttäjää ja bisneksen tarpeita. Minkä takia olemme alunperin lähteneet rakentamaan tätä palvelua? Kaikki muu on toissijaista.

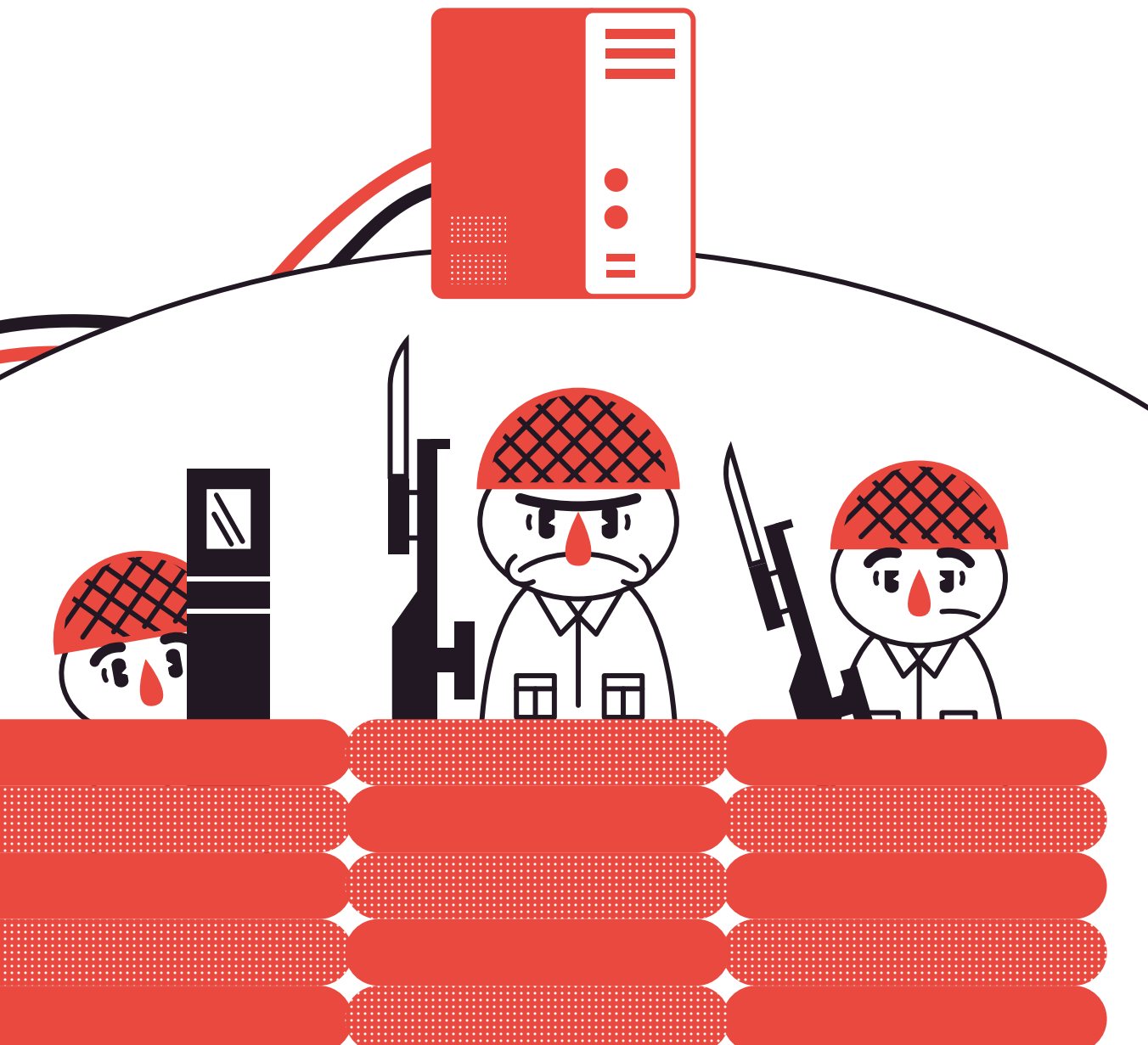


## Mieti tarkkaan milloin julkaiset palvelusi

Joulu ja juhannus ovat todella suosittuja ajankohtia mennä naimisiin. Näin myös projektien osalta: haluamme lanseerata palveluita jouluna ja juhannuksena. Miksi näin?

On totta, että uusi kausi on mukava aloittaa puhtaalta pöydältä, mutta onko joulukuu paras aika lanseerata uusi verkkopalvelu? Tuskin. Jos mietitään tarkemmin, miten vietämme juhlapäiviä, meillä kaikilla on omat perinteemme ja rutiinimme, joita noudatamme. Osa saattaa olla menossa jopa naimisiin. Se ei ole ehkä kuitenkaan se hetki kun alat tutustumaan ja ottamaan käyttöön uutta palvelua.

Mieti siis tarkkaan milloin julkaiset palvelusi.



## Softahankkeesi puolustusvoimat

Joskus ohjelmiston julkaisijan oma jännitys on kaataa koko hankkeen

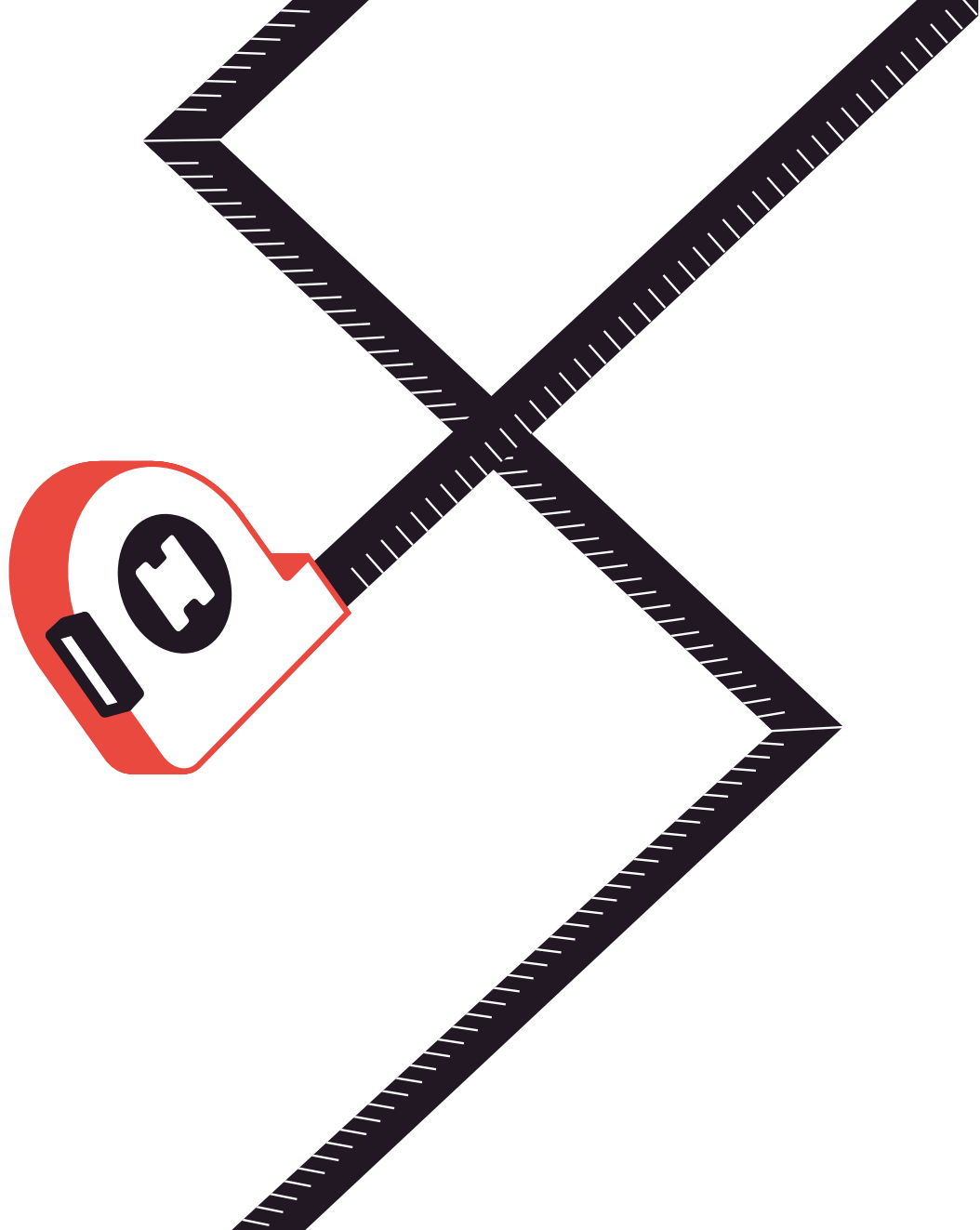
– kotoa ei uskalleta lähteä julkaisupäivänä tai alkaa muuten vain ihmeellinen sekoilu.

Meille on kunnia-asia, että projekti pääsee ulos ja autamme asiakkaamme yli julkaisukammosta. Julkaisukammo on muuten erittäin yleistä kun tehdään uusia tuotteita. Me jaksamme kun sinä et jaksa.

Me pidämme visiosi mielessä jos se meinaa unohtua.



**Arviointi**



## Viisastu mittaamalla

Projektilla kuin projektilla pitäisi olla onnistumisen mittarit – ne määritellään, jotta jo ideavaiheessa olisi mahdollista käydä dialogia loppukäyttäjän kanssa. Tämä jälkeen voidaan katsoa, onko mitattavilla tavoitteilla edellytykset täyttyä ideatasolla. Samoin prototyypin kanssa: meillä pitää olla mittarit, joiden avulla voimme haastatella loppukäyttäjää ja saada palautetta toiveiden toteutumisesta.

Yksi parhaista esimerkeistä hyvästä mittarista on kysyä loppukäyttäjältä, olisiko hän valmis maksamaan palvelusta? Toinen esimerkki, jota voi soveltaa prototyypin kanssa, on seurata loppukäyttäjän käyttämiä ominaisuuksia, kun hän saa vapaasti käyttää sovellusta. Näin näemme, mitkä asiat loppukäyttäjää palvelussasi oikeasti kiinnostavat. Usein myös huomataan, että prioriteettilistalla tärkeäksi koettu ominaisuus jää täysin käyttämättä, jolloin voidaan harkita resurssien uudelleen allokointia.



## Onnistuiko softaprojektisi oikeasti?

Usein yrityksissä saattaa vallita yleinen konsensus, että ohjelmistoprojektit ovat menneet aiemmin todella hyvin. Hyvä, jos todella on näin. Mutta kun aletaan kysymään kysymyksiä, kuten pitikö aikataulut tai pitikö budjetti, niin alkaakin selviämään asioita, joiden perusteella projekti ei mennytkään ihan niin hyvin kuin kuviteltiin.

Syy, minkä takia sanotaan, että ohjelmistoprojekti on mennyt hyvin on se, että vastoinkäymisistä huolimatta on päästy haluttuun lopputulokseen. Ammattilaisen silmin tämänkaltaiset projektit ovat silti jossain määrin epäonnistuneet.

Tästä pääsemme ydinkysymykseen: miten ohjelmistoprojekteja tulisi tehdä? Mielestämme projektit tulisi pilkkoa riittävän pieniin osiin, jotta kaikki osapuolet ymmärtävät mitä ollaan tekemässä. Aloita idean pohtimisesta ja jalosta ideoita eri kokoonpanoilla. Seuraavaksi mieti ja testaa ratkaisua. Toteuta. Vie tuotantoon.

Kun ratkaisu on mietitty tarkasti, sille on mahdollista antaa erittäin tarkka aikataulu- ja budjettiarvio.





## Niskalaukaus “ihan ok” projekteille

Huonoimmat projektit eivät ole menestyneitä, mutta eivät myöskään täysin epäonnistuneita. Liiketoimintana nämä voivat olla nipin napin kannattavia, mutta palavereihin ja muuhun päivittäiseen tekemiseen menee suuri määrä aikaa.

Jos resurssit vapautettaisiin johonkin muuhun, todennäköisesti keksittäisiin jotain parempaa. Tällaisissa projekteissa osalliset usein kokevat luopumisen tuskaa, koska siihen on mennyt niin paljon aikaa ja työtä, vaikka todellisuudessa projekti ottaa enemmän kuin antaa.

Mieti onnistumisen mittareita tarkasti. Näin näet itsekin, kun projekti ei saavuta sille asetettuja tavoitteita ja pystyt helpommin reagoida tekemällä muutoksia tai jopa lopettamalla projekti kokonaan.



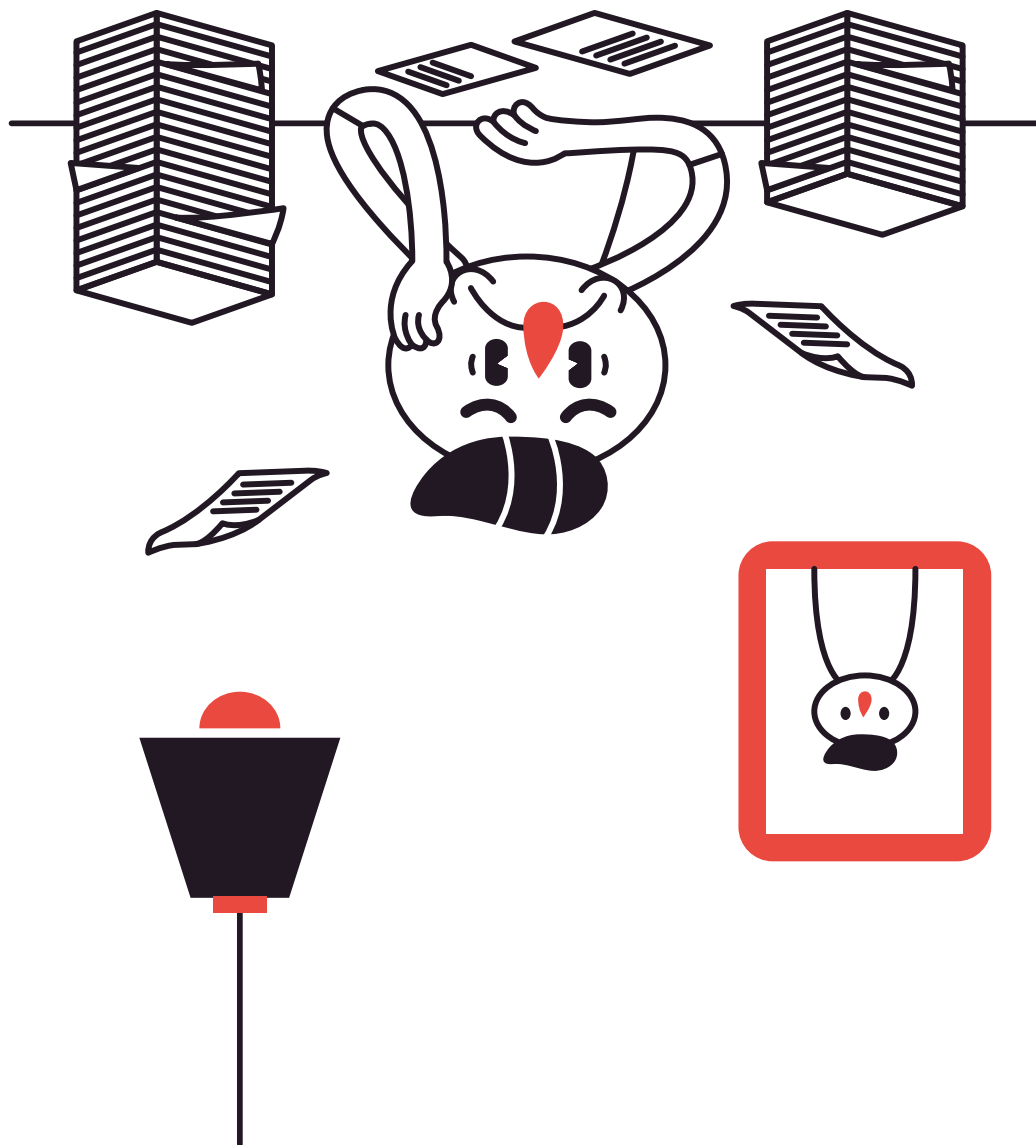
**Jatkokehitys**



## Palvelimet on %\$@#!

Onneksi olkoon! Olet ostanut onnistuneesti softaprojektin ja maksanut siitä parikymmentä, kenties yli 100 000 euroa ja pian ollaan maalissa. Nyt pitäisi miettiä, missä palvelua ylläpidetään. Otetaanko luotettavalta toimittajalta palvelin, joka maksaa 20€ kuukaudessa? Vai maksetaanko tästä Herokulle 200€ kuukaudessa? Melkein aina oikea vastaus on, että maksetaan Herokulle 200 euroa kuukaudessa.

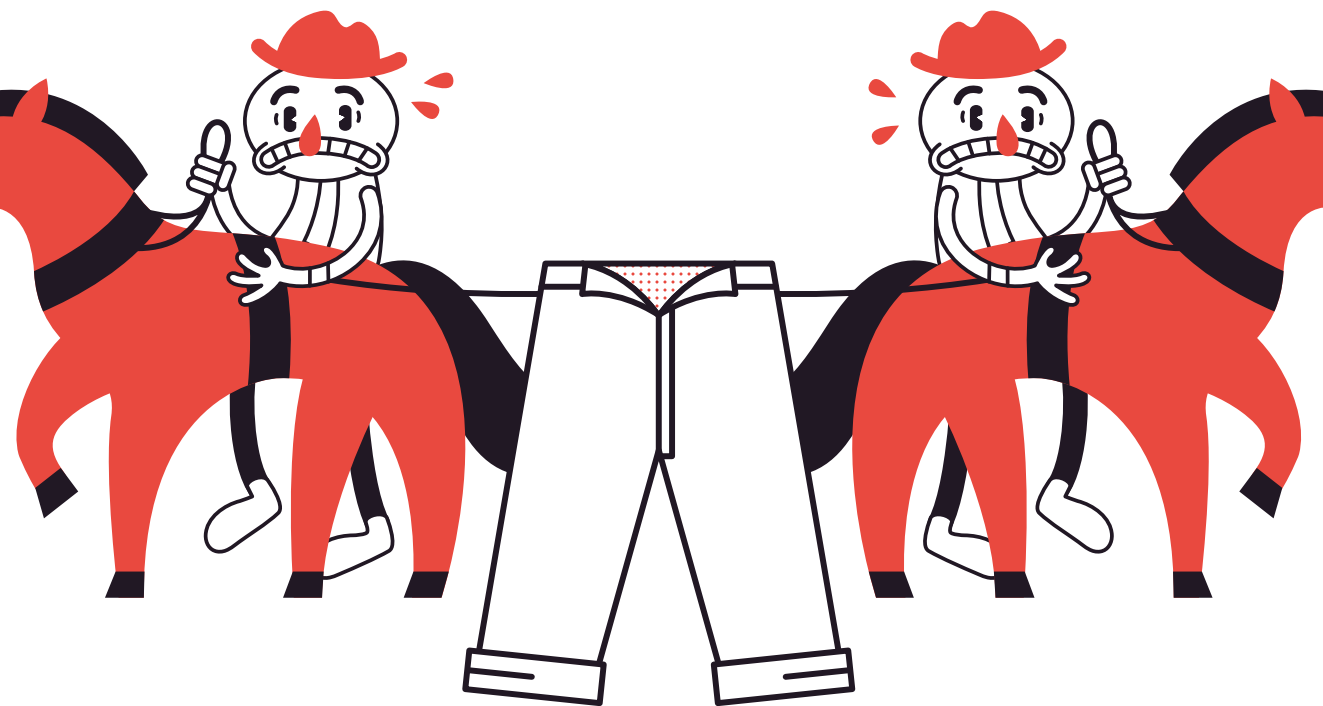
Mieti tilannetta, että olet säästänyt rahaa, ottanut sen halpispalvelimen ja jossain vaiheessa se palvelin kaatuu keskellä yötä. Onko siellä ketään laittamassa sitä takaisin käyntiin? Onko siellä ketään varmistamassa, että se lähtee käyntiin kunnolla? Jos olisit ottanut käyttöösi jonkin Platforms As A Service -palvelun, kuten Herokun, olisi sinulla joku aina huolehtimassa, että palvelusi pyörii ajasta riippumatta ja voit nukkua yösi rauhassa.



## Käännä kärsivällisyys ylösalaisin

Monia saattaa ärsyttää se, ettei tuote kovista odotuksista huolimatta ole lähtenyt kasvamaan räjähdysmäisesti. Silloin saattaa olla ongelma, että jokapäiväisessä tekemisessä saatetaan olla kärsivällisiä, hioa yksityiskohtia täydellisyyteen asti ja niin edelleen, mutta pitkällä aikavälillä ollaan todella kärsimättömiä.

Tämä ajattelumalli tulisi kääntää toisin päin: ole nopea joka päivä, tee kokeita ja ota pieniä askelia päivittäin. Pitkällä aikavälillä sinun pitää olla todella kärsivällinen, sillä järkevää liiketoimintaa tehdään 20 vuoden sykleissä.



## Telttatehtailija Levi Straussin tarina

Levi Strauss teki alunperin telttoja kullankaivajille. Hän kuitenkin piti korvansa auki ja kuuli, että telttoja enemmän kullankaivajat olisivat tarvinneet kestäviä housuja. Hän rupesti tekemään telttakankaista housuja ja loppu onkin historiaa.

Mitä tästä opimme? Softaprojektiakin tehtäessä kannattaa jalkautua asiakkaiden pariin. Mitä sinusta puhutaan kun sinun ei oleteta olevan kuulemassa? Mene eri foorumeille tai someen lukemaan käyttäjien suurimmista ongelmista, iloista ja suruista. Tämänkaltaisen tutkimustyön perusteella on helppo lähteä kehittämään palvelua suuntaan, johon käyttäjät ovat tyytyväisiä, etkä turhaan käytä resursseja vähemmän tärkeisiin ominaisuuksiin.



## Anna sovelluksesi kaatua

Jos sovelluksessasi tapahtuu odottamaton virhe, sovelluksen tulisi kertoa tästä käyttäjälle ja auttaa hänet takaisin normaalitilaan. Tämän jälkeen sinulla on aikaa tutkia ongelmaa, korjata se ja ilmoittaa käyttäjälle vian korjauksesta.

Mitä kenenkään ei tulisi tehdä, on jollakin epätoivoisella tavalla selviytyä sattuneesta virheestä. Tästä nimittäin seuraa usein vain lisää virheitä, jotka tapahtuvat myöhemmin. Tällöin ongelmaksi koituu selvittää, mistä alkuperäinen virhe aiheutui. Joissain tapauksissa joudutaan jopa tekemään työläitä varmuuskopiopalautuksia korruptoituneen datan takia. Anna siis sovelluksesi rohkeasti kaatua ja korjaa alkuperäinen virhe samantien.



**Strategia ja  
liiketoiminta**



## Palvelusi pyhät hetket

Kaikilla ammattilaisilla on omat pyhät hetkensä. Myyjälle pyhä hetki on asiakastapaaminen. Tämä tarkoittaa sitä, että myyjä on valmistautunut tapaamiseen, ottanut selvää asiakkaan liiketoiminnasta, miettinyt mistä tapaamisessa keskustellaan ja ehkä jopa tehnyt selvitystyötä tapaamiseen osallistuvista henkilöistä.

Digitaalisten palveluiden kohdalla tulisi yhtälailla miettiä pyhät hetket etukäteen. Mikä on tärkein hetki palvelua käytettäessä? Mitä asiakas odottaa palvelulta? Kauanko asiakkaan kanssa on aikaa? Mitä haluat tarjota asiakkaalle? Hoida palvelusi pyhät hetket mahdollisimman hyvin ja asiakkaasi todennäköisesti palaavat palvelusi pariin.





## Matalan riskin start-up rahoitusstrategia

Suurta osaa start-up pitchejä yhdistää niin sanottu "Hockey stick"-kaavio. Kaavio kuvaa liikevaihdon kehitystä ajan akselilla.

Kun tehdään isolla riskillä ja rahoituksella oletettavasti mullistavaa asiaa, on kiire tehdä läpimurto, sillä rahoittajat haluavat sijoitukselleen tuottoa mahdollisimman paljon ja mahdollisimman nopeasti. Tämä onnistuu toisinaan, mutta se on niin harvinaista, että tässä onnistuneita startuppeja kutsutaan yksisarvisiksi.

Mitä jos kehittäisit palveluasi tulosrahoitteisesti ja investoisit saadut eurot uusiin ominaisuuksiin, joista asiakas on valmis maksamaan? Uskomme, että hieman hitaammin ja rauhallisemmalla tahdilla pääset myös erittäin kannattavaan bisnekseen. Kaikki tämä ja vielä pienemmällä riskillä!

Ei niin villiä, ei niin hienoja bileitä, mutta uskomme kestävään kehitykseen myös liiketoiminnassa. Valitsetko yksisarvisen vai hevosen?

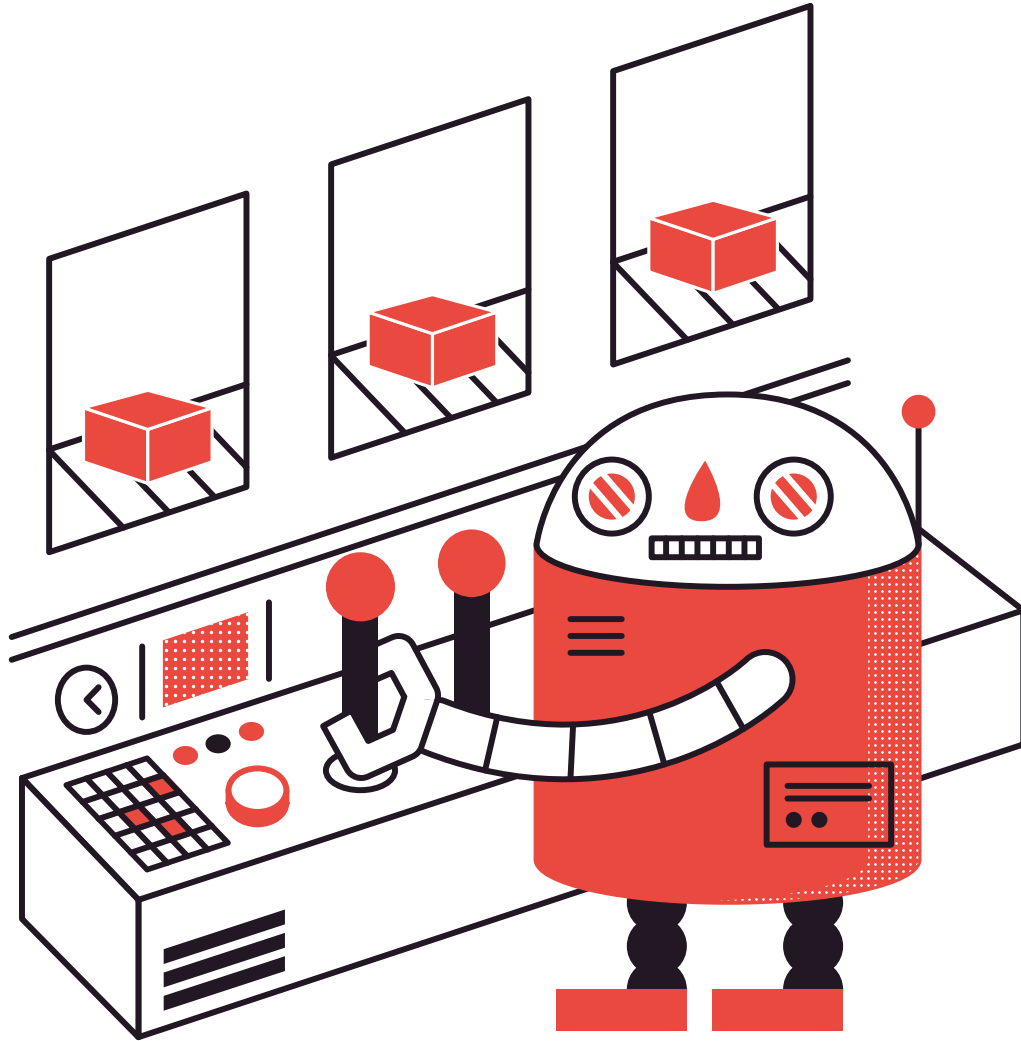


## Tee vaikeita asioita

Sinulla on kenties haave isosta ja mullistavasta palvelusta, jonka haluat toteuttaa. Välillä se saattaa tuntua liiankin isolta ja vaikealta. Meidän tehtävä on pilkkoa kokonaisuus järkeviin ja saavutettaviin välietappeihin, jotta päästään liikkeelle ja hiljaksiin kohti tavoitetta. Esimerkiksi jos olet menossa Mount Everestille; sinne on mahdollista päästä, mutta ei ilman opastusta eikä varsinkaan ilman välietappeja.

Saku Tuominen on sanonut: ”Kuulostaa vaikealta, tehdäänkö silti?”

Tee vaikeita asioita, sillä ne ovat todennäköisesti merkittäviä asioita. Kun ahdistus iskee päälle, pilko tai pyydä jotain muuta pilkkomaan hankkeesi sinulle.



## Muuttuvassa maailmassa tietä digitalisaatioon ei voi valita

Jo pidemmän aikaa on ollut vallalla ilmiö, että joka suunnasta tulee vinkkejä, miten digitalisoituvassa maailmassa tulisi muuttua ja mihin tulisi investoida. Tämä on todennäköisesti arvailua, sillä tulevaisuudesta ei kukaan voi täysin tietää. Olemme miettineet asiaa paljon ja päätyneet siihen, ettei ole olemassa yhtä ainoaa tietä tai ratkaisua, vaan se pitää löytää pienin askelin. Sen takia olemme käyttäneet työkaluina prototyyppejä ja kokeita jo kymmenen vuotta, edenneet pienin askelin ketterästi.

Se on selvästi myös tulevaisuuden suunta.

*Jos kadulla kuulemme ihmisen ennustavan  
maailmanloppua, suurin osa meistä tuomitsee  
hänet päässään mielenterveyspotilaaksi.*

*Töissä tulevaisuuden ennustajat saavat tittelin  
"visionääri".*

