НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
“КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ім. І. СІКОРСЬКОГО”
Теплоенергетичний факультет
Кафедра автоматизації проектування енергетичних процесів та систем

Магістерська дисертація на тему: “Інструментальні засоби прискорення тестування мобільних додатків”

Виконала:
студентка групи ТВ-з71мп
Зубарчук Олександра Дмитрівна
Керівник:
Варава Іван Андрійович

Київ - 2018

Актуальність

Основна перевага автоматизації тестування мобільних додатків - скорочення витрат на випробування програми після її модернізації та позбавлення при перевірках людського фактору. Проте, досить часто, швидкість виконання автоматизованих тестів, бажає бути кращою.

Більшість великих компаній, що розробляють мобільні додатки, змушені запускати тести з вечора, щоб побачити результат регресійного тестування зранку. Така кількість часу, витрачена на тестування, не завжди є доступною, особливо, якщо необхідно терміново виправити помилку в роботі мобільного додатку. За таких обставин, компанії, що розробляють мобільні додатки на маленькі або середні масштаби, нехтують прогонкою автоматизованих тестів.

Саме тому, задача прискорення виконання автоматизованих тестів мобільних додатків є дуже актуальною сьогодні.

Мета та задачі

На основі наявних проблем була визначена **мета роботи**: створення методології, використовуючи наявні інструментальні та програмні засоби, для прискорення тестування мобільних додатків.

Задачі:

1. Вивчити та проаналізувати літературу з проблем прискорення автоматизованого тестування;
2. Зробити аналіз засобів автоматизованого тестування;
3. Об’єднати та модифікувати доступні інструментальні засоби для прискорення тестування мобільних додатків;
4. На основі обраних засобів, розробити методику для прискорення тестування мобільних додатків;
5. Розробити автоматизовані тести за створеною методикою;

Переваги та недоліки автоматизації тестування мобільних додатків

+ Повторюваність.
+ Швидкість виконання.
+ Мінімальні затрати часу на аналіз результатів проведеного тестування.
+ Формування та розповсюдження звітів про результати тестування.
+ Виконання без втручання людини.

- Затрати часу на розробку автоматизованого тестування.
- Пропуск дрібних помилок.

Особливості автоматизації тестування мобільних додатків

1. Наявність доступу до інструменту розробника мобільного додатку, який тестується.
2. Наявність .app (iOS) чи .apk (Android) файлу додатку.
3. Перевірка версії ОС пристрою, на якому запускаються автотести.
4. Сценарійність тестування.

Інструментальні засоби автоматизації тестування мобільних додатків

Для автоматизації тестування потрібно замінити тестувальника на інструменти, які вміють взаємодіяти з одним або декількома інтерфейсами програми. Також будуть потрібні утиліти для запуску і формування набору тестів. Разом всі ці інструменти називаються стеком автотестування та поділені на чотири групи:

- драйвери
- надбудови
- фреймворки
- комбайни

Драйвери та надбудови для мобільного тестування

Провівши аналіз доступних драйверів для тестування мобільних додатків, для подальшої роботи було обрано Appium Driver. Порівняльну характеристика драйверів наведено в таблиці 1. Appium також є і надбудовою, що дозволяє використовувати один і той самий код для тестів на iOS і Android.

	підтримка Android	підтримка iOS	написання тестів на Java	зручність використання	можливість роботи з .apk \ .app файлами
UIAutomator	+	-	-	+	-
Espresso	+	-	+	-	-
XCUITest	-	+	-	-	-
Appium	+	+	+	+	+

Таблиця 1 - Порівняльна характеристика драйверів для тестування мобільних додатків

Фреймворки тестування мобільних додатків

Серед наявних фреймворків для тестування було обрано TestNG. Порівняльна характеристика наведена в таблиці 2.

	Driver	Мова програмування	Підтримка анотації кроків	Параметризація
XUnit	Web Driver	Java / C#	+	-
Cucumber	Web Driver	PHP	-	-
TestNG	Selenium / Appium Driver	Java	+	+
JUnit	Selenium	Java	-	+

Таблиця 2 - Порівняльна характеристика фреймворків тестування мобільних додатків

Комбайни для тестування мобільних додатків

Комбайни - ще одна група утилітів, що використовуються для автоматизації тестування мобільних додатків, які поєднують в собі і фреймворки, і драйвери (причому не тільки мобільні), і навіть можливості розробки.

Проте, комбайни не дають можливості для багатопотокового запуску автоматизованих тестів, а також мають дуже обмежений доступ до пристроїв, тому не використані в подальшій роботі.

Методика прискорення тестування мобільних додатків

Методика являє собою сукупність кроків, які необхідно виконати, аби проект з автоматизованими тестами, які розробляються, виконувався швидше. Вона передбачає:

1. Створення сценаріїв тестування мобільного додатку.
2. Пошук елементів сторінок у мобільному додатку.
3. Опис знайдених елементів в проекті тестування.
4. Групування елементів в окремі класи – об’єкти сторінок.
5. Використання анотацій кроків при програмуванні дій тестів.
6. Попередня ініціалізація об’єктів сторінок безпосередньо перед початком тесту або тестового сценарію.
7. Використання багато-потоковості при запуску автоматизованих тестів.
8. Запуск автоматизованих тестів на виділеному сервері.

Методика прискорення тестування мобільних додатків

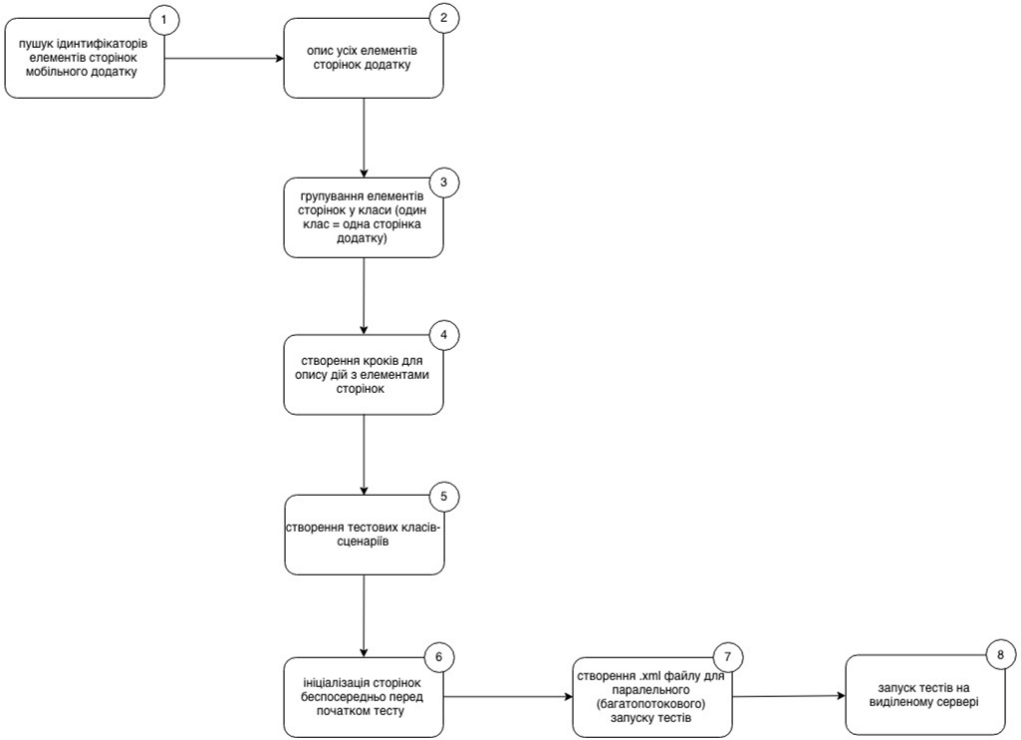

Схема 1 - Методика прискорення автоматизованого тестування мобільних додатків

Попередній опис елементів сторінок мобільного додатку

Пошук ідентифікатора елементів виконано за допомогою Appium Inspector. Приклад наведено на рисунку 1.

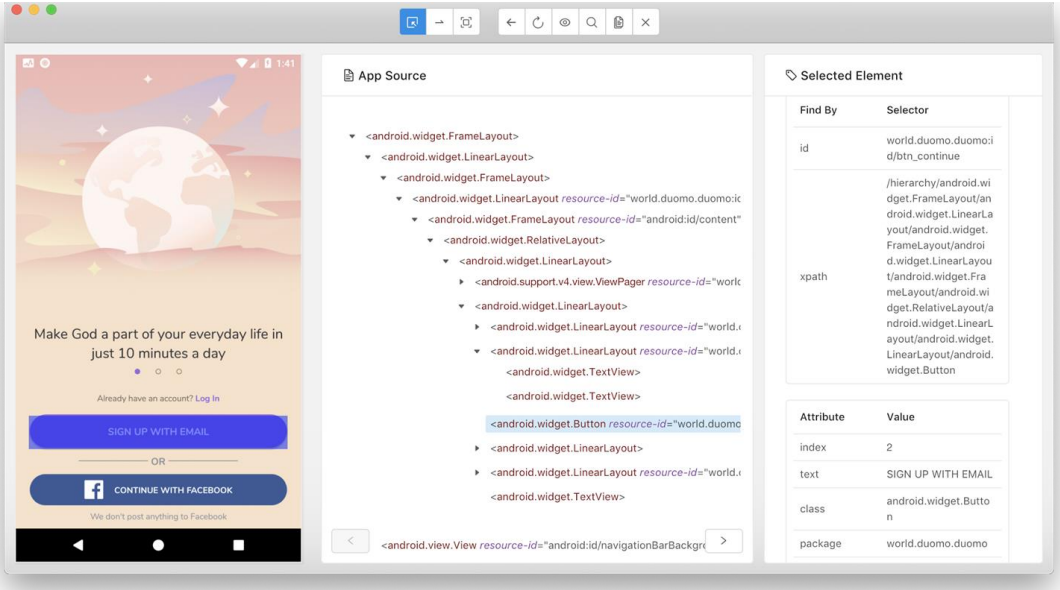

Рисунок 1 - Інформація про елемент сторінки мобільного додатку в Appium Inspector

Попередній опис елементів сторінок мобільного додатку

Приклад опису елементів сторінки в проекті тестування наведено на рисунку 2. Для опису елементів кожної окремої сторінки створюється клас, в конструкторі якого кожному елементу сторінки надається ідентифікатор, що відповідає даному елементу додатку.

```
public class LandingPageObject {

   public WebElement signUpBtn;
   public WebElement logInLink;
   public WebElement continueWithFb;
   public WebElement textView;

   public LandingPageObject(AppiumDriver driver) {
      signUpBtn = driver.findElementById("btn_continue");
      logInLink = driver.findElementById("ll_signin");
      continueWithFb = driver.findElementById("ll_facebook");
      textView = driver.findElementById("text_view");
   }
```

Рисунок 2 - клас сторінки Landing Page Object

Створення кроків виконання сценарію тестування мобільного додатку

Тестовий випадок/ситуація (Тест Кейс /Test Case) - це артефакт, що описує сукупність кроків, конкретних умов та параметрів, необхідних для перевірки реалізації функції, що тестується чи її частини. При створенні методів-кроків автоматизованого тесту, тест має чітку структуру дій: Action > Expected Result > Test Result.

Для опису кроків створюється окремий клас (для кожного сценарію окремий), в якому створюються методи (назва методу - назва кроку) з описом машинних дій, які необхідно зробити для того, аби виконати крок

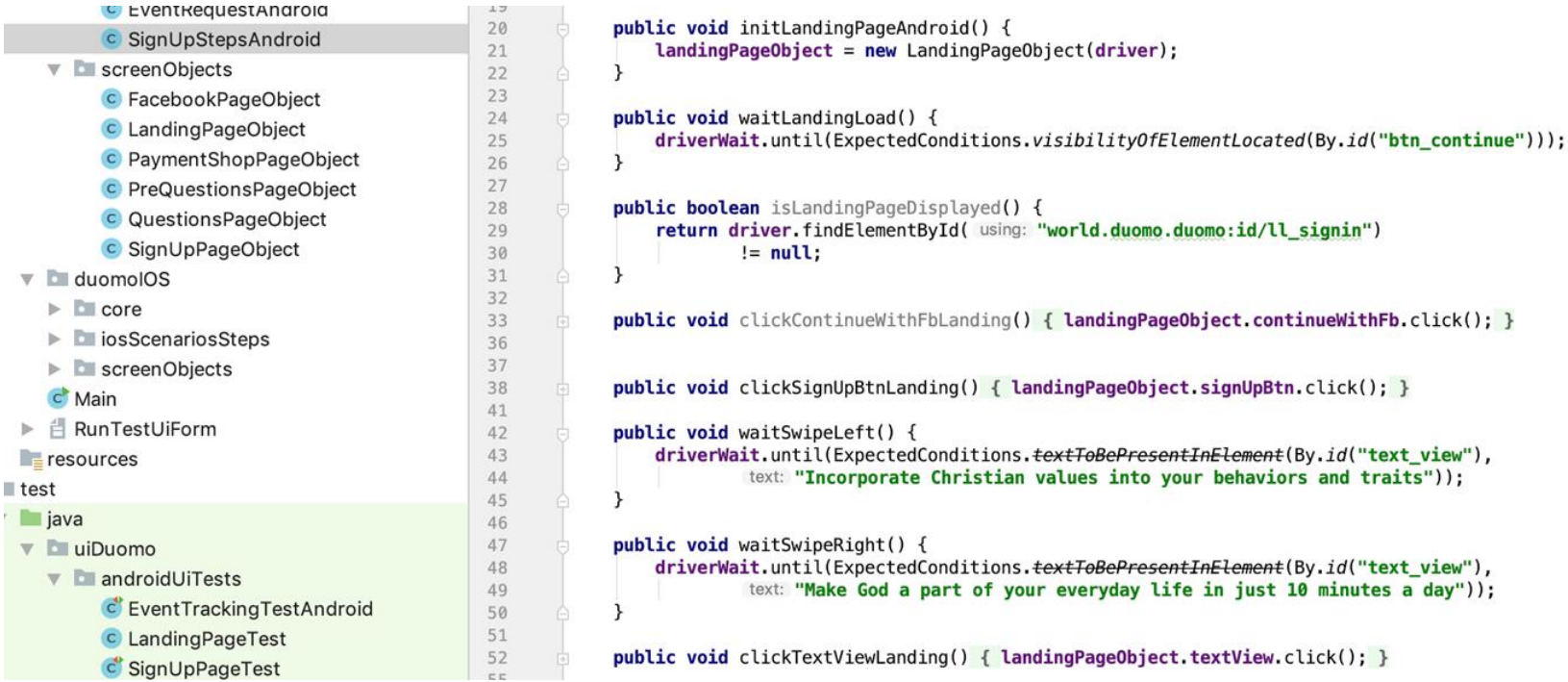

Рисунок 3 - Опис кроків для виконання тестового сценарію

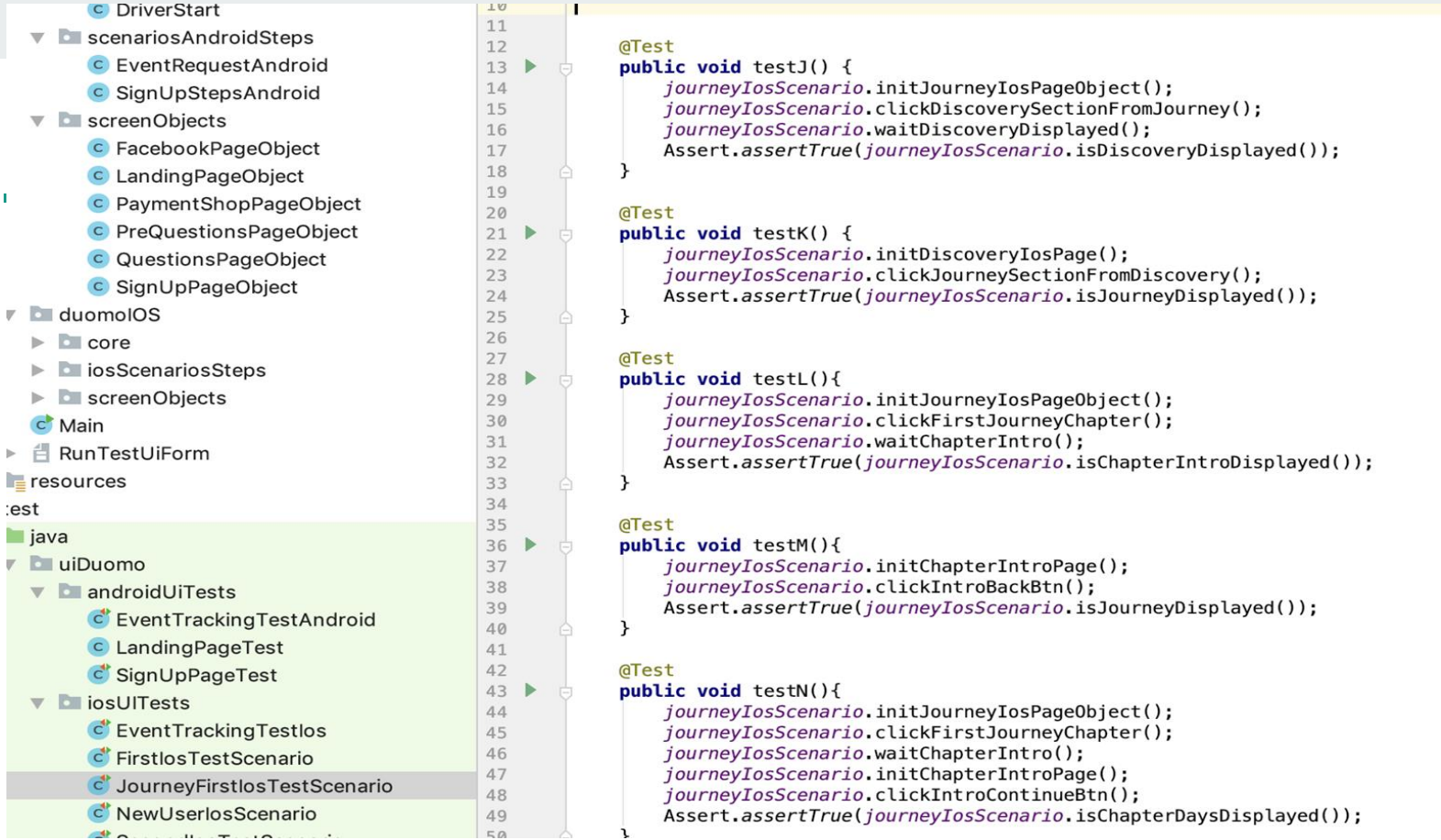

Рисунок 3.1 - автоматизовані сценарії тестування з попередньо описаними кроками

Allure Report

Використання Allure Report допомагає побачити згруповані результати проведеного тестування, На рисунку 4 зображено звіт про тестування без використання Allure, на рисунку 5 - з його використанням. Очевидно, використання Allure Report допомагає швидше побачити проблеми, якщо вони наявні та запевнитись у тому, що все гаразд, за відсутності помилок.

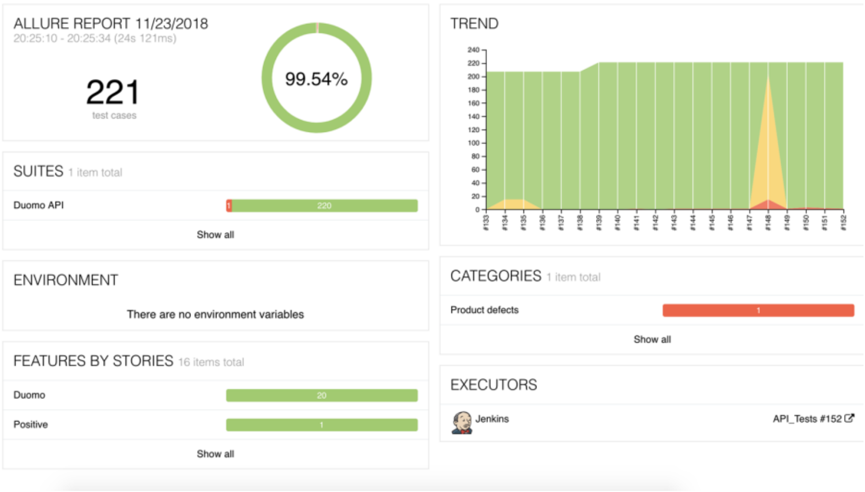

Рисунок 4 - Звіт про тестування в Allure Report

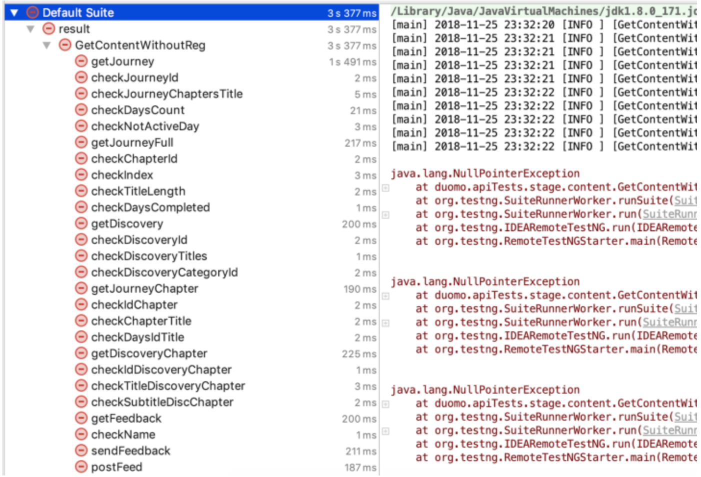

Рисунок 5 - Звіт про тестування в IntellIj IDEA

Багатопотокове виконання тестів

Мова програмування Java дає можливість запускати процеси у декілька потоків. Завдяки цьому, є можливість запустити виконання різних тестових сценарії у декілька потоків, що значно зменшує час виконання автоматизованих тестів. Для того, щоб розділити сценарії тестування на потоки, необхідно створити .xml файл запуску тестів, де буде описано, які саме сценарії необхідно запускати, які з них розділити на потоки, я які поєднати. Розділення сценарії тестуван

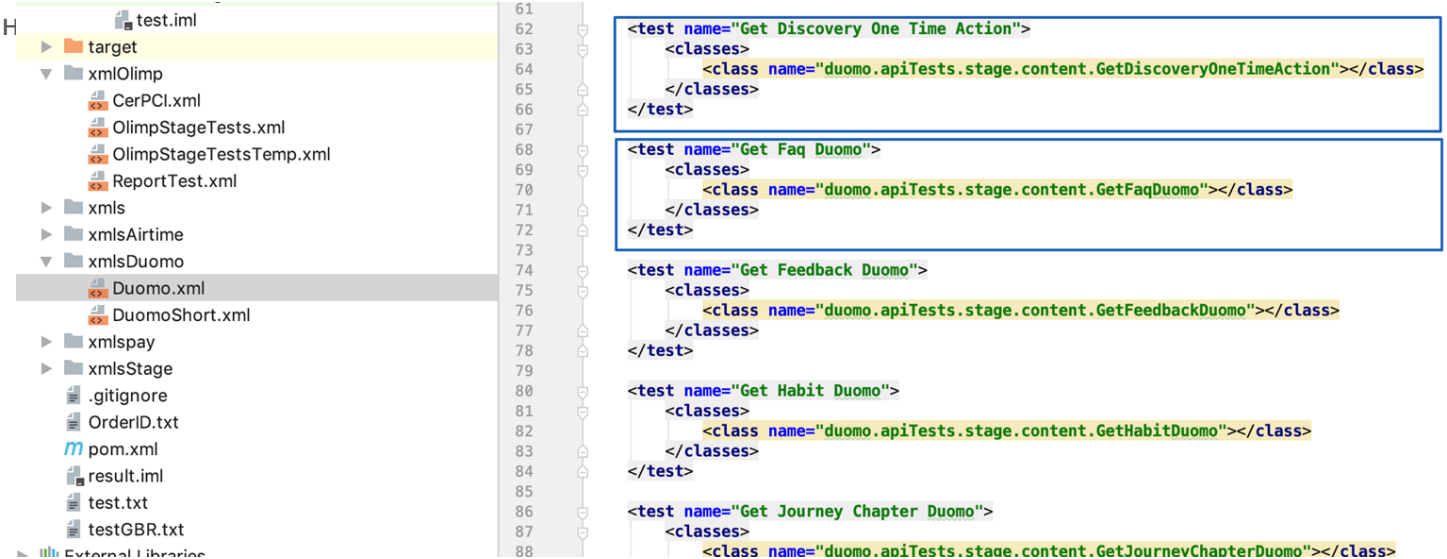

Рисунок 6 - Потоки виконання автоматизованих тестів у файлі .xml

Результати використання методики

За результатами запуску системи автоматизованого тестування на прикладі мобільного додатку “Duomo”, що була розроблена на основі створеної методики прискорення тестування можна зазначити, що час виконання автоматизованих тестів було скорочено більше, ніж в чотири рази з сорока дев’яти хвилин до одинадцяти. При цьому кількість та якість виконання тестів залишились незмін

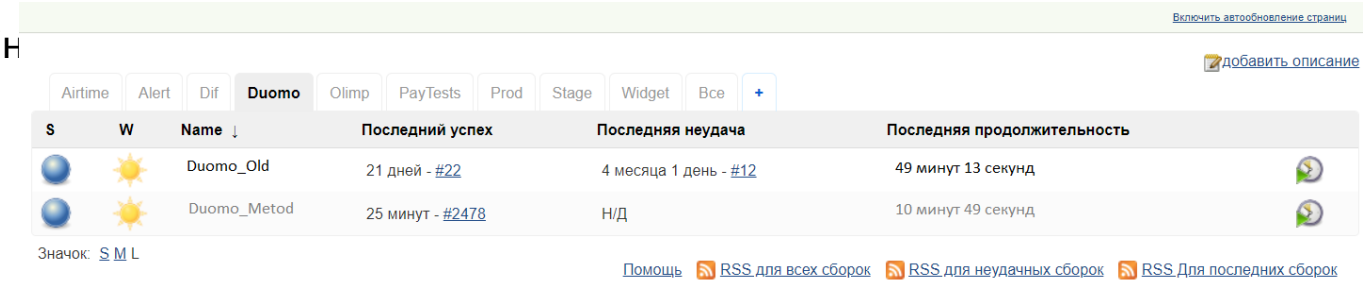

Рисунок 7 - Порівняльний час виконання автоматизованих тестів

Висновки

На основі проведеного аналізу було обрано методи та інструментальні засоби для створення методики прискорення автоматизованого тестування мобільних додатків.

В роботі представлено детальний опис методики та інструментальних засобів, необхідних для її використання. Методика може бути використана при тестуванні будь-яких мобільних додатків.

Було виконано практичну перевірку результативності методики. За результатами перевірки, можна стверджувати, що методика працює та дійсно дає змогу прискорити тестування програмного продукту.

Задачі, поставлені на початку даної роботи, виконані в повному обсязі. На даний момент методика прискорення тестування та система автоматизованого тестування мобільного додатку «Duomo» на її основі впроваджена в роботу ТОВ «Айті Ленд», на чиє замовлення була виконана робота.

Дякую за увагу!