



Компоненти програмної інженерії. Частина 2

Моделювання програмного забезпечення.

Аналіз вимог до програмного забезпечення

Робоча програма навчальної дисципліни (Силабус)

Реквізити навчальної дисципліни

Рівень вищої освіти	Перший (бакалаврський)
Галузь знань	12 Інформаційні технології
Спеціальність	121 Інженерія програмного забезпечення
Освітня програма	Інженерія програмного забезпечення інтелектуальних кібер-фізичних систем в енергетиці
Статус дисципліни	Обов'язкова (нормативна)
Форма навчання	Очна (денна)
Рік підготовки, семестр	2 курс, третій семестр
Обсяг дисципліни	4 кредитів ECTS /120 годин, з яких 54 години аудиторних (36 год лекції, 18 год практичні заняття), 66 годин становить самостійна робота
Семестровий контроль/ контрольні заходи	Залік, модульна контрольна робота, бліц-опитування, календарний контроль
Розклад занять	http://rozklad.kpi.ua/
Мова викладання	Українська
Інформація про керівника курсу / викладачів	Лектор: к.т.н., доцент Гагарін Олександр Олександрович, o.haharin@kpi.ua, Gagarin.info@gmail.com тел.098-048-96-61 Практика: : к.т.н., доцент Гагарін Олександр Олександрович, o.haharin@kpi.ua, Gagarin.info@gmail.com тел. 098-048-96-61
Розміщення курсу	Google classroom, Електронний кампус, платформа дистанційного навчання «Сікорський»

Програма навчальної дисципліни

1. Опис навчальної дисципліни, її мета, предмет вивчення та результати навчання

Навчальна дисципліна «Компоненти програмної інженерії. Частина 2. Моделювання програмного забезпечення Аналіз вимог до програмного забезпечення.», є базовою з циклу професійної підготовки. Програму навчальної дисципліни складено відповідно до освітньо-професійної програми «"Інженерія програмного забезпечення інтелектуальних кібер фізичних систем в енергетиці" першого рівня вищої освіти спеціальності 121 «Інженерія програмного забезпечення» та Стандарту ВО 121 (бакалавр), Наказ МОНУ №1166 від 29.10.2018.

Вивчається протягом 3 го семестру на кафедрі Інженерія програмного забезпечення в енергетиці - ІПЗЕ. Загальна характеристика курсу розподіл часу Всього 4 кредиту 120 годин 3 них 54- аудиторних 36

лекцій 18 практичних робіт 66 самостійна робота студента семестровий контроль знань МКР Підсумковий контроль знань залік

Основні завдання навчальної дисципліни є формування теоретичних знань базових визначень інженерії програмного забезпечення та суті методів, засобів і процесів, що розглядаються в інженерії програмного забезпечення, а також засвоєння студентами технологічних схем та інженерних методик моделювання ПЗ що проектується.

Предметом дисципліни "Компоненти програмної інженерії. Частина 2. Моделювання програмного забезпечення Аналіз вимог до програмного забезпечення.» є формування інформаційного поля з описом основних теоретичних положень процесів: формування вимог до ПЗ що створюється та розробки ПЗ засобами моделювання.

Метою навчальної дисципліни є формування у студентів здатностей:

ідентифікувати, класифікувати та формулювати вимоги до програмного забезпечення (K13: Стандарт ВО 121 (бакалавр), Наказ МОНУ №1166 від 29.10.2018);

брати участь у проектуванні програмного забезпечення, включаючи проведення моделювання (формальний опис) його структур, поведінки та процесів функціонування (K14: Стандарт ВО 121 (бакалавр), Наказ МОНУ №1166 від 29.10.2018).

Курс формує у студентів такі компетентності:

Загальні

- ЗК2.Здатність застосовувати знання у практичних ситуаціях
- ЗК5.Здатність вчитися і оволодівати сучасними знаннями
- ЗК6.Здатність до пошуку, оброблення та аналізу інформації з різних джерел
- ЗК7.Здатність працювати в команді.

Фахові

- ФК1 Здатність ідентифікувати, класифікувати та формулювати вимоги до програмного забезпечення
- ФК2 Здатність брати участь у проектуванні програмного забезпечення, включаючи проведення моделювання (формальний опис) його структури, поведінки та процесів функціонування
- ФК3 Здатність розробляти архітектури, модулі та компоненти програмних систем
- ФК4 Здатність формулювати та забезпечувати вимоги щодо якості програмного забезпечення у відповідності з вимогами замовника, технічним завданням та стандартами
- ФК5 Здатність дотримуватися специфікацій, стандартів, правил і рекомендацій в професійній галузі при реалізації процесів життєвого циклу
- ФК7 Володіння знаннями про інформаційні моделі даних, здатність створювати програмне забезпечення для зберігання, видобування та опрацювання даних.
- ФК 10 Здатність накопичувати, обробляти та систематизувати професійні знання щодо створення тестування і супроводження програмного забезпечення та визнання важливості навчання протягом всього життя

- *ФК 11. Здатність реалізувати фази та ітерації життєвого циклу програмних система інформаційних технологій на основі відповідних моделей і підходів розробки програмного забезпечення*
- *ФК 12 Здатність реалізувати фази та ітерації життєвого циклу програмних система інформаційних технологій на основі відповідних моделей і підходів розробки програмного забезпечення*
- *ФК 13 Здатність здійснювати процес інтеграції системи, застосовувати стандарти і процедури управління змінами для підтримки цілісності, загальної функціональності і надійності програмного забезпечення.*

Згідно з вимогами програми навчальної дисципліни студенти після засвоєння кредитного модуля мають продемонструвати такі результати навчання:

ЗНАННЯ:

- *ПРН1 Аналізувати, цілеспрямовано шукати і вибирати необхідні для вирішення професійних завдань інформаційно-довідникові ресурси і знання з урахуванням сучасних досягнень науки і техніки*
- *ПРН2 Знати кодекс професійної етики, розуміти соціальну значимість та культурні аспекти інженерії програмного забезпечення і дотримуватись їх в професійній діяльності.*
- *ПРН3. Знати основні процеси, фази та ітерації життєвого циклу програмного забезпечення.*
- *ПРН4. Знати і застосовувати професійні стандарти і інші нормативно-правові документи в галузі інженерії програмного забезпечення.*
- *ПРН7 Знати і застосовувати на практиці фундаментальні концепції, парадигми і основні принципи функціонування мовних, інструментальних і обчислювальних засобів інженерії програмного забезпечення*
- *ПРН8 Вміти розробляти людино-машинний інтерфейс*
- *ПРН9. Знати та вміти використовувати методи та засоби збору, формулювання та аналізу вимог до програмного забезпечення.*
- *ПРН10. Проводити передпроектне обстеження предметної області, системний аналіз об'єкта проектування.*
- *ПРН12. Застосовувати на практиці ефективні підходи щодо проектування програмного забезпечення.*
- *ПРН13 Знати і застосовувати методи розробки алгоритмів, конструювання програмного забезпечення та структур даних і знань.*
- *ПРН14 Застосовувати на практиці інструментальні програмні засоби доменного аналізу, проектування, тестування, візуалізації, вимірювань та документування програмного забезпечення*
- *ПРН15 Мотивовано обирати мови програмування та технології розробки для розв'язання завдань створення і супроводження програмного забезпечення*
- *ПРН16. Мати навички командної розробки, погодження, оформлення і випуску всіх видів програмної документації.*

- ПРН17 Вміти застосовувати методи компонентної розробки програмного забезпечення
- ПРН18 Знати та вміти застосовувати інформаційні технології обробки, зберігання та передачі даних
- ПРН23. Вміти документувати та презентувати результати розробки програмного забезпечення
- ПРН26 Вміти використовувати методи інженерії даних.
- типові підходи у організації проектних робіт зі створення програмних систем та продуктів, моделі життєвого циклу програмних систем, сучасні технології інжинірингу програмного забезпечення;

УМІННЯ:

- ПРН11. Вибирати вихідні дані для проектування, керуючись формальними методами опису вимог та моделювання
- ПРН22. Знати та вміти застосовувати методи та засоби управління ІТ-проектами, зокрема в енергетиці
- ПРН23. Вміти документувати та презентувати результати розробки програмного забезпечення.
- ПРН36 Вміти проектувати програмне забезпечення кібер-фізичних та енергетичних систем

2. Пререквізити та постреквізити дисципліни (місце в структурно-логічній схемі навчання за відповідною освітньою програмою)

У структурно-логічній схемі навчання дисципліна «Компоненти програмної інженерії Частина 2 Моделювання програмного забезпечення Аналіз вимог до програмного забезпечення..» розміщена тоді, коли студенти вже прослухали навчальні дисципліни «Компоненти програмної інженерії Частина 1. Вступ до програмної інженерії», «Алгоритми та структури даних», «Основи програмування». що достатньо для виконання практичних робіт з даної дисципліни.

Дисципліна «Компоненти програмної інженерії Частина 2. Моделювання програмного забезпечення. Аналіз вимог до програмного забезпечення» забезпечує вивчення навчальної дисципліни базової підготовки, а саме «Компоненти програмної інженерії Частина 3. Архітектура програмного забезпечення», «Компоненти програмної інженерії. Частина 4. Якість та тестування програмного забезпечення», «Об'єктно-орієнтований аналіз та конструювання програмних систем» які викладаються у наступних семестрах

3. Зміст навчальної дисципліни

Дисципліна “ Компоненти програмної інженерії. Частина 2. Аналіз вимог до програмного забезпечення. Моделювання програмного забезпечення.

РОЗДІЛ 1: АНАЛІЗ ВИМОГ ДО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ.

Тема 1.1. Вступ. Тезаурус курсу. ЕТАПИ РОЗРОБКИ ПЗ. Життєвий цикл ПЗ. Моделі ЖЦПЗ

Тема 1.2. Поняття і класифікація вимог, їх властивості. Процес аналізу вимог.

Тема 1.3. Формування вимог розробки ПЗ. Виявлення вимог.

Тема 1.4. Формування вимог бачення. Як планується проектне завдання.

Тема 1.5. Розширений аналіз вимог. Моделювання та прототипування.

Тема 1.6. Документування та перевірка вимог.

РОЗДІЛ 2. МЕТОДИ ТА ТЕХНОЛОГІЇ МОДЕЛЮВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Тема 2.1. Загальні підходи до моделювання ПЗ.

Тема 2.2. Основи проектування моделі ПЗ від даних (DFD - діаграми)

Тема 2.3. Основи структурного моделювання ПЗ. (SADT - діаграми)

Тема 2.4. Основи проектування бізнес процесів у розробці ПЗ. (IDEF - діаграми)

Тема 2.5. Проектування ПЗ засобами універсальної мови моделювання (UML- діаграми)

Актори Прецеденти Класи Події Взаємодія Поведінка Розміщення

Тема 2.6. Сучасні напрямки розвитку технологій моделювання ПЗ

4. Навчальні матеріали та ресурси

Базова література:

1. Кінах Я.І., Бойко І.В. *Методологія та технологія створення складальних програмних систем : навчально-методичний посібник / Укладачі : Кінах Я.І., Бойко І.В. – Тернопіль : Тернопільський національний технічний університет імені Івана Пулюя, 2017 – 40 с.*
2. Лавріщева К.М. *Програмна інженерія. – К. – 2008. – 319 с. ISBN 978–966–02–5052–9*
3. М.В. Бабенко *Конспект лекцій з дисципліни «Професійна практика програмної інженерії» для здобувачів вищої освіти першого (бакалаврського) рівня зі спеціальності 121 “Інженерія програмного забезпечення” усіх форм навчання / Укладач М.В. Бабенко. – Кам’янське: ДДТУ, 2021. – 88 с.*
4. Бабенко Л.П., Лаврищева Е.М. *Основи програмної інженерії. Посібник. – К.: Знання, 2001. – 269 с.*
5. Elizabeth Hull, Ken Jackson, Dick Jeremy, *“Requirements Engineering”* ДМК Пресс, 2017, 224 с.
6. Leszek A. Maciaszek. *Requirements analysis and system design. Development of information systems using UML : Вільямс, 2002.*
7. Буч, Г. *Введение в UML от создателей языка; пер. с англ. / Г. Буч, Дж. Рамбо, И. Якобсон – ДМК пресс, 2011.*
8. Buch G., Rambo D., Jacobson I. *UML language. User guide.: Пер. с англ. Мухин Н. – М.: ДМК Пресс, 2006. – 496 с.: ил.*

Допоміжна література:

1. *Introduction to Rational Unified Process / F. Kratchen : Вільямс, 2002. – 240 с.*
2. *Alistair Coburn. Modern methods for describing functional requirements for systems. Вільямс, 2009*
3. *Larman K. Application of UML 2.0 and design patterns : Вільямс, 2009. 7.*

Інформаційні ресурси:

1.. Електронний ресурс: *Software Engineering 2014 Curriculum Guidelines for Undergraduate Degree Programs in Software Engineering A Volum.*

3. Інтранет сайт кафедри [Електронний ресурс] – <<http://ipze.kpi.ua/metodichki>

4. Модульне середовище [Електронний ресурс] – < <https://ecampus.kpi.ua/>

5. *Semantic Portal. React Native.* [Електронний ресурс] / <http://semantic-portal.net/react-native> e of the *Computing Curricula Series 23 February 2015.*

6. Електронний ресурс: *Guide to the Software Engineering Body of Knowledge Version 3.0 SWEBOOK® A Project of the IEEE Computer Society.*

7. Електронний ресурс: <http://www.intuit.ru/studies/courses/2188/174/info>

Karl E. Wiegers Software Requirements Two-time winner of the Software Development Productivity Award MJ vsoft Press

Навчальний контент

5. Методика опанування навчальної дисципліни(освітнього компонента)

№ з/п	Назва теми лекції та перелік основних питань (перелік дидактичних засобів, посилання на літературу та завдання на СРС)
<i>РОЗДІЛ 1: АНАЛІЗ ВИМОГ ДО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ</i>	
<i>Тема 1.1. Вступ. Тезаурус курсу. ЕТАПИ РОЗРОБКИ ПЗ. Життєвий цикл ПЗ. Моделі ЖЦПЗ</i>	
1	<i>Вступ. Організаційні питання. Тезаурус курсу. Етапи розробки ПЗ .Моделі життєвого циклу ПЗ Визначення поняття вимоги. [1] с. 53-87. Завдання на СРС: [3] с. 47-89.</i>
<i>Тема 1.2. Поняття і класифікація вимог, їх властивості. Процес аналізу вимог.</i>	
2	<i>Класифікація вимог. Вимоги до продукту і процесу Рівні вимог. Системні вимоги та вимоги до програмного забезпечення. Функціональні, нефункціональні вимоги і характеристики продукту. Властивості вимог. Повнота. Ясність Коректність і узгодженість (несуперечність). Здатність до верифікації. Необхідність і корисність при експлуатації. Здійсненність. Відстежуваність. Впорядкованість за важливістю і стабільністю. Наявність кількісної метрики. [5] с. 127-149. Завдання на СРС: [8] с.127-149.</i>
<i>Тема 1.3.Формування вимог розробки ПЗ. Виявлення вимог</i>	
3	<i>Виявлення вимог</i>

	<p><i>Аналіз вимог, аналіз проблемної області, бізнес-аналіз. Методології бізнес-аналізу. Вимоги та архітектура ПЗ. Аналіз вимог і інші робочі потоки програмної інженерії. Джерела вимог.</i></p> <p><i>[1] с. 127-149.</i></p> <p><i>Завдання на СРС: [1] с.127-149.</i></p>
<p><i>Тема 1.4. Формування вимог бачення. Як планується проектне завдання.</i></p>	
4	<p><i>Бачення продукту і межі проекту. Концепція в стандартах. Бачення в RUP. Бачення / рамки MSF. Бачення продукту і межі проекту. Актори і варіанти використання. Глосарій. Специфікація варіантів використання. Вільний формат.</i></p> <p><i>[1] с. 149-1685.</i></p> <p><i>Завдання на СРС: [1] с.149-168.</i></p>
<p><i>Тема 1.5. Розширений аналіз вимог. Моделювання та прототипування.</i></p>	
5	<p><i>Які моделі використовувати. Моделі UML, що пояснюють функціональність системи. Діаграма варіантів використання. Діаграма дій, діаграма станів. Діаграми UML, що пояснюють внутрішню будову системи. Альтернативні мови моделювання. Діаграма потоків даних. Інші види моделей. Мета прототипування. Класифікація прототипів Горизонтальний і вертикальний прототипи.</i></p> <p><i>[1] с. 169-186.</i></p> <p><i>Завдання на СРС: [1] с.169-186.</i></p>
<p><i>Тема 1.6. Документування та перевірка вимог.</i></p>	
6	<p><i>Документування та перевірка вимог. Документування вимог у відповідність з стандартами. Опис вимог до системи у відповідність з стандартами. Документування вимог в RUP. Документування вимог на основі IEEE Standard 830-1998. Документування вимог MSF Верифікація і валідація. Двозначність вимог.</i></p> <p><i>[1] с. 187-202.</i></p> <p><i>Завдання на СРС: [1] с.187-202.</i></p>
<p>РОЗДІЛ 2. МЕТОДИ ТА ТЕХНОЛОГІЇ МОДЕЛЮВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ</p>	
<p><i>Тема 2.1. Загальні підходи до моделювання ПЗ.</i></p>	
7	<p><i>Основні підходи та концепції моделювання. Принципи моделювання. Моделювання від даних. Моделювання структурними елементами. Об'єктне моделювання. Вступ до UML. Концептуальна модель UML. Діаграми UML.</i></p> <p><i>[7] с. 17-132.</i></p> <p><i>Завдання на СРС: [7] с. 17-132.</i></p>
<p><i>Тема 2.2. Основи проектування моделі ПЗ від даних (DFD - діаграми)</i></p>	

8	<p>Моделювання ПЗ на базі процесу перетворення даних. Виявлення потоків. Фіксація джерел даних Перетворювачів, Сховища даних. Нотація DFD - діаграм [7] с. 133-220. Завдання на СРС: [7], с. 133-220.</p>
<p>Тема 2.3. Основи структурного моделювання ПЗ. (SADT - діаграми)</p>	
9	<p>Моделювання ПЗ на базі виявлення структурних елементів. Структурний елемент - процес перетворення даних. Характеристики процесу перетворення. Виявлення інформаційних та управляючих потоків. Перетворювачів, Сховища даних. Нотація SADT- діаграм. [7] с. 221-300. Завдання на СРС: [7], с. 221-300.</p>
<p>Тема 2.4. Основи проектування бізнес процесів у розробці ПЗ. (IDEF - діаграми)</p>	
10	<p>Бізнес процес у програмуванні. Моделювання ПЗ та виявлення бізнес процесів у проектуванні ПЗ. Нотація IDEFx- діаграм. [7] с. 301-365. Завдання на СРС: [7], с. 301-365.</p>
<p>Тема 2.5 Проектування ПЗ засобами універсальної мови моделювання (UML- діаграми)</p>	
11	<p>Вступ до UML. Концептуальна модель UML. Діаграми UML. Моделювання класів, об'єктів та екземплярів: Класи; [7] с. 367-448. Завдання на СРС: [7], с. 367-448.</p>
12	<p>Основні механізми доповнення і розширення будівельних блоків UML; Розширені класи. Класифікатори; Екземпляри. Моделювання зв'язків та відношень: Типові зв'язки; Розширені зв'язки; Інтерфейси, типи, ролі. [7] с. 367-448. Завдання на СРС: [7], с. 367-448.</p>
<p>Тема 2.6 .Сучасні напрямки розвитку технологій моделювання ПЗ</p>	
13	<p>Методики проектування на основі шаблонів та тестування Шаблони проектування GRASP: експерт, творець. Огляд шаблонів GOF. Породжуючі: абстрактна фабрика, фабричний метод [7] с. 367-448. Завдання на СРС: [7], с. 367-448.</p>

№ з/п	Назва практичної роботи	Кількість ауд. годин
1	Організаційні та технологічні питання. Визначення тематики проблемних областей	1
2	Виявлення високорівневих вимог і розробка документу "Бачення"	2
3	Розширений набір вимог користувачів, акторів і варіантів використання	2
4	Модель ПЗ з використанням діаграми UML. Короткий опис варіантів використання та виявлення базових класів	2
5	Модель ПЗ з використанням діаграми UML. Детальний опис ключових варіантів використання	2
6	Діаграмні технології потоків даних	2
7	Структурні елементи для виявлення архітектури ПС	2
8	Технологія RAD розробки	2
9	Аналіз і специфікація спеціальних вимог у моделі ПС. Документування компонентів опису ПС	2

6. Самостійна робота студента/аспіранта

№ з/п	Назва теми, що вноситься на самостійне опрацювання	Кількість годин СРС
1	Розділ 1, Тема 1.1. Вступ. Поняття і класифікація вимог: Визначення ІС. Класифікація ІС. Класифікація за масштабом. Класифікація по архітектурі. Класифікація за характером використання інформації. Класифікація по системі подання даних. Класифікація за підтримуваним стандартам управління і технологій комунікації. Класифікація за ступенем автоматизації. Завдання на СРС: [1] с. 103-126.	3
2	Розділ 1, Тема 1.2. Вимоги та їх властивості. Процес аналізу вимог: Яких вимог не повинно бути. Процес аналізу вимог. Робочий потік аналізу вимог. Хто створює та використовує вимоги. Організація роботи з вимогами на прикладі MSF. Завдання на СРС: [1] с.127-149.	3
3	Розділ 1, Тема 1.3. Контекст завдання аналізу вимог. Виявлення вимог:	3

	<i>Стратегії виявлення вимог. Інтерв'ю, Анкетування. Спостереження. Самостійний опис вимог. Спільні семінари. Прототипування. Завдання на СРС: [1] с.127-149.</i>	
4	<i>Розділ 1, Тема 1.4. Формування бачення. Специфікація вимог: Шаблон повного опису варіанти використання за А. Коберну. Табличні уявлення варіанти використання. Шаблон варіанти використання RUP. Вибір форми опису варіанти використання. Специфікація функціональних вимог. Атрибути вимог. Завдання на СРС: [1] с.149-165.</i>	3
5	<i>Розділ 1, Тема 1.5. Розширений аналіз вимог. Моделювання та прототипування: Одноразовий і еволюційні прототипи. Паперовий прототип. Розкадровка. Ілюстровані сценарії прецедентів. Орієнтири. Середні значення атрибутів об'єктів та обсяги. Середня інтенсивність використання. Завдання на СРС: [1] с.169-186.</i>	3
6	<i>Розділ 1, Тема 1.6. Документування та перевірка вимог: "Золочення" продукту. Мінімальна специфікація. Пропуск типів користувачів. Методи і засоби перевірки вимог. Неофіційні перегляди вимог. Інспекції. Розробка тестів. Визначення критеріїв прийнятності. Завдання на СРС: [1] с.187-202.</i>	3
7	<i>Розділ 1, Тема 1.7. Управління вимогами: Управління незапланованим зростанням обсягу. Процес контролю змін. Аналіз впливу зміни. Відстежуваність вимог. Завдання на СРС: [1] с.447-468.</i>	3
8	<i>Розділ 1, Тема 1.8. Удосконалення процесів роботи з вимогами: Створення та апробація нових процесів. Оцінювання результатів і прийняття рішень. Завдання на СРС: [1] с.469-492.</i>	3
9	<i>Розділ 1, Тема 1.9. Вимоги в управлінні проектами: Сучасні тенденції в розвитку АІС і технологій їх створення. Покупне або замовне ПЗ - критерії вибору. Стратегії вибору рішення. Аналіз вимог. Аналіз невідповідності. Підхід на основі найкращих практик. Процес вибору рішення. Завдання на СРС: [1] с.469-492.</i>	3
10	<i>Розділ 2, Тема 2.1. Загальні підходи до моделювання ПЗ: Правила UML. Завдання на СРС: [7] с. 17-132.</i>	3

11	<i>Розділ 2, Тема 2.2. Основи структурного моделювання ПЗ. Діаграми для моделювання статичних характеристик системи: Основні моделі статичних представлень системи; Діаграми класів; Діаграми об'єктів; Компоненти та інтерфейси. Діаграми компонентів. Завдання на СРС: [7], с. 133-220.</i>	3
12	<i>Розділ 2, Тема 2.3. Основи моделювання поведінки ПЗ: Моделювання взаємодії об'єктів: Стратегія моделювання взаємодії як обміну поведінками між об'єктами; Діаграми взаємодії; Діаграми діяльності. Завдання на СРС: [7], с. 221-300.</i>	3
13	<i>Розділ 2, Тема 2.4. Основи моделювання подій: Моделювання систем реального часу. Діаграми станів. Завдання на СРС: [7], с. 301-365.</i>	3
14	<i>Розділ 2, Тема 2.5. Основи моделювання архітектури ПЗ: Моделювання пакетів як спосіб організації елементів моделі. Завдання на СРС: [7], с. 367-448.</i>	3
15	<i>Залік</i>	6

Політика та контроль

1. Політика навчальної дисципліни (освітнього компонента)

Система вимог, які викладач ставить перед студентом:

- *Кодекс честі: <http://kpi.ua/code>;*
- *Правила внутрішнього розпорядку: <http://kpi.ua/admin-rule>;*
- *Положення про організацію освітнього процесу в КПІ ім. Ігоря Сікорського: <https://kpi.ua/regulations>.*

2. Види контролю та рейтингова система оцінювання результатів навчання (PCO)

Поточний контроль: експрес-опитування, опитування за темою заняття, МКР

Календарний контроль: проводиться двічі на семестр як моніторинг поточного стану виконання вимог Силабусу. Модульна контрольна робота складається з двох контрольних робіт з розділів:

1. *Аналіз вимог до програмного забезпечення. [1], с. 103-202, 447-492.*
2. *Моделювання програмного забезпечення. [7], с. 17-448.*

Семестровий контроль: Залік

Умови допуску до семестрового контролю: мінімально позитивна оцінка за індивідуальні завдання /зарахування усіх практичних робіт/ семестровий рейтинг більше 65 балів.

Максимальна оцінка за кожну практичну роботу складає 7 балів. Максимальна кількість балів за всі практичні роботи складає 60 балів.

Критерії оцінювання:

Підготовка до роботи (у відсотках від максимальної кількості балів за відповідну роботу):

- *протокол відповідає вимогам, охайний – 20 %;*
- *протокол відповідає вимогам, але є чисельні виправлення – 10 %.*

Виконання практичної роботи:

- робота виконана повністю і вірно протягом відведеного часу – 50 %;
- робота виконана пізніше зазначеного терміну – 20 %.

Якість захисту роботи:

- студент вірно і повністю відповів на запитання – 30 %;
- студент при відповіді допустив несуттєві неточності – 20 %;
- студент при відповіді на запитання допустив суттєві неточності, але самостійно виправив їх – 10 %.

2. Максимальна кількість балів за модульну контрольну роботу – 30 балів.

Критерії оцінювання такі самі як для практичних робіт.

3. Усний залік – максимальний ваговий бал 40.

Розрахунок шкали рейтинг $y(R)$.

Сума вагових балів контрольних заходів протягом семестру складає:

$$R = 45 + 10 + 45 = 100 \text{ балів.}$$

Таким чином, рейтингова шкала з дисципліни складає $R = 100$ балів.

Необхідною умовою допуску до заліку є зарахування усіх практичних робіт, а також стартовий рейтинг (rC) не менше 40% від R , тобто 40 балів. Для отримання студентом відповідних оцінок його рейтингова оцінка RD переводиться згідно з наступною таблицею.

Таблиця відповідності рейтингових балів оцінкам за університетською шкалою:

Кількість балів	Оцінка
100-95	Відмінно
94-85	Дуже добре
84-75	Добре
74-65	Задовільно
64-60	Достатньо
Менше 60	Незадовільно
Не виконані умови допуску	Не допущено

3. Додаткова інформація з дисципліни (освітнього компонента)

- Перелік питань, які виносяться на семестровий контроль:
 - Місце аналізу вимог у життєвому циклі ПЗ.
 - У чому складність процесу збору вимог до ПЗ?
 - Навіщо потрібен аналіз зібраних вимог до ПЗ?
 - Коли починається процес збору вимог?
 - Перелічіть основні типи вимог та назвіть їх характеристики.
 - Залежність вимог до ПЗ від цільового сегмента ринку.
 - Для чого і як формуються профілі користувачів?
 - Техніки вилучення вимог.
 - Що повинні включати сценарії використання програмного забезпечення?
 - Збір інформації про аналоги.
 - Структура таблиць звітності про аналоги.

- *Для чого створюється образ продукту?*
- *Яка інформація повинна входити в опис образу продукту?*
- *Застосування SWOT аналізу в процесі визначення образу продукту.*
- *Управління масштабом складних проектів.*
- *Визначення змісту проекту.*
- *Які існують способи написання специфікацій вимог до ПЗ?*
- *Для чого використовуються діаграми прецедентів?*
- *Створіть документ-концепцію «Віртуальної торгової площадки».*
- *Призначення діаграми варіантів використання.*
- *Для чого служить відношення включення?*
- *Пояснити сенс відношення розширення.*
- *Дайте визначення основному потоку подій.*
- *Дайте визначення альтернативному потоку подій.*
- *Що таке передумови і постумови?*
- *Дайте визначення точці розширення.*
- *Моделювання взаємодії об'єктів.*
- *Стратегія моделювання взаємодії як обміну поведінками між об'єктами.*
- *Діаграми взаємодії.*
- *Діаграми діяльності.*
- *Моделювання систем реального часу. Діаграми станів.*
- *Основи моделювання архітектури ПЗ.*
- *Моделювання пакетів як спосіб організації елементів моделі.*

Робочу програму навчальної дисципліни (Силабус): Компоненти програмної інженерії 2. Аналіз вимог до програмного забезпечення Моделювання програмного забезпечення.

Складено доцентом кафедри ІПЗЕ, к.т.н., доц., Гагарін Олександр Олександрович

Ухвалено кафедрою інженерії програмного забезпечення в енергетиці НН ІАТЕ (протокол № 34 від 10.05.2024 р.)

Погоджено Методичною комісією НН ІАТЕ (протокол № 9 від 31.05.2024 р.)