Osaamista
ja oivallusta
tulevaisuuden
tekemiseen

Micke Haapasaari

Datan visualisointi Elastic Cloud -palvelussa

Metropolia Ammattikorkeakoulu

Insinööri (AMK)

Tietotekniikan koulutusohjelma

Insinöörityö

2.1.2019

Tekijä Otsikko Sivumäärä Aika	Micke Haapasaari Datan visualisointi Elastic Cloud -palvelussa 41 sivua 2.1.2019
Tutkinto	Insinööri (AMK)
Tutkinto-ohjelma	Tietotekniikan koulutusohjelma
Ammatillinen pääaine	Tietojenkäsittely ja tietoliikenne
Ohjaajat	Ohjaava opettaja Janne Salonen

Insinöörityön tarkoituksena oli tutustua Elastic Cloud -julkaisun asentamiseen pilvipalvelimelle. Insinöörityö keskittyy helppoon ja nopeaan ohjelmistopalveluna tietokannan asentamiseen valmiiksi määritellyillä asetuksilla. Elastic Cloud -julkaisun asentamisen jälkeen insinöörityössä lähetetään dataa Elastic Cloudiin kuuluvaan Elasticsearchiin, mitä tutkitaan ja visualisoidaan Kibanan avulla.

Insinöörityössä saatiin rakennettua toimiva Elastic Cloud -julkaisu, mihin lähetettiin onnistuneesti dataa Python-kielellä. Elastic Cloud -julkaisu asennettiin 14 päivän ilmaisella kokeilujaksolla Elastic Cloud -sivulla. Kun käyttäjätunnukset Elastic Cloudiin oli saatu, aloitettiin Elastic Cloud -julkaisun asentaminen.

Insinöörityössä käydään kuvia hyväksikäyttäen läpi, mitä kaikkia määrittelymahdollisuuksia Elastic Cloud -julkaisuun voi tehdä. Tämän jälkeen insinöörityössä lähetettiin JSON dataa Elasticsearchiin. Lopuksi dataan tutustutaan ja sitä visualisoidaan Kibanan avulla.

Insinöörityön lopputuloksena on toimiva Elastic Cloud -julkaisu ja yleisnäkymä erilaisista Kibanalla tehdyistä visualisoinneista Elasticsearchiin lähetetystä datasta.

Johtopäätöksenä havaittiin, että Elastic Cloud -julkaisun asentaminen ja hoito oli helppoa. Elastic Cloud on varteenotettava tapa saada nopeasti hyvin konfiguroitu tietokanta asennettua. Elastic Cloudissa on hyvin otettu huomioon tietoturva-aspekteja sekä parhaiden asetusten määrittelyjä.

Avainsanat	Ohjelmistopalveluna, Elastic, Kibana

Author	Micke Haapasaari
Title	Data visualization in Elastic Cloud
Number of Pages	41 pages
Date	2 January 2019
Degree	Bachelor of Engineering
Degree Programme	Information Technology
Professional Major	Data processing and telecommunications
Instructors	Janne Salonen, Principal Lecturer

Purpose of this thesis was to create Elastic Cloud deployment in software as a service in-cloud environment. Main focus of this thesis was to create fast and easy deployment of database as a service with readymade configurations. Other focus of this thesis was to send data to Elasticsearch and visualize this data with Kibana.

In this thesis Elastic Cloud deployment was successfully created. Also, data sending and data visualization was a success. Elastic cloud was deployed with a 14-day free trial period offered by Elastic Cloud. When credentials were received the deployment of Elastic Cloud was started.

Thesis describes with photos, what different kind of settings can be adjusted while making Elastic Cloud deployment. After that JSON data was sent to Elasticsearch. And when data reached Elasticsearch it was explored and visualized with Kibana.

Result of this thesis is a working Elastic Cloud deployment and a dashboard packed with different kinds of visualizations made with Kibana.

The conclusion of this thesis was that maintaining and creating Elastic cloud deployment is easy. Elastic Cloud is a good way to have a fast and easily configurable database in cloud environment. It was noted that good security practices and measures are valid part of Elastic Cloud. of security

Keywords	SaaS, Elastic, Kibaba

Sisällys

1 Johdanto

Insinöörityön tavoitteena on tutustua Elastic Cloud -ohjelmistopalvelun asentamiseen ja datan visualisointiin Elasticsearchin päällä toimivan Kibanan avulla.

Insinöörityössä asennetaan Elastic Cloud -julkaisu, joka sisältää Elasticsearch- ja Kibana-palvelut. Asentamisen jälkeen Elasticsearchiin lähetetään Counter-Strike Global Offensive kollektiivi de_liverin harjoituspäivältä kerättyä JSON-dataa, jota visualisoidaan Kibanan avulla.

Kun mietittiin insinöörityön aihetta, haluttiin kokeilla jotain ohjelmistopalvelua niiden helppouden, nopeuden, valmiiden asetusten ja skaalautuvuuden takia.

Insinöörityön ensimmäisessä teoriaosuudessa käydään läpi pilvipalveluiden yleisimmät mallit ja kerrotaan hieman pilvipalveluiden hyödyistä. Toisessa teoriaosuudessa esitellään Elastic Cloudiin kuuluvien palveluiden Elasticsearchin ja Kibanan toimintoja.

Insinöörityön teko hetkellä Elastic Cloud -palvelu tarjoaa Elasticsearchin ja Kibanan lisäksi X-Packin ominaisuuksia, mutta niihin insinöörityössä ei tutustuta.

Insinöörityön pääpiste on tutustua Elastic Cloud -julkaisun asentamiseen ja julkaisun toimintoihin. Insinöörityö esittää myös, miten Kibana toimii Elasticsearchin päällä vaivattomasti ja on yksi tapa visualisoida haluttua dataa.

Datan lähetystä ja visualisointia esitetään, mutta niissä ei mennä syvälle. Insinöörityön lukemisen jälkeen lukijalla on hyvä peruspohja tutustua Elastic Cloudiin, Elasticsearchiin ja Kibanaan tarkemmin.

2 Pilvipalvelut

Pilvipalvelulla tarkoitetaan, etteivät käytetyt ohjelmat, tehtävät, suoritukset tai tiedot tapahdu käyttäjän omassa verkossa vaan palveluntarjoajan tarjoamalla palvelimella internetin yli. (1.) Pilvipalvelun sana ”pilvi” tulee siitä, miten internet usein esitetään tietoverkkoja kuvaavissa kaavioissa. (2.)

2.1 Yleisimmät pilvipalvelumallit

2.1.1 Infrastruktuuri palveluna

Infrastruktuuria palveluna tarjoava pilvipalvelumalli tunnetaan paremmin englanninkielisellä nimellä Infrastructure as a Service (IaaS). (3.)

Infrastruktuuri palveluna -mallissa palveluntarjoaja tarjoaa asiakkaalleen määritellyn laitteiston, mihin asiakas voi perustaa palvelimen. Palvelun ostajan vastuulle jää kokonaisvaltaisesti kaikki palvelimen asetusten määrittämisestä hallinnointiin. Infrastruktuuri palveluna -mallissa palvelun tarjoajan vastuulla on usein ainoastaan ostetun laitteiston tarjoaminen. (3.)

Infrastruktuuri palveluna sopii hyvin ostajalle, jolla on paljon tietoteknistä osaamista ja joka haluaa maksimaalisen joustavuuden palvelimensa asetusten määrittämiseen. Tämä tuo samalla myös paljon lisätyötä palvelun ylläpitoon ja tietoturvaan kohdistuvilla osa-alueilla. (3.)

2.1.2 Sovellusalusta palveluna

Sovellusalusta palveluna on pilvipalvelumalli, mikä paremmin tunnetaan englanninkielisellä nimellä Platform as a sevice (Paas). (3.)

Sovellusalusta palveluna -mallissa palveluntarjoaja tarjoaa valmiin pohjan sovelluksen toteuttamiselle. Näin sovelluksen kehittäjän tarvitsee vain siirtää valmis sovelluksensa ostetulle alustalle palveluntarjoajan tarjoamia yhteysteitä pitkin. (3.)

Sovellusalusta palveluna -mallissa palveluntarjoaja ylläpitää käyttöjärjestelmän ja varusohjelmistoja. Näin ostajan tulee huolehtia vain oman sovelluksensa ylläpidosta. (3.)

Sovellusalusta palveluna -malli sopii hyvin kehittäjille, jotka haluavat valmiiksi asennetun infrastruktuurin, jossa on kaikki tarvittava valmiina sovelluksen toimimista varten. Valmis sovellusalusta mahdollistaa nopean sovelluskehityksen, sillä sovelluksen takana olevan infrastruktuurin asetusten määrittämiseen ei kulu ylimääräistä aikaa. (3.)

2.1.3 Ohjelmisto palveluna

Ohjelmistoa tarjoava pilvipalvelumalli, mikä paremmin tunnetaan englanninkielisellä nimellä Software as a service (SaaS). (3.)

Ohjelmistoa palveluna tarjoava pilvipalvelumalli tarjoaa käyttäjälleen valmiiksi toimivan ohjelmiston pilvessä. Tällöin palveluntarjoaja hoitaa kaikki palvelun ylläpitämiseen liittyvät asiat. Näin palvelun ostajan vastuulle jää usein vain palvelun käyttö. (3.)

Ohjelmisto palveluna sopii erinomaisesti niille, jotka haluavat vain käyttää palvelua. Tällöin ollaan kuitenkin usein täysin riippuvaisia palveluntarjoajan tekemistä ratkaisuista palvelun tai sovelluksen kehityksessä. (3.)

2.2 Pilvipalveluiden hyödyt

Pilvipalvelut tuovat mukanaan kustannustehokkuuden, joustavuuden ja nopeuden palveluiden toteutuksessa. Pilvipalveluiden avulla voidaan testata ja ottaa käyttöön uusia palveluita ilman, että tarvitsee toteuttaa koko järjestelmää alusta asti tai sitoutua järjestelmään pysyvästi. (3.)

2.2.1 Kustannustehokkuus

Pilvipalveluita käyttämällä asiakkaan ei tarvitse ostaa omaa laitteistoa. Oman laitteiston ostamisen sijaan pilvipalveluntarjoajalta voidaan ostaa juuri tarvittava määrä laskentatehoa palvelimille. Näin säästetään ylimääräisen laitteiston osto-, korjaus-, huolto- ja sähkökuluja. (4.)

Pilvipalveluissa hintaan kuuluu usein myös palvelimen hoito ja ohjelmistojen ylläpito. Tämä vähentää ajallista tarvetta jatkuvalle ylläpidolle ja ohjelmistojen oikeanlaiselle päivitykselle. Näin palvelun käyttäjälle jää enemmän aikaa itse palvelun suunnittelulle ja kehitykselle. (2.)

Pilvipalveluiden avulla on nopeaa ja halpaa testata uusia toteutuksia ilman, että palvelun tuottamiseen tarvitsee sitoutua pitkäksi aikaa. Koska pilvipalveluiden käytöstä maksetaan vain niin paljon kuin palveluita käytetään. (2.)

2.2.2 Skaalautuvuus

Pilvipalvelut skaalautuvat ja ovat saatavilla erittäin nopeasti. Monesti toimivaan tietotekniseen ympäristöön tai sovellukseen vaaditaan monia erilaisia teknisiä alustoja ja toteutuksia. Tätä varten tarvitaan monesti useita eri instansseja. Pilvipalvelun tarjoajalta on helppoa ja nopeaa ostaa juuri senhetkiseen tilanteeseen tarvittava määrä palveluita ja kapasiteettia. (5.)

Pilvipalvelut ovat tehokkaita sovelluksille, joiden käyttö on kasvussa tai joiden käyttö vaihtelee suuresti. Kun palveluiden tarve kasvaa tai pienenee, voi muutamalla klikkauksella tai komennolla saada asennettua itselleen uuden ja sopivankokoisen ympäristön. (5.)

Pilvipalvelun avulla moni systeemi integraatio on tehty helposti hoidettavaksi graafisilla käyttöliittymillä. (6.)

2.2.3 Saatavuus

Pilviarkkitehtuuri avaa mahdollisuuksia useammalle käyttäjälle päästä käsiksi dataan ja digitaalisiin työkaluihin selaimella tai mobiililaitteella samanaikaisesti. Lisäksi pilvipalvelut huolehtivat usein työkalujen päivityksistä automaattisesti. (7.)

Hallinnoidun pilvialustan saatavuus on erittäin hyvä. Monet pilvipalvelut lupaavat ympärivuotisen 99.99 prosentin saatavuuden palveluilleen. Näin suuri saatavuus säästää paljon vaivaa palveluiden toimintavarmistusten osalta. Jos pilvipalvelun palvelin jostain syystä menee epäkuntoon, palvelu voidaan helposti siirtää seuraavalle palveluntarjoajan palvelimelle. (8.)

2.2.4 Varmennus

Monesti erilaisten tiedostojen ja palveluiden varmistusten teko on kallista, ja se vie aikaa, joten pienillä toimijoilla ei välttämättä ole resursseja täydellisten varmistusten tekemiseen. (7.)

Pilvipalvelussa voidaan helposti ottaa varmuuskopioita ja hajauttaa varmuuskopioiden säilytys useammalle eri palvelimelle. Moni pilvipalvelu mahdollistaa automaattisen varmuuskopioinnin ja niiden helpon elvyttämisen. Usein varmuuskopiointi sisältyykin suoraan palveluun. (6.)

2.2.5 Tietoturva

Pilvipalveluiden tarjoajat monitoroivat tietoturvaansa erittäin laajasti keskiverto käyttäjiin verrattuna. Pilvipalveluiden tietoturvasta vastaavat tietoturva-asiantuntijat. Monesti palveluita ei päästä edes käyttämään, jos määritettyjä tietoturva kriteereitä ei ole täytetty. Pilvipalvelut antavat usein tietoturvaan liittyviä vinkkejä, jotta asiakas voisi halutessaan parantaa tietoturvaansa. (9.)

Pilvipalveluita käytettäessä ohjelmistot ja tiedot ovat fyysisesti turvassa, vaikka käyttäjän laite hajoaisi tai häviäisi. Kun tiedot ovat tallessa pilvipalvelussa, niihin päästään käsiksi taas kirjautumalla palveluun. (1.) Salesforce.com-sivuston mukaan datat ovat pilvessä

enemmän turvassa kuin yrityksen omassa infrassa. Sivuston mukaan suuri osa datavarkauksista tapahtuukin yrityksen työntekijöiden toimesta. (9.)

Dataa pilveen tallennettaessa on tärkeää myös miettiä, missä maassa ostettu pilvipalvelu sijaitsee. Joissain tapauksissa voidaan edellyttää, että tietyt datat eivät esimerkiksi saisi poistua tietyn maan ulkopuolelle. Tämä mahdollistaa myös globaalin toiminnan yritykselle. (1.)

Tietotekniset toteutukset vaativat jatkuvaa valvomista ja päivitysten ajoa. Monet palveluntarjoajat mainostavatkin hoitavansa palveluidensa kunnossapitoa ja päivityksiä ympärivuorokautisesti. (1.)

3 Elastic Cloud

Elastic Cloud on Elasticsearchin kehittäjien ylläpitämä Saas-palvelu, jolla on helppo asentaa, operoida ja skaalata Elastic-tuotteita ja ratkaisuja Google Cloud Platform- ja Amazon WEB Services -palveluntarjoajilla. Elastic Cloud tarjoaa Elastic-tuotteista Elasticsearchin, Kibanan ja X-pacin. (10.)

3.1 Elasticsearch

Elasticsearch on hyvin skaalautuva avoin Javalla kehitetty hakumoottori, mikä perustuu Apachen Lucene -kirjastoon. (11.) Elasticsearchin avulla voidaan tallentaa, etsiä ja analysoida isoja määriä dataa lähes reaaliajassa. (12.) Elasticsearch tarjoaa käyttäjälleen hajautetun ja skeema vapaan JSON dokumentointikannan HTTP API -rajapinnalla. (11.)

Hakumoottorit ovat NOSQL-tietokantoja, jotka on erityisesti tarkoitettu datasisällön etsimiseen sinne tallennetusta datasta. Sen lisäksi, että hakumoottorit ovat optimoitu datasisällön nopeaan etsimiseen, on niillä monesti myös seuraavia ominaisuuksia: (13.)

- tuki monimutkaisiin kyselyihin
- mahdollisuus hakea osaa kokonaisesta tekstistä
- kyky arvottaa ja luokitella haetut tulokset
- geograaffinen haku
- hajautettu hakeminen mahdollistaen suuren skaalautuvuuden.

3.2 Elasticsearch-klusterointi

3.2.1 Klusteri

Klusteri on yhden tai useamman solmun kokoelma palvelimia. Klusteri pitää sisällään koko tietokannan datat mahdollistaen datan indeksoinnin sekä hakemisen kaikkien solmujen yli. Yhdessä klusterissa voi olla rajaton määrä solmuja. Jokainen solmu tulee erikseen liittää osaksi klusteria. On tärkeää huolehtia, ettei solmuja liitetä osaksi useampaa klusteria. (14.)

3.2.2 Solmut

Solmu on yksittäinen kone, joka on osa klusteria. Yhdessä klusterissa voi olla rajoittamaton määrä solmuja. Solmuun voidaan tallentaa dataa, ja se osallistuu samalla koko klusterin indeksointiin ja hakukapasiteettiin. (14.)

Elasticsearchin jokaisella klusterilla ja solmulla on oma uniikki nimensä. Elasticsearchin solmut liitetään klusteriin määrittämällä solmu liittymään klusterinimeen. Näin solmut etsivät automaattisesti heille osoitetun klusterin nimeä verkkonsa sisältä. (14.)

Isäntäsolmut

Isäntäsolmu on vastuussa vain klusterin toiminnasta ja hallinnoinnista. Isäntäsolmut eivät sisällä dataa, joten näiden solmujen ei tarvitse osallistua datan hakemiseen ja indeksointiin. Tämän takia isäntäsolmujen prosessori, muisti ja tallennuskapasiteetti vaatimukset ovat pienemmät. (15.)

Kuumat solmut

Nämä solmut tekevät kaikki indeksoinnit klusterissa. Lisäksi kuumat solmut pitävät viimeisimmät indeksit. Koska indeksointi on erittäin prosessori- ja levynopeustehoa vaativa operaatio. Kuumien solmujen prosessori ja levynopeus on oltava tarpeeksi tehokkaita. Kuumien solmujen määrää ja tehoa määrittäessä on hyvä miettiä, paljonko uutta dataa halutaan indeksoida. (15.)

Lämpimät solmut

Nämä solmut on suunniteltu käsittelemään suuria määriä vain lukuindeksejä, joihin ei hakuja usein kohdistu. Näihin solmuihin voidaan valita hieman hitaampaa ja halvempaa tallennustilaa. Oikeanlaista laitteistoa ja solmujen lukumäärää mietittäessä suositellaan tekemään testikyselyitä lämpimille solmuille, joiden avulla määritetään halutunlainen toimivuus. (15.)

3.2.3 Indeksi

Indeksi on kokoelma dokumentteja, joilla on samantapaisia datapiirteitä. Indeksejä voidaan perustaa klusterin sisään rajaton määrä. Näin jokaiseen käyttötarkoitukseen voidaan perustaa oma indeksinsä. Indeksit identifioidaan uniikilla nimellä. Indeksin nimeä käytetään, kun halutaan suorittaa indeksointeja, hakuja, päivityksiä tai poistoja kohdistuen indeksissä oleviin dokumentteihin. (14.)

3.2.4 Dokumentti

Dokumentti on kokoelma datakenttiä, jotka voidaan indeksoida dokumentin avulla yhdeksi kokonaisuudeksi. Yksi Indeksi voi sisältää rajattoman määrän dokumentteja. Elasticsearchin dokumentit esitetään JSON-muodossa. Dokumentit sijoitetaan aina indeksiin ja sen kentille tulee aina merkitä tyypitykset. (14.)

3.2.5 Tyypitys

Tyypitys on prosessi, millä määritellään dokumentin kenttien arvotyypit. Tyypityksellä määritetään, onko kentän arvo tekstiä, numeroarvo vai kenties esim. päivämääränä. (16.)

Elasticsearchissä dokumentin kentillä voi olla seuraavia tyypityksiä:

perusdatatyypitykset ovat:

- merkki
- päivämäärä
- numeerinen (Pitkä, lyhyt, kokonaisluku, bitti, tupla ja liukuluku)
- totuusarvo
- binääri
- alue.

Kompleksiset datatyypitykset ovat:

- lista
- objekti
- objektilista.

Geo-tyypitykset ovat:

- geo-piste
- geo-muoto.

Spesiaalit datatyypitykset:

- IPv4
- täydennys
- liite. (17.)

Täsmällinen tyypitys

Jokaiselle indeksin kentälle määritellään manuaalisesti tyypitystapa, mikä määrittää ,miten tallennettava dokumentti indeksoidaan Elasticsearchiin. Tyypitys sisältää aina dokumentin metakentät, indexin, typen, id:n, sourcen sekä varsinaiset datakentät. (16.)

Dynaaminen tyypitys

Elasticsearch tukee myös dynaamista tyypitystä, millä tarkoitetaan, ettei kenttien arvoja ole pakko määrittää etukäteen. Elasticsearch yrittää automaattisesti arvata kentän tyypin ja tallentaa sen arvaamallaan tyypillä. Tämä voi olla aluksi helppo tapa päästä tyypityksessä alkuun, mutta vaativamman käytön edetessä kannattaa käyttäjän itse tyypittää datansa haluttua käyttötarkoitusta varten. (16.)

3.2.6 Indeksin sirpalointi ja sirpaleiden jäljennökset

Yhteen indeksiin voidaan tallentaa suuri määrä dataa, mikä saattaa ylittää yhden solmun tallennuskapasiteetin. Tätä varten Elasticsearch tarjoaa mahdollisuuden jakaa indeksi useampaan sirpaleeseen. Jokainen näistä indeksin sirpaleista toimii omana palasenaan, mikä mahdollistaa indeksin jakamisen useammalle solmulle. (14.)

Sirpalointi on tärkeää, koska:

- sirpalointi mahdollistaa sisällön jakamisen horisontaalisesti solmujen kesken
- sirpaloinnin avulla laskentatehoa vaativia tehtäviä voidaan jakaa useamman koneen kesken.

Useamman sirpaleen indeksissä Elasticsearch vastaa täysin kyselyiden vastausten yhteenlasketusta palautuksesta. (14.)

3.2.7 Sirpaleiden jäljennökset

Sirpalemallin vikasietoisuuden varalle Elasticsearch on esitellyt sirpaleiden jäljennökset. Onkin tärkeää, että sirpaleiden jäljennökset tallennetaan aina eri solmuun kuin alkuperäinen Indeksin sirpale. (14.)

Sirpaleiden jäljennökset tarjoavat käyttäjälle paremman saatavuuden Elasticsearchin sisältöön tilanteissa, joissa jokin klusterin koneista katoaa klusterin näkyvistä. Tämän lisäksi jäljennösten avulla voidaan parantaa hakujen keston nopeutta, sillä yhteen sirpaleeseen kohdistuva haku kohdistetaan kaikkiin sirpaleen jäljennöksiin rinnakkaishakuna. (14.)

3.2.8 HTTP REST API -rajapinta

Yksi Elasticsearchin ominaisuuksista on sen HTTP REST API -rajapinta. HTTP REST API -rajapinnan avulla voidaan tutkia ja vaikuttaa Elasticsearchin toimintaan ja sinne tallennettuun dataan määritellyillä http-kutsuilla. HTTP REST API -rajapinnan kommunikointi tapahtuu JSON-viesteillä. (18.)

Elasticsearchin HTTP REST API -rajapinta on kattava. Ohessa muutama esimerkki Elasticsearchin sisältämistä HTTP REST API -rajapinoista.

Dokumentti API -rajapinta

- Dokumentti API -rajapintaa käytetään dokumenttien käsittelyyn Elasticsearchissä. Dokumentti API -rajapinnalla voidaan muun muassa luoda, päivittää, siirtää tai poistaa dokumentteja Elasticsearchissa. (18.)

Haku API -rajapinta

- Haku API -rajapinnalla voidaan tehdä erilaisia kyselyitä Elasticsearchin indekseissä olevasta datasta. (18.)

Indeksi API -rajapinta

- Indeksi API -rajapinnalla voidaan käsitellä Elasticsearchin indeksejä, tyypityksiä ja kaavoja. Indeksi API -rajapinnalla voidaan mm. luoda, poistaa uusi indeksi tai kysyä, onko kyseistä indeksiä olemassa ja miltä sen tyypitys näyttää. (18.)

Klusteri API -rajapinta

- Klusteri API -rajapinnan avulla voidaan hoitaa ja monitoroida Elasticsearch-klusteria. Klusteri API -rajapinnalla voidaan kysyä mm. klusterin tilaa ja toimintakuntoa. (18.)

3.2.9 Snapshot

Snapshot on varmuuskopio Elasticsearch-klusterista. Snapshot voidaan ottaa joko yksittäisistä klusterin indekseistä tai koko klusterista ja tallentaa haluttuun tallennuspaikkaan. Elasticsearch ottaa snapshoteja vähitellen, millä tarkoitetaan, että Elasticsearch koittaa

olla varmuuskopioimatta uudelleen jo varmuuskopioitua dataa. Näin varmuuskopioita voidaan ottaa tehokkaasti lyhyinkin väliajoin. (19.)

Snapshotit voidaan palauttaa päällä olevaan klusteriin Elasticsearchin Restore API -rajapinnan kautta. Indeksin nimeä ja joitakin sen asetuksia voidaan muuttaa snapshottia palauttaessa, mikä antaa joustavuutta snapshottien ja niiden palauttamisen käyttöön. (19.)

3.3 Kibana

Kibana on avoin datan analytiikka- ja visualisointialusta, mikä on suunniteltu toimimaan Elasticsearchin kanssa. Kibanalla voidaan olla vuorovaikutuksessa, etsiä tai näyttää dataa Elasticsearchin indekseistä. Kibanalla on helppo suorittaa erilaisia analyyseja ja visualisointeja datasta erilaisin taulukoin, kuvaajin ja esitystavoin. Kibana on helppo ja nopea asentaa Elasticsearchin rinnalle, jolloin päästään nopeasti tutkimaan Elasticsearchiin tallennettua dataa visualisoidussa muodossa lähes reaaliajassa. (20.)

3.3.1 Dev tools

Dev Tools on Kibanan tarjoama konsoli-ikkuna. Dev Toolsin avulla voidaan tehdä kutsuja Elasticsearchin REST API -rajapintaan ja nähdä kutsujen vastaukset Dev Toolsin vastausikkunassa. (21.)

3.3.2 Discover

Discovery-sivulla voidaan tutkia Elasticsearchiin tallennettua dataa interaktiivisesti. Jokaisen indeksin jokaista dokumenttia voidaan tarkastella lähemmin, kunhan valittu indeksirakenne on oikein. Discovery-sivulla voidaan tehdä hakuja, joita voidaan lisäksi filtteröidä. Näin haun tulokset statistiikan kera näkyvät käyttäjälle ja ovat hänen kosketeltavissaan. Lisäksi jos indeksille on määritetty aikakenttä, Discovery-sivu näyttää aikajana histogrammin haetuista datoista. (22.)

3.3.3 Visualize

Visualize mahdollistaa käyttäjälle Elasticsearch indeksien datan visualisoinnin. Näistä tallennetuista visualisoinneista käyttäjä voi kerätä yleisnäkymän, jossa esitetään siihen kerättyjä visualisointeja. (23.)

Kibanan visualisoinnit perustuvat Elasticsearch kyselyihin. Sarjoista Elasticsearchin keräelmiä voidaan ottaa ote ja prosessoida visualisoitu kaavio halutusta aiheesta. Datan visualisointi voidaan aloittaa joko puhtaalta pöydältä tai valmiiksi tehdyn haun avulla. (16)

Erilaisia visualisointitapoja Elasticsearchissä:

Perustaulukot

- viivakaavio
- aluekaavio
- pylväskaavio
- heat map
- ympyräkaavio

Data

- datataulukko
- arvo

Kartta

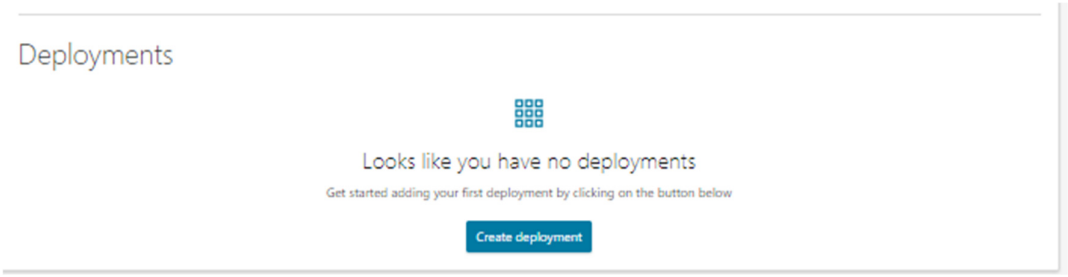
- koordinaattikartta
- aihekohtainen-kartta. (24.)

4 Elastic Cloudin luonti

4.1 Elastic Cloudin luonti

Insinöörityö aloitettiin osoitteesta https://www.elastic.co/cloud, josta saatiin 14 päivän ilmainen kokeilujakso Elastic Cloudiin. Syöttämällä sähköpostiosoite linkissä olevaan kenttään saatiin palveluntarjoajalta sähköpostilla vahvistusviestilinkki, joka johti salasanantekosivulle. Kun salasana oli vahvistettu, ohjautui sivu Elastic Cloud -konsolin yleiskatsaussivulle.

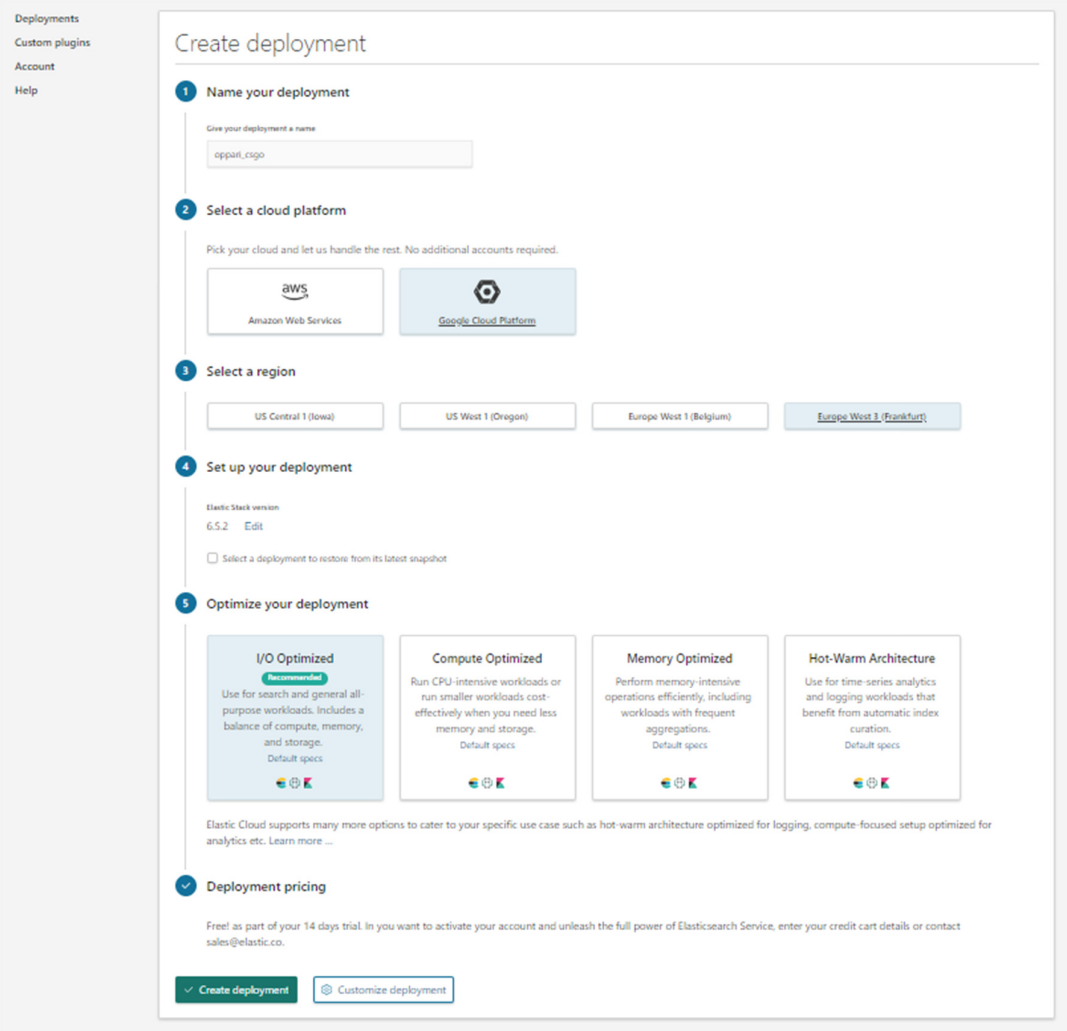
Koska yhtään Elastic Cloud -julkaisua ei ollut vielä luotu, painettiin sivun tarjoamaa nappulaa ”Create deployment”, josta voitin aloittaa uuden Elastic Cloud -julkaisun luonti.



Kuva 1. Ensimmäisen julkaisun aloitusikkuna.

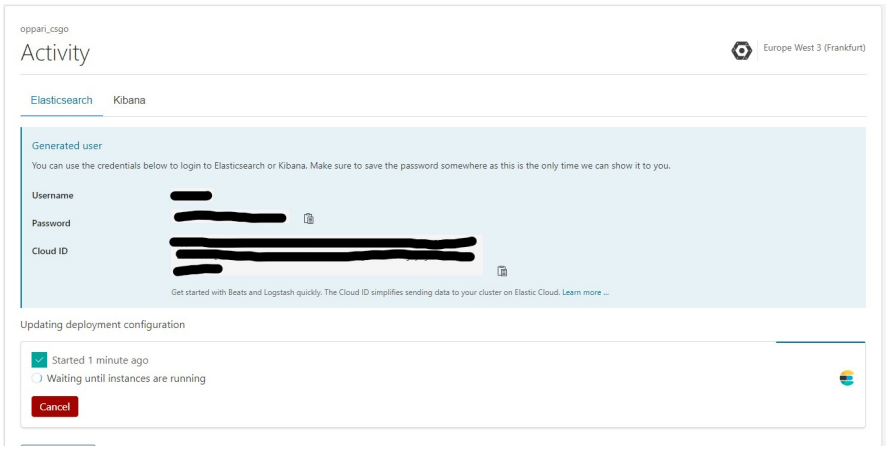
Tästä ohjauduttiin sivulle, jossa määriteltiin Elastic Cloud -julkaisun lähtötiedot:

1. Elastic Cloud -julkaisun nimi ”oppari_csgo”
2. pilvipalvelun tarjoaja Google Cloud Platform
3. palvelun sijainniksi Frankfurt
4. Elastic Stacin uusin versio 6.5.2
5. koneen resurssiksi ”I/O Optimized”.



Kuva 2. Elastic Cloud -julkaisun luonti.

Kun kyseiset valinnat oli tehty, voitiin luoda kyseinen julkaisu. Kun painettiin julkaisun-luomisnappia ”Create deployment”, ohjauduttiin julkaisun omalle Activity-sivulle. Tältä sivulta saatiin instanssin oletusarvokäyttäjätunnus ”elastic”, käyttäjätunnuksen salasana ja julkaisun Cloud Id. Samalla nähtiin palveluntarjoajan luovan instansseja valmiiksi.

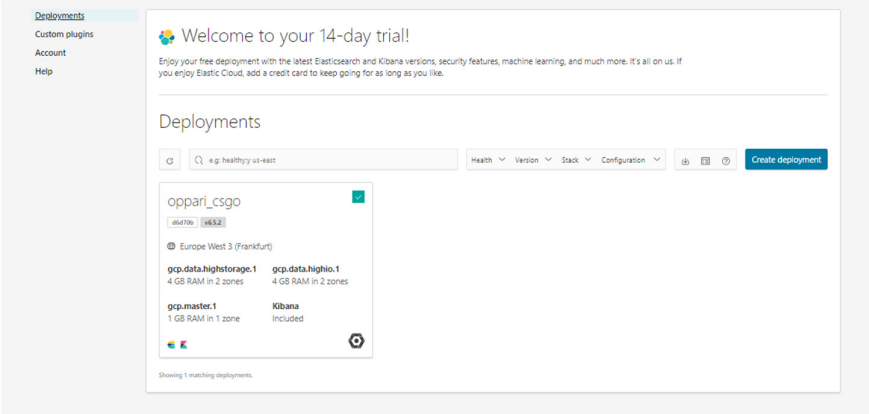


Kuva 3. Koneiden alustus ja tunnukset.

Instanssien luonti kesti noin kaksi minuuttia, minkä jälkeen Elasticsearch ja Kibana olivat käyttövalmiina.

4.2 Elastic Cloud -katsaus

Elastic Clouf -julkaisun luonnin jälkeen Elastic Cloudin pääsivulla nähtiin luotu instanssi ”oppari_csgo”. Julkaisua painamalla päästiin tarkastelemaan sitä lähemmin.

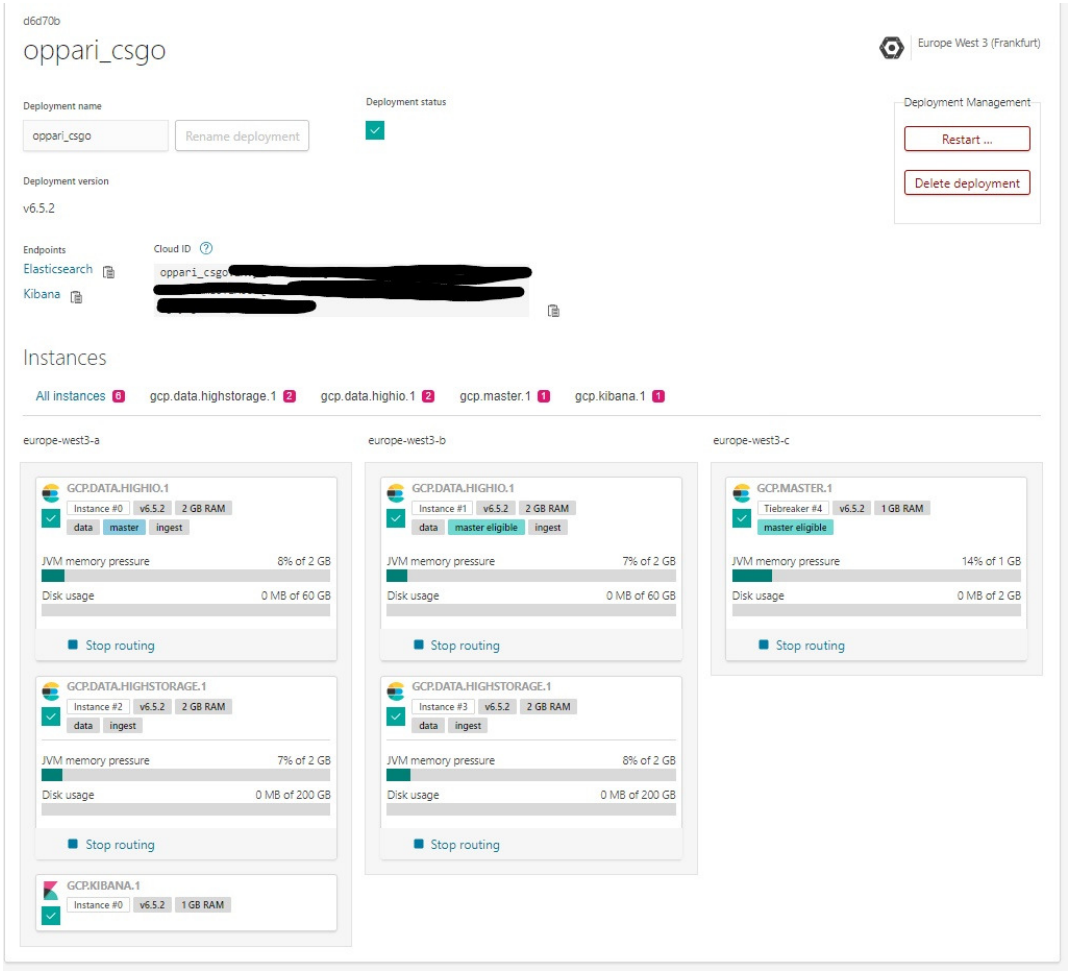


Kuva 4. Elastic Cloudin yleissivulla nähtävissä kaikki luodut julkaisut.

Painettiin julkaisua ”oppari_csgo”, josta päästiin tutkimaan julkaisua tarkemmin.

4.2.1 Julkaisun yleissivu

Elastic Cloud -julkaisun yleissivulta voitiin tarkastella julkaisun yleistilaa. Sivulla nähtiin julkaisun nimi, Elastic-versio, Cloud ID, Kibanan ja Elasticsearchin päätepisteiden osoitteet. Lisäksi julkaisun yleissivulta nähtiin julkaisun instanssit sekä instanssien muistiin ja levyntilan käyttöön liittyvät parametrit.

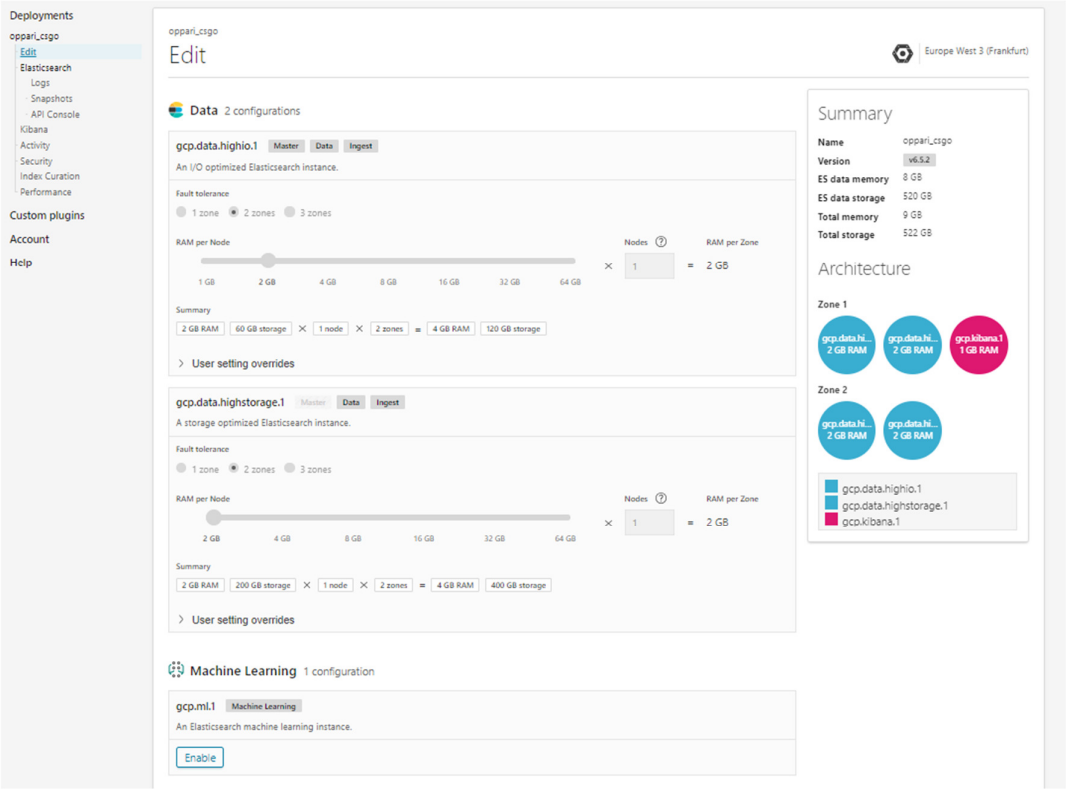


Kuva 5. Julkaisun yleissivun näkymä.

Tältä sivulta voitiin myös uudelleen käynnistää ja poistaa julkaisusivun yläkulman napeista ”restart ja ”delete deployment”.

4.2.2 Edit

Edit-sivulla voitiin muokata jokaisen instanssin kapasiteettia erikseen tai luoda uusia instansseja. Edit-sivulla voitiin myös muokata instanssien virhetoleranssia, eli kahdentaa instanssi useammalle eri palveluntarjoajan alueelle. Näiden lisäksi jokaisen instanssin muistia, levytilaa ja solmujen määrää voitiin muokata sivulla erikseen.

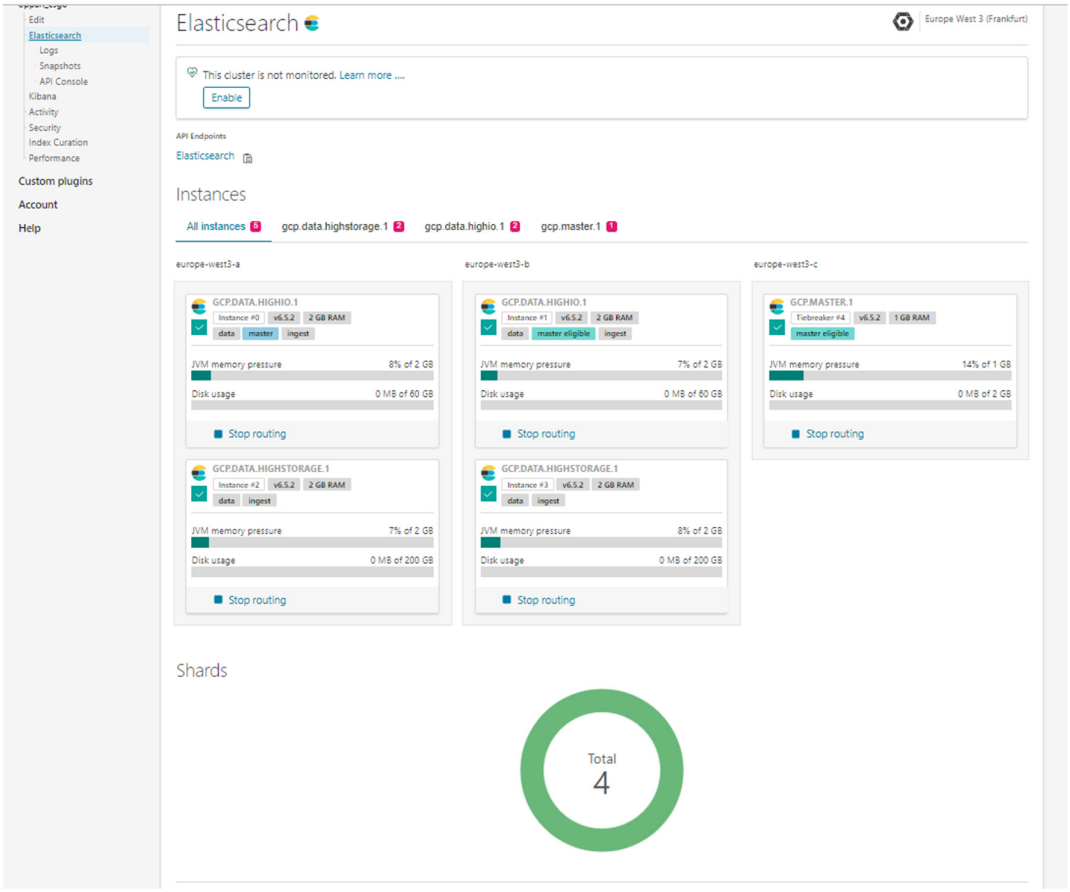


Kuva 6. Edit-sivun näkymää.

Edit-sivulla olisi voitu muokata myös eri instanssien lisäosia ja nähdä instanssien kapasiteetin yhteissumma sekä näiden yhteishinta.

4.2.3 Elasticsearch

Julkaisun Elasticsearch-kohdasta voitiin katsoa Elasticsearchiin liittyvien instanssien toimintaa ja siihen liittyviä parametreja. Sivulta olisi voitu aktivoida myös erillinen klusterin monitorointilisäosa.

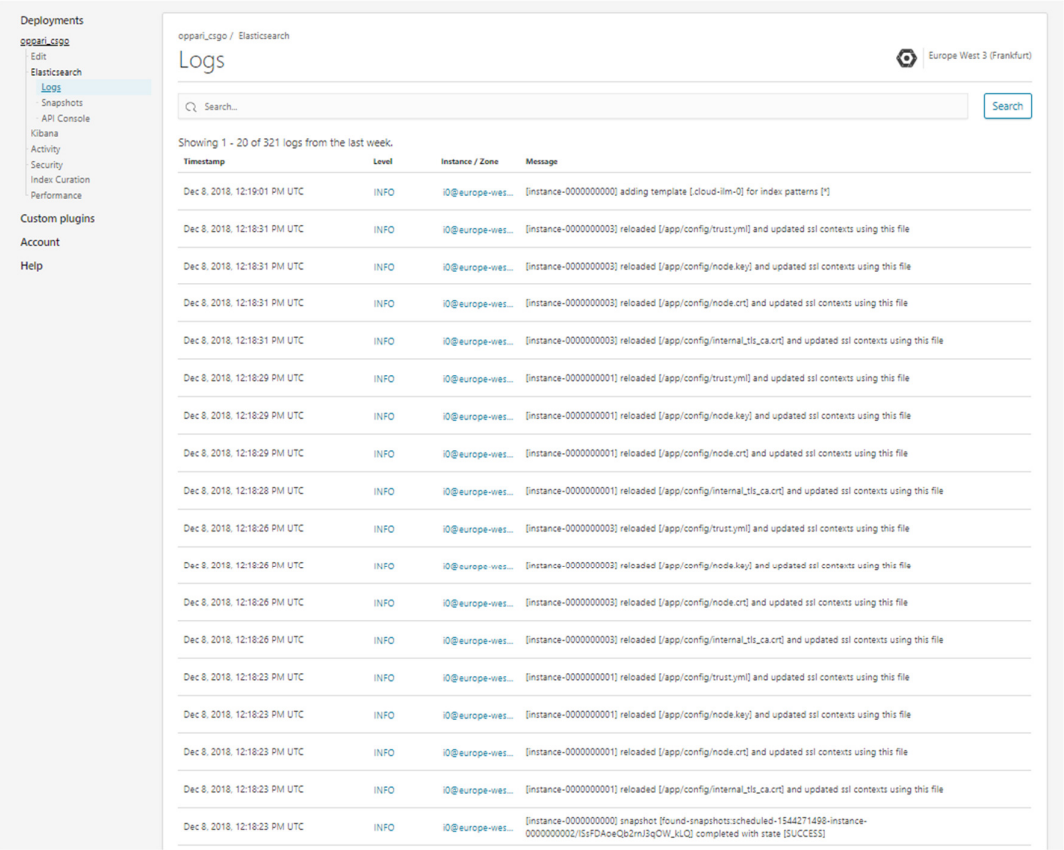


Kuva 7. Julkaisun Elasticsearch-instanssit.

Sivun alalaidasta nähdään myös Elasticsearch-indeksien sirpaleiden yhteismäärä.

Logs

Elasticsearch-kohdan alta löytyivät Elasticsearchin toimintaan liittyvät logit.

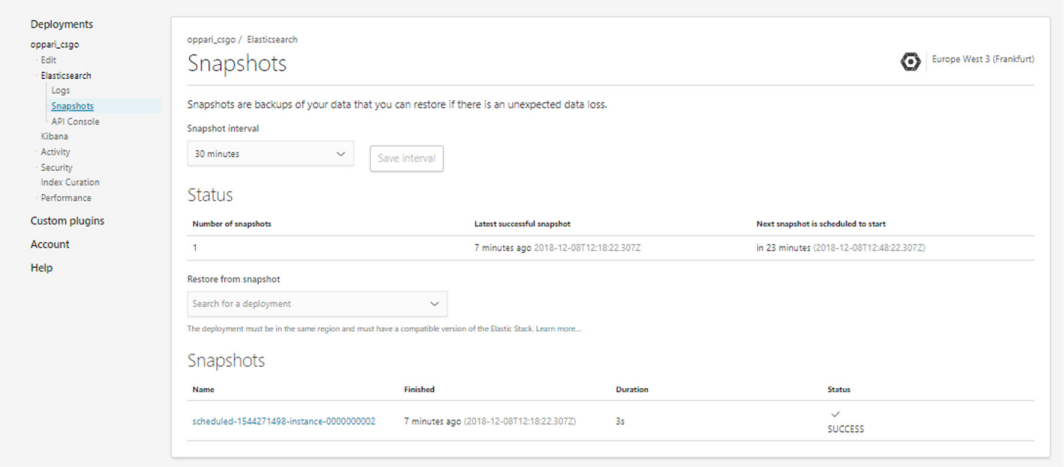


Kuva 8. Elasticsearch-logit.

Sivulla oli myös logien etsimistä helpottava hakukenttä.

Snapshots

Elasticsearchin valikoiden alta löytyi kohta ”Snapshot”, missä olisi voitu muokata Elasticsearchin snapshottien ottamistiheyttä. Sivulla nähtiin myös otetut snapshotit. Painamalla snapshottia voitiin nähdä tarkempia tietoja snapshotista, kuten Elasticsearch-versio, sen sisältämät indeksit ja snapshotin ottoajankohta.

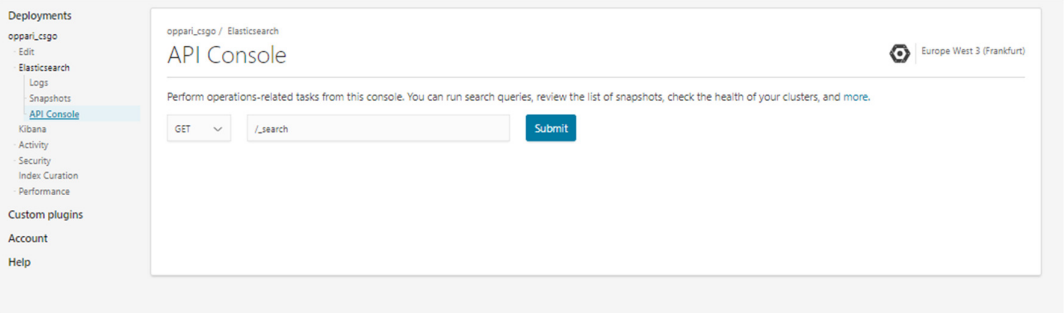


Kuva 9. Yleiskuva snapshot-sivulta.

Snapshot-sivulta olisi myös voitu palauttaa haluttu snapshot takaisin käyttöön.

4.2.4 Api Console

Elasticsearch-valikon alta löytyi myös Elasticsearchin API Console.

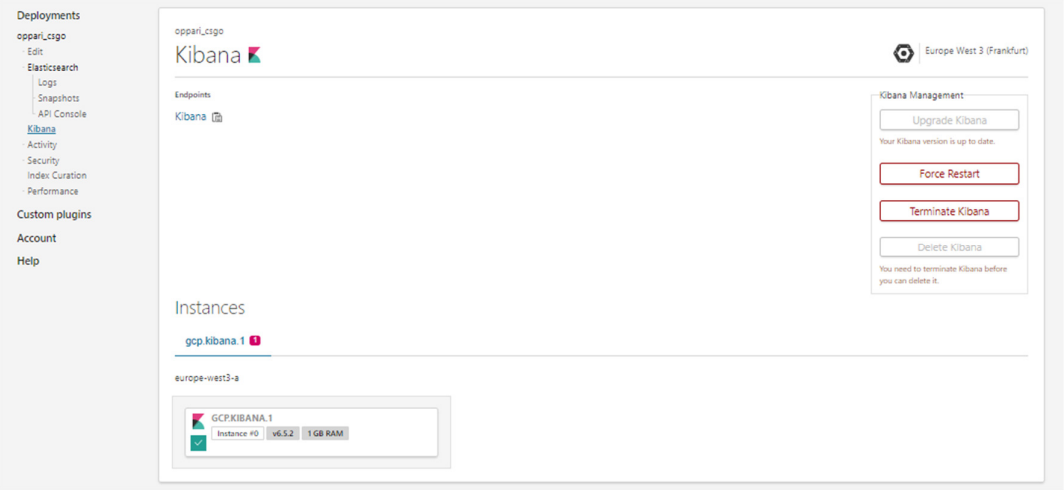


Kuva 10. Kuva API Consolesta.

Kyseistä API Consolea ei insinöörityössä käytetty vaan Elasticsearchin API-rajapinnan kanssa keskusteltiin Kibanan Dev Toolsin avulla.

4.2.5 Kibana

Julkaisun yleissivun Kibana-välilehdellä voitiin nähdä Kibana-instanssien toiminta. Sivulta voitiin mennä suoraan linkin kautta Kibana-instanssiin tai kopioida Kibanan osoite.

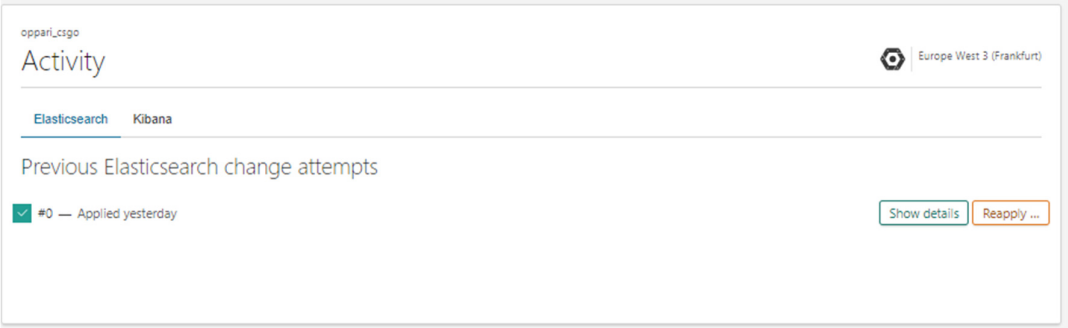


Kuva 11. Kibana-instanssit.

Sivulta olisi ollut mahdollista myös käynnistää uudelleen ja poistaa Kibana-instanssi.

4.2.6 Activity

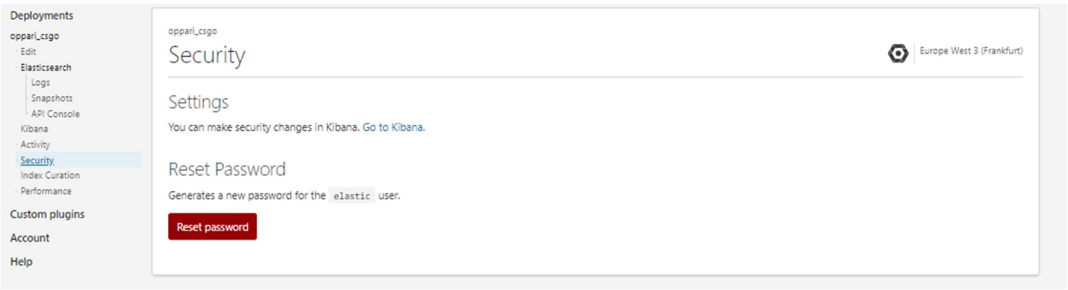
Activity-sivulta nähtiin Elasticsearchiin ja Kibanaan kohdistuneet aktiviteetit, joita tuolloin ei vielä paljoa ollut.



Kuva 12. Activity-sivu.

4.2.7 Security

Security-sivulla kerrottiin, että lisää turva-asetuksia voitaisiin tehdä Kibanassa.

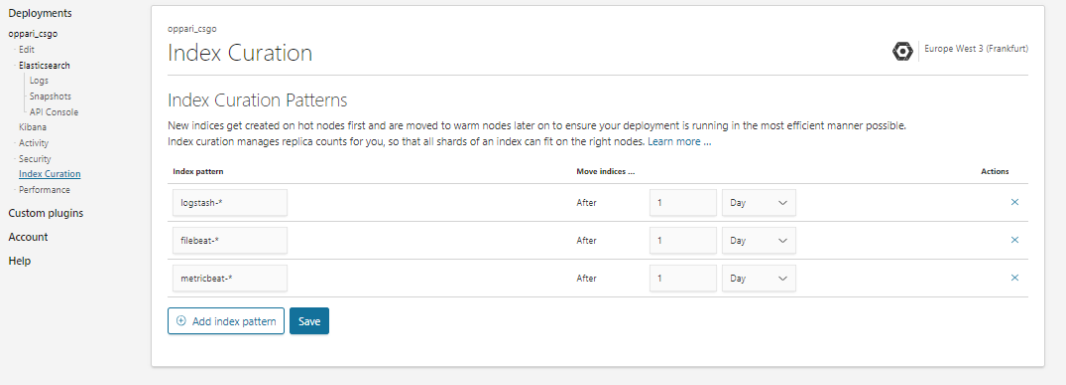


Kuva 13. Security-sivu.

Security-sivulta oli myös mahdollista resetoida Elastic Cloudin -oletusarvoisen käyttäjän ”elastic”-salanana.

4.2.8 Index Curation

Index Curation -sivulta pystyttiin muokata Elasticsearch-indeksien siirtoa kuumista solmuista lämpimiin solmuihin.

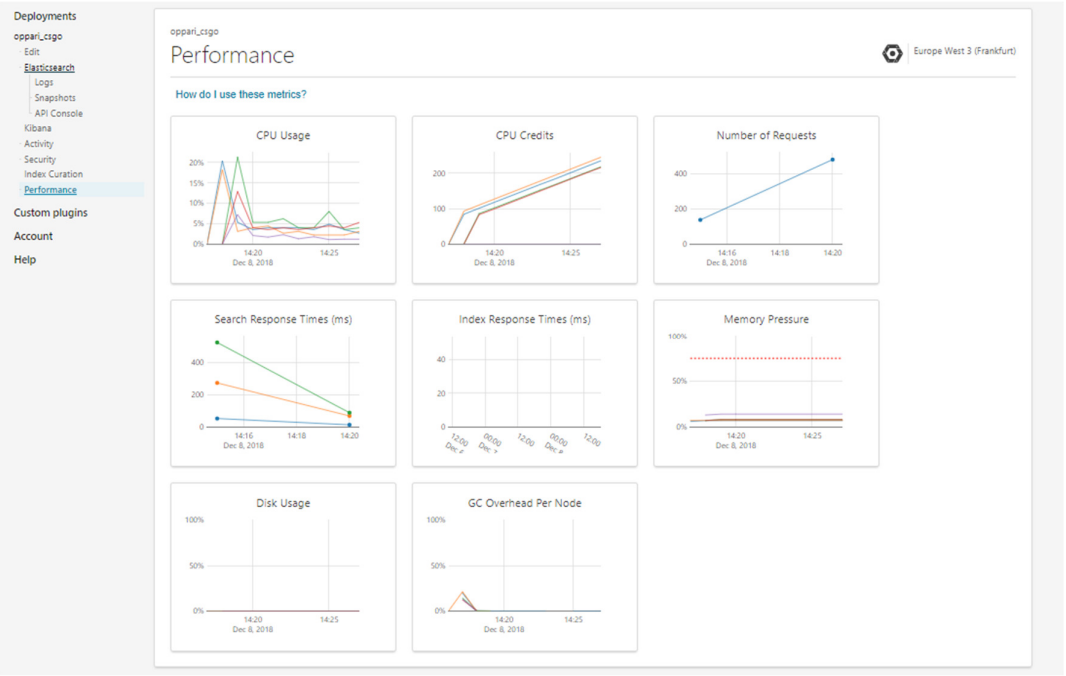


Kuva 14. Index Curation -sivu.

Indeksin siirto kuumasta solmusta lämpimään solmuun voitiin määritellä tuntitasolla.

4.2.9 Performance

Performance-sivulta voitiin nähdä koko julkaisun suorituskyvyn monitorointinäkymä.



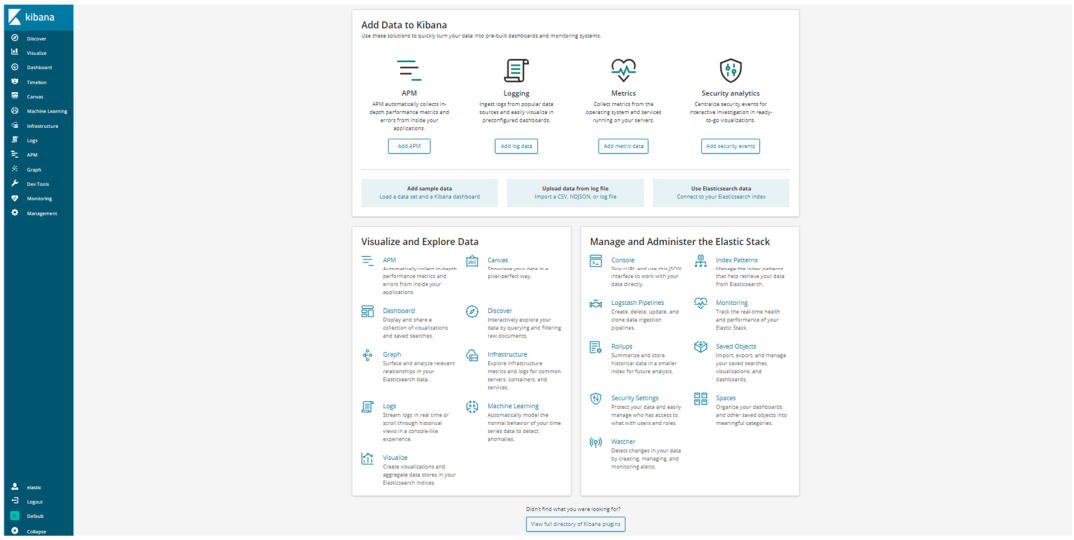
Kuva 15. Julkaisun suorituskyvyn mittarit.

Performance-sivun avulla voitaisiin tutkia, tarvitseeko instanssien laitteisto säätämistä.

4.3 Data Kibanassa

4.3.1 Yhteys Kibanaan

Varmistettiin Kibanan olemassaolo menemällä julkaisun yleissivun tarjoaman linkin kautta Kibanan-sivulle. Sivun kenttiin annettiin julkaisulle generoidut ”Username” ja ” Password”, minkä jälkeen painettiin ”Login”-nappulaa. Tästä aukesi Kibanan aloitussivu.



Kuva 16. Kibanan etusivu.

Näin varmistuttiin, että toimiva Kibana löytyi julkaisustamme. Seuraavaksi tarvitsimme Elasticsearchiin dataa, jotta Kibanaa voitiin tutkia enemmän.

4.3.2 Datan lähetys Elasticsearchiin

Insinöörityössä käytettiin CS:GO kollektiivi de_liverin yhdeltä harjoituspäivältä kerättyjä pelidatoja. Data oli kerätty yhden de_liver-pelaajan toimesta CSGO Manager -nimisen ohjelman avulla. CSGO Manageriin ladattiin pelattujen pelien dem-tiedostot, josta CSGO Manager laski erilaista statistiikkaa ja muodosti statistiikasta json-tiedoston jokaista peliä kohti.

CSGO Managerin muodostamat json-tiedostot olivat erittäin laajoja. Pelin pituudesta riippuen json-tiedosto sisälsi noin 100 000 – 400 000 ”riviä” tietoa, josta erittäin suuri osa oli insinöörityön kannalta epärelevanttia. Tästä syystä tehtiin Python-koodi, millä otettiin vain osa tiedoista. Vain tarvittavat tiedot lähetettiin saman koodin avulla Elasticsearchiin.

Koska CSGO Manager laittoi tiedostojen aikaleimaksi sen hetken, kun ohjelma oli analysoinut datat eikä itse pelihetkeä, muutettiin datasta pelien päivämäärä ja kellonajat, jotta kaikki datat eivät osuisi samaan ajankohtaan.

Pythonskripti luki kaikki määritetyssä kansiossa olleet json-tiedostot ja kävi läpi json-tiedoston pelaajaobjektit läpi yksitellen. Jos objektin pelaajan nimi löytyi de_liverin-pelaajalistasta, niin kyseisestä pelaajaobjektista kerättiin halutut tiedot omaksi json-tiedostoksi. Tämä json-objekti lähetettiin lopuksi dokumentiksi insinöörityössä tehdyn Elasticsearchin indeksiin ”csgo”.

```
import json, os
from elasticsearch import Elasticsearch
import certifi

auth = ('elastic', 'salasana')
es = Elasticsearch("ElasticsearchinOsoite", http_auth=auth, use_ssl=True,
cs_certs=certifi.where())
b=0
pelaajalista= ['turisti','batari', 'Laamaah', 'dEVI', 'jalopeura', 'T_Danger']

for filename in os.listdir('/home/oppari/json'):

    if filename.endswith(".json"):
        with open(filename) as jsoni:
                data = json.load(jsoni)

                for i in range(0,10):
                        if data['players'][i]['name'] in pelaajalista:

                                datab = {}
                                datab['date']= data['date']
                                datab['map_name'] = data['map_name']
                                datab['name']= data['players'][i]['name']
                                datab['kill_count'] = data['play-
ers'][i]['kill_count']
                                datab['score'] = data['players'][i]['score']
                                datab['assist_count'] = data['play-
ers'][i]['assist_count']
                                datab['trade_kill_count'] = data['play-
ers'][i]['score']
                                datab['trade_death_count'] = data['play-
ers'][i]['trade_death_count']
                                datab['5k_count'] = data['play-
ers'][i]['5k_count']
                                datab['4k_count'] = data['play-
ers'][i]['4k_count']
```

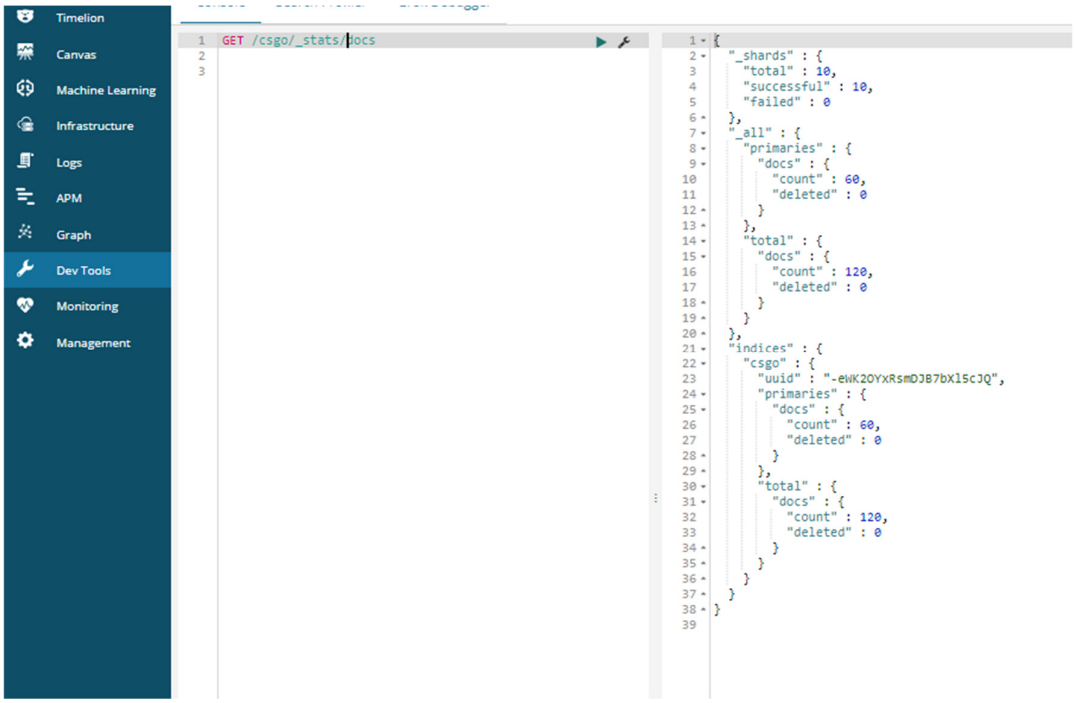
```
                                datab['3k_count'] = data['play-
ers'][i]['3k_count']
                                datab['2k_count'] = data['play-
ers'][i]['2k_count']
                                datab['1k_count'] = data['play-
ers'][i]['1k_count']
                                datab['1k_count'] = data['play-
ers'][i]['1k_count']
                                datab['kd'] = data['players'][i]['kd']
                                datab['hltv_rating'] = data['play-
ers'][i]['hltv_rating']
                                datab['entry_kill_won_count'] = data['play-
ers'][i]['entry_kill_won_count']
                                datab['entry_kill_loss_count'] = data['play-
ers'][i]['entry_kill_loss_count']
                                datab['entry_hold_kill_won_count'] =
data['players'][i]['entry_hold_kill_won_count']
                                datab['entry_hold_kill_loss_count'] =
data['players'][i]['entry_hold_kill_loss_count']
                                datab['clutch_loss_count'] = data['play-
ers'][i]['clutch_loss_count']
                                datab['clutch_won_count'] = data['play-
ers'][i]['clutch_won_count']
                                datab['total_damage_health_done'] =
data['players'][i]['total_damage_health_done']
                                datab['total_damage_armor_done'] = data['play-
ers'][i]['total_damage_armor_done']
                                datab['total_damage_health_received'] =
data['players'][i]['total_damage_health_received']
                                datab['total_damage_armor_received'] =
data['players'][i]['total_damage_armor_received']
                                datab['average_health_damage'] = data['play-
ers'][i]['average_health_damage']
                                datab['kill_per_round'] = data['play-
ers'][i]['kill_per_round']
                                datab['assist_per_round'] = data['play-
ers'][i]['assist_per_round']
                                datab['death_per_round'] = data['play-
ers'][i]['death_per_round']
                                datab['avg_time_death'] = data['play-
ers'][i]['avg_time_death']

                                json_data = json.dumps(datab)
                                c = str(b)
                                es.index(index="cs", doc_type='_doc', id=b,
body=json_data)
                                b = b + 1
```

Python-koodi ajettiin.

4.3.3 Datalähetyksen varmistus Kibanassa

Jotta voitiin varmistaa halutun datan olevan Elasticsearchissa, käytettiin Kibanan Dev Toolsia lähettämään dokumenttitila API-pyyntö luodulle csgo-indeksille.



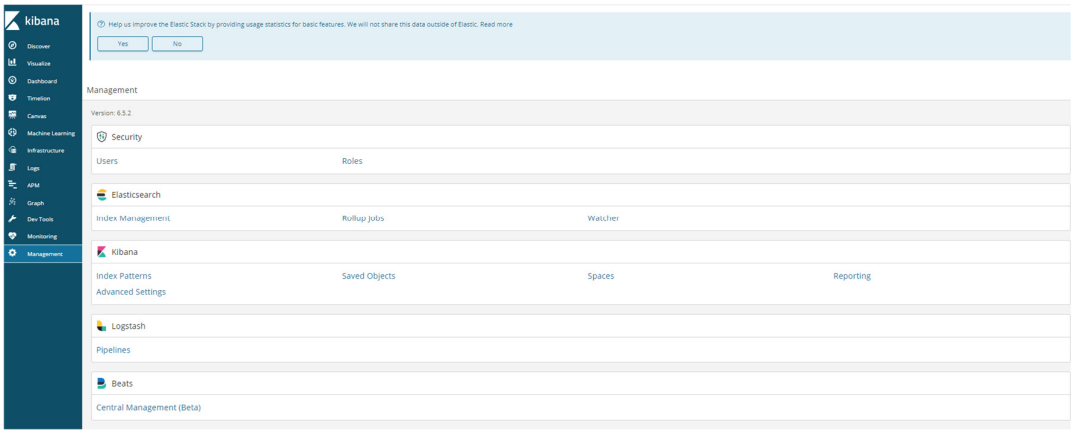
Kuva 17. Elasticsearch API -kysely Dev Toolsin avulla.

Vastauksesta varmistuttiin, että kaikki data oli mennyt perille.

4.3.4 Oletusarvoindeksin luonti

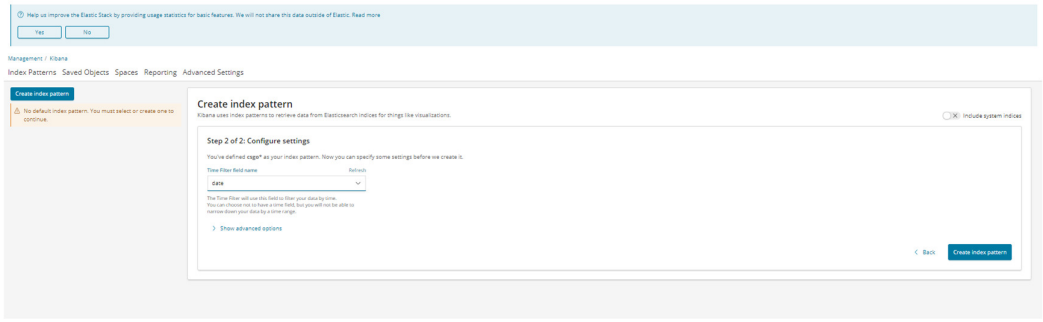
Jotta dataa pystyttiin katsomaan Kibanan Discovery-lehdellä, oli Kibanaan luotava oletusarvoindeksi. Oletusarvo-indeksiä Kibana käyttää automaattisesti oletusarvoisena indeksinä dataa hakiessaan.

Oletusarvo-indeksin luomiseksi oli mentävä Kibanan välilehdelle Mananagement – Kibana – Index Patterns.



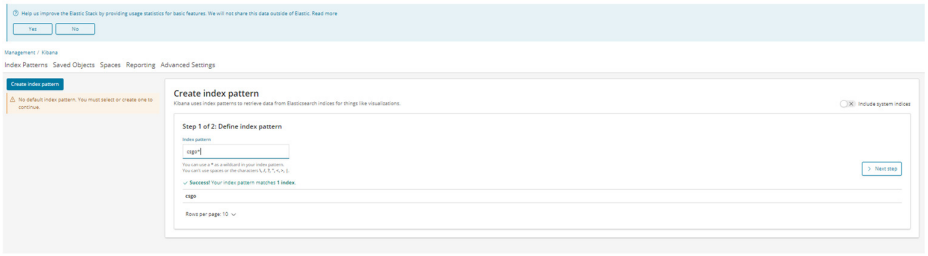
Kuva 18. Kibanan Management-sivu.

Tältä sivulta päästiin luomaan indeksikaava. Kaavaksi valittiin ”csgo*”. ”*”-merkkillä tarkoitetaan, että kaikki merkit csgo-indeksin perässä luetaan osaksi indeksikaavaa. Jos esimerkiksi tulevaisuudessa toteutukseen haluttaisiin luoda indeksi ”csgo-Pelipäivä2”, lukeutuisi se mukaan ”csgo*”-indeksikaavioon.



Kuva 19. Indeksikaavan luonti.

Seuraavassa askeleessa valittiin indeksikaavan aikakentäksi ”date”-kenttä, minkä avulla Kibana osaa pitää dokumentteja aikajärjestyksessä ja mahdollistaa niiden hakemisen aikavälien avulla.

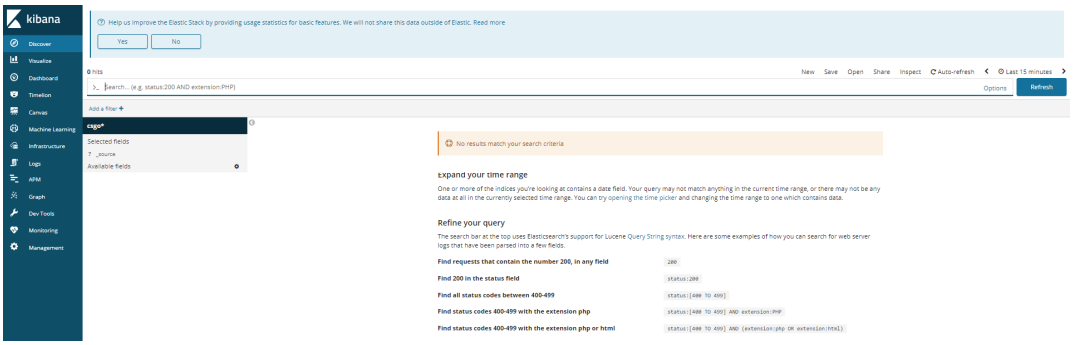


Kuva 20. Indeksikaavan luonti osa 2.

Tämän jälkeen indeksikaavio oli luotu. Koska luotu indeksikaavio oli ainut luotu indeksikaavio, niin se valittiin automaattisesti oletusarvoiseksi.

4.3.5 Discovery

Kun oletusarvoindeksi oli luotu, voitiin dataa tarkastella paremmin Kibanan Discovery-sivulla.

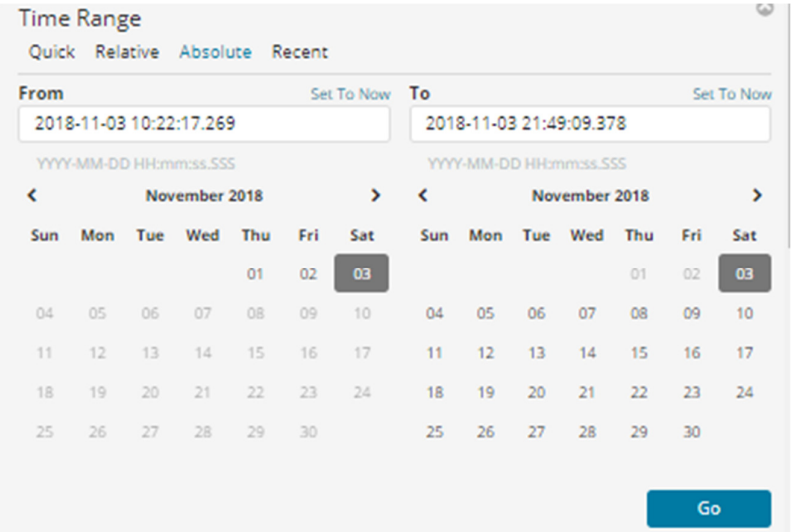


Kuva 21. Discovery-sivu ilman löydettyjä dokumentteja.

Vasemmasta reunasta voitiin nähdä valittu indeksikaavio ”csgo*”. Kuitenkaan indeksin dokumentteja ei vielä nähty, koska indeksikaaviossa oltiin valittu ”date”-kenttä järjestämään dokumentteja. Sivun oikeasta yläkulmasta nähtiin, että tarkasteltiin ainoastaan viimeisen viidentoista minuutin sisällä olevia dokumentteja.

Jotta dokumentit saatiin näkyviin Discovery-sivulla, jouduttiin dokumenttien etsimisen aikaväliä muuttamaan. Tämä tapahtui klikkaamalla sivun oikeaa yläkulmaa, mistä avautui

”Time Range”-kenttä. Koska dokumenttien tarkka päivä ja aika tiedettiin, voitiin dokumentteja etsiä suoraan kyseisellä päivämäärällä ”Absolute”-kentän kalenterin avulla.

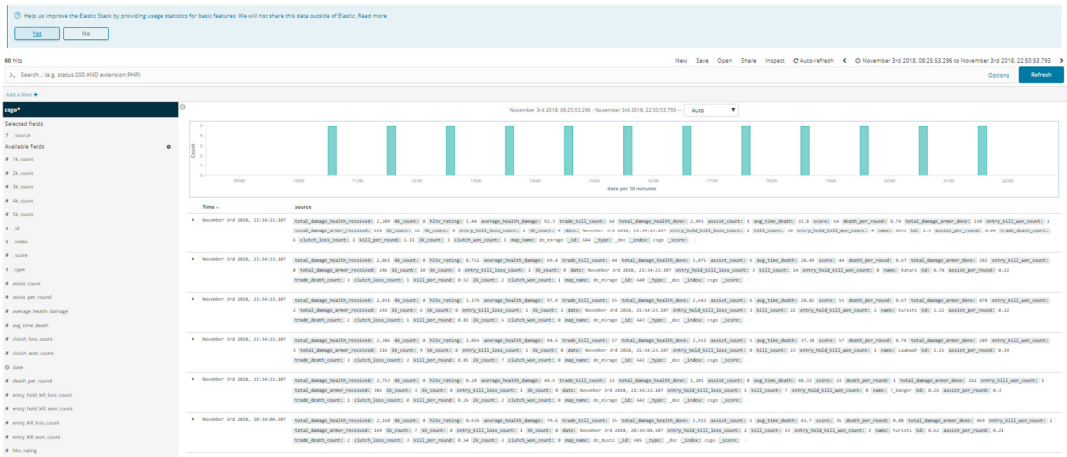


Kuva 22. Discoveryn Time Range Absolute -hakuikkuna.

Toinen tapa etsiä dokumentteja olisi ollut valita Time Range -kentästä ”Quick” ja etsiä dokumentteja esimerkiksi viimeisen 3 kuukauden ajalta. Tällä tavalla Discovery-sivun histogrammi ja document-pöytä olisivat näyttäneet, miltä kohdin viimeistä kolmea kuukautta dokumentit olisivat löytyneet.

Kuva 23. Discoveryn Time Range Quick -hakuikkuna.

Näin kaikki Elasticsearchin lähetetyt dokumentit löytyivät Discoverysta.

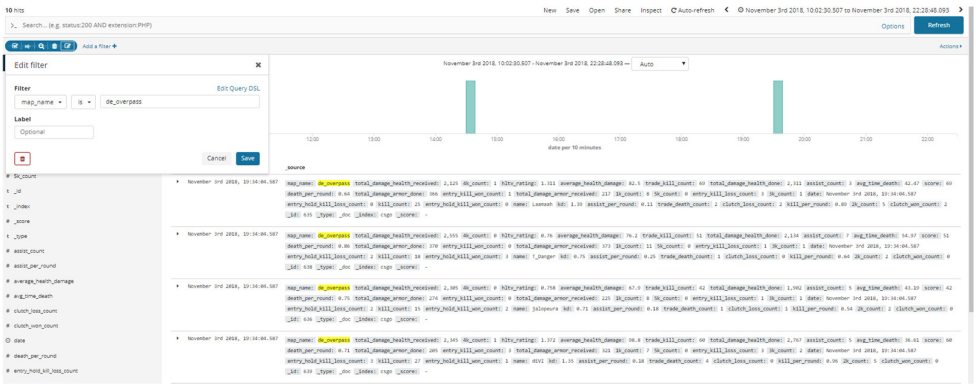


Kuva 24. Discovery löydettyjen dokumenttien kanssa.

Discoveryssa voitiin tarkastella yksittäisiä dokumentteja avaamalla ne dokumentin vierestä löytyvästä nuolesta.

Kuva 25. Dokumentin lähikuva Discoveryssa.

Sekä tehdä filtteröintejä esitettäville dokumenteille.



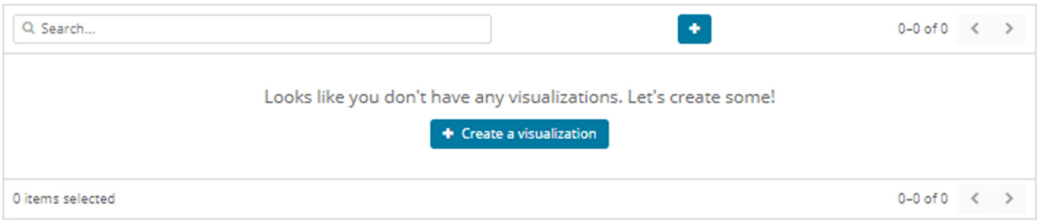
Kuva 26. Kibanan dokumenttien filtteröinti-ikkuna. Filtteröinti tehty map_name-parametrin mukaan.

Oikeat datat olivat löytyneet ja näyttivät olevan oikealla tavalla tallennettu Elasticsearchiin.

4.4 Datan visualisointi ja yleisnäkymän teko

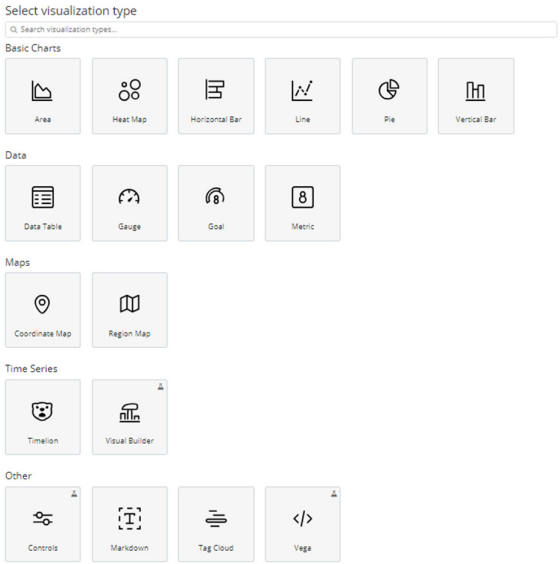
4.4.1 Visualisointien teko Kibanalla

Visualisointien teko aloitettiin menemällä Kibanan ”Visualize”-välilehdelle, josta pystyttiin aloittamaan uuden visualisoinnin tekeminen painamalla ”Create Visualization” -painiketta.



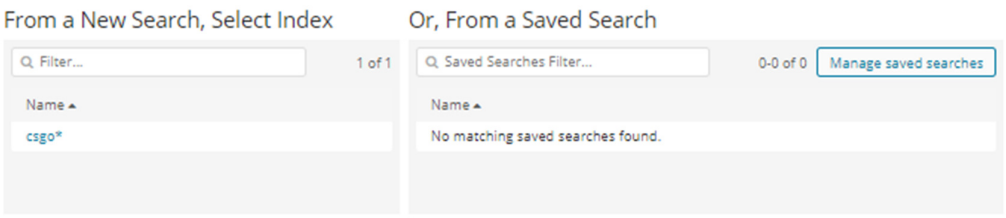
Kuva 27. Ensimmäisen visualisoinnin teko Kibanalla.

Tästä seuraavalla sivulla voitiin nähdä kaikki mahdolliset datan visualisointitavat Kibanassa. Painamalla haluttua visualisointitapaa päästiin tekemään halutunlainen visualisointi. Esimerkki visualisoinniksi valittiin viivakaavio, koska sillä saatiin aika ja päivämäärä helposti näkymään x-akselille.



Kuva 28. Kibanan visualisointivaihtoehdot.

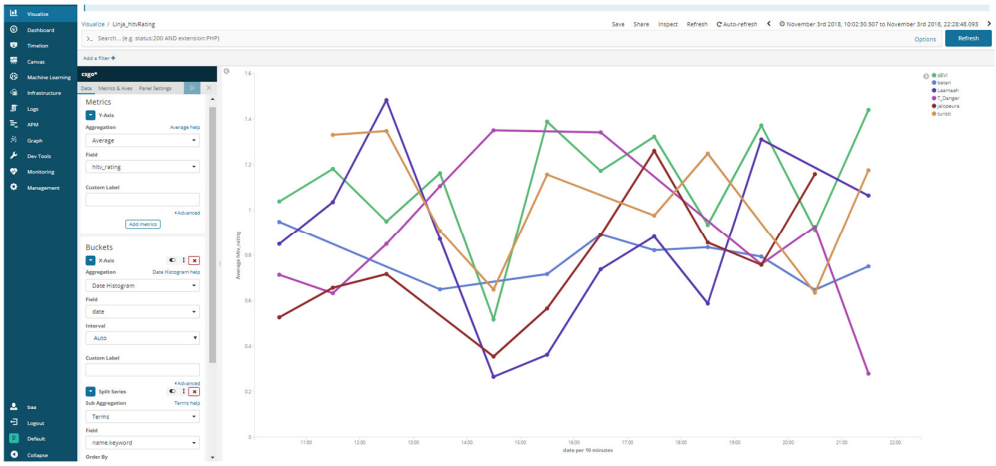
Kun oltiin painettu haluttua visualisointitapaa, pystyttiin valitsemaan, minkälaisesta hausta visualisointi tehdään. Koska valmiita hakuja ei ollut, tehtiin visualisointi ”csgo*”-indeksikaaviosta painamalla ”csgo*”-linkkiä uusien hakujen luettelosta.



Kuva 29. Visualisointiin valittavat Indeksi kaaviot ja haut.

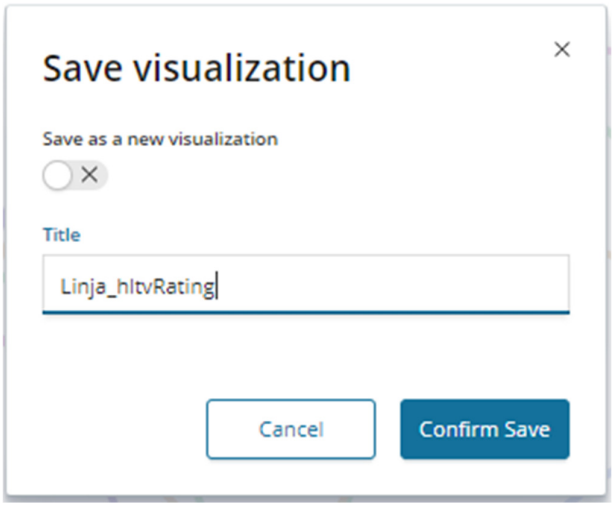
Insinöörityön esimerkki visualisoinnissa haluttiin näyttää pelaajien ”hltv-keskiarvo” aikaa vasten.

Datan Visualisoinnin teko aloitettiin valitsemalla y-akselille htlv_rating-keskiarvo. Sen jälkeen valittiin x-akselille aika ja päätettiin jakaa sarjat omiin viivoihinsa pelaajien nimen mukaan. Painamalla näiden määrittelyjen yläkulmasta löytyvää ”Play”-nappia muodostui haluttu Kaavio.



Kuva 30. Esimerkki visualisointi.

Visualisointi tallennettiin nimellä painamalla yläpalkista löytyvää ”save”-nappia.

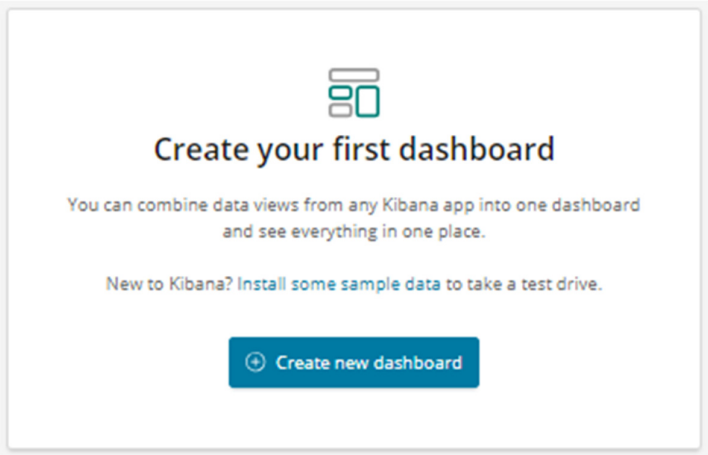


Kuva 31. Visualisoinnin tallennusikkuna.

Tämän jälkeen tehtiin vielä muutamia erilaisia visualisointeja datasta, jotta voitiin tehdä näyttävämpi yleisnäkymä Dashboardilla.

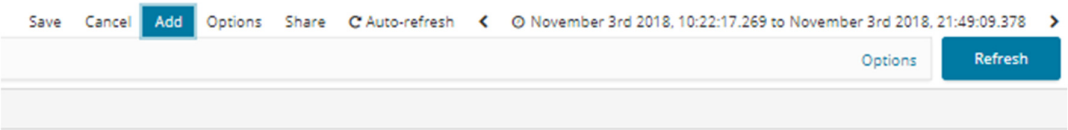
4.4.2 Dashboard

Visualisointien yleisnäkymän teko aloitettiin menemällä Kibanan-sivulle Dashboard. Sieltä avautui tervetuloilmoitus uuden dashboardin tekemisestä. Painettiin nappulaa ”Create new dashboard”.



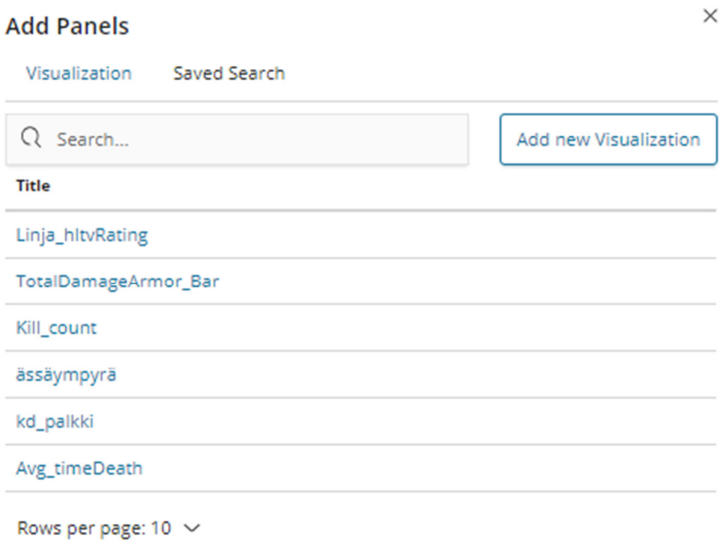
Kuva 32. Dashboardin teon aloitusikkuna.

Yleisnäkymään lisättiin aikaisemmin tehtyjä ja tallennettuja visualisointeja painamalla Dashboard-välilehden ”Add”-painiketta.



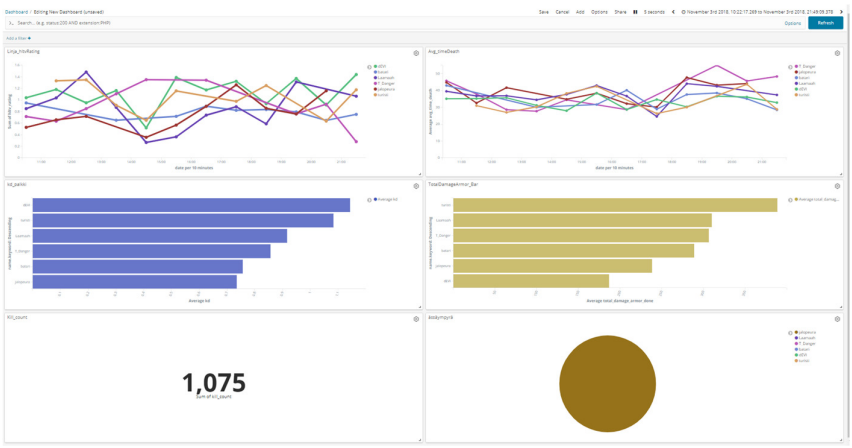
Kuva 33. Visualisoinnin lisääminen yleisnäkymään.

Tästä aukeavasta lehdestä voitiin valita halutut visualisoinnit visualisointien yleisnäkymään.



Kuva 34. Valittavat visualisoinnit.

Loppunäkymäksi saatiin kuuden visualisoinnin yleisnäkymä.



Kuva 35. Loppunäkymä.

Näin insinöörityö oli saatettu lopulliseen päämääräänsä.

5 Johtopäätökset

Insinöörityössä pyrittiin tutustumaan nopeaan Elastic Cloudin asentamiseen. Ennen insinöörityötä insinöörityön tekijä on asentanut Elasticsearch instanssinsa alusta alkaen itse virtuaalikoneelle tai infrastruktuuripilvipalveluun. Koska Elastic Cloud on Elasticsearchin kehittäjien muokkaama palvelu. Vaikutti myös siltä, että Elastic Cloudin instanssien asetukset oli optimoitu parhaita käytäntöjä noudattaen.

Insinöörityössä onnistuttiin asentamaan Elastic Cloudiin Elasticsearch ja Kibana. Lisäksi insinöörityössä saatiin onnistuneesti lähetettyä dataa Elasticsearchiin ja lopuksi saatiin tehtyä monipuolinen yleisnäkymä de_liver-kollektiivin tarjoamista datoista, joiden avulla esitettiin samalla myös Kibanan toimintoja.

Koska insinöörityö tehtiin 14 päivän ilmaiskokeilujaksolla, sille ei tullut minkäänlaista hintaa. Insinöörityön kannalta olisi ollut hyvä nähdä tämän toteutuksen ja instanssien oikea hinta. Ilmaisjaksosta riippuen myös instanssien nopea muokkaaminen oli rajallista.

Insinöörityössä opittiin Elastic Cloudin ja pilvipalveluiden nopea asennus. Insinöörityön tekemisen ohessa opittiin lisää Elasticsearchin toiminnasta ja asetuksista sekä yleisesti tietokantojen toiminnasta. Lisäksi opittiin toimimaan pilvipalveluympäristössä.

Insinöörityö antaa hyvän pohjan tutustua Elastic Cloudiin, Elaticsearchiin ja Kibanaan. Insinöörityössä on kaikki valmiina Elastic Cloud -julkaisun asentamisesta, json tiedon lähetyksestä Elasticsearchiin ja sen tutkimiseen Kibanalla. Insinöörityötä voidaan skaalata erittäin suuriin datamääriin tai reaaliaikaiseen datavirtaan erittäin helposti lisäämällä vain kapasiteettia instansseihin.

Elastic Cloudin avulla saa toimivan ja hyvin konfiguroidun Elasticsearch-hakumoottorin erittäin nopeasti toimintakuntoon.

Lähteet

1 Kankare, Ville. 2017. Mikä on pilvipalvelu? Verkkoaineisto <https://yksityisille.hub.elisa.fi/mika-on-pilvipalvelu/>Luettu 15.11.2018.

2 Wikipedia. 2017. Verkkoaineisto <https://fi.wikipedia.org/wiki/Pilvilaskenta> Luettu 15.11.2018.

3 Eronen, Heidi. 2016. IaaS, PaaS, SaaS? Mikä pilvipalvelu sopii yrityksellesi. Verkkoaineisto. <*https://blog.planeetta.net/iaas-paas-saas*>. 15.3.2016. Luettu 15.11.2018.

4 Turco, Kyle. 2014. 4 Ways Cloud Computing Can Save Your Company Money. Verkkoaineisto <https://technologyadvice.com/blog/marketing/4-ways-cloud-computing-can-save-money/> Luettu 1.6.2018.

5 Tieto. 2018. Millainen pilvipalvelu on sinulle paras? Verkkoaineisto <https://www.tieto.fi/palvelut/infrastruktuuriratkaisut/pilvi-kapasiteetti-ja-infrastruktuuripalvelut/millainen-pilvipalvelu-on-sinulle-paras>. Luettu 15.11.2018.

6 Hansen Steven. 2018. 10 Benefits and Advantages of Cloud Computing. Verkkoaineisto. <https://hackernoon.com/10-benefits-and-advantages-of-cloud-computing-3c20c7433814>. 13.10.2018. Luettu 12.16.2018

7 Smith Jason. 2018. 11 Benefits of Cloud Computing. Verkkoaineisto. <https://dotcms.com/blog/post/11-benefits-of-cloud-computing>. 16.4.2018. Luettu 16.12.2018

8 LevelCloud. 2018. Advantages and Disadvantages of Cloud Computing. Verkkoaineisto. <https://www.levelcloud.net/why-levelcloud/cloud-education-center/advantages-and-disadvantages-of-cloud-computing/>. Luettu 16.12.2018

9 Salesforce. 2018. 12 Benefits of cloud computing. Verkkoaineisto. <https://www.salesforce.com/hub/technology/benefits-of-cloud/>. Luettu 16.12.2018

10 Elasticsearch B.V. 2018. Centrally Orchestrate a Fleet of Elastisearch Cluster. Verkkoaineisto. <https://www.elastic.co/products/ece>. Luettu 6.12.2018.

11 Wikipedia. 2018. Wikipedia Verkkoaineisto <https://en.wikipedia.org/wiki/Elasticsearch> Luettu 1.12.2018.

12 Elasticsearch B.V. 2018. Getting Started. Verkkoaineisto. https://www.elastic.co/guide/en/elasticsearch/reference/current/getting-started.html. Luettu 1.12.2018.

13 Elasticsearch B.V. 2018. Search Engines. Verkkoaineisto. <https://db-engines.com/en/article/Search+Engines> Luettu 3.12.2018.

14 Elasticsearch B.V. 2018. Basic Concepts. Verkkoaineisto <https://www.elastic.co/guide/en/elasticsearch/reference/current/getting-started-concepts.html>. Luettu 3.12.2018.

15 Bennacer Samir. 2017. “Hot-Warm” Architecture in Elasticsearch 5.x. Verkkoaineisto. <https://www.elastic.co/blog/hot-warm-architecture-in-elasticsearch-5-x>. Luettu 16.12.2018

16 Elasticsearch B.V. 2018. Mapping. Verkkoaineisto. <https://www.elastic.co/guide/en/elasticsearch/reference/current/mapping.html>. Luettu 3.12.2018.

17 Elasticsearch B.V. 2018. Field datatypes. Verkkoaineisto. <https://www.elastic.co/guide/en/elasticsearch/reference/current/mapping-types.html> Luettu 6.12.2018.

18 Berman, Daniel. 2017. Elasticsearch API 101. Verkkoaineisto. <https://logz.io/blog/elasticsearch-api/>. Luettu 6.12.2018.

19 Elasticsearch B.V. 2018. Snapshot And Restore. Verkkoaineisto. <https://www.elastic.co/guide/en/elasticsearch/reference/current/modules-snapshots.html>. Luettu 16.12.2018.

20 Elasticsearch B.V. 2018. Introduction. Verkkoaineisto. <https://www.elastic.co/guide/en/kibana/current/introduction.html>. Luettu 6.12.2018.

21 Elasticsearch B.V. 2018. Console. Verkkoaineisto. <https://www.elastic.co/guide/en/kibana/current/console-kibana.html> Luettu 6.12.2018.

22 Elasticsearch B.V. 2018. Discover. Verkkoaineisto. <https://www.elastic.co/guide/en/kibana/current/discover.html> Luettu 6.12.2018.

23 Elasticsearch B.V. 2018. Visualize. Verkkoainesto. <https://www.elastic.co/guide/en/kibana/current/visualize.html>. Luettu 6.12.2018.

24 Elasticsearch B.V. 2018. Visualize. Verkkoainesto. <https://www.elastic.co/guide/en/kibana/current/createvis.html>Luettu 6.12.2018