

SVEUČILIŠTE U SPLITU,
FAKULTET ELEKTROTEHNIKE, STROJARSTAVA I BRODOGRANJE

Dolce Vita

Kolegij:

Korisnička sučelja

Mentor:

Mario Čagalj

Studenti:

Lea Bratić

Karla Miljak

Veljača, 2019./2020.

SADRŽAJ

1. UVOD	3
Korisnici	3
Ograničenja korištenja sustava	3
Zahtjevi i zadaće	3
2. LOW FIDELITY PROTOTIP	3
Desktop verzija	3
Mobilna verzija	10
3. HIGH FIDELITY PROTOTIP	15
Desktop verzija	15
Mobilna verzija	18
4. PRIMJENA PRINCIPA DOBROG DIZAJNA	19
Heuristike	19
Podudaranje stvarnog svijeta sa sustavom	19
Konzistencija i standardi	19
Vidljivost statusa sustava	19
Sloboda korisnika i korisnička kontrola	19
Sprečavanje grešaka	20
Detekcija i oporavak od greške	20
Prepoznavanje umjesto prisjećanja	21
Fleksibilnost i efikasnost korištenja	21
Estetika i minimalistički dizajn	21
Pomoć i dokumentacija	21
C.R.A.P. principi	21
Contrast	22
Repetition	22
Alignment	23
Proximity	23
5. ZAKLJUČAK	25

1. UVOD

Uživanju u slasticama rijetko tko može odoljeti! Na našoj web aplikaciji možete pronaći raznovrsnu ponudu torti i kolača. Iz udobnosti svog doma možete naručiti željeni slatkiš ili pročitati najnovije recepte za pripremu istih.

Cilj ovog projekta je izrada jednostavne web aplikacije koja je namijenjena svim populacijama.

Korisnici

Web aplikacija Dolce Vita namijenjena je svim populacijama koji se žele zasladiti. Baza korisnika uključuje i one ljude čija informatička pismenost na ispodprosječnoj razini. Korisnici tako mogu biti i osobe starije životne dobi (60 i više godina), koji sporo tipkaju, imaju slabiji vid, sluh.

Ograničenja korištenja sustava

Da bi se mogla koristiti web aplikacija Dolce Vita korisnici moraju imati:

- Pristup internetu
- Uređaj s web preglednikom
- Osnovno znanje rada na računalu

Zahtjevi i zadaće

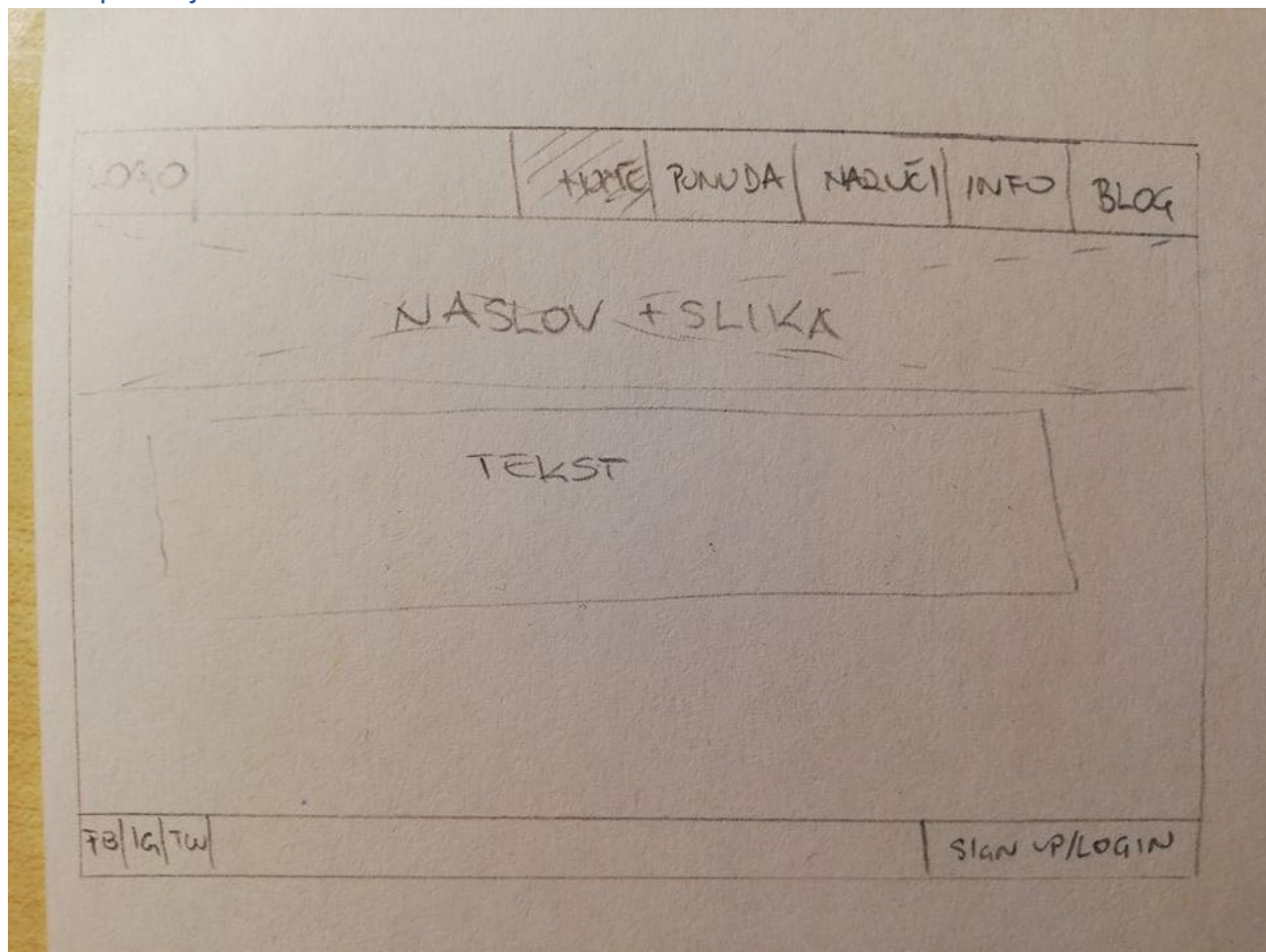
Zahtjevi i zadaće koje mora imati web aplikacija :

- Vaša aplikacija će se koristiti iz web preglednika
- Aplikacija će se koristiti na uređajima različite veličine (*smartphones*, tableti, laptopi i desktop računala)
- Korisnik može pretraživati/filtrirati proizvode ili usluge
- Broj proizvoda/usluga koje nudite putem web aplikacije veći je od 20
- Korisnik će koristiti aplikaciju kao gost ili će se logirati u vlastiti profil
- Broj *javnih* stranica (one stranice koje ne zahtijevaju prijavu u sustav) je 5
- Jedna od javnih stranica je blog (blog mora sadržavati minimalno 20 *postova* koje mogu uključivati slike, videa, *code snippet*-e)

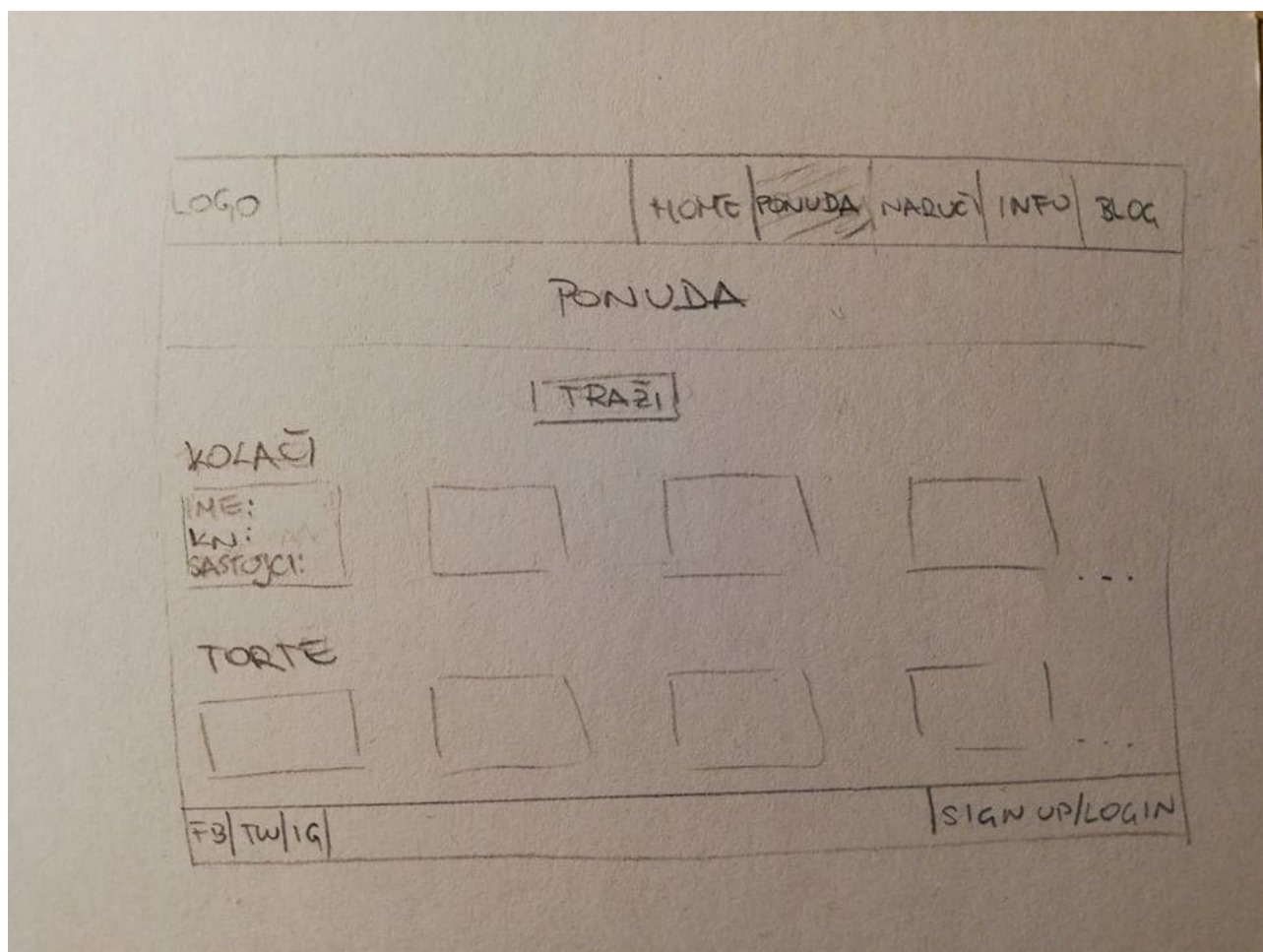
2. LOW FIDELITY PROTOTIP

Koristeći zadane zahtjeve i zadaće napravljen je *low fidelity* prototip uz pomoć olovke i papira. Na sljedećim slikama prikazan je njegov izgled.

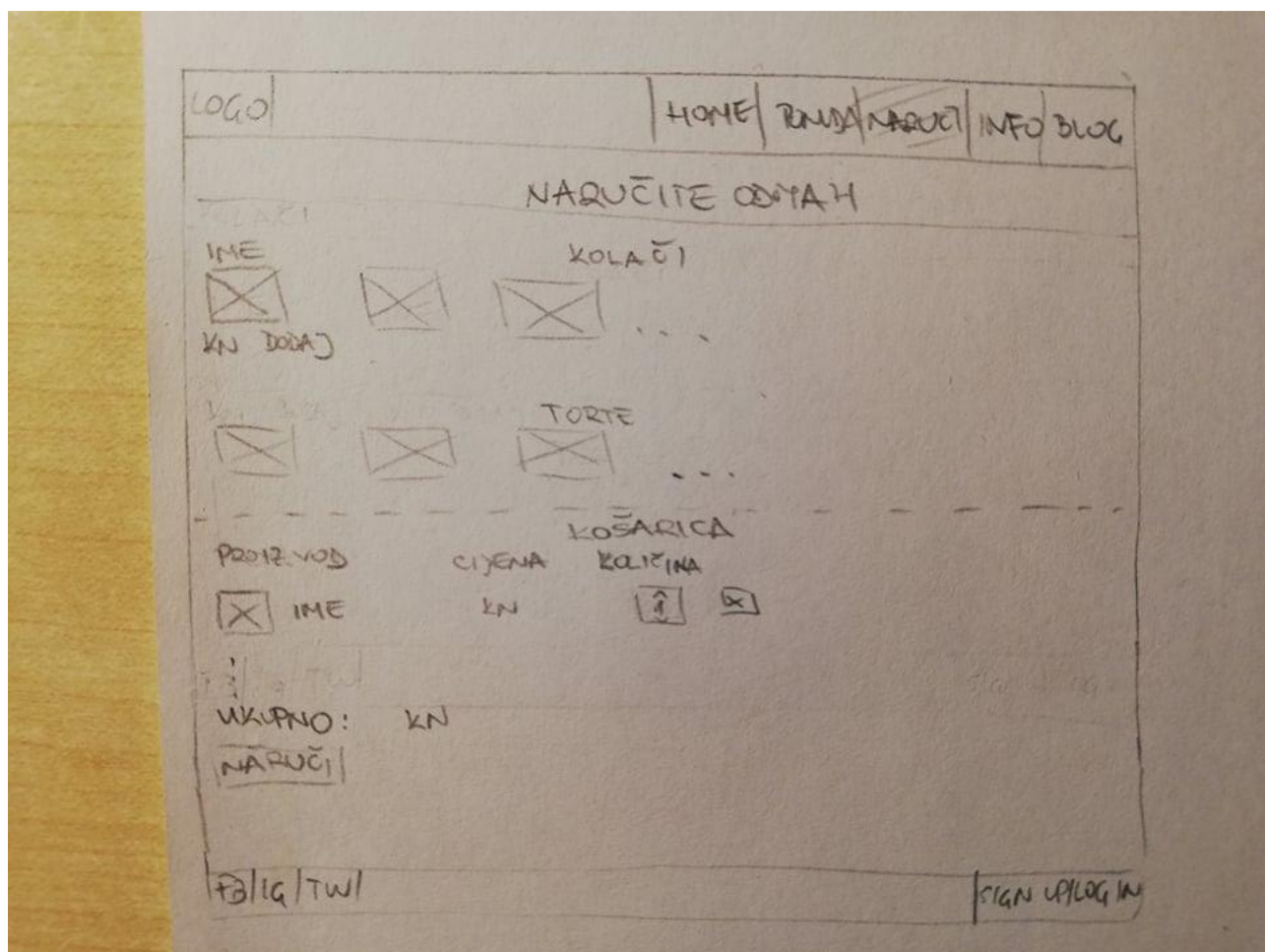
Desktop verzija



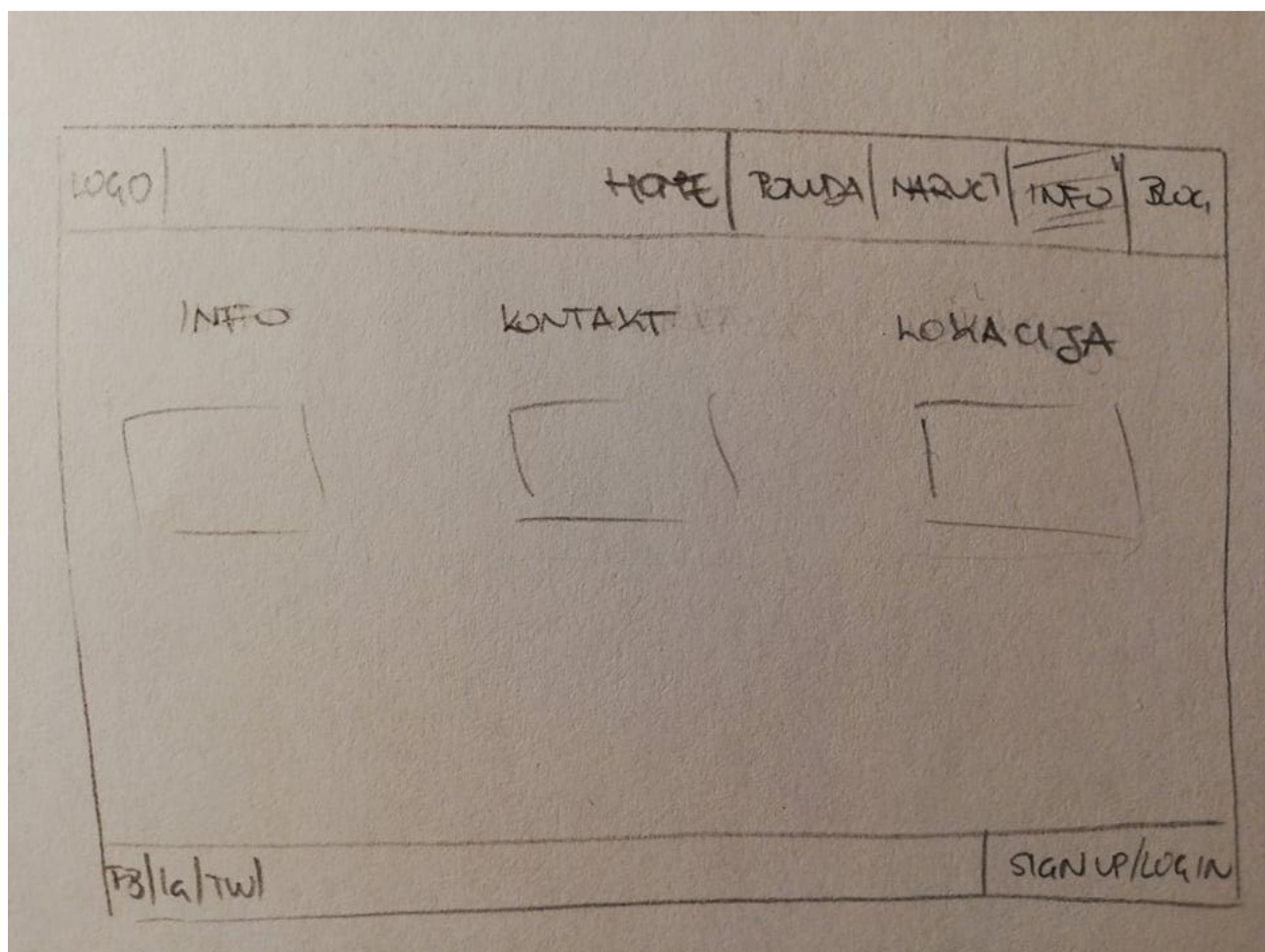
Slika 1 Low fidelity prototip - Početni zaslon



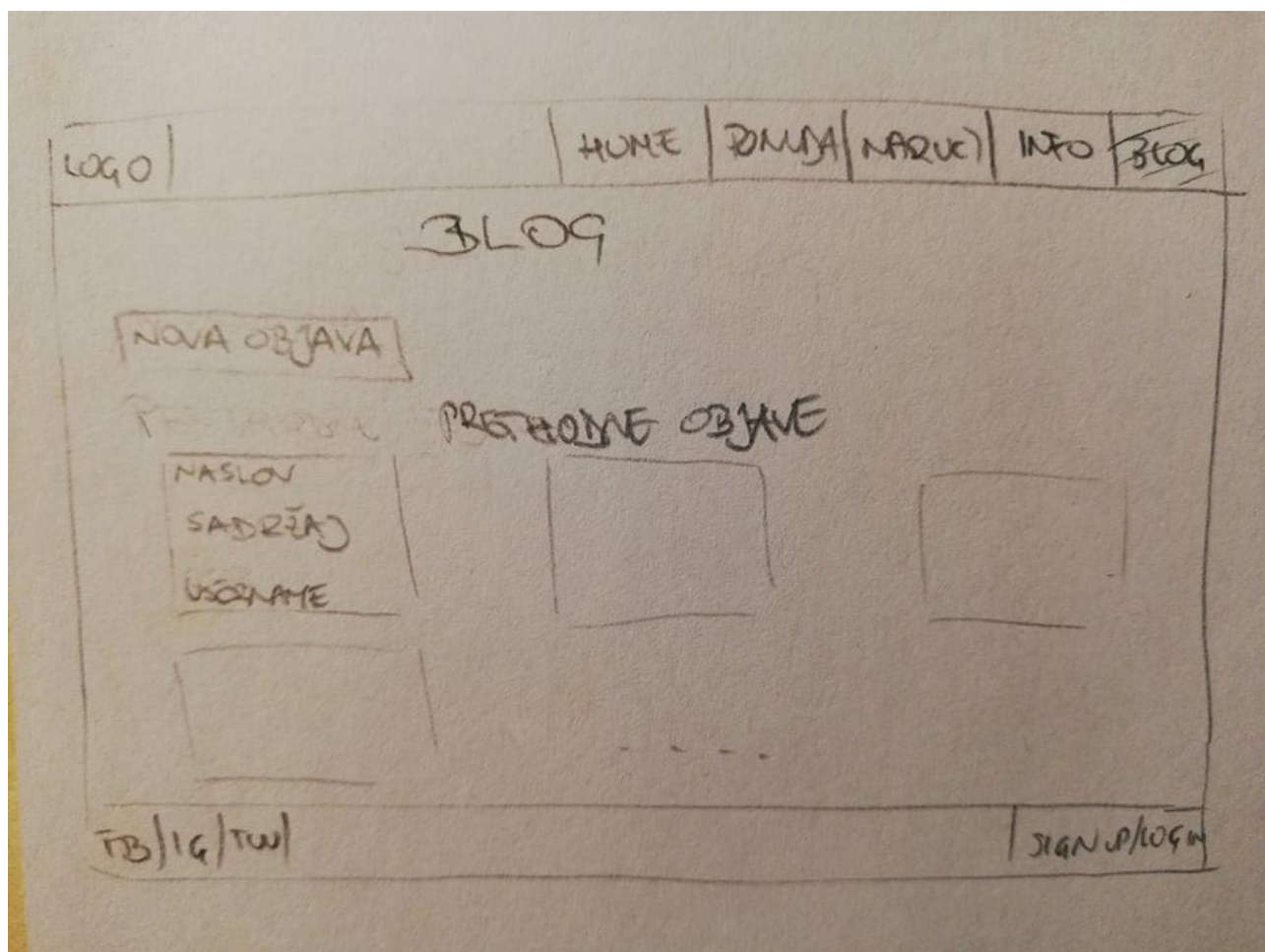
Slika 2 Low fidelity prototip - Prikaz ponude torti i kolača



Slika 3 Low fidelity prototip - Prikaz za naručivanje iz ponude



Slika 4 Low fidelity prototip – Prikaz za stranicu s Informacijama



Slika 5 Low fidelity prototip – Prikaz stranice Blog

The image shows two hand-drawn low fidelity prototypes for a web application, one for a sign-up page and one for a login page.

Sign Up Page:

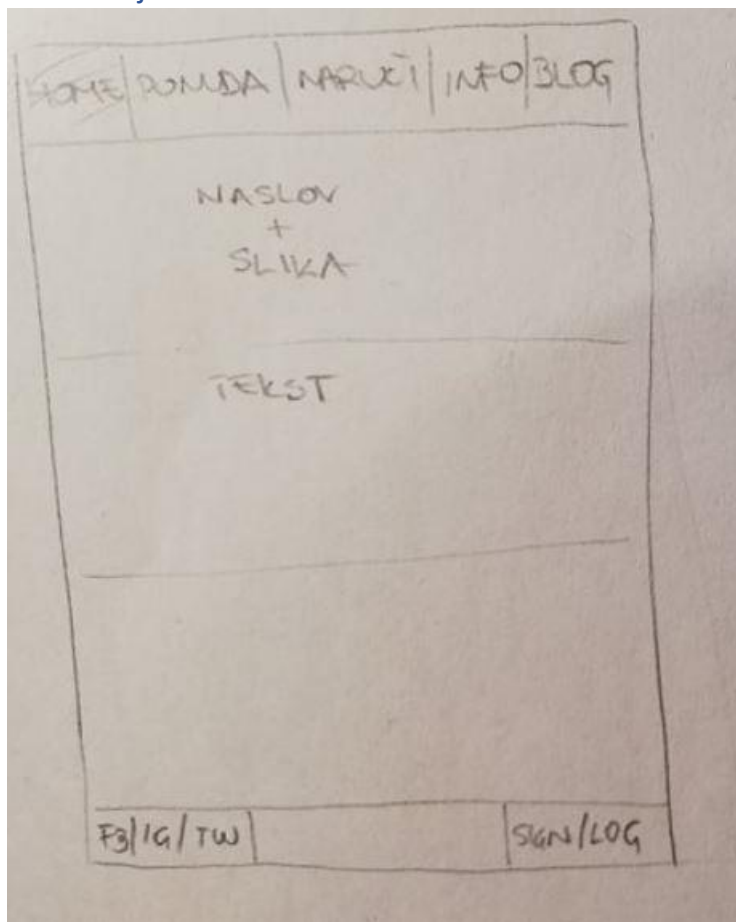
- Header: LOGO | HOME | POMA | NARUK | INFO | BLOG
- Title: SIGN UP
- Form fields:
 - IME:
 - MAIL:
 - LOZINKA:
 - PONOVLOZI:
- Buttons: PONIŠTI, SIGN UP
- Footer: SIGN UP / LOG

Log In Page:

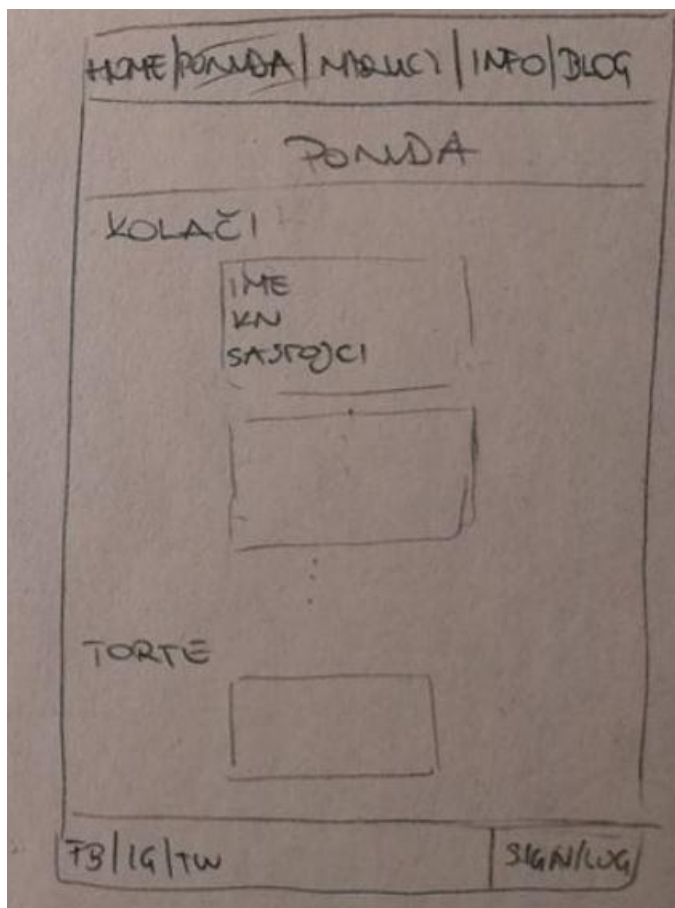
- Header: LOGO | HOME | POMA | NARUK | INFO | BLOG
- Title: LOG IN
- Form fields:
 - MAIL:
 - LOZINKA:
- Buttons: PONIŠTI, LOG IN
- Footer: SIGN UP / LOG

Slika 6 Low fidelity prototip - Stranice s formama za prijavu i registraciju

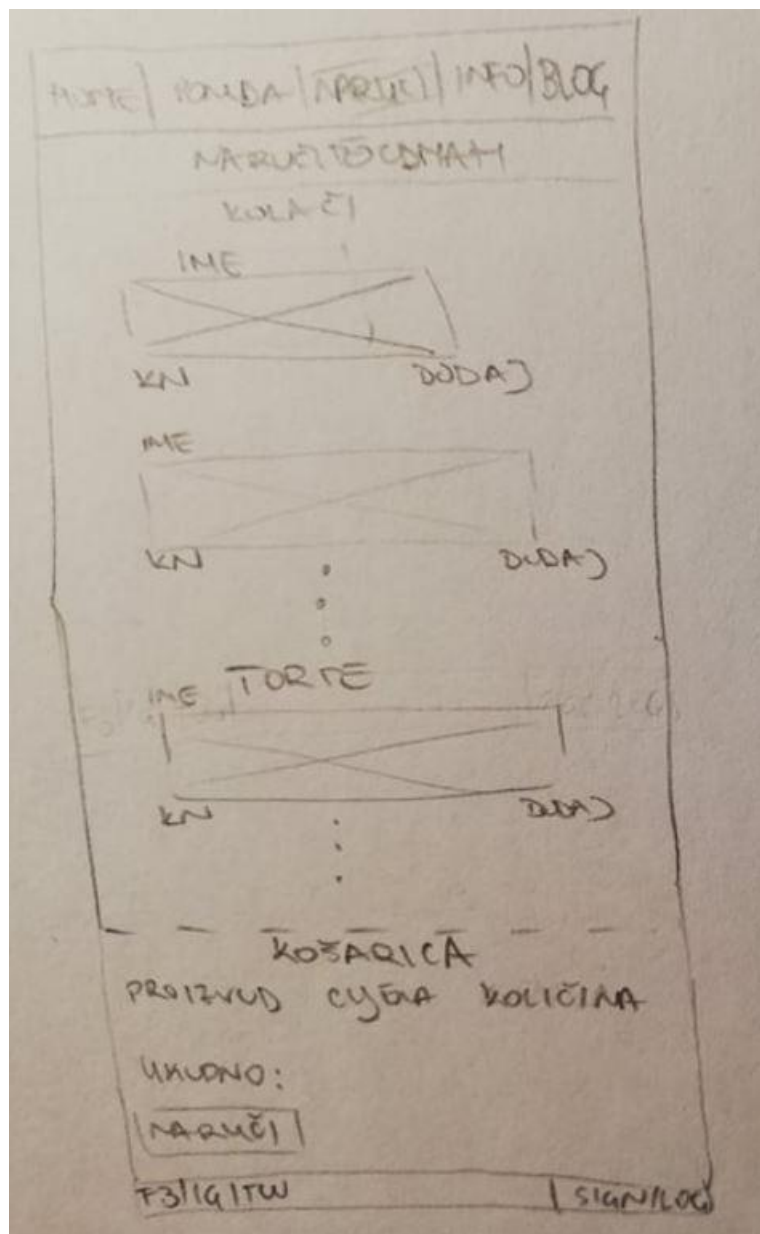
Mobilna verzija



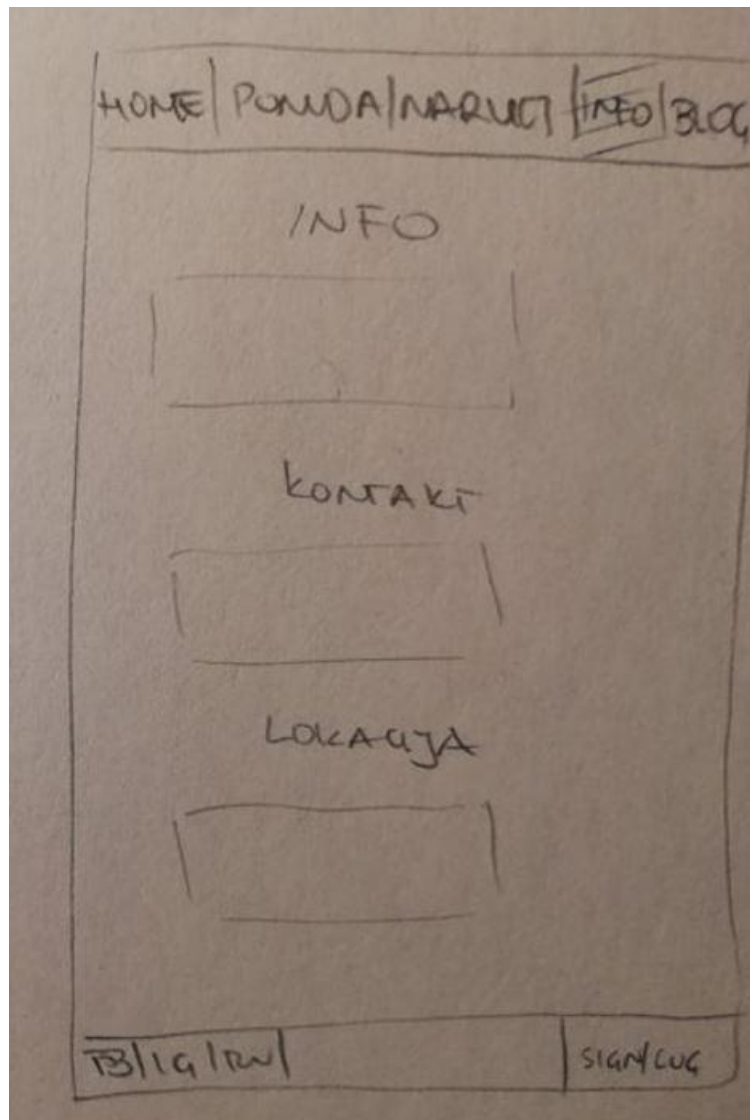
Slika 7 Low fidelity prototip - Početni zaslon na mobilnom uređaju



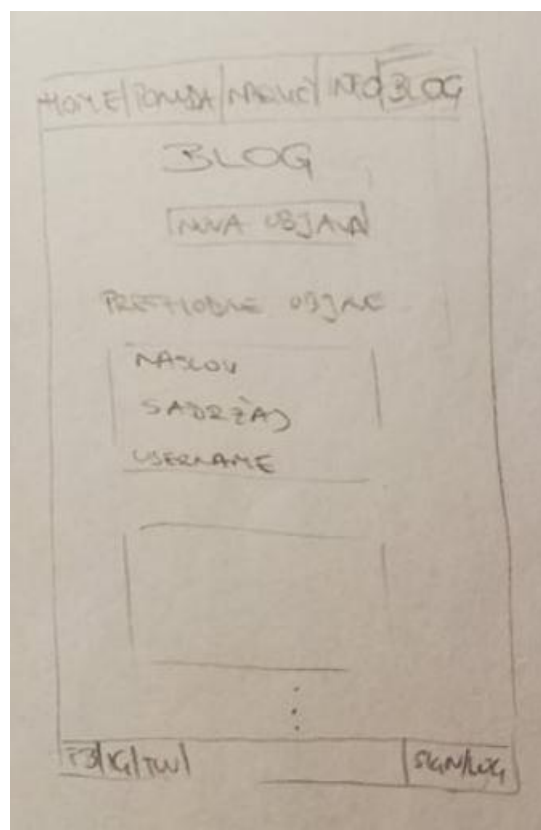
Slika 8 Low fidelity prototip - Prikaz ponude torti i kolača



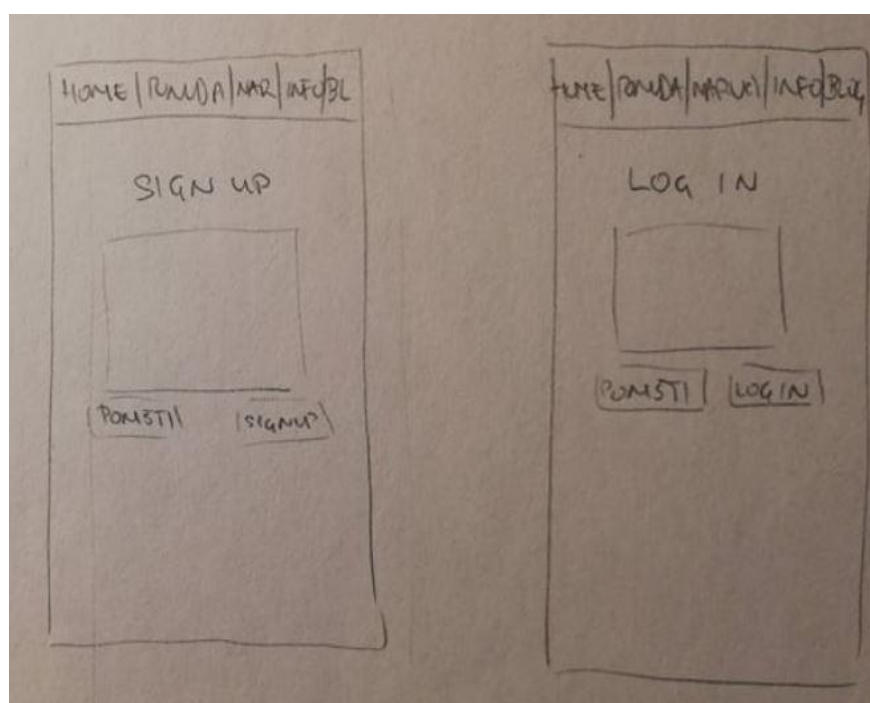
Slika 9 Low fidelity prototip – Prikaz stranice za naručivanje



Slika 10 Low fidelity prototip – Prikaz stranice s informacijama



Slika 11 Low fidelity prototip - Prikaz stranice Blog na mobilnom uređaju



Slika 12 Low fidelity prototip - Prikaz stranice za registraciju i logiranje

3. HIGH FIDELITY PROTOTIP

Nakon izrade *low fidelity* prototipa, slijedi izrada *high fidelity* prototipa. Za izradu prototipa korištena je aplikacija Figma s kojom se može simulirati ponašanje sučelja koje se dizajnira.

Desktop verzija



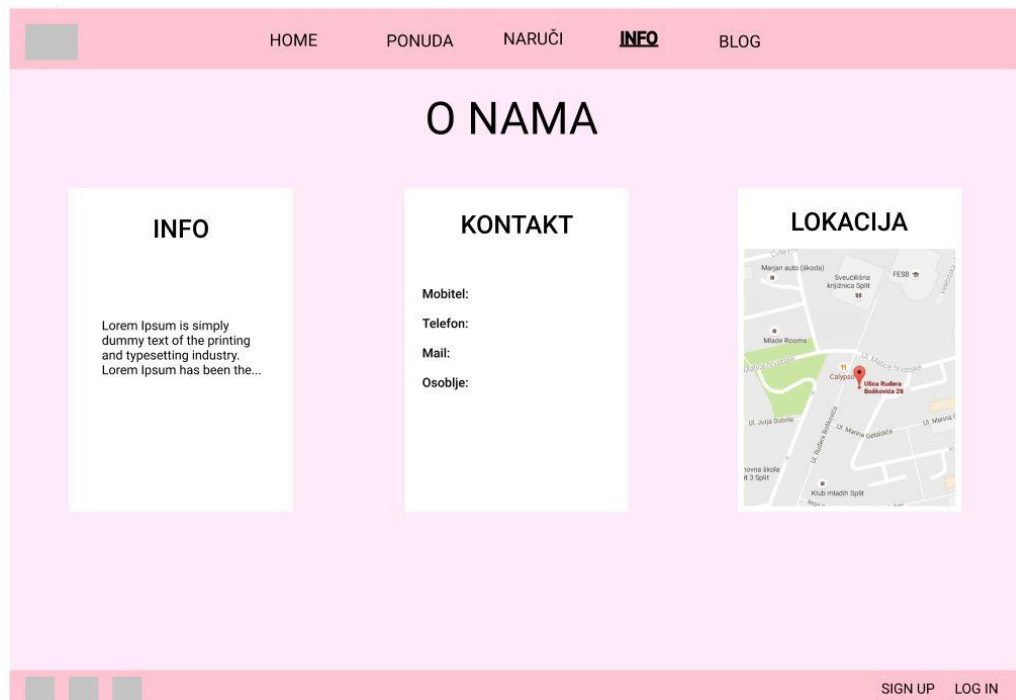
Slika 13 High fidelity prototip - Početna stranica



Slika 14 High fidelity prototip – Stranica s ponudom



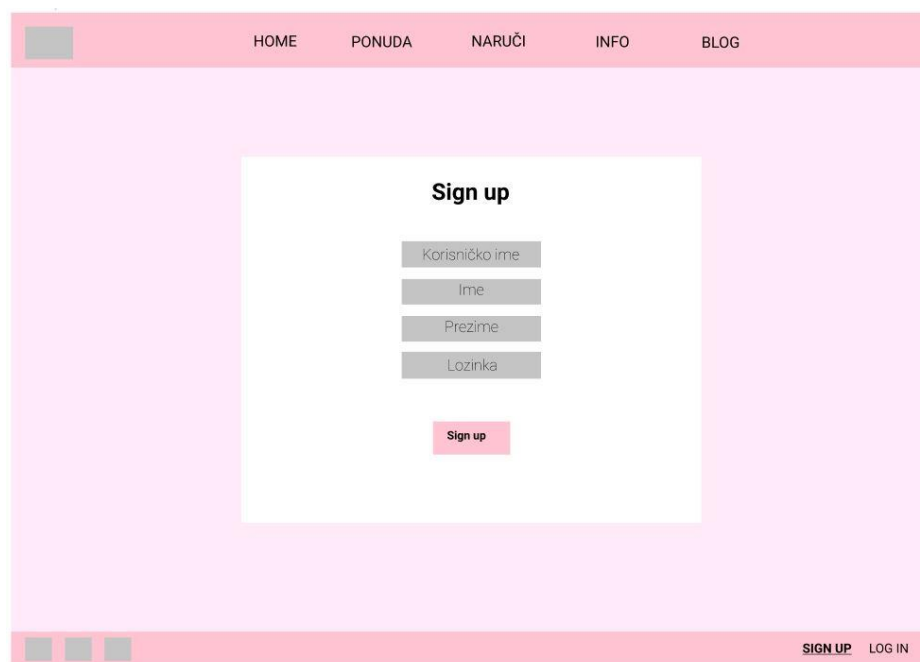
Slika 15 High fidelity prototip –S tranica za naručivanje



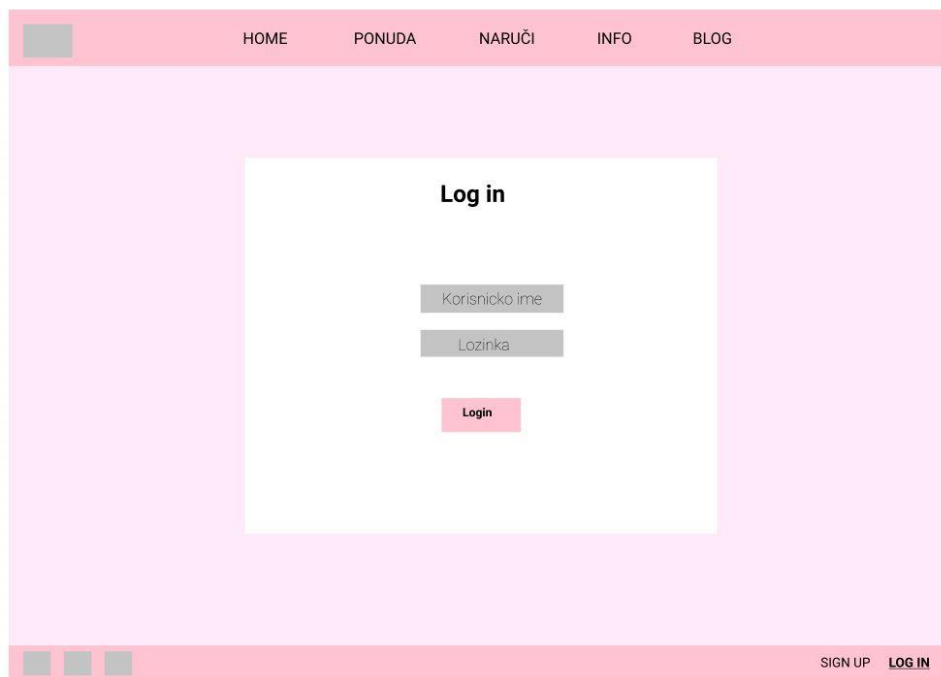
Slika 16 High fidelity prototip – Prikaz stranice s informacijama



Slika 17 High fidelity prototip – Prikaz stranice Blog



Slika 18 High fidelity prototip – Prikaz stranice za registraciju



Slika 19 High fidelity prototip – Prikaz stranice s informacijama

Mobilna verzija



Slika 20 High fidelity prototip - Početni zaslon na mobilnom uređaju

4. PRIMJENA PRINCIPA DOBROG DIZAJNA

Prototip je napravljen u Gatsbyu. Gatsby je generator statičkih stranica koji koristi React kao osnovu. Također, korišten je GraphQL za posluživanje podataka kako bi se stranice mogle izgraditi. Prilikom izrade stranice pazilo se da sučelje zadovoljava osnovne principe dobrog dizajna.

Heuristike

Za analizu sučelja koristit ćemo 10 heuristika koje su predstavljene u knjizi „Heuristička procjena korisničkog sučelja“ autora Jacoba Nielsena i Rolfa Molicha. Heuristike su smjernice pomoću kojih možemo testirati kvalitetu našeg korisničkog sučelja.

Podudaranje stvarnog svijeta sa sustavom

Ova smjernica savjetuje da bi sustav trebao „govoriti jezikom korisnika“, odnosno, izbjegavajući stručnu terminologiju, stranica treba biti razumljiva svakom prosječnom korisniku.

Web aplikacija Dolce Vita koristi hrvatski jezik i jednostavne riječi engleskog jezika koje nisu stručne terminologije. Ovaj pristup omogućava da aplikaciju mogu koristiti svi korisnici bez obzira na razinu informatičkog znanja.

Konzistencija i standardi

Ova heuristika nam nalaže da korisnici ne smiju pitati znače li različite riječi, situacije ili akcije isto, nego sustav mora slijediti inicijalno postavljane konvencije. Font, pozadina, stil i druge iste stvari bi trebale biti jednake kroz cijelu stranicu.

Budući da je naš sustav web aplikacija, koriste se standardi za prikaz informacija na internetu. Kroz cijelu aplikaciju proteže se isti font i ista pozadina.

Vidljivost statusa sustava

Ova smjernica nam govori da bi sustav u svakoj situaciji trebao informirati korisnika o tome što se događa kroz prikladni *feedback*.

Aplikacija korisnika obavještava o poziciji u navigacijskoj traci tako što je riječ podebljana i podcrtana te u URL-u piše gdje se nalazimo. U primjeru ispod <http://localhost:8000/info>.



Slika 21 Primjer vidljivosti statusa sustava

Sloboda korisnika i korisnička kontrola

Kod ove heuristike treba voditi računa o tome što korisnici mogu slučajno odabrati funkcionalnost koju nisu namjeravali te im trebamo omogućiti „izlaz u nuždi“ kako bi mogli s lakoćom napustiti neželjene opcije i brzo se vratiti na prethodno stanje stranice.

Sloboda korisnika ostvarena je preko trake izbornika, korisnik može uvijek pristupiti svim dijelovima aplikacije. Pritiskom na logo aplikacije, u slučaju potrebe, korisnik se vraća na početnu stranicu.

Kao primjer navodimo mogućnost vraćanja na početnu stranicu i ponavljanje postupka izbora sljedeće stranice. U web aplikaciji Dolce Vita *header* korisniku omogućava povratak na bilo koju stranicu po izboru.



Slika 22 Primjer korisničke kontrole

Sprečavanje grešaka

Ova smjernica savjetuje da je bitnije dizajnirati sustav koji sprječava pojavljivanje grešaka nego implementirati dobre obavijesti o greški. Važno je eliminirati dijelove sučelja sklonije greškama ili tražiti potvrde korisnika tijekom izvršavanja akcija.

Aplikacija je dizajnirana sukladno s ovom heuristikom stoga postoje samo jednostavne funkcionalnosti kod kojih ne bi trebalo dolaziti do grešaka prilikom korištenja.

Detekcija i oporavak od greške

Kod ove heuristike važno je da obavijesti o greškama moraju biti prikazane u razumljivom jeziku, prikaz problema mora biti očit i uključivati potencijalni način rješavanja problema.

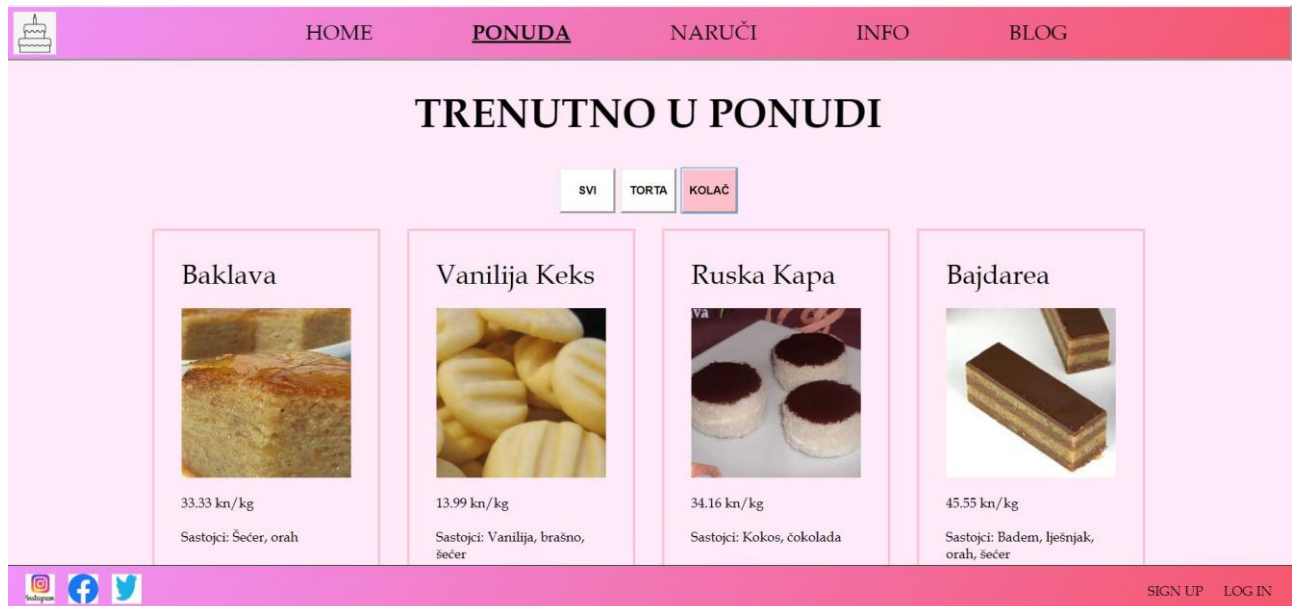
Primjer pridržavanja ove smjernice je obavijest o grešci prilikom prijave u sustav. Ako korisnik ne unese podatke u sva polja, sustav ga obavijesti da ih ispuni.

Slika 23 Primjer detekcije greške

Prepoznavanje umjesto prisjećanja

Ova smjernica nalaže da sustav treba minimizirati količinu informacija, objekata, akcije i opcija koje korisnik mora zapamtiti. Korisnik ne smije biti prisiljen pamti informacije između dijaloga. Uputu trebaju uvijek biti vidljive.

Kao primjer ove heuristike na stranici ponude imamo za odabir vrste slastice, odnosno korisnik bira hoće li pretraživati sve, samo torte ili samo kolače. Njegovim odabirom gumb promijeni boju te korisnik zna što pretražuje i što može naći.



Slika 24 Primjer prepoznavanja

Fleksibilnost i efikasnost korištenja

Ova heuristika nam govori da sustav treba biti takav da dobro odgovara i novom i uhodanom korisniku.

Estetika i minimalistički dizajn

Ova smjernica savjetuje da dijalozi ne smiju sadržavati informacije koje u tom trenutku nisu bitne ili se rijetko koriste. Svaka se dodatna informacija smanjuje vidljivost važnijih informacija.

Dizajn aplikacije je zamišljen minimalistički tako da korisnik ne bi bio preplavljen raznim informacijama. Sve informacije koje postoje za pojedine stvari su kratke i jasne bez detaljnog opisivanja.

Pomoć i dokumentacija

Zadnja heuristika nam govori da bi uz sustav trebala postojati dokumentacija koja bi korisniku pomogla da koristi sustav.

Za aplikaciju Dolce Vita ne postoji dokumentacija koja bi korisniku pomogla kako bi je mogao koristiti. No, pošto su sve prijašnje heuristike zadovoljene, prosječni korisnici ne bi trebali imati nekakvih problema s korištenjem aplikacije.

C.R.A.P. principi

C.R.A.P. principi su korišteni kako bi se postigao što bolji dizajn aplikacije. Svaki od tih principa objašnjen je i prikazan na primjeru web aplikacije Dolce Vita u sljedećim poglavljima:

Contrast

Princip kontrasta nalaže da elementi drugačije svrhe trebaju biti naglašeni na različite načine. Dominantniji elementi trebaju biti više naglašeni.

Na primjeru web aplikacije Dolce Vita kontrast je najviše vidljiv time što su važniji naslovi označeni većim fontom te su više istaknuti nego drugi tekst na stranici, header i footer su tamnije boje kako bi se istakla razlika između njih i centra stranice.



Slika 25 Primjer kontrasta

Repetition

Suprotan koncept od kontrasta, govori o tome kako slični elementi sučelja trebaju biti prikazani na isti način.

Primjeri korištenja *repetitiona*:

- Jedan osnovni tip slova
- Jedna osnovna boja
- Naslov stranice se uvijek nalazi na sredini
- Svaka kartica koja prikazuje tortu ili kolač izgleda jednako



Slika 26 Primjer kartica na stranici Blog

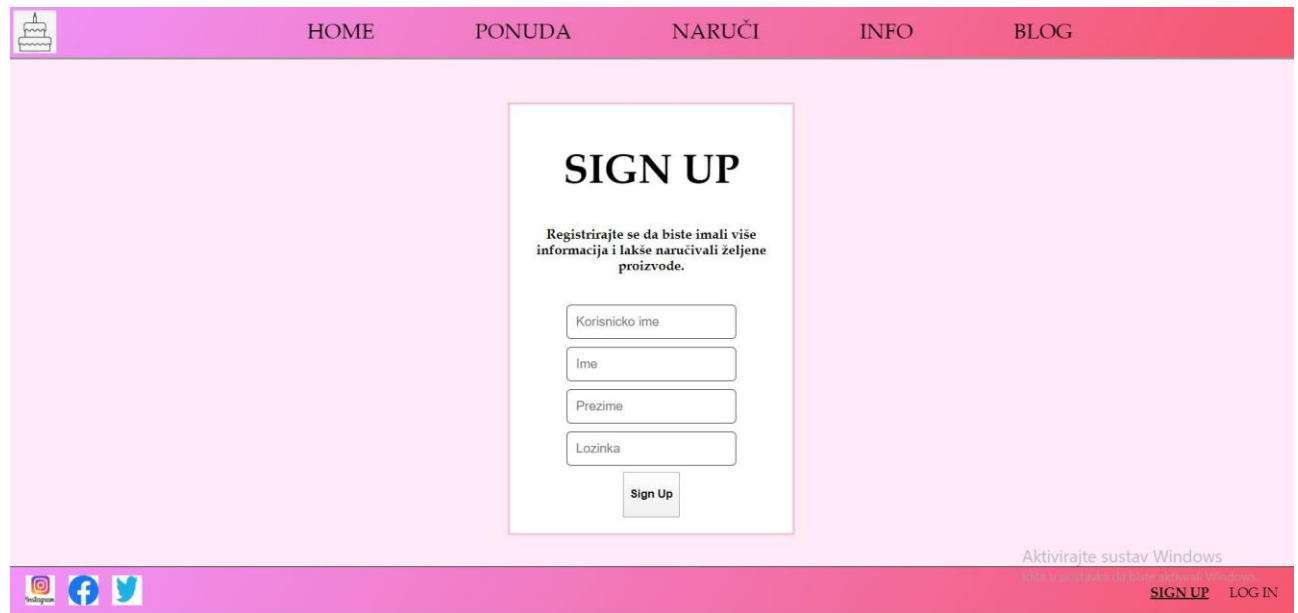
Alignment

Ništa na stranici ne smije biti na proizvoljnom mjestu. Čak kada i ne postoje linije koje povezuju elemente naše oko i mozak ih stvaraju i bolje reaguju kada su elementi vizualnog sučelja međusobno ovisno pozicionirani.

U aplikaciji pravilo poravnavanja primjenjuje se na svim prozorima, ovisno o elementima i njihovom grupiranju. Kao primjer u našoj aplikaciji možemo uzeti kartice ponuda unutar kojih su ime, slike, cijena i sastojci. Također primjer su i stranice za logiranje i registriranje kod kojih se koristilo centralno pozicioniranje.



Slika 27 Primjer poravnavanja



Slika 28 Primjer poravnavanja

Proximity

Važno je sučelje obogatiti bijelim prostorima na način da elemente koji su međusobno povezani vizualno približimo jedne drugima i onda te cjeline međusobno odvojimo praznim prostorom.

Kao primjer imamo na stranici novosti kod prikaza kartica gdje se uočava povezanost naslova i datuma.

Voćna keks torta 
11/05/2019
Torta od bijele čokolade i malina 
10/29/2019

Slika 29 Primjer proximity principa

5. ZAKLJUČAK

Kroz rad je postalo jasno koliko je bitno napraviti *low-fidelity* i *high-fidelity* prototip te koliko je zapravo teško na početku predvidjeti kako će sve izgledati. Zbog toga što tek kroz rad shvatimo zašto bi se nešto moglo napraviti drukčije i kvalitetnije postoje razlike između prototipa i stvarne web aplikacije. Također, korištenjem heuristike i C.R.A.P. principa osiguravamo dobar dizajn sučelja te to da korisnici ne bi trebali imati poteškoća prilikom korištenja aplikacije.