

Tallinna Ülikool
Matemaatika-loodusteaduskond
Informaatika osakond

Helena Kurvits

**Veebivormide disaini põhimõtted ja
evalvatsioonimeetodid**

Magistritöö

Juhendaja: Hans Põldoja

Autor:..... "....."..... 2007

Juhendaja:..... "....."..... 2007

Osakonna juhataja: "....."..... 2007

Sisukord

SISSEJUHATUS	4
1. VEEBIVORMIDE KOOSTAMISE PÕHIMÕTTED	7
1.1 VEEBIVORMIDE KOOSTAMISE EESMÄRGID, PÕHIMÕTTED JA PRINTSIIBID	9
1.2 VEEBIVORMIDE KOOSTAMISE KRITÈRIUMID	14
2.VEEBIVORMIDE TEHNILISED LAHENDUSED.....	18
2.1 KASUTAJA INFORMEERIMINE	19
2.2 VÄLJADE PEALKIRJADE KASUTAMISE PÕHIMÕTTED JA VÕIMALUSED.....	20
2.3 TEKSTI VÄLJADE JA TEKSTIALADE KASUTAMISE PÕHIMÕTTED JA VÕIMALUSED.....	27
2.4 EELTÄIDETUD VÄLJADE KASUTAMISE PÕHIMÕTTED JA VÕIMALUSED	29
2.5 RIPPLOENDI JA RIPPLOENDIKASTI KASUTAMISE PÕHIMÕTTED JA VÕIMALUSED	32
2.6 RAADIONUPPUDE JA MÄRKERUUTUDE KASUTAMISE PÕHIMÕTTED JA VÕIMALUSED	34
2.7 NUPPUDE KASUTAMISE PÕHIMÕTTED JA VÕIMALUSED	37
2.8 VEEBIVORMIDE PERSPEKTIIV JA VÕIMALUSED LÄHITULEVIKUS	39
3. VEEBIVORMIDE KUJUNDAMISE PÕHIMÕTTED.....	42
3.1 VEEBIVORMIDE STRUKTUUR JA ESITAMINE MITMEL VEEBILEHEL	43
3.2 PAIGUTUS	45
3.3 GRUPEERIMINE	50
3.4 TEKST, TAUST JA VÄRVID	50
3.5 WCAG METOODIKATE DISAINIJUHISED.....	51
4. VEEBIVORMIDE EVALVATSIOONI MEETODID	55
4.1 EVALVEERIMISE MEETODID	57
4.2 VEEBIVORMIDE EVALVATSIOON THINK ALOUD MEETODIGA.....	60
4.3 EVALVATSIOONIMEETODITE KOMPLEKT	62
5. LEMILL ÕPIOBJEKTIDE LOOMISE KESKKONNA VEEBIVORMIDE EVALVATSIOON 65	
5.1 AUTOMAATSETE VAHENDITEGA TESTIMISE TULEMUSED	66
5.2 THINK ALOUD MEETODIGA KASUTAJATESTIMISE TULEMUSED	70
5.3 JÄRELDUS	75
KOKKUVÕTE	77
KASUTATUD KIRJANDUS.....	78

LISA 1. THINK ALOUD MEETODI KASUTAJATESTIMISE JUHEND	81
LISA 2. VEEBIVORMIDE EVALVEERIMISE ÜLESANDED	82
SUMMARY.....	84

Sissejuhatus

Seoses veebikeskkondade muutumisega üha enam funktsionaalsemaks, pakkudes mitmeid erinevaid interaktsiooni või muid tagasiside ning suhtlemise võimalusi ekraanivormide kaudu, on neis esinevad probleemid või tõrked olulised informatsiooni eduka ja kiire liikumise mõjutajad. Sõltuvalt veebivormi tüübist on kogutava info kvaliteet erineva tundlikkuse astmega. Komplitseeritumate ekraanivormide puhul, milles soovitakse kasutajalt suuremahulist või konfidentsiaalset infot erinevate teenuste või hüvede osutamiseks, on ekraanivormide optimaalne toimimine väga oluline. Veebivormide kasutajakeskse disaini vajalikkust näitab asjaolu, et üha enam toimub erinevate toimingute viimine paberikandjalt internetipõhiseks (netipoed, pangateenused, erinevate teenuste broneerimine, e-õppe keskkonnad jne), kuna võimaldab pakkuda teenuse saajale paremat paindlikkust ning ei ole sõltuv asukohast.

Paljud ekraanivormid on üles ehitatud selliselt, et nendes orienteerumine nõuab tõsist pingutust ja palju aega, rääkimata ebaselgest struktuurist, mis võib mõjutada infost arusaamist ja seeläbi ka kogutava informatsiooni väärtust. Paljudel keerukamatel veebivormidel esineb kasutatavuse probleeme ja seetõttu peaks tähtsustama kasutajakesksema lähenemise olulisust. Viimastel aastatel on veebivormide kasutamisel esinevatele tõrgetele ja probleemidele korduvalt tähelepanu juhitud. Erik Duval püstitab isegi üleskutse "*Electronic forms must die*" (Duval, 2004).

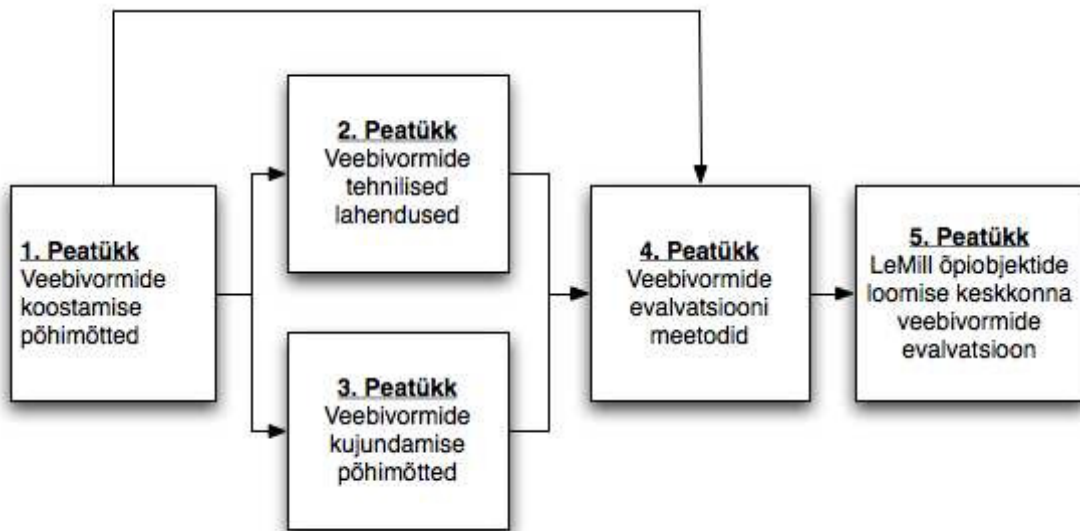
Kuna ekraanivormid on veebis väga oluline interaktsioonivahend, mille osatähtsuse suurenemist tingib igapäevatoimingute muutmine internetipõhiseks, tuleks senisest palju enam tähelepanu pöörata ekraanivormide kasutatavusele. Probleem, millele käesolevas töös lahendust otsitakse, on veebivormide kasutatavusega seotud raskused, millest tuleneb ka töö eesmärk.

Magistritöö eesmärgiks on pakkuda välja evalvatsioonimeetodite komplekt veebivormide kasutatavuse hindamiseks ja kasutatavusprobleemide avastamiseks.

Eesmärgi saavutamiseks püstitab autor järgmised ülesanded:

- Uurida ja analüüsida veebivormide koostamise põhimõtteid
- Uurida ja analüüsida veebivormide elementide koostamise võimalusi
- Uurida ja analüüsida veebivormide kujunduse põhimõtteid
- Koostada evalvatsioonimeetodite komplekt
- Teostada veebivormide analüüs valitud keskkonnas

Töö struktuur on esitatud joonisel 1, kus on näidatud peatükkide ülesehitus ja omavahelised seosed.



Joonis 1. Magistritöö struktuur

Magistritöö esimeses peatükis analüüsitakse veebivormide disainiga seotud teemasid nagu kasutajakeskse disaini põhimõtted, W3C spetsifikatsioonid ja WCAG meetodikad. Magistritöö eesmärgiks ei ole välja pakkuda reegleid, kuidas peab veebivormide kujundust teostama. Kõik juhised ja põhimõtted on soovituslikud ning osutavad erinevatele kasutatavusega seotud probleemidele.

Teises peatükis uuritakse veebivormide ja nende elementide kasutamise põhimõtteid ja võimalusi, sealhulgas tuues välja veebivormide perspektiivi ja arengusuunad.

Kolmandas peatükis käsitletakse veebivormide kasutajakeskse kujunduse võimalusi ja WCAG meetodikate disainijuhised.

Eeltoodud teooriate, soovitude ja meetodikate raamistikule tuginedes koostatakse neljandas peatükis evalvatsioonimeetodite komplekt ekraanivormide optimaalsema ja efektiivsema esitamise testimiseks, kus on rõhk neis esinevatel elementidel, veebivormi kujunduse põhimõtetel ja valideerimise läbiviimise analüüsil.

Evalvatsioonimeetodite komplektile tuginedes teostatakse magistritöö viiendas peatükis õpiobjektide loomiseks mõeldud ekraanivormide evalvatsioon õpiobjektide loomise keskkonnas LeMill (vt <http://lemill.net>), kus on tegemist piisavalt keeruliste vormide ja väga erineva arvutikasutusoskusega sihtgrupiga.

Veebivormide evalvatsioon hõlmab automaatsete vahenditega testimist, kontroll-loendiga testimist ja kasutajatega testimist. Veebivormides orienteerumist ja veebivormide käideldavusprobleeme uurides on võimalik teha erinevaid soovituslikke korrekture.

Magistritöö tulemusena valmib ekraanivormide analüüsiks evalvatsioonimeetodite komplekt, mille katsetamise etapis testitakse keskkonna LeMill ekraanivorme.

1. Veebivormide koostamise põhimõtted

Veebivormide disaini vajalikkus lähtub eesmärgist läbi interaktiivsuse koguda andmeid ning selle kaudu suhelda kasutajaga asünkroonse infovahetuse kaudu. Vormide koostamine erineb staatilise info edastamisest, kuna vormide puhul on tegemist piiratud tingimustele vastava süsteemiga nii kasutaja kui vormide koostaja seisukohast lähtudes. Vormide puhul on oluline kogutava info vastavus eesmärgile, mis tingib küsitavatele andmetele teatavad varieeruda võivad kriteeriumid nagu kompaktsus, koguseline piisavus, detailsus, ajalimiit, loogilisus, mitmeti-tõlgendatavuse välistamine ja soorituse kordade arv.

Kui staatilist infot töödeldes on võimalik seda korduvalt üle lugeda, teha täpsustusi ja ümber mõtestada siis vormide puhul võivad takistuseks saada piiravad tingimused ning nendest tulenevad kriteeriumid. Seetõttu on väga oluline veebivormide koostamisel lähtuda erinevatest põhimõtetest ja printsiipidest, leides konkreetseid vajadusi ja eesmärke silmas pidades sobivad lahendused.

Kuna iga vormi koostamise eesmärk on koguda infot siis on oluline, et kasutaja oleks nõus seda edastama, mistõttu tuleb luua selleks soodsad tingimused. Vormidest arusaamise, nende täitmise ja nendes orienteerumise aeg tuleks viia võimalikult väikseks ning kogutava info piisavus, täpsus, kompaktsus, detailsus ja loogilisus nii visuaalselt kui ka koode arvestades arusaadavalt selgeks. Kui kasutajal kulub vormidest arusaamiseks nendes orienteerumiseks ja täitmiseks liiga palju aega, suureneb tegevuse poolelijätmise tõenäosus kuluva aja suhtes.

Seetõttu on oluline viia:

- Vormidest arusaamise aeg miinimumini,
 - Vormi üldisest eesmärgist arusaamine
 - Vormi erinevatest elementidest arusaamine
 - Vormis navigeerimisest arusaamine
- Vormide täitmisele kuluv aeg miinimumini,
 - Vormide sisu üheseltmõistetavus
 - Vormidega seotud abiinfo kättesaadavus
 - Vormide elementidega tõrgeteta toimetamine

- Vormides orienteerumise aeg miinimumini.
 - Vormide struktuuri visuaalne arusaadavus
 - Vormides tõrgeteta liikumine
 - Vormide tuttavlikkus ja loogilisus

Nii nagu vormide eesmärgi ebaselge esitamine võib ka elementide ebaloogiline esitus ja sellega kaasnev piiratud arusaamine vormi täitmisest põhjustada ebavajaliku või puuduliku info edastamist.

Vormide täitmisel peaks sisu olema üheselt mõistetav ja loogiline, võimaldades abimaterjalide kergelt ülesleidmist ning erinevate vormielementide kasutust optimaalselt kasutajakesksuse põhimõtetest lähtudes. Kuna vormide täitmisega kaasneb ka nendes orienteerumine, liikudes erinevatelt etappidelt edasi või tagasi paranduste ning muudatuste sisseviimiseks siis on oluline üldine veebivormide struktuuri visuaalne arusaadavus, mis võimaldaks neis tõrgeteta liikuda, tuginedes loogilisusele ja tuttavlikkusele.

Kasutajakesksuse taseme oluliseks mõjutajaks lisaks kujundusele on W3C HTML ja WCAG spetsifikatsioonidele vastavus, mis parandab sõltumata kasutatavast keskkonnast ühilduvust ja seega ka ligipääsetavust ja kättesaadavust.

Veebivormidest saadava kasu maksimeerimisel on oluline tähelepanu pöörata nii üldistele kasutoovatele faktoritele, mis tulenevad majanduslikest või hariduslikest eesmärkidest ning ka spetsiifilisematele faktoritele nagu info kvaliteet selle kogumise efektiivsus. Seetõttu tuleb vormide planeerimise etapis lahti mõtestada soovitatav eesmärk ja selleks vajaminevad tegevused ning nõudmised millega neid tegevusi saavutada.

Seetõttu veebivormide kasutajakeskne disain:

- lisab usaldatavust üldiselt,
- tõstab konkurentsieelist alternatiivsete lahenduste ees,
- tõstab kasutatavust,
- parandab kogutava info kvaliteeti vastavuses nõudmistele,
- tõhustab info liikumist,
- vähendab eksimuste arvu vormide täitmisel,
- vähendab vormide täitmisele kuluvat aega,
- võimaldab ekraanilugejaga ligipääsu.

Eelnevast lähtudes soosivad mitmed nii välisest kui sisemisest keskkonnast tulenevad mõjurid kasutajakesksete disainivõtete tundmist ja nende rakendamist määral, mis võimaldab veebivormidega ilma tõrgeteta töötada.

1.1 Veebivormide koostamise eesmärgid, põhimõtted ja printsiibid

Veebivormide koostamisel on võimalik aluseks võtta erinevaid põhimõtteid ja printsiipe, valides neist sobivad lahendused konkreetseid vajadusi ja eesmärke silmas pidades.

Veebivormide eesmärgiks on info kogumine interaktiivselt ja selle rakendamiseks kasutatakse **HCI** (*Human Computer Interaction*) printsiipe:

- Tuleb mõista faktoreid, mis määravad kindlaks, kuidas inimesed süsteemi kasutavad;
- Arendada sobivaid vahendeid ja põhimõtteid kasutades selleks sobilikke süsteeme;
- Saavutada tõhus, efektiivne ja turvaline interaktsioon;
- Seada inimesed oma vajaduste ja soovidega esikohale; (Preece et al.,1994)

HCI printsiipide kohaselt tuleks veebivormid mugandada kasutajate soovide ja vajadustega kooskõlas, mitte sundida kasutajat õppima selgeks kasutatava süsteemi eripärasid ja toimimismehhanisme.

Väga oluline tegur veebivormide uurimisel on kasutatavus (ingl k. *usability*) see on kasutajaliidese atribuut, mis näitab kasutamise ja õppimise lihtsust, turvalisust, efektiivsust ning kasutajate suhtumist süsteemi. (Preece et al., 1994, lk 722)

Kuna kasutatavus on osa HCI printsiibist, siis on veebivormide disainimisel oluline pöörata tähelepanu selle tagamise põhimõtetele:

- Õppimislihtsus,
- Meeldejätmise lihtsus,
- Kasutusefektiivsus,
- Kasutuslihtsus,
- Veakindlus,
- Kasutamise nauditavus. (Preece et al.,1994)

Kasutatavust mõjutavad oluliselt tagasiside, visuaalne disain, sisestatava info formaat ja struktuur ning funktsionaalsus, seega loodava veebivormi parema interaktiivsuse saavutamiseks tuleks vajaminevad põhimõtteid stabiilselt kasutada.

Visuaalse disaini selguse ja nähtavuse all peetakse töö kontekstis silmas elementide vaheliste seoste loogilisust, mis tuleneb nende disainist. Funktsionaalsuse all peetakse silmas elementide tajutavat eesmärki, andes kasutajale läbi disaini mõista, milliseid operatsioone on võimalik teostada.

Veebivormide disainis on oluline, et tegemist oleks selge, loogilise ülesehitusega ja kasutaja suudaks iseseisvalt ilma kõrvalise abita vormidega töötada. Kui kasutajal kulub palju aega mõtlemisele, et vormidega töötamisel eesmärgini jõuda, tuleb selgeks teha vormide nõrgad küljed ja seeläbi eemaldada tekkida võivad küsimused. Mida vähem kõhklusi ja küsimusi tekib, seda loogilisem on vormide struktuur ja seda kiiremini suudab kasutaja neid läbi töötada ning kokkuvõttes suurendab see ka tõenäosust, et kasutajale keskkond meeldib.

Kuna veebivormide eesmärgiks on info kogumine siis võib vormide efektiivsema koostamise vajadust paremini mõista läbi järgneva mudeli. **Kolmekihiline vormide koostamise mudel** kujutab endast parema kasutajakesksuse tagamiseks vajaminevat lähenemist:

- püsivat paigutust,
- suhtlust küsimuste ja vastuste näol,
- eesmärgi ja struktuuri vahelisi seoseid. (Jarrett, 2000)

Kolmekihilise vormide koostamise mudeli rakendamine lisab eesmärgini jõudmise selgust ja selleks vajaminevate tegevuste paremini mõistmist. Mudel jagab veebivormid: paigutuse kihiks, suhtluskihiks ja ülesande eesmärgi kihiks, keskendudes rohkem kasutajakesksuse visuaalsele kui vormi koodilisele eripärale. Mudel rõhutab, et ennekõike on vaja läbimõeldud eesmärgi struktuuri ja seejärel loogilist elementide paigutust ning suhtlust läbi küsimuste ja vastuste.

Samuti peaks vormide koostamisel tähelepanu pöörama ametlike **W3C** spetsifikatsioonidega kooskõlastatusele, mille eesmärgid ja põhimõtted hõlmavad:

- Universaalset kättesaadavust ja ligipääsu kõigile,

- Semantilist veebi,
- Usaldust,
- Interoperaablust,
- Arenemisvõimelisust
- Detsentralisatsiooni ja paindlikkust
- Suuremat interaktiivsust ja tõrgeteta multimeedia kasutust. (World Wide Web Consortium, 2005a)

Veebistandarditega kooskõlastatus hõlmab kasutajakesksuslikke nüansse nagu erivajadustega kasutajate ligipääsu tagamine, koodi eripärast tulenev kättesaadavus seoses ühilduvuse ja paindlikkusega, eesmärgiga tagada erinevatest keskkondadest juurdepääs.

Kuna vormide koostamisel on oluline arvestada ka erivajadustega kasutajate tõrgeteta juurdepääsu tagamisega, siis tuleks aluseks võtta ka W3C **WAI** (Web Accessibility Initiative) meetodikad, mis keskenduvad:

- Sisule veebileheküljel või rakendusel (HTML spetsifikatsioonid)
 - Visuaalsele poolele nagu tekst, pildid ja helid
 - Koodile või märgistuskeelele, mis määrab struktuuri ja esituse
- Veebilehitsetajatele ja meedia edastajatele
- Abistavale tehnoloogiale nagu ekraanilugejad ja klaviatuurid
- Kasutajate teadmiste, oskuste, kogemuste
- Arendajatele, disaineritele, kodeerijatele, autoritele, erivajadustega arendajatele
- Tarkvarale, millele luuakse veebilehekülgi ja sisu
- Valideerimise tööriistadele, HTML validaatoritele, CSS validaatoritele. (W3C Web Accessibility Initiative, 2006a)

Juurdepääs erivajadustega kasutajatele on tagatud nimetatud komponentide ja nende omavahelise seotuse tugevusega. Kui mõni lülidest osutub nõrgaks, võib see piirata juurdepääsu või kasutamise efektiivsust. Iga kasutajakeskuse lülile keskendub erinev spetsifikatsioon. Veebivormide koostamisega on enam seotud **WCAG meetodikad** (Web Content Accessibility Guidelines), mis keskenduvad veebi sisu kättesaadavaks tegemisele erivajadustega kasutajatele, hõlmates teksti, vorme, pilte, heli jne. Enne erivajadustega kasutajatele mõeldud standarditega arvestamist, tuleb mõista, mida mõeldakse erivajaduste all:

- inimesed, kes ei näe, ei kuule, ei saa liigutada info töötlemiseks vajamineval hulgal;
- inimesed, kellel on raskusi teksti lugemise või sellest arusaamisega;
- inimesed, kes ei saa kasutada klaviatuuri või hiirt;
- inimesed, kellel on tekstipõhised ekraanid, väiked ekraanid või aeglane interneti ühendus;
- inimesed, kes ei räägi ja ei saa aru keelest, milles sisu on kirjutatud;
- inimesed kes on olukorras, kus nende silmad, kõrvad või käed on kinni või hõivatud teiste tegevustega;
- inimesed kellel on vanem lehitseja versioon, teist tüüpi lehitseja, häällehitseja või teine operatsiooni süsteem; (World Wide Web Consortium, 1999)

Seega WCAG standardite järgimine mõjutab väga tugevalt kasutajakesksust üldiselt ning saadav kasu laieneb erivajadustega kasutajatelt ka tavakasutajatele. Siinkohal võib eristada erivajadustega kasutajaid, kelle erivajadused on ajutised ja kasutajaid, kellel need on püsivad (nt. ealistest muutustest tulenevad). Seetõttu oleks vajalik tagada veebivormides jätkusuutlik tegutsemine pidades silmas kasutajate erinevatel eluetappidel ette tulevaid olukordi.

Vormide koostamise planeerimisel peaks lähtuma mitmetest põhimõtetest, mille on oma artiklis välja toonud Tony Arguelles:

- **Planeeri, seejärel disaini** — Kõigepealt tuleks tuleks läbi mõelda milliste küsimuste esitamine viib kõige paremini eesmärgi saavutamiseni koostöös töörühma või klientidega.
- **Selgitustöö ja veenmine** — Selles etapis tuleks välja selgitada, millist infot on vaja enne teada, kui vorme täitma hakata. Millise detailsuse tasemeni, millises mahus ja mille kohta peaks olema info kasutajale edastatud.
- **Andmete kogumise efektiivsus** — Määratakse kindlaks vormi elementide kasutus, mis tagaks kõige paremini vormide täitmise kiiruse, lihtsuse ja loogilisuse. Määrates sealhulgas grupeerimise, vormide paigutuse, eeltäidetud väljad ja kontrollides andmeid kliendipoolse mitte serveripoolsena.
- **Funktsionaalsed vormid visuaalse toetusega** — Vormide väljanägemine on nende efektiivsuse tagamisel määrava tähtsusega. Selles etapis arutatakse läbi visuaalse poole optimaalseimad lahendused nagu grupeerimine, paigutus, tühikud

ja järjekord. Samuti uuritakse kuidas kasutada edu tagamiseks erinevaid indikaatoreid.

- **Andmete valideerimise efektiivsus** — Kuna kasutajad teevad paratamatult vormide täitmisel vigu siis tuleb mõelda tavapärasemate veateadete edastamise kord, mis lihtsustaks edukat parandamist kasutaja poolt, tehes seda viisakalt ja võimalikult delikaatselt.
- **Kui vormi täitmine on lõpetatud** — Kui kasutaja edastab täidetud vormid koos infoga, siis milliseid teateid ja infot oleks vajalik seejärel talle edastada. (Arguelles, 2002)

Tegevuste ja toimingute ette planeerimine vähendab oluliselt tekkida võivate tõrgete haldamist veebivormide koostamise ja kujundamise protsessis. Samuti võimaldab see kokku hoida hiljem paranduste sisseviimiseks kuluvat aega ning sellega kaasnevat ressursikulu. Veebivormide koostamise planeerimise vajaduse tingib nende eripära ja loodetav eesmärk. Planeerimist võiks rakendada ennekõike suuremahuliste ja pikemate veebivormide koostamisel.

Võttes arvesse, et veebi disainimisel ja kasutajaliidese loomisel on olulised erinevused, tuleb tunnistada, et vormide disainimisel on GUI (*graphical user interface*) standarditest lähtudes ka palju sarnasusi ning neid saab vaadelda toetava raamistikuna. Jakob Nielsen on oma artiklis välja toonud **kümme olulisemat kasutatavuse heuristikut**:

- Süsteemi staatuse nähtavus (tagasiside toimuvast mõistliku aja vältel)
- Sobivus süsteemi ja reaalse maailma vahel (sisu ja fraaside tuttavlikkus, info loogiline ilmumine ja kadumine)
- Kasutaja kontroll ja vabadus (edasi-tagasi nupud)
- Terviklikkus ja standardid
- Vigade vältimine (disain, mis väldib veateadete esinemise vajadust, kinnituse küsimine enne täidesaatmist)
- Äratundmine on parem kui meenutamine (vajaminev info peaks olema nähtav ja äratuntav, mitte vajada meenutamist)
- Kasutuspaindlikkus ja tõhusus (luba kasutajal kohandada sarnaseid vastuseid)
- Esteetiline ja minimalistlik disain (ainult vajalik info, vähendab müra)

- Aita kasutajaid parandada ja ära tunda veateateid (peaks mitte sisaldama koodi ja võiks pakkuda lahendust)
- Abiinfo ja dokumentatsioon (dokumentatsiooni vajaduse tingib süsteemi eripära, peaks olema võimalus abimaterjali otsida). (Nielsen, 1994)

Ülaltoodud heuristikutele tuginedes saab teostada ka heuristilist evalvatsiooni, millest on ülevaade peatükis 4. Eelnevast lähtudes on veebivormide koostamisel võimalik lähtuda paljudest põhimõtetest, spetsifikatsioonides ja printsiipidest, seetõttu valiku nende hulgast määrab eesmärgi ja oodatava tulemuse eripära ning loodava süsteemi sihtgrupp.

1.2 Veebivormide koostamise kriteeriumid

Kuna olemasolevate teooriate, põhimõtete ja printsiipide raamistik on suur, võivad need konkreetse süsteemi puhul osutada liialt laialivalguvaks. Veebivormide koostamisel tuleks vajadusi silmas pidades määratleda konkreetsete kriteeriumide grupid, millega arvestada.

Vormide struktuurist arusaamise ja nendes orienteerumise lihtsus. (arusaamine)

Vormide koostamisel tuleks silmas pidada keskmiste oskustega arvutikasutaja teadmisi ja sellest lähtudes peaks kasutaja olema suuteline aru saada vormides orienteerumise ja täitmise üldisest struktuurist. Vormide täitmisel on nendest õigesti arusaamise põhiliseks eelduseks info efektiivne edastamine vajaminevas koguses, detailsuses ja nõutud aja jooksul. Seetõttu peaks pöörama tähelepanu loogilisele ja tuttavlikule esituse järjekorrale, kasutades ära juba välja kujunenud teadmisi ja oskusi analoogsetes süsteemides orienteerumisel. Seejuures peaks järgima erinevaid tunnustatud spetsifikatsioone ja soovitusi. Ennekõike peaks süsteemi struktuur olema mugandatud kasutaja nõudmisi ja vajadusi silmas pidades, mitte sundides kasutajat enne süsteemi kasutamist seda selgeks õppima. Erinevate tuttavlike esituste kasutamine annab lisaväärtust, kuna loob võimaluse eelnevatele kogemustele ja teadmistele tuginedes luua seoseid ja tõmmata paralleele, mis hõlbustavad vormide kasutamist olulisel määral. Äärmiselt vajalik on erinevate elementide kaudu struktuuri stabiilne järgimine kogu süsteemi ulatuses, mis hõlmab sarnast ülesehitust läbi terve küsimustiku, muutes selles orienteerumise ja täitmise arusaamisele kuluva aja

miinimumini. Mida vähem peab kasutaja oma aega läbi süsteemi toimimise printsiipide lahtimõtestamise kulutama, seda suurem on tõenäosus, et vormide täitmisest ei loobuta ja kogutav info on parema kvaliteediga.

Vormide struktuuri kasutamise lihtsus mõjutab otseselt kasutajakesksust, seega tuleks planeerimisel võtta arvesse kasutajakesksuse tagamise printsiibid ulatuses, mis puudutavad antud gruppi.

Kasutaja piisav informeerimine abimaterjali, veateadete ja tagasisidega. (suhtlus)

Vormide koostamisel tuleb planeerida nende täitmiseks piisav informatsiooni edastamise süsteem, mis hõlmab teavitamist vormide eesmärgist, vajaminevast abimaterjalist, tagasisidet ja veateadete esinemise korda. Asjakohase tagasiside kättesaadavus, lihtsus ja loogilisus peaks olema tagatud stabiilselt kõikides täitmise etappides ja esinema pidevalt või vastavalt tekkivale vajadusele. Edastatav info peab olema kooskõlas tunnustatud nõuetega, esinema põhjendatud mahus ja ettenähtud ajahetkel ning võimaldama sobivate andmete otsimist sõltumata süsteemi etapist.

Võttes aluseks keskmiste oskustega arvutikasutaja oleks asjakohane edastada vajaminev info tuttavlikult arvestades eelnevate kogemuste ja teadmistega paralleelide loomise võimalusega. Kasutaja ja süsteemivahelise suhtluse hõlbustamiseks peaks olema kindlasti iga vormi puhul selgitatud andmete kogumise eesmärgid ja vajadused, sealhulgas ka delikaatsete andmete töötlemise kord ja vajadus, et lisada usaldust ja tagada kvaliteedile vastava info kogumist. Vajamineva info kogumist mõjutab abimenüüde kättesaadavus ja veateadete esitamise läbimõeldus, mille puudulikul planeerimisel võivad esineda tõrked vormide täitmise kiiruses, nende täitmise lõpetamises või nõuetekohase info edastamises.

Vormide koostamise suhtluse etapis tuleb aluseks võtta lisaks kasutajakesksuse nõuetele ka vormide koostamise planeerimise põhimõtted, kuna veateadete esinemise sagedus, struktuur ja edastatav abiinfo peavad olema eelnevalt iga etapi puhul erinevate kasutajakäitumuslike otsuste kohta ette planeeritud, vähendades võimalikke süsteemi kasutamisel ettetulevaid takistusi ning hõlbustamaks ilma kõrvalise abita lahenduste leidmist. Seetõttu peaks nii abiinfo kui veateated lisaks tekkinud tõrgetele viitamisele pakkuma välja ka konkreetse ja konstruktiivse lahenduse.

Vormi elementide kasutamise eesmärgipärasus ning vastavus W3C ja WCAG spetsifikatsioonidele. (kasutamine)

Vormide koostamisel on võimalik valida paljude erinevate elementide ja nende parameetrite ning väärtuste hulgast sobivad, lähtudes püstitatud eesmärgist süsteemi sihtgrupi vajadustele ja nõuetele. Kooskõlastatus W3C spetsifikatsioonide ja erivajadustega kasutajate metoodikatega on selles etapis möödapääsmatult vajalik. Eelnev planeerimine vormide koostamise põhimõtete kohaselt annab lisaväärtust, kuid ei ole selle struktuuri etapiviisilise järgimise seiskohast primaarne. Oluline on teadvustada erinevatest võimalustest saadavat kasu ning kooslusest lähtudes sooritada sobiv valik.

Vormi elementide kasutamisel peaks jälgima nende optimaalsust valitud andmete edastamiseks või info kogumiseks, pidades silmas info struktuurilist ja teksti edastamise või kogumise eesmärkidest tulenevat eripära. Tuleb selgelt eristada tingimusi, millal oleks sobilik ja vajalik kasutada ripploendeid, ripploendikaste, raadionuppe ja märkeruute, sest nende ebastandardne kasutus muudab edastatava ja kogutava info arhitektuuri nõrgaks ja seetõttu kvaliteedi puudulikuks.

Valida tuleks optimaalseim elemendi tüüp antud info edastamise jaoks kooskõlas kasutaja tutvavlikkusega elemendi tajumisel. Ebastandardsete lahenduste esitamine võib kasutajates põhjustada lisaks muudele vormi täitmist takistavatele asjaoludele ka suutmatust luua paralleele eelnevate kogemuste ja teadmistega ning seetõttu takistada info kogumist.

Vormide kasutamise etapis on olulise tähtsusega kooskõlastatus W3C ja WCAG spetsifikatsioonidega, kuna tagab ligipääsu erinevate vajadustega ja erinevaid kasutajapoolseid süsteeme arvestades. Sihtgrupi ligipääsu peaks olema võimalik tagada sõltumata kasutatavast lehitsejast ja erivajadustest, mille võimalikud esinemise variandid on välja toodud eelnevas peatükis, erivajaduste selgitava loeteluna.

Vormide esteetiline ja minimalistlik disain kooskõlas kasutajakesksuse ja WCAG põhimõtetega. (visuaalne kasutatavus)

Veebivormide esteetiline ja minimalistlik disain peab sisaldama endas nii kasutajakesksuse kui ka WCAG nõudeid, mistõttu vajab põhjalikku läbimõeldust nende omavahelise mõju ja seotuse aspektist. Kujunduse etapis peaks keskenduma elementide visuaalsele esitamisele ja kasutamiskihtsusele, mis toetaks ühest küljest vormide koostamise eesmarke ja teisest küljest elementide kasutamise eesmärgipärasust, moodustades kasutajakeskse tulemuse.

Kujunduse tüübi tingib eesmärgi eripära, mis vormide puhul peaks olema ennekõike konservatiivne ja aimatav, mitte uudne ja üllatav. Kasulik oleks kujundusega anda mõista suuremahuliste vormide puhul nende omavahelisest seotusest ja rakendada disainivõtteid nii info kui vormi elementide edastamist toetava tegurina. Kasutajakesksuse nõudeid silmas pidades disain peaks võimaldama vormidega töötada erivajadustega kasutajatel ning lubama kasutada selleks eriliigilisi lehitsejaid, väljaprinti jne.

Erinevate vormide disaini mõjutavate faktoritena võib välja tuua grupeerimise, järjekorra, tausta ja värvid, horisontaalse ja vertikaalse disaini ning tekstiga seonduvad parameetrid, mille kõigi ebastandardne või ebasobiv kasutus võivad põhjustada tõsiseid tõrkeid. Vormide kujunduse esteetilise ja minimalistliku disaini tagamisel paraneb oluliselt nende kasutamiskihtsus ja nendes orienteerumine, mistõttu on võimalik kujundusega vormide täitmiseks kuluvat aega oluliselt minimeerida, sealhulgas tagada loogiline info esitamine ja edastamine arusaadavalt, lihtsalt ja kvaliteetselt.

2.Veebivormide tehnilised lahendused

Kuna veebivormide disainiks kasutatavate XHTML elementide ja nende atribuutide valik on sõltuvuses soovitatavast eesmärgist, siis ei saa kõiki reegleid võtta üheseltmõistetavana ning tuleb mugandada neid sõltuvalt kontekstist. Seega tuleb eelnevalt selgeks teha ja planeerida konkreetse keskkonna jaoks sobivad lahendused. Veebivormide elementide valimisel tuleb silmas pidada, kuidas kasutaja mõtleb ja reageerib, kuna tegemist on interaktiivsuse sellise etapiga, kus on äärmiselt oluline kasutajakesksus vajadustele vastava info kogumiseks.

Veebivormid võivad koosneda erinevatest elementidest, milleks võivad olla sildid (*labels*) ja nende kontrollimiselemendid (*controls*): lahtrid (*input*), tekstiväljad (*textarea*), ripploendid (*select menu*), ripploendikastid (*scolling lists*), raadionupud (*radiobutton*), märkeruudud (*checkbox*), nupud (*button*). Seoses elementide valiku langetamise ebaselgete reeglite ja elementide omaduste näilise sarnasusega, võib osutada keeruliseks andmete tüübile sobiva kontrollija valimine. Seetõttu tuleks kontrollimiselementide hulgast valiku langetamisel lähtuda nende kasutamise põhimõtetest ja vaadata millise kasutamine neist toob kaasa kõige väiksema vigade tekkimise ning seega info valesti sisestamise võimaluse. Elemente, mida vormides kasutatakse, tuleks alati esitada kindlate üldtuntud nõuete kohaselt, mis tekitab kasutajas tuttavlikkust ning välistab võimalikke edasiulatuvaid probleeme vajaminevate andmete kogumisel. Seega tuleb iga vormi elemendi kohta seoses vajalikkuse ja optimaalsusega läbi mõelda järgnevad punktid:

- Kas tegemist on kindlasti vajamineva infoga;
- Kas info on piisavalt oluline, et takistada kasutajate ligipääsu sellele, mis peitub vormi "taga", kui nad ei otsusta infot sisestada. (Crescimanno, 2005)

Vormide koostamisel on esmatähtis tekstilise info oskuslik edastamine, seejärel erinevate vormielementide sisulist poolt toetav kasutajakeskne disain, kuna ebavajaliku või puuduliku info edastamine või küsimine veebivormides võib mõjutada nende kasutatavust.

Järgnevides peatükkides on toodud välja XHTML koodide näited ja ekraanipildid vormide esitamiseks. Dan Cederholm toob välja erinevaid võimalusi ekraanivormide esitamiseks XHTML abil. Kasutatavuse seisukohalt lähtudes on valitud definitsiooniloendite kasutamine. (Cederholm, 2004) See võimalus parandab erivajadustega kasutajate ligipääsu, kuid lisab ka kasutajakeskust tavakasutajale.

2.1 Kasutaja informeerimine

Veebivormide oluliseks osaks lähtudes nende täitmise efektiivsusest, kiirusest ja info kvaliteedist on kasutaja informeerimine, mis jaguneb vormi erinevatel etappidel:

- sissejuhatavaks informeerimiseks,
- abitekstiks lahtrite täitmisel (abiinfo menüü),
- veateadeteks,
- tagasisideks.

Kõik vormid võiksid soovituslikult alata sissejuhatava osaga, selgitamaks vormi eesmärke ja iseärasusi, mille teadmine enne täitmist aitaks kasutajal teostada tööd sujuvamalt. Sissejuhatav tekst peaks asuma iga vormi puhul nähtavas ja kättesaadavas kohas, näiteks iga vormi ülaosas. Samuti peaks kajastuma sissejuhatavas osas vormi täitmiseks kuluv aeg, küsimuste arv ja konfidentsiaalsuse põhimõtted.

Vormide täitmiseks vajaminev abimaterjal peaks olema kättesaadav kogu töö vältel ja asuma kergesti ülesleitavas ning nähtavas kohas vormide juures.

Veateadete koostamisel on soovitatav lähtuda järgmistest soovitustest:

- selge viitamine, et midagi on läinud valesti,
- koodide ja lühendite asemel loetavas keeles selgitus,
- viisakas sõnastus, mis ei süüdistata kasutajat ega anna mõista, et ta on rumal või teeb midagi keelatud,
- täpne kirjeldus probleemi kohta,
- vea parandamise nõuande konstruktiivsus,
- nähtavus ja märgatavus. (Nielsen, 2001)

Veateadete puudulik esitamine võib põhjustada info edastamise kiiruse langust või hoopis selle saamata jäämist. Seetõttu tuleks eelisjärjekorras vältida veateadete koodi sisaldavat tagasisidet, millel ei ole kasutaja jaoks informeerivat väärtust. Selle asemel tuleks koostada sobiv veateadete süsteem, mis annaks täpsema ettekujutuse vea tüübi kohta. Soovitatav oleks kasutajale kuvada lisaks veateatega täidetava vormi asemel ainult valesti täidetud osa, mis lihtsustab ja kiirendab lõpptulemuseni jõudmist oluliselt. Sama põhimõtte kehtib ka täitmata unustatud kohustuslike väljade puhul. (Krug, 2000)

Tagasiside on oluline seetõttu, et anda mõista kasutajale töö õnnestumisest, informeerida edasistest toimingutest ja tänada tehtud töö eest. Vastasel juhul võivad tekkida lisaküsimused, millele vastuse saamiseks peab kasutaja leidma muud moodused.

Kasutatavuse tõstmise vajadusest tulenevalt on igasugune informeerimine vormidega töötamisel äärmiselt vajalik ja seega nende loomulik osa.

2.2 Väljade pealkirjade kasutamise põhimõtted ja võimalused

Väljade pealkirjad ehk sildid (*labels*) aitavad selgitada, mis tüüpi teksti on vaja sisestada ning selle sisestamise tingimusi. Väljade pealkirjad on tugevalt seotud erinevate kontrollijatega, mis võimaldavad sisestatava info jaoks teha valikuid. Iga välja pealkiri on seotud konkreetse kontrollijaga.

Väljade ja pealkirjade märkimisel võib kasutada järgnevat koodi.

```
<form action="path/to/script" id="veebivorm" method="post">
  <dl>
    <dt><label for="eesnimi">Eesnimi:</label></dt>
    <dd><input type="text" id="eesnimi" name="eesnimi" /></dd>
    <dt><label for="pnimi">Perekonna nimi:</label></dt>
    <dd><input type="text" id="pnimi" name="pnimi" /></dd>
    <dt><label for="email">E-meil:</label></dt>
    <dd><input type="text" id="email" name="email"/></dd>
  </dl>
</form>
```

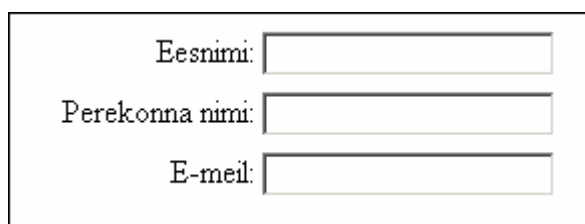
Näidis koodis on kasutatud defineeritud nimekirju parema paigutuse ja häälbrowseritega ligipääsu tagamiseks.

```
form dd {
  float:left;
  width:66%;
}
```

Siin määratakse kontrollijate asetus.

```
form dt{
  clear:both;
  width:10%;
  float:left;
  text-align:right;
}
```

Laadilehel määratakse elementide pealkirjade asetus. Antud näite korral aitab kood muuta elementide ja nende pealkirjade paigutuse horisontaalseks. Ilma laadileheta oleks paigutus vertikaalne.



The image shows a rectangular box containing three input fields. The first field is labeled 'Eesnimi:', the second 'Perekonna nimi:', and the third 'E-meil:'. Each label is positioned to the left of its corresponding input field, and the fields are stacked vertically.

Joonis 2. Näide väljade pealkirjade määramisest.

Tabel 1. Väljade pealkirja elemendi atribuut ja väärtus.

Element	Atribuut	Väärtus
label	for	kontrollija id

Kontrollijate ja pealkirjade määramisega enam esinevad probleemid on:

- puuduvad väljade pealkirjad,
- puudub ühemõttelisus,
- äraspidine loogika,
- puudub mõistetav märgede kohustuslike väljade kohta,
- ebäühtlane joondamine,
- puudub stabiilselt kasutatav sama paigutuse stiil,
- väljade pealkirjade suur kaugus kontrollijatest.

Järgnevalt on kõikide nimetatud punktide kohta toodud näited, et selgitada kui tihti esineb veebilehekülgedel kasutatavuse probleeme.

Puuduvad väljade pealkirjad

Väljade nimetused peaksid olema selged, arusaadavad ja ühemõttelised ning sõltumatud metafooridest, mis võivad olla regioonipõhised või mitmeti tõlgendatavad. Seetõttu on vajalik määrata igale kontrollijale pealkiri, lisaks vaikimisi väärtustele, välistades sellega võimalikke eksimusi valesti mõistmisel ning lisades vormidest arusaamise kiirust.

Keeleoskus		
	Kõnes	Kirjas
<input type="checkbox"/> eesti	Määramata	Määramata
<input type="checkbox"/> vene	Määramata	Määramata
<input type="checkbox"/> inglise	Määramata	Määramata
<input type="checkbox"/> saksa	Määramata	Määramata
<input type="checkbox"/> prantsuse	Määramata	Määramata
<input type="checkbox"/> soome	Määramata	Määramata
	Määramata	Määramata

Joonis 3. Veebivorm Hyppelaud keskkonnas. www.hyppelaud.ee.

Puudub ühemõttelisus

Pilet.ee keskkonnas on tegemist puuduliku pealkirjade ühemõttelisuse, grupeerimise ja ebaloogilise paigutusega. Selle tulemusel võib erineva arvutikasutamise oskusega sihtgrupp teha antud vormi mõistmisel erinevaid, ebatäpseid või isegi valesid järeldusi ja otsuseid, millel võib olla edasiulatuv mõju.

Teavitused

NB! Pilet.ee ei vastuta teavituste eest, mille toimimine sõltub vastavate teenuste osutajatest (nt. SMS, mis sõltub mobiiloperaatorist ja e-kiri, mis sõltub e-posti teenuse pakkujast)

Saadetakse pileti omanikule ostjale (kui pilet osteti teisele isikule)

E-postiga @eesti.ee aadressile (tasuta). Loe e-posti teavituse kohta lisa...

SMS (pileti hinnale lisandub 3 kr)

Pileti aktiveerimisel

2 päeva enne kehtivuse lõppu

Kehtivuse lõppemisel

Ei saadeta

Pileti aktiveerimisel

2 päeva enne kehtivuse lõppu

Kehtivuse lõppemisel

Mob. number:

Joonis 4. Veebivorm Pilet keskkonnas. www.pilet.ee.

Äraspidine loogika

Kõige ohtlikumaks soovitud info kogumisel võib osutada kontrollijate pealkirjade korral äraspidise loogika kasutamine, mille puhul keelamiseks tuleb märkida nõusolek või lubamiseks märkida keeldumine. Seetõttu on oluline tähelepanu juhtida väljade pealkirjade ühemõttelisusele ja loogilisele esitusele. Üsna sagedasti esineb raadionuppude ja märkeruutude vastupidist kasutust, mistõttu tuleb selgeks teha ühe või teise elemendi eelised ja puudused ning seeläbi mõista nende kasutamispõhimõtteid. Järgnevas näites on märkeruudu asemel kasutatud raadionuppe. Näites esitatud küsimus tuleks esitada kujul „Soovin saada CV-Online infokirju“ ja kasutada raadionuppude asemel märkeruutu, mille puhul kasutaja saaks soovi korral valida nõusoleku.

Kas soovite CV-Online infokirju?

Jah Ei

Joonis 5. Veebivorm CV-Online keskkonnas. www.cv.ee.

Puudub mõistetav märke kohustuslike väljade kohta

Kasutajale tuleb selgeks teha, milliste väljade täitmine on kohustuslik ja milliste valikuline, kuna see võib oluliselt parandada vormide täitmise tulemuslikkust. Kohustuslike väljade märgistamiseks on kõige sobivamaks ja laialdasemalt arusaadavaks kujunenud pealkirja tärniga tähistamine, mis võib hoida ära palju asjatuid kasutaja aega raiskavaid pingutusi vormide täitmisel. Antud kontekstis on kohustusliku välja märgistamiseks kasutatud pilti,

millel puudub alt väärtus, seega ei saa ekraanilugejad seda tõlgendada. Lisaks seostub hääumärk ning punane värv ka hoiatusega, mida kasutatakse erinevate ettevaatusele kutsuvate tähiste märgistamiseks.

Form fields and labels:

- Domeen:
- Ettevõtte nimi:
- Postiaadress:
- Telefon:
- Reg.nr:
- Kontaktisik:
- Isikukood:
- Postiaadress:
- Telefon:
- E-post:
- Hostingu pakett: -vali siit-
- Märkused:

Buttons: Saada!

Joonis 6. Veebivorm Sevenline keskkonnas. www.sevenline.ee.

Ebäühtlane joondamine

Seoses ebaprofessionaalse välimusega, toob taoline paigutus kaasa kodulehekülje kvaliteedi seostamise müüdavate toodete või pakutavate teenuste kvaliteediga. Kiiruga ja lohakalt valmistatud veebilehekülg võib kasutajale otseselt seostuda samasuguse teenindusega. Lisaks mõjutab ebäühtlane joondus vormides orienteerumise kiirust, kuna ülevaate saamiseks peavad silmad peatuma igal lahtri alguspunktil eraldi selle seostamiseks pealkirjaga. Ebäühtlase joonduse tõttu on kasutajal sõltumata grupeerimiselementide kasutamisest lahtreid komplektidena raske tõlgendada.

Kasutajaks registreerimine

Sisesta palun tämiga täidetud väljade andmed:

Isikuandmed:

Eesnimi*:

Perenimi*:

Sünnipäev:

Kontakt:

Aadress*:

Telefon*:

E-post*:

E-postile saadetakse registreerumisinfo!

kfoto kasutaja:

Kasutajanimi*:

Parool*:

Parool 2x*:

SALVESTA

Joonis 7. Veebivorm K-Arvutisalongi keskkonnas. www.k-arvutisalong.ee.

Puudub stabiilselt kasutatav sama paigutuse stiil

Väljade pealkirjade paigutus on oluline mitmetel põhjustel nagu kasutamiskiirus, kasutamiskiiirus, kasutamiskiiirus ja professionaalne lehekülje välimus. Kui valitud joenduse tüüpi pole stabiilselt kogu vormi ulatuses kasutatud, siis jätab see visuaalselt ebaprofessionaalse ja ebausaldusväärse mulje.

Registreerimine

Eesnimi: *

Perekonnanimi: *

Kasutajanimi: *

E-post: *

SMS e-post:

Selgitus melli kohta [Kontrolli](#)

Parool: *

Parool uuesti: *

Amet (soovitav):

Sugu: Mees

Snniaeg: 01 01 1900

Kool:

Aadress:

Vald/Linn:

Maakond: Harjumaa

Kinnitan, et minu poolt edastatud andmed on korrektsed.

Joonis 8. Ekraanivorm IT Koolituse keskkonnas. www.koolitus.ee.

Väljade pealkirjade suur kaugus kontrollijatest

Suureks probleemiks võib kujuneda pealkirjade asetus väljade suhtes, kuna tegemist on omavahelise tugevalt seotud elementide süsteemiga. Seetõttu on erinevad kasutajakesksusega tegelevad spetsialistid teinud mitmeid uurimusi seoses paigutuse efektiivsusega ning jõudnud erinevatele järeldustele, milline neist kõige paremini toimib. Erinevatest paigutuse tüüpidest tuleb lähemalt juttu veebivormide kujunduse peatükis.

Palun täida registreerimisvorm

Kasutajanimi: *

Parool: *

Parool veelkord: *

Sinu e-posti aadress: *

Palun sisesta kontrollkood, mida näed pildil: *



* märgitud väljad peavad olema täidetud.

Joonis 9. Ekraanivorm Delfi keskkonnas. www.delfi.ee.

2.3 teksti väljade ja tekstialade kasutamise põhimõtted ja võimalused

Teksti väljad (*input*) ja tekstialad (*textarea*) on kõige enam kasutatavad vormi elemendid, lubades kasutajal vabas vormis sisestada vastuseid.

Tabel 2. Elemendi `input` atribuudid ja nende väärtused.

Element	Attribuut	Väärtus
input	type	text password checkbox radio submit reset file hidden image button
	name	kontrollija nimi
	value	kontrollija väärtus
	size	Kontrollija laius pikslites,
	maxlength	Maksimaalne tähtede arv, mida saab sisestada
	checked	määrab vaikimisi väärtust
	src	Type:image korral määrab asukoha täideviimise nupu suhtes.

Tabel 3. Elemendi `textarea` atribuudid ja nende väärtused.

Element	Attribuut	Väärtus
textarea	name	Määrab kontrollija nime
	rows	Nähtavate tekstiridade arv
	cols	Nähtav laius

Järgneva näite abil saab koostada tekstilahtreid ja tekstialasid. Kasutatavuse tõstamiseks on kasutatud defineeritud nimekirju.

```
<form action=" path/to/script " id="yks" method="post">
  <dl>
    <dt><label for="eesnimi">Eesnimi:</label></dt>
    <dd><input type="text" id="eesnimi" name="eesnimi" /></dd>
    <dt><label for="pnimi">Perekonna nimi:</label></dt>
    <dd><input type="text" id="pnimi" name="pnimi" /></dd>
    <dt><label for="tekstilahter">Iseloomustus:</label></dt>
    <dd><textarea id="tekstilahter" name="tekstilahter" rows="9"
cols="15">
Esimene rida.
Teine rida.</textarea>
</dd></dl></form>
```

Laadilehel on määratud elementide paigutus. `form dd` määrab kontrollijate asetuse.

```
form dd {  
    float:left;  
    width:66%;  
    margin:0 0 0.5em 0.25em;  
}
```

```
form dt{  
    clear:both;  
    width:10%;  
    float:left;  
    text-align:right;  
}
```

`form dt` määrab elementide pealkirjade asetuse. Laadilehe lisamine aitab muuta elementide ja nende pealkirjade paigutuse horisontaalseks. Ilma laadileheta oleks paigutus vertikaalne.

Eesnimi:

Perekonna nimi:

Iseloomustus:

Joonis 10. Tekstiväljade ja tekstiala näidis.

Väljad on kõige tavapärasemad vormi elemendid, lubades kasutajal vabas vormis sisestada vastuseid, mille tõttu on nende täitmisel tekkida võivate vigade arvu raskem hallata ja seega ka kontrollida. Vajalik oleks seada väljadele tähemärkide arvu ja tüübi piiranguid, mis vähendavad oluliselt ebavajaliku, vales formaadis või liiga pika info sisestamist. Kui andmed võimaldavad, siis kasutajakeskuse huvides võiks suurte ja väikeste tähtede kasutamine olla lubatud. Kontrollelementide hulgast valiku langetamisel tuleks lähtuda põhimõttest, millise kasutamine neist toob kaasa kõige väiksema vigade tekkimise võimaluse ehk info valesti sisestamise tõenäosuse. Näiliselt, kuid ekslikult, võivad väljad tunduda sobivad kõikide andmetüüpide kogumiseks, aga oodatavat vastusetüüpi silmas pidades, tuleb alati valida optimaalseim kontrollija, mis tähendab ühest küljest kasutaja

võimaliku eksimuste viimist miinimumini ja teisest küljest küsitava info ning sellele vastamise selgust ja loogilisust.

Väljade kõrguse ja laiusega tuleb tagada kasutusmugavus, hõlmates nii kirjutamist kui ka sisestatud teksti ülelugemist ning vajadusel ka muutmist eesmärgiga rõhutada seost oodatavate sisestatavate andmete hulga ja lahtri pikkuse vahel.

Andmete sisestamist tekstiväljadesse peetakse põhjendatuks järgmistes olukordades:

- struktureerimata andmed nagu inimeste nimed, aadressid, kirjeldused ja kommentaarid;
- turvalisust nõudvad andmed nagu kasutajanimed ja paroolid;
- rahvusvaheliselt varieeruvad andmed nagu telefoni numbrid ja postiindeksid;
- andmed, mida kasutaja kindlasti peast teab nagu aadress või sünnikuupäev. (Johnson, 2003, lk 124)

Võttes arvesse kasutajakesksuse põhimõtted, tuleks lahtrite täitmine muuta eksimatult selgeks ja loogiliseks, muutes selleks kas väljade kujundust sisestatavale infole vastavalt või pakkuda juhtnööre.

2.4 Eeltäidetud väljade kasutamise põhimõtted ja võimalused

Eeltäitmise all peetakse silmas nii lahtrite, raadionuppude, märkeruutude kui ka menüüde ühe võimaliku valiku esile toomist sellist, et anda kasutajale mõista täitmise struktuurist, võimalustest või tavapärasemast vastusest.

Järgnevalt on esitatud näidiskood eeltäidetud väljade esitamiseks.

```
<form action=" path/to/script " id="yks" method="post">
  <dl>
    <dt><label for="url">Sisesta aadress:</label></dt>
    <dd><input type="text" id="url" name="url" value="http://"
  /></dd>
</dl>
</form>
```

Siin määratakse elemendi pealkirja ja kontrollijate asetuse parameetrid CSS laadilehe abil.

```
form dd {
    float:left;
    width:66%;
}

form dt{
    clear:both;
    width:10%;
    float:left;
    text-align:right;
}
```

A screenshot of a web form. It features a label 'Sisesta aadress:' followed by a text input field. The input field contains the text 'http://'. The entire form is enclosed in a rectangular border.

Joonis 11. Eeltäidetud välja näidis.

Kuna vormi elemente on võimalik sõltuvalt küsitavatest andmetest eeltäita, saab seeläbi oluliselt vähendada kasutajal vormide täitmiseks kuluvat aega ja kindlustada soovitud formaadis andmete saamist. Eeltäitmist vajavate kontrollijate ning nende hulga peab kindlaks määrama küsitavate andmete keerukuse, lihtsustamise vajaduse ning ajalise ressursi piiratus. Kasutajakesksuse põhimõtetest lähtudes on soovitatav määrata vormi elementidele vaikumisi enamvastatavad väärtused.

Võttes arvesse eeltäidetud väljade eeliseid veebivormide kasutatavuse tõstmiseks, tuleb tunnistada, et esineb andmeid, mille puhul eeltäidetud lahtreid ei saa, ei ole soovitatav kasutada või avaldab nende kasutus oodatule vastupidist mõju. Järgnevalt on välja toodud kaks olulist põhjust eeltäidetud väljade kasutuse vältimiseks:

- kui ei eksisteeri mõistlikku ja kasulikku ning sobivat vaikumisi väärtust nagu rahvuste ja elukohtade aadresside korral,
- sotsiaalsetest, poliitilistest ja õiguslikest parameetritest tulenevad nõuded eesmärgiga vältida igasugust kallutatust. (Johnson, 2003)

Eeltäidetud väljade kasutamise puhul on eesmärgiks ausa vastuse saamine, ilma et etteantud näidis kasutajat mõjutaks või teda arvamuste ja andmete edastamisel erinevate võimaluste poole kallutaks. Kasutajate mõjutatavuse ohutegurit võib lugeda keskmisest kõrgemaks, eriti kui tegemist on uue keskkonnaga, millega puuduvad eelnevad kogemused

ja arusaamised süsteemi toimimise põhimõtetest ning mudelitest ei ole kujunenud selget ettekujutust. Kuna kasutajad võivad eeltäidetud andmeid ekslikult pidada enam soosituteks või õigemateks, võib mõjutatavuse ohutegurit hinnata kõrgeks.

Eeltäidetud väljade puhul tuleb viia see kooskõlla nende täitmise põhimõtetega hõlmates valikute langetamise võimaluste arvu ja korda, et mitte jätta kasutajale ekslikult valet muljet täitmata jätmise võimaluse suhtes. Eeltäidetud raadionuppude valikutest peaks üks neist olema vaikimisi määratud, et anda kasutajale mõista valikute võimalusest, milleks on kas üks või teine mitte null või kaks valikut.

Kui vaikimisi valikud ei ole veebiarendaja poolt määratud siis tavaliselt tõlgendab lehitseja selleks nimekirja esimese väärtuse. Seetõttu võivad vaikimisi määrangud osutada äärmiselt eksitavaks ning nende määramine tuleks läbi mõelda ja planeerida vastavusse soovitava eesmärgiga. (Nielsen, 2005)

Eesmärgiks võib olla anda kasutajale mõista eeltäidetud valikute kaudu sagedamini vastatud väärtustest, soovitada kasulikumat või enamlevinud valikut. Ennekõike on selle rakendamine vajalik turvalisuse tagamisega kasutaja valikute sooritamisel, millel on edasiulatuv ja tagasipööramatu mõju. See vähendab oluliselt võimalust, et kasutaja eksib ja ei saa oma tehtud vigu enam tagasi pöörata ning peab seetõttu vormide täitmise katkestama.

2.5 Ripploendi ja ripploendikasti kasutamise põhimõtted ja võimalused

Ripploendi (*select menu*) ja ripploendikasti (*scrolling lists*) kasutamisel tuleb jälgida nende eesmärgipõhisust ning sellest lähtuvalt mugandada see omakorda kasutajakesksusega. Kuna menüüdega saab esitada väga erinevaid andmete väärtusi siis on oluline leida selleks optimaalne lahendus. Ripploendite eelised teiste valikelementide kasutamise ees on kompaktsus ja ruumisäästlikkus, kuna kasutaja saab valida sobiva vastuse paljude hulgast. Seejuures loogilises sorteeritud alfabeedilises või tüpograafilises järjekorras suurest hulgast valiku tegemine võtab oluliselt vähem aega ja säästab ekraanipinna ruumi.

Tabel 4. Eelemendi `select` atribuudid ja väärtused.

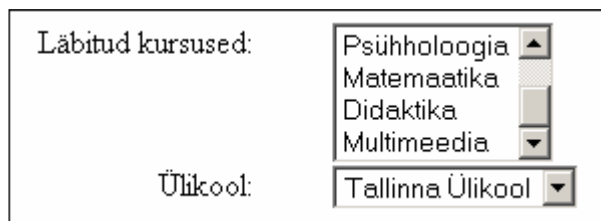
Element	Atribuut	Väärtus
select	name	Määrab kontrollija nime
	size	Nimekirja ridade arv
	multiple	Kui määratud lubab mitu valikut.

Järgnevalt on esitatud ripploendikasti ja ripploendi ülesmärkimise kood, kasutades selleks defineeritud nimekirju.

```
<form action="path/to/script" id="veebivorm" method="post">
  <dl>
    <dt><label for="kursused">L&uuml;bitud kursused:</label></dt>
    <dd> <select id="kursused" name="kursused" size="4">
      <option value="1">Loogika</option>
      <option value="2"> Filosoofia </option>
      <option value="3"> Ps&uuml;hholoogia</option>
      <option value="4"> Matemaatika</option>
      <option value="5"> Didaktika</option>
      <option value="6"> Multimeedia</option>
    </select>
  </dd>
  <dt><label for="kool">&Uuml;likool:</label></dt>
  <dd><select id="kool" name="kool">
    <option value="tlu"> Tallinna &Uuml;likool</option>
    <option value="tu">Tartu &Uuml;likool</option>
    <option value="ttu">TT&uuml;</option></select></dd>
  </dl>
</form>
```

Laadilehtede kood muudab paigutuse kasutajakesksemaks.

```
form dd {  
    float:left;  
    width:66%;  
}  
form dt{  
    clear:both;  
    width:10%;  
    float:left;  
    text-align:right;  
}
```



Läbitud kursused:	Psühholoogia Matemaatika Didaktika Multimeedia
Ülikool:	Tallinna Ülikool

Joonis 12. Ripploendi ja ripploendikasti näidis

Ripploenditest valikute langetamine vähendab oluliselt valede, ebatäielike või ebavajaliku info sisestamist. Ripploendite kasutamine ebastandardsetes olukordades, ebaselge eesmärgiga või kasutajakesksusega arvestamata võib loodetava kasu asemel vastupidise efekti saavutada. Ripploendite kasutamisel tuleks vältida:

- interaktsioone põhjustavad menüüd, mille puhul muutus ühes menüüs kutsub esile muutuse teises,
- väga pikad menüüd, mis ei mahu korraga ekraanile ja nõuavad seetõttu vastuste nägemiseks kerimist,
- menüüsid riigi lühenditega, mida oleks kasutajal lihtsam ja ajasäästlikum sisestada lahtrisse,
- menüüsid kasutajale teadaolevate andmetega nagu sünnikuupäevad. (Nielsen, 2000)

Jakob Nielsen on ripploendi puudusena välja toodud omadust mahutada palju infot, millele saab ligi pääseda üksnes kerimisriba kasutades. Seda võib käsitleda ka eelisena, kuna võimaldab mahuka info mõistlikus põhjendatud mahus näitamise seejuures mitte ajades kasutajat segadusse korraga ekraanile ilmuva liiga suure hulga valikutega.

Ripploendite ja ripploendikastide põhiliseks ohuks seoses kasutajakeskuse ja andmete õigsusega võib kujuneda olukord, kus hiire rullikuga tahtmatult edasiliikumise kavatsuse asemel muudetakse juba määratud väärtus. (Nielsen, 2000) Taoline olukord võib tekitada suuri segadusi nii kasutaja kui info koguja seiskohast lähtudes ning põhjustada mõlemale osapoolle suuri ebameeldivusi.

2.6 Raadionuppude ja märkeruutude kasutamise põhimõtted ja võimalused

Raadionuppude ja märkeruutude kasutamisel tuleb kontrollijate pealkirjad esitada loogiliselt ja selgelt. Oluline on eristada elementide kasutamise põhimõtteid ja neid omavahel mitte segi ajada, kuna see võib tekitada olulisi kasutatavuse probleeme. Lisaks tuleb arvestada nende ülesmärkimisel erivajadustega kasutajatele ligipääsu tagamisega, mistõttu peaks võimaldama häälbrowseritega ja ainult klaviatuuriga kasutamist.

Tabel 5. Elemendi select atribuudid ja väärtused.

Element	Attribuut	Väärtus
select	name	Määrab kontrollija nime
	size	Nimekirja ridade arv
	multiple	Kui määratud lubab mitu valikut.

Järgnev kood näitab märkeruutude ja raadionuppude üles märkimist, kasutades selleks `fieldset` ja `legend` elemente. WCAG metoodikad soovitavad raadionuppude ja märkeruutude gruppide korral kasutada `legend` ja `fieldset` elemente, et tagada ekraanilugejatega paremat kasutatavust ning elementide grupeerimist. Antud näites loeks häälbrowser vorme nii, et iga valiku ette täpsustatakse ka `legend` väärtus. See võib osutuda möödapääsmatult vajalikuks, kui tegemist on näiteks numbrite või koodide loeteluga.

```

<form action="path/to/script" id="yks" method="post">
<fieldset>
<legend>Haridus:</legend><br />
<input id="kesk" type="checkbox" name="checkbox" value="checkbox" />
<label for="kesk">Keskharidus</label><br />
<input id="bak" type="checkbox" name="checkbox2" value="checkbox" />
<label for="bak">Bakalaureus</label><br />
<input id="mag" type="checkbox" name="checkbox3" value="checkbox" />
<label for="mag">Magister</label>
</fieldset>
<fieldset>
<legend>Sugu:</legend><br />
<input id="mees" type="radio" name="radio" value="mees" />
<label for="mees">Mees</label><br />
<input id="naine" type="radio" name="radio" value="naine" />
<label for="naine">Naine</label><br />
</fieldset>
</form>

```

CSS laadilehel oleva koodiga on muudetud elementide paigutust.

```

form fieldset{
  clear:both;
  width:10%;
  float:left;
  text-align:left;
}

```

Joonis 13. Raadionuppude ja märkeruutide näidis

Raadionuppude ja märkeruutude kasutamisel tuleb ennekõike mõelda selge tekstilise ülesehituse peale, pakkudes kasutajale loogilisi valikuid ja mitte teineteist osaliselt või täielikult katvaid võimalusi. Kuna raadionuppude ja märkeruutude kaudu esitatavad küsimused on reeglina ei või jah vastuseid nõudvad siis automaatselt valides ühe kirje on juba küsimusele vastatud ning pole vajadust vastupidise küsimuse esitamiseks. Kui

valikuks on pakutud kirje, mille mittemärkimisel on selge, et valikut ei soovita, siis järgnevate märkeruutude või raadionuppudega ei peaks kasutajat sundima mittesoovitud kirjed uuesti kinnitama.

W3C spetsifikatsioonid ja üldlevinud graafilise kasutajaliidese põhimõtted annavad järgmised seletused:

- Raadionupud — kasutatakse pakkumaks kahte või enam valikut, mis on teineteist välistavad, lubades kasutajal teha nende hulgast ühe sobiva valiku. Seega ühe kirje valimisel välistatakse teiste kirjete sobivus.
- Märkeruudud — kasutatakse kui soovitakse kasutajal lasta valida kirjete hulgast null kuni mitu sobivat vastet. Seega erinevad nad raadionuppudest selle poolest et pole omavahel seotud valikute arvuga.
- Eraldiasetsev märkeruut — kasutajalt soovitakse nõusolekut, millega ta saab nõustuda või keelduda. (Nielsen, 2004)

Märkeruutude ja raadionuppude disaini puhul on oluline jääda standardsuse piiridesse, et kasutajakesksuse huvides äratada tuttavlikkust. Kuna märkeruudud ja raadionupud võivad sõltuvalt vormi tüübist olla valdavad siis peab nende grupeerimisele pöörama erilist tähelepanu. Elementide grupeerimisel lähtutakse ennekõike nende sisulisest eripärast ning alles seejärel vormilisest.

Igasugused erinevat tüüpi nimekirjad tuleks lihtsuse ja kiiruse huvides paigutada vertikaalselt üksteise alla vältides horisontaalpaigutust kui segadust tekitavat ebaloogilist lahendust.

Raadionuppude eelised võrreldes ripploenditega on valikute visuaalne nähtavus, mis võib aidata kasutajat lihtsamini järgmiste etappideni jõuda. Raadionupud erinevad ripploendist lihtsuse ja selguse poolest, pakkudes tavaliselt vähem valikuid. Seetõttu sobivad nad kompaktsema info jaoks paremini kui suure hulga valikvastuste jaoks. Samas võtavad nad oluliselt rohkem ruumi ripploenditega võrreldes. Seega tuleb leida sõltuvalt eesmärgist sobiv lahendus ja kaaluda mõlemate elementide tugevaid ning nõrki külgi.

Erinevate valikute pakkumine kirjetega peab olema põhjalikult läbimõeldud ja ette planeeritud arvestades kõiki võimalikke sihtgrupe. Välistada tuleks võimalikud olukorrad, kus kasutaja ei leia nimekirjast sobivaid kirjeid ega alternatiivset lahendust. Kasulik oleks pakkuda valikut “teine variant“ või sisestatavat tekstilahtrit. Parema kasutajakesksuse huvides peaks valiku märkimisel võimaldama kasutajal seda sooritada lisaks nupule ka

lahtri pealkirjal vajutades, laiendades seeläbi märkeala. Samuti on kasulik määrata ka klaviatuurilt juurdepääsu kombinatsioonid, kuna see mõjutab erivajadustega kasutajate ligipääsu ja muudab kasutamise mugavamaks. Seejuures tuleb tähelepanu pöörata stabiilsusele klahvikombinatsioonide märkimisel.

2.7 Nuppude kasutamise põhimõtted ja võimalused

Veebivormide kujundamisel ja loomisel eristatakse kolme tüüpi nuppe: täideviimise nupud (*submit*), väljad puhastamise nupud (*reset*), nupud erinevate tegevuste sooritamiseks (*push button*).

Nuppude kasutamisel on oluline nende nimetuse ühemõttelisus ja tuttavlikkus, mis võimaldab ilma eelneva kokkupuuteta nendega toimetamist. Erivajadustega kasutajale ligipääsu tagamiseks ja kasutatavuse tõstmiseks peaks olema määratud nuppudele `alt` väärtus. Sellisel juhul saab häälbrauser tõlgendada pildi väärtust ja graafilisel brauseril puuduva pildi asemele selle nimetuse kuvada.

Tabel 6. Elemendi `button` atribuudid ja väärtused.

Element	Attribuut	Väärtus
button	name	Kontrollija nimi
	value	Algne väärtus
	type	submit button reset

Järgnev koodi näide on nuppude ülesmärgistamise kohta.

```
<form action="path/to/script" method="post">
  <button type="submit" name="submit" value="submit" alt="Saada" >
    Saada </button>
  <button type="reset" name="reset" value="reset"
alt="T&uuml;hjenda" >
    T&uuml;hjenda</button><br />
<p>
  <button type="submit" name="submit" value="submit" >
    </button>
    <button type="reset" name="reset" value="reset">
    </button></p>
</form>
```



Joonis 14. Nuppude näidis.

Joonisel 14 ülemises reas on välja toodud nuppude „Saada“ ja „Tühjenda“ esitamine. Alumisel real on näidatud elemendi `alt` kasutamise vajadus, kui mingil põhjusel nuppude esitamiseks määratud pildid ei ole brauserile kättesaadavad, siis kuvab brauser ikkagi `alt` väärtusena nupu nimetuse.

Nupud käivitavad soovitud tegevuse, kas saadavad info ära või tühjendavad väljad. Täideviimise nuppe peaks vormi kohta olema ainult üks, kandes vastavat märget või nimetust. Ka väljade puhastamise nuppude puhul peaks mäрге olema piisavalt selge eristamiseks kustutamise ja esialgse info taastamise funktsioone. Vastasel juhul võib tekkida soovitule vastupidine olukord, kus vajalik info kaob.

Nupud peaksid asetsema vormide all vasakule joondatud. Uuringud on näidanud et kasutajad otsivad täideviimise nuppe esmajärjekorras vormide alt. (Brinck et al., 2002, lk 325)

Nuppude olulisust vormi koostamisel ei tohiks alahinnata, kuna tegemist on vormi täitmise kõige olulisema etapiga, mille õnnestumisel või sooritamisel on suur tähtsus kasutatavuse määramisel. Kui vormide täitmise järgselt ei suudeta infot edastada või tegevust sooritada, siis võib üsna tõenäoliselt pidada loobumist või frustratsiooni tingituna kulutatud ajast ja pingutusest.

2.8 Veebivormide perspektiiv ja võimalused lähitulevikus

Veebivormide kasutuselevõtu ja internetirakenduste arenguga on kerkinud esile mitmed probleemid ning nendest tulenevalt tekkinud vajadus lisavõimaluste ja uute lahenduste järele. W3C loodud veebivormide viimane uuendus on tehtud aastal 1999, mis on liiga pikk aeg, arvestades kui kiiresti arenevad erinevad infotehnoloogilised rakendused. W3C on välja pakkunud kaks võimalust veebivormide edasiarenduse või alternatiivina kasutamiseks Web Forms 2.0 ja Xforms 1.0.

Xforms on XML rakendus, mis esindab järgmise põlvkonna veebivorme, jagades traditsioonilised XHTML vormid kolmeks: Xforms mudel, andmed ja kasutajaliides. (W3C, 2005) Xforms ei ole iseseisev ülesmäärgistamise formaat, vaid teda saab integreerida XHTML ja SVG'ga. Lisaks saab seda kasutada erinevatel platvormidel nagu pc'd, pihuarvutid.

Võimalused mida pakub Xforms ja mida ei pakkunud HTML vormid on:

- kontrollida andmete väärtust samal ajal, kui kasutaja trüüb;
- näidata, et teatud väljad on kohustuslikud ja vormi ei saa edastada neid täitmata jättes;
- esitada vormide andmeid XML kujul;
- integreerida veebiteenustega nagu SOAP ja XML RPC;
- esitada sama vormi erinevatele serveritele (näiteks otsingu andmed erinevatele otsingumootoritele);
- salvestada ja taastada väärtusi failis ja failist;
- kasutada esitatud tulemusi sisendina edasistes vormides;
- saada algsed andmed vormi jaoks välistest dokumentidest;
- arvutada edastatavad väärtused teistest väärtustest;
- piirata väärtusi kindlal viisil ja nõudes, et nad on kindlas vahemikus;
- teha ostukorve ja viisardeid ning kujundada vorme ilma, et oleks vaja appi võtta skripte. (W3C, 2006b)

Xforms tehnoloogia suurimaks puuduseks peetakse asjaolu, et tänapäeva levinumad veebibrauserid ei toeta seda ning kasutamiseks on vajalik alla laadida ja intalleerida lisaks plugin. Selline lisatöö enne vormidega toimetamist võib kujuneda suureks tõrkeks juurutamisel. Seevastu Webforms on täielikult ühilduv brauseritega. Veebivormide standardile lisab perspektiivsust ühilduvus ja sobivus erinevate süsteemidega, seega on lähitulevikus suund kõikvõimalike erinevate süsteemidega ühilduvusele. See tähendab ennekõike kasutaja ligipääsemist erinevate operatsioonisüsteemide ja brauseritega.

Web Forms 2.0 on HTML 4 ja XHTML 1.0 laiendus ja edasiarendus. Samuti saab Web Forms 2.0 olema oluliseks osaks HTML 5.0 spetsifikatsioonis.

Põhilised Web Forms'i 2.0 eesmärgid on:

- tagasiulatuv ühilduvus, kus võimalik;
- haldamise lihtsus autoritel, kes on tuttavad HTML ja ECMAScript 'iga, aga teadmised XML kohta on piiratud;
- pakkuda uusi kontrollimisvõimalusi enam kasutatavatele elemendi tüüpidele, et autorid ei peaks korduvalt disainima keerulisi vidinaid nagu kalendrid;
- lihtsam kliendipoolne valideerimine;
- dünaamilisuse lisamine lahtritele kliendipoolselt ja ilma skriptideta;
- XML kaasamine ;
- võimaldada vorme luua välistest andmeallikatest, sel juhul ei pea autorid uuesti kirjutama vormi sisu, et vorme täita, mille asemel saab kasutada staatilisi lehekülgi skriptidega, mis dünaamiliselt genereerivad ainult andmete osa;
- spetsifikatsiooni rakendatavus täieliku vahendina piiratud ressurssidele; (W3C, 2005b)

Erinevalt Xformsist ei ole Web Forms'i eesmärgiks uueks märgistamiskeeleks kujunemine praeguste veebivormide asendamiseks vaid nende laiendamine ja täiendamine. Web Forms 2.0 eesmärgiks on ka lihtsustada XForms 1.0 dokumentide ümbertõlgendamist, mida saab sel juhul esitada html veebibrauseritega, mis ei toeta Xforms'i. Seega on tegemist justkui kahe teineteist täiendava lahendusega, millest võivad aja jooksul edasiarenduste käigus kujuneda teineteisele ka alternatiivid.

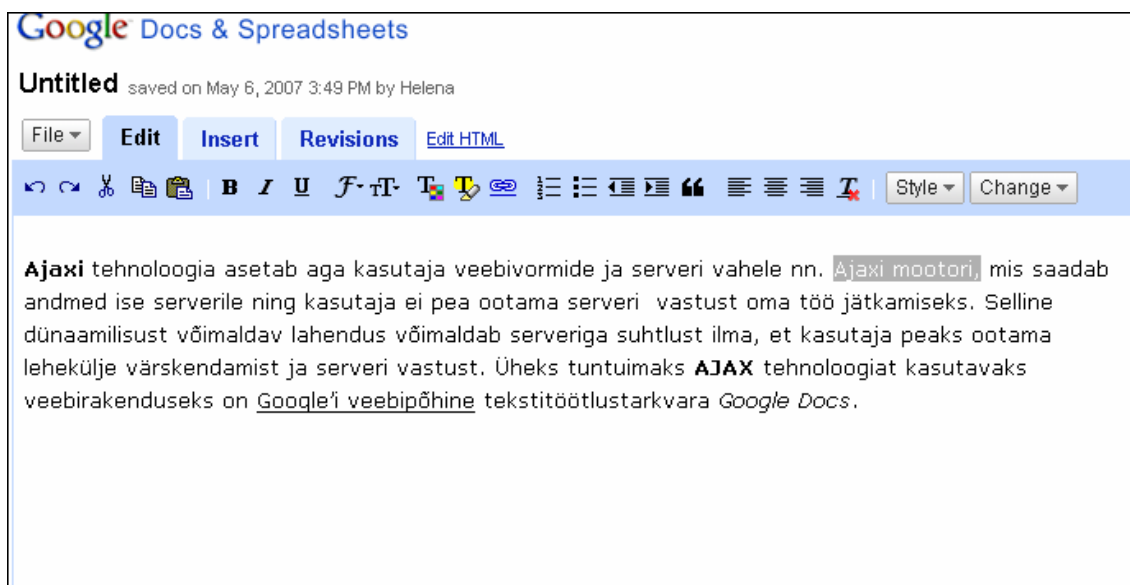
Lisaks eelnevatele veebivormide ülesmärgistamise võimalustele on kasutajakesksuse seisukohast vajalik ka vormide kui veebirakendustega toimetamise kiirus ja mugavus,

mille tagamiseks kasutavad paljud keskkonnad (Google) Ajax'i asünkroonset serveriga suhtlemist.

Ajax hõlmab endas mitut tehnoloogiat: XHTML ja CSS, Document Object Model, XML ja XSLT, XMLHttpRequest ja JavaScripti. (Garrett, 2005)

Tavalise veebivormi puhul toimub serverile andmete saatmine peale täideviimise nupu vajutamist ning peale töötlemist saadetakse tagasi uus lehekülg. Tagasiside ootamine muudab aga sõltuvalt serveri andmete töötlemise kiirusest interaktiivsuse taseme madalamaks, kuna kasutajad ei saa kohest vastust. Seda mõjutavad mitmed tegurid nagu näiteks interneti kiirus.

Ajaxi tehnoloogia asetab aga kasutaja veebivormide ja serveri vahele nn. Ajaxi mootori, mis saadab andmed ise serverile ning kasutaja ei pea ootama serveri vastust oma töö jätkamiseks. Selline dünaamilisust võimaldav lahendus võimaldab serveriga suhtlust ilma, et kasutaja peaks ootama lehekülje värskendamist ja serveri vastust. Üheks tuntuimaks AJAX tehnoloogiat kasutavaks veebirakenduseks on Google'i veebipõhine tekstitöötlustarkvara Google Docs.



Joonis 15. Google Docs veebipõhine tekstitöötlustarkvara.

Seoses veebivormide muutumisega üha olulisemaks rakenduseks internetis, kuna võimaldab oluliselt lihtsustada erinevate tehingute läbiviimist, on omaks võetud arusaam, et veebivormide kasutamisel võiks olla mõlema osapoole, nii andmete edastaja kui nende küsija, saadav kasu võimalikult suur. Seetõttu on tänase seisuga veebivormide kasutamise lisavõimaluste ja uute lahenduste järele nõudlus ning on välja pakutud erinevaid alternatiivseid lahendusi kas täiustamiseks või edasiarendamiseks.

3. Veebivormide kujundamise põhimõtted

Veebivormide kujundamise etapis tuleb lähtuda kasutajakesksuse ja WCAG põhimõtetest, pidades eelkõige silmas süsteemi visuaalse poole kasutamisihtsust.

Veebivormide kujundamisel tuleks lähtuda erinevatest põhimõtetest, mis puudutavad kasutaja käitumist internetis. Jakob Nielsen on uurinud kuidas inimesed veebi sisu loevad kasutades selleks kas F või E kujulisi mustreid. (Nielsen, 2006)



Joonis 16. Jakob Nielsen'i eyetracking mudeli tulemused.

Sellest lähtuvalt peaks vormide kujundamisel tegema järeldotsi elementide paigutamise võimalusi silmas pidades. F-kujulise mudeli järgimisvajaduse tingib kasutajate lugemis- ja kirjutamisharjumustega paralleelide tõmbamine. Kuna internetis on andmete käsitlemine, interpreteerimine ja haldamine erinev paberkandjatel olevatest, siis tuleb kujundamisel arvestada lugemismustreid.

Seoses internetis oleva info koguse ja sellest vajamineva leidmisega, on keskmisel arvutikasutajal välja kujunenud oskus olulise info eristamist ebaolulisest ehk vajamineva eristamine ebavajalikust. Seetõttu on oluline veebivormide disainimisel, et eesmärgile jõudmist ehk vormide täitmist ei segaks ega tähelepanu ei hajutaks taustamüra, milleks võivad olla pildid ja liigne disain erinevatel vormielementidel või nende taustal. Oluline on

püüda vormide täitmist kiirendada ja mitte panna kasutajaid mõtlema kõrvalise taustamüra üle.

Tähtis on mõista veebilehekülje ja veebivormide kujunduse erisusi, mis tulenevad nii nende sisulistest kui ka esituslikest eesmärkidest. Vormide disaini mõjutavate näitajatena võib välja tuua horisontaalse ja vertikaalse paigutuse, elementide grupeerimise, järjekorra, tausta ja värvid, ning tekstiga seonduvad parameetrid, mille kõigi ebastandardne või ebasobiv kasutus võivad põhjustada tõsiseid tõrkeid nii üldises kui ka erivajadustega kasutajate sihtgrupis.

3.1 Veebivormide struktuur ja esitamine mitmel veebilehel

Veebivormide struktuur hõlmab vormide esitamise erinevaid põhimõtteid: pikkust, tekstilist esitust ja elementidest tulenevat eripära eesmärgiga tagada paremat kasutatavust. Kui tegemist on pikkade vormidega mis ei mahu ekraanile ilma kerimisriba kasutamata, siis võib tekkida vajadus kaaluda nende esitamise otstarbekust ja loogilisust.

Soovituslikud nõuanded on järgmised:

- Üks vorm peaks mahtuma ühele leheküljele,
- Üks teema peaks mahtuma ühele leheküljele,
- Lehekülje kerimist võiks lubada kahe ekraanipikkuse ulatuses, mis lihtsustab ka printimist,
- Parem on panna vormi elemendid kahe tulba asemel ühte. (Jarrett, 2005)

Vormide poolitamine ebaloogilisel moel võib põhjustada tõrkeid ja raskendada nendes orienteerumist, seetõttu peaks mahukamate vormide esitamisel järgima põhimõtet jagada vormide esitamine erinevatele lehekülgedele teemade kaupa, seejuures mitte pikemalt kui kahe ekraanitäie ulatuses. Kui vormide pikkus ületab kahe ekraani pikkuse, võib see põhjustada info edastamisel segadust ja orienteerumise ning täitmise raskusi. Printimise vajadusega arvestamine sõltub konkreetsest vormi tüübist ja selle täitmise eripärast.

Vormide struktuuri hallatavuse eesmärgil on vajalik näidata kasutajale vormi pikkust koos selles olevate lehekülgede arvuga ning hetkeasukohta. Selleks on mitmeid võimalusi nagu sakkide ja menüüde kasutamine. Kõigil neil on omad puudused ja eelised, millega tuleks

eelnevalt sihtgrupi arvutikasutuse taset silmas pidades arvestada. Sakkide puhul on võimalik, et kasutaja ei taju nende toimimise mehhanisme näiteks muudatuste sisseviimisel, salvestamisel ja orienteerumisel. Kui tegemist on suurema arvu lehekülgedega, siis võib see põhjustada asjatut info otsimist ja ei pruugi viia tulemuseni. Elementide esitamine leheküljel peaks olema võimalusel ühe tulbana, kuna välistab vajamineva info märkamatuks jäämise. Seejuures võib kahetulbaliselt paigutatud elementide korral väheneda vormide ekraani kerimise vajadus ning nad on visuaalselt paremini haaratavad. Valiku langetamisel tuleb kahetulbaliste elementide korral arvestada grupeeriva kujunduse kasutamise vajadusega, mis rõhutaks elementide omavahelisi seoseid, nende täitmise järjekorda ning hulka. Tavaliselt täidavad kasutajad vorme lugemise järjekorras horisontaalselt, aga kahetulbalise esituse puhul võib alati tekkida segadus täitmise järjekorra suhtes, seega tuleks võimalusel eelistada ühetulbalist esitust, mille puhul on tegemist vertikaalse täitmisega. (Olsen, 2005a)

Veebivormide struktuuri puhul on oluline täitmist ja selle järjekorda puudutav asjaolu: kas tegemist on lineaarse või hargneva küsimustikuga. Kõige vähem tõrkeid esineb lineaarse struktuuri korral. Kuna alati pole aga võimalik seda rakendada, siis tuleb erinevate valikute pakkumisel lähtuda üheseltmõistetavusest ja loogilisusest.

Paljude valikute esitamisel kasutajale, suureneb ka valede valikute tegemise võimalus, kuna infot ei mõisteta üheselt ning kasutajate eelnevad teadmised ja kogemused on erinevad. Seetõttu tuleks viia valikute arv võimaliku miinimumini või asetada osaliselt kattuvad valikud ühe küsimuse alla. Kui näiteks kahes erinevas küsimuses on vajalik lisaks muule ka sarnase või sama info sisestamine siis võib neid küsimusi kombineerida vajamineval määral. (Olsen, 2005b)

Veebivormide navigeerimise planeerimisel on oluline selle asukoht, mis tagab kättesaadavuse ja kasutatavuse ning navigeerimise struktuur, mis on koostatud kasutaja vajadusi silmas pidades.

Mark Hurst propageerib isegi lihtsat kolmeetapilist lähenemist navigeerimise kasutatavusega seoses:

- Määra igal leheküljel kasutaja eesmärgid
- Eemalda või ära rõhuta elemente, mis ei aita eesmärki saavutada
- Rõhuta elemente ja vorme, mis viivad kasutaja eesmärgi saavutamisele lähemale või aitavad seda saavutada. (Hurst, 2004)

Vormide puhul ei peaks navigeerimise elementide kujundust üritama siduda lehekülje teiste osadega, kuna tegemist on erinevat eesmärki kandvates lehekülje osades. Seoses sellega peaks kogu viidete, menüüde ja linkide süsteem olema minimaalne ning kandma endas konkreetseid ja ainult eesmärgi saavutamiseks vajalikke andmeid. Elemente peaks kasutama tuttavlikul ja traditsioonilisel moel ning arvestama kasutajate eelnevate kogemustega.

3.2 Paigutus

Vormi elementide paigutamine stabiilselt ühtse struktuuri järgi määrab kõige suuremal määral nende visuaalse kasutatavuse.

Vormi elementide paigutuse korrektsuse vajalikkus on tingitud ühelt poolt erivajadustega kasutajate ligipääsu tagamiseks ekraanilugejatega ja teiselt poolt kasutajakesksuse tagamiseks. Kuna ekraanilugejad töötlevad infot suunaga vasakult paremale, järgides võimalikke ridasid, siis elemendid, mis on paigutatud üksteise alla, võivad põhjustada info valesse lahtrisse sisestamist. Kasutajakesksuse poole pealt vaadatuna peaks olema võimalik vormides liikumine kasutades selleks klaviatuuri *tab* klahvi, mille põhimõtte kohaselt liigub kursor samuti suunaga vasakult paremale, järgides võimalike ridade olemasolu. (WebAim, n.d)

Kujundamise etapis tuleb leida tasakaal disaini ja infoedastuse vahel selliselt, et loetavuse huvides oleks eesmärgi saavutamisele kaasa aitav minimalistlik taust ja põhjendatud mahus vaba ruumi. Vormide paigutamisel tuleb silmas pidada, et ühele leheküljele ei satuks liiga palju infot.

Erinevad kasutajakesksusega tegelevad spetsialistid on teinud mitmeid uurimusi seoses paigutuse efektiivsusega ning jõudnud erinevatele järeldustele, milline paigutus kõige paremini toimib.

Enamlevinud paigutuste liigid on:

- vertikaalne paigutus (joonis 17),
- vasakule joondatud horisontaalne paigutus (joonis 18),
- paremale joondatud horisontaalne paigutus (joonis 19),

- tausta, värvide ja joontega paigutus andmete grupeerimiseks või eraldamiseks (joonis 20).

Järgnevalt on toodud näiteid erinevate paigutuse liikide kohta.

Kasutajainfo	Isikuandmed	Kontaktandmed	Haridus ja töö	Muu info
Kasutajanimi *		E-posti aadress *		
<input type="text"/>		<input type="text"/>		
Minu CV staatuse määramine (vali palun sobiv variant) *				
- vali - <input style="float: right;" type="button" value="v"/>				
Parool *		Parool uuesti *		
<input type="text"/>		<input type="text"/>		
<p>Muudatuste salvestamiseks vajutage palun "EDASI" või "TAGASI" nuppu. Kõikide sisestatud andmete lõplikuks salvestamiseks, liikuge CV valiku "Muu info alla" ning vajutage nuppu "Lisa CV".</p>				
<input type="button" value="EDASI »"/>				

Joonis 17. Vertikaalne paigutus HypeLaud keskkonnas. www.hypeLaud.ee

* - väljad on kohustuslikud

Ees- ja perekonnanimi:

E-mail:

Telefon:

Asukoht:

Postiindeks:

Soovitud ajakirja numbrid:

Lisainfo:

Joonis 18. Vasakule joondatud horisontaalne paigutus ajakiri Paat keskkonnas. www.ajakiripaate.ee

* Eesnimi:	<input type="text"/>
* Perekonnanimi:	<input type="text"/>
* Sünnikuupäev:	1 <input type="text"/> jaanuar <input type="text"/> - <input type="text"/>
* E-mail:	<input type="text"/>
* Aadress:	<input type="text"/>
* Linn:	<input type="text"/>
* Postiindeks:	<input type="text"/>
Telefon kodus:	<input type="text"/>
Telefon tööl:	<input type="text"/>
Mobiil:	<input type="text"/>
<input type="button" value="Liitun"/>	

Joonis 19. Paremale joondatud horisontaalne paigutus ajakiri Otto keskkonnas.
www.otto.ee

Sisesta teabenõue: valige õige MTA struktuuriüksus	
Kategooria	<input type="text" value="Teabenõue"/>
MTA struktuuriüksus, kellele:	<input type="text" value="MAKSU- JA TOLLIAMET"/>
Teie isikukood/asutuse reg.kood	<input type="text"/>
Teie nimi/asutuse nimetus	<input type="text"/>
Teie sideandmed:	
Postiaadress	<input type="text"/>
e-posti aadress	<input type="text"/>
Faksinumber	<input type="text"/>
Telefoninumber	<input type="text"/>
Soovin vastust viisil	<input type="text" value="telefoni teel"/>
Soovin järgmist teavet:	<input type="text"/>

Joonis 20. Taustavärvi ja joontega paigutus andmete grupeerimiseks või eraldamiseks Maksuameti keskkonnas www.maksuamet.ee.

Matteo Penzo uurimuse tulemused on järgnevad:

- Vertikaalne paigutus — Kasutaja suudab korraka fikseerida välja ja pealkirja, mis vähendab orienteerumise aega.

- Vasakule joondatud horisontaalne paigutus — pealkirjade asetus lihtsustab väljade leidmist. Miinuseks on väljade ja nende pealkirjade vahel olev tühi ruum, mis pikendab orienteerumiseks kuluvat aega.
- Paremale joondatud horisontaalne paigutus — väljade ja nende pealkirjade vahelistes seostes orienteerumine on poole kiirem kui vasakule joondatud horisontaalse paigutuse korral.
- Tumedas trükis väljade pealkirjad — võivad aeglustada väljade täitmist. (Penzo, 2006)

Uurimuse tulemusena toimus kõige efektiivsemalt vertikaalne paigutus, mille korral on vasakule joondatud pealkirjad väljade kohal. Kui otsustatakse siiski kasutada horisontaalset paigutust, siis tuleks eelistada paremale joondamist. Tumedas trükis väljade pealkirjade kasutamine osutus orienteerumiskiirust arvestades ebavajalikuks, kuna aeglustas seda.

Luke Wroblewski soovib valida erinevaid paigutuse tüüpe lähtudes andmete eripärast.

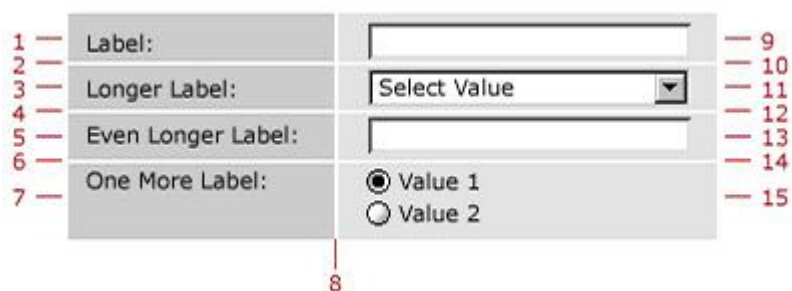
Paigutused erinevate andmete puhul:

- **vertikaalne paigutus** — andmed, mida kasutaja teab peast nagu nimi, vanus, jne. Kasutaja suudab samaaegselt seostada väljad ja nende pealkirjad ning ei pea selleks eraldi otsima. Puudusena on välja toodud vormide võimalik pikenedamine, kuna vähem horisontaalset vaba ruumi on kasutatud.
- **vasakule joondatud horisontaalne paigutus** — kasutatakse andmete korral, mis ei pruugi olla kasutajale tuttavad. Selle tüübi puhul ülevaate saamine küsimustest lüheneb ajaliselt ja väljade seostamise kiirus lahtritega pikeneb sõltuvalt nende vahel asuvast tühjast ruumist. Kasutaja saab otsida vasakust veerust sobivaid andmete pealkirju ilma, et kontrollijate tüübid ja vaikimisi väärtused teda eksitaks ning seeläbi kiirendada vormide täitmist. Eelisenä on välja toodud vormide lühenedamine kuna rohkem horisontaalset vaba ruumi on kasutatud.
- **paremale joondatud horisontaalne paigutus** — kasutatakse andmete korral, mis ei pruugi olla kasutajale tuttavad. Selle tüübi puhul väljade seostamise kiirus lahtritega väheneb ja ülevaate saamine küsimustest pikeneb ajaliselt. Eelisenä on välja toodud vormide lühenedamine, kuna rohkem horisontaalset vaba ruumi on kasutatud. Puudusena on välja toodud kasutaja lugemisharjumus

lugeda vasakule joondatud teksti, mistõttu ülevaate saamine küsimustest pikeneb ajaliselt.

- **Tumedas trükis väljade pealkirjad** – suurendavad vormides orienteerumise kiirust, kuna välistavad väljade ja nende pealkirjade visuaalse sama kaalu.
- **Visuaalsete elementide kasutamine** – erineva taustavärviga lahtrite ja nende pealkirjade eraldamine kaheks või enamaks veeruks, joontega ridade ning vormi elementide seoste rõhutamine. (Wroblewski, 2005)

Visuaalsete elementide kasutamine vormide kujundusel võib kaasa tuua ka ebavajalikus taustamüra orienteerumise. Kasutaja silmad peatuvad eraldi igal pealkirjal, taustal, joonel ja lahtril, mis lisab kasutaja jaoks juurde informatsiooni, mida arvestades tuleb tegutseda ning see omakorda võib pikendada vormide täitmisele kuluvat aega. Joonisel 19 on nummerdatud silmade peatumise punktid vormist ülevaate saamisel.



Joonis 21. Silmade peatumise punktide järjekord vormis.

Eri paigutuste uuringute läbiviimise meetoditeks oli valitud silmade liikumistee jälgimine (*eyetracking*), mille puhul saab mõõta läbitud vaatepunktide vahelisi distantse ja nende läbimise kiirust. Kuna veebivormide kasutajakesksust näitab ka nendes orienteerumise ja nende täitmise kiirus, siis on nimetatud uurimuse meetod asjakohane osutamaks uuritava probleemi kitsaskohtadele.

Vormi paigutuse tüübi valimisel tuleb arvesse võtta sihtgruppi, kogutavate andmete tüüpi, ajalimiiti ja soorituse kordade arvu. Kasutaja seisukohast lähtudes võib nende parameetrite mõju kas lihtsustada või raskendada vormidest arusaamist ja nendes orienteerumist ning kokkuvõttes mõjutada oluliselt nende täitmisele kuluvat aega.

3.3 Grupeerimine

Grupeerimise eesmärgiks on muuta vormide visuaalne pool kergesti kasutatavaks, vajalike andmete ja lahtrite kiiresti ülesleidmise ja täitmise seisukohast lähtudes. Tuginedes uurimusele, mis on läbi viidud Tom Tullis'e poolt võib väita, et täitmise aega mõõdikuks võttes pole isegi paigutuse tüüp vormide täitmisel sellise tähtsusega, kui nende grupeerimine. (Fowler, 2004, lk 70)

Grupeerimisel arvestatakse kogutavate andmete omavahelist loogilist seostatust või seostamatust, mille alusel tuleb nad kas grupeerida või üksteisest eraldada. Grupeerimiseks kasutatakse disainielementidena tabeleid, jooni, värve, tühialasid või erinevaid kombinatsioone nendest kõikidest.

Grupeerimise vajalikkust rõhutab sisutihe ja paljude kontrollijatega küsimustik, kus on vaja määrata või rõhutada gruppikuuluvust. Grupeerimine peaks looma kasutajale ettekujutuse vormi läbimise etappidest ja nende etappide arvust.

Elementide grupeerimisel lähtutakse ennekõike nende sisulisest eripärast alles seejärel vormilisest. Pikkade mitmeleheküljeliste vormide korral võib olla vajalik alamgruppide moodustamine, kus tuleb näidata seotust ülemgrupiga ja sõltuvalt tekstilisest ülesehitusest kasutada sobivat liigendust. Selleks võib kasutada sakkidega vormi ülesehitust.

3.4 Tekst, taust ja värvid

Veebivormide värvi ja taustalahenduste disainimisel peaks lähtuma põhimõttest, mille kohaselt kujundus peab olema esteetiliselt nauditav ja samas minimalistlikult kompaktne. Kujunduse kasutamine on tulemuslik ainult teadlikel otsustel põhinedes.

Vormide kasutatavus on ennekõike seotud visuaalse mõistetavusega, mida on võimalik parandada kasutades selleks elementide ja tausta vahelist põhjendatud värvivalikut, kontraste, elementide konteksti sobivat suurust ja positsiooni.

Kohustuslike väljade märkimisel ei ole soovitatav kasutada erivajadustega kasutajate huve silmas pidades:

- Ainult teist värvi kirjeid (8% meestel esineb punase ja roheline värvi pimedust)

- Tumedas trükis kirja või teist kirjastiili(ekraanilugejad ei loe kirjastiili) (Fowler, 2004, lk 81)

Värvide planeerimisel on oluline saavutada elementide ning nendega seotud tausta omavaheline tasakaal, mille korral taust ei domineeriks andmete üle. Eesmärgiks on teatavaid üksikuid elemente, nende omavahelisi grupe või gruppide seotust terviku vormiga visuaalselt rõhutada või esile tuua nende erisust ülejäänutest. Seega on igal disainiprotsessis tehtaval otsusel kindel edasiulatuv mõju, mida teadlikult järgides saab rakendada kasutajakesksuse parandamiseks.

3.5 WCAG metoodikate disainijuhised

Erivajadustega kasutajatele mõeldud standardite uurimisel on aluseks võetud viimane versioon WCAG 1.0, mis on loodud 1999. aastal.

Erivajadustega kasutajateks peetakse inimesi: kellel on raskusi teksti lugemise või sellest arusaamisega, kes ei saa kasutada klaviatuuri või hiirt, kellel on tekstipõhised või väiksed ekraanid, aeglane interneti ühendus, kes ei räägi ja ei saa aru keelest millega sisu on kirjutatud, kelle silmad, kõrvad või käed on hõivatud teiste tegevustega, kellel on vanem lehitseja versioon või häällehitseja, teine operatsiooni süsteem. Seega WCAG meetodite järgimine mõjutab väga tugevalt kasutajakesksust ning saadav kasu laieneb erivajadustega kasutajatelt ka tavakasutajatele. Seega WCAG meetoditega kooskõlastatus tagab ligipääsu ja kasutatavuse erivajadustega sihgruupile. Erineva tähtsusega juhised on jagatud prioriteetideks tähtsuse järjekorras:

- Prioriteet 1 — Veebiarendaja peab selle punktiga arvestama. Vastasel juhul on võimatu leheküljel olevale infole ligi pääseda.
- Prioriteet 2 — Veebiarendaja peaks selle punktiga arvestama. Vastasel juhul on ühel või enamal erivajadustega kasutajategrupil keeruline leheküljel olevale infole ligi pääseda.
- Prioriteet 3 — Veebiarendaja võiks selle punktiga arvestada. Vastasel juhul on ühel või enamal erivajadustega kasutajategrupil mõnevõrra keeruline leheküljel olevale infole ligi pääseda.

Veebilehekülgede standardile vastavust on võimalik kontrollida aadressil <http://www.w3.org/TR/WCAG10/full-checklist.html>, kus saab märkida erinevate prioriteetide all olevatele küsimustele vastused ja nii määrata vormi valiidsust. Kergem on aga võtta vastavad spetsifikatsioonid arvesse juba koodi kirjutamise etapis. Samuti on võimalik kontrollida vastavust WCAG metoodikatele ka automaatsete vahenditega, millena W3C pakub välja WebXACT'i aadressil (<http://webxact.watchfire.com/>).

Veebivormidega seotud disainijuhiseid käsitleb *HTML Techniques for Web Content Accessibility Guidelines 1.0*, mis kuulub WCAG 1.0 tehniliste dokumentide alla.

Arvestades uurimuse konteksti on vajalik selgitada veebivormidega seotud võimalikke probleeme, mis võivad mõjutada erivajadustega kasutajaid. Alljärgnevalt on esitatud väljavõtte veebivormidega seonduvatest probleemidest prioriteetide järjekorras ning lisatud selgitus.

1.1 Võimalda teksti väärtust `alt` ja `longdesc` elementidele. See hõlmab pilte, freime, skripte, pilte kasutatuna nimekirjade tärnidena, graafilisi nuppe, helisid, videosid. [Prioriteet 1]

```
<form action="http://example.com/prog/text-read"
method="post">
<input type="image" name=submit src="button.gif" alt="Submit"
/>
</form>
```

Erinevate piltide kasutamine nuppude kujundusel võib muuta vormid raskesti ligipääsetavaks kui pole kasutatud teksti vastet, mida mittegraafilised brauserid tõlgendada ei oska. Kui mingil põhjusel on pildifail brauseri poolt kättesaamatu, siis on võimalik ekraanilt aru saada, kus pilt peaks asuma.

12.3 Kus on loogiline ja võimalik võiks jagada suured info blokid hallatavamateks väiksemateks gruppideks. [Prioriteet 2]

```
<form action="http://example.com/adduser" method="post">
<fieldset>
<legend>Isikuandmed</legend>
<label for="firstname">Eesnimi: </label>
<input type="text" id="firstname" tabindex="1" />
<label for="lastname">Perekonna nimi: </label>
<input type="text" id="lastname" tabindex="2" />
</fieldset>
<fieldset>
<legend>Muu info</legend>
...vajaminev teave...
</fieldset>
</form>
```

Vormi elementide grupeerimine loogilisteks osadeks tuleb kasutada `fieldset` elementi ja legend elemendiga märkida pealkirjad.

12.4 Seosta väljade pealkirjad nende kontrollijatega. [Prioriteet 2]

```
<label for="firstname">Eesnimi:  
  <input type="text" id="firstname" tabindex="1" />  
</label>
```

Pealkirjad on vaikumisi alati seotud kontrollijatega. Oluline aga on määrata nendevaheline seostatus HTML-keele abil.

10.2 Veendu, et välja pealkiri on õigesti paigutatud ja asub õiges kohas. [Prioriteet 2]

Erinevad brauserid tõlgendavad html keelt erinevalt, seetõttu võivad esineda mõningad kujunduse erisused. Soovitav oleks määrata asukoht elementidega, mitte lasta brauseril tõlgendada.

10.4 Määra vaikumisi väärtus kasutajaagendile. [Prioriteet 3]

```
<form action="http://example.com/prog/text-read"  
method="post">  
  <p>  
    <textarea name="yourname" rows="20" cols="80">  
    Palun sisesta siia oma iseloomustus.  
    </textarea>  
    <br />  
    <input type="submit" value="Saada" />  
    <input type="reset" value="Tühjenda" />  
  </p>  
</form>
```

Vaikumisi väärtus aitab mittegraafilisel kasutajaagendil vorme ja nende täitmise põhimõtteid selgemalt esitada.

9.4 Loo loogiline järjekord linkide, kontrollijate ja objektide jaoks. [Prioriteet 3]

```
<form action="submit" method="post">  
  <input tabindex="2" type="text" name="field1" />  
  <input tabindex="1" type="text" name="field2" />  
  <input tabindex="3" type="submit" name="submit" />  
</form>
```

Erinevate paigutuste, grupeerimise ja elementide järjekorra puhul peab säilima nende läbimise järjekord ka *tab* klahvi kasutades.

9.5 Võimalda klaviatuurilt ligipääsu tähtsatele kontrollijatele ja kontrollijate gruppidele. [Prioriteet 3]

```
<form action="submit" method="post">  
<label for="user" accesskey="U">Nimi</label>  
<input type="text" id="user" />  
</form>
```

Ligipääsu saab aktiveerida võtme, milleks on näites määratud „U“, vajutusega.

Ligipääsuklahviga saab muuta raadionupu väärtust, tekstiväli lubab sisestada andmeid, lingi puhul järgib kasutajaagent aadressi. (W3C, 2000)

4. Veebivormide evalvatsiooni meetodid

Evalveerimisel kogutakse andmeid valmis toodete, prototüüpide, uurimise käigus loodud mudelite kasutatavuse kohta sihtgrupilt, asetades testijad reaalses elu ettetulevatesse olukordadesse või keskkondadesse.

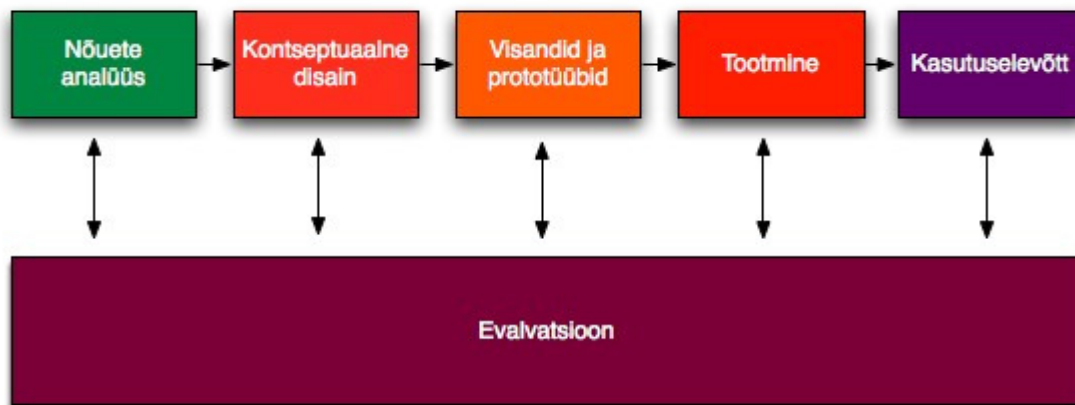
Evalveerimise käigus mõistetakse kuidas kasutaja erinevates olukordades käitub ja reageerib ning neid teadmisi kogu vormide loomise protsessis aluseks võttes saab potentsiaalseid tõrkeid ennetada või kõrvaldada ning seeläbi tagada vormide efektiivsema kasutatavuse. Ilma evalveerimiseta on raske hinnata veebivormide kasutatavust konkreetse sihtgrupi hulgas.

Evalveerimise vajadus on määratletud nelja põhjusega:

- Reaalse maailma mõistmine;
- Võrdlev disain;
- Eesmärgipõhine rakendamine;
- Kontrollida vastavust standarditele. (Preece et al., 1994)

Evalveerimisega peaks alustama juba veebivormide planeerimise etapis, kus toimub ideede ja eesmärkide kaardistamine ning kestma läbi erinevate etappide kuni valmis tooteni. Erinevatel disainietappidel tehtava evalvatsiooni tagasiside aitab kokku hoida aega ja kulusid ning tagab lõpptulemuse vastavuse eesmärgile.

Evalvatsiooni osatähtsust ja seotust erinevate disainietappidega näitab Brinck'i, Gergle'i ja Wood'i kõikehõlmav kasutatavuse protsessi mudel (*Pervasive Usability Model*). (Brinck, 2002) Antud mudeli järgi toimubki evalvatsioon paralleelselt kõikide nimetatud disainiprotsessidega. Samuti võimaldab mudel valida lähtuvalt eesmärgist ja vajadusest evalvatsiooni kohandamiseks sobiv etapp: kas teostada see pralleelselt või spiraalselt kõigi tegevustega või jätta mõne etapi puhul vahele.



Joonis 22: Kasutajaliidese loomise mudel. Brinck, Gergle, Wood, 2002.

Evalveerimise üldiseks eesmärgiks veebivormide kontekstis on kasutatavuse parandamine, kõrvaldades arusaamisel, navigeerimisel ja täitmisel ettetulevad võimalikud takistused.

Selleks tuleb:

- Võtta aluseks veebivormide koostamise kriteeriumid, milleks on arusaamine, suhtlus, kasutamine ja visuaalne kasutatavus.
- Määratleda sihgrupp ja sihtgrupi oskused, vajadused ning ootused.
- Määratleda evalveerimise läbiviimise tingimused
- Määrata evalveerimise hindamise parameetrid, milleks on teooria põhjal koostatud veebivormide evalveerimise komplekt.

Oluline on teha kasutaja jaoks veebivormide konseptsioon mõistetavaks, lihtsaks ja loogiliseks. Evalveerimise kõige olulisemaks etapiks on välja selgitada küsimused ja probleemid, millele oleks vaja lahendused leida.

Valitud evalvatsioonitüübist sõltumata on evalveerimisel oluline pöörata tähelepanu:

- kasutajate karakteristikutele nagu kogemused, vanus, sugu, psühholoogilised ja füsioloogilised näitajad;

- Tegevuste ja võimalike toimingute tüübid, mida kasutaja peab tegema või teeb, mis võivad olla selgelt defineeritud ja kontrollitavad evalveeriya poolt või tegevused mille üle otsustab kasutaja ise;
- Uurimuse keskkond, milleks võib olla kontrollitud situatsioon laboratooriumis mingi konkreetse ülesande kohta;
- Artefakti olemus, milleks võib olla sari sketse, töötav tarkvara, prototüüp või valmisarendatud toode; (Preece, 1994, lk 602)

Seega on äärmiselt oluline evalvatsiooni ettevalmistus ja planeerimine ning mõõdetavate tingimuste määramine, mida saaks edaspidi disainiprotsessis või muudatuste sisseviimisel aluseks võtta.

4.1 Evalveerimise meetodid

Sõltuvalt uurimisvajaduse olemusest tuleb valida sobiv evalvatsiooni tüüp, mis kataks kõiki vajalikke probeeme, ei oleks liiga pealiskaudne ega ebavajalikult detailne. Samuti tuleb arvestada olemasolevaid ressursse nii testijate kui testimiseks vajaminevate vahendite suhtes. Kokkuvõttes peab evalveerimine olema põhjendatud mahus, piisavalt spetsiifiline ning põhjendatud kulutustega. Alljärgnevalt on toodud võrdlus erinevate kasutatavuse evalveerimise meetodite UEMs (*Usability Evaluation Methods*) kohta. (Axup, 1998) Tabelis 7 esitatud kasutatavuse evalveerimise meetodid on selguse huvides jäetud eesti keelde tõlkimata.

Tabel 7. Kasutatavuse evalveerimise meetodid. (Axup, 1998)

Kasutatavuse evalveerimise mudelid	Kirjeldus ja ülevaade	Eelised	Puudused
Think Aloud User Testing	Kasutaja töötamist tootega filmitakse või salvestatakse ja analüüsitakse. Kasutatakse stsenaariumi või ülesannet, erineva taustaga inimesi, kes peavad mõtlema valjusti. Suhtlus kasutaja ja läbiviija vahel on minimaalne. Vaja on ühte inimest.	Väga lähedane hinnang tegelikule kasutusele. Vaja on vähe testijaid, seega on odavam.	Instruktsioon mõtle valjusti võib töötada halvasti kuna on kasutajale ebaloomulik.

Constructive Interaction	Kahe kasutaja suhtlust süsteemiga filmitakse/salvestatakse ja analüüsitakse. Üks kasutaja juhib kasutamist. Kasutatakse stsenaariumi või ülesannet, mida lahendatakse koostöös. Kasutatakse erineva taustaga inimesi, kes peavad mõtlema valjusti. Suhtlus kasutaja ja läbiviija vahel on minimaalne.	Pakub suuremat verbaalset tagasisidet, seetõttu võib peegeldada reaalselt maailma. Võib leida sarnaseid tähelepanekuid nagu 1 kasutajaga uuring.	Rohkem testijaid, seega on kallim. Hindab koostöös probleemide lahendamist umbkaudu.
Focus Groups	Kasutatakse konseptuaalse disaini etapis. Kasutajatelt küsitakse arvamusi toodete kohta, kasutades selleks stsenaariume või demosid. Kasutajateks on grupp inimesi.	Pakub palju võimalikke, ka uusi disainilahendusi ja võib parandada kliendisuhteid.	Algselt turunduses loodud lahendus. Pole teaduslik ega läbi viidud kontrollitud keskkonnas. Ei testi kasutaja tegelikku süsteemi kasutust vaid salvestab andmeid selle kohta kuidas kasutajad arvavad, et nad käituvad mitte kuidas nad reaalselt käituvad.
Expert Review	Peab olema piisav kogemus või vastav kraad. Toodet analüüsitakse paranduste sisseviimiseks eraldi või gruppidega. Pole juhendit, stsenaariume, ülesannet. Reeglina kasutatakse eksperte.	Kasutatakse eksperte. Võib katta kogu probleemide ala	Muudatusi disainis võib olla keeruline ratsionaalselt standardiseerida või kategoriseerida.
Expert Walkthrough	HCI eksperdid kasutavad ülesandeid ja stsenaariume, et juhtida kasutajaliidese analüüsi.	Kasutatakse eksperte. Võib keskenduda juba teadaolevale probleemidele.	Sisemine kallutatus ülesande valikul. Ei pruugi katta kogu probleemide ala.
Heuristic Evaluation	HCI eksperdid vaatavad kasutajaliidese eraldi üle ja kategoriseerivad need heuristikute(põhireeglite) komplektina. Kasutatakse eksperte, lühikesi juhendeid, ülesandeid. Stsenaariume ei kasutata.	Kiiresti teostatav. Kasutatakse eksperte. Sisaldab mitme kriitiku tagasisidet.	J. Nielsen juhendite valiidsuses on kaheldud ja eksisteerivad alternatiivsed juhendid.
Heuristic Walkthrough	Sarnaneb Heuristilisele evalvatsioonile. Kasutatakse eksperte, lühikesi juhendeid, ülesandeid ja stsenaariume.	Kasutatakse eksperte. Võib keskenduda juba teadaolevatele probleemidele.	Sisemine kallutatus ülesande valikul.

Guidelines	Juhendid ulatuvad sadadest kuni tuhandete lehekülgedeni. Kontrollib, kas kasutajaliides vastab nendele juhenditele.	Võib olla odavam, kuna teostavad mitteeksperdid.	Ajakulukas ja tüütu, kuna juhendid võivad olla omavahel konfliktis, väga pikad või kallutatud.
Guidelines Walkthrough	Kasutatakse pikki juhendeid, lisaks ülesandeid ja stsenaariume, et juhtida analüüsi.	On odavam kuna teostavad mitteeksperdid. Võib keskenduda juba teadaolevatele robleemidele.	Ajakulukas ja tüütu, kuna juhendid võivad olla omavahel konfliktis või kallutatud. Sisemine kallutatus ülesande valikul. Ei kata kogu probleemiala.
Cognitive Walkthrough	Võetakse täielikult kasutusse ülesanded/stsenaariumid, fookuseerides kasutaja kognitiivsetele protsessidele ja tajudele ning mudelitele millega juhitakse analüüsi.	Fookuses on kasutaja. Keskendub teadaolevatele probleemi aladele.	Tüütu, kuna püüab teha disainerist kasutajat. Sisemine kallutatus ülesande valikul. Ei kata kogu probleemiala. Pole päris kasutajaid.

Veebivormide evalveerimiseks sobiva meetodi valikul tuleb arvestada olemasolevaid ressursse nii testijate kui testimiseks vajaminevate vahendite suhtes. Sellest tulenevalt ei sobi veebivormide evalveerimiseks antud töö kontekstis erinevad eksperte vajavad meetodid (Expert Review, Expert Walkthrough, Heuristic Evaluation, Heuristic Walkthrough), kuna Eestis ei ole veebivormide kasutatavusega sellisel tasemel kursis olevaid spetsialiste, kes võiksid taolist uurimust teostada ning lisaks oleks see ka liiga kulukas. Kuna veebivormidega töötamisel omab suurt tähtsust just tagasiside realselt kasutajalt siis ei sobi ainult erinevate juhendite ja meetodite (Guidelines, Guidelines Walkthrough, Cognitive Walkthrough) alusel testimine. Enamus juhendeid ja spetsifikatsioone ei ole töötatud välja koostöös kasutajatega ja seetõttu jääb sellise meetodi puhul puudu kasutaja ja veebivormi interaktsioonide jälgimise võimalustest. Fookusgruppidega testimine ei aseta kasutajaid samuti reaalsesse olukordadesse ning seetõttu ei sobi antud töö kontekstis rakendamiseks. Kuna veebivormide täitmist tuleb reeglina täita individuaalselt, siis on eriti oluline kuidas kasutaja sellega üksi hakkama saab, ilma et abi saamiseks kellegi poole pöörduks, mistõttu tuleks välistada ka kahe kasutajaga testimise meetodi sobivus (Constructive Interaction).

Think Aloud meetodiga kasutaja testimisel on mitmeid väga vajalikke võimalusi. Kõige tähtsam evalveerimise juures on saada võimalikult adekvaatset tagasisidet, mis vastaks reaalsele tekkida võivatele olukordadele. Evalveerimisel on oluline kasutaja otsene

tagasiside, kasutaja asetamine reaalsesse olukordadesse ja ühe kasutaja veebivormidega töötamise jälgimine. Seda kõike on võimalik teostada Think Aloud meetodiga. Oluliseks osutub veel väheste testijate vajadus, mis muudab protsessi vähem ressursi ning ajakulukaks. Seega arvestades veebivormide evalveerimise eripärasid ja olemasolevaid ressursse, töö eesmärki ning mahtu, oleks kõige sobivam kasutatavuse evalveerimise meetod Think Aloud kasutajatestimine.

4.2 Veebivormide evalvatsioon Think Aloud meetodiga

Enne kasutajate testimise teostamist on vaja määrata testimise eesmärk, mida saavutada tahetakse, tingimused ja parameetrid, mida ja kuidas mõõta ning eesmärgi saavutamiseks vajaminevad tegevused.

Testimise eesmärgiks on kontrollida koostatud evalvatsioonimeetodite komplekti kasutatavust veebivormide hindamisel õpiobjektide loomiseks mõeldud keskkonnas LeMill.

Testimise parameetrid näitavad, mida testimise käigus mõõtma hakatakse ja tingimused, kuidas evalvatsiooni teostatakse. Evalvatsiooni parameetriteks on järgmises peatükis põhjalikumalt käsitletav evalvatsioonimeetodite komplekt, mille koostamisel lähtutakse magistritöös väljatoodud kasutatavuse probleemidest seoses veebivormidega.

Evalvatsiooni tingimused on määratud valitud kasutajatestimise meetodi Think Aloud läbiviimise põhimõtetest lähtudes. Kokkuvõtliku ülevaate andmiseks on evalveerimiseks vajaminev teave edastatud alljärgneva tabelina.

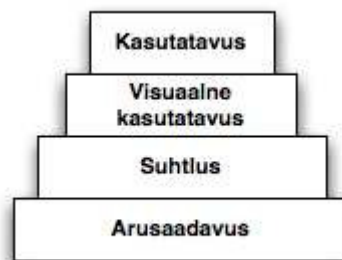
Tabel 8. Andmed Think Aloud kasutajatestimise kaardistamiseks.

Andmed evalvatsiooni kaardistamiseks	Veebivormide evalveerimine LeMill keskkonnas
Vajamineva info	Kasutaja reaktsioon, käitumine, valjusti mõtlemisel saadud teave.
Süsteemi eripära	Õpiobjektide loomise veebivormid
Kasujaliidese hindamise parameetrid	Evalvatsioonimeetodite komplekti kontroll-loend hinnatavatest probleemidest.
Evalveeritav disainitsükli etapp	kasutuselevõtt
Kogutavate andmete tüüp	Kvalitatiivne
Ressursside olemasolu	Testijad, arvuti, videokaamera, läbiviijad

Evalveermine Think aloud meetodiga viiakse läbi ühe kasutajaga, kellel palutakse sooritada etteantud tegevused valmisolevate abimaterjalide ja failidega samal ajal oma tegevusi kommenteerides. Evalveerimiseks on valitud LeMill Multimeedia lehekülje koostamise veebivorm. Enne testimise läbiviimist tutvustatakse testijale vajamineval määral sooritavate toiminguid ning läbiviimise põhimõtteid. Samuti tuleb selgitada, et testija eesmärgiks on oma käitumise ja mõtlemise võimalikult tõetruu edastamine. Kogu testimise aja vältel kasutaja tegevust filmitakse. Mõttepauside tekkimisel suunab läbiviija testijat oma mõtteid välja ütleva ja tegevusi kirjeldama, ilma oma isiklikku arvamust avaldamata. Salvestatud andmed töödeldakse ja analüüsitakse koos automaatsete valideerimise ja kontroll-loendi tulemustega. Kogutud andmed ja tulemused kokku moodustavad veebivormide evalvatsioonimeetodite komplekti kasutamisel saadud tagasiside, mida arvesse võttes saab veebivormide kasutatavust tõsta.

4.3 Evalvatsioonimeetodite kompleks

Evalvatsioonimeetodite kompleks koosneb kolmest osast: automaatselt läbiviidavatest testidest, kontroll-loendist ja kasutajaga testimisest. Evalvatsioonimeetodite komplektille tuginedes on võimalik koostada veeraporteid, kus on kompleks esinenud vigadest. Automaatselt läbiviidavad testid hõlmavad HTML spetsifikatsioonide ja WCAG ja meetodikate valideerimist. Kontroll-loend sisaldab teooria põhjal koostatud probleemide nimistut, mida veebivormide koostamisel peaks arvesse võtma. Kasutajatestimine hõlmab Think Aloud meetodiga LeMill õpiobjektide loomise keskkonnas veebivormide testimist. Tavaliselt on erinevate soovitude ja põhimõtete komplektid koostatud lähtudes programmeerija vaatenurgast ja eelistustest, mis ei pruugi olla kasutatavuse seisukohast kõige otstarbekam, kuna veebivormide täitmisega puutuvad kokku ka väga väheste arvutikasutuse oskustega kasutajad. Seetõttu on antud magistritöös aluseks võetud kriteeriumide raamistik koostatud lähtuvalt kasutaja vaatenurgast. See aitab veebivormide koostajal eristada olulist ebaolulisest ning mõista kasutatavuse tagamisega seotud moodsikuid. Kontroll-loend evalvatsioonimeetodite kompleksis on jagatud nelja kriteeriumi alusel arusaamise, suhtluse, visuaalse kasutatavuse ja kasutamise tasandiks.



Joonis 23. Evalvatsioonimeetodite komplekti kuuluva kontroll-loendi ülesehitus

Kontroll-loend

I Arusaamine: kas kasutaja mõistab vormide arhitektuuri ja navigeerimist?

- Kas navigeerimise skeem on arusaadav?
- Kas kasutaja saab aru oma asukohast vormis liikudes?
- Kas kasutaja leiab vajamineva info?
- Kas navigeerimisribal on piisavalt valikuid?
- Kas navigeerimisriba valikud on loogilises järjekorras?
- Kas nupud on ülesleitavad?
- Kas nupud on arusaadavad?
- Kas lahtrititel on pealkirjad olemas?
- Kas lahtrite pealkirjad on arusaadavad, ühemõttelised?
- Kas vormi elementidega toimetamine on arusaadav?
- Kas edasi-tagasi liikumise nupud on olemas?
- Kas edasi-tagasi liikumise nuppude toimimise põhimõtted on arusaadavad?
- Kas vormide ülesehituse struktuur on eraldi välja toodud või lahti seletatud?
- Kas erinevatel lehekülgedel liikudes on selge, millisel asutakse?
- Kas vormide struktuur on loogiline ja tuttavlik?
- Kas esitatud küsimuste järjekord on loogiline ?
- Kas kasutaja eksib vormides navigeerimisel?
- Kas kasutaja saab kasutada oma eelnevaid kogemusi ja teadmisi vormidega töötamisel?
- Kas kohustuslikud väljad on mõistetavalt märgistatud?
- Kas tekst on ühemõtteline?
- Kas kujunduse komponendid toetavad eesmärki?
- Kas kuupäev ja kellaaeg on rahvusvaheliselt arusaadavad?
- Kas telefoninumber on rahvusvaheliselt arusaadav?
- Kas aadress ja postindeks on rahvusvaheliselt arusaadavad?

II Suhtlus: kas kasutaja saab veebivormidega tõrgeteta vajamineval määral suhelda.

- Kas sissejuhatav tekst on olemas?
- Kas võimalus otsida abiinfot on olemas?
- Kas veateated on arusaadavad?
- Kas veateade pakub välja lahenduse, millest on kasu?
- Kas tagasiside on olemas?
- Kas pakutakse hiirega üleliikumise tagasisidet?
- Kas hiirega üleliikumise tagasiside oleks vajalik?
- Kas enne riskantsete või tähtsate tegevuste sooritamist küsitakse kasutajalt kinnitust?
- Kas vastuste saamiseks on võimalik kasutada otsingut?

III Kasutamine: kas kasutaja saab veebivormide kasutamisega hakkama?

- Kas vormielemente on õigesti kasutatud?
- Kas pildid sisaldavad alt väärtust?
- Kas pilte salvestatakse võimalikult suure kompressiooniga?
- Kas kasutajad peavad meelde jätma infot erinevatelt lehekülgedelt?
- Kas riskantseid või tähtsaid tegevusi on võimalik tühistada või muuta?
- Kas sisestusvigade vältimine on reguleeritud?
- Kas eeltäidetud välju on kasutatud põhjendatud kohtades?
- Kas lahtrid ja tekstialasid on kasutatud vastavalt nende eesmärgile?
- Kas ripploendeid ja ripploendikaste on kasutatud vastavalt nende eesmärgile?
- Kas raadionuppe ja märkeruute on kasutatud vastavalt nende eesmärgile?
- Kas nupud toimivad vastavalt nende kasutamise põhimõtetele?
- Kas esineb graafika/mm elemente mis ei avane?
- Kas vormid avanevad enamkasutatavates brauserites?
- Kas vormid on loetavad/täidetavad häälbrauserit kasutades?

IV Visuaalne kasutamiskihtsus: kas veebivormide visuaalne väljanägemine lihtsustab kasutamist?

- Kas vormide joondus on ühtlane?
- Kas küsimused on grupeeritud?
- Kas ühe vormi piires kasutatakse sama paigutuse stiili?
- Kas väljade ja pealkirjade vaheline kaugus on mõistlik?
- Kas ekraanipilti peab horisontaalselt kerima vormi nägemiseks?
- Kas ekraanipilti peab vertikaalselt kerima vormi nägemiseks?
- Kas paigutusel on üks teema ühel leheküljel?
- Kas on arvestatud printimisega vormide pikkuse määramisel ühele ekraanile?
- Kas värvide valikul on tekstil sobiv kontrastsus?
- Kas mingid värvid või disainilahendused domineerivad teiste üle?
- Kas värvide valik on sobiv?
- Kas värvide valik toimib ka mustvalgel ekraanil?
- Kas tekst on loetavas kirjas?
- Kas tekst on loetavas suuruses?
- Kas tekst on loetavas värvis?
- Kas vaba ruumi on põhjendatud mahus?
- Kas lehekülje servadel on tühi ääreala?
- Kas esineb kirja, stiilivigu?

5. LeMill õpiobjektide loomise keskkonna veebivormide evalvatsioon

Evalvatsioonimeetodite komplekti katsetatakse LeMill keskkonna veebivormidega (vt. <http://lemill.net>). LeMill on Euroopa Liidu 6. raamprogrammi IST projekti CALIBRATE (vt. <http://calibrate.eun.org>) raames arendatav õppematerjalide loomise keskkond, mille sihtgrupiks on üldhariduskoolide õpetajad.

Evalvatsioonimeetodite katsetamiseks on valitud LeMill keskkonna õpiobjektide mallid, mis võimaldavad kasutajatel luua kolme eri tüüpi veebipõhiseid õpiobjekte.

- Multimeedia lehekülg (teksti, pilte, heli-ja videoklippe sisaldav veebilehekülg)
- Esitlus (esitlusslaide, kommentaare ja esitluse helisalvestust sisaldav veebilehekülg)
- PILOT (mingit konkreetset teemat tutvustav Flash animatsioon koos küsimustega)

LeMill keskkonnas olevad veebivormid on oma ülesehituselt ja eesmärgilt erineva raskusastmega, sisaldades nii pildi kui multimeedia elementidega toimetamise võimalusi. Keskkonna sihtgrupp on väga erineva arvutikasutuse oskusega ning kasutajad ei pruugi eelnevalt olla sarnaste keskkondadega töötamisega tuttavad. Kuna LeMill keskkonnal on kasutajaid paljudest eri maadest, siis on oluline, et vormide disain oleks üheselt arusaadav ja nendega toimetamine võimalik sõltumata rahvusest ja kultuuritaustast. Testitud veebivormide aadressid ja kirjeldused on esitatud tabelis 9.

Tabel 9. Valideeritavate lehekülgede andmed

Veebivorm	Kirjeldus	Analüüsitud lehekülgede arv
Multimeedia lehekülg	Leheküljel on võimalik luua pilte, heli-ja videoklippe sisaldav materjal.	3
Esitlus	Leheküljel on võimalik luua helisalvestusega slaide sisaldav materjal.	4
PILOT	Leheküljel on võimalik luua Flashi animatsioone sisaldav materjal.	4

5.1 Automaatsete vahenditega testimise tulemused

HTML ja WCAG validaatoriga testiti kolme veebivormi LeMill keskkonnast: multimeedia lehekülje koostamise vormi, esitluse koostamise vormi ja Flashi animatsioonide koostamise vormi. Nii multimeedia, esitluse, kui ka PILOT lehekülje vormide valideerimisel HTML validaatoriga esinesid samalaadsed vead.

Kuna magistritöö eesmärk ei ole vigade parandamine, vaid evalvatsioonimeetodite komplekti katsetamine, siis ei keskenduta töös vigade selgitusele. Sellest tulenevalt on järgnevalt välja toodud esinenud vigade liigid tabelina, mis annab ülevaate nende arvust ja tüüpidest. Valideerimisel esinenud vigade arvu ei saa võtta üheseltmõistatavana, kuna paljud neist on omavahel seotud selliselt, et kõrvaldades ühe neist, mõjutab see vigade koguarvu vähenemist suuremal hulgal. Kuigi tabeli kirjelduse veerus sulgudesse pandud väärtused erinevad sõltuvalt lehekülje koodist, on siiski tegemist sama tüüpi vigadega.

Tabel 10. Valideerimise tulemused HTML validaatoriga kontrollimisel

Viga	Kirjeldus	Vigade arv
<html xmlns="http://www.w3.org /1999/xhtml" xml:lang=""	Syntax of attribute value does not conform to declared value	22
value="Philosophy" type="checkbox" id="subject_area_1" />	ID "(subject_area_1)" already defined	108
id="subject_area_1" />	ID "(subject_area_1)" first defined here	108
<label for="subject_area">Ainev aldkond</label>	reference to non-existent ID "(subject_area)"	30
<xml id="dataisland" style="display:none">	there is no attribute "(id)"	10
<xml id="dataisland" style="display:none">	element "(xml)" undefined.	23
..emill.net/content/lis a.jpg/image_large" />	required attribute "(alt)" not specified.	12

Järgnevalt on näidisenä esitatud veateadete selgitus, kuidas toimiks kirjelduse lahtiseletamine parandamise etapis.

1. Error *Line 5, column 53*: syntax of attribute value does not conform to declared value .

```
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang=""
```

2. Error *Line 6, column 12*: syntax of attribute value does not conform to declared value .

```
lang="">
```

Atribuuti `xml:lang` kasutatakse keele määramiseks, mida sisus kasutatud on. Kui atribuuti on kasutatud, siis peab olema määratud ka selle väärtus. Näiteks määrates väärtuseks `<p xml:lang="en">English</p>` või `<p xml:lang="et">Eesti keel</p>`. Tühja `xml:lang` atribuuti kasutatakse, et üle kirjutada tema eelmised väärtused, ilma et peaks selleks uue keele määrama.

Järgnev viga on seotud `id=""` ja `label for=""` väärtuste määramisega, mis peavad vastama kahele tingimusele: olema unikaalsed ja omavahel sobivuse väärtustega nagu `<input id="subject_area_2" />` ja `<label for="subject_area_2">`.

3. Error *Line 724 column 43*: ID "subject_area_1" already defined.

```
value="Philosophy" type="checkbox" id="subject_area_1" />
```

4. Info *Line 605 column 8*: ID "subject_area_1" first defined here.

```
id="subject_area_1" />
```

Id atribuutide väärtused peavad olema erinevad ja unikaalsed iga kord kui neid kasutatakse. Kogu metaandmete lehekülje ulatuses esineb märkeruutude koodides samasugune viga, elemendi `input`, `id` ja `label for` väärtusi on kasutatud rohkem kui ühel korral.

Järgnev viga on seotud kas mitte eksisteeriva `input`, `select` või `textarea` elemendiga, puuduva `id` atribuudiga või trükiveaga.

5. *Line 588 column 22*: reference to non-existent ID "subject_area".

```
<label for="subject_area">Ainevaldkond</label>
```

Antud juhul on tegemist puuduva `input` elemendiga, kuna kasutatud on `label for` elementi märkeruutude grupile pealkirja andmiseks.

WCAG spetsifikatsioonidele vastavuse automaatseks kontrollimiseks pakutavaid vahendeid on väga erinevaid, sõltuvalt vajadusest ja eesmärgist. Antud töös on kasutatud WCAG spetsifikatsioonidele vastavuse kontrollimiseks validaatorit A-Checker (aadressil <http://checker.atrc.utoronto.ca/index.html>), mis võtab võrdlemisel arvesse WCAG 1.0 versiooni.

Tulemused „Multimeedia lehekülg“, „Esitluse“ ja „PILOT“ veebivormide valideerimisel olid sarnased. Kõigil kolmel leheküljel esinenud vead on välja toodud tabelis nr. 11.

Koodis esinevate vigade hulk võib olla mõjutatud kasutaja toimingutest nagu meediafailide üleslaadimine, mille puhul puuduvate `alt` väärtuste arv on sõltuvuses üleslaetavate failide arvust. Vigade arvu peaks proovima viia miinimumini, võttes arvesse, et ka üks viga võib raskendada erivajadustega kasutajate ligipääsu.

Tabel 11. Valideerimise tulemused WCAG validaatoriga kontrollimisel

Viga	Kirjeldus	Vigade arv
<code><table...summary="Header" width="760"/></code>	Layout table has a summary. All layout tables have an empty summary attribute or no summary attribute.	8
<code><address>< /address></code>	Content missing address of page author. Content must have an address for author.	3
<code><html xmlns="http://www.w3.org/1999/xhtml" xml:lang="" lang=""></code>	Document language not identified. Document has required <code>lang</code> attribute(s)	3
<code><fieldset> <legend>...</legend> </fieldset></code>	Form missing fieldset and legend to group multiple radio buttons/checkboxes. All radio button and checkbox groups are marked using fieldset and legend elements.	3
<code><input tabindex="1" type="text" name="field1"></code>	Input element, type of "text", missing a tab index.	3
<code><input tabindex="1" type="radio" name="field1"></code>	Input element, type of "radio", missing a tab index.	12
<code><input name="new_group_name" type="text" value="" /></code>	Input element, type of "text", missing an associated label. The <code>input</code> element has an <code>id</code> attribute value that matches the <code>for</code> attribute value of a <code>label</code> element.	12

<pre><input id="subject_area_1" name="subject_area:1 ist" type="checkbox" value="Natural sciences"/></pre>	<p>Input element, type of "checkbox", missing a tab index. All input elements, type of "checkbox", have a valid tab index.</p>	60
<pre></pre>	<p>Image has suspicious Alt text (empty string ""). Non-Decorative or decorative images must have Alt text.</p>	9
<pre><select id="language" name="language" tabindex="0"/></pre>	<p>Select element's label is not positioned close to control.</p>	1
<pre><TABLE align="center" border="0" cellpadding="0" cellspacing="0" id="tb-portal- header" style="margin-top: 25px;" summary="Header" width="760"/></pre>	<p>Content may not be readable when stylesheets are removed. Remove or disable any style attributes and view the page.</p>	4
<pre><SCRIPT src="http://lemill.net/portal_javascripts/LeMill%20Tables/ploneScripts9541.js" type="text/javascript"> </SCRIPT></pre>	<p>Content possibly not readable when stylesheets are removed (script). The content must be readable when SCRIPTs are not applied</p>	17
<pre><INPUT class="context" name="form.button.editChapter" onclick="javascript:document.getElementById('bodyText_edited').value=0" type="submit" value="Muuda"/></pre>	<p>Script not keyboard accessible - onclick missing onkeypress. All onclick event handlers have an associated onkeypress event handler</p>	45

Kõige rohkem esines vigu seoses `tabindex` atribuudi ja tema väärtuste puudumisega. See atribuut määrab klaviatuurilt ligipääsu järjekorra. Kuna erivajadustega kasutajatele peaks olema tagatud ligipääs, siis võiks olla ka võimalus vorme täita ainult klaviatuuri kasutades.

Teise suure vigade grupi moodustasid javascripti kasutamisest tulenevad ligipääsetavuse probleemid. Validaatori järgselt ei ole ekraanivormid skripti tõttu klaviatuurilt ligipääsetavad. Näiteks kui rippmenüüdes kasutada `onChange` sündmusetöötlevat, mis reageerib hiirevajutusele ja kasutaja liigub menüüs nooleklahvidega, siis teostab javascript iga uue vajutusega `onChange` sünduse ning kasutaja suunatakse uuele leheküljele.

Kolmanda vigade grupi moodustasid puuduvad alt väärtused. Sõltuvalt veebivormide eesmärgist ja eripärast võib taoline viga osutada äärmiselt tõsiseks. Kuna LeMill vormide eesmärgiks on pakkuda paljude ja erinevate meediafailide ülesalaadimist siis tekstipõhised brauserid ei saa neid tõlgendada.

Kokkuvõttes ei saa ükski programm kontrollida, kas veebilehekülg on ligipääsetav vaid saab kontrollida kas kood vastab WCAG 1.0 spetsifikatsioonile. Paljud WCAG validaatori poolt osutatud vead tuleb üle kontrollida ja nende kõrvaldamise vajadus otsustada koodist ning eesmärkidest sõltuvalt. Valideerimisel saadud tulemused on vajalikud selleks, et juhtida tähelepanu võimalikele takistustele erivajadustega kasutajatele ligipääsu tagamisel. Vigade ja probleemide tõsiduse ja nende kõrvaldamise vajaduse ning mooduse peab programmeerija otsustama kooskõlas WCAG spetsifikatsioonide ja veebivormide eesmärgiga.

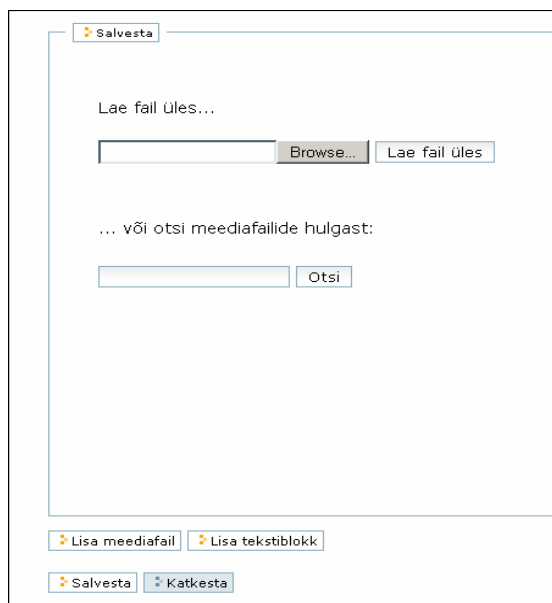
5.2 Think Aloud meetodiga kasutajatestimise tulemused

Think Aloud meetodiga kasutajatestimise tulemuste hindamisel on aluseks võetud kontrollloendi punktid, mis lihtsustavad tähelepanu fokuseerimist olulistele probleemidele.

Arusaamine

Esimesena kerkisid esile veebivormides navigeerimisega seotud probleemid. Vormide arhitektuuri ja navigeerimise arusaadavus oli lähtuvalt evalveerimise tulemustest väga oluline, kuna seoses sellega esinesid kõige suuremad takistused. Et kasutajal oleks üldse võimalik vormi elementidega toimetada ja neid õigesti kasutada on ennekõike vajalik süsteemi toimimise mõistmine ja oskus seal navigeerida.

Puudulik oli arusaamine nii vormi kogupikkusest kui ka hetkeasukohast, samuti edasi ja tagasi liikumise võimalustest ning millal ja kuidas salvestada. Seega navigeerimisribal pole piisavalt valikuid või pole need ühemõtteliselt selged. Arusaamatusi põhjustas ka vormi täitmiseks vajalike tegevuste sooritamise järjekord, mis võib lisaks tulemuseni jõudmise takistamise kaasa tuua vormide täitmiseks kuluva aja pikenemise. Navigeerimisribal asuvate valikute järjekorra loogilisust hinnates, võiks paremini paigutatud olla salvestamise nupp, mida kasutajad reeglina otsivad vormi alt, mitte aga selle kohalt. Samuti jäi kasutajale selgusetuks kahe salvestamispupu vajadus ja eesmärk.



Joonis 24. Salvestamise nupuga seotud probleemid.

Kui teha järeldusi nuppude arusaadavuse kohta siis esineb joonisel 24 vormides kolme erineva kujundusega nuppe, mille disain ei ole üheselt mõistetav. Näiteks on rõhutatud salvestamise nupu asemel katkestamise nuppu. Meediafaili üleslaadimise nupp on sama kujundusega nagu lahtrid teksti sisestamiseks, asudes tasapinnast madalamal ja jättes pigem lahtri kui nupu mulje. Sama veebivormi ulatuses võiks elementide kujundus olla kasutatavuse tõstmiseks järjepidev.

Nupud ei ole arusaadavad kuna nende pealkirjad pole ühemõttelised. Näiteks nupu „lisa tekstiblokk“ all pidas kasutaja võimalikuks sisestada korraga kõiki vajaminevaid multimeediaelemente kui ka salvestada eelnev töö. Sõna tekstiblokk ei ole üheselt mõistetav. Kohustusliku välja märgisena kasutatud punane ruut ei ole üheselt mõistetav, kuna punasega tähistatakse tavaliselt ettevaatusele kutsuvaid tähiseid.

Vormi elementidega toimetamine pole arusaadav, kuna pole selgelt välja toodud iga bloki eraldi salvestamise vajadus. Sisuliselt pakutakse kasutajale kahte salvestamise nuppu, millel on võrdne kaal. Kumbki neist ei domineeri ega vihja, mida peab esimesena vajutama. Selgusetuks jääb, kas vormi elementide täitmise järjekord on oluline. Seetõttu kahtles kasutaja, kas eelnevalt peaks laadima kõik meediafailid pakutud kausta ja alles siis hakkama looma veebiesitlust või saab seda teha samaaegselt ja järjest. Suureks puuduseks tuleb vormide struktuuri tuttavlikkuse puudulikkus ja seetõttu eksis kasutaja vormide täitmisel korduvalt.

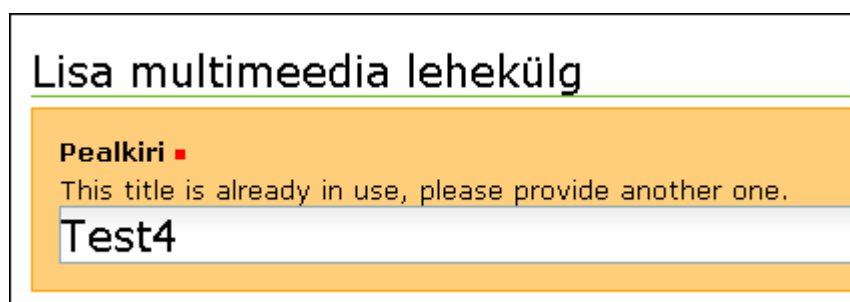
Suhtlemine

Veebivormidega tõrgeteta suhtlemise hindamisel ei ole täidetud ükski kontroll-loendi punkt. Enne vormidega tööle asumist ei esitata sissejuhatavat teksti. Samuti ei pakuta võimalust otsida abiinfot. Jooniselt 25 on näha, et veateadete kasu ei ole suur, kui ei pakuta välja lahenduskäiku. Selle asemel võiks olla viide abimenüüle või konkreetne lahendus probleemile.



Joonis 25. Veateade ei paku kasutajale lahendust.

Antud kontekstis ei oma kasutajale esitatav tagasiside nii suurt tähtsust kui näiteks erinevate teenuste osutamisel. Küll aga oleks vajalik esitada eesti keelsel leheküljel juhendav materjal ja väljade pealkirjad inglise keele asemel eesti keeles (joonis 26). Kui puudub võimalus otsida abiinfot võiks lisada hiirega üleliikumise tagasiside, mis võimaldaks saada lisainfot vormide täitmise kohta. Kuna puudus abiinfo kasutamise võimalus, siis ei saanud kasutaja lahendada esmaseid arusaamisega seotud probleeme. Seetõttu tekkis edasises vormide kasutamise etapis takistusi, mis oleks võinud täitmise lõpetamise täielikult välistada.



Joonis 26. Veateade on eesti keele asemel inglise keeles.

Kinnituse küsimine enne salvestamist ei ole antud keskkonnas väga vajalik kuna materjali on võimalik ise muuta ja kustutada.

Kasutamine

Veebivormide ja nende elementide kasutamise hindamisel esines samuti hulgaliselt probleeme, mis olid tingitud nii arusaamise kui kasutaja informeerimise puudulikkusest. Paljudele lisatud multimeediaelementidele ei ole määratud alt elementi ja selle väärtust. Seega pole vormid häälbrowseriga ligipääsetavad. Seoses mediafailide üleslaadimise ebaõnnustumisega kasutajaga testimise käigus oli tunda, et kasutaja on sellest oluliselt häiritud, mis võib mõjutada edasist vormide täitmist. Sellest lähtuvalt võib paralleelsele tömmata teiste võimalike kasutajate käitumisega analoogsetes olukordades. Testija ei vajutanud kordagi ülemisele salvestamise nupule, mistõttu võib järeldada, et salvestamisnuppude kasutamine praeguses asukohas pole põhjendatud (joonis 27).

Multimeedia lehekülje toimetamine			
Pealkiri ■			
viimane			
Põhitekst			
Muuda Liiguta üles Liiguta alla Kustuta			
First Name	Email	Home	
	Middle Initial	Computer type	Work
Last Name		<input type="radio"/> Mac <input type="radio"/> PC	
	Internet speed	<input type="radio"/> 28.8 <input type="radio"/> 56k <input type="radio"/> T1	
Muuda Liiguta üles Liiguta alla Kustuta			
viimane			
Lisa meediafail Lisa tekstiblokk			
Salvesta Katkesta			

Joonis 27. Elementide liigutamisel jääb nende pealkiri (Põhitekst) samasse kohta.

Elementide ja nende kontrollijate vaheline seotus joonisel 29 ei tule esile, kuna „kirjelduse lisamise“ silt asetseb lahtri horisontaalsest keskmest allpool. Testijal ei õnnestunud multimeedia faili üles laadimine, kuna sobivatele failiformaatidele polnud viidatud.

Seetõttu ei saanud testija ühtegi vihjet elementide kasutamise kohta ja iga ebaõnnestunud üleslaadimise järgselt kuvati sama sisestamise lehekülge.

Visuaalne kasutatavus

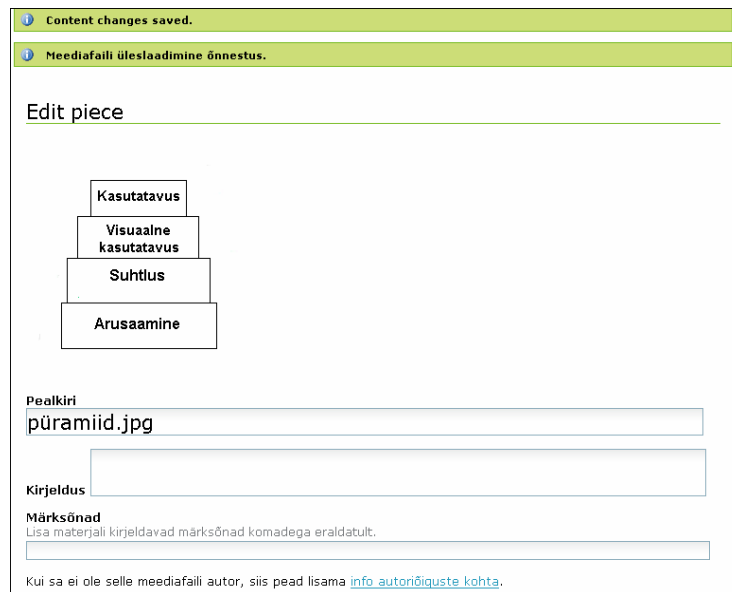
Vormide visuaalse kasutatavuse hindamisel esineb samuti olulisi vigu, mis raskendavad arusaamist ja kasutamist. Horisontaalset kerimist leheküljed ei nõua, küll aga on vertikaalse kerimise vajadus tingitud sisestava materjali hulgast, mis antud konteksti arvestades võib olla väga mahukas.



Joonis 28. Salvestamise nupuga seotud probleemid.

Värvide valiku hindamisel joonisel 28 võib tähelepanu juhtida nuppude kujundusele, kus võiks domineerida katkestamise nupu asemel salvestamise nupp. Samuti kõikide nuppude kujundus võiks olla sama vormi ulatuses järjepidev, sest praegusel juhul on kasutatud kolme erinevat võimalust. Sõltumata sellest ei tule välja nende erinev eesmärk.

Vormi täitmise mustandi faasis jjonisel 28 tuleks arvestada, et juhendav tekst kuvatakse otse sisestatud meediafaili peale ja kui tegemist on musta värvi pildiga siis tekst ei paista välja.



Joonis 29. Salvestamise nupuga seotud probleemid.

Vormides on valdavalt kasutatud vertikaalset paigutust, siiski esineb joonisel 28 nii horisontaalne kui vertikaalne paigutuse stiil. Järjepidevuse katkestab horisontaalse paigutuse kasutamine meediafailidele kirjelduse lisamise elementidel. Ebauhtlane joondus ja erinevad paigutuse stiilid võivad mõjutada lehekülje professionaalset väljanägemist ja seeläbi vähendada süsteemi usaldatavust.

5.3 Järeldus

Kõige tavapärasemad ja esmased valideerimise vahendid peaksid olema avalikult kättesaadavad ja lihtsa kasutajaliidesega automaatsed valideerimise programmid. Nendega võiks valideerida nii koodi vastavust spetsifikatsioonidele kui ka erivajadustega kasutajate ligipääsetavust. Kindlasti ei ole ühelgi nendest meetoditest eraldiseisvana piisavalt kaalu tegemaks lõplikke järeldusi kasutatavuse kohta. Seetõttu on ka magistritöös rakendatud erinevaid võimalusi. Isegi nelja kirjeldatud meetodi kasutamine ei pruugi tagada täielikku kasutatavust, kuna pakutavaid valideerijaid on lisaks väga palju: CSS validaator ja simulaatorid erinevat liiki värvipimeduse demonstreerimiseks. Sõltumata sellest on mitme erineva evalvatsiooni meetodi kasutamine kõige lähemal oluliste vigade tuvastamisele.

Lisaks sellele on väga oluline roll arendajal endal, kuna tuleb otsustada, milliseid vigasid ja kuidas ning kas on üldse vajalik parandada.

Automaatsete valideerijate eelised seisnevad nende kiiruses, mugavuses ja lihtsalt kasutatavuses. Kuna W3C poolt pakutavad validaatorid on otseselt seotud konkreetsete spetsifikatsiooni lõikudega, on nendega töötamine lisaks mugavusele ka kiirete paranduste sisseviimise eelduseks. Reeglina on kõik XHTML validaatoriga leitud vead parandatavad. WCAG spetsifikatsioonidele vastavuse kontrollimiseks validaatoritega on vaja sõltuvalt tekkinud vea raskusastmest arendajal otsustada kõrvaldamise võimalused ja vajalikkus, sest tihti on välja pakutud lahendused üldised ja ei pruugi arvestada koodi ning konteksti. Seega võib tekkida olukord kus kõiki WCAG validaatoriga leitud vigasid ei ole mõttekas või isegi võimalik parandada, kuna see võib osutada liiga ressursimahukas.

Kasutajaliideste hindamiseks mõeldud kontroll-loenditega on hea konkreetseid probleeme tuvastada. Nende eeliseks on samuti kiirus ja mugavus, kuna arendaja suudab ja saab seda teha iseseisvalt. Puuduseks võib lugeda olukorda, kus kontroll-loend on liiga üldine ega suuda tuvastada konkreetseid probleeme. Sellest tulenevalt oleks kasulik, kui kontroll-loend kataks mingit konkreetsemat osa veebilehel. Vastasel juhul on saadav kasu minimaalne. Kontroll-loendit saab kasutada ka kooskõlas kasutajatestimisega, mille korral saab testijat jälgides loetelu punkte täita.

Kasutajaga testimise läbiviimine annab kõikidest meetoditest enam adekvaatset tagasisidet. Kuna kasutaja on asetatud realselt ettetulevatesse olukordadesse, siis liiguvad uurimuse keskmesse olulised kasutatavusega seotud probleemid. Kasutajatestimine on läbiviimise ja analüüsimise seisukohalt väga ressursimahukas. Aega tuleb varuda nii ettevalmistamise, kui ka tõlgendamise jaoks piisavalt, kuna sellest oleneb tulemuste õigsus ja põhjalikkus.

Mitme evalvatsiooni meetodi rakendamine on kõige tulemuslikum lahendus kasutatavuse tagamisel, kuna toob välja palju erinevaid probleeme, mida ühe validaatoriga test ei võimalda.

Kokkuvõte

Magistritöö eesmärgiks oli pakkuda välja evalvatsioonimeetodite komplekt veebivormide kasutatavuse hindamiseks ja kasutatavusprobleemide avastamiseks. Eesmärk oli tingitud veebivormide kasutatavusega seotud raskustest. Kuna veebivormid on kujunemas üha olulisemaks interaktsiooni vahendiks erinevate teenuste ja hüvede pakkumisel nagu e-poed, pangateenused, e-õpe, e-maksuamet, e-kindlustus, siis on oluline tagada veebivormide parem kasutatavus ja ligipääsetavus lisaks ka erivajadustega kasutajatele.

Uurimisprotsess tugines veebivormide koostamiseks vajaminevatele spetsifikatsioonidele, põhimõtetele ja meetodikatele, mille tulemusel valmis:

- spetsifikatsioonide, meetodikate ja põhimõtete raamistik, mida veebivormide disainprotsessis aluseks võtta,
- veebivormidega enam seotud probleemide nimestik,
- evalvatsioonimeetodite komplekt, mida testiti õpiobjektide loomise keskkonna LeMill veebivormide peal.

Magistritöö eesmärgiks ei olnud välja pakkuda reegleid, kuidas peab veebivormide kujundust teostama. Kõik esitatud juhised ja põhimõtted on soovituslikud ning osutavad erinevatele kasutatavusega seotud probleemidele.

Evalvatsioonimeetodite komplekti koostamiseks viidi läbi erinevate spetsifikatsioonide, kasutatavuse põhimõtete ja printsiipide analüüs. Seejärel esitati veebivormide kasutatavusega seotud probleemistik, mille mõistmine võimaldab hinnata evalveerimise komplekti koostamise vajalikkust.

Veebivormidega seotud põhimõtete ja spetsifikatsioonide raamistikule tuginedes koostati evalvatsioonimeetodite komplekt, mis koosneb testidest automaatsete validaatoritega ja kontroll-loendiga ning kasutajaga testimisest Think Aloud meetodil.

Järgnevalt testiti evalvatsioonimeetodite komplekti õpiobjektide loomise keskkonna LeMill veebivormide peal. Testimise tulemustest lähtudes on mitme evalvatsiooni meetodi rakendamine kõige tulemuslikum lahendus kasutatavuse tagamisel, kuna toob välja palju erinevaid probleeme, mida ühe validaatoriga test ei võimalda.

Kasutatud kirjandus

Arguelles, T. (2002). Guidelines for Designing Web Forms. Loetud internetis 2.12.2006. Aadress <http://www.phptr.com/articles/article.asp?p=29892&rl=1>

Axup, J. (1998). Comparison of Usability Evaluation Methods (UEMs). Loetud internetis 3.03.2007. Aadress http://www.userdesign.com/usability_uem.html

Brinck, T., Gergle, D., Wood, S, D. (2002). Usability for the Web: designing web sites that work. San Francisco [etc.] : Morgan Kaufmann.

Cederholm, D. (2004). Loetud internetis 10.11.2006. Aadress <http://www.devarticles.com/c/a/Web-Design-Standards/Forms/2/>

Duval, E. (2004). Electronic forms must die. Loetud internetis 5.09.2006. Aadress <http://www.cs.kuleuven.ac.be/~hmdb/publications/publicationDetails.php?id=41316>

Fowler, S. (2004). Web application design handbook: best practices for web-based software. Amsterdam [etc.]: Morgan Kaufmann

Garrett, J. (2005). Ajax: A New Approach to Web Applications. Loetud internetis 10.03.2007. Aadress <http://www.adaptivepath.com/publications/essays/archives/000385.php>.

Hurst, M. (2004). The Page Paradigm. Loetud Internetis 20.01.2007. Aadress <http://www.goodexperience.com/blog/archives/000028.php>

Jarrett, C. (2000). Designing usable forms: the three-layer model of the form. Loetud internetis 10.11.2006. Aadress <http://www.formsthatwork.com/ftp/DesigningUsableForms.pdf>

Jarrett, C. (2005). Long Forms: Scroll or Tab. Loetud internetis 14.01.2007. Aadress <http://www.usabilitynews.com/news/article2352.asp>

Johnson, J. (2003). Web bloopers: 60 common web design mistakes and how to avoid them. Amsterdam [etc.] : Morgan Kaufmann.

Krug, S. (2000). Don't Make Me Think: A Common Sense Approach to Web Usability. Indianapolis, IN : New Riders.

Nielsen, J. (1994). Heuristic evaluation. In: Nielsen, Jakob, Mack, Robert L. (Eds.) "Usability Inspection Methods". John Wiley and Sons Crescimanno, B. New York NY

Nielsen, J. (2001). Error Message Guidelines. Loetud internetis 6.01.2007. Aadress <http://www.useit.com/alertbox/20010624.html>

Nielsen, J. (2005). The Power of Defaults. Loetud 10.01.2007. Aadress <http://www.useit.com/alertbox/defaults.html>

Nielsen, J. (2000). Drop-Down menus: Use Sparingly. Loetud internetis 12.01.2007. Aadress <http://www.useit.com/alertbox/20001112.html>

Nielsen, J. (2004). Checkboxes vs. Radio Buttons. Loetud internetis 12.01.2007. Aadress <http://www.useit.com/alertbox/20040927.html>

Nielsen, J. (2006). F-Shaped Pattern for Reading Web Content. Loetud internetis 22.02.2007. Aadress http://www.useit.com/alertbox/reading_pattern.html

Olsen, H. (2005a). Designing intersection flows. Loetud internetis 15.01.2007. Aadress http://www.guuui.com/issues/02_05.php

Olsen, H. (2005b). Navigation Blindness. Loetud internetis 15.01.2007. Aadress http://www.guuui.com/issues/01_05.php

Penzo, M. (2006). Label Placement in Forms. Loetud internetis 2.02.2007. Aadress <http://www.uxmatters.com/MT/archives/000107.php>

Preece, J. (1994). Human Computer Interaction. Addison Wesley.

Sensible Forms: A Form Usability Checklist. Loetud internetis 2.09.2006. Aadress <http://alistapart.com/articles/sensibleforms>

WebAim. (n.d). Loetud internetis 2.02.2007. Aadress http://www.webaim.org/techniques/forms/screen_reader.php

Wroblewski, L. (2005). Web Application Worm design. Loetud internetis 15.02.2007. Aadress http://www.lukew.com/resources/articles/web_forms.html

W3C. (1999). Web Content Accessibility Guidelines 1.0. Loetud internetis 2.12.2006. Aadress <http://www.w3.org/TR/WCAG10/>

W3C. (2000). Loetud internetis 20.02.2007. Aadress <http://www.w3.org/TR/WCAG10-HTML-TECHS/#forms>

W3C. (2005a). ...In 7 points. Loetud internetis 15.11.2006. Aadress <http://www.w3.org/Consortium/Points/>

W3C. (2005b). XForms - Xforms 1.0. Loetud internetis 10.04.2007. Aadress <http://www.w3.org/TR/2006/REC-xforms-20060314/>

W3C Web Accessibility Initiative. (2006a). Essential Components of Web Accessibility. Loetud internetis 20.11.2006. Aadress <http://www.w3.org/WAI/intro/components.php>

W3C. (2006b). The Next Generation of Web Forms. Loetud internetis 15.04.2007. Aadress <http://www.w3.org/MarkUp/Forms/>

Lisa 1. Think Aloud meetodi kasutajatestimise juhend

Testimist teostatakse LeMill keskkonna veebivormide peal. Lemill on õpiobjektide loomise keskkond. Veebivormide testimise eesmärgiks on hinnata kujunduse kasutatavust läbi selle tugevuste ja nõrkuste. Testitakse veebivormide kasutatavust, mitte kasutajat. Ülesande sooritamisel ei ole õigeid ega valesid lahenduskäike, kuna veebivormide kasutatavust ja loogilisust hinnatakse läbi toimingute.

Testijale tagatakse anonüümsus ning tema tegevusi ei avalikustata kõrvalistele isikutele.

Kogu protsessi vältel salvestatakse testija reaktsioone ja arvamusi. Salvestus on konfidentsiaalne ja kõrvalised inimesed sellega kokku ei puutu. Kui testija ei soovi, et teda filmitakse, tuleb tal sellest enne testimise algust teada anda.

Testija osalemine uurimuses on vabatahtlik, mis annab talle õiguse vahele jätta küsimusi või lõpetada tegevus soovitud ajahetkel.

Ülesande lahendamine ei tohiks võtta kauem kui 60 minutit. Selle aja vältel lahendatakse ülesanne, kasutades selleks LeMill veebivorme ja ettevalmistatud materjali. Testimise läbiviijal on õigus kogu protsessi vältel teha märkmeid testija käitumise kohta.

Evalveerimise alguses tuleb testijal lugeda läbi sooritamiseks vajaminev info ülesannete lehelt. Kui testija puutub ülesande lahendamise käigus kokku probleemide või takistustega siis testimise läbiviija ei saa teda aidata, kuna eesmärk on näha, kuidas testija probleemid iseseisvalt lahendab.


Ülesannete lahendamise käigus tuleb testijal mõelda valjusti, edastades kõik mõtted ja reaktsioonid, mis töö käigus tekivad. Testija ei pea oma tegevusi põhjendama ega mõtteid enne välja ütlemist viimistlema. Kui testijal tekivad mõttepausid, siis on testimise läbiviijal õigus teda suunata oma mõtete väljendamisele. Muul moel juhendavate või suunavate küsimustega testi läbiviija ei sekku.

Kuigi tegemist on ebaloomuliku ja kunstlikult tekitatud olukorraga testija salvestamise ja oma mõtete väljütlemise tõttu, peaks testija siiski proovima sooritada ülesandeid samalaadselt nagu kodus, koolis või tööl seda tavapäraselt teeb.

Kas testijal on lisaküsimusi?

Lisa 2. Veebivormide evalveerimise ülesanded

LeMill Edit, tips for use and more...



Web Forms

last modified 2007-05-02 13:42

This learning resource is in draft status.

Forms are used for all kinds of interactive purposes on the web. Forms allow users to select and purchase merchandise, to fill in surveys and questionnaires, to register for courses, to search for information on the web, and a long list of other actions.

When we talk about the accessibility of forms, we are usually referring to their accessibility to people who use screen readers. People with other types of disabilities are generally less affected by faulty forms. It should be noted, however, that everyone benefits from a well-organized, highly usable form, especially those with cognitive disabilities.

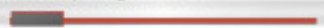
First Name	Email	Home Phone	
	Middle Initial	Computer type	Work Phone
LAST Name		<input type="checkbox"/> Mac <input type="checkbox"/> PC	Country
	Internet speed	<input type="radio"/> 28.8 <input type="radio"/> 56k <input type="radio"/> T1	

Visually, this table makes at least some sense (even if it is a bit confusing). For the most part, a person with sight can figure out that the First Name label goes with the control beneath it, and that the Email label goes with the control to the right of it. What happens when we remove the colors and strip it of its formatting, though? A screen reader will read the above table in this order: The visual nature of the following table will limit its usefulness to users of screen readers. The point that is being illustrated is that screen readers start with the top row, reading the entire content of each cell from left to right across each column in the row, until the end of the table is reached. Select this link to skip over the table.

1	2	3	
4	5	6	7
8	9	10	11
12	13	14	

Even though a screen reader will not render the colors that have been shown, they were left in for comparison purposes. If you compare the reading order to the visual order in the table above it, you can see that the labels and the form elements do not match up very well. The person using the screen reader is likely to become very confused, and may fill in the form incorrectly, typing information in the wrong places.

Let's hear what the form sounds like to a screen reader:

[Feb_Forms\(helena\)-jaws.mp3](#)
Audio clip length: 00:22
 [\[Download\]](#)

Using the Tab key:

First Name	<input type="text"/>
Last Name	<input type="text"/>
Middle Initial	<input type="text"/>
Home Phone	<input type="text"/>
Work Phone	<input type="text"/>
Country	<input type="text"/>
Computer Type <input type="radio"/> Mac <input type="radio"/> PC <input type="radio"/> Linux	Internet Speed <input type="radio"/> 28.8 <input type="radio"/> 56k <input type="radio"/> T1

There are two things that we can do to the previous table to fix it:

1. Put the labels adjacent to their control
2. Use HTML markup to associate the controls explicitly with their labels

One way to fix the accessibility problems of the above table would be to reduce the table to two columns, placing the labels directly to the left of the form elements, then adding in the appropriate HTML elements

[hpr.mp3](#)

Audio clip length: 00:27



[\[Download\]](#)



This work is licensed under a [Creative Commons Attribution-ShareAlike 2.5 License](#).

Summary

Title: Design Principles and Evaluation Methods for Web Forms

Keywords: usability, design, web forms.

The master thesis focuses on the issues of design principles and evaluation methods for web forms. Forms are used for all kinds of interactive purposes on the web: shopping, banking, e-learning environments etc. Therefore it is important to focus on increasing usability and accessibility.

The main goal of this thesis was to offer a set of methods to evaluate the usability and accessibility of web forms.

The outcomes of this thesis are:

- an analysis of specification, methods and principles which can be used to design web forms,
- a set of most common problems with web forms,
- a set of evaluation methods, which has been tested on LeMill web forms.

In order to reach the goal different principles and specifications for web forms and their influences on set of evaluation methods were analyzed.

Most common problems with web forms emphasize the importance of using evaluation methods.

A set of evaluation methods for web forms has been compiled and evaluated. The set consists of HTML and WCAG validation, usability checklist and Think Aloud user testing. Learning object authoring templates in LeMill environment (see <http://lemill.net>) were used for evaluating the set of evaluation methods

The results of evaluation show that using a set of evaluation methods will point out more usability and accessibility problems than using only one method.

The length of the thesis is 77 pages. 29 figures and 11 tables are included. 32 items of literature and web resources are referenced. The thesis is written in Estonian.