Feedback by Patrick Garvey

Technologies used :
Java : built and compiled with version 11.0.4 (the most recent LTS version)

Maven for building.
-POM dependencies include :
*Selenium for ui testing
*TestNG for test assertions and runs
*REST-Assured for api testing
*JSON-simple for json manipulation
*SLF4J for simple logging
* Some javax bind dependencies that are no longer bundled in later verions of javaSE (Now they are in javaEE only) but are needed to initiate rest-assured.

The project was built using the page object model. There is only one page in this assignemt but by using this model it allows for future growth.

The project can be run by right clicking and running the testNG.xml found in the resources folder.

The file *suite.properties* contains a variable for which browser you desire to run your test against (I've only tested firefox), and the url the application is running on. It also contains variables to specify where browser drivers are stored.

Data providers were used for some invalid scenario testing as it quickly allowed me to add additional tests.

A DateUtils class was made and heavily used for getting dates. Most dates used are calculated (plus or minus days and months) from the date of execution.

In rescources>browserDrivers there are 3.exe files. These are the drivers needed to run selenium 3 against each of the specified browser types. This is why the file size is large. I will highlight this in an email incase your security policy forbids downloading these exes.

SLF4j was used as a logging tool so as not to have constant System.outs

Page elements were mostly identified by className. Ideally location by id is preferred but there were'nt many ids on the page. Xpaths were used for buttons as they contained the same className.

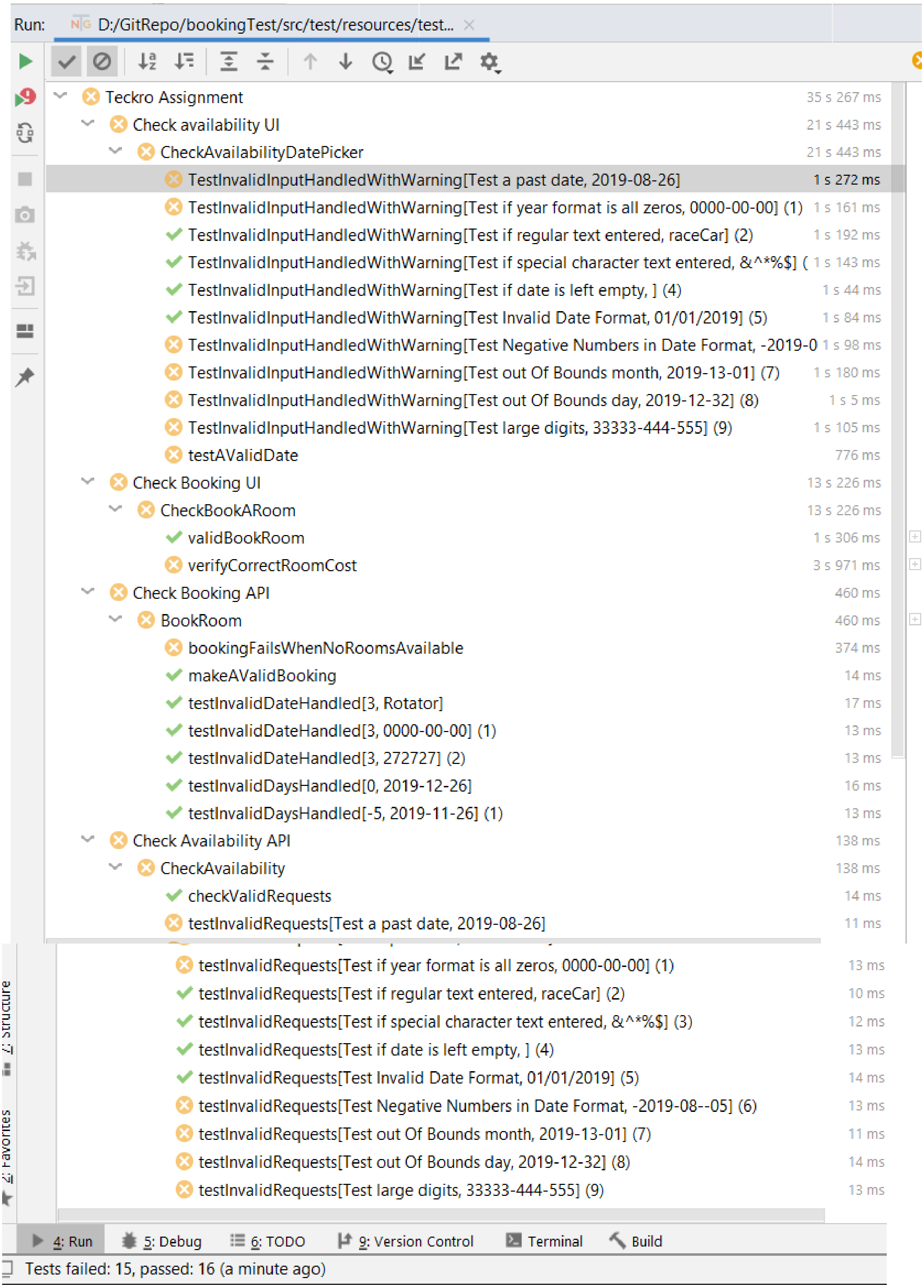
Suites are broken into 4 classes.
Check Availability UI
Check Availability API
Check Booking UI
Check Booking API
(See last page)

The tests.
Checking Availability of the room:
Entering a valid date in the correct YYYY-mm-dd format works as expected.
Esentially entering anything in the format <digit>-<digit>-<digit> is accepted as valid input.
Days of the month can go out of bounds (loops at 31, i.e 32 would return 01),
Month numbers can go out of bounds,
Year can be any number of digits,
Past dates can be entered,
Negative number dates can be entered,

Some cases that are handled correctly and throw a 400 error. These seem to include :
Entering text,
Entering special charaters (&^*%$),
Entering dates in another format i.e 2019/01/01
The input cannot be left empty.

Booking a room
tests and outcomes for the date picker were the same as the picker for availabilty.
Booking for a zero or negitive number of days is not prevented.
The cost of a stay is not calculated correctly if compared with how much a room a room costs individually each night.

Post execution :



```
⊗ Teckro Assignment                                                                              35 s 267 ms
  ⊗ Check availability UI                                                                        21 s 443 ms
    ⊗ CheckAvailabilityDatePicker                                                                21 s 443 ms
      ⊗ TestInvalidInputHandledWithWarning[Test a past date, 2019-08-26]                          1 s 272 ms
      ⊗ TestInvalidInputHandledWithWarning[Test if year format is all zeros, 0000-00-00] (1)      1 s 161 ms
      ✓ TestInvalidInputHandledWithWarning[Test if regular text entered, raceCar] (2)             1 s 192 ms
      ✓ TestInvalidInputHandledWithWarning[Test if special character text entered, &^*%$] (      1 s 143 ms
      ✓ TestInvalidInputHandledWithWarning[Test if date is left empty, ] (4)                       1 s 44 ms
      ✓ TestInvalidInputHandledWithWarning[Test Invalid Date Format, 01/01/2019] (5)               1 s 84 ms
      ⊗ TestInvalidInputHandledWithWarning[Test Negative Numbers in Date Format, -2019-0          1 s 98 ms
      ⊗ TestInvalidInputHandledWithWarning[Test out Of Bounds month, 2019-13-01] (7)             1 s 180 ms
      ⊗ TestInvalidInputHandledWithWarning[Test out Of Bounds day, 2019-12-32] (8)                  1 s 5 ms
      ⊗ TestInvalidInputHandledWithWarning[Test large digits, 33333-444-555] (9)                 1 s 105 ms
      ⊗ testAValidDate                                                                              776 ms
  ⊗ Check Booking UI                                                                             13 s 226 ms
    ⊗ CheckBookARoom                                                                             13 s 226 ms
      ✓ validBookRoom                                                                             1 s 306 ms
      ⊗ verifyCorrectRoomCost                                                                     3 s 971 ms
  ⊗ Check Booking API                                                                               460 ms
    ⊗ BookRoom                                                                                      460 ms
      ⊗ bookingFailsWhenNoRoomsAvailable                                                            374 ms
      ✓ makeAValidBooking                                                                            14 ms
      ✓ testInvalidDateHandled[3, Rotator]                                                           17 ms
      ✓ testInvalidDateHandled[3, 0000-00-00] (1)                                                    13 ms
      ✓ testInvalidDateHandled[3, 272727] (2)                                                        13 ms
      ✓ testInvalidDaysHandled[0, 2019-12-26]                                                        16 ms
      ✓ testInvalidDaysHandled[-5, 2019-11-26] (1)                                                   13 ms
  ⊗ Check Availability API                                                                          138 ms
    ⊗ CheckAvailability                                                                             138 ms
      ✓ checkValidRequests                                                                           14 ms
      ⊗ testInvalidRequests[Test a past date, 2019-08-26]                                            11 ms
      ⊗ testInvalidRequests[Test if year format is all zeros, 0000-00-00] (1)                        13 ms
      ✓ testInvalidRequests[Test if regular text entered, raceCar] (2)                               10 ms
      ✓ testInvalidRequests[Test if special character text entered, &^*%$] (3)                       12 ms
      ✓ testInvalidRequests[Test if date is left empty, ] (4)                                        13 ms
      ✓ testInvalidRequests[Test Invalid Date Format, 01/01/2019] (5)                                14 ms
      ⊗ testInvalidRequests[Test Negative Numbers in Date Format, -2019-08--05] (6)                  13 ms
      ⊗ testInvalidRequests[Test out Of Bounds month, 2019-13-01] (7)                                11 ms
      ⊗ testInvalidRequests[Test out Of Bounds day, 2019-12-32] (8)                                  14 ms
      ⊗ testInvalidRequests[Test large digits, 33333-444-555] (9)                                    13 ms

Tests failed: 15, passed: 16 (a minute ago)
```