|  |  |  |
| --- | --- | --- |
| Univerzitet Singidunum Beograd | March 2  2020 | |
| **Nastavni predmeti: Specifikacija i modelovanje softvera Baze podataka Projekat: Rukovalac Informacionim Resursima (InfHandler)**   |  |  | | --- | --- | | **Verzija** | **Datum** | | **0.1** | **02.03.2020.g.** | | **Predmetni nastavnik:** | Prof.dr Branko Perišić, red.prof. | | **Asistent: Specifikacija i modelovanje softvera** | MSc Aleksandra Mitrović, asistent | | **Asistent: Baze podataka** | MSc Mladen Vidović, asistent | | |  |

SADRŽAJ

[Predgovor 3](#_Toc30319736)

[SOFTVERSKI PROCES: 4](#_Toc30319737)

[SOFTVERSKI PROIZVOD: 5](#_Toc30319738)

[1. Kako FORMALIZOVATI IDEJU? - 6](#_Toc30319739)

[Osnovni koncepti 6](#_Toc30319740)

[2. DOMEN REŠENJA - šta treba da istakne rešenje koje je predmet razvoja? 7](#_Toc30319741)

[3. OPŠTE OSOBINE ALATA - GUI KOMPONENTA 8](#_Toc30319742)

[4. Prototipski razvoj - Evolutivni prototip 9](#_Toc30319743)

[4.1. Osnovna verzija prototipa 9](#_Toc30319744)

[4.2. Rad sa odabranim informacionim resursom 10](#_Toc30319745)

[4.3. Rad sa povezanim informacionim resursima 11](#_Toc30319746)

[4.3. Rad sa povezanim informacionim resursima - više nivoa 12](#_Toc30319747)

[4.4. Model strukture informacionih resursa 13](#_Toc30319748)

[4.4. Konceptualni model šeme relacione baze podataka 14](#_Toc30319749)

[4.5. Fizički model šeme relacione baze podataka 15](#_Toc30319750)

[5. Prilozi 16](#_Toc30319751)

[5.1. Format modela zahteva 16](#_Toc30319752)

[5.2. Primer modela zahteva 17](#_Toc30319753)

[5.3. Model interakcija korisnika i funkcija programskog proizvoda 18](#_Toc30319754)

[5.3.1. Primer - Dijagram visokog nivoa 18](#_Toc30319755)

[5.4. Model statičke strukture - UML Dijagram KLASA 19](#_Toc30319756)

[5.4.1. Primer modela softverskog proizvoda 19](#_Toc30319757)

# Predgovor

Predmeti **Specifikacija i modelovanje softvera i Baze podataka** okupljeni su oko zajedničkog projekta koji omogućava specifikaciju i modelovanje programskog proizvoda za rukovanje informacionim resursima:

1. na strani klijenta - GUI orijentisani, interaktivni, događajima upravljani softverski alat
2. na strani servera - podacima/informacijama upravljani rukovalac oslonjen na sistem za upravljanje datotekama (serijske, sekvencijalne i indeksne organizacije) i sistemom za upravljanje relacionom bazom podataka (MySQL)

# SOFTVERSKI PROCES:

* + ***Model životnog ciklusa***:
    - ***Prototipski razvoj*** - evolutivni prototip
  + ***Metodologija - Modelom upravljani razvoj softvera:***
    - ***Izrada vizije softverskog proizvoda i njeno formalno predstavljanje*** (tekstualna specifikacije);
    - ***Specifikacija i modelovanje zahteva*** *(uz oslonac na usvojeni alat za modelovanje);*
    - ***Specifikacija i modelovanje poslovnih funkcija i interakcija*** *(UML - USE CASE, BUSINESS USE CASE formalizmi uz oslonac na usvojeni alat za modelovanje)*
    - ***Specifikacija i modelovanje arhitekture -*** *(MVC arhitektonski obrazac uz paketiranje modela i koda, transformacija funkcionalni dijagrama u statičku strukturu softverskog sistema)*
    - ***Specifikacija i modelovanje statičke strukture komponenti*** *- (UML - KLASNI MODELI uz oslonac na usvojeni alat za modelovanje)*
    - ***Specifikacija i modelovanje ponašanja*** *- (UML dinamički dijagrami odabrani iz skupine usvojenih formalizama za opis ponašanja softverskih sistema - dijagram stanja, dijagram aktivnosti, dijagram sekvence i dijagram saradnje)*
    - ***Specifikacija i modelovanje podataka*** *(Konceptualni. Logički i Fizički model podataka oslonjen na organizaciju podataka i baze podataka zasnovane na relacionom modelu podataka)*
    - ***Specifikacija i modelovanje serverske strane*** *(poslovna logika, CRUDE operacije -* ***CR****eate,****U****pdate,* ***DE****lete, upiti i izveštavanje)*

# SOFTVERSKI PROIZVOD:

* + ***Interaktivan, GUI orijentisan, Događajima upravljan programski Alat***
  + ***Standardizacija arhitekture***:
    - ***Spoljašnja arhitektura*** - komponente
      * Komponente koje manipulišu sa radnim okruženjem:
      * Komponente koje manipulišu sa ***artifaktima*** ***softverskog proizvoda*** (proizvodi koje softverski proizvod kreira i sa njima rukuje)
      * Uputstva i pomoć pri ovladavanju programskim proizvodom;
    - ***Unutrašnja arhitektura komponenti***:
      * MVC (Model-View-Controller) - arhitektonski obrazac
    - ***Podešavanje programskog proizvoda*** (*Parametrizacija, Lokalizacija i personalizacija*)
      * Logotip i o aplikaciji
        + ***Algoritmi, strukture podataka i organizacija podataka***
    - ***Model informacionog resursa***
    - ***Serijalizacija i deserijalizacija - (****čuvanje radnog prostora, kolekcija informacionih resursa, informacionih rsursa i produkata pretrage i izveštavanja****)***
  + ***Prototip*** *- Evolutivni prototip uz oslonac na Objektno Orijentisanu Paradigmu (OO programski jezik* ***Python*** *i QT* ***grafička biblioteka*** *i* ***razvojnog okruženja*** *u skladu sa ranije usvojenim znanjima i veštinama)*

# 1. Kako FORMALIZOVATI IDEJU? -

# Osnovni koncepti

Rukovalac informacionim resursima: (*InfHandler*)

* **Domen problema: - Dizajniranje i operativna upotreba informacionih resursa** (*Informacioni resurs - uređena kolekcija podataka*)
* **Zainteresovane strane: - *Organizacioni sistemi i pojedinci koji u osnovi svoje delatnosti imaju rad sa informacionim resursima slobodne strukture i organizacije.***
  + **Koga kontaktirati u cilju smanjenja stepena nepoznavanja domena problema i domena rešenja?**
    - **Eksperte iz domena problema -** operativno se bave rukovanjem informacionim resursima (poslovni analitičari) i poseduju neophodna iskustva vezana za domen izvršavanja;
    - **Eksperte iz domena upotrebe -** operativni korisnici informacionih resursa;
    - **Poslovni eksperti** - poseduju operativne pokazatelje (podatke/informacije/bazu znanja) vezane za poslovne aspekte softverskog proizvoda;
    - **Eksperte za razvoj softvera -** vladaju konceptima izgradnje fleksibilnih programskih proizvoda.
* **Vrsta proizvoda koji želimo razviti: - Alat (proširivi radni okvir)**
* **Način plasmana: Slobodno tržište - ne projektuje se za poznatog kupca.**

# 2. DOMEN REŠENJA - šta treba da istakne rešenje koje je predmet razvoja?

* Rukovanje Informacionim resursima - Opis, definisanje, operativna upotreba i trajno čuvanje informacionih resursa proizvoljne strukture - platformski nezavisno i platformski zavisno (serijska, sekvencijalna i indeksna datoteka, sistem za upravljanje relacionom bazom podataka);
* Rukovanje Kolekcijama informacionih resursa - Slobodno formiranje kolekcija datoteka i tabela u bazi podataka u skladu sa modelom informacionog resursa (Prilog 1.);
* Rukovanje pravima pristupa i rada sa informacionim resursima.
* Rukovanje Radnim prostorima alata - Slobodno formiranje radnih prostora koji sadrže kolekcije informacionih resursa različite organizacije i tehnike skladištenja;
* Rukovanje Parametrima Komponenti Arhitekture Softverskih Proizvoda

# 3. OPŠTE OSOBINE ALATA - GUI KOMPONENTA

Opšta Arhitektura GUI komponente softverskog proizvoda:

1. **Naslovna linija ( *titlebar* )**

2. **Glavna linija menija ( *menubar* )**

3. **Paleta alatki ( *toolbar* )**

4. **Radna površina ( *workspace*)**

5. **Statusna linija ( *statusbar* )**



# 4. Prototipski razvoj - Evolutivni prototip

## 4.1. Osnovna verzija prototipa

**Osnovni ram aplikacije**

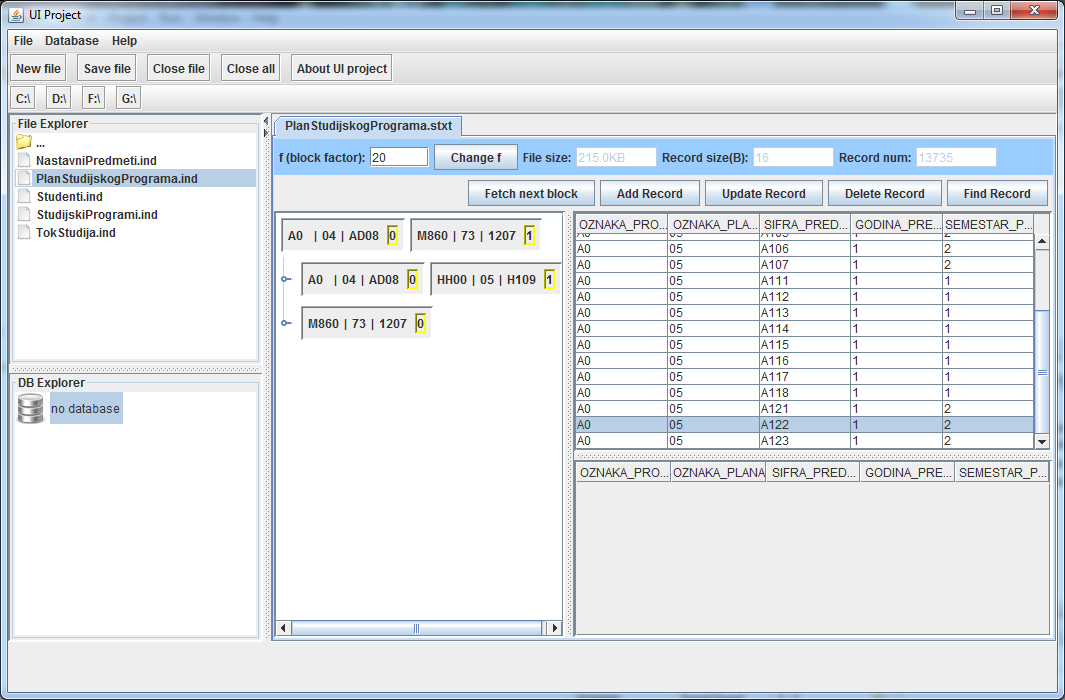
# 

**Informacioni resursi**

**Radna zona Otvorenih Informacionih resursa**

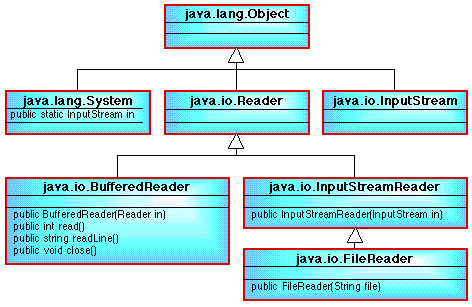
**Radna zona Povezanih Informacionih resursa**

## 4.2. Prototip druga verzija

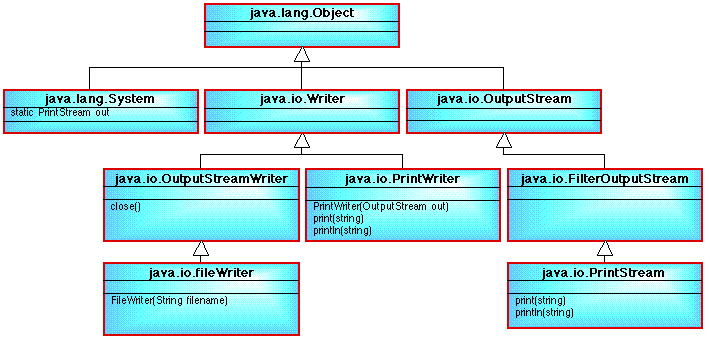


## 4.3. Java - rad sa datotekama

### 4.3.1. Java - Preuzimanje sadržaja iz datoteke



### 4.3.2. Java - Ažuriranje sadržaja datoteke



### 4.3.3. Java - Rukovanje informacionim resursima - JTable









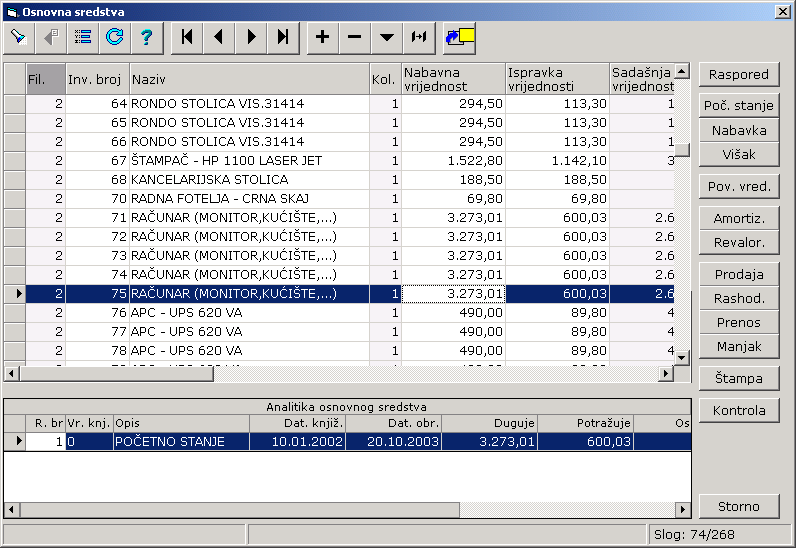




## 4.4. Rad sa odabranim informacionim resursom

# Slika01_StrukturaStandardneForme

## 4.5. Rad sa povezanim informacionim resursima



Osnovni

browser

Vezani

browser

## 4.6. Rad sa povezanim informacionim resursima - više nivoa

# 

## 4.7. Model strukture informacionih resursa

# 

## 4.8. Konceptualni model šeme relacione baze podataka

# 

## 4.9. Fizički model šeme relacione baze podataka

# 

# 5. Prilozi

## 5.1. Format modela zahteva

Model zahteva je neophodno predstaviti u donjoj tabelarnoj formi:

Tabela 5.1.1. Format modela zahteva

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| R.br. | Opis | Tip | Prioritet | Zainteresovane strane | Rizik | Status | Procena napora |
| 1. | *Tekstualna formulacija zahteva* |  |  |  |  |  | *Efektivni broj dana* |
| 2. |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
| n- |  |  |  |  |  |  |  |

|  |  |
| --- | --- |
| Status | Značenje |
| Dr | Draft |
| De | Definisan |
| Po | Potvrđen |

|  |  |
| --- | --- |
| Rizik | Značenje |
| Vi | Visok |
| Sr | Srednji |
| Ni | Nizak |

|  |  |
| --- | --- |
| Oznaka | Naziv |
| 1 |  |
| 2 |  |
| 3 |  |
|  |  |
| k |  |

|  |  |
| --- | --- |
| Prioritet | Opis tipa |
| ES | Esencijalni |
| VP | Vrlo poželjan |
| PO | Poželjan |
| OP | Opcion |
| NP | Nepoželjan |

|  |  |
| --- | --- |
| Tip | Opis tipa |
| F | Funkcionalni |
| N | Nefunkcionalni |

Model zahteva treba da ima potpun obuhvat (svi grupi zahtevi) uz specificiranje detalja u skladu sa rezultatima pojedinačnih etapa u procesu prikupljanja i analize zahteva.

## 5.2. Primer modela zahteva

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| R.br. | Opis | Tip | Prioritet | Zainteresovane strane | Rizik | Status | Procena napora |
| 1. | **Instalacija**  Neophodno je obezbediti podršku instalaciji PROGRAMSKOG PROIZVODA. | N | Op | Administrator alata | Vi | De | *35* |
| 2. | **Pokretanje programskog proizvoda**  Neophodno je obezbediti pokretanje programskog proizvoda na način kako je to definisano prilikom njegove instalacije. Prilikom pokretanja neophodno je dovesti programski proizvod u kontekst u kom je ostavljen kod poslednjeg prekida rada. | N | Es | Operativni korisnik | Vi | De | 12 |
| 3. | **Operativna upotreba programskog proizvoda** | F | Es | Operativni korisnik | Vi | Dr | 0 |
| 4. | **Prekid rada programskog proizvoda**  Neophodno je, u bilo kom trenutku operativne upotrebe, obezbediti kontrolisani prekid rada programskog proizvoda. Prilikom prekida rada neophodno je očuvati konzistentnost svih artifakata sa kojima je manipulisano u toku rada. Potrebno je sačuvati kontekst upotrebe programskog alata u cilju njegove restauracije kod sledećeg pokretanja. | F | Es | Operativni korisnik | Vi | De | 12 |
|  | . |  |  |  |  |  |  |
|  | . |  |  |  |  |  |  |
|  | . |  |  |  |  |  |  |
| n- |  |  |  |  |  |  |  |

## 5.3. Model interakcija korisnika i funkcija programskog proizvoda

### 5.3.1. Primer - Dijagram visokog nivoa



## 5.4. Model statičke strukture - UML Dijagram KLASA

### 5.4.1. Primer modela softverskog proizvoda

