

Lisa Ponsteen, i6397659

DataStorage tests

```
✓ DataStorageTest (data_management 2 sec 48 ms) ✓ Tests passed: 4 of 4 tests – 2 sec 48 ms
  ✓ testOrderingOfRecords() 1 sec 39 ms
  ✓ testAddPatientData() 1 ms
  ✓ testGetLastRecord() 1 sec 7 ms
  ✓ testDuplicateRecords() 1 ms

/Users/lisa/Library/Java/JavaVirtualMachines/corretto-23.0.2/Contents/Home/bin/java ...

Process finished with exit code 0
```

FileReader tests

```
FileReaderTest
  ✓ testReadFileAndParse (data_management 147 ms) ✓ Tests passed: 1 of 1 test – 147 ms

/Users/lisa/Library/Java/JavaVirtualMachines/corretto-23.0.2/Contents/Home/bin/java ...

Process finished with exit code 0
```

```
52 assertEquals( expected: 3, records.size()); // Check if 3 records are retrieved (so the white blood cells
53 assertEquals( expected: 100.0, records.get(0).getMeasurementValue()); // Validate first record
54 assertEquals( expected: 101.0, records.get(2).getMeasurementValue()); // Validate that the 3rd record is
55 assertEquals( expected: "alert", records.get(1).getRecordType()); //validate second record (the triggered
```

```
java
  data_management
    101 assertEquals( expected: 2, records.size()); // Check if the last 2 valid
    102 assertEquals( expected: 1, records.get(0).getPatientId()); // Validate re
    103 }

FileReaderTest.testEdgeCase
FileReaderTest (data_management 70 ms) ✓ Tests passed: 1 of 1 test – 70 ms
  ✓ testEdgeCase() 70 ms

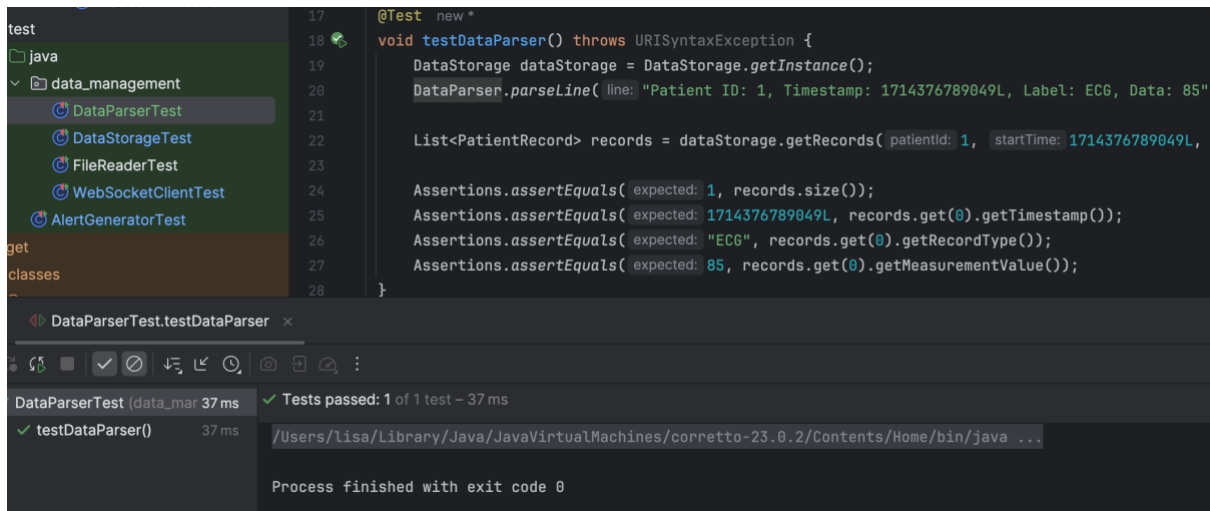
/Users/lisa/Library/Java/JavaVirtualMachines/corretto-23.0.2/Contents/Home/bin/java ...

did not parse timestamp correctly: -1
could not parse measurement value: 100patient1
Unknown data variable100bpm
did not parse measurement value correctly: NaN
did not parse measurement value correctly: NaN

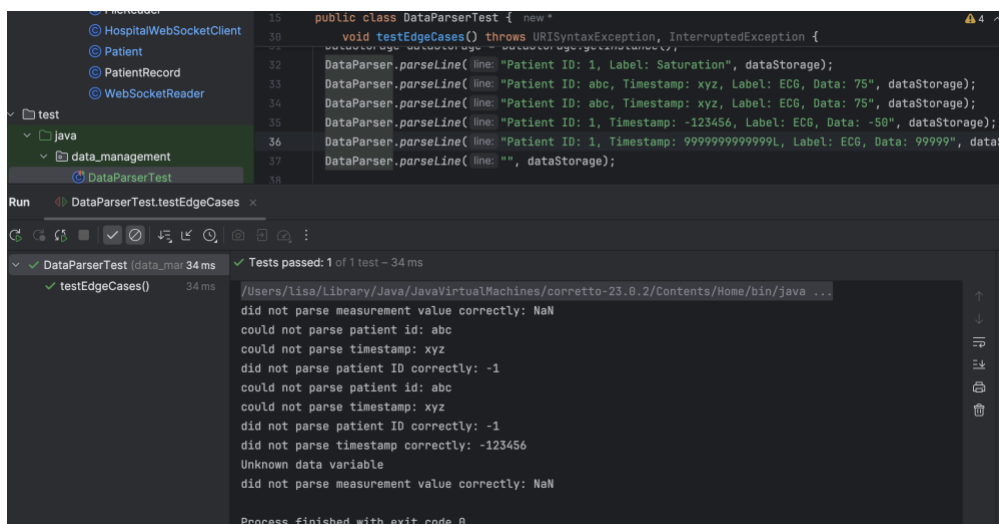
Process finished with exit code 0
```

DataParser tests

basic case:

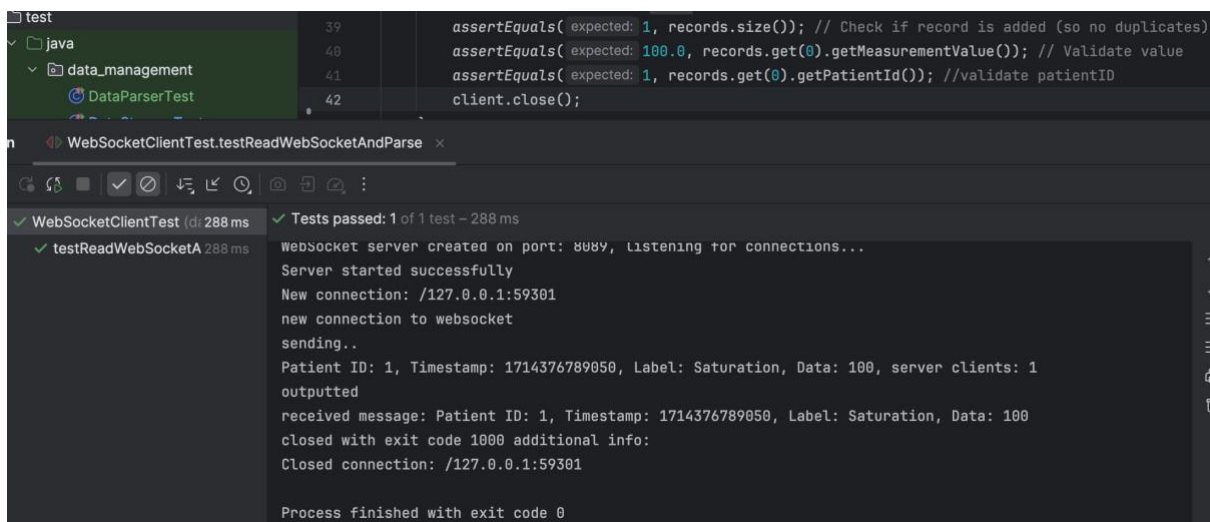


edge cases: invalid data doesn't stop the program



WebSocket tests

Base case ReadWebSocketAndParse:



Testing sending multiple records:

```

PatientRecord 58 out.output( patientId: 1, timestamp: 1714376789049L + 3, label: "BloodPressure", data: "130");
WebSocketReader 59
est 60
java 61 Thread.sleep( millis: 5000);
62 List<PatientRecord> records = storage.getRecords( patientId: 1, startTime: 1714376789049L, en
assertEquals( expected: 4, records.size()); //has it received 4 records

WebSocketClientTest.testSendMultipleRecords x

WebSocketClientTe 5 sec 274 ms ✓ Tests passed: 1 of 1 test – 5 sec 274 ms
✓ testSendMultiple 5 sec 274 ms Patient ID: 1, Timestamp: 1714376789049, Label: Saturation, Data: 99, server clients: 1
outputted
Patient ID: 1, Timestamp: 1714376789050, Label: Saturation, Data: 100, server clients: 1
outputted
Patient ID: 1, Timestamp: 1714376789051, Label: ECG, Data: 5, server clients: 1
outputted
Patient ID: 1, Timestamp: 1714376789052, Label: BloodPressure, Data: 130, server clients: 1
outputted
received message: Patient ID: 1, Timestamp: 1714376789049, Label: Saturation, Data: 99
received message: Patient ID: 1, Timestamp: 1714376789050, Label: Saturation, Data: 100
received message: Patient ID: 1, Timestamp: 1714376789051, Label: ECG, Data: 5
received message: Patient ID: 1, Timestamp: 1714376789052, Label: BloodPressure, Data: 130
closed with exit code 1000 additional info:
Closed connection: /127.0.0.1:59670

```

Test invalid data values:

```

test 86 // one message should be stored: the large number case
87 assertEquals( expected: 1, allRecords.size());
88 assertEquals( expected: 99999.0, allRecords.get(0).getMeasurementValue());
89

WebSocketClientTest.testInvalidData x

WebSocketClientTe 2 sec 290 ms ✓ Tests passed: 1 of 1 test – 2 sec 290 ms
✓ testInvalidData() 2 sec 290 ms received message: Patient ID: 1, Timestamp: -123456, Label: ECG, Data: -50
did not parse timestamp correctly: -123456
received message: Patient ID: 1, Timestamp: 9999999999999, Label: ECG, Data: 99999
received message: Patient ID: 1, Timestamp: 1714376789050, Label: Saturation, Data: sometext
could not parse measurement value: sometext
did not parse measurement value correctly: NaN
received message: Patient ID: 1, Timestamp: 0, Label: Saturation, Data: 0.0
did not parse timestamp correctly: 0
closed with exit code 1000 additional info:
Closed connection: /127.0.0.1:60082

Process finished with exit code 0

```

Testing invalid messages:

```
✓ WebSocketClientTe 1 sec 306 ms ✓ Tests passed: 1 of 1 test - 1 sec 306 ms
  ✓ testInvalidMessa 1 sec 306 ms
    received message: Patient ID: 1, Timestamp: 1714376789050, Label: Saturation, Data: 99
    received message: 1, 1714376789050, Saturation, 99
    Unknown data variable1
    Unknown data variable1714376789050
    Unknown data variableSaturation
    Unknown data variable99
    did not parse measurement value correctly: NaN
    received message: Patient ID: 1, Label: Saturation
    did not parse measurement value correctly: NaN
    received message: abc xyz ECG 75
    Unknown data variableabcxyzECG75
    did not parse measurement value correctly: NaN
    received message:
    Unknown data variable
    did not parse measurement value correctly: NaN
    received message: Patient ID: 1
    did not parse measurement value correctly: NaN
    received message: Patient ID: 1, Label: ECG, Timestamp: 1714376789051, Data: 80
    closed with exit code 1000 additional info:
    Closed connection: /127.0.0.1:60368

    Process finished with exit code 0
```

Integration test of data storage and parser, and websocket client, -reader and -outputstrategy. Test updating of duplicate data:

```
© Patient
© PatientRecord
© WebSocketReader

test
  java
    WebSocketClientTest.testDuplicateData

public class WebSocketClientTest {
    List<PatientRecord> records = storage.getRecords( patientId: 1, startTime: 1714376789050L, ...
    assertEquals( expected: 1, records.size()); // just one record in the list
    assertEquals( expected: 100.0, records.get(0).getMeasurementValue()); //record is updated
    client.close();
}

WebSocketClientTest.testDuplicateData
  ✓ WebSocketClientTe 5 sec 678 ms ✓ Tests passed: 1 of 1 test - 5 sec 678 ms
    ✓ testDuplicateDat 5 sec 678 ms
      Server started successfully
      New connection: /127.0.0.1:59712
      new connection to websocket
      Patient ID: 1, Timestamp: 1714376789050, Label: Saturation, Data: 99, server clients: 1
      outputted
      received message: Patient ID: 1, Timestamp: 1714376789050, Label: Saturation, Data: 99
      Patient ID: 1, Timestamp: 1714376789050, Label: Saturation, Data: 99, server clients: 1
      outputted
      received message: Patient ID: 1, Timestamp: 1714376789050, Label: Saturation, Data: 99
      Patient ID: 1, Timestamp: 1714376789050, Label: Saturation, Data: 100, server clients: 1
      outputted
      sending..
      received message: Patient ID: 1, Timestamp: 1714376789050, Label: Saturation, Data: 100
      closed with exit code 1000 additional info:
      Closed connection: /127.0.0.1:59712
```

Alert Generator tests

```
✓ AlertGeneratorTest 96 ms ✓ Tests passed: 1 of 1 test - 96 ms
  ✓ testMakeAlert() 96 ms
    /Users/Lisa/Library/Java/JavaVirtualMachines/corretto-23.0.2/Contents/Home/bin/java ...
    patientId=1, condition:BloodOxygenAlert -> Saturation=90.0, timestamp=1747914364876
    Alert should be triggered with timestamp: 1747914364876

    Process finished with exit code 0
```

```
test
├── java
│   └── data_management
│       ├── DataParserTest
│       ├── DataStorageTest
│       ├── FileReaderTest
│       ├── WebSocketClientTest
│       ├── AlertGeneratorTest
│       └── IntegrationTest
├── target
│   ├── classes
│   └── com
└── ...

31 @Test Lisa Ponsteen *
32 void testCombinedAlerts() {
33     DataStorage dataStorage = DataStorage.getInstance();
34
35     dataStorage.addPatientData( patientId: 1, measurementValue: 90, recordType: "Saturation", System
36     dataStorage.addPatientData( patientId: 1, measurementValue: 89, recordType: "SystolicPressure", S
37
38     AlertGenerator alertGenerator = new AlertGenerator(dataStorage);
39     alertGenerator.evaluateData( patientId: 1);
40
41     System.out.println("a CombinedAlert, BloodOxygenAlert and BloodPressureAlert should be tri
42 }
43
```

Run AlertGeneratorTest.testCombinedAlerts

AlertGeneratorTest 52 ms Tests passed: 1 of 1 test - 52 ms

testCombinedAle 52 ms

```
/Users/lisa/Library/Java/JavaVirtualMachines/corretto-23.0.2/Contents/Home/bin/java ...
patientId=1, condition:BloodOxygenAlert -> Saturation=90.0, timestamp=1747914447394
patientId=1, condition:BloodPressureAlert -> SystolicPressure=89.0, timestamp=1747914447397
patientId=1, condition:CombinedAlert -> SystolicPressure=89.0, timestamp=1747914447397
a CombinedAlert, BloodOxygenAlert and BloodPressureAlert should be triggered for patient 1
```

AlertGeneratorTest 42 ms Tests passed: 1 of 1 test - 42 ms

testAlertDecorators() 42 ms

```
/Users/lisa/Library/Java/JavaVirtualMachines/corretto-23.0.2/Contents/Home/bin/java ...
REPEATED Alert: [patientId=1, condition:Saturation=90, timestamp=1747919194269 ]
PRIORITY Alert: [patientId=1, condition:Saturation=90, timestamp=1747919194269]
a repeated alert and priority alert should be triggered for patient 1

Process finished with exit code 0
```

```
15 public class AlertGeneratorTest { Lisa Ponsteen *
16     @Test Lisa Ponsteen *
17     void triggeringAlertDecorator() throws InterruptedException {
18         DataStorage dataStorage = DataStorage.getInstance();
19         dataStorage.addPatientData( patientId: 1, measurementValue: 90, recordType: "Saturation", Syst
20         AlertGenerator alertGenerator = new AlertGenerator(dataStorage);
21         alertGenerator.evaluateData( patientId: 1);
22         Thread.sleep( millis: 3000);
23         dataStorage.addPatientData( patientId: 1, measurementValue: 89, recordType: "Saturation", Syst
24     }
25 }

AlertGeneratorTest.triggeringAlertDecorator

AlertGenerat 9 sec 65 ms Tests passed: 1 of 1 test - 9 sec 65 ms



triggering 9 sec 65 ms



```
/Users/lisa/Library/Java/JavaVirtualMachines/corretto-23.0.2/Contents/Home/bin/java ...
patientId=1, condition:BloodOxygenAlert -> Saturation=90.0, timestamp=1747919030198
PRIORITY Alert: [patientId=1, condition:BloodOxygenAlert -> Saturation=90.0, timestamp=1747919030198]
REPEATED Alert: [patientId=1, condition:BloodOxygenAlert -> Saturation=90.0, timestamp=1747919030198]
rechecking on value89.0
REPEATED Alert: [patientId=1, condition:BloodOxygenAlert -> Saturation=90.0, timestamp=1747919030198], repeated
rechecking on value88.0
REPEATED Alert: [patientId=1, condition:BloodOxygenAlert -> Saturation=90.0, timestamp=1747919030198], repeated
rechecking on value100.0
a priority alert should be triggered and a repeated alert that is repeated 2 times

Process finished with exit code 0
```


```