

UNIVERZA V LJUBLJANI
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Miha Lampret

**Uporabniški vmesniki in uporabniška
izkušnja**

DIPLOMSKO DELO

VISOKOŠOLSKI ŠTUDIJSKI PROGRAM
PRVE STOPNJE
RAČUNALNIŠTVO IN INFORMATIKA

MENTOR: izr. prof. dr. Narvika Bovcon

Ljubljana, 2018

COPYRIGHT. Rezultati diplomske naloge so intelektualna lastnina avtorja in Fakultete za računalništvo in informatiko Univerze v Ljubljani. Za objavo in koriščenje rezultatov diplomske naloge je potrebno pisno privoljenje avtorja, Fakultete za računalništvo in informatiko ter mentorja.

Besedilo je oblikovano z urejevalnikom besedil L^AT_EX.

Fakulteta za računalništvo in informatiko izdaja naslednjo nalogo:

Tematika naloge:

Načrtovanje vmesnikov naj bi temeljilo na zagotavljanju dobre uporabniške izkušnje. V diplomski nalogi predstavite vidike, ki sestavljajo dobro uporabniško izkušnjo, metode za njeno zagotavljanje in kriterije za ocenjevanje. Uvide s področja raziskav človeške zaznave povežite s smernicami za oblikovanje računalniških vmesnikov. Na primerih pokažite, kako razvoj tehnologije vpliva na trende v oblikovanju sodobnih vmesnikov, npr. spletnih strani.

Kazalo

Povzetek

Abstract

1	Uvod	1
1.1	Tipi uporabniških vmesnikov	1
1.2	Motivacija	2
2	Pregled področja	5
2.1	Uporabniška izkušnja	5
2.2	Uporabnost	10
2.3	Uporabniško usmerjeno načrtovanje (UCD)	13
3	Človeško zaznavanje in smernice za oblikovanje uporabniških vmesnikov	19
3.1	Zaznavanje je pogojeno	19
3.2	Človeški vid	23
3.3	Človeški spomin	34
4	Metode za ocenjevanje uporabniških vmesnikov	41
4.1	Pregled izbranih metod in tehnik za zagotavljanje dobre uporabniške izkušnje	47
5	Trendi izdelave spletnih uporabniških vmesnikov	67
5.1	Trendi tehnologij	67

5.2	Trendi oblikovanja	70
6	Zaključek	79
6.1	Sklepne ugotovitve	79
	Literatura	84

Seznam uporabljenih kratic

kratica	angleško	slovensko
CLI	command-line interface	konzolno usmerjeni vmesnik
TUI	text-based user interface	dvodimenzionalni tekstovni uporabniški vmesnik
GUI	graphical user interface	grafični uporabniški vmesnik
NUI	natural user interface	naravni uporabniški vmesnik
MOU	usability evaluation methods	metode ocenjevanja uporabnosti
UX	user experience	uporabniška izkušnja
JSX	JavaScript syntax extension	razširjena sintaksa JavaScript
API	application programming interface	programski vmesnik
XML	extensible markup language	razširljivi označevalni jezik
SEO	search engine optimizer	optimizacija spletne strani

Povzetek

Naslov: Uporabniški vmesniki in uporabniška izkušnja

Avtor: Miha Lampret

Danes si težko zamislimo idejo, ki še ne bi obstajala v obliki spletne ali mobilne rešitve. Za uspešen nastop našega produkta na trgu je odločilnega pomena dobra uporabniška izkušnja. Vedno več razvijalcev produkte razvija samostojno ali pa v manjših agilnih skupinah, zato poznavanje osnov človeškega zaznavanja, metod ocenjevanja in testiranja uporabnosti, ter trendov oblikovanja in tehnologij, ni pomembno le za oblikovalce uporabniške izkušnje, ampak tudi za razvijalce programske opreme. Dobra komunikacija in usklajevanje med njimi sta ključnega pomena za učinkovitost razvoja in kakovost končnega produkta.

V diplomii opišem, kako deluje človeško zaznavanje ter katere osnovne smer-nice in navodila za oblikovanje računalniških vmesnikov so se izoblikovale na podlagi njegovih sposobnosti in omejitev. Za razvoj uporabnih vmesnikov je testiranje med uporabniki ključnega pomena, zato predstavim prednosti, slabosti in nekatera orodja izbranih metod za ocenjevanje uporabniških vme-snikov, s katerimi lahko hitro, poceni in brez veliko izkušenj, povečamo upo-rabnost vmesnikov. Na podlagi sodobnih spletnih predstavitev, v zadnjem delu analiziram trende oblikovanja in opišem pogosto uporabljene tehnolo-gije.

Ključne besede: zaznavanje, oblikovanje, uporabniška izkušnja, upo-rabniški vmesniki.

Abstract

Title: User interface and user experience

Author: Miha Lampret

Today it is hard to imagine an idea that would not already exist in form of a web or mobile solution. For a successful appearance of our product on the market, a good user experience is decisive. More and more developers develop products independently or in smaller agile groups, therefore knowledge of basic human perception, evaluation and usability testing methods, latest technology and modern design trends are not only important for designers of user experience, but also for software developers. Good communication and coordination between them are crucial for efficient development and the quality of the final product.

In the diploma thesis I describe how human perceptions work and what basic guidelines and instructions for designing computer interfaces have been developed based on its capabilities and limitations. For the development of useful interfaces, testing among users is of key importance; therefore, I define advantages, disadvantages and some tools of selected methods for product and user testing that can quickly, cheap and without much experience, increase the usability of interfaces. On the basis of modern online presentations, in the latter part, I analyze the design trends and latest commonly used technologies.

Keywords: human perception, design, user experience, user interfaces.

Poglavje 1

Uvod

1.1 Tipi uporabniških vmesnikov

Uporabniški vmesniki se v zadnjih letih hitro spreminjajo. Od začetnih konzolno usmerjenih vmesnikov z ukazno vrstico (CLI Command-line interface) in dvodimenzionalnih tekstovnih vmesnikov (TUI Text-based user interface), za katere je bilo potrebno pomnjenje ukazov in je uporabnikova interakcija s sistemom potekala le preko tipkovnice, do današnjih grafično podprtih (GUI Graphical user interface) in naravnih uporabniških vmesnikov (NUI Natural user interface).

Za razliko od CLI in TUI, interakcija uporabnika z grafičnimi vmesniki poteka z uporabo tipkovnice in miške, vedno bolj pa se razvija tudi področje naravnih vmesnikov, kjer za njihovo upravljanje ne potrebujemo vmesnih naprav in interakcija poteka z uporabo naravnih gibov. Osnovni cilj razvoja uporabniških vmesnikov je torej v njihovi lažji uporabi in naučljivosti. Kljub hitremu razvoju naravnih, so vmesniki kjer uporabnik s sistemom komunicira preko vhodnih in izhodnih naprav (zaslone na dotik, tipkovnica, miška, grafična tablica, ...) še vedno najbolj pogost način interakcije človek-računalnik.

Sistemske vmesnike, ki predstavljajo prvo skupino najbolj pogostih grafičnih vmesnikov, se razvijajo kot programi in aplikacije za določene ope-

racijske sisteme, kot so Windows, OsX, iOS, Android, itn., in se izvajajo lokalno, na strojni opremi sistema. Njihov razvoj in oblikovanje narekujejo za posamezen sistem značilne smernice in zahteve.

Druga skupina grafičnih vmesnikov so spletni uporabniški vmesniki. Slednji so najbolj pogosti kot spletne predstavitve, ki delujejo preko spletnih in mobilnih brskalnikov in se ponavadi izvajajo nelokalno, na zunanem strežniku. Razvoj spletnih vmesnikov je za razliko od sistemskih veliko manj omejen in posledično bolj kompleksen. Njihova uporabnost je odvisna od hitrosti internetnih povezav, zahteva pa tudi prilagoditve za različne velikosti zaslonov in zahteve brskalnikov.

Področje uporabniških vmesnikov je široko, v posameznih delih diplome je poudarek na zagotavljanju dobre uporabniške izkušnje spletnih uporabniških vmesnikov, čeprav veliko ugotovitev na splošno velja za vse vmesnike.

1.2 Motivacija

Uporaba spletnih aplikacij in strani za potrebe poslovanja narašča in predstavlja vedno bolj pomemben del sodobne ekonomije. Poleg uporabe za trgovanje, medije in odnose z javnostjo, se tudi v segmentu javne uprave, podjetij in končnih potrošnikov njihova uporaba povečuje [8].

Dobiček spletnega mesta ESPN.com se je za kar 35 % povišal, ko so s pomočjo mnenj uporabnikov prenovili svojo spletno stran [48]. Lastniki podjetij se vedno bolj zavedajo, da posvečanje le osnovnim poslovnim ciljem, naprednim funkcionalnostim ter zmogljivostim programske in strojne opreme ni dovolj. Danes že skoraj vsaka ideja obstaja v obliki spletne ali mobilne rešitve in ponavadi obstaja tudi več ponudnikov istih ali podobnih storitev, zato je glavni dejavnik, s katerim lahko konkuriramo na trgu, osredotočanje na končnega uporabnika in dobra uporabniška izkušnja.

Uporabniški vmesnik v veliko primerih predstavlja prvi stik uporabnika s podjetjem ali storitvijo. Dober prvi vtis je v 94 % pogojen z lepo oblikovanim vmesnikom [44]. Po nekaterih raziskavah kar 38 % uporabnikov spletno stran

zapusti le zaradi neprivlačnega izgleda [4], 88 % pa jih meni, da se zaradi slabe izkušnje na spletno stran ne bi vrnil [26]. Lep in funkcionalen produkt je relativno lahko in poceni doseči, vendar to še ne pomeni, da je uporaben in komercialno uspešen.

Za razvoj in zagotovitev dobre uporabniške izkušnje je potrebno veliko izkušenj in znanja. S poznavanjem osnov kognitivne psihologije, ki proučuje človeške spoznavne procese, lahko bolje razumemo nastanek smernic in navodil za oblikovanje in jih lažje implementiramo pri načrtovanju uporabnih uporabniških vmesnikov.

Pri razvoju uporabnih vmesnikov je vmesno testiranje z uporabniki ključnega pomena. Veliko metod za ocenjevanje uporabnosti (metode MOU), s katerimi pridobivamo informacije o uporabnikih in pomankljivostih sistema, je včasih predstavljalo velik strošek za manjša podjetja in samostojne razvijalce programske opreme. S hitrim razvojem orodij in programske opreme so metode postale bolj dostopne in hitrejše. Veliko metod je avtomatiziranih, enostavnih za izvedbo in ne zahtevajo posebnega strokovnega znanja, z njimi pa lahko močno izboljšamo uporabnost sistemov in celotno uporabniško izkušnjo.

Diplomsko delo je sestavljeno iz naslednjih poglavij:

- **Uvod**, v katerem predstavim motivacijo oz. smisel obravnavane teme.
- **Pregled področja**, kjer definiram uporabniško izkušnjo in uporabnost, ter predstavim njune lastnosti. Pojasnim, kaj je uporabniško usmerjeno načrtovanje in njegov pomen za uporabniško izkušnjo.
- **Človeško zaznavanje in smernice za oblikovanje uporabniških vmesnikov**, kjer opišem človeške sposobnosti in omejitve zaznavanja, ter kako so vplivale na nastanek uveljavljenih smernic oblikovanja uporabniških vmesnikov. S praktičnimi primeri opišem, kako so pogojeni človeško zaznavanje, človeški vid in spomin.

- **Metode za ocenjevanje uporabniških vmesnikov**, kjer najprej predstavim in umestim metode MOU, v drugem delu pa analiziram metode in tehnike testiranja uporabnosti, preglede ekspertov in poizvedbe. Opišem njihov okvirni potek in definiram njihove prednosti, slabosti in orodja, ki se za njihovo izvedbo najbolj pogosto uporabljajo.
- **Trendi izdelave spletnih uporabniških vmesnikov**, kjer v prvem delu na kratko predstavim izbrane trende spletnih tehnologij za razvoj uporabniških vmesnikov. V drugem delu pa opišem in analiziram sodobne trende pri oblikovanju spletnih predstavitev, ter z uporabo tehnologij prvega dela prikažem nekaj primerov.
- **Zaključek**, kjer povzamem glavne ugotovitve in nakažem, kako bo pridobljeno teoretsko znanje podlaga za praktično delo.

Poglavje 2

Pregled področja

2.1 Uporabniška izkušnja

Da je vmesnik prijazen za uporabo in zagotavlja čim več vidikov dobre uporabniške izkušnje, je potrebno zagotoviti več kriterijev. Med najbolj pomembnimi kriteriji načrtovanja so zagotovitev uporabnosti, poznavanje potreb, okolja in ciljev končnega uporabnika ter prilagoditev sistema okoli njih.

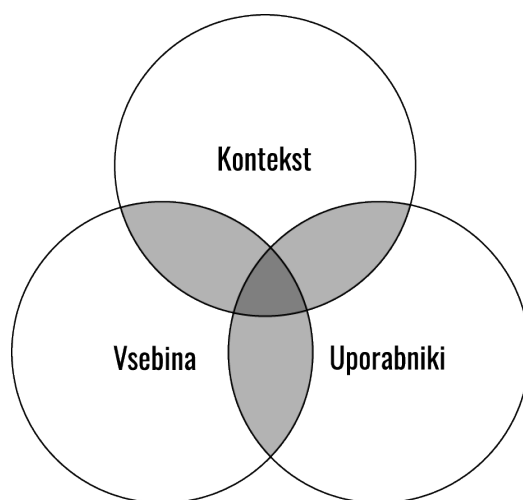
Uporabniško izkušnjo (UX) je težko jasno definirati, saj vsak avtor drugače opisuje, kaj vse ta izraz vključuje. Po Nielsen Norman Group, vodilni strokovni organizaciji na področju kakovosti in uporabe spletnih tehnologij, je definirana kot: "Uporabniška izkušnja obsega vse vidike interakcije uporabnika s podjetjem, njegovimi storitvami in izdelki" [36]. Osredotoča se na globoko razumevanje uporabnikov, njihovih potreb, vrednot, sposobnosti in omejitev. Poleg uporabnika pa se osredotoča tudi na cilje in vrednote podjetja, ki izdelke in storitve nudi.

Uporabniška izkušnja ni objektivni pojem, ampak zajema vsa subjektivna doživljanja, ki so pogojena s človeškimi čustvi, z mislimi in odnosi z okolico. Vsak človek oziroma uporabnik ima svojo zgodovino in vsi ljudje stvari psihološko ne dojemamo enako, zato je še toliko težje enolično definirati uporabniško izkušnjo, kaj vse vključuje in kako jo zagotoviti [8]. Velik del ocenjevanja in zagotavljanja dobre uporabniške izkušnje predstavljajo zato

tudi kognitivna, eksperimentalna in inženirska psihologija [35], ki se osredotočajo na človekove sposobnosti zaznavanja in se ukvarjajo s preučevanjem človeškega vedenja v vidikih, povezanih z uporabo storitev.

Eden glavnih problemov pri zagotavljanju uporabniške izkušnje je njeno težavno merjenje, saj je ta skupek posamičnih občutij, ki je večji od svoje vsote in pogosto zelo subjektivno pogojen (vsak človek ima na primer svoje priljubljene barve). Psihološki vpliv je lahko še odločilnejši, saj ljudje interaktivne sisteme, kot so spletne strani in aplikacije, pogosto ocenjujemo za boljše ali slabše, le na podlagi izgleda [35, 8].

Peter Morville, oblikovalec in inženir informacijske arhitekture, je uporabniško izkušnjo predstavil z diagramom informacijske arhitekture (Slika 2.1), ki dobro predstavlja medsebojno povezavo ključnih elementov uporabniške izkušnje: vsebine, konteksta in uporabnikov.



Slika 2.1: Diagram informacijske arhitekture po Morvillu

S podrobnejšim raziskovanjem tematike je Morville ugotovil, da je diagram pomanjkljiv in ne vsebuje vseh vidikov uporabniške izkušnje. Za prikaz in delitev slednje je zato ustvaril diagram v obliki satja (Slika 2.2), s šestimi atributi [25]:



Slika 2.2: Diagram uporabniške izkušnje po Morvillu

Funkcionalnost (angl. Usefulness) zagotavlja, da sistem izpolnjuje zahteve, potrebe in želje vseh uporabnikov in lastnikov, ki so v povezavi s sistemom.

Uporabnost (angl. Usability), ki skupaj z atributom **koristnosti** (angl. Utility) tvorita funkcionalnost sistema, je sestavljen iz naučljivosti, učinkovitosti, zapomnljivosti, števila napak in zadovoljstva uporabnikov sistema. Brez uporabnosti sistema o uporabniški izkušnji ne moremo govoriti.

Zaželjenost (angl. Desirability) loči uporaben produkt od produkta, ki si ga uporabnik želi uporabljati oziroma kupiti in je prav tako uporaben. Zaželjenost je ponavadi povezana s predstavitvijo multimedijskih vsebin in zgodb, ki uporabniku vzpodbudijo pozitivna čustva. Po Donu Normanu [35] pozitivna čustva v povezavi z uporabo interaktivnih sistemov zbudimo kot

odziv na treh ravneh: na visceralni (tj. notranji, iracionalni, instinktivni) ravni, na ravni vedenja in na ravni razmišljanja.



Slika 2.3: Normanove ravni pozitivnih čustev po Normanu

Načrtovanje visceralnega odziva (angl. Visceral design) se osredotoča na to, kako bo končen produkt izgledal in kakšne občutke vzpodbudi v uporabniku. Cilj je, da s pomočjo čustev, ki jih vzpodbudi vizualna predstavitev, pri uporabniku doseže boljšo uporabniško izkušnjo, lahko pa tudi služi poslovnim interesom (npr. uporabnika z vplivom na njegova čustva prepriča, da opravi nakup). Načrtovanje vedenjskega odziva (angl. Behavioural design) se osredotoča na čustva, ki jih uporabnik začuti pri doseganju ali nedoseganju ciljev pri uporabi produkta. Osredotoča se na lastnosti uporabnosti in kako z njimi pozitivno vplivati na uporabnikova čustva. Načrtovanje razmišljujočega odziva (angl. Reflective design) se posveča uporabnikovi racionalizaciji in intelektualizaciji produkta. Osredotoča se na zavestno razmišljanje uporabnikov o pozitivnih in negativnih lastnostih produkta, torej racionalni presoji pomena produkta. Če se uporabniki na nivoju instinktivnega odziva sprašujejo "Ali mi je pametna ura všeč?", se na razmišljujočem nivoju sprašujejo "Kako me bodo prijatelji presojali, če nosim pametno uro?". Zaradi razmišljujočega načrtovanja uporabniku

uporabnost lahko ne predstavlja glavne lastnosti produkta, saj poišče druge nefunkcionalne lastnosti, ki jo pretehtajo. [20]

Dostopnost (angl. Accessibility) nam pove, kako lahko je uporabljati sistem, predvsem za ljudi, ki imajo težave z vidom, sluhom, gibalnimi sposobnostmi, spominom itd. Ker ljudje s posebnimi potrebami predstavljajo kar sedmino celotnega prebivalstva sveta, je dostopnost sistema ena ključnih lastnosti dobre uporabniške izkušnje [21]. Zagotavljanje dostopnosti ni pomembno le zaradi lažje uporabe sistema za ljudi s posebnimi potrebami, ampak tudi za ljudi iz določenih starostnih skupin in ljudi, ki sistem uporabljajo v določenih težjih razmerah, zaradi okolja uporabe ali naprave. Slednje danes velja predvsem pri uporabi mobilnih naprav, tablic in pametnih telefonov, saj je njihova uporabnost lahko odvisna od okolja, v katerem jih uporabljajo. Sistemi, ki se uporabljajo na prostem, na primer zaradi vpliva sončne svetlobe lahko zahtevajo drugačno prilagoditev vmesnika. Podobno mobilne aplikacije, namenjene mlajšim otrokom, ponavadi zahtevajo drugačne prilagoditve in načrtovanja kot za ljudi v višjih starostnih skupinah.

Najdljivost (angl. Findability) v povezavi z uporabniško izkušnjo ponavadi povezujemo s tem, kako lahko je uporabnikom poiskati produkt oziroma kako zagotoviti, da trženi produkt doseže ciljno publiko. V okviru uporabe produkta pa najdljivost uporabnikom zagotavlja lahek dostop do informacij, ki jih iščejo.

Kredibilnost (angl. Credibility) je za uspešnost spletnih strani ali aplikacij ključnega pomena in zagotavlja njihovo zaupanje v produkt ali storitev. Kredibilnost spletnih predstavitev zagotovimo z zaupanja vrednimi in legitimnimi opisi produktov, storitev in podatki podjetja.

Med pomembnejše lastnosti dobre uporabniške izkušnje spadajo tudi prenosljivost, kompatibilnost in konsistentnost.

Prenosljivost programske opreme omogoča, da oprema s podobno aplikacijsko logiko in vmesniki deluje na različnih platformah. Prenosljivost navadno zmanjša stroške, ki bi sicer nastali pri prilagajanju programske opreme različnim platformam.

Kompatibilnost je zmožnost dveh ali več produktov (programske ali strojne opreme, iste ali različne izdaje), da med seboj delujejo brez potrebnih modifikacij in uporabljajo enake podatkovne formate. Pogosto razvoj Android in iOS mobilnih aplikacij zaradi njihovih omejitev in smernic zahtevata drugačne načine interakcije in posledično drugačen izgled uporabniških vmesnikov.

S **konsistenco** dosežemo, da je način interakcije in prikaz grafičnih elementov uporabniškega vmesnika na različnih platformah in sistemih, kot tudi znotraj teh, enak oziroma čimbolj podoben. Uporabnikom ta lastnost omogoča lažjo naučljivost in posledično uporabo, ko na primer isto storitev ali produkt uporabljajo z uporabo tako njihove spletne kot tudi mobilne aplikacije.

2.2 Uporabnost

Ko govorimo o uporabniški izkušnji, je eden najbolj pomembnih vidikov za zagotavljanje, ocenjevanje in preučevanje le te, uporabnost uporabniških vmesnikov. Slednjo je v primerjavi s celotno uporabniško izkušnjo lažje meriti in ocenjevati, ter predstavlja pomembno lastnost praktične sprejemljivosti sistemov.

V literaturi je uporabnost zelo različno definirana, saj je širok pojem sestavljen iz več komponent in dejavnikov. Na kratko bi uporaben vmesnik lahko opisali kot vmesnik, ki je preprost za uporabo, dobro izgleda in zagotavlja tri glavne lastnosti [12]:

- Nov uporabnik se na vmesnik hitro prilagodi in ga zna uporabljati.
- Novemu uporabniku je vmesnik enostaven za uporabo in mu pri uporabi hitro zagotovi željene cilje.

- Vmesnik ne obremenjuje uporabnikovega spomina ter mu omogoča hitro pomnjenje potrebnih interakcij za doseg željenega cilja.

V standardu ISO 9241-11 [2] je uporabnost opisana kot: »Merilo, kako učinkovito, storilno in s kakšno mero zadovoljivosti lahko določeni uporabniki uporabljajo produkt v določenem kontekstu uporabe.« Ta definicija zelo splošno opisuje ocenjevanje uporabnosti programske opreme in za današnji čas, kjer spletne in mobilne predstavitve prevladujejo nad programskimi aplikacijami, ni popolna.

Eden glavnih predstavnikov na področju uporabniške izkušnje Jakob Nielsen, je uporabnost opredelil kot večdimenzionalno lastnost uporabniškega vmesnika s petimi glavnimi atributi [33]:

- Naučljivost.
- Učinkovitost.
- Zapomnljivost.
- Število napak.
- Zadovoljstvo.

Naučljivost sistema spada med najbolj pomembne komponente uporabnosti in nam pove, s kakšno stopnjo težavnosti se uporabniki lahko naučijo uporabe glavnih funkcionalnosti sistema, ki ga uporabljajo prvič.

Učinkovitost nam pove, kako hitro uporabnik, ki sistem pozna, opravi določeno nalogo. Učinkovitost sistema v praksi lahko merimo s številom opravljenih nalog ali s časom, potrebnim za doseg določenega cilja.

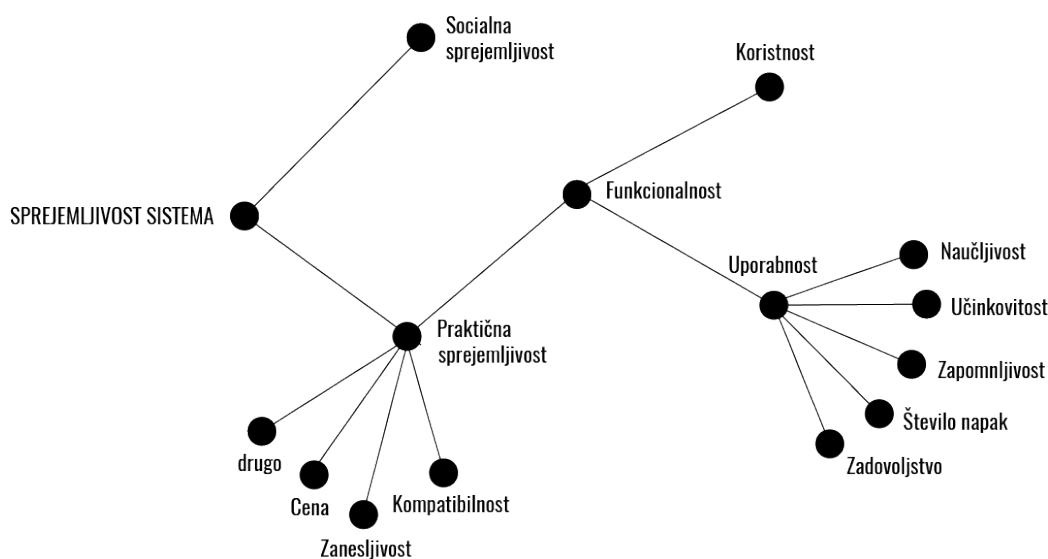
Zapomnljivost definira zmožnost sistema, da uporabniku omogoča lahko pomnjenje in izvrševanje njegovih osnovnih funkcionalnosti in principov delovanja tudi po daljši neuporabi sistema.

Število napak nam pove, koliko napak uporabnik, glede na njegovo predhodno poznavanje sistema, stori ob uporabi. Napake merimo tudi glede na njihovo kritičnost in vpliv na samo izkušnjo, ter kako hitro jih uporabniki znajo odpraviti.

Zadovoljstvo opisuje subjektivno stopnjo zadovoljstva uporabnika pri uporabi sistema.

Poleg uporabnosti obstaja še veliko drugih lastnosti dobrega uporabniškega vmesnika, kot so zmogljivost, zanesljivost, zaščita in stroški, ki skupaj s koristnostjo predstavljajo praktično sprejemljivost sistema.

Nielsen je za namene zadovoljitve želj in potreb uporabnikov, lastnikov in drugih interesnih skupin sistema, oblikoval model lastnosti sprejemljivega sistema [27]. Model predstavlja pomembnost tako praktične, kot tudi družbene sprejemljivosti sistema (Slika 2.4).



Slika 2.4: Sprejemljivost sistema po Niensenu

Koristnost sistema (angl. Utility), ki spada med bolj pomembne lastnosti sistema, nam pove, v kolikšni meri sistem ponuja rešitve za določene

cilje uporabnika. Pove nam, ali njegove komponente in posebne značilnosti, uporabnikom koristijo pri doseganju ciljev, torej ali imajo funkcionalen pomen. **Funkcionalnost sistema** (angl. Usefulness) zavzema tako lastnosti koristnosti kot tudi uporabnosti in na splošno meri, ali določeni sistem izpolnjuje zahteve, potrebe in želje vseh deležnikov (uporabnikov in lastnikov), ki so direktno ali posredno povezani s sistemom.

2.3 Uporabniško usmerjeno načrtovanje (UCD)

Več kot ima produkt funkcionalnosti in bolj kot so te slabo zasnovane, težje so implementacija, vzdrževanje, učenje in uporaba takega sistema, posledično pa je tudi težje zagotoviti vse vidike dobre uporabniške izkušnje. Pogoste potrebe in naloge produkta je preprosto definirati vnaprej, težje pa definiramo potrebe, ki se pokažejo manj pogosto, vendar so prav te ključnega pomena za dobro uporabniško izkušnjo [43]. Uporabniško usmerjen dizajn uporabniško izkušnjo izboljša.

Uporabniško usmerjen dizajn (angl. User centered design UCD) je pristop za razvoj in oblikovanje uporabniških vmesnikov, ki vključuje uporabnike na vsakem koraku razvojnega cikla projekta. Z uporabo različnih metod za ocenjevanje in testiranje z uporabniki se osredotoča tudi na potrebe, ki so manj pogoste in težje opazne. UCD se ne osredotoča le na razumevanje uporabnikov, ampak tudi na razumevanje njihovih nalog, ciljev ter okolja, v katerem bodo sistem uporabljali [47].

Po ISO 9241-210:2010 [3], ki predpisuje zahteve in priporočila glede načrtovanja sistema za interakcijo s človekom, je pomembnih šest načel:

- Razumevanje uporabnikov, njihovih nalog, ciljev in okolja na določenem področju uporabe.
- Sodelovanje in vključenost aktivnih uporabnikov pri načrtovanju in razvoju izdelkov.

- Oblikovanje izdelkov temelji na in se izboljšuje s pomočjo povratne informacije uporabnikov in uporabniško usmerjenih evalvacij.
- Ponovitev načrtovanja in razvoja z vključitvijo povratnih informacij končnih uporabnikov.
- Načrtovanje in razvoj obravnavata celotno uporabniško izkušnjo izdelkov.
- Raznolikost oblikovalske ekipe in vključevanje večdisciplinarnih spretnosti ter pogledov.

Štiri glavne dejavnosti uporabniško usmerjenega dizajna so [3]:

- Definiranje in razumevanje konteksta uporabe sistema.
- Definiranje potreb uporabnikov in njihove organizacije.
- Zagotavljanje vmesne rešitve s pomočjo prototipov.
- Evalvacija vmesnih rešitev s pomočjo uporabnikov.

Cilj izdelave vmesnikov je ponavadi njihova čim širša komercialna raba, ta pa se seveda doseže, ko jih uporabniki uporabljajo in so z njimi zadovoljni. Še pred nekaj leti je večina podjetij poslovala tako, da so najprej oblikovala produkt, potem pa iskala uporabnike, ki bi jih ta zanimal. Nasprotno se UCD pred začetkom razvoja, v poizvedovalni fazi, osredotoča na iskanje in analizo bodočih uporabnikov. Z analizo uporabnikov lahko ugotovimo njihovo starost, spol, socialni status, izobrazbo, pričakovanja, zahteve in veliko drugih pomembnih lastnosti, ki nam pomagajo pri razvoju uporabniške izkušnje. Za dosego zadovoljstva pri uporabnikih in uspešnost produkta je potrebno poznati tudi raznolikost njihovih nalog, zahtev, potreb, omejitev in želj. UCD torej temelji na globokem poznavanju in raziskovanju uporabnikov, njihovih navad, načina interakcije s produktom in pričakovanj, kako bi produkt moral izgledati in se obnašati [37].

Pristop izdelave vmesnika, ki se osredotoča na uporabnika na vsakem koraku procesa izdelave, je manj tvegan, tj. dovzeten za neuspeh, saj veliko problemov dizajna in funkcionalnosti prepoznamo in lahko popravimo že v začetnih fazah razvojnega cikla (v fazi oblikovanja). Vključevanje uporabnikov lahko tudi pospeši izdelavo vmesnikov, saj nam olajša dileme pri izbirah in odločitvah pri razvoju projekta.

Pomemben del uporabniško usmerjenega načrtovanja je tudi izbira ustrezne metodologije razvoja programske opreme. UCD obstaja v več variacijah in lahko poteka z različnimi pristopi razvoja. Izbira ustreznega pristopa je odvisna od zahtev, ekipe, časa, okolja in drugih dejavnikov, ki direktno ali posredno vplivajo na projekt.

Izbira klasičnega slapovnega modela, ki ga po vrsti sestavljajo koraki postavljanja zahtev, načrtovanja, izvedbe, vpeljave, končnega testiranja in izdaje, bi bil za uporabniško usmerjeno načrtovanje neprimeren, saj je uporabnik vključen le v prvem in zadnjih dveh korakih metode načrtovanja (postavljanje zahtev, končno testiranje in izdaja). Slabost slapovnega modela za UCD je tudi ta, da vsak korak obravnava posamično in se na naslednjega usmeri šele, ko je prejšnji končan. Tak način razvoja pomeni, da prostora za vmesno evalvacijo s pomočjo končnih uporabnikov ni. Zaradi teh pomanjkljivosti so nastale različne modifikacije slapovnega modela za razvoj programske opreme, z dodanimi povratnimi zankami v predhodnih korakih, ki omogočajo vmesno evalvacijo [45].

Za zagotavljanje dobre uporabniške izkušnje je bolj primeren in pogost iterativni način uporabniško usmerjenega načrtovanja. Pri bolj obsežnih in zahtevnih projektih pa se uporablja tudi spiralno načrtovanje. Značilnost obeh je, da se korake slapovnega modela izvaja v več ponovitvah oziroma iteracijah, na koncu vsake pa izda tudi vmesni produkt ali prototip, ki se ga na koraku vrednotenja iteracije lahko testira med uporabniki.

Prototipi zagotavljajo hitrejša in cenejša testiranja, lažje spremembe, pri papirnatih prototipih pa tudi manj časovnih izgub načrtovalcev zaradi nepo-

membnih podrobnosti. S testiranjem papirnatih prototipov lahko dobimo informacije o ustreznosti konceptualnega modela, potrebnih funkcionalnostih, toku navigiranja uporabnikov, ter ustreznosti terminologije in vsebine. Pomankljivost takšnih prototipov je, da ne zagotovijo informacij o učinkovitosti produkta kot so odzivni časi in da testiranje ni enako končni uporabi naprave. Za večjo natančnost v smislu izgleda in uporabe (angl. Look and feel) se zato uporabljajo računalniški prototipi, ki so pravzaprav interaktivna simulacija delovanja programske opreme. Z računalniško simulacijo lahko testiramo vse lastnosti in pomankljivosti papirnatih prototipov, dodatno pa lahko zagotovimo povratne informacije o izgledu produkta in drugih pomembnih lastnosti dobre uporabniške izkušnje [45]. Za testiranje papirnatih prototipov poleg uporabnikov potrebujemo opazovalca, ki test ocenjuje in dela zapiske, ter pomočnika, ki uporabniku predstavi naloge in ga vzpodbuja. Za izdelavo računalniških prototipov pa je na voljo veliko programskih orodij kot so Adobe XD CC, Axure in Mockplus.

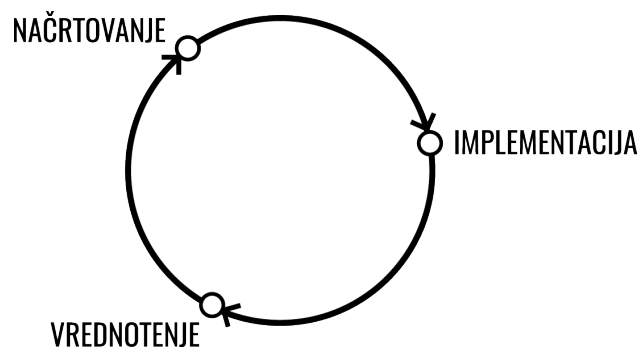
Za razliko od iterativnega, spiralni model pri začetnih iteracijah, ko je verjetnost za napake višja, uporablja manj natančne, poceni prototipe [45]. Slednji v začetnih iteracijah omogočajo, da v primeru nesprejetja ali večjih sprememb, še ne predstavljajo tako velikega stroška.

Pri razvoju programske opreme se vedno bolj uporabljajo agilne metode, ki temeljijo na iskanju rešitev in zahtev med sodelovanjem samoorganiziranih skupin z različnih področij, kar omogoča sprotno vključevanje sprememb zahtev naročnika. Metode so preproste, uporabljajo minimalno količino dokumentacije in omogočajo hiter razvoj, testiranje in delovanje opreme. Tako agilne metode kot uporabniško usmerjeno načrtovanje temeljita na komunikaciji in testiranju z naročniki ter uporabniki v vseh fazah razvoja. Kar filozofiji še bolj povezuje, pa je iterativni način razvoja. [22]

Vsako ponovitev oziroma iteracijo iterativnega razvoja sestavljajo trije koraki:

- Načrtuj.

- Implementiraj.
- Vrednoti.



Slika 2.5: Iterativni razvoj

Predvsem pri razvoju spletnih in mobilnih rešitev velja, da večje število ponovljenih iteracij pomeni višjo izmerjeno uporabnost. Za učinkovitost razvoja je v vsakem koraku potrebno izvesti vsaj 2 iteraciji. Torej izdati vsaj 3 verzije; prvi prototip in dve izboljšavi. Ključni kazalniki uspešnosti se ob 6 iteracijah lahko izboljšajo za kar 233%. [31]

Poglavje 3

Človeško zaznavanje in smernice za oblikovanje uporabniških vmesnikov

Vsebina tega poglavja je v večini povzeta iz knjige *Designing with the Mind in Mind* [18].

3.1 Zaznavanje je pogojeno

Človeški možgani niso prilagojeni na abstraktno razmišljanje in pomnjenje podatkov, ki jih uporaba interaktivnih sistemov, kot so spletne strani in aplikacije, pogosto zahtevajo. Velik delež smernic za uporabnost izhaja prav iz teh človekovih kognitivnih omejitev [30].

Ljudje smo različni, saj različno zaznavamo in dojemamo svet okoli nas. Če neko abstraktno sliko pokažemo različnim osebam, je velika verjetnost, da si bo njen pomen vsaka predstavljala drugače. Naše zaznavanje je namreč pogojeno s tem, kar pričakujemo. Na to, kar pričakujemo, pa vplivajo trije dejavniki:

- Preteklost: pretekle izkušnje.

- Sedanjost: trenutni kontekst.
- Prihodnost: naši cilji.

3.1.1 Zaznavanje na podlagi izkušenj

Kako izkušnje vplivajo na naše dojemanje, ponazarja slika R. C. Jamesa (Slika 3.1). Ob prvem pogledu na sliko težko vidimo kaj drugega, kot naključno razporejene sledi črnih. Šele ko nam nekdo pove, da slika prikazuje psa dalmatinca, lahko naš vizualni sistem sledi organizira v povezano sliko.



Slika 3.1: R.C. James, Pes dalmatinec [39]

Podobno je pri interakciji s pogovornimi okni z gumboma »naprej« in »nazaj«. Iz izkušenj predhodne uporabe pogovornih oken smo navajeni, da je gumb »naprej« navadno na desni, »nazaj« pa na levi strani. V kolikor imamo takih pogovornih oken več zapored, navadno besedila gumbov ne bemo, temveč podzavestno poklikamo na desni gumb »naprej«. Zamenjan vrstni red gumbov bi lahko uporabnika privedel do klika na neželjeni gumb. Konsistenca pozicije in oblike gradnikov ter načina interakcije je zato ena od pomembnih smernic oblikovanja uporabniških vmesnikov. Takšna konsistenca uporabnikom omogoča, da se na sistem hitro navadijo in ga znajo

uporabljati.

Način, kako naše zaznavanje deluje na podlagi izkušnje, se imenuje privajanje. Če smo pogosto izpostavljeni enakim ali zelo podobnim dejavnikom, bo občutljivost zaznavanja na te dejavnike manjša. Eden primerov v povezavi z uporabniškimi vmesniki je pri nalaganju programske opreme na računalnik, kjer pravilnikov in pogojev na pogovornem oknu pogosto sploh ne preberemo, vendar nadaljujemo na naslednji korak.

3.1.2 Zaznavanje na podlagi konteksta

Vizualno zaznavanje ni le proces od spodaj navzgor, kjer bi zaznavali vsak znak, krivuljo, črto, kot, vzorec ali rob posebej in jih nato združili v nam pomenljive objekte. Predvsem pri branju znake zaznavamo s procesom od zgoraj navzdol, kjer zaznavamo s pomočjo izkušenj v dolgotrajnem spominu. Posamezne črke v besedi in besede v stavkih zaznamo odvisno od sosednjih črk v besedi oziroma besed v stavkih (Slika 3.2).



THE CAT

Slika 3.2: Na to, ali preberemo črko »H« ali »A«, vplivajo črke, ki znak obdajajo

Zaznavanje glede na kontekst se ne pojavlja le pri vizualnem zaznavanju, ampak tudi pri ostalih štirih čutih. Če smo pri vizualnem zaznavanju, kot je branje, pozorni na to, kaj obdaja stimulus, smo pri zvočnem zaznavanju pozorni na to, kaj slišimo pred ali po stimulusu. Na to, kar v nekem trenutku vidimo ali čutimo, pa lahko vpliva tudi to, kar v tistem trenutku slišimo in vonjamo ter obratno. Npr. ko na računalniškem zaslonu obvestilo enkrat utripne, poleg utripa pa proizvede dva zvočna signala, se nam zdi, da

obvestilo utripne dvakrat.

3.1.3 Zaznavanje na podlagi ciljev

Na človeško zaznavanje vplivajo tudi cilji v danem trenutku. Ko uporabniki iščejo določeno informacijo po spletni strani, ne berejo in ne pregledujejo vsakega dela strani, da pridejo do iskane informacije. Spletno stran na hitro preletijo in iščejo objekte, ki so po njihovem mnenju povezani z iskanim ciljem.

Poznamo dva mehanizma, kako cilji vplivajo na zaznavanje:

- **Vpliv osredotočanja pogleda:** Zaznavanje je aktivno. Odvisno od konteksta, vedno uporabljamo oči, ušesa, roke in ostale organe, da nas privedejo do cilja. Ko na spletni strani iščemo določeno informacijo, se osredotočimo le na podatke, ki bi nas lahko pripeljali do cilja. Kar ni povezano z našim ciljem, sicer ne ignoriramo, ampak ponavadi sploh ne opazimo, saj možgani v dani situaciji nerelevantne podatke podzavestno filtrirajo.
- **Prilagajanje zaznavanja:** Ko nekaj iščemo, možgani zaznavanje prilagodijo tako, da je še posebej občutljivo na značilnosti objekta, ki ga iščemo. Če imamo v spletni trgovini na isti strani predstavljenih več majic različnih barv in želimo poiskati le modre, se naše zaznavanje prilagodi le na iskanje modre barve, s čimer majice najdemo hitreje.

3.1.4 Smernice prilagojene za zaznavanje

Izogibanje večpomenskosti: V izogib večpomenskosti uporabniških vmesnikov je potrebno s testiranjem med uporabniki zagotoviti, da uporabniki predstavitev informacij na vmesniku razumejo enako.

Konsistenca: Konsistenco lahko zagotovimo s postavitvijo informacij in gradnikov sistema na enak položaj na uporabniškem vmesniku. V primeru spletnih strani naj bodo gradniki z enako funkcionalnostjo prikazani na vseh podstraneh na enakem položaju, z isto barvo, tipografijo, velikostjo in ostalimi vizualnimi lastnostmi.

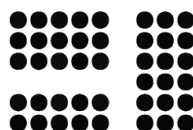
Razumevanje ciljev: Poznavanje uporabnikovih ciljev pri uporabi sistema je pri razvoju ključnega pomena. Zavedati se je potrebno, da so cilji uporabnikov lahko zelo različni in da njihovi cilji vplivajo na to, kako stvari zaznavajo.

3.2 Človeški vid

3.2.1 Gestaltovi principi

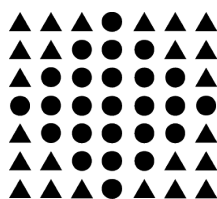
Gestaltovi principi opisujejo, kako je naša vidna zaznava prilagojena na zaznavanje strukture in oblike. S poznavanjem principov pri izdelavi uporabniških vmesnikov lahko vplivamo na vedenje uporabnikov in jim zagotovimo intuitivno ter zadovoljivo uporabniško izkušnjo. Opisal bom šest najbolj pogosto uporabljenih principov, ki vplivajo na uporabnost uporabniških vmesnikov:

- **Bližina:** Princip bližine opisuje, da človekov vid objekte, ki so razporejeni blizu drug zraven drugega, zazna kot združeno skupino objektov. Na (Sliki 3.3) tako vidimo črne pike, ki jih zaradi medsebojnega razmika dojemamo kot tri skupine pik. Princip je neposredno povezan tudi z gradnjo uporabniških vmesnikov. Najpogosteje se uporablja pri gradnji pogovornih oken in na spletnih straneh, kjer se namesto ločevalnih črt za ločevanje podskupin uporabi razmike med vsako podskupino objektov.
- **Podobnost:** Princip podobnosti opisuje, da objekte, ki so med seboj podobni, zaznavamo kot del iste podskupine. Objekte iz iste podsku-



Slika 3.3: Primer principa bližine

pine ponavadi tudi dojemamo, kot da imajo podobne funkcionalnosti. Na zaznavanje podobnosti lahko vplivajo oblike, barve ali ikone objektov. Spletne predstavitve princip pogosto uporabljajo za ločevanje obiskanih in neobiskanih povezav in za ločevanje funkcionalnosti, ki jih ima posamezen del na strani.



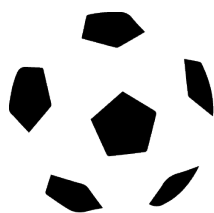
Slika 3.4: Primer principa podobnosti

- **Kontinuiteta, nepretrganost:** Princip opisuje, da objekte razporejene na ravni črti ali na krivulji, zaznavamo kot povezane oz. sorodne. Na Sliki 3.5 pike, ki so razporejene na krivulji, bolj povezujemo s črnimi pikami, ki so prav tako na krivulji, kot z rdečimi, ki so postavljene vodoravno. Kontinuiteta je pogosto uporabljena v spletnih trgovinah, kjer so sorodni izdelki prikazani na vodoravni črti.



Slika 3.5: Primer principa nepretrganosti

- **Zaprta oblika:** Princip zaprtosti opisuje, da ob pogledu na kompleksno postavitev vizualnih elementov, iščemo le eno obliko ali vzorec, ki nam je poznan/a. Z drugimi besedami objekte, kot so liki, črke, slike in podobno zaznamo kot celoto, čeprav je njihov prikaz pomanjkljiv. Na Sliki 3.6 kljub temu, da je prikazanih le šest nepovezanih večkotnikov, prepoznamo, da gre za okroglo obliko nogometne žoge.

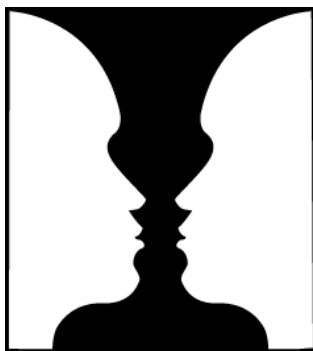


Slika 3.6: Primer principa zaprtosti oblike

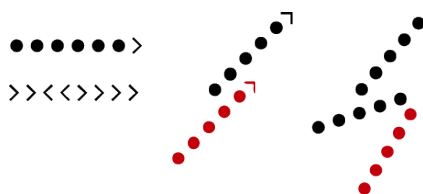
- **Figura / ozadje:** Ko uporabnik prvič obiše spletno stran, ponavadi najprej poskuša razločiti, kaj spada v ospredje in kaj je del ozadja. Princip figura / ozadje opisuje sposobnost človeškega zaznavanja, da vizualne objekte med seboj ločimo glede na podlago oziroma ozadje, na katerem so postavljeni. Z drugimi besedami, ločimo objekte predstavljene v ozadju in v ospredju. Zmožnost ločevanja med njimi je pogojena z uporabo kontrasta, vizualnega učinka dvignjenosti gradnikov in hierarhije podatkov.
- **Princip skupne usode:** Opisuje, da objekte zaznavamo kot del iste skupine, če se premikajo ali spreminjajo v isti smeri ali v isto obliko. Pri izdelavi spletnih strani s prekinitvijo sinhronega premikanja lahko preusmerimo uporabnikovo pozornost na druge objekte.

3.2.2 Iščemo in uporabljamo strukturo

Ko brskamo po internetnih straneh ali se po njih poskušamo orientirati, navadno ne beremo vsake besede, da pridemo do cilja. Strani hitro pregledujemo



Slika 3.7: Primer principa figura / ozadje



Slika 3.8: Primer principa skupne usode

in tako iščemo za nas ustrezno informacijo. Če želimo, da je uporabniški vmesnik uporaben, je zato pomembno, da podatke na vmesniku predstavimo strukturirano, kar omogoča enostavno navigacijo in berljivost podatkov. Vse hiperpovezave z navodili (Slika 3.9) se začnejo z istim besedilom (>How to...<), kar otežuje iskanje navodil.

Med pomembnejšimi cilji pri strukturiranju podatkov je vizualna hierarhija. Vizualna hierarhija je metoda oblikovanja informacij, ki zagotavlja, da bo informacija ustrezno sporočena in razumljena. Polno strukturiranost podatkov dosežemo, ko podatke oziroma besedilo najprej strukturiramo vsebinsko, na drugi stopnji pa ga s pomočjo Gestaltovih principov prevedemo v vizualno hierarhijo, ki jo uporabnik enostavno zazna in razume. Na preprostem primeru (Slika 3.10), kjer so podatki strukturirani, delujejo principi bližine, simetrije in podobnosti. Za razliko od govora, branje človeku ni naravno, zato je pomembno, da so besedila predstavljena tako,



Renewals, Duplicates, and Information Changes for Driver Licenses and/or ID Cards

- [How to renew your driver license in person](#)
- [How to renew your driver license by mail](#)
- [How to renew your driver license by Internet](#)
- [How to renew your instruction permit](#)
- [How to apply for a duplicate driver license or identification \(ID\) card](#)
- [How to change your name on your driver license and/or identification \(ID\) card](#)
- [How to notify DMV of my change of address](#)
- [How to register for the organ donor gift of life program](#)

Slika 3.9: Primer slabše berljive predstavitve podatkov

da jih uporabnik čim lažje bere ali poišče.

Z vizualno hierarhijo:

- Podatke razdelimo v različne odseke in večje dele informacije razdelimo v več odstavkov.
- Naslove vsakega odstavka ustrezno ločimo od besedila odstavka.
- Odstavke razdelimo tako, da so tisti na višjem nivoju bolj poudarjeni kot na nižjem.

Besedilo, ki vsebuje za uporabnika pomembne informacije, namesto predstavitve v navadni nestrukturirani obliki, uredimo in prikažemo tako, da uporabnik pomembne informacije brez težav zazna in razume. (Slika poletov)

Vizualna hierarhija ni pomembna le pri členjenju daljših besedil, ampak tudi pri besedilu in gradnikih v pogovornih oknih in interaktivnih obrazcih. Večjo strukturiranost podatkov pri razvoju uporabniških vmesnikov omogočajo gradniki za vnos specifičnih podatkov, kot so datum, elektronski naslov, kraj itd. Za vnos datuma namesto treh vnosnih polj za dan, mesec in leto, na primer uporabimo pop-up koledar, kjer uporabnik datum vnese z enim, namesto s tremi vnosi.

Unstructured:
You are booked on United flight 237, which departs from Auckland at 14:30 on Tuesday 15 Oct and arrives at San Francisco at 11:40 on Tuesday 15 Oct.

Structured:
Flight: United 237, Auckland → San Francisco
Depart: 14:30 Tue 15 Oct
Arrive: 11:40 Tue 15 Oct

Slika 3.10: Nestrukturiranost podatkov (zgoraj) in strukturiranost podatkov (spodaj)

3.2.3 Periferni vid je slab

Za razliko od digitalnih kamer, ki imajo receptorje za zajem slike enakomerno porazdeljene v matriki, kar rezultira v konstantno prostorsko ločljivost po celotnem okvirju zajete slike, človeški vid deluje drugače.

Čepki so veliko bolj gosto porazdeljeni na sredini kot na robovih očesne mrežnice, ta del se imenuje očesno brezno ali fovea in predstavlja le 1 % površine celotne mrežnice, kljub temu pa ji možganska skorja posveča kar 50 % zaznavanja. Posledično je človeški vid veliko bolj oster na sredini vidnega polja, ki pa predstavlja le 1 % celotnega vidnega polja. Ostalih 99 % vidnega polja imenujemo periferni vid, kjer je ločljivost slaba. Vseeno je za človeški vid zelo pomemben, saj usmerja našo osredotočenost pogleda in dobro zaznava premikanje. Ko na primer na določenem izdelku iščemo rok trajanja, bi potrebovali veliko časa, če bi datum iskali z branjem celotnega besedila. Periferni vid nam pomaga, da hitreje opazimo strukturo zapisa datuma.

Čeprav v danem trenutku z visoko ločljivostjo vidimo le 1 % vidnega polja, se nam zdi, da temu ni tako. Razlog za to je, da oko stalno premikamo, kar nam daje občutek, da celotno vidno polje vidimo z visoko ločljivostjo.

Poleg usmerjanja centra pozornosti pogleda in zaznavanja premikov, periferni vid omogoča tudi boljši vid v temnem okolju. Za razliko od čutnih celic paličic, ki se najbolj odzivajo v svetlem okolju, čepki v nizki svetlobi ne delujejo dobro, zato tudi v temnih prostorih prevlada delovanje paličic.

Delovanje paličic v temnih prostorih se imenuje skotopični vid (angl. scotopic vision). Ker v očesnem breznu ni paličic, v slabo osvetljenih prostorih predmete bolje vidimo, ko jih ne gledamo direktno oziroma jih gledamo s perifernim vidom.

Pri oblikovanju uporabniških vmesnikov je poznavanje značilnosti perifernega vida pomembno predvsem pri oblikovanju obvestil za napake. Pomembno vlogo pa ima tudi pri oblikovanju strani in aplikacij oziroma za uspešno trženje produktov.

Pogled v visoki ločljivosti predstavlja približno 1-2 centimetra na računalniškem zaslonu (če predpostavimo, da je od zaslona normalno oddaljen). Pri prikazovanju sporočil za napako ali stanje sistema je zato pomembno, da so ustrezno postavljena in oblikovana, in sicer glede na osredotočenost pogleda uporabnika v trenutku, ko se napaka pojavi. Na primer, ko uporabnik narobe izpolni podatke za prijavo na spletni strani ali aplikaciji, je pomembno, da sistem napako izpiše blizu njegove pozornosti, tj. ob gumbu »Prijava«.

Iz zgornjih lastnosti človeškega vida izhajajo tudi naslednja navodila za zagotavljanje vidljivosti napak:

- **Napake izpišemo, kjer je uporabnikova pozornost.**
- **Napake izpisujemo zraven vzroka za napako.**
- **Uporabljamo simbole za napake.**
- **Rdečo barvo uporabljamo le za napake:** Če rdeča barva predstavlja podobo podjetja, za katero izdelujemo spletno stran, za napake uporabimo drugo barvo in jim dodamo simbol. V takih primerih lahko uporabimo tudi druge metode za zagotavljanje vidljivosti kot so pop-up, zvok, premik/utripanje, vendar zmerno.

3.2.4 Zaznavanje barv

Človeško zaznavanje barv ima svoje pozitivne in tudi negativne lastnosti, ki lahko vplivajo na težavnost uporabe uporabniškega vmesnika. Poseben primer so ljudje z barvno slepoto, za katere vmesnik lahko posebej prilagodimo.

Velik vpliv na to, kako zaznavamo barve, ima očesna mrežnica. Ta se nahaja na zadnji strani očesa in je sestavljena iz dveh vrst receptorjev, ki so v obliki čepkov in paličic. Čepki so zadolženi za zaznavanje barv in so občutljivi na tri različne frekvence svetlobe, paličice pa so zadolžene za zaznavanje splošne jakosti svetlobe. Danes telo za vidno zaznavanje le redko uporablja paličice, saj je oko navajeno na osvetljene prostore, ki hitro zapolnijo ves razpon njihove občutljivosti.

Vid je tako večinoma odvisen od treh vrst čepkov, ki zaznavajo:

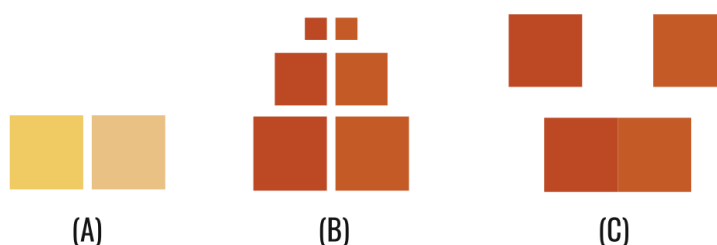
- **Nizko frekvenco:** Ti čepki so najbolj občutljivi na srednjo (rumeno) in nizko (rdečo) frekvenco.
- **Srednjo frekvenco:** Ti čepki na svetlobo reagirajo od visokih frekvenc modre svetlobe, do nižjih srednje-frekvenčnih oranžne in rumene.
- **Visoko frekvenco:** Ti čepki so najbolj občutljivi na svetlobo zgornjega dela vidnega spektra svetlobe – modre in vijolične. Poleg zgornjega dela spektra pa so rahlo občutljivi tudi na srednje frekvence, kot je zelena. Od vseh treh čepkastih receptorjev so ti najmanj občutljivi in jih je na mrežnici najmanj.

S pomočjo procesa odštevanja signalov iz čepkov, naši možgani signale združijo v širok spekter barv, ki jih zaznamo. Iz zgornjega lahko sklepamo, da je naše oko najmanj občutljivo na modro in vijolično barvo, v primerjavi z drugimi.

Kaj vpliva na razlikovanje barv

Na to, kako dobro razlikujemo barve, lahko vpliva tudi, kako so barve predstavljene, kar moramo seveda upoštevati pri gradnji uporabniških vmesnikov. Na naše zaznavanje barv vplivajo trije dejavniki: barvna nasičenost, velikost in razmik med obarvanimi objekti (Slika 3.11).

Manj kot so objekti barvno nasičeni oziroma bolj kot so blede, težje jih med seboj razlikujemo po barvi. Obarvane objekte med seboj težje razlikujemo tudi, če so objekti manjših dimenzij in če so med seboj preveč oddaljeni.

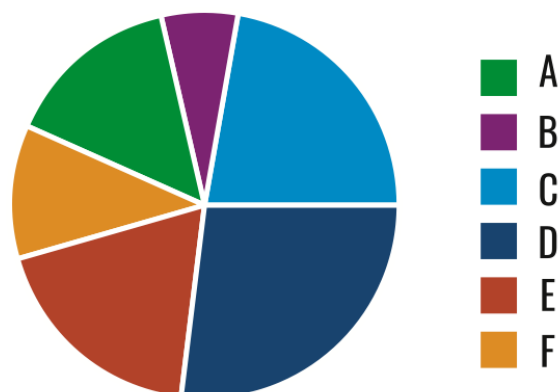


Slika 3.11: Vpliv nasičenosti (A), velikosti (B), in oddaljenosti (C) objektov na zaznavanje njihovih barv

Zgoraj opisane značilnosti zaznavanja največkrat upoštevamo pri prikazu podatkov z grafikoni, kjer z barvami ločujemo različne dele celote, in pri prikazovanju njihovih legend.

Do težjega razlikovanja med barvami lahko pride tudi pri izbiri barv za hiperpovezave, katere je uporabnik že obiskal, in hiperpovezave, ki so še neobiskane. Ker so pogosto prikazane v obliki seznama, tesno ena ob drugi, z modro in vijolično barvo, uporabnik težje razlikuje med obiskanimi povezavami.

Pri gradnji uporabniških vmesnikov moramo upoštevati tudi napravo, na kateri se bo vmesnik uporabljal, in okolje, v katerem se uporablja. Če želimo prikazati določeno barvo, je zato pomembno, da prikazovanje barve preizkusimo na več različnih napravah / zaslonih, saj odvisno od tehnologije, gonilnikov in barvnih nastavitvev drugače prikazujejo barve.



Slika 3.12: Primer grafikona z dobrim razlikovanjem barv

Barvna slepota

Da je uporabniški vmesnik uporabniku prijazen in dostopen, pomeni, da je prilagojen tudi za osebe z barvno slepoto oziroma daltonizmom. Pri osebah z barvno slepoto se barvni signali med seboj ne odštevajo pravilno in posledično določenih barv ne razlikujejo. Ker ima kar 8% moških in 0.5% žensk vsaj eno izmed vrst barvne slepote [6], je pomembno, da vmesnike, kjer z barvami nekaj sporočamo, ustrezno prilagodimo. Najbolj pogosta vrsta barvne slepote je rdeče-zelena, pri kateri ljudje ne razlikujejo med temno rdečo in črno, modro in vijolično, zeleno in temno oranžno, ter svetlo zeleno in belo barvo. Pri oblikovanju vmesnika se moramo takim barvnim kombinacijami zato čimbolj izogniti oziroma jih ne smemo uporabljati za ločevanje med objekti vmesnika.

Navodila za izbor barv

Razlikuj barve z nasičenostjo, svetlostjo, barvnim tonom in temperaturo: Kontrast med barvami naj bo čim večji, kar dosežemo z velikim kontrastom v svetlosti. To lahko preverimo s pretvorbo barv v sivine, ki morajo biti različno temne na lestvici od bele do črne. Nekateri barvni

kontrasti so lahko moteči (npr. močan toplo-hladni kontrast) in jih je zato treba uporabljati zmerno in premišljeno (kot akcent na manjših ploskvah), ali pa so premalo izraziti (npr. kontrast barvne kvalitete oz. razlika v nasičenosti) in jih zato uporabljamo v kombinaciji z drugimi.

Uporaba lahko razlikovalnih barv: Rdeča, zelena, rumena, modra, črna in bela so barve, katere človeško oko najlažje razlikuje. Vse ostale barve, ki so skupek več barvnih kanalov, oko težje razlikuje.

Izogibaj se barvnim kombinacijam, ki otežujejo razlikovanje barv barvno slepim: Izogibanje kombinacijam, kot so temno rdeča in črna, temno rdeča in temno zelena, modra in vijolična, svetlo zelena in bela.

Za razlikovanje ne uporabljaj le barve: Za razlikovanje med gradniki in objekti na uporabniškem vmesniku poleg barve uporabljaj tudi druge načine kot so ikone, simboli, različne velikost itd. Za razlikovanje vrednosti na grafikonih in legendah poleg ustreznih barv lahko uporabimo tudi vzorce in teksture.

Med seboj loči komplementarne barve: Uporaba komplementarnih barv v plasteh ali na sosednjih ploskvah lahko povzroči neprijeten občutek utripanja, zato se ji izogibamo.

Prilagojenost vida na kontrast

Ker se barve v naših možganih predstavijo s pomočjo odštevanja, je naš vid veliko bolj občutljiv na razlike med barvami in svetlobo, kot na absolutno jakost svetlobe. Z drugimi besedami, naš vid je prilagojen na zaznavanje kontrasta. Moč svetlobe tako ne vpliva na to, kako zaznamo barvo, zato rumen tulipan vidimo enake barve v senci kot na soncu.

3.3 Človeški spomin

Tako kot vidni sistem, ima tudi človeški spomin svoje dobre in slabe lastnosti, ki vplivajo na uporabo interaktivnih sistemov. Poznamo senzorični, kratkotrajni ali delovni spomin in dolgotrajni spomin. Dolgotrajni spomin si lahko predstavljamo kot trdi disk računalnika, kratkotrajni pa kot registre v procesorju in spomin RAM.

3.3.1 Senzorični spomin

Vsak človeški organ za zaznavanje ima svoje senzorične receptorje, ki sprejemajo veliko količino informacij. S pomočjo pozornosti lahko nekatere informacije posreduje v kratkotrajni spomin.

Najbolj pogosto sprejemanje informacij je preko vidnih in slušnih senzoričnih receptorjev. Vizualne informacije so v senzoričnem spominu kodirane kot fizična slika (dolžina, ukrivljenost, robovi, barve, intenziteta, frekvenca), zvočne pa kot fizični zvok (zvok, barva zvoka, jakost, frekvenca). Za razliko od vizualnih informacij, ki jih pozabimo v 200 ms (70 – 1000 ms), zvočne informacije v spominu ostanejo dlje, 1500 ms (900 – 3500 ms).

Čeprav je sledenje vizualnim kanalom lažje kot sledenje zvočnim (vizualna dominanca), slednji v spominu ostanejo dlje. Ko med pogovorom s prijateljem nismo pozorni na njegove besede, se nekaj sekund pozneje vseeno lahko spomnimo, kaj je povedal.

3.3.2 Dolgotrajni spomini

Spomini, ki jih posedujemo v dolgoročnem spominu, so pravzaprav množica med seboj povezanih nevronov. V človeških možganih je približno 86 milijard nevronov, kar pomeni, da med njimi lahko obstaja veliko kombinacij. Ko se spominjamo določenega trenutka, se aktivira enaka množica nevronov, kot je bila vzdražena ob nastanku trenutka. Večino spominov iz našega kratkotrajnega spomina pozabimo. Spomin iz kratkotrajnega spomina v dolgotrajni

preide šele ob ponovitvi določenih delov informacije.

Množica nevronov, ki se aktivira ob določenem dogodku, je odvisna tako od načina zaznavanja, kot od konteksta, v katerega smo ob dogodku postavljeni. Ko se npr. sprehajamo po ulici med grmenjem in udarom strel, se nam vključi drugačna množica nevronov, kot če bi se po isti ulici med nevihto peljali z avtom. Čeprav je kontekst zelo podoben in je razlika med dogodkoma le način premikanja, se prav zaradi slednjega vzdražijo drugačni nevroni. Zavedamo se namreč, da smo v avtomobilu pred udarom strel varni. Bolj kot sta si dogodka podobna, torej več kot imata skupnega konteksta in načina zaznave, večje je število istih nevronov iz njunih množic, ki se vzdražijo ob dogodku. Na spomin vplivajo tudi emocije, zato se dogodkov, ki se zgodijo prvič, spominjamo bolje, kot dogodkov, ki se zgodijo n-tič.

Dolgotrajni spomin ima ogromno kapaciteto. Nekateri znanstveniki trdijo, da informacije v dolgotrajnem spominu nikoli ne izginejo, pač pa le zbledijo poti oziroma se zniža vzdraženost nevronov, ki vodijo do določene informacije. S primerjavo s področja računalništva, si ga lahko predstavljamo kot uporabo orodja za stiskanje podatkov z veliko izgubami.

Implikacije dolgoročnega spomina za oblikovanje uporabniških vmesnikov

Ljudje že od nekdaj uporabljamo pripomočke, kot so zgodbe, knjige, zapiski, telefonski imenik, itd., ki nam pomagajo pri dolgoročnem pomnjenju. Čeprav se tehnologija hitro spreminja in se interaktivni sistemi razvijajo tako, da zmeraj manj obremenjujejo uporabnikov spomin, je pri določenih interakcijah še vedno potrebno pomnjenje.

Naravna avtentikacija: Problem avtentikacije pri večini današnjih spletnih aplikacij je, da zahteva pomnjenje. Geslo si seveda lahko zapišemo na list papirja, vendar si moramo nato zapomniti tudi, kam shranimo list, še bolj pa je vprašljiva varnost.

S pomočjo za izgubljena gesla uporabnikov olajšamo kritičnost pomnje-

nja, še vedno pa ta, zaradi varnosti, zahteva odgovore na vnaprej zastavljena vprašanja in posledično dodatno pomnjenje. Varnostno vprašanje, kot je: »Kam ste hodili v šolo?«, je lahko problematično, saj je uporabnik lahko hodil na več šol, kar pomeni, da si mora zapomniti, katero je izbral v odgovoru.

Ena od implikacij je, da namesto vnaprej določenih vprašanj za obnovitev gesla, uporabniku pustimo, da lahko sestavi svoja.

Če tehnologija omogoča, uporabimo avtentikacijo s prstnim odtisom ali prepoznavanjem obraza, kjer uporabniku ni potrebno pomnjenje.

Konsistenca: S konsistenco načina dostopanja do funkcij sistema in uporabe le teh zmanjšamo število informacij, ki si jih mora uporabnik zapomniti za uporabo sistema.

Če bi naslednji Windows operacijski sistem povsem spremenil način uporabe njegovih funkcionalnosti, bi se uporabnik, navajen na prejšnjo verzijo sistema, vsega moral naučiti od začetka.

3.3.3 Kratkotrajni spomin

Kratkotrajni spomin oziroma delovni spomin je pravzaprav kombinacija zaznavanja, pozornosti in spominjanja iz dolgotrajnega spomina. Vsako od človeških zaznavnih čutil oziroma receptorjev ima svoj nekakšen kratkotrajni senzorični spomin, ki se uporabi kot posledica odvečnih nevronske aktivnosti, ko zaznavanje dražljajev preneha. Dokler te nevronske aktivnosti potekajo, so na voljo kot vhodni podatki za človekovo pozornost in mehanizme za spomin, kateri vhodne podatke iz različnih zaznavnih čutov združijo, se osredotočijo na določene dele teh podatkov in nekatere shranijo v dolgotrajni spomin.

Drugi način, kako podatki vstopajo v delovni spomin, je tudi preko spominjanja oziroma dostopanja do informacij v dolgotrajnem spominu. Kapaciteta delovnega spomina je 4 +1 kosov informacije, kjer kos predstavlja enoto zaznavanja ali spomina oziroma aktivacijo izkušnje iz preteklosti. Kosi v delovnem spominu ostanejo približno 10 sekund.

Delovni spomin torej združuje informacije iz vseh naših senzoričnih receptorjev in dolgotrajnega spomina in je pravzaprav naše združeno osredotočenje pozornosti; vse, česar se zavedamo v določenem trenutku.

Implikacije kratkoročnega spomina za oblikovanje uporabniških vmesnikov

Delovni spomin človeka je v tesni povezavi z njegovo uporabo uporabniških vmesnikov. Ker ima delovni spomin majhno kapaciteto, veliko smernic/principov oblikovanja temelji na lažji zapomnljivosti ključnih informacij ob uporabi sistema.

Načini (angl. Modes): Uporaba načinov interaktivnemu sistemu omogoča več funkcionalnosti z manjšim številom načinov interakcije. Tako npr. v risalnem programu s klikom in premikom miške premaknemo narisani objekt, ob vključenem načinu za risanje krogov, pa z enako interakcijo narišemo krog. Pri uporabi načinov je pomembno, da zagotovimo dovolj očitno povratno informacijo o tem, v katerem načinu se uporabnik nahaja.

Rezultati iskanja (angl. Search results): Kadar s pomočjo iskalnika na določeni spletni strani iščemo želene informacije, ponavadi v iskalno okno vpisujemo izraze in besede, nato pa dobljene rezultate iskanja pregledujemo in iščemo povezavo z našim ciljem. Niz besed, ki smo jih iskali, pogosto hitro pozabimo, zato je pomembno, da poleg rezultatov iskanja prikažemo tudi iskalni niz.

Poziv k ukrepanju (angl. Calls to action): Pri oblikovanju interaktivnih sistemov je priporočljivo, da ima vsaka stran sistema le en jasn cilj oz. ukrepanje, ki ga mora uporabnik izvesti na svoji poti do cilja. S pomočjo grafičnega prikazovanja napredka po zaporednih ukrepih uporabniku olajšamo uporabo sistema.

Navodila (angl. Instructions): Pri kuhanju z receptom si ponavadi ne najprej zapomnimo vseh postopkov in se nato lotimo kuhanja, ampak imamo recept med kuhanjem pri sebi in med kuhanjem po potrebi gledamo recept. Podobno je pri zagotavljanju navodil pri interaktivnih sistemih. Ta morajo biti uporabniku na voljo skozi celoten proces uporabe.

Globina navigacije (angl. Navigation depth): Navigacija v produktih programske opreme ali spletnih straneh mora jasno prikazovati, kje se nahajamo, če ne želimo preobremenjevati uporabnikovega spomina.

Home / T-SHIRTS / Pineapple Cube Tee

Slika 3.13: Prikaz globine navigacije v spletni trgovini Shop-Esquire

3.3.4 Prepoznavanje proti pomnjenju

Zaznavanje oz. dražljaji, ki jih sprejmejo senzorični receptorji, potujejo do možganov, kjer potekajo kompleksne nevronske aktivnosti. Te so odvisne tako od lastnosti, ki jih zaznamo, kot od konteksta, v katerem zaznavamo. Ponovitev določenih vzorcev nevronskih aktivnosti v prihodnje omogoča lažjo aktivacijo istih vzorcev. Medsebojne povezave med vzorci nevronov se skozi čas razvijejo tako, da en vzorec lahko aktivira drugega. Torej vzorec nekih spominov lahko vzpodbudi nove spomine.

Če možgani sprejmejo dražljaj, ki je podoben prejšnjemu zapisanemu v dolgotrajnem spominu, poleg tega pa je njun kontekst dovolj podoben, dražljaji v možganih povzročijo podobne nevronske aktivnosti. Tako deluje **prepoznavanje**, ki je za razliko od spominjanja veliko hitrejše. Spominjanje je počasnejše, saj zahteva aktivacijo starih nevronskih vzorcev v dolgotrajnem spominu, brez podobnih zunanjih dražljajev, ki bi spodbudili podobne nevronske aktivnosti.

Implikacije prepoznavanja pri oblikovanju uporabniških vmesnikov

Lahkotnost, s katero prepoznamo stvari, namesto da se jih spominjamo, je osnova grafičnih uporabniških vmesnikov.

Poglej in izberi namesto spomni se in napiši: To pravilo je bilo glavno vodilo za razvoj grafičnih vmesnikov in prevlado nad vmesniki z ukazno vrstico (CLI). Namesto da si morajo uporabniki zapomniti ukaze za izvedbo določene operacije, grafični vmesniki omogočajo prikaz ukazov, med katerimi uporabnik lahko izbere.

Uporaba slik, kjer je mogoče: Ljudje hitro prepoznamo slike, zato uporaba slik pri ikonah, obvestilih o napakah in podobnih elementih omogoča uporabniku hitrejšo prepoznavanje in učenje funkcionalnosti. Uporaba podob iz realnega sveta je prav tako priporočljiva, saj od uporabnika ne zahteva dodatnega pomnjenja.

Uporaba predoglednih sličic: Uporaba predoglednih sličic (angl. Thumbnail) omogoča uporabnikom pregled nad več elementi hkrati, kot so podatki, zgodovina iskanja, izbire. Ni nujno, da so predogledne slike dejanski manjši prikazi originalnih slik. Uporaba le nekaj značilnosti iz originalne slike je ponavadi dovolj, da se nam vzdražijo isti nevronske vzorci in manjšo sličico prepoznamo.

Pogosteje uporabljene funkcionalnosti naj bodo bolj vidne: Funkcionalnosti interaktivnega sistema, ki bodo bolj pogosto uporabljene, naj bodo prikazane bolj vidno, da s tem uporabnikom omogočamo prepoznavanje. Funkcionalnosti, ki so manj pogoste, ponavadi uporabljajo uporabniki z več izkušnjami s sistemom, ki potrebujejo manj pomoči pri prepoznavanju.

Poglavje 4

Metode za ocenjevanje uporabniških vmesnikov

Na področju uporabniške izkušnje obstaja veliko metod in tehnik ocenjevanja in zagotavljanja uporabnosti (MOU metode), od starejših laboratorijsko nadzorovanih metod do novejših avtomatiziranih in oddaljenih metod. Še pred nekaj leti so si testiranje in raziskovanje uporabnosti lahko privoščila le večja podjetja, kot so Apple in Microsoft, saj je predstavljalo velik strošek. Metoda opazovanja premika oči je npr. zahtevala drago strojno in programsko opremo, poleg tega pa so jo izvajali le eksperti. Danes za izvedbo metode potrebujemo le spletno kamero in eno izmed številnih programskih oprem, ki so na voljo.

S hitrim razvojem orodij in programske opreme je evalviranje uporabniške izkušnje bolj dostopno, hitrejše, s tem pa bolj primerno za agilne pristope. Razlog za to je tudi v avtomatizaciji, ki danes prevladuje predvsem v obliki zajemanja podatkov (spletne analitike, pozornost pogleda, ...) in avtomatizirani analizi le-teh.

Kljub temu, da lahko z avtomatizacijo hitreje ocenjujemo vmesnike in vedno bolj natančno predvidimo obnašanje uporabnikov, pa poglobitve evalvacijske metode še vedno vključujejo testiranje s pomočjo pravih končnih uporabnikov [56].

V realnosti je za vsak načrtovan projekt potrebna analiza in izbira najbolj primernih metod, orodij in pristopov evalvacije. Večje število izvedenih metod in optimiziranje vsake lastnosti pomeni bolj izpopolnjen in uporaben produkt, vendar je zaradi različnih omejitev ponavadi potrebno iskanje kompromisov in izbira le najbolj ključnih.

Najbolj pogosti omejitvi pri izbiri metod in tehnik so finančna sredstva in čas, dodatno pa lahko na izbiro metod vplivajo tudi:

- Stopnja razvoja, v kateri se projekt nahaja.
- Izkušnje v ekipi.
- Zahteve uporabnikov.
- Način načrtovanja.
- Ekonomske potrebe.
- Težavnost uporabe.
- Uporabnost.
- Primernost (učinkovitost in praktičnost).
- Estetika.
- Cena.
- Podpora.
- Tip pridobljenih podatkov.

Glede na pristop evalvacije, lahko metode MOU razdelimo v pet razredov:

- Testiranje uporabnosti (angl. usability testing).
- Pregled eksperta (angl. inspection).

- Poizvedba (angl. inquiry).
- Analitično modeliranje (angl. analytical modeling).
- Simulacija (angl. simulation).

Metode MOU v razredih testiranja, pregledovanja in poizvedbe z zajemanjem informacij so primerne za ugotavljanje specifičnih uporabnostnih problemov in za pridobivanje splošnih ocen uporabnosti. Analitično modeliranje in simulacija pa temeljita na bolj inženirskem pristopu, ki pomaga ocenjevalcem predvideti uporabnosti vmesnika. S temi metodami evaluatorji oziroma cenilci lahko predvidijo in simulirajo obnašanje uporabnikov, morebitne napake in ostala kvantitativna merila [56]. V nadaljevanju se bom posvetil metodam prvih treh razredov.

Testiranje uporabnosti se osredotoča na opazovanje uporabnikov pri uporabi vmesnika in izvajanju podanih nalog, pri katerih ga spremlja eden ali več opazovalcev, ki si morebitne napake in izboljšave beležijo. Obstajata dva pristopa za testiranje uporabnosti s pomočjo uporabnikov: vodeno ali nadzorovano in prosto ali nenadzorovano.

Pregled eksperta ponavadi zahteva visoko izkušnost pregledovalca, ki vmesnik glede na smernice, pravila, navodila in svoje izkušnje pregleda in išče probleme uporabnosti.

Podobno kot pri testiranju uporabnosti, se **metode poizvedbe** osredotočajo na subjektivna mnenja sodelujočih opazovanih uporabnikov, ki ponavadi s pomočjo intervjujev, anket, vprašalnikov, fokusnih skupin in drugih metod, skozi celoten razvoj podajajo za projekt pomembne informacije o uporabnosti produkta in vmesnih izdaj/prototipov.

Metode testiranja uporabnosti in poizvedbe lahko predstavimo na tridimenzionalnem grafu (Slika 4.1) [41], ki metode razdeli na:

- **Odnosne in vedenjske metode:** Odnosne metode se osredotočajo na subjektivna mnenja in poglede testiranih uporabnikov, medtem ko pri vedenjskih metodah opazujemo uporabnikova dejanja pri uporabi izdelka. Večina metod za zagotavljanje uporabnosti se sicer osredotoča na

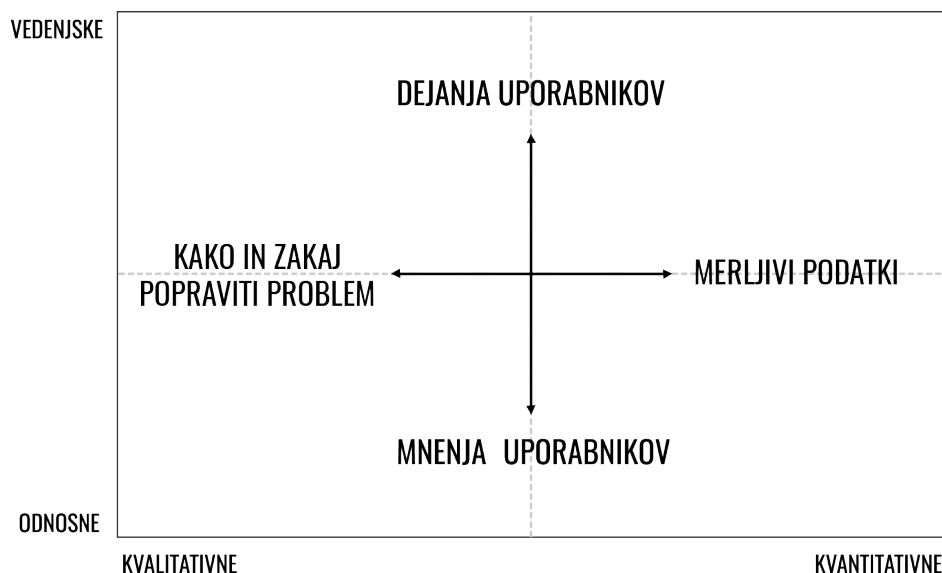
podatke, pridobljene s pomočjo vedenjskih metod, vseeno pa so tudi odnose za oblikovalce, razvijalce in predvsem za trženje produkta lahko zelo pomembne, saj pomagajo pri zbiranju informacij za pomoč pri oblikovanju informacijske arhitekture in pridobivanju zahtev uporabnikov in njihovih povratnih informacij. Vedenjske metode zagotavljajo bolj konkretne informacije o tem, kje je potrebno produkt še izboljšati in kako poteka uporabnikova interakcija z njim.

Nekatere metode pa se osredotočajo tako na vedenje kot na mnenja uporabnikov. V ta segment spada tudi metoda terenske oziroma kontekstne raziskave, kjer se bodoče uporabnike produkta opazuje v njihovem delovnem okolju in od njih zbira informacije o trenutnih potekih dela ter uporabi tehnologij.

- **Kvalitativne in kvantitativne metode:** Kvalitativne metode podatke o problemih uporabnosti vmesnika pridobijo z neposrednim opazovanjem uporabnikov. Kvantitativne pa navadno zagotovijo podatke s posrednim opazovanjem, z uporabo različnih orodij za meritev, npr. z uporabo anketnih obrazcev ali orodij za analitike. Slednje ponavadi zajemajo veliko število podatkov, zato so ugotovitve, pridobljene s kvantitativnimi metodami, ponavadi produkt matematičnih analiz.

Kvalitativne metode testiranja običajno zahtevajo manjšo skupino sodelujočih uporabnikov (5-8), kvantitativne metode pa pogosto več kot 30 sodelujočih [9].

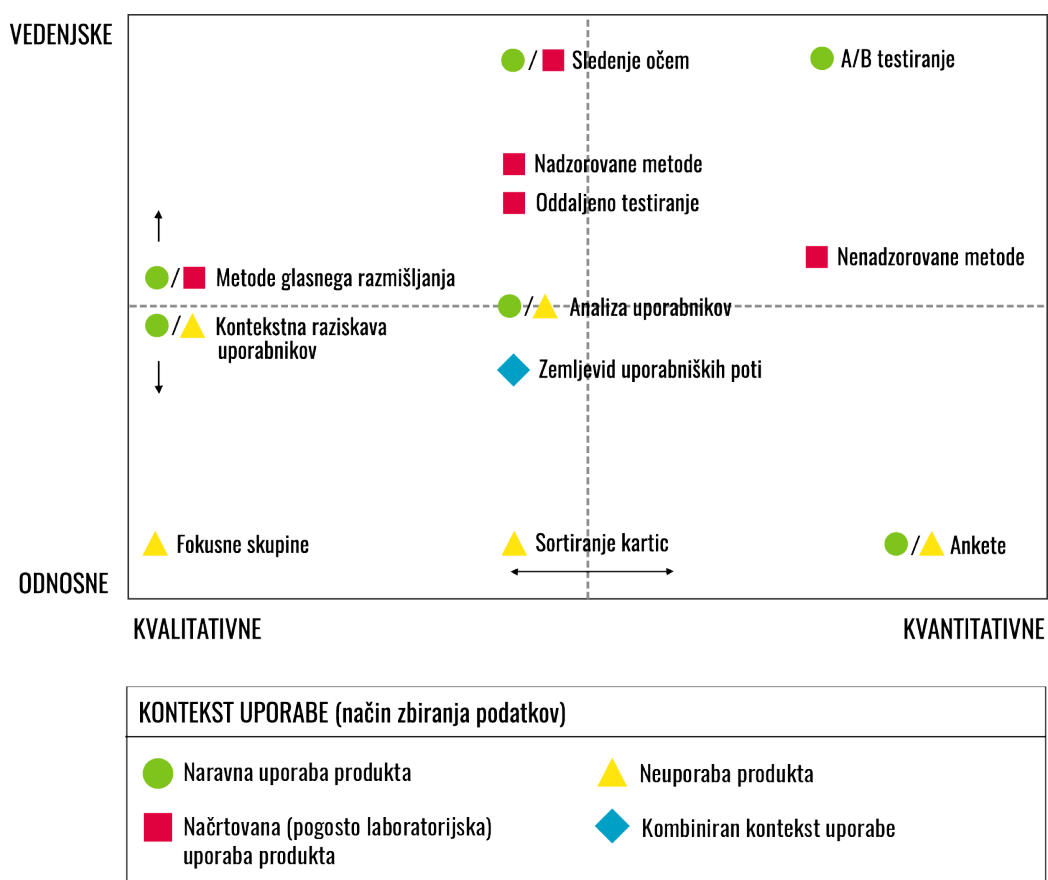
- **Kontekst uporabe:** Tretje razlikovanje med metodami je kontekst uporabe, ki metode razlikuje po naslednjih načinih uporabe ocenjevalnega produkta:
 - »Naravna« uporaba produkta.
 - Uporaba z vnaprej določenim scenarijem.
 - Produkta se med raziskavo ne uporablja.
 - Hibridna uporaba zgornjih načinov.



Slika 4.1: Kvalitativne metode so primerne predvsem za odgovore na vprašanja, »kako ali zakaj«, medtem ko so kvantitativne bolj primerne za odgovore na vprašanja »koliko?«

V prvem koraku izbire metod in tehnik si lahko pomagamo z vprašanji, kaj nas sploh zanima in zakaj oziroma kako bodo pridobljeni podatki v pomoč pri razvoju projekta. Pogosta vprašanja, ki lahko vplivajo na izbiro metod [17] so:

- Kdo so uporabniki?
- Kaj so cilji in potrebe uporabnikov ter kaj je njihova motivacija in kakšno je obnašanje?
- Kako trenutno uporabljajo produkt?
- Katere ostale produkte uporabljajo?
- Kje imajo težave z delovnimi procesi?
- Ali jim je produkt všeč?



Slika 4.2: Prikaz izbranih metod na tridimenzionalnem grafu

4.1 Pregled izbranih metod in tehnik za zagotavljanje dobre uporabniške izkušnje

4.1.1 Metode testiranja uporabnosti

Protokol glasnega razmišljanja

Metode glasnega razmišljanja že 25 let spadajo med najbolj pogosto uporabljene metode, saj so poceni, robustne, prilagodljive in enostavne za implementacijo.

»V testih glasnega razmišljanja sodelujoče uporabnike prosimo, da uporabljajo uporabniški vmesnik oziroma produkt, medtem pa neprekinjeno razmišljajo na glas – misli, ki se jim porajajo med uporabo uporabniškega vmesnika, povedo na glas.« [32]

Z metodami neposredno dobivamo povratne informacije s strani uporabnika, opazujemo in poslušamo, kako si razlagajo določene elemente vmesnika in kje določeni elementi nimajo želenega učinka. Metodo se lahko uporabi v katerikoli razvojni fazi produkta in tudi za evalvacijo katerekoli vrste uporabniškega vmesnika, z izjemo glasovnih, kjer je izvedba otežena. Njena izvedba je cenovno ugodna, saj zahteva le uporabnike in opazovalca, ki si opažanja zapisuje. Ponavadi zadostuje okoli 3 do 5 uporabnikov in približno en dan za izvedbo. Za razliko od kvantitativnih raziskovalnih metod, kjer že manjša napaka lahko pomeni nenatančnost rezultata, je metoda glasnega razmišljanja zelo robustna in proizvede zelene rezultate tudi, če je ne izvaja strokovnjak.

Slabost metode je, da je za uporabnike daljše razmišljanje na glas lahko zahtevno in na začetku nenaravno in neprijetno. Nekateri uporabniki lahko s težavo izražajo iskrene misli, ki jih sprva dobijo, zato najprej premislijo, katera mnenja si želijo izraziti, kar lahko vpliva na uspešnost raziskave. Z metodo lahko pridobimo zelo uporabne informacije, vendar je za dobro evalvacijo potrebna kombinacija z drugimi metodami.

Prednosti metode:

- Rezultati metode so opisi konkretnih problemov v uporabnosti in njihovih vzrokov.
- Zahteva malo sodelujočih (3 do 5).
- Uporaba v katerikoli razvojni fazi.
- Ne zahteva veliko izkušenj opazovalca.

Slabosti metode:

- Razmišljanje na glas lahko vpliva na uporabnikovo uporabo sistema.
- Odvisnost rezultatov od iskrenosti uporabnikov.

Oddaljeno testiranje

Oddaljeno testiranje uporabnosti je metoda testiranja uporabnosti, ki temelji na testiranju uporabniškega vmesnika s pomočjo programske opreme za komunikacijo med uporabnikom in ocenjevalcem. Med oddaljeno testiranje spadajo vse metode testiranja uporabnosti, kjer ocenjevalec in udeleženec/-ci nista na isti lokaciji. Uporabnik lahko vmesnik uporablja v svojem domačem ali delovnem okolju, kar je ena od pglavitnih prednosti te metode, saj so reakcije testiranih uporabnikov v naravnem okolju ponavadi bolj iskrene in natančne, njihova mnenja pa nepristranska. Za razliko od laboratorijskih, je oddaljeno testiranje tudi lažje in cenejše za izvedbo.

Izvedba metode je lahko nadzorovana, kjer ocenjevalec udeleženca opazuje v živo preko spletne kamere ali oddaljenega dostopa, ali nenadzorovana, kjer udeleženec naloge rešuje neopazovano, brez interakcije z ocenjevalcem. Za razliko od nadzorovanega ocenjevanja, nenadzorovano omogoča hitrejše testiranje z večjim številom udeležencev in ponavadi rezultira v obliki kvantitativnih podatkov, saj so naloge sestavljene vnaprej.

Prednosti metode:

- Rezultati metode so opisi konkretnih problemov v uporabnosti in njihovih vzrokov.
- Udeleženci naloge rešujejo v njihovem naravnem okolju.
- Nenadzorovani način udeležencem omogoča lastno izbiro termina za ocenjevanje.

Slabosti metode:

- Možnost tehničnih motenj med sejami.
- Ponavadi ocenjevalec udeleženca ne vidi, kar preprečuje ocenjevanje govornice telesa.
- Ocenjevanje papirnatih prototipov zaradi oddaljenosti ni mogoče.
- Nenadzorovan način onemogoča prilagajanje vprašanj in testiranj za posameznega udeleženca, saj so ta sestavljena vnaprej.
- Pri nenadzorovanem načinu je uspešnost testiranja lahko odvisna od spletnih analitik, kar zahteva dodatno analizo.

Orodja:

- LookBack.io
- Zoom.us
- Google Hangouts
- Skype

Sledenje očem

Sledenje očem je proces opazovanja osredotočenosti pogleda ali premika očesa glede na pozicijo glave s pomočjo video ali spletne kamere. Uporablja se za psihološke in tržne potrebe raziskav, ter evalvacije vizualnih interaktivnih sistemov z uporabniki.

Pri evalvaciji spletnih strani in aplikacij, med bolj uporabne podatke, ki jih proces lahko zagotovi, spadajo: kje in koliko časa je uporabnikova osredotočenost pogleda na spletni strani, kako se ta na njej spreminja in katere predele spletne strani uporabnik spregleda. Pridobi lahko tudi podatke, kako uporabnik po spletni strani navigira, ter kako velikost, dizajn in postavitev elementov na spletni strani vplivajo na pozornost uporabnika. [51]

Sledenje očem je pred leti predstavljalo drag način testiranja uporabnosti, saj je zahtevalo drago programsko in strojno opremo ter eksperte za analizo. Danes obstaja veliko cenejših orodij v obliki programske opreme, ki avtomatizirano zajemajo in analizirajo dobljene podatke, te pa predstavijo v obliki toplotnega zemljevida (angl. Heat Map) ali sledi skokovitega gibanja očesa med točkami fiksacije (angl. Saccade pathways) [51].

Toplotni zemljevid s pomočjo barvnih meril od modre do rdeče prikazuje, kam in koliko časa je bil uporabnik na spletni strani osredotočen, kjer rdeče obarvani predeli strani prikazujejo najdaljšo osredotočenost pogleda (Slika 4.3).

Prednosti:

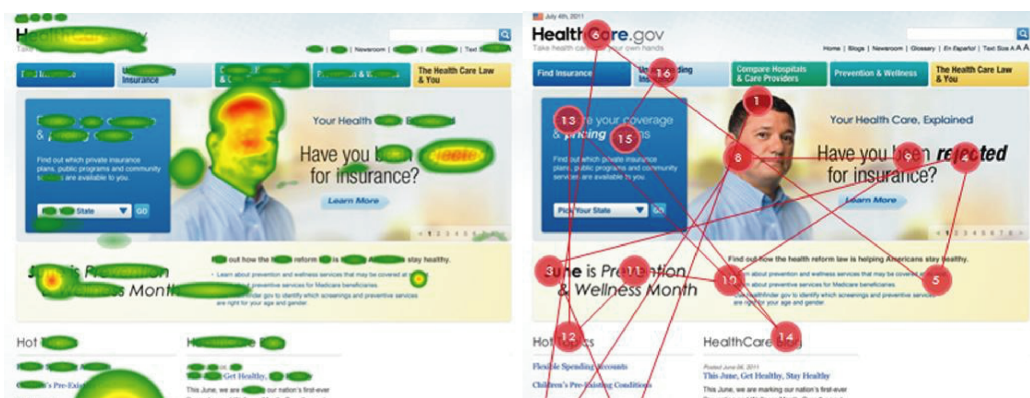
- Hitrost in uporabnost predvsem v fazi testiranja prototipov.
- S souporabo drugih metod zagotovi uporabne informacije.
- Vizualna predstavitev rezultatov.

Slabosti:

- Nenatančnost pridobljenih podatkov, saj se uporabnik med testiranjem v določen predel na strani lahko osredotoča zavestno.
- Nenatančnost zaradi nezmožnosti sledenja perifernemu vidu.
- Pove nam, kam je uporabnik osredotočen, ne pa tudi zakaj.
- Možnost netočnosti podatkov zaradi očal, kontaktnih leč, majhnih zenic, ...

Orodja:

- Crazy Egg
- xLabs
- GazePointer



Slika 4.3: Sledi toplotnega zemljevida (levo) in skokovitega gibanja očesa med točkami fiksacije (desno) [51]

4.1.2 Metode ekspertnega pregledovanja

Hevristična evalvacija

Leta 1990 sta Jakob Nielsen in Rolf Molich, ena glavnih poznavalcev uporabnosti, sestavila seznam desetih smernic oblikovanja uporabniških vmesnikov [34]. Te smernice oziroma hevristike se še danes uporabljajo tudi v najbolj uspešnih podjetjih, kot so Apple, Google in Adobe. Poleg omenjenih hevristik obstaja še veliko drugih, kot so 8 zlatih pravil Bena Schneidermana, smernice Smitha in Mosiera, smernice Motif in druge.

Ker se tehnologija zelo hitro spreminja in razvija, sledenje le omenjenim smernicam lahko ni dovolj. Pri hevristični evalvaciji je priporočljivo, da se vmesnik ocenjuje s pomočjo zanj primernih smernic, ki se določijo v prvem koraku evalvacije. Sledenje Nielsenovim in Molichovim smernicam za potrebe ocenjevanja uporabnosti vmesnikov mobilnih naprav in spletnih skupnosti, ki se vseskozi spreminjajo, bi še vedno pozitivno vplivalo na končni produkt, vendar danes to ne bi bilo dovolj. Smernice se danes uporabljajo bolj kot splošna usmeritev za definiranje svojih, za vsak projekt prilagojenih smernic. Podjetja, kot so Apple in Google, glede na trende in razvoj tehnologije prilagajajo svoje smernice za grajenje vmesnikov na platformah iOS [5] in Android [14], vse pa izhajajo iz osnovnih hevristik. Večina hevrističnih evalvacij zahteva približno pet do deset smernic, ki jih izberemo tako, da ne zanemarimo končne uporabnosti sistema, spletne strani, aplikacije itd. Nekatere smernice se med seboj prekrivajo ali pa so si v konfliktu, kar lahko otežuje delo ocenjevalca v primeru, da jih je veliko (več kot deset) [53].

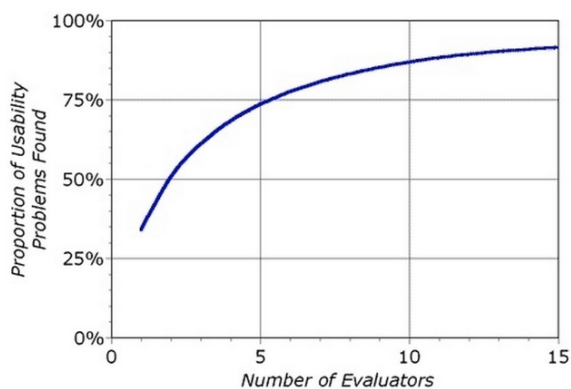
Za izvedbo hevristične evalvacije uporabniki niso potrebni. Metodo ponavadi izvajajo izkušeni ocenjevalci/eksperti uporabnosti, ki se spoznajo na področje in okolje, v kateri se bo produkt uporabljal.

Z raziskavo leta 1992 je Jakob Nielsen ugotovil, da različni ocenjevalci ugotovijo različno število in različne tipe problemov v uporabnosti. Če nam razpoložljiva sredstva to dopuščajo, je priporočljivo, da produkt oceni med tri in pet ocenjevalcev [28]. V začetnih stopnjah razvoja pa sta lahko učinkovita že eden ali dva. Večje število ocenjevalcev pomeni boljše rezultate

in več najdenih problemov uporabnosti, se pa število ocenjevalcev in delež odkritih problemov ne povečujeta linearno (Slika 4.4). Nielsen in Landauer sta to dokazala, z matematično formulo.

$$NP(i) = N(1 - (1 - k)^i), \quad (4.1)$$

kjer $NP(i)$ predstavlja število različnih problemov uporabnosti, najdenih s strani ocenjevalcev. N predstavlja število vseh najdenih problemov uporabnosti, k pa odstotek problemov, ki jih odkrije posamezen ocenjevalec.



Slika 4.4: Nelinearnost med številom ocenjevalcev (os x) in deležem odkritih problemov (os y) [28]

Prednosti:

- Poceni izvedba.
- Odkrije veliko problemov v uporabnosti.
- Ocenjevalci ponavadi podajo rešitve za najdene probleme.

Slabosti:

- Problemi, identificirani s strani ocenjevalcev, niso vedno kritičnega pomena oziroma sploh niso problemi.
- Najdeni problemi so močno odvisni od ocenjevalcev in njihovih izkušenj.
- Le 33% najdenih problemov je kritičnih za uporabnost produkta [7].

Kognitivni sprehod

Kognitivni sprehod je evalvacijska metoda, katero ponavadi izvaja eden ali več ocenjevalcev, ki sistem ocenjujejo s perspektive končnih uporabnikov. Glavni cilj kognitivnega sprehoda je testiranje naučljivosti, ki je ena od glavnih atributov uporabnosti in se osredotoča na to, kako hitro se neizkušen uporabnik lahko nauči uporabljati sistem. Pri izvedbi metode nas zanima, ali bo uporabnik znal opraviti podane naloge [38].

Proces izvedbe se začne z definiranjem nalog, primernih za analizo. Nalogo ponavadi sestavlja več različnih sekvenc korakov, ki uporabnika pripeljejo do cilja. Ocenjevalec se nato sprehodi skozi vmesnik in skuša s perspektive uporabnika priti do zadanega cilja naloge. Na vsakem koraku v sekvenci si ocenjevalec lahko pomaga z naslednjimi vprašanji za iskanje problemov v uporabnosti [52]:

1. Ali bo uporabnik poskušal doseči učinek, ki ga ima podnaloge? Npr. ali uporabnik razume, da je ta podnaloge potrebna za dosego cilja?
2. Ali bo uporabnik opazil, da je na voljo ustrezna interakcija? Npr. ali je gumb dovolj viden?
3. Ali bo uporabnik razumel, da je zelena podnaloge dosegljiva z določeno interakcijo? Npr. ustrezen gumb je uporabniku viden, vendar teksta v njem ne razume in ga zato ne bo kliknil.
4. Ali uporabnik dobi ustrezno povratno informacijo? Npr. ali bo uporabnik vedel, kdaj je storil pravilno interakcijo.

Med testiranjem si ocenjevalec zapisuje vsa opažanja in morebitne rešitve za izboljšave, ki jih potem skupaj z ostalo ekipo analizirajo in po potrebi ukrepajo.

Rick Spencer je metodo za potrebe hitrega razvoja še poenostavil. Namesto štirih vprašanj, ki si jih ocenjevalci tekom sprehajanja skozi korake sekvenc zastavljajo, je predlagal dve [46]:

1. Ali bo uporabnik vedel, kaj storiti v tem koraku?
2. Če uporabnik stori pravilen korak, ali bo vedel, da je bil korak pravilen in da je na pravi poti do cilja?

Prednosti:

- Izvedljiva je brez končnih uporabnikov.
- Osredotoča se na konkretne naloge uporabnika.
- Uporabimo jo lahko na kateremkoli koraku razvoja.
- Je hitra in poceni z uporabo poenostavljene različice.

Slabosti:

- Dobljeni rezultati so odvisni od znanja ocenjevalcev.
- Zagotavlja analizo, ki se v večini osredotoča le na tekstovno in grafično predstavitev sistema.
- Ne zagotavlja podatka o pogostosti napak in problemov.

Preverjanje konsistence

Do nekonsistentnosti pogosto pride pri integraciji novih produktov z obstoječimi ali kadar produkt razvija več razvijalcev in oblikovalcev [23].

Konsistenca uporabniških vmesnikov uporabniku zagotavlja, da so funkcije, razporejenost in izgled grafičnih elementov vmesnika enake na vseh korakih njihove uporabe in na vseh platformah, ki jih storitev ponuja. Če konsistenca ni zagotovljena, lahko uporaba vmesnika predstavlja dodatno obremenitev uporabnikovega spomina.

Izvedba metode je najbolj primerna v začetnih korakih razvoja produkta, ko so pripravljene prototipi in dokumenti oblikovanja. Preverjanje v zadnjih fazah razvoja ni priporočljivo, saj so popravki in spremembe zaradi nekonstenc drage. Priporočljivo je, da pri izvedbi metode sodelujejo oblikovalec, razvijalec programske opreme in vodja projekta, z izkušnjami na področju uporabniške izkušnje.

4.1.3 Metode poizvedbe

Analiza nosilcev projekta

Z metodo analize nosilcev projekta se v začetni fazi razvoja osredotočimo na zahteve in cilje, ki jih definirajo naročniki projekta. S pomočjo intervjujev in s pogovorom z naročnikom skupaj definiramo glavne cilje, ki jih naročnik z razvojem novega sistema želi doseči. Pomemben je tudi sestanek glede razpoložljivih finančnih sredstev in pomembnih funkcionalosti.

Konkurenčna analiza

S konkurenčno analizo v začetni fazi razvoja podrobno analiziramo produkte konkurence in definiramo njihove prednosti in slabosti. Če slabosti konkurence prepoznamo, lahko zanje poiščemo ustrezne rešitve in jim s tem konkuriramo na trgu. Metoda je primerna predvsem za projekte z veliko podobnimi konkurenčnimi rešitvami.

Prednosti:

- Lahko privede do novih idej uporabe različnih tehnologij in metodologij za načrtovani projekt.

- Poišče pomankljivosti v storitvah in produktih konkurence.
- Pomaga pri strateškem in kompetenčnem postavljanju cen za načrtovan projekt.

Slabosti:

- Stroški in čas raziskave.
- Slaba analiza konkurence lahko privede do slabih poslovnih odločitev.
- Preveč oziranja na konkurenco lahko zmanjša inovativnost.

Analiza uporabnikov

Analiza uporabnikov je metoda, s katero fizično, intelektualno in osebnostno analiziramo bodoče uporabnike, na podlagi kvalitativnih in kvantitativnih podatkov.

Uporabniki se lahko razlikujejo po:

- Fizični sposobnosti in delovnem okolju.
- Kognitivnih in zaznavnih sposobnostih.
- Osebnostnih lastnostih.
- Kulturnih razlikah.
- Telesnih zmožnostih in nezmožnostih.
- Starosti.

Za uspešno analizo uporabnikov je potrebno razmisliti o vseh načinih njihove interakcije s storitvijo, ki jo razvijamo. Pomembno je, da vključimo vse

uporabnike, tudi tiste z malo izkušnjami in tiste z določenimi nezmožnostmi.

V začetni fazi projekta je potrebno ugotoviti [40]:

- Kdo so bodoči uporabniki ter kakšne naloge in zahteve opravljajo.
- Kako naloge opravljajo trenutno.
- Kateri so problemi, s katerimi se soočajo pri uporabi trenutnega sistema.
- Katere nove funkcionalnosti potrebujejo za doseg ciljev.

Za pomoč pri nabiranju informacij o uporabnikih in njihovi analizi si lahko pomagamo z različnimi metodami in tehnikami, kot so zemljevid uporabnikove poti (angl. user journey map), kontekstna raziskava uporabnikov, intervjuji, ankete ipd.

Eden glavnih produktov analize uporabnikov so t. i. osebnosti (angl. Personas). Te predstavljajo ključno skupino bodočih uporabnikov, opisujejo njihove potrebe in pričakovanja, upoštevajo njihove vrednote in ozadja, ter nam dajejo sliko o tem, kako bodo uporabljali produkt.

Prednosti:

- Pomaga pri definiranju potrebnih funkcionalnosti.
- Pomaga pri izdelavi informacijske arhitekture, obnašanju vmesnikov.
- Pomaga pri oblikovanju vmesnikov.
- Na podlagi obnašanja uporabnikov, pomaga pri izbiri razvojnih pristopov.
- Piscem pomaga pri sestavi besedil za določene uporabnike.

Kontekstna raziskava uporabnikov

Z metodo kontekstne raziskave uporabnikov, s pomočjo intervjuja raziščemo uporabnikove naloge in cilje. S tem želimo uporabnikom zagotoviti čim lažje in čim hitrejše delo z novim sistemom. V kolikor gre za naloge, ki jih z novim sistemom želimo avtomatizirati oziroma pospešiti, uporabnike opazujemo, kako trenutno opravljajo svoje delo, med tem pa si zapisujemo, kako bi procese nalog lahko optimizirali oziroma jih vključili v sistem. V prvem koraku identificiramo različne uporabnike in vsakega opazujemo pri delu, jih sprašujemo o njihovih vrednotah, trenutnih omejitvah pri delu in skrbeh, ki jih imajo pri izvajanju nalog ali pri uvedbi novega sistema. V zadnjem koraku analize pridobljene podatke analiziramo in jih implementiramo v koraku implementacije [13].

Časovno priprava raziskave in vprašanj ponavadi zahteva enako časa kot opazovanje uporabnikov in analiza ugotovitev.

Prednosti:

- Vpogled v dejanski kontekst bodočih uporabnikov.
- Zagotavlja informacije o vplivih okolja.
- Zagotavlja pomemben vir informacij za nekatere druge metode (določanje osebnosti, pisanje scenarijev).

Slabosti:

- Odvisnost od sposobnosti opazovalca.
- Zagotavlja le informacije o trenutnem izvajanju nalog in okolju.
- Prisotnost opazovalca lahko spremeni obnašanje uporabnikov.

Ankete

Spoznavanje uporabnikov in prejemanje povratnih informacij o sistemu s pomočjo anket spada med najcenejše metode analize uporabnikov. Uporabnike lahko anketiramo preko telefona, elektronske pošte, spletnih aplikacij ali v živo. Anketo sestavimo z dobro zasnovanimi vprašanji, s katerimi v začetni, raziskovalni fazi razvoja, čimbolj podrobno spoznamo uporabnike, kaj si želijo v novem sistemu ter kakšne težave imajo trenutno pri nalogah.

Pri razvoju spletnih aplikacij in trgovin lahko v začetni fazi uporabnike anketiramo tudi o njihovih splošnih preferencah in mnenjih, kot so: kje in kaj najraje kupujejo, kaj jim je pri spletnem nakupu najbolj pomembno, je proces nakupa lahek, ali bi na tej spletni strani še nakupovali itd.

Ankete v srednji fazi razvoja nam zagotovijo uporabne povratne informacije o prototipih, v zadnji, evalvacijski fazi razvoja pa povratno informacijo o uporabnosti vmesnika. [16]

Prednosti:

- Nizka cena in možnost velikega števila anketirancev s pomočjo spletnih anket.
- Avtomatizirano shranjevanje in analiza spletnih anket.

Slabosti:

- Hitra in nepremišljena sestavljena vprašanja lahko rezultirajo nenačane odgovore.
- Neiskrenost anketirancev.

Orodja:

- SurveyMonkey
- Google Forms

Fokusne skupine

Metoda fokusnih skupin je razprava, kjer moderator sodelujoče vodi skozi vprašanja na določeno tematiko. Pogosto se uporabljajo v začetnih fazah razvoja, za zbiranje povratnih informacij o uporabnikih, produktih, konceptih, prototipih, nalogah, strategijah in okolju. Končni produkti fokusnih skupin so lahko v obliki zapiskov, video in zvočnih posnetkov ali pa so novo nastale tematike in predstavitve za ostale deležnike projekta.

Prednosti:

- Uporaba v začetnih fazah projekta zagotavlja vpoglede in odgovore na vprašanja različnih uporabnikov in deležnikov.
- Če predpostavimo, da so sodelujoči iz iste geografske regije, metodo lahko izvedemo hitro in poceni.

Slabosti:

- Metoda temelji na zgodbah in ne opazuje konkretnih interakcij s sistemom in uporabnikom / deležnikom.
- Pridobljeni podatki so odvisni od iskrenosti in natančnosti pomnjenja sodelujočih.
- Možnost konfliktov med sodelujočimi.
- Uspešnost metode je odvisna od izkušenosti voditelja.

A/B testiranje

A/B testiranje je eksperiment, kjer bodoči uporabniki in ostali udeleženci testirajo dve različni izdaji produkta (prototipi, oglasi, vizualni elementi, ...). Testiranje se pogosto uporablja pri razvoju spletnih strani za e-trgovine,

kjer se testira različne tokove nakupov (angl. purchase funnel), saj imajo že manjše spremembe lahko velik vpliv na prodajo. Učinkovita pa je tudi pri testiranju inovativnih rešitev v primerjavi z obstoječimi in za primerjanje vizualnih lastnosti elementov aplikacij, kot so barve gumbov, razporeditev, itd.

A/B testiranje ne zagotavlja merjenja uporabnosti in prepoznavanja napak ter pomanjkljivosti pri uporabi izdane verzije, zato je najbolj učinkovito s souporabo drugih metod. Priporočljivo je, da se obe izdaji najprej vsaj do neke mere testira za uporabnost, nato pa uporabi A/B testiranje [29].

Prednosti:

- Preprosta izvedba.
- Jasni in preprosti kvantitativni rezultati testiranja.
- Uporaba za primerjanje vizualnih lastnosti.

Slabosti:

- Lahko zahteva veliko časa.
- Za učinkovitost testiranja potrebujemo veliko udeležencev.
- Hitri rezultati so pogosto napačni.

Orodja:

- Optimizely
- Google Optimize
- Abtasty

Zemljevid uporabniških poti

Uporabniški zemljevid poti predstavlja vizualizacijo kronološko razporejenih procesov, ki jih uporabnik opravlja pri uporabi storitve oziroma produkta. Zemljevid (Slika 4.5) se osredotoča na perspektivo uporabnika, njegov odnos s storitvijo, produktom ali z organizacijo in pri tem vključuje njegovo miselnost, misli ter emocije.

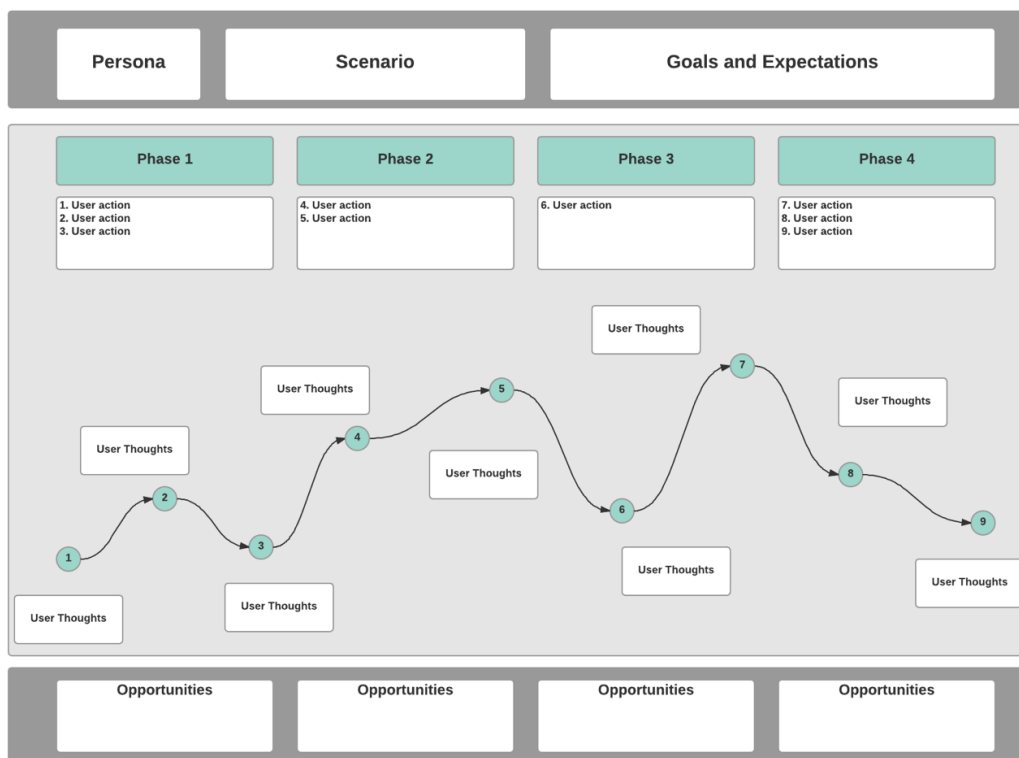
Učinkovito načrtovanje zemljevida zahteva pet visokonivojskih korakov [19]:

1. Težnja in zavezniki: Definiranje pobude in ciljev mapiranja ter vključevanje in izobraževanje deležnikov.
2. Interna preiskava: Zbiranje že obstoječih informacij o uporabnikih in intervjuji z deležniki.
3. Predpostavka formulacije: Sinteza trenutnega stanja uporabniške poti in planiranje dodatnih raziskav o uporabnikih.
4. Zunanja raziskava: Zbiranje novih informacij o uporabnikih za potrditev ali opustitev trenutnega stanja uporabniške poti.
5. Vizualizacija zgodb: Vizualiziranje pridobljenih informacij.

Za ustrezno vizualizacijo uporabniških poti je potrebno definirati:

- Osebnosti (angl. Personas): potrebe, cilje, misli, emocije, mnenja, demografijo in pričakovanja tipičnega uporabnika v obliki zgodbe.
- Časovnico: določitev trajanja uporabniške poti ali faze uporabe sistema.
- Čustva: vrhunec in dna uporabnikovih občutkov (veselje, strah, razočaranje,...) v posamezni fazi uporabe sistema.
- Povezave: postopke in interakcije uporabnika z organizacijo.

- Kanale: kje in kako poteka interakcija uporabnika z organizacijo.



Slika 4.5: Primer zemljevida uporabniških poti [10]

Orodja:

- Smaply
- Gliffy

Sortiranje kartic (angl. Card sorting)

Sortiranje kartic se osredotoča na evalvacijo informacijske arhitekture programskih rešitev s pomočjo dobljenih kvalitativnih podatkov [50]. V uporabniško usmerjenem dizajnu se pogosto uporablja tudi za razvoj delovnih

procesov (angl. workflows), menujev, orodnih vrstic in drugih elementov oblikovanja sistema.

Cilj metode je, da s pomočjo kartic, papirja ali spletnih orodij udeleženci prispevajo k organiziranju posameznih tematik v njim logične kategorije in k poimenovanju le-teh. Pri izdelavi spletnih strani metoda lahko pomaga pri oblikovanju strukture spletišča, odločitvah o tematikah, primernih za domačo spletno stran, in poimenovanju kategorij ter navigacije.

Metoda sortiranja kartic je izvedljiva v dveh oblikah, odprtem in zaprtem sortiranju, lahko pa se izvede tudi v kombinaciji obeh. Pri odprtem sortiranju udeleženci organizirajo tematike vsebine s spletne strani v njim logične skupine in jih nato poimenujejo. V kolikor so kategorije že definirane, pa je bolj primerno zaprto sortiranje, ki od udeležencev zahteva sortiranje tematik v predpostavljene kategorije.

Prednosti:

- Preprosta in cenovno ugodna izvedba.
- Možnost izvedbe veliko sortiranj v kratkem času.
- Temelji na kvalitativnih podatkih, pridobljenih neposredno s strani udeležencev.

Slabosti:

- Ponavadi ne vključuje nalog uporabnika, ampak se osredotoča le na organiziranje vsebine.
- Dobljeni rezultati se lahko med seboj močno razlikujejo, saj so odvisni od posameznih udeležencev.
- Analiza kvalitativnih podatkov je lahko dolgotrajna.

Orodja:

- Optimalworkshop
- User Zoom

Poglavje 5

Trendi izdelave spletnih uporabniških vmesnikov

5.1 Trendi tehnologij

Danes je na voljo veliko različnih ogrodij (angl. framework), Javascript knjižnic, CSS razširitev in drugih tehnologij, namenjenih za lažjo in hitrejšo izdelavo spletnih in mobilnih vmesnikov. Ker je razvoj novih orodij in tehnologij hiter in se spreminja iz leta v leto, je pomembno tako sledenje trendom oblikovanja, kot tudi novim tehnologijam razvoja programske opreme. Izbira tehnologije je odvisna od posameznih potreb in ciljev projekta ter izkušenj razvijalcev.

Angular

Angular [15] je platforma in Javascript ogrodje za razvoj spletnih aplikacij, podprto s strani Googla. Uporabljajo jo podjetja, kot so YouTube, Paypal, Telegram ipd. Temelji na arhitekturi model-pogled-krmilnik (angl. model-view-controller) in arhitekturi model-pogled-pogled-model (angl. model-view-view-model), s čimer ločuje poslovno logiko in razvoj ospredja, s tem pa olajšuje razvoj v skupini ter testiranje. Ogrodje se pogosto uporablja za razvoj dinamičnih mobilnih aplikacij, saj uporablja dvosmerno poveza-

nost podatkov (angl. two-way data binding), kar omogoča avtomatizirano sinhronizacijo modelov in pogledov.

Zaradi obsežne dokumentacije je v primerjavi z drugimi težje naučljiv.

React

React [11] je Facebookova JavaScript knjižnica za razvoj uporabniških vmesnikov spletnih strani. Uporabljajo jo podjetja, kot so Netflix, Instagram, Yahoo! Mail, WhatsApp ipd. Knjižnica skrbi za hierarhijo in prikaz med komponentami uporabniškega vmesnika ter podpira tako čelni, kot strežniški del. React v arhitekturi model-pogled-krmilnik predstavlja le pogled, zato so za ohranjanje takšne arhitekture potrebne dodatne knjižnice, kot je Redux. Prednost knjižnice je v možnosti ponovne uporabe komponent kode in lahki naučljivosti, saj uporablja podobne koncepte kot nekatere druge knjižnice in ogrodja.

Komponente v React-u so ponavadi zapisane v JSX (kombinaciji JavaScript in XML jezika) in s pomočjo spremenljivke `this.state` shranjujejo svoje stanje. Temelji na enosmerni povezanosti podatkov (angl. one-way data binding), kar omogoča, da komponente s pomočjo spremenljivke `this.props`, svoje lastnosti oz. podatke lahko posredujejo na druge komponente.

Za razvoj mobilnih aplikacij je Facebook razvil React Native, ki deluje podobno, vendar za razliko od React-a, ne temelji na HTML jeziku, temveč uporablja svojo sintakso in API kot most za prikaz komponent na mobilnih napravah.

Če so za projekt ustrezne smernice t. i. materialnega oblikovanja (angl. Material Design) [14] in jih želimo implementirati pri oblikovanju vmesnika s pomočjo React-a ali ReactNative-a, lahko uporabimo knjižnico Material UI [1], ki razširja knjižnico React.

Bootstrap

Bootstrap [49] je najbolj popularno HTML, CSS in Javascript ogrodje

za razvoj čelnega dela odzivnih mobilnih spletnih strani, v lasti podjetja Twitter. Za razliko od ostalih ogrodij ima veliko več virov, kot so članki, spletni seminarji, različne Javascript razširitve in predloge za tipografijo, obrazce, gumbe, navigacijo in ostale komponente vmesnika.

Pure.CSS

Pure.CSS [54] je ogrodje za CSS, ki ga je leta 2013 razvil Yahoo. Uporablja se za izdelavo hitrih in odzivnih spletnih strani, zaradi njegove majhne velikosti (le 4,5kB) pa je primerno tudi za razvoj mobilnih spletnih strani. Sestavlja ga seznam CSS modulov (podlog) za izdelavo odzivnih gumbov, menujev, mrež, tabel in drugih komponent, ki se lahko uporabijo na kateremkoli projektu, saj temelji le na CSS jeziku.

Vue.js

Vue.js [55] je v zadnjem letu eno bolj priljubljenih odprtokodnih Javascript ogrodij za hiter in preprost razvoj uporabniških vmesnikov. Evan You, nekdanji razvijalec programske opreme pri Angular, ga je razvil za potrebe manj kompleksne izdelave vmesnikov in lažje naučljivosti, po čemer se tudi najbolj razlikuje od ogrodij, kot sta Angular in React. Vue.js podpira deklarativno interpretacijo, kar pomeni, da se bolj osredotoča na končni produkt kot na postopek.

Uporablja se tako za razvoj preprostih kot tudi bolj zahtevnih spletnih aplikacij ter omogoča visoko kompatibilnost komponent z ostalimi projekti, saj podatke o strukturi, logiki in izgledu komponent shranjuje v eni datoteki. Pri spremembah stanja aplikacije, Vue.js avtomatizirano pregleda minimalno število komponent, potrebnih za ponoven prevod, in njihovih odvisnosti z ostalimi komponentami ter uporabi minimalno število DOM manipulacij. Celostno in dobro napisana dokumentacija omogoča hitro naučljivost, vendar zaenkrat večina skupnosti razvijalcev ni angleško govorečih.

5.2 Trendi oblikovanja

Izgled uporabniških vmesnikov včasih uporabnikom pomeni celo več kot uporabnost sistema. Zaradi pozitivnih občutkov vizualne podobe uporabniki postanejo bolj tolerantni do manjših problemov v uporabnosti in verjamejo, da boljši izgled vmesnika pomeni tudi njegovo boljše delovanje [24]. Pojav je posledica estetsko-uporabnostnega učinka (angl. *aesthetic-usability effect*), ki sta ga na področju interakcije človek-računalnik prva raziskovala Masaaki Kursou in Kaori Kashimura.

Kar 38% ljudi naj bi stran zapustilo, če jim ta vizualno ni všeč [4]. Uporabniški vmesniki so ponavadi prvi predmet interakcije uporabnikov s produktom, zato sta njihov izgled in sledenje zadnjim trendom še toliko bolj pomembna. Sledenje trendom pri oblikovanju uporabniških vmesnikov vpliva tudi na uporabnikovo zaupanje, saj dober izgled vmesnika podzavestno povezujejo z dobrim poslovanjem in storitvami podjetja. Vseeno pa pojav novih trendov ni le posledica estetskih potreb, ampak tudi potreb po višji meri uporabnosti in na splošno boljši uporabniški izkušnji.

Animacije in ilustracije

S hitrim razvojem tehnologij je uporaba animacij na spletnih predstavitev vedno bolj razširjena. Poleg Googlevega Material Design-a, vedno več podjetij, kot sta IBM in Salesforce, izdeluje svoje smernice za oblikovanje grafičnih elementov v gibanju. Uporabo Flash animacij, ki so za izdelavo in brskalnike bolj zahtevne, so nadomestile CSS in Javascript animacije. Izdelava animacij s pomočjo CSS in Javascript tehnologij, kot je WebGL, omogoča izdelavo praktičnih, dostopnih in predvsem učinkovitih animacij, ki dodajajo novo dimenzijo oblikovanja za splet in aplikacije. Zmerna uporaba animiranih slik, 3D slik, logotipov, interaktivnih 360 stopinjskih pogledov in animacij za mikrointerakcije, uporabniško izkušnjo izboljša.

Ker za oblikovanje spletnih strani še vedno prevladujejo ploskoviti (angl. *Flat design*) in minimalistični pristopi, uporabnikom pri uporabi le-teh primanjkuje namigov, ki pa jih lahko brez velikih vplivov na izgled vmesnikov

zagotovimo z uporabo preprostih animacij. Glavni cilj uporabe animacij je torej uporabnost. Preproste animacije lahko uporabnika vodijo po uporabniškem vmesniku, prikazujejo napredek in spremembe ter mu dajejo namige, kot npr. kateri gumb pritisniti. Velik trend oblikovanja so tudi personalizirane ilustracije, ki se uporabljajo za številne namene, kot so animacije, dodajanje osebnosti podjetju, dodatno vizualno predstavitev informacij in drugo.

Včasih je transformacija elementov zahtevala spremembe atributov »width«, »height« in »padding«. Z »webkit« razširitvijo lahko transformacijo dosežemo bolj preprosto (Slika 5.1).

```
.povecava:hover{  
-webkit-transform: scale(1.5);  
-ms-transform: scale(1.5);  
transform: scale(1.5);  
}
```

Slika 5.1: Primer preproste animacije le s pomočjo CSS jezika, kjer se element razreda .povecava ob preletu z miško postopoma poveča do 50 %

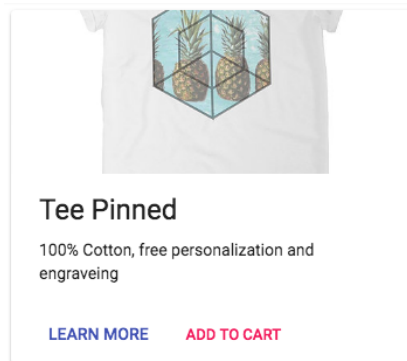
Kartice

S povečanjem potreb po oblikovanju vmesnikov za manjše zaslone, predvsem zaradi vzpona uporabe mobilnih naprav, se povečuje tudi uporaba kartic. Kartice so način organiziranja vsebine, primeren za odzivne spletne strani, saj se njihova velikost in porazdelitev prilagodita glede na velikost zaslona. Prednost kartic je intuitivnost, saj so preproste za uporabo in pregledne, tudi pri predstavitvi večje količine vsebin.

Tipografija

Veliki odebeljeni naslovi vedno bolj izpodrivajo fotografijo. Razlog za to verjetno izhaja iz popularnosti ploskovitega oblikovanja.

Uporaba velikih napisov omogoča nov način, kako pritegniti uporabnika,



Slika 5.2: Primer preproste kartice s pomočjo React knjižnice in Material UI razširitvenih komponent

```
function MediaCard(props) {
  const { classes } = props;
  return (
    <Card className={classes.card}>
      <CardActionArea>
        <CardMedia
          className={classes.media}
          image="tees/pinned.jpg"
          title="Majica Pinned"
        />
        <CardContent>
          <Typography gutterBottom variant="h5" component="h2">
            Lizard
          </Typography>
          <Typography component="p">
            100% Cotton, free personalization and engraving
          </Typography>
        </CardContent>
      </CardActionArea>
      <CardActions>
        <Button size="big" color="primary">
          Learn more
        </Button>
        <Button size="small" color="secondary">
          Add to cart
        </Button>
      </CardActions>
    </Card>
  );
}
```

Slika 5.3: Koda za primer preproste kartice s pomočjo React knjižnice in Material UI razširitvenih komponent

prav tako pa tudi povečuje uporabnost, predvsem dostopnost, saj uporabniki večje napise hitreje in lažje preberejo. Na drugi strani se z večjimi pisavami tudi znižuje potreba po kontrastu med besedilom in ozadjem za lažje razločevanje med znaki besedila, saj tega velikost pisave nadomesti.

Mikrointerakcije

Mikrointerakcije so manjši vizualni dogodki med interakcijo uporabnika z uporabniškim vmesnikom. Glavni cilj njihove uporabe je ustvariti privlačne dogodke, ki čimbolj naravno povezujejo uporabnika z vmesnikom in s tem olajšajo njegovo uporabo ter izboljšajo uporabniško izkušnjo.

Z mikrointerakcijami lahko na bolj zabaven in interaktiven način prikazujemo status napredka, povratne informacije in namige, izboljšamo občutek direktne manipulacije, izboljšamo navigacijo, usmerjamo uporabnikovo pozornost in uporabniku vzpodbudimo pozitivna čustva in zadovoljstvo ob uporabi, s čimer povečamo zaželenost produkta.

Dan Saffer v knjigi *Microinteractions: Designing with Details* [42] mikrointerakcije razčleni v štiri komponente: sprožilce, pravila, povratne informacije in zanke/načine. Sprožilci so dogodki, ki sprožijo mikrointerakcije in se delijo na uporabniške ter systemske. Prvi se sprožijo ob interakciji uporabnika z vmesnikom (npr. pritisk na gumb), drugi pa se sprožijo avtomatizirano, ko se dosežejo določeni pogoji (npr. pojavno okno z obvestilom ob prekoračitvi seje na spletni strani). Pravila določajo dogodke, ki se zgodijo ob sprožitvi mikrointerakcij (npr. preusmeritev ob pritisku na gumb). Povratne informacije predstavljajo potrditve uporabnikom za storjene dogodke in so lahko v obliki zvočnih, vizualnih ali fizičnih signalov (npr. vibracija mobilnega telefona ob preklopu na tihi način). Zadnja komponenta so zanke in načini. Zanke sistemu sporočajo dolžino trajanja mikrointerakcije in ali je potrebna njena ponovitev (npr. ponovna pojavitev obvestila po določenem času, ko ga uporabnik zapre). Načini pa spreminjajo način delovanja mikrointerakcij, ko so te v različnih načinih (npr. vklopitev načina za barvno slepe spremeni določene načine interakcije z vmesnikom).

Eksperimentalna navigacija

Horizontalno postavitve navigacije na vrhu uporabniških vmesnikov vedno bolj nadomeščajo novi načini navigacij, kot so »hamburger« menu, vertikalna navigacija, horizontalno drsenje, robna navigacija, skrita navigacija, »pop-up« navigacija in druge. Razlog za ta trend je predvsem v večjih potrebah po mobilnih vmesnikih, ki jih omejuje velikost zaslona, in potreba po novih inovativnih načinih, kako pritegniti uporabnike k uporabi produkta.

Glavni cilj navigacije je nedvoumno usmerjanje uporabnikov po vseh podstraneh. Pri oblikovanju novih vrst navigacije je glavna ovira njihova uporabnost. Zaradi preteklih izkušenj z vmesniki, so uporabniki navajeni na navigacije, ki so postavljene horizontalno na vrhu zaslona, in na pregledovanje spletnih vmesnikov v obliki črke F, zato je pri oblikovanju novih načinov pomembno zagotoviti čim večjo intuitivnost uporabe, s katero se uporabniki na vmesnik lahko hitro privadijo. Poleg intuitivnosti mora biti dobra navigacija tudi preprosta, jasna in konsistentna na vseh podstraneh.

Novi načini navigacije so uporabnikom lahko zabavni in jih pritegnejo k uporabi, vendar napačna uporaba lahko hitro povzroči nezadovoljstvo, zato je izbira eksperimentalnih navigacij primerna predvsem za manjše projekte oziroma spletne strani, kot so preproste predstavitevne strani in strani, ki vodijo le do enega cilja.

Celozaslonski video

Ljudje si zapomnimo 10% tega, kar preberemo, 20% tega, kar slišimo in 50% tega, kar vidimo in slišimo. Uporaba videa narašča, saj je to hiter in učinkovit način predstavitev zgodb podjetij in njihovih sporočil. Vedno bolj tudi nadomešča slike pri oglaševalskih akcijah ter postaja glavni medij socialnih omrežij. Uporaba videa v ozadju vstopne spletne strani pritegne uporabnika, da na njej ostane dlje časa, posledično pa obstaja večja verjetnost za konverzijo spletne prodaje.

Pri implementaciji videa na uporabniški vmesnik je potrebna previdnost pri uporabi zvokov, kvalitete videa in njegove dolžine, saj te uporabniško

izkušnjo lahko poslabšajo.

```
<div style={{
  width: '500px',
  height: '500px',
  overflow: 'hidden',
}}>

  <video
    style={{
      objectFit: 'cover',
      width: '100%',
      height: '100%',
    }}
    src="video.mp4"
  />
</div>
```

Slika 5.4: Primer uporabe videa s prilagajanjem razmerja višine in širine, s pomočjo React in CSS

CSS lastnost »`objectFit: 'cover'`« (Slika 5.4), omogoča avtomatizirano zapolnjevanje div elementa, ne glede na velikost zaslona.

Gradient

Gradient je postopno mešanje oziroma prehod med barvami. Ko je ploskovito oblikovanje pred nekaj leti postalo trend oblikovanja uporabniških vmesnikov, so se oblikovalci uporabe gradientov izogibali. Vendar uporaba ploskovitih barv lahko hitro rezultira v enoličnost in podobnost med uporabniškimi vmesniki.

Podjetje Spotify je leta 2015 zasnovalo trende gradientov, ko so začeli uporabljati dvotonske gradientne filtre na slikah njihove spletne strani. Danes se gradientne barve uporablja na številnih komponentah uporabniških vmesnikov, zadnji trend pa je njihova uporaba za naslovna besedila, gumbe, 3D elemente, ikone, prehode med stranmi itd. Uporabo gradienta opazimo tudi

pri smernicah oblikovanja vmesnikov, npr. v materialnem oblikovanju [14]. Gradient pri uporabniku vizualno vzpodbudi zanimanje, lahko pa tudi izboljša uporabnost in uporabnika po vmesniku navigira, saj spletne strani ponavadi podzavestno pregledujemo od svetlejših do temnejših predelov.

```
body {
  width: 100wh;
  height: 90vh;
  color: #fff;
  background: linear-gradient(-45deg, #EE7452, #E43C7E, #23A6D5, #23D5AB);
  background-size: 400% 400%;
  -webkit-animation: Gradient 5s ease infinite;
}

@-webkit-keyframes Gradient {
  0% {
    background-position: 0% 80%
  }
  50% {
    background-position: 100% 50%
  }
  100% {
    background-position: 0% 50%
  }
}
```

Slika 5.5: Primer kode preprostega animiranega gradienta za ozadje, z mešanjem štirih barv, s pomočjo CSS

Enostranske spletne strani

Razlog za pojav trenda oblikovanja enostranskih spletnih strani je v veliki meri zaradi potreb po preprostih, hitro odzivnih in z vsemi platformami kompatibilnih straneh. Premikanje po takih straneh s pomočjo drsnika na miški omogoča, da je stran lažje prilagodljiva, tudi za mobilne platforme. Včasih je izdelava takih strani, predvsem pri večjih projektih, bolj zahtevna kot izdelava spletnih mest z več stranmi, saj moramo integrirati vse potrebne elemente, pri tem pa vseeno zagotoviti nedvoumno uporabo vmesnika.

Izdelava enostranskih spletnih strani in aplikacij je najbolj primerna za projekte, kjer je glavni cilj produkta le eden. Pri predstavitvi zgodbe ali prodaji enega produkta oziroma storitve, tak pristop izdelave vmesnika uporabnika vodi le v en željen cilj.

Ker enostranske spletne strani zahtevajo, da je vsa vsebina predstavljena na eni strani, je v primeru veliko vsebine težje zagotoviti hitrost, vizualno hierarhijo in nekatere druge poglede uporabnosti, zato je boljša izbira izdelava večstranskih strani. Dodatna omejitev pri izdelavi takih strani je ta, da ne podpira SEO praks. Ker je celotna vsebina predstavljena na eni strani z enim naslovom URL, iskalniki stran indeksirajo le enkrat, kar znižuje najdljivost strani.

Paralaksa (angl. Parallax)

Paralaksa oz. zamik elementov na spletnih straneh ob drsenju z miško ustvari 3D učinek, s čimer lahko vpliva na to, kako uporabniki sprejemajo vsebino. Učinek se doseže, ko se ob drsenju z miško ozadje spletne strani premika počasneje kot elementi v ospredju.

Čeprav učinek zamika v smislu uporabnosti spletne strani ne izboljša, lahko z njegovo uporabo vsebino na strani predstavimo bolj interaktivno in zabavno, v obliki zgodbe, kar uporabniško izkušnjo izboljša.

```
<style>

body, html {
  height: 100%;
}

.parallax {
  /* Slika ozadja */
  background-image: url("parallax.jpg");

  /* Polna višina */
  height: 100%;

  /* Parallax drsenje z miško */
  background-attachment: fixed;
  background-position: center;
  background-repeat: no-repeat;
  background-size: cover;
}
</style>
```

Slika 5.6: Primer preprostega CSS oblikovanja ozadja za učinek paralakse

Poglavje 6

Zaključek

6.1 Sklepne ugotovitve

Zagotovitev osnovnih zahtev, funkcionalosti in zmogljivosti je za delovanje končnega produkta kritičnega pomena. Delovanje produkta pa še ne pomeni tudi uporabnega produkta in zadovoljstva pri uporabnikih, kar bi danes moralo biti glavno vodilo pri izdelavi vmesnikov.

Uporabnost uporabniških vmesnikov predstavlja eno glavnih lastnosti praktične sprejemljivosti sistemov in dobre uporabniške izkušnje. Dobro uporabniško izkušnjo je v celoti težko zagotoviti in meriti, saj je zelo širok pojem in jo predstavljajo subjektivna doživljanja uporabnikov, ki so pogojeni z njihovimi čustvi, mislimi in okolico. Atributi uporabnosti (naučljivost, učinkovitost, zapomnljivost, število napak in zadovoljstvo) so osnovne in najbolj pomembne lastnosti dobre uporabniške izkušnje prav zaradi njihovega lažjega merjenja in posledično zagotavljanja.

Zagotovitev dobre uporabniške izkušnje pa je že pri malo večjih projektih lahko zelo zahteven proces. Za uporaben, funkcionalen, zaželen, dostopen, kredibilen, najdljiv in korsiten produkt je potrebno konstantno vključevanje uporabnikov in prejemanje njihovih povratnih informacij.

Uporabniško usmerjeno načrtovanje uporabnike vključuje na vsakem koraku, vključuje pa tudi kontekst uporabe sistema in iterativno evalviranje

vmesnih rešitev (prototipov) z uporabniki. Za učinkovit razvoj uporabnega sistema sta za vsak izdan prototip na koncu iteracije potrebni vsaj dve izboljšavi.

V praksi se zelo malo razvijalcev posveča tudi uporabnosti vmesnikov, agilni pristopi navadno niso vključevali oblikovanja uporabniške izkušnje. Razlog za to je verjetno tudi ta, da testiranje uporabnikov zahteva čas, ki ga pri hitrih razvojih (tedenskih sprintih) ponavadi ne posvetijo tudi za dobro izkušnjo.

S poznavanjem osnov, kako vid, spomin in pričakovanja vplivajo na zaznavanje, lahko bolje razumemo nastanek smernic/hevrsitik in navodil za oblikovanje in jih lažje implementiramo v načrtovanje uporabnih uporabniških vmesnikov. Veliko metod ocenjevanja uporabnosti je danes bolj dostopnih, lahko izvedljivih in avtomatiziranih, kar omogoča hitrejše ocenjevanje in lažje vključevanje v agilne pristope razvoja programske opreme. Izvedba le nekaj metod lahko močno izboljša uporabnost produkta in izboljša uporabniško izkušnjo.

Hiter razvoj spletnih tehnologij kot so JavaScript, CSS, PHP itn., razvijalcem in oblikovalcem omogočajo skoraj neomejen razvoj idej za spletne predstavitve. Dodatne knjižnice in razširitve tehnologij pa pri razvoju omogočajo še več možnosti in predvsem hitrejši in lažji način razvoja. Z razvojem tehnologije se večajo tudi pričakovanja uporabnikov. V veliko primerih je izgled vmesnika uporabniku celo bolj pomemben kot njegova funkcionalnost, zato je pomembno sledenje zadnjim trendom oblikovanja. Izgled vmesnika, ki je še pred nekaj leti veljal za modernega, uporabnika lahko hitro odvrne od uporabe. Vseeno pa pojav novih smernic ni le posledica estetskih potreb, ampak tudi potreb po višji meri uporabnosti.

Poznavanje osnov človeških sposobnosti zaznavanja, metod za ocenjevanje uporabnosti, trendov oblikovanja in novih tehnologij razvoja vmesnikov, ni pomembno le za oblikovalca uporabniške izkušnje, ampak tudi za razvijalca programske opreme. Dobra komunikacija in usklajevanje med

njima sta ključnega pomena za učinkovitost razvoja in kakovost končnega produkta. V podjetjih in predvsem startupih pa je veliko povpraševanja tudi po razvijalcih s poznavanjem tako razvoja programske opreme, zagotavljanja uporabniške izkušnje, kot tudi oblikovanja vmesnikov.

V prihodnosti bodo grafično podprte vmesnike verjetno vedno bolj nadomeščali naravni. Vmesniki, ki sprejemajo ukaze preko naravnih gibov, gest obraza in glasov, bodo vedno bolj razširjeni. Spletne predstavitve bodo morda nadomestile predstavitve v 3D navidezni resničnosti. Ker je način interakcije uporabnika z naravnimi vmesniki drugačen od grafično podprtih, kjer za komunikacijo uporabljamo vmesne vhodno/izhodne naprave, bo to pomenilo tudi veliko sprememb v razvoju takih rešitev in pomenu uporabniške izkušnje, posledično pa bodo nastale nove smernice.

Temo sem si izbral, ker me programiranje za ospredje, psihologija in oblikovanje spletnih predstavitev in vsebine, ki sestavljajo uporabniško izkušnjo, od nekdaj zanimajo. Bolje sem spoznal, kako pomembno je danes zagotavljanje uporabniške izkušnje za poslovanje podjetja in uspešnost produkta. S poznavanjem osnov človeškega zaznavanja tudi bolje razumem pomen in nastanek smernic, kar mi bo pri izdelavi vmesnikov pomagalo. Zanimivo je, kako lahko z majhnimi podrobnostmi na vmesniku vplivamo na to, kako uporabnik sprejema nekatere informacije in kako navigira po strani. Spoznal sem, da za izvedbo metod ocenjevanja uporabnosti ni potrebno veliko in se jih lahko hitro in poceni uporabi že pri manjših projektih, njihov učinek pa lahko močno vpliva na uporabnost sistema. S pregledom trendov spletnih tehnologij sem spoznal prednosti in slabosti novejših ogrodij, knjižnic in razširitev za spletno javascript in CSS programiranje in jih bom v prihodnje spoznal še bolj podrobno ter jih uporabljal. Pred diplomsko raziskavo in študijem področja načrtovanja uporabniške izkušnje, pri uporabi nekaterih spletnih predstavitev nisem vedel, kaj konkretno na vmesnikih znižuje uporabnost. Zdaj hitreje najdem in razumem probleme in rešitve vmesnikov in

bolj opažam majhne detajle in njihov pomen za dobro uporabniško izkušnjo.

Literatura

- [1] React components that implement google's material design. Dosegljivo: <https://material-ui.com/>. [Dostopano: 10. 10. 2018].
- [2] ISO 9241-11:2018. Ergonomics of human-system interaction – part 11: Usability: Definitions and concepts. Dosegljivo: <https://www.iso.org/standard/63500.html>, 2018. [Dostopano: 8. 10. 2018].
- [3] ISO 9241-210:2010. Ergonomics of human-system interaction – part 210: Human-centred design for interactive systems. Dosegljivo: <https://www.iso.org/standard/52075.html>, 2010. [Dostopano: 9. 10. 2018].
- [4] Adobe. The state of content: Expectations on the rise. Dosegljivo: <https://blogs.adobe.com/creative/files/2015/12/Adobe-State-of-Content-Report.pdf>, 2015. [Dostopano: 10. 10. 2018].
- [5] Apple Corp. Mac OS X human interface guidelines, 2011.
- [6] Colour Blind Awareness. Colour blindness. Dosegljivo: <http://www.colourblindawareness.org/colour-blindness/>. [Dostopano: 10. 10. 2018].
- [7] Robert W. Bailey, Robert W. Allan, and P. Raiello. Usability testing vs. heuristic evaluation: A head-to-head comparison. *Proceedings of the Human Factors Society Annual Meeting*, 36(4):409–413, 1992.
- [8] Roman Broz and Viktorija Sulčič. Vpliv uporabniške izkušnje na uspešnost e-poslovanja. *Management (18544223)*, 4(2), 2009.

- [9] Raluca Budiu. Quantitative vs. Qualitative Usability Testing. Dosegljivo: <https://www.nngroup.com/articles/quant-vs-qual/>, 2017. [Dostopano: 10. 10. 2018].
- [10] Emily Christensen. How to create a customer journey map. Dosegljivo: <https://www.lucidchart.com/blog/how-to-build-customer-journey-maps>, 2017. [Dostopano: 11. 10. 2018].
- [11] Facebook. A JavaScript library for building user interfaces. Dosegljivo: <https://reactjs.org/>. [Dostopano: 11. 10. 2018].
- [12] Interaction Design Foundation. Usability. Dosegljivo: <https://www.interaction-design.org/literature/topics/usability>. [Dostopano: 9. 10. 2018].
- [13] Gerry Gaffney. Contextual enquiry - a primer. Dosegljivo: <https://www.sitepoint.com/contextual-enquiry-primer/>, 2004. [Dostopano: 11. 10. 2018].
- [14] Google. Material design guidelines. Dosegljivo: <https://material.io/design/introduction/#principles>. [Dostopano: 10. 10. 2018].
- [15] Google. Superheroic JavaScript MVW Framework. Dosegljivo: <https://angularjs.org/>. [Dostopano: 11. 10. 2018].
- [16] Chris Gray. Better user research through surveys. Dosegljivo: <https://uxmastery.com/better-user-research-through-surveys/>, 2014. [Dostopano: 11. 10. 2018].
- [17] Tom Hall. How to choose a user research method. Dosegljivo: <https://uxplanet.org/how-to-choose-a-user-research-method-985112051d84>, 2017. [Dostopano: 10. 10. 2018].
- [18] Jeff Johnson. *Designing with the Mind in Mind: Simple Guide to Understanding User Interface Design Rules*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2010.

- [19] Kate Kaplan. The 5 steps of successful customer journey mapping. Dosegljivo: <https://www.nngroup.com/articles/customer-journey-mapping-process/>, 2017. [Dostopano: 11. 10. 2018].
- [20] Andreas Komninos. Norman's three levels of design. Dosegljivo: <https://www.interaction-design.org/literature/article/norman-s-three-levels-of-design>, 2018. [Dostopano: 8. 10. 2018].
- [21] Nicholas Kramer. A Primer to Web Accessibility for Designers. Dosegljivo: <https://uxplanet.org/a-primer-to-web-accessibility-for-designers-2c548448c612>, 2018. [Dostopano: 8. 10. 2018].
- [22] Darius Kumana and Jon Dickinson. Agile and user-centered design. Dosegljivo: <https://www.thoughtworks.com/insights/blog/agile-and-user-centered-design-0>, 2014. [Dostopano: 9. 10. 2018].
- [23] Practical UX Methods. Consistency inspection. Dosegljivo: <http://practicaluxmethods.com/product/consistency-inspection/>. [Dostopano: 11. 10. 2018].
- [24] Kate Moran. The aesthetic-usability effect. Dosegljivo: <https://www.nngroup.com/articles/aesthetic-usability-effect/>, 2017. [Dostopano: 10. 10. 2018].
- [25] Peter Morville. User experience design. Dosegljivo: http://semanticstudios.com/user_experience_design/, 2004. [Dostopano: 8. 10. 2018].
- [26] David Moth. Site speed: case studies, tips and tools for improving your conversion rate. Dosegljivo: <https://econsultancy.com/site-speed-case-studies-tips-and-tools-for-improving-your-conversion-rate/>, 2012. [Dostopano: 11. 10. 2018].
- [27] Jakob Nielsen. *Usability Engineering*. Morgan Kaufmann, 1993.

-
- [28] Jakob Nielsen. How to Conduct a Heuristic Evaluation. Dosegljivo: <https://www.nngroup.com/articles/how-to-conduct-a-heuristic-evaluation/>, 1995. [Dostopano: 10. 10. 2018].
- [29] Jakob Nielsen. Putting A/B Testing in Its Place. Dosegljivo: <https://www.nngroup.com/articles/putting-ab-testing-in-its-place/>, 2005. [Dostopano: 11. 10. 2018].
- [30] Jakob Nielsen. Short-Term Memory and Web Usability. Dosegljivo: <https://www.nngroup.com/articles/short-term-memory-and-web-usability/>, 2009. [Dostopano: 10. 10. 2018].
- [31] Jakob Nielsen. Parallel and Iterative Design + Competitive Testing = High Usability. Dosegljivo: <https://www.nngroup.com/articles/parallel-and-iterative-design/>, 2011. [Dostopano: 9. 10. 2018].
- [32] Jakob Nielsen. Thinking Aloud: The Number 1 Usability Tool. Dosegljivo: <https://www.nngroup.com/articles/thinking-aloud-the-1-usability-tool/>, 2012. [Dostopano: 10. 10. 2018].
- [33] Jakob Nielsen. Usability 101: Introduction to usability. Dosegljivo: <https://www.nngroup.com/articles/usability-101-introduction-to-usability/>, 2012. [Dostopano: 9. 10. 2018].
- [34] Jakob Nielsen and Rolf Molich. Heuristic evaluation of user interfaces. In *Conference on Human Factors in Computing Systems, CHI 1990, Seattle, WA, USA, April 1-5, 1990, Proceedings*, pages 249–256, 1990.
- [35] D.A. Norman. *Emotional Design*. Basic Books, 2003.
- [36] Nielsen Norman. The Definition of User Experience (UX). Dosegljivo: <https://www.nngroup.com/articles/definition-user-experience/>, 2007. [Dostopano: 8. 10. 2018].

-
- [37] Ekaterina Novoseltseva. User-centered design: An introduction. Dosegljivo: <https://usabilitygeek.com/user-centered-design-introduction/>, 2017. [Dostopano: 9. 10. 2018].
- [38] Usability Body of Knowledge. Cognitive walkthrough. Dosegljivo: <http://www.usabilitybok.org/cognitive-walkthrough>. [Dostopano: 11. 10. 2018].
- [39] R.C. James. Dalmatian, 1973. [Dostopano: 29. 10. 2018].
- [40] User research community. User research in discovery. Dosegljivo: <https://www.gov.uk/service-manual/user-research/user-research-in-discovery>, 2016. [Dostopano: 11. 10. 2018].
- [41] Christian Rohrer. When to Use Which User-Experience Research Methods. Dosegljivo: <https://www.nngroup.com/articles/which-ux-research-methods/>. [Dostopano: 10. 10. 2018].
- [42] Dan Saffer. *Microinteractions: Designing with Details*. O'Reilly Media, Inc., 2013.
- [43] Ben Schneiderman and Catherine Plaisant. *Designing the User Interface*. Addison Wesley, 2004.
- [44] Elizabeth Sillence, Pamela Briggs, Lesley Fishwick, and Peter Harris. Trust and mistrust of online health sites. Conference on Human Factors in Computing Systems - Proceedings. Dosegljivo: https://www.researchgate.net/publication/221516871_Trust_and_mistrust_of_online_health_sites, 2004. [Dostopano: 11. 10. 2018].
- [45] Aleš Smrdel, Ciril Bohak, Miha Amon, and Franc Jager. *Zbirka nalog iz načrtovanja uporabniških vmesnikov*. Založba UL FRI, 2017.
- [46] Rick Spencer. The Streamlined Cognitive Walkthrough Method, Working Around Social Constraints Encountered in a Software Development

- Company. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '00, pages 353–359, New York, NY, USA, 2000. ACM.
- [47] Debbie Stone, Caroline Jarrett, Mark Woodroffe, and Shailey Minocha. *User Interface Design and Evaluation*. Morgan Kaufmann, 2005.
- [48] Mindtouch Digital Team. 5 Internet Marketing Must Do's. Dosegljivo: <https://mindtouch.com/resources/5-internet-marketing-must-dos-in-2010>, 2009. [Dostopano: 11. 10. 2018].
- [49] Twitter. The most popular HTML, CSS, and JS library in the world. Dosegljivo: <https://getbootstrap.com/>. [Dostopano: 11. 10. 2018].
- [50] Usability.gov. Card sorting. Dosegljivo: <https://www.usability.gov/how-to-and-tools/methods/card-sorting.html>. [Dostopano: 11. 10. 2018].
- [51] Usability.gov. Eye tracking. Dosegljivo: <https://www.usability.gov/how-to-and-tools/methods/eye-tracking.html>. [Dostopano: 10. 10. 2018].
- [52] Cathleen Wharton, John Rieman, Clayton Lewis, and Peter Polson. Usability inspection methods. chapter The Cognitive Walkthrough Method: A Practitioner's Guide, pages 105–140. John Wiley & Sons, Inc., New York, NY, USA, 1994.
- [53] Euphemia Wong. Heuristic Evaluation: How to Conduct a Heuristic Evaluation. Dosegljivo: <https://www.interaction-design.org/literature/article/heuristic-evaluation-how-to-conduct-a-heuristic-evaluation>, 2018. [Dostopano: 10. 10. 2018].
- [54] Yahoo. A set of small, responsive CSS modules that you can use in every web project. Dosegljivo: <https://purecss.io/>. [Dostopano: 11. 10. 2018].

-
- [55] Evan You. The Progressive JavaScript Framework. Dosegljivo: <https://vuejs.org/>. [Dostopano: 11. 10. 2018].
- [56] Melody Yvette Ivory. An Empirical Foundation for Automated Web Interface Evaluation. Phd dissertation, UC Berkeley Computer Science Division, 2001.