

Specifikacija softverskih zahtjeva

za

<Naziv projekta>

Kolegij: Programsko inženjerstvo

Akadska godina: <akadska godina>

Autor: <ime i prezime>

Datum: <datum izrade>

1. UVOD

1.1.Svrha

<Opišite svrhu SRS dokumenta (ne softverskog rješenja). Navedite kome je namijenjen dokument, tj. tko treba dokument čitati i razumjeti.>

1.2.Opseg

<Ukratko objasnite problemsku domenu kojom se rješenje bavi. U kojem kontekstu će se softversko rješenje upotrebljavati? Dajte ime/naziv softverskom rješenju kojim će ga se u ostatku dokumenta moći naslovljavati, te po potrebi verziju. Objasnite radi li se o potpuno novom rješenju ili nadogradnji. Navedite što će softver raditi i što neće raditi. Koje dobrobiti i unaprjeđenja očekujemo da će softversko rješenje donijeti.>

1.3.Definicije, akronimi i skraćenice

<Definirajte osnovne pojmove, akronime i skraćenice koji se koriste u specifikaciji, a koje su potrebne da bi se moglo ispravno razumjeti SRS dokument.>

1.4.Reference

<Referencirajte sve dokumente, web stranice, standarde, druge specifikacije koji se spominju u SRS-u.>

1.5.Struktura dokumenta

<Ukratko opišite kako je ostatak dokumenta organiziran i što sadrži.>

2. OPĆENITI OPIS

2.1.Perspektiva proizvoda

<Stavite softversko rješenje u kontekst i odnos s drugim povezanim sustavima. Navedite je li softversko rješenje neovisno i u potpunosti samostalno, ili predstavlja dio većeg sustava (što je čest slučaj), ili je zamjena za neki postojeći sustav. Ako je dio većeg sustava potrebno je staviti u odnos zahtjeve cjelokupnog sustava sa zahtjevima našeg softverskog rješenja, te jasno definirati sučelja između njih.>

Osim odnosa s eventualnim nadređenim sustavom, navesti i vanjske sustavima i softver koji vaše softversko rješenje koristi. To mogu biti npr.: DMBS, operacijski sustav, web servisi i API-ji.

U slučaju da softversko rješenje izravno koristi hardver i komunikacijske tehnologije opišite i značajke sučelja s njima (vrsta hardvera, port, komunikacijski protokoli, bluetooth, NFC, IC, TCP, format razmjene podataka i sl.).>

2.2.Funkcije proizvoda

<Navedite glavne funkcije koje softversko rješenje treba sadržavati, bez ulaženja u detalje svake od tih funkcija (detaljni zahtjevi će biti sadržani u poglavlju 3). Funkcije trebaju biti organizirane i opisane na način da već pri prvom čitanju budu razumljive svim čitateljima dokumenta, uključujući i naručitelja/korisnika. Za formulaciju funkcija softverskog rješenja na ovoj razini može poslužiti specifikacija više razine, npr. specifikacija korisničkih zahtjeva (ako postoji).>

2.3.Karakteristike korisnika

<Identificirajte grupe/uloge korisnika za koje očekujemo da će koristiti softversko rješenje (npr. nastavnik, profesor, administrator, blagajnica, bankovni službenik, računovođa, ...). Navedite opće karakteristike identificiranih korisnika, navedite po čemu se oni razlikuju. To može uključivati razinu obrazovanja, iskustva, računalne i tehničke pismenosti, zatim predviđenu učestalost korištenja softverskog rješenja, razinu dodijeljenih dozvola, skupa funkcija softvera kojeg će koristiti.>

2.4.Ograničenja

<Navesti aspekte problemske domene i samog rješenja koji mogu djelovati ograničavajuće na razvoj softverskog rješenja. To može uključivati: zakonske i korporativne propise i regulative (npr. GDPR odredbe, fiskalizacija, sigurnosne politike, i sl.); hardverska ograničenja (npr. u razvoju za male uređaje – baterija, memorija, procesor, povezivost, i sl.); potrebu za prilagodbom drugim sustavima (npr. interakcija s postojećim i potencijalno zastarjelim sustavima); sigurnosnu kritičnost aplikacije (npr. u slučaju da aplikacija upravlja opasnim strojevima, medicinskim uređajima, ili radi sa osjetljivim i povjerljivim podacima, i sl.); zahtjeve za pouzdanošću (npr. može biti zahtijevano korištenje određenog pristupa, alata praksi i standarda u programiranju, modeliranju i testiranju softverskog rješenja i sl.) >

2.5.Pretpostavke i ovisnosti

<Navedite pretpostavke i otvorena pitanja (za razliku od poznatih činjenica) čiji naknadni ishod može utjecati na zahtjeve navedene u ovom dokumentu. To su vrlo često okolnosti koje nisu pod našom

odgovornosti (npr. ako postoji vjerojatna mogućnost ili najava promjene zakonske regulative koja će rezultirati modificiranjem definiranih zahtjeva; ili izlazak nove verzije API-ja o kojem naš softver ovisi).>

2.6.Ostalo

<Navedite ostale aspekte problemske domene i budućeg softverskog rješenja koje smatrate bitnima a koji nisu predviđeni u ostalim dijelovima dokumenta.>

3. FUNKCIONALNI ZAHTJEVI

<Definirajte funkcionalne zahtjeve za softversko rješenja i to na način da pružite dovoljno informacija dizajnerima i programerima da mogu započeti sa osmišljavanjem i implementacijom rješenja, a testerima da osmisle testne slučajeve. Zahtjevi navedeni ovdje se temelje na funkcijama proizvoda opisanim u poglavlju 2.2., ali su opisani s višom razinom detalja. Fokus je na funkciji i ograničenjima sustava. Svakom zahtjevu treba dodijeliti jedinstven identifikator. Zahtjeve je moguće i grupirati po različitim kriterijima, kao npr. korisnicima (nastavnik, student, blagajnik, administrator...), slučaju korištenja, domenskim konceptima, i sl.

Za svaki zahtjev postavite pitanje: da softver ne nudi ovu funkcionalnost, bi li i dalje bio upotrebljiv i koristan?

Identifikator	Jedinstveni identifikator zahtjeva.
Zahtjev	Opis zahtjeva u obliku „Sustav će omogućiti <funkcionalnost> <objekt> uz <ograničenja> “. Stil pisanja treba biti ujednačen. Prilikom formulacije vodite se prihvaćenim smjernicama za definiranje zahtjeva, kao npr. onima navedenim u <i>INCOSE Smjernicama za pisanje zahtjeva</i> .
Obrazloženje	Obrazloženje zašto zahtjev postoji/zašto je potreban.
Način provjere	Kriterij provjere ili testni scenarij koji će omogućiti utvrđivanje je li zahtjev ispunjen ili nije.
Prioritet [1-5]	Prioritet zahtjeva (1 – najveći prioritet, 5 najmanji prioritet)
Izvor/Porijeklo	Naziv dokumenta kojim je zahtjev propisan ili dionika koji je podnio zahtjev.

>

3.1.Dinamika realizacije zahtjeva (engl. Apportioning of requirements)

<Navedite hoće li svi zahtjevi biti realizirani u inicijalnoj verziji softvera, ili će neki biti ostavljeni za buduće verzija. Navedite ukoliko postoje još neki zahtjevi koje se planira realizirati u budućnosti. >

4. NEFUNKCIONALNI ZAHTJEVI

4.1. Izgled softvera

<Navedite zahtjeve (ukoliko postoje) koji su povezani s izgledom i vizualnim stilom softvera. To može uključiti zahtjeve da se npr. koristi formalan i korporativan stil (u slučaju poslovne aplikacije), ili zaigran stil (npr. računalna igra za djecu); zahtjev za korištenjem konkretne palete boja ili kontrola kako bi aplikacija bila u skladu s brendiranjem poduzeća i sl.>

4.2. Upotrebljivost softvera

<Navedite zahtjeve (ukoliko postoje) koji su povezani s lakoćom učenja i korištenja softvera, prilagodbom za osobe s poteškoćama, lokalizacijom. To može uključiti zahtjeve vezane uz krivulju učenja, brzinu korištenja aplikacije (npr. brzina unosa podataka), lakoću pamćenja opcija softvera, frekvenciju grešaka koje korisnik napravi u radu sa softverom, mogućnost odabira jezika, mogućnost korištenja od strane slijepih osoba i sl.>

4.3. Performanse softvera

<Navedite zahtjeve (ukoliko postoje) koji su povezani s performansama softvera. To može uključiti zahtjeve vezane uz brzinu procesiranja zadataka, odziv, preciznost rezultata, kapacitet pohrane, skalabilnost, dostupnost sustava i sl.>

4.4. Izvođenje softvera i okruženje

<Navedite zahtjeve (ukoliko postoje) koji su povezani s izvođenjem softvera i okruženjem u kojem se softver izvodi. To može uključivati zahtjeve vezane uz fizičko okruženje u kojem se nalazi sustav (glasno okruženje, jako osvijetljenje, prašina i sl.), drugi postojeći sustavi unutar kojih se softver treba izvoditi ili s kojima treba imati interakciju (operacijski sustav, drugi softver od kojeg preuzimamo podatke ili ih šaljemo). >

4.5. Sigurnost i privatnost

<Navedite zahtjeve (ukoliko postoje) koji su povezani sa pitanjima sigurnosti i privatnosti podataka, te standardima i propisima vezanima uz tu problematiku. To može uključivati zahtjeve za korištenjem propisanih sigurnosnih procedura, tehnologija, usklađenost sa pravnim okvirima i sl.>

4.6.Ostalo

<Navedite ostale nefunkcionalne zahtjeve koji nisu prethodno navedeni.>

5. SKICE ZASLONA

<Vizualizirajte značajke interakcije između krajnjeg korisnika i softverskog rješenja kroz skice zaslona (engl. wireframe). Svrha skica je da na vizualan način predložimo i komuniciramo što aplikacija treba raditi, a ne da izradimo realan dizajn grafičkog sučelja. Pri tome skice možete na bilo koji način nacrtati (npr. ručno na papiru + slikanje s mobitelom, MS Paint, MS Word, Excel...)>