

Verkkopalvelualusta - miten valitsen oikein?

Kun yrityksessä tai yhteisössä suunnitellaan verkkopalvelun kehitysprojektia, nousee heti alkuvaiheessa esille käytettävän palvelualustan valinta. Kysymys on oleellinen, sillä se vaikuttaa palvelun käytettävyyteen, kustannuksiin ja kehitysmahdollisuuksiin koko verkkopalvelun elinkaaren ajan. Usein toteutuskumppanin ja palvelualustan valinta kulkee käsi kädessä.

En väitä, että otsikon kysymykseen olisi yhtä oikeaa vastausta - vaikka uskon oman ratkaisumme vahvuksiin, tiedän myös että tyydyttävä lopputulos on saavutettavissa muillakin välineillä - kun ne vain soveltuvat käyttötarpeeseen. Siksi esitänkin tässä listan asioista, joihin minusta on hyvä kiinnittää huomiota valintaprosessissa. Myönnän että teksti on melko pitkä, mutta toivon että jaksat lukea sen loppuun saakka.

Aloitetaan käytettävyykysymyksillä - vaikeakäyttöinen järjestelmä on huono työväline samaan tapaan kuin väärän kokoinen lenkkiavain tai tylsä saha. Hommat saattaa saada tehdyksi, mutta se ei ole mukavaa. Vertaile järjestelmävalintaa tehdessäsi esimerkiksi seuraavia asioita:

Toimiminen järjestelmän kanssa

- Kuinka helppoa on tehdä uusia pääsivuja tai alasivuja, tai sijoittaa niitä eri kohtaan valikossa (sivukartassa)
- Kuinka helppoa on lisätä palveluun kuvia, tiedostoja, flash - elementtejä?
 - Voiko niitä päivittää?
- Kuinka helposti löydän haluamani toiminnon?
- Näenkö suoraan miltä muokkaamani sivu näyttää, vai täytyykö mennä erillisen esikatselun kautta?
- Voinko antaa muille oikeuksia muokata sisältöjä, ja vain tiettyjä sisältöjä?
- Voivatko käyttäjät rekisteröityä itse sivuille?
- Voinko tehdä sivuille osioita, jotka näkyvät vain tietyille käyttäjille?
- Voiko käyttäjiä ryhmitellä?
 - Kuinka ryhmiä muodostetaan?
- Voinko hyödyntää palvelua muuhun, kuin tiedon jakamiseen asiakkaille?
 - Maksaako se erikseen?
 - Voiko verkkopalvelusta olla muutakin hyötyä liiketoiminnassa?
- Onko käyttäjärekisteri helppokäyttöinen, saanko sieltä haluamani tiedot?
- Käytön helppous: Jos haluan muuttaa vaikkapa etusivulla olevaa tekstiä, kuinka nopeasti se onnistuu?
 - yli 2 minuuttia tai liian monta vaihetta, kunnes muutoksia päästään tekemään ja syntyy kynnys tehdä muutosta.

Ongelmatilanteet

- Kuinka tuotetuki on järjestetty? Onko tukea? Mitä se maksaa?

Kustannukset ja elinkaari

- Kuinka paljon palvelun hankinta maksaa?
- Mitä ylläpito maksaa (esim vuosittainen kustannus) ja mitä siihen sisältyy?
- Liittyykö hankintaan muita kuluja?
- Mikä on tuotteen elinkaari?
 - Onko päivittäminen mahdollista ja mitkä ovat sen kustannukset?
 - Ulkoasu
 - järjestelmä
 - sisällöt
 - Kehittykö tuote?
 - Kuka vastaa kehityksestä?
 - Vastaako kehityssuunta tarpeitani?
- Laajennusmahdollisuudet?
- Räätelöinti asiakaskohtaisiin tarpeisiin
 - mahdollista?
 - Nopeaa vai hidasta? Miksi?
 -

Tämä ei ole kattava lista, mutta toimii toivottavasti hyvänä lähtökohtana ja ajatusten herättäjänä - ja sisältää jo kohtuullisen määrän kohtia joita vertailla eri ratkaisujen välillä.

Entäpä ideologia - Avoimen lähdekoodin ratkaisu, vai suljettu järjestelmä?

Ns. avoimen lähdekoodin sisällönhallintajärjestelmiä ovat esimerkiksi *Simple CMS*, *Drupal*, *Wordpress* ja niin edelleen. Oma järjestelmämme, *iCMS*, pohjautuu omaan tuotantomme ja edustaa tässä vertailussa suljettua järjestelmää.

Avoimen lähdekoodin järjestelmän eduksi lasketaan yleensä ainakin seuraavat asiat:

- laaja kehittäjäyhteisö
- mahdollisuus tehdä itse muutoksia alustaan
- nopea kehitys
- mahdollisuus jatkokehittää palvelua toisella yrityksellä

Moni järjestelmä on jopa tuotteistettu niin, että sen käyttöönotto onnistuu ilman syvällistä asiantuntemusta tai paneutumista. Mutta entäpä jos tuote ei sellaisenaan sovellukaan tarpeisiisi? Entä kuinka hyvin järjestelmä vastaa tarpeitasi esimerkiksi kolmen vuoden kuluttua?

Kolikon kääntöpuolia ovat nähdäkseni esimerkiksi seuraavat:

- Kukaan ei ota vastuuta tuotteesta
- Kehityssuuntaan on vaikea vaikuttaa - on tuurista kiinni, soveltuuko se tarpeisiisi
- Omat muutokset järjestelmään rikkovat helposti yhteensopivuuden, jolloin päivitettävyyden menetetään
- Tukipalvelua ei yleensä ole (poislukien vertaistuki kehittäjäfoorumien kautta)
- Järjestelmän koodi (jota siis joudutaan muokkaamaan muutoksia tehtäessä) voi olla hyvinkin hajanaista, johtuen juuri laajasta kehittäjäyhteisöstä.

Oleellista on myös **palvelun toimittajan** ideologia ja toimintamalli.

Yksi yleinen lähestymistapa on tarjota vaihtoehtoina useita ns. avoimia ratkaisuja eri tarkoituksiin. Minusta tämä tarkoittaa myös sitä, että järjestelmätoimittajalla ei ole lainkaan fokusta sen suhteen, mitä järjestelmää toimitetaan.

Prosessia ei nähdä jatkuvana kehityksenä, vaan yksittäisinä asiakasprojekteina, jotka toteutetaan ja unohdetaan sen jälkeen oman onnensa nojaan.

Innoventumin mallissa järjestelmätoimittaja on sitoutunut järjestelmän kehittämiseen - ja mikä parasta - palvelumallin ansiosta päivitykset siirtyvät automaattisesti myös asiakkaillemme. Ilman lisäkustannuksia.

Mielenkiintoinen kysymys on myös se, **kuinka avoin on avoimelle lähdekoodille rakennettu verkkopalvelu.**

Vastaus tähän on helppo: Aivan niin avoin kuin asiakkaan ja toteuttajan välinen sopimus määrittelee. Avoimuus myyntipuheessa ei välttämättä tarkoita sitä, että voisit kuitenkin käyttää muita kuin yhtä toteuttajaa palvelusi kehittämiseen, tai saisit siitä mitään - edes tuottamaasi sisältöä - mukaasi, jos haluat vaihtaa toimittajaa - Missä tällaisessa tapauksessa on avoimen lähdekoodin hyöty?

Kehitysajatuksin

Janne Leinonen

toimitusjohtaja, Innoventum Oy

Ps. iCMS:ään voi liittää ulkoisia 3. osapuolen toteuttamia osajärjestelmiä rajapinnan kautta.