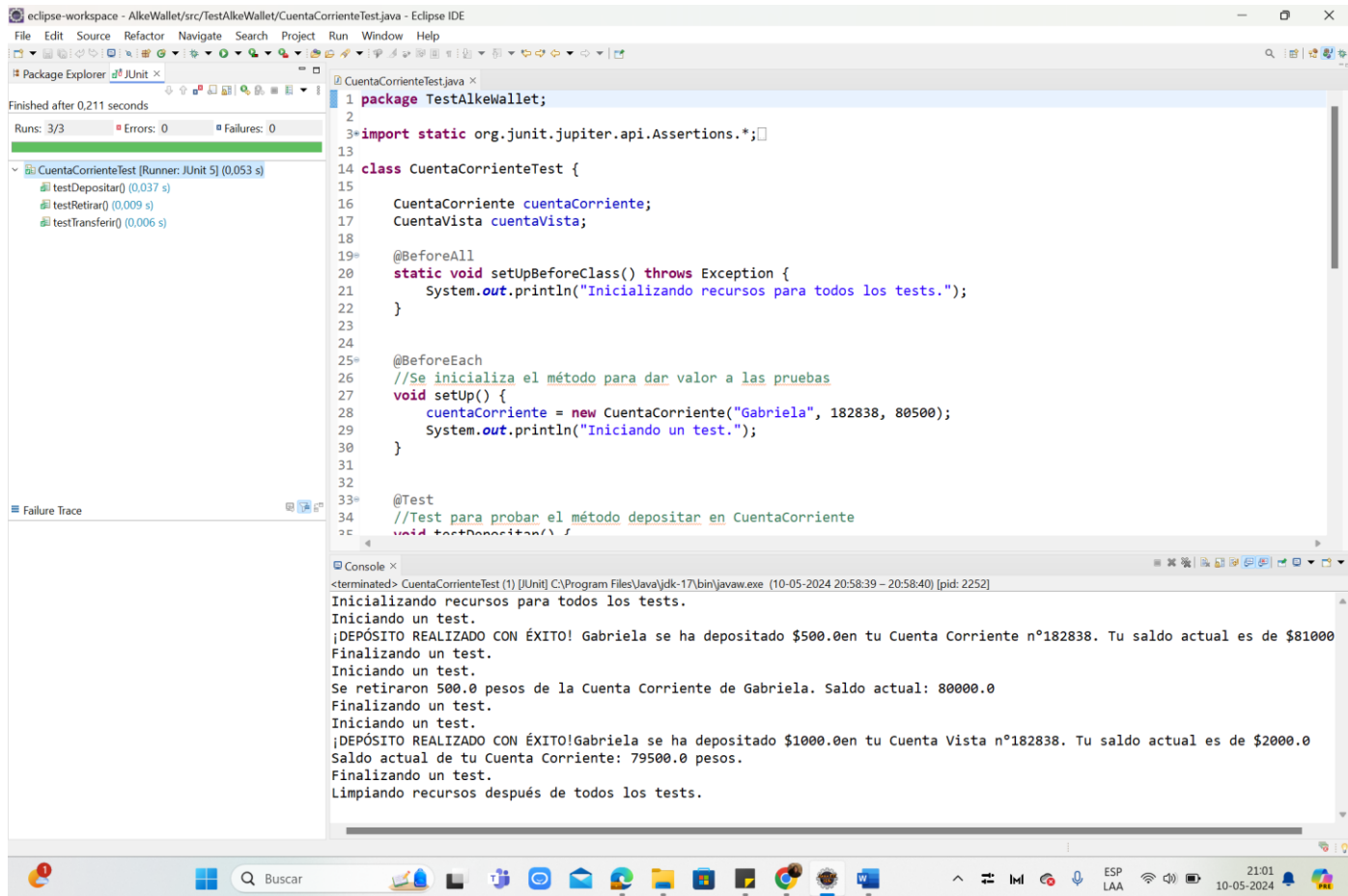


VISUALIZACIÓN DE LA DOCUMENTACIÓN

I. EVIDENCIA DE LAS PRUEBAS JUNIT 5

1. Clase CuentaCorriente → CuentaCorrienteTest



```
1 package TestAlkeWallet;
2
3 import static org.junit.jupiter.api.Assertions.*;
4
5 class CuentaCorrienteTest {
6
7     CuentaCorriente cuentaCorriente;
8     CuentaVista cuentaVista;
9
10    @BeforeAll
11    static void setUpBeforeClass() throws Exception {
12        System.out.println("Inicializando recursos para todos los tests.");
13    }
14
15    @BeforeEach
16    //Se inicializa el método para dar valor a las pruebas
17    void setUp() {
18        cuentaCorriente = new CuentaCorriente("Gabriela", 182838, 80500);
19        System.out.println("Iniciando un test.");
20    }
21
22    @Test
23    //Test para probar el método depositar en CuentaCorriente
24    void testDepositar() {
25
26    }
27
28    @Test
29    //Test para probar el método retirar en CuentaCorriente
30    void testRetirar() {
31
32    }
33
34    @Test
35    //Test para probar el método transferir en CuentaCorriente
36    void testTransferir() {
37
38    }
39
40 }
```

Finished after 0.211 seconds

Runs: 3/3 Errors: 0 Failures: 0

CuentaCorrienteTest [Runner: JUnit 5] (0.053 s)

- testDepositar() (0.037 s)
- testRetirar() (0.009 s)
- testTransferir() (0.006 s)

Failure Trace

Console

```
<terminated> CuentaCorrienteTest (1) [JUnit] C:\Program Files\Java\jdk-17\bin\javaw.exe (10-05-2024 20:58:39 - 20:58:40) [pid: 2252]
Inicializando recursos para todos los tests.
Iniciando un test.
¡DEPÓSITO REALIZADO CON ÉXITO! Gabriela se ha depositado $500.0en tu Cuenta Corriente n°182838. Tu saldo actual es de $81000
Finalizando un test.
Iniciando un test.
Se retiraron 500.0 pesos de la Cuenta Corriente de Gabriela. Saldo actual: 80000.0
Finalizando un test.
Iniciando un test.
¡DEPÓSITO REALIZADO CON ÉXITO! Gabriela se ha depositado $1000.0en tu Cuenta Vista n°182838. Tu saldo actual es de $2000.0
Saldo actual de tu Cuenta Corriente: 79500.0 pesos.
Finalizando un test.
Limpiando recursos después de todos los tests.
```

2. Clase CuentaVista → CuentaVistaTest

The screenshot shows the Eclipse IDE with the `CuentaVistaTest.java` file open. The code defines a test class with the following structure:

```
13 class CuentaVistaTest {
14     CuentaCorriente cuentaCorriente;
15     CuentaVista cuentaVista;
16
17     @BeforeAll
18     static void setUpBeforeClass() throws Exception {
19         System.out.println("Inicializando recursos para todos los tests.");
20     }
21
22     @BeforeEach
23     //Se inicializa el método para dar valor a las pruebas
24     void setUp() {
25         cuentaVista = new CuentaVista("Gabriela", 19772, 15000);
26         System.out.println("Iniciando un test.");
27     }
28
29     @Test
30     //Test para probar el método depositar en CuentaVista
31     void testDepositar() {
32         double cantidadDeposito = 5000;
33         double saldoEsperado = 20000;
34         cuentaVista.depositar(cantidadDeposito);
35         assertEquals(saldoEsperado, cuentaVista.getSaldo());
36     }
37 }
```

The Package Explorer on the left shows the test results for `CuentaVistaTest` (JUnit 5) with 3 runs, 0 errors, and 0 failures. The console output shows the following sequence of events:

```
<terminated> CuentaVistaTest (1) [JUnit] C:\Program Files\Java\jdk-17\bin\javaw.exe (10-05-2024 21:01:58 - 21:01:59) [pid: 18772]
Inicializando recursos para todos los tests.
Iniciando un test.
¡DEPÓSITO REALIZADO CON ÉXITO! Gabriela se ha depositado $5000.0en tu Cuenta Vista nº19772. Tu saldo actual es de $20000.0
Finalizando un test.
Iniciando un test.
Se retiraron 1000.0 pesos de la Cuenta Vista de Gabriela. Saldo actual: 14000.0
Finalizando un test.
Iniciando un test.
¡DEPÓSITO REALIZADO CON ÉXITO! Gabriela se ha depositado $1000.0en tu Cuenta Corriente nº12918. Tu saldo actual es de $6000.0
Saldo actual de tu Cuenta Vista: 14000.0 pesos.
Finalizando un test.
Limpiando recursos después de todos los tests.
```

3. Clase Dólar → DolarTest

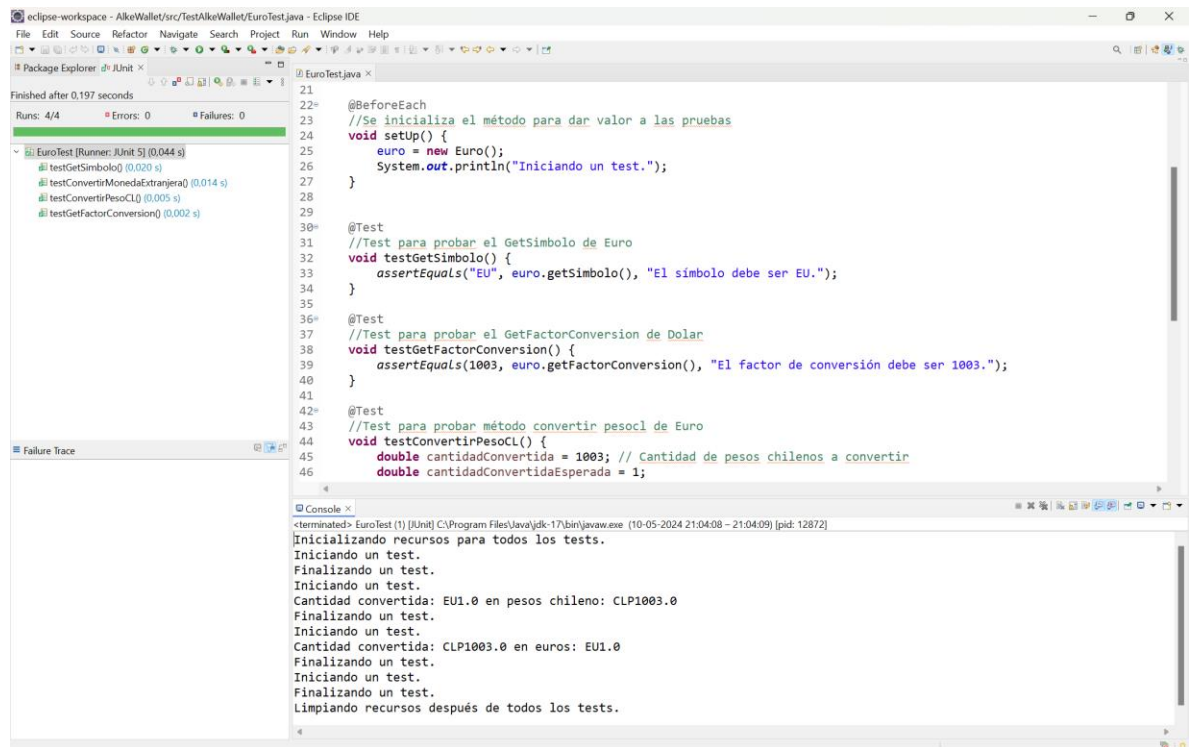
The screenshot shows the Eclipse IDE with the `DolarTest.java` file open. The code defines a test class with the following structure:

```
15 Dolar dolar;
16
17 @BeforeAll
18 static void setUpBeforeClass() throws Exception {
19     System.out.println("Inicializando recursos para todos los tests.");
20 }
21
22 @BeforeEach
23 //Se inicializa el método para dar valor a las pruebas
24 void setUp() {
25     dolar = new Dolar();
26     System.out.println("Iniciando un test.");
27 }
28
29 @Test
30 //Test para probar el GetSimbolo de Dolar
31 void testGetSimbolo() {
32     assertEquals("USD", dolar.getSimbolo(), "El simbolo debe ser USD.");
33 }
34
35 @Test
36 //Test para probar el GetFactorConversion de Dolar
37 void testGetFactorConversion() {
38     assertEquals(929, dolar.getFactorConversion(), "El factor de conversión debe ser 929.");
39 }
40 }
```

The Package Explorer on the left shows the test results for `DolarTest` (JUnit 5) with 4 runs, 0 errors, and 0 failures. The console output shows the following sequence of events:

```
<terminated> DolarTest (1) [JUnit] C:\Program Files\Java\jdk-17\bin\javaw.exe (10-05-2024 21:02:44 - 21:02:45) [pid: 19656]
Inicializando recursos para todos los tests.
Iniciando un test.
Finalizando un test.
Iniciando un test.
Cantidad convertida: USD1.0 en pesos chileno: CLP929.0
Finalizando un test.
Iniciando un test.
Cantidad convertida: CLP929.0 en dólares: USD1.0
Finalizando un test.
Iniciando un test.
Finalizando un test.
Limpiando recursos después de todos los tests.
```

4. Clase Euro → EuroTest



The screenshot shows the Eclipse IDE with the file `EuroTest.java` open. The code is a JUnit test class for the `Euro` class. It includes a `@Before` method for setup, a `@Test` method for `getSimbolo`, another `@Test` method for `getFactorConversion`, and a `@Test` method for `convertirPesoCL`. The console output shows the test execution results, including the initialization of resources, the execution of each test, and the final cleanup.

```
21
22 @Before
23 //Se inicializa el método para dar valor a las pruebas
24 void setUp() {
25     euro = new Euro();
26     System.out.println("Iniciando un test.");
27 }
28
29
30 @Test
31 //Test para probar el GetSimbolo de Euro
32 void testGetSimbolo() {
33     assertEquals("EU", euro.getSimbolo(), "El símbolo debe ser EU.");
34 }
35
36 @Test
37 //Test para probar el GetFactorConversion de Dolar
38 void testGetFactorConversion() {
39     assertEquals(1003, euro.getFactorConversion(), "El factor de conversión debe ser 1003.");
40 }
41
42 @Test
43 //Test para probar método convertir pesos de Euro
44 void testConvertirPesoCL() {
45     double cantidadConvertida = 1003; // Cantidad de pesos chilenos a convertir
46     double cantidadConvertidaEsperada = 1;
47 }
```

Console Output:

```
<terminated> EuroTest (1) [JUnit] C:\Program Files\Java\jdk-17\bin\javaw.exe (10-05-2024 21:04:08 - 21:04:09) [pid: 12872]
Iniciando recursos para todos los tests.
Finalizando un test.
Iniciando un test.
Cantidad convertida: EU1.0 en pesos chileno: CLP1003.0
Finalizando un test.
Iniciando un test.
Cantidad convertida: CLP1003.0 en euros: EU1.0
Finalizando un test.
Iniciando un test.
Finalizando un test.
Limpiando recursos después de todos los tests.
```