Tehnične specifikacije

KAZALO VSEBINA

1 Uvod in izhodišča 4
2 Zahteve tehničnih specifikacij 4
2.1 Uporabnost 4
2.2 Razpoložljivost 5
2.2.1 Zagotavljanje razpoložljivosti 5
2.2.2 Redundantna infrastruktura 6
2.2.3 Varnostne kopije in obnova sistema 7
2.2.4 Neodvisnost razpoložljivosti 7
2.3 Zanesljivost 7
2.3.1 Zahteve za visoko stopnjo zanesljivosti sistema 8
2.3.2 Samodejno obnavljanje sistema po napakah 8
2.3.3 Nadzor delovanja in zaznavanje napak 9
2.3.4 Zanesljivost v kontekstu uporabniške izkušnje 9
2.4 Razširljivost 10
2.4.1 Zahteve za skalabilnost sistema 10
2.4.2 Horizontalno skaliranje 10
2.4.3 Vertikalno skaliranje 10
2.4.4 Razširljivost v kontekstu integracij 11
2.5 Zmogljivost 11
2.5.1 Odzivnost in sočasna obdelava zahtevkov 11
2.5.2 Arhitekturni pristopi za zagotavljanje zmogljivosti 12
2.5.3 Nadzor in prilagajanje zmogljivosti 12
2.6 Uporabniška izkušnja 12
2.7 Interoperabilnost 13
2.7.1 Zahteve za integracijo z drugimi informacijskimi sistemi 14
2.7.2 Standardizirani protokoli in formati za izmenjavo podatkov 14
2.7.3 Nadzor in obvladovanje integracij 14
2.7.4 Pripravljenost na nadaljnje razširitve 14
2.8 Vzdržljivost 15
2.8.1 Vzdržljivost aplikacijskega in podatkovnega sloja 15
2.8.2 Vzdržljivost poslovnih procesov 15
2.8.3 Vzdržljivost ob nadgradnjah in spremembah 15
2.8.4 Vzdržljivost integracij 15
2.9 Prilagodljivost 16
2.9.1 Prilagodljivost funkcionalnosti in poslovnih pravil 16
2.9.2 Prilagodljivost uporabniških vlog in dostopov 16
2.9.3 Prilagodljivost vsebin in podatkovnih struktur 16
2.9.4 Prilagodljivost infrastrukturnega okolja 16

2.10 Varnost ... 17
2.10.1 Avtentikacija in avtorizacija ... 17
2.10.2 Varovanje podatkov ... 17
2.10.3 Revizijska sled in sledljivost ... 17
2.10.4 Varnost integracij ... 18
2.10.5 Operativna varnost in vzdrževanje ... 18
2.10.6 Upravljanje piškotkov (Cookie management) ... 18
2.11 Enoten sistem prijave s preostalim delom NIST (Single sign-in) ... 19
2.11.1 Obstoječe stanje avtentikacije in avtorizacije v NiST ... 19
2.11.2 Zahteve in izhodišča za morebitno nadgradnjo ... 19
2.11.3 Pristop v okviru bodoče rešitve Slovenia Green ... 19
2.12 Arhitekturna zasnova rešitve Slovenia Green in njena povezljivost z NiST ... 20
2.13 Uporaba umetne inteligence (AI) ... 20
2.14 Primer minimalne konfiguracije sistema ... 21
2.15 Osnovne zahteve za podatkovni center ... 22
3 Integracije ... 23
3.1 Seznam integracij bodoče rešitve ... 23
3.1.1 Integracije znotraj ekosistema NiST ... 24
3.1.2 Integracija sistema za upravljanje uporabnikov (SSO) ... 24
3.1.3 Integracije za obdelavo in hrambo dokumentov ... 24
3.1.4 Integracije za analitiko in poročanje ... 24
3.1.5 Integracije z zunanjimi informacijskimi viri ... 24
3.1.6 Integracija s sistemom HubSpot (CRM) ... 25
3.1.7 Pridobivanje podatkov o registriranih naravnih lepotah ... 25
3.1.8 Pridobivanje podatkov o stopnji čiščenja odpadnih voda ... 26
3.1.9 Pridobivanje podatkov o dostopnosti z javnim potniškim prometom ... 28
3.1.10 Podatki snovne in nesnovne kulturne dediščine ... 29
3.1.11 Preverjanje zadovoljstva prebivalcev ... 32
3.2 Tehnična osnova integracij ... 34
3.2.1 Integracija z informacijskim sistemom NiST ... 34
3.2.2 Integracija za upravljanje uporabnikov (priprava na SSO) ... 35
3.2.3 Integracija za hrambo in upravljanje dokumentov ... 35
3.2.4 Integracije za analitiko in poročanje ... 35
3.2.5 Splošna načela tehnične izvedbe integracij ... 36
3.2.6 Prenos podatkov iz NiST v rešitev SG – nabor podatkov in API klici ... 36
4 Opis zahtev glede arhitekture in tehničnih značilnosti rešitve ... 39
4.1 Arhitekturna zasnova rešitve ... 39
4.2 Tehnološka izhodišča in uporaba odprtokodnih orodij ... 39
4.3 Uporabniški vmesnik in aplikacijski sloj ... 40
4.4 Podatkovni sloj in upravljanje podatkov ... 40

4.5 Integracijski sloj in tehnična povezljivost 40
4.6 API gateway in objava odprtih podatkov 41
4.7 Oblačna infrastruktura in gostovanje 42
4.8 Kontejnerizacija, prenosljivost in obvladljivost 42
4.9 Kompatibilnost brez prisilne združitve sistemov 42
5 Osnutek ER diagrama 43
5.1 Seznam ključnih entitet 43
5.2 Relacije med ključnimi entitetami – opis povezav in njihove vloge 46
5.3 ER diagram 50
6 Načrt migracije podatkov 52
6.1 Pregled virov podatkov 52
6.2 Migracija podatkov iz obdobja pred letom 2021 52
6.2.1 Tehnični pristop k obdelavi in ekstrakciji podatkov 53
6.2.2 Preslikava v ciljni podatkovni model (ER) 53
6.2.3 Načelo ponovljivosti in tehnična izvedba vnosa 53
6.2.4 Namen in omejitve uporabe zgodovinskih podatkov 54
6.2.5 Primer ekstrakcije in vnosa zgodovinske presoje (ilustrativni primer) 54
6.3 Migracija podatkov iz obstoječe aplikacije (presoje po letu 2021) 56
6.3.1 Strategija migracije podatkov 57
6.3.2 Tehnični pristop k prenosu podatkov 57
6.3.3 Preslikava v ciljni podatkovni model (ER) 57
6.3.4 Primer strukturiranega izvoza (ilustrativni primer) 58
6.3.5 Primer vzorčnega SQL vnosa (ilustrativni primer) 59
6.3.6 Ponovljivost in končni prenos 59

Kazalo slik

Slika 1: Izbrani parametri iskanja 27
Slika 2: Rezultat iskanja 27
Slika 3: Pridobivanje podatkov za občino Ajdovščina 28
Slika 4: Register nepremične kulturne dediščine 29
Slika 5: Seznam nepremične kulturne dediščine 30
Slika 6: Rezultat iskanja nepremične kulturne dediščine 30
Slika 7: Register nesnovne kulturne dediščine 31
Slika 8: Seznam nesnovne kulturne dediščine 31
Slika 9: Rezultat iskanja nesnovne kulturne dediščine 32
Slika 10: ER diagram izdelan v dbdiagram.io 51

1 Uvod in izhodišča

Dokument predstavlja tehnične specifikacije bodoče rešitve za informacijsko vključitev standarda *Slovenia Green* v okviru Nacionalnega turističnega informacijskega središča (NiST) in temelji na naboru vhodnih gradiv in spoznanj, pridobljenih v predhodnih fazah projekta.

Namen tehnične specifikacije je opredeliti arhitekturna, tehnološka in infrastrukturna izhodišča, ki bodo omogočala učinkovito implementacijo rešitve, hkrati pa zagotavljala dolgoročno vzdržnost, razširljivost in obvladljivost sistema skozi njegov celoten življenjski cikel.

Izhodišče tehnične analize je dejstvo, da je bodoča rešitev vsebinsko, podatkovno in procesno povezana z ekosistemom NiST. Rešitev se navezuje na obstoječe podatkovne vire, klasifikacije, uporabniške vloge, analitične pristope, poročanje ter obstoječo oblačno infrastrukturo in uporabljena orodja NiST. Ta povezanost predstavlja pomembno dodano vrednost projekta, saj omogoča uporabo že vzpostavljenih standardov, podatkovnih tokov in operativnih praks.

Hkrati pa je pri tehnični zasnovi nujno jasno razlikovati med **povezanostjo** in **odvisnostjo** sistemov. Rešitev *Slovenia Green* predstavlja specializiran informacijski sistem z lastno poslovno logiko, procesi, podatkovnimi modeli ter uporabniškimi skupinami, ki se po namenu in funkcionalnem obsegu bistveno razlikujejo od osrednjih funkcionalnosti sistema NiST. Posledično tehnične zahteve izhajajo iz načela, da mora biti rešitev zasnovana kot **samostojen in neodvisno razvit sistem**, ki pa je hkrati **pripravljen na povezovanje in integracijo** z NiST ter drugimi sorodnimi sistemi.

Takšen pristop omogoča:

- neodvisno načrtovanje in izvajanje razvoja ter nadgradenj rešitve *Slovenia Green*,
- zmanjševanje tehničnih, organizacijskih in časovnih tveganj, ki izhajajo iz tesno povezanega razvoja več sistemov,
- jasnejšo razmejitev odgovornosti in lastništva posameznih funkcionalnih in podatkovnih sklopov,
- večjo fleksibilnost pri nadaljnjih tehnoloških in arhitekturnih odločitvah.

Kot osrednje vodilo je uporabljeno načelo **ohlapno sklopljenih, a dobro integriranih sistemov** (*loosely coupled, well-integrated systems*). Skupna področja, kot so uporabniki, enoten sistem prijave, destinacije, ponudniki ter izmenjava podatkov, so obravnavana na način, ki omogoča deljenje in povezovanje podatkov, ne pa vzpostavljanja neposrednih izvedbenih ali razvojnih odvisnosti med sistemi.

Tehnična zasnova mora omogočati, da je sistem možno namestiti v različnih infrastrukturnih okoljih, neodvisen od specifičnih nastavitev posamezne oblačne platforme ter primeren za sodobne pristope razvoja, uvajanja in vzdrževanja informacijskih sistemov.

2 Zahteve tehničnih specifikacij

V nadaljevanju so opisane posamezne zahteve skupaj z njihovimi tehničnimi implikacijami za bodočo rešitev.

2.1 Uporabnost

Uporabnost sistema predstavlja enega ključnih nefunkcionalnih vidikov bodoče rešitve, saj neposredno vpliva na učinkovitost dela uporabnikov, kakovost vnesenih podatkov ter sprejetost

sistema med različnimi ciljnimi skupinami. Tehnične specifikacije izhajajo iz ugotovitev, rešitev in zahtev, opisanih v dokumentih »Uporabniški vmesnik« in »Funkcionalne specifikacije«.

Bodoča rešitev mora podpirati več uporabniških vlog z bistveno različnimi nalogami, stopnjami digitalne pismenosti in pogostostjo uporabe sistema (npr. zeleni koordinatorji, ponudniki, ocenjevalci, vodje ocenjevalcev in administratorji). Posledično mora biti uporabniška izkušnja prilagojena posamezni vlogi in njenemu kontekstu uporabe, kar je upoštevano tako na ravni informacijskega modela kot na ravni tehnične zasnove uporabniškega vmesnika.

Tehnične zahteve glede uporabnosti sistema vključujejo zlasti:

- jasno in konsistentno navigacijsko strukturo, ki temelji na uporabniških tokovih in vsebinskih sklopih,
- logično razporeditev funkcionalnosti glede na uporabniške vloge in stanja poslovnih procesov,
- zmanjševanje kognitivne obremenitve uporabnikov z uporabo postopnega razkrivanja vsebin (*progressive disclosure*),
- podporo ponovni uporabi podatkov, avtomatskim predizpolnitvam in kontekstnim namigom, kjer je to smiselno in varno.

Z vidika tehnične implementacije mora sistem omogočati, da se uporabniški vmesnik in funkcionalnosti prilagajajo glede na:

- uporabniško vlogo in dodeljene pravice,
- trenutno stanje presoje, kriterija ali kazalnika,
- kontekst uporabe (npr. vnos podatkov, pregled, ocenjevanje, analiza).

Pomemben vidik uporabnosti je tudi odzivnost sistema in jasna povratna informacija uporabniku. Tehnična zasnova mora zagotavljati:

- predvidljivo in konsistentno obnašanje sistema,
- jasno označevanje stanj (npr. shranjeno, v obdelavi, zaklenjeno, oddano),
- ustrezna obvestila o napakah, validacijah in uspešno izvedenih akcijah.

Uporabnost sistema je neposredno povezana tudi z obvladljivostjo kompleksnih vsebin in procesov, značilnih za standard *Slovenia Green*. Tehnična zasnova mora omogočati modularno gradnjo uporabniškega vmesnika in ponovno uporabo skupnih komponent, kar prispeva k večji konsistentnosti, lažjemu učenju sistema in enostavnejšemu vzdrževanju.

2.2 Razpoložljivost

Razpoložljivost sistema je ena izmed ključnih nefunkcionalnih zahtev bodoče rešitve. Rešitev predstavlja transakcijsko aplikacijo z omejenim številom uporabnikov, katere kratkotrajno nedelovanje praviloma ne povzroči neposredne zaustavitve širših poslovnih operacij naročnika. Kljub temu so izpadi z vidika uporabniške izkušnje in zaupanja v sistem nesprejemljivi, zato mora tehnična zasnova zagotavljati stabilno, predvidljivo in uporabnikom prijazno delovanje sistema.

Sistem mora zagotavljati razpoložljivost 24/7 z minimalnimi načrtovanimi in nenačrtovanimi izpadi, pri čemer mora biti stopnja razpoložljivosti sorazmerna z dejansko kritičnostjo sistema in obsegom njegove uporabe.

2.2.1 Zagotavljanje razpoložljivosti

Bodoča rešitev Slovenia Green mora biti zasnovana in gostovana v sodobnem oblačnem okolju, ki omogoča visoko stopnjo razpoložljivosti, stabilno delovanje ter odpornost na izpade posameznih

infrastrukturnih komponent. Razpoložljivost predstavlja ključno nefunkcionalno zahtevo, saj sistem podpira poslovno kritične procese vnosa podatkov, presoje, ocenjevanja ter analitike, ki morajo biti uporabnikom dostopni neprekinjeno in predvidljivo.

Z vidika infrastrukturne zasnove se predvideva uporaba oblačne storitve, ki je skladna z arhitekturnimi načeli sistema NiST ter omogoča izvedbo vseh zahtev, opredeljenih v tem dokumentu. Gostovanje mora zagotavljati ustrezno raven operativne neodvisnosti rešitve Slovenia Green, hkrati pa omogočati varno in nadzorovano integracijo v širši ekosistem NiST.

Zagotavljanje razpoložljivosti mora temeljiti na naslednjih tehničnih izhodiščih:

- **uporabi mehanizmov visoke razpoložljivosti** na aplikacijskem in podatkovnem sloju (npr. več instanc aplikacije, replikacija podatkovnih baz, avtomatski failover),
- **redundantni zasnovi ključnih komponent**, tako da izpad posamezne instance ali storitve ne povzroči nedelovanja celotnega sistema,
- **centraliziranem nadzoru delovanja**, ki vključuje spremljanje stanja storitev, zaznavanje napak ter avtomatizirano obveščanje v primeru incidentov,
- **ločeni organizaciji okolij**, pri čemer morajo biti produkcijsko, testno in razvojno okolje jasno ločeni tako z vidika podatkov kot tudi infrastrukturnih virov,
- **jasno opredeljenih postopkih vzdrževanja**, ki omogočajo izvajanje nadgradenj in sprememb z minimalnimi vplivi na razpoložljivost sistema (npr. rolling deployments, načrtovani vzdrževalni intervali).

Oblačna infrastruktura mora biti izbrana in konfigurirana tako, da omogoča:

- dolgoročno stabilno delovanje rešitve 24/7,
- skalabilnost glede na rast števila uporabnikov, destinacij in količine podatkov,
- varno izvajanje integracij z NiST brez neposredne odvisnosti delovanja sistema od razpoložljivosti drugih komponent ekosistema,
- jasno razmejitev odgovornosti med upravljavci infrastrukture in vzdrževalci aplikacije.

Ključno načelo pri zasnovi razpoložljivosti je, da mora biti rešitev Slovenia Green obravnavana kot samostojen informacijski sistem, katerega delovanje mora ostati stabilno tudi v primeru motenj ali vzdrževalnih aktivnosti v drugih delih okolja NiST. Integracije morajo biti zato izvedene na način, ki omogoča degradirano delovanje in nadzorovano obravnavo napak, brez vpliva na osnovne funkcionalnosti sistema.

Takšen pristop zagotavlja, da bo rešitev Slovenia Green gostovana v infrastrukturnem okolju, ki je skladno s sodobnimi praksami oblačnega gostovanja, omogoča visoko razpoložljivost ter podpira dolgoročno vzdržnost, nadgradljivost in operativno zanesljivost sistema.

Spodaj je naveden KPI, ki ga mora izvajalec doseči pred prevzemom.

KPI: Dokumentirana zasnova razpoložljivosti in obnove sistema
Priprava tehnična dokumentacija o redundanci, varnostnih kopijah in postopkih obnove sistema.
Zahtevana vrednost: Dokumentacija potrjena s strani naročnika pred produkcijsko uvedbo Faze 1.

2.2.2 Redundantna infrastruktura

Ne glede na izbrani izvedbeni scenarij mora biti infrastruktura zasnovana redundantno na ključnih slojih sistema, pri čemer se redundanca uvaja racionalno in sorazmerno z dejanskimi potrebami sistema.

Minimalne zahteve vključujejo:

- možnost delovanja več instanc aplikacijskega sloja,
- visoko razpoložljivo konfiguracijo podatkovnega sloja,
- ločitev produkcijskega okolja od testnih in razvojnih okolij.

Spodaj so navedeni KPI-ji, ki jih mora izvajalec doseči pred prevzemom.

KPI: Razpoložljivost sistema v času integracijskega in sistemskega testiranja
Delež časa, v katerem je sistem v testnem okolju dosegljiv in funkcionalen v času planiranih testnih aktivnosti.
Zahtevana vrednost: ≥ 99 %.

KPI: Razpoložljivost sistema v produkcijskem okolju
Delež časa, v katerem je sistem v produkcijskem okolju dosegljiv in polno funkcionalen.
Zahtevana vrednost: ≥ 99,9 %.

2.2.3 Varnostne kopije in obnova sistema

Za oba scenarija mora biti zagotovljen enakovreden pristop k izdelavi varnostnih kopij in obnovi sistema, ki vključuje:

- avtomatizirano izdelavo varnostnih kopij podatkovnih baz in ključnih vsebin,
- shranjevanje varnostnih kopij na ločenih in varnih lokacijah,
- jasno definirane postopke obnove sistema in podatkov.

Spodaj ali so navedeni KPI-ji, ki jih mora izvajalec doseči pred prevzemom.

KPI: Uspešnost dnevnih varnostnih kopij po uvedbi Faze 1
Delež uspešno izvedenih dnevnih backup postopkov podatkovne baze v produkciji.
Zahtevana vrednost: 100 % v toku Faze 2 in dve naključni testiranji z uspešno obnovo iz dnevne varnostne kopije.

KPI: Čas ponovne vzpostavitve sistema po simuliranem izpadu
Čas, potreben za ponovno vzpostavitev delovanja sistema po načrtovanem izpadu v testnem okolju s testno podatkovno bazo – obnovitev aplikativne rešitve in testne podatkovne baze.
Zahtevana vrednost: ≤ 4 ure.

2.2.4 Neodvisnost razpoložljivosti

Ne glede na izbrani scenarij gostovanja mora biti delovanje rešitve zasnovano tako, da njena razpoložljivost ni neposredno odvisna od razpoložljivosti ali razvojne dinamike drugih sistemov. Integracije z NiST in drugimi zunanjimi sistemi morajo omogočati nadzorovano in degradirano delovanje v primeru začasne nedosegljivosti posameznih virov.

Spodaj je naveden KPI, ki ga mora izvajalec doseči pred prevzemom.

KPI: Neodvisnost rešitve ob hkratni povezljivosti z NiST
Delež preverjenih scenarijev, kjer sistem deluje samostojno in omogoča standardizirano povezovanje z NiST.
Zahtevana vrednost: 100 % izvedeni testi integracije, kjer je sistem NiST dostopen ali nedostopen.

2.3 Zanesljivost

Zanesljivost sistema predstavlja sposobnost rešitve, da tudi ob delnih napakah, povečanih obremenitvah ali začasnih izpadih posameznih komponent deluje pravilno, predvidljivo in brez izgube podatkov. Pri bodoči rešitvi je zanesljivost posebej pomembna zaradi transakcijske narave sistema,

kjer se obdelujejo strukturirani podatki, poslovna stanja in odločitve, ki se uporabljajo v nadaljnjih fazah presoje, ocenjevanja in poročanja.

Tehnična zasnova mora zagotoviti visoko stopnjo zanesljivosti sistema na vseh ključnih ravneh, zlasti:

- na ravni poslovnih procesov in stanj,
- na ravni obdelave podatkov in shranjevanja,
- na ravni integracij z zunanjimi sistemi.

2.3.1 Zahteve za visoko stopnjo zanesljivosti sistema

Sistem mora biti zasnovan tako, da preprečuje izgubo podatkov, nekonsistentna stanja in nepredvidljivo obnašanje tudi v primeru delnih napak. Ključne zahteve za zagotavljanje zanesljivosti vključujejo:

- dosledno upravljanje transakcij in poslovnih stanj, tako da so spremembe podatkov izvedene atomarno in sledljivo,
- jasno definirana in nadzorovana stanja poslovnih procesov (npr. presoja, kriterij, kazalnik), skladno z opisi v dokumentu »Funkcionalne specifikacije«,
- mehanizme za preprečevanje podvajanja, nekonsistentnih zapisov in nenamernih prepisov podatkov,
- revizijsko sled za ključne spremembe podatkov in stanj, ki omogoča sledljivost in analizo v primeru napak.

Zanesljivost sistema mora biti zagotovljena ne glede na izbrani scenarij gostovanja (znotraj infrastrukture NiST ali kot samostojna rešitev pri neodvisnem ponudniku oblačnih storitev).

Spodaj je naveden KPI, ki ga mora izvajalec doseči pred prevzemom.

KPI: Število kritičnih napak v fazi sistemskega testiranja
Kritične napake, identificirane v fazi sistemskega testiranja, v kateri se preverja delovanje celotnega sistema kot integrirane celote (vključno z vsemi moduli, podatkovnimi tokovi in integracijami) v testnem okolju pred produkcijsko uvedbo. Število napak, ki povzročijo izgubo podatkov ali onemogočijo nadaljevanje testiranja.
Zahtevana vrednost: 0 pred produkcijsko uvedbo Faze 1.

2.3.2 Samodejno obnavljanje sistema po napakah

Pomemben vidik zanesljivosti predstavlja sposobnost sistema, da se po napakah samodejno obnovi brez potrebe po takojšnjem ročnem posredovanju. Tehnična zasnova mora predvideti naslednje postopke samodejnega obnavljanja:

- samodejni ponovni zagon aplikacijskih komponent v primeru zaznanih napak ali nedelovanja posamezne instance,
- samodejno prerazporejanje obremenitev na delujoče komponente sistema,
- zaznavanje in obravnavo neuspelih ali delno izvedenih transakcij na način, ki ne povzroča nekonsistentnih podatkovnih stanj.

V primeru napak v integracijah z zunanjimi sistemi mora sistem omogočati:

- ponovni poskus obdelave (*retry*) ob ustreznih časovnih intervalih,
- začasno odloženo obdelavo (*retry / delayed processing*), kjer je to smiselno,
- jasno označevanje stanj in obveščanje uporabnikov v primeru, da posamezen podatek ni bil uspešno pridobljen ali posredovan.

Spodaj je naveden KPI, ki ga mora izvajalec doseči pred prevzemom.

KPI: Neodvisnost delovanja ob nedosegljivosti integracij
Delež testnih scenarijev, kjer sistem ob simulirani nedosegljivosti zunanjega integriranega sistema ohrani osnovne funkcionalnosti in pokriva proces iz faze I.
Zahtevana vrednost: 100 % uspešno obravnavanih scenarijev.

2.3.3 Nadzor delovanja in zaznavanje napak

Za zagotavljanje zanesljivega delovanja mora sistem vključevati osnovne mehanizme za nadzor in zaznavanje napak, ki omogočajo:

- spremljanje delovanja ključnih komponent sistema,
- beleženje napak in izjem na aplikacijski in integracijski ravni,
- hitro identifikacijo ponavljajočih se ali kritičnih napak.

Nadzorni mehanizmi morajo podpirati tako samodejno obnavljanje sistema kot tudi učinkovito odpravljanje napak v primeru kompleksnejših incidentov.

Spodaj so navedeni KPI-ji, ki jih mora izvajalec doseči pred prevzemom.

KPI: Pokritost monitoringa za produkcijsko uvedbo Faze 1
Delež ključnih komponent (API, baza, integracijski sloj), vključenih v monitoring (npr. Kibana).
Zahtevana vrednost: 100 %.

KPI: Čas zaznave incidenta v produkciji (MTTD)
Povprečni čas do zaznave kritične napake (opozorilni email in/ali SMS).
Zahtevana vrednost: ≤ 5 minut.

2.3.4 Zanesljivost v kontekstu uporabniške izkušnje

Zanesljivost sistema se mora odražati tudi v uporabniški izkušnji. Sistem mora uporabnikom zagotavljati jasno in predvidljivo obnašanje, kar vključuje:

- jasno povratno informacijo o uspešnosti ali neuspešnosti posameznih akcij,
- preprečevanje izgube uporabniških vnosov v primeru napak,
- možnost nadaljevanja dela po odpravi napake brez ponovnega vnosa podatkov, kjer je to izvedljivo.

Spodaj so navedeni KPI-ji, ki jih mora izvajalec doseči pred prevzemom.

KPI: Skladnost implementacije z odobrenimi uporabniškimi tokovi
Delež ključnih procesov (vnos podatkov, oddaja presoje, pregled rezultatov), izvedenih skladno z dogovorjenimi (opisani v dokumentu »Funkcionalne specifikacije« in dodatno določeni) uporabniškimi tokovi.
Zahtevana vrednost: 100 %.

KPI: Uspešnost uporabniških testov brez pomoči
Delež testnih uporabnikov, ki uspešno izvedejo predvidene naloge brez dodatnih navodil (Unmoderated Usability Testing).
Zahtevana vrednost: ≥ 90 %.

KPI: Število kritičnih UX ugotovitev v fazi testiranja
Število ugotovitev uporabnikov, ki bistveno otežujejo ali onemogočajo uporabo sistema.
Zahtevana vrednost: 0 pred produkcijsko uvedbo Faze 1.

2.4 Razširljivost

Razširljivost sistema predstavlja sposobnost rešitve, da se učinkovito prilagaja spremembam v obsegu uporabe, količini podatkov in kompleksnosti poslovnih procesov skozi čas. Pri bodoči rešitvi je potrebno razširljivost obravnavati z vidika pričakovane rasti sistema, ne pa z vidika masovne uporabe ali ekstremnih obremenitev.

Bodoča rešitev je zasnovana kot transakcijska aplikacija z omejenim številom sočasnih uporabnikov, vendar z jasno pričakovano rastjo:

- števila destinacij in ponudnikov,
- količine zbranih podatkov in dokazil,
- obsega analitičnih in poročilnih vsebin,
- števila integracij z zunanjimi sistemi.

Tehnična zasnova mora zato omogočati postopno in nadzorovano povečevanje zmogljivosti sistema brez potrebe po večjih arhitekturnih posegih.

2.4.1 Zahteve za skalabilnost sistema

Sistem mora biti zasnovan tako, da omogoča:

- povečanje števila uporabnikov brez degradacije uporabniške izkušnje,
- obvladovanje rasti podatkov skozi čas, zlasti pri hranjenju dokumentov, zgodovinskih presoj in revizijskih sledi,
- razširitev funkcionalnosti in dodajanje novih vsebinskih sklopov brez negativnega vpliva na obstoječe delovanje sistema.

Razširljivost mora biti zagotovljena na vseh ključnih slojih sistema, pri čemer se posebna pozornost nameni ločevanju odgovornosti med posameznimi sloji (uporabniški vmesnik, aplikacijska logika, podatkovni sloj).

Spodaj je naveden KPI, ki ga mora izvajalec doseči pred prevzemom.

KPI: Sočasna uporaba sistema v produkciji
Število sočasnih uporabnikov, pri katerem sistem ohrani zahtevani odzivni čas.
Zahtevana vrednost: ≥ 25 sočasnih uporabnikov.

2.4.2 Horizontalno skaliranje

Horizontalno skaliranje predstavlja osnovni in prednostni mehanizem za prilagajanje sistema povečanemu številu uporabnikov in zahtevkov. Tehnična zasnova mora omogočati:

- dodajanje novih instanc aplikacijskega sloja brez sprememb v aplikacijski kodi,
- porazdelitev obremenitev med več instanc sistema,
- neprekinjeno delovanje sistema ob dinamičnem prilagajanju števila instanc.

Horizontalno skaliranje je posebej primerno za aplikacijski sloj in uporabniški vmesnik ter omogoča učinkovito obvladovanje sezonskih ali občasnih povečanj obremenitev (npr. obdobja oddaje presoj).

2.4.3 Vertikalno skaliranje

Vertikalno skaliranje predstavlja dopolnilni mehanizem razširljivosti in se uporablja tam, kjer horizontalno skaliranje ni smiselno ali izvedljivo. Tehnična zasnova mora omogočati:

- povečanje zmogljivosti posameznih komponent (npr. procesorska moč, pomnilnik, zmogljivost podatkovne baze),
- prilagajanje zmogljivosti glede na dejanske potrebe sistema,

- izvedbo nadgradenj z minimalnim vplivom na delovanje sistema.

Vertikalno skaliranje je posebej relevantno za podatkovni sloj in podporne storitve, kjer je konsistentnost podatkov ključnega pomena.

2.4.4 Razširljivost v kontekstu integracij

Razširljivost sistema mora upoštevati tudi postopno povečevanje števila integracij z zunanjimi sistemi. Tehnična zasnova mora omogočati:

- dodajanje novih integracij brez vpliva na obstoječe funkcionalnosti,
- obvladovanje povečanega obsega podatkovnih prenosov,
- izolacijo napak posameznih integracij, tako da te ne vplivajo na stabilnost celotnega sistema.

Spodaj je naveden KPI, ki ga mora izvajalec doseči pred prevzemom.

KPI: Pripravljenost sistema na federirano avtentikacijo
V testnem okolju mora biti izvedena uspešna prijava uporabnika preko referenčnega OpenID Connect ponudnika (npr. Keycloak), pri čemer se integracija izvede izključno s konfiguracijo brez sprememb poslovne logike aplikacije.
Zahtevana vrednost: Uspešno preverjena prijava uporabnika prek federirane avtentikacije (OpenID Connect).

2.5 Zmogljivost

Zmogljivost sistema se nanaša na sposobnost rešitve, da uporabnikom zagotavlja hitro odzivnost, nizko zakasnitev in stabilno delovanje tudi v obdobjih povečanih obremenitev. Pri bodoči rešitvi je zmogljivost ključna predvsem z vidika uporabniške izkušnje in učinkovitosti dela, ne pa obvladovanja ekstremno velikih količin sočasnih zahtevkov.

Bodoča rešitev je transakcijska aplikacija z omejenim številom uporabnikov, pri kateri se obremenitve pojavljajo predvsem občasno, npr. v obdobjih intenzivnega vnosa podatkov, oddaje presoj ali pregleda in ocenjevanja. Kljub temu mora sistem zagotavljati predvidljivo in odzivno delovanje tudi v teh primerih.

2.5.1 Odzivnost in sočasna obdelava zahtevkov

Tehnična zasnova mora zagotavljati, da so ključne uporabniške operacije izvedene v razumnem in predvidljivem času. To vključuje zlasti:

- hitro nalaganje ključnih zaslonov in pregledov,
- odzivno izvajanje osnovnih transakcij (shranjevanje, oddaja, potrjevanje),
- tekoče delovanje uporabniškega vmesnika brez opaznih zamikov.

Odzivnost sistema mora podpirati učinkovito delo uporabnikov in zmanjšujejo kognitivno obremenitev, zlasti pri delu z obsežnejšimi vsebinami (kriteriji, kazalniki, dokazila).

Sistem mora omogočati hkratno obdelavo večjega števila zahtevkov brez opaznega zmanjšanja zmogljivosti. To vključuje:

- podporo sočasnemu delu več uporabnikov v različnih vlogah,
- obvladovanje sočasnih transakcij brez vpliva na konsistentnost podatkov,
- enakomerno porazdelitev obremenitev na aplikacijskem sloju.

Tehnična zasnova mora preprečevati situacije, kjer bi obremenitev enega dela sistema negativno vplivala na celotno delovanje rešitve.

Spodaj je naveden KPI, ki ga mora izvajalec doseči pred prevzemom.

KPI: Odzivni čas ključnih operacij v produkciji Faze 1
Čas izvajanja nalaganja pregleda, shranjevanja podatkov in oddaje presoje.
Zahtevana vrednost: ≤ 2 sekundi za 95 % zahtevkov (ne šteje se izdelava AI predloga ali izpisov in poročil).

2.5.2 Arhitekturni pristopi za zagotavljanje zmogljivosti

Za zagotavljanje ustrezne zmogljivosti mora sistem uporabljati preverjene arhitekturne pristope, kot so:

- ločevanje uporabniškega vmesnika, aplikacijske logike in podatkovnega sloja,
- optimizirano upravljanje podatkovnih poizvedb in prenosov,
- uporaba predpomnjenja (*caching*) tam, kjer je to smiselno in ne ogroža konsistentnosti podatkov.

Ti pristopi omogočajo stabilno delovanje sistema tudi ob povečanju števila uporabnikov ali obsega podatkov.

Zmogljivost sistema mora upoštevati tudi vpliv integracij z zunanjimi sistemi. Tehnična zasnova mora omogočati:

- asinhrono obdelavo zahtevkov, kjer neposreden odziv ni potreben,
- nadzorovano obravnavo počasnih ali začasno nedosegljivih zunanjih virov,
- preprečevanje blokiranja ključnih funkcionalnosti zaradi integracij.

Spodaj je naveden KPI, ki ga mora izvajalec doseči pred prevzemom.

KPI: Čas obravnave napake pri nedosegljivosti zunanjega sistema
Čas, v katerem sistem zazna napako in prikaže obvestilo uporabniku.
Zahtevana vrednost: ≤ 5 sekund.

2.5.3 Nadzor in prilagajanje zmogljivosti

Zmogljivost sistema mora biti spremljana in prilagajana glede na dejansko uporabo. Tehnična zasnova mora omogočati:

- spremljanje ključnih kazalnikov zmogljivosti,
- pravočasno zaznavanje ozkih grl,
- postopno prilagajanje zmogljivosti brez večjih posegov v arhitekturo sistema.

2.6 Uporabniška izkušnja

Uporabniška izkušnja (UX) je celovito obravnavana v dokumentu »Uporabniški vmesnik«, kjer so bili definirani uporabniški tokovi, struktura uporabniškega vmesnika, žični modeli ter izvedbeni oblikovni pristop. Namen teh tehničnih specifikacij ni ponavljanje vsebinskih in oblikovnih rešitev, temveč opredelitev tehničnih zahtev, ki omogočajo njihovo kakovostno, stabilno in dolgoročno izvedbo.

Tehnična zasnova sistema mora zagotavljati, da se v praksi uresničijo UX cilji, opredeljeni v dokumentu »Uporabniški vmesnik«, zlasti:

- jasnost in predvidljivost obnašanja sistema,
- konsistentnost uporabniškega vmesnika skozi celoten sistem,
- prilagoditev uporabniške izkušnje glede na uporabniško vlogo in kontekst uporabe,
- odzivno in tekoče delovanje uporabniškega vmesnika.

Z vidika tehnične implementacije mora sistem omogočati:

- modularno gradnjo uporabniškega vmesnika in ponovno uporabo skupnih komponent,
- podporo za različne naprave in velikosti zaslonov,
- jasno ločevanje predstavitvenega sloja od poslovne logike, kar omogoča lažje vzdrževanje in nadaljnje nadgradnje.

Pomemben vidik uporabniške izkušnje je tudi obvladovanje napak in izjemnih stanj. Sistem mora uporabnikom zagotavljati:

- razumljiva in kontekstualna sporočila ob napakah,
- zaščito pred izgubo vnesenih podatkov,
- možnost nemotenega nadaljevanja dela po odpravi napake.

Uporabniška izkušnja je neposredno povezana z zanesljivostjo, zmogljivostjo in razpoložljivostjo sistema. Tehnična analiza zato obravnava UX kot rezultat usklajenega delovanja vseh tehničnih slojev sistema, ne kot samostojen ali izoliran vidik.

Spodaj je naveden KPI, ki ga mora izvajalec doseči pred prevzemom.

KPI: Uvedba lokalizacije uporabniškega vmesnika v času razvoja Faze II brez sprememb poslovne logike
Po produkcijski uvedbi Faze I mora biti v okviru razvoja Faze II izvedena uvedba dodatnega jezika oziroma lokalizacije uporabniškega vmesnika na konfiguracijski način, brez sprememb poslovne logike sistema in brez vpliva na produkcijsko delovanje Faze I.
Zahtevana vrednost: 1 uspešno uveden dodatni jezik v testnem ali produkcijskem okolju (prevod je lahko avtomatski).

Dostopnost spletnega uporabniškega vmesnika
Spletni uporabniški vmesnik bodoče rešitve STO Zeleno shemo mora biti zasnovan in implementiran v skladu z veljavnimi zahtevami glede dostopnosti spletnih vsebin, z namenom zagotavljanja enakovrednega dostopa do informacij in funkcionalnosti za vse uporabnike, vključno z osebami z različnimi oblikami oviranosti.

Rešitev mora biti skladna z Zakonom o dostopnosti spletišč in mobilnih aplikacij ter s smernicami **Web Content Accessibility Guidelines (WCAG) 2.1**, in sicer najmanj na ravni **AA**.

Pri zasnovi in izvedbi uporabniškega vmesnika je treba upoštevati zlasti naslednja načela dostopnosti:

- zaznavnost vsebin (ustrezni barvni kontrasti, berljiva tipografija, alternativna besedila za nebesedilne vsebine),
- upravljivost (polna uporaba vmesnika s tipkovnico, logična navigacija, ustrezni fokusni indikatorji),
- razumljivost (jasna struktura vsebin, predvidljivo obnašanje uporabniškega vmesnika),
- robustnost (združljivost z asistivnimi tehnologijami in različnimi uporabniškimi agenti).

Dostopnost mora biti upoštevana že v fazi načrtovanja uporabniške izkušnje ter preverjena v okviru testiranja rešitve. Morebitna odstopanja od zahtev dostopnosti morajo biti utemeljena in dokumentirana.

2.7 Interoperabilnost

Interoperabilnost sistema predstavlja sposobnost rešitve, da se na standardiziran, zanesljiv in nadzorovan način povezuje z drugimi informacijskimi sistemi ter izmenjuje podatke brez ustvarjanja tesnih tehničnih ali izvedbenih odvisnosti. Pri bodoči rešitvi je interoperabilnost ključna predvsem v

kontekstu povezovanja z obstoječim sistemom NiST ter z drugimi zunanjimi viri podatkov, ki so relevantni za delovanje standarda Slovenia Green.

Tehnična zasnova mora omogočati integracije, ki podpirajo dolgoročno vzdržnost sistema, neodvisno nadgrajevanje posameznih komponent ter postopno dodajanje novih povezav.

2.7.1 Zahteve za integracijo z drugimi informacijskimi sistemi

Sistem mora omogočati integracijo z drugimi informacijskimi sistemi zaradi:

- pridobivanja podatkov iz zunanjih virov,
- posredovanja podatkov drugim sistemom,
- podpore analitiki, poročanju in nadaljnji obdelavi podatkov.

Integracije morajo biti zasnovane tako, da:

- so jasno definirane in dokumentirane,
- omogočajo nadzorovano uporabo podatkov,
- ne povzročajo neposredne odvisnosti delovanja sistema od razpoložljivosti zunanjih virov.

Posebna pozornost mora biti namenjena integracijam z NiST, kjer je potrebno zagotoviti semantično skladnost podatkov ter jasno razmejitev odgovornosti med sistemi.

Spodaj je naveden KPI, ki ga mora izvajalec doseči pred prevzemom.

KPI: Dokumentirana zasnova integracij z drugimi informacijskimi sistemi
Pripravljena tehnična dokumentacija integracij, ki vključuje opis vmesnikov, podatkovnih tokov ter opredelitev odgovornosti med sistemi, zlasti za integracijo z NiST, ob zagotavljanju semantične skladnosti podatkov.
Zahtevana vrednost: Dokumentacija potrjena s strani naročnika pred produkcijsko uvedbo Faze 1.

2.7.2 Standardizirani protokoli in formati za izmenjavo podatkov

Za zagotavljanje interoperabilnosti mora sistem uporabljati standardizirane in splošno uveljavljene protokole ter podatkovne formate. Tehnična zasnova mora predvideti:

- uporabo standardnih spletnih protokolov za izmenjavo podatkov,
- uporabo strukturiranih in razširljivih podatkovnih formatov,
- jasno opredeljene podatkovne modele in sheme za izmenjavo podatkov.

Uporaba standardov omogoča:

- lažjo integracijo z obstoječimi in bodočimi sistemi,
- zmanjševanje tveganj pri nadgradnjah in zamenjavah posameznih komponent,
- večjo preglednost in sledljivost podatkovnih tokov.

2.7.3 Nadzor in obvladovanje integracij

Tehnična zasnova mora omogočati nadzor nad delovanjem integracij, kar vključuje:

- zaznavanje napak in nedosegljivosti zunanjih sistemov,
- obvladovanje počasnih ali neuspelih klicev,
- nadzorovano ponavljanje zahtevkov, kjer je to smiselno.

Integracije morajo biti zasnovane tako, da v primeru napak omogočajo degradirano delovanje sistema in ne vplivajo na razpoložljivost ali zanesljivost osnovnih funkcionalnosti.

2.7.4 Pripravljenost na nadaljnje razširitve

Interoperabilnost mora biti zasnovana tako, da omogoča postopno dodajanje novih integracij brez večjih arhitekturnih posegov. Sistem mora biti pripravljen na:

- vključevanje novih podatkovnih virov,
- spremembe obstoječih integracij,
- razširitev obsega izmenjave podatkov glede na prihodnje potrebe naročnika.

2.8 Vzdržljivost

Vzdržljivost sistema predstavlja sposobnost rešitve, da dolgoročno stabilno deluje kljub spremembam v obsegu podatkov, nadgradnjam funkcionalnosti, tehnološkim posodobitvam ter občasnim motnjam v delovanju posameznih komponent. Pri bodoči rešitvi je vzdržljivost posebej pomembna zaradi večletnega življenjskega cikla sistema, ponavljajočih se letnih presoj ter potrebe po ohranjanju zgodovinskih podatkov in revizijskih sledi.

Sistem mora biti zasnovan tako, da omogoča neprekinjeno uporabo skozi daljše časovno obdobje brez potrebe po večjih posegih v arhitekturo ali podatkovni model.

2.8.1 Vzdržljivost aplikacijskega in podatkovnega sloja

Tehnična zasnova mora omogočati, da aplikacijski in podatkovni sloj:

- preneseta dolgoročno rast količine podatkov (npr. dokumentacija, dokazila, arhiv presoj),
- omogočata redne nadgradnje brez izgube podatkov ali daljših izpadov,
- ohranjata konsistentnost poslovnih stanj tudi ob ponovnih zagonih ali napakah.

Podatkovni sloj mora biti zasnovan tako, da omogoča učinkovito arhiviranje zgodovinskih podatkov in njihovo nadaljnjo uporabo za analitiko, poročanje in primerjave skozi čas.

2.8.2 Vzdržljivost poslovnih procesov

Vzdržljivost sistema se ne nanaša zgolj na infrastrukturo, temveč tudi na stabilnost poslovnih procesov, ki jih sistem podpira. Rešitev mora omogočati:

- ponovljivost procesov skozi več let (npr. letne presoje),
- ohranjanje podatkov in stanj preteklih presoj brez vpliva na tekoče procese,
- prilagajanje procesov in pravil brez potrebe po posegih v temeljno arhitekturo sistema.

To je še posebej pomembno v kontekstu sprememb standardov, metodologij in pravil, ki se lahko skozi čas razvijajo.

2.8.3 Vzdržljivost ob nadgradnjah in spremembah

Tehnična zasnova mora omogočati izvajanje nadgradenj in sprememb sistema na način, ki:

- minimalno vpliva na delovanje sistema,
- ne zahteva dolgotrajnih izpadov,
- omogoča povratek na prejšnje stanje v primeru nepredvidenih težav.

Vzdržljivost vključuje tudi sposobnost sistema, da se prilagodi spremembam v infrastrukturnem okolju, vključno z morebitno zamenjavo ali nadgradnjo oblačnega ponudnika.

2.8.4 Vzdržljivost integracij

Sistem mora biti vzdržljiv tudi v kontekstu integracij z zunanjimi sistemi. Integracije morajo biti zasnovane tako, da:

- spremembe zunanjih sistemov ne povzročijo nedelovanja celotne rešitve,
- je mogoče posamezne integracije prilagoditi ali nadomestiti brez vpliva na osnovne funkcionalnosti,
- so integracije jasno ločene od osrednje poslovne logike sistema.

2.9 Prilagodljivost

Prilagodljivost sistema predstavlja sposobnost rešitve, da se učinkovito prilagaja spremembam v poslovnih zahtevah, organizacijskem okolju, metodologiji presoje ter tehnoloških izhodiščih, brez potrebe po temeljnih arhitekturnih spremembah ali obsežnih posegih v obstoječo rešitev.

Pri bodoči rešitvi Slovenia Green je prilagodljivost posebej pomembna zaradi:

- razvoja in nadgradenj standardov, kriterijev in kazalnikov skozi čas,
- vključevanja novih vrst uporabnikov, vlog ali postopkov,
- postopnega širjenja obsega sistema (npr. dodatni tipi presoj, analitike, poročanja),
- morebitnih sprememb v organizacijskem ali tehničnem okolju naročnika.

2.9.1 Prilagodljivost funkcionalnosti in poslovnih pravil

Tehnična zasnova mora omogočati, da se spremembe v poslovni logiki in pravilih:

- izvajajo konfiguracijsko, kjer je to smiselno,
- ne zahtevajo sprememb temeljnega podatkovnega modela,
- ne vplivajo na obstoječe zgodovinske podatke in zaključene presoje.

Sistem mora omogočati sočasno sobivanje različnih različic procesov ali pravil (npr. različne metodologije presoj po letih), kar je ključno za ohranjanje primerljivosti in sledljivosti podatkov skozi čas.

2.9.2 Prilagodljivost uporabniških vlog in dostopov

Rešitev mora omogočati prilagodljivo upravljanje uporabniških vlog in pravic, kar vključuje:

- dodajanje novih vlog ali pod-vlog brez posegov v osnovno arhitekturo,
- prilagajanje pravic glede na vlogo, stanje procesa in tip presoje,
- možnost razširitve sistema z novimi uporabniškimi skupinami v prihodnosti.

Prilagodljivost na tem področju je posebej pomembna v kontekstu sodelovanja z zunanjimi deležniki in morebitnih organizacijskih sprememb.

2.9.3 Prilagodljivost vsebin in podatkovnih struktur

Sistem mora biti zasnovan tako, da omogoča:

- dodajanje novih vsebinskih sklopov, kriterijev in kazalnikov,
- prilagoditve strukture podatkov brez potrebe po migracijah celotnega sistema,
- razširitev metapodatkov in povezav z drugimi sistemi.

Podatkovni model mora biti dovolj fleksibilen, da podpira razvoj sistema, hkrati pa dovolj stabilen, da zagotavlja konsistentnost in zanesljivost podatkov.

2.9.4 Prilagodljivost infrastrukturnega okolja

Prilagodljivost vključuje tudi sposobnost sistema, da se prilagodi spremembam v infrastrukturnem okolju. Tehnična zasnova mora omogočati:

- razmestitev sistema v različnih oblačnih ali hibridnih okoljih,
- prilagoditve infrastrukturnih nastavitev brez sprememb aplikacijske logike,
- morebitno selitev sistema med ponudniki oblačnih storitev.

Ta vidik prilagodljivosti je neposredno povezan z obvladljivostjo, prenosljivostjo in dolgoročno vzdržnostjo rešitve.

2.10 Varnost

Varnost sistema predstavlja nefunkcionalni vidik bodoče rešitve, saj sistem obdeluje strukturirane podatke, dokumentacijo, dokazila ter rezultate presoj, ki so dostopni različnim uporabniškim vlogam z jasno določenimi pravicami. Tehnična zasnova mora zagotavljati celovito varovanje sistema in podatkov skozi celoten življenjski cikel rešitve.

Varnostni pristop mora biti sorazmeren z naravo sistema: gre za transakcijsko aplikacijo z omejenim številom znanih uporabnikov, vendar z visokimi zahtevami glede zaupanja, integritete podatkov in sledljivosti sprememb.

2.10.1 Avtentikacija in avtorizacija

Sistem mora zagotavljati zanesljivo in varno avtentikacijo uporabnikov ter dosledno avtorizacijo dostopov glede na uporabniško vlogo, stanje poslovnih procesov in kontekst uporabe.

Tehnične zahteve vključujejo:

- jasno ločevanje avtentikacije (identiteta uporabnika) in avtorizacije (pravice dostopa),
- centralno upravljanje uporabniških vlog in pravic,
- omejevanje dostopa do podatkov in funkcionalnosti glede na vlogo uporabnika in stanje presoje.

Posebna pozornost mora biti namenjena preprečevanju nepooblaščenega dostopa do podatkov drugih uporabnikov, destinacij ali ponudnikov.

Zahteve so opisane v sekciji »3.2.2 Integracija za upravljanje uporabnikov (priprava na SSO)«.

2.10.2 Varovanje podatkov

Sistem mora zagotavljati varovanje podatkov v vseh fazah njihovega življenjskega cikla, vključno z:

- vnosom podatkov,
- shranjevanjem,
- obdelavo,
- izmenjavo z drugimi sistemi.

Tehnična zasnova mora preprečevati:

- izgubo podatkov,
- nepooblaščene spremembe podatkov,
- nepooblaščen vpogled v podatke.

Posebej pomembno je varovanje dokumentov in dokazil, ki jih uporabniki nalagajo v sistem, ter zagotavljanje, da so ti dostopni izključno pooblaščenim osebam.

Spodaj je naveden KPI, ki ga mora izvajalec doseči pred prevzemom.

KPI: Uspešnost avtorizacijskih kontrol pred uvedbo Faze 1
Delež testnih scenarijev, kjer sistem zavrne dostop uporabnikom brez ustreznih pravic.
Zahtevana vrednost: 100 %.

2.10.3 Revizijska sled in sledljivost

Sistem mora omogočati revizijsko sled za ključne operacije in spremembe, zlasti:

- spremembe podatkov in poslovnih stanj,
- oddaje in potrditve presoj,
- ocenjevanje in zaključevanje postopkov,

- upravljanje uporabnikov in pravic.

Revizijska sled mora omogočati pregled nad tem, kdo je izvedel posamezno akcijo, kdaj in v kakšnem kontekstu, kar je ključno za zagotavljanje transparentnosti, zaupanja in skladnosti z internimi pravili naročnika.

Spodaj je naveden KPI, ki ga mora izvajalec doseči pred prevzemom.

KPI: Revizijska sled za vse ključne akcije v produkciji Faze 1
Delež akcij (oddaja presoje, sprememba ocene, administrativne spremembe), ki so zapisane v revizijski dnevnik.
Zahtevana vrednost: 100 %.

2.10.4 Varnost integracij

Integracije z drugimi sistemi morajo biti zasnovane varno in nadzorovano. Tehnična zasnova mora zagotavljati:

- nadzorovan dostop do integracijskih vmesnikov,
- jasno definirane pravice in omejitve za izmenjavo podatkov,
- zaščito pred nepooblaščenimi ali zlonamernimi klici.

V primeru varnostnih incidentov ali zlorab mora biti omogočeno hitro omejevanje ali onemogočanje posameznih integracij brez vpliva na delovanje celotnega sistema.

2.10.5 Operativna varnost in vzdrževanje

Varnost sistema mora biti upoštevana tudi v kontekstu njegovega delovanja in vzdrževanja. Tehnična zasnova mora omogočati:

- varno izvajanje nadgradenj in vzdrževalnih posegov,
- omejen in sledljiv dostop skrbnikov sistema,
- preprečevanje nenamernih posegov v produkcijsko okolje.

Varnostni mehanizmi morajo biti zasnovani tako, da podpirajo stabilno in zanesljivo delovanje sistema, brez nepotrebnega oteževanja uporabe za končne uporabnike.

2.10.6 Upravljanje piškotkov (Cookie management)

Spletni uporabniški vmesnik bodoče rešitve Slovenia Green mora zagotavljati ustrezno upravljanje piškotkov (cookie management) v skladu z veljavno zakonodajo s področja varstva osebnih podatkov in elektronskih komunikacij.

Tehnična zasnova mora omogočati uporabo piškotkov, ki so nujno potrebni za varno in pravilno delovanje sistema (npr. vzdrževanje uporabniške seje, zaščita pred zlorabami), ter jasno ločevanje med nujnimi in nenujnimi piškotki.

Uporaba nenujnih piškotkov je dovoljena izključno na podlagi predhodno pridobljenega soglasja uporabnika. Rešitev mora omogočati, da uporabnik soglasje kadarkoli pregleda, spremeni ali prekliče, brez vpliva na osnovno uporabo sistema.

Upravljanje piškotkov mora biti izvedeno na pregleden in uporabniku razumljiv način ter ne sme po nepotrebnem negativno vplivati na uporabniško izkušnjo. Ta zahteva se nanaša izključno na uporabniški vmesnik rešitve in ne vpliva na notranje delovanje aplikacijskega ali podatkovnega sloja sistema.

2.11 Enoten sistem prijave s preostalim delom NIST (Single sign-in).

Enoten sistem prijave uporabnikov predstavlja pomemben strateški cilj z vidika poenostavitve uporabniške izkušnje, upravljanja identitet in dolgoročne konsistentnosti ekosistema informacijskih sistemov naročnika.

2.11.1 Obstoječe stanje avtentikacije in avtorizacije v NiST

NiST uporablja lasten mehanizem avtentikacije uporabnikov, ki je integriran v obstoječo aplikacijsko zasnovo sistema. Trenutna rešitev omogoča prijavo uporabnikov v okviru posamezne aplikacije, vendar **ne izpostavlja standardiziranega identitetnega ali avtentikacijskega vmesnika**, ki bi omogočal enostavno in varno vključitev zunanjih sistemov v enoten prijavni mehanizem.

NiST trenutno:

- ne razpolaga s centralnim identitetnim ponudnikom (*Identity Provider*), namenjenim uporabi zunanjih ali dodatnih aplikacij,
- ne izpostavlja standardiziranih protokolov za federirano avtentikacijo (npr. SAML, OpenID Connect) za zunanje sisteme,
- ne omogoča neposredne integracije na ravni SSO brez posegov v obstočo aplikacijsko logiko NiST.

Posledično v trenutnem stanju ne obstaja **tehnično ustrezen integracijski vmesnik**, ki bi omogočal implementacijo enotnega sistema prijave za novo rešitev Slovenia Green brez večjih posegov v obstoječo rešitev NiST.

2.11.2 Zahteve in izhodišča za morebitno nadgradnjo

Kljub navedenim omejitvam je enoten sistem prijave smiselna in zaželena dolgoročna usmeritev. Tehnična analiza zato opredeljuje zahteve, ki jih mora izpolnjevati morebitna nadgradnja sistema avtentikacije in avtorizacije v NiST, da bi bila integracija izvedljiva:

- vzpostavitev ali izpostavitev centralnega sistema za upravljanje identitet,
- podpora standardiziranim protokolom za avtentikacijo in avtorizacijo,
- jasno definirani uporabniški identifikatorji in življenjski cikel uporabniških računov,
- možnost federacije identitet med sistemi brez neposredne odvisnosti aplikacijskih kod.

2.11.3 Pristop v okviru bodoče rešitve Slovenia Green

Glede na trenutno stanje NiST tehnična analiza predvideva, da bo rešitev Slovenia Green v začetni fazi uporabljala samostojen sistem avtentikacije in avtorizacije, ki:

- je zasnovan skladno s sodobnimi varnostnimi praksami,
- omogoča jasno upravljanje uporabnikov, vlog in pravic,
- je tehnično pripravljen na kasnejšo integracijo v enoten sistem prijave.

Rešitev mora biti zasnovana tako, da:

- omogoča prehod na enoten SSO brez sprememb poslovne logike,
- omogoča mapiranje uporabnikov med sistemi,
- ne ustvarja dolgoročne tehnične blokade za povezovanje z NiST ali drugimi sistemi.

Enoten sistem prijave z NiST je prepoznan kot dolgoročno smiseln cilj, **vendar trenutno stanje NiST ne omogoča neposredne implementacije SSO brez predhodnih nadgradenj**. Zato je v okviru tega projekta predvidena samostojna rešitev za upravljanje uporabnikov, ki je arhitekturno in tehnično pripravljena na prihodnjo integracijo, ko bodo za to izpolnjeni pogoji na strani NiST.

2.12 Arhitekturna zasnova rešitve Slovenia Green in njena povezljivost z NiST

Rešitev Slovenia Green mora biti zasnovana kot samostojen informacijski sistem, ki je sposoben neodvisnega razvoja, uvajanja sprememb in vzdrževanja, hkrati pa omogoča povezovanje z informacijskim sistemom NiST.

Pri načrtovanju arhitekture mora izvajalec zagotoviti jasno ločitev funkcionalnih in podatkovnih odgovornosti med sistemoma ter takšno zasnovo rešitve, ki omogoča nadzorovano izmenjavo podatkov in funkcionalnosti med njima.

Arhitektura sistema Slovenia Green mora zlasti zagotavljati:

- jasno opredeljene funkcionalne meje sistema in njegovo samostojno delovanje;
- možnost povezovanja z informacijskim sistemom NiST prek standardiziranih integracijskih mehanizmov;
- možnost neodvisnega razvoja, nadgradenj in uvajanja novih verzij sistema Slovenia Green;
- arhitekturno pripravljenost na postopno širjenje integracij z informacijskim sistemom NiST v prihodnosti.

Pri tem mora biti rešitev zasnovana na način, ki omogoča dolgoročno interoperabilnost z informacijskimi sistemi naročnika ter ne ustvarja nepotrebnih tehnoloških ali razvojnih odvisnosti med posameznimi sistemi.

Podrobnejše zahteve glede integracijskih mehanizmov, vmesnikov in izmenjave podatkov so opredeljene v drugih relevantnih poglavjih tega dokumenta.

2.13 Uporaba umetne inteligence (AI)

Bodoča rešitev mora biti zasnovana tako, da omogoča uporabo in integracijo funkcionalnosti, podprtih z umetno inteligenco (AI), skladno s poslovno logiko in uporabniškimi scenariji, opisanimi v dokumentu »Funkcionalne specifikacije«.

Uporaba AI je v sistemu predvidena kot **podporni mehanizem**, namenjen izboljšanju učinkovitosti dela uporabnikov, preglednosti podatkov, kakovosti vsebinske obdelave ter napredni analitiki, pri čemer AI ne nadomešča odločanja uporabnikov, temveč deluje kot pomoč pri interpretaciji, povzetkih, predlogih in primerjavah.

Tehnična analiza ne določa konkretne AI tehnologije, ponudnika ali storitve. Izbira AI pristopov, modelov in zunanjih storitev bo predmet **odločitev v začetni fazi implementacijskega projekta**, ob upoštevanju takratnih tehnoloških možnosti, varnostnih zahtev, zakonodajnih omejitev ter proračunskih okvirov naročnika.

Ne glede na končno izbiro mora tehnična zasnova bodoče rešitve izpolnjevati naslednja izhodišča:
• arhitektura mora omogočati modularno vključevanje AI funkcionalnosti (npr. prek ločenih storitev ali API vmesnikov),
• uporaba AI mora biti konfigurabilna in razširljiva, brez vpliva na osnovno delovanje sistema v primeru, da AI ni aktiviran ali začasno ni na voljo,
• zagotovljena mora biti sledljivost uporabe AI, vključno z jasno označitvijo AI-podprtih predlogov, povzetkov ali analiz v uporabniškem vmesniku,
• sistem mora omogočati nadzor nad podatki, ki se uporabljajo za AI obdelavo, ter spoštovati zahteve varstva podatkov in informacijske varnosti.

Pri uporabi zunanjih AI-storitev (npr. velikih jezikovnih modelov – LLM, dostopnih prek API) veljajo naslednje zahteve:
• obdelava podatkov pri zunanjem ponudniku AI-storitve mora potekati izključno znotraj EU/EGP;
• ponudnik AI-storitve naročnikovih, zaupnih ali osebnih podatkov ne sme uporabljati za učenje ali izboljševanje svojih modelov;
• zaupni in osebni podatki se zunanjemu AI-ponudniku ne smejo posredovati brez ustreznih pogodbenih in tehničnih jamstev; vhodni podatki se minimizirajo in, kjer je mogoče, psevdonimizirajo;
• vsak zunanji ponudnik AI-storitve, ki obdeluje osebne podatke, mora biti vključen kot podobdelovalec s sklenjeno pogodbo o obdelavi podatkov z enakimi obveznostmi, kot veljajo za izvajalca;
• izvajalec mora pred uvedbo posamezne AI-funkcionalnosti naročniku predložiti opis toka podatkov (katere vrste podatkov, kam se posredujejo, kje se obdelujejo in hranijo) in oceno tveganj ter pridobiti potrditev naročnika.

Tehnična zasnova mora torej zagotoviti, da je uporaba umetne inteligence sestavni, a ne obvezni del sistema, ki se lahko postopno nadgrajuje in prilagaja glede na potrebe naročnika ter razvoj projekta.

2.14 Primer minimalne konfiguracije sistema

Za potrebe izvedbe in vzpostavitve rešitve Slovenia Green v okviru okolja NiST je v nadaljevanju podan primer minimalne konfiguracije sistema, ki predstavlja osnovno infrastrukturo za delovanje aplikacije v produkcijskem okolju. Konfiguracija vključuje ključne komponente strojne in programske opreme, potrebne za stabilno delovanje aplikacijskega in podatkovnega sloja, ter upošteva standardne zahteve glede redundance, varnosti in obvladljivosti.
Minimalna konfiguracija je pripravljena kot referenčni primer, ki temelji na tipični izvedbi sodobnih informacijskih sistemov v kontejnerskem (Kubernetes) in virtualizacijskem okolju ter vključuje elemente, ki so običajno predmet obračuna storitev podatkovnega centra (npr. procesorski viri, pomnilnik, podatkovna baza, varnostne kopije in osnovna visoka razpoložljivost).
Konfiguracija ne vključuje posebnih zahtev ali dodatnih virov, povezanih z uporabo umetne inteligence, temveč predstavlja osnovni izhodiščni scenarij.
Takšna konfiguracija predstavlja minimalno stabilno osnovo za produkcijsko delovanje sistema in omogoča nadaljnje nadgradnje glede na rast obsega podatkov, števila uporabnikov ali dodatne funkcionalne zahteve.

Primer minimalne konfiguracije za kontejnersko (K8s) okolje:
Za storitvi K8S in S3-kompatibileno hrambo se uporablja skupna strežniška gruča. Zaradi redundance in osnovnih zmogljivostnih zahtev mora gruča obsegati najmanj 6 strežnikov – po 3 "master" in 3 "worker" vozlišča. Infrastruktura mora biti zasnovana modularno, z možnostjo enostavne razširitve, skladno s prihodnjimi potrebami in projekcijami.
Ti strežniki poleg izvajanja storitev K8s hkrati služijo tudi kot S3-kompatibilna hramba podatkov.
Priporočene specifikacije za osnovno vozlišče:

- 2× Intel Xeon ali AMD EPYC CPU
- Vsaj 16 jeder na CPU
- Vsaj 512 GB RAM
- 2× SSD disk v RAID-1 za OS
- 2× 4TB NVMe disk (RAID1, oziroma odvisno od predvidene potrebe po kapaciteti shrambe)
- 1× dvovratna 25GbE omrežna kartica

Primer minimalne konfiguracije za virtualizacijsko okolje:
Virtualizacijska infrastruktura mora vključevati najmanj dva fizična strežnika (zaradi redundance), ki delujeta kot gostitelja za virtualne strežnike. Zaščita pred izpadom je obvezna.
Priporočene specifikacije za eno vozlišče:

- CPU: 2× Intel Xeon ali AMD EPYC
- vsaj 16 jeder na CPU
- vsaj 512 GB RAM
- 2× SSD v RAID-1 za operacijski sistem
- 1× dvovratna (dual-port) 25GbE omrežna kartica

Poleg gostiteljskih strežnikov je potreben tudi namenski strežnik za hrambo virtualnih strežnikov in podatkov, ki se na teh strežnikih nahajajo. Priporočamo uporabo strežnika za blokovno shrambo z naslednjimi lastnostmi:

- Dvojni (dual) krmilnik za zagotavljanje redundance
- 44TB (RAW) SSD storage DRAID6
- 54TB (RAW) HDD (nearline SAS) storage DRAID6
- 2× dvovratna 25GbE omrežna kartica (multipathing)

Ocenjene minimalne zahteve glede prostora za shranjevanje so:

- **Aplikacijski sloj (Kubernetes okolje)**
  - 3× delovna vozlišča (worker nodes)
  - Vsako vozlišče: 4 vCPU, 16 GB RAM
  - Podpora za osnovno horizontalno skaliranje aplikacije
- **Podatkovni sloj**
  - Relacijska podatkovna baza (npr. PostgreSQL)
  - Minimalno: 2 vCPU, 8 GB RAM
  - Replikacija oziroma osnovna visoka razpoložljivost (primary + standby)
- **Shramba**
  - 100 GB SSD prostora za podatke in dokumentacijo
  - Ločen prostor za varnostne kopije
- **Redundanca in omrežje**
  - Load balancer za dostop do aplikacije
  - Redundantna povezava do interneta
  - TLS certifikati za varen dostop (HTTPS)
- **Osnovne sistemske storitve**
  - Monitoring (npr. Prometheus/Grafana)
  - Centralizirano beleženje dnevnikov (log management)
  - Avtomatizirani backup postopki.

2.15 Osnovne zahteve za podatkovni center

IT oprema mora biti nameščena v ustreznem podatkovnem centru, ki zagotavlja visoko razpoložljivost in ima v redundanci vse ključne sisteme. Podatkovni center mora izpolnjevati najmanj naslednje zahteve:

1. Podatkovni center z ustreznimi kapacitetami (zadostna fizična in energetska zmogljivost za predvideno IT opremo in morebitne prihodnje razširitve)

2. Hlajenje v konfiguraciji N+1, z neodvisnimi hladilnimi enotami in nadzorom temperature ter vlage
3. Napajanje preko UPS naprav z redundanco 2N, ki omogoča neprekinjeno napajanje ob izpadu enega sistema
4. Rezervno napajanje z dizelskim agregatom, ki se samodejno vključi ob izpadu primarnega napajanja
5. Internetna povezljivost preko več (vsaj dveh) neodvisnih ponudnikov, z ločenimi optičnimi potmi in ustrezno pasovno širino
6. Implementirani sistemi za fizično varovanje, vključno z videonadzorom, elektronskim nadzorom pristopa in vodenjem dnevnikov dostopa
7. Alarmni sistem z več-conskimi senzorji gibanja, požara, izlitja vode in temperaturnih odstopanj
8. Zagotavljanje 24/7 prisotnosti dežurne ekipe za nujne intervencije v primeru izpadov ali okvar
9. Zagotavljanje SLA z najmanj 99,9 % razpoložljivosti za ključne storitve podatkovnega centra
10. Stalni (24/7) monitoring kritičnih IT in podpornih sistemov, z avtomatiziranim obveščanjem ob zaznanih anomalijah
11. Veljavni certifikati ISO 9001, ISO 14001 in ISO 27001 ali enakovredni standardi in dokazila, ki zagotavljajo ustrezno raven organizacije.

3 Integracije

Poglavje opredeljuje zahteve glede arhitekture, uporabljenih tehnologij, oblačne infrastrukture in tehničnih značilnosti bodoče rešitve. Namen teh zahtev je zagotoviti tehnično kompatibilnost z obstoječim sistemom NiST, ob hkratni ohranitvi samostojnosti rešitve, njene obvladljivosti in možnosti neodvisnega razvoja ter vzdrževanja.

3.1 Seznam integracij bodoče rešitve

Sekcija določa in opisuje informacijske sisteme, s katerimi se bo bodoča rešitev *Slovenia Green* smiselno povezovala. Seznam integracij predstavlja osnovo za nadaljnjo tehnično analizo integracij, ter za načrtovanje arhitekture, podatkovnih tokov in varnostnih mehanizmov sistema.

Pri opredeljevanju integracij izhajamo iz dejanskih poslovnih potreb sistema, njegove vloge v ekosistemu NiST ter izhodišč, da mora biti rešitev samostojna, a pripravljena na povezovanje z drugimi sistemi.

Integracije bodoče rešitve morajo:

- podpirati poslovne procese presoje, ocenjevanja in poročanja,
- omogočati izmenjavo podatkov z NiST in drugimi relevantnimi sistemi,
- ne ustvarjati neposredne tehnične ali izvedbene odvisnosti,
- biti zasnovane modularno in nadzorovano.

Vse integracije so predvidene kot **tehnično ločene povezave**, ki jih je mogoče postopno uvajati, nadgrajevati ali po potrebi zamenjati.

V nadaljevanju je podan pregled ključnih integracij, razdeljenih glede na njihov pomen in stopnjo nujnosti.

3.1.1 Integracije znotraj ekosistema NiST

Namen integracije z informacijskim sistemom NiST je izmenjava izbranih podatkov, ki so relevantni za oba sistema, predvsem:

- osnovni podatki o destinacijah,
- osnovni podatki o ponudnikih,
- agregirani ali izbrani rezultati presoj in kazalnikov za potrebe analitike in poročanja.

Integracija je predvidena kot **enostranska ali dvosmerna**, odvisno od konkretnega nabora podatkov, pri čemer je potrebno jasno določiti, kateri sistem je »master« za posamezno podatkovno domeno.

3.1.2 Integracija sistema za upravljanje uporabnikov (SSO)

Integracija omogoča uporabo centralnega sistema za upravljanje identitet in enoten sistem prijave uporabnikov med sistemi NiST in Slovenia Green, ko bo na volji v NiST-u.

V trenutni fazi integracija ni tehnično izvedljiva zaradi omejitev na strani NiST, vendar mora biti rešitev:

- arhitekturno pripravljena na tovrstno integracijo,
- sposobna mapiranja uporabnikov in vlog,
- neodvisna v svojem osnovnem delovanju.

Ta integracija bo relevantna predvsem za:

- upravljanje uporabnikov,
- uporabniške tokove prijave in dostopov,
- dolgoročno uporabniško izkušnjo.

3.1.3 Integracije za obdelavo in hrambo dokumentov

Sistem Slovenia Green upravlja z dokazili in dokumentacijo, ki jih uporabniki nalagajo v okviru presoj.

Integracija lahko omogoča:

- varno hrambo večjih količin dokumentov,
- dolgoročno arhiviranje,
- razbremenitev aplikacijskega in podatkovnega sloja.

Ta integracija je relevantna za:

- funkcionalne zahteve vnosa dokazil,
- uporabniške vmesnike za nalaganje in pregled dokumentov.

3.1.4 Integracije za analitiko in poročanje

Namen integracije z analitičnimi in poročevalskimi orodji (znotraj NiST ali zunanjimi) je omogočiti:

- nadaljnjo obdelavo podatkov o presojah in kazalnikih,
- primerjalne analize skozi čas,
- podporo strateškemu odločanju.

Integracija je relevantna za:

- analitične prikaze in poročila,
- agregacijo podatkov iz več virov,
- podporo poročanju naročnika.

3.1.5 Integracije z zunanjimi informacijskimi viri

Po potrebi je možno vključiti integracije z zunanjimi informacijskimi sistemi, ki zagotavljajo referenčne ali podporne podatke (npr. šifranti, klasifikacije, javne evidence).

Takšne integracije:

- niso nujne za osnovno delovanje sistema,
- se uvajajo postopno glede na potrebe,
- ne smejo vplivati na stabilnost osnovnih funkcionalnosti.

Identificirane integracije predstavljajo realen in obvladljiv nabor povezav, ki podpirajo delovanje bodoče rešitve Slovenia Green brez ustvarjanja nepotrebnih tehničnih tveganj ali odvisnosti.

3.1.6 Integracija s sistemom HubSpot (CRM)

V okviru bodoče rešitve Slovenia Green je predvidena možnost integracije s sistemom za upravljanje odnosov s strankami HubSpot. Namen integracije je podpora komunikacijskim, promocijskim ali analitičnim procesom naročnika, ki so vezani na delo z uporabniki, deležniki ali drugimi ciljnimi skupinami.

V trenutni fazi projekta podrobne tehnične specifikacije integracije še niso razpoložljive. Integracija je zato v tem dokumentu obravnavana na konceptualni ravni in je vključena v seznam potrebnih integracij kot izhodišče za nadaljnjo obravnavo v implementacijski fazi projekta.

Bodoča rešitev mora biti arhitekturno in tehnično zasnovana tako, da omogoča izvedbo integracije s sistemom HubSpot brez večjih posegov v osnovno arhitekturo, ob upoštevanju standardnih integracijskih pristopov in varnostnih mehanizmov.

3.1.7 Pridobivanje podatkov o registriranih naravnih lepotah

Za izpolnjevanje kriterija 2.1 je zaželeno pridobivanje podatkov o registriranih naravnih lepotah v okviru destinacije. Podatki o registriranih naravnih lepotah (naravnih vrednotah) v Republiki Sloveniji so javno dostopni v okviru uradnega registra, ki ga vodi Agencija RS za okolje (ARSO). Register vsebuje prostorske podatke o območjih naravnih vrednot, kot so zavarovani naravni pojavi, geomorfološke posebnosti, vodni viri, ekosistemi ter druga območja naravne dediščine.
Uradni dostop do teh podatkov je omogočen preko državnega okoljskega geografskega portala Atlas okolja, kjer je sloj »Register naravnih vrednot (območja)« objavljen kot javno dostopna kartografska storitev:

Atlas okolja (ARSO): http://gis.arso.gov.si/atlasokolja/

Opis postopka pridobivanja podatkov
Programska rešitev mora omogočati, da uporabnik izbere posamezno občino Republike Slovenije, sistem pa na podlagi uradnih prostorskih podatkov pridobi in prikaže seznam naravnih vrednot, ki se nahajajo na območju izbrane občine.
V ta namen mora izvajalec zagotoviti naslednji logični proces:

- Najprej sistem določi prostorski obseg izbrane občine, pri čemer uporablja uradne administrativne meje občin kot geometrijski poligon. Občina je lahko izbrana preko seznama ali vnosa naziva.
- Nato sistem dostopi do sloja naravnih vrednot, objavljenega v Atlasu okolja, in pridobi vse prostorske objekte naravnih vrednot skupaj z njihovimi osnovnimi atributi (npr. naziv, identifikator, vrsta naravne vrednote).
- V naslednjem koraku sistem izvede prostorsko primerjavo med območjem občine in območji naravnih vrednot. V rezultat se vključijo vse naravne vrednote, ki ležijo znotraj občine oziroma se z njenim območjem prostorsko prekrivajo.

- Na koncu mora sistem uporabniku prikazati seznam naravnih vrednot za izbrano občino v pregledni obliki (tabela ali seznam), pri čemer mora biti za vsako naravno vrednoto prikazan najmanj naziv ter osnovna klasifikacija. Po potrebi mora biti omogočen tudi izvoz rezultatov.

Pričakovani rezultat
Za izbrano občino (npr. Nova Gorica ali Ajdovščina) mora programska rešitev vrniti seznam registriranih naravnih lepot na njenem območju, kot so evidentirane v uradnem registru ARSO, ter s tem omogočiti pregled naravne dediščine po posameznih lokalnih skupnostih.

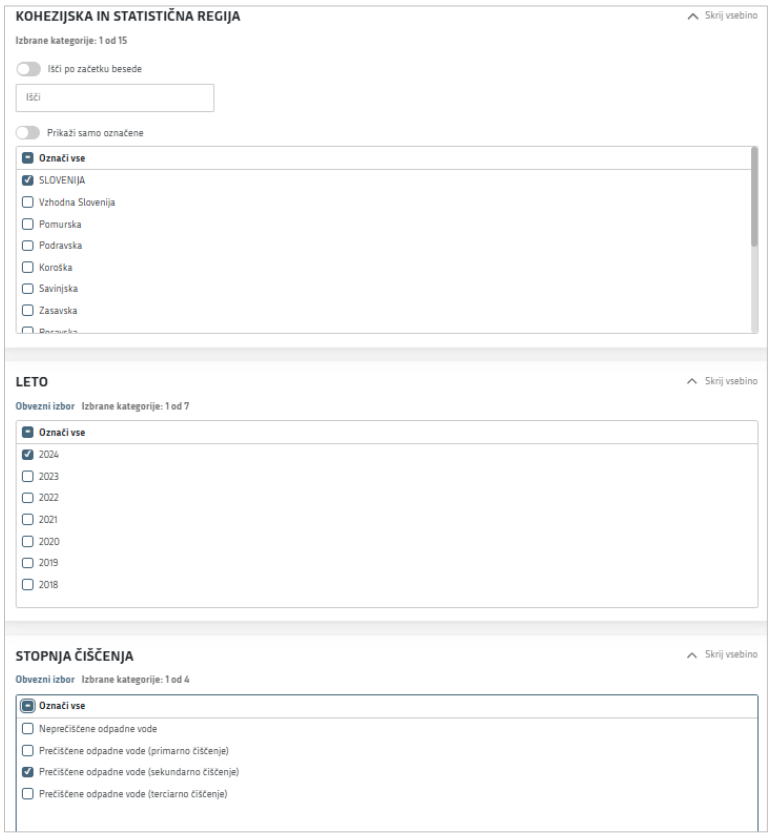
3.1.8 Pridobivanje podatkov o stopnji čiščenja odpadnih voda

Za potrebe izpolnjevanja kriterija 3.7 je poleg kazalnika na ravni ZSST destinacije (prepis iz poglavja 1.15) potrebno vključiti tudi nacionalni referenčni podatek o stopnji čiščenja odpadnih voda v Republiki Sloveniji.
V ta namen se v sistem uporabi indikator: **Odstotek (%) odpadnih voda, ki so očiščene najmanj do sekundarne stopnje**, Slovenija, zadnje razpoložljivo leto.

Vir podatkov je Statistični urad Republike Slovenije (SURS), link:
https://pxweb.stat.si/SiStatData/pxweb/sl/Data/-/2750230S.px

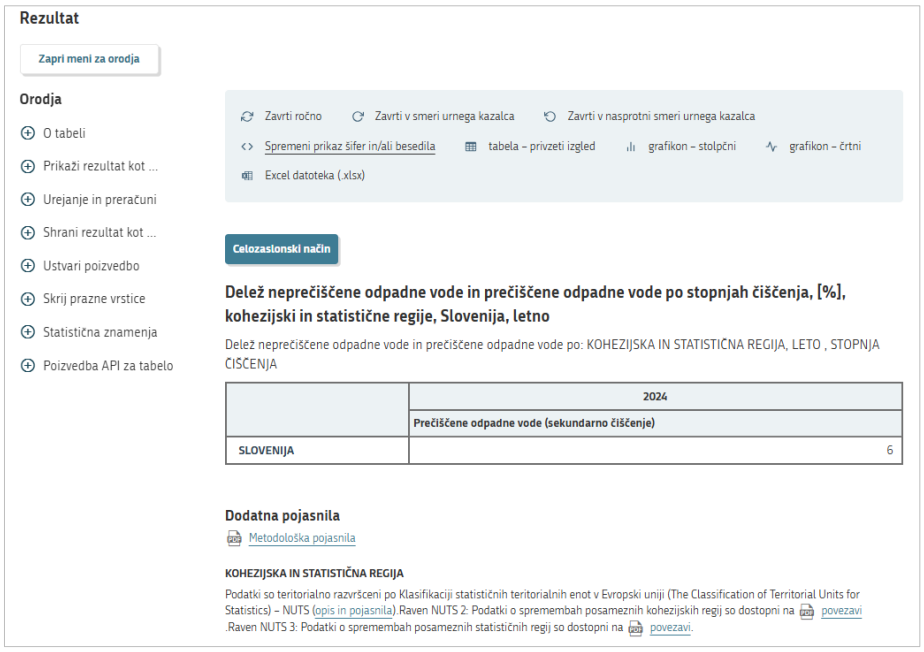
Postopek pridobivanja podatka iz SURS (PXWeb) mora biti v celoti avtomatiziran in sprožen iz aplikacije, brez ročnega izbiranja filtrov na strani SURS.
Aplikacija v integracijskem sloju naj izvede avtomatizirano zajemanje podatkov iz PXWeb v naslednjih korakih:

- **Dostop do vira:** sistem odpre uradno PXWeb stran indikatorja in preveri dosegljivost ter pričakovano strukturo strani (prisotnost filtrov in rezultatnega prikaza).
- **Programska nastavitev filtrov:** sistem samodejno nastavi potrebne izbire v filtrih (npr. stopnja čiščenja ter zadnje razpoložljivo leto). Zadnje razpoložljivo leto določi tako, da iz ponujenih možnosti izbere najnovejšo vrednost.



Slika 1: Izbrani parametri iskanja

- **Izvedba poizvedbe:** sistem sproži generiranje rezultatov (enakovredno potrditvi obrazca na PXWeb), pri čemer zagotovi, da se poizvedba izvede za pravilno kombinacijo izbranih parametrov.
- **Prevzem rezultata:** sistem na rezultatni strani prebere iskano vrednost iz prikaza/tabele (odstotek), kot je definirano v uporabniškem primeru, ter jo prenese v aplikacijo kot nacionalni referenčni podatek.



Slika 2: Rezultat iskanja

Ker se spletni vmesnik SURS lahko spremeni (spremembe strukture HTML, poimenovanja filtrov, vrstni red parametrov ali oblika rezultatne tabele), mora biti implementirano obvladovanje napak:

- če filtrov ni mogoče programsko nastaviti ali rezultatne vrednosti ni mogoče razbrati, sistem integracijo prekine, napako zabeleži in uporabniku prikaže jasno sporočilo, da samodejni prevzem podatka trenutno ni mogoč zaradi spremembe ali nedosegljivosti zunanjega vira,
- aplikacija mora v tem primeru še vedno delovati naprej (degradirano delovanje), pri čemer se za prikaz kriterija uporabi zadnja uspešno pridobljena vrednost (če obstaja) ali pa se uporabniku označi, da nacionalni podatek trenutno ni na voljo.

3.1.9 Pridobivanje podatkov o dostopnosti z javnim potniškim prometom

Za kriterij 3.15 mora aplikacija samodejno pridobiti podatek o dostopnosti z javnim potniškim prometom (JPP) iz zunanjega vira Observatorij mobilnosti, brez ročnega posega uporabnika. Postopek deluje tako, da integracijski sloj aplikacije odpre stran z indikatorjem in programsko izvede zajem podatkov: identificira ustrezne vhodne parametre (izbor občine) ter iz prikaza rezultatov prebere vrednost kazalnika za izbrano občino. Aplikacija mora pri tem upoštevati, da je lahko destinacija sestavljena iz več občin – v tem primeru se zajem izvede za vsako občino posebej, sistem pa izpiše vrednost za vsako občino ločeno.

Na podlagi pridobljene vrednosti za občino in primerjalne referenčne vrednosti (slovensko povprečje), aplikacija samodejno generira interpretacijo:
“Dostopnost destinacije z javnim potniškim prometom je NAD/POD slovenskim povprečjem.”

Če je destinacija sestavljena iz več občin, aplikacija izvede primerjavo in zaključno interpretacijo za vsako občino posebej (tj. NAD/POD za vsako občino).

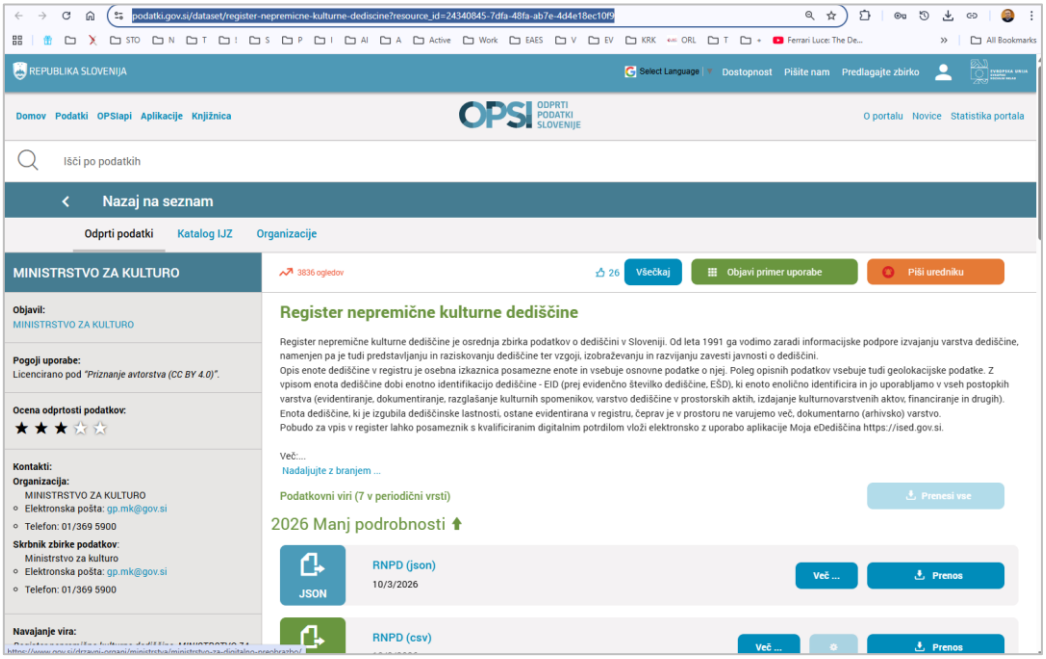
Ker gre za zunanji spletni vir, ki se lahko spremeni (struktura strani, poimenovanja občin, način prikaza rezultatov), mora biti integracija odporna na napake: če podatka ni mogoče samodejno prebrati, sistem napako zazna in uporabniku prikaže jasno sporočilo, da prevzem trenutno ni mogoč zaradi sprememb ali nedosegljivosti vira, pri čemer osnovno delovanje aplikacije ne sme biti blokirano (degradirano delovanje).
Link: https://www.observatorij-mobilnosti.si/podatki/dostopnost-do-javnega-potniskega-prometa



Slika 3: Pridobivanje podatkov za občino Ajdovščina

3.1.10 Podatki snovne in nesnovne kulturne dediščine

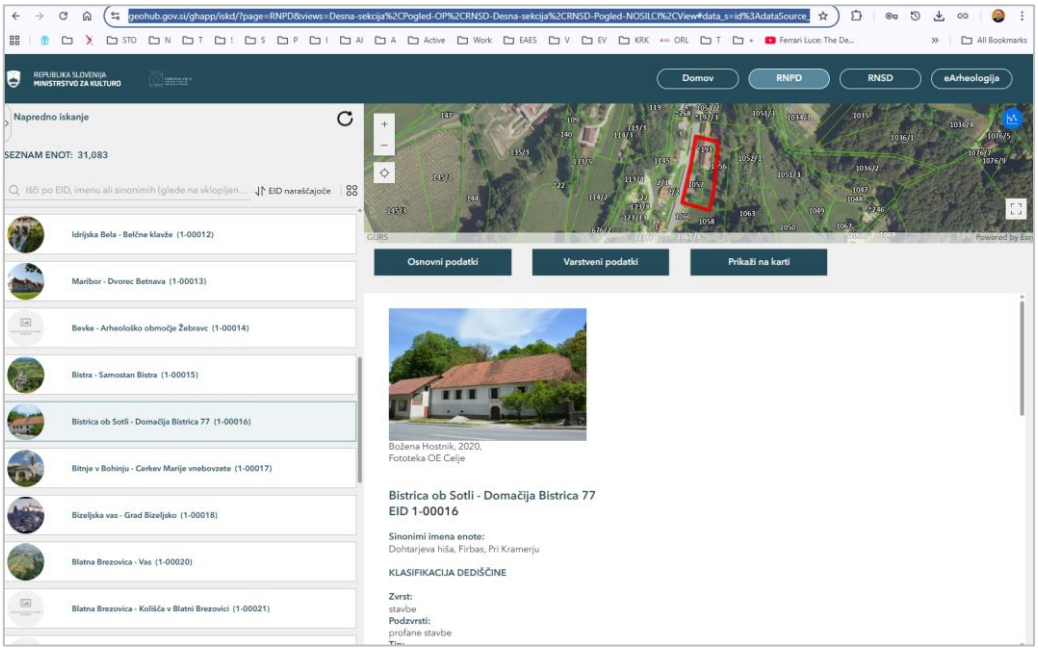
Za potrebe spremljanja kulturne dediščine mora aplikacija omogočati samodejno pridobivanje podatkov o snovni (nepremični) in nesnovni kulturni dediščini iz javno dostopnih državnih evidenc. Vir podatkov predstavljajo uradni registri, objavljeni na portalu odprtih podatkov Republike Slovenije ter na geografskem portalu Geohub, kjer so objavljeni registri kulturne dediščine. Postopek deluje tako, da integracijski sloj aplikacije dostopa do javno objavljenih registrov ter iz njih pridobi seznam enot kulturne dediščine. Pri tem sistem upošteva lokacijo oziroma območje destinacije ter omogoča prikaz relevantnih enot kulturne dediščine, ki se nanašajo na izbrano območje.



Slika 4: Register nepremične kulturne dediščine

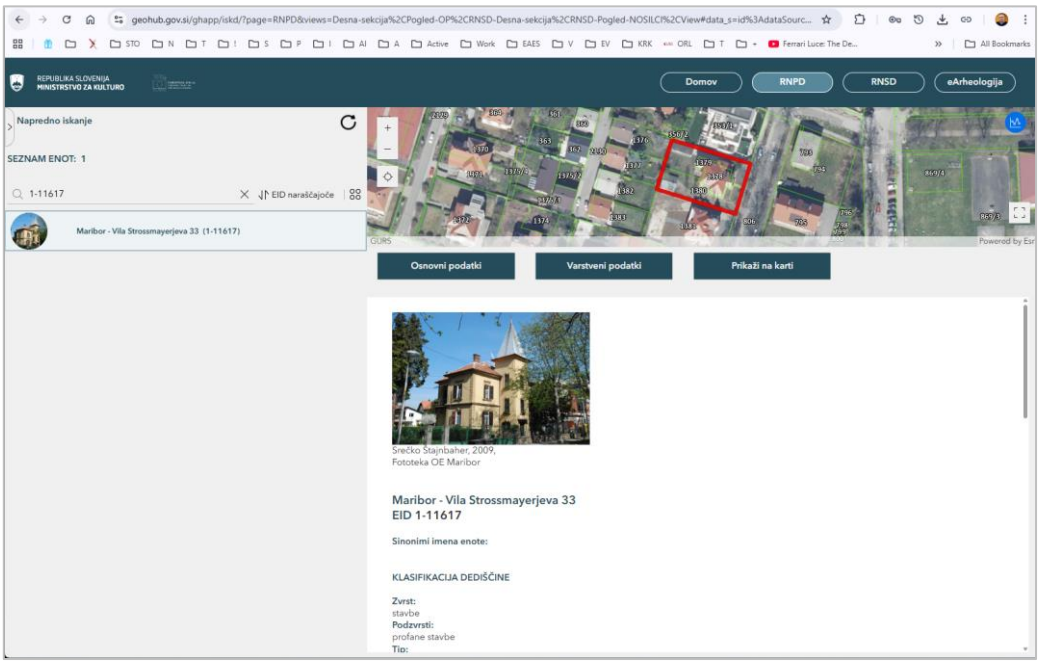
Za snovno (nepremično) kulturno dediščino aplikacija uporablja register nepremične kulturne dediščine, ki je objavljen kot javni podatkovni vir: https://podatki.gov.si/dataset/register-nepremicne-kulturne-dediscine?resource_id=24340845-7dfa-48fa-ab7e-4d4e18ec10f9

Poleg podatkovnega vira je na voljo tudi pregledovalnik na portalu Geohub, kjer je mogoče pregledovati posamezne enote kulturne dediščine in njihove osnovne podatke: https://geohub.gov.si/ghapp/iskd/?page=RNPD&views=Desna-sekcija%2CPogled-OP%2CRNSD-Desna-sekcija%2CRNSD-Pogled-NOSILCI%2CView#data_s=id%3AdataSource_2-MK_RNSD_4402-3623%3A16%2Cid%3AdataSource_1-MK_RNPD_3598-3652%3A1236



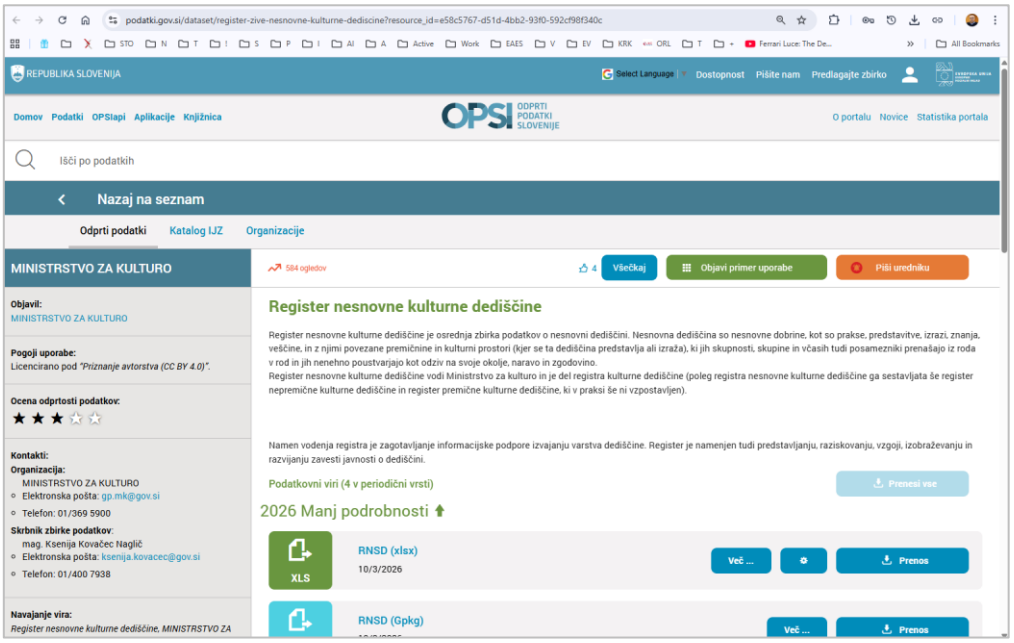
Slika 5: Seznam nepremične kulturne dediščine

Iskanje se izvede z uporabo EID kode, ki se nahaja v registru.



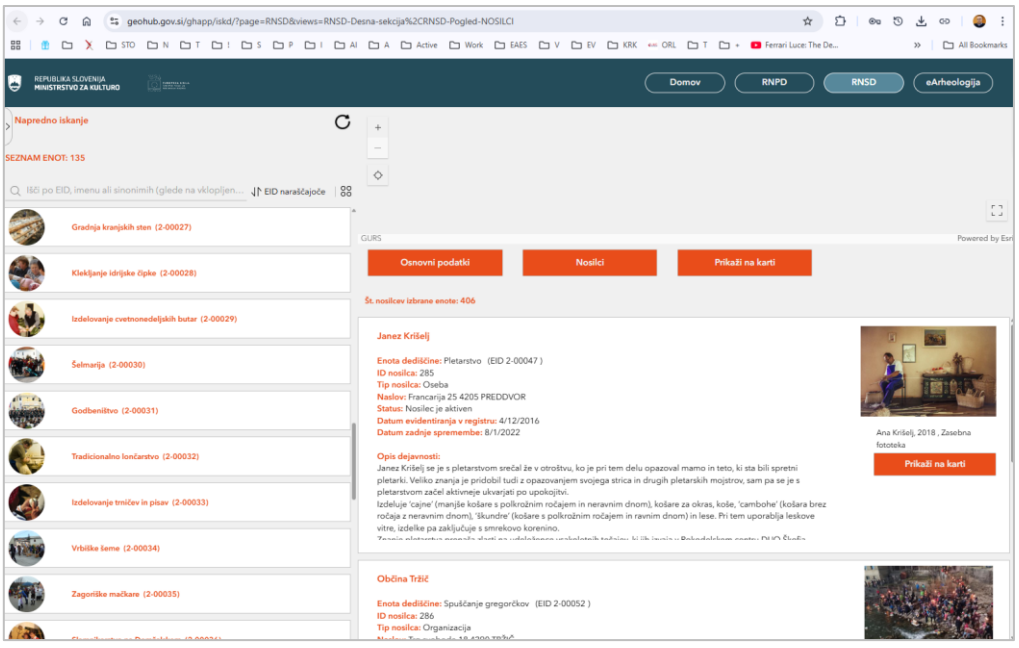
Slika 6: Rezultat iskanja nepremične kulturne dediščine

Za nesnovno kulturno dediščino aplikacija uporablja register žive nesnovne kulturne dediščine, ki je prav tako objavljen kot javni podatkovni vir: https://podatki.gov.si/dataset/register-zive-nesnovne-kulturne-dediscine?resource_id=e58c5767-d51d-4bb2-93f0-592cf98f340c



Slika 7: Register nesnovne kulturne dediščine

Dodatni pregled teh podatkov je mogoč tudi preko pregledovalnika Geohub: https://geohub.gov.si/ghapp/iskd/?page=RNSD&views=RNSD-Desna-sekcija%2CRNSD-Pogled-NOSILCI#data_s=id%3AdataSource_2-MK_RNSD_4402-3623%3A16

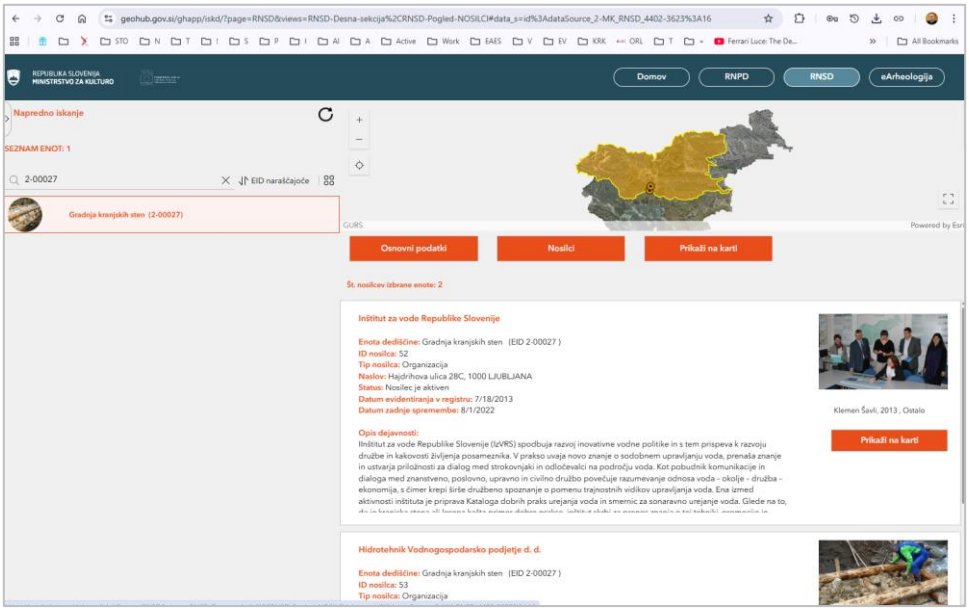


Slika 8: Seznam nesnovne kulturne dediščine

Iskanje se izvede z uporabo EID kode, ki se nahaja v registru.

Ker gre za zunanje podatkovne vire, ki se lahko s časom spremenijo (npr. struktura podatkov, način dostopa ali vsebina registrskih zapisov), mora biti integracija zasnovana robustno in odporna na napake. V primeru, da podatkov ni mogoče samodejno pridobiti (npr. zaradi nedosegljivosti vira ali spremembe strukture), mora sistem napako zaznati, jo zabeležiti in uporabniku prikazati jasno sporočilo, da trenutno ni mogoče samodejno pridobiti podatkov iz zunanjega vira.

V takšnem primeru mora aplikacija še naprej normalno delovati (degradirano delovanje), pri čemer se uporabniku lahko prikaže zadnji uspešno pridobljen seznam ali pa informacija, da podatki trenutno niso na voljo.



Slika 9: Rezultat iskanja nesnovne kulturne dediščine

3.1.11 Preverjanje zadovoljstva prebivalcev

Pri spremljanju zadovoljstva prebivalcev z razvojem turizma je pomembno, da anketa vključuje **statistično ustrezen vzorec prebivalcev**. Velikost vzorca mora biti dovolj velika, da rezultati ankete zanesljivo predstavljajo mnenje celotne populacije.

Velikost potrebnega vzorca je odvisna predvsem od:

- skupnega števila prebivalcev (velikost populacije),
- želene stopnje statistične zanesljivosti,
- dovoljene statistične napake.

Enak način izračuna uporabljajo tudi različni spletni kalkulatorji velikosti vzorca, na primer kalkulator na platformi SurveyMonkey. Izvajalec mora implementirati rešitev, da se enak izračun **izvede neposredno v aplikaciji**, brez uporabe zunanjih orodij.

Vhodni podatek

Program prejme naslednji podatek:

- **N** – skupno število prebivalcev destinacije (velikost populacije).

Ta podatek predstavlja celotno populacijo, iz katere želimo pridobiti mnenje z anketo.

Statistični parametri

Za izračun se uporabljajo standardni statistični parametri, ki se pogosto uporabljajo pri družboslovnih raziskavah:

- **Stopnja zaupanja (confidence level): 95 %**
- **Z-vrednost: 1.96**
- **Dovoljena statistična napaka (margin of error): 5 %**
- **Delež populacije (p): 0.5**

Delež 0.5 se uporablja kot konservativna ocena, saj daje največjo potrebno velikost vzorca in tako zagotavlja zadostno reprezentativnost.

Koraki izračuna

1. Izračun vzorca za zelo veliko populacijo

Najprej se izračuna teoretična velikost vzorca za zelo veliko populacijo:

$$n_0 = \frac{Z^2 \cdot p \cdot (1-p)}{e^2}$$

kjer:

- **Z** = 1.96
- **p** = 0.5
- **e** = 0.05

Rezultat tega izračuna je približno:

$$n_0 \approx 384$$

Ta vrednost predstavlja velikost vzorca za zelo veliko populacijo.

2. Popravek za končno populacijo

Ker ima vsaka destinacija omejeno število prebivalcev, je treba rezultat prilagoditi dejanski velikosti populacije. Uporabi se tako imenovana **korekcija za končno populacijo**:

$$n = \frac{n_0}{1 + \frac{n_0 - 1}{N}}$$

kjer:

- **N** – število prebivalcev
- **n** – priporočena velikost vzorca

Dobljeni rezultat se **zaokroži navzgor na najbližje celo število**.

3. Minimalni zahtevani vzorec

Za potrebe ocenjevanja se upošteva, da mora anketa doseči **vsaj 80 % statistično priporočene velikosti vzorca**.

Minimalno zahtevano število odgovorov se izračuna:

$$n_{min} = 0.8 \times n$$

Tudi ta rezultat se **zaokroži navzgor** na celo število.

Rezultat izračuna

Program mora vrniti dva podatka:

- **Priporočena velikost vzorca (n)** – število prebivalcev, ki bi jih bilo optimalno vključiti v anketo.
- **Minimalno zahtevano število odgovorov (n_min)** – najmanjše število odgovorov, ki še zagotavlja statistično ustreznost.

Primer izračuna

Če ima destinacija **10.000 prebivalcev**:

1. izračun priporočene velikosti vzorca:

$$n \approx 370$$

2. minimalno število potrebnih odgovorov:

$$n_{min} = 0.8 \times 370 = 296$$

V tem primeru mora anketa zbrati **vsaj 296 odgovorov**.

3.2 Tehnična osnova integracij

V tem poglavju je podana tehnična osnova identificiranih integracij bodoče rešitve Slovenia Green. Analiza temelji na razpoložljivih tehničnih specifikacijah, zlasti na priloženi OpenAPI specifikaciji in opisu podatkovnih struktur, ter predstavlja podlago za izvedbeno fazo projekta.

Integracije so zasnovane tako, da omogočajo standardizirano, nadzorovano in razširljivo izmenjavo podatkov, brez neposredne odvisnosti med sistemi.

3.2.1 Integracija z informacijskim sistemom NiST

Podatkovni vir: Informacijski sistem NiST.

Namen integracije:
Izmenjava izbranih podatkov, ki so relevantni za:

- osnovne podatke o destinacijah in ponudnikih,
- agregirane rezultate presoj in kazalnikov,
- podporo analitiki in poročanju znotraj NiST.

Tehnična izvedba integracije:
Integracija je predvidena kot **REST API** integracija, skladna z OpenAPI specifikacijo. Komunikacija poteka prek standardnih HTTP metod (GET, POST, PUT), pri čemer so podatkovne strukture jasno definirane v OpenAPI specifikaciji.

Integracijski sloj rešitve Slovenia Green mora omogočati:

- klicanje API vmesnikov NiST,
- validacijo vhodnih in izhodnih podatkov,
- ločeno obravnavo integracijskih napak.

Podatkovna struktura:
Podatkovni modeli so definirani v OpenAPI specifikaciji in vključujejo:

- identifikatorje entitet,
- osnovne opisne podatke,
- povezave med entitetami.

V tehnični implementaciji se uporablja mapiranje podatkovnih struktur med sistemoma, pri čemer se ohranja jasna razmejitev odgovornosti glede lastništva podatkov.

Frekvenca osveževanja podatkov:

- periodično (npr. sinhronizacija referenčnih podatkov),
- ali dogodkovno, kjer je to smiselno (npr. ob zaključku presoje).

Prožilec osvežitve:

- časovno sproženi procesi,
- poslovni dogodki (zaključek ali potrditev postopka).

Obravnava napak:
V primeru nedosegljivosti NiST mora sistem omogočati:

- ponovne poskuse (*retry*),
- beleženje napak,
- nadaljevanje osnovnega delovanja sistema brez prekinitve ključnih funkcionalnosti.

3.2.2 Integracija za upravljanje uporabnikov (priprava na SSO)

Podatkovni vir: Interni sistem za upravljanje uporabnikov v rešitvi Slovenia Green; v prihodnje potencialno centralni sistem NiST.

Namen integracije: Omogočiti prihodnjo integracijo v enoten sistem prijave uporabnikov, ko bo na strani NiST na voljo ustrezen identitetni vmesnik.

Tehnična izvedba integracije:
V trenutni fazi integracija ni aktivna, vendar mora biti tehnična zasnova pripravljena na:

- integracijo prek standardiziranih protokolov (npr. OAuth 2.0 / OpenID Connect),
- federirano avtentikacijo uporabnikov.

Podatkovna struktura:
Uporabniški podatki vključujejo:

- enolični identifikator uporabnika,
- vlogo oziroma nabor pravic,
- povezavo z destinacijo ali ponudnikom.

Frekvenca in prožilci:

- ob prijavi uporabnika,
- ob spremembah uporabniških pravic.

3.2.3 Integracija za hrambo in upravljanje dokumentov

Podatkovni vir: Interni modul ali zunanji sistem za hrambo dokumentov.

Namen integracije: Upravljanje dokazil in dokumentacije, ki so del presoj in ocenjevalnih postopkov.

Tehnična izvedba integracije:
Integracija poteka prek API vmesnikov za:

- nalaganje dokumentov,
- dostop do dokumentov,
- metapodatke o dokumentih.

Podatki o dokumentih (metapodatki) so shranjeni v jedrnem podatkovnem modelu, dejanske datoteke pa v namenskem sistemu za hrambo.

Frekvenca osveževanja: Dogodkovna (ob nalaganju, posodobitvi ali brisanju dokumenta).

3.2.4 Integracije za analitiko in poročanje

Podatkovni vir: Podatkovni sloj rešitve Slovenia Green.

Namen integracije: Omogočiti uporabo podatkov za analitične in poročevalske namene, bodisi znotraj NiST bodisi v zunanjih orodjih.

Tehnična izvedba integracije:

- izvoz podatkov prek API vmesnikov,
- ali neposreden dostop do agregiranih podatkov prek vnaprej pripravljenih poizvedb.

Frekvenca osveževanja:

- periodično,
- ali na zahtevo.

3.2.5 Splošna načela tehnične izvedbe integracij

Za vse integracije veljajo naslednja skupna tehnična izhodišča:

- uporaba REST API pristopa,
- skladnost z OpenAPI specifikacijo,
- uporaba strukturiranih podatkovnih formatov (npr. JSON),
- ločitev integracijskega sloja od poslovne logike,
- nadzorovana obravnava napak in nedosegljivosti.

3.2.6 Prenos podatkov iz NiST v rešitev SG – nabor podatkov in API klici

Prenos podatkov iz NiST je izveden prek **Data API** vmesnika, definiranega v priloženi **OpenAPI specifikaciji (openapi.yml)**. Prenos je zasnovan kot klicanje posameznih podatkovnih sklopov prek namenskih endpointov (večinoma **POST**), pri čemer se zahteve (filtri) podajo v request body (npr. leta, destinacije, občine), rezultati pa se vračajo kot seznam objektov v JSON formatu.

Tabela podatkovnih sklopov in pripadajočih API endpointov

Podatkovni sklop (API podatki.xlsx)	Časovna raven	Teritorialna raven	Endpoint (openapi.yml)	Metoda	Request body (filter)	Response schema (glavni zapis)	Opomba
ZSST destinacije	statično	Slovenija	/api/data/zsst-destinations	POST	brez (brez filtrov)	ZSSTDestination[]	Referenčni šifrant destinacij + metapodatki.
ZSST ponudniki	statično	Slovenija	/api/data/zsst-providers	POST	brez (brez filtrov)	ZSSTProvider[]	Referenčni šifrant ponudnikov.
Prihodi turistov + Prenočitve	letno	destinacija	/api/data/arrivals-and-stays-yearly-by-zsst-destinations	POST	YearsDestinations (leta, destinacije)	ArrivalsStays[]	Vrača arrivals in stays po letu in destinaciji.
Povprečna dolžina bivanja	letno	destinacija	/api/data/average-length-of-stay-yearly-by-zsst-destinations-and-markets	POST	YearsDestinations (leta, destinacije)	(schema iz OpenAPI)	Vključuje dimenzijo **trga/market** v response (skladno z endpointom).
Trgi izvora turistov	statično	globalno	(posredno) /api/data/average-length-of-stay-yearly-by-zsst-destinations-and-markets	POST	YearsDestinations	(schema iz OpenAPI)	Seznam trgov je impliciten v response podatkih (če je potreben ločen šifrant, je to dodatna zahteva).
Sezonalnost	letno	občina	/api/data/seasonality-yearly-by-municipalities	POST	YearsMunicipalities (leta, občine)	(schema iz OpenAPI)	Sezonalnost na ravni občin.
Sezonalnost	letno	destinacija	/api/data/seasonality-yearly-by-zsst-destinations	POST	YearsDestinations (leta, destinacije)	(schema iz OpenAPI)	Sezonalnost na ravni ZSST destinacij.
Turistične kapacitete + Status kapacitet	letno	občina	/api/data/tourism-capacity-status-by-municipality-and-type	POST	MunicipalitiesOnly (občine)	(schema iz OpenAPI)	Kapacitete po tipu in statusu, po občini.
Zaposleni v turizmu	letno	destinacija	/api/data/employees-in-tourism-yearly-by-zsst-destinations	POST	YearsDestinations	(schema iz OpenAPI)	Zaposleni v turizmu po destinaciji in letu.
Ogljični odtis (letni)	letno	občina	/api/data/carbon-footprint-estimate-yearly-municipalities	POST	YearsMunicipalities	CarbonFootprintYearly[]	Ocena CO_2 (letno) po občini.
Ogljični odtis (mesečni)	mesečno	občina	/api/data/carbon-footprint-estimate-monthly-municipalities	POST	MonthsMunicipalities (meseci, občine)	CarbonFootprintMonthly[]	Ocena CO_2 (mesečno) po občini.

Podatkovni sklop (API podatki.xlsx)	Časovna raven	Teritorialna raven	Endpoint (openapi.yml)	Metoda	Request body (filter)	Response schema (glavni zapis)	Opomba
Ogljični odtis (mesečni)	mesečno	destinacija	/api/data/carbon-footprint-estimate-monthly-zsst-destinations	POST	Month...Destinations (meseci, destinacije)	CarbonFootprintMonthly[]	Ocena CO_2 (mesečno) po destinaciji.
Odpadki	letno	destinacija	/api/data/waste-yearly-by-zsst-destinations	POST	YearsDestinations	(schema iz OpenAPI)	Letni podatki o odpadkih po destinaciji.
Dostopne nastanitve	letno	občina	/api/data/accessible-accommodations-by-zsst-destinations	POST	DestinationsOnly (destinacije)	(schema iz OpenAPI)	Dostopnost nastanitev (obseg odvisen od definicije sheme).
Nepremična kulturna dediščina	letno	občina	/api/data/cultural-heritage-immovable-by-zsst-destinations	POST	DestinationsOnly	(schema iz OpenAPI)	Kulturna dediščina – nepremična (po destinaciji).
Nesnovna kulturna dediščina	letno	občina	/api/data/cultural-heritage-intangible-yearly-by-zsst-destinations	POST	YearsDestinations	(schema iz OpenAPI)	Kulturna dediščina – nesnovna (letno).
Občine (šifrant)	statično	Slovenija	(ni izpostavljeno v openapi.yml)	–	–	–	Če je potreben ločen šifrant občin, je potrebna dopolnitev API (ali uporaba obstoječih šifrantov NiST).

Opomba o metodah: OpenAPI specifikacija uporablja **POST** tudi za “list” operacije, kar pomeni, da se filtri (npr. leta, destinacije, občine) praviloma podajajo v request body.

Konkreten primer iz openapi.yml – “Arrivals and stays yearly by destination”

Spodaj je kratek, reprezentativen izsek, ki kaže **endpoint**, **request body** (leta + destinacije) in **response shemo**:

```
paths:
  /api/data/arrivals-and-stays-yearly-by-zsst-destinations:
    post:
      summary: Arrivals and stays yearly by destination
      requestBody:
        $ref: '#/components/requestBodies/YearsDestinations'
      responses:
        '200':
          description: OK
          content:
            application/json:
              schema:
                type: array
                items:
                  $ref: '#/components/schemas/ArrivalsStays'

components:
  requestBodies:
    YearsDestinations:
      content:
```

```
    application/json:
      schema:
        type: object
        properties:
          years:
            type: array
            items:
              type: integer
          destinations:
            type: array
            items:
              type: string
              example: 'Bela krajina'

schemas:
  ArrivalsStays:
    type: object
    properties:
      year:
        type: integer
      destination:
        type: string
      arrivals:
        type: integer
      stays:
        type: integer
```

Frekvenca osveževanja in prožilci (praktično za ta sistem)

V tem poglavju so opredeljene frekvence osveževanja podatkov in sprožitveni mehanizmi, prilagojeni potrebam konkretnega sistema Slovenia Green, ob upoštevanju razpoložljivosti virov ter zahtev glede zanesljivosti in neprekinjenega delovanja.

- **Statični šifranti** (ZSST destinacije, ponudniki): osveževanje **na zahtevo** ali **periodično** (npr. 1× dnevno).
- **Letni agregati** (prihodi/prenočitve, odpadki, zaposleni ipd.): osveževanje **periodično** (npr. 1× mesečno) ali **po objavi novih podatkov**.
- **Mesečni podatki** (CO_2): osveževanje **periodično** (npr. 1× tedensko ali 1× mesečno), odvisno od razpoložljivosti in objav vira.

V primeru nedosegljivosti NiST mora integracijski sloj rešitve omogočati **retry**, beleženje napak in nadaljevanje osnovnega delovanja (degradirano delovanje) brez blokade transakcijskih procesov Slovenia Green.

Kazalniki uspešnosti (KPI) – E3 Tehnična analiza integracij (projektna faza)

Za preverjanje ustreznosti tehnične analize integracij in njene izvedbe v projektni fazi se opredelijo naslednji ključni kazalniki uspešnosti (KPI):

- **Popolnost tehnične specifikacije integracij**:
  Delež integracij, za katere so pripravljeni vsi zahtevani tehnični elementi (podatkovni vir, struktura klica, tehnična izvedba, frekvenca osveževanja, prožilci, dodatni viri informacij).
  Zahtevana vrednost: 100 %.

- **Uspešnost integracijskih testov**:
  Delež vnaprej definiranih integracijskih testnih scenarijev, ki se izvedejo uspešno in brez napak. Zahtevana vrednost: 100 % uspešnih testov.
- **Pravilna obravnava napak in izjem v integracijah**:
  Delež testnih primerov, kjer sistem ob neuspelem integracijskem klicu pravilno zazna napako, jo ustrezno evidentira in ne povzroči nekonsistentnih podatkovnih ali poslovnih stanj. Zahtevana vrednost: 100 %.

4 Opis zahtev glede arhitekture in tehničnih značilnosti rešitve

Izhodišče arhitekturne zasnove je zahteva naročnika, da se bodoča rešitev, kjer je to smiselno, opira na enaka ali primerljiva tehnološka izhodišča kot NiST, z namenom kasnejše možnosti povezovanja oziroma kompatibilnosti sistemov, izvorne kode, podatkovnih struktur in infrastrukturnega okolja.

4.1 Arhitekturna zasnova rešitve

Bodoča rešitev mora biti zasnovana kot samostojen aplikacijski sistem z jasno ločenimi arhitekturnimi sloji:

- uporabniški vmesnik,
- aplikacijski in poslovni sloj,
- podatkovni sloj,
- integracijski sloj.

Takšna arhitekturna zasnova omogoča:

- jasno ločevanje odgovornosti med posameznimi deli sistema,
- boljšo obvladljivost kompleksnosti,
- lažje povezovanje z zunanjimi sistemi, vključno z NiST,
- neodvisno nadgrajevanje posameznih slojev sistema.

Arhitektura mora slediti načelom modularnosti, razširljivosti in ločenosti odgovornosti ter omogočati dolgoročno vzdržno delovanje sistema.

Arhitekturna zasnova rešitve mora predvideti možnost vključevanja AI-podprtih storitev kot ločenega funkcionalnega sloja ali zunanjih storitev, dostopnih prek standardiziranih vmesnikov (API).

AI funkcionalnosti ne smejo predstavljati tesne tehnične odvisnosti osnovnega sistema, temveč morajo biti zasnovane kot zamenljiv in razširljiv komponentni sklop, ki omogoča kasnejšo nadgradnjo, zamenjavo ali razširitev brez posegov v jedro aplikacije.

4.2 Tehnološka izhodišča in uporaba odprtokodnih orodij

Pri izbiri tehnologij in orodij za bodočo rešitev je smiselno izhajati iz tehnoloških izhodišč, ki so uporabljena v sistemu NiST, kot so opredeljena v Prilogi 1, zlasti z vidika razvojnih praks, arhitekturnih vzorcev in infrastrukturnega okolja.

Hkrati je pomembno poudariti, da se tehnična zasnova rešitve zavestno usmerja v uporabo uveljavljenih odprtokodnih (open-source) tehnologij in orodij, kjer je to smiselno in izvedljivo. Uporaba odprtokodnih rešitev omogoča:

- zmanjševanje odvisnosti od posameznih ponudnikov (vendor lock-in),
- večjo preglednost delovanja sistema,
- lažje dolgoročno vzdrževanje in nadgrajevanje,
- vključevanje dodatnih izvajalcev ali vzdrževalcev.

Uporaba enakih ali primerljivih tehnologij kot v NiST pomeni zagotavljanje tehnične kompatibilnosti in primerljivih razvojnih pristopov, ne pa združitve izvorne kode ali enotnega code-base-a.

Ne glede na zgornje usmeritve veljajo naslednje zavezujoče zahteve:

- rešitev mora biti zgrajena z uveljavljenimi, splošno razširjenimi tehnologijami, brez izključne vezave naročnika na izvajalca ali na specifičnega ponudnika oblačne infrastrukture;
- izvajalec ne sme uporabiti zaprtih/lastniških komponent ali licenc, ki bi naročnika trajno vezale na enega ponudnika, razen z izrecnim predhodnim pisnim soglasjem naročnika;
- licence vgrajenih komponent tretjih oseb morajo naročniku omogočati trajno uporabo, vzdrževanje in nadaljnji razvoj brez izvajalca;
- vsi podatki morajo biti kadar koli izvozljivi v odprtih, strojno berljivih formatih (npr. CSV, JSON, XML) brez dodatnih stroškov;

postopek namestitve mora biti dokumentiran tako, da je rešitev mogoče postaviti v okolju, neodvisnem od izvajalca.

4.3 Uporabniški vmesnik in aplikacijski sloj

Uporabniški vmesnik mora temeljiti na sodobnih spletnih tehnologijah, ki omogočajo:

- odzivno delovanje na različnih napravah,
- konsistentno uporabniško izkušnjo,
- jasno ločevanje predstavitvenega sloja od poslovne logike.

Aplikacijski sloj mora biti zasnovan modularno, z jasno strukturirano poslovno logiko, ki podpira:

- obvladovanje kompleksnih procesov presoje in ocenjevanja,
- dolgoročno razširljivost funkcionalnosti,
- integracijo z zunanjimi sistemi prek standardiziranih vmesnikov.

4.4 Podatkovni sloj in upravljanje podatkov

Podatkovni sloj mora temeljiti na preverjenih odprtokodnih podatkovnih tehnologijah ter omogočati:

- jasno strukturiran in razširljiv podatkovni model,
- dolgoročno hrambo podatkov, dokumentacije in revizijskih sledi,
- učinkovito obvladovanje rasti količine podatkov skozi čas.

Kompatibilnost podatkovnih struktur z NiST pomeni predvsem:

- semantično usklajenost ključnih entitet (npr. uporabniki, destinacije, ponudniki),
- možnost mapiranja in izmenjave podatkov med sistemi,
- ne pa nujno deljenja ali združevanja podatkovnih baz.

4.5 Integracijski sloj in tehnična povezljivost

Rešitev mora omogočati integracijo z NiST in drugimi zunanjimi sistemi prek jasno definiranega integracijskega sloja. Integracije morajo temeljiti na:

- standardiziranih protokolih in podatkovnih formatih,
- jasno dokumentiranih API vmesnikih,
- ločitvi integracij od osrednje poslovne logike sistema.

Takšen pristop omogoča stabilno delovanje sistema tudi ob spremembah ali nadgradnjah zunanjih virov.

4.6 API gateway in objava odprtih podatkov

V okviru bodoče rešitve Slovenia Green je predvidena možnost objave izbranih podatkovnih sklopov kot odprtih podatkov za nadaljnjo uporabo v zunanjih informacijskih sistemih, analitičnih orodjih ali javnih podatkovnih portalih. Z namenom zagotavljanja varnosti, stabilnosti, nadzorovane uporabe in dolgoročne vzdržnosti objave odprtih podatkov mora biti tehnična zasnova rešitve podprta z uporabo enotne vstopne točke za dostop do API vmesnikov (API gateway oziroma API management sloj).
API gateway predstavlja centralno komponento integracijskega sloja, prek katere se izpostavljajo vsi javni oziroma odprti API vmesniki bodoče rešitve. Notranje aplikacijske storitve in podatkovni viri niso neposredno dostopni zunanjim odjemalcem, temveč se vsi dostopi izvajajo izključno prek API gatewaya.
Minimalne funkcionalne zahteve API gatewaya vključujejo:

- centralizirano upravljanje in objavo API vmesnikov,
- podporo verzioniranju API-jev (npr. /api/v1, /api/v2),
- podporo za OpenAPI (Swagger) specifikacije in avtomatsko generirano dokumentacijo,
- nadzor nad uporabo API-jev (rate limiting, throttling),
- osnovne varnostne mehanizme (npr. CORS politika, IP omejitve, možnost uporabe API ključev),
- centralizirano beleženje dostopov in napak ter osnovne metrike uporabe API-jev.

Takšen pristop omogoča enotno upravljanje odprtih podatkov, preprečuje nenadzorovano izpostavljanje podatkovnih virov ter zagotavlja skladnost z arhitekturnimi in varnostnimi načeli naročnika.

Pristop k dodajanju novih odprtih API vmesnikov
Tehnična zasnova mora omogočati enostavno, ponovljivo in nadzorovano dodajanje novih API vmesnikov za odprte podatke v prihodnosti, brez posegov v temeljno arhitekturo sistema. Uvajanje novih API vmesnikov sledi standardiziranemu postopku:

1. **Definicija API kontrakta**
   Vsak nov API vmesnik se najprej definira na ravni OpenAPI specifikacije, ki vključuje opis endpointov, podatkovnih struktur, parametrov, odzivnih kod in morebitnih omejitev uporabe.
2. **Upoštevanje enotnih tehničnih standardov**
   API vmesniki morajo slediti enotnim pravilom glede poimenovanja, verzioniranja, paginacije, filtriranja, obravnave napak in formatiranja podatkov, kar zagotavlja konsistentnost in enostavno uporabo za zunanje odjemalce.
3. **Verzioniranje in združljivost**
   Spremembe, ki niso združljive za nazaj, se uvajajo izključno v novi verziji API-ja. Obstoječe verzije ostanejo dostopne skladno z dogovorjeno politiko življenjskega cikla API-jev.
4. **Objava prek API gatewaya**
   Novi API vmesniki se objavijo izključno prek API gatewaya, ki prevzame vlogo enotne vstopne točke, nadzora dostopov, beleženja in omejevanja uporabe.
5. **Testiranje in objava**
   Pred objavo v produkcijskem okolju se API vmesniki preverijo v testnem okolju, skladno z integracijskimi in varnostnimi zahtevami.
6. **Dokumentacija in katalog odprtih podatkov**
   Vsak objavljen API vmesnik mora biti ustrezno dokumentiran in vključen v katalog odprtih

API-jev oziroma odprtih podatkov, kar omogoča transparentnost in enostavno ponovno uporabo podatkov.

Opisan pristop zagotavlja, da je bodoča rešitev Slovenia Green tehnično pripravljena na postopno širitev nabora odprtih podatkov, hkrati pa ohranja nadzor nad kakovostjo, varnostjo in stabilnostjo objavljenih API vmesnikov skozi celoten življenjski cikel sistema.

4.7 Oblačna infrastruktura in gostovanje

Bodoča rešitev mora biti zasnovana tako, da omogoča gostovanje:

- bodisi znotraj obstoječe oblačne infrastrukture NiST,
- bodisi kot samostojna rešitev pri izbranem ponudniku oblačnih storitev.

Ne glede na izbrani scenarij mora arhitektura omogočati:

- jasno ločitev produkcijskega in neprodukcijskih okolij,
- varno in nadzorovano povezovanje z NiST,
- neodvisno upravljanje virov in okolij.

4.8 Kontejnerizacija, prenosljivost in obvladljivost

Za zagotavljanje prenosljivosti, obvladljivosti in neodvisnosti od konkretnega infrastrukturnega okolja mora biti rešitev zasnovana tako, da podpira kontejneriziran način razmestitve (npr. Docker).

Kontejnerizacija omogoča:

- enotno izvajalno okolje v različnih fazah življenjskega cikla sistema,
- lažje upravljanje odvisnosti in konfiguracij,
- prenosljivost rešitve med različnimi oblačnimi okolji ali ponudniki.

Ta pristop predstavlja ključen tehnični temelj za dolgoročno vzdržnost in morebitne prihodnje infrastrukturne spremembe.

4.9 Kompatibilnost brez prisilne združitve sistemov

Cilj arhitekturne zasnove ni prisilna združitev sistemov NiST in Slovenia Green, temveč zagotavljanje:

- tehnične in arhitekturne kompatibilnosti,
- standardiziranih integracijskih točk,
- možnosti prihodnjega povezovanja ali delne konsolidacije, če se za to pojavi potreba.

Takšen pristop omogoča profesionalno, obvladljivo in dolgoročno vzdržno sobivanje obeh sistemov, brez tveganj, ki bi jih prinesla prezgodnja ali neustrezna združitev.

Kazalniki uspešnosti (KPI)

Za preverjanje ustreznosti arhitekturne zasnove in tehničnih izhodišč bodoče rešitve se v projektni fazi opredelijo naslednji ključni kazalniki uspešnosti (KPI):

- **Skladnost izbora tehnologij z dogovorjenimi smernicami**:
  Delež uporabljenih tehnologij, orodij in platform, ki so skladne z zahtevanimi in dogovorjenimi tehničnimi in organizacijskimi smernicami projekta. Zahtevana vrednost: 100 %.
- **Priprava rešitve za kontejnerizacijo in prenosljivost**:
  Potrjena možnost zagona rešitve v kontejneriziranem okolju oziroma v več infrastrukturnih okoljih brez sprememb aplikacijske kode. Zahtevana vrednost: potrjena izvedljivost v enem od primernih okolij (npr. Kubernetes).

- **Uspešnost postavitve referenčnega testnega okolja**:
  Delež arhitekturnih komponent, ki so v testnem okolju uspešno postavljene in povezane skladno z arhitekturno zasnovo. Zahtevana vrednost: 100 %.

5 Osnutek ER diagrama

To poglavje podaja razširjen konceptualni opis podatkovnega modela bodoče rešitve Slovenia Green. Namen opisa ni zgolj identifikacija entitet, temveč razlaga njihove vloge v sistemu, življenjskega cikla ter medsebojnih povezav v podporo poslovnim procesom presoje, ocenjevanja, poročanja in spremljanja napredka.

Podatkovni model mora omogočati:

- dolgoročno uporabo sistema skozi več zaporednih let presoj,
- ohranjanje zgodovinskih podatkov in primerljivost skozi čas,
- neodvisno delovanje sistema od NiST, ob hkratni povezljivosti,
- podporo kompleksnim procesom, vključno z akcijskim načrtom in spremljanjem ukrepov.

5.1 Seznam ključnih entitet

V tem poglavju je predstavljen seznam ključnih entitet podatkovnega modela bodoče rešitve *Slovenia Green*. Entitete temeljijo na analizi poslovnih procesov, pravil in vsebinskih sklopov ter predstavljajo osnovne poslovne objekte, s katerimi sistem upravlja v okviru presoj, ocenjevanja in poročanja.

Seznam entitet zagotavlja strukturirano in pregledno izhodišče za razumevanje podatkovnega modela ter predstavlja osnovo za nadaljnjo opredelitev relacij in grafični prikaz ER diagrama.

1. Identitete, vloge in organizacijska struktura

Uporabnik
Entiteta Uporabnik predstavlja fizično osebo, ki vstopa v sistem in izvaja aktivnosti v skladu s svojo vlogo in pooblastili. Uporabniki so lahko notranji (administratorji, ocenjevalci) ali zunanji (koordinatorji destinacij, predstavniki ponudnikov).

Podatkovni model mora omogočati:

- obstoj uporabnikov, ki so aktivni v več organizacijah hkrati,
- spremembe vlog uporabnika skozi čas brez izgube zgodovinskih podatkov,
- sledljivost vseh aktivnosti uporabnika prek revizijske sledi.

Uporabnik je vedno povezan z vlogami in organizacijami, nikoli pa neposredno z vsebinskimi podatki brez konteksta presoje.

Vloga
Vloga predstavlja abstrakcijo pravic in odgovornosti v sistemu. Vloge niso zgolj tehnični mehanizem dostopa, temveč neposredno vplivajo na:

- vidnost podatkov,
- dovoljene akcije,
- zaporedje korakov v poslovnih procesih.

Podatkovni model mora omogočati:

- dodajanje novih vlog brez posegov v temeljno strukturo sistema,
- sočasno uporabo več vlog pri istem uporabniku,
- vezavo vlog na procesna stanja (npr. ocenjevalec vidi drugačne podatke pred in po oddaji).

Organizacija, Destinacija in Ponudnik
Organizacija predstavlja osnovni subjekt presoje. Model mora omogočati, da je organizacija:

- bodisi destinacija,
- bodisi ponudnik,
- v prihodnosti potencialno tudi drug tip subjekta (razširljivost).

Destinacija in ponudnik sta specializaciji organizacije, ki:

- delita skupne lastnosti (uporabniki, presoje, certifikati),
- imata hkrati specifične atribute in procese.

Pomembno je, da podatkovni model omogoča:

- več zaporednih presoj iste organizacije,
- hrambo podatkov tudi po prenehanju aktivnosti organizacije,
- povezovanje organizacij z zunanjimi identifikatorji (npr. NiST).

2. Standard, kriteriji, kazalniki in vsebinska konfiguracija

Standard in verzioniranje
Entiteta *StandardDefinition* predstavlja vsebinski okvir sistema. Ključno je, da model omogoča:

- več verzij standarda,
- sočasno uporabo starejših verzij za pretekle presoje,
- uporabo novejših verzij za aktualne cikle.

Standard ni statičen; skozi čas se razvija, zato mora biti podatkovni model odporen na spremembe strukture kriterijev in kazalnikov.

Kategorije, kriteriji in kazalniki
Kategorije, kriteriji in kazalniki tvorijo hierarhično strukturo standarda. Ta struktura:

- definira, kaj se presoja,
- določa obseg in kompleksnost posamezne presoje,
- vpliva na obseg dokazil in ocenjevanja.

Podatkovni model mora omogočati:

- ponovno uporabo istih kriterijev v različnih ciklih presoj,
- spreminjanje opisa ali pravil kazalnikov brez vpliva na zgodovinske vnose,
- pogojno prikazovanje kazalnikov glede na tip organizacije ali prejšnje odgovore.

Tipi kazalnikov in poslovna pravila
Tipi kazalnikov določajo način vnosa podatkov in obnašanje uporabniškega vmesnika. Poslovna pravila določajo:

- kdaj je kazalnik obvezen,
- kdaj zahteva dokazila,
- kako vpliva na nadaljnji potek presoje.

Pomembno je, da so pravila:

- podatkovno modelirana,
- ločena od same kode,
- sledljiva in razumljiva tudi z vidika vsebinskih odločevalcev.

3. Presoje, cikli in procesna stanja

Cikel presoje
Cikel presoje predstavlja časovni okvir, v katerem poteka več presoj. Cikel omogoča:

- enotno upravljanje rokov,
- primerljivost rezultatov med organizacijami,

- poročanje na ravni obdobja.

Podatkovni model mora omogočati več ciklov, ki se časovno ne prekrivajo ali se delno prekrivajo, odvisno od organizacijskih potreb.

Presoja
Presoja je osrednja transakcijska entiteta sistema. Združuje:

- organizacijo,
- cikel,
- standard,
- vnose, ocene, komentarje in končne rezultate.

Presoja ima jasno definiran življenjski cikel (priprava, oddaja, pregled, dopolnitve, zaključek), pri čemer mora model omogočati:

- prehode med stanji,
- blokade ali omejitve glede na status,
- ohranjanje vseh vmesnih stanj za revizijo.

4. Vnosi, dokazila in sodelovanje

Vnosi kazalnikov
Vnosi predstavljajo dejanske podatke, ki jih vnesejo uporabniki. Model mora omogočati:

- večkratno shranjevanje (osnutki),
- oddajo kot zaključeno stanje,
- naknadne dopolnitve po zahtevah ocenjevalcev.

Vnos je vedno kontekstualen: vezan na kazalnik, presojo in uporabnika.

Dokazila in mediateka
Dokazila niso zgolj priloge, temveč ključni del presoje. Model mora omogočati:

- več dokazil na posamezen kazalnik,
- uporabo istega dokazila pri več kazalnikih,
- dolgoročno hrambo in revizijsko sled.

Mediateka služi kot centralno skladišče dokumentov, ki preprečuje podvajanje in omogoča preglednost.

Komentarji, opombe in obvestila
Komunikacija med deležniki je integralni del procesa. Model mora omogočati:

- strukturirane niti komentarjev,
- ločevanje javnih komentarjev in internih opomb,
- avtomatsko obveščanje ob spremembah ali zahtevah za dopolnitve.

5. Ocenjevanje, rezultati in certifikati

Ocenjevanje
Ocenjevanje poteka na več ravneh:

- kazalnik,
- kriterij,
- celotna presoja.

Model mora omogočati:

- več ocenjevalcev,
- sledljivost ocen skozi čas,
- ločevanje med preliminarnimi in končnimi ocenami.

Rezultati in poročila
Rezultat presoje je agregacija ocen in predstavlja osnovo za:

- izdajo certifikata,
- poročanje,
- analizo napredka skozi čas.

Poročila so generirana iz podatkov in morajo biti verzionirana, saj se lahko ponovno generirajo po dopolnitvah.

Certifikati
Certifikat predstavlja uradni izid presoje. Model mora omogočati:

- časovno veljavnost,
- več zaporednih certifikatov za isto organizacijo,
- povezavo na konkretno presojo in standard.

6. Akcijski načrt in spremljanje ukrepov
Akcijski načrt ni ločen modul, temveč nadaljevanje presoje. Temelji na:

- rezultatih ocen,
- priporočilih ocenjevalcev,
- strateških ciljih organizacije.

Podatkovni model mora omogočati:

- planiranje ukrepov,
- spremljanje statusa izvedbe,
- povezavo ukrepov s konkretnimi kriteriji in kazalniki,
- poročanje o napredku skozi čas.

Akcijski načrt ima lastno časovno dimenzijo in presega posamezen cikel presoje.

7. Analitika in integrirani podatki (NiST)
Analitični podatki (npr. iz NiST) so ločeni od transakcijskih podatkov, vendar konceptualno povezani z destinacijami, časovnimi obdobji in kazalniki.
Model mora omogočati:

- jasno označevanje vira podatkov,
- ločevanje ročnih in uvoženih podatkov,
- uporabo podatkov za prikaze, primerjave in podporo odločanju.

8. Revizija, sledljivost in dolgoročna vzdržnost
Revizijska sled je prečna funkcionalnost celotnega modela. Vsaka pomembna sprememba mora biti sledljiva glede tega, **kdo** jo je izvedel, **kdaj** je bila izvedena, **kaj** je bilo spremenjeno in **v kakšnem kontekstu**.

To je ključno za zaupanje v sistem, transparentnost in dolgoročno uporabo.

5.2 Relacije med ključnimi entitetami – opis povezav in njihove vloge

V tem razdelku so opisane ključne relacije med entitetami v podatkovnem modelu. Namen opisa je pojasniti, kako so entitete med seboj povezane, kakšna je narava teh povezav ter kako te relacije podpirajo poslovne procese sistema *Slovenia Green*.
Relacije so opisane na konceptualni ravni, brez tehničnih implementacijskih podrobnosti, vendar z razlago njihovega pomena in uporabe.

1. Relacije identitet, vlog in organizacij

Uporabnik ↔ Vloga (M:N)

Uporabnik ima lahko dodeljenih več vlog, če v sistemu nastopa v različnih funkcijah. Enaka vloga je lahko dodeljena več uporabnikom.
Ta relacija omogoča:

- fleksibilno upravljanje pravic,
- prilagajanje dostopov glede na kontekst,
- dolgoročno razširljivost sistema brez sprememb osnovnega modela.

Uporabnik ↔ Organizacija (M:N, prek UserOrganizationMembership)
Uporabnik je lahko povezan z več organizacijami (npr. svetovalec, ki sodeluje z več destinacijami), organizacija pa ima več uporabnikov.
Povezava vsebuje dodatne informacije (npr. vlogo uporabnika v organizaciji, obdobje veljavnosti), kar omogoča:

- delegiranje nalog,
- časovno omejene povezave,
- sledljivost odgovornosti skozi čas.

Organizacija → Destinacija / Ponudnik (1:1 ali 1:N – specializacija)
Destinacija in ponudnik sta specializirani obliki organizacije. Ta relacija omogoča:

- skupno obravnavo osnovnih podatkov (uporabniki, certifikati),
- hkrati pa podporo specifičnim procesom (npr. destinacijska presoja vs. ponudnikova prijava).

Druge vrste organizacij so lahko tudi DMO, občine ali druge vključene v procese Slovenia Green.

2. Relacije standarda, kriterijev in kazalnikov

Standard → Kategorija (1:N)
En standard vsebuje več kategorij. Ta relacija omogoča:

- strukturiranje vsebine standarda,
- modularno prikazovanje vsebin v uporabniškem vmesniku,
- lažje vzdrževanje in nadgradnje standarda.

Kategorija → Kriterij (1:N)
Vsaka kategorija vsebuje več kriterijev. Relacija omogoča:

- logično združevanje kriterijev,
- poročanje in ocenjevanje po vsebinskih sklopih,
- enostavno prilagoditev strukture standarda v prihodnosti.

Kriterij → Kazalnik (1:N)
Kazalniki predstavljajo konkretne merljive elemente kriterija. Ta relacija omogoča:

- razčlenitev kriterijev na operativno raven,
- različno število kazalnikov glede na kompleksnost kriterija,
- ponovno uporabo kazalnikov v različnih presojah.

Kazalnik → Tip kazalnika (N:1)
Vsak kazalnik ima določen tip, ki definira način vnosa, validacije in obdelave podatkov. Relacija omogoča:

- standardizacijo obnašanja kazalnikov,
- poenostavljeno implementacijo uporabniškega vmesnika,
- lažje uvajanje novih tipov kazalnikov.

3. Relacije presoj in časovnih ciklov

Cikel presoje → Presoja (1:N)
En cikel presoje vključuje več posameznih presoj. Ta relacija omogoča:

- skupno upravljanje rokov,

- poročanje na ravni cikla,
- primerljivost rezultatov med organizacijami v istem obdobju.

Organizacija → Presoja (1:N)
Organizacija je lahko predmet več presoj skozi čas. Ta relacija je ključna za:

- dolgoročno spremljanje napredka,
- zgodovinski pregled rezultatov,
- večkratno certificiranje iste organizacije.

Presoja → Status (N:1)
Vsaka presoja ima v danem trenutku en status. Relacija omogoča:

- nadzor nad potekom procesa,
- omejevanje dovoljenih akcij glede na stanje,
- transparentnost za vse vpletene uporabnike.

4. Relacije vnosa podatkov, dokazil in komunikacije

Presoja → Vnos kazalnika (1:N)
Presoja vsebuje več vnosov kazalnikov. Ta relacija predstavlja dejansko izpolnjevanje standarda in omogoča:

- postopno izpolnjevanje,
- shranjevanje osnutkov,
- naknadne dopolnitve.

Kazalnik → Vnos kazalnika (1:N)
Kazalnik se v vsaki presoji pojavi kot ločen vnos. Relacija omogoča:

- ponovno uporabo definicije kazalnika,
- ločevanje vsebine standarda od transakcijskih podatkov.

Vnos kazalnika ↔ Dokazilo (1:N)
En vnos kazalnika ima lahko več dokazil. Relacija omogoča:

- podajanje več dokazov za isti kazalnik,
- ločevanje dokazil po vsebini in kontekstu.

Mediateka ↔ Dokazilo (1:N ali M:N)
Dokazila so shranjena v mediateki, ki omogoča:

- centralno upravljanje dokumentov,
- ponovno uporabo istega dokumenta,
- boljši nadzor nad dostopi in hrambo.

Presoja / Kriterij / Vnos ↔ Komentar (1:N)
Komentarji so lahko vezani na različne ravni presoje. Relacija omogoča:

- ciljno usmerjeno komunikacijo,
- jasen kontekst komentarjev,
- strukturirano obravnavo dopolnitev.

5. Relacije ocenjevanja, rezultatov in certifikatov

Presoja ↔ Ocenjevalec (M:N, prek AssessorAssignment)
Presoja lahko vključuje več ocenjevalcev, ocenjevalec pa lahko sodeluje pri več presojah. Relacija omogoča:

- razdelitev dela,
- večstopenjsko ocenjevanje,
- jasno določitev odgovornosti.

Vnos kazalnika → Ocena kazalnika (1:1 ali 1:N)
Vsak vnos ima eno ali več ocen (npr. preliminarna in končna). Relacija omogoča:

- sledljivost sprememb ocen,
- transparentnost ocenjevalnega procesa.

Presoja → Rezultat presoje (1:1)
Rezultat presoje predstavlja agregacijo vseh ocen. Relacija omogoča:

- enoznačno povezavo med procesom in izidom,
- osnovo za poročila in certifikate.

Rezultat presoje → Certifikat (1:0..1)
Na podlagi rezultata presoje se lahko izda certifikat. Relacija omogoča:

- sledljivost certifikata do konkretne presoje,
- upravljanje veljavnosti certifikatov.

6. Relacije akcijskega načrta

Organizacija → Akcijski načrt (1:N)
Organizacija ima lahko več akcijskih načrtov skozi čas. Relacija omogoča:

- dolgoročno planiranje izboljšav,
- povezovanje ukrepov z različnimi cikli presoj.

Akcijski načrt → Ukrep (1:N)
Akcijski načrt vsebuje več ukrepov. Relacija omogoča:

- razčlenitev ciljev na izvedljive aktivnosti,
- spremljanje napredka po posameznih ukrepih.

Ukrep ↔ Kriterij / Kazalnik (M:N)
Ukrep je lahko povezan z več kriteriji ali kazalniki, in obratno. Relacija omogoča:

- sledljivost ukrepov do ugotovitev presoje,
- merjenje učinkov ukrepov.

7. Relacije analitike in integracij (NiST)

Destinacija / Občina ↔ Uvožen podatek (1:N)
Analitični podatki iz NiST so vezani na prostorsko in časovno dimenzijo. Relacija omogoča:

- primerjave skozi čas,
- kontekstualno uporabo podatkov v presoji.

Uvožen podatek → Vir podatkov (N:1)
Vsak uvožen podatek ima jasno definiran vir. Relacija omogoča:

- transparentnost izvora podatkov,
- ločevanje ročnih in avtomatsko pridobljenih podatkov.

8. Relacije revizije in sledljivosti

Entiteta ↔ Revizijska sled (1:N)
Vsaka ključna entiteta ima več revizijskih zapisov. Relacija omogoča:

- popolno sledljivost sprememb,
- nadzor nad delovanjem sistema,
- podporo revizijam in nadzoru.

5.3 ER diagram

V tem poglavju je prikazan ER diagram bodoče rešitve Slovenia Green, ki grafično predstavlja podatkovni model sistema, vključno z entitetami, njihovimi atributi in relacijami med njimi. Diagram omogoča pregleden vpogled v podatkovno arhitekturo rešitve, podpira razumevanje poslovne logike sistema ter predstavlja referenčno podlago za nadaljnje faze implementacije in migracije podatkov.

Spodaj je naveden KPI, ki ga mora izvajalec doseči pred prevzemom.

Kazalniki uspešnosti (KPI) – E5 ER diagram (projektna faza)

Za preverjanje kakovosti, popolnosti in izvedljivosti podatkovnega modela bodoče rešitve se v okviru projekta opredelijo naslednji ključni kazalnik uspešnosti (KPI):

- **Skladnost ER diagrama z najboljšimi praksami in potrditev le-tega**:

ER diagram, izdelan z uporabo izbranega orodja ter v skladu z uveljavljenimi standardi modeliranja in najboljšimi praksami na področju podatkovnega modeliranja. Formalna potrditev ER diagrama s strani naročnika. Zahtevana vrednost: ER diagram potrjen.



Slika 10: ER diagram izdelan v dbdiagram.io

6 Načrt migracije podatkov

Namen izhodiščnega načrta migracije podatkov je opredeliti osnovna načela, pristope in omejitve prenosa podatkov iz obstoječih in predhodnih informacijskih rešitev v novo rešitev Slovenia Green, skladno z zasnovanim ER diagramomter poslovno logiko, opisano v dokumentu »Funkcionalne specifikacije«.
Načrt migracije je pripravljen kot **izhodiščni tehnični okvir**, namenjen podpori naslednje implementacijske faze projekta. Predstavlja ključne usmeritve, vire podatkov, vrste prenosov ter predvidene pristope k transformaciji in ponovljivosti migracijskih postopkov.
Pri pripravi načrta so bila upoštevana naslednja izhodišča:

- podatkovni model bodoče rešitve je namensko zasnovan z omejenim naborom **ključnih entitet**, ki podpirajo osrednje poslovne procese,
- zgodovinski podatki se v nov sistem prenašajo v obsegu, ki omogoča analitiko, primerjave in sledljivost, ne pa popolne rekonstrukcije vseh preteklih presoj,
- migracija mora biti zasnovana tako, da omogoča **ponovljivost, testiranje in kontroliran končni prenos** ob prehodu v produkcijsko uporabo.

6.1 Pregled virov podatkov

Za potrebe migracije podatkov sta identificirana dva glavna vira podatkov, ki se po vsebini, strukturi in zahtevnosti bistveno razlikujeta:

- podatki o presojah, izvedenih **pred letom 2021**, ki obstajajo pretežno v dokumentni obliki,
- podatki iz **trenutno delujoče aplikacije Zelene sheme Slovenia Green** (presoje po letu 2021).

Za vsak vir je predviden ločen migracijski pristop, prilagojen naravi podatkov in ciljni uporabi v novem sistemu.

6.2 Migracija podatkov iz obdobja pred letom 2021

Podatki o presojah, izvedenih pred letom 2021, so na voljo predvsem v obliki zaključenih poročil o presoji (audit report), pripravljenih kot samostojni dokumenti v elektronski obliki (najpogosteje PDF). Tipičen primer takšnega dokumenta je poročilo o presoji destinacije Bled iz leta 2018, ki vsebuje jasno razmejene vsebinske sklope, kot so podatki o presoji, rezultati presoje, agregirane ocene po kategorijah ter priporočila za izboljšave.
Ti dokumenti niso bili pripravljeni z namenom neposredne strojne obdelave in niso strukturno usklajeni z današnjim podatkovnim modelom kriterijev in kazalnikov. Predstavljajo **interpretiran rezultat presoje**, namenjen poročanju in odločanju, ne pa podrobni rekonstrukciji posameznih vnosov.
Na podlagi analize tipičnih poročil iz tega obdobja je ugotovljeno, da je zanesljivo in smiselno prenašati naslednje podatke:

- identifikacijske podatke presoje (destinacija, leto, tip presoje),
- agregirane rezultate presoje na nivoju kategorij,
- skupne ocene in certifikacijski status (npr. gold, silver, bronze),
- kazalnike skladnosti s standardom (npr. odstotek izpolnjenih meril),
- strukturirana besedilna priporočila in roke za izboljšave.

Podatkov na nivoju posameznih kriterijev in kazalnikov iz teh dokumentov ni mogoče enoznačno in dosledno preslikati v sodobni podatkovni model, zato se ti **ne prenašajo kot strukturirani operativni podatki**.

6.2.1 Tehnični pristop k obdelavi in ekstrakciji podatkov

Migracija podatkov iz obdobja pred letom 2021 je zasnovana kot **enkratni pripravljalni postopek z možnostjo ponovne uporabe rezultatov**, ki temelji na naslednjih tehničnih fazah:

a) Zajem izvornih dokumentov (ingest)

- izvorni PDF dokumenti se zberejo v centralnem repozitoriju,
- vsak dokument dobi enolični identifikator,
- vzpostavi se povezava med dokumentom, destinacijo in letom presoje.

b) Ekstrakcija vsebine s pomočjo AI

Za vsak dokument se uporabi kombinacija:

- OCR obdelave (če je dokument skeniran),
- semantične analize besedila,
- prepoznavanja tabel, naslovov in vsebinskih razdelkov.

AI procesi identificirajo:

- strukturo dokumenta (npr. »Rezultati presoje«, »Priporočila«),
- numerične vrednosti ocen,
- besedilne vsebine priporočil in komentarjev,
- semantične povezave med kategorijami in ocenami.

Rezultat te faze je **strukturiran izhodni zapis** (npr. JSON ali tabelarična oblika), ki predstavlja normalizirano vsebino dokumenta.

c) Validacija in poenotenje podatkov

Izluščeni podatki se:

- ročno ali polavtomatsko validirajo,
- poenotijo glede poimenovanj kategorij,
- prilagodijo ciljnemu naboru vrednosti (npr. lestvice ocen).

Ker se zgodovinski dokumenti ne spreminjajo, se ta korak izvede **enkrat**, rezultat pa se shrani kot referenčni migracijski vir.

6.2.2 Preslikava v ciljni podatkovni model (ER)

Izluščeni in validirani podatki se preslikajo izključno v **ključne entitete**, opredeljene v ER diagramu, in sicer:

- entiteta *Presoja* – kot zgodovinska presoja z oznako vira »pred 2021«,
- entiteta *Destinacija* – povezava z obstoječim šifrantom destinacij,
- entiteta *Kategorija* – agregirani rezultati in ocene,
- entitete za agregirane rezultate in metapodatke presoje,
- entiteta *Dokument* – izvorni PDF kot arhivska priloga.

Podatki se ne vnašajo v strukture za kazalnike, vnose ali dokazila, saj to presega informacijsko vrednost in zanesljivost izvornih dokumentov.

6.2.3 Načelo ponovljivosti in tehnična izvedba vnosa

Na podlagi strukturiranega izhodnega zapisa se pripravi **ponovljiv migracijski postopek**, ki omogoča:

- večkratni uvoz istih zgodovinskih podatkov (testno, produkcijsko),
- sledljivost med izvorno datoteko in ciljnimi zapisi,
- enostavno razširitev na dodatne zgodovinske dokumente.

Tehnična izvedba vnosa (npr. skripta, ETL proces, uvozni servis) se dokončno opredeli v implementacijski fazi, pri čemer so izhodni podatki že prilagojeni ciljnemu ER modelu.

6.2.4 Namen in omejitve uporabe zgodovinskih podatkov

Zgodovinski podatki iz obdobja pred letom 2021 so v novem sistemu namenjeni:

- analitiki in trendnim pregledom,
- primerjavam med leti in presojami,
- kontekstualnemu vpogledu v razvoj destinacije.

Ti podatki niso namenjeni neposredni uporabi v aktivnih procesih presoje ali avtomatskemu prenosu v nove presoje, temveč služijo kot **referenčna in analitična plast sistema**.

6.2.5 Primer ekstrakcije in vnosa zgodovinske presoje (ilustrativni primer)

V nadaljevanju je prikazan poenostavljen primer, ki temelji na poročilu o presoji destinacije Bled iz leta 2018, in ilustrira:

- kakšen je lahko izhod ekstrakcije s pomočjo AI,
- kako se tak izhod preslika v ključne entitete ciljne baze,
- kako se zagotovi ponovljiv in sledljiv vnos podatkov.

a) Primer strukturiranega izhoda AI-ekstrakcije

Rezultat obdelave izvornega PDF dokumenta je strukturiran zapis (npr. JSON), ki združuje ključne vsebinske elemente poročila:


```
{
  "source_document_id": "SG_AUDIT_BLED_2018",
  "source_type": "audit_report_pre_2021",
  "destination": {
    "code": "BLED",
    "name": "Bled"
  },
  "assessment": {
    "year": 2018,
    "assessment_type": "destination",
    "overall_score": 8.3,
    "certification_level": "gold",
    "gstc_compliance_percent": 85.3
  },
  "category_results": [
    {
      "category_code": "DEST_MANAGEMENT",
      "category_name": "Destinacijski management",
      "score": 8.8
    },
    {
      "category_code": "NATURE_LANDSCAPE",
      "category_name": "Narava in pokrajina",
      "score": 8.7
    },
    {
      "category_code": "ENV_CLIMATE",
      "category_name": "Okolje in podnebje",
      "score": 8.2
    },
    {
      "category_code": "CULTURE_TRADITION",
      "category_name": "Kultura in tradicija",
      "score": 8.6
    },
    {
      "category_code": "SOCIO_ECONOMIC",
      "category_name": "Družbeno-ekonomska situacija",
```

```
      "score": 7.9
    }
  ],
  "recommendations": [
    {
      "category_code": "DEST_MANAGEMENT",
      "text": "Destinacija bo v roku 6 mesecev pripravila akcijski načrt
trajnostnega razvoja turizma.",
      "deadline": "2018-10-20"
    },
    {
      "category_code": "ENV_CLIMATE",
      "text": "Vzpostaviti je treba sistem merjenja in zmanjševanja
ogljičnega odtisa obiskovalcev.",
      "deadline": null
    }
  ]
}
```

Ta zapis predstavlja **kanonični migracijski format**, ki je:

- neodvisen od implementacije,
- ponovljivo uporaben,
- neposredno preslikljiv v ER model.

b) Preslikava v ciljne entitete (konceptualni pogled)

Na podlagi zgornjega zapisa se podatki preslikajo v naslednje ključne entitete:

- *Presoja*
  – leto = 2018
  – tip = zgodovinska presoja
  – vir = »pred 2021 – audit report«
- *Destinacija*
  – obstoječa entiteta (npr. BLED)
- *Rezultat kategorije*
  – ena vrstica na kategorijo
  – ocena, leto presoje
- *Priporočilo*
  – besedilna vsebina
  – vezava na presojo in kategorijo
- *Dokument*
  – izvorni PDF (arhivska priloga)

c) Primer vzorčnega SQL vnosa (ilustrativni primer)

Spodnji primer prikazuje poenostavljen SQL za vnos zgodovinske presoje in rezultatov kategorij. Imena tabel in stolpcev so zgolj ilustrativna in bodo v implementaciji usklajena z dejanskim ER modelom.

-- vnos zgodovinske presoje

```
INSERT INTO assessment (
    assessment_id,
    destination_id,
    year,
    assessment_type,
    source_type,
    overall_score,
    certification_level,
    gstc_compliance_percent
)
VALUES (
    'ASSESS_BLED_2018',
    'DEST_BLED',
    2018,
```

```
        'DESTINATION',
        'PRE_2021_AUDIT_REPORT',
        8.3,
        'GOLD',
        85.3
    );
```

-- vnos rezultatov po kategorijah

```
INSERT INTO assessment_category_result (
  assessment_id,
  category_code,
  score
)
VALUES
('ASSESS_BLED_2018', 'DEST_MANAGEMENT', 8.8),
('ASSESS_BLED_2018', 'NATURE_LANDSCAPE', 8.7),
('ASSESS_BLED_2018', 'ENV_CLIMATE', 8.2),
('ASSESS_BLED_2018', 'CULTURE_TRADITION', 8.6),
('ASSESS_BLED_2018', 'SOCIO_ECONOMIC', 7.9);

-- vnos priporočila
INSERT INTO recommendation (
  assessment_id,
  category_code,
  recommendation_text,
  deadline
)
VALUES (
  'ASSESS_BLED_2018',
  'DEST_MANAGEMENT',
  'Priprava akcijskega načrta trajnostnega razvoja turizma.',
  '2018-10-20'
);
```

d) Vloga primera v okviru tehnične analize

Namen zgornjega primera je:

- pokazati, **kaj je realno izluščiti iz zgodovinskih poročil**,
- demonstrirati, kako se podatki **zavestno zreducirajo na ključne entitete**,
- ponuditi implementacijskemu projektu jasen koncept za:
    - AI-ekstrakcijo,
    - staging podatke,
    - ponovljiv migracijski vnos.

Primer ne predstavlja dokončne implementacije, temveč **referenčni vzorec**, ki se v fazi razvoja prilagodi izbrani tehnologiji, končni strukturi baze in integracijskim mehanizmom.

6.3 Migracija podatkov iz obstoječe aplikacije (presoje po letu 2021)

Za presoje, izvedene po letu 2021, se uporablja obstoječa aplikacija Zelene sheme Slovenia Green, ki podpira strukturiran vnos podatkov v skladu s standardom Green Destinations (npr. V2). Sistem omogoča:

- strukturiran zajem podatkov po temah, kriterijih in kazalnikih,
- shranjevanje odgovorov destinacije in pripomb presojevalcev,
- povezavo na dokazila in priloge,
- izvoz podatkov v strojno berljivi obliki (npr. JSON),
- generiranje poročil (audit report, remarks report, action plan).

Za vsako presojo je tako na voljo **celovit, strukturiran nabor podatkov**, ki je primeren za avtomatiziran prenos v nov sistem.

6.3.1 Strategija migracije podatkov

Migracija podatkov iz obstoječe aplikacije je zasnovana kot **ponovljiv in nadzorovan proces**, ki temelji na naslednjih načelih:

- **enoten migracijski mehanizem** za vse presoje po letu 2021,
- ločitev med testnimi in produkcijskimi prenosi,
- možnost večkratnega zagona migracije v času razvoja in testiranja,
- izvedba **končnega produkcijskega prenosa** ob prehodu na novo aplikacijo.

Migracija se izvaja na podlagi uradnih izvozov iz obstoječega sistema, brez neposrednega poseganja v njegovo podatkovno bazo.

6.3.2 Tehnični pristop k prenosu podatkov

Migracijski postopek je razdeljen na naslednje tehnične faze:

a) Izvoz podatkov iz obstoječe aplikacije

Za vsako presojo se pripravi izvoz, ki vključuje:

- osnovne podatke o destinaciji in presoji,
- strukturo standarda (teme, podteme, kriteriji),
- odgovore destinacije (organization answers),
- presojevalne ocene in pripombe (audit answers, remarks),
- akcijske načrte (če obstajajo),
- povezave na dokazila in dokumente.

Primer takšnega izvoza je strukturiran JSON zapis za presojo destinacije Lendava po standardu Green.

b) Staging in transformacija podatkov

Izvoženi podatki se najprej naložijo v **vmesno (staging) plast**, kjer se izvede:

- validacija strukture in obveznih podatkov,
- poenotenje šifrantov (npr. kategorije, kriteriji),
- preslikava identifikatorjev izvornega sistema v interne identifikatorje novega sistema,
- označevanje vira podatkov (»obstoječi sistem – po 2021«).

Ta faza omogoča jasno ločitev med izvorno strukturo in ciljnim ER modelom.

c) Vnos v ciljni sistem (load)

Na podlagi podatkov iz staging plasti se izvede vnos v ciljni sistem preko:

- migracijskih skript,
- ETL postopkov,
- ali uvoznih servisov nove aplikacije.

Način izvedbe se dokončno opredeli v implementacijski fazi.

6.3.3 Preslikava v ciljni podatkovni model (ER)

Podatki iz obstoječe aplikacije se preslikajo v ciljni ER model v največjem možnem obsegu, in sicer:

- *Presoja*
  – leto, tip presoje, status, standard (npr. GD V2),
- *Destinacija*
  – povezava na obstoječo entiteto destinacije,

- *Kategorija / Tema*
  – agregirani in podrobni rezultati,
- *Kriterij*
  – posamezni kriteriji z odgovori in ocenami,
- *Kazalnik* (če obstaja v ER modelu)
  – strukturirani odgovori in vrednosti,
- *Pripomba / Remark*
  – pripombe presojevalca,
- *Akcijski načrt / Ukrep*
  – načrtovane aktivnosti, roki, odgovorne osebe ,
- *Dokument*
  – dokazila, poročila in priloge (npr. audit report, remarks report, action plan) .

Za razliko od presoj pred letom 2021 so ti podatki v novem sistemu na voljo **do nivoja posameznega kriterija**, kar omogoča poglobljeno analitiko in časovne primerjave.

6.3.4 Primer strukturiranega izvoza (ilustrativni primer)

Spodaj je poenostavljen primer izseka strukturiranega izvoza presoje iz obstoječe aplikacije:


```
{
  "destination": {
    "code": "LENDAVA",
    "name": "Lendava"
  },
  "assessment": {
    "year": 2025,
    "standard": "Green Destinations Standard V2",
    "status": "completed"
  },
  "category": {
    "code": "DEST_MANAGEMENT",
    "name": "Destinacijski management"
  },
  "criterion": {
    "code": "1.1",
    "name": "Zeleni koordinator",
    "organization_answer": "yes",
    "audit_answer": "Partially compliant"
  },
  "indicator": [
    {
      "description": "Ime koordinatorja",
      "value": "Sara Alt"
    },
    {
      "description": "E-pošta",
      "value": "sara@ztr.si"
    }
  ],
  "remarks": [
    {
      "author": "Auditor",
      "text": "Delovno mesto destinacijskega managerja ni zasedeno."
    }
  ]
```

```
}
```

6.3.5 Primer vzorčnega SQL vnosa (ilustrativni primer)

-- vnos presoje

```
INSERT INTO assessment (
    assessment_id,
    destination_id,
    year,
    standard,
    source_type
)
VALUES (
    'ASSESS_LENDAVA_2025',
    'DEST_LENDAVA',
    2025,
    'GD_V2',
    'EXISTING_SYSTEM_POST_2021'
);
```

-- vnos kriterija

```
INSERT INTO assessment_criterion (
    assessment_id,
    criterion_code,
    organization_answer,
    audit_answer
)
VALUES (
    'ASSESS_LENDAVA_2025',
    '1.1',
    'YES',
    'PARTIALLY_COMPLIANT'
);

-- vnos pripombe
INSERT INTO assessment_remark (
    assessment_id,
    criterion_code,
    remark_text
)
VALUES (
    'ASSESS_LENDAVA_2025',
    '1.1',
    'Delovno mesto destinacijskega managerja ni zasedeno.'
);
```

6.3.6 Ponovljivost in končni prenos

Migracijski program za prenos podatkov iz obstoječe aplikacije je zasnovan tako, da omogoča:

- večkratne testne migracije,
- primerjavo rezultatov med zagoni,
- kontroliran končni prenos vseh relevantnih presoj ob prehodu v novo aplikacijo.

S tem se zagotovi zanesljiv, sledljiv in tehnično obvladljiv prehod podatkov brez izgube vsebinske vrednosti.

Primerjalni pregled migracije podatkov po obdobjih

Vidik	Presoje pred letom 2021	Presoje po letu 2021 (obstoječa aplikacija)

Izvor podatkov	Zaključna poročila o presoji (PDF, priloge)	Strukturirani podatki v aplikaciji Zelene sheme
Struktura podatkov	Nestrukturirani ali polstrukturirani dokumenti	Popolnoma strukturirani podatki (teme, kriteriji, kazalniki)
Tehnični dostop	Datoteke v repozitoriju	Uradni izvoz podatkov (npr. JSON)
Metoda obdelave	AI-ekstrakcija + validacija	Avtomatiziran prenos
Stopnja avtomatizacije	Delna (ekstrakcija + ročna validacija)	Visoka (ETL / migracijski program)
Pogostost obdelave	Enkratna priprava podatkov	Večkratni, ponovljivi prenosi
Namen migracije	Zgodovinski in analitični pregled	Operativna in analitična uporaba
Nivo podrobnosti v ciljnem sistemu	Kategorija / agregirani rezultati	Kriterij (in kjer je mogoče tudi kazalnik)
Prenos ocen	Agregirane ocene	Posamezne ocene po kriterijih
Prenos priporočil	Da (besedilno, agregirano)	Da (vezano na kriterije)
Prenos akcijskih načrtov	Ne (ni strukturirano na voljo)	Da (strukturirani akcijski načrti)
Dokazila in priloge	Shranjena kot arhivski dokumenti	Povezana z ustreznimi kriteriji
Preslikava v ER model	Le ključne entitete	Skoraj celoten ER model
Možnost ponovne presoje	Ne	Da (uporaba dokumentov)
Analitika in primerjave	Do nivoja kategorij	Do nivoja kriterijev
Tveganje napačne interpretacije	Zmerno (odvisno od kakovosti dokumentov)	Nizko (strukturirani podatki)
Vloga v novem sistemu	Referenčna zgodovinska plast	Aktivna in referenčna podatkovna plast

Migracijski pristop je zavestno diferenciran glede na obdobje nastanka podatkov.
Za presoje pred letom 2021 in po letu 2021 je poudarek na **ohranjanju zgodovinske vrednosti** in podpori analitiki, hkrati pa nudijo dokumentacijo, ki jo je možno uporabiti v nadaljnjem delu.
Ta ločitev omogoča tehnično obvladljivo migracijo, hkrati pa zagotavlja, da sistem ohrani kontinuiteto podatkov brez nepotrebne kompleksnosti.

Spodaj so navedeni KPI-ji, ki jih mora izvajalec doseči pred prevzemom.

KPI: Pravilno preslikani migrirani podatki za uvedbo Faze 1
Delež migriranih zapisov, skladnih s ciljnim podatkovnim modelom v obsegu, ki je zahtevan za posamezno vrsto podatkov.
Zahtevana vrednost: ≥ 99 %.

KPI: Ponovljivost migracijskega postopka v testnem okolju
Število zaporednih migracij brez ročnih popravkov.
Zahtevana vrednost: najmanj 2.