

SIZiF – i-sistem izpitov na Zdravstveni fakulteti Univerze v Ljubljani

Miha Fošnarč¹, Samo Penič²

¹Univerza v Ljubljani, Zdravstvena fakulteta, Zdravstvena pot 5, 1000 Ljubljana, Slovenija

²Univerza v Ljubljani, Fakulteta za elektrotehniko, Tržaška cesta 25, 1000 Ljubljana, Slovenija

E-pošta: miha.fosnaric@zf.uni-lj.si, samo.penic@fe.uni-lj.si

Povzetek. SIZiF je informacijsko-tehnološko podprt sistem za izdelavo, razpoznavo in oceno pisnih preizkusov znanja na Zdravstveni fakulteti Univerze v Ljubljani. Omogoča sestavljanje in urejanje izpitnih pol ter avtomatsko ocenjevanje in objavo študentskih izdelkov. SIZiF je nastal in se razvija na podlagi lastnega znanja in izkušenj. Vsa programska oprema je odprtokodna, torej je izvorna koda dostopna vsakomur za uporabo, prilagajanje in nadaljnji razvoj ter vključitev v druge rešitve. Članek opisuje tehnološke rešitve, na katerih sloni proces sestavljanja, izvedbe, ocenjevanja in objave pisnih preizkusov znanja, ter praktične izkušnje z uvajanjem sistema na Zdravstveni fakulteti Univerze v Ljubljani.

Ključne besede: pisni preizkusi znanja, optična razpoznavna, odprta koda, avtomatsko ocenjevanje, avtomatska generacija

SIZiF – an IT System for Exams at the Faculty of Health Sciences

SIZiF is a system based on information technology for generation, optical recognition and automatic assessment of written exams used at the Faculty of Health Sciences, University of Ljubljana. It is developed and maintained on the basis of our own knowledge and experience. All the software is open sourced, which means that it is freely accessible to everyone to use, develop and further implement functionality. The paper describes technological solutions behind the whole process from generation of questions and problem sheet, to the assessment and publication of students' results, as well as our practical experience with SIZiF at the Faculty of Health Sciences, University of Ljubljana.

1 UVOD

Finančna podhranjenost našega visokega šolstva pogosto privede do previsokih kvot študentov na pedagoškega delavca. Kako ustaviti ta trend, ki posledično zmanjšuje kakovost pedagoškega dela, je kompleksen problem [1], [2], [3]. V tem prispevku predstavljamo tehnološko rešitev, ki omogoča izboljšave pedagoškega procesa v že obstoječem finančnem in kadrovskem okviru fakultete in univerze.

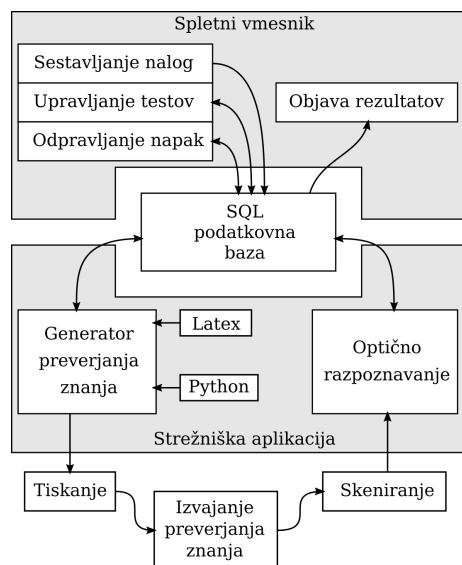
V visokošolskem izobraževanju so praviloma pisni preizkusi znanja tisti, ki dajo študentu prvo resno in sistematizirano povratno informacijo o kakovosti njegovega znanja, kar je še dodaten motiv, da so ti izvedeni po čim višjih pedagoških standardih [4], [5], [6], [7]. Takšni preizkusi znanja so pogosto množično obiskani. Na Zdravstveni fakulteti Univerze v Ljubljani se pogosto srečujemo s pisnimi preverjanji znanja, ki

jih opravlja okoli sto ali več študentov. Pogosto so nosilci predmetov ali (so)izvajalci zunanji sodelavci in večino bremena logistične izvedbe množičnih testov prevzamejo študentski referat in drugi sodelavci. Zato je še toliko pomembnejše, da je priprava in izvedba takšnih preizkusov znanja čim bolj sistematizirana in zagotavlja čim večjo objektivnost in korektnost.

SIZiF, informacijsko-tehnološki sistem za izpite na Zdravstveni fakulteti Univerze v Ljubljani, je nastal po zgledu že obstoječih rešitev [8], [9], [10] in predvsem na podlagi dolgoletnih izkušenj avtorjev z razvojem in uporabo podobnega sistema [11], [12]. Kljub prisotnosti podobnih rešitev pa je SIZiF sestavljen in spisan povsem na novo in prilagojen potrebam Zdravstvene fakultete Univerze v Ljubljani.

SIZiF omogoča izdelavo in urejanje izpitnih pol ter avtomatsko razpoznavo in ocenjevanje študentskih izdelkov. S sistemom lahko sestavimo za vsakega študenta unikatno polo z vprašanji in ponujenimi odgovori. Tudi kadar želimo, da študentje rešujejo isti vsebinski nabor nalog, lahko sistem premeša vrstni red nalog in ponujenih odgovorov. Unikatne pole z vprašanji natisnemo in razdelimo med študente. Ko študenti zaključijo izpolnjevanje pol, pole pošljemo skozi optični čitalnik, ki njihove slike pošlje strežniku. Ta oceni vsak izdelek in ga poveže s študentom, ki ga je izpolnjeval. Če izvajalec predmeta želi, omogoči študentu, da si lahko svoj pisni preizkus, popravljen in ocenjen, ogleda na spletu ter sporoči morebitne pripombe. Za varstvo osebnih podatkov študentov je pri tem poskrbljeno.

Vsa programska oprema je odprtokodna [13] in je torej prosto dostopna za uporabo in nadaljnji razvoj ter vključitev v druge rešitve.



Slika 1: Shema zgradbe sistema SIZiF.

V nadaljevanju predstavimo zgradbo in tehnološke rešitve SIZiFa ter izkušnje s sistemom v praksi.

2 ZGRADBA IN DELOVANJE SIZiFa

Posamezne segmente programske opreme SIZiFa medsebojno povezuje podatkovna baza (sredina sheme na sliki 1). Podatki so v skladu z zgradbo podatkovnih baz SQL urejeni v tabele. Te se medsebojno povezujejo z referencami na posamezna polja in tvorijo celoto. Uporabili smo odprtokodno podatkovno bazo `postgresql`, vendar je mogoče uporabiti katerokoli odprtokodno ali komercialno podatkovno bazo. Spletni vmesnik in strežniška aplikacija sta napisana v visokonivojskem programskem jeziku Python, ki že s svojo zasnovno spodbuja uporabo in razvoj številnih odprtokodnih knjižnic ter tako skupaj s svojo razširjenostjo nudi odlično odprto programsko okolje znanstvenikom in inženirjem [14].

Pisni preizkus znanja se v SIZiFu začne s sestavljanjem nalog ter določanjem roka preizkusa znanja. Pisne preizkuse znanja v SIZiFu sestavljamo in urejamo prek spletnega vmesnika (slika 2), pri čemer vsa komunikacija teče prek varnega protokola HTTPS. Spletna stran je zgrajena v spletnem okvirju (angl. web framework) Django 2.2, ki omogoča enostavno komunikacijo s podatkovno bazo v ozadju ter čisto in pragmatično obliko [15].

Vsaka naloga je svoja tekstovna datoteka. SIZiF ima svoj minimalističen označevalni jezik (angl. markup language), s katerim poleg besedila naloge podamo tudi opis, kako naj se besedilo oblikuje ter kateri so pravilni in napačni odgovori [11], [12].

Sestavljene naloge so v celoti spravljene v podatkovni bazi in pripravljene za vključitev v preizkus znanja. Posamezno nalogo lahko uporabimo pri več preizkusih

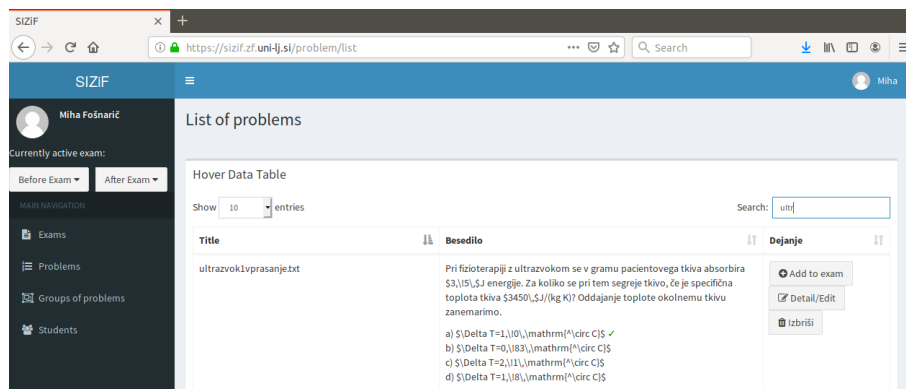
ali pa za vsak preizkus sestavimo nove naloge. Ko smo zadovoljni z naborom nalog in preostalimi parametri preizkusa znanja (datum, naslov, ime predmeta, seznam pišočih, ...), podamo zahtevo za generacijo unikatnih izdelkov.

Generator preverjanja znanja je aplikacija, ki deluje na strežniku in se odzove na zahtevo po generaciji. Najprej SIZiFov razčlenjevalnik (angl. parser) v datoteki poišče posamezne dele naloge, ustvari vrednosti spremenljivk in izračuna rezultate. Za numerične in simbolične izračune uporabljamo knjižnici `numpy` in `sympy`. Nato vse naloge sestavi v izpitno polo in vse izpitne pole za uporabo prilagodljive oblikovne predloge zloži v skupno \LaTeX -ovo datoteko. To z \LaTeX -ovim prevajalnikom prevedemo v datoteko PDF, ki jo lahko natisnemo in pole razdelimo med študente (slika 3).

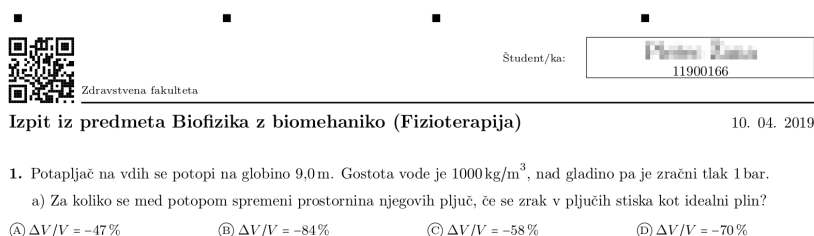
Ko študentje oddajo izpolnjene izpitne pole, te pošljemo skozi optični čitalnik, ki slike vsake strani izpitne pole v slikovnem formatu TIFF pošlje strežniku sistema SIZiF. Izbiramo lahko med več protokoli prenosa, ki jih optični čitalnik podpira. Največkrat je to kar protokol FTP (File Transfer Protocol). Strežnik opravi optično razpoznavo, poveže strani v celoten preizkus znanja posameznega študenta, v bazi študentov poišče podatke o študentih, ter oceni izdelke. Za optično razpoznavo vzorcev smo uporabili knjižnico `OpenCV` in Pythonove ovoje (angl. wrapper) zanjo. `OpenCV` uporabljamo tudi za urejanje slike: določanje sivin, čiščenje slik, izločanje posameznih števk identifikacijske številke ter iskanje kvadratkov na robovih izpitnih pol, ki služijo koordinatnemu sistemu pole. Za branje QR-kode smo uporabili knjižnico `pyzbar`. Številke razpoznavamo z nevronske mreže, definiranimi v knjižnici `sklearn`, pri čemer smo za učenje nevronske mreže uporabili metodo vzratnega razširjanja (angl. `backpropagation`) iz te knjižnice in jo naučili prepoznavanja števk od 0 do 9 na osnovi majhnega vzorca okoli 30 posameznih števk iz predhodnih testnih slik, pridobljenih na istem optičnem čitalniku.

Posamezni elementi na sliki izpitne pole služijo optični razpoznavi:

- Prek QR-kode v bazi poiščemo podatke o poli, predvsem matriko pravih odgovorov. Opcijsko je lahko v QR-kodi tudi identifikacijska številka pišočega, prek katere lahko izdelek z uporabo baze povežemo z imenom in priimkom.
- Če podatka o identifikaciji pišočega ni v QR-kodi, jo algoritem za optično razpoznavo poskusi pridobiti iz temu namenjenega polja na formularju. Študent polje izpolni tako, da pobarva segmente, ki spominjajo na 7-segmentni prikazovalnik, kot smo ga navajeni na primer iz digitalnih ur. Primer izpolnjene identifikacijske številke je prikazan na sliki 4.
- Črni kvadrati na zgornjem in desnem robu strani določajo koordinate ponujenih odgovorov. Na



Slika 2: Primer pogleda v spletnem vmesniku, prek katerega v SIZiFu urejamo preizkuse znanja ter posamezna vprašanja ali sklope vprašanj sestavljamo v naloge, ki bodo del preizkusa.



Slika 3: Primer glave izpitne pole s prvim vprašanjem. Zaradi varovanja osebnih podatkov je ime študenta na sliki zakrito. SIZiF omogoča tudi generacijo in razpoznavo izpitnih pol brez vnaprej natisnjenih imen in vpisnih števil. V tem primeru jih študenti sami vpišejo na izpitne pole, ki jih lahko med študente naključno razdelimo.

Vpisna št.: **00900204**

Slika 4: Primer izpolnjene identifikacijske številke piscočega. V primeru Zdravstvene fakultete je to vpisna številka študenta.

1. Potapljač na vdih se potopi na globino 9,0 m. Gostota vode je 1000 kg/m^3 , nad gladino pa je zračni tlak 1 bar.
- a) Za koliko se med potopom spremeni prostornina nje
- $\Delta V/V = -47\%$ $\Delta V/V = -84\%$ $\Delta V/V = -58\%$ $\Delta V/V = -70\%$

Slika 5: Lepo označen odgovor pri nalogi. Z uporabo kvadratkov na robovih bo označen odgovor na poli enostavno locirati in ugotoviti, da ga je pišoči študent označil.

presečiščih teh koordinat se potencialno nahajajo izbrani odgovori, ki jih pišoči označi tako, da pobarva krožec pred odgovorom (prikaz dela odgovorov neke naloge je na sliki 5). Algoritem zazna počrtnjene odgovore ter sestavi matriko izbranih odgovorov.

Rezultati optičnega razpoznavanja se spravijo v bazo,

skupaj s povezavo do slik, ki pa se shranijo na disk v obliki datotek tipa PNG, primernih tudi za prikaz prek spletnih brskalnikov. S primerjavo matrike pravilnih odgovorov in matrike označenih odgovorov se izračuna končni rezultat preverjanja znanja – ocena. Upravlja-vec preverjanja znanja si lahko ogleda rezultate vseh študentov skupaj s sliko njihovega preverjanja znanja ter odpravi morebitne pomanjkljivosti, ki so se pojavile med procesom optične zaznave. Hkrati mu stran nudi statistiko reševanja preizkusa znanja in izvoz ocen v obliki tabele.

Zadnji spletni vmesnik je namenjen pišočim – študentom. Rezultate preizkusa znanja si lahko ogledajo s prijavo na spletni strani. Po prijavi si ogledajo celotno zgodovino opravljanj preizkusov znanja s sistemom SIZiF. Za vsak preizkus dobijo podatek o rezultatu in statistiko celotnega preizkusa znanja ter si ogledajo svoj izdelek. Študentu je omogočeno tudi prijavljanje napake pri ocenjevanju ali zastavljanje vprašanj o nalogah. S tem lahko spletna stran popolnoma nadomesti fizični ogled preizkusov znanja z obiskom asistenta/profesorja, ki je izvajal in ocenjeval preizkus.

Zaradi zagotavljanja varnosti osebnih podatkov je strežnik SIZiF na računalniku, ki je fizično priso-

ten znotraj Zdravstvene fakultete Univerze v Ljubljani. Strojne zahteve strežnika niso velike. Posamezne komponente preizkušeno delujejo na odprtokodnih distribucijah GNU/Linux. Za namestitev sistemski inženir potrebuje poznavanje operacijskega sistema Linux ter programskega jezika Python. Za lažjo namestitev smo pripravili konfiguracijsko datoteko za virtualizacijo na nivoju operacijskega sistema Docker, ki namestitev popolnoma avtomatizira. Slika (angl. Docker image), ki jo ustvari, za osnovo uporablja distribucijo Linuxa Ubuntu 18.04. Nanjo namesti vse potrebne pakete in pripravi mrežne in diskovne dostopne točke. Tako zgrajeno sliko lahko namestimo na poljuben operacijski sistem, ki podpira sistem Docker.

Sistem SIZiF smo razdelili na tri dele zaradi lažjega razvoja posameznih funkcionalnosti sistema. Spletni vmesnik skupaj s konfiguracijsko datoteko `Dockerfile` je prosto dostopen v GIT-repozitoriju [16], generator preverjanj znanja v GIT-repozitoriju [17] ter algoritmi za optično razpoznavo v GIT-repozitoriju [18]. Če bomo uporabili sistem Docker, potrebujemo le prvi del, ki ga snamemo z repozitorija [16], Docker pa poskrbi za prenos preostalih komponent sistema.

Uporaba Dockerja ima za nas, razvijalce, še dodatno prednost. Ker se sistem zelo hitro razvija, lahko z uporabo Dockerjevih slik spremembe hitro prenašamo v produkcijo, ne da bi pri tem povzročili izpad oz. nedelovanje strežnika. Nove funkcionalnosti ali popravki se tako dnevno ali celo večkrat na dan pojavijo v novi Dockerjevi sliki, ki nadomesti staro. Izdelava novih slik je lahko avtomatizirana v obliki sprožilca ob potisku spremembe na repozitorij GIT ali kako drugače. Uporabnik storitve SIZiFa zamenjave slike niti ne zazna, saj je menjava hipna. V primeru težav z novo sliko pa lahko hitro povrnemo staro stanje, dokler ne popravimo nepravilnosti na razvojnem strežniku in ne ponovimo procesa integracije razvojne kode v produkcijsko.

3 REZULTATI IN ZAKLJUČEK

Pri zasnovi, razvoju in uporabi SIZiFa smo gradili na izkušnjah z razvojem in uporabo podobnega sistema, ki je v uporabi pri nekaterih predmetih na Fakulteti za elektrotehniko in Biotehnični fakulteti Univerze v Ljubljani [11], [12]. Vendar pa je SIZiF zasnovan in napisan povsem na novo, s številnimi povsem novimi funkcionalnostmi ter možnostmi za nadgradnjo. Tako smo se ob uvajanju sistema na Zdravstveni fakulteti Univerze v Ljubljani ob odpravljanju napak programske kode, seveda srečevali tudi z zagonskimi težavami pri postavitvi in vzdrževanju infrastrukture strežnika in optičnega čitalnika ter pri vpeljavi novih funkcionalnosti in uporabniških izkušnjah izvajalcev preizkusov znanja.

Kljub temu smo SIZiF na Zdravstveni fakulteti Univerze v Ljubljani v letih 2018 in 2019 uspešno uporabili za preizkuse znanja pri predmetih biofizika, biofizika z biomehaniko, medicinska pedevtika, klinična pro-

pedevtika in diagnostični laboratorij, zdravstvena nega kirurškega bolnika ter organizacija in management v zdravstveni negi. Pri tem smo izvedli več kot 20 pisnih preizkusov znanja ter sestavili in obdelali več kot tisoč izpitnih pol študentov.

Po nekaj začetnih težavah z implementacijo in izvedbo je SIZiF uspešno zadihal. Študentje so ga pozitivno sprejeli, ustrežata jim objektivnost takšnega preverjanja znanja in hitra povratna informacija o njihovi uspešnosti. Izvajalci izpita so zadovoljni, saj je zaradi unikatnih pol oteženo goljufanje, SIZiF pa odpravlja tudi duhomorno ročno popravljanje pisnih izdelkov študentov.

S finančnega stališča uvedba SIZiFa na Zdravstveni fakulteti ni zahtevala dodatnih investicij. Strojne zahteve za strežnik in optični bralnik so dovoljevale, da smo uporabili kar infrastrukturo, ki jo fakulteta že ima na razpolago.

SIZiF na disku trenutno zaseda 16 GB prostora, pri čemer 10 GB zasedata operacijski sistem in Dockerjeva slika sistema SIZiF z vsemi potrebnimi dodatki, baza metapodatkov zaseda 16 MB, slabih 6 GB pa zasedajo slike študentskih pol.

Tiskanje izpitnih pol in optično branje izpolnjenih izdelkov potekata na večfunkcijski napravi Xerox Work-Centre 5875. Za uspešno optično razpoznavo izdelkov je bilo treba primerno nastaviti sivine in kontraste pri tiskanju ter na optičnem bralniku. Pri menjavi tiskalnika ali optičnega bralnika bi bilo ta postopek uglaševanja sistema treba opraviti ponovno. Koristno je tudi, da imajo izvajalci roka že nekaj izkušenj s SIZiFom, saj je kakovost pravilne razpoznave izdelkov odvisna tudi od primernega izpolnjevanja študentov, na primer izbire pisala in brisala ter pravilnega označevanja izbranih odgovorov.

Največ napak seveda nastaja ob optični razpoznavi. IZpitnih pol, pri katerih so potrebni ročni popravki, je pri uspešno izvedenem izpitu sicer le nekaj odstotkov, vseeno pa je potrebna previdnost izvajalca pred objavo rezultatov.

Odprto ostaja vprašanje systemske uvedbe SIZiFa na fakulteti. Sistem namreč še vedno zahteva nekaj ekspertnega znanja pri sestavljanju preizkusov znanja, pri pregledovanju optične razpoznavne in avtomatskega ocenjevanja izdelkov ter pri vzdrževanju strežnika. Za širšo in rutinsko uporabo bi bilo treba prilagoditi uporabniški vmesnik za sestavljanje testov in pregledovanje ocenjenih izdelkov, ki bi morala biti enostavnejša in uporabniku prijaznejša. Poleg tega bi morali systemsko urediti tehnično podporo, saj gre lahko, kot pri vsakem informacijsko-tehnološkem procesu, vsakič kaj narobe. Ozko grlo sta predvsem tiskanje in optična razpoznavna, ki pa sta ključna člena, zaradi katerih smo sistem sploh razvili. Zaobiti te bi pomenilo, da mora imeti v času preizkusa znanja vsak študent na razpolago informacijsko-tehnološko vhodno enoto, prek katere sistemu sporoča svoje odgovore (na primer pametni telefon ali tablico,

uporaba računalniških učilnic na fakulteti ali posebej temu namenjena strojna in programska oprema).

Za konec naj še enkrat poudarimo, da je celoten sistem spisan in zasnovan na odprtokodnih tehnologijah. Potencialne uporabnike in razvijalce zato vabimo k uporabi in nadaljnjemu razvoju. Ena izmed koristnih razširitev bi bila na primer integracija s sistemom Moodle, ki je zelo razširjen tudi v slovenskem izobraževalnem sistemu.

ZAHVALA

Zahvaljujemo se Alešu Berkopcu, ki sorazvija in uporablja soroden sistem, na osnovi katerega smo gradili SIZiF.

LITERATURA

- [1] B. Milena, "Učinkovitost in pravičnost visokošolskega izobraževanja v Sloveniji," *Teorija in praksa let*, vol. 40, pp. 90–102, 2003.
- [2] A. Faganel and A. Trnavčević, "Marketizacijski diskurz v izobraževanju," *UPP Monograph Series*, 2017.
- [3] B. Čar Dijkhuizen, "Vpliv sistema financiranja visokega šolstva na učinkovitost in pravičnost izobraževanja: primerjava Slovenije, Nizozemske in Portugalske," Ph.D. dissertation, Univerza v Ljubljani, Fakulteta za družbene vede, 2018.
- [4] P. Ramsden, *Learning to teach in higher education*. Routledge, 2003.
- [5] D. Kember and K.-P. Kwan, "Lecturers' approaches to teaching and their relationship to conceptions of good teaching," *Instructional science*, vol. 28, no. 5, pp. 469–490, 2000.
- [6] R. Kane, S. Sandretto, and C. Heath, "Telling half the story: A critical review of research on the teaching beliefs and practices of university academics," *Review of educational research*, vol. 72, no. 2, pp. 177–228, 2002.
- [7] C. Henderson, A. Beach, and N. Finkelstein, "Facilitating change in undergraduate stem instructional practices: An analytic review of the literature," *Journal of research in science teaching*, vol. 48, no. 8, pp. 952–984, 2011.
- [8] G. Kortemeyer, E. Kashy, W. Benenson, and W. Bauer, "Experiences using the open-source learning content management and assessment system lon-capa in introductory physics courses," *American Journal of Physics*, vol. 76, no. 4, pp. 438–444, 2008.
- [9] A. Rane, A. Kumar, H. Saini, M. Sasikumar, and R. Marg, "Extending moodle to support offline assessments," in *Proceedings of national seminar on e-learning and e-learning technologies (ELELTECH)*, 2009, pp. 31–39.
- [10] J. A. Fisteus, A. Pardo, and N. F. García, "Grading multiple choice exams with low-cost and portable computer-vision techniques," *Journal of Science Education and Technology*, vol. 22, no. 4, pp. 560–571, 2013.
- [11] A. Berkopec, S. Penič, and M. Fošnarič, "Havoc – sistem za izdelavo, razpoznavo in oceno preizkusov znanja," in *Zbornik sedemnajste mednarodne Elektrotehniške in računalniške konference ERK 2008*, V. B. Z. in A. Trost, Ed. Slovenska sekcija IEEE, Sep. 2008, pp. 360–363.
- [12] S. Penič, M. Fošnarič, and A. Berkopec, "Sistem za izdelavo, razpoznavo in oceno pisnih preizkusov znanja," in *Informacijska komunikacijska tehnologija v pedagoških študijskih programih Univerze v Ljubljani*, 2019 (v tisku).
- [13] Wikipedia contributors, "Open-source software — Wikipedia, the free encyclopedia," 2019, [Online; accessed 14-June-2019]. [Online]. Available: https://en.wikipedia.org/wiki/Open-source_software
- [14] K. J. Millman and M. Aivazis, "Python for scientists and engineers," *Computing in Science & Engineering*, vol. 13, no. 2, pp. 9–12, 2011.
- [15] The Django Software Foundation, "Django," 2019, [Online; accessed 18-July-2019]. [Online]. Available: <https://www.djangoproject.com>
- [16] S. Penič, "Sizif - web public module," 2019, [Online; accessed 18-July-2019]. [Online]. Available: https://git.penic.eu/summary/havoc%2Fsizif_web_public.git
- [17] —, "Sizif - problem set generation module," 2019, [Online; accessed 18-July-2019]. [Online]. Available: <https://git.penic.eu/summary/havoc%2Fberki-parse.git>
- [18] —, "Sizif - optical recognition module," 2019, [Online; accessed 18-July-2019]. [Online]. Available: <https://git.penic.eu/summary/havoc%2Fsizif-ocr.git>

Miha Fošnarič je diplomiral na Oddelku za fiziko na Fakulteti za matematiko in fiziko Univerze v Ljubljani in doktoriral na Fakulteti za elektrotehniko Univerze v Ljubljani. Med letoma 2000 in 2018 je bil zaposlen kot mladi raziskovalec in nato kot asistent za fiziko na Fakulteti za elektrotehniko Univerze v Ljubljani. Od leta 2018 je zaposlen kot visokošolski učitelj za biofiziko na Zdravstveni fakulteti Univerze v Ljubljani.

Samo Penič je leta 2015 doktoriral s področja elektrotehnike na Univerzi v Ljubljani. Je asistent na Fakulteti za elektrotehniko Univerze v Ljubljani. Med letoma 2007 in 2011 je razvijal sistem za avtomatsko ocenjevanje HaVoC na Fakulteti za elektrotehniko. Izkušnje pri nastajanju sistema je uporabil za načrtovanje in izvedbo sistema SIZiF na Zdravstveni fakulteti Univerze v Ljubljani.