Zał. nr 4 do ZW 64/2012

FACULTY Informatics and Management / DEPARTMENT of Informatics

SUBJECT CARD

Name in Polish Wprowadzenie do Inżynierii Oprogramowania
Name in English Introduction to Software Engineering
Main field of study (if applicable): Informatics
Specialization (if applicable): ……………………..
Level and form of studies: 1st/ ~~2nd~~* level, full-time / ~~part-time~~*
Kind of subject: obligatory~~ / optional / university-wide~~*
Subject code INZ0263Wcl
Group of courses YES~~/ NO~~*

	Lecture	Classes	Laboratory	Project	Seminar
Number of hours of organized classes in University (ZZU)	30	15	15		
Number of hours of total student workload (CNPS)	60	30	60		
Form of crediting	Examination / ~~crediting with grade~~*	~~Examination /~~ crediting with grade*	~~Examination /~~ crediting with grade*	Examination / crediting with grade*	Examination / crediting with grade*
For group of courses mark (X) final course	X				
Number of ECTS points	2	1	2		
including number of ECTS points for practical (P) classes			2		
including number of ECTS points for direct teacher-student contact (BK) classes	1,2	0,6	1,2		

*delete as applicable

PREREQUISITES RELATING TO KNOWLEDGE, SKILLS AND OTHER COMPETENCES

1. Knowledge of object-oriented programming language (e.g. java)

\

SUBJECT OBJECTIVES

C1. To obtain basic knowledge of software engineering, including knowledge of: development life cycle models, requirements engineering issues, domain analysis and software testing.
C2. To gain practical skills in requirements specification, domain modeling and test cases definition.

SUBJECT EDUCATIONAL EFFECTS

relating to knowledge:
PEK_W01. A student knows basic software life-cycle models, and languages used for software system specification
PEK_W02. A student knows basic diagrams used for requirement specification and domain modeling.
PEK_W03. A student distinguishes different kind of tests, and test levels.

relating to skills:
PEK_U01. A student creates simple models representing functional requirements and basic

entities in a modeled domain
PEK_U02. A student proposes a user-interface, and necessary documents
PEK_U03. A student specifies in a formal language business constraints
PEK_U04. A student specifies unit tests for source code, and functional test-cases

PROGRAMME CONTENT

Form of classes – lecture		Number of hours
Lec 1	Introduction. Basic notions in software engineering.	2
Lec 2	Software life-cycle processes. Software life-cycle models.	2
Lec 3	Fundamentals of requirement engineering.	2
Lec 4	Modeling of structure and behavior of software systems.	2
Lec 5	OCL as a formal specification language.	2
Lec 6	Human-computer interaction. Standards.	2
Lec 7	Software design.	2
Lec 8	Construction. Style guidelines. Source code versioning.	2
Lec 9	Software quality during the development.	2
Lec 10	Testing – types, techniques; automatic testing	2
Lec 11	Software system deployment.	2
Lec 12	Problems and methods of software maintenance.	2
Lec 13	Organization of software design.	2
Lec 14	Configuration management and change management.	2
Lec 15	Repetition.	2
	Total hours	30

Form of classes - class		Number of hours
Cl 1	Introduction	1
Cl 2	Use-case models	2
Cl 3	Use-case specifications	2
Cl 4	User-interface prototype	2
Cl 5	Domain model	2
Cl 6	Formal specification of business constraints	2
Cl 7	Unit, and functional test-cases specification	2
Cl 8	Final test	2
	Total hours	15

Form of classes – laboratory		Number of hours
Lab 1	Introduction	1

Lab 2	Use-case models	2
Lab 3	Use-case scenarious	2
Lab 4	GUI prototype	2
Lab 5	Domain models	2
Lab 6	OCL constraints	4
Lab 7	Test-cases	2
	Total hours	15

TEACHING TOOLS USED

N1. Informative lecture supported by multimedia presentations
N2. Case tools, IDE used for programming and testing
N3. Examples of documents
N4. E-learning system used for materials publication

EVALUATION OF SUBJECT EDUCATIONAL EFFECTS ACHIEVEMENT

Evaluation (F – forming (during semester), P – concluding (at semester end)	Educational effect number	Way of evaluating educational effect achievement
F0 – class activity	PEK_U01,… PEK_U04	A student can receive up to 1 point during each classes for activity (presenting task solution, discussion)
F1 – class test	PEK_U01,… PEK_U04	A test with several practical task to be solved. The tasks are assessed against their correctness, completeness, etc. and scored in points
F2 – class final grade	PEK_U01,… PEK_U04	A grade calculated on the base of NP=F0+F1 according to the formula: <50%, 60%) → 3.0 <60%, 70%) →3.5 <70%, 80%) → 4.0 <80%, 90%) → 4.5 <90%, 95%) → 5.0 <95%, 100%> → 5.5 Where 100% represents the max. of test points
F3 – lab 2-3	PEK_U01	Correctness, and completeness of solution of specified tasks, scored in points (max 25% of the lab final grade)
F4 – lab 4-5	PEK_U01, PEK_U02	Correctness, and completeness of solution of specified tasks, scored in points (max 25% of the lab final grade)
F5 – lab 6	PEK_U03	Correctness, and completeness of solution of specified tasks, scored in points (max 25% of the lab final grade)
F6 – lab 7	PEK_U04	Correctness, and completeness of solution of specified tasks, scored in points (max 25% of the lab final grade)
F7 – lab final grade	PEK_U01, PEK_U02, PEK_U03	A grade calculated on the base of NP=F3+F4+F5+F6 according to the formula: <50%, 60%) → 3.0 <60%, 70%) →3.5 <70%, 80%) → 4.0 <80%, 90%) → 4.5 <90%, 95%) → 5.0

		<95%, 100%> → 5.5
F8 – exam	PEK_W01, PEK_W02, PEK_W03	Test scored in points. The final grade evaluated by the formula: <50%, 60%) → 3.0 <60%, 70%) →3.5 <70%, 80%) → 4.0 <80%, 90%) → 4.5 <90%, 95%) → 5.0 <95%, 100%> → 5.5
P – final grade	All	The grade is calculated as: 0.3 * F2 + 0.4 * F7 + 0.3 * F8

PRIMARY AND SECONDARY LITERATURE

PRIMARY LITERATURE:

[1] R. Pressman, Software Engineering: A Practitioner's Approach 7th edition, McGraw-Hill 2009
[2] M. D. Hamlet, J. Maybee, The Engineering of Software: A Technical Guide for the Individual, Addison-Wesley 2000

SECONDARY LITERATURE:

[1] M. Fowler, UML Distilled 3rd edition, Addison-Wesley 2004
[2] T. Clark, Object Modeling with the OCL: The rationale behind the Object Constraint Language, Springer 2002

SUBJECT SUPERVISOR (NAME AND SURNAME, E-MAIL ADDRESS)

Iwona.Dubielewicz, Iwona.Dubielewicz@pwr.wroc.pl
Bogumiła Hnatkowska, Bogumila.Hnatkowska@pwr.wroc.pl

MATRIX OF CORRELATION BETWEEN EDUCATIONAL EFFECTS FOR SUBJECT
Introduction to Software Engineering
AND EDUCATIONAL EFFECTS FOR MAIN FIELD OF STUDY
Informatics
AND SPECIALIZATION ……………………………..

Subject educational effect	**Correlation between subject educational effect and educational effects defined for main field of study and specialization (if applicable)****	**Subject objectives*****	**Programme content*****	**Teaching tool number*****
PEK_W01	K1INF_W07	C1	Lec2… Lec14	N1, N4
PEK_W02	K1INF_W07	C1	Lec3 … Lec7	N1, N3, N4
PEK_W03	K1INF_W07	C1	Lec9 … Lec10	N1, N4
PEK_U01	K1INF_U03	C2	Cl2, Cl3, Cl5 Lab2, Lab3, Lab5	N2, N4
PEK_U02	K1INF_U03	C2	Cl4, Lab4	N2, N4
PEK_U03	K1INF_U03	C2	Cl5, Lab6	N2, N4
PEK_U04	K1INF_U03	C2	Cl7, Lab7	N2, N4

** - enter symbols for main-field-of-study/specialization educational effects
*** - from table above