



СИЛАБУС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

«ТЕХНОЛОГІЯ ПРОГРАМУВАННЯ СКЛАДНИХ СИСТЕМ»

Галузь знань			15 – «Автоматизація та приладобудування»		Освітній рівень		бакалавр	
Спеціальність			151 – «Автоматизація та комп'ютерно-інтегровані технології»		Семестр		Повний денне/заочне	
							7,8/9	
							Прискорений	
							4/5	
Освітньо-професійна програма			Автоматизація та комп'ютерно-інтегровані технології		Тип дисципліни		Обов'язкова	
Факультет			Машинобудування		Кафедра		Автоматизація виробничих процесів	
Обсяг:	Кредитів ECTS	Годин	За видами занять (денне/заочне)					
			Лекцій	Практичних занять	Самостійна підготовка	Вид контролю		
	7,5	225	30/12	30/12	120/156	Іспит		
			Курсова робота (денне/заочне)					
			Лекцій	Практичних занять	Самостійна підготовка	Вид контролю		
				26/8	19/37	Діф. залік		
Обсяг:	Кредитів ECTS	Годин	За видами занять (денне/заочне) прискорений курс					
			Лекцій	Практичних занять	Самостійна підготовка	Вид контролю		
	3	90	26/12	26/4	38/74	Іспит		

ВИКЛАДАЧІ

Картамишев Дмитро Олександрович, ауд. 2206, , e-mail: dmytro.kartamyshev@gmail.com



Кандидат технічних наук, асистент кафедри АВП ДДМА.

Наукові праці та навчально-методичні посібники:

ORCID: <https://orcid.org/0000-0003-3240-8919>

Scopus Author ID: 57196149104

<https://www.scopus.com/authid/detail.uri?authorId=57196149104>

Провідний лектор з дисциплін: «Технологія програмування складних систем», «Інженерія програмного забезпечення», «Комп'ютерні технології та програмування»

АНОТАЦІЯ КУРСУ

Взаємозв'язок у структурно-логічній схемі

Освітні компоненти, які передують вивченню	Комп'ютерна технології та програмування, Технічні засоби автоматизації, Проектування систем управління на базі ПЛК, Проектування систем автоматизації
Освітні компоненти для яких є базовою	Кваліфікаційна робота бакалавра

Компетенції відповідно до освітньо-професійної програми

Soft- skills / Загальні компетентності (ЗК)	Hard-skills / Спеціальні (фахові) компетенції
<ul style="list-style-type: none"> – Здатність застосовувати знання у практичних ситуаціях. – Здатність до пошуку, опрацювання та аналізу інформації з різних джерел. – Здатність діяти свідомо та соціально-відповідально за результати прийняття стратегічних рішень. – Здатність до навчання та саморозвитку. 	<ul style="list-style-type: none"> – Здатність використовувати для вирішення професійних завдань новітні технології у галузі автоматизації та комп'ютерно-інтегрованих технологій, зокрема, проектування багаторівневих систем керування, збору даних та їх архівування для формування бази даних параметрів процесу та їх візуалізації за допомогою засобів людино-машинного інтерфейсу. – Здатність вільно користуватись сучасними комп'ютерними та інформаційними технологіями для вирішення професійних завдань, програмувати та використовувати прикладні та спеціалізовані комп'ютерно-інтегровані середовища для вирішення задач автоматизації. – Здатність оформляти отримані робочі результати у вигляді презентацій, науково-технічних звітів.

Результати навчання відповідно до освітньо-професійної (програмні результати навчання – ПРН)

- Вміти застосовувати сучасні інформаційні технології та мати навички розробляти алгоритми та комп'ютерні програми з використанням мов високого рівня та технологій об'єктно-орієнтованого програмування, створювати бази даних та використовувати інтернет-ресурси.
- Вміти проектувати багаторівневі системи керування і збору даних для формування бази параметрів процесу та їх візуалізації за допомогою засобів людино-машинного інтерфейсу, використовуючи новітні комп'ютерно-інтегровані технології
- Оцінювати ризики та здійснювати запобіжні дії їх уникнення, вести професійну діяльність з урахуванням доброчесності та авторського права.
- Усвідомлювати необхідність навчання та саморозвитку продовж усього життя з метою поглиблення знань .
- Вміти оцінювати отримані результати та аргументовано захищати прийняті рішення.

ЗАГАЛЬНА ІНФОРМАЦІЯ

Анотація	Актуальність вивчення дисципліни «Технологія програмування складних систем» у зв'язку із завданням освітньо-професійної підготовки бакалаврів за спеціальністю 174 – «Автоматизація, комп'ютерно-інтегровані технології та робототехніка» зумовлена необхідністю розробки комплексних програмних систем для автоматизації управління на промислових підприємствах
Мета	отримання загальних відомостей і орієнтація студентів в сутності такої області діяльності, як створення прикладного програмного забезпечення
Формат	Лекції (очний, дистанційний формат), практичні заняття (очний, дистанційний формат), консультації (очний, дистанційний формат), підсумковий контроль – іспит (очний, дистанційний формат); Курсова робота (денна повна/заочна повна): практичні заняття (очний, дистанційний формат), консультації (очний, дистанційний формат), підсумковий контроль – діф. залік (очний, дистанційний формат),

**«Правила
гри»**

- Курс передбачає роботу в колективі.
 - Середовище в аудиторії є дружнім, творчим, відкритим до конструктивної критики.
- Політика щодо дедлайнів та перескладання**
- Освоєння дисципліни передбачає обов'язкове відвідування лекцій і практичних занять, а також самостійну роботу.
 - Самостійна робота включає в себе теоретичне вивчення питань, що стосуються тем лекційних занять, які не ввійшли в теоретичний курс, або ж були розглянуті коротко, їх поглиблена проробка за рекомендованою літературою.
 - Усі завдання, передбачені програмою, мають бути виконані у встановлений термін. Якщо студент відсутній з поважної причини, він презентує виконані завдання під час самостійної підготовки на консультації викладача.
- Політика академічної доброчесності**
- Під час роботи над завданнями не допустимо порушення академічної доброчесності: при використанні Інтернет ресурсів та інших джерел інформації студент повинен вказати джерело, використане в ході виконання завдання.
 - Політика академічної доброчесності регламентується «ПОЛОЖЕННЯ про академічну доброчесність науково-педагогічних, наукових, педагогічних працівників та здобувачів вищої освіти ДДМА» (<http://surl.li/laufq>)

СТРУКТУРА ДИСЦИПЛІНИ

Лекція 1	Визначення і основні поняття програмної інженерії.	Практична робота 1	Вивчення архітектури, візуальних інтерфейсів та інструментальних засобів CASE-системи Visual Paradigm Community Edition	Самостійна робота	Поняття, ролі, методології
Лекція 2	Життєвий цикл програмного забезпечення.	Практична робота 2	Аналіз вимог та розробка UML – діаграм концептуального рівня проектування програмної системи		Визначення, етапи, моделі, екстремальне програмування, SCRUM
Лекція 3	Командна робота та методи керування над програмним проектом.	Практична робота 3	Розробка UML – діаграм логічного рівня проектування програмної системи: моделювання статичних аспектів		Методи керування, рольові групи, мотивація програміста, СРМ
Лекція 4	Об’єктна модель.	Практична робота 4	Розробка UML – діаграм логічного рівня проектування програмної системи: моделювання динамічних аспектів		Об’єкти, інтерфейси, нотація класів в UML
Лекція 5	Конфігураційне керування	Практична робота 5	Розробка UML – діаграм фізичного рівня проектування програмної системи		Конфігурація ISO/IEC12207, git, baseline
Лекція 6	Інженерія вимог.	Практична робота 6	Розробка діаграми класів та специфікацій інтерфейсів в Visual Paradigm Community Edition.		Стейкхолдери. Типи вимог. Трасування вимог
Лекція 7	Архітектура програмного забезпечення.	Практична робота 7	Архітектура та дизайн системи.		Архітектура бази даних, клієнт-серверна архітектура, REST, peer-to-peer, мікросервіси
Лекція 8	Варіанти використання(Use Cases).	Практична робота 8	Використання патернів проектування.		Актори. Use Case. Traceability matrix.
Лекція 9	Аналіз предметної області та моделювання.				Діаграма класів аналізу
Лекція 10	Проектування: розподіл відповідальностей.				Діаграма послідовностей.
Лекція 11	Принципи об’єктно-орієнтованого проектування.				SOLID
Лекція 12	Визначення систем.				Стани. Мікростани і макростани. Події. Контекстні діаграми.
Лекція 13	Діаграми станів та об’єктна мова обмежень OCL. Діаграми автоматів UML.				OCL
Лекція 14	Архітектурне проектування.				Діаграми пакетів. Діаграми компонентів.
Лекція 15	Детальне проектування.				Діаграми діяльності. Діаграми розгортання.

МАТЕРІАЛЬНО-ТЕХНІЧНЕ ЗАБЕЗПЕЧЕННЯ ДИСЦИПЛІНИ

Комп’ютери AMD Ryzen 5-3400 (15 од.). Принтер Ecosys P2235dn, Сканер EpsonPerfection V19, Графічний планшет Wacom One Medium (CTL-672-N), Проектор Epson EHTW5820, Екран Walfix 120
 Пакети прикладних програм (тільки ліцензоване та відкрите ПЗ): Visual Paradigm Community Edition
 Система дистанційного навчання і контролю Moodle – <http://moodle-new.dgma.donetsk.ua/course/view.php?id=268>

ІНФОРМАЦІЙНЕ ЗАБЕЗПЕЧЕННЯ ДИСЦИПЛІНИ

Основна література	<ol style="list-style-type: none"> 1. Pressman, Roger, Maxim, Bruce. Software Engineering: A Practitioner's Approach. – NY: McGraw-Hill Education, 2019. – 705p. 2. Sommerville, Ian. Software Engineering, 10th Edition. - Pearson, 2016. – 811p. 3. Voorhees, David. Guide to Efficient Software Design An MVC Approach to Concepts, Structures, and Models. - Springer Nature Switzerland AG, 2020. – 519p. 4. Bruegge, Bernd, Dutoit, Allen. Object-oriented software engineering : using UML, Patterns, and Java. - Harlow, UK: Pearson Education Limited, 2014. – 723 p. 5. Gaopande, Laxmidhar. Software Engineering: A Practical Approach. – 2020.- 241p. 6. Marsic, Ivan. Software Engineering. - Rutgers University, New Brunswick, New Jersey, 2012. – 627p. 7. Бородкіна, Ірина, Бородкин, Георгій. Інженерія програмного забезпечення. Посібник для студентів вищих навчальних закладів.- К.: ТОВ «Видавництво "Центр навчальної літератури"», 2018. – 204 с. 8. Лавріщева К.М. Програмна інженерія.-К.– 2008.–319 с. 9. Кучерук Г. І., Чумаченко В. М., Яковлев В. С. Візуальне моделювання з UML. – Київ: Наукова думка, 2008. – 256 с. 10. Сахацький О. Л. Об'єктно-орієнтоване моделювання програмних систем. – Львів: Новий Світ-2000, 2007. – 318 с. 	Додаткові джерела	<ol style="list-style-type: none"> 1. Booch G., Rumbaugh J., Jacobson I. The Unified Modeling Language User Guide. 2nd ed. – Boston: Addison-Wesley Professional, 2005. – 496 p. 2. Fowler M. UML Distilled: A Brief Guide to the Standard Object Modeling Language. 3rd ed. – Boston: Addison-Wesley Professional, 2003. – 208 p. 3. Larman C. Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development. 3rd ed. – Upper Saddle River: Prentice Hall, 2004. – 736 p. 4. Ambler S. W., Lines M. Agile Modeling: Effective Practices for eXtreme Programming and the Unified Process. – New York: John Wiley & Sons, 2002. – 400 p. <p>Web-ресурси</p> <ol style="list-style-type: none"> 1. https://www.lucidchart.com/ 2. https://plantuml.com/ 3. https://www.diagrams.net/ 4. https://creately.com/lp/uml-diagram-tool/
--------------------	---	-------------------	---

**ГРАФІК НАВЧАЛЬНОГО ПРОЦЕСУ ТА КОНТРОЛЮ ЗНАТЬ І ПЕРЕЗДАЧ З ДИСЦИПЛІНИ ДЛЯ СТУДЕНТІВ
ПОВНОГО КУРСУ НАВЧАННЯ**

Денна повна форма навчання																	
Вид навчальних занять / контролю	Розподіл між учбовими тижнями (1 семестр)																
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15		
Лекції	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2		
Практичні заняття	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2		
Сам. робота	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8		
Консультації						К											К
Контр. роботи							КР1										КР2
Змістовні модулі	ЗМ1						ЗМ2										
Контроль по модулю	ПР1	ПР2		ПР3		ПР4		ПР5		ПР6		ПР7		ПР8			
Вид навчальних занять / контролю	Розподіл між учбовими тижнями (2 семестр)																
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
Лекції																	
Курсова робота	2	2	2	2	2	2	2	2	2	2	2	2	2				
Сам. робота	1	1	1	1	1	1	1	2	2	2	2	2	2				
Консультації																	
Контр. роботи																	
Змістовні модулі																	

Денна прискорена форма навчання																	
Вид навчальних занять / контролю	Розподіл між учбовими тижнями																
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15		
Лекції	2	2	2	2	2	2	2	2	2	2	2	2	2				
Лабораторне заняття	2	2	2	2	2	2	2	2	2	2	2	2	2				
Сам. робота	3	3	3	3	3	3	3	3	3	3	3	3	3				
Консультації				К					К								
Контр. роботи							КР1						КР2				
Змістовні модулі	ЗМ1						ЗМ2										
Контроль по модулю	ПР1	ПР2		ПР3		ПР4		ПР5		ПР6		ПР7					

ПЕРЕЛІК ОБОВ'ЯЗКОВИХ КОНТРОЛЬНИХ ТОЧОК ДЛЯ ОЦІНЮВАННЯ ЗНАНЬ ТА ВМІНЬ

№ з/п	Назва і короткий зміст контрольного заходу	Max балів	Характеристика критеріїв досягнення результатів навчання для отримання максимальної кількості балів
1	2	3	4
1	Вивчення архітектури, візуальних інтерфейсів та інструментальних засобів CASE-системи Visual Paradigm Community Edition	8	Студент здатний продемонструвати глибоке розуміння архітектури та візуальних інтерфейсів CASE-системи Visual Paradigm Community Edition, а також ефективно використовувати її інструментальні засоби для моделювання
2	Аналіз вимог та розробка UML – діаграм концептуального рівня проектування програмної системи	8	Студент здатний продемонструвати знання та вміння у аналізі вимог та розробці концептуальних UML-діаграм, використовуючи ці засоби для відображення основних концепцій та вимог системи.
3	Розробка UML – діаграм логічного рівня проектування програмної системи: моделювання статичних аспектів	8	Студент демонструє розуміння статичних структур програмної системи через створення UML-діаграм логічного рівня, включаючи діаграми класів та пакетів, які відображають взаємозв'язки між компонентами.
4	Розробка UML – діаграм логічного рівня проектування програмної системи: моделювання динамічних аспектів	8	Студент володіє навичками моделювання динамічних аспектів програмної системи, включаючи створення діаграм послідовностей та діаграм активностей, що ілюструють поведінкові сценарії та взаємодії в системі.
5	Контрольна робота 1 за лекційним матеріалом	18	Студент відповів на всі питання контрольної роботи з лекційного матеріалу
6	Розробка UML – діаграм фізичного рівня проектування програмної системи	8	Студент здатний застосувати навички для створення UML-діаграм фізичного рівня, які відображають розподіл компонентів системи по різних обладнаннях та серверах, а також взаємодії між ними.
7	Розробка діаграми класів та специфікацій інтерфейсів в Visual Paradigm Community Edition.	8	Студент ефективно використовує Visual Paradigm Community Edition для створення діаграм класів та специфікації інтерфейсів, продемонструвавши здатність детально моделювати структуру та інтерфейси системи.
8	Архітектура та дизайн системи	8	Студент демонструє розуміння основних принципів архітектури та дизайну системи, включаючи здатність інтегрувати різні архітектурні стилі та патерни у процесі проектування
9	Використання патернів проектування.	8	Студент здатний ідентифікувати та застосовувати відповідні патерни проектування для вирішення конкретних задач розробки, показуючи розуміння їх важливості у поліпшенні гнучкості та повторного використання коду.
10	Контрольна робота 2 за лекційним матеріалом	18	Студент відповів на всі питання контрольної роботи з лекційного матеріалу
Поточний контроль		100 (x0,5)	Студент виконав всі контрольні точки, навів аргументовані відповіді на завдання, що відповідають програмним результатам успішного навчання з дисципліни
Підсумковий контроль (іспит)		100 (x0,5)	Студент виконав тестові завдання та навів аргументовані відповіді на ситуаційні завдання, що відповідають програмним результатам успішного навчання з дисципліни «Технологія програмування складних систем»
Всього		100	

СИСТЕМА ОЦІНКИ			
Сума балів	Оцінка	ECTS	Рівень компетентності
90-100	Відмінно (зараховано)	A	Високий Повністю забезпечує вимоги до знань, умінь і навичок, що викладені в робочій програмі дисципліни. Власні пропозиції студента в оцінках і вирішенні практичних задач підвищує його вміння використовувати знання, які він отримав при вивченні інших дисциплін, а також знання, набуті при самостійному поглибленому вивченні питань, що відносяться до дисципліни, яка вивчається.
81-89	Добре (зараховано)	B	Достатній Забезпечує студенту самостійне вирішення основних практичних задач в умовах, коли вихідні дані в них змінюються порівняно з прикладами, що розглянуті при вивченні дисципліни
75-80		C	Достатній Конкретний рівень, за вивченим матеріалом робочої програми дисципліни. Додаткові питання про можливість використання теоретичних положень для практичного використання викликають утруднення
65-74	Задовільно (зараховано)	D	Середній Забезпечує достатньо надійний рівень відтворення основних положень дисципліни
65-64		E	Середній Є мінімально допустимим у всіх складових навчальної програми з дисципліни
30-54	Незадовільно (не зараховано)	FX	Низький Не забезпечує практичної реалізації задач, що формуються при вивченні дисципліни
0-29		F	Незадовільний Студент не підготовлений до самостійного вирішення задач, які окреслює мета та завдання дисципліни

Силабус за змістом повністю відповідає робочій програмі навчальної дисципліни

Опитування з приводу оцінювання якості викладання дисципліни


Якість викладання дисциплін контролюється анонімним он-лайн-опитуванням студентів. Вивчається думка здобувачів вищої освіти відносно якості викладання дисциплін.

Необхідно оцінити вказані якості за шкалою: 1 бал – якість відсутня; 2 бали – якість проявляється зрідка; 3 бали – якість проявляється на достатньому рівні; 4 бали – проявляється часто; 5 балів – якість проявляється практично завжди.


Анкета є анонімною. Відповіді використовуються в узагальненому вигляді.

https://docs.google.com/forms/d/e/1FAIpQLScRkbRi84CDLtuC_29icduAgVbz0pXSba6VIncjp7X11xQ3Cw/viewform?usp=sharing

Розробник:

 /Дмитро КАРТАМИШЕВ /
« 2 » квітня 2024 р.


Гарант освітньої програми:

 /Олексій РАЗЖИВІН /
«08» травня 2024 р..

Розглянуто і схвалено на засіданні
кафедри АВП

Протокол №13 від 06 травня 2024 р.

Завідувач кафедри

 /Олег МАРКОВ/

Затверджую:

Декан факультету

Машинобудування

 /Валерій КАССОВ/

« 27 » травня 2024 р.

