**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**

**Національний Технічний Університет України**

**«Київський Політехнічний Інститут»**

*Факультет інформатики та обчислювальної техніки*

*Кафедра обчислювальної техніки*

**Лабораторна робота №3**

*з дисципліни «Інженерія програмного забезпечення»*

*на тему: «Структурні шаблони проектування.*

*Шаблони Composite, Decorator, Proxy»*

**Виконав:**

студент 2-го курсу ФІОТ

групи ІО-44

*Барабаш Т.А.*

**Номер залікової книжки:** 4403

**Варіант:** 11

**Перевірив:**

Старший викладач

к.т.н, с.н.с.

*Антонюк А.І.*

**Київ – 2015**

**ЛАБОРАТОРНА РОБОТА №3**

*Структурні шаблони проектування. Шаблони Composite, Decorator, Proxy.*

**Мета:** Ознайомлення з видами шаблонів проектування ПЗ. Вивчення структурних шаблонів. Отримання базових навичок з застосування шаблонів Composite, Decorator та Proxy.

**I. Завдання**

**1.** Ознайомитись з призначенням та видами шаблонів проектування ПЗ. Вивчити класифікацію шаблонів проектування ПЗ. Знати назви шаблонів, що відносяться до певного класу.

**2.** Вивчити структурні шаблони проектування ПЗ. Знати загальну характеристику структурних шаблонів та призначення кожного з них.

**3.**  Детально вивчити структурні шаблони проектування Composite, Decorator та Proxy. Для кожного з них:

• вивчити Шаблон, його призначення, альтернативні назви, мотивацію, випадки, коли його застосування є доцільним та результати такого застосування;

• знати особливості реалізації Шаблону, споріднені шаблони, відомі випадки його застосування в програмних додатках;

• вільно володіти структурою Шаблону, призначенням його класів та відносинами між ними;

• вміти розпізнавати Шаблон в UML діаграмі класів та будувати сирцеві коди Java-класів, що реалізують шаблон.

**4.** В підготованому проекті (ЛР1) створити програмний пакет com.lab111.labwork3. В пакеті розробити інтерфейси і класи, що реалізують завдання (згідно варіанту) з застосуванням одного чи декількох шаблонів (п.3). В класах, що розробляються, повністю реалізувати методи, пов'язані з функціюванням Шаблону. Методи, що реалізують бізнес-логіку, закрити заглушками з виводом на консоль інформації про викликаний метод та його аргументи. Приклад реалізації бізнес-методу:

*void draw(int x, int y){*

*System.out.println(“Метод draw з параметрами x=”+x+” y=”+y);*

*}*

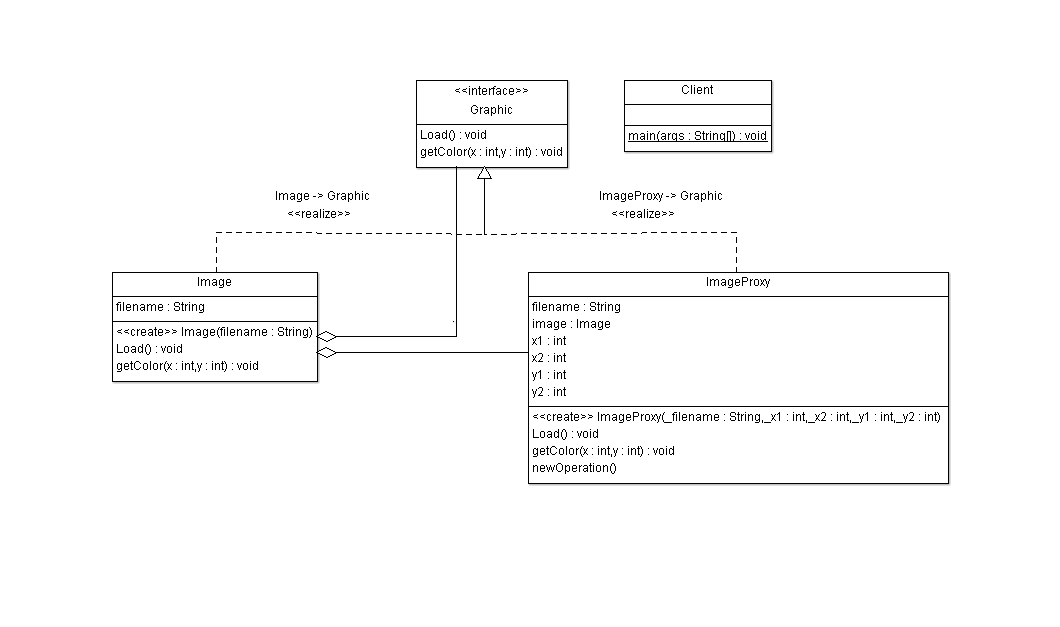
**5.** За допомогою автоматизованих засобів виконати повне документування розроблених класів (також методів і полів), при цьому документація має в достатній мірі висвітлювати роль певного класу в загальній структурі Шаблону та особливості конкретної реалізації.

**6.** **Визначення варіанту завдання:**

**Номер залікової книжки:** 4403;

**Варіант:** 4403 *mod* 12 = **11** – *Визначити специфікації класів та реалізацію методів для маніпулювання зображеннями з можливістю контролювання доступу до об'єкта — доступ відкритий лише до точок чиї координати (х, у) лежать в межах х1<х<х2 и у1<у<у2 (значення х1, х2, у1, у2 задаються в конструкторі). Реалізувати бізнес- метод для визначення кольору точки за його координатами.*

**II. Діаграма класів**

****

**IIІ. Код програми**

**1. Client**

**package** com.lab111.labwork3;

**public** **class** Client {

**public** **static** **void** main(String[] args){

Graphic image1 = **new** ImageProxy("image.jpg", 1, 5, 10, 0);

image1.getColor(4, 5);

image1.getColor(10, 5);

image1.Load();

}

}

**2. Graphic**

**package** com.lab111.labwork3;

/\*\*

\* Describes a graphical element

\* **@author** Taras Barabash

\*

\*/

**public** **interface** Graphic {

/\*\*

\* Loads an image from hard-drive

\*/

**void** Load();

/\*\*

\* Outputs to console the color of pixel with coordinates (x and y)

\* **@param** x abscissa of pixel

\* **@param** y ordinate of pixel

\*/

**void** getColor(**int** x, **int** y);

}

**3. Image**

**package** com.lab111.labwork3;

/\*\*

\* Describes the image - real object

\* **@author** Taras Barabash

\*

\*/

**public** **class** Image **implements** Graphic {

/\*\*

\* Name of file-image

\*/

**private** String filename;

/\*\*

\* Creates the object of class

\* **@param** filename name of file

\*/

Image(String filename){

**this**.filename = filename;

}

**public** **void** Load() {

System.***out***.println("Loads the image from hard drive...");

}

**public** **void** getColor(**int** x, **int** y) {

System.***out***.println("Returns the color of pixel x = " + x + ", y = " + y);

}

}

**4. ImageProxy**

**package** com.lab111.labwork3;

/\*\*

\* Restricts the access to the image

\* **@author** Taras Barabash

\*

\*/

**public** **class** ImageProxy **implements** Graphic {

/\*\*

\* image name

\*/

**private** String filename;

/\*\*

\* image that is being worked on

\*/

**private** Image image;

/\*\*

\* lowest available abscissa of pixel

\*/

**private** **int** x1;

/\*\*

\* highest available abscissa of pixel

\*/

**private** **int** x2;

/\*\*

\* lowest available ordinate of pixel

\*/

**private** **int** y1;

/\*\*

\* highest available ordinate of pixel

\*/

**private** **int** y2;

/\*\*

\* creates the object of class Proxy, access area is restricted by lines

\* x = x1, x = x2; y = y1, y = y2.

\* **@param** \_filename - path to the image

\* **@param** \_x1 abscissa line 1, which restricts the access area

\* **@param** \_x2 abscissa line 2, which restricts the access area

\* **@param** \_y1 ordinate line 1, which restricts the access area

\* **@param** \_y2 ordinate line 2, which restricts the access area

\*/

ImageProxy(String \_filename, **int** \_x1, **int** \_x2, **int** \_y1, **int** \_y2){

image = **new** Image(\_filename);

filename = \_filename;

**if**(\_x1 > \_x2) {

x1 = \_x2;

x2 = \_x1;

} **else** {

x1 = \_x1;

x2 = \_x2;

}

**if**(\_y1 > \_y2){

y1 = \_y2;

y2 = \_y1;

} **else** {

y1 = \_y1;

y2 = \_y2;

}

}

**public** **void** Load() {

System.***out***.println("Name of the image: " + filename + " ");

image.Load();

}

**public** **void** getColor(**int** x, **int** y) {

**if**(x >= x1 && x <= x2 && y >= y1 && y <= y2) image.getColor(x, y);

**else** System.***out***.println("Access denied! You can only read the pixels: " + x1 + " <= x <= " + x2 + ", " + y1 + " <= y <= " + y2);

}

}

**ІV. Висновок**

Ознайомились з видами шаблонів проектування програмного забезпечення. Розроблена відповідна тестова програма. Результати успішної роботи тестової програми наведені нижче підтверджують правильність обраних рішень.

Returns the color of pixel x = 4, y = 5

Access denied! You can only read the pixels: 1 <= x <= 5, 0 <= y <= 10

Name of the image: image.jpg

Loads the image from hard drive...

**C**

[**Client**](file:///C:\Workspace\template\doc\com\lab111\labwork3\Client.html) - Class in [com.lab111.labwork3](file:///C:\Workspace\template\doc\com\lab111\labwork3\package-summary.html)

[**Client()**](file:///C:\Workspace\template\doc\com\lab111\labwork3\Client.html#Client--) - Constructor for class com.lab111.labwork3.[Client](file:///C:\Workspace\template\doc\com\lab111\labwork3\Client.html)

[com.lab111](file:///C:\Workspace\template\doc\com\lab111\package-summary.html) - package com.lab111

[com.lab111.labwork3](file:///C:\Workspace\template\doc\com\lab111\labwork3\package-summary.html) - package com.lab111.labwork3

**F**

[**filename**](file:///C:\Workspace\template\doc\com\lab111\labwork3\Image.html#filename) - Variable in class com.lab111.labwork3.[Image](file:///C:\Workspace\template\doc\com\lab111\labwork3\Image.html)

Name of file-image

[**filename**](file:///C:\Workspace\template\doc\com\lab111\labwork3\ImageProxy.html#filename) - Variable in class com.lab111.labwork3.[ImageProxy](file:///C:\Workspace\template\doc\com\lab111\labwork3\ImageProxy.html)

image name

**G**

[**getColor(int, int)**](file:///C:\Workspace\template\doc\com\lab111\labwork3\Graphic.html#getColor-int-int-) - Method in interface com.lab111.labwork3.[Graphic](file:///C:\Workspace\template\doc\com\lab111\labwork3\Graphic.html)

Outputs to console the color of pixel with coordinates (x and y)

[**getColor(int, int)**](file:///C:\Workspace\template\doc\com\lab111\labwork3\Image.html#getColor-int-int-) - Method in class com.lab111.labwork3.[Image](file:///C:\Workspace\template\doc\com\lab111\labwork3\Image.html)

[**getColor(int, int)**](file:///C:\Workspace\template\doc\com\lab111\labwork3\ImageProxy.html#getColor-int-int-) - Method in class com.lab111.labwork3.[ImageProxy](file:///C:\Workspace\template\doc\com\lab111\labwork3\ImageProxy.html)

[**Graphic**](file:///C:\Workspace\template\doc\com\lab111\labwork3\Graphic.html) - Interface in [com.lab111.labwork3](file:///C:\Workspace\template\doc\com\lab111\labwork3\package-summary.html)

Describes a graphical element

**I**

[**Image**](file:///C:\Workspace\template\doc\com\lab111\labwork3\Image.html) - Class in [com.lab111.labwork3](file:///C:\Workspace\template\doc\com\lab111\labwork3\package-summary.html)

Describes the image - real object

[**Image(String)**](file:///C:\Workspace\template\doc\com\lab111\labwork3\Image.html#Image-java.lang.String-) - Constructor for class com.lab111.labwork3.[Image](file:///C:\Workspace\template\doc\com\lab111\labwork3\Image.html)

Creates the object of class

[**image**](file:///C:\Workspace\template\doc\com\lab111\labwork3\ImageProxy.html#image) - Variable in class com.lab111.labwork3.[ImageProxy](file:///C:\Workspace\template\doc\com\lab111\labwork3\ImageProxy.html)

image that is being worked on

[**ImageProxy**](file:///C:\Workspace\template\doc\com\lab111\labwork3\ImageProxy.html) - Class in [com.lab111.labwork3](file:///C:\Workspace\template\doc\com\lab111\labwork3\package-summary.html)

Restricts the access to the image

[**ImageProxy(String, int, int, int, int)**](file:///C:\Workspace\template\doc\com\lab111\labwork3\ImageProxy.html#ImageProxy-java.lang.String-int-int-int-int-) - Constructor for class com.lab111.labwork3.[ImageProxy](file:///C:\Workspace\template\doc\com\lab111\labwork3\ImageProxy.html)

creates the object of class Proxy, access area is restricted by lines x = x1, x = x2; y = y1, y = y2.

**L**

[**Load()**](file:///C:\Workspace\template\doc\com\lab111\labwork3\Graphic.html#Load--) - Method in interface com.lab111.labwork3.[Graphic](file:///C:\Workspace\template\doc\com\lab111\labwork3\Graphic.html)

Loads an image from hard-drive

[**Load()**](file:///C:\Workspace\template\doc\com\lab111\labwork3\Image.html#Load--) - Method in class com.lab111.labwork3.[Image](file:///C:\Workspace\template\doc\com\lab111\labwork3\Image.html)

[**Load()**](file:///C:\Workspace\template\doc\com\lab111\labwork3\ImageProxy.html#Load--) - Method in class com.lab111.labwork3.[ImageProxy](file:///C:\Workspace\template\doc\com\lab111\labwork3\ImageProxy.html)

**M**

[**main(String[])**](file:///C:\Workspace\template\doc\com\lab111\labwork3\Client.html#main-java.lang.String:A-) - Static method in class com.lab111.labwork3.[Client](file:///C:\Workspace\template\doc\com\lab111\labwork3\Client.html)

[**main(String[])**](file:///C:\Workspace\template\doc\com\lab111\TestMain.html#main-java.lang.String:A-) - Static method in class com.lab111.[TestMain](file:///C:\Workspace\template\doc\com\lab111\TestMain.html)

Invokes at application startup.

**T**

[**TestMain**](file:///C:\Workspace\template\doc\com\lab111\TestMain.html) - Class in [com.lab111](file:///C:\Workspace\template\doc\com\lab111\package-summary.html)

Template first class.

[**TestMain()**](file:///C:\Workspace\template\doc\com\lab111\TestMain.html#TestMain--) - Constructor for class com.lab111.[TestMain](file:///C:\Workspace\template\doc\com\lab111\TestMain.html)

Constructor.

**X**

[**x1**](file:///C:\Workspace\template\doc\com\lab111\labwork3\ImageProxy.html#x1) - Variable in class com.lab111.labwork3.[ImageProxy](file:///C:\Workspace\template\doc\com\lab111\labwork3\ImageProxy.html)

lowest available abscissa of pixel

[**x2**](file:///C:\Workspace\template\doc\com\lab111\labwork3\ImageProxy.html#x2) - Variable in class com.lab111.labwork3.[ImageProxy](file:///C:\Workspace\template\doc\com\lab111\labwork3\ImageProxy.html)

highest available abscissa of pixel

**Y**

[**y1**](file:///C:\Workspace\template\doc\com\lab111\labwork3\ImageProxy.html#y1) - Variable in class com.lab111.labwork3.[ImageProxy](file:///C:\Workspace\template\doc\com\lab111\labwork3\ImageProxy.html)

lowest available ordinate of pixel

[**y2**](file:///C:\Workspace\template\doc\com\lab111\labwork3\ImageProxy.html#y2) - Variable in class com.lab111.labwork3.[ImageProxy](file:///C:\Workspace\template\doc\com\lab111\labwork3\ImageProxy.html)

highest available ordinate of pixel