

МІНІСТЕРСТВО ОСВІТИ І НАУКИ, МОЛОДІ ТА СПОРТУ УКРАЇНИ
ДВНЗ «ДОНЕЦЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ»

Н.О.Маслова, Р.О.Сорокін

**Методичні вказівки до виконання курсового
проектування
з дисципліни
«Основи візуального програмування»
для студентів очної та заочної форм навчання
за напрямом «Комп'ютерні науки»**

Донецьк,
ДонНТУ,
2013

МІНІСТЕРСТВО ОСВІТИ І НАУКИ, МОЛОДІ ТА СПОРТУ УКРАЇНИ
ДВНЗ «ДОНЕЦЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ»

Н.О.Маслова, Р.О.Сорокін

**Методичні вказівки до виконання курсового
проектування
з дисципліни
«Основи візуального програмування»
для студентів очної та заочної форм навчання
за напрямом «Комп'ютерні науки»**

Розглянуто на засіданні кафедри
«Програмне забезпечення
інтелектуальних систем»,
протокол № 6 від 6.12.2012

Затверджено на засіданні
навчально-видавничої ради
Донецького національного
технічного університету
протокол № 1 від 21.02.2013р

Донецьк,
ДонНТУ,
2013

УДК 004.4'236

Маслова Н.О., Сорокін Р.О.

Методичні вказівки до виконання курсового проектування з дисципліни «Основи візуального програмування» / [укл. Н.О.Маслова, Р.О.Сорокін]. - Донецьк : ДонНТУ, 2013. - 33 с.

Методичний посібник призначений для студентів, що навчаються за напрямом підготовки - 6.050101 «Комп'ютерні науки».

Методичні вказівки містять інформацію про організацію, порядок захисту та методику виконання курсового проектування з дисципліни «Основи візуального програмування». В них включено відомості про структуру, зміст, порядок оформлення пояснювальної записки, наведено тематику курсових проектів, приклади оформлення типових частин пояснювальної записки, вимоги до створення й опису програмного продукту.

Укладачі: Н.О.Маслова, Р.О.Сорокін

Рецензенти:

-доктор технічних наук, професор, завідуючий кафедрою «Обчислювальна математика й програмування» Павлиш В.М.,

-кандидат технічних наук, доцент, заступник завідуючого кафедрою «Системи штучного інтелекту» Вороной С.М.

ВСТУП

Візуальне програмування в даний час є однією з найбільш популярних парадигм програмування. При використанні візуального середовища у програміста з'являється можливість проектувати частину додатку, наприклад, інтерфейси майбутнього продукту, із застосуванням візуальних засобів додавання і налаштування спеціальних бібліотечних компонентів. Результатом візуального проектування є макет майбутньої програми, в яку вже внесено відповідні коди. Остаточний вид додаток набуває після написання окремих блоків кодів, які призначені для відпрацювання реакції на нестандартні властивості, події та методи продукту.

Курсове проектування є невід'ємним етапом оволодіння основами візуального програмування, воно призначено для відточення навичок та знань, придбаних при вивченні основного курсу, таких як побудова MDI-додатків, створення компонент програміста, робота у багатовіконному режимі, проектування діалогів, формування довідкової системи, робота з графічними елементами та базами даних.

Навички, які будуть придбані в процесі виконання курсового проекту можуть використовуватися при розробці програм і програмних комплексів із застосуванням сучасних методик візуального програмування. У методичних вказівках на прикладі розробки тестово-довідкової системи «Teacher» показано виконання основних етапів проекту, приділено увагу поясненням основних особливостей.

Методичні вказівки розроблені на основі кваліфікаційних вимог, що пред'являються до фахівців в області програмної та комп'ютерної інженерії, комп'ютерних наук, системної інженерії та інформатики й є однією з складових частин навчального процесу.

1. ЦІЛІ Й ЗАДАЧІ КУРСОВОГО ПРОЕКТУВАННЯ

Цілі курсового проектування:

- систематизація, розширення і закріплення теоретичних і практичних знань з дисципліни «Основи візуального програмування»;
- закріплення й розвинування навичок самостійного вирішення завдань в області візуального проектування і програмування;
- отримання підтвердження можливості ефективної реалізації програмних продуктів із застосуванням візуального підходу;
- розвинування творчих здібностей в галузі розробки програмних продуктів.

Основна мета курсового проектування - підготовка студентів до самостійної практичної інженерної діяльності в області створення програмних продуктів із застосуванням сучасних технологій на основі знань, отриманих в процесі навчання у вищому навчальному закладі.

Основна задача курсового проектування – вдосконалення навичок самостійної роботи в напрямку проектування та розробки сучасного програмного забезпечення.

Загальна назва проекту – «Використання технологій візуальної розробки додатків при побудові тестово-довідкової системи», а далі – обрана тематика з додатку А. Студент може пропонувати й свою тему, узгодивши її з керівником проекту.

Курсовий проект є самостійною роботою студента. За всі прийняті в курсовому проекті технічні рішення, повноту, цілісність і достовірність даних несе відповідальність студент - автор курсового проекту

Результатом завершального етапу курсового проектування повинна бути пояснювальна записка до курсового проекту і розроблений програмний продукт.

2. ЕТАПИ КУРСОВОГО ПРОЕКТУВАННЯ

В рамках курсового проекту студентом розробляється тестово-довідкова система. Процес розробки ґрунтується на застосуванні технологій візуального програмування та включає в себе етапи:

1. Вибір предметної області (варіанти дивись у додатку А), формулювання теми курсового проекту, складання та затвердження Листа Завдання (Додаток Б), формулювання постановки задачі: вихідні дані, обмеження, результати, зв'язок між вихідними даними і результатами.
2. Складання й затвердження технічного завдання (Додаток В).
3. Визначення загальної структури та архітектури системи.

В рамках цього кроку має бути визначена спільна архітектура додатка, проведено виділення модулів, організовано їх взаємозв'язок. При відображенні структури системи та взаємодії модулів рекомендовано застосування мови графічного опису UML.

4. Розробка алгоритмів функціонування програмного продукту.

Виконується опис усіх алгоритмічних структур – головної програми, модулів, допоміжних функцій й т.д. Для опису алгоритмів роботи системи рекомендовано застосування уніфікованої мови моделювання UML.

5. Розробка загального інтерфейсу.

Розробляється інтерфейс взаємодії з користувачем та адміністратором системи, загальний вид усієї системи. Інтерфейс може складатися з простої послідовності форм, бути MDI або багато-віконним додатком. Розробляється загальний вид меню, довідкової системи, основний вид та стиль запитів та повідомлень. Вирішується необхідність строки стану, статусної строки, вид та сенс підказок, міста їх виникнення. Визначаються методи аутентифікації і авторизації користувачів (при необхідності), регламентується доступ до окремих форм та режимів.

6. Вибір та оформлення довідково-інформаційних даних по обраному напрямку (згідно обраної теми курсового проекту). Формування презентацій, текстових файлів за різними розділами, літературних та Інтернет – посилань, пошук відео-уроків. Обрання способу відображення цього матеріалу в тестово-довідковій системі, що розробляється в проекті.

7. Проектування і наповнення бази даних.

В результаті цього кроку має бути повністю визначена схема БД додатка — набір таблиць, типи полів, зв'язки між таблицями, первинні ключі і альтернативні ключі таблиць. Визначається, яким чином буде організовано доступ до елементів бази даних (BDE, ADO, dbExpress, або інший) . База даних може бути створена як в рамках використовуваної СУБД, так й програмним шляхом.

На цьому кроці треба підготувати первинні дані для наповнення бази й зробити це. Подалі база може бути доповнена тестовими даними, необхідними для перевірки працездатності як самої бази, так і додатка в цілому. Необхідний для цього режим треба передбачити на етапі програмування.

8. Проектування та розробка форм.

При проектуванні форм враховуються аспекти зручності використання. В результаті цього кроку мають бути отримані точні описи або проекти всіх форм додатка у візуальному редакторі. Треба вказати тип форм (головна, дочірня чи модальна, скрита чи екранна), зробити перелік компонент, що розміщуються на формах, описати їх основні властивості (у тому числі й розмір та розташування), вказати зв'язок між формами, переходи та логічні послідовності відображення форм на екрані.

9. Написання та налагодження програми.

На цьому кроці розробляється програмний код. Програмуються події та методи, розробляються й розміщуються на палітрі компоненти програміста, виділяються DLL. Також налагоджується обраний зв'язок моделі даних та програм обробки. Відпрацьовуються програмні коди властивостей, методів та подій. При необхідності створюються компоненти програміста та елементи бібліотек DLL. В результаті виконання цього кроку має бути

написаний програмний код моделі даних додатка на мові C++ та виконано його налагодження в середовищі C++ Builder.

10. Розробка допоміжних режимів.

У цій частині виконується проектування та розробка таких режимів, як «Допомога», «Довідка про систему», «Інформація про розробника», «Інструкція користувачеві».

11. Тестування системи.

На цьому етапі визначається послідовність дій, які треба виконати для перевірки працездатності кожного режиму та форми. В результаті цього кроку для кожної з форм додатка має бути складений список сценаріїв роботи з даною формою та її найближчими сусідами, в ході яких перевіряються всі функції і можливості, надані цією формою і спільна робота всіх форм.

12. Вибір й створення варіанту постачання системи користувачеві, створення завантажувального диску.

Курсовий проект рекомендується виконувати у відповідності з графіком, наведеним у таблиці 2.1.

Таблиця 2.1. - Графік виконання курсового проекту

№	Етапи роботи	Термін виконання (№ тижня)
1	Вибір предметної області, складання та затвердження Листа Завдання, виконання постановки задачі	1–2
2	Складання та затвердження технічного завдання. Визначення загальної структури та архітектури системи	2 – 3
3	Розробка алгоритмів функціонування програмного продукту. Розробка загального інтерфейсу	3–4
4	Вибір та оформлення Довідково-інформаційних даних по обраному напрямку	5
5	Проектування і наповнення бази даних	6
6	Проектування та розробка форм	7
7	Написання та налагодження програми	8–9
8	Розробка допоміжних режимів. Тестування системи. Вибір й створення варіанту постачання	10
9	Створення пояснювальної записки	11 – 14
10	Захист курсового проекту	14 – 16

Зверніть увагу, що 5 та 10 тижні є контрольними термінами проекту, коли студент зобов'язаний показати частину готової роботи. Так, на 5-му тижні розглядається та оцінюються робота, виконана згідно пунктів 1-4 графіку, а на 10-му тижні семестру – перевіряється виконання усунення зауважень по попереднім пунктам та виконання п.5-8 графіку включно.

3. ЗАВДАННЯ НА КУРСОВЕ ПРОЕКТУВАННЯ

В рамках курсового проекту за допомогою візуальних засобів розробляється тестово-довідкова система. У процесі розробки тестово-довідкової системи повинно бути реалізовано наступне.

1. Тестово-довідкова система повинна бути розроблена з використанням візуальних технологій програмування та в діалоговому режимі проводити тестування знань користувача з обраної тематики, оцінювати ці знання і давати рекомендації по подальшій підготовці. Система повинна мати дружній користувачеві інтерфейс, захищати його від невірних дій (обробка виключень), встановлюватися та працювати на різних комп'ютерах (розробника, користувача, викладача).

2. Система повинна передбачити можливість виконання режимів:

- «Ознайомчий»;
- «Тестування»;
- «Оцінювання»;
- «Рекомендації»;
- «Адміністрування»;
- «Довідка про систему».

Режим «Ознайомчий» може включати, наприклад, правила тестування, інформацію про предмет, можливість вибору питань по всьому курсу або за окремим розділом теми (рівень тестування), у цьому розділі можна провести пробне опитування.

Режим «Тестування» пропонує питання тесту, обрані у випадковому порядку, обмежує час реакції користувача, приймає відповідь і реєструє його в системі. При цьому до кожного питання запропонувати 4-5 варіантів відповідей з яких хоча б одна повинна бути вірною.

Режим «Оцінювання» на підставі відповідей користувача і шаблонів, закладених в базі даних оцінює знання випробуваного (критерій пропонує розробник системи), виявляє розділ, за яким дано найбільшу кількість невірних відповідей.

Режим «Рекомендації» повинен містити поради щодо подальшої підготовки користувача системи і за його вибором відображати інформацію для подальшого вивчення у вигляді посилань на текстовий документ (*. Doc, *. Pdf), презентацію, підбір літературних джерел, Інтернет-сайтів або відео-уроків.

Режим «Адміністрування» повинен мати захищений вхід (пароль) і виконувати операції з базою даних (доповнення, зміна, видалення, завдання запитів і вибірку інформації з бази даних), управління часом тестування і володіти іншими функціями, необхідність в яких буде виявлена в процесі розробки.

Режим «Довідка про систему» повинен містити відомості про розробку і її автора, правила роботи з тестами (інструкцію користувачу), опис критеріїв оцінювання і т.п. Бажано передбачити і закритий паролем режим

довідки для адміністратора, в якому вказати, наприклад, перелік та особливості формування і підключення файлів довідок, інструкцію про настрійку часу тестування системи, та ін.

Режими можна виділити окремими блоками, або об'єднувати в спільні – в залежності від структури системи і алгоритмічних рішень розробника проекту.

4. Теоретичний матеріал для тестування готується студентом самостійно і його обсяг повинен бути таким, щоб тестування можна було провести по не менше ніж трьом підтемам (розділам), для кожної з яких розробити не менш 10 тестових запитань та 4-5 відповідей до них. Питання повинні бути змістовними і відповідати тематиці, обраної для розробки проекту.

5. Довідковий матеріал для розділу «Рекомендації» також готується студентом самостійно і повинен охоплювати тему в цілому. При цьому будь-який з довідкових матеріалів повинен включати відповіді та роз'яснення на поставлені питання, явно не задаючи їх.

6. Програмний продукт поставляється з пояснювальною запискою до курсового проекту у файлі FIO.doc, керівництвом користувача у файлі Help_fio.doc, вихідними кодами у вигляді проекту середовища розробки в папці FIO, виконавчим файлом програми fio.exe і файлом із даними про розробника readme_fio.txt. Варіант постачання та спосіб виготовлення завантажувального диску обирається розробником.

4. РЕКОМЕНДАЦІЇ ДО РОЗРОБКИ

У рамках курсової роботи розробляється тестово-довідкова система. Розробка програмного продукту повинна включати наступні моменти:

- визначення переліку типових завдань в межах заданої тематики;
- застосування графічних елементів супроводу вирішення завдань або теоретичних розділів;
- обрання способу схематичного зображення алгоритмів і процесів, логіки роботи програми (текстовий опис, блок-схеми, R-схеми, мова функціональних схем, універсальна мова моделювання UML чи інше (що саме));
- розробку форматів файлів для зберігання вхідної та вихідної інформації;
- розробку інтерфейсу системи згідно з функціональними вимогами;
- розробку меню, що відображає усі основні режими роботи системи;
- відпрацювання текстів діалогів і повідомлень, необхідних для роботи системи;
- визначення способів і місць відображення довідок та підказок;

Програмний продукт по можливості повинен мати модульну структуру, являти собою автономну систему і мати:

- інтуїтивно-зрозумілий інтерфейс;
- спокійну кольорову гаму,

- загальне й допоміжні меню (з підрозділами);
- контекстні підказки;
- виведення повідомлень про помилки (обробку виключень);
- довідкову інформацію з описом принципів роботи й можливостей системи, координатами розробника й т.д.;
- єдину мову інтерфейсу.

При написанні програмного продукту слід дотримуватися наступного:

- програма повинна бути структурована (основні блоки виділені в підпрограми і модулі), в тому числі - зовні (візуальна структурованість операторів і блоків програми);
- всі основні блоки повинні мати пояснювальні коментарі;
- при застосуванні підпрограм з параметрами, вони теж повинні включати коментарі, які відображають призначення як підпрограми, так і її параметрів;
- імена констант, типів, змінних, процедур і функцій, записувати таким чином, щоб по імені об'єкта було зрозуміло його призначення;
- призначення констант, змінних та масивів треба описати.

Перелік вимог може уточнюватися і доповнюватися керівником роботи.

Приклади коментарів:

```
//Директиви препроцесора
#include <vcl.h>
#pragma hdrstop

// макроси, що підключають файли ресурсів і форм
USERES("mainhello.res");
USEFORM("hello.cpp", Form1);

// об'ява класу TForm1 як похідного від базового бібліотечного класу TForm
class TForm1 : public TForm

// прототипи функцій - обробників подій
void __fastcall Button1Click(TObject*Sender);

// встановлення блакитного коліру й відображення прямокутника
Image1->Canvas->Pen->Color=clBlue;
Image1->Canvas->Rectangle(50,40,LX,LY);
```

5. РЕКОМЕНДАЦІЇ ДО ОФОРМЛЕННЯ ПОЯСНЮВАЛЬНОЇ ЗАПИСКИ

Пояснювальна записка до курсового проекту оформлюється відповідно до вимог нормативних документів з курсового проектування Міністерства освіти і науки України, основним з яких є ДСТУ 3008-95 «Документація. Звіти у сфері науки і техніки. Структура і правила оформлення». Витяг з

цього стандарту у частині вимог до оформлення студентських робіт викладено у методичних вказівках по оформленню студентських робіт [9].

Пояснювальна записка повинна буди скріплена й мати обкладинку. До неї додається (вшивається) конверт (файл) з носієм програмного продукту (Flash-карта, компакт-диск). Конверт та носій повинен мати заголовок, у якому позначається тема розробки й ПІБ автора.

5.1. СТРУКТУРА ПОЯСНЮВАЛЬНОЇ ЗАПИСКИ

Пояснювальна записка повинна включати розділи й елементи, які наведені у Таблиці 5.1.

Таблиця 5.1 - Структура пояснювальної записки

№ розділу	Назва структурного елемента	Орієнтовна кількість сторінок
	Титульний лист	1
	Лист завдання	1
	Реферат	1
	Зміст	1
	Перелік позначень, символів, одиниць, скорочень і термінів	1
	Вступ	1
1	Постановка задачі	1
2	Опис предметної області	1
3	Огляд і аналіз існуючих аналогів	1
4	Вибір і обґрунтування технології побудови системи	2
5	Алгоритм роботи системи	2-3
6	Проектування Діалогів, меню і інтерфейсу	1
7	Опис програмної реалізації	1-3
8	Комплект постачання	1-2
9	Тестування системи	1-4
	Висновки	1
	Перелік літератури	1
	Додаток А. Технічне завдання	3-5
	Додаток Б. Інструкція користувачеві	1-3
	Додаток В. Екранні форми	1-3
	Додаток Г. Результати роботи режиму «Тестування»	1-2
	Додаток Д. Результати роботи режиму «Адміністратор»	1-2
	Додаток Е. Результати роботи режиму «Навчання»	1-2
	Додаток Ж. Лістинг програмних модулів	5-10

Приклад оформлення розділів «ТИТУЛЬНИЙ ЛИСТ», «ЛИСТ ЗАВДАННЯ», «РЕФЕРАТ», «ЗМІСТ», «ПЕРЕЛІК ПОЗНАЧЕНЬ...» наведено у Додатках до методичних вказівок.

5.2. ТИТУЛЬНИЙ ЛИСТ

Титульний лист оформляється відповідно до вимог, викладених у методичних вказівках з оформлення студентських робіт. При здачі курсового проекту на перевірку студент зобов'язаний поставити свій підпис і дату здачі пояснювальної записки. Якщо норма контроль курсового проекту здійснюють не керівники роботи, то додатково на титульний лист крім блоків «Перевірів» і «Виконав» вноситься блок «Нормоконтроль» із зазначенням посади та П.І.Б. перевіряючого. Приклад титульного листа наведений у додатку Г.

5.3. ЛИСТ ЗАВДАННЯ

Лист завдання оформляється відповідно до прикладу, наведеному у Додатку Б. Лист розташовується безпосередньо за титульним аркушем і містить відомості про розробника, а також короткий перелік вимог до розробки. Тема курсового проекту в аркуші завдання повинна збігатися з темою, винесеною на титульний лист.

У пункті «Вхідні дані до проекту» зазначаються:

- розділи теоретичної частини, які необхідно висвітлити в роботі;
- перелік постійних величин, використовуваних при розрахунках (для обчислювальних задач);
- перелік методів рішення і т.д.

У пункті «Перелік очікуваних результатів» приводять найменування розроблювальних частин курсового проекту.

Пункт «Рекомендована література» повинен містити список джерел з тематики курсового проекту.

Кількість джерел - не менше двох. Кожне джерело повинно бути оформлено згідно з вимогами (див. методичні вказівки з оформлення студентських робіт [9]).

5.4. РОЗДІЛ «ВСТУП»

Обов'язковий розділ. Розділ не нумерується. Тут наводиться оцінка сучасного стану розв'язуваної в даній роботі науково-технічної задачі, обґрунтування необхідності виконання проекту. У вступі обов'язково висвітити актуальність і практичну значимість виконання курсового проектування.

5.5. РОЗДІЛ «ПОСТАНОВКА ЗАДАЧІ»

Включає в себе безпосередньо постановку задачі:

- Повне формулювання теми курсового проекту;
- Цілі та завдання розробки; основні характеристики наявних об'єктів;
- Опис вхідних та вихідних даних (які дані надходять і звідки, як вони взаємодіють, які перетворення виконуються для одержання результату);
- Опис вимог до розроблюваного програмного забезпечення.

5.6. Розділ «ПРЕДМЕТНА ОБЛАСТЬ»

Наводяться основні теоретичні відомості з тематики роботи.

Студент виконує повний опис предметної області, обґрунтування вибору використовуюваного технологічного походу. Опис предметної області повинен відповідати отриманому при видачі завдання на курсове проектування варіанту і деталізований в технічному завданні. Варіанти предметних областей наведено в додатку А.

Робиться словесний опис обраної предметної області, схема інформаційних потоків та пояснення до неї, формується словник предметної області та по-необхідності, об'єктно-орієнтований словник. Це дозволить виділити необхідні для вирішення завдання класи, властивості і методи, що полегшує опис і подальше рішення задачі.

5.7. Розділ «ОГЛЯД І АНАЛІЗ ІСНУЮЧИХ АНАЛОГІВ»

Включає огляд існуючих технологій, методів, програм і пакетів, що реалізують в тій чи іншій мірі поставлене завдання із зазначенням їх переваг та недоліків.

Крім того, в даному розділі бажано привести характеристики описаних вище програмних засобів з метою подальшого порівняння з тими, що розроблюються в рамках курсової роботи.

5.8. Розділ «ВИБІР І ОБҐРУНТУВАННЯ ТЕХНОЛОГІЇ ПОВУДОВИ СИСТЕМИ»

У даному розділі студент коротко наводить основні теоретичні положення візуальної парадигми (технології) в розробці програмних продуктів (проектування, програмування, клас, об'єкт, компонент, форма, властивості, події, методи). Описує переваги застосування візуального підходу у вирішенні завдань цього класу, наводить опис основних візуальних підходів, використаних при побудові системи, використані візуальні компоненти, обґрунтовує необхідність їх застосування та особливості. Доводить доцільність розробки власних компонент і формулює вимоги до їх створення.

5.9. Розділ «АЛГОРИТМ РОБОТИ СИСТЕМИ»

Обов'язковий розділ. Містить опис алгоритму системи в цілому і роботу його основних частин. Опис алгоритму може бути виконано у вигляді стандартної блок-схеми, R-схеми або з використанням графічної мови UML (рекомендується).

Крім графічного зображення алгоритму необхідно привести докладні коментарі до нього.

5.10. Розділ «ПРОЕКТУВАННЯ ДІАЛОГІВ, МЕНЮ І ІНТЕРФЕЙСУ»

Обов'язковий розділ. Необхідно привести обґрунтування вибору основного виду інтерфейсу (командна строчка, графічний, діалоговий або комбінований інтерфейс) і його особливості.

Додатково вказати:

- можливі підказки й повідомлення про помилки;
- рекомендовану послідовність роботи з інтерфейсними елементами.

Для графічного інтерфейсу привести:

- опис організації інтерфейсу (загальні положення);
- опис застосованих елементів інтерфейсу (назви, типи, призначення).

При проектуванні графічного інтерфейсу зробити акцент на елементи:

- меню;
- підменю;
- кнопка;
- статусна строчка;
- вікна виводу інформації;
- діалогові вікна та ін.

Меню програми повинно включати всі основні функції, які дозволяє виконувати програма. При необхідності окремі пункти меню можуть включати в себе підменю. Підменю можна організовувати або у вигляді випадаючого списку, або у вигляді окремого рядка меню. Назви пунктів меню повинні відображати їх призначення і, по можливості, спосіб виклику.

Приклади:

- пункт «F1 Допомога», означає, що для виклику довідкової інформації необхідно натиснути клавішу F1;
- пункт «**H**elp», означає, що для виклику довідкової інформації необхідно натиснути клавішу з буквою «H»;
- пункт виду «?», означає, що для виклику довідкової інформації необхідно натиснути клавішу з символом «?».

Кнопки пунктів меню й підменю можуть бути прикрашені графічними символами (піктограми, рисунки і пр.).

Статусна строчка (підказка) відображає інформацію наступного виду:

- підказку користувачу щодо подальших дій, наприклад: «Для вибору пункту меню натисніть відповідну клавішу», «Для повернення в головне меню натисніть Esc»;
- інформацію про призначення кнопки меню або підменю, наприклад «Виклик режиму адміністрування»;
- аналогічну інформацію.

Повідомлення про помилки можуть відображатися, наприклад, в спеціальних діалогових вікнах і мати вигляд: «Неможливо відобразити

довідкову інформацію по розділу – відсутній текст довідки». Приклад програмної реалізації блоку:

```
// перевірка, чи введена чи якась інформація у довідці до розділу
if(Edit1->Text == "")
{ ShowMessage("Неможливо відобразити довідкову інформацію по розділу –
відсутній текст довідки");
return ; }
```

5.11. Розділ «ОПИС ПРОГРАМНОЇ РЕАЛІЗАЦІЇ»

Обов'язковий розділ.

Опис програмної реалізації включає наступні пункти:

- функціонально-логічну схему програми;
- опис структур и форматів вхідних і вихідних даних (у тому числі файлів);
- схему взаємодії програмних модулів та їх опис;
- опис основних процедур и функцій (за виключенням стандартних, які середовище C++ Builder формує автоматично);
- схему потоків даних;
- діаграму класів і діаграму об'єктів (бажано, для полегшення розуміння задачі).

Опис логіки роботи програми (функціонально-логічна схема). Зазвичай представляється у вигляді блок-схеми або її аналога в системі UML.

Схема відображає укрупнений алгоритм роботи програми з урахуванням інтерфейсних рішень. Назви блоків у схемі повинні відповідати їх смислового призначенням. Не рекомендується позначати блоки абстрактними іменами виду «Блок 1», «Блок 2», «Виконання дії 1» і т.п. Також не рекомендується використовувати необґрунтовані скорочення слів, наприклад, замість «Вив. рез-тов» необхідно писати «Вивід результатів».

Схема відображає укрупнений алгоритм роботи. При описі функціонально-логічної схеми програми бажано зробити посилання на модулі або підпрограми, в яких реалізований той або інший блок, проте в самій схемі імена підпрограм не вказують.

Для наочності відображення логіки роботи програми можна використовувати будь-які з існуючих стандартів опису.

Обов'язково робиться опис структур та форматів даних (вхідних, вихідних і проміжних). Крім того треба навести опис структури застосованої бази даних, форматів її полів, ключові вирази, що застосовуються для зв'язку таблиць.

Приклад опису структури бази даних

База даних даного проекту представлена трьома таблицями: «Питання», «Теми», «Розділи» («Підтеми»). У першій з них зберігаються власне питання і відповіді, друга являє собою довідник тем, третя - розділів відповідної теми.

Таблица «Питання» складається з 8 полів - «Текст питання», «Відповідь № 1», «Відповідь № 2», «Відповідь № 3», «Відповідь № 4», «Відповідь № 5», «код правильної відповіді», «код теми» (див.рис.5.1). Таким чином, вся необхідна інформація про питання міститься в одному записі. При заповненні бази ми вводимо питання, відповіді та код правильної відповіді з метою повідомити програму, яка з введених відповідей вірна. У режимі опитування програма приймає варіант відповіді від користувача, порівнює його зі свідомо правильним і на цій підставі або дає студенту додатковий бал, або ні. При підрахунку балів визначається процентне співвідношення правильних і помилкових відповідей, які і видаються в спеціальному вікні по завершенні тесту.

Рис.5.1 Структура таблиці «Питання»

Структура таблиці «Теми» приведена на рис. 5.2. Поле «Код теми» є ключовим для зв'язку таблиць «Питання» і «Теми».

Рис.5.2 Таблица «Теми»

Кожній з можливих тем відповідає деяке число - «код теми». При цьому кожному запису в таблиці «Питання» можна поставити у відповідність код теми, на яку задається питання.

Зв'язок між таблицями здійснюється по ключовому полю «код теми», як це показано на рис.5.3:

Аналогічну структуру має й таблиця «Розділи» («Підтеми»). Вона включає поля «Код підтеми», «Підтема», «Код теми». Ключовим для зв'язку таблиць є також поле «Код теми».

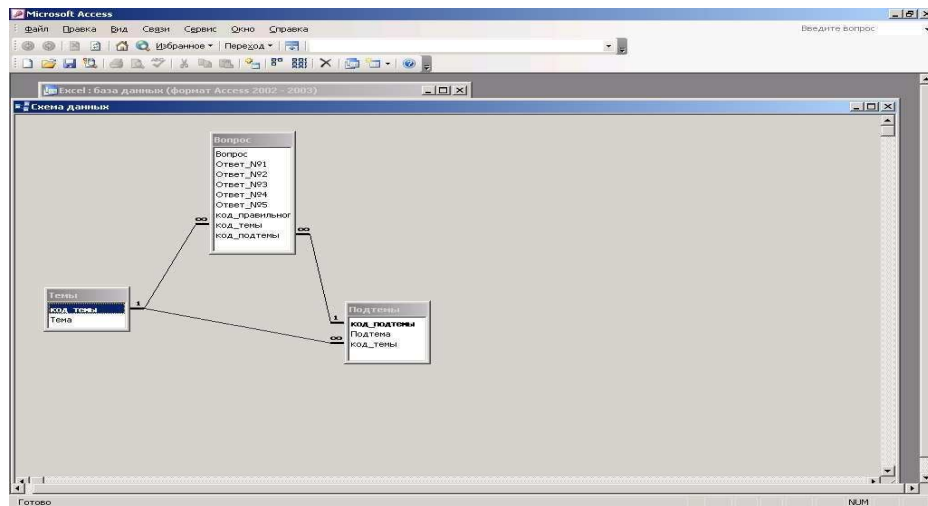


Рис.5.3 Схема даних

Слід звернути увагу, що при описі бази даних треба привести не тільки копію екрана з загальним виглядом таблиці «Питання» (рис.5.4), а й окрему таблицю тестових запитань та відповідей до них (таблиця 5.2).

Вопрос	Ответ_№1	Ответ_№2	Ответ_№3	Ответ_№4	Ответ_№5	код	код_п	код_г	
Что означает оператор Option Explicit?	тип новых переменных	Запрещение использования	Разрешение использования	оператор	н	Определение	2	4	12
Укажите оператор или структуру для объявления оператора Private	Оператор Private	Оператор Option	Оператор Dim	<	Оператор Left	Оператор Const	3	4	12
Найдите ошибочный оператор кода, on cmdEnd.Caption: txtNum1.Text="&txtNum2.Caption: lblRes1="&frmNew.Caption	1	4	1				1	4	1
Для создания новой формы проекта ну?	Команду меню Ф	Команду View - Т	Команду View - Р	Нажать люф	Команду Close		2	4	2
Для управления выводом панели элементов	Команду меню \	Команду Project	Команду View - Р	Нажать люф	Команду View		2	4	2
Для изменения свойств объектов объектов	Открыть окно св	Выполнить кома	Выполнить кома	Нажать люф	Войти в меню		2	4	2
врповорврповрпро	keeeee	kkkkk	rrrrr	ooooo	bbbbbb		2	4	1
Константу в VBA можно описать следующим Dim ИМЯ(Разме Const ИМЯ = Выл Dim ИМЯ As Тип Public Функ RedDim Presen	3	4	12				3	4	12
Какие значения содержит переменная	1	2	3	4	5		2	4	12
Массив в VBA можно описать так	6	7	8	9	10		5	4	12
Каково назначение клавиши F5?	Копирование файла	Перенос файла	Выход из Norton	Пользовате	Удаление фай		1	5	3
Каково назначение клавиши F4?	Выдача справочн	Удаление файла	Перенос файла	Просмотр ф	Редактирован		5	5	4
Для чего предназначена комбинация к/	Выбор диска на	Поменять панел	Создание нового	Убрать праи	Убрать левую		5	5	3

Рис.5.4. Копія екрана з загальним виглядом таблиці «Питання»

Перелік питань й відповідей до них для проведення тестування за обраною темою наведено у Таблиці 5.2.

Таблиця 5.2. Перелік питань й відповідей для тестування

№ п/п	ФОРМУЛІРОВКА ПИТАННЯ	РОЗДІЛ	ТЕМА
11	Які значення містить змінна типу Boolean(Логічний)?	3	8
ВІДПОВІДІ			
1	Цифрові		
2	Символьні		
3	0 або 1		
4	True/False		
5	Текстові		
Код правильної відповіді		4	

Кількість символів, що відведено для формулювання питання та відповідей на нього залежить від формату полів таблиці, опис яких треба навести, у вигляді таблиці, або копії підтверджуючого екрану:

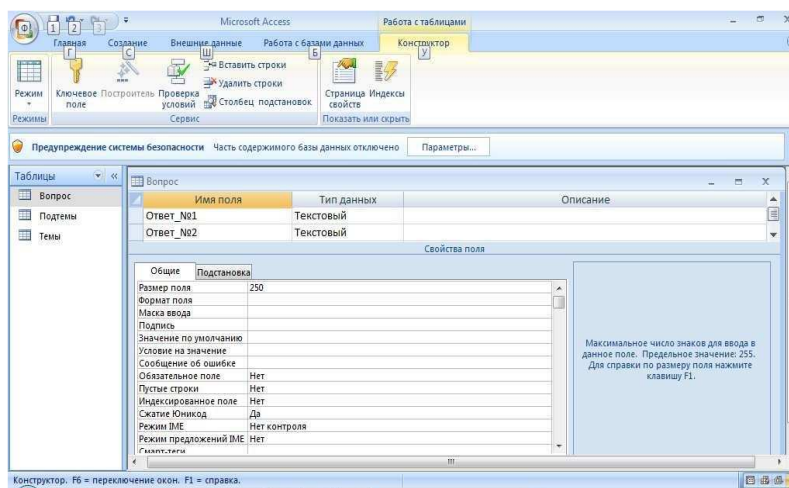


Рис.5.5. Копія екрану з описом поля «Відповідь№1» таблиці «Питання»

Кількість можливих варіантів відповідей повинно бути не менш 4 або 5. Колонка «№ п/п» - це порядковий номер питання (вверху таблиці) або номер відповіді. Графа «Тема» – номер теми, що розробляється (уточнюється у викладача), «Розділ» - порядковий номер теоретичного розділу (підтеми) з обраної тематики (повинно бути не менш трьох підтем).

Якщо в якості даних виступають файли, то наводиться опис їх формату й призначення.

Опис програмних модулів. Наводиться опис основних програмних модулів (назва, призначення і т.п.), а також схема їх взаємодії (порядок звернень модулів один до одного).

Якщо в процесі програмування розроблялися спеціальні функції або процедури, то їх опис виконується в наступному форматі:

- Назва;
- Призначення процедури або функції;
- Параметри підпрограми (імена, типи, призначення)

Якщо в процесі розробки були реалізовані власні класи, то необхідно також привести їх опис і ієрархію.

5.12. Розділ «КОМПЛЕКТ ПОСТАЧАННЯ»

Обов'язковий розділ.

При описі комплекту поставки вказують за допомогою якого продукту створено завантажувальний диск, перелік файлів і папок, які входять в комплект і описують їх призначення.

При описі порядку установки програми на комп'ютері користувача необхідно вказати:

- на якому носії поставляється програмний продукт;
- які файли і папки необхідно скопіювати з носія (або запустити на виконання безпосередньо з зовнішнього носія);
- які допоміжні файли або програми мають бути додатково скопійовані або встановлені на комп'ютер користувача, і в які папки;
- які настройки треба виконати для роботи системи.

5.13. Розділ «ТЕСТУВАННЯ СИСТЕМИ»

Обов'язковий розділ.

У даному розділі наводиться порядок і результати тестування програми в різних режимах. Розділ може бути оформлений у вигляді додатка.

Якщо в результаті роботи програми створюються файли результатів, то тексти файлів можна також помістити в додаток, а в даному розділі привести відповідні посилання.

5.14. Розділ «ВИСНОВКИ»

Є обов'язковим розділом. У структурі документа даний розділ не нумерується. Студент повинен вказати в даному розділі поставлені і виконані завдання в процесі виконання курсового проектування. Також необхідно вказати переваги і недоліки спроектованої системи і можливі перспективи щодо подальшого поліпшення і використання проекту.

5.15. Розділ «ПЕРЕЛІК ЛІТЕРАТУРИ»

Є обов'язковим розділом. У структурі документа даний розділ не нумерується. Містить список літературних джерел, використаних студентом

при виконанні курсового проектування. Допускається використання загальнодоступних електронних джерел.

Джерела повинні бути оформлені відповідно до існуючих стандартів (див. методичні вказівки з оформлення студентських робіт [9] та ДСТУ ГОСТ 7.1:2006.

5.16. Розділ «ДОДАТКИ»

Додатки до пояснювальної записки розташовують після списку літератури. У змісті відображають лише назву додатка та номер сторінки, з якої він починається. Зміст додатків повинен відповідати їх назві.

5.16.1. ТЕХНІЧНЕ ЗАВДАННЯ

Обов'язковий розділ.

Оформлюється відповідно до вимог, викладених у методичних вказівках з оформлення студентських робіт [9]. Для даного курсового проекту виставлені наступні вимоги до виконання розділів та пунктів технічного завдання.

Пункт «Загальні відомості». Включає назву теми курсового проекту, інформацію про розробника, терміни початку і закінчення роботи, дати проміжних звітних періодів, нормативні документи, на підставі яких виконується робота.

Рекомендації до порядку і термінів виконання курсового проекту були приведені в табл. 2.1.

Пункт «Призначення і цілі створення програми» містить відомості про коло користувачів, для яких призначений розроблювальний програмний продукт (ПП) і про можливі сфери його застосування.

Пункт «Характеристика об'єкту автоматизації» включає опис предметної області відповідно до Завдання на курсове проектування та відомості про умови експлуатації.

Пункт «Вимоги до програмного продукту» включає в себе:

- вимоги до розробки в цілому;
- вимоги до режимів і функцій, виконуваних програмою;
- вимоги до технічного забезпечення (опис мінімальної конфігурації комп'ютера, яка необхідна для нормального функціонування програмного продукту);
- вимоги до програмного забезпечення (опис характеристик програмного забезпечення, необхідного для реалізації поставлених завдань; вказівки до використання конкретних мов і середовищ програмування);
- вимоги до організаційного забезпечення (включають в себе перелік документів і додатків, які входять в програмну документацію: пояснювальна записка; технічне завдання, керівництво користувача, екранні форми, тексти програм, результати роботи), а також вимоги до установки і постачання програмного продукту.

До таких вимог, безперечно, відносяться й спеціальні вимоги до програмного продукту, що розробляється в рамках курсового проекту, так:

- робота в режимах тестування, адміністрування й довідки;
- захист від несанкціонованого доступу до режиму адміністрування (поповнення (змінення) питань);
- наявність тестової бази – не менш 3-х розділів, у кожному – не менш, ніж 10 питань;
- обмеження на час пошуку відповіді (має налаштовуватися адміністратором);
- простий програмний інтерфейс;
- зручність адміністрування;
- повна автоматизація процесу тестування;
- можливість користувачу вибрати для тестування тему взагалі або її окремий розділ;
- відображення результатів тестування на екрані й в файлі (або папері);
- наявність демонстраційно-довідкового матеріалу і підказок;
- працездатність додатків, ефективність за часом реакції системи;
- пропозиції про постачання додатків користувачу.

5.16.2. ІНСТРУКЦІЯ КОРИСТУВАЧЕВІ

Обов'язковий додаток.

Містить опис основних функціональних можливостей програми, порядок її встановлення, докладний опис функціонування.

В даному додатку допускається більш вільний стиль викладу матеріалу в порівнянні з основним текстом пояснювальної записки (наприклад, у вигляді звернення до користувача). Текст «Інструкції Користувача» повинен відповідати тому, що розміщено в режимі «Довідка про систему».

5.16.3. ЕКРАННІ ФОРМИ

Є обов'язковим додатком. Містить екранні форми вікон виконання програми (порядок форм повинен відповідати реальній послідовності виконання програми, до кожної форми треба надати пояснення). Кожна екранна форма повинна бути представлена як малюнок. Підпис до малюнка повинен відображати призначення представленого вікна програми.

Приклади опису екранних форм.

«У даному режимі програми передбачено режим доповнення і зміни бази даних. Вхід в базу в цьому режимі здійснюється після вводу паролю викликом вікна «Редагування бази» (див. рис.5.6). Тут адміністратор може доповнити і змінити довідник запитань і відповідей. У вікні відображається кількість записів в базі по тій або іншій темі».

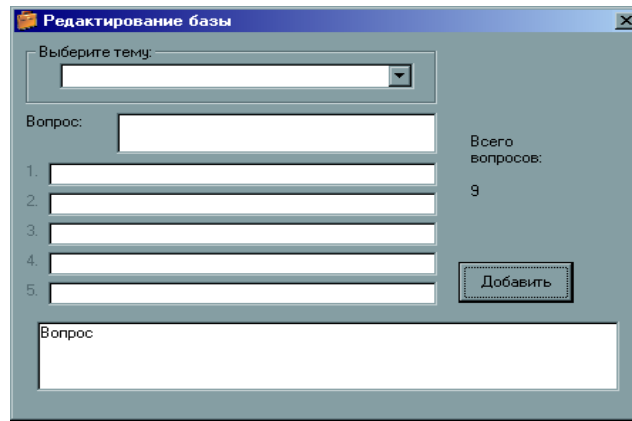


Рис.5.6 Вікно редагування бази даних

«Вікно тестування. На ньому відображається номер та зміст питання та варіанти відповідей на нього. Час реакції на запитання задається у розділі «Адміністрування» й при тестуванні не змінюється. Рекомендовано відводити на одне питання до 30 сек. Кнопка «Далі» переводить до слідуєчого запитання».

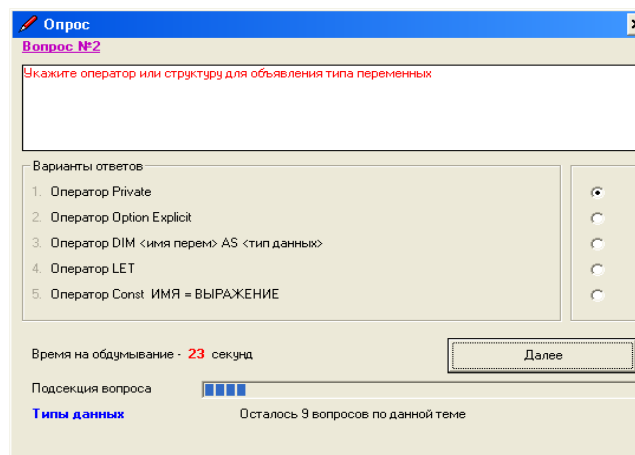


Рис.5.7 Вікно тестування

5.16.4. ЛІСТИНГ ПРОГРАМНИХ МОДУЛІВ

Є обов'язковим додатком. Лістинг програмних модулів оформлюється відповідно до рекомендацій методичних вказівок з оформлення студентських робіт. Лістинг кожного окремого модуля повинен в обов'язковому порядку містити назву модуля і короткий опис його призначення. Коментарі повинні бути присутніми в достатній кількості для розуміння описуваного ними програмного коду. Вітається користування певними нотаціями іменування об'єктів програми (змінних, класів і методів).

З метою поліпшення візуального представлення програмного коду, допускається використання шрифтів, відмінних від Times New Roman. Роздруківка лістингу у дві колонки допускається в тому випадку, якщо при цьому не порушується читабельність програмного коду.

ПЕРЕЛІК РЕКОМЕНДОВАНОЇ ЛІТЕРАТУРИ

1. Шамис В. М.: Нолидж, Borland C++ Builder 3. Техника визуального программирования. М: ЗАО “Издательство БИНОМ”, 1998. - 512 с.
2. Аленский Н.А. Визуальное объектно-ориентированное программирование в задачах. : Учеб.пособие,— Мн.: БГУ, 2008, — 114с.
3. Архангельский А. Я. Справочное пособие по C++Builder 6. Книга 1. Язык C++ — М: ЗАО “Издательство БИНОМ”, 2002.
4. Архангельский А. Я. C++Builder 6. Справочное пособие. Книга 2. Классы и компоненты. — М: ЗАО “Издательство БИНОМ”, 2002. — 528с.
5. Культин Н. C++ Builder в задачах и примерах. – СПб.: БХВ-Петербург, 2007. – 336с
6. Мацяшек Л. Анализ требований и проектирование систем. Разработка информационных систем с использованием UML / Лешек А. Мацяшек. - М.: Вильямс, 2002. - 432 с.
7. Буч Г. Язык UML. Руководство пользователя / Буч Г., Рамбо Дж., А. Якобсон. – М.: ДМК Пресс, 2007. - 2-е изд. - 496 с.
8. Кормен Т. Алгоритмы: Построение и анализ / Кормен Т., Лейзерсон Ч., Ривест Р., Штайн К. - М.: Вильямс, 2005. - 1296 с.
9.Золотухина О.А. Методические указания по оформлению студенческих работ / О.А. Золотухина, Е.А. Маслова, Н.П. Пулинец. - Донецк : ИПИИ «Наука і освіТа», 2010. - 48 с.
10. Курс: Технология создания программных и интеллектуальных систем /Центр дистанционного обучения ДонНТУ, - [Электронный ресурс]. http://temp1.donntu.edu.ua/course/view.php?id=48
11. Леоненков А.В. Язык UML 2 в анализе и проектировании программных систем и бизнес-процессов [Электронный ресурс]. - Режим доступа : http://www.intuit.ru/department/se/uml2/

ДОДАТОК А. ПЕРЕЛІК РЕКОМЕНДОВАНИХ ТЕМ

1. Мови й середовища програмування
2. Соціальні мережі
3. Програмування мовою С++
4. Програмування у середовищі С++ Builder
5. Технології програмування
6. Технічні засоби програмування
7. Об'єктно - орієнтоване програмування
8. Основи комп'ютерної безпеки
9. Телекомунікації
10. Бездротові технології
11. Розробка й розвиток сайтів
12. Технологія пошуку даних в Інтернеті
13. Структура Інтернет-комунікацій
14. Організація електронної пошти
15. Основи мережевих технологій
16. Комп'ютерна електроніка
17. Алгоритми й структури даних
18. Історія інформатики
19. Комп'ютерна логіка
20. Дискретна математика
21. Правила дорожнього руху
22. Іноземна мова
23. Кінологія
24. Інтер'єр
25. Сільське господарство
26. Декор
27. Садівництво
28. Теорія музики
29. Культурологія
30. Історія України

Додаток Б. ПРИКЛАД ОФОРМЛЕННЯ ЛИСТА ЗАВДАННЯ

МІНІСТЕРСТВО ОСВІТИ І НАУКИ, МОЛОДІ ТА СПОРТУ УКРАЇНИ
ДВНЗ «ДОНЕЦЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ»

Факультет: _____
Кафедра: Програмного забезпечення інтелектуальних систем
Спеціальність: _____

ЗАВДАННЯ НА КУРСОВИЙ ПРОЕКТ з дисципліни «ОСНОВИ ВІЗУАЛЬНОГО ПРОГРАМУВАННЯ»

Студенту _____ групи _____
(фамілія, ім'я, по батькові)

Тема проекту «Використання технологій візуальної розробки додатків
при побудові тестово-довідкової системи «<Назва предметної області>»

Вхідні дані Теоретичні відомості з технології візуального програмування,
Перелік тем для розробки тестів, рекомендована структура бази даних,
вимоги (завдання) до розробки програмного продукту й результату його роботи.

Перелік очікуваних
результатів Програмний продукт, побудований за технологією
візуального програмування, що в діалоговому режимі тестує знання
користувача, проводить оцінку рівня володіння предметною областю, та
надає довідкову інформацію, яка необхідна для подальшого вивчення теми.
Довідкова інформація подається у вигляді: а) презентації; б) текстового файлу
з теоретичними відомостями з розділу, де виникла основна проблема;
в) переліку рекомендованої літератури; д) посилання на інтернет-ресурс, у якому
розкрито бажана тема; г) відео-роліку.

Рекомендована
література Архангельский А. Я. Программирование в С++Builder 6 —
М: ЗАО “Издательство БИНОМ”, 2003. — 1152с.

Дата видачі завдання _____
Дата захисту проекту _____

Керівник _____
(підпис) _____ (посада, П.І.Б.)

Консультант _____
(підпис) _____ (посада, П.І.Б.)

Розробник _____
(підпис) _____ (П.І.Б.)

**Додаток В. ПРИКЛАД ОФОРМЛЕННЯ «ТЕХНІЧНОГО
ЗАВДАННЯ»**

МІНІСТЕРСТВО ОСВІТИ І НАУКИ, МОЛОДІ ТА СПОРТУ УКРАЇНИ
ДВНЗ «ДОНЕЦЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ»

Д080403.1.01.05/085.КП

Кафедра програмного забезпечення
інтелектуальних систем

ТЕХНІЧНЕ ЗАВДАННЯ

до курсового проекту
з дисципліни «ОСНОВИ ВІЗУАЛЬНОГО ПРОГРАМУВАННЯ»
тема: «Використання технологій візуальної розробки додатків при побудові
тестово-довідкової системи «<Назва предметної області>»

Керівник:

(дата)

(підпис)

(посада, П.І.Б.)

Консультант:

(дата)

(підпис)

(посада, П.І.Б.)

Розробник:

ст. гр. _____

(дата)

(підпис)

(П.І.Б.)

Донецьк,
20____ р.

Додаток Г. ПРИКЛАД ОФОРМЛЕННЯ ТИТУЛЬНОГО ЛИСТА

МІНІСТЕРСТВО ОСВІТИ І НАУКИ, МОЛОДІ ТА СПОРТУ УКРАЇНИ
ДВНЗ «ДОНЕЦЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ»

Д080403.1.01.05/085.КП

Кафедра програмного забезпечення
інтелектуальних систем

КУРСОВИЙ ПРОЕКТ

з дисципліни «ОСНОВИ ВІЗУАЛЬНОГО ПРОГРАМУВАННЯ»
тема: «Використання технологій візуальної розробки додатків при побудові
тестово-довідкової системи «<Назва предметної області>»

Керівник:

(дата) (підпис) (посада, П.І.Б.)

Консультант:

(дата) (підпис) (посада, П.І.Б.)

Розробник:

ст. гр. _____

(дата) (підпис) (П.І.Б.)

Донецьк,
20____ р.

Додаток Д. ПРИКЛАД ОФОРМЛЕННЯ ЛИСТА «РЕФЕРАТ»

РЕФЕРАТ

Пояснювальна записка: 46 с., 10 рис., 2 табл., 5 джерел, 4 додатка

Об'єктом розробки є тестово-довідкової системи «<Назва предметної області>».

Ціль курсового проектування є закріплення й розвинення навичок самостійного вирішення завдань в області візуального проектування і програмування на прикладі розробки тестово-довідкової системи.

Для досягнення цієї цілі були застосовані такі методи й технології, як: <перелік застосованих методів>.

В результаті виконання курсового проекту було досягнуто: <перелік досягнутих результатів>.

Програмний продукт має такі особливості: <характеристики програмного продукту> й може бути застосований <область застосування>.

<Висновки по роботі в цілому>

ВІЗУАЛЬНЕ ПРОГРАМУВАННЯ, ПРОЕКТУВАННЯ, КОМПОНЕНТ, ВЛАСТИВОСТІ, ПОДІЇ, МЕТОДИ, КЛАСИ, ТЕСТУВАННЯ, ДОВІДКА

					Д050103.1.01.09/190.КП			
		Прізвище	Підпис	Дата				
Розробник					<Тема курсового проекту>	Літера	Лист	Листів
Керівник. Проекту						у	3	46
Консультант								
Н. контроль								
Зав. каф.	А.І. Шевченко							
						каф. ПЗІС група _____		

Додаток Е. ПРИКЛАД ОФОРМЛЕННЯ ЛИСТА «ЗМІСТ»

	Вступ	5
1	Постановка задачі	6
2	Опис предметної області	7
3	Огляд і аналіз існуючих аналогів	8
4	Вибір і обґрунтування технології побудови системи	9
5	Алгоритм роботи системи	11
6	Проектування Діалогів, меню і інтерфейсу	13
7	Опис програмної реалізації	14
8	Комплект постачання	16
9	Тестування системи	18
	Висновки	20
	Перелік літератури	21
	Додаток А. Технічне завдання	22
	Додаток Б. Інструкція користувачеві	25
	Додаток В. Екранні форми	29
	Додаток Г. Результати роботи режиму «Тестування»	32
	Додаток Д. Результати роботи режиму «Адміністратор»	35
	Додаток Е. Результати роботи режиму «Навчання»	39
	Додаток Ж. Лістинг програмних модулів	40

Розробник	Прізвище	Підпис	Дата	Д050103.1.01.09/190.КП	Лист
ст.гр. _____	Х.У.ХХХХХ				4

Додаток Є. ПРИКЛАД ОФОРМЛЕННЯ ЛИСТА «ПЕРЕЛІК ПОЗНАЧЕНЬ»

ПЕРЕЛІК ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

- ВП - Візуальне програмування
ОО - Об'єктно – орієнтований
UML - Unified Modeling Language
ПП - Програмний продукт
ПЗ - Програмне забезпечення
ПрО - Предметна область

Разроботал	Фамилия	Подпись	Дата	Д050103.1.01.09/190.КП	Лист
ст.гр. ПО-09в	А.О. Шевченко				5

Додаток Ж. ПОРЯДОК РОЗРОБКИ, ОЦІНЮВАННЯ І ЗАХИСТУ ПРОЕКТУ

Студент спільно з керівником формулює тему курсової роботи на початку курсового проектування. Тема фіксується у вигляді назви на титульному аркуші роботи, назва теми пов'язане з обраної тематики проекту (перелік предметних областей для вибору тем наведено у додатку А). Заміна теми після її затвердження не допускається.

Отримавши завдання на курсову роботу, студент разом з керівником складає календарний план, який служить надалі документом для контролю та координації робіт по її виконанню (див. розділ 2).

В ході роботи студент на заняттях, а також самостійно вивчає інформацію по заданій тематиці, робить замітки, виписки, складає необхідні алгоритми, діаграми, класи та програми і налагоджують їх на ЕОМ, компонує матеріали за розділами, формує документацію до програмного продукту

Керівники відстежують виконання роботи, роблять відповідні позначки. Контрольні терміни - 5 і 10 тижні семестру (для денного відділення).

Крім того у відповідності з встановленим графіком студент зобов'язаний відвідувати консультації.

Результатом завершального етапу курсового проектування повинна бути пояснювальна записка до курсового проекту і розробити програмний продукт.

Пояснювальну записку студент представляє на попередню захист з відміткою нормоконтролера не пізніше, ніж за тиждень до дати захисту (для студентів заочного відділення - не пізніше, ніж за 2 тижні до дати захисту), для визначення ступеня готовності проекту.

Після попереднього захисту студент усуває виявлені недоліки, пов'язані з роботою програмного продукту, оформленням і змістом пояснювальної записки проекту.

Захист курсового проекту проводиться перед комісією, до складу якої входить не менше двох викладачів.

Графік захистів складається керівниками проекту, затверджується завідувачем кафедри і вивіщується на загальнодоступних дошках оголошень не пізніше п'яти днів до першого захисту.

Оцінка по курсовому проектування є комплексною і включає в себе оцінку:

- роботи студента на протязі всього семестру;
- роботоздатності програмного продукту;
- повноти пояснювальної записки;

- якості захисту курсового проекту.

При оцінюванні роботи студента протягом всього семестру враховується:

- повнота виконання частин курсового проекту;
- своєчасність виконання задач і частин курсового проекту.

При оцінці програмного продукту враховується:

- правильність роботи додатків;
- відповідність програм вимогам технічного завдання до функціональних характеристик, вимогам до інтерфейсу та ін. (вимоги до оформлення технічного завдання наведено у розділі 5).

При оцінюванні пояснювальної записки враховується:

- Відповідність змісту пояснювальної записки до вимог;
- правильність оформлення;
- стиль и повнота викладення матеріалу.

В процесі захисту курсового проекту враховується:

- Доповідь студента (подання роботи);
- ступінь самостійності виконання роботи;
- повнота і правильність відповідей на запитання комісії.

У разі якщо студент отримав незадовільну оцінку, йому видається нова тема курсового проекту, і призначаються нові терміни його виконання.

Курсовий проект рекомендується виконувати у відповідності з графіком, наведеним у розділі 2.

ЗМІСТ

ВСТУП.....	4
1. ЦІЛІ Й ЗАДАЧІ КУРСОВОГО ПРОЕКТУВАННЯ.....	5
2. ЕТАПИ КУРСОВОГО ПРОЕКТУВАННЯ.....	5
3. ЗАВДАННЯ НА КУРСОВЕ ПРОЕКТУВАННЯ.....	8
4. РЕКОМЕНДАЦІЇ ДО РОЗРОБКИ.....	9
5. РЕКОМЕНДАЦІЇ ДО ОФОРМЛЕННЯ ПОЯСНЮВАЛЬНОЇ ЗАПИСКИ.....	10
5.1. Структура пояснювальної записки.....	11
5.2. Титульний лист.....	12
5.3. Лист завдання.....	12
5.4. Розділ «ВСТУП».....	12
5.5. Розділ «ПОСТАНОВКА ЗАДАЧІ».....	12
5.6. Розділ «ПРЕДМЕТНА ОБЛАСТЬ».....	13
5.7. Розділ «ОГЛЯД І АНАЛІЗ ІСНУЮЧИХ АНАЛОГІВ».....	13
5.8. Розділ «ВИБІР І ОБГРУНТУВАННЯ ТЕХНОЛОГІЇ ПОБУДОВИ СИСТЕМИ».....	13
5.9. Розділ «АЛГОРИТМ РОБОТИ СИСТЕМИ».....	13
5.10. Розділ «ПРОЕКТУВАННЯ ДІАЛОГІВ, МЕНЮ І ІНТЕРФЕЙСУ».....	14
5.11. Розділ «ОПИС ПРОГРАМНОЇ РЕАЛІЗАЦІЇ».....	15
5.12. Розділ «КОМПЛЕКТ ПОСТАЧАННЯ».....	19
5.13. Розділ «ТЕСТУВАННЯ СИСТЕМИ».....	19
5.14. Розділ «ВИСНОВКИ».....	19
5.15. Розділ «ПЕРЕЛІК ЛІТЕРАТУРИ».....	19
5.16. Розділ «ДОДАТКИ».....	20
5.16.1. Технічне завдання.....	20
5.16.2. Інструкція користувачеві.....	21
5.16.3. Екранні форми.....	21
5.16.4. Лістинг програмних модулів.....	22
ПЕРЕЛІК РЕКОМЕНДОВАНОЇ ЛІТЕРАТУРИ.....	23
Додаток А. ПЕРЕЛІК РЕКОМЕНДОВАНИХ ТЕМ.....	24
Додаток Б. ПРИКЛАД ОФОРМЛЕННЯ ЛИСТА ЗАВДАННЯ.....	25
Додаток В. ПРИКЛАД ОФОРМЛЕННЯ «ТЕХНІЧНОГО ЗАВДАННЯ».....	26
Додаток Д. ПРИКЛАД ОФОРМЛЕННЯ ЛИСТА «РЕФЕРАТ».....	28
Додаток Е. ПРИКЛАД ОФОРМЛЕННЯ ЛИСТА «ЗМІСТ».....	29
Додаток Є. ПРИКЛАД ОФОРМЛЕННЯ ЛИСТА «ПЕРЕЛІК ПОЗНАЧЕНЬ».....	30
Додаток Ж. ПОРЯДОК РОЗРОБКИ, ОЦІНЮВАННЯ І ЗАХИСТУ ПРОЕКТУ.....	31