

НАСТАВНО-НАУЧНОМ ВЕЋУ

Предмет: Подобност теме и кандидата Милоша Милића за израду докторске дисертације

Одлуком наставно-научног већа ФОН-а бр. 33-11 од 11.03.2015. године именовани смо за чланове Комисије за оцену подобности теме и кандидата Милоша Милића за израду докторске дисертације и научне заснованости теме „Побољшање објектно-оријентисаних софтверских система коришћењем стандарда квалитета софтвера“

На основу материјала приложеног уз Захтев кандидата, Комисија подноси следећи

РЕФЕРАТ

1. Подаци о кандидату

1.1. Биографски подаци

Милош Милић је рођен 26. марта 1983. године у Пожаревцу, Република Србија, где је завршио основну и средњу техничку школу. Факултет организационих наука Универзитета у Београду уписао је 2002. године. Дипломирао је на Факултету организационих наука Универзитета у Београду (смер Информациони системи и технологије) 2006. године са просечном оценом 9.47 током студија. Награђиван је од стране Факултета организационих наука и Министарства просвете и науке за постигнуте резултате у току основних студија. Дипломске академске (мастер) студије је завршио је 2007. године на Факултету организационих наука, студијски програм Информациони системи и технологије, програмско подручје (модул) Софтверско инжењерство, са постигнутом просечном оценом 10 током студија.

Од 2004. године до 2008. године ради као лаборант на Факултету организационих наука на предметима који се баве информационим системима и софтверским инжењерством. Након завршетка мастер студија запослио се на Факултету организационих наука најпре као стручни сарадник, а затим као асистент, и ангажован на извођењу наставе на следећим предметима основних студија:

- Пројектовање софтвера,
- Софтверски патерни,
- Напредне Јава технологије,
- Програмирање 1,
- Основе програмирања,
- Софтверски процес и еволуција софтвера,
- Основе информационо-комуникационих технологија.

Поред тога, ангажован је и на извођењу наставе на следећим предметима мастер академских студија:

- Напредне софтверске технологије,
- Софтверски захтеви,
- Софтверски процес и
- Напредне софтверске технологије 2.

У анонимним анкетама студената - за оцену квалитета рада наставника и сарадника, оцењен је високим просечним оценама, преко 4.7 (на скали од 1 до 5).

На Факултету организационих наука је у последњих неколико година држао неколико основних и виших курсева Јаве који су високо оцењени од стране полазника тих курсева. Завршио је курсеве Java IA-32 и Java IA-34 на Електротехничком факултету Универзитета у Београду, CSM (Certified Scrum Master) курс, курс NetBeans platform training, Летњу школу доменски специфичних језика (DSL Summer School). На Факултету организационих наука учествовао је у организацији CSM (Certified Scrum Master) курса 2009. године, као и у организацији Летње школе доменски специфичних језика 2013. и 2014. године. Задњих година се изузетно истакао у раду на међународном пројекту КОСТМОД у развоју софтверског система, који је рађен на Факултету организационих наука у оквиру Лабораторије за софтверско инжењерство. Такође, учествовао је и на пројекту анализе и унапређења пословних процеса у области „Листе чекања“ и доступности јавних података здравствених установа и њиховог повезивања, као и изради софтверских апликација за подршку доступности јавних података са сајтова здравствених установа и посебно повезивања сајтова апотекарских установа, који су рађени за потребе Републичког фонда за здравствено осигурање (РФЗО). Запослен је на Факултету организационих наука.

1.2. Стечено научноистраживачко искуство

Милош Милић је у октобру 2009. године уписао докторске академске студије на Факултету организационих наука, на студијском програму Информациони системи и менаџмент, изборно подручје – Софтверско инжењерство. Положио је све испите са просечном оценом 10 (десет):

1. Пројектовање софтвера – одабрана поглавља
2. Управљање софтверским пројектима – одабрана поглавља
3. Квалитет софтвера – одабрана поглавља
4. Софтверски процес и одржавање софтвера – одабрана поглавља
5. Методологија научно – истраживачког рада
6. Интеракција човека и рачунара - одабрана поглавља
7. Прикупљање захтева – одабрана поглавља
8. Рачунарске мреже – одабрана поглавља
9. Конструкција софтвера – одабрана поглавља

Милош Милић је у мају 2015. године одбранио приступни рад под називом "Побољшање процеса учења објектно-оријентисаних софтверских технологија коришћењем стандарда квалитета софтвера".

Милош Милић је објавио 16 радова и један уџбеник. Научни рад огледа се кроз следећу библиографију радова:

1. Наставни материјали - уџбеник
 - 1.1. Влајић, С., Савић, Д., Антовић, И., Станојевић, В., **Милић, М.**: *Пројектовање софтвера - Напредне Јава технологије*, Златни пресек, Београд, 2008. (ISBN: 978-86-86887-03-0)
2. Радови објављени у научним часописима међународног значаја (M23)
 - 2.1. Sekulić, G., Antović, I., Vlajić, S., Lazarević, S., Savić, D., Stanojević, V., **Milić, M.**: *Conceptual Model of Software Architecture in Instruction Java Web Frameworks*, International Journal of Engineering Education, Vol 31, No 1, 2015, pp. 127-140, ISSN: 0949-149X. IF(2015) = 0.559
 - 2.2. Antović, I., Vlajić, S., **Milić, M.**, Savić, D., Stanojević, V.: *Model and software tool for automatic generation of user interface based on use case and data model*, - IET Software, Vol 6, No 6, pp. 559-573, ISSN: 1751-8806, IF(2012) = 0.658
 - 2.3. **Milić, M.**, Vlajić, S., Antović, I., Savić, D., Stanojević, V., Lazarević, S., *Software Quality Standards and Lean Approach in Teaching and Learning Programming*, International Journal of Engineering Education, ISSN: 0949-149X, IF(2015) = 0.559 (рад је прихваћен за штампу)
3. Радови објављени у часописима националног значаја (M50)
 - 3.1. **Milić, M.**, Antović, I., Savić, D., Stanojević, V., Vlajić, S.: *Povećanje transparentnosti podataka od javnog značaja i unapređivanje zdravstvenog sistema Srbije korišćenjem aplikacije "Pronađi lek"*, - INFO M, Vol 14, No 51, str. 26-32, ISSN: 1451-4397
 - 3.2. Ristin, N., Vlajić, S., Antović, I., **Milić, M.**, Stanojević, V.: *Komparativna analiza Java i .NET web servisa*, - INFO M, Vol 10, No 40, str. 29-37, ISSN: 1451-4397
4. Радови саопштени на скупу међународног значаја (M33)
 - 4.1. Stanojević V., Lazarević S., **Milić M.**: *Comparative Analysis of UML Modeling Tools with Focus on Business Logic Specification*, SYMORG 2016, Zlatibor, pp. 1272-1280, ISBN: 978-86-7680-326-2
 - 4.2. **Milić, M.**, Vlajić, S., Lazarević, S.: *Improving Serbian healthcare system with "Find a medicine"*, - Proceedings of XIV International Symposium SYMORG 2014, Zlatibor 2014., pp. 985-992. ISBN: 978-86-7680-295-1
 - 4.3. Savić, D., da Silva A. R., Vlajić, S., Lazarević, S., Stanojević, V., Antović, I., **Milić, M.**: *Preliminary experience using JetBrains MPS to implement a requirements specification language*, - Proceedings of IX International Conference on the Quality of Information and Communications Technology QUATIC 2014, pp.134-137, ISBN: 978-1-4799-6133-7
 - 4.4. Savić, D., da Silva A. R., Vlajić, S., Lazarević, S., Stanojević, V., Antović, I., **Milić, M.**: *Use Case Specification at Different Levels of Abstraction*, - Proceedings of VIII International Conference on the Quality of Information and Communications Technology QUATIC 2012, pp.187-192, ISBN: 978-1-4673-2345-1
 - 4.5. **Milić, M.**, Antović, I., Vlajić, S.: *Comparative analysis of Spring framework and EJB technology*, - Proceedings of XIII International Symposium SYMORG 2012, Zlatibor 2012, pp. 994-999. ISBN: 978-86-7680-255-5
 - 4.6. Stanojević, V., Vlajić, S., **Milić, M.**, Ognjanović, M.: *Guidelines for framework Development process*, - Proceedings of VII Central and Eastern European Software Engineering Conference in Russia CEE-SECR 2011, pp. 1-9, ISBN: 978-1-4673-0843-4

- 4.7. Savić, D., Antović, I., Vlajić, S., Stanojević, V., **Milić, M.**: *Language for Use Case Specification*, - XXXIV Software Engineering Workshop SEW 2011, pp. 19-26, ISBN: 978-1-4673-0245-6
- 4.8. Puric, A., **Milić, M.**, Antović, I., Stanojević, V., Savić, D.: *A contribution to defining and determining software quality*, - International Conference on Education and New Learning Technologies EDULEARN10, pp. 2188-2194, ISBN: 978-84-613-9386-2
5. Радови саопштени на скупу националног значаја (M63)
 - 5.1. Nikolić, A., Antović, I., Vlajić, S., **Milić, M.**, Savić, D., Lazarević, S.: *Komparativna analiza Hibernate i EJB tehnologije*, INFOFEST 2011, Budva 2011, str. 108-119, ISBN:
 - 5.2. Antović, I., Savić, D., Stanojević, V., **Milić, M.**, Vlajić, S.: *Alati i metode softverskog inženjerstva po swebok projektu*, Zbornik radova YU INFO 2008, Kopaonik 2008., ISBN: 978-86-85525-03-2
 - 5.3. **Milić, M.**, Vlajić, S., Antović, I., Stanojević, V., Savić, D.: *Softverske metrike kao tehnika za evaluaciju i poboljšanje kvaliteta softvera*, Zbornik radova XI Simpozijum SymOrg 2008, Beograd, 2008., str. 336-343, ISBN: 978-86-7680-161-9

1.3. Оцена подобности кандидата за рад на предложеној теми

Кандидат Милош Милић је завршио основне и мастер студије на Факултету организационих наука Универзитета у Београду. На докторским студијама Факултета организационих наука положио је свих девет испита предвиђених наставним планом и програмом студија и одбранио приступни рад на докторским студијама, чиме је остварио 120 ЕСПБ бодова. Од завршетка мастер студија запослен је на Факултету организационих наука у Београду, где је ангажован на предметима основних студија: Пројектовање софтвера, Софтверски патерни, Напредне Јава технологије, Програмирање 1, Основе програмирања, Софтверски процес и еволуција софтвера, као и на предметима мастер академских студија: Напредне софтверске технологије, Софтверски захтеви, Софтверски процес и Напредне софтверске технологије 2.

У току свог научно-истраживачког рада кандидат је објавио више радова који су непосредно повезани са темом приступног рада и доктората, као и уџбеник који се користи у процесу наставе на предметима Пројектовање софтвера, Напредне Јава технологије и Напредне софтверске технологије.

У току докторских студија стекао је различита знања и искуства из области пројектовања софтвера, квалитета софтвера, софтверског процеса и одржавања софтвера, управљања софтверским пројектима, интеракције човека и рачунара, прикупљања захтева, конструкције софтвера и методологије научно-истраживачког рада, која су неопходна за израду дефинисане теме.

На основу свега наведеног, сматрамо да кандидат Милош Милић поседује потребно знање и искуство за израду докторске дисертације на тему „**Побољшање објектно-оријентисаних софтверских система коришћењем стандарда квалитета софтвера**“.

2. Предмет и циљ истраживања

Предмет истраживања докторске дисертације биће развој SilabQOSS (енг. Silab Quality Method for Object Oriented Systems) методе за побољшање објектно-оријентисаних софтверских система коришћењем стандарда квалитета софтвера.

Софтверски системи су нашли широку примену у данашњем пословном окружењу. У претходним деценијама је порасла потреба за квалитетом софтвера. Стога се може рећи да осигурање квалитета софтверског производа и процеса његовог развоја представља неминовност и потребу сваке софтверске компаније. Мерењем различитих

карактеристика посматраног софтверског производа и процеса његовог развоја могу се предузети акције које би требале да доведу до повећања нивоа квалитета и поузданости софтвера. Да би се осигурао одговарајући квалитет потребно је извршити правилну спецификацију и евалуацију квалитета софтвера. Ово се може остварити дефинисањем одговарајућих атрибута квалитета софтвера, узимајући у обзир сврху примене софтвера. У том смислу се атрибути квалитета могу користити за дефинисање нивоа квалитета софтверског производа, али и за дефинисање квалитета процеса развоја софтвера. Због тога ће у докторској дисертацији бити дат преглед стандарда квалитета софтвера и софтверских метрика које се могу применити у процесу развоја и побољшања објектно-оријентисаних софтверских система. У процесу израде докторске дисертације ће бити развијен сопствени алат који је заснован на стандардима квалитета софтвера и интегрисан са NetBeans развојним окружењем.

Евалуацију квалитета софтвера могуће је извршити применом стандарда квалитета софтвера које дефинишу Међународна организација за стандардизацију и Међународна електротехничка комисија. Стандарди квалитета ISO/IEC 25000, ISO/IEC 14598 и ISO/IEC 9126 су посебно значајни за софтверско инжењерство. Стандард квалитета ISO/IEC 14598 дефинише поступак за евалуацију квалитета софтверског производа при чему се као модел квалитета користи ISO/IEC 9126 стандард. Стандард квалитета ISO/IEC 9126 дефинише моделе квалитета софтвера, атрибуте квалитета софтвера, као и софтверске метрике за сваки атрибут квалитета. Ове метрике се користе за мерење квалитета софтверског система. Нова серија стандарда ISO/IEC 25000 треба да замени и прошири ISO/IEC 9126 и ISO/IEC 14598 стандарде.

У процесу развоја софтвера примењују се различите методе развоја софтвера (Јединствени процес развоја софтвера, Ларманова метода развоја софтвера, Екстремно програмирање,...) које су засноване на различитим моделима развоја софтвера (Модел водопода, Итеративно-инкрементални модел, Спирални модел,...), које користе различите стратегије развоја софтвера (засноване на случајевима коришћења, моделима, тестовима,...). У фази пројектовања софтвера користе се општи и објектно-оријентисани принципи пројектовања софтвера, стратегије пројектовања софтвера и патерни пројектовања софтвера. У том смислу ће у докторској дисертацији бити дефинисане и објашњене методе развоја софтвера, општи и објектно-оријентисани принципи пројектовања софтвера, патерни пројектовања софтвера, стратегије пројектовања софтвера и њихова веза са стандардима квалитета софтвера.

Имајући у виду све изнесено може се закључити да развој софтвера представља скуп сложених активности којима је потребно управљати на одговарајући начин. У том смислу је потребно истражити:

а) утицај стандарда, метода и алата квалитета софтвера на процес развоја објектно-оријентисаних софтверских система и

б) везу метода развоја софтвера, општих и објектно-оријентисаних принципа пројектовања софтвера, патерна пројектовања софтвера и стратегије пројектовања софтвера са стандардима квалитета софтвера.

Докторска дисертација је усмерена на развој сопствене методе за побољшање објектно-оријентисаних софтверских система коришћењем стандарда квалитета софтвера и алата за статичку анализу квалитета софтвера.

Циљеви докторске тезе су вишеструки. Циљ је, најпре, да се истражи постојећа литература везана за стандарде квалитета софтвера и да се укаже на могућности примене стандарда квалитета софтвера и софтверских метрика у процесу развоја објектно-оријентисаних софтверских система. У том смислу ће бити дат преглед алата за статичку анализу квалитета софтвера и извршена њихова компаративна анализа.

Наредни циљ докторске дисертације је развој сопственог алата заснованог на стандардима квалитета софтвера и софтверским метрикама који се може применити у процесу развоја и побољшања објектно-оријентисаних софтверских система.

С обзиром на сложеност процеса развоја софтвера, циљ докторске дисертације је да укаже на методе развоја софтвера, опште и објектно-оријентисане принципе пројектовања софтвера, патерне пројектовања софтвера, стратегије пројектовања софтвера и њихову везу са стандардима квалитета софтвера у циљу побољшања објектно-оријентисаних софтверских система.

Све заједно би требало да резултира идентификовању и развоју нове методе за побољшање објектно-оријентисаних софтверских система коришћењем стандарда квалитета софтвера, што је и примарни циљ докторске дисертације.

2.1. Списак изабраних референци

- [1] Abrahamsson, P. (2002). Agile Software Development Methods: Review and Analysis (VTT publications).
- [2] Abran, A., Khelifi, A., Suryn, W., & Seffah, A. (2003). Usability meanings and interpretations in ISO standards. *Software Quality Journal*, 11(4), pp. 325-338.
- [3] Alsultanny, Y. A., & Wohaishi, A. M. (2009, December). Requirements of Software Quality Assurance Model. In *Environmental and Computer Science, 2009. ICECS'09. Second International Conference on* (pp. 19-23). IEEE.
- [4] Antović, I., Vlajić, S., Milić, M., Savić, D., & Stanojević, V. (2012). Model and software tool for automatic generation of user interface based on use case and data model. *IET Software*, 6(6), pp. 559-573.
- [5] Ayewah, N., Hovemeyer, D., Morgenthaler, J. D., Penix, J., & Pugh, W. (2008). Using static analysis to find bugs. *Software, IEEE*, 25(5), pp. 22-29.
- [6] Badri, L., & Badri, M. (2004). A Proposal of a new class cohesion criterion: an empirical study. *Journal of Object Technology*, 3(4), pp. 145-159.
- [7] Ball, T. (1999). The concept of dynamic analysis. In *Software Engineering - ESEC/FSE'99* (pp. 216-234). Springer Berlin Heidelberg.
- [8] Boehm, B. W., Brown, J. R., & Kaspar, H. (1978). Characteristics of software quality.
- [9] Boehm, B. W. (1988). A spiral model of software development and enhancement. *Computer*, 21(5), pp. 61-72.
- [10] Bourque, P., & Fairley, R. E. (Eds.) (2014). *Guide to the Software Engineering Body of Knowledge (SWEBOK (R)): Version 3.0*. IEEE Computer Society Press.
- [11] Carver, J., Jaccheri, L., Morasca, S., & Shull, F. (2003, September). Issues in using students in empirical studies in software engineering education. In *Software Metrics Symposium, 2003. Proceedings. Ninth International* (pp. 239-249). IEEE.
- [12] Chidamber, S. R., & Kemerer, C. F. (1994). A metrics suite for object oriented design. *Software Engineering, IEEE Transactions on*, 20(6), pp. 476-493.
- [13] Cockburn, A. (2000). *Writing effective use cases*. Addison-Wesley Longman Publishing Co., Inc.
- [14] Cohn, M. (2004). *User stories applied: For agile software development*. Addison-Wesley Professional.
- [15] Crosby, P. B. (1980). *Quality is free*. Signet Book.
- [16] Emanuelsson, P., & Nilsson, U. (2008). A comparative study of industrial static analysis tools. *Electronic notes in theoretical computer science*, 217, pp. 5-21.
- [17] Firesmith, D. G. (1996). Use cases: the pros and cons. *Wisdom of the Gurus: A Vision for Object Technology*, pp. 171-180.
- [18] Fowler, M., & Highsmith, J. (2001). The agile manifesto. *Software Development*, 9(8), pp. 28-35.

- [19] Frost, A. A., & Campo, M. J. (2007). Advancing defect containment to quantitative defect management.
- [20] Gomes, I., Morgado, P., Gomes, T., & Moreira, R. (2009). An overview on the static code analysis approach in software development. Faculdade de Engenharia da Universidade do Porto, Portugal.
- [21] Harmon, P. (2010). Business process change: A guide for business managers and BPM and Six Sigma professionals. Morgan Kaufmann.
- [22] Harrison, R., Counsell, S. J., & Nithi, R. V. (1998). An evaluation of the MOOD set of object-oriented software metrics. *Software Engineering, IEEE Transactions on*, 24(6), pp. 491-496.
- [23] Hitz, M., & Montazeri, B. (1996). Chidamber and Kemerer's metrics suite: a measurement theory perspective. *Software Engineering, IEEE Transactions on*, 22(4), pp. 267-271.
- [24] Hofer, T. (2010). Evaluating Static Source Code Analysis Tools (Doctoral dissertation, École Polytechnique Fédérale de Lausanne).
- [25] Hovemeyer, D., & Pugh, W. (2004). Finding bugs is easy. *ACM Sigplan Notices*, 39(12), pp. 92-106.
- [26] ISO/IEC 14598, Information technology -- Software product evaluation, <http://www.iso.org>
- [27] ISO/IEC/IEEE 24765:2010 Systems and software engineering -- Vocabulary, <http://www.iso.org>
- [28] ISO/IEC 25000, Systems and software engineering -- Systems and software Quality Requirements and Evaluation (SQuaRE) -- Guide to SQuaRE, <http://www.iso.org>
- [29] ISO/IEC 9126, Software Engineering - Product Quality, <http://www.iso.org>
- [30] Jacobson, I., Spence, I., & Bittner, K. (2011). Use Case 2.0: The Guide to Succeeding with Use Cases. IJI SA.
- [31] Jacobson, I., Christerson, M., Jonsson, P., & Övergaard, G. (1992). Object-oriented software engineering: a use case driven approach.
- [32] Jacobson, I., Booch, G., Rumbaugh, J. (1999). The unified software development process (Vol. 1). Reading: Addison-Wesley.
- [33] Janzen, D. S., & Saiedian, H. (2008). Does test-driven development really improve software design quality?. *Software, IEEE*, 25(2), 77-84.
- [34] Johnson, B., Song, Y., Murphy-Hill, E., & Bowdidge, R. (2013). Why don't software developers use static analysis tools to find bugs?. In *Software Engineering (ICSE), 2013 35th International Conference on* (pp. 672-681). IEEE.
- [35] Kan, S. H. (2002). Metrics and models in software quality engineering. Addison-Wesley Longman Publishing Co., Inc..
- [36] Kniberg, H., & Skarin, M. (2010). Kanban and Scrum-making the most of both. Lulu.com.
- [37] Kruchten, P. (2004). The rational unified process: an introduction. Addison-Wesley Professional.
- [38] Kruchten, P. (2004). The rational unified process: an introduction. Addison-Wesley Professional.
- [39] Larman, C. (2005). Applying UML and patterns: an introduction to object-oriented analysis and design and iterative development, Third Edition. Pearson Education.
- [40] Laval, J., Bergel, A., & Ducasse, S. (2008, October). Assessing the Quality of your Software with MoQam. In *FAMOOSr 2008-2nd Workshop on FAMIX and Moose in Reengineering*.
- [41] Letouzey, J. L., & Ilkiewicz, M. (2012). Managing technical debt with the sqale method. *IEEE software*, (6), pp. 44-51.

- [42] Letouzey, J. L. (2012, June). The SQALE method for evaluating technical debt. In *Proceedings of the Third International Workshop on Managing Technical Debt* (pp. 31-36). IEEE Press.
- [43] Liu, Y., Khoshgoftaar, T. M., & Seliya, N. (2010). Evolutionary optimization of software quality modeling with multiple repositories. *Software Engineering, IEEE Transactions on*, 36(6), pp. 852-864.
- [44] Lorenz, M., & Kidd, J. (1994). *Object-oriented software metrics: a practical guide*. Prentice-Hall, Inc..
- [45] Martin, R. C. (2003). *Agile software development: principles, patterns, and practices*. Prentice Hall PTR.
- [46] Mayer, T., & Hall, T. (1999). A critical analysis of current OO design metrics. *Software Quality Journal*, 8(2), pp. 97-110.
- [47] McCabe, T. J. (1976). A complexity measure. *Software Engineering, IEEE Transactions on*, (4), pp. 308-320.
- [48] McCabe, T. J., & Butler, C. W. (1989). Design complexity measurement and testing. *Communications of the ACM*, 32(12), 1415-1425.
- [49] Mishra, A., & Dubey, D. (2013). A comparative study of different software development life cycle models in different scenarios. *International Journal*, 1(5).
- [50] Pfleeger, S. L., & Atlee, J. M. (2010). *Software engineering: theory and practice*. Pearson Education.
- [51] Picek, R., & Strahonja, V. (2007, September). Model Driven Development-future or failure of software development. In *IIS* (Vol. 7, pp. 407-413).
- [52] Poppendieck, M., & Poppendieck, T. (2003). *Lean software development: an agile toolkit*. Addison-Wesley Professional.
- [53] Pressman, R. S. (2010). *Software engineering: a practitioner's approach*. Palgrave Macmillan.
- [54] Rutar, N., Almazan, C. B., & Foster, J. S. (2004, November). A comparison of bug finding tools for Java. In *Software Reliability Engineering, 2004. ISSRE 2004. 15th International Symposium on* (pp. 245-256). IEEE.
- [55] Schackmann, H., Jansen, M., & Lichter, H. (2009). Tool support for user-defined quality assessment models. *Proc. of MetriKon*, 9.
- [56] Schmidt, D. C. (2006). Guest editor's introduction: Model-driven engineering. *Computer*, 39(2), pp. 25-31.
- [57] Schneidewind, N. F. (2002). Body of knowledge for software quality measurement. *Computer*, 35(2), pp. 77-83.
- [58] Shepperd, M. (1988). A critique of cyclomatic complexity as a software metric. *Software Engineering Journal*, 3(2), pp. 30-36.
- [59] Sommerville, I. (2011). *Software Engineering, Ninth Edition*. Addison-Wesley.
- [60] Watson, A. H., McCabe, T. J., & Wallace, D. R. (1996). Structured testing: A testing methodology using the cyclomatic complexity metric. *NIST special Publication*, 500(235), pp. 1-114.
- [61] Womack, J. P., & Jones, D. T. (2010). *Lean thinking: banish waste and create wealth in your corporation*. Simon and Schuster.

3. Полазне хипотезе

На основу анализе доступне литературе и постављеног предмета и циља истраживања може се поставити општа хипотеза као и посебне хипотезе докторске дисертације, које ће у раду бити потврђене или оповргнуте.

Општа хипотеза:

- Објектно-оријентисане софтверске системе могуће је унапредити коришћењем стандарда квалитета софтвера.

Посебне хипотезе:

- Стандарди квалитета софтвера могу се применити у процесу развоја и побољшања објектно-оријентисаних софтверских система.
- Софтверске метрике се могу повезати са општим и објектно-оријентисаним принципима пројектовања софтвера, патернима пројектовања софтвера и стратегијама и методама развоја софтвера.
- Могуће је развити сопствени алат за статичку анализу квалитета софтвера који се може применити у процесу развоја и побољшања објектно-оријентисаних софтверских система.
- Сопствени алат за статичку анализу квалитета софтвера могуће је уградити у NetBeans развојно окружење.
- Могуће је развити сопствену методу за побољшање објектно-оријентисаних софтверских система коришћењем стандарда квалитета софтвера и алата за статичку анализу квалитета софтвера.

4. Научне методе истраживања

Да би се успешно реализовала идеја истраживања и потврдиле (или оповргле) постављене хипотезе, у раду ће бити коришћен основни метод истраживања који се базира на постојећим теоријским резултатима и експерименталном раду. Сакупљање и проучавање доступне литературе, њена анализа и систематизација, све са циљем да се покаже оправданост и корисност нове методе побољшања објектно-оријентисаних софтверских система коришћењем стандарда квалитета софтвера.

У раду ће се користити следеће методе истраживања: анализа-синтеза, индукција-дедукција, конкретизација-генерализација, метода моделовања, компаративна метода, метода студије случаја, статистичка метода.

5. Очекивани научни допринос

Допринос који се жели остварити овом докторском дисертацијом се огледа у систематизацији постојећих знања које се односе на стандарде квалитета софтвера и развоју сопствене методе за побољшање објектно-оријентисаних софтверских система, коришћењем стандарда квалитета софтвера и сопственог алата за статичку анализу квалитета софтвера.

Сходно томе, као очекивани резултати овог рада могу се навести:

- Приказ стандарда квалитета софтвера који се могу се применити у процесу развоја и побољшања објектно-оријентисаних софтверских система.
- Преглед и класификација софтверских метрика.
- Преглед и анализа постојећих алата за статичку анализу квалитета софтвера

- Дефинисање и објашњење општих и објектно-оријентисаних принципа пројектовања софтвера, патерна пројектовања софтвера и стратегија и метода развоја софтвера.
- Развој сопственог алата за статичку анализу квалитета софтвера који се може применити у процесу развоја и побољшања објектно-оријентисаних софтверских система.
- Повезивање сопственог алата за статичку анализу квалитета софтвера и NetBeans развојног окружења.
- Развој сопствене методе за побољшање објектно-оријентисаних софтверских система коришћењем стандарда квалитета софтвера и алата за статичку анализу квалитета софтвера.

6. План истраживања и структура рада

У структурном смислу докторска дисертација ће имати следеће целине:

- У уводном делу биће дефинисани предмет, проблем и циљеви истраживања. На основу тога ће бити дефинисане хипотезе истраживања и научне методе истраживања.
- У другом делу ће бити описани објектно-оријентисани софтверски системи. У том смислу биће приказани кључни концепти објектно-оријентисаних софтверских система као што су класе, атрибути, методе, интерфејси, апстрактне класе и сл.
- У трећем делу ће бити описани стандарди квалитета софтвера. У том смислу биће дефинисан квалитет софтвера и дат преглед стандарда квалитета софтвера.
- У четвртном делу биће описане софтверске метрике и њихова веза са стандардима квалитета софтвера.
- У петом делу ће бити дат преглед и анализа постојећих алата за евалуацију квалитета софтвера.
- У шестом делу ће бити представљени механизми побољшања објектно-оријентисаних софтверских система: општи принципи пројектовања софтвера, принципи објектно-оријентисаног пројектовања софтвера, стратегије развоја софтвера и софтверски патерни.
- У седмом делу ће бити представљена оригинална метода за побољшање објектно-оријентисаних софтверских система коришћењем стандарда квалитета софтвера.
- У осмом делу биће извршена евалуација дефинисане методе за побољшање објектно-оријентисаних софтверских система коришћењем стандарда квалитета софтвера. У процесу евалуације користиће се метода студије случаја. У том смислу ће бити спроведен експеримент са две групе студената како би се проверила могућност практичне примене дефинисане методе. На крају ће студенти бити анкетирани. На основу резултата евалуације биће извршена дискусија.
- У деветом делу биће извршена закључна разматрања и дефинисани могући правци даљег истраживања.
- У десетом делу биће дат преглед коришћене литературе у процесу израде докторске дисертације.

На основу изложеног може се дефинисати оквирни садржај докторске дисертације:

1. Увод
2. Објектно-оријентисани софтверски системи
 - 2.1. Објектно-оријентисани концепти
 - 2.2. Објектно-оријентисане технологије
 - 2.3. Окружења за развој објектно-оријентисаних софтверских система
3. Стандарди квалитета софтвера
 - 3.1. Квалитет софтвера
 - 3.2. Преглед стандарда квалитета софтвера и модела квалитета софтвера
4. Софтверске метрике
 - 4.1. Мерења и софтверске метрике
 - 4.2. Карактеристике софтверских метрика
 - 4.3. Софтверске метрике у стандардима квалитета софтвера
 - 4.4. Класификација софтверских метрика
 - 4.5. Преглед софтверских метрика
5. Преглед и анализа постојећих алата за статичку анализу квалитета софтвера
 - 5.1. Значај алата за статичку анализу квалитета софтвера
 - 5.2. Преглед алата за статичку анализу квалитета софтвера
 - 5.3. Компаративна анализа алата за статичку анализу квалитета софтвера
6. Механизми побољшања објектно-оријентисаних софтверских система
 - 6.1. Општи принципи пројектовања софтвера
 - 6.2. Принципи објектно-оријентисаног пројектовања софтвера
 - 6.3. Стратегије пројектовања софтвера
 - 6.4. Методе развоја софтвера
 - 6.5. Патерни пројектовања
7. SilabQOSS метода за побољшање објектно-оријентисаних софтверских система коришћењем стандарда квалитета софтвера
 - 7.1. Опис предложене методе
 - 7.2. Активности предложене методе
 - 7.3. Артефакти предложене методе
8. Евалуација предложене методе за побољшање објектно-оријентисаних софтверских система коришћењем стандарда квалитета софтвера
 - 8.1. Студија случаја
 - 8.2. Примена предложене методе за дату студију случаја
 - 8.3. Анализа резултата примене предложене методе
9. Закључак
10. Литература

7. Закључак и предлог

На основу претходно изложеног, комисија је констатовала да предложена тема докторске дисертације, под називом **„Побољшање објектно-оријентисаних софтверских система коришћењем стандарда квалитета софтвера“**, по предмету истраживања, циљевима и методологији има научну заснованост и одговара нивоу докторске дисертације. Предложена тема представља значајну област истраживања са теоријског и практичног становишта.

Комисија је констатовала да кандидат Милош Милић испуњава формалне услове и поседује све потребне квалификације за одобравање израде докторске дисертације за предложену тему.

На основу изнетог, комисија за оцену испуњености услова кандидата и научне заснованости теме докторске дисертације предлаже Наставно-научном већу Факултета организационих наука да кандидату Милошу Милићу одобри израду дисертације под називом:

„Побољшање објектно-оријентисаних софтверских система коришћењем стандарда квалитета софтвера“

и да се за ментора именује проф. др Сениша Влајић, ванредни професор Факултета организационих наука.

Београд, 05.05.2017.

Комисија

Др Сениша Влајић, ванредни професор
Универзитет у Београду, Факултет организационих наука

Др Саша Лазаревић, ванредни професор
Универзитет у Београду, Факултет организационих наука

Др Драган Бојић, ванредни професор
Универзитет у Београду, Електротехнички факултет