

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

Затверджено

На засіданні
кафедри системного проектування
факультету електроніки та комп'ютерних
технологій
Львівського національного університету
імені Івана Франка
(протокол № 1 від 28.8 2023 р.)

Завідувач кафедри:



Роман ШУВАР

Силабус з навчальної дисципліни
“Вступ в інженерію програмного забезпечення”,
що викладається в межах ОПП
«Інженерія програмного забезпечення»
першого (бакалаврського) рівня вищої освіти для здобувачів з
спеціальності 121 – Інженерія програмного забезпечення

Львів 2023 р.

Назва дисципліни	Вступ в інженерію програмного забезпечення
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, 79011
Факультет та кафедра, за якою закріплена дисципліна	Факультет електроніки та комп'ютерних технологій Кафедра системного проектування
Галузь знань, шифр та назва спеціальності	12 – інформаційні технології 121 – Інженерія програмного забезпечення
Викладачі дисципліни	Ткаченко Олександр Миколайович Місюк Роман Володимирович
Контактна інформація викладачів	oleksandr.trachenko@lnu.edu.ua, https://electronics.lnu.edu.ua/employee/tkachenko-oleksandr , roman.mysiuk@lnu.edu.ua, https://electronics.lnu.edu.ua/employee/mysiuk-r-roman
Консультації з питань навчання по дисципліні відбуваються	Консультації в день проведення лекцій/лабораторних занять (за попередньою домовленістю): ауд. 205, корпус факультету електроніки та комп'ютерних технологій, вул. Драгоманова 50, м. Львів
Сторінка курсу	https://electronics.lnu.edu.ua/course/web-tekhnolohii-ipz
Коротка анотація дисципліни	Дисципліна “Вступ в інженерію програмного забезпечення” є нормативною дисципліною з спеціальності 121 – Інженерія програмного забезпечення для освітньої програми “Інженерія програмного забезпечення”, яка викладається в 1-му семестрі в обсязі 4-ох кредитів (за Європейською Кредитно-Трансферною Системою ECTS).
Інформація про дисципліну	Навчальну дисципліну розроблено таким чином, щоб надати студентам необхідні знання, обов'язкові для того, щоб оволодіти базовими поняттями, пов'язаними з організацією та використанням сучасних інформаційних технологій, оглядово зрозуміти технології та інструменти для роботи протягом навчання. Тому у дисципліні представлено як огляд базових понять та інструментів інформаційних технологій, так і засобів, які потрібні для вирішення типових завдань при використанні, налаштуванні та управлінні проектами, розробки перших програм та програмних інтерфейсів.
Мета та цілі дисципліни	<i>Мета:</i> оволодіння базовими поняттями, теоретичними знаннями та практичними навичками використання інформаційних технологій в різних галузях людської діяльності, а також ознайомлення з роботою в команді, проектуванні програмного забезпечення та розробки простих програм. <i>Цілі:</i> забезпечити знайомство студентів з основними процесами та інструментами при розробці програмного забезпечення.
Література для вивчення дисципліни	1) Software engineering / Ian Sommerville. - 9th ed. - 790 p. - ISBN-13: 978-0-13-703515-1. 2) Titus Winters, Tom Manshreck, Hyrum Wright. Software Engineering at Google. – O'Reilly, 2020 (https://abseil.io/resources/swe-book) 3) І. Л. Бородкіна, Г. О. Бородкін. Інженерія програмного забезпечення: Посібник для студентів вищих навчальних закладів / М-во освіти і науки

	<p>України, Національний університет біоресурсів та природокористування України. – Київ: 2018 / 251 с.</p> <p>4) Rajib Mall. Fundamentals of software engineering. - [Режим доступу]: https://davcollegetitilagarh.org/wp-content/uploads/2020/09/fundamentals-of-software-engineering-fourth-edition-rajib-mall.pdf</p> <p>5) W3School Online Web Tutorials [Електронний ресурс] : [Веб-сайт]. – Електронні дані. – Режим доступу: https://www.w3schools.com/</p> <p>6) Robson Elisabeth. Head First HTML and CSS: A Learner’s Guide to Creating Standards-Based Web Pages / Elisabeth Robson, Eric Freeman. – Canada : O’Reilly Media, Inc., 2017. – 764 p. https://www.pdfdrive.com/head-first-html-and-css-a-learners-guide-to-creating-standards-based-web-pages-e158237724.html</p> <p>7) Git Book [Електронний ресурс] : [Веб-сайт]. – Електронні дані. – Режим доступу: https://git-scm.com/book/uk/v2</p> <p><i>Допоміжна:</i></p> <p>8) Бурячок, В. Л. Інформаційний та кіберпростори: проблеми безпеки, методи та засоби боротьби: підручник: затверджено МОН України / В. Л. Бурячок, Г. М. Гулак, В. Б. Толубко. – Львів : Магнолія 2006, 2018. – 448 с. : іл. – (Cybersecurity)</p> <p>9) Грицюк, Ю. І. Аналіз вимог до програмного забезпечення: навч. посіб. / Ю. І. Грицюк. – Львів : Вид-во Львівської політехніки, 2018. – 453 с. : іл.</p> <p>10) Інформаційна безпека [Електронний ресурс]: рекомендаційний покажчик / С. Л. Бондар, Т. А. Сіденко, Т. А. – Чернігів: Наукова бібліотека НУ "Чернігівська політехніка", 2020. – 44 с. – Електронна копія існує. Режим доступу: http://ir.stu.cn.ua/handle/123456789/20038.</p> <p>11) Вступ до інженерії програмного забезпечення: навч. посібник / Є.В. Левус, Н.Б. Мельник – Львів: Видавництво Львівської політехніки, 2017. - 280 с.</p> <p>12) Мельник, Р. А. Програмування веб-застосувань (фронт-енд та бек-енд) : навч. посіб. / Р. А. Мельник. – Львів : Вид-во Львівської політехніки, 2018. – 247 с. : іл.</p>
Обсяг курсу	Загальний обсяг: 120 годин. Аудиторних занять: 64 год., з них 32 год. лекційних та 32 годин лабораторних робіт. Самостійної роботи: 56 год.
Очікувані результати навчання	<p>У результаті вивчення даного курсу студент буде:</p> <p>знати:</p> <p>основні поняття, визначення і проблеми курсу; вимоги до постановки основних задач програмування веб сторінок; призначення й особливості застосування основних складових розробки веб сторінки в мережі Інтернет;</p> <p>вміти:</p> <p>володіти базовими знаннями розмітки, стилів та скриптування веб сторінок; застосовувати технології розробки веб сторінок для прикладних задач; реалізувати прикладні задачі в розробці веб сторінок на персональних комп’ютерах.</p> <p>Після вивчення даного курсу здобувачі набудуть таких Загальних та Фахових компетентностей та Програмних результатів навчання:</p> <p>ЗК01. Здатність до абстрактного мислення, аналізу та синтезу.</p> <p>ЗК02. Здатність застосовувати знання у практичних ситуаціях.</p>

	<p>ЗК04. Здатність спілкуватися іноземною мовою як усно, так і письмово.</p> <p>ЗК07. Здатність працювати в команді.</p> <p>ФК13. Здатність ідентифікувати, класифікувати та формулювати вимоги до програмного забезпечення.</p> <p>ФК20. Здатність застосовувати фундаментальні і міждисциплінарні знання для успішного розв'язання завдань інженерії програмного забезпечення.</p> <p>ФК25. Здатність обґрунтовано обирати та освоювати інструментарій з розробки та супроводження програмного забезпечення.</p> <p>ФК29. Здатність здійснювати розробку програмного забезпечення використовуючи різні методології програмування (в тому числі паралельне, об'єктно-орієнтоване, функціональне програмування та інші)</p> <p>ПРН01. Аналізувати, цілеспрямовано шукати і вибирати необхідні для вирішення професійних завдань інформаційно-довідникові ресурси і знання з урахуванням сучасних досягнень науки і техніки.</p> <p>ПРН02. Знати кодекс професійної етики, розуміти соціальну значимість та культурні аспекти інженерії програмного забезпечення і дотримуватись їх в професійній діяльності.</p> <p>ПРН04. Знати і застосовувати професійні стандарти і інші нормативно-правові документи в галузі інженерії програмного забезпечення.</p> <p>ПРН06. Вміння вибирати та використовувати відповідну задачі методологію створення програмного забезпечення.</p> <p>ПРН09. Знати та вміти використовувати методи та засоби збору, формулювання та аналізу вимог до програмного забезпечення.</p> <p>ПРН20. Знати підходи щодо оцінки та забезпечення якості програмного забезпечення.</p>
Ключові слова	ІТ індустрія, ІТ позиції, Git, веб сторінка, мова програмування Python, управління проектами
Формат курсу	Очний
Теми	<p>Тема 1. <i>Вступ. Огляд індустрії та дослідження ринку послуг комп'ютерного програмування. Типи компаній.</i></p> <p>Тема 2. <i>Створення резюме. Представлення себе як спеціаліста. Помилки та труднощі</i></p> <p>Тема 3. <i>Soft skills – важлива складова майбутнього. Підготовка до публічного виступу</i></p> <p>Тема 4. <i>ІТ – не лише розробники. Огляд існуючих ролей, напрямків та професій в індустрії</i></p> <p>Тема 5. <i>Керування та комунікація на проектах. Project Manager, Business Analyst, Scrum master, Product owner & etc.</i></p> <p>Тема 6. <i>Розробки. Frontend vs Backend. Обов'язки та основні інструменти та мови програмування, які використовують розробники.</i></p> <p>Тема 7. <i>Quality Assurance engineers. Методи тестування, рівні тестування, життєвий цикл, організація процесу. Автоматизоване тестування</i></p> <p>Тема 8. <i>Методології (Agile, Waterfall, Lean & etc).. Порівняння, застосування та приклади.</i></p> <p>Тема 9. <i>Системи контролю версій.</i></p> <p>Тема 10. <i>Remote work is a “new black”. How to work remotely? Взаємодія у команді.</i></p> <p>Тема 11. <i>Tools: Jira, Git, Trello & other. Робота з інструментами.</i></p>

	<p><i>Планування дня. Приклад планування захисту курсової чи іншого виду навчальної роботи у цих інструментах.</i></p> <p>Тема 12. <i>Всесвітня павутина. Хмарні технологія – нова ера роботи з даними. Огляд існуючих хмарних середовищ</i></p> <p>Тема 13. <i>Захист даних. Алгоритми шифрування перестановкою, Відженера та Цезаря.</i></p> <p>Тема 14. <i>Популярні мови програмування. Рейтинг найзатребуваніших спеціалістів</i></p> <p>Тема 15. <i>Системи контролю версій. GIT. Взаємодія у написанні коду на великих проектах</i></p> <p>Тема 16. <i>Управління часом. Вчимося вчитись та чому час грає проти нас? Планування часу. Чому це важливо? Як знайти час та чому його не вистачає?</i></p>
Підсумковий контроль, форма	Екзамен
Пререквізити	<p>Для вивчення даного курсу студентам потрібні базові знання з курсів:</p> <ul style="list-style-type: none"> - комп'ютерні інформаційні мережі і системи; - основи програмування;
Навчальні методи та техніки, які будуть використовуватися під час викладання курсу	<p>Інформаційні методи (лекція, бесіда, ілюстрація, демонстрація); дедуктивні методи на основі узагальнень; евристичні методи (проблемна лекція); інтерактивні методи (дискусія).</p>
Необхідне обладнання	<p>Для проведення лекційних занять:</p> <p>Комп'ютер (мінімальні характеристики): процесор Intel Core i3, 8ГБ оперативної пам'яті, 50ГБ вільного місця на диску, відеокарта Nvidia GEFORCE GT1030 2048MB), доступ до мережі Internet, засоби мультимедіа (в т.ч. проектор).</p> <p>Для проведення лабораторних занять:</p> <p>Комп'ютер (мінімальні характеристики: процесор Intel Core i3, 8ГБ оперативної пам'яті, 50ГБ вільного місця на диску), необхідне програмне забезпечення включає в себе ОС Ubuntu 20.04 LTS, програмне забезпечення для побудови діаграм draw.io, Git Bash.</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. • Змістовий модуль 2: 25% семестрової оцінки за виконані лабораторні роботи. • екзамен: 50% семестрової оцінки, максимальна кількість балів 50. <p>Підсумкова максимальна кількість балів 100.</p> <p>Оцінки за лабораторні заняття розподіляються наступним чином: виконання лабораторних завдань – 60 %, відповіді на запитання викладача</p>

	<p>по темі заняття – 40 %.</p> <p>Академічна доброчесність: Очікується, що роботи студентів будуть оригінальними дослідженнями чи міркуваннями. Списування та втручання в роботу інших студентів становлять, але не обмежують, приклади можливої академічної недоброчесності. Виявлення ознак академічної недоброчесності в написанні завдань є підставою для її незарахування викладачем, незалежно від масштабів плагіату чи обману.</p> <p>Відвідання занять є важливою складовою навчання. Очікується, що всі студенти відвідають усі лекції та лабораторні заняття курсу. Студенти повинні інформувати викладача про неможливість відвідати заняття. У будь-якому випадку студенти зобов'язані дотримуватися термінів визначених для виконання всіх видів робіт, передбачених курсом.</p> <p>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.</p> <p>Політика виставлення балів. Враховуються бали, набрані при поточному контролі та бали підсумкового тестування. При цьому обов'язково враховуються присутність на заняттях та активність студента під час практичного 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.</p> <p>Жодні форми порушення академічної доброчесності не толеруються.</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 (зокрема, https://training.epam.ua/ua) за тематикою навчальної дисципліни. Кількість балів визначається відсотком покриття результатів відповідної активності до вимог результатів навчання з навчальної дисципліни.</p>
<p>Питання до заліку чи екзамену.</p>	<p>Псевдокод. Задача на псевдокод та (або) опис процесу Основи Python. Ключові слова, роботу із списками в Python. Побудова команди до того чи іншого проекту та опис портрету користувача даного проекту. Порівняння Soft та Hard навичок. Типи компаній Технічна революція Методології розробки програмного забезпечення Рекрутингова сфера. Складові резюме Scrum master та проектний менеджер. Відмінності. Front-end developer та пов'язані мови програмування Back-end developer та пов'язані мови програмування Тестування як процес розробки програмного забезпечення Мережеві технології Проекти та управління ними. Інструменти</p>

Опитування	Анкету-оцінку з метою оцінювання якості курсу буде надано по завершенню курсу.
-------------------	--

**Схема курсу “Вступ в інженерію програмного забезпечення”
для студентів спеціальності 121 – Інженерія програмного забезпечення**

Тиж .	Тема, план, короткі тези	Форма діяльності (заняття)* *лекція, самостійна, дискусія, групова робота)	Література.** * Ресурси в інтернеті	Завдання, год	Термін виконання
1	Вступ. Огляд індустрії та дослідження ринку послуг комп’ютерного програмування. Типи компаній.	Лекція	[1], [2], Сайт курсу	2	кінець поточного тижня
1	Лаб. 1. Типи ІТ компаній	Лабораторн а робота	Сайт курсу	2	кінець поточного тижня
2	Створення резюме. Представлення себе як спеціаліста. Помилки та труднощі	Лекція	Сайт курсу	2	кінець поточного тижня
2	Лаб. 2. Створення власного резюме	Лабораторн а робота	Сайт курсу	2	кінець поточного тижня
3	Soft skills – важлива складова майбутнього. Підготовка до публічного виступу	Лекція	Сайт курсу	2	кінець поточного тижня
3	Лаб. 3. Підготовка презентацію про власні захоплення, відомих особистостей в ІТ чи інше.	Лабораторн а робота	Сайт курсу	2	кінець поточного тижня
4	ІТ – не лише розробники. Огляд існуючих ролей, напрямків та професій в індустрії	Лекція	[1], [2], Сайт курсу	2	кінець поточного тижня
4	Лаб. 4. Робота над моделюванням поділу відповідальності в команді	Лабораторн а робота	Сайт курсу	2	кінець поточного тижня
5	Керування та комунікація на проєктах. Project Manager, Business Analyst, Scrum master, Product owner & etc.	Лекція	[1], [2],[3] Сайт курсу	2	кінець поточного тижня
5	Лаб. 5. Створення моделей ПЗ в draw.io.	Лабораторн а робота	Сайт курсу	2	кінець поточного тижня
6	Розробки. Frontend vs Backend. Обов’язки та основні інструменти та мови програмування, які використовують розробники.	Лекція	[4], [5], [6], Сайт курсу	2	кінець поточного тижня
6	Лаб. 6. Написання програми на Python.	Лабораторн а робота	Сайт курсу	2	кінець

		а робота			поточного тижня
7	Remote work is a “new black”. How to work remotely? Взаємодія у команді.	Лекція	Сайт курсу	2	кінець поточного тижня
7	Лаб. 7. Робота з команді над написанням коду мовою програмування Python	Лабораторн а робота	Сайт курсу	2	кінець поточного тижня
8	Методології (Agile, Waterfall, Lean & etc. Порівняння, застосування та приклади.	Лекція	[4], [5], Сайт курсу	2	кінець поточного тижня
8	Підсумкове заняття ЗМ 1	Лабораторн а робота	Сайт курсу	2	кінець поточного тижня
9	Системи контролю версій.	Лекція	[8] Сайт курсу	2	Кінець поточного тижня
9	Лаб. 8. Робота з текстовими файлами в Python	Лабораторн а робота	Сайт курсу	2	кінець поточного тижня
10	Quality Assurance engineers. Методи тестування, рівні тестування, життєвий цикл, організація процесу. Автоматизоване тестування	Лекція	[4], [5] Сайт курсу	2	кінець поточного тижня
10	Лаб. 9. Тестування написаної програми, та використання умовних виразів для введеного тексту	Лабораторн а робота	Сайт курсу	2	кінець поточного тижня
11	Tools: Jira, Git, Trello & other. Робота з інструментами. Планування дня. Приклад планування захисту курсової чи іншого виду навчальної роботи у цих інструментах.	Лекція	Сайт курсу	2	кінець поточного тижня
11	Лаб.10. Оцінювання складовості та часу роботи використовуючи Poker Planning	Лабораторн а робота	Сайт курсу	2	кінець поточного тижня
12	Всесвітня павутина. Хмарні технологія – нова ера роботи з даними. Огляд існуючих хмарних середовищ	Лекція	[9], Сайт курсу	2	кінець поточного тижня
12	Лаб. 11. Хостинг веб сторінки в GitHub Pages	Лабораторн а робота	Сайт курсу	2	кінець поточного тижня
13	Захист даних. Алгоритми шифрування перестановкою, Відженера та Цезаря.	Лекція	[3], [7], [10] Сайт курсу	2	кінець поточного тижня
13	Лаб. 12. Шифрування та дешифрування слів за допомогою алгоритмів перестановки, Відженера та Цезаря	Лабораторн а робота	Сайт курсу	2	кінець поточного тижня
14	Системи контролю версій. GIT. Взаємодія у написанні коду на великих проєктах	Лекція	Сайт курсу	2	кінець поточного тижня
14	Лаб. 13. Запуск першого коду в систему контролю версій Github	Лабораторн а робота	Сайт курсу	2	кінець поточного тижня
15	Популярні мови програмування. Рейтинг найзатребуваніших спеціалістів	Лекція	[11],[12] Сайт курсу	2	кінець поточного тижня

15	Підсумкове заняття ЗМ 2	Лабораторна робота	Сайт курсу	2	кінець поточного тижня
16	Управління часом. Вчимося вчитись та чому час грає проти нас? Планування часу. Чому це важливо? Як знайти час та чому його не вистачає?	Лекція	Сайт курсу	2	кінець поточного тижня
16	Підсумкове заняття	Лабораторна робота	Сайт курсу	2	кінець поточного тижня

Сайт курсу: <https://moodle.elct.lnu.edu.ua/course/view.php?id=162>