



Jani Salo

Käyttäjäkokemuksen parantaminen responsiivisella suunnittelulla

Metropolia Ammattikorkeakoulu

Insinööri (AMK)

Tieto- ja viestintäteknikan tutkinto-ohjelma

Insinöörityö

11.3.2025

URN:NBN:fi:amk-202504287871

Tiivistelmä

Tekijä:	Jani Salo
Otsikko:	Käyttäjäkokemuksen parantaminen responsiivisella suunnittelulla
Sivumäärä:	39 Sivua
Aika:	11.3.2025
Tutkinto:	Insinööri (AMK)
Tutkinto-ohjelma:	Tieto- ja viestintätekniikka
Ammatillinen pääaine:	Mobiilikehitys
Ohjaajat:	Lehtori Ulla Sederlöf

Responsiivinen suunnittelu on lähestymistapa, joka mahdollistaa verkkosivustojen ja sovellusten mukautumisen erilaisiin näyttölaitteisiin ja näyttökokoihin saumattomasti. Tämä tarkoittaa käyttäjäystävällisen käyttökokemuksen varmistamista riippumatta siitä, käytetäänkö laitetta tietokoneella, älypuhelimella tai tabletilla. Tärkeimmät responsiivisen suunnittelun periaatteet ovat joustavuus, jossa sisältö skaalautuu automaattisesti, sisällön priorisointi, jolloin pienemmällä näytöllä saatetaan siirtää merkityksettömpiä asioita seuraavalle näytölle tai piilottaa kokonaan ja media queryt, jotka mahdollistavat sisällön muokkaamisen eri näyttöresoluutioille. Lisäksi responsiivisessa suunnittelussa korostetaan suorituskyvyn optimointia, jotta sivusto latautuu nopeasti kaikilla laitteilla. Näiden periaatteiden noudattaminen takaa, että käyttäjät voivat nauttia saumattomasta ja miellyttävästä käyttökokemuksesta riippuen siitä, mitä laitetta he käyttävät.

Projektin tuloksena syntyi React-pohjainen verkkosovellus, joka on suunniteltu tehostamaan työnhakijoiden tietojen hallintaa rekrytointiprosessissa. Sovellus mahdollistaa työnantajille hakijatietojen keräämisen, järjestämisen ja tarkastelun tehokkaasti, mikä vähentää manuaalista työtä ja parantaa rekrytinnin sujuvuutta. Käyttöliittymä on responsiivinen ja mukautuu saumattomasti eri näyttökokoihin tarjoten sujuvan käyttökokemuksen kaikilla laitteilla. Järjestelmän taustalla toimii Node.js-pohjainen backend, ja tietojen turvallinen sekä skaalautuva tallennus on toteutettu Azuren SQL-tietokannalla. Lisäksi sovellus sisältää selkeän hallintapaneelin, jonka kautta rekrytoijat voivat helposti hallita hakijatietoja, vertailla hakijoita ja tehdä tietoon perustuvia rekrytointipäätöksiä.

Avainsanat: responsiivinen suunnittelu, React, CSS, käyttäjäkokemus

Tämän opinnäytetyön alkuperä on tarkastettu Turnitin Originality Check -ohjelmalla.

Abstract

Author: Jani Salo
Title: Improving the user experience with responsive design
Number of Pages: 39 pages
Date: 3 March 2025

Degree: Bachelor of Engineering
Degree Programme: Information and Communication technology
Professional Major: Mobile Solutions
Supervisors: Lecturer Ulla Sederlöf

Responsive design is an approach that enables websites and applications to adapt seamlessly to different devices and screen sizes. This ensures a user-friendly experience regardless of whether the device is a computer, smartphone, or tablet. The key principles of responsive design include flexibility, where content scales automatically; content prioritization, which may involve moving less important elements to the next screen or hiding them entirely on smaller displays; and media queries, which allow content to be adjusted for different screen resolutions. Additionally, responsive design emphasizes performance optimization to ensure fast loading times across all devices. By following these principles, users can enjoy a seamless and pleasant experience regardless of the device they are using.

The project resulted in a React-based web application designed to streamline the management of job applicants' information in the recruitment process. The application allows employers to collect, organize, and review applicant data efficiently, reducing manual workload and improving the overall recruitment workflow. It features a responsive user interface that adapts seamlessly to different screen sizes, ensuring a smooth user experience across devices. The system is built with a Node .js backend and utilizes an Azure SQL database for secure and scalable data storage. Additionally, administrators can access an intuitive dashboard to manage applicant information, making it easier to compare candidates and make informed hiring decisions.

Keywords: Responsive design, React, CSS, User experience

Lähteet

1	Johdanto	2
2	React-projekti	3
3	Käsitteet ja perusteet	5
3.1	Kehykset	10
4	Käyttäjäkokemus: merkitys ja elementit	17
4.1	Käyttäjäkokemuksen merkitys	17
4.2	Käyttäjäkokemuksen elementit	19
4.2.1	Käytettävyys	19
4.2.2	Esteettisyys	22
4.2.3	Suorituskyky	23
4.2.4	Luotettavuus	24
4.2.5	Relevantti sisältö	25
4.2.6	Saavutettavuus	25
5	Psykologian vaikutus	27
5.1	Värien vaikutus	27
5.2	Fonttien vaikutus	28
5.3	Visuaalisten elementtien vaikutus	28
6	Haasteet ja ratkaisut	30
7	Käytettävyystestaus ja palautteen hyödyntäminen	31
7.1	Käytettävyystestauksen merkitys	31
7.2	Käytettävyystestauksen menetelmät	32
8	Tulevaisuuden näkymät	33
9	Yhteenveto	34
	Lähteet	1

1 Johdanto

Insinööriyön tarkoituksena on syventyä responsiivisen suunnittelun merkitykseen ja vaikutuksiin käyttäjäkokemuksessa, erityisesti mobiilisovellusten kontekstissa. Nykypäivänä mobiililaitteiden käyttö lisääntyy jatkuvasti, ja käyttäjät odottavat saavansa saumattoman ja nautinnollisen kokemuksen riippumatta siitä, millä laitteella he sovellusta käyttävät. Responsiivinen suunnittelu on lähestymistapa, joka mahdollistaa sovellusten toimivan optimaalisesti erilaisilla näytöillä ja laitteilla.

Insinööriyössä käsitellään syvällisesti tärkeimpiä käsitteitä ja perusteita responsiivisessa suunnittelussa. Tarkastelun kohteena on erityisesti käyttäjäkokemuksen keskeinen merkitys mobiilisovellusten menestyksessä sekä vastaus responsiivisen suunnittelun asettamiin haasteisiin eri näkökulmista. Lisäksi työssä tarkastellaan erilaisia ratkaisuja, joita responsiivinen suunnittelu tarjoaa näiden haasteiden voittamiseksi.

Käsittelyssä ovat myös tärkeimmät teknologiat ja työkalut, joita responsiivisessa suunnittelussa hyödynnetään.

Työssä viitataan React-pohjaiseen sovellukseen, jonka tarkoituksena on helpottaa rekrytointiprosessia tarjoamalla käyttäjälle selkeä ja tehokas käyttöliittymä esimerkiksi hakemusten hallintaan ja aikataulutukseen. Sovellus on suunniteltu toimimaan sekä mobiililaitteilla, tableteilla että tietokoneilla. Työssä pyritään käsittelemään responsiivista suunnittelua mahdollisimman laajasti, mukaan lukien eri laitekokojen – kuten mobiili- ja tietokonenäkymien – väliset erot ja niiden huomioiminen käyttöliittymän suunnittelussa.

Insinööriyö pyrkii antamaan lukijalle kokonaisvaltaisen käsityksen responsiivisen suunnittelun merkityksestä ja sen vaikutuksista mobiilisovellusten kehityksessä. Tämä ymmärrys voi auttaa kehittäjiä ja suunnittelijoita luomaan entistä parempia ja käyttäjäystävällisempiä mobiilisovelluksia tulevaisuudessa.

2 React-projekti

Tässä opinnäytetyössä esitetty React-pohjainen verkkosovellus on kehitetty tukemaan työnhakijoiden tietojen hallintaa rekrytointiprosessissa. Sovelluksen pääasiallinen tavoite on virtaviivaistaa työnhakuprosessia ja tarjota työnantajille tehokas työkalu työnhakijatiетоjen tallentamiseen, hallintaan ja tarkastamiseen. Sen avulla rekrytointiorganisaatiot ja yritykset voivat käsitellä suuria määriä hakijatiетоja selkeästi ja järjestelmällisesti, mikä parantaa prosessien sujuvuutta ja vähentää manuaalisen työn määrää.

Tämä projekti toteutettiin erityisesti yrityksen tarpeisiin, joka tarvitsi nykyaikaisen ja helppokäyttöisen ratkaisun hakijatiетоjen käsittelyyn. Sovellus mahdollistaa työnhakijalle joustavan ja helpon tavan antaa tarvittavat tiedot. Työnhakijalle lähetetään automaattisesti linkki, jonka kautta hän voi täyttää omat perustietonsa sekä syvempää tietoa osaamisestaan. Täyttölomake on suunniteltu siten, että osa kysymyksistä on kaikille hakijoille samoja, kun taas osa kysymyksistä mukautuu hakemuksen kohteena olevan työnimikkeen mukaan. Tällä tavoin varmistetaan, että työnantaja saa juuri kyseiseen työtehtävään tarvittavat tiedot ilman, että hakijan tarvitsee täyttää ylimääräisiä tai epäolennaisia tietoja.

Sovellus mukauttaa kysymyksiä esimerkiksi työnimikkeen ja työtehtävän erityisvaatimusten mukaan, jolloin lomakkeeseen voidaan sisällyttää tarpeellisia lisätietoja, kuten edellytyksiä ajokortille, omalle autolle tai tiettyihin pätevyysiin liittyviä kysymyksiä. Näin työnantaja voi nopeasti arvioida, täyttääkö hakija työn kannalta oleelliset vaatimukset, ja saada kokonaisvaltaisen kuvan hakijan pätevydestä sekä osaamisalueista.

Järjestelmän kautta työnantaja voi helposti hakea, muokata ja tarkastella hakijoiden tietoja. Hakijatiedot on mahdollista järjestää selkeiksi kokonaisuuksiksi, mikä tukee hakijoiden vertailua ja nopeuttaa päätöksentekoa. Tämä vähentää rekrytoijien manuaalista työtaakkaa ja antaa mahdollisuuden tehdä tietoon perustuvia päätöksiä nopeasti ja tehokkaasti. Sovellus auttaa myös säilyttämään

hakijatietoja turvallisesti ja tietoturvakäytäntöjä noudattaen, mikä on keskeistä erityisesti henkilötietoja käsitteleville yrityksille.

Opinnäytetyön esimerkkisovellus toimii näin ollen myös oppimateriaalina niille, jotka haluavat ymmärtää, miten moderneilla teknologioilla, kuten Reactilla, voidaan toteuttaa tehokas rekrytointisovellus. Tämä projekti on rakennettu skaalautuvaksi, ja sitä voidaan jatkossa laajentaa erilaisilla lisäominaisuuksilla, kuten hakijaprofiilien automaattinen arviointi ja pisteytys, mikä entisestään nopeuttaisi ja tehostaisi rekrytointiprosessia.

Sovelluksen tekninen toteutus perustuu moderneihin web-kehitysteknologiaihin. Frontend-puoli on kehitetty React-kirjastolla, joka mahdollistaa dynaamisen ja responsiivisen käyttöliittymän rakentamisen. Backend-toiminnallisuudet on toteutettu Node.js-ympäristössä, joka tarjoaa suorituskykyä ja skaalautuvuutta tarvittavien palveluiden tarjoamiseen. Tietokantana on käytetty Azuren SQL-palvelua, joka takaa tietojen turvallisen tallentamisen ja helpon skaalautuvuuden.

Kun käyttäjä syöttää tietonsa sovelluksen etusivulla, tiedot lähetetään backend-palvelimelle REST-rajapinnan avulla. Backend käsittelee saapuvat pyynnöt, validoi ja tallentaa tiedot Azuren SQL-tietokantaan. Admin-oikeuksilla varustettu käyttäjä voi kirjautua sisään sivustolle ja siirtyä admin-sivulle, jossa hänellä on mahdollisuus tarkastella kaikkien työnhakijoiden tietoja excel-taulukon kaltaisessa näkymässä. Tämä näkymä mahdollistaa hakijoiden tietojen tarkastelun, muokkaamisen ja tarvittaessa poistamisen.

Sovellus on suunniteltu tukemaan rekrytointin toiminnallisia vaatimuksia ja tarjoamaan käyttäjälleen nopean ja luotettavan käyttökokemuksen. Frontend- ja backend-komponentit kommunikoivat keskenään RESTful-rajapintojen kautta, mikä mahdollistaa hajautetun rakenteen. Azuren tarjoamat palvelut, kuten SQL-tietokanta ja palvelin tarjoavat luotettavan ja skaalautuvan alustan sovelluksen toiminnalle.

3 Käsitteet ja perusteet

Responsiivinen suunnittelu on keskeinen lähestymistapa mobiilisovellusten ja verkkosivustojen kehittämisessä, jotta ne toimivat saumattomasti ja tarjoavat erinomaisen käyttäjäkokemuksen eri laitteilla ja näyttökoilla. Sen tavoitteena on varmistaa, että sovellusta tai verkkosivustoa voi käyttää sujuvasti älypuhelimella, tabletilla tai tietokoneella ilman, että käyttäjän tarvitsee suurentaa tai vieittää näkymää vaakasuunnassa. [1; 2.]

Keskeinen osa responsiivista suunnittelua on joustava rakenne, jossa käyttöliittymän elementit mukautuvat automaattisesti eri näyttökokoihin ja resoluutioihin. Tämä saavutetaan hyödyntämällä suhteellisia mittayksiköitä, kuten prosenttiosuuksia, pikselien sijaan. Esimerkiksi asettamalla elementin leveydeksi 25 % varmistetaan, että se vie aina neljäsosan näytön leveydestä riippumatta laitteen koosta. Tällainen dynaaminen mukautuminen parantaa käyttökokemusta ja käytettävyyttä. [1; 2; 3.]

Responsiivisen suunnittelun toteutuksessa hyödynnetään moderneja web-tekniologioita, kuten HTML5:ta ja CSS3:a. HTML5 mahdollistaa rakenteellisesti rikkaampien ja monipuolisempien sivustojen luomisen, kun taas CSS3 tarjoaa kehittyneitä tyylittelyvaihtoehtoja ja visuaalisia ominaisuuksia. [3.]

Eriyisen tärkeässä roolissa responsiivisessa suunnittelussa ovat media query -tekniikat, joilla voidaan määrittää erilaisia tyylejä ja asetteluja näytön koon ja ominaisuuksien perusteella. Media queryjen avulla kehittäjät voivat asettaa ehtoja, joiden perusteella tiettyjä tyylejä sovelletaan, esimerkiksi näyttöleveyden tai -korkeuden muuttuessa. Tämä mahdollistaa sivuston ulkoasun ja toiminnallisuuden mukauttamisen dynaamisesti eri laitteille ja käyttöympäristöille. [1; 2; 3.]

Yksi media queryn tärkeimmistä käyttökohteista on erilaisten tyylikäännösten, fonttikokojen ja layout-ratkaisujen määrittäminen sekä tiettyjen elementtien piilottaminen tai näyttäminen sen mukaan, millainen käyttäjän näyttö on. Näin varmistetaan, että sivusto tai sovellus näyttää hyvältä ja toimii moitteettomasti niin puhelimella, tabletilla kuin tietokoneellakin. [1; 2; 3.]

```
@media only screen and ( max-width: 600px ) {  
    .piilotettava-elementti {  
        display: none;  
    }  
}
```

Esimerkkikoodi 1. Media queryllä piilotettava elementti.

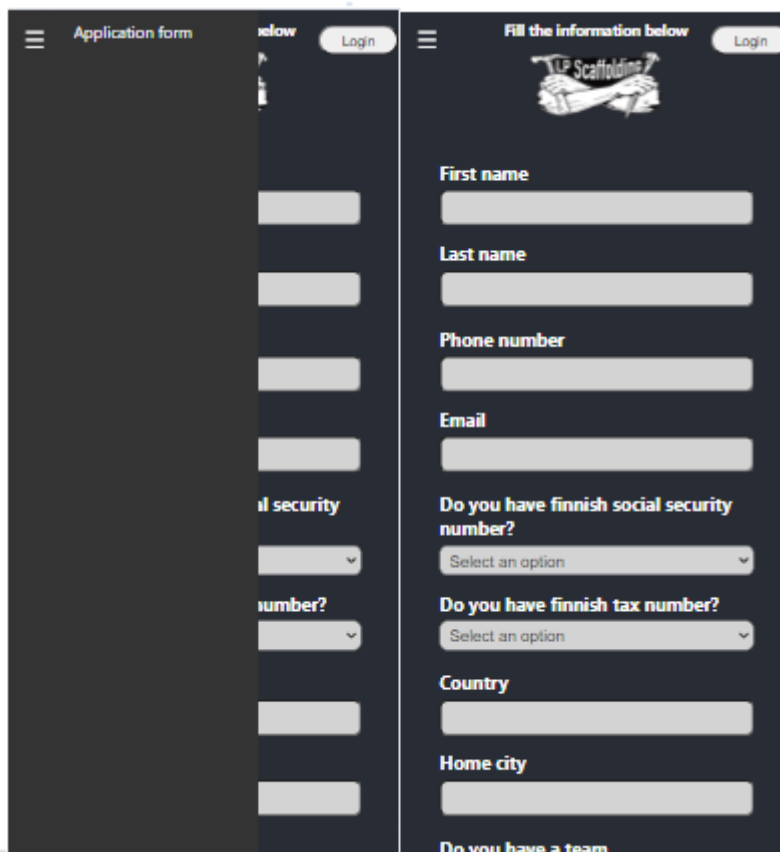
Esimerkkikoodissa 1 piilotettava-elementti -luokkaa käytetään elementteihin, jotka halutaan piilottaa näytöltä, kun näytön leveys on enintään 600 pikseliä, mikä voi vastata esimerkiksi mobiililaitteen näyttöä.

Lisäksi responsiivisen suunnittelun toteuttamiseen voi käyttää muita CSS3-ominaisuuksia, kuten flexboxia ja grid-layoutia, jotka tarjoavat tehokkaita tapoja sivustojen ja sovellusten rakentamiseen ja järjestämiseen eri näyttökoissa. Näitä tekniikoita yhdistelemällä voidaan luoda monipuolisia ja hienostuneita responsiivisia käyttöliittymiä, jotka tarjoavat käyttäjille miellyttävän ja saumattoman kokemuksen. [3; 4; 5.]

Vaikka responsiivinen suunnittelu mahdollistaa eri näyttökokoihin mukautumisen, on tärkeää säilyttää sovelluksen tai verkkosivuston visuaalinen yhtenäisyys. Brändi-identiteetin ja ulkoasun tulee pysyä tunnistettavina eri laitteilla, jotta käyttäjät tuntevat olevansa saman sovelluksen parissa, oli käytössä sitten älypuhelin, tabletti tai tietokone. Tämä saavutetaan yleensä hyödyntämällä johdonmukaisesti typografiaa, värimaailmaa, logon sijoittelua ja muita visuaalisia elementtejä, jotka ovat olennainen osa brändi-identiteettiä. [1; 2.]

Samalla on tärkeää varmistaa, että responsiivinen suunnittelu ei vain mukaudu näytön koon muutoksiin, vaan myös optimoi käyttäjäkokemuksen kunkin laitteen ominaispiirteiden ja käyttötarkoituksen mukaan. Tämä saavutetaan ymmärtämällä käyttäjien tarpeita eri alustoilla. Esimerkiksi mobiililaitteilla navigaatio on yleensä sijoitettu näytön yläreunaan jonkin valikkoa kuvaavan kuvan alle, tai vaihtoehtoisesti näytön alareunaan, jolloin se on selkeästi esillä ja helposti käytettävissä, eikä se vie olennaista näyttötilaa sivun tai applikaation muilta osilta.

React projektin navigaatioissa on käytetty painiketta, joka avaa liukuvan valikon. Tämä ratkaisu mahdollistaa sen, että navigaatio ei jatkuvasti vie näytöltä tilaa, vaan se avautuu tarvittaessa. Erityisesti tilanteissa, joissa käyttäjä ei ole aktiivisesti navigoimassa näkymillä, tämä ratkaisu optimoi näytön tilankäytön. Menetelmä on kätevä, sillä se yhdistää käytettävyyden ja tilankäytön tehokkuuden, mikä parantaa käyttäjäkokemusta sovelluksen käytössä. Navigaatiota on esitelty kuvassa 1.



Kuva 1. Kuvassa navigaatiomenetelmä, jota käytetään React-sovelluksessa.

```
.hamburger-menu {
  Position: fixed;
  Top: 0;
  Left: -250px;
  Width: 250px;
  Height: 100%;
  Background-color: #333;
  Transition: left 0.3s ease;
}

.hamburger-menu.open {
  Left: 0;
}

Const HamburgerMenu = ({ setSelectedOption }) => {
  // State to control the open/closed state of the menu
  Const [isOpen, setIsOpen] = useState(false);
```

Esimerkkikoodi 2. Hamburger-menu

Ensimmäinen osa esimerkkikoodi 2. on osa CSS-koodia, joka määrittelee ulkoasun ja käyttäytymisen ".hamburger-menu" -luokalle, joka edustaa navigaation liukuvalikkoa. Luokka on sijoitettu kiinteästi näytön yläreunaan, ja on piilotettu näytöltä antamalla "left" -ominaisuudelle arvo -250px. Kun valikko avataan, muuttuu arvo nolnaan, jolloin navigaatio liukuu näytölle. Alemmassa kuvassa on JavaScript useState -koukku, joka seuraa valikon tilaa, klikkauksella valikko avataan tai suljetaan ja tila päivittyy automaattisesti.

Sovelluksessa käytettävän lomakkeen ulkoasu on suunniteltu mukautumaan käyttäjän laitteen mukaan, jotta käyttökokemus olisi mahdollisimman sujuva ja selkeä niin tietokoneella kuin mobiililaitteillakin. Erilaiset näyttökoot ja laiteympäristöt vaikuttavat merkittävästi siihen, miten käyttäjät kokevat ja käyttävät sovellusta, joten oli tärkeää toteuttaa käyttöliittymä siten, että se palvelee eri laitteita optimaalisesti.

Tietokoneen suuremmalla näytöllä lomake on toteutettu kahden rinnakkaisen kentän layoutilla, jolloin tiedot on aseteltu vierekkäin riveittäin. Tämä lähestymistapa hyödyntää suuremman näytön tarjoamaa laajempaa tilaa tehokkaasti, mikä tekee lomakkeesta visuaalisesti selkeämmän ja nopeamman täyttää, koska käyttäjän ei tarvitse vierittää sivua niin paljon. Vierekkäin asetellut kentät auttavat myös jäsentämään tietoja ryhmittäin, jolloin lomakkeen täyttäminen tuntuu

loogiselta ja suoraviivaiselta. Tietokoneversiossa käyttäjä pystyy näkemään laajemman osan lomaketta yhdellä kertaa, mikä nopeuttaa tietojen syöttöä ja tekee siitä käyttäjälle vaivattomampaa.

Mobiiliversiossa lomake on suunniteltu siten, että kullakin rivillä näkyy vain yksi kenttä kerrallaan. Tämä johtuu mobiililaitteiden kapeammasta näyttötilasta, jonka vuoksi kahden kentän rinnakkaisasettelu voisi tehdä lomakkeesta ahtaan ja vaikeasti luettavan. Yksittäisten kenttien asettelu pystysuuntaisesti mobiililaitteissa varmistaa, että teksti ja kentät ovat riittävän suuria, jotta niitä on helppo käyttää kosketusnäytöllä. Näin lomake säilyy helposti luettavana ja käyttäjäystävällisenä myös mobiilissa, eikä käyttäjä joudu zoomaamaan tai vierittämään sivua liiallisesti.

Teknisesti tämä muutos on toteutettu hyödyntämällä responsiivista suunnittelua, joka mukauttaa sovelluksen käyttöliittymän automaattisesti käyttäjän laitteen mukaan. CSS media query -sääntöjen avulla lomakkeen layout vaihtelee näytön leveyden perusteella. Jos näytön leveys ylittää tietyn pikselimäärän, lomake mukautuu tietokoneelle optimoiduksi kahden rinnakkaisen kentän asetteluksi. Jos leveys on kapeampi, esimerkiksi mobiililaitteilla, lomake siirtyy yhden kentän asetteluun per rivi. Tällä tavoin sama sovellus mukautuu automaattisesti eri laitteille ilman, että käyttöliittymää tarvitsee erikseen koodata jokaiselle laitteelle erikseen.

Ottamalla kuvat molemmista layout-versioista, olen halunnut havainnollistaa tätä responsiivisen suunnittelun toteutusta käytännössä. Näin käyttäjälle tai lukijalle käy selvästi ilmi, kuinka sovellus mukautuu eri näyttökokoihin ja miten eri laiteympäristöissä suunnitteluratkaisut vaikuttavat käyttöliittymän käytettävyyteen. Tämä toteutus parantaa käyttäjäkokemusta, sillä se varmistaa, että lomake toimii sulavasti sekä tietokoneella että mobiililaitteilla, jolloin käyttäjät voivat täyttää lomakkeen miellyttävästi laitteesta riippumatta. Layout-versioita esitellään kuvassa 2.

Kuva 2. React sovelluksen lomake tietokoneversiona ja mobiilissa.

3.1 Kehykset

CSS3:lle on myös kehyksiä ja sovittimia, jotka mahdollistavat CSS-koodin kirjoittamisen tehokkaammin ja helpommin. CSS-kehykset ovat valmiita tyyllityskirjastoja ja -rakenteita, joita käytetään web-kehityksessä sivustojen ulkoasun ja toiminnallisuuden toteuttamiseen. Ne tarjoavat valmiita tyylejä, komponentteja ja layout-ratkaisuja, jotka nopeuttavat kehitystyötä ja varmistavat yhdenmukaisen ulkoasun eri osissa sivustoa. [6]

CSS-kehykset ovat hyödyllisiä monilla eri tasoilla web-kehityksessä. Ne voivat auttaa kehittäjiä luomaan responsiivisia sivustoja helpottamalla niiden sovittamista eri näyttökokoille ja laitteille. Lisäksi ne tarjoavat valmiita tyylejä ja komponentteja, mikä vähentää tarvetta luoda kaikki ulkoasu alusta alkaen ja säästää arvokasta kehitysaikaa. [6.]

Kehykset nopeuttavat kehitystyötä tarjoamalla valmiita komponentteja ja tyylejä, jotka ovat helposti integroitavissa projektiin. Tämä vähentää tarvetta kirjoittaa koodia alusta asti ja säästää aikaa kehittäjältä. Kehykset tarjoavat myös valmiita tyyllitoteutuksia ja teemoja, mikä helpottaa ulkoasun suunnittelua ja yhtenäistää sivuston tai sovelluksen ilmettä. Kehittäjä voi valita valmiista tyyleistä sopivat ja muokata tyylejä tarpeidensa mukaan, mikä säästää aikaa ja vaivaa. Lisäksi

kehykset ovat yleensä testattuja ja vakiintuneita ratkaisuja, mikä vähentää yhteensopivuusongelmien ja virheiden riskiä. [6; 5.]

Kehyksien dokumentaatio ja yhteisötuki tarjoavat kehittäjälle arvokasta apua ja ohjausta kehitystyön eri vaiheissa. Yhteisöt ovat aktiivisia ja tarjoavat usein vastauksia kysymyksiin ja ratkaisuja ongelmiin erilaisissa foorumeissa. Lisäksi kehüksien dokumentaatio on yleensä kattavaa ja selkeää, mikä auttaa kehittäjiä hyödyntämään niiden tarjoamia ominaisuuksia tehokkaasti.

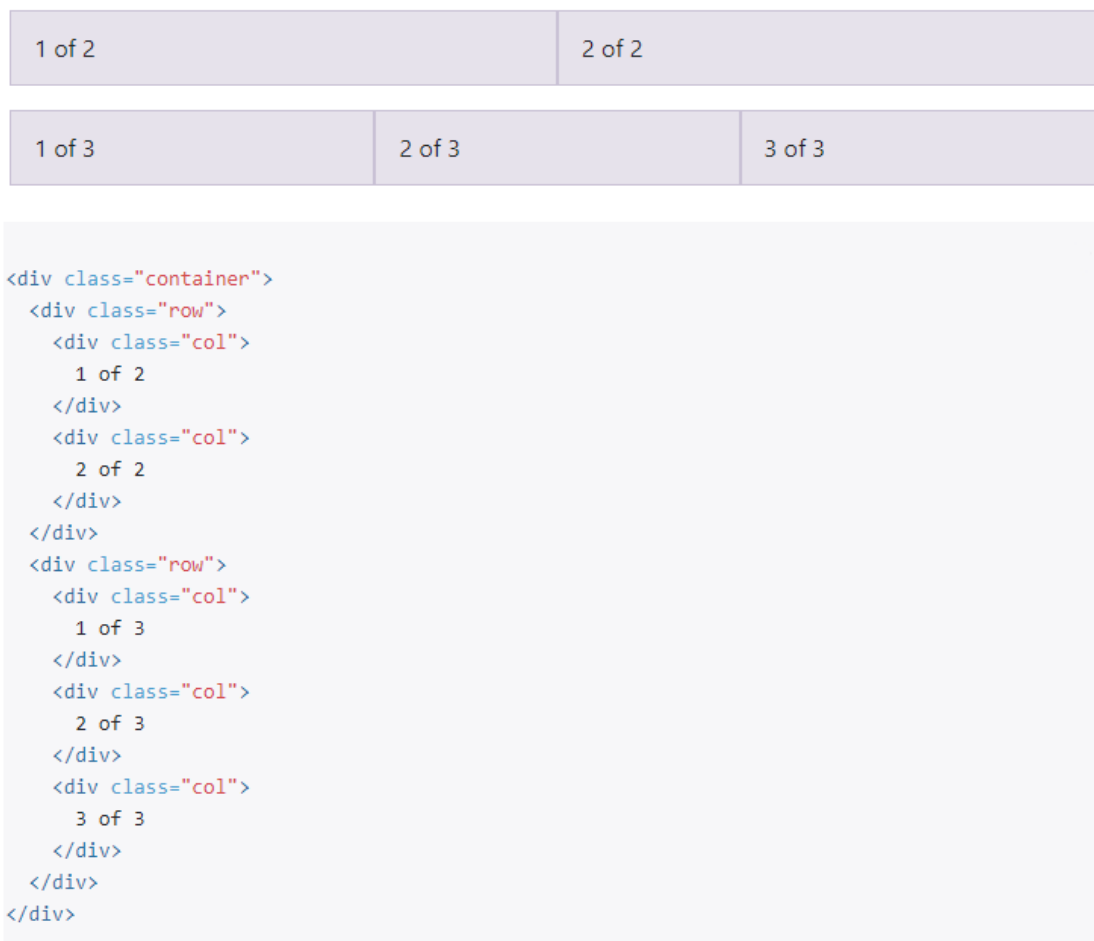
Kaiken kaikkiaan kehykset ovat arvokas työkalu web-kehityksessä, mikä nopeuttaa kehitystyötä, parantaa lopputulosta ja helpottaa kehittäjän työtä. Niiden käyttö on vakiintunut käytäntö nykyaikaisessa web-kehityksessä ja niiden merkitys kasvaa jatkuvasti kehittävän teknologisen ympäristön myötä.

Bootstrap on yksi suosituimmista CSS-kehyksistä, mikä tarjoaa laajan valikoiman valmiita tyylejä, komponentteja ja responsiivisen grid-järjestelmän. Grid-järjestelmä on suunnittelun ja layoutin organisointimenetelmä, joka helpottaa verkkosivustojen ja sovellusten rakenteen joustavuutta ja mukautuvuutta erilaisiin näyttökokoihin. Järjestelmä jakaa sivun tai sovelluksen layoutin useisiin sarakkeisiin ja riveihin, mikä helpottaa sisällön sijoittamista ja järjestämistä. Responsiivisessa grid-järjestelmässä nämä sarakkeet ja rivit mukautuvat dynaamisesti näytön koon ja resoluution mukaan. [7.]

Kuvassa 3 näkyy esimerkki Bootstrapin grid-järjestelmän käytöstä. Layout on jaettu useisiin riveihin, ja jokainen rivi sisältää sarakkeita. Jokainen sarakeista mukautuu automaattisesti näytön koon mukaan ja jakaa tilan tasaisesti rivin sisällä. Esimerkissä ensimmäisellä rivillä on kaksi saraketta ja toisella rivillä kolme saraketta.

Grid layoutin käyttö nopeuttaa kehittäjää monin tavoin. Se tekee layoutin suunnittelusta nopeampaa tarjoamalla valmiit rakenteet sarakkeille ja riveille, mikä vähentää tarvetta manuaaliselle koodin kirjoitukselle. Järjestelmä mahdollistaa myös automaattisen mukautumisen eri näyttökokoja ja laitteita varten, joten kehittäjän ei tarvitse erikseen suunnitella eri versioita eri laitteille. Lisäksi grid-

järjestelmä tarjoaa valmiit luokat sarakkeille ja riveille, mikä vähentää tarvetta ylimääräisen CSS-koodin kirjoittamiseen ja tekee koodista selkeämpää ja helpommin ymmärrettävää. Tämä kaikki yhdessä johtaa nopeampaan kehitysprosessiin ja helpottaa responsiivisen suunnittelun toteuttamista.



Kuva 3. Bootstrap grid-järjestelmä.

Foundation on toinen merkittävä CSS-kehiksen vaihtoehto, joka tarjoaa monipuolisen valikoiman komponentteja ja responsiivisen grid-järjestelmän. Se tunnetaan erityisesti joustavuudestaan ja monipuolisista ominaisuuksistaan. [8.]

```


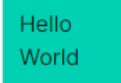
<ul class="dropdown menu" data-dropdown-menu>
<li><a href="#"> dropdown 0 </a></li>
<li><a href="#1"> dropdown 1 </a></li>
</ul>

```

Esimerkkikoodi 3. Foundation-framework

Esimerkkikoodissa 3 on käytetty Foundation-frameworkin dropdown-valikkoa. Dropdown-valikko on HTML-lista, johon on lisätty "dropdown menu" -luokka Foundationin käyttöä varten. Kuvassa on kaikki koodi, mitä kehittäjän tarvitsee kirjoittaa yksinkertaista valikkoa varten käyttäessään Foundation-frameworkia.

Bulma on moderni CSS-kehysten vaihtoehto, joka korostaa helppokäyttöisyyttä ja joustavuutta. Se tarjoaa selkeän ja modulaarisen rakenteen, mikä helpottaa sivustojen rakentamista ja muokkaamista.

#	Color	color class		background class	
	 --bulma-primary	has-text-primary	Hello World	has-background-primary	 Hello World
<pre>Color Background</pre>					

Kuva 4. Bulma CSS-kehys.

Kuvassa 4 olevassa koodissa käytetään Bulma CSS-kehystä, ja siinä määritellään erilaisia tyyliluokkia käytten Bulma-tyylitiedostossa määriteltyjä värejä ja taustavärejä. Bulma-primary luokka määrittelee ensisijaisesti värin Bulma-tyylitiedostosta, mikä on yleensä sivuston tai sovelluksen pääväri. has-text-primary ja has-background-primary -luokat puolestaan määrittelevät tekstin jaustan värin vastaavasti. Esimerkiksi has-text-primary asettaa tekstin värin vastaamaan ensisijaiseksi määriteltyä väriä. [9.]

Tailwind CSS poikkeaa perinteisistä CSS-kehyksistä lähestymistavallaan, joka perustuu pieniin abstraktioihin ja luokkiin (esimerkkikoodi 4). Kehittäjät voivat rakentaa omat tyylit yksittäisten luokkien avulla, mikä tarjoaa suuren joustavuuden ja räätälöintävyyden. [10.]

```
// Width of 16 by default, 32 on medium screens, and 48 on large screens
```

```
</img>
```

Esimerkkikoodi 4. Tailwindin valmiiksi määritellyt luokat.

Esimerkkikoodi 4:ssä on käytetty Tailwindin responsiivisia leveyden määritteleviä luokkia. Ne asettavat kuvan leveyden eri näyttökokojen mukaan. Kun sivustoa tai sovellusta käytetään eri kokoisilla näytöillä kuvan leveys, muuttuu automaattisesti vastaamaan käytettyä näyttökokoja. Tämä auttaa varmistamaan, että kuva näyttää hyvältä ja skaalautuu oikein eri laitteilla ja näytöillä. Samalla luokkien käyttö on kehittäjälle yksinkertaista ja vähentää tarvittavien media queryjen manuaalista kirjoittamista.

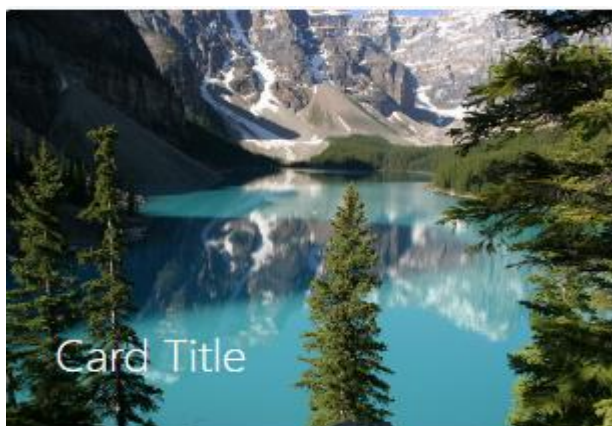
SM	640px	@media (min-width: 640px) { ... }
MD	768px	@media (min-width: 768px) { ... }
LG	1024px	@media (min-width: 1024px) { ... }
XL	1280px	@media (min-width: 1280px) { ... }
2XL	1536px	@media (min-width: 1536px) { ... }

Taulukossa esitetään Tailwind CSS-kehiksen yleisiä näyttökokojen breakpoint arvoja, jotka toimivat suuntaviivoina komponenttien koon määrittelyssä eri kokoisilla näytöillä. Nämä arvot antavat kehittäjille joustavuutta sovittaa komponenttien ulkoasua ja käyttäytymistä eri näyttöko'oilte optimaalisen käyttäjäkokemuksen takaamiseksi. Lyhenteitä ("sm, md, lg, xl ja 2xl"), käytetään luokkien määrittelyssä, mikä mahdollistaa esimerkiksi kuvien koon säätämisen eri näyttökokoja varten. Kehittäjällä on vapaus muuttaa näitä arvoja omien tarpeidensa mukaisesti, mikäli siihen on tarvetta tai jos halutaan tarkemmin hienosäätää komponenttien käyttäytymistä eri tilanteissa.

Esitelty lähestymistapa antaa kehittäjille mahdollisuuden hallita responsiivisen suunnittelun yksityiskohtia ja varmistaa, että käyttäjät saavat miellyttävän ja toimivan kokemuksen sivustosta tai sovelluksesta riippumatta, millä laitteella tai näytöllä he sitä käyttävät.

Materialize CSS perustuu Googlen Material Design- suunnitteluohjeisiin ja tarjoaa valmiita tyyliä ja komponentteja, jotka on suunniteltu noudattamaan Material Designin periaatteita. Se tarjoaa modernin ja tyylikkään ulkoasun sivustolle. [11.]

Koodiesimerkki 5:ssä, sekä kuvassa 5 on esimerkki Materialize CSS-kehyksellä tehdystä kortista, joka sisältää kuvan, otsikon, sisällön, sekä toimintopainikkeen. Kortin alla on koodi, jolla kortti on toteutettu. Koodissa on määritelty kortin koko ("col s12 m7"), eli kortti vie koko rivin pienillä näytöillä ("s12") ja 7/12 rivistä keskisuurilla näytöillä ("m7"). Seuraavilla riveillä määritellään luotu elementti, joka sisältää kuvan, otsikon, sisällön sekä toimintopainikkeen. Kortin kuva ja otsikko asetetaan "`<div class="card-image">`" sisään ja kortin sisältö "`<div class="card-content">`" sisään.



I am a very simple card. I am good at containing small bits of information. I am convenient because I require little markup to use effectively.

[THIS IS A LINK](#)

Kuva 5. Kortti materialize CSS-kehyksellä.

```
<div class="row">
<div class="col s12 m7">
<div class="card">
<div class="card-image">
</img>
<span class="card-title">Card Title</span>
</div>
<div class="card-content">
  <p> I am a very simple card... </p>
<div class="card-action">
  <a href="#">This is a link</a>
</div>
</div>
</div>
```

Koodiesimerkki 5. Kortti materialize CSS-kehyksellä.

JavaScript on yleisesti käytetty ohjelmointikieli, jota käytetään verkkosivujen ja -sovellusten kehittämisessä. Se on monipuolinen ja dynaaminen kieli, joka toimii selaimessa suorittaen koodia käyttäjän laitteella. JavaScriptillä voidaan

manipuloida verkkosivun rakennetta, lisätä interaktiivisuutta, hallita käyttäjän toimintoja ja kommunikoida verkon kanssa. Se on olennainen osa modernia web-kehitystä, ja sen avulla voidaan luoda monimutkaisia sovelluksia ja toteuttaa erilaisia toiminnallisuuksia, kuten animaatioita, lomakkeita, pelejä ja verkkopalveluita. Lisäksi JavaScriptiä voidaan käyttää myös palvelin puolella, esimerkiksi Node.js ympäristössä, mikä mahdollistaa koko web-sovelluksen kehittämisen yhdellä kielellä. [12.]

Nämä responsiivisen suunnittelun peruseräatteen ovat avainasemassa varmistamassa, että mobiilisovellus tai verkkosivu tarjoaa optimaalisen käyttäjäkokemuksen kaikilla laitteilla. Niiden huomioiminen suunnittelutyössä auttaa varmistamaan, että sovellus toimii saumattomasti ja tarjoaa käyttäjille parhaan mahdollisen kokemuksen.

4 Käyttäjäkokemus: merkitys ja elementit

Käyttäjäkokemus (UX) kuvaa käyttäjän kokemusta ja tunnetta vuorovaikutuksesta tuotteen, palvelun tai järjestelmän kanssa. Se on keskeinen käsite suunnittelussa, sillä se vaikuttaa suoraan siihen, kuinka käyttäjät kokevat ja arvioivat käyttökokemuksen laadun. Tässä osiossa käsitellään käyttäjäkokemuksen merkitystä ja keskeisiä elementtejä, jotka vaikuttavat siihen.

4.1 Käyttäjäkokemuksen merkitys

Käyttäjäkokemus on olennainen tekijä tuotteen tai palvelun menestyksessä, sillä se vaikuttaa suoraan käyttäjän tyytyväisyyteen, sitoutumiseen ja jatkuvuuteen. Positiivinen käyttäjäkokemus on voimakas kannustin käyttäjien sitoutumiseen ja uskollisuuteen. Tyytyväiset käyttäjät ovat todennäköisemmin valmiita suosittelemaan tuotetta tai palvelua ystävilleen ja verkostolleen, mikä edistää brändin mainetta ja leviämistä positiivisen suullisen markkinoinnin kautta. Toisaalta negatiivinen käyttäjäkokemus voi johtaa käyttäjät hylkäämään sovelluksen sekä vähentää yrityksen uskottavuutta markkinoilla. [1; 13; 14; 15.]

Hyvä käyttäjäkokemus ei ole pelkästään asiakkaiden tyytyväisyyden lähde, vaan sillä on myös suoria vaikutuksia yrityksen taloudelliseen menestykseen. Positiivinen käyttäjäkokemus voi parantaa tuotteen tai palvelun tehokkuutta ja käytettävyyttä, mikä puolestaan lisää asiakkaiden pysyvyyttä ja auttaa hankkimaan uusia asiakkaita. [1; 13; 14; 15.]

Hyvä esimerkki huonosta käyttäjäkokemuksesta on Amazonin Prime Video palvelu, jonka käyttöliittymää kritisoitiin pitkään sen sekavuudesta. Elokuvien ja TV-ohjelmien löytäminen oli haastavaa, ja jäsenille ilmaiset ja maksulliset sisällöt sekoittuivat helposti. Huono käyttöliittymä johti siihen, että lukuisilla keskustelupalstoilla ihmiset suosittelivat jättämään tuotteen tilaamatta ja valitsemaan mieluummin jonkin toisen suoratoistopalvelun.

Nyt Amazon on julkaissut täysin uuden käyttöliittymän, joka tekee navigoinnista paljon intuitiivisempaa. Pystysuuntaiset navigointikuvakkeet ja karusellissa esitetyt uusimmat tarjoukset on lisätty korjaamaan käyttöliittymän ongelmia. Nämä pystysuuntaiset navigointikuvakkeet helpottavat käyttäjiä löytämään haluamansa sisällön nopeammin ja helpommin, kun valikossa voi liikkua ylös ja alaspäin sivuttain liikkumisen sijaan.

Uuden käyttöliittymän myötä on myös otettu käyttöön uusia visuaalisia vihjeitä, jotka selkeästi osoittavat, mitkä videot sisältyvät käyttäjän Prime-jäsenyyteen (merkitty sinisellä tarkistusmerkillä) ja mitkä ovat vuokrattavissa, ostettavissa tai tilattavissa (merkitty kultaisella ostoskassikuvakkeella). Lisäksi kotinäkömään My Subscriptions rivin lähellä voi nyt päästä kaikkiin Prime-jäsenyyden sisältyviin videoihin vain yhdellä napsautuksella. Nämä parannukset ovat merkittävästi parantaneet käyttäjäkokemusta ja saaneet positiivista palautetta käyttäjiltä. [16.]

Insinööriön React-projektissa toteutettiin konkreettinen parannus käyttäjäkokemukseen työnhakijatietojen hallintaa varten. Alkuperäisessä versiossa työnhakijoiden tietoja pystyi tarkastelemaan vain sovelluksen omalla Excel-tyylisellä sivulla, joka sisälsi rajalliset haku ehdot ja rajoitti tiedon käsittelymahdollisuuksia.

Tämä ratkaisu tarjosi perustason toiminnallisuudet, mutta ei täysin vastannut asiakkaan tarpeita tiedon jatkokäsittelyn ja analyysin kannalta.

Sovelluksen testaamisesta saaman palautteen perusteella huomasimme, että asiakkaalle oli tärkeää voida siirtää tietoja omiin järjestelmiinsä ja muokata niitä itselleen tutussa ympäristössä. Sovellukseen päätettiin lisätä ominaisuus, jonka avulla käyttäjä voi ladata joko koko tietokannan tai hakuerojen avulla suodatetut työnhakijatiedot Excel-tiedostona. Tämän parannuksen ansiosta asiakas pystyi käyttämään itselleen tuttua Exceliä jatkokäsittelyyn, mikä lisäsi työn tehokkuutta ja sujuvuutta.

Ratkaisu osoittautui merkittäväksi käyttäjäkokemuksen parantajaksi: asiakas koki sovelluksen intuitiivisemmaksi ja joustavammaksi työkaluksi, joka mukautui heidän omiin tarpeisiinsa. Lisäksi se tarjosi käyttäjille mahdollisuuden analysoida ja hyödyntää dataa omalla tavallaan, mikä toi lisäarvoa projektin kokonaisuudelle.

4.2 Käyttäjäkokemuksen elementit

Käyttäjäkokemuksen laatu muodostuu useista eri elementeistä, jotka vaikuttavat siihen, kuinka käyttäjä kokee vuorovaikutuksen tuotteen tai palvelun kanssa.

4.2.1 Käytettävyys

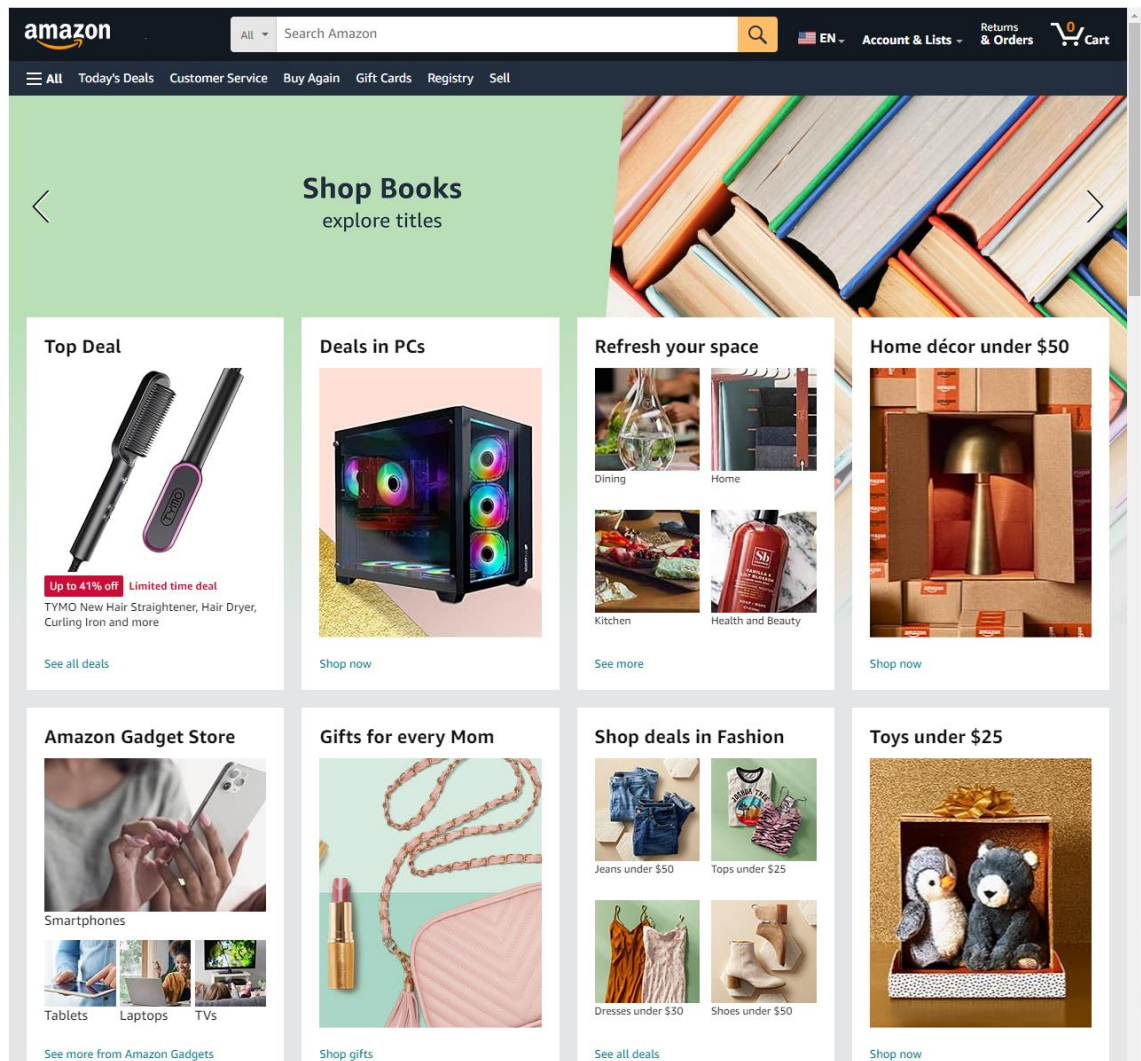
Käytettävyys kattaa tuotteen tai palvelun helppokäyttöisyyden ja tehokkuuden. Hyvä käytettävyys on olennainen osa positiivista käyttäjäkokemusta. Esimerkiksi selkeä navigointi, yksinkertaiset käyttöliittymät ja intuitiiviset toiminnot ovat kaikki tekijöitä, jotka parantavat käytettävyyttä ja siten käyttäjäkokemusta. [17; 18; 19.]

Käytettävyys on tärkeää, koska se vaikuttaa suoraan siihen, kuinka helposti ja sujuvasti käyttäjä voi saavuttaa tavoitteensa käyttäessään tuotetta. Esimerkiksi

verkkosivuton helppo navigointi ja selkeästi merkityt toiminnot voivat auttaa käyttäjää löytämään tarvitsemansa tiedot nopeasti ja vaivattomasti. [17; 18; 19.]

Yksi hyvä esimerkki hyvästä käytettävyydestä on Amazonin verkkokauppa, joka tarjoaa selkeän ja intuitiivisen käyttöliittymän, joka helpottaa tuotteiden etsimistä ja ostamista. Amazon on optimoinut sivustonsa navigaation ja hakutoiminnon, jotta käyttäjät voivat löytää haluamansa tuotteet nopeasti ja vaivattomasti. Lisäksi Amazon tarjoaa asiakkailleen selkeitä ohjeita ja vaiheittaisia prosesseja ostotapahtuman aikana, mikä tekee ostamisesta sujuvaa ja miellyttävää.

Kuten kuvassa 6 näkyy, Amazonin verkkokaupan etusivu tarjoaa selkeän ja intuitiivisen käyttöliittymän, jonka avulla on helppo etsiä ja ostaa tuotteita. Sivusto on optimoitu navigaation ja hakutoiminnon osalta, jotta käyttäjät löytävät haluamansa tuotteet nopeasti ja vaivattomasti.



Kuva 6. Kuva Amazonin verkkokaupan etusivusta [20].

Toisaalta huono käytettävyys voi johtaa käyttäjän turhautumiseen ja jopa tuotteen tai palvelun hylkäämiseen. Esimerkiksi verkkosivusto, jolla on monimutkainen ja epäselvä navigointi, voi vaikeuttaa käyttäjän tavoitteiden saavuttamista ja johtaa sivuston hylkäämiseen. Lisäksi liian monimutkaiset tai piilotetut toiminnot voivat aiheuttaa käyttäjälle sekaannusta ja epämukavuutta.

Hyvä käytettävyys on keskeinen osa positiivista käyttäjäkokemusta ja edellyttää huolellista suunnittelua ja jatkuvaa käyttäjäpalautteen analysointia jolla varmistetaan, että tuote tai palvelu on helpokäyttöinen ja tehokas käyttäjilleen.

4.2.2 Esteettisyys

Visuaalinen suunnittelu ja ulkoasu vaikuttavat käyttäjän ensivaikutelmaan ja kokemukseen. Hyvä esteettinen suunnittelu voi lisätä käyttäjän sitoutumista ja mielihyvää useilla tavoilla.

Visuaalisesti houkutteleva käyttöliittymä voi herättää positiivisia tunteita käyttäjässä ja tehdä tuotteesta tai palvelusta houkuttelevamman käyttää. Esimerkiksi selkeä värien käyttö, harkittu typografia ja hyvin suunnitellut kuvakkeet voivat tehdä käyttöliittymästä miellyttävän katsella. [21; 22.]

Hyvä esteettinen suunnittelu voi helpottaa tiedon hahmottamista ja navigointia. Esimerkiksi visuaaliset hierarkiat ja selkeästi erotellut osiot voivat auttaa käyttäjää hahmottamaan sivuston rakenteen ja löytämään tiedot nopeasti. [21; 22.]

Visuaalisella hierarkialla tarkoitetaan käyttöliittymän tai graafisen suunnittelun elementtien järjestämistä ja korostamista niiden tärkeyden tai merkityksen mukaan. Se auttaa käyttäjiä hahmottamaan ja ymmärtämään sisältöä, sekä ohjaa heitä löytämään tärkeät tiedot tai toiminnot helpommin. Visuaalinen hierarkia voidaan muodostaa esimerkiksi käyttämällä erilaisia kokoja, fontteja, värejä ja asettelua korostamaan tärkeitä elementtejä ja erottamaan ne vähemmän tärkeistä. Tämä tekee käyttöliittymästä selkeämmän ja helpommin luettavan käyttäjille. [23.]

Hyvä esimerkki esteettisesti onnistuneesta suunnittelusta on Applen tuotteiden käyttöliittymät, kuten iOS ja macOS. Applen käyttöliittymät ovat tunnettuja selkeydestään, tyylikkäästä ulkoasustaan ja yksinkertaisuudestaan. Ne käyttävät harkittuja värejä ja selkeitä kuvakkeita, mitkä tekevät niistä miellyttäviä käyttää.

Huono esteettinen suunnittelu voi heikentää käyttäjäkokemusta monin eri tavoin. Esimerkiksi liian monimutkainen tai sotkuinen ulkoasu voi häiritä käyttäjää ja vaikeuttaa tiedon löytämistä. Liiallinen visuaalinen häly voi myös väsyttää käyttäjän silmiä ja vähentää käyttökokemuksen miellyttävyyttä.

Vaikka esteettisyys on yleensä tärkeä osa käyttäjätyytyväisyyttä, tässä sovelluksessa se ei ollut asiakkaan toiveiden kärjessä. Projektin pääpaino oli toimivassa ja tehokkaassa sovelluksessa, joka suorittaa tarvittavat toiminnot vaivattomasti ja mahdollistaa tietojen käsittelyn asiakkaan toivomalla tavalla. Käyttöliittymän suunnittelussa keskityttiin selkeyteen ja yksinkertaisuuteen, mikä takasi, että käyttäjät pystyivät suorittamaan tehtävänsä nopeasti ja ilman ylimääräisiä esteitä. Tämä ratkaisukeskeinen lähestymistapa varmisti, että sovellus täytti juuri ne vaatimukset, jotka olivat asiakkaalle olennaisimpia – suoraviivaisen ja helpokäyttöisen työkalun, joka mahdollistaa tehokkaan työskentelyn.

4.2.3 Suorituskyky

Suorituskyvyllä tarkoitetaan tuotteen tai palvelun kykyä toimia nopeasti ja tehokkaasti käyttäjän tarpeiden mukaan. Hyvä suorituskyky on olennainen osa positiivista käyttäjäkokemusta, kun taas heikko suorituskyky voi johtaa käyttäjän turhautumiseen ja käyttökokemuksen heikentymiseen. [24.]

Hyvä esimerkki suorituskyvystä voisi olla verkkosivusto, joka latautuu nopeasti ja reagoi käyttäjän toimiin välittömästi. Käyttäjä pystyy navigoimaan sivustolla sujuvasti ilman merkittäviä odotusaikoja

Toisaalta huono esimerkki suorituskyvystä voisi olla mobiilisovellus, joka kärsii jatkuvista kaatumisista tai hidastumisista. Tällainen sovellus aiheuttaa käyttäjälle turhautumista ja voi johtaa siihen, että käyttäjä lopettaa sovelluksen käytön kokonaan.

Hyvä suorituskyky on tärkeää erityisesti nykypäivänä, jolloin käyttäjät odottavat nopeaa ja saumatonta kokemusta käyttämistään tuotteista ja palveluista. Tehokas suorituskyky varmistaa, että käyttäjät voivat hyödyntää tuotetta tai palvelua ilman turhautumista ja saavat positiivisen käyttökokemuksen.

Sovelluksen suorituskyvystä onnistuttiin saamaan erittäin hyvä, mikä näkyy käyttäjille nopeana ja sulavana käyttökokemuksena. Tämä saavutettiin

keskittymällä useisiin suorituskkyä parantaviin ratkaisuihin. Ensinnäkin, sovellukseen ei tarvittu kuvia, ja käyttöliittymän esteettinen ilme pidettiin pelkistettynä, mikä vähensi tarpeettomien resurssien lataamista. Suoraviivainen ja selkeä ulkoasu varmistaa, että sovellus keskittyy olennaiseen eikä vie turhaan latausaikaa visuaalisilla elementeillä.

Koodi pyrittiin kirjoittamaan mahdollisimman puhtaasti ja yksinkertaisesti, mikä vähentää virheiden ja pullonkaulojen riskiä. Projektin edetessä suorituskkyä parannettiin myös optimoimalla tietokantakyselyitä: ylimääräisiä SQL-pyyntöjä poistettiin, jotta käyttäjälle näytetään vain olennaiset tiedot ilman, että tietoja haetaan jatkuvasti tietokannasta. Tämä ratkaisu vähentää tietokantayhteyksien kuormitusta ja parantaa sovelluksen responssinopeutta merkittävästi. Kaiken kaikkiaan sovellus on rakennettu niin, että se latautuu nopeasti ja tarjoaa käyttäjälle juuri sen tiedon, mitä he tarvitsevat kyseisellä hetkellä.

4.2.4 Luotettavuus

Luotettavuus viittaa siihen, että käyttäjä voi luottaa tuotteen tai palvelun toimivan odotetulla tavalla ja tarjoavan johdonmukaisen käyttökokemuksen. Hyvä luotettavuus on keskeinen osa positiivista käyttäjäkokemusta, kun taas epäluotettavuus voi johtaa käyttäjän turhautumiseen ja luottamuksen menettämiseen. [25.]

Hyvä esimerkki luottavuudesta voisi olla verkkopalvelu, joka harvoin kohtaa teknisiä ongelmia tai käyttökatkoksia. Käyttäjät voivat luottaa siihen, että palvelu on käytettävissä aina tarvittaessa ja toimii odotetulla tavalla.

Huonona esimerkkinä voitaisiin pitää mobiilisovellusta, joka kaatuu säännöllisesti tai toimii epäjohdonmukaisesti eri tilanteissa. Tällainen epäluotettavuus aiheuttaa käyttäjille turhautumista ja johtaa helposti siihen, että käyttäjä menettää luottamuksen sovellukseen.

Luotettavuus on tärkeää, koska se luo käyttäjille turvallisen ja miellyttävän kokemuksen tuotteen tai palvelun käytöstä. Kun käyttäjät voivat luottaa siihen, että tuote tai palvelu toimii odotetulla tavalla, he todennäköisemmin sitoutuvat siihen ja palavat käyttämään sitä uudelleen tulevaisuudessa. [25.]

4.2.5 Relevantti sisältö

Relevantilla sisällöllä viitataan siihen, että tuotteen tai palvelun tarjoama informaatio ja materiaali vastaavat käyttäjät tarpeisiin ja tavoitteisiin. Hyvä relevantti sisältö tarjoaa käyttäjälle merkityksellistä ja hyödyllistä tietoa, mikä parantaa käyttäjäkokemusta.

Hyviä esimerkkejä relevantista sisällöstä voivat olla esimerkiksi verkkokaupan tuotesivut, jotka tarjoavat selkeät kuvaukset ja ominaisuudet tuotteista, sekä asiakasarvioita ja suosituksia, jotka auttavat käyttäjiä tekemään ostopäätöksiä.

Relevanttiuden puolesta huonona voitaisiin pitää sivua, joka tarjoaa käyttäjälle liikaa tarpeetonta ja epäolennaista tietoa, mikä vaikeuttaa käyttäjää löytämästä etsimäänsä tai ymmärtämään olennaista sisältöä.

Relevantti sisältö on tärkeää, koska se auttaa käyttäjiä saavuttamaan tavoitteensa ja ratkaisemaan ongelmiaan tehokkaasti. Kun käyttäjät kokevat saavansa tarvitsemaansa tietoa ja apua, he todennäköisemmin sitoutuvat tuotteeseen tai palveluun ja jatkavat sen käyttöä. Lisäksi relevantti sisältö auttaa luomaan positiivista käyttäjäkokemusta ja lisää käyttäjän tyytyväisyyttä.

4.2.6 Saavutettavuus

Saavutettavuus viittaa siihen, että tuotteen tai palvelun käyttö on mahdollista kaikille käyttäjille riippumatta heidän fyysisistä tai teknisistä rajoitteistaan. Tämä tarkoittaa esimerkiksi sitä, että tuotteeseen tai palveluun on helppo päästä käsiin, siinä on selkeä käyttöliittymä ja se tarjoaa vaihtoehtoisia tapoja saavuttaa tavoitteensa eri käyttäjäryhmille. [26.]

Hyvänä esimerkkinä saavutettavuudesta voitaisiin pitää verkkosivustoa, joka tarjoaa vaihtoehtoisia tekstivaihtoehtoja kuvilla varustelluille sisällöille näkövammaisille käyttäjille. Lisäksi sivusto voi tarjota selkeän navigaation ja mahdollisuuden käyttää näppäimistöä hiiren sijaan liikkumiseen, mikä helpottaa niiden käyttöä, joilla on motorisia rajoitteita.

Huono esimerkki saavutettavuudesta voisi olla verkkosivusto tai mobiilisovellus, joka vaatii tietyn selaimen tai laitteen käyttöä. Esimerkiksi jos verkkosivusto toimii vain tietyillä selaimilla tai mobiilisovellus on saatavilla vain tiettyihin käyttöjärjestelmiin, se rajoittaa käyttäjien mahdollisuuksia päästä sisällön tai palvelun pariin. Tämä aiheuttaa turhautumista ja haittaa käyttökokemusta niille käyttäjille, joilla ei ole pääsyä vaadittuihin teknisiin välineisiin tai ohjelmistoihin.

Saavutettavuus on tärkeää, koska se varmistaa, että kaikki käyttäjät voivat hyödyntää tuotetta tai palvelua täysimääräisesti. Se lisää käyttäjien mukavuutta ja helpottaa käyttöä eri tilanteissa. Lisäksi saavutettavuus voi auttaa yrityksiä saavuttamaan laajemman käyttäjäkunnan ja vahvistamaan brändinsä arvoja, kuten sisällyttämällä kaikki käyttäjät yhdenvertaisesti palveluihinsa. [26.]

Vaikka saavutettavuus ja käyttäjäkokemus ovat yleensä keskeisiä asioita verkkosivustojen tai sovellusten suunnittelussa, tässä projektissa niillä ei ollut yhtä suurta merkitystä. Sovellus on suunniteltu vain yhden asiakkaan käyttöön, eikä sen tarkoituksena ole palvella laajaa tai monimuotoista käyttäjäkuntaa. Tietoja syöttävät henkilöt kutsutaan sovellukseen henkilökohtaisesti ja pääsevät siihen käsiksi suoraan lähetetyn linkin kautta. Tämän vuoksi laajemmat saavutettavuusvaatimukset, kuten vaihtoehtoiset navigointitavat tai esteettömyysratkaisut, eivät olleet tässä projektissa tarpeellisia. Suunnittelun pääpaino oli toimivuudessa ja tehokkuudessa asiakkaan tarpeiden mukaisesti.

Yhteenvedon voidaan todeta, että käyttäjäkokemus on monimutkainen kokonaisuus, joka muodostuu useista eri elementeistä. Hyvä käyttäjäkokemus vaatii huolellista suunnittelua ja jatkuvaa arviointia, jotta voidaan varmistaa käyttäjien tarpeiden ja odotusten täyttäminen.

5 Psykologian vaikutus

Responsiivisessa suunnittelussa käyttöliittymän psykologisilla tekijöillä on merkittävä rooli käyttäjäkokemuksen muodostumisessa eri laitteilla. Seuraavassa osiossa käsitellään värien, fonttien ja visuaalisten elementtien vaikutusta responsiiviseen suunnitteluun, sekä niiden potentiaalista vaikutusta käyttäjän käyttäytymiseen ja kokemukseen.

5.1 Värien vaikutus

Värit voivat vaikuttaa voimakkaasti käyttäjän mielialaan, huomioon ja toimintaan eri laitteilla. Kontrastiset värit voivat auttaa erottamaan tärkeitä elementtejä pienillä mobiililaitteiden näytöillä, kun taas suuremmilla näytöillä hienovaraisempi väripaletti voi tarjota miellyttävämmän visuaalisen kokemuksen. Lisäksi värien psykologiset vaikutukset on otettava huomioon suunnittelussa käyttäjäkokemuksen optimoimiseksi. [27.]

Värien psykologiset vaikutukset ovat merkittävä tekijä suunnittelussa, sillä ne voivat vaikuttaa ihmisten mielialaan, käyttäytymiseen ja kokemukseen ympäristöstä. Esimerkiksi punainen voi herättää voimakkaita tunteita, kuten intohimoa tai vaaraa, samalla kun sininen liitetään usein rauhaan ja luottamukseen. Vihreä voi edistää tasapainoa ja harmoniaa, kun taas keltainen voi herättää iloisuutta ja energisyyttä. Musta voi symboloida voimaa ja arvovaltaa, mutta myös synkkyyttä, jos sitä käytetään liikaa. Näiden psykologisten vaikutusten ymmärtäminen auttaa suunnittelijoita luomaan haluttuja tunnetiloja ja reaktioita käyttäjissä, parantaen siten kokonaisvaltaista käyttäjäkokemusta. [27.]

Värit eivät vain lisää visuaalista kiinnostavuutta, vaan ne myös parantavat käyttäjäkokemusta ja käytettävyyttä. Ne auttavat houkuttelemaan käyttäjiä ja ohjaamaan heidän huomioitaan tärkeisiin elementteihin, parantaen siten sivuston tai sovelluksen tehokkuutta. [27.]

5.2 Fonttien vaikutus

Typografia on keskeisessä asemassa digitaalisessa suunnittelussa, vaikuttaen merkittävästi käyttäjäkokemukseen. Oikea valinta fonttien, kokojen, välien ja muotoilun suhteen voi parantaa lukukelpoisuutta, herättää tunteita ja välittää brändipersonallisuutta. [21.]

Selkeä ja luettava typografia on olennaista optimaalisen lukukelpoisuuden varmistamiseksi. Hyvin valittu fontti asianmukaisella kirjainten välillä ja rivivälillä varmistaa, että käyttäjät voivat helposti kuluttaa sisältöä ilman räsitusta. Sans-serif-fontteja suositellaan yleisesti digitaalisissa käyttöliittymissä niiden siistin ja modernin ulkonäön vuoksi, mutta fontin valintaan tulisi vaikuttaa suunnittelun konteksti ja kohdeyleisö. [21.]

Hierarkian muodostaminen on tärkeää käyttäjien huomion ohjaamiseksi ja tietojen priorisoimiseksi. Erilaisia fonttikokoja, -tyylejä ja -painoja käyttämällä voidaan korostaa tärkeitä kohtia, jotka helpottavat käyttäjiä sisällön lukemisessa. [21.]

Fontit voivat välittää tunteita ja brändin persoonallisuutta, ja niiden valinta tulisi sovittaa brändin identiteettiin ja haluttuihin tunteisiin. Saavutettavuuden ja inklusiivisuuden huomioiminen tarjoaa mahdollisuuden parantaa käyttäjäkokemusta laajemmalle yleisölle. [21.]

Yhtenäisyys typografiassa brändin eri alustoilla edistää brändin tunnistettavuutta ja vahvistaa sen identiteettiä. Lisäksi fonttien valinnalla voi olla vaikutusta latausnopeuteen ja suorituskykyyn, joten on tärkeää valita kevyitä fontteja ja optimoida niiden käyttöä.

5.3 Visuaalisten elementtien vaikutus

Kuvat, videot ja ikonit ovat keskeisiä elementtejä verkkosivustojen ja sovellusten suunnittelussa, ja ne voivat vaikuttaa merkittävästi käyttäjän kokemukseen ja toimintaan eri laitteilla. Esimerkiksi, korkealaatuiset kuvat voivat lisätä

käyttäjän sitoutumista ja kiinnostusta tuotteeseen tai palveluun, kun taas liian suuret kuvat voivat hidastaa sivuston latausaikoja mobiililaitteilla. Lisäksi visuaalisten elementtien sijoittelu ja käyttöliittymän rakenne voivat vaikuttaa merkittävästi käyttäjän navigointiin ja toimintaan eri laitteilla. [28.]

React-projektissa käyttäjäkokemusta parannettiin harkitulla värien ja visuaalisten elementtien käytöllä. Sivuston värimaailma koostui pääosin tummansinisestä/harmaasta, valkoisesta ja mustasta, jotka yhdessä loivat selkeän ja rauhallisen ilmeen. Tämä väripaletti mahdollisti hyvän luettavuuden ja vähensi visuaalista hälyä, mikä sopi sovelluksen pelkistettyyn ja toiminnalliseen tarkoitukseen.

Erityistilanteissa, kuten adminin käytössä olevissa toiminnoissa, värien käyttöä laajennettiin korostamaan tärkeimpiä toimintoja. Esimerkiksi tietojen poistopainikkeessa käytettiin punaista väriä luomaan visuaalinen varoitus siitä, että toiminto on peruuttamaton ja voi vaikuttaa merkittävästi tietoihin. Vastaavasti vihreä väri valittiin tietojen latauspainikkeeseen korostamaan positiivista ja turvallista toimintaa. Nämä värivalinnat eivät ainoastaan parantaneet käytettävyyttä, vaan auttoivat käyttäjiä ymmärtämään toimintojen merkityksen intuitiivisesti.

Tämä suunnitteluratkaisu yhdisti yksinkertaisuuden ja visuaalisen selkeyden, mikä teki sovelluksesta helppokäyttöisen ja tehokkaan. Värien johdonmukainen ja psykologisesti harkittu käyttö tukee käyttäjän navigointia ja auttaa välttämään virheitä kriittisissä tilanteissa. Näin projektissa hyödynnettiin värien suunnittelullisia ja psykologisia vaikutuksia parantamaan kokonaisvaltaista käyttäjäkokemusta.

Yhteenvetona voidaan todeta, että värien, fonttien ja visuaalisten elementtien huolellinen valinta ja käyttöliittymän rakenteen harkittu suunnittelu voivat parantaa merkittävästi responsiivisen suunnittelun käyttäjäkokemusta eri laitteilla. Jatkuvan testauksen ja iteroinnin avulla voidaan varmistaa, että käyttäjät saavat parhaan mahdollisen kokemuksen ja voivat saavuttaa tavoitteensa sujuvasti erilaisissa käyttöympäristöissä.

6 Haasteet ja ratkaisut

Responsiivisen suunnittelun toteuttaminen on monimutkainen prosessi, joka tuo mukanaan lukuisia haasteita, mutta samalla tarjoaa useita ratkaisuja näiden haasteiden voittamiseen. Suurimmat haasteet liittyvät muun muassa näyttökojen ja resoluutioiden hallintaan, sisällön priorisointiin, navigoinnin ja käytettävyyden optimointiin mobiililaitteille sekä kuvien ja medioiden optimointiin. [29; 30.]

Näyttökokojen ja resoluutioiden hallinnassa keskeistä on käyttää joustavia rakenteita, kuten prosentiosuuksia, suhteellisia mittayksiköitä (em, rem) ja media query -tekniikoita. Näin kehittäjä voi automaattisesti sovittaa elementit eri näyttökokoihin ja resoluutioihin. [29; 30.]

Sisällön priorisointi on tärkeä osa responsiivista suunnittelua, sillä mobiilikäyttäjille on näytettävä vain olennaisimmat tiedot. Vähemmän tärkeät elementit voidaan siirtää erilliseen näkymään tai jopa piilottaa kokonaan tarpeen mukaan. [29; 30.]

Navigoinnissa suositellaan käytettäväksi mobiilille suunniteltua navigointia, kuten liukuvalikkoa (hamburger-valikko), joka avautuu tarvittaessa. Näin navigointi ei vie juurikaan tilaa näytöltä, mikä parantaa käyttäjäkokemusta erityisesti pienillä näytöillä. Lisäksi on tärkeää käyttää selkeitä kuvakkeita ja painikkeita, jotka sopivat kosketusnäyttöjen käyttöön. [29; 30.]

Responsiivisen suunnittelun haasteisiin vastaaminen edellyttää huolellista suunnittelua, testausta ja optimointia. On tärkeää muistaa, että mobiililaitteiden ja näyttöjen kirjo jatkaa kasvamistaan, joten responsiivisen suunnittelun periaatteiden ja ratkaisuiden päivittäminen ja kehittäminen on jatkuvaa työtä. [29; 30.]

Lisäksi responsiivisen suunnittelun haasteet voivat liittyä esimerkiksi näppäimistön käyttöön eri mobiililaitteilla, erilaisten käyttöympäristöjen huomioimiseen (esimerkiksi ulkona kirkkaassa auringonvalossa), erityisryhmien tarpeiden

huomioimiseen (kuten esteettömän suunnittelun periaatteet) sekä monikielisyyteen ja kulttuuriseen moninaisuuteen liittyviin haasteisiin. [29; 30.]

Ratkaisut näihin haasteisiin voivat sisältää esimerkiksi käyttäjätestauksen ja käyttäjäpalautteen hyödyntämistä suunnitteluprosessissa, dynaamisen sisällön toimittamiseen käyttäjän sijainnin tai käyttäjän aiemman toiminnan perusteella sekä adaptiivisen suunnittelun käytön, jossa sisältö ja käyttöliittymä muokkautuvat käyttäjän tarpeiden mukaan. [29; 30]

Kokonaisuudessaan responsiivisen suunnittelun merkitys korostuu entisestään digitaalisten palveluiden kehityksessä, ja suunnittelun huolellinen toteuttaminen voi merkittävästi parantaa käyttäjäkokemusta erilaisilla laitteilla ja ympäristöissä. [29; 30.]

7 Käytettävyydestaus ja palautteen hyödyntäminen

Käytettävyydestaus on keskeinen vaihe responsiivisen suunnittelun toteutuksessa, jonka tarkoituksena on arvioida suunnitellun sovelluksen tai verkkosivuston käytettävyyttä sekä tunnistaa mahdollisia ongelmakohtia, jotka voivat vaikuttaa käyttäjäkokemukseen eri laitteilla ja näyttöympäristössä. Tämä luku käsittelee käytettävyydestauksen merkitystä ja menetelmiä sekä palautteen hyödyntämistä responsiivisen suunnittelun jatkuvassa kehityksessä. [31.]

7.1 Käytettävyydestauksen merkitys

Käytettävyydestauksen avulla varmistetaan, että responsiivisesti suunniteltu sovellus tai verkkosivusto toimii saumattomasti erilaisilla laitteilla ja tarjoaa käyttäjille optimaalisen käyttökokemuksen. Testauksen avulla voidaan havaita mahdolliset ongelmakohdat, kuten toimimattomat navigointielementit, hankalasti löydettävät toiminnot tai epäselvä sisältö, ja tehdä tarvittavia korjauksia ennen julkaisua. [31.]

Käytettävyydestä antaa myös arvokasta tietoa siitä, miten käyttäjät vuorovai-
kuttavat sovelluksen tai verkkosivuston kanssa eri laitteilla ja millaisia haasteita
he kohtaavat. Näiden havaintojen perusteella voidaan tehdä suunnitteluratkai-
suja, jotka parantavat käyttäjäkokemusta ja lisäävät käyttäjien tyytyväisyyttä.

[31.]

7.2 Käytettävyydestä testauksen menetelmät

Käytettävyydestä testauksen menetelmät voivat vaihdella riippuen testattavasta so-
velluksesta tai verkkosivustosta sekä käytettävissä olevista resursseista. Ylei-
simpiä menetelmiä ovat käyttäjätestaus, A/B-testaus, asiantuntijan arviointi ja
käyttäjäkyselyt. Käyttäjätestauksessa käyttäjille annetaan tehtäviä, joiden avulla
heidän odotetaan käyttävän sovellusta tai verkkosivustoa erilaisilla laitteilla.
Käyttäjien toimintaa ja reaktioita seurataan, ja heiltä kerätään palautetta käyttö-
kokemuksesta. A/B-testauksessa sovelluksen tai verkkosivuston eri versioita
testataan samanaikaisesti ja niiden suorituskykyä verrataan toisiinsa, mikä
mahdollistaa erilaisten suunnitteluratkaisujen arvioinnin ja parhaiden käytäntö-
jen tunnistamisen. Asiantuntijan arvioinnissa kokeneet käytettävyyssiantuntijat
tarkastelevat suunnittelua ja antavat palautetta sen käytettävyydestä ja käyttäjä-
kokemuksesta. Käyttäjäkyselyissä puolestaan käyttäjiltä pyydetään palautetta
sovelluksen tai verkkosivuston käytettävyydestä sekä mahdollisista parannus-
ehdotuksista.

Käytettävyydestä testauksen keräämää palautetta voidaan hyödyntää responsiivisen
suunnittelun jatkuvassa kehityksessä parantamalla käyttäjäkokemusta ja opti-
moimalla sovelluksen tai verkkosivuston suorituskykyä eri laitteilla. Tätä varten
on tärkeää seurata ja analysoida testauksen tuloksia sekä tehdä tarvittavia
muutoksia suunnitteluun ja toteutukseen. [31.]

Palautteen perusteella voidaan tunnistaa toistuvia ongelmia ja haasteita, jotka
vaikuttavat käyttäjäkokemukseen eri laitteilla. Näiden ongelmien korjaamiseksi
voidaan tehdä muutoksia käyttöliittymän rakenteeseen, navigointielementteihin,
sisällön esitystapaan tai visuaaliseen suunnitteluun. [31.]

Lisäksi palautteen avulla voidaan tunnistaa myös positiivisia käyttäjäkokemuksen elementtejä ja vahvuuksia, joita voidaan korostaa ja hyödyntää suunnittelussa entistä paremman käyttäjäkokemuksen saavuttamiseksi.

Kokonaisuudessaan käytettävyydestä ja palautteen hyödyntäminen ovat olennainen osa responsiivisen suunnittelun prosessia, mikä mahdollistaa käyttäjäkeskeisen suunnittelun ja varmistaa, että sovellus tai verkkosivusto tarjoaa optimaalisen käyttökokemuksen eri laitteilla ja käyttöympäristössä. [31.]

8 Tulevaisuuden näkymät

Teknologisten innovaatioiden jatkuva kehittyminen vaikuttaa suoraan responsiivisen suunnittelun tulevaisuuteen. Erityisesti laitteiden ja näyttöjen monimuotoisuuden lisääntyminen asettaa uusia vaatimuksia ja tarjoaa samalla mahdollisuuksia responsiiviselle suunnittelulle. Perinteisten mobiililaitteiden, kuten älypuhelimien ja tablettien rinnalle on noussut uusia laitekategorioita, kuten älykellot, älylasit, älykaiuttimet ja älykodinlaitteet. Näiden erilaisten laitteiden käyttöympäristöt ja tilanteet vaihtelevat merkittävästi, mikä luo suunnittelijoille haasteen luoda käyttökokemuksia, jotka ovat saumattomia ja intuitiivisia riippumatta käytetystä laitteesta tai alustasta. [32.]

Teknologisten läpimurtojen, kuten 5G-verkkojen yleistyminen vaikuttaa responsiiviseen suunnitteluun. Nopeammat yhteydet mahdollistavat raskaampien sisältöjen, kuten korkearesoluutioisten videoiden ja virtuaalitodellisuussovellusten käytön mobiililaitteilla. Tämä asettaa suunnittelijoille paineita varmistaa, että responsiiviset verkkosivut ja -sovellukset toimivat moitteettomasti myös näissä tilanteissa, ilman merkittäviä viiveitä tai suorituskykyongelmia. [32.]

Älykaiuttimien ja älyavustajien, kuten Amazonin Alexan ja Google Assistantin yleistyminen muuttaa tapaa, jolla käyttäjät vuorovaikuttavat digitaalisten palveluiden kanssa. Tämä vaikuttaa suoraan responsiiviseen suunnitteluun, sillä käyttöliittymien on tuettava ääniohjausta ja puheentunnistusta. Näin ollen

suunnittelijoiden on luotava uudenlaisia käyttöliittymiä ja interaktiivisia elementtejä, jotka vastaavat näihin muuttuviin käyttäjätarpeisiin. [32.]

Data-analytiikka on keskeisessä roolissa responsiivisen suunnittelun jatkuvassa kehittämisessä ja optimoinnissa. Käyttäjäkokemustiedot tarjoavat arvokasta tietoa suunnittelijoille, mikä auttaa ymmärtämään paremmin käyttäjien tarpeita ja käyttäytymistä eri laitteilla ja alustoilla. Tämän tiedon avulla suunnittelijat voivat tehdä perusteltuja päätöksiä suunnitteluratkaisuista ja priorisoida kehitystoimenpiteitä responsiivisen käyttökokemuksen parantamiseksi.

Kokonaisuudessaan responsiivisen suunnittelun tulevaisuus näyttää lupaavalta, mutta samalla se asettaa suunnittelijoille ja kehittäjille uusia haasteita, jotka vaativat jatkuvaa innovointia ja sopeutumista muuttuviin teknologisiin ja -käyttäjätarpeisiin. Sopeutumalla muutoksiin ja kehittämällä uusia toteutustapoja voidaan varmistaa, että digitaaliset palvelut tarjoavat saumattoman ja tehokkaan käyttökokemuksen kaikilla alustoilla ja laitteilla. [32.]

9 Yhteenveto

Projektin lopputuloksena syntyi React-pohjainen sovellus, joka vastasi täsmällisesti asiakkaan tarpeisiin ja tehosti heidän päivittäisiä prosessejaan. Suunnittelussa painotettiin suorituskykyä ja tehokkuutta, sillä sovelluksen käyttäjäkunta oli rajattu, ja palveluun pääsi ainoastaan kutsulinkin kautta.

Käyttöliittymä toteutettiin minimalistisella väripaletilla, jossa pääväreinä olivat tummansininen, harmaa, valkoinen ja musta. Korostusväreinä käytettiin punaista ja vihreää tärkeissä toiminnoissa, kuten tietojen poistamisessa ja lataamisessa, mikä paransi toimintojen selkeyttä ja vähensi virheiden mahdollisuutta. Lisäksi painikkeiden sijoittelua ja hierarkiaa selkeytettiin erityisesti mobiilikäyttäjiä ajatellen.

Responsiivinen suunnittelu oli keskeinen osa kehitysprosessia, jotta sovellus toimisi saumattomasti eri laitteilla. Media queryjen ja joustavien rakenteiden

avulla varmistettiin, että sovelluksen käyttöliittymä mukautui eri näyttöko'oilte sujuvasti. Mobiiliversioon lisättiin tilaa säästävää hamburger-valikko, joka teki navigoinnista intuitiivista ja vapautti tilaa sisällölle.

Suorituskykyä optimoitiin karsimalla turhia SQL-kyselyitä ja parantamalla koodin tehokkuutta, minkä ansiosta latausajat lyhenivät ja tietokantakuorma keveni. Asiakkaan toiveesta sovellukseen lisättiin myös mahdollisuus ladata tietoja Excel-muodossa, mikä sujuvoitti heidän työnkulkuaan.

Responsiivinen suunnittelu ja käyttäjäkokemuksen optimointi ovat keskeisiä tekijöitä nykyaikaisten sovellusten kehityksessä. Käyttäjäkokemus rakentuu muun muassa käytettävyydestä, esteettisyydestä, suorituskyvystä ja saavutettavuudesta. Näiden tekijöiden huolellinen yhdistäminen luo toimivia ja miellyttäviä digitaalisia ratkaisuja.

Projektin onnistuminen perustui selkeään suunnitteluun, optimointiin ja iteratiiviseen kehitysprosessiin. Asiakkaan tarpeiden ymmärtäminen ja niihin vastaaminen mahdollistivat sovelluksen, joka ei ainoastaan täyttänyt odotuksia vaan myös paransi käyttäjien arkea merkittävästi. Teknologian ja suunnitteluperiaatteiden jatkuva kehitys edellyttää sovelluskehittäjiltä joustavuutta ja kykyä mukautua uusiin vaatimuksiin, jotta käyttökokemus säilyy saumattomana kaikilla laitteilla ja alustoilla.

Lähteet

1. How important is responsive design? 2021. Verkkoaineisto. Devrix. <<https://devrix.com/tutorial/important-responsive-design/>>. 2021. Luettu 3.9.2023.
2. Friedman, Vitaly. 2018. Verkkoaineisto. Responsive Web Design: What it is and How to Use it. Verkkoaineisto. <<https://smashingmagazine.com/2011/01/guidelines-for-responsive-web-design/>>. 2018. Luettu 6.9.2023.
3. Galante, Andrés. 2020. Verkkoaineisto. A complete guide to CSS media Queries. <css-tricks.com/a-complete-guide-to-css-media-queries/>. 2020. Luettu 6.9.2023.
4. What is CSS. Verkkoaineisto. <https://developer.mozilla.org/en-US/docs/Learn/CSS/First_steps/What_is_CSS>. Luettu 11.9.2023.
5. Responsive web design – media queries. <https://www.w3schools.com/css/css_rwd_mediaqueries.asp>. Luettu 12.2.2024.
6. Bose, Shereya. Top 5 CSS frameworks for developers and designers. 2023. Verkkoaineisto. <<https://www.browserstack.com/guide/top-css-frameworks>>. 2023. Luettu 1.3.2024
7. Bootstrap. 2025. Verkkoaineisto. <<https://getbootstrap.com/>>. 2025. Luettu 11.3.2025
8. Foundation. 2025. Verkkoaineisto. <<https://get.foundation/>>. 2025. Luettu 11.3.2025
9. Bulma. 2025. Verkkoaineisto. <<https://bulma.io/>>. 2025. Luettu 11.3.2025.

10. Tailwind. 2025. Verkkosivu. <<https://tailwindcss.com/>>. 2025. Luettu 11.3.2025.
11. Materialize CSS. 2025. Verkkosivu. <<https://materializecss.com/>>. 2025. Luettu 11.3.2025
12. Khuram, Falak. 2018. Verkkoaineisto. Top 7 tools to use for responsive web design. <<https://www.iqvis.com/blog/7-tools-for-responsive-design/>>. Luettu 11.9.2023
13. User Experience. Verkkoaineisto. ProductPlan. <<https://www.productplan.com/glossary/user-experience/>>. Luettu 6.9.2023.
14. Käyttäjäkokemuksen merkitys verkkokaupassa. Verkkoaineisto. 2023. <<https://www.domainkeskus.com/kayttajakokemuksen-merkitys-verkko-kaupassa/>>. 24.11.2023. Luettu. 1.3.2024.
15. Importance of user experience. Rocket55. Verkkoaineisto. <<https://www.rocket55.com/lab-note/the-importance-of-user-experience/>>. Luettu 1.3.2024.
16. Spangler, Todd. Amazon is giving prime video's user interface a much-needed redesign. 2022. Verkkoaineisto. <<https://variety.com/2022/digital/news/amazon-prime-video-redesign-user-interface-1235317952/>>. 18.7.2022. Luettu 15.2.2024.
17. Usability. Interaction Design Foundation. Verkkoaineisto. <<https://www.interaction-design.org/literature/topics/usability>>. Luettu 1.3.2024.
18. Martinez, Angela. 2023. Verkkoaineisto. Usability of web applications: what it is and recommendations. <<https://www.occamagenciadigital.com/en/blog/usability-of-web-applications-what-is-it-and-recommendations>>. Luettu 1.3.2024.

19. Kader, Nurhayati. 2022. Verkkoaineisto. The role of usability in web design. <<https://www.linkedin.com/pulse/role-usability-web-design-nurhayati-kader/?trk=pulse-article>>. 26.12.2022. Luettu 1.3.2024
20. Amazon. 2025, Verkkosivu. <<https://www.amazon.co.uk/>>. Luettu 11.3.2025
21. Vadhada, Keyur. 2023. Verkkoaineisto. The impact of typography on user experience. <<https://bootcamp.uxdesign.cc/the-impact-of-typography-on-user-experience-e57c702395f3>>. 3.8.2023. Luettu. 2.3.2024.
22. Joseph, Timonthy. 2023. Verkkoaineisto. <<https://blog.qa-source.com/es/aesthetics-of-mobile-apps-how-to-improve-user-experience>>. 3.5.2023. Luettu. 2.3.2024.
23. Interaction design foundation. Verkkoaineisto. <<https://www.interaction-design.org/literature/topics/visual-hierarchy>>. Luettu. 2.3.2024
24. How to improve mobile app performance. Verkkoaineisto. <<https://uxcam.com/blog/how-to-improve-mobile-app-performance/>>. Luettu 4.3.2024.
25. How to make mobile app more readable. Verkkoaineisto. <<https://www.linkedin.com/advice/0/how-can-you-make-your-mobile-app-more-readable>>. Luettu 4.3.2024.
26. Mordaunt, Lily. 2022. How to make app more accessible. Verkkoaineisto. <<https://blog.usablenet.com/5-points-to-remember-to-make-your-app-more-accessible>>. 1.4.2022. Luettu 4.3.2024.
27. Memorisely. The psychology of colour in UX/UI. Verkkoaineisto. <<https://www.memorisely.com/blog/the-psychology-of-colour-in-ux-ui>>

design-understanding-how-colour-affects-user-experience>. Luettu 2.3.2024.

28. Aftab, Zainab. 2024 Role of visual design in user experience. Verkkoaineisto. <<https://hapy.design/journal/role-of-visual-design-in-user-experience/>>. 4.3.2024. Luettu. 2.4.2024.
29. Nimritee. 2020. Verkkoaineisto. 10 responsive web design challenges and their solution. <<https://www.labdatest.com/blog/10-responsive-web-design-challenges-and-their-solution/>>. 2020. Luettu. 9.9.2023.
30. Strelchenko, Kirill. 2014. Verkkoaineisto. 10 responsive design problems and fixes. <<https://ux-mag.com/articles/10-responsive-design-problems-and-fixes>>. 14.7.2011. Luettu. 10.9.2023.
31. Mustain, Sk. A guide to usability testing. 2023. <<https://medium.com/@skmustain/a-guide-to-usability-testing-improving-designs-through-user-feedback-e3e29743bd3>>. 1.10.2023. Luettu. 12.3.2024
32. UI Designz. Verkkoaineisto. 2023. <<https://www.linkedin.com/pulse/responsive-design-future-web-development-ui-designz/>>. 21.2.2023. Luettu 11.9.2023.