

## ТЕХНОЛОГІЯ ПРОЕКТУВАННЯ ЕЛЕКТРОННОГО ДОДАТКА ДО УРОКУ МАТЕМАТИКИ В СЕРЕДОВИЩІ ВІЗУАЛЬНОГО ПРОГРАМУВАННЯ

Скрипка Г.В.

**Анотація.** В статті досліджується актуальна проблема технології проектування електронного додатку до уроку математики в середовищі візуального програмування.

**Ключові слова:** ІКТ-компетентність, етапи створення додатку, середовище візуального програмування, розробка програмного засобу, Visual Studio.

★ ★ ★

**Постановка проблеми в загальному вигляді та її зв'язок із важливими науковими та практичними завданнями.** Підвищення кваліфікації учителя в умовах модернізації освіти повинно відображати перспективні тенденції розвитку інформаційних й інноваційних педагогічних технологій. Основною метою цього процесу є підготовка конкурентоздатного на ринку праці педагога, який вільно володіє своєю професією й орієнтується в суміжних галузях знань, здатний до ефективної роботи за фахом на рівні світових стандартів, а також готовий і здатний до постійного професійного зростання. Отже, основне завдання підвищення кваліфікації вчителів — це забезпечення умов для формування і розвитку всіх груп компетентностей.

У зв'язку із стрімким розвитком ІКТ-технологій усе актуальнішою стає проблема формування й розвитку ІКТ-компетентності педагога, а тому вищезгадані питання є обов'язковими як під час курсів підвищення кваліфікації, так і в міжкурсовий період.

Нині науковці багатьох країн вивчають питання ІКТ-компетентності вчителя як складової професійної компетентності. Уже розроблені програми з формування ІКТ-компетентності, які дозволяють вчителю опанувати основними прийомами роботи з комп'ютером. Це такі, як «Microsoft. Партнерство у навчанні» [3], «Курс цифрових технологій» [2], а також різноманітні авторські спецкурси, семінари і тренінги в рамках курсів підвищення кваліфікації.

Проте на сьогоднішній день недостатньо вивчене питання розвитку ІКТ-компетентності вчителя-предметника, який має достатній рівень знань в галузі ІКТ і бажає поглибити свої знання.

Тому нами було розроблено систему семінарів-тренінгів зі створення власних програмних засобів у середовищі візуального програмування для вчителів математики. У рамках вищезгаданого тренінгу розроблено технологію проектування програми до уроку математики в середовищі візуального програмування.

**Аналіз останніх досліджень і публікацій, у яких започатковано розв'язання даної проблеми і на які спирається автор.** Питання використання інформаційно-комунікаційних технологій у навчально-виховному процесі, а також питання ІКТ-компетентності педагога вивчали такі науковці, як П.В. Беспалов, А.А. Єлізаров, Н.В. Насирова, О.В. Овчарук, М.І. Жалдак, Н.В. Морзе, Л.А. Чернікова та ін.

Детальний аналіз змісту вказаних праць дозволяє твердити, що попри вивчення визначених вище аспек-

тів проблеми, питання розвитку ІКТ-компетентності вчителів, які мають достатній рівень ІКТ-компетентності, розроблено недостатньо. Зокрема, недостатньо розроблена технологія створення власного програмного продукту, і, зокрема, проектування програми для підтримки навчання математики.

**Виділення невирішених раніше частин загальної проблеми, котрим присвячується стаття.** Актуальність проблеми проектування програми для підтримки навчання математики очевидна, тому вона потребує спеціального дослідження. Необхідно з'ясувати:

1. Передумови використання середовища візуального програмування.
2. Етапи уроку, на яких можна застосувати створені програмні засоби.
3. Приклади створених у середовищі візуального програмування програм, які можна використати на уроці математики.
4. Етапи створення програми (Windows-дodatка) в середовищі візуального програмування.
5. Основні правила розробки інтерфейсу користувача.

**Мета цієї статті** — проаналізувати технологію проектування додатку для підтримки навчання математики.

Як зазначено у документі ICT competency standards for teachers: competency standards modules [1], концептуальні рамки проекту ЮНЕСКО за нормами компетентності вчителів у використанні ІКТ створюються на перетині трьох підходів до реформи освіти, заснованих на розвитку людських здібностей — технічної грамотності, поглиблення знань і створенні знань — і шести компонентів системи освіти — політики, програм, педагогіки, ІКТ, організації і підготовки вчителів. Згідно з ним, учителі, компетентні в застосуванні підходу на основі створення знань, повинні вміти розробляти навчальні посібники і заняття з використанням ІКТ; використовувати ІКТ для вироблення в учнів навичок створення знань і критичного мислення; надавати підтримку безперервному розумовому процесу; створювати для своїх учнів і колег суспільство знань. У зв'язку з цим розвиток ІКТ-компетентності вчителя-предметника під час підвищення кваліфікації і в міжкурсовий період передбачає навчання основ програмування.

Досліджуючи передумови використання середовища візуального програмування, учителями для створення власних програмних засобів, ми виокремили такі:



- наявність помилок у змістовому наповненні готових педагогічних програмних засобів (ППЗ);
- недостатня функціональність існуючих ППЗ;
- проблематичність інсталяції на комп'ютері деяких програмних засобів;
- невідповідність системних вимог програмних засобів наявній у навчальному закладі техніці;
- невідповідність можливостей ППЗ поставленим задачам на тому чи іншому уроці;
- висока вартість готових програмних засобів;
- бажання вчителів проявити свою творчість, креативність або просто стати відомим.

Ми пропонуємо вчителям розпочати працювати із середовищем Visual Studio, зокрема, з мовою програмування Visual Basic, оскільки вона має такі переваги порівняно з іншими:

- наочність;
- можливість програмування без програмування;
- Basic — мова для початківців (Beginner's All-purpose Symbolic Instruction Code — універсальна система символічного кодування для початківців).

Серед прикладів створених у середовищі візуального програмування програм, які можна використати на уроці математики, ми розглядаємо такі:

- тестові програми;
- програми-калькулятори (обчислення площ, об'ємів, відстаней тощо);
- інтерактивна побудова графіків функцій;
- побудова геометричних об'єктів;
- побудова динамічних об'єктів (імітація руху);
- створення секундомірів, таймерів тощо;
- робота з рекурсивними функціями.

Створені програмні засоби можна застосувати як в мультимедійній аудиторії (з використанням проектора і комп'ютера або декількох комп'ютерів одночасно), так і в звичайному комп'ютерному класі. Серед етапів уроку, на яких доцільно застосовувати створені програми, виділяємо такі:

- актуалізація опорних знань (робота з тестовими програмами, програмами-калькуляторами, програмами інтерактивної побудови графіків функцій тощо);
- вивчення нового матеріалу з використанням створених програмних засобів можливе в таких варіантах:
  - подання матеріалу вчителем (можлива робота з будь-яким із вищезгаданих типів створених програм);
  - індивідуальне вивчення матеріалу з подальшим обговоренням, узагальненням чи навчанням інших;
- закріплення нового матеріалу;
- формування вмінь і навичок;
- самоконтроль і контроль.

Як і будь-який інший вид діяльності, процес створення програми (Windows-додатка) в середовищі візуального програмування має свої чітко визначені етапи:

- розробка інтерфейсу користувача;
- проектування додатка з використанням TOE-таблиці;

- написання програмного коду;
- тестування і налагодження програми;
- збереження проекту і компіляція (перетворення проекту на додаток, який здатен самостійно працювати за межами середовища програмування). Розглянемо детальніше кожен з етапів створення програми.

#### Правила розробки інтерфейсу користувача:

- всю інформацію потрібно розміщувати або вертикально, або горизонтально. Найбільш важливі дані розташовують у лівому верхньому куті екрана;
- кнопки розміщують або внизу екрана, або у правому нижньому чи правому верхньому куті екрана;
- написи розташовують або над текстовими полями, або зліва від них;
- кнопка не повинна містити більше трьох слів (перше слово з великої літери);
- якщо форма міститиме велику кількість полів, необхідно об'єднати їх в блоки для полегшення сприйняття;
- людське око спочатку сприймає графічний об'єкт, а вже потім — текстову інформацію, а тому малюнки потрібно використовувати лише в разі необхідності;
- необхідно використовувати шрифти без засічок — вони краще сприймаються користувачем;
- одна форма — один тип шрифту й не більше двох розмірів;
- підкреслення і курсив ускладнюють сприйняття тексту;
- під час вибору кольорової гами потрібно користуватися стандартами Windows: текст чорного кольору на сірому чи білому тлі;
- в одній формі не повинно бути більше трьох кольорів.

#### Проектування додатку з використанням TOE-таблиці

TOE-таблицю (Task, Object, Event, тобто, **Завдання, Об'єкт, Подія**) використовують для того, аби сформулювати задачі, які виконуватиме додаток, а також визначити об'єкти і події, з якими пов'язані ці задачі. Прикладом може слугувати таблиця для програми, яка рахуватиме площу довільного трикутника за відомими висотою та основою (табл. 1).

Таблиця 1

Задача	Об'єкт	Подія
Отримання у користувача інформації (довжина основи трикутника та його висота)	VisotaTextBox OsноваTextBox (текстові поля)	немає
Вивід підказок для користувача (слова: основа трикутника, висота трикутника та площа трикутника)	OsноваLabel VisotaLabel PloshaLabel (написи)	Немає
Обчислення площі трикутника	CalcButton (кнопка)	Click (клацання мишею)
Відображення результату обчислення	PloshaTextBox (текстове поле)	Немає

У результаті буде створений додаток, представлений на рис. 1.

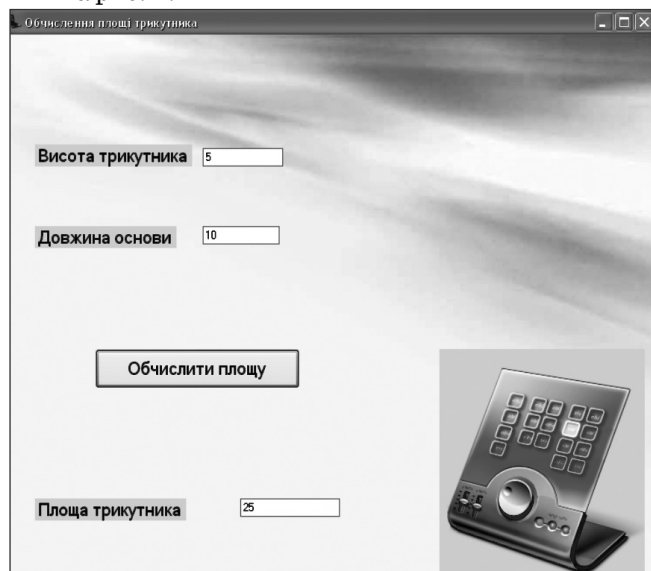


Рис. 1

### Написання програмного коду

Робота над цим етапом вимагає знання основ програмування, зокрема, алфавіту мови програмування Visual basic, складових додатка, основних операцій і операторів. Тому наразі недоцільно зупинятися детальніше на цьому етапі створення програми.

### Тестування і налагодження програми

Налагодження програми — це виявлення й усунення допущених у програмі помилок. Існує три основні методи налагодження програми:

- налагодження вручну, тобто перегляд тексту програми і ретельний його аналіз;
- використання програмних засобів комп'ютера — так званих налагоджувачів для пошуку помилок. Як варіант цього методу можна розглядати трасування, або можливість покрокового виконання програми і слідування за значеннями змінних у процесі виконання програми, яке є доступним у середовищі Visual Studio;
- використання одночасно налагодження і тестування окремих частин програми.

Тестування — це виконання комплексу вправ (завдань) для перевірки працездатності програми за будь-яких умов. Деякі принципи тестування програм:

- використання підходу захисного програмування (захист програми від некоректних даних);
- тестування граничних умов: якщо програма працює правильно при граничних значеннях тестових даних, то скоріше за все вона поведеться коректно і у звичайних умовах;
- обов'язковий аналіз результатів тестування (обчислення для порівняння результату іншим способом, наприклад, на калькуляторі; використання табличних даних; порівняння з результатом роботи іншої програми, яка обчислює той самий результат іншим способом; обернення результату тесту — чи одержуються вхідні дані, чи ні?

### Збереження проекту і компіляція

Після виконання тестування вчитель повинен перетворити проект на додаток, який здатний самостійно працювати за межами середовища програмування.

Під час компіляції програми у VS в одній з папок проекту (папка BIN) створюється запускний файл, ім'я якого одне й те саме, що й усього проекту, а розширення — *exe*. Тобто, для запуску додатку достатньо двічі клацнути на вищевказаному файлі. Подібний варіант спрацьовує, якщо програма використовуватиметься або на цьому ж комп'ютері, або на комп'ютері з однаковою версією операційної системи і системних файлів. Виходом із подібної ситуації є створення інсталяційного файлу на основі створеного проекту, який би встановлював програму на інших комп'ютерах. Цей процес охоплює такі етапи:

1. Підготовка проекту (сюди входить копіювання всіх необхідних файлів, на які є посилання у програмі, у папку Bin; проте, це не стосується завантажених файлів на етапі проектування).
2. Створення інсталяційного проекту.
3. Інсталяція на комп'ютері і робота з налагодженою програмою.

Отже, дотримання технології створення Windows-дodatка дає можливість вчителю зменшити витрати часу на конструювання й реалізацію власної програми.

**Висновки.** Отже, створення нового програмного продукту — цікавий і захоплюючий процес, який допомагає вчителю розвиватися, вдосконалюватися й оптимізувати навчальний процес. Проте ця робота передбачає ретельне планування і дотримання технології створення додатку в середовищі Visual Studio. Нагадаємо, що ця технологія передбачає:

- розробку інтерфейсу користувача;
- проектування додатка з використанням TOE-таблиці;
- написання програмного коду;
- тестування і налагодження програми;
- збереження проекту і компіляція (перетворення проекту на додаток, який здатний самостійно працювати за межами середовища програмування).

★ ★ ★

**Скрипка А.В. Технологія проектування електронного приложення к уроку математики в среде визуального программирования**

**Аннотация.** В статье исследуется актуальная проблема технологии проектирования электронного приложения к уроку математики в среде визуального программирования.

**Ключевые слова:** ИКТ-компетентность, этапы создания приложения, среда визуального программирования, разработка программного средства, Visual Studio.

### Список використаних джерел

1. ICT competency standards for teachers: competency standards modules, [Електронний ресурс]. <http://unesdoc.unesco.org/images/0015/001562/156207e.pdf>. — Заголовок з екрану.
2. Домашня сторінка курсу цифрових технологій [Електронний ресурс]. <http://www.microsoft.com/about/corporatecitizenship/citizenship/giving/programs/up/digitalliteracy/ukr/default.mspx> — Заголовок з екрану.
3. Домашня сторінка програми Партнерство у навчанні [Електронний ресурс]. <http://ua.partnersinlearningnetwork.com/Pages/default.aspx> — Заголовок з екрану.