



Osnovna šola Polje

Raziskovalna naloga

# Spletna aplikacija za generiranje nalog iz fizike

Računalništvo

Avtor:

Matej Lajevec, 9. b

Mentorica:

mag. Polona Theuerschuh

Ljubljana, 2021

## **ZAHVALA**

Zahvaljujem se mentorici mag. Poloni Theuerschuh za podporo in pomoč pri raziskovalni nalogi.

Zahvaljujem se tudi svoji družini, ki me je podpirala pri delu.

Matej Lajevec

## **POVZETEK**

Letošnje šolsko leto zaradi COVID-19 poteka pouk na daljavo z uporabo računalnika.

Pouk na daljavo naj bi zagotavljal nespremenjeno raven prejetega znanja.

Za potrebe učenja na daljavo sem izdelal spletno aplikacijo za učenje in ocenjevanje fizike, ki zagotavlja večje število avtomatsko generiranih nalog za učenje in ponuja rešitev za bolj pošteno ocenjevanje znanja v času dela na daljavo.

Uporabnik aplikacije lahko na spletni strani dostopa do vnaprej pripravljenih fizikalnih nalog, katerih vhodnih podatki pa so vedno razlikujejo, saj jih program generira naključno. Program nato preveri odgovor, ki ga je vpisal uporabnik, na koncu pa njemu oz. (v primeru ocenjevanja) učitelju, posreduje rezultate.

Aplikacija je narejena iz različnih programskih jezikov (HTML, CSS, JS) in ločena na dve verziji: Vadnik je namenjen utrjevanju, Ocenjevalnik pa preverjanjem ali ocenjevanjem znanja.

# 1 UVOD

V zadnjem času je šola zaradi epidemije virusa Covid-19 potekala na daljavo.

Pouk in ocenjevanja smo imeli preko google obrazcev, zooma itd.

Zaradi ovir, ki jih ta čas prinaša, sem se odločil, da bom za raziskovalno nalogo naredil spletno aplikacijo, ki bo učencem in učiteljem olajšala delo v času izolacije.

V okviru raziskovalne naloge sem naredil aplikacijo, ki sama generira številske podatke pri nalogah iz fizike ali matematike, saj je nadzorovanje učencev pri online testih težko izvedljivo. Tako si učenci ne bi mogli med sabo pošiljati rezultatov in bi bila možnost za goljufanje veliko manjša.

Raziskovalna naloga je sestavljena iz teoretičnega dela, kjer povem več o sestavi programske opreme in iz raziskovalnega dela, kjer razložim delovanje aplikacije. Za njeno razumevanje pa je potrebno razumevanje teoretičnih izhodišč.

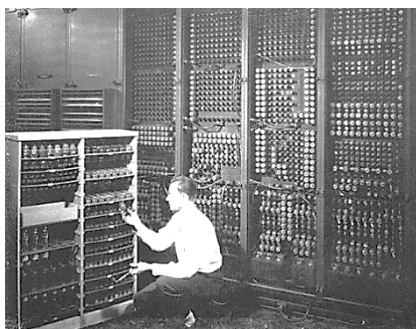
Aplikacija je napisana v treh programskih jezikih, v CSS, JavaScript in HTML. Jezik CSS sem uporabil za dizajn strani v sklopu tega pa tudi Bulma Framework, namenjen slogu elementov na strani. Največ kode sem napisal v JavaScript jeziku, ki omogoča, da spletna stran interaktivno deluje.

Aplikacija je sestavljena iz dveh verzij Vadnika in Ocenjevalnika, ki sta ločeni spletni strani. Vadnik je namenjen reševanju fizikalnih nalog, pri čemer naj bi bilo učencu na voljo določeno število nalog, ki bi bile shranjene v naboru nalog, kamor bi jih sproti še lahko dodajali. Ocenjevalnik je namenjen pisanju testov, pri čemer bi učitelj objavil naloge na spletni strani ob začetku ure.

## 2 TEORETIČNA IZHODIŠČA

### 2.1 Pogled v zgodovino

Leta 1946 je bil ustvarjen prvi komercialni elektronski računalnik, katerega namen je bil reševanje zahtevnih matematičnih izračunov.<sup>1</sup> Do leta 1977 so bili računalniki namenjeni za znanost, službeno dejavnost in inženirstvo. Potem se je uporaba razširila na domačo rabo, saj so na trg prišli mikroročunalniki, ki so bili cenovno ugodni. Z njimi so ljudje igrali igrice, pisali besedila, programirali.<sup>2</sup> Sedaj je računalnik samoumeven, v gospodinjstvu imamo nešteto naprav, računalnike, tablice in telefone, ki so mnogo zmogljivejši od prvih računalnikov.



Replacing a bad tube meant checking among ENIAC's 19,000 possibilities.

Slika 1: ENIAC prvi računalnik



Slika 2: Osebni računalnik

Za iskanje informacij potrebujemo splet. Splet je razvil Tim Berners-Lee leta 1989, z namenom, da bi avtomatiziral izmenjavo informacij med znanstveniki različnih fakultet. Sčasoma pa se je razširil in postal svetoven. Leta 1993 je CERN dal programsko opremo za svetovni splet v javno uporabo, kar je omogočilo razcvet spleta.<sup>3</sup>

<sup>1</sup> ENIAC. Pridobljeno 12.1.2021 s <https://sl.wikipedia.org/wiki/ENIAC>

<sup>2</sup> History of personal computers. Pridobljeno 12.1.2021 s [https://en.wikipedia.org/wiki/History\\_of\\_personal\\_computers](https://en.wikipedia.org/wiki/History_of_personal_computers)

<sup>3</sup> A short history of the Web. Pridobljeno 12.1.2021 s <https://home.cern/science/computing/birth-web/short-history-we>



Slika 3: Tim Berners-Lee (1994)

## 2.2 Računalnik

Računalnik je naprava, ki s pomočjo algoritmov upravlja svoje delovanje. Računalnik je sestavljen iz strojne in programske opreme. Da pa bi vedeli kako deluje, moramo vedeti več o programski opremi. <sup>4</sup>Z računalniki lažje in hitreje opravljamo šolsko/sluzbeno delo, komuniciramo, uporabljamo jih za zabavo in učenje (obdelavo besedil, programiranje). Značilnosti računalnikov so, da vsebujejo ali shranijo veliko več informacij kakor človek, računalniki se ne utrudijo in vedno delajo.<sup>5</sup>



Slika 4: Igralni računalnik



Slika 5: Delovni računalnik

Računalnike uporabljamo za različne namene: službeno delo, izobraževanje, delo doma, znanost, prodajo itn. Osredotočil se bom na uporabo računalnika za izobraževanje v šoli/doma. <sup>6</sup>

<sup>4</sup> Computer. Pridobljeno 12.1.2021 s <https://en.wikipedia.org/wiki/Computer>

<sup>5</sup> Reason of why people use computer. Pridobljeno 12.1.2021 s <https://sites.google.com/site/liptonyunz/home/reason-of-why-people-use-computer>

<sup>6</sup> Computer Basics: 20 Examples of Computer Uses. Pridobljeno 12.1.2021 s <https://turbofuture.com/computers/7-Uses-of-Computers>

Računalnik kot pripomoček se v šoli lahko uporablja za predvajanje avdiovizualnih posnetkov, interaktivnih vaj in za učenje na daljavo. V tem času je zelo pomemben za povezovanje učenca z učitelji, vzdrževanje in spremljanje uspešnosti učenca ter njegovo ocenjevanje znanja. Obstaja veliko pripomočkov za učenje na računalniku doma in ocenjevanje znanja npr. Google Forms, Quizlet, Kahoot, Padlet itd. Njegova uporaba pa se pri različnih področjih oz. predmetih spreminja, saj poteka tam delo drugače.<sup>7</sup>

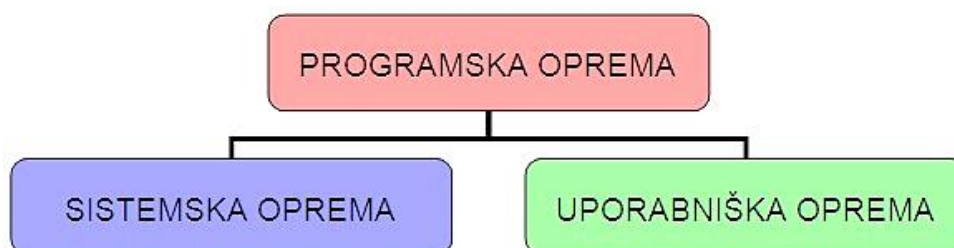
Pri pouku fizike se računalnik uporablja za analizo in vizualizacijo podatkov, sporočanje rezultatov, izvajanje preizkusov in različnih simulacij. Pomemben je tudi za učenje fizike, za kar obstaja veliko programov.<sup>8</sup>

## 2.3 Programska oprema

Je skupek programov, ki skupaj s strojno opremo računalnika sestavlja računalnik in je potrebna za delovanje računalnika, izvajanje določenih nalog itd. Programska oprema je izraz, ki se uporablja za programe, aplikacije, skripte. Ponavadi je shranjena v pomnilniku.<sup>9</sup>

Delimo jo na:

- sistemsko programsko opremo in
- uporabniško ali aplikativno programsko opremo.<sup>10</sup>



Slika 6: Delitev programske opreme

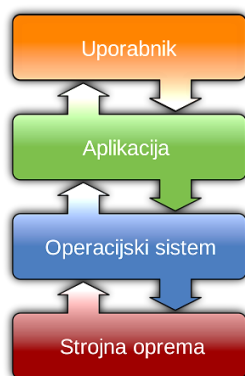
<sup>7</sup>Computer Basics: 20 Examples of Computer Uses. Pridobljeno 12.1.2021 s <https://turbofuture.com/computers/7-Uses-of-Computers>

<sup>8</sup>Role of computers in physics education – A review. Pridobljeno 14.1.2021 s <https://www.journalcra.com/article/role-computers-physics-education-%E2%80%93-review>

<sup>9</sup>Programska oprema. Pridobljeno 14.1.2021 s [https://sl.wikipedia.org/wiki/Programska\\_oprema](https://sl.wikipedia.org/wiki/Programska_oprema)

<sup>10</sup>Types of programs. Pridobljeno 14.1.2021 s

[http://www.cs.ucc.ie/~gavin/cs1001/Notes/chap01/ch01\\_9.html](http://www.cs.ucc.ie/~gavin/cs1001/Notes/chap01/ch01_9.html)



Slika 7: Diagram komunikacije med strojno opremo in uporabnikom

### 2.3.1 Sistemska programska oprema

Sistemska programska oprema je skupek programov, ki omogočajo delovanje operacijskega sistema in posledično računalnika, prav tako kot aplikacijske programske opreme itd.<sup>11</sup>

Pod sistemsko programsko opremo štejemo:

- operacijske sisteme,
- gonilnike, ki skrbijo za pravilno delovanje priključenih naprav,
- komunikacijske programe za vzpostavljanje povezav z drugimi napravami
- in sistemska orodja oz. pripomočke, ki se uporabljajo neodvisno od operacijskega sistema in so programi za izkoriščanje:
  - prostora na disku, stiskanje podatkov ali razdeljevanje pomnilnika, npr. program za delo z datotekami (Raziskovalec), programa za stiskanje datotek (WinZip, WinRar), protivirusni programi (AVG, Bitdefender, McAfee).<sup>12</sup>



Slika 8: Operacijski sistem Windows



Slika 9: Protivirusni program Bitdefender

<sup>11</sup>Sistemska programska oprema. Pridobljeno 14.1.2021 s <http://racunalnistvo2ggl.splet.arnes.si/programska-oprema/sistemska-programaska-oprema/>

<sup>12</sup>Sistemska programska oprema. Pridobljeno 14.1.2021 s [http://egradivo.ecnm.si/KIT/sistemska\\_programaska\\_oprema.html](http://egradivo.ecnm.si/KIT/sistemska_programaska_oprema.html)



### 2.3.2 Uporabniška ali aplikativna programska oprema ali aplikacija

Programi, ki spadajo pod uporabniško ali aplikativno programsko opremo, ki ji na kratko imenujemo aplikacije, so namenjeni uporabnikom, ki opravljajo različna opravila neposredno. Lahko jih imenujemo tudi programska orodja.<sup>13</sup>

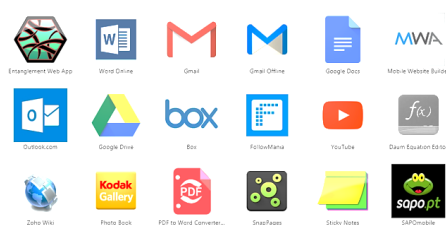
Uporabniki z njihovo pomočjo pišejo pisma, računajo, prebirajo novice, rišejo slike in kleptajo po internetu.

Poznamo več vrst uporabniških programov, katerih namen je:

- delo z besedili, preglednicami, predstavitvami, npr. MS Word, LaTeX, MS Excel, MS Powerpoint itd.,
- obdelava podatkov, npr. MS Access, Lotus, Adobe Photoshop, katerih naloga je spreminjanje, hranjenje, iskanje in urejanje podatkov,
- grafično spreminjanje, npr. MS PowerPoint, Slikar, CorelDraw, ..., s katerimi nastajajo risbe in slike,
- komunikacijsko povezovanje, npr. Internet Explorer, Windows Messenger, Mozilla Firefox itn. ki omogočajo vzpostavljanje zvez med drugimi računalniki, prejemanje, pošiljanje in pogovor med uporabniki.<sup>14</sup>

Glede na območje, kjer se aplikacije nahajajo, ločimo:

- mobilne aplikacije,
- namizne aplikacije,
- spletne aplikacije.<sup>15</sup>



Slika 10: Uporabniške/ aplikativne aplikacije

<sup>13</sup>Application software. Pridobljeno 16.1.2021 s [https://en.wikipedia.org/wiki/Application\\_software](https://en.wikipedia.org/wiki/Application_software)

<sup>14</sup>Uporabniška programska oprema. Pridobljeno 16.1.2021 s [http://egradivo.ecnm.si/KIT/uporabnika\\_programska\\_oprema.html](http://egradivo.ecnm.si/KIT/uporabnika_programska_oprema.html)

<sup>15</sup>Computer Basics-Understanding Applications. Pridobljeno 16.1.2021 s <https://edu.gcfglobal.org/en/computerbasics/understanding-applications/1/>

**Mobilna aplikacija** je programska aplikacija, ki deluje na mobilni napravi, na telefonu, tablici ali uri. Lahko jo prenesemo s katere koli platforme nekega operacijskega sistema. Med poznane štejemo App Store (iOS) in Google Play Store. Sprva so bile aplikacije namenjene podpori produktivnosti kot so e-mail, koledar itd. Sčasoma pa je javno povpraševanje povzročilo hitro širitev tudi na druga področja, kot so tovarniške avtomatizacije in mobilne igre.<sup>16</sup>



Slika 11: Mobilne aplikacije

**Namizna aplikacija** je aplikacija, ki se nahaja na računalniku ali prenosniku in deluje lokalno na strežniku naprave. Glavni značilnosti namizne aplikacije sta visoka učinkovitost in prilagodljivost glede na zahteve uporabnika. Te aplikacije se lahko prenese s spletnih mest ali pa prek optičnih pogonov.<sup>17</sup>



Slika 12: Namizne aplikacije

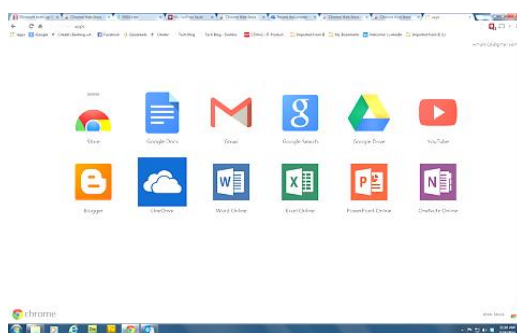
**Spletna aplikacija** je aplikacija, spletna stran ali skupek spletnih strani, ki deluje na spletnem strežniku. Do nje dostopamo preko brskalnika z internetno povezavo, ki je običajno napisan v CSS, HTML5 ali JavaScript. Pri uporabi brskalnika se odpre stran s povezavami. Uporabnik

<sup>16</sup>Mobile app. Pridobljeno 16.1.2021 s [https://en.wikipedia.org/wiki/Mobile\\_app](https://en.wikipedia.org/wiki/Mobile_app)

<sup>17</sup>What is Desktop Application. Pridobljeno 16.1.2021 s <https://iphotechnologies.com/what-is-desktop-application/>

tam izbere in klikne na zelen zadetek, kar ga preusmeri na določen URL, kjer se pokaže spletna stran. Dobra stran prenašanja spletnih aplikacij je to, da se jih shrani že s klikom na zaznamek in da zasedajo majhno količino pomnilnika. Slaba stran pa je, da vedno zahtevajo dobro internetno povezavo za njihovo delovanje, kar je tudi glavna pomanjkljivost spletnih aplikacij.

Za natančnejše razumevanje moramo vedeti, kaj je program, poznati vrste programov, vedeti kaj je spletna stran, brskalnik, strežnik in URL.<sup>18</sup>



Slika 13: Spletne aplikacije

## Spletna stran

je dokument z nadbesedilom, ki se prikaže v spletnem brskalniku. Na njem so lahko različne vsebine: besedilo, slike, videoposnetki in povezave. Napisan je v jeziku HTML, prav tako je lahko napisan v PHP, JavaScript, JQuery, CSS ali drugih. Več povezanih spletnih strani tvori spletno mesto, kot je npr. Wikipedia.org. Vsa javna spletna mesta sestavljajo splet, zasebna pa se nahajajo v zasebnem omrežju.<sup>19</sup>

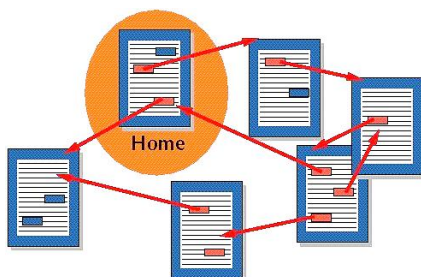
## Nadbessedilo ali hipertekst

je način označevanja besedila ali drugih grafičnih elementov, ki vsebuje povezave do drugih informacij, do katerih lahko uporabnik dostopa že s samo klikom miške.<sup>20</sup>

<sup>18</sup>Web application. Pridobljeno 16.1.2021 s [https://en.wikipedia.org/wiki/Web\\_application](https://en.wikipedia.org/wiki/Web_application)

<sup>19</sup>Spletna stran. Pridobljeno 17.1.2021 s [https://sl.wikipedia.org/wiki/Spletna\\_stran](https://sl.wikipedia.org/wiki/Spletna_stran)

<sup>20</sup>Hipertekst. Pridobljeno 17.1.2021 s <https://sl.wikipedia.org/wiki/Hipertekst>



Slika 16: Hipertekst

## Brskalnik

je program, ki prikazuje spletne strani in jih pregleduje. Z njimi se povezuje preko njihovih strežnikov, z njim dostopamo do informacij iz spletnega mesta. Spletne brskalnike ločimo glede na njihovo hitrost, zasebnost in prilagajanje.<sup>21</sup> Med bolj priljubljene brskalnike sodijo Google Chrome, Mozilla Firefox in Apple Safari.<sup>22</sup>



Slika 17: Brskalnik Google Chrome

## Strežnik

je izraz, ki ima v računalništvu več pomenov: to je program, ki daje na voljo storitve drugim programom ali računalnik v omrežju, ki daje svoje podatke na voljo drugim računalnikom. Poznamo več vrst strežnikov, pri tej raziskovalni nalogi pa bomo govorili le o spletnem strežniku.<sup>23</sup>

## Spletni strežnik

je strežnik, katerega naloga je shranjevanje, obdelava in pošiljanje spletnih strani odjemalcem. Komunikacija med njima poteka s protokolom prenosa za hipertekst, ki se uporablja za razširjanje informacij na svetovnem spletu in se imenuje odjemalec/strežnik.<sup>24</sup>

<sup>21</sup>Web browser. Pridobljeno 17.1.2021 s [https://en.wikipedia.org/wiki/Web\\_browser](https://en.wikipedia.org/wiki/Web_browser)

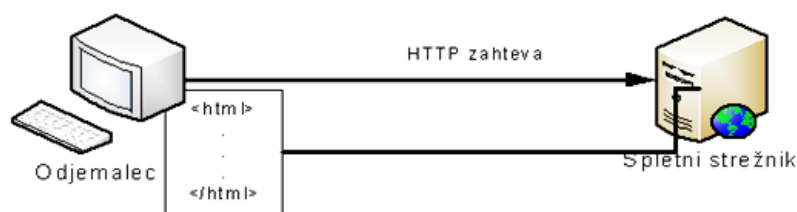
<sup>22</sup>The best browser 2021. Pridobljeno 17.1.2021 s <https://www.techradar.com/best/browser>

<sup>23</sup>Strežnik. Pridobljeno 17.1.2021 s <https://sl.wikipedia.org/wiki/Stre%C5%BEnik>

<sup>24</sup>Spletni strežnik. Pridobljeno 17.1.2021 s [https://sl.wikipedia.org/wiki/Spletni\\_stre%C5%BEnik](https://sl.wikipedia.org/wiki/Spletni_stre%C5%BEnik)

## Povezava odjemalec/strežnik

Povezava poteka med odjemalcem in strežnikom. Odjemalec je vsak računalnik, ki je na daljavo povezan z gostiteljskim računalnikom, ki mu zagotavlja storitev. Gostiteljski računalnik pa je strežnik, ki poganja programe za opravljanje storitev v omrežju. Neko podjetje lahko uporablja programe s pomočjo drugih strežnikov na internetu ali pa ustvari svojega, s katerim drugi uporabniki preko njega dobijo informacije ali poslujejo.<sup>25</sup>



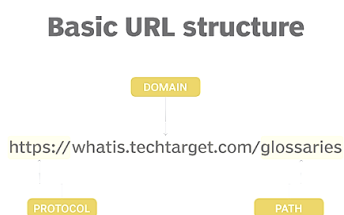
Slika 18: Povezava odjemalec/strežnik

## URL

je naslov spletnih strani v svetovnem spletu. Je zaporedje črk in znakov, ki so v naslovni vrstici brskalnika. Med seboj se razlikujejo, saj poimenujejo različne storitve in kategorije.<sup>26</sup>

Ločimo jih na dinamične in statične:

- dinamične so tiste, kjer domeni sledijo skupine sekcij, kategorij,
- statične pa so tiste, kjer je vidna drevesna struktura map in podmap.<sup>27</sup>



Slika 19: Statičen URL



Slika 20: URL

<sup>25</sup>Odjemalec/strežnik. Pridobljeno 17.1.2021 s

<http://www2.arnes.si/~sspmpese/i/Informatika2/Prispevki/Borko2aOdjemalecStreznik.htm>

<sup>26</sup>URL. Pridobljeno 17.1.2021 s <https://sl.wikipedia.org/wiki/URL>

<sup>27</sup>Kaj je URL naslov in kako ga pravilno uporabiti pri izdelavi spletne strani. Pridobljeno 17.1.2021 s <https://spletnik.si/blog/kaj-je-url-naslov-in-kako-ga-pravilno-uporabiti-pri-izdelavi-spletne-strani/>

## 2.4 Kaj je program?

Je zaporedje ukazov v strojni kodi, v obliki, ki jo računalnik lahko izvede.<sup>28</sup>

Algoritmi (set navodil) so razvrščeni po vrsti in tečejo drug za drugim. Z njegovim delovanjem ljudje izvedejo različna opravila oziroma komande.<sup>29</sup>

Računalniški programi pomagajo nadzorovati in organizirati delovanje naprav v današnjem svetu, npr. semaforjev, pralnih strojev, telefonov itd.

Program je del programske opreme. Ustvari ga programer v programskem jeziku. Ko programerji napišejo kodo, se človeku berljiva koda prevede v strojno kodo, ki pa jo lahko računalnik izvede neposredno.<sup>30</sup>

Obstajata dve vrsti programov - interaktivni in serijski.<sup>31,32</sup>

### Interaktivni programi

so programi, ki sledijo navodilom in potrebujejo za delovanje interakcijo uporabnika, pri kateri lahko uporabnik vnaša podatke, upravlja/spreminja informacije ali opravlja kaj drugega, npr. spletni brskalnik, word, excel.<sup>33</sup>



Slika 14: Google Chrome



Slika 15: Microsoft Word

### Serijski program

delujejo na zahtevo interaktivnega programa in ga človek ne more upravljati, npr. program, ki računa in nato natisne račune podjetij.<sup>34</sup>

<sup>28</sup> Vloga programa in programskega jezika. Pridobljeno 11.5.2021 s [http://colos.fri.uni-lj.si/eri/INFORMATIKA/PROGRAMIRANJE/vloga\\_programa\\_prog\\_jezika.html](http://colos.fri.uni-lj.si/eri/INFORMATIKA/PROGRAMIRANJE/vloga_programa_prog_jezika.html)

<sup>29</sup> Algorithm. Pridobljeno 12.1.2021 s <https://techterms.com/definition/algorithm>

<sup>30</sup> Vloga programa in programskega jezika. Pridobljeno 11.5.2021 s [http://colos.fri.uni-lj.si/eri/INFORMATIKA/PROGRAMIRANJE/vloga\\_programa\\_prog\\_jezika.html](http://colos.fri.uni-lj.si/eri/INFORMATIKA/PROGRAMIRANJE/vloga_programa_prog_jezika.html)

<sup>31</sup> Interactive program? Pridobljeno 12.1.2021 s <https://www.computerhope.com/jargon/i/inteprog.htm>

<sup>32</sup> What is a computer Program and How It Work? Pridobljeno 12.1.2021 s <https://www.youtube.com/watch?v=IP5s13qEB4A>

<sup>33</sup> Interactive program. Pridobljeno 12.1.2021 s <https://www.computerhope.com/jargon/i/inteprog.htm>

<sup>34</sup> What is a computer Program and How It Work? Pridobljeno 12.1.2021 s <https://www.youtube.com/watch?v=IP5s13qEB4A>

## 2.5 Programiranje

Programiranje je dejavnost pri kateri programer zapisuje kodo oz. programira.

Programiranje je ustvarjalen proces, pri katerem ni navodil. S programiranjem se ustvarjajo računalniški programi, rešujejo težave ali se izvajajo naloge.

Pri tem uporabljajo različne računalniške jezike: Java, C, HTML, C++, Python, JavaScript.

Končni cilj programiranja je stvaritev spletne strani, programske opreme ali lepe slike. Da pa bi lahko programsko opremo razvili, moramo pred tem opraviti številne faze: specifikacija ali opis problema, analiza problema, načrtovanje algoritma, kodiranje in preskušanje ter vzdrževanje programa.<sup>35</sup>



Slika 21: JavaScript jezik



Slika 22: CSS jezik



Slika 23: Programer/programiranje

### 2.5.1 Programski jezik

Programski jezik je umetni jezik, ki je razvit z namenom, da ga razume neka naprava. Sprva so bili narejeni zaradi razvoja računalnikov in so jim bili podrejeni.

Programskih jezikov je čez 500, vendar jih je večina zasnovana na isti osnovi.<sup>36</sup>

Poznamo več vrst programskih jezikov, ki so se skozi čas spreminjali in jih je mogoče razvrstiti po moči osnovnih operacij, namenu uporabe, naravi osnovnih objektov in drugih kriterijih.

<sup>35</sup>Računalniško programiranje. Pridobljeno 22.1.2021 s [https://sl.wikipedia.org/wiki/Ra%C4%8Dunalni%C5%A1ko\\_programiranje](https://sl.wikipedia.org/wiki/Ra%C4%8Dunalni%C5%A1ko_programiranje)

<sup>36</sup>Programski jeziki. Pridobljeno 22.1.2021 s <https://osnoveprogramiranja.weebly.com/programski-jeziki.html>

Poznamo nižje in višje programske jezike.

## Nižji programski jeziki

Delimo jih na strojni in zbirni jezik.

- Strojni jezik je najbližje računalniku. Sestavljen je iz nizov kombinacij binarnih simbolov končne dolžine, ki predstavljajo ukaze nekega procesorja.<sup>37</sup> Odvisen je od vrste procesorja, delo v njem je zamudno in nepregledno. Programiranje je zelo zahtevno, obstaja velika možnost napak, ki se jih tudi težko odkrije. Uporabljenih je zelo veliko binarnih števil za opis enostavne funkcionalnosti. Programi niso prenosljivi med računalniki z različno strojno opremo.<sup>38</sup>

```
11101000 10110010 00000110 00000000 00000000 01011001 01101000 10100000
10110000 01000000 00000000 01101010 00000000 11101000 11000010 10010111
00000000 00000000 10100011 11011111 10110000 01000000 00000000 01101010
00000000 11101001 10001010 10010101 00000000 00000000 11101001 00101111
00000111 00000000 00000000 00110011 11000000 10100000 11001100 10110000
01000000 00000000 11000011 10100001 11011111 10110000 01000000 00000000
11000011 11001100 10111001 10110000 00000000 00000000 00000000 00001011
11001001 01110100 00111001 10000011 00111101 11010111 10110000 01000000
00000000 00000000 01110011 00001010 10111000 11100010 00000000 00000000
00000000 11101000 11100011 11111111 11111111 11111111 01101000 10110000
00000000 00000000 00000000 01101010 01000000 11101000 10110110 10010111
00000000 00000000 00001011 11000000 01110101 00001010 10111000 11100010
00000000 00000000 00000000 11101000 11001001 11111111 11111111 11111111
01010000 11111111 00110101 11010111 10110000 01000000 00000000 11101000
11100100 10010111 00000000 00000000 11000011 10111001 10110000 00000000
00000000 00000000 00001011 11001001 01110100 00011001 11101000 11000011
10010111 00000000 00000000 10100011 11010111 10110000 01000000 00000000
10000011 11111000 00000000 01110011 10100101 10111000 11100010 00000000
00000000 00000000 11101000 10011010 11111111 11111111 11111111 11000011
10000011 00111101 11010111 10110000 01000000 00000000 00000000 01110010
00010101 11111111 00110101 11010111 10110000 01000000 00000000 11101000
10100110 10010111 00000000 00000000 00001011 11000000 01110100 00000110
01010000 11101000 01100000 10010111 00000000 00000000 11000011 10000011
```

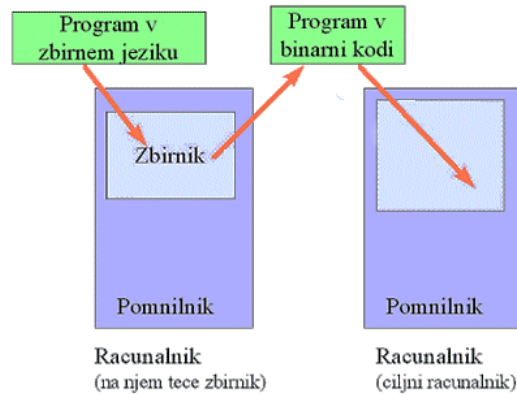
Slika 24: Strojni jezik

- Zbirni jezik izhaja iz strojnega jezika. Pri zapisu zbirnega jezika so si pomagali z opisi ukazov, kar je bilo enostavnejše kakor z uporabo binarnih ukazov. Te opise so imenovali mnemoniki. Vsak ukaz so nadomestili z enim mnemonikom. Problem pa je bil, da programa, zapisanega v zbirnem jeziku, procesor neposredno ni razumel in ga je bilo potrebno pretvoriti v strojni jezik. To so naredili z dodatnim programom, ki so ga imenovali zbirnik.

<sup>37</sup>Delitev programskih jezikov. Pridobljeno 22.1.2021 s [http://colos.fri.uni-lj.si/eri/RACUNALNISTVO/PROG\\_JEZIKI\\_ORODJA/O2DelitevProgJezikov/O2\\_delitevProgramskihJezikov.html](http://colos.fri.uni-lj.si/eri/RACUNALNISTVO/PROG_JEZIKI_ORODJA/O2DelitevProgJezikov/O2_delitevProgramskihJezikov.html)

<sup>38</sup>Gradivo: programski jeziki-predstavitev. Pridobljeno 23.1.2021 s [https://dijaski.net/gradivo/rif\\_ref\\_programski\\_jeziki\\_01\\_\\_predstavitev](https://dijaski.net/gradivo/rif_ref_programski_jeziki_01__predstavitev)





Slika 25: Zbirni jezik

- Nadaljevanje zbirnika so bili makroji. Makroji so bili grupe povprečno petih ukazov, ki so jih programerji v kodi večkrat ponovno rabili in so jih združili v en močnejši ukaz. Njihov namen je bil lajšanje dela programerju, končna koda z njihovo uporabo pa ni bila nič krajša od zbirne.<sup>39</sup>

```

3  MACRO ALIGN =x
4  IF x > 1
5  IF (x & 1) = 1
6  ERROR odd parameter to ALIGN
7  ELSE
8  ALIGN (x>>1)
9  . = .+(ABS(.) & (x>>1))
10 ENDIF
11 ENDIF
12 ENDMAC
13
14 ALIGN 1; no operation
15 B 1
16 ALIGN 2; align to a halfword boundary
17 H 2
18 ALIGN 4; align to a word boundary
19 W 3
20 ALIGN 3; an improper alignment
20 ERROR odd parameter to ALIGN

```

Definicija makroja

Uporaba makroja

Slika 26: Makro zbirni jezik

## Višji programski jeziki

So uporabniško usmerjeni jeziki. Večina jih ima prevajalnik, npr. C++, nekateri tolmača in nekateri oboje, npr. Java, Python.<sup>40</sup>

Delimo jih:

- Glede na strukturo prevedenega programa v primerjavi s strukturo izvirnega programa: strukturirani in nestrukturirani.

<sup>39</sup>Delitev programskih jezikov. Pridobljeno 23.1.2021 s [http://colos.fri.uni-lj.si/eri/RACUNALNISTVO/PROG\\_JEZIKI\\_ORODJA/02DelitevProgJezikov/02\\_delitevProgramskihJezikov.html](http://colos.fri.uni-lj.si/eri/RACUNALNISTVO/PROG_JEZIKI_ORODJA/02DelitevProgJezikov/02_delitevProgramskihJezikov.html)

<sup>40</sup>Gradivo: programski jeziki-predstavitev. Pridobljeno 23.1.2021 s [https://dijaski.net/gradivo/rif\\_ref\\_programski\\_jeziki\\_01\\_\\_predstavitev](https://dijaski.net/gradivo/rif_ref_programski_jeziki_01__predstavitev)

- Za strukturiran program je značilno, da po procesu prevajanja struktura programa ostane enaka. Strukturni programski jezik programerju omogoča, da razdeli celoten program na manjše enote. Strukturirano programiranje ni primerno za razvoj velikih programov npr. C, Python...



Slika 27: C jezik



Slika 28: Python jezik

- Program v nestrukturiranem jeziku je ponavadi sestavljen iz zaporedno razvrščenih ukazov, ki omogočajo preskok na katero koli vrstico v programu. Programiranje v tem jeziku omogoča številne zanke in skoke, kar omogoča uresničevanje podprogramov, npr. Basic.

```

READY
10 FOR X=1 TO 10
20 PRINT "HOLA WIKIPEDIA"
30 NEXT X
RUN
HOLA WIKIPEDIA
HOLA WIKIPEDIA
HOLA WIKIPEDIA
HOLA WIKIPEDIA
HOLA WIKIPEDIA
HOLA WIKIPEDIA
HOLA WIKIPEDIA
HOLA WIKIPEDIA
HOLA WIKIPEDIA
HOLA WIKIPEDIA
READY

```

Slika 29: Program v nestrukturiranem jeziku

- Glede na to, kaj pri kodiranju programiramo: postopkovni in deklarativni.
  - Postopkovni programski jezik je jezik, pri katerem opisujemo pot, po kateri se pride do rešitve. Izdelan je v obliki recepta, npr. Pascal.

```

static char* get_boundary(apr_pool_t* p, const char* ctype) {
char* ret = NULL;
if ( ctype ) {
char* lctype = lccopy(p, ctype);
char* bdy = strstr(lctype, "boundary");
if ( bdy ) {
char* ptr = strchr(bdy, '=');
if ( ptr ) {
bdy = (char*) ctype + ( ptr - lctype ) + 1;
for ( ptr = bdy; *ptr; ++ptr )
if ( *ptr == ';' || isspace(*ptr) )
*ptr = 0;
ret = apr_pstrdup(p, bdy);
}
}
}
return ret;
}

```

Slika 30: Postopkovni programski jezik

- Deklerativni jezik pa je jezik, pri katerem programiranje sloni na zastavljanju vprašanj in opredelitvi pravil in dejstev, npr. Prolog in Lisp.

```
Prolog in Elixir
?- assert(append([], Xs, Xs)).
true
?- assert((append([X | Ls], Ys, [X | Zs]) :- append(Ls, Ys, Zs))).
true
?- append([1,2],[a,b],X).
X = [1,2,a,b]
true
?- append(X,Y,[1,2,3]).
X = []
Y = [1,2,3];
X = [1]
Y = [2,3];
X = [1,2]
Y = [3];
X = [1,2,3]
Y = [];
false
?-
```

Slika 31: Deklerativni jezik

- Glede na to, kaj je osnovni element programa: proceduralni in modularni/objektivni.
  - Proceduralni jezik je jezik, ki zaporedoma sledi naboru ukazov, npr. C, Pascal, Basic.



Slika 32: Proceduralni jezik C

- Za modularni jezik je značilno, da je osnovni element modul. Pri tem programiranju gre za poudarjanje ločevanja funkcionalnosti programa na neodvisne in zamenljive module. Vsak modul tako vsebuje vse potrebno za izvajanje ene funkcionalnosti, npr. Java, C++.<sup>41</sup>



Slika 33: Modularni jezik C++

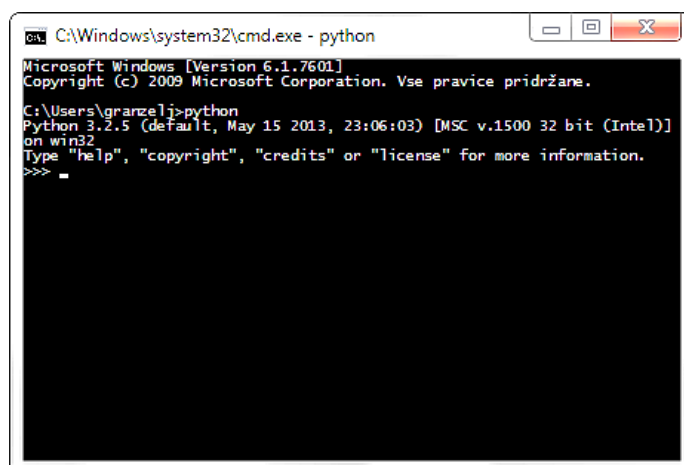
<sup>41</sup>Delitev programskih jezikov. Pridobljeno 24.1.2021 s [http://colos.fri.uni-lj.si/eri/RACUNALNISTVO/PROG\\_JEZIKI\\_ORODJA/02DelitevProgJezikov/02\\_delitevProgramskihJezikov.html](http://colos.fri.uni-lj.si/eri/RACUNALNISTVO/PROG_JEZIKI_ORODJA/02DelitevProgJezikov/02_delitevProgramskihJezikov.html)

## Prevajalnik

Prevajalnik je program, ki prevede izvorno kodo napisano v višjem programskem jeziku v strojni jezik. Rezultat je izvedljiv program, ki ga računalnik razume in ga zna prevesti.

## Tolmač

Je program, ki izvede izvorni program in običajno poteka počasi. Stavke izvornega programa prevede v strojni jezik in ukaze posreduje procesorju. Prednost je interaktivnost ali odzivnost.<sup>42</sup>



```
C:\Windows\system32\cmd.exe - python
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. Vse pravice pridržane.

C:\Users\granzelj>python
Python 3.2.5 (default, May 15 2013, 23:06:03) [MSC v.1500 32 bit (Intel)]
on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> =
```

Slika 34: Tolmač

---

<sup>42</sup>Algoritmi navodila/jeziki. Pridobljeno 24.1.2021 s [http://erid.tsckr.si/4/algoritmi\\_navodila/jeziki.html](http://erid.tsckr.si/4/algoritmi_navodila/jeziki.html)

## 2.6 Kaj vse sem uporabil pri tej raziskovalni nalogi

Pod programskimi jeziki bom opisal tri jezike, ki sem jih uporabil pri tej raziskovalni nalogi.

### HTML

Je označevalni jezik za objavlanje na svetovnem spletu. Preko njega se izdelujejo spletne strani, lahko se ga piše tudi v preprostem urejevalniku besedil. Za pisanje se ne potrebuje matematičnega znanja, saj ni enak drugim programskim jezikom. Sestavljen je iz HTML elementov, ki brskalniku sporočajo kako naj prikaže vsebino spletne strani. Uporabnik lahko oblikuje ali organizira dokumente tako kot v Microsoft Wordu.<sup>43</sup>

Primer HTML:



Slika 35: Primer HTML dokumenta

### CSS

Css je preprost oblikovalni jezik, namenjen slogu besedila, slik, naslovov itd., ki so v HTML. Z njim lahko določamo barvo, velikost, odmike, poravnave in vrsto drugih atributov za vsak element v HTML. Ustvarjalci tako ločijo vsebino spletnih strani od dizajna. Učenje tega jezika je enostavno in razumljivo, tako kot HTML.<sup>44</sup>

<sup>43</sup>XHTML. Pridobljeno 25.1.2021 s <http://www.s-sers.mb.edus.si/gradiva/w3/html/index.html>

<sup>44</sup>CSS. Pridobljeno 25.1.2021 s <http://zaversnik.fmf.uni-lj.si/Gradiva/css/index.php>

Primer CSS:

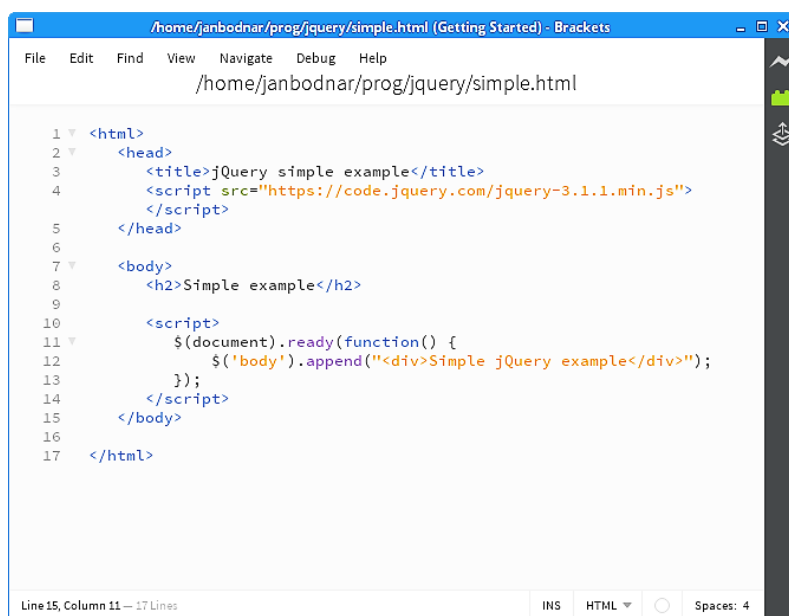
```
h1 {
  font-family: courier, courier-new, serif;
  font-size: 20pt;
  color: blue;
  border-bottom: 2px solid blue;
}
p {
  font-family: arial, verdana, sans-serif;
  font-size: 12pt;
  color: #6B6BD7;
}
.red_txt {
  color: red;
}
```

Slika 36: Primer CSS dokumenta

## JavaScript

Je objektni skriptni programski jezik, ki omogoča ustvarjanje interaktivne spletne strani v povezavi s HTML kodo od katere je odvisen. Omogoča odpiranje novih oken, preverjanje pravilnosti vnesenih podatkov in enostavne izračune. Za svoje delovanje mora sodelovati s predmeti na uporabniški strani in njihovimi lastnostmi ter postopki. Razlika med JavaScriptom in Javo je v tem, da je Java zmoglivejši in kompleksnejši jezik.<sup>45</sup>

Primer JavaScript:



```
1 <html>
2   <head>
3     <title>jQuery simple example</title>
4     <script src="https://code.jquery.com/jquery-3.1.1.min.js">
5     </script>
6   </head>
7   <body>
8     <h2>Simple example</h2>
9
10    <script>
11      $(document).ready(function() {
12        $('body').append("<div>Simple jQuery example</div>");
13      });
14    </script>
15  </body>
16
17 </html>
```

Slika 37: Primer JavaScript dokumenta

<sup>45</sup>JavaScript. Pridobljeno 24.1.2021 s <https://sl.wikipedia.org/wiki/JavaScript>

## Okvir CSS

Je orodje ali zbirka tabel stilov CSS, ki so pripravljene za uporabo. Z njimi uporabnik lažje oblikuje dizajn. Vsebuje veliko kode, ki bi jo bilo treba pisati večkrat. Poznamo več različnih odprtokodnih CSS okvirjev, ki pa se razlikujejo v dizajnu, npr. Bulma, Bootstrap. Pri uporabi uporabnik poleg njihovega osnovnega dizajna še sam oblikuje različne attribute.<sup>46</sup>



Slika 38: Okvir CSS – Bootstrap

## Bulma

Bulma je eden od brezplačnih odprtokodnih CSS okvirjev, ki temelji na flexboxu (enodimenzionalni sistem postavitve, ki nam lajša izdelavo in načrtovanje spletnih strani, da ni treba pisati zapletene kode v CSS).<sup>47</sup>



Slika 39: Bulma CSS okvir

## Github

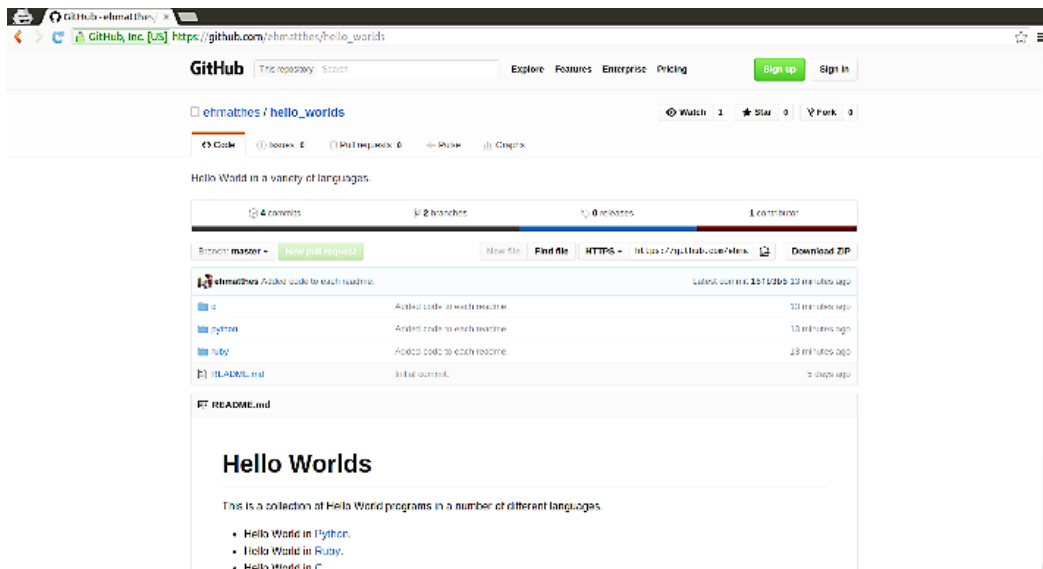
Pri raziskovalni nalogi, sem si kodo med izdelavo programa shranjeval na platformi Github. Github je spletna stran oz. platforma za gostovanje kod za nadzor različic in sodelovanje. Če želimo razumeti GitHub, moramo najprej razumeti git. Git je odprtokodni sistem za nadzor različic, podoben je drugim sistemom npr. Subversion, CVS, Mercurial. Različica je

---

<sup>46</sup>What is CSS Framework. Pridobljeno 24.1.2021 s <https://medium.com/html-all-the-things/what-is-a-css-framework-f758ef0b1a11>

<sup>47</sup>What is bulma? Advantages & disadvantages of using Bulma. Pridobljeno 24.1.2021 s <https://mobiosolutions.com/what-is-bulma-advantages-disadvantages-of-using-bulma-2/>

aplikacija, pri kateri ustvarjalci nenehno spreminjajo kodo. Koda se shranjuje v centralnem repozitoriju, kjer vsak razvijalec lahko vidi spremembe in prispeva.<sup>48</sup>



Slika 40: Spletna stran Github

<sup>48</sup>What Is GitHub, and What Is It Used For? Pridobljeno 24.1.2021 s <https://www.howtogeek.com/180167/htg-explains-what-is-github-and-what-do-geeks-use-it-for/>



## **3 RAZISKOVALNI DEL**

### **3.1 Cilji raziskovalne naloge**

Cilj raziskovalne naloge je bil sestaviti generator nalog iz nalog iz fizike.

Generator nalog iz nalog iz fizike bo programska oprema, ki se bo uporabljala za ustvarjanje testov in nalog za preverjanja znanja in vajo.

Odločil sem se, da bom naredil generator testov in nalog za pouk fizike in da bo spletna aplikacija lahko vsem na voljo.

Program bo imel dve različici, Vadnik in Ocenjevalnik.

Obe verziji bosta po izgledu različni in bosta samostojni spletni strani, saj bo Vadnik namenjen utrjevanju snovi, Ocenjevalec pa ocenjevanju snovi.

### **3.2 Postavitev hipotez**

Na podlagi zgornjih ciljev sem postavil 3 hipoteze.

1. Za programiranje aplikacije bom potreboval veliko časa.
2. Aplikacijo bodo lahko uporabljali učenci samostojno.
3. Aplikacija bo uporabna tudi pri pouku (npr. ocenjevanje na daljavo).

### **3.3 Raziskava**

Raziskovalna metoda dela je analiza virov in programiranje.

## 4 REZULTATI - GENERATOR TESTOV IN NALOG IZ FIZIKE

Generator testov iz nalog iz fizike je programska oprema, ki se uporablja za ustvarjanje testov in nalog za preverjanja znanja in vajo.

Odločil sem se, da bom naredil generator testov in nalog za pouk fizike in da bo spletna aplikacija lahko vsem na voljo. Program ima dve različici, Vadnik in Ocenjevalec.

Obe verziji sta si po izgledu različni in sta samostojni spletni strani, saj je Vadnik namenjen utrjevanju snovi, Ocenjevalec pa ocenjevanju snovi.

Generator je sestavljen iz treh različnih programskih jezikov, ki pa imajo drugačne naloge:

HTML omogoča delovanje spletne strani, Javascript poskrbi delovanje programa, CSS pa uredi dizajn.

Generator testov in nalog iz fizike je objavljen na mojem Github računu:

<https://matejlajavec.github.io/>

### 4.1 Delovanje generatorja

Ker se Vadnik in Ocenjevalec rahlo razlikujeta, opišemo delovanje vsakega posebej.

### 4.1.1 Delovanje Vadnika

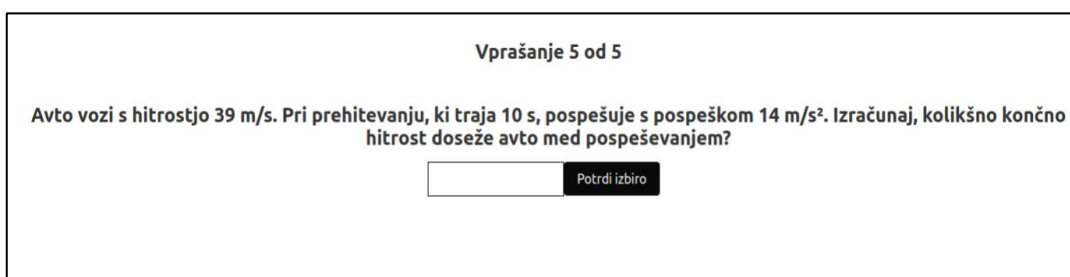
Vadnik se zažene, ko učenec odpre spletno stran. Ponudi se mu možnost za izbiro števila nalog. Nato se prikažejo naloge, ki jih učenec številsko reši. V polja vpiše zgolj številsko vrednost odgovora in ne enot. Po končanih vajah se pokažejo odstotki pravilno rešenih nalog. Če pa učenec želi še podatke katere naloge je pravilno rešil, lahko vpiše e-mail naslov, na katerega bodo poslani rezultati njegovega dela.



**Pozdravljen/a, vnesi število nalog, ki bi jih danes želel/a rešiti!**

Vse odgovore zaokroži na dve decimalni mesti in zapiši le številsko vrednost odgovora v osnovnih enotah.

Slika 41: Prva stran Vadnika



**Vprašanje 5 od 5**

Avto vozi s hitrostjo 39 m/s. Pri prehitevanju, ki traja 10 s, pospešuje s pospeškom 14 m/s<sup>2</sup>. Izračunaj, kolikšno končno hitrost doseže avto med pospeševanjem?

Slika 42: Spletna stran nalog


Test končan

**Dobil si 80%**

Če želiš rezultat in svoje rešitve poslati na elektronski naslov, ga vnesi spodaj:

Pošlji mail

Slika 43: Zadnja stran Vadnika

 **ustvarjalecنالog@gmail.com** <ustvarjalecنالog@gmail.com>  
Za jaz ▾

21.2.2021 ob 20:22 si pri preverjanju znanja pravilno rešil/a **80%** nalog.

Rešen test:

1. Avto vozi s hitrostjo 78 m/s. Pri prehitevanju, ki traja 81 s, pospešuje s pospeškom 92 m/s<sup>2</sup>. Izračunaj, kolikšno končno hitrost doseže avto med pospeševanjem?  
Odgovor: 7530 **Pravilno!**
2. Raketa se po startu giblje s pospeškom 12 m/s<sup>2</sup>. Kolikšno hitrost ima po 88 s gibanja? Izrazi v km/h.  
Odgovor: 3801.6 **Pravilno!**
3. Tovornjak vozi s hitrostjo 6 km/h. V nekem trenutku začne enakomerno pospešeno zavirati. Po 18 m se ustavi. Koliko časa se ustavlja?  
Odgovor: 21.6 **Pravilno!**
4. Telo začne pri hitrosti 34 m/s enakomerno pospeševati do hitrosti 88 m/s. Pospeševanje traja 97 s. Izračunaj povprečno hitrost.  
Odgovor: 1.26 **Napačno.**
5. Povprečna hitrost opazovanega telesa je enaka 67 m/s. Kolikšna je končna hitrost telesa, če je bila začetna enaka nič?  
Odgovor: 134 **Pravilno!**

Lep pozdrav, Ustvarjalec nalog.

Slika 44: Avtomatsko generiran mail

#### 4.1.2 Delovanje Ocenjevalnika

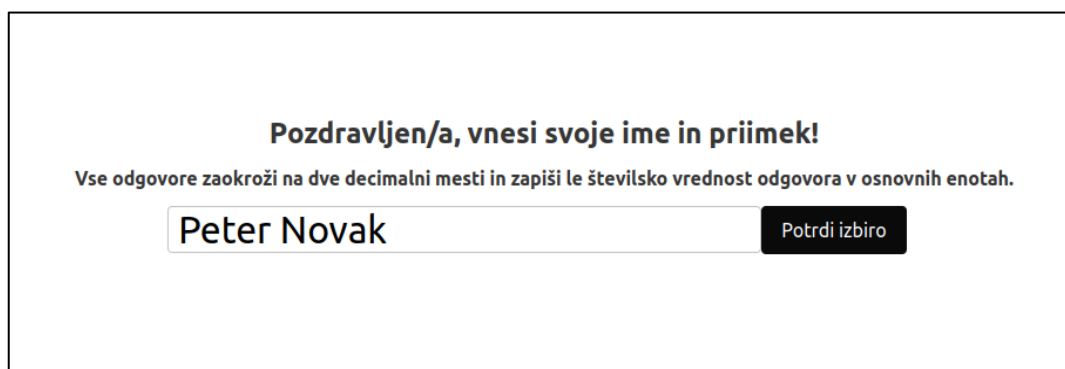
Ocenjevalnik se zažene podobno kot Vadnik.

Pojavi se polje, namenjeno vpisu imena in priimka. Nato se pokažejo naloge, ki se jih rešuje tako kakor v Vadniku, spet se zapisujejo samo številske vrednosti odgovorov.

Naloge v Ocenjevalnik naloži učitelj oz. tisti, ki test sestavi. Na koncu se prikaže rezultat in polje namenjeno vpisu e-mail naslova.

S klikom na polje »Potrdi izbiro« pošljemo avtomatsko generirano elektronsko sporočilo z osnovnimi podatki našega testa, rezultatom in rešitvami po nalogah. Zadnje je še posebej uporabno, če je učenec nalogo izračunal prav, potem pa morda narobe zaokrožil ali pa uporabil namesto decimalne vejice piko. V obeh primerih bi namreč program rezultat zaznal kot napačen.

Ocenjevalnik je program, ki ponuja nekakšno rešitev za bolj pošteno ocenjevanje v času šolanja na daljavo, saj vsi učenci rešujejo iste naloge, ki jih nanj naloži učitelj, vendar imajo vsi različne podatke in tako tudi rezultate. To seveda zmanjša možnosti prepisovanja.



**Pozdravljen/a, vnesi svoje ime in priimek!**

Vse odgovore zaokroži na dve decimalni mesti in zapiši le številsko vrednost odgovora v osnovnih enotah.

Peter Novak

Slika 45: Prva stran Ocenjevalnika

**Vprašanje 5 od 5**

**Tovornjak vozi s hitrostjo 12 km/h. V nekem trenutku začne enakomerno pospešeno zavirati. Po 16 m se ustavi. Koliko časa se ustavlja?**


Slika 46: Spletna stran nalog

**Test končan**

**Dobil si 100%**

**Če želiš rezultat in tvoje rešitve poslati na elektronski naslov, ga vnesi spodaj:**

Slika 47: Zadnja stran Ocenjevalnika


**ustvarjalecnalog@gmail.com** <ustvarjalecnalog@gmail.com>
21. feb. 2021 21:06 (pred 12 urami) ☆

Za Jaz ▾

21.1.2021 ob 21:06 je Peter Novak pri preverjanju znanja pravilno rešil/a **100%** nalog.

Rešen test:

- Voznik avtobusa opazi, da se bo na semaforju prižgala rdeča luč. Zavirati začne pri hitrosti 85 km/h in zavira 64 s preden se ustavi. Izračunaj velikost pospeška.  
Odgovor: 0.37 **Pravilno!**
- Raketa se po startu giblje s pospeškom 81 m/s<sup>2</sup>. Kolikšno hitrost ima po 20 s gibanja? Izrazi v km/h.  
Odgovor: 5832 **Pravilno!**
- Avtomobil s hitrostjo 83 m/s zaleti v oviro in se ustavi. Trk z oviro traja 71 s. Predpostavimo, da je to gibanje enakomerno pospešeno. S kolikšnim negativnim pospeškom se avto ustavlja?  
Odgovor: -1.17 **Pravilno!**
- Telo začne pri hitrosti 59 m/s enakomerno pospeševati do hitrosti 91 m/s. Pospeševanje traja 49 s. Izračunaj povprečno hitrost.  
Odgovor: 3.06 **Pravilno!**
- Tovornjak vozi s hitrostjo 17 km/h. V nekem trenutku začne enakomerno pospešeno zavirati. Po 57 m se ustavi. Koliko časa se ustavlja?  
Odgovor: 24.14 **Pravilno!**

Lep pozdrav, Ustvarjalec nalog.

Slika 48: Avtomatsko generiran mail

## 4.2 Struktura kode

### 4.2.1 Vadnik

Na začetku programa vnesemo število nalog, ki jih želimo narediti. Pri tem se naključno izbere seznam nalog, ki jih bomo rešili, ki je velik toliko, kolikor smo izbrali številno nalog.

Prikaže se prvo vprašanje, njegova pozicija (pos) oz. številka vprašanja je 1.

Ta zanka se ponavlja dokler ni njegova pozicija enaka številu vseh vprašanj.

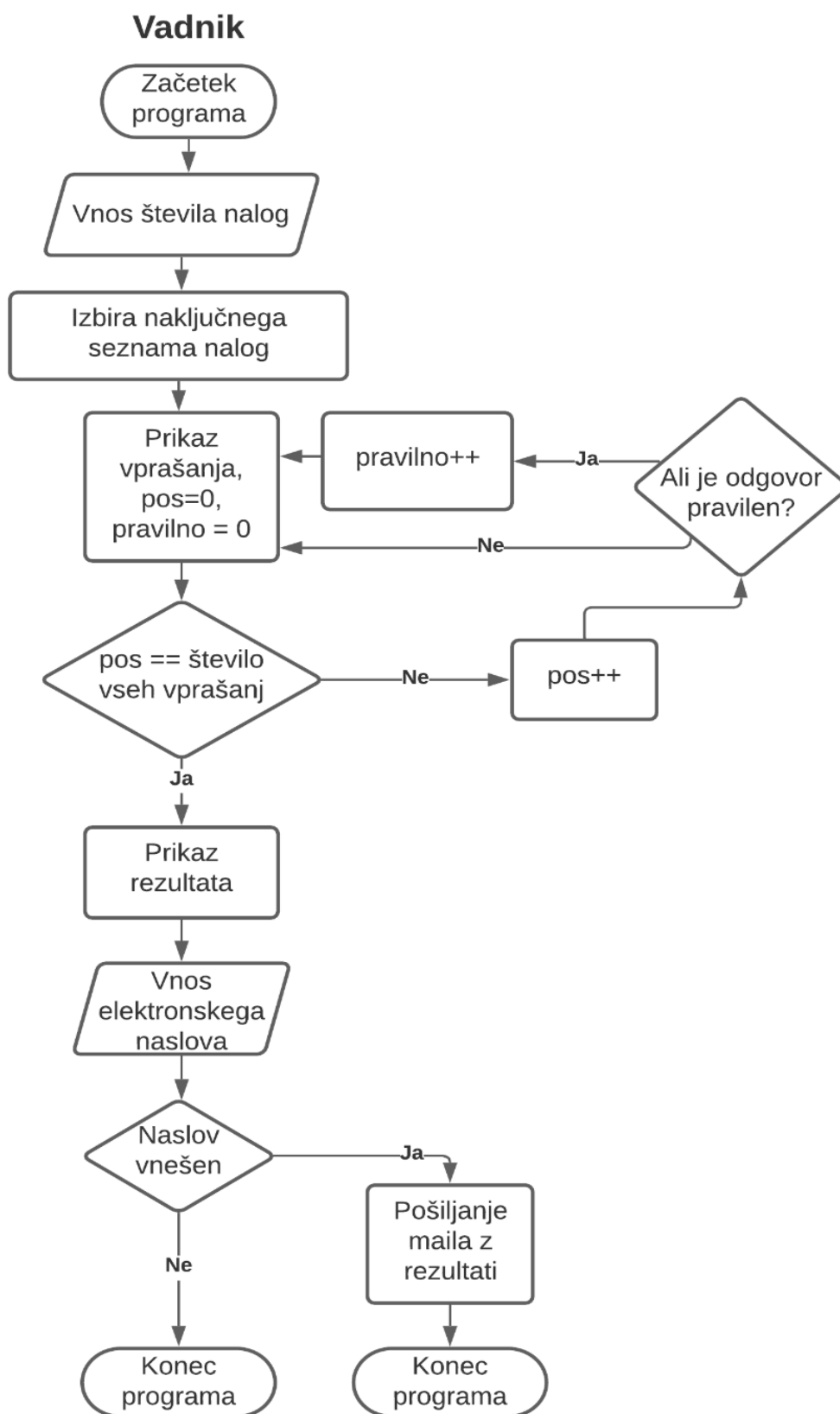
Če pozicija ni enaka, se poveča za 1 in se prikaže naslednje vprašanje.

Po rešenem vprašanju je lahko odgovor pravilen ali nepravilen.

V primeru, da je odgovor pravilen, se spremenljivki pravilno, ki nam beleži, koliko odgovorov smo rešili pravilno, prišteje 1, če vprašanje ni pravilno, se ne zgodi nič. V obeh primerih se nato spremenljivki pos prišteje 1.

Če je pozicija enaka številu vseh vprašanj, torej ko smo prišli do konca testa, se prikaže rezultat v odstotkih in lahko vnesemo elektronski naslov.

Naslov lahko vnesemo ali ne. Če ga ne vnesemo, je konec programa, če pa ga vnesemo, si lahko pošljemo rezultate na svoj e-mail.

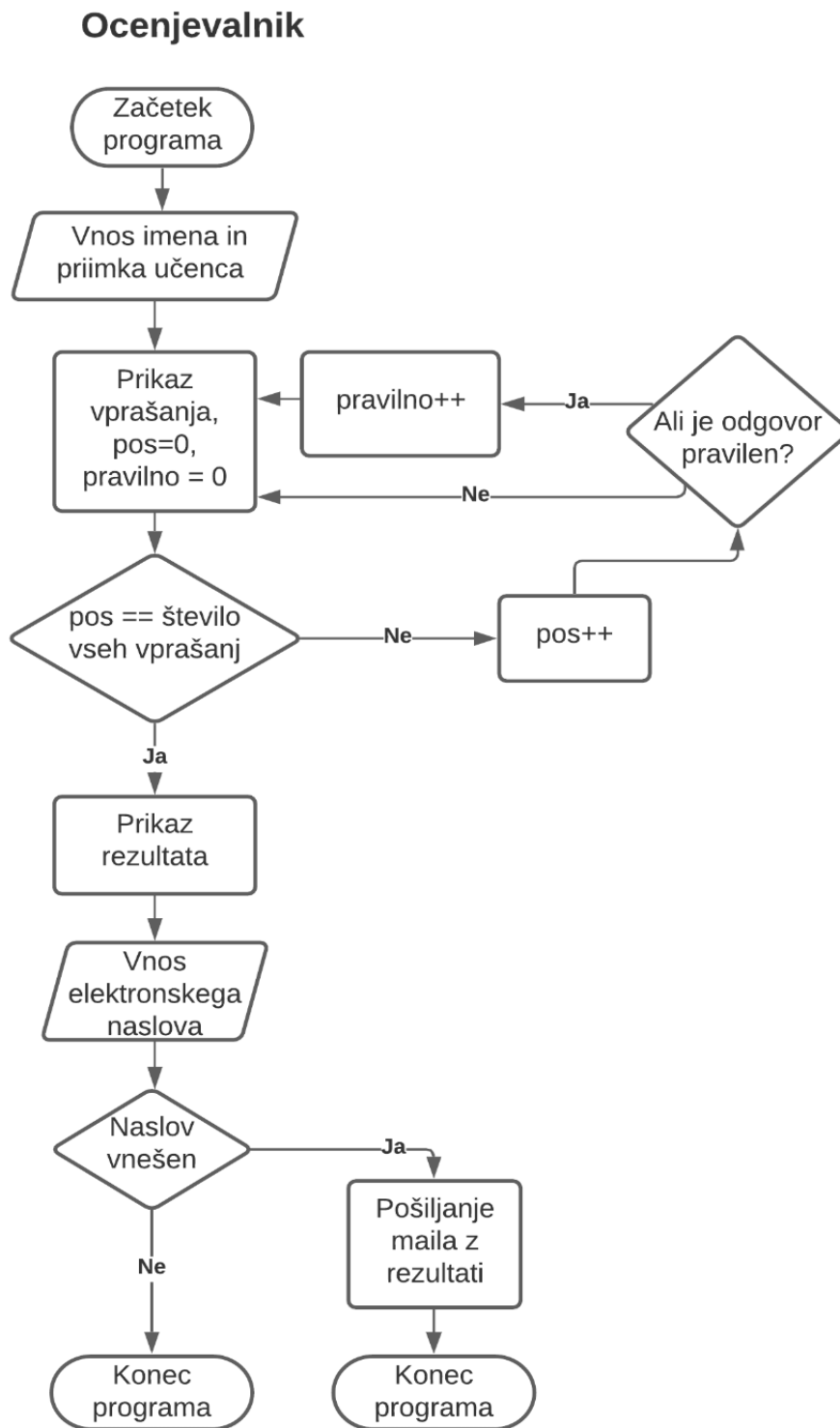


Slika 49: Shema delovanja Vadnika



## 4.2.2 Ocenjevalnik

Ocenjevalnik poteka po podobnem principu kot Vadnik.



Slika 50: Shema delovanja Ocenjevalnika

Razlika je v začetni zgradbi.

Program Ocenjevalnik se začne z vpisom imena in priimka učenca, ki bo pisal test. Nato pa se nadaljuje tako kakor pri Vadniku.

Rešene naloge in rezultat testa učenec lahko pošlje na e-mail učitelja, tako da vnese elektronski naslov, lahko pa tudi ne in je programa konec.

### **4.3 Testiranje aplikacije**

Ko sem izdelal aplikacijo, sem jo najprej testiral sam, nato pa sem povezavo poslal učencem in učiteljici. Aplikacija je delovala. Testne osebe pa so opozorile na nekaj napak pri sestavi in rešitvah nalog iz fizike.

Izkazalo se je tudi, da uporabniki zelo slabo (ali sploh ne) preberejo navodila (npr. kako zapisati rešitev, na koliko decimalnih mest natančno...).

Aplikacijo lahko testiramo z GitHub Pages na naslovu: <https://matejlajevc.github.io/>, kjer je objavljen Ocenjevalnik.

## 5 RAZPRAVA

### 5.1 Potrjevanje hipotez

1. hipoteza: **»Za programiranje aplikacije bom potreboval veliko časa.«** je potrjena.

Hipotezo potrjujem saj sem za izdelavo aplikacije porabil 3 mesece.

2. hipoteza: **»Aplikacijo bodo lahko uporabljali učenci samostojno.«** je potrjena.

Namenjena je učencem v času pouka in doma. Pri čemer je enostavna in učinkovita za utrjevanje in učenje.

3. hipoteza: **»Aplikacija bo uporabna tudi pri pouku (npr. ocenjevanje na daljavo).«** je

potrjena. Aplikacija bo lahko uporabna tudi pri pouku, saj je Ocenjevalnik namenjen prav preverjanju znanja med poukom. Možnost uporabe bo lahko na daljavo ali pa kar v šoli.

Seveda pa bo potreben večji nabor nalog različnih težavnosti.

### 5.2 Ugotovitve

Prvič sem se lotil izdelave spletne aplikacije, kar je bilo zame osebno velik izziv. Ugotovil sem, da je programiranje na začetku težko, sčasoma pa vse lažje.

Aplikacija, ki je narejena, bi lahko pomagala učencem, predvsem pa učiteljem, saj bi bili z uporabo aplikacije rezultati ocenjevanj bolj realni, ocenjevanja pa bolj poštena. S testiranjem aplikacije (preskusil sem jo sam, preskusili so jo učenci in učitelji) sem ugotovil, da aplikacija deluje. Uporabniki so opozorili na napake, ki so se nanašale predvsem na fizikalni del (sestave nalog, rešitve), kjer je bilo potrebno nekaj popravkov.

Ugotovil sem tudi, da uporabniki zelo slabo berejo navodila, ali pa jih sploh ne berejo, ampak se takoj lotijo reševanja. Navodila preberejo šele, ko se jim kaj zalomi.

## 6 ZAKLJUČEK

Ob raziskovanju sem izvedel veliko novega o programski opremi in programiranju.

Zanimalo me je, ali bi se ta aplikacija obnesla pri fiziki v osnovnih šolah.

Spletna stran omogoča nov način ocenjevanja testov in utrjevanja nalog, kako pa se bo to obneslo v praksi oziroma kako bo potekal nadaljnji pouk pri delu na daljavo, pa bomo šele videli.

Precej enostavno bi aplikacijo lahko uporabljali tudi pri matematiki.

Aplikacijo bi lahko sčasoma še nadgradil oz. izboljšal, saj je še veliko možnosti za izboljšave. Aplikacija naloge sedaj izbira iz ene baze. Lahko bi naredil več različnih baz nalog za naloge različnih težavnosti. Aplikacija bi lahko tudi nudila možnost, da si učenec izbere, kako zahtevne naloge bo reševal.

Učenec, ki ima šibko znanje, bi na začetku najbrž izbral iz baze enostavnejših nalog, učenec, ki ima odlično znanja, pa iz baze najzahtevnejših nalog. Lahko pa bi se učenec odločil tudi za mešane naloge.

Učenec bi pri Vadniku izbral število nalog in stopnjo težavnosti le-teh.

Pri Ocenjevalniku bi lahko dodal tudi dodatno funkcijo, ki bi učiteljem omogočila izdelavo večjega števila testov z naključnimi številskimi podatki, ki bi jih nato lahko natisnili in uporabili pri ocenjevanju v šoli.

Vsak začetek je težak in tudi moj začetek pri programiranju aplikacije ni bil najlažji. Ob delu sem se veliko naučil, sama aplikacija pa nudi veliko možnosti za nadgradnjo.

## 7 LITERATURA IN VIRI

### Viri besedila:

A short history of the Web. Pridobljeno 12.1.2021 s <https://home.cern/science/computing/birth-web/short-history-we>

Algorithm. Pridobljeno 12.1.2021 s <https://techterms.com/definition/algorithm>

Algoritmi navodila/jeziki. Pridobljeno 24.1.2021 s [http://erid.tsckr.si/4/algoritmi\\_navodila/jeziki.html](http://erid.tsckr.si/4/algoritmi_navodila/jeziki.html)

Application software. Pridobljeno 16.1.2021 s [https://en.wikipedia.org/wiki/Application\\_software](https://en.wikipedia.org/wiki/Application_software)

Computer Basics: 20 Examples of Computer Uses. Pridobljeno 12.1.2021 s <https://turbofuture.com/computers/7-Uses-of-Computers>

Computer Basics-Understanding Applications. Pridobljeno 16.1.2021 s <https://edu.gcfglobal.org/en/computerbasics/understanding-applications/1/>

Computer program. Pridobljeno 12.1.2021 s [https://en.wikipedia.org/wiki/Computer\\_program](https://en.wikipedia.org/wiki/Computer_program)

CSS. Pridobljeno 25.1.2021 s <http://zaversnik.fmf.uni-lj.si/Gradiva/css/index.php>

CSS Tutorial. Pridobljeno 21.2.2021 s <https://www.w3schools.com/css/default.asp>

Delitev programskih jezikov. Pridobljeno 22.1.2021 s [http://colos.fri.uni-lj.si/eri/RACUNALNISTVO/PROG\\_JEZIKI\\_ORODJA/02DelitevProgJezikov/02\\_delitevProgramskihJezikov.html](http://colos.fri.uni-lj.si/eri/RACUNALNISTVO/PROG_JEZIKI_ORODJA/02DelitevProgJezikov/02_delitevProgramskihJezikov.html)

ENIAC. Pridobljeno 12.1.2021 s <https://sl.wikipedia.org/wiki/ENIAC>

Gradivo: programski jeziki-predstavitev. Pridobljeno 23.1.2021 s [https://dijaski.net/gradivo/rif\\_ref\\_programski\\_jeziki\\_01\\_\\_predstavitev](https://dijaski.net/gradivo/rif_ref_programski_jeziki_01__predstavitev)

Hipertekst. Pridobljeno 17.1.2021 s <https://sl.wikipedia.org/wiki/Hipertekst>

History of personal computers. Pridobljeno 12.1.2021 s [https://en.wikipedia.org/wiki/History\\_of\\_personal\\_computers](https://en.wikipedia.org/wiki/History_of_personal_computers)

HTML Tutorial. Pridobljeno 20.2.2021 s <https://www.w3schools.com/html/default.asp>

Interactive program. Pridobljeno 12.1.2021 s <https://www.computerhope.com/jargon/i/inteprog.htm>

JavaScript. Pridobljeno 24.1.2021 s <https://sl.wikipedia.org/wiki/JavaScript>

Javascript Tutorial. Pridobljeno 20.2.2021 s <https://www.w3schools.com/js/default.asp>

Javascript Tutorial: Make a quiz. Pridobljeno 21.2.2021 s <https://www.youtube.com/watch?v=C7NslRhoWuE>

Kaj je URL naslov in kako ga pravilno uporabiti pri izdelavi spletne strani. Pridobljeno 17.1.2021 s <https://spletnik.si/blog/kaj-je-url-naslov-in-kako-ga-pravilno-uporabiti-pri-izdelavi-spletne-strani/>

Mobile app. Pridobljeno 16.1.2021 s [https://en.wikipedia.org/wiki/Mobile\\_app](https://en.wikipedia.org/wiki/Mobile_app)

Odjemalec/strežnik. Pridobljeno 17.1.2021 s <http://www2.arnes.si/~ssmpese/i/Informatika2/Prispevki/Borko2aOdjemalecStreznik.htm>

Programska oprema. Pridobljeno 14.1.2021 s [https://sl.wikipedia.org/wiki/Programska\\_oprema](https://sl.wikipedia.org/wiki/Programska_oprema)

Programski jeziki. Pridobljeno 22.1.2021 s <https://osnoveprogramiranja.weebly.com/programski-jeziki.html>

Računalnik. Pridobljeno 12.1.2021 s <https://sl.wikipedia.org/wiki/Ra%C4%8Dunalnik>

Računalniško programiranje. Pridobljeno 22.1.2021 s [https://sl.wikipedia.org/wiki/Ra%C4%8Dunalni%C5%A1ko\\_programiranje](https://sl.wikipedia.org/wiki/Ra%C4%8Dunalni%C5%A1ko_programiranje)

Reason of why people use computer. Pridobljeno 12.1.2021 s <https://sites.google.com/site/liptonyunz/home/reason-of-why-people-use-computer>

Role of computers in physics education – A review. Pridobljeno 14.1.2021 s <https://www.journalcra.com/article/role-computers-physics-education-%E2%80%93-review>

Sistemska programska oprema. Pridobljeno 14.1.2021 s [http://egradivo.ecnm.si/KIT/sistemska\\_programska\\_oprema.html](http://egradivo.ecnm.si/KIT/sistemska_programska_oprema.html)

Sistemska programska oprema. Pridobljeno 14.1.2021 s <http://racunalnistvo2ggl.splet.arnes.si/programska-oprema/sistemska-programska-oprema/>

Spletna stran. Pridobljeno 17.1.2021 s [https://sl.wikipedia.org/wiki/Spletna\\_stran](https://sl.wikipedia.org/wiki/Spletna_stran)

Spletni strežnik. Pridobljeno 17.1.2021 s [https://sl.wikipedia.org/wiki/Spletni\\_stre%C5%BEnik](https://sl.wikipedia.org/wiki/Spletni_stre%C5%BEnik)

Stack Owerflow. Pridobljeno 21.2.2021 s <https://stackoverflow.com/>

Strežnik. Pridobljeno 17.1.2021 s <https://sl.wikipedia.org/wiki/Stre%C5%BEnik>

Test-generator meaning. Pridobljeno 21.2.2021 s <https://www.yourdictionary.com/test-generator>

The best browser 2021. Pridobljeno 17.1.2021 s <https://www.techradar.com/best/browser>

Types of programs. Pridobljeno 14.1.2021 s [http://www.cs.ucc.ie/~gavin/cs1001/Notes/chap01/ch01\\_9.html](http://www.cs.ucc.ie/~gavin/cs1001/Notes/chap01/ch01_9.html)

Uporabniška programska oprema. Pridobljeno 16.1.2021 s  
[http://egradivo.ecnm.si/KIT/uporabnika\\_programska\\_oprema.html](http://egradivo.ecnm.si/KIT/uporabnika_programska_oprema.html)

URL. Pridobljeno 17.1.2021 s <https://sl.wikipedia.org/wiki/URL>

Web application. Pridobljeno 16.1.2021 s [https://en.wikipedia.org/wiki/Web\\_application](https://en.wikipedia.org/wiki/Web_application)

Web browser. Pridobljeno 17.1.2021 s [https://en.wikipedia.org/wiki/Web\\_browser](https://en.wikipedia.org/wiki/Web_browser)

What is a computer Program and How It Work? Pridobljeno 12.1.2021 s  
<https://www.youtube.com/watch?v=IP5s13qEB4A>

What is bulma? Advantages & disadvantages of using Bulma. Pridobljeno 24.1.2021 s  
<https://mobiosolutions.com/what-is-bulma-advantages-disadvantages-of-using-bulma-2/What-Is-GitHub,-and-What-Is-It-Used-For/>  
Pridobljeno 24.1.2021 s  
<https://www.howtogeek.com/180167/htg-explains-what-is-github-and-what-do-geeks-use-it-for/>

What is CSS Framework. Pridobljeno 24.1.2021 s <https://medium.com/html-all-the-things/what-is-a-css-framework-f758ef0b1a11>

What is Desktop Application. Pridobljeno 16.1.2021 s <https://iphtechnologies.com/what-is-desktop-application/>

Vloga programa in programskega jezika. Pridobljeno 11.5.2021 s [http://colos.fri.uni-lj.si/eri/INFORMATIKA/PROGRAMIRANJE/vloga\\_programa\\_prog\\_jezika.html](http://colos.fri.uni-lj.si/eri/INFORMATIKA/PROGRAMIRANJE/vloga_programa_prog_jezika.html)

XHTML. Pridobljeno 25.1.2021 s <http://www.s-sers.mb.edus.si/gradiva/w3/html/index.html>

#### **Viri slik:**

Slika 1: <https://i.redd.it/souti8knfr911.jpg> (15.1.2021)

Slika 2: <https://upload.wikimedia.org/wikipedia/commons/thumb/7/70/Apple-II.jpg/250px-Apple-II.jpg> (15.1.2021)

Slika 3: <https://qz.com/1568970/tim-berners-lee-on-founding-the-world-wide-web-30-years-ago/>  
(15.1.2021)

Slika 4: [https://apollo-ireland.akamaized.net/v1/files/3dr49vqbi7zu1-LB/image;s=644x461;olx-st/\\_1\\_.jpg](https://apollo-ireland.akamaized.net/v1/files/3dr49vqbi7zu1-LB/image;s=644x461;olx-st/_1_.jpg) (15.1.2021)

Slika 5: [https://business.inquirer.net/files/2020/07/istock\\_598255172.0c31b095529.original-620x386.jpg](https://business.inquirer.net/files/2020/07/istock_598255172.0c31b095529.original-620x386.jpg) (15. 1. 2021)

Slika 6: <https://siol.net/media/img/40/d2/3bb101b047afe2b76064.jpeg>(19.1.2021)

Slika 7: <https://www.computerhope.com/jargon/w/word.gif>

Slika 8: <http://www.geministyle.si/images/racunalnistvo/delitev-programske-opreme.jpg>  
(19.1.2021)

Slika 9:

[https://upload.wikimedia.org/wikipedia/commons/thumb/9/9b/Operating\\_system\\_placement-sl.svg/1200px-Operating\\_system\\_placement-sl.svg.png](https://upload.wikimedia.org/wikipedia/commons/thumb/9/9b/Operating_system_placement-sl.svg/1200px-Operating_system_placement-sl.svg.png)(19.1.2021)

Slika 10: [https://internet-lab.ru/sites/internet-lab.ru/files/styles/shirokiy/public/2019-02/windows.png?itok=FVv7M\\_4Q](https://internet-lab.ru/sites/internet-lab.ru/files/styles/shirokiy/public/2019-02/windows.png?itok=FVv7M_4Q)(19.1.2021)

Slika 11: [https://pbs.twimg.com/profile\\_images/1228219239808417794/65vjbHgf\\_400x400.png](https://pbs.twimg.com/profile_images/1228219239808417794/65vjbHgf_400x400.png)  
(20.1.2020)

Slika 12: [https://www.builtbybel.com/media/tz\\_portfolio\\_plus/article/cache/cloneapp-ua-9-h\\_o.png](https://www.builtbybel.com/media/tz_portfolio_plus/article/cache/cloneapp-ua-9-h_o.png) (20.1.2021)

Slika 13: [https://miro.medium.com/max/6048/0\\*5pDOle8wGUOzXG2D](https://miro.medium.com/max/6048/0*5pDOle8wGUOzXG2D)(20.1.2021)

Slika 14: <https://i.ytimg.com/vi/TMS-QF9pqGE/maxresdefault.jpg>(20.1.2021)

Slika 15: <http://everybestof.com/images/office.png>(20.1.2021)

Slika 16: [https://upload.wikimedia.org/wikipedia/commons/4/41/Sistema\\_hipertextual.jpg](https://upload.wikimedia.org/wikipedia/commons/4/41/Sistema_hipertextual.jpg)  
(20.1.2021)

Slika 17: <https://siol.net/media/img/40/d2/3bb101b047afe2b76064.jpeg>(21.1.2021)

Slika 18: <https://gradiva.txt.si/m/racunalnistvo/nacrtovanje-in-razvoj-spletnih-aplikacij/img/komunikacija1.png>(21.1.2021)

Slika 19: [https://cdn.ttgtmedia.com/rms/onlineimages/networking-basic\\_url\\_structure.png](https://cdn.ttgtmedia.com/rms/onlineimages/networking-basic_url_structure.png)(21.1.2021)

Slika 20: <https://i.ytimg.com/vi/Flrgsfsvzll/hqdefault.jpg>(21.1.2021)

Slika 21: [https://p.kindpng.com/picc/s/171-1718046\\_javascript-programming-language-logo-hd-png-download.png](https://p.kindpng.com/picc/s/171-1718046_javascript-programming-language-logo-hd-png-download.png) (21.1.2021)

Slika 22:

[https://upload.wikimedia.org/wikipedia/commons/thumb/d/d5/CSS3\\_logo\\_and\\_wordmark.svg/1200px-CSS3\\_logo\\_and\\_wordmark.svg.png](https://upload.wikimedia.org/wikipedia/commons/thumb/d/d5/CSS3_logo_and_wordmark.svg/1200px-CSS3_logo_and_wordmark.svg.png)(21.1.2021)

Slika 23: <https://remotejobtips.com/wp-content/uploads/2020/11/computer-programming-3ASZ5BD-1024x683.jpg> (21.1.2021)

Slika 24: [http://colos.fri.uni-](http://colos.fri.uni-lj.si/eri/RACUNALNISTVO/PROG_JEZIKI_ORODJA/02DelitevProgJezikov/slike/machineLanguage.png)

[lj.si/eri/RACUNALNISTVO/PROG\\_JEZIKI\\_ORODJA/02DelitevProgJezikov/slike/machineLanguage.png](http://colos.fri.uni-lj.si/eri/RACUNALNISTVO/PROG_JEZIKI_ORODJA/02DelitevProgJezikov/slike/machineLanguage.png)  
(21.1.2021)

Slika 25: [http://colos.fri.uni-](http://colos.fri.uni-lj.si/eri/RAC_SISTEMI_OMREZJA/html/DELO_RAC_ZBIRNIK/zbirnik_zbiranje_files/image010.gif)

[lj.si/eri/RAC\\_SISTEMI\\_OMREZJA/html/DELO\\_RAC\\_ZBIRNIK/zbirnik\\_zbiranje\\_files/image010.gif](http://colos.fri.uni-lj.si/eri/RAC_SISTEMI_OMREZJA/html/DELO_RAC_ZBIRNIK/zbirnik_zbiranje_files/image010.gif)(21.1.2021)



Slika 26: [http://colos.fri.uni-lj.si/eri/RACUNALNISTVO/PROG\\_JEZIKI\\_ORODJA/02DelitevProgJezikov/slike/macro.GIF](http://colos.fri.uni-lj.si/eri/RACUNALNISTVO/PROG_JEZIKI_ORODJA/02DelitevProgJezikov/slike/macro.GIF)(22.1.2021)

Slika 27: [https://www.kindpng.com/picc/m/355-3559027\\_c-programming-language-logo-clipart-png-download-c.png](https://www.kindpng.com/picc/m/355-3559027_c-programming-language-logo-clipart-png-download-c.png)(22.1.2021)

Slika 28: <https://sdtimes.com/wp-content/uploads/2019/01/opengraph-icon-200x200.png>(22.1.2021)

Slika 29: <https://upload.wikimedia.org/wikipedia/commons/thumb/c/c9/AtariBasic.png/300px-AtariBasic.png>(22.1.2021)

Slika 30: [https://miro.medium.com/max/1200/1\\*2u1kXxhZMN6gG-\\_aB71bKA.png](https://miro.medium.com/max/1200/1*2u1kXxhZMN6gG-_aB71bKA.png)(22.1.2021)

Slika 31: [https://miro.medium.com/max/572/1\\*R0cClyt18NQM477WP6jS0Q.png](https://miro.medium.com/max/572/1*R0cClyt18NQM477WP6jS0Q.png)(22.1.2021)

Slika 32: [https://www.kindpng.com/picc/m/355-3559027\\_c-programming-language-logo-clipart-png-download-c.png](https://www.kindpng.com/picc/m/355-3559027_c-programming-language-logo-clipart-png-download-c.png)(22.1.2021)

Slika 33:  
[https://upload.wikimedia.org/wikipedia/commons/thumb/1/18/ISO\\_C%2B%2B\\_Logo.svg/1200px-ISO\\_C%2B%2B\\_Logo.svg.png](https://upload.wikimedia.org/wikipedia/commons/thumb/1/18/ISO_C%2B%2B_Logo.svg/1200px-ISO_C%2B%2B_Logo.svg.png)(22.1.2021)

Slika 34: <https://lusy.fri.uni-lj.si/ucbenik/book/1201/img/python-windows.png>(22.1.2021)

Slika 35: <https://image.slidesharecdn.com/html5-131115133810-phpapp02/95/html5-features-with-examples-12-638.jpg?cb=1384523319>(23.1.2021)

Slika 36: <https://startingelectronics.org/tutorials/arduino/ethernet-shield-web-server-tutorial/CSS-introduction/CSS-ex1.png>(23.1.2021)

Slika 37: <https://zetcode.com/img/web/jquery/brackets.png>(23.1.2021)

Slika 38: <https://itriangletechnolabs.com/blog/wp-content/uploads/2020/05/bootstrap-illustration.png>(23.1.2021)

Slika 39: <https://bulma.io/images/bulma-banner.png>(23.1.2021)

Slika 40:  
[https://ehmatthes.github.io/images/github\\_pages\\_non\\_rubyist/repo\\_home\\_page\\_800px.png](https://ehmatthes.github.io/images/github_pages_non_rubyist/repo_home_page_800px.png)  
(24.1.2021)

Slike 41 – 48: Zaslonske slike

Slika 49: Shematični prikaz delovanja Vadnika

Slika 50: Shematični prikaz delovanja Ocenjevalca

## KAZALO

ZAHVALA .....	2
POVZETEK .....	3
1 UVOD.....	4
2 TEORETIČNA IZHODIŠČA.....	5
2.1 Pogled v zgodovino .....	5
2.2 Računalnik .....	6
2.3 Programska oprema .....	7
2.3.1 Sistemska programska oprema .....	8
2.3.2 Uporabniška ali aplikativna programska oprema ali aplikacija .....	9
2.4 Kaj je program? .....	14
2.5 Programiranje.....	15
2.5.1 Programski jezik .....	15
2.6 Kaj vse sem uporabil pri tej raziskovalni nalogi.....	21
3 RAZISKOVALNI DEL.....	25
3.1 Cilji raziskovalne naloge .....	25
3.2 Postavitev hipotez .....	25
3.3 Raziskava .....	25
4 REZULTATI - GENERATOR TESTOV IN NALOG IZ FIZIKE .....	26
4.1 Delovanje generatorja.....	26
4.1.1 Delovanje Vadnika.....	27
4.1.2 Delovanje Ocenjevalnika .....	29
4.2 Struktura kode.....	31
4.2.1 Vadnik.....	31
4.2.2 Ocenjevalnik .....	33
4.3 Testiranje aplikacije.....	34
5 RAZPRAVA .....	35
5.1 Potrjevanje hipotez .....	35
5.2 Ugotovitve.....	35
6 ZAKLJUČEK.....	36
7 LITERATURA IN VIRI .....	37
KAZALO .....	42