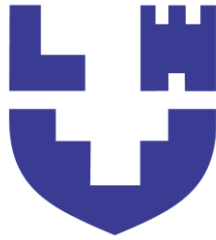


Міністерство освіти і науки України
Луцький національний технічний університет



Проектування інформаційних систем

Методичні вказівки до самостійної роботи
для здобувачів першого (бакалаврського) рівня вищої освіти
освітньої програми «Комп'ютерні науки»
галузі знань 12 (F) Інформаційні технології
спеціальності 122 (F3) Комп'ютерні науки
денної та заочної форм навчання

Луцьк 2026

УДК 004.65 (07)

П97

Рекомендовано до видання вченою радою факультету КІТ ЛНТУ,
протокол № від « ___ » 2026 року

Голова вченої ради факультету КІТ

І.С. Кондіус

Електронна копія друкованого видання передана для внесення в репозитарій ЛНТУ

Директор бібліотеки

Н. П. Поліщук

Розглянуто і схвалено на засіданні кафедри комп'ютерних наук
ЛНТУ, протокол № від « » 2026 року

Завідувач кафедри КН

В. О. Ліщина

Укладач:

В. А. Кошелюк, кандидат технічних наук,
доцент кафедри комп'ютерних наук ЛНТУ

Рецензент:

С. В. Лавренчук, кандидат технічних наук,
доцент кафедри комп'ютерної інженерії та
безпеки ЛНТУ

Відповідальний
за випуск:

В. О. Ліщина, кандидат технічних наук,
завідувач кафедри комп'ютерних наук ЛНТУ

П97

Проектування інформаційних систем: методичні вказівки до самостійної роботи для здобувачів першого (бакалаврського) рівня вищої освіти освітньої програми «Комп'ютерні науки» галузі знань 12 (F) Інформаційні технології спеціальності 122 (F3) Комп'ютерні науки денної та заочної форм навчання / уклад. В.А. Кошелюк – Луцьк: ЛНТУ, 2026. 28 с.

Методичні вказівки для самостійної роботи спрямовано на ґрунтовну підготовку здобувачів освіти з курсу «Проектування інформаційних систем». У ньому визначено мету, завдання курсу, а також наведено перелік необхідних інформаційних джерел, які допоможуть ефективно опрацювати, винесених на самостійне вивчення, матеріал.

Призначене для здобувачів першого (бакалаврського) рівня вищої освіти спеціальності 122 (F3) Комп'ютерні науки.

ВСТУП

Методичні вказівки до самостійної роботи з дисципліни «Проектування інформаційних систем» спрямовані на формування у здобувачів освіти системного підходу до аналізу, розроблення та впровадження сучасних інформаційних систем. У сучасних умовах цифрової трансформації суспільства та бізнесу зростає потреба у фахівцях, здатних ефективно проектувати інформаційні рішення, враховуючи як технічні, так і організаційні аспекти.

Самостійна робота є важливою складовою навчального процесу, оскільки сприяє поглибленню теоретичних знань, набуттю практичних навичок та розвитку критичного мислення. У межах даної дисципліни вона передбачає опрацювання теоретичного матеріалу, виконання практичних завдань, аналіз предметних областей, моделювання бізнес-процесів, а також розроблення структурних і функціональних моделей інформаційних систем.

Основною метою цих методичних вказівок є надання студентам чітких рекомендацій щодо організації самостійної роботи, послідовності виконання завдань та критеріїв оцінювання результатів. Вказівки містять опис ключових тем курсу, перелік завдань для самостійного опрацювання, вимоги до оформлення робіт та рекомендації щодо використання сучасних інструментів і технологій проектування.

Особлива увага приділяється застосуванню стандартів і методологій проектування інформаційних систем, таких як структурний та об'єктно-орієнтований підходи, використання UML-діаграм, CASE-засобів, а також принципів життєвого циклу програмного забезпечення. Це дозволяє студентам отримати комплексне уявлення про процес створення інформаційних систем від постановки задачі до впровадження та супроводу.

Застосування цих методичних вказівок сприятиме формуванню професійних компетентностей, необхідних для успішної діяльності у сфері інформаційних технологій, а також підготує студентів до виконання складних практичних і науково-дослідних завдань.

ЗМІСТ

1.	Мета та завдання дисципліни.....	5
2.	Загальні вказівки щодо організації самостійної роботи.....	7
3.	Тематичне планування самостійної роботи.....	10
4.	Тестові завдання для самоконтролю.....	16
5.	Підготовка до семестрового контролю.....	21
6.	Список використаних джерел.....	25

I. МЕТА ТА ЗАВДАННЯ ДИСЦИПЛІНИ

Освітня компонента «Проектування інформаційних систем» має важливе значення в теоретичній підготовці майбутніх фахівців і є обов'язковою компонентою підготовки здобувачів першого (бакалаврського) рівня вищої освіти ОПП «Комп'ютерні науки» спеціальності 122 «Комп'ютерні науки».

Метою вивчення дисципліни «Проектування інформаційних систем» є формування у здобувачів вищої освіти теоретичних знань та практичних навичок щодо змісту технологічних операцій створення ІС на різних рівнях ієрархії, а також засобів автоматизації проектних робіт, формалізації процесу проектування та методів управління проектуванням ІС для забезпечення базової профілюючої підготовки за фахом.

Основними завданнями вивчення дисципліни є: формування у майбутніх фахівців знань про основні підходи, принципи, технології, інструментальні засоби, шаблони та стандарти проектування інформаційних систем. Вивчення дисципліни «Проектування інформаційних систем» передбачає формування у студентів теоретичних знань і практичних навичок з розроблення, моделювання та впровадження сучасних інформаційних систем на базі хмарних сервісів Amazon Web Services (AWS). Особливу увагу приділено принципам архітектурного проектування, безпеки, масштабованості та автоматизації процесів у середовищі AWS, що забезпечує підготовку фахівців до роботи в реальних умовах хмарної інфраструктури.

Предметом дисципліни є процеси, методи та інструменти проектування інформаційних систем із використанням сервісів AWS – від аналізу вимог та вибору архітектури до розгортання, тестування й підтримки систем у хмарному середовищі. До предметної області належать такі сервіси, як Amazon EC2, S3, RDS, Lambda, CloudFormation, IAM, VPC, CloudWatch тощо.

Найменування та опис компетентностей, формування котрих забезпечує вивчення дисципліни:

– інтегральна компетентність:

здатність розв'язувати складні спеціалізовані задачі та практичні проблеми у галузі комп'ютерних наук або у процесі навчання, що передбачає застосування теорій та методів інформаційних технологій і характеризується комплексністю та невизначеністю умов;

– загальні компетентності:

ЗК1. Здатність до абстрактного мислення, аналізу та синтезу.

ЗК2. Здатність застосовувати знання у практичних ситуаціях.

ЗК3. Знання та розуміння предметної області та розуміння професійної діяльності.

ЗК6. Здатність вчитися й оволодівати сучасними знаннями.

– спеціальні (фахові, предметні) компетентності:

СК8. 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.

СК10. 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.

Результати навчання. Результати навчання вивчення дисципліни базуються на програмних результатах навчання:

ПР8. Використовувати методологію системного аналізу об'єктів, процесів і систем для задач аналізу, прогнозування, управління та проектування динамічних процесів в макроекономічних, технічних, технологічних і фінансових об'єктах.

ПР11. Володіти навичками управління життєвим циклом програмного забезпечення, продуктів і сервісів інформаційних технологій відповідно до вимог і обмежень замовника, вміти розробляти проектну документацію (техніко-економічне обґрунтування, технічне завдання, бізнес-план, угоду, договір, контракт).

ПР14. 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.

Дисципліна забезпечує комплексне розуміння процесу проектування сучасних інформаційних систем у хмарних середовищах і сприяє формуванню компетенцій майбутніх фахівців у сфері ІТ-архітектури та управління інфраструктурою.

Знання з дисципліни «Проектування інформаційних систем» становлять основу для подальшого поглибленого засвоєння матеріалу з того чи іншого розділу. З позицій випереджаючої освіти навчання тільки за конспектом лекцій і основною літературою, зазначеною у навчальній програмі, є недостатнім. У більшості випадків належна підготовка потребує вмінь швидко знаходити та опрацьовувати необхідний матеріал за першоджерелами, науковою і спеціальною літературою та коректно цитувати знайдене. Перелік такої літератури, як правило, наводиться у навчально-методичному комплексі навчальної дисципліни. Тому завдання студента зводиться до самостійного знаходження цих матеріалів шляхом пошуку у паперових або електронних фондах бібліотек.

II. ЗАГАЛЬНІ ВКАЗІВКИ ЩОДО ОРГАНІЗАЦІЇ САМОСТІЙНОЇ РОБОТИ

Самостійна робота студента під час лекції

Лекційний матеріал призначається для спрямування студентів у найбільш раціональному напрямі щодо вивчення навчальної дисципліни і акцентуванні уваги на найбільш складних, вузлових питаннях навчальної дисципліни. Належне ведення конспекту під час лекції сприяє збереженню необхідної інформації та дає студенту змогу в подальшому проаналізувати її. За умови подання лекційного матеріалу в усній формі одночасно засво-юється до 20% інформації.

Робота над конспектами лекцій, планами лабораторних занять

При підготовці до практичних занять студент має спиратися на складений ним конспект лекції. При опрацюванні матеріалу лекції потрібно зіставити законспектований матеріал з планом практичного заняття, що міститься у методичних матеріалах для практичних занять або у навчально-методичному комплексі. Якщо у конспекті бракує матеріалу з окремих питань лекції або недостатньо розкриті деякі питання практичного заняття, або вони винесені на самостійне опрацювання, студент повинен звернутися до рекомендованих підручників, навчальних посібників і відповідних методичних матеріалів. Підготовку для лабораторного заняття найкраще здійснювати з використанням ПК зі встановленим на ньому відповідним програмним забезпеченням.

Поради із планування й організації часу, необхідного для виконання самостійної роботи

Раціональне планування і організація самостійної роботи студентів є найважливішою умовою її ефективності.

Планування самостійної роботи направлено на формування логічно вибудованої, прозорої, зрозумілої, доступної і ефективної системи організації самостійної роботи та її оцінки.

При цьому необхідно пам'ятати, що самостійна робота студентів виконує в навчальному процесі кілька функцій:

- розвиваючу (підвищення культури розумової праці, привчання до творчих видів діяльності, вдосконалення інтелектуальних здібностей студентів);
- інформаційно-навчальну (навчальна діяльність на аудиторних заняттях, непідкріплена самостійною роботою, стає мало результативною);
- орієнтуючу і стимулюючу (процесу навчання надається прискорення і мотивація);
- виховну (формується і розвиваються професійні якості фахівця);

– дослідницьку (новий рівень професійно-творчого мислення).

В основі самостійної роботи студентів лежать наступні принципи: розвиток творчої діяльності, цільове планування, особистісно-діяльнісний підхід.

Самостійну роботу можна назвати ефективною тільки в тому випадку, якщо вона організована і реалізується в освітньому процесі як цілісна система на всіх етапах навчання.

Можна виділити кілька об'єктивних закономірностей організації самостійної роботи студентів:

- творча складова самостійної роботи зростає в міру навчання;
- в процесі організації самостійної роботи виникає потреба в методичному забезпеченні;
- застосування інформаційних технологій стає частиною організації і моніторингу самостійної роботи студентів на всіх її етапах.

У процесі самостійної роботи студент набуває навиків самоорганізації, самоконтролю, самоврядування, саморефлексії і стає активним самостійним суб'єктом навчальної діяльності.

Самостійна робота повинна давати важливий вплив на формування особистості майбутнього фахівця. Кожен, хто навчається самостійно планує режим своєї роботи з урахуванням часу роботи бібліотеки, профільних лабораторій, комп'ютерних класів і т.п. Він виконує самостійну роботу за особистим індивідуальним планом, в залежності від його підготовки, часу та інших умов.

Першим завданням в організації позааудиторної самостійної роботи є складання розкладу, що відображає час занять і їх характер, перерви на обід, вечеря, відпочинок, сон, проїзд і т.п. Із самого початку студенту не потрібно прагнути робити відразу найважчу її частину. Доцільно вибрати щось середнє за складністю. Після цього, перейти до більш важкої роботи, легке залишивши наостанок. Розумову працю необхідно не тільки правильно організувати, а й стимулювати. Важливо вміти підтримувати стійку увагу до досліджуваного матеріалу. Вироблення уваги вимагає значних вольових зусиль від студента. Стійка увага з'являється тоді, коли людина ставиться до справи з інтересом.

Слід правильно організувати свої заняття за часом: 50 хвилин – робота, 5-10 хвилин – перерва, після 3 годин роботи перерва – 20-25 хвилин. Інакше наростаюча втома спричинить нестійкість уваги. Організація активного відпочинку передбачає чергування розумової та фізичної діяльності, що відновлює працездатність людини.

Робота з бібліотечними фондами та дистанційними джерелами з метою пошуку необхідної інформації

Найважливішим інформаційним джерелом вивчення навчальної дисципліни «Проектування інформаційних систем» є ресурси мережі Інтернет. Основна частина матеріалу в Інтернеті розрахована на професіоналів, тому при вивченні навчальної дисципліни спочатку необхідно користуватися літературою навчального характеру.

При опрацюванні матеріалу потрібно дотримуватись таких правил:

- зосередитися на тому, що читаєш;
- виділити головну думку автора;
- виділити основні питання тексту від другорядних;
- зрозуміти думку автора чітко і ясно, що допоможе виробити власну думку;
- уявити ясно те, що читаєш.

У процесі роботи над темою тлумачення незнайомих слів і спеціальних термінів слід знаходити у фаховій літературі, термінологічних словниках. Незрозумілі місця, фрази, вирази доречно перечитувати декілька разів, щоб зрозуміти їх зміст.

Після прочитання тексту необхідно:

- усвідомити зв'язок між теоретичними положеннями і практикою;
- закріпити прочитане у свідомості;
- пов'язати нові знання з попередніми у даній галузі;
- перейти до заключного етапу засвоєння і опрацювання – записам.

Складання плану, або тез логічно закінченого за змістом уривка тексту, сприяє кращому його розумінню. План може бути простий або розгорнутий, тобто більш поглиблений, особливо при опрацюванні додаткової літератури за даною темою. Записи необхідно вести розбірливо і чітко. Вони можуть бути короткі або розгорнуті залежно від рівня знань студента, багатства його літературної і професійної лексики, навичок самостійної роботи з книгою.

Знання з дисципліни «Проектування інформаційних систем» становлять основу для подальшого поглибленого засвоєння матеріалу з того чи іншого розділу. З позицій випереджаючої освіти навчання тільки за конспектом лекцій і основною літературою, зазначеною у навчальній програмі, є недостатнім. У більшості випадків належна підготовка потребує вміння швидко знаходити та опрацьовувати необхідний матеріал за першоджерелами, науковою і спеціальною літературою та коректно цитувати знайдене. Перелік такої літератури, як правило, наводиться у навчально-методичному комплексі навчальної дисципліни. Тому завдання студента зводиться до самостійного знаходження цих матеріалів шляхом пошуку у паперових або електронних фондах бібліотек.

III. ТЕМАТИЧНЕ ПЛАНУВАННЯ САМОСТІЙНОЇ РОБОТИ

САМОСТІЙНА РОБОТА № 1

Тема. Проектування IT-інфраструктури.

Мета: формування у студентів теоретичних знань і практичних навичок щодо принципів, методів та інструментів проектування IT-інфраструктури, зокрема аналізу вимог, вибору архітектурних рішень, планування ресурсів, забезпечення надійності, масштабованості та безпеки інформаційних систем.

Завдання: опрацювати матеріал та дати відповіді на запитання.

Рекомендована література: [1; 3; 7; 13; 19; 22].

Запитання для самоконтролю

1. Що розуміють під терміном IT-інфраструктура, і які її основні компоненти?
2. Які типи серверів та мережевого обладнання зазвичай використовуються при побудові корпоративної IT-інфраструктури?
3. Поясніть відмінність між on-premises, cloud та hybrid інфраструктурами.
4. Які критерії слід враховувати при виборі апаратного забезпечення для IT-інфраструктури?
5. Що таке мережевий дизайн та які основні топології мереж використовуються в корпоративних системах?
6. Які методи забезпечення відмовостійкості та резервування застосовують у проектуванні IT-інфраструктури?
7. Поясніть принципи віртуалізації серверів і мереж та їх роль у проектуванні сучасної інфраструктури.
8. Що включає в себе планування масштабованості IT-інфраструктури та чому це важливо?
9. Які інструменти або технології використовують для моніторингу та управління IT-інфраструктурою?
10. Назвіть основні етапи проектування IT-інфраструктури для організації та поясніть їх послідовність.

САМОСТІЙНА РОБОТА № 2

Тема. Аналіз IT-інфраструктури та її елементів, як об'єктів проектування та управління.

Мета: дослідження IT-інфраструктури та її основних компонентів як об'єктів проектування і управління, а також аналіз їх структури, взаємодії та функціональних характеристик у процесі створення інформаційних систем.

Завдання: опрацювати матеріал та дати відповіді на запитання.

Рекомендована література: [2; 4; 8; 14; 20; 21].

Запитання для самоконтролю

1. Що таке IT-інфраструктура і які основні її складові?
2. Які етапи аналізу IT-інфраструктури використовуються при проектуванні інформаційних систем?
3. Як класифікуються елементи IT-інфраструктури за функціональним призначенням?
4. У чому полягає роль серверного обладнання та систем зберігання даних у загальній архітектурі IT-інфраструктури?
5. Як мережеві компоненти впливають на продуктивність та надійність інформаційної системи?
6. Які методи оцінки ефективності та надійності IT-інфраструктури можна застосовувати на етапі проектування?
7. Як аналіз IT-інфраструктури допомагає визначити критичні точки для управління ризиками?
8. В чому полягає взаємозв'язок між апаратними, програмними та організаційними елементами IT-інфраструктури?
9. Які інструменти та методи моделювання використовуються для аналізу та оптимізації IT-інфраструктури?
10. Як результати аналізу IT-інфраструктури впливають на процес ухвалення рішень щодо її розвитку та модернізації?

САМОСТІЙНА РОБОТА № 3

Тема. Управління IT-інфраструктурою.

Мета: ознайомлення з основами управління IT-інфраструктурою та формування базових навичок її планування, організації й підтримки ефективної роботи в інформаційних системах.

Завдання: опрацювати матеріал та дати відповіді на запитання.

Рекомендована література: [3; 5; 9; 14; 15; 20].

Запитання для самоконтролю

1. Що розуміють під терміном IT-інфраструктура та які її основні компоненти?
2. Які моделі управління IT-інфраструктурою існують і чим вони відрізняються?
3. Опишіть роль сервісного підходу (IT Service Management, ITSM) в управлінні IT-інфраструктурою.
4. Які інструменти моніторингу та контролю IT-ресурсів найбільш поширені?

5. Що таке інвентаризація ІТ-ресурсів, і чому вона важлива для управління інфраструктурою?
6. Які принципи автоматизації управління ІТ-інфраструктурою та які її переваги?
7. Як забезпечується безпека та відмовостійкість ІТ-інфраструктури?
8. Що таке віртуалізація та хмарні сервіси, і яку роль вони відіграють у сучасній ІТ-інфраструктурі?
9. Опишіть процес планування розвитку ІТ-інфраструктури у відповідності до потреб організації.
10. Які метрики ефективності управління ІТ-інфраструктурою використовуються для оцінки її стану та продуктивності?

САМОСТІЙНА РОБОТА № 4

Тема. Проектування окремих рівнів ІТ-інфраструктури.

Мета: формування навичок проектування окремих рівнів ІТ-інфраструктури з урахуванням ефективності, безпеки та масштабованості.

Завдання: опрацювати матеріал та дати відповіді на запитання.

Рекомендована література: [4; 5; 6; 12; 18; 19].

Запитання для самоконтролю

1. Що таке ІТ-інфраструктура і які основні її рівні виділяють при проектуванні?
2. Які функції виконує рівень апаратного забезпечення (Hardware) в ІТ-інфраструктурі?
3. Що входить до складу рівня програмного забезпечення (Software) і як його компоненти взаємодіють між собою?
4. Поясніть принципи проектування мережевого рівня ІТ-інфраструктури.
5. Які критерії вибору серверного обладнання для корпоративної ІТ-інфраструктури?
6. Як забезпечується масштабованість і відмовостійкість окремих рівнів ІТ-інфраструктури?
7. Що таке віртуалізація, і яку роль вона відіграє на рівні апаратних та програмних ресурсів?
8. Опишіть основні методи захисту інформації на різних рівнях ІТ-інфраструктури.
9. Які існують підходи до інтеграції хмарних сервісів у корпоративну ІТ-інфраструктуру?
10. Як проектування ІТ-інфраструктури впливає на ефективність роботи інформаційних систем організації?

САМОСТІЙНА РОБОТА № 5

Тема. Об'єкти проектування та управління IT-інфраструктури

Мета: ознайомитися з об'єктами проектування та принципами управління IT-інфраструктури, сформувані практичні навички їх аналізу, моделювання та оптимізації для забезпечення ефективної роботи інформаційних систем.

Завдання: опрацювати матеріал та дати відповіді на запитання.

Рекомендована література: [3; 5; 11; 13; 14; 18].

Запитання для самоконтролю

1. Що розуміють під об'єктами проектування IT-інфраструктури? Наведіть приклади.
2. Які основні компоненти IT-інфраструктури слід враховувати при проектуванні інформаційних систем?
3. В чому полягає різниця між апаратними та програмними об'єктами управління в IT-інфраструктурі?
4. Що таке серверна архітектура та які її ключові елементи необхідно враховувати при проектуванні?
5. Опишіть поняття мережевої інфраструктури та поясніть, які мережеві елементи потребують управління.
6. Які методи управління IT-інфраструктурою існують і як вони впливають на ефективність інформаційної системи?
7. Що таке сервісна модель (SaaS, PaaS, IaaS) та як вона використовується при проектуванні IT-інфраструктури?
8. Які критерії ефективності управління IT-інфраструктурою слід враховувати при розробці системи?
9. Поясніть поняття віртуалізації і її роль у проектуванні та управлінні IT-ресурсами.
10. Як забезпечується надійність та відмовостійкість IT-інфраструктури, і які елементи проектування впливають на це?

САМОСТІЙНА РОБОТА № 6

Тема. Основні підходи до побудови ПЗ управління IT-інфраструктурою.

Мета: ознайомитися з основними підходами до проектування та побудови програмного забезпечення для управління IT-інфраструктурою, оцінити їхні переваги та застосування на практиці.

Завдання: опрацювати матеріал та дати відповіді на запитання.

Рекомендована література: [2; 3; 6; 13; 17; 21].

Запитання для самоконтролю

1. Що таке ПЗ управління IT-інфраструктурою і які його основні цілі?
2. Які основні підходи до побудови ПЗ управління IT-інфраструктурою існують?
3. В чому полягає відмінність між централізованим та децентралізованим підходом управління IT-інфраструктурою?
4. Які функціональні компоненти зазвичай включає система управління IT-інфраструктурою?
5. Що таке автоматизація управління IT-інфраструктурою і які переваги вона дає?
6. Як роль оркестрації (orchestration) відрізняється від простого моніторингу в системах управління?
7. Назвіть популярні програмні продукти або платформи для управління IT-інфраструктурою та їхні ключові особливості.
8. Які основні методи інтеграції ПЗ управління IT-інфраструктурою з існуючими системами підприємства?
9. Як підходи до побудови ПЗ управління впливають на безпеку та надійність IT-інфраструктури?
10. Які сучасні тенденції розвитку ПЗ управління IT-інфраструктурою (наприклад, хмарні сервіси, AI/ML, DevOps-підхід)?

САМОСТІЙНА РОБОТА № 7

Тема. Використання Dockerfile. Створення образів. Взаємозв'язок з контейнером.

Мета: ознайомитися із принципами створення Dockerfile, формуванням образів та їх взаємодією з контейнерами для ефективного проєктування та розгортання інформаційних систем.

Завдання: опрацювати матеріал та дати відповіді на запитання.

Рекомендована література: [1; 2; 7; 8; 14; 16; 20].

Запитання для самоконтролю

1. Що таке Dockerfile і яку роль він виконує у створенні Docker-образів?
2. Які основні інструкції Dockerfile ви знаєте і для чого вони використовуються (наприклад, FROM, RUN, COPY, CMD)?
3. Поясніть, що таке Docker-образ і чим він відрізняється від контейнера.
4. Як створити Docker-образ із Dockerfile за допомогою команди docker build? Наведіть приклад.
5. Що відбувається під час створення контейнера з існуючого Docker-образу?

6. Як можна переглянути список усіх наявних Docker-образів та контейнерів у системі?
7. Поясніть, що таке шари образу (image layers) і як вони впливають на ефективність зберігання та розгортання.
8. Як у Dockerfile реалізується передача файлів та налаштувань до контейнера?
9. Що таке CMD і ENTRYPOINT у Dockerfile та у чому різниця між ними?
10. Як відстежувати взаємозв'язок між контейнером і образом, наприклад, при внесенні змін у контейнер?

САМОСТІЙНА РОБОТА № 8

Тема. Інфраструктура, як код. Terraform.

Мета: вивчити принципи «Інфраструктури як код» та практично застосувати Terraform для автоматизації створення та управління інфраструктурою.

Завдання: опрацювати матеріал та дати відповіді на запитання.

Рекомендована література: [3; 5; 8; 9; 10; 22].

Запитання для самоконтролю

1. Що означає термін «Інфраструктура як код» (IaC) і які основні переваги цього підходу?
2. Які відмінності між декларативним та імперативним підходами при описі інфраструктури?
3. Що таке Terraform, і яку роль він відіграє у впровадженні IaC?
4. Які основні компоненти конфігурації Terraform (наприклад, provider, resource, variable) і яку функцію вони виконують?
5. Поясніть різницю між командами Terraform: terraform init, terraform plan, terraform apply, terraform destroy.
6. Що таке state-файл Terraform і чому важливо його правильно зберігати та контролювати?
7. Як у Terraform реалізується повторне використання конфігурацій за допомогою modules?
8. Які типи провайдерів підтримує Terraform і як вибір провайдера впливає на управління інфраструктурою?
9. Як Terraform забезпечує відстеження змін інфраструктури і конфліктів при спільній роботі декількох користувачів?
10. Наведіть приклади практичних сценаріїв використання Terraform у проєктах розгортання інформаційних систем.

IV. ТЕСТОВІ ЗАВДАННЯ ДЛЯ САМОКОНТРОЛЮ

1. Який підхід найкраще підходить для систем із високою невизначеністю вимог та частими змінами?
 - A. Waterfall
 - B. V-model
 - C. Agile (Scrum)
 - D. Spiral без ітерацій
2. У контексті Domain-Driven Design (DDD), що означає "Bounded Context"?
 - A. Межі бази даних
 - B. Контекст, у якому модель має чітке значення
 - C. Обмеження апаратного забезпечення
 - D. Контроль доступу до системи
3. Яка проблема вирішується нормалізацією бази даних?
 - A. Збільшення швидкодії процесора
 - B. Усунення надлишковості даних
 - C. Покращення UI
 - D. Зменшення кількості користувачів
4. Який тип діаграми UML найкраще підходить для моделювання бізнес-процесів?
 - A. Діаграма класів
 - B. Діаграма діяльності
 - C. Діаграма розгортання
 - D. Діаграма компонентів
5. У чому головна відмінність між SOA та мікросервісною архітектурою?
 - A. SOA не використовує мережу
 - B. Мікросервіси мають менший рівень гранулярності та автономність
 - C. SOA не підтримує API
 - D. Мікросервіси не масштабуються
6. Який принцип CAP-теореми означає неможливість гарантувати одночасно узгодженість і доступність при мережевих збоях?
 - A. Consistency
 - B. Availability
 - C. Partition Tolerance
 - D. Trade-off між C та A
7. Який шаблон проектування використовується для розділення логіки представлення і бізнес-логіки?
 - A. Singleton
 - B. Observer
 - C. MVC

- D. Factory
8. Що є основною метою використання CQRS (Command Query Responsibility Segregation)?
- A. Зменшення кількості серверів
 - B. Розділення операцій читання і запису
 - C. Збільшення швидкості мережі
 - D. Зменшення обсягу пам'яті
9. Яка характеристика властива NoSQL базам даних?
- A. Строга схема
 - B. Вертикальне масштабування
 - C. Гнучка схема
 - D. Відсутність індексів
10. Який тип тестування перевіряє взаємодію між модулями?
- A. Unit testing
 - B. Integration testing
 - C. Acceptance testing
 - D. Load testing
11. Яка модель життєвого циклу передбачає оцінку ризиків на кожній ітерації?
- A. Waterfall
 - B. Spiral
 - C. Agile
 - D. RAD
12. Що означає принцип "Loose Coupling"?
- A. Сильну залежність між компонентами
 - B. Незалежність компонентів системи
 - C. Відсутність API
 - D. Високу складність коду
13. Яка метрика використовується для оцінки складності коду?
- A. Throughput
 - B. Latency
 - C. Cyclomatic Complexity
 - D. Bandwidth
14. Що є ключовою особливістю Event-Driven Architecture?
- A. Централізоване управління
 - B. Використання подій для комунікації
 - C. Відсутність масштабування
 - D. Жорстка структура даних
15. Який підхід найкраще підходить для побудови високонавантажених систем?
- A. Моноліт
 - B. Мікросервіси

- C. Локальні додатки
 - D. Desktop-рішення
16. Яка роль API Gateway у мікросервісній архітектурі?
- A. Зберігання даних
 - B. Єдина точка входу для клієнтів
 - C. Виконання бізнес-логіки
 - D. Управління БД
17. Що означає "idempotent" операція в REST?
- A. Виконується лише один раз
 - B. Дає однаковий результат при повторенні
 - C. Потребує авторизації
 - D. Працює лише з GET
18. Яка стратегія міграції передбачає повну переробку системи?
- A. Rehost
 - B. Refactor
 - C. Rebuild
 - D. Replace
19. Який інструмент використовується для контейнеризації додатків?
- A. Jenkins
 - B. Docker
 - C. Git
 - D. Maven
20. Яка основна мета DevOps практик?
- A. Зменшення кількості розробників
 - B. Автоматизація та інтеграція процесів розробки і експлуатації
 - C. Виключення тестування
 - D. Локалізація програм
21. Що таке REST API?
- A. База даних
 - B. Архітектурний стиль взаємодії систем
 - C. Мова програмування
 - D. Тип протоколу
22. Що є основною метою проектування інформаційної системи?
- A. Підвищення продуктивності комп'ютера
 - B. Створення ефективної системи обробки інформації
 - C. Розробка апаратного забезпечення
 - D. Зменшення вартості програмного забезпечення
23. Який етап передує проектуванню ІС?
- A. Тестування
 - B. Впровадження

- C. Аналіз вимог
 - D. Супровід
24. UML використовується для:
- A. Програмування
 - B. Моделювання систем
 - C. Адміністрування БД
 - D. Захисту інформації
25. Яка діаграма UML описує взаємодію об'єктів у часі?
- A. Діаграма класів
 - B. Діаграма послідовності
 - C. Діаграма компонентів
 - D. Діаграма станів
26. Що таке ER-діаграма?
- A. Метод тестування
 - B. Модель даних «сутність-зв'язок»
 - C. Мова програмування
 - D. Тип серверу
27. Який підхід передбачає розбиття системи на модулі?
- A. Монолітний
 - B. Модульний
 - C. Інтеграційний
 - D. Лінійний
28. Що описує функціональна модель системи?
- A. Структуру бази даних
 - B. Зовнішній вигляд інтерфейсу
 - C. Бізнес-процеси та функції
 - D. Апаратне забезпечення
29. Який методологічний підхід використовує ітерації?
- A. Waterfall
 - B. Agile
 - C. Cascade
 - D. Linear
30. Що є результатом логічного проєктування БД?
- A. SQL-запити
 - B. Таблиці та зв'язки
 - C. Програмний код
 - D. Інтерфейс користувача
31. Яка нормальна форма усуває транзитивні залежності?
- A. 1NF
 - B. 2NF

- C. 3NF
 - D. BCNF
32. Що таке use case?
- A. Тип даних
 - B. Сценарій взаємодії користувача із системою
 - C. Метод тестування
 - D. Програмний модуль
33. Яка діаграма відображає структуру класів та їх зв'язки?
- A. Діаграма станів
 - B. Діаграма класів
 - C. Діаграма діяльності
 - D. Діаграма розгортання
34. Що таке прототипування?
- A. Створення фінальної системи
 - B. Створення попередньої моделі системи
 - C. Тестування коду
 - D. Встановлення ПЗ
35. Який принцип означає приховування внутрішньої реалізації?
- A. Наслідування
 - B. Інкапсуляція
 - C. Поліморфізм
 - D. Абстракція
36. Що таке архітектура клієнт-сервер?
- A. Тип бази даних
 - B. Модель розподілу обчислень між клієнтом і сервером
 - C. Мова програмування
 - D. Тип мережі
37. Який документ описує вимоги до системи?
- A. Технічне завдання
 - B. Інструкція користувача
 - C. Журнал помилок
 - D. Код програми
38. Що таке CASE-засоби?
- A. Пристрої зберігання даних
 - B. Інструменти автоматизації розробки ІС
 - C. Мови програмування
 - D. Сервери

V. ПІДГОТОВКА ДО СЕМЕСТРОВОГО КОНТРОЛЮ

Завдання для підсумкового контролю знань. Підготовка до іспиту відбувається після закінчення теоретичної частини семестру. Час, відведений на підготовку та проведення семестрового контролю з дисциплін поточного семестру, формує сесію.

До іспиту допускаються студенти, які повністю виконали всі інші види навчальної роботи, які передбачено навчальним планом з цієї дисципліни. Іспит проводиться в очній формі. На іспит виносяться питання, кожне з яких оцінюється у 100 балів, а результуюча як середня зважена оцінка. Питання, які виносяться на іспит, формується на основі теоретичного курсу та самостійної роботи студента.

Критерії оцінювання відповіді. Підсумкова оцінка (за 100-бальною шкалою) з дисципліни «Проектування інформаційних систем» визначається як середньозважена величина, залежно від питомої ваги кожної складової залікового кредиту. Кожна самостійна робота та модульне тестування оцінюється у 100-бальній системі. Підсумковий бал (за 100-бальною шкалою) з дисципліни «Проектування інформаційних систем» визначається як середньозважена величина, залежно від вагових коефіцієнтів кожної складової освітньої компоненти.

	Поточний контроль	Модульний контроль		Підсумковий контроль	Підсумкова оцінка
	Змістовий модуль 1	Змістовий модуль 2	Заліковий модуль 3	екзамен	
Ваговий коефіцієнт	30%	15%	15%	40%	100%
Максимальна кількість балів (за 100-бальною шкалою)	100	100	100	100	100

Оцінювання результатів навчання здобувачів вищої освіти орієнтоване на заохочення їх до активної участі у забезпеченні якості освітнього процесу. Мета оцінювання результатів навчання здобувачів вищої освіти:

- підтримка здобувачів вищої освіти у забезпеченні можливості досягнення програмних результатів навчання через наявність ефективного зворотного зв'язку;
- кількісна оцінка рівня досягнення результатів навчання;
- розвиток у здобувачів вищої освіти здатності самооцінювання для забезпечення їх ефективного подальшого навчання.

Поточна перевірка знань студентів з питань, що виносяться на самостійне опрацювання, відбувається з використанням автоматизованої системи тестування. Тестування студента відбувається в складі групи в присутності

викладача. Оцінка виставляється автоматично. Результати доводяться студентам лише після завершення тестування всієї групи.

Відповіді студентів оцінюються автоматично від 0 до 100 балів. Оцінка розраховується як сума балів отриманих студентами за правильні відповіді на запитання тесту. Кількість балів, в яку оцінюється кожна правильна відповідь, залежить від складності запитання. Мінімальну суму – 0 балів – одержують студенти, які не вказали жодної правильної відповіді. Максимальна сума – 100 балів – виставляється, якщо всі вказані відповіді були правильними.

Критерії підсумкового оцінювання:

90 – 100 балів отримує здобувач освіти, який вільно володіє програмним обсягом матеріалу, виявляє і демонструє особисті творчі здібності, вміє самостійно здобувати нові знання, демонструє ґрунтовні знання, вміння та практичні навички; без допомоги викладача знаходить та опрацьовує необхідну інформацію, використовує набуті знання і вміння для прийняття рішень у нестандартних ситуаціях, переконливо аргументує відповіді, вміє використовувати методи наукового обґрунтування власних рішень, самостійно розкриває власні обдарування й нахили.

85 – 89 балів отримує здобувач освіти, який вільно володіє програмним обсягом матеріалу, застосовує його на практиці, вільно розв'язує вправи і задачі у стандартних ситуаціях, самостійно виправляє допущені помилки, кількість яких незначна, вміє обґрунтувати на аргументувати свою думку.

75 – 84 балів отримує здобувач освіти, який вміє зіставляти, узагальнювати, систематизувати інформацію під керівництвом викладача, в цілому, самостійно застосовувати її на практиці; контролювати власну діяльність; виправляти помилки, серед яких є суттєві, добирати окремі аргументи для підтвердження своїх думок.

65 – 74 балів отримує здобувач освіти, який відтворює значну частину теоретичного матеріалу, демонструє знання і розуміння основних положень з допомогою викладача, поверхнево відтворює і аналізує навчальний матеріал, виправляє помилки, серед яких є значна кількість суттєвих.

60 – 64 балів отримує здобувач освіти, який володіє навчальним матеріалом на рівні, вищому за початковий, значну частину його відтворює на репродуктивному рівні або володіє матеріалом на рівні окремих фрагментів, що становлять незначну частину навчального матеріалу.

35 – 59 балів отримує здобувач освіти, який володіє матеріалом на рівні окремих фрагментів, що становлять незначну частину навчального матеріалу.

0 – 34 балів отримує здобувач освіти, який володіє матеріалом на рівні елементарного розпізнавання і відтворення окремих фактів, елементів, об'єктів.

Шкала оцінювання:

За шкалою ЛНТУ	За національною шкалою	За шкалою ECTS
90–100	відмінно	A (відмінно)
85–89	добре	B (дуже добре)
75–84		C (добре)
65–74		D (задовільно)
60–64	задовільно	E (достатньо)
35–59		FX (незадовільно з можливістю повторного складання)
1–34	незадовільно	F (незадовільно з обов'язковим повторним курсом)

Перелік питань, які виносяться на іспит

1. Що таке інформаційна система і які її основні компоненти?
2. Поясніть поняття «життєвий цикл інформаційної системи».
3. Які типи інформаційних систем існують?
4. В чому різниця між автоматизованою і комп'ютерною інформаційною системою?
5. Що таке проектна документація інформаційної системи?
6. Основні принципи проектування ІС.
7. Поясніть різницю між системним і модульним підходом у проектуванні.
8. Які основні задачі аналізу вимог?
9. Що таке функціональні та нефункціональні вимоги?
10. Роль користувача в процесі проектування ІС.
11. Що таке модель інформаційної системи?
12. Основні види моделей ІС: концептуальні, логічні та фізичні.
13. Що таке ER-діаграма і для чого вона використовується?
14. Основні елементи ER-діаграми: сутність, атрибут, зв'язок.
15. Поясніть поняття «нормалізація даних» і її етапи.
16. Що таке DFD (Data Flow Diagram)?
17. Поясніть рівні DFD (контекст, рівень 0, деталізація).
18. Що таке UML-діаграми і їх призначення?
19. Різниця між класовою діаграмою і діаграмою послідовності в UML.
20. Як моделюються бізнес-процеси у BPMN?
21. Поясніть різницю між клієнт-серверною та багаторівневою архітектурою.
22. Що таке база даних і її основні типи?
23. Поясніть роль сервера додатків у ІС.
24. Що таке веб-сервіси та їхні типи (SOAP, REST)?
25. Основні компоненти програмного забезпечення ІС.
26. Що таке middleware і яку функцію він виконує?
27. Поясніть поняття «інтеграція систем».
28. Які існують способи забезпечення масштабованості ІС?
29. Поясніть різницю між централізованою та децентралізованою системою.

30. Роль API у сучасних інформаційних системах.
31. Основні принципи проектування баз даних.
32. Що таке реляційна модель даних?
33. Основні типи зв'язків між таблицями (1:1, 1:M, M:M).
34. Що таке первинний і зовнішній ключ?
35. Поясніть поняття індексації і її значення.
36. Що таке транзакція в базі даних?
37. Основні властивості транзакцій (ACID).
38. Різниця між SQL і NoSQL базами даних.
39. Що таке нормалізація і денормалізація?
40. Методи забезпечення безпеки баз даних.
41. Основні принципи розробки інтерфейсів користувача.
42. Що таке прототипування інтерфейсу?
43. Поясніть концепцію UX/UI у проектуванні ІС.
44. Методи тестування користувацьких інтерфейсів.
45. Що таке управління конфігураціями в ІС?
46. Поясніть поняття «життєвий цикл програмного продукту».
47. Роль систем автоматизованого управління проектами (PMS).
48. Що таке контроль якості програмного забезпечення?
49. Основні стандарти проектування ІС (ISO, IEEE).
50. Які сучасні тренди у проектуванні інформаційних систем?
51. Системний підхід до проектування програмного забезпечення.
52. Класифікація інформаційних систем.
53. Процес створення інформаційних систем.
54. Етапи створення ІС.
55. Життєвий цикл програмного забезпечення
56. Моделі життєвого циклу програмного забезпечення
57. Методологія та методи проектування інформаційних систем.
58. Основні методології проектування інформаційних систем.
59. CASE-технології аналізу та проектування.
60. Інструментальні засоби проектування ІС
61. Case-засобів для побудови діаграм UML.
62. Інструментальні засоби проектування ІС
63. Практика застосування UML. Вплив UML на процес розробки.
64. Шаблони проектування програмного забезпечення
65. API : архітектурні стилі, автентифікація, використання
66. Обмеження частоти запитів API.
67. Робота з Webhooks. Пошук та усунення проблем, пов'язаних з API.
68. Технологія контейнеру для розгортання додатку. Безперервна інтеграція/безперервне розгортання (CI/CD).
69. Основні принципи DevOps. Dev-Ops як метод автоматизації процесів розробки.
70. Основи написання скриптів автоматизації.
71. Інструменти автоматизації : ansible, puppet, chef.
72. Інфраструктура як код. Автоматизація тестування.

VI. СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Andrushchak I., Koshelyuk V., Kominko V., Shepeliuk S., Levchuk M. SEO analysis of modern and current software products. The 9th International scientific and practical conference «Scientists and existing problems of human development». Zagreb, Croatia. International Science Group. 2023. pp. 348-352.
2. Dennis, A., Wixom, B.H., Tegarden, D. Systems Analysis and Design: An Object-Oriented Approach with UML. 8th Edition: Wiley, 2024. 544 p.
3. INCOSE. Systems Engineering Handbook. 5th Edition: Wiley, 2023. 211 p.
4. Issa, T., et al. Management Information Systems: Managing the Digital World: Springer, 2024. 57 p.
5. Kendall, K.E., Kendall, J.E. Systems Analysis and Design. 11th Edition: Pearson, 2023. 609 p.
6. Kleppmann, M. Designing Data-Intensive Applications: The Big Ideas Behind Reliable, Scalable, and Maintainable Systems. 2nd Edition: O'Reilly Media, 2024. 590 p.
7. Martin, R.C. Clean Architecture: A Craftsman's Guide to Software Structure and Design (Updated Edition). Prentice Hall, 2023. 420 p.
8. Martsenyuk V.P., Sverstyuk A.S., Andrushchak I.Ye., Kosheliuk V.A., Matviiv Yu.Ya. Information modeling of the calculation of composite bodies with cracks by the joint action of mechanical and thermal loads: Monographю Lutsk: RVV Lutsk NTU, 2021. 208 p.
9. Peters, M. DevOps: The Complete Guide to Software Development Lifecycle. Packt Publishing, 2024. 182 p.
10. Shouhong Wang, Hai Wang. Information Systems Analysis and Design (2nd Edition). Systems Acquisition Approach. Universal-Publishers, 2022. 190 p.
11. Tilley, S. Systems Analysis and Design. 13th Edition: Cengage Learning, 2024. 576 p.
12. van Steen, M., Tanenbaum, A.S. Distributed Systems. 4th Edition: Distributed Systems, 2023. 683 p.
13. Xu, A., Lam, S. System Design Interview. An Insider's Guide: Volume 1 & 2. ByteByteGo, 2023. 434 p.
14. Андрущак І. Є., Кошелюк В. А. Особливості захисту хмарного середовища на основі BlockChain. Proceedings of the 10th International scientific and practical conference. CPN Publishing Group. Tokyo, Japan. 2022. pp. 161-168.
15. Андрущак І., Кошелюк В., Ясашний Д. Аудит безпеки оркестрації легких контейнерних платформ. Modern Trends in the Development of Economy, Technology and Industry: Collection of Scientific Papers «International Scientific

Unity» with Proceedings of the 3rd International Scientific and Practical Conference. April 9-11, 2025. Toronto, Canada. pp. 169-174.

16. Андрущак І., Кошелюк В., Ясашний Д. Про ефективність використання легких контейнерних платформ. The 15th International scientific and practical conference «Scientific research: integration of science and practice for effective development». Florence, Italy. International Science Group. 2025. pp. 47-52.

17. Гломозда Д.К. Проектування, системний аналіз і розробка корпоративних інформаційних систем: навч. посібник Нац. ун-т «Києво-Могилян. акад.». Київ: НаУКМА, 2023. 296 с.

18. Зінов'єва О.Г. Проектування інформаційних систем з використанням CASE-засобів. Українські студії в європейському контексті. 2023. №7. с. 220-227.

19. Ізмайлова О.В. Проектування інформаційних систем: навчальний посібник. Київ: КНУБА, 2022. 87 с.

20. Литвин В.В., Пасічник В.В., Шаховська Н.Б. Проектування інформаційних систем : навчальний посібник. Львів: Видавництво «Магнолія 2006», 2023. 380 с.

21. Марченко А.В. Проектування інформаційних систем: Посібник для студентів спеціальності 126 Інформаційні системи та технології. Київ: ДУІКТ, 2023. 166 с.

22. Трегуб В.Г. Проектування систем автоматизації. Київ : Видавництво «Ліра-К», 2023. 484 с.

ДЛЯ ПОДАТОК

Проектування інформаційних систем: методичні вказівки до самостійної роботи для здобувачів першого (бакалаврського) рівня вищої освіти освітньої програми «Комп’ютерні науки» галузі знань 12 (F) Інформаційні технології спеціальності 122 (F3) Комп’ютерні науки денної та заочної форм навчання / уклад. В.А. Кошелюк – Луцьк: ЛНТУ, 2026. 28 с.

Методичні вказівки для самостійної роботи спрямовано на ґрунтовну підготовку здобувачів освіти з курсу «Проектування інформаційних систем». У ньому визначено мету, завдання курсу, а також наведено перелік необхідних інформаційних джерел, які допоможуть ефективно опрацювати, винесених на самостійне вивчення, матеріал.

Призначене для здобувачів першого (бакалаврського) рівня вищої освіти спеціальності 122 (F3) Комп’ютерні науки.

Комп’ютерний набір та верстка: В.А. Кошелюк

Редактор: В.А. Кошелюк

Підп. до друку «_____» _____ 2026 р.
Формат 60×84/16. Папір офс. Гарн. Таймс.
Ум. друк. арк. _____ Тираж _____ прим. Зам. _____

Відділ іміджу та промоції
Луцького національного технічного університету
43018, м. Луцьк, вул. Львівська, 75