

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
«ХАРКІВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»

МЕТОДИЧНІ ВКАЗІВКИ

до виконання лабораторної роботи

Розробка діаграми класів у середовищі

Umbrello UML Modeller

з курсу “Основи архітектури програмних систем”
 (“Fundamentals of program systems architecture”)

Хацько Н.Є., Гавриленко С.Ю.

Харків НТУ «ХПІ»

2019

УДК 004.2; 004.6; 519.6

Рецензенти:

О.С. Назаров, к.т.н., доцент, заступник декана факультету комп'ютерних наук Харківського національного університету радіоелектроніки.

Методичні вказівки до виконання лабораторної роботи “Розробка діаграми класів у середовищі Umbrello UML Modeller” : для студентів спеціальності 121 – Інформаційні технології. Англійською та українською мовами. / Уклад. Хацько Н.Є., Гавриленко С.Ю. – Харків : НТУ «ХПІ», 2019. – 39 с.

Укладачі: Хацько Н.Є., Гавриленко С.Ю.

Рецензент Назаров О.С.

Кафедра програмної інженерії та інформаційних технологій управління, факультет комп'ютерних наук та програмної інженерії.

УДК 004.2; 004.6; 519.6

© Хацько Н.Є., Гавриленко С.Ю. 2019

© НТУ «ХПІ», 2019

TABLE OF CONTENTS

Introduction.....	3
1 Creation of class diagram	4
2 Adding a class to a class diagram and editing its properties	6
3 Development of the diagram of classes of the atm model.....	13
4 Adding and editing class attributes.....	16
5 Adding and editing class operations	18
6 Attribute and transaction specification for the ATM Transaction class.....	21
7 Adding an association to a class diagram and editing its properties	23
8 Adding aggregation and composition relationships to a class diagram and editing their properties	27
9 Adding a generalization relationship to a class diagram and editing its properties	28
10 Final construction of the ATM model class diagram	31
Conclusion	38
Bibliography	38

INTRODUCTION

The class diagram is the most directly applied when it comes to coding. Most UML tools will generate a set of classes in many programming languages straight from the class diagram. They are not always complete; especially where collections or templates are concerned, but they can save a lot of repetitive typing.

Class diagrams are also the most complex, and certainly the most discussed when modelling object oriented concepts such as composition and aggregation. Hopefully, we can avoid such arguments for now and accept that class diagrams are able to model many object oriented, and even entity relationship particulars.

Діаграма класів найбільш безпосередньо застосовується, коли мова йде про кодування. Більшість інструментів UML генерують набір класів у багатьох мовах програмування прямо з діаграми класів. Вони не завжди повноцінні; особливо це стосується колекцій чи шаблонів, але вони можуть зекономити багато повторюваних текстів.

Діаграми класів є також найскладнішими і, безумовно, найбільш обговорюваними при моделюванні об'єктно-орієнтованих понять, таких як склад та агрегація. Сподіваємось, ми можемо поки що уникнути подібних аргументів і прийняти, що діаграми класу здатні моделювати багато деталей, орієнтованих на об'єкти, і навіть особливості відношення сутності.

1 CREATION OF CLASS DIAGRAM

(СТВОРЕННЯ ДІАГРАМИ КЛАСІВ)

The class diagram is the main logical representation of the model and contains detailed information about the internal structure of object-oriented software system or using modern terminology, about the architecture of the software system. There are several ways to activate the class diagram window:

- the class diagram window appears by default in the diagram window after creating a new project;
- open the Logical View in the project browser and select the class diagram element;
- select the main menu item Diagram-> New-> class diagram.

At the same time, a new window appears with a clean worksheet of the class diagram and a special toolbar. It contains buttons with the image of the graphic primitives. These primitives need to develop the class diagram. The purpose of the individual panel buttons can also be found in tooltips.

Діаграма класів є основним логічним поданням моделі і містить детальну інформацію про внутрішню структуру об'єктно-орієнтованої програмної системи або з використанням сучасної термінології, про архітектуру програмної системи. Існує кілька способів активації вікна діаграми класів:

- вікно діаграми класів з'являється за замовчуванням у вікні діаграми після створення нового проекту;
- відкрити логічний вигляд у браузері проекту та вибрати елемент діаграми класів;
- виберіть пункт головного меню Діаграма-> Створити-> Діаграма класу.

Одночасно з'являється нове вікно з чистим робочим аркушем діаграми класів та спеціальною панеллю інструментів. Він містить кнопки із зображенням графічних примітивів. Цим примітивам потрібно розробити схему класів. Призначення окремих кнопок панелі також можна знайти в підказках.

Table 1 - Button assignment of a special class diagram toolbar

(Призначення кнопок спеціальної панелі інструментів діаграми класів)

Графическое изображение	Всплывающая подсказка	Назначение кнопки
	Select	Перетворює зображення курсору у стрілку для наступного виведення елементів на діаграмі
	Note	Додає на діаграму примітку
	Anchor	Додає зв'язок примітки з відповідними графічними елементами діаграм
	Label	Додає на діаграму текстову область
	Box	Додає на діаграму коробку (пакет)
	Class	Додає на діаграму клас
	Interface	Додає на діаграму інтерфейс
	Datatype	Додає на діаграму тип даних
	Enum	Додіє на діаграму перелік
	Package	Додає на діаграму пакет
	Association	Додає на діаграму асоціацію
	Directional Association	Додає на діаграму направлену асоціацію
	Dependency	Додіє на діаграму відношення залежності
	Implements	Додіє на діаграму відношення приналежності
	Composition	Додіє на діаграму відношення композиції
	Aggregation	Додіє на діаграму відношення агрегації
	Containment	Додіє на діаграму відношення «міститься»

2 ADDING A CLASS TO A CLASS DIAGRAM AND EDITING ITS PROPERTIES

(ДОДАВАННЯ КЛАСА НА ДІАГРАМУ КЛАСІВ И РЕДАГУВАННЯ ЙОГО ВЛАСТИВОСТЕЙ)

To add a class to class diagram, use left mouse button to click class icon in special toolbar, release left mouse button and left-click on an empty spot on the chart workspace. A class image with markers for changing its geometric dimensions and the default name NewClass proposed by the environment will appear on the diagram. Continuing the development of the ATM model as end-to-end example of the project, we construct the following canonical diagram for this model – the class diagram. With this purpose, the default name of the class diagram should be changed to “ATM Class Diagram” and name of the class added to the diagram should be changed to “ATM Transaction”.

Для додавання класу на діаграму класів потрібно за допомогою лівої кнопки миші натиснути кнопку із зображенням піктограми класу на спеціальній панелі інструментів, відпустити ліву кнопку миші і клацнути лівою кнопкою миші на вільному місці робочого листа діаграми. На діаграмі з'явиться зображення класу з маркерами зміни його геометричних розмірів і запропонованим середовищем ім'ям за умовчанням NewClass. Продовжуючи розробку моделі банкомату в якості наскрізного прикладу проекту, побудуємо для цієї моделі наступну канонічну діаграму - діаграму класів. З цією метою слід змінити запропоноване за замовчуванням ім'я діаграми «class diagram» на «Діаграма класів АТМ», а ім'я доданого на діаграму класу - на «Транзакція Банкомату».

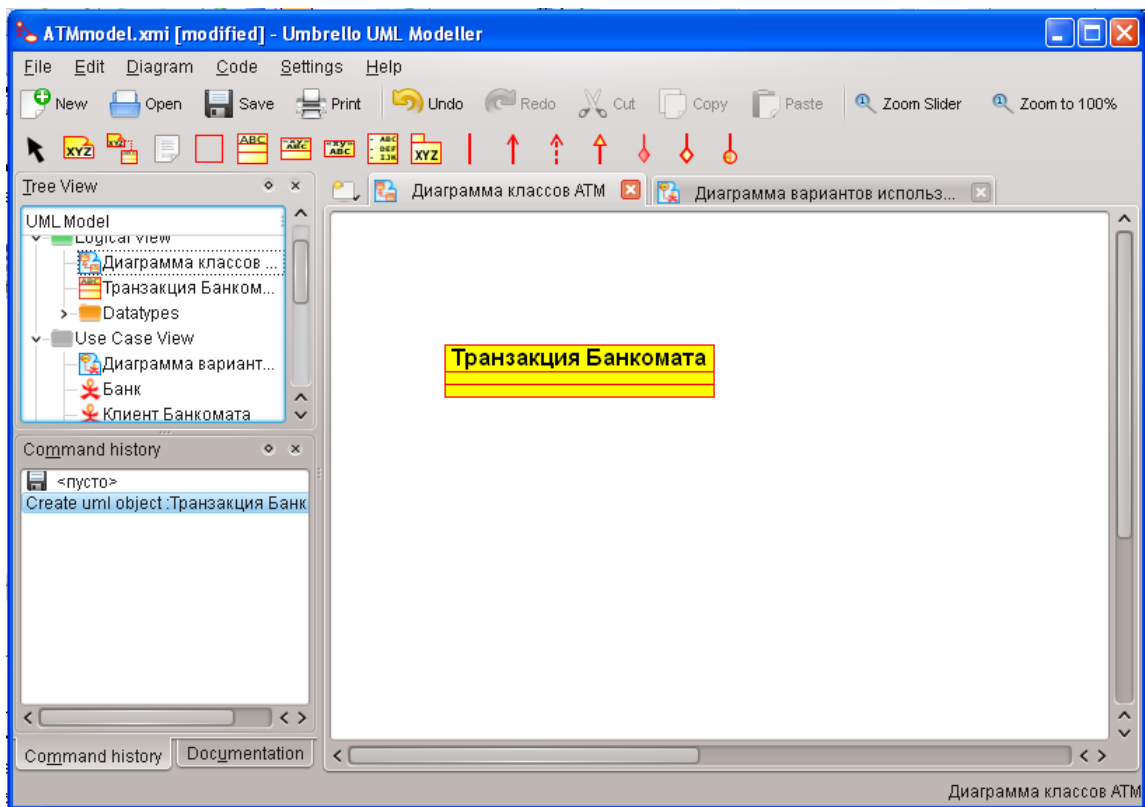


Figure 1 - Class diagram of an ATM model after adding the “ATM Transaction” class to it

As the project is completing and the model is implemented in some programming language, the names of the corresponding classes, attributes and operations must writer according to the rules of the corresponding language and without spaces. In the context of a model-driven architecture, the first model is also called platform independent, and the second model is platform-dependent.

As the project is completed and the model is implemented in some programming language, the names of the corresponding classes, attributes and operations must be written according to the rules of the respective language and without spaces. In the context of model-driven architecture, the first model is also

По мірі просування проекту та реалізації моделі деякою мовою програмування назви відповідних класів, атрибутів та операцій повинні бути записані згідно з правилами відповідної мови та без пробілів. У контексті архітектурної моделі, перша модель також називається незалежною платформою, а друга модель - залежною від платформи.

Для класу «Транзакція Банкомату» можна уточнити його призначення в моделі за допомогою вказівки стереотипу і пояснювального тексту у вигляді документації. З цією метою слід натисканням правої кнопки миші на зображенні цього класу на діаграмі відкрити діалогове вікно специфікації

called platform independent and the second model is platform independent.

For the “ATM Transaction” class, you can clarify its purpose in the model by indicating the stereotype and explanatory text in the documentation form. For this purpose, right-click on the image of this class in the diagram to open the dialog box for specifying the properties of this class (Fig. 2) and specify the «entity» stereotype in the "Stereotype name" category.

властивостей цього класу (рис. 2) і на вкладці General вказати в категорії «Stereotype name» стереотип «entity» (сутність).

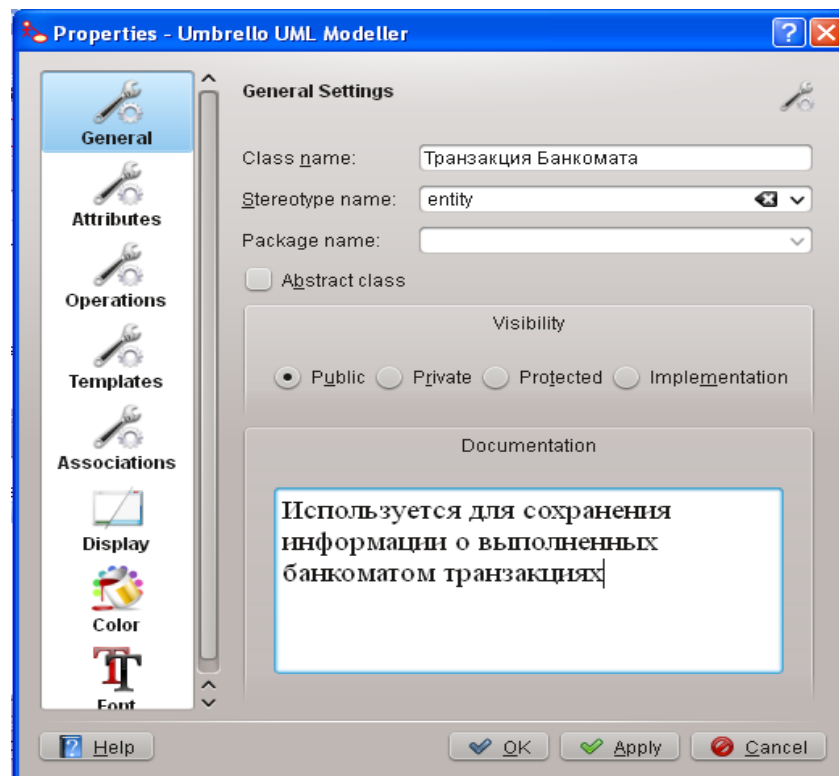


Figure 2 - Class Properties Dialog Box

The choice of this stereotype means that the corresponding class is designed to store information that must be stored in the system after the destruction of objects of this class. Then, in the documentation section of this class, you can enter the

Вибір даного стереотипу означає, що відповідний клас призначений для зберігання інформації, яка повинна зберігатися в системі після знищення об'єктів даного класу. Далі в секцію документації даного класу можна

explanatory text: "Used to save information about transactions performed by the ATM" and click the Apply or OK button to save the results of editing the properties of the selected class. After the stereotype is assigned to the ATM Transaction class, the text of this stereotype appears in angle brackets above the name of this class (Figure 3).

ввести пояснювальний текст: "Використовується для збереження інформації про виконані банкоматом транзакціях" і натиснути кнопку Apply або ОК, щоб зберегти результати редагування властивостей вибраного класу. Після призначення стереотипу класу Транзакція банкомату текст даного стереотипу в кутових дужках з'явиться вище імені даного класу (рисунок 3).

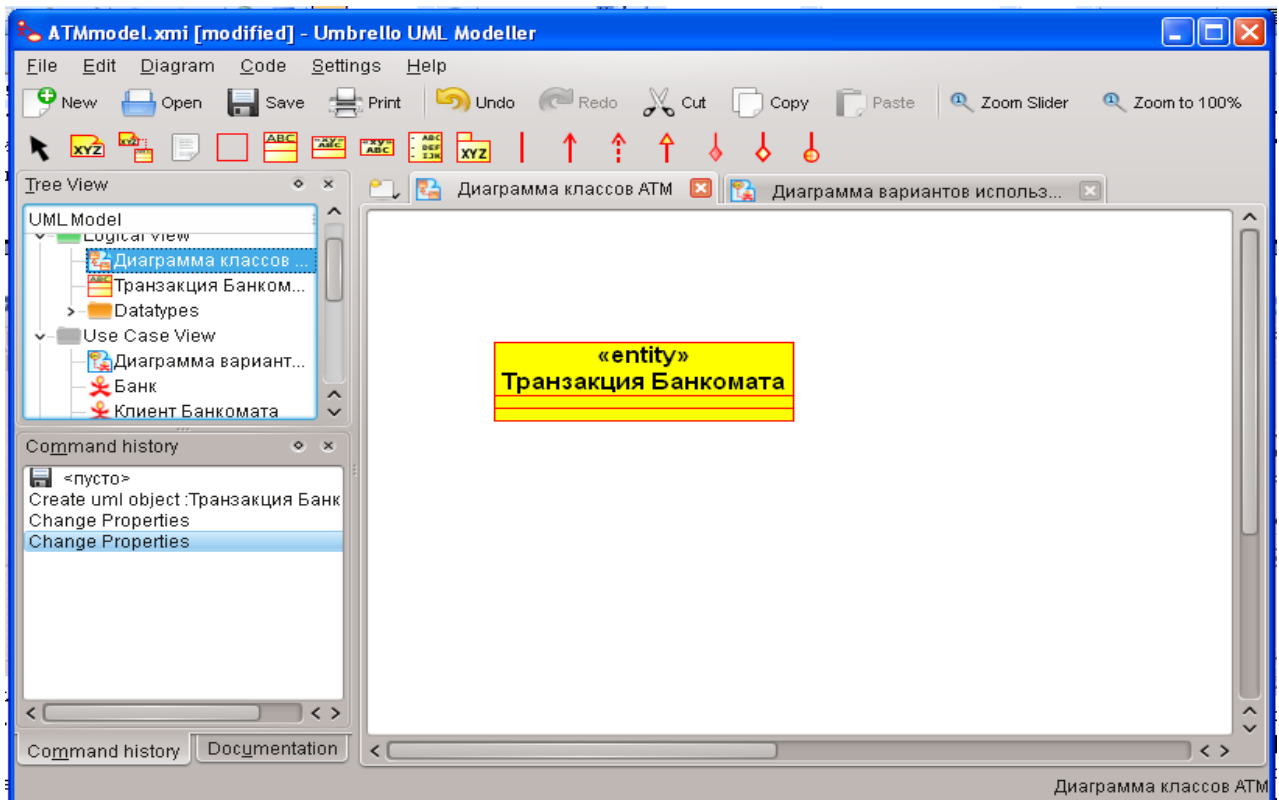


Figure 3 - Diagram of the ATM model classes after selecting a stereotype for the ATM Transaction class

For a particular class, you can also clarify its other properties, available for editing on the Attributes, Operations, Templates, Associations tabs of the properties specification window for this class. For example, on the Attributes tab,

Для окремого класу можна уточнити також і інші його властивості, доступні для редагування на вкладках Attributes, Operations, Templates, Associations вікна специфікації властивостей цього класу. Наприклад, на вкладці Attributes

you can create a new attribute (field) for the class in question, which will be of type (bool, char, double, float, int, long, short, string, unsigned int, unsigned long and unsigned short), name, initial value, stereotype name, the ability to specify the static qualifier. By default, the visibility of the new attribute (field) of the class will have a private access specifier.

можна створити новий атрибут (поле) для розглянутого класу, який буде мати тип (bool, char, double, float, int, long, short, string, unsigned int, unsigned long і unsigned short), ім'я, початкове значення, ім'я стереотипу, можливість вказівки специфікатор static. За замовчуванням видимість нового атрибута (поля) класу матиме специфікатор доступу «private».

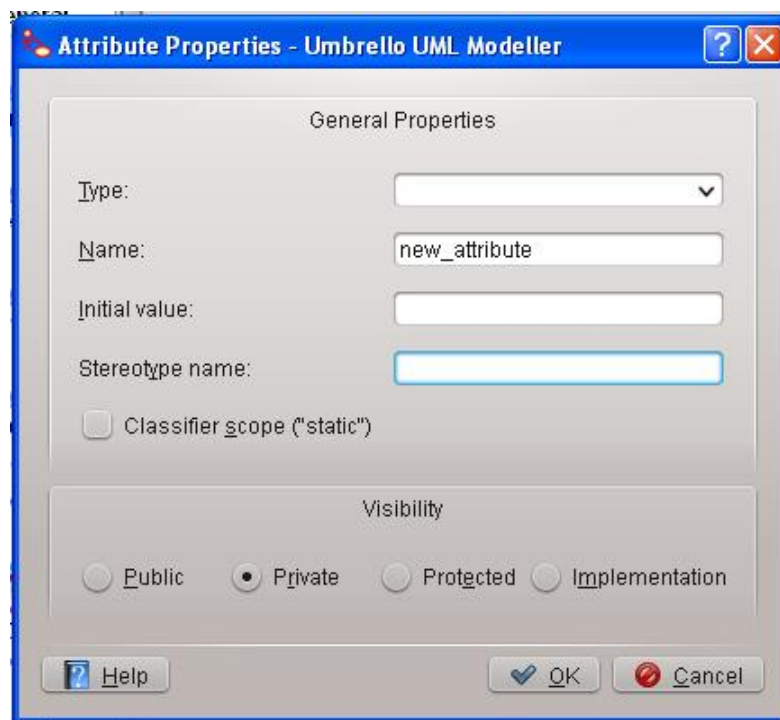


Figure 4 - Adding a new class attribute

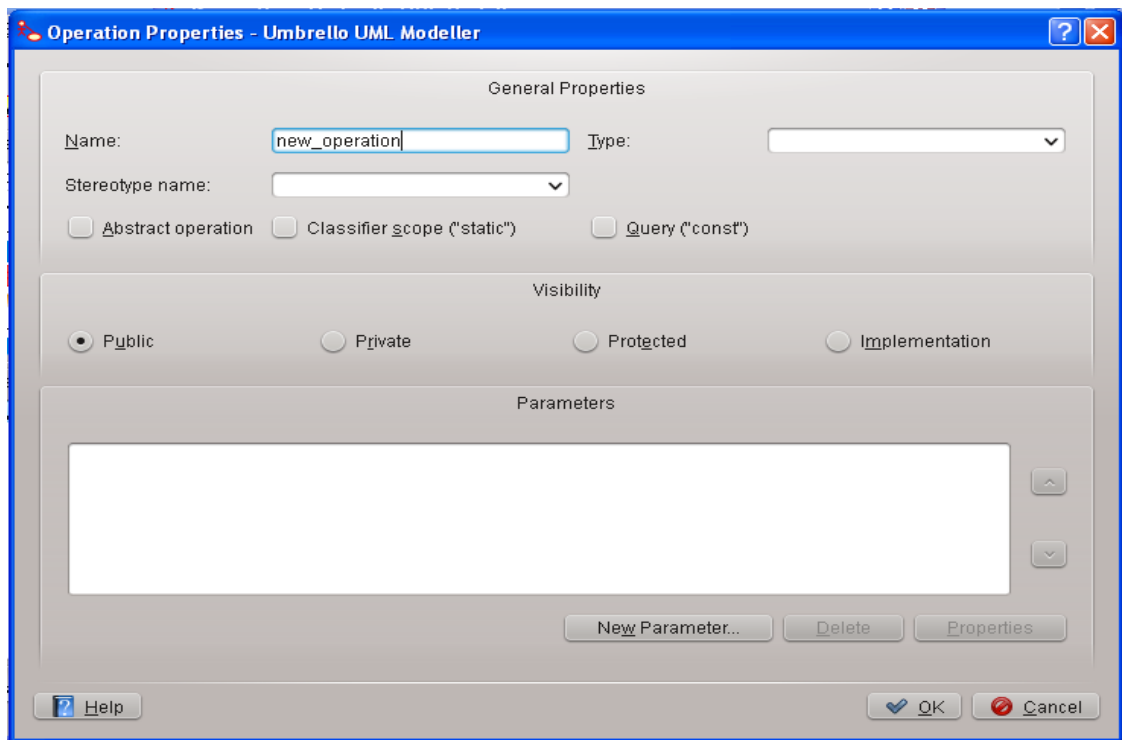


Figure 5 - Adding a new class method

You can also create a new function for the class - the class method, using the Operations tab of the class properties dialog box. At the same time, the name, stereotype name, type of the return value and a list of parameters (see Figure 5) are indicated; const, static and the affiliation of the operation to abstract (virtual) can be specified. By default, the visibility of the new class method will have a public access specifier.

On the Templates tab of the class properties dialog box, you can specify the parameters of the template added to the class description (Figure 6). Associations tab sets associations of this class with other chart elements. The Display, Color, and Font tabs allow you to control the display of the class, for example, on the Display tab, you can specify whether

Також для класу можна створити нову функцію - метод класу, використовуючи вкладку Operations діалогового вікна властивостей класу. При цьому вказується ім'я, ім'я стереотипу, тип значення і список параметрів (див. рис. 5) можна вказати специфікатор const, static і приналежність операції до абстрактних (віртуальним). За замовчуванням видимість нового методу класу матиме специфікатор доступу «public».

На вкладці Templates діалогового вікна властивостей класу можна вказати параметри доданого в опис класу шаблону (рис. 6). На вкладці Associations задаються асоціації даного класу з іншими елементами діаграми. Виберіть Display, Color і Font дозволяють керувати відображенням класу, наприклад на вкладці Display

certain class elements (fields, methods, templates, stereotypes, etc.) will be displayed on the field of the class diagram. "

можна вказати чи будуть певні елементи класу (поля, методи, шаблони, стереотипи і інше) відображатися на поле діаграми класів.

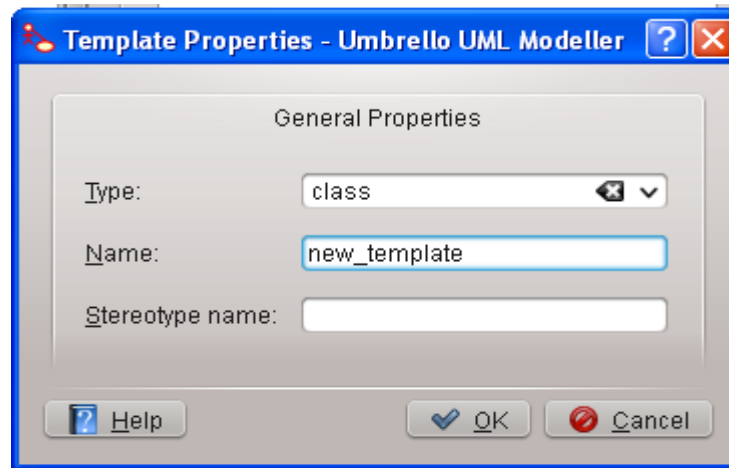


Figure 6 - Adding a template

In order to specify the class as abstract, i.e. having no instances, you should put a mark in the Abstract property on the same tab. In relation to the model under consideration for the class "ATM Transaction", the mark for the Abstract property should be left blank. It should be noted that in order to prevent loss of information about the developed model and the results of editing the properties of its graphic elements, it is necessary to periodically save the model in an external file. To do this, perform the main menu operation: File-> Save (File -> Save) or press the key combination: Ctrl + S. For the same purpose, the corresponding button on the standard toolbar serves.

Для того, щоб специфікувати клас як абстрактний, тобто не має примірників, слід на цієї ж вкладці виставити позначку в властивості Abstract. Що стосується аналізованої моделі для класу «Транзакція банкомату» слід позначку для властивості Abstract залишити порожньою. Слід зауважити, що для запобігання втрати інформації щодо розробляємої моделі і результатів редагування властивостей її графічних елементів потрібно часто зберігати модель в зовнішньому файлі. Для цього слід виконати операцію головного меню: File-> Save або натиснути комбінацію клавіш: Ctrl + S. Для цієї ж мети служить відповідна кнопка на стандартній панелі інструментів.

3 DEVELOPMENT OF THE DIAGRAM OF CLASSES OF THE ATM MODEL

(РАЗРАБОТКА ДИАГРАММЫ КЛАССОВ МОДЕЛИ БАНКОМАТА)

On the class diagram under development, a textual method of depicting class stereotypes was selected, in which the stereotype is written in angle quotes above the name of the corresponding class.

You can select all graphic elements on the class diagram, as well as on any other diagram of the model, using the main menu operation: Edit-> Select All (or use the keyboard shortcut Ctrl + A). Continuing the development of the ATM model, we add the second class to the diagram with the name "ATM Controller", for which we select the control stereotype in the property specification window and enter the text as the documentation: "Implements the logic of ATM functioning". In this case, the attributes and operations of this class will be absent. The corresponding fragment of the class diagram after adding the control class, the ATM Controller will have the following form (Figure 7).

На діаграмі класів, що розробляється, обрано текстовий спосіб зображення стереотипів класів, при якому стереотип записується в кутових лапках вище імені відповідного класу.

Виділити всі графічні елементи на діаграмі класів, також як і на будь-якій іншій діаграмі моделі, можна за допомогою виконання операції головного меню: Edit-> Select All або за допомогою комбінації клавіш Ctrl + A. Продовжуючи розробку моделі банкомату, додамо на діаграму другий клас з ім'ям «Контролер Банкомату», для якого у вікні специфікації властивостей виберемо стереотип control (керуючий клас), а в якості документації введемо текст: «Реалізує логіку функціонування банкомату». При цьому атрибути і операції у даного класу будуть відсутні. Відповідний фрагмент діаграми класів після додавання керуючого класу Контролер Банкомату матиме такий вигляд (рис. 7).

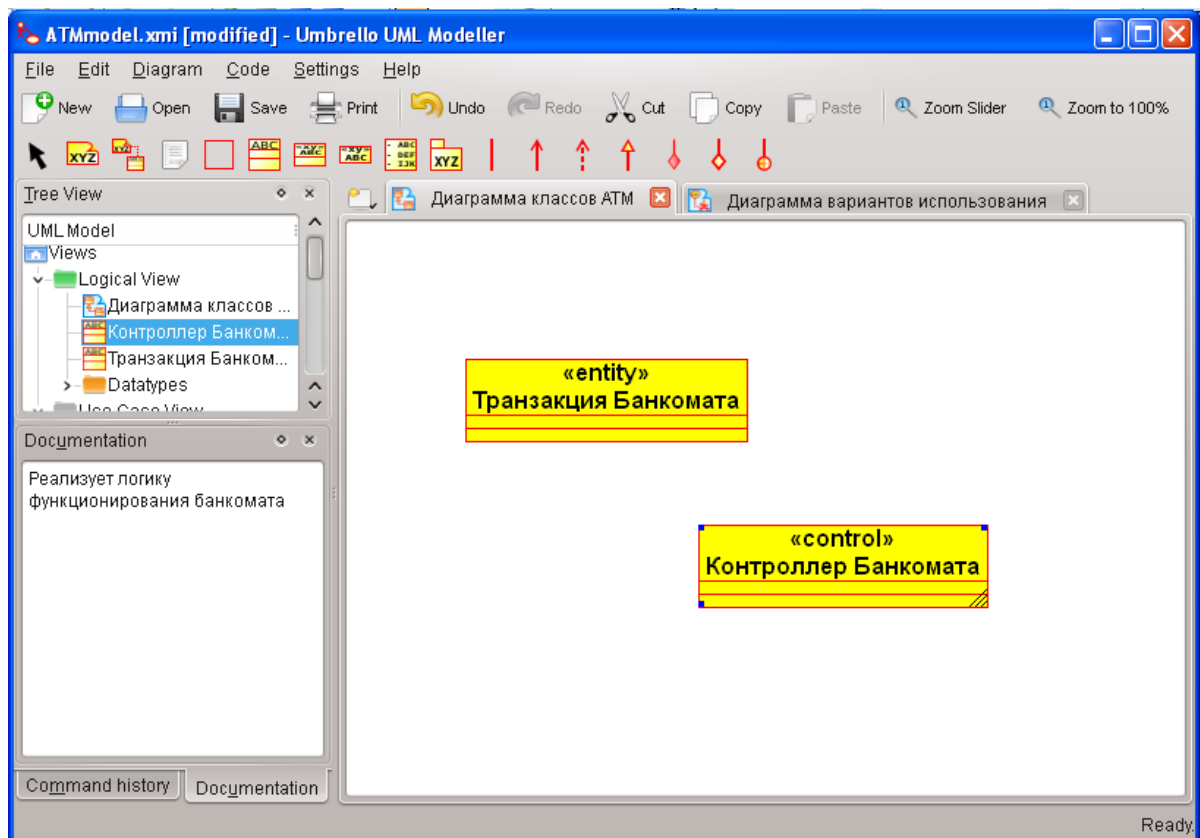


Figure 7 - Class diagram after adding the “ATM Controller” class

Continuing the development of the ATM model, we add a third class to the diagram with the name “Card Reader”, for which we will select the “boundary” stereotype (boundary class) in the properties specification window. The use of this stereotype means that this class is on the border of the simulated system, which is considered the model of an ATM. After that, an explanatory text can be entered into the documentation section of this class: “Installed at an ATM”. Next, add an interface with the name “Bank Controller”, for it the stereotype “Interface” will be set by default, which means that the ATM uses the Bank's services when processing its transactions.

Продовжуючи розробку моделі банкомату, додамо на діаграму третій клас з ім'ям «Пристрій читання картки», для якого у вікні специфікації властивостей виберемо стереотип «boundary» (граничний клас). Застосування цього стереотипу означає, що даний клас знаходиться на кордоні модельованої системи, в якості якої розглядається модель банкомату. Після цього в секцію документації даного класу можна ввести пояснювальний текст: «Встановлюється на банкоматі». Далі слід додати інтерфейс з ім'ям «Контроллер Банку», для нього стереотип «Interface» буде встановлено

Note that the first letter in the name of this class is the English "I", which serves in the UML language to indicate the interface. The corresponding fragment of the class diagram after adding the classes "Card Reader" and "I Bank Controller" to it will have the following form (Figure 8).

за умовчанням, що означає, що банкомат користується послугами Банку при обробці своїх транзакцій. Зауважимо, що першою літерою в імені цього класу є англійське «I», яке служить в мові UML для вказівки інтерфейсу. Відповідний фрагмент діаграми класів після додавання на неї класів «Пристрій читання картки» та «ІКонтроллерБанка» матиме такий вигляд (рис. 8).

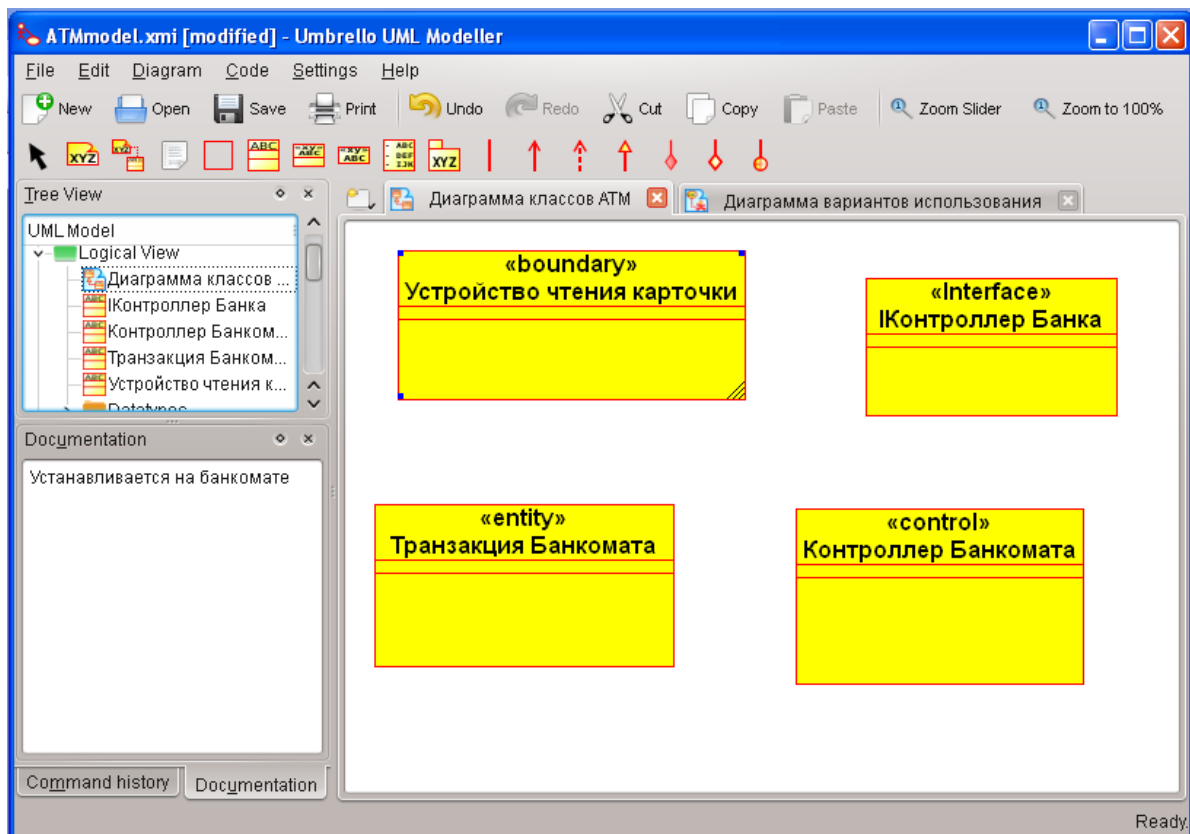


Figure 8 - ATM model classes

4 ADDING AND EDITING CLASS ATTRIBUTES (ДОДАВАННЯ ТА РЕДАГУВАННЯ АТРИБУТІВ КЛАСА)

Of all the graphical elements of the Umbrello UML Modeller environment, the class has the maximum set of properties, the main of which are its attributes and operations. Since it is the class diagram that is used in the Umbrello UML Modeller environment to generate program code, we will consider in detail the corresponding properties of attributes and operations.

You can add an attribute to a previously created class in one of the following ways:

- using the tab `c` of the Properties context menu for the class highlighted in the class diagram.
- using the context menu operation `New-> Attribute` for the class selected in the project browser (Tree View). In this case, the text input cursor is activated in the area of the hierarchical representation of the class in the project browser under the name of the corresponding class.

After adding an attribute to the class, by default it is given the name `new_attribute` and some visibility quantifier (Fig. 9).

For the considered ATM model, the name of the added attribute should be changed to “card ID”. Class attribute names and operations must begin with a lowercase letter.

To edit attribute properties, use the special `Attribute Properties` attribute

З усіх графічних елементів середовища Umbrello UML Modeller клас володіє максимальним набором властивостей, головними з яких є його атрибути і операції. Оскільки саме діаграма класів використовується в середовищі Umbrello UML Modeller для генерації програмного коду, докладно розглянемо відповідні властивості атрибутів і операцій.

Додати атрибут до створеного раніше класу можна одним з таких способів:

- `c` допомогою вкладки `c` контекстного меню `Properties` для класу, виділеного на діаграмі класів.
- `c` допомогою операції контекстного меню `New-> Attribute` для класу, виділеного в браузері проекту (Tree View). В цьому випадку активізується курсор введення тексту в області ієрархічного уявлення класу в браузері проекту під ім'ям відповідного класу.

Після додавання атрибута до класу за замовчуванням йому присвоюється ім'я `new_attribute` і деякий квантор видимості (рис. 9).

Для даної моделі банкомата ім'я доданого атрибута слід змінити на «ідентифікатор картки». Імена атрибутів і операцій класів повинні починатися з малої літери.

Для редагування властивостей атрибутів призначене спеціальне

specification dialog box, which is opened by clicking on the line of the selected attribute in the class properties specification window. In the properties window of an individual class attribute, you can specify the attribute data type and its initial value, as well as assign a stereotype to the attribute from the drop-down list or change its visibility quantifier.

For the “card identifier” attribute, select the type Int (integer) as the type of its valid values from the Type nested list, and to set the visibility quantifier, select the Public quantifier in the Visibility group. Since the initial value for this attribute is not defined, the corresponding input field should be left blank. An explanatory text can be entered into the documentation section of this class attribute: “The card reader reads the value of this attribute from the customer’s credit card”. To save the results of editing these attribute properties, click the Apply or OK button. The corresponding attribute property specification window is shown in Figure 9.

діалогове вікно специфікації атрибута Attribute Properties, яке відкривається натисканням миші на рядку обраного атрибута у вікні специфікації властивостей класу. У вікні властивостей окремого атрибута класу можна задати тип даних атрибута і його початкове значення, а також призначити атрибуту стереотип із списку або змінити його квантор видимості.

Для атрибута «ідентифікатор картки» в якості типу його допустимих значень з вкладеного списку Type слід вибрати тип Int (цілочисельний), а для завдання квантора видимості слід вибрати в групі Visibility квантор Public. Оскільки початкове значення для даного атрибута не визначено, відповідне поле введення слід залишити порожнім. У секцію документації даного атрибута класу можна ввести пояснювальний текст: «Пристрій читання картки зчитує значення цього атрибута з кредитної картки клієнта». Щоб зберегти результати редагування цих властивостей атрибута, потрібно натиснути кнопку Apply або OK. Відповідне вікно специфікації властивостей атрибута наведено на рис. 9.

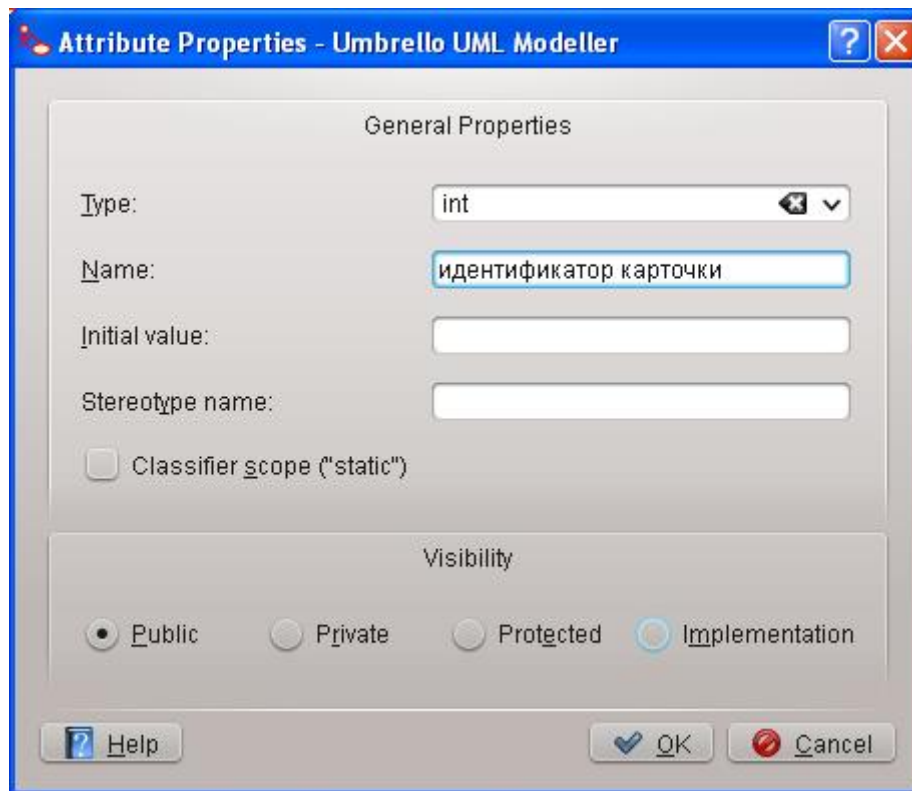


Figure 9 - Setting attribute properties

You can define the attribute as static by checking the Static selection line. A static attribute by definition has the same value for all objects of the class in question.

Можна визначити атрибут як статичний, виставивши позначку в рядку вибору Static. Статичний атрибут по визначенню має одне і теж значення для всіх об'єктів даного класу.

5 ADDING AND EDITING CLASS OPERATIONS. (ДОДАВАННЯ ТА РЕДАГУВАННЯ ОПЕРАЦІЙ КЛАСІВ)

The functioning of an ATM is based on the performance of certain actions by its individual devices. In an ATM structure model, all actions are represented using class operations. Thus, the next step in developing a class diagram is related to the specification of class operations.

Функціонування банкомату засноване на виконанні окремими його пристроями тих чи інших дій. У моделі структури банкомату всі дії реалізуються за допомогою операцій класів. Таким чином, наступний етап розробки діаграми класів пов'язаний зі специфікацією операцій класів.

You can add an operation to a previously created class in one of the following ways:

- using the context menu operation New-> Operation for the class highlighted in the class diagram. In this case, the Operation Properties dialog box opens.

- using the context menu operation New-> Operation for the class selected in the project browser. In this case, the input cursor is activated in the area of the hierarchical representation of the class in the browser under the name of the corresponding class.

After adding an operation to the class, by default it is given the name `new_operation` and the visibility quantifier `public`.

In the context of the ATM model under consideration, the name of the first operation for the class "ATM Transaction" should be set to: "create a new transaction." In this case, the brackets are not written when specifying the name of the operation, since the Umbrello UML Modeller program automatically adds them. However, following the rules for naming operations in the UML language, operation names will be indicated with brackets in the lecture text.

Each of the class operations has its own dialog box for the specification of Operation Properties properties, which can be opened by double-clicking on the name of the operation on the corresponding tab of the class specification or on the name of this

Додати операцію до створеного раніше класу можна одним з таких способів:

- с допомогою операції контекстного меню New-> Operation (Нова операція) для класу, виділеного на діаграмі класів. У цьому випадку відкривається діалогове вікно Operation Properties.

- с допомогою операції контекстного меню New-> Operation для класу, виділеного в браузері проекту. В цьому випадку активізується курсор введення в області ієрархічного уявлення класу в браузері під ім'ям відповідного класу.

Після додавання операції до класу за замовчуванням їй присвоюється ім'я `new_operation` і квантор видимості `public`.

В контексті даної моделі банкомата в якості імені першої операції для класу «Транзакція Банкомату» слід задати: «створити нову транзакцію». При цьому дужки при завданні імені операції не заносяться, оскільки програма Umbrello UML Modeller додає їх автоматично. Однак, дотримуючись правил іменування операцій в мові UML, в тексті лекції імена операцій будуть вказуватися з дужками.

Кожна з операцій класів має власне діалогове вікно специфікації властивостей Operation Properties, яке може бути відкрито за подвійним клацанням на імені операції на відповідній вкладці специфікації класу або на імені цієї операції в браузері

operation in the project browser. For the operation "create a new transaction ()" as the visibility quantifier, select the public quantifier from the nested list. You can enter the explanatory text in the documentation section of this class operation: "It is called after the credit card is inserted into the Card Reader" and click the Apply or OK button to save the results of editing the properties of this operation.

For class operations, in addition to the visibility quantifier, you can also specify: arguments and their type (Parameters-> New Parameter), return type (Type), operation stereotype (Stereotype name), and also indicate the abstractness of the operation, constancy and static.

проекту. Для операції «створити нову транзакцію ()» в якості квантора видимості слід вибрати з вкладеного списку квантор public. У секцію документації даної операції класу можна ввести пояснювальний текст: «Викликається після того, як кредитна картка вставлена в пристрій читання картки» і натиснути кнопку Apply або OK, щоб зберегти результати редагування властивостей цієї операції.

Для операцій класів крім квантора видимості можна також задати: аргументи і їх тип (Parameters-> New Parameter), тип повертаємого результату (Type), стереотип операції (Stereotype name), а також вказати абстрактність операції, константність і статичність.

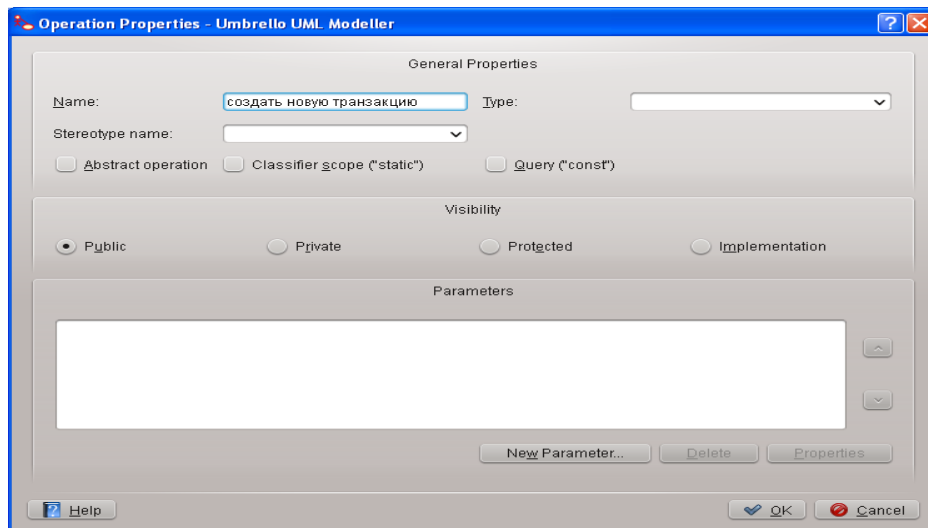


Figure 10 - Setting class operation properties

6 ATTRIBUTE AND TRANSACTION SPECIFICATION FOR THE ATM TRANSACTION CLASS

(СПЕЦИФІКАЦІЯ АТРИБУТІВ І ОПЕРАЦІЙ ДЛЯ КЛАСУ «ТРАНЗАКЦІЯ БАНКОМАТУ»)

To complete the specification of the ATM Transaction class in the same way, add 3 more attributes and 2 operations with the following properties:

- “value of the PIN code of the card” with the visibility quantifier public. As the type of this attribute, select the type Integer (integer), and enter the explanatory text in the attribute documentation section: "The card reader reads the value of this attribute from the client's credit card."

- "entered PIN code" with the visibility quantifier public. As the type of this attribute, select the type Integer (integer), and enter the explanatory text in the attribute documentation section: "The value of this attribute is entered by the client from the ATM keyboard."

- "entered cash amount" with the visibility quantifier public. The type of float should be selected as the type of this attribute, and the explanatory text should be entered in the attribute documentation section: "The value of this attribute is entered by the client from the ATM keyboard."

- “check the correctness of the PIN code ()” with the visibility quantifier public. As the return type for this operation, you should select the bool type (logical), and enter the explanatory text in

Щоб закінчити специфікацію класу Транзакція Банкомату аналогічним способом слід додати ще 3 атрибута і 2 операції з наступними властивостями:

- «значення ПІН-коду картки» з квантором видимості public. В якості типу цього атрибута слід вибрати тип Integer (цілочисельний), а в секцію документації атрибута ввести пояснювальний текст: «Пристрій читання картки зчитує значення цього атрибута з кредитної картки клієнта».

- «введений ПІН-код» з квантором видимості public. В якості типу цього атрибута слід вибрати тип Integer (цілочисельний), а в секцію документації атрибута ввести пояснювальний текст: «Значення цього атрибута вводиться клієнтом з клавіатури банкомату».

- «введена сума готівки» з квантором видимості public. В якості типу цього атрибута слід вибрати тип float, а в секцію документації атрибута ввести пояснювальний текст: «Значення цього атрибута вводиться клієнтом з клавіатури банкомату».

- «перевірити правильність ПІН-коду ()» з квантором видимості public. В якості типу повертається результату для цієї операції слід вибрати тип bool (логічний), а в секцію її документації

the section of its documentation: “Called after the client has entered the PIN code value from the ATM keyboard”.

- “complete transaction ()” with the visibility quantifier public. Enter the explanatory text in the section of its documentation: “Called after completion of all actions of the ATM to serve the customer.”

The corresponding fragment of the class diagram after adding and specifying attributes and operations for the class “ATM Transaction” will have the following form (Fig. 11).

ввести пояснювальний текст: «Викликається після того, як клієнт ввів значення ПІН-коду з клавіатури банкомату».

- «завершити транзакцію ()» з квантором видимості public. У секцію її документації ввести пояснювальний текст: «Викликається після завершення всіх дій банкомату по обслуговуванню клієнта».

Відповідний фрагмент діаграми класів після додавання і специфікації атрибутів і операцій для класу «Транзакція Банкомату» матиме такий вигляд (рис. 11).

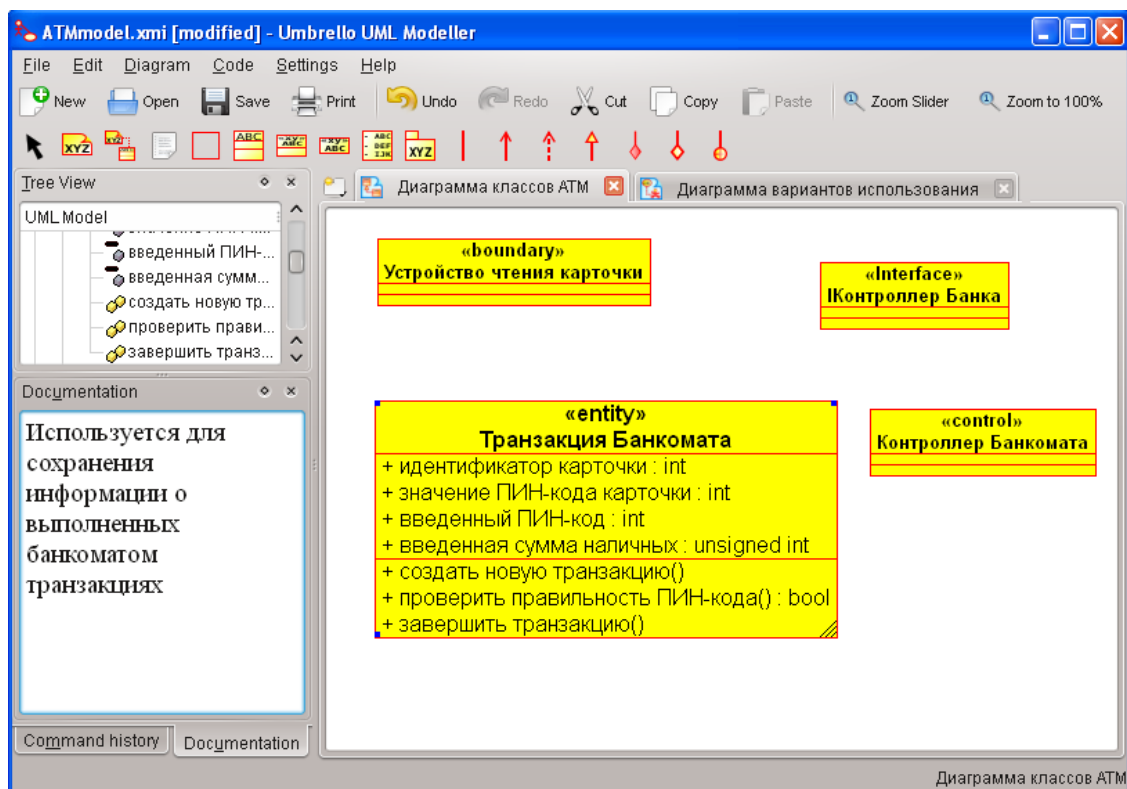


Figure 11 - A fragment of the class diagram of the ATM model after adding attributes and operations for the class "ATM transaction"

7 ADDING AN ASSOCIATION TO A CLASS DIAGRAM AND EDITING ITS PROPERTIES

(ДОДАВАННЯ АСОЦІАЦІЇ НА ДІАГРАМУ КЛАСІВ ТА РЕДАГУВАННЯ ЇЇ ВЛАСТИВОСТЕЙ)

The class diagram is a logical representation of the structure of the model, so it should contain as many classes as necessary for the implementation of the entire project. In this case, for a complete presentation of the structure of the model, it is necessary to establish and specify the relations between classes.

On a special toolbar, you must click the button with the image of the directional association icon and release the left mouse button. If the association is directed, then on the class diagram it is necessary to select the first association element or source from which the arrow is emanating, and having released the left mouse button, move its pointer to the second relation element or receiver to which the arrow is directed. After moving to the second element, the mouse button should be pressed again, as a result of which a directional association between the two selected classes will be added to the class diagram.

Continuing the development of the class diagram of the ATM model, we add to it in the described way the directional association between the “ATM

Діаграма класів є логічним поданням структури моделі, тому вона повинна містити стільки класів, скільки необхідно для реалізації всього проекту. При цьому для повного уявлення структури моделі необхідно встановити і уточнити відносини між класами.

На спеціальній панелі інструментів необхідно натиснути кнопку із зображенням піктограми спрямованої асоціації і відпустити ліву кнопку миші. Якщо асоціація спрямована, то на діаграмі класів треба виділити перший елемент асоціації або джерело, від якого виходить стрілка, і, відпустивши натиснуту ліву кнопку миші, перемістити її покажчик до другого елемента відносини або приймача, до якого спрямована стрілка. Після переміщення до другого елемента кнопку миші слід ще раз натиснути, в результаті чого на діаграму класів буде додана спрямована асоціація між двома вибраними класами.

Продовжуючи розробку діаграми класів моделі банкомату, додамо на неї в описаний спосіб спрямовану асоціацію між класом «Контролер

Controller” class and the “ATM Transaction” class (Figure 12).

Change the name for this association, by default the name of the association is missing. This can be done using the association property specification window. You can access the Association Properties dialog box by right-clicking the association line in the class diagram.

Банкомату» і класом «Транзакція Банкомату» (рис. 12)

Змінимо ім'я для даної асоціації, за замовчуванням ім'я асоціації відсутня. Це можна виконати за допомогою вікна специфікації властивостей асоціації. Доступ до діалогового вікна властивостей асоціації Association Properties можна отримати після натискання правою кнопкою миші на лінії асоціації на діаграмі класів.

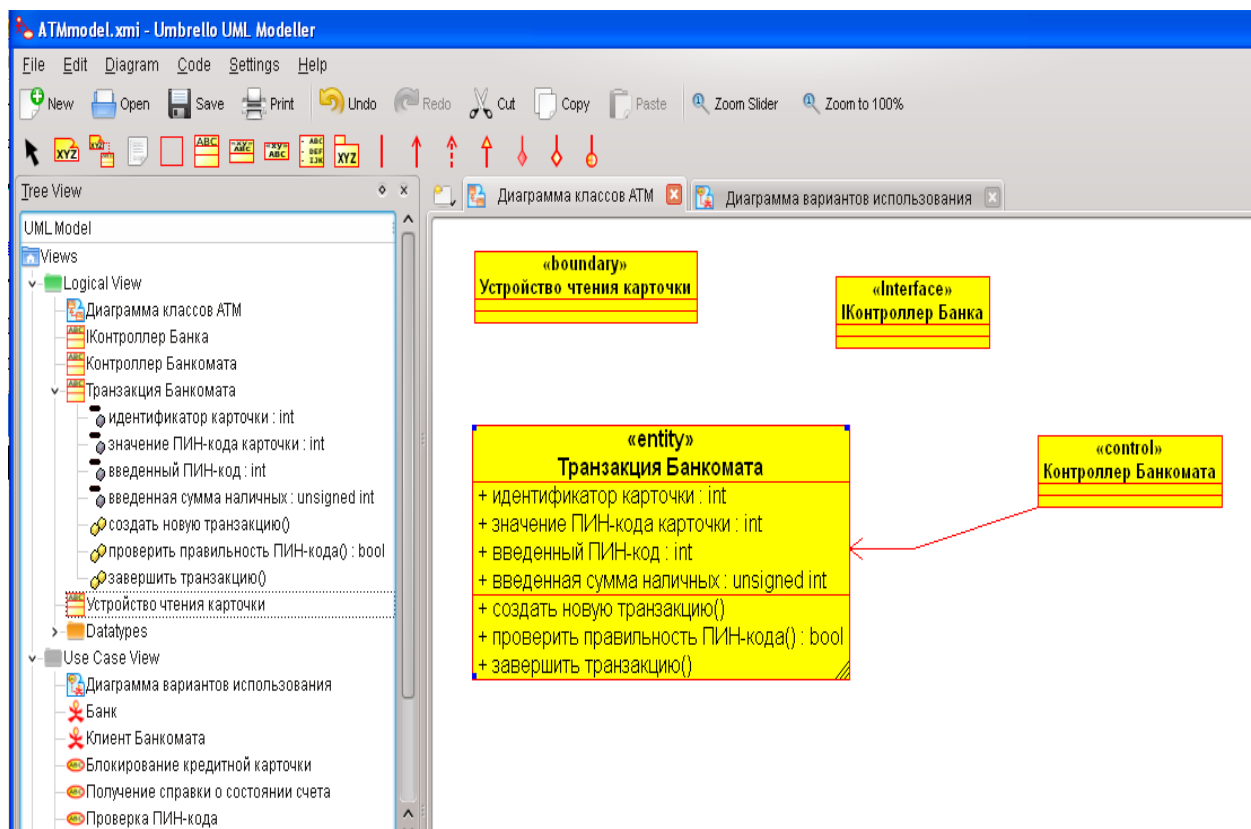


Figure 12 - Fragment of the class diagram of an ATM model with a directed association added to it

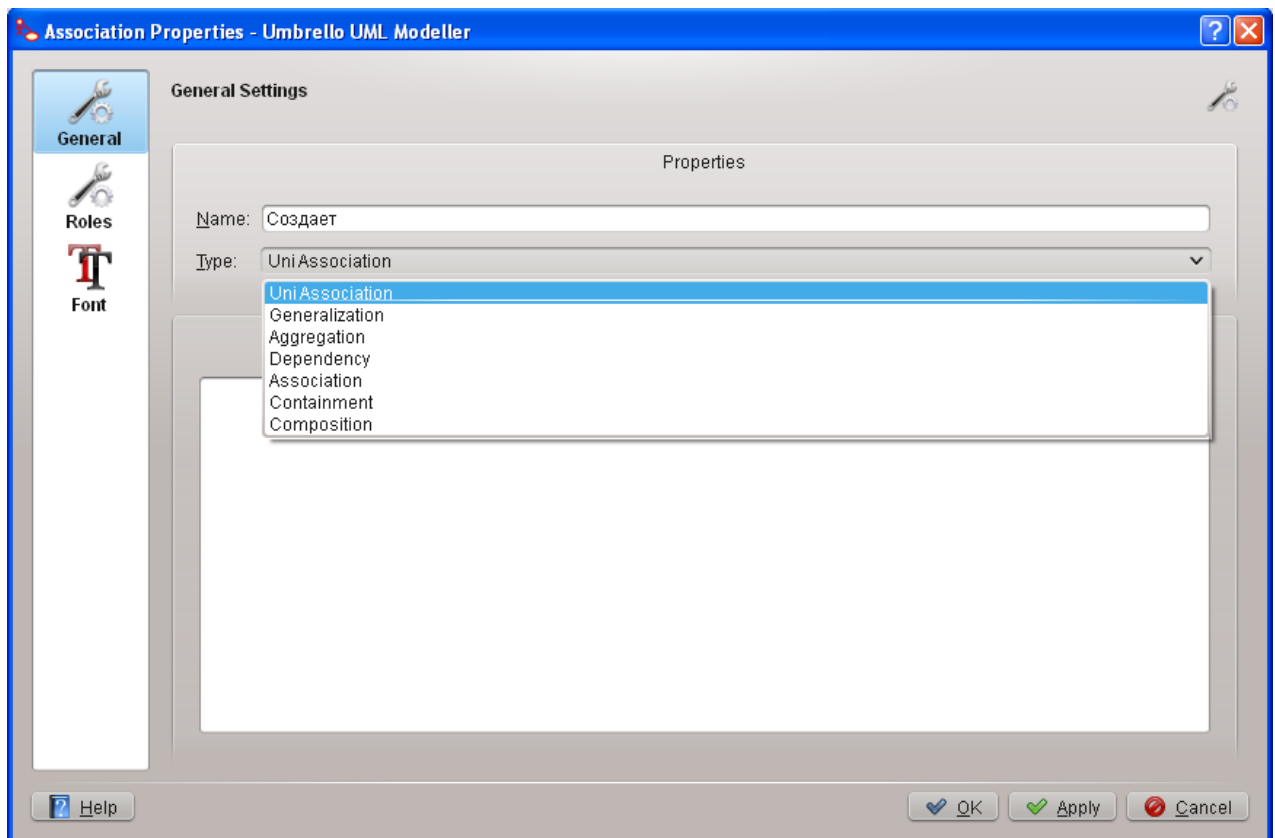


Figure 13 - General tab of the Association Properties menu

To set the name of the association, on the General tab in the Name field, enter the text of its name: “Creates” and click the Apply or OK button to save the results of editing the name of the association (Figure 13). As you can see in Figure 13, the Umbrello UML Modeller has various types of associations, namely UniAssociation, Generalization, Aggregation, Dependency, Assosiation, Containment, Composition, which can be selected in the Type field. We establish between the class “ATM Controller” and the class “ATM Transaction” a directed association of the UniAssociation type (one-to-one association).

Для завдання імені асоціації слід на вкладці General (Загальні) в поле введення Name (Ім'я) ввести текст її імені: «Створює» і натиснути кнопку Apply або OK, щоб зберегти результати редагування імені асоціації (рисунок 13). Як видно з малюнка 13 в Umbrello UML Modeller існують різні типи асоціацій, а саме UniAssociation, Generalization, Aggregation, Dependency, Assosiation, Containment, Composition, які можна вибрати в полі Type. Встановимо між класом «Контролер Банкомату» і класом «ТранзакціяБанкомата» спрямовану асоціацію типу UniAssociation (асоціація «один до одного»).

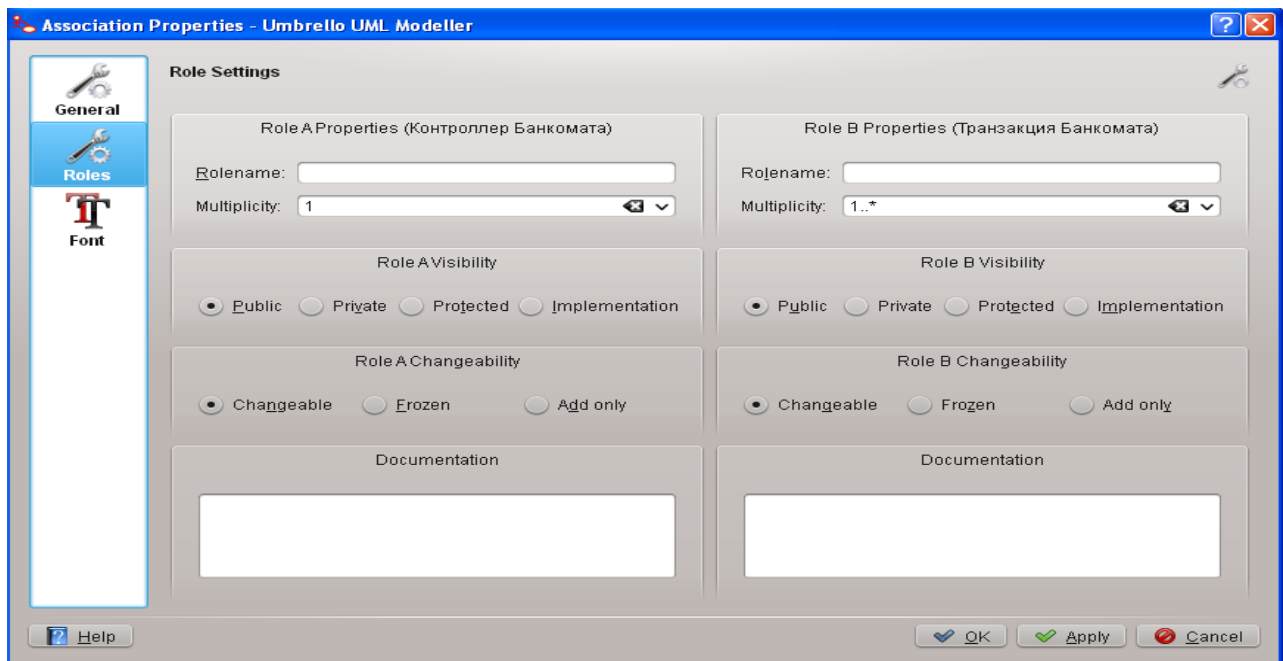


Figure 14 - Changing the properties of roles when setting associations

For an association, you can also specify the multiplicity of each of the ends of the association, a stereotype, use restrictions and roles, as well as some other properties. For the association added to the class diagram, set the multiplicity of the end of the association for the ATM controller class to 1. To do this, go to the Roles tab in the specification list of the association properties and select 1 in the Role A Properties field from the Multiplicity sub-list. Similarly, you should set the multiplicity of the end of the association for the ATM Transaction class to 1..n, for which, on the Role B Detail tab, select the value 1 .. * from the Multiplicity sub-list (Fig. 14). In essence, this will mean that each object of the ATM Controller class can be associated with one or more objects of the ATM Transaction class.

Для асоціації можна задати також кратність кожного з кінців асоціації, стереотип, використовувати обмеження і ролі, а також деякі інші властивості. Для доданої на діаграму класів асоціації задамо кратність кінця асоціації у класу Контролер Банкомату, рівну 1. Для цього слід у вікні специфікації властивостей асоціації перейти на вкладку Roles (Role Settings) і в поле Role A Properties вибрати значення 1 з вкладеного списку Multiplicity. Аналогічним чином слід задати кратність кінця асоціації у класу Транзакція Банкомату рівну 1..n, для чого на вкладці Role B Detail і слід вибрати значення 1 .. * з вкладеного списку Multiplicity (рис. 14). Змістовно це буде означати, що кожен об'єкт класу Контролер Банкомату може бути пов'язаний з одним або декількома об'єктами класу Транзакція Банкомати.

8 ADDING AGGREGATION AND COMPOSITION RELATIONSHIPS TO A CLASS DIAGRAM AND EDITING THEIR PROPERTIES

(ДОДАВАННЯ ВІДНОШЕНЬ АГРЕГАЦІЇ ТА КОМПОЗИЦІЇ НА ДІАГРАМУ КЛАСІВ І РЕДАГУВАННЯ ЇХ ВЛАСТИВОСТЕЙ)

You can add an aggregation relation between two classes to a chart in the following ways:

- Click on the button with the image the aggregation relation on a special toolbar and draw an aggregation line from one class to another.
- Draw an association line between the selected classes and change its properties in such a way as to turn this association into an aggregation.

In the second case, you should open the Association Properties window and on the General tab in the Type field put a mark in the Aggregation selection line.

Додати на діаграму відношення агрегації між двома класами за допомогою таких дій:

- Натиснути на кнопку із зображенням відношення агрегації на спеціальній панелі інструментів і провести лінію агрегації від одного класу до іншого.
- Провести лінію асоціації між обраними класами і змінити її властивості таким чином, щоб перетворити цю асоціацію в агрегацію.

У другому випадку слід відкрити вікно властивостей асоціації Association Properties і на вкладці General в поле Type виставити позначку в рядку вибору Aggregation.

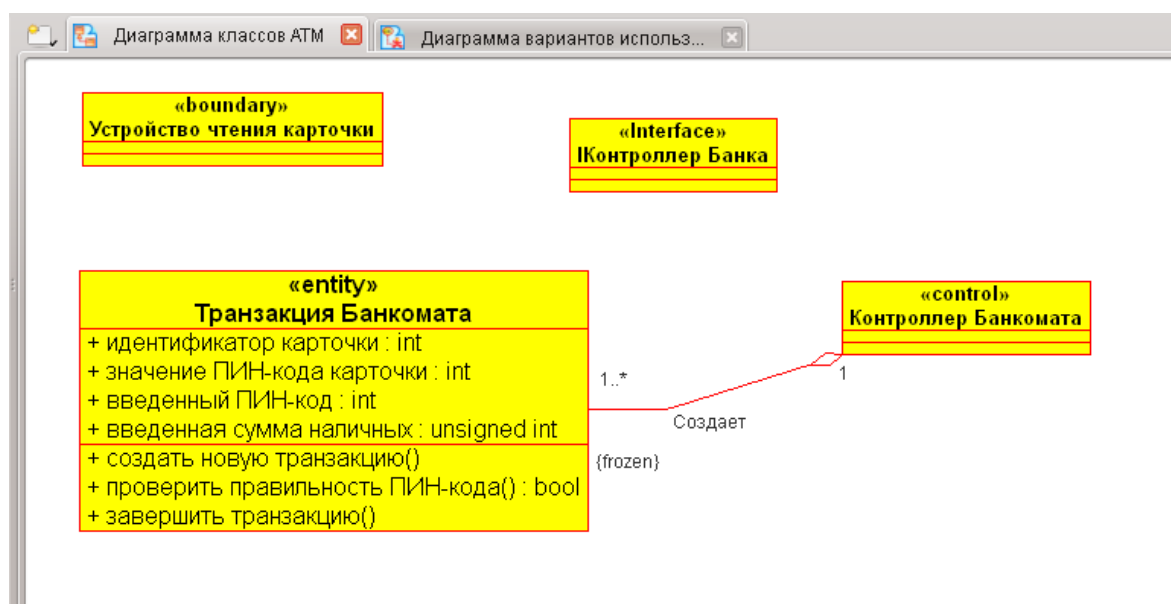


Figure 15 - Aggregation ratio

As an example, we will change the type of the association created earlier and make it aggregation. In essence, this will mean that the "ATM Controller" class will include the "ATM Transaction" class as an integral part; at the same time, the destruction of any object of the "ATM Controller" class should not lead to the destruction of the objects of the "ATM Transaction" class associated with it. For this purpose, on the Role B Changeability tab of the association "ATM Transaction" class, you should put a mark in the Frozen selection line (Figure 15).

В качестве примера изменим тип созданной ранее ассоциации и сделаем ее агрегацией. Содержательно это будет означать, что класс «Контроллер Банкомата» будет включать в себя в качестве составной части класс «Транзакция Банкомата»; при этом уничтожение любого объекта класса «Контроллер Банкомата» не должно привести к уничтожению ассоциированных с ним объектов класса «Транзакция Банкомата». С этой целью на вкладке Role B Changeability ассоциации класса «Транзакция Банкомата» следует выставить отметку в строке выбора Frozen (рис. 15).

9 ADDING A GENERALIZATION RELATIONSHIP TO A CLASS DIAGRAM AND EDITING ITS PROPERTIES

(ДОБАВЛЕНИЕ ОТНОШЕНИЯ ОБОБЩЕНИЯ НА ДИАГРАММУ КЛАССОВ И РЕДАКТИРОВАНИЕ ЕЕ СВОЙСТВ)

On a special toolbar, you must click the button with the image of the association icon and release the left mouse button. Next, on the class diagram, you need to select the first generalization element or the descendant from which the arrow emanates, and, releasing the left mouse button, move its pointer to the second relation element or ancestor to which the arrow is directed. After moving to the second element, the mouse button should be released, as a result of which an association line between two selected

На специальной панели инструментов необходимо нажать кнопку с изображением пиктограммы ассоциации и отпустить левую кнопку мыши. Далее на диаграмме классов надо выделить первый элемент обобщения или потомок, от которого исходит стрелка, и, отпустив нажатую левую кнопку мыши, переместить ее указатель ко второму элементу отношения или предку, к которому направлена стрелка. После перемещения ко второму элементу

classes will be added to the class diagram.

кнопку мыши следует отпустить, в результате чего на диаграмму классов будет добавлена линия ассоциации между двумя wybranными классами.

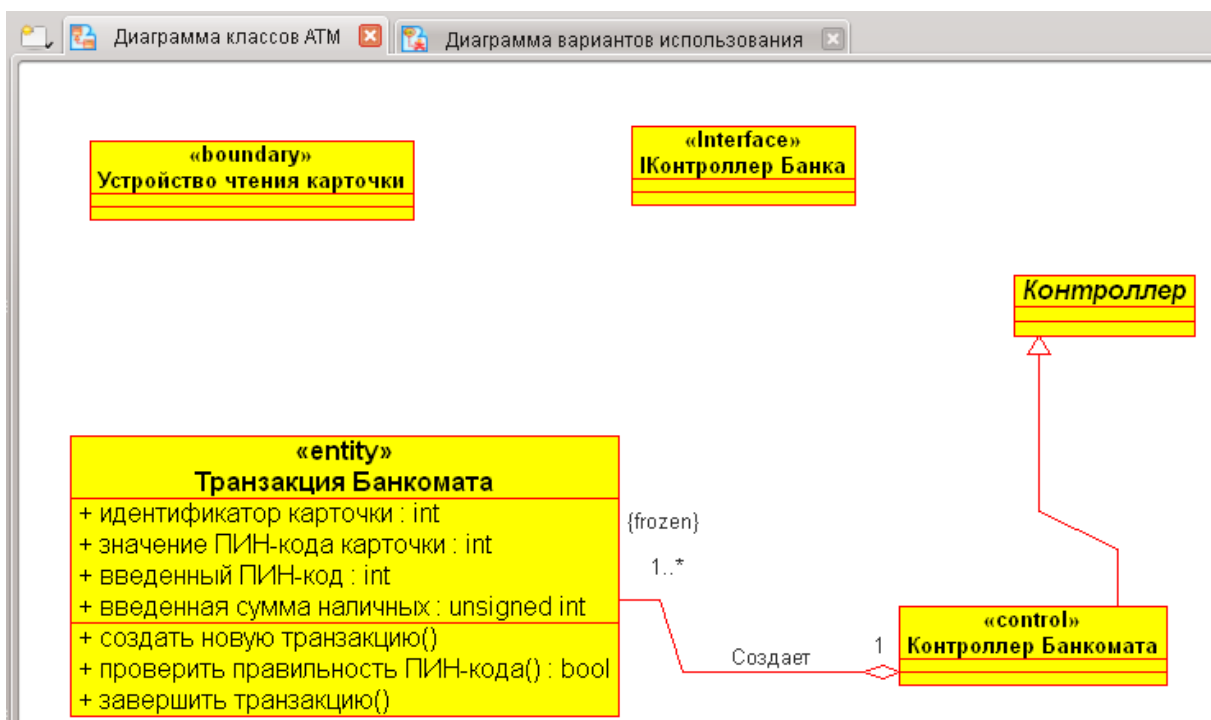


Рисунок 16 – Диаграмма классов модели банкомата после добавления на нее отношения обобщения

Continuing the development of the class diagram of the ATM model, we add to it in the described way the directional association between the class "ATM Controller" and the additionally created abstract class "Controller" (Figure 6.5). The last class may be intended for the specification of system attributes and operations necessary for the execution of the corresponding program. Recall that the abstract character of a class is indicated by writing in italics in ital name,

Продолжая разработку диаграммы классов модели банкомата, добавим на нее описанным способом направленную ассоциацию между классом «Контроллер Банкомата» и дополнительно созданным абстрактным классом «Контроллер» (рисунок 6.5). Последний класс может быть предназначен для спецификации системных атрибутов и операций, необходимых при исполнении соответствующей программы.

and for the specification of this class property, it is necessary to put a mark in the Abstract selection line on the Properties tab of the properties window of the class controller properties. The generalization relationship must be specified by choosing the type of association Generalization on the Association Properties tab.

Change the name of the generic relationship proposed by the default environment. This can be done using the association properties window. You can access the Association Properties generalization relationship properties dialog box after selecting the generalization line in the project class diagram and right-clicking on it.

To specify the name of the generalization, on the only tab General Settings in the input field Name, enter the text of its name: Inherit and click the Apply or OK button to save the results of editing the name of the association.

Напомним, что на абстрактный характер класса указывает написание курсивом его имени, а для спецификации данного свойства класса необходимо на вкладке Properties (Свойства) окна спецификации свойств класса «Контроллер» выставить отметку в строке выбора Abstract. Отношение обобщения необходимо указать выбрать на вкладке Association Properties тип ассоциации Generalization.

Изменим имя отношения обобщения, предложенное средой по умолчанию. Это можно выполнить с помощью окна свойств ассоциации. Доступ к диалоговому окну свойств отношения обобщения Association Properties можно получить после выделения линии обобщения на диаграмме классов проекта и щелчка на ней правой кнопки мыши.

Для задания имени обобщения следует на единственной вкладке General Settings (Общие Настройки) в поле ввода Name (Имя) ввести текст ее имени: Наследует и нажать кнопку Apply или ОК, чтобы сохранить результаты редактирования имени ассоциации.

10 FINAL CONSTRUCTION OF THE ATM MODEL CLASS DIAGRAM (ОКОНЧАТЕЛЬНОЕ ПОСТРОЕНИЕ ДИАГРАММЫ КЛАССОВ МОДЕЛИ БАНКОМАТА)

For the final construction of the class diagram of the considered ATM model, it is necessary to add the remaining classes and associations as described above, and also to specify the stereotypes, attributes and operations of these classes. To this end, you must perform the following steps:

1. For the class “Bank Controller” add the operation: “check card ID (card ID: Integer)” with the visibility quantifier public. For the type of return result for this operation, select the Boolean type (logical), and set the identifier of the card as an integer argument. To set an argument, you need to go to the Operation Properties tab of the specification window for the properties of this operation and, after adding the argument, use the New Parameter button to enter the argument name and its type Integer in the corresponding input fields.

2. For the class “Bank Controller” add the operation: “open a client account (card ID: Integer)” with the visibility quantifier public. As an integer argument to this operation, you must specify the card ID.

Для окончательного построения диаграммы классов рассматриваемой модели банкомата следует описанным выше способом добавить оставшиеся классы и ассоциации, а также специфицировать стереотипы, атрибуты и операции этих классов. С этой целью следует выполнить следующие действия:

1. Для класса «Контроллер Банка» добавить операцию: «проверить идентификатор карточки (идентификатор карточки: Integer)» с квантором видимости public. В качестве типа возвращаемого результата для этой операции следует выбрать тип Boolean (логический), а в качестве целочисленного аргумента задать идентификатор карточки. Для задания аргумента необходимо перейти на вкладку Operation Properties окна спецификации свойств данной операции и после добавления аргумента с помощью кнопки New Parameter ввести имя аргумента и его тип Integer в соответствующие поля ввода.

2. Для класса «Контроллер Банка» добавить операцию: «открыть счет клиента (идентификатор карточки: Integer)» с квантором видимости public. В качестве целочисленного аргумента этой операции следует задать идентификатор карточки.

3. For the “IController Bank” class, add the operation: “check the client's balance (card ID: Integer, entered cash amount: float)” with the visibility quantifier public. As the return type for this operation, select the Boolean type (logical). As the first integer argument of this operation, you must specify the card identifier, and as the second argument, enter the amount of cash entered with the float type (depending on the programming language for generating the code, there may be Currency - Money).

4. For the class “Bank Controller”, add the operation: “reduce the client’s account (card ID: Integer, entered cash amount: float)” with the visibility quantifier public. As the return type for this operation, select the Boolean type (logical). As the first integer argument of this operation, you should specify the identifier of the card, and as the second argument, enter the amount of cash entered with the type float (Cash).

5. For the class "Card reader" add the operation: "read the identifier of the card ()" with the visibility quantifier public. For the type of return result for this operation, select the type Integer (integer), and in the documentation section of this operation, enter the explanatory text: “It is

3. Для класса «ИКонтроллер Банка» добавить операцию: «проверить баланс клиента (идентификатор карточки: Integer, введенная сумма наличных: float)» с квантором видимости public. В качестве типа возвращаемого результата для этой операции следует выбрать тип Boolean (логический). В качестве первого целочисленного аргумента этой операции следует задать идентификатор карточки, а в качестве второго аргумента - введенная сумма наличных с типом float (в зависимости от языка программирования для генерации кода может быть Currency - Денежный).

4. Для класса «ИКонтроллер Банка» добавить операцию: «уменьшить счет клиента (идентификатор карточки: Integer, введенная сумма наличных: float)» с квантором видимости public. В качестве типа возвращаемого результата для этой операции следует выбрать тип Boolean (логический). В качестве первого целочисленного аргумента этой операции следует задать идентификатор карточки, а в качестве второго аргумента - введенная сумма наличных с типом float (Денежный).

5. Для класса «Устройство чтения карточки» добавить операцию: «прочитать идентификатор карточки()» с квантором видимости public. В качестве типа возвращаемого результата для этой операции следует выбрать тип Integer (целочисленный), а

called after the credit card is inserted into the Card Reader”.

6. For the class “Card reader” add the operation: “read PIN-code ()” with the visibility quantifier public. For the type of return result for this operation, select the type Integer (integer), and in the documentation section of this operation, enter the explanatory text: “It is called after the credit card is inserted into the Card Reader”.

7. For the class “Card reader” add the operation: “return the credit card ()” with the visibility quantifier public. An explanatory text should be entered into the documentation section of this operation: “Called after the transaction is completed”.

8. For the class “Card reader” add the operation: “block credit card ()” with the visibility quantifier public. An explanatory text should be entered in the documentation section of this operation: “Called after the fact of the loss of a credit card by the owner has been established”.

9. Add a class with the name “ATM Screen” for which to select the boundary stereotype. This class is also located on

в секцию документации данной операции следует ввести поясняющий текст: «Вызывается после того, как кредитная карточка вставлена в Устройство чтения карточки».

6. Для класса «Устройство чтения карточки» добавить операцию: «прочитать ПИН-код()» с квантором видимости public. В качестве типа возвращаемого результата для этой операции следует выбрать тип Integer (целочисленный), а в секцию документации данной операции следует ввести поясняющий текст: «Вызывается после того, как кредитная карточка вставлена в Устройство чтения карточки».

7. Для класса «Устройство чтения карточки» добавить операцию: «вернуть кредитную карточку()» с квантором видимости public. В секцию документации данной операции следует ввести поясняющий текст: «Вызывается после завершения транзакции».

8. Для класса «Устройство чтения карточки» добавить операцию: «блокировать кредитную карточку()» с квантором видимости public. В секцию документации данной операции следует ввести поясняющий текст: «Вызывается после того, как установлен факт утраты кредитной карточки владельцем».

9. Добавить класс с именем «Экран Банкомата», для которого выбрать стереотип boundary. Данный класс

the border of the simulated system, as indicated by this stereotype. An explanatory text should be entered into the documentation section of this class: “Installed at an ATM”.

10. For the “ATM Screen” class, add the operation: “show the options menu ()” with the visibility quantifier public.

11. For the "ATM Screen" class, add the operation: "show the withdrawal menu ()" with the visibility quantifier public.

12. A little rest and preserve the masterpiece you created.

13. Add a class with the name “ATM Keyboard” for which to select the boundary stereotype. An explanatory text should be entered into the documentation section of this class: “Installed at an ATM”.

14. For the “ATM Keyboard” class, add the operation: “enter the PIN code ()” with the visibility quantifier public. As the return type for this operation, select the Integer type, and enter the explanatory text in the documentation section of this operation: “Called after the client has entered the PIN code value from the keyboard”.

15. For the “ATM Keyboard” class, add the operation: “enter transaction type ()” with the visibility quantifier public. As the return type for this operation, select

также находится на границе моделируемой системы, на что и указывает этот стереотип. В секцию документации данного класса следует ввести поясняющий текст: «Устанавливается на банкомате».

10. Для класса «Экран Банкомата» добавить операцию: «показать меню опций()» с квантором видимости public.

11. Для класса «Экран Банкомата» добавить операцию: «показать меню снятия суммы()» с квантором видимости public.

12. Немного отдышаться и сохранить сотворенный Вами шедевр.

13. Добавить класс с именем «Клавиатура Банкомата», для которого выбрать стереотип boundary. В секцию документации данного класса следует ввести поясняющий текст: «Устанавливается на банкомате».

14. Для класса «Клавиатура Банкомата» добавить операцию: «ввести ПИН-код()» с квантором видимости public. В качестве типа возвращаемого результата для этой операции следует выбрать тип Integer, а в секцию документации данной операции следует ввести поясняющий текст: «Вызывается после того, как клиент ввел значение ПИН-кода с клавиатуры».

15. Для класса «Клавиатура Банкомата» добавить операцию: «ввести тип транзакции()» с квантором видимости public. В качестве типа

the Boolean type (logical), and enter the explanatory text in the documentation section of this operation: "Returns True if the client chooses a cash withdrawal, and False if the client chooses to receive a statement about the account status".

16. For the "ATM Keyboard" class, add the operation: "enter the cash withdrawal amount ()" with the visibility quantifier public. For the type of return result for this operation, select the type float (Money), and in the documentation section of this operation, enter the explanatory text: "It is called after the client has entered the value of the withdrawal amount from the keyboard".

17. Add a class with the name "Cash Dispensing Device" for which to select the boundary stereotype. An explanatory text should be entered into the documentation section of this class: "Installed at an ATM".

18. For the class "Cash withdrawal device" add the operation: "issue cash ()" with the visibility quantifier public. An explanatory text should be entered into the documentation section of this operation: "Called after the withdrawal of the amount requested by the client from the account has been completed."

возвращаемого результата для этой операции следует выбрать тип Boolean (логический), а в секцию документации данной операции следует ввести поясняющий текст: «Возвращает значение Истина, если клиент выбирает снятие наличных, и значение Ложь, если клиент выбирает получение справки о состоянии счета».

16. Для класса «Клавиатура Банкомата» добавить операцию: «ввести сумму снятия наличных()» с квантором видимости public. В качестве типа возвращаемого результата для этой операции следует выбрать тип float (Денежный), а в секцию документации данной операции следует ввести поясняющий текст: «Вызывается после того, как клиент ввел значение снимаемой суммы с клавиатуры».

17. Добавить класс с именем «Устройство выдачи наличных», для которого выбрать стереотип boundary. В секцию документации данного класса следует ввести поясняющий текст: «Устанавливается на банкомате».

18. Для класса «Устройство выдачи наличных» добавить операцию: «выдать наличные()» с квантором видимости public. В секцию документации данной операции следует ввести поясняющий текст: «Вызывается после того, как выполнено снятие запрошенной клиентом суммы со счета».

19. Add a class with the name “ATM Printer” for which to select the boundary stereotype. An explanatory text should be entered into the documentation section of this class: “Installed at an ATM”.

20. For the class "ATM Printer" add the operation: "print the check ()" with the visibility quantifier public. An explanatory text should be entered into the documentation section of this operation: “Called upon an additional request from the client”

21. Add a directional association from the “ATM Controller” class to the “Card Reader” class. Set the multiplicity of the ends of this association to 1.

22. Add a directed association from the “ATM Controller” class to the “ATM Printer” class. Set the multiplicity of the ends of this association to 1.

23. Add a directed association from the “ATM Controller” class to the “ATM Keyboard” class. Set the multiplicity of the ends of this association to 1.

24. Add a directed association from the “ATM Controller” class to the “Cash Dispensing Device” class. Set the multiplicity of the ends of this association to 1.

25. Add a directed association from the “ATM Controller” class to the “ATM

19. Добавить класс с именем «Принтер Банкомата», для которого выбрать стереотип boundary. В секцию документации данного класса следует ввести поясняющий текст: «Устанавливается на банкомате».

20. Для класса «Принтер Банкомата» добавить операцию: «распечатать чек()» с квантором видимости public. В секцию документации данной операции следует ввести поясняющий текст: «Вызывается по дополнительному запросу клиента».

21. Добавить направленную ассоциацию от класса «Контроллер Банкомата» к классу «Устройство чтения карточки». В качестве кратности концов этой ассоциации установить значение 1.

22. Добавить направленную ассоциацию от класса «Контроллер Банкомата» к классу «Принтер Банкомата». В качестве кратности концов этой ассоциации установить значение 1.

23. Добавить направленную ассоциацию от класса «Контроллер Банкомата» к классу «Клавиатура Банкомата». В качестве кратности концов этой ассоциации установить значение 1.

24. Добавить направленную ассоциацию от класса «Контроллер Банкомата» к классу «Устройство выдачи наличных». В качестве кратности концов этой ассоциации установить значение 1.

25. Добавить направленную ассоциацию от класса «Контроллер

Screen” class. Set the multiplicity of the ends of this association to 1.

26. Add a directional association from the class “ATM Controller” to the class “I Controller Bank”. As the multiplicity of the end of this association for the first class, set the value to *, and the multiplicity of the end of the association for the second class to set the value 1. As a stereotype of this association, select <<communicate>> from the sub-list. The use of this stereotype means that between these classes there must be a physical relationship. The class diagram constructed as a result of the indicated actions will have the following form (Fig.17).

Банкомата» к классу «Экран Банкомата». В качестве кратности концов этой ассоциации установить значение 1.

26. Добавить направленную ассоциацию от класса «Контроллер Банкомата» к классу «IКонтроллер Банка». В качестве кратности конца этой ассоциации для первого класса установить значение *, а кратности конца ассоциации для второго класса установить значение 1. В качестве стереотипа данной ассоциации выбрать из вложенного списка значение <<communicate>>. Применение данного стереотипа означает, что между этими классами должна существовать физическая взаимосвязь. Построенная в результате указанных действий диаграмма классов будет иметь следующий вид (рис. 17).

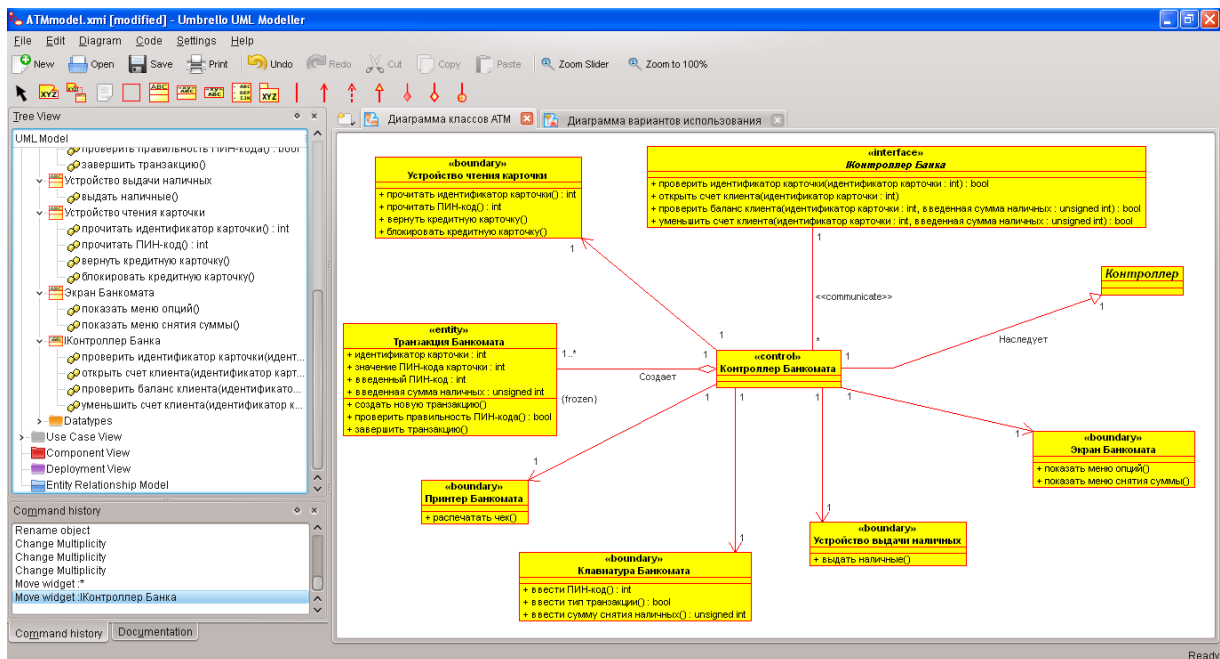


Рисунок 17 – Окончательный вид диаграммы классов для разрабатываемой модели банкомата

CONCLUSION

ЗАКЛЮЧЕНИЕ

Class diagrams underlie the application architecture. They can be simple and complex.

The only way to master them is to model them.

Class diagrams, as you understand, are extremely rich in available methods for determining their properties and relationships.

Диаграммы классов лежат в основе архитектуры приложения. Они могут быть простыми и сложными.

Единственный способ освоить их - это смоделировать их.

Диаграммы классов, как вы понимаете, чрезвычайно богаты доступными методами для определения их свойств и отношений.

BIBLIOGRAPHY

ЛИТЕРАТУРА

1. Леоненков А.В. Самоучитель UML. 2-е издание - СПб.: "БХВ-Петербург", 2004. - 432 с.
2. Буч Г., Джекобсон А., Рамбо Дж. Язык UML. Руководство пользователя - М.: ДМК, 2000. - 432 с.
3. Буч Г., Рамбо Дж., Якобсон А. UML: специальный справочник - СПб: "Питер", 2001. - 656 с.
4. Леффингуэлл Д., Уидриг Д. Принципы работы с требованиями к программному обеспечению. Унифицированный подход - М.: "Вильямс", 2002. - 448 с.
5. Starr L. Executable UML. How to build class models - Prentice Hall PTR, 2002. - 418 p.
6. Wampler B.E. The Essence Object-Oriented Programming with Java and UML - Addison-Wesley, 2002, - 290 p.

Навчальне видання

ХАЦЬКО Наталія Євгенівна
ГАВРИЛЕНКО Світлана Юріївна

МЕТОДИЧНІ ВКАЗІВКИ
до виконання лабораторної роботи
Розробка діаграми класів у середовищі
Umbrello UML Modeller

з курсу “Основи архітектури програмних систем”
 (“Fundamentals of program systems architecture”)

Роботу до видання рекомендував Горілий О.В.

В авторській редакції

Самостійне електронне видання