|  |  |  |
| --- | --- | --- |
|  | УНИВЕРЗИТЕТ У НОВОМ САДУ  **ФАКУЛТЕТ ТЕХНИЧКИХ НАУКА У НОВОМ САДУ** |  |

Име и презиме Немања Шобо

**НАСЛОВ РАДА**

ДИПЛОМСКИ РАД

- Основне академске студије -

Нови Сад, година

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | УНИВЕРЗИТЕТ У НОВОМ САДУ ⚫ **ФАКУЛТЕТ ТЕХНИЧКИХ НАУКА**  21000 НОВИ САД, Трг Доситеја Обрадовића 6 | | | |
| **КЉУЧНА ДОКУМЕНТАЦИЈСКА ИНФОРМАЦИЈА** | | | |
| Редни број, **РБР**: | | |  | |
| Идентификациони број, **ИБР**: | | |  | |
| Тип документације, **ТД**: | | | Монографска публикација | |
| Тип записа, **ТЗ**: | | | Текстуални штампани документ/ ЦД | |
| Врста рада, **ВР**: | | | Дипломски рад | |
| Аутор, **АУ**: | | | име презиме | |
| Ментор, **МН**: | | | др име презиме, звање | |
| Наслов рада, **НР**: | | | наслов | |
| Језик публикације, **ЈП**: | | | Српски (писмо – ћирилица или латиница) | |
| Језик извода, **ЈИ**: | | | Српски/енглески | |
| Земља публиковања, **ЗП**: | | | Србија | |
| Уже географско подручје, **УГП**: | | | Војводина | |
| Година, **ГО**: | | | година | |
| Издавач, **ИЗ**: | | | Ауторски репринт | |
| Место и адреса, **МА**: | | | Факултет техничких наука (ФТН), Д. Обрадовића 6, 21000 Нови Сад | |
| Физички опис рада, **ФО**: (поглавља/страна/ цитата/табела/слика/графика/прилога) | | | ?/?/?/?/?/?/? | |
| Научна област, **НО**: | | | Електротехничко и рачунарско инжењерство | |
| Научна дисциплина, **НД**: | | | Примењене рачунарске науке и информатика | |
| Предметна одредница/Кључне речи, **ПО**: | | | кључне речи | |
| **УДК** | | |  | |
| Чува се, **ЧУ**: | | | Библиотека ФТН, Д. Обрадовића 6, 21000 Нови Сад | |
| Важна напомена, **ВН**: | | |  | |
| Извод, **ИЗ**: | | | сажетак | |
| Датум прихватања теме, **ДП**: | | |  | |
| Датум одбране, **ДО**: | | |  | |
| Чланови комисије, **КО**: | | Председник: | др име презиме, звање |
|  | | Члан: | др име презиме, звање | Потпис ментора |
|  | | Члан, ментор: | др име презиме, звање |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | UNIVERSITY OF NOVI SAD ⚫ **FACULTY OF TECHNICAL SCIENCES**  21000 NOVI SAD, Trg Dositeja Obradovića 6 | | | |
| **KEY WORDS DOCUMENTATION** | | | |
| Accession number, **ANO**: | | |  | |
| Identification number, **INO**: | | |  | |
| Document type, **DT**: | | | Monographic publication | |
| Type of record, **TR**: | | | Textual material, printed/CD | |
| Contents code, **CC**: | | | Bachelor thesis | |
| Author, **AU**: | | | ime prezime | |
| Mentor, **MN**: | | | dr ime prezime, zvanje na engleskom | |
| Title, **TI**: | | | naslov na engleskom | |
| Language of text, **LT**: | | | Serbian | |
| Language of abstract, **LA**: | | | Serbian/English | |
| Country of publication, **CP**: | | | Serbia | |
| Locality of publication, **LP**: | | | Vojvodina | |
| Publication year, **PY**: | | | godina | |
| Publisher, **PB**: | | | Author’s reprint | |
| Publication place, **PP**: | | | Faculty of Technical Sciences, D. Obradovića 6, 21000 Novi Sad | |
| Physical description, **PD**: (chapters/pages/ref./tables/pictures/graphs/appendixes) | | | ?/?/?/?/?/?/? | |
| Scientific field, **SF**: | | | Electrical and computer engineering | |
| Scientific discipline, **SD**: | | | Applied computer science and informatics | |
| Subject/Key words, **S**/**KW**: | | | ključne reči na engleskom | |
| **UC** | | |  | |
| Holding data, **HD**: | | | Library of Faculty of Technical Sciences, D. Obradovića 6, 21000 Novi Sad | |
| Note, **N**: | | |  | |
| Abstract, **AB**: | | | sažetak na engleskom | |
| Accepted by the Scientific Board on, **ASB**: | | |  | |
| Defended on, **DE**: | | |  | |
| Defended Board, **DB**: | | President: | dr ime prezime, zvanje na engleskom |
|  | | Member: | dr ime prezime, zvanje na engleskom | Menthor's sign |
|  | | Member, Mentor: | dr ime prezime, zvanje na engleskom |  |

|  |  |  |
| --- | --- | --- |
|  | УНИВЕРЗИТЕТ У НОВОМ САДУ ⚫ **ФАКУЛТЕТ ТЕХНИЧКИХ НАУКА**  21000 НОВИ САД, Трг Доситеја Обрадовића 6 | Датум: |
|  |
| **ЗАДАТАК ЗА ИЗРАДУ ДИПЛОМСКОГ (BACHELOR) РАДА** | Лист/Листова: |
| 1/1 |

*(Податке уноси предметни наставник - ментор)*

| Врста студија: | Основне академске студије  Основне струковне студије |
| --- | --- |
| Студијски програм: |  |
| Руководилац студијског програма: |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Студент: |  | Број индекса: |  |
| Област: |  | | |
| Ментор: |  | | |
| НА ОСНОВУ ПОДНЕТЕ ПРИЈАВЕ, ПРИЛОЖЕНЕ ДОКУМЕНТАЦИЈЕ И ОДРЕДБИ СТАТУТА ФАКУЛТЕТА  ИЗДАЈЕ СЕ ЗАДАТАК ЗА ДИПЛОМСКИ (Bachelor) РАД, СА СЛЕДЕЋИМ ЕЛЕМЕНТИМА:   * проблем – тема рада; * начин решавања проблема и начин практичне провере резултата рада, ако је таква провера неопходна; * литература | | | |

**НАСЛОВ ДИПЛОМСКОГ (BACHELOR) РАДА:**

|  |
| --- |
|  |

**ТЕКСТ ЗАДАТКА:**

|  |
| --- |
|  |

|  |  |
| --- | --- |
| Руководилац студијског програма: | Ментор рада: |
|  |  |

|  |
| --- |
| Примерак за:  - Студента;  - Ментора |

Образац **Q2.НА.15-04** - Издање 1

**SPISAK KORIŠĆENIH SKRAĆENICA**

Skraćenica Puni naziv

(spisak sortirati po skraćenicama)

**SADRŽAJ**

**1. UVOD**

**2. OPIS REŠAVANOG PROBLEMA**

**3. OPIS KORIŠĆENIH TEHNOLOGIJA I ALATA**

**4. OPIS REŠENJA PROBLEMA**

# **5. Zaključak**

**6. LITERATURA**

**PRILOG 1**

**PODACI O KANDIDATU**

(nazivi poglavlja u sadržaju su okvirni; UVOD i ZAKLJUČAK pisati na kraju, nakon završetka preostalih poglavlja)

**1. UVOD**

Navesti kratak pregled rada.

Navesti kratak prikaz ostalih poglavlja rada.

Orijentaciona dužina uvoda je oko jedne stranice teksta.

Navesti detaljan opis problema rešavanog u radu. Pozvati se na relevantnu literaturu i prodiskutovati već postojeća rešenja razmatranog problema.

Ispod svake tabele (slike ili listinga) navesti broj i naziv, kao za Tabelu 2.1.

|  |  |  |
| --- | --- | --- |
|  | kolona 1 | kolona 2 |
| red 1 | element 1,1 | element 1,2 |

Tabela 2.1 Primer tabele

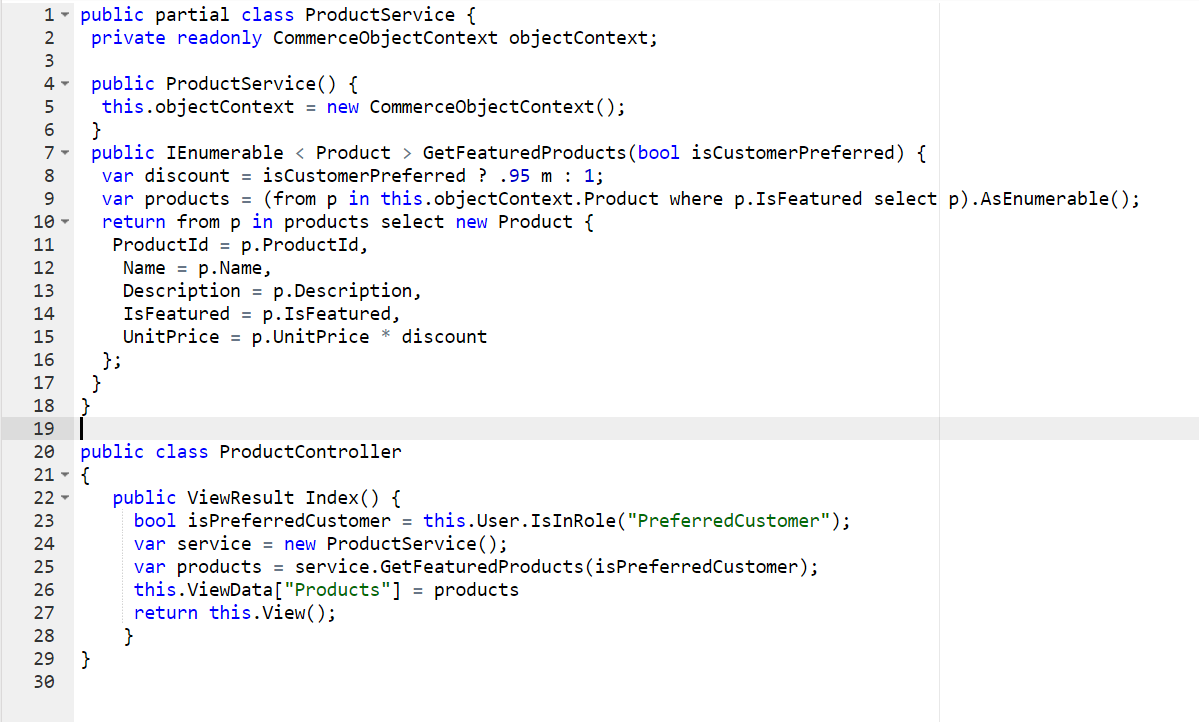
U tekstu obavezno prodiskutovati svaku tabelu (sliku ili listing).

Korišćena literatura se citira navođenjem rednog broja reference između srednjih zagrada, na primer [1]. U tekstu obavezno citirati svaku referencu iz literature.

**2. OPIS REŠAVANOG PROBLEMA**

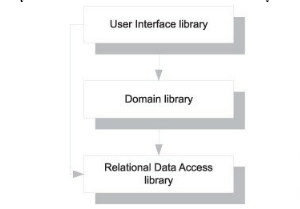
Živimo u vremenu kada posao od ljudi zahteva čestu promenu i konstantan razvoj i proširenje postojećih softverskih rešenja. Veoma često klijenti posle prvog puštanja nekog softverskog proizvoda u produkciju dolaze do ryličitih zaključaka i promene mišljenja kako bi unapredili svoje rešenje kao i ispoštovali zahteve korisnika koji usled korištenja aplikacije očekuju promene koje će im korištenje iste učiniti lakšim i pristupačnijim. Takođe se dešava da klijenti menjaju mišljenje o servisima koji vrše određeni deo posla na različitim slojevima aplikacije kako bi optimizovali njen rad ali i zamenili zastrele servise novim koji određeni deo posla rade mnogo bolje i njihovo održavanje je mnogo lakše i isplativije. Obzirom na gore navedeno na nama je da sve to učinimo što lakšim kako bi smo rešenju omogućili konstantan napredak i održivost na visokom nivou.

Projektovanjem i kreiranjem softvera koji je interno čvrsto spregnut činimo održivost i proširivost velikih projekata skoro nemogućim i samim tim onemogućujemo njegov dalji razvoj i unapređenje. Čvrsta spregnutost koda ili Tight Coupling je uobičajeno loša praksa pisanja a njene posledice su slaba fleksibilnost, gotovo nemoguća ponovna iskorsitivost, dodavanje određenih izmena je veoma problematično i testiranje aplikacije je veoma otežano. Tightly Coupled komponente su one komponente koje mnogo znaju jedna o drugoj, odnosno o samoj implementaciji i toliko su spregnute da izmena jedne nužno znači i izmenu druge.



Slika 2.1 Primer čvrsto spregnutog koda

Na slici 2.1 vidimo primer čvrsto spregnutih slojeva aplikacije. Čvrsta spregnutost se najviše ogleda u dirketnom instanciranju komponenti iz drugih slojeva( u ovom slučaju library-a) aplikacije i samim tim potpuno onemogućujemo izlociju ovih modula tako da mogu da rade zasebno bez oslanjanja jedan na drugog. Glavna prednost višeslojnih aplikacija je u tome što nam omogućava totalno razdvajanje poslova za koji je svaki sloj zadužen, kao što je na primer dobavljanje podataka iz baze, biznis logika, prikaz podataka itd. Povezivanje slojeva na ovaj način spregnuli smo aplikaciju tako da naše komponente ne mogu da egzistiraju jedna bez druge i samim tim narušili koncept slabo spregnutih rešenja.



Slika 2.2. Graf zavisosti između slojeva aplikacije

Slika 2.2. nam prikazihe zavisnost između slojeva (Library) aplikacije oslikavajući kod sa slike 2.1. User Interface library sloj u kom se nalazi ProductController klasa čiji je zadatak da podatke koje dobija iz servisa salje na komponente koje se bave prikazom, referencira Domain library odnosno servisni sloj u kome se nalazi ProductService klasa a zbog svoje visoke spregnutosti sa njom mora da referencira i RelationalDataAccess library u kojoj se nalazi kontekstna klasa odnosno CommerceObjectContext. U ovom slučaju jedino sloj koji je zmenjiv je UserInterface koji bi smo mogli zameniti sa bilo kojim drugim slojem z aprikaz podataka tako što bi smo u njega opet referencirali i servisni i sloj koji je zadužen za pristum bazi podataka(Domain i Relational Data Access library).

Jedna od mnogih definicija : Dependency Injection(onaj font koji je rekao) je set softverskih dizajn principa i paterna koji nam omogućavaju da pisemo slabo spegnut programski kod(loosely coupled).

Slabo spregnut kod nam omogućava visok nivo proširivosti , što nam samim tim na dalje omogućava

lako održavanje.

Kompozicija objekata(OBJECT COMPOSITION), presretanje(INTERCEPTION) i upravljane životnim ciklusom(LIFTIME MANAGAMENT) su tri dimeyije DI.

2.1 Kompozicija objekata(Object composition)

2.2 Presretanje(Interception)

2.3 Upravljanje životnim ciklusom (Liftime managament)

[1 John Munsch et al., “How to explain Dependency Injection to a 5-year old,” 2009, http://stackoverflow.com/questions/1638919/how-to-explain-dependency-injectionto-a-5-year-old](http://stackoverflow.com/questions/1638919/how-to-explain-dependency-injection-to-a-5-year-old)

*When you go and get things out of the refrigerator for yourself, you can cause problems. You might leave the door open, you might get something Mommy or Daddy doesn’t want you to have. You might even be looking for something we don’t even have or which has expired.*

*What you should be doing is stating a need, “I need something to drink with lunch,” and then we will make sure you*

*have something when you sit down to eat.*

paterna

Ovde pisati jos malo o loosy couple kodu, predstaviti ga kao uzrok problema bla bla bla.

Prednosti slabo spregnutog koda :

1. Servisi su lako zamenjivi drugim servisima. Vrlo značajno kad je u pitanju standardan softver prilikom izmene zahteva i načina implementacije.
2. Kod je proširiv kao i ponovo iskoristiv u situacijama koje nisu bile planirane niti predviđene da bi do njih moglo doći.
3. Paralelni razvoj rešenja odnosno aplikacije što je veoma bitno kod velikih i kompleksnih aplikacija. Kod malih aplikacija ova prednost ne dolazi do izražaja.
4. Veoma održiv kod, klase kojima je jasno definisan njihova odgovornost su lake za održavanje.
5. Takođe u slučaju korištenja unite testova pisanje slabo spregnutnog koda je tehnika koja testiranje čini lakšim.

Pisanje slabo spregnutog koda nam omogućava da pisemo kod koji je otvoren za proširivanje ali ne i za modfikaciju. To je takozvani open-closed princip.

paterna

**3. OPIS KORIŠĆENIH TEHNOLOGIJA I ALATA**

Samo napraviti pregled raspoloživih tehnologija i alata, navesti razloge izbora tehnologije i alata, kao i njihova bitna svojstva i prednosti**.**

# **4. Opis rešenja problema**

# **5. Zaključak**

Osvrt na rešenje problema, uz ukazivanje na njegove dobre i eventualno loše strane, kao i na moguće pravce daljeg istraživanja. Osvrt na rešenje pisati u prošlom vremenu.

Orijentaciona dužina zaključka je oko jedne stranice teksta.

**6. LITERATURA**

[1] ***Cleaveland, J.C., Building Application Generators, 1988, Software, IEEE, Jul 1988, Vol 5, Is 4***

[2] ***M. E. Lesk and E. Schmidt, Lex - A Lexical Analyzer Generator, 2005, www.compilertools.net***

[3] ***Wikipedia, Regular expression, 2006, Wikipedia***

[4] ***ISO/IEC, EBNF - ISO 14977 standard, 1996, ISO/IEC***

[5] ***Free Software Foundation, Inc., Bison manual, 2005, http://www.gnu.org/software/bison/bison.html***

[6] ***A.V. Aho, R. Sethi, and J.D. Ullman, Compiler: Principles, Techniques and Tools, 1986, Addison-Wesley***

[7] [***www.w3.org***](http://www.w3.org), spisak ključnih reči prezentacije kako bi bila dostupna i u slučaju promene URL

**PRILOG 1**

**PODACI O KANDIDATU**

Kandidat ime i prezime je rođen(a) ?. godine u mestu. Završio(la) je srednju vrsta školu naziv u mestu ?. godine. Fakultet tehničih nauka u Novom Sadu je upisao(la) ?. godine. Ispunio(la) je sve obaveze i položio(la) je sve ispite predviđene studijskim programom.