

	<p style="text-align: center;"><b>Силабус навчальної дисципліни</b> <b><u>«Основи програмної інженерії»</u></b></p> <p>Галузь знань: <b><u>12 «Інформаційні технології»</u></b>          Спеціальності: <b><u>121 «Інженерія програмного забезпечення»</u></b></p> <p>Освітні програми: <b><u>«Розробка програмного забезпечення»</u></b></p>
<b>Статус дисципліни</b>	Навчальна дисципліна є <i>нормативною</i>
<b>Курс</b>	2-3
<b>Семестр</b>	4-5
<b>Обсяг дисципліни, кредити ЄКТС / загальна кількість годин</b>	6 кредитів /180 год.
<b>Мова викладання</b>	<b>українська</b>
<b>Що буде вивчатися (предмет навчання)</b>	Основними завданнями є глибоке та ґрунтовне засвоєння студентами основ програмної інженерії, ключових понять, методів і засобів програмної інженерії як діяльності, спрямованої на створення програмних систем, що відповідають потребам замовників, застосування інженерами не тільки теоретичних методів і засобів розробки ПЗ, але і стандартів, правил і методика управління розробкою ПЗ, а також питань проведення інженерних оцінок якості.
<b>Чому це цікаво / необхідно вивчити(мета)</b>	Метою викладання навчальної дисципліни “ Основи програмної інженерії” є формування у студентів розуміння необхідності застосування даних принципів програмної інженерії при розробці програмного забезпечення.
<b>Чому можна навчитись (компетентності)</b>	<p>Інтегральна компетентність:          Здатність вирішувати типові спеціалізовані задачі інженерії програмного забезпечення, що вимагає застосування положень і методів відповідних наук (математики, інформатики, інформаційних технологій, тощо) та може характеризуватися певною невизначеністю умов; нести відповідальність за результати своєї діяльності; здійснювати контроль інших осіб у визначених ситуаціях.</p> <p>Загальні компетентності:  <b>ЗК05.</b> Знання та розуміння предметної області та розуміння професійної діяльності.  <b>ЗК06.</b> Здатність до пошуку, оброблення та аналізу інформації з різних джерел.</p> <p>Спеціальні компетентності:  <b>СК02.</b> Здатність вдосконалювати знання і навички в галузі інформаційних технологій та усвідомлення важливості навчання протягом усього життя.</p> <p>Результати навчання:  <b>РН14.</b> Розуміти предметну область, застосовувати знання у професійній діяльності.  <b>РН15.</b> Аналізувати та узагальнювати необхідну інформацію з різних джерел та ресурсів для розв’язання професійних задач з урахуванням сучасних досягнень інформаційних технологій.</p>

<p><b>Як можна користуватись набутимизнаннями і вміннями (результати навчання)</b></p>	<p>Згідно з вимогами освітньо-професійної програми студенти повинні:</p> <p><i>знати:</i></p> <ul style="list-style-type: none"> <li>• принципи системного підходу до розробки ПЗ;</li> <li>• основи інженерії вимог до ПЗ;</li> <li>• основні концепції уніфікованої мови моделювання UML;</li> <li>• технології розробки ПЗ;</li> <li>• стратегії і методи проектування ПЗ;</li> <li>• нотації та засоби підтримки проектування;</li> <li>• процеси менеджменту проектів, життєвий цикл ПЗ;</li> <li>• поняття якості та культури програмного забезпечення.</li> </ul> <p><i>вміти:</i></p> <ul style="list-style-type: none"> <li>• здійснювати аналіз вимог, розробляти специфікацію програмних вимог, виконувати їхню верифікацію та атестацію;</li> <li>• моделювати різні аспекти системи, для якої створюється ПЗ;</li> <li>• домовлятися з партнерами про базові угоди;</li> <li>• володіти основами управління проектами;</li> <li>• визначати та вимірювати атрибути якості.</li> </ul>
<p><b>Пререквізити</b></p>	
<p><b>Постреквізити</b></p>	<p>«Організація комп'ютерних мереж», «Переддипломна практика», «Кваліфікаційна робота»</p>
<p><b>Навчальна логістика</b></p>	<p><i>Тема 1.</i> Програмна інженерія як дисципліна. SWEBOOK: Керівництво до зводу знань з програмної інженерії.</p> <p><i>Тема 2.</i> Процес створення програмного забезпечення. Моделі процесу створення. Еволюція програмних систем.</p> <p><i>Тема 3.</i> Робочий продукт. Дисципліна обов'язків. Проект.</p> <p><i>Тема 4.</i> Тривалість проекту та залучення персоналу. Методи оцінювання проекту. Алгоритмічне моделювання вартості проекту.</p> <p><i>Тема 5.</i> Керування якістю. Якість продукту та виробництва.</p> <p><i>Тема 6.</i> Архитектура ПЗ</p> <p><i>Тема 7.</i> Керування вимогами. Функціональні та не функціональні вимоги. Вимоги користувача. Системні вимоги. Документування вимог. Розробка вимог.</p> <p><i>Тема 8.</i> Прототипування в процесі розробки програмного забезпечення. Технології швидкого прототипування. Формальні специфікації в процесі розробки програмного забезпечення.</p> <p><i>Тема 9.</i> Конфігураційне керування.</p> <p><i>Тема 10.</i> Діаграмні техніки в роботі зі знаннями</p> <p><i>Тема 11.</i> Структурування системи. Модульна декомпозиція. Проблемно-залежні архітектури. Архитектура розподілених об'єктів. CORBA.</p> <p><i>Тема 12.</i> Об'єктно-орієнтоване проектування. Проектування систем реального часу. Системи нагляду і керування.</p> <p><i>Тема 13.</i> Покомпонентна розробка. Проектні патерни. Принципи проектування інтерфейсів користувачів. Оцінювання інтерфейсу.</p> <p><i>Тема 14.</i> Надійність систем. Працездатність і безвідмовність. Безпечність та захищеність. Специфікація критичних систем.</p> <p><i>Тема 15.</i> Тестування програмного забезпечення. Тестування дефектів. Тестування системи. Інструментальні засоби тестування.</p>

<b>Інформаційне забезпечення</b>	<b>3. Рекомендована література</b> <b>Базова</b> <ol style="list-style-type: none"><li>1. "Software Engineering: A Practitioner's Approach" by Roger S. Pressman Pressman, Roger S. Software Engineering: A Practitioner's Approach.</li><li>2. "Code Complete: A Practical Handbook of Software Construction" by Steve McConnell McConnell, Steve. Code Complete: A Practical Handbook of Software Construction.</li><li>3. "The Pragmatic Programmer: Your Journey to Mastery" by Dave Thomas and Andy Hunt Thomas, Dave, and Hunt, Andy. The Pragmatic Programmer: Your Journey to Mastery.</li><li>4. "Agile Software Development, Principles, Patterns, and Practices" by Robert C. Martin Martin, Robert C. Agile Software Development, Principles, Patterns, and Practices.</li><li>5. "Introduction to the Art of Programming Using Scala" by Mark C. Lewis Lewis, Mark C. Introduction to the Art of Programming Using Scala.</li><li>6. "The Mythical Man-Month: Essays on Software Engineering" by Frederick P. Brooks Jr. Brooks Jr., Frederick P. The Mythical Man-Month: Essays on Software Engineering.</li></ol>
--------------------------------------	---

<p><b>Політика навчальної дисципліни, оцінювання результатів навчання та академічна доброчесність</b></p>	<p><b>Політика щодо відвідування та проведення занять.</b> Під час лекцій, практичних та лабораторних занять використовуються різноманітні інтерактивні технології навчання, які допомагають не тільки засвоїти теми курсу, а й розвинути навички критичного мислення, вміння працювати з інформацією, презентувати результати власних досліджень.</p> <p>Передбачається обов'язкова присутність студента на кожному занятті, тому що для отримання ефекту занурення у проблематику дисципліни необхідне групове обговорення певних завдань та шляхи їх вирішення («мозковий штурм»).</p> <p>Слід відзначити, що через відсутність студента на занятті можна втратити логіку опанування теоретичного та практичного матеріалу, якою пов'язані всі теми курсу. Як правило, викладач попереджає це на вступній лекції, на якій відбувається знайомство зі структурно-логічною схемою курсу.</p> <p>У випадку, якщо була поважна причина відсутності студента на занятті, необхідно відвідати консультацію та з викладачем обговорити проблемні питання теми або низки тем через розбір «скрізних» питань, виконати практичні завдання.</p> <p>Під час вивчення курсу можна використовувати як рекомендовану літературу, так й різні інформаційні ресурси. Викладач контролює якість інформації, яку використовують здобувачі під час виконання завдань, вчить їх працювати з науковою інформацією, формує навички відрізняти якісну інформацію від неякісної. Мобільні пристрої під час проведення занять дозволяється використовувати лише для навчальних та наукових цілей.</p> <p><b>Політика щодо академічної доброчесності.</b> Політика щодо академічної доброчесності побудована на основі Положення про академічну доброчесність в ВСП «ФКЗІ ДУІТЗ». Усі види письмових робіт повинні бути написані здобувачами самостійно та мати високий рівень оригінальності.</p> <p>До заліку допускаються здобувачі, які виконали індивідуальне завдання. Здобувач, який не з'явився на залік або не був допущений на момент його проведення, має право повторно його пройти у визначені викладачем терміни під час консультацій/ відпрацювань.</p> <p><b>Загальна оцінка з дисципліни</b> – максимум 100 балів. У випадку отримання менше 60 балів, здобувач обов'язково здійснює перескладання для ліквідації академічної заборгованості.</p>
<p><b>Локація та</b></p>	<p><b>Навчальна аудиторія</b> (дошка, проектор, ноутбук, інше)</p>

<b>матеріально-технічне забезпечення</b>	обладнання). <i>Дистанційна</i> – сучасні платформи та онлайн-сервіси дистанційного навчання.	
<b>Семестровий контроль</b>	<i>Залік, іспит</i>	
<b>Циклова комісія</b>	Інформаційних технологій	
<b>Викладач</b>		<b>ПІБ</b> Петровський Руслан Ігорович
		<b>викладач</b>
		<b>E-mail:</b> legion2017wow@gmail.com