

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ВСП «ОДЕСЬКИЙ ТЕХНІЧНИЙ ФАХОВИЙ КОЛЕДЖ ОНТУ»**

**Спеціальність: 121 «Інженерія програмного забезпечення»**

**Освітня програма: «Розробка програмного забезпечення»**

**Група: 4РП-06**

# **Дипломний проект**

**здобувача освіти денної форми навчання**

**РП.06.13.000.ДП**

**КУЗНЄЦОВ  
ОЛЕКСАНДР ОЛЕГОВИЧ**

**м. Одеса  
2023 р.**

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ВСП «ОДЕСЬКИЙ ТЕХНІЧНИЙ ФАХОВИЙ КОЛЕДЖ ОНТУ»

Спеціальність: 121 «Інженерія програмного забезпечення»

Освітня програма: «Розробка програмного забезпечення»

Група: 4РП-06

## ПОЯСНЮВАЛЬНА ЗАПИСКА

до дипломного проекту (роботи) на тему:

### Розробка програмного модуля "Електронний гаманець співробітника" для ERP-системи документообігу складу

Проектний матеріал складається з пояснювальної записки на 60 сторінках та графічного (презентаційного) матеріалу на 10 аркушах (слайдах).

Дипломник \_\_\_\_\_ (Кузнєцов О.О.)

Керівник \_\_\_\_\_ (Селіванова А.В.)

#### Консультанти:

з економічної частини \_\_\_\_\_ (Копайгородська Т.Г.)

з охорони праці \_\_\_\_\_ (Чорновол Н.І.)

з дотримання вимог ЄСКД \_\_\_\_\_ (Петрашова В.І.)

старший консультант \_\_\_\_\_ (Кунуп Т.В.)

#### До захисту допущений

Голова циклової комісії \_\_\_\_\_ (Кривченко Ю.В.)

Завідувач відділення \_\_\_\_\_ (Скорнякова О.В.)

Захист «23» \_\_\_\_\_ 06 2023 р. Протокол ДКК № 2

Оцінка ДКК \_\_\_\_\_ 3 (задовільно)

Секретар ДКК \_\_\_\_\_

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ВСП «ОДЕСЬКИЙ ТЕХНІЧНИЙ ФАХОВИЙ КОЛЕДЖ ОНТУ»

Відділення комп'ютерних систем Комісія КТ та III  
Спеціальність 121 «Інженерія програмного забезпечення»  
Освітня програма «Розробка програмного забезпечення»

ЗАТВЕРДЖУЮ:

Заст. дир. з НВР Беркань І.В.  
“ ” 2023 р.

## ЗАВДАННЯ

на дипломний проект (роботу)

Здобувачеві (здобувачці) освіти Кузнєцову Олександр Олександровичу  
(прізвище, ім'я, по батькові)

1. Тема проекту (роботи) Розробка програмного модуля "Електронний гаманець співробітника" для ERP-системи документообігу складу

затверджена наказом по коледжу від “ 13 ” 12 2022 р. № 306-A2-ОД

2. Термін здачі закінченого проекту (роботи) 26.06.23

3. Вихідні данні до проекту (роботи) Системи планування ресурсів підприємства.

Обґрунтування структури системи планування ресурсів. Обґрунтування вибору технічного обладнання для роботи з системою планування ресурсів підприємства. Розробка програмного модуля на мові PHP, MySQL.

4. Зміст розрахунково-пояснювальної записки (перелік питань, які необхідно розробити)

Вступ. 1 Технологічний розділ

2. Економічний розділ

4. Охорона праці. Висновки. Перелік використаних джерел

5. Перелік графічного (презентаційного) матеріалу (з точним зазначенням обов'язкових креслень, кількості слайдів)  
Презентація. 10 слайдів 1. Тема 2. Мета дипломного проекту 3. Зв'язок різних ERP- систем. 4. Основні

функціональні модулі. 5. Архітектура клієнт-сервер. 6. Модель системи в інтернет. 7. Схема структурная системи. 8.

Початок роботи в ERP- системі. 9. Скріншот програмного модуля «Електронний гаманець співробітника».

10. Висновки.

6. Консультанти по проекту (роботі), із зазначенням розділів проекту, що їх стосується

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв
1-2	Селіванова А.В.		
3	Копайгородська Т.Г.		
4	Чорновол Н.І.		
ЄСКД	Петрашова В.І.		

7. Дата видачі завдання 22.05.23

Керівник

(підпис)

Завдання прийняв до виконання

(підпис)

КАЛЕНДАРНИЙ ПЛАН

№ з/р	Назва етапів дипломного проекту (роботи)	Термін виконання етапів дипломного проекту (роботи)	Відмітка про виконання
1	Робота над вступом.	22.05.23	Виконано
2	Аналіз теми ДП та огляд літературних джерел	23.05.23	Виконано
3	Розробка конструкторського розділу ДП	24.05.23	Виконано
4	Обґрунтування вибору ПЗ	25.05.23	Виконано
3	Обґрунтування структури програмного забезпечення	26.05.23	Виконано
4	Розробка функціональної схеми	28.05.23	Виконано
5	Розробка ПЗ	29.05.23	Виконано
6	Виконання економічного розділу	30.05.23	Виконано
7	Розрахунок економічних показників проекту	01.06.23	Виконано
8	Виконання розділу «Охорона праці»	02.06.23	Виконано
9	Перевірка якості виконання розділу «Охорона праці»	03.06.23	Виконано
10	Виконання пояснювальної записки ДП	05.06.23	Виконано
11	Перевірка якості виконання пояснювальної записки ДП	06.06.23	Виконано
12	Виконання мультимедійної презентації	07.06.23	Виконано
13	Перевірка якості виконання презентації	08.06.23	Виконано
14	Малий захист	09.06.23	Виконано

Дипломник

(підпис)

Керівник

(підпис)



## ЗМІСТ

ВСТУП.....	7
1. ТЕХНОЛОГІЧНИЙ РОЗДІЛ .....	8
1.1 Огляд предметної галузі, аналіз технологій .....	8
1.1.1 Актуальність тематики роботи .....	9
1.1.2 Аналіз аналогів .....	11
Аналог 1 .....	11
Аналог 2 .....	11
Аналог 3 .....	12
1.2 Мета розробки, постановка задачі, вибір функціональних елементів реалізації системи.....	12
1.3 Вибір технологій та інструментів реалізації .....	14
1.3.1 Обрання мови програмування та середовища розробки .....	14
1.4 Основні етапи проєктування .....	17
1.4.1 Визначення вимог до системи.....	17
1.4.2 Проєктування інтерфейсу .....	18
1.5 Опис етапів реалізації програмного продукту .....	20
1.5.1 Опис реалізації програмного продукту .....	20
1.5.2 Опис реалізації інтерфейсу .....	21
1.6 Тестування, розробленого програмного продукту.....	22
2. ЕКОНОМІЧНА ЧАСТИНА .....	25
2.1 Резюме .....	25
2.2 Розрахунок ціни програмного продукту нормативним методом .....	26
2.2.1 Визначення трудомісткості розробки програмного забезпечення. ....	26

					РП 06. 13 000. 00 ДП ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		5

2.2.2 Розрахунок ціни програмного продукту. ....	31
3. ОХОРОНА ПРАЦІ .....	34
3.1 Вступ .....	34
3.2 Аналіз умов праці програміста .....	34
3.3 Розробка заходів з охорони праці .....	35
3.3.1 Вимоги до виробничих приміщень.....	35
3.3.2 Освітлення .....	35
3.3.3 Параметри мікроклімату .....	36
3.3.4 Шум і вібрація .....	37
3.3.5 Електромагнітні та іонізуючі випромінювання .....	38
3.3.6 Ергономічні вимоги до робочого місця.....	38
3.4 Пожежна безпека .....	39
ВИСНОВКИ .....	40
ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ .....	42
ДОДАТОК А Фрагмент лістингу .....	43
ДОДАТОК В Фрагмент лістингу.....	66

## ВСТУП

У сучасному інформаційному суспільстві мобільні програми відіграють важливу роль у нашому повсякденному житті. Вони забезпечують легкий доступ до різних сервісів та функцій, які допомагають нам у різних сферах діяльності. Одним із таких напрямків є безпека дорожнього руху. Статистика показує зростання аварій, пов'язаних із неправильним чи недостатнім знанням правил дорожнього руху (ПДР).

У зв'язку з цим створення та розвиток додатків, присвячених правилам дорожнього руху, стало першочерговим завданням. Додаток «Правила дорожнього руху» надає водіям, пішоходам та іншим учасникам дорожнього руху достовірну інформацію щодо правил та вимог, пов'язаних з безпекою дорожнього руху. Це дозволяє користувачам отримати доступ до останньої версії правил дорожнього руху, виконувати швидкий пошук та отримувати докладні пояснення різних правил, а також навчатися та покращувати свої навички водіння.

Результати цієї роботи будуть корисні водіям, які бажають бути в курсі останніх змін правил дорожнього руху та підвищувати свої навички водіння, а також усім учасникам дорожнього руху, які прагнуть безпеки дорожнього руху.

					РП 06. 13 000. 00 ДП ПЗ	Арк.
						7
Зм.	Арк	№ докум.	Підпис	Дата		

# 1. ТЕХНОЛОГІЧНИЙ РОЗДІЛ

## 1.1 Огляд предметної галузі, аналіз технологій

Розроблюваний проект призначений для вивчення Правил Дорожнього Руху України та тестування вже добутих знань.

Огляд предметної галузі:

Правила дорожнього руху (ПДР) є основою безпеки на дорогах. Вони є обов'язковими для всіх учасників дорожнього руху та спрямовані на запобігання аваріям та захист життя і здоров'я людей. Однак, через складність та обсяг ПДР, не завжди всі учасники дорожнього руху мають достатні знання про правила та вимоги, що може призводити до порушень та небезпек на дорозі.

У сучасній цифровій епохі мобільні додатки стають важливою складовою нашого повсякденного життя. Вони надають зручний доступ до різноманітних сервісів та функціональності, яка допомагає нам у різних сферах діяльності. Однією з таких сфер є безпека на дорогах. Статистика свідчить про зростання автомобільних пригод, пов'язаних з неправильним або недостатнім знанням ПДР.

У зв'язку з цим, створення та розвиток додатків, присвячених ПДР, стає актуальною задачею. Додаток ПДР надає можливість водіям, пішоходам та іншим учасникам дорожнього руху отримувати надійну інформацію про правила та вимоги, що стосуються безпеки на дорогах. Він дозволяє користувачу отримати доступ до актуальних версій ПДР, здійснювати швидкий пошук та отримувати детальні пояснення до різних правил, а також навчатися та покращувати свої навички водіння.

Метою даної дипломної роботи є розробка та реалізація додатку ПДР, який забезпечить зручний та доступний інтерфейс для отримання інформації про правила дорожнього руху. У роботі будуть досліджені існуючі додатки даного типу, проведений аналіз їх функціональності та користувацького досвіду, а також визначені ключові вимоги до розроблюваного додатку.

					РП 06. 13 000. 00 ДП ПЗ	Арк.
						8
Зм.	Арк	№ докум.	Підпис	Дата		

Для досягнення поставленої мети будуть виконані такі завдання: вивчення ПДР та їх актуальних змін, проектування користувацького інтерфейсу, розробка функціональності додатку, тестування та оптимізація його роботи, а також оцінка ефективності додатку серед користувачів.

### 1.1.1 Актуальність тематики роботи

Актуальність тематики "Приложение ПДД" обумовлена кількома факторами, які впливають на безпеку дорожнього руху та поведінку учасників дорожнього руху.

По-перше, Правила дорожнього руху є основою безпеки на дорогах. Актуальні та коректні знання ПДР є важливими для кожного водія, пішохода або учасника дорожнього руху, оскільки вони допомагають уникати аварійних ситуацій та запобігати травмам та загибелі.

По-друге, зростання автомобільного транспорту та швидкі темпи розвитку технологій ставлять перед учасниками дорожнього руху нові виклики та вимагають оновлення знань про ПДР. Нові типи транспорту, електромобілі, автономні автомобілі та інші інновації потребують адаптації правил та розуміння їх впливу на дорожню безпеку.

По-третє, зростання використання мобільних технологій та смартфонів надає нові можливості для розробки додатків, спрямованих на полегшення доступу до інформації про ПДР. Мобільні додатки, які надають швидкий та зручний доступ до правил, актуальних оновлень та додаткових пояснень, можуть значно поліпшити розуміння та дотримання ПДР.

Отже, розробка застосунку ПДР є актуальним завданням, оскільки вона сприяє підвищенню обізнаності учасників дорожнього руху щодо Правил дорожнього руху, полегшує доступ до актуальної інформації та сприяє покращенню дорожньої безпеки.

					РП 06. 13 000. 00 ДП ПЗ	Арк.
						9
Зм.	Арк	№ докум.	Підпис	Дата		

Основні проблеми предметної галузі та засоби їх вирішення за допомогою розробки програмного забезпечення

У предметній галузі вивчення правил дорожнього руху (ПДР) існують кілька основних проблем, з якими стикаються учасники дорожнього руху. Розробка програмного забезпечення може виступати як засіб їх вирішення. Основні проблеми та способи їх вирішення таким засобом включають:

1. Недостатні знання ПДР: Багато водіїв та пішоходів мають обмежені знання про ПДР або не мають доступу до актуальної інформації. Розробка мобільного додатку ПДР надає можливість користувачам отримувати зручний доступ до правил дорожнього руху, включаючи актуалізовану інформацію, пояснення та приклади.

2. Незручність доступу до ПДР: Класичні паперові видання ПДР можуть бути незручними для використання, особливо в дорозі або в ситуації, коли потрібно швидко знайти конкретне правило. Розробка додатку ПДР дозволяє створити зручний та легкодоступний інтерфейс, який дає можливість швидко знайти потрібну інформацію та використовувати її в реальному часі.

3. Зміни в ПДР: ПДР постійно оновлюються, щоб враховувати нові технології, типи транспорту та сучасні реалії дорожнього руху. Розробка додатку ПДР дозволяє швидко оновляти і вносити зміни до правил у вигляді оновлень, що забезпечує користувачів актуальною та точною інформацією.

4. Навчання та набуття навичок: Багато водіїв потребують додаткового навчання та покращення навичок водіння. Додаток ПДР може включати розділи з навчальними матеріалами, відеоуроками та тестами, що допоможуть водіям покращити свої знання та вміння.

					РП 06. 13 000. 00 ДП ПЗ	Арк.
						10
Зм.	Арк	№ докум.	Підпис	Дата		

5. Свідомість та безпека: Зростання свідомості щодо ПДР та забезпечення безпеки на дорогах є важливими аспектами. Розробка додатку ПДР може включати розділи з важливими порадами, інформацією про безпекові заходи та можливість подачі повідомлень про порушення ПДР.

Застосування програмного забезпечення, зокрема розробка застосунку ПДР, може сприяти вирішенню цих проблем, поліпшити розуміння ПДР та забезпечити безпеку та свідомість учасників дорожнього руху.

### **1.1.2 Аналіз аналогів**

Аналіз аналогів є важливим етапом при розробці додатку ПДР, оскільки він дозволяє оцінити наявні рішення на ринку та зробити висновки щодо їх переваг та недоліків. Декілька популярних аналогів додатків ПДР включають:

#### **Аналог 1**

“Тести ПДД 2023” - Офіційний мобільний додаток, розроблений Державтоінспекцією України, який містить повний текст Правил дорожнього руху України. Він надає зручний доступ до правил, знаків та обмежень, а також пропонує інтерактивні тести для перевірки знань.

<https://vodiy.ua/ru/process/>

#### **Аналог 2**

“ПДД 2023 Украина” - Мобільний додаток, який містить текст Правил дорожнього руху України, знаки та обмеження. Він також пропонує функцію навчання та інтерактивні тести для перевірки знань.

<https://xn--h1afceeb4a.xn--j1amh/>

					РП 06. 13 000. 00 ДП ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		11

Аналог 3

“Официальные Тесты ПДД Украины ” - Додаток, який містить інформацію про правила дорожнього руху, знаки та обмеження в Україні. Він надає функцію навчання, інтерактивні тести та можливість перевірити свої знання.

<https://pdr.infotech.gov.ua/>

## **1.2 Мета розробки, постановка задачі, вибір функціональних елементів реалізації системи**

Метою розробки додатку ПДД є створення зручного та доступного інструменту для учасників дорожнього руху, який надасть актуальну інформацію про Правила дорожнього руху, допоможе уникати порушень та підвищить рівень дорожньої безпеки.

Для досягнення цієї мети ставляться наступні задачі:

1. Розробка зручного інтерфейсу: Створення інтуїтивно зрозумілого та зручного інтерфейсу, що дозволяє користувачам швидко та легко знаходити необхідну інформацію про ПДР. Інтерфейс повинен бути приємним для використання та підтримувати зручну навігацію.

2. Постачання актуальної інформації: Забезпечення наявності оновленої та достовірної інформації про ПДР. Додаток повинен регулярно оновлюватися з урахуванням змін у правилах дорожнього руху та надавати користувачам актуальні матеріали.

3. Навчальні матеріали: Розробка розділу з навчальними матеріалами, включаючи інтерактивні тести, відеоуроки та пояснення. Це дозволить користувачам покращити свої знання ПДР та набути необхідні навички для безпечного руху.

					РП 06. 13 000. 00 ДП ПЗ	Арк.
						12
Зм.	Арк	№ докум.	Підпис	Дата		

4. Функції персоналізації: Забезпечення можливості налаштування додатку відповідно до потреб користувача. Включення функцій, таких як закладки, підсвічування важливих правил або настройка нагадувань, дозволить кожному користувачу налаштувати додаток згідно зі своїми особистими потребами.

5. Взаємодія з користувачами: Забезпечення можливості зворотного зв'язку, включаючи можливість повідомляти про помилки, неправильну інформацію або надавати пропозиції щодо покращення додатку. Це дозволить покращити функціонал та задовольнити потреби користувачів.

Вибір функціональних елементів для реалізації системи базується на аналізі потреб користувачів та кращих практик існуючих додатків. Ключові функціональні елементи можуть включати:

1. База з правилами: Розробка та підтримка бази даних з актуальними Правилами дорожнього руху, яка включатиме пояснення, ілюстрації та приклади.

2. Навігація та пошук: Реалізація зручної системи навігації, що дозволяє швидко знайти необхідну інформацію. Включення можливості пошуку за ключовими словами та категоріями.

3. Навчальні матеріали: Розробка розділу з навчальними матеріалами, включаючи тести, відеоуроки та інтерактивні завдання.

4. Персоналізація: Включення можливості налаштування додатку, створення закладок, підсвічування важливих правил та настройка нагадувань.

5. Зворотний зв'язок: Реалізація механізму для зворотного зв'язку з користувачами, таких як повідомлення про помилки, пропозиції та оновлення.

					РП 06. 13 000. 00 ДП ПЗ	Арк.
						13
Зм.	Арк	№ докум.	Підпис	Дата		

Ці функціональні елементи відображаються в завданнях розробки та сприяють досягненню мети створення зручного та корисного додатку ПДР.

Проаналізував по вже діючим програмам які знайшов:

<https://sovety.pp.ua/index.php/ru/stati/android/drugoe/1848-uchim-pravila-dorozhnogo-dvizheniya-na-mobilnykh-ustrojstvakh-pk-i-onlajn-ukraina>

### **1.3 Вибір технологій та інструментів реалізації**

#### **1.3.1 Обрання мови програмування та середовища розробки**

При обиранні мови програмування та середовища розробки для моєї програми з вивчення правил дорожнього руху я враховую наступні фактори:

1. Функціональність мови програмування: Я переконуюся, що обрана мова програмування має всі необхідні функції та бібліотеки для реалізації функціональності моєї програми. Важливо, щоб я міг легко працювати з базами даних, обробляти дані, візуалізувати інформацію та взаємодіяти з користувачем.

2. Мій досвід розробки: Я враховую свій власний досвід та вміння у використанні певних мов програмування та середовищ розробки. Якщо у мене є досвід у певній мові програмування, це може полегшити розробку та підтримку моєї програми.

3. Екосистема та підтримка: Я досліджую екосистему мови програмування та середовищ розробки, звертаючи увагу на наявність багатофункціональних інструментів, бібліотек, форумів та підтримки спільноти. Це допоможе мені вирішувати поточні завдання та забезпечить доступ до необхідної документації та ресурсів.

З огляду на ці фактори, я намагаюся знайти оптимальне поєднання мови програмування та середовища розробки, яке задовольняє мої потреби та дозволить ефективно реалізувати програму з вивчення правил дорожнього руху.

					РП 06. 13 000. 00 ДП ПЗ	Арк.
						14
Зм.	Арк	№ докум.	Підпис	Дата		

Я обрав мову програмування C++ для реалізації своєї програми з вивчення правил дорожнього руху з кількох причин:

1. Продуктивність: C++ є мовою низького рівня, що дозволяє ефективно використовувати ресурси комп'ютера. Це особливо важливо для програм, які можуть містити великі обсяги даних або потребують швидкої обробки, наприклад, відео або графіки.

2. Близькість до мови машинного коду: C++ дозволяє пряме керування пам'яттю та має можливості для оптимізації роботи програми. Це важливо, коли потрібно забезпечити швидку роботу програми та оптимальне використання ресурсів.

3. Розширюваність: C++ надає можливість створювати бібліотеки та використовувати їх у програмі. Це дозволяє легко розширювати функціональність програми, використовуючи існуючі рішення та інструменти.

4. Підтримка ООП: C++ підтримує об'єктно-орієнтоване програмування, що сприяє створенню структурованих та модульних програм. Це полегшує розробку, тестування та підтримку коду.

5. Широке застосування: C++ є однією з найпопулярніших мов програмування, використовуваних у різних сферах, включаючи розробку системного програмного забезпечення, вбудовані системи, графічні додатки та багато іншого. Це означає, що є велика кількість ресурсів, документації та підтримки спільноти для C++.

Враховуючи ці переваги, мова програмування C++ відповідає моїм потребам для реалізації програми з вивчення правил дорожнього руху, забезпечуючи продуктивність, оптимізацію ресурсів та розширюваність, а також надаючи можливості об'єктно-орієнтованого програмування.

					РП 06. 13 000. 00 ДП ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		15

Я обрав середовище розробки Visual Studio для реалізації своєї програми з вивчення правил дорожнього руху з кількох причин:

1. Інтегроване середовище розробки: Visual Studio надає повноцінне інтегроване середовище розробки (IDE), що включає в себе текстовий редактор, компілятор, відлагоджувач, систему керування версіями і багато інших корисних інструментів. Це забезпечує зручну та продуктивну роботу з кодом.

2. Підтримка мови програмування C++: Visual Studio має вбудовану підтримку мови програмування C++ і надає багато корисних функцій та інструментів для розробки на цій мові. Це дозволяє зручно писати, налагоджувати і тестувати код C++.

3. Візуальний відлагоджувач: Visual Studio має потужний візуальний відлагоджувач, який дозволяє крок за кроком відлагоджувати програму, аналізувати значення змінних, виконувати точки зупинки і багато іншого. Це спрощує виявлення та виправлення помилок у програмі.

4. Платформна незалежність: Visual Studio підтримує розробку програм для різних платформ, включаючи Windows, Linux та мобільні пристрої. Це дозволяє легко переносити програму на різні середовища та платформи.

5. Велика спільнота користувачів: Visual Studio є одним з найпопулярніших середовищ розробки, що має велику та активну спільноту користувачів. Це означає, що є велика кількість ресурсів, документації, плагінів та форумів, де можна знайти підтримку та відповіді на питання.

З огляду на ці переваги, я вважаю, що Visual Studio є ідеальним середовищем розробки для мого проекту. Воно надає зручність, потужність і ефективність у

					РП 06. 13 000. 00 ДП ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		16

розробці програми з вивчення правил дорожнього руху на мові програмування C++.

<https://visualstudio.microsoft.com/>

<https://medium.com/nuances-of-programming/%D1%8F%D0%B7%D1%8B%D0%BA%D0%B8-c-%D0%B8-c-%D0%B3%D0%B4%D0%B5-%D0%B5%D1%89%D1%91-%D0%B8%D1%85-%D0%B8%D1%81%D0%BF%D0%BE%D0%BB%D1%8C%D0%B7%D1%83%D1%8E%D1%82-%D0%B8-%D0%B7%D0%B0%D1%87%D0%B5%D0%BC-9ff56559d5bf>

## **1.4 Основні етапи проєктування**

### **1.4.1 Визначення вимог до системи**

Визначення вимог до системи є важливим етапом у розробці додатку ПДД. Вимоги до системи визначають функціональні та нефункціональні характеристики, які повинна мати система для задоволення потреб користувачів і досягнення мети розробки. Основними категоріями вимог є:

1. Функціональні вимоги: Це конкретні функції і можливості, які має надавати додаток ПДД. До них можуть належати:

- відображення актуальних Правил дорожнього руху з ілюстраціями та поясненнями;
- пошук і навігація за ключовими словами або категоріями правил;
- надання навчальних матеріалів, включаючи тести та відеоуроки;
- можливість персоналізації, така як створення закладок або налаштування нагадувань;
- механізм зворотного зв'язку, що дозволяє користувачам повідомляти про помилки та пропозиції.

2. Нефункціональні вимоги: Це характеристики системи, які не стосуються безпосередньо її функціональності, але визначають її якість, продуктивність та безпеку. До них можуть належати:

					РП 06. 13 000. 00 ДП ПЗ	Арк.
						17
Зм.	Арк	№ докум.	Підпис	Дата		

- продуктивність: швидкість відгуку системи, обробка великого обсягу даних, оптимізація роботи з базою даних;
- надійність: стабільна робота системи, відновлення після збоїв, захист від помилок та некоректних даних;
- безпека: захист даних користувачів, автентифікація та авторизація, захист від несанкціонованого доступу;
- сумісність: підтримка різних платформ та операційних систем, які можуть використовуватись користувачами;
- ергономіка та користувацький інтерфейс: зручність використання, інтуїтивно зрозумілий інтерфейс, адаптація до різних розмірів екранів.

Для визначення вимог до системи можна провести аналіз потреб користувачів, провести співбесіди зі зацікавленими сторонами та врахувати вимоги, впливаючі з контексту розробки додатку ПДД. Одержані вимоги слід документувати і використовувати як основу для подальшого процесу розробки системи.

#### 1.4.2 Прєєктування інтерфейсу

Прєєктування інтерфейсу є важливою складовою розробки програми з вивчення правил дорожнього руху. Головна мета прєєктування інтерфейсу - створення зручного, зрозумілого та ефективного інтерфейсу, що дозволяє користувачеві легко взаємодіяти з програмою і досягати своїх цілей

Основні кроки прєєктування інтерфейсу включають:

1. Збір вимог користувачів: Важливо врахувати потреби та очікування користувачів програми. Це можна зробити шляхом опитувань, спостережень, інтерв'ю з майбутніми користувачами.

					РП 06. 13 000. 00 ДП ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		18

2. Створення структури: Визначення основних екранів та їх взаємозв'язку. Наприклад, головний екран, екран навчання, екран тестування тощо. Важливо розташувати елементи інтерфейсу логічно та зручно для користувача.

3. Розробка вигляду інтерфейсу: Вибір кольорової палітри, шрифтів, графічних елементів, що відповідають загальному стилю програми. Важливо створити чистий та привабливий дизайн, щоб залучити користувачів.

4. Розташування елементів керування: Розміщення кнопок, полів вводу, панелей інструментів та інших елементів керування на екрані. Важливо дотримуватися принципу зручності та інтуїтивності, щоб користувачам було легко знаходити та використовувати необхідні елементи.

5. Навігація: Розробка зручної системи навігації, що дозволяє користувачеві легко переміщатися між різними екранами та функціями програми. Наприклад, використання меню, вкладок, пошукових полів тощо.

6. Тестування ітерфейсу: Перевірка роботи інтерфейсу з реальними користувачами та збір їхнього фідбеку. Це допоможе виявити можливі проблеми та внести необхідні зміни для поліпшення інтерфейсу.

Важливо також враховувати принципи юзабіліті та доступності, щоб забезпечити зручну взаємодію для всіх користувачів, незалежно від їхніх навичок та особливостей.

Проектування інтерфейсу вимагає уваги до деталей та розуміння потреб користувачів. Відправною точкою є створення прототипу інтерфейсу, який можна тестувати та удосконалювати перед реалізацією програми.

Під час проектування інтерфейсу для моєї дипломної роботи з програми з вивчення правил дорожнього руху, я врахував кілька ключових факторів. По-

					РП 06. 13 000. 00 ДП ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		19

перше, я зосередився на створенні інтерфейсу, який був би зрозумілим і легким у використанні для користувачів будь-якого рівня технічної підготовки.

Збір вимог користувачів був важливим етапом проектування. Я провів опитування серед потенційних користувачів, щоб з'ясувати їхні потреби та очікування щодо програми. Це дозволило мені отримати цінний фідбек та зрозуміти, які функції та можливості є найбільш важливими для користувачів.

На основі зібраних вимог, я створив структуру інтерфейсу, визначивши основні екрани та їх взаємозв'язок. При цьому я дбав про логічність та зручність розміщення елементів інтерфейсу. Наприклад, головний екран містить основні функції програми, такі як навчання, тестування та статистика.

Окрему увагу я приділив дизайну інтерфейсу. Я вибрав приємну кольорову палітру та чіткий шрифт, щоб забезпечити зручність читання та сприяти зосередженості користувача на змісті програми. Також я використовував піктограми та інші графічні елементи, щоб доповнити текстовий інтерфейс і зробити його більш інтуїтивно зрозумілим.

Я використав меню та кнопки для швидкого доступу до основних функцій.

У процесі розробки інтерфейсу я також здійснив тестування залученими користувачами, щоб отримати їхній фідбек та внести необхідні зміни для поліпшення інтерфейсу.

<https://habr.com/ru/articles/699592/>

## **1.5 Опис етапів реалізації програмного продукту**

### **1.5.1 Опис реалізації програмного продукту**

При розробці моєї дипломної роботи, я зосереджувався на створенні зручного та привабливого інтерфейсу програми з вивчення правил дорожнього руху. Моя мета полягала в тому, щоб забезпечити користувачам зручну та ефективну платформу для навчання та тестування їхніх знань.

					РП 06. 13 000. 00 ДП ПЗ	Арк.
						20
Зм.	Арк	№ докум.	Підпис	Дата		

Головний екран макету буде включати логотип або назву програми, щоб візуально ідентифікувати її. На цьому екрані користувач зможе вибрати потрібну опцію, таку як "почати тестування".

Розділ "Тестування" надасть користувачам можливість перевірити свої знання через інтерактивні тести.

Макет інтерфейсу буде враховувати принципи юзабіліті, такі як чіткість, послідовність та легка навігація. Я планую використовувати приємні кольори, логічно розташовувати елементи керування та забезпечити зручність взаємодії для користувачів будь-якого рівня технічної підготовки.

Враховуючи потреби користувачів та мої цілі, я працюватиму над розробкою інтуїтивно зрозумілого та привабливого інтерфейсу, що допоможе користувачам ефективно вивчати правила дорожнього руху та підготуватися до тестування

### **1.5.2 Опис реалізації інтерфейсу**

Насамперед, я створив основний інтерфейс програми, який складається з простого вікна, де розміщені необхідні елементи, такі як кнопки та графічні зображення дорожніх знаків на фоні. Цей інтерфейс буде зрозумілим та простим для користувачів.

Наступним кроком є реалізація функціональності програми. Це наступні елементи:

1. Я розмістив на основному інтерфейсі кнопки "Почати" (яка переносить користувача до тестів) та "Вихід" (кнопка відповідає за повне закриття програми)

					РП 06. 13 000. 00 ДП ПЗ	Арк.
						21
Зм.	Арк	№ докум.	Підпис	Дата		

2. Тестування знань: Я створив блок питань і варіантів відповідей, де користувачі зможуть відповідати на запитання про правила дорожнього руху.

3. Тести складаються всього лиш з дванадцяти питань які я зробив тому що це примітивна програма для огляду

4. На вікнах тестів розташовано дві або три кнопки “Завершити тест” (відповідно завершує тест та переносить користувача на вікно з набраними балами) “Назад” (повертає користувача до попереднього запитання, якщо він захоче відповісти на запитання на яке він не дав відповіді, або хоче зміти рішення ) “Далі” (переносить користувача до наступного питання)

Така примітивна програма з вивчення правил дорожнього руху має на меті надати користувачам основні знання та перевірити їх розуміння. Вона може бути створена швидко та просто, засновуючись на простих елементах інтерфейсу та логіці програми.

### 1.6 Тестування, розробленого програмного продукту

Запуск може відбуватись як з Visual Studio так і за ярлика, після чого користувача зустрічає головне меню програми



Рис. 1 “Головна”

Кнопка “Почати” починає тест — працює

					РП 06. 13 000. 00 ДП ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		22

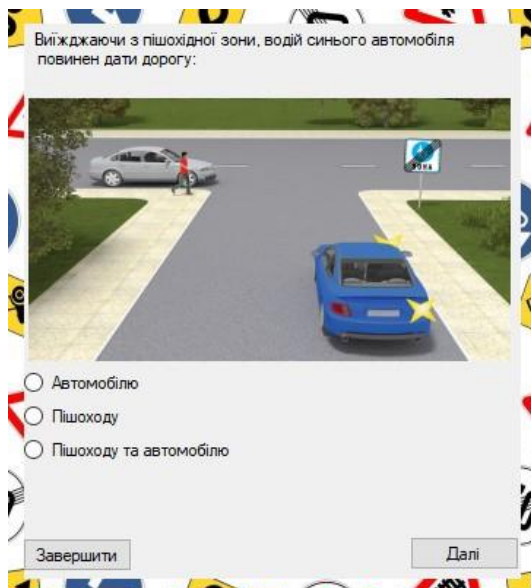


Рис. 2 “Перше питання”

Кнопка “Далі” переносить до наступного питання — працює

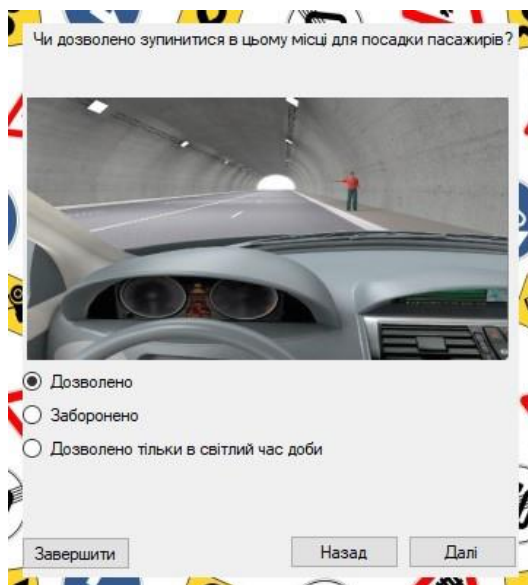


Рис. 3 “Питання з обраною відповіддю”

Кнопка “Назад” повертає до попереднього питання — працює

Чек-бокси — працюють

Кнопка “Завершити” завершує тест, сумує бали за правильні відповіді та виводить на вікні — працює



Рис. 4 “Кінець тесту та набрані бали”

Кнопка “Закінчити” повертає до головного меню програми - працює

Кнопка “Вихід” призначена для виходу з програми - працює

					РП 06. 13 000. 00 ДП ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		24

## 2. ЕКОНОМІЧНА ЧАСТИНА

### 2.1 Резюме

У цій дипломній роботі я розробив програму з вивчення правил дорожнього руху. Головною метою роботи є створення зручного та ефективного інструменту, який допоможе користувачам вивчити правила дорожнього руху та підготуватися до тестування.

У процесі роботи я проводжу огляд предметної галузі, аналізую технології та аналоги, що вже існують на ринку. Визначаю актуальність тематики та роблю висновки про необхідність розробки даної програми.

Для реалізації програмного продукту я обираю мову програмування C++ та середовище розробки Visual Studio. Ці інструменти дозволяють забезпечити ефективність, стабільність та зручність розробки.

Під час реалізації програми я звертаю увагу на зручність взаємодії з користувачем, надаючи зрозумілий та привабливий інтерфейс.

Усі етапи розробки програми проводжу з урахуванням вимог до системи та потреб користувачів. Після завершення реалізації я провів тестування та виправив помилки.

Результатом моєї роботи є функціональна програма з вивчення правил дорожнього руху, яка забезпечить користувачам зручну та ефективну платформу для тестування їх знань.

Ефективність кожного програмного продукту визначається його якістю та ефективністю процесу розробки. Якість ПП визначається наступними

					РП 06. 13 000. 00 ДП ПЗ	Арк.
						25
Зм.	Арк	№ докум.	Підпис	Дата		

складовими: з точки зору користувача; з позиції використання ресурсів; виконання вимог до програмного забезпечення. Оцінка якості програмного продукту включає визначення трудомісткості і вартості його створення.

Проведемо розрахунки визначення трудомісткості розробки та вартості даного програмного продукту.

## 2.2 Розрахунок ціни програмного продукту нормативним методом

### 2.2.1 Визначення трудомісткості розробки програмного забезпечення.

Тривалість розробки програмного продукту залежить від його обсягу, трудомісткості розробки, кваліфікації виконавців, а також планових термінів, визначених умовами ринку.

Методом структурної аналогії по відповідних каталогах аналогів програмного забезпечення визначається обсяг програмних засобів, у тисячах умовних машинних команд програми аналога.

У таблиці 2.1 представлені аналоги програмного забезпечення, функції яких, у більшому або меншому ступені, виконує розроблений програмний продукт.

Таблиця 2.1 Каталог аналогів

Найменування ПП	Обсяг функції ПП – $V_o$ , усл. машинних командах.
1. ПП СУБД	2500 – 9800
2. Комплексні системи ведення БД	950 – 7430
3. ПП введення інформації	1060 – 5750
4. ПП оптимізації розрахунків	1300 – 4200
5. ПП автоматизації засобів по каталогу	680 – 7000
6. ПП автоматизованих розрахунків	1300 – 8600
7. ПП загальної математики і ПП імітаційного моделювання	7800 – 8800

					РП 06. 13 000. 00 ДП ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		26

Продовження таблиці 2.1

8. ПП організації обчислювального процесу	13000 – 10200
9. ПП оптимізаційних розрахунків	1300 – 4200

Вибравши аналог ПП, що містить  $V_0$  в умовних машинних командах, трудомісткості визначати на основі табл.2.2

Таблиця.2.2

Обсяг ПП, тис.умов.машинних команд	Норма часу, люд/год
1.00	229
2.00	244
3.00	262
4.00	283
5.00	306
6.00	330
7.00	357
8.00	385
9.00	414
10.00	445
12.00	510
14.00	580
16.00	654
18.00	731
20.00	812

На підставі отриманого значення, по таблиці 2.2, визначається укрупнена норма часу на розробку аналога програмного забезпечення, яка коректується поправочним коефіцієнтом враховуючої умови розробки ПП, тобто в умовах комп'ютера,  $K_k=0,7\div 0,8$ :  $T^a_p = 262 \times 0,8 = 209,6$  (люд/годин).

Трудомісткість програмного продукту визначається по кожному етапу розробки окремо на підставі трудомісткості аналога з урахуванням складності розробки, ступеня новизни і ступеня використання в розробці стандартних модулів на підставі формул:

$$1. \text{ Розробка технічного завдання } T_{ТЗ} = T^a_p \times L_1 \times K_H \quad (2.1)$$

$$2. \text{ Розробка технічного проекту } T_{ТП} = T^a_p \times L_2 \times K_H \quad (2.2)$$

$$3. \text{ Розробка робочого проекту } T_{РП} = T^a_p \times L_3 \times K_H \times K_T \quad (2.3)$$

Для розрахунку необхідні наступні коефіцієнти:

$L_i$  – питома вага  $i$ -го етапу розробки (див. табл. 2.3.);

$K_H$  – поправочний коефіцієнт, що враховує ступінь новизни (див. табл. 2.4.);

$K_T$  – поправочний коефіцієнт, що враховує ступінь використання в розробці типових програм (див. табл. 2.5.).

Таблиця 2.3. Значення питомих коефіцієнтів трудомісткості стадії в загальній трудомісткості розробки ПП.

Код стадії	Ступінь новизни		
	А	Б	В
ТЗ ( $L_1$ )	0,15	0,12	0,12
ТП ( $L_2$ )	0,16	0,15	0,11
РП ( $L_3$ )	0,55	0,58	0,61

Таблиця 2.4. Значення поправочного коефіцієнта, що враховує ступінь новизни

Код ступеня новизни	Ступінь новизни	Значення $K_H$
А	Принципово нові ПП	1,75 – 1,2
Б	ПП–розвиток визначеного параметричного ряду	1,0 – 0,8
В	ПП маючий аналог	0,7

Тому що розробка системи є ПП, що має аналоги програмних продуктів, то код ступеня новизни для мого ПП – В, а значення коефіцієнта  $K_H = 0.7$ . По таблиці 2.4, знаючи код ступеня новизни, тепер можна визначити значення питомих коефіцієнтів трудомісткості (по таблиці 2.3):

$$L_1 = 0,12$$

$$L_2 = 0,11$$

$$L_3 = 0,61$$

Таблиця 2.5. Значення коефіцієнта ступеня використання в розробці типових програм

Ступінь охоплення реалізованих функцій розроблювального ПП типовими програмами, %	Значення $K_T$
60 і вище	0,6
40-60	0,7
20-40	0,8
До 20	0,9

У розробленому програмному продукті використовується від 60 до вище відсотків існуючих функцій, це значить, що  $K_T = 0,6$  Тепер потрібно розрахувати трудомісткість по кожному етапу окремо:

Трудомісткість технічного завдання

$$T_{ТЗ} = T^a p * L_1 * K_H = 209,6 \times 0,12 \times 0,7 = 17,60 \text{ (люд/годин)} \quad (2.1)$$

Трудомісткість розробки технічного проекту

$$T_{ТП} = T^a p * L_2 * K_H = 209,6 \times 0,11 \times 0,7 = 16,13 \text{ (люд/годин)} \quad (2.2)$$

Трудомісткість розробки робочого проекту

$$T_{РП} = T^a p * L_3 * K_H * K_T = 209,6 \times 0,61 \times 0,7 \times 0,6 = 53,69 \text{ (люд/годин)} \quad (2.3)$$

Для подальших розрахунків необхідно визначити кількість папера, витраченого на кожен етап.  $N_{ТЗ} = 2$  (стр),  $N_{ТП} = 13$  (стр),  $N_{РП} = 4$  (стр),  $N_{ПЗ} = 38$  (стр) – технічне завдання, розробка технічного проекту, розробка робочого проекту, пояснювальна записка відповідно. Розрахунок зведений у таблицю 2.6

Таблиця 2.6. Розрахунок трудомісткості ПП

Найменування етапів	Розрахунок, години.		
	Розробка ПП	Контроль керівника	Нормоконтроль
1.ТЗ	$T_{ТЗ} = 17,60$	$T_{КК} = 0,7 * N_{ТЗ}$ $= 0,7 \times 2 = 1,4$	$T_{НК} = 0,15 * N_{ТЗ}$ $= 0,15 \times 2 = 0,3$
2.Розробка ТП	$T_{ТП} = 16,13$	$T_{КК} = 0,7 * N_{ТП}$ $= 0,7 \times 13 = 9,1$	$T_{НК} = 0,15 * N_{ТП}$ $= 0,15 \times 13 = 1,95$
3.Розробка РП	$T_{РП} = 53,69$	$T_{КК} = 0,7 * N_{РП}$ $= 0,7 \times 4 = 2,8$	$T_{НК} = 0,15 * N_{РП}$ $= 0,15 \times 4 = 0,6$
4.Розробка пояснювальної записки	$T_{ПЗ} = 1,5 * N_{ПЗ}$ $= 1,5 \times 38 = 57$	$T_{КК} = 0,7 * N_{ПЗ}$ $= 0,7 \times 38 = 26,6$	$T_{НК} = 0,15 * N_{ПЗ}$ $= 0,15 \times 38 = 5,7$
Усього, в т.ч.:	$\Sigma T = 192,87$		
- на розробку	$\Sigma T_p = 144,42$		
- контроль керівника		$\Sigma T_{КК} = 39,9$	
- нормоконтроль			$\Sigma T_{НК} = 8,55$

На основі таблиці 2.6 розрахуємо тривалість розробки в роках:

					РП 06. 13 000. 00 ДП ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		30

$$T_{пп} = T / (8,0 * 0,73 * 360) = 192,87 / (8,0 * 0,73 * 360) = 0,09 \text{ (р)}, (2.4)$$

де

8,0 – тривалість робочого дня;

0,73 – коефіцієнт перекладу в календарні дні;

### 2.2.2 Розрахунок ціни програмного продукту.

У цьому розділі для визначення ціни розраховуємо основну заробітну плату виконавців, матеріальні витрати, вартість машино – години і витрати на розробку ПО. Розрахунок основної заробітної плати виконавців приведений у таблиці 6.7. Відповідно до статті 8 «Закону про Державний бюджет України на 2023» встановлено мінімальну заробітну плату у місячному розмірі з 1 січня 2023 року – 6700 гривень; мінімальну погодинну тарифну ставку – 40.46 грн.

Таблиця 2.7 Розрахунок основної заробітної плати виконавців.

Найменування робіт	Трудомісткість робіт, роб.години	Годинна тарифна ставка,грн..	Розрахунок, грн.
1.Розробка ПП	$\Sigma T_p = 144,42$	40,46	$Z_{op} = 5843,2$
2.Контроль керівника	$\Sigma T_{kk} = 39,9$	100	$Z_{okk} = 3990$
3.Нормоконтроль	$\Sigma T_{нк} = 8,55$	100	$Z_{онк} = 855$
Усього ( $Z_o$ )	-	-	$\Sigma Z_o = 10688,2$

Розрахунок основної заробітної плати виконуємо по формулі:

$$Z_o = T_{mj} \times Z_{год}, (2.5)$$

					РП 06. 13 000. 00 ДП ПЗ	Арк.
						31
Зм.	Арк	№ докум.	Підпис	Дата		

де

$T_{mj}$  – трудомісткість  $j$  – того виду робіт, робоч. год;

Згод – погодинна тарифна ставка, грн.

Зробимо розрахунок матеріальних витрат на розробку ПП. Розрахунок зведемо в таблицю 2.8

Таблиця 2.8 Розрахунок матеріальних витрат на розробку ПО

Найменування матеріальних витрат	Тип, модель	Кількість, шт	Ціна одиниці, грн.	Вартість, грн.
Папір	Лист А4	42	0,50	20
.....				
Разом	-	-	-	$V_{mi}=20$
Транспортно – заготівельні Витрати 10%				$V_{трз} = 0,1 \times V_{m1}$ $=0,1 \times 20=2$
Усього				$V_m=V_{m1}+V_{трз}=22$

На підставі отриманих даних по окремих статтях витрат складена калькуляція планової собівартості в цілому ПП за формою, приведеною в таблиці 2.9

Таблиця 2.9. Розрахунок статей витрат планової собівартості

Стаття витрат	Значення, грн.	Формула розрахунку
1. Матеріали	22	$V_m$ (див. табл. 2.8)
2. Основна заробітна плата	10688,2	$Z_o$ (див. табл. 2.7)
3.Додаткова заробітна плата	1068,82	$Z_d = 0,1 \times Z_o$ $=0,1 \times 10688,2$

Продовження таблиці 2.9

4. Відрахування до єдиного фонду соціального внеску	2586,5	$\text{Вє.с.в.} = 0,22 \times (\text{З}_0 + \text{З}_д)$ $= 0,22 \times (10688,2 + 1068,82)$
5. Накладні витрати	6412,92	$\text{Внак.} = 0,6 \times \text{З}_0$ $= 0,6 \times 10688,2$
6. Повна собівартість	21078,44	$\text{С}_{\text{пов}} = \text{В}_м + \text{З}_0 + \text{З}_д + \text{Вє.с.в.} + \text{Внак.}$ $= 22 + 10688,2 + 1068,82 + 2586,5$ $+ 6412,92$

Розмір прибутку, що включається в ціну, визначається по наступній формулі:

$$\text{П} = (\text{С}_{\text{пов}} * \text{Р}) / 100 = (21078,44 \times 10) / 100 = 2107,8 \text{ (грн)}, \text{ (2.6)}$$

Де  $p$  – плановий рівень рентабельності (10-15%).

Оптова ціна (кошторисна вартість) визначається по формулі:

$$\text{Ц}_0 = \text{С}_{\text{пов}} + \text{П} = 21078,44 + 2107,8 = 23186,24 \text{ (грн)}; \text{ (2.7)}$$

Податок на додану вартість визначається по наступній формулі:

$$\text{ПДВ} = 0,2 * \text{Ц}_0 = 0,2 \times 23186,24 = 4637,24 \text{ (грн)}; \text{ (2.8)}$$

Виходячи з отриманих даних, ціна реалізації розробленого програмного продукту на основі наступної формули, становитиме:

$$\text{Ц}_р = \text{Ц}_0 + \text{ПДВ} = 23186,24 + 4637,24 = 27823,48 \text{ грн}; \text{ (2.9)}$$

### 3. ОХОРОНА ПРАЦІ

#### 3.1 Вступ

У зв'язку з розвитком технічного та наукового прогресу дуже велику роль грає можливість безпечного виконання людьми своїх обов'язків у якості працівників.

Охорона праці є однією з найважливіших прогресивних форм наукової охорони праці, за можливості зведення до мінімуму ручної, небезпечної, малокваліфікованої праці та створення умов для мінімізації або виключення професійних захворювань з життя людей. На робочому місці будь-якої людини повинні бути створенні міри захисту від можливого впливу шкідливих або небезпечних факторів праці та виробництва.

Даний розділ дипломного проекту визначає оптимальні умови праці програміста

#### 3.2 Аналіз умов праці програміста

У наш час комп'ютерна техніка використовується майже у всіх областях діяльності людини. При роботі з комп'ютерною технікою людина піддається досить немалій кількості небезпечних або шкідливих виробничих факторів, таких як електромагнітні поля, інфрачервоне випромінювання, шумів, вібрацій, електричний струм тощо. Робота з комп'ютером являє собою велику розумову та емоційну напругу, нервову навантаження, великою інтенсивністю зорової роботи і досить великим навантаженням на руки при роботі з клавіатурою та мишею. Тому величезне значення має правильне розташування елементів робочого місця, для того, щоб підтримувати оптимальну робочу позу працівника. В процесі роботи з комп'ютерною технікою потрібно слідувати правильному режиму праці та відпочинку, інакше спостерігається завелике навантаження зорового апарату у працівників, що у свою чергу веде до скарг на незадовільну роботу, мігрень, роздратування, порушення сну, втомленість, біль у спині, шиї тощо.

					РП 06. 13 000. 00 ДП ПЗ	Арк.
						34
Зм.	Арк	№ докум.	Підпис	Дата		

### 3.3 Розробка заходів з охорони праці

#### 3.3.1 Вимоги до виробничих приміщень

Розмір одного робочого місця має становити не менше 6 квадратних метрів. При необхідності, суміжні робочі місця співробітників, що працюють з комп'ютером, слід розділити перегородками висотою до 2 метрів.

Фарбування приміщень на роботі повинно сприяти створенню позитивних умов для зору працівників, для піднесеного настрою.

Рекомендуються наступні варіанти фарбування:

- жовтий о Активізує зір, має заспокійливий ефект для нервової системи. -  
зелений о Це колір заспокоєння. Він стимулює ритмічну роботу серця, прибирає  
запаморочення

У приміщеннях, де знаходиться комп'ютер, необхідно забезпечити наступні величини коефіцієнта відображення: для стелі: 60 - 70%, для стін: 40 - 50%, для підлоги: близько 30%. Для інших поверхонь та робочих меблів: 30 -40%.

#### 3.3.2 Освітлення

Правильно спроектоване та встановлене освітлення покращує умови для зорової роботи, сприяє підвищенню ефективності праці. Позитивно впливає на виробниче середовище, надаючи позитивний психологічний вплив на працюючого, підвищує безпеку праці та знижує травматизм.

У свою чергу, недостатній степінь освітлення призводить до навантаження на зір людини, послаблює увагу, призводить до швидкої втоми. В той самий час, надмірно високий рівень освітлення викликає ефект осліплення та викликає роздратування очей. Неправильне направлення світла може створювати різкі тіні, бліки, та викликати у працівника дезорієнтацію. Всі вищезазначені фактори можуть призвести до нещасного випадку або професійного захворювання.

Природне освітлення - освітлення приміщень денним світлом, що проникає через світлові прорізи у зовнішніх конструкціях приміщень.

Природне висвітлення характеризується тим, що змінюється в широких

					РП 06. 13 000. 00 ДП ПЗ	Арк.
						35
Зм.	Арк	№ докум.	Підпис	Дата		

межах залежно від часу дня, пори року, характеру області та низки інших факторів.

Штучне освітлення застосовується при роботі в темний час доби і вдень, коли не вдається забезпечити нормовані значення коефіцієнта природного освітлення (хмарна погода, короткий світловий день).

Освітлення, у якому недостатнє за нормами природне освітлення доповнюється штучним, називається суміщеним освітленням.

Як джерело світла при штучному освітленні повинні застосовуватися, як правило, люмінесцентні лампи типу ЛБ. При обладнанні відбивного освітлення у виробничих та адміністративно-громадських приміщеннях можуть застосовуватися металогалогенові лампи потужністю до 250 Вт. Допускається у світильниках місцевого освітлення застосовувати лампи розжарювання.

Рівень освітленості на робочому столі в зоні розташування документів має бути в межах 300 - 500 лк.

У разі неможливості забезпечити даний рівень освітленості системою загального освітлення допускається застосування світильників місцевого освітлення, але при цьому не повинно бути відблисків на поверхні екрану та збільшення освітленості екрану більше ніж до 300 лк. Світильники місцевого освітлення повинні мати напівпрозорий відбивач світла з захисним кутом не меншим за 40 град. Необхідно передбачити обмеження прямого блиску від джерела природного та штучного освітлення, при цьому яскравість поверхонь, що світяться (вікна, джерела штучного світла) і перебувають у полі зору, повинна бути не більшою за 200 кд/кв.м. Необхідно обмежувати відбитий блиск шляхом правильного вибору типів світильників та розміщення робочих місць відносно джерел природного та штучного освітлення.

### 3.3.3 Параметри мікроклімату

Принцип нормування мікроклімату — створення оптимальних умов для

					РП 06. 13 000. 00 ДП ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		36

теплообміну тіла людини з навколишнім середовищем.

Нормовані параметри мікроклімату для приміщень з ВДТ та ПЕОМ:

Пора року	Категорія робіт	Температура повітря. С	Відносна вологість повітря. %	Швидкість руху повітря, м/с
		оптимальна	оптимальна	оптимальна
Холодна	Легка 1-а	22-24	40-60	0.1
	Легка 1-б	21-23	40-60	0.1
Тепла	Легка 1-а	23-25	40-60	0.1
	Легка 1-б	22-24	40-60	0.2

Рівні іонізації повітря приміщень при роботі на ВДТ та ПЕОМ:

Рівні	Кількість іонів в 1 см. Куб повітря	
	Мінімально необхідні	400
Оптимальні	1500-3000	3000-5000
Максимально допустимі	50000	50000

Для забезпечення комфортних умов використовуються як організаційні методи (раціональна організація проведення робіт в залежності від пори року та доби, чергування праці та відпочинку), так і технічні засоби (вентиляція, кондиціонування повітря, опалювальна система).

### 3.3.4 Шум і вібрація

Під впливом шуму знижується концентрація уваги, порушуються фізіологічні функції, з'являється втома у зв'язку з підвищеними енергетичними витратами і нервово-психічною напругою, погіршується мовна комутація. Усе це знижує працездатність людини та її продуктивність, якість і безпеку праці. Для забезпечення нормованих рівнів шуму у виробничих приміщеннях та на робочих

					РП 06. 13 000. 00 ДП ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		37

місцях застосовуються шумопоглинальні засоби, вибір яких обґрунтовується спеціальними інженерно-акустичними розрахунками.

### **3.3.5 Електромагнітні та іонізуючі випромінювання**

Потужність експозиційної дози рентгенівського випромінювання на відстані 0,05 м від екрана та корпусу відеотерміналу при будь-яких положеннях регульовальних пристроїв відповідно до Норм радіаційної безпеки України (НРБУ97), затверджених постановою державного санітарного лікаря Міністерства охорони здоров'я України від 18.08.97 N 58, не повинна перевищувати 7,74 x 10 в ступ. - 12 А/кГ, що відповідає еквівалентній дозі 0,1 мбер/год (100 мкР/год). Вміст озону в повітрі робочої зони не повинен перевищувати 0,1 мг/куб.м; вміст оксиду азоту - 5 мг/куб.м; вміст пилу - 4 мг/куб.м

### **3.3.6 Ергономічні вимоги до робочого місця**

При організації робочого місця програміста необхідно дотримуватись наступних вимог. Оптимальне розміщення робочого обладнання та достатній робочий простір для комфортного переміщення та руху. Аспектами ергономіки є висота робочої поверхні, розміри простору для кінцівок, характеристики робочого крісла, відстань від очей до монітору, клавіатури тощо.

Головними елементами робочого місця саме для програміста є його крісло та стіл. Основним робочим положенням є сидяче. Вищезазначена робоча поза викликає найменшу степінь втоми для працівника. Раціональне планування робочого місця передбачає чіткий порядок та сталість розміщення предметів, засобів праці та документації. Те, що потрібно для виконання робіт частіше, розташоване в зоні легкої досяжності робочого простору.

Для комфортної роботи стіл повинен задовольняти такі умови:

- Висота столу повинна бути обрана з урахуванням можливості сидіти вільно, в зручній позі, при необхідності спираючись на підлокітники;
- Нижня частина столу має бути сконструйована так, щоб програміст міг зручно сидіти, не був змушений підтискати ноги;

					РП 06. 13 000. 00 ДП ПЗ	Арк.
						38
Зм.	Арк	№ докум.	Підпис	Дата		

- Поверхня столу повинна мати властивості, що виключають появу відблисків у полі зору програміста;

- Конструкція столу повинна передбачати наявність висувних ящиків (не менше 3 для зберігання документації, лістингів, канцелярського приладдя).

- Висота робочої поверхні рекомендується в межах 680-760мм. Висота поверхні, на яку встановлюється клавіатура, повинна бути близько 650мм.

### **3.4 Пожежна безпека**

Пожежна безпека входить в комплекс заходів з охорони праці, і організаційна робота в цій сфері на об'єктах господарювання включає широкий спектр заходів, а саме:

- 1) створення умов для безпечної праці,
- 2) мінімізації ризику виникнення пожеж,
- 3) своєчасне і повноцінне забезпечення технічними засобами для запобігання займання та усунення самих пожеж та їх наслідків,
- 4) контроль дотримання протипожежних вимог і норм законодавства
- 5) розробка і впровадження регламентів по гасінню пожеж, евакуації та порятунку з місць пожежі й задимлення людей і майна (матеріальних цінностей),
- 6) внутрішнє і зовнішнє навчання співробітників.

Пожежна безпека при роботі з комп'ютером передбачає обережність при обслуговуванні, ремонтних та профілактичних роботах та виконання всіх інструкцій працівником щодо пожежної безпеки. Приміщення, в яких розміщено ПК, слід оснащувати переносними газовими вогнегасниками ( вуглекислотними ) з розрахунку один вогнегасник ВВК-1,4 чи ВВК-2, але не менше ніж один вогнегасник зазначених типів на приміщення

					РП 06. 13 000. 00 ДП ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		39

## ВИСНОВКИ

Основні висновки можуть включати наступне:

### 1. Актуальність теми:

- Визначено, що розробка приложения ПДД має велике значення в контексті підвищення безпеки дорожнього руху та покращення свідомості водіїв про правила дорожнього руху.

### 2. Розроблений програмний продукт:

- Встановлено, що розроблений програмний продукт є ефективним інструментом для навчання та дотримання правил дорожнього руху.

- Він має інтуїтивно зрозумілий інтерфейс, широкий функціонал і забезпечує доступ до актуальної інформації щодо ПДД.

### 3. Вигоди для користувачів:

- Встановлено, що користувачі отримують ряд переваг від використання приложения ПДД, включаючи підвищення рівня свідомості про правила дорожнього руху, зменшення ризику дорожньо-транспортних пригод та поліпшення навичок водіння.

### 4. Економічна ефективність:

- Зроблено висновок, що розробка і впровадження приложения ПДД є економічно доцільними, з урахуванням позитивного впливу на безпеку дорожнього руху та зниження витрат на ліквідацію наслідків дорожніх пригод.

### 5. Рекомендації:

- Запропоновано ряд рекомендацій щодо подальшого вдосконалення програмного продукту, таких як розширення функціональності, вдосконалення інтерфейсу та посилення співпраці з правоохоронними органами.

					РП 06. 13 000. 00 ДП ПЗ	Арк.
						40
Зм.	Арк	№ докум.	Підпис	Дата		

У цілому, на основі проведеного дослідження можна зробити висновок, що розробка приложення ПДД має великий потенціал для покращення безпеки дорожнього руху та підвищення свідомості водіїв про правила дорожнього руху. Розроблений програмний продукт є ефективним інструментом, який забезпечує користувачам зручний доступ до необхідної інформації та допомагає зберегти життя і запобігти дорожнім пригодам.

					РП 06. 13 000. 00 ДП ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		41

## ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Тести ПДД 2023 [Електронний ресурс] – Режим доступу:  
<https://vodiy.ua/ru/process/>
2. ПДД 2023 Україна [Електронний ресурс] – Режим доступу: <https://xn--h1afceeb4a.xn--j1amh/>
3. Официальные Тесты ПДД Украины [Електронний ресурс] – Режим доступу: <https://pdr.infotech.gov.ua/>
4. Приклад діючих програм [Електронний ресурс]  
<https://sovety.pp.ua/index.php/ru/stati/android/drugoe/1848-uchim-pravila-dorozhnogo-dvizheniya-na-mobilnykh-ustrojstvakh-pk-i-onlajn-ukraina>
5. Інформація про Visual Studio [Електронний ресурс] – Режим доступу:  
<https://visualstudio.microsoft.com/>
6. Інформація про C++ [Електронний ресурс] – Режим доступу:  
<https://medium.com/nuances-of-programming/%D1%8F%D0%B7%D1%8B%D0%BA%D0%B8-%D0%B8-%D0%B3%D0%B4%D0%B5-%D0%B5%D1%89%D1%91-%D0%B8%D1%85-%D0%B8%D1%81%D0%BF%D0%BE%D0%BB%D1%8C%D0%B7%D1%83%D1%8E%D1%82-%D0%B8-%D0%B7%D0%B0%D1%87%D0%B5%D0%BC-9ff56559d5bf>

					РП 06. 13 000. 00 ДП ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		42

## ДОДАТОК А Фрагмент лістингу

### Програмний код форми

```
#include "MyForm.h"
using namespace System;
using namespace System::Windows::Forms;

[STAThreadAttribute]
int main(array<String>^ args) {
    Application::SetCompatibleTextRenderingDefault(false);
    Application::EnableVisualStyles();
    pdd::MyForm form;
    Application::Run(% form);
}
```

### Програмний код роботи програми

```
}
private: System::Void button28_Click(System::Object^ sender, System::EventArgs^
e) {
    this->pq4->Location = System::Drawing::Point(520, 20);
    this->pq3->Location = System::Drawing::Point(12, 12);
}
private: System::Void button26_Click(System::Object^ sender, System::EventArgs^
e) {
    this->pq3->Location = System::Drawing::Point(520, 20);
    this->pq2->Location = System::Drawing::Point(12, 12);
}
private: System::Void button25_Click(System::Object^ sender, System::EventArgs^
e) {
    this->pq2->Location = System::Drawing::Point(520, 20);
    this->pq1->Location = System::Drawing::Point(12, 12);
}
private: System::Void button21_Click(System::Object^ sender, System::EventArgs^
e) {
```

```
int points = 0;
if (radioButton51->Checked == true)
{
    points = points + 1;
}
if (radioButton3->Checked == true)
{
    points = points + 1;
}
if (radioButton7->Checked == true)
{
    points = points + 1;
}
if (radioButton12->Checked == true)
{
    points = points + 1;
}
if (radioButton46->Checked == true)
{
    points = points + 1;
}
if (radioButton56->Checked == true)
{
    points = points + 1;
}
if (radioButton45->Checked == true)
{
    points = points + 1;
}
if (radioButton37->Checked == true)
```

```

    {
        points = points + 1;
    }
    if (radioButton32->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton26->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton22->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton16->Checked == true)
    {
        points = points + 1;
    }
    p->Text = "Набрано балів: " + (points);
    this->pq10->Location = System::Drawing::Point(520, 20);
}

private: System::Void button1_Click_1(System::Object^ sender,
System::EventArgs^ e) {
    int points = 0;
    if (radioButton51->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton3->Checked == true)

```

```
{
    points = points + 1;
}
if (radioButton7->Checked == true)
{
    points = points + 1;
}
if (radioButton12->Checked == true)
{
    points = points + 1;
}
if (radioButton46->Checked == true)
{
    points = points + 1;
}
if (radioButton56->Checked == true)
{
    points = points + 1;
}
if (radioButton45->Checked == true)
{
    points = points + 1;
}
if (radioButton37->Checked == true)
{
    points = points + 1;
}
if (radioButton32->Checked == true)
{
    points = points + 1;
}
```

```

    }
    if (radioButton26->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton22->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton16->Checked == true)
    {
        points = points + 1;
    }
    p->Text = "Набрано балів: " + (points);
    panell->Location = System::Drawing::Point(87, 120);
    this->pq1->Location = System::Drawing::Point(520, 20);
}
private: System::Void button4_Click(System::Object^ sender, System::EventArgs^
e) {
    int points = 0;
    if (radioButton51->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton3->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton7->Checked == true)
    {

```

```
        points = points + 1;
    }
    if (radioButton12->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton46->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton56->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton45->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton37->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton32->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton26->Checked == true)
    {
        points = points + 1;
    }
```

```

if (radioButton22->Checked == true)
{
    points = points + 1;
}
if (radioButton16->Checked == true)
{
    points = points + 1;
}
p->Text = "Набрано балів: " + (points);
panel1->Location = System::Drawing::Point(87, 120);
this->pq2->Location = System::Drawing::Point(520, 20);
}
private: System::Void button6_Click(System::Object^ sender, System::EventArgs^
e) {
    int points = 0;
    if (radioButton51->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton3->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton7->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton12->Checked == true)
    {
        points = points + 1;
    }
}

```

```
}  
if (radioButton46->Checked == true)  
{  
    points = points + 1;  
}  
if (radioButton56->Checked == true)  
{  
    points = points + 1;  
}  
if (radioButton45->Checked == true)  
{  
    points = points + 1;  
}  
if (radioButton37->Checked == true)  
{  
    points = points + 1;  
}  
if (radioButton32->Checked == true)  
{  
    points = points + 1;  
}  
if (radioButton26->Checked == true)  
{  
    points = points + 1;  
}  
if (radioButton22->Checked == true)  
{  
    points = points + 1;  
}  
if (radioButton16->Checked == true)
```

```

    {
        points = points + 1;
    }
    p->Text = "Набрано балів: " + (points);
    panel1->Location = System::Drawing::Point(87, 120);
    this->pq3->Location = System::Drawing::Point(520, 20);
}
private: System::Void button13_Click(System::Object^ sender, System::EventArgs^
e) {
    int points = 0;
    if (radioButton51->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton3->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton7->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton12->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton46->Checked == true)
    {
        points = points + 1;
    }
}

```

```
if (radioButton56->Checked == true)
{
    points = points + 1;
}
if (radioButton45->Checked == true)
{
    points = points + 1;
}
if (radioButton37->Checked == true)
{
    points = points + 1;
}
if (radioButton32->Checked == true)
{
    points = points + 1;
}
if (radioButton26->Checked == true)
{
    points = points + 1;
}
if (radioButton22->Checked == true)
{
    points = points + 1;
}
if (radioButton16->Checked == true)
{
    points = points + 1;
}
p->Text = "Набрано балів: " + (points);
panell->Location = System::Drawing::Point(87, 120);
```

```

        this->pq9->Location = System::Drawing::Point(520, 20);
    }
private: System::Void next12_Click(System::Object^ sender, System::EventArgs^
e) {
    int points = 0;
    if (radioButton51->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton3->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton7->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton12->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton46->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton56->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton45->Checked == true)

```

```

    {
        points = points + 1;
    }
    if (radioButton37->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton32->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton26->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton22->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton16->Checked == true)
    {
        points = points + 1;
    }
    p->Text = "Набрано балів: " + (points);
    panel1->Location = System::Drawing::Point(87, 120);
    this->pq12->Location = System::Drawing::Point(520, 20);
}
private: System::Void button8_Click(System::Object^ sender, System::EventArgs^
e) {
    int points = 0;

```

```
if (radioButton51->Checked == true)
{
    points = points + 1;
}
if (radioButton3->Checked == true)
{
    points = points + 1;
}
if (radioButton7->Checked == true)
{
    points = points + 1;
}
if (radioButton12->Checked == true)
{
    points = points + 1;
}
if (radioButton46->Checked == true)
{
    points = points + 1;
}
if (radioButton56->Checked == true)
{
    points = points + 1;
}
if (radioButton45->Checked == true)
{
    points = points + 1;
}
if (radioButton37->Checked == true)
{
```

```

        points = points + 1;
    }
    if (radioButton32->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton26->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton22->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton16->Checked == true)
    {
        points = points + 1;
    }
    p->Text = "Набрано балів: " + (points);
    panell->Location = System::Drawing::Point(87, 120);
    this->pq4->Location = System::Drawing::Point(520, 20);
}
private: System::Void button9_Click(System::Object^ sender, System::EventArgs^
e) {
    int points = 0;
    if (radioButton51->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton3->Checked == true)

```

```
{
    points = points + 1;
}
if (radioButton7->Checked == true)
{
    points = points + 1;
}
if (radioButton12->Checked == true)
{
    points = points + 1;
}
if (radioButton46->Checked == true)
{
    points = points + 1;
}
if (radioButton56->Checked == true)
{
    points = points + 1;
}
if (radioButton45->Checked == true)
{
    points = points + 1;
}
if (radioButton37->Checked == true)
{
    points = points + 1;
}
if (radioButton32->Checked == true)
{
    points = points + 1;
}
```

```

    }
    if (radioButton26->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton22->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton16->Checked == true)
    {
        points = points + 1;
    }
    p->Text = "Набрано балів: " + (points);
    panell->Location = System::Drawing::Point(87, 120);
    this->pq11->Location = System::Drawing::Point(520, 20);
}
private: System::Void button11_Click(System::Object^ sender, System::EventArgs^
e) {
    int points = 0;
    if (radioButton51->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton3->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton7->Checked == true)
    {

```

```
        points = points + 1;
    }
    if (radioButton12->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton46->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton56->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton45->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton37->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton32->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton26->Checked == true)
    {
        points = points + 1;
    }
}
```

```

if (radioButton22->Checked == true)
{
    points = points + 1;
}
if (radioButton16->Checked == true)
{
    points = points + 1;
}
p->Text = "Набрано балів: " + (points);
panel1->Location = System::Drawing::Point(87, 120);
this->pq6->Location = System::Drawing::Point(520, 20);
}
private: System::Void button15_Click(System::Object^ sender, System::EventArgs^
e) {
    int points = 0;
    if (radioButton51->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton3->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton7->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton12->Checked == true)
    {
        points = points + 1;
    }
}

```

```
}  
if (radioButton46->Checked == true)  
{  
    points = points + 1;  
}  
if (radioButton56->Checked == true)  
{  
    points = points + 1;  
}  
if (radioButton45->Checked == true)  
{  
    points = points + 1;  
}  
if (radioButton37->Checked == true)  
{  
    points = points + 1;  
}  
if (radioButton32->Checked == true)  
{  
    points = points + 1;  
}  
if (radioButton26->Checked == true)  
{  
    points = points + 1;  
}  
if (radioButton22->Checked == true)  
{  
    points = points + 1;  
}  
if (radioButton16->Checked == true)
```

```

    {
        points = points + 1;
    }
    p->Text = "Набрано балів: " + (points);
    panel1->Location = System::Drawing::Point(87, 120);
    this->pq8->Location = System::Drawing::Point(520, 20);
}
private: System::Void end11_Click(System::Object^ sender, System::EventArgs^ e)
{
    int points = 0;
    if (radioButton51->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton3->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton7->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton12->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton46->Checked == true)
    {
        points = points + 1;
    }
}

```

```
if (radioButton56->Checked == true)
{
    points = points + 1;
}
if (radioButton45->Checked == true)
{
    points = points + 1;
}
if (radioButton37->Checked == true)
{
    points = points + 1;
}
if (radioButton32->Checked == true)
{
    points = points + 1;
}
if (radioButton26->Checked == true)
{
    points = points + 1;
}
if (radioButton22->Checked == true)
{
    points = points + 1;
}
if (radioButton16->Checked == true)
{
    points = points + 1;
}
p->Text = "Набрано балів: " + (points);
panell->Location = System::Drawing::Point(87, 120);
```

```

    this->pq11->Location = System::Drawing::Point(520, 20);
}
private: System::Void button19_Click(System::Object^ sender, System::EventArgs^
e) {
    int points = 0;
    if (radioButton51->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton3->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton7->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton12->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton46->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton56->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton45->Checked == true)

```

```

    {
        points = points + 1;
    }
    if (radioButton37->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton32->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton26->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton22->Checked == true)
    {
        points = points + 1;
    }
    if (radioButton16->Checked == true)
    {
        points = points + 1;
    }
    p->Text = "Набрано балів: " + (points);
    panell->Location = System::Drawing::Point(87, 120);
    this->pq7->Location = System::Drawing::Point(520, 20);
}
private: System::Void home_Click(System::Object^ sender, System::EventArgs^ e)
{
    this->label13->Location = System::Drawing::Point(143, 83);
}

```

**\* Розробка  
програмного  
забезпечення для  
вивчення ПДР**

**Виконав:**

**Кузнєцов Олександр Олегович**

**Керівник:**

**К.Т.Н., доц. Селіванова Алла Віталіївна**

1

По-перше, безпека на дорозі є однією з найважливіших проблем сучасного світу. Кількість дорожньо-транспортних пригод і травм, пов'язаних з порушеннями правил дорожнього руху, залишається досить високою. Отже, розробка програмного продукту, що допомагатиме ефективно вивчати та запам'ятовувати правила дорожнього руху, може сприяти підвищенню рівня безпеки на дорогах.

**\* Актуальність**

2

\* Метою даної роботи є розробка програмного забезпечення для вивчення ПДР

\* Постановка задачі:

1. Огляд предметної галузі: Вивчення основних принципів та правил дорожнього руху, аналіз законодавства, включаючи Правила дорожнього руху та інші відповідні документи.
2. Аналіз технологій: Дослідження різних технологій та інструментів, що використовуються для розробки програмного забезпечення, зокрема для створення програм з навчання правилам дорожнього руху. Розгляд можливих підходів, мов програмування, середовищ розробки та інших компонентів.
3. Аналіз аналогів: Вивчення існуючих програм або додатків, що займаються навчанням правил дорожнього руху. Аналіз їх функціональності, інтерфейсу, переваг та недоліків, а також реакції користувачів.
4. Визначення вимог до системи: Формулювання основних вимог до розроблюваної програми, включаючи функціональні, нефункціональні, інтерфейсні та інші вимоги. Врахування потреб та очікувань користувачів.
5. Моделювання системи: Створення моделі програми з вивчення правил дорожнього руху, що включає в себе структуру, компоненти, взаємодію та потоки даних. Розробка діаграм або інших інструментів для візуалізації системи.
6. Проектування інтерфейсу: Розробка зручного та інтуїтивно зрозумілого інтерфейсу для користувачів, який дозволить ефективно вивчати правила дорожнього руху, проходити тестування та отримувати зворотний зв'язок.
7. Опис реалізації програмного продукту: Розробка програмного коду, реалізація функціональності, взаємодії з базою даних та інших компонентів. Тестування, виправлення помилок та оптимізація продукту.
8. Валідація та оцінка продукту: Перевірка правильності реалізації, оцінка ефективності та зручності використання програми. Збір зворотного зв'язку від користувачів та оцінка їх задоволення результатами.

## \* Мета, задачі

3

3

	Тести ПДД 2023	ПДР 2023 Україна	Официальные Тесты ПДД Украины
Зручність інтерфейсу	+	+	+
Ознайомлення з теорією	+	+	+
Авторизованість	+	-	+
Дизайн	Зрозумілий, легкий, нативний	Зрозумілий, легкий, нативний, застарілий	Зрозумілий, легкий, нативний
Оцінка користувача	+	-	+
Постійне оновлення даних	+	-	+

## \* Аналіз аналогів

4

4

- Мова програмування: C++
- Інтегроване середовище розробки (IDE): Visual Studio
- Графічна бібліотека: Windows Forms
- Взаємодія з користувачем: елементи управління Windows Forms (такі як кнопки, тексти, календар і т.д.)
- Застосування: програма для вивчення правил дорожнього руху

## \* Засоби реалізації

5

- Початок тесту
- Тест
- Оцінка
- Вихід

## \* Схема інтерфейсу

6

## \* Елементи коду

```
if (radioButton16->Checked == true)
{
    points = points + 1;
}
p->Text = "Найпача баня: " + (points);
panel1->Location = System.Drawing::Point(57, 120);
this->pq1->Location = System.Drawing::Point(20, 20);
private: System::Void button19_Click(System::Object* sender, System::EventArgs* e) {
int points = 0;
if (radioButton51->Checked == true)
{
    points = points + 1;
}
if (radioButton3->Checked == true)
{
    points = points + 1;
}
if (radioButton7->Checked == true)
{
    points = points + 1;
}
if (radioButton12->Checked == true)
{
    points = points + 1;
}
if (radioButton6->Checked == true)
{
    points = points + 1;
}
if (radioButton6->Checked == true)
{
    points = points + 1;
}
if (radioButton45->Checked == true)
{
    points = points + 1;
}
if (radioButton37->Checked == true)
{
    points = points + 1;
}
if (radioButton32->Checked == true)
{
    points = points + 1;
}
```

## \* Особливості розробки

7

7

```
pragmas endregion
private: System::Void MyForm_Load(System::Object* sender, System::EventArgs* e) {
private: System::Void button1_Click(System::Object* sender, System::EventArgs* e) {
this->Close();
private: System::Void pictureBox1_Click(System::Object* sender, System::EventArgs* e) {
private: System::Void go_Click(System::Object* sender, System::EventArgs* e) {
this->label13->Location = System::Drawing::Point(658, 120);
this->panel5->Location = System::Drawing::Point(658, 120);
this->pq1->Location = System::Drawing::Point(12, 12);
private: System::Void next1_Click(System::Object* sender, System::EventArgs* e) {
this->pq1->Location = System::Drawing::Point(520, 20);
this->pq2->Location = System::Drawing::Point(12, 12);
private: System::Void button24_Click(System::Object* sender, System::EventArgs* e) {
private: System::Void next2_Click(System::Object* sender, System::EventArgs* e) {
this->pq2->Location = System::Drawing::Point(520, 20);
this->pq3->Location = System::Drawing::Point(12, 12);
private: System::Void next3_Click(System::Object* sender, System::EventArgs* e) {
this->pq3->Location = System::Drawing::Point(520, 20);
this->pq4->Location = System::Drawing::Point(12, 12);
private: System::Void next4_Click(System::Object* sender, System::EventArgs* e) {
this->pq4->Location = System::Drawing::Point(520, 20);
this->pq5->Location = System::Drawing::Point(12, 12);
private: System::Void next5_Click(System::Object* sender, System::EventArgs* e) {
this->pq5->Location = System::Drawing::Point(520, 20);
this->pq6->Location = System::Drawing::Point(12, 12);
}
```

## \* Особливості розробки

8

8

## \* Робота із програмою розпочинається з ГОЛОВНОГО МЕНЮ:



9

9



10

## \* Інтерфейс

10



## \*Інтерфейс

11

11



## \*Інтерфейс

12

12



## \* Інтерфейс

13

13

Основні висновки можуть включати наступне:

1. Актуальність теми:

- Визначено, що розробка приложения ПДД має велике значення в контексті підвищення безпеки дорожнього руху та покращення свідомості водіїв про правила дорожнього руху.

2. Розроблений програмний продукт:

- Встановлено, що розроблений програмний продукт є ефективним інструментом для навчання та дотримання правил дорожнього руху.

- Він має інтуїтивно зрозумілий інтерфейс, широкий функціонал і забезпечує доступ до актуальної інформації щодо ПДД.

3. Вигоди для користувачів:

- Встановлено, що користувачі отримують ряд переваг від використання приложения ПДД, включаючи підвищення рівня свідомості про правила дорожнього руху, зменшення ризику дорожньо-транспортних пригод та поліпшення навичок водіння.

4. Економічна ефективність:

- Зроблено висновок, що розробка і впровадження приложения ПДД є економічно доцільними, з урахуванням позитивного впливу на безпеку дорожнього руху та зниження витрат на ліквідацію наслідків дорожніх пригод.

5. Рекомендації:

- Запропоновано ряд рекомендацій щодо подальшого вдосконалення програмного продукту, таких як розширення функціональності, вдосконалення інтерфейсу та посилення співпраці з правоохоронними органами.

У цілому, на основі проведеного дослідження можна зробити висновок, що розробка приложения ПДД має великий потенціал для покращення безпеки дорожнього руху та підвищення свідомості водіїв про правила дорожнього руху. Розроблений програмний продукт є ефективним інструментом, який забезпечує користувачам зручний доступ до необхідної інформації та допомагає зберегти життя і запобігти дорожнім пригодам.

## \* Висновки

14

14

## РЕЦЕНЗІЯ

на дипломний проект (роботу) здобувача (здобувачки) освіти  
відділення комп'ютерних систем

**Кузнєцова Олександра Олегівна**

(прізвище, ім'я та по батькові)

Спеціальність 121 Інженерія програмного забезпечення

Освітня програма Розробка програмного забезпечення

Керівник дипломного проекту (роботи) Селіванова Алла Віталіївна

(прізвище, ім'я та по батькові)

Тема дипломного проекту (роботи) **Розробка програмного модуля "Електронний гаманець співробітника" для ERP-системи документообігу складу**

Обсяг розрахунково-пояснювальної записки 60 сторінок

Обсяг графічної (презентаційної) частини 10 аркушів (слайдів)

### ХАРАКТЕРИСТИКА ДИПЛОМНОГО ПРОЕКТУ (РОБОТИ)

а) заключення про ступінь відповідності виконаного дипломного проекту (роботи) завданню

Робота відповідає технічному завданню до дипломного проекту. Виконана у відповідності з вимогами

б) характеристика виконання кожного розділу дипломного проекту (роботи)

При виконанні дипломного проекту здобувач продемонстрував уміння використовувати останні досягнення науки та техніки, уміння працювати з літературою. Так, здобувач грамотно дослідив та проаналізував програмні засоби створення програмного модуля на мові РНР для системи управління документообігом складу з дистанційним управлінням.

в) оцінка якості виконання пояснювальної записки та графічної частини дипломного проекту (роботи) Пояснювальна записка та графічна частина відповідає вимогам, виконана якісно та відображає основні елементи проектування модуля. Містить етапи створення програмного модуля «Електронний гаманець співробітника» для системи управління складом

г) перелік позитивних якостей дипломного проекту (роботи) \_\_\_\_\_

Тема дипломного проекту є актуальною, обрана мова програмування PHP –  
що є сучасною для створення динамічного вебсайту що працюють із  
підключеною до нього базою даних MySQL

д) основні недоліки дипломного проекту (роботи) \_\_\_\_\_

У тексті пояснювальної записки відсутні посилання на використану  
літературу. Для підвищення ефективності дослідження можна було б провести  
порівняння із застосуванням інших програмних засобів для створення  
динамічних сайтів

Оцінка розрахункової частини \_\_\_\_\_ 4(добре)

Оцінка графічної частини \_\_\_\_\_ 4 (добре)

Загальна оцінка \_\_\_\_\_ 4(добре)

Прізвище, ім'я, по батькові рецензента Кривченко Юрій Вікторович

Місце роботи і посада рецензента \_\_\_\_\_  
ВСП “Одеський технічний фаховий коледж ОНТУ”, голова циклової  
комісії комп'ютерних технологій та програмної інженерії

Підпис: \_\_\_\_\_

« 16 » \_\_\_\_\_ червня 2022 р.

**ВІДГУК**

керівника на дипломний проект здобувача (здобувачки) освіти  
відділення комп'ютерних систем

Кузнецова Олександра Олеговича

(прізвище, ім'я та по батькові)

Спеціальність: 121 Інженерія програмного забезпечення

Освітня програма: Розробка програмного забезпечення

Тема дипломного проекту: **Розробка програмного модуля "Електронний гаманець співробітника" для ERP-системи документообігу складу**

**ХАРАКТЕРИСТИКА ДИПЛОМНОГО ПРОЕКТУ**

а) обсяг і якість виконання проекту (графічного матеріалу і розрахунково-пояснювальної записки) \_\_\_\_\_

Пояснювальна записка виконана якісно, у достатньому обсязі відповідно до теми дипломного проекту та складає 60 аркушів ф.А4. Розділи пояснювальної записки відповідають етапам рішення завдання, поставленому у дипломному проекті.

Презентація виконана якісно у достатньому обсязі. Презентація наочно демонструє результати роботи.

б) самостійність роботи над проектом: \_\_\_\_\_

Здобувач Кузнецов О.О. самостійно обрав тему дипломного проекту та виявив навички самостійного опрацювання матеріалу над темою. Дипломний проект перевірено на плагіат системою Unichек. Схожість складає 39.7%

в) теоретична підготовка випускника (випускниці): \_\_\_\_\_

Кузнецова О.О. відповідає вимогам, що надаються до молодшого спеціаліста за спеціальністю 121 «Інженерія програмного забезпечення»

г) вміння розв'язувати виробничі та конструкторські питання \_\_\_\_\_  
В процесі роботи над дипломним проектом Кузнецов Олександр Олегович показав  
вміння розв'язувати виробничі та конструкторські питання, а саме вміння проектувати  
програмне забезпечення, реалізувати функціональні елементи системи із  
застосуванням відповідних інструментальних засобів.

Оцінка розрахункової частини \_\_\_\_\_ 4 (добре)

Оцінка графічної частини \_\_\_\_\_ 3(задовільно)

Загальна оцінка \_\_\_\_\_ 4 (добре)

Прізвище, ім'я, по батькові керівника дипломного проекту \_\_\_\_\_

Селіванова Алла Віталіївна

Місце роботи і посада керівника дипломного проекту \_\_\_\_\_

доцент кафедри інформаційних технологій та кібербезпеки ОНТУ

Підпис \_\_\_\_\_



« 08 » червня 2023 р.

**ДОЗВІЛ  
НА РОЗМІЩЕННЯ  
ВИПУСКНОЇ КВАЛІФІКАЦІЙНОЇ РОБОТИ  
В ЕЛЕКТРОННОМУ РЕПОЗИТАРІЇ ВСП «ОТФК ОНТУ»**

Ми, що нижче підписалися,

*Кузнєцов Олександр Олегович,*  
здобувач освіти гр. 4РП-06, та

*Селіванова Алла Віталіївна,*  
керівник дипломного проекту,

не заперечуємо щодо розміщення електронного варіанту пояснювальної записки до випускної кваліфікаційної роботи молодшого спеціаліста на тему:

**«Розробка програмного модуля "Електронний гаманець співробітника" для ERP-системи документообігу складу» (автор роботи – Кузнєцов О.О., керівник роботи – Селіванова А.В.)**

виконаного у ВСП «Одеський технічний фаховий коледж Одеського національного технологічного університету» в 2023 році, у повному обсязі в електронному репозитарії ВСП «ОТФК ОНТУ» для вільного доступу через мережу Інтернет.

Несемо відповідальність за ідентичність електронного та друкованого варіантів випускної кваліфікаційної роботи, і даємо згоду на обробку персональних даних.

Виконавець  / Кузнєцов О.О./

Керівник  / Селіванова А.В./

« 08 » \_\_\_\_\_ червня \_\_\_\_\_ 2023 \_\_\_\_\_ р.

Ім'я користувача:  
Наталія Вікторівна Копусь

ID перевірки:  
1015594918

Дата перевірки:  
14.06.2023 09:35:27 EEST

Тип перевірки:  
Doc vs Internet + Library

Дата звіту:  
14.06.2023 09:36:08 EEST

ID користувача:  
100011688

Назва документа: 4РП -06\_Олександр\_Кузнецов

Кількість сторінок: 40 Кількість слів: 5276 Кількість символів: 39506 Розмір файлу: 964.66 KB ID файлу: 1015243960

## 39.7% Схожість

Найбільша схожість: 19.1% з Інтернет-джерелом (<https://subj.ukr-lit.com/informacijni-sistemi-i-texnologij%d1%97-na-pid>).

39.7% Джерела з Інтернету

625

Сторінка 42

Не знайдено джерел з Бібліотеки

## 0% Цитат

Вилучення цитат вимкнене

Вилучення списку бібліографічних посилань вимкнене

## 0% Вилучень

Немає вилучених джерел

## Модифікації

Виявлено модифікації тексту. Детальна інформація доступна в онлайн-звіті.

Замінені символи

4