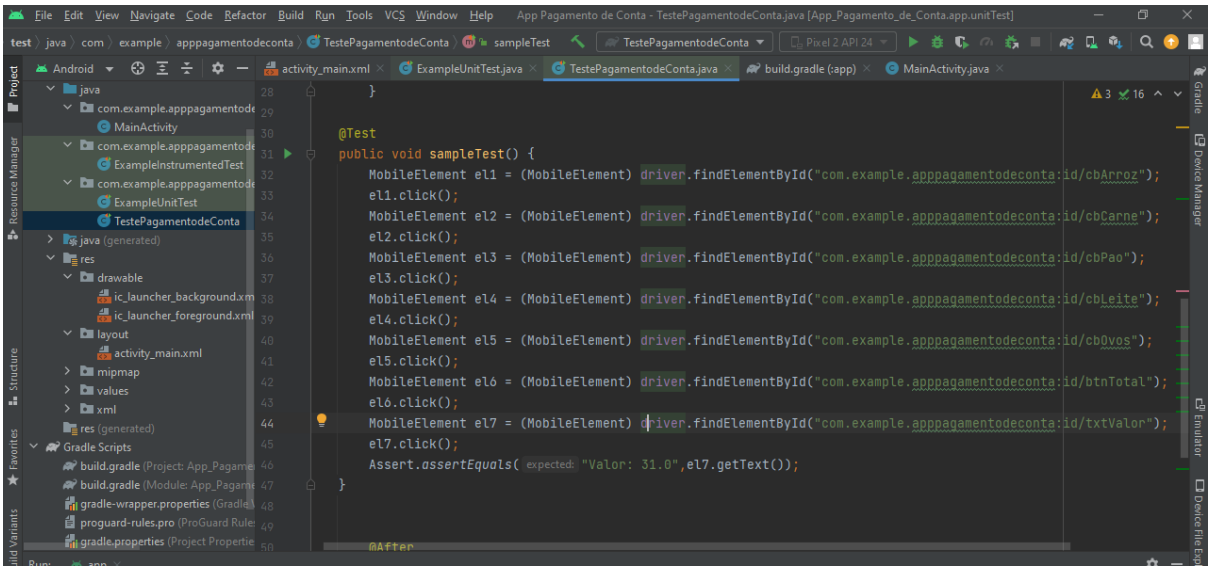
Avaliação - Teste Funcional

nome: Andréia de Oliveira Araújo

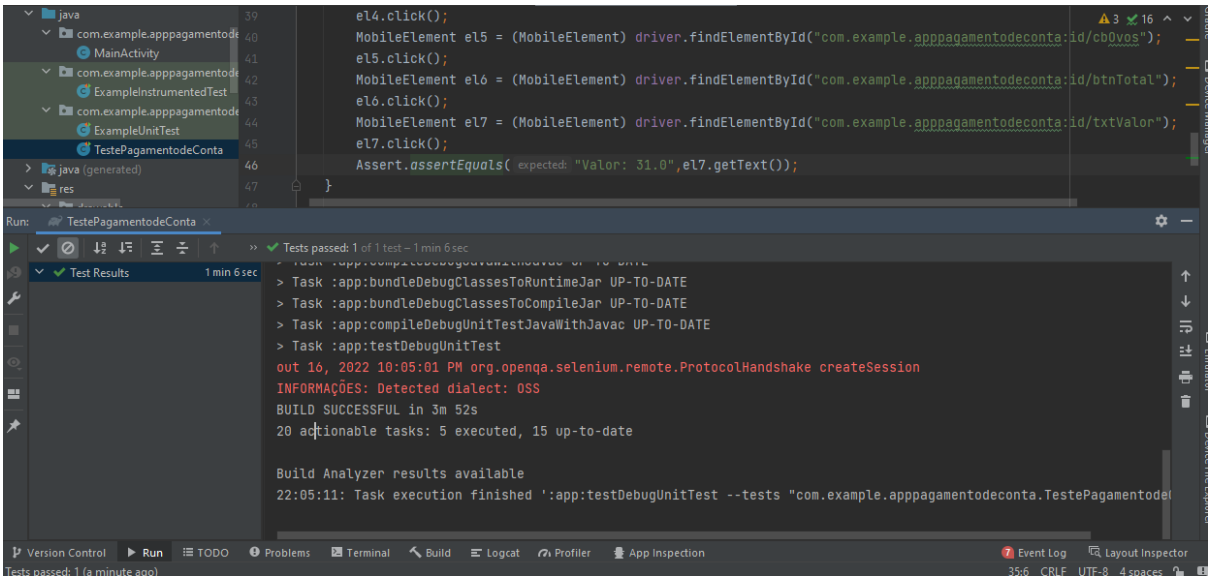
1 .Crie um caso de teste para qualquer APP desde que não seja os utilizados nos slides.
2.Faça os teste automatizados utilizando o APPIUM.

R= Print classe teste, com os testes realizados no Appium.



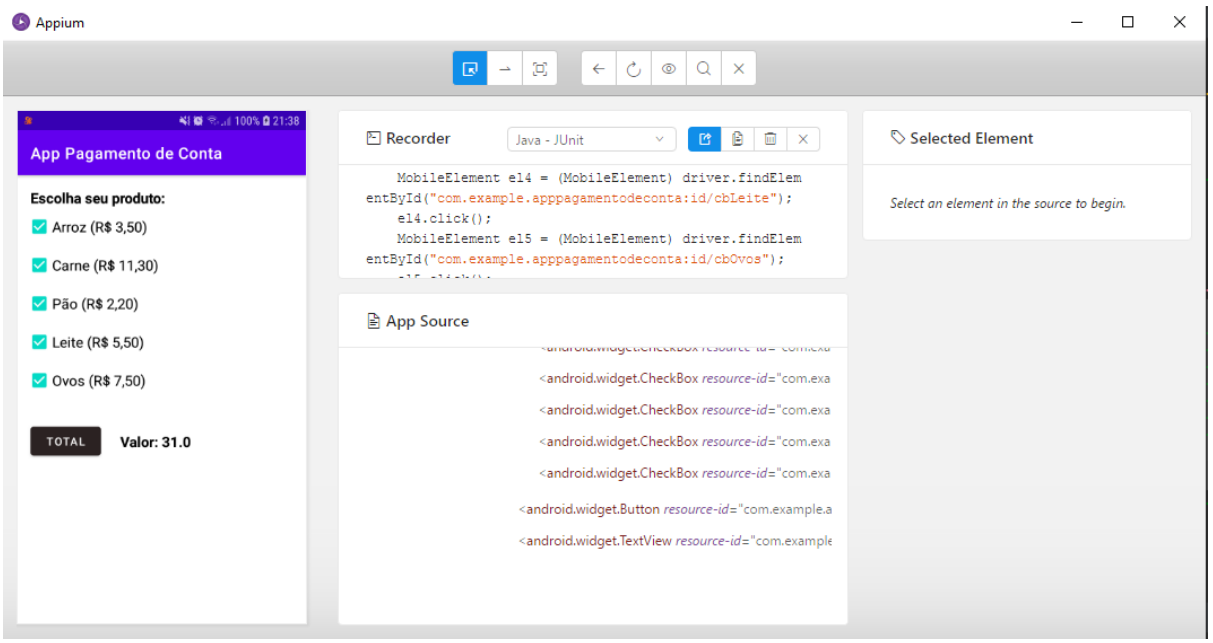
```
@Test
public void sampleTest() {
    MobileElement el1 = (MobileElement) driver.findElementById("com.example.apppagamentodeconta:id/cbArroz");
    el1.click();
    MobileElement el2 = (MobileElement) driver.findElementById("com.example.apppagamentodeconta:id/cbCarne");
    el2.click();
    MobileElement el3 = (MobileElement) driver.findElementById("com.example.apppagamentodeconta:id/cbPao");
    el3.click();
    MobileElement el4 = (MobileElement) driver.findElementById("com.example.apppagamentodeconta:id/cbLeite");
    el4.click();
    MobileElement el5 = (MobileElement) driver.findElementById("com.example.apppagamentodeconta:id/cbOvos");
    el5.click();
    MobileElement el6 = (MobileElement) driver.findElementById("com.example.apppagamentodeconta:id/btnTotal");
    el6.click();
    MobileElement el7 = (MobileElement) driver.findElementById("com.example.apppagamentodeconta:id/txtValor");
    el7.click();
    Assert.assertEquals( expected: "Valor: 31.0",el7.getText());
}
```

Print do teste sendo realizado com sucesso.



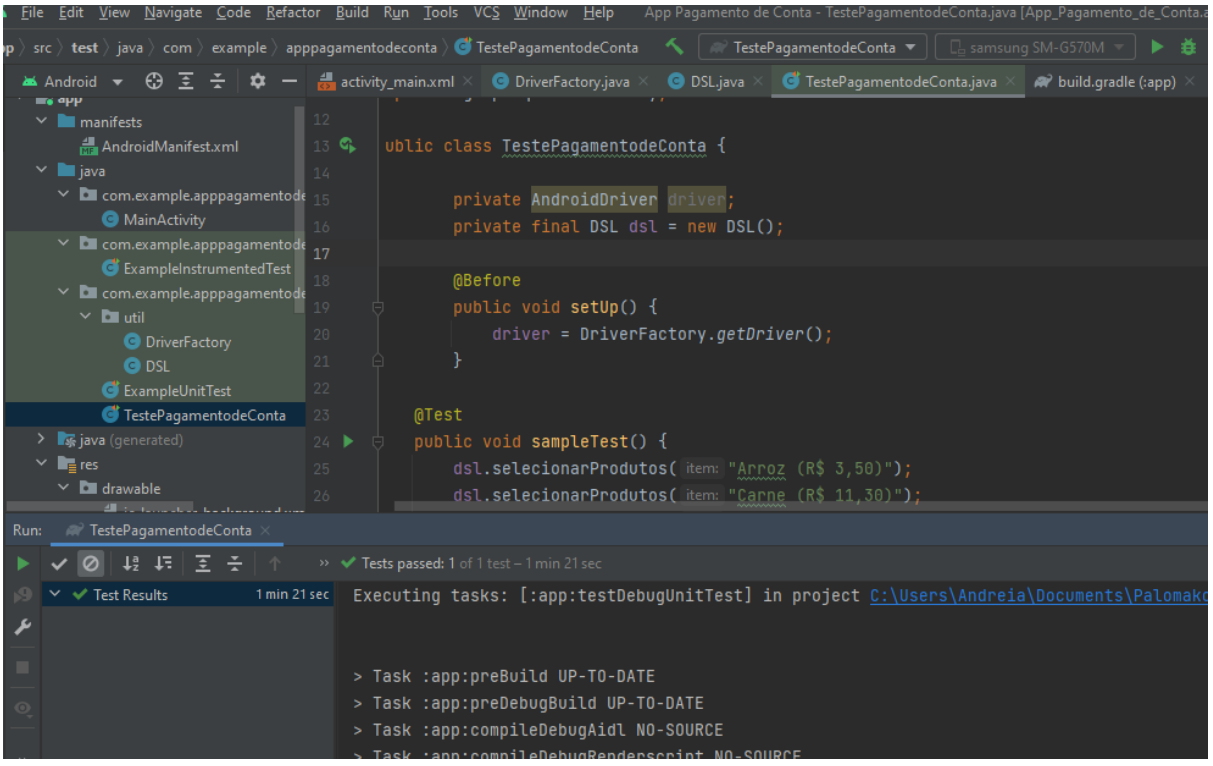
```
> Task :app:bundleDebugClassesToRuntimeJar UP-TO-DATE
> Task :app:bundleDebugClassesToCompileJar UP-TO-DATE
> Task :app:compileDebugUnitTestJavaWithJavac UP-TO-DATE
> Task :app:testDebugUnitTest
out 16, 2022 10:05:01 PM org.openqa.selenium.remote.ProtocolHandshake createSession
INFORMAÇÕES: Detected dialect: OSS
BUILD SUCCESSFUL in 3m 52s
20 actionable tasks: 5 executed, 15 up-to-date

Build Analyzer results available
22:05:11: Task execution finished ':app:testDebugUnitTest --tests "com.example.apppagamentodeconta.TestePagamentode
```

Print do teste feito no Appium

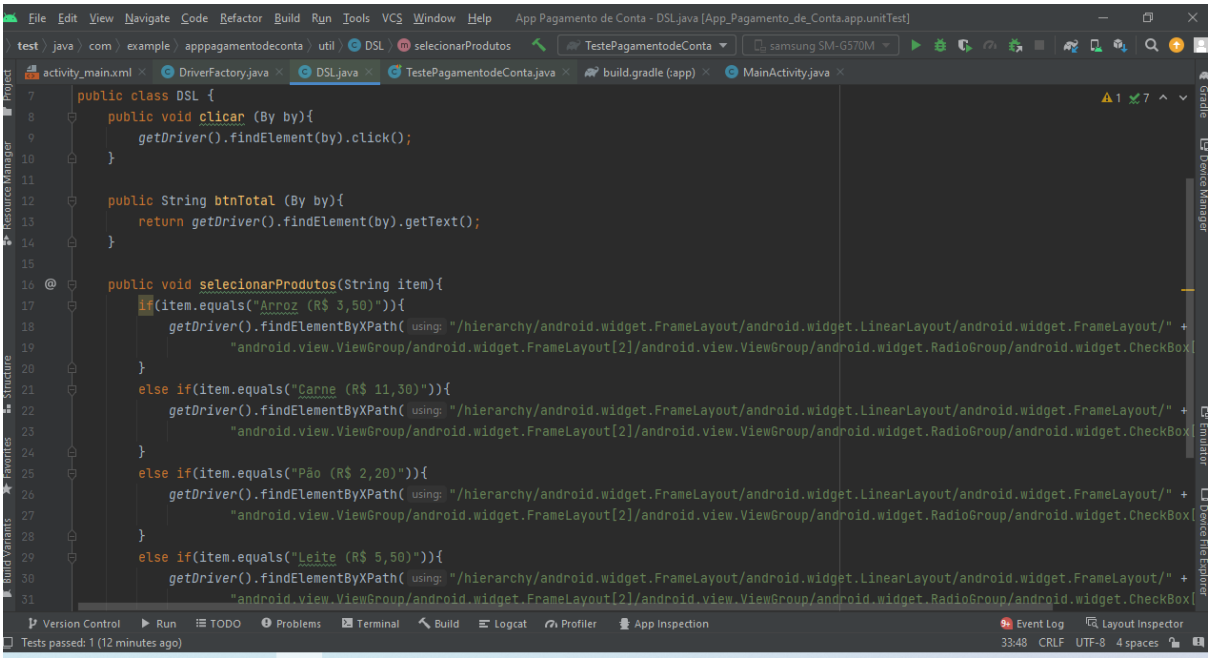


3.Melhore o código introduzindo DriverFactory e DSL

R= Print teste realizado com sucesso e as classes

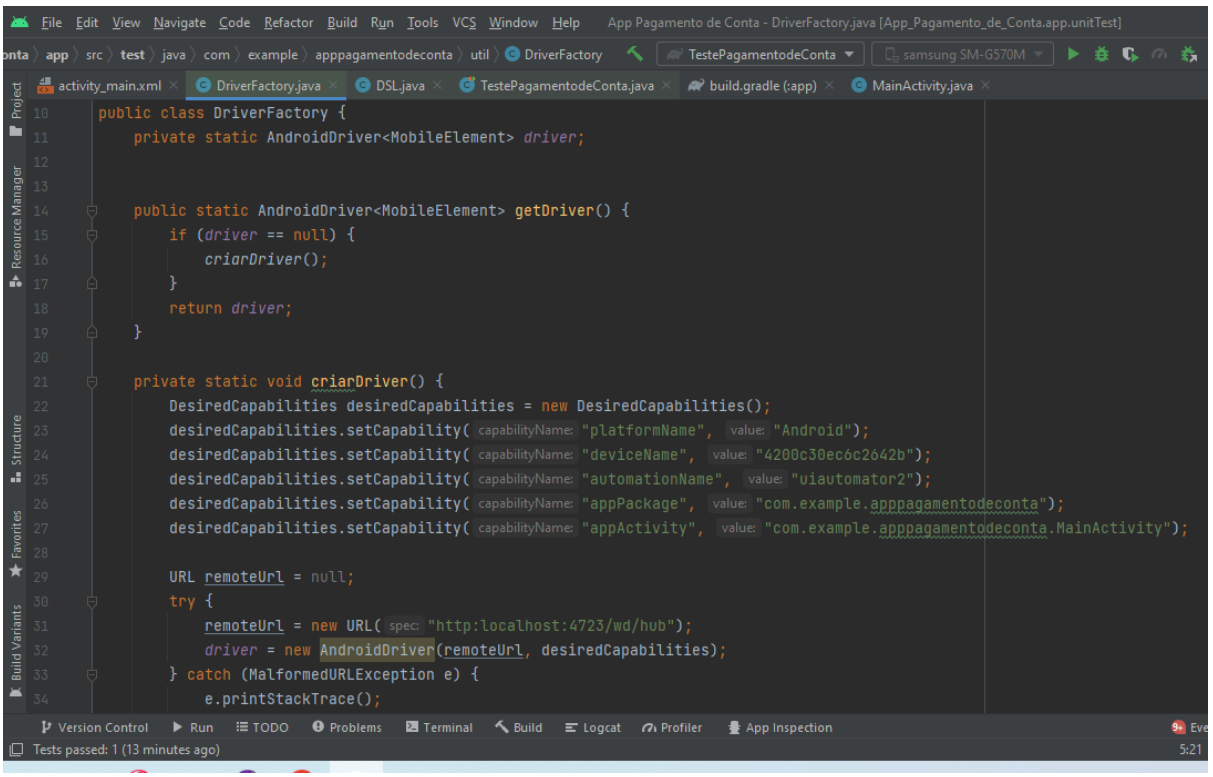


DSL



```
public class DSL {
    public void clicar (By by){
        getDriver().findElement(by).click();
    }

    public String btnTotal (By by){
        return getDriver().findElement(by).getText();
    }

    public void selecionarProdutos(String item){
        if(item.equals("Arroz (R$ 3,50)")){
            getDriver().findElementByXPath( using: "/hierarchy/android.widget.FrameLayout/android.widget.LinearLayout/android.widget.FrameLayout/" +
                    "android.view.ViewGroup/android.widget.FrameLayout[2]/android.view.ViewGroup/android.widget.RadioGroup/android.widget.CheckBox[
        }
        else if(item.equals("Carne (R$ 11,30)")){
            getDriver().findElementByXPath( using: "/hierarchy/android.widget.FrameLayout/android.widget.LinearLayout/android.widget.FrameLayout/" +
                    "android.view.ViewGroup/android.widget.FrameLayout[2]/android.view.ViewGroup/android.widget.RadioGroup/android.widget.CheckBox[
        }
        else if(item.equals("Pão (R$ 2,20)")){
            getDriver().findElementByXPath( using: "/hierarchy/android.widget.FrameLayout/android.widget.LinearLayout/android.widget.FrameLayout/" +
                    "android.view.ViewGroup/android.widget.FrameLayout[2]/android.view.ViewGroup/android.widget.RadioGroup/android.widget.CheckBox[
        }
        else if(item.equals("Leite (R$ 5,50)")){
            getDriver().findElementByXPath( using: "/hierarchy/android.widget.FrameLayout/android.widget.LinearLayout/android.widget.FrameLayout/" +
                    "android.view.ViewGroup/android.widget.FrameLayout[2]/android.view.ViewGroup/android.widget.RadioGroup/android.widget.CheckBox[
```

DriverFactory



```
public class DriverFactory {
    private static AndroidDriver<MobileElement> driver;


    public static AndroidDriver<MobileElement> getDriver() {
        if (driver == null) {
            criarDriver();
        }
        return driver;
    }

    private static void criarDriver() {
        DesiredCapabilities desiredCapabilities = new DesiredCapabilities();
        desiredCapabilities.setCapability( capabilityName: "platformName", value: "Android");
        desiredCapabilities.setCapability( capabilityName: "deviceName", value: "4200c30ec6c2642b");
        desiredCapabilities.setCapability( capabilityName: "automationName", value: "uiautomator2");
        desiredCapabilities.setCapability( capabilityName: "appPackage", value: "com.example.apppagamentodeconta");
        desiredCapabilities.setCapability( capabilityName: "appActivity", value: "com.example.apppagamentodeconta.MainActivity");

        URL remoteUrl = null;
        try {
            remoteUrl = new URL( spec: "http:localhost:4723/wd/hub");
            driver = new AndroidDriver(remoteUrl, desiredCapabilities);
        } catch (MalformedURLException e) {
            e.printStackTrace();
```