



VAASAN AMMATTIKORKEAKOULU
UNIVERSITY OF APPLIED SCIENCES

Olli Hanhimäki

Verkkosovellus tulosgraafiikoiden luomiseen

Case: Frisbeegolf

Liiketalous
2017

TIIVISTELMÄ

Tekijä	Olli Hanhimäki
Opinnäytetyön nimi	Verkkosovellus tulosgrafiikoiden luomiseen Case: Frisbeegolf
Vuosi	2017
Kieli	suomi
Sivumäärä	29 + 1 liite
Ohjaaja	Kenneth Norrgård

Tässä opinnäytetyössä käsittelen frisbeegolfin tulosgrafiikoiden luomiseen tarkoitettua verkkosovelluksen suunnittelua ja toteutusta. Työ toteutettiin HTML5:n canvas-elementin ja JavaScriptin avulla. Työssä kerron myös käytettävistä tekniikoista ja esittelen käytetyt JavaScript-kirjastot.

Tavoitteena oli toteuttaa sovellus joka helpottaa tulosgrafiikoiden tekemistä ja näin nopeuttaa videoiden jälkieditointia. Sovellukseen lisättiin maltillisesti ominaisuuksia, mutta tarjottiin työkalut myös vaativammalle käyttäjälle.

Verkkosovelluksen suunnitteluosuudessa rajasin ensin sovellukseen tulevat ominaisuudet analysoimalla muiden tulosgrafiikoita. Kun ominaisuudet oli päätetty, mietin minkälainen käyttöliittymä niille tarvitaan. Lopuksi suunnittelin ulkoasun vaatimusten pohjalta.

Toteutusosuudessa kerron, kuinka ensin tein ulkoasun vaatimusten pohjalta ja tämän jälkeen aloin toteuttaa toiminnallisuutta. Toiminnallisuuden toteutuksesta kerron, kuinka eri elementtien lisäämisen, tulosten kirjoituksen ja kuvien prosessoinnin funktiot toimivat.

Työ onnistui hyvin ja sovelluksesta tuli toimiva. Käytetyt tekniikat voidaan todeta hyvin toimiviksi ratkaisuksiksi projektin suorittamiseen.

ABSTRACT

Author	Olli Hanhimäki
Title	A Web App for Creating Score Graphics Case: Disc Golf
Year	2017
Language	Finnish
Pages	29 + 1 Appendix
Name of Supervisor	Kenneth Norrgård

In this thesis designing and developing a web app for creating score graphics for disc golf videos was studied. The web app was built with a HTML5 canvas element and JavaScript. Also different techniques and JavaScript-libraries used in development were introduced.

The goal was to develop an app to ease the making of score graphics and that way speed up the editing process. To the web app I built a moderate amount of features but also added tools for more demanding users.

In web app designing section I analyzed the score graphics made by others and picked the features I considered were most needed. After defining the features for the web app, I figured out what controls were needed for those. As the last step, the layout and the style for this web app were designed using defined demands.

In the development phase I first discussed the building of the front end with given demands. Functionality was built after that and in this thesis it is explained how adding elements, writing scores and processing pictures work in terms of functions.

Building the web app was successful and the web app worked as it was expected. It can be stated that the chosen techniques were well-working choices.

SISÄLLYS

TIIVISTELMÄ

ABSTRACT

1	JOHDANTO	3
2	OHJELMOINTIKIELET JA -TEKNIIKAT	4
	2.1 HTML	4
	2.2 CSS	4
	2.3 Material Design Lite	5
	2.4 JavaScript	5
	2.4.1 JavaScript-kirjastot	6
3	SUUNNITTELU	7
	3.1 Analysoitavat materiaalit	7
	3.2 Ominaisuudet	13
	3.2.1 Taustan luomisen työkalut	13
	3.2.2 Tiedonsyöttö, tekstin asettelu ja grafiikoiden luominen	13
	3.3 Ulkoasu	15
	3.3.1 Taustan luomisen työkalujen ulkoasu	16
	3.3.2 Tulosten syötön ja tekstin asettelun työkalujen ulkoasu	17
4	TOTEUTUS	18
	4.1 Ulkoasu	18
	4.2 Ominaisuudet	21
	4.2.1 Canvasin perustoiminnot	21
	4.2.2 Elementtien lisääminen	21
	4.2.3 Tietojen syöttö ja tekstin asettelu	22
	4.2.4 Kuvien prosessointi	23
5	TYÖN PROSESSI	25
6	YHTEENVETO	27
	LÄHTEET	29
	LIITTEET	

KUVIO- JA TAULUKKOLUETTELO

Kuvio 1. Näyttökaappaus tulosgrafiikasta(Jomez Productions, 2017)	7
Kuvio 2. Näyttökaappaus tulosgrafiikasta (Central Coast Disc Golf, 2017)	8
Kuvio 3. Näyttökaappaus tulosgrafiikasta (Central Coast Disc Golf, 2016)	9
Kuvio 4. Näyttökaappaus tulosgrafiikasta (The SpinTV, 2016).....	10
Kuvio 5. Näyttökaappaus tulosgrafiikasta (7k Frisbeegolf, 2016)	11
Kuvio 6. Näyttökaappaus tulosgrafiikasta (Prodigy Disc, 2017).....	12
Kuvio 7. Näyttökaappaus tulosgrafiikasta (Vuorelan Veikot, 2017).....	12
Kuvio 8. Mockup, työkalut kiinni.	15
Kuvio 9. Mockup, työkalu avattu.....	16
Kuvio 10. Kuva valmiista ulkoasusta.	19
Kuvio 11. Työkalun piilottamiseen käytetty koodi.....	19
Kuvio 12. Työkalu avattuna ja vieritys ominaisuus valmiina.....	20
Kuvio 13. Kuvio projektin vaiheista.	25

1 JOHDANTO

Opinnäytetyössä tavoitteenani on toteuttaa frisbeegolf-videoille tarvittavien grafiikoiden luomiseen tarkoitettu verkkosovellus lajin harrastajille. Työssä esittelen sovelluksen toteutukseen käytetyt ohjelmointitekniikat ja koodikirjastot. Tämän jälkeen tutkin ja määritän sovellukseen tarvittavat ominaisuudet ja viimeisessä vaiheessa toteutan sovelluksen.

Työn idea lähti omasta tarpeesta ja sopivan työvälineen puuttumisesta. Frisbeegolf on maailmalla kasvava laji ja Suomessa kasvu on ollut räjähdysmäistä muutaman vuoden ajan. Lisäksi nykypäivänä matkassa on aina vähintään puhelimen kamera, joten videoiden ottaminen on helppoa ja tämä näkyy videoiden määrässä. Palvelua tai sovellusta, joka tekisi saman työn jonka suunnittelemani verkkosovellus tekee yhtä helposti ja vaivattomasti ei ole löytynyt. Sovelluksen pitäisi osata tehdä joka väylän jälkeisestä tilanteesta videoon liitettävä kuva, kun sille on annettu haluttu pohja tekstien asettelulla, kierroksen tulokset sekä väylien par-luvut. Työvälineestä tehtiin verkkosovellus, koska sieltä se on helposti kaikkien saatavilla ja lisäksi palveluiden vieminen pilveen ja selaimessa suoritettavien ohjelmien mahdollisuudet kiinnostavat minua.

Sovellus rakennetaan osaksi omia sivujani. Työssä tärkeimpänä välineenä käytän Codeanywhere-pilvipalvelu ohjelmointiympäristöä. Toiminnallisuuden ohjelmalle rakennan JavaScriptillä ja muutamalla koodikirjastolla HTML:n Canvas-elementin ympärille. Ulkonäöstä vastaa HTML:n ja CSS:n lisäksi Googlen Material Design Lite sovelluskehys.

2 OHJELMOINTIKIELET JA -TEKNIIKAT

Tässä kappaleessa esittelen sovelluksen toteutukseen käytetyt kielet ja tekniikat. Sovelluksen ulkoasu toteutettiin HTML:n, CSS:n ja Material Design Liten avulla. Toiminnallisuus rakennettiin HTML:n Canvas elementin ympärille JavaScriptillä ja apuna käytettiin Jscolor, JsZip ja JsFilesaver JavaScript-kirjastoja.

2.1 HTML

Kaikki verkkosivut on kirjoitettu HTML-kielillä. Sen avulla on mahdollista muotoilla sivuston tekstiä ja lisätä kuvia ja videota - kaikki tämä tallennetaan tekstitiedostoon, jota mikä tahansa tietokone kykenee tulkitsemaan. (Castro 2007, 13.)

Nykyään monimutkaisempien tekniikoiden yleistyessä HTML:ää (Hypertext Markup Language) käytetään vähenevässä määrin koko sivuston sisällön esittämiseen, mutta sillä toteutetaan sivuston rakenne ja luodaan pohja monimutkaisemmille toiminnoille.

Kielen kehittämisestä vastaa World Wide Web Consortium, jota usein kutsutaan lyhenteellä W3C (Castro 2007, 17). Vuonna 2014 valmistui uusimman version HTML5:den suositus (w3c, 2014). HTML5 toi uusia ominaisuuksia, kuten esimerkiksi canvas-elementin, jota käytän suunnittelemani sovelluksen toiminnallisuuden toteutukseen.

2.2 CSS

CSS (Cascading Style Sheets) on kieli, joka nykyään vastaa suurelta osin sivujen ulkonäöstä eli tyylistä. Sillä määritellään ulkonäkö erilaisille HTML- elementeille.

Semanttinen eli rakenteellinen merkkkaus tarjoaa kehittäjille useita tärkeitä etuja, rakenteellisia sivuja on paljon helpompi työstää kuin ulkoasuun keskittyviä (Buddy, Moll & Collison 2007, 4.). Ennen CSS:ää HTML-koodia käytettiin ulkoasujen luomiseen ja esimerkiksi H1 tagin sijaan otsikoissa käytettiin lihavoitua ja kirjainkokoja visuaalisen efektin luomiseen. Taulukoilla luotiin sivuille sisällön asettelua, sen sijaan että niitä olisi käytetty tietojen esittämiseen. CSS:n tultua voitiin

siirtyä taas HTML-koodissa rakenteelliseen merkkaukseen ja sen jälkeen toteuttaa visuaalinen ilme CSS:llä. CSS mahdollisti myös sivuston yhtenäisen ulkonäön luomisen ja helpon muokkaamisen, koska elementtejä voidaan muotoilla suoraan ilman, että tarvitsee luoda muita tunnistimia.

2.3 Material Design Lite

Material Design on Googlen kehittämä käyttöliittymien suuntaviiva ja työkalu. Sen tavoitteena on yhdistää vanhat periaatteet hyvästä käyttöliittymäsuunnittelusta yhteen innovaation ja uuden teknologian mahdollisuuksien kanssa, sekä mahdollistaa joka laitteelle yhdenmukaisten käyttöliittymien luominen(Google, 2017).

Material Design Lite sovelluskehys mahdollistaa Material Designin käyttämisen verkkosivuilla, se käyttää CSS, JavaScript ja HTML-kieliä. Sovelluskehyksessä käytetään BEM- (Block, Element, Modifier) menetelmää luokkien nimeämiseen.

Material Design Liteä käytetään sisällyttämällä siihen kuuluvat kaksi tyylitiedostoa ja yksi JavaScript tiedosto omaan koodiin ja sen jälkeen seurataan sovelluskehysten verkkosivuilta, miten sivusto rakennetaan ja minkälaisia luokkia eri elementeille annetaan.

2.4 JavaScript

JavaScript on ohjelmointikieli, jota voi käyttää lisäämään verkkosivuille interaktiivisuutta(Negrino & Smith 2007, 2). JavaScript on asiakaspuolen ohjelmointikieli, mikä tarkoittaa sitä, että se suoritetaan käyttäjän selaimessa eikä palvelimen puolella.

Javascriptillä luodaan sivuille toiminnallisuus. Sitä voidaan käyttää esimerkiksi tarkistamaan lomakkeeseen täytetyt tiedot, esittämään päivämäärä tai hakemaan käyttäjälle hänen tarvitsemaansa tietoa.

JavaScript ei salli asiakaskoneilla tiedostojen lukemista ja kirjoittamista tai tiedostojen luomista palvelinkoneille. JavaScriptin avulla ei voida sulkea ikkunoita, joita kyseisen kielen avulla ei ole luotu. JavaScript ei kykene lukemaan toisen verk-

kosivun tietoja, jotka ovat peräisin joltain toiselta palvelimelta. (Negrino & Smith 2007, 7)

JavaScriptiksi kutsuttua skriptauskieltä on kutsuttu erilaisilla nimillä (riippuen siitä kenen tuote on ollut kyseessä) ja siitä on tehty lähes kymmenen erilaista versiota. JavaScriptin ohella on kehitetty myös JScript ja ECMAScript. JavaScriptin ensimmäinen versio (jota kutsuttiin alkujaan LiveScriptiksi) julkaistiin Netscape Navigator 2.0 -selaimen osana.

2.4.1 JavaScript-kirjastot

Jscolor on sivulle lisättävä värin valitsin komponentti, jonka tavoitteena on olla helppokäyttöinen niin ohjelmoijalle kuin myös käyttäjälle (East Desire, 2017). Se on JavaScript koodikirjasto, joka ei tarvitse toimiakseen sovelluskehysä. Se toimii kaikilla selaimilla ja on helppokäyttöinen myös kosketusnäytöllä. Komponentit sisältävät paljon kustomointivaihtoehtoja, joten niistä saa sivujen ulkonäköön hyvin sopivat. Jscolor on GNU GPL v3. lisensoitu.

JSZip on JavaScript koodikirjasto, joka mahdollistaa zip-tiedostojen luomisen, lukemisen ja muokkaamisen käyttäjän selaimessa. JSZip ei tarvitse toimiakseen muita sovelluskehysä ja toimii kaikissa uusissa selaimissa. JSZip on GNU GPL v3. ja MIT lisensoitu. (Knightley, 2017)

FileSaver.js on JavaScriptin koodikirjasto, joka tuo "Tallenna tiedosto"-ominaisuuden web-sivuihin ja ohjelmiin. Tiedostojen tallennus tapahtuu selaimen puolella. FileSaver toimii useilla eri tiedostotyypeillä. FileSaver.js ei tarvitse toimiakseen muita sovelluskehysä uudemmilla selaimilla, mutta joissakin vanhemmissa selaimissa voidaan mahdollistaa toimivuus Blob.js koodikirjastolla. FileSaver.js on MIT lisensoitu. (Grey, 2017)

3 SUUNNITTELU

Tässä kappaleessa määritellään sovelluksen tarvittavat ominaisuudet. Sovellus pyrkii täyttämään harrastelijoiden tarpeet tulosgrafiikoiden teossa, nopeuttaen videoiden editointiprosessia ja minimoi käyttäjän virheet. Sovellus tuottaa png-kuvat väylien jälkeisistä tuloksista yhtenäiselle pohjalle samalla tekstin asettelulla, tähän lisätään animaatio videon editoinnissa käytettävällä sovelluksella ja saadaan näyttävää materiaalia pienellä vaivalla.

3.1 Analysoitavat materiaalit

Lähden rajaamaan ominaisuuksia käymällä läpi, minkälaisia grafiikoita frisbeegolf-videoilla esiintyy.



Kuvio 1. Näyttökaappaus tulosgrafiikasta(Jomez Productions, 2017)

Jomez Productions on tällä hetkellä yksi parhaista frisbeegolf-videoiden tekijöistä. Jomez Productionsille tulosgrafiikat tekee Overstable Studios. Kuvion 1 tulosgrafiikka sisältää kehittyneitä animaatioefektejä, joita ei ole helppo toteuttaa ohjelman tuottamalla png-kuvilla. Tulosgrafiikan tausta on monimutkainen, sisältäen pelaajien kuvat sekä liukuvärjättyjä elementtejä. Pelaajien edellisen väylän erotus par-tulokseen erotetaan numeron värillä, vihreä on alle, harmaa par ja punainen

yli. Alhaalla näkyy pelaajien tulos suhteessa par-tulokseen tällä kierroksella kuin myös tulos yhteensä, koska kyseessä on monen kierroksen kisa. Tekstin asettelu- sa pelaajien nimet ja muut tiedot ovat keskenään samalla korkeudella, pystysuun- nassa nimi ja tulos on sivusuunnassa keskitetty ja alimmalla rivillä kierroksen ja kisan tulos, jotka ovat suurin piirtein yhtä kaukana nimen ja väylän tuloksen kes- kilinjasta. Väylänumero, par, turnauksen nimi, kierros ja väylän mitta näkyvät ylä- laidassa omassa grafiikassaan.



Kuvio 2. Näyttökaappaus tulosgrafiikasta (Central Coast Disc Golf, 2017)

Central Coast Disc Golf on pitkään frisbeegolf-videoita tehnyt ryhmä. Kesäkuussa Central Coast Disc Golf alkoi käyttää uutta kuviossa 2 esiintyvää tulosgrafiikkaa. Uusi tulosgrafiikka sisältää kehittyneitä animaatioefektejä, joita on vaikea toteuttaa suunniteltavan ohjelman png-kuvilla. Tausta on monimutkainen sisältäen pelaajien kuvat ja taustakuvan raidallisella kuosilla. Video pyörii taustalla ja siihen lisättiin blur-efekti kun tulosgrafiikka tuli esiin. Pelaajien edellisen väylän tuloksen erotus par-tulokseen esitetään ympyröillä, jotka ovat alle parin ja neliöillä, jotka ovat yli parin. Kierroksen ja kisan tulokset näkyvät väylän tuloksen oikealla puolella. Tekstin asettelussa pelaajien nimet ja muut tiedot ovat korkeudessa samalla tasolla. Pelaajan nimi on keskitetty ja sen alapuolella on väylän tulos vasemmalla ja kierros sekä kisatulokset näkyvät väylän tuloksen oikealla päällekkäin. Väylänumero, turnauksen nimi ja radan nimi ovat osana tulosgrafiikkaa.



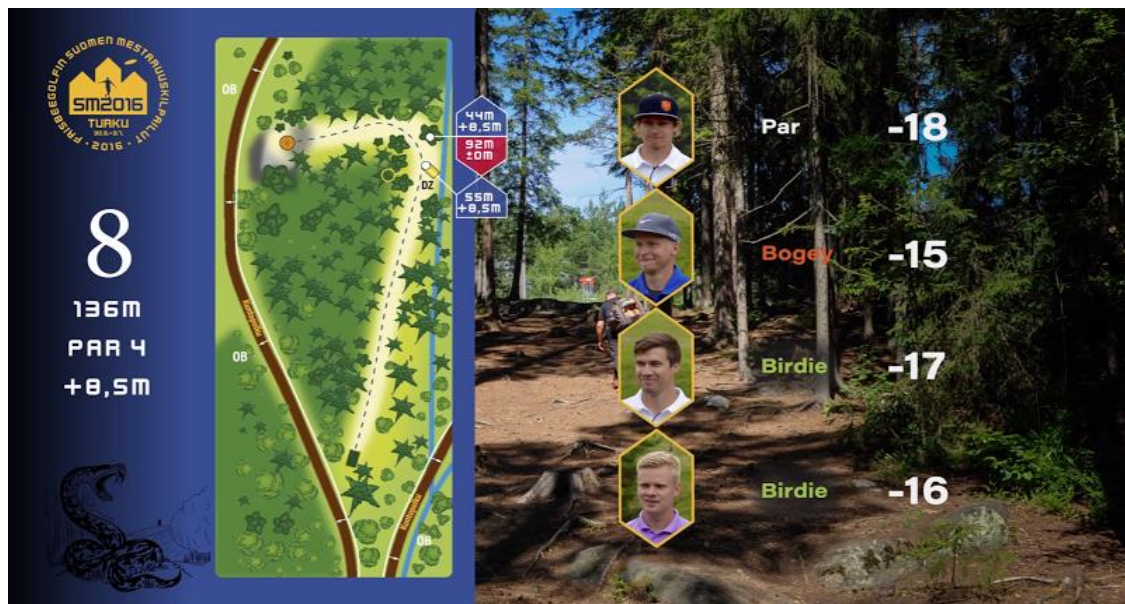
Kuvio 3. Näyttökaappaus tulosgrafiikasta (Central Coast Disc Golf, 2016)

Kuvio 3 sisältää toisen esimerkin Central Coast Disc Golf ryhmän videosta, tämä on viime vuoden maailmanmestaruuskisoista. Grafiikka ei sisällä vaikeita animaatioefektejä. Tausta ei ole hirveän monimutkainen, erilaisia laatikoita on paljon ja reunat on varjostettu. Video pyörii taas taustalla ja siihen lisätään blur-efekti ennen tulosgrafiikan näyttämistä. Pelaajien etunimi on valkoisella tekstillä ja sukunimi seuraavalla rivillä sinisellä. Väylännumero, par ja pituus ovat omalla grafiikallaan yläkulmassa.



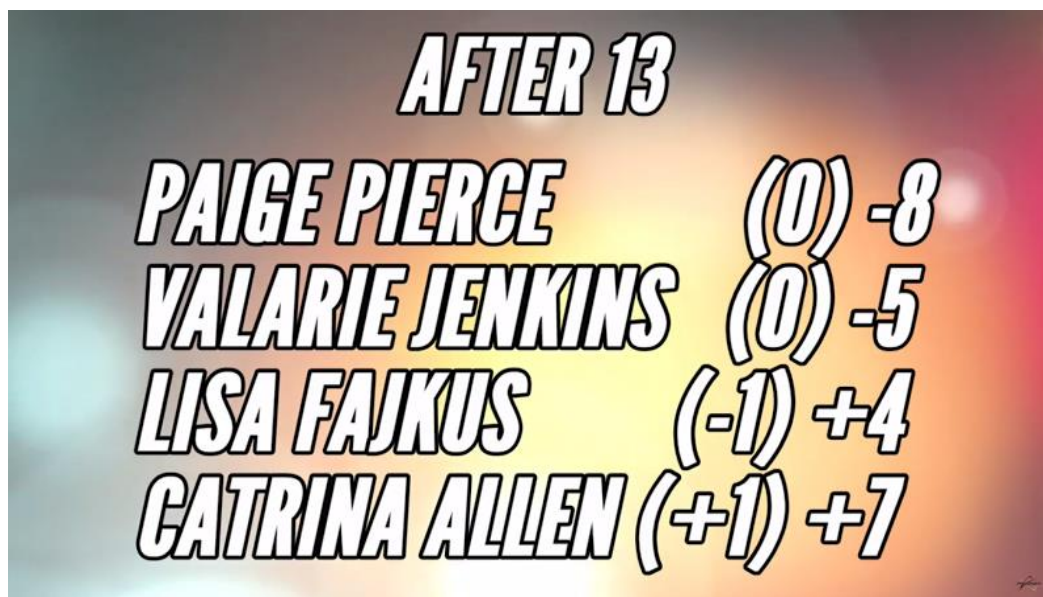
Kuvio 4. Näyttökaappaus tulosgrafiikasta (The SpinTV, 2016)

The SpinTV on tilatuin frisbeegolf-kanava. Kuvio 4:ssä on Nokialla käydyin European Open 2016 -kisan tulosgrafiikka. Tässä grafiikassa ei ole mitään animaatioita. Tausta sisältää staattisen taustakuvan ja pelaajien kuvat. Pelaajien väylän tuloksen erotus pariin esitetään värillä, vihreä on alle, valkoinen tasan ja harmaa yli. Pelaajien nimi on kahdella rivillä. Kokonaistulos tasan parin kanssa esitetään isolla e-kirjaimella, joka tulee englannin kielen sanasta “even”. Myös kierroksen ja turnauksen tuloksissa käytetään väriä osoittamaan, kuinka pelaajalla menee suhteessa pariin. Pelaajien nimen edessä on lippu, jolla osoitetaan kansalaisuus. Grafiikasta löytyy tiedot väylänumerosta, kisan nimestä, kierroksesta ja pelaavasta ryhmästä.



Kuvio 5. Näyttökaappaus tulosgrafiikasta (7k Frisbeegolf, 2016)

7k Frisbeegolf on 7k Frisbeegolf-seuran youtube-kanava. Kuviossa 5 on Suomenmestaruuskisojen tulosgrafiikasta kuva. Tässä ei ole mitään animaatioita. Taustalla on väyläkarta ja kuva väylästä. Pelaajista on kasvokuvat, mutta ei ollenkaan nimiä. Väylän tulos kerrotaan golf-termeillä ja värillä, kuviossa esiintyy birdie, joka tarkoittaa yksi alle parin, par eli tasan ja bogey on yhden yli. Kuluvan kierroksen tulosta ei näy erikseen vaan pelkästään kisan kokonaistulos suhteessa pariin. Tämä on ensimmäinen esimerkki, jossa pelaajat ovat pystysuunnassa. Väylän tuloksen kertova teksti ja kokonaistulos ovat pystysuunnassa noin kuvan keskikohdan korkeudella. Väylänumero, par, etäisyys ja kisan nimi löytyvät taustakuvan väyläkartasta.



Kuvio 6. Näyttökaappaus tulosgrafiikasta (Prodigy Disc, 2017)

Kuviossa 6 on Prodigy Discin videosta tulosgrafiikka. Taustalla on aiheeseen liittyvän esteettinen video, tekstit liukuvat ruutuun ylhäältä ja alhaalta. Pelaajat pystysuunnassa. Väylänumero toimii ikään kuin otsikkona.



Kuvio 7. Näyttökaappaus tulosgrafiikasta (Vuorelan Veikot, 2017)

Kuviossa 7 on Parifrisbeegolfin suomenmestaruuskisojen finaalikierroksen tulosgrafiikka. Valkoinen tausta ja väyläkartta. Edellisen väylän tulos ei näy, vaan ainoastaan koko kisan ja suluissa kierroksen tulos. Pelaajat ovat pystysuunnassa.

3.2 Ominaisuudet

Tässä kappaleessa rajataan sovellukseen tulevat ominaisuudet. Sovellus pidetään yksinkertaisena kuitenkin mahdollistaen tyylikkaiden tulosgrafiikoiden toteutus.

3.2.1 Taustan luomisen työkalut

Oletuksena canvasin koko on 1920*1080 eli fullhd, mutta sen kokoa pystytään muuttamaan. Canvas esitetään sivulla pienempänä, jotta sitä olisi helpompi muokata, mutta se tulostaa kuvat sen mukaan, paljonko sille on annettu koko arvoiksi. Tarvittavat syötteet ovat leveys ja korkeus.

Sovelluksella tulee pystyä tekemään erilaisia nelikulmioita. Näillä pystytään toteuttamaan yksinkertainen tausta tuloksille. Nelikulmioiden kokoa, väriä ja läpinäkyvyyttä pystytään vapaasti muuttamaan. Tarvittavat syötteet ovat Sijainti x&y, leveys & korkeus, väri ja läpikuultavuus.

Tekstin lisääminen taustaan pitää mahdollistaa. Tällä pystytään lisäämään esimerkiksi kisan tai radan nimi. Tekstin kokoa, fonttia ja sijaintia pitää pystyä muuttamaan vapaasti. Tarvittavat syötteet ovat teksti, sijainti x&y, fontin koko, fontti, väri ja läpikuultavuus.

Kuvien tuominen koneelta mahdollistaa logojen, väyläkarttojen tai vaikka valmiin pohjan tuomisen. Tuodun kuvan sijaintia ja kokoa pitää pystyä muuttamaan. Kuva pystytään liittämään muokatun pohjan päälle ja kuvan päälle pystytään muokkaamaan. Kuvan liittämiseen tarvitaan syötteiksi itse kuva, sijainti x&y, sekä leveys ja korkeus.

3.2.2 Tiedonsyöttö, tekstin asettelu ja grafiikoiden luominen

Sovellukselle annetaan väylien par, pelaajien nimet ja pelaajien tulokset. Väylien par ja pelaajien tulokset annetaan samaan tekstikenttään joko pilkuilla tai välilyönneillä erotettuina, koska näin ne saadaan helposti syötettyä tulosten merkkauksen ohjelmista ja palveluista.

Sovellukseen ei anneta väylien numeroita vaan ne tulevat automaattisesti numerosta yksi eteenpäin, vaikka tämän tiedetään harvinaisissa tapauksissa tuottavan mahdollisia ongelmia.

Tulosgrafiikan luominen mahdollistetaan yhdestä kuuteen pelaajalle. Tarvittavat syötteet ovat pelaajien määrä, parit, pelaajien nimet ja tulokset.

Pelaajien asettelu mahdollistetaan pysty ja vaakasuunnassa. Tietojen asettelu on aina riippuvainen muiden tietojen asettelusta, näin pidetään käyttöliittymä selkeänä ja mahdollistetaan käytetyin ja näyttävin tyyli. Fontteja ja fonttikokoja pystytään muuttaa nimiin, väylän tuloksiin ja kokonaistuloksiin erikseen.

Ensimmäisessä versiossa ei ole tukea kisan tulokselle vaan ainoastaan kierroksen tulos, koska harrastelijoilla harvemmin on tarvetta usean kierroksen tulos grafiikoille.

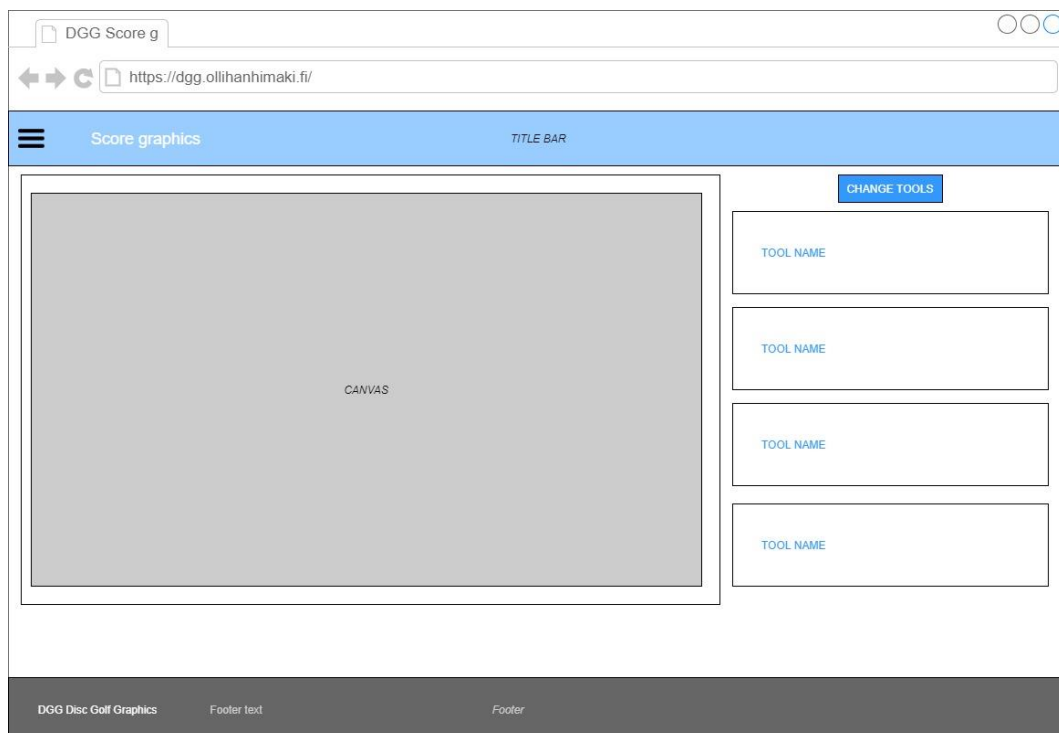
Pelaajien ollessa pystysuunnassa heidän tiedot ovat vaakasuunnassa. Pelaajien nimi on tasattu vasemmalle ja väylän tulos sekä kierroksen tulos on keskitetty. Tietojen riviväliä ja sarakkeiden väliä pystytään vapaasti muuttamaan.

Pelaajien ollessa vaakasuunnassa tiedot ovat pystysuunnassa. Pelaajan nimi, väylän tulos sekä kierroksen tulos on keskitetty. Sarakkeiden ja rivien väliä pystytään vapaasti muuttamaan. Tarvittavat syötteet ovat vaaka vai pysty- valitsin, sijainti x&y, pelaajien välinen etäisyys, nimen ja väylätuloksen välinen etäisyys, väylätuloksen ja yhteistuloksen välinen etäisyys, fontin koko, -tyyli ja -väri nimelle, väylätulokselle ja yhteistulokselle sekä valitsin väylätuloksen tulokseen perustuvalla värille.

Kun pohja on luotu, tiedot annettu ja tekstin asettelu määritelty voidaan antaa sovellukselle käsky luoda grafiikat. Sovellus luo joka väylän jälkeisen tulosgrafiikan ja tallentaa ne .png- tiedostona .zip- tiedostoon. Kun .zip-tiedosto on valmis, sovellus aloittaa sen lataamisen.

3.3 Ulkoasu

Ulkoasu toteutetaan Googlen material design litellä. Käytetään yksinkertaista ylä-tunnisteriviä joka jättää sovellukselle ison avaran tilan käytettäväksi ja mahdollistaa sivuston käyttöliittymän pysyvän selkeänä myös uusien ominaisuuksien lisäämisen jälkeen.



Kuvio 8. Mockup, työkalut kiinni.

Isolla ruudulla canvas näkyy vasemmassa reunassa ja työkalut oikealla. Työkalut jaotellaan kahteen ryhmään: taustan luomisen työkalut ja tulosten syöttö sekä asetelu työkalut. Työkaluista näkyy yksi ryhmä kerrallaan.

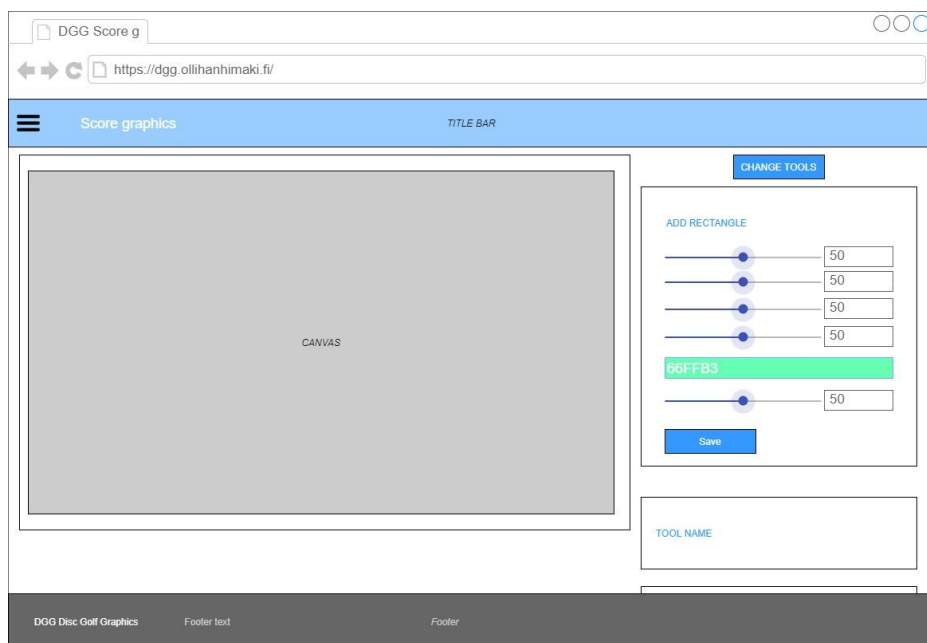
Pienemmillä ruuduilla canvas on ylälaidassa ja työkalut sen alla. Työkaluista näytetään taas yksi ryhmä kerrallaan.

Canvas sekä kaikki työkalut ovat omilla korteillaan material design -tyylin mukaisesti. Työkaluista näkyy aluksi vain otsikko, josta painamalla saadaan työkalu avattua tai piilotettua. Kuviossa 8 suunnitelmia vastaava mockup.

3.3.1 Taustan luomisen työkalujen ulkoasu

Ensimmäinen kortti on resoluution muuttaminen. Resoluutiota vaihdetaan kahdella tekstilaatikolla ja se päivittyy aina, kun tekstilaatikon tietoja muokataan. Oletuksena koko on 1920x1080 eli fullHD.

Seuraava kortti on nelikulmioiden lisääminen. Nelikulmioiden sijaintia ja kokoa muutetaan liukusäädinten ja tekstilaatikkojen avulla. Liukusäädin mahdollistaa nopean muokkauksen, tekstilaatikko taas tarkemman jäljen. Värin valinta tapahtuu Jscolorin tekstilaatikko värivalitsimella ja läpikuultavuus liukusäätimellä.



Kuvio 9. Mockup, työkalu avattu.

Seuraava kortti on tekstin lisääminen taustaan. Haluttu teksti syötetään tekstilaatikkoon. Sijainti valitaan kahdella liukusäätimen ja tekstilaatikon yhdistelmällä. Koko valitaan tekstilaatikolla ja fontti valitaan monivalintaelementillä. Väri valitaan Jscolorin tekstilaatikko värivalitsimella ja läpikuultavuus liukusäätimellä.

Valmiiden kuvien tuominen on seuraava kortti. Kuvan tuonti tapahtuu painamalla napista, joka avaa tiedostovalitsimen. Kun kuva on tuotu, sen sijaintia ja kokoa pystytään muuttamaan liukusäätimillä.

3.3.2 Tulosten syötön ja tekstin asettelun työkalujen ulkoasu

Ensimmäinen kortti on tietojen syöttö. Tietojen syötössä aluksi valitaan pelaajien määrä. Tämä tapahtuu monivalintaelementillä. Seuraavaksi syötetään tekstilaatikkoon väylien parit. Tämän jälkeen tekstilaatikoita on nimelle ja tuloksille niin monta kuin pelaajia on valittu.

Seuraava kortti on tekstin asettelu. Ensin valitaan, esitetäänkö pelaajat pysty- ja vaakasuunnassa. Tämän jälkeen on liukusäädin ja tekstilaatikko tietojen pysty- ja vaakasijainnille, pelaajien tietojen etäisyydelle, nimen ja väylätuloksen etäisyydelle sekä väylätuloksen ja kokonaistuloksen etäisyydelle. Sen jälkeen nimelle, väylätulokselle ja kokonaistulokselle on fontin, fontin värin ja fontin koon valitsin, väylätuloksen kohdalla on valitsin, jolla aktivoidaan tulokseen perustuva fontin väri. Viimeisessä kortissa on painike grafiikoiden tulostustyön aloittamiselle.

4 TOTEUTUS

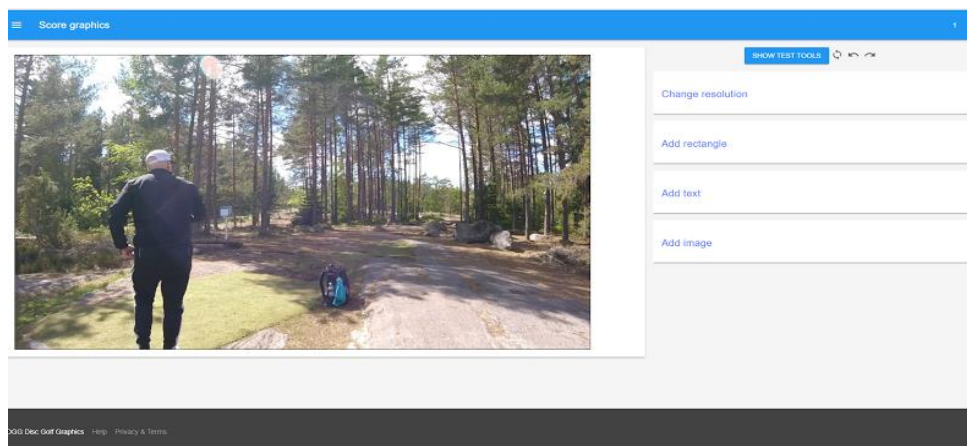
Sovelluksen tarvittavat ominaisuudet on rajattu analysoitavien materiaalien tarkastelun jälkeen ja ominaisuuksien sekä ulkoasun tarkempi suunnittelu toteutettu. Nyt alkaa projektin toteutus. Aloitan toteutuksen tekemällä ensin ulkoasun valmiiksi suunnitelmien perusteella. Tarkat suunnitelmat ja ominaisuuksien tarvitsemien syötteiden rajaus tekevät ulkoasun toteutuksen helpoksi, lisäksi Googlen material design antaa aika tarkat suositukset komponenttien ulkoasun suhteen.

Toteutuksen aikana nousi mieleen vielä uusia ominaisuuksia, joista muutama lisättiin sovellukseen. Näitä ominaisuuksia olivat kumoa- ja tee uudestaan-toiminnot, päivitä toiminto ja ulkoasuun lisättiin työkaluille ohjeet. Näiden lisäksi canvasille annettiin taustakuva, jotta läpikuultavuuden hahmottaminen olisi helpompaa.

4.1 Ulkoasu

Aloitin sovelluksen toteutuksen tekemällä ensin suunnitelmia vastaavan ulkoasun. Googlen material design litten lisäksi tein oman css-tiedoston, jolla sain elementit halutuille paikoille ja sopivaan muotoon. Esimerkiksi kortin pienentämistä varten tarvitsi elementille antaa uusi pienin mahdollinen korkeusarvo, koska material design litessä se oli määritetty liian isoksi.

Ulkoasun toteutuksessa tein index.php sivun, johon include komennolla haettiin header, content ja footer. Headerissa oli material designin pieni ylätunnisterivi, jonka vasemmasta laidasta saatiin hampurilaisvalikko auki. Footerissa oli material designin minimaalisin alatunniste. Content eli sisältö-tiedostoon rakentui itse ohjelma. Tiedostoon rakennettiin kaksi saraketta material designin grid-ohjeiden mukaisesti, tämä mahdollistaa sisällön responsiivisuuden. Toiseen sarakkeeseen tuli canvas ja toiseen työkalut omille korteilleen. Canvasille annettiin prosenttiin perustuva leveys, joten se täyttää ja mahtuu aina sille määriteltyyn tilaan. Canvasille annettiin taustakuva, jotta elementtien läpikuultavuuden hahmottaminen helpottuu.



Kuvio 10. Kuva valmiista ulkoasusta.

Ulkoasun toiminnallisuuden toteutuksessa isossa roolissa oli Javascriptin ja css:n yhdistelmällä toteutettu elementin piilotus. Otsikoiden ja nappien painaminen kutsuu funktiota, joka kytkee elementille piilotus luokan päälle tai pois päältä. Tällä on toteutettu oikean työkaluryhmän näyttö, jokaisen työkalun laajeneminen ja pelaajien tietojen syötössä oikea määrä tekstilaatikoita. Kuviossa 11 on koodiesimerkki elementin piilotukseen käytetystä koodista.

```

.hidden {
  display: none !important;
}

<div class="mdl-card__supporting-text">
  <a id="srectangle">
    <h5>
      Add rectangle
    </h5>
  </a>
  <div class="hidden toolcard" id="rectangle">
    ...

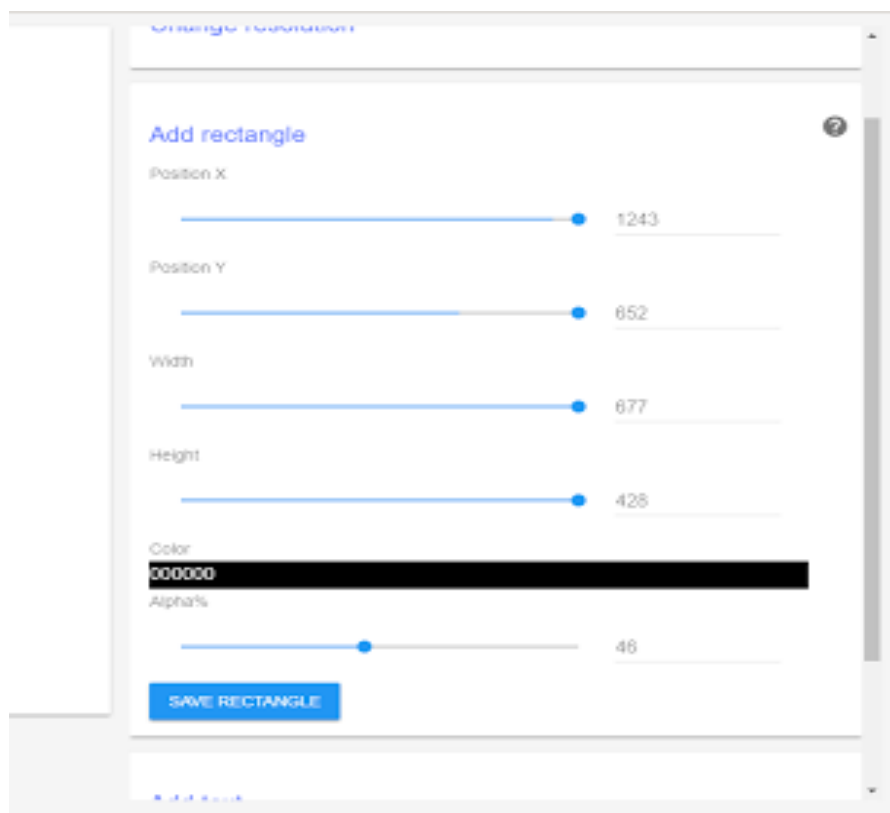
var srectangle = document.getElementById('srectangle');
var rectangle = document.getElementById('rectangle');
function showrectangle(){
  rectangle.classList.toggle('hidden');
}
srectangle.addEventListener("click", showrectangle, false);

```

Kuvio 11. Työkalun piilottamiseen käytetty koodi.

Ulkoasun muodostumisen myötä nousi esiin käytettävyyteen liittyvä ongelma. Työkalut veivät korkeussuunnassa koko ruudun verran tilaa, jonka jälkeen canvas katosi osittain näkyvistä. Tämä kuitenkin saatiin korjattua css:n overflow-ominaisuudella, jolla mahdollistettiin työkalujen vieritys niin, että canvas pysyy paikoillaan. Samalla tekniikalla saatiin pienemmille ruuduille canvas pysymään koko ajan näkyvissä ja työkalujen rullaus ja käyttö ruudun alalaitaan.

Ohjelmaan toteutusvaiheessa lisättiin vielä päivitys-, kumoamis- ja uudelleentekemis-ominaisuudet, joille lisättiin painikkeet työkaluryhmän vaihto-painikkeen viereen. Lisäksi lisättiin vielä työkaluille lyhyet ohjeet. Kun työkalun kortti laajennetaan, ohje löytyy viemällä osoitin oikeassa yläkulmassa olevan kysymysmerkki ikonin päälle.



Kuvio 12. Työkalu avattuna ja vieritys ominaisuus valmiina.

4.2 Ominaisuudet

Lähdetään toteuttamaan ohjelman toiminnallisuutta ja määriteltyjä ominaisuuksia. Ulkoasu on valmiina ja paikat suunnitelmissa vaadituille käyttäjäsyötteille tehtyinä.

4.2.1 Canvasin perustoiminnot

Canvasin koon vaihtaminen toteutettiin kahdella tekstilaatikolla, leveys ja korkeus. Näille määriteltiin koodissa seuraaja joka seuraa, mikäli arvoa vaihdetaan ja päivittää canvasin kokoa sen mukaan.

Seuraavaksi tehtiin canvasin alustus. Tehtiin funktio savecanvas, joka tallentaa canvasin sen hetken tilanteen. Toteutuksen aikana lisättiin kumoamis- ja uudelleentekemis- toiminnot, joiden toimiminen toteutettiin sillä, että savecanvas tallentaa sen hetkisen tilanteen kuvadatan aina taulukkoon, tämän jälkeen kumoamis- ja uudelleentekemis- toiminnoilla voidaan siirtyä taulukossa taakse- ja eteenpäin. Canvasin päivitys toteutettiin tyhjentämällä canvas ja hakemalla tuorein tallennettu versio tallennettujen canvaasien taulukosta. Kumoa, tee uudelleen ja päivitys- toiminnoille tehtiin painikkeet työkalujen vaihtopainikkeen viereen.

4.2.2 Elementtien lisääminen

Nelikulmion lisääminen toteutetaan drawrect funktiolla. Nelikulmion sijainti, koko, väri ja läpikuultavuus otetaan käyttäjän syötteistä. Ennen nelikulmion piirtämistä kutsutaan aina refreshcanvas funktiota, joka tyhjentää canvasin ja hakee tuoreimman tallennetun canvasin. Nelikulmion piirto funktiota kutsutaan sen työkalun käyttäjä syötteistä. Syötteisiin on laitettu html- koodiin onInput ja onChange tapahtumille käsky kutsua drawrect- funktiota. Liukusäätimien kanssa onInput mahdollistaa elementin automaattisen päivytyksen canvasilla jo liukusäädintä käytettäessä. Syötteiden suurimpien arvojen päivitys canvasin koon mukaan sekä liukusäädinten ja tekstilaatikoiden synkronointi on toteutettu kahdella funktiolla, joita kutsutaan aina, kun elementin arvoa on muutettu, riippuen onko muutettu tekstilaatikkoa vai liukusäädintä.

Tekstin lisääminen toteutetaan `addtext`-funktiolla. Haluttu teksti, sijainti, fontin koko, fontti ja väri sekä läpikuultavuus otetaan käyttäjän syötteistä. Samaan tapaan kuin nelikulmiota tehtäessä myös tekstiä lisättäessä `canvas` aina tyhjennetään ja haetaan uusin tallennettu kuva `canvasin` pohjalle. Myös syötteiden `onInput` ja `onChange` käskyt sekä liukusäädinten ja tekstilaatikoiden päivittäminen toimii samalla lailla kuin nelikulmioita lisätessä. Fontteja laitettiin aluksi käytettäväksi neljä.

Kuvan lisäämistä varten on ikoni. Kun tästä painetaan, aukeaa käyttäjälle tiedoston valitsin. Kun tiedosto on valittu, viedään se kuvanluku-funktioon, jossa se luetaan ja sen jälkeen kutsutaan kuvan asettelu-funktiota. Kuvan asettelu-funktio kutsuu taas aina aluksi `refreshcanvas`-funktiota, joka tyhjentää `canvasin` ja lataa uusimman tallennetun `canvasin`. Kuvan sijainti haetaan käyttäjän syötteistä ja kuvan korkeutta ja leveyttä pystyy vaihtamaan nolasta 200 prosenttiin. Syötteet kutsuvat taas kuvanlisäysfunktiota `onInput` ja `onChange` tapahtumilla, sekä liukusäädinten käyttö päivittää tekstilaatikoita ja tekstilaatikon päivitys liukusäätimiä.

4.2.3 Tietojen syöttö ja tekstin asettelu

Tietojen syötössä on tekstilaatikko par-tuloksille ja tekstilaatikoita nimille ja tuloksille niin monta, kuin pelaajia on valittu. Par ja tulos tekstilaatikoita muokattaessa kutsutaan aina `inputseparator` funktiota, joka erottelee pilkuilla, välilyönneillä ja tabulaattoreilla syötteen sille tarkoitettuun taulukkoon.

Tekstin asettelu on yhdessä kuvien prosessoinnin funktioiden kanssa työn monimutkaisin alue. Käyttöliittymässä on taas käyttäjän syötteille liukusäätimiä ja tekstilaatikoita, jotka päivittävät ruudulla näkyvää esikatselua siirrettäessä sekä päivittävät arvojaan ristiin. Tietojen asettelusta saa valita sijainnin, pelaajien välisen etäisyyden, nimen ja väylätuloksen välisen etäisyyden sekä väylätuloksen ja kokonaistuloksen etäisyyden. Nimelle, väylätulokselle ja kokonaistulokselle saadaan valittua värit erikseen, sen lisäksi väylätulokselle saadaan tulokseen perustuva väri. Pelaajien asettelu saadaan valittua vaaka- tai pystysuoraan.

Pelaajien vaaka- tai pystysuoraan asetelun toteutin tekemällä molemmille omat funktiot ja kutsumalla oikeaa funktiota sen mukaan kumpi vaihtoehto on valittu. Funktiot ovat pitkälti samanlaisia, mutta sellaiset arvot, jotka toisella vaikuttavat x sijaintiin vaikuttavatkin toisessa funktiossa y sijaintiin ja toisin päin. Tulostenkirjoitus-funktio taas ensin päivittää canvasin, tämän jälkeen piirtää canvasille tiedot annettujen tietojen ja syötteiden perusteella. Esikatselussa näkyy ensimmäisen väylän tilanne.

Väylätuloksen tulokseen perustuva väri haetaan funktiolla `holescorecolor`. Funktiota kutsutaan aina jokaisen pelaajan väylätulosta piirrettäessä. Funktio ensin tarkastaa, onko tulokseen perustuva väritys valittuna, tämän jälkeen mikäli tulokseen perustuva väritys on valittuna vertaa funktio sitä väylän `par`:iin. Väylän `par` saadaan tänne `par`-taulukon ja väylänumeron kertovan globaalin muuttujan avulla. Jos tulos on alle `par` se saa vihreän värin, yli niin punaisen ja tasan `par` niin annetaan käyttäjän syötteessä oleva väri. Mikäli tulokseen perustuva väri on pois käytöstä, funktio antaa käyttäjän syötteessä olevan värin.

Ohjelma tekee väylän jälkeisistä tuloksista taulukot jokaiselle pelaajalle. Taulukon `i`-solussa on väylän `i+1` jälkeinen yhteistulos. Kokonaistuloksen piirtoa varten kutsutaan väylän jälkeistä kokonaistulosta parametrina käyttäen aina `totalplusorminus`-funktioita. Tämä funktio tarkistaa kokonaistuloksen tilanteen suhteessa `par`:iin ja antaa sille etumerkin '+' jos se on yli, korvaa tekstillä "par" jos se on tasan, tai palauttaa alkuperäisen arvon, koska negatiivisissa luvuissa on jo miinusmerkki edessä.

4.2.4 Kuvien prosessointi

Nyt ollaan käyty läpi yhden väylän tulosgrafiikan piirtäminen. Käydään läpi kuvien prosessointi lyhyesti. "Start processing"-painikkeella aloitetaan grafiikoiden tulostus. `Zipmakercaller`-funktio ohjaa työn etenemistä. Ohjelman kierron pituus katsotaan väyliä `par`-taulukon pituuden mukaan. Kun väyliä on vielä jäljellä, funktio kutsuu aina aiemmin esiteltyä piirtofunktioita ja sen jälkeen funktiota, joka lisää tämän hetkisen canvasin kuvan `zip`-tiedostoon, sekä korottaa väylän numeroa

yhdellä. Kun väylänumero saavuttaa annettujen par-tulosten määrän kutsuu funktio zip-tiedoston tallennus- ja latausfunktiota.

Käydään läpi tarkemmin funktioiden toimintaa. Eli painikkeella käynnistetään toiminta. Zipmakercaller- funktio saa kutsun ja ensimmäiseksi hakee väylien määrän par taulukon pituudesta, jotta tiedetään montako kierrosta funktion pitää tehdä. Seuraavaksi funktio saapuu if-lausekkeeseen, jonka ehtona tarkistetaan, onko väylä-numero pienempi kuin väylien määrä. Jos ehto on tosi, niin funktio kutsuu scoredirection-funktiota ja antaa parametriksi callback-funktion. Scoredirection-funktio kutsuu tulosten piirron suunnan mukaan joko pysty- tai vaakasuuntaan tulokset piirtävää funktiota välittäen sille annetun parametrin. Funktio lähetetään callbackinä sen sijaan, että se olisi laitettu Zipmakercaller-funktioon seuraavaksi koska muuten javascript lähtisi suorittamaan jo sitä osuutta, eikä odottaisi, että aiempi vaihe olisi tullut valmiiksi.

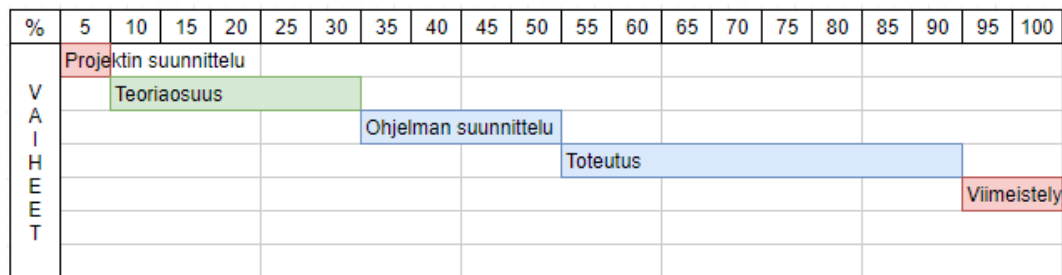
Callback-funktion kutsu on rakennettu tulosten-piirto funktioihin siten, että ensin tarkistetaan, onko callback annettu, ja mikäli on niin, se suoritetaan. Callback funktio sisältää käskyn lisätä väylä-numeroa yhdellä ja tämän jälkeen kutsua zip-maker-funktiota.

Zipmaker-funktio ottaa canvasin sen hetkisen kuvan ja lisää sen tulevan zip-tiedoston muuttujaan png muodossa, käyttäen nimeä "scoregraphics" + väylänumero + ".png". Tämän jälkeen funktio tulee SetTimeout käskyyn, jossa sitä käsketään kutsumaan zipmakercaller-funktiota puolen sekunnin kuluttua. Tämä puolen sekunnin viive on sen takia että ohjelma kerkeää tallentaa tiedoston zip-tiedostoon.

Ohjelma on palannut taas zipmakercaller-funktioon. Jos väylänumero on saavuttanut saman arvon väylien määrän kanssa, if-lausekkeen ehto ei enää täyty ja mennään else-osioon. Else-osiossa kutsutaan sekunnin viiveellä savezip-funktiota ja alustetaan väylä-numero uudelleen. Savezip-funktiossa haetaan päivämäärä tiedot muuttujaan. Tämän jälkeen tallennetaan luotu zip-tiedosto ja aloitetaan sen lataus Filesaverjs:n avulla. Nimeksi tulee "ScoreGraphics" + MD + ".zip". Lopuksi funktiossa alustetaan zip-tiedosto uudelleen.

5 TYÖN PROSESSI

Tässä kappaleessa käydään läpi, miten ajankäyttö projektissa jakaantui. Työ voidaan jakaa kahteen isoon kokonaisuuteen, teoria- ja toteutusosuuteen. Teoriaosuudessa käydään läpi projektissa käytetyt tekniikat. Toteutuksessa käydään läpi ohjelman suunnittelu ja toteutus.



Kuvio 13. Kuvio projektin vaiheista.

Aloitin projektin selvittämällä, mitä tekniikoita työssäni tarvitsisin ja yritin karottaa projektin aikana mahdollisesti ilmeneviä riskejä. Projektiin tarvittavat tekniikat löytyivät ja ne todettiin toimiviksi. Tähän käytettiin projektissa noin 5 % ajasta.

Kielet ja tekniikat on nyt päätetty ja teoriaosuus voidaan aloittaa. Etsin kirjastosta lähteitä paremmin tunnetuista tekniikoista. Pienemmistä tekniikoista käytin lähteenä kehittäjien niille tekemiä kotisivuja. Teoriaosuuteen käytetty aika projektissa oli noin 25 %.

Ohjelman suunnittelu jakautuu kahteen vaiheeseen, ohjelmaan tarvittavien ominaisuuksien rajaamiseen ja käyttöliittymän ulkoasun suunnitteluun. Tähän työvaiheeseen ja sen dokumentointiin kului projektissa aikaa noin 20 %.

Ohjelman toteutus aloitettiin toteuttamalla ulkoasu. Tämän jälkeen toteutettiin ominaisuudet. Toteutukseen meni odotetusti eniten aikaa ja testauksien ja dokumentoinnin kanssa siihen meni 40 % ajasta.

Työn viimeistelyyn kuten yhteenveto- ja tiivistelmäkappaleisiin kului projektissa aikaa 10 %.

Eri osa-alueisiin aikaa kului suurin piirtein niin kauan kuin olin ajatellutkin. Kokonaisuuksien sisällä oli toki suhteessa isojakin heittoja, kun esimerkiksi ulkonäön toteutuksessa oli jokin asia ristiriidassa Material Designing kanssa tai jokin ominaisuus ei heti toiminutkaan ajatellulla tavalla.

6 YHTEENVETO

Projektissa tavoitteena oli toteuttaa verkkosovellus, jolla pystytään luomaan tulosgrafiikoita frisbeegolfvideoille. Opinnäytetyössä käyn läpi ensin lyhyesti tekniikoita, joita projektissa käytetään. Työssä määriteltiin tulevat ominaisuudet, tehtiin suunnitelma niiden toteutuksesta ja määriteltiin käyttöliittymän tarpeet. Tämän jälkeen toteutettiin käyttöliittymä ja ohjelman toiminnallisuus.

Työssä onnistuin mielestäni hyvin. Sovellusta pystytään käyttämään valmiiden taustojen kanssa tai luomaan omia. Se sisältää erilaisia ominaisuuksia, joilla saadaan useimpien vaatimukset täytettyä, kuitenkin pysyen erittäin yksinkertaisena. Sovellus saatiin siihen pisteeseen, että sen julkistaminen voidaan suorittaa ja sitä on myös helppo lähteä jatkokehittämään.

Liitteessä 1 on ruutukaappaukset jokaisesta sovelluksen vaiheesta, sekä kaksi esimerkkiä valmiista tulosgrafiikoista. Kuvista näkee työkalujen kontrollit ja niistä pystytään hahmottamaan niiden toimintaa. Valmiit tulosgrafiikat ovat siis png-tiedostomuodossa ja ne liitetään editorilla videoiden päälle.

Projekti kehitti minua alani osaamisessa todella paljon. Kehityin hurjasti työssä läpikäytyjen tekniikoiden kanssa. Projektin aikana opin paljon uutta myös sellaisista asioista, jotka olin rajannut opinnäytetyön ulkopuolelle, kuten esimerkiksi git-versiohallintaohjelmiston käytön perusteista ja oman palvelimen käyttöönotosta ja hallinnasta. Projekti myös aktivoi minua tutkimaan muita tekniikoita, esimerkiksi Reactjs herätti mielenkiintoa ja se olisi voinut olla erittäin hyödyllinen tässä projektissa.

Sovellukseen mahdollisia tulevia ominaisuuksia tuli mieleen jatkuvasti koko projektin ajan. Esimerkiksi valmiita taustoja, tulosten tekstin asettelun vienti & tuonti ja väylä-tuloksen ilmaiseminen suhteessa par-tulokseen neliöillä sekä ympyröillä. Samanlaisella tekniikalla toteutettavia vähän erilaisia ohjelmia voisi myös lisätä sivustolle, esimerkiksi ohjelma jolla saa grafiikan väylän numero-, pituus- ja partiedoista näkymään aina väylän ajaksi. Myös ohjelma, jolla saataisiin samaan taustaan vain vaihtuva teksti, voisi olla hyödyllinen. Tätä voitaisiin käyttää fris-

beegolf videoilla, vaikka kertomaan kiekon nimi, mutta myös timelapse-videoita editoidessa olen myös tarvinnut tällaista sovellusta kertomaan esimerkiksi, mikä kuukausi videolla on meneillään.

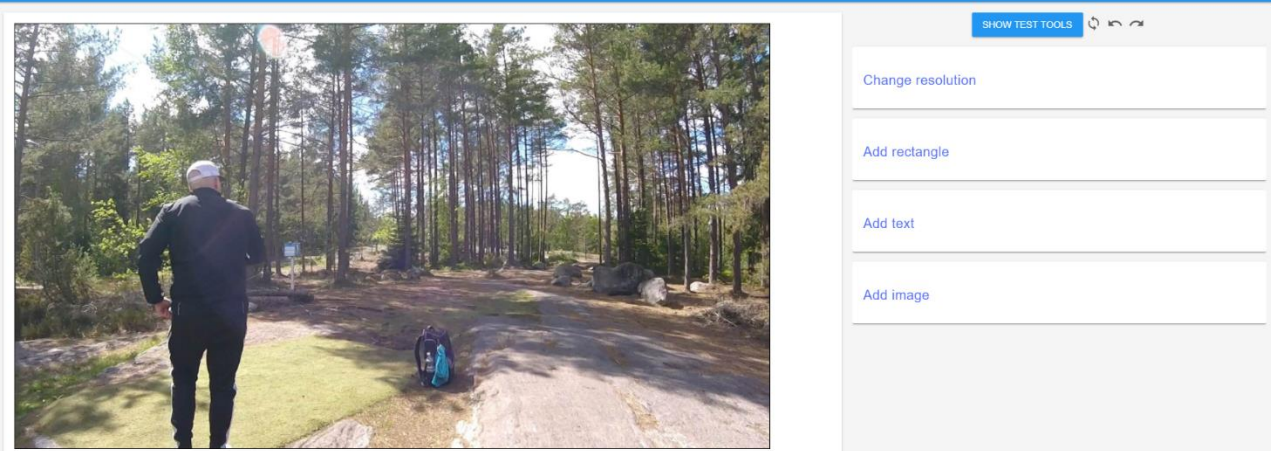
Sovelluksen tulevaisuus on vielä avoin. Suunnitelmissani on sen tarjoaminen frisbeegolf-aiheisille sivustoille, esimerkiksi suomen frisbeegolfliitolle. Näin sovellukselle voitaisiin löytää paikka, josta harrastajien on se helppo löytää ja käyttää. Muiden harrastajien auttaminen ja itselleni työelämään hyvän referenssin luominen olivat suurimmat saavutukset, joita uskalsin toivoa saavuttavani tällä työllä.

LÄHTEET

- 7k Frisbeegolf. SM2016 Etuysi FINAL. 1.9.2016.
<https://www.youtube.com/watch?v=dDcRHGMvu8c>
- Buddy, A & Moll, C & Collison, S. 2007. CSS Tehokas hallinta. Jyväskylä.
Gummerus Kirjapaino Oy
- Castro, E. 2007. Kotisivut kuntoon - HTML, XHTML ja CSS. Jyväskylä.
Gummerus Kirjapaino Oy
- Central Coast Disc Golf. 2016 Worlds MPO Round 1 Part 1 (Wysocki, Lauber, Callaway, Brown). 10.8.2016. <https://www.youtube.com/watch?v=z0GqAOwZt8s>
- Central Coast Disc Golf. 2017 PDGA Worlds Round 3 Part 2 (Wysocki, McCray, Owens, Colglazier). 24.6.2017. <https://www.youtube.com/watch?v=MH-1Z3HqJQQ>
- East Desire. js Color Picker. Viitattu 28.6.2017. <http://jscolor.com/>
- Google. Material Design. Viitattu 26.6.2017.
<https://material.io/guidelines/#introduction-goals>
- Grey, E. FileSaver.js. Viitattu 28.6.2017. <https://github.com/eligrey/FileSaver.js>
- Jomez Productions. 2.7.2017. 2017 Red, White & Boom | Round 1, Back 9 | Gibson,Brown,McDonald,Napier.
<https://www.youtube.com/watch?v=8dm3xZJXg5U>
- Knightley, S. JSZip. Viitattu 28.6.2017. <https://stuk.github.io/jszip/>
- Negrino, T & Smith, D. 2007. JavaScript - Tehokas hallinta. Jyväskylä. Gummerus Kirjapaino Oy
- The SpinTV. 2016. DGWT EO2016 Lead Card Final Round, Front 9 (Wysocki, Nybo, Lizotte, McBeth). 6.6.2016
<https://www.youtube.com/watch?v=8GGvpc4K4yM>
- Vuorelan Veikot. Parifrisbeegolfin-SM 2017 Kierros 3 (Final round). 26.6.2017.
<https://www.youtube.com/watch?v=yongGV1qcrU&t=288s>
- W3C. HTML5. Viitattu 19.6.2017. <https://www.w3.org/TR/html5/>

Sovelluksen etusivu

☰ Score graphics



SHOW TEST TOOLS

Change resolution

Add rectangle

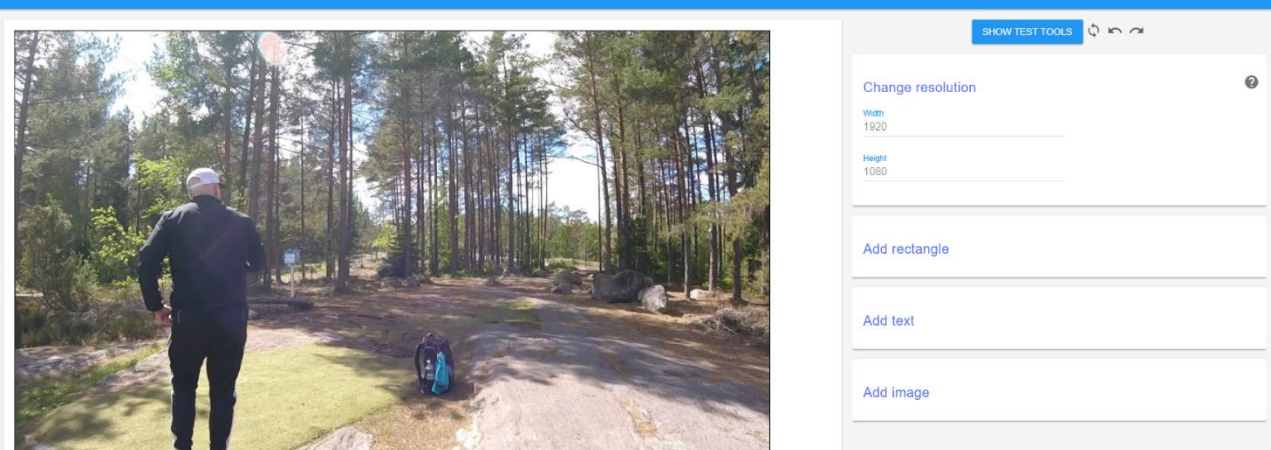
Add text

Add image

DGG Disc Golf Graphics Help Privacy & Terms

Resolution vaihto

☰ Score graphics



SHOW TEST TOOLS

Change resolution

Width: 1920

Height: 1080

Add rectangle

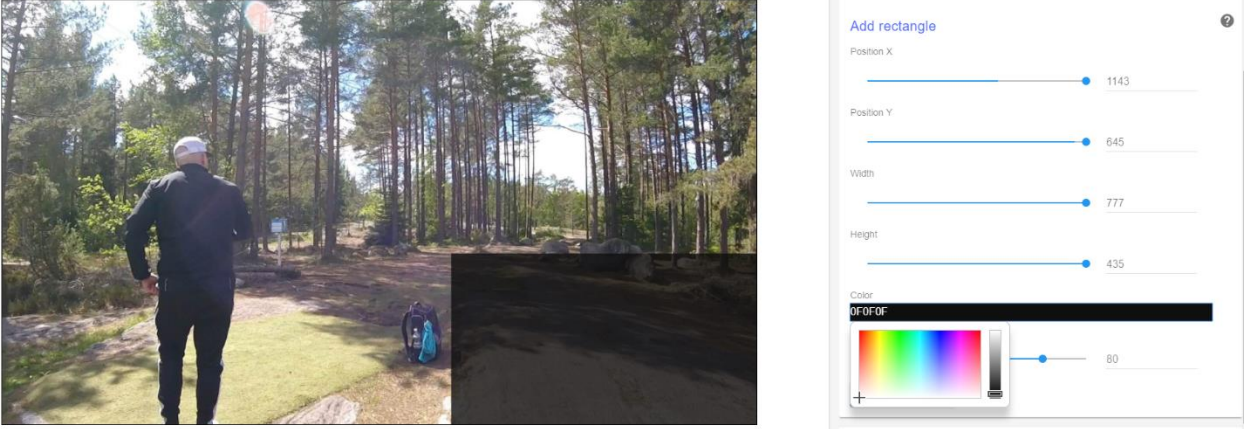
Add text

Add image

DGG Disc Golf Graphics Help Privacy & Terms

Nelikulmion lisääminen

Score graphics



Add rectangle

Position X: 1143

Position Y: 645

Width: 777

Height: 435

Color: 0F0F0F

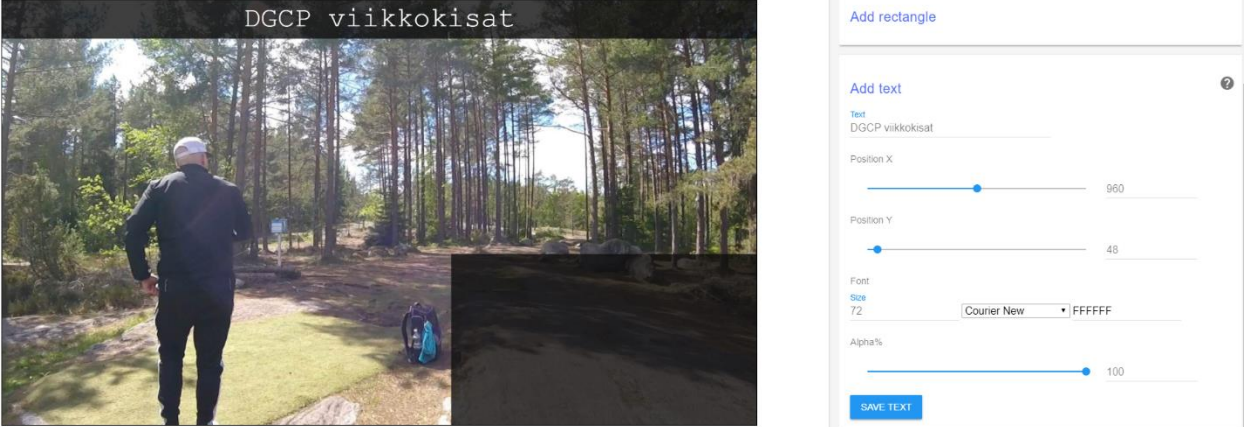
80

Add text

DGG Disc Golf Graphics Help Privacy & Terms

Tekstin lisääminen

Score graphics



Add rectangle

Add text

Text: DGCP viikkokisat

Position X: 960

Position Y: 48

Font: Courier New

Size: 72

Alpha: 100

SAVE TEXT

Add image

DGG Disc Golf Graphics Help Privacy & Terms

Kuvan lisääminen

Score graphics

DGCP viikkokisat

Add rectangle

Add text

Add image

Position X: 1730

Position Y: 451

Width%: 20

Height%: 20

SAVE IMAGE

DGG Disc Golf Graphics Help Privacy & Terms

Tietojen syöttö

Score graphics

DGCP viikkokisat

Pars, names and scores

4

Hole pars: 3 3 3 3 3 3 3 4 4 3

First Player: Matti

Scores: 3 2 3 2 3 3 2 4 4 4

Second Player: Mikko

Scores: 4 4 2 3 3 3 3 4 4 4

Third Player: Aleksi

Scores: 3 3 3 2 3 3 4 5 2


Fourth Player: Vesa

Scores: 3 3 2 3 2 4 5 3 5

DGG Disc Golf Graphics Help Privacy & Terms

Tulosten tekstin asettelu

Score graphics



DGCP viikkokisat

Matti	3	par
Mikko	4	+1
Alexi	3	par
Vesa	3	par

Text layout

Vertical Horizontal

Position X: 1184

Position Y: 719

Player to Player: 292

Name to hole score: 390

Hole score to total: 200

Player name font:
Size: 70 | Arial | F3F3F3


Hole score font:
 Use color depending on score
Size: 90 | Arial | F3F3F3

Round total font:
Size: 70 | Arial | F3F3F3

DGG Disc Golf Graphics Help Privacy & Terms

Grafiikoiden prosessoinnin aloittaminen

Score graphics



DGCP viikkokisat

Matti	3	par
Mikko	4	+1
Alexi	3	par
Vesa	3	par

SHOW BACKGROUND EDITING TOOLS

Pars, names and scores

Text layout

Make score graphics

START PROCESSING

DGG Disc Golf Graphics Help Privacy & Terms

Esimerkki valmiista tulosgraafiikasta. Väylä 4

DGCP viikkokisat



Matti	2	-2
Mikko	3	+1
Alexi	2	-1
Vesa	3	+2

Esimerkki valmiista tulosgraafiikasta. Väylä 17

DGCP viikkokisat



Matti	4	-5
Mikko	5	+1
Alexi	4	+2
Vesa	4	par