

**Методичні вказівки
до виконання курсових проєктів
з дисципліни «Технології веброзробки»
зі спеціальності «Системний аналіз»**

**Міністерство освіти і науки України
Вінницький національний технічний університет**

**Методичні вказівки
до виконання курсових проєктів
з дисципліни «Технології веброзробки»
зі спеціальності «Системний аналіз»**

Вінниця
ВНТУ
2025

Рекомендовано до видання Радою з якості освіти Вінницького національного технічного університету Міністерства освіти і науки України (протокол № 3 від 23.10.2025 р.)

Рецензенти:

К. В. Овчинников, кандидат технічних наук, доцент

Є. М. Крижановський, кандидат технічних наук, доцент

Д. Х. Штофель, кандидат технічних наук, доцент

Методичні вказівки до виконання курсових проєктів з дисципліни «Технології веброзробки» зі спеціальності «Системний аналіз» [Електронний ресурс] / уклад. С. О. Жуков, А. В. Лосенко. – Вінниця : ВНТУ, 2025. – (PDF, 46 с.)

У методичних вказівках наведено рекомендації щодо виконання, а також вимоги до оформлення та змісту курсових проєктів з дисципліни «Технології веброзробки» для студентів спеціальності 124 «Системний аналіз» кафедри Системного аналізу та інформаційних технологій Вінницького національного технічного університету.

ЗМІСТ

1 ЗАГАЛЬНІ ПОЛОЖЕННЯ КУРСОВОГО ПРОЄКТУВАННЯ.....	4
1.1 Курсовий проєкт.....	4
1.2 Складові частини курсового проєкту.....	4
1.3 Загальні вимоги до виконання курсового проєкту	5
1.4 Тематика курсових проєктів	5
2 ВИМОГИ ДО ПОРЯДКУ ВИКЛАДЕННЯ МАТЕРІАЛУ КУРСОВОГО ПРОЄКТУ	6
2.1 Структура курсового проєкту	6
2.2 Титульний аркуш.....	6
2.3 Індивідуальне завдання	7
2.4 Анотація	7
2.5 Зміст.....	7
2.6 Вступ.....	8
2.7 Основна частина курсового проєкту.....	8
2.7.1 Аналіз сучасного стану питання та обґрунтування теми.....	9
2.7.2 Практична частина	9
2.8 Висновки	13
2.9 Список використаних джерел	13
2.10 Додатки.....	13
2.11 Ілюстративна частина	14
3 ВИМОГИ ДО ОФОРМЛЕННЯ ПОЯСНЮВАЛЬНОЇ ЗАПИСКИ.....	15
3.1 Загальні правила	15
3.2 Вимоги до основної частини.....	15
3.3 Вимоги до оформлення розділів та підрозділів	16
3.4 Правила написання тексту.....	17
3.5 Оформлення формул.....	19
3.6 Оформлення ілюстрацій	20
3.7 Оформлення таблиць	21
4 ПОРЯДОК ЗАХИСТУ КУРСОВОГО ПРОЄКТУ	24
РЕКОМЕНДОВАНА ЛІТЕРАТУРА	27
Додатки.....	28
Додаток А Зразок титульного аркуша до курсового проєкту	29
Додаток Б Зразок індивідуального завдання до курсового проєкту.....	30
Додаток В Зразок анотації курсового проєкту.....	32
Додаток Г Зразок змісту курсового проєкту	33
Додаток Д Форми запису літературних джерел відповідно до ДСТУ 8302:2015	34
Додаток Е Перелік тем для виконання курсового проєкту.....	36

1 ЗАГАЛЬНІ ПОЛОЖЕННЯ КУРСОВОГО ПРОЄКТУВАННЯ

Виконання курсового проєкту з обов'язкової професійної навчальної дисципліни «Технології веброзробки» передбачено навчальним планом спеціальності 124 – «Системний аналіз».

Згідно з нормативними вимогами вищої школи та інших актів законодавства України з питань освіти, курсові проєкти з дисципліни виконуються з метою закріплення, поглиблення та узагальнення знань, одержаних студентами за час навчання, та їх застосування до вирішення конкретного завдання розробки програмних засобів.

Курсовий проєкт має бути виконаний самостійно та з урахуванням вимог «Положення про запобігання академічному плагіату та порядок його виявлення у наукових, кваліфікаційних, навчальних та науково-методичних роботах у Вінницькому національному технічному університеті» [1] та «Положення про академічну доброчесність у ВНТУ» [2].

Курсові проєкти з дисципліни виконуються з метою закріплення, поглиблення і узагальнення знань, одержаних студентами за час навчання та їх застосування до вирішення конкретного завдання розробки вебтехнологій.

1.1 Курсовий проєкт

Курсовий проєкт (КП) – навчальна самостійна робота з дисципліни «Технології веброзробки», яка містить задачі навчального та прикладного характеру по створенню програмних продуктів із використанням сучасних вебтехнологій.

1.2 Складові частини курсового проєкту

Практична частина КП представляється застосуванням відомих підходів до розробки вебдодатків з метою вирішення окремої прикладної задачі.

Науково-дослідна частина передбачає пошук (опрацювання) новітніх технологій чи систем з метою вибору запропонованого варіанта або технології, дослідження окремих параметрів чи модулів (характеристик, підходів) системи тощо.

Курсовий проєкт може виконуватись за матеріалами конкретної розробки вебсистеми. Метою написання КП є закріплення теоретичних знань з курсу, розвитку вміння застосовувати їх для вирішення конкретних практичних задач, набуття навичок роботи з літературою, прикладними інформаційними технологіями та сучасними програмними бібліотеками розробки вебдодатків.

На захист курсового проєкту представляється пояснювальна записка та ілюстративні матеріали (алгоритми, рисунки інтерфейсу, схеми баз даних), обсяг яких визначається для даної дисципліни змістом спеціальності з відповідним рішенням кафедр.

1.3 Загальні вимоги до виконання курсового проєкту

У курсовому проєкті студент має розкрити зміст обраної теми, продемонструвати знання літературних джерел і нормативних актів.

Зміст КП повинен відповідати робочій програмі дисципліни «Технології веброзробки» і відображати основні аспекти теми, що розглядається.

На виконання КП відводиться 45 годин самостійної роботи студента (1,5 кредити ЄКТС).

Курсовий проєкт повинен відповідати таким вимогам:

– обсяг основної частини КП повинен складати не менше 20 сторінок формату А4;

– ілюстративні матеріали можуть бути представлені в тексті пояснювальної записки у вигляді відповідних рисунків або винесені в додатки, з обов'язковим посиланням на них в індивідуальному завданні;

– індивідуальне завдання має включати не лише різноманітні вихідні дані, а й вимагати самостійного викладу студентом тексту пояснювальної записки.

До комплексних курсових проєктів висуваються такі ж вимоги, як і до типових, однак загальний обсяг таких проєктів збільшується пропорційно кількості студентів, які працюють над спільним проєктом.

1.4 Тематика курсових проєктів

Тематика курсових проєктів визначається керівником КП відповідно до програми дисципліни. Студент також має можливість запропонувати власну тему, обґрунтувавши її важливість (наприклад, на основі наукових досліджень).

Теми курсових проєктів затверджуються на засіданні кафедри.

Об'єктом курсового проєкту може бути частина господарських чи державних наукових робіт, що виконуються в рамках наукового напрямку кафедри, за умови відповідності змісту дисципліни.

У деяких випадках завдання до курсового проєкту може передбачати виготовлення дослідного зразка, що вказується в індивідуальному завданні. У таких випадках обсяг і зміст пояснювальної записки узгоджуються між керівником і студентом. Приблизний перелік тем курсових проєктів наведено в Додатку Е.

Студенти можуть надсилати на перевірку завершений етап або повний текст КП через інструмент «Файл-Експрес» системи JetIQ у вигляді одного файлу у форматі Portable Document Format (*.pdf) відповідно до розділу 11 «Положення про курсове проєктування у Вінницькому національному технічному університеті» [1].

2 ВИМОГИ ДО ПОРЯДКУ ВИКЛАДЕННЯ МАТЕРІАЛУ КУРСОВОГО ПРОЄКТУ

2.1 Структура курсового проєкту

Загальна структура курсового проєкту з дисципліни «Технології веброзробки» передбачає такі складові: вступна частина, основна частина, додатки (за потреби) та ілюстративна частина. Вступна частина повинна містити такі складові:

- титульний аркуш;
- індивідуальне завдання;
- анотація;
- зміст;
- перелік умовних позначень, символів, одиниць, скорочень і термінів (за необхідності).

Основна частина повинна містити такі складові:

- вступ;
- змістова частина, яка включає аналітичну, розрахункову та технічну складові;
- висновки;
- список використаних джерел.

Додатки до курсового проєкту можуть містити матеріали, які не увійшли до основної частини, довідкові або допоміжні матеріали, великі рисунки або таблиці, які допомагають розкрити сутність виконаної роботи.

Пояснювальна записка має відповідати індивідуальному завданню і бути оформлена згідно з актуальними стандартами, які необхідно враховувати під час виконання роботи, з урахуванням усіх чинних офіційних змін.

Конкретний зміст пояснювальної записки (вихідні дані та перелік питань, які необхідно розробити) визначається керівником курсового проєкту.

Текст пояснювальної записки рекомендується подавати лаконічно та аргументовано.

2.2 Титульний аркуш

Титульний аркуш є першою сторінкою курсового проєкту, яка не нумерується, але враховується в загальну кількість сторінок. Він оформлюється за встановленим зразком, який наведено в додатку А.

2.3 Індивідуальне завдання

Індивідуальне завдання – це основний документ, що визначає зміст і структуру конкретного курсового проєкту, встановлює кінцевий термін подання виконаного КП. Воно повинно містити достатній обсяг вхідних даних для виконання дослідження або розробки, що є змістом курсового проєкту.

Зміст і форму індивідуального завдання визначає керівник курсового проєкту.

Індивідуальне завдання остаточно формуються керівником курсового проєкту, затверджується завідувачем кафедри, підписується керівником і здобувачем вищої освіти із зазначенням дати видачі завдання.

Індивідуальне завдання до курсового проєкту видається не пізніше другого тижня семестру, в якому виконується курсовий проєкт. Воно розміщується на другій сторінці після титульного аркуша, однак воно не нумерується і не входить до загальної кількості сторінок.

Зразок оформлення індивідуального завдання наведено у додатку Б.

2.4 Анотація

Анотація призначена для швидкого ознайомлення зі змістом та основними результатами, отриманими при виконанні курсового проєкту.

Анотація оформляється на окремому аркуші, який нумерується і враховується до загальної кількості сторінок.

Анотація повинна містити прізвище та ініціали автора курсового проєкту, тему курсового проєкту, стислий опис змісту та основних результатів курсового проєкту, а також перелік ключових слів.

Обсяг анотації – не більше однієї сторінки. Додатково анотація може подаватись однією або більшою кількістю іноземних мов. Зразок анотації наведено в додатку В.

2.5 Зміст

Зміст розміщується після анотації, починаючи з нової сторінки. Зміст нумерується і враховується до загальної кількості сторінок.

До змісту включають усі подальші структурні елементи курсового проєкту, які мають назву:

- перелік умовних позначень, символів, одиниць, скорочень і термінів (за необхідності);
- вступ;
- назви всіх розділів, підрозділів, пунктів і підпунктів (за наявності заголовків) основної частини курсового проєкту;
- висновки;
- список використаних джерел;
- назви всіх додатків (за наявності).

Для всіх елементів змісту навпроти назв вказуються номери сторінок, де починається відповідний матеріал.

Назви заголовків змісту повинні однозначно відповідати назвам заголовків пояснювальної записки за текстом. Нумерація сторінок повинна бути наскрізною. У змісті не рекомендується розривати слова знаками переносу.

Зразок змісту наведено в додатку Г.

2.6 Вступ

Вступ пишеться на новій пронумерованій сторінці з заголовком «Вступ», починаючи з абзацу. Вступ призначений для ознайомлення з предметною областю, в якій виконано курсовий проєкт, та обґрунтування теми курсового проєкту. Текст вступу має бути лаконічним, а кількість сторінок вступу не повинна перевищувати трьох.

У вступі стисло, але аргументовано висвітлюються такі питання:

- актуальність та значення теми курсового проєкту;
- оцінка сучасного стану об'єкта досліджень або розробки, сучасний рівень науки і технології за темою курсового проєкту;
- світові або регіональні тенденції розв'язання задач, поданих в індивідуальному завданні;
- мета та завдання курсового проєкту;
- об'єкт та предмет дослідження;
- галузь використання та призначення результатів дослідження;
- взаємозв'язок з іншими роботами, навчальними чи науковими проєктами (за наявності).

Якщо завдання до курсового проєкту передбачає вибір схеми з декількох варіантів, у вступі слід обґрунтувати обраний варіант.

2.7 Основна частина курсового проєкту

Пропонується така структура змістової частини курсового проєкту:

- *перший розділ* є аналітично-оглядовим і присвячений загальній характеристиці проблеми розробки вебдодатків, аналізу предметної області та її специфіки, а також огляду існуючих систем-аналогів із зазначенням їхніх переваг і недоліків і містить аналіз сучасних реалізацій таких вебдодатків, а також описує постановку задачі курсового проєкту;
- *другий розділ* включає в себе обґрунтування інструментів та технологій, що будуть застосовуватись для реалізації клієнтської та серверної частини вебдодатку;
- *третій розділ* присвячено проєктуванню архітектури вебдодатку в цілому, та опису його структурних складових, а також опис безпосередньої розробки вебдодатку.

2.7.1 Аналіз сучасного стану питання та обґрунтування теми

Цей розділ є обов'язковим і повинен містити посилання на відомі організації або спільноти, що пов'язані зі сферою веброзробки, а також відображати сучасні тенденції в цій галузі.

Порівняльний аналіз і обґрунтування теми проєкту мають виконуватись на інженерному рівні мислення із всебічним використанням сучасних науково-технічних досягнень.

У цьому розділі слід також описати основні принципи функціонування предметної області, для якої створюється вебдодаток. Повнота огляду визначається глибиною аналізу в географічному і часовому аспектах. Необхідно провести достатній аналіз патентної, науково-технічної літератури, включаючи вітчизняні та зарубіжні джерела, за останні кілька років, а також електронних ресурсів (електронних книг, сайтів, вебсторінок).

Рекомендований обсяг цього розділу становить 5-10 сторінок.

2.7.2 Практична частина

Практична частина складається з двох розділів і є найважливішою частиною пояснювальної записки курсового проєкту як за обсягом, так і за змістом. Структура практичної частини визначається темою та індивідуальним завданням курсового проєкту.

При виконанні цієї частини проєкту необхідно використовувати обґрунтований і логічний стиль викладу, враховуючи можливі варіанти розв'язання задачі на основі проведеного аналізу відомих рішень. Аргументація має підкріплюватися розрахунками, графіками, діаграмами, таблицями тощо.

Ця частина проєкту обов'язково має включати обґрунтування обраних технологій для розробки, розробку структури вебдодатку, що створюється, а саме опис реалізації клієнтської та серверної частини вебдодатку (включно із загальним описом сутностей, що будуть реалізовані у базі даних у вигляді ER-діаграми), а також загальний опис його архітектури та аргументація вибору інструментів та програмного забезпечення для розробки.

Особливу увагу необхідно приділити демонстрації ходу думок під час вибору технологій, мов програмування, бібліотек і фреймворків, що використовуються у реалізації вебсайту. Такий підхід дозволяє показати не лише кінцеве рішення, а й процес прийняття рішень на основі аналізу альтернативних варіантів, їх переваг і недоліків. Обґрунтування вибору повинно бути логічно послідовним і базуватися на вимогах до функціональності, продуктивності, безпеки та зручності подальшої підтримки вебсистеми.

Не менш важливою складовою є формування бази даних та

проектування архітектури вебсистеми. У цій частині необхідно показати структуру даних, описати взаємозв'язки між сутностями та механізми збереження й обробки інформації. Слід навести приклади архітектурних діаграм, ER-діаграм, блок-схем або інших рисунків, що ілюструють логіку взаємодії компонентів системи, процеси обробки запитів користувача та обмін даними між клієнтською і серверною частинами. Приклади таких діаграм і рисунків наведені нижче.

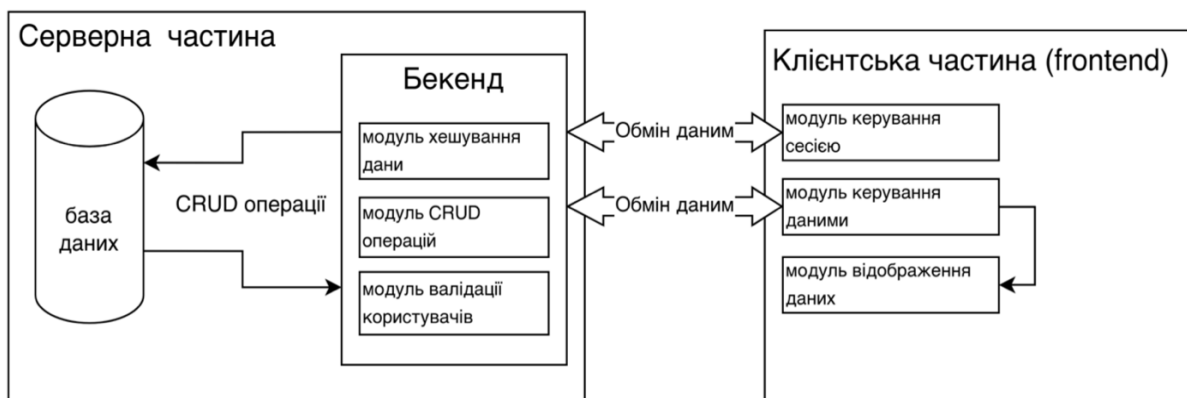


Рисунок 2.1 – Структурна схема архітектури вебсайту

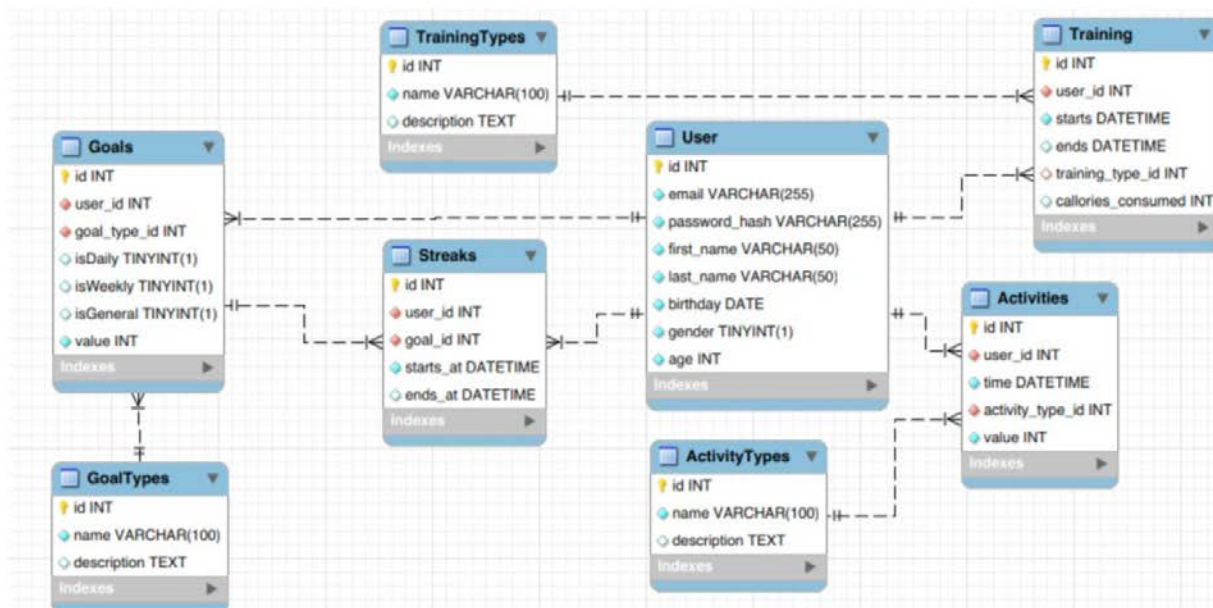


Рисунок 2.2 – Результуюча ER-діаграма типів предметної області

Окрім цього, обов'язковим елементом практичної частини є демонстрація користувацького інтерфейсу розробленого вебдодатку та наведення лістингів коду основних модулів. Це дозволяє показати рівень реалізації функціональності, структуру програмного коду, застосовані шаблони програмування, а також продемонструвати практичне втілення проектних рішень, описаних у попередніх підрозділах. Приклади інтерфейсів, фрагментів коду наведені нижче.



Рисунок 2.3 – Приклад демонстрації інтерфейсу вебсайту

The interface includes a 'Вийти' (Logout) button and a 'Привіт!' (Hello!) greeting. It features three main forms for data entry:

- Додати транзакцію (Add transaction):** Includes fields for 'Категорія' (Category: Work (income)), 'Сума' (Amount: 20000), 'Опис' (Description: Зарплата), and 'Дата' (Date: 19.06.2025).
- Додати категорію (Add category):** Includes fields for 'Назва' (Name: Work) and 'Тип' (Type: Доходи).
- Додати бюджет (Add budget):** Includes fields for 'Категорія' (Category: Work (income)), 'Ліміт' (Limit), and 'Місяць' (Month).

Below the forms is a section titled 'Ваші транзакції' (Your transactions) displaying a list item: '20000 грн — Зарплата (2025-06-19)'.

Рисунок 2.4 – Приклад демонстрації інтерфейсу вебсайту

При зображенні лістингів коду необхідно наводити лише найбільш показові фрагменти, які демонструють реалізацію ключових елементів системи. Зокрема, варто показати частини коду, що відповідають за основну функціональність вебсайту – такі як обробка запитів користувачів, робота з базою даних, авторизація, логіка бізнес-процесів чи інтеграція з зовнішніми сервісами. Кожен зразок коду має супроводжуватися коротким поясненням його призначення, контексту використання та ролі у загальній структурі програмного продукту. Загальний лістинг коду розробленого вебсайту потрібно навести в окремому додатку. Приклади таких зразків коду наведені нижче.

```

12     user,
13     create_database_if_not_exists,
14 )
15 from fastapi import FastAPI, HTTPException, Query
16 from fastapi.middleware.cors import CORSMiddleware
17 from pydantic import BaseModel
18 from sqlalchemy import and_, create_engine, or_
19 from sqlalchemy.orm import sessionmaker
20
21 app = FastAPI()
22
23 # KOPC
24 app.add_middleware(
25     CORSMiddleware,
26     allow_origins=["http://localhost:3000"],
27     allow_credentials=True,
28     allow_methods=["*"],
29     allow_headers=["*"],
30 )
31
32 DB_PATH = 'database.db'
33 engine = create_engine(f'sqlite:/// {DB_PATH}')
34 Session = sessionmaker(bind=engine)
35
36 @app.get("/db/status")
37 def db_status():
38     exists = os.path.exists(DB_PATH)
39     return {"exists": exists}
40

```

Рисунок 2.5 – Приклад лістингу коду серверної частини вебсайту

```

5
6 userRouter.get("/getuser/:id", auth, userController.getuser);
7
8 userRouter.get("/getallusers", auth, userController.getallusers);
9
10 userRouter.post("/login", userController.login);
11
12 userRouter.post("/register", userController.register);
13
14 userRouter.post("/forgotpassword", userController.forgotpassword);
15
16 userRouter.post("/resetpassword/:id/:token", userController.resetpassword);
17
18 userRouter.put("/updateprofile", auth, userController.updateprofile);
19
20 userRouter.put("/changepassword", auth, userController.changepassword);
21
22
23 userRouter.delete("/deleteuser", auth, userController.deleteuser);

```

Рисунок 2.6 – Приклад лістингу коду клієнтської частини вебсайту

2.8 Висновки

Висновки оформлюються з нової пронумерованої сторінки і починаються з абзацу. Вони є завершальною частиною, де підсумовуються конструкторські рішення, досягнуті результати та переваги розробленого вебдодатку в порівнянні з існуючими аналогами. У висновках можуть бути подані рекомендації щодо прикладного використання розробки та пропозиції щодо можливих шляхів удосконалення спроектованого вебдодатку.

Крім того, в тексті змістової частини рекомендується робити короткі висновки наприкінці кожного розділу, які водночас ставлять задачі до наступного розділу.

2.9 Список використаних джерел

Список використаних джерел, на які посилається автор в основній частині курсового проєкту, наводиться з нової сторінки після висновків і перед додатками. У список вносять всі літературні джерела, нормативні акти, охоронні документи, наукові, навчально-методичні, інструктивні джерела інформації та онлайн-ресурси, які використані при написанні курсового проєкту, у порядку згадування в тексті основної частини.

Джерела подаються у вигляді їх бібліографічних посилань відповідно до ДСТУ 8302:2015 «Інформація та документація. Бібліографічне посилання». Загальні положення та правила складання. На джерела, подані в списку, обов'язково повинні бути посилання в тексті курсового проєкту.

Бібліографічні посилання подають мовою оригіналу джерела. У списку кожне джерело записують з абзацу та нумерують арабськими цифрами, починаючи з одиниці. Порядкові номери бібліографічних посилань в списку повинні відповідати посиланням на них у тексті основної частини, де вони наводяться в квадратних дужках [] безпосередньо після згадки відповідної інформації.

Посилання в тексті пояснювальної записки на джерела вказуються їх порядковим номером із переліку посилань у квадратних дужках, наприклад: «... у роботах [1–7] ...».

Форма запису літературних джерел відповідно до ДСТУ 8302:2015 наведено в додатку Д.

2.10 Додатки

Додатки призначені для розміщення додаткових текстових, нетекстових та інших обов'язкових або допоміжних (довідкових) документів, які доповнюють зміст курсового проєкту. Додатки розміщують після основної частини курсового проєкту.

До них можуть належати ілюстрації, таблиці та тексти допоміжного

характеру. Додатки оформлюються як продовження документа і розташовуються в порядку, в якому на них посилаються в тексті пояснювальної записки.

Посилання на додатки в тексті оформлюються так: «... наведено в додатку А», «... наведено в таблиці В.5» або (додаток Б).

Кожен додаток починають з нової сторінки, зверху посередині рядка вказується слово «Додаток» і через пробіл його позначення. Додатки позначають великими українськими літерами в порядку їх слідування, за винятком букв Г, Є, З, І, Ї, Й, О, Ч, Ї. Наприклад, Додаток А, Додаток Б і так далі. Якщо кількість додатків перевищує кількість дозволених букв, використовують арабські цифри.

Якщо додаток є обов'язковим, під його позначенням вказують (обов'язковий), а для інформативного додатка – (довідковий).

Кожен додаток має мати тематичний заголовок, записаний посередині рядка малими літерами, починаючи з великої. Ілюстрації, таблиці, формули в межах додатка нумерують, зазначаючи його позначення, наприклад: «Рисунок Б.3 – Найменування»; «Таблиця В.5 – Найменування».

Нумерація сторінок документа та додатків повинна бути наскрізною. Усі додатки включаються до змісту з зазначенням номера, заголовка та сторінок, з яких вони починаються.

2.11 Ілюстративна частина

Ілюстративна частина курсового проекту є набором демонстраційного матеріалу у вигляді рисунків, світлин, графіків, схем тощо, які розкривають суть основної частини. Ілюстративна частина не потребує обов'язкового дотримання вимог стандартів і може бути оформлена як один додаток. Вимоги до ілюстративної частини зазначаються в індивідуальному завданні.

3 ВИМОГИ ДО ОФОРМЛЕННЯ ПОЯСНЮВАЛЬНОЇ ЗАПИСКИ

3.1 Загальні правила

При оформленні матеріалів курсового проєкту необхідно дотримуватись вимог «Положення про курсове проєктування у Вінницькому національному технічному університеті» [3]. Пояснювальна записка повинна бути виконана на аркушах формату А4 з дотриманням вимог до нормативно-технічних документів.

Текст пояснювальної записки має бути набраний шрифтом Times New Roman, кегль 14, чорного кольору, прямого накреслення, через півтора інтервали. Кегль шрифту може бути зменшений в таблицях, у написах на рисунка, у додатках, але не у їх назвах.

Рекомендовані розміри полів: верхнє та нижнє – не менше 20 мм, лівє – не менше 25 мм, правє – не менше 10 мм. Абзацний відступ повинен бути однаковим упродовж усього тексту і складати 1,25 см.

Нумерація сторінок проводиться наскрізь арабськими цифрами, включаючи додатки, шрифтом Times New Roman чорного кольору прямого накреслення, кегль 14. Номер сторінки проставляється праворуч у нижньому куті рамки без крапки в кінці. Титульний аркуш включається до загальної нумерації, але на ньому номер сторінки не проставляється.

Пояснювальна записка відноситься до текстових документів і має бути написана технічною мовою. Ілюстративна інформація повинна бути представлена у вигляді ілюстрацій (схеми, рисунки, графіки, діаграми тощо), а цифрові дані – у вигляді таблиць.

3.2 Вимоги до основної частини

Обсяг основної частини, як правило, встановлюється відповідно до кількості годин, передбачених для вивчення дисципліни, і не повинен перевищувати 50 сторінок разом із теоретичною частиною. Теоретична частина курсового проєкту вводиться для пояснення основних положень обраних технологій і може становити до 25% загального обсягу основної частини. Решта 75% мають бути присвячені обґрунтуванню прийнятих рішень, необхідним розрахункам та розробкам.

Практична частина повинна бути логічно пов'язана з теоретичними положеннями теми і супроводжуватися ілюстративними матеріалами (графіками, схемами, діаграмами) або таблицями з обов'язковими посиланнями на ці рисунки (таблиці) у тексті пояснювальної записки.

Забороняється переписування матеріалів з літературних джерел та сканування рисунків, пов'язаних із технічною частиною проєкту. Дозволяється використовувати скановані рисунки з довідкової літератури (в тому числі схеми) в оглядовій частині («Аналіз...»), обов'язково вказуючи джерело. Частину описового змісту, розрахунків (таблиць) та

ілюстративної інформації бажано розміщувати у додатках.

У тексті пояснювальної записки повинні бути посилання на рисунки, таблиці та додатки, що входять до змісту проекту. Ілюстративну частину можна подавати як у тексті пояснювальної записки, так і в додатках, що чітко визначаються керівником у індивідуальному завданні.

Якщо під час проектування виникає необхідність в експериментальних дослідженнях або машинному моделюванні, ця частина повинна містити детальне обґрунтування та аналіз отриманих результатів.

Розрахунки та ілюстративні матеріали, що входять до пояснювальної записки, варто виконувати за допомогою ліцензійного програмного забезпечення (наприклад, MS Office). Також можна використовувати програмні засоби, розроблені кафедрою системного аналізу та інформаційних технологій.

Якщо в проекті застосовуються сучасні іноземні програмні продукти, їх позначення слід подавати у вигляді відповідних зображень, щоб уникнути суперечностей із чинними державними стандартами. Умовні ілюстративні позначення, використовувані у програмному середовищі, можуть бути винесені в додатки окремим аркушем. У проекті потрібно вказувати назву програмного продукту і забезпечити повне розуміння його використання. В розділі моделювання слід поєднувати існуючі позначення з іноземними або отримати згоду кафедри на використання цього програмного продукту.

3.3 Вимоги до оформлення розділів та підрозділів

Структурні елементи основної частини пояснювальної записки включають розділи, підрозділи, пункти, підпункти та переліки.

Розділ – це основна одиниця поділу тексту, яка має номер і заголовок.

Підрозділ – частина розділу, також позначена номером і має заголовок.

Пункт – це частина розділу або підрозділу, позначена номером і може мати заголовок.

Підпункт – це частина пункту, також має номер і може мати заголовок.

Заголовки цих структурних елементів повинні бути пронумеровані тільки арабськими цифрами. Текст може бути розміщений між заголовками розділів та підрозділів, а також між заголовками підрозділів і пунктів.

Кожен розділ рекомендується починати на новій сторінці.

Заголовки розділів і підрозділів мають бути обов'язковими. Заголовки розділів розміщують посередині сторінки, симетрично до тексту великими літерами, напівжирним шрифтом, без крапки в кінці. Заголовки підрозділів, пунктів та підпунктів потрібно друкувати з абзацного відступу з великої літери, без крапки в кінці. Якщо заголовок складається з кількох речень, вони розділяються крапками. Не дозволяється переносити слова в заголовках.

Назви розділів, підрозділів, пунктів і підпунктів не можна розміщувати на останньому рядку сторінки.

Розділи нумеруються послідовними номерами в межах усього документа (1, 2 тощо). Після номера крапку не ставлять, а залишають один пробіл.

Підрозділи нумеруються в межах кожного розділу, пункти – в межах підрозділу за схемою (наприклад: 3.1, 3.2, 3.2.1, 3.2.2 і т.д.). Цифри, що позначають номер, не повинні виходити за межі абзацу.

Посилання на розділи у тексті виконується за зразком: «...наведено в розділі 3».

Якщо розділ або підрозділ містить лише один пункт або підпункт, його не нумерують.

У тексті документа можуть наводитися переліки, які рекомендується нумерувати малими літерами українського алфавіту з дужкою або тире перед текстом. Для подальшої деталізації використовують арабські цифри з дужкою. Кожен елемент переліку пишеться з абзацу, починаючи з малої літери, і закінчується крапкою з комою, а останній пункт – крапкою.

Приклад оформлення переліку:

а) текст переліку та його... продовження;

б) текст переліку:

1) текст переліку подальшої деталізації та його продовження;

2) . . . ;

в) останній перелік.

Примітки наводять у тексті, якщо є потреба в додаткових поясненнях до тексту, таблиць або рисунків. Примітки розміщуються одразу після тексту, під рисунком (перед його назвою) або під таблицею (у її межах). Якщо примітка одна, її не нумерують. Слово «Примітка» друкується кеглем 12 через один інтервал з абзацного відступу, з великої літери та крапкою в кінці. Після нього, через пробіл, друкується текст примітки тим самим шрифтом.

Приклад

Примітка 1. _____

Якщо приміток дві та більше, їх подають після тексту, якого вони стосуються і нумерують арабськими цифрами.

Приклад

Примітка 1. _____

Примітка 2. _____

3.4 Правила написання тексту

При написанні тексту слід дотримуватися таких правил:

а) Текст повинен бути викладений лаконічно, обґрунтовано і в

технічному стилі.

б) Умовні буквені позначення фізичних величин та ілюстративні позначення компонентів повинні відповідати встановленим стандартам. Перед буквеним позначенням фізичної величини слід надавати її пояснення (наприклад, витрата води Q , концентрація C).

в) Числа з розмірністю записують цифрами, а без розмірності – словами (наприклад: відстань – 2 мм, відміряти три рази).

г) Позначення одиниць слід писати разом із числовим значенням в одному рядку, без переносу на наступний рядок. Між числом та позначенням одиниці робиться пробіл (наприклад, 100 мг/л, 2 м³/с).

д) Якщо наводиться ряд числових значень однієї фізичної величини, одиницю вказують лише після останнього значення (наприклад: 1,5; 1,75; 2 мм).

е) Позначення величин з граничними відхиленнями слід записувати у форматі: 100 ± 5 мм.

ж) Буквені позначення одиниць у добутку розділяються крапкою на середній лінії (\cdot); знак ділення замінюють косою рисою ($/$).

и) Порядкові числівники слід записувати цифрами з відмінковими закінченнями (9-й день, 4-а лінія). Якщо кілька порядкових числівників ідуть поспіль, відмінкове закінчення додають до останнього (наприклад, 3,4,5-й графіки). Кількісні числівники записують без відмінкових закінчень (наприклад, на 20 аркушах). Закінчення не додають у датах (21 жовтня) і при римських числах (XXI століття).

к) Скорочення слів не допускаються, окрім загальноприйнятих в українській мові або скорочень, прийнятих для позначень на виробі. Такі скорочення слід виділяти великими літерами (наприклад, ON, OFF), а якщо напис складається з цифр або знаків, його беруть у лапки. Лапки також використовуються для позначення назв команд, режимів, сигналів (наприклад, «Запуск»).

л) Допускається використання математичних виразів.

м) Не дозволяється:

- Використовувати професійні або місцеві слова та вирази (техніцизми);
 - Заміняти знак множення зірочкою *;
 - Писати слово «місяць» після назви місяця (правильно: «в травні», неправильно: «в травні місяці»);
 - Використовувати вирази на зразок «цього року» або «минулого року»
- слід вказувати точну дату (наприклад, «в червні 2001 року»);
- Використовувати одиниці вимірювання без числових значень (наприклад, не «кілограми», а «кілька кілограмів»), за винятком оформлення таблиць і формул;
 - Поєднувати текст з умовними позначеннями фізичних величин за допомогою математичних знаків (правильно: «швидкість дорівнює 5 км/год», неправильно: «швидкість = 5 км/год»);
 - Використовувати математичні знаки $<$, $>$, №, %, sin, cos, tg, log тощо без цифрових або буквених позначень. У тексті слід писати словами

(наприклад, «нуль», «номер», «логарифм»);

– Використовувати індекси стандартів (ДСТУ, СніП, СТП) без реєстраційного номера.

3.5 Оформлення формул

Формули та рівняння розміщують посередині сторінки, симетрично до тексту, окремим рядком безпосередньо після згадування у тексті. Відстань між формулою (або рівнянням) та попереднім і наступним текстом повинна бути не менше одного рядка.

Номер формули або рівняння зазначають на рівні формули праворуч у крайній позиції в круглих дужках. У випадку багаторядкових формул або рівнянь номер проставляють на рівні останнього рядка.

Пояснення позначень, що входять до формули або рівняння, слід надавати безпосередньо під формулою у тій самій послідовності, в якій вони з'являються у формулі. Після формули ставлять кому, а потім із нового рядка подають пояснення до кожного символу, розділяючи їх крапкою з комою. Перший рядок пояснення починають словом «де», яке пишуть без абзацу і без жодних додаткових знаків після нього.

Усі формули нумерують у межах розділу арабськими цифрами. Номер формули вказують у круглих дужках праворуч в кінці рядка на рівні закінчення формули. Номер формули складається з номера розділу і порядкового номера формули в межах цього розділу, розділених крапкою. Допускається також нумерація формул в межах всього документа.

Приклад

Таким чином, математична модель В. А. Фролова – І. Д Родзиллера для консервативних речовин, тобто речовин, які не вступають в хімічні реакції:

$$\frac{dx(t)}{dt} = -F(t) \cdot [x(t) - x^*], x(0) = x_0, \quad (3.1)$$

де x^ – значення концентрації речовини у так званому створі повного змішування;*

$F(t)$ – деяка нелінійна функція, вираз якої виведений В. А. Фроловим на основі аналізу розмірностей – характеризує зменшення концентрації x за рахунок процесів розбавлення.

Одиницю вимірювання, при необхідності, беруть в квадратні дужки

$$I = \frac{U}{R} [A]. \quad (3.2)$$

Числову підстановку і розрахунок виконують з нового рядка не нумеруючи. Одиницю вимірювання беруть в круглі дужки. Наприклад,

$$I = \frac{220}{100} [A].$$

Розмірність одного й того ж параметра повинна залишатися незмінною в межах усього документа.

Якщо формула занадто довга, її можна переносити на наступні рядки. Перенос виконують за допомогою математичних знаків, повторюючи знак на початку наступного рядка. При цьому знак множення «·» замінюють на «×».

Формули є частиною речення, тому до них застосовуються ті ж правила пунктуації, що й до інших членів речення. Якщо формула стоїть наприкінці речення, після неї ставлять крапку. Формули, які слідують одна за одною без тексту між ними, розділяють комами.

Посилання на формули в тексті дають в круглих дужках за формою: «... в формулі (3.1)»; «... в формулах (3.1, ..., 3.5)».

3.6 Оформлення ілюстрацій

Для наочного пояснення тексту рекомендується використовувати графіки, креслення та інші ілюстрації, які можна виконувати чорною тушшю, простим олівцем середньої твердості або за допомогою комп'ютерної графіки.

Ілюстрації розміщують або в тексті, або в додатках. У тексті їх розташовують симетрично до тексту після першого посилання на них або на наступній сторінці, якщо ілюстрація не поміщається на поточній сторінці без повороту.

Усі ілюстрації в тексті пояснювальної записки мають супроводжуватися посиланнями, наприклад: «... показано на рисунку 3.1» або в дужках (рисунок 3.1). Якщо посилаються на частину ілюстрації, зазначають: «... показано на рисунку 3.2, б». Посилання на раніше наведені ілюстрації оформлюють зі скороченим словом «див.» (наприклад, (див. рисунок 1.3)).

Між ілюстрацією та текстом пропускають один рядок (еквівалентно 3 інтервали). Усі ілюстрації позначають як «рисунки» і нумерують під ілюстрацією симетрично за зразком: «Рисунок 3.5 – Найменування рисунка». Крапку в кінці не ставлять, а знак переносу не використовують. Якщо назва рисунка занадто довга, її продовжують у наступному рядку, починаючи від самого найменування.

Ілюстрації нумерують у межах розділів, використовуючи номер розділу і порядковий номер ілюстрації, розділені крапкою (наприклад, 3.1, 3.2 тощо). Дозволяється нумерація в межах всього документа.

Графу «№ з/п» не використовують. У разі необхідності нумерації, номери вказують у боковій частині перед найменуванням рядка.

Заголовки граф можуть складатися із заголовків та підзаголовків, які записують в однині, симетрично до тексту графи, малими літерами, починаючи з великої. Якщо підзаголовок складається з одного речення із заголовком, його починають з малої літери. У кінці заголовків і підзаголовків граф крапку не ставлять. Допускається оформлення заголовків та підзаголовків через один інтервал.

Якщо всі величини в таблиці мають однакову одиницю виміру, її скорочене позначення розміщують над таблицею (наприклад, мм). Якщо ж параметри мають різні одиниці виміру, то їх зазначають у заголовках граф після коми (наприклад, «Довжина, мм»).

Текст заголовків граф можна замінювати буквеною символікою, якщо ці символи пояснені у попередньому тексті або на ілюстраціях (наприклад, D – діаметр, H – висота). Однакові буквені позначення групують послідовно в порядку зростання індексів, наприклад: (L1, L2, ...).

Найменування рядків записують у боковій частині таблиці як заголовки в називному відмінку однини, малими літерами, починаючи з великої, і з однієї позиції. Крапку в кінці заголовків не ставлять. Позначення одиниць виміру фізичних величин вказують у заголовках після коми.

Для опису визначеного інтервалу значень в найменуваннях граф і рядків таблиці можна використовувати слова: «більше», «менше», «не більше», «не менше», «в межах». Ці слова розміщують після одиниці фізичної величини:

(Концентрація, C, не більше),

а також використовують слова «від», «більше», «до»:

(Від 10 до 15; більше 15; до 20)

Дані, що наводяться в таблиці, можуть бути словесними і числовими. Слова записують в графах з однієї позиції. Якщо рядки таблиці не розділені лініями, то текст, який повторюється і складається з одного слова дозволяється замінювати лапками («»). Якщо текст складається з двох і більше слів, то при першому повторенні його замінюють словами «те ж», а далі лапками. При розділенні таблиці горизонтальними лініями – ніякої заміни не виконують.

Числа записують посередині графи так, щоб їх однакові розряди по всій графі були точно один під одним, за винятком випадку, коли вказують інтервал. Інтервал вказують від меншого числа до більшого з тире між ними:

12–35, 122–450.

Дробові числа наводять у вигляді десяткових дробів, з однаковою кількістю знаків після коми в одній графі. Розміри в дюймах можна записувати у вигляді: $1/2''$, $1/4''$, $1/8''$.

Ставити лапки замість цифр чи математичних символів, які

повторюються, не можна. Якщо цифрові чи інші дані в таблиці не наводяться, то ставиться риска.

Таблиці нумерують в межах розділів і позначають зліва над таблицею з абзацу за формою: «Таблиця 4.2 – Найменування таблиці». Крапку в кінці не ставлять. Номер таблиці складається з номера розділу і порядкового номера таблиці в розділі, розділених крапкою. Дозволяється нумерувати в межах всього документа.

Таблиця може бути великою як в горизонтальному, так і в вертикальному напрямках або іншими словами може мати велику кількість граф і рядків. В таких випадках таблицю розділяють на частини і переносять на інші сторінки або розміщують одну частину під іншою чи поряд.

Якщо частини таблиці розміщують поряд, то в кожній частині повторюють головку таблиці, а при розміщенні однієї частини під іншою – повторюють боковик.

Якщо в кінці сторінки таблиця переривається і її продовження буде на наступній сторінці, в першій частині таблиці нижню горизонтальну лінію, що обмежує таблицю, не проводять.

При перенесенні частин таблиці на інші сторінки, повторюють або продовжують найменування граф. Допускається виконувати нумерацію граф на початку таблиці і при перенесенні частин таблиці на наступні сторінки повторювати тільки нумерацію граф.

У всіх випадках найменування (при його наявності) таблиці розміщують тільки над першою частиною, а над іншими частинами зліва пишуть «Продовження таблиці 4.2» без крапки в кінці.

Інші вимоги до виконання таблиць – відповідно до чинних стандартів на технічну документацію.

4 ПОРЯДОК ЗАХИСТУ КУРСОВОГО ПРОЄКТУ

Порядок захисту курсового проєкту визначається «Положенням про курсове проєктування у ВНТУ» [3].

Здобувачі вищої освіти можуть надсилати на перевірку завершений етап або повний текст КР через інструмент «Файл-Експрес» системи JetIQ у вигляді одного файлу у форматі Portable Document Format (*.pdf).

Керівник зобов'язаний дати відповідь на надісланий файл протягом семи діб з моменту отримання файлу. У відповіді керівник повинен вказати, що він приймає етап чи повний текст КР до захисту або зазначити, що потрібно виправити чи доопрацювати. Під час прийняття рішення враховується якість виконання завдань та оформлення КР, а також відповідність вимогам академічної доброчесності.

Повернення КР для виправлення чи доопрацювання допускається тільки протягом теоретичного семестру. Файли, надіслані після завершення теоретичного семестру, вважаються остаточними, а їх заміна або виправлення не допускаються.

Якщо КР виконано у повному обсязі, у відповідності до індивідуального завдання, не містить ознак академічної недоброчесності, не містить суттєвих помилок, оформлена згідно з встановленими вимогами та надіслана здобувачем вищої освіти у вигляді одного файлу у форматі Portable Document Format (*.pdf), керівник приймає КР до захисту, про що повідомляє здобувача вищої освіти у відповіді на файл, надісланий через інструмент «Файл-Експрес» системи JetIQ.

Якщо остаточний файл КР виконано не в повному обсязі, або не у відповідності до індивідуального завдання, або він містить ознаки академічної недоброчесності, або суттєві помилки, або не оформлено згідно з встановленими вимогами, або файл не надійшов через інструмент «Файл-Експрес» системи JetIQ (станом на визначений у розкладі день захисту), така КР визнається керівником недопущеною до захисту із виставленням незадовільної оцінки у відомість успішності (від 0 до 59 балів). Якщо оцінка за стобальною шкалою склала від 35 до 59 балів включно, то здобувач вищої освіти має право на доопрацювання і захист КР з виставленням оцінки у другу відомість. Якщо оцінка за стобальною шкалою склала від 0 до 34 балів включно, то здобувач вищої освіти вважається таким, що має академічну заборгованість. Для її ліквідації здобувач повинен виконати КР за новою темою (або зміненим індивідуальним завданням).

Для захисту курсових проєктів кафедрою призначається комісія у складі не менше двох викладачів, один з яких керівник КР.

Студенти, які виконали навчальний план з дисципліни, допускаються до складання іспиту незалежно від захисту КР з цієї дисципліни.

Захист курсового проєкту проводиться публічно за встановленим графіком перед комісією, склад якої затверджується завідуючим кафедрою.

Якщо здобувач вищої освіти не з'явився на захист КР у зв'язку з поважною причиною (перешкода стихійного характеру, хвороба, воєнні дії на території перебування здобувача або інші обставини, які позбавили його можливості особисто і своєчасно прибути на захист), такому здобувачу вищої освіти надається можливість захистити КР в інший день за узгодженням з керівником, але не пізніше останнього дня заліково-екзаменаційної сесії. Про поважну причину неявки на захист здобувач вищої освіти повинен повідомити керівника і деканат відповідного факультету протягом доби від призначеного часу захисту.

Якщо здобувач вищої освіти не з'явився на захист КР без поважної причини, такий здобувач вищої освіти отримує незадовільну оцінку (від 0 до 59 балів за стобальною шкалою) з виставленням її у відомість успішності.

На захисті студент робить доповідь з теми до 5-10 хвилин, в якій в стислому вигляді повинен розкрити: актуальність теми, структуру проєкту, об'єкт дослідження, висновки з аналізом, а також продемонструвати розроблений ним програмний продукт. Після чого члени комісії задають питання теоретичного і практичного характеру на тему проєкту. За результатами захисту комісія на закритому засіданні визначає оцінку, яка потім оголошується студенту. Курсовий проєкт оцінюється по 100-бальній системі.

При оцінці курсового проєкту до уваги беруться такі критерії:

- ступінь і якість виконання індивідуального завдання;
- ступінь і якість виконання основної частини КП;
- ступінь і якість виконання ілюстративної частини;
- відповідність встановленим вимогам до змісту і оформлення КП;
- повнота розкриття теми і розгляд відповідної проблеми;
- глибина використаних знань теоретичних основ бізнес-аналізу великих масивів даних та застосування моделей аналізу даних;
- використання програмних засобів бізнес-аналітики при розгляді конкретної проблеми;
- склад і обсяг використаних літературних джерел;
- правильність проведених розрахунків по досліджуваному об'єкту;
- теоретична і практична повнота та істотність висновків;
- рівень і якість представлення результатів курсового проєктування здобувачем вищої освіти під час захисту;
- відповіді на запитання в процесі захисту КП;
- уміння студента формулювати суть питання, розкривати, доводити і відстоювати свою точку зору на проблему в процесі захисту.

Критерії оцінювання КП наведені в таблиці 4.1.

У випадку виявлення керівником проєкту або членами комісії факту несамотійного виконання роботи, студент до захисту курсового проєкту не допускається.

Таблиця 4.1 – Критерії оцінювання знань, умінь та навичок студентів

<u>Рівень компетентності</u>	<u>За бальною шкалою</u>	<u>За шкалою ЄКТС</u>	<u>Критерії оцінювання</u>
<u>IV</u> <u>Високий</u> <u>(творчий)</u>	90–100	<u>A</u>	КП виконано якісно, без помилок, при відповіді на питання здобувач вищої освіти виявляє всебічні, систематизовані, глибокі знання змісту КП, змісту й алгоритму робіт на усіх етапах виконання КП у відповідності з індивідуальним завданням, вільне орієнтування в джерелах інформації і нормативних документах, навички роботи з технічною документацією на рівні творчого використання.
<u>III</u> <u>Достатній</u> <u>(конструктивний)</u>	82–89	<u>B</u>	КП виконано якісно, без суттєвих помилок. Повні знання з поставлених питань і задач. Уміння викладати основні ідеї та результати. Уміння професійно відстоювати свій погляд. Допускаються несуттєві неточності у викладенні матеріалу та у відповідях.
	75–81	<u>C</u>	КП виконано з незначними помилками. Достатньо повні знання з поставлених питань і задач. Уміння викладати основні ідеї та результати. Здатність самостійно застосовувати матеріал на рівні стандартних ситуацій, наводити власні приклади на підтвердження власних тверджень. Уміння доводити правильність своїх рішень. Несуттєві неточності у відповідях.
<u>II</u> <u>Середній</u> <u>(репродуктивний)</u>	64–74	<u>D</u>	У КП допущена одна суттєва помилка. Здобувач вищої освіти може відтворити значну частину опрацьованого матеріалу й отриманих результатів, виявляє знання та розуміння основних положень, з допомогою викладача може аналізувати матеріал та робити висновки. Пояснення неповні, не завжди точні. Відповіді на питання неповні, містять неточності.
	60–63	<u>E</u>	У КП допущено дві суттєвих помилки. Задовільні знання матеріалу на рівні, вищому за початковий. Здатність за допомогою викладача логічно відтворювати значну частину матеріалу і отриманих результатів. Дає відповіді не на всі запитання, або відповіді неповні, містять неточності.
<u>I</u> <u>Низький</u> <u>(недостатній)</u>	35–59	<u>FX</u>	У КП виявлено більше двох суттєвих помилок. Здобувач вищої освіти володіє теоретичним матеріалом на фрагментарному рівні, викладає матеріал і результати курсового проектування уривчасто, з помилками. Не може обґрунтувати прийняті конструкторські або проєктні рішення, на запитання комісії дає неправильні відповіді (40–60 %), або відповіді й пояснення не дають змоги оцінити знання і навички.
	0–34	<u>F</u>	У КП виявлено більше двох суттєвих помилок. Теорією володіє на рівні фрагментів, викладає матеріал уривчасто. Не може обґрунтувати прийняті рішення, на запитання викладача дає неправильні відповіді (60–100 %).

РЕКОМЕНДОВАНА ЛІТЕРАТУРА

1. Положення про запобігання академічному плагіату та порядок його виявлення у наукових, кваліфікаційних, навчальних та науково-методичних роботах у Вінницькому національному технічному університеті. 2024. URL: <https://vntu.edu.ua/uploads/2024/Stateofplag.pdf> (дата звернення: 23.08.2025).
2. Положення про академічну доброчесність у ВНТУ. 2024. URL: <https://vntu.edu.ua/uploads/2022/acad.pdf> (дата звернення: 23.08.2025).
3. Положення про курсове проектування у Вінницькому національному технічному університеті / уклад. Д. Х. Штофель. Вінниця : ВНТУ, 2024. 52 с. URL: <https://vntu.edu.ua/uploads/2024/StateofKurs.pdf> (дата звернення: 23.08.2025).
4. ДСТУ 3008:2015. Звіти у сфері науки і техніки. Структура та правила оформлювання. [Чинний з 2015-06-22]. Київ : Держспоживстандарт України, 2015. 31 с. (Система стандартів з інформації та документації) (Національний стандарт України).
5. ДСТУ 8302:2015. Бібліографічне посилання. Загальні положення та правила складання. [Чинний з 2016-07-01]. Київ : Держспоживстандарт України, 2016. 17 с. (Система стандартів з інформації та документації) (Національний стандарт України).
6. Молчанов В. П., Пандорін О. К. Технології розробки Web-ресурсів : електрон. навч. посіб. Харків : ХНЕУ ім. С. Кузнеця, 2019. 130 с.
7. Трофименко О. Г., Козін О. Б., Задерейко О. В., Плачінда О. Є. Веб-технології та веб-дизайн : електрон. навч. посіб. Одеса : Фенікс, 2019. 284 с.
8. Базові поняття і терміни веб-технологій / А. В. Кільченко та ін.; за ред. А. В. Кільченка. Київ : ІТЗН НАПН України, 2014. 49 с.
9. Бородкіна І. Л., Бородкін Г. О. Web-технології та Web-дизайн: застосування мови HTML для створення електронних ресурсів : навч. посіб. Київ : Видавництво Ліра-К, 2020. 212 с.
10. Пустюльга С. І., Самчук В. П. Технології вебдизайну : навч. посіб. Луцьк : Вежа, 2023. 604 с.

Додатки

Додаток А
Зразок титульного аркуша до курсового проєкту

Вінницький національний технічний університет
Факультет інтелектуальних інформаційних технологій та автоматизації
Кафедра системного аналізу та інформаційних технологій

КУРСОВИЙ ПРОЄКТ

з дисципліни «Технології веброзробки»
на тему: «Розробка програмного забезпечення для автоматизації бізнес-процесів у сфері фінансів»

08-34.ТВР.000.00.000 ПЗ

Виконав студент 2-го курсу групи
СА-22б
спеціальності 124 «Системний
аналіз»
Петро Петренко

Керівник асистент каф. САІТ
PhD, Арсен ЛОСЕНКО

Кількість балів: _____

Оцінка ЄКТС _____

Члени комісії:

Лосенко А. В.

Жуков С. О.

Вінниця – 2025 рік

Додаток Б
Зразок індивідуального завдання до курсового проєкту

Вінницький національний технічний університет
Факультет інтелектуальних інформаційних технологій та автоматизації
Кафедра системного аналізу та інформаційних технологій
Рівень вищої освіти перший (бакалаврський)
Спеціальність «Системний аналіз»
Освітньо-професійна програма «Системний аналіз»

ЗАТВЕРДЖУЮ

Завідувач кафедри САІТ
д. т. н., професор Віталій МОКІН
« ____ » _____ 202__ року

ІНДИВІДУАЛЬНЕ ЗАВДАННЯ

на курсовий проєкт з дисципліни «Технології веброзробки»
Петренку Петру Петровичу, студенту гр. СА-22б

Тема проєкту: Розробка програмного забезпечення для автоматизації бізнес-процесів у сфері фінансів.

Завдання: розробити план розробки програмного продукту згідно з послідовністю етапів життєвого циклу ПЗ.

Вхідні дані до виконання проєкту:

Файл розкладу навчального процесу
База даних відомостей про навчальні дисципліни
Документація бібліотеки розробки серверної частини FastAPI

Зміст пояснювальної записки:

Анотація
Зміст
Вступ
1 Огляд сучасних реалізацій аналогічних вебдодатків
2 Вибір технологій та бібліотек для розробки вебдодатку
3 Проектування архітектури вебдодатку та розробка

Висновки

Список використаних джерел

Додатки

Додаток А (обов'язковий) Ілюстративна частина

Зміст ілюстративної частини: Блок-схема взаємодії з програмним продуктом, ER-діаграма сутностей бази даних, діаграма архітектури вебдодатку, макети інтерфейсу додатку.

Дата видачі завдання: 15 січня 2025 р.

Строк подання завершеного проєкту: 10 червня 2025 р.

Керівник _____ Арсен ЛОСЕНКО

Завдання отримав _____ Петро ПЕТРЕНКО

Додаток В
Зразок анотації курсового проєкту

АНОТАЦІЯ

Петренко П. П. Розробка програмного забезпечення для автоматизації бізнес-процесів у сфері фінансів. Курсовий проєкт з дисципліни «Технології веброзробки».

У курсовому проєкті реалізовано застосування відповідного програмного забезпечення та прикладних програмних бібліотек для розробки вебсистеми. Проаналізовано існуючі аналоги таких вебсистем, досліджена їх функціональність та методи взаємодії.

Під час виконання проєкту було використано бібліотеки та фреймворки мов програмування Python та Javascript. У ході дослідження розроблено та протестовано клієнтську та серверну частини вебсистеми. В рамках проєктування побудовано ER-діаграми сутностей, що містяться у базі даних серверної частини вебсистеми, узагальнені діаграми архітектури вебдодатку та опис його структурних складових.

Результати реалізовані мовою програмування Python та Javascript із застосуванням бібліотек FastAPI, Node.js, React.

Ключові слова: веброзробка, клієнт-серверна архітектура, бази даних, Javascript, Python, React, FastAPI.

Додаток Г
Зразок змісту курсового проєкту

ЗМІСТ

ВСТУП.....	3
1 ОГЛЯД СУЧАСНИХ РЕАЛІЗАЦІЙ АНАЛОГІЧНИХ ВЕБДОДАТКІВ.....	5
1.1 Аналіз предметної області.....	5
1.2 Огляд існуючих реалізацій.....	7
1.3 Постановка задачі на розробку програмного забезпечення	13
2 ВИБІР ТЕХНОЛОГІЙ ТА БІБЛІОТЕК ДЛЯ РОЗРОБКИ ВЕБДОДАТКУ	14
2.1 Обґрунтування вибору середовища та мови програмування	10
2.2 Обґрунтування вибору технології для реалізації клієнтської частини .	14
2.3 Обґрунтування вибору технологій для реалізації серверної частини ...	16
3 ПРОЄКТУВАННЯ АРХІТЕКТУРИ ВЕБДОДАТКУ ТА РОЗРОБКА ..	22
3.1 Опис структурних частин вебдодатку	22
3.2 Проектування архітектури вебдодатку	
3.3 Розробка клієнтської частини вебдодатку.....	27
3.4 Розробка серверної частини вебдодатку.....	27
ВИСНОВКИ.....	33
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	34
Додаток А (довідковий) Зразки лістингу коду розробленого додатку.....	35
Додаток Б (довідковий) Приклади інтерфейсу користувача	39
Додаток В (обов'язковий) Ілюстративна частина.....	42

					08-34.ТВР.000.00.000 ПЗ			
Змн.	Арк.	№ докум.	Підпис	Дата	Розробка програмного забезпечення для автоматизації бізнес-процесів у сфері фінансів	Літ.	Арк.	Архівів
Розроб.		Петренко П.П.					4	37
Перевіс.		Лосенко А.В.				СА-226		
Решенз.								
Н. Контр.		Лосенко А.В.						
Затверд.								

Додаток Д
Форми запису літературних джерел відповідно до
ДСТУ 8302:2015

Характеристика джерела	Приклад оформлення
Книги з одним автором	Мокін В. Б. Математичні моделі для контролю та управління якістю річкових вод : монографія. Вінниця : УНІВЕРСУМ-Вінниця, 2005. 172 с.
Книги із двома авторами	Мокін В. Б., Мокін Б. І. Математичні моделі та програми для оцінювання якості річкових вод : монографія. Вінниця : УНІВЕРСУМ-Вінниця, 2000. 152 с.
Книги із трьома авторами	Мокін В. Б., Крижановський Є. М., Боцула М. П. Інформаційна технологія інтегрування математичних моделей у геоінформаційні системи моніторингу поверхневих вод : монографія. Вінниця : ВНТУ, 2011. 152 с.
Книги із чотирма авторами	Методи та засоби комп'ютерних обчислень : електронний навчальний посібник / Є. М. Крижановський, В. Б. Мокін, Г. В. Горячев, В. І. Варчук. Вінниця : ВНТУ, 2016. 90 с.
Книги із п'ятьма і більше авторами	Інформаційні технології автоматизації обробки параметрів геоінформаційних систем з геометричними мережами : монографія / В. Б. Мокін та ін. ; за ред. В. Б. Мокіна. Вінниця : ВНТУ, 2014. 196 с.
Матеріали конференцій, з'їздів	Мокін В. Б., Варчук І. В. Геоінформаційна технологія оптимізації топологічної спостережуваності багатозв'язних просторово-розподілених систем. <i>Summer InfoCom Advanced Solutions 2016</i> : матеріали II Міжнародної науково-практичної конференції, м. Київ, 1–3 черв. 2016 р. Київ, 2016. С. 37–39.
Законодавчі та нормативні документи	1. Деякі питання стипендіального забезпечення : Постанова Кабінету Міністрів України від 28.12.2016 р. № 1050. <i>Офіційний вісник України</i> . 2017. № 4. С. 530–543. 2. Про вищу освіту : Закон України від 01.07.2014 р. № 1556-VII. Дата оновлення: 28.09.2017. URL: http://zakon2.rada.gov.ua/laws/show/1556-18

Стандарти	ДСТУ 3582:2013. Бібліографічний опис. Скорочення слів і словосполучень українською мовою. Загальні вимоги та правила (ISO 4:1984, NEQ; ISO832:1994, NEQ). [На заміну ДСТУ 3582-97; чинний від 2013-08-22]. Вид. офіц. Київ : Мінекономрозвитку України, 2014. 15 с. (Інформація та документація).
Дисертації	Варчук І. В. Інформаційна технологія аналізу та оптимізації топологічної спостережуваності багатозв'язних геоінформаційних систем : дис. ... канд. техн. наук : 05.13.06 / Вінницький національний технічний університет. Вінниця, 2016. 151 с.
Автореферати дисертацій	Варчук І. В. Інформаційна технологія аналізу та оптимізації топологічної спостережуваності багатозв'язних геоінформаційних систем : автореф. дис. ... канд. техн. наук : 05.13.06. Київ, 2016. 24 с.
Патенти	Люмінісцентний матеріал: пат. 25742 Україна : МПК6 C09K11/00, G01T1/28, G21H3/00. №200701472 ; заявл. 12.02.07 ; опубл. 27.08.07, Бюл. № 13. 4 с.
Електронні ресурси	<p>Мокін В. Б., Крижановський Є. М., Скорина Л. М., Гораш А. М. Побудова ГІС-інтегрованої системи даних та моделей на основі XML-формалізації для моделювання процесів у річках. <i>Наукові праці Вінницького національного технічного університету</i>. Вінниця, 2018. № 2. URL: https://doi.org/10.31649/2307-5376-2018-2-42-51 (дата звернення: XX.XX.XXXX)</p> <p>Варчук І. В., Мокін В. Б. Технологія ідентифікації та оптимізації топологічної спостережуваності багатозв'язних просторово-розподілених систем за їх математичними та геоінформаційними моделями. <i>XLV Регіональна науково-технічна конференція професорсько-викладацького складу, співробітників та студентів університету з участю працівників науково-дослідних організацій та інженерно-технічних працівників підприємств м. Вінниці та області</i> : зб. матеріалів конференції. Вінниця, 2016. URL: http://ir.lib.vntu.edu.ua/bitstream/handle/123456789/10598/858.pdf (дата звернення: XX.XX.XXXX)</p>

Додаток Е

Перелік тем для виконання курсового проєкту

1. Розробка програмного забезпечення для автоматизації бізнес-процесів у сфері фінансів.

Постановка задачі: Розробити вебдодаток для ведення фінансового обліку малого бізнесу.

Можливості системи: управління рахунками, категоризація витрат і доходів, автоматичний розрахунок балансу, експорт звітів у CSV/PDF.

Технології: Django, FastAPI, PostgreSQL, HTML, Tailwind CSS, HTMX, Pandas, NumPy.

2. Розробка системи автоматизованого тестування вебдодатків.

Постановка задачі: Створити інструмент для автоматизованого тестування вебдодатків.

Можливості системи: тестування UI, тестування API, генерація звітів.

Технології: Python, Selenium, Playwright, PyTest, SQLite.

3. Розробка програмного забезпечення для автоматизації складського обліку та логістики.

Постановка задачі: Розробити вебдодаток для управління товарними залишками.

Можливості системи: додавання товарів, контроль залишків, формування замовлень, генерація звітів.

Технології: Django, FastAPI, PostgreSQL, HTML, Tailwind CSS, HTMX.

4. Розробка аналітичної системи для обробки великих обсягів даних.

Постановка задачі: Розробити систему для обробки та аналізу великих даних.

Можливості системи: імпорт CSV, агрегування даних, побудова графіків.

Технології: Django, FastAPI, PostgreSQL, Pandas, NumPy, Chart.js, HTMX.

5. Розробка вебдодатку для автоматизації бізнес-процесів у сфері торгівлі.

Постановка задачі: Створити вебдодаток для управління товарами та замовленнями.

Можливості системи: каталог товарів, управління замовленнями, інтеграція з платіжними системами.

Технології: Django, FastAPI, HTML, Tailwind CSS, HTMX, Stripe API.

6. Розробка інтерактивного вебдодатку для онлайн-освіти та дистанційного навчання.

Постановка задачі: Розробити вебплатформу для проведення онлайн-курсів.

Можливості системи: реєстрація студентів, завантаження навчальних матеріалів, тестування.

Технології: Django, FastAPI, PostgreSQL, HTML, Tailwind CSS, HTMX.

7. Розробка геолокаційних сервісів та додатків для навігації та маршрутизації.

Постановка задачі: Розробити вебдодаток для визначення маршрутів.

Можливості системи: введення початкової та кінцевої точок, розрахунок маршруту, відображення карти.

Технології: Django, FastAPI, HTML, Tailwind CSS, Leaflet.js, OpenStreetMap API.

8. Розробка вебдодатку для обробки та аналізу геоданих та картографічної інформації.

Постановка задачі: Створити вебсистему для аналізу геоданих.

Можливості системи: завантаження файлів з координатами, відображення на карті, обробка геоданих.

Технології: Django, FastAPI, HTML, Tailwind CSS, GeoDjango, PostGIS.

9. Розробка програмного забезпечення для управління бізнес-процесами у сфері телекомунікацій.

Постановка задачі: Розробити вебсистему для обліку клієнтів і тарифних планів.

Можливості системи: управління абонентами, контроль тарифних планів, збір статистики.

Технології: Django, FastAPI, PostgreSQL, HTML, Tailwind CSS, HTMX.

10. Розробка інтелектуальної системи підтримки прийняття рішень для бізнесу та управління.

Постановка задачі: Розробити систему для аналізу бізнес-даних.

Можливості системи: введення показників, побудова аналітичних звітів, прогнозування тенденцій.

Технології: Django, FastAPI, HTML, Tailwind CSS, Chart.js, Pandas, Scikit-Learn.

11. Розробка вебдодатку для автоматизації управління бізнес-процесами у сфері туризму та гостинності.

Постановка задачі: Розробити вебдодаток для управління бронюванням, розкладом подорожей та обслуговуванням клієнтів у сфері туризму.

Можливості системи: управління бронюванням готелів і турів, інтеграція з платіжними системами, автоматичні нагадування, аналіз відгуків клієнтів.

Технології: Django, FastAPI, PostgreSQL, HTML, Tailwind CSS, HTMX, Stripe API.

12. Розробка вебсистеми для візуалізації та аналізу даних за допомогою інтерактивних графіків та діаграм.

Постановка задачі: Розробити вебдодаток для візуального аналізу даних з можливістю інтерактивної взаємодії.

Можливості системи: завантаження даних, побудова інтерактивних графіків, аналіз трендів, експорт графіків у зображення.

Технології: Django, FastAPI, PostgreSQL, HTML, Tailwind CSS, Chart.js, D3.js, HTMX.

13. Розробка системи ідентифікації та аутентифікації користувачів для забезпечення кібербезпеки та захисту від несанкціонованого доступу.

Постановка задачі: Розробити вебсистему для безпечної автентифікації користувачів.

Можливості системи: підтримка двофакторної аутентифікації, управління ролями користувачів, шифрування даних, інтеграція з OAuth 2.0.

Технології: Django, FastAPI, PostgreSQL, HTML, Tailwind CSS, JWT, OAuth 2.0, HTMX.

14. Розробка системи аналізу та прогнозування трафіку в мережах зв'язку та телекомунікацій.

Постановка задачі: Розробити вебдодаток для аналізу та прогнозування мережевого трафіку.

Можливості системи: моніторинг трафіку, аналіз пікових навантажень, прогнозування використання мережевих ресурсів.

Технології: Django, FastAPI, PostgreSQL, HTML, Tailwind CSS, Pandas, Scikit-Learn, HTMX.

15. Розробка аналітичної вебсистеми для виявлення та аналізу паттернів у фінансових даних для прийняття інвестиційних рішень.

Постановка задачі: Розробити вебдодаток для аналізу фінансових ринків та виявлення інвестиційних можливостей.

Можливості системи: завантаження фінансових даних, аналіз трендів, прогнозування ризиків, генерація інвестиційних рекомендацій.

Технології: Django, FastAPI, PostgreSQL, HTML, Tailwind CSS, Pandas, Scikit-Learn, HTMX.

16. Розробка вебдодатку для аналізу та прогнозування погодних умов на основі даних з супутників та метеорологічних станцій.

Постановка задачі: Розробити вебсистему для збору, аналізу та прогнозування погодних умов.

Можливості системи: отримання метеоданих, побудова прогнозів, візуалізація кліматичних змін.

Технології: Django, FastAPI, PostgreSQL, HTML, Tailwind CSS, OpenWeather API, Pandas, HTMX.

17. Розробка інтелектуальної вебсистеми аналізу та моніторингу екологічних параметрів та забруднень навколишнього середовища.

Постановка задачі: Розробити вебдодаток для збору та аналізу даних про стан навколишнього середовища.

Можливості системи: моніторинг рівня забруднення повітря, аналіз екологічних змін, візуалізація даних на карті.

Технології: Django, FastAPI, PostgreSQL, HTML, Tailwind CSS, Leaflet.js, Pandas, HTMX.

18. Розробка системи відстеження та контролю виробничих процесів для підвищення ефективності та якості виробництва.

Постановка задачі: Розробити вебдодаток для моніторингу виробничих процесів.

Можливості системи: контроль етапів виробництва, відстеження продуктивності, автоматичне формування звітів.

Технології: Django, FastAPI, PostgreSQL, HTML, Tailwind CSS, HTMX, Pandas.

19. Розробка вебдодатку для бронювання зустрічей та управління розкладом

Постановка задачі: Розробити вебдодаток для планування зустрічей та управління графіком.

Можливості системи: створення подій, інтеграція з календарями, нагадування про зустрічі, управління доступом до розкладу.

Технології: Django, FastAPI, PostgreSQL, HTML, Tailwind CSS, FullCalendar.js, HTMX.

20. Розробка вебсистеми для ведення персональних фінансів та управління бюджетом

Постановка задачі: Розробити вебдодаток для контролю витрат та планування бюджету.

Можливості системи: категоризація витрат, прогнозування бюджету, інтеграція з банківськими сервісами, генерація фінансових звітів.

Технології: Django, FastAPI, PostgreSQL, HTML, Tailwind CSS, Pandas, HTMX.

21. Розробка онлайн-каталогу товарів з фільтрацією та пошуком

Постановка задачі: Розробити вебдодаток для перегляду товарів з можливістю пошуку та фільтрації за категоріями.

Можливості системи: створення каталогу товарів, фільтрація за параметрами, пошук товарів, відображення деталей товару.

Технології: Django, FastAPI, PostgreSQL, HTML, Tailwind CSS, HTMX, Elasticsearch.

22. Розробка вебдодатку для управління особистими завданнями та нотатками

Постановка задачі: Розробити вебдодаток для організації та ведення особистих завдань і нотаток.

Можливості системи: створення та редагування завдань, категоризація, нагадування, збереження нотаток.

Технології: Django, FastAPI, PostgreSQL, HTML, Tailwind CSS, HTMX, FullCalendar.js.

23. Розробка вебдодатку для відстеження фізичної активності користувачів

Постановка задачі: Розробити вебдодаток для моніторингу фізичної активності та прогресу користувачів.

Можливості системи: введення даних про тренування, візуалізація прогресу, встановлення цілей, аналіз активності.

Технології: Django, FastAPI, PostgreSQL, HTML, Tailwind CSS, HTMX, Chart.js.

24. Розробка інтерактивного тестового сервісу для самоперевірки знань (квізи, тести)

Постановка задачі: Розробити платформу для проходження онлайн-тестів із автоматичною перевіркою відповідей.

Можливості системи: створення тестів і квізів, автоматичне оцінювання, збереження результатів, таймер тестування.

Технології: Django, FastAPI, PostgreSQL, HTML, Tailwind CSS, HTMX, Alpine.js.

25. Розробка вебсистеми для організації подій та заходів з онлайн-реєстрацією

Постановка задачі: Розробити вебдодаток для створення подій і реєстрації учасників.

Можливості системи: створення подій, реєстрація користувачів, управління списками учасників, сповіщення.

Технології: Django, FastAPI, PostgreSQL, HTML, Tailwind CSS, HTMX, FullCalendar.js.

26. Розробка вебдодатку для управління замовленнями у сфері e-commerce

Постановка задачі: Розробити вебсистему для управління замовленнями в інтернет-магазині.

Можливості системи: додавання товарів до кошика, оформлення замовлень, відстеження статусу, інтеграція з платіжними системами.

Технології: Django, FastAPI, PostgreSQL, HTML, Tailwind CSS, HTMX, Stripe API.

27. Розробка системи керування контентом (CMS) для створення блогів та статей

Постановка задачі: Розробити вебдодаток для створення, редагування та публікації статей.

Можливості системи: створення та редагування записів, категоризація контенту, управління користувачами, коментарі.

Технології: Django, FastAPI, PostgreSQL, HTML, Tailwind CSS, HTMX, Markdown.

28. Розробка вебплатформи для спільної роботи над проектами та обміну файлами

Постановка задачі: Розробити систему для спільної роботи над файлами та управління проектами.

Можливості системи: створення проєктів, управління завданнями, обмін файлами, коментування.

Технології: Django, FastAPI, PostgreSQL, HTML, Tailwind CSS, HTMX, WebSockets.

29. Розробка вебдодатку для бронювання номерів у готелях з інтеграцією платежів

Постановка задачі: Розробити вебдодаток для бронювання номерів у готелях.

Можливості системи: пошук доступних номерів, бронювання, інтеграція з платіжними сервісами, скасування бронювання.

Технології: Django, FastAPI, PostgreSQL, HTML, Tailwind CSS, HTMX, Stripe API.

30. Розробка системи електронного голосування та опитувань

Постановка задачі: Розробити вебсистему для створення опитувань і проведення електронного голосування.

Можливості системи: створення голосувань, авторизація користувачів, підрахунок голосів, візуалізація результатів.

Технології: Django, FastAPI, PostgreSQL, HTML, Tailwind CSS, HTMX, Chart.js.

31. Розробка платформи для публікації та обговорення технічної документації

Постановка задачі: Розробити вебплатформу для розміщення, редагування та обговорення технічної документації.

Можливості системи: створення та редагування документації, підтримка версійності, коментування, категоризація документів.

Технології: Django, FastAPI, PostgreSQL, HTML, Tailwind CSS, HTMX, Markdown, WebSockets.

32. Розробка вебдодатку для управління благодійними пожертвами та волонтерською діяльністю

Постановка задачі: Розробити вебдодаток для збору пожертв і координації волонтерів.

Можливості системи: управління кампаніями, облік пожертв, інтеграція з платіжними системами, реєстрація волонтерів.

Технології: Django, FastAPI, PostgreSQL, HTML, Tailwind CSS, HTMX, Stripe API.

33. Розробка вебдодатку для автоматизації медичних записів та управління прийомами пацієнтів

Постановка задачі: Розробити вебсистему для ведення електронних медичних записів та управління розкладом прийомів.

Можливості системи: створення медичних записів, планування прийомів, нагадування, управління лікарями та пацієнтами.

Технології: Django, FastAPI, PostgreSQL, HTML, Tailwind CSS, HTMX, FullCalendar.js.

34. Розробка CRM-системи для малого та середнього бізнесу

Постановка задачі: Розробити систему для управління клієнтами, угодами та комунікацією.

Можливості системи: управління клієнтською базою, автоматизація продажів, інтеграція з email та SMS-розсилками.

Технології: Django, FastAPI, PostgreSQL, HTML, Tailwind CSS, HTMX, WebSockets.

35. Розробка вебдодатку для інтерактивного навчання з адаптивною системою рекомендацій

Постановка задачі: Розробити вебдодаток для навчання з персоналізованими рекомендаціями.

Можливості системи: проходження курсів, тестування знань, адаптивні рекомендації, відстеження прогресу.

Технології: Django, FastAPI, PostgreSQL, HTML, Tailwind CSS, HTMX, Scikit-Learn.

36. Розробка онлайн-системи для моніторингу IoT-пристроїв та аналітики даних

Постановка задачі: Розробити платформу для збору, обробки та аналізу даних з IoT-пристроїв.

Можливості системи: збір телеметричних даних, аналітика в реальному часі, віддалене керування пристроями.

Технології: Django, FastAPI, PostgreSQL, HTML, Tailwind CSS, HTMX, MQTT, WebSockets.

37. Розробка вебплатформи для обробки та візуалізації великих обсягів даних у реальному часі

Постановка задачі: Розробити систему для інтерактивної аналітики та візуалізації потокових даних.

Можливості системи: збір і обробка даних, дашборди, інтерактивні графіки, аналітика.

Технології: Django, FastAPI, PostgreSQL, HTML, Tailwind CSS, HTMX, Apache Kafka, D3.js.

38. Розробка системи керування логістичними процесами з інтеграцією GPS-моніторингу

Постановка задачі: Розробити систему для відстеження та оптимізації логістичних процесів.

Можливості системи: управління маршрутами, GPS-моніторинг, контроль доставки, генерація звітів.

Технології: Django, FastAPI, PostgreSQL, HTML, Tailwind CSS, HTMX, Google Maps API.

39. Розробка вебдодатку для автоматизованого тестування якості вебсайтів та продуктивності серверів

Постановка задачі: Розробити платформу для перевірки продуктивності та безпеки вебсайтів.

Можливості системи: тестування швидкості завантаження, перевірка SEO-показників, аналіз безпеки.

Технології: Django, FastAPI, PostgreSQL, HTML, Tailwind CSS, HTMX, Lighthouse API.

40. Розробка вебдодатку для керування проектами з використанням методології Agile.

Постановка задачі: Розробити систему для управління проектами за методологією Agile.

Можливості системи: управління спринтами, створення беклогу, відстеження прогресу, дошка Kanban.

Технології: Django, FastAPI, PostgreSQL, HTML, Tailwind CSS, HTMX, FullCalendar.js.

Електронне навчальне видання

**Сергій Олександрович Жуков
Арсен Володимирович Лосенко**

**Методичні вказівки до виконання курсових проєктів з дисципліни
«Технології веброзробки» зі спеціальності «Системний аналіз»**

Рукопис оформив *А. Лосенко*

Редактор *Н. Кравчук*

Оригінал-макет виготовлено в *PBB ВНТУ*

Підписано до видання 07.11.2025 р.

Гарнітура Times New Roman

Замовлення № P2025-162

Видавець та виготовлювач
Вінницький національний технічний університет,
Редакційно-видавничий відділ.

ВНТУ, ГНК, к. 114.

Хмельницьке шосе, 95,

м. Вінниця, 21021.

press.vntu.edu.ua;

E-mail: rvv.vntu@gmail.com

Свідоцтво суб'єкта видавничої справи
серія ДК № 3516 від 01.07.2009 р.