

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЗ „ЛУГАНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА”



Навчально-науковий інститут математики та інформаційних технологій
Кафедра математики та інформатики

О. О. Смагіна, В. Ю. Козуб, Г. О. Козуб

ПРОЄКТУВАННЯ ІНФОРМАЦІЙНИХ СИСТЕМ
методичні вказівки до лабораторних робіт
для здобувачів вищої освіти першого (бакалаврського) рівня
спеціальності F3 – «Комп'ютерні науки»

Лубни - 2026

УДК 004.272:37

Рекомендовано до друку вченою радою
ДЗ «Луганський національний університет імені Тараса Шевченка»,
(протокол № 7 від 30.01.2026 р.)

Укладачі:

Смагіна О. О. – кандидат педагогічних наук доцент.

Козуб В. Ю. – доктор філософії з комп'ютерних наук;

Козуб Г. О. – кандидат технічних наук, доцент.

Рецензенти:

Гоменюк С. І. – доктор технічних наук, професор, декан математичного факультету
Запорізького національного університету;

Могильний Г. А. – кандидат технічних наук, доцент, директор навчально-наукового
інституту математики та інформаційних технологій ДЗ «Луганський національний
університет імені Тараса Шевченка».

Смагіна О. О., Козуб В. Ю., Козуб Г. О. Проєктування інформаційних систем:
методичні вказівки до лабораторних для здобувачів вищої освіти першого бакалаврського
рівня спеціальності F3 – «Комп'ютерні науки». Лубни: ДЗ «ЛНУ імені Тараса Шевченка»,
2026. 32 с.

Методичні вказівки призначені для здобувачів вищої освіти першого
(бакалаврського) рівня, які навчаються за денною та заочною формами з метою розвитку та
закріплення практичних навичок під час підготовки та виконання лабораторних робіт.
Представлені основні відомості, практичні завдання та контрольні питання до
лабораторних робіт з освітнього компоненту «Проєктування інформаційних систем».

© Смагіна О.О., Козуб В.Ю., Козуб Г.О., 2026

© ДЗ «ЛНУ імені Тараса Шевченка», 2026

ПЕРЕДМОВА

Мета вивчення дисципліни: основною метою дисципліни «Проектування інформаційних систем» – це формування для здобувачів сучасних уявлень про методології та практики розробки складних програмних систем, для оволодіння практичними навичками ефективного використання інструментальних засобів моделювання та проектування в умовах вирішення реальних практичних завдань. Також до цілей освоєння дисципліни відносяться формування у здобувачів здібностей з реалізації оптимального підходу під час аналізу предметної області, вибору архітектурних рішень та проектування компонентів інформаційних систем.

В результаті освоєння курсу здобувач опанує: основи методологій проектування інформаційних систем та принципів структурного та об'єктно-орієнтованого аналізу; нотації моделювання бізнес-процесів та функціональності систем; підходи до проектування архітектури програмних систем; методи збору та специфікації вимог до інформаційних систем; технології проектування баз даних та інтерфейсів користувача; основні патерни проектування та інтеграції систем; принципи забезпечення безпеки інформаційних систем; стандарти документування програмних проєктів; сучасні інструментальні засоби автоматизованого проектування.

Вміти: проводити аналіз предметної області та моделювати бізнес-процеси організації; збирати та документувати вимоги до інформаційних систем; розробляти архітектурні рішення відповідно до специфікації вимог; створювати концептуальні, логічні та фізичні моделі баз даних, використання нотації UML та BPMN для моделювання систем; проектувати користувацькі інтерфейси з урахуванням принципів зручності; застосовувати гнучкі методології в процесі проектування; розробляти специфікації API та системи інтеграції; проектувати механізми забезпечення безпеки систем; створювати технічну документацію та тестову документацію.

Володіти: навичками роботи з професійними CASE-інструментами для моделювання систем; методиками аналізу та декомпозиції складних бізнес-процесів; техніками проведення інтерв'ю та збору вимог від стейкхолдерів; інструментами проєктування архітектури мікросервісних систем; засобами прототипування користувацьких інтерфейсів; практиками спринтового планування та управління беклогом; методами проєктування RESTful та GraphQL API; техніками моделювання загроз безпеки; інструментами автоматизованого тестування та безперервної інтеграції; навичками створення комплексної проєктної документації.

ЗМІСТ

ТЕМА 1. АНАЛІЗ ТА МОДЕЛЮВАННЯ БІЗНЕС-ПРОЦЕСІВ ОРГАНІЗАЦІЇ	6
ТЕМА 2. ПРОЄКТУВАННЯ АРХІТЕКТУРИ ІНФОРМАЦІЙНИХ СИСТЕМ	8
ТЕМА 3. ЗБІР ТА АНАЛІЗ ВИМОГ ДО ІНФОРМАЦІЙНОЇ СИСТЕМИ	10
ТЕМА 4. ПРОЄКТУВАННЯ БАЗ ДАНИХ ДЛЯ ІНФОРМАЦІЙНИХ СИСТЕМ	12
ТЕМА 5. МОДЕЛЮВАННЯ ФУНКЦІОНАЛЬНОСТІ СИСТЕМИ ЩО ВИКОРИСТОВУЮТЬ UML	14
ТЕМА 6. ПРОЄКТУВАННЯ КОРИСТУВАЦЬКОГО ІНТЕРФЕЙСУ	16
ТЕМА 7. ЗАСТОСУВАННЯ AGILE-МЕТОДОЛОГІЙ У ПРОЄКТУВАННІ ІС (інформаційних систем).....	18
ТЕМА 8. ПРОЄКТУВАННЯ СИСТЕМ ІНТЕГРАЦІЇ ТА АРІ ЧАСТИНА 1..	20
ТЕМА 8. ПРОЄКТУВАННЯ СИСТЕМ ІНТЕГРАЦІЇ ТА АРІ ЧАСТИНА 2..	22
ТЕМА 9. ПРОЄКТУВАННЯ БЕЗПЕКИ ІНФОРМАЦІЙНИХ СИСТЕМ ЧАСТИНА 1.....	24
ТЕМА 9. ПРОЄКТУВАННЯ БЕЗПЕКИ ІНФОРМАЦІЙНИХ СИСТЕМ ЧАСТИНА 2.....	26
ТЕМА 10. ДОКУМЕНТУВАННЯ ТА ТЕСТУВАННЯ ПРОЄКТІВ ІС (Інформаційних системах) ЧАСТИНА 1	28
ТЕМА 10. ДОКУМЕНТУВАННЯ ТА ТЕСТУВАННЯ ПРОЄКТІВ ІС (Інформаційних системах) ЧАСТИНА 2	30

ТЕМА 1. АНАЛІЗ ТА МОДЕЛЮВАННЯ БІЗНЕС-ПРОЦЕСІВ ОРГАНІЗАЦІЇ

Мета роботи: набути практичних навичок аналізу діяльності організації, виявлення основних бізнес-процесів, їх декомпозиції та моделювання з використанням нотації BPMN 2.0, освоїти методики створення моделей поточного стану процесів та проектування оптимізованих моделей, навчитися використовувати професійні інструменти для автоматизованого моделювання бізнес-процесів.

Практичне завдання

1. Необхідно провести аналіз та моделювання бізнес-процесу обробки замовлень в інтернет-магазині електронної техніки. Процес починається з моменту отримання замовлення від клієнта через веб-сайт й закінчується доставкою товару та отриманням оплати. Необхідно виявити всіх учасників процесу, їх ролі та відповідальність, визначити послідовність виконуваних операцій, точки прийняття рішень та можливі виключні ситуації.

2. Процес включає етап перевірки наявності товару на складі, де може виникнути ситуація відсутності необхідної кількості, що потребує взаємодії з постачальниками або пропозиції клієнту альтернативного товару. Якщо товар є в наявності, відбувається резервування товару та формування рахунку для оплати. Клієнт може обрати різні методи оплати, включаючи оплату карткою онлайн, оплату готівкою при отриманні або банківський переказ, що створює різні гілки виконання процесу.

3. Також треба створити детальну BPMN-модель, яка відображає всі перераховані аспекти процесу, використовуючи відповідні типи подій для початку процесу при надходженні замовлення та завершення після підтвердження доставки. Необхідно використати різні типи шлюзів для моделювання точок прийняття рішень, зокрема ексклюзивні шлюзи для вибору одного з варіантів та паралельні шлюзи для операцій, що можуть виконуватися одночасно.

4. Обидві моделі повинні бути створені з дотриманням стандартів нотації BPMN 2.0 та супроводжуватися текстовим описом процесу, переліком учасників з описом їх ролей, аналізом проблемних місць у поточній моделі та обґрунтуванням запропонованих покращень у майбутній моделі

Контрольні питання

1. Опишіть розуміння основних елементів нотації BPMN 2.0 та їх призначення, зокрема необхідно пояснити різницю між різними типами шлюзів, навести приклади ситуацій, коли доцільно використовувати ексклюзивний, паралельний або інклюзивний шлюз, а також описати роль подій у моделюванні бізнес-процесів.

2. Поясніть методології аналізу бізнес-процесів, де необхідно розкрити сутність понять модель поточного стану та модель майбутнього стану процесу, пояснити для чого потрібне створення обох моделей, описати методи виявлення неефективних операцій та проблемних місць у процесах, а також обґрунтувати важливість залучення різних стейкхолдерів до процесу моделювання.

3. Потрібно описати як на практичному застосуванні моделюванні бізнес-процесів, які були використані інструментальні засоби використовуються для автоматизованого моделювання, назвати переваги використання стандартизованих нотацій перед довільним графічним представленням процесів, пояснити як результати моделювання можуть бути використані для подальшого проєктування інформаційної системи та автоматизації процесів.

Література

1. Khlorchuk V. Моделювання бізнес-процесів компанії з розробки програмного забезпечення. 2022. URL: <https://is.gd/daMJAB>
2. Kononenko Zhanna, Sharavara Roman, Yakovenko Tetyana. Economy and Society. 2024. No. 4007. URL: <https://economyandsociety.in.ua/index.php/journal/article/view/4007/3937>

ТЕМА 2. ПРОЄКТУВАННЯ АРХІТЕКТУРИ ІНФОРМАЦІЙНИХ СИСТЕМ

Мета роботи: набути практичних навичок розробки архітектури інформаційних систем, освоїти принципи вибору архітектурного стилю відповідно до вимог проєкту, навчитися створювати діаграми компонентів та розгортання з використанням UML, опанувати методики проєктування мікросервісної архітектури та RESTful API, розвинути розуміння компромісів між різними архітектурними рішеннями.

Практичне завдання

1. Необхідно спроектувати архітектуру корпоративної системи управління проєктами, яка повинна підтримувати роботу розподілених команд, забезпечувати планування задач та спринтів, відстеження часу виконання робіт, управління документацією, інтеграцію з системами контролю версій та засобами комунікації. Система повинна масштабуватися для підтримки від декількох десятків до сотень користувачів та забезпечувати високу доступність критичних функцій.

2. Для кожного виділеного мікросервісу необхідно визначити його відповідальність та бізнес-можливості, які він реалізує, специфікувати надавані ним API endpoints з описом методів та форматів даних, визначити залежності від інших сервісів та зовнішніх систем, описати модель даних та вимоги до сховища даних, оцінити очікуване навантаження та вимоги до масштабування. Особливу увагу слід приділити проєктуванню API Gateway як одна точка входу для клієнта та клієнтських додатків, яка здійснює маршрутизацію запитів до відповідних мікросервісів, автентифікацію та авторизацію, агрегацію даних з декількох сервісів, обмеження швидкості та надсилання запитів відповідно до додатку.

3. Необхідно розробити діаграму компонентів UML, яка показує основні мікросервіси системи як компоненти, їх надані та вимагані інтерфейси, залежності між компонентами через використання інтерфейсів. Діаграма

повинна відображати як мікросервіси взаємодіють один з одним, які зовнішні системи та сервіси використовуються, де розташовуються точки інтеграції.

Контрольні запитання

1. Зробіть порівняльний аналіз різних архітектурних стилів, необхідно детально пояснити переваги та недоліки монолітної, клієнт-серверної та мікросервісної архітектур, навести критерії вибору того чи іншого підходу залежно від характеристик проєкту, описати типові проблеми та виклики, які виникають при використанні кожного архітектурного стилю, а також пояснити в яких випадках міграція з монолітної архітектури до мікросервісної є виправданою.

2. Потрібно описати основні принципи архітектурного стилю REST та їх практичне значення, пояснити як правильно використовувати HTTP-методи для операцій над ресурсами, навести приклади проєктування URI для ресурсів з урахуванням ієрархії та відносин, описати механізми версіонування API для забезпечення зворотної сумісності, розглянути питання обробки помилок та повернення відповідних HTTP статус-кодів.

3. Опишіть практичні аспекти побудови мікросервісної архітектури, необхідно описати патерни організації взаємодії між мікросервісами включаючи синхронну комунікацію через REST API та асинхронну через черги повідомлень, пояснити роль API Gateway в архітектурі та його основні функції, розглянути проблему розподілених транзакцій та підходи до забезпечення консистентності даних через паттерн Saga, описати стратегії розгортання та масштабування мікросервісів з використанням контейнерних технологій.

Література

1. Reza Jakaria Rahmanto. Design Of Strategic Information System Blueprint With Enterprise Architecture Planning Method. 2023. URL: <https://is.gd/BT3hrU>
2. N Zulfarian, I D Rosiyadi. Designing Enterprise Architecture for Academics Information System Platform using the Open Group Architecture Framework Architecture Development Method. 2020. URL: <https://is.gd/DL5Fkg>

ТЕМА 3. ЗБІР ТА АНАЛІЗ ВИМОГ ДО ІНФОРМАЦІЙНОЇ СИСТЕМИ

Мета роботи: набуття навичок роботи зі збором та аналізом вимог до інформаційних систем, освоїти методи виявлення потреб стейкхолдерів, навчитися класифікувати та документувати функціональні та нефункціональні вимоги, опанувати техніки створення специфікації вимог відповідно до міжнародних стандартів, розвинути вміння формалізувати користувацькі історії та сценарії використання системи.

Практичне завдання

1. Необхідно виконати збір та аналіз вимог для проєктування системи електронного документообігу навчального закладу. Система призначена для автоматизації процесів створення, погодження, підписання та зберігання документів, забезпечення контролю виконання документів, організації архіву та пошуку документів. Користувачами системи будуть адміністрація навчального закладу, співробітники різних підрозділів, керівники підрозділів, діловоди, архіваріуси.

2. Необхідно визначити всі групи стейкхолдерів та описати їх характеристики, потреби та очікування від системи. Для адміністрації важливі можливості моніторингу стану документів, аналітика по типах та обсягам документообігу, контроль термінів виконання. Співробітники потребують зручного інтерфейсу для створення та редагування документів, можливості прикріплення файлів, простого направлення документів на погодження. Керівники підрозділів мають швидко переглядати документи, що потребують їх погодження, додавати коментарі, приймати рішення про погодження або відхилення.

3. Вимоги до пошуку та фільтрації документів повинні передбачати пошук за реквізитами документа такими як номер, дата, тип документа, автор, відповідальний виконавець, збереження та частотою використовуваних запитів.

Контрольні запитання

1. Описати методи збору вимог та їх ефективного застосування, необхідно описати основні техніки виявлення вимог включаючи інтерв'ю, анкетування, спостереження, аналіз документів, пояснити в яких ситуаціях доцільно використовувати кожний метод, розглянути переваги та обмеження різних підходів, навести рекомендації щодо планування та проведення інтерв'ю з стейкхолдерами для максимальної ефективності збору інформації.

2. Поясніть різницю між функціональними та нефункціональними вимогами з наведенням конкретних прикладів, описати характеристики якісно сформульованої вимоги такі як однозначність, повнота, несуперечливість, можливість перевірки, трасованість, розглянути структуру документа специфікації вимог та призначення його основних розділів, пояснити важливість присвоєння унікальних ідентифікаторів вимогам.

3. Поясніть формат написання користувацьких історій та їх переваги в гнучких методологіях розробки, описати що таке критерії прийняття та як вони допомагають у валідації реалізації вимог, розглянути структуру детального сценарію використання з основним потоком подій, альтернативними потоками та виключними ситуаціями, пояснити як користувацькі історії та сценарії використання співвідносяться з формальними вимогами в специфікації.

Література

1. Zowghi D., Coulin Ch. Requirements Elicitation: A Survey of Technique, Approaches and Tools. 2005. URL: <https://is.gd/6wwHs2>
2. Mariyam F., Mehfuz Sh., Asim M., Ahuja R., Sadiq Mohd. A rough-set based goal-oriented methodology for requirements analysis in information system design. 2025. URL: <https://is.gd/b5fY6U>

ТЕМА 4. ПРОЄКТУВАННЯ БАЗ ДАНИХ ДЛЯ ІНФОРМАЦІЙНИХ СИСТЕМ

Мета роботи: набути практичних навичок проєктування реляційних баз даних для інформаційних систем, освоїти методи концептуального, логічного та фізичного моделювання даних, навчитися створювати ER-діаграми та нормалізувати структури баз даних, опанувати техніки оптимізації схем баз даних для забезпечення продуктивності, розвинути розуміння вибору між реляційними та NoSQL базами даних відповідно до вимог системи.

Практичне завдання

1. Необхідно розробити базу даних для системи управління медичним обслуговуванням для медичної клініки, яка повинна підтримувати облік пацієнтів з їх медичними картками та історією хвороб, управління лікарями різних спеціалізацій з їх графіками роботи.

2. На етапі концептуального моделювання необхідно виявити основні сутності предметної області. Сутність Пацієнт представляє людей які отримують медичні послуги в клініці з атрибутами унікальний номер медичної картки як ідентифікатор пацієнта, прізвище ім'я по-батькові, дата народження для визначення віку.

3. На етапі логічного проєктування необхідно перетворити ER-діаграму в набір таблиць реляційної бази даних.

Контрольні питання

1. Детально описати що створюється на етапі концептуального проєктування та яка його мета, пояснити різницю між концептуальною та логічною моделями даних з точки зору абстракції та залежності від технологій, розкрити завдання фізичного проєктування та які конкретні рішення приймаються на цьому етапі включаючи типи даних індекси. Обґрунтувати важливість послідовного проходження всіх етапів проєктування для забезпечення якості бази даних та уникнення дорогих виправлень на пізніх стадіях.

2. Пояснити що таке надмірність даних та які конкретні проблеми вона створює через аномалії вставки видалення оновлення з прикладами, описати вимоги першої другої та третьої нормальних форм з детальним поясненням атомарності часткових залежностей транзитивних залежностей.

3. Описати всі типи обмежень цілісності в реляційних базах даних включаючи PRIMARY KEY; FOREIGN KEY; UNIQUE; NOT NULL CHECK, DEFAULT та їх призначення для забезпечення коректності даних, пояснити роль індексів у продуктивності баз даних включаючи як індекси прискорюють пошук та як вони впливають на вставку оновлення.

Література

1. Kellou-Menouer K., Kedad Z. SchemaDecrypt++: Parallel on-line Versioned Schema Inference for Large Semantic Web Data sources. 2020. URL: <https://is.gd/7ZiJqF>
2. Cecez-Kecmanovic D. The discipline of information systems: Issues and challenges. 2002. URL: <https://is.gd/uJpBec>

ТЕМА 5. МОДЕЛЮВАННЯ ФУНКЦІОНАЛЬНОСТІ СИСТЕМИ ЩО ВИКОРИСТОВУЮТЬ UML

Мета роботи: набуття практичних навичок моделювання функціональності інформаційних систем з використанням уніфікованої мови моделювання UML, освоїти створення різних типів діаграм для представлення статичної структури та динамічної поведінки системи, навчитися визначати варіанти використання та деталізувати взаємодію об'єктів, опанувати методики об'єктно-орієнтованого аналізу та проєктування, розвинути розуміння вибору відповідних типів діаграм для моделювання різних аспектів системи.

Практичне завдання

1. Необхідно виконати комплексне моделювання інформаційної системи управління готелем з використанням різних типів UML-діаграм.
2. Система повинна підтримувати реєстрацію гостей та управління їх профілями, пошук та бронювання номерів з різними типами та цінами, онлайн бронювання через веб-сайт та бронювання через адміністратора, check-in та check-out гостей
3. Необхідно побудувати діаграму класів яка показує основні класи системи та їх відношення.

Контрольні питання

1. Пояснити фундаментальну різницю між структурними діаграмами які моделюють статичну архітектуру та поведінковими діаграмами які моделюють динаміку з наведенням прикладів кожного типу.
2. Описати коли використовується діаграма варіантів використання на якому етапі життєвого циклу та яку інформацію вона передає стейкхолдерам, розкрити призначення діаграми класів як центральної діаграми об'єктно-орієнтованого проєктування та які елементи вона містить класи атрибути операції відношення.
3. Пояснити різницю між діаграмою послідовності та діаграмою комунікації для моделювання взаємодії об'єктів з точки зору акценту на хронології versus

структурі, описати коли доцільно використовувати діаграму станів для яких типів об'єктів вона найбільш корисна.

Література

1. Makoiedova V. Проєктування інформаційної системи приймальної комісії на основі UML-діаграм. 2023. URL: <https://is.gd/mt5pJ8>
2. Broy M., Cengarle M. V., Rumpe B. Towards a System Model for UML. The Structural Data Model. 2006. URL: <https://is.gd/d5ye0t>

ТЕМА 6. ПРОЄКТУВАННЯ КОРИСТУВАЦЬКОГО ІНТЕРФЕЙСУ

Мета роботи: набути практичних навичок проєктування користувацьких інтерфейсів інформаційних систем з урахуванням принципів зручності та доступності, освоїти методики створення wireframes та інтерактивних прототипів, навчитися застосовувати принципи застосовування UX/UI дизайну для покращення взаємодії з користувачами. Опанувати та навчитись інструментальними засобами прототипування інтерфейсів, розвинути розуміння методів тестування юзабіліті та збору зворотного зв'язку від користувачів.

Практичне завдання

1. Необхідно спроектувати користувацький інтерфейс для мобільного додатку системи управління особистими фінансами. Додаток призначений для відстеження доходів та витрат, складання бюджету, аналізу фінансових звичок, досягнення фінансових цілей. Цільова аудиторія включає людей віком від вісімнадцяти до сорока п'яти років з різним рівнем фінансової грамотності, які хочуть краще контролювати свої фінанси та заощаджувати на важливі цілі.

2. Спочатку необхідно провести дослідження користувачів та створити щонайменше дві user personas, які представляють різні сегменти цільової аудиторії. Перша персона може бути молодим професіоналом віком двадцять п'ять років, який нещодавно почав працювати, має нерегулярні доходи від фрілансу, часто витрачає імпульсивно та хоче навчитися заощаджувати на подорожі. Його потреби включають простий спосіб швидко записувати витрати на ходу, розуміння куди йдуть гроші через візуальну аналітику, мотивацію дотримуватися бюджету через геміфікацію. Його болі включають відсутність часу на детальне планування, забування записувати дрібні витрати, відсутність дисципліни в дотриманні бюджету.

3. Друга персона може бути сімейною людиною тридцяти восьми років з двома дітьми, які мають стабільний дохід від зарплати обох партнерів, планують великі покупки та освіту дітей, хочуть оптимізувати сімейний

бюджет. Її потреби включають можливість спільного управління бюджетом з партнером, планування довгострокових фінансових цілей, відстеження витрат по категоріях та членах сім'ї, автоматизацію рутинних операцій обліку. Її болі включають складність координації фінансів між партнерами, забування врахувати всі витрати сім'ї, брак часу на фінансовий аналіз.

Контрольні питання

1. Пояснити різницю між UX та UI дизайном та як вони співвідносяться, описати основні принципи проєктування інтерфейсів такі як консистентність, зворотний зв'язок, запобігання помилкам, гнучкість, та мінімалізм.

2. Обґрунтуйте процес проєктування інтерфейсів та дослідження користувачів, що таке user persona та як вона допомагає в проєктуванні

3. Поясніть що таке usability testing та які проблеми воно виявляє, описати як планувати та проводити сесії тестування юзабіліті включаючи підготовку задач та рекрутинг учасників.

Література

1. Galindo-Losada J., Ayala-Tipan C., Santórum M., Carrión-Toro M., Acosta-Vargas P. WYDISWYG: A Method to Design User Interfaces Combining Design Principles and Quality Factors. 2023. URL: <https://is.gd/LGXr8A>

2. Faudzi M. A., Che Cob Z., Omar R., Sharudin Sh. A., Ghazali M. Investigating the User Interface Design Frameworks of Current Mobile Learning Applications: A Systematic Review. 2023. URL: <https://is.gd/5RAgFR>

ТЕМА 7. ЗАСТОСУВАННЯ AGILE-МЕТОДОЛОГІЙ У ПРОЄКТУВАННІ ІС (ІНФОРМАЦІЙНИХ СИСТЕМ)

Мета роботи: набути практичних навичок застосування гнучких методологій розробки в процесі проєктування інформаційних систем, освоїти фреймворк Scrum та його артефакти, навчитися планувати спринти та управляти беклогом продукту, опанувати методики декомпозиції вимог на користувацькі історії, розвинути розуміння ролей в Agile команді та їх взаємодії, освоїти інструменти для управління Agile проєктами.

Практичне завдання

1. Необхідно спланувати та симулювати виконання спринту розробки для проєкту створення CRM системи для малого бізнесу. Система призначена для управління контактами клієнтів, відстеження комунікацій, управління можливостями продажу, автоматизації маркетингу. Необхідно виконати роботу всіх ролей Scrum команди та пройти через всі події спринту, створити відповідні артефакти, використати інструмент управління проєктом.

2. Спочатку в ролі Product Owner треба створити початковий Product Backlog з користувацьких історій для CRM системи. Історії повинні охоплювати різні функціональні області включаючи управління контактами з можливістю додавати нові контакти з детальною інформацією, імпортувати контакти з CSV файлу, шукати та фільтрувати контакти за різними критеріями, сегментувати контакти на групи, переглядати історію взаємодій з контактом.

3. Область комунікацій включає логування телефонних дзвінків з контактом, запис зустрічей та їх результатів, перегляд хронології всіх взаємодій. Область аналітики та звітності включає перегляд воронки продажів з конверсією між стадіями, звіт по продуктивності менеджерів, прогноз продажів на період, динаміку додавання нових контактів. Кожна користувацька історія має бути написана за форматом як Роль: «я хочу Функціональність щоб Цінність та доповнена критеріями прийняття».

Контрольні питання

1. Розкажіть про ключові принципи Agile та їх практичне застосування.

2. Фреймворк Scrum та його основні компоненти. Потрібно детально описати три ролі в Scrum команді та їх відповідальність, пояснити призначення та структуру всіх Scrum подій включаючи Sprint Planning, Daily Scrum, Sprint Review.

3. Пояснити як створюються та оцінюються користувацькі історії з використанням story points, описати техніки оцінювання складності такі як Planning Poker, T-shirt sizing, розглянути поняття velocity команди та як воно використовується для планування, пояснити різницю між Scrum та Kanban підходами до управління роботою.

Література

1. Ndou V., Ingrosso A., Di Girolamo A. Framework for Agile Transformation: Guiding Organizations Through Cultural, Structural, and Competency Shifts in Project Management. 2024. URL: <https://is.gd/gtkO1Y>
2. Guerrero-Ulloa G., Rodríguez-Domínguez C., Hornos M. J. Agile Methodologies Applied to the Development of Internet of Things (IoT)-Based Systems: A Review. 2023. URL: <https://is.gd/Xwyk2j>

ТЕМА 8. ПРОЄКТУВАННЯ СИСТЕМ ІНТЕГРАЦІЇ ТА API ЧАСТИНА 1

Мета роботи: набути практичних навичок проєктування систем інтеграції між різними інформаційними системами, освоїти принципи побудови RESTful та GraphQL API, навчитися специфікувати інтерфейси взаємодії між компонентами, опанувати патерни інтеграції корпоративних додатків, розвинути розуміння методів забезпечення надійності та безпеки інтеграційних рішень, освоїти інструменти документування та тестування API.

Практичне завдання

1. Необхідно спроектувати систему інтеграції для платформи електронної комерції, яка включає декілька мікросервісів: сервіс каталогу продуктів, сервіс управління замовленнями, сервіс управління інвентарем, платіжний сервіс, сервіс доставки, сервіс сповіщень. Необхідно спроектувати RESTful API для синхронної взаємодії між клієнтом та сервісами, асинхронну інтеграцію через події для координації між сервісами, API Gateway для зовнішніх клієнтів, реалізацію Saga патерну для процесу оформлення замовлення.

2. Спроектувати RESTful API для сервісу управління замовленнями, який надає endpoints для створення нового замовлення через POST до orders з тілом запиту що містить ідентифікатор клієнта, список товарів з кількістю, адресу доставки, метод оплати, відповідь містить створене замовлення з унікальним ідентифікатором та статусом, статус код двісті один Created та Location header з URI створеного замовлення. Отримання замовлення через GET до orders з ідентифікатором повертає детальну інформацію про замовлення включаючи товари з цінами на момент замовлення, загальну суму, статус обробки, інформацію про оплату та доставку.

3. Отримання списку замовлень клієнта через GET до orders з query параметрами для фільтрації за статусом, датою, пагінації через параметри page та size, сортування через параметр sort, відповідь містить масив замовлень та метадані пагінації такі як загальна кількість, кількість сторінок, поточна

сторінка. Оновлення статусу замовлення через PATCH до orders з ідентифікатором з тілом що містить новий статус, виконується валідація допустимих переходів статусів, публікується подія про зміну статусу для інших сервісів. Скасування замовлення через DELETE до orders з ідентифікатором або POST до orders з ідентифікатором cancel, перевіряється можливість скасування залежно від поточного статусу, ініціюється Saga для відміни резервувань та платежу.

Контрольні питання

1. Опишіть основні патерни інтеграції такі як: передача файлів (file transfer), спільна база даних (shared database), віддалений виклик процедур (remote procedure invocation), обмін повідомленнями (messaging) поясненням їх переваг та недоліків, пояснити коли доцільно використовувати синхронну інтеграцію а коли асинхронну, розглянути роль Message Broker в інтеграційній архітектурі та які можливості він надає, описати топології обміну повідомленнями point to point та publish subscribe.

2. Потрібно детально описати обмеження архітектурного стилю REST та їх практичне значення, пояснити як правильно проєктувати URI для ресурсів з урахуванням ієрархії та відносин, розглянути використання HTTP методів для операцій над ресурсами з поясненням ідемпотентності, описати застосування HTTP статус кодів для індикації результатів обробки запитів, пояснити підходи до версіонування API для забезпечення зворотної сумісності.

Література

1. Pramudaya T., Agustina F. Analysis and Design of Integration Model for API Management and CI/CD at Directorate General of Taxation. 2024. URL: <https://is.gd/FM3INN>

2. Dávila Nicanor L., Banda Madrid A., Martínez Hernández J. E., Aguilar Juárez I. Performance of API Design for Interoperability of Medical Information Systems. 2024. URL: <https://is.gd/m7YvZC>

ТЕМА 8. ПРОЄКТУВАННЯ СИСТЕМ ІНТЕГРАЦІЇ ТА API ЧАСТИНА 2

Мета роботи: набути практичних навичок проєктування систем інтеграції між різними інформаційними системами, освоїти принципи побудови RESTful та GraphQL API, навчитися специфікувати інтерфейси взаємодії між компонентами, опанувати патерни інтеграції корпоративних додатків, розвинути розуміння методів забезпечення надійності та безпеки інтеграційних рішень, освоїти інструменти документування та тестування API.

Практичне завдання

1. Для кожного endpoint треба задокументувати специфікацію включаючи HTTP метод та URI шаблон, параметри шляху, query параметри, заголовки запиту, тіло запиту з JSON схемою та прикладом, можливі статус коди відповіді з описом, тіло відповіді з JSON схемою та прикладом, опис помилок що можуть виникнути. Необхідно спроектувати обробку помилок з консистентною структурою відповідей про помилки що містять код помилки, повідомлення зрозуміле людині, деталі помилки такі як які поля не пройшли валідацію, trace ID для відстеження в логах.

2. Необхідно визначити структуру повідомлень для кожної події включаючи метадані такі як ідентифікатор події, тип події, timestamp, ідентифікатор кореляції для відстеження розподіленої транзакції, версія схеми події, та payload з бізнес даними. Треба обрати формат серіалізації повідомлень такий як JSON для читабельності та сумісності або Avro для ефективності та еволюції схеми, визначити топіки або черги для кожного типу подій, встановити гарантії доставки at least once з ідемпотентною обробкою повідомлень в consumers.

3. Для API Gateway спроектувати зовнішній API який клієнтські додатки використовують, який відрізняється від внутрішніх API сервісів. Gateway агрегує дані з декількох сервісів для зменшення кількості запитів від клієнта, наприклад endpoint GET до api orders з ідентифікатором details може агрегувати інформацію про замовлення з сервісу замовлень, деталі товарів з

сервісу каталогу, статус доставки з сервісу доставки в один об'єднаний response. Gateway виконує автентифікацію перевіряючи JWT токен в Authorization header, авторизацію перевіряючи права доступу до ресурсів, rate limiting обмежуючи кількість запитів від клієнта за часовий період.

Контрольні питання

1. Поясніть які проблеми REST вирішує GraphQL, описати структуру GraphQL схеми з типами, queries, mutations, subscriptions, розглянути як клієнт формує запити в GraphQL з можливістю точно специфікувати потрібні поля, пояснити що таке резолвери та як вони отримують дані, порівняти переваги та недоліки GraphQL та REST для різних типів додатків.

2. Описати переваги асинхронної комунікації через події порівняно з синхронними викликами, пояснити роль Message Broker та гарантії доставки повідомлень, розглянути Saga патерн для управління розподіленими транзакціями з поясненням orchestration та choreography підходів, описати як реалізуються компенсуючі транзакції для відміни змін при невдачі, навести приклади коли доцільно використовувати Saga патерн.

3. Опишіть функції API Gateway в мікросервісній архітектурі включаючи маршрутизацію, агрегацію, автентифікацію, rate limiting, пояснити переваги використання API Gateway та потенційні проблеми, розглянути OpenAPI Specification як стандарт документування RESTful API та його структуру, описати інструменти для роботи з OpenAPI такі як Swagger Editor, Swagger UI, Postman, пояснити важливість автоматизованого тестування API та які типи тестів необхідні.

Література

1. Pramudaya T., Agustina F. Analysis and Design of Integration Model for API Management and CI/CD at Directorate General of Taxation. 2024. URL: <https://is.gd/FM3INN>

2. Dávila Nicanor L., Banda Madrid A., Martínez Hernández J. E., Aguilar Juárez I. Performance of API Design for Interoperability of Medical Information Systems. 2024. URL: <https://is.gd/m7YvZC>

ТЕМА 9. ПРОЄКТУВАННЯ БЕЗПЕКИ ІНФОРМАЦІЙНИХ СИСТЕМ

ЧАСТИНА 1

Мета роботи: набути практичних навичок проєктування механізмів безпеки інформаційних систем, освоїти методики моделювання загроз та аналізу ризиків, навчитися проєктувати системи автентифікації та авторизації, опанувати техніки захисту даних при зберіганні та передачі, розвинути розуміння стандартів та best practices інформаційної безпеки, освоїти інструменти для аналізу вразливостей та тестування безпеки.

Практичне завдання

1. Необхідно спроектувати комплексну систему безпеки для веб платформи онлайн освіти, яка включає користувачів трьох типів студенти, викладачі, адміністратори, курси з навчальними матеріалами, тести та завдання, персональні дані користувачів та результати навчання, форуми для комунікації. Необхідно виконати моделювання загроз, спроектувати систему автентифікації та авторизації, визначити механізми захисту даних, розробити політики безпеки, забезпечити відповідність GDPR.

2. Створити архітектурну діаграму системи з компонентами включаючи веб додаток frontend, API gateway, на діаграмі позначити потоки даних між компонентами, границі довіри де дані переходять між різними зонами безпеки, точки входу де зовнішні актори взаємодіють з системою.

Контрольні питання

1. Необхідно пояснити тріаду CIA та додаткові властивості безпеки систем, описати методологію STRIDE для категоризації загроз з прикладами кожного типу, розглянути процес моделювання загроз включаючи ідентифікацію, оцінку ризиків, визначення контрзаходів, пояснити чому моделювання загроз повинно виконуватися на ранніх етапах проєктування системи.

2. Описати три фактори автентифікації та їх приклади, пояснити переваги багатофакторної автентифікації, розглянути вразливості парольної автентифікації та best practices, описати токен базовану автентифікацію з

використанням JWT включаючи структуру токена та процес валідації, пояснити різницю між stateless та stateful токенами.

Література

1. Lu X., Dong R., Wang Q., Zhang L. Information Security Architecture Design for Cyber-Physical Integration System of Air Traffic Management. 2023. URL: <https://is.gd/pr7pGb>
2. Carter B., Adams S., Bakirtzis G., Sherburne T., Beling P., Horowitz B., Fleming C. A Preliminary Design-Phase Security Methodology for Cyber–Physical Systems. 2019. URL: <https://is.gd/F9elhc>

ТЕМА 9. ПРОЄКТУВАННЯ БЕЗПЕКИ ІНФОРМАЦІЙНИХ СИСТЕМ ЧАСТИНА 2

Мета роботи: набути практичних навичок проєктування механізмів безпеки інформаційних систем, освоїти методики моделювання загроз та аналізу ризиків, навчитися проєктувати системи автентифікації та авторизації, опанувати техніки захисту даних при зберіганні та передачі, розвинути розуміння стандартів та best practices інформаційної безпеки, освоїти інструменти для аналізу вразливостей та тестування безпеки.

Практичне завдання

1. Для кожного компоненту та потоку даних виконати аналіз загроз STRIDE. Для компоненту веб додаток ідентифікувати Spoofing загрози такі як фішингові сайти що імітують платформу.

2. Для кожної ідентифікованої загрози оцінити ризик за шкалою DREAD включаючи Damage потенційна шкода, Reproducibility легкість відтворення атаки, Exploitability необхідні зусилля для експлуатації, Affected Users кількість постраждалих, Discoverability легкість виявлення вразливості. Пріоритизувати загрози з найвищим ризиком для розробки контрзаходів. Для топ п'яти загроз визначити конкретні контрзаходи з описом як вони мітигують загрозу.

3. Скласти документ Security Design Review який містить опис архітектури системи з точки зору безпеки, результати моделювання загроз з ідентифікованими ризиками, спроектовані контрзаходи для мітигації загроз.

Контрольні питання

1. Описати різні моделі контролю доступу DAC, MAC, RBAC, ABAC з їх перевагами та обмеженнями, детально розглянути RBAC включаючи визначення ролей, призначення дозволів, принцип найменших привілеїв, навести приклади коли ABAC надає переваги над RBAC через гнучкість політик, пояснити як імплементувати перевірку дозволів в розподіленій системі з мікросервісами.

2. Пояснити різницю між симетричним та асиметричним шифруванням з прикладами алгоритмів, описати як працює TLS протокол для захисту мережевої комунікації включаючи handshake процес, розглянути best practices для захисту даних при зберіганні через encryption at rest, key management, розглянути вимоги GDPR до захисту персональних даних та принципи Privacy by Design, описати що таке data breach та які дії необхідні при його виявленні.

3. Описати які логи необхідно збирати для security monitoring та incident response, пояснити як виявляти аномальні активності та потенційні атаки через аналіз логів, розглянути різницю між vulnerability assessment та penetration testing, описати що таке OWASP Top Ten як список найпоширеніших вразливостей веб додатків, навести приклади контрзаходів для захисту від SQL injection, XSS, CSRF атак, пояснити важливість security awareness training для користувачів та розробників.

Література

1. Lu X., Dong R., Wang Q., Zhang L. Information Security Architecture Design for Cyber-Physical Integration System of Air Traffic Management. 2023. URL: <https://is.gd/pr7pGb>

2. Carter B., Adams S., Bakirtzis G., Sherburne T., Beling P., Horowitz B., Fleming C. A Preliminary Design-Phase Security Methodology for Cyber–Physical Systems. 2019. URL: <https://is.gd/F9elhc>

ТЕМА 10. ДОКУМЕНТУВАННЯ ТА ТЕСТУВАННЯ ПРОЄКТІВ ІС (ІНФОРМАЦІЙНИХ СИСТЕМАХ) ЧАСТИНА 1

Мета роботи: Оволодіти методологією створення технічної документації інформаційних систем, засвоїти принципи побудови стратегій тестування програмного забезпечення, набути практичних навичок розробки автоматизованих тестів різних рівнів та налаштування конвеєрів безперервної інтеграції для забезпечення якості розроблюваних систем протягом усього життєвого циклу.

Практичне завдання

1. Розробити комплексну документацію та план тестування для інформаційної системи управління бібліотечним фондом університету. Система забезпечує автоматизацію процесів каталогізації літератури, обліку книговидачі студентам та викладачам, управління резервуваннями видань, контролю термінів повернення з нарахуванням штрафів за прострочення, формування статистичних звітів про використання фонду.

2. Створити специфікацію вимог до програмного забезпечення, що включає функціональні вимоги для ролей бібліотекаря, читача та адміністратора системи. Деталізувати процеси реєстрації користувачів з верифікацією належності до університетської спільноти, пошуку видань за різними критеріями.

3. Розробити документ архітектури системи з описом багаторівневої структури застосунку, що відокремлює презентаційний шар веб-інтерфейсу від бізнес-логіки та шару доступу до даних. Обґрунтувати вибір реляційної бази даних для зберігання структурованої інформації про книжковий фонд та облікових записів користувачів.

Контрольні запитання

1. Обґрунтуйте роль технічної документації у забезпеченні підтримуваності складних інформаційних систем протягом багаторічного періоду експлуатації. Проаналізуйте наслідки відсутності актуальної

документації для процесів модифікації функціональності системи та усунення дефектів у виробничому середовищі.

2. Поясніть взаємозв'язок між якістю специфікації вимог до програмного забезпечення та ефективністю процесу тестування розробленої системи. Визначте критерії повноти опису вимоги, що дозволяють створити верифікований тестовий випадок для перевірки факту реалізації цієї вимоги.

Література

1. Milosz M., Plechawska-Wójcik M., Dzieńkowski M. Testing the Quality of the Mobile Application Interface Using Various Methods—A Case Study of the T1DCoach Application. 2024. URL: <https://is.gd/3cU6uZ>

2. Otkhozoria N., Petriashvili L., Zhvania T., Imerlishvili A. Advancing information system testing: challenges, methods, and practical recommendations. 2025. URL: <https://is.gd/OC5bqW>

ТЕМА 10. ДОКУМЕНТУВАННЯ ТА ТЕСТУВАННЯ ПРОЄКТІВ ІС (ІНФОРМАЦІЙНИХ СИСТЕМАХ) ЧАСТИНА 2

Мета роботи: Оволодіти методологією створення технічної документації інформаційних систем, засвоїти принципи побудови стратегій тестування програмного забезпечення, набути практичних навичок розробки автоматизованих тестів різних рівнів та налаштування конвеєрів безперервної інтеграції для забезпечення якості розроблюваних систем протягом усього життєвого циклу.

Практичне завдання

1. Розробити стратегію автоматизації тестування для інформаційної системи управління бібліотечним фондом з визначенням пріоритетів покриття функціональності автоматизованими перевітками. Створити набір модульних тестів для критичних алгоритмів бізнес-логіки з використанням фреймворку тестування для обраної мови програмування.

2. Спроектувати інтеграційні тести для перевірки взаємодії між REST API системи та реляційною базою даних.

3. Створити тестові сценарії для ендпоінтів оформлення видачі книги читачеві, що верифікують коректність створення запису про видачу в базі даних.

Контрольні запитання

1. Обґрунтуйте економічну доцільність інвестицій у розробку автоматизованих тестів для довгострокових проєктів розробки програмного забезпечення. Оцініть фактори, що впливають на період окупності витрат на автоматизацію тестування через економію ресурсів на регресійному тестуванні при внесенні змін до системи.

2. Поясніть вплив практик безперервної інтеграції та безперервного розгортання на швидкість доставки нової функціональності кінцевим користувачам та якість програмного забезпечення у виробничому середовищі. Оцініть ризики автоматизованого розгортання змін без ручного затвердження

та механізми їхньої мітигації через стратегії поступового впровадження нових версій.

Література

1. Milosz M., Plechawska-Wójcik M., Dzieńkowski M. Testing the Quality of the Mobile Application Interface Using Various Methods—A Case Study of the T1DCoach Application. 2024. URL: <https://is.gd/3cU6uZ>
2. Otkhozoria N., Petriashvili L., Zhvania T., Imerlishvili A. Advancing information system testing: challenges, methods, and practical recommendations. 2025. URL: <https://is.gd/OC5bqW>

Навчальне видання

**Ольга Олександрівна Смагіна
Владислав Юрійович Козуб
Галина Олександрівна Козуб**

**ПРОЄКТУВАННЯ ІНФОРМАЦІЙНИХ СИСТЕМ методичні
вказівки до лабораторних робіт
для здобувачів вищої освіти першого (бакалаврського) рівня
спеціальності F3 – «Комп'ютерні науки»**

Комп'ютерне макетування – О. О. Смагіна
Коректор – Ю. Г. Козуб

Здано до склад. 23.04.2026 р. Підп. до друку 30.05.2026 р. Формат 60x84 1/16. Папір офсет.
Гарнітура Times New Roman. Друк ризографічний. Ум. друк. арк. 1,5. Наклад 100 прим.
Зам. № 130

Видавництво Державний заклад «Луганський національний університет імені Тараса
Шевченка»

м. Полтава, вул. Ковалю, 3

e-mail: mail@luguniv.edu.ua www.luguniv.edu.ua

Свідоцтво суб'єкта видавничої справи ДК № 3459 від 09.04.2009 р.
