

# Pametna vaga i visinomjer

---

Vukonić, Luka

Master's thesis / Diplomski rad

2024

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Rijeka, Faculty of Engineering / Sveučilište u Rijeci, Tehnički fakultet**

Permanent link / Trajna poveznica: <https://um.nsk.hr/um:nbn:hr:190:453767>

Rights / Prava: [Attribution 4.0 International](#)/[Imenovanje 4.0 međunarodna](#)

Download date / Datum preuzimanja: **2025-02-21**



Repository / Repozitorij:

[Repository of the University of Rijeka, Faculty of Engineering](#)



SVEUČILIŠTE U RIJECI  
**TEHNIČKI FAKULTET**  
Diplomski studij računarstva

Diplomski rad

## **Pametna vaga i visinomjer**

Rijeka, rujan 2024.

Luka Vukonić  
0069078050

SVEUČILIŠTE U RIJECI  
**TEHNIČKI FAKULTET**  
Diplomski studij računarstva

Diplomski rad

## **Pametna vaga i visinomjer**

Mentor: prof. dr. sc. Mladen Tomić

Rijeka, rujan 2024.

Luka Vukonić  
0069078050

Umjesto ove stranice umetnuti zadatak  
za završni ili diplomski rad

## Izjava o samostalnoj izradi rada

Izjavljujem da sam samostalno izradio ovaj rad.

Rijeka, rujan 2024.

-----  
Ime Prezime

# Zahvala

Zahvaljujem svojoj obitelji na podršci tijekom studiranja i pisanja ovoga rada, a posebno ocu Ronaldu na stručnoj pomoći i potrebnim alatima. Zahvaljujem svojoj curi, prijateljima i kolegama na pomoći u pisanju i hrabrosti za obavljena mjerenja. Hvala prof. dr. sc. Mladenu Tomiću na mentoriranju i dvogodišnjem strpljenju za izradu rada.

# Sadržaj

<b>Popis slika</b>	<b>viii</b>
<b>Popis tablica</b>	<b>x</b>
<b>Popis isječaka programskog koda</b>	<b>xi</b>
<b>1 Uvod</b>	<b>1</b>
<b>2 Sklop i programska podrška</b>	<b>3</b>
2.1 Opis sklopa . . . . .	3
2.1.1 Mjerna ćelija . . . . .	4
2.1.2 Kombinator opterećenja . . . . .	7
2.1.3 HX711 modul . . . . .	8
2.1.4 HC-SR04 senzor udaljenosti . . . . .	8
2.1.5 Raspberry Pi 3 B+ . . . . .	10
2.1.6 Razvodna ploča . . . . .	11
2.1.7 Ekran osjetljiv na dodir . . . . .	11
2.2 Opis programske podrške . . . . .	13
2.2.1 Raspberry Pi OS . . . . .	13
2.2.2 Django radni okvir i web aplikacija . . . . .	13
2.2.3 Git sustav za verzioniranje . . . . .	15
<b>3 Konstrukcija pametne vage</b>	<b>16</b>
3.1 Izrada konstrukcije pametne vage . . . . .	16
3.2 Izrada programske podrške . . . . .	19

## Sadržaj

<b>4</b>	<b>Mjerenja i rezultati</b>	<b>26</b>
4.1	Mjerenja . . . . .	26
4.1.1	Mjerenje mase . . . . .	27
4.1.2	Mjerenje visine . . . . .	27
4.2	Rezultati . . . . .	28
4.2.1	Rezultati mjerenja mase . . . . .	28
4.2.2	Rezultati mjerenja visine . . . . .	31
<b>5</b>	<b>Diskusija</b>	<b>36</b>
5.1	Iznos greške pri mjerenju visine . . . . .	36
5.2	Položaj mjernih ćelija . . . . .	37
5.3	Šum u sustavu mjerenja mase . . . . .	38
5.4	Raspberry Pi 3B+ kao platforma . . . . .	38
<b>6</b>	<b>Zaključak</b>	<b>40</b>
	<b>Literatura</b>	<b>42</b>
	<b>Pojmovnik</b>	<b>45</b>
	<b>Sažetak</b>	<b>46</b>



## Popis slika

2.1	Mjerna ćelija . . . . .	4
2.2	Princip rada tenzometarske trake [5] . . . . .	5
2.3	Wheatstoneov most [6] . . . . .	6
2.4	Prikaz spajanja mjernih ćelija i pločice kombinatora [8] . . . . .	7
2.5	Spoj HX711 modula i pločice kombinatora . . . . .	8
2.6	HC-SR04 modul [11] . . . . .	9
2.7	Vremenski dijagram HC-SR04 modula . . . . .	10
2.8	Pojednostavljena shema spajanja senzora i uređaja Raspberry Pi . .	11
2.9	Razvodna ploča u kućištu sa spojenim sensorima, GPIO pinovima te dodatne pločice za napajanje . . . . .	12
2.10	Ekran osjetljiv na dodir u 3D ispisanom kućištu . . . . .	13
2.11	Prikaz ekrana u načinu rada s mogućnošću pohranjivanja izmjerenih vrijednosti . . . . .	14
2.12	Primjer grafa za praćenje izmjerenih vrijednosti unutar stranice profila korisnika . . . . .	15
3.1	Konstrukcija pametne vage . . . . .	17
3.2	Shema razvodne ploče korištena za jetkanje . . . . .	18
3.3	Zasebna pločica za napajanje . . . . .	18
3.4	Početni ekran nakon paljenja uređaja . . . . .	25
3.5	Prikaz aktivirane virtualne tipkovnice pri unosu u formu . . . . .	25
4.1	Bland-Altman Graf. Os x prikazuje prosječnu vrijednost težine u kilogramima ručnog i mjerenja pomoću pametne vage. Os y prikazuje razliku između ručnog mjerenja i mjerenja pomoću pametne vage . .	31

*Popis slika*

4.2	Graf odstupanja. Na X osi nalaze se vrijednosti razlike mjerenja pametne vage u odnosu na ručno mjerenje. Na Y osi nalazi se ručno izmjerena masa poredana uzlazno . . . . .	33
4.3	Bland-Altman Graf. Os x prikazuje prosječnu vrijednost visine u centimetrima ručnog i mjerenja pomoću pametne vage. Os y prikazuje razliku između ručnog mjerenja i mjerenja pomoću pametne vage . .	34
4.4	Graf odstupanja. Na X osi nalaze se vrijednosti razlike mjerenja pametne vage u odnosu na ručno mjerenje. Na Y osi nalazi se ručno izmjerena visina poredana uzlazno . . . . .	35
5.1	HC-SR04 modul i pripadajući nosač . . . . .	37
5.2	Prikaz ispisa aplikacije htop . . . . .	39

## Popis tablica

4.1	Izmjerene vrijednosti visine. . . . .	29
4.2	Izmjerene vrijednosti mase. . . . .	30
4.3	Ukupan broj mjerenja te vrijednosti srednje pogreške, maksimalne pogreške i standardne devijacije pogreške mase . . . . .	31
4.4	Ukupan broj mjerenja te vrijednosti srednje pogreške, maksimalne pogreške i standardne devijacije pogreške visine . . . . .	32
5.1	Specifikacije Raspberry Pi modela 3B+ . . . . .	38

## Popis isječaka programskog koda

3.1	Mjerenje udaljenosti . . . . .	19
3.2	Mjerenje mase . . . . .	20
3.3	Servis za pokretanje mjerenja mase . . . . .	21
3.4	Učitavanje mjerenja mase iz datoteke i priprema podataka za prikaz .	22
3.5	Dinamičko ažuriranje vrijednosti mjerenja . . . . .	23
3.6	Kiosk skripta . . . . .	24

# Poglavlje 1

## Uvod

Tijekom školovanja i radnog vijeka inženjeri veliki dio svojeg vremena provode za radnim stolom ponajviše radi posla, ali djelomično i u slobodno vrijeme. Sedentarni način života utječe na metabolizam, fizičku funkciju i na zdravlje općenito [1]. Ako prehrana ne prati sjedilački način života, postoji rizik od razvoja prekomjerne tjelesne težine. Prema Svjetskoj zdravstvenoj organizaciji pretilost i prekomjerna težina su definirani kao nenormalno ili prekomjerno nakupljanje masti koje može biti štetno za zdravlje [2]. Svjetska zdravstvena organizacija pretilost smatra globalnom epidemijom. Godine 2022. svaka osma osoba je pretila, dok 43% odraslih osoba ima prekomjernu tjelesnu težinu. Organizacija također smatra da su pretilost i prekomjerna tjelesna težina u velikom broju slučajeva sprječive, odnosno upravljive.

Jedan od načina sprječavanja prekomjerne tjelesne težine je i pravilno praćenje vlastite tjelesne težine. Kako bi bilo lakše odrediti ima li osoba prekomjernu tjelesnu težinu koristi se indeks tjelesne mase (BMI). Radi se o jednostavnom izračunu koji je prikazan u Formuli 1.1.

$$\text{BMI} = \frac{\text{masa (kg)}}{\text{visina (m)}^2} \quad (1.1)$$

Kako bi bilo moguće izračunati index tjelesne mase potrebno je znati masu i visinu. Učestalo korištene konvencionalne mehaničke vage koje mjere masu i visinu često zahtijevaju ručno podešavanje te ih nije lagano samostalno koristiti. Za mjerenje mase potrebno je pomicati protuuteg kako bi se vaga dovela u ravnotežu. Za mjerenje visine potrebno je namjestiti nastavak za glavu na vrh tjemena.

Pametna vaga zamišljena je da ubrza proces mjerenja te ponudi mogućnost digi-

## *Poglavlje 1. Uvod*

talnog zapisa za praćenje rezultata kroz duži period vremena. Vaga je zamišljena i konstruirana kao ugradbeni sustav. Ugradbeni sustav je računalni sustav s namjenskim funkcijama unutar većeg električnog ili mehaničkog sustava[3]. Raspberry Pi 3 B+ je računalo veličine kreditne kartice sa svim potrebnim komponentama na jednoj pločici te se koristi kao središnji element ugradbenog sustava. Za mjerenje visine korišten je modul HC-SR04 koji putem odašiljanja i primanja ultrazvučnih signala mjeri udaljenost. Za mjerenje mase korištene su mjerne ćelije s HX711 modulom koje rade na principu Wheatstoneovog mosta. HX711 pretvara analogni signal u digitalni te uz to radi i kao pojačalo. Kako bi hardverski sklop bio modularan bilo je potrebno izraditi razvodnu ploču te dodatnu pločicu za napajanje s pripadajućim ožičenjem. Za razvoj web aplikacije korišten je Django programski okvir prvenstveno radi utilizacije Python programskog jezika za očitavanje mjerenih podataka na poslužiteljskoj strani te HTML, CSS, JS i AJAX.

## Poglavlje 2

### Sklop i programska podrška

Središnji element pametne vage je Raspberry Pi 3B+. On je zadužen za prikupljanje podataka od senzora, obradu te prikazivanje istih u web aplikaciji. Sklop koristi četiri identične mjerne ćelije za mjerenje mase spojene preko pločice za kombiniranje opterećenja na HX711 modul koji pretvara analogni signal u digitalni te ga pojačava. Uz to koristi se i jedan HC-SR04 ultrazvučni senzor za mjerenje visine. Raspberry Pi s operacijskim sustavom Raspbian poslužuje Django web aplikaciju koja se prikazuje korisniku pomoću Chromium internetskog preglednika na Joy-IT RD-LCD7-2 7 inčnom ekranu osjetljivom na dodir.

#### 2.1 Opis sklopa

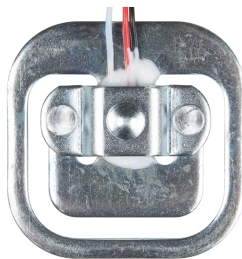
Pametna vaga sastoji se od aluminijskog postolja i vertikalne aluminijske letve s 3D ispisanim kućištem. Postolje je izrađeno u L profilu kvadratnog oblika s predviđenim mjestom za pričvršćivanje vertikalne letve sa stražnje strane s tri vijka. Na gornjoj plohi profila postolja nalaze se četiri mjerne ćelije kvadratnog oblika dimenzija 34 mm x 34 mm. Ispod svakog senzora nalaze se rupe promjera 27 milimetara koje omogućuju središnjem elementu senzora prostor za savijanje istog prema dolje. Mogućnost savijanja je bitna za ispravan rad senzora. Sva četiri senzora spojena su na kombinator opterećenja koji je spojen na HX711 pojačalo. Kako bi opterećenje težine bilo jednoliko na sva četiri senzora koristi se ravna metalna ploča kvadratnog oblika s bridom duljine 450 mm te debljine 6 mm. Ploča ima masu od 11 kilograma te služi i kao uteg kojim se spušta centar ravnoteže pametne vage i time dobiva na

## Poglavlje 2. Sklop i programska podrška

stabilnosti. Šuplja vertikalna letva visine 240 cm, širine 6 cm te debljine 2 cm služi za namještanje HC-SR04 senzora udaljenosti na željenu visinu pomoću 3D ispisanog pomičnog elementa. Na letvi se nalazi pomični držač za kućište pametne vage. 3D ispisanu kućište sadrži Raspberry Pi 3B+, ručno napravljenu razvodnu ploču i ekran osjetljiv na dodir.

### 2.1.1 Mjerna ćelija

Pametna vaga koristi četiri mjerne ćelije kvadratnog rasporeda. Svaka ćelija ima maksimalno opterećenje od 50 kg što znači da je maksimalna nosivost vage 200 kg. Korištene su mjerne ćelije s tenzometarskim trakama. Tenzometrija je eksperimentalna metoda mehanike kojom se na konstrukciji ili modelu određuje duljinska deformacija da bi se odredila naprezanja na površini konstrukcije[4]. Tenzometrijska traka u mirovanju na konstantnoj temperaturi ima stalan otpor. Korištene mjerne ćelije koriste tenzometrijske trake otpora 1 k $\Omega$ . Prije korištenja mjernim je instrumentom provjeren otpor mjernih ćelija kako bi se utvrdilo da su ispravne. Slika 2.1 prikazuje



Slika 2.1 Mjerna ćelija

dva elementa: vanjski prsten koji treba biti punom površinom na ravnoj plohi, te unutarnji element koji mora imati mogućnost da se savije prema dolje. Na spoju između vanjskog i unutarnjeg dijela nalaze se tenzometarske trake što znači da se savijanjem unutarnjeg prstena savija i traka. Tenzometarskoj traci se deformacijom odnosno promjenom dimenzija mijenja i otpor na način koji je opisan na Slici 2.2. Iz jednadžbe 2.1 razvidno je kako se smanjivanjem poprečnog presjeka uz povećanje duljine vodiča električni otpor povećava te da se povećanjem poprečnog presjeka uz



## Poglavlje 2. Sklop i programska podrška

smanjivanje duljine vodiča električni otpor smanjuje.

$$R = \rho \frac{l}{A} \quad (2.1)$$

Gdje su:

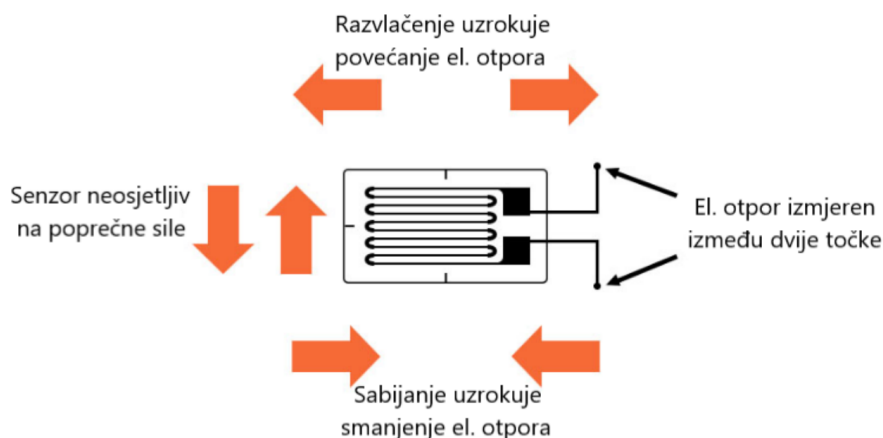
$R$  – električni otpor vodiča [ $\Omega$ ]

$\rho$  – električna otpornost materijala vodiča [ $\Omega\text{m}$ ]

$l$  – duljina vodiča [m]

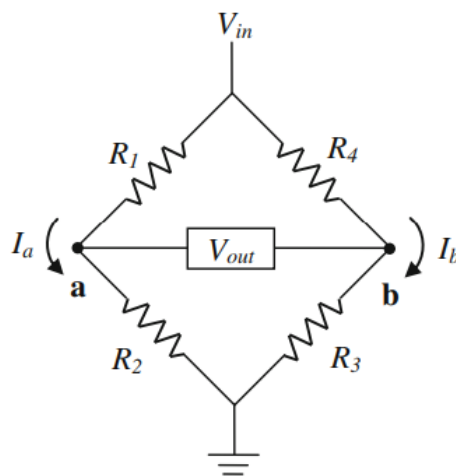
$A$  – površina poprečnog presjeka vodiča [ $\text{m}^2$  ]

Radi veće preciznosti i smanjenja vanjskih utjecaja na tenzometarsku traku, često se koristi konfiguracija Wheatstoneovog mosta. Wheatstoneov most koristi se za mjerenje promjene električnog otpora ili za mjerenje apsolutnog iznosa otpora. Radi se o električnom krugu koji se sastoji od četiri otpornika u dijamantnoj konfiguraciji. Otpornici su spojeni tako da dva para serijski spojenih otpornika čine paralelni spoj. Djelovanjem napona između  $V_{in}$  iz uzemljenja kao prema Slici 2.3, most je u ravnoteži ( $V_{out}=0$ ) ako su svi omjeri otpora u granama jednaki ili ako su svi otpori jednaki. Ako se otpor u nekoj grani promjeni, most više nije u ravnoteži te slijedi da je  $V_{out} \neq 0$ .



Slika 2.2 Princip rada tenzometarske trake [5]

## Poglavlje 2. Sklop i programska podrška



Slika 2.3 Wheatstoneov most [6]

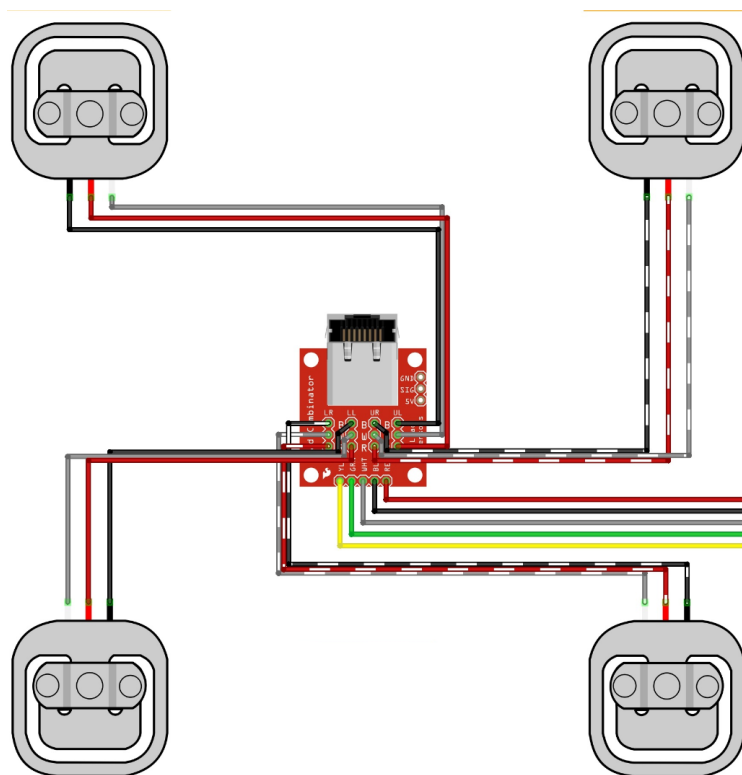
$V_{out}$  moguće je izračunati pomoću Formule 2.2

$$V_{out} = \left( \frac{R_3}{R_3 + R_4} + \frac{R_2}{R_1 + R_2} \right) V_{in} \quad (2.2)$$

Postoji više tipova Wheatstoneovog mosta: puni, polovični i četvrtinski tip. Tipovi se razlikuju po broju korištenih promjenjivih otpornika - u ovom slučaju tenzometrijskih traka. Puni Wheatstoneov most koristi četiri tenzometrijske trake. Prednosti ove konfiguracije bolja su kompenzacija temperaturnih promjena i povećana osjetljivost u odnosu na polu-most te četvrtinski most [7]. Za razliku od polu-mosta i četvrtinskog mosta koji aproksimiraju vezu između primijenjene sile i izlaznog napona, puni most daje linearni izlaz signala koji je izravno proporcionalan primijenjenoj sili [5]. Polu-most također koristi četiri otpornika, no u ovoj konfiguraciji dva su otpornika konstantna dok su ostala dva otpornika zapravo tenzometrijske trake. Četvrtinski most sastoji se od tri konstantna otpornika te jedne tenzometrijske trake. Mjerne ćelije pametne vage koriste polu-most. Postoje dvije konfiguracije polu-mosta. U jednoj varijaciji tenzometrijske trake nalaze se na poziciji otpornika  $R_1$  i  $R_2$  sa Slike 2.3, dok se u drugoj varijaciji zvanoj i dijagonalni most trake nalaze na pozicijama otpornika  $R_1$  i  $R_3$ . U prvom slučaju trake su izložene jednakom naprezanju po iznosu i predznaku dok su u drugom slučaju također izložene jednakom naprezanju, ali suprotnog predznaka [7].

### 2.1.2 Kombinator opterećenja

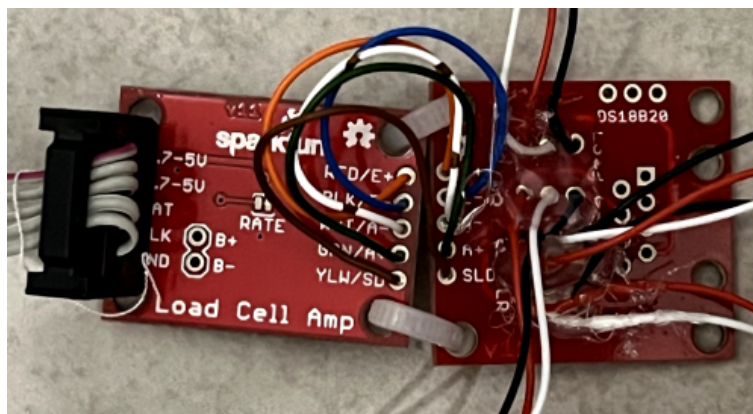
Pametna vaga koristi četiri mjerne ćelije koje je moguće međusobno ručno spojiti, no radi jednostavnosti i robusnosti korišten je kombinator opterećenja. Kombinator opterećenja je pločica koja pojednostavljuje spajanje više mjernih ćelija na jedno pojačalo. Izlaz pločice kombinira 12 ulaza (četiri mjerne ćelije po tri žice) u standardna četiri izlaza Wheatstoneovog mosta. Iako kombinator ima pripremu i za RJ45 konektor za primjene gdje su mjerne ćelije na različitoj lokaciji od ostatka sustava, on u ovom slučaju nije korišten. Prikaz spajanja mjernih ćelija prikazan je na Slici 2.4. Mjerne ćelije koriste izrazito tanke žice koje su se tijekom istraživanja i slaganja pametne vage više puta pucale i odvajale od pločice. U finalnom izdanju spoj mjernih ćelija i pločice dodatno je zaštićeno ljepljivom na bazi kopolimera etilen-vinil acetata što je prikazano na Slici 2.5.



Slika 2.4 Prikaz spajanja mjernih ćelija i pločice kombinatora [8]

### 2.1.3 HX711 modul

*General-purpose input/output (GPIO)* pinovi na Raspberry Piju digitalne su prirode te ne bi mogli smisljeno očitati signal koji dolazi s pločice kombinatora. Kako bi Raspberry Pi dobio korisne podatke, potrebno je u sustav dodati analogno - digitalni pretvarač. Uz analogno digitalnu pretvorbu potrebno je naglasiti da Wheatstoneov most na izlazu daje očitavanja napona reda veličine milivolti [9]. Osjetljivost na izlazu mjerne ćelije je otprilike 1 mv/v što znači da jedna ćelija pri maksimalnom opterećenju pri ulaznom naponu od 5 V može proizvesti diferencijalni napon od 5 mV . Kako bi pretvorba signala bila sa što većom preciznošću i rezolucijom. HX711 modul 24 je bitni analogno - digitalni pretvarač s opcijom pojačanja signala 32 fiksno na B kanalu te 64 ili 128 na A kanalu. Pojačanju od 64 puta pripada diferencijalni napon od  $\pm 40$  mV, a pojačanju od 128 puta pripada diferencijalni napon od  $\pm 20$  mV[10]. Radi spomenutih malih vrijednosti dobivenih s mosta, korišten je A kanal s pojačanjem od 128. Na Slici 2.5 prikazan je spoj pločice kombinatora te HX711 modula.



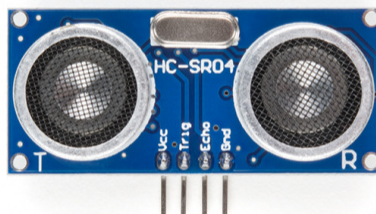
Slika 2.5 Spoj HX711 modula i pločice kombinatora

### 2.1.4 HC-SR04 senzor udaljenosti

Za beskontaktno mjerenje visine pametna vaga koristi jedan HC-SR04 senzor udaljenosti. Radi se o jeftinom i lako dostupnom senzoru s velikom bazom korisnika. HC-SR04 koristi ultrazvučnu tehnologiju za mjerenje udaljenosti. Na Slici 2.6 prikazana su dva elementa označena na pločici s T i R, radi se o odašiljaču i prijemniku.

## Poglavlje 2. Sklop i programska podrška

Udaljenost se mjeri pomoću odašiljanja i primanja 40 kHz valova. Na Slici 2.7 [12]



Slika 2.6 HC-SR04 modul [11]

je prikazano da proces mjerenja počinje impulsom trajanja  $10 \mu\text{s}$  na ulazu senzora (Trigger). Potom odašiljač pošalje osam 40 kHz valova te odmah po završetku odašiljanja Echo pin postavlja se u visoko stanje. Echo pin ostat će u visokom stanju sve do trenutka kada prijemnik ne detektira reflektirani val, tada on prelazi u nisko stanje[12]. Pomoću izmjenjenog vremena koje je Echo pin proveo u visokom stanju moguće je izračunati udaljenost pomoću Formule 2.3

$$d = v \cdot \frac{t}{2}, \quad (2.3)$$

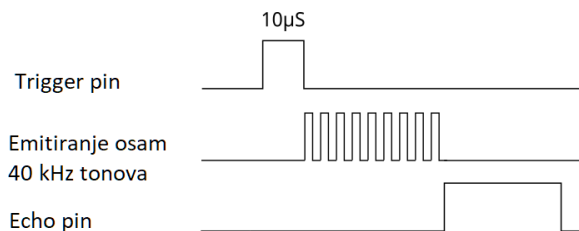
Gdje su:

$v$  – brzina zvuka u trenutnim atmosferskim uvjetima [ $\text{m}/\text{s}^2$ ]

$t$  – vrijeme između odašiljanja i primanja signala [s ]

Prema tehničkoj specifikaciji uređaja Raspberry Pi maksimalan napon na GPIO ulazima može biti 3.3 V [13]. Echo pin na HC-SR04 modulu ima napon jednak ulaznom naponu, u ovom slučaju 5 V te je potrebno smanjiti napon na pinu kako ne bi došlo do oštećenja. Za smanjenje napona korištena je BZX85C3v0 zener dioda spojena na ground pin kako bi smanjila napon pina na 3 V. Iako je račun za udaljenost jednostavan, pouzdanost može biti smanjena zbog niza različitih faktora. Bitno je da je kut između senzora i objekta čiju udaljenost mjerimo između  $45^\circ$  i  $90^\circ$  kako bi

## Poglavlje 2. Sklop i programska podrška



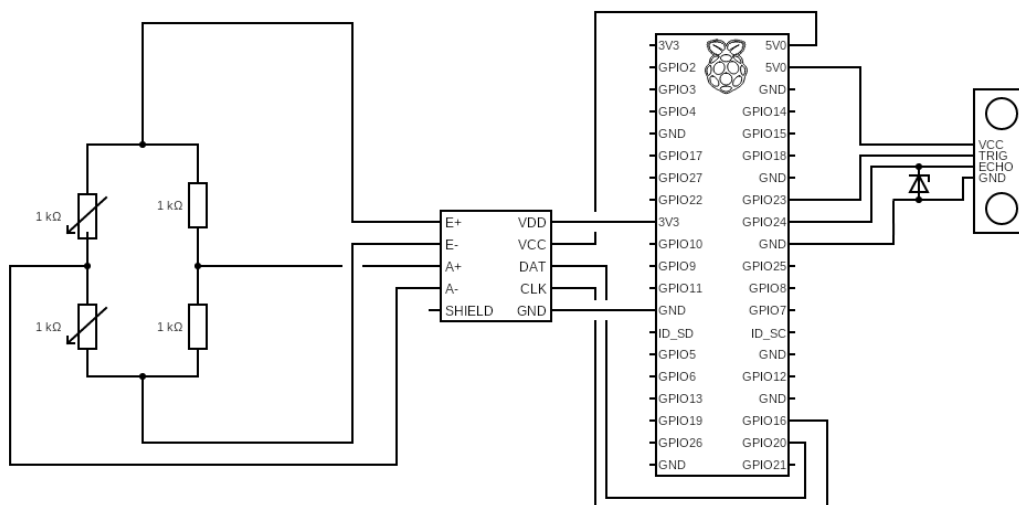
Slika 2.7 Vremenski dijagram HC-SR04 modula

se odbijeni ultrazvučni val mogao očitati na prijemu [14]. Gustoća kose ili veća frizura također može biti faktor koji će rezultirati smanjenju preciznosti rezultata.

### 2.1.5 Raspberry Pi 3 B+

Raspberry Pi 3B+ je *System-on-chip (SoC)* minijaturno računalo s dimenzijama 86 x 54 mm koje na sebi sadrži sve komponente potrebne za rad. Najvažniji dio uređaja je 64-bitni Broadcom BC2837 ARM-8 četverojezgreni procesor s integriranom Cortex - A53 grafikom. Model 3B+ posjeduje 1 GB radne memorije i koristi MicroSD karticu za spremanje podataka [15]. Uređaj se napaja putem micro USB konektora s 5 V i 2.5 A, dok mu je potrošnja između 2.20 W i 4 W [16] što ga čini idealnim izborom za projekte gdje se zahtjeva kontinuirani rad[15]. Pametna vaga koristi Raspberry Pi zbog ulaznih/izlaznih pinova (GPIO). "General purpose input/output" odnosno GPIO je tip sučelja koji se koristi za komunikaciju između perifernih uređaja i mikroprocesora[17]. Raspberry Pi model 3B + ima 40 GPIO pinova s dva reda po dvadeset pinova. Svaki se pin može programirati kao ulazni ili kao izlazni pin te također može biti korišten za dodatne funkcije [18](npr. I2C, UART i SPI sučelje). Na Slici 2.8 simbolično je prikazana shema spajanja. Pametna vaga koristi pinove GPIO20 i GPIO16 za očitavanje podataka s HX711 modula te GPIO23 i GPIO24 za očitavanje podataka s HS-SR04 modula. Za napajanje HX711 modula koristi se 3.3 V pin, dok se za napajanje HC-SR04 modula koristi 5 V pin. Oba modula koriste i pinove uzemljenja za napajanje senzora.

## Poglavlje 2. Sklop i programska podrška



Slika 2.8 Pojednostavljena shema spajanja senzora i uređaja Raspberry Pi

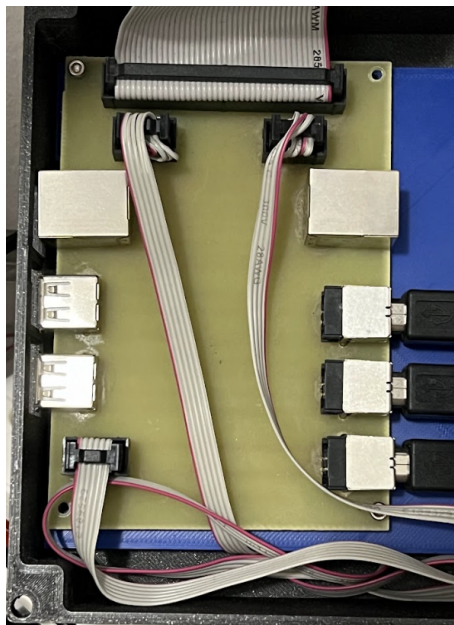
### 2.1.6 Razvodna ploča

Razvodna ploča izrađena je radi modularnosti, olakšanja povezivosti pametne vage i optimizacije kabliranja. Kao što je prikazano na Slici 2.9 razvodna ploča smještena je uz lijevi rub kućišta kako bi USB i internet sučelja bila lako dostupna korisniku. Osim navedenih sučelja na ploči je dostupno i svih 40 GPIO pinova uređaja Raspberry Pi, no koristi se samo 9 pinova za mjerne ćelije te ultrazvučni senzor udaljenosti. Spomenuti senzori spajaju se na dva odvojena konektora. Ploča sadrži i konektor na koji se spaja ručno izrađena pločica s Micro USB konektorom za napajanje u podnožju vage sa stražnje strane. Spomenuti konektor do USB-A konektora koji se koristi za spajanje s uređajem Raspberry Pi spajaju samo dva vodiča koja se koriste za napajanje. Za navedeno sučelje nisu potrebni vodiči za prijenos podataka kao za ostala dva USB sučelja na razvodnoj ploči.

### 2.1.7 Ekran osjetljiv na dodir

Radi lakšeg korištenja i boljeg korisničkog sučelja pametna vaga koristi Joy-IT RD-LCD7-2 ekran osjetljiv na dodir. Kao što je prikazano na Slici 2.10, Ekran je integriran u kućište pametne vage. Radi se o 7 inčnom LCD ekranu maksimalne rezolucije

## Poglavlje 2. Sklop i programska podrška



Slika 2.9 Razvodna ploča u kućištu sa spojenim sensorima, GPIO pinovima te dodatne pločice za napajanje

1024x600 s IPS panelom. Prema Krantz et.al svjetljivost od minimalno  $180 \text{ cd/m}^2$  prihvatljiva je za upotrebu u unutarnjem prostoru pri svim uvjetima osvjetljenja[19]. Korišteni ekran ima svjetljivost od  $250 \text{ cd/m}^2$  te je stoga prikladan za korištenje u unutarnjem prostoru. Ekran je s uređajem Raspberry Pi spojen HDMI te USB priključkom gdje je prvi zadužen za prijenos video signala, a drugi za mogućnost korisničke interakcije putem dodira te struju potrebnu za rad. Ekran nativno podržava Raspberry Pi te nije bilo potrebno dodavati posebne knjižnice ili konfiguracije.





Slika 2.10 Ekran osjetljiv na dodir u 3D ispisanom kućištu

## 2.2 Opis programske podrške

### 2.2.1 Raspberry Pi OS

Raspberry Pi je svestrana platforma te postoji veliki broj različitih operacijskih sustava za različite namjene. Budući da pametna vaga zahtjeva vizualno korisničko sučelje korišten je službeni operacijski sustav za navedenu platformu Raspberry Pi OS with desktop. Uz nekoliko iznimki, većina operacijskih sustava za Raspberry Pi bazirana je na Linuxu [20]. Prednost korištenja operacijskog sustava baziranog na Linuxu je mogućnost korištenja programa i knjižnica otvorenog koda, ali i poznatost alata. Za automatsko pokretanje aplikacije korišteni su *systemd* te *bash* skripte koje se izvršavaju pri podizanju sustava.

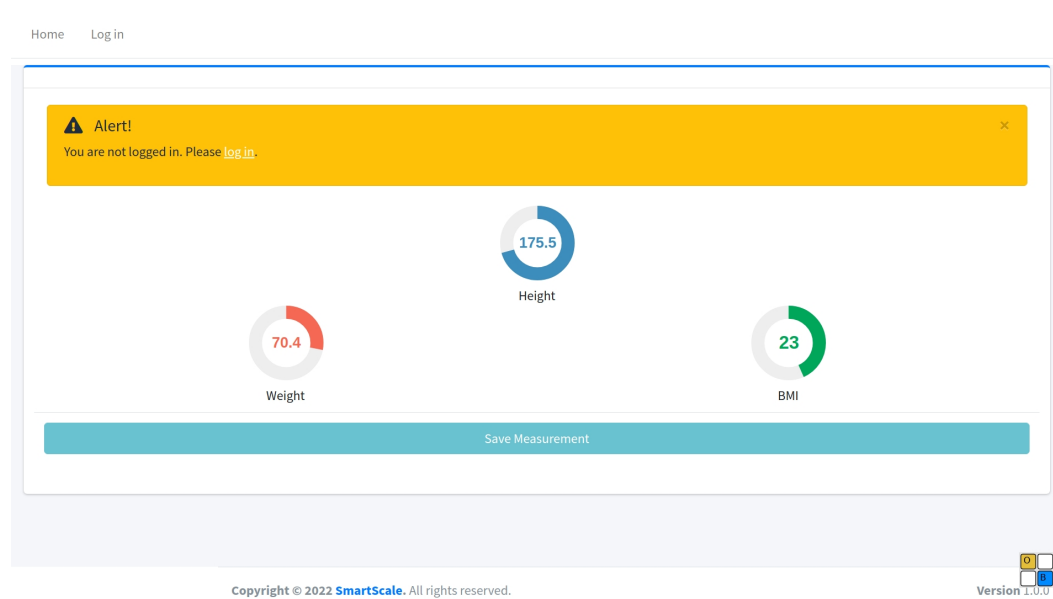
### 2.2.2 Django radni okvir i web aplikacija

Django razvojni okvir baziran je na programskom jeziku Python te je time omogućena integracija programskog koda vezanog za dohvaćanje podataka sa senzora i web aplikacije. Razvojni okvir podržava korištenje virtualnih okoliša za instalaciju i korištenje programskih paketa, što je korisno radi umanjivanja interferencije s ostatkom

## Poglavlje 2. Sklop i programska podrška

sustava. Iako u službenoj dokumentaciji nisu navedene specifikacije potrebne za rad i razvoj Django web aplikacije. Informacije na forumima korisnika govore da najveći utjecaj na zahtjeve radnog okvira ima broj kompleksnih izračuna te broj paralelnih korisnika. Ovaj sustav namijenjen je samo jednom korisniku istovremeno te nema kompleksnih izračuna. Za instalaciju paketa korišten je PIP sustav za upravljanje paketima. Web tehnologije korištene u projektu su HTML, CSS, JavaScript (JS), AJAX, JQuery te knjižnice Bootstrap i AdminLTE.

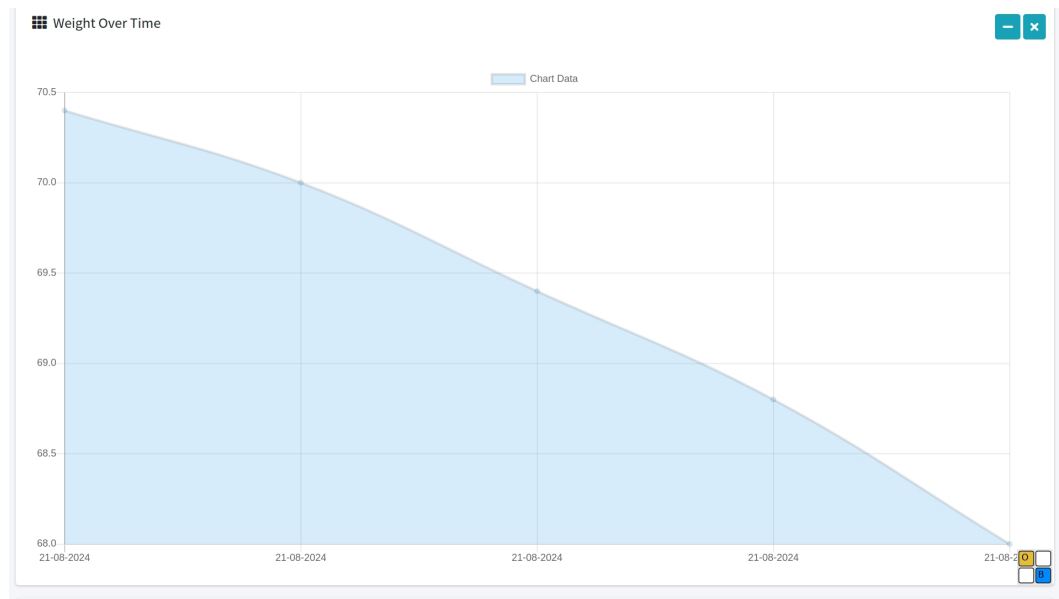
Web aplikacija pametne vage sastoji se od tri razine pristupa: gost, korisnik s računom i administrator. Gost može koristiti način brzog mjerenja gdje na ekranu može samo pratiti svoju visinu, težinu i index tjelesne mase (BMI). Korisnik prijavljen s računom može odabrati način brzog mjerenja, ali može koristiti i način mjerenja s pohranom podataka prikazan na Slici 2.11. Jednom kada su podaci pohranjeni,



Slika 2.11 Prikaz ekrana u načinu rada s mogućnošću pohranjivanja izmjerenih vrijednosti

korisnik može pratiti kretanje svojih mjera kroz vrijeme putem grafova poput prikazanog na Slici 2.12 koji se nalazi na stranici profila. Administrator koristi integrirano Django Admin sučelje gdje ima pristup cjelokupnoj bazi podataka nad kojom ima potpunu kontrolu. Administrator sustava može se postati samo ručnim unosom kroz terminal naredbama Django radnog okvira.

## Poglavlje 2. Sklop i programska podrška



Slika 2.12 Primjer grafa za praćenje izmjerenih vrijednosti unutar stranice profila korisnika

### 2.2.3 Git sustav za verzioniranje

Tijekom rada na projektu korišten je git sustav za verzioniranje te GitHub platforma za repozitorij.

# Poglavlje 3

## Konstrukcija pametne vage

### 3.1 Izrada konstrukcije pametne vage

Konstrukciju pametne vage izradili su kolege studenti strojarstva i elektrotehnike u okviru svojih studentskih projekata. Stella Podgornik izradila je postolje, a Matej Lovrić vertikalnu letvu s kućištem. Potrebno je bilo vijcima spojiti zasebne elemente te je time konstrukcija bila potpuna. Uz postolje nisu bili isporučeni nosači za senzore te je bilo potrebno izbušiti rupe kako bi mjerne ćelije imale dovoljno prostora za savijanje pomičnog elementa pod opterećenjem.

Prije korištenja ispravnost mjernih ćelija testirana je mjernim instrumentom. Ćelije je bilo potrebno zalemiti na kombinator opterećenja a zatim potonjeg spojiti žicama na HX711 modul. Mjerne ćelije same po sebi nemaju predviđene točke za fiksiranje na konstrukciju te je korištena ljepljiva traka. Nakon što su mjerne ćelije sa kombinatorom i modulom bile postavljene, potrebno je bilo izmjeriti udaljenost od kućišta do modula (Između točaka 1 i 2 na Slici 3.1). Kada je udaljenost bila poznata potrebno je bilo zakrimpati konektore na obje strane trakastog kabela.

Pomični element na vertikalnoj letvi na sebi nosi HC-SR04 modul. Prije spajanja na Raspberry Pi potrebno je bilo provjeriti nalazi li se zener dioda za smanjenje napona na Echo pinu kako ne bi došlo do oštećenja. Ponovno je bilo potrebno izraditi kabel koji će spojiti modul i razvodnu ploču.

Razvodna ploča nalazi se u 3D ispisanom kućištu. Shema razvodne ploče izrađena je u ECAD programu EAGLE, a sami vodiči na tiskanoj pločici prilagođeni

### Poglavlje 3. Konstrukcija pametne vage



Slika 3.1 Konstrukcija pametne vage

su postupkom jetkanja. Jetkanje je kemijska reakcija kojom se na površinu tiskane pločice djeluje razrijeđenom kiselinom te se tako uklanja višak bakrenog sloja[21]. Nakon jetkanja potrebno je bilo izbušiti rupe za konektore te zalemiti iste. Na Slici 3.2 prikazano je da su konektori (označeni ETH0 i ETH1 na Slici) spojeni na pojednostavljen način kako bi pločica mogla biti jednostavno izrađena samo na gornjoj



### Poglavlje 3. Konstrukcija pametne vage

su sve komponente spojene na razvodnu ploču, potrebno je bilo napraviti trakasti kabel s 40 vodiča kako bi spojili GPIO pinove uređaja Raspberry Pi s razvodnom pločom. Osim GPIO pinova, spojena su još dva USB sučelja kako bi bilo moguće koristiti USB sučelja na razvodnoj ploči koji su dostupni na lijevoj bočnoj plohi kućišta. Naposljetku je bilo potrebno spojiti ekran osjetljiv na dodir s HDMI te USB kabelom na Raspberry Pi za video signal, napajanje te mogućnost osjetljivosti na dodir.

## 3.2 Izrada programske podrške

Proces izrade programske podrške započinje formatiranjem microSD kartice koja se koristi kao računalna memorija umjesto tvrdog diska. Formatiranje je potrebno kako bi minimizirali mogućnost interferencije prijašnjeg korištenja s trenutnim. Potom je na microSD karticu pomoću programa balenaEtcher zapisan službeni Raspberry Pi OS baziran na debian verziji 10 (Buster) s 5.10.63 verzijom linux jezgre. Kako bi na uređaj mogli instalirati Django radni okvir, potrebno je bilo provjeriti postoji li instalirana zadovoljavajuća verzija programskog jezika Python na uređaju.

Sljedeće je bilo potrebno razviti web aplikaciju koja će korisniku omogućiti očitavanje rezultata mjerenja senzora te mogućnost spremanja te manipulacije istih. Radi se o jednostavnoj aplikaciji te će većina implementacije koja nije vezana za rad senzora biti izostavljena. Očitavanje senzora udaljenosti smješteno je u views.py datoteci kako bi izbjegao dodatne pozive funkcija izvan aplikacije.

---

```
1 GPIO.setmode(GPIO.BCM)
2 GPIO.setwarnings(False)
3 TRIG=24
4 ECHO=23
5 GPIO.setup(TRIG, GPIO.OUT)
6 GPIO.setup(ECHO, GPIO.IN)
7 GPIO.output(TRIG, False)
8 GPIO.output(TRIG, True)
9 time.sleep(0.00001)
10 GPIO.output(TRIG, False)
11 while GPIO.input(ECHO) ==0:
12     pulse_start = time.time()
```

### Poglavlje 3. Konstrukcija pametne vage

```
13 while GPIO.input(ECHO)==1:
14     pulse_end = time.time()
15 pulse_duration = pulse_end - pulse_start
16 distance = pulse_duration * 17150
17 distance = 224 - distance
18 distance = math.ceil(distance * 2) / 2
```

---

#### Isječak koda 3.1 Mjerenje udaljenosti

Isječak koda 3.1 sadrži kod za očitavanje udaljenosti. Kod koristi RPi.GPIO modul za očitavanje s modula. Prvo se koristi `GPIO.setmode()` kako bi se tijekom rada mogle koristiti simbolične oznake GPIO pinova. Kao što je prethodno spomenuto u opisu rada modula, mjeri se vrijeme koje je potrebno da se *Echo* pin vrati iz visokog stanja u nisko stanje. Nakon što se definiralo da se za *Echo* koristi pin 23 te za *Trig* pin 24 podesio se i njihov smjer na ulazni i izlazni respektivno. Za mjerenje se koriste dvije *While* petlje koje će zabilježiti posljednje unix vrijeme kada je pin bio u niskom, odnosno visokom stanju.

$$d = 224\text{cm} - \left( t_p \frac{v_t \times \frac{100\text{cm}}{1\text{m}}}{2} \right) \quad (3.1)$$

Gdje su:

$d$  – udaljenost od senzora [cm]

$t_p$  – vrijeme koje je *Echo* pin proveo u visokom stanju [s]

$v_t$  – brzina zvuka u trenutnim uvjetima [m/s]

Dvije spomenute vrijednosti se oduzmu te se dobije vrijednost  $t_p$  iz Formule 3.1. Za vrijednost  $v_t$  korištena je brzina zvuka u suhom zraku pri 20 stupnjeva Celzijusa od 343 m/s[22] pretvorena u brzinu u centimetrima u sekundi. Dobivenu vrijednost potrebno je bilo prepoloviti imajući u obzir da signal dva puta prelazi udaljenost od senzora do ispitanika. Nakon što je od visine senzora oduzeta dobivena vrijednost mjerenja rezultat je zaokružen na najbliži 0.5 višekratnik.

---

```
1 def write_weight_to_file():
2
3     try:
4         GPIO.setmode(GPIO.BCM)
```



### Poglavlje 3. Konstrukcija pametne vage

```
5     hx = HX711(dout_pin=20, pd_sck_pin=16)
6     err = hx.zero()
7     hx.set_scale_ratio(-23.729865979381444)
8     while True:
9         with open("/home/pi/Desktop/loadcell/HX711-master/
HX711_Python3/measurements.txt", "w") as f:
10             f.write(str(hx.get_weight_mean(10)))
11             f.flush()
12             time.sleep(1)
13
14     except (KeyboardInterrupt, SystemExit):
15         print('Exiting')
16
17     finally:
18         GPIO.cleanup()
19
20 if __name__ == "__main__":
21     write_weight_to_file()
```

---

#### Isječak koda 3.2 Mjerenje mase

Isječak koda 3.2 sadrži kod za očitavanje mase. Kod koristi dodatno HX711 knjižnicu koja sadrži metode za rad s istoimenim modulom. Kao i u prijašnjem isječku koristi se BCM način imenovanja pinova. Inicijalizira se HX711 instanca te postavljaju Dout i Sck pinovi na GPIO20, odnosno GPIO16. Manji nedostatak mjernih ćelija, odnosno knjižnice koju koriste je činjenica da je potrebna kalibracija iste. Kalibracija se u idealnom slučaju treba provesti prije svakog mjerenja. Budući da višekratna kalibracija nije optimalna, ona je podešena u kodu pomoću `set_scale_ratio()` metode s poznatom fiksnom vrijednosti dobivenom inicijalnom kalibracijom. `get_weight_mean()` metoda vraća prosjek određenog broja mjerenja. Korišteno je računanje prosjeka na 10 mjerenja kako bi mjerenje bilo što učestalije. Korištenje prethodno spomenutih metoda u `views.py` datoteci rezultiralo je vremenima učitavanja preko 10 sekundi te je cijeli dio mjerenja mase pretvoren u servis `background_weight_measure` čija je implementacija prikazana u Isječku koda 3.3.

---

```
1 [Unit]
2 Description=Run weight measurement with HX711 in background at
```

### Poglavlje 3. Konstrukcija pametne vage

```
    startup
3
4 [Service]
5 ExecStart=/usr/bin/python3.7 /home/pi/Desktop/loadcell/HX711-
    master/HX711_Python3/djangoLoadcell.py
6 Restart=always
7
8 [Install]
9 WantedBy=multi-user.target
```

---

#### Isječak koda 3.3 Servis za pokretanje mjerenja mase

Spomenuti servis pokreće se tijekom inicijalnog podizanja sustava. `ExecStart` pokreće Python datoteku koja sadrži kod za očitavanje mase sa senzora. `Restart=always` će rezultirati automatskim ponovnim pokretanjem servisa u slučaju greške. Kako bi osigurali da se servis pokrene prilikom pokretanja sustava korišten je `WantedBy=multi-user.target`. Točnije, servis će se pokrenuti u trenutku kada sustav dosegne mogućnost paralelnog korištenja većeg broja korisnika. Kao što je prikazano na Slici 3.2 na liniji devet, masa se zapisuje u datoteku na sustavu `measurements.txt` koju `views.py` učitava u `get_sensor_values()` metodi unutar Django radnog okvira (Isječak koda 3.4).

---

```
1 with open("/home/pi/Desktop/loadcell/HX711-master/HX711_Python3
    /measurements.txt", "r") as f:
2     weightstr = f.read().strip()
3     while weightstr == "":
4         weightstr = f.read().strip()
5     weight = float(weightstr)
6     bmi = weight / (distance **2)
7     vector = [round(distance), round(weight /1000), bmi]
8     GPIO.cleanup()
9     return JsonResponse({'sensor_value':vector})
```

---

Isječak koda 3.4 Učitavanje mjerenja mase iz datoteke i priprema podataka za prikaz

`Views.py` nakon učitane mase i prethodno očitane visine računa index tjelesne mase. Kako se GPIO pinovi nakon izvršenja metode ili nenadanog prekida ne bi ostavili u

### Poglavlje 3. Konstrukcija pametne vage

nedefiniranom stanju poziva se `GPIO.cleanup()`. Nakon toga mjerenja se spremaju u vektor te formatiraju u JSON pa šalju na mjesto poziva metode.

```
1 function updateSensorValue() {
2     $.ajax({
3         url: '{% url 'scaleGui:getSensorValues' %}',
4         type: 'GET',
5         dataType: 'json',
6         success: function(response) {
7             var vector=response.sensor_value;
8             $('knob')[0].value=vector[0];
9             $('knob')[1].value=vector[1];
10            $('knob')[2].value=vector[2];
11            $('knob')[0].trigger('change');
12            $('knob')[1].trigger('change');
13            $('knob')[2].trigger('change');
14
15
16        },
17
18        error: function(xhr, status, error) {
19            console.error(error);
20        }
21    });
22 };
23
24 $(document).ready(function(){
25     updateSensorValue();
26
27     setInterval(function() {
28         updateSensorValue();
29     }, 1500);
30 });
```

---

#### Isječak koda 3.5 Dinamičko ažuriranje vrijednosti mjerenja

U Isječku koda 3.5 prikazano je da se prethodno spomenuta metoda poziva putem

### Poglavlje 3. Konstrukcija pametne vage

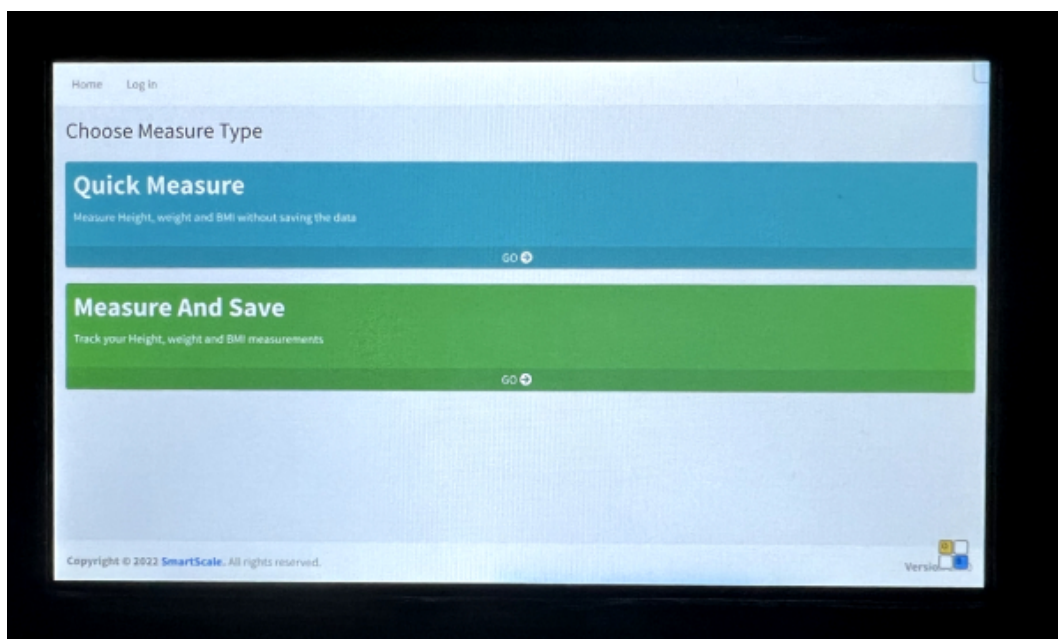
AJAX-a te se tako dinamično osvježava svakih 1500 ms. Kako bi pametna vaga mogla biti korištena kao ugradbeno računalo, potrebno je bilo omogućiti automatsko pokretanje web aplikacije te svih popratnih programa. Za automatsko pokretanje korištena je autostart datoteka LXDE-pi okruženja radne površine kako bi bili sigurni da su u trenutku pokretanja web aplikacije zadovoljeni svi uvjeti za njeno izvršavanje. U autostart datoteku dodana je naredba za pokretanje bash skripte kiosk.sh.

```
1 #!/bin/bash
2
3 unclutter &
4 source /home/pi/ScaleWeb/scaleenv/bin/activate
5 python /home/pi/ScaleWeb/scalesite/manage.py runserver &
6 onboard &
7 chromium-browser --kiosk --disable-infobars --disable-session-
  crashed-bubble --noerrdialogs --disable-translate --force-
  device-scale-factor=1.75 --disable-features=Translate --no-
  first-run "http://127.0.0.1:8000/"
```

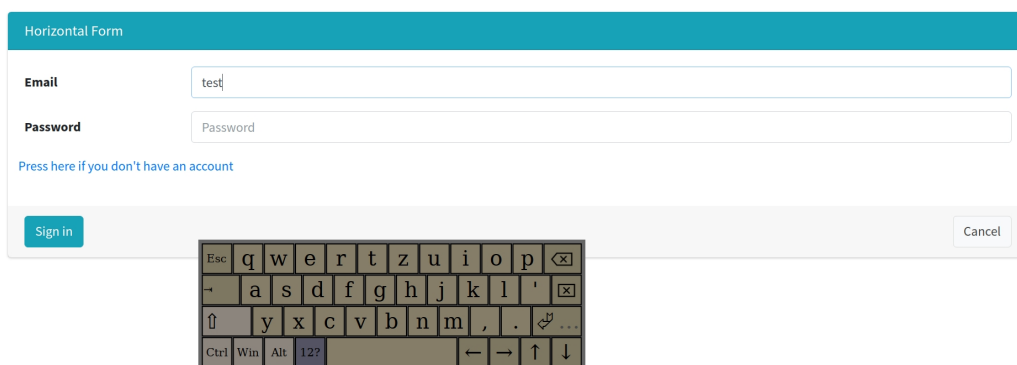
#### Isječak koda 3.6 Kiosk skripta

Spomenuta skripta nalazi se u Isječku koda 3.6. U skripti se prvo poziva naredba *unclutter* kako bi sakrila pokazivač miša, koji nije potreban s obzirom na to da je primarna metoda unosa ekran osjetljiv na dodir. Pomoću naredbe *source* aktivira se virtualno okruženje za rad Django aplikacije koja se pokreće s *runserver*. Sljedeća naredba *Onboard* pokreće virtualnu tipkovnicu u pozadini koja se automatski aktivira u prvi plan u slučaju potrebe za tekstualnim unosom. Virtualna tipkovnica se može aktivirati u prvi plan (Slika 3.5) i pritiskom na plutajuću ikonu prikazanu u donjem desnom kutu na Slici 3.4. Nakon što se sve potrebne skripte i servisi pokrenu, početni ekran izgleda kao na Slici 3.4. Kako bi se dobilo na brzini i uštedilo nešto radne memorije onemogućeno je automatsko pokretanje pojedinih servisa koji nisu potrebni u sustavu. Radi se o *bluetooth*, *cups*, *hciuart*, *alsa-state* i *cups-browsed* servisima koji su onemogućeni sa `systemctl mask`.

### Poglavlje 3. Konstrukcija pametne vage



Slika 3.4 Početni ekran nakon paljenja uređaja



Slika 3.5 Prikaz aktivirane virtualne tipkovnice pri unosu u formu

# Poglavlje 4

## Mjerenja i rezultati

### 4.1 Mjerenja

Zbog potražnje većeg broja mjerenja u dostupnom vremenskom periodu, mjerenja su obavljena na dvije lokacije. Mjerenja od 1 do 11 napravljena su na prvoj lokaciji, dok su mjerenja od 12 do 24 napravljena na drugoj lokaciji. Sama mjerenja odvijala su se u dvije faze koje će biti opisane zasebno u nastavku. Svrha dvije faze mjerenja bila je dobiti referentnu vrijednost ručnim mjerenjem visine i mase te usporediti tu vrijednost s mjerenjima pomoću pametne vage. Od ispitanika se nije tražilo da skidaju teže komade odjeće, prazne džepove niti da skidaju obuću. Bilo je potrebno da ispitanik obavi ručno mjerenje te mjerenje na pametnoj vagi bez promjena visine i težine uzrokovane vlastitim radnjama.

Prije mjerenja potrebno je bilo upaliti pametnu vagu te pokrenuti samu aplikaciju. Budući da se radi o ugradbenom sustavu, sve što je potrebno za pokrenuti pametnu vagu je uključiti ju u struju pomoću micro USB priključka. Nakon uključivanja sustav pomoću `systemctl` servisa izvršava skriptu koja pokreće Django server i instancu Chromium web preglednika. Prema argumentima iz izvršene skripte Chromium preglednik otvara lokalnu adresu na ulazu kojeg poslužuje Django server. Za mjerenja koja su sljedila korišten je jednostavniji od dvaju načina rada pametne vage zbog same tečnosti postupka. Korišteni "Quick Measure" način rada korisniku nudi samo vizualnu informaciju o mjerenim veličinama bez mogućnosti spremanja i praćenja istih kroz određeni vremenski period.

### 4.1.1 Mjerenje mase

Zbog izostanka mjesta na senzoru za pričvršćivanje za okvir na podu bilo je potrebno kalibrirati vrijednosti senzora za težinu između dva mjerenja. Prilikom kalibriranja vrijednosti spomenutog senzora oba puta korištena je vlastita masa koja je iznosila 98 kilograma. Kalibracija je provedena sa odvojenom Python skriptom koja treba biti pokrenuta dok se na mjernim ćelijama nalazi samo stajna ploča. Jednom kada se dobila vrijednost kalibracije koja skalira izmjerene vrijednosti, ona je unesena u web aplikaciju. Prva faza sastojala se od mjerenja mase u kilogramima pomoću kućne vage zaokružene na jednu decimalu. Kako bi mjerenja bila usporediva pametna vaga je također programirana da krajnji rezultat bude u kilogramima zaokružen na jednu decimalu. Nakon što je rezultat mjerenja mase zabilježen, uslijedilo je ručno mjerenje visine te mjerenje na pametnoj vagi koje je također zabilježeno. Time je ispitanik završio svoje sudjelovanje.

### 4.1.2 Mjerenje visine

Ultrazvučni senzor udaljenosti se prilikom oba mjerenja nalazio na identičnoj visini na vertikalnoj letvi te je dodatnim mjerenjem to bilo potvrđeno. Vrijednost udaljenosti od senzora mjerenja postavljeno je na rezoluciju od 0,5 centimetara zato što je ispitivanjem tržišta utvrđeno da pametne vage s beskontaktnim mjerenjem visine većinom imaju rezoluciju 0,5 centimetara. Prva faza mjerenja sastojala se od ručnog mjerenja visine pomoću krojačkog metra. Od ispitanika se tražilo da stane petama do zida, bude uspravnog držanja te da drži glavu u neutralnom položaju. Neutralni položaj glave definira se kao položaj glave u stajaćem položaju opuštenog tijela i glave prilikom kojeg ispitanik gleda u udaljenu imaginarnu točku u razini očiju[23]. Kada se ispitanik namjestio, jednim krakom kutnika namještenim na vrh tjemena i drugim krakom prslonjenim na zid označena je točka na zidu koja predstavlja visinu ispitanika. Nakon što se ispitanik pomaknuo u stranu izmjerena je udaljenost od poda do spomenute točke te je vrijednost zabilježena. Potom je uslijedilo istovremeno mjerenje visine i težine na pametnoj vagi. Ponovo je od ispitanika zatraženo da uspravno stane na vagu s glavom u neutralnom položaju u kojem je trebao ostati nekoliko sekundi da bi se zabilježio ispravan rezultat.

## 4.2 Rezultati

Rezultati mjerenja prikazani su u Tablici 4.1 i Tablici 4.2. Obje tablice sadrže tri stupca. Lijevi stupac predstavlja redni broj mjerenja, srednji stupac predstavlja vrijednost izmjerenu pomoću senzora na pametnoj vagi, a desni stupac predstavlja ručno izmjerenu vrijednost tehnikama koje su prethodno opisane. Uvidom u mjerenja prikazanim u Tablicama 4.1 i 4.2 može se primijetiti odstupanje u mjerenjima. Odstupanja naizgled imaju određenu tendenciju u pozitivnom odnosno negativnom smjeru u slučaju mjerenja težine odnosno visine. Pojedina mjerenja pomoću senzora su veća, dok su pojedina mjerenja manja nego ručno izmjerene vrijednosti te nema izraženog odstupanja samo u jednom predznaku.

### 4.2.1 Rezultati mjerenja mase

Odstupanje u mjerenjima mase možemo vizualizirati Bland-Altmanovim grafom na Slici 4.1. S navedenog grafa možemo primijetiti da nešto veći broj mjerenja ima pogrešku u negativnom predznaku. U Tablici 4.3 može se uočiti da je srednja pogreška 0,293, a standardna devijacija pogreške 1,6297. Interval pogreške je otprilike  $[-4, 4]$ . S grafa se također može uočiti da je iznos greške nešto manji u intervalu  $[68, 84]$ . U prethodno spomenutom intervalu nalazi se i najveći broj mjerenja. Prosječno apsolutno odstupanje rezultata mjerenog s pametnom vagom u odnosu na ručno izmjereni rezultat je 1,27 kg. Iz grafa nije moguće naći uzorak greške u mjerenjima usprkos tome što je vaga kalibrirana na 98 kilograma te je očekivano da će greška konvergirati prema nuli s obje strane što je mjerenje bliže kalibriranoj vrijednosti. Na grafu sa Slike 4.2 vidljivo je da greška ne konvergira prema nuli što se mjerenja mase približavaju masi od 98 kg, ali na grafu je vidljivo da s trenutnim podacima mjerenja greška ima tendenciju pada na intervalu  $[60, 98]$  s ekstremima koji se smanjuju, ali i dalje postoje. Na intervalu  $[98, 120]$  upisana su samo dva mjerenja te iz njih nije moguće napraviti značajniju opservaciju. Za razliku od Bland-Altmanovog grafa, ovdje su odstupanja prikazana u postocima. Interval greške je otprilike  $[-3.7, 5.5]$  te osim otprilike pet nešto većih odstupanja većina mjerenja ima apsolutnu grešku manju od 2,5% .



Poglavlje 4. Mjerenja i rezultati

Tablica 4.1 Izmjerene vrijednosti visine.

redni broj mjerenja	vrijednost izmjerena pomoću senzora [cm]	ručno izmjerena vrijednost
1	173,0	174,0
2	180,0	183,0
3	175,0	176,0
4	184,0	187,0
5	178,0	179,0
6	185,0	185,0
7	174,0	175,0
8	176,5	177,0
9	178,0	177,5
10	177,5	177,5
11	167,0	165,5
12	171,0	172,0
13	178,0	178,0
14	168,0	166,0
15	170,0	169,5
16	183,0	184,0
17	177,0	178,5
18	183,0	185,0
19	180,0	181,5
20	175,0	175,0
21	159,0	158,0
22	165,0	167,0
23	162,5	162,0
24	169,0	169,0
25	185,5	187,0
26	163,0	162,5
27	175,0	175,0
28	190,5	191,5
29	180,5	179,5

Poglavlje 4. Mjerenja i rezultati

Tablica 4.2 Izmjerene vrijednosti mase.

redni broj mjerenja	vrijednost izmjerena pomoću senzora [kg]	ručno izmjerena vrijednost
1	89,9	90,3
2	112,0	116,2
3	88,3	87,2
4	102,0	100,0
5	86,2	83,4
6	84,1	82,5
7	69,0	68,7
8	85,5	83,0
9	75,7	76,6
10	80,3	80,7
11	91,2	89,5
12	79,0	79,2
13	85,4	87,8
14	58,1	60,5
15	73,0	71,3
16	91,2	91,1
17	83,9	84,0
18	76,0	74,8
19	95,0	96,5
20	97,0	97,5
21	67,2	63,7
22	64,0	62,3
23	65,0	64,5
24	68,0	67,7
25	100,7	101,3
26	61,5	61,4
27	77,4	78,0
28	105,8	105,0
29	85,8	85,0

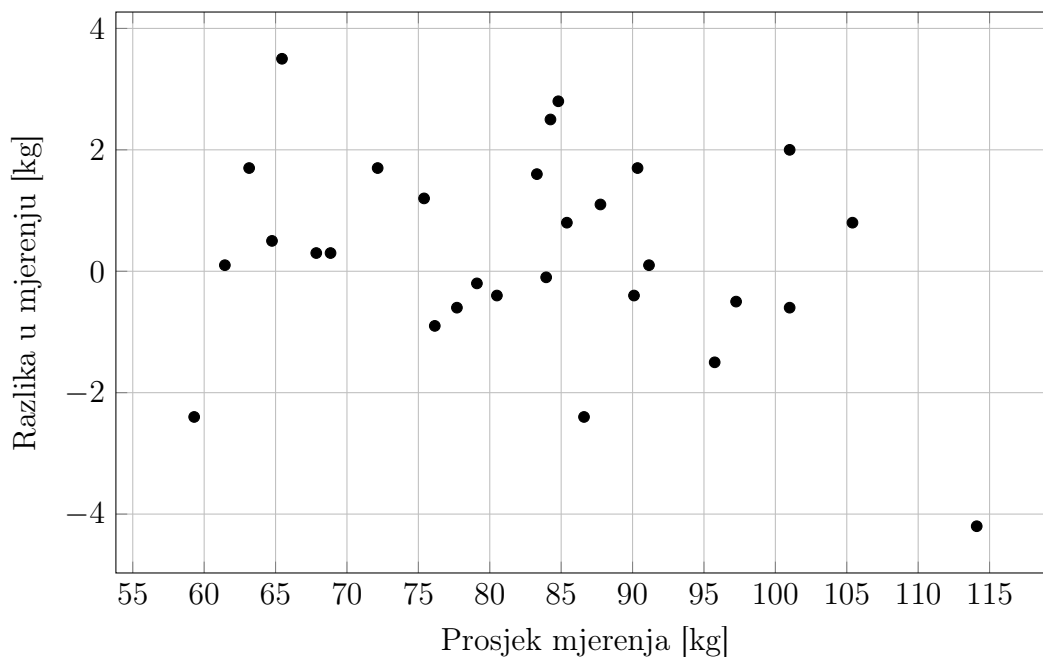
## Poglavlje 4. Mjerenja i rezultati

Ukupan broj mjerenja	Srednja pogreška	Maksimalna pogreška	Standardna devijacija pogreške
29	0,293	-4,2	1,629667

Tablica 4.3 Ukupan broj mjerenja te vrijednosti srednje pogreške, maksimalne pogreške i standardne devijacije pogreške mase

### 4.2.2 Rezultati mjerenja visine

Odstupanje u mjerenjima visine možemo vizualizirati Bland-Altmanovim grafom na Slici 4.3. S navedenog grafa možemo primijetiti da nešto veći broj mjerenja ima pogrešku u negativnom predznaku. U Tablici 4.4 vidljivo je da srednja pogreška iznosi 0,5 , a standardna devijacija 1,2034. Interval pogreške je  $[-3, 2]$ . Prosječno apsolutno odstupanje rezultata mjerenog s pametnom vagom u odnosu na ručno izmjereni rezultat je 1,07 cm. S grafa se može uočiti da je iznos greške manji u intervalu  $[169, 178]$ . U prethodno spomenutom intervalu nalazi se i najveći broj



Slika 4.1 Bland-Altman Graf. Os x prikazuje prosječnu vrijednost težine u kilogramima ručnog i mjerenja pomoću pametne vage. Os y prikazuje razliku između ručnog mjerenja i mjerenja pomoću pametne vage

#### Poglavlje 4. Mjerenja i rezultati

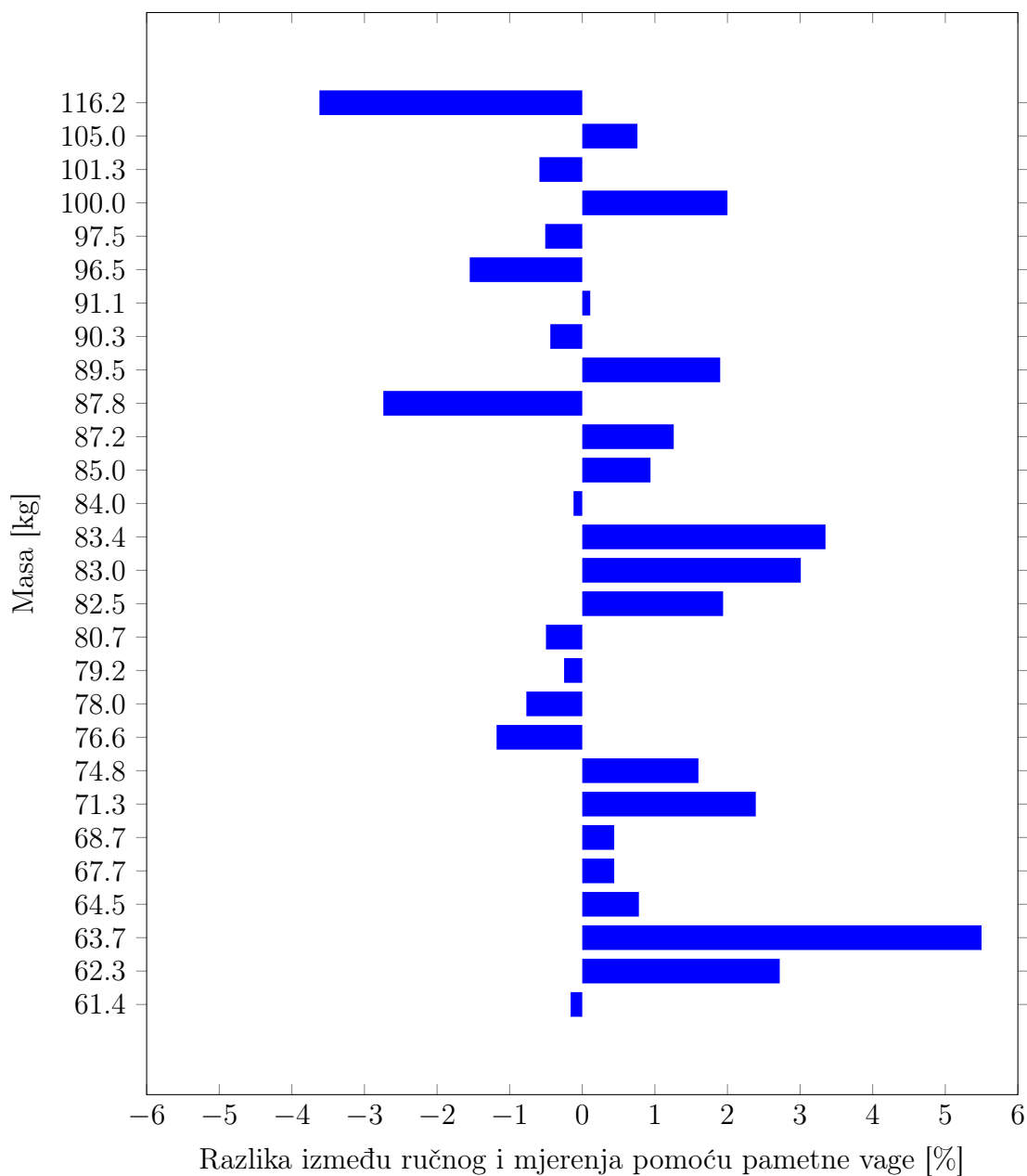
mjerenja. Iz grafa nije moguće naći značajniji uzorak greške u mjerenjima.

Ukupan broj mjerenja	Srednja pogreška	Maksimalna pogreška	Standardna devijacija pogreške
29	-0,5	-3	1,20344334

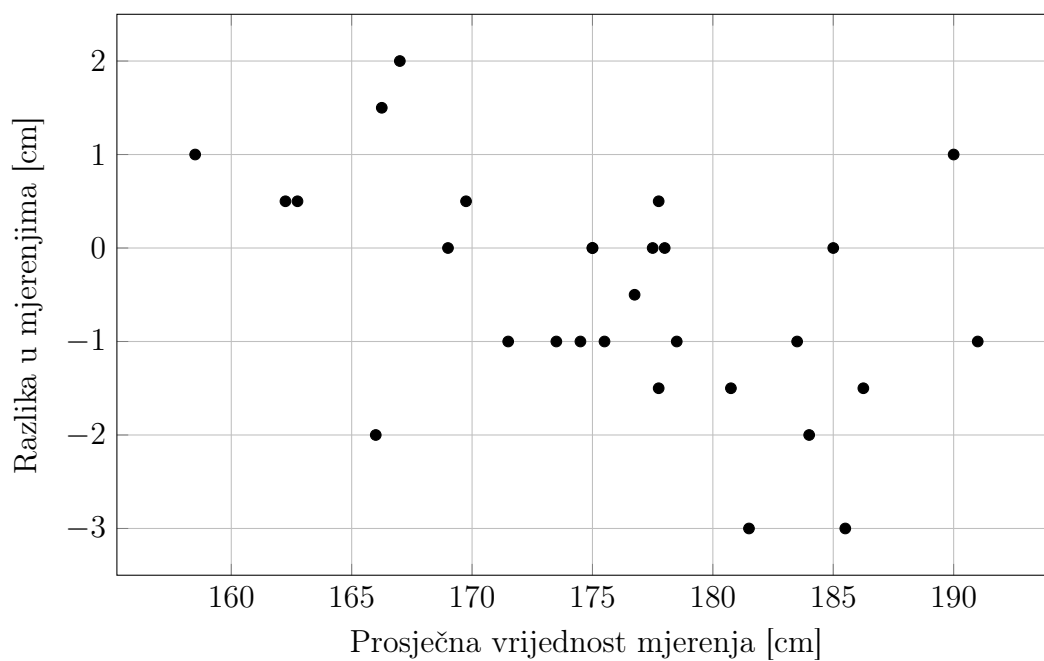
Tablica 4.4 Ukupan broj mjerenja te vrijednosti srednje pogreške, maksimalne pogreške i standardne devijacije pogreške visine

Na grafu sa Slike 4.4 moguće je primijetiti da su odstupanja u postocima nešto manja od odstupanja u mjerenju mase te se nalaze na intervalu  $[-1.7, 1.25]$ . Odstupanja u negativnom predznaku osim tri najveća odstupanja imaju odstupanje manje od 1%.

Poglavlje 4. Mjerenja i rezultati

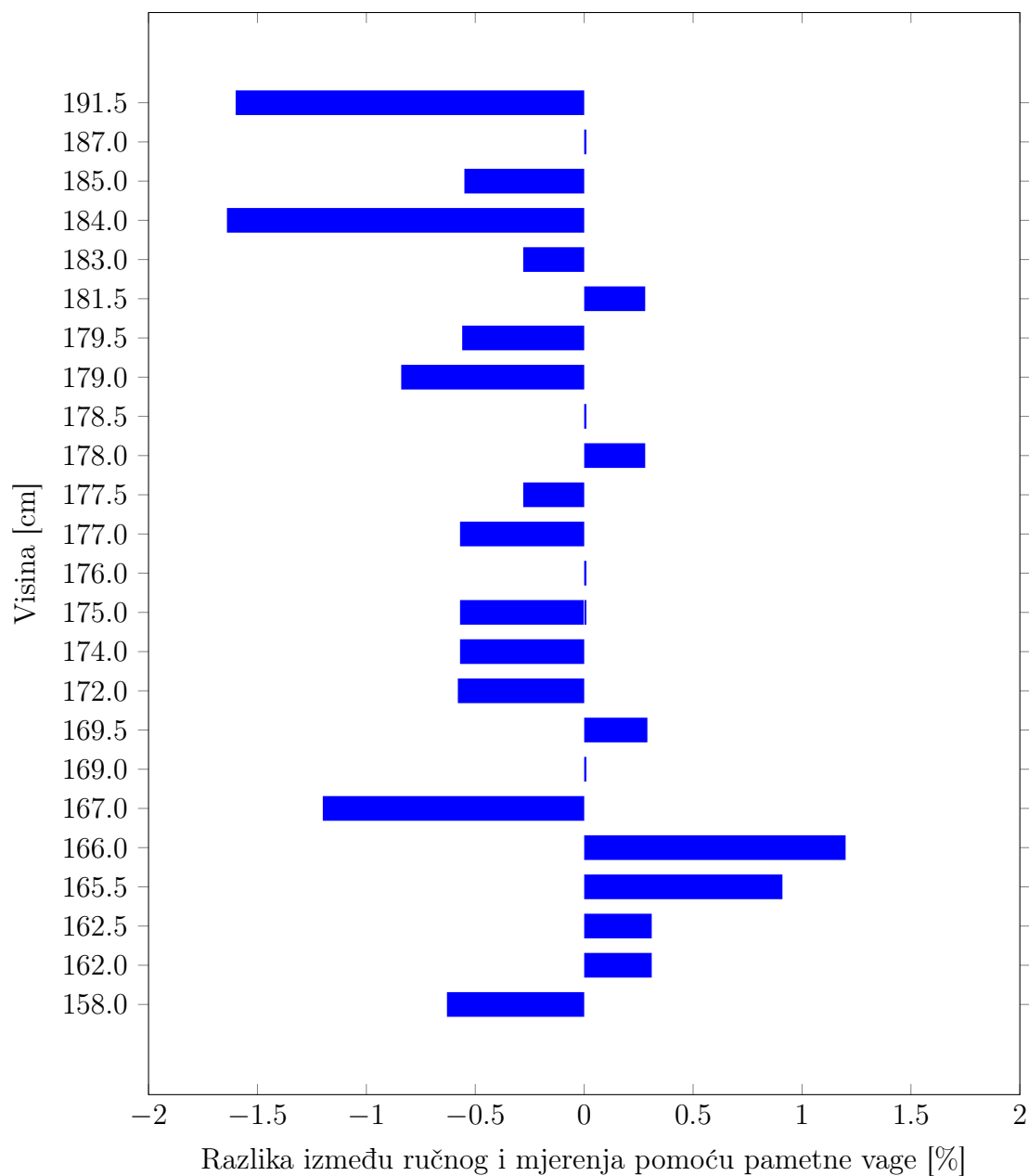


Slika 4.2 Graf odstupanja. Na X osi nalaze se vrijednosti razlike mjerenja pametne vage u odnosu na ručno mjerenje. Na Y osi nalazi se ručno izmjerena masa poredana uzlazno



Slika 4.3 Bland-Altman Graf. Os x prikazuje prosječnu vrijednost visine u centimetrima ručnog i mjerenja pomoću pametne vage. Os y prikazuje razliku između ručnog mjerenja i mjerenja pomoću pametne vage

Poglavlje 4. Mjerenja i rezultati



Slika 4.4 Graf odstupanja. Na X osi nalaze se vrijednosti razlike mjerenja pametne vage u odnosu na ručno mjerenje. Na Y osi nalazi se ručno izmjerena visina poredana uzlazno

# Poglavlje 5

## Diskusija

### 5.1 Iznos greške pri mjerenju visine

Prilikom istraživanja tržišta za slične pametne vage koje mjerenje visine baziraju na ultrazvučnom senzoru propisana greška najčešće je  $\pm 0.5$  cm. U poglavlju Rezultati mjerenja visine ustanovljeno je da je apsolutna prosječna pogreška 1,07 cm. Radi se o 114% većoj grešci za koju postoji nekoliko pretpostavki uzroka iste. Nosač HC-SR04 modula izrađen je od plastike debljine 2 mm. Kako bi senzor imao najbolju moguću poziciju za ispravno mjerenje, on se nalazi na samom rubu spomenutog plastičnog nosača te je moguće zamijetiti savijanje nosača. Uz savijanje nosača zamijećeno je da se i sama baza nosača klizi prema dolje što je moguće primijetiti na Slici 5.1. U poglavlju Rezultati primijećeno je da je većina mjerenja visine imala odstupanje od ručnog mjerenja s negativnim predznakom. Crna crta na lijevoj bočnoj plohi vertikalne letve sa Slike označena je kao vrijednost gdje je nosač inicijalno postavljen tako da se podudara gornji brid počne plohe nosača sa spomenutom crtom. Budući rad obuhvaćao bi unaprjeđenje nosača tako da bi trebao biti manje savitljiv te bolje učvršćen na vertikalnu letvu.

Drugi mogući uzorak za manju preciznost mjerenja je veća gustoća kose na većoj udaljenosti od tjemena ili frizura koja značajni volumen kose ima vertikalno uzdignutu iznad tjemena. Prilikom mjerenja subjektivno je procijenjeno imaju li ispitanici frizuru koja bi mogla utjecati na preciznost mjerenja. Radi se o mjerenjima broj 1,3,4,8,15,21 i 25. Apsolutna prosječna greška navedenih mjerenja iznosi 1,42 cm odnosno od 32,4% većoj apsolutnoj prosječnoj grešci. Ako zanemarimo mje-





Slika 5.1 HC-SR04 modul i pripadajući nosač

renja gdje su ispitanici imali frizure koje bi mogle utjecati na preciznost mjerenja prosječna apsolutna greška je 0,72 cm odnosno 31% više nego uobičajena propisana greška drugih pametnih vaga na tržištu.

## 5.2 Položaj mjernih ćelija

U poglavlju Rezultati moguće je bilo vidjeti da su rezultati mjerenja težine nekonzistentni. U prijašnjem projektu u okviru kolegija Ugradbeni računalni sustavi rezultati mjerenja težine imali su manje odstupanje nego mjerenja provedena tijekom testiranja pametne vage. Jedan od razloga zasigurno leži u nedostatku nosača za korišteni tip mjerne ćelije. Prethodno korišteni tip mjerne ćelije imala je rupe predviđene za pričvršćivanje na postolje i mjernu plohu dok trenutne ćelije nemaju. Svakim stupanjem na vagu te silaženjem s nje izvjesno je da se položaj mjerne ćelije promijenio u odnosu na postolje, ali i stajnu ploču. S promjenom položaja, promijenila bi se i količina deformacije tenzometrijske trake, a samim time i signal koji bi Raspberry Pi primio na ulaznim pinovima. Budući rad na pametnog vagi obuhvaćao bi izradu

potrebnog nosača. Uz nosač, za preciznije rezultate trebalo bi fiksirati spoj mjerne ćelije i stajne ploče.

### 5.3 Šum u sustavu mjerenja mase

Tijekom izrađivanja pametne vage zamijećena je fluktuacija signala pri mjerenju mase. Pretpostavljeno je bilo da bi se moglo raditi o šumu uzrokovanim duljinom kabela između uređaja Raspberry Pi i hx711 modula. Pretraživanjem dostupnih članaka i foruma zaključeno je da je bitno samo da je udaljenost između mjerne ćelije i hx711 modula što kraća. Stvarni uzročnici šuma bili su loši lemovi na pločici kombinatora te izrazito malen poprečni presjek žica mjernih ćelija koje su se tijekom spajanja nerijetko odvajale od pločice.

### 5.4 Raspberry Pi 3B+ kao platforma

Rad web aplikacije u aplikaciji Chromium bio je tijekom cijelog procesa razvoja usporen tijekom učitavanja stranica. Pretpostavljeni razlog tome leži u specifikacijama samog Raspberry Pi 3 B+ modela. U Tablici 5.1 možemo primijetiti da uređaj koristi

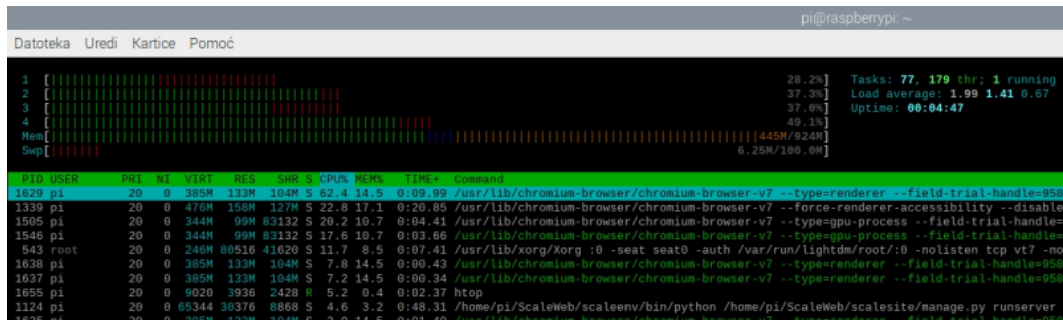
Komponenta	Detalji
Procesor	1.4GHz 64-bitni četverojezgreni ARM Cortex-A53 (Broadcom BCM2837B0)
Radna memorija	1 GB LPDDR2 SDRAM
Mrežna kartica	Gigabit Ethernet (300 Mbps), Dual-band 802.11ac Wi-Fi
Bluetooth	Bluetooth 4.2/BLE
USB Sučelja	4 x USB 2.0
GPIO	40-pin header
Napajanje	5 V/2.5 A DC micro USB konektor
Dimenzije	85.6 mm x 56.5 mm x 17 mm

Tablica 5.1 Specifikacije Raspberry Pi modela 3B+

četverojezgreni procesor nižeg takta s proizvodnim datumom iz 2016. godine. Uz slabiji procesor, uređaj koristi i izrazito zastarjelu LPDDR2 memoriju od samo 1 Gigabajta. Pogledom na ispis aplikacije htop na Slici 5.2 koja je zabilježena u trenutku učitavanja stranice brzog mjerenja vidimo da navedene komponente ne uzrokuju usko

## Poglavlje 5. Diskusija

grlo. U trenutku zabilježenja slike preostalo je još 445 od 924 MB memorije te je i opterećenje po jezgrama bilo manje od 50 posto na sve četiri jezgre. Aplikacija je zamišljena da ju istovremeno koristi samo jedan korisnik te je pretpostavljeno da opterećenje sustava ne bi trebalo porasti puno više od izmjerenih vrijednosti na Slici 5.2.



```
pi@raspberrypi: ~  
Datoteka Uredi Kartice Pomoć  
1 [|||||] 28.2% Tasks: 77, 179 thr: 1 running  
2 [|||||] 37.3% Load average: 1.99 1.41 0.67  
3 [|||||] 37.0% Uptime: 00:04:47  
4 [|||||] 49.1%  
Mem[|||||] 1445M/924M  
Swp[|||||] 6.25M/100.0M  
PID USER PRI NI VIRT RES SHR S CPU% MEM% TIME+ Command  
1629 pi 20 0 385M 113M 104M S 62.4 14.5 0:09.99 /usr/lib/chromium-browser/chromium-browser-v7 --type=renderer --field-trial-handle=958  
1339 pi 20 0 470M 158M 127M S 22.8 17.1 0:20.85 /usr/lib/chromium-browser/chromium-browser-v7 --force-renderer-accessibility --disable  
1505 pi 20 0 344M 99M 83132 S 20.2 10.7 0:04.41 /usr/lib/chromium-browser/chromium-browser-v7 --type=gpu-process --field-trial-handle=958  
1546 pi 20 0 344M 99M 83132 S 17.6 10.7 0:03.66 /usr/lib/chromium-browser/chromium-browser-v7 --type=gpu-process --field-trial-handle=958  
543 root 20 0 246M 86516 41620 S 11.7 8.5 0:07.41 /usr/lib/xorg/Xorg :0 -seat seat0 -auth /var/run/lightdm/root/:0 -nolisten tcp vt7 -no  
1638 pi 20 0 385M 133M 104M S 7.8 14.5 0:00.43 /usr/lib/chromium-browser/chromium-browser-v7 --type=renderer --field-trial-handle=958  
1637 pi 20 0 385M 133M 104M S 7.2 14.5 0:00.34 /usr/lib/chromium-browser/chromium-browser-v7 --type=renderer --field-trial-handle=958  
1655 pi 20 0 9020 3936 2428 R 5.2 0.4 0:02.37 htop  
1124 pi 20 0 65344 30376 8868 S 4.6 3.2 0:48.31 /home/pi/ScaleWeb/scaleenv/bin/python /home/pi/ScaleWeb/scalesite/manage.py runserver  
1636 pi 20 0 385M 133M 104M S 3.6 14.5 0:01.40 /usr/lib/chromium-browser/chromium-browser-v7 --type=renderer --field-trial-handle=958
```

Slika 5.2 Prikaz ispisa aplikacije htop

# Poglavlje 6

## Zaključak

Zadatak ovog rada bio je osmisliti hardversko rješenje i programsku podršku za dobitvenu konstrukciju pametne vage koja će mjeriti tjelesnu težinu stupanjem na postolje vage te istovremeno beskontaktno mjeriti visinu. Uz znanja vezana za računarstvo bilo je potrebno posjedovati i nadograditi osnovna znanja iz elektrotehnike kako bi se mogao razviti cjelokupan ugradbeni sustav. Pametna vaga namijenjena je svima te je njezin cilj podizanje kolektivne svijesti o prekomjernoj tjelesnoj težini i pretilosti koje se smatraju globalnom epidemijom od strane Svjetske zdravstvene organizacije.

Pametna vaga je implementirana kao ugradbeni sustav sa sensorima za visinu i masu. Raspberry Pi 3 B+ korišten je kao središnji dio sustava zadužen za obradu podataka sa senzora te kao korisničko sučelje pomoću web aplikacije razvijene u Django programskom okviru s ekranom osjetljivim na dodir. Za modularnost sustava izrađene su tiskane pločice korištene kao razvodna ploča za sučelja s uređaja Raspberry Pi te zasebna pločica za napajanje. U radu su opisani principi rada korištenih senzora. Wheatstoneov most koji deformacijom tenzometrijskih traka tj. promjenom otpora na izlazu stvara napon koji se analogno-digitalnom pretvorbom i pojačanjem može očitati na pinovima uređaja Raspberry Pi. Mjerenje udaljenosti izvedeno je pomoću refleksije ultrazvučnih valova te mjerenju vremena potrebnog za povratak vala . Prezentirani su rezultati mjerenja te je uspoređena preciznost sustava u odnosu na slične uređaje dostupne na tržištu gdje je zamijećeno da izrađeni sustav ima nešto manju preciznost. U diskusiji su analizirani problemi sustava te ponuđeni vjerojatni uzroci i poboljšanja za iste.

Budući rad obuhvaćao bi poboljšanja na konstrukciji koja bi osigurala fiksiranost

## *Poglavlje 6. Zaključak*

senzora kako bi se povećala preciznost mjerenja. Raspberry Pi 3 B+ 8 je godina staro računalo te bi korištenje novijih inačica tog računala poboljšalo iskustvo korisnika. Za poboljšanje korisničkog iskustva u vidu brzine učitavanja web stranica unutar web aplikacije bilo bi uputno zamijeniti osam godina star Raspberry Pi 3 B+ model s novijom inačicom.

## Literatura

- [1] M. S. Tremblay, R. C. Colley, T. J. Saunders, G. N. Healy i N. Owen, “Physiological and health implications of a sedentary lifestyle,” *Applied physiology, nutrition, and metabolism*, sv. 35, br. 6, str. 725–740, 2010.
- [2] Ž. Šnjarić, “Debljina kao javnozdravstveni problem,” disertacija, University North. University centre Varaždin. Department of Nursing, 2021.
- [3] P. Kuštro, “Izgradnja ugradbenih sustava korištenjem Yocto Projecta,” disertacija, University of Zagreb. Faculty of Electrical Engineering i Computing, 2015.
- [4] I. Heidl, *Inženjerski priručnik - temelji inženjerskih znanja*. Matutinović, Želimir (ur.): Školska knjiga, 1996.
- [5] Pristupljeno: 2024-08-05. adresa: <https://www.allaboutcircuits.com/textbook/direct-current/chpt-9/strain-gauges/>.
- [6] J.-Y. Yoon, *Introduction to biosensors: from electric circuits to immunosensors*. Springer, 2016.
- [7] F. Zelić, “Idejno konstrukcijsko rješenje uređaja za mjerenje aerodinamičnih opterećenja,” mag. rad, Sveučilište u Zagrebu Fakultet strojarstva i brodogradnje Zagreb, 2020.
- [8] Pristupljeno: 2024-08-07. adresa: <https://learn.sparkfun.com/tutorials/load-cell-amplifier-hx711-breakout-hookup-guide/all?print=1>.
- [9] Pristupljeno: 2024-08-20. adresa: <https://www.sparkfun.com/datasheets/Sensors/loadsensor.pdf>.
- [10] J. Baričević, “Projektiranje sustava elektroničke vage za mjerenje mase košnice,” mag. rad, Sveučilište u Zagrebu, Fakultet strojarstva i brodogradnje, 2017.

## LITERATURA

- [11] Pristupljeno: 2024-08-10. adresa: <https://www.sparkfun.com/products/15569>.
- [12] Pristupljeno: 2024-08-13. adresa: <https://cdn.sparkfun.com/datasheets/Sensors/Proximity/HCSR04.pdf>.
- [13] Pristupljeno: 2024-08-15. adresa: <https://www.raspberrypi.com/documentation/computers/raspberry-pi.html>.
- [14] L. Vukonić i M. Tomić, "Ultrasonic Sensors in IoT Applications," *2022 45th Jubilee International Convention on Information, Communication and Electronic Technology (MIPRO)*, IEEE, 2022., str. 415–420.
- [15] M. Sałuch, D. Tokarski, T. Grudniewski i dr., "Raspberry PI 3B+ microcomputer as a central control unit in intelligent building automation management systems," *MATEC Web of Conferences*, EDP Sciences, sv. 196, 2018., str. 04 032.
- [16] G. Bekaroo i A. Santokhee, "Power consumption of the Raspberry Pi: A comparative analysis," *2016 IEEE International Conference on Emerging Technologies and Innovative Business Practices for the Transformation of Societies (EmergiTech)*, IEEE, 2016., str. 361–366.
- [17] S. Monk, *Programming the Raspberry Pi: getting started with Python*. Mcgraw-hill, 2013.
- [18] R. Suehle i T. Callaway, *Raspberry Pi Hacks: Tips & Tools for Making Things with the Inexpensive Linux Computer*. " O'Reilly Media, Inc.", 2013.
- [19] J. H. Krantz, L. D. Silverstein i Y.-Y. Yeh, "Visibility of transmissive liquid crystal displays under dynamic lighting conditions," *Human factors*, sv. 34, br. 5, str. 615–632, 1992.
- [20] A. K. Dennis, *Raspberry pi computer architecture essentials*. Packt Publishing Ltd, 2016.
- [21] M. Babić, "Gitarska overdrive pedala," mag. rad, Sveučilište Josipa Jurja Strossmayera u Osijeku Fakultet elektrotehnike, računarstva i informacijskih tehnologija Osijek, 2022.
- [22] A. J. Zuckerwar, "Speed of sound in fluids," *Handbook of Acoustics*, str. 61–71, 1998.

## LITERATURA

- [23] A. Lundström, F. Lundström, L. Le Bret i C. Moorrees, "Natural head position and natural head orientation: basic considerations in cephalometric analysis and research," *European Journal of Orthodontics*, sv. 17, br. 2, str. 111–120, 1995.



## Pojmovnik

**AJAX** Asynchronous JavaScript and XML. 2, 14, 24

**BMI** Body Mass Index. 1, 14

**CSS** Cascading Style Sheet. 2, 14

**ECAD** Electronic computer-aided design. 16

**GPIO** General-purpose input/output. 8–11, 19, 20, 22, 38

**HDMI** High-Definition Multimedia Interface. 12, 19

**HTML** HyperText Markup Language. 2, 14

**IPS** In-plane switching. 12

**JS** JavaScript. 2, 14

**JSON** JavaScript Object Notation. 23

**LCD** Liquid-crystal display. 11

**SoC** System-on-chip. 10

**SPI** Serial Peripheral Interface. 10

**UART** Universal asynchronous receiver-transmitter. 10

**USB** Universal Serial Bus. 10–12, 19, 26, 38

# Sažetak

Rad opisuje principe rada ultrazvučnog senzora udaljenosti i mjernih ćelija baziranih na Wheatstoneovom mostu. Dan je pregled korištenog tehnološkog stoga, konstrukcije i hardvera. Demonstriran je proces prikupljanja i obrade podataka web aplikacije za prikaz na korisničkom sučelju. Prikazan je skup servisa i skripti potreban za rad pametne vage u kiosk načinu rada. Opisan je proces ručnog mjerenja i mjerenja pomoću pametne vage te je dan pregled istih. Rezultati dobiveni mjerenjima pametnom vagom imali su veće odstupanje, nego slični proizvodi na tržištu. Objašnjeni su mogući razlozi većih odstupanja u odnosu na tržište te su predložena moguća poboljšanja u provedbi mjerenja i samom sklopu.

***Ključne riječi*** — **Wheatstoneov most, ultrazvučni senzor udaljenosti, ugradbeni računalni sustav, mjerna ćelija, Python, Django**

## Abstract

The paper describes the working principles of the ultrasonic distance sensor and load cells based on the Wheatstone bridge. An overview of the utilized technological stack, construction and hardware is given. The process of collecting and processing sensor data by web application for display on the user interface was demonstrated. The set of services and scripts required for the operation of the smart scale in kiosk mode is presented. The process of manual measurement and measurement using a smart scale is described and a general overview is given. The results obtained by measurements with a smart scale had a greater deviation than similar products on the market. Possible reasons for larger deviations compared to the market are listed, and possible improvements in the implementation of measurements and the assembly itself are suggested.

***Keywords*** — **Wheatstone bridge, ultrasonic distance sensor, embedded computer system, load cell. Python, Django**