

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

**ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ ЕКОНОМІЧНИЙ УНІВЕРСИТЕТ
ІМЕНІ СЕМЕНА КУЗНЕЦЯ**

ІНЖЕНЕРІЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

**Методичні рекомендації
до виконання курсового проєкту
для здобувачів вищої освіти спеціальності
121 "Інженерія програмного забезпечення"
освітньої програми "Інженерія програмного забезпечення"
першого (бакалаврського) рівня**

**Харків
ХНЕУ ім. С. Кузнеця
2024**

УДК 004.415.2(072.034)

I-62

Укладачі: І. О. Золотарьова
О. М. Беседовський

Затверджено на засіданні кафедри інформаційних систем.
Протокол № 7 від 21.12.2023 р.

Самостійне електронне текстове мережеве видання

Інженерія програмного забезпечення [Електронний ресурс] :
I-62 методичні рекомендації до виконання курсового проекту для
здобувачів вищої освіти спеціальності 121 "Інженерія програмного
забезпечення" освітньої програми "Інженерія програмного забез-
печення" першого (бакалаврського) рівня / уклад. І. О. Золотарьо-
ва, О. М. Беседовський. – Харків : ХНЕУ ім. С. Кузнеця, 2024. – 36 с.

Викладено питання організації курсового проектування. Наведено ви-
моги до структури й оформлення курсового проекту, методичні рекомендації
до розроблення його структурних елементів.

Рекомендовано для здобувачів вищої освіти спеціальності 121 "Інже-
нерія програмного забезпечення" освітньої програми "Інженерія програмного
забезпечення" першого (бакалаврського) рівня.

УДК 004.415.2(072.034)

© Харківський національний економічний
університет імені Семена Кузнеця, 2024

Вступ

У сучасному світі програмне забезпечення стає не лише ключовим складником технологічного прогресу, але й необхідним елементом у різних сферах життя – від промисловості та бізнесу до користування особистими пристроями. Тому важливість розуміння методів та інструментів інженерії програмного забезпечення і володіння ними є безсумнівною.

Виконання курсового проєкту з інженерії програмного забезпечення передбачає глибоке розуміння процесу його розроблення, зокрема аналізу вимог, проєктування, моделювання тощо. Ці рекомендації мають на меті сприяти здобувачам вищої освіти в успішному виконанні курсового проєкту.

Важливим аспектом є системність підходу до розроблення програмного забезпечення. Це означає не лише розуміння окремих кроків у процесі розроблення, але й їх взаємозв'язок та вплив на якість кінцевого продукту. Методичні рекомендації допоможуть здобувачам вищої освіти у систематизації знань та навичок, а також у здійсненні комплексного аналізу й управлінні різними етапами розроблення програмного забезпечення.

Для успішного виконання курсового проєкту з інженерії програмного забезпечення потрібна самостійна робота здобувачів вищої освіти з технічною літературою, сучасними мовами, технологіями та середовищами, спрямованими на поліпшення роботи фахівців на всіх етапах створення програмного продукту.

Курсовий проєкт є важливим етапом підготовки здобувачів вищої освіти. Він передбачає більш глибоке осмислення та закріплення теоретичних знань, здобутих під час вивчення навчальної дисципліни, і вдосконалення практичних навичок розроблення проєктної документації до програмного забезпечення, що має бути створено.

Ці методичні рекомендації встановлюють єдині правила та порядок виконання, оформлення і захисту курсового проєкту.

Результати навчання та компетентності, які формує й закріплює курсовий проєкт, подано в табл. 1.

**Результати навчання та компетентності, які формує
освітній компонент**

Результати навчання	Компетентності, якими має оволодіти здобувач вищої освіти
РН06	ЗК01, ЗК02, СК10, СК11
РН14	ЗК02, ЗК03, СК03, СК04, СК13

Примітка.

РН06. Уміння вибирати та використовувати відповідну задачі методологію створення програмного забезпечення.

РН14. Застосовувати на практиці інструментальні програмні засоби доменного аналізу, проєктування, тестування, візуалізації, вимірювань та документування програмного забезпечення.

ЗК01. Здатність до абстрактного мислення, аналізу та синтезу.

ЗК02. Здатність застосовувати знання у практичних ситуаціях.

ЗК03. Здатність спілкуватися державною мовою як усно, так і письмово.

СК03. Здатність розробляти архітектури, модулі та компоненти програмних систем.

СК04. Здатність формулювати та забезпечувати вимоги щодо якості програмного забезпечення у відповідності з вимогами замовника, технічним завданням та стандартами.

СК10. Здатність накопичувати, обробляти та систематизувати професійні знання щодо створення і супроводження програмного забезпечення та визнання важливості навчання протягом всього життя.

СК11. Здатність реалізовувати фази та ітерації життєвого циклу програмних систем та інформаційних технологій на основі відповідних моделей і підходів розробки програмного забезпечення.

СК13. Здатність обґрунтовано обирати та освоювати інструментарій з розробки та супроводження програмного забезпечення.

1. Мета й завдання курсового проєктування

Мета курсового проєкту з інженерії програмного забезпечення полягає в наданні здобувачам вищої освіти можливості засвоєння та практичного застосування ключових принципів, методів та інструментів розроблення програмного забезпечення. Крім того, проєкт спрямовано на розвиток навичок аналізу вимог і проєктування архітектури програмного продукту.

Курсовий проєкт також ставить за мету навчити здобувачів вищої освіти ефективно працювати в команді, використовуючи сучасні методи колективної роботи та системи контролю версій. У результаті виконання проєкту здобувачі вищої освіти будуть здатні розробляти вимоги до програмного забезпечення, яке відповідає вимогам замовника і має стабільну архітектуру.

Під час розроблення курсового проєкту здобувач вищої освіти має показати знання:

- предметної області відповідно до постановки завдання;
- процесів розроблення програмного забезпечення;
- методик розроблення програмного забезпечення;
- принципів версійного контролю та колективної роботи.

Здобувач вищої освіти має уміти:

- виконувати постановку завдання та її аналіз;
- аналізувати та формулювати вимоги;
- проєктувати архітектуру системи;
- працювати в команді;
- управляти проєктом;
- комунікувати в процесі розроблення;
- розробляти та оформлювати потрібну документацію.

Робота над курсовим проєктом певною мірою визначає загально-теоретичну та спеціальну підготовку здобувача вищої освіти і в остаточному підсумку готує його до майбутнього виконання більш складного й завершального етапу навчального процесу – дипломного проєктування. Здобувач вищої освіти має розглядати роботу над курсовим проєктом як своєрідну "репетицію" дипломного проєктування.

2. Організація курсового проєктування

Якісне виконання курсового проєкту вимагає чіткої організації роботи здобувача вищої освіти з моменту вибору теми проєкту й до його захисту.

Керівництво курсовим проєктуванням здійснюють викладачі кафедри інформаційних систем. Керівник курсового проєкту рекомендує здобувачу вищої освіти основну літературу, орієнтує його на розроблення необхідних проєктних рішень.

Здобувачу вищої освіти можуть призначити тему курсового проєкту з переліку рекомендованих тем (додаток А). Також йому надають право самостійного вибору теми проєкту. Якщо тему пропонує здобувач вищої освіти, то її слід обговорити й погодити з керівником курсового проєкту.

Матеріали виконаного курсового проєкту в друкованому та електронному варіантах, презентацію доповіді в електронному варіанті здобувач вищої освіти має здати на перевірку керівникові не пізніше визначеного строку.

Захист курсових проєктів організовує кафедра інформаційних систем за затвердженням графіком.

Під час захисту здобувач вищої освіти коротко доповідає про поставлене завдання, ухвалені проєктні рішення, одержані результати та відповідає на запитання. Доповідь супроводжують демонстрацією презентації, яка має містити слайди, визначені здобувачем вищої освіти для більш повного розкриття теми.

3. Структура та обсяг курсового проєкту

Курсовий проєкт складається з тексту та інших обов'язкових матеріалів (схем, діаграм, графіків, таблиць, рисунків тощо), що розробляють відповідно до завдання.

Обсяг курсового проєкту – 20 – 30 друкованих сторінок формату А4 (без додатків).

У табл. 2 наведено структуру курсового проєкту.

Структура курсового проєкту

Структурні елементи пояснювальної записки	Кількість сторінок
Титульний аркуш	1
Реферат	1
Зміст	1
Вступ	1
1 Бізнес-аналіз предметної області <назва предметної області>	4 – 10
1.1 Бізнес-вимоги (Business Requirements)	
1.1.1 Обґрунтування (Background)	
1.1.2 Можливість для бізнесу (Business Opportunity)	
1.1.3 Бізнес-цілі та критерії успіху (Business Objectives and Success Criteria)	
1.1.4 Потреби клієнтів або ринку (Customer or Market Needs)	
1.1.5 Бізнес-ризики (Business Risks)	
1.2 Бачення рішення (Vision of the Solution)	
1.2.1 Заява про бачення (Vision Statement)	
1.2.2 Основні характеристики (Major Features)	
1.2.3 Припущення та залежності (Assumptions and Dependencies)	
1.3 Сфера застосування та обмеження (Scope and Limitations)	
1.3.1 Обсяг початкового випуску застосунку (Scope of Initial Release)	
1.3.2 Обмеження та виключення (Limitations and Exclusions)	
1.4 Бізнес-контекст (Business Context)	
1.4.1 Профілі зацікавлених сторін (Stakeholder Profiles)	
1.4.2 Операційне середовище (Operating Environment)	
2 Специфікація вимог до модуля (системи)	10 – 17
2.1 Аналіз наявних рішень (аналогів, конкурентів)	
2.2 Специфікація функціональних вимог	
2.3 Нефункціональні вимоги	
2.4 Проєктування інтерфейсу користувача (User interfaces)	
2.5 Деталізація нефункціональних вимог. Логічна структура даних (Detailed Non-Functional Requirements. Logical Structure of the Data)	
Висновки	1
Список використаних джерел	1
Додатки	

4. Методичні рекомендації до розроблення структурних елементів курсового проєкту

Загальними вимогами до тексту курсового проєкту є логічна послідовність викладення матеріалу, чіткість і конкретність викладення теоретичних і практичних результатів роботи, суті постановки завдання та мети роботи, методів дослідження, прийнятих рішень, доведеність висновків і обґрунтованість рекомендацій. У тексті проєкту потрібно дотримуватися єдиної термінології. Вона не має бути перевантаженою малоінформативним матеріалом, описом загальновідомих даних, виведенням формул тощо. Слід посилалися на джерела інформації.

Текст курсового проєкту не слід викладати від першої особи, має переважати безособова форма дієслова (наприклад, "обчислено", "знайдено", "досліджено").

Під час викладення матеріалу не використовують:

розмовні звороти;

жаргонні слова та звороти;

різні терміни для позначення одного поняття;

іншомовні слова та терміни за наявності в українській мові рівнозначних слів і термінів;

скорочення слів і словосполучень, крім установлених правилами орфографії та нормативними документами.

Титульний аркуш є першою сторінкою курсового проєкту. Він містить дані, які подають у такій послідовності:

відомості про виконавця роботи;

повна назва документа;

відомості про керівника;

підписи членів комісії;

рік написання.

Приклад титульного аркуша наведено в додатку Б.

До змісту включають: вступ; послідовно перелічені назви всіх розділів, підрозділів та пунктів основної частини роботи; висновки; список використаних джерел; додатки і номери сторінок, які містять початок матеріалу.

Приклад змісту курсового проєкту наведено в додатку В.

У *вступі* потрібно ідентифікувати та сформулювати проблему бізнесу, яка виникла в компанії, обґрунтувати актуальність теми проєкту для розв'язання цієї проблеми на основі розроблюваного модуля або системи; коротко охарактеризувати функціональність модуля або системи. Слід охарактеризувати технічну та програмну платформу розроблення автоматизованого модуля. Потрібно сформулювати мету та завдання проєкту, визначити об'єкт і описати предметну область проєктування. Також слід навести інформацію про засоби проєктування, використані в курсовому проєкті, та можливі галузі застосування отриманих результатів.

Перший розділ "Бізнес-аналіз предметної області <назва предметної області>". Метою розділу є проведення детального аналізу проблеми, яка виникла на об'єкті управління (компанії) під час ведення бізнесу, вибір шляхів її розв'язання.

Підрозділ 1.1 "Бізнес-вимоги (Business Requirements)".

Бізнес-вимоги забезпечують основу та посилання для розроблення всіх детальних вимог. Бізнес-вимоги можна отримати від вищого керівництва замовника або організації-розробника, виконавчого спонсора, менеджера застосунку, відділу маркетингу чи інших осіб, які мають чітке уявлення про те, навіщо проєкт здійснюють, і про кінцеву цінність, яку він надасть як компанії, так і клієнтам.

У пункті 1.1.1 "Обґрунтування (Background)" слід коротко викласти обґрунтування нового застосунку. Подайте загальний опис історії або ситуації, яка приводить до визнання того, що цей застосунок має бути створено.

У пункті 1.1.2 "Можливість для бізнесу (Business Opportunity)" опишіть наявну ринкову можливість або бізнес-проблему, яку розв'язують. Опишіть ринок, на якому буде конкурувати комерційний застосунок, або середовище, у якому використовуватимуть застосунок. Визначте проблеми, які сьогодні неможливо розв'язати без застосунку, і те, як застосунок відповідає ринковим тенденціям або стратегічним напрямкам компанії.

У пункті 1.1.3 "Бізнес-цілі та критерії успіху (Business Objectives and Success Criteria)" опишіть важливі бізнес-цілі застосунку кількісно

та вимірно. Цінність, надана клієнтам, буде описано в пункті 1.1.4, тому цей пункт треба присвятити цінності, наданій бізнесу. Він може включати оцінювання доходів або економії коштів, аналіз рентабельності інвестицій або цільові дати випуску. З'ясуйте, як буде визначено та виміряно успіх цього проекту, і опишіть фактори, які, імовірно, матимуть найбільший вплив на досягнення цього успіху, зокрема ті, які безпосередньо контролюються організацією, а також зовнішні фактори. Установіть вимірювані критерії, щоб оцінити, чи досягнуто бізнес-цілі.

Приклад.

Замовник:

- високі витрати на персонал;
- потреба збільшення кількості відвідувачів.

У пункті 1.1.4 "Потреби клієнтів або ринку (Customer or Market Needs)" опишіть потреби типових клієнтів або сегментів ринку, зокрема потреби, які ще не задовольняються ринком або наявними системами. Опишіть проблеми, із якими сьогодні стикаються клієнти, і які буде (або не буде) розв'язувати новий застосунок, а також те, як клієнти використовуватимуть застосунок. Визначте апаратне та програмне середовище клієнта, у якому має працювати застосунок. Визначте на високому рівні будь-який відомий критичний інтерфейс або вимоги до продуктивності. Уникайте включення будь-яких деталей дизайну чи реалізації. Надайте вимоги у формі пронумерованого списку, щоб можна було простежити більш детальні користувацькі або функціональні вимоги до них.

Приклад.

Клієнти:

- 1) незручна процедура оформлення замовлення;
- 2) довге очікування.

У пункті 1.1.5 "Бізнес-ризики (Business Risks)" узагальніть основні бізнес-ризики, пов'язані з розробленням цього застосунку, як-от конкуренція на ринку, проблеми з часом, прийняття користувачами, проблеми з упровадженням або можливий негативний вплив на бізнес. Оцініть серйозність ризиків і визначте будь-які заходи щодо зменшення ризиків, яких можна вжити.

Проілюструйте бізнес-вимоги за допомогою карти цілей (map of goals (Purpose – Direction – Function) (приклад наведено в додатку Г) та карти впливів (Impact Mapping) (приклад наведено в додатку Д).

Підрозділ 1.2 "Бачення рішення (Vision of the Solution)" визначає довгострокове бачення застосунку, який потрібно створити для досягнення бізнес-цілей. Це бачення забезпечить контекст для ухвалення рішень протягом усього життєвого циклу розроблення застосунку. Бачення не має містити детальні функціональні вимоги або інформацію про планування проєкту.

У пункті 1.2.1 "Заява про бачення (Vision Statement)" подайте стислу заяву про бачення, яка підсумовує мету та наміри нового застосунку та описує, що буде відбуватися, коли користувач буде використовувати цей застосунок. Заява про бачення має відображати збалансований погляд, який задовольнить потреби різноманітних клієнтів, а також потреби організації, що розвивається. Він може бути дещо ідеалістичним, але має ґрунтуватися на реаліях наявних або очікуваних ринків клієнтів, архітектурі підприємства, організаційних стратегічних напрямках, а також обмеженнях витрат і ресурсів (рис. 1).

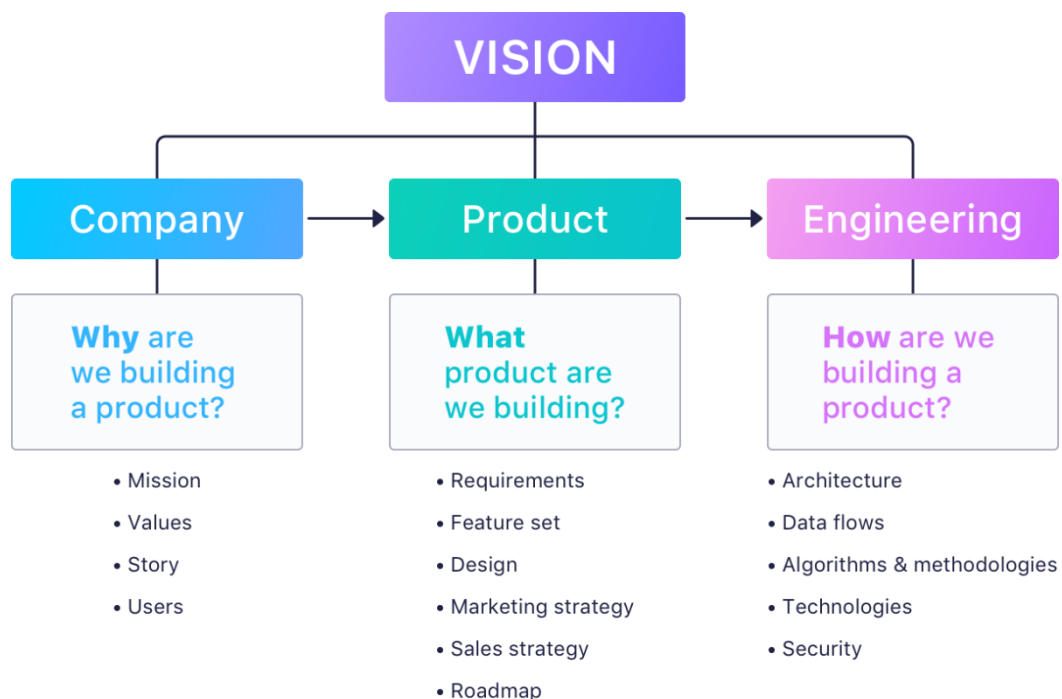


Рис. 1. Структура заяви про бачення

Формула

<Назва застосунку/проєкту> це <категорія застосунку> для <цільова аудиторія>, який <основне завдання> за допомогою <унікальність>.

Замовник планує заробляти/мати іншу вигоду за допомогою [спосіб монетизації].

У пункт 1.2.2 "Основні характеристики (Major Features)" включіть пронумерований перелік основних характеристик нового застосунку, підкреслюючи ті характеристики, які відрізняють його від попередніх застосунків або застосунків-конкурентів. Конкретні вимоги користувача та функціональні вимоги можуть бути пов'язані з цими функціями.

У пункті 1.2.3 "Припущення та залежності (Assumptions and Dependencies)" наведіть будь-які припущення, зроблені під час розроблення проєкту та написання цього документа про бачення й обсяг. Зверніть увагу на будь-які основні залежності, на які проєкт має покладатися для успіху, як-от певні технології, сторонні постачальники, партнери з розроблення тощо.

Підрозділ 1.3 "Сфера застосування та обмеження (Scope and Limitations)". Обсяг проєкту визначає концепцію та діапазон запропонованого рішення. Також важливо з'ясувати, що не входить в застосунок. Уточнення обсягу та обмежень допомагає встановити реалістичні очікування багатьох зацікавлених сторін. Це також забезпечить еталонну систему, за якою можна оцінити запропоновані функції та зміни вимог. Запропоновані вимоги, які виходять за межі передбаченого застосунку, треба відхилити, якщо тільки вони не є настільки вигідними, що їх слід розширити (із супутніми змінами в бюджеті, графіку та/або ресурсах).

У пункті 1.3.1 "Обсяг початкового випуску застосунку (Scope of Initial Release)" визначте заплановані основні функції, які буде включено в початковий випуск застосунку. Опишіть переваги, які застосунок має надати різним групам споживачів, а також характеристики застосунку та характеристики якості, які дозволять йому надати ці переваги. Уникайте включення кожної можливої функції, яка колись може знадобитися будь-якій категорії потенційних клієнтів. Слід зосередитися на тих функціях і характеристиках застосунку, які забезпечать найбільшу цінність за найприйнятнішу вартість розроблення для найширшого кола користувачів.

У пункті 1.3.2 "Обмеження та виключення (Limitations and Exclusions)" визначте будь-які функції або характеристики застосунку, які зацікавлена сторона може передбачити, але які не планують включити в новий застосунок: фінансові, тимчасові, законодавчі, технічні.

У підрозділі 1.4 "Бізнес-контекст (Business Context)" підсумовують бізнес-проблеми, пов'язані з проєктом, зокрема профілі основних кате-

горій клієнтів, припущення, які увійшли в концепцію проекту, і пріоритети управління проектом.

Потрібно подати глосарій у формі таблиці (табл. 3).

Таблиця 3

Глосарій

Термін	Визначення терміна
1. Основні поняття й категорії предметної області та проекту	
2. Користувачі системи	

Пункт 1.4.1 "Профілі зацікавлених сторін (Stakeholder Profiles)". Стейкхолдери – це особи, групи чи організації, які беруть активну участь у проекті, на яких впливає його результат або які можуть вплинути на його результат. Профілі зацікавлених сторін ідентифікують клієнтів цього застосунку та інші зацікавлені сторони, а також зазначають їхні основні інтереси в застосунку. Охарактеризуйте клієнтів бізнес-рівня, цільові ринкові сегменти та різні класи користувачів, щоб зменшити ймовірність появи неочікуваних вимог, які не можна задовільнити через обмеження графіка чи масштабу. Для кожної категорії зацікавлених сторін профіль включає основну цінність або переваги, які вони отримують від застосунку, їхнє ймовірне ставлення до застосунку, основні функції та характеристики, що цікавлять, і будь-які відомі обмеження, які треба врахувати.

Приклад глосарію наведено в додатку Е.

Приклади цінностей для зацікавлених сторін:

- підвищення активності;
- скорочене опрацювання інформації;
- економія коштів;
- налагоджені бізнес-процеси;
- автоматизація завдань, які раніше виконували вручну;
- здатність виконувати абсолютно нові завдання чи функції;

- відповідність чинним стандартам або правилам;
- покращена зручність використання або зменшений рівень розчарування, як порівняти з поточними програмами.

Приклад наведення профілів зацікавлених сторін подано в додатку Ж.

У пункті 1.4.2 "Операційне середовище (Operating Environment)" опишіть середовище, у якому використовуватимуть систему, і визначте основні вимоги до доступності, надійності, застосовуваності та цілісності. Ця інформація суттєво вплине на визначення архітектури системи. Розгляньте такі питання:

- Чи географічно розподілені користувачі, чи розташовані близько один до одного? У скількох часових поясах вони знаходяться?
- Коли користувачам у різних місцях потрібен доступ до системи?
- Де генерують та використовують дані? Як далеко одне від одного знаходяться ці місця? Чи слід об'єднувати дані з кількох місць?
- Чи відомий конкретний максимальний час відповіді для доступу до даних, які можуть зберігатися віддалено?
- Чи можуть користувачі терпіти перебої в обслуговуванні, чи безперервний доступ до системи є критичним для роботи їхнього бізнесу?
- Які засоби контролю безпеки доступу та вимоги до захисту даних необхідні?

Другий розділ "Специфікація вимог до модуля (системи)". Метою є розроблення і детальна специфікація вимог до модуля (системи), що розробляють.

Специфікація вимог до модуля (системи) (SRS) має містити достатньо інформації для того, щоб розробники могли завершити створення описаного програмного забезпечення. Вона не тільки містить опис програмного забезпечення, що розробляють, але й його мету: що має робити програмне забезпечення і як воно має працювати.

Документ SRS зазвичай містить такі елементи:

- мету програмного забезпечення, що розробляють;
- загальний опис програмного забезпечення;
- функціональність програмного забезпечення, або те, що воно має робити;
- застосовуваність програмного забезпечення у виробничій ситуації;
- нефункціональні вимоги;

зовнішні інтерфейси або те, як програмне забезпечення буде взаємодіяти з обладнанням або іншим програмним забезпеченням, до якого воно має підключатися;

обмеження проєктування чи обмеження середовища, у якому буде працювати програмне забезпечення.

У підрозділі 2.1 "Аналіз наявних рішень (аналогів, конкурентів)" потрібно навести коротку характеристику аналогів / конкурентів (не менше ніж три-п'ять).

Порівняльний аналіз проводять за критеріями (табл. 4).

Таблиця 4

Порівняльний аналіз наявних рішень за критеріями

	Аналог 1	Аналог 2	...	Аналог 5
Критерій 1				
Критерій 2				
...				
Критерій 6				

Порівняльний аналіз слід завершити висновком щодо переваг / недоліків, а також потреби розроблення власного рішення.

Підрозділ 2.2 "Специфікація функціональних вимог". Функціональні вимоги до системи, що розробляють – це вимоги до програмного забезпечення, які описують внутрішню роботу системи, її поведінку: калькулювання даних, маніпулювання даними, оброблення даних та інші специфічні функції, що має виконувати система. На відміну від нефункціональних вимог, які визначають, якою система має бути, функціональні вимоги визначають, що система має робити.

Розроблення варіантів використання містить:

діаграму варіантів використання;

специфікацію варіантів використання.

Діаграма варіантів використання відображає функціональність, яку буде реалізовано в програмному застосунку. Варіант використання

можна розглядати як функцію, що реалізовується системою. Однак будь-яка функція повинна мати цінність і давати змогу отримати кінцевий результат для кінцевого користувача застосунку або послуги. Тому під час специфікації варіанта використання серед усієї функціональності системи виділяють лише ту функціональність, яка:

корисна конкретному кінцевому користувачеві;

дозволяє користувачеві отримувати конкретні закінчені результати.

Специфікація варіантів використання.

Кожен варіант використання повинен мати опис. У проекті слід навести опис варіантів використання, що реалізують основну функціональність (зазвичай, крім ведення довідників) у формі таблиці (табл. 5).

Таблиця 5

Специфікація варіантів використання

Назва варіанта використання	
Короткий опис	
Дійові особи	
Передумова	
Основний потік подій (сценарій)	1. 2. ...
Альтернативний потік подій	1. 2. ...
Постумова	
Виключення	

До кожного варіанта використання треба надати діаграму діяльності (Activity).

Підрозділ 2.3 "Нефункціональні вимоги" має містити:

вимоги продуктивності;

вимоги надійності;

вимоги безпеки;

атрибути якості програмного застосунку;

вимоги інтернаціоналізації;
вимоги бази даних.

У підрозділі 2.4 "Проектування інтерфейсу користувача (User Interfaces)" слід розробити графічний інтерфейс користувача із застосунком у формі каркаса (wireframe) або макета (mockup), які використовують для отримання схвалення зацікавлених осіб щодо пропонованої концепції.

У цьому підрозділі потрібно описати:

1. Логіку між користувачами та програмним забезпеченням. Слід приділити увагу детальному опису – від зображень на екрані до макетів програм, від змісту меню до звітів тощо.

2. Способи оптимізації інтерфейсу. Треба перерахувати важливі рекомендації та обмеження для оптимізації інтерфейсу користувача.

Далі слід описати будь-які інтерфейси зв'язку з іншими системами або пристроями, такими як локальні мережі, віддалені послідовні пристрої тощо. Описаний проєкт має підтримувати всі типи веб-браузерів.

Для візуалізації розроблюваного програмного застосунку потрібно додавати до SRS каркаси програми, макети та прототипи (рис. 2).

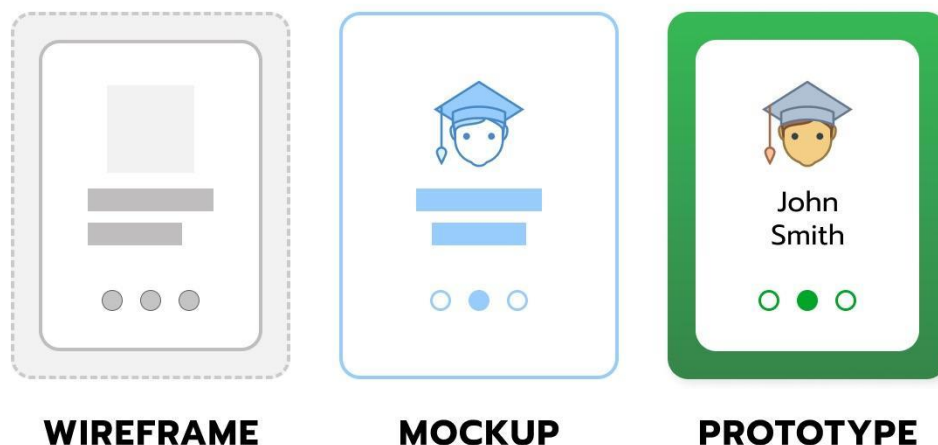


Рис. 2. Приклади каркаса, макета, прототипу

- Каркас (wireframe) – це зображення конструкції з низьким рівнем деталізації. Зазвичай це чорно-білий план сторінки застосунку. У ньому намічено розташування основних елементів: кнопок, зображень, текстів. Він не виконує жодних реальних функцій сайту. Результати взаємодій, кліків, анімацій слід описувати додатково в коментарях. Каркас використовують, щоб визначити, де і який контент буде розміщено.

- Макет (mockup) – це детальне статичне представлення дизайну. Він містить кольори, зображення, типографіку. Макет більше схожий на хороше зображення програми або вебсайту. Макет використовують, щоб створити стиль проекту, представити та погодити візуальні деталі із замовником.

- Прототип (prototype) – це імітація взаємодії користувача з інтерфейсом застосунку, інтерактивна версія каркаса. Прототип використовують, щоб проілюструвати розташування блоків і кнопок та провести тестування зручності використання.

Метою проектування інтерфейсу є отримання ранньої реакції користувачів на пропоновану концепцію системи за допомогою недорогих засобів. У якості інструментальних засобів використовують Marvel, Pencil Project, Use Your Interface, UX Myths, Mobile Patterns, Good UI, Microsoft Visio, Microsoft PowerPoint тощо.

Підрозділ 2.5 "Деталізація нефункціональних вимог. Логічна структура даних (Detailed Non-Functional Requirements. Logical Structure of the Data)". Проектування структури даних застосунку включає проектування бази даних і проектування діаграми класів. Ці два підходи необхідні, оскільки застосунок має зберігати дані.

Отже, обговорення проектування системи буде включати діаграму "сутність-зв'язок" (ER) та діаграму класів.

Після визначення вимог до застосунку, потрібно визначити різні сутності, що беруть у ньому участь. По-перше, це логічна ER-діаграма – діаграма "сутність-зв'язок", у якій сутності мають бути нормалізованими для встановлення належних взаємозв'язків. Також слід зробити детальний опис таблиць бази даних для розроблюваної системи.

Далі треба навести UML-діаграму класів (Class Diagram), що реалізують основну бізнес-логіку програмної системи, та її опис.

Приклади логічної ER-діаграми, опис таблиць бази даних та UML-діаграми класів наведено в додатку І.

Висновки до роботи – це резюме за результатами всієї роботи. Ця частина є особливо важливою, оскільки тут наводять підсумкові результати роботи.

У курсовому проєкті подають висновки за кожним етапом виконаного проєкту та за роботою загалом, які слід співвідносити з метою і завданням на курсове проєктування.

Необхідно зазначити практичну цінність результатів роботи, дати рекомендації для подальшого вдосконалення об'єкта проєктування. Зазначаючи практичну цінність одержаних результатів, важливо окреслити ступінь їх готовності до використання, масштабів використання.

Список використаних джерел має містити відомості про літературні джерела, використані під час розроблення проєкту.

Список літератури – це перелік використаних джерел за темою проєкту у якнайширшому значенні. Тому не слід обмежуватися лише цитованою літературою. У список варто включати всі матеріали, прочитані, переглянуті, проаналізовані під час роботи над курсовим проєктом, які стосуються його теми. Бажано виявляти джерела якомога повніше, пам'ятаючи, що бібліографічний список до проєкту – це підсумок вивчення проблеми і передумова подальших наукових досліджень.

Список використаних джерел має містити не менш ніж 30 позицій. Його подають мовою оригіналу, розміщують в алфавітному порядку прізвищ перших авторів або назв, та нумерують в порядку їх зростання. Нумерація безперервна.

Роботи одного автора розташовують за алфавітом назв або в хронології їх написання. Алфавітний список подають у такій послідовності:

- література зведеного кириличного алфавіту – для джерел мовами з кириличною графікою (українська, болгарська та ін.);

- література мовами з латинською графікою;

- електронні ресурси в тій самій послідовності, що й друковані видання (спочатку кирилицею, а потім латиницею).

Опис джерела обов'язково має містити прізвище та ініціали автора, повну назву джерела, місце видання, назву видавництва та рік видання, кількість сторінок чи посилання на сторінки тощо. Загальний обсяг книжки в сторінках указують, якщо посилання на неї дають повністю, а сторінки (від ... до) зазначають, якщо посилання стосуються окремої частини літературного джерела.

У додатках уміщують матеріал, який є необхідним для повноти роботи, але не може бути послідовно розміщеним у її основній частині через великий обсяг або з інших причин.

Ілюстрації (діаграми бізнес-процесів, сценарії діалогів та ін.), таблиці, проміжні математичні докази, формули та розрахунки, текст допоміжного характеру тощо можна оформити як додатки.

5. Вимоги до оформлення курсового проєкту

Важливе значення під час роботи над курсовим проєктом має його оформлення, до якого висувають певні вимоги.

Під час оформлення тексту курсового проєкту слід керуватися Методичними рекомендаціями до оформлення звітів, курсових проєктів та дипломних робіт (проєктів) [1].

6. Критерії оцінювання роботи

Курсовий проєкт не повинен містити академічного плагіату, фальсифікації, фабрикації. Курсовий проєкт обов'язково перевіряють на наявність академічного плагіату за допомогою технічних засобів.

Курсовий проєкт перевіряють на унікальність. Порядок перевірки та відсоток унікальності визначають протоколом засідання кафедри. Результати перевірки є задовільними, якщо значення оцінок подібності нижче критичних значень метрик, отриманих під час перевірки.

За порушення академічної доброчесності здобувачі вищої освіти відповідно до Закону України "Про освіту" та Кодексу академічної доброчесності ХНЕУ ім. С. Кузнеця [4] мають бути притягнені до академічної відповідальності.

Курсовий проєкт оцінюють за 100-бальною шкалою.

Результат у балах за курсовий проєкт становить: "60 і більше балів – зараховано", "59 і менше балів – не зараховано".

Оцінка за курсовий проєкт складається з двох частин: виконання курсової роботи (до 70 балів) і захисту (до 30 балів).

Оцінку "відмінно" (90 – 100 балів) ставлять, якщо здобувач вищої освіти:

- 1) показав глибокі теоретичні знання тієї теми, за якою виконано курсовий проєкт;
- 2) оволодів сучасними інструментами та методологіями, необхідними в інженерії програмного забезпечення;
- 3) провів детальний аналіз та надав свої пропозиції і рекомендації за вибраною темою;
- 4) виконав роботу грамотно літературною українською мовою;
- 5) оформив курсовий проєкт відповідно до вимог і подав його до захисту у визначений кафедрою термін;
- 6) на захисті продемонстрував глибокі знання вибраної теми, основ інженерії програмного забезпечення, твердо і впевнено відповів на запитання членів комісії.

Оцінку "добре" (74 – 89 балів) ставлять, якщо здобувач вищої освіти:

- 1) показав досить високі теоретичні знання тієї теми, на яку виконано курсовий проєкт;
- 2) оволодів сучасними інструментами та методологіями, необхідними в інженерії програмного забезпечення, але не завжди правильно їх використовує в своєму проєкті;
- 3) провів детальний аналіз та надав свої пропозиції і рекомендації за вибраною темою, однак відчуває труднощі щодо їх обґрунтування;
- 4) виконав курсовий проєкт грамотно літературною українською мовою, але допустив нечисленні граматичні та стилістичні помилки;
- 5) оформив курсовий проєкт відповідно до вимог і подав його до захисту у визначений кафедрою термін;
- 6) на захисті продемонстрував добрі знання з вибраної теми, основ інженерії програмного забезпечення, відповів на запитання членів комісії.

Оцінку "задовільно" (60 – 73 бали) ставлять, якщо здобувач вищої освіти:

- 1) показав достатні теоретичні знання тієї теми, на яку виконано проєкт;
- 2) в основному оволодів сучасними інструментами та методологіями, необхідними в інженерії програмного забезпечення, однак допускає порушення принципів їх використання;

3) не може сформулювати пропозиції та рекомендації з теми дослідження, або обґрунтувати їх;

4) допустив помилки в оформленні курсового проєкту та/або подав його до захисту з запізненням по термінам, визначеним кафедрою;

5) допустив численні граматичні та стилістичні помилки;

6) на захисті продемонстрував задовільні знання з вибраної теми, основ інженерії програмного забезпечення, але не зумів впевнено й чітко відповісти на додаткові запитання членів комісії.

Оцінку "незадовільно" (менше 60 балів) ставлять у тому разі, якщо на захисті здобувач вищої освіти виявив повне незнання вибраної теми, не зумів задовільно відповісти на поставлені запитання, що свідчить про несамотійне виконання курсового проєкту.

Рекомендована література

Основна

1. Методичні рекомендації до оформлення звітів, курсових проєктів та дипломних робіт (проєктів) для здобувачів вищої освіти спеціальностей 121 "Інженерія програмного забезпечення", 122 "Комп'ютерні науки", 126 "Інформаційні системи і технології" [Електронний ресурс] / уклад. І. О. Ушакова, Г. О. Плеханова, О. М. Беседовський. – Харків : ХНЕУ ім. С. Кузнеця, 2021. – 45 с. – Режим доступу : <http://repository.hneu.edu.ua/handle/123456789/27413>.

Додаткова

2. Бородкіна І. Л. Інженерія програмного забезпечення / І. Л. Бородкіна, Г. О. Борожкін. – Київ : Центр навч. літ., 2020. – 204 с.

3. Інженерія програмного забезпечення [Електронний ресурс] : робоча програма навчальної дисципліни для здобувачів вищої освіти спеціальності 121 "Інженерія програмного забезпечення" освітньої програми "Інженерія програмного забезпечення" першого (бакалаврського) рівня / уклад. І. О. Золотарьова. – Харків : ХНЕУ ім. С. Кузнеця, 2024. – 13 с. – Режим доступу : <http://repository.hneu.edu.ua/handle/123456789/32174>.

4. Кодекс академічної доброчесності ХНЕУ ім. С. Кузнеця. – Харків : ХНЕУ ім. С. Кузнеця, 2019. – 10 с. – Режим доступу : <https://www.hneu.edu.ua/wp-content/uploads/2019/11/Kodeks-akademichnoyi-dobrochesnosti.pdf>.

Інформаційні ресурси

5. Інженерія програмного забезпечення (6.121.010): курсовий проєкт // Сайт ПНС ХНЕУ ім. С. Кузнеця. – Режим доступу : <https://pns.hneu.edu.ua/course/view.php?id=6597>.

6. Guide to Software Requirements Specification (SRS) Document 2022 [Electronic resource]. – Access mode : <https://youteam.io/blog/guide-to-software-requirements-specification-srs-document/>.

7. How to Write a Software Requirements Specification (SRS Document) [Electronic resource]. – Access mode : <https://www.perforce.com/blog/alm/how-write-software-requirements-specification-srs-document>.

8. Software Requirements Specification document with example [Electronic resource]. – Access mode : <https://krazytech.com/projects/sample-software-requirements-specificationsrs-report-airline-database>.

9. The Airline Ticket Booking System Example [Electronic resource]. – Access mode : <https://flylib.com/books/en/2.96.1.35/1/>.

Додатки

Додаток А

Перелік рекомендованих тем курсових проєктів

Розумні міста

1. Система моніторингу та оптимізації вуличного освітлення:

Розроблення програмного забезпечення для збирання даних про освітленість вулиць, датчики руху та прогнозування потреб в освітленні.

Створення алгоритмів для динамічного керування освітленням із метою економії енергії та поліпшення видимості.

2. Система моніторингу та управління паркувальними місцями:

Розроблення мобільного додатка для пошуку вільних паркувальних місць у реальному часі.

Створення системи датчиків для відстеження зайнятості паркувальних місць.

3. Система моніторингу та прогнозування забруднення повітря:

Розроблення програмного забезпечення для збирання даних про якість повітря з датчиків, розташованих у місті.

Створення алгоритмів для прогнозування забруднення повітря та оповіщення жителів.

4. Система моніторингу та управління муніципальними відходами:

Розроблення мобільного додатка для повідомлення про переповнені сміттєві баки та замовлення вивезення сміття.

Створення системи датчиків для моніторингу рівня заповнення сміттєвих баків.

Карта бомбосховищ

5. Мобільний додаток із картою бомбосховищ:

Розроблення мобільного додатка з інтерактивною картою бомбосховищ, який включає інформацію про їхнє розташування, місткість, доступність та інші характеристики.

Додавання функції пошуку найближчого бомбосховища та прокладання маршруту до нього.

6. Вебсайт із картою бомбосховищ:

Створення вебсайту з картою бомбосховищ, який містить інформацію про їхнє розташування, місткість, доступність тощо.

Додавання функції пошуку бомбосховищ за адресою, районом або іншими критеріями.

7. Система онлайн-реєстрації для отримання соціальних послуг:

Розроблення програмного забезпечення для онлайн-реєстрації людей, які потребують соціальних послуг.

Створення алгоритмів для автоматизованого розподілу ресурсів та підтримання зв'язку з людьми, які потребують допомоги.

8. Система моніторингу та підтримки людей похилого віку:

Розроблення мобільного додатка для моніторингу стану здоров'я та життєдіяльності людей похилого віку.

Створення системи датчиків для відстеження показників здоров'я та повідомлення про надзвичайні ситуації.

9. Система онлайн-навчання для людей з обмеженими можливостями:

Розроблення вебплатформи з онлайн-курсами та інтерактивними матеріалами для людей з обмеженими можливостями.

Створення адаптивного інтерфейсу та доступного контенту для людей із різними типами інвалідності.

10. Система онлайн-платформи для волонтерів:

Розроблення вебплатформи для з'єднання волонтерів з людьми, які потребують допомоги.

Створення алгоритмів для підбору волонтерів на основі їхніх навичок та досвіду.

Інші теми

11. Система моніторингу та управління водними ресурсами міста.

12. Система раннього попередження про стихійні лиха.

13. Система онлайн-голосування для участі в місцевому самоврядуванні.

14. Система моніторингу та поліпшення доступності громадського простору.

15. Система онлайн-платформи для обміну знаннями та досвідом.

16. Система онлайн-платформи для збирання коштів на соціальні проекти.

17. Система моніторингу та поліпшення якості життя в громаді.

Зразок оформлення титульного аркуша

ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ ЕКОНОМІЧНИЙ УНІВЕРСИТЕТ
ІМЕНІ СЕМЕНА КУЗНЕЦЯ
ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
КАФЕДРА ІНФОРМАЦІЙНИХ СИСТЕМ

Курсовий проєкт

з навчальної дисципліни "Інженерія програмного забезпечення"
на тему "Проектування модуля «Назва предметної області»"

Здобувача вищої освіти 3-го курсу
групи 6.04.121.010.21.01
спеціальності 121 "Інженерія програмного
забезпечення"
першого (бакалаврського) рівня
Вареника Г. В.
Керівник: професор кафедри ІС,
канд. екон. наук, доцент Золотарьова І. О.

Національна шкала _____

Кількість балів: _____ Оцінка: ECTS ____

Члени комісії

_____	_____
(підпис)	(прізвище та ініціали)
_____	_____
(підпис)	(прізвище та ініціали)

Харків – 202_

Зразок оформлення змісту курсового проєкту

ЗМІСТ

Вступ	6
1 Бізнес-аналіз предметної області <назва предметної області>	7
1.1 Бізнес-вимоги (Business Requirements).....	7
1.1.1 Обґрунтування (Background).....	8
1.1.2 Можливість для бізнесу (Business Opportunity).....	9
1.1.3 Бізнес-цілі та критерії успіху (Business Objectives and Success Criteria)	10
1.1.4 Потреби клієнтів або ринку (Customer or Market Needs).....	11
1.1.5 Бізнес-ризики (Business Risks).....	12
1.2 Бачення рішення (Vision of the Solution).....	13
1.2.1 Заява про бачення (Vision Statement).....	13
1.2.2 Основні характеристики (Major Features).....	14
1.2.3 Припущення та залежності (Assumptions and Dependencies).....	14
1.3 Сфера застосування та обмеження (Scope and Limitations).....	15
1.3.1 Обсяг початкового випуску застосунку (Scope of Initial Release).....	15
1.3.2 Обмеження та виключення (Limitations and Exclusions).....	15
1.4 Бізнес-контекст (Business Context).....	16
1.4.1 Профілі зацікавлених сторін (Stakeholder Profiles).....	16
1.4.2 Операційне середовище (Operating Environment).....	17
2 Специфікація вимог до модуля (системи).....	18
2.1 Аналіз наявних рішень (аналогів, конкурентів).....	18
2.2 Специфікація функціональних вимог.....	21
2.3 Нефункціональні вимоги.. ..	24
2.4 Проєктування інтерфейсу користувача (User interfaces).....	27
2.5 Деталізація нефункціональних вимог. Логічна структура даних (Detailed Non-Functional Requirements. Logical Structure of the Data)	28
Висновки.....	30
Список використаних джерел.....	31
Додатки	32

Приклад карти цілей



Рис. Г.1. Аналіз вимог бізнесу у формі карти цілей

Приклад карти впливів

Таблиця Д.1

Аналіз вимог бізнесу у формі карти впливів

Навіщо (мета)	Хто (дійові особи)	Як (вплив, дія)	Що (завдання, вимоги)
Зменшити витрати на утримання офіціантів на X	Менеджер	Керувати електронним меню	Керувати базою даних доступних страв
			Перевіряти та підтверджувати замовлення
	Офіціант	Зменшити обсяг виконуваної роботи	Займатися тільки подачею страв та отриманням оплати
	Замовник	Не звертатися до офіціантів для замовлення	Використовувати електронне меню
Збільшити кількість сучасних клієнтів на N	PR-менеджер	Провести PR-кампанію введення електронного меню	Зробити sms- та email-розсилку постійним клієнтам
			Купити рекламу в соціальних мережах

Приклад глосарію

Таблиця Е.1

Глосарій

Термін	Визначення терміна
1. Основні поняття й категорії предметної області та проєкту	
Електронне меню	Вебдодаток, який має доступ до бази даних доступних для замовлення товарів та надає користувачу функціонал для виконання замовлення
Замовлення	Список страв, які вибрав замовник
2. Користувачі застосунку	
Замовник	Людина, яка має доступ до системи для виконання замовлення
Менеджер	Людина, яка має доступ до системи, з особливими правами для керування інформацією

Приклад профілів зацікавлених сторін

Таблиця Ж.1

Профілі зацікавлених сторін

Класи користувачів	Характеристики користувачів
Менеджер	Менеджер – це користувач, який керує базою замовлень. <u>Список можливостей:</u> 1. Додавання інформації про нові страви. 2. Редагування та видалення інформації про наявні страви. 3. Підтвердження замовлень.
Офіціант	Офіціант – це користувач, який відносить готове замовлення замовнику. <u>Список можливостей:</u> 1. Отримання інформації про місцезнаходження замовника. 2. Віднесення готових страв замовнику. 3. Прийняття оплати за замовлення.
Замовник	Замовник – це користувач, який робить замовлення на певні страви. <u>Список можливостей:</u> 1. Ознайомлення зі списком доступних для замовлення страв. 2. Виконання замовлення. 3. Проведення оплати за замовлення.

Приклад оформлення підрозділу з проєктування бази даних

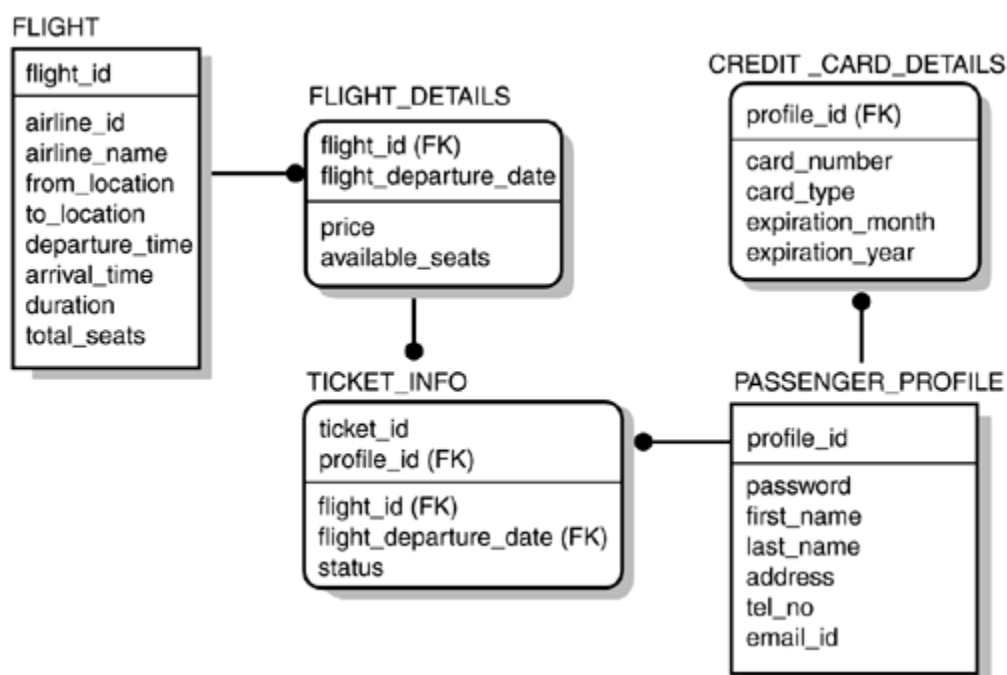


Рис. И.1. Логічна модель БД

Таблиця И.1

Структура таблиці "FLIGHT"

Column Name	Data Type	Remarks
flight_id	NUMBER	Represents the unique identifier for a flight of an airline.
airline_id	NUMBER	Represents the unique identifier for an airline.
airline_name	VARCHAR(20)	Represents the unique identifier for an airline.
from_location	VARCHAR(20)	Represents the location from which the flight begins.
to_location	VARCHAR(20)	Represents the destination of the flight.
departure_time	DATETIME	Represents the time the flight departs every day.
arrival_time	DATETIME	Represents the time the flight arrives every day.
duration	NUMBER	Represents the time taken for the flight.
total_seats	NUMBER	Represents the number of seats on this flight.

Таблиця И.2

Структура таблиці "FLIGHT_DETAILS"

Column Name	Data Type	Remarks
flight_id	NUMBER	Represents the unique identifier for a flight. This is a foreign key.
flight_departure_date	DATETIME	Represents the departure date for the flight.
price	NUMBER	Represents the price of the flight on a specific date.
available_seats	NUMBER	Represents the seats available on a specific date for a flight.

Таблиця И.3

Структура таблиці "PASSENGER_PROFILE"

Column Name	Data Type	Remarks
profile_id	NUMBER	Represents the unique identifier for a passenger. This is a primary key.
password	VARCHAR(20)	Represents the password used by the passenger to authenticate with the application.
first_name	VARCHAR(20)	Represents the first name of the passenger.
last_name	VARCHAR(20)	Represents the last name of the passenger.
address	VARCHAR(20)	Represents the address of the passenger.
tel_no	VARCHAR(20)	Represents the telephone number of the passenger.
email_id	VARCHAR(20)	Represents the e-mail ID of the passenger.

Таблиця И.4

Структура таблиці "TICKET_INFO"

Column Name	Data Type	Remarks
ticket_id	NUMBER	Represents the unique identifier for a ticket.
profile_id	VARCHAR(8)	Represents the unique identifier for a passenger. This is a foreign key.
flight_id	NUMBER	Represents the flight number of the flight booked by the passenger. This is a foreign key.
flight_departure_date	DATETIME	Represents the departure date of the flight booked by the passenger.
status	VARCHAR(10)	Represents the status of the ticket purchased by a passenger.

Таблиця И.5

Структура таблиці "CREDIT_CARD_DETAILS"

Column Name	Data Type	Remarks
profile_id	NUMBER	Represents the unique identifier for a passenger. This is a foreign key.
card_number	VARCHAR(20)	Represents the credit card number used by the passenger to purchase tickets.
card_type	VARCHAR(20)	Represents the credit card type used by the passenger.
expiration_month	VARCHAR(2)	Represents the expiration month of the credit card of the passenger.

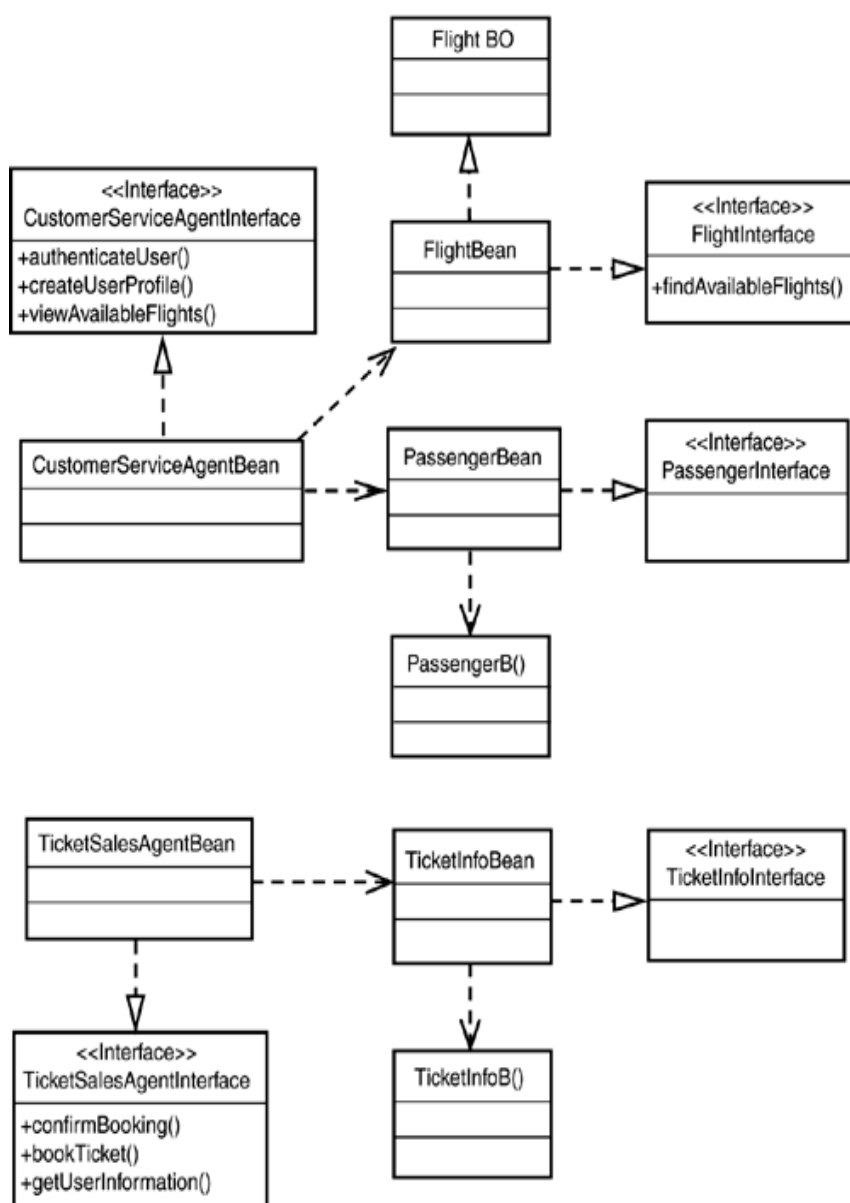


Рис. И.2. UML-діаграма класів

Зміст

Вступ.....	3
1. Мета й завдання курсового проєктування	5
2. Організація курсового проєктування	6
3. Структура та обсяг курсового проєкту	6
4. Методичні рекомендації до розроблення структурних елементів курсвого проєкту.....	8
5. Вимоги до оформлення курсового проєкту.....	20
6. Критерії оцінювання роботи	20
Рекомендована література.....	22
Основна	22
Додаткова	22
Інформаційні ресурси	23
Додатки.....	24

НАВЧАЛЬНЕ ВИДАННЯ

ІНЖЕНЕРІЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

**Методичні рекомендації
до виконання курсового проєкту
для здобувачів вищої освіти спеціальності
121 "Інженерія програмного забезпечення"
освітньої програми "Інженерія програмного забезпечення"
першого (бакалаврського) рівня**

Укладачі: **Золотарьова** Ірина Олександрівна
Беседовський Олексій Миколайович

Відповідальний за видання *Д. О. Бондаренко*

Редактор *Н. Г. Войчук*

Коректор *В. О. Дмитрієва*

План 2024 р. Поз. № 79 ЕВ. Обсяг 36 с.

Видавець і виготовлювач – ХНЕУ ім. С. Кузнеця, 61165, м. Харків, просп. Науки, 9-А

*Свідоцтво про внесення суб'єкта видавничої справи до Державного реєстру
ДК № 4853 від 20.02.2015 р.*