Національний технічний університет України
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»

Емблема кафедри
(за наявності)

Department of Computer Engineering

COURSEWORK ON SOFTWARE ENGINEERING

Syllabus

Requisites of the Course

Cycle of Higher Education	***First cycle of higher education (Bachelor's degree)***
Field of Study	*12 Information Technologies*
Speciality	*123 Computer Engineering*
Education Program	*Computer Systems and Networks*
Type of Course	*Normative*
Mode of Studies	*full-time*
Year of studies, semester	*2nd year, 4th semester*
ECTS workload	*1 credits (30 hours of self-study)*
Testing and assessment	*Coursework / defense of coursework*
Course Schedule	*self-study*
Language of Instruction	*English*
Course Instructors	Teacher of coursework: PhD, Associate Professor, Andrii Antoniuk, mobile +380503588019, email antoniuk.andrii@lll.kpi.ua, personal web page http://se-111.blogspot.com
Access to the course	*Cisco, Google, TeamViewer, Zoom, http://se-111.blogspot.com*

Outline of the Course

1. **Course description, goals, objectives, and learning outcomes**

The purpose of studying the credit module "Coursework in Software Engineering" is to consolidate, deepen, and generalize the theoretical knowledge and practical skills that students acquire during the study of the discipline "Software Engineering", the formation of a number of competencies in students, namely:

- *application of the principles of designing and building software for computer systems;*
- *application of a template approach to the design of software systems and the UML language;*
- *application of methods and means of collective software development;*
- *application of an object-oriented approach to collective software development.*

The subject of the credit module "Coursework in Software Engineering" is methods, methods, patterns, tools and systems for research, automated and automatic design, debugging, production and operation, design documentation, standards, procedures and means of supporting software life cycle management.

Learning outcomes. After mastering the credit module, students must demonstrate the following learning outcomes *(ФК3, ФК8, ПРН10)::*

knowledge:

- *stages, methods and standards of software development;*
- *patterns design method;*
- *basic patterns of software design and the principles of their construction and application;*
- *principles of component design.*

skill:

- *determine the sources of requirements and ensure the process of their formation;*
- *develop user requirements specifications;*
- *perform requirements analysis;*
- *develop a specification of software requirements;*
- *model various aspects of the system;*
- *design components of an architectural solution;*
- *apply and create reusable components.*

experience:

- *pattern design method;*
- *development of user requirements specifications;*
- *modeling various aspects of the system;*
- *application and creation of reusable components.*

2. Prerequisites and post-requisites of the course (the place of the course in the scheme of studies in accordance with curriculum)

A list of disciplines, or knowledge and skills, possession of which is necessary for a student:

- *a basic level of knowledge, skills and abilities in such disciplines as "Discrete Mathematics", "Differential and Integral Calculus" and "Physics";*
- *basic level of knowledge of Java programming language, or JavaScript, or C ++, or Python;*
- *basic level of mastery of the tools of the integrated development environment Eclipse or IntelliJ IDEA;*
- *basic level of mastery of UML modeling tools ArgoUML or Umbrello.*

The following disciplines are based on the learning outcomes of this discipline:

- *computer architecture;*
- *computer networks;*
- *parallel programming;*
- *system software;*
- *coursework on computer architecture;*
- *coursework on system software.*

3. Content of the course

The coursework is performed according to an individual task and is prepared for defense in the final period of theoretical training. The coursework must be prepared for the defense within the term stipulated in the assignment and agreed with the teacher. An explanatory note in the composition is submitted to the defense of the coursework:

- *title page;*
- *technical task;*
- *content;*
- *introduction;*
- *main part; includes the following sections:*
  - *an overview of collective software development methods;*
  - *software application design, development of possible precedents of the developed software application;*
  - *development of a software application;*
  - *software application testing;*
- *conclusions;*

- *references;*
- *applications.*

3.1 Stages of coursework

The main stages of coursework:

Getting a topic and task
Selection and study of literature
Formation of the technical task
Overview of collective software development methods
Development of precedents of the developed software application
Development of a software application
Software application testing
Issuance of an explanatory note
Submission of a coursework for review
Protection of the coursework

4. Coursebooks and teaching resources

Basic literature:

1. *Design Patterns: elements of reusable object-oriented software / Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides. Indianapolis: - Addison-Wesley, 1994. - 417 p. ISBN: 0201633612.*
2. *Mark Grand. Patterns in Java: A Catalog of Reusable Design Patterns Illustrated with UML, Volume 1, 2nd Edition. - Wiley Publishing, 2002. - 480 p. ISBN: 0471258393*
3. *Інженерія програмного забезпечення: методичні вказівки до виконання лабораторних робіт для студентів напряму підготовки 6.050102 «Комп'ютерна інженерія». Ч. І. Структурні шаблони. / Уклад.: А. О. Болдак, О. Н. Абу Усбах. - К.: НТУУ «КПІ», 2011. – 40 с.*
4. *Інженерія програмного забезпечення: методичні вказівки до виконання лабораторних робіт для студентів напряму підготовки 6.050102 «Комп'ютерна інженерія». Ч. ІІ. Шаблони поведінки та породжувальні шаблони. / Уклад.: А. О. Болдак, О. Н. Абу Усбах. - К.: НТУУ «КПІ», 2011. - 44 с.*
5. *Theo Mandel. The Elements of User Interface Design. - John Wiley & Sons, 2005. - 468 p. ISBN: 0471162671*
6. *Bruce Eckel. Thinking in Java, 4th ed. – Printice hall, 2006. – 1057 p. ISBN: 0131872486*

Additional literature:

1. *Програмна інженерія: [Підручник] / Лавріщева К.М. – К.: Академперіодика, 2008. – 319 с. ISBN 978–966–02–5052–9*
2. *Об'єктно-орієнтоване програмування: [Підручник] / В.В. Бублик. – К.: ІТ-книга, 2015. – 624 с.*
3. *Бандура В.В., Храбатин Р.І. Архітектура та проектування програмного забезпечення: конспект лекцій. — Івано-Франківськ: ІФНТУНГ, 2012. — 240 с.*
4. *Проектування інформаційних систем: Загальні питання теорії проектування ІС (конспект лекцій): навч. посіб. для студ. спеціальності 122 «Комп'ютерні науки» / КПІ ім.*

Ігоря Сікорського; уклад.: О. С. Коваленко, Л. М. Добровська. – Київ : КПІ ім. Ігоря Сікорського, 2020. – 192 с.

5. Поморова О.В., Говорущенко Т.О. Проектування інтерфейсів користувача: навч.посібник - Хмельницький: ХНУ, 2011. - 206 с.
6. Проектування та моделювання програмного забезпечення сучасних інформаційних систем / Г. В. Табунщик, Т.І. Каплієнко, О.А. Петрова – Запоріжжя : Дике Поле, 2016. – 250 с.
7. Навчальний посібник з дисципліни «Технології розробки програмного забезпечення» для студентів спеціальності 123 «Комп’ютерна інженерія» / Дегтярьова Л.М., Гроза П.М., Сомов С.В. – Полтава: ПолтНТУ, 2017. – 218 с.
8. Буч Г., Якобсон А., Рамбо Дж. UML. Классика CS. 2-е изд./ Пер. с англ.; Под общей редакцией проф. С. Орлова — СПб.: Питер, 2007. —736 с.
9. Вебер Дж. Технология Java / Пер. с англ. – СПб.: БХВ-Петербург, 1997. – 1104 с.
10. Кериевски Дж. Рефакторинг с использованием шаблонов (паттернов проектирования) / Пер. с англ. — М.: «Вильямс», 2006. — 400 с.
11. Эккель Б. Философия Java. Библиотека программиста / Пер. с англ. – С-Пб.: Питер, 2003. – 976 с.
12. Ларман К. Применение UML 2.0 и шаблонов проектирования / Пер. с англ. — М.: «Вильямс», 2013. — 736 с.
13. Ноутон П., Шилт Г. Java 2 / Пер. с англ. – СПб,: БХВ-Петербург, 2000. – 1072 с.
14. Фаулер М. Архитектура корпоративных программных приложений. — М.: «Вильямс», 2007. — 544 с.
15. Хорстманн К., Корнелл Г. Java 2. Библиотека профессионала, том 1. Основы. / Пер. с англ. — 9-е изд. — М.: «Вильямс», 2014. — 864 с.
16. Хорстманн К., Корнелл Г. Java 2. Библиотека профессионала, том 2. Тонкости программирования. / Пер. с англ. — 9-е изд. — М.: «Вильямс», 2014. — 1008 с.
17. Шилдт Г., Холмс Дж. Искусство программирования на Java / Пер. с англ.— М.: «Диалектика», 2005. — 336 с.
18. Шмуллер Дж. Освой самостоятельно UML 2 за 24 часа. Практическое руководство / Пер. с англ. — М.: «Вильямс», 2005. — 416 с.
19. Эмблер С., Садаладж П. Рефакторинг баз данных: эволюционное проектирование. — М.: «Вильямс», 2007. — 368 с.
20. Гамма Э. и др., Приемы объектно-ориентированного проектировании. Паттерны проектированя. / Пер. с англ. – С-Пб.: Питер, 2001. – 366 с.
21. Гранд М. Шаблоны проектирования в Java. / Пер. с англ. – М.: Новое знание, 2004. – 560 с.
22. Мандел Т. Разработка пользовательского интерфейса. / Пер. с англ. – М.: ДМК Пресс, 2001. – 416 с.
23. ДСТУ ГОСТ 2.001:2006 Єдина система конструкторської документації. Загальні положення (ГОСТ 2.001-93, IDT).
24. ДСТУ ГОСТ 2.051:2006 Єдина система конструкторської документації. Електронні документи. Загальні положення (ГОСТ 2.051-2006, IDT).
25. ДСТУ ISO 5457:2006 (ISO 5457:1999, IDT) Національний стандарт України. Документація технічна на вироби. Кресленики. Розміри та формати.
26. ГОСТ 2.701-2008 ЕСКД. Схемы. Виды и типы. Общие требования к выполнению.
27. ДСТУ ГОСТ 2.702:2013 ЄСКД. Правила виконання електричних схем (ГОСТ 2.702-2011, IDT).

28. *ГОСТ 2.721-74 ЕСКД. Обозначения условные графические в схемах. Обозначения общего применения.*
29. *ГОСТ 2.747-68 ЕСКД. Обозначения условные графические в схемах. Размеры условных графических обозначений.*
30. *ДСТУ 3008-2015 «Державний стандарт України. Документація. Звіти в сфері науки і техніки. Структура і правила оформлення».*

Educational content

5. Methodology

Course work consists of nine stages, which are listed in the table 5.1.

Table 5.1

Semester week	The name of the work stage
3	*Getting a topic and task*
4-5	*Selection and study of literature*
6-7	*Formation of the technical task*
8-9	*Overview of collective software development methods*
10-11	*Development of precedents of the developed software application*
12-13	*Development of a software application*
14	*Software application testing*
15	*Issuance of an explanatory note*
16	*Submission of coursework for review*
17	*Protection of coursework*

6. Self-study

No. in order	***The name of the topic that is submitted for self-study***	***Number of hours of SS***
1	*Obtaining a topic and assignment for a term paper*	*1*
2	*Analysis of the task, selection and study of the literature*	*5*
3	*Development of precedents of the developed software application and development of the software application itself*	*20*
4	*Drawing up an explanatory note to the coursework*	*4*
	In total:	*30*

Policy and Assessment

7. Course policy

The design of the term paper must meet the requirements for reports on research work (ДСТУ 3008-2015 «Державний стандарт України. Документація. Звіти в сфері науки і техніки. Структура і правила оформлення»).

All illustrative material in the course work must be done using computer tools. The content of the illustrative material should sufficiently reflect the main provisions that are being defended.

Both the teacher and the student are obliged to adhere to the Code of Honor of the National Technical University of Ukraine "Kyiv Polytechnic Institute".

The main provisions of the policy:

- the topic of the course work can be coordinated with the topic of the future qualification work of the bachelor;

- the stages of the course work must be completed according to the established calendar schedule of work;

- the developed software application must be tested, the results of testing the software application are given in the text of the main part of the course work;

- in the case of detection of academic dishonesty and plagiarism, the coursework is returned for radical revision with a possible change of topic;

- untimely completion of the stage of the course work entails a reduction of the points received for it by 10% if the delay is no more than two weeks, by 20% if the delay is more than two weeks.

The following factors are taken into account when evaluating coursework:

- *complete completion of an individual coursework assignment;*
- *correctness of developed precedents;*
- *timeliness of coursework according to the schedule;*
- *independent performance of coursework and absence of signs of plagiarism;*
- *answers to questions about the content of the coursework during its defense.*

8. Types of control and rating system for evaluating learning outcomes (RSE)

*The system of evaluating the success of students in the discipline "Coursework in Software Engineering" is based on the "Regulations on the system of evaluating learning results in “Igor Sikorsky Kyiv Polytechnic Institute” (***https://document.kpi.ua/files/2020_1-273.pdf***), Namely, the Rating System of Evaluation (RSE) of the second type (RSE-2).*

RSE-2 for coursework (RK) consists of two components:

- *starting component ($\boldsymbol{R_C}$);*
- *component of protection ($\boldsymbol{R_3}$).*

$$\boldsymbol{R_K = R_C + R_3}$$

The first (starting) component characterizes the student's course work and its result - the quality of the explanatory note and the developed software application. The second component characterizes the quality of the student's defense of the course work.

The size of the scale of the first component equals 80 points, and the second component - 20 points.

The quality of the explanatory note and the degree of compliance with the work schedule

*Weight score – **80** (R_C). The criteria for evaluating the components of the explanatory note are given in Table 8.1.*

Table 8.1 – Criteria for evaluating the implementation of the components of the explanatory note

Stage No	***Composite works***	***The maximum number of points for timely performance***	***Taking into account the timeliness of execution***
1	*Layout of the title page*	*2*	***100% of the grade if the work schedule is***
2	*The availability of a technical task for the coursework*	*2*	
3	*Availability and content of the album description*	*2*	
4	*Availability of content*	*2*	

5	Availability and content of the introduction	2	**followed** **90% in case of delay up to 2 weeks** **80% in delay case of more than 2 weeks**
6	Availability and content of the overview of the methods of collective development of the software application	20	
7	The presence and content of development precedents of the software application	15	
8	Availability and content of the software application	20	
9	Availability and content of software application testing results	5	
10	Availability and content of conclusions	5	
11	Availability and registration of the list of sources	5	
	In total	**80**	

A student is allowed to defend a coursework on the condition that he has an initial R_C component of at least 60% of the maximum value, which is

80 x 0.6 = 48 points.

Quality of protection

*Weight score – **20** (R_Z).*

Evaluation criteria for a presentation based on the materials of the coursework and answers to questions:

– mastery of theoretical material up to 10 points;

– the degree of mastery of software application development methods in total up to 10 points.

The defense of the coursework is considered successful if R_Z is at least 60% of its maximum value, i.e.

20 x 0.6 = 12 points.

After the completion of the defense of the course work, the RK is determined, which is later translated into an assessment on the university scale according to the table:

Scores	Rating
100-95	Perfectly
94-85	Very good
84-75	Fine
74-65	Satisfactorily
64-60	Enough
Less than 60	Unsatisfactorily
Admission conditions not met	Not allowed

Syllabus of the course

Is designed by teacher PhD, Associate Professor, Andrii Antoniuk

Adopted by Department of Computing Technics (protocol № 10, 25 May 2022)

Approved by the Faculty Board of Methodology (protocol № 10, 09 June 2022)