**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**

**Чернігівський національний технологічний університет**

КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА

Організаційно-методичні вказівки до виконання  
випускної кваліфікаційної роботи бакалавра  
за спеціальністю 123 - “Комп’ютерна інженерія”

ЗАТВЕРДЖЕНО

на засіданні кафедри

інформаційних та комп’ютерних систем

протокол № 8 від 27.02.20

**Чернігів ЧНТУ 2020**

Кваліфікаційна робота бакалавра. Організаційно-методичні вказівки до виконання випускної кваліфікаційної роботи бакалавра для студентів спеціальності 123 - “Комп’ютерна інженерія””. /Укл.: Акименко А.М., Бивойно П.Г., Пріла О.А., Роговенко А.І., Стасюк С.С.– Чернігів: ЧНТУ, 2020. – 39 с.

Укладачі:

Акименко Андрій Миколайович, канд. фіз.-мат. наук, доцент

Бивойно Павло Георгійович, канд. техн. наук, доцент

Пріла Ольга Анатоліївна, канд. техн. наук, доцент

Роговенко Андрій Іванович, ст. викладач

Стасюк Сергій Станіславович, ст. викладач

Відповідальний за випуск: В.М. Базилевич, зав. кафедрою інформаційних та комп’ютерних систем, канд. економ. наук, доцент.

Рецензент: Білоус І.В., канд. техн. наук, доцент кафедри інформаційних технологій та програмної інженерії Чернігівського національного технологічного університету

ЗМІСТ

[1 Вимоги до бакалавра 7](#_Toc32348408)

[2 Контрольні точки виконання кваліфікаційної роботи 7](#_Toc32348409)

[3 Вимоги щодо вмісту та оформлення кваліфікаційної роботи 8](#_Toc32348410)

[3.1 Анотація 8](#_Toc32348411)

[3.2 Вступ 9](#_Toc32348412)

[3.3 Основна частина роботи 10](#_Toc32348413)

[3.4 Аналіз задачі створення системи 10](#_Toc32348414)

[3.4.1 Аналіз предметної області(ПО) 10](#_Toc32348415)

[3.4.1.1 Побудова базової моделі предметної області 10](#_Toc32348416)

[3.4.1.2 Вивчення існуючих систем 13](#_Toc32348417)

[3.4.1.3 Уточнення загальної моделі 13](#_Toc32348418)

[3.4.2 Аналіз вимог до інтерфейсу системи 14](#_Toc32348419)

[3.4.3 Аналіз вимог до апаратної частини (якщо є) 14](#_Toc32348420)

[3.4.3.1 Аналіз вимог до локальної обчислювальної мережі 14](#_Toc32348421)

[3.4.3.2 Аналіз вимог до апаратної підсистеми 15](#_Toc32348422)

[3.4.4 Постановка задачі на розробку ІКС (вимоги до системи) 15](#_Toc32348423)

[3.5 Розробка ІКС 16](#_Toc32348424)

[3.5.1 Вибір технічних засобів побудови системи 17](#_Toc32348425)

[3.5.2 Архітектура системи 17](#_Toc32348426)

[3.5.3 Розробка програмної підсистеми 17](#_Toc32348427)

[3.5.4 Розробка програмної підсистеми 18](#_Toc32348428)

[3.5.4.1 Структура програмної підсистеми 18](#_Toc32348429)

[3.5.4.2 Розробка класів сутностей системи 18](#_Toc32348430)

[3.5.4.3 Розробка бази даних 21](#_Toc32348431)

[3.5.4.4 Розробка технології роботи з даними 23](#_Toc32348432)

[3.5.4.5 Розробка алгоритмів взаємодії між компонентами системи 24](#_Toc32348433)

[3.5.4.6 Моделювання виконання операцій 26](#_Toc32348434)

[3.5.4.7 Діаграми класів сутностей, класів бізнес-логіки та допоміжних класів 27](#_Toc32348435)

[3.5.4.8 Розробка WEB-компонент системи 28](#_Toc32348436)

[3.5.5 Розробка апаратної підсистеми 31](#_Toc32348437)

[3.5.5.1 Проектування локальної обчислювальної мережі 31](#_Toc32348438)

[3.5.5.2 Проектування апаратної підсистеми 32](#_Toc32348439)

[3.6 Реалізація системи 34](#_Toc32348440)

[3.6.1 Реалізація програмної підсистеми 34](#_Toc32348441)

[3.6.1.1 Результати реалізації бази даних 35](#_Toc32348442)

[3.6.1.2 Діаграма пакетів 36](#_Toc32348443)

[3.6.1.3 Протоколи класів 36](#_Toc32348444)

[3.6.1.4 Опис процесу автоматизації збірки та установки застосування 37](#_Toc32348445)

[3.6.1.5 Тестування програмного забезпечення 37](#_Toc32348446)

[3.6.1.6 Діаграма розгортання ІКС 37](#_Toc32348447)

[3.6.1.7 Реалізація інтерфейсу користувача 38](#_Toc32348448)

[3.6.2 Реалізація апаратної частини 39](#_Toc32348449)

Вступ

Робота над випускною кваліфікаційною роботою є важливою складовою підготовки бакалавра. На цьому етапі студент вирішує задачі, пов’язані з побудовою комп’ютерної системи, демонструючи цим готовність до практичної роботи.

Дипломна робота студента має засвідчити, що він придатний до виконання виробничих функцій, які відповідають освітньо-кваліфікаційному рівню бакалавр.

Виконання кваліфікаційної роботи починається до початку восьмого семестрі. Перш за все студент має визначитися з керівником та темою роботи. Далі має бути сформовано індивідуальне завдання на виконання кваліфікаційної роботи і визначено перелік задач з виконання кваліфікаційної роботи. Впродовж семестру студент працює над теоретичною частиною роботи, зокрема, знайомиться з предметною областю, працює з інформаційними джерелами, аналізує існуючі підходи до вирішення проблеми та приймає рішення стосовно основних моментів розробки системи.

Під час проходження практики студент визначає кількісні характеристики виробничого процесу, аналізує потоки інформації та документи, уточнює перелік завдань, що мають бути вирішені та вимоги до системи. Результатом проходження практики має бути розділ «Розробка системи» кваліфікаційної роботи, який є звітом з виконання практики.

Під час дипломного проектування студент реалізує деякі елементи системи і займається оформленням результатів.

Державна атестація студента здійснюється Державною екзаменаційною комісією з метою встановлення фактичної відповідності рівня підготовки до вимог освітньо-кваліфікаційної характеристики. Державна екзаменаційна комісія за результатами захисту кваліфікаційної роботи повинна перевірити теоретичну та практичну підготовку випускників, вирішити питання про присвоєння їм освітньо-кваліфікаційного рівня бакалавра і кваліфікації бакалавра комп’ютерної інженерії.

# Вимоги до бакалавра

Бакалавр – це [освітній ступінь](https://uk.wikipedia.org/w/index.php?title=%D0%9E%D1%81%D0%B2%D1%96%D1%82%D0%BD%D1%96%D0%B9_%D1%81%D1%82%D1%83%D0%BF%D1%96%D0%BD%D1%8C&action=edit&redlink=1), що здобувається на першому рівні [вищої освіти](https://uk.wikipedia.org/wiki/%D0%92%D0%B8%D1%89%D0%B0_%D0%BE%D1%81%D0%B2%D1%96%D1%82%D0%B0) і відповідає сьомому рівню [Національної рамки кваліфікацій](https://uk.wikipedia.org/wiki/%D0%9D%D0%B0%D1%86%D1%96%D0%BE%D0%BD%D0%B0%D0%BB%D1%8C%D0%BD%D0%B0_%D1%80%D0%B0%D0%BC%D0%BA%D0%B0_%D0%BA%D0%B2%D0%B0%D0%BB%D1%96%D1%84%D1%96%D0%BA%D0%B0%D1%86%D1%96%D0%B9). Відповідно до вимог цього рівня кваліфікації здобувач освітнього рівня бакалавр має продемонструвати здатність розв’язувати складні спеціалізовані задачі та практичні проблеми під час професійної діяльності в комп’ютерній галузі або навчання, що передбачає застосування теорій та методів комп’ютерної інженерії і характеризується комплексністю та невизначеністю умов

Здобувач освітньо-кваліфікаційного рівня «бакалавр» спеціальності 123 – «Комп’ютерна інженерія» під час захисту випускної кваліфікаційної роботи має продемонструвати відповідність вимогам стандарту вищої освіти [1] до таких результатів навчання, таблиця 1.

|  |  |
| --- | --- |
| Таблиця 1 – Вимоги стандарту вищої освіти до результатів навчання бакалаврів спеціальності 123-“Комп’ютерна інженерія” | |
| Знання | N1. Знати і розуміти наукові положення, що лежать в основі функціонування комп’ютерних засобів, систем та мереж. |
| N2. Мати навички проведення експериментів, збирання даних та моделювання в комп’ютерних системах. |
| N3. Знати новітні технології в галузі комп’ютерної інженерії. |
| N4. Знати та розуміти вплив технічних рішень в суспільному, економічному, соціальному і екологічному контексті. |
| N5. Мати знання основ економіки та управління проектами |
| Уміння | N6. Вміти застосовувати знання для ідентифікації, формулювання і розв’язування технічних задач спеціальності, використовуючи методи, що є найбільш придатними для досягнення поставлених цілей. |
| N7. Вміти розв’язувати задачі аналізу та синтезу засобів, характерних для спеціальності. |
| N8. Вміти системно мислити та застосовувати творчі здібності до формування нових ідей. |
| N9. Вміти застосовувати знання технічних характеристик, конструктивних особливостей, призначення і правил експлуатації програмно-технічних засобів комп’ютерних систем та мереж для вирішення технічних задач спеціальності. |
| N10. Вміти розробляти програмне забезпечення для вбудованих і розподілених застосувань, мобільних і гібридних 9 систем, розраховувати, експлуатувати, типове для спеціальності обладнання. N11. Вміти здійснювати пошук інформації в різних джерелах для розв’язання задач комп’ютерної інженерії. |
| N12. Вміти ефективно працювати як індивідуально, так і у складі команди. |
| N13. Вміти ідентифікувати, класифікувати та описувати роботу комп’ютерних систем та їх компонентів. |
| N14. Вміти поєднувати теорію і практику, а також приймати рішення та виробляти стратегію діяльності для вирішення завдань спеціальності з урахуванням загальнолюдських цінностей, суспільних, державних та виробничих інтересів. |
| N15. Вміти виконувати експериментальні дослідження за професійною тематикою. |
| N16. Вміти оцінювати отримані результати та аргументовано захищати прийняті рішення. |

# Контрольні точки виконання кваліфікаційної роботи

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Етапи роботи |  |  |  | Тижні весняного семестру | | | | | | | | | | | | | | | | |  |  |  |
| -2 | -1 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
|  |  |  | лютий | | | березень | | | | | квітень | | | | травень | | | | червень | | | |
|  |  |  |  | | | | | | | | | зчт | сесія | | практика | | проектування | | | | захист | |
| Визначитися з керівником |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Визначитися з темою роботи |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Індивідуальне завдання |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Знайомство з проблемою, інформаційними джерелами, аналіз існуючих рішень. |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Прийняття рішень стосовно основних моментів розробки |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Звіт з першої частини роботи (ІЗ, Вступ та аналіз задачі) |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Збори з питань практики |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Розробка системи |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Звіт з практики, розділ розробка |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Реалізація системи |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Попередній захист |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Підпис роботи керівником |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Переплетення роботи |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Перевірка на «плагіат» |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Рецензування роботи |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Підпис завідувача кафедрою |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Захист кваліфікаційної роботи |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

# Вимоги щодо вмісту та оформлення кваліфікаційної роботи

Кваліфікаційна робота бакалавра може стосуватися розробки програмної або апаратної частини комп’ютерної системи. Можлива розробка і комбінації обох складових.

Оформлення роботи має відповідати вимогам «Методичних рекомендацій щодо виконання та оформлення випускних кваліфікаційних робіт (проектів) здобувачів вищої освіти освітніх ступенів «бакалавр» і «магістр» Чернігівського національного технологічного університету» [2] зі змінами, що затверджені наказом ректора № 09 від 21.10.2019 [3].

Нижче наведено деякі рекомендації [4] стосовно вмісту деяких структурних розділів роботи

## Анотація

* + 1. Анотація призначена для ознайомлення з роботою. Вона має бути стислою, інформативною і містити відомості, які дозволяють прийняти рішення про доцільність читання всієї роботи.
    2. Анотація має бути розміщена безпосередньо за індивідуальним завданням, починаючи з нової сторінки.
    3. Анотація повинена мати:
* відомості про обсяг роботи, кількість ілюстрацій, таблиць, додатків, кількість джерел згідно з переліком посилань (усі відомості наводять, включаючи дані додатків);
* текст анотації;
* перелік ключових слів.
  + 1. Текст анотації повинен відображати подану у роботі інформацію i, як правило, у такій послідовності:
* об’єкт дослідження або розробки;
* мета роботи;
* методи дослідження та апаратура;
* основні конструктивні, технічні, експлуатаційні характеристики та показники;
* ступінь впровадження;
* взаємозв’язок з іншими роботами;
* рекомендації щодо використання результатів роботи;
* галузь застосування;
* економічна ефективність;
* значимість роботи та висновки;
* прогнозні припущення про розвиток об’єкту дослідження або розроблення.

Частини тексту анотації, щодо яких відсутні відомості, випускають.

* + 1. Анотацію належить виконувати обсягом не більш, як 500 слів, i, бажано, щоб він уміщався на сторінці формату А4.
    2. Ключові слова, що э визначальними для розкриття суті звіту, розміщують після тексту анотації.

## Вступ

* + 1. Вступ розкриває сутність і стан проблеми (задачі) та її значимість, підстави і вихідні дані для розробки, обґрунтування необхідності вирішення задачі.

Далі подають загальну характеристику роботи в рекомендованій нижче послідовності:

* актуальність теми;
* зв'язок роботи з іншими роботами (якщо є);
* мета і завдання розробки;
* об’єкт і предмет розробки;
* методи досліджень і розробки;
* практичне значення одержаних результатів;
* апробація результатів роботи;
* публікації, виступи на конференціях.
  + 1. Актуальність роботи

Актуальність роботи містить положення і докази, які свідчать на користь прикладної значущості виконаної роботи. Студенту слід показати, що існувала проблема, яку необхідно було заповнити з точки зору вирішення проблем практики.

Висвітлення актуальності не повинно бути багатослівним. Досить кількома реченнями висловити головне.

* + 1. Зв'язок роботи з науковими програмами, планами, темами

У цій частині вступу коротко викладається зв'язок виконаної розробки з планами і програмами кафедри та університету.

* + 1. Мета і завдання розробки

Формулюють мету і завдання, які необхідно вирішити для досягнення поставленої мети. Метою може бути розробка комп’ютерної системи з певними властивостями, а завданнями – розробка бази даних, інтерфейсу користувача, сайту.

* + 1. Об’єкт і предмет розробки

Об’єкт дослідження– це процес або явище, що породжує проблемну ситуацію й обране для розв’язання цієї проблеми. Визначення об’єкта пов’язано з відповіддю на питання: чого стосується дана робота? Це можуть бути, наприклад, системи штучного інтелекту.

Предмет розробки міститься в межах об’єкта.

Об’єкт і предмет розробки як категорії наукового процесу співвідносяться між собою як загальне і часткове. В об’єкті виділяється та його частина, яка є предметом розробки. Саме на нього спрямована основна увага, оскільки предмет дослідження визначає тему кваліфікаційної роботи, яка визначається на титульному аркуші як її назва.

* + 1. Методи досліджень і розробки

Подають перелік використаних методів дослідження і розробки для досягнення поставленої в роботі мети. Особливу увагу приділити методам розробки. Перераховувати їх треба не відірвано від змісту роботи, а коротко та змістовно визначаючи, що саме досліджувалось чи розроблялось тим чи іншим методом. Це дасть змогу пересвідчитися в логічності та прийнятності вибору саме цих методів.

Про існуючі методи наукових досліджень та розробки програмного забезпечення можна знайти інформацію в мережі інтернет[5]

## Основна частина роботи

Основна частина роботи, відповідно до методичних рекомендацій [2], може складатися з трьох розділів – теоретичний, аналітичний та практичний, але з огляду на технічний характер кваліфікаційної роботи, кафедра пропонує такі складові основної частини роботи:

* аналіз задачі створення системи;
* розробка системи;
* реалізація системи.

## Висновки

Висновки починаються таким чином: «За результатами розробки (зазначається відповідно до мети) сформовано наступні висновки:»

Далі йдуть самі висновки. Висновки нумерують.

У першому пункті висновків коротко викладають формулювання розв’язаної проблеми (задачі), її значення для науки і практики , порівняння з відомими аналогами.

Далі у висновках розкривають методи вирішення поставленої задачі.

У висновках бажано наголосити на якісних і кількісних показниках здобутих результатів.

На завершення формулюють рекомендації щодо практичного використання здобутих результатів.

## Аналіз задачі створення системи

Цей розділ може включати наступні підрозділи:

* аналіз предметної області;
* аналіз вимог до системи;
* постановка задачі на розробку ІКС.

Нижче наведена стисла інформація щодо можливого змісту підрозділів.

### Аналіз предметної області(ПО)

Мета цього етапу роботи – визначити особливості діяльності організації і коло вирішуваних в процесі дипломного проектування задач.

В аналізі ПО виділяють наступні кроки:

* побудова базової моделі предметної області;
* вивчення існуючих в даній ПО систем;
* уточнення загальної моделі.

#### Побудова базової моделі предметної області

У цьому підрозділі може бути наведена наступна інформація:

* архітектурна схема підприємства (кількісні характеристики виробничого процесу, територіальне розташування, схема потоків інформації та інше);
* опис проблеми, вирішення якої має забезпечити розроблювана система;
* концептуальна модель предметної області, для якої створюється система;
* перелік первинних документів та тих, що мають бути сформовані;
* опис задачі та попередній перелік завдань, що мають бути вирішені;
* наявність додаткових обмежень (швидкість обробки інформації, місткість накопичувачів інформації, використання платформ, мов програмування, СУБД та інше), якщо такі мають місце;
* перелік ролей користувачів та розподілення прав доступу).

Для систематизації збору інформації про діяльність підприємства можна використовувати схему Захмана або архітектурну схему підприємства. Вона має представляти в структурованому вигляді цілі організації і базові правила, за якими вона функціонує; організаційну структуру; сутності та дані, з якими має справу організація; виконувані функції; часові та географічні характеристики діяльності організації. Мають бути наведені лише ті аспекти функціонування підприємства, що стосуються виконуваної кваліфікаційної роботи**.**

Для кожної сутності, визначеної на етапі аналізу ПО, необхідно описати властивості, якими вона характеризується.

Результатом цього етапу аналізу ПО може бути концептуальна модель предметної області, що містить сутності та визначає зв’язки між ними.

Концептуальна модель створюється без орієнтації на конкретні програмні та технічні засоби виконання задач ПО, які будуть додаватися в подальшому процесі проектування системи. Приклади зображення концептуальної моделі ПО наведені на рисунках 3.1, 3.2.

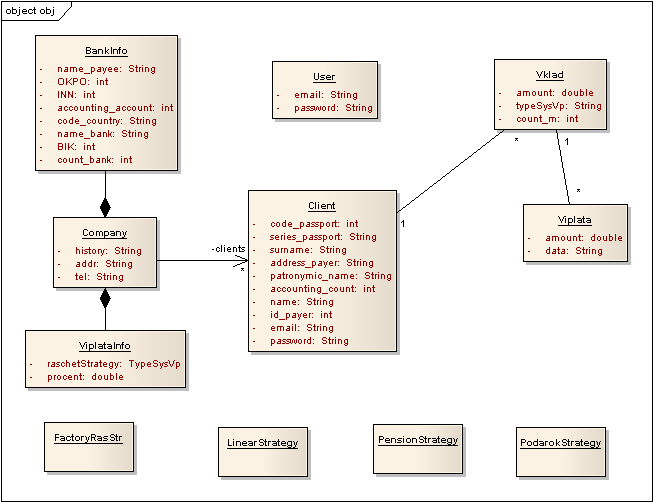


Рисунок 3.1 — Доменна модель об’єктів предметної області

Image2

Рисунок 3.2 – Концептуальна модель ПО

#### Вивчення існуючих систем

Виходячи з результатів аналізу проведеного на першому кроці, формується базовий перелік критеріїв або вимог (як функціональних, так і не функціональних), яким повинна задовольняти система, що створюється.

Наступним кроком є аналіз вже існуючих систем. Мають бути представлені результати збору, обробки, аналізу та узагальнення наукової-технічної інформації, вітчизняного та зарубіжного досвіду в області застосування. Подібні системи розглядаються з метою визначення механізмів реалізації, а також удосконалення та доповнення переліку ключових абстракцій отриманих раніше.

У підрозділі повинні бути наведені результати аналізу технічної реалізації залучених прототипів (їх архітектури і/або структури, способи, методи, алгоритми, застосовані для досягнення поставленої мети; схеми алгоритмів, програм, взаємодії програм, роботи системи, ресурсів системи; принципи дії об'єкта та основні принципові рішення, що дають уявлення про його будову тощо).

Огляд літератури не зводиться просто до її викладу. У ньому повинні:

* наводитися висновки про те, чим те чи інше рішення корисно для вирішення поставленого завдання;
* проведено порівняльний аналіз прототипів;
* виділені ті недоліки, які будуть усунені при вирішенні поставленої задачі.

Також аналогічні системи розглядаються з метою з’ясування відповідності їх до тих критеріїв та вимог, що були сформовані. Результати порівняльного аналізу прототипів представляються в стандартизованому вигляді, приклад наведено в таблиці 3.1.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Таблиця 3.1 – Порівняльні характеристики розглянутих систем | | | | | |
| Критерии оцінки | Система 1 | Система 2 | … |  | Система n |
| Критерій 1 | **+** | **–** |  | **–** | **+** |
| Критерій 2 | **–** | **+** |  | ? | **–** |
| … |  |  |  |  |  |
| Критерій n | **+** | ? |  | **–** | **+** |

#### Уточнення загальної моделі

Метою цього етапу є:

* обґрунтування необхідності створення нової системи;
* конкретизація вимог до створюваної програмної системи з урахуванням проведених досліджень предметної області.

Для представлення отриманої інформації при аналізі предметної області можна використовувати діаграми структурного аналізу – діаграми потоків даних (DFD) (нотації Йордана-ДеМарко, Гейна-Сарсона). Для представлення структури даних можна використовувати діаграми сутностей та зв’язків (entity-relationship diagrams).

Результати проведеного аналізу мають бути враховані в наступному підрозділі при постановці задачі на розробку програмної системи.

### Аналіз вимог до інтерфейсу системи

Слід чітко визначити категорії користувачів системи, їхні особливості та потреби щодо використання застосування, в тому числі фізичні, вікові, психологічні критерії, розумові здібності. Якщо система передбачає наявність декількох груп користувачів, то необхідно навести характеристики та описання кожної з цих груп.

Приклади груп: інваліди, студенти, медичний персонал, люди с поганим зором / слухом, пенсіонери.

Необхідно розглянути і взяти до уваги і такі характеристики проекту, як частота використання системи, мова інтерфейсу, його вигляд (наприклад, наявність потреби у голосовому супроводженні).

### Аналіз вимог до апаратної частини (якщо є)

#### Аналіз вимог до локальної обчислювальної мережі

Інформаційне дослідження функціонування підприємства. в цілому має визначити:

* внутрішні інформаційні потоки;
* зовнішні інформаційні потоки;
* спільні ресурси мережі;
* наявність файлових серверів, серверів баз даних, серверів доступу до Інтернет, серверів голосових комунікацій, тощо.

Для кожного відділу (підрозділу, робочої групи) підприємства повинні бути визначені такі параметри:

* функції відділу;
* кількість робочих місць;
* програмне забезпечення;
* спільні ресурси мережі, які використовуються;
* ресурси, що виділяються відділом для інших користувачів мережі;
* вимоги до рівня безпеки сегменту мережі даного відділу;
* вимоги до системи забезпечення безпеки мережі, тобто сервіси автентифікації, авторизації та обліку, мережні екрани, тощо.

Отримані результати аналізу повинні дозволити планування розміщення та пропускної здатності каналів зв'язку, фізичного та/або логічного розмежування сегментів мережі, попереднє визначення політик безпеки та адміністрування.

#### Аналіз вимог до апаратної підсистеми

У випадку, коли технічне завдання передбачає розробку апаратної підсистеми, що вирішує деякі локальні проблеми, або є центральною у системі слід аналіз має бути спрямований не тільки на формування переліку функціональних вимог до підсистеми. Мають бути визначені:

* місце розташування підсистеми на території підприємства;
* параметри зовнішнього середовища, в яких буде функціонувати підсистема;
* наявність електричних та інформаційних комунікацій;
* вимоги до надійності системи (потреба у дублюванні);
* елементна бази, що використовується в інших апаратних системах, для забезпечення уніфікації;
* кваліфікація обслуговуючого персоналу та користувачів системи.

### Постановка задачі на розробку ІКС (вимоги до системи)

Після аналізу предметної області та огляду вже існуючих рішень зазначеної проблеми, необхідно чітко визначити мету і завдання роботи.

Окремо слід навести вимоги до програмної, та окремо до апаратної частин розробки.

Для програмної частини бажано виділити функціональні вимоги (задачі, що вирішує система; функції, сервіси системи) та не функціональні вимоги (аспекти захищеності та безпеки програмної системи; надійність, час відгуку, швидкість реагування та інші аспекти продуктивності системи, що мають місце).

Вимоги до системи можна представляти у вигляді UML діаграм використання (Use Case). Для кожної ролі користувача розробляється своя діаграма, приклад наведено на рисунку 3.3.

Для апаратної частини бажано сформулювати вимоги до основних параметрів і характеристик передбачуваних результатів (технічні та/або експлуатаційні характеристики апаратури, очікувані результати розробки тощо).

Бажано, щоб уся робота з дипломного проектування була зафіксована на кафедральному Project Manager для курсових (та дипломних) робіт студентів каф. ІКС http://kurs.stu.cn.ua у розділі [Diplom research](http://kurs.stu.cn.ua/projects/diplom-research).



Рисунок 3.3 – Приклад діаграми використання для ролі користувача

## Розробка ІКС

У розділі наводяться запропоновані рішення щодо побудови інформаційно-комп’ютерної системи (ІКС). Має бути запропонована архітектура ІКС, визначені головні компоненти системи та способи їх взаємодії.

Цей розділ може складатися з таких підрозділів:

* вибір технічних засобів побудови системи;
* архітектура ІКС;
* розробка програмної підсистем;
* розробка апаратної підсистеми.

### Вибір технічних засобів побудови системи

У цьому підрозділі необхідно обґрунтувати вибір платформ програмування, вибір операційного середовища розробки та мов та мов програмування, вибір СУБД, вибір елементної бази для апаратної частини. Слід зауважити, що обґрунтування не передбачає копіювання усієї технічної документації про технічний засіб, та перелічення усіх існуючих його версій.

### Архітектура системи

У цьому підрозділі наводиться графічне зображення архітектури системи та детальний опис її елементів та їх взаємодії, а також обґрунтування архітектурних рішень.

Приклади зображення архітектури системи наведено на рисунках 3.4 ‑ 3.5.

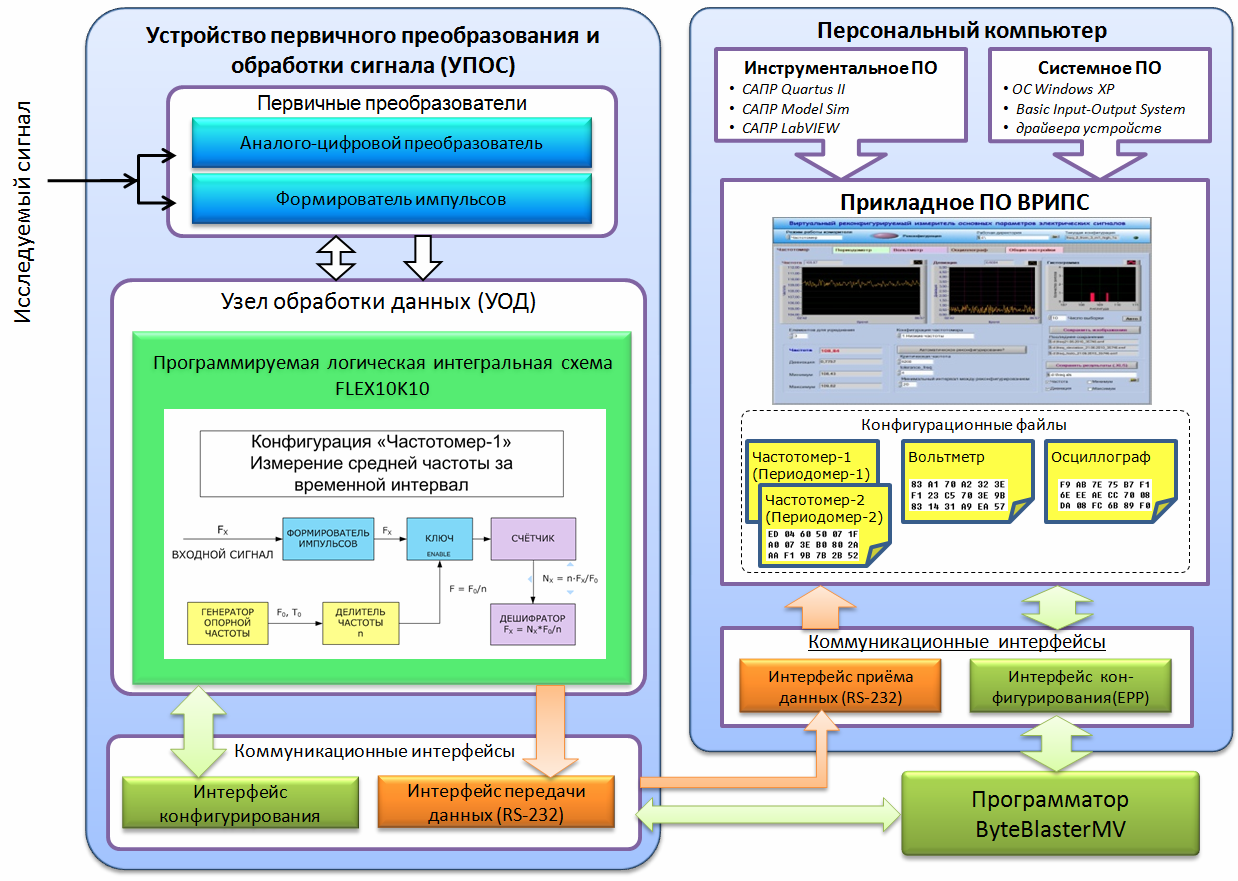


Рисунок 3.4 – Архітектура віртуального вимірювача параметрів електричних сигналів



Рисунок 3.5 – Приклад зображення архітектури системи

### Розробка програмної підсистеми

До цього підрозділу можуть входити такі пункти:

* структура програмної підсистеми;
* розробка класів сутностей системи;
* розробка бази даних;
* розробка форматів вхідних та вихідних даних;
* схема потоків даних;
* схеми алгоритмів обчислень;
* проектування інтерфейсу користувача;
* карта сайту для веб-застосувань;
* діаграми переходів;
* діаграма класів;
* діаграма пакетів;
* діаграми компонентів та інші з поясненням до них;
* UML-діаграми (діяльності, послідовності, бізнес-процесів та інші), що відображають аспекти функціонування системи;
* забезпечення не функціональних вимог до системи (механізми організації безпеки та ін..).

#### Структура програмної підсистеми

У цьому пункті наводимо докладний опис модулів програмної підсистеми, розкриваємо їх структуру та взаємодію. Приклад зображення структури програмної підсистеми наведено на рисунку 3.6.

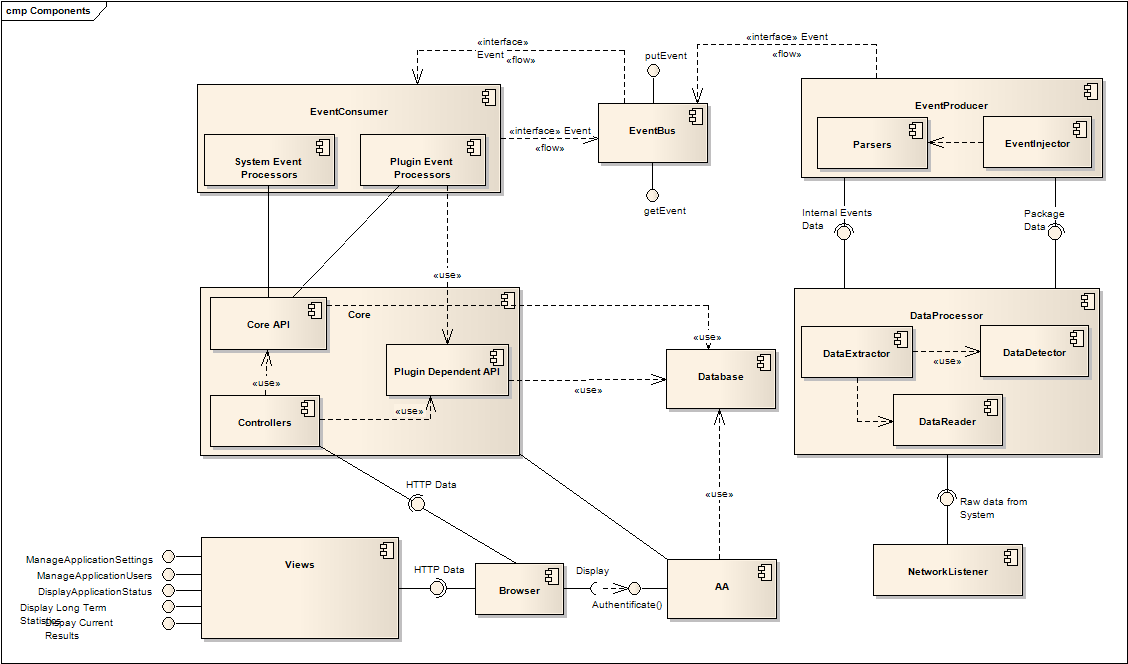


Рисунок 3.6 - Структура програмної підсистеми

#### Розробка класів сутностей системи

Якщо при розробці програмної системи використовується ORM-рішення, то у підрозділі 3.6.1, має бути сформульовано та обґрунтовано вибір ORM-технології. В залежності від принципу використання ORM результати розробки можуть бути наведені одним з наступних варіантів:

* 1. спочатку лістинги класів об’єктів постійності з переліком властивостей класів та метаданих відображення (у вигляді анотацій або файлів конфігурації) з поясненнями до них у цьому підрозділі. У цьому випадку схема БД, яку буде створено за допомогою ORM, наводиться у наступному підрозділі без докладних поянень;
  2. спочатку навести підрозділ розробки БД з повними поясненнями до схеми БД, після чого лістинги генерованих класів.

У лістингу 3.1 наведено приклад класу об’єкту постійності.

Обов’язково слід навести пояснення до механізмів відображення зв’язків між об’єктами, організації наслідування тощо.

Лістинг 3.1– Приклад класу об’єкта постійності (persistent object)

@@Entity

@@Table(name = "GIFT")

@@NamedQueries( {

@@NamedQuery(name = QueryNames.GIFT\_GET\_BY\_OWNER,

query = "SELECT x FROM Gift x WHERE x.owner.id = ?1"),

@@NamedQuery(name = QueryNames.GIFT\_GET\_BY\_PRESENTER,

query = "SELECT x FROM Gift x WHERE x.presenter.id = ?1"),

@@NamedQuery(name = QueryNames.GIFT\_GET\_AVAIL\_BY\_OWNER,

query = "SELECT x FROM Gift x WHERE x.owner.id = ?1

AND x.presenter = NULL") })

public class Gift extends DomainObject {

/\*\* \*/

private static final long serialVersionUID = 1L;

/\*\* Name of the gift \*/

@@NotNull(message = "Name of the gift cannot be null")

@@NotEmpty(message = "Name of the gift cannot be empty")

private String name;

/\*\* Url of the gift image \*/

private String imgUrl;

/\*\* Url of the gift page \*/

private String url;

/\*\* Description of the gift \*/

@@Lob

private String description;

/\*\* Date the gift was added \*/

@@Temporal(TemporalType.DATE)

@@NotNull(message = "Date the gift was added cannot be null")

private Calendar dateAdd = Calendar.getInstance(getDefLocale());

/\*\* Date and time the gift was selected by presenter \*/

@@Temporal(TemporalType.TIMESTAMP)

private Calendar dateTimeSelected;

/\*\*

\* User that requests a gift

\*/

@@ManyToOne(fetch = FetchType.EAGER)

@@NotNull

private User owner;

/\*\*

\* User that will present a gift

\*/

@@ManyToOne(fetch = FetchType.EAGER)

private User presenter;

…

}

#### Розробка бази даних

У цьому пункті описуються основні етапи розробки БД. Він може складатися з таких підпунктів:

* розробка схеми БД;
* розробка збережених процедур та тригерів.

**Розробка схеми БД**

Розробка схеми БДскладається з таких етапів:

1. на основі концептуальної моделі ПО створюються базові таблиці БД. Таблиця в БД відповідає конкретній сутності в концептуальній моделі ПО. При створенні таблиці необхідно:
2. визначити список колонок (колонки виводяться із властивостей (атрибутів) сутності концептуальної моделі ПО);
3. визначити тип даних для кожної колонки;
4. визначити ім’я таблиці, яке може співпадати з ім’ям сутності концептуальної моделі ПО;
5. на основі бізнес-правил визначити обмеження на значення даних в колонках.
6. створити додаткові таблиці для реалізації зв’язку типу «багато-до-багатьох», якщо він є в концептуальній моделі ПО.
7. виконати підтримку цілісності за посиланнями:
   1. визначити первинні ключі для кожної таблиці;
   2. визначити зовнішні ключі в дочірніх таблицях;
   3. визначити правила підтримки цілісності за посиланнями (каскадне оновлення і видалення).
8. створити індекси для базових таблиць БД та обґрунтувати вибір колонок, з яких формуються ці індекси.

Всі етапи розробки схеми БД повинні бути докладно описані в пояснювальній записці.

Результатом даного етапу має бути схема БД. Приклад схеми БД наведено на рисунку 3.7.

**Розробка тригерів та збережених процедур**

В цьому підпункті описуються збережені процедури та тригери, які визначаються бізнес-логікою ПО.

Для кожного тригеру необхідно описати для чого він призначений, які дії виконує з БД. Крім того, необхідно вказати таблицю, з якою зв’язаний тригер, подію, на яку він реагує і час спрацьовування.

Для кожної збереженої процедури необхідно описати для чого вона призначена і які дії виконує з БД, розрахункові формули та, можливо, схеми алгоритмів. Слід визначити ім’я, вхідні та вихідні параметри, їх типи.

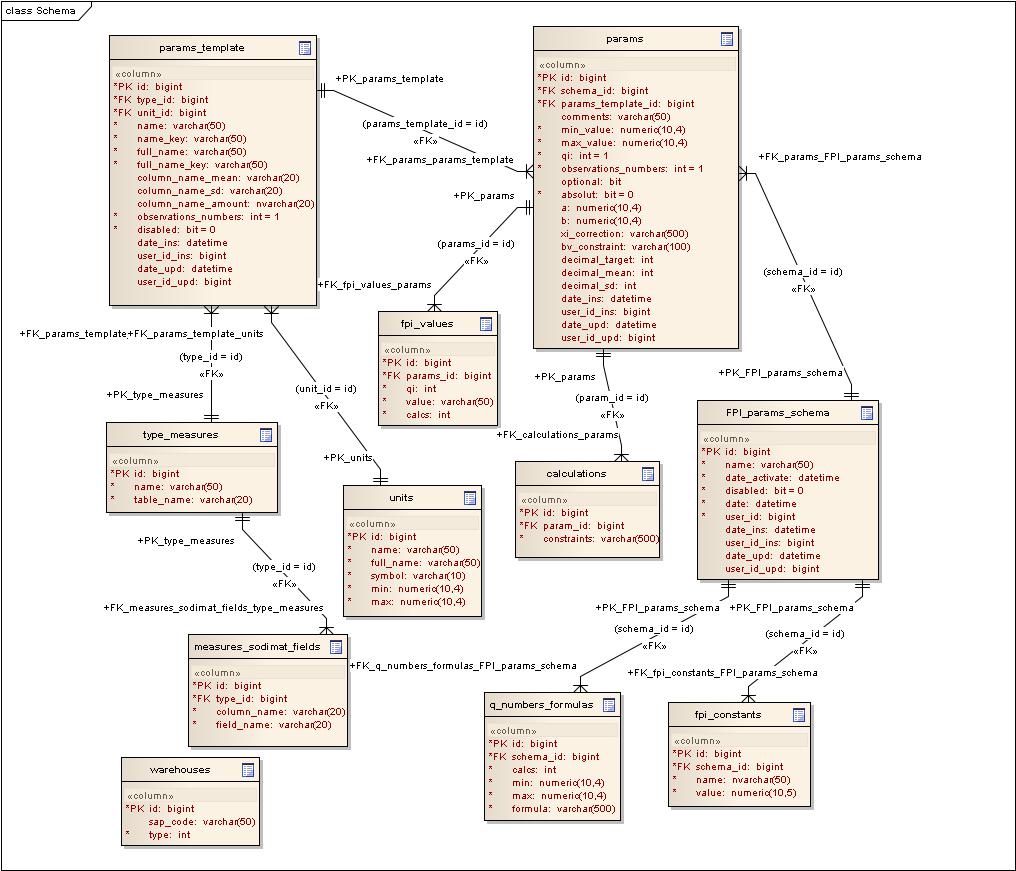


Рисунок 3.7 – Схема бази даних

#### Розробка технології роботи з даними

Цей пункт потрібен у тих випадках коли відбувається обмін даними між віддаленими компонентами системи. Тут можна навести:

* запропоновану схему потоків даних (приклад на рисунку 3.8) та її опис;
* навести запропонований формат передачі даних (приклад у лістингу 3.2);



Рисунок 3.8 – Приклад зображення схеми потоків даних

Лістинг 3.2– Формат даних



#### Розробка алгоритмів взаємодії між компонентами системи

У цьому пункті наводиться опис взаємодії між компонентами системи під час її функціонування. У якості ілюстрації до цього пункту можна навести UML-діаграми послідовностей, рисунки 3.9 та 3.10.

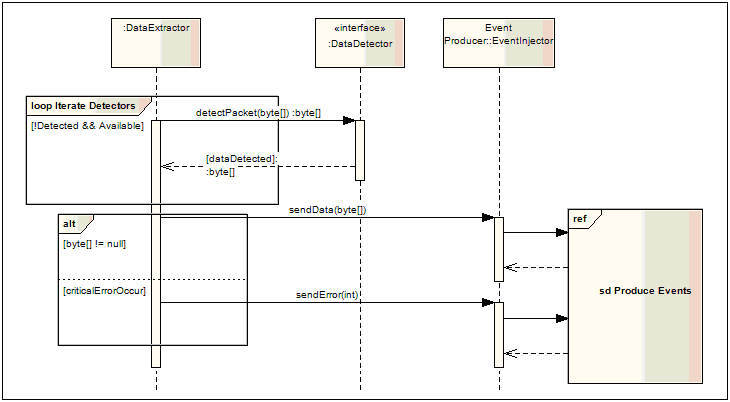


Рисунок 3.9 – Діаграма послідовності обробки даних



Рисунок 3.10 – Діаграма послідовності OpenID-аутентифікації

#### Моделювання виконання операцій

У цьому пункті слід надати інформацію про особливості алгоритмічної та логічної реалізації виконання операцій в системі.

Для основних операцій (або обчислень) має бути наведена інформація про використані алгоритми, схеми алгоритмів та пояснення до цих схем. Таких схем має бути, як мінімум дві. Для зображення схем алгоритмів слід застосовувати рекомендації ЕСПД (рисунок 3.11).

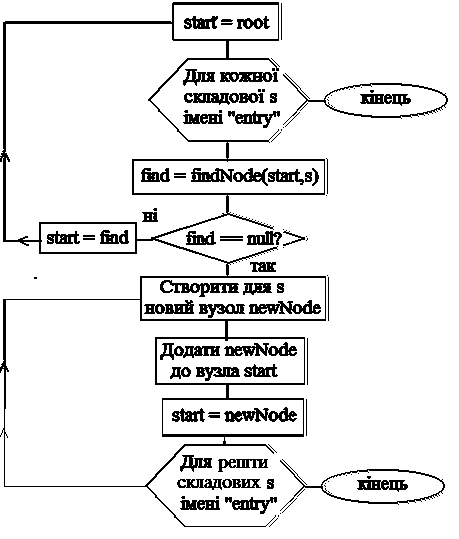


Рисунок 3.11 – Схема алгоритму обчислення у нотації ЕСПД

Для надання інформації про поведінку системи (або її окремих об’єктів) при виконанні операцій можна використовувати UML-нотацію (діаграми діяльності, станів та ін..). Приклад наведено на рисунку 3.12.

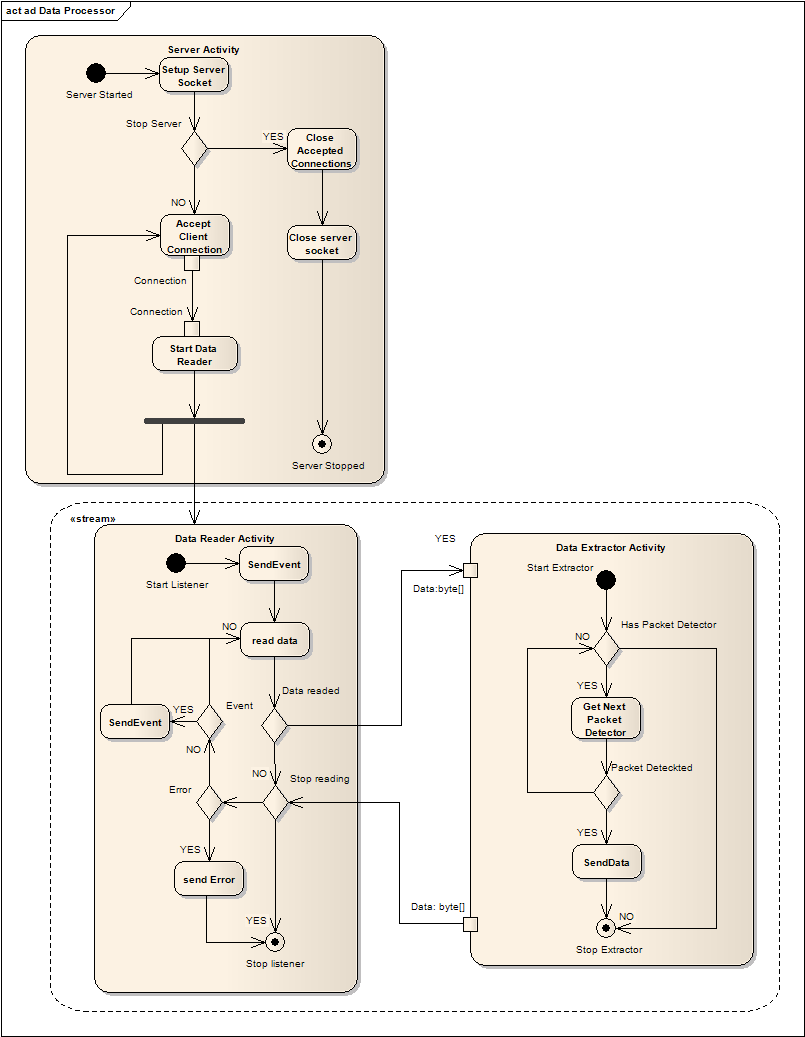


Рисунок 3.12 – Особливості реалізації операцій системи у нотації UML

#### Діаграми класів сутностей, класів бізнес-логіки та допоміжних класів

У розділі мають бути наведені діаграми класів системи з поясненнями до них. Якщо в архітектурі системи має місце розподілення на шари, то розділ слід ділити на відповідні підрозділи (окремо навести діаграму класів сутностей, бізнес-логіки, допоміжних класів). Якщо властивості класів об’єктів постійності та метадані об’єктно-реляційного відображення були наведені в розділі 3.2.2.2, то в цьому розділі навести пояснення лише до методів (наприклад, що задають бізнес-логіку системи). На рисунку 3.13 наведено приклад діаграми класів.

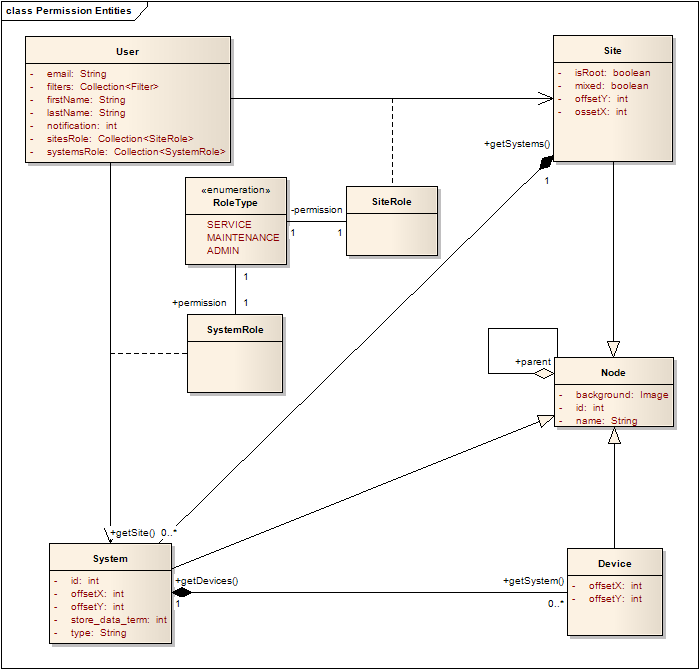


Рисунок 3.13 – Діаграма класів

#### Розробка WEB-компонент системи

Розробка WEB–інтерфейсу користувача може складатися з таких етапів:

* розробка ескізу інтерфейсу користувача;
* розробка сценарію використання інтерфейсу;
* розробка карти сайту;
* діаграма переходів між сторінками.

**Розробка ескізу інтерфейсу користувача** має базуватися на переліку задач, що вирішує система та проведеному аналізі користувачів. Слід проаналізувати особливості кожної із задач та знайти відповіді на питання: Навіщо? Яким чином? За допомогою чого? Що має бути результатом?

Слід враховувати наявність передумов для задач, виключні ситуації. Описання кожної задачі має включати:

* мету;
* передумови;
* частоту використання задачі;
* мову інтерфейсу;
* час, відведений на виконання задачі;
* з ким треба контактувати, під час виконання;
* що може піти не так;
* необхідність голосового супроводу.

В аналізі користувачів слід виділити групу людей, які будуть використовувати вашу систему, розділити їх на класи та описати кожен клас:

* вік, стать, етнічна приналежність;
* освіта;
* фізіологічні особливості;
* базовий комп’ютерний досвід;
* навички;
* досвід предметної області;
* робоче середовище користувача.

Використовуючи аналіз завдання, перетворіть Ваші завдання в попередній графічний дизайн. Він складається з декількох замальовок вікон застосування або діалогових вікон, що включають меню управління і контролю, які може використовувати користувач і коротко опишіть призначення кожного з них.

**Сценарії використання інтерфейсу**. Для найбільш найважливіших завдань проекту, розробіть "сценарій" – конкретний, реалістичний приклад завдання. Там, де завдання поставлене абстрактно, наприклад "Відправка електронного звіту", сценарій має бути повним, завершеним з вказівкою деталей. Наприклад: "Вибираємо файл електронного звіту, переміщаємо його в список для відправлення, вибираємо зі списку адресатів потрібної нам людини, вказуємо час відправки....і т. д". Сценарій можна представити у вигляді схеми алгоритму або UML - діаграм діяльності.

**Розробка карти сайту**. Карта сайту відображає усі розділи, підрозділи і сторінки системи. Карти сайту часто представляють у вигляді дерева. У загальному випадку карти сайту дозволяють: побудувати інформаційну архітектуру і продумати шляхи навігації по сайту, згрупувати інформацію в найбільш зрозумілому і зручному для користувача виді.   
На проектованій карті показують усі елементи кожного рівня, кожен з яких виділяється і може бути позначений різним кольором для чіткішого візуального розмежування рівнів. Кольорові виділення також можуть використовуватися для позначення сторінок, що входять в зону обмеженого доступу. Приклади зображення карти сайту наведені на рисунках 3.14-3.15.

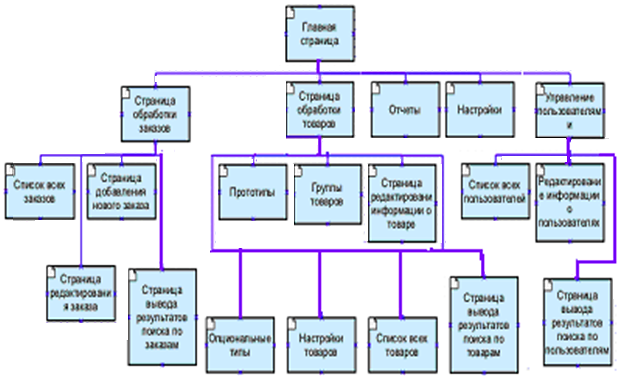


Рисунок 3.14 — Карта адміністративної частини сайту



Рисунок 3.15 – Карта сайту клієнтської частини системи

**Діаграми переходів між сторінками** – це схеми роботи користувача з функціональністю системи. Ця діаграма показує, як користувач виконує одне зі своїх завдань, переходячи від сторінки до сторінки.   
На відміну від карти сайту і схеми навігації, діаграми переходів відображають не усю систему, а її конкретний модуль і детальний процес роботи користувача з ним. Приклад діаграми переходів наведений на рисунку 3.16.



Рисунок 3.16 – Діаграма переходів між сторінками для користувача системи

### Розробка апаратної підсистеми

На кафедрі за звичай використовуються два варіанти виконання цього розділу роботи.

Перший варіант – це проектування локальної обчислювальної мережі як складової частини системи. Такий варіант буде доцільним у тому випадку, коли у системі багато пов’язаних між собою комп’ютерів і може бути доречним тільки для спеціальностей КС та СП.

Другий варіант – це проектування апаратної підсистеми, що вирішує деякі локальні проблеми (для спеціальностей КС та СП), або є центральною у системі (для спеціальності СК).

#### Проектування локальної обчислювальної мережі

У цьому розділі повинен бути описаний творчий процес проектування локальної мережі підприємства з виходом в мережу Інтернет. Під час розробки повинні бути передбачені системні сервіси для підтримки функціонування мережі: DNS, DHCP, файлові сервіси, сервіси конфігурацій, тощо. Вихідними даними для розробки є поповерховий план будівлі з позначеними на ньому місцями розташування обчислювальної техніки та мереженого обладнання, розміщенням відділів підприємства та виділеними кімнатами для розміщення обладнання. Проект повинен бути спрямований на перспективний розвиток мережі, а не описувати поточний стан. Все обладнання повинно обиратися з обґрунтованих міркувань після проведення аналізу потреб та вартості, з посиланням на використані джерела. Повинні бути наведені технічні характеристики використаного обладнання так само з посиланням на джерела Процес розробки локальної обчислювальної мережі повинен спиратися на проведене інформаційне дослідження підприємства.

Проектування повинно спиратися на стандарти структурованих кабельних систем, але не повинно забезпечувати весь необхідний для побудови СКС пакет документів. На етапі проектування повинен бути виконаний ескіз мережі та її моделювання. Приклади міжнародних стандартів, які можуть бути використані під час розробки:

* EN 50173:2007 Information Technology - Generic cabling systems;
* TIA/EIA-568-B.1 ﻿Commercial Building Telecommunications Cabling Standard. Part 1. General requirements (May 2001);
* ISO/IEC 15018 (2004) Integrated cabling for all services other than mains power in homes, SOHO (Small Office, Home Office), and buildings.

Оформлення технічного проекту повинно здійснюватися у відповідності з ДСТУ Б А.2.4-42:2009. В проекті повинні бути присутні принаймні такі документи:

* структурна (функціональна) схема вузлів комутації;
* структурна схема СКС;
* журнали кабельних з’єднань.

Умовні графічні зображення елементів виконуються згідно з ДСТУ Б А.2.4-40:2009 «ТЕЛЕКОМУНІКАЦІЇ. ПРОВОДОВІ ЗАСОБИ ЗВ'ЯЗКУ. УМОВНІГРАФІЧНІ ЗОБРАЖЕННЯ НА СХЕМАХ ТА ПЛАНАХ».

#### Проектування апаратної підсистеми

У розділі описується творчий процес проектування цільової системи та виділення її основних елементів. У цьому ж розділі проводиться розробка і виконання схеми структурної апаратної підсистеми згідно з ГОСТ 2.702-75. Схема наводиться в додатку. У цьому ж розділі, окремим підрозділом наводиться опис процесу побудови схеми електричної принципової. В якості ілюстрацій наводяться фрагменти схеми електричної принципової, які демонструють електричні з'єднання окремих функціональних частин з поясненнями принципів з'єднання та вибору номіналів елементів. Обов'язкове посилання на довідкову літературу. Наводяться зрозумілі міркування з посиланням на літературні джерела щодо причини застосування того чи іншого елемента. Наводяться технічні характеристики цих елементів так само з посиланням на джерела. Обов'язково наводяться матеріали, які доводять, що характеристики та параметри зазначені в ТЗ будуть забезпечені. Наводяться відомості про використані інтерфейси з ілюстрацією часових діаграм, заголовків протоколів, призначень ліній зв’язку тощо. Підсумком роботи над розділом є технічний проект котрий розміщується в додатку.

Оформлення технічного проекту є однією з вимог для успішного захисту кваліфікаційної роботи фахівця. Рекомендації щодо оформлення та приклади можна знайти за посиланнями:

* <https://sites.google.com/site/diplomics2/home>
* <https://sites.google.com/site/diplomics2/courses>

Хоча технічний проект в чистому вигляді є етапом повномасштабної розробки зразка, і має бути виконаний в обсязі, що дозволяє почати детальну конструкторську розробку всіх його елементів і зразка в цілому, але в контексті виконуваної студентами кваліфікаційної роботи, він носить спрощений характер. Зокрема він зазвичай складається з декількох аркушів А4 в додатку і 1-2 листи А1 у вигляді плакатів використовуваних під час доповіді. У складі додатку кваліфікаційної роботи бакалавра, після титульної сторінки додатка, першою сторінкою технічного проекту йде відомість технічного проекту, яка оформляється згідно з ГОСТ 2.106. Якщо коротко, то в ній перераховуються всі документи які входять до складу ТП. Зазвичай це:

* безпосередньо сам аркуш ТП;
* схема електрична структурна (Е2);
* схема електрична принципова (Е3);
* перелік елементів (ПЕ);
* текст програми до програмованих елементів апаратної підсистеми (12).

Шаблон даної сторінки доступний у додатку методичних вказівок. Другою (третьою, а іноді четвертою) сторінкою йде перелік елементів до схеми електричної принципової (зазвичай група СК розробляє якийсь електронний пристрій) або до схеми електричних з'єднань (КС і СП можуть розробляти ЛОМ). Перелік елементів оформляється згідно з ГОСТ 2.104-68 (форма 2 та 2а) та ГОСТ 2.702-84. Шаблон даної сторінки для MS WORD доступний у додатку методичних вказівок. Схеми на форматі А1 виносяться у вигляді ілюстративного матеріалу до доповіді і безпосередньо перед виступом поряд з іншими плакатами вивішуються на огляд комісії. Схема електрична принципова, електрична структурна, електричних з'єднань оформляється згідно з ГОСТ 2.701-84 та ГОСТ 2.702–75

Номери діючих ГОСТ, які обумовлюють правила виконання схем наведені у таблиці 3.2.

|  |  |
| --- | --- |
| Таблиця 3. 2 – Стандарти, що обумовлюють правила виконання схем | |
| Номер ГОСТ | Назва |
| ГОСТ 2.701 | Схемы. Виды и типы. Общие требования к выполнению. |
| ГОСТ 2.702 | Правила выполнения электрических схем. |
| ГОСТ 2.705 | Правила выполнения электрических схем обмоток и изделий с обмотками. |
| ГОСТ 2.708 | Правила выполнения электрических схем цифровой вычислительной техники. |
| ГОСТ 2.709 | Система обозначения цепей в электрических схемах. |
| ГОСТ 2.710 | Обозначения буквенно-цифровые в электросхемах. |

Умовні графічні позначення (УГП) елементів виконуються згідно з с ЕСКД. Номера відповідних ГОСТ наведені у таблиці 3. 2

|  |  |
| --- | --- |
| Таблиця 3.3 – Стандарти для умовних графічних позначень елементів | |
| Номер ГОСТ | Назва |
| ГОСТ 2.711 | Схема деления изделия на составные части. |
| ГОСТ 2.721 | Обозначения общего применения. |
| ГОСТ 2.722 | Машины электрические. |
| ГОСТ 2.723 | Катушки индуктивности, дроссели, трансформаторы, автотрансформаторы и магнитные усилители. |
| ГОСТ 2.725 | Устройства коммутирующие. |
| ГОСТ 2.726 | Токосъемники. |
| ГОСТ 2.727 | Разрядники, предохранители. |
| ГОСТ 2.728 | Резисторы, конденсаторы. |
| ГОСТ 2.729 | Приборы измерительные. |
| ГОСТ 2.730 | Приборы полупроводниковые. |
| ГОСТ 2.731 | Прибору электровакуумные. |
| ГОСТ 2.732 | Источники света. |
| ГОСТ 2.733 | Обозначения условные детекторов ионизирующих излучений в схемах. |
| ГОСТ 2.734 | Линии сверхвысокой частоты и их элементы. |
| ГОСТ 2.735 | Антенны. |
| ГОСТ 2.736 | Элементы пьезоэлектрические и магнитострикционные. Линии задержки. |

## Реалізація системи

У цьому розділі наводиться інформація про результати реалізації системи. Розділ може складатися з двох підрозділів:

* реалізація програмної підсистеми;
* реалізація апаратної підсистеми.

У цей розділ можна включити довідку про впровадження системи.

### Реалізація програмної підсистеми

Розділ має включати основні результати етапу реалізації системи. Може включати наступну інформацію:

* результати реалізації бази даних;
* діаграми пакетів;
* протоколи класів;
* опис механізмів автоматизації зборки та установки застосування;
* діаграми розгортання;
* лістинги файлів конфігурації (наприклад, для Spring, Hibernate, JSF, дескриптор розгортання web.xml та ін..);
* тестування програмного забезпечення, залучені види тестування та їх результати;
* результати розробки у вигляді декількох основних сценаріїв використання системи з наведенням відповідних екранів інтерфейсу користувача.

#### Результати реалізації бази даних

Підрозділ має містити повний опис всіх таблиць БД з урахуванням особливостей конкретної СКБД (типів даних, синтаксису введення обмежень та ін..) та тексти вбудованих процедур

Виконуючи повний опис таблиць БД необхідно:

* вказати назви колонок;
* для кожної колонки визначити тип даних;
* якщо необхідно, то вказати значення атрибутів за умовчанням а також обов’язковість вводу значень атрибутів;
* описати обмеження, які накладаються на значення даних атрибутів.

Опис таблиць СКБД PostgreSQL може виглядати так, як показано у таблицях 3.4, 3.5.

В цьому ж підрозділі можна навести листинги збережених процедур та тригерів.

|  |  |  |  |
| --- | --- | --- | --- |
| Таблиця 3.4 – Опис таблиці «Supermarket» | | | |
| Назва колонки | Тип даних | Обмеження | Опис |
| id | serial | not null | первинний ключ, унікальний ідентифікатор супермаркету |
| name | character varying (50) | not null | назва супермаркету |
| phone | bigint | not null | телефон супермаркету |
| boss | character varying (50) | not null | ПІП директора |
| id\_city | integer | not null | зовнішній ключ для зв’язку з таблицею «City» (ідентифікатор міста, в якому розташований супермаркет) |

|  |  |  |  |
| --- | --- | --- | --- |
| Таблиця 3.5 – Опис таблиці «Seller» | | | |
| Назва колонки | Тип даних | Обмеження | Опис |
| id | serial | not null | первинний ключ, унікальний ідентифікатор продавця |
| name | character varying (50) | not null | ПІП продавця |
| age | integer | * not null * default 18 * check (age >=18) | вік продавця |
| post | character varying (50) | not null | посада |
| rating | real | * not null * default 0 * check (rating>=0 and rating <=100) | рейтинг продавця |
| id\_supermarket | integer | not null | зовнішній ключ для зв’язку з таблицею «supermarket» (ідентифікатор супермаркету, в якому працює продавець) |

#### Діаграма пакетів

У цьому пункті наводяться результати об’єднання класів у пакети, відповідно до їх функціонального спрямування. Тут слід навести діаграму пакетів, де відображені зв’язки між пакетами, рисунок 3.17.

#### Протоколи класів

У цьому пункті наводяться протоколи класів, згруповані за пакетами. Під протоколом класу тут мається на увазі прокоментований перелік полів та заголовків методів, щось на зразок Javadoc.

Повний текст класів наводиться в додатках;

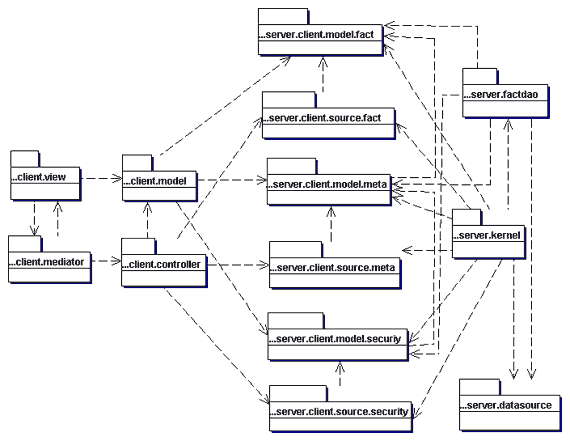


Рисунок 3.17 – Діаграма пакетів

#### Опис процесу автоматизації збірки та установки застосування

Цей пункт доречний у тому випадку, коли складання програмного проекту відбувалося за допомогою засобів автоматизації, наприклад, Ant.

#### Тестування програмного забезпечення

У цьому пункті слід навести залучені види тестування програмної підсистеми, якщо такі засоби використовувалися та їхні результати.

#### Діаграма розгортання ІКС

Діаграма розгортання системи має супроводжуватися відповідним описанням. Приклад діаграми наведено на рисунку 3.18.

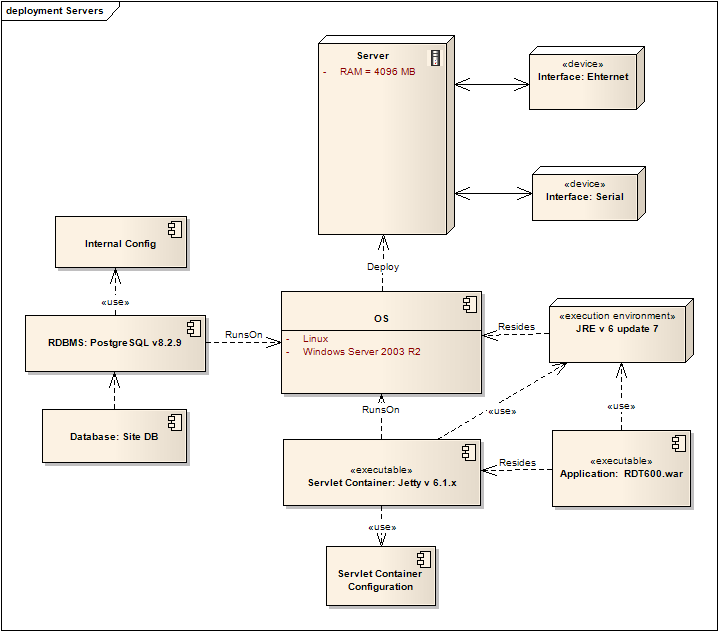


Рисунок 3.18 – Діаграма розгортання ІКС

#### Реалізація інтерфейсу користувача

Опишіть внутрішні особливості вашої реалізації, не заглиблюючись в надра проекту. Опишіть основні дизайнерські рішення, які ви застосували при реалізації проекту. Також опишіть як проблеми реалізації можуть вплинути на використання вашого інтерфейсу. Прослухати

Словник - [Переглянути докладний словник](http://www.google.com.ua/dictionary?source=translation&hl=uk&q=&langpair=)

Цей пункт може включати такі складові:

* основні форми графічного інтерфейсу користувача;
* результати тестування прототипу.

Представляючи основні форми графічного інтерфейсу користувача,  
докладаємо скріншоти отриманих форм інтерфейсу користувача, з описом основних полів, заголовків, форм, кнопок. Рекомендується використовувати виноски.

Наводячи результати тестування комп'ютерного прототипу   
опишіть, як Ви проводили тестування. Як ви знайшли користувачів для тестування, і як вони представляють користувачів системи. Який брифінг ви з ними проводили і які завдання видавали. Які критичні ситуації виникали в процесі тестування. Які проблеми використання залишилися в проекті і які Ви бачите варіанти їх вирішення.

### Реалізація апаратної частини

Цей підрозділ може бути включений до роботи, якщо спроектований підсистема, або пристрій були дійсно реалізовані. У цьому випадку можна навести фотографії зовнішнього вигляду пристрою та його внутрішньої компоновки. Навести часові діаграми або осцилограми його функціонування. Навести метрологічні оцінки вихідних параметрів тощо.

**Перелік посилань**

а) посилання на книгу : 1. Кузнецова С.А. Фінансовий менеджмент: у схемах і таблицях: навчальний посібник / С.А. Кузнецова. – Дніпропетровськ: Дніпропетровський університет імені Альфреда Нобеля, 2011. – 180 с.

б) посилання на статтю: 1. Кузнецова С.А. Парадигма управління грошовими потоками в хаотично структурованій економіці / С.А. Кузнецова, В.М. Вареник // Академічний огляд. – 2012. – №1. – С. 63-68.

в) посилання на електронні джерела інформації : 1. Монетарний огляд за 2011 рік // Офіційний сайт Національного банку України / [Електронний ресурс]. – Режим доступу: http:// [www.bank.gov.ua](http://www.bank.gov.ua)

г) посилання на нормативні документи : 1. Закон України «Про Антимонопольний комітет України»: за станом на 26 листопада 1993р. / Верховна Рада України. / [Електронний ресурс]. – Режим доступу: http://zakon.rada.gov.ua/cgi-bin/laws/main.cgi?nreg=365912&p=1311770615678004.

1 .Наказ Міністерства освіти і науки України від 19.11.2018 № 1262. Про затвердження стандарту вищої освіти за спеціальністю 123 «Комп’ютерна інженерія» для першого (бакалаврського) рівня вищої освіти. / [Електронний ресурс]. – Режим доступу: <https://mon.gov.ua/storage/app/media/vishcha-osvita/zatverdzeni%20standarty/123-kompyuterna-inzheneriya.pdf>

2. Методичні рекомендації щодо виконання та оформлення випускних кваліфікаційних робіт (проектів) здобувачів вищої освіти освітніх ступенів «бакалавр» і «магістр» Чернігівського національного технологічного університету Чернігів 2016 15с. / [Електронний ресурс]. – Режим доступу:<https://www.stu.cn.ua/media/files/pdf/metodi.pdf>

3. Наказ ректора № \_09 від 21.01.2019, Про внесення змін до «Методичних рекомендацій щодо виконання та оформлення випускних кваліфікаційних робіт (проектів) ЗВО освітніх ступенів «бакалавр» і «магістр» ЧНТУ»

4. Комп’ютерні системи, мережі та системне програмування. Методичні вказівки з виконання кваліфікаційної роботи магістра та самостійної роботи для студентів спеціальностей: 8.05010201 – “Комп’ютерні системи та мережі”, 8.05010202 – “Системне програмування”, 8.05010203 – “Спеціалізовані комп’ютерні системи” / Укл.: Казимир В.В., Вервейко О.І. – Чернігів: ЧДТУ, 2012. – 34 с.

5.Методологія розробки програмного забезпечення. / [Електронний ресурс].

– Режим доступу:  
 <https://uk.wikipedia.org/wiki/>Методологія\_розробки\_програмного\_забезпечення

Додаток А

**Приклад оформлення титульного аркуша**

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ЧЕРНІГІВСЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНОЛОГІЧНИЙ УНІВЕРСИТЕТ

Навчально-науковий інститут електронних та інформаційних технологій

Кафедра інформаційних та комп’ютерних систем

**Допущено до захисту**

Завідувач кафедри

к.е.н., доцент Базилевич В.М.

«\_\_\_» \_\_\_\_\_\_\_\_\_ 2020 р.

ВИПУСКНА КВАЛІФІКЦІЙНА РОБОТА

за освітньо-професійною програмою бакалавра

ІНФОРМАЦІЙНА КОМП’ЮТЕРНА СИСТЕМА   
ДЛЯ ПОСТАЧАЛЬНОЇ ОРГАНІЗАЦІЇ

Спеціальність 123 – Комп’ютерна інженерія

Галузь знань 12 – Інформаційні технології

Виконавець:

студент гр. КІ–162

Ступак Сергій Олександрович

*(підпис)*

Керівник:

|  |  |  |
| --- | --- | --- |
| *доцент* |  | *к.т.н., доцент* |
| **–––––––––––––––––––––––––––––** |  | **–––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––** |
| *(посада)* |  | *(науковий ступінь, вчене звання)* |
|  |  |  |

Бивойно Павло Георгійович

*(підпис)*

Чернігів 2020

Додаток Б

**Індивідуальне завдання на випускну кваліфікаційну роботу (проекту)**

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ЧЕРНІГІВСЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНОЛОГІЧНИЙ УНІВЕРСИТЕТ

Навчально-науковий інститут електронних та інформаційних технологій

Кафедра інформаційних та комп’ютерних систем

ЗАТВЕРДЖУЮ:

Зав. кафедри

к.е.н., доцент Базилевич В.М.

“\_\_\_\_” \_\_\_\_\_\_\_\_\_\_\_2020г.

**ІНДИВІДУАЛЬНЕ ЗАВДАННЯ**

НА ВИПУСКНУ КВАЛІФІКАЦІЙНУ РОБОТУ  
 ЗДОБУВАЧА ВИЩОЇ ОСВІТИ

ЗА ОСВІТНЬО-ПРОФЕСІЙНОЮ ПРОГРАМОЮ БАКАЛАВРА

Ступака Сергія Олександровича

Тема роботи: ІНФОРМАЦІЙНА КОМП’ЮТЕРНА СИСТЕМА ДЛЯ ПОСТАЧАЛЬНОЇ ОРГАНІЗАЦІЇ

Тему затверджено наказом ректора

від "\_ \_"\_\_ \_\_\_ 2020р. №\_\_\_\_с\_

1. **Вхідні дані до роботи:**

Кількість найменувань товарів до 1000. Кількість постачальників до 100. Кількість клієнтів до 100. Операційна система Linux. Є робітники із вадами зору (дальтонізм).

1. **Зміст розрахунково-пояснювальної записки:**

Кваліфікаційна робота складається з вступу, основної частини та висновків. Основна частина складається з трьох розділів: «Аналіз задачі створення системи», «Розробка системи» та «Реалізація системи».

1. **Демонстраційні матеріали:**

7 слайдів для презентації роботи

**4.Календарний план**

|  |  |  |  |
| --- | --- | --- | --- |
| № | Назва етапів роботи | Термін виконання | Примітки |
|  | Визначитися з тематикою роботи | 10.02.2020 |  |
|  | Знайомство з проблемою, інформаційними джерелами, аналіз існуючих рішень | 06.03.2020 |  |
|  | Аналіз задачі створення системи | 01.04.2020 | Звіт |
|  | Розробка бази даних та задач обробки даних | 16.04.2020 |  |
|  | Розробка сайту та АРМ робітника складу | 18.05.2020 |  |
|  | Залік з переддипломної практики | 18.05.2020 | Звіт |
|  | Реалізація системи | 01.06.2019 |  |
|  | Підготовка текстової частини і слайдів | 01.06.2020 | Звіт |
|  | Попередній захист дипломної роботи | 01.06.2020 |  |
|  | Завершення оформлення, рецензування, перевірка на плагіат | 15.06.20 |  |
|  | Захист дипломної роботи | з 15.06.20 |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

|  |  |  |
| --- | --- | --- |
| Завдання підготував: |  |  |
| керівник |  | Бивойно Павло Георгійович |
| *(підпис)* |  |  |
|  |  | «\_\_\_» \_\_\_\_\_\_\_\_\_ 2020 р. |
| Завдання одержав: |  |  |
| студент |  | Ступак |
| *(підпис)* |  |  |
|  |  | «\_\_\_» \_\_\_\_\_\_\_\_\_ 2020 р. |