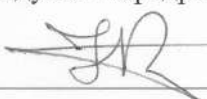


МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Львівський національний університет імені Івана Франка
Факультет електроніки та комп'ютерних технологій
Кафедра радіофізики та комп'ютерних технологій

Затверджено

На засіданні кафедри радіофізики та комп'ютерних технологій факультету електроніки та комп'ютерних технологій Львівського національного університету імені Івана Франка (протокол № 15/23 від 29.08 2023 р.)

Завідувач кафедри:


Іван КАРБОВНИК

Силабус з навчальної дисципліни
“Технології створення програмних продуктів”,
що викладається в межах ОПП “Комп'ютерні науки”
першого (бакалаврського) рівня вищої освіти для здобувачів з
спеціальності 122 – Комп'ютерні науки

Львів 2023 р.

Назва дисципліни	Технології створення програмних продуктів
Адреса викладання дисципліни	Корпус факультету електроніки та комп'ютерних технологій, Львівський національний університет імені Івана Франка, вул. Драгоманова 50, м. Львів, 79005, вул. Ген. Тарнавського 107, м. Львів, 79017
Факультет та кафедра, за якою закріплена дисципліна	Факультет електроніки та комп'ютерних технологій, кафедра радіофізики та комп'ютерних технологій
Галузь знань, шифр та назва спеціальності	12 – Інформаційні технології, 122 – Комп'ютерні науки
Викладачі дисципліни	Кушнір Олексій Олександрович, канд. фіз.-мат. наук, доцент
Контактна інформація викладачів	oleksiy.kushnir@lnu.edu.ua, https://electronics.lnu.edu.ua/employee/kushnir-o-o
Консультації з питань навчання по дисципліні відбуваються	Консультації в день проведення лекційних занять (за попередньою домовленістю). Також можливі онлайн консультації через MS Teams. Для погодження часу онлайн консультацій слід писати на електронну пошту викладача.
Сторінка дисципліни	https://e-learning.lnu.edu.ua/course/view.php?id=4501
Інформація про дисципліну	Дисципліна «Технології створення програмних продуктів» є нормативною дисципліною зі спеціальності 122 Комп'ютерні науки, яка викладається в 7 семестрі в обсязі 4,0 кредити (за Європейською Кредитно-Трансферною Системою ECTS).
Коротка анотація дисципліни	Навчальну дисципліну розроблено таким чином, щоб надати учасникам необхідні знання та навички для побудови комплексних програмних рішень у команді: проектування архітектури, розробки та контролю якості програмного продукту.
Мета та цілі дисципліни	Метою вивчення нормативної дисципліни «Технології створення програмних продуктів» є ознайомлення студентів із сукупністю виробничих процесів створення програмних засобів, які можуть ефективно використовуватися в процесах розробки програмного забезпечення, зокрема при аналізі вимог, моделюванні та проектуванні програмних систем.
Література для вивчення дисципліни	Основна література: 1. Wiegers, Karl; Beatty, Joy (2013). Software Requirements (3rd ed.). Microsoft Press. ISBN 978-0-7356-7966-5. 2. The history of coding and software engineering. Available at: https://www.galvanize.com/blog/the-history-of-coding-and-software-engineering/ 3. Software Engineering Software Design Process. Available at: https://www.geeksforgeeks.org/software-engineering-software-design-process/ 4. Software Engineering Body of Knowledge (SWEBOK). Available at: https://www.computer.org/education/bodies-of-knowledge/software-engineering 5. Будай А. Дизайн патерни – просто, як двері / А. Будай – 2012 – 90 с. 6. Jira tutorials: Learn agile with Jira Software. Available at: https://www.atlassian.com/agile/tutorials 7. Kan, Stephen H. (2003). Metrics and models in software quality engineering (2nd ed.). Boston: Addison-Wesley. ISBN 0-201-72915-6. Додаткова література: 8. Ambler, Scott William (2004). The Object Primer: Agile Model Driven

	<p>Development with UML 2</p> <p>9. Scrum Guides. Available at: https://scrumguides.org/</p> <p>10. Git. Available at: https://git-scm.com/</p>
Обсяг курсу	64 години аудиторних занять. З них 32 години лекцій, 32 години лабораторних робіт та 56 годин самостійної роботи
Очікувані результати навчання	<p>Після завершення цього курсу студент буде:</p> <ul style="list-style-type: none"> - Знати принципи складання вимог до програмних продуктів, моделі життєвого циклу розробки програмних продуктів, концепції проектування програмних продуктів, принципи конструювання, техніки досягнення гарантії якості та принципи керування різними етапами створення програмних продуктів - Вміти вибирати стратегії для планування життєвого циклу системи, визначати організаційну, економічну, технічну та операційну здійсненність проекту, проектувати компоненти програмного забезпечення, проектувати інтерфейс інформаційних систем, реалізовувати прототипи архітектури програмного забезпечення, реалізовувати та тестувати компоненти програмного забезпечення, інтегрувати компоненти в систему. <p>Після вивчення даного курсу здобувачі набудуть таких Загальних (ЗК), Спеціальних/Фахових (СК) компетентностей та Програмних результатів навчання (ПР):</p> <p>ЗК3. Знання та розуміння предметної області та розуміння професійної діяльності.</p> <p>ЗК9. Здатність працювати в команді.</p> <p>ЗК10. Здатність бути критичним і самокритичним.</p> <p>ЗК12. Здатність оцінювати та забезпечувати якість виконуваних робіт.</p> <p>СК 6. Здатність до системного мислення, застосування методології системного аналізу для дослідження складних проблем різної природи, методів формалізації та розв'язування системних задач, що мають суперечливі цілі, невизначеності та ризики.</p> <p>СК 8. Здатність проектувати та розробляти програмне забезпечення із застосуванням різних парадигм програмування: узагальненого, об'єктно-орієнтованого, функціонального, логічного, з відповідними моделями, методами й алгоритмами обчислень, структурами даних і механізмами управління.</p> <p>СК 9. Здатність реалізувати багаторівневу обчислювальну модель на основі архітектури клієнт-сервер, включаючи бази даних, знань і сховища даних, виконувати розподілену обробку великих наборів даних на кластерах стандартних серверів для забезпечення обчислювальних потреб користувачів, у тому числі на хмарних сервісах.</p> <p>СК 10. Здатність застосовувати методології, технології та інструментальні засоби для управління процесами життєвого циклу інформаційних і програмних систем, продуктів і сервісів інформаційних технологій відповідно до вимог замовника.</p> <p>СК 12. Здатність забезпечити організацію обчислювальних процесів в інформаційних системах різного призначення з урахуванням архітектури, конфігурування, показників результативності функціонування операційних систем і системного програмного забезпечення.</p> <p>ПР 11. Володіти навичками управління життєвим циклом програмного забезпечення, продуктів і сервісів інформаційних технологій відповідно до вимог і обмежень замовника, вміти розробляти проектну</p>

	<p>документацію (техніко-економічне обґрунтування, технічне завдання, бізнес-план, угоду, договір, контракт).</p> <p>ПР 15. Застосовувати знання методології та CASE-засобів проектування складних систем, методів структурного аналізу систем, об'єктно-орієнтованої методології проектування при розробці і дослідженні функціональних моделей організаційно-економічних і виробничо-технічних систем.</p>
Ключові слова	Програмна інженерія, архітектурні патерни, життєвий цикл програмного продукту, патерн програмування, тестування та контроль якості, командна робота
Формат курсу	Очний
	Проведення лекцій, лабораторних робіт та консультації для кращого розуміння тем
Теми	Див. СХЕМА КУРСУ
Підсумковий контроль, форма	Іспит в кінці семестру
Пререквізити	Для вивчення курсу студенти потребують базових знань з дисциплін «Алгоритмізація та проектування», «Об'єктно-орієнтовне програмування», «Кросплатформне проектування», «Теорія алгоритмів».
Навчальні методи та техніки, які будуть використовуватися під час викладання курсу	Презентація, лекції, лабораторні роботи, обговорення, дискусія.
Необхідне обладнання	<p>Мультимедіа, платформа Moodle, комп'ютерне програмне забезпечення</p> <p>Для проведення лекційних занять: комп'ютер (мінімальні характеристики: процесор Intel Core i3(4 ядра/8 потоків), або краще, 8ГБ, або більше оперативної пам'яті, 50ГБ вільного місця на диску, вбудована або дискретна відеокарта), доступ до мережі Internet, засоби мультимедіа (в т.ч. проектор).</p> <p>Для проведення лабораторних занять: Комп'ютер (мінімальні характеристики: процесор з апаратною підтримкою віртуалізації Intel Core i3(4 ядра/8 потоків), або краще, 8ГБ, або більше оперативної пам'яті, 50ГБ вільного місця на диску, вбудована або дискретна відеокарта) доступ до мережі інтернет. Необхідне програмне забезпечення включає в себе ОС Windows/Linux/Mac, середовище розробки MS Visual Studio/PyCharm/Vim/VS Code/інше, компілятор мови програмування C++/Python, система контролю версій git, доступ до системи управління проектами Atlassian JIRA.</p>
Критерії оцінювання (окремо для кожного виду навчальної діяльності)	<p>Оцінювання проводиться за 100-бальною шкалою. Бали нараховуються за наступним співвідношенням:</p> <ul style="list-style-type: none"> • Змістовий модуль 1: 25% семестрової оцінки за виконані лабораторні роботи та змістовий модуль 1. • Змістовий модуль 2: 25% семестрової оцінки за виконані лабораторні роботи та змістовий модуль 2. • екзамен: 50% семестрової оцінки, максимальна кількість балів 50. <p>Підсумкова максимальна кількість балів 100.</p> <p>Контрольні заміри проводяться у формі тестових завдань.</p> <p>Модульний контроль проводиться у формі презентації: модуль 1 – ідея та архітектура проекту, модуль 2 – реалізація проекту.</p> <p>Екзамен проводиться у формі тесту із 50 випадкових питань рівної</p>

складності.

Академічна доброчесність: Очікується, що лабораторні та контрольні роботи студентів будуть їх оригінальними дослідженнями чи міркуваннями. Відсутність посилань на використані джерела, фабрикування джерел, списування, втручання в роботу інших студентів становлять, але не обмежують, приклади можливої академічної недоброчесності. Виявлення ознак академічної недоброчесності в роботі студента є підставою для її незарахування викладачем, незалежно від масштабів плагіату чи обману.

Відвідання занять є важливою складовою навчання. Очікується, що всі студенти відвідають усі лекції і лабораторні заняття курсу. Студенти мають інформувати викладача про неможливість відвідати заняття. Студенти зобов'язані дотримуватися усіх термінів визначених для виконання усіх видів робіт, передбачених курсом.

Література. Уся література, яку студенти не зможуть знайти самостійно, буде надана викладачем виключно в освітніх цілях без права її передачі третім особам. Студенти заохочуються до використання також й іншої літератури та джерел, яких немає серед рекомендованих.

Політика виставлення балів. Враховуються бали набрані на поточному тестуванні, самостійній роботі та бали підсумкового тестування. При цьому обов'язково враховуються присутність на заняттях та активність студента під час лабораторного заняття; недопустимість пропусків та запізнь на заняття; користування мобільним телефоном, планшетом чи іншими мобільними пристроями під час заняття в цілях не пов'язаних з навчанням; списування та плагіат; несвоєчасне виконання поставленого завдання і т. ін.

Жодні форми порушення академічної доброчесності не толеруються.

Оцінювання лабораторних робіт (16 лабораторних робіт, максимальна кількість балів: 32) відбувається шляхом оцінки роботи студента під час проведення лабораторної роботи в аудиторії (0-5 балів за одну роботу) та захисту звіту по виконаній лабораторній роботі (0-5 балів за одну роботу). У підсумку, всі набрані бали множаться на коефіцієнт (0.2) для переведення у 32-а бальну шкалу.

Бали оцінювання лабораторних робіт нараховуються за наступним співвідношенням:

5 – студент в повному обсязі володіє навчальним матеріалом, має повне розуміння розглянутої теми, надає правильні відповіді на запитання по темі, код програми функціонує відповідно до завдання;

4 – студент достатньо розуміє розглянутий матеріал та принципи написаного ним коду програми, присутні неточності та незначні помилки у відповідях на запитання по темі, код програми функціонує відповідно до завдання (або з несуттєвими недоліками);

3 – студент не досить добре розуміє розглянутий матеріал та написаний ним код програми, вагається та надає неточні/не конкретні відповіді на запитання по темі, код програми функціонує неточно, або з помірними недоліками;

2 – студент погано розуміє розглянутий матеріал та написаний ним код програми, студент в більшості надає помилкові відповіді на питання по темі, код програми функціонує з суттєвими недоліками;

1 - студент погано розуміє розглянутий матеріал та написаний ним код програми, код програми не функціонує належним чином;

	<p>0 - студент зовсім не засвоїв розглянутий матеріал, написаний ним код програми не відповідає темі/не функціонує взагалі.</p> <p>Оцінювання змістових модулів (2 змістових модулів, 9 балів за кожний) — за результатами написаних студентом есе, тестів, програм, тощо.</p> <p>Бали оцінювання змістових модулів нараховуються за представлення презентацій: якість ілюстративного матеріалу 0-3 бали, об'єм виконаної роботи 0-4 бали, обґрунтованість висновків 0-2 бали. Бали виставляють викладачі та/або студенти та/або запрошені експерти методом закритого голосування. Виставляється середня оцінка із заокругленням у більшу сторону.</p> <p>Оцінювання екзамену проводиться у формі тесту із 50 випадкових питань рівної складності. Оцінка за екзамен виставляється пропорційно до кількості правильних відповідей, які студент надав під час проходження тестування (1 правильна відповідь – 1 бал). Перелік тем див. у розділі Питання до заліку чи екзамену.</p> <p>Критерії оцінювання результатів неформальної освіти: Нарахування балів відбувається за написання студентом тез доповідей на конференціях, наукових статей, участь у діяльності наукових гуртків, участь у наукових семінарах та круглих столах, конкурсах, участь у заходах неформальної освіти за отримання сертифікатів про проходження навчання на різних освітніх платформах (Coursera, Prometheus тощо), курсах на провідних ІТ компаніях за тематикою навчальної дисципліни. Кількість балів визначається відсотком покриття результатів відповідної активності до вимог результатів навчання з навчальної дисципліни.</p>
<p>Питання до контрольних робіт</p>	<p>Орієнтовний перелік питань та завдань для проведення підсумкової оцінки знань:</p> <ol style="list-style-type: none"> 1. Етапи створення програмного продукту. 2. Вимоги, класифікація вимог. 3. Інженерія, специфікація та трасування вимог. 4. Вимоги до інтерфейсу користувача. 5. Засоби контролю версій. 6. Життєвий цикл програмного продукту. Процеси життєвого циклу. 7. Моделі життєвого циклу програмних продуктів. 8. Стандарти розробки програмних продуктів. 9. Архітектура програмного продукту. 10. Підходи до проектування архітектури програмних продуктів. 11. Види архітектур програмних продуктів, основні архітектурні шаблони. 12. CASE-методи та їх вплив на розробку. 13. Патерни проектування та їх класифікація. 14. Структурні патерни. 15. Патерни поведінки класів/об'єктів. 16. Твірні патерни. 17. Поняття якості програмного продукту. Причини виникнення помилок. 18. Шаблон баг-репорту, його необхідні складові. 19. Класифікація видів тестування.

	<p>20. Рівні тестування.</p> <p>21. Автоматизоване тестування.</p> <p>22. Валідація та верифікація програмних продуктів.</p> <p>23. Методи доведення правильності програм.</p> <p>24. Підтримка програмних продуктів.</p> <p>25. Підходи до оновлення програмних продуктів.</p>
Опитування	Анкету-оцінку з метою оцінювання якості курсу буде надано по завершенню курсу.

СХЕМА КУРСУ

Тиж.	Тема, план, короткі тези	Форма діяльності (заняття)	Література. Ресурси в Інтернеті	Завдання (лабораторна робота), год	Термін виконання
1	Поняття програмного продукту, основна термінологія. Вступ і загальний огляд дисципліни. Основна термінологія курсу. Етапи створення програмного продукту.	Лекція	1, 2, 3, Сайт курсу https://e-learning.lnu.edu.ua/course/view.php?id=4501	2	1 тиж. семестру
1	Вступне заняття. Інструкція з техніки безпеки.	Лабораторна робота	1, 2, Сайт курсу https://e-learning.lnu.edu.ua/course/view.php?id=4501	2	1 тиж. семестру
2	Вимоги до програмних продуктів. Поняття вимог. Напрямки розробки вимог. Класифікація вимог. Інженерія, специфікація та трасування вимог. Вимоги до інтерфейсу користувача.	Лекція	1, 2, 3, Сайт курсу https://e-learning.lnu.edu.ua/course/view.php?id=4501	2	2 тиж. семестру
2	Витягнення та опрацювання вимог клієнта.	Лабораторна робота	2, Сайт курсу https://e-learning.lnu.edu.ua/course/view.php?id=4501	2	2 тиж. семестру
3	Засоби контролю версій. Історія виникнення засобів контролю версій. Принципи роботи та сучасні варіації. Використання Git та його модифікацій.	Лекція	1, 4, 10, Сайт курсу https://e-learning.lnu.edu.ua/course/view.php?id=4501	2	3 тиж. семестру
3	Написання технічного завдання.	Лабораторна робота	1, 2, 3 Сайт курсу https://e-learning.lnu.edu.ua/course/view.php?id=4501	2	3 тиж. семестру
4	Управління проектами та командна робота. Підходи до управління проектами, гнучка розробка, Scrum та Kanban підходи, схожість та принципова різниця.	Лекція	1, 2, 3, 9, Сайт курсу https://e-learning.lnu.edu.ua/course/view.php?id=4501	2	4 тиж. семестру
4	Налаштування системи контролю версій у проекті, синхронізація, спільна розробка.	Лабораторна робота	1, 2, 10, Сайт курсу https://e-learning.lnu.edu.ua/course/view.php?id=4501	2	4 тиж. семестру
5	Життєві цикли програмних продуктів. Поняття життєвого циклу програмного продукту. Процеси життєвого циклу. Моделі життєвого циклу програмних продуктів: каскадна модель, інкрементна модель, ітераційна	Лекція	2, 3, 8, Сайт курсу https://e-learning.lnu.edu.ua/course/view.php?id=4501	2	5 тиж. семестру

	модель, еволюційна модель, V-подібна модель.				
5	Вибір моделі життєвого циклу програмного продукту.	Лабораторна робота	2, 8, Сайт курсу https://e-learning.lnu.edu.ua/course/view.php?id=4501	2	5 тиж. семестру
6	Стандарти розробки програмних продуктів. Історія виникнення стандартів до розробки програмних продуктів. ДСТУ для розробки програмних продуктів. Стандарт ISO: процеси життєвих циклів за ISO, зміст основних процесів, стадії створення, оцінка якості за стандартом ISO. Галузевий стандарт IEEE – SWEBOOK: стандарти процесів підприємництва за SWEBOOK, модель зрілості компаній розробників, визначення рівнів зрілості, області процесів розробки для різних рівнів зрілості. Переваги та недоліки різних стандартів розробки програмних продуктів.	Лекція	3, 4, 7, Сайт курсу https://e-learning.lnu.edu.ua/course/view.php?id=4501	2	6 тиж. семестру
6	Налаштування та використання для управління командною роботою Atlassian JIRA	Лабораторна робота	6, Сайт курсу https://e-learning.lnu.edu.ua/course/view.php?id=4501	2	6 тиж. семестру
7	Проектування програмних продуктів. Поняття проектування та архітектури програмного продукту. Основні поняття універсальної мови моделювання UML. Рекомендації щодо побудови архітектури.	Лекція	2, 3, 4, Сайт курсу https://e-learning.lnu.edu.ua/course/view.php?id=4501	2	7 тиж. семестру
7	Розподіл обов'язків у команді, трекінг завдань	Лабораторна робота	6, Сайт курсу https://e-learning.lnu.edu.ua/course/view.php?id=4501	2	7 тиж. семестру
8	Архітектура програмних засобів. Стандартизований та загальносистемний підходи до проектування архітектури програмних продуктів. Види архітектур, основні архітектурні шаблони.	Лекція	4, 5, Сайт курсу https://e-learning.lnu.edu.ua/course/view.php?id=4501	2	8 тиж. семестру
8	Підсумкове заняття ЗМ 1	Лабораторна робота	1, 2, 3, 4, 6, Сайт курсу https://e-learning.lnu.edu.ua/course/view.php?id=4501	2	8 тиж. семестру
9	Автоматизація розробки програмних продуктів. Класифікація інструментальних середовищ розробки програмних продуктів. Комп'ютерні технології розробки програмних продуктів – CASE: класифікація CASE-технологій, їхні переваги, основні зміни життєвого циклу при використанні CASE, зміна	Лекція	1, 3, 8, Сайт курсу https://e-learning.lnu.edu.ua/course/view.php?id=4501	2	11 тиж. семестру

	трудовитрат по фазах життєвого циклу програмних продуктів. Структурний аналіз: засоби функціонального, інформаційного та подійного моделювання.				
9	Побудова загальної архітектури програмного засобу.	Лабораторна робота	3, 8, Сайт курсу https://e-learning.lnu.edu.ua/course/view.php?id=4501	2	9 тиж. семестру
10-11	Патерни проектування. Історія виникнення шаблонів проектування – патернів. Класифікація патернів. Структурні патерни. Патерни поведінки класів/об'єктів. Твірні патерни.	Лекція	5, 8, Сайт курсу https://e-learning.lnu.edu.ua/course/view.php?id=4501	2	11 тиж. семестру
10	Проектування модулів програмного продукту.	Лабораторна робота	5, 8, Сайт курсу https://e-learning.lnu.edu.ua/course/view.php?id=4501	2	10 тиж. семестру
11	Використання патернів для рішення стандартних завдань.	Лабораторна робота	5, Сайт курсу https://e-learning.lnu.edu.ua/course/view.php?id=4501	2	11 тиж. семестру
12	Контроль якості програмних продуктів. Поняття якості. Шаблону рапорту про помилку (баг-репорту). Тестування програмних систем: класифікація видів тестування, статичні методи тестування, динамічні методи тестування, функціональне тестування.	Лекція	3, Сайт курсу https://e-learning.lnu.edu.ua/course/view.php?id=4501	2	12 тиж. семестру
12	Ознайомлення із засобами автоматизованого високорівневого тестування.	Лабораторна робота	Сайт курсу https://e-learning.lnu.edu.ua/course/view.php?id=4501	2	12 тиж. семестру
13	Тестування програмних продуктів. Рівні тестування: компонентне або модульне тестування, інтеграційне тестування, системне тестування, приймальне тестування. Автоматизоване тестування. Засоби автоматизації тестування Навантажувальне тестування.	Лекція	7, Сайт курсу https://e-learning.lnu.edu.ua/course/view.php?id=4501	2	13 тиж. семестру
13	Ознайомлення із засобами автоматизованого високорівневого тестування.	Лабораторна робота	7, Сайт курсу https://e-learning.lnu.edu.ua/course/view.php?id=4501	2	13 тиж. семестру
14	Верифікація та валідація інформаційних систем. Валідація програмних продуктів. Верифікація програмних продуктів: верифікація об'єктних моделей, базові методи доведення правильності програм, модель доведення програми за твердженнями, верифікація композиції компонентів, майбутнє верифікації програм.	Лекція	7, Сайт курсу https://e-learning.lnu.edu.ua/course/view.php?id=4501	2	14 тиж. семестру
14	Підготовка модульних тестів.	Лабораторна робота	7, Сайт курсу https://e-learning.lnu.edu.ua/course/view.php?id=4501	2	14 тиж. семестру

15	Засоби (підтримки) построзробки програмних систем Підтримка програмних продуктів після розробки та її складові. Сучасні засоби автоматизації збору та опрацювання даних від користувачів. Оновлення програмних продуктів.	Лекція	4, 8, Сайт курсу https://e-learning.lnu.edu.ua/course/view.php?id=4501	2	15 тиж. семестру
15	Робота із засобами підтримки програмних продуктів.	Лабораторна робота	4, 8, Сайт курсу https://e-learning.lnu.edu.ua/course/view.php?id=4501	2	15 тиж. семестру
16	Представлення результатів Як представити результати роботи замовнику? Загальні підходи та приклади. Підготовка презентації продукту.	Лекція	Сайт курсу https://e-learning.lnu.edu.ua/course/view.php?id=4501	2	16 тиж. семестру
16	Підсумкове заняття ЗМ 2	Лабораторна робота	Сайт курсу https://e-learning.lnu.edu.ua/course/view.php?id=4501	2	16 тиж. семестру