

Lyhyt selvitys palvelun käyttöönotosta:

Palvelulle pitäisi ensinnäkin hankkia oma kotisivutila joltakin palvelimelta. Toisaalta, jos kyseessä on avoimesti opiskelija-aktivistien tekemä kampanja, voitaneen kampanjaa pyörittää myös Aallon palvelimilta käsin. Ammattimaisempaa on kuitenkin vuokrata kampanjalle erillinen kotisivutila. Käyttöönottoon liittyy tietenkin myös domainin ostos.

Palvelun käyttöönotossa suurin haaste on tietojen turvallinen tallentaminen. Suurimmaksi osaksi sivustolla ei käsitellä kovin arkaluontoista tietoa: kokeiluversiossa kuvia jakaneiden nimeä ei tallenneta, ja muutenkaan kuvakilpailun aihe ei kannusta paljastamaan kuvissa mitään kovin henkilökohtaista. Kilpailuun kuitenkin osallistutaan antamalla oma sähköpostiosoite kuvan lähettämisen yhteydessä, ja tämän arkaluontoisen tiedon varastointi lisää sivuston tietokannalle olennaisia tietoturvasuoritusvaatimuksia. Niinpä tiedon tallentaminen vaivaisena JSON-stringinä LocalStorageen ei ole missään nimessä vaihtoehto, sillä tiedot jäävät sinne määrättömäksi ajaksi, kunnes ne päällekirjoitetaan tai poistetaan, ja lisäksi ylimääräisen henkilön on verrattain helppoa päästä kyseiseen taltioon käsiksi sivun lähdekoodia muokkaamalla. SessionStorage puolestaan on samojen tietoturvasuoritusvaatimusten lisäksi kelvoton muihin kuin testaustarkoituksiin myös siksi, että nimensä mukaisesti se säilöo tietoja vain aukiolevan istunnon ajan.

Toteutuessaan palvelun tulisi siis muistaa kilpailuun osallistujat sekä heidän lähettämänsä kuvat tykkäyksineen koko kisan aukiolon ajan. Lisäksi kilpailuun eri laitteilla ja selaimilla osallistuneiden pitää pystyä näkemään toistensa lähettämä sisältö. Tarvitaan siis jonkin sortin selaimen ulkoinen tietokanta; tällainen on esimerkiksi MyJSONin kaltainen pilvipohjainen palvelu. Myös tämä keino on kuitenkin huolimattomasti käytettynä altis esimerkiksi lähdekoodin scriptien muokkaamisen kautta tehdyille urkinnoille.

Oman haasteensa juuri tämän kyseisen palvelun oikeaan käyttöönottoon asettaa se, että palvelu keskittyy kuvien jakamiseen. Teoriassa kuvatiedosto on mahdollista purkaa binäärilukujonoksi ja säilöä osana JSON-merkkijonoa, mutta tämä on hidasta, raskasta ja altista virheille. Niinpä vaihtoehto on epärelevantti, erityisesti, koska kyseessä on ennen kaikkea juuri kuviin keskittyvä palvelu – kuvien säilömisestä on tapahduttava sujuvasti. Niinpä kuvien vastaanottamiseen ja tallentamiseen tarvitaan erillistä tallennustilaa, joka sopii muuhunkin kuin JSON-jonojen varastointiin.

Palvelun pyörittäminen vaatii siis erillisen, turvallisen tietokannan. Ilmeisesti esimerkiksi MySQL on varsin laajalti käytetty, myös vapaan ohjelmiston lisenssillä käytettävä tietokantaratkaisu, mutta tämän käyttö vaatii jo tämän kurssin laajuuden ylittävää ymmärrystä ja esimerkiksi Pythonin opettelua. Tietokannalle on varattava palvelimelta huomattavasti muistikapasiteettia, sillä kuvat ovat yleensä varsin suuria tiedostoja.

Kenties kampanjaan kuuluvan kilpailun perustamiseen olisi siis ketterintä tehdä kokonaan jollakin valmiilla kotisivualustalla. Esimerkiksi WordPressille on saatavissa erilaisia kuvanjakopalvelu- ja äänestysliitännäisiä, joiden avulla kilpailusivun voisi rakentaa. Ratkaisu on todennäköisesti koodillisesti vielä sotkuisempi, mutta näissä ainakin tietokannan hallinta on ulkoistettu ja mukavan käyttöliittymän päässä.

Kuvanjakopalvelun kehittämisessä olisi tärkeää ratkaista myös moderointi. Olisi itse asiassa mielenkiintoista selvittää, millaisia keinoja loukkaavan tai muutoin vahingollisen sisällön suodattamiseen on kentällä käytössä. Facebookin ja Instagramin yliherkät sensuurialgoritmit ovat surullisenkuuluisia esimerkkejä, mutta miten asia oli ratkasitu pienemmissä kuvanjakamispalveluissa? Tämän kokoisen palvelun moderointi lienee juuri ja juuri mahdollista tehdä edelleen ihmisvoimin. Moderoinnin ja käyttäjän katumapäälle tulemisen kannalta olennaista jatkokehitystä olisi myös lisätä palveluun poista-nappi.

Kaiken kaikkiaan yllättävää, miten haastavaa kuvien lisääminen mukaan yhtälöön tekee palvelun suunnittelusta. Tästä syystä olemme demossamme käyttäneet valmista kuvapankkia, josta voi valita kuvan. Jatkokehitystä tosiaan vaatii sellaisen toiminnallisuuden laatiminen, että palvelu aidosti ottaisi vastaan käyttäjän laitteeltaan lataaman kuvan ja tallentaisi sen tietokantaan uudelleennimeten ja kytkien sen vastaavaan JSON-rakenteeseen luotuun GridElement-olioon.

Olemme käyttäneet kolmannen osapuolen koodia kahden ponnahdusikkunan ja kuvagridin pohjana, mutta varsinkin jälkimmäistä on muokattu varsin raskaalla kädellä.