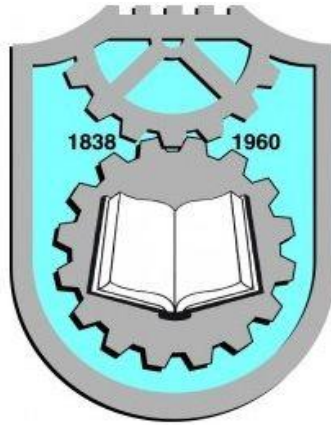


Факултет инжењерских наука Универзитета у Крагујевцу



Пројектовање информационих система и база података

**Тема: Развој информационог система за
имплементацију рада скриптарнице**

Студент:

Јелена Глишић 566/2015

Професор:

др Милан Ерић

Крагујевац, 2019.

Садржај

1. Увод – информационе потребе посматраног реалног система	3
2. Релевантни документи реалног система.....	5
3. Дијаграми контекста, стабла активности и декомпозиције реалног система	8
3.1. Дијаграм контекста	8
3.2. Дијаграм стабла активности.....	10
3.3. Дијаграм декомпозиције.....	12
4. Модел тока података и логичка шема базе података	14
4.1. Модел (дијаграм) тока података	14
4.2. Логичка шема базе података	16
5. Физичка шема базе података и имплементација са тестним подацима	19
6. Развој апликације (софтвера) реалног система.....	23
7. Литература.....	31

1. Увод – информационе потребе посматраног реалног система

Корен речи скриптарнице је скрипта (*lat. scriptum-написано*). У данашње време, скриптом се подразумевају нека сажета издања, нешто што је скраћено и лакше формулисано од свог оригинала. Потреба за развојем скрипти јавила се још у средњем веку када је вршено преписивање књига како би се исте умножиле и биле доступне што широј јавности. То се вршило у библиотекама и скрипторијумима где су се прво ручно преписивале књиге а касније су машине обављале тај посао. Треба се подсетити и наших средњовековних преписивачких школа које су уствари били манастири где су монаси преписивали дела, најпознатија је Ресавска преписивачка школа. Упркос значају и утицајима Ресавске школе, данас је нажалост у друштву раширен израз „Ресавска школа“ који има потпуно другачије значење.

Данас, под појмом скриптарнице подразумева се просторија у згради неке установе где се може извршити копирање, куповина материјала, куповина школског прибора, куповина издавачке делатности те установе... Скриптарница представља функционалну целину у којој се обављају помоћне делатности за које је установа регистрована – трговина на мало у специјализованим продавницама, односно трговина на мало посредством поште или интернета.

Обзиром да је ово доба – доба високе технологије и доба у коме је све окренуто томе да се додатно олакша и подреди човеку, идеја пројекта јесте развој информационог система за имплементацију рада једне скриптарнице. Информациони систем који је развијен има за циљ олакшано добијање информација о тренутном стању услуга у скриптарници, њиховој цени, шифри, о неким основним карактеристикама услуга и производа. Наиме, овакав систем би запосленом у скриптарници омогућио све информације о услугама и производима без тога да мора да гледа по књигама како је шта уписано, под којим бројем... Основни циљ је смањити папирологију и прећи на дигитализацију. Уместо трагања по папирима, запослени би могао једним кликом да излиста све информације које би га интересовале. Препорука је да се производи пакују по истом редоследу као шифре у систему, да би се још олакшало и скратило време трагања. Овај

информациони систем би смањιο редовне и напорне гужве у скриптарницама и омогућио лакше привикавање ново запослених.

Сврха овог документа је да јасно и детаљно опише функционалност и могућности система који представља систем за имплементацију рада скриптарнице.



Слика 1. Изглед скриптарнице

2. Релевантни документи реалног система

Како би развој концепта оваквог информационог система био могућ, потребно је прикупити релевантне документе тренутног система и целокупног окружења где би тај систем био имплементиран и на основу њих направити нова унапређена решења. Ово је можда и најважнија активност приликом пројектовања информационог система. Унутар ње се врши тачно дефинисање захтева корисника. Током ове активности, пројектант информационог система се упознаје са потребама и жељама корисника, као и са свим изазовима које реална имплементација унутар радног окружења доноси са собом.

Унутар ове фазе потребно је такође дефинисати и следеће ствари:

- Дефинисање захтева из докумената
- Дефинисање захтева интервјуом
- Дефинисање матрице односа
- Анализа захтева корисника

Током фазе дефинисање захтева из докумената, потребно је прикупити како улазне тако и излазне документе. Такође, није лоше имати и узорке које корисници већ користе, да би се нови извештаји на што бољи начин креирали – задржали оно што је најбоље и притом добили још напредније – квалитетније ставке. Дефинисање захтева из докумената извршено је уз помоћ прикупљања података са интернета разних сајтова факултета на којима постоје скриптарнице као и копијерница како би се увиделе реалне цене услуга копирања. Тако је и направљена база података скриптарнице и њених карактеристика које су биле потребне за рад информационог система.

Дефинисање захтева интервјуом заузима посебан значај приликом дефинисања захтева које један информациони систем треба да испуни. Током ове активности се најбоље разумеју проблеми (које виде корисници – непосредни извршиоци, овде запослени) као и циљеви које је потребно испунити. Ова фаза је спроведена над насумично одабраним запосленима у скриптарници, различитог пола и старости. На питање „како бисмо Вам могли олакшати рад у скриптарници“, као одговор у већини случајева добио се: „постављање некаквог система који би имао шифре и цене за сваки производ у радњи“. На основу одговора, закључено је да је веома

битно обезбедити корисницима увид у шифре, називе, цене производа и услуга у скриптарници.

Након дефинисања захтева може се почети са дефинисањем матрица односа. Унутар ове активности, повезују се ентитети добијеном анализом докумената са активностима из стабла активности.

За сваки ентитет се одређује на који начин свака од активности користи тај ентитет, то се постиже помоћу CRUD матрице и на основу те матрице, ентитет се у оквиру неке активности може:

- Креирати (Create – C)
- Претражити (Retrieve – R)
- Ажурирати (Update – U)
- Обрисати (Delete – D)

Испод је дата табела која дефинише CRUD матрицу за разматрани концепт информационог система.

Назив активности	Назив ентитета	Однос - CRUD			
Одржавање шифарника	Књиге	C	R	U	D
	Школски прибор	C	R	U	D
	Услуге	C	R	U	D
Одржавање података	Књиге	C	R	U	D
	Школски прибор	C	R	U	D
	Услуге	C	R	U	D
Израда извештаја	Књиге		R		
	Школски прибор		R		
	Услуге		R		

Табела 1 – Дефинисање CRUD матрице

Једна од најважнијих ставки приликом пројектовања информационог система је тачно дефинисање захтева корисника. То је анализа захтева корисника. Спровеђењем ове активности, пројектант, односно програмер, упознаје се са потребама и жељама корисника како би се лакше развила апликација која ће у потпуности одговарати захтевима корисника. Потреба за развојем ове апликације настала је услед потребе запослених у скриптарници да буду што прецизније обавештени о карактеристикама производа и услуга који су на располагању у скриптарници. Овај систем им је сигуран начин да имају увид у све информације које их интересују и једним кликом би све сазнали. Такође, био би решен проблем дугачких редова чекања. Увидом у постојећу документацију, анализирани су следећи документи:

- Листа производа и услуга који постоје у скриптарници
- Извештаји о датим производима и услугама и најосновније информације о истим

Да би се сагледала целокупна садашња слика, изводи се дефинисање захтева интервјуом. Интервјуом су дефинисани захтеви који не постоје у већ наведеним документима и које би требало уградити у будуће решење:

- Неопходно је омогућити процес додавања производа и услуга из система скриптарнице
- Неопходно је омогућити процес брисања производа и услуга из система скриптарнице
- Неопходно је омогућити процес мењања података производа и услуга из система скриптарнице

После спроведене анализе докумената и додатних захтева добијених интервјуом, може се доћи до закључка да све податке који се уносе, треба памтити у бази података распоређених у три табеле – Knjige, SkolskiPribor и Usluge.

3. Дијаграми контекста, стабла активности и декомпозиције реалног система

На почетку развоја информационог система, потребно је дефинисати дијаграм контекста, стабла активности као и декомпозиције посматраног реалног система. Током ове активности, највише времена треба уложити у дефинисање захтева од стране корисника овог информационог система.

3.1. Дијаграм контекста

Први корак је израда контекстног дијаграма. Контекстни дијаграм се користи да би се одмах, на самом почетку, дефинисале границе наведеног система. Наведени дијаграм се дефинише једним правоугаоником који уједно представља и границу модела који се пројектује и развија. Помоћу стрелица се дефинише ток информација како унутар, тако и ван наведеног система. Важно је напоменути да је контекстни дијаграм највиши ниво апстракције који се касније преводи у ниже нивое апликације. На Слици 2 је приказан контекстни дијаграм развојног концепта информационог система рада скриптарнице.



Слика 2. Контекстни дијаграм

Сама активност је дефинисана глаголском фразом „Праћење рада скриптарнице“ док су стрелице дефинисане и груписане као:

○ Улазни документи:

- ◆ Захтев за потврду ауторизованог корисника
- ◆ Захтев за унос нове књиге
- ◆ Захтев за унос новог производа
- ◆ Захтев за унос нове услуге копирања
- ◆ Захтев за ажурирање књиге
- ◆ Захтев за ажурирање производа
- ◆ Захтев за ажурирање услуге копирања
- ◆ Захтев за брисање књиге
- ◆ Захтев за брисање производа
- ◆ Захтев за брисање услуге копирања

- Излазни документи:
 - ◆ Извештај о књигама
 - ◆ Извештај о производима
 - ◆ Извештај о услугама копирања
- Контроле су упутства и правилници, тачније:
 - ◆ Поступак дефинисања шифарника
 - ◆ Поступак извештавања
 - ◆ Упутство о пословању скриптарнице
- Механизам који се дефинише као:
 - ◆ Корисници
 - ◆ SUBP (Систем за управљање базама података)

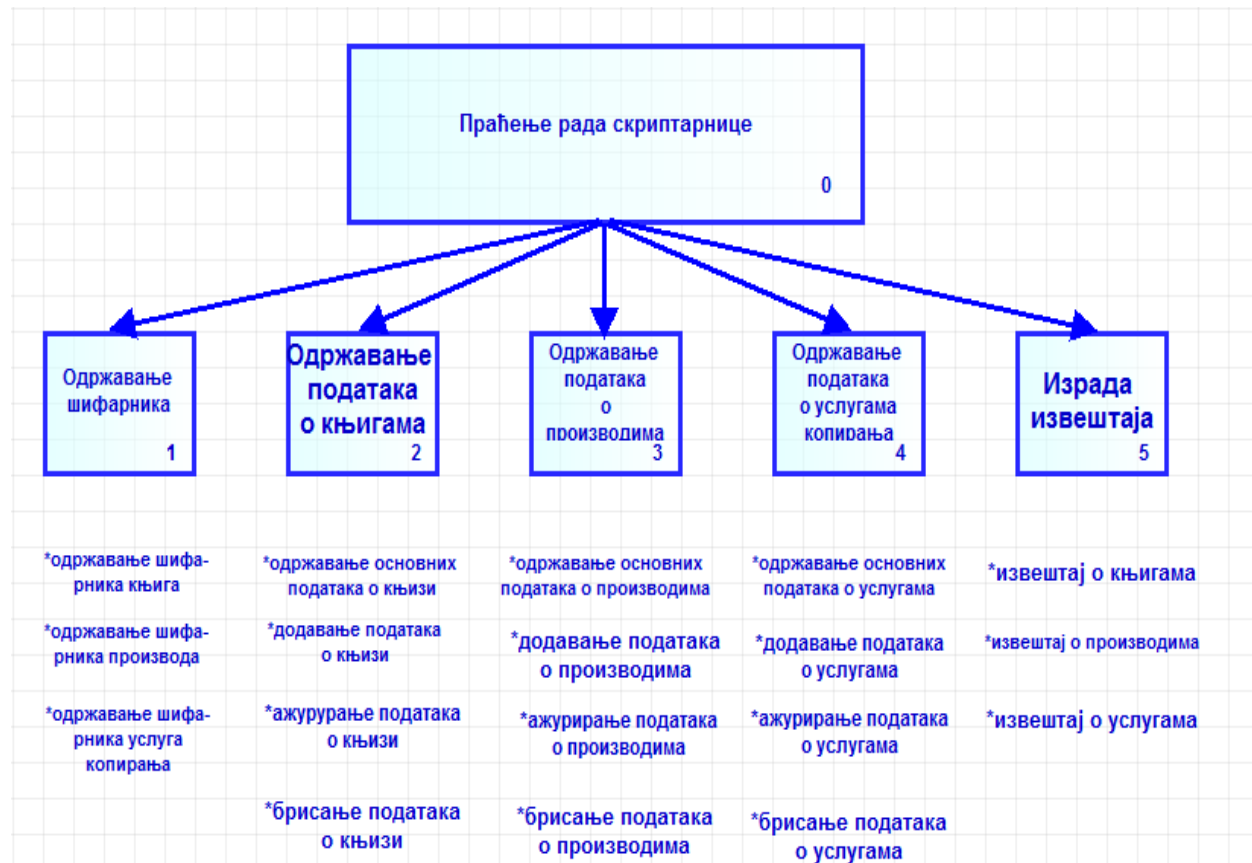
3.2. Дијаграм стабла активности

Стабло активности се дефинише применом методе решавања проблема одозго надоле (top-down) када се сложена активност раставља на више подрешених активности а затим се приступа решавању једноставних подређених активности. То је тип вертикалне хијерархијске везе између активности. Односно, полазна сложена активност се развија у хијерархију подређених активности чија је структура тип стабла. Корен стабла (највиши чвор стабла, root) садржи полазну активност и означава се бројем нула. Бројеви се користе да би се приказало колико детаља садржи одређена активност. Затим, ради се декомпоновање активности тако што се она раздваја на своје делове, ти делови се означавају ознаком која на почетку садржи ознаку свог родитеља као и редни број свог извршавања. Надређена активност се назива родитељ (parent) док су подређене активности – деца (child). Деца или листови, тј. чворови, немају потомке али садрже активности чије је решавање релативно једноставно. Решавањем свих подређених активности из листова, решена је и полазна сложена активност.

Дакле, стабло активности представља хијерхију дефинисаних активности, очишћену од стрелица и омогућује функционалну декомпозицију и увид у дубину одвијања веза између активности.

Пракса је показала да се активност може раздвојити на две или максимално шест подређених активности како би се сматрало да је активност правилно разбијена, односно да унутар једног нивоа нема превише детаља.

За контекстни дијаграм „Праћење рада скриптарнице“, стабло активности приказано је на Слици 3.



Слика 3. Стабло активности

Приликом формирања овог стабла, полази се од опште чињенице да се већина докумената може поделити на:

- Одржавање шифарника
- Одржавање података о књигама
- Одржавање података о производима
- Одржавање података о услугама копирања
- Израда извештаја.

На основу анализе и спроведених интервјуа, активност 1 „Одржавање шифарника“ може се поделити на подређене активности:

- Одржавање шифарника књига
- Одржавање шифарника производа
- Одржавање шифарника услуга копирања.

Активност 2 „Одржавање података о књигама“ обухвата:

- Одржавање основних података о књизи
- Додавање података о књизи
- Ажурирање података о књизи
- Брисање података о књизи.

Активност 3 „Одржавање података о производима“ обухвата:

- Одржавање основних података о производима
- Додавање података о производима
- Ажурирање података о производима
- Брисање података о производима.

Активност 4 „Одржавање података о услугама копирања“ обухвата:

- Одржавање основних података о услугама копирања
- Додавање података о услугама копирања
- Ажурирање података о услугама копирања
- Брисање података о услугама копирања.

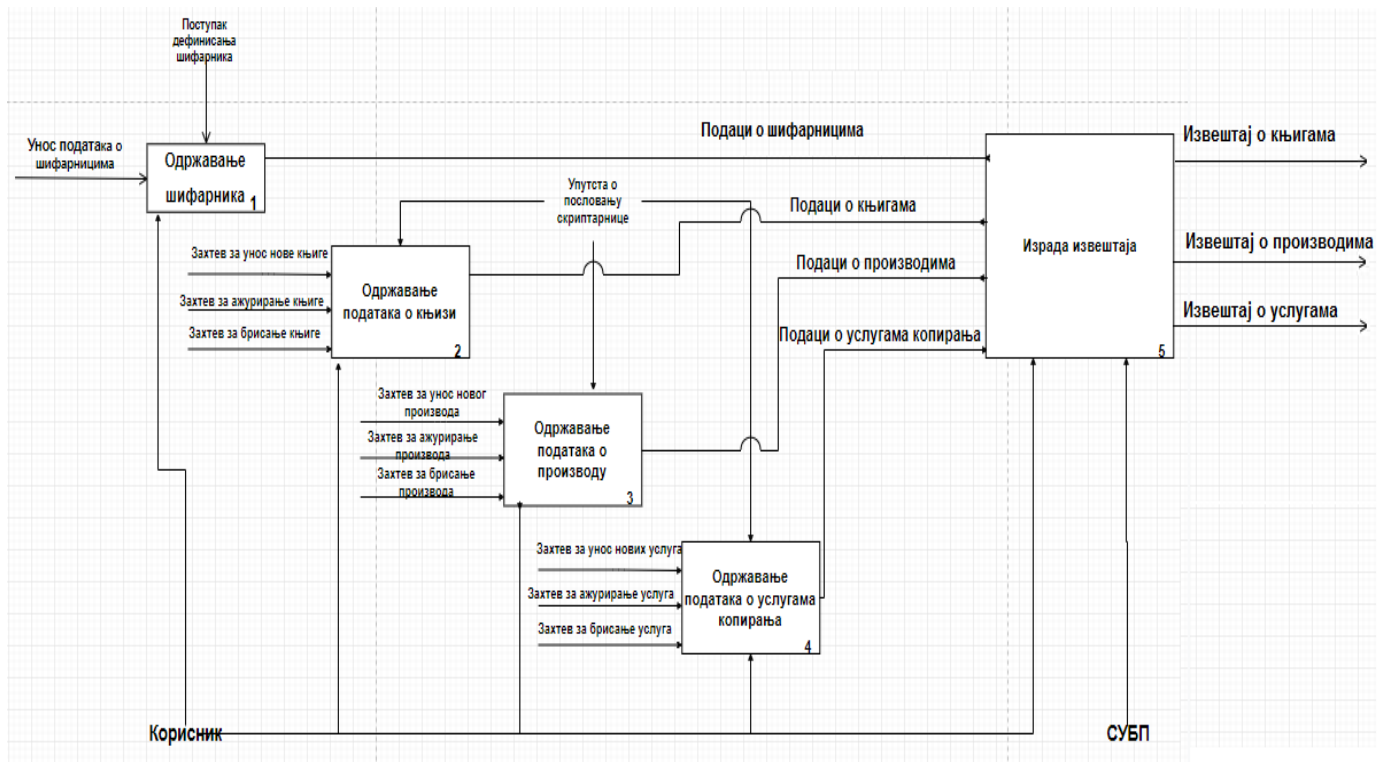
Активност 5 „Израда извештаја“ обухвата:

- Извештај о књигама
- Извештај о производима
- Извештај о услугама копирања.

3.3. Дијаграм декомпозиције

Дефинисање детаљних захтева се изводи помоћу дефинисања декомпозиционог дијаграма. Уколико је потребно, израђује се и детаљно стабло активности. У овом случају, детаљно стабло активности није потребно јер су све подређене активности једноставне и сам принцип њиховог извођења, извршавања је познат.

Следећа ставка за извршење је дефинисање декомпозиционог дијаграма. На тај начин, успостављају се хоризонталне везе између подактивности које су повезане стрелицама. На основу претходно дефинисаног контекстног дијаграма и стабла активности, дефинише се декомпозициони дијаграм који је приказан на Слици 4.



Слика 4. Декомпозициони дијаграм

Као што се може видети на слици изнад, све граничне стрелице контекстног дијаграма су пренете и уједно су све активности приказане. Граничне стрелице су повезане са одговарајућим активностима али су и уведене интерне стрелице које међусобно повезују наведене активности.

За активност 1 „Одржавање шифарника“ дефинисан је интерни излаз „Подаци о шифарницима“ који је уједно интерни улазни податак активности 5 „Израда

извештаја“. Активност 2, 3 и 4 („Одржавање података о књизи“, „Одржавање података о производима“, „Одржавање података о услугама копирања“) имају интерне излазе „Подаци о књигама“, „Подаци о производима“ и „Подаци о услугама копирања“, респективно. Сви ови излази представљају улазе активности 5 „Израда извештаја“. Такође, битно је напоменути да се поред корисника као једног вида механизма, у активности 5 „Израда извештаја“ појављује и SUBP који брине о подацима који су смештени унутар базе података. На крају се добијају три излазна податка у виду докумената који представљају извештаје о књигама, производима и услугама копирања.

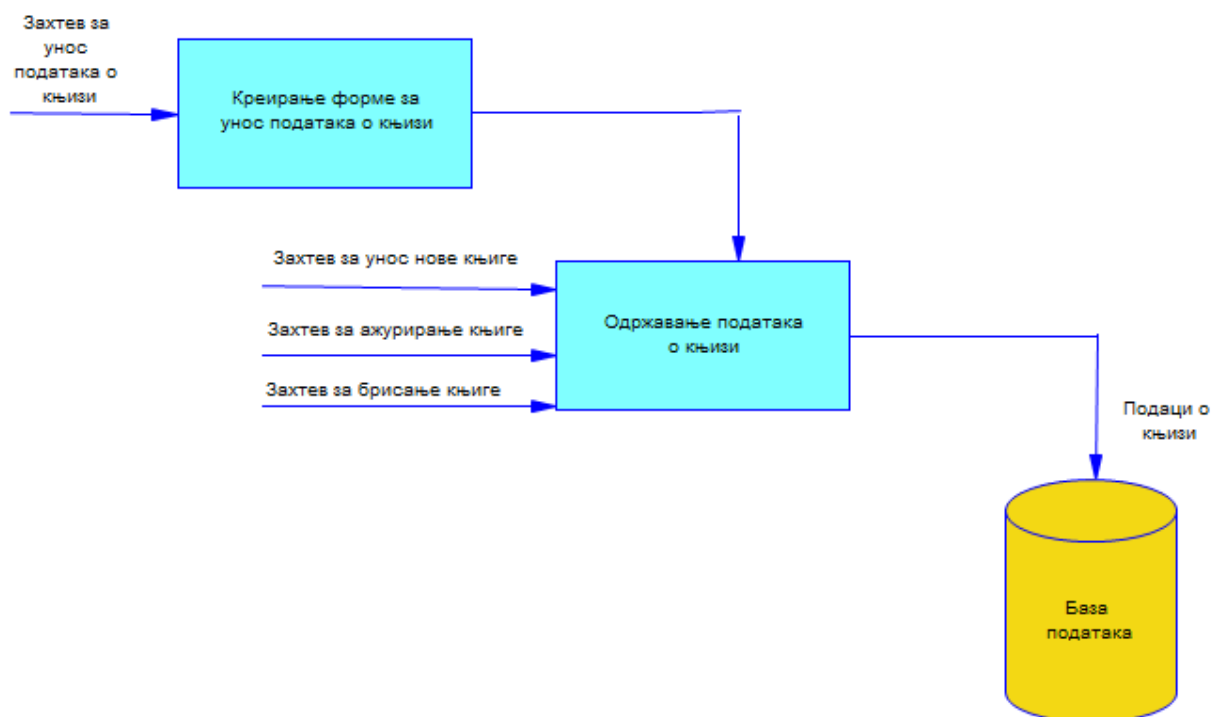
4. Модел тока података и логичка шема базе података и

Наредна фаза је дефинисање дијаграма тока података (DTP), логичке шеме базе података као и релационих ограничења концептуалног информационог система. Ово уједно представља и последњу активност која се бави дефинисањем логичког дела информационог система и после ње се може почети са реализацијом физичког дела.

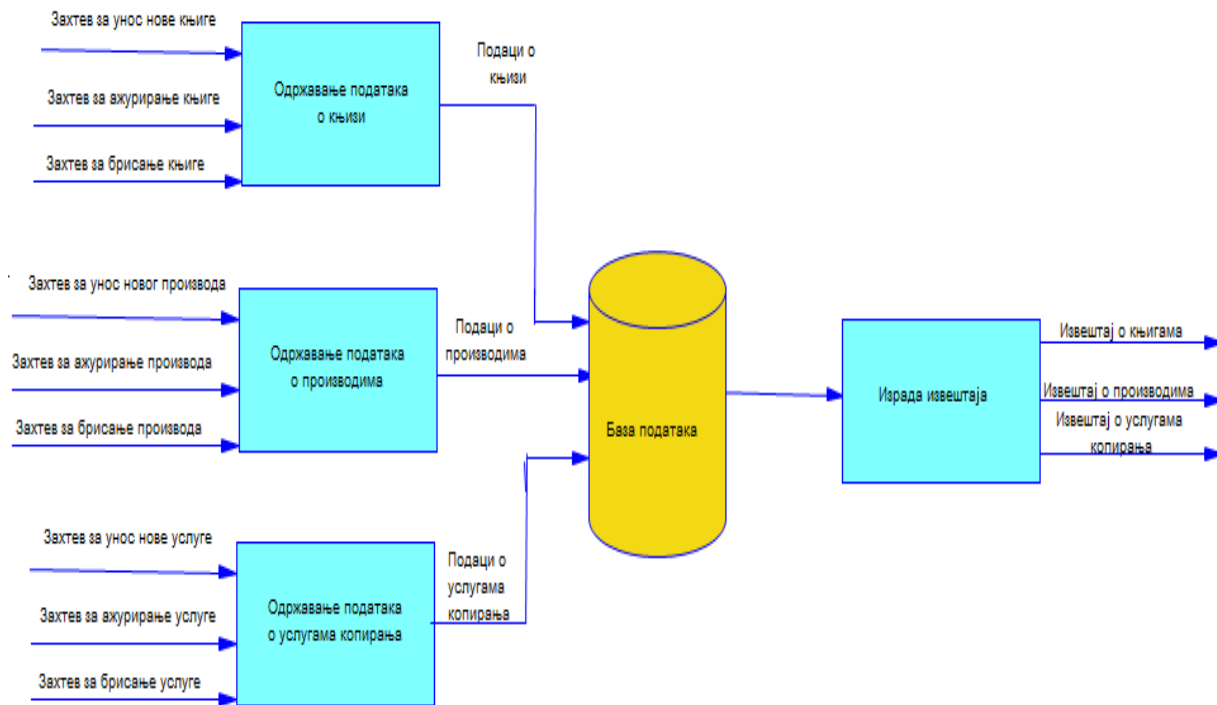
4.1. Модел (дијаграм) тока података

Дијаграм тока података (DTP) представља графички приказ повезивања процеса, токова информација/података као и складишта података. Другим речима, дијаграм тока података се користи за приказ на оном нивоу где се могу дефинисати улази, излази, токови, складишта и процеси. Дијаграм тока података фокусира проблем токова података између процеса и уједно врши анализу складишта података ради максималног повећања њихове расположивости и смањења времена претраживања. Дијаграм тока података се углавном дефинише за ниже нивое декомпозиционог дијаграма.

У овој документацији, приказаће се дијаграми тока података за две активности: „Одржавање података о књизи“ и „Израда извештаја“. Дати су на Слици 5 и Слици 6, респективно.



Слика 5. Дијаграм тока података активности 2 „Одржавање података о књизи“



Слика 6. Дијаграм тока података активности 5 „Израда извештаја“

4.2. Логичка шема базе података

Дефинисање логичке шеме базе података омогућава дефинисање системске документације која се може користити за даљи развој базе података као и апликације намењене крајњим корисницима. Такође, даје и јаснију слику о пословним правилима као и логичку слику базе података која се касније користи приликом генерисања система за управљање базама података – SUBP.

Моделирање података се углавном обрађује током дефинисања ER дијаграма. Пошто су већ у једној од претходних фаза дефинисани ентитети и њихови атрибути, сада се може почети идентификација кандидата за кључеве као и креирање базе података.

Одабрани ентитети су:

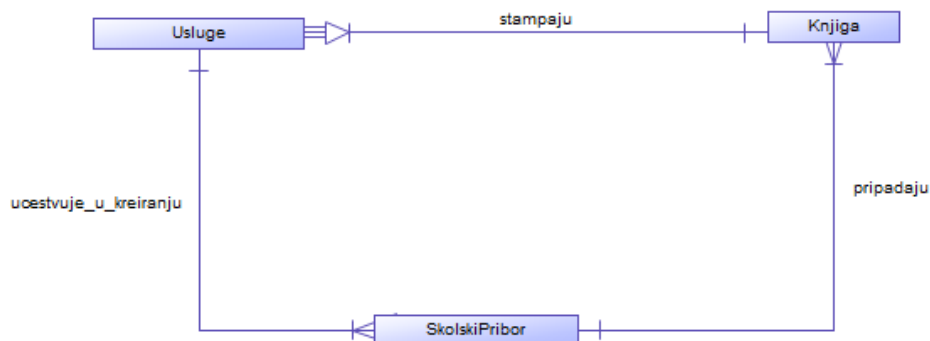
- Knige
- SkolskiPribor
- Usluge

Пошто се одабиром ентитета обезбедило да редувантност података приликом смештања унутар тока података који мирује (складишта) буде минимална, може се прећи и на идентификацију веза између ентитета. Веза представља комуникацију међу објектима. Дефинисањем веза се уједно дефинишу и зависности између ентитета, односно описују се начини узајамног деловања између наведених ентитета.

Идентификација веза је крајње једноставна. Сви ентитети су међусобно повезани и те везе се могу дефинисати на следећи начин:

- „Usluge“ штампају „Knjige“
- „Knjige“ припадају „SkolskiPribor“
- „SkolskiPribor“ учествује у креирању „Usluge“

Прва релација значи да књига настаје штампањем, односно употребом услуга копирања. Друга релација се односи да су књиге део школског прибора који се може купити у скриптарници и у последњој релацији се мисли да се школски прибор користи у копирању, односно папир, хефталица, мастило, штампач...



Слика 7. ER дијаграм система

На Слици 7 је дат ER дијаграм система и он представља резултат свих анализа које су претходно изведене. Уједно је и добар приказ анализе која је урађена помоћу прикупљених докумената и потреба (захтева) корисника. ER моделом дефинисани су оквири модела података. На Слици 8 дат је приказ ER дијаграма са дефинисаним атрибутима за све ентитете.

Ентитети имају следеће атрибуте:

- Knjige:

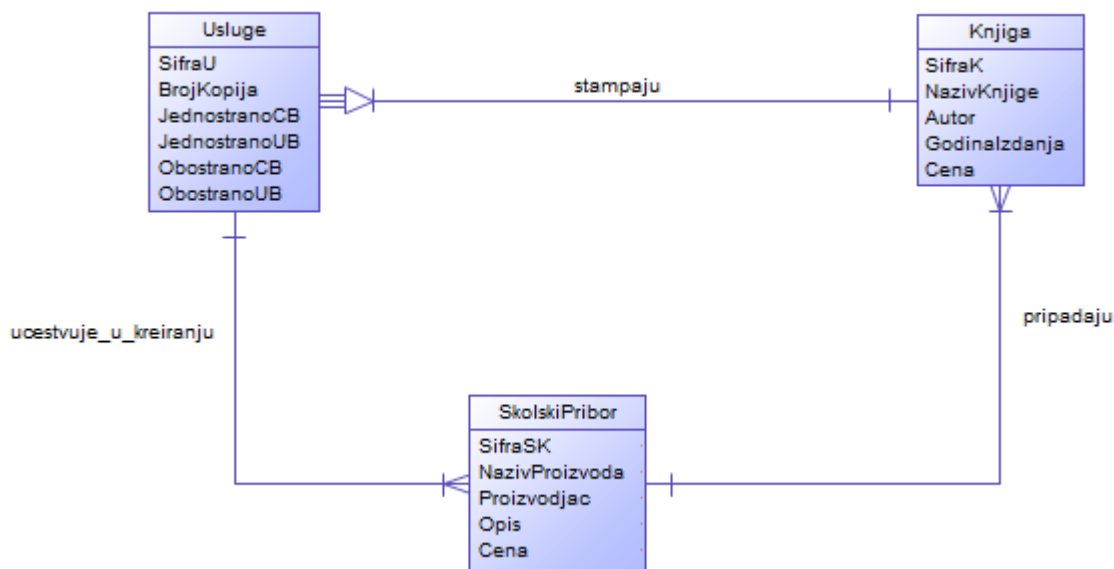
SifraK, NazivKnjige, Autor, GodinaIzdanja, Cena

- SkolskiPribor:

SifraSK, NazivProizvoda, Proizvodjac, Opis, Cena

- Usluge:

SifraU, BrojKopija, JednostranoCB, JednostranoUB, ObostranoCB, ObostranoUB



Слика 7. ER дијаграм система са дефинисаним атрибутима

5. Физичка шема базе података и имплементација са тестним подацима

Завршетком горе наведених активности, уједно је завршен тзв. логички дизајн што уједно представља и почетак физичког дизајна изабраног система за управљање базом података – SUBP. Систем за управљање базама података је у суштини софтверски систем за чување и претраживање података чија је основна улога да се на захтев апликативних програма врши манипулација подацима. Важно је напоменути да је база података (BP) скуп узајамно повезаних података који су меморисани унутар неке меморијске локације. Одмах на почетку, потребно је одредити SUBP који ће се користити. База података је креирана унутар програма „SQLite Studio“ и одмах након креирања може се почети са дефинисањем табела и колона.

Како су ентитети већ дефинисани а подаци у оквиру базе уређени и груписани по тим истим ентитетима, може се почети са смештањем информација (атрибута ентитета). За сваки ентитет се формира посебна табела док се за сваки атрибут ентитета формира посебна колона и ту је важно напоменути да назив сваког атрибута мора бити јединствен (unique) унутар једне табеле, док је могуће имати исте називе атрибута у различитим табелама. Може се почети са креирањем физичког дизајна.

Следећа табела представља приказ ентитета као табела и атрибута као колона.

Ентитети	Табеле	Атрибути	Колоне
Knjige	Knjige	SifraK	SifraK (PK, int), Unique, not null
		NazivKnjige	Naziv (varchar), not null
		Autor	Autor (text), not null
		Godinalzdanja	Godinalzdanja (varchar), not null
		Cena	Cena (double), not null
SkolskiPribor	SkolskiPribor	SifraSK	SifraSK (PK, int), Unique, not null
		NazivProizvoda	NazivProizvoda (varchar), not null
		Proizvodjac	Proizvodjac (varchar), not null
		Opis	Opis (varchar), not null
		Cena	Cena (double), not null
Usluge	Usluge	SifraU	SifraU (PK, int), Unique, not null
		BrojKopija	BrojKopija (varchar), not null
		JednostranoCB	JednostranoCB (double), not null
		JednostranoUB	JednostranoUB (double), not null
		ObostranoCB	ObostranoCB (double), not null
		ObostranoUB	ObostranoUB (double), not null

Табела 2.. Упоредни приказ ентитета – табела и атрибута – колоне

У основи се налази база података која је смештена на серверу преко које корисници могу да приступају подацима ради манипулације (унос и обрада) истим. Битно је напоменути да се база налази на потпуно независној инстанци SQLite платформе која унутар себе садржи сопствену главну базу. Овакво решење нуди висок ниво заштите података као и сигурност приликом приступа тим подацима – приступ је могућ само путем апликације за коју је намењена наведена база. Шема базе података представља физички представљене табеле и колоне као и међусобне релације (везе). Превођењем ентитета у табеле, атрибута у колоне и веза у релације (референцијални интегритет) добија се физички модел шеме базе података коришћењем логичког модела. Такав начин генерисања шеме базе података се назива директним инжењерством (forward engineering).

У наставку је приказан изворни код помоћу којег су креирани табеле, атрибути и релациони односи. То су изворни DDL фајлови SQLite-a.

```
CREATE TABLE Knjige (  
SifraK INTEGER PRIMARY KEY  
UNIQUE  
NOT NULL,  
NazivKnjige VARCHAR NOT NULL,  
Autor TEXT NOT NULL,  
GodinaIzdanja VARCHAR NOT NULL,  
Cena DOUBLE NOT NULL  
);          // изворни код табеле Knjige
```

```
CREATE TABLE SkolskiPribor (  
SifraSK INTEGER PRIMARY KEY  
UNIQUE  
NOT NULL,  
NazivProizvoda VARCHAR NOT NULL,  
Proizvodjac VARCHAR NOT NULL,  
Opis VARCHAR NOT NULL,  
Cena DOUBLE NOT NULL  
);          // изворни код табеле SkolskiPribor
```

```
CREATE TABLE Usluge (  
SifraU INTEGER PRIMARY KEY  
UNIQUE  
NOT NULL,  
BrojKopija VARCHAR NOT NULL,  
JednostranoCB DOUBLE NOT NULL,  
JednostranoUB DOUBLE NOT NULL,  
ObostranoCB DOUBLE NOT NULL,  
ObostranoUB DOUBLE NOT NULL  
);          // изворни код табеле Usluge
```

Такође, одмах је потребно проверити валидност и исправност базе података тако што се имплементира са тестним подацима. Испод су приказане табеле са тест подацима (база података) које се користи приликом развоја концепта информационог система.

На Сликама 8, 9 и 10 приказани су тест подаци за сваку табелу у бази података у програму „SQLite Studio“.

	Sifra	NazivKnjige	Autor	GodinaIzdania	Cena
1	1	Nacrtna geometrija i tehnicko crtanje	Lozica Ivanovic	2018.	356.5
2	5	Kompozitne konstrukcije	Zorica Djordjevic	2018.	593
3	6	Elektrotehnika sa elektronikom	Jasna Radulovic	2018.	720
4	9	Model reinzenjeringa tehnoloskih procesa malih preduzeca	Milan D. Eric	2017.	525.99
5	10	Dinamika vozila	Danijela Miloradovic	2017.	331.25
6	12	Nauka o kvalitetu	Slavko Arsovski	2016.	378.55
7	14	Osnovi održavanja	Petar Todorovic	2016.	479.75
8	35	Osnovi bioinzenjeringa	Nenad Filipovic	2012.	550.65
9	40	Termodinamika	Milorad Bojic	2011.	1200.25
10	42	Naucne metode i tehnicki razvoj	Miroslav Demic	2011.	942.25

Слика 8. Табела Knjige са тестним подацима

	Sifra	NazivProizvoda	Proizvodjac	Opis	Cena
1	1	Deda Mraz Decorative	Pelivan	Hemijska olovka	100
2	2	B Junior	Staedler	Grafitna olovka	32
3	9	HB Junior	Pelivan	Grafitna olovka	32.1
4	12	Deli	Pixi	Atase mapa A4	370
5	17	Tempera 1k	Brilo	Tamno zelena tempera 16 ml	40.5
6	25	Lepcic	Connect	Lepak u stiku 15g	60
7	36	Gliniks	Connect	Glina 1kg	199.99
8	39	MF 550	Memoris	Heftalica	110
9	41	Masina Panda	Deli	Heft masina Deli Panda	170
10	49	Frozen	Pelivan	Sveska A4 formata	169.99

Слика 9. Табела SkolskiPribor са тестним подацима

	Sifra	BrojKopija	JednostranoCB	JednostranoUB	ObostranoCB	ObostranoUB
1	11	1-50	8	25	5	23
2	12	50-100	5	20	3.5	17
3	14	101-500	3.5	19	3	16
4	19	501-2000	3	16	2	14
5	25	2001-5000	2.2	14	1.8	12
6	36	Preko 5000	1.8	12	1.5	10
7	41	Preko 10000	1.5	10	1.3	5

Слика 10. Табела Usluge са тестним подацима

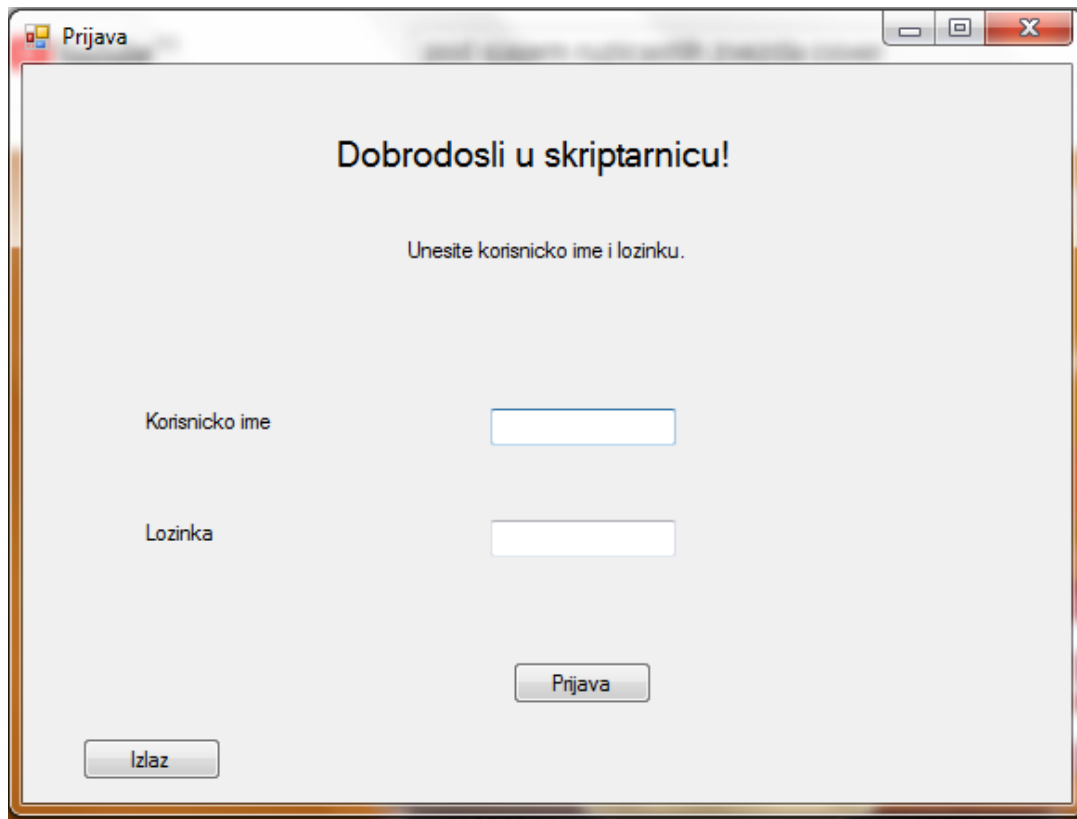
Верификација шеме базе података је последња фаза активности путем које се врши последња провера и верификација физичког модела базе података. Након ове фазе, почиње се са израдом апликације која је намењена корисницима, односно запосленима.

6. Развој апликације (софтвера) реалног система

Када је у потпуности урађена шема базе података, као и дефинисање конкретних захтева будућих корисника, може се почети са израдом саме апликације. Унутар ове подактивности, потребно је дефинисати изглед менија, форме, упита као и извештаја.

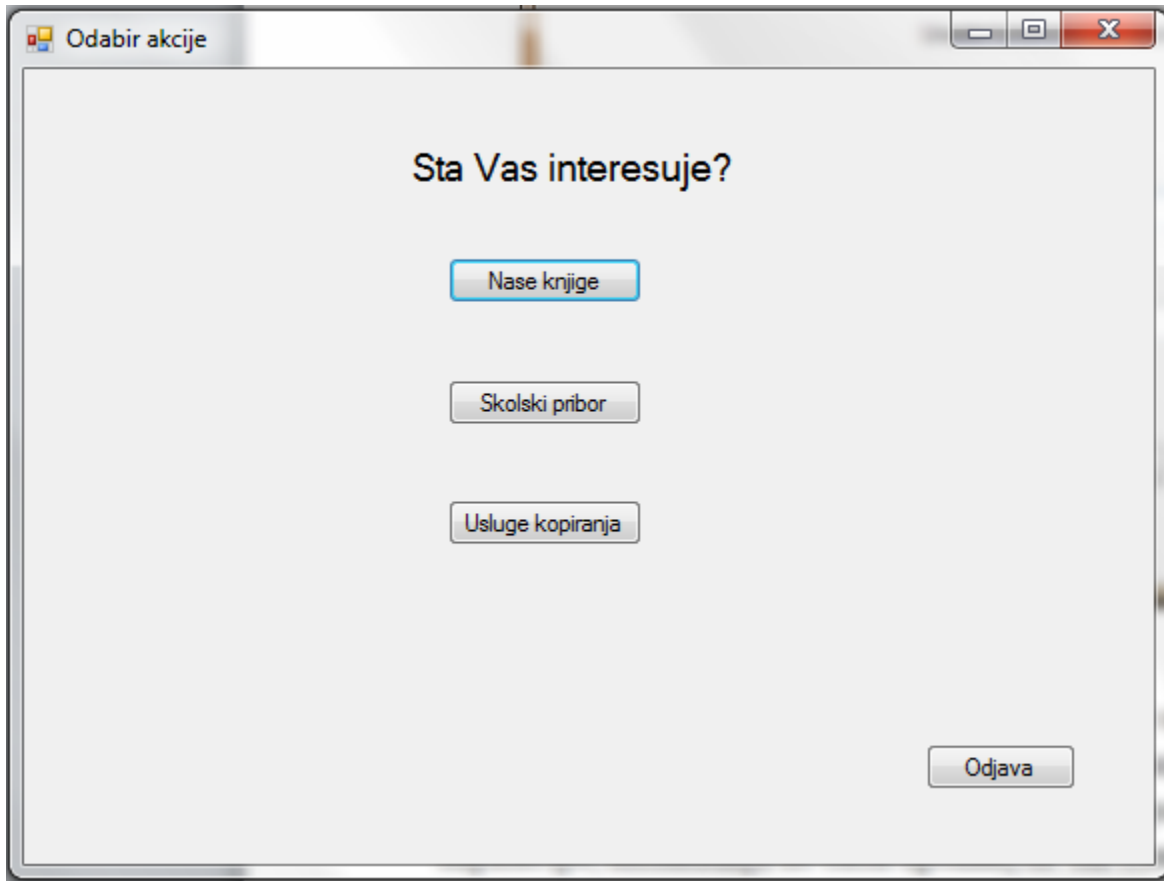
Апликација је реализована помоћу програма: Visual Studio 2017 (креирање апликације и њено програмирање помоћу програмског језика C#) и SQLite Studio (креирање базе података и смештање података). Током дефинисања информационог система, закључак је да наведена апликација треба да садржи: ауторизовани приступ, форму за унос нових података, форму за ажурирање података, форму за брисање података као и форму за преглед свих наведених извештаја.

Изглед почетног прозора апликације је приказан на слици испод, и ту је дат приказ унутар којег је потребно унети приступне параметре.



Слика 11. Почетни прозор апликације

Уколико су унети тачни параметри (корисничко име и лозинка), појавиће се прозор који је одабир акција за даље коришћење апликације. Корисник ће моћи да бира између књига, школског прибора и услуга копирања. Уколико унесе нетачне параметре, апликација ће бити прекинута. На слици испод дат је приказ одабира акција које су на располагању кориснику.

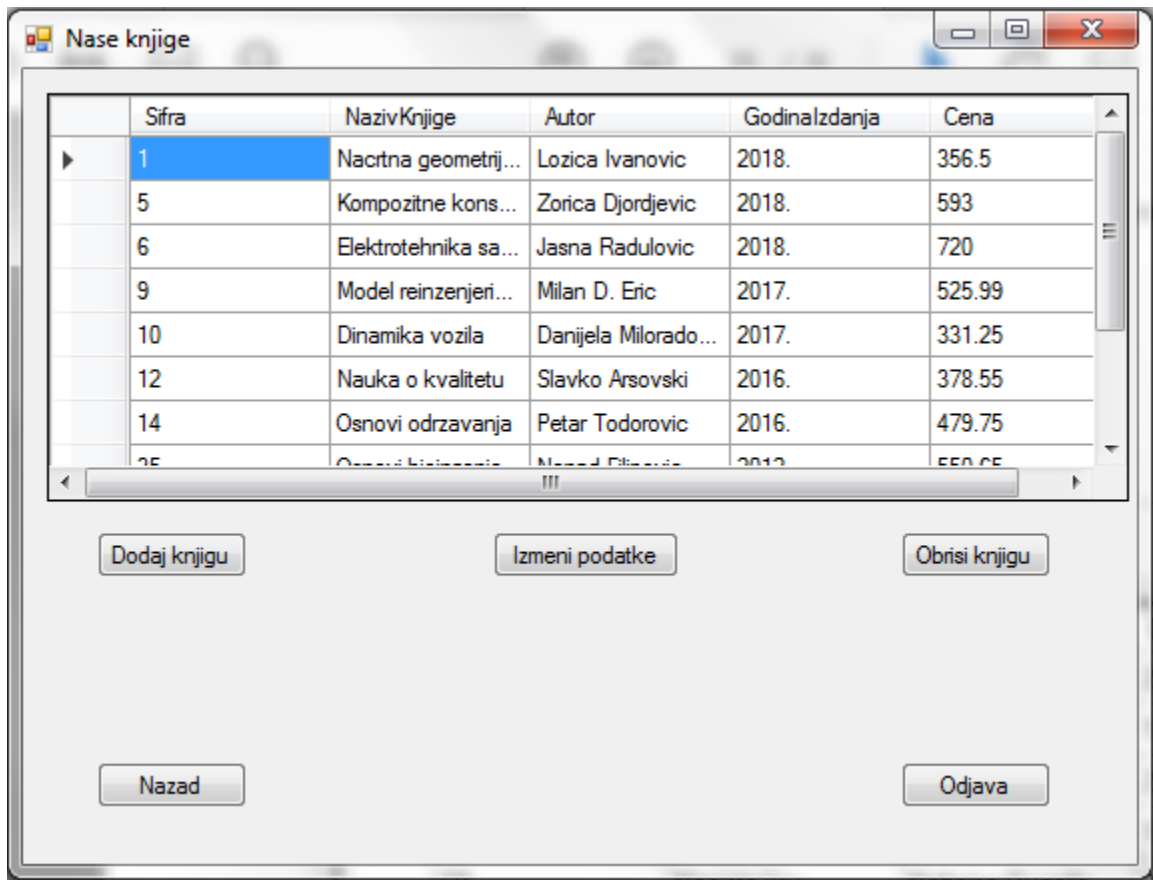


Слика 12. Приказ прозора одабира акције

Постављено је и дугме „Odjava“ којим ће корисник моћи да се врати на почетак апликације, уколико је то потребно.

У даљој документацији, акција „Nase knjige“ ће бити објашњена јер су и остале две акције („Skolski pribor“ и „Usluge kopiranja“) на потпуно исти начин одрађене а и исто функционишу.

Кликом на дугме „Nase knjige“ кориснику ће се појавити прозор као на следећој слици.



Слика 13. Приказ прозора акције „Nase knjige“

Прозор на слици изнад садржи листу књига у скрпитарници. На тај начин корисник има увид у: шифру, назив књиге, име аутора, годину издања и цену сваке књиге која је у бази података.

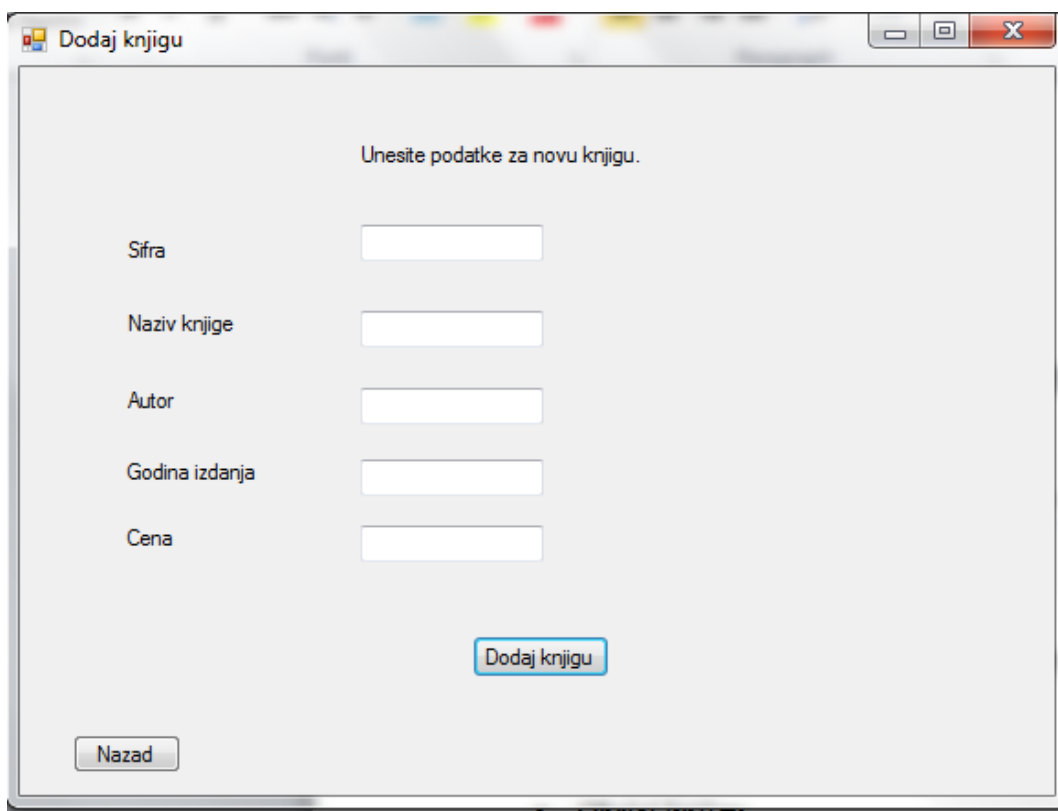
Кликом на дугме „Nazad“, кориснику је омогућено да се врати на прозор где је одабир акција (Слика 12).

Кликом на дугме „Odjava“, корисник се одјављује са система и прелази на приказ почетног прозора апликације (Слика 11).

Кориснику су понуђене три опције, три дугмета:

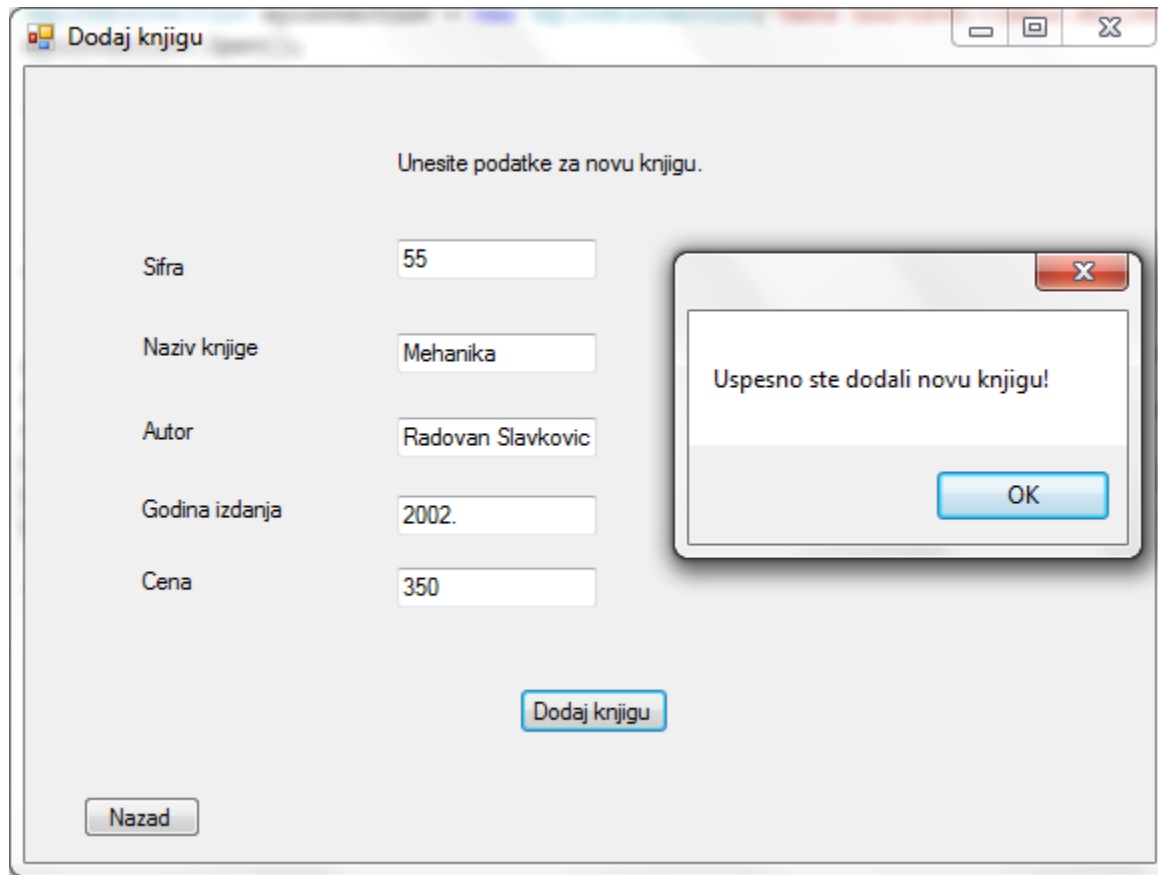
- Dodaj knjigu
- Izmeni podatke
- Obrisi knjigu

Кликом на дугме „Dodaj knjigu“, корисник изражава жељу да постави нову књигу у базу скриптарнице. Изглед ове акције дат је на Слици 14.



Слика 14. Приказ прозора акције „Dodaj knjigu“

Од корисника се очекује да попуни поља: Sifra, Naziv knjige, Autor, Godina izdanja, Cena затим кликне на дугме „Dodaj knjigu“. Ако му се појави прозор као на Слици 15 значи да је успешно завршио додавање књиге.



Слика 15. Приказ додавања нове књиге

Када је исписана порука „Uspesno ste dodali novu knjigu!“, корисник треба притиснути дугме „ОК“ и кликом на дугме „Nazad“ видеће у својој листи да је књига додата.

Уколико корисник одабере дугме „Izmeni podatke“ биће му приказана апликација као на Слици 16.

	Sifra	NazivKnjige	Autor	Godinalzdanja	Cena
▶	1	Nacrtna geometrij...	Lozica Ivanovic	2018.	356.5
	5	Kompozitne kons...	Zorica Djordjevic	2018.	593
	6	Elektrotehnika sa...	Jasna Radulovic	2018.	720
	9	Model reinzenjeri...	Milan D. Eric	2017.	525.99
	10	Dinamika vozila	Danijela Milorado...	2017.	331.25
	12	Nauka o kvalitetu	Slavko Arsovski	2016.	378.55
	14	Osnovi održavanja	Petar Todorovic	2016.	479.75

Unesite sifru knjige koju zelite da izmenite:

Naziv knjige

Autor

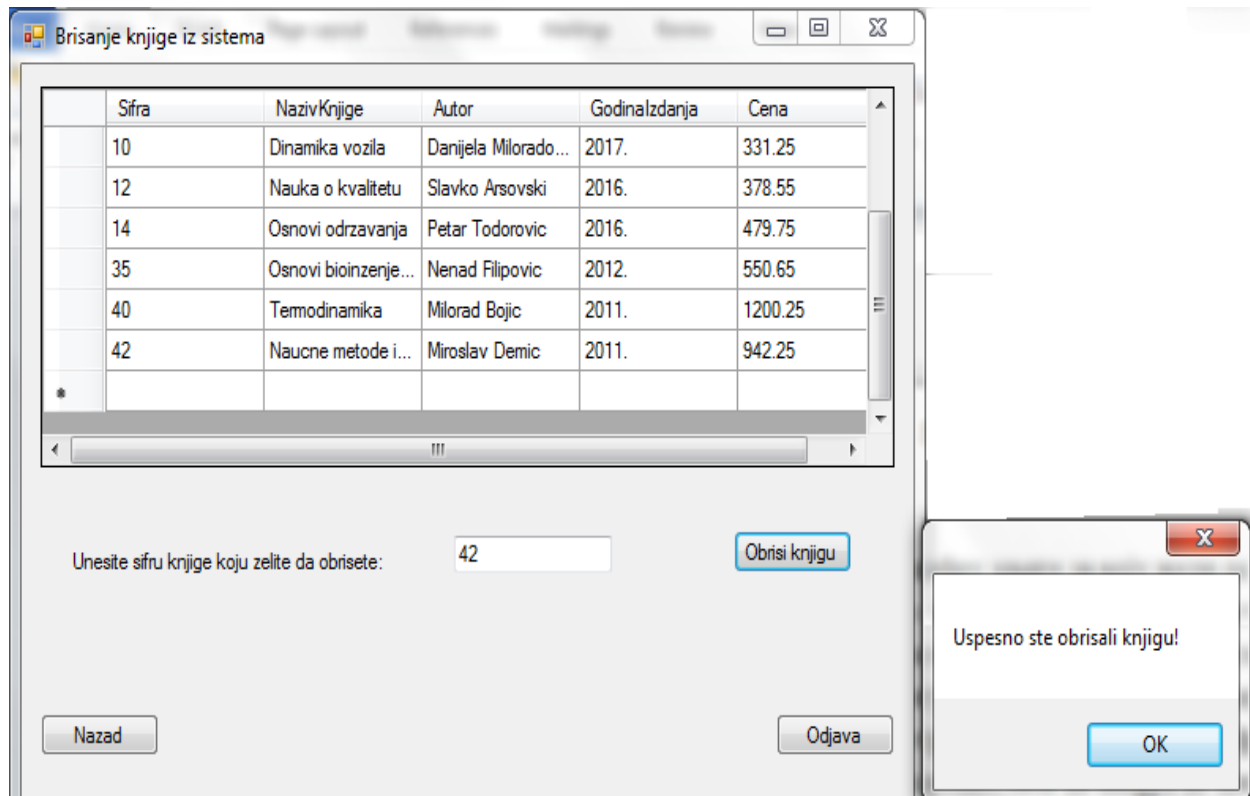
Godina izdanja

Cena

Слика 16. Приказ измене података књига

Корисник треба да унесе прво шифру књиге за коју жели да промени податке а онда да унесе промењени назив књиге, аутора, годину издања и цену и да кликне на дугме „Izmeni podatke“. Кликом на дугме „Nazad“, корисник може да види своје ажуриране податке. Уколико нису унета сва поља, апликација ће упозорити поруком да се унесу сви параметри. Ако корисник зада шифру књиге која не постоји у бази, апликација ће га обавестити да шифра не постоји.

Ако корисник одабере „Obrisi knjigu“ отвориће му се нови прозор где треба унети шифру књиге коју жели да обрише из базе скриптарнице. То је приказано на Слици 17.



Слика 17. Приказ брисања података књига

Кликом на дугме „Nazad“, корисник може видети да је књига обрисана из базе података. Уколико корисник унесе шифру која не постоји, апликација ће га подсетити да је шифра непостојана у систему.

7. Литература

1. Вељовић А.: Пројектовање информационих система, Компјутер библиотека, 2003.
2. <http://www.antikafotokopirnice.rs/cenovnik/>
3. http://mfkg.rs/index.php?option=com_content&view=category&id=42&Itemid=257&limitstart=30
4. https://www.cicakshop.rs/kategorija-proizvoda/sve-za-skolu/?gclid=EAIaIQobChMIw-ayr7-84gIVwo2yCh2_IQWSEAAAYAAAEgK_j_D_BwE