Sistem za upravljanje predmetima u sudovima-Veritas

Smjernice za dizajn

Verzija 1.0

Istorija revizija

|  |  |  |  |
| --- | --- | --- | --- |
| **Datum** | **Verzija** | **Opis** | **Autor** |
| 21.04.2022 | 1.0 | Završna verzija dokumenta | Danijela Milanović |

Sadržaj

1. Uvod 4

1.1 Svrha 4

1.2 Opseg 4

1.3 Definicija, akronimi i skrećenice 4

1.4 Reference 4

1.5 Pregled 4

2. Opšte smjernice za dizajn i implementaciju 4

2.1 Mapiranje od dizajna do implementacije 4

2.2 Specificiranje interfejsa za podsisteme 4

2.3 Dokumentovanje operacija 4

2.4 Dokumentovanje poruka 4

2.5 Detektovanje, opsluživanje i izvještaji o greškama 4

2.6 Upravljanje memorijom 4

2.7 Distribucija softvera 4

2.8 Kako predstaviti komponente za višekratnu upotrebu 5

2.9 Upotreba već kreiranih biblioteka 5

2.10 Ponovna iskoristivost komponenti 5

3. Smjernice za dizajn baze podataka 5

4. Smjernice za dizajn arhitekture 5

5. UML Stereotipi 5

Smjernice za dizajn

# Uvod

## Svrha

Svrha dokumenta Smjernice za dizajn je specificiranje detaljnog opisa standarda za dizajn, konvencija i idioma koji se koriste pri dizajnu sistema.

## Opseg

Dokumennt Smjernice za dizajn obuhvata precizne smjernice vezane za projektovanje informacionog sistema, odnosno dizajn aplikacije.

## Definicija, akronimi i skrećenice

Svi korišteni, nepoznati pojmovi, akronimi i skraćenice opisani su u dokumentu Riječnik[1].

## Reference

[1] Riječnik,

[2] Smjernice za programiranje,

[3] Dokument o arhitekturi softvera

## Pregled

U nastavku dokumenta nalazi se opisi na koji način su projektovani odrđeni dijelovi sistema. Dokument je podjeljen u sledeće dijelove: opšte smjernice za dizajn, smjernice za dizajn baze podataka, smjernice za dizajn arhitekture.

# Opšte smjernice za dizajn i implementaciju

## Mapiranje od dizajna do implementacije

Za projektovanje informacionog sistema se korišten je softverski alat SAP PowerDesigner. Projektovanje sistema se vrši na apstraktnom nivou, a zatim se prelazi direktno na kod. Kostur sistema se generiše iz modela. Održavanje modela se vrši u skladu sa promjenama koda. Baza podataka se takođe projektuje u softverskom alatu SAP PowerDesigner.

## Specificiranje interfejsa za podsisteme

Svi podsistemi koji čine cijelinu projektovanog informacionog sistema modeluju se odvojeno od modela osnovnog sistema.

## Dokumentovanje operacija

U programskom kodu su dokumentovane definicije svih klasa, svih programerski napisanih funkcija, kao i svih argumenata koje funkcije primaju., povratna vrijednost koju vraćaju. Argumenti klasa nisu dokumentovani ukoliko to nije bilo potrebno. Sve konvencije imenovanja i način kodiranja opisan je u dokumentu Smjernice za programiranje[2].

## Dokumentovanje poruka

Parametri poruka nisu dokumentovani, već samo konkretne poruke i kontekst u kojem se javljaju.

## Detektovanje, opsluživanje i izvještaji o greškama

Za upravljanje izuzecima i greškama koristi se mehanizam izuzetaka. Za neke visoko rizične dijelove sistema, može se koristiti parameter, koji će imati različite vrijednosti u odnosu da li je neka operacija uspješno ili neuspješno završena, i samim tim se koristiti za upravljanje greškama.

## Upravljanje memorijom

Za upravljanje memorijom koristiće se mehanizam Java programskog jezika, odnosno Garbage collector.

## Distribucija softvera

Softver za upravljanje informacionim sisitemom se distribuira i instalira od strane licencirane osobe. Softver se instalira na više računara. Baza podataka informacionog sistema se nalazi na serverima koji se distriburaju zajedno sa softverom.

## Kako predstaviti komponente za višekratnu upotrebu

Sve komponente koje su korištene u izradi sistema su projektovane od strane razvojnog tima.

## ****Upotreba već kreiranih biblioteka****

Pri implementaciji softvera, korišćene su ugrađene biblioteke programskog jezika koji je korišćen. Sve biblioteke koje su naknadno dodate su dokumentovane.

## ****Ponovna iskoristivost komponenti****

Pojedine komponente, koje su generički projektovane, mogu ponovo da se koriste, ali cjelina je jedinstvena i ne smije biti ponovno korištena.

# Smjernice za dizajn baze podataka

* Baza podataka se modeluje u pomoću alata SAP PowerDesigner. Neće se koristiti ORM sistemi. Za modelovanje konceptualnog modela koristi se ER+Merise notacija. Veze više-na-više se modeluju tako što između njih se pravi vezna tabela.
* Svi korišteni tipovi podataka korišteni pri modelovanju su podržani od strane Java programskog jezika.
* Konceptualni model se prebacuje u logički model preko alata SAP PowerDesigner. U logičkom modelu se vrši redukcija kljičeva. Zatim se prebacuje u fizički model na isti način.
* Za imenovanje se koriste smislena imena imenica u jednini. Nazivi kolona i tabela se pišu velikim početnim sovom. Za razdvajanje riječi se koristi razmak. Nazivi kolona i tabela se ne smiju podudarati.

# Smjernice za dizajn arhitekture

Arhitektura softvera je podijeljena na module, tako da pri odstranjivanju jednog modula ostatak sistema nesmetano funkcioniše. Softver obezbjeđuje nesmetanu zamjenu dijelova sistema drugim verzijama. Arhitektura softvera se temelji na MVC šablonu. Detaljan izgled dizajna arhitekture sistema opisan je u dokumentu Dokument o arhitekturi softvera[3].

# UML Stereotipi

Za projektovanje sistema se nisu korišćeni nikakvi dodatni stereotipi, osim onih koje su unaprijed definisani UML i Rational Unified Process-ima.