

RTU studiju kurss "Programmatūras inženierija (pamatkurss)"

33000 Datorzinātnes, informācijas tehnoloģijas un enerģētikas fakultāte

Vispārējā informācija

Kods	DIP218
Nosaukums	Programmatūras inženierija (pamatkurss)
Studiju kursa statuss programmā	Obligātais/Ierobežotās izvēles
Atbildīgais mācītbspēks	Natālija Prokofjeva - Doktors, Vadošais pētnieks
Mācītbspēks	Jeļena Jevsjukova - Vieslektors Sabina Kataļņikova - Asistents Oksana Zavjalova - Lektors Jānis Amoliņš - Lektors
Apjoms daļās un kredītpunktos	1 daļa, 3.0 kredītpunkti
Studiju kursa īstenošanas valodas	LV, EN
Anotācija	Programmatūras izstrāde, modeļi un prasību specifikācija. Programmatūras projektēšana. Projektēšanas process un posmi. Arhitektūras un detalizētā projektēšana. Diagrammas. Lietotāja interfeisa projektēšanas principi. Grafiskais un komandu lietotāja interfeiss. Lietotāja ceļvedis. Programmatūras implementēšana un atklūdošana, testēšana un uzturēšana.
Mērķis un uzdevumi, izteikti kompetencēs un prasmēs	Priekšmeta mērķis ir iepazīstināt studentus ar programminženierijas modeļiem un programmatūras sistēmu izstrādes posmiem, kā arī iemācīt analizēt un dokumentēt prasības, izstrādāt lietotāja interfeisu un testpiemērus programmu testēšanai. Priekšmeta uzdevumi: 1) aplūkot programmatūras dzīves ciklu, analizējot katra posma mērķus un uzdevumus; 2) analizēt programmatūras izstrādes modeļus: kādām programmatūras klasēm ir lietderīgs, kādas tehnikas un kā tiek lietotas, pozitīvās īpašības un trūkumi; 3) iemācīt studentiem praktiski lietot iegūtās zināšanas uzdotās programmatūras sistēmas izstrādē.
Patstāvīgais darbs, tā organizācija un uzdevumi	Laboratorijas darbu izpilde: 1) programmatūras specifikēšana; 2) lietotāja interfeisa projektēšana; 3) testpiemēru izstrāde (uzdevums katram laboratorijas darbam ir izklāstīts atsevišķā dokumentā).
Literatūra	1. Zaiceva L. Programmatūras izstrādes tehnoloģija. – 2. izd. – Rīga : RTU, 2002. – 244 lpp. 2. Sommerville I. Software Engineering. – 8th Edition. – Addison-Wesley Pub. Comp, 2010.- 792 p. 3. Pfleeger S.L., Atlee J.M. Software Engineering: Theory and Practice. – 4th Edition. – PRENTICE HALL, 2010. – 792 p. 4. Van Vliet H. Software Engineering. Principles and Practice. – John Wiley & Sons, 2008. – 740 p. 5. Bell D. Software Engineering for Students. – Pearson Education (US), 2005.- 448 p. 6. McCaffrey J.D. Software Testing: Fundamental Principles and Essential Knowledge. – BookSurge Publishing, 2009. – 118 p. 7. Ammann P., Offutt J. Introduction to Software Testing. – Cambridge University Press, 2008. – 344 p. 8. Гагарина Л.Г., Кокорева Е.И., Виснадул Б.Д. Технология разработки программного обеспечения : учебное пособие. – М.: ИД «ФОРУМ»: ИНФРА-М, 2008. – 400 с.
Nepieciešamās priekšzināšanas	Programmēšana, Datu bāzu pārvaldības sistēmas

Studiju kursa saturs

Saturs	Pilna un nepilna laika klātienes studijas		Nepilna laika neklātienes studijas	
	Kontakt stundas	Patstāv. darbs	Kontakt stundas	Patstāv. darbs
Programminženierijas jēdziens un klasifikācija. Programmatūras krīze un tās cēloņi. Programmatūras dzīves cikls.	2	0	0	0
Programmatūras izstrādes procesa modeļi.	3	1	0	0
Prasību analīze, definēšana, specifikēšana un dokumentēšana. Funkcionālās un nefunkcionālās prasības.	5	12	0	0
Programmatūras projektēšanas process, posmi un metodes. Sistēmas specifikācija.	4	3	0	0
Lietotāja interfeisa projektēšana. Grafiskais lietotāja interfeiss. Komandu interfeiss. Lietotāja ceļvedis.	4	16	0	0
Programmatūras implementēšanas metodes. Programmu atklūdošana.	4	3	0	0
Verifikācijas un validācijas procesa mērķi un tehnikas. Statiskā verifikācija.	2	1	0	0
Programmatūras testēšanas melnās un baltās kastes metodes. Testēšanas dokumentācija.	6	12	0	0
Programmatūras uzturēšanas process. Darbības un problēmas programmatūras uzturēšanas procesā.	2	0	0	0
Kopā:	32	48	0	0

Sasniedzamie studiju rezultāti un to vērtēšana

Sasniedzamie studiju rezultāti	Rezultātu vērtēšanas metodes
Pārzina programmatūras izstrādes posmus un modeļus, darbus, kas jāizpilda katrā posmā, posmu nodevumus un dokumentus.	Nokārtots eksāmens, par kuru iegūta pozitīvā atzīme.

Spēj veidot sistēmas modeli, definēt un specifiķēt prasības, kā arī sagatavot prasību dokumentu.	Pozitīvs vērtējums par patstāvīgi izpildīto 1. laboratorijas darbu.
Spēj izstrādāt un aprakstīt lietotāja interfeisu nelielai programmatūras sistēmai.	Pozitīvs vērtējums par patstāvīgi izpildīto 2. laboratorijas darbu.
Spēj izveidot testpiemērus programmatūras sistēmas testēšanai.	Pozitīvs vērtējums par patstāvīgi izpildīto 3. laboratorijas darbu.

Studiju kursa plānojums

Daļa	KP	Stundas			Pārbaudījumi		
		Lekcijas	Prakt d.	Laborat	Ieskaite	Eksām.	Darbs
1.	3.0	1.0	0.0	1.0		*	