

Mladi za napredek Maribora 2018

35. srečanje

APLIKACIJA ZA PREVERJANJE TEŽE ŠOLSKE TORBE

Raziskovalno področje:

Računalništvo

Inovacijski predlog

Avtor: AMADEJA REK

Mentor: MARKO PODPEČAN, PATRIK REK

Šola: OŠ ANGELA BESEDNJAKA MARIBOR

Maribor, februar 2018

KAZALO

ZAHVALA	5
POVZETEK	6
SEZNAM UPORABLJENIH SIMBOLOV IN KRATIC	7
1 UVOD.....	8
2 METODOLOGIJA DELA.....	9
2.1 Ideja.....	9
2.2 Izbira programskega jezika	9
2.3 Zbiranje informacij.....	9
3 TEORETIČNI DEL.....	10
3.1 Računalniško programiranje	10
3.1.1 JavaScript.....	10
3.1.2 jQuery	11
3.1.3 PHP	11
3.2 HTML	11
3.3 CSS	12
3.4 Šolska torba	12
4 IMPLEMENTACIJA	14
4.1 PHP-koda.....	14
5 RAZISKAVA.....	33
5.1 Anketa	33
5.2 Tehtanje torbe	35
6 RAZPRAVA	36
7 DRUŽBENA ODGOVORNOST	37
8 SKLEP.....	38
9 VIRI.....	39
PRILOGA	40

KAZALO SLIK

Slika 1: Povezava	14
Slika 2: Omogočanje uporabe šumnikov	14
Slika 3: Neuspešna povezava	14
Slika 4: Začetek drugega PHP-dokumenta.....	15
Slika 5: Osrednji del strani – naslovi	15
Slika 6: Obrazec POST-metoda	15
Slika 7: Ime in priimek učenca.....	16
Slika 8: Teža učenca.....	16
Slika 9: Izbira razreda	16
Slika 10: Teža prazne torbe.....	17
Slika 11: Gumb za potrditev.....	17
Slika 12: Končni izgled obrazca.....	17
Slika 13: Pojasnilo teže prazne torbe	18
Slika 14: Noga programske kode	18
Slika 15: Zaključek programske kode.....	18
Slika 16: Končni izgled prve strani	18
Slika 17: Začetek tretjega PHP-dokumenta	19
Slika 18: Priključitev baza.php.....	19
Slika 19: Napaka pri pošiljanju obrazca.....	19
Slika 20: Spremenljivki za težo in razred.....	20
Slika 21: Spremenljivka \$sql.....	20
Slika 22: Spremenljivka \$sql_teza	21
Slika 23: Spremenljivka \$result_teza in polje.....	21
Slika 24: Seštevek potrebščin in torbe	21
Slika 25: Rezultat	21
Slika 26: Razred glava.....	22
Slika 27: Izgled zgornjega dela druge strani	22
Slika 28: Preverjanje teže.....	23
Slika 29: Ponedeljek.....	23
Slika 30: Implementacija gumba prikaži/skrij	23
Slika 31: Izpisovanje potrebščin	24
Slika 32: Razred "dan"	24
Slika 33: Switch-stavek za ime dneva.....	24
Slika 34: Switch-stavek za težo.....	25
Slika 35: Razred "potrebščina"	25
Slika 36: Ni rezultatov.....	26
Slika 37: Zaključek programa	26
Slika 38: Izgled telesa prve strani	26
Slika 39: Prvi naslov prve strani	27
Slika 40: Tretji naslov prve strani	27
Slika 41: Pisava	27
Slika 42: Izgled noge.....	28
Slika 43: Izgled telesa druge strani	28
Slika 44: Prvi naslov druge strani	28
Slika 45: Drugi naslov razreda dan	29

Slika 46: Izgled razreda dan	29
Slika 47: Izgled gumbov prikaži/skrij	29
Slika 48: Obarvan gumb.....	30
Slika 49: Neobarvan gumb	30
Slika 50: Torba ni pretežka	30
Slika 51: Okvir »pretežko«	30
Slika 52: Drugi naslov »pretežko«.....	31
Slika 53: Torba je pretežka.....	31
Slika 54: Pisava druge strani	31
Slika 55: Razred »potrebščina«.....	31
Slika 56: Izpisana potrebščina.....	32
Slika 57: Končni izgled druge strani 1	32
Slika 58: Končni izgled druge strani 2	32

KAZALO GRAFOV

Graf 1: Uporaba aplikacije	33
Graf 2: Uporabnost aplikacije	34

KAZALO TABEL

Tabela 1: Teža torbe.....	35
---------------------------	----

ZAHVALA

Rada bi se zahvalila mentorjema, ki sta me pri delu usmerjala in mi pomagala. Zahvala gre tudi knjižničarki in učencem, ki so mi pomagali pri tehtanju potrebščin in staršem, ki so testirali aplikacijo ter odgovorili na kratko anketo. Hvala tudi šolski svetovalni delavki, ki je anketo posredovala staršem.

POVZETEK

Skozi leta postaja govor o pretežkih šolskih torbah vse bolj pogost. Torba naj ne bi presegla desetine otrokove teže, da ne škoduje zdravju otroka. Pravo težo torbe je težko določiti »po občutku«, zato bi morali starši torbe tehtati.

Da bi staršem naše šole čim bolj olajšala delo, sem naredila aplikacijo za preverjanje teže šolske torbe, v katero so že vnesene teže učbenikov, zvezkov in ostalih potrebščin za posamezne predmete. Tako lahko le z nekaj kliki starši preverijo težo otrokove torbe za posamezne dni, s tem pa še pravi čas ukrepajo v zvezi z otrokovim zdravjem.

SEZNAM UPORABLJENIH SIMBOLOV IN KRATIC

CSS – kaskadne stilske podloge, slogovni jezik za oblikovanje spletnih vmesnikov (angl. *Cascading Style Sheets*)

HTML – označevalni jezik za izdelavo spletnih strani (angl. *Hyper Text Markup Language*)

JS – objektni skriptni programski jezik (angl. *JavaScript*)

MySQL – sistem za upravljanje s podatkovnimi bazami

PHP – strežniški skriptni jezik (angl. *Hypertext Preprocessor*)

SQL – strukturirani povpraševalni jezik za delo s podatkovnimi bazami (angl. *Structured Query Language*)

XHTML – označevalni jezik, ki ima enak namen kot HTML, vendar je usklajen s sintakso XML (angl. *Extensible HyperText Markup Language*)

1 UVOD

V današnjih dneh postaja problem pretežkih šolskih torb vse bolj pogost. Ljudje o tem veliko govorijo, ukrenejo pa nič. Mnogo staršev se niti ne potruzi pozanimati o poškodbah in posledicah pretežke torbe, kaj šele, da bi si vzeli čas in torbo stehali. Otrokovo zdravje je pomembno, posledice pretežkih torb lahko pripeljejo do resnih in dolgotrajnih poškodb.

Danes je tehnologija že zelo razvita, uporabljamo jo vsi. Veliko časa preživimo pred računalniki, telefoni, tablicami ... Zato sem dobila idejo, da bi težavo povezala z računalništvom. Starši bi lahko s pomočjo računalnika ali telefona na hiter način ugotovili ali je teža otrokove šolske torbe primerna, aplikacija pa bi bila dostopna tudi otrokom.

Moj cilj je bil, da bi starši aplikacijo znali uporabljati, ker bi bila dovolj enostavna in bi se jim zdela koristna, izračunana teža pa bi bila dokaj natančna oz. podobna dejanski teži torbe.

Preverila sem ali takšna aplikacija že obstaja, a nisem našla nobene podobne. To me je še bolj spodbudilo k zbiranju idej in ustvarjanju aplikacije. Ugotovila sem, da bo ta staršem in otrokom najlažje dostopna na spletu. Za ustvarjanje spletne aplikacije sem si izbrala programski jezik PHP, saj se mi je zdel dovolj enostaven za učenje, a hkrati najbolj primeren za izdelavo.

Tako se je začela gradnja aplikacije, za katero sem upala, da bo pomagala staršem in otrokom.

2 METODOLOGIJA DELA

2.1 Ideja

Najprej sem na papir zbrala svoje ideje, kaj sploh želim narediti, kakšen končni izgled aplikacije si želim itd. Nato sem ideje pokazala mentorju in začela z natančnejšim načrtovanjem.

Za začetek sem se odločila, da bom naredila aplikacijo le za našo šolo, predvsem pa sem bila osredotočena na svoj razred.

2.2 Izbira programskega jezika

Prišla sem do prvega bistvenega vprašanja, in sicer, kateri programski jezik izbrati?

Izbrala sem PHP, saj je ta enostaven za uporabo. Jezik sem se učila že pred začetkom ustvarjanja mojega inovacijskega predloga, zato mi je bilo delo olajšano. S pomočjo mentorja sem se naučila še nekaj dodatnih veščin PHP-ja, ki sem jih potrebovala za uspešno izdelavo aplikacije. Podatke – teža, potrebščine, urnik ... – sem morala zapisati v podatkovno bazo. Za to sem uporabila MySQL, sistem za upravljanje s podatkovnimi bazami, ki uporablja jezik SQL.

Za oblikovanje vmesnika aplikacije sem uporabila CSS, ki sem ga prav tako že poznala. Uporabljala sem še JavaScript in jQuery. Pri učenju navedenih jezikov sem si pomagala s spletno stranjo W3Schools.

2.3 Zbiranje informacij

Preden sem lahko začela s programiranjem svoje aplikacije, sem morala zbrati potrebne informacije. Stehtala sem vse učbenike, delovne zvezke, zvezke, prazno torbo ter nekatere druge potrebščine, ki jih učenci potrebujejo. Pri tem mi je pomagala šolska knjižničarka.

Podatke sem vpisala v MySQL-bazo, ustvarila sem tabele »potrepscine«, »predmeti«, »razredi«, »tip« in »urnik«. V tabelo »urnik« sem zapisala še urnik vsakega razreda.

Podatki so bili vpisani v bazo in tako sem pričela zapisovati programsko kodo.

3 TEORETIČNI DEL

3.1 Računalniško programiranje

Program je nabor navodil, ki po korakih usmerjajo računalnik, da opravlja naloge, ki jih želite narediti in ustvari zelene rezultate. Skupine pravil, ki računalniku povedo, katere operacije je treba izvesti, se imenujejo programski jeziki. Obstaja več programskih jezikov, kot so Java, C++, Pascal ... [1]

Programiranje je metoda za pisanje kode na visoki ravni, vendar še vedno berljivi obliki za programerje. Večina programskih jezikov danes uporablja prevajalce, to je vir (napisan kot skript), preveden v računalniško kodo. Predstavljeno je na način, ki ga lahko izvrši računalnik, npr. kot izvršljive datoteke. [2]

V programskem procesu (razvoju programa) je 5 glavnih korakov:

1. Opredelitev problema;
2. Načrtovanje rešitve;
3. Kodiranje programa;
4. Testiranje programa;
5. Dokumentiranje programa. [1]

Spodaj so opisani jeziki, ki sem jih uporabljala.

3.1.1 JavaScript

JavaScript (JS) je polnopravni dinamični programski jezik, ki lahko zagotovi dinamično interaktivnost na spletnih straneh. Sicer je najbolj znan kot skriptni jezik za spletne strani, vendar se lahko uporablja tudi v mnogih okoljih, ki niso v brskalniku. Gre za dinamičen protokol, ki temelji na multi-paradigmu, ki podpira objektno orientirane, nujne in funkcionalne stile programiranja. JS se izvaja na odjemalski strani spleta, ki se lahko uporablja za oblikovanje oz. programiranje obnašanja spletnih strani ob pojavu dogodka. [3]

3.1.2 jQuery

jQuery je okvir JavaScript-a, katerega namen je lažja uporaba le-tega na spletnem mestu. Lahko bi ga opisali kot abstrakcijsko plast, saj vsebuje veliko funkcionalnosti, ki bi jih morali napisati v številnih vrsticah JavaScript-a, da bi to dosegli, in jih vključili v funkcije, ki jih lahko pokličemo z eno samo vrstico kode. jQuery nam torej ponuja nekaj sintaktičnih bližnjic, koda pa je še vedno JavaScript-koda. [4]

3.1.3 PHP

PHP je široko uporabljen odprtokodni skriptni jezik, ki je še posebej primeren za pisanje spletnih strani. PHP-skripte se izvajajo na strežniku, rezultat pa se pošlje v brskalnik kot navaden HTML. Mogoče ga je povezati s številnimi bazami podatkov (npr. MySQL, PostgreSQL, Oracle ...) in lahko vključuje običajne HTML-spletne strani. Tudi nekatere svetovno znane spletne strani, kot so *Facebook*, *Yahoo*, *Flickr* in *Wikipedija*, so ustvarjene z jezikom PHP. [5]

Oznake, ki sem jih v predlogu uporabljala:

- `<?php ?>` za oznako PHP-dokumenta.
- `$`spremenljivke za shranjevanje informacij.
- `$_POST`-spremenljivko za sprejemanje podatkov iz obrazca.
- `If...else` -ogojne stavke, ki izvedejo eno kodo, če je pogoj izpolnjen in drugo, če ni.
- `While`-zanke, ki izvajajo neko kodo, dokler je pogoj resničen.
- `Echo`-oznako za izpisovanje besedila.
- `Switch`-stavke za izvajanje različnih dejavnosti ob različnih pogojih.

3.2 HTML

HTML je označevalni jezik, namenjen ustvarjanju spletnih strani. Te si nato lahko ogleda kdor koli, ki je povezan z internetom. Sestavljen je iz vrste kratkih kod – oznak (angl. *tags*), ki jih avtor vnese v besedilno datoteko, shranjeno kot HTML-datoteka. Brskalnik bere datoteko in prevede tekst v vidno obliko. Poleg HTML-ja uporabljamo tudi številne druge podporne jezike, ki omogočajo nove stvari, kot npr. kaskadne slogovne predloge (CSS), ki se uporabljajo za

nadzor nad tem, kako bo stran predstavljena in jo naredijo bolj dostopno, ter JavaScript, ki zagotavlja osnovne posebne učinke in interakcije. [6]

Oznake, ki sem jih uporabljala:

- <!DOCTYPE html> in </html> za oznako HTML5-dokumenta;
- <head> in </head> za oznako glave dokumenta;
- <title> za naslov strani;
- <body> in </body> za oznako telesa/osrednjega dela dokumenta;
- <h> za naslove;
- <i> za poševno (*italic*) besedilo;
- <form> oz. obrazce za pridobitev nekaterih podatkov otroka (npr. teža, razred ...).

3.3 CSS

Cascading Style Sheets oz. CSS je preprost oblikovni jezik, ki naj bi poenostavil postopek izdelovanja predstavitev spletnih strani.

CSS obravnava videz in občutek dela spletne strani. Z njim lahko nadziramo barvo besedila, slog pisave, razmik med odstavki ...

Najpogosteje je kombiniran s HTML ali XHTML. [7]

Medtem ko se HTML uporablja za strukturiranje vsebine, se CSS uporablja za oblikovanje strukturirane vsebine. CSS je bil ustvarjen, da bi spletnim oblikovalcem omogočil prefinjene možnosti postavitve, ki jih podpirajo vsi brskalniki. Hkrati pa ločevanje slogov predstavitev vsebine dokumentov olajša vzdrževanje spletnega mesta. [8]

3.4 Šolska torba

Šolske torbe naj, glede na uradna priporočila, ne bi presegale 10 % otrokove teže. Če je šolska torba pretežka, lahko povzroči bolečine v hrbtenici, pokvari držo otroka in povzroči težave nadaljnjega razvoja otroka. Zato je pomembno, da je torba primerne teže. [9]

Raziskava *Naj torba ne bo borba* društva IndiJanez v letu 2016, ki je bila omenjena tudi v članku *Slovenski šolarji nosijo pretežke šolske torbe. Tako težke, da so zdravju nevarne*, je pokazala, da kar 84 % otrok dveh mariborskih šol nosi pretežno šolsko torbo. Povprečna teža šolskih torb je znašala približno 17 % otrokove teže, najvišja teža torbe pa je bila 12,3 kg, kar je predstavljalo 39 % otrokove teže. [9]

4 IMPLEMENTACIJA

4.1 PHP-koda

Najprej sem morala kodo povezati s podatkovno bazo, kot prikazuje slika 1.

```
1 <?php
2 $servername = "ime serverja";
3 $username = "uporabniško ime";
4 $password = "geslo";
5 $database = "podatkovna baza";
```

Slika 1: Povezava

Lasten arhiv, 2017

Nato sem aplikaciji, kot je prikazano na sliki 2, pri delu z bazo nastavila nabor znakov UTF-8, ki podpira uporabo šumnikov.

```
9 mysqli_set_charset($conn,"utf8"); // za šumnike
```

Slika 2: Omogočanje uporabe šumnikov

Lasten arhiv, 2017

Na sliki 3 lahko vidimo, kaj se zgodi, če povezava ne uspe.

```
11 if ($conn->connect_error) {
12     die("Connection failed: " . $conn->connect_error);
13 }
14
15 ?>
```

Slika 3: Neuspešna povezava

Lasten arhiv, 2017

V novem PHP-dokumentu sem najprej zapisala osnovno HTML-kodo, nato pa v glavi (<head>) ponovno nastavila nabor znakov UTF-8, tokrat za prikaz na spletni strani (slika 4). Potem sem

poimenovala aplikacijo in jo povezala s CSS-dokumentom, v katerem sem aplikacijo urejala (style2.css).

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <meta charset="UTF-8">
5   <title>Aplikacija za preverjanje teže šolske torbe</title>
6   <link rel="stylesheet" type="text/css" href="style2.css">
7 </head>
```

Slika 4: Začetek drugega PHP-dokumenta

Lasten arhiv, 2017

Nato sem se lotila osrednjega dela strani (<body>). Pod prvi naslov sem zapisala ime aplikacije, pod drugi pa navodila za uporabo (slika 5).

```
8 <body>
9   <h1>Aplikacija za preverjanje teže šolske torbe</h1>
10  <h3>Vnesite ime, težo in razred otroka.</h3>
```

Slika 5: Osrednji del strani – naslovi

Lasten arhiv, 2017

Naredila sem obrazec (angl. *form*), ki ga mora uporabnik izpolniti, da lahko aplikacija izračuna težo šolske torbe (slika 6). Obrazec sem povezala z naslednjim PHP-dokumentom (preveri.php) in določila metodo pošiljanja informacij obrazca POST, saj ta pošilja vrednosti preko HTTP POST-metode, kjer se podatki/vrednosti ne prikažejo v URL-ju (medtem, ko se v GET-metodi vrednosti prenašajo preko URL-ja). Tako je POST-metoda varnejša in dovoljuje prenos več znakov; (GET-metoda se uporablja predvsem, ko želimo omogočiti shranjevanje (bookmark) strani, ki ga jaz v tem primeru ne potrebujem).

```
11 <form action="preveri.php" method="post">
```

Slika 6: Obrazec POST-metoda

Lasten arhiv, 2017

Naredila sem prvo polje (slika 7), in sicer za ime in priimek učenca (z načinom vnosa besedila). Polje je zahtevano (required).

```
12 Ime in priimek učenca: <input type="text" name="ime" required="required"><br>
```

Slika 7: Ime in priimek učenca

Lasten arhiv, 2017

Naslednje polje je bilo namenjeno vnosu teže učenca (slika 8). Tokrat je bil uporabljen način vnosa števila. Tudi to polje je zahtevano (required).

```
14 Teža učenca: <input type="number" name="teza" required="required">kg<br>
```

Slika 8: Teža učenca

Lasten arhiv, 2017

Naredila sem polje z možnostjo izbire razreda otroka z <option> oznako (slika 9). Tudi to polje je zahtevano (required).

```
16 Razred: <select name="razred" required="required">
17   <option value="1a">1. a</option>
18   <option value="1b">1. b</option>
19   <option value="2a">2. a</option>
20   <option value="2b">2. b</option>
21   <option value="2c">2. c</option>
22   <option value="3a">3. a</option>
23   <option value="3b">3. b</option>
24   <option value="3c">3. c</option>
25   <option value="4a">4. a</option>
26   <option value="4b">4. b</option>
27   <option value="4c">4. c</option>
28   <option value="5a">5. a</option>
29   <option value="5b">5. b</option>
30   <option value="6a">6. a</option>
31   <option value="6b">6. b</option>
32   <option value="7a">7. a</option>
33   <option value="7b">7. b</option>
34   <option value="8a">8. a</option>
35   <option value="8b">8. b</option>
36   <option value="9a">9. a</option>
37   <option value="9b">9. b</option>
38 </select><br>
39
```

Slika 9: Izbira razreda

Lasten arhiv, 2017

V naslednje polje je prav tako potrebno vpisati število (slika 10). Tukaj mora uporabnik vnesti težo prazne torbe oz. torbe z le osnovnimi potrebščinami (peresnico, mapo, dodatnimi potrebščinami potrebnimi za pouk). Že teža prazne torbe se pri vsakem otroku razlikuje, saj poznamo veliko različnih torb. Tako bodo rezultati bolj realni. Je pa v polje že vnesena približna teža prazne torbe (2000 g). Polje je zahtevano (required).

```
40 Teža prazne torbe z osnovnimi potrebščinami*: <input type="number" name="torba" required="required" value="2000">g<br>
```

Slika 10: Teža prazne torbe

Lasten arhiv, 2017

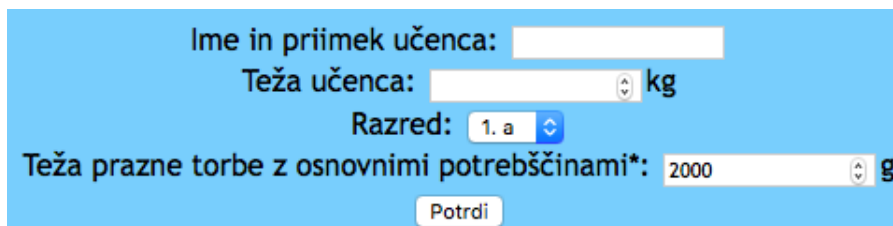
Nato sem vključila še gumb za potrditev (slika 11) in zaključila obrazec (form).

```
42 <input type="submit" value="Potrdi">  
43 </form>
```

Slika 11: Gumb za potrditev

Lasten arhiv, 2017

Končen izgled obrazca lahko vidimo na sliki 12.



Ime in priimek učenca:
Teža učenca: kg
Razred:
Teža prazne torbe z osnovnimi potrebščinami*: g

Slika 12: Končni izgled obrazca

Lasten arhiv, 2017

Podala sem pojasnilo teže prazne torbe z osnovnimi potrebščinami, saj je pri testiranju pri nekaterih starših prišlo do nejasnosti (slika 13).

```
43 <p style="font-size: 80%;">*Sem je všteta sama torba, peresnica, mapa, beležka, copati in ostale potrebščine, ki jih učenec naknadno potrebuje.</p>
```

Slika 13: Pojasnilo teže prazne torbe

Lasten arhiv, 2017

Z razredom sem označila nogo (footer) strani, v katero sem vključila priimka mentorjev in sebe (slika 14).

```
45 <div class="footer">
```

Slika 14: Noga programske kode

Lasten arhiv, 2017

Nato sem zaključila dokument (slika 15).

```
49 </body>
50 </html>
```

Slika 15: Zaključek programske kode

Lasten arhiv, 2017

Končen izgled prve strani vidimo na sliki 16.

Aplikacija za preverjanje teže šolske torbe

Vnesite ime, težo in razred otroka.

Ime in priimek učenca:

Teža učenca: : kg

Razred: 1.5

Teža prazne torbe z osnovnimi potrebščinami*: 2000 : g

*Sem je všteta sama torba, peresnica, mapa, beležka, copati in ostale potrebščine, ki jih učenec naknadno potrebuje.

Slika 16: Končni izgled prve strani

Lasten arhiv, 2017

Nato sem naredila nov PHP-dokument in ga poimenovala preveri.php. V glavi dokumenta sem nastavila UTF-8 nabor znakov, poimenovala stran, klicala jQuery-datoteko z oznako <script> in PHP-dokument povezala s CSS-dokumentom (style.css), v katerem sem stran kasneje urejala (slika 17).

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8">
  <title>Aplikacija za preverjanje teže šolske torbe</title>
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js"></script>
  <link rel="stylesheet" type="text/css" href="style.css">
</head>
```

Slika 17: Začetek tretjega PHP-dokumenta

Lasten arhiv, 2017

Dokumentu preveri.php sem priključila dokument baza.php, v katerem so bili podatki za povezavo (slika 18).

```
34 <?php
35 include 'baza.php';
```

Slika 18: Priključitev baza.php

Lasten arhiv, 2017

Če pride do napake pri pošiljanju podatkov obrazca, se izpiše »Napaka!« in uporabnika preusmeri nazaj na prvo stran (slika 19).

```
37 if (!isset($_POST["razred"]))
38 {
39     echo "Napaka!";
40     header("Location: index.php");
41 }
42
43 else
```

Slika 19: Napaka pri pošiljanju obrazca

Lasten arhiv, 2017

V kolikor pri obrazcu oz. pošiljanju podatkov ni prišlo do napake, se program nadaljuje. Ustvarila sem spremenljivko \$teza, ki je enaka vneseni teži učenca, vendar je pomnožena s 1000 (tako kilograme spremenimo v grame), saj so tudi podatki tež potrebščin v bazi vpisani v gramih. Naslednja spremenljivka, \$razred_temp, je enaka vrednosti (value) izbranega razreda, torej brez pike med razredom in oddelkom, npr. 1a. Spremenljivka \$razred je namenjena izpisu razreda, zato sem tu dodala še presledek in piko med razredom in oddelkom (npr. 1. a). \$razred_st pa pomeni samo številčni del razreda (npr. 1), ki ga potrebujem pri zbiranju podatkov iz MySQL-baze (slika 20).

```
43     else
44     {
45
46         $teza = $_POST["teza"] * 1000;
47         $razred_temp = $_POST["razred"]; // razred skupaj brez pike
48         $razred = substr($razred_temp, 0, 1).".".substr($razred_temp, 1); // dodamo piko in presledek za izpis
49         $razred_st = substr($razred_temp, 0, 1); // samo številčni del razreda
```

Slika 20: Spremenljivki za težo in razred

Lasten arhiv, 2017

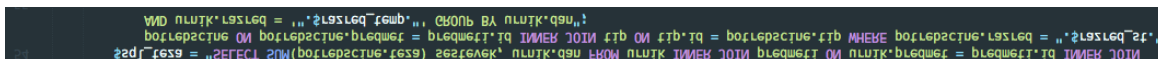
Iz baze »urnik« izberemo dan, naziv predmeta, krajšavo, naziv potrebščine, težo potrebščine in tip potrebščine (slika 21). Z »INNER JOIN« povežemo stolpec predmeta iz tabele »urnik« z id predmeta iz tabele »predmeti«, predmet iz tabele »potrebščine« z id predmeta iz tabele »predmeti«, id tipa iz tabele »tip« s tipom potrebščine iz tabele »potrebščine«, kjer (WHERE) je razred iz tabele »potrebščine« enak spremenljivki \$razred_st in, kjer je razred v tabeli »urnik« enak spremenljivki \$razred_temp. Podatke se razporedi po vrstnem redu dnevov (od ponedeljka do petka).

```
$sql = "SELECT urnik.dan dan, predmeti.naziv nazivPredmeta, predmeti.krajšava krajšava, potrebscine.naziv nazivPotrebscine, potrebscine.
teza teza, tip.naziv tip FROM urnik INNER JOIN predmeti ON urnik.predmet = predmeti.id INNER JOIN potrebscine ON potrebscine.predmet =
predmeti.id INNER JOIN tip ON tip.id = potrebscine.tip WHERE potrebscine.razred = ".$razred_st." AND urnik.razred = ".$razred_temp."
ORDER BY dan ASC";
```

Slika 21: Spremenljivka \$sql

Lasten arhiv, 2017

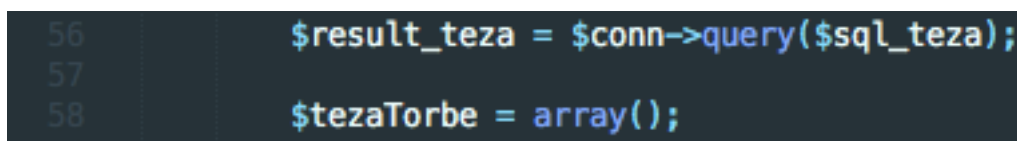
Nastavila sem spremenljivko \$sql_teza, ki predstavlja SQL-poizvedbo, ki sešteje težo vseh potrebščin za posamezni dan (slika 22). Kot v prejšnjem koraku, sem tudi tukaj ponovno povezala tabele. Podatke sem združila (GROUP BY) po dnevih.



Slika 22: Spremenljivka \$sql_teza

Lasten arhiv, 2017

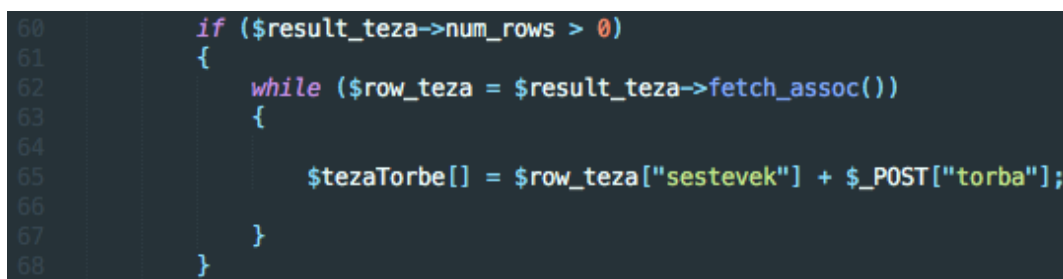
Ustvarila sem spremenljivko \$result_teza, ki predstavlja rezultat prej omenjene poizvedbe teže vseh potrebščin (\$sql_teza). Naredila sem polje (array) \$tezaTorbe (slika 23).



Slika 23: Spremenljivka \$result_teza in polje

Lasten arhiv, 2017

Polje \$tezaTorbe napolnimo s posameznimi seštevki potrebščin po dnevih (\$row_teza) in prazne torbe, ki jo je uporabnik vnesel v obrazcu (slika 24).



Slika 24: Seštevki potrebščin in torbe

Lasten arhiv, 2017

Rezultat \$result nastavimo na vrednost poizvedbe \$sql (slika 25).



Slika 25: Rezultat

Lasten arhiv, 2017

Spremenljivko \$trenutniDan sem nastavila na ponedeljek (številka 1 predstavlja prvi dan v tednu). Nato sem kodo napisala pod razred 'glava'. Pod prvim naslovom se je izpisalo ime učenca ter v oklepaju razred. V naslednji vrstici se je izpisala vnesena teža učenca v kilogramih (zato sem \$tezo pomnožila s 1000, da sem vnovič dobila kilograme). Pod drugim naslovom se je izpisala najvišja dovoljena teža šolske torbe (spremenljivko \$teza sem pomnožila z 0.1, da sem dobila 10 % otrokove teže). Nato sem še zapisala vneseno težo prazne torbe ter opozorila, da so pri posameznem učencu možna odstopanja. Prav tako sem dodala pojasnilo gumba prikaži/skrij, saj le-ta staršem pri testiranju aplikacije ni bil jasen. S tem sem zaključila razred (slika 26).

```
72     $trenutniDan = 1;
73     echo "<div class='glava'>";
74     echo "<h1>".$_POST["ime"]." ($razred)<br>";
75     echo "<h1>Vnešena teža učenca: ";
76     echo $teza/1000;
77     echo " kg</h1>";
78     echo "<h2>Najvišja dovoljena teža šolske torbe: ".$teza*0.1." g</h2>";
79     echo "V težo torbe so vštete vse potrebščine vključno s peresnico, copati in samo torbo (".$_POST["torba"]." g). Pri posameznem učencu so
80     možna odstopanja.<br>";
81     echo "(S klikom na gumb prikaži/skrij si lahko ogledate podrobnejše podatke o teži potrebščin za posamezen dan.)";
82     echo "</div>";
```

Slika 26: Razred glava

Lasten arhiv, 2017

Izgled zgornjega dela druge strani (preveri.php) vidimo na sliki 27 (ime, razred, teža učenca in torbe so izmišljeni).

Ana Novak (9. b)

Vnešena teža učenca: 55 kg

Najvišja dovoljena teža šolske torbe: 5500 g

V težo torbe so vštete vse potrebščine vključno s peresnico, copati in samo torbo (2000 g). Pri posameznem učencu so možna odstopanja. (S klikom na gumb prikaži/skrij si lahko ogledate podrobnejše podatke o teži potrebščin za posamezen dan.)

Slika 27: Izgled zgornjega dela druge strani

Lasten arhiv, 2017

V naslednjem razredu se v kodi izpiše »pretezk«, če je teža torbe v ponedeljek višja od spremenljivke \$teza pomnožene z 0.1, torej 10 % otrokove teže (slika 28).

```

81     echo "<div class='dan";
82     if ($stezaTorbe[$trenutniDan-1] > $steza*0.1)
83         echo " pretezko";
84     echo "'>";

```

Slika 28: Preverjanje teže

Lasten arhiv, 2017

Pod prvi naslov sem zapisala »Ponedeljek«. Nato se je izpisala skupna teža torbe za ta dan (v gramih). Pod tem sem naredila gumb »Prikaži/skrij«, ki prikaže/skrije podrobnejše podatke o potrebščinah, ki jih ima učenec v torbi (slika 29).

```

85     echo "<h1>Ponedeljek</h1>";
86     echo "<h2>Skupna teža torbe: ".$stezaTorbe[$trenutniDan-1]." g</h2>";
87     echo "<button id='pon'>Prikaži/skrij</button>";

```

Slika 29: Ponedeljek

Lasten arhiv, 2017

V <script> oznaki sem z uporabo JavaScripta implementirala funkcionalnost gumbov, ki prikažejo ali skrijejo podatke o teži posameznih potrebščin po dnevih (slika 30).

```

13     <script>
14         $(document).ready(function(){
15             $("#pon").click(function(){
16                 $("#potrebscinePon").toggle();
17             });
18             $("#tor").click(function(){
19                 $("#potrebscineTor").toggle();
20             });
21             $("#sre").click(function(){
22                 $("#potrebscineSre").toggle();
23             });
24             $("#cet").click(function(){
25                 $("#potrebscineCet").toggle();
26             });
27             $("#pet").click(function(){
28                 $("#potrebscinePet").toggle();
29             });
30
31         });
32     </script>

```

Slika 30: Implementacija gumba prikaži/skrij

Lasten arhiv, 2017

Začne se razred »potrebscinePon«. Dokler je število vrstic večje od nič, se rezultati izpisujejo. Če je konec trenutnega dneva, se rezultati ne izpisujejo več, trenutni dan pa se poveča za 1 (slika 31).

```
88     echo "<div id='potrebscinePon' style='display: none;'>";
89     if ($result->num_rows > 0)
90     {
91         while($row = $result->fetch_assoc())
92         {
93             if ($trenutniDan != $row["dan"]) // če je konec trenutnega dneva
94             {
95                 $trenutniDan++;
96                 echo "</div>";
97                 echo "</div>";

```

Slika 31: Izpisovanje potrebščin

Lasten arhiv, 2017

Začne se razred »dan«. Tukaj sem ponovno zapisala kodo za preizkus, če je torba pretežka. V tem primeru se v kodi izpiše »pretežko« (slika 32).

```
98     echo "<div class='dan";
99     if ($tezaTorbe[$trenutniDan-1] > $teza*0.1)
100         echo " pretežko";
101     echo ">";

```

Slika 32: Razred »dan«

Lasten arhiv, 2017

S switch-stavkom se glede na \$trenutniDan pod prvi naslov izpiše ime dneva (slika 33).

```
102     switch ($trenutniDan) {
103         case 2:
104             echo "<h1>Torek</h1>";
105             break;
106         case 3:
107             echo "<h1>Sreda</h1>";
108             break;
109         case 4:
110             echo "<h1>Četrtek</h1>";
111             break;
112         case 5:
113             echo "<h1>Petek</h1>";
114             break;
115         default:
116             break;
117     }

```

Slika 33: Switch-stavek za ime dneva

Lasten arhiv, 2017

Nato s switch-stavkom nastavljam dan, ki se tudi izpiše pod drugim naslovom. Prav tako se povsod pojavi gumb »Prikaži/skrij«, pod katerim so podrobnejši podatki za potrebščine za posamezni dan (slika 34).

```
118     echo "<h2>Skupna teža torbe: ".$tezaTorbe[$trenutniDan-1]." g</h2>";
119     switch ($trenutniDan) {
120         case 2:
121             echo "<button id='tor'>Prikaži/skrij</button>";
122             echo "<div id='potrebscineTor' style='display: none;'>";
123
124             break;
125         case 3:
126             echo "<button id='sre'>Prikaži/skrij</button>";
127             echo "<div id='potrebscineSre' style='display: none;'>";
128             break;
129         case 4:
130             echo "<button id='cet'>Prikaži/skrij</button>";
131             echo "<div id='potrebscineCet' style='display: none;'>";
132             break;
133         case 5:
134             echo "<button id='pet'>Prikaži/skrij</button>";
135             echo "<div id='potrebscinePet' style='display: none;'>";
136             break;
137         default:
138             break;
139     }
140
141
142 }
```

Slika 34: Switch-stavek za težo

Lasten arhiv, 2017

Pod razredom »potrebščina« se pod tretjim naslovom izpiše naziv predmeta, pod četrtem naslovom naziv potrebščine, pod petim teža potrebščine, pod i-oznako, ki besedilo naredi poševno, pa tip potrebščine (podatki so najdeni v bazi) (slika 35).

```
144         // po predmetih
145
146         {
147             echo "<div class='potrebscina'>";
148             echo "<h3>".$row["nazivPredmeta"]."</h3>";
149             echo "<h4>".$row["nazivPotrebscine"]."</h4>";
150             echo "<h5>Teža: ".$row["teza"]." g</h5>";
151             echo "<i>".$row["tip"]."</i>";
152             echo "</div>";
153         }
154     }
155
156
157
158
159 }
```

Slika 35: Razred »potrebščina«

Lasten arhiv, 2017

V kolikor program podatkov v bazi ne najde, se izpiše: »Ni rezultatov« (slika 36).

```
161     else
162     {
163         echo "Ni rezultatov.";
164     }
165
166 }
```

Slika 36: Ni rezultatov

Lasten arhiv, 2017

Nato se povezava zapre in program se zaključi (slika 37).

```
168 $conn->close();
169     echo "</div>";
170     echo "</div>";
171 ?>
172
173 </body>
174 </html>
```

Slika 37: Zaključek programa

Lasten arhiv, 2017

4.2 CSS-koda

V dokumentu style2.css sem z uporabo stilskega jezika CSS urejala izgled prve strani (index.php). Telesu sem določila barvo ozadja Light Skyblue, besedilo sem poravnala na sredino, okoli besedila pa sem naredila okvir. Ta okvir je širine 3 px, barve Mesium Slate Blue, polmer roba pa je 30 px (slika 38).

```
1  body {
2      background-color: lightskyblue;
3      text-align: center;
4      border: 3px solid mediumslateblue;
5      border-radius: 30px;
```

Slika 38: Izgled telesa prve strani

Lasten arhiv, 2017

Za prvi naslov sem določila barvo besedila Navy, sredinsko poravnavo in pisavo Lucida Sans Unicode, Lucida Grande (vrsta sans-serif, torej brez posebnega zaključka črk) (slika 39).

```
10 h1 {  
11   color: navy;  
12   text-align: center;  
13   font-family: "Lucida Sans Unicode", "Lucida Grande", sans-serif;  
14 }
```

Slika 39: Prvi naslov prve strani

Lasten arhiv, 2017

Za tretji naslov sem nastavila le sredinsko poravnavo (slika 40).

```
17 h3 {  
18   text-align: center;  
19 }
```

Slika 40: Tretji naslov prve strani

Lasten arhiv, 2017

Za celotno prvo stran sem nastavila pisavo Trebuchet MS, Helvetica (vrsta sans-serif) (slika 41).

```
21 * {  
22   font-family: "Trebuchet MS", Helvetica, sans-serif;  
23 }  
24 }
```

Slika 41: Pisava

Lasten arhiv, 2017

Za nogo (footer) strani sem nastavila velikost pisave 10 px, sredinsko poravnavo, absolutno pozicijo, položaj na dnu strani, na sredini, 25 % levo (slika 42).

```
26  .footer {
27      font-size: 10px;
28      text-align: center;
29      position: absolute;
30      bottom: 0%;
31      width: 50%;
32      left: 25%;
33  }
```

Slika 42: Izgled noge

Lasten arhiv, 2017

V dokumentu style.css sem urejala izgled druge strani (preveri.php). Za barvo ozadja telesa sem nastavila Light Sky Blue (slika 43).

```
1  body {
2      background-color: lightskyblue;
3
4  }
```

Slika 43: Izgled telesa druge strani

Lasten arhiv, 2017

Prvi naslov sem obarvala v Medium Blue (slika 44).

```
6  h1 {
7      color: mediumblue;
8  }
```

Slika 44: Prvi naslov druge strani

Lasten arhiv, 2017

Drugemu naslovu razreda dan sem dodala okvir, širine 5 px, rdeče barve (slika 45).

```
10  h2.dan {
11      border: 5px solid red;
12  }
```

Slika 45: Drugi naslov razreda dan

Lasten arhiv, 2017

Celotnemu razredu dan sem dodala okvir barve Medium Blue, širine 2 px, mejnega polmera 10 px, z marginom 10 px in paddingom 10 px (slika 46).

```
14  .dan {
15      border: 2px solid mediumblue;
16      border-radius: 10px;
17      margin: 10px;
18      padding: 10px;
19  }
```

Slika 46: Izgled razreda dan

Lasten arhiv, 2017

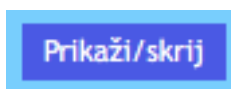
Gumb »pon« sem obarvala v belo, besedilo v črno, obrobo širine 2 px pa v Royal Blue. Ko se z miško postavimo na gumb (hover), se barva ozadja gumba spremeni v Royal Blue, barva besedila v belo, miška/kazalec v pointer, tranzicija/sprememba pa traja 0.3 sekunde. Enako se zgodi z gumbi za preostale dneve (slika 47).

```
21  #pon {
22      background-color: white;
23      color: black;
24      border: 2px solid royalblue;
25  }
26
27  #pon:hover {
28      background-color: royalblue;
29      color: white;
30      transition-duration: 0.3s;
31      cursor: pointer;
32  }
```

Slika 47: Izgled gumbov prikaži/skrij

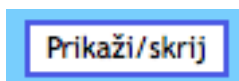
Lasten arhiv, 2017

Na sliki 48 je gumb, ko je obarvan, na sliki 49 pa gumb, ko je nepobarvan.



Slika 48: Obarvan gumb

Lasten arhiv, 2017



Slika 49: Nepobarvan gumb

Lasten arhiv, 2017

Primer dneva, ko torba ni pretežka lahko vidimo na sliki 50.



Slika 50: Torba ni pretežka

Lasten arhiv, 2017

Če je v kodi oz. razredu oznaka »pretežko«, se barva okvirja spremeni v rdečo, širine 6 px (slika 51). Drugi naslov se v tem primeru spremeni v rdečo (slika 52).

```
86  .pretežko {
87      border: 6px solid red;
88  }
```

Slika 51: Okvir »pretežko«

Lasten arhiv, 2017

```
90  .pretezko h2 {
91      color: red;
92  }
```

Slika 52: Drugi naslov »pretezko«

Lasten arhiv, 2017

Primer dneva, ko je torba pretežka, vidimo na sliki 53.



Slika 53: Torba je pretežka

Lasten arhiv, 2017

Celotno besedilo na strani je v pisavi Trebuchet MS, Helvetica (vrsta sans-serif) (slika 54).

```
94  * {
95      font-family: "Trebuchet MS", Helvetica, sans-serif;
96  }
97  }
```

Slika 54: Pisava druge strani

Lasten arhiv, 2017

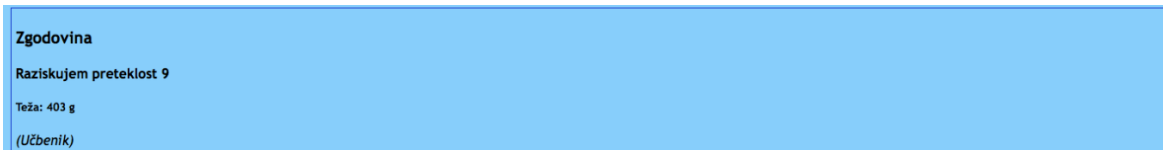
Razred »potrebščina« ima okvir širine 1 px, barve Medium Blue, z marginom 5 px in paddingom 5 px (slika 55).

```
108  .potrebščina {
109      border: 1px solid mediumblue;
110      margin: 5px;
111      padding: 5px;
112  }
```

Slika 55: Razred »potrebščina«

Lasten arhiv, 2017

Primer izpisane potrebščine je na sliki 56.



Slika 56: Izpisana potrebščina

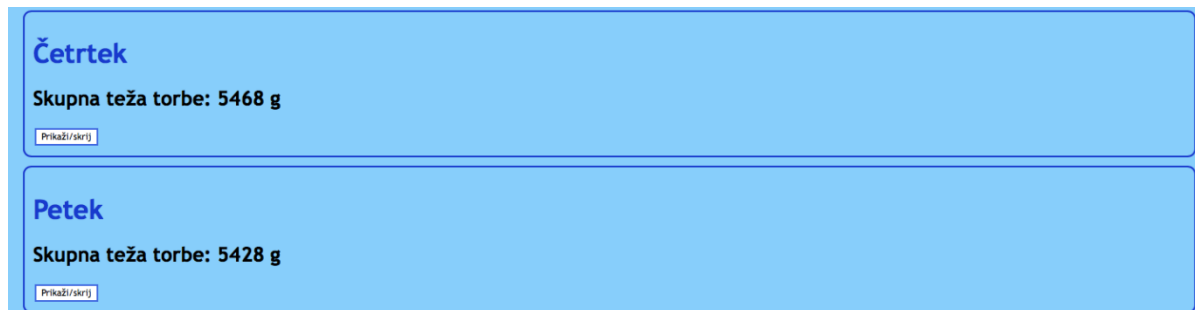
Lasten arhiv, 2017

Končni izgled druge strani prikazujeta sliki 57 in 58.



Slika 57: Končni izgled druge strani 1

Lasten arhiv, 2017



Slika 58: Končni izgled druge strani 2

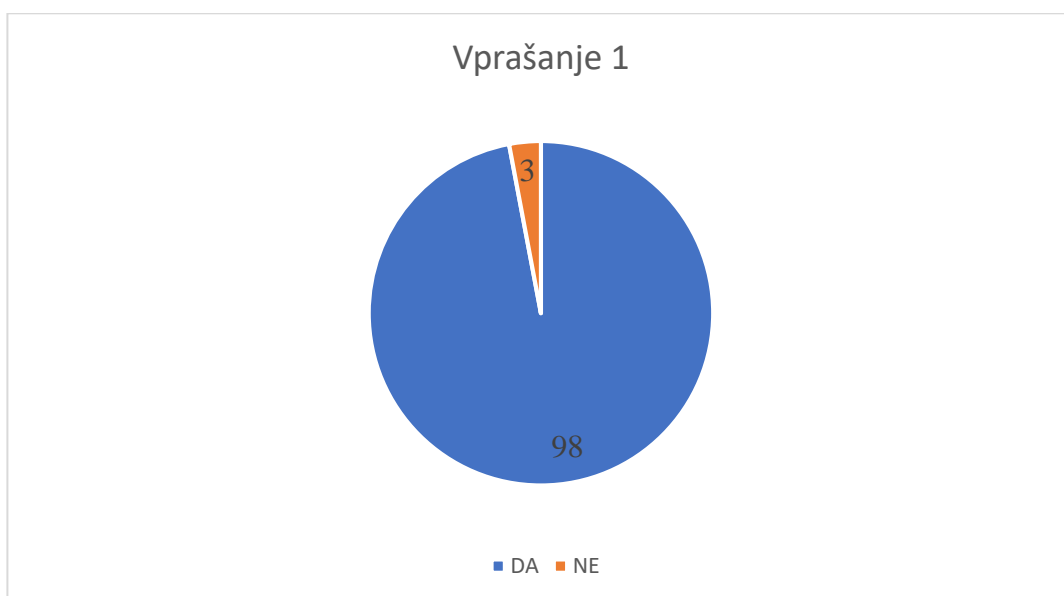
Lasten arhiv, 2017

5 RAZISKAVA

5.1 Anketa

Da bi pridobila povratne informacije, ki bi mi pomagale izboljšati aplikacijo in preveriti, ali sem dosegla svoje cilje, sem naredila spletno anketo, ki sem jo poslala vsem staršem naše šole. Prosila sem jih, da testirajo aplikacijo, nato pa odgovorijo na anketo. Dobila sem 101 odgovor.

Vprašanje 1: Ste aplikacijo znali uporabiti?

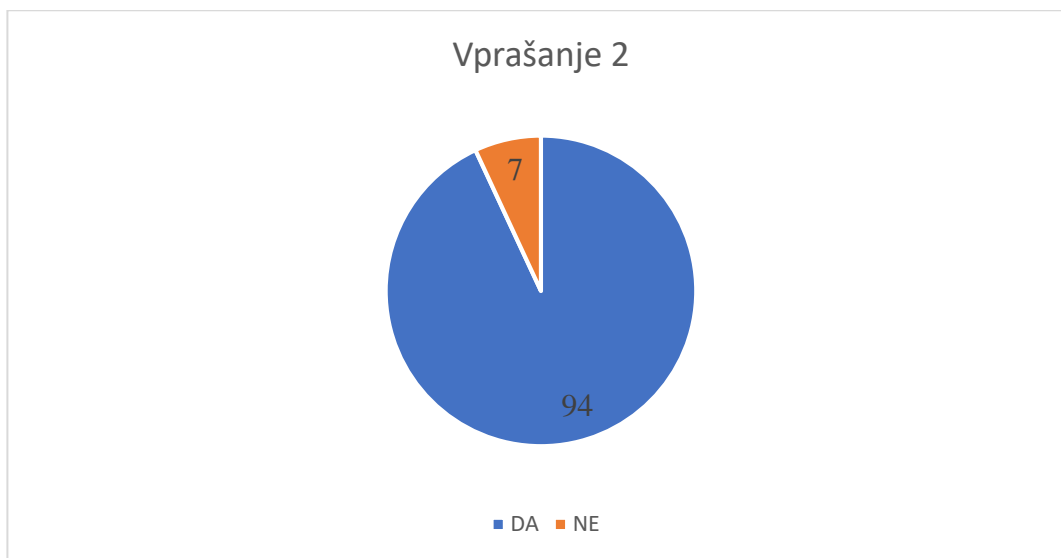


Graf 1: Uporaba aplikacije

Lasten arhiv, 2017

Na prvo vprašanje, če so aplikacijo znali uporabiti, je pritrdilno odgovorilo 98 staršev oz. 97 %. Le 3 starši aplikacije niso znali uporabiti (graf 1).

Vprašanje 2: Se vam zdi aplikacija uporabna?



Graf 2: Uporabnost aplikacije

Lasten arhiv, 2017

Pri drugem vprašanju je razvidno, da se kar 94 staršem oz. 93 % aplikacija zdi uporabna. Takšen rezultat mi je seveda všeč, saj sem uspela narediti aplikacijo, s katero je zadovoljna ciljna skupina – starši (graf 2).

Vprašanje 3: Imate kakšen predlog/izboljšavo za aplikacijo?

Predlogov nisem dobila veliko, tisti, ki pa so izstopali in so v prihodnosti tudi izvedljivi, pa so:

- zapisovanje podatkov na strežnik in izdelava statistike le-teh;
- možnost ročnega vnosa potrebščin;
- izdelava aplikacije še za srednje šole/več šol;
- pojasnitev gumba prikaži/skrij (že popravljeno/uporabljeno);
- pojasnitev teže prazne torbe z osnovnimi potrebščinami (že popravljeno/uporabljeno).

Nekatere predloge sem že uporabila, saj so bili dovolj preprosti, ostale, ki so bolj dolgoročni, pa imam namen vpeljati v prihodnosti.

Žal pa večina staršev, ki so menili, da aplikacija ni uporabna, ni podala nobenega predloga za izboljšavo, zato težko sklepam, kaj jih je zmotilo in, kako aplikacijo približati tudi njim.

5.2 Tehtanje torbe

Da bi videli, kako natančna je moja aplikacija, sem stehala svojo šolsko torbo s potrebščinami za posamezni dan in težo primerjala z izračunom teže v aplikaciji. V obeh sem upoštevala težo torbe z osnovnimi potrebščinami 3100 g. Dobila sem podatke, ki jih prikazuje tabela 1.

Tabela 1: Teža torbe

Dan	Dejanska teža (v g)	Teža v aplikaciji (v g)	Razlika (v g)
Ponedeljek	7220	7138	82
Torek	7100	7002	98
Sreda	5500	5566	66
Četrtek	6600	6568	32
Petek	6610	6528	82

V tabeli 1 je razvidno, da je razlika med težama manjša od 100 g. Teža potrebščin in torbe v aplikaciji je seveda bolj natančna, ker je bila uporabljena tudi natančnejša tehnika.

6 RAZPRAVA

Na začetku inovacijskega predloga sem si zadala nekaj ciljev, ki sem jih tudi uresničila. Za začetek mi je že uspelo sestaviti program, ki izračuna težo otrokove torbe, kar je bil moj primarni cilj. Spletno aplikacijo velika večina staršev zna uporabljati in se jim zdi tudi uporabna, kot prikazuje anketa. To me seveda veseli, saj je aplikacija namenjena prav njim. Prav tako je tudi natančna oz. izračunana teža je zelo podobna dejanski teži torbe, kot sem tudi sama preverila. S tem so izpolnjeni vsi moji cilji.

Aplikacija je uresničljiva v praksi, ker je preprosta za uporabo. Dostopna je na internetu in trenutno imajo možnost povezave vsi starši naše šole. Spletno povezavo bi lahko poslali še na druge šole. Je brezplačna, zato ne prinaša nobenih stroškov, torej ne staršem, ne šoli in ne meni. Aplikacija na zanimiv in drugačen način ozavešča starše in šolo o teži šolskih torb. Poleg tega pa staršem omogoča lažji vpogled v težo torbe in jih opozori, da še pravi čas ukrepajo v zvezi z otrokovim zdravjem.

7 DRUŽBENA ODGOVORNOST

Aplikacija za preverjanje teže šolske torbe je namenjena preverjanju teže šolskih torb, ki postajajo vse večji problem. Velja tudi kot opozorilo, da moramo v zvezi s šolskimi torbami ukrepati. Posledično izboljšamo zdravje otrok oz. preprečimo in zmanjšamo dolgotrajne poškodbe. Aplikacija je namenjena in dostopna vsem staršem in otrokom naše šole. Imam namen, da bi jo razširila na več šol in tako bi bila dostopna čim večji množici ljudi. Pri izboljšavi aplikacije sem in bom upoštevala predloge mentorjev, učiteljev ter seveda uporabnikov, saj se zavedam, da je namenjena prav njim.

Tako moj inovacijski predlog prispeva in se navezuje na osnovna načela družbene odgovornosti po ISO 26000.

8 SKLEP

V inovacijskem predlogu sem ugotovila, da je mogoče narediti spletno aplikacijo, ki izračuna težo šolske torbe otroka, zato sem jo tudi naredila. Naučila sem se osnov nekaterih jezikov, kot so PHP, JS, CSS, HTML, MySQL in jQuery. Kodo aplikacije sem podrobno opisala tudi v predlogu.

Spletna aplikacija za preverjanje teže šolske torbe je zasnovana tako, da jo lahko starši enostavno uporabijo. Z raziskavo sem ugotovila, da sem aplikacijo uspela približati ciljni skupini – staršem.

V prihodnosti imam namen aplikacijo še izboljšati in upoštevati predloge, ki sem jih dobila:

- shranjevanje podatkov o teži torb na strežnik;
- nastavitve izbire potrebščin oz. predmetov;
- omogočiti registracijo uporabnikov.

Glavni cilj je aplikacijo razširiti na več šol, ki bi bile pripravljene sodelovati, saj bo le tako problem pretežkih šolskih torb postal bolj znan in opozoril, da moramo v zvezi z otrokovim zdravjem nujno ukrepati. Aplikacija namreč na zanimiv in inovativen način opozarja na dotičen problem, ki je bil moj primarni cilj.

9 VIRI

- [1] Computer Programming,“ [Elektronski]. Dostopno na: <http://homepage.cs.uri.edu/faculty/wolfe/book/Readings/Reading13.htm>. [Poskus dostopa 21. januar 2018].
- [2] S. Bebbington, „Learn Programming,“ [Elektronski]. Dostopno na: <http://yearofcodes.tumblr.com/what-is-coding>. [Poskus dostopa 21. januar 2018].
- [3] Mozilla, „About JavaScript,“ 13. junij 2017. [Elektronski]. Dostopno na: https://developer.mozilla.org/en-US/docs/Web/JavaScript/About_JavaScript. [Poskus dostopa 2. februar 2018].
- [4] jquery-tutorial.net, „What is jQuery?,“ [Elektronski]. Dostopno na: <http://www.jquery-tutorial.net/introduction/what-is-jquery/>. [Poskus dostopa 21. januar 2018].
- [5] Tutorial Republic, „PHP Introduction,“ [Elektronski]. Dostopno na: <https://www.tutorialrepublic.com/php-tutorial/php-introduction.php>. [Poskus dostopa 21. januar 2018].
- [6] R. Shannon, „What is HTML?,“ 21. avgust 2012. [Elektronski]. Dostopno na: <http://www.yourhtmlsource.com/starthere/whatishtml.html>. [Poskus dostopa 21. januar 2018].
- [7] Tutorials Point, „What is CSS?,“ [Elektronski]. Dostopno na: https://www.tutorialspoint.com/css/what_is_css.htm. [Poskus dostopa 30. januar 2018].
- [8] „CSS Tutorial,“ [Elektronski]. Dostopno na <http://html.net/tutorials/css/lesson1.php>. [Poskus dostopa 30. januar 2018].
- [9] A. T. Košir, „Slovenski šolarji nosijo pretežke šolske torbe. Tako težke, da so zdravju nevarne,“ 7. marec 2016. [Elektronski]. Dostopno na: <https://siol.net/trendi/odnosi/slovenski-solarji-nosijo-pretezke-solske-torbe-tako-tezke-da-so-zdravju-nevarne-video-407209>. [Poskus dostopa 30. januar 2018].

PRILOGA

APLIKACIJA ZA PREVERJANJE TEŽE ŠOLSKE TORBE – VPRAŠALNIK

Letos sem se odločila, da napišem inovacijski predlog, zato sem naredila spletno aplikacijo za preverjanje teže šolske torbe, ki ste jo prej poskusno uporabili. Prosim vas, da odgovorite na naslednja vprašanja, saj bom le tako dobila povratne informacije, ki mi bodo pomagale pri izboljšavi aplikacije. Vaši odgovori bodo anonimni.

Hvala za pomoč in sodelovanje.

1. Ste aplikacijo znali uporabiti?

DA

NE

2. Se vam zdi aplikacija uporabna?

DA

NE

3. Imate kakšen predlog/izboljšavo za aplikacijo?
