

Хмельницький національний університет

ДИПЛОМНИЙ ПРОЕКТ

*Методичні вказівки щодо його виконання
для студентів спеціальності
121 «Інженерія програмного забезпечення»*

*Затверджено на засіданні кафедри
інженерії програмного забезпечення.
Протокол № 8 від 21.05.2020*

Хмельницький 2020

Дипломний проект : методичні вказівки щодо його виконання
для студентів спеціальності 121 «Інженерія програмного забезпечення» /
Л. П. Бедратюк, Г. І. Радельчук, Ю. В. Форкун, О. М. Яшина. – Хмель-
ницький : ХНУ, 2020. – 77 с.

Укладачі: Бедратюк Л. П., д-р фіз.-мат. наук, проф.;
Радельчук Г. І., канд. техн. наук, доц.;
Форкун Ю. В., канд. техн. наук, доц.;
Яшина О. М., канд. техн. наук, доц.

Відповідальний
за випуск: Бедратюк Л. П., д-р фіз.-мат. наук, проф.

Редактор-коректор: Яремчук В. С.

Технічне редагування і верстка: Чопенко О. В.

Макетування здійснено редакційно-видавничим відділом Хмель-
ницького національного університету (м. Хмельницький, вул. Інститут-
ська, 7/1). Підп. до друку 09.06.2020. Зам. № 73/20, тир. 100 прим., 2020.

© ХНУ, 2020

ВСТУП

Бакалавр – академічний ступінь вищої освіти, який відповідає сьомому кваліфікаційному рівню Національної рамки кваліфікацій та присвоюється випускнику в результаті успішного завершення навчання за програмою підготовки бакалавра та проходження державної атестації.

Атестація випускників за освітнім ступенем «бакалавр» здійснюється екзаменаційною комісією (ЕК) після повного виконання ними відповідної освітньої програми з метою встановлення відповідності засвоєних здобувачами вищої освіти рівня та обсягу знань, умінь, інших компетентностей вимогам стандартів вищої освіти. Підставою для атестації особи є оцінка рівня досягнення результатів навчання та їх відповідність вимогам, передбаченим рівнем Національної рамки кваліфікацій та нормативним змістом освітньої програми підготовки фахівців за певною спеціальністю.

Дипломний проект (ДП) – це випускна кваліфікаційна робота, що виконується відповідно до освітнього стандарту та освітньо-професійної програми підготовки бакалаврів з метою публічного захисту та отримання академічного ступеня бакалавра.

Під час дипломного проектування майбутній інженер має не лише закріпити теоретичні знання, але й одержати навички практичної роботи зі спеціальності «Інженерія програмного забезпечення», набуті первинного досвіду самостійної розробки завершених програмних продуктів (ПП). Зокрема, здобувач вищої освіти має набути навичок постановки завдання із проектування програмного забезпечення (ПЗ), вибрати з існуючих сучасних засобів проектування ті, які найбільше підходять до розв'язання завдання, виконати основні етапи проектування ПЗ – розробку та аналіз вимог до ПЗ, розробку і формалізацію моделей різних видів, проектування, кодування, тестування, налагодження та документування ПЗ.

Методичні вказівки щодо виконання ДП містять вимоги до обсягу, структури, змісту, оформлення та захисту ДП з урахуванням специфіки спеціальності та майбутньої професійної діяльності бакалаврів, форми необхідних документів, приклади, зразки тощо, а також роз'яснення та рекомендації, які дозволяють студентам технічно грамотно виконати ДП та підготуватись до його захисту.

При виконанні ДП студентам слід дотримуватись вимог до його оформлення у відповідності з державними стандартами та нормативними документами Хмельницького національного університету.

Укладачі сподіваються, що ці методичні поради допоможуть продуктивній роботі студентів, дозволять уникнути зайвих витрат часу під час виконання ДП та сприятимуть підвищенню їх якості.

1 МЕТА ТА ЗАВДАННЯ ДИПЛОМНОГО ПРОЕКТУВАННЯ

ДП є кваліфікаційною роботою, що підводить підсумки навчання студента і характеризує його підготовленість до роботи за фахом «Інженерія програмного забезпечення».

Метою ДП є підтвердження студентом відповідного освітнього рівня, зокрема, його здатності розв'язувати складні спеціалізовані завдання або практичні проблеми інженерії програмного забезпечення (ІПЗ), що характеризуються комплексністю та невизначеністю умов, із застосуванням теорій та методів інформаційних технологій.

Відповідно до мети виконання ДП передбачає розв'язання студентом однієї з актуальних практичних задач у галузі ІПЗ, а також отримання конкретного прикладного результату у вигляді завершеного і функціонального придатного програмного засобу.

Основними **завданнями** дипломного проектування є такі:

– систематизація, закріплення й розширення теоретичних знань та практичних навичок за фахом;

– розвиток умінь та навичок проводити цільовий пошук інформації у друкованих виданнях і в мережі Інтернет;

– застосування студентами отриманих знань та навичок при вирішенні конкретних технічних, інженерних та виробничих завдань у галузі інформаційних технологій та ІПЗ;

– розвиток навичок самостійної роботи й оволодіння технологіями розробки завершеного ПЗ на всіх етапах життєвого циклу (ЖЦ);

– засвоєння сучасних методик проектування математичного, алгоритмічного та програмного забезпечення із застосуванням сучасних методологій, методів, інформаційних та програмних технологій;

– розвиток навичок аналітичного, графічного й літературного викладу тексту, оформлення відповідного текстового, програмного та ілюстративного матеріалу у формі проектної документації, розрахунку й обґрунтування ухвалених рішень, а також набуття досвіду публічного захисту своєї розробки;

– оцінка рівня підготовленості студента до самостійної професійної діяльності в сучасних умовах.

При виконанні ДП студенти мають підтвердити основні фахові компетенції, визначені в освітній програмі спеціальності:

– здатність ідентифікувати, класифікувати та формулювати вимоги до ПЗ;

– здатність брати участь у проектуванні ПЗ, включаючи проведення моделювання (формальний опис) його структури, поведінки та процесів функціонування;

- здатність розробляти архітектуру, модулі та компоненти ПЗ;
- здатність формулювати та забезпечувати вимоги щодо якості ПЗ відповідно до вимог замовника, технічного завдання та стандартів;
- здатність дотримуватися специфікацій, стандартів, правил і рекомендацій в професійній галузі при реалізації процесів ЖЦ ПЗ;
- здатність аналізувати, вибирати і застосовувати методи і засоби для забезпечення інформаційної безпеки;
- володіння знаннями про інформаційні моделі даних, здатність створювати ПЗ для зберігання та опрацювання даних;
- здатність застосовувати фундаментальні і міждисциплінарні знання для успішного розв’язання завдань у галузі ПЗ;
- здатність оцінювати і враховувати економічні, технологічні та інші чинники, що впливають на сферу професійної діяльності;
- здатність накопичувати, обробляти та систематизувати професійні знання щодо створення і супроводження ПЗ та визнання важливості навчання протягом всього життя;
- здатність реалізовувати фази та ітерації життєвого циклу ПЗ та інформаційних технологій на основі відповідних моделей та підходів до розробки ПЗ;
- здатність здійснювати процес інтеграції системи, застосовувати стандарти і процедури управління змінами для підтримки цілісності, загальної функціональності та надійності ПЗ;
- здатність обґрунтовано обирати та освоювати інструментарій з розробки та супроводження ПЗ;
- здатність до алгоритмічного та логічного мислення.

При захисті ДП перед екзаменаційною комісією студенти мають показати глибину знань, умінь і навичок у процесі проектування та розробки ПЗ, а також здатність до їх практичного застосування під час вирішення технічних, інженерних, виробничих завдань у різноманітних сферах людської діяльності, з оптимальним вибором методів та засобів інформаційних технологій.

За результатами захисту ДП екзаменаційна комісія визначає рівень теоретичної та практичної підготовки випускника, його готовність до самостійної роботи за фахом і приймає рішення про присвоєння йому відповідного освітнього рівня та видачі диплома бакалавра.

За всі відомості, викладені у ДП, використання фактичного матеріалу та іншої інформації під час дипломного проектування, обґрунтованість і достовірність висновків та положень несе відповідальність студент – автор дипломного проекту.

2 ОРГАНІЗАЦІЯ ДИПЛОМНОГО ПРОЕКТУВАННЯ

Навчання студентів осітнього рівня бакалавр завершується періодом виконання випускної кваліфікаційної роботи, яка подається у формі дипломного проекту.

Згідно з навчальним планом підготовки бакалаврів зі спеціальності «Інженерія програмного забезпечення» дипломне проектування виконується у три етапи:

- 1) переддипломна практика;
- 2) суто дипломне проектування;
- 3) захист дипломного проекту (в т.ч. попередній захист).

Процес виконання ДП складається з орієнтовних етапів, які наведені у таблиці 2.1.

Таблиця 2.1 – Етапи дипломного проектування

Зміст етапу	Термін виконання
1 Вибір та затвердження теми ДП; розробка завдання на ДП; складання календарного графіка виконання ДП	Січень
2 Вивчення предметної області, в якій планується використання програмного продукту; аналіз вимог до ПЗ	Січень–лютий
3 Проектування та розробка загальної архітектури і структури ПЗ, інтерфейсу користувача; вибір засобів реалізації ПЗ	Лютий–березень
4 Програмна реалізація та тестування ПЗ	Квітень
5 Написання тексту пояснювальної записки та розробка графічних матеріалів	Травень
6 Остаточне коригування ДП з урахуванням зауважень керівника; оформлення ДП як документа відповідно до вимог	Травень
7 Отримання супровідних документів (відгуку керівника, рецензії, довідки про перевірку на плагіат); нормоконтроль	Червень
8 Підготовка до захисту та захист ДП	Червень

ДП виконується студентом самостійно під контролем керівника. Керівництво дипломним проектуванням здійснюють провідні викладачі кафедри інженерії програмного забезпечення, які призначаються кафедрою і затверджуються наказом ректора.

Студент при виконанні ДП має:

- обрати і узгодити з керівником тему та отримати завдання на ДП;
- дотримуватись календарного плану виконання ДП;
- систематично відвідувати консультації керівника ДП;
- пройти процедуру попереднього захисту ДП на кафедрі;
- оформити ДП згідно з вимогами;
- відповідно до графіка захистити ДП на засіданні ЕК.

Студент **має право**:

- отримувати консультації будь-якого рівня стосовно ДП;
- користуватись всіма необхідними йому науково-методичними матеріалами, які є на кафедрі.

Керівник ДП має:

– консультувати студентів з різних питань (вибір теми ДП, розроблення плану проектування, добір літературних та інших джерел, виконання та оформлення проекту, підготовка ДП до захисту тощо);

- визначити поетапні терміни виконання роботи;
- скласти графік консультацій та дотримуватись його;
- контролювати хід та стан виконання ДП;
- періодично надавати відомості про виконання індивідуального графіка роботи закріпленими за ними студентами-дипломниками завідувачу кафедри;
- інформувати на засіданні кафедри щодо виконання студентами календарного плану;
- перевірити ДП та оцінити його;
- надати студенту допомогу при його підготовці до захисту ДП.

Керівник **має право**, у разі порушення термінів виконання ДП, низької якості чи несамостійного виконання проекту, подати підтверджену рішенням засідання кафедри пропозицію ректору про відрахування студента, як такого, що не виконує навчальний план.

Завідувач кафедри має:

- організувати методичне та інформаційне забезпечення виконання ДП;
- контролювати виконання графіку консультацій;
- розглядати на засіданнях кафедри стан виконання ДП;
- вирішувати суперечливі питання, що виникають між керівником роботи та дипломником;
- контролювати об'єктивність оцінювання ДП;
- здійснювати допуск ДП до захисту.

Завідувач кафедри **має право** не допустити дипломника до захисту ДП, якщо ДП не відповідає встановленим вимогам.

Загальний контроль за ходом дипломного проектування здійснюється випусковою кафедрою.

3 ТЕМАТИКА ДИПЛОМНОГО ПРОЕКТУВАННЯ

Тематика ДП має бути орієнтованою на дослідження та розроблення питань створення ПЗ для різноманітних предметних областей: підприємств, страхових компаній, фінансово-кредитних установ, фірм та компаній різних напрямів підприємницької діяльності, навчальних закладів, сфер послуг та розваг, документообігу, діловодства тощо.

Тема ДП має задовольняти наступним вимогам:

- відповідність сучасному стану розвитку науки, технологій, методів та засобів розробки ПЗ; актуальність; практичність;
- відповідність об'єкту діяльності бакалавра за спеціальністю «Інженерія програмного забезпечення»;
- комплексність, яка є достатньою для демонстрації теоретичних знань та практичних навичок, отриманих під час навчання.

Редакція теми ДП має бути конкретною, містити процедуру діяльності та продукт, що має бути отриманий у результаті виконання ДП. У назві теми слід уникати формулювань, що починаються зі слів «Проектування та розробка ...», «Дослідження деяких шляхів ...», «Аналіз ...» тощо, у яких не відображена у достатній мірі суть задачі та не визначений кінцевий результат дипломного проектування.

Приклад

Неправильно: *«Проектування та розробка програмного забезпечення для автоматизації споживчих бізнес-процесів ІТ-компаній»*

Правильно: *«Програмне забезпечення для автоматизації споживчих бізнес-процесів ІТ-компаній»*

У назві теми ДП не повинно бути скорочень слів (словосполучень) та абревіатур (за винятком загальноприйнятих).

ДП, який виконується одним студентом-дипломником, носить закінчений характер і має самостійне значення.

Тема ДП може реалізовуватись кількома студентами (**комплексний** ДП), але кожний ДП повинен бути закінченим у межах теми. Комплексні ДП присвячуються розробці двох або більше взаємозалежних задач, об'єднаних єдиною метою. Назва теми такого ДП складається з двох речень: *«Загальна тема. Тема проекту студента»*.

Приклад 1 (виконують два студенти)

Програмна система для медичної клініки. Серверна частина.

Програмна система для медичної клініки. Клієнтська частина.

Приклад 2 (виконують три студенти)

Комп'ютерна гра в жанрі RPG на базі Unreal Engine. Моделі об'єктів та керування персонажами.

Комп'ютерна гра в жанрі RPG на базі Unreal Engine. Звукове та музичне супроводження гри.

Комп'ютерна гра в жанрі RPG на базі Unreal Engine. Інтерфейс гри та тар-моделювання.

Для студентів, які поєднують навчання з трудовою діяльністю, доцільним є виконання ДП за матеріалами компанії, де вони працюють.

Відповідно до освітньої програми студентам рекомендуються наступні типові напрями проектування:

– розробка ПЗ для автоматизації діяльності різноманітних сфер діяльності (підприємств, організацій, компаній, фірм тощо);

– розробка ПЗ функціонування розподілених довідково-інформаційних систем;

– розробка клієнтських та/або серверних додатків для створення і підтримки сучасних веб-сайтів;

– розробка програмних засобів для мобільних пристроїв, що працюють під управлінням різних операційних систем (в тому числі ігрових додатків).

Приклади типових тем ДП

1. Програмне забезпечення для автоматизації планування та управління рекламною діяльністю підприємства.

2. Програмна система для віддаленого керування замовленнями у закладах громадського харчування.

3. Веб-орієнтована система для роботи з клієнтами хімічистки.

4. Веб-додаток для автоматизації робочого місця диспетчера кур'єрської служби доставки.

5. Довідково-інформаційна система дитячого клубу (автопарку, санаторію, туристичної фірми, салону краси тощо).

6. Мобільна версія Інтернет-магазину на платформі Android.

7. Мережевий ігровий Android-додаток «Ping-pong» на базі крос-платформеного фреймворку «Cocos2d-x».

Тематику ДП розробляє кафедра ПЗ до початку 8-го семестру. Студент також має право запропонувати власну тему з обґрунтуванням доцільності її розробки і за погодженням з керівником ДП.

Тематика дипломних проектів затверджується наказом ректора за поданням завідувача кафедри.

4 СТРУКТУРА ТА ЗМІСТ ДИПЛОМНОГО ПРОЕКТУ

4.1 Загальні вимоги

ДП включає пояснювальну записку, графічну частину та розроблений програмний засіб.

Пояснювальна записка – це текстовий документ, у якому приводиться обґрунтування, розрахунок та опис прийнятих у ДП аналітичних, проектних і програмних рішень. Вона повинна розкривати творчий задум проекту, включати моделі та методи проектування, використані алгоритми і технології вирішення задачі, опис програмної реалізації, проведених експериментів (тестування ПЗ), їх аналіз та висновки тощо і, за необхідністю, супроводжуватись таблицями, ілюстраціями, графіками, діаграмами тощо.

Записка складається з таких структурних елементів:

- титульний аркуш;
- завдання на дипломний проект;
- анотація;
- відомість документів (відомість проекту);
- зміст;
- скорочення та умовні позначки (за необхідності);
- вступ;
- основна частина;
- висновки;
- перелік джерел посилання;
- додатки.

Пояснювальна записка повинна мати загальний обсяг 65–75 с. (без додатків). Обсяг додатків не регламентується.

До **графічної частини** ДП відносяться: креслення, схеми, алгоритми, моделі тощо, які необхідні студенту під час захисту ДП. Склад графічного матеріалу та його обсяг узгоджується з керівником для кожного дипломника індивідуально у відповідності з темою ДП і повинен відповідати змісту роботи.

Графічна частина виконується як обов'язковий додаток до пояснювальної записки у вигляді презентаційних слайдів і подається за допомогою мультимедійних засобів.

За узгодженням з керівником ДП графічна частина може містити також плакати формату А1 або А2.

Зміст ілюстративного матеріалу повинен у достатній мірі відображати основні положення ДП, які виносяться на захист.

4.2 Титульний аркуш

Титульний аркуш є першою сторінкою пояснювальної записки. Він містить дані, які подають у наступній послідовності:

- назва університету, факультету та кафедри;
- назва документа (великими літерами);
- шифр документа;
- підписи розробника документа та відповідальних осіб;
- рік складання.

Шифр формується згідно з ДСТУ ГОСТ 2.201–80 (рисунок 4.1).



Рисунок 4.1 – Порядок формування шифру ДП

Приклад

ДПІПЗ.013285.01.06.ПЗ

Форма титульного аркуша подана у додатку А.

4.3 Завдання на дипломний проект

Завдання є вихідним документом на виконання ДП. Цей документ складається керівником ДП відповідно до обраної теми і видається студенту. Підписують завдання керівник ДП і студент-дипломник. Затверджує завдання завідувач кафедри.

Завдання на ДП друкується з обох сторін аркуша ф. А4 або оформлюється на стандартному бланку.

Шаблон завдання на ДП подано у додатку Б.

4.4 Анотація

В «Анотації» подають стислий опис основних аспектів ДП.

«Анотація» має містити:

- тему ДП;

- прізвище, ім'я, по батькові автора проекту;
- прізвище, ім'я, по батькові керівника проекту;
- обсяг пояснювальної записки та графічної частини, кількість рисунків, таблиць, додатків, джерел згідно з переліком посилань;
- перелік ключових слів;
- стислий опис виконаної роботи;
- підпис автора та дату подання ДП до захисту.

Ключові слова подаються великими літерами в рядок із прямим порядком слів у називному відмінку однини, розташованих за абеткою мови ДП та розділених комами.

Основний текст «Анотації» повинен характеризувати мету ДП, методи та засоби інформаційних технологій, використані для досягнення мети, містити коротку інформацію про досягнуті результати, їх практичну значимість, ступінь впровадження та область застосування, а також висновки та пропозиції щодо розвитку об'єкта розроблення.

Обсяг «Анотації» – до 500 слів. Її слід розміщувати на окремому аркуші ф. А4. Зразок «Анотації» подано у додатку В.

4.5 Відомість документів (відомість проекту)

У «Відомостях документів» вказується перелік та кількість матеріалів, що входять до складу ДП.

4.6 Зміст

До «Змісту» включають «Перелік скорочень» (за наявності), «Вступ», номери та назви всіх розділів, підрозділів та пунктів пояснювальної записки, «Висновки», «Перелік джерел посилання», додатки, з вказівкою номерів сторінок, які містять початок матеріалу відповідного структурного елемента.

4.7 Скорочення та умовні позначки

Цей структурний елемент є необов'язковим; його слід використовувати лише тоді, якщо у пояснювальній записці є більше п'яти скорочень. Його зазвичай називають «Перелік скорочень».

Переліки скорочень слід розташовувати стовпцем за абеткою. Ліворуч наводять скорочення спочатку українською мовою, а потім іншими мовами (за наявності), а праворуч – їх розшифрування.

Приклад «Переліку скорочень» наведений у додатку Г.

4.8 Вступ

У «Вступі» висвітлюють основні тенденції розвитку та стану предметної області; окреслюють проблему, на розв'язання якої спрямований ДП, та завдання, які необхідно вирішити; також дають оцінку сучасному стану конкретного завдання ПЗ, що вирішується у проекті, та його актуальність. Окрім того, слід подати обґрунтування необхідності вирішення завдання, галузь застосування та призначення розробки.

Актуальність теми – це важливість, суттєве значення, відповідність теми ДП сучасним потребам певної галузі та перспективам її розвитку, практичним завданням відповідної сфери діяльності.

Актуальність теми ДП може визначатися: об'єктивною потребою у створенні нового ПЗ; об'єктивною потребою у модернізації бізнес-процесів у конкретних предметних областях; комерційною привабливістю результатів ДП тощо.

У «Вступі» також формулюються **мета та завдання** дипломного проектування. Формулювання мети повинне логічно випливати з обґрунтування актуальності теми ДП та відображати кінцевий бажаний результат проектування. Тому у формулюванні мети повинно бути відображено, ДЛЯ чого потрібне розроблюване ПЗ, наприклад, «*Розробити програмне забезпечення ..., яке дозволяє підвищити ефективність ...*» або «*Створити інформаційну технологію ..., яка забезпечує зниження ресурсомісткості ...*» або «*Розробити веб-додаток для підтримки прийняття рішень при ...*».

Досягнення поставленої мети здійснюється шляхом її деталізації за допомогою систематизованого плану цілеспрямованих дій – **задач (завдань) проектування**.

Рекомендується формулювати задачі наступним чином: «*виконати аналіз ...*», «*встановити особливості предметної області...*», «*проаналізувати ...*», «*виявити ...*», «*визначити залежності ...*», «*розробити (архітектуру, моделі, алгоритми, ...) ...*», «*виконати програмну реалізацію ...*» «*провести апробацію ...*» тощо.

Під час визначення завдань слід пам'ятати, що жодне з них не може повторювати мету або бути ширшим за неї. Мета досягається через розв'язання завдань, а тому кожне з них повинне просувати дослідження до наміченої мети. У підсумку результат, отриманий від вирішення всіх завдань, повинен відповідати поставленій меті.

Загальна мета «Вступу» – сфокусувати увагу читача на вказаному напрямі застосування інформаційних технологій. Орієнтовний обсяг – 2–3 с.

Узагальнений зразок «Вступу» поданий у додатку Д.

4.9 Основна частина

4.9.1 Загальні вимоги

Незалежно від теми ДП в основній частині пояснювальної записки повинні бути відображені основні процеси життєвого циклу ПЗ: *аналіз, проектування, реалізація та тестування*. Ця частина повинна відображати етапи розробки ПЗ та містити відомості про предметну область (ПрО), архітектуру та структуру ПЗ, структуру бази даних (БД), обрані засоби для реалізації ПЗ, особливості реалізації та тестування ПЗ, необхідні умови та особливості застосування ПЗ тощо.

Орієнтовний загальний обсяг основної частини пояснювальної записки – 55–65 с.

Відповідно до основних процесів ЖЦ ця частина пояснювальної записки повинна включати наступні чотири розділи:

- 1 Дослідження предметної області та постановка задачі**
- 2 Проектування програмного забезпечення**
- 3 Програмна реалізація**
- 4 Тестування програмного забезпечення**

Структуризація кожного розділу здійснюється за підрозділами/пунктами/підпунктами і узгоджується з керівником ДП.

Структура та зміст розділу «**1 Дослідження предметної області та постановка задачі**» є типовими для будь-якої тематики ДП. У розділі слід проаналізувати ситуацію в організаційному, технічному та програмному забезпеченні обраної ПрО, провести аналіз літературних та Інтернет-джерел, сформулювати список практичних задач, які необхідно виконати в ході ДП, тобто зробити розгорнуту постановку задачі. Структура цього розділу може бути наступною:

1.1 Змістовний аналіз предметної області, її структурних та функціональних особливостей

1.2 Аналіз наявного програмно-технічного забезпечення предметної області

1.3 Визначення вимог до програмного забезпечення та розробка технічного завдання

У підрозділі «**1.1 Змістовний аналіз предметної області, її структурних та функціональних особливостей**» слід здійснити огляд та опис ПрО, для якої планується розробка ПЗ.

Вивчення ПрО проводиться з метою визначення проблем та невирішених питань з точки зору впровадження інформаційних технологій, автоматизації виробничих процесів, процесів опрацювання та передачі інформації тощо. На основі результатів аналізу ПрО описується проблема, що буде вирішена за допомогою майбутнього ПЗ.

Аналіз ПрО можна, за потреби, супроводжувати її модельними представленнями. Наприклад, при аналізі бізнес-процесів фірми (організації, підприємства тощо) можна використати IDEF0-моделювання (приклади IDEF0-діаграм наведені на рисунках 4.2 та 4.3).

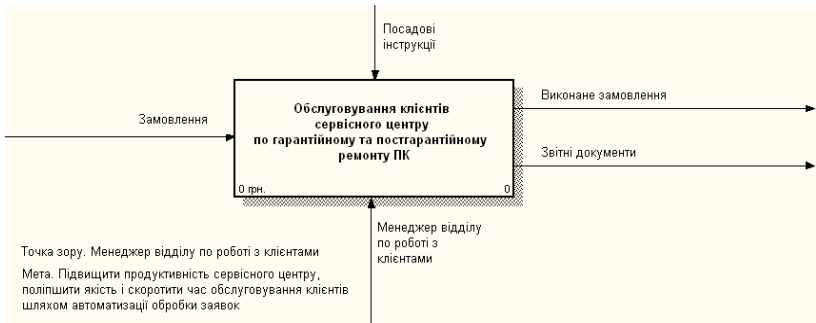


Рисунок 4.2 – Контекстна IDEF0-діаграма

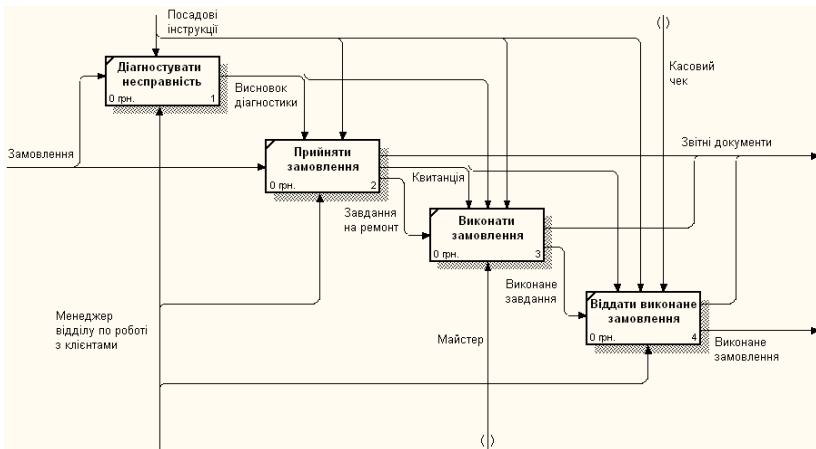


Рисунок 4.3 – IDEF0-діаграма декомпозиції першого рівня

Примітка. Методології моделювання та види моделей для унаочнення опису ПрО слід обирати індивідуально, залежно від теми ДП.

Також визначаються кінцеві користувачі ПЗ та їх інформаційні потреби. Для визначення інформаційних потреб користувачів аналізуються первинні документи та наводиться перелік їх реквізитів для зберігання у таблицях оперативної інформації, описується процес перетворення комплексу вхідної інформації у вихідну, аналізуються безпекові аспекти (ідентифікація та аугентифікація користувачів) тощо.

У підрозділі «**1.2 Аналіз наявного програмно-технічного забезпечення предметної області**» слід передбачити аналіз ПЗ, яке вже застосовується у ПрО за темою ДП. Метою такого огляду є вивчення досвіду провідних фірм-розробників ПЗ та використання їх рішень при виконанні ДП. Це сприятиме тому, що розроблюване у проекті ПЗ буде відповідати сучасним потребам ринку програмних продуктів.

У процесі аналізу для кожного аналогу слід розкрити: призначення програмного продукту; фірма-розробник; основні інтерфейсні вікна; переваги та недоліки тощо. Перелік недоліків та переваг аналогів доцільно звести у порівняльній таблиці. Знаючи ці ознаки для деякої множини аналогічного ПЗ, слід логічно вибрати з них найближчі до тих, які потрібно реалізувати у розроблюваному ПЗ.

У підрозділі «**1.3 Аналіз вимог до програмного забезпечення та розробка технічного завдання**» на основі аналізу ПрО повинні бути визначені та описані вимоги до розроблюваного ПЗ, розгляд яких слід супроводжувати побудовою низки моделей.

Примітка. На сьогодні існують багато методик, візуальних представлень, що дозволяють моделювати вимоги до ПЗ. У кожному конкретному випадку слід визначити доцільність використання тих чи інших прийомів. Аналіз вимог повинен відповідати тому, ЩО має робити ПЗ, абстрагуючись від деталей реалізації, тобто від того, ЯК воно повинно це робити.

Основним інструментом інженерії вимог до ПЗ у межах об'єктно-орієнтованої парадигми є мова UML.

У першу чергу розробляються моделі **варіантів використання** (ВВ). Діаграма ВВ відображає виконання конкретних обов'язків користувачами з використанням ПЗ. При створенні такої моделі доцільно попередньо скласти короткі описи користувачів ПЗ (акторів) та необхідних їм сервісів (ВВ). Приклади наведені у таблицях 4.1 та 4.2.

Таблиця 4.1 – Приклад опису користувачів (акторів)

Актор	Короткий опис
Менеджер	Розміщує замовлення у виробництві та стежить за їх виконанням. При необхідності може вносити оперативні корективи в описи замовлень, зняти замовлення
Диспетчер	Планує завантаження виробничих ресурсів у відповідності з чергою замовлень, що надходять від менеджерів. Передає плани (змінні завдання) в цех. Аналізує виконання планів. Вносить оперативні зміни у плани
Майстер цеху	Призначає виконавців кожної з робіт змінного завдання. Стежить за виконанням робіт, звітує перед диспетчером

Таблиця 4.2 – Приклад опису варіантів використання

Актор	Найменування ВВ	Опис ВВ
Менеджер	Реєстрація замовлення	Цей ВВ дозволяє менеджеру передавати у виробництво нові замовлення
Диспетчер	Планування нового замовлення	Диспетчер розміщує замовлення в план у «хвіст» черги
Майстер цеху	Призначення виконавців	Майстер цеху призначає виконавцям (цеховому персоналу) роботи зі змінного завдання

Кожний ВВ доцільно супроводжувати його специфікацією (описом), яка має відображати:

- передумову (початковий стан);
- дії, якими запускається ВВ;
- сценарії виконання ВВ (основний та альтернативні);
- умови завершення ВВ;
- результат, отриманий після завершення ВВ;
- післяумову (кінцевий стан).

Для формалізації подання сценарію ВВ можуть використовуватися UML-діаграми діяльності або станів. Мають бути описані основний, альтернативні та заборонені шляхи виконання ВВ, взаємодії акторів із системою та інформація, якою вони обмінюються.

Таким чином, у результаті моделювання може бути створена ієрархія діаграм, яка відображає різні аспекти розроблюваного ПЗ (структурні, функціональні, поведінкові, інформаційні тощо).

Приклади діаграм ВВ наведені на рисунках 4.4 та 4.5.

При аналізі вимог до інтерфейсу користувача слід визначити вимоги до його зовнішнього вигляду, форми взаємодії з користувачами, вимоги щодо доступу до внутрішньої функціональності ПЗ тощо.

Рекомендується у цьому підрозділі привести тільки основні, необхідні для розуміння постановки задачі, вимоги. Детальний виклад вимог до ПЗ повинен бути оформлений у вигляді документа «*Технічне завдання*» (ТЗ), який розміщується у додатках.

Приклад ТЗ (за ДСТУ ГОСТ 19.201–78) подано у додатку Е.

Розділ слід закінчити короткими висновками, наприклад:

«Проведено аналіз існуючого програмного та інформаційного забезпечення предметної області, в результаті якого визначено ...»;

«Визначені функціональні та нефункціональні вимоги до ПЗ ...».

Орієнтовний обсяг розділу 1 не повинен перевищувати 20 % від загального обсягу основної частини пояснювальної записки.

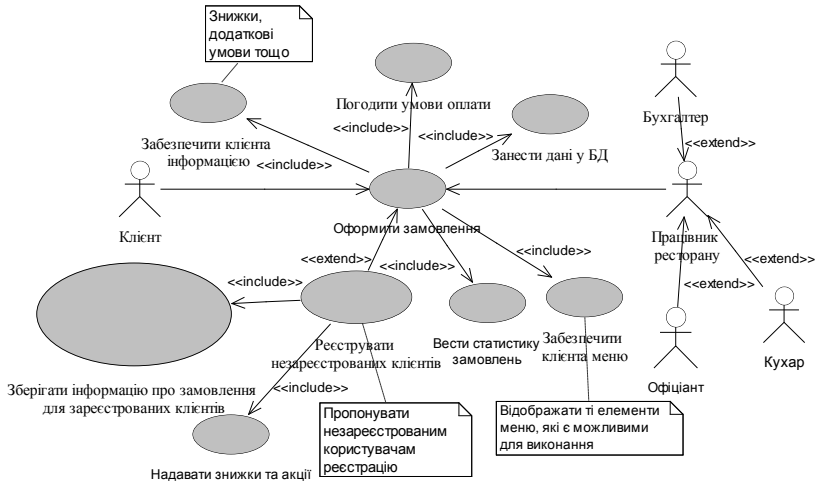


Рисунок 4.4 – Приклад діаграми ВВ для обробки замовлень ресторану

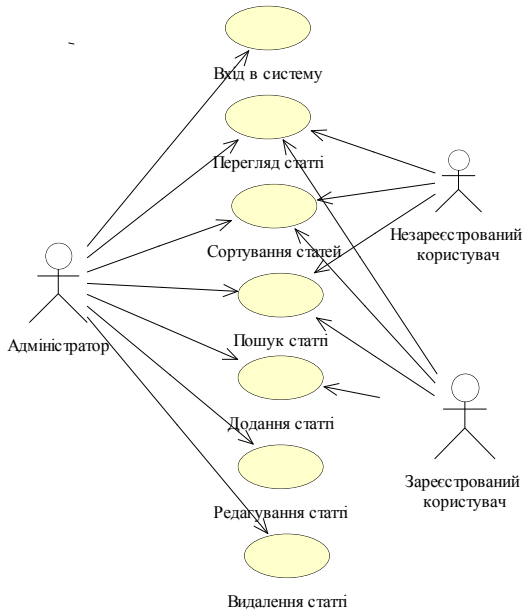


Рисунок 4.5 – Приклад діаграми ВВ для довідково-інформаційної системи

Розділ «2 Проектування програмного забезпечення»

Після аналізу ПрО та визначення вимог до ПЗ потрібно дослідити можливі способи вирішення поставлених задач. Іншими словами, якщо у процесі аналізу слід дати відповідь на питання: «ЩО має робити програмний засіб?», то на етапі проектування має бути отримана відповідь на питання: «ЯК це реалізувати?».

Під час розробки проекту обґрунтовуються проектні рішення, які дають змогу реалізувати вимоги ТЗ, забезпечити сумісність та взаємодію різних компонентів ПЗ тощо. Моделі реальної системи, отримані на етапі аналізу, у цьому розділі розширюються і коригуються таким чином, щоб вони могли бути реалізовані програмою.

Проектування ПЗ у загальному випадку виконується на таких стадіях: **ескізний проект** (архітектурний дизайн); **технічний проект** (детальне проектування). Інформаційні зв'язки цього процесу схематично зображені на рисунку 4.6.

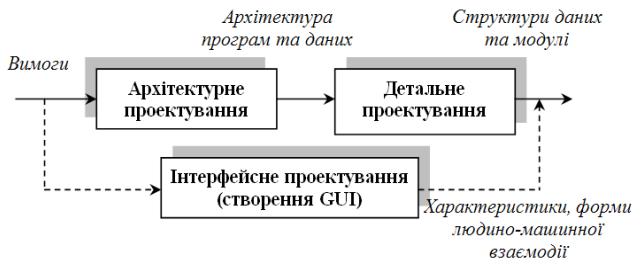


Рисунок 4.6 – Інформаційні зв'язки процесу проектування ПЗ

Основні завдання розробки архітектури ПЗ:

- виділення програмних підсистем і відображення на них зовнішніх функцій ПЗ;
- визначення способів взаємодії між підсистемами.

Зазвичай, у загальному випадку, розробляється чотири архітектурні моделі ПЗ:

- статична структурна модель, у якій представлені підсистеми/компоненти/модулі, які надалі розроблятимуться незалежно;
- динамічна модель процесів (процесна модель), у якій представлена організація процесів під час роботи ПЗ;
- інтерфейсна модель, що визначає сервіси, які надаються кожною підсистемою/компонентом через загальний інтерфейс.
- моделі відношень, у яких показані відношення між частинами системи (наприклад, потік даних між підсистемами).

Побудова архітектури ПЗ може бути заснована на деякому архітектурному патерні (шаблоні), наприклад, MVC (*Модель–Вид–Контролер*), клієнт-сервер тощо.

Залежно від типу архітектури, проект може розбиватися на рівні, що у подальшому вимагатиме додаткових описів при декомпозиції та проектуванні (наприклад, компоненти інформаційної системи по виконуваних функціях можна розділити на три рівні: рівень представлення, рівень бізнес-логіки та рівень доступу до даних).

Проектування архітектури ПЗ завершується створенням опису, в якому відображені зафіксовані проектні рішення, логічна та фізична структура системи, а також способи взаємодії об'єктів.

Примітка. Виконання стадії архітектурного проектування не є строго обов'язковим. Якщо основні проектні рішення є очевидними, то ця стадія може бути виключена із загальної послідовності робіт.

Після проектування системної структури та визначення принципів управління структурою можна зразу ж виконати декомпозицію підсистем на модулі. Фактично цю роботу можна вважати введенням у детальне проектування, яке конкретизує архітектурні рішення. Задача декомпозиції – це задача визначення внутрішнього вмісту кожної підсистеми (компонента). Результатом її вирішення є формування структури підсистеми – набору модулів та відношень їх взаємодії.

З урахуванням ухвалюваних на цьому етапі рішень проводиться подальша конкретизація функціональних специфікацій: структура програми, об'єктна модель (діаграма класів), логічна та фізична моделі БД, ключові методи та алгоритми, проект інтерфейсу користувача тощо. Для цього слід використовувати загальноприйняті засоби візуалізації: схеми, UML-діаграми, засоби запису алгоритмів (псевдокод) тощо.

Статична модель представляється діаграмами класів та специфікаціями класів. Динамічна модель представляє кооперації, які є реалізаціями ВВ відповідними діаграмами взаємодії. Логічні моделі структури даних подаються діаграмами класів або «сутність–зв'язок». Логічними модулями є підпрограми (процедури, функції).

При розробці структури ПЗ потрібно:

- визначити основні модулі та інформаційні ресурси (та аргументувати цей вибір), з яких має складатися програмний продукт;
- описати функціональне призначення основних модулів та інформаційних ресурсів, їх взаємозв'язок, а також обмін даними.

Модулі слугують фізичними «контейнерами», у яких оголошуються класи та об'єкти логічної розробки. Клас є основним «будівельним» блоком детального проектування, а UML-діаграми класів – основним засобом для представлення структури ПЗ.

Рекомендується будувати дерево модулів, які відображають структурну схему пакета: керуючі модулі (класи); модулі, які виконують службові функції; робочі модулі; інші модулі.

Розробка структури ПЗ може бути заснована на деякому шаблоні проектування.

Примітка. В деяких мовах (наприклад, C++) модуль є самостійною мовною конструкцією, яка забезпечує створення окремих проектних рішень. В деяких мовах модулів немає, а єдиною фізичною одиницею є клас.

У проекті інтерфейсу користувача(-ів) визначається:

- елементи інтерфейсу, необхідні для виконання ВВ (функцій);
- зв'язок елементів інтерфейсу один з одним;
- застосування елементів інтерфейсу у різних ВВ тощо.

Опис інтерфейсу користувача доцільно супроводжувати схемами екранних форм, систем меню, діалогів тощо.

Також у цьому розділі здійснюється аналіз технологій і методів реалізації ПЗ із зазначенням переваг та недоліків в контексті їх використання для реалізації проекту: платформи, фреймфорки, графічні «руші», системи керування базами даних (СКБД), мови програмування, мови розмітки, мови запитів до БД, існуючі засоби створення ПЗ тощо. У результаті аналізу визначається оптимальний варіант.

На закінчення розділу слід сформулювати короткі висновки.

Орієнтовний обсяг розділу 2 – до 25 % від загального обсягу основної частини пояснювальної записки.

Розділ «3 Програмна реалізація»

Реалізація ПЗ передбачає створення працездатного і функціонально придатного програмного продукту на основі розробленого проекту. У цьому розділі розглядаються питання, що безпосередньо стосуються конструювання ПЗ, а саме:

- опис модулів ПЗ (з вказуванням конкретних назв) та їх призначення, а також особливості передачі даних між ними;
- опис реалізації модулів програми;
- опис процесу створення БД* (з необхідними ілюстраціями);
- опис реалізації інтерфейсу користувача;
- інструкції для користувачів (з відповідними ілюстраціями), в яких викладається інформація про принципи та умови використання ПЗ (опис інтерфейсу користувача; послідовність дій користувачів при роботі з програмою; умови, необхідні для користування ПЗ тощо);

* За її наявності в проекті

– технічні характеристики розробленого ПЗ (вимоги до складу апаратури та операційної системи (ОС); необхідний обсяг оперативної і дискової пам'яті; спеціальні пристрої, додаткові програми, необхідні для функціонування ПЗ тощо).

При викладенні матеріалу для пояснення особливостей реалізації моделей та алгоритмів, опису модулів, класів, методів, SQL-запитів тощо рекомендується використовувати фрагменти програмного коду. Повний код програми зазвичай виноситься у додатки.

На закінчення розділу мають бути сформульовані висновки.

Орієнтовний обсяг розділу 3 – до 25 % від загального обсягу основної частини пояснювальної записки.

Розділ «4 Тестування програмного забезпечення»

Метою розділу є емпіричне дослідження, спрямоване на доведення працездатності розробленого ПЗ та його відповідності ТЗ (тобто функціональної придатності). У розділі слід визначити стратегію тестування, описати та обґрунтувати вибрані методи та методіку тестування (методи «чорної» та «білої» скриньки, функціональне та нефункціональне тестування, тестування продуктивності, навантажувальне тестування, тестування рівня даних, тестування зручності використання тощо), сформулювати вимоги до проведення експериментів, визначити обсяг кожного експерименту відповідно до функціональних специфікацій та обмежень тощо. Кінцевою метою є зіставлення очікуваних та фактичних результатів тестування.

За необхідності відзначаються рівні тестування – модульне, інтеграційне та системне тестування.

Розділ також може містити опис інструментальних засобів, за допомогою яких проводилось ***автоматизоване*** тестування.

Тест-кейси повинні мати реальний характер і бути достатніми для перевірки працездатності та правильності функціонування ПЗ. У відповідності з цим для кожного елемента ПЗ має бути складений список всіх ситуацій, які необхідно перевірити як при «правильних», так і «неправильних» вихідних даних. Слід докладно описати дані, які використовуються при тестуванні ПЗ, з вказівкою джерела їх отримання. Можуть застосовуватися дані з первинних документів, з нормативно-довідковою документацією, а також дані, генеровані автоматично.

Загальний вигляд структури тест-кейса подано на рисунку 4.7.

Примітка. Залежно від інструменту управління тест-кейсами зовнішній вигляд їх запису може трохи відрізнятися (можуть бути додані або прибрані окремі поля), але концепція залишається незмінною.

Ідентифікатор	Пріоритет	Зв'язана з тест-кейсом вимога	Модуль додатку	Підмодуль додатку	Вихідні дані, необхідні для виконання тест-кейсу	Очікуваний результат по кожному кроку тест-кейсу
UG_U1.12	A	R97	Галерея	Завантаження файла	<p><i>Підготовка: створити непорожній файл з ім'ям ...</i></p> <ol style="list-style-type: none"> 1. Натиснути кнопку «Завантажити картинку». 2. Натиснути кнопку «Вибрати». 3. Вибрати зі списку підготовлений файл. 4. Натиснути кнопку ОК. 5. Натиснути кнопку «Додати в Галерею». 	<ol style="list-style-type: none"> 1. З'являється вікно завантаження картинки. 2. З'являється діалогове вікно браузера вибору файла для завантаження. 3. Ім'я вибраного файла відображається у полі «Файл». 4. Діалогове вікно файла закривається, у полі «Файл» з'являється повне ім'я файла. 5. Вибраний файл з'являється у списку файлів Галереї.

Рисунок 4.7 – Загальний вигляд структури тест-кейса

В очікуваних результатах завжди описується коректна робота програми. При цьому коректна робота програми цілком може передбачати відображення повідомлень про неправильні дії користувача або про деякі критичні ситуації. Так, повідомлення «Неможливо зберегти файл за вказаним шляхом: на цільовому носії недостатньо вільного місця» – це не помилка програми, а її правильна робота. Помилкою програми (в цій же ситуації) була б відсутність такого повідомлення.

Зазвичай тест-кейси об'єднуються у тестові набори. До окремого підвиду наборів тест-кейсів можна віднести сценарії використання.

Примітка. Не варто описувати всі тест-кейси; достатньо навести декілька основних прикладів.

У розділі слід також зазначити порядок тестування, навести процедури перевірки із зазначенням очікуваних при правильній роботі ПЗ результатів, а також опис основних результатів, отриманих у процесі тестування ПЗ.

Якщо програма має працювати на різному обладнанні або в різних операційних середовищах (на різних платформах), то вона має бути перевірена на всіх цих платформах.

Після виконання тест-кейсів створюється звіт про результати тестування, який містить інформацію по кожному виконаному тест-кейсу та результат його виконання – успіх чи невдачу. Запис про результат проходження кожного тест-кейса містить таку інформацію:

- ідентифікатор тест-кейса;
- короткий опис тест-кейса;
- перерахування всіх вхідних значень тест-кейса;
- перерахування всіх очікуваних вихідних значень тест-кейса.
- перерахування всіх реальних вихідних значень тест-кейса;

– для кожної пари «очікуване і реальне вихідне значення» – інформацію про збіг або розбіжності цих значень;
 – повідомлення про те, пройдений чи не пройдений тест-кейс.
 Для більшої наочності, підсумкові дані тестування доцільно оформити у вигляді таблиць і графіків.

Приклад сценарію тестування наведений у таблиці 4.3.

Таблиця 4.3 – Приклад сценарію тестування (фрагмент)

Дія	Реакція програми	Результат
1. Клацнути на кнопці «System» та вибрати пункт меню «System Management Applet»	З'явиться вікно введення логіну та пароля	Правильно
2. У вікні введення ввести ім'я користувача guest1 та пароль guest . Далі натиснути кнопку «Login»	Вікно «System Management Applet» повинно закритися	<i>Неправильно.</i> Вікно має назву «System Management Application»
3. Завершити сеанс роботи з аплетом, натиснувши кнопку «Logout»	Вікно «System Management Applet» повинно закритися	Правильно

Примітка. За результатами тестування можна сформувати документ «Акт тестування» та включити його у додатки.

У кінці розділу подаються короткі висновки про результати тестування, а також про ступінь працездатності ПЗ та його відповідність (невідповідність, часткову відповідність) вимогам ТЗ, наприклад:

«В результаті проведених експериментів ... виявлено, що ...»;

«База функціональність реалізована у відповідності з вимогами до ПЗ і є повністю/частково працездатною»;

«Існують некритичні проблеми, пов'язані з ...»;

Орієнтовний обсяг розділу 4 – 20 % від загального обсягу основної частини пояснювальної записки.

4.9.2 Деталізація розділів основної частини для типової тематики дипломного проектування

Структуру та зміст розділів основної частини розглянемо на прикладах типової тематики ДП.

Приклад 1. Програмне забезпечення для автоматизації (бізнес-процесів, управління, діяльності фірми чи організації тощо)

Будь-яка організація – це сукупність взаємодіючих елементів (підрозділів, процесів тощо), кожний з яких може мати свою структуру.

Елементи пов'язані між собою функціонально, тобто вони виконують окремі види робіт у рамках єдиного бізнес-процесу, обмінюючись інформацією. Окрім того, ці елементи взаємодіють із зовнішніми системами. Ця ситуація справедлива практично для всіх організацій, яким би видом діяльності вони не займалися. Такий загальний погляд дозволяє сформулювати основні принципи побудови програмних систем, які виявляються у ході аналізу Про.

Структура та зміст розділів основної частини пояснювальної записки може бути наступною.

1 Дослідження предметної області та постановка задачі

У розділі характеризується місія об'єкта автоматизації (підприємства, фірми, процесу тощо), наводиться схема організаційної структури об'єкта, характеризуються бізнес-процеси, існуючі програмні рішення тощо. На основі аналізу описується проблема, що буде вирішена за допомогою розроблюваного ПЗ, визначаються задачі, які необхідно автоматизувати. При аналізі вимог до ПЗ слід подати характеристику функціональних (бізнес-вимоги, вимоги до користувачів, системні вимоги тощо) та нефункціональних вимог (бізнес-правила, атрибути якості, зовнішній інтерфейс, обмеження тощо).

2 Проектування програмного забезпечення

У розділі викладаються питання, пов'язані з вибором та розробкою архітектури ПЗ, проводиться декомпозиція проекту, здійснюється опис інтерфейсів об'єктної моделі, інтерфейсу користувача тощо.

Цей розділ рекомендовано структурувати наступним чином.

2.1 Вибір типу архітектури та шаблонів проектування

Тут детально описується вибір типу архітектури та зразків проектування. Для цього виділяється декілька архітектур та здійснюється їх детальний аналіз; на основі проведеного аналізу обирається архітектура, яка найкраще відповідає вимогам. Вибір здійснюють на основі порівнянь та виключень. В залежності від типу архітектури та складності ПЗ проект може бути розбитий на відповідні рівні.

2.2 Опис декомпозиції

Тут здійснюється декомпозиція проекту, що, в залежності від типу вибраної архітектури, відображається в описі загальних декомпозицій, модульної декомпозиції, декомпозиції на паралельні процеси, декомпозиції даних, декомпозиції моделі переходів станів, декомпозиції ВВ тощо. У разі поділу додатка на декілька рівнів опис декомпозиції слід виконувати для кожного рівня окремо.

2.3 Опис залежностей

Описуються залежності між варіантами декомпозицій, представленої в п. 2.2. Такий опис може складатися з наступних блоків: блок міжмодульної залежності, блок міжпроцесних залежностей, блок

залежності всередині даних, блок залежності між станами, рівнями тощо. При наявності рівнів міжрівневі залежності слід описати окремо.

2.4 Опис інтерфейсів

Тут, з врахуванням розробленої структури ПЗ, описуються інтерфейси об'єктної моделі, де поряд з класами одного рівня можуть описуватися класи іншого рівня (за наявності). Цей підрозділ можна розділити на опис міжмодульних інтерфейсів, у яких описують інтерфейси наявних пакетів, та опис інтерфейсів процесів (у разі, якщо всі класи в усіх пакетах є відкритими, то інтерфейси будуть складатись з усіх класів та методів).

2.5 Аналіз та вибір технологій і методів реалізації додатка

У цьому підрозділі здійснюється огляд, аналіз та обґрунтування вибору технологій та засобів для розробки ПЗ.

3 Програмна реалізація

У розділі розглядаються питання, що стосуються діяльності, яка слідує за вибором архітектури. Основною метою розділу є опис та реалізація спроектованого додатку. Розроблена архітектура повинна бути приведена до детального проекту, реалізація якого пов'язана з реалізацією ПЗ. Реалізація ПЗ передбачає створення, на основі проекту, працездатного і функціонально придатного програмного засобу.

Цей розділ може включати такі підрозділи.

3.1 Детальне проектування модулів

У підрозділі наводяться етапи детального проектування модулів ПЗ із вказівкою їх назв та структурних елементів, які в них входять (слід подавати та описувати кожний структурний елемент). Опис структурних елементів має опиратися на діаграми класів, послідовності, переходів станів та ВВ. Окрім того, в деталізацію архітектурних рішень доцільно включати блок-схеми алгоритмів та псевдокод, а також опис інтерфейсу користувача.

3.2 Програмна реалізація модулів

У цьому підрозділі здійснюється опис реалізації усіх модулів ПЗ. При викладенні матеріалу для пояснення особливостей реалізації моделей та алгоритмів, модулів, класів, методів, інтерфейсів (у тому числі інтерфейсу користувача) тощо рекомендується використовувати фрагменти програмного коду.

3.3 Детальне проектування даних

Подається опис структур даних, які не використовуються при описі класів у попередніх підрозділах. Подаються діаграми потоків даних, опис таблиць даних та їх зв'язки, інфологічна та фізична моделі БД. За умови, що у вибраному архітектурному рішенні не виявиться окремих структур даних, у цьому підрозділі подається інформація про дані, які використовуються у модулях, та їх обробка.

3.4 Розробка бази даних

Тут слід навести опис процесу створення БД (з необхідними ілюстраціями) та її підключення до модулів ПЗ. Цей підрозділ має також містити описи збережених процедур, функцій і тригерів БД (якщо вони були розроблені) з приведенням вихідного коду в додатках. Якщо створена модель БД повинна бути спільною для багатьох додатків чи користувачів, то для цього слід використовувати сучасну СКБД, яка підтримує мережеву обробку «клієнт-сервер» та забезпечує ефективний доступ користувачів до даних БД.

3.5 Керівництво користувача

У підрозділі наводяться дані, які містять основну інформацію про принципи та умови використання ПЗ: зазначається інформація, необхідна для розуміння функцій ПЗ та його експлуатації; вказуються умови, необхідні для використання додатку; подається опис використання ПЗ для різних типів користувачів (зокрема, вказується послідовність дій користувача, які забезпечують завантаження, запуск, процес роботи з ПЗ та завершенням його роботи); наводяться (за необхідності) приклади користування додатком з відповідними ілюстраціями.

3.6 Вимоги до технічних та програмних засобів

Тут вказуються технічні характеристики розробленого ПЗ та основні вимоги до апаратно-програмних засобів для його повноцінного і безперебійного функціонування. Вказується мінімальний склад та коротка характеристика системного ПЗ, додаткового ПЗ, драйверів, спеціальних пристроїв тощо.

4 Тестування програмного забезпечення

У розділі здійснюється верифікація та валідація ПЗ. Розділ має містити обґрунтування вибору методів тестування, розробку тестових наборів даних, а також аналіз результатів тестування. Кожне з питань повинно закінчуватись короткими узагальнюючими висновками. У розділ також доцільно включити рекомендації чи пропозиції щодо вдосконалення визначених проблемних аспектів.

Цей розділ може містити такі підрозділи.

4.1 Вибір та обґрунтування методів тестування додатку

У цьому підрозділі проводиться огляд та аналізуються методи і засоби тестування, які найкраще підходять до розробленого ПЗ.

4.2 Доведення працездатності програми

У підрозділі подаються тести, за допомогою яких здійснюється перевірка компонентів додатка та правильності роботи ПЗ в цілому.

4.3 Валідація та верифікація програмного забезпечення

У цьому підрозділі здійснюється опис перевірки на валідність окремих модулів ПЗ та додатка в цілому. Верифікація здійснюється

шляхом перевірки відповідності функцій, реалізованих у готовому ПЗ, до вимог, визначених у ТЗ.

4.4 Аналіз результатів тестування

У підрозділі подається узагальнююча інформація, що стосується аналізу результатів тестування ПЗ. Ця інформація може подаватись у вигляді таблиць, графіків, діаграм, карт-схем тощо.

Приклад 2. Довідково-інформаційна система

Довідково-інформаційна система (ДІС) – це сховище інформації, яке включає засоби введення, зберігання, захисту, пошуку та подання повідомлень. ДІС вирішують задачі по забезпеченню користувача максимально точною (релевантною) інформацією за темою, яка його цікавить. Вони мають низку переваг і можливостей, зокрема:

- можливість компактно зберігати великі обсяги інформації;
- можливість структуровано відображати збережену інформацію;
- можливість швидкого пошуку потрібних даних тощо.

Інтерфейс ДІС із зовнішнім світом, а саме інформаційні потоки між системою й зовнішніми сутностями, з якими вона повинна бути зв'язана, складається з інформаційних потоків між користувачами і самою ДІС, яка обробляє та зберігає інформацію в БД.

Структура та зміст розділів основної частини пояснювальної записки може бути наступною.

1 Дослідження предметної області та постановка задачі

В цьому розділі, поряд із загальними вимогами до нього, слід приділити значну увагу аналізу та визначенню видів і змісту інформаційного забезпечення розроблюваної ДІС. Особлива роль у цьому відводиться класифікації та кодуванню інформації, що забезпечує взаємний обмін інформацією між користувачем та ДІС. Подається також опис користувачів системи (наприклад, адміністратор, зареєстрований користувач, незареєстрований користувач тощо). Для унаочнення інформаційних потоків можна використовувати діаграми потоків даних.

2 Проектування довідково-інформаційної системи

Основу будь-якої ДІС становить БД – сукупність упорядкованої інформації, що використовується при функціонуванні ДІС і має на меті забезпечити взаємообмін інформацією між структурними одиницями ДІС. З огляду на це і здійснюється проектування ДІС. Особлива увага також приділяється логіці роботи додатку, організації взаємодії між програмою та БД, між користувачами та ДІС.

Цей розділ рекомендовано структурувати наступним чином.

2.1 Проектування архітектури та структури системи

У підрозділі наводиться опис обраної архітектури ДІС («файл-сервер», «клієнт-сервер») та опис детального проектування модулів ПЗ

із вказівкою їх назв та структурних елементів, які в них входять (слід подавати та описувати кожний структурний елемент). Опис структурних елементів має опиратися на діаграми класів, послідовності, переходів станів та ВВ. Окрім того, в деталізацію архітектурних рішень доцільно включати блок-схеми алгоритмів та псевдокод. Схема взаємозв'язку модулів (класів) та інформаційних ресурсів має відображати взаємозв'язок програмного та інформаційного забезпечення комплексу задач, і може бути представлена декількома схемами, кожна з яких відповідає певному режиму.

Функціональна структура ДІС має орієнтуватися на ті інформаційні потреби кінцевих користувачів, які змінюються в умовах ринку, та відображати зміст і специфіку необхідних функцій.

2.2 Проектування логічної моделі бази даних

Для створення логічної моделі БД у першу чергу необхідно уніфікувати та стандартизувати її складові. На цьому етапі потрібно:

- визначити перелік сутностей, інформація про які зберігатиметься у БД, навести їх опис;
- визначити перелік атрибутів сутностей, навести їх опис;
- визначити зв'язки між сутностями БД, навести їх опис;
- визначити типи зв'язків та необхідні обмеження;
- визначити первинні ключі для кожного об'єкта БД;
- виконати нормалізацію даних (до третьої нормальної форми).

Логічну модель БД доцільно будувати на основі діаграм класів (можна також використовувати діаграми «сутність-зв'язок»). В діаграмі класів повинні бути відображені сутності, атрибути та асоціації, які є необхідними для опису структури БД.

2.3 Проектування інтерфейсу користувача

У цьому підрозділі здійснюється опис проектування структури системи управління змістом ДІС, схема роботи інтерфейсу (у тому числі логічна організація інформації на сторінках ресурсу) для користувачів з різними правами доступу. Якщо, згідно з ТЗ, ДІС є веб-орієнтованою системою, то розробляється логічна структура веб-сайту, подається опис системи навігації, створюються прототипи необхідних веб-сторінок та попередній дизайн тощо. Вказана інформація може подаватись у вигляді рисунків, діаграм, блок-схем тощо.

2.4 Аналіз та вибір технологій і методів реалізації системи

У підрозділі виконується аналіз та обґрунтування вибору технологій і засобів розробки ДІС. Здійснюється також короткий аналіз мов програмування, вибір СКБД з необхідним обґрунтуванням. Якщо створена модель БД повинна бути спільною для багатьох додатків чи користувачів, то слід використовувати сучасну СКБД, яка підтримує

мережеву обробку «клієнт-сервер» та забезпечує ефективний доступ користувачів до даних БД (Microsoft SQL Server, MySQL тощо).

3 Програмна реалізація

У розділі детально описується програмна реалізація ДІС, тобто подається опис програмного коду модулів системи та структури її компонентів. Етапами реалізації ДІС є опис створення БД, опис складу, структури, змісту та функцій розробленого ПЗ та взаємне функціонування (в тому числі на основі веб-інтерфейсу). При викладенні матеріалу для пояснення особливостей реалізації алгоритмів, опису функцій, процедур, класів, методів, SQL-запитів слід подавати фрагменти програмного коду. При реалізації ПЗ слід передбачати видачу повідомлень про помилки користувача. Цей розділ може включати такі підрозділи.

3.1 Реалізація бази даних

У підрозділі викладаються питання, пов'язані зі створенням БД засобами обраної СКБД (включаючи управління версіями, оновлення БД тощо). Спочатку, на основі побудованої логічної моделі БД, описують цю модель у термінах обраної СКБД, тобто будується фізична модель БД. Далі наводяться пояснення зі створення і заповнення таблиць БД, подаються описи спроектованих тригерів та процедур тощо. При описі зазначених елементів вказують, які елементи реалізуються засобами СКБД, а які – засобами обраної мови програмування.

3.2 Реалізація модулів системи

У цьому підрозділі здійснюється опис програмної реалізації усіх компонентів ДІС, а також інтерфейсу користувачів (з різними правами доступу). При викладенні матеріалу для пояснення особливостей реалізації модулів, класів, методів, алгоритмів тощо рекомендується використовувати фрагменти програмного коду. У підрозділі описується також технологія взаємодії додатку, СКБД та БД.

3.3 Інструкція користувача

Тут викладаються питання, що містять опис умов та принципів використання ДІС. Опис роботи ДІС повинен містити наступні дані: загальні відомості про розроблену систему; її функціональне призначення; вхідні та вихідні дані; копії вікон інтерфейсу, результатів роботи програми тощо, а також послідовність дій користувача, які забезпечують завантаження, виконання та завершення програми. Загалом, повинна бути вказана інформація, достатня для розуміння основних функцій системи та її експлуатації.

3.4 Вимоги до апаратно-програмних засобів

У підрозділі вказується мінімально необхідний для нормальної роботи ДІС склад та характеристика апаратних засобів, вимоги до системного ПЗ, інформація про додаткові пристрої тощо (ці характеристики описуються окремо для серверної та клієнтської частин додатку).

3.5 Розгортання та встановлення системи*

В розділі описується процес налагодження серверної та клієнтської частин ДІС. Окремо описуються серверна частина, в якій подається (за потреби) встановлення ОС, встановлення та налаштування СКБД, налагодження компонентів (як в ОС, так і в СКБД) тощо. Описується сам процес перенесення БД. Для клієнтської частини вказується, яким чином вона встановлюється, налагоджується, здійснює доступ до серверної частини тощо.

4 Налагодження та тестування системи

Описується процес налагодження та тестування ДІС. Досягнуті результати демонструються шляхом показу можливих станів програми, введенням коректних та некоректних даних із відображенням відповідних повідомлень, які передбачені при розробці додатку. Результати подаються у вигляді копій екрана з поясненнями.

Цей розділ розділ може містити такі підрозділи.

4.1 Вибір та обґрунтування методів тестування системи

У цьому підрозділі проводиться огляд та аналізуються методи і засоби тестування, які застосовні до розробленої системи.

4.2 Валідація та верифікація системи

Для оцінювання практичних аспектів роботи розробленої ДІС застосовується верифікація та валідація. Під час верифікації та валідації програма перевіряється на правильність реалізації алгоритмів, правильність роботи ДІС для всіх наборів вхідних даних (перевірка форматів введення даних, перевірка типів даних, перевірка приналежності даних до певних діапазонів тощо) та пошук і видалення дефектів, а також перевірка реалізованих функцій на відповідність ТЗ.

4.4 Аналіз результатів тестування системи

У підрозділі подається узагальнююча інформація, що стосується аналізу результатів тестування ДІС. Ця інформація може подаватись у вигляді таблиць, графіків, діаграм, карт-схем тощо.

Приклад 3. Веб-додаток

Веб-додаток – клієнт-серверний додаток, в якому клієнт взаємодіє з веб-сервером за допомогою браузера. Логіка веб-дodatка розподілена між сервером та клієнтом; зберігання даних здійснюється переважно на сервері; обмін інформацією відбувається по мережі. Браузер може бути реалізацією «тонких клієнтів» – логіка додатка зосереджується на сервері, а функція браузера полягає переважно у відображенні інформації, завантаженої мережею з сервера, та передачі назад даних

*Якщо ДІС має клієнт-серверну архітектуру

користувача. Однією з переваг такого підходу є той факт, що клієнти не залежать від конкретної ОС користувача.

Структура та зміст розділів основної частини пояснювальної записки може бути наступною.

1 Дослідження предметної області та постановка задачі

В цьому розділі здійснюється опис предметної області (призначення веб-додатка, ролей, можливих операцій), аналіз існуючих подібних розробок відповідно до сучасних потреб (зокрема, бізнесу, економіки тощо). Також описуються особливості та обмеження для розроблюваного додатка та подається обґрунтування доцільності розробки. Результатом цього розділу має бути розроблене ТЗ, де вказуються всі вимоги, обмеження, функціонал та призначення.

2 Проектування веб-додатка

У цьому розділі проводиться аналіз та вибір архітектури розроблюваного додатку, проектування структури БД, проектування серверної та/або клієнтської частин, проектування інтерфейсу користувача. Також подається опис, схематичні макети (або макети .psd) структури веб-додатку, опис системи навігації, створюються прототипи необхідних веб-сторінок та попередній дизайн.

Цей розділ рекомендовано структурувати наступним чином.

2.1 Аналіз та вибір архітектури веб-додатка

У цьому підрозділі здійснюється аналітичний опис та обґрунтування вибору архітектури розроблюваного ПЗ. Оскільки мова йде про онлайн-додаток, що передбачає інтерактив, то, як правило, обирається клієнт-серверна архітектура.

2.2 Опис структури даних та моделі бази даних

В цьому підрозділі здійснюється розгляд схеми БД та подання всіх таблиць із зазначенням типів полів і поясненнями, а також відношень між таблицями. Дану інформацію потрібно подавати у вигляді нормалізованої схеми БД. Змодельовану БД доцільно візуалізувати за допомогою ER-діаграми (або діаграми класів).

2.3 Проектування серверної частини веб-додатка

У цьому підрозділі здійснюється налаштування серверу для реалізації веб-додатка, зокрема розгляд методів створення форм для відправки даних на сервер та реалізація сценаріїв, які обробляють отримані дані (оскільки основна задача серверної частини додатка – це отримання та повна обробка даних, що надходять від клієнта).

2.4 Проектування клієнтської частини веб-додатка

У цьому підрозділі здійснюється опис проектування клієнтської частини додатку, де вказується структура системи управління змістом, схема роботи інтерфейсу (у тому числі логічна організація інформації на сторінках ресурсу). Вказана інформація може подаватись у вигляді рисунків, діаграм, блок-схем тощо.

2.5 Створення макета веб-додатка та дизайн

У цьому підрозділі здійснюється проектування структури веб-додатка, де вказується організація даних, ієрархія матеріалів, необхідна для представлення їх кінцевому користувачу. Процес створення структури додатка можна розділити на два етапи: 1) структуризація інформації; 2) візуальне представлення структури. Структуризація інформації полягає у необхідності класифікації та групування розрізних матеріалів, об'єднанні їх в категорії або рубрики, присвоєнні їм зрозумілих для користувачів назв. Також приділяється увага кольірному, графічному та стильовому рішенню.

2.6 Аналіз та вибір технологій і методів реалізації веб-додатка

У цьому підрозділі здійснюється огляд, аналіз та обґрунтування вибору технологій та засобів для розробки веб-додатка: мова розмітки HTML; складові DHTML; каскадні таблиці стилів (CSS); об'єктна модель документа (DOM); методи створення JavaScript-сценаріїв у складі веб-сторінок; керування елементами сторінок на основі DOM; організація взаємодії з користувачем на основі подій; засоби бібліотеки jQuery для організації взаємодії JavaScript та HTML; мова програмування PHP та відповідні фреймворки; методи взаємодії веб-сценаріїв та СКБД, мова запитів до баз даних MySQL; підхід до побудови інтерактивних інтерфейсів користувача з використанням технології Ajax; протоколи та формати обміну даними у мережі Інтернет та методи їх обробки тощо.

Примітка. Найчастіше серверна частина веб-додатка програмується засобами мов PHP чи JavaScript, однак розробник може обирати й інші засоби за вподобанням та рівнем володіння.

3 Програмна реалізація

У розділі розглядаються питання, що безпосередньо стосуються діяльності, метою якої є підготовка проекту до реалізації та реалізація проекту. Опис процесу розробки веб-додатка та його компонентів доцільно супроводжувати відповідними «скріншотами» та фрагментами програмного коду. Також слід навести керівництва для адміністратора та користувача. Цей розділ може включати наступні підрозділи.

3.1 Розробка бази даних

Тут здійснюється опис розробки БД, підключення її до інших модулів додатку. Цей підрозділ включає створення БД, керування версіями, оновлення БД та інші налаштування.

3.2 Розробка програмних модулів

У цьому підрозділі здійснюється опис реалізації усіх запланованих модулів додатку, зокрема модулів авторизації, реєстрації, системи оплати (у разі розробки, наприклад, Інтернет-магазину) тощо.

3.3 Керівництво користувача

У цьому підрозділі здійснюється опис покрокового користування веб-додатком для різних категорій користувачів: адміністратора, зареєстрованого користувача, гостя тощо.

3.4 Технічні характеристики веб-додатка

У цьому підрозділі вказуються технічні характеристики розробленого додатка (як його серверної частини, так і клієнтської) та основні вимоги для його правильного і безперебійного функціонування.

3.5 Завантаження веб-додатка на хостинг

У підрозділі здійснюється опис розміщення веб-додатка на хостингу. Для ДП достатньо зареєструватися на безкоштовному хостингу (наприклад, <https://000webhost.com/>), розмістити на ньому файли та коротко описати процес розміщення додатка на хостингу.

4 Тестування веб-додатка

У розділі здійснюється верифікація і валідація веб-додатка. Розділ має містити обґрунтування вибору методів тестування, розробку тестових наборів даних, а також аналіз результатів тестування. Кожне з питань має закінчуватись короткими узагальнюючими висновками, які висловлюють власну думку автора. У розділ доцільно включити рекомендації чи пропозиції щодо вдосконалення визначених проблемних аспектів предмета дослідження. Розділ може містити такі підрозділи:

4.1 Вибір та обґрунтування методів тестування веб-додатка

У цьому підрозділі проводиться огляд та аналізуються методи і засоби тестування, що найкраще підходять до розроблюваного ПЗ.

4.2 Перевірка на помилки за допомогою unit-тестів

Подаються тести, за допомогою яких здійснюється перевірка компонентів додатку. Можна навести приклади запущених unit-тестів на виконання з отриманими результатами, у яких відображається, чи співпадає очікуваний результат з отриманим.

4.3 Валідація та верифікація веб-додатка

У цьому підрозділі здійснюється опис перевірки на валідність окремих модулів веб-додатка і додатка в цілому. Валідація може здійснюватись за допомогою валідаторів (наприклад, <https://validator.w3.org/>). Верифікація здійснюється шляхом перевірки відповідності функцій, реалізованих у готовому додатку, до вимог, визначених у ТЗ.

4.4 Аналіз результатів тестування веб-додатка

Подається узагальнююча інформація щодо аналізу результатів тестування веб-додатка. Вона може бути представлена у вигляді таблиць, графіків, діаграм, карт-схем тощо.

Приклад 4. Мобільний додаток

Розробка ПЗ для мобільних пристроїв потребує врахування їх обмежень та можливостей. Мобільні пристрої працюють на акумуля-

торі та мають менш потужні процесори, ніж ПК, а також мають більше функцій (наприклад, геолокація, наявність камери). Розробникам також доводиться враховувати розміри дисплею, різні технічні характеристики та конфігурації обладнання через сильну конкуренцію мобільного ПЗ та зміни в кожній платформі.

Структуризація та зміст розділів основної частини пояснювальної записки може бути наступною.

1 Дослідження предметної області та постановка задачі

В цьому розділі проводиться аналіз ПрО із виокремленням її меж. Також здійснюється опис огляду існуючих мобільних додатків (близьких до теми ДП), розроблених *за останні два роки*, та обґрунтування доцільності розробки. Результатом аналізу є складання ТЗ, а також опис особливостей та обмежень додатка.

2 Проектування мобільного додатка

У розділі обирається архітектура додатку, створюються та подаються карти чи моделі, що наочно демонструють усі функції додатка. Також здійснюється створення прототипів, які відображають всі екрани додатка та схеми переходів по них. Варіант подачі схем може бути довільним. Залежно від потреб прототипи можуть бути статичними або інтерактивними. Також слід розробити дизайн всіх екранів додатка із промальовуванням різних станів для всіх сценаріїв використання.

Цей розділ рекомендовано структурувати наступним чином.

2.1 Архітектура та функціональна структура додатка

У підрозділі здійснюється аналіз та вибір архітектури розроблюваного додатка. Вказуються переваги та недоліки обраного архітектурного рішення, схематично подається його функціональна структура.

2.2 Проектування структури бази даних

За наявності БД у мобільному додатку в цьому підрозділі має бути розроблена нормалізована схема БД відповідно до алгоритму нормалізації на декілька відношень на базі функціональних залежностей.

2.3 Проектування інтерфейсу користувача

В цьому підрозділі здійснюється опис проектування інтерфейсу користувача. Тут продумується дизайн інтерфейсу із врахуванням особливостей структури діалогу, який буде відбуватися між користувачем і мобільним пристроєм. Цей етап визначає структуру високого рівня макетів екрана, а також потік, поведінку та організацію продукту. Всі елементи графічного інтерфейсу піддаються юзабіліті-дослідженню, щоб упевнитися, що прийняті дизайн-рішення є ергономічними та дозволяють користувачам ефективно вирішувати свої завдання. Опис спроектованого інтерфейсу подається у вигляді структурних схем, макетів тощо.

2.4 Розробка алгоритму роботи мобільного додатка

У цьому підрозділі подаються інструкції, що описують порядок дій виконавця для досягнення результату вирішення задачі за ви-

значене число дій. При розробці алгоритму слід визначити повний набір вихідних даних завдання (початковий стан об'єкта); мету створення алгоритму (кінцевий стан об'єкта); систему команд виконавця (набір команд, які виконавець розуміє і може виконати). Розроблений алгоритм можна подати у вигляді блок-схеми, моделі тощо.

2.5 Створення прототипу мобільного додатка

В цьому підрозділі здійснюється прототипування головних частин додатка за допомогою сучасних засобів. На основі готових прототипів та задокументованих рішень будується схема роботи додатка.

2.6 Аналіз та вибір технологій і методів реалізації додатка

У підрозділі проводиться аналіз сучасних засобів та технологій для виконання поставленого завдання. Здійснюється вибір мови програмування (в т.ч. необхідних фреймворків) залежно від операційної системи та відповідно до потреб, актуальності і доцільності в сучасних умовах.

3 Програмна реалізація

В розділі здійснюється опис реалізації мобільного додатка та його компонентів (модулів), що супроводжується скріншотами, а також, за необхідності, фрагментами програмного коду. За визначеними вимогами реалізація кодової частини здійснюється залежно від операційної системи (Android чи iOS). Тут також описуються технічні характеристики розробленого додатка та основні вимоги для його правильного і безперебійного функціонування. Розділ може включати підрозділи.

3.1 Реалізація логіки мобільного додатку

У підрозділі здійснюється опис активностей додатка із визначенням дій, які виконує користувач. Описується також процес роботи класу Activity, його створення, запуск та перехід до інших активностей.

3.2 Реалізація розмітки мобільного додатка

У підрозділі здійснюється вибір відповідного контейнера (розмітки) мобільного додатка та опис всіх його компонентів (наприклад, розмітка для Android-програми будується з використанням ієрархії View та ViewGroup-об'єктів). Може бути наведена кодова частина реалізації розмітки відповідно до обраного контейнера.

3.3 Розробка бази даних

За наявності БД у цьому підрозділі необхідно реалізувати всю її обробку, включаючи створення, управління версіями, оновлення БД тощо. У додатку, при підключенні до БД, необхідно вказати засіб підтримки БД та його версію. Наприклад, в Android існує вбудована підтримка БД – SQLite (підтримуються всі функції SQLite, надається API-оболонка із сумісним інтерфейсом). API Android SQLite є типовим.

3.4 Керівництво користувача

У підрозділі розглядається докладна довідкова інформація для користувача, а також особливості роботи додатка.

3.4 Технічні характеристики мобільного додатка

Цей підрозділ повинен містити інформацію щодо технічних характеристик пристрою для гарантованого успішного виконання розробленого мобільного додатка (версію, ОС, API, обсяг пам'яті тощо).

4 Тестування мобільного додатка

Додаток перевіряється та тестується на відповідність ТЗ. Додаток встановлюється на пристрої і працює так, як би це було в Google Play чи AppStore. Цей розділ може містити такі підрозділи.

4.1 Аналіз методів тестування мобільного додатка

Здійснюється аналіз, вибір та обґрунтування методів і засобів тестування, які доцільно використовувати для мобільних додатків.

4.2 Тестування додатка за допомогою емулятора

У цьому підрозділі здійснюється опис результатів тестування мобільного додатка за допомогою автоматизованих засобів. Тестування мобільних додатків відбувається в середовищі розробки з використанням емулятора. Після цього додаток тестується на пристрої. Емулятори є простим способом тестувати додаток на мобільному телефоні, не використовуючи його фізично.

4.3 Аналіз результатів тестування мобільного додатка

В даному підрозділі подається аналітична інформація результатів здійсненого тестування. Результати можуть подаватись у вигляді схем, графіків, таблиць тощо.

Примітка. У випадку реалізації студентом нетипової або комбінованої теми ДП структура та зміст розділів основної частини узгоджується з керівником проекту (можна вилучати деякі підрозділи або додавати нові). Також підрозділи можна поділяти на пункти/підпункти, якщо це узгоджується із загальною логікою проектування.

4.10 Висновки

Цей структурний елемент є завершальним за результатами дипломного проектування і повинен містити узагальнені підсумки виконаної роботи. У ньому надається опис використаних методів і засобів інформаційних технологій для реалізації поставленої мети; надається опис робіт, які виконувались для вирішення поставленої задачі (роботи розглядаються в їх взаємозв'язку, з дотриманням послідовності їх виконання та визначенням результатів, що отримані на кожному етапі дипломного проектування); описуються отримані результати; робиться загальний висновок за результатами проектування.

Доцільно також сформулювати, які переваги користувачам надасть впровадження розробленого програмного продукту (наприклад, дозволить скоротити затрати часу, заощадити людські та грошові ресу-

рси, підвищити ефективність управління, прискорити швидкість та покращити якість обслуговування клієнтів тощо). Також варто зазначити, в яких ще практичних галузях доцільно використовувати розроблене ПЗ, навести результати впровадження, якщо вони отримані (акти впровадження, тези доповідей на конференціях тощо), а також оцінити можливі напрямки продовження роботи.

Всі матеріали повинні викладатися коротко, як підсумки виконаної роботи, а також відповідати визначеним завданням ДП.

Результати можна формулювати на основі висновків, зроблених в кінці кожного розділу, але вони не повинні підмінятися механічним підсумовуванням цих висновків.

Загальний обсяг структурного елемента «Висновки» – 2–4 с.

4.11 Перелік джерел посилання

Структурний елемент «Перелік джерел посилання» має містити перелік джерел, використаних у ДП. Такими джерелами можуть бути книги, періодичні видання (журнали), нормативно-технічні документи (стандарты, патенти, каталоги), електронні ресурси тощо. У цьому структурному елементі забороняється вказувати сторінки Вікіпедії, Студопедії, веб-сайти рефератів та інші подібні ресурси.

В основному тексті на всі джерела повинні бути посилання, тому перелік джерел розташовують у порядку посилань на них.

4.12 Додатки

У додатках розміщують матеріал, що унаочнює або доповнює основний текст документа (рисунок, таблиці великого формату, опис алгоритмів, лістинги програм, протоколи тестування ПЗ, акти про впровадження та інші матеріали, які допомагають докладніше розкрити задум та шляхи реалізації ДП).

Додатки можуть мати такий статус: обов'язковий; довідковий.

У *довідковому* додатку наводять довідкові відомості (вихідні дані до проекту, бланки документів тощо).

В *обов'язковому* додатку подають виклад окремих положень документа, щоб уникнути переобтяження основного тексту (ТЗ, моделі та алгоритми, лістинги програм, презентаційні матеріали тощо).

В основному тексті документа на всі додатки повинні бути посилання, тому додатки розташовують у порядку посилань на них.

Всі додатки повинні бути перераховані у «Змісті» із зазначенням їх позначень та назв (статус додатка вказувати не потрібно).

5 ВИМОГИ ДО ОФОРМЛЕННЯ ДИПЛОМНОГО ПРОЕКТУ

5.1 Основні вимоги до пояснювальної записки

Загальними вимогами до пояснювальної записки є логічна послідовність викладення матеріалу, чіткість та конкретність результатів проектування, суті постановки завдання та мети ДП, методів дослідження, прийнятих рішень, обґрунтованість висновків тощо. Текст не повинен бути перевантажений малоінформативним матеріалом, описом загальновідомих методів. Текст пояснювальної записки має бути стислим, чітким, лаконічним та добре відредагованим.

При викладі обов'язкових вимог у тексті слід застосовувати такі слова та словосполучення: *«повинен», «впливає», «необхідно», «потрібно, щоб», «дозволяється тільки», «не допускається», «забороняється»* тощо. При викладі інших положень варто застосовувати такі слова, як: *«можуть бути», «як правило», «за необхідності»* тощо. При цьому допускається використовувати оповідальну форму викладу тексту, вживаючи такі слова: *«застосовують», «вважають», «рекомендують»*. У пояснювальній записці не рекомендується вести викладення матеріалу «від першої особи»: *«Я визначив ...», «Я вважаю ...», «Мені здається ...», «На мою думку ...»* тощо, а викладати текст слід безособовою формою за всім текстом.

Текст викладають, дотримуючись норм чинного українського правопису та лексики, використовуючи стиль ділового мовлення, придатний для службових документів. У тексті повинні застосовуватись терміни, позначення та визначення, встановлені чинними стандартами, а при їх відсутності – загальноприйняті у науково-технічній літературі.

Також не бажано вживати іншомовні слова та терміни за наявності рівнозначних слів і термінів в українській мові.

Безпосередньо у тексті документа замість цифр слід писати їх словесне значення (наприклад, *«чотири ознаки»*, але *«3 МГц»*); числа 10, 11, ... записуються цифрами.

У тексті документа, за винятком формул, таблиць та рисунків, не дозволяється вживати:

– символ « \emptyset » як позначення діаметра (треба писати слово «діаметр»); зазначаючи розмір або граничні відхилення діаметра на рисунках, перед його числовим значенням треба ставити знак « \emptyset »;

– без числових значень математичні знаки, наприклад: « $>$ » (більше), « $<$ » (менше), « $=$ » (дорівнює), « \geq » (більше або дорівнює), « \leq » (менше або дорівнює), « \neq » (не дорівнює), а також «№» (номер), «%» (відсотки) та «°C» (градус Цельсія);

– математичний знак мінус (–) перед від’ємним значенням величини (треба писати слово «мінус»).

Коли наводять допустимі значення відхилень зазначених норм, вимог, треба вживати словосполучення «не повинен бути більший ніж (від, за)», «менший ніж (від, за)», «не повинно (-ен, -а) перевищувати».

Коли наводять найбільше або найменше значення величини, треба вживати словосполучення «повинно бути не більше (-ий, -а) ніж (від, за)», «не менше (-ий, -а) ніж (від, за)».

У тексті документа дозволено вживати:

– загальноприйняті скорочення: див. – дивись; мін. – мінімальне; макс. – максимальне тощо (згідно ДСТУ 3582–2013);

– скорочення: абс. – абсолютний; с. – сторінка; р. – рік; грн – гривня та інші скорочення, які вживають з числовими значеннями.

Замість скорочень «і т. д.» (і так далі), «і т. ін.» (і таке інше), «і т. п.» (і тому подібні) рекомендовано вживати «тощо».

Якщо в тексті прийнято особливу систему скорочення слів або назв, то перелік прийнятих скорочень повинен бути наведений після змісту у структурному елементі «Перелік скорочень».

5.2 Вимоги до оформлення основного тексту

Вимоги щодо оформлення ДП регламентуються державними стандартами України та нормативними документами університету.

Пояснювальна записка ДП повинна бути оформлена на аркушах ф. А4 (210 мм × 297 мм) з використанням форм 9 і 9а стандарту ДСТУ ГОСТ 2.106–96; при цьому основний напис виконують відповідно до вимог ДСТУ ГОСТ 2.104:2006 (форма 2 для «Відомості документів» та «Змісту», форма 2а для наступних аркушів). У разі потреби можна використовувати аркуші ф. А3 (297 мм × 420 мм).

Примітка. Титульний аркуш, завдання на ДП, анотація та додатки оформлюються на звичайних аркушах ф. А4 (**без рамки**).

На формах відстань від рамки форми до меж тексту на початку і в кінці рядків має бути не менше ніж 3 мм (рекомендовано 5 мм). Відстань від верхнього чи нижнього рядка тексту до верхньої чи нижньої рамки має бути не менше ніж 10 мм (рекомендовано 10 мм).

Відстані рамки форми до країв аркуша мають бути наступними: від лівого краю – не менше ніж 20 мм, від правого, верхнього та нижнього – 5 мм.

Для титульного аркуша, анотації та додатків рекомендовані такі береги сторінок: верхній і нижній – 20 мм, лівий – не менше ніж 20 мм, правий – 10 мм; для завдання на ДП – всі береги по 20 мм.

Рекомендується також дотримуватись наступних вимог: основний шрифт – Times New Roman; накреслення шрифту – звичайне (за винятком назв структурних елементів та заголовків розділів); висота набору – 14 пт; колір шрифту – чорний; міжрядковий інтервал – 1,5 (півтора інтервали); вирівнювання основного тексту – по ширині; абзацний відступ – 1,25 см.

Не дозволяється в останньому рядку абзацу розміщувати лише одне слово. Якщо такий випадок має місце, то слід відповідним чином переформулювати текст абзацу або використовувати ущільнений інтервал між символами (але не більше ніж на 0,2 пт).

5.3 Нумерація сторінок пояснювальної записки

Сторінки пояснювальної записки слід **нумерувати арабськими цифрами** (без крапки в кінці), додержуючись наскрізної нумерації впродовж усього документа, включаючи додатки. На формах номер сторінки проставляється у рамці справа в полі *Арк.*

Для додатків нумерація продовжується, але номер сторінки проставляється у **верхньому колонититулі справа**.

Титульний аркуш включають до загальної нумерації сторінок (з номером 1), але номер сторінки на ньому не проставляють.

«Завдання на дипломний проект» (з номером 2) та «Анотація» (з номером 3) також входять до загальної нумерації сторінок документа, але номер сторінки на них не проставляють.

«Відомість документів» виконується на формі 9 стандарту ДСТУ ГОСТ 2.106–96; при цьому основний напис виконують відповідно до вимог ДСТУ ГОСТ 2.104:2006 (форма 2). «Відомість документів» є самостійним документом, що **не включається** у загальну кількість сторінок пояснювальної записки, має номер 1, який проставляється у рамці справа в полі *Арк.* У полі *Аркушів* проставляється також загальна кількість сторінок «Відомості документів» (зазвичай 1).

Зразок-шаблон «Відомості документів» подано у додатку Ж.

«Зміст» виконується на формах 9 та 9а ГОСТ 2.106–96; при цьому основний напис виконують відповідно до вимог стандарту ДСТУ ГОСТ 2.104:2006 – ф. 2 для першої сторінки «Змісту», ф. 2а (з «малою» рамкою) – для наступної сторінки «Змісту» (якщо така є), а також для наступних сторінок тексту.

«Зміст» включають у загальну кількість сторінок документа. На першому аркуші «Змісту» у рамці справа проставляється номер сторінки (номер 4 у полі *Арк.*), а також загальна кількість сторінок пояснювальної записки (у полі *Аркушів*).

Приклад оформлення «Змісту» та шаблон форми 2а (для наступних сторінок) подано у додатках И та К, відповідно.

5.4 Вимоги до викладення тексту

5.4.1 Поділ тексту

Текст документа, залежно від його розуміння за змістом, поділяють на розділи, підрозділи, пункти і підпункти, що нумеруються арабськими цифрами: розділи – у межах усього документа, підрозділи – у межах кожного розділу, пункти – у межах підрозділу, підпункти – у межах пункту. Кожен розділ слід починати з нової сторінки. Номер розділу записують без крапки в кінці.

Номер підрозділу повинен складатися з номера розділу, крапки-розмежувача і номера підрозділу; наприкінці номера крапку не ставлять. Наприклад, 2.1 – перший підрозділ другого розділу.

Пункти нумерують арабськими цифрами у межах кожного підрозділу. Номер пункту повинен складатися з номерів розділу, підрозділу і пункту, розділених крапками (але без крапки в кінці). Наприклад, 2.1.3 – третій пункт першого підрозділу другого розділу.

Структурні елементи «Анотація», «Відомість документів», «Зміст», «Перелік скорочень», «Вступ», «Висновки», «Перелік джерел посилання» не нумерують.

Заголовки структурних елементів та розділів слід друкувати з абзацного відступу **ВЕЛИКИМИ ЛІТЕРАМИ** напівжирним шрифтом без крапки в кінці. Дозволено їх розміщувати посередині рядка (в цьому випадку – без абзацного відступу).

Заголовки підрозділів, пунктів і підпунктів слід друкувати з абзацного відступу з великої літери без крапки в кінці. Розривати слова знаком переносу у будь-яких заголовках не можна.

Відстань між заголовком і подальшим або попереднім текстом має бути не менше ніж *подвійний міжрядковий інтервал*. Відстань між рядками заголовка і між двома заголовками *приймають такою, як у тексті*.

Не можна розміщувати назву підрозділу/пункту/підпункту у нижній частині сторінки, якщо після назви розміщено менше ніж два рядки тексту. Якщо такий випадок має місце, то допускається (в межах окремих сторінок) змінювати міжрядковий інтервал, але не більше ніж на 0,02 (рекомендовані значення множника від 1,48 до 1,52).

5.4.2 Переліки

Усередині підрозділів/пунктів/підпунктів можуть бути наведені переліки. Позиції переліку формулюються у вигляді одного речення.

Якщо подають переліки одного рівня підпорядкованості, то перед кожним із переліків ставлять знак «←» («тире»). При наявності

більше одного рівня підпорядкованості перший рівень позначають малими українськими літерами, далі (другий рівень) – арабськими цифрами, далі (третій рівень) – через «-». Після цифри або літери переліку ставлять круглу дужку.

Текст кожної позиції переліку слід починати з абзацного відступу відносно попереднього рівня підпорядкованості та закінчувати символом «;». Позиції переліку першого рівня підпорядкованості починаються з абзацного відступу відносно основного тексту. В кінці тексту (або позиції переліку попереднього рівня) перед переліком ставлять двокрапку. В кінці останньої позиції переліку ставиться крапка.

Кожну позицію переліку слід починати з *малої літери* (окрім випадків, коли перелік починається словом, яке, згідно з правилами правопису, повинно записуватись з великої літери).

Узагальнений формат переліку поданий на рисунку 5.1.

- | |
|---|
| <p>а) елемент першого рівня підпорядкованості:</p> <ul style="list-style-type: none">1) елемент другого рівня підпорядкованості:<ul style="list-style-type: none">– елемент третього рівня підпорядкованості;– елемент третього рівня підпорядкованості;–2) елемент другого рівня підпорядкованості; <p>б) елемент першого рівня підпорядкованості:</p> <ul style="list-style-type: none">1) елемент другого рівня підпорядкованості;2) елемент другого рівня підпорядкованості;3) елемент другого рівня підпорядкованості; <p>в) елемент першого рівня підпорядкованості.</p> |
|---|

Рисунок 5.1 – Узагальнений формат переліку

Приклад однорівневого переліку

В описі програми повинні бути присутні наступні пункти:

- загальні положення;
- функціональне призначення;
- опис логічної структури;
- технічні засоби, що використовуються;
- вхідні та вихідні дані;
- інструкція з використання.

Приклад багаторівневого переліку

В описі програми повинні бути присутні наступні пункти:

- а) загальні положення:
 - 1) інструментальні засоби розробки;

- 2) мови конструювання:
 - мови програмування;
 - мови розмітки;
- б) функціональне призначення;
- в) опис логічної структури;
- г) технічні засоби, що використовуються;
- д) вхідні та вихідні дані;
- е) інструкції з використання:
 - 1) для системного адміністратора;
 - 2) для користувачів.

5.4.3 Формули та рівняння

Формула входить до речення як його рівноправний елемент. Тому в кінці формул і в тексті перед ними розділові знаки ставлять відповідно до правил пунктуації. Невеликі формули, що не мають самостійного значення і на них немає посилання у тексті, вписують всередині рядків тексту. Складніші формули розміщують на окремих рядках по центру. Вище та нижче кожної формули (рівняння) повинен бути **один вільний рядок**.

Для введення формул та рівнянь рекомендовано використовувати редактор формул (Microsoft Equation 3.0 або MathType).

Основні параметри формули:

- стиль – математичний;
- шрифт – Times New Roman, прямий, 14 пт (винятки: для змінної – *курсив*; для матриці-вектора – прямий **напівжирний**);
- крупний індекс – 9 пт;
- звичайний індекс – 7 пт;
- крупний символ (наприклад, символи суми, добутку) – 18 пт;
- звичайний символ – 14 пт.

Формули, на які є посилання в тексті, розташовують в окремому рядку і нумерують наскрізною нумерацією арабськими цифрами. У разі великої кількості формул використовується нумерація у межах розділу. У цьому випадку номер формули має складатися з номера розділу і порядкового номера формули, розділених крапкою. Номер формули ставиться на рівні формули (або на рівні нижнього рядка рівняння), до якої він відноситься, у круглих дужках у правому положенні відносно сторінки. Якщо в тексті лише одна формула, то її нумерують відповідно (1).

Розділовими знаками між формулами, котрі йдуть одна за одною і не відокремлені текстом, можуть бути «кома» або «крапка з комою» безпосередньо за формулою (до її номера). Для економії місця кілька коротких однотипних формул можна подати в одному рядку.

Значення символів і коефіцієнтів, що входять у формулу, повинні бути наведені під формулою. Значення кожного символу друкують з нового рядка за тією послідовністю, за якою вони наведені у формулі. Перший рядок розшифровки повинен починатися зі слова «де» *без абзацного відступу* та *без двокранки* після нього. Позначки і пояснення до них рекомендовано вирівнювати у вертикальному напрямку.

Приклад

Ймовірність того, що в n випробуваннях подія A відбудеться k раз, обчислюється за формулою Бернуллі:

$$P_n(k) = C_n^k \cdot p^k \cdot q^{n-k}, \quad (5.1)$$

де C_n^k – кількість комбінацій з n по k , $C_n^k = \frac{n!}{k! \cdot (n-k)!}$;

p – ймовірність події A в кожному випробуванні;

$q = 1 - p$.

Для формул, елементи яких вживаються з позначками одиниць виміру, у поясненні додатково вказують позначки таких одиниць.

Приклад

Час виконання програми процесором наближено оцінюється за допомогою формули:

$$T = \frac{N \cdot A}{F}, \quad (5.2)$$

де T – час виконання програми, с;

N – кількість команд, які виконуються у програмі;

A – середнє число тактів процесора на одну команду;

F – тактова частота процесора, Гц.

Посилання на формулу подають так: «у формулі (3.7)»; «... з рівнянь (1.3)–(1.5)». Перше посилання на формулу має їй передувати.

5.4.4 Таблиці

Таблиці у загальному випадку повинні мати форму, наведену на рисунку 5.2. Напростишою таблицею є таблиця з колонками і рядками без їх розбиття чи об'єднання.

Таблицю варто поміщати після першого згадування про неї в тексті або (якщо вона не поміщається на цій же сторінці) на черговій сторінці; за потреби таблиці можна розміщувати у додатках. Дозволе-

но розташовувати таблицю вздовж довгого боку аркуша (альбомна орієнтація аркуша).

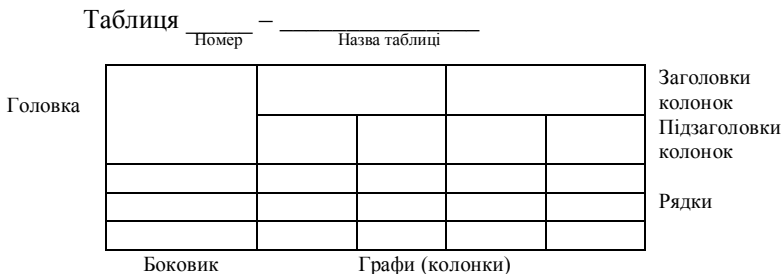


Рисунок 5.2 – Узагальнена форма таблиці

Таблиця має бути розташована на відстані **один вільний рядок** від попереднього й наступного тексту.

Таблиці нумерують наскрізно арабськими цифрами (крім таблиць у додатках). Дозволено таблиці нумерувати у межах розділу. У цьому разі номер таблиці складається з номера розділу та порядкового номера таблиці, відокремлених крапкою (наприклад, «Таблиця 1.2», тобто друга таблиця першого розділу). Якщо в документі лише одна таблиця, її позначають «Таблиця 1». Назву пишуть з першої великої літери і розташовують над таблицею зліва з **абзацного відступу**. В кінці назви крапку не ставлять, наприклад, «Таблиця 4.1 – Порівняльні характеристики». Не дозволено розмішувати назву таблиці на останньому рядку сторінки і розривати з самою таблицею.

Примітка. Нумерація рядків подається поряд з текстом у першій колонці без крапки в кінці номера. Спеціальну колонку зліва № з/п («Номер за порядком») у таблицях подають як **виняток**, позначаючи її як «Ч.ч» («Чергове число»).

Приклад

Таблиця 5.1 – Системні вимоги, необхідні для інсталяції ПЗ

Компонент	Мінімальні вимоги	Рекомендовані вимоги
1 Процесор	Pentium/ Celeron, AMD K6/Athlon/ Duron та ін., 550 МГц	Pentium IV, 1 ГГц
2 Дисковий простір	30 Мб	500 Мб
3 Оперативна пам'ять	256 Мб	512 Мб
4 Операційна система	Windows XP	Windows 7 або пізнішої версії
5 Додатки	Microsoft Office 2003	Microsoft Office 2007 або пізнішої версії

При переносі частини таблиці на іншу сторінку слово «Таблиця», а також її номер та назву вказують один раз зліва над першою частиною таблиці; над іншими частинами з абзацного відступу пишуть слова «Продовження таблиці» або «Кінець таблиці» і вказують її номер (наприклад, «Продовження таблиці 1.2» або «Кінець таблиці 1.2»). При цьому головку таблиці на перенесених частинах рекомендується замінити відповідно номерами колонок, проставивши ці номери у першій частині таблиці. Нижню обмежувальну горизонтальну лінію наводять лише у кінцевій частині таблиці.

Приклад

Таблиця 5.2 – Динаміка товарних запасів фірми у 2020 році

Місяць	Товарні запаси, тис. грн		Відсоток відповідності товарних запасів нормативу	Відхилення (+, -), тис. грн
	норматив	факт		
1	2	3	4	5
Січень	225,00	227,78	101,2	+2,78
Лютий	191,70	183,67	95,8	-8,03
...

Продовження таблиці 5.2

1	2	3	4	5
Травень	191,7	198,77	103,7	+7,07
...

Кінець таблиці 5.2

1	2	3	4	5
Листопад
Грудень

Розмір шрифту для написання заголовків у рядках і колонках таблиць та пояснювальних даних у таблицях може встановлювати автор ДП. Допускається використовувати розмір шрифту, менший ніж в основному тексті (але не менше ніж 12 пт). В таблицях допускається також одинарний міжрядковий інтервал.

На кожну таблицю має бути посилання в тексті із зазначенням її номера. Посилання на таблицю має передувати самій таблиці.

Приклади

«Динаміка товарних запасів оптової фірми у 2020 році представлена у таблиці 5.2».

«У таблиці 2.2 подані варіанти використання та їх описи».

«В результаті випробувань визначені системні вимоги, необхідні для інсталяції ПЗ (таблиця 3.4)».

5.4.5 Рисунки

Графіки, схеми, креслення, фото тощо іменуються рисунками. Кожний рисунок (ілюстрація) має відповідати тексту і навпаки.

Рисунки потрібно розміщувати у документі безпосередньо після тексту, де вони згадуються вперше, або на наступній сторінці (а за потреби – в додатках), симетрично до тексту (задавши положення рисунка – «в тексті»). Відстань від рисунка до попереднього і наступного тексту – **один вільний рядок**.

Примітка. Ілюстрації великих розмірів допускається виконувати на аркушах формату А3 і розміщувати їх в додатках.

Усі рисунки повинні мати однаковий підпис «Рисунок», який розміщують під рисунком симетрично до нього.

Рисунки нумерують наскрізно арабськими цифрами (окрім рисунків у додатках). Дозволено нумерувати їх в межах кожного розділу; в цьому разі номер рисунка складається з номера розділу та номера рисунка у цьому розділі, які відокремлюють крапкою (наприклад, «Рисунок 2.5 – Назва рисунка», тобто п'ятий рисунок другого розділу). Якщо у документі лише один рисунок, то він позначається «Рисунок 1». Назва рисунка має відображати його зміст і бути стислою. Назву пишуть з великої літери; в кінці назви крапка не ставиться.

Приклади оформлення рисунків та посилань на них див. у п. 4.9.1 – рисунки 4.2 – 4.5.

Якщо до рисунка треба додати пояснювальні дані, то їх розташовують після графічного матеріалу перед назвою рисунка.

Приклад

Основні принципи роботи мобільних додатків, розроблених за допомогою різних підходів, показані на рисунку 5.3.



Рисунок 5.3 – Основні підходи до розробки мобільних додатків

Примітка. Якщо елементи на рисунку позначені цифрами, то пояснювальний текст під рисунком можна подавати так:

- 1 – проміжний перетворювач; 2 – різницевий пристрій;
- 3 – блок перетворення; 4 – монітор.

Для запису пояснювальних даних на самому рисунку, задля їх компактного розміщення, розмір шрифту для них (а також накреслення, стиль тощо) може встановлюватись автором ДП.

На всі рисунки мають бути посилання в тексті.

Приклади

«Інфологічна модель бази даних подана на рисунку 2.2».

«В результаті аналізу потреб майбутніх користувачів ПЗ побудованна діаграма варіантів використання (рисунок 1.2)».

Примітка. Рисунок (так само, як і таблиця чи формула), не може з'явитись у тексті раніше першого посилання на нього. В подальшому, на один елемент може бути декілька посилань у тексті.

Рисунки подають у **чорно-білому** чи **кольоровому** зображенні.

5.4.6 Оформлення програмного коду та його фрагментів

Лістинги програмних кодів, як правило, виносять у додатки. Фрагменти коду, які є ключовими для вирішення завдання, за потребою, можуть бути наведені в основній частині пояснювальної записки у вигляді тексту для кращого опису реалізації ПЗ. Текст програми (як і її фрагмента) повинен добре «читатися» за рахунок правильного структурування та форматування коду, а також використання коментарів.

Лістинги коду та його фрагмента, як правило, наводять моноширинним шрифтом (наприклад, **Courier New**) з вирівнюванням «зліва» без абзацних відступів. Допускається зменшення розміру шрифту (до 10 пт), а також одинарний міжрядковий інтервал.

Фрагмент програмного коду розташовують на відстані **один вільний рядок** від попереднього й наступного тексту.

Приклад

Наведемо код програмної реалізації динамічного розподілу пам'яті під одновимірний масив:

```
int *a, n;
printf("Введіть розмір масиву: ");
scanf_s("%d", &n);
// виділення пам'яті під масив
a = (int *) malloc(n*sizeof(int));
```

```
// перевірка виділення пам'яті
if (!a)
{
    puts ("Помилка при виділенні пам'яті\n");
    _getch ();
    return 1;
}
```

5.4.7 Посилання

У тексті пояснювальної записки можна вказувати посилання на її структурні елементи та інші джерела.

При посиланнях на структурні елементи зазначають номери розділів, підрозділів, пунктів, підпунктів, рисунків, формул, таблиць, додатків. Посилаючись, слід використовувати вирази: «у розділі 3», «відповідно до 3.3.2», «(рисунок 2.5)», «на рисунку В.1», «відповідно до таблиці 1.3», «згідно з формулою (2.2)», «(додаток Д)» тощо.

У посиланні можна використовувати загальноприйняті та стандартизовані скорочення слів («згідно з рис. 3.4», «див. табл. 2.1»). Як правило, посилання зі скороченим словом здійснюються на ілюстрації (таблиці, формули), вже згадані раніше. Посилання в тексті на джерела треба зазначати порядковим номером за переліком посилань, виділеним двома квадратними дужками.

Приклади

«...у статті [6]...»;
 «...у роботах [1–4]...»;
 «...використовується модель ... [8]».

5.4.8 Вимоги до оформлення переліку джерел посилання

Бібліографічні описи у «Переліку джерел посилання» наводять відповідно до стандартів з бібліотечної і видавничої справи. Кожне джерело має свій порядковий номер, а весь список – єдину наскрізну нумерацію. Бібліографічний опис будь-якого джерела включають у перелік посилань лише один раз.

Використовують різні способи групування джерел, основними з яких є алфавітний та нумераційний (за першим згадуванням джерела в тексті). Рекомендується у переліку джерел бібліографічні описи подавати у порядку, за яким джерела вперше згадуються у тексті.

У відповідних місцях тексту мають бути посилання на джерела. Порядкові номери описів у переліку джерел мають відповідати посиланням на них у тексті.

У додатку Л наведені приклади бібліографічних описів.

5.4.9 Вимоги до оформлення додатків

Додатки оформлюють як продовження основного документа.

Додатки позначають послідовно великими літерами українського алфавіту, починаючи з А (за винятком Г, Є, З, І, Ї, Й, О, Ч, Ї), наприклад, «ДОДАТОК Д». Якщо в документі тільки один додаток, його позначають «ДОДАТОК А».

Кожний додаток треба починати з нової сторінки із зазначенням зверху посередині сторінки слова «ДОДАТОК» і його позначення (прямим шрифтом великими літерами). У наступному рядку посередині має бути статус додатка, який друкують прямим шрифтом малими літерами; далі, через рядок, посередині друкують назву додатка (**напівжирним шрифтом ВЕЛИКИМИ** літерами).

Текст кожного додатка можна поділити на розділи, підрозділи, пункти, підпункти, які нумерують у межах кожного додатка. Перед номерами ставлять літерне позначення цього додатка: «А.1», «В.2.4».

Якщо в додатках є рисунки, таблиці, формули, то їх також нумерують у межах кожного додатка («*рисунок А.3*», «*таблиця С.1*», «*формула (Д.2)*»). Якщо в додатку лише один рисунок (таблиця, формула), їх нумерують так: «*рисунок А.1*», «*таблиця Г.1*», «*формула (С.1)*».

Наскрізна нумерація рисунків, таблиць, формул (на відміну від основного тексту) у додатках не дозволяється.

При посиланні у тексті пояснювальної записки (чи додатків) на рисунки, таблиці та формули, які розміщені у додатках, слід писати: «...на *рисунку А.2*», «(*див. рис. А.2*)»; «...у *таблиці Б.3*», «...у *табл. Б.3*»; «...за *формулою (В.4)*».

Тексти лістингів програмних кодів у додатках допускається виділяти відмінним від основного тексту типом шрифту та/або іншим розміром шрифту (але не менше ніж 10 пт). Допускається також зменшення міжрядкового інтервалу до 1.

Якщо в додатку наводять документ, що має самостійне значення (наприклад, ТЗ, презентаційні матеріали), то у цьому разі, перед матеріалом додатку, на окремому аркуші посередині друкують великими літерами слово «ДОДАТОК» та його позначення, а також статус і назву додатка (за тими самими правилами). Аркуш з цією інформацією також нумерують.

Кількість та зміст додатків визначає студент-дипломник за погодженням з керівником проекту.

Примітка. Повний опис вимог до оформлення текстових документів наведений у стандартах ДСТУ 3008:2015, ДСТУ ГОСТ 7.1:2006, ДСТУ 8302:2015, ДСТУ 3582:2013 та СОУ 207.01:2017.

5.5 Вимоги до оформлення графічної частини

Презентаційні матеріали створюються у вигляді слайдів за допомогою відповідних програмних засобів (типу Microsoft PowerPoint) і передбачають використання під час захисту ДП проекційної техніки.

Стиль оформлення слайдів дипломник обирає самостійно.

Роздруковані слайди (ф. А4) включаються у додатки.

Якщо у «Завданні на проектування» передбачена графічна частина у вигляді демонстраційних плакатів, то вона оформлюється на аркушах креслярського паперу формату А2 і супроводжується основним написом та додатковими графами до нього відповідно до вимог ДСТУ ГОСТ 2.104:2006 – форма 2. Таку графічну частину виконують за допомогою сучасних комп'ютерних засобів та пакетів програм (Case-засоби, Corel Draw тощо). Аркуші нумеруються – номер аркуша проставляється у рамці справа в полі *Арк.* (1, 2, ...). Також проставляється загальна кількість аркушів (у полі *Аркушів*). Аркуші підписує студент, керівник ДП, нормоконтролер та завідувач кафедри.

6 ПІДГОТОВКА ДО ЗАХИСТУ ДИПЛОМНОГО ПРОЕКТУ

6.1 Підготовка документації

Для проведення захисту дипломник не менш ніж за п'ять днів до дати захисту має подати на кафедру готову і зшити (у «твердій» обкладинці) пояснювальну записку з підписами відповідальних осіб; аркуші записки зшиваються у послідовності, вказаній у п. 4.1.

До пояснювальної записки додають:

- 1 Відгук керівника ДП.
- 2 Рецензія на ДП.
- 3 Довідка про проходження перевірки на плагіат.
- 4 Графічні матеріали (якщо вони планувались).
- 5 Інші документи (довідка про впровадження результатів ДП, наукова стаття тощо), якщо такі є.

Примітка. Пояснювальна записка разом з розробленим ПЗ надається рецензенту для незалежного оцінювання. Рецензент повинен детально ознайомитися з теоретичними та практичними результатами дипломного проектування і дати оцінку цим результатам у вигляді письмової рецензії. Склад рецензентів затверджується завідувачем кафедри.

Відгук керівника, рецензія, довідка про проходження перевірки на плагіат та інші документи вкладаються у конверт, наклеєний на внутрішньому боці обкладинки пояснювальної записки.

Допуск ДП до захисту засвідчується підписом завідувача кафедри на титульному аркуші пояснювальної записки. ДП, не підписаний завідувачем кафедри, до захисту не допускається.

Примітка. Завідувач кафедри має право допустити студента до захисту ДП у разі негативного відгуку керівника. Негативна рецензія також не є підставою для відхилення проекту від його захисту.

6.2 Перевірка, виявлення плагіату та його усунення

Згідно з «Положенням про дотримання академічної доброчесності в Хмельницькому національному університеті» всі ДП підлягають обов'язковій процедурі перевірки на наявність плагіату.

Перевірка рівня запозичень у ДП здійснюється не пізніше ніж за три дні до захисту проекту і виконується відповідальною особою (далі – відповідальний), призначеною завідувачем кафедри.

Функціями відповідальної особи є:

- завантаження ДП (пояснювальної записки) у систему Anti-Plagiarism та здійснення її комп'ютерної перевірки на плагіат;
- оформлення довідки за результатами перевірки;
- архівування ДП у репозитарії;
- збереження конфіденційності інформації щодо ДП.

Відповідальний приймає завершений ДП, підписаний керівником, у друкованому вигляді, а також його електронну версію у форматі *.rtf, *.doc, *.docx, *.pdf. Відповідальний здійснює вибірккову перевірку на предмет збігу між друкованою та електронною версіями ДП. Якщо друкована та електронна версії не збігаються, ДП повертається дипломнику для усунення розбіжностей.

Після перевірки пояснювальної записки на плагіат відповідальний видає дипломнику відповідну довідку, яка додається до ДП.

Для ДП допускається максимальний збіг з однією роботою не більше ніж 40%. Наявність помилок у проекті не повинна перевищувати межу 20 %, а у разі використання специфічних термінів (які можуть бути відсутніми у словниках) – 30 %. Також робота не повинна мати менше ніж 60 000 знаків (300 лексем).

У разі виявлення плагіату, що перевищує встановлені нормативи (на підставі довідки), дипломник не допускається до захисту ДП до моменту усунення порушень та повторного проходження перевірки на плагіат. Допускається не більше ніж три перевірки одного ДП.

6.3 Нормоконтроль

Завданням нормоконтролю є забезпечення дотримання у документації норм, вимог та правил, установлених стандартами та іншими нормативними документами. Основні вимоги цих стандартів та нормативних документів університету викладені у розділі 5 (с. 40) даних методичних вказівок. Для здійснення нормоконтролю кафедра признає відповідальною особу – **нормоконтролера**.

До представлення ДП на нормоконтроль матеріали повинні бути роздруковані та підписані дипломником та керівником проекту. З метою якісного проведення нормоконтролю і виправлення помилок пояснювальна записка повинна бути закінчена орієнтовно за сім днів до початку захисту ДП.

Зміни та виправлення, зазначені нормоконтролером і пов'язані з порушенням діючих стандартів та інших нормативно-технічних документів, є обов'язковими для внесення у документи ДП.

Відповідність ДП вимогам чинних стандартів нормоконтролер засвідчує своїм підписом. Виправляти чи змінювати підписані нормоконтролером документи без його відома не допускається.

Нормоконтролер несе відповідальність за дотримання у документації вимог чинних стандартів та інших нормативно-технічних документів на рівні з розробниками документації.

6.4 Підготовка доповіді

Доповідь повинна стисло і технічно грамотно відображати: сутність поставленої задачі; обґрунтування актуальності теми, мету і завдання ДП, призначення розробки; сутність проведеного аналізу; функціональний склад розробки; обрані засоби для розробки ПЗ; результати проектування; основні характеристики розробленого ПЗ; висновки (короткі результати всього проектування, можливі шляхи подальшого вдосконалення, використання або впровадження).

Доповідь має включати три взаємопов'язані частини: **вступ**, **основну частину** та **висновки**.

Вступ слід почати із звернення до членів ЕК та презентації теми ДП, наприклад: *«Шановні члени екзаменаційної комісії! Вашій увазі пропонується дипломний проект за темою...»*. Далі у вступі потрібно визначати область, до якої відноситься тема ДП, висвітлити актуальність розробки, вказати мету та основні задачі проектування.

В **основній частині** слід коротко розглянути можливі варіанти вирішення задачі; пояснити, як вирішувалася задача, обґрунтувати правильність прийнятого рішення; розкрити основні результати проектування; продемонструвати рівень вирішення поставлених задач, а також основні характеристики розробленого ПЗ.

Висновки повинні бути акцентовані на основних результатах проектування, досягненні мети ДП, практичній значимості проекту, рекомендаціях тощо. Висновки та рекомендації слід наводити в узагальненій формі, уникаючи зайвої деталізації.

Доповідь повинна закінчуватись словами: *«Дякую за увагу, доповідь закінчено»*.

Запропонована структура доповіді є узагальненою і може конкретизуватися та змінюватися залежно від змісту ДП, отриманих результатів та представлених наочних (демонстраційних) матеріалів.

6.5 Підготовка наочних матеріалів

Наочні (демонстраційні) матеріали мають послідовно ілюструвати доповідь і забезпечувати повноту висвітлення всіх положень ДП.

Перший слайд презентації повинен містити назву кафедри, тему ДП, прізвище та ім'я студента, прізвище, ініціали, науковий ступінь та вчене звання керівника ДП.

На другому слайді розміщують чітко сформульовану постановку задачі, мету та задачі проектування.

На наступних слайдах (1–2) надають інформацію щодо актуальності поставленої задачі, а також результати аналізу існуючих рішень та висновки, зроблені на основі цього аналізу (перелік аналогів розробленого ПЗ із зазначенням їх недоліків та обмежень).

Далі на слайдах розміщують результати проектування: методи, формули, зміст виконаної роботи (моделі, алгоритми, структурна схема розробленого ПЗ, формати і алгоритми передачі та обробки даних, схема інтерфейсу, «скріншоти» вікон розробленого ПЗ тощо).

На завершення наводять чітко й лаконічно сформульовані висновки із зазначенням результатів ДП та їх практичної цінності.

Презентація та доповідь повинні бути узгоджені в часі. Виступ студента має бути розрахований на 10–20 хв.

6.6 Підготовка до відповідей на питання

Питання під час захисту ДП задаються членами ЕК та присутніми особами, як правило, за темою ДП та наведеними у пояснювальній записці і доповіді результатами. Тому дипломник повинен вільно орієнтуватися у проєкті, знати суть викладеного у ньому матеріалу, добре розуміти принципи роботи розробленої програми тощо.

Кількість та характер питань значною мірою залежать від якості доповіді. Студенту слід бути готовим до несподіваних питань типу: *«А навіщо це потрібно?»* або стосуватись деталей, якими студент не займався. Не слід відповідати: *«А так вимагав замовник»* чи *«Я цього не знаю»*. Краще проявити свою ерудицію та винахідливість, наприклад: *«Це не входило у завдання дипломного проєкту»*.

Питання, які для студента є не зрозумілими, слід уточнювати.

6.7 Репетиція доповіді

Репетиція доповіді є дуже важливою з погляду визначення необхідного для неї часу. Під час репетиції доповіді рекомендується задіювати всю апаратуру, яка знадобиться під час захисту ДП, а також всі необхідні наочні (демонстраційні) матеріали. Особливу увагу слід звернути на підготовку та налаштування необхідного обладнання для демонстрації розробленого програмного засобу. Для надійності бажано також мати відео з демонстрацією роботи розробленої програми.

7 ЗАХИСТ ДИПЛОМНОГО ПРОЕКТУ

Публічний захист ДП є завершальним етапом роботи над дипломним проектом, у ході якого дипломник має продемонструвати свої професійні якості, вміння показати результати своєї роботи, а також презентувати розроблений проект.

Захист ДП здійснюється перед *екзаменаційною комісією*, утвореною наказом ректора.

Примітка. Захист комплексного ДП, як правило, планується і проводиться на одному засіданні ЕК. Дипломники, які виконували комплексний проект, повинні бути повною мірою обізнані з його загальною частиною і готові до запитань членів ЕК не тільки з індивідуальної, а й із загальної частини ДП.

До захисту ДП допускаються студенти, які виконали всі вимоги навчальних планів і програм, подали у встановлений термін дипломний проект, одержали позитивні рецензію та відгук керівника ДП, пройшли процедуру нормоконтролю та перевірки на плагіат, а також попередній захист за встановленим графіком.

В день захисту студент повинен здати відповідальному секретарю ЕК всю документацію по ДП.

Захист ДП проводиться на відкритому засіданні ЕК за участю не менше ніж половини її складу за обов'язкової присутності її голови.

Захист ДП включає в себе усну доповідь дипломника, демонстрацію роботи розробленого програмного засобу, відповіді на запитання членів ЕК, відповіді на зауваження рецензента та керівника ДП.

Процедура захисту ДП є наступною:

- представлення студента та поданих документів секретарем ЕК;
- доповідь студента про сутність проекту (бажано здійснювати доповідь по пам'яті та супроводжувати її демонстрацією слайдів чи відеоматеріалів);
- демонстрація роботи розробленого ПЗ;
- відповіді студента на запитання членів ЕК;
- виступ рецензента або оголошення його рецензії;
- відповіді студента на зауваження рецензента;
- виступ керівника або представлення його відгуку;
- відповіді студента на зауваження керівника;
- обговорення проекту та його захисту студентом і прийняття рішення комісією щодо загальної оцінки ДП.

Процедура захисту ДП протоколюється секретарем ЕК.

Рішення ЕК про оцінку знань та вмінь, виявлених при захисті ДП, а також про присвоєння студенту кваліфікації та видачу диплома, приймається на закритому засіданні ЕК відкритим голосуванням зви-

чайною більшістю голосів членів ЕК. За однакової кількості голосів, поданих за два варіанти оцінки, голос голови комісії є вирішальним.

Примітка. Оцінюванню підлягає особистий внесок дипломника, тому в доповіді слід звернути особливу увагу та чітко вказати, де особисті нароби, а де використано загальнодоступні елементи та компоненти сторонніх розробників (фреймворки, бібліотеки, засоби генерації програмних компонентів, існуючі математичні моделі та алгоритми тощо).

Результати захисту ДП оголошуються у той же день після оформлення відповідних документів та протоколів засідання ЕК.

Студенту, який успішно захистив ДП, рішенням ЕК присуджується відповідний рівень вищої освіти, присвоюється ступінь «бакалавр» і видається документ про вищу освіту державного зразка, а також додаток до диплома європейського зразка.

Студенту, який отримав семестрові оцінки «відмінно» не менше як з 75 % навчальних дисциплін, індивідуальних завдань та практик, передбачених навчальним планом, а з інших навчальних дисциплін, індивідуальних завдань і практик – оцінки «добре», захистив ДП з оцінкою «відмінно», видається документ про вищу освіту **з відзнакою**.

Якщо оцінка захисту ДП є незадовільною, то студент відраховується з університету і отримує **академічну довідку**.

Студент має право на повторний захист ДП упродовж трьох років після відрахування з університету. У цьому випадку, на підставі його заяви та за поданням декана факультету, наказом ректора студент (на контрактних умовах) поновлюється на період дипломного проектування і захисту ДП із зазначенням його теми та прізвища керівника.

У випадках, коли захист ДП та/або якість його виконання визнаються незадовільними, а обсяг недостатнім, ЕК встановлює можливість повторного захисту ДП за тією ж темою за умови її доопрацювання, або виконання проекту за новою темою. Ці умови відмічаються у протоколі засідання ЕК і в наказі ректора на відрахування студента.

Якщо захист ДП не відбувся з поважних причин (про що студент має подати відповідні документи до ЕК), ректор університету може подовжити студенту термін навчання до наступного терміну роботи ЕК, але не більше ніж на один рік.

У випадку незгоди з отриманою оцінкою випускник має право на **апеляцію**. Порядок подання апеляції студентом та її розгляду апеляційною комісією регламентується «Положенням про атестацію здобувачів вищої освіти у Хмельницькому національному університеті».

ДП після захисту передається до архіву університету.

8 КРИТЕРІЇ ОЦІНЮВАННЯ

Відповідно до «Положення про контроль і оцінювання навчальних досягнень студентів» університету ДП оцінюється за національною чотирибальною шкалою та шкалою ЄКТС.

Система оцінювання ДП спирається на такі параметри – оцінку якості: змісту пояснювальної записки; її оформлення; розробленого студентом програмного засобу, а також захисту ДП.

Критерії оцінювання якості змісту пояснювальної записки:

- актуальність теми і практична значущість проекту;
- відповідність змісту темі ДП;
- відповідність виконаної роботи завданню на проектування;
- об'єктивність висвітлення стану питання з творчим використанням сучасних джерел інформації;
- повнота дослідження предметної області;
- чіткість і повнота постановки задачі;
- наявність нових ідей та рішень;
- обґрунтованість вибору методів та засобів вирішення поставленої задачі;
- рівень проектних і програмних рішень та їх обґрунтування;
- застосування сучасних технологій та мов програмування;
- наочність та якість ілюстративного матеріалу;
- ступінь самостійності студента;
- наявність/відсутність дублювання, описового матеріалу, стереотипних рішень, що не впливають на суть отриманих результатів.

Критерії оцінювання оформлення пояснювальної записки:

- відповідність оформлення чинним стандартам;
- органічний зв'язок текстового матеріалу з графічним;
- загальна та професійна грамотність, лаконізм і логічна послідовність викладення матеріалу.

Критерії оцінювання якості програмного засобу:

- працездатність та функціональна придатність ПЗ;
- відповідність ПЗ поставленій задачі та специфікації вимог;
- простота та зручність інтерфейсу;
- можливості впровадження програмного засобу.

Критерії оцінювання якості захисту ДП:

- якість і повнота доповіді при захисті ДП: відповідність доповіді темі і меті проекту; володіння матеріалом, послідовність, логіка, грамотність викладення матеріалу; уміння аргументовано обґрунтувати прийняті в проекті рішення, коротко пояснити призначення і роботу розробленого ПЗ, робити висновки тощо;

– правильність і повнота відповідей на питання при захисті ДП: уміння сформулювати аргументовану відповідь на питання, відпо-

відати на нестандартні (проблемні) питання, обґрунтувати власну позицію у проблемних ситуаціях.

При оцінюванні ДП враховують також оцінки керівника проекту та рецензента.

Оцінку **«відмінно»** студент отримує, якщо він виконав ДП у повному обсязі, з дотриманням всіх вимог, а при захисті показав: грамотний, логічний виклад доповіді, правильні та повні відповіді на питання (у т.ч. нестандартні); глибоке і повне опанування змісту навчального матеріалу; уміння пов'язувати теорію з практикою, обґрунтовувати свої судження, робити висновки; володіння різносторонніми навиками, прийомами і компетенціями. Пояснювальна записка повністю відповідає вимогам до її змісту та оформлення і розкриває всі положення проекту. Розроблений програмний продукт відповідає ТЗ і є повнофункціональним; використано сучасні засоби розробки. Веб-додаток та мобільний додаток завантажені на репозиторій GitHub із вказівкою посилання. Мобільний додаток переданий в Google Play чи AppStore для публікації.

Оцінка **«добре»** виставляється студенту у разі, коли він виконав ДП у повному обсязі, з дотриманням вимог, а при захисті демонструє тверде знання матеріалу проекту, грамотно і за суттю викладає його, не допускає суттєвих неточностей у відповідях на питання, правильно застосовує теоретичні положення при вирішенні практичних завдань, володіє необхідними навичками і прийомами їх виконання. Пояснювальна записка в достатній мірі відповідає вимогам і розкриває ключові положення проекту. Розроблений програмний продукт відповідає ТЗ, виконує основні функції; використано сучасні засоби розробки.

Оцінку **«задовільно»** заслуговує студент, який виконав ДП за завданням, але припустився неточностей при виконанні; при захисті виявив знання основного матеріалу в обсязі, необхідному для професійної діяльності; засвоїв і набув практичних навичок у галузі, в основному справляється з виконанням практичних завдань, але допускає порушення логічної послідовності у викладі матеріалу, помилки у відповідях на питання, відчуває труднощі при відповідях на видозмінені питання. Пояснювальна записка переважно відповідає вимогам і розкриває більшість положень ДП. Розроблене ПЗ виконує більшість необхідних функцій або його реалізація виконана у спрощеному вигляді (наприклад, сайт-візитівка).

Оцінка **«незадовільно»** виставляється, коли студент неякісно виконав ДП, а при захисті показав безсистемні знання, не вміє виділяти головне і другорядне, припускається помилок у визначенні понять, перекручує їх зміст, хаотично і невпевнено викладає матеріал, не може застосовувати знання при вирішенні практичних завдань. Пояснювальна записка не відповідає вимогам, недостатньо розкриває положення проекту. Розроблений програмний продукт виконує недостатню кількість функцій або не відповідає ТЗ або виходить за межі теми ДП.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Стандарт вищої освіти України за спеціальністю 121 «Інженерія програмного забезпечення» для першого (бакалаврського) рівня вищої освіти. – Київ : МОН України, 2018. – 24 с.
2. Освітньо-професійна програма підготовки бакалавра зі спеціальності «Інженерія програмного забезпечення» [Електронний ресурс] / Веб-сайт ХНУ. – Режим доступу: <http://www.khnu.km.ua/root/page.aspx?l=0&r=50&p=5&f=%D0%91>. – Назва з екрана.
3. Система внутрішнього забезпечення якості освітньої діяльності : зб. нормат. документів / упоряд.: В. І. Бегняк, Г. В. Красильникова. – Хмельницький : ХНУ, 2015. – 445 с.
4. Положення про атестацію здобувачів вищої освіти у Хмельницькому національному університеті від 29.01.2015 р. [Електронний ресурс] / Веб-сайт ХНУ. – Режим доступу: <http://www.khnu.km.ua/root/res/700-100-2-1.pdf>. – Назва з екрана.
5. Положення про дотримання академічної доброчесності в Хмельницькому національному університеті [Електронний ресурс] / Веб-сайт ХНУ. – Режим доступу: <https://www.khnu.km.ua/root/files/01/06/03/0005.pdf>. – Назва з екрана.
6. ДСТУ 3008:2015. Інформація та документація. Звіти у сфері науки і техніки. Структура та правила оформлювання. – Київ : ДП «УкрНДНЦ», 2016. – 26 с.
7. ДСТУ ГОСТ 7.1:2006. Система стандартів з інформації, бібліотечної та видавничої справи. Бібліографічний запис. Бібліографічний опис. Загальні вимоги та правила складання. – На заміну ГОСТ 7.1–84, ГОСТ 7.18–79, ГОСТ 7.34–81, ГОСТ 7.40–82 ; чинний від 2007–07–01. – Київ : Держспоживстандарт України, 2007. – 47 с.
8. ДСТУ 8302:2015. Інформація та документація. Бібліографічне посилання. Загальні положення та правила складання. – Уведено вперше ; чинний від 2016–07–01. – Київ : ДП «УкрНДНЦ», 2016. – 17 с.
9. ДСТУ 3582:2013. Інформація та документація. Бібліографічний опис. Скорочення слів і словосполучень українською мовою. Загальні вимоги та правила. – На заміну ДСТУ 3582–97 ; чинний від 2013–08–22. – Київ : Мінекономрозвитку України, 2014. – 15 с.
10. Текстові документи. Загальні вимоги. СОУ 207.01:2017 / Ю. Бойко, Г. Красильникова, Л. Першина, Т. Косянчук. – Хмельницький : ХНУ, 2017. – 45 с.
11. Дипломна робота магістра : Методичні вказівки щодо її виконання для студентів спеціальності «Інженерія програмного забезпечення» / уклад. Г. І. Радельчук. – Хмельницький : ХНУ, 2017. – 80 с.
12. Комплексний курсовий проект : методичні вказівки щодо його виконання для студентів спец. «Інженерія програмного забезпечення» / уклад. Г.І. Радельчук. – Хмельницький : ХНУ, 2019. – 70 с.

ДОДАТКИ

ДОДАТОК А
(довідковий)

ФОРМА ТИТУЛЬНОГО АРКУША

Хмельницький національний університет
Факультет програмування
та комп'ютерних і телекомунікаційних систем
Кафедра інженерії програмного забезпечення

ДИПЛОМНИЙ ПРОЕКТ

Бакалавр

Освітній рівень

Назва теми

Галузь знань 12 «Інформаційні технології»
Спеціальність 121 «Інженерія програмного забезпечення»
Освітня програма Освітньо-професійна програма
«Інженерія програмного забезпечення»

Шифр _____

Виконав: студент ___ курсу, група _____

Підпис, дата Ініціали, прізвище

Керівник

Підпис, дата Ініціали, прізвище

Нормоконтролер

Підпис, дата Ініціали, прізвище

До захисту допускаю:

Зав. кафедри інженерії
програмного забезпечення

Підпис, дата Ініціали, прізвище

_____ 20__ р.

Хмельницький 20__

ДОДАТОК Б
(довідковий)

ШАБЛОН ЗАВДАННЯ НА ДИПЛОМНИЙ ПРОЕКТ

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет Програмування та комп'ютерних і телекомунікаційних систем

Кафедра Інженерії програмного забезпечення

Освітній рівень бакалавр

Галузь знань 12 «Інформаційні технології»

Спеціальність 121 «Інженерія програмного забезпечення»

Освітня програма Освітньо-професійна програма

«Інженерія програмного забезпечення»

ЗАТВЕРДЖУЮ:

Завідувач кафедри _____

_____ 20__

**ЗАВДАННЯ
НА ДИПЛОМНИЙ ПРОЕКТ**

Прізвище, ім'я, по батькові студента

1 Тема проекту _____

Керівник проекту _____

Прізвище, ім'я, по батькові, науковий ступінь, вчене звання

Затверджено наказом ректора університету від ____ ____ 20__ р. № ____

2 Строк подання студентом проекту на кафедру: _____

3 Вихідні дані до проекту _____

4 Зміст пояснювальної записки (перелік питань, які потрібно розробити)

5 Перелік графічного матеріалу (із зазначенням обов'язкових креслень)

Продовження ДОДАТКА Б

6 Консультанти розділів дипломного проекту

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв

7 Дата видачі завдання _____ 20__ р.

КАЛЕНДАРНИЙ ПЛАН

Назва етапів (розділів) дипломного проекту	Строк виконання етапів проекту	Примітка
1		
2		
3		
...		

Студент

Підпис

Ініціали, прізвище

Керівник проекту

Підпис

Ініціали, прізвище

ДОДАТОК В
(довідковий)

ЗРАЗОК «АНОТАЦІЇ»

АНОТАЦІЯ

Тема дипломного проекту: «Програмне забезпечення підтримки прийняття рішень при просуванні на ринок програмних продуктів».

Автор проекту: Якобчук Василь Семенович.

Керівник проекту: Яшина Оксана Миколаївна.

Пояснювальна записка: 60 с., 12 рис., 5 табл., 4 дод., 15 джерел.

Графічна частина: 15 презентаційних слайдів.

КОРПОРАТИВНИЙ РИНОК, ПРОГРАМНИЙ ПРОДУКТ, ІНТЕРНЕТ-МАРКЕТИНГ, СИСТЕМА ПРИЙНЯТТЯ РІШЕНЬ, PHP, APACHE, MYSQL.

Метою проекту є розробка комплексу моделей та програмного забезпечення підтримки прийняття рішень при просуванні тиражних програмних продуктів на корпоративному ринку суб'єктів малого і середнього бізнесу з використанням інструментів Інтернет-маркетингу.

У дипломному проекті визначена специфіка маркетингової діяльності ІТ-компаній, що спеціалізуються на розробці і тиражуванні власних програмних продуктів; проведено аналіз методів та інструментів просування тиражних ПП на корпоративному ринку; розроблена функціональна модель організації процесу просування ПП на ринок з урахуванням специфіки використання мережі Інтернет як основного каналу комунікацій; розроблено моделі та алгоритми підтримки прийняття рішень на різних стадіях процесу просування ПП на корпоративному ринку з використанням інструментів Інтернет-маркетингу.

Для реалізації програмної системи використано мову програмування PHP, сервер бази даних MySQL і веб-сервер Apache.

В результаті проектування здійснена програмна реалізація системи підтримки прийняття рішень при організації просування ПП на корпоративному ринку, а також проведено практичну апробацію отриманих результатів та впровадження програмного комплексу.

Підпис студента

Дата

ДОДАТОК Г
(довідковий)

ПРИКЛАД «ПЕРЕЛІКУ СКОРОЧЕНЬ»

ПЕРЕЛІК СКОРОЧЕНЬ

АРМ	–	автоматизоване робоче місце
БД	–	база даних
ЕОД	–	електронний обмін даними
ЖЦ	–	життєвий цикл
ЄСПД	–	єдина система програмної документації
ІКТ	–	інформаційно-комунікаційні технології
КІС	–	корпоративна інформаційна система
МАС	–	мультиагентна система
МПЗ	–	модель подання знань
ОС	–	операційна система
ТЗ	–	технічне завдання
ACPI	–	Advanced Configuration and Power Interface
BIOS	–	Basic Input-Output System
CUDA	–	Compute Unified Device Architecture
ISDN	–	Integrated Services Digital Network
FAT	–	File Allocation Table
HTML	–	HyperText Markup Language
OLE	–	Object Linking and Embedding)
USB	–	Universal Serial Bus
WWW	–	World Wide Web

ДОДАТОК Д
(довідковий)

УЗАГАЛЬНЕНИЙ ЗРАЗОК «ВСТУПУ»

(на прикладі теми «Програмне забезпечення підтримки прийняття рішень при просуванні на ринок програмного продукту»)

Аналіз української та зарубіжної науково-технічної літератури за темою дипломного проекту показав, що на сьогодні практично розв'язані наступні задачі: Однак, ще існує достатньо задач у галузі інженерії програмного забезпечення, на розв'язання яких спрямовані зусилля науковців та інженерів. Зокрема, слід відзначити наступні задачі

На сьогодні світові тенденції розвитку ... систем пов'язані з розв'язанням наступних задач:

Актуальність теми дипломного проекту полягає у

Мета проекту – розробити програмне забезпечення для підтримки прийняття рішень при просуванні тиражних програмних продуктів на корпоративному ринку суб'єктів малого і середнього бізнесу з використанням інструментів Інтернет-маркетингу.

Завдання, які необхідно вирішити для досягнення мети:

- визначити специфіку маркетингової діяльності ІТ-компаній, що спеціалізуються на розробці і тиражуванні власних ПП;
- провести аналіз методів та інструментів просування тиражних ПП на корпоративному ринку, а також пакетів прикладних програм підтримки прийняття рішень при організації цих процесів;
- розробити функціональну модель організації процесу просування програмного продукту на ринок з урахуванням специфіки використання мережі Інтернет як основного каналу комунікацій;
- розробити моделі та алгоритми підтримки прийняття рішень на різних стадіях процесу просування програмного продукту на корпоративному ринку малих і середніх компаній з використанням інструментів Інтернет-маркетингу;
- виконати програмну реалізацію проекту;
- провести тестові випробування ПЗ.

Розроблюване ПЗ може бути корисним для ІТ-компаній, сферою діяльності яких є розробка та просування на корпоративному ринку тиражних програмних продуктів.

ДОДАТОК Е (довідковий)

ПРИКЛАД ДОКУМЕНТА «ТЕХНІЧНЕ ЗАВДАННЯ»

Введення

Робота виконується в рамках проекту автоматизації управління торговельним підприємством.

1 Підстава для розробки

Підставою для розробки є «Завдання на дипломний проект», затверджене завідувачем кафедри інженерії програмного забезпечення.

Найменування розробки: Автоматизована інформаційна система «Склад оптової торгівлі».

2 Призначення розробки

Автоматизована інформаційна система «Склад оптової торгівлі» призначена для узагальнення інформації про рух та наявність товарів, придбаних для оптової торгівлі.

Користувачами програми є менеджери складу, відділ обліку, відділ прийому та оформлення замовлень.

Придбання товарів від постачальників здійснюється на підставі договорів купівлі-продажу, в яких обговорюються умови поставки. Дані первинних документів по приходу товарів узагальнюються в журналі надходження товарів, що містить назву прибуткового документа, його дату і номер, коротку характеристику документа, дату реєстрації документа, відомості про товари, що надійшли.

Оформлення та облік реалізації товарів залежать від способу розрахунку за придбані товари між покупцем і продавцем. Товари реалізуються за готівковий і безготівковий розрахунок. Менеджер веде журнал обліку відпуску товарів, де вказується: номер за порядком, дата відпуску товару, найменування товарів, кількість одиниць і сума відпуску, прізвище, ініціали та підпис особи, яка видала товар.

Дані первинних документів фіксуються в картках обліку, які виконують роль реєстрів складського та бухгалтерського обліку. Бухгалтер не рідше ніж один раз на тиждень здійснює перевірку записів в книгах або картках. Підставою для відображення в бухгалтерському обліку операцій з надходження та вибуття товарів є товарні звіти з доданими до них документами.

3 Вимоги до програми

3.1 Вимоги до функціональних характеристик

Автоматизована інформаційна система «Склад оптової торгівлі» повинна забезпечувати виконання наступних функцій:

- введення, зберігання, пошук і обробку інформації по приходу і реалізації товарів на складі;
- ведення журналу реєстрації прибуткових і видаткових документів;
- своєчасне отримання інформації про наявність товарів на складі;
- формування звітів, необхідних менеджеру і бухгалтеру, які містять всі дані про надходження та реалізацію товарів.

Нормативно-довідкова інформація автоматизованої інформаційної системи «Склад оптової торгівлі» представлена довідниками контрагентів, номенклатури, одиниць виміру.

Первинні документи для обліку товарів в оптовій торгівлі:

- видаткові накладні, касові і товарні чеки, які містять дату реалізації, перелік товарів, що реалізуються, їх кількість, ціну і загальну суму продажу; видаткові накладні заповнюються на підставі замовлень на товар;
- документи на надходження товарів від постачальника, що містять наступну інформацію: дата надходження товарів на склад, відомості про постачальника, перелік товарів, що надійшли, кількість, ціна і загальна сума.

Вихідними даними може бути:

- звіт про надходження товарів за певний період, що містить відомості про постачальників, перелік товарів, що надійшли, їх кількість, ціну, суму надходження по кожному найменуванню товару та загальну суму надходження;
- звіт про продажі за певний період, що містить перелік реалізованих товарів, їх ціну, кількість, суму продажів по кожному найменуванню проданих товарів та загальну суму реалізації;
- інвентарна картка, яка містить дані про надходження та реалізацію кожного товару.

У програмі необхідно передбачити: можливість коригування налаштувань системи; резервне збереження даних; можливість зміни пароля входу в систему; наявність вбудованої довідкової системи; швидкий пошук необхідних документів та довідкової інформації тощо;

3.2 Вимоги до надійності

Розроблюване ПЗ повинно мати:

- можливість самовідновлення після збоїв (відключення електроживлення, збої в операційній системі тощо);
- парольний захист при запуску програми;
- обмеження несанкціонованого доступу до даних;
- можливість резервного копіювання інформаційної бази;
- розмежування прав користувача;

- виключення несанкціонованого тиражування програми;
- контроль інформації, що вводиться, та блокування некоректних дій користувача при роботі з системою.

3.3 Вимоги до складу та параметрів технічних засобів

Системні вимоги для роботи ПЗ повинні бути наступними: тактова частота процесора – 1200 Гц; обсяг оперативної пам'яті – 64 Мб; обсяг вільного дискового простору – 50 Мб; роздільна здатність монітора 1024×768; наявність пристрою читання компакт-дисків.

3.4 Вимоги до інформаційної та програмної сумісності

Програма повинна працювати в операційних системах Windows XP/7/10. Все звіти, що формуються, повинні мати можливість експортування в редактор електронних таблиць MS Office Excel.

3.5 Вимоги до транспортування та зберігання

Програма поставляється на лазерному носії інформації.

Програмна документація поставляється в електронному та друкованому вигляді.

Умови експлуатації програмного забезпечення збігаються з умовами експлуатації ПК.

3.6 Спеціальні вимоги

Програма повинна мати дружній інтерфейс, розрахований на користувача середньої кваліфікації (з точки зору комп'ютерної грамотності).

З огляду на обсяг проекту завдання передбачається вирішувати поетапно. При цьому модулі ПЗ, створені в різний час, повинні передбачати можливість нарощування системи і бути сумісні один з одним; тому документація на прийняте експлуатаційне ПЗ повинна містити повну інформацію, необхідну для роботи з ним програмістів.

Мова програмування визначається вибором виконавця, при цьому він повинен забезпечувати можливість інтеграції програмного забезпечення з пакетом MS Office 2003/2007.

4 Вимоги до програмної документації

В ході розробки програми повинні бути підготовлені: текст програми, опис програми, програма і методика випробувань, керівництво користувача, техніко-економічне обґрунтування.

5 Техніко-економічне обґрунтування

При виконанні операцій по реєстрації надходження товарів на склад та їх відвантаження зі складу використовується ручна праця, а саме: ведуться книги надходжень і витрат товарів. Очевидно, що використання програми значно скоротить час, що витрачається на реєстрацію товару. Окрім того, на складі для кожної одиниці товару існує інвентарна картка; в неї теж заносяться відомості про надходження, витрачання й залишку даного товару на даний момент часу. Для отримання

мання цих відомостей по конкретному товару потрібно не менше 8–10 хв. З використанням програми витрати часу скорочуються до 1–2 хв.

У кінці кожного місяця відповідальний працівник складу складає звіти про обороти товару на складі та виводить залишок товарів по кожній позиції. На цю операцію йде 1–2 дні, тобто 6–12 год. Формування оборотної відомості в комп'ютері займає 3–4 хв.

Окрім того, передбачається можливість отримання звітів за будь-який період часу. При ручному створенні звітів людиною можуть бути допущені помилки, тоді як правильно складений алгоритм програми, що розробляється, помилки виключає.

Отже, економічний ефект від впровадження автоматизованої інформаційної системи «Склад оптової торгівлі» очікується за рахунок скорочення часу на операції, що виконуються менеджерами, недопущенні помилок при формуванні звітів, збільшення часу на аналіз господарської діяльності тощо.

6 Стадії та етапи розробки

Стадії та етапи розробки базуються на відомостях, наведених у таблиці 2.1.

7 Порядок контролю та приймання

Контроль і приймання розробки здійснюються на основі розробленої методики випробувань. При цьому перевіряється виконання всіх функцій програми. Хід проведення випробувань документується у «Протоколі проведення випробувань». На основі цього протоколу підписується акт приймання-здачі програми в експлуатацію.

Примітка 1. У ТЗ допускається включати додатки, де, за необхідності, наводять: зразки вхідних і вихідних документів та звітів, описи файлів даних тощо; перелік науково-дослідних та інших робіт; моделі, схеми, таблиці, описи, розрахунки тощо; інші джерела розробки.

Примітка 2. Залежно від особливостей розроблюваного ПЗ допускається уточнювати зміст розділів/підрозділів ТЗ, вилучати несуттєві розділи чи підрозділи (наприклад, «Техніко-економічні показники», «Умови експлуатації» тощо), вводити нові розділи або об'єднувати деякі з них.

ДОДАТОК И
(довідковий)

БРАЗОК ОФОРМЛЕННЯ «ЗМІСТУ»

ЗМІСТ

Перелік скорочень (якщо є).....	5
Вступ.....	6
1 Дослідження предметної області та постановка задачі	9
1.1 Змістовний аналіз предметної області, її структурних та функціональних особливостей.....	9
1.2 Назва підрозділу.....
1.2.1 Назва пункту
1.2.2
1.3
2 Проектування програмного забезпечення.....	24
2.1 Назва підрозділу.....
3 Програмна реалізація
4 Тестування програмного забезпечення.....
Висновки.....	64
Перелік джерел посилання.....	66
Додаток А Загальна архітектура додатку.....
Додаток Б Код (лістинг) програми.....
Додаток В Інтерфейсні вікна програмного засобу.....
Додаток К Презентаційні матеріали.....

					ДПШПЗ.003080.01.06.ПЗ					
Змн.	Арк.	№ докум.	Підпис	Дата	Програмна система «Бюро знахідок» з використанням фреймворка Symfony 2			Літ.	Арк.	Аркуше
Виконав	Іванов І. І.									
Керівник	Яшина О. М.							<i>ХНУ, ПЗ-18-1</i>		
Рецензент										
Н. контр.										
Зав. каф.	Бедратюк Л.П.				Пояснювальна записка					

ДОДАТОК К
(довідковий)

ШАБЛОН ФОРМИ 2а

					ДПШПЗ.003080.01.06.ПЗ		Арк.
Зм.	Арк.	№ докум.	Підпис	Дата			

ДОДАТОК Л
(довідковий)

ПРИКЛАДИ БІБЛОГРАФІЧНИХ ОПИСІВ

Л.1 Книги

Спека М. В. Создание Web-сайтов : самоучитель / М. В. Спека. – М. : Издательский дом «Вильямс», 2015. – 288 с.

Пасічник В. В. Організація баз даних та знань / В. В. Пасічник, В. А. Резніченко. – Київ : BHV, 2013. – 384 с.

Коннолли Т. Базы данных. Проектирование, реализация и сопровождение. Теория и практика / Т. Коннолли, К. Бегг ; пер. с англ. – М. : Вильямс, 2017. – 1440 с.

Воробйов П. С. Програмування С# : посіб. для вищої школи / П. С. Воробйов, А. О. Гриценко, К. І. Лісовик ; за заг. ред. П. С. Воробйова. – 2-ге вид. – Київ ; Харків : Вища школа, 2014. – 704 с.

Кучерук І. М. Загальний курс фізики : навч. посіб. У 3 т. Т. 3. Оптика. Квантова фізика / І. М. Кучерук [та ін.]. – Київ : Техніка, 2013. – 520 с.

Л.2 Стаття з книги або серіального видання

Гнатчук Є. Г. Інформаційна технологія подання та опрацювання знань на основі нечіткої логіки в експертних системах діагностування комп'ютерних засобів // Вісник Хмельницького національного університету. – Хмельницький : ХНУ. – 2008. – № 4. – С. 55–61.

Поморова О. В. Опрацювання якісної інформації у нечітких експертних системах діагностування комп'ютерних засобів / О. В. Поморова, Є. Г. Гнатчук // Радіоелектронні і комп'ютерні системи : матеріали Міжнар. наук.-практ. конф. «Дні науки 2005» (Дніпропетровськ, 15–27 квітня 2005 р.). – Дніпропетровськ : Наука і освіта, 2005. – Т. 34. Сучасні інформаційні технології. – С. 13–15.

Л.3 Стандарти

Інформація та документація. Звіти у сфері науки і техніки. Структура та правила оформлювання : ДСТУ 3008:2015. – [Чинний від 2017–07–01] – Київ : ДП «УкрНДНЦ», 2016. – 26 с.

Единая система программной документации. Техническое задание. Требования к содержанию и оформлению : ГОСТ 19.201–78. – [Чинний від 1980–01–01] ; [Прийнято методом підтвердження]. – М. : Изд-во стандартов, 1978. – 4 с.

Л.4 Електронні ресурси

Бібліографічний опис ресурсу віддаленого доступу (бібліографічна база даних)

Електронний каталог Національної парл. біб-ки України [Електронний ресурс] : [база даних містить відом. про вітчизн. та зарубіж. кн., брош., що надходять у фонд НІБ України]. – Електронні дані (803 438 записів). – Київ : Нац. парлам. б-ка України, 2002–2015. – Режим доступу: catalogue.nplu.org (дата звернення: 30.03.2015). – Назва з екрана.

Бібліографічний опис складової частини електронного ресурсу (стаття з електронного журналу, веб-сайту, веб-порталу)

Логинова Л. Г. Сущность результата дополнительного образования детей [Электронный ресурс] / Л. Г. Логинова // Образование: исследовано в мире : междунар. науч.-пед. Интернет-журн. – 2000. – № 15. – С. 263–272. – Режим доступа: <http://www.oim.ru/reader.asp/number15/366> (дата обращения: 22.09.2020). – Назв. с экрана.

Головин А. Преобразование XML в HTML [Электронный ресурс] / А. Головин // Портал «IZone – информация из мира PC». – Режим доступа: <http://www.izone.kiev.ua/articles/xml/18/> (дата обращения: 21.10.2020). – Назв. с экрана.

Махортов С. Д. Основы Internet-технологий для математиков. Введение в XML : Ч. 1. [Электронный ресурс] / С. Д. Махортов. – Воронеж : Изд-во ВГУ, 2002. – 22 с. – Режим доступа: http://www.ict.edu.ru/lib/index.php?a=presdir&c=getForm&r=resDesc&d=1ight&id_res=5362 (дата обращения: 21.01.2020). – Назв. с экрана.

Приклади бібліографічних записів [Електронний ресурс] / Кн. палата України. – Електрон. дані. – Київ : Кн. палата України, 2009. – Режим доступу: <http://www.ukrbook.net> (16.05.2020). – Назва з екрана.

Детальний перелік прикладів бібліографічних описів знаходиться за посиланнями:

- а) Книжкова палата України:
URL: <http://www.ukrbook.net/biblzak.html>
- б) Електронний університет ХНУ:
меню *Довідка, Бібліографічний опис*
- в) IEEE style:
URL: https://ztu.edu.ua/ua/science/files/1_IEEE-style.pdf

ЗМІСТ

Вступ	3
1 Мета та завдання дипломного проектування.....	4
2 Організація дипломного проектування.....	6
3 Тематика дипломного проектування	8
4 Структура та зміст дипломного проекту.....	10
5 Вимоги до оформлення дипломного проекту	39
6 Підготовка до захисту дипломного проекту	53
7 Захист дипломного проекту	57
8 Критерії оцінювання	59
Список використаних джерел	61
Додатки	62