

КУРСОВА РОБОТА

**З ДИСЦИПЛІНИ «МАТЕМАТИЧНІ МЕТОДИ ТА ТЕХНОЛОГІЇ
ТЕСТУВАННЯ І ВЕРИФІКАЦІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ»
ТЕМА: «СКЛАДАННЯ ТЕСТІВ, ПРОВЕДЕННЯ ТЕСТУВАННЯ ТА
ОЦІНКА ПОВНОТИ ПРОВЕДЕНИХ ТЕСТІВ НА ПРИКЛАДІ
ПРОГРАМНОЇ СИСТЕМИ «КАЛЬКУЛЯТОР ВАЛЮТ»**

**Виконала: студентка 2 курсу, групи КБ-22,
Семенякіна Владислава Сергіївна
Керівник: ст. викл. Мелкозьорова О.М.**

Харків – 2019

Тестування

- Тестування програмного забезпечення (ПЗ) (Software Testing) – перевірка відповідності між реальним і очікуваним поведінкою програми, що здійснюється на кінцевому наборі тестів, обраному певним чином.
- Усі види тестування поділяються на:
 - функціональні;
 - нефункціональні;
 - пов'язані зі змінами.

Функціональне тестування

- Функціональне тестування – тип тестування, за допомогою якого перевіряється, що кожна функція програмного додатка, працює відповідно до специфікації вимог. Це тестування, в основному, включає в себе тестування методом Чорного Ящика (Black box testing), тобто не потребує вихідного коду програми.
- Головною метою функціонального тестування є перевірка функціональності програмної системи. В основному воно базується на:
 - тестуванні основних функцій програми (**Mainline functions**);
 - тестуванні зручності використання системи (**Basic Usability**);
 - перевірці доступності системи для користувача (**Accessibility**);
 - тестуванні перевірки на наявність помилок (**Error Conditions**). Тобто, перевірці, відображаються відповідні повідомлення про помилки чи ні.

Переваги

- Ключові переваги функціонального тестування:
 - функціональне тестування ПЗ повністю імітує фактичне використання системи;
 - дозволяє своєчасно виявити системні помилки ПЗ і, тим самим, уникнути безліч проблем при роботі з ним надалі;
 - економія, за рахунок виправлення помилок на більш ранньому етапі життєвого циклу ПЗ.

Напрямки функціонального тестування

- Регресійне тестування : тестування функціональності продукту після виправлення помилок або реалізації нових функціональних можливостей
- Smoke-тестування : короткий цикл тестів для виявлення правильної роботи основних функцій додатка.
- Забезпечення тестового покриття : оцінка щільності покриття системи тестами.
- Тестування зручності використання : визначення ступеня зручності використання, зрозумілості та привабливості продукту, що розробляється.

Програми на функціональне тестування:

- Selenium – популярний інструмент функціонального тестування з відкритим вихідним кодом



- QTP – дуже зручний інструмент функціонального тестування HP.



- JUnit – Використовується в основному для додатків Java, і це може бути використано в тестуванні пристроїв і систем



- IBM Rational Functional Tester – є автоматизованим інструментом функціонального тестування і регресійного тестування. Це програмне забезпечення надає можливості автоматизованого тестування для функціонального, регресійного, графічного інтерфейсу і тестування, керованого даними.



IBM Rational Functional Tester

переваги

- Повторне використання: тести можуть миттєво працювати з різними версіями програми, що допомагає скоротити час, проведений під час регресійного тестування
- Консистенція: у кожному тесті, який виконується з скрипту RFT, такі ж кроки будуть проходити кожного разу.
- Продуктивність: автоматизоване тестування є швидким і може виконуватися в будь-який час без необхідності додаткових ресурсів.

Ручне та автоматизоване тестування

■ Ручне тестування:

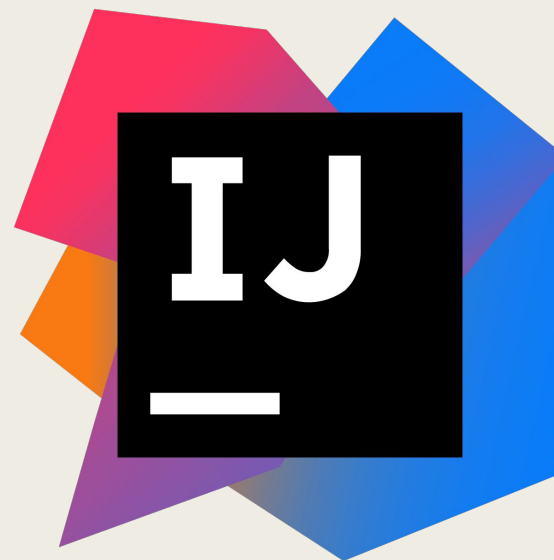
При ручному тестуванні (manual testing) тестувальники вручну виконують тести, не використовуючи ніяких засобів автоматизації. Ручне тестування - самий низькорівневий і простий тип тестування, які не потребують великої кількості додаткових знань.

■ Автоматизоване тестування:

Автоматизоване тестування передбачає використання спеціального програмного забезпечення (крім тестованого) для контролю виконання тестів і порівняння очікуваного фактичного результату роботи програми.

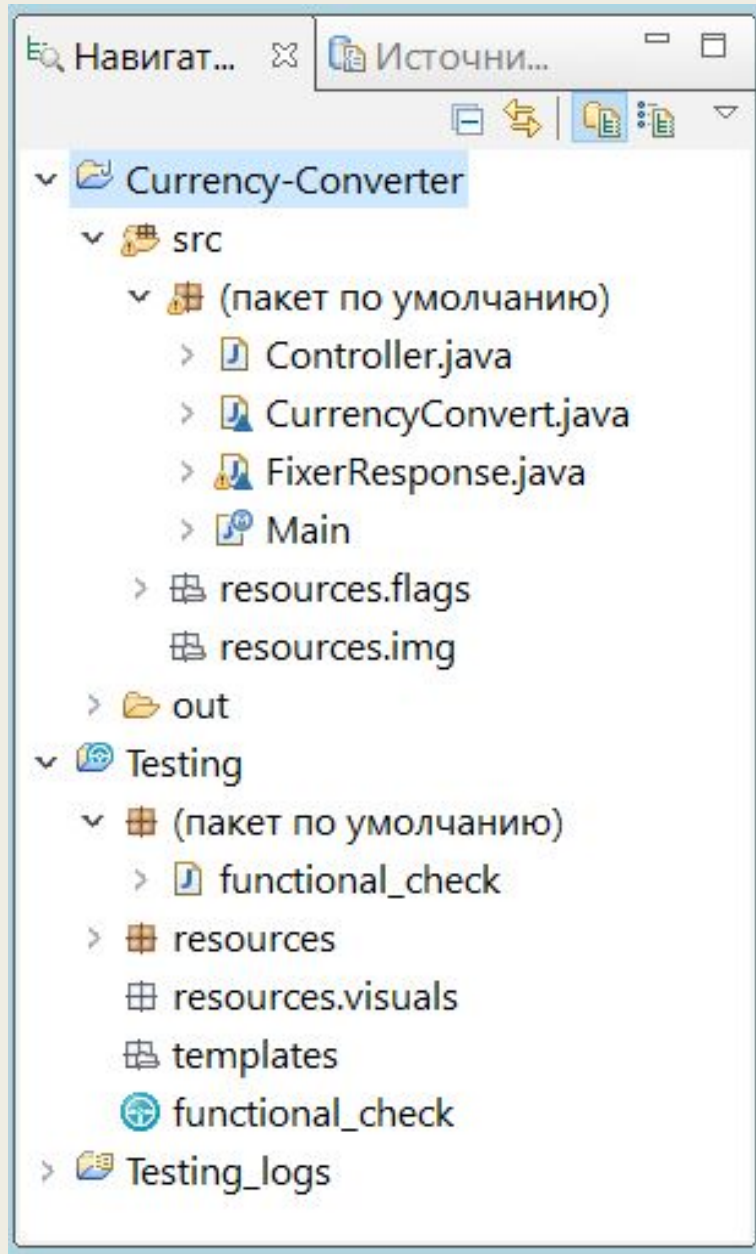
Середовище розробки

- Середовище розробки (IDE), яке необхідне для реалізації прикладних задач та написанню сценаріїв тестування, на якій використовувалась мова програмування Java - IntelliJ IDEA.

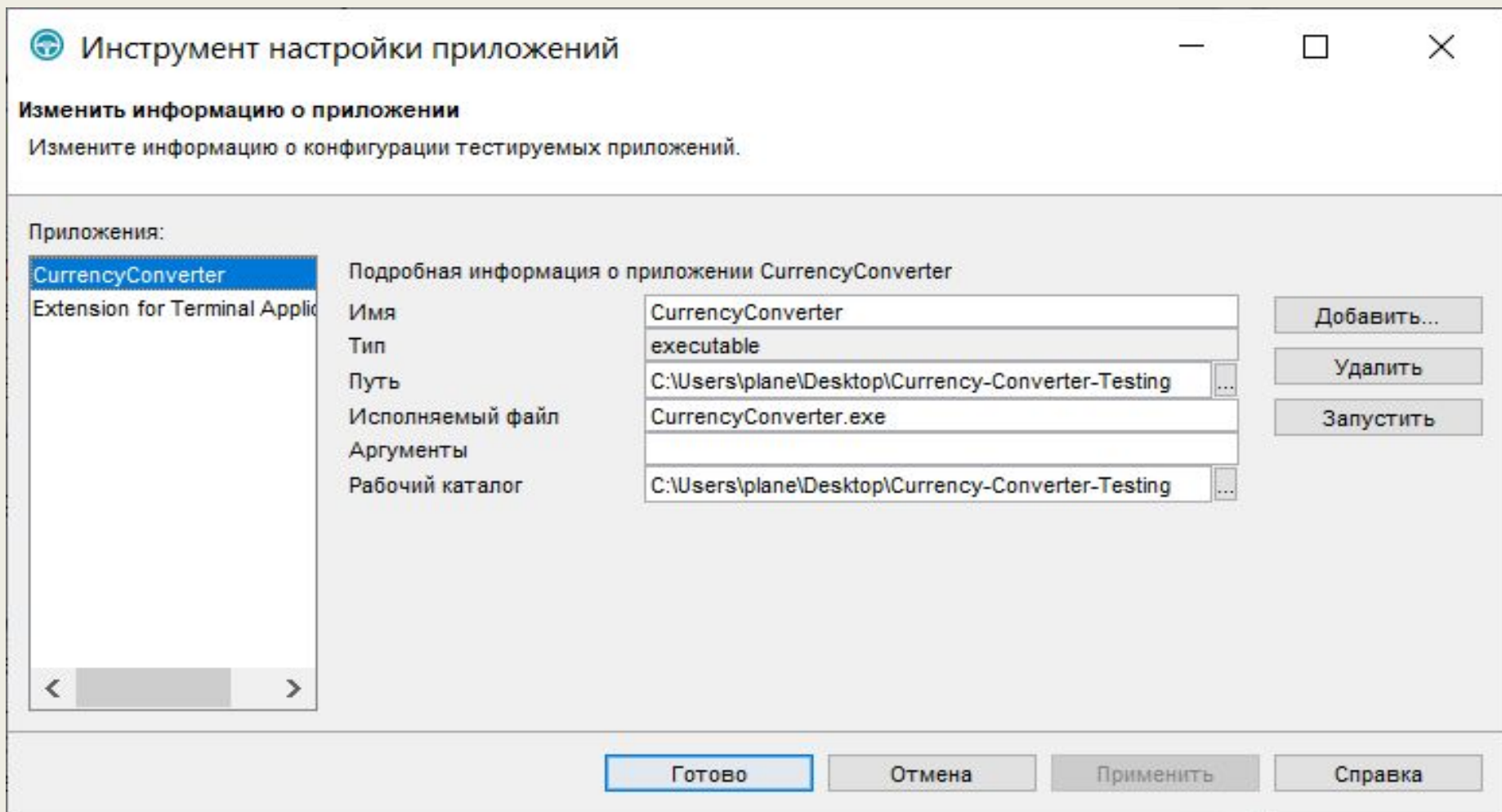




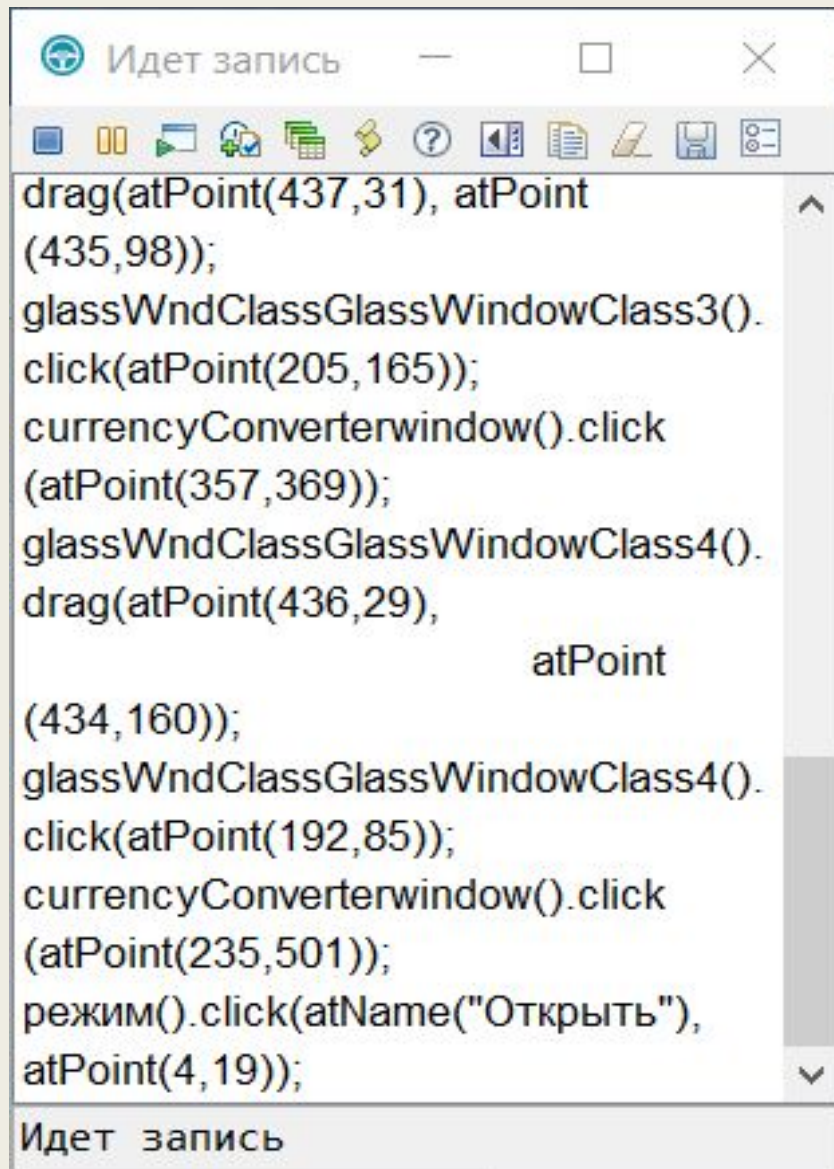
Зовнішній вигляд тестуємого додатку «Калькулятор валют». Написано на JavaFX.



Ієрархія класів та файлів додатку у які входять: класи з Java та написані тести



Инструмент налаштування додатку.



```
drag(atPoint(437,31), atPoint
(435,98));
glassWndClassGlassWindowClass3().
click(atPoint(205,165));
currencyConverterwindow().click
(atPoint(357,369));
glassWndClassGlassWindowClass4().
drag(atPoint(436,29),
                atPoint
(434,160));
glassWndClassGlassWindowClass4().
click(atPoint(192,85));
currencyConverterwindow().click
(atPoint(235,501));
режим().click(atName("Открыть"),
atPoint(4,19));
```

Идет запись

Всі дії користувача записуються інструментом RFT, котрий після завершення запису, перетворює сценарій у скрипт.

Currency-Converter-Testing - Функциональное тестирование - Testing/functional_check.java - IBM

Файл Изменить Исходный код Рефакторинг Навигация Поиск Проект Сценарий Настро

Currency Converter

Enter Amount
10

Франция

Ямайка

CONVERT

Воспроизведение

functional_check 51
currencyConverterwindow().click

check.java Controller.java

Generated : 27 мая 2019 г. 2:56:16
Description : Functional Test Script
Original Host : WinNT Version 10.0 Build 17763 ()

@since 2019/05/27
@author Vladislava

```
public void testMain(Object[] args)

    startApp("CurrencyConverter");

    // Window: java.exe: Currency Converter
    currencyConverterwindow().inputKeys("{Num1}{Num0}");
    currencyConverterwindow().click(atPoint(357,310));

    // Window: java.exe:
    glassWndClassGlassWindowClass3().drag(atPoint(443,32),atPoint(448,115));
    glassWndClassGlassWindowClass3().click(atPoint(245,97));

    // Window: java.exe: Currency Converter
    currencyConverterwindow().click(atPoint(336,366));

    // Window: java.exe:
    glassWndClassGlassWindowClass4().drag(atPoint(447,27), atPoint(447,150));
    glassWndClassGlassWindowClass4().click(atPoint(141,171));

    // Window: java.exe: Currency Converter
    currencyConverterwindow().click(atPoint(201,491));
    currencyConverterwindow().click(atPoint(335,359));
    currencyConverterwindow().click(atPoint(97,239));
    currencyConverterwindow().inputChars("p");
    currencyConverterwindow().click(atPoint(213,498));
    currencyConverterwindow().click(atPoint(137,218));
    currencyConverterwindow().inputKeys("{BKSP}{Num.}{Num5}");
    currencyConverterwindow().click(atPoint(206,497));
    currencyConverterwindow().click(atPoint(336,350));
    currencyConverterwindow(ANY,MAY_EXIT).click(CLOSE_BUTTON);
}
```

Запись Инт. вставка 30:41

Відтворення
записаного скрипту.

Browser window showing a test protocol report for 'functional_check'.

Address bar: C:\Users\plane\Desktop\Currency-Converter-Testing\Testing_lo

Протокол: functional_check

Ошибки <Нет>	27 мая 2019 г. 20:28:25 EEST	Запуск сценария [functional_check]
		<ul style="list-style-type: none">• <i>line_number</i> = 1• <i>script_iter_count</i> = 0• <i>script_name</i> = functional_check• <i>script_id</i> = functional_check.java
Предупреждения <Нет>	27 мая 2019 г. 20:28:25 EEST	Запуск приложения [CurrencyConverter]
		<ul style="list-style-type: none">• <i>name</i> = CurrencyConverter• <i>line_number</i> = 32• <i>script_name</i> = functional_check• <i>script_id</i> = functional_check.java• <i>startapp_type</i> = EXECUTABLE• <i>startapp_executable</i> = CurrencyConverter.exe• <i>startapp_working_directory</i> = C:\Users\plane\Desktop\Currency-Converter-Testing• <i>startapp_arguments</i> =
Точки проверки <Нет>	27 мая 2019 г. 20:28:45 EEST	Пройдено Завершение работы сценария [functional_check]
		<ul style="list-style-type: none">• <i>script_name</i> = functional_check• <i>script_id</i> = functional_check.java

Звіт тестування додатку