

Jani Rasinkangas

AVOIMEN LÄHDEKOODIN DIGITAL ASSET MANAGEMENT –  
CASE: PORIN KAUPUNKI

Tietojenkäsittelyn koulutusohjelma  
2018

# AVOIMEN LÄHDEKOODIN DIGITAL ASSET MANAGEMENT – CASE: PORIN KAUPUNKI

Jani Rasinkangas  
Satakunnan ammattikorkeakoulu  
Tietojenkäsittelyn koulutusohjelma  
Maaliskuu 2018  
Ohjaaja: Petri Nuutinen  
Sivumäärä: 26

Asiasanat: avoin lähdekoodi, järjestelmät, tarvemäärittely

---

Tämän opinnäytetyön tarkoituksena oli DAM – järjestelmän tarvemäärittely ja järjestelmätestauksen toteutus Porin Kaupungille. Tarkoitus oli löytää potentiaalinen avoimen lähdekoodin DAM – järjestelmä käyttöönottoon.

Testaukseen valitut järjestelmät valittiin tarvemäärittelyiden ja erilaisten parhaat avoimen lähdekoodin DAM – järjestelmät listojen avulla. Testatuista järjestelmistä luotiin pisteytystaulukko, josta saatiin jonkinlainen kuva, vastaisiko mikään järjestelmistä toimeksiantajan toiveita.

Työni ei anna valmiita ratkaisuja, mutta auttaa ymmärtämään DAM – järjestelmien toimintaa. Työssäni käydään myös läpi avoimen lähdekoodin ja omisteisen ohjelmistojen hyviä ja huonoja puolia, jotka voivat vaikuttavat DAM – järjestelmän mahdolliseen valintaan.

## OPEN SOURCE DIGITAL ASSET MANAGEMENT – CASE: PORI CITY

Jani Rasinkangas

Satakunnan ammattikorkeakoulu, Satakunta University of Applied Sciences

Degree Programme in Information and Communication Technology

March 2018

Supervisor: Petri Nuutinen

Number of pages: 26

Keywords: open source, digital asset management, software, system requirements

---

The purpose of this thesis was to define the needs of the DAM system and to implement system testing for Pori City. The aim was to find the potential open source code DAM system.

The systems selected for testing were selected by the needs specification and the various listings of best open source DAM systems. From the tested systems, a scoring chart was created, from which an analysis was obtained, whether any of the systems would fill the client's wishes.

My work does not provide ready-made solutions, but it helps to understand the functions of DAM systems. My work also compares good and bad aspects of open source and proprietary software which can influence the choice of a DAM system

# SISÄLLYS

1	JOHDANTO.....	5
2	AVOIMEN LÄHDEKOODIN DIGITAL ASSET MANAGEMENT - JÄRJESTELMÄT .....	6
2.1	Avoimen lähdekoodin järjestelmät .....	6
2.2	Digital Asset Management - järjestelmät.....	7
2.3	Saatavilla olevien järjestelmäratkaisujen vertailu.....	9
2.4	Järjestelmien alustavaatimukset.....	14
2.5	Sovellettavat teknologiat.....	16
	2.5.1 EXIF	16
	2.5.2 AD / LDAP	17
3	JÄRJESTELMIEN TARVEMÄÄRITTELY .....	18
3.1	Järjestelmien vaatimukset .....	18
3.2	Järjestelmien valinta.....	19
4	JÄRJESTELMIEN TESTAUS.....	20
4.1	Testausympäristö .....	20
4.2	Järjestelmien asennus.....	20
4.3	Järjestelmien käyttö ja testaaminen .....	22
5	JÄRJESTELMIEN ARVIOINTI .....	23
6	JOHTOPÄÄTÖKSET .....	24
	LÄHTEET.....	25

# 1 JOHDANTO

Digitalisoituminen on tuonut monille yrityksille ja organisaatioille kasan digitaalista sisältöä. DAM-järjestelmillä voidaan hallinnoida tätä sisältöä keskitetysti. Kuvat, videot, logot ja muut oleelliset tiedostot ovat järjestelmän avulla uudelleenkäytettävissä esim. yrityksen markkinoinnissa, verkkokaupassa, verkkosivuilla ja eri kanavien julkaisutoiminnoissa.

Toimeksiantaja Porin kaupungin ICT-yksikön toimesta tehtiin tarvemäärittely mahdolliselle DAM-järjestelmän käyttöönotolle. Tartuin haasteeseen, vaikka itselläni ei ollut aiempaa kokemusta kyseisistä järjestelmistä. Työssäni vertaan ja testaan avoimen lähdekoodin DAM-järjestelmiä, jotka täyttäisivät toimeksiantajan toiveita.

Tavoitteena oli luoda opinnäytetyö hyvällä teoriapohjalla, jotta saataisiin hyvä käsitys siitä, mitä DAM-järjestelmät voivat tarjota. Vaikka aiheenani oli avoimen lähdekoodin DAM-järjestelmät, vertauksia tehtiin avoimen lähdekoodin ja omisteisen ohjelmiston välillä. Toinen tavoite oli löytää potentiaalinen järjestelmä, joka voitaisiin ottaa käyttöön toimeksiantajan toiveiden perusteella.

Työssä on käytetty opiskeluni ohessa tutuksi tullutta VMware:n virtuaaliympäristöä järjestelmien asennuksiin ja testauksiin. Valmiita ratkaisuja tässä työssä ei ole, mutta teoriapohjan ja järjestelmien vertailujen sekä testauksien avulla voidaan saada kuva siitä, minkälaisia ominaisuuksia DAM-järjestelmillä yleensä on. Tämä työ tarjoaa näkökulman, kannattaako organisaation ottaa käyttöön avoimen lähdekoodin vai omisteisen ohjelmiston DAM-järjestelmä.

## 2 AVOIMEN LÄHDEKOODIN DIGITAL ASSET MANAGEMENT - JÄRJESTELMÄT

### 2.1 Avoimen lähdekoodin järjestelmät

Määritteenä avoin lähdekoodi on ohjelmistoa, jossa alkuperäinen lähdekoodi on vapaasti käytettävissä ja jota voidaan jakaa ja muokata käyttäjän vaatimuksen mukaisesti. Avoimen lähdekoodin ohjelmisto on sellainen, jossa lähdekoodi on tavallisesti saatavilla muokattavaksi, parannettavaksi, uudelleenkäytettäväksi ja jaettavaksi kaikille. (Economictimes.com)

Avoimen lähdekoodin järjestelmät ovat nimensä mukaisesti avointa lähdekoodia, jota kuka tahansa voi tarkastella, muokata ja parantaa. Lähdekoodi on osa ohjelmistoa/järjestelmää, jota useimmat käyttäjät eivät ikinä tule näkemään; ohjelmoijat voivat muokata sitä muuttaakseen järjestelmän toimintoja tai millä tavoin järjestelmä toimii. Lähdekoodia muokkaamalla voidaan lisätä järjestelmään ominaisuuksia tai korjata osia, jotka eivät toimi niin kuin niiden pitäisi. (Opensource.com)

Joissain järjestelmissä lähdekoodi on patentin takana. Tätä lähdekoodia voi muokata vain se henkilö, ryhmä -tai organisaatio joka sen on luonut. Tätä voidaan kutsua myös ”closed source softwareksi”, eli omisteiseksi ohjelmistoksi. Käyttäjän pitää hyväksyä käyttöoikeuslisenssi, joka sallii ohjelmiston käyttämisen tietyin ehdoin, ilman erillistä lupaa muokata tai levittää lähdekoodia. Hyvä esimerkki tästä on Microsoft Office. (Opensource.com)

Avoin lähdekoodi eroaa omisteisesta siten, että käyttäjän pitää silti hyväksyä käyttöoikeuslisenssi, mutta lähdekoodia saa mm. muokata, kopioida, muuttaa ja levittää eteenpäin. Yleisesti ottaen avointa lähdekoodia saa käyttää mihin tarkoitukseen haluaa. Useimmat käyttöoikeuslisenssit tosin vaativat, että järjestelmää muokattaessa ja eteenpäin jaettaessa myös lähdekoodi olisi saatavilla, eikä siitä tarvitsisi maksaa lisenssimaksuja tai erillisiä kuluja. (Opensource.com)

Useimmat avoimen lähdekoodin järjestelmät käyttävät jotakin versiota GNU:n GPL:stä (GNU General Public License). Yksinkertaisesti ajateltuna GPL on kuin julkinen kuva: molempia voidaan muokata, päivittää ja käyttää uudelleen, miten halutaan. GPL antaa ohjelmoijille ja käyttäjille luvan käyttää ja muuttaa lähdekoodia. GNU lyhenne viittaa avoimeen lähdekoodiin perustuvaan GNU -nimiseen käyttöjärjestelmään, joka oli ja on vieläkin merkittävä projekti avoimen lähdekoodin tekniikoissa. Nimi GNU valittiin hakkeritraditiota seuraten itsensä sisältävänä akronyyminä lauseesta "GNU's Not Unix." (Midrack 2017, Stallman)

Yleisimmät avoimen lähdekoodin lisenssit:

- Apache License 2.0
- BSD 3-Clause "New" or "Revised" license
- BSD 2-Clause "Simplified" or "FreeBSD" license
- GNU General Public License (GPL)
- GNU Library or "Lesser" General Public License (LGPL)
- MIT license
- Mozilla Public License 2.0
- Common Development and Distribution License
- Eclipse Public License  
(Opensource.org)

## 2.2 Digital Asset Management - järjestelmät

Digital Asset Management-järjestelmällä (DAM) hallitaan, järjestetään ja julkaistaan digitaalista sisältöä keskeisestä tietokannasta. DAM-järjestelmä on yksinkertaistettuna sisällönhallintaa. Jotta voitaisiin ymmärtää, mitä DAM-järjestelmällä voidaan hallita, pitää tietää mitä digitaalinen sisältö on; valokuvia, videoita, äänitiedostoja, dokumentteja, suunnitelmia, esitelmiä jne. On myös tärkeää ymmärtää, että näillä kaikilla on digitaalinen elinkaari (Kuva 1), joka tulisi olla myös DAM-järjestelmässä jollain tavoin esillä.

(Digitalassetmanagement.com).



*Kuva 1: Digitaalisen sisällön elinkaari  
(Digitalassetmanagement.com)*

DAM-järjestelmällä voidaan kontrolloida ja hallita digitaalista sisältöä niiden elinkaaren aikana. Elinkaareen kuuluu:

1. **luonti:** Digitaalisen sisällön luonti voi lähteä jo esim. valokuvan ottohetkestä tai videopätkän suunnittelusta.
2. **hallinta:** Sisältää hyväksynnän, versionhallinnan ja logistiikan, jolla tietyt henkilöt pääsevät käsiksi tiettyihin tiedostoihin.
3. **jakelu:** Sisäiset ja ulkoiset ryhmät ovat osana jakelua. Sisäisessä jakelussa tiedosto voi kulkea osastolta osastolle tai tietyille henkilöille, esim. myyntiedustajalle. Ulkoisessa jakelussa tiedosto voidaan liittää esim. organisaation sosiaaliseen mediaan.
4. **säilytys:** Olipa sisältö tallennettuna säännölliseen käyttöön tai se on arkistoitu ja unohtettu, tieto siitä, mitä sinulla on ja voitko käyttää kyseistä sisältöä, on keskeinen osa digitaalisen sisällön elinkaarta. (Digitalassetmanagement.com).

Digitaalisuuden seurauksena tiedostojen määrä organisaatioissa on räjähtänyt. Organisaatioiden pitäisi saada tiedostot hallintaan jo ajankäytön ja brändinhallinnan laadun kannalta. DAM (Digital Asset Management) tarkoittaa suomeksi yrityksen digitaalisen sisällön hallintaa. Toimivalla DAM-järjestelmällä hallinnoidaan yrityksen digitaaliset tiedostot keskitetysti. DAM-järjestelmällä tuotteiden kuvat, logot, videot ja muu oleellinen ajantasainen materiaali ovat kaikkien niitä tarvitsevien hyödynnettävänä. Näitä aineistoja uudelleen käytetään markkinoinnissa, verkkokaupassa, verkkosivuilla ja eri kanavien julkaisutoiminnassa. (Koppatz 2017.)

DAM-järjestelmillä on yhteensä 10 eri ydinominaisuutta, jotka määrittävät voidaanko kyseistä järjestelmää edes pitää DAM-järjestelmänä:

- 1) Järjestelmällä voidaan säilöä sisältöä yksittäisinä tai massatiedostoina ja näitä tiedostoja tai niiden metadatta voidaan muokata yksittäin tai massana.
- 2) Järjestelmä turvaa sisällön joka on talletettuna; järjestelmän tietoturva pystyy määrittämään käyttöoikeuksien hallintaa sisällölle sekä määrittämään roolit käyttäjille, jotka järjestelmää käyttävät.
- 3) Järjestelmät tallentavat sisältöä sekä binaarisessa, että metatietomuodossa. Järjestelmät voivat tallentaa useita tiedostotyyppjä salliten metatietokenttien ja metatietojen muokkaamista niissä kentissä, jotka ovat liitettyinä tallennettuihin tiedostoihin.



- 4) Järjestelmä muuttaa sisältöä uusiin muotoihin, kuten pienoiskuvakkeisiin ja esikatselukuviin. Uudet sisältömuodot, jotka ovat luotu alkuperäisen sisällön muunnoksen avulla, olisi myös tallennettuna alkuperäisen tiedoston lisäksi.
- 5) Järjestelmä parantaa sisällönhallintaa metatietokenttien ja erilaisten hakujen avulla, jotka koskevat sisällön käyttöä ja uudelleenkäyttöä niiden koko elinkaaren ajan.
- 6) Järjestelmä vertailee sisältöä seuraamalla alkuperäisestä sisältöä sekä eri versioiden suhteita. Versiointi ja versionhallintatyökalut ovat keskeinen ominaisuus DAM-järjestelmässä.
- 7) Järjestelmät säätelevät jäseneltyä prosessia hallintaan, luomiseen ja tarkistamiseen työnkulun työkaluilla. Ohjelmoitujen työnkulkujen avulla DAM-järjestelmät mahdollistavat hajautetun yhteistyön keskitetyssä järjestelmässä.
- 8) Järjestelmien avulla käyttäjät voivat etsiä sisältöä ja hakea kyseisiä tiedostoja metatietojen, kokoelmien ja kulunvalvontatyökalujen avulla. Helpottamalla sellaisten tiedostojen löytämistä, jotka eivät ennen ole helposti löydettävissä, DAM – järjestelmä auttaa työntekijöitä hyödyntämään nykyistä sisältöä mahdollisimman suurella potentiaalilla.
- 9) Järjestelmissä on esikatselutoiminto, jonka avulla käyttäjät voivat tarkastella sisältöä ennen tiedostojen lataamista tai avaamista. Kun käyttäjät voivat tarkastella sisältöä nopeasti lataamatta niitä erikseen, DAM-järjestelmät vähentävät käyttäjien ylimääräistä haku-aikaa.
- 10) Järjestelmät tuottavat / julkaisevat sisältöä tarjoamalla menetelmiä, joiden avulla sisältöä voidaan jakaa, linkittää tai jakaa muulla tavoin järjestelmän ulkopuolelle. Tämä toiminto voi olla esim. URL-osoitteen generointi tiedostoon tai toiminto, jonka avulla käyttäjät voivat kerätä tiedostoja sisältäviä kokoelmia ja jakaa niitä eri työryhmiin / osastoihin.

(Widen.com, DAM Foundation Board Q4 2014)

### 2.3 Saatavilla olevien järjestelmäratkaisujen vertailu

Suuri kysymys, kun valitaan DAM-järjestelmää, on se, että hallitaanko sitä pilvestä, vai hallitaanko sitä itse. Jos ratkaisuksi valitaan itsehallinta, sinulla on täysi päätäntö-

valta, miten ja millä tavoin järjestelmää käytetään. Tosiasia on kuitenkin se, että järjestelmän ylläpito vie paljon resursseja ja voi pidemmän päältä maksaa jopa enemmän kuin pilvipalveluversio. Pilvipalveluversiossa järjestelmän hallinta ulkoistetaan, joten sinun ei tarvitse huolehtia siitä. Voi kuitenkin herätä kysymys siitä, missä yrityksen/organisaation data oikeasti sijaitsee, ja kuka niihin voi päästä käsiksi. Myös se pitää huomioida, että pilvipalveluissa voi olla mahdollisia käyttökatkoksia sekä niiden vaikutus yrityksen toimintaan. Mitä enemmän dataa on käytössä, sitä kalliimmaksi pilvipalvelu tulee. Myös järjestelmän käyttäjien määrä tulee vaikuttamaan lisenssimaksuihin. (Digitalassetmanagementguide.com).

Valittaessa DAM-järjestelmää pitää myös pohtia, ostetaanko tuki suoraan DAM-tarjoajalta, vai tehdäänkö kaikki itse. Jotkut DAM-tarjoajat tarjoavat paketteja, jossa tarjotaan sekä pilvipalvelua että tukea. Tällä tavoin yrityksen ei tarvitse kuin kirjautua järjestelmään ja käyttää sitä huoletta. Tämä ratkaisu voi maksaa muutamasta kymmenistä eurosta tuhansiin euroihin vuodessa riippuen järjestelmän ja tilan käytöstä. Jos yrityksen sisällä on paljon osaamista ja se on valmis uhraamaan resursseja järjestelmän ylläpitoon, voidaan kaikki tehdä itse asennuksesta ylläpitoon asti. Tämä toisaalta vaatii sen, että DAM-järjestelmä on avointa lähdekoodia. Yleensä lähdekoodin järjestelmään löytää DAM-tarjoajien sivuilta, tai GitHubista. Suurin osa DAM-tarjoajista tosin omistaa lähdekoodin järjestelmään, joten ainoa ratkaisu saada käyttöön juuri kyseinen järjestelmä on maksaa siitä. (DAM-tarjoajat).

Myös yhdistelmiä näistä järjestelmistä löytyy. Jotkut DAM-tarjoajat voivat tarjota esim. maksua vastaan tukea järjestelmään, jota yritys itse ylläpitää. Jotkin järjestelmien myyjät tarjoavat maksullisia lisäominaisuuksia muuten ilmaisen tuotteen päälle. Oikeaa DAM-järjestelmää valittaessa ei ole kultaista keskitietä, vaan pitää verrata mihin tarkoitukseen järjestelmää halutaan ja kuinka paljon siitä ollaan valmiita maksamaan. Oikeanlaisen järjestelmän valitseminen on vaikeaa, vääränlaisen järjestelmän valitseminen on taas kallista. (DAM-tarjoajat).

DAM-tarjoaja Brandfolderin mielestä avoimen lähdekoodin DAM-järjestelmät ovat huono vaihtoehto. Avoimen lähdekoodin DAM-järjestelmät kärsivät usein hitaasta tai viivästyneestä kehityksestä. Kun jokin järjestelmä ”kärsii” useista vapaaehtoisista ke-

hittäjästä, siitä voi seurata viivästyksiä tai hidastuksia järjestelmän kehityksen kannalta. Sillä välin kilpailijat voivat kehittää järjestelmänsä ja päästä edelle kilpailussa, ja olet taas lähtöpisteessä ennen DAM-järjestelmän mahdollista käyttöönottoa. Tämä on ongelma varsinkin, kun kaikki avoimen lähdekoodin ratkaisut eivät ole ilmaisia. Jos maksat DAM-tarjoajalle, joka ei tarjoa sinulle ratkaisua mistä hyötyisit, tarvitset uuden ratkaisun. (Prater 2017).

Avoimen lähdekoodin DAM-järjestelmät voivat kärsiä työvoiman puutteesta, jotka johtavat viivästyksiin, joihin sinulla ei ole varaa. Jos päätät käyttää avoimen lähdekoodin DAM-järjestelmää, otat tietoisesti riskin siitä, että sinulla on järjestelmä, jolla ei välttämättä ole tarpeeksi työvoimaa säilyttääkseen itse järjestelmää tai sinun sisältöä nykyisissä standardeissa. Kun mietit potentiaalista DAM-ratkaisua, pohdi, kuinka monta mahdollista kehittäjää työskentelee kyseisen järjestelmän parissa, mikä on heidän uusien ominaisuuksien ja kehityksen tilanne. (Prater 2017).

Avoimen lähdekoodin DAM-järjestelmä saattaa maksaa enemmän ja vaatia enemmän resursseja IT-osastolta, kuin valmis ratkaisu. Joissakin avoimen lähdekoodin DAM-ratkaisuissa ei ole tukea saatavilla, tai niissä saattaa olla tuki, joka vastaa liian hitaasti tukipyyntöihin. Myös yhteistyö tai järjestelmän muokkaaminen juuri sinun tarpeisiin voi olla takkuista. Jos sinun pitää palkata erikseen joku hallitsemaan DAM-järjestelmää, se on periaatteessa turhaa, jos tarkoituksesi oli säästää rahaa ja aikaa. (Prater 2017).

Avoimen lähdekoodin DAM-järjestelmän käyttö voi johtaa ”crowdsourcing”, eli joukkoistamis- teknologioihin. Jos et halua, että ratkaisusi ja teknologiasi on kaikkien saatavilla, vältä avointa lähdekoodia. Vaikka joukkoistamisella tuotettuja ominaisuuksia voidaan kehittää nopeasti, niitä pitää myös tukea ja päivittää samalla vauhdilla. Jos DAM-ratkaisussasi ei ole tarvittavia päivityksiä, tukea, tai tarvittavaa työvoimaa ylläpitää näitä muutoksia, se voi tehdä enemmän haittaa, kun hyötyä organisaatiollesi. (Prater 2017).

Avoimen lähdekoodin DAM-järjestelmät hyödyntävät laajennuksia ja lisäosia. Lisäosat ja laajennukset järjestelmiin tulevat lopulta vanhenemaan, ja voi olla hankalaa löytää ja päivittää/korjata niitä. Jos lisäosa pitää korvata jollain toisella, oikeanlaisen

ja toimivan lisäosan löytäminen voi olla erittäin hankalaa. Tämä kuluttaa organisaation arvokasta aikaa ja resursseja ja tämä tekee ratkaisustasi raskaan ylläpidettävän ongelman. (Prater 2017).

Kaikki järjestelmäratkaisut eivät ole suunniteltu toimimaan toistensa kanssa. Monet ovat omisteista lähdekoodia, jotka on suunniteltu toimimaan vain muiden saman valmistajan myymien ratkaisujen kanssa. DAM-järjestelmissä onkin erityinen avointa lähdekoodia hyödyntävien ratkaisujen puute. Valitsemalla avoimen lähdekoodin ratkaisun, voit aina luoda tarvittavat toiminnot itse, teoriassa ainakin. Järjestelmästä riippuen tarvitset osaamista esim. Javan (jsr /jcr), Apachen ja erilaisten API:en kanssa. (Zimmerer 2015).

Valitsemalla avoimen lähdekoodin ratkaisun, sinulla on vapautta sekä joustavuutta valita paras ratkaisu ja kasata omat mukautetut järjestelmän osat, jotka tekevät kaiken mitä tarvitset: jaat helposti tarvitsemasi tiedot ja sisällön, jotta voit toimittaa johdonmukaista ja yksilöllistä toimintaa organisaation osastojen välillä sekä ulkopuolelle. Avoimen lähdekoodin ratkaisut yleensä omaksuvat tämänkaltaisen avoimen filosofian. (Zimmerer 2015).

Avoimen lähdekoodin DAM-järjestelmiä voi olla hankala löytää, mutta niitä on silti olemassa. Jotkin järjestelmät silti tukevat avointa lähdekoodia, vaikka muuten olisivatkin omisteista lähdekoodia. Jos kuitenkin etsit varta vasten avoimen lähdekoodin ratkaisua, keskity DAM- järjestelmiin, jotka tukevat avoimia standardeja, kuten CMIS, JSR, JCR, Apache yms. (Zimmerer 2015).

Yksi tärkeä ominaisuus DAM- järjestelmän valintaan on yksinkertaisuus; järjestelmän pitäisi tarjota käyttäjilleen käyttöliittymän, mitä on miellyttävä käyttää. Vaikka järjestelmä olisi kuinka kehittynyt ja kattava, sen käyttäjät eivät välttämättä ole tekniikan ammattilaisia. Heitä eivät kiinnosta järjestelmän ominaisuuksien toimintaperiaatteet, vaan se, että järjestelmää olisi helppo ja selkeä käyttää. (Cotton).

Tosiasia on, että kaikki DAM-järjestelmät eivät ole samanlaisia. Jotkut järjestelmät sopivat tiettyihin organisaatioihin, kun taas toiset eivät. Siksi oman organisaation tar-

peiden ymmärtäminen on ensisijaisen tärkeää, että löytää ratkaisun, joka toimii kaikille osapuolille. On suositeltavaa luoda asiakirja, jossa tulee ilmi järjestelmän vaatimukset. Millaista toimintaa organisaatiosi harjoittaa? Millä alalla organisaatiosi on? Teetkö yhteistyötä ulkopuolisten kumppaneiden kanssa, jotka tarvitsevat pääsyn järjestelmään? Onko järjestelmän käyttäjillä useita kielivaatimuksia? Näiden kysymysten jälkeen tulisi ottaa selvää tarkemmat tekniset tiedot kuten:

- Sisällön tyypit: videot, valokuvat, dokumentit, äänitykset, muu media
- Sisällön koko: kuinka paljon tiedostoja tallennetaan järjestelmään mm. Videoiden pituus ja valokuvien resoluutio vaikuttavat.
- Tiedostotyyppit: MP4, MPEG, MPG, MOV, JPEG, GIF, PNG, TIFF, MPS, WAV, M4A, AIFF, CSV, XLS, AI, EPS, PDF, yms.
- Hinnoittelu: Ostetaanko järjestelmä DAM-tarjoajalta. käytetäänkö avoimen lähdekoodin järjestelmää. Ostetaanko järjestelmän lisäksi tuki ja asennus. Asennetaanko järjestelmä organisaation omiin tiloihin vai hommataanko SaaS – ratkaisu. Palkataanko järjestelmälle erikseen ylläpitäjä vai ulkoistetaanko ylläpito kokonaan.
- Erikoistoiminnot: Mitä erikoistoimintoja järjestelmässä pitää olla juuri sinun organisaatiolle tärkeissä tehtävissä. Esim. kansiodien ja kokoelmien hallinta, käyttäjähallinta, erityiset hakuominaisuudet tai kuvien esikatselutila. Joillekin organisaatioille Some – integraatio voi olla keskeisin osa järjestelmää, kun taas toiselle organisaatiolle sille ei löydy käyttöä ollenkaan.
- Implementaatio: Kuinka paljon ulkopuolista apua tarvitset järjestelmän käyttöönotossa. Millä ja mille aikavälille tulet ottamaan järjestelmän käyttöön.
- Tuki: Millainen tuki järjestelmälle hankitaan; 24/7, henkilökohtaista vai puhelintukea, Online-chat tuki. Hankitaanko tukea ulkopuolelta ollenkaan ja turvaudutaan vain oman talon väkeen.
- Budjetti: Kuinka paljon olet valmis maksamaan kokonaisuudesta.  
(Digitalassetmanagement.com)

DAM – järjestelmätarjoaja MediaValetin mukaan helposti saatavilla ja organisoidulla digitaalisella sisällöllä on avainasema hyvin toimivassa yrityksessä. Jos digitaalista sisältöä **ei hallita** millään tavoin;

- Työntekijät tuhlaavat jopa yhden työpäivän viikossa sisällön hallintaan, siirtämiseen ja julkaisemiseen.
- Työntekijät löytävät haetun tiedoston vain 35% ajastaan.
- 65% myyntiedustajista eivät löydä oikeata sisältöä tarvittaessa.
- Yritykset korvaavat digitaalista sisältöä vuosittain 300 miljardin \$ edestä turhaan.

(Mediavalet 2016).

## 2.4 Järjestelmien alustavaatimukset

Useimmat DAM-järjestelmät tarvitsevat allensa tietyt osat toimiakseen: Ensimmäinen palvelimen, mistä kyseistä järjestelmää tuotetaan, käyttöjärjestelmän, webpalvelimen, tietokannat ja ohjelmistoalustan. (DAM-tarjoajat).

Koska DAM-järjestelmillä hallitaan digitaalista sisältöä, se vaatii myös jonkin verran tehoja raudalta, millä sitä pyöritetään. Varsinkin kuvien ja videoiden lataaminen palvelimelle, mahdollinen muokkaaminen ja julkaisu vaativat keskusmuistia ja suoritusnopeutta. Joissain järjestelmissä on myös mahdollista renderoida uusia versioita sisällöstä, esim. luoda digitaalinen kuva eri resoluutiolla. Usein esiintyvät luvut suosituksista olivat 3GHz suoritusnopeutta ja 8GB keskusmuistia. Suurin osa järjestelmäntarjoajista kehottaa kuitenkin lisäämään keskusmuistia mahdollisimman paljon (16/32GB) toimintojen nopeuttamiseksi. Jos samalla palvelimella pyöritetään sekä itse DAM-järjestelmää, että tietokantaa mihin sisältö tallennetaan, tulisi levytilaa varata sen mukaan. (DAM-tarjoajat).

DAM-järjestelmät, joita päivitetään säännöllisesti suosivat myös käyttöjärjestelmiä, jotka ovat ajan tasalla. Windowsin viimeisimmät versiot (Microsoft Windows 7, 10, Windows Server 2012 R2, 2016) ovat yleisimmät tuetut käyttöjärjestelmät Windows-käyttöliittymistä. Linux ja UNIX-pohjaiset käyttöliittymät ovat myös usein, ellei aina tuettu. Linux käyttöjärjestelmistä useimmiten esiintyi Red Hat/CentOS, Debian, Ubuntu ja openSUSE. UNIX-pohjaisista varsinkin BSD-tyyliset (FreeBSD,

OpenBSD, NetBSD) käyttöjärjestelmät olivat usein tuettu. Jotkin DAM – järjestelmät tukivat myös Mac OS:ää (DAM-tarjoajat).

Suurin osa DAM-järjestelmistä on hallittavissa tai käytettävissä verkkoselaimen kautta, joten sitä varten tarvitaan webpalvelin. Ylivoimaisesti eniten tuettu, sekä käytetty muuallakin maailmalla on avointa lähdekoodia käyttävä Apache http-palvelin. Myös Microsoftin IIS (Internet Information Services) esiintyi tuetuissa webpalvelimissä muutamana kerran, varsinkin jos tuetuissa käyttöjärjestelmissä oli jokin Windowsin käyttöjärjestelmistä. (DAM-tarjoajat).

Tietokannat ovat tärkeä osa toimivaa DAM-järjestelmää. Kaikki sisältö, mitä käsitellään järjestelmän kautta, on tallennettava johonkin tietokantaan. DAM-järjestelmät tukevat useimmiten MySQL, MS SQL tai H2 tietokantoja. MySQL on suosituin ja tuetuin näistä. MS SQL esiintyi taas niissä järjestelmien tukilistoissa, joissa Microsoftin käyttöjärjestelmiä myös tuettiin. H2 tietokanta on Java pohjainen, jota jotkut DAM-järjestelmäntarjoajat tukevat. Se ei ole kuitenkaan yhtä vakaa tai turvallinen kuin MySQL, joten siitä varoitettiin, että sitä ei kannata käyttää tuotannon yhteydessä. (DAM-tarjoajat).

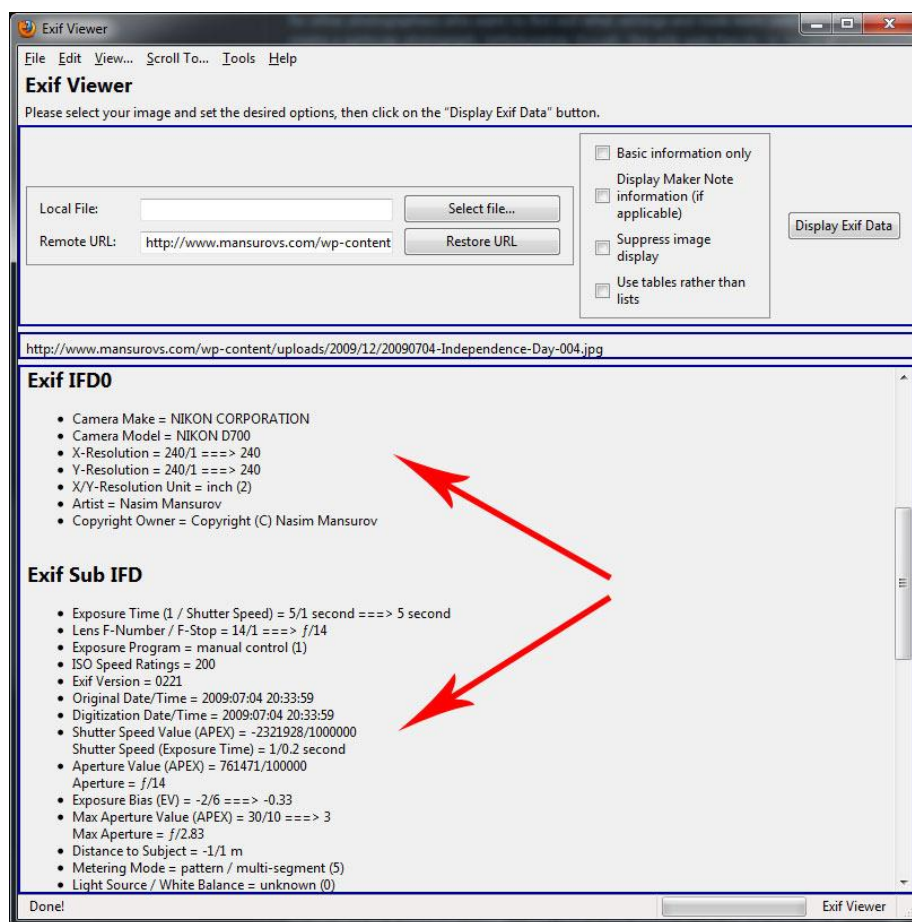
Suurinta osaa DAM-järjestelmistä hallitaan verkkoselaimen kautta. Ohjelmistoalustoina yleisimmät kielet olivat Java ja PHP. Muutamissa järjestelmissä kaikki muutokset tehtiin Java tai PHP lähdekoodia muokaten, joka on melko vaikeata keskiverto-käyttäjälle, kun taas useimmissa järjestelmissä ominaisuuksia pystyi muokkaamaan selaimen kautta graafisella käyttöliittymällä. (DAM-tarjoajat).

## 2.5 Sovellettavat teknologiat

### 2.5.1 EXIF

Digitaalisen sisällön hallinnassa keskeinen asia on tiedostojen metadata. Mitä enemmän sisällöllä on eri tietueita, sitä tarkemmin sitä voidaan hakea tietokannasta ja sitä nopeammin kyseinen sisältö saadaan käyttöön.

Valokuvista ja videoista metadataa luettaessa käytetään EXIF-tietuetta (Exchangeable Image File Format). EXIF datalla voi saada selville mm. kameran valmistajan, mallin, valotusajan, aikaleimat, paikkatiedot, kuvaajan tiedot, kuvan selityksen ja tekijänoikeustiedot. Vaikka nämä tiedot ovat tarkoitettu enemmänkin valokuvaajille, näiden avulla voidaan hakea helposti tiettyjä valokuvia tietokannoista. EXIF tiedot lisäävät tiedostoon hieman kokoa (n. 3-5% (macricksandtips.com, www-lähde.)), joten jos tietokannassa on esim. 10 000 kuvaa, ja jokaisessa on EXIF-tietue (Kuva 2), se vaatii enemmän levytilaa. (Manjurov 2017, Powell 2013).



Kuva 2: Esimerkki EXIF - tietueesta (photographylife.com)



## 2.5.2 AD / LDAP

Jos DAM-järjestelmällä on useita eri käyttäjiä, ja näillä eri käyttäjillä pitäisi olla erilaisia oikeuksia digitaalisen sisällön hallintaan, niin käyttäjiä pitäisi hallita jollain tavoin. Useimmissa, ellei kaikissa DAM-järjestelmissä on järjestelmän sisäinen käyttäjänhallinta työkalu. Sillä voidaan lisätä ja hallinnoida käyttäjiä ja heidän pääsyä eri alueille järjestelmässä. Entäpä jos organisaatiolla on jo käyttäjätietokanta eikä haluta lisätä lisää tunnuksia ja salasanoja muistettavaksi käyttäjille? Suurimassa osassa DAM-järjestelmissä on olemassa AD/LDAP-integraatio (DAM-tarjoajat). Tällä integraatiolla voidaan liittää olemassa oleva käyttäjähakemisto kyseiseen DAM-järjestelmään, jolloin käyttäjä kirjautuu esim. samoilla tunnuksilla työkoneelleen, että DAM-järjestelmään.

Useimmissa organisaatioissa ja yrityksissä käytetäänkin lyhennettyä termiä ”AD”, eli Active Directory. Se on Microsoftin kehittämä käyttäjähakemisto-ohjelmisto. Toisaalta se toimii muidenkin, kun Microsoftin kehittämien tietojärjestelmien kanssa ja AD onkin kehittynyt yleistermiksi, kun puhutaan käyttäjähakemistoista. AD:sta puhuttaessa yleensä tarkoitetaan yrityksen sisäverkossa olevaa palvelinta, joka varastoi yrityksen henkilökunnan käyttäjätunnukset ja monia muita perustietoja. Tähän palvelimeen organisaation eri tietojärjestelmät sitten ottavat yhteyttä, kun halutaan selvittää, onko käyttäjällä X oikeus päästä käyttämään järjestelmää vai ei. (Tolvanen 2011)

LDAP (Lightweight directory access protocol) ei varsinaisesti ole ohjelmisto, vaan protokolla, joka määrittelee tietyt palvelut, ja eri ohjelmistotoimittajat hoitavat itse hakemiston toteutuksen parhaaksi näkemällään tavallaan. Esim. Microsoftin Active Directory perustuu LDAP-määrittelyyn, koska Microsoft on halunnut AD:n olevan mahdollisimman yhteensopiva järjestelmien kanssa, jotka tukevat LDAP:ia. Monille DAM-järjestelmille onkin varsin samantekevää, millainen käyttäjähakemisto organisaatiolla / yrityksellä on, koska nykypäiväiset järjestelmät osaavat hyödyntää suoraan useiden eri käyttäjähakemistojen tietoja LDAP-protokollan avulla. (Tolvanen 2011)

### 3 JÄRJESTELMIEN TARVEMÄÄRITTELY

Tarvemäärittelyn taustana oli, että Porin kaupungin ICT-yksikkö halusi koekäyttää avoimen lähdekoodin DAM-järjestelmiä. Koekäytön tarkoituksena oli selvittää, vastaavatko tarjolla olevat järjestelmät käyttötarvetta. Porin kaupungilla on käytössä useita erilaisia kuvapankkisovelluksia, joten koekäytöllä tutkitaan, voidaanko kuvapankkisovellukset korvata DAM-järjestelmällä. Tarvemäärittelyt laadittiin Porin ICT-yksikön ICT-asiantuntijan Mikko Viitapohjan toimesta.

#### 3.1 Järjestelmien vaatimukset

DAM-järjestelmää tulnaisiin ensisijaisesti käyttämään valokuvien arkistointiin ja jakamiseen. Myös muita tallennemuotoja, kuten videoita olisi hyvä saada tallennettua järjestelmään. Järjestelmän käytön tulisi tapahtua selaimella siten, että se ei vaadi asiakasohjelmiston asennusta käyttäjän työasemaan tai päätelaitteeseen.

Järjestelmän tulee tukea seuraavia toimintoja:

- Digitaalisen kuvan tallennus järjestelmään
- Kuvan tagit
- Kuvan EXIF-tietue
- Kuvaajan tiedot
- Kuvauspaikka
- Kuvausaika
- Kuvaan liittyvät tiedot (vapaatietokenttä)
- Kuvan käyttöaika
- Kuvan käyttöoikeudet
- Kuvan poistaminen
- Useiden kuvien yhtäaikainen lähetys järjestelmään

- Kuvan tallentajan tiedot
- Kuvien hakeminen
- Kuvan julkaisu Internetissä
- Usean kuvan ryhmittely ja jakaminen (albumi/kansio)
- AD-integraatio

Järjestelmän toissijaiset toiminnot:

- DAM-järjestelmään tallennettujen tietojen vienti kuvan EXIF – tietueeseen, tai päinvastoin
- Liityntä someen ja pilvipalveluihin (Twitter, Flickr, Facebook, Instagram, OneDrive)
- Julkaistavan kuvan resoluution muuttaminen
- Karttaintegraatio kuvien paikkatiedoille, jos GPS sijainti tunnetaan(GPS-tagitus)

### 3.2 Järjestelmien valinta

Järjestelmien valintaan vaikutti haluttujen ominaisuuksien lisäksi se, että kyseisissä järjestelmissä kehitystyö on ollut aktiivista ja järjestelmää päivitettäisiin tulevaisuudessaakin. Koska DAM-järjestelmiä on lukuisia erilaisia, testattavaksi otettiin viisi parhaiten soveltuvaa järjestelmää.

Valinta tehtiin vaatimusmäärittelyiden, google hakujen ja erilaisten parhaat DAM-järjestelmät listojen avulla. Järjestelmät, jotka esiintyivät useasti eri parhaat DAM-järjestelmät listauksissa, otettiin tarkempaan tarkasteluun. Järjestelmiä myös tarkasteltiin yhdessä Porin ICT-asiantuntijan kanssa eri palaverissa, joissa otettiin myös huomioon hänen näkökulma järjestelmien toiminnoista ja ominaisuuksista. Loppujen lopuksi järjestelmät jotka otettiin testattavaksi, olivat: **Nuxeo, ResourceSpace, Razuna, Alfresco ja Phraseanet**. Nämä kaikki järjestelmät olivat saaneet hyvät pisteetykset

erilaisissa listauksissa, ja sisälsivät suurimman osan järjestelmien halutuista ominaisuuksista. Näihin järjestelmiin oltiin myös tehty aktiivisesti kehitystä ja päivityksiä viimeisen vuoden aikana.

## 4 JÄRJESTELMIEN TESTAUS

### 4.1 Testausympäristö

Testausympäristönä käytettiin SAMK:n boksi virtuaaliympäristöä; VMwaren Vsphe-rea, jota on opintojen yhteydessä käytetty jo ennestään. Järjestelmät asennettiin virtuaalikoneille, jotka olivat tarvittaessa yhteydessä jo ennestään rakennettuun AD – käyttäjähakemistoon. Tällä tavoin pystyttäisiin jollain tavoin simuloimaan mahdollista työympäristöä, missä järjestelmää käytettäisiin.

### 4.2 Järjestelmien asennus

Järjestelmät asennettiin Linux (CentOS 7, 64bit) palvelimille. Levytilaa varattiin 40GB ja muistia 8GB. Myös graafinen käyttöliittymä (Gnome GUI) asennettiin erikseen, jotta voitaisiin testata järjestelmien ominaisuuksia paikallisesti selaimella. Järjestelmäkohtaisesti palvelimille tarvittaessa rakennettiin myös MySQL-tietokannat ja Apache webpalvelimet. Järjestelmien lähdekoodi saatiin DAM-tarjoajien kotisivuilta tai GitHubista.

Muihin järjestelmiin verrattuna, Nuxeo vaati poikkeuksellisesti kaksiydin – prosessorin. Myös Java JDK 1.8.0 vaadittiin palvelimelle. Erillistä webpalvelinta ei tarvinnut pystyttää, koska Nuxeossa on sisäänrakennettu Apache Tomcat. Tietokantaakaan ei tarvinnut testikäyttöön erikseen pystyttää koska Nuxeossa on sisäänrakennettu H2 tietokanta. Jotta jotkut ominaisuudet toimisivat Nuxeossa, seuraavat kolmannen osapuolen lisäosat piti asentaa palvelimelle: Docker, Exiftools, FFmpeg, Ghostscript, ImageMagick, LibreOffice ja pdftohmtl.

ResourceSpace vaati LAMP (Linux, Apache, MySQL, PHP) –alustan toimiakseen. ResourceSpace vaati myös seuraavat kolmannen osapuolen lisäosat toimiakseen: antiword, ExifTools, FFmpeg, Ghostscript, ImageMagick ja poppler.

Razuna vaati pelkästään Java JDK 1.8.0 toimiakseen. Razunassa on sisäänrakennetut H2 tietokannat ja Apache Tomcat, joten erillisiä tietokantoja tai webpalvelimia ei tarvittu. Tosin H2 tietokantaa ei suositella, kuin testikäyttöön, joten MySQL – tietokanta olisi hyvä olla tuotannossa. Seuraavat kolmannen osapuolen lisäosat asennettiin, jotta Razunan ominaisuudet toimisivat: DCraw, ExifTools, FFmpeg, Ghostscript, ImageMagick, MP4Box. Näistä DCraw ja MP4box olivat vapaaehtoisia lisäosia, joita tarvittiin vain, jos järjestelmällä käsiteltäisiin digitaalikameroiden RAW-tiedostoja tai videoiden suoratoistoa.

Alfresco erosi muista järjestelmistä siten, että ilmaisversiossa joitain toimintoja ei tuettu, mitkä taas toimivat Alfrescon maksullisessa versiossa. Toimiakseen Alfresco vaati myös LAMP-alustan. Kolmannen osapuolen lisäosista vaadittiin Ghostscript, ImageMagick ja LibreOffice. Alfrescolla on myös oma lisäosa – sivusto: Alfresco addons, josta voi löytää haluttuja ominaisuuksia järjestelmään. Vaikka suurin osa lisäosista ovat avointa lähdekoodia, joistakin pitää maksaa erillinen lisenssimaksu, jotta sinulla on oikeus käyttää niitä. Myös lisäosien asennus vaati ylimääräistä säätöä: Jotta lisäosia voitaisiin asentaa järjestelmään, vaadittiin Alfresco WAR: se on JAR- tiedosto, jota käytetään jakamaan tiedostoja (Java Server-sivut, Java-servletit, Java-luokat, XML-tiedostot, tunniste - kirjastot ja staattiset verkkosivut), jotka yhdessä muodostavat verkkosovelluksen. Haluttu lisäosa ladataan ja puretaan oikeaan hakemistoon, sen jälkeen Alfrescon oma AMP (Alfresco Module Package) suorittaa kyseisen lisäosan asennuksen.

This command installs the files found in the AMP into the Alfresco WAR. If the module represented by the AMP is already installed and the installing AMP is of a higher release version, then the files for the older version are removed from the WAR and replaced with the newer files.

The following commands show examples of how to install the `example-amp.amp`, and assumes that the AMP file is in the same directory as the WAR file:

```
java -jar alfresco-mmt.jar install example-amp.amp alfresco.war -preview
```

Review the modification to check the changes that will update any existing files.

The following example will install the AMP file:

```
java -jar alfresco-mmt.jar install example-amp.amp alfresco.war -verbose
```

The modified Alfresco WAR can then be redeployed back into your application server.

On restarting the application server, the console will show that the custom class was initialized during startup.

Verify that the AMP is installed using the MMT list command. For example:

```
java -jar alfresco-mmt.jar list <WARfileLocation>
```

This command provides a detailed listing of all the modules currently installed in the WAR file specified.

*Kuva 3: esimerkki Alfresco-lisäosan asennuksesta*

Phraseanet vaatii myös toimiakseen LAMP – alustan. Phraseanet erosi muista järjestelmistä siten, että se vaati kaksi tietokantaa. Lisäksi seuraavat kolmannen osapuolen lisäosat vaadittiin: FFmpeg, Ghostscript, ImageMagick, MP4Box, RabbitMQ, SWFTools, Unoconv ja XPDF.

### 4.3 Järjestelmien käyttö ja testaaminen

Järjestelmien asennusten jälkeen ne otettiin testikäyttöön paikallisesti: järjestelmää hallitaan ja käytetään samalta palvelimelta. Järjestelmiä hallittiin selaimesta, joten otettiin yhteys localhostiin (esim. <http://localhost/razuna>). Ensiksi luotiin järjestelmän pääkäyttäjän tunnukset ja testikäyttäjät. Tämän jälkeen alettiin testaamaan vaatimusmäärittelyn mukaisesti haluttuja toimintoja. Kansioiden ja hakemistojen luonti sekä eri käyttäjien pääsyoikeudet niihin. Kuvan tai kuvien lataaminen järjestelmään sekä niiden poistaminen järjestelmästä. Kuvien EXIF-tietueiden toimivuus. Kuvien haku eri parametreilla (EXIF, tagit, otsikko yms.) järjestelmästä. Kuvien ominaisuudet ja muokattavuus. AD – integraation toiminta. Testauksien perusteella eri ominaisuudet pisteytettiin ja niitä vertailtiin toisiinsa. Järjestelmiä myös esiteltiin Porin ICT -yksikön ICT –asiantuntijalle palavereissa, jotta saatiin myös hänen mielipide järjestelmien ominaisuuksien toiminnoista.

## 5 JÄRJESTELMIEN ARVIOINTI

Järjestelmien käytön ja testauksen jälkeen laadittiin taulukko, jossa eri ominaisuuksia pisteytettiin 0-100 pisteen välillä. Pisteytykseen vaikutti ominaisuuden toimivuus, helppokäyttöisyys ja ylipäättänsä ominaisuuden olemassaolo järjestelmässä.

Pisteytys on tehty täysin minun näkökulmasta verrattuna tarvemäärittelyn ominaisuuksiin. Pisteytyksestä voi nähdä, mikä järjestelmistä voisi olla potentiaalinen vaihtoehto Porin kaupungille ottaa käyttöön.

	Nuxeo	ResourceSpace	Razuna	Alfresco	Phraseanet
Digitaalisen kuvan tallennus järjestelmään	100	100	100	100	100
Kuvan tagitus	100	100	100	100	100
Kuvan EXIF-tietue	50	50	70	100	30
Kuvaajan tiedot	0	50	70	100	0
Kuvauspaikka	0	100	0	50	0
Kuvausaika	0	100	100	100	0
Kuvaan liittyvät tiedot	0	70	100	100	100
Kuvan käyttöaika	0	0	50	50	0
Kuvan poistaminen	100	100	100	100	100
Useiden kuvien yhtäaikainen lähetys järjestelmään	50	100	100	100	100
Kuvan tallentajan tiedot	50	50	100	50	50
Kuvien hakeminen	20	100	100	50	50
Kuvan julkaisu Internetissä	0	30	50	20	0
Usean kuvan ryhmittely ja jakaminen (albumi/kansio)	30	0	100	50	10
AD-integraatio	50	50	100	50	50
tietojen vienti kuvan EXIF-tietueeseen tai päinvastoin	0	0	50	70	0
SOME-integraatio	0	70	20	20	70
Pilvipalvelu	0	0	30	10	0
Kuvan resoluution muuttaminen	20	100	100	100	70
GPS-tagit	0	50	0	0	0
Mobiiliversio	50	0	0	50	0
Helppokäyttöisyys	20	40	100	50	50
Käyttöliittymä	20	20	100	90	30
	640	1260	1540	1420	880

Kuva 4: DAM - järjestelmien tarvemäärittely pisteytystaulukko

## 6 JOHTOPÄÄTÖKSET

Vaikka pisteytyksen perusteella voisi päätellä, että Razuna olisi paras vaihtoehto, pitää ottaa huomioon mm. järjestelmän käyttöliittymän helppokäyttöisyys. Esimerkiksi Nu-xeo vaikutti hyvältä järjestelmältä paperilla, mutta sen käyttöliittymä oli todella sekava ja hankala navigoida. ResourceSpace oli myös hyvä järjestelmä, mutta sen kansio/albumirakenne oli todella sekava ja hankala käyttää. Alfresco olisi ollut myös hyvä järjestelmä, mutta lisäosien kanssa kikkailu ja järjestelmän joidenkin ominaisuuksien tuki maksumuurin takana olivat huono ominaisuus järjestelmässä. Phraseanet oli todella pelkistetty DAM-järjestelmä, siinä oli paljon eri ominaisuuksia, mutta niiden toiminta oli puutteellista.

Pisteytyksen perusteella Razuna oli voittaja näistä järjestelmistä ja sen helppokäyttöisyys, sekä selkeä käyttöliittymä puhuvat puolestaan. Toisaalta Razunan dokumentointi oli puutteellista ja jokseenkin vanhentunutta, vaikka järjestelmää oltiin aktiivisesti päivitetty.

Kaiken kaikkiaan oikeaa avoimen lähdekoodin järjestelmää tiettyyn tarkoitukseen on omasta mielestäni erittäin vaikea löytää ja vaikka oikeanlainen järjestelmä löytyisi, sen ylläpito veisi paljon aikaa ja resursseja. Omasta mielestäni vain Razuna täytti suurimman osan toimeksiantajan toiveita, mutta ei läheskään kaikkia. Testatut ilmaiset avoimen lähdekoodin järjestelmät eivät mielestäni täyttäneet toimeksiantajan toiveita tarpeita tarpeeksi, koska ne olivat käytettävyydeltään vajavaisia. DAM – tarjoajat eivät kovin innokkaasti mainostaneet heidän avoimen lähdekoodin versiota, vaan kauppassivat suoraan jonkinlaista maksullista pakettia, mikä on luonnollista bisneksen näkökulmasta. Itse ottaisin tällaisen paketin, jos järjestelmä, joka täyttäisi tarpeeni löytyisi.



## LÄHTEET

Cotton A. DAM Simple: how to select the right Digital Asset Management system if you are a marketer. Viitattu 17.1.2018. <http://digitalassetmanagementnews.org/features/dam-simple-how-to-select-the-right-digital-asset-management-system-if-you-are-marketer/>

Dammaturitymodel.org. List of accredited vendors with 10 core certifications. Viitattu 14.1.2018. <http://dammaturitymodel.org/10-core-vendors/>

Digitalassetmanagement.com. DAM Basics Viitattu 14.1.2018. <http://digitalassetmanagement.com/dam-basics/>

Digitalassetmanagementguide.com. Self-Hosted vs Cloud, Viitattu 17.1.2018. <http://www.digitalassetmanagementguide.com.au/self-hosted-vs-cloud/>

Economicstimes.com. Definition of open source. Viitattu 26.2.2018. <https://economic-times.indiatimes.com/definition/open-source>

Gehin, N. 2017. Know your DAM requirements. Viitattu 24.1.2018. <http://digitalassetmanagement.com/blog/know-your-dam-requirements/>

Koppatz, R. 2017. Mikä on DAM ja miksi sitä tarvitaan? Viitattu 26.2.2018. <http://blogi.communicationpro.com/artikkelit/mika-on-dam-ja-miksi-sita-tarvitaan>

Mansurov, N. 2017. What is EXIF Data? Viitattu 21.1.2018. <https://photographylife.com/what-is-exif-data>

Mediavalet 2016. Why DAM is in every marketer's forecast [Infographic] 1.7.2016. Viitattu 17.1.2018. <https://www.mediavalet.com/dam-every-marketers-forecast-infographic/>

Midrack, RL. 2017. How users benefit from OSS. Viitattu 26.2.2018. <https://www.lifewire.com/what-is-open-source-software-4147547>

Opensource.com. What is open source software? Viitattu 19.1.2018. <https://opensource.com/resources/what-open-source>

Opensource.org. Popular Licenses. Viitattu 13.1.2018. <https://opensource.org/licenses>

Powell, J. 2013. Clean up image and save space by. Viitattu 23.1.2018. <http://www.mactricksandtips.com/2013/07/clean-up-images-and-save-space-by-removing-exif-data.html>

Prater, M. 2017. Why open source software is bad. Viitattu 29.1.2018. <https://blog.brandfolder.com/blog/why-open-source-software-is-bad/>

Stallman, R. A stark moral choice. Viitattu 26.2.2018. <http://www.gnu.org/gnu/thegnuproject.html>

Tolvanen, P. 2011. Käsitteet ojennukseen: Active Directory(AD), LDAP, SSO ja identiteetin hallinta. Viitattu 23.1.2018 & 24.1.2018. <https://intranet-ostaja-nopas.fi/2011/04/29/kasitteet-ojennukseen-active-directory-ad-ldap-sso-ja-identiteetin-hallinta/>

Widen.com. Widen's response to the 10 characteristics of a Digital Asset Management System. Viitattu 15.1.2018. <https://www.widen.com/blog/widens-response-to-the-10-characteristics-of-a-digital-asset-management-system>

Zimmerer, J. 2015. Why we like open source digital asset management solutions. Viitattu 31.1.2018. <http://www.topdownsystems.com/resources/blog/why-we-like-open-source-digital-asset-management-solutions>

DAM – tarjoajat:

<https://www.alfresco.com>

<https://www.bynder.com/en/>

<https://www.capterra.com/digital-asset-management-software/>

<https://www.cmscritic.com/dir/dam>

<http://damvendors.com/>

<https://www.digikam.org/>

<https://entermediadb.org/>

<http://www.fedora-commons.org/features>

[https://www.getapp.com/marketing-software/digital-asset-management/?pricingcategories\\_slug%5B%5D=open-source&sort=recommended](https://www.getapp.com/marketing-software/digital-asset-management/?pricingcategories_slug%5B%5D=open-source&sort=recommended)

<https://www.nuxeo.com/>

<http://www.open-source-guide.com/en/Solutions/Applications/Dam>

<http://www.opensourcedigitalassetmanagement.org/>

<https://www.phraseanet.com/en/>

<https://www.quora.com/Digital-Asset-Management-Free-Opensource-DAMS-for-small-school-setting>

<https://www.razuna.com/>

<https://www.resourcespace.com/>