



## **Specifikacija zahtev za Sklop 2**

**Razvoj informacijske rešitve za generiranje testov in izvedbo ocenjevanj ter aplikacije za upravljanje nalog, aplikacije za interaktivne naloge, aplikacije za preračunane naloge, aplikacije za programiranje s koščki**

Verzija: v0.3

3. 4. 2025

## STATUS DOKUMENTA

<b>Dokument:</b>	2
<b>Vsebina:</b>	Glej kazalo vsebine
<b>Oznaka dokumenta:</b>	303-93/2024-3350-5
<b>Status:</b>	Končna različica
<b>Verzija:</b>	0.3
<b>Datum verzije:</b>	3.4.2025

Verzija	Datum zadnje spremembe	Opombe
0.1	29.1.2025	Priprava prvega osnutka dokumenta
0.2	4.3.2025	Uskladitev poimenovanj
0.3	3.4.2025	Dokumentacija za pripravo čistopisa

<b>Naročnik</b>	Ministrstvo za vzgojo in izobraževanje Masarykova 16 1000 Ljubljana
<b>Oznaka javnega naročila</b>	303-93/2024-3350-5
<b>Naziv javnega naročila</b>	Razvoj osmih aplikacij za potrebe Ministrstva za vzgojo in izobraževanje
<b>Sklop javnega naročila</b>	Sklop 2 IT rešitev za generiranje testov in izvedbo ocenjevanj z aplikacijo za teste in naslednje aplikacije iz IT rešitve za upravljanje nalog: aplikacija za upravljanje nalog, aplikacija za interaktivne naloge, aplikacija za preračunane naloge, aplikacija za programiranje s koščki

## KAZALO VSEBINE

Uvod .....	5
Pojmovnik .....	5
Uporabniške in funkcionalne zahteve .....	5
Uporabniški nivoji IS GTO .....	5
Uporabniške in funkcionalne zahteve IS GTO .....	6
Generiranje testov .....	8
Načrtovanje testiranja (samo elektronsko) .....	9
Izvedba testa (v elektronski obliki) .....	10
Ocenjevanje (v elektronski obliki) .....	10
Pregled nad opravljenimi testi (v elektronski obliki) .....	11
Uvoz natisnjenih testov .....	11
Po uspešnem uvozu rešitve testa, se ta shrani v sistem in učenca preko elektronske pošte obvesti o novem vnosu / povratni informaciji (oceni) s povezavo za pregled nad opravljenimi testi. ....	12
Uporabniške in funkcionalne zahteve IS UN .....	12
Funkcionalnosti vezane na gradnjo nalog (avtorska orodja) .....	13
Funkcionalnosti vezane na naloge za programiranje z delčki .....	13
Funkcionalnosti vezane na naloge za preračunane naloge .....	14
Funkcionalnosti vezane na interaktivne naloge .....	16
Funkcionalnosti vezane na preverjanje nalog .....	21
Funkcionalnosti vezane na upravljanje skupin mentorjev .....	22
Visokonivojska arhitektura sistema .....	23
IS UN .....	23
IS GTO .....	23
Uporabnost .....	24
Razpoložljivost .....	26
Zmogljivost .....	26
Nadgradljivost in skalabilnost .....	27
Priprava grafične podobe .....	28
Varnost .....	28
<b>Metodološke zahteve .....</b>	<b>29</b>
Vodenje projekta na strani izvajalca .....	29
<b>Zahteve glede izvedbe ključnih aktivnosti .....</b>	<b>29</b>
Projekt za izvedbo (PZI) .....	29
Testiranje in zagotavljanje kakovosti .....	31



Tehnična in uporabniška dokumentacija .....	32
Uporabniška dokumentacija.....	32
Usposabljanje uporabnikov .....	32

## **KAZALO SLIK**

---

Slika 1: shema procesov v IS GTO .....	8
--	---

## Uvod

Ministrstvo za vzgojo in izobraževanje želi ponuditi v uporabo IT rešitev za generiranje testov in izvedbo ocenjevanj z aplikacijo za teste. Prav tako želi ponuditi tudi aplikacijo za upravljanje nalog, aplikacijo za interaktivne naloge, aplikacijo za preračunane naloge in aplikacijo za programiranje s koščki znotraj IT rešitve za upravljanje nalog.

Navedeni IT rešitvi služita kot podpora pri izvajanju pouka. IT rešitev za generiranje testov in izvedbo ocenjevanj znanja omogoča učinkovitejšo pripravo testov in izvedbo ocenjevanja. IT rešitev za upravljanje nalog omogoča učinkovitejše ustvarjanje nalog različnih tipov in njihovo objavljane na platformah z izobraževalnimi vsebinami, vključno na platformi z nalogami, ki bo razvita v okviru IT rešitve za upravljanje nalog.

V nadaljevanju so podane uporabniške in funkcionalne zahteve obeh informacijskih rešitev.

IT rešitev za generiranje testov in izvedbo ocenjevanj (v nadaljevanju IS GTO) je povezana z IT rešitvijo za upravljanje nalog (v nadaljevanju IS UN). Povezave med IS GTO in IS UN so opisane v poglavju Seznam in opis izmenjave podatkov.

## Pojmovnik

MVI	Ministrstvo za vzgojo in izobraževanje
PZI	Projektno izvedbena dokumentacija
VIZ	Vzgojno-izobraževalni zavod
IS UN	informacijska rešitev za upravljanje nalog
IS GTO	informacijska rešitev za generiranje testov in ocenjevanje
ARNES	Akadska in raziskovalna mreža Slovenije
Arnes AAI	Avtentikacijska in Avtorizacijska Infrastruktura
API	Aplikacijski programski vmesnik
WebComponent	(ang. Web Component) sklop tehnologij, ki omogočajo ustvarjanje ponovljivih in enkapsuliranih komponent za spletne strani in aplikacije. Gre za standardiziran način ustvarjanja prilagojenih HTML elementov, ki delujejo neodvisno od drugih delov strani.
iframe	(inline frame) je HTML element, ki naloži drugo HTML stran znotraj dokumenta.
Lock down browser	Je vrsta spletnega brskalnika, zasnovana za preprečevanje goljufanja med spletnimi izpiti. Učenci ga uporabljajo pri testih in izpiti, kjer je treba omejiti dostop do drugih virov na računalniku.

## Uporabniške in funkcionalne zahteve

V tem poglavju so podane funkcionalne zahteve za IS GTO in IS UN. IS GTO predstavlja vstopno točko za uporabnika. IS UN je informacijska rešitev, ki uporabniku nudi kontrolo nad generiranimi nalogami v IS GTO in javno objavljenimi.

### Uporabniški nivoji IS GTO

IS GTO omogoča štiri uporabniške nivoje:

- **učenec** je uporabnik, ki je prijavljen v sistem, omogočen mu je pregled opravljenih testov in izpolnjevanje dodeljenih testov;
  - uporabniška vloga se dodeli uporabniku, ki je vpisan v aplikacijo z uporabo identitete Arnes AAI na podlagi atributa AAI.

- **učitelj** je uporabnik, ki je prijavljen v sistem, omogočena mu je možnost generiranja testov in njihova objava, uvoz, ocenjevanje ter pregled;
  - uporabniška vloga se dodeli uporabniku, ki je vpisan v aplikacijo z uporabo identitete Arnes AAI, ki ima atribut "učitelj".
- **mentor** je uporabnik, ki je prijavljen v sistem, omogočena mu je funkcionalnost potrjevanja nalog
  - Uporabniška vloga mentor je vezana na področje naloge: vsak mentor je zadolžen za določeno področje nalog. Področje je nabor predmetov v enem ali več razredov.
  - uporabniško vlogo dodeli administrator.
- **administrator** upravlja z uporabniškimi vlogami mentor, jih dodaja in odvzema. Uporabnika določi naročnik.

Dostop do posameznih funkcionalnosti se omeji na podlagi informacij v identiteti ArnesAAI.

Dostop se omogoči poleg uporabe identitete ArnesAAI tudi z registracijo z e-naslovom. Ob prvi prijavi z e-naslovom uporabnik izbere uporabniški nivo (učenec, učitelj).

#### Uporabniške in funkcionalne zahteve IS GTO

Funkcionalne zahteve IS GTO opisujejo, kaj mora IS GTO omogočati uporabnikom. Poleg specifičnih funkcionalnih zahtev so opisane tudi splošne funkcionalne zahteve.

IS GTO služi za generiranje testov, izvajanje testiranja, ocenjevanje testov in pregled nad opravljenimi testi. Teste definiramo kot skupek naslednjih elementov:

- Seznam nalog.
- Kriterij - predstavlja distribucijo točk po nalogah in izračun uspešnosti tako na ravni posamezne naloge kot celotnega testa.
- Metapodatki:
  - Vsak test mora biti opremljen z naslednjimi metapodatki:
    - naslov testa,
    - tema, ki jo naslavlja,
    - avtorji testa,
    - ravno izobraževanja, razred, predmet (tekom izvajanja naročnik priskrbi šifrant predmetov).

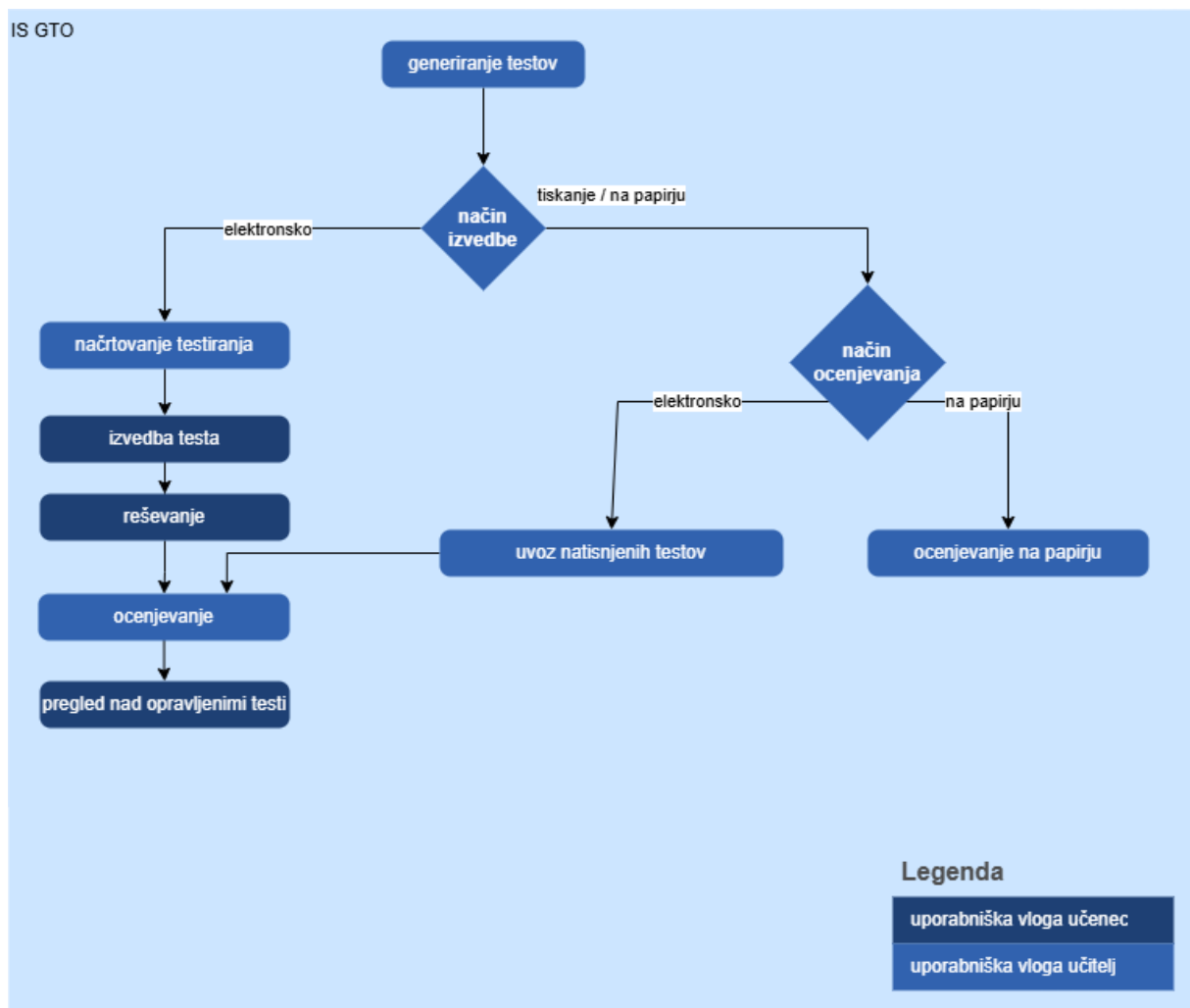
Test definiran v skladu z zgornjimi kriteriji nudi povratno informacijo o znanju učenca na določeno temo.

IS GTO nudi naslednje funkcionalnosti: (V oklepajih poleg posamezne funkcionalnosti je navedena uporabniška vloga, kateri je funkcionalnost namenjena.)

- generiranje testov individualno ali v sodelovalnem načinu (učitelj);
- ustvarjanje skupin učencev z ročnim vnosom seznama elektronskih naslovov ali množični uvoz (podpora za množični uvoz datotekam z zapisom csv, json ali xml) (učitelj);
- obveščanje skupin po elektronski pošti – vnos preko spletnega obrazca (obogateno besedilo) (učitelj);

- uvoz optično prebranih natisnjenih rešenih testov (učitelj);
- načrtovanje testiranja (učitelj)
- izvedba testa v kontroliranem okolju (učenec);
- uvoz vključenih nalog pri izvedbi testiranja na papirju (učitelj);
- ocenjevanje z možnostjo vnosa povratne informacije za posamezno nalogo v oddani rešitvi testa in celoten test (učitelj) za avtomatizirano (samodejno) ocenjevanje;
- reševanje testov (učenec);
- pregled uspešnosti reševanja po nalogah in testih za celoten razred (učitelj);
- analitični pregled nad uspešnostjo reševanja po posameznih nalogah in testih, učencih in razredih po naslednjih parametrih dosežene točke in čas reševanja z grafično interaktivno vizualizacijo za dostop do posamezne rešitve v prikazu statistike (učitelj).
- pregled nad opravljenimi testi (učenec).

Sosledje navedenih procesov z uporabniškimi vlogami, ki izvajajo posamezne procese, prikazuje slika 1. Opomba: Pregled nad opravljenimi testi je omogočen obema uporabniškima vlogama.



*Slika 1: shema procesov v IS GTO*

Vsaka od navedenih funkcionalnosti je podrobneje opisana v podpoglavjih.

#### Generiranje testov

Generiranje testov je proces, pri katerem je uporabniku z uporabniško vlogo učitelj ali mentor ali administrator omogočena funkcionalnost sestavljanja (generiranja) testov. Učitelj lahko samostojno ali sodelovalno z drugimi učitelji sestavi test iz nalog, ki jih pridobi iz zbirke nalog (s povezovanjem na IS UN; povezovanje je opisano v poglavju Seznam in opis izmenjave podatkov) ter jih po potrebi prilagodi z uporabo avtorskega orodja za gradnjo nalog, ki je sestavni del IS UN. Prav tako ima učitelj pregled nad generiranimi testi. Z generiranim testom lahko nadalje učitelj izvede elektronsko testiranje (preverjanje in ocenjevanje znanja) ali pa izvozi v PDF in natisne za izvedbo testiranja (preverjanja in ocenjevanja znanja) na papirju.

Učitelju so ponujene naslednje funkcionalnosti:

- upravljanje testov:
  - metapodatki o testu;
  - pregled nad generiranimi testi,
  - ustvarjanje novega testa iz:
    - baze predhodno ustvarjenih nalog – podrobneje opisano v nadaljevanju seznama.
    - dodajanje nove naloge v test – v iframe se odpre Avtorsko orodje za gradnjo nalog IS UN (opisano v poglavju Uporabniške in funkcionalne zahteve IS UN),
  - arhiviranje testov,
  - pregled seznama arhiviranih testov (iz arhiva se testi po 3 letih samodejno brišejo), možnost obnovitve posameznega testa.
  - določanje različic (variant) testa:
    - Za vsako nalogo v testu uporabnik določi N-število alternativnih vprašanj (nalog). Sistem samodejno tvori različne variante testa glede na zastopanost alternativnih variant nalog.
  - določanje vrstnega reda vprašanj z uporabo načina povleci in spusti;
  - vrednotenje (določanje točk ipd.) po posameznih vprašanjih.
- dodajanje nalog iz baze nalog:
  - omogoči se dodajanje nalog s povezovanjem na IS UN (opisano v poglavju Seznam in opis izmenjave podatkov) in vključitev vmesnika Avtorsko orodje za generiranje nalog (opisan v poglavju Uporabniške in funkcionalne zahteve IS UN)
  - dodajanje nabora nalog glede na raven izobraževanja, razred in predmet (šifrant predmetov bo priskrbel naročnik ob začetku izvajanja projekta) iz baze nalog IS UN,



- naključna izbira nalog in prenos uporabniško izbranega števila nalog iz nabora v test; (Ponujen nabor tipov nalog je odvisen od načina izvedbe testiranja, saj izvedba na papir ni možna za vse tipe nalog, podano v tabeli 1). Uporabnik v prejšnjem koraku določi nabor nalog, v tem koraku pa sistem samodejno naključno izbere naloge iz nabora za posamezen test. S tem omogočimo samodejno tvorjenje različic testov.
- objavljanje generiranih testov za urejanje:
  - javno – vidno vsem prijavljenim učiteljem;
  - skupinsko – izbor imen (elektronskih naslovov) uporabnikov učiteljev
  - zasebno – ni na voljo drugim uporabnikom.

Po uspešnem generiranju testa ima uporabnik z uporabniško vlogo učitelj možnost izvesti testiranje na dva načina:

- elektronsko – v tem primeru informacijska rešitev uporabnika vodi do procesa načrtovanja testiranja, ki je opisan v naslednjem poglavju.
- na papirju – v primeru, ko je uporabnik izbral možnost izvoza testov v pdf formatu in njihovo tiskanje. Pri tem se na vsako stran testa doda identifikator strani (niz alfanumeričnih znakov in QR koda), ki zakodirajo naslednje informacije:
  - identifikator učenca;
  - številka strani;
  - identifikator testa.
- Testiranje se izvede analogno na papir, ob koncu testiranja lahko učitelj ali učenec izvede proces uvoza natisnjenih testov, ki je opisan v poglavju Uvoz natisnjenih testov.

#### Načrtovanje testiranja (samo elektronsko)

Če želi učitelj izvesti elektronsko testiranje, mora pred pričetkom testiranja nastaviti parametre testiranja in dodeliti dostope. Proces načrtovanja testiranja tako vključuje:

- določanje parametrov testa:
  - metapodatki o testu (naslov, raven izobraževanja, razred, predmet, tema, licenca)
  - vidnost kriterija in vrednosti posamezne naloge med reševanjem (v odstotkih, točkah ipd.);
  - reševanje v zaprtem okolju (lock down browser ali odprti način izvedbe – uporaba drugih programov in orodij dovoljena)
- testno reševanje generiranih testov:
  - uporabniku je omogočena preverba pravilnega delovanja testa pred njegovo objavo za reševanje;
- objava generiranih testov za reševanje:
  - javno – vidno vsem prijavljenim uporabnikom (učenec in učitelj);
  - prednastavljeni skupini učencev;

- skupini učencev – izbor imen (elektronskih naslovov) uporabnikov učencev, ki jim je test namenjen.

Objava v reševanje se izvede na podlagi vnosa posameznih učencev ali razreda, ki mu je test namenjen; elektronski naslovi in imena za prikaz se pridobijo iz podatkov iz Arnes AAI. Izbranim uporabnikom (učencem) sistem pošlje elektronsko sporočilo s povezavo do vmesnika za reševanje testov. Objava v reševanje je omogočena tudi preko prikaza QR kode povezave s časovno omejenim trajanjem aktivnosti povezave ali alfanumerične kode.

Ko se učencem deli test v reševanje, lahko pričnejo s procesom "Izvedba testa", ki je opisan v nadaljevanju.

*Izvedba testa (v elektronski obliki)*

Učenec lahko izvede samopreverjanje z reševanjem javno objavljenih testov ali izvede testiranje, ki mu ga je namenil učitelj.

Učenec ima po prijavi dostop do prikaza seznama javno objavljenih testov.

Izvedba testa, ki mu ga je namenil učitelj, se začne, ko učenec klikne na povezavo za reševanje testa, vnese alfanumerično kodo testa ali skenira QR kodo.

Elektronska izvedba testov se lahko izvaja:

- v omejenem okolju (lock down browser – samo na osebnih računalnikih) na način, da uporabniku onemogoči uporabo drugih orodij in virov v povezavi s spletnimi brskalniki, ki so namenjeni takšni uporabi (v omejenem okolju);
- v neomejenem okolju, ki dovoljuje uporabo drugih orodij in programske opreme nameščene na računalniku.

Učenec ima po prijavi v spletni vmesnik za izvedbo testov na voljo naslednje funkcionalnosti:

- prikaz javno objavljenih testov; omogočeno iskanje po:
  - naslovu testa,
  - temi, ki jo test obravnava,
  - predmetu,
  - ravni izobraževanja,
  - razredu;
- prikaz testov, dodeljenih s strani učitelja;
- reševanje testov v zaklenjenem načinu (lock down browser);
- opozorilo pred oddajo o nerešenih nalogah;
- reševanje testov v odprtem načinu – dostop do drugih orodij ni omejen.

Način reševanja testov določi uporabnik z uporabniško vlogo učitelj v procesu generiranja testov (opisan v poglavju Generiranje testov).

*Ocenjevanje (v elektronski obliki)*

Proces ocenjevanja testov izvede uporabnik z uporabniško vlogo učitelj po preteku časa testiranja ali ko je posamezen test (vsaj en) oddan v ocenjevanje.

Učitelju so na voljo funkcionalnosti ocenjevanja in uporabniškega popravljanja samodejno generiranih ocen ter komentiranje s podajanjem povratne informacije (opisne in/ali številčne ocene) učencu, ki je reševal test.

Funkcionalnosti, ki so ponujene uporabniku, so naslednje:

- samodejno vrednotenje testov (seštevek točk po nalogah, delež uspešnosti), v kolikor to naloga omogoča (odvisno od tipa naloge – podprto za naloge zaprtega tipa);
- popravljanje samodejno določenih ocen iz prve alineje za vse tipe vprašanj;
- komentiranje dodeljene vrednosti naloge in/ali ocene celotnega testa s strani učitelja;

Ko je ocena objavljena, sistem pošlje elektronsko sporočilo učencu o novi povratni informaciji (komentarju) in povezavo do pregleda nad opravljenim testom, ki je opisan v nadaljevanju.

[Pregled nad opravljenimi testi \(v elektronski obliki\)](#)

Uporabnik z uporabniško vlogo učenec ima možnost pregleda nad opravljenimi testi. Učenec ima dostop do opravljenih testov in informacij o uspešnosti reševanj.

Učencu so v vmesniku na voljo naslednje funkcionalnosti:

- prikaz preteklih reševanj testov z uspešnostjo reševanja (oceno) in povratnimi informacijami s strani učitelja.

[Uvoz natisnjenih testov](#)

V primeru, ko učitelj želi izvesti testiranje na papirju, IS GTO ne izvede procesov načrtovanja testiranja, izvedbe testiranja in reševanja, ampak te izvede učenec na papir. Z uporabo procesa uvoza natisnjenih testov se tudi za te teste lahko izvede proces ocenjevanja znotraj IS GTO.

Uporabniku z uporabniško vlogo učitelj ali učenec proces omogoča uvoz skeniranih (rešenih) testov za kasnejše ocenjevanje. Po izvedenem postopku reševanja (na papirju):

- je uporabniku z uporabniško vlogo učitelj prikazan vmesnik za ocenjevanje. vmesnik implementira naslednje funkcionalnosti:
  - nalaganje datotek optično prebranih (skeniranih) rešenih testov in samodejno razvrščanje skenov pripadajočim učencem glede na identifikator (QR koda na vsakem listu);
  - možnost ocenjevanja – označevanja (anotiranja) skeniranega dokumenta – slike.
- je za uporabnika z uporabniško vlogo učenec implementirana spletni uporabniški vmesnik, ki po vpisu uporabnika v sistem IS GTO omogoča:
  - fotografiranje rešenih testov;
  - samodejna zaznava in obrezovanje posameznih listov;
  - določanje vrstnega reda ter združevanje listov v komplet ter
  - oddaja rešenih testov v ocenjevanje učitelju.

Po uspešnem uvozu rešitve testa, se ta shrani v sistem in učenca preko elektronske pošte obvesti o novem vnosu / povratni informaciji (oceni) s povezavo za pregled nad opravljenimi testi. Uporabniške in funkcionalne zahteve IS UN

Funkcionalne zahteve IS UN opisujejo, kaj mora IS UN omogočati uporabnikom. Poleg specifičnih funkcionalnih zahtev so opisane tudi splošne funkcionalne zahteve.

IS UN omogoča:

- upravljanje nalog, tj. shranjevanje, ustvarjanje in urejanje (z avtorskim orodjem), iskanje in dostop do nalog (baza nalog) in **s strani drugih aplikacij preko aplikacijskega programskega vmesnika**. Preko API so drugim IT rešitvam na voljo:
  - WebComponent za interaktivno vizualizacijo nalog za reševanje;
  - Iframe za ustvarjanje nalog znotraj druge IT rešitve – vdelan element (avtorsko orodje za izbran tip naloge) je na voljo uporabniku v drugi IT rešitvi.
- Grafični uporabniški vmesnik implementira naslednje funkcionalnosti:
  - ustvarjanje nalog z grafičnim uporabniškim vmesnikom, ki je ločen za vsakega od tipov nalog.
  - interaktivna vizualizacija nalog za reševanje;
  - preverjanje nalog - ocenjevanje;
  - upravljanje skupin mentorjev (glej poglavje Uporabniški nivoji IS GTO).

Naloge definiramo kot elemente, ki so sestavljeni iz:

- navodila (sestavljeno iz besedila in/ali slike in/ali videa in/ali zvočnega posnetka),
- interaktivnega dela za reševanje naloge,
- metapodatki naloge: ključne besede, razred, tema, avtor, sodelavci (soavtorji), oznaka »preverjeno«.

Sistem nudi podporo različnim nalogam. Vsaki od vrst nalog se prilagodi interaktivna vizualizacija nalog za reševanje in avtorsko orodje. Najprej so našteje vrste nalog, v nadaljevanju so za vsako vrsto nalog podane lastnosti avtorskega orodja in vizualizacije nalog za reševanje.

Sistem naj podpira naslednje tri tipe nalog:

1. Naloge **programiranja z delčki** so namenjene učenju programiranja na intuitiven in vizualen način, primeren za osnovnošolce in srednješolce. Uporablja sistem sestavljanja blokov, ki predstavljajo programske gradnike.
2. **Preračunane naloge** so matematične in logične naloge, pri katerih se rezultati samodejno generirajo na podlagi določenih parametrov. Omogoča prilagodljivost zahtevnosti in različne tipe nalog.
3. **Interaktivne naloge** so naloge, ki vključujejo interaktivne elemente. Vsaka od naštetih vrst nalog ima pripadajočo vizualizacijo in avtorsko orodje.

Za vsakega od tipov nalog se razvije svoje avtorsko orodje, ki se vključuje v IS GTO z uporabo iframe.

V nadaljevanju so opisani gradniki za:

- gradnjo nalog (avtorska orodja),
- vizualizacijo nalog (vizualizacija nalog za reševanje);
- preverjanje nalog ter
- upravljanje skupin mentorjev.

Funkcionalnosti vezane na gradnjo nalog (avtorska orodja)

Avtorska orodja so namenjena gradnji nalog. Vsako od avtorskih orodij implementira naslednje funkcionalnosti (velja za avtorska orodja za vse tipe nalog):

- določanje parametrov nalog:
  - obvezni parametri (pri vsakem od naštetih le ena izbira):
    - časovna omejitev (nastavljivo trajanje v sekundah/minutah),
    - težavnost naloge: lahka, srednja, zahtevna,
    - stopnja izobraževanja, razred in predmet
  - opcijski parametri:
    - več izbir: vsebinski sklop (tema), šifra učnega cilja
- samodejno določanje pravilnosti podanih predlogov za rešitve z dodanim pojasnilom o pravilnosti oziroma nepravilnosti rešene naloge
- podpora za naloge, ki vključujejo slike, avdio in video vsebine.

Glede na vrsto nalog so v podpoglavjih opisane funkcionalnosti vsakega od avtorskih orodij glede na tip naloge.

Prav tako so v teh poglavjih opisane funkcionalnosti, ki se vežejo na vizualizacijo nalog za reševanje.

Funkcionalnosti vezane na naloge za programiranje z delčki

Naloge za programiranje z delčki omogočajo tvorbo programov z izborom elementov iz programske knjižnice Blockly<sup>1</sup>. Imena elementov in parametri morajo biti poslovenjeni.

Avtorsko orodje za naloge za programiranje z delčki je sestavljeno iz gradnika za izbor podmnožice elementov Blockly in gradnika za določanje vizualnih elementov naloge.

Avtorsko orodje nalog programiranja z delčki ponuja uporabniku z uporabniško vlogo avtor naslednje funkcionalnosti:

- izbor nabora blokov naloge (nabor blokov naloge je podmnožica blokov knjižnice Blockly, ki jo izbere avtor naloge). Potrebna je prilagoditev na način, da so zagotovljeni slovenski prevodi za vse vsebovane gradnike.
- določanje vizualizacije elementov naloge z izbiro predlog in elementov iz knjižnic ozadij in elementov za
  - interaktivni element – lik (slika), ki se giba po eno ali dvo dimenzionalnem prostoru (mreži):

<sup>1</sup> <https://github.com/google/blockly>

- začetni in končni element – npr. hišica in cilj
- elementi na poti – dodatni elementi, ovire;
- teme / ozadja

Izvajalec pripravi 20 tem, začetnih in končnih elementov ter elementov na poti za vsako od vrst prikazov, skupaj najmanj 100 elementov. Izvajalec prilagodi le tiste elemente za različne vrste prikazov, za katere je to smiselno. Vrste prikazov so:

- labirint (2D, pot);
- sprehod junakov po točkah – junak mora obiskati vse točke (varianta, kjer je vrstni red vnaprej določen),
- 1D sprehod (naprej, nazaj).

Teme morajo biti pripravljene z vizualno bogatim ozadjem in določenimi polji interaktivnosti – ozadja nalog z opisom možnosti uporabe, ki sledijo zahtevanim funkcionalnostim.

Vizualizacija za reševanje nalog programiranja z delčki implementira naslednje:

- kontrolniki (naslednji korak, izvedi (celoto), na konec, na začetek) in prikaz napredka (animacija) za izvajanje po korakih in v enem kosu,
- kontrolniki za izvedbo preverjanja in animiran prikaz morajo delovati tudi v primeru nepravilno podane rešitve (programa).

Funkcionalnosti vezane na naloge za preračunane naloge

Preračunana naloga sestoji iz:

- vsebina naloge, ki vsebuje parametrizirane izraze;
- rezultata, ki vsebuje besedilo odgovora in več vnosnih polj za odgovore.

Avtorsko orodje za preračunane naloge implementira naslednje funkcionalnosti:

- generiranje nalog iz podanih (tudi kombinacije, kjer je smiselno):
  - matematičnih funkcij z aritmetičnimi operacijami (seštevanje, odštevanje, množenje, deljenje);
  - linearna funkcija, kvadratna funkcija, polinomske funkcije, racionalne funkcije, eksponentna funkcija, logaritemska funkcija, korenska funkcija, potenčna funkcija, krivulje drugega reda oz. stožnice;
  - osnovnih matematičnih funkcij (potence, kvadratni koreni, absolutna vrednost, odvod, integral, kompleksna števila);
  - sistemov linearnih enačb z eno, dvema in tremi neznankami.
  - algebrskih funkcij: razstavljanje in poenostavljanje izrazov, reševanje enačb in neenačb (linearna, kvadratna);
  - podpora numeričnim izračunom;
  - trigonometričnih funkcij: sinus, kosinus, tangens in kotangens, arc sin, arc tg, arc cos, arc ctg;
  - možnost dodajanja lastnih metod za izračunavanje.

- generiranje nalog, ki pokrivajo številske sisteme:
  - pretvorba med sistemi (dvojiški, šestnajstiški in desetiški);
  - operacije v različnih številskih sistemih.
- Generiranje nalog, ki vključujejo vektorje:
  - grafična interpretacija vektorja;
  - zapis vektorja;
- generiranje nalog, ki pokrivajo področje geometrije:
  - zapis grških črk, simbolov;
  - pravokotnost, vzporednost, leži na premici, ne leži na premici, skladnost, kot;
  - izračun ploščin, obsegov, površin in prostornin;
  - uporabo Pitagorovega izreka;
  - prikaz grafov geometrijskih funkcij v koordinatnem sistemu z uporabo knjižnice `jsxgraph`<sup>2</sup>.
- generiranje nalog, ki vključujejo kombinatoriko in verjetnost:
  - permutacije, kombinacije, variacije,
  - osnovne naloge verjetnosti,
  - logičnih izrazov.
- Generiranje nalog, ki pokrivajo področje statistike:
  - prikazovanje podatkov (stolpčni, pozicijski, tortni diagram, histogram, razsevni diagram, linijski in krivuljni diagram, škatla z brki),
  - aritmetična sredina, mediana, modus,
  - variacijski razmik, standardni odklon, medčetrtnski razmik.
- Generiranje nalog s področja množic:
  - element,
  - množica,
  - pripadnost elementa množici,
  - podmnožica,
  - prazna množica,
  - univerzalna množica,
  - vennov diagram,
  - presek, unija, razlika, komplement množic,
  - lastnosti operacij z množicami,
  - potenčna množica kartezični produkt množic,

---

<sup>2</sup> <https://jsxgraph.uni-bayreuth.de/wp/index.html>



- moč množice, moč potenčne množice.
- Generiranje nalog s področja osnovne logike:
  - izjave.

Orodje mora podpirati naloge:

- kjer je potrebno preračunavanje v več korakih;
- z več rešitvami – več odgovorov znotraj ene naloge (več vnosnih polj);
- parametrizirane naloge:
  - podpora za dinamične parametre, ki omogočajo več različic iste naloge z različnimi vrednostmi.
- ki predvidevajo vnos rešitve v obliki ulomka.
  - možnost nastavitve težavnosti preko števila spremenljivk ali kompleksnosti izrazov.

Zagotovljen mora biti prikaz matematičnih formul z uporabo JavaScript knjižnice.

Vizualizacija za reševanje preračunanih nalog naj pri kompleksnejših nalogah vključuje namige z napotki in koraki za reševanje

Funkcionalnosti vezane na interaktivne naloge

Avtorsko orodje za interaktivne naloge implementira naslednje funkcionalnosti gradnike s funkcionalnostmi:

- izbira vrste interaktivnega gradnika iz knjižnice. Vrste interaktivnih gradnikov so opisane v tabeli 1;
- podpora za dinamične parametre, ki omogočajo več različic iste naloge z različnimi vrednostmi, opisano v tabeli 1;
- možnost nastavitve težavnosti preko števila spremenljivk ali kompleksnosti izrazov. (Z nastavitvijo npr. števila spremenljivk lahko povečujejo oz. zmanjšajo težavnost naloge.)
- Možnost namiga pri vsakem tipu naloge, ki jo lahko avtor testa pripravi ali ne. Privzeto je možnost vklopljena.



Tabela 1: Tipi nalog s predvidenimi povratnimi informacijami, namigi in parametri

Tip naloge,	Kratek opis	Povratna informacija	Dinamični parametri, ki omogočajo več različic iste naloge z različnimi vrednostmi
Naloga z eno ali več izbirami (multiple choice with one answer, multiple choice with multiple answers)	vprašanja z enim ali več pravilnimi odgovori	Da, o pravilnosti vsakega od možnih odgovorov	Ne
Naloga s kratkim odgovorom v obliki besede ali besedne zveze (short answer)	Učenec odgovori z eno besedo ali besedno zvezo	Povratna informacija temelji na naboru možnih pravih odgovorov	Ne
Dopolni manjkajoča polja (fill in the blanks)	Učenec vpiše pravilni odgovor (besedo) v prazna polja med besedilom	Povratna informacija temelji na naboru možnih pravih odgovorov	Ne
Dopolni manjkajoča polja iz nabora možnih odgovorov (fill in the blanks)	Učenec med besedilom v praznih poljih v naboru možnih odgovorov, ki se pojavi v spustnem seznamu, izbere ustrezne besede ali besedne zveze ipd.	Da, o pravilnosti oziroma nepravilnosti vsakega od zahtevanih odgovorov	Ne
Označi v besedilu (mark sentence/mark the words)	Učenec označi različne črke, dele besed, besede, besedne zveze ali stavke v podanem besedilu, pri čemer uporablja različne barve ali	Da, o pravilnosti oziroma nepravilnosti vsakega od zahtevanih odgovorov	Ne



	nastavitve (npr. podčrta, okrepi)		
Povleci besede v besedilo (drag words in sentence)	Učenec povleče in spusti besedo v prazne prostorčke v besedilu	Da, o pravilnosti oziroma nepravilnosti vsakega od zahtevanih odgovorov	Ne
Popravljanje besed v besedilu (correct words in sentence)	Učenec popravi in/ali odstrani nepravilne črke/zloge/besede ... v podanem besedilu	Da, o pravilnosti oziroma nepravilnosti vsakega od zahtevanih odgovorov	Ne
Dopolni preglednico (fill in the table)	Učenec dopolni celice v tabeli	Da, o pravilnosti oziroma nepravilnosti vsakega od zahtevanih odgovorov	Ne
Preglednica z vprašanji (question table, question table with multiple answers)	Učenec za pripravljene trditve označi ustrezne odgovore (na primer Drži, Ne drži). Pri tem je mogočih v posamezni vrstici oziroma stolpcu tudi več pravih odgovorov	Da, o pravilnosti oziroma nepravilnosti vsakega od zahtevanih odgovorov	Da, kot dodatna možnost, ki jo lahko izkoristi ali pa ne (priporočeno pa je, da jo izkoristi)
Označevanje slike z vpisom besede (annotate picture)	Učenec vnese ustrezne besede, besedne zveze ali druge znake v prazne okvirčke na sliki	Da, o pravilnosti oziroma nepravilnosti vsakega od zahtevanih odgovorov	Ne
Označevanje slike s povleci in spusti (annotate picture, drag and drop)	Učenec povleče in spusti ustrezne besede, besedne zveze ali druge znake v prazne okvirčke na sliki	Da, o pravilnosti oziroma nepravilnosti vsakega od zahtevanih odgovorov	Ne
Najdi vročo točko (find the)	Učenec na sliki označi ustrezen	Da, o pravilnosti oziroma	Da, kot dodatna možnost, ki jo

hotspot/image hotspot)	del (na primer roko kot del telesa)	nepravilnosti vsakega od zahtevanih odgovorov	lahko izkoristi ali pa ne (priporočeno pa je, da jo izkoristi)
Tabla (whiteboard)	Učenec uporabi orodja za risanje (čopič, barva ...) in pisanje, da odgovori na vprašanje, na primer označi lokacijo na zemljevidu ali pobarva delček celote	Da, o pravilnosti vsakega od zahtevanih odgovorov	Ne
Razvrščanje besed, besednih zvez, stavkov in drugih delov besedila v vrstni red (word & sentence ordering)	Učenec s »primi in spusti« razvrsti besede, besedne zveze, stavke in druge dele besedila v ustrezno zaporedje, da sestavljajo smiselno celoto.  Na tak način lahko tudi sestavlja časovno zaporedje.	Da, o pravilnosti vsakega od zahtevanih odgovorov	Da, kot dodatna možnost, ki jo lahko izkoristi ali pa ne (priporočeno pa je, da jo izkoristi)
Časovni trak (timeline)	Učenec v okvirčke na časovnem traku povleče slike ali besede/besedne zveze iz predpripravljenega nabora slik oziroma besed/besednih zvez, ki ga pripravi avtor. Avtor lahko izbere različno število elementov na časovnem traku in vnese letnice	Da, o pravilnosti vsakega od zahtevanih odgovorov	Da

Križanka (crossword)	Križanka se sestavi na podlagi podanih besed in opisov	Da, o pravilnosti oziroma nepravilnosti vsakega od zahtevanih odgovorov, po možnosti tako, da se pravilne obarva v eni barvi, nepravilne v drugi barvi	Ne
Razvrščanje slik v ustrezno zaporedje (picture ordering)	Učenec s »primi in spusti« razvrstijo slike v ustrezno zaporedje – na primer časovno zaporedje, zaporedje po pomembnosti	Da, o pravilnosti oziroma nepravilnosti vsakega od zahtevanih odgovorov	Da, kot dodatna možnost, ki jo lahko izkoristi ali pa ne (priporočeno pa je, da jo izkoristi)
Povezovanje besed, slik, zvoka in besede/slike in podobno (text match, text-picture match, picture-picture match, audio-text match, audio-picture match)	Učenec s črto poveže ustrezne besede, besedne zveze, stavke oziroma druge dele besedila. Namesto besedila je lahko tudi slika, zvočni posnetek, video posnetek ...	Da, o pravilnosti oziroma nepravilnosti vsakega od zahtevanih odgovorov	Da, kot dodatna možnost, ki jo lahko izkoristi ali pa ne (priporočeno pa je, da jo izkoristi)
Razvrščanje besedila oziroma slik na ustrezno mesto, na primer v ustrezno skupino (text drag and drop, picture drag and drop)	Učenec potegne in spusti besede, stavke, slike ... v ustrezno skupino (na primer stolpec)	Da, o pravilnosti oziroma nepravilnosti vsakega od zahtevanih odgovorov	Da, kot dodatna možnost, ki jo lahko izkoristi ali pa ne (priporočeno pa je, da jo izkoristi)
Številski odgovor (numeric answer)	Učenec odgovori s številko, ulomkom, eksponentom, vektorjem ali matriko.	Da, o pravilnosti oziroma nepravilnosti zahtevanega odgovora	Da, kot dodatna možnost, ki jo lahko izkoristi ali pa ne (priporočeno pa je, da jo izkoristi)
Enačbe (Equations)	Učenec odgovori na matematična, fizikalna, kemijska in druga vprašanja z uporabo	Da, o pravilnosti oziroma nepravilnosti	Da, kot dodatna možnost, ki jo lahko izkoristi ali pa ne

	pojavnega okna s simboli enačb	zahtevanega odgovora	(priporočeno pa je, da jo izkoristi)
Uravnavaj kemijsko enačbo (Balance chemical equation)	Učenec uravnava dano kemijsko enačbo, izpolni tabele reaktantov in produktov in podobno	Da, o pravilnosti oziroma nepravilnosti zahtevanega odgovora	Da, kot dodatna možnost, ki jo lahko izkoristi ali pa ne (priporočeno pa je, da jo izkoristi)
Graditelj ionskih spojin (Ionic compound builder)	Učenec kombinira ione, da tvori molekulo, opisano z dano kemijsko formulo	Da, o pravilnosti oziroma nepravilnosti zahtevanega odgovora	Ne
Interaktivni videoposnetek	Vstavljjanje nalog v videoposnetek – izbirni odgovori, naloga s kratkim odgovorom v obliki besede/besedne zveze, dopolni manjkajoča polja, preglednica z vprašanji (drži – ne drži), razvrščanje besed, besednih zvez, stavkov in drugih delov besedila v vrstni red, razvrščanje slik v vrstni red, povezovanje. Naloge se lahko umesti na katerokoli mesto v posnetku, v posnetek pa se lahko naloži do 5 nalog.	Da, o pravilnosti oziroma nepravilnosti zahtevanega odgovora  Videoposnetek se nadaljuje, če je učenec odgovoril pravilno	Odvisno od naloge

Funkcionalnosti vezane na preverjanje nalog

IS GTO implementira grafični uporabniški vmesnik za preverjanje nalog. Dostop do vmesnika je dovoljen uporabnikom z uporabniško vlogo mentor.

Vmesnik omogoča uporabnikom z uporabniško vlogo mentor dostop do naslednjih funkcionalnosti:

- prikaz seznama oddanih nepreverjenih nalog:
  - omogočeno iskanje po metapodatkih nalog;



- filtriranje prikaza nalog po področjih.
- prevzem naloge v potrjevanje:
  - urejanje naloge – ustvari se nova različica naloge, mentor se doda kot kontributor.
  - Potrjevanje naloge – nalogi se doda oznaka »potrjeno«.

Funkcionalnosti vezane na upravljanje skupin mentorjev

IS GTO implementira grafični uporabniški vmesnik za upravljanje skupin mentorjev. Dostop do vmesnika je dovoljen uporabnikom z uporabniško vlogo administrator.

Vmesnik implementira naslednje funkcionalnosti:

- Dodaja ali odstrani mentorja za področje nalog;
- Dodaja ali odstrani področje nalog.

## Implementacijske zahteve

V nadaljevanju navajamo implementacijske zahteve in podamo visoko nivojsko arhitekturo celotnega sistema.

### Visokonivojska arhitektura sistema

Arhitektura sledi načelom za zagotovitev možnosti integracije z obstoječimi sistemi in kasnejše nadgradnje.

Visoko nivojsko arhitekturo IS GTO in IS UN prikazuje slika 2.

#### IS UN

IS UN sestoji iz avtorskih orodij za gradnjo posameznih tipov nalog (preračunane naloge, interaktivne naloge in naloge za programiranje z delčki). Naloga se, ko je definirana, izvozi v obliki WebComponent in shrani v zaledni sistem v bazo nalog. Vsako avtorsko orodje mora omogočati izvoz oblikovane naloge v format, ki ga definira izvajalno okolje in temelji na standardu WebComponents ter uporablja API izvajalnega okolja za shranjevanje in nalaganje podatkov.

Avtorska orodja so neodvisne aplikacije (v poljubni tehnologiji, ogrodju...), ki morajo omogočati kreiranje/spreminjanje/brisanje/kopiranje, shranjevanje, izvajanje (vizualizacija) nalog in preverjanje rezultatov. Omogočati morajo izvoz oblikovane naloge v standardizirani format, ki ga definira aplikacija IS GTO oz. izvajalno okolje. Avtorska orodja morajo omogočati tudi izvajanje nalog znotraj avtorskega orodja za preverbo pravilnosti prikaza nalog.

Grafični uporabniški vmesniki avtorskih orodij za gradnjo nalog se lahko uporabljajo iz drugih sistemov (IS GTO – funkcionalnost generiranja testov) z uporabo iframe.

#### IS GTO

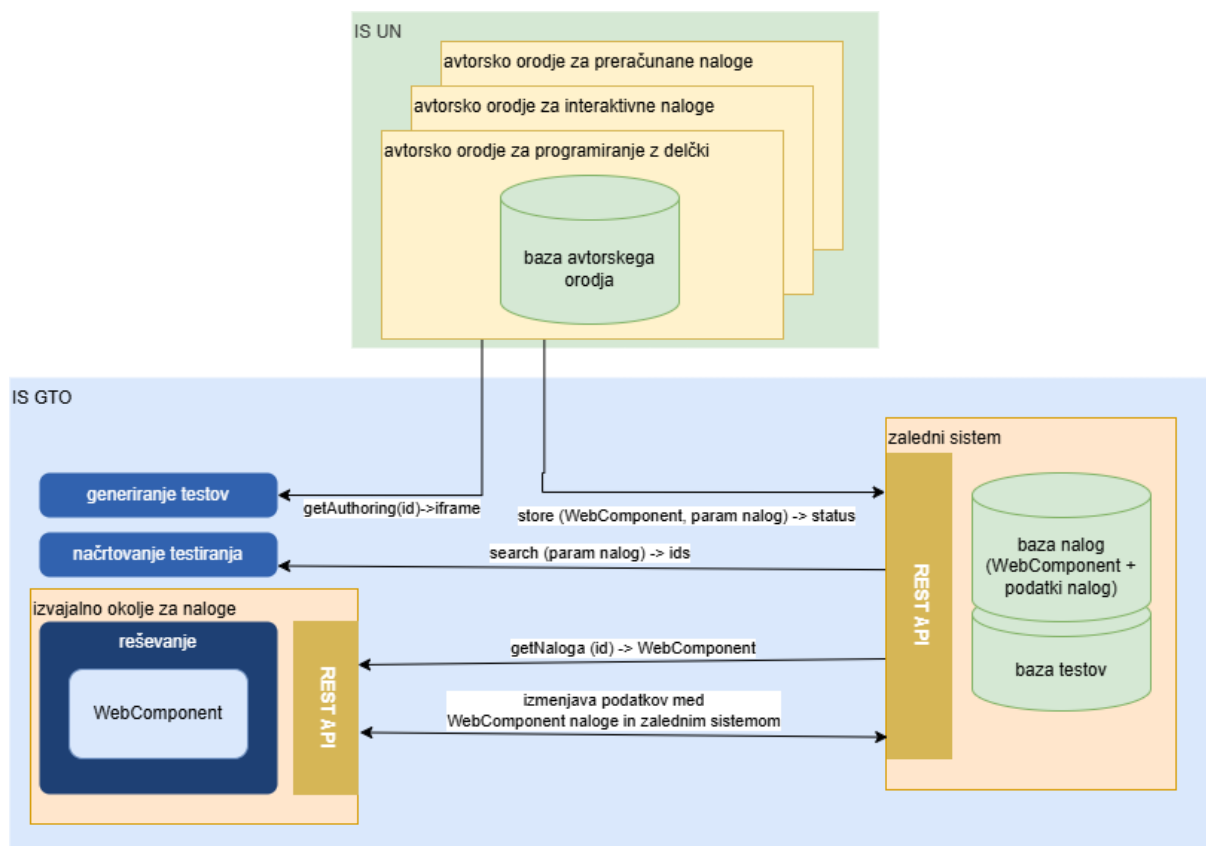
IS GTO shranjuje podatke o testih v bazo testov. Funkcionalnost načrtovanja testiranja dostopa do podatkov o testih iz zalednega sistema.

Pri reševanju se pridobi vizualizacija naloge za reševanje iz baze nalog. Ta je podana v obliki WebComponent in vključena v tej obliki v izvajalsko okolje za naloge. Naloge v obliki WebComponents uporabljajo API izvajalskega okolja za shranjevanje in nalaganje podatkov – baza nalog. Naloga izvajalca je, da definira API, ki ga bo izvajalsko okolje implementiralo. Izvajalec definira dodatne API klice, ki so potrebni za zagotavljanje funkcionalnosti, ki so vezane na reševanje nalog. Celotna naloga teče v brskalniku (JavaScript). Format definira tudi metapodatke, ki jih mora naloga določiti:

- ime naloge,
- tip naloge,
- URL avtorskega orodja,
- ključne besede,
- kategorija,
- avtor naloge in
- druge potrebne za zagotovitev vseh opredeljenih funkcionalnosti.

Izvajalno okolje implementira API, ki ga lahko uporabljajo WebComponente (naloge) kot so `storeObject(id naloge, object)`, `loadObject(id naloge)`. Prav tako pa mora izvajalno okolje definirati vmesnik, ki ga morajo implementirati vse

WebComponent (naloge): init, getScore in druge relevantne za zagotovitev specificiranih funkcionalnosti.



Slika 2: Visoko nivojska arhitektura IS GTO in IS UN

### Uporabnost

IS<sup>3</sup> mora zagotavljati enostavno (intuitivno) uporabo funkcionalnosti. Kjer koli je to izvedljivo, mora uporabniku ponuditi takojšen odziv (angl. real time response) in biti čim bolj prilagojen učinkoviti uporabi (najmanjše možno število korakov za izvedbo določenega postopka, čim hitrejši in enostaven dostop do kakovostnih informacij ...).

Ključna zahteva glede funkcionalnosti uporabniškega vmesnika je enostavnost uporabe, hitra odzivnost in podpora različnim brskalnikom. Med drugim mora uporabniški vmesnik slediti naslednjim smernicam:

- ustrezno segmentiranje in zaporedje vnosnih form, ki uporabniku dajeta jasno informacijo glede statusa in procesa zajema podatkov.
- uporaba samoizpolnjevanja ali izbire možnih vrednosti, kjer je to le mogoče.
- ponovna uporaba podatkov, ki so že zajeti v IS.
- pregledna zasnova strani, ki zagotavlja enostavnost in preglednost.
- dvosmerna komunikacija z uporabnikom, kjer so na enoten način uporabljeni elementi, kot so sporočila o napakah, obvestila o napredku obdelave in podobno.

<sup>3</sup> Vse navedbe IS v nadaljevanju, ki ne opredeljujejo natančneje zadevnega informacijskega sistema, se nanašajo na IS GTO in IS UN.





- elementi so oblikovani tako, da v različnih brskalnikih in operacijskih sistemih omogočajo podoben izgled in enakovredno funkcionalnost.

Uporabniški vmesnik IS za zajem podatkov je uporabniku prijazen in omogoča:

- kjer je to mogoče oziroma smiselno (in ne moti funkcionalnosti), oznake polj vsebujejo kratka navodila za vnašanje (npr. število znakov za geslo);
- polja, ki jih je potrebno obvezno izpolniti, se morajo na uporabniškem vmesniku jasno razlikovati od neobveznih polj, in sicer z drugačnim oblikovanjem;
- z namenom zmanjševanja napak pri vnosu podatkov se morajo ti v največji meri samodejno preverjati (npr. format zapisa podatka);
- polje z napačnim vnosom se vidno označi, ob napaki se izpišejo smiselna navodila;
- kjer je ustrezno, je stanje uporabnika v postopkih prikazano z grafičnim lokatorjem;

IS se vidno odziva na uporabnikove akcije. Vsaka akcija povzroči vidno (logično in prepoznavno) spremembo na uporabniškem vmesniku.

Ko orodje ni odzivno, uporabniku to vidno prikaže (sprememba kurzorja, prikaz poteka postopka ...).

Pomoč je uporabniku dosegljiva na mestu, kjer jo le-ta dejansko potrebuje (pomensko povedni naslovi polj/labele, namigi/tooltipi, baloni, pop-up/okna ...).

Velikost in podoba črk je prilagojena dobri vidljivosti in berljivosti:

- Sistem mora podpirati vnos, prikazovanje in tiskanje vseh znakov po standardnem naboru za lokalno okolje. Poleg znakov za lokalno okolje mora IS omogočati enostaven vnos (kombinacija tipk ali izbira z enim klikom iz nabora) znakov, ki se pojavljajo v lastnih imenih ali drugih podatkih iz drugih držav.
- Sistem mora imeti možnost povečevanja in pomanjševanja velikosti besedila.
- Sistem mora vsebovati funkcionalnosti za prilagoditev prikazane vsebine tiskanju.

Delo na vnosni formi je prilagojeno delu s tipkovnico:

- Standardno delovanje ključnih tipk.
- Smiselno prehajanje med polji (tipka Tab).
- Smiselno fokusiranje na ukazne gumbе (delovanje tipke Enter).
- Smiselna navigacija (tipke za listanje, puščice, Esc ...).

Pri izbirnih poljih z večjim številom (več kot 10) možnih izbir:

- je možno na ravni šifranta določiti pogoste izbire, ki se v izbirnem polju prikažejo na vrhu oziroma se uporabniku prikažejo samo pogoste izbire s klikom na »Prikaži vse« pa lahko prikaže vse izbire;

- se namesto posamezne ključne besede prikaže več podatkov, ki uporabniku olajšajo izbiro med različnimi podobnimi možnostmi.

#### Razpoložljivost

Razpoložljivost in nekatere ostale karakteristike IS so zahtevane glede na zahtevano razpoložljivost posameznih funkcionalnosti sistema. Funkcionalnosti, namenjene uporabnikom, morajo biti razpoložljive ob delavnikih znotraj delovnega časa oziroma od 6:00 do 19:00. Funkcionalnosti IS morajo biti v času razpoložljivosti na voljo v 99,9 % glede na letno raven. V okviru enega meseca je dopustnih največ osem (8) ur izpada, na letnem nivoju je dopustnih največ 24 ur izpada. Posamezen izpad v delovnem času naročnika ne sme biti daljši od štirih (4) ur.

Redne posodobitve in tehnične izboljšave morajo biti praviloma izvedene tako, da to ne pomeni nedelovanja IS znotraj opredeljenega delovnega časa naročnika. Če se nedelovanju ni mogoče izogniti, mora biti vsaka načrtovana prekinitve delovanja (zaradi npr. nujnih popravkov programske ali strojne opreme, nadgradenj in drugih nujnih del) naročniku sporočena najmanj tri (3) dni pred dejansko prekinitvijo. V sporočilu mora biti naveden razlog za prekinitve ter čas, v katerem bo izvedena prekinitve. Načrtovana prekinitve delovanja IS se lahko izvede izven delovnega časa naročnika oz. skladno z delovnim procesom upravljavca infrastrukture IS, kar dogovorita izvajalec in naročnik.

Zgornje zahteve se nanašajo na razpoložljivost IS in storitev, ki tečejo v okviru IS, ne pa tudi na zunanje sisteme, s katerimi se IS integrira ali povezuje.

Predvideva se sklenitev posebnega protokola med naročnikom in upravljavcem infrastrukture IS, v katerem bodo natančno določeni pogoji zagotavljanja virov, kapacitet, odzivnosti, razpoložljivosti in podobno.

#### Zmogljivost

IS mora biti zasnovan tako, da bo brez težav sposoben:

- zagotavljati nemoteno delovanje in izvajanje funkcionalnosti vsem uporabnikom IS;
- zagotavljati nemoteno izmenjavo podatkov z zunanjimi informacijskimi sistemi, kot je opredeljeno v funkcionalnih zahtevah;
- odzivni čas uporabniškega vmesnika IS za zajem podatkov oz. čas od zahtevka brskalnika http(s) do celotnega prikaza strani, tudi ob maksimalni predvideni obremenitvi IS, ne sme biti nad 1 sekundo;
- odzivni čas uporabniškega vmesnika IS za pregled podatkov in prikaz poročil bo natančno opredeljen v okviru aktivnosti analize in specifikacije zahtev končne rešitve. Zahteva naročnika je, da se v okviru aktivnosti analize in specifikacije zahtev opredeli in uskladi tipe vpogleda v podatke in tipe poročil, način njihove priprave (preko spletnega vmesnika, asinhrona priprava poročil v ozadju, predpriprava poročil itd.) ter zahtevane odzivne čase za posamezno kategorijo. Pri tem mora izvajalec zagotoviti takšne odzivne čase za pripravo vpogledov v podatke in poročila, da bodo omogočali zadovoljivo uporabniško izkušnjo.

V primeru, da izvajalec zahtevanih odzivnih časov ne more doseči, mora identificirati vzrok, ki ga bo obravnaval skupaj z naročnikom. V primeru, da vzrok za težave z zmogljivostjo (performančne težave) ne izvira iz drugih sistemov (ki se npr. odzivajo prepočasi pri izmenjavi podatkov itd.), bo moral le-tega odpraviti.

#### Nadgradljivost in skalabilnost

IS mora biti zasnovan na način, ki bo omogočal enostavno (tehnološko nezahtevno) in hitro izvajanje nadgradenj sistema. Arhitekturna in tehnična zasnova morata omogočati dovolj enostavno dodajanje novih sklopov funkcionalnosti, modulov oziroma rešitev, ki bi izhajale iz naslova novih potreb oziroma zahtev naročnika.

**IS GTO** mora izpolnjevati naslednje pogoje:

- uporablja se le baza PostgreSQL;
- baza PostgreSQL teče na ločeni virtualki, poleg "write" instance (leader) je lahko nameščenih več "read" instanc (follower);
- implementiran mora biti na način, da deluje kot **stateless** sistem, kar pomeni:
  - **ne shranjuje stanja na strežniku med zahtevki uporabnikov.** Vsaka zahteva mora vsebovati vse potrebne podatke za obdelavo, ne da bi se zanašala na prejšnje interakcije;
  - **seje uporabnikov ne smejo biti shranjene na strežniku.** Uporaba sej (session storage na strežniku) ni dovoljena. Namesto tega se lahko uporabljajo žetoni (tokens), kot so **JWT (JSON Web Token)**, ki se prenašajo med zahtevki;
  - **obdelava podatkov mora biti neodvisna od posameznega strežnika.** Če se aplikacija izvaja v porazdeljenem okolju, mora biti vsaka instanca sposobna obdelati katero koli zahtevo ne glede na prejšnje zahteve istega uporabnika;
  - **za shranjevanje stanja naj se uporablja zunanji persistentni sistem.** Podatki naj se hranijo v **bazi podatkov, objektnih shrambah ali cache sistemih** (npr. Redis) in ne v pomnilniku aplikacijskega strežnika.
- mora imeti možnost konfiguriranja URL za
  - bazo za pisanje ("write") in
  - bazo za branje ("read"),s čimer je omogočena skalabilnost metod branja in pisanja v baze.
- Mora omogočati preverjanje zdravja posameznih modulov na url-ju (npr. /health);
- Implementiran mora biti kot kontejnerizirana rešitev, kar pomeni, da:
  - mora biti pakirana in distribuirana v obliki **kontejnerjev** (npr. **Docker**);
  - vsaka komponenta sistema mora biti lahko ločeno kontejnerizirana in neodvisna;

- vsebovati mora navodila za postavitve kontejnerskega okolja (odprtokodno) (npr. Rancher community) na vsaj treh virtualnih strežnikih skupaj z registrom slik kontejnerjev za posamzeno aplikacijo (image).

Oba IS morata biti izdana v GitLab naročnika pod licenco MIT. Vsebovati mora navodila in procedure za prevajanje, pripravo paketa (image) in proceduro za avtomatsko namestitve.

#### Priprava grafične podobe

Poglavje opisuje način priprave grafične podobe za aplikacije, ki so predmet tega javnega naročila.

Cilj je zagotoviti enotno, privlačno in uporabniku prijazno vizualno izkušnjo, ki bo ustrezala identiteti naročnika.

Grafična podoba IT rešitev mora biti usklajena s celostno grafično podobo projekta eTorba.

Zajema naj pojavnost logotipov skladno s priročnikom celostne grafične podobe NOO: [https://www.gov.si/assets/organi-v-sestavi/URSOO/Logotipi/Prirocnik\\_CGP\\_NOO\\_maj\\_2022.pdf](https://www.gov.si/assets/organi-v-sestavi/URSOO/Logotipi/Prirocnik_CGP_NOO_maj_2022.pdf)

Vizualna oblikovna rešitev mora biti uporabna na osebnih računalnikih in mobilnih napravah. Temeljne razpoznavne prvine celostne grafične podobe naj bodo zastavljene tako, da dopuščajo uporabo s temno in s svetlo podlago.

#### Specifikacije grafične podobe:

- Razvoj ene enotne barvne palete, ki bo uporabljena v vseh aplikacijah. Barvna shema mora vključevati osnovne in dodatne barve, ki bodo skladne z vizualno identiteto naročnika.
- Definiranje enotne tipografije za vse aplikacije, vključno z izborom pisav za naslove, podnaslove, besedila in druge elemente.
- Oblikovanje osnovnega izgleda logotipov za posamezno aplikacijo. Logotipi naj bodo medsebojno skladni in naj se razlikujejo po specifičnih detajlih, ki bodo ločili aplikacije med seboj, vendar bodo vseeno ohranili skupno vizualno identiteto.
- Izbor osnovnih grafičnih elementov, kot so ikone in vzorci, ki bodo konsistentni z barvno paleto in tipografijo ter primerni za uporabo v vseh aplikacijah.

#### Varnost

Komponente IS, ki so predmet javnega naročila, morajo biti izdelane z upoštevanjem vseh dobrih praks, ki zagotavljajo zahtevano stopnjo informacijske varnosti.

IS mora izpolnjevati naslednje varnostne zahteve:

- Spletni vmesnik mora biti odporen na penetracijske napade informacijskih sistemov (kot npr. SQL injection, XSS (cross site scripting), file inclusion, error handling, URL parameter manipulation, buffer overflow ...);
- spletni vmesnik mora imeti vgrajen mehanizem za odjavo uporabnikov v primeru poteka seje – v primeru daljše neaktivnosti IS zahteva ponovno prijavo uporabnika.

## Metodološke zahteve

---

### Vodenje projekta na strani izvajalca

Tekom izvajanja aktivnosti, ki so predmet javnega naročila, so predvideni koordinacijski sestanki, ki jih bo vodil vodja projekta na strani naročnika z namenom razreševanja odprtih vsebinskih vprašanj in podajanja pojasnil oziroma usmerjanja pri pripravi izdelkov. Vodja projekta na strani izvajalca se je dolžan redno udeleževati planiranih koordinacijskih sestankov in pripraviti zapisnik dogovorov sestanka. Na zahtevo naročnika ali izvajalca se po potrebi koordinacijskih sestankov udeležijo tudi ostali člani projektne skupine.

## Zahteve glede izvedbe ključnih aktivnosti

---

### Projekt za izvedbo (PZI)

Dokument Projekt za Izvedbo (v nadaljevanju: PZI) je namenjen natančnemu popisu in specifikacijam bodočega informacijskega sistema.

Dokument PZI mora vsebovati najmanj:

- funkcionalne specifikacije in funkcionalno dekompozicijo,
- seznam poslovnih procesov z opisi,
- seznam gradnikov z opisi,
- specifikacije podatkovnih struktur,
- specifikacijo struktur XML,
- specifikacije spletnih storitev,
- specifikacijo aplikacije za prikaz podrobnosti delovanja vseh vključenih komponent,
- arhitekturo sistema za implementacijo (predlog uporabe vzorcev, topologija strežnikov, uporabljeni tehnološki standardi, tehnologija podatkovnih zbirk),
- načrt spletnih mest (žični okvir) s specifikacijami vsebin in funkcionalnostmi za uporabnike (npr. katere informacije vsebuje, kako naj izgleda uporabniški vmesnik, kakšno uporabniško okolje bo ponujeno uporabniku, kako bodo organizirane vsebine, kakšne oblike iskanj bodo omogočene, do katere mere je mogoče doreči politiko umeščanja novih vsebin v portal, da se še zagotavlja preglednost),
- varnostne in zaščitne mehanizme,
- navedene in natančno popisane predvidene integracije z zunanjimi sistemi,
- terminski načrt.

Za IS mora izvajalec pripraviti podrobno specifikacijo zahtev, s katerimi bodo posledično dosežene višja kakovost, višja učinkovitost in izboljšana koordinacija v razvojnem procesu ter boljši pregled nad napredkom procesa razvoja.

Kot podlago za pripravo specifikacije zahtev mora izvajalec preučiti razpisno dokumentacijo in druge dokumente, ki mu jih preda naročnik.

V okviru specifikacije se pričakuje naslednje:

- pripravljena je v slovenskem jeziku;
- uporabljana terminologija bo prilagojena naročniku;
- izvajalec mora spoznati naročnikovo poslovno področje;
- v specifikaciji zahtev uporabljene diagramске in druge tehnike so primerno pojasnjene;
- specifikacija zahtev vključuje zahteve glede uporabnosti rešitve.

Če izvajalec oceni, da drugače ne bo mogel zajeti naročnikovih zahtev za IS, mora pri pripravi specifikacije uporabiti prototipe.

Specifikacija mora slediti nekaterim osnovnim načelom:

- ustreznost: vsaka izmed podanih zahtev v specifikaciji mora biti relevantna za razvoj in vzpostavitev IS;
- jasnost: v specifikaciji podane zahteve morajo biti zapisane na tak način, da jih vsi udeleženci enako interpretirajo;
- skladnost: podane zahteve si med seboj ne nasprotujejo, temveč se medsebojno dopolnjujejo;
- celovitost:
  - nabor podanih zahtev mora opisovati vse pomembne funkcije, zmogljivosti, lastnosti in omejitve IS;
  - nabor zahtev opisuje vse odzive IS na vse možne (veljavne in neveljavne) vhodne podatke oziroma vrste podatkov;
- jedrnatost: vsaka od zahtev mora biti navedena samo na enem mestu. Opisi zahtev morajo vsebovati samo pomembna določila in ne smejo biti preobsežni;
- pomembnost: iz specifikacije morajo biti jasno razvidne izvedbene prioritete;
- preverljivost: vse v specifikaciji podane zahteve mora biti možno preveriti in potrditi;
- sledljivost: za vsako od podanih zahtev mora biti znan vir le-te.

Izvajalec mora zajeti:

- funkcionalne zahteve,
- nefunkcionalne zahteve (zmogljivost, varnost ...),
- implementacijske zahteve.

Analizirane zahteve mora izvajalec primerjati z zahtevami, podanimi v tem dokumentu ter ostali predani dokumentaciji, in jih po potrebi uskladiti z naročnikom in uporabniki.

Izvajalec mora podati končen predlog podrobne specifikacije IS, jo nato uskladiti z naročnikom ter pridobiti njegovo potrditev specifikacije zahtev.

#### Testiranje in zagotavljanje kakovosti

Nove verzije/popravki se najprej namestijo na testno okolje.

Razvojno okolje in testno okolje za razvojno testiranje mora vzpostaviti izbrani izvajalec na svoji infrastrukturi. Prav tako izvajalec zagotovi uvoz podatkov za testiranje in šolanje testnih uporabnikov v testnem okolju kot tudi pripravo okolja za izvedbo šolanja uporabnikov.

Osebe naročnika mora biti neposredno soudeleženo pri testiranju funkcionalnosti IS. Izvajalec za potrebe izvedbe prevzemnega testiranja s strani naročnika zagotovi orodje za prijavo in spremljanje napak (orodje mora biti uporabniku prijazno, omogočati mora opis napake, vstavljanje slik zaslonских mask in pripenjanje dokumentov, kjer se je napaka pojavila ter spremljanje statusa odpravljanja evidentirane napake).

#### Poročilo o testiranju

Poročilo o testiranju mora vsebovati spodaj opisane podatke:

- splošni podatki o testiranju:
  - oznaka in verzija izdelka, ki je bil predmet testiranja,
  - konfiguracija strojne in programske opreme, ki je bila uporabljena pri izvedbi testiranja,
  - navedena ali pripeta dokumentacija, ki je bila uporabljena kot podlaga za izvedbo testiranja;
- statistika ugotovljenih neskladnosti:
  - za vsako testiranje je treba navesti:
    - oznako, ali je bila npr. dostavljena dokumentacija programske podpore ustrezna/neustrezna,
    - grobo oceno poslovne podpore (npr. ustrezna/delno ustrezna/neustrezna),
    - pregled ugotovljenih napak pri izvedbi testiranja,
    - koliko napak je bilo odpravljenih in koliko je takih, ki so še v reševanju;
- rezultati testiranja:
  - zahteve in priporočila v zvezi z dokumentacijo:
    - opisno opredeljeno zadovoljstvo glede pridobljene dokumentacije,
    - pričakovanja v zvezi z morebitno dopolnitvijo dokumentacije s strani naročnika ali izvajalca;
  - testiranje funkcionalnosti IS ter podatkov:
    - opisi rezultatov testiranja,
    - plan v zvezi z odpravo napak;
- zaključna ocena:
  - zaključna ocena v zvezi s potekom in rezultati testiranja,



- ob identifikaciji tveganj se opredeli, ali je izdelek primeren za uvedbo v produkcijsko okolje ali ne.

#### Tehnična in uporabniška dokumentacija

Tako uporabniška kot tehnična dokumentacija oziroma navodila za skrbnike rešitve morajo biti pripravljena v slovenskem jeziku.

#### Uporabniška dokumentacija

Izvajalec mora v okviru razvoja pripraviti uporabniško dokumentacijo oziroma navodila/priročnik za uporabnike IS. Uporabniška dokumentacija mora vsebovati vsaj naslednje elemente:

- osnovne podatke o sistemu:
  - namen sistema in njegove glavne funkcionalnosti,
  - vrste uporabnikov, ki jim je dokumentacija namenjena in
  - kontaktna oseba za pomoč pri morebitnih nejasnostih glede uporabe sistema;
- opis uporabniškega vmesnika mora pokriti sledeče tematike:
  - Prikaz uporabniškega vmesnika, poimenovanje posameznih delov uporabniškega vmesnika in kratko pojasnilo, čemu posamezen element služi;
  - opis menijev (glavni, pomožni ...), navedba postavk v menijih in opis akcije, ki se s to postavko izvede;
  - opis vseh orodnih vrstic;
  - opis vsebinskega dela (okna) uporabniškega vmesnika, kjer se prikazujejo podatki.
- uporabo aplikacije: podroben opis uporabe aplikacije za posamezna vsebinska področja in postopke znotraj njih v obliki videa:
  - videonavodila naj vsebujejo prikaz uporabe ključnih funkcionalnosti, kot so: vpis uporabnika, generiranje testov, tvorjenje nalog, izvedba testiranja. Videonavodila naj bodo primerna uporabniški skupini (na primer za učenca ustrezna raven zahtevnosti uporabljenega besedišča, privlačnost).

#### Usposabljanje uporabnikov

V okviru aktivnosti usposabljanja mora izvajalec usposobiti uporabnike naročnika za uporabo sistema. Usposabljanja morajo biti izvedena v obliki delavnic, kjer slušatelji preko uporabe razvitega IS izvajajo praktične postopke.

Po izvedenem usposabljanju morajo biti udeleženci usposabljanja, ne glede na tip uporabnika, sposobni samostojno uporabljati posamezne funkcionalnosti sistema, ki so bile predmet usposabljanja.