**Kratak pregled:** Sadržaj dobrog SRS dokumenta karakteriše formalan jezik i dosljednost u poštovanju standarda po kojem se dokument piše. Kvaliteti dobro napisanog SRS dokumenta ogledaju se u tome što znatno pojednostavljuje implementaciju sistema, jer daje detaljan opis svih zahjeva i slikovito prikazuje procese (pomoću dijagrama). Ovaj dokument daje opis sistema za pomoć na putu. Cilj je da se specifikuju zahjevi softvera koji se razvija.

**Ključne riječi:** dispečer, radnik, administrator, supervisor, specifikacija softvera, specifikacija zahtjeva

**Uvod**

Opisani su preporučeni pristupi za specifikaciju softverskih zahtjeva. Dokument je zasnovan na modelu u kojem je rezultat procesa specifikacije softverskih zahtjeva nedvosmislen I potpuni document specifikacije. Trebalo bi da pomogne:

1. Kupcima softvera da precizno opišu šta žele da dobiju;
2. Dobavljačima softvera da razumiju šta tačno korisnik želi;
3. Pojedincima da postignu sljedeće ciljeve:
4. Izraditi standardnu specifikaciju softvera (SRS) za svoje organizacije;
5. Definisati format i sadržaj specifičnih specifičnih specifikacija softverskih zahtjeva;
6. Razviti dodatne lokalne podrške kao što su kontrolna lista kvaliteta SRS ili priručnik za one koji pišu SRS dokument.

Klijentima, dobavljačima i drugim pojedincima, dobar SRS treba da pruži nekoliko specifičnih koristi, kao što su:

* Uspostaviti osnovu za sporazum između kupaca i dobavljača o tome šta softver treba da uradi. Kompletan opis funkcija koje treba da izvrši softver naveden u SRS-u će pomoći potencijalnim korisnicima da utvrde da li određeni softver zadovoljava njihove potrebe ili kako softver mora biti modifikovan kako bi se zadovoljile njihove potrebe.
* Smanjiti razvojni napor. Priprema SRS-a primorava razne zainteresovane grupe u organizaciji kupca da rigorozno razmotre sve zahtjeve prije početka dizajna i smanji naknadni redizajn i ponovno testiranje. Pažljiv pregled zahtjeva u SRS-u može otkriti propuste, nesporazume i nedosljednosti ranije u razvojnom ciklusu kada su ovi problem lakši za ispravljanje.
* Obezbijediti osnovu za procjenu troškova i rasporeda. Opis proizvoda koji se razvija kako je dat u SRS-u je realna osnova za procjenu troškova projekta i može se koristiti za dobijanje saglasnosti za ponude ili procjene cijena.
* Obezbijediti osnove za validaciju i verifikaciju. Organizacije mogu razviti svoje planove validacije i verifikacije mnogo produktivnije od dobrog SRS-a. U sklopu ugovora o razvoju, SRS obezbjeđuje osnovnu tačku prema kojoj se može mjeriti usklađenost.
* Omogućiti transfer. SRS olakšava prenos softvera na nove korisnike ili nove mašine. Korisnicima je lakše prenijeti softver u druge dijelove svoje organizacije, a dobavljačima je lakše prenijeti na nove kupce.
* Služi kao osnova za poboljšanje. Pošto SRS diskutuje o proizvodu, služi kao osnova za kasnije poboljšanje gotovog proizvoda. SRS će možda morati da se izmijeni, ali on pruža osnovu za kontinuiranu procjenu proizvodnje.

1. **Pregled**

Pregled je podijeljen na pet klauzula. Klauzula 1 objašnjava obim ovog dokumenta. Klauzula 2 navodi reference na druge standarde. Klauzula 3 daje definicije specifičnih izraza koji se koriste. Klauzula 4 daje osnovne informacije za pisanje dobrog SRS-a. Klauzula 5 razmatra svaki od bitnih dijelova SRS-a. Dokument takođe imam dva aneksa, jedan koji obezbjeđuje alternativne šablone formata i onaj koji daje smjernice za usklađenost sa IEEE/EIA 12207.1-1997.

* 1. **Obim**

Obim dokumenta ukratko opisuje sistem koji se razvija. Sistem je namijenjen za brzu i efikasnu pomoć na putu u slučaju manjih ili većih kvarova vozila na putu. Aplikacija sa strane dispečera, sa komunikacijom sa radnicima na terenu (dispečer šalje podatke o mjestu potrebe, radnik na terenu vraća izvještaj). Sistem se izvršava u realnom vremenu. Sistem prati dostupnost vozila (radnika). Korisnici imaju slojevit pristup (dispečer, terenski radnik, sistemski administrator, supervizor). Dispečer se bavi organizacijom poziva i intervencija. Sistemski administrator upravlja nalozima. Supervizor ima pristup izvještajima i radnicima.

1. **Preporuke**

Ova preporučena praksa će se koristiti zajedno sa sljedećim publikacijama:

ASTM E1340-96, Standard Guide for Rapid Prototyping of Computerized Systems.

IEEE Std 610.12-1990, IEEE Standard Glossary of Software Engineering Terminology. IEEE Std 730-1998, IEEE Standard for Software Quality Assurance Plans. IEEE Std 730.1-1995, IEEE Guide for Software Quality Assurance Planning.

IEEE Std 828-1998, IEEE Standard for Software Configuration Management Plans.

IEEE Std 982.1-1988, IEEE Standard Dictionary of Measures to Produce Reliable Software.

IEEE Std 982.2-1988, IEEE Guide for the Use of IEEE Standard Dictionary of Measures to Produce Reli-able Software.

IEEE Std 1002-1987 (Reaff 1992), IEEE Standard Taxonomy for Software Engineering Standards.

IEEE Std 1012-1998, IEEE Standard for Software Verification and Validation.

IEEE Std 1012a-1998, IEEE Standard for Software Verification and Validation: Content Map to IEEE/EIA 12207.1-1997.

IEEE Std 1016-1998, IEEE Recommended Practice for Software Design Descriptions. IEEE Std 1028-1997, IEEE Standard for Software Reviews.

IEEE Std 1042-1987 (Reaff 1993), IEEE Guide to Software Configuration Management.

IEEE P1058/D2.1, Draft Standard for Software Project Management Plans, dated 5 August 1998.

IEEE Std 1058a-1998, IEEE Standard for Software Project Management Plans: Content Map to IEEE/EIA 12207.1-1997.

IEEE Std 1074-1997, IEEE Standard for Developing Software Life Cycle Processes.

IEEE Std 1233, 1998 Edition, IEEE Guide for Developing System Requirements Specifications.

1. **Definicije**

Generalno, definicije pojmova koje se koriste u ovoj preporučenoj praksi odgovaraju zahtjevima IEEE Std 610.12-1990. Sljedeće definicije su ključni termini koji se koriste u ovoj preporučenoj praksi:

* 1. **dispečer:** Dispečer je osoba koja se bavi organizacijom poziva i intervencija.
  2. **radnik:** Radnik kreira izvještaje.
  3. **administrator:** Administrator upravlja nalozima.
  4. **supervizor:** Supervizor ima pristup izvještajima i radnicima.

1. **Razmatranja za pisanje dobrog SRS-a**

Ova klauzula pruža osnovne informacije koje treba uzeti u obzir prilikom pisanja SRS-a. Ovo uključuje sljedeće:

1. Prirodu SRS-a;
2. Okruženje SRS-a;
3. Karakteristike dobrog SRS-a;
4. Zajednička priprema SRS-a;
5. Evolucija SRS-a;
6. Izrada prototipa;
7. Dizajn ugradnje u SRS;
8. Uključivanje projektnih zahtjeva u SRS.
   1. **Priroda SRS-a**

SRS je specifikacija za određeni softverski proizvod, program ili skup programa koji obavljaju određene funkcije u određenom okruženju. SRS može pisati jedan ili više predstavnika isporučitelja, jedan ili više predstavnika kupaca, ili oboje. Podklauzula 4.4 preporučuje oboje.

Osnovna pitanja koja će pisac SRS-a adresirati su sljedeća:

1. *Funkcionalnost.* Šta softver treba da radi? Softver pruža usluge pomoći na putu. Detaljniji opis u tački 1.1.
2. *Spoljašnji interfejs.* Kako softver interaguje sa ljudima, hardverom sistema, drugim hardverom i drugim softverom? Softver interaguje sa ljudima putem mobilnog telefona. Hardver sistema su računari i mobilni uređaji. Nema sekundarnog softvera.
3. *Performanse.* Kolika je brzina, raspoloživost, vrijeme odziva, vrijeme oporavka različitih softverskih funkcija,...? Raspoloživost i brzina su sekvencijalni jer postoji samo po jedan član (dispečer, radnik, administrator, supervizor) koji obavljaju poslove jedan za drugim. Zbog ovoga, vrijeme odziva može biti relativno sporo. Za pojedine izuzetne slučajeve su implementirane posebne funkcionalnosti koje ih razrješavaju.
4. *Atributi.* Koja su razmatranja za prenosivost, ispravnost, održivost, bezbijednost i sl? Aplikacija je za korisnika prvenstveno namijenjena kao mobilna aplikacija, dok se sa serverske strane koriste računari. Kao što je rečeno pod tačkom d), neki izuzeci će biti dodatno riješeni. Sistem je održiv i bezbijedan (**NAKNADNO**).
5. *Ograničenja dizajna nametnuta implementaciji.* Postoje li zahtjevani standardi na snazi, programski jezik za implementaciju, pravila za integritet baze podataka, ograničenja resursa, radno okruženje,...? Za implementaciju se koristi programski jezik Java, pravila za integritet **NAKNADNO**, ograničenje resursa se ogleda u tome što postoji samo po jedan član (dispečer, radnik, administrator, supervizor). Radno okruženje **NAKNADNO**

Pisci SRS-a bi trebalo da izbjegavaju postavljanje zahtjeva za projektovanje ili projekat u SRS. Za preporučeni sadržaj SRS-a pogledajte klauzulu 5.

* 1. **Okruženje SRS-a**

Važno je razmotriti dio koji SRS igra u ukupnom projektnom planu, koji je definisan u IEEE Std 610.12-1990. Softver može u suštini sadržavati svu funkcionalnost projekta ili može biti dio većeg sistema. U posljednjem slučaju tipično će biti SRS koji će navesti interfejse između sistema i njegovog softverskog dijela i postaviti spoljašnje performanse i funkcionalnost na softverski dio. Naravno, SRS se onda treba složiti sa ovim sistemskim zahtjevima i proširiti ih.

IEEE Std 1074-1997 opisuje korake u životnom ciklusu softvera i primjenljive ulaze za svaki korak. Ostali standardi, kao što su navedeni u članu 2, odnose se na druge dijelove životnog ciklusa softvera i tako mogu dopunjavati softverske zahtjeve.

Pošto SRS ima specifičnu ulogu u procesu razvoja softvera, pisci SRS-a bi trebalo da budu pažljivi da ne prelaze granice te uloge. To znači da bi SRS:

1. Trebalo da pravilno definiše sve softverske zahtjeve. Zahtjev softvera