UČNI NAČRT PREDMETA / COURSE SYLLABUS	
Predmet:	Sodobne metode razvoja programske opreme
Course title:	Modern software development methods

Študijski program in stopnja Study programme and level	Študijska smer Study field	Letnik Academic year	Semester Semester
Interdisciplinarni magistrski študijski program Računalništvo in matematika	ni smeri	1 in 2	drugi
Interdisciplinary Masters study programme Computer Science and Mathematics	none	1 in 2	second

Vrsta predmeta / Course type	izbirni

Univerzitetna koda predmeta / University course code: 63515

Predavanja Lectures	Seminar Seminar	Vaje Tutorial	Klinične vaje work	Druge oblike študija	Samost. delo Individ. work	ECTS
45	10	20			105	6

Nosilec predmeta / Lecturer: Damjan Vavpotič

Jeziki / Languages:	Predavanja / Lectures:	slovenski/Slovene, angleški/English
	Vaje / Tutorial:	slovenski/Slovene, angleški/English

Pogoji za vključitev v delo oz. za opravljanje študijskih obveznosti:	Prerequisites:

Vsebina:	Content (Syllabus outline):

Predavanja:	Lectures:
Vsebina predmeta se prilagaja trenutno aktualnim trendom na področju razvoja programske opreme. Trenutno je prilagojena obravnavi agilnih metod in empiričnemu ovrednotenju njihove učinkovitosti:	The contents adapts to current trends in software development. At the moment the focus is on agile methods for software development and empirical evaluation of these methods:
1. Pregled značilnosti agilnih metod za razvoj programske opreme	1. Overview of agile methods for software development and their characteristics
2. Vodenje agilnih projektov razvoja programske opreme	2. Agile software project management
iterativen in inkrementalen razvoj	Iterative and incremental development
metoda Scrum	Scrum
3. Tipične prakse ekstremnega programiranja	3. Typical Extreme Programming practices
Programiranje v parih	Pair programming
Testno voden razvoj programske opreme	Test driven development (TDD)
Sprotno preoblikovanje kode (refactoring)	Refactoring
Orodja za podporo testno vodenemu razvoju in preoblikovanju kode	Software tools that support TDD and refactoring
4. Pomanjkljivosti ekstremnega programiranja	4. Weaknesses of Extreme Programming
5. Kanban in vitek razvoj programske opreme	5. Kanban and lean software development
6. Metrike v programski opremi in merjenje učinkovitosti razvojnega procesa	6. Metrics in Software Engineering and measuring the effectiveness of the software development process
7. Študija primera: empirično ovrednotenje posameznih praks iz točk 2 in 3	7. Case study: Empirical evaluation of the aforementioned practices
8. Primerjava agilnega pristopa z modeli za discipliniran razvoj programske opreme (Personal Software Process, Team Software Process, Capabilty Maturity Model)	8. Comparison of agile approach to traditional disciplined software development processes (Personal Software Process, Team Software Process, Capability Maturity Model)
Vaje:	Lab practice:
	The purpose of lab practice is twofold:

Namen vaj je dvojen: 1. seznanjanje s sodobnimi orodji za razvoj programske opreme, 2. empirično ovrednotenje posameznih pristopov k razvoju programske opreme na podlagi praktičnega dela na projektih, ki so čim bolj podobni realnim. Delo izven kontaktnih ur: Študenti razvijajo programe, ki so sestavni del projekta, ki služi kot osnova za študijo primera.	1. to acquaint students with modern software development tools, 2. to empirically evaluate different approaches to software development through practical work on (almost) real software projects. Individual work outside of contact hours: Students develop programs that are part of the project that serves as a case study.

Temeljni literatura in viri / Readings:

1. K. Schwaber: Agile Project Management with Scrum, Microsoft Press, 2004.

2. K. Beck: Extreme Programming Explained, Addison-Wesley, več izdaj.

3. D. Anderson, Kanban – Successful Evolutionary Change for Your Technology Business, Sequim, WA: Blue Hole Press, 2010.

4. K. Beck: Test-Driven Development: By Example, Addison-Wesley, 2003.

5. M. Cohn: User stories applied, Addison-Wesley, 2004.

Dodatna literatura:

B. Boehm, R. Turner: Balancing Agility and Discipline – A Guide for the Perplexed, Pearson Education, 2004.

2. CMMI for Development (CMMI-DEV), Version 1.2. CMU/SEI-2006-TR-008, Software Engineering Institute, Carnegie Mellon University, 2006.

Cilji in kompetence: Objectives and competences:

Cilji in kompetence:	Objectives and competences:
Cilj predmeta je poglobljena obravnava sodobnih (trenutno aktualnih) metod razvoja programske opreme v primerjavi z že uveljavljenimi pristopi. Študenti delajo na obsežnejšem projektu, ki služi kot študija primera za ovrednotenje novih pristopov, da bi ugotovili njihove prednosti in pomanjkljivosti.	In depth treatment and empirical evaluation of modern software development methods in comparison to traditional approach. Students work on a project that serves as a case study for evaluation of modern approaches in order to find their strengths and weaknesses. The competences students gain are: the ability to define, understand and solve

Predvidene kompetence:

sposobnost opredelitve, razumevanja in reševanja kreativnih strokovnih izzivov na področju računalništva in informatike,

sposobnost uporabe pridobljenega znanja pri samostojnem delu za reševanje tehničnih in znanstvenih problemov na področju računalništva in informatike, sposobnost nadgradnje pridobljenega znanja,

sposobnost skupinskega dela v profesionalnem delovnem okolju, vodenje manjše strokovne skupine,

sposobnost administrativnega vodenja procesov, povezanih z raziskovanjem, industrijo, izobraževanjem in drugimi področji,

sposobnost razumevanja in uporabe znanja računalništva in informatike na drugih tehničnih in relevantnih področjih (ekonomija, organizacijske vede ipd.),

praktično znanje in spretnosti, potrebne za uspešno strokovno delo na področju računalništva in informatike,

samostojno reševanje zahtevnih razvojnih, inženirskih in organizacijskih nalog kot tudi povprečno zahtevnih raziskovalnih nalog na področju računalništva in informatike.

creative professional challenges in computer and information science,

the ability to apply acquired knowledge in independent work for solving technical and scientific problems in computer and information science, the ability to upgrade acquired knowledge.

the ability of teamwork within the professional environment, management of a small professional team,

the ability for administrative management of processes related to research, industry, education and other fields,

the ability to understand and apply computer and information science knowledge to other technical and relevant fields (economics, organisational science, etc),

practical knowledge and skills of computer hardware, software and information technology necessary for successful professional work in computer and information science,

independently tackle demanding developmental, engineering, and organisational tasks as well as moderately demanding research tasks in their fields of study.

Predvideni študijski rezultati:

Po uspešnem zaključku tega predmeta bo študent zmožen:
poznati in razumeti sodobne pristope k razvoju programske opreme

razumeti ključne prednosti sodobnih metod razvoja programske opreme (v primerjavi z že uveljavljenim tradicionalnim pristopom)

uporabljati orodja, ki podpirajo sodobne

Intended learning outcomes:

After the completion of the course a student will be able to:
know and understand modern approaches to software development

understand key benefits of modern software development methods (in comparison to traditional software development)

use tools that support modern software

metode razvoja programske opreme	development methods
uporabiti sodobne metode razvoja programske opreme pri razvoju dejanskih softverskih projektov	apply modern software development methods in the context of a real software development project
empirično ovrednotiti rezultate novih metod	empirically evaluate the outcomes of new methods
bolje razumeti različne faktorje in okoliščine, ki vplivajo na uspešnost softverskih projektov	increase understanding of different factors and circumstances that affect the success of a software development project
izboljšati sposobnosti za skupinsko delo, vodenje, načrtovanje in organizacijo, medsebojno komuniciranje, pisno in ustno poročanje.	increase professional skills like team-work, management, planning and organization, written and oral communication

Metode poučevanja in učenja: **Learning and teaching methods:**

Predavanja z aktivno udeležbo študentov (razlaga, diskusija, primeri, reševanje problemov). Laboratorijske vaje s praktičnim delom na večjem projektu, ki služi kot študija primera za ovrednotenje posameznih pristopov k razvoju programske opreme.	Lectures with active participation on the part of students (discussion, examples, problem solving). Lab practice requires practical work on an almost real project that serves as a case study for evaluation of different approaches to software development.

Načini ocenjevanja:	Delež (v %) / Weight (in %)	Assessment:
Način (pisni izpit, ustno izpraševanje, naloge, projekt): Sprotno preverjanje (domače naloge, kolokviji in projektno delo)		Type (examination, oral, coursework, project): Continuing (homework, midterm exams, project work)
Končno preverjanje (pisni in ustni izpit)		Final (written and oral exam)
	50%	
Ocene: 6-10 pozitivno, 5 negativno		Grading: 6-10 pass, 5 fail (according to the rules of University of Ljubljana).
(v skladu s Statutom UL).	50%	

Reference nosilca / Lecturer's references:

Damjan Vavpotič:

– BAJEC, Marko, VAVPOTIČ, Damjan. A framework and tool-support for reengineering software development methods. Informatica, ISSN 0868-4952, 2008, vol. 19, no. 3, str. 321-344, ilustr. [COBISS.SI-ID 6701396]

– VAVPOTIČ, Damjan, BAJEC, Marko. An approach for concurrent evaluation of technical and social aspects of software development methodologies. Information and software technology, ISSN 0950-5849. [Print ed.], 2009, vol. 51, no. 2, str. 528-545, ilustr. [COBISS.SI-ID 6803284]

– VAVPOTIČ, Damjan, VASILECAS, Olegas. An approach for assessment of software

development methodologies suitability. Elektronika ir elektrotechnika. 2011, vol. 114, no. 8,

str. 107-110 [COBISS.SI-ID 8713044]

– VAVPOTIČ, Damjan, HOVELJA, Tomaž. Improving the evaluation of software development methodology adoption and its impact on enterprise performance. Computer science and information systems, ISSN 1820-0214. [Print ed.], Jan. 2012, vol. 9, no. 1, str. 165-187, ilustr. [COBISS.SI-ID 8880212]

– VRHOVEC, Simon, TRKMAN, Marina, KUMER, Aleš, KRISPER, Marjan, VAVPOTIČ, Damjan.

Outsourcing as an economic development tool in transition economies : scattered global software development. Information technology for development. ISSN 0268-1102, 2015, vol. 21, no. 3, str. 445-459 [COBISS.SI-ID 10420564]